Learning in the quantum universe

Thesis by
Hsin-Yuan Huang

In Partial Fulfillment of the Requirements for the
Degree of
Doctor of Philosophy

Caltech

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2024
Defended Aug. 17th, 2023

ACKNOWLEDGEMENTS

I feel incredibly lucky to be advised by John Preskill and Thomas Vidick. John is always supportive of my unconventional ideas and offers unique perspectives on the topics I am studying. I enjoy every meeting with John. Thomas has played a crucial role in sharpening my thinking and work by identifying critical issues in my thoughts, ideas, and proofs. I owe much of the clarity in my work to Thomas. When I was a first-year graduate student with a big ambition to build quantum artificial intelligence, despite having no background in quantum information and quantum computing, the willingness of John and Thomas to engage with me is invaluable.

I am extremely grateful to my long-time collaborators, John Preskill, Richard Kueng, Jarrod McClean, Michael Broughton, Jordan Cotler, Sitan Chen, and Jerry Li, for their significant contributions. Without their help, none of the results presented in this thesis would have been possible. I am especially thankful to John Preskill and Richard Kueng for working on so many exciting projects together. The discussions with my thesis committee members - Manuel Endres, Joel Tropp, and Fernando Brandao - were also instrumental in shaping my thesis.

I want to extend my sincere gratitude to Walter Kortschak for providing the Kortschak scholarship during my first two years of graduate school. This scholarship enabled me to explore research fields beyond my undergraduate studies and delve into a completely different field. I am also thankful to Yisong Yue and Anima Anandkumar for encouraging me to take this leap. Furthermore, the fellowship provided by J. Yang & Family Foundation during my third year and the Google Ph.D. fellowship during my last two years have been tremendously helpful.

I would also like to express my appreciation to Chun-Ju Wu and Amy Huang for keeping me company during the COVID pandemic with video games, binge-watching, and supermarket shopping. Finally, I am deeply indebted to my parents, Hsuan-Cheng Huang, Hsueh-Fen Juan, and my wife, Chi-Yun Cheng, for their unwavering emotional support. I cannot move through all the obstacles and arrive at where I am now without you.

ABSTRACT

In this thesis, I will present our progress in building a rigorous theory to understand how scientists, machines, and future quantum computers could learn models of our quantum universe. The thesis begins with an experimentally feasible procedure for converting a quantum many-body system into a succinct classical description of the system, its classical shadow. Classical shadows can be applied to efficiently predict many properties of interest, including expectation values of local observables and few-body correlation functions. I will then build on the classical shadow formalism to answer two fundamental questions at the intersection of machine learning and quantum physics: Can classical machines learn to solve challenging problems in quantum physics? And can quantum machines learn exponentially faster and predict more accurately than classical machines? The thesis answers both questions positively through mathematical analysis and experimental demonstrations.

PUBLISHED CONTENT AND CONTRIBUTIONS

Lewis, Laura et al. (2024). "Improved machine learning algorithm for predicting ground state properties". In: *nature communications* 15.1, p. 895. DOI: 10.1038/s41467-024-45014-7.
H.-Y. H. supervised the first author Laura Lewis (a former Caltech undergraduate) on this project. H.-Y. H. conceived the project, developed the key ideas for the proofs, and participated in the completion of the proofs and the writing of the manuscript.

Huang, Hsin-Yuan, Sitan Chen, and John Preskill (2023). "Learning to predict arbitrary quantum processes". In: *PRX Quantum* 4.4, p. 040337. DOI: 10.1103/PRXQuantum.4.040337.
H.-Y. H. conceived the project, developed the key ideas for the proofs, participated in the completion of the proofs, and wrote the majority of the manuscript.

Huang, Hsin-Yuan, Michael Broughton, Jordan Cotler, et al. (2022). "Quantum advantage in learning from experiments". In: *Science* 376.6598, pp. 1182–1186. DOI: 10.1126/science.abn7293.
H.-Y. H. conceived the project, conducted the numerical experiments, completed the theoretical analysis, participated in developing the key ideas and running the physical experiments, and writing the manuscript.

Huang, Hsin-Yuan, Richard Kueng, Giacomo Torlai, et al. (2022). "Provably efficient machine learning for quantum many-body problems". In: *Science* 377.6613, eabk3333. DOI: 10.1126/science.abk3333.
H.-Y. H. conceived the project, developed the key ideas for the proofs, completed most of the theoretical analysis and the proofs, and participated in the numerical experiments and the writing of the manuscript.

Chen, Sitan et al. (2021). "Exponential separations between learning with and without quantum memory". In: *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, pp. 574–585. DOI: 10.1109/FOCS52979.2021.00063.
The author list is ordered alphabetically (all authors contributed equally). H.-Y. H. participated in conceiving the project, developing the key ideas, completing the proofs, and writing the manuscript.

Huang, Hsin-Yuan, Michael Broughton, Masoud Mohseni, et al. (May 2021). "Power of data in quantum machine learning". In: *Nature Communications* 12.1, p. 2631. DOI: 10.1038/s41467-021-22539-9.
H.-Y. H. conceived the project, developed the key ideas for the proofs, completed the theoretical analysis and the proofs, and participated in the numerical experiments and the writing of the manuscript.

Huang, Hsin-Yuan, Richard Kueng, and John Preskill (July 2021a). "Efficient Estimation of Pauli Observables by Derandomization". In: *Phys. Rev. Lett.* 127 (3),

p. 030503. doi: 10.1103/PhysRevLett.127.030503.
H.-Y. H. conceived the project, developed the key ideas for the proofs, completed the numerical experiments, and participated in the theoretical analysis, the proofs, and the writing of the manuscript.

Huang, Hsin-Yuan, Richard Kueng, and John Preskill (2021b). “Information-theoretic bounds on quantum advantage in machine learning”. In: *Phys. Rev. Lett.* 126 (19), p. 190505. doi: 10.1103/PhysRevLett.126.190505.
H.-Y. H. conceived the project, developed the key ideas for the proofs, conducted the numerical experiments, completed the theoretical analysis, and participated in writing the manuscript.

– (2020). “Predicting many properties of a quantum system from very few measurements”. In: *Nature Physics*. doi: 10.1038/s41567-020-0932-7.
H.-Y. H. conceived the project, developed the key ideas for the proofs, conducted the numerical experiments, and participated in completing the proofs and writing the manuscript.

CONTENTS

LIST OF FIGURES

Number *Page*

NOMENCLATURE

n-qubit system. A qubit is the basic unit of quantum information, and describes the state of a quantum system with two levels (zero and one). An n-qubit system is a system with n qubits, where n is considered as the scaling parameter that can be very large. Any finite quantum system can be discretized and be represented by an n-qubit system.

Classical learning agent. An agent that can receive classical information from the world through experiments and measurements, process the classical information with classical computation, and store the classical information in a classical memory.

Classical shadow. Classical shadow is a succinct classical representation of a quantum state constructed from randomized measurements. The classical shadow representation enables efficient prediction of many properties of the quantum state, including expectation values of local observables, many-body fidelity, and few-body correlation functions.

Complexity. Complexity measures how hard, challenging, complex a problem, system, function is. For example, computational time complexity characterizes how much time is required to solve the problem computationally, sample complexity measures how many samples is needed to learn about an unknown object, state complexity inquires how many gates must be used to create a certain state.

CPTP map. CPTP map refers to a completely positive trace-preserving map over positive semidefinite matrices. Every possible process in the quantum world can be written as a CPTP map. The map captures the input-output relation between quantum states and maps an input state to an output state.

Distinguishing task. The problem of distinguishing between multiple (often finite) hypotheses from a collection of data obtained from experiments.

Empirical average. The average of the observed set of random numbers. Also called sample mean.

Generalization error. When an algorithm learns on a dataset, the performance it achieves on the dataset will generally be worse than it's performance on new inputs. The additional error incurred on new inputs is known as the generalization error.

Information-theoretic bounds. Information theory is a field that focuses on the information and disregards the computational cost. For example, given an unknown object, finding out its weight and volume is an information-theoretic task. However, given its weight and volume, figuring out its density is a computational task. Information-theoretic bounds provide upper and lower bounds on how much information is required to achieve a task.

Learning. Learning is an act of gathering information about an unknown entity (a distribution, a quantum system, a function, a quantum process, etc.), processing information through computation, and storing the processed information, such that, subsequently, one could achieve a certain task with the stored information.

Quantum advantage. Quantum machines are machines that operate under quantum-mechanical principles and are generalizations of classical machines. In some problems, quantum machines can address the problem strictly faster and/or strictly better than all classical machines. This phenomenon is called quantum advantage.

Quantum benchmarking. Quantum systems are hard to control and often are error-prone. Quantum benchmarking is a field of study to understand how to see if the quantum system has been engineered to perform according to our design.

Quantum information theory. Our universe is intrinsically quantum-mechanical. Hence information is fundamentally quantum. Quantum information theory characterizes what aspects of classical information still preserve in quantum, and what are the surprising properties that stem from the quantum nature of information.

Quantum learning machine. An agent that can receive quantum information from the world through quantum sensors, process the quantum information with quantum computation, and store the quantum information in a quantum memory.

Quantum many-body problems. Computational problems that arise from studying quantum many-body systems. Examples include solving ground states (finding the lowest energy state in a quantum many-body system) and identifying phases of matter.

Quantum many-body system. A system with many constituents that behaves quantum mechanically. The constituents could be qubits, electrons, atoms, photons, phonons, superconducting currents, etc.

Quantum process. Quantum process refers to all possible processes in the quantum world. A quantum process is represented by a CPTP map.

Quantum world. The macroscopic world that we experience every day is best described by the laws of classical mechanics. The microscopic world operates under a different set of laws, called quantum mechanics. The quantum world emphasizes the quantum nature of the microscopic world.

Part I

Introduction

Chapter 1

LEARNING IN THE QUANTUM UNIVERSE

1.1 Motivations

A central goal of science is to develop models that allow us to understand and make accurate predictions about the world around us. Predictive models created by humans and machines have enabled significant technological advancement. Because our universe is inherently quantum, understanding how to make predictions in the quantum world could lead to many advances, including the design of better catalysts, materials, and pharmaceuticals, novel insights into the behavior of exotic quantum matter, and the engineering of powerful quantum devices for computation, communication, and sensing.

In this thesis, I will describe our progress in building a mathematical foundation for understanding how scientists, machines, and future quantum computers could learn models of our inherently quantum universe. The mathematical foundation enables the discovery of new algorithmic tools that enhance one's ability to make predictions about the quantum world. By utilizing quantum information theory, learning theory, quantum complexity theory, high-dimensional probability, and quantum many-body physics, progress has been made in answering the following driving questions.

How to efficiently learn about complex quantum systems?

Given a quantum many-body system that corresponds to a newly engineered quantum device, an exotic quantum matter, or a synthetic molecule/material, the ability to learn properties or representations of the quantum system is central to the understanding of the system. For example, topological properties of an exotic phase of matter could enable us to discover new physical phenomena, and a representation of the device allows us to identify what needs to be improved. However, the intrinsic exponential complexity in a quantum system with n constituents makes learning and making predictions challenging. For example, traditional approaches (Hradil, 1997; O'Donnell and Wright, 2016; Haah et al., 2017) for learning a representation of an n-qubit system require exponential resources in n. This curse of dimensionality is unavoidable when one aims to construct a complete model of the system. Could one achieve much higher efficiency by considering effective models that make accurate predictions instead of insisting on a complete characterization of the system?

How can learning algorithms advance quantum technology and science?

There has been considerable interest in using classical machine learning (ML) algorithms to solve challenging problems in physical science, such as finding ground states in quantum many-body systems or classifying quantum phases of matter (Carleo and Troyer, 2017a; Carleo, Cirac, et al., 2019; Carrasquilla and Roger G Melko, 2017a). So far, these approaches are mostly heuristic. While shown to be effective in some intermediate-size experiments (Bohrdt et al., 2019; Rem et al., 2019; Torlai, Timar, et al., 2019), these methods are not backed by convincing theoretical arguments to ensure good performance, particularly for problem instances where traditional classical algorithms falter. Is it possible to develop rigorous learning algorithms for addressing physical problems? Could learning algorithms solve challenging problems that non-learning algorithms fail?

Could quantum machines predict more accurately than classical machines?

While classical computers have facilitated many profound advances in science and technology, they will ultimately fail to simulate every phenomenon in our quantum universe. The advent of quantum computers allows us to reach beyond classical computation (Arute et al., 2019). It is natural to wonder how much additional predictive power quantum computers will provide. When the exact model for a quantum evolution is available, quantum computers can simulate the dynamics and predict what would happen beyond the capability of classical computers (Lloyd, 1996; Childs et al., 2018). But what if the exact model is not known? Could quantum computers still learn to predict better than classical machines can? Quantum machines are not all-powerful, and there are known limitations in what they can learn (Regev, 2010; Arunachalam, Grilo, and Sundaram, 2019). Understanding both restrictions and advantages will allow us to design useful quantum machine learning models (Biamonte et al., 2017) and make full use of future quantum technology.

1.2 Overview

The thesis is divided into three parts. *Part I, Introduction* gives an overview of the thesis (Chapter 1), reviews key results in quantum information theory and learning theory (Chapter 2), and elucidates several key concepts uncovered by our works on learning about the quantum universe (Chapter 3). *Part II, Learning with classical machines* dives into our recent results studying the power of classical machines in learning about the quantum universe (Chapter 4, 5, and 6). *Part III, Learning with quantum machines* unravels the power of quantum machines in learning about the quantum universe (Chapter 7, 8, and 9). In the following, I will provide a summary of the results presented in this thesis.

Learning with classical machines

The wave function of an n-qubit system is a 2^n-dimensional complex vector. Hence, quantum many-body systems generally require exponentially-many classical bits to describe. The classical complexity of describing quantum systems suggests a significant challenge in learning and making predictions in the quantum world using classical machines. To learn about the quantum world efficiently, classical machines must be able to efficiently describe quantum systems. In this thesis, we propose a new representation of quantum systems, the *classical shadow representation*, that enables efficient predictions of many properties of any quantum many-body system (Chapter 4). Then, we will build on the *classical shadow representation* to show how one could design rigorous and efficient classical machine learning algorithms to solve challenging quantum many-body problems, including classifying quantum phases of matter and predicting ground states (Chapter 5). Finally, we will show how to generalize classical shadow representation to learn an efficient representation of any unitary/process generated by quantum many-body dynamics, even when the evolution time is arbitrarily long (Chapter 6).

Chapter 4: Predicting many properties of quantum systems

To address the exponential scaling in existing methods for learning a quantum many-body system, in an article (Huang, Richard Kueng, and Preskill, 2020) published in *Nature Physics*, Richard Kueng, John Preskill, and I developed a provably efficient algorithm for learning a succinct approximate classical representation of an unknown large-scale quantum system using very few measurements. This description, called a *classical shadow*, can be used to predict many different properties: order $\log M$ measurements suffice to accurately predict M functions of the state with high success probability. The number of measurements is independent of the system size. Moreover, the protocol allows one to specify target properties after the actual data acquisition phase (measurements) has been completed. In (Huang, Richard Kueng, and Preskill, 2020), we applied classical shadows to predict quantum fidelities, entanglement entropies, two-point correlation functions, expectation values of local observables, and the energy variance of many-body local Hamiltonians. We observed substantially higher prediction accuracy and lower computational time relative to previously known methods.

The core of this work is a versatile mathematical tool for analyzing random unitaries, known as unitary t-designs. The measurement procedure we performed contains a random quantum evolution followed by a computational basis measurement. The random quantum evolution scrambles the quantum information stored in the quantum system across the entire system, and a subsequent basis measurement can easily extract this information. The concept of unitary t-designs allowed us to capture and capitalize upon this intuition rigorously.

The practical efficiency of this algorithm led to multiple collaborations with experimentalists. In collaboration with Rainer Blatt's experimental group and Peter Zoller's theory group at the University of Innsbruck, we considered the ability of classical shadows to detect entanglement in a mixed quantum state. The result is a new entanglement certification protocol that reveals entanglement based on existing experimental data when its presence was previously unknown. The result led to an article (Andreas Elben, Richard Kueng, et al., 2020a) published in *Physical Review Letters*.

In collaboration with Jordan Cotler, Soonwon Choi, Hannes Pichler, and Manuel Endres' experimental group at Caltech, we applied the idea of randomized quantum evolution to benchmark a Rydberg atom system (J. Choi et al., 2021; J. S. Cotler et al., 2021). Due to the experimental limitation to performing accurate control of time-dependent quantum evolution, we can only perform a chaotic quantum evolution (that is not random). Our original theory (Huang, Richard Kueng, and Preskill, 2020) does not cover this setting. Although chaotic, the dynamics are inherently deterministic. However, with a modified procedure, we can accurately predict the fidelity of the Rydberg atom system under various circumstances (J. Choi et al., 2021). To build an intuition for why and when this works, we developed a new theoretical concept that relates t-designs and chaotic evolutions in (J. S. Cotler et al., 2021). This new benchmarking protocol (J. Choi et al., 2021) was recently accepted by *Nature*.

Chapter 5: Solving quantum many-body problems

Solving quantum many-body problems, such as finding ground states of quantum systems or predicting outcomes of quantum dynamics, has far-reaching consequences for physics, materials science, and chemistry. However, these problems are notoriously hard to solve using classical computers.

In collaboration with Google Quantum AI, we studied how classical machines can learn to solve quantum-mechanical problems using small training data sizes and efficient computational time. In an article (Huang, Broughton, Masoud Mohseni, Babbush, Boixo, Neven, and Jarrod R. McClean, 2021b) published in *Nature Communications*, we showed that the computational power of classical machines can be elevated by learning from classical data obtained in quantum experiments. Because the data are generated by the quantum universe, these data contain power beyond classical computation. The power of data enables classical machine learning algorithms that learn from the data to accomplish computational problems that are impossible to solve efficiently using classical algorithms. As a result, with the power of data, classical machine learning algorithms have the potential to address challenging quantum many-body problems that no classical algorithms can accomplish.

In a recent article (Huang, Richard Kueng, Torlai, et al., 2022) published in *Science*, John Preskill, Richard Kueng, Giacomo Torlai, Victor Albert, and I combine a series of ideas, including the insight that data provide computational power (Huang, Broughton, Masoud Mohseni, Babbush, Boixo, Neven, and Jarrod R. McClean, 2021b), the classical shadow formalism (Huang, Richard Kueng, and Preskill, 2020) for representing quantum systems on classical machines, mathematical tools in computational learning theory, and spectral flow formalism in mathematical physics, to give provably efficient classical machine learning (ML) algorithms for solving quantum many-body problems. We rigorously proved that after obtaining polynomial-size classical data from quantum experiments, the proposed polynomial-time algorithm could learn to predict ground state representations for new quantum many-body systems accurately. In contrast, we showed that under a widely-accepted complexity-theoretic conjecture, no classical polynomial-time algorithm without data could predict ground state properties as accurately as the ML algorithm trained with data.

Because the ground state of a physical system captures many of its fundamental properties, our result rigorously shows how scientists can use learning algorithms to address challenging physically-relevant problems, even for problem instances where traditional approaches falter. This work (Huang, Richard Kueng, Torlai, et al., 2022) provides a particularly interesting example showing the power of data and illustrates how classical machine learning can be helpful for the development of quantum technology and physical science. Furthermore, using a similar set of mathematical and algorithmic techniques combining classical shadow and kernel machine learning models, we also found in (Huang, Richard Kueng, Torlai, et al.,

2022) that classical machines can provably learn to classify a wide range of quantum phases of matter efficiently.

Chapter 6: Learning to predict quantum dynamics

We have seen that classical machines can learn to predict properties of quantum many-body systems, classify quantum phases of matter, and predict ground state properties. Another central problem in quantum mechanics is simulating complex quantum dynamics. In the last chapter on learning with classical machines, we focus on the problem of learning to predict quantum dynamics, a fundamental problem at the intersection of machine learning (ML) and quantum physics.

Given an unknown n-qubit completely positive trace-preserving (CPTP) map $\mathcal{E}$ that represents a quantum process happening in nature or in an experimental laboratory, we consider the task of learning to predict functions of the form

$$f(\rho, O) = \mathrm{tr}(O\mathcal{E}(\rho)), \tag{1.1}$$

where ρ is an n-qubit state and O is an n-qubit observable. Related problems arise in many fields of research, including quantum machine learning, variational quantum algorithms, machine learning for quantum physics, and quantum benchmarking. As an example, for predicting outcomes of quantum experiments (Huang, Richard Kueng, and Preskill, 2021; Melnikov et al., 2018; Huang, Broughton, J. Cotler, et al., 2022), we consider ρ to be parameterized by a classical input x, $\mathcal{E}$ is an unknown process happening in the lab, and O is an observable measured at the end of the experiment. Another example is when we want to use a quantum ML algorithm to learn a model of a complex quantum evolution with the hope that the learned model can be faster (Cirstoiu et al., 2020; Gibbs et al., 2022; Caro, Huang, Ezzell, et al., 2023).

Due to the exponential complexity encoded in an arbitrary CPTP map $\mathcal{E}$, all known works require an exponential number of data samples to guarantee a small constant error for predicting outcomes $\mathrm{tr}(O\mathcal{E}(\rho))$ in an arbitrary process $\mathcal{E}$ under a general input state ρ. While recent works (Caro, Huang, Marco Cerezo, et al., 2022; Caro, Huang, Ezzell, et al., 2023; Huang, Richard Kueng, and Preskill, 2021; Huang, Broughton, J. Cotler, et al., 2022) have shown that only a polynomial amount of data samples is required to learn $\mathrm{tr}(O\mathcal{E}(\rho))$ when $\mathcal{E}$ is restricted to being generated by a polynomial number of gates, these results still require exponential computation time. This raises the question of whether a classical ML algorithm can efficiently learn and predict an arbitrary quantum process.

In a recent work (Huang, Sitan Chen, and Preskill, 2023), Sitan Chen, John Preskill, and I answered this question in the affirmative by presenting a computationally-efficient classical ML algorithm that can learn a model of an arbitrary unknown n-qubit process $\mathcal{E}$, such that given ρ sampled from a wide range of distributions over arbitrary n-qubit states and any O in a physically-relevant class of observables, the ML algorithm can accurately predict $f(\rho, O) = \mathrm{tr}(O\mathcal{E}(\rho))$.

The proposed classical ML algorithm exhibits several surprising properties. First of all, the training and prediction of the proposed ML model are both efficient even if the unknown process $\mathcal{E}$ is an exponential-sized quantum circuit. This demonstrates the ability to compress any process into a succinct model through learning. Second of all, the computation is entirely classical apart from obtaining few-body reduced density matrices (RDMs) in ρ, which may require a quantum computer. Hence, if the RDMs can be obtained classically, then the proposed ML model is classical. Furthermore, the ML model can predict outcomes for highly entangled states ρ after learning from a training set that only contains data for random product states. This shows a form of strong generalization beyond what is seen in the training set.

Learning with quantum machines

Because our universe is inherently quantum, one would expect that quantum machines have a stronger learning and prediction ability compared to classical machines. In particular, one may hope that to learn some aspects of our quantum universe, a quantum learning machine could learn much faster and predict more accurately than a classical learning machine. However, it is not clear what problems quantum machines could demonstrate a significant quantum advantage. This viewpoint is also challenged by the existence of good classical ML algorithms for addressing challenging quantum many-body problems and predicting quantum many-body dynamics in the aforementioned works (Huang, Broughton, Masoud Mohseni, Babbush, Boixo, Neven, and Jarrod R. McClean, 2021b; Huang, Richard Kueng, Torlai, et al., 2022; Lewis et al., 2024; Huang, Sitan Chen, and Preskill, 2023). As a result, it is necessary to provide rigorous mathematical analysis to understand when significant quantum advantages in learning are possible or impossible.

In Chapter 7, we show that for a wide range of problems, a large quantum advantage in terms of the number of experiments is not possible due to an information-theoretic bound on quantum advantage. In Chapter 8, we revisit the role of data in elevating the computational power of classical machine learning algorithms and show that there are various problems where a quantum advantage in prediction performance is not possible. While the first two chapters focus on establishing impossibility results for significant quantum advantages, these impossibility results also carve out spaces where a large quantum advantage is possible. In Chapter 8, we give rigorous proofs and experiments demonstrating that for a set of learning problems, quantum machines can learn exponentially faster than classical machines.

Chapter 7: Information-theoretic bounds on quantum advantage

In this chapter, we focus on an important class of learning problems motivated by quantum mechanics.‘ Namely, we are interested in predicting functions of the form

$$f(x) = \mathrm{tr}(O\mathcal{E}(|x\rangle\langle x|)), \tag{1.2}$$

where x is a classical input, $\mathcal{E}$ is a completely positive and trace preserving (CPTP) map, and O is a known observable. Equation (1.2) encompasses *any* physical process that takes a classical input and produces a real number as output. The problem is to learn a function $h(x)$ that is approximately the same as $f(x)$ using as few accesses to $\mathcal{E}$ as possible.

A particularly important special case of setup (1.2) is training an ML model to predict what would happen in physical experiments (Melnikov et al., 2018). Such experiments might explore, for instance, the outcome of a reaction in quantum chemistry (Z. Zhou, Xiaocheng Li, and Zare, 2017), ground state properties of a novel molecule or material (Parr, 1980; Car and Parrinello, 1985; Becke, 1993; Steven R White, 1993a; Peruzzo et al., 2014; Kandala et al., 2017; Gilmer et al., 2017), or the behavior of neutral atoms in an analog quantum simulator (Buluta and Nori, 2009; Levine et al., 2018; Bernien et al., 2017). In these cases, the input x subsumes parameters that characterize the process, e.g., chemicals involved in the reaction, a description of the molecule, or the intensity of lasers that control the neutral atoms. The map $\mathcal{E}$ characterizes a quantum evolution happening in the lab. Depending on the parameter x, it produces the quantum state $\mathcal{E}(|x\rangle\langle x|)$. Finally, the experimentalist measures a certain observable O at the end of the experiment. The goal is to predict the measurement outcome for new physical experiments with new values of x that have not been encountered during the training process.

Motivated by these physical applications, we want to understand the power of classical and quantum ML models in learning functions of the form given in Equation (1.2). On the one hand, we consider classical ML models that can gather classical experimental data $\{(x_i, o_i)\}_{i=1}^{N_{\mathrm{C}}}$, where o_i is the outcome when we perform a POVM measurement on the state $\mathcal{E}(|x_i\rangle\langle x_i|)$. We denote by N_{C} the number of such experiments performed during training in the classical ML setting. On the other hand, we consider quantum ML models in which multiple runs of the CPTP map $\mathcal{E}$ can be composed coherently to collect quantum data, and predictions are produced by a quantum computer with access to the quantum data. We denote by N_{Q} the number of times $\mathcal{E}$ is used during training in the quantum setting.

We focus on the question of whether quantum ML models can have a large advantage over classical ML models: to achieve a small average prediction error, can the optimal N_{Q} in the quantum ML setting be much less than the optimal N_{C} in the classical ML setting? While one may expect that a large quantum advantage is possible since $\mathcal{E}$ is a quantum process, we found the contrary. In a manuscript (Huang, Richard Kueng, and Preskill, 2021) published in *Physical Review Letters*, Richard Kueng, John Preskill, and I proved that, for any $\mathcal{E}$, O, and $\mathcal{D}$, and for any quantum ML model, one can always design a classical ML model achieving a similar average prediction error such that N_{C} is larger than N_{Q} by at worst a small polynomial factor. Hence, there is no exponential quantum advantage in the number of experiments if the problem is to achieve a small average prediction error.

Chapter 8: Power of data and quantum advantage

While there is no large quantum advantage in the number of experiments to achieve a small average prediction error, there is still hope for quantum advantage in computational time. Even though a small amount of classical data contains sufficient information to identify $h(x)$, it may still be computationally hard to find $h(x) \approx f(x)$ with classical computers. There are various recent quantum ML proposals, such as quantum neural network (Farhi and Neven, 2018) and quantum kernel method (Havlicek et al., 2019). The justification that these quantum ML models exceed the capabilities of classical ML models typically follows from the conjecture that the quantum circuits involved in the quantum ML models are not classically simulatable.

In an article (Huang, Broughton, Masoud Mohseni, Babbush, Boixo, Neven, and Jarrod R. McClean, 2021b) published in *Nature Communications*, we showed that this picture is incomplete in a learning setting where some data is provided. This perspective connects to Chapter 5 and 6, where we show that the availability of data in a machine learning problem could elevate the classical ML models to accomplish computational problems that are hard for classical computers. The provided data can elevate classical ML models to rival quantum ML models, even when the underlying quantum problems are hard to solve classically. This chapter provides rigorous prediction error bounds for training classical and quantum ML methods based on kernel functions Cortes and Vapnik, 1995; Schölkopf, Alexander J Smola, Bach, et al., 2002; Mohri, Rostamizadeh, and Talwalkar, 2018; Jacot, Gabriel, and Hongler, 2018; Novak, L. Xiao, Hron, J. Lee, Alexander A Alemi, et al., 2019; Arora et al., 2019; Havlicek et al., 2019; Blank et al., 2020; Bartkiewicz et al., 2020; Y. Liu, Arunachalam, and Temme, 2020 to learn quantum mechanical models.

We use our prediction error bounds to devise a flowchart for testing potential quantum prediction advantage, the separation between prediction errors of quantum and classical ML models for a fixed amount of training data. Moreover, the application of these tools to existing quantum ML models in the literature rules many of them out immediately, providing a powerful sieve for focusing the development of new quantum ML algorithms. Following these constructions, in numerical experiments, we find that a variety of common quantum models in the literature perform similarly or worse than classical ML on both classical and quantum datasets. Quantum ML models involving quantum circuits that are hard to simulate classically are hence insufficient to guarantee a quantum advantage in learning problems.

Chapter 9: Quantum advantage in learning from experiments

Given so many impossibility results of quantum advantage, as shown in previous chapters, one may conclude that all hope is lost. However, the impossibility results also guide us to where the quantum advantage could lie. Throughout the study, we found that there are some learning problems in quantum physics that are not amenable to any of the impossibility results and are not efficiently addressable with any classical ML algorithms we could think of. An example of the learning problem goes back to the first chapter of *Learning with classical machines*. While classical shadow enables classical machines to predict many properties of a quantum many-body system efficiently, there is a class of properties that is not applicable to classical shadows. This is the class of Pauli observables $\{I, X, Y, Z\}^{\otimes n}$ in an unknown n-qubit system ρ. We originally thought that this is a weakness of classical shadow, but after thinking about this problem for some years, we are more and more convinced that this is a weakness of learning using classical machines.

In a manuscript (Huang, Richard Kueng, and Preskill, 2021) published in *Physical Review Letters*, Richard Kueng, John Preskill, and I showed an unconditional exponential lower bound for predicting all Pauli observables using classical machines. We considered classical ML models that can gather and process classical data by performing measurements on the unknown quantum system. And we proved that *all possible* classical ML models require at least $\Omega(2^n)$ experiments to predict all Pauli observables accurately. We also considered quantum ML models that can obtain quantum information about the quantum system using a quantum sensor and perform quantum data processing. And we explicitly constructed a quantum ML algorithm that uses only $O(n)$ experiments. This work shows that a quantum machine could achieve a learning task using exponentially fewer experiments than its classical counterpart.

To further understand the nature of the exponential quantum advantage, I collaborated with learning theorists Sitan Chen and Jerry Li and a high-energy physicist, Jordan Cotler, to construct a mathematical framework for establishing exponential separations in problem size between classical and quantum learning algorithms. We showed that such an exponential advantage is evident in many tasks, including predicting properties of quantum systems (shadow tomography) (Aaronson, 2018; Aaronson and Rothblum, 2019), classifying if a random quantum evolution preserves certain symmetry (e.g., time-reversal symmetry), testing the purity of a quantum state, etc. This led to a manuscript published in *FOCS 2021*.

In collaboration with Google Quantum AI, we built on this mathematical framework to show exponential quantum advantages in more tasks, including performing principal component analysis in a physical system (Lloyd, Masoud Mohseni, and Rebentrost, 2014), predicting output states of physical processes, and other tasks that could be readily achieved with near-term quantum computers. Conducting experiments with up to 40 superconducting qubits and 1300 quantum gates, we demonstrate that a substantial quantum advantage can be realized using today's relatively noisy quantum processors. Our results highlight how quantum technology can enable powerful new strategies to learn about nature and led to an article (Huang, Broughton, J. Cotler, et al., 2022) published in *Science*.

Chapter 2

PRELIMINARIES ON QUANTUM INFORMATION AND LEARNING THEORY

2.1 A brief review on quantum information theory

In this section, we review some relevant definitions and basic results in quantum information theory, which are leveraged throughout our problem statements and proofs. Specifically, we will discuss quantum processes, which are a general mathematical formalism for describing physical processes, and positive operator-valued measures (POVMs), which encompass all possible physical measurements. Readers familiar with these concepts can skip this section.

Definition and properties of quantum processes

For concreteness, let us consider a Hilbert space $\mathcal{H}_S \simeq \mathbb{C}^d$. Here the subscript S stands for 'system' since the Hilbert space describes the space of states of some particular system we wish to study. Given a density matrix ρ on this Hilbert space, we might ask: how can it evolve in time? The Schrödinger equations tell us that a state can evolve via unitary time evolution, and as such a density matrix can evolve by $\rho \mapsto U\rho U^\dagger$. However, there is a more general type of time evolution allowed by quantum mechanics. Suppose that we append our Hilbert space by another $\mathcal{H}_E \simeq \mathbb{C}^{d'}$ which describes an external environment. The joint Hilbert space is then the tensor product $\mathcal{H}_S \otimes \mathcal{H}_E$. We can imagine having an initial state $\rho_S \otimes \rho_E$ which factorizes between the system and environment, and then evolving the state by a unitary on the joint Hilbert space which couples the system and environment: $\rho_S \otimes \rho_E \mapsto U_{SE}(\rho_S \otimes \rho_E)U_{SE}^\dagger$. If we only have access to $\mathcal{H}_S$, then our knowledge of $U_{SE}(\rho_S \otimes \rho_E)U_{SE}^\dagger$ is described by performing a partial trace over the environment, namely $\text{tr}_E\left(U_{SE}(\rho_S \otimes \rho_E)U_{SE}^\dagger\right)$. As such, if we are only aware of the initial density matrix ρ_S on $\mathcal{H}_S$, then only having access to the system Hilbert space $\mathcal{H}_S$ the time evolution would appear to be

$$\rho_S \longmapsto \text{tr}_E\left(U_{SE}(\rho_S \otimes \rho_E)U_{SE}^\dagger\right) . \tag{2.1}$$

Note that, viewed as time evolution on ρ_S alone, the above map is not unitary. This is because information in ρ_S can leak into the environment, and similarly, information from the environment can influence the state of our system S of interest. This mapping is an example of a quantum process, which can be more compactly notated as $\rho_S \mapsto C[\rho_S]$. Here, C is a map from density matrices on $\mathcal{H}_S$ to (other) density matrices on $\mathcal{H}_S$. We visualize this dynamical process in Supp. Fig. 2.1.

Our quantum process C has two properties that are worth highlighting:

1. *C is trace-preserving*. This means that $\text{tr}(C[\rho_S]) = \text{tr}(\rho_S)$. The equality follows from the definition of $C[\rho]$ via the right-hand side of (2.1), since

$$\text{tr}(C[\rho_S]) = \text{tr}_S\Big(\text{tr}_E\Big(U_{SE}(\rho_S \otimes \rho_E)U_{SE}^\dagger\Big)\Big) = \text{tr}\Big(U_{SE}(\rho_S \otimes \rho_E)U_{SE}^\dagger\Big) \tag{2.2}$$

$$= \text{tr}(\rho_S \otimes \rho_E) = \text{tr}(\rho_S)\,\text{tr}(\rho_E) = \text{tr}(\rho_S)\,, \tag{2.3}$$

where we have used the cyclicity of the trace to cancel U_{SE} with $U_{SE}^\dagger$, and have also leveraged $\text{tr}(\rho_E) = 1$.

2. *C is completely positive*. Suppose we append our system Hilbert space $\mathcal{H}_S$ by ancillas $\mathcal{H}_A$ to arrive at the joint Hilbert space $\mathcal{H}_A \otimes \mathcal{H}_S$. Then complete positivity means that for any density matrix ρ_{AS} on this joint system (and for any choice of ancilla Hilbert space), $(\text{Id}_A \otimes C)[\rho_{AS}]$ is positive-semidefinite; here Id_A acts as the identity on the ancillas. To see why this property holds, we can write out $(\text{Id}_A \otimes C)[\rho_{AS}]$ more explicitly:

$$(\text{Id}_A \otimes C)[\rho_{AS}] = \text{tr}_E\Big((I_A \otimes U_{SE})(\rho_{AS} \otimes \rho_E)(I_A \otimes U_{SE}^\dagger)\Big)\,.$$

Since the right-hand side is merely performing a unitary transformation on the density matrix $\rho_{AS} \otimes \rho_E$ and then tracing out a subsystem (i.e. the environment subsystem), positive semi-definiteness is preserved.

We have thus shown that our C is a completely positive, trace-preserving (CPTP) linear map from density matrices on $\mathcal{H}_S$ to density matrices on $\mathcal{H}_S$. Henceforth, when we refer to a map as being CPTP, we will implicitly suppose that the map is linear. Moreover, we will interchangeably call a CPTP map a quantum process.

What is not immediately obvious is the following fact:

Theorem 1 (Stinespring dilation)**.** *Any CPTP map C taking density matrices on $\mathcal{H}_S \simeq \mathbb{C}^d$ to density matrices on $\mathcal{H}_S \simeq \mathbb{C}^d$ can be written in the form*

$$C[\rho_S] = \text{tr}_E\Big(U_{SE}(\rho_S \otimes \rho_E)U_{SE}^\dagger\Big)$$

for any ρ_S, where U_{SE}, ρ_E, and the dimension of the environment d' are all fixed.

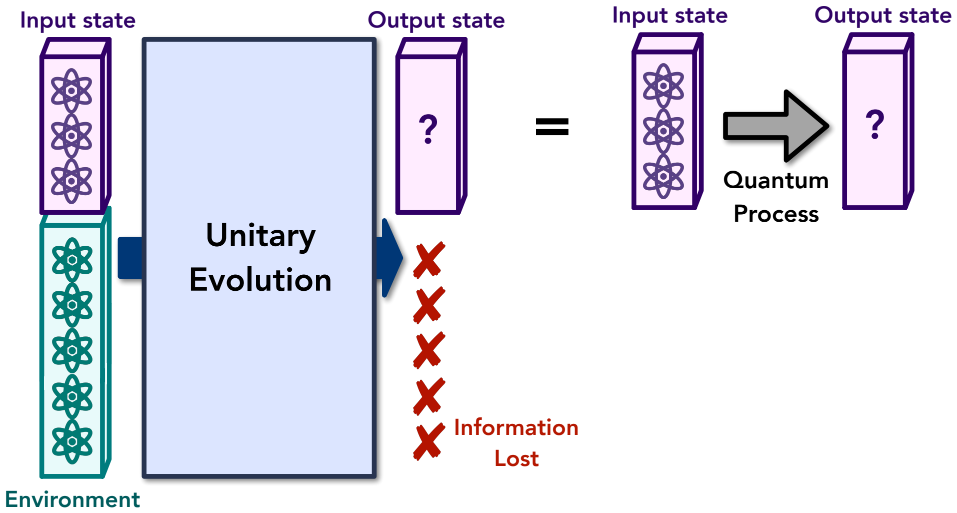


Figure 2.1: *Illustration of quantum processes: a formalism for describing physical processes.* Quantum process is also known as quantum operation or quantum dynamical map, and is often referred to as quantum channel in quantum communication theory.

This theorem means that any CPTP map on a density matrix can be realized as a unitary operation on a larger system, i.e. coupling the density matrix to an appropriate environment and evolving the joint state and ultimately tracing out the environment. In this sense, a quantum process is the most general form of evolution of a density matrix. Note that a special case of a quantum process is simply a unitary channel, i.e. $C[\rho] = U\rho U^\dagger$. A way of summarizing the above Theorem is that a quantum process that is not a unitary channel can be thought of as implementing open system dynamics.

Definition and properties of POVMs

The most conventional way to measure a quantum state $|\psi\rangle$ is by decohering it with respect to a complete orthonormal basis. More specifically, suppose that $|\psi\rangle \in \mathbb{C}^d$ and we choose some complete orthonormal basis $\{|i\rangle\}_{i=0}^{d-1}$ of $\mathbb{C}^d$. Then upon measuring $|\psi\rangle$ with respect to this basis, we will measure $|\psi\rangle$ to be in the state $|i\rangle$ with probability $\text{Prob}(i) = |\langle i|\psi\rangle|^2$. Analogously for a density matrix ρ on the same Hilbert space, if we measure it with respect to the same orthonormal basis we will measure the state to be $|i\rangle\langle i|$ with probability $\text{Prob}(i) = \text{tr}(|i\rangle\langle i|\,\rho)$.

There is a nice way of conceptualizing measurements which will admit useful generalizations. First, let us develop some notation. We define $\Pi_i = |i\rangle\langle i|$ which is the projector onto state $|i\rangle$, and will speak of the collection of projectors $\{\Pi_i\}_{i=0}^{d-1}$. It is readily seen that $\sum_{i=0}^{d-1} \Pi_i = I$ since this is just a resolution of the identity. Observe that each Π_i is Hermitian and positive semi-definite. Now suppose we append to our Hilbert space another copy $\mathbb{C}^d$. Then we can define a unitary on both copies which acts by

$$U\big(|\psi\rangle \otimes |0\rangle\big) = \sum_{i=0}^{d-1} \Pi_i |\psi\rangle \otimes |i\rangle \tag{2.4}$$

for any $|\psi\rangle$. Note that, as required of a unitary,

$$\big(\langle\psi| \otimes \langle 0|\big) U^\dagger U \big(|\psi\rangle \otimes |0\rangle\big) = \sum_{i,j=0}^{d-1} \langle\psi|\Pi_i \Pi_j|\psi\rangle\langle i|j\rangle$$
$$= \sum_{i=0}^{d-1} \langle\psi|\Pi_i^2|\psi\rangle = \sum_{i=0}^{d-1} \langle\psi|\Pi_i|\psi\rangle = 1$$

on account of $\Pi_i^2 = \Pi_i$ and $\sum_{i=0}^{d-1} \Pi_i = I$. Given the right-hand side of (2.4), we can make a measurement on the appended Hilbert space in the $\{|i\rangle\}_{i=0}^{d-1}$ basis; we will then measure the appended register to be in the state $|i\rangle$ with probability

$$\text{Prob}(i) = \big(\langle\psi| \otimes \langle 0|\big) U^\dagger \big(I \otimes |i\rangle\langle i|\big) U \big(|\psi\rangle \otimes |0\rangle\big) = \text{tr}(\Pi_i |\psi\rangle\langle\psi|) = |\langle i|\psi\rangle|^2 \,. \tag{2.5}$$

Similarly, if we consider $\rho \otimes |0\rangle\langle 0|$, conjugate by U, and then measure the state of the ancilla, the probability of measuring the ancilla to be $|i\rangle$ is $\text{Prob}(i) = \text{tr}(\Pi_i \rho) = \text{tr}(|i\rangle\langle i|\, \rho)$.

We can think about the above in terms of the following procedure. First we prepare a state $|\psi\rangle$; then we bring in an ancilla $|0\rangle$ and cause the two states to interact such that the ancilla goes into a state $|i\rangle$ upon coupling with the $|i\rangle$-component of $|\psi\rangle$. This leads to the right-hand side of (2.4). The ancilla can be thought of as a proxy for the readout of a measurement apparatus: upon reading off the value of $|i\rangle$, we are informed that the state $|\psi\rangle$ has been projected into its $|i\rangle$-component.

This type of procedure can be generalized as follows. Suppose we have a set of N $d \times d$ operators $\{M_i\}_{i=0}^{N-1}$ satisfying the completeness relation $\sum_{i=0}^{N-1} M_i^\dagger M_i = I$. Let us append to our Hilbert space $\mathbb{C}^d$ and ancillary Hilbert space $\mathbb{C}^N$ with complete orthonormal basis $\{|i\rangle\}_{i=0}^{N-1}$. Then we can consider a unitary map

$$U\big(|\psi\rangle \otimes |0\rangle\big) = \sum_{i=0}^{N-1} M_i |\psi\rangle \otimes |i\rangle \,. \tag{2.6}$$

The fact that $\big(\langle\psi| \otimes \langle 0|\big) U^\dagger U \big(|\psi\rangle \otimes |0\rangle\big) = 1$ can be checked using the completeness relation $\sum_{i=0}^{N-1} M_i^\dagger M_i = I$. Now if we measure the ancilla with respect to the

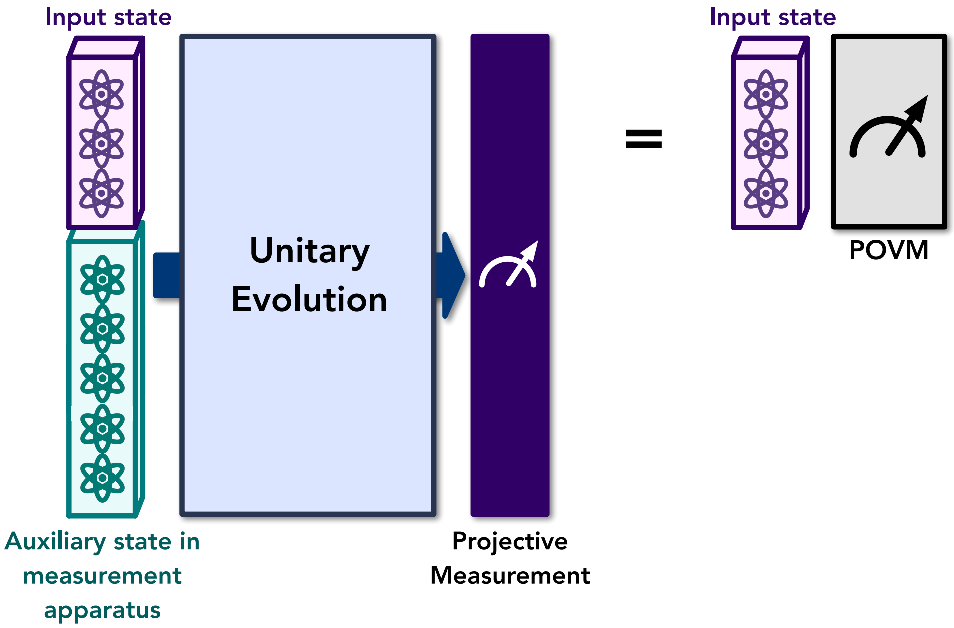


Figure 2.2: *Illustration of POVM: a formalism encompassing all physical measurements.* POVM considers a composition of the input state with an auxiliary state in the measurement apparatus (which can be thought of as a set of ancilla qubits) that undergoes an unitary evolution, followed by a projective measurement.

$\{|i\rangle\}_{i=0}^{N-1}$ basis, then we will measure the ancilla to be in the state $|i\rangle$ with probability $\text{Prob}(i) = |M_i|\psi\rangle|^2$. If we performed an analogous procedure at the level of density matrices, namely starting with a state $\rho \otimes |0\rangle\langle 0|$, conjugating both sides by U, and then measuring the ancilla in the $\{|i\rangle\}_{i=0}^{N-1}$ basis, we would measure the ancilla to be $|i\rangle$ with probability $\text{Prob}(i) = \text{tr}(M_i^\dagger M_i \rho)$. We visualize the above procedure in Supp. Fig. 2.2.

We can abstract this procedure into what is called a *positive operator-valued measure* (POVM):

Definition 1 (POVM)**.** *A POVM is a set Hermitian, positive semi-definite operators* $\{F_i\}_{i=0}^{N-1}$ *on* $\mathbb{C}^d$ *satisfying the completeness relation* $\sum_{i=0}^{N-1} F_i = I$*. A POVM measurement is a procedure in which, given a state* ρ *on* $\mathbb{C}^d$*, an ancillary measurement apparatus registers the index* i *with probability* $\text{tr}(F_i \rho)$.

This relates to our previous procedure as follows. We simply decompose $F_i = M_i^\dagger M_i$ (say, by a Cholesky decomposition) and perform the procedure previously stated with the M_i's.

We remark that the term 'measure' is used above in two distinct ways. When we speak of a POVM, the M means measure in the sense of measure theory, since we can think of $\{F_i\}_{i=0}^{N-1}$ as comprising a type of discrete measure on the space of operator on $\mathbb{C}^d$. Otherwise, we use 'measure' in the sense of measurement.

A useful fact is that given a POVM $\{F_i\}_{i=0}^{N-1}$, we can refine it into another, larger POVM $\{F_{i,j}\}_{i=0,j=0}^{N-1,d-1}$ such that (1) each $F_{i,j}$ is rank-1, and (2) a POVM measurement of $\{F_{i,j}\}_{i=0,j=0}^{N-1,d-1}$ can simulate a POVM measurement of $\{F_i\}_{i=0}^{N-1}$. Let us explain this construction. Since each F_i is a positive semi-definite Hermitian operator, we can diagonalize each operator as $F_i = \sum_{j=0}^{d-1} \lambda_j^{(i)} |v_j^{(i)}\rangle\langle v_j^{(i)}|$. Then let $F_{i,j} := \lambda_j^{(i)} |v_j^{(i)}\rangle\langle v_j^{(i)}|$ which is manifestly positive semi-definite, Hermitian, and rank-1; it is also clear that $\sum_{i=0}^{N-1} \sum_{j=0}^{d-1} F_{i,j} = \sum_{i=0}^{N-1} F_i = I$. We can use a POVM measurement of $\{F_{i,j}\}_{i=0,j=0}^{N-1,d-1}$ to simulate a POVM measurement of $\{F_i\}_{i=0}^{N-1}$ by simply summing measurement results:

$$\sum_{j=0}^{d-1} \mathrm{tr}(F_{i,j}\rho) = \mathrm{tr}(F_i\rho)\,. \tag{2.7}$$

Accordingly, we can, without loss of generality, choose to work with rank-1 POVMs, since we can use these to simulate any other POVMs.

2.2 A brief review on statistical learning theory

Statistical learning theory provides indispensable tools to understand our ability and inability to learn. The central theme of statistical learning theory is to understand how learning can be achieved given a collection of data. In particular, given some random set of data, how to infer the underlying hidden object or mechanism that generates the data. Statistical learning theory is closely related to high-dimensional probability, which studies the pattern that arises in a collection of random objects. In statistical learning theory, random objects are often considered to be the data collected from experiments. The data are often assumed to have inherent randomness that emerges from uncontrollable factors in the process of generating the data. By studying the pattern that arises in a collection of data, we can understand if the data can tell us about the underlying object or mechanism.

Concentration inequality

In probability theory, an important concept for describing the pattern emerging from a collection of random objects is called concentration inequality. The simplest form of concentration inequality is called Hoeffding's inequality. Hoeffding's inequality says that when we have a collection of n random numbers, the average of the n

numbers will be close to the true average of the random number. Furthermore, the probability that the average is not close decays exponentially with the distance t^2.

Theorem 2 (Hoeffding's inequality)**.** *Consider $X_1, \ldots, X_n$ to be independent random numbers that take values between* 0 *and* 1*. We have*

$$\Pr\left[\left|\frac{1}{n}\sum_i X_i - \mathbb{E}\, X_i\right| \geq t\right] \leq 2\exp\left(-2nt^2\right), \tag{2.8}$$

for any $t > 0$.

In terms of learning, this inequality states that by taking a dataset of n random numbers, we can learn the true average of the random number up to ϵ error with high probability by taking the empirical average, i.e., the average over the dataset, when the dataset size n is of order $1/\epsilon^2$. This statement may seem somehow trivial and one may think that this would always hold. However, this intuition is not correct because there is actually a condition on Hoeffding's inequality, i.e., the numbers must be between 0 and 1. For example, if these random numbers can be extremely large, then taking the empirical average of a moderate-size dataset does not guarantee that we will learn the true average. Hence, Hoeffding's inequality already tells us something about when the true average of a random number can be learned.

Concentration inequalities of random numbers (formally called random variables) enable one to prove more sophisticated concentration inequalities. In learning theory, one often wants to have concentration inequalities for all functions in a family of functions $\mathcal{G}$. To be more concrete, consider a data generation process that samples an independent random object z. The random object z could be an image, a high-dimensional vector, or a measurement outcome from physical experiments. Consider each function $g_j \in \mathcal{G}$ to be a test for a possible hypothesis h_j, where $g_j(z)$ is close to zero if z is likely generated from the hypothesis h_j, otherwise $g_j(z)$ is close to one. Then, the real number

$$\mathbb{E}_z[g_j(z)] \tag{2.9}$$

tells us whether hypothesis h_j gives rise to the random object z (closer to zero means h_j is more likely it gives rise to z). Hence, a simple learning algorithm would be to find a hypothesis h_j such that $\mathbb{E}_z[g_j(z)]$ is the lowest possible.

The problem is that we don't have access to the true probability distribution over z. Hence, we do not know what $\mathbb{E}_z[g_j(z)]$ is. By conducting multiple rounds of experiments, we can obtain a dataset of independent and identically distributed (i.i.d.) random objects $z_1, \ldots, z_N$. This dataset enables us to evaluate the empirical average

$$\frac{1}{N}\sum_{i=1}^{N} g_j(z_i), \tag{2.10}$$

which may or may not be close to the true average $\mathbb{E}_z[g_j(z)]$. This raises the following questions. When we see that the empirical average for g_j is close to zero, can we trust that the true average for g_j is close to zero? Even if we can trust the empirical average for a single g_j, can we trust it for all possible tests $g \in \mathcal{G}$? Could it be that the hypothesis $h^{\#}$ with the lowest $\frac{1}{N}\sum_{i=1}^{N} g^{\#}(z_i)$ is very different from the hypothesis h^* with the smallest $\mathbb{E}_z[g^*(z)]$?

Intuitively, if there are too many diverse tests g in the set $\mathcal{G}$, the empirical average of some tests can deviate too far from the true average due to random fluctuations. In statistical learning theory, one approach to characterize the vague concept of "diversity" in $\mathcal{G}$ is through Rademacher complexity,

$$\mathbb{E}_\sigma\left[\sup_{g\in\mathcal{G}} \frac{1}{N}\sum_{i=1}^{N} \sigma_i g(z_i)\right], \tag{2.11}$$

where $\sigma_1, \ldots \sigma_N$ are independent and uniform random variables over ± 1. The smaller the Rademacher complexity is, the less diverse the set $\mathcal{G}$ is, and hence the empirical average will be closer to the true average for all the tests. The concentration inequality is given by the following theorem, which shows that for all $g \in \mathcal{G}$, the true average is not too different from the empirical average. Because the inequality tells us how the test generalizes from a finite dataset to the unknown true distribution, one often refer to this as the generalization error bound.

Theorem 3 (See Theorem 3.3 in (Mohri, Rostamizadeh, and Talwalkar, 2018))**.** *Let $\mathcal{G}$ be a family of function mappings from a set $\mathcal{Z}$ to $[0, 1]$. Then for any $\delta > 0$, with probability at least $1 - \delta$ over N i.i.d. samples from $\mathcal{Z}$: $z_1, \ldots, z_N$, we have*

$$\mathbb{E}_z[g(z)] \le \frac{1}{N}\sum_{i=1}^{N} g(z_i) + 2\mathbb{E}_\sigma\left[\sup_{g\in\mathcal{G}} \frac{1}{N}\sum_{i=1}^{N} \sigma_i g(z_i)\right] + 3\sqrt{\frac{\log(2/\delta)}{2N}}, \tag{2.12}$$

for all $g \in \mathcal{G}$, where $\sigma_1, \ldots \sigma_N$ are i.i.d. uniformly random ± 1.

Characterizing the Rademacher complexity for a class of possible hypotheses is the key challenge in various problems we will consider in this thesis. For example, we will show in Chapter 8 and Chapter 5 that the Rademacher complexity is not too large for classifying quantum phases of matter and for predicting efficiently-learnable quantum machine learning models.

Rademacher complexity is a powerful tool for many learning problems. However, sometimes, a simpler solution would also work. For example, suppose that the family of functions $\mathcal{G}$ is a finite set. In this case, we can use Hoeffding's inequality to see that for any $g \in \mathcal{G}$,

$$\Pr\left[\mathbb{E}_z[g(z)] > \frac{1}{N}\sum_{i=1}^{N} g(z_i) + t\right] \leq 2\exp\left(-2Nt^2\right). \tag{2.13}$$

Hence, by union bound, we have

$$\Pr\left[\exists g \in \mathcal{G}, \mathbb{E}_z[g(z)] > \frac{1}{N}\sum_{i=1}^{N} g(z_i) + t\right] \leq \sum_{g\in\mathcal{G}} 2\exp\left(-2Nt^2\right). \tag{2.14}$$

This immediately leads to the following theorem based on the cardinality of $\mathcal{G}$.

Theorem 4 (Generalization error from the size of $\mathcal{G}$)**.** *Let $\mathcal{G}$ be a family of functions from a set $\mathcal{Z}$ to $[0,1]$. Then for any $\delta > 0$, with probability at least $1-\delta$ over N i.i.d. samples from $\mathcal{Z}$: $z_1, \ldots, z_N$, we have*

$$\mathbb{E}_z[g(z)] \leq \frac{1}{N}\sum_{i=1}^{N} g(z_i) + \sqrt{\frac{\log(|\mathcal{G}|/\delta)}{2N}} \tag{2.15}$$

for all $g \in \mathcal{G}$.

We can see that the Rademacher complexity is not present, but a cardinality $|\mathcal{G}|$ appears. This generalization error bound is typically worse than the Rademacher complexity but is very easy to obtain. Of course, one could immediately see that the inequality becomes useless when $\mathcal{G}$ contains infinitely many functions.

There is also a simple method to obtain a generalization error bound when the family $\mathcal{G}$ contains infinitely many functions. The idea is to obtain an ϵ covering net $\mathcal{N}_\epsilon(\mathcal{G})$, which is defined as the smallest subset of $\mathcal{G}$, such that

$$\forall g \in \mathcal{G}, \exists g' \in \mathcal{N}_\epsilon(\mathcal{G}), \|g - g'\| := \sup_{z\in\mathcal{Z}} |g(z) - g'(z)| < \epsilon. \tag{2.16}$$

Intuitively, for any function g in $\mathcal{G}$, there is a function g' in $\mathcal{N}_\epsilon(\mathcal{G})$, such that the function g' behaves similarly to g. In any compact and infinite set $\mathcal{G}$, the covering net $\mathcal{N}_\epsilon(\mathcal{G})$ will be finite. Hence, we can use the covering net to obtain the following generalization error bound.

Theorem 5 (Generalization error from the covering net)**.** *Let $\mathcal{G}$ be a family of functions from a set $\mathcal{Z}$ to $[0,1]$. Then for any $\delta > 0$, with probability at least $1-\delta$ over N i.i.d. samples from $\mathcal{Z}$: $z_1, \ldots, z_N$, we have*

$$\mathbb{E}_z[g(z)] \leq \frac{1}{N}\sum_{i=1}^{N} g(z_i) + \sqrt{\frac{\log(|\mathcal{N}_\epsilon(\mathcal{G})|/\delta)}{2N}} + 2\epsilon \tag{2.17}$$

for all $g \in \mathcal{G}$.

Proof. From Theorem 4, we have

$$\mathbb{E}_z[g'(z)] \leq \frac{1}{N}\sum_{i=1}^{N} g'(z_i) + \sqrt{\frac{\log(|\mathcal{N}_\epsilon(\mathcal{G})|/\delta)}{2N}} \tag{2.18}$$

for all $g' \in \mathcal{N}_\epsilon(\mathcal{G})$. Consider some $g \in \mathcal{G}$, there is $g' \in \mathcal{N}_\epsilon(\mathcal{G})$ such that $\|g - g'\| < \epsilon$. By recalling the definition of $\|g - g'\|$, we have

$$\mathbb{E}_z[g(z)] \leq \mathbb{E}_z[g'(z)] + \epsilon \leq \frac{1}{N}\sum_{i=1}^{N} g'(z_i) + \sqrt{\frac{\log(|\mathcal{N}_\epsilon(\mathcal{G})|/\delta)}{2N}} + \epsilon \tag{2.19}$$

$$\leq \frac{1}{N}\sum_{i=1}^{N} g(z_i) + \sqrt{\frac{\log(|\mathcal{N}_\epsilon(\mathcal{G})|/\delta)}{2N}} + 2\epsilon. \tag{2.20}$$

This concludes the proof. □

We will see the covering net generalization error bound in play when we consider an information-theoretic lower bound on quantum advantage in Chapter 7 and the quantum advantage on learning polynomial-time quantum processes in Chapter 9.

Information-theoretic lower bounds

Concentration inequality is useful in proving that a cleverly-designed learning algorithm can successfully learn the underlying mechanism. However, there are often problems that are too hard for any learning algorithm to perform. In order to show that no efficient learning algorithms exist, the concept and techniques in information-theoretic lower bounds become very important. Information-theoretic lower bound studies how much data or experiments are necessary (in terms of a lower bound on the number) to accomplish a certain learning task.

There are all kinds of learning problems. In some problems, we would like to test if certain hypothesis is true. In other problems, we would like to learn how the underlying mechanism works, e.g., by estimating parameters describing the mechanism. There are also problems with the goal of learning an effective model that behaves approximately the same as the true unknown underlying mechanism but could be intrinsically very different from the true mechanism. For most of these problems, we can obtain a good information-theoretic lower bound by reducing the problem to a *distinguishing task*.

The simplest distinguishing task is binary hypothesis testing, where the goal is to distinguish between two hypotheses from some observation $S \in \mathcal{S}$.

1. **Null hypothesis**: Under the null hypothesis, we observe an outcome S (we can think of S as an entire dataset) with probability $q_0(S)$.

2. **Alternative hypothesis**: Under the alternative hypothesis, we observe an outcome S with probability $q_1(S)$.

By looking at the total variation distance between q_0 and q_1,

$$d_{\mathrm{TV}}(q_0, q_1) := \frac{1}{2} \sum_{S \in \mathcal{S}} |q_0(S) - q_1(S)|\,, \tag{2.21}$$

We can tell whether there is an algorithm that has good distinguishing power. When the total variation distance is small, then no good algorithm exists.

Fact 1. *(Lower bound for binary hypothesis testing) Given distributions q_0, q_1 over a domain $\mathcal{S}$, if $d_{TV}(q_0, q_1) < 1/3$, there is no algorithm $\mathcal{A} : \mathcal{S} \to \{0, 1\}$ for which*

$$\Pr_{S \sim q_i} [\mathcal{A}(S) = i] \geq 2/3 \tag{2.22}$$

for both $i = 0, 1$.

Proof. Recall that the total variation distance satisfies the following identity,

$$d_{\mathrm{TV}}(q_0, q_1) = \sup_{\mathcal{S}'' \subseteq \mathcal{S}} |q_0(\mathcal{S}'') - q_1(\mathcal{S}'')|. \tag{2.23}$$

Let $\mathcal{S}' \subseteq \mathcal{S}$ denote the set of elements S for which $\mathcal{A}(S) = 0$. Then observe that

$$\begin{aligned} \Pr_{S \sim q_0} [\mathcal{A}(S) = 1] + \Pr_{x \sim q_1} [\mathcal{A}(x) = 0] &= 1 - q_0(\mathcal{S}') + q_1(\mathcal{S}') && (2.24) \\ &\geq 1 - \sup_{\mathcal{S}'' \subseteq \mathcal{S}} |q_0(\mathcal{S}'') - q_1(\mathcal{S}'')| && (2.25) \\ &= 1 - d_{\mathrm{TV}}(q_0, q_1) \geq 2/3, && (2.26) \end{aligned}$$

so at least one of the terms on the left-hand side is at least $1/3$. □

In many learning tasks, there are many hypotheses that we need to distinguish in order to accomplish the learning task. However, sometimes, distinguishing between every pair of hypotheses is not hard, i.e., the total variation distance is high, but the learning task is still hard. To achieve a better lower bound, we often have to consider a many-versus-one distinguish task.

In a many-versus-one distinguish task, we are given an observation S. And we would like to know if it is more likely to come from a singleton set $Q_0 = \{q_0\}$ (the null hypothesis) or from one of the probability distributions in a large set Q_1 (all the possible alternative hypotheses). While there are many alternative hypotheses, we can reduce this problem to a binary hypothesis testing problem by considering the following average-case version of the distinguishing task.

1. **Null hypothesis**: We observe S with probability $q_0(S)$, where q_0 is the (unique) element of Q_0.
2. **Mixture of alternatives**: We observe S with probability
$$\mathop{\mathbb{E}}_{q\sim\mathcal{D}(Q_1)} q(S) = \sum_{q\in Q_1} p_{\mathcal{D}}(q)q(S), \tag{2.27}$$
where the specific probability distribution $\mathcal{D}(Q_1)$ over all possible alternative hypotheses Q_1 is free for us to choose.

We can use this to show that, in order to prove a lower bound for the original distinguishing task, it suffices to bound $d_{\mathrm{TV}}(q_0, \mathbb{E}_{q\sim\mathcal{D}}[q])$:

Lemma 1 (Le Cam's two-point method). *If there exists a distribution $\mathcal{D}(Q_1)$ over the set Q_1 of alternative hypotheses for which*

$$d_{TV}(q_0, \mathop{\mathbb{E}}_{q\sim\mathcal{D}(Q_1)}[q]) < 1/3, \tag{2.28}$$

there is no algorithm $\mathcal{A}$ which maps observation S to $\{0, 1\}$ for which

$$\Pr_{S\sim q}[\mathcal{A}(S) = i] \geq 2/3 \tag{2.29}$$

for any $q \in Q_i$ and $i = 0, 1$.

Proof. Suppose to the contrary that there existed such an algorithm. Let $q_1 := \mathbb{E}_{q\sim\mathcal{D}(Q_1)}[q]$. Then

$$2/3 \leq \mathop{\mathbb{E}}_{q\sim\mathcal{D}(Q_1)}\left[\Pr_{S\sim q}[\mathcal{A}(S) = i]\right] = \Pr_{S\sim q_1}[\mathcal{A}(S) = i]. \tag{2.30}$$

By Fact 1, this would contradict the fact that $d_{\mathrm{TV}}(q_0, \mathbb{E}_{q\sim\mathcal{D}(Q_1)}[q]) < 1/3$. □

In the distinguishing tasks we consider in this thesis, the choice of $\mathcal{D}$ will be fairly clear (usually a uniform distribution suffices), so the primary technical difficulty for us will be upper bounding the total variation distance between q_0 and $\mathbb{E}_{q\sim\mathcal{D}(Q_1)}[q]$.

Sometimes, the lower bound would be tighter if we consider distinguishing many hypotheses directly. Suppose we have m possible hypotheses $q_1, \ldots, q_m$. Given hypothesis q_i, the observed outcome S is distributed according to the probability distribution $q_i(S)$. Suppose that each hypothesis is chosen uniformly at random. We have the following inequality known as Fano's inequality.

Lemma 2 (Fano's inequality)**.** *Consider a random hypothesis q^* chosen uniformly from $q_1, \ldots, q_m$, and the observation S be sampled according to q^*. For any algorithm $\mathcal{A}$ mapping observation S to $\{q_1, \ldots, q_m\}$, we have*

$$\Pr\left[\mathcal{A}(S) = q^*\right] \log m \leq I(q^* : S) + \log 2, \tag{2.31}$$

where $I(q^* : S)$ is the mutual information between the random hypothesis q^* and the observation S.

The intuition behind Fano's inequality is that if there is an algorithm that can recover the chosen hypothesis q^* from the observation S with high probability, then the mutual information between q^* and S must be of order $\log m$. A common scenario to prove lower bounds using Fano's inequality is to then consider an upper bound of the mutual information $I(q^* : S)$ by studying the structure of the observation S. For example, in Chapter 4, we will upper bound the mutual information by the number of measurements conducted on an unknown quantum state. Together with the lower bound on mutual information given by Fano's inequality, we can obtain a lower bound for the number of measurements needed to learn about an unknown quantum state.

2.3 A brief review on tensor network diagrams

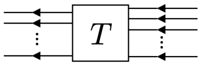
It will be convenient to review the diagrammatic notation for tensor contraction, which we will leverage in several proofs. These so-called 'tensor networks' will render the index contraction of higher-rank tensors more transparent than standard notations. We also refer the interested reader to Landsberg, 2012; Bridgeman and Chubb, 2017 for a more comprehensive overview of tensor networks.

Diagrams for individual tensors

For our purposes, a rank (m, n) tensor is a multilinear map $T : \mathcal{H}^{*\otimes m} \otimes \mathcal{H}^{\otimes n} \to \mathbb{C}$. If $\{|i\rangle\}$ is an orthonormal basis for $\mathcal{H}$, then in bra-ket notation T can be expressed as

$$T = \sum_{\substack{i_1,\ldots,i_m \\ j_1,\ldots,j_n}} T^{i_1\cdots i_m}_{j_1\cdots j_n} \big(|i_1\rangle \otimes \cdots \otimes |i_m\rangle\big)\big(\langle j_1| \otimes \cdots \otimes \langle j_n|\big)\,. \tag{2.32}$$

for some $T^{i_1\cdots i_m}_{j_1\cdots j_n} \in \mathbb{C}$. It is clear that a quantum state $|\Psi\rangle$ on $\mathcal{H}$ is a rank $(1, 0)$ tensor, being a map from $\mathcal{H}^* \to \mathbb{C}$. Accordingly, its dual $\langle\Psi|$ is a $(0, 1)$ tensor. Moreover a matrix $M = \sum_{ij} M^i_j\,|i\rangle\langle j|$ is a $(1, 1)$ tensor. We elect to represent T diagrammatically by



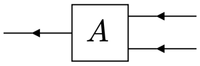
(2.33)

which has m outgoing legs on the left and n incoming legs on the right. Each leg in the diagram may be associated with an index of the coefficients $T^{i_1\cdots i_m}_{j_1\cdots j_n}$. We set the convention that outgoing legs are ordered counter-clockwise and incoming legs are ordered clockwise. For instance, in (2.33) the top-left outgoing leg corresponds to i_1, the leg below to i_2, and so on. Likewise the top-right incoming leg corresponds to j_1, the leg below to j_2, and so on.

Tensor contraction

We next explain how to depict tensor network contractions diagrammatically. For sake of illustration, suppose we have a rank $(2, 1)$ tensor

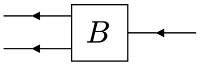
$$A = \sum_{ijk} A^i_{jk}\,|i\rangle\,(\langle j| \otimes \langle k|) \;=$$



(2.34)

and a rank $(1, 2)$ tensor

$$B = \sum_{\ell mn} B^{mn}_{\ell}\,(|m\rangle \otimes |n\rangle)\,\langle\ell| \;=$$

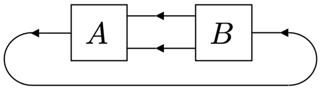


(2.35)

Now suppose we want to compute the tensor network contraction corresponding to

$$\sum_{ijk} A^i_{jk} B^{jk}_i\,. \tag{2.36}$$

Here lower indices are contracted with upper indices because this represents contracting vectors with covectors. The contraction in (2.36) is depicted diagrammatically as

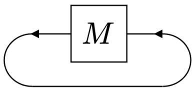


(2.37)

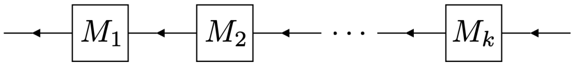
Comparing the diagram with (2.36), we see that contracted indices corresponding to outgoing and incoming lines which are glued together. The fact that vectors are to be contracted with covectors is reflected in the fact that we are only allowed to glue together lines in a manner consistent with their orientations.

As another example, given a matrix $M = \sum_{ij} M^i_j \, |i\rangle\langle j|$, the trace can be written as

$$\mathrm{tr}(M) = \sum_i M^i_i = \tag{2.38}$$



If $M_1, M_2, ..., M_k$ are matrices, then the product $M_1 M_2 \cdots M_k$ is depicted by

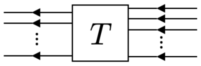


(2.39)

Multiplication by a scalar

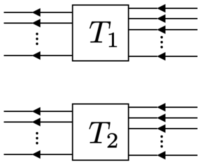
Given a tensor T, multiplication by a scalar α is often denoted by αT. In our diagrammatic notation, we will simply write

$$\alpha \tag{2.40}$$



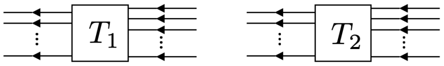
Tensor products

Given two tensors T_1, T_2, we can form the tensor product $T_1 \otimes T_2$. We will denote this diagrammatically as



(2.41)

or also

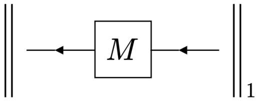


(2.42)

More generally, how to read off the order of a tensor product (e.g. $T_1 \otimes T_2$ or $T_2 \otimes T_1$) from a diagram will be clear in context.

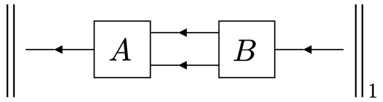
Taking norms

Often it will be convenient to compute the norm of a matrix in tensor notation. For instance, if M is a matrix, then its 1-norm $||M||_1$ can be expressed diagrammatically as



(2.43)

Here we are simply taking the diagrammatic notation for M as a stand-in within the expression $\|M\|_1$. This is particularly convenient in circumstances where M is given by a tensor network contraction whose structure we wish to emphasize; for instance, the 1-norm of $M = \sum_{ijk\ell} A^i_{k\ell} B^{k\ell}_j \, |i\rangle\langle j|$ is conveniently depicted by

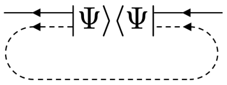


(2.44)

Tensors with legs of different dimensions

So far we have considered rank (m, n) tensors as maps $T : \mathcal{H}^{*\otimes m} \otimes \mathcal{H}^{\otimes n} \longrightarrow \mathbb{C}$. More generally we can consider tensors $T : \left(\mathcal{H}_1^* \otimes \cdots \otimes \mathcal{H}_m^*\right) \otimes \left(\mathcal{H}_{m+1} \otimes \cdots \otimes \mathcal{H}_{m+n}\right) \longrightarrow \mathbb{C}$ where the tensored Hilbert spaces in the domain need not be isomorphic. We can use the same diagrammatic notation as above, with the additional restriction that tensor legs can be contracted if they both carry the same dimension (i.e., correspond to a Hilbert space and a dual Hilbert space of the same dimension).

As an example, we can consider the state $|\Psi\rangle$ in $\mathbb{C}^2 \otimes \mathbb{C}^3$, and form its density matrix $|\Psi\rangle\langle\Psi|$. In our tensor diagram corresponding to this state, the $\mathbb{C}^2$ (qubit) legs will be solid lines and the $\mathbb{C}^3$ (qutrit) legs will be dotted lines. Performing a partial over the qutrit legs is expressed diagrammatically as

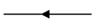


(2.45)

We will discuss the diagrammatic notation of partial traces in more detail below.

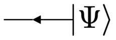
Identity operator

The identity operator on a Hilbert space $\mathcal{H}$ can be expressed diagrammatically as an oriented line



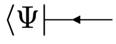
(2.46)

We can clearly see that given a state in the Hilbert space



(2.47)

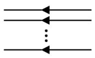
if we left-multiply by the identity diagram we will get the same tensor diagram and thus the same state. Likewise for the dual state



(2.48)

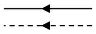
if we right-multiply by the identity diagram then we return the same tensor diagram.

Likewise, the identity operator on k copies of the Hilbert space $\mathcal{H}^{\otimes k}$ is just



(2.49)

In the setting that the Hilbert space under consideration is $\mathcal{H} \otimes \mathcal{H}'$ where each tensor factor has a different dimension, it is convenient to represent tensor legs in $\mathcal{H}$ by solid lines and tensor legs in $\mathcal{H}'$ be dotted line; in this setting the identity operator is

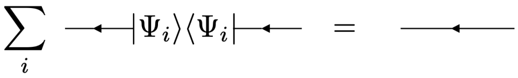


(2.50)

which readily generalizes if there are more than two Hilbert spaces with differing dimensions.

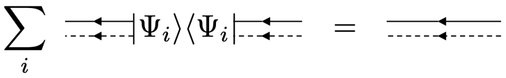
Resolutions of the identity

Suppose $\{|\Psi_i\rangle\}_i$ is an orthonormal basis for $\mathcal{H}$. Then the resolution of the identity $\sum_i |\Psi_i\rangle\langle\Psi_i| = \mathbb{1}$ can be expressed diagrammatically as



(2.51)

If instead $\{|\Psi_i\rangle\}_i$ is a resolution of the identity for $\mathcal{H} \otimes \mathcal{H}'$ where the two Hilbert spaces have different dimensions, we may analogously denote this diagrammatically by

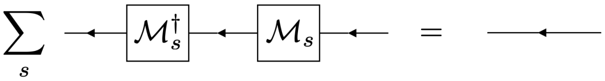


(2.52)

Similarly, if $\{M_s^\dagger M_s\}_s$ is a POVM on $\mathcal{H}$ then the resolution of the identity

$$\sum_s M_s^\dagger M_s = \mathbb{1} \tag{2.53}$$

can be written as

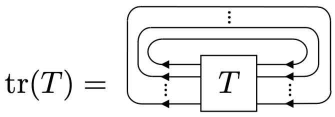


(2.54)

and analogously if the Hilbert space is $\mathcal{H} \otimes \mathcal{H}'$ or has even more tensor factors.

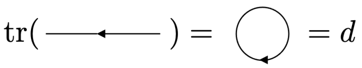
Taking traces and partial traces

Suppose we have a rank (n, n) tensor $T : \mathcal{H}^{*\otimes n} \otimes \mathcal{H}^{\otimes n}$. Then its trace is given by $\mathrm{tr}(T) = \sum_{i_1,\ldots,i_n} T^{i_1\cdots i_n}_{i_1\cdots i_n}$, or diagrammatically



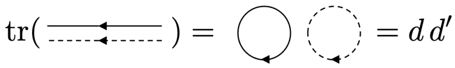
(2.55)

A very useful diagrammatic identity is the trace of the identity matrix, which can be regarded as a rank $(1,1)$ tensor $\mathbb{1} = \sum_i |i\rangle\langle i|$. We have



$$\mathrm{tr}(\quad) = \quad = d \tag{2.56}$$

and so we see that a closed loop in tensor diagrams equals the dimension of the Hilbert space associated that curve. As another example, if we have the identity $\mathbb{1}_{d\times d} \otimes \mathbb{1}_{d'\times d'}$ on $\mathcal{H} \otimes \mathcal{H}'$, where $\dim(\mathcal{H}) = d$, $\dim(\mathcal{H}') = d'$ and we have used subscripts on the identity matrices for emphasis, we have



$$\mathrm{tr}(\quad) = \quad = d\,d' \tag{2.57}$$

where the solid line corresponds to the $\mathcal{H}$ Hilbert space and the dotted line corresponds to the $\mathcal{H}'$ Hilbert space.

We can also take partial traces in similar fashion. We define the partial trace over the 'kth subsystem' by

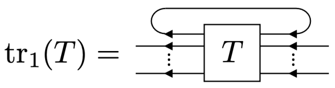
$$\mathrm{tr}_k(T) = \tag{2.58}$$

$$\sum_{\substack{i_1,\ldots,i_{k-1},i_{k+1},\ldots,i_n\\ j_1,\ldots,j_{k-1},j_{k+1},\ldots,j_n}} \left(\sum_{i_k} T^{i_1\cdots i_n}_{j_1\cdots j_n}\right) |i_1\rangle\langle j_1| \otimes \cdots \otimes |i_{k-1}\rangle\langle j_{k-1}| \tag{2.59}$$

$$\otimes\, |i_{k+1}\rangle\langle j_{k+1}| \otimes \cdots \otimes |i_n\rangle\langle j_n|\,. \tag{2.60}$$

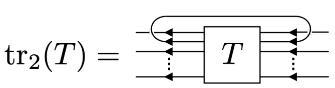
Note that $\mathrm{tr}_\ell(\mathrm{tr}_k(T)) = \mathrm{tr}_k(\mathrm{tr}_\ell(T)))$. Since the operation of taking partial traces is commutative we can use the notation $\mathrm{tr}_{k,\ell}(T)$. Notice that $\mathrm{tr}_{1,\ldots,n}(T) = \mathrm{tr}(T)$. That is, taking the partial trace over all subsystems in the tensor is the same as taking the trace of the entire tensor.

Diagrammatically, the partial trace over the first subsystem is given by



$$\mathrm{tr}_1(T) = \tag{2.61}$$

The partial trace over the second subsystem is

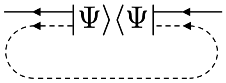


$$\mathrm{tr}_2(T) = \tag{2.62}$$

and so on.

If we have an tensor with legs corresponding to Hilbert spaces of different dimensions, we can still in some cases take traces or partial traces. In particular, if

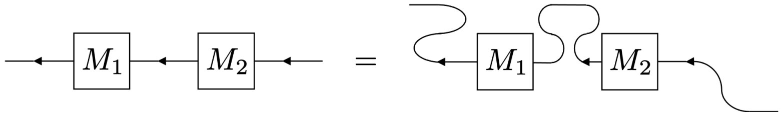
$T : (\mathcal{H}_1^* \otimes \cdots \otimes \mathcal{H}_n^*) \otimes (\mathcal{H}_1' \otimes \cdots \otimes \mathcal{H}_m') \to \mathbb{C}$, then if $\mathcal{H}_k = \mathcal{H}_k'$ we can still compute the partial trace $\mathrm{tr}_k(T)$. As a simple example consider the state $|\Psi\rangle$ living on $\mathcal{H} \otimes \mathcal{H}'$. Then its density matrix $|\Psi\rangle\langle\Psi|$ can be regarded as a $(2, 2)$ tensor taking $(\mathcal{H}^* \otimes \mathcal{H}'^*) \otimes (\mathcal{H} \otimes \mathcal{H}') \to \mathbb{C}$. Then we have

$\mathrm{tr}_2(|\Psi\rangle\langle\Psi|) =$  (2.63)

which is the same example as (2.45); a similar diagram expresses $\mathrm{tr}_1(|\Psi\rangle\langle\Psi|)$.

Isotopies

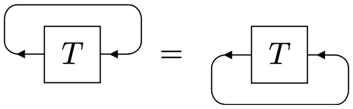
We remark that tensor network diagrams are to be understood up to isotopy of the tensor legs; that is, deforming or bending the tensor legs does not change the interpretation of the diagram. For instance, for a product of matrices $M_1 M_2$ we have equivalences like



(2.64)

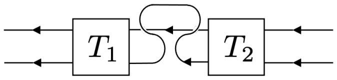
and similarly for all other kinds of tensors.

The isotopies are not required to be planar; for instance



(2.65)

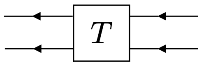
We also can allow legs to cross, for instance



(2.66)

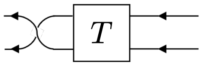
We will disregard whether such crossings are overcrossings or undercrossings.

However, we set the convention that we do not change the relative order of the endpoints of the outgoing or incoming legs. The reason is that permuting the order of the endpoints corresponds would correspond to permuting the tensor factors on which the tensor is defined. As a transparent example, let $T : (\mathcal{H}_1^* \otimes \mathcal{H}_2^*) \otimes (\mathcal{H}_1 \otimes \mathcal{H}_2) \to \mathbb{C}$ be denoted by



(2.67)

Then the diagram



(2.68)

corresponds to a tensor $(\mathcal{H}_2^* \otimes \mathcal{H}_1^*) \otimes (\mathcal{H}_1 \otimes \mathcal{H}_2) \to \mathbb{C}$ where we note that $\mathcal{H}_1^*$ and $\mathcal{H}_2^*$ have been permuted. See also the discussion of permutation operators below.

Permutation operators

Consider the permutation group on k elements, S_k, and let τ be an element of the group. We define a representation of τ, namely $\mathrm{Perm}(\tau)$, which acts on a k-copy Hilbert space $\mathcal{H}^{\otimes k}$ as follows. Letting $|\psi_1\rangle \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_n\rangle$ be a product state on $\mathcal{H}^{\otimes k}$, we define

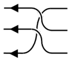
$$\mathrm{Perm}(\tau)|\psi_1\rangle \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_n\rangle = |\psi_{\tau^{-1}(1)}\rangle \otimes |\psi_{\tau^{-1}(2)}\rangle \otimes \cdots \otimes |\psi_{\tau^{-1}(n)}\rangle \tag{2.69}$$

which extends to the entire Hilbert space $\mathcal{H}^{\otimes k}$ by linearity. With these conventions, the representations $\mathrm{Perm}(\tau)$ enjoy the property

$$\mathrm{Perm}(\tau) \cdot \mathrm{Perm}(\sigma) = \mathrm{Perm}(\tau\sigma) \tag{2.70}$$

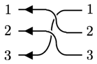
where $\tau\sigma$ is shorthand for the group product, i.e. the composition $\tau \circ \sigma$.

These representations of S_k admit a very intuitive tensor diagrams. Consider, for instance, S_3 and $\tau = (123)$. Then the corresponding tensor diagram is



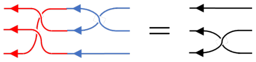
(2.71)

This is made very clear by labeling the endpoints of the diagram by



(2.72)

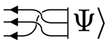
This notation generalized accordingly for other permutation representations. The group product structure is also transparent; for instance $\mathrm{Perm}((123)) \cdot \mathrm{Perm}((12))$ is depicted diagrammatically by



(2.73)

where $\mathrm{Perm}((123))$ is given in red and $\mathrm{Perm}((12))$ is given in blue for clarity; the allowed diagrammatic manipulations of performing isotopies without rearranging the endpoints of the tensor legs show that the result of the product is $\mathrm{Perm}((23))$. A nice feature of the diagrams is that the diagram for $\mathrm{Perm}(\tau^{-1})$ can be obtain from the diagram for $\mathrm{Perm}(\tau)$ by flipping the latter horizontally.

As another example, if we multiply $\mathrm{Perm}((123))$ by a state $|\Psi\rangle$ in $\mathcal{H}^{\otimes 3}$, then we get



(2.74)

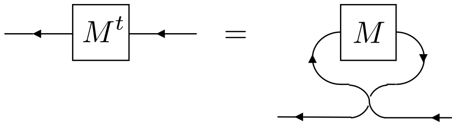
from which it is clear that $\mathrm{Perm}((123))$ permutes the tensor factors of the state according to $(123)^{-1} = (132)$.

In some later proofs where there is no ambiguity, we will denote $\mathrm{Perm}(\tau)$ simply by τ.

Transposes and partial transposes

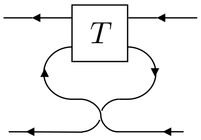
Suppose we have a matrix $M = \sum_{i,j} M^i_j \, |i\rangle\langle j|$ viewed as a rank $(1,1)$ tensor. We can represent its transpose $M^t = \sum_{i,j} M^i_j \, |j\rangle\langle i|$ diagrammatically by

Here we are dualizing each leg by changing the direction of each arrow, and then reorganizing the legs via isotopy so that the in-arrow comes in from the right and the out-arrow comes out to the left; this isotopy is done in order to match the arrow configuration in the diagram on the left.



(2.75)

If we have a higher-rank tensor, such as a rank $(2,2)$ tensor $T = \sum_{ijk\ell} T^{ij}_{k\ell} \, |i\rangle\langle k| \otimes |j\rangle\langle \ell|$, then we can also perform a partial transposition on a subsystem; for instance, the partial transposition on the second subsystem $\sum_{ijk\ell} T^{ij}_{k\ell} \, |i\rangle\langle k| \otimes |\ell\rangle\langle j|$ is given by

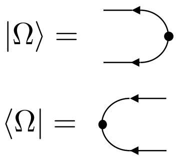


(2.76)

This notation extends to higher rank tensors in an analogous fashion.

Maximally entangled state

The maximally entangled state is given by $|\Omega\rangle = \sum_i |i\rangle|i\rangle$ where $\{|i\rangle\}$ is the computational basis. We treat $|\Omega\rangle$ as unnormalized, and it and its Hermitian conjugate are denoted by



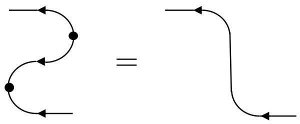
(2.77)

Letting $\mathcal{H}_A \simeq \mathcal{H}_B \simeq \mathcal{H}_C$, we have the identities

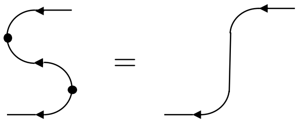
$$(\mathbb{1}_A \otimes \langle\Omega|_{BC})\,(|\Omega\rangle_{AB} \otimes \mathbb{1}_C) = \sum_i |i\rangle_A\langle i|_C \tag{2.78}$$

$$(\langle\Omega|_{AB} \otimes \mathbb{1}_C)\,(\mathbb{1}_A \otimes |\Omega\rangle_{BC}) = \sum_i |i\rangle_C\langle i|_A \tag{2.79}$$

which can be expressed diagrammatically as



(2.80)



(2.81)

We can think of the black dot as being a transpose operation since it changes the orientation of the tensor leg; moreover, two black dots annihilate one another since taking two transposes is the identity operation.

Chapter 3

KEY CONCEPTS FOR LEARNING IN THE QUANTUM UNIVERSE

3.1 Computational power by learning from data

In this chapter, we dive into the computational power classical machines obtain by learning from data. We will look at a motivating example and a rigorous complexity-theoretic argument. The power of data and how to utilize them will be a central concept that appears in many chapters of this thesis. In Chapter 5, we will see how classical machines that learn from data obtained in quantum experiments can solve very challenging quantum many-body problems that no classical machines could solve. In Chapter 6, we will also show that classical machines learned from data can predict the outcomes of a long-time quantum evolution accurately. In Chapter 7, we will see a broad class of problems where classical machines learned from sampled measurement data can predict as accurately as quantum machines with coherent quantum access to the underlying quantum process. In Chapter 8, we will look at how the power of data can elevate and challenge quantum advantage in machine learning problems.

Let us begin with a simple motivating example for studying how data can increase the power of classical machines that learn from the data. Suppose that we have a collection of N training examples $\{(\mathbf{x}_i, y_i)\}$, where $\mathbf{x}_i$ is the input data, and y_i is an associated label or value. We assume that $\mathbf{x}_i$ are sampled independently from a data distribution $\mathcal{D}$ and consider $y_i \in \mathbb{R}$ to be generated by some quantum process $f(x)$ with $y_i = f(\mathbf{x}_i)$.

We now show how the availability of data in machine learning (ML) tasks can change computational hardness. Consider data points $\{\mathbf{x}_i\}_{i=1}^N$ that are p-dimensional classical vectors with $\|\mathbf{x}_i\|_2 = 1$, and use amplitude encoding Grant et al., 2019; Schuld, Bocharov, et al., 2020; LaRose and Coyle, 2020 to encode the data into an n-qubit state $|\mathbf{x}_i\rangle = \sum_{k=1}^p x_i^k |k\rangle$, where x_i^k is the individual coordinate of the vector $\mathbf{x}_i$. If U is a time-evolution under a many-body Hamiltonian, then the function $f(\mathbf{x}) = \langle\mathbf{x}| U^\dagger O U |\mathbf{x}\rangle$ is in general hard to compute classically Aram W Harrow and Montanaro, 2017b, even for a single input state. In particular, we have the following proposition showing that if a classical algorithm can compute $f(\mathbf{x})$ efficiently, then quantum computers will be no more powerful than classical computers. The proof is given later in this section.

Proposition 1. *If a classical algorithm without training data can compute $f(\mathbf{x})$ efficiently for any U and O, then BPP=BQP.*

Nevertheless, it is incorrect to conclude that training a classical machine learning model from data to learn this evolution is hard. To see this, we write out the expectation value as

$$\begin{aligned} f(x_i) &= \left(\sum_{k=1}^{p} x_i^{k*} \langle k|\right) U^\dagger O U \left(\sum_{l=1}^{p} x_i^l |l\rangle\right) \\ &= \sum_{k=1}^{p}\sum_{l=1}^{p} B_{kl} x_i^{k*} x_i^l, \end{aligned} \tag{3.1}$$

which is a quadratic function with p^2 coefficients $B_{kl} = \langle k| U^\dagger O U |l\rangle$. Using the theory developed later in this thesis, we can show that, for any U and O, training a specific classical ML model on a collection of N training examples $\{(\mathbf{x}_i, y_i = f(\mathbf{x}_i))\}$ would give rise to a prediction model $h(\mathbf{x}_i)$ with

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}} |h(\mathbf{x}) - f(\mathbf{x})| \leq c\sqrt{\frac{p^2}{N}}, \tag{3.2}$$

for a constant $c > 0$. The proof of this statement is given later in this section. Hence, with $N \propto p^2/\epsilon^2$ training data, one can train a classical ML model to predict the function $f(\mathbf{x})$ up to an additive prediction error ϵ. This elevation of classical machines through some training samples is illustrative of the power of data. In the later part of this section, we give a rigorous complexity-theoretic argument on the computational power provided by data.

Rigorous proofs for statements regarding the motivating example

We first give a simple proof that the motivating example $f(\mathbf{x})$ considered earlier is, in general, hard to compute classically. Then, we show that training a classical ML model to predict the function $f(\mathbf{x})$ is easy on a classical computer.

Proposition 2 (Restatement of Proposition 1)**.** *Consider input vector $\mathbf{x} \in \mathbb{R}^p$ encoded into an n-qubit state $|\mathbf{x}\rangle = \sum_{k=1}^{p} x_k |k\rangle$. If a randomized classical algorithm can compute*

$$f(\mathbf{x}) = \langle \mathbf{x}| U^\dagger O U |\mathbf{x}\rangle \tag{3.3}$$

up to 0.15*-error with high probability over the randomness in the classical algorithm for any n, U, and O in a time polynomial to the description length of U and O, the input vector size p, and the qubit system size n, then*

$$BPP = BQP. \tag{3.4}$$

Proof. We consider $p = 1$ and $|\mathbf{x}\rangle = |0^n\rangle$ the all zero computational basis state. A language L is in BQP if and only if there exists a polynomial-time uniform family of quantum circuits $\{Q_n : n \in \mathbb{N}\}$, such that

1. For all $n \in \mathbb{N}$, Q_n takes an n-qubit computational basis state as input, apply Q_n on the input state, and measures the first qubit in the computational basis as output.

2. For all $z \in L$, the probability that output of $Q_{|z|}$ applying on the input z is one is greater than or equal to $2/3$.

3. For all $z \notin L$, the probability that output of $Q_{|z|}$ applying on the input z is zero is greater than or equal to $2/3$.

If we have the randomized classical algorithm that can compute $f(x)$, then for all z: input bitstring, we consider the unitary quantum neural network given by

$$U = Q_{|z|} \bigotimes_{i=1}^{n} X_i^{z_i}, \tag{3.5}$$

where X_i is the Pauli-X matrix acting on the i-th qubit, and the observable O is given by Z_1. Hence, we have

1. For all $z \in L$, $f(\mathbf{x}) = \langle \mathbf{x}| U^\dagger O U |\mathbf{x}\rangle = \langle z| Q_{|z|}^\dagger Z_1 Q_{|z|} |z\rangle = \Pr[$the output of $Q_{|z|}$ applying on the input z is one $] - \Pr[$the probability that output of $Q_{|z|}$ applying on the input z is zero$] \geq 2/3 - 1/3 = 1/3$.

2. For all $z \notin L$, $f(\mathbf{x}) = \langle \mathbf{x}| U^\dagger O U |\mathbf{x}\rangle = \langle z| Q_{|z|}^\dagger Z_1 Q_{|z|} |z\rangle = \Pr[$the output of $Q_{|z|}$ applying on the input z is one $] - \Pr[$the probability that output of $Q_{|z|}$ applying on the input z is zero$] \leq 1/3 - 2/3 = -1/3$.

By assumption, we can use the randomized classical algorithm to compute an estimate $\hat{f}(\mathbf{x})$ such that $|\hat{f}(\mathbf{x}) - f(\mathbf{x})| < 0.15$ with high probability over the randomness of the classical algorithm. Therefore with high probability, $\hat{f}(\mathbf{x}) > 0$ if $z \in L$ and

$\hat{f}(\mathbf{x}) < 0$ if $z \notin L$. We can use the indication of whether $\hat{f}(\mathbf{x})$ is positive or negative to determine if $z \in L$ or $z \notin L$ with high probability over the randomness of the classical algorithm. This implies that $L \in$ BPP.

Together, the existence of the randomized classical algorithm implies that BQP $\subseteq$ BPP. By definition, we have BPP $\subseteq$ BQP, hence BPP = BQP. □

We will now give a classical machine learning algorithm that could learn $f(\mathbf{x})$ efficiently using few samples. Recall that the data point is given by $\{\mathbf{x}_i\}_{i=1}^{N}$, where $\mathbf{x}_i \in \mathbb{R}^p$. Now, we consider a classical ML model with the kernel function $k(\mathbf{x}_i, \mathbf{x}_j) = (\sum_{l=1}^{p} x_{il} x_{jl})^2$, which can be evaluated in time linear in the dimension p. Note that this definition of kernel is equivalent to the quantum kernel $\mathrm{tr}(\rho(x_i)\rho(x_j)) = |\langle x_i \rangle x_j|^2$ for the encoding $|x_i\rangle = \sum_{k=1}^{p} x_{ik} |k\rangle$. We will now use the theoretical framework we developed in Chapter 8. In particular, we will use the prediction error of the quantum kernel method given in Eq. 8.9. It shows that for any observable O and quantum circuit U, the prediction error after training from N data points $\{(x_i, y_i = f(x_i))\}$ is given by

$$\mathbb{E}_{\mathbf{x}\in\mathcal{D}}|h(\mathbf{x}) - f(\mathbf{x})| \leq c\sqrt{\frac{\min(d, \mathrm{tr}(O^2))}{N}}, \tag{3.6}$$

where d is the Hilbert space dimension of $\{\rho(x_i)\}_{i=1}^{N}$. Because we have $\rho(x_i) = |x_i\rangle\langle x_i|$ and $|x_i\rangle = \sum_{k=1}^{p} x_{ik} |k\rangle$, the dimension of the Hilbert space is upper bounded by p^2. Therefore,

$$\mathbb{E}_{\mathbf{x}\in\mathcal{D}}|h(\mathbf{x}) - f(\mathbf{x})| \leq c\sqrt{\frac{\min(d, \mathrm{tr}(O^2))}{N}} \leq c\sqrt{\frac{p^2}{N}}. \tag{3.7}$$

For more details about the machine learning models, the prediction error bound, and the proof for the prediction error bound of quantum kernel methods, see Section 8.6 and 8.7 in Chapter 8.

Complexity-theoretic argument for the power of data

So far, we have seen an argument based on a simple example to demonstrate the power of data. However, this is not satisfactory when we want to put the power of data on a rigorous footing. To demonstrate this fact from a rigorous standpoint, let us capture classical ML algorithms that can learn from data by means of a complexity class, which we refer to as BPP/samp. A language L of bit strings is in BPP/samp if and only if the following holds: There exist probabilistic Turing machines D and M. D generates samples x with $|x| = n$ in polynomial time for any input size n. D

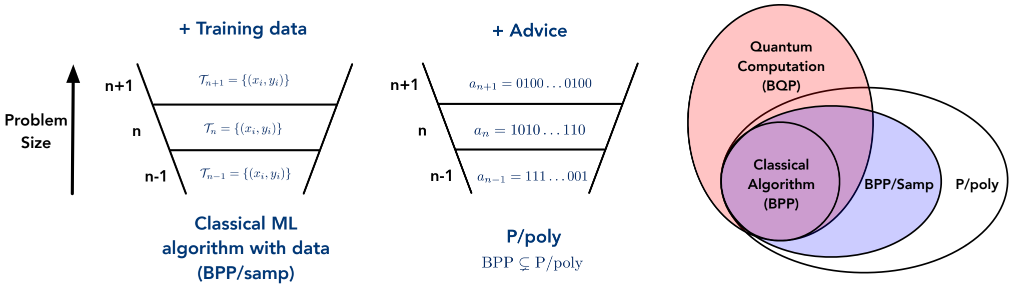

+ Training data
Problem Size
n+1
n
n-1
$\mathcal{T}_{n+1} = \{(x_i, y_i)\}$
$\mathcal{T}_n = \{(x_i, y_i)\}$
$\mathcal{T}_{n-1} = \{(x_i, y_i)\}$
Classical ML algorithm with data (BPP/samp)
+ Advice
n+1
n
n-1
$a_{n+1} = 0100\ldots0100$
$a_n = 1010\ldots110$
$a_{n-1} = 111\ldots001$
P/poly
$\mathrm{BPP} \subsetneq \mathrm{P/poly}$
Quantum Computation (BQP)
Classical Algorithm (BPP)
BPP/Samp
P/poly

Figure 3.1: We present an illustration of the complexity class for classical machine learning algorithms with the availability of data. To the right, we have a diagram showing the relations between different complexity classes.

defines a sequence of input distributions $\{\mathcal{D}_n\}$. M takes an input x of size n along with $\mathcal{T} = \{(x_i, y_i)\}_{i=1}^{\mathrm{poly}(n)}$ of polynomial size, where x_i is sampled from $\mathcal{D}_n$ using Turing machine D and y_i conveys language membership: $y_i = 1$ if $x_i \in L$ and $y_i = 0$ if $x_i \notin L$. Moreover, we require

- The probabilistic Turing machine M to process all inputs x in polynomial time (polynomial runtime).
- For all $x \in L$, M outputs 1 with probability greater than or equal to $2/3$ (probability completeness).
- For all $x \notin L$, M outputs 1 with probability less than or equal to $1/3$ (probability soundness).

If the Turing machine M neglects the sampled data $\mathcal{T}$, this is equivalent to the definition of BPP. Hence BPP is contained inside BPP/samp.

We can also see that $\mathcal{T}$ is a restricted form of randomized advice string. It is not hard to show that BPP/samp is contained in P/poly based on the same proof strategy for Adleman's theorem. We consider a new probabilistic Turing machine M' that runs M for $18n$ times. Each time, we use an independently sampled training set $\mathcal{T}$ from $\mathcal{D}_n$. Then we take a majority vote from the $18n$ runs. By Chernoff bound, the probability of failure for any given x with $|x| = n$ would be at most $1/\mathrm{e}^n$. Hence by union bound, the probability that all x with $|x| = n$ succeeds is at least $1 - (2/\mathrm{e})^n$. This implies the existence of a particular choice of the $18n$ training sets and $18n$ random bit-strings used in each run of the probabilistic Turing machine M, such that for all x with $|x| = n$ the decision of whether $x \in L$ is correct. We simply define the advice string a_n to one particular choice of the $18n$ training sets and $18n$ random bit-strings, which will be a string of size polynomial in n. Hence we know that BPP/samp is contained in P/poly. An illustration is given in Supplementary Figure 3.1. We leave open the question of whether BPP/samp is strictly contained in P/poly, i.e., there is a problem in P/poly but not in BPP/samp.

The separation between P/poly and BPP is often illustrated by undecidable unary languages. The separation between BPP/samp and BPP could also be proved using a similar example. Actually, an undecidable unary language serves as an equally good example. Here, we choose to present a slightly more complicated example to demonstrate what BPP/samp could do. Let us consider an undecidable unary language $L_{\text{hard}} = \{1^n | n \in A\}$, where A is a subset of the natural numbers $\mathbb{N}$ and a classically easy language $L_{\text{easy}} \in \text{BPP}$. We assume that for every input size n, there exists an input $a_n \in L_{\text{easy}}$ and an input $b_n \notin L_{\text{easy}}$. We define a new language as follows:

$$L = \bigcup_{n=1}^{\infty} \{x | \forall x \in L_{\text{easy}}, 1^n \in L_{\text{hard}}, |x| = n\} \cup \{x | \forall x \notin L_{\text{easy}}, 1^n \notin L_{\text{hard}}, |x| = n\}. \tag{3.8}$$

For each size n, if $1^n \in L_{\text{hard}}$, L would include all $x \in L_{\text{easy}}$ with $|x| = n$. If $1^n \notin L_{\text{hard}}$, L would include all $x \notin L_{\text{easy}}$ with $|x| = n$. By definition, if we can output whether $x \in L$ for an input x using a classical algorithm (BPP), we can output whether $1^n \in L_{\text{hard}}$ by computing whether $x \in L_{\text{easy}}$. This is however impossible due to the undecidability of L_{hard}. Hence the language L is not in BPP. On the other hand, for every size n, a classical machine learning algorithm can use a single training data point (x_0, y_0) to decide whether $x \in L$. An algorithm is as follows. Using y_0, we know whether $x_0 \in L_{\text{easy}}$. Hence, we know whether $1^n \in L_{\text{hard}}$. Then for any input x with size n, we can output the correct answer by using the knowledge of whether $1^n \in L_{\text{hard}}$ combined with a classical computation to decide whether $x \in L_{\text{easy}}$. This example nicely illustrates the power of data and how machine learning algorithms can utilize it. In summary, the data provide information that is hard to compute with a classical computer (e.g., whether $1^n \in L_{\text{hard}}$). Then the classical machine learning algorithm would perform the classical computation to infer the solution from the given knowledge (e.g., computing whether $x \in L_{\text{easy}}$). The same language L also yields a separation between BPP/samp and BQP because L is constructed to be undecidable.

From a practical perspective, it is impossible to obtain training data that is undecidable. But it is still possible to obtain data that cannot be efficiently computed with a classical computer since the universe operates quantum mechanically. If the universe computes classically, then the data we can obtain will be computable by BPP and there is no separation between classical ML algorithm with data from BPP and BPP. We now present a simple argument for a separation between classical algorithm learning with data coming from quantum computation and BPP. This follows from a similar argument as the previous example. Here, we assume that there is a sequence of quantum circuits such that the Z measurement on the first qubit (being +1 with probability $> 2/3$ or $< 1/3$) is hard to decide classically. This defines a unary language L'_{hard} that is outside BPP, but inside BQP. We can then use L'_{hard} in replace of L_{hard} for the example above. When the data comes from BQP, the class classical ML algorithms that can learn from the data would not have a separation from BQP. Together, the motivating example and the complexity-theoretic analysis give a solid foundation for the computational power of data.

3.2 Proving quantum advantages in learning

One of the central ingredients for establishing exponential quantum advantage in learning is to prove an exponential lower bound for any classical learning algorithm. In this chapter, we present a mathematical framework and the key techniques for proving such lower bounds. The framework is designed to show a lower bound for any algorithm that only has an external classical memory, i.e., the algorithm cannot carry quantum information from a previous experiment to the next experiment. This class of learning algorithms covers all possible classical learning algorithms (since classical machines cannot carry quantum information) but also covers some restricted quantum learning algorithms. This framework enables us to establish a suite of exponential advantages in various physically-relevant tasks, as we will show in Chapter 9. The purpose of this chapter is to provide the readers with the essential tools to prove quantum advantages in the tasks they want to study.

The basic tools include the tree representation of learning algorithms, reduction to distinguishing tasks, and information-theoretic lower bounds. Many of these techniques were introduced and leveraged in (Sitan Chen, J. Cotler, et al., 2021b). We also present a novel *partially-revealed many-versus-one distinguishing task* that is crucial for realizing the advantage in practice. Then, we discuss how having noise in the unknown physical states and dynamics only makes the lower bounds for conventional experiments larger. This result on the presence of noise is simple to establish but also crucial in practice because there is often noise in the unknown physical states and dynamics. There are some other techniques presented in a theory paper written by some of the authors (Sitan Chen, J. Cotler, et al., 2021b), such as a multi-linear tensor analysis on the learning tree, which may be of interest to some readers.

Tree representation

We begin by presenting the tree representation for analyzing algorithms with only classical memory (Sitan Chen, J. Cotler, et al., 2021b). The key idea is to track changes in the classical memory state in the algorithm using a graph, which we can take to be a rooted tree. We consider each node u of the graph to be a classical memory state. Based on the memory state, the algorithm performs an experiment to obtain a measurement outcome s.

Experiments for learning physical world

To motivate the definitions in the sequel, we separately describe the two types of experimental setups that we focus on in this chapter: one on learning an unknown physical state and the other on learning an unknown process (Huang, Richard Kueng, and Preskill, 2021; Aharonov, J. S. Cotler, and Qi, 2021; Sitan Chen, J. Cotler, et al., 2021b).

- *Learning an unknown physical state*: A physical state is represented by a density matrix ρ. An algorithm leveraging the classical memory state u measures the physical system ρ using a rank-1 POVM $\{w_s^u |\phi_s^u\rangle\!\langle\phi_s^u|\}$ with $\sum_s w_s^u |\phi_s^u\rangle\!\langle\phi_s^u| = \mathrm{Id}$. Note that from the discussion in Section 2.1, we can always consider rank-1 POVMs only. The measurement outcome s occurs with probability
$$w_s^u \langle\phi_s^u| \rho |\phi_s^u\rangle . \tag{3.9}$$
Here, the rank-1 POVM $\{w_s^u |\phi_s^u\rangle\!\langle\phi_s^u|\}$ depends on the classical memory state u.

- *Learning an unknown physical process*: A physical process is represented by a quantum process $\mathcal{E}$ (equivalently, a CPTP map). An algorithm leveraging the memory state u prepares an initial state $|\psi^u\rangle$, feeds it into the physical evolution $\mathcal{E}$, and measures the output state $\mathcal{E}(|\psi^u\rangle\!\langle\psi^u|)$ with a rank-1 POVM $\{w_s^u |\phi_s^u\rangle\!\langle\phi_s^u|\}$ with $\sum_s w_s^u |\phi_s^u\rangle\!\langle\phi_s^u| = \mathrm{Id}$. The outcome s is obtained from the experiment with probability
$$w_s^u \langle\phi_s^u| \mathcal{E}(|\psi^u\rangle\!\langle\psi^u|) |\phi_s^u\rangle . \tag{3.10}$$

In this case, both the initial state and the measurement depend on the classical memory state u.

We consider the initial state $|\psi^u\rangle$ to be an $(n+n')$-qubit state, where $\mathcal{E}$ acts on the first n qubits. The rank-1 POVM $\{w_s^u |\phi_s^u\rangle\langle\phi_s^u|\}$ is on an $(n+n')$-qubit state.

Dynamics of the learning algorithm

The classical memory state of the learning algorithm is initialized in a certain state, which we represent by the root node r. The memory state of the algorithm begins at the root r. Each measurement outcome s resulting from a single experiment causes the algorithm to transition to a node neighbor of r. Whenever the algorithm obtains a measurement outcome s, the memory state changes. This is represented by a transition from a node u to another node v,

$$u \xrightarrow{s} v. \tag{3.11}$$

The directed edge $e = (u, v, s)$ from u to v represents the transition of the memory in an algorithm when we receive the measurement outcome s. We illustrate the transition under a single experiment in Supp. Fig. 3.2(a).

If a different s leads us to the same node, then the algorithm is not retaining full information of the measurement outcome. An example is given in Supp. Fig. 3.2(b). Since we do not limit the size of the classical memory, there is no need to lose (or forget) information. Hence, all the outgoing edges of the root node r indexed by the measurement outcome s will point to distinct nodes. The same argument holds for any node in the graph. More precisely, every outgoing edge from a node u will connect to a node v, such that v has exactly one incoming edge (the edge is from u). The only node in the graph without an incoming edge is the root r. This is exactly the definition of a directed rooted tree $\mathcal{T}$. We will focus on an algorithm that performs T experiments. This means the depth of the tree, namely the number of edges in any root-to-leaf path, will be T. The tree representation is shown in Supp. Fig. 3.2(c).

When we execute the algorithm to achieve a certain task (such as to verify entanglement, or learn a model of the physical system), the entire dynamic process of how the memory state changes will be represented by a path from the root r to a leaf ℓ in the tree $\mathcal{T}$,

$$u_0 = r \xrightarrow{s_1} u_1 \xrightarrow{s_2} u_2 \xrightarrow{s_3} \ldots \xrightarrow{s_{T-1}} u_{T-1} \xrightarrow{s_T} u_T = \ell. \tag{3.12}$$

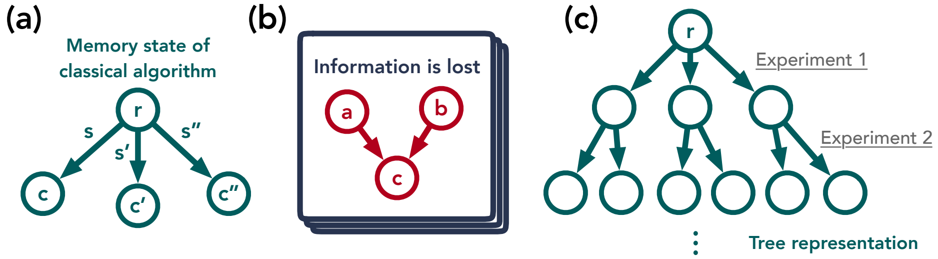


Figure 3.2: *Illustration of the tree representation for a learning algorithm. (a) Dynamics of memory.* The memory state changes based on the measurement outcome s. *(b) No cycles.* If two memory states a, b transition into the same memory state c, then some information is lost. *(c) Tree representation of an algorithm.* When no information is lost, the transition graph of the memory states must be a directed tree. Each layer of the tree corresponds to one experiment. After T experiments, the memory state is represented by a node in the T-th layer.

To establish a lower bound against any learning algorithm for a particular task, we need to analyze each such path along with the probability that the path is taken.

Many-versus-one distinguishing tasks

Reduction

In a learning task, we often want the learning algorithm to be able to make accurate predictions about some properties of the unknown physical system or dynamics. We will have a set of states (the mathematical representation of a physical system) or a set of channels (the mathematical representation of physical dynamics) to which we assume the unknown system or dynamics belong. The basic technique we employ in all of our proofs is to pick out one of the states/channels as the null hypothesis, and consider all the rest as the alternative hypothesis (Huang, Richard Kueng, and Preskill, 2021; Sitan Chen, J. Cotler, et al., 2021b). Let $\mathcal{X}$ denote the set of possible states/channels.

- *Null hypothesis*: The unknown state/channel is an element $X_0 \in \mathcal{X}$. To establish a tight lower bound, we should choose an X_0 that we think is *close* to every other state/channel in $\mathcal{X}$.
- *Alternative hypothesis*: The unknown state/channel is a random element in $\mathcal{X} \setminus \{X_0\}$.

Furthermore, we need to choose X_0 such that the desired property we would like to

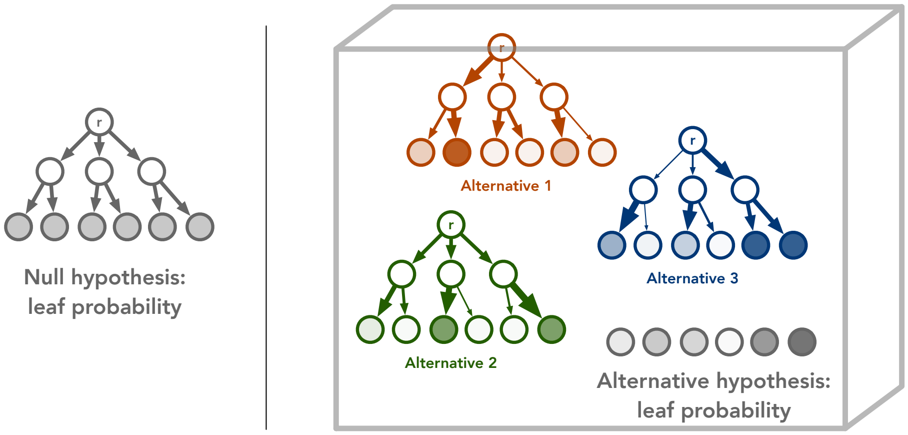


Figure 3.3: *Illustration for the leaf probability distribution.* The leaf probability distribution depends on the unknown physical state/process and the learning algorithm. In the null hypothesis, we have a single state/process, which gives rise to a probability distribution over leaves. In the alternative hypothesis, there are multiple possible states/processes. Each state/process produces a different leaf probability distribution. The leaf probability distribution for the alternative hypothesis is the average of all the leaf probability distributions.

learn enables us to distinguish between X_0 and the entire set of $\mathcal{X} \setminus \{X_0\}$.

To prove a lower bound against any classical algorithm, we try to answer the following question.

How hard is it to distinguish the alternative hypothesis from the null hypothesis?

Because the alternative hypothesis consists of many elements and the null hypothesis consists of only one element, we refer to this distinguishing task as the *many-versus-one distinguishing task.*

Information-theoretic lower bound

In order to establish a lower bound for the many-versus-one distinguishing task, we need to first discuss the leaf probability distribution in the tree representation of the learning algorithm (Huang, Richard Kueng, and Preskill, 2021; Sitan Chen, J. Cotler, et al., 2021b). Recall that depending on the unknown state/process, the transition probabilities among the memory states in the learning algorithm will be different. This is because the outcome probability for each experiment differs when the unknown state/process differs. Therefore, the probability to traverse a certain path in the tree representing an execution of the learning algorithm will change according to the unknown state/process. Hence, the probability to arrive at a particular leaf node in the depth-T tree will change. An illustration is given in Supp. Fig. 3.3.

For each element X in $\mathcal{X}$, the set of all admissible states/channels, we write the probability distribution over leaves as

$$p_X(\ell), \quad \ell : \text{leaf node of the tree.} \tag{3.13}$$

The probability distribution over the leaves ℓ for the null hypothesis and for the alternative hypothesis are respectively

$$p_{X_0}(\ell) \;\text{ and }\; \mathop{\mathbb{E}}_{X\in\mathcal{X}\setminus\{X_0\}} p_X(\ell). \tag{3.14}$$

The probability distribution over $X \in \mathcal{X} \setminus \{X_0\}$ in the expectation $\mathbb{E}_{X\in\mathcal{X}\setminus\{X_0\}}$ is arbitrary. We should choose the probability distribution that yields the largest lower bound.

Suppose that the null hypothesis and the alternative hypothesis are true with probability $1/2$ each. If we want to use the memory state of the learning algorithm to distinguish between the null hypothesis and the alternative hypothesis, then the success probability of any procedure is upper bounded by

$$\frac{1}{2}+\frac{1}{2}\mathrm{TV}\left(p_{X_0}, \mathop{\mathbb{E}}_{X\in\mathcal{X}\setminus X_0} p_X\right) = \frac{1}{2}+\frac{1}{4}\sum_{\ell}\left|p_{X_0}(\ell) - \mathop{\mathbb{E}}_{X\in\mathcal{X}\setminus X_0} p_X(\ell)\right|, \tag{3.15}$$

which is also known as LeCam's two-point method. $\mathrm{TV}(p_0, p_1)$ is the total variation distance between the two probability distributions p_0, p_1.

Lemma 3 (Le Cam's two-point method, see e.g. Lemma 1 in (B. Yu, 1997))**.** *Consider a learning algorithm without quantum memory that is described by a rooted tree $\mathcal{T}$. The probability that the learning algorithm solves the many-versus-one distinguishing task correctly is upper bounded by*

$$\frac{1}{2}+\frac{1}{2}\sum_{\ell\in\mathrm{leaf}(\mathcal{T})}\left|\mathop{\mathbb{E}}_{x} p^{\rho_x}(\ell) - p^{\mathrm{Id}/2^n}(\ell)\right|. \tag{3.16}$$

Intuitively, as we perform more experiments, the depth of the tree increases, and the total variation distance between the leaf probability distribution increases. If we want to achieve a prediction accuracy of $p \geq \frac{1}{2}$, then we need the total variation distance to be lower bounded by,

$$\mathrm{TV}\left(p_{X_0}, \mathop{\mathbb{E}}_{X\in\mathcal{X}\setminus X_0} p_X\right) \geq 2p - 1. \tag{3.17}$$

On the other hand, the total variation distance can be upper bounded by a monotonically increasing function of the number of experiments T equal to the depth of the tree. Altogether, this allows us to lower bound the number of experiments T by a function of the success probability p.

It is tempting to try to apply Lemma 3 by uniformly upper bounding the quantity $\left|\mathbb{E}_x\, p^{\rho_x}(\ell) - p^{\mathrm{Id}/2^n}(\ell)\right|$ for all leaves ℓ. Unfortunately, it turns out that for some leaves, this quantity can be very large. The good news however is that we do have a uniform *one-sided* bound: as we will show, $p^{\mathrm{Id}/2^n}(\ell) - \mathbb{E}_x\, p^{\rho_x}(\ell)$ (without the absolute values) can always be upper bounded by a very small value. It turns out that such a one-sided bound already suffices for applying Le Cam's two-point method.

Lemma 4 (One-sided bound suffices for Le Cam)**.** *Consider a learning algorithm without quantum memory that is described by a rooted tree $\mathcal{T}$. If we have*

$$\frac{\mathbb{E}[x] * p^{\rho_x}(\ell)}{p^{\mathrm{Id}/2^n}(\ell)} \geq 1 - \delta, \quad \forall \ell \in \mathrm{leaf}(\mathcal{T}). \tag{3.18}$$

then the probability that the learning algorithm solves the many-versus-one distinguishing task correctly is upper bounded by δ.

Proof. We utilize the basic fact that $\frac{1}{2}\sum_i |p(i) - q(i)| = \sum_{i:p(i)\geq q(i)} p(i) - q(i)$, hence

$$\frac{1}{2} \sum_{\ell\in\mathrm{leaf}(\mathcal{T})} \left|\mathop{\mathbb{E}}_x p^{\rho_x}(\ell) - p^{\mathrm{Id}/2^n}(\ell)\right| = \sum_{\substack{\ell\in\mathrm{leaf}(\mathcal{T}) \\ p^{\mathrm{Id}/2^n}(\ell)\geq\mathbb{E}_x p^{\rho_x}(\ell)}} p^{\mathrm{Id}/2^n}(\ell) - \mathop{\mathbb{E}}_x p^{\rho_x}(\ell) \tag{3.19}$$

$$\leq \sum_{\substack{\ell\in\mathrm{leaf}(\mathcal{T}) \\ p^{\mathrm{Id}/2^n}(\ell)\geq\mathbb{E}_x p^{\rho_x}(\ell)}} p^{\mathrm{Id}/2^n}(\ell)\delta \quad \leq \quad \delta. \tag{3.20}$$

We can then apply Lemma 3 and conclude the proof of the lemma. □

Many-versus-many distinguishing task

Sometimes, it is easier to first reduce the learning task to a many-versus-many distinguishing task before reducing it to a many-versus-one task. This technique is used in Section 9.6 to prove an exponential advantage for quantum principal component analysis. Consider $\mathcal{X}$ to be the set of allowed states/channels. We consider a subset $\mathcal{A} \subseteq \mathcal{X}$, and define $\mathcal{B} = \mathcal{X} \setminus \mathcal{A}$. Here, we consider the following two hypotheses.

- Hypothesis A: The unknown state/channel is a random element in $\mathcal{A}$.

- Hypothesis B: The unknown state/channel is a random element in $\mathcal{B}$.

Assume each hypothesis happens with probability $1/2$. The goal is to distinguish which hypothesis is true. Because $\mathcal{A}, \mathcal{B}$ can contain many elements in $\mathcal{X}$, we refer to this as the many-versus-many distinguishing task. In Supp. Fig. 3.4, we visualize the difference between the *many-versus-many distinguishing task* and the other tasks.

Following a similar derivation as the many-versus-one distinguishing task, for any learning algorithm in the conventional setting, we represent the algorithm as a learning tree. Given a tree representation $\mathcal{T}$, the success probability for any procedure to distinguish hypothesis A and B using the final memory state of the learning algorithm is upper bounded by

$$\frac{1}{2} + \frac{1}{2}\mathrm{TV}\left(\mathop{\mathbb{E}}_{X \in \mathcal{A}} p_X, \mathop{\mathbb{E}}_{X \in \mathcal{B}} p_X\right), \tag{3.21}$$

where p_X is a probability distribution over the leaf nodes of the tree $\mathcal{T}$ when the unknown state/channel is X. Hence, if we want to achieve a prediction accuracy of $p \geq \frac{1}{2}$, then we need the total variation distance to be lower bounded by

$$\mathrm{TV}\left(\mathop{\mathbb{E}}_{X \in \mathcal{A}} p_X, \mathop{\mathbb{E}}_{X \in \mathcal{B}} p_X\right) \geq 2p - 1\,. \tag{3.22}$$

This last inequality will be important for establishing the lower bound on the number of experiments T.

Partially-revealed many-versus-one distinguishing task

In some tasks, it can be challenging to verify if an algorithm has learned accurately without revealing some information to the learning algorithm. Here, we consider a setting where after learning, the algorithm is additionally given some partial information about the underlying state/process.

The set $\mathcal{X}$ of all admissible states/processes can be represented as follows,

$$X = (\xi, \chi) \in \mathcal{X}, \tag{3.23}$$

where ξ is the information that will be revealed during prediction and χ remains hidden. After performing all experiments, the algorithm can obtain ξ^*, such that the unknown state/process X is guaranteed to be either

$$X_0 \quad \text{or} \quad (\xi^*, \chi) \in \mathcal{X} \setminus \{X_0\}, \tag{3.24}$$

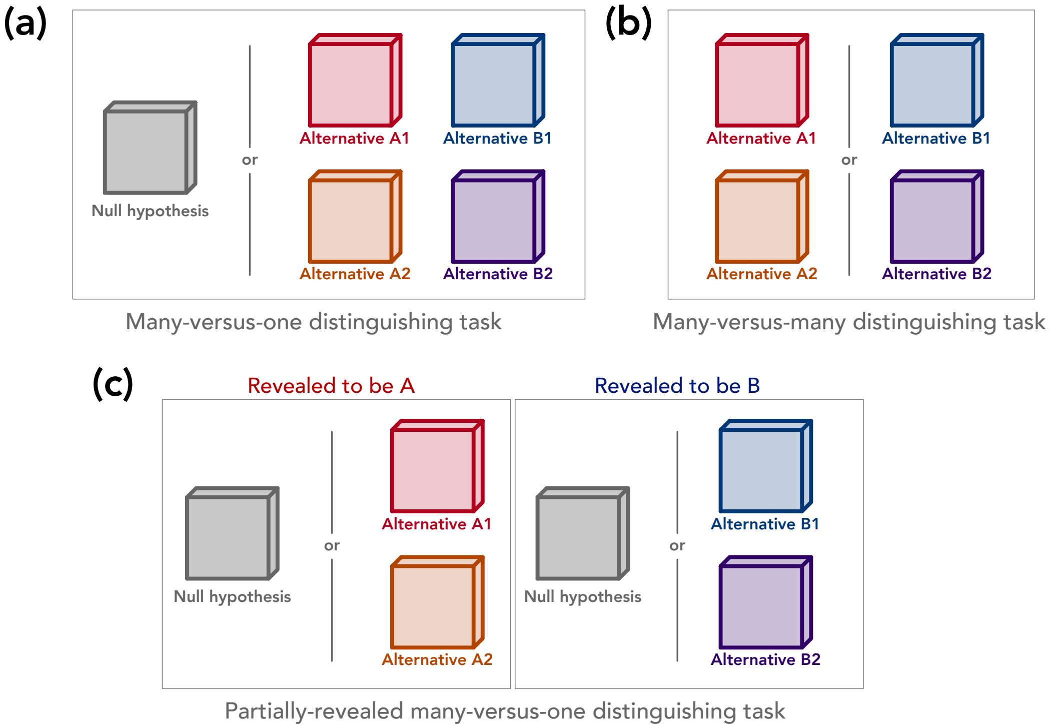


Figure 3.4: *Visualization of the different distinguishing tasks.* *(a)* In the many-versus-one distinguishing task, we are distinguishing between the null hypothesis and the alternative hypothesis (which can be one of many alternatives). *(b)* In the many-versus-many distinguishing task, we want to distinguish between two hypotheses (each of which can be one of many alternatives). *(c)* In the partially-revealed many-versus-one distinguishing task, some information about the alternatives is revealed, which makes the distinguishing task easier.

i.e. the null hypothesis or an element of the alternative hypothesis. In Supp. Fig. 3.4, we visualize the difference between the *partially-revealed many-versus-one distinguishing task* and other tasks. Due to the additional information revealed to the learning algorithm, the distinguishing task becomes easier. However, in many examples, we show that revealing a significant amount of information to a learning algorithm that only has external classical memory will not significantly help its distinguishing power.

Suppose that after revealing the information ξ^* to the learning algorithm, the conditional probability for whether the unknown state/process X is X_0 (null hypothesis) or one of $(\xi^*, \chi) \in \mathcal{X} \setminus \{X_0\}$ (alternative hypothesis) is still uniform, i.e., $1/2$ and $1/2$. Then similar to when the information is not revealed, the success probability of any procedure to distinguish between null and alternative hypothesis is upper bounded by

$$\frac{1}{2}+\frac{1}{2}\mathrm{TV}\left(p_{X_0}, \mathop{\mathbb{E}}_{(\xi^*,\chi)\in\mathcal{X}\setminus\{X_0\}} p_{(\xi^*,\chi)}\right) = \frac{1}{2}+\frac{1}{4}\sum_{\ell}\left|p_{X_0}(\ell) - \mathop{\mathbb{E}}_{(\xi^*,\chi)\in\mathcal{X}\setminus\{X_0\}} p_{(\xi^*,\chi)}(\ell)\right|. \tag{3.25}$$

When ξ^* is chosen randomly, the average success probability is upper bounded by

$$\frac{1}{2}+\frac{1}{2}\mathop{\mathbb{E}}_{\xi^*}\mathrm{TV}\left(p_{X_0}, \mathop{\mathbb{E}}_{(\xi^*,\chi)\in\mathcal{X}\setminus\{X_0\}} p_{(\xi^*,\chi)}\right). \tag{3.26}$$

Again, similar to the discussion before, in order to achieve a prediction accuracy of $p \geq 1/2$, we need to satisfy the inequality

$$\mathop{\mathbb{E}}_{\xi^*}\mathrm{TV}\left(p_{X_0}, \mathop{\mathbb{E}}_{(\xi^*,\chi)\in\mathcal{X}\setminus\{X_0\}} p_{(\xi^*,\chi)}\right) \geq 2p-1. \tag{3.27}$$

The left-hand side of the above inequality can be upper bounded by a monotonically increasing function of T, hence we can obtain a lower bound on T. Note that by Jensen's inequality, the left hand side of the above inequality is larger than the left hand side in Eq. (3.17), so we will obtain a weaker lower bound on T. This makes sense because making accurate predictions with partially-revealed information is easier.

Presence of noise

So far, we have considered protocols for learning quantum states or quantum processes in the absence of noise. There are several forms of noise we can consider: (i) noise on input states, (ii) noise on the POVMs which are measured, and (iii) noise on the quantum process (if there is one). Let us prove the following result:

Theorem 6 (Noise cannot decrease the lower bound)**.** *If the upper bound*

$$\mathrm{TV}\left(\mathop{\mathbb{E}}_{X\in\mathcal{A}} p_X, \mathop{\mathbb{E}}_{X\in\mathcal{B}} p_X\right) \leq 2p-1 \tag{3.28}$$

holds for all learning protocols with a classical memory, then this same bound holds for all learning protocols with a classical memory in the presence of noise. Because the upper bound on total variation distance applies when noise is present, so does the lower bound on the number of experiments needed to achieve the distinguishing task.

Proof. Consider first the setting of learning an unknown physical state ρ. Suppose we have a learning protocol with a classical memory described by a learning tree

$\mathcal{T}$. At node u in the protocol, we measure the state ρ with the POVM $\{F_s^u\}_s$. We will measure the sth outcome with probability

$$\mathrm{tr}(F_s^u \rho)\,. \tag{3.29}$$

If there is noise on ρ, we can use $\rho \mapsto \mathcal{N}[\rho]$ for some noise quantum process $\mathcal{N}$. Likewise if there is noise on the POVM, we can use $F_s^u \mapsto \mathcal{M}^\dagger[F_s^u]$ for a noise quantum process $\mathcal{M}$. Then the probability of the sth outcome is instead

$$\mathrm{tr}(\mathcal{M}^\dagger[F_s^u]\,\mathcal{N}[\rho]) = \mathrm{tr}((\mathcal{N}^\dagger \circ \mathcal{M}^\dagger)[F_s^u]\,\rho)\,. \tag{3.30}$$

But $\{(\mathcal{N}^\dagger \circ \mathcal{M}^\dagger)[F_s^u]\}_s$ also forms a POVM. We can apply this same argument to each node in the tree; note that the noise channels can be node-dependent. The result is that we simply get a new learning tree with classical memory, with POVM's augmented by the noise channels. But since by hypothesis (3.28) holds for all learning protocols with a classical memory, the bound evidently still holds in the noisy setting.

In the setting where we are learning a physical process $\mathcal{E}$, the argument is similar. Given a learning tree $\mathcal{T}$ for learning the physical process, at node u we (i) prepare the state ρ^u, (ii) apply the physical process $\mathcal{E}$, and (iii) measure with the POVM $\{F_s^u\}_s$ and obtain outcome s with probability

$$\mathrm{tr}(F_s^u\,\mathcal{E}[\rho^u])\,. \tag{3.31}$$

If the initial state is noisy, we can implement this by a channel mapping $\rho^u \mapsto \mathcal{N}[\rho^u]$. If $\mathcal{E}$ is noisy, this can be implemented by $\mathcal{E} \mapsto \mathcal{D} \circ \mathcal{E} \circ \mathcal{F}$. Finally, if the POVM is noisy, we can implement this by $F_s^u \mapsto \mathcal{M}^\dagger[F_s^u]$. In these circumstances, the probability of the sth outcome is instead

$$\mathrm{tr}(\mathcal{M}^\dagger[F_s^u]\,(\mathcal{D} \circ \mathcal{E} \circ \mathcal{F})[\mathcal{N}[\rho^u]]) = \mathrm{tr}((\mathcal{D}^\dagger \circ M^\dagger)[F_s^u]\,\mathcal{E}[\mathcal{F} \circ \mathcal{N}[\rho^u]])\,. \tag{3.32}$$

But $\{(\mathcal{D}^\dagger \circ M^\dagger)[F_s^u]\}_s$ also forms a POVM, and $(\mathcal{F} \circ \mathcal{N})[\rho^u]$ is also a valid choice of input state. The same argument can be used for noise channels applied at every node in the tree, and furthermore the noise can be node-dependent. The result is that we just get a modified learning tree with classical memory, which by assumption satisfies (3.28), as desired. □

In summary, we have shown that if a task is hard for all learning protocols with classical memory, then the task is still just as hard (if not harder) in the presence of noise. Thus the presence of noise always makes learning with classical machines more challenging.

Part II

Learning with classical machines

Chapter 4

PREDICTING MANY PROPERTIES OF QUANTUM SYSTEMS

Making predictions based on empirical observations is a central topic in statistical learning theory and is at the heart of many scientific disciplines, including quantum physics. There, predictive tasks, like estimating target fidelities, verifying entanglement, and measuring correlations, are essential for building, calibrating and controlling quantum systems. Recent advances in the size of quantum platforms (Preskill, 2018) have pushed traditional prediction techniques — like quantum state tomography — to the limit of their capabilities. This is mainly due to a curse of dimensionality: the number of parameters needed to describe a quantum system scales exponentially with the number of its constituents. Moreover, these parameters cannot be accessed directly, but must be estimated by measuring the system. An informative quantum mechanical measurement is both destructive (wave-function collapse) and only yields probabilistic outcomes (Born's rule). Hence, many identically prepared samples are required to estimate accurately even a single parameter of the underlying quantum state. Furthermore, all measurement outcomes must be processed and stored in memory for subsequent prediction of relevant features. In summary, reconstructing a full description of a quantum system with n constituents (e.g. qubits) necessitates a number of measurement repetitions exponential in n, as well as an exponential amount of classical memory and computing power.

Several approaches have been proposed to overcome this fundamental scaling problem. These include matrix product state (MPS) tomography (Cramer et al., 2010) and neural network tomography (Torlai, Mazzola, et al., 2018; Carrasquilla, Torlai, et al., 2019). Both only require a polynomial number of samples, provided that the underlying state has suitable properties. However, for general quantum systems, these techniques still require an exponential number of samples.

Pioneering a conceptually very different line of research, Aaronson (Aaronson, 2018) pointed out that demanding full classical descriptions of quantum systems may be excessive for many concrete tasks. Instead it is often sufficient to accurately predict certain properties of the quantum system. In quantum mechanics, interesting properties are often *linear* functions of the underlying density matrix ρ, such as the expectation values $\{o_i\}$ of a set of observables $\{O_i\}$:

$$o_i(\rho) = \mathrm{trace}(O_i \rho) \quad 1 \leq i \leq M. \tag{4.1}$$

The fidelity with a pure target state, entanglement witnesses, and the probability distribution governing the possible outcomes of a measurement are all examples that fit this framework. A *nonlinear* function of ρ such as entanglement entropy, may also be of interest. Aaronson coined the term (Aaronson, 2018; Aaronson and Rothblum, 2019) *shadow tomography*[1] for the task of predicting properties without necessarily fully characterizing the quantum state, and he showed that a polynomial number of state copies already suffice to predict an exponential number of target functions. While very efficient in terms of samples, Aaronson's procedure is very demanding in terms of quantum hardware — a concrete implementation of the proposed protocol requires exponentially long quantum circuits that act collectively on all the copies of the unknown state stored in a quantum memory.

4.1 Central ideas of classical shadow tomography

In this section, we present the key ideas of classical shadow tomography. In the next sections, we will dive more deeply into understanding classical shadows.

We restrict attention to n-qubit systems, and ρ is a fixed but unknown quantum state in $d = 2^n$ dimensions. To extract meaningful information, we repeatedly perform a simple measurement procedure: apply a random unitary to rotate the state ($\rho \mapsto U\rho U^\dagger$) and perform a computational-basis measurement. The unitary U is selected randomly from a fixed ensemble. Upon receiving the n-bit measurement outcome $|\hat{b}\rangle \in \{0,1\}^n$, we store an (efficient) classical description of $U^\dagger |\hat{b}\rangle\!\langle\hat{b}| U$ in classical memory. It is instructive to view the average (over both the choice of unitary and the outcome distribution) mapping from ρ to its classical snapshot $U^\dagger |\hat{b}\rangle\!\langle\hat{b}| U$ as a quantum channel:

$$\mathbb{E}\left[U^\dagger |\hat{b}\rangle\!\langle\hat{b}| U\right] = \mathcal{M}(\rho) \implies \rho = \mathbb{E}\left[\mathcal{M}^{-1}\left(U^\dagger |\hat{b}\rangle\!\langle\hat{b}| U\right)\right]. \tag{4.2}$$

This quantum channel $\mathcal{M}$ depends on the ensemble of (random) unitary transformations. Although the inverted channel $\mathcal{M}^{-1}$ is not physical (it is not completely positive), we can still apply $\mathcal{M}^{-1}$ to the (classically stored) measurement outcome $U^\dagger |\hat{b}\rangle\!\langle\hat{b}| U$ in a completely classical post-processing step.[2] In doing so, we produce

[1] According to Ref. (Aaronson, 2018) it was actually S.T. Flammia who originally suggested the name shadow tomography.

[2] $\mathcal{M}$ is invertible if the ensemble of unitary transformations defines a tomographically complete set of measurements.

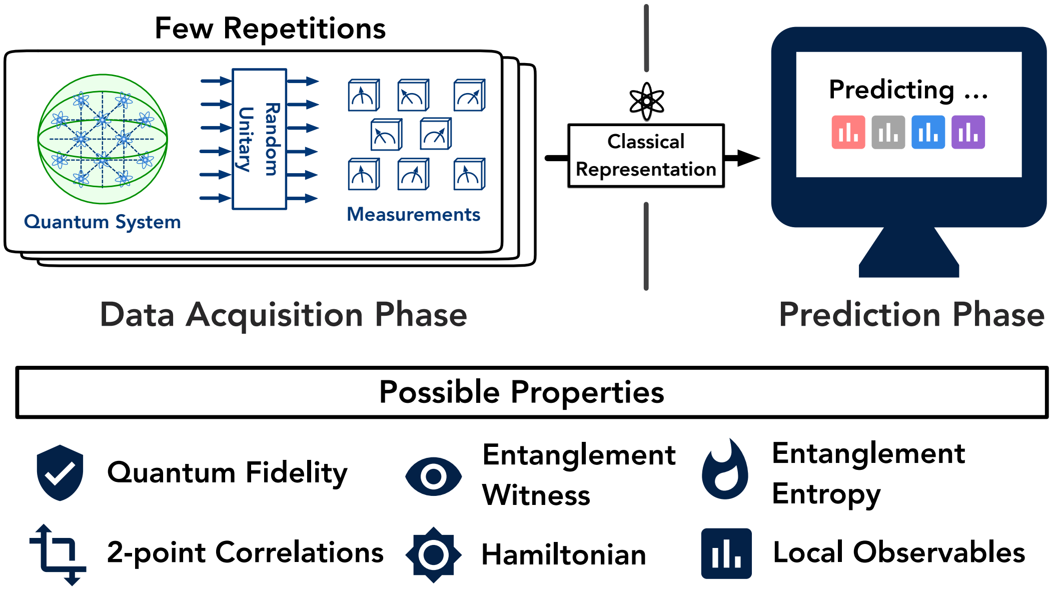


Figure 4.1: An illustration for constructing a classical representation, the *classical shadow*, of a quantum system from randomized measurements. In the data acquisition phase, we perform a random unitary evolution and measurements on independent copies of an n-qubit system to obtain a classical representation of the quantum system — the *classical shadow*. Such classical shadows facilitate accurate prediction of a large number of different properties using a simple median-of-means protocol.

a single classical snapshot $\hat{\rho} = \mathcal{M}^{-1}\left(U^\dagger|\hat{b}\rangle\langle\hat{b}|U\right)$ of the unknown state ρ from a single measurement. By construction, this snapshot exactly reproduces the underlying state in expectation (over both unitaries and measurement outcomes): $\mathbb{E}[\hat{\rho}] = \rho$. Repeating this procedure N times results in an array of N independent, classical snapshots of ρ:

$$\mathsf{S}(\rho;N) = \left\{\hat{\rho}_1 = \mathcal{M}^{-1}\left(U_1^\dagger\,|\hat{b}_1\rangle\langle\hat{b}_1|\,U_1\right),\ldots,\hat{\rho}_N = \mathcal{M}^{-1}\left(U_N^\dagger\,|\hat{b}_N\rangle\langle\hat{b}_N|\,U_N\right)\right\}. \tag{4.3}$$

We call this array the *classical shadow* of ρ. Classical shadows of sufficient size N are expressive enough to predict many properties of the unknown quantum state efficiently. To avoid outlier corruption, we split the classical shadow up into equally-sized chunks and construct several independent sample mean estimators. Subsequently, we predict linear function values (4.1) via *median of means estimation* (Jerrum, Leslie G. Valiant, and V. V. Vazirani, 1986; Nemirovsky and Yudin, 1983). This procedure is summarized in Algorithm 1. For many physically relevant properties O_i and measurement channels $\mathcal{M}$, Algorithm 1 can be carried out very

Algorithm 1 *Median of means prediction* based on a classical shadow $\mathsf{S}(\rho, N)$.

```
function LinearPredictions($O_1, \ldots, O_M, \mathsf{S}(\rho; N), K$)
    Import $\mathsf{S}(\rho; N) = [\hat{\rho}_1, \ldots, \hat{\rho}_N]$          ▷ Load classical shadow
    Split the shadow into $K$ equally-sized parts and set          ▷ Construct $K$ estimators of $\rho$
        $$\hat{\rho}_{(k)} = \frac{1}{\lfloor N/K \rfloor} \sum_{i=(k-1)\lfloor N/K \rfloor + 1}^{k\lfloor N/K \rfloor} \hat{\rho}_i$$
    for $i = 1$ to $M$ do
        Output $\hat{o}_i(N, K) = \mathrm{median}\left\{\mathrm{tr}\left(O_i \hat{\rho}_{(1)}\right), \ldots, \mathrm{tr}\left(O_i \hat{\rho}_{(K)}\right)\right\}$.          ▷ Median of means
```

efficiently without explicitly constructing the large matrix $\hat{\rho}_i$.

The median of means prediction with classical shadows can be defined for any distribution of random unitary transformations. Two prominent examples are: (i) random n-qubit Clifford circuits; and (ii) tensor products of random single-qubit Clifford circuits. Example (i) results in a clean and powerful theory, but also practical drawbacks, because $n^2/\log(n)$ entangling gates are needed to sample from n-qubit Clifford unitaries. The corresponding inverted quantum channel is $\mathcal{M}_n^{-1}(X) = (2^n + 1)X - \mathbb{I}$. Example (ii) is equivalent to measuring each qubit independently in a random Pauli basis. Such measurements can be routinely carried out in many experimental platforms. The corresponding inverted quantum channel is $\mathcal{M}_P^{-1} = \bigotimes_{i=1}^n \mathcal{M}_1^{-1}$. We refer to examples (i) / (ii) as random Clifford / Pauli measurements, respectively. In both cases, the resulting classical shadow can be stored efficiently in a classical memory using the stabilizer formalism.

Classical shadow tomography satisfies the following rigorous guarantee.

Theorem 7 (informal version)**.** *Classical shadows of size N suffice to predict M* arbitrary *linear target functions* $\mathrm{tr}(O_1\rho), \ldots, \mathrm{tr}(O_M\rho)$ *up to additive error ϵ given that $N \geq$ (order)* $\log(M) \max_i \|O_i\|_{\mathrm{shadow}}^2 / \epsilon^2$. *The definition of the norm* $\|O_i\|_{\mathrm{shadow}}$ *depends on the ensemble of unitaries used to create the classical shadow.*

Theorem 7 is most powerful when the linear functions have a bounded norm that is independent of system size. In this case, classical shadows allow for predicting a large number of properties from only a logarithmic number of quantum measurements. The norm $\|O_i\|_{\mathrm{shadow}}$ in Theorem 7 plays an important role in defining the space of linear functions that can be predicted efficiently.

For random Clifford measurements, $\|O\|_{\text{shadow}}^2$ is closely related to the Hilbert-Schmidt norm $\operatorname{tr}(O^2)$. As a result, a large collection of (global) observables with a bounded Hilbert-Schmidt norm can be predicted efficiently. For random Pauli measurements, the norm scales exponentially in the locality of the observable, not the actual number of qubits. For an observable O_i that acts non-trivially on (at most) k qubits, $\|O_i\|_{\text{shadow}}^2 \leq 4^k \|O_i\|_\infty^2$, where $\|\cdot\|_\infty$ denotes the operator norm[3]. This guarantees the accurate prediction of many local observables from only a much smaller number of measurements.

The following paragraphs present three illustrative examples for using classical shadow tomography.

Quantum fidelity estimation. Suppose we wish to certify that an experimental device prepares a desired n-qubit state. Typically, this target state $|\psi\rangle\langle\psi|$ is pure and highly structured, e.g. a a GHZ state (Greenberger, Horne, and Zeilinger, 1989) for quantum communication protocols, or a toric code ground state (Dennis et al., 2002a) for fault-tolerant quantum computation. Theorem 7 asserts that a classical shadow (Clifford measurements) of dimension-independent size suffices to accurately predict the fidelity of *any* state in the lab with *any* pure target state. This improves on the best existing result on direct fidelity estimation (Steven T. Flammia and Y.-K. Liu, 2011) which requires $O(2^n/\epsilon^4)$ samples in the worst case. Moreover, a classical shadow of polynomial size allows for estimating an exponential number of (pure) target fidelities all at once.

Entanglement verification. Fidelities with pure target states can also serve as (bipartite) *entanglement witnesses* (Gühne and Tóth, 2009). For every (bipartite) entangled state ρ, there exists a constant α and an observable $O = |\psi\rangle\langle\psi|$ such that $\operatorname{tr}(O\rho) > \alpha \geq \operatorname{tr}(O\rho_s)$, for all (bipartite) separable states ρ_s. Establishing $\operatorname{tr}(O\rho) > \alpha$ verifies the existence of entanglement in the state ρ. Any $O = |\psi\rangle\langle\psi|$ that satisfies the above condition is known as an entanglement witness for the state ρ. Classical shadows (Clifford measurements) of logarithmic size allow for checking a large number of potential entanglement witnesses simultaneously.

Predicting expectation values of local observables. Many near-term applications of quantum devices rely on repeatedly estimating a large number of local observables. For example, low-energy eigenstates of a many-body Hamiltonian may be prepared

[3] This scaling can be further improved to 3^k if O_i is a tensor product of k single-qubit observables.

and studied using a variational method, in which the Hamiltonian, a sum of local terms, is measured many times. Classical shadows constructed from a logarithmic number of random Pauli measurements can efficiently estimate polynomially many such local observables. Because only single-qubit Pauli measurements suffice, this measurement procedure is highly efficient. Potential applications include quantum chemistry (Kandala et al., 2017) and lattice gauge theory (Kokail et al., 2019).

The non-example above raises an important question: does the scaling of the required number of measurements with the Hilbert-Schmidt norm or with the locality of observables arise from a fundamental limitation, or is it merely an artifact of prediction with classical shadows? A rigorous analysis reveals that this scaling is no mere artifact; rather, it stems from information-theoretic reasons.

Theorem 8 (informal version)**.** *Any procedure based on single-copy measurements, that can predict* any *M linear functions* $\mathrm{tr}(O_i\rho)$ *up to additive error ϵ, requires at least (order)* $\log(M)\max_i \|O_i\|^2_{\text{shadow}}/\epsilon^2$ *measurements.*

Here, $\|O_i\|^2_{\text{shadow}}$ could be taken as the Hilbert-Schmidt norm $\mathrm{tr}(O_i^2)$ or as a function scaling exponentially in the locality of O_i. The proof results from embedding the abstract prediction procedure into a communication protocol. Quantum information theory imposes fundamental restrictions on any quantum communication protocol and allows us to deduce stringent lower bounds. We refer to Section 4.11 and 4.12 for details and proofs.

The two main technical results complement each other nicely. Theorem 7 equips classical shadows with a constructive performance guarantee: an order of

$$\log(M)\max_i \|O_i\|^2_{\text{shadow}}/\epsilon^2 \tag{4.4}$$

single-copy measurements suffice to accurately predict an *arbitrary* collection of M target functions. Theorem 8 highlights that this number of measurements is unavoidable in general.

The classical shadow $S(\rho;N)=\{\hat\rho_1,\ldots,\hat\rho_N\}$ of the unknown quantum state ρ may also be used to predict non-linear functions $f(\rho)$. We illustrate this with a quadratic function $f(\rho)=\mathrm{tr}(O\rho\otimes\rho)$, where O acts on two copies of the state. Because $\hat\rho_i$ is equal to the quantum state ρ in expectation, one could predict $\mathrm{tr}(O\rho\otimes\rho)$ using two independent snapshots $\hat\rho_i,\hat\rho_j, i\neq j$. Because of independence, $\mathrm{tr}(O\hat\rho_i\otimes\hat\rho_j)$ correctly predicts the quadratic function in expectation:

$$\mathbb{E}\,\mathrm{tr}(O\hat\rho_i\otimes\hat\rho_j)=\mathrm{tr}(O\,\mathbb{E}\,\hat\rho_i\otimes\mathbb{E}\,\hat\rho_j)=\mathrm{tr}(O\rho\otimes\rho). \tag{4.5}$$

To reduce the prediction error, we use N independent snapshots and symmetrize over all possible pairs: $\frac{1}{N(N-1)}\sum_{i\neq j}\operatorname{tr}(O\hat{\rho}_i\otimes\hat{\rho}_j)$. We then repeat this procedure several times and form their median to further reduce the likelihood of outlier corruption (similar to median of means). Rigorous performance guarantees are given in Supplementary Section 4.10. This approach readily generalizes to higher order polynomials using U-statistics (Hoeffding, 1992).

One particularly interesting nonlinear function is the second-order Rényi entanglement entropy, given by $-\log(\operatorname{tr}(\rho_A^2))$, where A is a subsystem of the n-qubit quantum system. We can rewrite the argument in the log as $\operatorname{tr}(\rho_A^2)=\operatorname{tr}(S_A\rho\otimes\rho)$ — where S_A is the local swap operator of two copies of the subsystem A — and use classical shadows to obtain very accurate predictions. The required number of measurements scales exponentially in the size of the subsystem A, but is independent of total system size. Probing this entanglement entropy is a useful task and a highly efficient specialized approach has been proposed in (Brydges et al., 2019). We compare this *Brydges et al. method* to classical shadows in the numerical experiments.

For nonlinear functions, unlike linear ones, we have not derived an information-theoretic lower bound on the number of measurements needed, though it may be possible to do so by generalizing our methods.

4.2 Data acquisition and classical shadows

We now dive deeper into the general-purpose strategy of classical shadow tomography for predicting many properties of this unknown state. To extract meaningful information about ρ, we need to perform a collection of measurements.

Definition 2 (measurement primitive)**.** *We can apply a restricted set of unitary evolutions $\rho\mapsto U\rho U^\dagger$, where U is chosen from an ensemble $\mathcal{U}$. Subsequently, we can measure the rotated state in the computational basis $\{|b\rangle : b\in\{0,1\}^n\}$. Moreover, we assume that this collection is tomographically complete, i.e. for each $\sigma\neq\rho$ there exist $U\in\mathcal{U}$ and b such that $\langle b|U\sigma U^\dagger|b\rangle\neq\langle b|U\rho U^\dagger|b\rangle$.*

Based on this primitive, we repeatedly perform a simple randomized measurement procedure: randomly rotate the state $\rho\mapsto U\rho U^\dagger$ and perform a computational basis measurement. Then, after the measurement, we apply the inverse of U to the resulting computational basis state. This procedure collapses ρ to

$$U^\dagger|\hat{b}\rangle\langle\hat{b}|U \quad\text{where}\quad \Pr\left[\hat{b}=b\right]=\langle b|U\rho U^\dagger|b\rangle,\ b\in\{0,1\}^n \quad\text{(Born's rule).} \tag{4.6}$$

This random snapshot contains valuable information about ρ in expectation:

$$\mathbb{E}\left[U^\dagger|\hat{b}\rangle\langle\hat{b}|U\right] = \mathbb{E}_{U\sim\mathcal{U}} \sum_{b\in\{0,1\}^n} \langle b|U\rho U^\dagger|b\rangle U^\dagger|b\rangle\langle b|U = \mathcal{M}(\rho). \tag{4.7}$$

For any unitary ensemble $\mathcal{U}$, this relation describes a quantum channel $\rho \mapsto \mathcal{M}(\rho)$. Tomographic completeness ensures that $\mathcal{M}$ — viewed as a linear map — has a unique inverse $\mathcal{M}^{-1}$ and we set

$$\hat{\rho} = \mathcal{M}^{-1}\left(U^\dagger|\hat{b}\rangle\langle\hat{b}|U\right) \qquad \text{(classical shadow)}. \tag{4.8}$$

The classical shadow is a modified post-measurement state that has a unit trace, but need not be positive semi-definite. However, it is designed to reproduce the underlying state ρ exactly in expectation: $\mathbb{E}\left[\hat{\rho}\right] = \rho$. This classical shadow $\hat{\rho}$ corresponds to the linear inversion (or least squares) estimator of ρ in the single-shot limit. Linear inversion estimators have been used to perform full quantum state tomography (Sugiyama, P. S. Turner, and Murao, 2013; Guta et al., 2020), where an exponential number of measurements is needed. We wish to show that $\hat{\rho}$ can predict many properties from only very few measurements.

4.3 Predicting linear functions with classical shadows

Classical shadows are well suited to predict *linear* functions in the unknown state ρ:

$$o_i = \mathrm{tr}\left(O_i\rho\right) \quad 1 \leq i \leq M. \tag{4.9}$$

To achieve this goal, we simply replace the (unknown) quantum state ρ by a classical shadow $\hat{\rho}$. Since classical shadows are random, this produces a random variable that yields the correct prediction in expectation:

$$\hat{o}_i = \mathrm{tr}\left(O_i\hat{\rho}\right) \quad \text{obeys} \quad \mathbb{E}\left[\hat{o}\right] = \mathrm{tr}\left(O_i\rho\right). \tag{4.10}$$

Fluctuations of $\hat{o}$ around this desired expectation are controlled by the variance.

Lemma 5. *Fix O and set $\hat{o} = \mathrm{tr}\left(O\hat{\rho}\right)$, where $\hat{\rho}$ is a classical shadow* (4.8). *Then*

$$\mathrm{Var}\left[\hat{o}\right] = \mathbb{E}\left[\left(\hat{o} - \mathbb{E}\left[\hat{o}\right]\right)^2\right] \leq \left\| O - \tfrac{\mathrm{tr}(O)}{2^n}\mathbb{I} \right\|_{\mathrm{shadow}}^2. \tag{4.11}$$

The norm $\|\cdot\|_{\mathrm{shadow}}$ only depends on the measurement primitive:

$$\|O\|_{\mathrm{shadow}} = \max_{\sigma:\ \mathit{state}} \left(\mathbb{E}_{U\sim\mathcal{U}} \sum_{b\in\{0,1\}^n} \langle b|U\sigma U^\dagger|b\rangle \langle b|U\mathcal{M}^{-1}\left(O\right)U^\dagger|b\rangle^2\right)^{1/2}. \tag{4.12}$$

It is easy to check that $\|O\|_{\text{shadow}}$ is nonnegative and homogeneous ($\|0\|_{\text{shadow}} = 0$). After some work, one can verify that this expression also obeys the triangle inequality, and so is indeed a norm.

Proof. Classical shadows have unit trace by construction ($\mathrm{tr}(\hat{\rho}) = 1$). This feature implies that the variance only depends on the traceless part $O_0 = O - \frac{\mathrm{tr}(O)}{2^n}\mathbb{I}$ of O, not O itself:

$$\hat{o} - \mathbb{E}[\hat{o}] = \mathrm{tr}\left(O\hat{\rho}\right) - \mathrm{tr}\left(O\rho\right) = \mathrm{tr}\left(O_0\hat{\rho}\right) - \mathrm{tr}\left(O_0\rho\right). \tag{4.13}$$

Moreover, it is easy to check that the inverse of $\mathcal{M}$ (4.7) is self-adjoint,

$$\mathrm{tr}\left(X\mathcal{M}^{-1}(Y)\right) = \mathrm{tr}\left(\mathcal{M}^{-1}(X)Y\right) \tag{4.14}$$

for any pair of matrices X, Y with compatible dimension. These two observations allow us to rewrite the variance in the following fashion:

$$\begin{aligned} \mathrm{Var}\left[\hat{o}\right] &= \mathbb{E}\left[(\hat{o} - \mathbb{E}\hat{o})^2\right] = \mathbb{E}\left[(\mathrm{tr}(O_0\hat{\rho}))^2\right] - \left(\mathrm{tr}\left(O_0\,\mathbb{E}\left[\hat{\rho}\right]\right)\right)^2 && (4.15)\\ &= \mathbb{E}\left[\langle\hat{b}|U\mathcal{M}^{-1}(O_0)U^\dagger|\hat{b}\rangle^2\right] - \left(\mathrm{tr}\left(O_0\rho\right)\right)^2. && (4.16)\end{aligned}$$

Classical shadows arise from mixing two types of randomness: (i) a (classical) random choice of unitary $U \sim \mathcal{U}$ and (ii) a random choice of computational basis state $|\hat{b}\rangle$ that is governed by Born's rule (4.6). Inserting the average over computational basis states produces a (squared) norm that closely resembles the advertised expression, but does depend on the underlying state:

$$\mathbb{E}\langle\hat{b}|U\mathcal{M}^{-1}(O_0)U^\dagger|\hat{b}\rangle^2 = \mathbb{E}_{U\sim\mathcal{U}}\sum_{b\in\{0,1\}^n}\langle b|U\rho U^\dagger|b\rangle\langle b|U\mathcal{M}^{-1}(O_0)U^\dagger|b\rangle^2. \tag{4.17}$$

Maximizing over all possible states σ removes this implicit dependence and produces a universal upper bound on the variance. Ignoring the subtraction of $\left(\mathrm{tr}\left(O_0\rho\right)\right)^2$ (which can only make the bound tighter), we obtain (4.11). □

Lemma 5 sets the stage for successful linear function estimation with classical shadows. A single classical shadow (4.8) correctly predicts *any* linear function $o_i = \mathrm{tr}(O_i\rho)$ in expectation. Convergence to this desired target can be boosted by forming empirical averages of multiple independent shadow predictions. The *empirical mean* is the canonical example for such a procedure. Construct N independent classical shadows $\hat{\rho}_1, \ldots, \hat{\rho}_N$ and set

$$\hat{o}_i(N, 1) = \frac{1}{N}\sum_{j=1}^{N}\mathrm{tr}\left(O_i\hat{\rho}_j\right). \tag{4.18}$$

Each summand is an independent random variable with correct expectation and variance bounded by Lemma 5. Convergence to the expectation value $\mathrm{tr}(O_i\rho)$ can be controlled by classical concentration arguments (e.g. Chernoff or Hoeffding inequalities). In order to achieve a failure probability of (at most) δ, the number of samples must scale like $N = \mathrm{Var}\left[\hat{o}_i\right]/(\delta\epsilon^2)$. While the scaling in variance and approximation accuracy ϵ is optimal, the dependence on $1/\delta$ is particularly bad. Unfortunately, this feature of sample mean estimators cannot be avoided without imposing additional assumptions (that do not apply to classical shadows). *Median of means* (Nemirovsky and Yudin, 1983; Jerrum, Leslie G. Valiant, and V. V. Vazirani, 1986) is a conceptually simple trick that addresses this issue. Instead of using all samples to construct a single empirical mean (4.18), construct K independent sample means and form their median:

$$\hat{o}_i(N,K) = \mathrm{median}\left\{\hat{o}_i^{(1)}(N,1),\ldots,\hat{o}_i^{(K)}(N,1)\right\} \tag{4.19}$$

$$\text{where} \quad \hat{o}_i^{(k)} = \tfrac{1}{N}\sum_{j=N(k-1)+1}^{Nk} \mathrm{tr}\left(O_i\hat{\rho}_j\right) \tag{4.20}$$

for $1 \leq k \leq K$. This estimation technique requires NK samples in total, but it is much more robust with respect to outlier corruption. Indeed, $|\hat{o}(N,K) - \mathrm{tr}(O\rho)| > \epsilon$ if and only if more than half of the empirical means individually deviate by more than ϵ. The probability associated with such an undesirable event decreases exponentially with the number of batches K. This results in an exponential improvement over sample mean estimation in terms of failure probability. The main result of derandomizing randomized measurements capitalizes on this improvement.

Theorem 9. *Fix a measurement primitive $\mathcal{U}$, a collection $O_1,\ldots,O_M$ of $2^n \times 2^n$ Hermitian matrices and accuracy parameters $\epsilon,\delta \in [0,1]$. Set*

$$K = 2\log(2M/\delta) \quad \textit{and} \quad N = \frac{34}{\epsilon^2}\max_{1\leq i\leq M} \|O_i - \tfrac{\mathrm{tr}(O_i)}{2^n}\mathbb{I}\|_{\mathrm{shadow}}^2, \tag{4.21}$$

where $\|\cdot\|_{\mathrm{shadow}}$ denotes the norm defined in Eq. (4.12)*. Then, a collection of NK independent classical shadows allow for accurately predicting all features via median of means prediction* (4.20)*:*

$$|\hat{o}_i(N,K) - \mathrm{tr}\left(O_i\rho\right)| \leq \epsilon \quad \textit{for all } 1 \leq i \leq M \tag{4.22}$$

with probability at least $1-\delta$.

Proof. The claim follows from combining the variance estimates from Lemma 5 with a rigorous performance guarantee for median of means estimation (Nemirovsky and Yudin, 1983; Jerrum, Leslie G. Valiant, and V. V. Vazirani, 1986): Let X be a random variable with variance σ^2. Then, K independent sample means of size $N = 34\sigma^2/\epsilon^2$ suffice to construct a median of means estimator $\hat{\mu}(N, K)$ that obeys $\Pr\left[|\hat{\mu}(N, K) - \mathbb{E}[X]| \geq \epsilon\right] \leq 2\mathrm{e}^{-K/2}$ for all $\epsilon > 0$. The parameters N and K are chosen such that this general statement ensures $\Pr\left[|\hat{o}_i(N, K) - \mathrm{tr}\,(O_i\rho)| \geq \epsilon\right] \leq \frac{\delta}{M}$ for all $1 \leq i \leq M$. Apply a union bound over all M failure probabilities to deduce the claim. □

Remark 1 (Constants in Theorem 9)**.** *The numerical constants featuring in N and K result from a conservative (worst case) argument that is designed to be simple, not tight. We expect that the actual constants are* much smaller *in practice.*

Each classical shadow is the result of a single quantum measurement on ρ. Viewed from this angle, Theorem 9 asserts that a total of

$$N_{\mathrm{tot}} = \mathcal{O}\left(\frac{\log(M)}{\epsilon^2} \max_{1\leq i\leq M} \left\| O_i - \tfrac{\mathrm{tr}(O_i)}{2^n}\mathbb{I} \right\|_{\mathrm{shadow}}^2\right) \quad \text{(sample complexity)} \tag{4.23}$$

measurement repetitions suffice to accurately predict a collection of M linear target functions $\mathrm{tr}(O_i\rho)$.

Importantly, this sample complexity only scales logarithmically in the number of target functions M. Moreover, the problem dimension 2^n does not feature explicitly. The sample complexity does, however, depend on the measurement primitive via the norm $\|\cdot\|_{\mathrm{shadow}}$. This term reflects expressiveness and structure of the measurement primitive in question. This subtle point is best illustrated with two concrete examples. We defer technical derivations to subsequent sections and content ourselves with summarizing the important aspects here.

Example 1: Random Clifford measurements Clifford circuits are generated by CNOT, Hadamard and Phase gates and form the group $\mathrm{Cl}(2^n)$. The "random global Clifford basis" measurement primitive — $\mathcal{U} = \mathrm{Cl}(2^n)$ (endowed with uniform weights) — implies the following simple expression for classical shadows and the associated norm $\|\cdot\|_{\mathrm{shadow}}$:

$$\hat{\rho} = (2^n + 1)U^\dagger|\hat{b}\rangle\langle\hat{b}|U - \mathbb{I} \quad \text{and} \quad \left\| O - \tfrac{\mathrm{tr}(O)}{2^n}\mathbb{I} \right\|_{\mathrm{shadow}}^2 \leq 3\mathrm{tr}(O^2). \tag{4.24}$$

We refer to Section 4.9 for details and proofs. Combined with Eq. (4.23), this ensures that $\mathcal{O}(\log(M) \max_i \mathrm{tr}(O_i^2)/\epsilon^2)$ random global Clifford basis measurements suffice to accurately predict M linear functions. This prediction technique is most powerful, when the target functions have constant Hilbert-Schmidt norm. In this case, the sample rate is completely independent of the problem dimension 2^n. Prominent examples include estimating quantum fidelities (with pure states), or entanglement witnesses.

Example 2: Random Pauli measurements Although (global) Clifford circuits are believed to be much more tractable than general quantum circuits, they still feature entangling gates, like CNOT. Such gates are challenging to implement reliably on today's devices. The "random Pauli basis" measurement primitive takes this serious drawback into account and assumes that one is only able to apply single-qubit Clifford gates, i.e. $U = U_1 \otimes \cdots \otimes U_n \sim \mathcal{U} = \mathrm{Cl}(2)^{\otimes n}$ (endowed with uniform weights). This is equivalent to assuming that we can perform arbitrary Pauli (basis) measurements, i.e., measuring each qubit in the X-, Y- and Z-basis, respectively. Such basis measurements decompose nicely into tensor products ($U|\hat{b}\rangle = \bigotimes_{j=1}^n U_j|b_j\rangle$ for $b = (b_1, \ldots, b_n) \in \{0,1\}^n$) and respect locality. The associated classical shadows and the norm $\|\cdot\|_{\text{shadow}}$ inherit these desirable features:

$$\hat{\rho} = \bigotimes_{j=1}^{n} \left(3U_j^\dagger|\hat{b}_j\rangle\langle\hat{b}_j|U_j - \mathbb{I}\right) \quad \text{and} \quad \left\|O - \tfrac{\mathrm{tr}(O)}{2^n}\right\|_{\text{shadow}}^2 \leq 4^{\mathrm{locality}(O)}\|O\|_\infty^2. \tag{4.25}$$

Here, locality(O) counts the number of qubits on which O acts nontrivially. We refer to Section 4.9 for details and proofs. Combined with Eq. (4.23) this ensures that $O\left(\log(M)4^k/\epsilon^2\right)$ local Clifford (Pauli) basis measurements suffice to predict M bounded observables that are at most k-local. For observables that are the tensor product of k single-qubit observables, the sample complexity can be further improved to $O\left(\log(M)3^k/\epsilon^2\right)$. This prediction technique is most powerful when the target functions do respect some sort of locality constraint. Prominent examples include k-point correlators, or individual terms in a local Hamiltonian.

4.4 Information-theoretic optimality

These two examples complement each other nicely. Random Clifford measurements excel at performing useful subroutines in quantum computing and communication tasks, such as certifying (global) entanglement, which will be feasible using sufficiently advanced hardware. Their practical utility, however, hinges on the ability to execute circuits with many entangling gates. Random Pauli measurements, on the other hand, are much less demanding from a hardware perspective. In today's NISQ era, local Pauli operators can be accurately measured using available hardware platforms. While not well-suited for predicting global features, Pauli measurements excel at making local predictions. Furthermore, for both kinds of randomized measurements, linear prediction based on classical shadows saturates fundamental lower bounds from information theory.

Theorem 10 (random Clifford measurements; informal version)**.** Any *procedure based on a fixed set of single-copy measurements that can predict, with additive error* ϵ*, M arbitrary linear functions* $\mathrm{tr}(O_i\rho)$*, requires at least* $\Omega(\log(M)\max_i \mathrm{tr}(O_i^2)/\epsilon^2)$ *copies of the state* ρ*.*

Theorem 11 (random Pauli measurements; informal version)**.** Any *procedure based on a fixed set of single-copy local measurements that can predict, with additive error* ϵ*, M arbitrary k-local linear functions* $\mathrm{tr}(O_i\rho)$*, requires at least* $\Omega(\log(M)3^k/\epsilon^2)$ *copies of the state* ρ*.*

We refer to Section 4.11 (Clifford) and 4.12 (Pauli) for further context, details and proofs. In the random Pauli basis measurement setting, classical shadows provably saturate this lower bound only for tensor product observables. For general k-local observables, there is a small discrepancy between 4^k (upper bound) and 3^k (lower bound).

4.5 Predicting nonlinear functions with classical shadows

Feature prediction with classical shadows readily extends beyond the linear case. Here, we shall focus on quadratic functions, but the procedure and analysis readily extend to higher order polynomials. Every quadratic function in an unknown state ρ can be recast as a linear function acting on the tensor product $\rho \otimes \rho$:

$$\hat{o}_i = \mathrm{tr}\left(O_i \rho \otimes \rho\right) \quad 1 \leq i \leq M. \tag{4.26}$$

An immediate generalization of linear feature prediction with classical shadows suggests the following procedure. Take two independent snapshots $\hat{\rho}_1, \hat{\rho}_2$ of the unknown state ρ and set

$$\hat{o}_i = \mathrm{tr}\left(O_i \hat{\rho}_1 \otimes \hat{\rho}_2\right) \quad \text{such that} \quad \mathbb{E}\hat{o}_i = \mathrm{tr}\left(O_i \mathbb{E}\hat{\rho}_1 \otimes \mathbb{E}\hat{\rho}_2\right) = \mathrm{tr}\left(O_i \rho \otimes \rho\right) = o_i. \tag{4.27}$$

This random variable is designed to yield the correct target function in expectation. Similar to linear function prediction we can boost convergence to this desired target by forming empirical averages. To make the best of use of N samples, we average over all $N(N-1)$ (distinct) pairs:

$$\hat{o}_i(N,1) = \frac{1}{N(N-1)} \sum_{j \neq l} \operatorname{tr}\left(O_i \hat{\rho}_j \otimes \hat{\rho}_l\right). \tag{4.28}$$

This idea provides a systematic approach for constructing estimators for nonlinear (polynomial) functions. Estimators of this form always yield the desired target in expectation. For context, we point out that the estimator (4.28) closely resembles the sample variance, while estimators of higher order polynomials are known as *U-statistics* (Hoeffding, 1992). Fluctuations of $\hat{o}_i(N,1)$ around its desired expectation are once more controlled by the variance. U-statistics estimators are designed to minimize this variance and therefore considerably boost the rate of convergence.

Lemma 6. *Fix O and a sample size N. Then, the variance of the U-statistics estimator* (4.28) *obeys*

$$\begin{aligned}\operatorname{Var}[\hat{o}(N,1)] \leq \frac{2}{N}\Big(& \operatorname{Var}[\operatorname{tr}(O\hat{\rho}_1 \otimes \rho)] + \operatorname{Var}[\operatorname{tr}(O\rho \otimes \hat{\rho}_1)] \\ & + \frac{1}{N} \operatorname{Var}[\operatorname{tr}(O\hat{\rho}_1 \otimes \hat{\rho}_2)]\Big).\end{aligned} \tag{4.29}$$

We emphasize that this variance decreases with the number of samples N. This sets the stage for successful quadratic function prediction with classical shadows. Similar to the linear case, we will not use all samples to construct a single U-statistics estimator. Instead, we construct K of them and form their median:

$$\begin{aligned}\hat{o}_i(N,K) =& \text{median}\left\{\hat{o}_i^{(1)}(N,1), \ldots, \hat{o}_i^{(K)}(N,1)\right\}, \quad \text{where} \\ \hat{o}_i^{(k)}(N,1) =& \tfrac{1}{N(N-1)} \sum_{\substack{j \neq l \\ j,l \in \{N(k-1)+1,\ldots,Nk\}}} \operatorname{tr}\left(O_i \hat{\rho}_j \otimes \hat{\rho}_l\right) \quad \text{for } 1 \leq k \leq K.\end{aligned} \tag{4.30}$$

This renders the entire estimation procedure more robust to outliers and exponentially suppresses failure probabilities.

Theorem 12. *Fix a measurement primitive $\mathcal{U}$, a collection $O_1, \ldots, O_M$ of quadratic target functions and accuracy parameters $\epsilon, \delta \in [0,1]$. Set*

$$K = 2\log(2M/\delta) \quad \textit{and}$$

$$N = \frac{34}{\epsilon^2} \max_{1 \leq i \leq M} 8 \times \max\Bigg(\operatorname{Var}[\operatorname{tr}(O_i \rho \otimes \hat{\rho}_1)], \operatorname{Var}[\operatorname{tr}(O_i \hat{\rho}_1 \otimes \rho)], \tag{4.31}$$

$$\sqrt{\operatorname{Var}[\operatorname{tr}(O_i \hat{\rho}_1 \otimes \hat{\rho}_2)]}\Bigg). \tag{4.32}$$

Then, a collection of NK independent classical shadows allow for accurately predicting all quadratic features via the median of U-statistics estimators (4.30)*:*

$$|\hat{o}_i(N,K) - \mathrm{tr}\,(O_i\rho \otimes \rho)| \leq \epsilon \quad \textit{for all } 1 \leq i \leq M \tag{4.33}$$

with probability at least $1-\delta$*.*

Proof. The proof is similar to the argument for linear prediction. We combine the bound on the variance of U-statistics estimators from Lemma 6 with a rigorous performance guarantee for median estimation (Nemirovsky and Yudin, 1983; Jerrum, Leslie G. Valiant, and V. V. Vazirani, 1986). Let Z be a random variable with variance at most $\epsilon^2/34$. Then, setting $\hat{\mu} = \mathrm{median}\,\{Z_1, \ldots, Z_k\}$ produces an estimator that obeys $\Pr\,[|\hat{\mu} - \mathbb{E}\,[Z]| \geq \epsilon] \leq 2\mathrm{e}^{-K/2}$. The parameter N is chosen ensure that each $\hat{o}_i^{(k)}(N,1)$ has variance at most $\epsilon^2/34$. The parameter K is chosen such that each probability of failure is at most δ/M. The advertised statement then follows from taking a union bound over all M target estimations. □

Remark 2 (Constants in Theorem 12)**.** *The numerical constants featuring in N and K result from a conservative (worst case) argument that is designed to be simple, not tight. We expect that the actual constants are* much smaller *in practice.*

Theorem 12 is a general statement that provides upper bounds for the sample complexity associated with predicting quadratic target functions:

$$N_{\mathrm{tot}} = O\Big(\frac{\log(M)}{\epsilon^2} \max_{1\leq i\leq M} \max\Big(\mathrm{Var}[\mathrm{tr}(O_i\rho\otimes\hat{\rho}_1)], \mathrm{Var}[\mathrm{tr}(O_i\hat{\rho}_1\otimes\rho)] \tag{4.34}$$

$$\sqrt{\mathrm{Var}[\mathrm{tr}(O_i\hat{\rho}_1\otimes\hat{\rho}_2)]}\Big)\Big) \tag{4.35}$$

independent randomized measurements suffice to accurately predict a collection of M nonlinear target functions $\mathrm{tr}(O_i\rho\otimes\rho)$. This sampling rate once more depends on the measurement primitive and it is instructive to consider concrete examples.

Example 1: Random Pauli measurements We first discuss the practically more relvant example for today's NISQ era: classical shadows constructed from random single-qubit Pauli basis measurements. This measurement primitive remains well-suited for predicting *local* quadratic features $\mathrm{tr}(O\rho\otimes\rho)$. Suppose that O acts nontrivially on k qubits in the first state copy and on k qubits in the second state copy. Thus, when viewed as an observable for a $2n$-qubit system, O is $2k$-local. A technical argument shows that the maximum of the variances in Equation (4.34) is bounded by 4^k. We emphasize that this scaling is much better than the naive guess 4^{2k} – one of the key advantages of U-statistics. Hence we only need a total number of $N_{\mathrm{tot}} = \mathcal{O}(\log(M)4^k/\epsilon^2)$ random Pauli basis measurements to predict M quadratic functions $\mathrm{tr}(O_i\rho \otimes \rho)$. An important concrete application of this procedure is the prediction of subsystem Rényi-2 entanglement entropies.

Example 2: Random Clifford measurements Theorem 12 also applies to the global Clifford measurement primitive. There, the maximum of the variances in Equation (4.34) can be bounded by $\sqrt{9+6/2^n}\,\mathrm{tr}(O_i^2) \simeq 3\,\mathrm{tr}(O_i^2)$. Hence we only need a total number of $N_{\mathrm{tot}} = \mathcal{O}(\log(M)\max_i \mathrm{tr}(O_i^2)/\epsilon^2)$ random Clifford basis measurements to predict M quadratic functions $\mathrm{tr}(O_i\rho\otimes\rho)$. While a clean extension of linear feature prediction with Clifford basis measurements, the applicability of this result seems somewhat limited. Interesting global quadratic features tend to have prohibitively large Hilbert-Schmidt norms. The purity $\mathrm{tr}(\rho^2)$ provides an instructive non-example. It can be written as $\mathrm{tr}\left(S\rho\otimes\rho\right)$, where $S|\psi\rangle \otimes |\phi\rangle = |\phi\rangle \otimes |\psi\rangle$ denotes the swap operator. Alas, $\mathrm{tr}(S^2) = \mathrm{tr}(\mathbb{I}) = 2^n$ which scales exponentially in the number of qubits. Nonetheless, quadratic feature prediction with Clifford measurements is by no means useless. For instance, it can help provide statistical *a posteriori* guarantees on the quality of linear feature prediction — for example, by estimating sample variances to construct confidence intervals.

4.6 Numerical experiments

One of the key features of prediction with classical shadows is scalability. The data acquisition phase is designed to be tractable for state of the art platforms (Pauli measurements) and future quantum computers (Clifford measurements), respectively. The resulting classical shadow can be stored efficiently in classical memory. For may important features – such as local observables or global features with efficient stabilizer decompositions – scalability moreover extends to the computational cost associated with median of means prediction.

These design features allowed us to conduct numerical experiments for a wide range of problems and system sizes (up to 160 qubits). The computational bottleneck is not feature prediction with classical shadows, but generating synthetic data, i.e. classically generating target states and simulating quantum measurements. Needless to say, this classical bottle-neck does not occur in actual experiments. We then use this synthetic data to learn a classical representation of ρ and use this representation to predict various interesting properties.

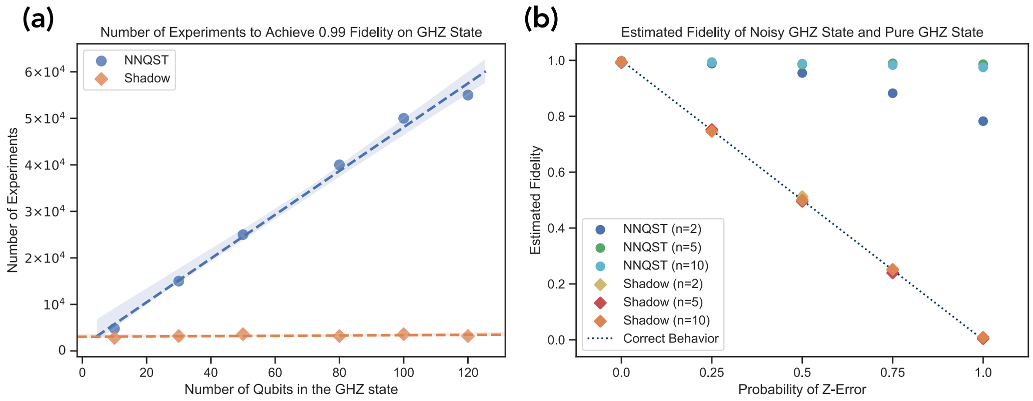


Figure 4.2: *Predicting quantum fidelities using classical shadows (Clifford measurements) and NNQST.*
(a) *(Left)***:** Number of measurements required to identify an n-qubit GHZ state with 0.99 fidelity. The shaded regions are the standard deviation of the needed number of experiments over ten independent runs.
(b) *(Right)***:** Estimated fidelity between a perfect GHZ target state and a noisy preparation, where Z-errors can occur with probability $p \in [0, 1]$, under 6×10^4 experiments. The dotted line represents the true fidelity as a function of p.
NNQST can only estimate an upper bound on quantum fidelity efficiently, so we consider this upper bound for NNQST and use quantum fidelity for the classical shadow.

Machine learning based approaches (Carrasquilla, Torlai, et al., 2019; Torlai, Mazzola, et al., 2018) are among the most promising alternative methods that have applications in this regime, where the Hilbert space dimension is roughly comparable to the total number of silicon atoms on earth ($2^{160} \simeq 10^{48}$). For example, a recent version of *neural network quantum state tomography* (NNQST) is a generative model that is based on a deep neural network trained on independent quantum measurement outcomes (local SIC/tetrahedral POVMs (Renes et al., 2004)). In this section, we consider the task of learning a classical representation of an unknown quantum state, and using the representation to predict various properties, addressing the relative merit of classical shadows and alternative methods.

Predicting quantum fidelities

Here we focus on classical shadows based on random Clifford measurements which are designed to predict observables with bounded Hilbert-Schmidt norm. When the observables have efficient representations — such as efficient stabilizer decompositions — the computational cost for performing median of means prediction can also be efficient.[4] An important example is the quantum fidelity with a target state. In (Carrasquilla, Torlai, et al., 2019), the viability of NNQST is demonstrated by considering GHZ states with a varying number of qubits n. Numerical experiments highlight that the number of measurement repetitions (size of the training data) to learn a neural network model of the GHZ state that achieves target fidelity of 0.99 scales linearly in n. We have also implemented NNQST for GHZ states and compared it to median of means prediction with classical shadows. The left-hand side of Figure 4.2 confirms the linear scaling of NNQST and the assertion of Theorem 7: classical shadows of *constant* size suffice to accurately estimate GHZ target fidelities, regardless of the actual system size. In addition, we have also tested the ability of both approaches to detect potential state preparation errors. More precisely, we consider a scenario where the GHZ-source introduces a phase error with probability $p \in [0,1]$:

$$\rho_p = (1-p)|\psi^+_{\mathrm{GHZ}}(n)\rangle\langle\psi^+_{\mathrm{GHZ}}(n)| + p|\psi^-_{\mathrm{GHZ}}(n)\rangle\langle\psi^-_{\mathrm{GHZ}}(n)|, \tag{4.36}$$

$$|\psi^{\pm}_{\mathrm{GHZ}}(n)\rangle = \tfrac{1}{\sqrt{2}}\left(|0\rangle^{\otimes n} \pm |1\rangle^{\otimes n}\rangle\right). \tag{4.37}$$

We learn a classical representation of the GHZ-source and subsequently predict the fidelity with the pure GHZ state. The right hand side of Figure 4.2 highlights that the classical shadow prediction accurately tracks the decrease in target fidelity as the error parameter p increases. NNQST, in contrast, seems to consistently overestimate this target fidelity. In the extreme case ($p = 1$), the true underlying state is completely orthogonal to the target state, but NNQST nonetheless reports fidelities close to one. This shortcoming arises because the POVM-based machine learning approach can only efficiently estimate an upper bound on the true quantum fidelity efficiently. To estimate the actual fidelity, an exceedingly large number of measurements is needed.

Predicting two-point correlation & subsystem entanglement entropy

Classical shadows based on random Clifford measurements excel at predicting quantum fidelities. However, random Clifford measurements can be challenging to implement in practice, because many entangling gates are needed to implement general Clifford circuits. Next we consider classical shadows based on random local Pauli measurements, which are easier to perform experimentally. The subsystem properties can be predicted efficiently by constructing the reduced density matrix from

[4] The runtime of Algorithm 1 is dominated by the cost of computing quadratic functions $\langle\hat{b}|UOU^\dagger|\hat{b}\rangle$ in 2^n dimensions. If $O = |\psi\rangle\langle\psi|$ is a stabilizer state, the Gottesman-Knill theorem allows for evaluation in $O(n^2)$-time.

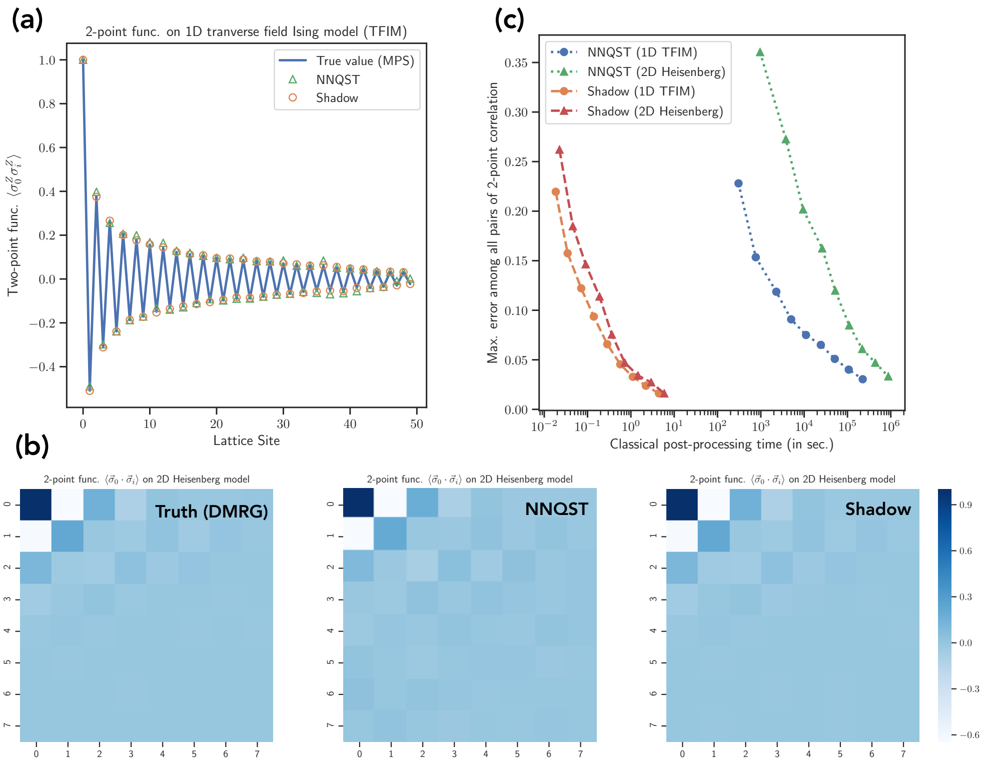


Figure 4.3: *Predicting two-point correlation functions using classical shadows (Pauli measurements) and NNQST.*
(a) *(Top Left)***:** Predictions of two-point functions $\langle\sigma_0^Z\sigma_i^Z\rangle$ for ground states of the one-dimensional critical anti-ferromagnetic transverse field Ising model with 50 lattice sites. These are based on $2^9 \times 1000$ random Pauli measurements.
(b) *(Bottom)***:** Predictions of two-point functions $\langle\sigma_0 \cdot \sigma_i\rangle$ for the ground state of the two-dimensional anti-ferromagnetic Heisenberg model with 8×8 lattice sites. The predictions are based on $2^9 \times 1000$ random Pauli measurements.
(c) *(Top Right)***:** The classical processing time (CPU time in seconds) and the prediction error (the largest among all pairs of two-point correlations) over different number of measurements: $\{2^1, \ldots, 2^9\} \times 1000$. The quantum measurement scheme in classical shadows (Pauli) is the same as the POVM-based neural network tomography (NNQST) in (Carrasquilla, Torlai, et al., 2019). The only difference is the classical post-processing. As the number of measurements increases, the processing time increases, while the prediction error decreases.

the classical shadow. Therefore, the computational complexity scales exponentially only in the subsystem size, rather than the size of the entire system. Our numerical experiments confirm that classical shadows obtained using random Pauli measurements excel at predicting few-body properties of a quantum state, such as two-point correlation functions and subsystem entanglement entropy.

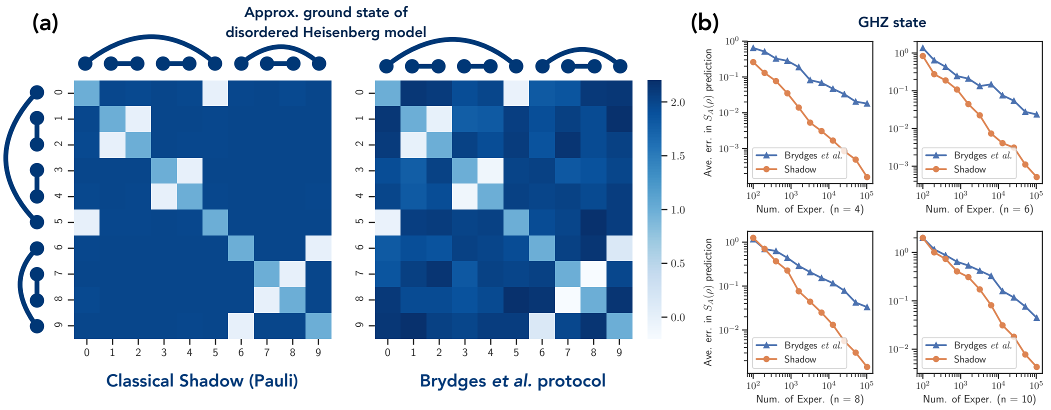


Figure 4.4: *Predicting entanglement Rényi entropies using classical shadows (Pauli measurements) and the Brydges et al. protocol.*
(a) *(Left)***:** Prediction of second-order Rényi entanglement entropy for all subsystems of size at most two in the approximate ground state of a disordered Heisenberg spin chain with 10 sites and open boundary conditions. The classical shadow is constructed from 2500 quantum measurements. The predicted values using the classical shadow visually match the true values with a maximum prediction error of 0.052. The Brydges *et al.* protocol (Brydges et al., 2019) results in a maximum prediction error of 0.24.
(b) *(Right)***:** Comparison of classical shadows and the Brydges *et al.* protocol (Brydges et al., 2019) for estimating second-order Rényi entanglement entropy in GHZ states. We consider the entanglement entropy of the left-half subsystem with size $n/2$.

Two-point correlation functions. NNQST has been shown to predict two-point correlation functions effectively (Carrasquilla, Torlai, et al., 2019). Here, we compare classical shadows with NNQST for two physically motivated test cases: ground states of the anti-ferromagnetic transverse field Ising model in one dimension (TFIM) and the anti-ferromagnetic Heisenberg model in two dimensions. The Hamiltonian for TFIM is $H = J\sum_i \sigma_i^Z\sigma_{i+1}^Z + h\sum_i \sigma_i^X$, where $J > 0$, and we consider a chain of 50 lattice sites. The critical point occurs at $h = J$ and exhibits power-law decay of correlations rather than exponential decay. The Hamiltonian for the 2D Heisenberg model is $H = J\sum_{\langle i,j\rangle} \sigma_i \cdot \sigma_j$, where $J > 0$, and we consider an 8×8 triangular lattice. We follow the approach in (Carrasquilla, Torlai, et al., 2019), where the ground state is approximated by a tensor network found using the density matrix renormalization group (DMRG). Random Pauli measurements on the ground state may then be simulated using this tensor network. The two methods are compared in Figure 4.3. On the top left (a) and bottom (b), we can see that both the classical shadow (with Pauli measurements) and NNQST perform well at predicting two-point correlations. However, NNQST has a larger error for the 2D Heisenberg model; note that for larger separations (the lower right corner of the surface plot), NNQST produces some fictitious oscillations that are not visible in the results from DMRG and classical shadows. The two approaches use the same quantum measurement data; the only difference is the classical post-processing. On the top right side (c) of Figure 4.3, we compare the cost of this classical post-processing, finding roughly a 10^4 times speedup in classical processing time using the classical shadow instead of NNQST.

Subsystem entanglement entropies. An important nonlinear property that can be predicted with classical shadows is subsystem entanglement entropy. The required number of measurements scales exponentially in subsystem size, but is independent of the total number of qubits. Moreover, these measurements can be used to predict many subsystem entanglement entropies at once. This problem has also been studied extensively in (Brydges et al., 2019), where a specialized approach (which we refer to here as the "Brydges *et al.* protocol") was designed to efficiently estimate second-order Rényi entanglement entropies using random local measurements. In (Brydges et al., 2019), a random unitary rotation is reused several times. Predictions using classical shadows could also be slightly modified to adapt to this scenario. Results from our numerical experiments are shown in Figure 4.4. On the left (a), we predict the entanglement entropy for all subsystems of size ≤ 2 from only 2500 measurements of the approximate ground state of the disordered Heisenberg model in one dimension. This is a prototypical model for studying many-body localization (Nandkishore and Huse, 2015). The ground state is approximated by a set of singlet states $\{\frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)\}$ found using the strong-disorder renormalization group (Ma, Dasgupta, and C.-k. Hu, 1979; Dasgupta and Ma, 1980). Both, the classical shadow protocol and the Brydges et al. method use random single-qubit rotations and basis measurements to find a classical representation of the quantum state; the only difference between the methods is in the classical post-processing. For these small subsystems, we find that the prediction error of the classical shadow is smaller than the error of the Brydges et al. protocol. On the right hand side of Figure 4.4 (b), we consider predicting the entanglement entropy in a GHZ state for system sizes ranging from $n = 4$ to $n = 10$ qubits. We focus on the entanglement entropy of the left-half subsystem with system size $n/2$. Note that this entanglement entropy is equal to one bit for any system size n. To achieve an error of 0.05, classical

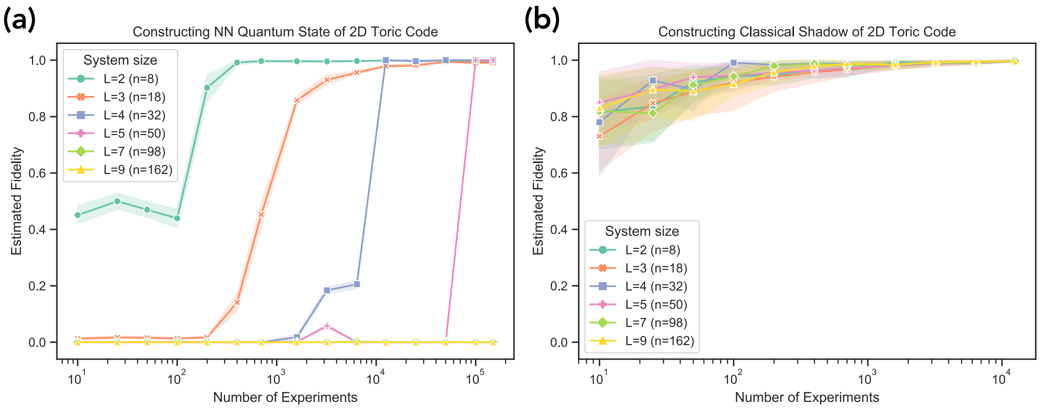


Figure 4.5: *Comparison between classical shadow and neural network tomography (NNQST); toric code.*
(a) *(Left)***:** Number of measurements required for neural network tomography to identify a particular toric-code ground state. We use classical fidelity for NNQST, which is an upper bound for quantum fidelity.
(b)*(Right)***:** Performance of classical shadows for the same problem. We use quantum fidelity for classical shadows. The shaded regions are the standard deviation of the estimated fidelity over ten runs.

shadows require several times fewer measurements and the discrepancy increases as we require smaller error.

Direct fidelity estimation for the toric code ground state

In the main text, we have considered direct fidelity estimation for GHZ states and compared it with neural network quantum state tomography (NNQST). While highly instructive from a theoretical perspective, GHZ states comprised of 100 qubits are very fragile and challenging to implement in practice. To conduct experiments for more physical target states, we consider *Toric code ground states* (Dennis et al., 2002a). Not only are they the most prominent example of a topological quantum error correcting code and thus highly relevant for quantum computing devices. They also correspond to ground states of a Hamiltonian: $H = -\sum_v A_v - \sum_p B_p$, where A_v and B_p denote vertex- and plaquette operators[5]. The ground space of H is four-fold degenerate and we select the superposition of all closed-loop configurations ($|\psi\rangle \propto \sum_{S:\text{ closed loop}} |S\rangle$) as a test state for both classical shadows and NNQST: how many measurement repetitions are required to accurately identify this toric code

[5] A_v is the product of four Pauli-X operators around a vertex v, while B_p is the product of four Pauli-Z operators around the plaquette p.

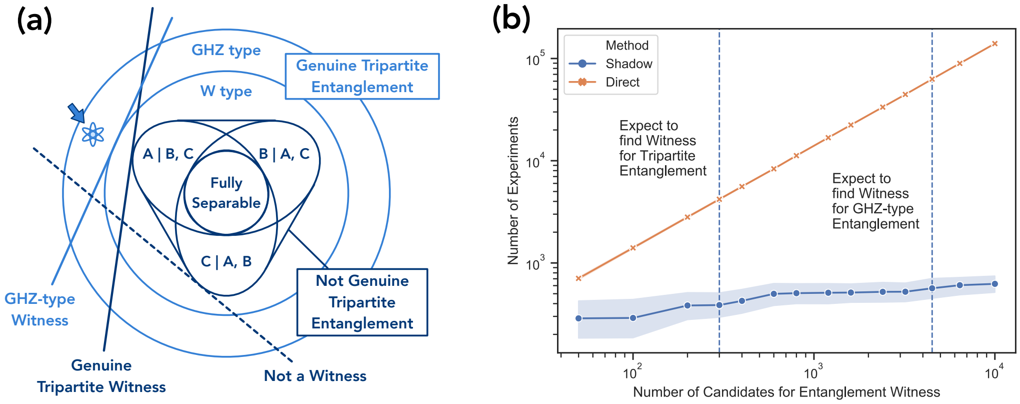


Figure 4.6: *Detection of GHZ-type entanglement for 3-qubit states.*
(a) *(Left)***:** Schematic illustration of 3-partite entanglement. Entanglement witnesses are linear functions that separate part of one entanglement class from all other classes.
(b) *(Right)***:** Number of entanglement witnesses vs. number of experiments required to accurately estimate all of them. The dashed lines represent the expected number of (random) entanglement witnesses required to detect genuine three-partite entanglement and GHZ-type entanglement in a randomly rotated GHZ state. The shaded region is the standard deviation of the required number of experiments over ten independent repetitions of the entire setup.

ground state with high fidelity? The results are shown in Supplementary Figure 4.5. Neural network tomography based on a deep generative model seems to require a number of samples that scales unfavorably in the system size n (left). In contrast, fidelity estimation with classical shadows is completely independent of the system size. The difficulty of NNQST in learning 2D toric code may be related to some observed failures of deep learning (Shalev-Shwartz, O. Shamir, and Shammah, 2017) for learning patterns with combinatorial structures. In Supplementary Section 4.8, we provide further evidence for potential difficulties when using machine learning approaches to reconstruct some simple quantum states due to a well-known computational hardness conjecture.

Witnesses for tripartite entanglement

Entanglement is at the heart of virtually all quantum communication and cryptography protocols and an important resource for quantum technologies in general. This renders the task of detecting entanglement important both in theory and practice (Friis et al., 2019; Gühne and Tóth, 2009). While bipartite entanglement is comparatively well-understood, multi-partite entanglement has a much more involved structure. Already for $n = 3$ qubits, there is a variety of inequivalent entanglement classes. These include fully-separable, as well as bi-separable states, W-type states and finally GHZ-type states. The relations between these classes are summarized in Supplementary Figure 4.6 and we refer to (Acín et al., 2001) for a complete characterization. Despite this increased complexity, entanglement witnesses remain a simple and useful tool for testing which class a certain state ρ belongs to. However, any given entanglement witness only provides a one-sided test – see Supplementary Figure 4.6 (left) for an illustration – and it is often necessary to compute multiple witnesses for a definitive answer.

Classical shadows based on random Clifford measurements can considerably speed up this search: according to Theorem 7 a classical shadow of moderate size allows for checking an entire list of fixed entanglement witnesses simultaneously. Supplementary Figure 4.6 (right) underscores the economic advantage of such an approach over measuring the individual witnesses directly. Directly measuring M different entanglement witnesses requires a number of quantum measurements that scales (at least) linearly in M. In contrast, classical shadows get by with $\log(M)$-many measurements only.

More concretely, suppose that the state to be tested is a local, random unitary transformation of the GHZ state. Then, this state is genuinely tripartitely entangled and moreover belongs to the GHZ class. The dashed vertical lines in Supplementary Figure 4.6 (right) denote the expected number of (randomly selected) witnesses required to detect genuine tripartite entanglement (first) and GHZ-type entanglement (later). From the experiment, we can see that classical shadows achieve these thresholds with an exponentially smaller number of samples than the naive direct method. Finally, classical shadows are based on random Clifford measurements and do not depend on the structure of the concrete witness in question. In contrast, direct estimation crucially depends on the concrete witness in question and may be considerably more difficult to implement.

Application to quantum simulation of the lattice Schwinger model

Simulations of quantum field theory using quantum computers may someday advance our understanding of fundamental particle physics. Although high impact discoveries may still be a ways off, notable results have already been achieved in studies of one-dimensional lattice gauge theories using quantum platforms.

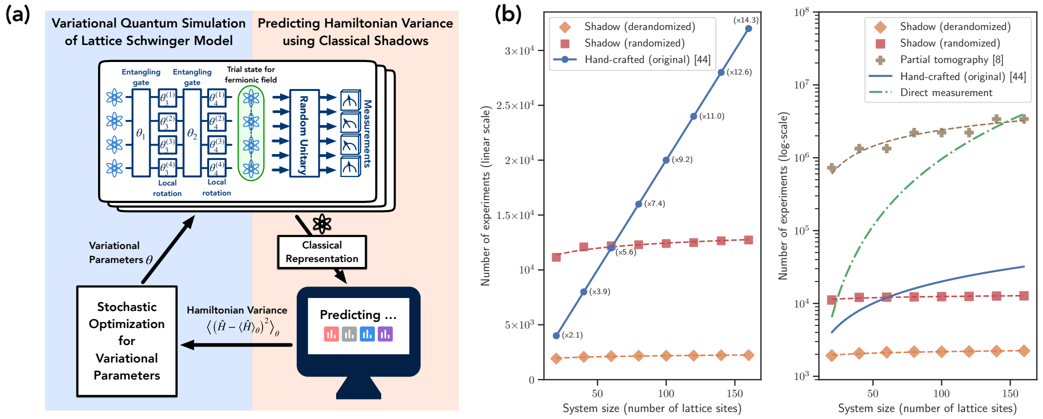


Figure 4.7: *Application of classical shadows (Pauli measurements) to variational quantum simulation of the lattice Schwinger model.*
(a) *(Left)***:** An illustration of variational quantum simulation and the role of classical shadows.
(b) *(Right)***:** The comparison between different approaches in the number of measurements needed to predict all 4-local Pauli observables in the expansion of $\langle(\hat{H} - \langle\hat{H}\rangle_\theta)^2\rangle_\theta$ with an error equivalent to measuring each Pauli observable at least 100 times. We include a linear-scale plot that compares classical shadows with the original hand-designed measurement scheme in (Kokail et al., 2019) and a log-scale plot that compares with other approaches. In the linear-scale plot, $(\times T)$ indicates that the original scheme uses T times the number of measurements compared to classical shadows (derandomized).

For example, in (Kokail et al., 2019) a 20-qubit trapped ion analog quantum simulator was used to prepare low-energy eigenstates of the lattice Schwinger model (one-dimensional quantum electrodynamics). The authors prepared a family of quantum states $\{|\psi(\theta)\rangle\}$, where θ is a variational parameter, and computed the variance of the energy $\langle(\hat{H} - \langle\hat{H}\rangle_\theta)^2\rangle_\theta$ for each value of θ. Here $\hat{H}$ is the Hamiltonian of the model, and $\langle\hat{O}\rangle_\theta = \langle\psi(\theta)|\hat{O}|\psi(\theta)\rangle$ is the expectation value of the operator $\hat{O}$ in the state $|\psi(\theta)\rangle$. Because energy eigenstates, and only energy eigenstates, have vanishing energy dispersion, adjusting θ to minimize the variance of energy prepares an energy eigenstate.

After solving the Gauss law constraint to eliminate the gauge fields, the Hamiltonian $\hat{H}$ of the Schwinger model is 2-local, though not geometrically local in one dimension. Hence the quantity $\langle(\hat{H} - \langle\hat{H}\rangle_\theta)^2\rangle_\theta$ is a sum of expectation values of 4-local observables, which can be measured efficiently using a classical shadow derived from random Pauli measurements. This is illustrated on the left side of

Figure 4.7 (a). On the right side of Figure 4.7 (b), we compare the performance of classical shadows to the measurement scheme for 4-local observables designed in (Kokail et al., 2019), and also to a recent method (Bonet-Monroig, Babbush, and T. E. O'Brien, 2019) for measuring local observables, as well as the standard approach that directly measures all observables independently.

The results show, for the methods we considered, the number of copies of the quantum state needed to measure the expectation value of all 4-local Pauli observables in $\langle (\hat{H} - \langle \hat{H} \rangle_\theta)^2 \rangle_\theta$ with an error equivalent to measuring each of these observables at least 100 times. In (Kokail et al., 2019), such a relatively small number of measurements per local observable already yielded results comparable to theoretical predictions based on exact diagonalization. We find that the performance of the classical shadow method is better than the method used in (Kokail et al., 2019) only for system size larger than 50 qubits and may actually be worse for small system sizes. However, classical shadows provide a good prediction for any set of local observables, while the method of (Kokail et al., 2019) was hand-crafted for the particular task of estimating the variance of the energy in the Schwinger model.

To make a more apt comparison, we constructed a deterministic version of classical shadows, using a fixed set of measurements rather than random Pauli measurements, specifically adapted for the purpose of estimating $\langle (\hat{H} - \langle \hat{H} \rangle_\theta)^2 \rangle_\theta$ in the lattice Schwinger model. This deterministic collection of Pauli measurements is obtained by a powerful technique called *derandomization* (Raghavan, 1988; J. Spencer, 1994). This procedure simulates the classical shadow scheme based on randomized measurements and makes use of the rigorous performance bound we developed. When a coin is tossed in the randomized scheme to decide which measurement to perform next, the next measurement in the derandomized version is chosen to have the best possible performance bound for the rest of the protocol. It turns out that this derandomization of the classical shadow method can be carried out very efficiently; full details will be presented in the next section. Not surprisingly, the derandomized version, also included in Figure 4.7, outperforms the randomized version by a considerable margin. We then find that the derandomized classical shadow method is significantly more efficient than the other methods we considered, including the hand-crafted method from (Kokail et al., 2019). Finally, we emphasize that the derandomization procedure is fully automated (see `https://github.com/momohuang/predicting-quantum-properties` for open-source code) and not problem-specific. It could be used for any pre-specified set of local observables.

4.7 Derandomizing randomized measurements

So far, we have seen that randomized measurements are very powerful in learning an approximate description of a quantum many-body system. A natural question is to ask if the randomness involved is necessary. If there is an approach to remove the randomness, could such an approach learn even more efficiently? From the last numerical experiments on speeding up the quantum simulation of the lattice Schwinger model, we have seen that performing derandomization gives a substantial boost in performance. In this section, we will present how derandomization works and why it can give much better results.

Statistical background

Let ρ be a fixed, but unknown, quantum state on n qubits. We want to accurately predict L expectation values

$$\omega_\ell(\rho) = \mathrm{tr}(O_{\mathbf{o}_\ell}\rho) \quad \text{for } 1 \leq \ell \leq L, \tag{4.38}$$

where each $O_{\mathbf{o}_\ell} = \sigma_{\mathbf{o}_\ell[1]} \otimes \cdots \otimes \sigma_{\mathbf{o}_\ell[n]}$ is a tensor product of single-qubit Pauli matrices, i.e. $\mathbf{o}_\ell = [\mathbf{o}_\ell[1], \ldots, \mathbf{o}_\ell[n]]$ with $\mathbf{o}_\ell[k] \in \{I, X, Y, Z\}$. To extract meaningful information, we perform M (single-shot) Pauli measurements on independent copies of ρ. There are 3^n possible measurement choices. Each of them is characterized by a full-weight Pauli string $\mathbf{p}_m \in \{X, Y, Z\}^n$ and produces a random string of n outcome signs $\mathbf{q}_m \in \{\pm 1\}^n$.

Not every Pauli measurement $\mathbf{p}_m$ $(1 \leq m \leq M)$ provides actionable advice about every target observable $\mathbf{o}_\ell$ $(1 \leq \ell \leq L)$. The two must be compatible in the sense that the latter corresponds to a marginal of the former, i.e. it is possible to obtain $\mathbf{o}_\ell$ from $\mathbf{p}_m$ by replacing some local non-identity Paulis with I. If this is the case, we write $\mathbf{o}_\ell \triangleright \mathbf{p}_m$ and say that measurement $\mathbf{p}_m$ "hits" target observable $\mathbf{o}_\ell$. For instance, $[X, I], [I, X], [X, X] \triangleright [X, X]$, but $[Z, I], [I, Z], [Z, Z] \not\triangleright [X, X]$. We can approximate each $\omega_\ell(\rho)$ by empirically averaging (appropriately marginalized) measurement outcomes that belong to Pauli measurements that hit $\mathbf{o}_\ell$:

$$\hat{\omega}_\ell = \frac{1}{h(\mathbf{o}_\ell; [\mathbf{p}_1, \ldots, \mathbf{p}_M])} \sum_{m\colon \mathbf{o}_\ell \triangleright \mathbf{p}_m} \prod_{j : \mathbf{o}_\ell[j] \neq I} \mathbf{q}_m[j], \tag{4.39}$$

where $h(\mathbf{o}_\ell; [\mathbf{p}_1, \ldots, \mathbf{p}_M]) = \sum_{m=1}^{M} \mathbf{1}\{\mathbf{o}_\ell \triangleright \mathbf{p}_m\} \in \{0, 1, \ldots, M\}$ counts how many Pauli measurements hit target observable $\mathbf{o}_\ell$.

It is easy to check that each $\hat{\omega}_\ell$ exactly reproduces $\omega_\ell(\rho)$ in expectation (provided that $h(\mathbf{o}_\ell;\mathbf{P}) \geq 1$). Moreover, the probability of a large deviation improves exponentially with the number of hits.

Lemma 7 (Confidence bound)**.** *Fix* $\varepsilon \in (0,1)$ *(accuracy) and* $1-\delta \in (0,1)$ *(confidence). Suppose that Pauli observables* $\mathbf{O} = [\mathbf{o}_1,\ldots,\mathbf{o}_L]$ *and Pauli measurements* $\mathbf{P} = [\mathbf{p}_1,\ldots,\mathbf{p}_M]$ *are such that*

$$\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P}) := \sum_{\ell=1}^{L} \exp\left(-\tfrac{\varepsilon^2}{2} h(\mathbf{o}_\ell;\mathbf{P})\right) \leq \frac{\delta}{2}. \tag{4.40}$$

Then, the associated empirical averages (4.39) *obey*

$$|\hat{\omega}_\ell - \omega_\ell(\rho)| \leq \varepsilon \quad \textit{for all } 1 \leq \ell \leq L \tag{4.41}$$

with probability (at least) $1-\delta$.

See Section 4.7 for a detailed derivation. We call the function defined in Eq. (4.40) the *confidence bound*. It is a statistically sound summary parameter that checks whether a set of Pauli measurements ($\mathbf{P}$) allows for confidently predicting a collection of Pauli observables ($\mathbf{O}$) up to accuracy ε each.

Randomized Pauli measurements

Intuitively speaking, a small confidence bound (4.40) implies a good Pauli estimation protocol. But how should we choose our M Pauli measurements ($\mathbf{P}$) in order to achieve $\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P}) \leq \delta/2$? The randomized measurement toolbox (Ohliger, Nesme, and Eisert, 2013a; A. Elben et al., 2019a; Huang, Richard Kueng, and Preskill, 2020; Paini and Kalev, 2019; Andreas Elben, Richard Kueng, et al., 2020a) provides a perhaps surprising answer to this question. Let $\mathrm{w}(\mathbf{o}_\ell)$ denote the *weight* of Pauli observable $\mathbf{o}_\ell$, i.e. the number of qubits on which the observable acts nontrivially: $\mathrm{w}(\mathbf{o}_\ell) = \sum_{k=1}^{n} \mathbf{1}\{\mathbf{o}_\ell[k] \neq I\}$. These weights capture the probability of hitting $\mathbf{o}_\ell$ with a completely random measurement string: $\Pr_{\mathbf{p}}[\mathbf{o}_\ell \triangleright \mathbf{p}] = 1/3^{\mathrm{w}(\mathbf{o}_\ell)}$. In turn, a total of M randomly selected Pauli measurements will on average achieve $\mathbb{E}_{\mathbf{P}}[h(\mathbf{o}_\ell;\mathbf{P})] = M/3^{\mathrm{w}(\mathbf{o}_\ell)}$ hits, regardless of the actual Pauli observable $\mathbf{o}_\ell$ in question. This insight allows us to compute expectation values of the confidence bound (4.40):

$$\mathbb{E}_{\mathbf{P}}\left[\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P})\right] = \sum_{\ell=1}^{L} \left(1 - \nu/3^{\mathrm{w}(\mathbf{o}_\ell)}\right)^M, \tag{4.42}$$

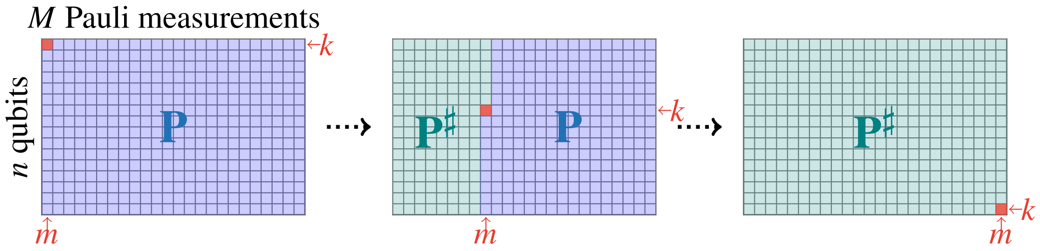


Figure 4.8: *Derandomization algorithm (Algorithm 2):* We envision M randomized n-qubit measurements as a 2-dimensional array comprised of $n \times M$ Pauli labels. Blue squares are placeholders for random Pauli labels, while green squares denote deterministic assignments (either X, Y or Z). Starting with a completely unspecified array (*left*), the algorithm iteratively checks how a concrete Pauli assignment (red square) affects the confidence bound, Eq. (4.40), averaged over all remaining assignments. A simple update rule, Eq. (4.45), replaces the initially random label with a deterministic assignment that keeps the remaining confidence bound expectation as small as possible (*centre*). Once the entire grid is traversed, no randomness is left (*right*) and the algorithm outputs M deterministic n-qubit Pauli measurements.

where $\nu = 1-\exp(-\varepsilon^2/2) \in (0,1)$. Each of the L terms is exponentially suppressed in $\varepsilon^2 M/3^{\mathrm{w}(\mathbf{o}_\ell)}$. Concrete realizations of a randomized measurement protocol are extremely unlikely to deviate substantially from this expected behavior, see e.g. (Evans, Harper, and Steven T Flammia, 2019). Combined with Lemma 7, this observation implies a powerful error bound.

Theorem 13 (Theorem 3 in Ref. (Evans, Harper, and Steven T Flammia, 2019))**.** *Empirical averages* (4.39) *obtained from M randomized Pauli measurements allow for ε-accurately predicting L Pauli expectation values* $\mathrm{tr}(O_{\mathbf{o}_1}\rho), \ldots, \mathrm{tr}(O_{\mathbf{o}_L}\rho)$ *up to additive error ε given that* $M \propto \log(L) \max_\ell 3^{\mathrm{w}(\mathbf{o}_\ell)}/\varepsilon^2$.

In particular, order $\log(L)$ randomized Pauli measurements suffice for estimating any collection of L low-weight Pauli observables. It is instructive to compare this result to other powerful statements about randomized measurements, most notably the "classical shadow" paradigm (Huang, Richard Kueng, and Preskill, 2020; Paini and Kalev, 2019). For Pauli observables and Pauli measurements, the two approaches are closely related. The estimators (4.39) are actually simplified variants of the classical shadow protocol (in particular, they don't require median of means prediction) and the requirements on M are also comparable. This is no coincidence; information-theoretic lower bounds from (Huang, Richard Kueng, and Preskill, 2020) assert that there are scenarios where the scaling $M \propto \log(L) \max_\ell 3^{\mathrm{w}(\mathbf{o}_\ell)}/\varepsilon^2$ is asymptotically

Algorithm 2 **(Derandomization)**

Input: measurement budget M, accuracy ε, and L n-qubit Pauli $\mathbf{O} = [\mathbf{o}_1, \ldots, \mathbf{o}_L]$
Output: M Pauli measurements $\mathbf{P}^\sharp \in \{X, Y, Z\}^{n \times M}$

```
function DERANDOMIZATION($\mathbf{O}, M, \varepsilon$)
    initialize $\mathbf{P}^\sharp = [\ [\quad]\ ]$ (empty $n \times M$ array)
    for $m = 1$ to $M$ do                                        ▷ loop of over measurements
        for $k = 1$ to $n$ do                                    ▷ loop over qubits
            for $W = X, Y, Z$ do compute
                $f(W) \;=\; \mathbb{E}_{\mathbf{P}}\left[\mathrm{Conf}_\varepsilon(\mathbf{O}; \mathbf{P}) | \mathbf{P}^\sharp, \mathbf{P}[k, m] = W\right]$
                (see Eq. (4.43) for a precise formula)
            $\mathbf{P}^\sharp[k, m] \leftarrow \mathrm{argmin}_{W \in \{X,Y,Z\}} f(W)$
    output $\mathbf{P}^\sharp \in \{X, Y, Z\}^{n \times M}$
```

optimal and cannot be avoided.

Nevertheless, this does not mean that randomized measurements are *always* a good idea. High-weight observables do pose an immediate challenge, because it is extremely unlikely to hit them by chance alone.

Derandomized Pauli measurements

The main result of derandomization is a procedure for identifying "good" Pauli measurements that allow for accurately predicting many (fixed) Pauli expectation values. This procedure is designed to interpolate between two extremes: (i) *completely randomized measurements* (good for predicting many local observables) and (ii) *completely deterministic measurements* that directly measure observables sequentially (good for predicting few global observables).

Note that we can efficiently compute concrete confidence bounds (4.40), as well as expected confidence bounds averaged over all possible Pauli measurements (4.42). Combined, these two formulas also allow us to efficiently compute expected confidence bounds for a list of measurements that is partially deterministic and partially randomized. Suppose that $\mathbf{P}^\sharp$ subsumes deterministic assignments for the first $(m-1)$ Pauli measurements, as well as concrete choices for the first $(k-1)$ Pauli labels of the m-th measurement, see Fig. 4.8 (center). There are three possible choices for the next Pauli assignment: $\mathbf{P}^\sharp[k, m] = W$ with $W = X, Y, Z$. For each choice, we can explicitly compute the resulting conditional expectation value:

$$\mathbb{E}_{\mathbf{P}}\left[\textsc{Conf}_\varepsilon(\mathbf{O}; \mathbf{P}) | \mathbf{P}^\sharp, \mathbf{P}[k, m] = W\right] \tag{4.43}$$

$$= \sum_{\ell=1}^{L} \exp\left(-\frac{\varepsilon^2}{2} \sum_{m'=1}^{m-1} \prod_{k'=1}^{n} \mathbf{1}\left\{\mathbf{o}_\ell[k'] \triangleright \mathbf{P}^\sharp[k', m']\right\}\right)$$

$$\times\left(1-\nu\frac{\mathbf{1}\left\{\mathbf{o}_\ell[k]\rhd W\right\}}{3^{\mathrm{w}_{\neg k}(\mathbf{o}_\ell)}}\prod_{k'=1}^{k-1}\mathbf{1}\left\{\mathbf{o}_\ell[k']\rhd\mathbf{P}^\sharp[k',m]\right\}\right)$$
$$\times\left(1-\nu 3^{-\mathrm{w}(\mathbf{o}_\ell)}\right)^{M-m},$$

where $\nu = 1-\exp(-\varepsilon^2/2)$, $\mathrm{w}_{\neg k}(\mathbf{o}_\ell) = \mathrm{w}([\mathbf{o}_\ell[k+1],\ldots,\mathbf{o}_\ell[n]])$ and $\mathbf{o}_\ell[k'] \rhd \mathbf{P}^\sharp[k',m]$ if $\mathbf{o}_\ell[k'] = I$ or $\mathbf{o}_\ell[k'] = \mathbf{P}^\sharp[k',m]$. This formula allows us to build deterministic measurements one Pauli-label at a time.

We start by envisioning a collection of M completely random n-qubit Pauli measurements. That is, each Pauli label is random and Eq. (4.42) captures the expected confidence bound averaged over *all* $3^n \times 3^M = 3^{n+M}$ assignments. There are three possible choices for the first label in the first Pauli measurement: $\mathbf{P}[1,1] = X$, $\mathbf{P}[1,1] = Y$ and $\mathbf{P}[1,1] = Z$. At least one concrete choice does not further increase the confidence bound averaged over all remaining Pauli signs:

$$\begin{aligned}&\min_{W\in\{X,Y,Z\}}\mathbb{E}_{\mathbf{P}}\left[\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P})|\mathbf{P}[1,1]=W\right]\\ \leq&\tfrac{1}{3}\sum_{W\in\{X,Y,Z\}}\mathbb{E}_{\mathbf{P}}\left[\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P})|\mathbf{P}[1,1]=W\right]\\ =&\mathbb{E}_{\mathbf{P}}\left[\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P})\right].\end{aligned}\tag{4.44}$$

Crucially, Eq. (4.43) allows us to efficiently identify a minimizing assignment:

$$\mathbf{P}^\sharp[1,1] = \operatorname*{argmin}_{W\in\{X,Y,Z\}}\mathbb{E}_{\mathbf{P}}\left[\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P})|\mathbf{P}[1,1]=W\right]\tag{4.45}$$

Doing so, replaces an initially random single-qubit measurement setting by a concrete Pauli label that minimizes the conditional expectation value over all remaining (random) assignments. This procedure is known as derandomization (Motwani and Raghavan, 1995; Alon and J. H. Spencer, 2008; V. V. Vazirani, 2001) and can be iterated. Fig. 4.8 provides visual guidance, while pseudo-code can be found in Algorithm 2. There are a total of $n \times M$ iterations. Step (k,m) is contingent on comparing three conditional expectation values $\mathbb{E}_{\mathbf{P}}\left[\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P})|\mathbf{P}^\sharp,\mathbf{P}[k,m]=W\right]$ and assigning the Pauli label that achieves the smallest score. These update rules are constructed to ensure that (appropriate modifications of) Eq. (4.44) remain valid throughout the procedure. Combining all of them implies the following rigorous statement about the resulting Pauli measurements $\mathbf{P}^\sharp$.

Theorem 14 (Derandomization)**.** *Algorithm 2 is guaranteed to output Pauli measurements* $\mathbf{P}^\sharp$ *with below average confidence bound:*

$$\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P}^\sharp) \leq \mathbb{E}_{\mathbf{P}}\left[\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P})\right].\tag{4.46}$$

We see that derandomization produces deterministic Pauli measurements that perform at least as favorably as (averages of) randomized measurement protocols. But the actual difference between randomized and derandomized Pauli measurements can be much more pronounced. In the examples we considered, derandomization reduces the measurement budget M by at least an order of magnitude, compared to randomized measurements. Furthermore, because Algorithm 2 implements a greedy update procedure, we have no assurance that our derandomized measurement procedure is globally optimal or even close to optimal. Using dynamic programming, the derandomization algorithm runs in time $O(nML)$.

Illustrative derandomization examples

The exact workings of Algorithm 2 depend on the structure of the set of Pauli observables. In this section, we provide several examples to illustrate the mechanism of the derandomization procedure.

Many local Pauli observables. Many near-term applications of quantum devices rely on repeatedly estimating a large number of low-weight Pauli observables. For example, low-energy eigenstates of a many-body Hamiltonian may be prepared and studied using a variational method, in which the Hamiltonian, a sum of local terms, is measured many times. Using randomized measurements, we can predict many low-weight observables simultaneously at comparatively little cost. It is known that a logarithmic number of randomized Pauli measurements allows for accurately predicting a polynomial number of low-weight observables (Huang, Richard Kueng, and Preskill, 2020).

This desirable feature provably extends to derandomized measurements. From Theorem 14 and Eq. (4.42), we infer that the measurement budget

$$M = 4\log(2L/\delta)\max_{\ell} 3^{\mathrm{w}(\mathbf{o}_\ell)}/\varepsilon^2 \tag{4.47}$$

suffices to ensure that Algorithm 2 outputs Pauli measurements $\mathbf{P}^\sharp$ that obey $\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P}) \leq \delta/2$. With Lemma 7, we may convert this into an error bound: empirical averages (4.39) formed from appropriate measurement outcomes are guaranteed to obey $\left|\hat{\omega}_\ell - \mathrm{tr}(O_{\mathbf{o}_\ell}\rho)\right| \leq \varepsilon$ for all $1 \leq \ell \leq L$ with high probability (at least $1-\delta$). This error bound is roughly on par with the best rigorous result about predicting local Pauli observables from randomized Pauli measurements (Evans, Harper, and Steven T Flammia, 2019). But this argument implicitly assumes that $\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P}^\sharp)$ (which we can compute) is comparable to $\mathbb{E}_{\mathbf{P}}\left[\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P})\right]$ (which is characterized by Eq. (4.42)). This assumption is extremely pessimistic, because often $\textsc{Conf}_{\varepsilon}(\mathbf{O};\mathbf{P}^{\sharp}) \ll \mathbb{E}_{\mathbf{P}}\left[\textsc{Conf}_{\varepsilon}(\mathbf{O};\mathbf{P})\right]$. If this is the case, derandomized Pauli measurements perform substantially better.

Few global Pauli observables. We have seen that derandomized measurements never perform worse than randomized measurements. But they can perform much better. This discrepancy is best illustrated with a simple example: design Pauli measurements to predict both a complete Y-string ($\mathbf{o}_1 = [Y, \ldots, Y]$) and a complete Z-string ($\mathbf{o}_2 = [Z, \ldots, Z]$). Here, randomized measurements are a terrible idea, because it is exponentially unlikely to hit either string by chance alone.

Contrast this with derandomization. For the very first assignment ($k = 1$,$m = 1$), Algorithm 2 starts by computing three conditional expectations. Comparing them reveals $f(Y) = f(Z) < f(X)$ and the algorithm determines that assigning X is likely a bad idea. The two remaining choices should be equivalent and the algorithm assigns, say, $\mathbf{P}^{\sharp}[1,1] = Y$. This initial choice does affect the expected confidence bound associated with the second Pauli label ($k = 2$,$m = 1$): $f(Y) < f(X) = f(Z)$. Taking into account the already assigned first Pauli label, both X and Z become equally unfavorable and the algorithm sticks to assigning $\mathbf{P}^{\sharp}[2,1] = Y$. This situation now repeats itself until the first Pauli measurement is completely assigned: $\mathbf{p}_1^{\sharp} = [Y, \ldots, Y] = \mathbf{o}_1$. The algorithm has successfully kept track of an entire global Pauli string.

It is now time to assign the first Pauli label of the second Pauli measurement ($k = 1$, $m = 2$). While X is still a bad idea, taking into account that we have already measured $\mathbf{o}_1$ once also breaks the symmetry between Y and Z assignments: $f(Z) < f(Y) < f(X)$. So the algorithm chooses $\mathbf{P}^{\sharp}[1,2] = Z$ and subsequently sticks to assigning Z for all qubits: $\mathbf{p}_2^{\sharp} = [Z, \ldots, Z] = \mathbf{o}_2$. Having measured both $\mathbf{o}_1$ and $\mathbf{o}_2$ an equal number of times restores the initial symmetry and the algorithm basically resets. This process resets until all M Pauli measurements are assigned and Algorithm 2 outputs $\mathbf{P}^{\sharp} = [\mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_1, \mathbf{o}_2]$. In words: measure both global observables equally often. Although statistically optimal, this measurement protocol is neither surprising nor particularly interesting. What is encouraging, though, is that Algorithm 2 has (re-)discovered it all by itself.

Very many global Pauli observables (non-example): The derandomization algorithm is not without flaws. The greedy update rule in line 8 of Algorithm 2 can be misguided to produce non-optimal results. This happens, for instance, for a very large collection of global Pauli observables that appears to have favorable structure but actually doesn't. For instance, set $\mathbf{o}_1 = [X, \ldots, X]$ and $\mathbf{o}_\ell = [Z; \tilde{\mathbf{o}}_\ell]$, where $\tilde{\mathbf{o}}_\ell \in \{X, Y, Z\}^{n-1}$ ranges through all 3^{n-1} possible Pauli strings of size $(n-1)$. There are $L = 3^{n-1} + 1$ target observables, all of which are global and therefore incompatible. However, 3^{n-1} of them start with a Pauli-Z label. This imbalance leads the algorithm to believe that assigning $\mathbf{P}^\sharp[1, m] = Z$ for all $1 \leq m \leq M$ is always a good idea (provided that M is not much larger than 3^{n-1}). By doing so, it completely ignores the first target observable which starts with an X-label. But at the same time, it cannot capitalize on this particular decision, because observables $\mathbf{o}_2$ to $\mathbf{o}_L$ are actually incompatible. This results in an imbalanced output $\mathbf{P}^\sharp$ that treats observables $\mathbf{o}_2$ to $\mathbf{o}_L$ roughly equally, but completely forgets about $\mathbf{o}_1$. Needless to say, the resulting confidence bound will not be minimal either. We emphasize that this highly stylized non-example is not motivated by actual applications. Instead it is intended to illustrate how greedy update procedures can get stuck in local minima.

Additional details and proofs

Proof of Lemma 7 Let us briefly recapitulate the general setting. A n-qubit Pauli measurement $\mathbf{p} \in \{X, Y, Z\}^n$ produces a random string of n signs $\hat{\mathbf{q}} \in \{\pm 1\}^n$. Information about the underlying n-qubit state ρ is encoded in the distribution of outcome strings

$$\Pr\left[\hat{\mathbf{q}} = \mathbf{q} | \mathbf{p}, \rho\right] = \mathrm{tr}\left(\bigotimes_{j=1}^{m} \tfrac{1}{2}\left(\sigma_I + \mathbf{q}[j]\sigma_{\mathbf{p}[j]}\right)\rho\right) \quad \text{for all } \mathbf{q} \in \{\pm 1\}^n. \tag{4.48}$$

Now, suppose that $\mathbf{o} \in \{I, X, Y, Z\}^n$ is another Pauli string that is hit by $\mathbf{p}$ ($\mathbf{o} \triangleright \mathbf{p}$). Then, we can appropriately marginalize n-qubit outcome strings $\mathbf{q} \in \{\pm 1\}^n$ to reproduce $\omega(\rho) = \mathrm{tr}\left(O_{\mathbf{o}}\rho\right)$ in expectation:

$$\mathbb{E} \prod_{j:\mathbf{o}[j] \neq I} \mathbf{q}[j] \tag{4.49}$$

$$= \sum_{\mathbf{q} \in \{\pm 1\}^n} \Pr\left[\mathbf{q} | \mathbf{p}, \rho\right] \prod_{j:\, \mathbf{o}_j \neq I} \mathbf{q}[j] \tag{4.50}$$

$$= \sum_{\mathbf{q} \in \{\pm 1\}^n} \mathrm{tr}\left(\bigotimes_{j:\mathbf{o}[j] \neq I} \tfrac{1}{2}\left(\mathbf{q}[j] + \sigma_{\mathbf{p}[j]}\right) \bigotimes_{j:\mathbf{o}[j] = I} \tfrac{1}{2}\left(\sigma_I + \mathbf{q}[j]\sigma_{\mathbf{p}[j]}\right)\rho\right)$$

$$= \tfrac{1}{2^n} \sum_{\mathbf{q} \in \{\pm 1\}^n} \mathrm{tr}\left(\bigotimes_{j:\, \mathbf{o}[j] \neq I} \sigma_{\mathbf{o}[j]} \bigotimes_{j:\, \mathbf{o}[j] = I} \sigma_I \rho\right) = \mathrm{tr}\left(\bigotimes_{j=1}^{n} \sigma_{\mathbf{o}[j]}\rho\right) = \mathrm{tr}\left(O_{\mathbf{o}}\rho\right),$$

whenever $\mathbf{o} \triangleright \mathbf{p}$ (which ensures $\mathbf{o}[j] = \mathbf{p}[j]$ whenever $\mathbf{o}[j] \neq I$). Now, suppose that we perform a total of M Pauli measurements $\mathbf{p}_1, \ldots, \mathbf{p}_M$. The above relation suggests to approximate Pauli observables $\omega_\ell(\rho) = \mathrm{tr}(O_{\mathbf{o}_\ell}\rho)$ by empirical averages:

$$\hat{\omega}_\ell = \begin{cases} \frac{1}{h(\mathbf{o}_\ell;\mathbf{P})} \sum_{m:\mathbf{o}_\ell \triangleright \mathbf{p}_m} \prod_{j:\mathbf{o}_\ell[j]\neq I} \mathbf{q}_m[j] & \text{if } h(\mathbf{o}_\ell;\mathbf{P}) \geq 1 \\ 0 & \text{if } h(\mathbf{o}_\ell;\mathbf{P}) = 0. \end{cases} \tag{4.51}$$

Here, $h(\mathbf{o}_\ell;\mathbf{P}) = \sum_{m=1}^{M} \mathbf{1}\{\mathbf{o}_\ell \triangleright \mathbf{p}_m\}$ denotes the *hitting count*, i.e. the number of times a Pauli measurement $\mathbf{p}_m$ provides meaningful information about observable $\mathbf{o}_\ell$. If $h(\mathbf{o}_\ell;\mathbf{P}) = 0$, not a single Pauli measurement is compatible with the target observable in question and we set $\hat{\omega}_\ell = 0$, because we do not have any actionable advice. The above procedure allows us to jointly estimate L Pauli observables based on M Pauli measurement outcomes. The quality of reconstruction is exponentially suppressed in the number of times we hit each target Pauli observable.

Lemma 8. *Fix a collection of M Pauli measurements $\mathbf{P} = [\mathbf{p}_1, \ldots, \mathbf{p}_M]$, a collection of L Pauli observables $\omega_\ell(\rho) = \mathrm{tr}\left(O_{\mathbf{o}_\ell}\rho\right)$. Then, for all $\varepsilon > 0$*

$$\Pr\left[\max_{1\leq \ell \leq L} |\hat{\omega}_\ell - \omega_\ell(\rho)| \geq \varepsilon\right] \leq 2\sum_{\ell=1}^{L} \exp\left(-\tfrac{\varepsilon^2}{2} h(\mathbf{o}_\ell;\mathbf{P})\right). \tag{4.52}$$

Lemma 7 in the main text is an immediate consequence of this concentration inequality.

Proof. The union bound – also known as Boole's inequality – states that the probability associated with a union of events is upper bounded by the sum of individual event probabilities. For the task at hand, it implies

$$\Pr\left[\max_{1\leq \ell \leq L} |\hat{\omega}_\ell - \omega_\ell(\rho)| \geq \varepsilon\right] = \Pr\left[\bigcup_{\ell=1}^{L} \{|\hat{\omega}_\ell - \omega_\ell| \geq \varepsilon\}\right] \tag{4.53}$$

$$\leq \sum_{\ell=1}^{L} \Pr\left[|\hat{\omega}_\ell - \omega_\ell(\rho)| \geq \varepsilon\right]. \tag{4.54}$$

This allows us to treat individual deviation probabilities separately. Fix $1 \leq \ell \leq L$ and note that $\hat{\omega}_\ell$ is an empirical average of $M_\ell = h(\mathbf{o}_\ell;\mathbf{P})$ random signs $s_i^{(\ell)} = \prod_{j:\, \mathbf{o}_\ell[j]\neq I} \mathbf{q}_i[j] \in \{\pm 1\}$ that are independent each (they arise from different measurement outcomes). Empirical averages of independent signed random variables tend to concentrate sharply around their true expectation value $\mathbb{E}s_i^{(\ell)} = \mathrm{tr}(O_{\mathbf{o}_\ell}\rho)$. Hoeffding's inequality makes this intuition precise and asserts for any $\varepsilon > 0$

$$\Pr\left[|\hat{\omega}_\ell - \omega_\ell(\rho)| \geq \varepsilon\right] = \Pr\left[\left|\frac{1}{M_\ell}\sum_{i=1}^{M_\ell}\left(s_i^{(\ell)} - \mathbb{E}s_i^{(\ell)}\right)\right| \geq \varepsilon\right] \leq 2\exp\left(-\frac{\varepsilon^2}{2}M_\ell\right). \tag{4.55}$$

The claim follows, because such an exponential bound is valid for each term in Eq. (4.54). This also includes terms with zero hits ($M_\ell = 0$), because

$$\Pr\left[|\hat{\omega}_\ell - \omega_\ell| \geq \varepsilon\right] \leq 1 = \exp(-0/2) \tag{4.56}$$

and the claim follows. □

Derivation of Eq. (4.43) Note that each hitting count

$$h(\mathbf{o}_\ell;\mathbf{P}) = \sum_{m=1}^{M} \mathbf{1}\left\{\mathbf{o}_\ell \triangleright \mathbf{p}_m\right\} \tag{4.57}$$

is a sum of M indicator functions that can take binary values each. This structure allows us to rewrite the confidence bound (4.40) as

$$\begin{aligned}\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P}) &= \sum_{\ell=1}^{L}\exp\left(-\frac{\varepsilon^2}{2}h(\mathbf{o}_\ell;\mathbf{P})\right) = \sum_{\ell=1}^{L}\prod_{m'=1}^{M}\exp\left(-\frac{\varepsilon^2}{2}\mathbf{1}\left\{\mathbf{o}_\ell \triangleright \mathbf{p}_{m'}\right\}\right) \\ &= \sum_{\ell=1}^{L}\prod_{m'=1}^{M}\left(1 - \nu\mathbf{1}\left\{\mathbf{o}_\ell \triangleright \mathbf{p}_{m'}\right\}\right),\end{aligned} \tag{4.58}$$

where $\nu = 1 - \exp\left(-\varepsilon^2/2\right) \in (0,1)$. Next, note that each remaining indicator function can be further decomposed into a product of more elementary indicator functions:

$$\mathbf{1}\left\{\mathbf{o}_\ell \triangleright \mathbf{p}_{m'}\right\} = \prod_{k'=1}^{n}\mathbf{1}\left\{\mathbf{o}_\ell[k'] \triangleright \mathbf{p}_{m'}[k']\right\} \tag{4.59}$$

$$= \prod_{k'=1}^{n}\left(\mathbf{1}\left\{\mathbf{o}_\ell[k'] = I\right\} + \mathbf{1}\left\{\mathbf{o}_\ell[k'] = \mathbf{p}_{m'}[k']\right\}\right). \tag{4.60}$$

Finally, note that a randomly assigned single-qubit label $\mathbf{p}_m[j] \in \{X,Y,Z\}$ hits non-identity Pauli label $\mathbf{o}_\ell[j] \neq I$ with probability $1/3$. More precisely,

$$\mathbb{E}_{\mathbf{p}_m[j]}\left[\mathbf{1}\left\{\mathbf{o}_\ell[j] \triangleright \mathbf{p}_m[j]\right\}\right] = \Pr_{\mathbf{p}_m[j]}\left[\mathbf{o}_\ell[j] \triangleright \mathbf{p}_m[j]\right] \tag{4.61}$$

$$= (1/3)^{\mathbf{1}\{\mathbf{o}_\ell[j]\neq I\}} = \begin{cases} 1/3 & \text{if } \mathbf{o}_\ell[j] \neq I, \\ 1 & \text{if } \mathbf{o}_\ell[j] = I. \end{cases} \tag{4.62}$$

Together with independence, this observation allows us to compute expectation values of confidence bounds that are partially assigned already. Let $\mathbf{P}^\sharp$ denote the already assigned part that encompasses the first $m-1$ Pauli measurements, as well as the first k single-qubit labels of the m-th Pauli measurement: $\mathbf{P}^\sharp = \left[\mathbf{p}_1^\sharp, \ldots, \mathbf{p}_{m-1}^\sharp\right] \cup \left[\mathbf{p}_m^\sharp[1], \ldots, \mathbf{p}_m[k]^\sharp\right]$. We also assume that all remaining Pauli labels are assigned independently and uniformly at random ($\Pr\left[\mathbf{p}_{m'}[k'] = X\right] = \Pr\left[\mathbf{p}_{m'}[k'] = Y\right] = \Pr\left[\mathbf{p}_{m'}[k'] = Z\right] = 1/3$). Independence ensures that the conditional expectation factorizes nicely into individual components:

$$\mathbb{E}_{\mathbf{P}}\left[\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P})|\mathbf{P}^\sharp\right] \tag{4.63}$$

$$= \sum_{\ell=1}^{L} \prod_{m'=1}^{m-1} \left(1 - \nu \mathbf{1}\{\mathbf{o}_\ell \triangleright \mathbf{p}_{m'}^\sharp\}\right) \tag{4.64}$$

$$\times \left(1 - \nu \prod_{k'=1}^{k} \{\mathbf{o}_\ell[k'] \triangleright \mathbf{p}_m[k']\} \prod_{k'=k+1}^{n} \mathbb{E}_{\mathbf{p}_m[k']}\{\mathbf{o}_\ell[k'] \triangleright \mathbf{p}_m[k']\}\right)$$

$$\times \prod_{m'=m+1}^{M} \left(1 - \nu \prod_{k'=1}^{n} \mathbb{E}_{\mathbf{p}_{m'}[k']} \mathbf{1}\{\mathbf{o}_\ell[k'] \triangleright \mathbf{p}_{m'}[k']\}\right)$$

$$= \sum_{\ell=1}^{L} \prod_{m'=1}^{m-1} \left(1 - \nu \mathbf{1}\{\mathbf{o}_\ell \triangleright \mathbf{p}_{m'}^\sharp\}\right) \left(1 - \nu \prod_{k'=1}^{k} \{\mathbf{o}_\ell[k'] \triangleright \mathbf{p}_m[k']\} \prod_{k'=k+1}^{n} (1/3)^{\mathbf{1}\{\mathbf{o}_\ell[k']\neq I\}}\right)$$

$$\times \prod_{m'=m+1}^{M} \left(1 - \nu \prod_{k'=1}^{n} (1/3)^{\mathbf{1}\{\mathbf{o}_\ell[k']\neq I\}}\right).$$

Now, note that the exponent $\sum_{k'=k+1}^{n} \mathbf{1}\{\mathbf{o}_\ell[k'] \neq I\} = \mathrm{w}_{\neg k}(\mathbf{o}_\ell)$ captures the weight of the reduced Pauli string $[\mathbf{o}_\ell[k+1], \ldots, \mathbf{o}_\ell[n]]$ (in particular, $\mathrm{w}_{\neg 0}(\mathbf{o}_\ell) = \mathrm{w}(\mathbf{o}_\ell)$) Reading Eq. (4.58) backwards to recognize

$$\prod_{m'=1}^{m-1} \left(1 - \nu \mathbf{1}\{\mathbf{o}_\ell \triangleright \mathbf{p}_{m'}^\sharp\}\right) = \exp\left(-\tfrac{\varepsilon^2}{2} h(\mathbf{o}_\ell; [\mathbf{p}_1^\sharp, \ldots, \mathbf{p}_{m-1}^\sharp])\right) \tag{4.65}$$

further simplifies the expression:

$$\mathbb{E}_{\mathbf{P}}\left[\textsc{Conf}_\varepsilon(\mathbf{O};\mathbf{P}^\sharp)|\mathbf{P}^\sharp\right] \tag{4.66}$$

$$= \sum_{\ell=1}^{L} \exp\left(-\tfrac{\varepsilon^2}{2} h(\mathbf{o}_\ell; [\mathbf{p}_1^\sharp, \ldots, \mathbf{p}_{m-1}^\sharp])\right) \left(1 - \nu \prod_{k'=1}^{k} \{\mathbf{o}_\ell[k'] \triangleright \mathbf{p}_m[k']\} 3^{-\mathrm{w}_{\neg k}(\mathbf{o}_\ell)}\right) \tag{4.67}$$

$$\times \left(1 - \nu 3^{-\mathrm{w}(\mathbf{o}_\ell)}\right)^{M-m}.$$

4.8 Details regarding numerical experiments

In this section, we present all the details for reproducing the numerical experiments we discussed in previous sections.

Predicting quantum fidelities

This numerical experiment considers classical shadows based on random Clifford measurements. We exploit the Gottesman-Knill theorem for efficient classical computations. This well-known result states that Clifford circuits can be simulated efficiently on classical computers; see also (Aaronson and Gottesman, 2004) for an improved classical algorithm. This has allowed us to address rather large system sizes (more than 160 qubits). To test the performance of feature prediction with classical shadows we first have to simulate the (quantum) data acquisition phase. We do this by repeatedly executing the following (efficient) protocol:

1. Sample a Clifford unitary U from the Clifford group using the algorithm proposed in (Koenig and J. A. Smolin, 2014). This Clifford unitary is parameterized by $(\alpha, \beta, \gamma, \delta, r, s)$ which fully characterize its action on Pauli operators:

$$UP_j^X U^\dagger = (-1)^{r_j} \Pi_{i=1}^n (P_i^X)^{\alpha_{ji}} (P_i^Z)^{\beta_{ji}} \tag{4.68}$$
$$UP_j^Z U^\dagger = (-1)^{s_j} \Pi_{i=1}^n (P_i^X)^{\gamma_{ji}} (P_i^Z)^{\delta_{ji}} \tag{4.69}$$

   for all $j = 1, \ldots, n$. Here, P_j^X, P_j^Z are the Pauli X, Z-operators acting on the j-th qubit, and $\alpha_{ji}, \beta_{ji}, \gamma_{ji}, \delta_{ji}, r_j, s_j \in \{0, 1\}$.

2. Given a unitary U parameterized by $(\alpha, \beta, \gamma, \delta, r, s)$, we can apply U on any stabilizer state by changing the stabilizer generators and the destabilizers as defined in (Aaronson and Gottesman, 2004).

3. A computational basis measurement can be simulated using the standard algorithm provided in (Aaronson and Gottesman, 2004).

Although originally designed for pure target states $|\psi_i\rangle\langle\psi_i|$, we can readily extend this strategy to mixed states $\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$. Operationally speaking, mixed states arise from sampling from a pure state ensemble. This mixing process can be simulated efficiently on classical machines.

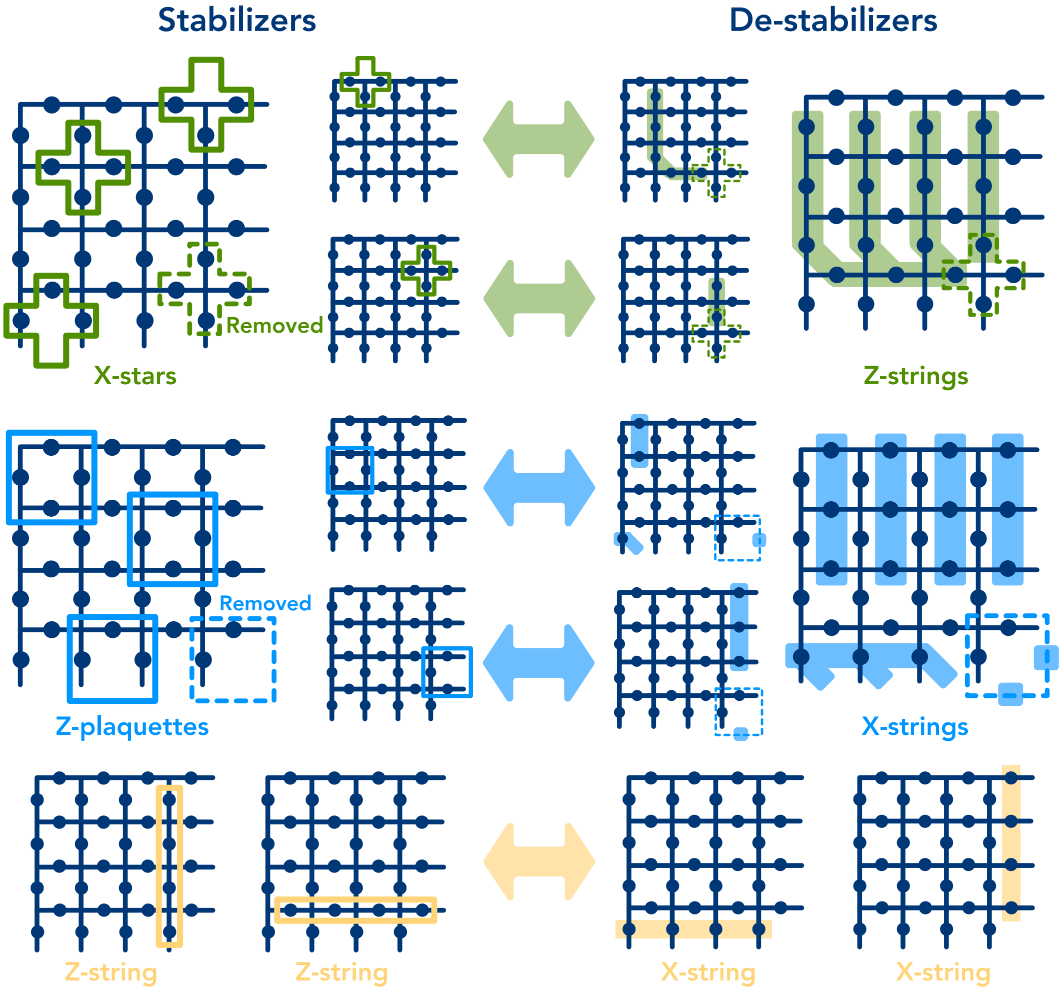


Figure 4.9: Stabilizers and de-stabilizers of the toric code that encodes $|00\rangle$.

For neural network quantum state tomography, we use the open-source code provided by the authors (Carrasquilla, Torlai, et al., 2019). The main challenge is generating training data, i.e. simulating measurement outcomes. For pure and noisy GHZ states, we use the tetrahedral POVM (Carrasquilla, Torlai, et al., 2019). For the toric code ground state, we use the Psi2 POVM (which is a measurement in the computational (Z-) basis). Note that measuring in the Z-basis is not a tomographically complete measurement, but we found machine learning models to perform better using Psi2. This is possibly because the pattern is much more obvious (closed-loop configurations) and the figure of merit used in NNQST is a classical fidelity.

A concrete algorithm for creating training data for pure GHZ states is included in the aforementioned open-source implementation of (Carrasquilla, Torlai, et al., 2019). It uses matrix product states to simulate quantum measurements efficiently. The training data for noisy GHZ states is a slight modification of the existing code. With probability $1 - p$, we sample a measurement outcome from the original state $|\psi^{+}_{\mathrm{GHZ}}\rangle = \frac{1}{\sqrt{2}}(|0\rangle^{\otimes n} + |1\rangle^{\otimes n})$. And with probability p, we sample a measurement outcome from $|\psi^{-}_{\mathrm{GHZ}}\rangle = \frac{1}{\sqrt{2}}(|0\rangle^{\otimes n} - |1\rangle^{\otimes n})$ (phase error). Since the figure of merit is the fidelity with the pure GHZ state in both pure and noisy GHZ experiment, we reuse the implementation provided in (Carrasquilla, Torlai, et al., 2019).

Creating training data for toric code is somewhat more involved. The goal is to sample a closed-loop configuration on a 2D torus uniformly at random. This can again be done using classical simulations of stabilizer states (Aaronson and Gottesman, 2004). The main technical detail is to create a tableau that contains both the stabilizer and the de-stabilizer for the state in question. The rich structure of the toric code renders this task rather easy. The stabilizers are the X-stars and the Z-plaquettes, with two Z-strings over the two loops of the torus. The de-stabilizer of each stabilizer is a Pauli-string that anticommutes with the stabilizer, but commutes with other stabilizers and other de-stabilizers. The full set of stabilizers and de-stabilizers for the toric code can be seen in Supplementary Figure 4.9.

Potential obstacles for learning certain quantum states

In our numerical studies, we have seen that neural network quantum state tomography based on deep generative models seems to have difficulty learning toric code ground states.

Here, we take a closer look at this curious aspect and construct a simple class of quantum states where efficient learning of the quantum state from the measurement data would violate a well-known computational hardness conjecture. First of all, each computational (Z-) basis measurement of the toric code produces a random bit-string. Most bits are sampled uniformly at random from $\{0, 1\}$ and the remaining bits are binary functions that only depend on these random bits. Consider a simple class of quantum states that mimic this property. Given $a \in \{0, 1\}^{n-1}$ and $f_a(x) = \sum_i a_i x_i$ (mod 2), we define $|a\rangle = \frac{1}{\sqrt{2^{n-1}}} \sum_{x \in \{0,1\}^{n-1}} |x\rangle \otimes |f_a(x)\rangle$. Such states can be created by preparing $|+\rangle$ on the first $n - 1$ qubits, $|0\rangle$ on the n-th qubit followed by CNOT gates between i-th qubit and n-th qubit for every $a_i = 1$. Measuring $|a\rangle$ in the computational (Z-) basis is equivalent to sampling the first $n - 1$ bits x uniformly at random. The final bit is characterized by the deterministic formula $f_a(x)$. Now, consider a (globally) depolarized version of this pure state:

$$\rho_a = \mathcal{D}_\eta(|a\rangle\langle a|) = (1-\eta)|a\rangle\langle a| + \tfrac{\eta}{2^n}\mathbb{I}^{\otimes n} \quad \text{for some } \eta \in (0,1). \tag{4.70}$$

One of the most widely used conjectures for building post-quantum cryptography is the hardness of learning with error (LWE) (Regev, 2009). LWE considers the task of learning a linear n-ary function f over a finite ring from noisy data samples $(x, f(x)+\eta)$, where x is sampled uniformly at random and η is some independent error. An efficient learning algorithm for LWE will be able to break many post-quantum cryptographic protocals that are believed to be hard even for quantum computers. The simplest example of LWE is called learning parity with error, where $f(x) = \sum_i a_i x_i \pmod 2$ for $x \in \{0,1\}^n$ and some unknown $a \in \{0,1\}^n$. Learning parity with error is also conjectured to be computationally hard (Blum, Kalai, and Wasserman, 2003). Since learning $|a\rangle$ from computational (Z-) basis measurements on ρ_a is equivalent to learning parity with error, it is unlikely there will be a neural network approach that can learn ρ_a efficiently.

Predicting witnesses for tripartite entanglement

This numerical experiment considers classical shadows based on random Clifford measurements. The numerical studies regarding entanglement witnesses are based locally rotated 3-qubit ($n=3$) GHZ states:

$$|\psi\rangle = U_A \otimes U_B \otimes U_C |\psi^+_{\mathrm{GHZ}}\rangle \quad \text{where } U_A, U_B, U_C \text{ are random single-qubit rotations.} \tag{4.71}$$

For $\rho = |\psi\rangle\langle\psi|$, we hope to verify the tripartite entanglement present in the system. To this end, we consider a simple family of entanglement witnesses with compatible structure:

$$O := O(V_A, V_B, V_C) = V_A \otimes V_B \otimes V_C |\psi^+_{\mathrm{GHZ}}\rangle\langle\psi^+_{\mathrm{GHZ}}| V_A^\dagger \otimes V_B^\dagger \otimes V_C^\dagger. \tag{4.72}$$

The single-qubit unitaries V_A, V_B, V_C parametrize different witnesses.

A complete characterization of entanglement in three-qubit systems can be found in Supplementary Figure 4.6. The expectation value of an entanglement witness $O(V_A, V_B, V_C)$ in the tripartite state ρ can certify that ρ belongs to a particular entanglement class. For example, it is known from the analysis in (Acín et al., 2001) that for any state ρ_s with only bipartite entanglement, $\mathrm{tr}(O\rho_s) \leq .5$, while for any state ρ_s with at most W-type entanglement, $\mathrm{tr}(O\rho_s) \leq .75$. Therefore verifying that $\mathrm{tr}(O\rho) > .5$ certifies that ρ has tripartite entanglement, while $\mathrm{tr}(O\rho) > .75$ certifies that ρ has GHZ-type entanglement.

After choosing random unitaries U_A, U_B, U_C to specify the GHZ-type state $|\psi\rangle$, we generate a list of random V_A, V_B, V_C to specify a set of potential entanglement witnesses for $|\psi\rangle$:

$$O_1 = O(V_{A,1}, V_{B,1}, V_{C,1}), \ldots, O_M = O(V_{A,M}, V_{B,M}, V_{C,M}). \tag{4.73}$$

If the randomly generated $O_i = O(V_{A,i}, V_{B,i}, V_{C,i})$ satisfies $\mathrm{tr}(O_i \left|\psi\right\rangle\left\langle\psi\right|) > 0.5$, then O_i is an entanglement witness for genuine tripartite entanglement, and if $\mathrm{tr}(O_i \left|\psi\right\rangle\left\langle\psi\right|) > 0.75$, then O_i is a witness for GHZ-type entanglement. We can compute the expected number of random candidates we have to test to find an observable O such that $\mathrm{tr}(O \left|\psi\right\rangle\left\langle\psi\right|) > 0.5$ or $\mathrm{tr}(O \left|\psi\right\rangle\left\langle\psi\right|) > 0.75$; these numbers are indicated as the dashed lines on the right side of Supplementary Figure 4.6.

Given the list of randomly generated witness candidates $O_1, \ldots, O_M$, we would like to predict $\mathrm{tr}(O_i|\psi\rangle\langle\psi|)$ for all $1 \leq i \leq M$. The naive approach is to directly measure all observables (witnesses). We refer to this as the direct measurement approach. For this approach, we consider the number of total experiments required to estimate every $\mathrm{tr}(O_i|\psi\rangle\langle\psi|)$ up to an error 0.1. Note that the number of required samples may vary from witness to witness — it depends on the variance associated with the estimation. In the worst case, one would need ≈ 100 measurements for each witness candidate.

Instead of this direct measurement approach, one could use classical shadows (Clifford measurements) to predict *all* the observables (witnesses) $O_1, \ldots, O_M$ at once. Because, $\mathrm{tr}(O_i^2) = 1$ for al $1 \leq i \leq M$, the shadow norm obeys $\|O_i\|^2_{\mathrm{shadow}} \leq 3\,\mathrm{tr}\left(O_i^2\right) = 3$, according to the analysis in Supplementary Section 4.3. Hence Theorem 7 shows that classical shadows can predict the expectation values of many candidate witnesses very efficiently.

In the numerical experiment, we gradually increased the number of random Clifford measurements we use to construct classical shadows until the classical shadows could accurately predict all $\mathrm{tr}(O_i \left|\psi\right\rangle\left\langle\psi\right|)$ up to 0.1-error. The results are shown in Supplementary Figure 4.6. Because the system size is small ($n = 3$ qubits), we simulate the quantum experiments classically by storing and processing all $2^3 = 8$ amplitudes. In practice, one should use statistics, like sample variance estimation or the bootstrap (Efron and R. J. Tibshirani, 1993), to determine confidence intervals and a posteriori guarantees. Quadratic function prediction with classical shadows (Clifford measurements) can be used to achieve this goal efficiently.

Predicting two-point correlation functions

Predicting two-point correlation function could be done efficiently using classical shadows based on random Pauli measurements. To facilitate direct comparison, this numerical experiment is designed to reproduce one of the core examples in in (Carrasquilla, Torlai, et al., 2019). In particular, we use the same data, downloaded from `https://github.com/carrasqu/POVM_GENMODEL`. The classical shadow (based on random Pauli basis measurements) replaces the original machine learning based approach for predicting local observables. We use multi-core CPU for training and making prediction with the machine learning model. The reported time is the total CPU time. Predicting local observables O using the (Pauli) classical shadow can be done efficiently by creating the reduced density matrix ρ_A, where A is the subsystem O acts on. The reduced density matrix ρ_A can be created by simply neglecting the data for the rest of the system. Importantly, $\mathcal{M}^{-1}(U^\dagger|\hat{b}\rangle\langle\hat{b}|U)$ is never created as an $2^n \times 2^n$ matrix. Taking the inner product of ρ_A with the local observables O yields the desired result.

Predicting subsystem Rényi entanglement entropies

We consider classical shadows based on random Pauli measurements for predicting subsystem entanglement entropies. In the first part of the experiment, we consider the ground state of a disordered Heisenberg model. The associated Hamiltonian is $H = \sum_i J_i \langle S_i \cdot S_{i+1} \rangle$, where each J_i is sampled uniformly (and independently) from the unit interval $[0, 1]$. The approximate ground state is found by implementing the recursive procedure from (Refael and E. Altman, 2013): identify the largest J_i, forming singlet for the connected sites, and reduce the system by removing J_i. We refer to (Refael and E. Altman, 2013) for details. In the experiment, we perform single-shot random Pauli basis measurements on the approximate ground state. I.e. we measure the state in a random Pauli basis only once and then choose a new random basis. However, in physical experiments, it is often easier to repeat a single Pauli basis measurement many times before re-calibrating to measure another Pauli basis. Performing a single random basis measurement for many repetitions can be beneficial experimentally compared to measuring a random basis every single time. Classical shadows (Pauli) are flexible enough to incorporate economic measurement strategies that take this discrepancy into account. We refer to the open source implementation in `https://github.com/momohuang/predicting-quantum-properties` for the exact details.

To obtain a reasonable benchmark, we compare this procedure with the approach proposed by Brydges *et al.* (Brydges et al., 2019). For a subsystem A comprised of k qubits, the approach proposed in (Brydges et al., 2019) for predicting the Rényi entropy works as follows. First, one samples a random single-qubit unitary rotations independently for all k qubits. Then, one applies the single-qubit unitary rotation to the system and measures the system in the computational basis to obtain a string of binary values $s \in \{0,1\}^k$. For each random unitary rotation, several repetitions are performed. The precise number of repetitions for a single random basis is a hyper-parameter that has to be optimized. The estimator for the Rényi entropy takes the following form:

$$\mathrm{tr}(\rho_A^2) = 2^k \sum_{s,s' \in \{0,1\}^k} (-2)^{-H(s,s')} \overline{P(s)P(s')}. \tag{4.74}$$

The function $H(s,s')$ is the Hamming distance between strings s and s' (i.e, the number of positions at which individual bits are different), while $P(s)$ and $P(s')$ are the probabilities for measuring ρ and obtaining the outcomes s and s', respectively. The probability $P(s)$ is a function that depends on the randomly sampled single-qubit rotation. $\overline{P(s)P(s')}$ is the expectation of $P(s)P(s')$ averaged over the random single-qubit rotations.

The random single-qubit rotations could be taken as single-qubit Haar-random rotations or single-qubit random Clifford rotations. The latter choice is equivalent to random Pauli measurements – the measurement primitive we consider for classical shadows also. For the test cases we considered, using random Pauli measurements yields similar (and sometimes improved) performance compared to single-qubit Haar-random unitary rotation. This allows the approach by (Brydges et al., 2019) and the procedure based on classical shadows to be compared on the same ground. We follow the strategy in (Brydges et al., 2019) to estimate the formula in Eq. (4.74). First, we sample N_U random unitary rotations. For each random unitary rotation, we perform N_M repetitions of rotating the system and measuring in the computational basis. The N_M measurement outcomes allow us to construct an empirical distribution for $P(s)$. Thus we could use the N_M measurement outcomes to estimate $2^k \sum_{s,s' \in \{0,1\}^k} (-2)^{-H(s,s')} P(s)P(s')$ for a single random unitary rotation. We then take the average over N_U different random unitary rotations. Choosing a suitable parameter for N_U and N_M is nontrivial. We employ the strategy advocated in (Brydges et al., 2019) for finding the best parameter for N_U and N_M. This strategy is called grid search and is performed by trying many different choices for N_U, N_M and recording the best one.

Variational quantum simulation of the lattice Schwinger model

The application for variational quantum simulation uses classical shadows based on random Pauli measurements which is designed to predict a large number of local observables efficiently. It is based on the seminal work presented in (Kokail et al., 2019). After a Kogut-Susskind encoding to map fermionic configurations to a spin-$1/2$ lattice with an even number N of lattice sites and a subsequent Jordan-Wigner transform, the Hamiltonian becomes

$$\hat{H} = \underbrace{\frac{w}{2}\sum_{j=1}^{N-1} P_j^X P_{j+1}^X}_{\hat{\Lambda}_X} + \underbrace{\frac{w}{2}\sum_{j=1}^{N-1} P_j^Y P_{j+1}^Y}_{\hat{\Lambda}_Y} + \underbrace{\sum_{j=1}^{N} d_j P_j^z + \sum_{j=1}^{N-2}\sum_{j'=j+1}^{N-1} c_{j,j'} P_j^z P_{j'}^z}_{\hat{\Lambda}_Z} . \tag{4.75}$$

Here, P_j^X, P_j^Y, P_j^Z denote Pauli-X, Y, Z operators acting on the j-th qubit ($1 \leq j \leq N$). This Hamiltonian has very advantageous structure. Each of the three contributions can be estimated by performing a single Pauli basis measurement (measure every qubit in the X basis to determine $\hat{\Lambda}_X$, measure every qubit in the Y basis to determine $\hat{\Lambda}_Y$ and measure every qubit in the Z basis to determine $\hat{\Lambda}_Z$). The measurement of the Hamiltonian variance $\langle \hat{H}^2 \rangle - \langle \hat{H} \rangle^2$ is more complicated, because $\langle \hat{H}^2 \rangle$ does not decompose nicely. To determine its value, we must first measure $\hat{\Lambda}_X^2$, $\hat{\Lambda}_Y^2$ and $\hat{\Lambda}_Z^2$. This is the easy part, because 3 measurement bases once more suffice. However, in addition, we must also estimate the anti-commutators $\{\hat{\Lambda}_X, \hat{\Lambda}_Y\}$, $\{\hat{\Lambda}_X, \hat{\Lambda}_Z\}$, $\{\hat{\Lambda}_Y, \hat{\Lambda}_Z\}$. This may be achieved by measuring the following k-local observables (with k at most 4):

$$\begin{aligned}
&\{\hat{\Lambda}_X, \hat{\Lambda}_Y\} : \ P_j^X P_{j+1}^X P_{j'}^Y P_{j'+1}^Y, \\
&\qquad \forall j, j' \in \{1, N-1\}, \text{ s.t. } j \neq j', j \neq j'+1, j+1 \neq j', \\
&\{\hat{\Lambda}_X, \hat{\Lambda}_Z\} : \ P_j^X P_{j+1}^X P_{j'}^Z P_{j''}^Z, \\
&\qquad \forall j, j', j'' \in \{1, N-1\}, \text{ s.t. } j \neq j', j \neq j'', j+1 \neq j', j+1 \neq j'', j' < j'', \\
&\{\hat{\Lambda}_X, \hat{\Lambda}_Z\} : \ P_j^X P_{j+1}^X P_{j'}^Z, \\
&\qquad \forall j, j' \in \{1, N-1\}, \text{ s.t. } j \neq j', j+1 \neq j', \\
&\{\hat{\Lambda}_Y, \hat{\Lambda}_Z\} : \ P_j^Y P_{j+1}^Y P_{j'}^Z P_{j''}^Z, \\
&\qquad \forall j, j', j'' \in \{1, N-1\}, \text{ s.t. } j \neq j', j \neq j'', j+1 \neq j', j+1 \neq j'', j' < j'', \\
&\{\hat{\Lambda}_Y, \hat{\Lambda}_Z\} : \ P_j^Y P_{j+1}^Y P_{j'}^Z, \\
&\qquad \forall j, j' \in \{1, N-1\}, \text{ s.t. } j \neq j', j+1 \neq j',
\end{aligned} \tag{4.76}$$

Although local, estimating all observables of this form is the main bottleneck of the entire procedure. To minimize the number of measurement bases, the original work (Kokail et al., 2019) has performed an analysis of symmetry in the lattice Schwinger model. First, the target Hamiltonian in Equation (4.75) satisfies $[\hat{H}, \sum_i P_i^Z] = 0$, which corresponds to a charge conservation symmetry in the scalar fermionic field. (Kokail et al., 2019) further consider a charge symmetry subspace with $\sum_i P_i^Z = 0$, which corresponds to a $\hat{CP}$ symmetry. In this subspace, we have $\langle\{\hat{\Lambda}_X, \hat{\Lambda}_Z\}\rangle = \langle\{\hat{\Lambda}_Y, \hat{\Lambda}_Z\}\rangle$. This ensures that we only have to estimate local observables corresponding to $\{\hat{\Lambda}_X, \hat{\Lambda}_Y\}$ and $\{\hat{\Lambda}_X, \hat{\Lambda}_Z\}$. In the original setup (Kokail et al., 2019), this task was achieved by measuring roughly $2N$ bases in total. We refer to (Kokail et al., 2019, Appendix B and Appendix C) for further details and explanation. We propose to replace this original approach by linear feature prediction with classical shadows (Pauli measurements).

For classical shadows based on random Pauli measurements, every measurement basis is an independent random X, Y, or Z measurement for every qubit. This randomized general-purpose procedure does not take into account the fact that we want to measure a specific set of k-local observables given in Equation (4.76). The derandomized version of classical shadows is based on the concept of pessimistic estimators (Raghavan, 1988; J. Spencer, 1994) (see also (Wigderson and D. Xiao, 2008) for an application with quantum information context). It removes the original randomness by utilizing the knowledge of this specific set of k-local observables. When we throw a dice (or coin) to decide whether we want to measure in either, the $X-$, the $Y-$, or the $Z-$basis, the derandomized version would choose the measurement basis (X, Y, or Z) that would lead to the best expected performance on the set of k-local observables given in Equation (4.76). The expected performance is computed based on random Pauli basis measurements and the analysis in later sections. The derandomized version of classical shadows would perform at least as well as the original randomized version. Furthermore, due to the dependence on the specific set of observables for choosing the measurement bases, the derandomized version can exploit advantageous structures in the set of observables we want to measure. As detailed in the main text, classical shadows based on random Pauli measurements provide improvement only for larger system sizes (more than 50 qubits). A derandomized version of classical shadows improves upon the randomized version and leads to a substantial improvement in efficiency and scalability over a wide range of system sizes. As an added benefit, derandomization can be completely automated and does not depend on the concrete set of target observables. We refer to `https://github.com/momohuang/predicting-quantum-properties` for a (roughly linear time) algorithm that derandomizes random Pauli measurements for any collection of target observables with Pauli structure.

4.9 Additional computations and proofs for predicting linear functions

Background: Clifford circuits and the stabilizer formalism

Clifford circuits were introduced by Gottesman (Gottesman, 1997) and form an indispensable tool in quantum information processing. Applications range from quantum error correction (Michael A. Nielsen and Isaac L. Chuang, 2000), to measurement-based quantum computation (Raussendorf and Hans J. Briegel, 2001; H. J. Briegel et al., 2009) and randomized benchmarking (Emerson, Alicki, and Życzkowski, 2005; Knill et al., 2008; Magesan, Gambetta, and Emerson, 2011). For systems comprised of n qubits, the Clifford group is generated by CNOT, Hadamard and phase gates. This results in a finite group of cardinality $2^{O(n^2)}$ that maps (tensor products of) Pauli matrices to Pauli matrices upon conjugation. This underlying structure allows for efficiently storing and simulating Clifford circuits on classical computers – a result commonly known as Gottesman-Knill theorem. The n-qubit Clifford group $\mathrm{Cl}(2^n)$ also comprises a *unitary 3-design* (Webb, 2016; Zhu, 2017; Richard Kueng and David Gross, 2015). Sampling Clifford circuits uniformly at random reproduces the first 3 moments of the full unitary group endowed with the Haar measure. For $k = 1, 2, 3$

$$\mathbb{E}_{U\sim\mathrm{Cl}(2^n)}\left(UXU^\dagger\right)^{\otimes k} = \int_{U(d)} (UAU^\dagger)^{\otimes k}\mathrm{d}\mu_{\mathrm{Haar}}(U) \quad \text{for all } 2^n \times 2^n \text{ matrices } A. \tag{4.77}$$

The right hand side of this equation can be evaluated explicitly by using techniques from representation theory, see e.g. (D. Gross, Krahmer, and R. Kueng, 2015, Sec. 3.5). This in turn yields closed-form expressions for Clifford averages of linear and quadratic operator-valued functions. Choose a unit vector $x \in \mathbb{C}^{2^n}$ and let $\mathbb{H}_{2^n}$ denote the space of Hermitian $2^n \times 2^n$ matrices. Then,

$$\mathbb{E}_{U\sim\mathrm{Cl}(2^n)} U^\dagger|x\rangle\langle x|U^\dagger\langle x|UAU^\dagger|x\rangle \tag{4.78}$$

$$=\frac{A+\mathrm{tr}(A)\mathbb{I}}{(2^n+1)2^n} = \frac{1}{2^n}\mathcal{D}_{1/(2^n+1)}(A) \quad \text{for } A \in \mathbb{H}_{2^n}, \tag{4.79}$$

$$\mathbb{E}_{U\sim\mathrm{Cl}(2^n)} U^\dagger|x\rangle\langle x|U\langle x|UB_0U^\dagger|x\rangle\langle x|UC_0U^\dagger|x\rangle \tag{4.80}$$

$$=\frac{\mathrm{tr}(B_0C_0)\mathbb{I}+B_0C_0+C_0B_0}{(2^n+2)(2^n+1)2^n} \quad \text{for } B_0, C_0 \in \mathbb{H}_{2^n} \text{ traceless.} \tag{4.81}$$

Here, $\mathcal{D}_p(A) = pA + (1-p)\frac{\mathrm{tr}(A)}{2^n}\mathbb{I}$ denotes a n-qubit depolarizing channel with loss parameter p. Linear maps of this form can be readily inverted. In particular,

$$\mathcal{D}^{-1}_{1/(2^n+1)}(A) = (2^n+1)A - \mathrm{tr}(A)\mathbb{I} \quad \text{for any } A \in \mathbb{H}_{2^n}. \tag{4.82}$$

These closed-form expressions allow us to develop very concrete strategies and rigorous bounds for classical shadows based on (global and local) Clifford circuits.

Performance of classical shadows based on random Clifford measurements

Proposition 3. *Adopt a "random Clifford basis" measurement primitive, i.e. each rotation $\rho \mapsto U\rho U^\dagger$ is chosen uniformly from the n qubit Clifford group* $\mathrm{Cl}(2^n)$*. Then, the associated classical shadow is*

$$\hat{\rho} = (2^n + 1)U^\dagger|\hat{b}\rangle\langle\hat{b}|U - \mathbb{I}, \tag{4.83}$$

where $\hat{b} \in \{0,1\}^n$ is the observed computational basis measurement outcome (of the rotated state $U\rho U^\dagger$). Moreover, the norm defined in Eq. (4.12) *is closely related to the Hilbert-Schmidt norm:*

$$\mathrm{tr}\left(O_0^2\right) \leq \|O_0\|_{\mathrm{shadow}}^2 \leq 3\mathrm{tr}\left(O_0^2\right) \quad \textit{for any traceless } O_0 \in \mathbb{H}_{2^n}. \tag{4.84}$$

Note that passing from O to its traceless part $O_0 = O - \frac{\mathrm{tr}(O)}{2^n}\mathbb{I}$ is a contraction in Hilbert-Schmidt norm:

$$\mathrm{tr}\left(O_0^2\right) = \mathrm{tr}(O^2) - \frac{\mathrm{tr}(O)^2}{2^n} \leq \mathrm{tr}(O^2). \tag{4.85}$$

Hence, we can safely replace the upper bound in Eq. (4.84) by $3\mathrm{tr}(O^2)$ — the Hilbert Schmidt norm (squared) of the original observable.

Proof. Eq. (4.79) readily provides a closed-form expression for the measurement channel defined in Eq. (4.7):

$$\begin{aligned}\mathcal{M}(\rho) &= \sum_{b\in\{0,1\}^n} \mathbb{E}_{U\sim\mathrm{Cl}(2^n)}\langle b|U\rho U^\dagger|b\rangle U^\dagger|b\rangle\langle b|U &(4.86)\\ &= \sum_{b\in\{0,1\}^n} \frac{1}{2^n}\mathcal{D}_{1/(2^n+1)}(\rho) = \mathcal{D}_{1/(2^n+1)}(\rho). &(4.87)\end{aligned}$$

This depolarizing channel can be readily inverted, see Eq. (4.82). In particular,

$$\hat{\rho} = \mathcal{M}^{-1}\left(U^\dagger|\hat{b}\rangle\langle\hat{b}|U\right) = (2^n+1)U^\dagger|\hat{b}\rangle\langle\hat{b}|U - \mathbb{I} \quad \text{and} \quad \mathcal{M}^{-1}(O_0) = (2^n+1)O_0 \tag{4.88}$$

for any traceless matrix $O_0 \in \mathbb{H}_{2^n}$. The latter reformulation considerably simplifies the expression for the norm $\|O_0\|_{\mathrm{shadow}}^2$ defined in Eq. (4.12). A slight reformulation allows us to furthermore capitalize on Eq. (4.81) to exactly compute this norm for traceless observables:

$$\|O_0\|^2_{\text{shadow}} = \max_{\sigma \text{ state}} \operatorname{tr}\big(\sigma \sum_{b\in\{0,1\}^n} \mathbb{E}_{U\sim \mathrm{Cl}(2^n)} U^\dagger |b\rangle\!\langle b| U \langle b|U(2^n+1)O_0U^\dagger|b\rangle^2\big)$$

$$= \max_{\sigma \text{ state}} \operatorname{tr}\left(\sigma \, \frac{(2^n+1)^2\left(\operatorname{tr}(O_0^2)\mathbb{I}+2O_0^2\right)}{(2^n+2)(2^n+1)2^n}\right) \tag{4.89}$$

$$=\frac{2^n+1}{2^n+2}\max_{\sigma \text{ state}}\left(\operatorname{tr}(\sigma)\operatorname{tr}(O_0^2)+2\operatorname{tr}\left(\sigma O_0^2\right)\right). \tag{4.90}$$

To further simplify this expression, recall $\operatorname{tr}(\sigma)=1$ and note that

$$\max_{\sigma \text{ state}} \operatorname{tr}(\sigma O_0^2) = \|O_0^2\|_\infty, \tag{4.91}$$

where $\|\cdot\|_\infty$ denotes the spectral norm. The bound Eq. (4.84) then foloows from the elementary relation between the spectral and Hilbert-Schmidt norms: $\|O_0^2\|_\infty \leq \operatorname{tr}(O_0^2)$. □

Performance of classical shadows based on random Pauli measurements

Proposition 4. *Adopt a "random Pauli basis" measurement primitive, i.e. each rotation $\rho \mapsto U\rho U^\dagger$ is a tensor product $U_1 \otimes \cdots \otimes U_n$ of randomly selected single-qubit Clifford gates $U_1, \ldots, U_n \in \mathrm{Cl}(2)$. Then, the associated classical shadow is*

$$\hat{\rho} = \bigotimes_{j=1}^{n}\left(3U_j^\dagger|\hat{b}_j\rangle\!\langle\hat{b}_j|U_j - \mathbb{I}\right) \tag{4.92}$$

$$\textit{where} \quad |\hat{b}\rangle = |\hat{b}_1\rangle\otimes\cdots\otimes|\hat{b}_n\rangle \textit{ and } \hat{b}_1,\ldots,\hat{b}_n \in \{0,1\}. \tag{4.93}$$

Moreover, the norm defined in Eq. (4.12) *respects locality. Suppose that $O \in \mathbb{H}_2^{\otimes k}$ only acts nontrivially on k-qubits, e.g. $O = \tilde{O}\otimes \mathbb{I}^{\otimes(n-k)}$ with $\tilde{O} \in \mathbb{H}_2^{\otimes k}$. Then $\|O\|_{\text{shadow}} = \|\tilde{O}\|_{\text{shadow}}$, where $\|\tilde{O}\|_{\text{shadow}}$ is the same norm, but for k-qubit systems.*

Proof. Unitary rotation and computational basis measurements factorize completely into tensor products. This insight allows us to decompose the measurement channel $\mathcal{M}$ defined in Eq. (4.7) into a tensor product of single-qubit operations. For elementary tensor products $X_1\otimes\cdots\otimes X_n \in \mathbb{H}_2^{\otimes n}$ we can apply Eq. (4.79) separately for each single-qubit action and infer

$$\mathcal{M}\left(X_1\otimes\cdots\otimes X_n\right) = \bigotimes_{j=1}^{n}\big(\sum_{b_j\in\{0,1\}} \mathbb{E}_{U_j\sim\mathrm{Cl}(2)} U_j^\dagger|b\rangle\!\langle b|U_j\langle b|U_jX_jU_j^\dagger|b\rangle\big)$$

$$= \bigotimes_{j=1}^{n} \Big(\sum_{b_j \in \{0,1\}} \frac{1}{2} \mathcal{D}_{1/(2+1)}(\rho_j) \Big) = \mathcal{D}_{1/3}^{\otimes n} \left(X_1 \otimes \cdots \otimes X_n \right). \tag{4.94}$$

Linear extension to all of $\mathbb{H}_2^{\otimes n}$ yields the following formula for $\mathcal{M}$ and its inverse:

$$\mathcal{M}(X) = \left(\mathcal{D}_{1/3}\right)^{\otimes n}(X) \quad \text{and} \quad \mathcal{M}^{-1}(X) = \left(\mathcal{D}_{1/3}^{-1}\right)^{\otimes n}(X) \quad \text{for all } X \in \mathbb{H}_2^{\otimes n}, \tag{4.95}$$

where $\mathcal{D}_{1/3}^{-1}(Y) = 3Y - \mathrm{tr}(Y)\mathbb{I}$ according to Eq. (4.82). This formula readily yields a closed-form expression for the classical shadow. Use

$$U^\dagger |\hat{b}\rangle\langle\hat{b}| U = \bigotimes_{j=1}^{n} U_j |\hat{b}_j\rangle\langle\hat{b}_j| U_j \tag{4.96}$$

to conclude

$$\hat{\rho} = \mathcal{M}^{-1}\left(U^\dagger |\hat{b}\rangle\langle\hat{b}| U\right) = \bigotimes_{j=1}^{n} \mathcal{D}_{1/3}^{-1}\left(U_j^\dagger |\hat{b}_j\rangle\langle\hat{b}_j| U_j\right) = \bigotimes_{j=1}^{n}\left(3U_j^\dagger |\hat{b}_j\rangle\langle\hat{b}_j| U - \mathbb{I}\right). \tag{4.97}$$

For the second claim, we exploit a key feature of depolarizing channels and their inverses. The identity matrix is a fix-point, i.e. $\mathcal{D}_{1/3}^{-1}(\mathbb{I}) = \mathbb{I} = \mathcal{D}_{1/3}(\mathbb{I})$. For k-local observables, e.g. $O = \tilde{O} \otimes \mathbb{I}^{\otimes(n-k)}$, this feature ensures

$$\mathcal{M}^{-1}\left(\tilde{O} \otimes \mathbb{I}^{\otimes(n-k)}\right) = \left(\left(\mathcal{D}_{1/3}^{-1}\right)^{\otimes k}(\tilde{O})\right) \otimes \mathbb{I}^{\otimes(n-k)} = \tilde{\mathcal{M}}^{-1}(\tilde{O}) \otimes \mathbb{I}^{\otimes(n-k)}, \tag{4.98}$$

where $\tilde{\mathcal{M}}^{-1}(X) = (\mathcal{D}_{1/3}^{-1})^{\otimes k}(X)$ denotes the inverse channel of a k-qubit local Clifford measurement procedure. This observation allows us to compress the norm (4.12) to the "active" subset of k qubits. Exploit the tensor product structure $U = U_1 \otimes \cdots \otimes U_n$ with $U_i \sim \mathrm{Cl}(2)$ to conclude

$$\begin{aligned} &\left\|\tilde{O} \otimes \mathbb{I}^{\otimes(n-k)}\right\|_{\text{shadow}}^2 \\ &= \max_{\sigma:\,\text{state}} \mathbb{E}_{U \sim \mathrm{Cl}(2)^{\otimes n}} \sum_{b \in \{0,1\}^n} \langle b|U\sigma U^\dagger|b\rangle \langle b|U\mathcal{M}^{-1}(O \otimes \mathbb{I}^{\otimes(n-k)}U^\dagger|b\rangle^2 \\ &= \max_{\sigma:\,\text{state}} \mathbb{E}_{U \sim \mathrm{Cl}(2)^{\otimes k}} \sum_{b \in \{0,1\}^k} \langle b|U\mathrm{tr}_{k+1,\ldots,n}(\sigma)U^\dagger|b\rangle \langle b|U\tilde{\mathcal{M}}^{-1}(\tilde{O})U^\dagger|b\rangle^2, \end{aligned} \tag{4.99}$$

where $\mathrm{tr}_{k+1,\ldots,n}(\sigma)$ denotes the partial trace over all "inactive" subsystems. Partial traces preserve the space of all quantum states. So maximizing over all partial traces $\mathrm{tr}_{k+1,\ldots,n}(\sigma)$ is equivalent to maximizing over all k-qubit states and we exactly recover the norm $\|\tilde{O}\|_{\text{shadow}}^2$ on k qubits. Finally, it is easy to check that the actual location of the active k-qubit support of O does not affect the argument. □

Recall that the (squared) norm $\|\cdot\|^2_{\text{shadow}}$ is the most important figure of merit for feature prediction with classical shadows. According to Theorem 9, $\max_i \|O_i\|^2_{\text{shadow}}$ determines the number of samples required to accurately predict a collection of linear functions $\mathrm{tr}(O_1\rho),\ldots,\mathrm{tr}(O_M\rho)$. Viewed from this angle, Proposition 4 has profound consequences for predicting (collections of) local observables under the local Clifford measurement primitive. For each local observable O_i, the norm $\|O_i\|^2_{\text{shadow}}$ collapses to its active support, regardless of its precise location. The size of these supports is governed by the locality alone, not the total number of qubits!

It is instructive to illustrate this point with a simple special case first.

Lemma 9. *Let O be a single k-local Pauli observable, e.g. $O = P_{p_1}\otimes\cdots\otimes P_{p_k}\otimes \mathbb{I}^{\otimes(n-k)}$, where $p_j\in\{X,Y,Z\}$. Then, $\|O\|^2_{\text{shadow}} = 3^k$, for any choice of the k qubits where nontrivial Pauli matrices act. This scaling can be generalized to arbitrary elementary tensor products supported on k qubits,* e.g. $O = O_1\otimes\cdots\otimes O_k\otimes\mathbb{I}^{\otimes(n-k)}$.

Proof. Pauli matrices are traceless and obey, $P^2_{p_j} = \mathbb{I}$ and $\mathcal{D}^{-1}_{1/3}(P_{p_j}) = 3P_{p_j}$ for each $p_j\in\{X,Y,Z\}$. Proposition 4 and the tensor product structure of the problem then ensure

$$\begin{aligned}
&\|O\|^2_{\text{shadow}}\\
=&\|P_{p_1}\otimes\cdots\otimes P_{p_k}\|^2_{\text{shadow}}\\
=&\max_{\sigma:\text{ state}} \mathbb{E}_{U\sim\mathrm{Cl}(2)^{\otimes k}} \sum_{b\in\{0,1\}^n} \langle b|U^\dagger\sigma U|b\rangle\langle b|U(\mathcal{D}^{-1}_{1/3})^{\otimes k}(P_1\otimes\cdots\otimes P_k)U^\dagger|b\rangle^2\\
=&\max_{\sigma:\text{ state}} \mathrm{tr}\Big(\sigma\bigotimes_{j=1}^{k}\big(\sum_{b_j\in\{0,1\}}\mathbb{E}_{U_j\sim\mathrm{Cl}(2)}U^\dagger|b_j\rangle\langle b_j|U\langle b_j|U3P_jU^\dagger U|b_j\rangle^2\big)\Big)\\
=&\max_{\sigma:\text{ state}} \mathrm{tr}\Big(\sigma\bigotimes_{j=1}^{k}\big(9\sum_{b\in\{0,1\}}\frac{\mathrm{tr}\left(P_j^2\right)\mathbb{I}+2P_j^2}{(2+2)(2+1)2}\big)\Big) = \max_{\sigma:\text{ state}}\mathrm{tr}\Big(\sigma\bigotimes_{j=1}^{k}3\mathbb{I}\Big) = 3^k, \quad (4.100)
\end{aligned}$$

where we have used Eq. (4.81) to explicitly evaluate the single qubit Clifford averages.

We leave the extension to more general tensor product observables as an exercise for the dedicated reader. □

The norm expression in Lemma 9 scales exponentially in the locality k, but is independent of the total number of qubits n. The compression property (Prop. 4) suggests that this desirable feature should extend to general k-local observables.

And, indeed, it is relatively straightforward to obtain crude upper bounds that scale with 3^{2k}. The additional factor of two, however, effectively doubles the locality parameter and can render conservative feature prediction with classical shadows prohibitively expensive in concrete applications.

The main result of this section considerably improves upon these crude bounds and *almost* reproduces the (tight) scaling associated with k-local Pauli observables.

Proposition 5. *Let O be a k-local observable, e.g. $O = \tilde{O} \otimes \mathbb{I}^{\otimes(n-k)}$ with $\tilde{O} \in \mathbb{H}_2^{\otimes k}$ Then,*

$$\|O\|_{\mathrm{shadow}}^2 \leq 4^k \|O\|_\infty^2, \quad \textit{where } \|\cdot\|_\infty \textit{ denotes the spectral/operator norm.} \tag{4.101}$$

The same bound holds for the shadow norm of the traceless part of O: $\|O - \frac{\mathrm{tr}(O)}{2^n}\mathbb{I}\|_{\mathrm{shadow}}^2 \leq 4^k \|O\|_\infty^2$.

The proof is considerably more technical than the proof of Lemma 9 and relies on the following auxiliary result.

Lemma 10. *Fix two k-qubit Pauli observables $P_{\mathbf{p}} = P_{p_1} \otimes \cdots \otimes P_{p_k}, P_{\mathbf{q}} = P_{q_1} \otimes \cdots \otimes P_{q_k}$ with $\mathbf{p}, \mathbf{q} \in \{\mathbb{I}, X, Y, Z\}^k$. Then, the following is true for any state σ:*

$$\mathbb{E}_{U \sim \mathrm{Cl}(2)^{\otimes k}} \sum_{b \in \{0,1\}^k} \langle b|U\sigma U^\dagger|b\rangle \langle b|U(\mathcal{D}_{1/3}^{-1})^{\otimes k}(P_{\mathbf{p}})U^\dagger|b\rangle \langle b|U(\mathcal{D}_{1/3}^{-1})^{\otimes k}(P_{\mathbf{q}})U^\dagger|b\rangle \tag{4.102}$$

$$= f(\mathbf{p}, \mathbf{q}) \mathrm{tr}\left(\sigma P_{\mathbf{p}} P_{\mathbf{q}}\right), \tag{4.103}$$

where $f(\mathbf{p}, \mathbf{q}) = 0$ whenever there exists an index i such that $p_i \neq q_i$ and $p_i, q_i \neq \mathbb{I}$. Otherwise, $f(\mathbf{p}, \mathbf{q}) = 3^s$, where s is the number of non-identity Pauli indices that match ($s = |\{i : p_i = q_i, p_i \neq \mathbb{I}\}|$).

This combinatorial formula follows from a straightforward, but somewhat cumbersome, case-by-case analysis based on the (single-qubit) relations (4.79) and (4.81). We include a proof at the end of this subsection.

Proof of Proposition 5. Proposition 4 allows us to restrict our attention to the relevant k-qubit region on which $\tilde{O} \in \mathbb{H}_2^{\otimes k}$ acts nontrivially. Next, expand $\tilde{O}$ in the (tensor product) Pauli basis, i.e. $\tilde{O} = \sum_{\mathbf{p}} \alpha_{\mathbf{p}} P_{\mathbf{p}}$ with $\mathbf{p} \in \{\mathbb{I}, X, Y, Z\}^k$. Fix an arbitrary k-qubit state σ and use Lemma 10 to conclude

$$\|\tilde{O}\|_{\mathrm{shadow}}^2 \tag{4.104}$$

$$
\begin{aligned}
&= \max_{\sigma \text{ state}} \mathbb{E}_{U\sim \mathrm{Cl}(2)^{\otimes k}} \sum_{b\in\{0,1\}^k} \langle b|U\sigma U^\dagger|b\rangle \langle b|U(\mathcal{D}_{1/3}^{-1})^{\otimes k}(\tilde{O})U^\dagger|b\rangle^2 \\
&= \max_{\sigma \text{ state}} \sum_{\mathbf{p},\mathbf{q}} \alpha_{\mathbf{p}}\alpha_{\mathbf{q}} \mathbb{E}_{U\sim \mathrm{Cl}(2)^{\otimes k}} \sum_{b\in\{0,1\}^k} \langle b|U\sigma U^\dagger|b\rangle \qquad (4.105) \\
&\qquad \langle b|U(\mathcal{D}_{1/3}^{-1})^{\otimes k}(P_{\mathbf{p}})U^\dagger|b\rangle \langle b|U(\mathcal{D}_{1/3}^{-1})^{\otimes k}(P_{\mathbf{q}})U^\dagger|b\rangle \\
&= \max_{\sigma \text{ state}} \sum_{\mathbf{p},\mathbf{q}} \alpha_{\mathbf{p}}\alpha_{\mathbf{q}} f(\mathbf{p},\mathbf{q}) \mathrm{tr}\left(\sigma P_{\mathbf{p}} P_{\mathbf{q}}\right) = \max_{\sigma \text{ state}} \mathrm{tr}\left(\sigma \sum_{\mathbf{p},\mathbf{q}} \alpha_{\mathbf{p}}\alpha_{\mathbf{q}} f(\mathbf{p},\mathbf{q}) \mathrm{tr}\left(\sigma P_{\mathbf{p}} P_{\mathbf{q}}\right)\right) \\
&= \left\| \sum_{\mathbf{p},\mathbf{q}} \alpha_{\mathbf{p}}\alpha_{\mathbf{q}} f(\mathbf{p},\mathbf{q}) \mathrm{tr} P_{\mathbf{p}} P_{\mathbf{q}} \right\|_\infty , \qquad (4.106)
\end{aligned}
$$

where $f(\mathbf{p},\mathbf{q})$ is the combinatorial function defined in Lemma 10. The last equality follows from the dual characterization of the spectral norm:

$$
\|A\|_\infty = \max_{\sigma\text{: state}} \mathrm{tr}(\sigma A) \qquad (4.107)
$$

for any positive semidefinite matrix A.

We can further simplify this expression by introducing a partial order on Pauli strings $\mathbf{q},\mathbf{s} \in \{\mathbb{I}, X, Y, Z\}^n$. We write $\mathbf{q} \triangleright \mathbf{s}$ if it is possible to obtain $\mathbf{q}$ from $\mathbf{s}$ by replacing some local non-identity Paulis with $\mathbb{I}$. Moreover, let $|\mathbf{q}| = |\{i : \; q_i \neq \mathbb{I}\}|$ denote the number of non-identity Pauli's in the string $\mathbf{q}$. Then,

$$
\begin{aligned}
\left\| \sum_{\mathbf{p},\mathbf{q}} \alpha_{\mathbf{p}}\alpha_{\mathbf{q}} f(\mathbf{p},\mathbf{q}) \mathrm{tr} P_{\mathbf{p}} P_{\mathbf{q}} \right\|_\infty &= \left\| \tfrac{1}{3^k} \sum_{\mathbf{s}\in\{X,Y,Z\}^k} \left(\sum_{\mathbf{q}\triangleright\mathbf{s}} 3^{|\mathbf{q}|} \alpha_{\mathbf{q}} P_{\mathbf{q}} \right)^2 \right\|_\infty \qquad (4.108) \\
&\leq \frac{1}{3^k} \sum_{\mathbf{s}\in\{X,Y,Z\}^k} \left(\sum_{\mathbf{q}\triangleright\mathbf{s}} 3^{|\mathbf{q}|} \alpha_{\mathbf{q}} P_{\mathbf{q}} \right)^2 , \qquad (4.109)
\end{aligned}
$$

where we have used $\|P_{\mathbf{q}}\|_\infty = 1$ for all Pauli strings. Next, note that for fixed $\mathbf{s} \in \{X, Y, Z\}^k$,

$$
\sum_{\mathbf{q}\triangleright\mathbf{s}} 3^{|\mathbf{q}|} = 3^k + k3^{k-1} + \binom{k}{2} 3^{k-2} + \cdots + 1 = 4^k. \qquad (4.110)
$$

Together with Cauchy-Schwarz, this numerical insight implies

$$
\begin{aligned}
\tfrac{1}{3^k} \sum_{\mathbf{s}\in\{X,Y,Z\}^k} \left(\sum_{\mathbf{q}\triangleright\mathbf{s}} 3^{|\mathbf{q}|} |\alpha_{\mathbf{q}}| \right)^2 &\leq \tfrac{1}{3^k} \sum_{\mathbf{s}\in\{X,Y,Z\}^k} \left(\sum_{\mathbf{q}\triangleright\mathbf{s}} 3^{|\mathbf{q}|} \right) \left(\sum_{\mathbf{q}\triangleright\mathbf{s}} 3^{|\mathbf{q}|} |\alpha_{\mathbf{p}}^2 \right) \qquad (4.111) \\
&= 4^k \sum_{\mathbf{s}\in\{X,Y,Z\}} \sum_{\mathbf{q}\triangleright\mathbf{s}} 3^{|\mathbf{q}|-k} |\alpha_{\mathbf{q}}|^2. \qquad (4.112)
\end{aligned}
$$

Finally, observe that every $\mathbf{q} \in \{\mathbb{I}, X, Y, Z\}^k$ is dominated by exactly $3^{k-|\mathbf{q}|}$ different strings $\mathbf{s} \in \{X, Y, Z\}^k$. This ensures

$$4^k \sum_{\mathbf{s}\in\{X,Y,Z\}} 3^{|\mathbf{q}|-k}|\alpha_{\mathbf{q}}|^2 = 4^k \sum_{\mathbf{q}\in\{\mathbb{I},X,Y,Z\}} |\alpha_{\mathbf{q}}|^2 = 4^k 2^{-k}\|\tilde{O}\|_2^2, \tag{4.113}$$

because Pauli matrices are proportional to an orthonormal basis of $\mathbb{H}_2^{\otimes k}$: $\sum_{\mathbf{q}} |\alpha_{\mathbf{q}}|^2 = \sum_{\mathbf{q}} \left|2^{-k}\mathrm{tr}\left(\sigma_{\mathbf{q}}\tilde{O}\right)\right|^2 = 2^{-k}\|\tilde{O}\|_2^2$. The general claim then follows from the fundamental relation among Schatten norms: $\|\tilde{O}\|_2^2 \leq 2^k\|\tilde{O}\|_\infty^2 = 2^k\|O\|_\infty^2$.

The bound on traceless parts O_0 of observables is nearly analogous, because the transition from O to O_0 respects locality. E.g. $O = \tilde{O} \otimes \mathbb{I}^{\otimes(n-k)}$ obeys $O_0 = \tilde{O}_0 \otimes \mathbb{I}^{\otimes(n-k)}$. To get the same bound, we use that this transition is a contraction in Hilbert-Schmidt norm:

$$\|O_0\|_{\text{shadow}}^2 = \|\tilde{O}_0\|_{\text{shadow}}^2 \leq 4^k 2^{-k}\|\tilde{O}_0\|_2^2 \leq 4^k 2^{-k}\|\tilde{O}\|_2^2 \leq 4^k\|\tilde{O}\|_\infty^2 = \|O\|_\infty^2.$$

This concluded the proof. □

Proof of Lemma 10. Since Pauli observables decompose nicely into tensor products, this claim readily follows from extending a single-qubit argument. Note that $\mathcal{D}_{1/3}^{-1}(P_p) = 3P_p$ for $p \neq \mathbb{I}$ and $\mathcal{D}_{1/3}^{-1}(\mathbb{I}) = \mathbb{I}$. It is straightforward to evaluate the single-qubit expression for the trivial case $P_p = P_q = \mathbb{I}$. Fix a state σ and compute

$$\mathbb{E}_{U\sim\mathrm{Cl}(2)} \sum_{b\in\{0,1\}} \langle b|U\sigma U^\dagger|b\rangle\langle b|U\mathcal{D}_{1/3}^{-1}(\mathbb{I})U^\dagger|b\rangle^2 = \mathbb{E}_{U\sim\mathrm{Cl}(2)} \sum_{b\in\{0,1\}} \langle b|U\sigma U^\dagger|b\rangle \tag{4.114}$$

$$= \mathbb{E}_{U\sim\mathrm{Cl}(2)}\mathrm{tr}(\sigma) = \mathrm{tr}\left(\sigma\mathbb{I}^2\right). \tag{4.115}$$

Next, suppose $P_q = \mathbb{I}$, but $P_p \neq \mathbb{I}$. This single-qubit case is covered by Eq. (4.79):

$$\begin{aligned}&\mathbb{E}_{U\sim\mathrm{Cl}(2)} \sum_{b\in\{0,1\}} \langle b|U\sigma U^\dagger|b\rangle\langle b|U\mathcal{D}_{1/3}^{-1}(P_p)U^\dagger|b\rangle\langle b|U\mathcal{D}_{1/3}^{-1}\mathbb{I}U^\dagger|b\rangle \\ =&\mathrm{tr}\Big(\sigma \sum_{b\in\{0,1\}} U^\dagger|b\rangle\langle b|U\langle b|U3P_pU^\dagger|b\rangle\Big) = 3\mathrm{tr}\Big(\sigma \sum_{b\in\{0,1\}} \frac{1}{2}\mathcal{D}_{1/3}(P_p)\Big) = \mathrm{tr}\left(\sigma P_p\mathbb{I}\right),\end{aligned} \tag{4.116}$$

because $\mathcal{D}_{1/3}(P_p) = \frac{1}{3}P_p$. The case $P_p = \mathbb{I}$ and $P_q \neq \mathbb{I}$ leads to analogous results. Finally, suppose that both $P_p, P_q \neq \mathbb{I}$. By assumption $\mathcal{D}_{1/3}^{-1}(P_p)$, $\mathcal{D}_{1/3}^{-1}(P_q)$ and both matrices are traceless. Hence, we can resort to Eq. (4.81) to conclude

$$\mathbb{E}_{U\sim\mathrm{Cl}(2)^{\otimes n}} \sum_{b\in\{0,1\}^k} \langle b|U\sigma U^\dagger|b\rangle\langle b|U(\mathcal{D}_{1/3}^{-1})^{\otimes k}(P_p)U^\dagger|b\rangle\langle b|U(\mathcal{D}_{1/3}^{-1})^{\otimes k}(P_q)U^\dagger|b\rangle$$

$$=\mathrm{tr}\Big(\sigma \sum_{b\in\{0,1\}} U^\dagger|b\rangle\langle b|U\langle b|U3P_pU^\dagger|b\rangle\langle b|U3P_qU^\dagger|b\rangle\Big) \tag{4.117}$$

$$=9\mathrm{tr}\Big(\sigma \sum_{b\in\{0,1\}} \frac{\mathrm{tr}(P_pP_q)\mathbb{I}+P_pP_q+P_qP_p}{(2+2)(2+1)2}\Big) \tag{4.118}$$

for any state σ. Pauli matrices are orthogonal ($\mathrm{tr}(P_pP_q)=2\delta_{p,q}$) and anticommute ($P_pP_q+P_qP_p=2\delta_{p,q}$). This implies that the above expression vanishes whenever $p\neq q$. If $p=q$ it evaluates to $3\mathrm{tr}(\sigma P_pP_q)$ and we can conclude that the single qubit average always equals

$$f(p,q)\mathrm{tr}\left(\sigma P_pP_q\right) \quad \text{where} \quad f(p,q)=\begin{cases}1 & \text{if } p=\mathbb{I} \text{ or } q=\mathbb{I},\\ 3 & \text{if } p=q\neq\mathbb{I},\\ 0 & \text{else.}\end{cases} \tag{4.119}$$

The statement then follows from extending this formula to tensor products of k Pauli matrices. □

4.10 Additional computations and proofs for predicting nonlinear functions

We focus on the particularly relevant task of predicting quadratic functions with classical shadows, using

$$\hat{o}(N,1)=\frac{1}{N(N-1)}\sum_{j\neq l}\mathrm{tr}(O\hat{\rho}_i\otimes\hat{\rho}_j) \quad \text{to predict} \quad \mathrm{tr}\left(O\rho\otimes\rho\right)=\mathbb{E}\,\hat{o}(N,1). \tag{4.120}$$

General variance bound

Lemma 11 (Variance)**.** *The variance associated with the estimator $\hat{O}(N,1)$ obeys*

$$\begin{aligned}&\mathrm{Var}[\hat{o}(N,1)] && (4.121)\\ &=\binom{N}{2}^{-1}\Big(2(N-2)\,\mathrm{Var}[\mathrm{tr}(O_s\hat{\rho}_1\otimes\rho)]+\mathrm{Var}[\mathrm{tr}(O_s\hat{\rho}_1\otimes\hat{\rho}_2)]\Big)\\ &\leq\frac{4}{N^2}\,\mathrm{Var}[\mathrm{tr}(O\hat{\rho}_1\otimes\hat{\rho}_2)]+\frac{2}{N}\,\mathrm{Var}[\mathrm{tr}(O\hat{\rho}_1\otimes\rho)]+\frac{2}{N}\,\mathrm{Var}[\mathrm{tr}(O\rho\otimes\hat{\rho}_1)], && (4.122)\end{aligned}$$

where $O_s=(O+SOS)/2$ is the symmetrized version of O and S denotes the swap operator ($S|\psi\rangle\otimes|\phi\rangle=|\phi\rangle\otimes|\psi\rangle$).

Proof. First, note that $\hat{o}(N,1)$ and the target $\mathrm{tr}(O\rho\otimes\rho)$ are invariant under symmetrization. This ensures

$$\hat{o}(N,1)=\binom{N}{2}\sum_{i<j}\mathrm{tr}\left(O_s\hat{\otimes}\hat{\rho}_j\right) \quad \text{and moreover} \quad \mathrm{tr}\left(O\rho\otimes\rho\right)=\mathrm{tr}\left(O_s\rho\otimes\rho\right). \tag{4.123}$$

Thus, we may without loss replace the original observable O by its symmetrized version O_s. Next, we expand the definition of the variance:

$$\begin{aligned}
\mathrm{Var}[\hat{o}(N,1)] =& \mathbb{E}\left[(\hat{o}(N,1) - \mathrm{tr}(O_s\rho \otimes \rho))^2\right] \\
=& \binom{N}{2}^{-2} \sum_{i<j} \sum_{k<l} \left(\mathbb{E}\left[\mathrm{tr}(O_s\hat{\rho}_i \otimes \hat{\rho}_j)\, \mathrm{tr}(O_s\hat{\rho}_k \otimes \hat{\rho}_l) \right] - \mathrm{tr}(O_s\rho \otimes \rho)^2 \right) \\
=& \binom{N}{2}^{-2} \sum_{i<j} \mathbb{E}\left[\mathrm{tr}(O_s\hat{\rho}_i \otimes \hat{\rho}_j)^2 \right] - \mathrm{tr}(O_s\rho \otimes \rho)^2 \Big) \\
+& 2\binom{N}{2}^{-2} \sum_{i<j} \sum_{l\neq i,j} \left(\mathbb{E}\left[\mathrm{tr}(O_s\hat{\rho}_i \otimes \hat{\rho}_j)\, \mathrm{tr}(O_s\hat{\rho}_i \otimes \hat{\rho}_l) \right] - \mathrm{tr}(O_s\rho \otimes \rho)^2 \right) \\
=& \binom{N}{2}^{-1} \mathrm{Var}[\mathrm{tr}(O_s\hat{\rho}_1 \otimes \hat{\rho}_2)] + \binom{N}{2}^{-1} 2(N-2)\, \mathrm{Var}[\mathrm{tr}(O_s\hat{\rho}_1 \otimes \rho)].
\end{aligned} \tag{4.124}$$

We can use the inequality $\mathrm{Var}[(A+B)/2] \leq (\mathrm{Var}[A] + \mathrm{Var}[B])/2$ (for any pair of random variables A, B) to obtain a simplified upper bound:

$$\mathrm{Var}[\hat{o}(N,1)] \tag{4.125}$$

$$\begin{aligned}
&= \binom{N}{2}^{-1} \mathrm{Var}[\mathrm{tr}(O_s\hat{\rho}_1 \otimes \hat{\rho}_2)] + \binom{N}{2}^{-1} 2(N-2)\, \mathrm{Var}[\mathrm{tr}(O_s\hat{\rho}_1 \otimes \rho)] \\
&\leq \frac{4}{N^2} \mathrm{Var}[\mathrm{tr}(O_s\hat{\rho}_1 \otimes \hat{\rho}_2)] + \frac{4}{N} \mathrm{Var}[\mathrm{tr}(O_s\hat{\rho}_1 \otimes \rho)] \\
&\leq \frac{4}{N^2} \mathrm{Var}[\mathrm{tr}(O\hat{\rho}_1 \otimes \hat{\rho}_2)] + \frac{2}{N} \mathrm{Var}[\mathrm{tr}(O\hat{\rho}_1 \otimes \rho)] + \frac{2}{N} \mathrm{Var}[\mathrm{tr}(O\rho \otimes \hat{\rho}_1)].
\end{aligned} \tag{4.126}$$

□

Concrete variance bounds for random Pauli measurements

Proposition 6. *Suppose that O describes a quadratic function $\mathrm{tr}(O\rho \otimes \rho)$ that acts on at most k-qubits in the first system and at most k-qubits in the second system and obeys $\|O\|_\infty \geq 1$. Then,*

$$\max\left(\mathrm{Var}[\mathrm{tr}(O\rho \otimes \hat{\rho}_1)], \mathrm{Var}[\mathrm{tr}(O\hat{\rho}_1 \otimes \rho)], \sqrt{\mathrm{Var}[\mathrm{tr}(O\hat{\rho}_1 \otimes \hat{\rho}_2)]} \right) \leq 4^k \|O\|_\infty^2 . \tag{4.127}$$

Proof. Because of the single-qubit tensor product structure in the random Pauli measurement and the inverted quantum channel $\mathcal{M}_P^{-1}$, the tensor product of two snapshots $\hat{\rho}_1 \otimes \hat{\rho}_2$ of the unknown quantum state ρ may be viewed as a single snapshot of the tensor product state $\rho \otimes \rho$:

$$\begin{aligned}\hat{\rho}_1 \otimes \hat{\rho}_2 &= \bigotimes_{i=1}^{n} \left(\mathcal{M}_1^{-1}(U_1^{(i)}|b_1^{(i)}\rangle\langle b_1^{(i)}|(U_1^{(i)})^\dagger) \right) \bigotimes_{i=1}^{n} \left(\mathcal{M}_1^{-1}(U_2^{(i)}|b_2^{(i)}\rangle\langle b_2^{(i)}|(U_2^{(i)})^\dagger) \right) \\ &= \bigotimes_{i=1}^{2n} \mathcal{M}_1^{-1}(U^{(i)}|b^{(i)}\rangle\langle b^{(i)}|(U^{(i)})^\dagger) =: \hat{\rho}. \end{aligned} \tag{4.128}$$

Hence $\mathrm{tr}(O\hat{\rho}_1 \otimes \hat{\rho}_2) = \mathrm{tr}(O\hat{\rho})$ and, by assumption, O is an observable that acts on $k + k = 2k$ qubits only. The claim then follows from invoking the variance bounds for linear feature prediction presented in Proposition 5. □

Concrete variance bounds for random Clifford measurements

In contrast to the Pauli basis setup, variances for quadratic feature prediction with Clifford basis measurements cannot be directly reduced to its linear counterpart. Nonetheless, a more involved direct analysis does produces bounds that do closely resemble the linear base case.

Proposition 7. *Suppose that O describes a quadratic function* $\mathrm{tr}(O\rho \otimes \rho)$ *and obeys* $\mathrm{tr}(O^2) \geq 1$. *Then, the variance associated with classical shadow estimation (random Clifford measurements) obeys*

$$\max \Big(\mathrm{Var}[\mathrm{tr}(O\rho \otimes \hat{\rho}_1)], \mathrm{Var}[\mathrm{tr}(O\hat{\rho}_1 \otimes \rho)], \tag{4.129}$$

$$\sqrt{\mathrm{Var}[\mathrm{tr}(O\hat{\rho}_1 \otimes \hat{\rho}_2)]} \Big) \leq \sqrt{9 + 6/2^n}\, \mathrm{tr}(O^2). \tag{4.130}$$

The pre-factor $\sqrt{9+6/2^n}$ converges to the constant 3 at an exponential rate in system size.

This claim is based on the following technical Lemma and insights regarding linear feature prediction.

Lemma 12. *Suppose that O describes a quadratic function* $\mathrm{tr}(O\rho \otimes \rho)$. *Then,*

$$\mathrm{Var}[\mathrm{tr}(O\hat{\rho}_1 \otimes \hat{\rho}_2)] \leq 9\, \mathrm{tr}(O^2) + \frac{6}{2^n} \|O\|_\infty^2. \tag{4.131}$$

Proof of Proposition 7. The variance of $\mathrm{tr}(O\rho \otimes \hat{\rho}_1)$ is equivalent to the variance of $\mathrm{tr}(\tilde{O}_\rho \hat{\rho})$, where $\tilde{O}_\rho = \mathrm{tr}_1 \left(\rho \otimes \mathbb{I} O \right)$ describes a linear function. According to Proposition 3, this variance term obeys

$$\mathrm{Var}\left[\mathrm{tr}\left(O\rho \otimes \hat{\rho} \right) \right] = \mathrm{Var}\left[\mathrm{tr}\left(\tilde{O}_\rho \hat{\rho}_1 \right) \right] \leq 3\mathrm{tr}\left(\tilde{O}_\rho^2 \right) = \mathrm{tr}\left(\mathrm{tr}_1\left(\rho \otimes \mathbb{I} O \right)^2 \right) \leq 3\mathrm{tr}(O^2), \tag{4.132}$$

because $\mathrm{tr}(\rho) = 1$ and $\mathrm{tr}(\rho^2) \leq 1$. A similar argument takes care of the second variance contribution $\mathrm{Var}\left[\mathrm{tr}\left(O\hat{\rho}_1 \otimes \rho\right)\right]$. Lemma 12 supplies a bound for the square of the final contribution. By assumption $\sqrt{\mathrm{tr}(O^2)} \leq \mathrm{tr}(O^2)$ and the claim follows. □

The remainder of this section is devoted to proving Lemma 12. Unfortunately, there does not seem to be a direct way to relate this task to variance bounds for linear feature prediction. Instead, we base our analysis on the 3-design property (4.81) of Clifford circuits and a reformulation of this feature in terms of permutation operators. This strategy is inspired by the approach developed in (Brandão et al., 2019), but conceptually and technically somewhat simpler. We believe that similar arguments extend to variances associated with higher order polynomials, but do refrain from a detailed analysis. Instead, we carefully outline the main ideas and leave a rigorous extension to future work.

Problem statement and reformulation: We will ignore symmetrization (which can only make the variance smaller) and focus on bounding the variance of

$$\mathrm{tr}\left(O\hat{\rho}_1 \otimes \hat{\rho}_2\right), \tag{4.133}$$

where each $\hat{\rho}_i$ is an independent classical shadow. To simplify notation, we set $d = 2^n$ and define the following traceless variants of O:

$$\begin{aligned} O_0^{(1)} =&\mathrm{tr}_2(O) - \frac{\mathrm{tr}\left(O\right)}{d}\mathbb{I}, \quad \text{and} \quad O_0^{(2)} = \mathrm{tr}_1(O) - \frac{\mathrm{tr}(O)}{d}\mathbb{I}, \quad \text{as well as} \\ O_0^{(1,2)} =&O - \mathrm{tr}_2(O) \otimes \frac{\mathbb{I}}{d} - \frac{\mathbb{I}}{d} \otimes \mathrm{tr}_1(O) + \mathrm{tr}(O)\frac{\mathbb{I}}{d} \otimes \frac{\mathbb{I}}{d}. \end{aligned} \tag{4.134}$$

Here, $\mathrm{tr}_a(O)$ with $a = 1, 2$ denotes the partial trace over the first and second system, respectively. All three operators are traceless (recall $\mathrm{tr}\left(\mathrm{tr}_a(O)\right) = \mathrm{tr}(O)$) and the final (bipartite) operator has the additional property that both partial traces vanish identically: $\mathrm{tr}_a\big(O_0^{(1,2)}\big) = 0$.

Proposition 3 asserts $\hat{\rho}_a = (d+1)U_a^\dagger |\hat{b}_a\rangle\langle\hat{b}_a| U_a - \mathbb{I}$, where each $U_a \in \mathrm{Cl}(d)$ is a random Clifford unitary and $\hat{b}_a \in \{0,1\}^n$ is the outcome of a computational basis measurement. These explicit formulas allow us to decompose the expression of interest in the following fashion:

$$\mathrm{tr}\left(O\hat{\rho}_1 \otimes \hat{\rho}_2\right) =(d+1)^2 \mathrm{tr}\left(O_0^{(1,2)} U_1^\dagger |\hat{b}_1\rangle\langle\hat{b}_1| U_1 \otimes U_2^\dagger |\hat{b}_1\rangle\langle\hat{b}_2| U_2\right) + \frac{\mathrm{tr}(O)^2}{d^2}$$

$$+\frac{d+1}{d}\mathrm{tr}\left(O_0^{(1)}U_1^\dagger|\hat{b}_1\rangle\langle\hat{b}_1|U_1\right)+\frac{d+1}{d}\mathrm{tr}\left(O_0^{(2)}U_2^\dagger|\hat{b}_2\rangle\langle\hat{b}_2|U_2\right). \tag{4.135}$$

The variance corresponds to the expected square of this expression. The second term is constant and does not contribute. We analyze the remaining terms on a case-by case basis.

Linear terms: The third and fourth terms in Eq. (4.135) are linear feature functions in one classical shadow only. Their (squared) contribution to the overall variance is characterized by Proposition 3:

$$\mathbb{E}\left[\left(\frac{d+1}{d}\mathrm{tr}\left(O_0^{(a)}U_a^\dagger|\hat{b}_a\rangle\langle\hat{b}_a|U_a\right)\right)^2\right]\leq\frac{3}{d^2}\left\|O_0^{(a)}\right\|_2^2\quad\text{for } a=1,2. \tag{4.136}$$

Both bounds can be related to the Hilbert-Schmidt norm (squared) of the original observable:

$$\frac{3}{d^2}\left\|O_0^{(a)}\right\|_2^2\leq\frac{3}{d^2}\|\mathrm{tr}_a(O)\|_2^2\leq 3\|O\|_2^2=3\mathrm{tr}\left(O^2\right). \tag{4.137}$$

Leading-order term: We need to bound

$$\mathbb{E}\big[(d+1)^4\mathrm{tr}\left(O_0^{(1,2)}U_1^\dagger|\hat{b}_1\rangle\langle\hat{b}_1|U_1\otimes U_2^\dagger|\hat{b}_2\rangle\langle\hat{b}_2|U_2\right)^2\big], \tag{4.138}$$

where $O_0^{(1,2)}$ has the special property that both partial traces vanish identically: $\mathrm{tr}_a\big(O_0^{(1,2)}\big)=0$ for $a=1,2$. Moreover, the Hilbert-Schmidt norm (squared) of this operator factorizes nicely:

$$\left\|O_0^{(1,2)}\right\|_2^2=\|O\|_2^2-\frac{1}{d}\|O_0^{(1)}\|_2^2-\|O_0^{(2)}\|_2^2-\frac{\mathrm{tr}(O)^2}{d^2}. \tag{4.139}$$

Not only is this expression bounded by the original Hilbert-Schmidt norm $\|O\|_2^2$. The norms of partial traces also feature explicitly with a minus sign. This will allow us to fully counter-balance the variance contributions (4.137) from the linear terms.

Next, we use the 3-design property (4.77) of Clifford circuits in dimension $d=2^n$:

$$\mathbb{E}_{U_a\sim\mathrm{Cl}(d)}\left[\left(U_a^\dagger|b_a\rangle\langle b_a|U_a\right)^{\otimes 3}\right]=\binom{d+2}{3}^{-1}P_{\vee^3}, \tag{4.140}$$

where $P_{\vee^3}$ is the projector onto the totally symmetric subspace of $\mathbb{C}^d\otimes\mathbb{C}^d\otimes\mathbb{C}^d$. This formula implies

$$\mathbb{E}\left[(d+1)^4\mathrm{tr}\left(O_0^{(1,2)}U_1^\dagger|\hat{b}_1\rangle\langle\hat{b}_1|U_1\otimes U_2^\dagger|\hat{b}_2\rangle\langle\hat{b}_2|U_2\right)^2\right] \tag{4.141}$$

$$\leq \mathrm{tr}\left(O_0^{(1,2)} \otimes O_0^{(1,2)} \otimes \rho \otimes \rho \; P_{\vee^3}^{(\mathrm{odd})} \otimes P_{\vee^3}^{(\mathrm{even})}\right), \tag{4.142}$$

where the superscripts "even" and "odd" indicate on which subset of tensor factors the projectors act.

Next, we exploit the fact that symmetric projectors can be decomposed into permutation operators: $(3!)P_{\vee^3} = \sum_{\pi \in S_3} W_\pi$, where S_3 is the group of all six permutations of three elements and the permutation operators act like $W_\pi |\psi_1\rangle \otimes |\psi_2\rangle \otimes |\psi_3\rangle = |\psi_{\pi^{-1}(1)}\rangle \otimes |\psi_{\pi^{-1}(2)}\rangle \otimes |\psi_{\pi^{-1}(3)}\rangle$:

$$\mathrm{tr}\left(O_0^{(1,2)} \otimes O_0^{(1,2)} \otimes \rho \otimes \rho \; P_{\vee^3}^{(\mathrm{odd})} \otimes P_{\vee^3}^{(\mathrm{even})}\right) \tag{4.143}$$

$$= \sum_{\pi,\tau \in S_3} \mathrm{tr}\left(O_0^{(1,2)} \otimes O_0^{(1,2)} \otimes \rho \otimes \rho \; W_\pi^{(\mathrm{odd})} \otimes W_\tau^{(\mathrm{even})}\right). \tag{4.144}$$

The specific structure of $O_0^{(1,2)}$ implies that several contributions must vanish. Permutations that have either 1 or 2 as a fix-point lead to a partial trace of $O_0^{(1,2)}$ that evaluates to zero. There are only three permutations that do not have such fix-points: The flip $(1,2,3) \mapsto (2,1,3)$ and the two cycles $(1,2,3) \mapsto (3,1,2)$, $(1,2,3) \mapsto (2,3,1)$. There are in total $9 = 3^2$ potential combinations of such permutations. Each of them results in a trace expression that can be upper-bounded by Hilbert-Schmidt norms. For instance the pair flip and flip produces

$$\mathrm{tr}\left(O_0^{(1,2)} O_0^{(1,2)}\right) \mathrm{tr}(\rho)^2 = \left\| O_0^{(1,2)} \right\|_2^2. \tag{4.145}$$

All other 8 contributions can also be bounded by this expression and we conclude

$$\mathbb{E}\left[(d+1)^4 \mathrm{tr}\left(O_0^{(1,2)} U_1^\dagger |\hat{b}_1\rangle\langle\hat{b}_1| U_1 \otimes U_2^\dagger |\hat{b}_2\rangle\langle\hat{b}_2| U_2\right)^2\right] \leq 9 \left\| O_0^{(1,2)} \right\|_2^2 \tag{4.146}$$

Bounds on cross-terms: Cross-terms are considerably easier to evaluate, because one (or both) random matrices only feature linearly. We can use $\mathbb{E}\left[U_a^\dagger |\hat{b}_a\rangle\langle\hat{b}_a| U_a\right] = \mathcal{D}_{1/(d+1)}(\rho) = \frac{\rho + \mathbb{I}}{d+1}$ to effectively get rid of the linear contribution. For instance,

$$\left(\frac{d+1}{d}\right)^2 \mathbb{E}\left[\prod_{a=1,2} \mathrm{tr}\left(O_0^{(1)} U_a^\dagger |\hat{b}_a\rangle\langle\hat{b}_a| U_a\right)\right] \tag{4.147}$$

$$= \frac{1}{d^2} \mathrm{tr}\left(O_0^{(1)} \rho\right) \mathrm{tr}\left(O_0^{(2)} \rho\right) \leq \frac{1}{2d^2}\left(\|O_0^{(1)}\|_\infty^2 + \|O_0^{(2)}\|_\infty^2\right), \tag{4.148}$$

where $\|\cdot\|_\infty$ denotes the operator norm. Cross terms that do feature the leading order term require slightly more work but can be addressed in a similar fashion. Using linearity in one snapshot reduces the expression to an expectation of a quadratic function in one snapshot only. The remaining computation is similar to the proof of Proposition 3 and yields

$$\frac{(d+1)^3}{d}\mathbb{E}\left[\mathrm{tr}\left(O_0^{(1,2)}U_1^\dagger|\hat{b}_1\rangle\langle\hat{b}_1|U_1\otimes U_2^\dagger|\hat{b}_2\rangle\langle\hat{b}_2|U_2\right)\mathrm{tr}\left(O_0^{(a)}U_a^\dagger|\hat{b}_a\rangle\langle\hat{b}_a|U_a\right)\right] \tag{4.149}$$

$$\leq \frac{3}{2d^2}\left(\|\tilde{O}_\rho^{(a)}\|_2^2+\|O_0^{(a)}\|_2^2\right), \tag{4.150}$$

for $a=1,2$, as well as $\tilde{O}_\rho^{(1)}=\mathrm{tr}_2\left(\mathbb{I}\otimes\rho O\right)$ and $\tilde{O}_\rho^{(2)}=\mathrm{tr}_1\left(\rho\otimes\mathbb{I}O\right)$, respectively.

Full variance bound: We are now ready to combine all individual bounds to control the full variance:

$$\mathrm{Var}\left[\hat{o}\right] \tag{4.151}$$

$$\begin{aligned}
&\leq\mathbb{E}\Bigg((d+1)^2\mathrm{tr}\left(O_0^{(1,2)}U_1^\dagger|\hat{b}_1\rangle\langle\hat{b}_1|U_1\otimes U_2^\dagger|\hat{b}_2\rangle\langle\hat{b}_2|U_2\right)\\
&\qquad+\sum_{a=1,2}\frac{d+1}{d}\mathrm{tr}\left(O_0^{(a)}U_a^\dagger|\hat{b}_a\rangle\langle\hat{b}_a|U_a\right)\Bigg)^2\\
&\leq 9\|O_0^{(1,2)}\|_2^2+\frac{6}{2d^2}\left(\|\mathrm{tr}_2\left(\mathbb{I}\otimes\rho O\right)\|_2^2+\|O_0^{(1)}\|_2^2\right)+\frac{6}{2d^2}\left(\|\mathrm{tr}_1\left(\rho\otimes\mathbb{I}O\right)\|_2^2\right)\\
&+\frac{3}{d^2}\|O_0^{(1)}\|_2^2+\frac{3}{d^2}\|O_0^{(2)}\|_2^2+\frac{1}{2d^2}\left(\|O_0^{(1)}\|_\infty^2+\|O_0^{(2)}\|_\infty^2\right).
\end{aligned} \tag{4.152}$$

Standard norm inequalities, as well as the explicit expression for $\|O_0^{(1,2)}\|_2^2$, allow for counter-balancing some of the sub-leading terms, and we conclude

$$\mathrm{Var}\left[\hat{o}\right]\leq 9\|O_0\|_2^2+\frac{3}{d^2}\left(\|\mathrm{tr}_2\left(\mathbb{I}\otimes\rho O\right)\|_2^2+\|\mathrm{tr}_1\left(\rho\otimes\mathbb{I}O\right)\|_2^2\right)\leq 9\|O_0\|_2^2+\frac{6}{d}\|O\|_\infty^2. \tag{4.153}$$

4.11 Information-theoretic lower bound with scaling in Frobenius norm

Before stating the content of the statement, we need to introduce some additional notation. In quantum mechanics, the most general notion of a quantum measurement is a POVM (positive operator-valued measure). A d-dimensional POVM F consists of a collection $F_1,\ldots,F_N$ of positive semidefinite matrices that sum up to the identity matrix: $\langle x|F_i|x\rangle\geq 0$ for all $x\in\mathbb{C}^d$ and $\sum_i F_i=\mathbb{I}$. The index i is associated with different potential measurement outcomes and Born's rule asserts $\mathrm{Pr}\left[i|\rho\right]=\mathrm{tr}(F_i\rho)$ for all $1\leq i\leq M$ and any d-dimensional quantum state ρ. We present a simplified version of the proof by consider the relevant case where $M\leq\exp(2^n/32)$. The full proof can be found in (Huang and Richard Kueng, 2019).

Detailed statement and proof idea

Theorem 15 (Detailed restatement of Theorem 8 for Hilbert-Schmidt norm)**.** *Fix a sequence of POVMs $F^{(1)}, \ldots, F^{(N)}$. Suppose that given any M features $0 \leq O_1, O_2, \ldots, O_M \leq I$ with $\max_i \left(\|O_i\|_2^2 \right) \leq B$, there exists a machine (with arbitrary runtime as long as it always terminates) that can use the measurement outcomes of $F^{(1)}, \ldots, F^{(N)}$ on N copies of an unknown d-dimensional quantum state ρ to ϵ-accurately predict $\mathrm{tr}(O_1\rho), \ldots, \mathrm{tr}(O_M\rho)$ with high probability. Assuming $M \leq \exp(d/32)$, then necessarily*

$$N \geq \Omega\left(\frac{B \log(M)}{\epsilon^2}\right). \tag{4.154}$$

It is worthwhile to put this statement into context and discuss consequences, as well as limitations. Theorem 7 (Clifford measurements) equips classical shadows with a *universal* convergence guarantee: (order) $\log(M) \max_i \mathrm{tr}(O_i^2)/\epsilon^2$ single-copy measurements suffice to accurately predict *any* collection of M target functions in *any* state. Theorem 15 implies that there are cases where this number of measurements is unavoidable. This highlights that the sample complexity of feature prediction with classical shadows is optimal in the worst case – a feature also known as minimax optimality.

Minimax optimality, however, does not rule out potential for further improvement in certain best-case scenarios. Advantageous structure in ρ or the O_i's (or both) can facilitate the design of more efficient prediction techniques. Prominent examples include matrix product state tomography (MPST) (Cramer et al., 2010; Lanyon et al., 2017) and neural network tomography (NNQST) (Carrasquilla, Torlai, et al., 2019). Such tailored approaches, however, hinge on additional assumptions about the states to be measured or the properties to be predicted.[6]

Finally, we emphasize that Theorem 8 only applies to single-copy measurements. Another way to bypass this lower bound is to use joint quantum measurements that act on all copies of the quantum state ρ simultaneously. Although very challenging to implement, such procedures can get by with substantially fewer state copies while still being universal. Shadow tomography (Aaronson, 2018; Aaronson and Rothblum, 2019) is a prominent example.

[6] Although tractable in theory, MPST becomes prohibitively expensive if ρ is not well-approximated by a MPS with small bond dimension. Likewise, NNQST seems to struggle to identify quantum states with intricate combinatorial structures, such as toric code ground states. We refer to the other supplementary sections for numerical (Section 4.6) and theoretical (Section 4.8) support of this claim.

Proof idea: We adapt a versatile proof technique for establishing information-theoretic lower bounds on tomographic procedures that is originally due to Flammia *et al.* (Steven T Flammia et al., 2012); see also (Haah et al., 2017; Roth et al., 2018) for adaptations and refinements. The key idea is to consider a communication task in which Alice chooses a quantum state from among an alphabet of possible states and then sends copies of her chosen state to Bob, who measures all the copies hoping to extract a classical message from Alice. If we choose Alice's alphabet suitably, then by learning many properties of Alice's state, Bob will be able to identify the state, hence decoding Alice's message. Information-theoretical lower bounds on the number of copies Bob needs to decode the message can therefore be translated into lower bounds on how many copies Bob needs to learn the properties.

To be more specific, suppose Alice chooses her state from an ensemble of M possible n-qubit signal states $\{\rho_1, \rho_2, \ldots \rho_M\}$ and suppose there are M linear operators $\{O_1, O_2, \ldots O_M\}$, each with $\mathrm{tr}\left(O_i^2\right) \leq B$, such that learning the expectation values of all the operators $\{O_i\}$ up to an additive error ϵ suffices to determine ρ_i uniquely. Suppose furthermore that if Bob receives N copies of *any* n-qubit state, and measures them one at a time, he is able to learn all of the properties $\{O_i\}$ with an additive error no larger than ϵ with high success probability. This provides Bob with a method for identifying the state ρ_i with high probability. Therefore, if Alice chooses her signal state uniformly at random from among the M possible states, by performing the appropriate single-copy measurements, Bob can acquire $\log_2 M$ bits of information about Alice's message. A lower bound on how many copies Bob needs to gain $\log_2 M$ bits of information about Alice's state, then, becomes a lower bound on how many copies Bob needs to learn the M properties $\{O_i\}$. To get the best possible lower bound, we choose Alice's signal ensemble $\{\rho_i\}$ so that it is as hard as possible for Bob to distinguish the signals using properties with $\mathrm{tr}\left(O_i^2\right) \leq B$.

So far, this lower bound on N would apply even if Bob has complete knowledge of Alice's signal states and the properties he should learn to distinguish them. We can derive a stronger lower bound on N by invoking a powerful feature of classical shadows — that Bob must make his measurements *before* he finds out which properties he must learn. To obtain this stronger bound, we introduce into the communication scenario a third party, named Loki[7], who tampers with the signal states. Loki chooses a Haar-random n-qubit unitary U, and replaces all N copies

[7] In Norse mythology, Loki is infamous for mischief and trickery. However, not entirely malicious, he often shows up in the nick of time to remedy the dire consequences of his actions.

of Alice's signal state ρ_i by the rotated states $U\rho_i U^\dagger$ before presenting the states to Bob (Loki's mischief).

If Bob knew Loki's unitary U, he could modify his measurement procedure to learn the rotated properties $\{UO_iU^\dagger\}$. These rotated properties are just as effective for distinguishing the rotated states as the unrotated properties were effective for distinguishing the unrotated states. However, Loki keeps U secret, so Bob is forced to perform his measurements on the rotated states without knowing U. Only after Bob's data acquisition phase is completed does Loki confide in Bob and provide him with a full classical description of the unitary he applied earlier (Loki's redemption). This three-party scenario is illustrated in Supplementary Figure 4.10.

Suppose, though, that using the classical shadow based on his measurements, Bob can predict *any* M properties (with additive error bounded by ϵ and with high success probability), provided that the Hilbert-Schmidt norm is no larger than $\sqrt{B}$ for each property. Then he is just as well equipped to learn $\{UO_iU^\dagger\}$ as $\{O_i\}$, and can therefore decode Alice's message successfully once Loki reveals U. It must be, then, that Bob's measurement outcomes provide $\log_2 M$ bits of information about Alice's prepared state, when U is known. This is the idea we use to derive the stronger upper bound on N, and hence prove Theorem 15.

We emphasize again that quantum feature prediction with classical shadows can cope with Loki's mischief, by merely rotating the features Bob predicts, because the predicted features need not be known at the time Bob measures. The lower bound in Theorem 15 does not apply to the task of learning features that are already known in advance. We also emphasize again that Theorem 15 assumes that the copies of the state are measured individually. It does not apply to protocols where collective measurements are applied across many copies.

Description of the communication protocol

We show how Alice can communicate any integer in $\{1, \ldots, M\}$ to Bob. Alice and Bob first agree on a codebook for encoding any integer selected from $\{1, \ldots, M\}$ in a d-dimensional quantum state. We denote these codebook states by $\rho_1, \ldots, \rho_M$. Alice and Bob also agree on a set of linear features $O_1, \ldots, O_M$ that satisfies

$$\mathrm{tr}(O_i\rho_i) \geq \max_{j\neq i} \mathrm{tr}(O_j\rho_i) + 3\epsilon. \tag{4.155}$$

Therefore, if each feature can be predicted with additive error ϵ, these features can be used to identify the state ρ_i. The communication protocol between Alice and

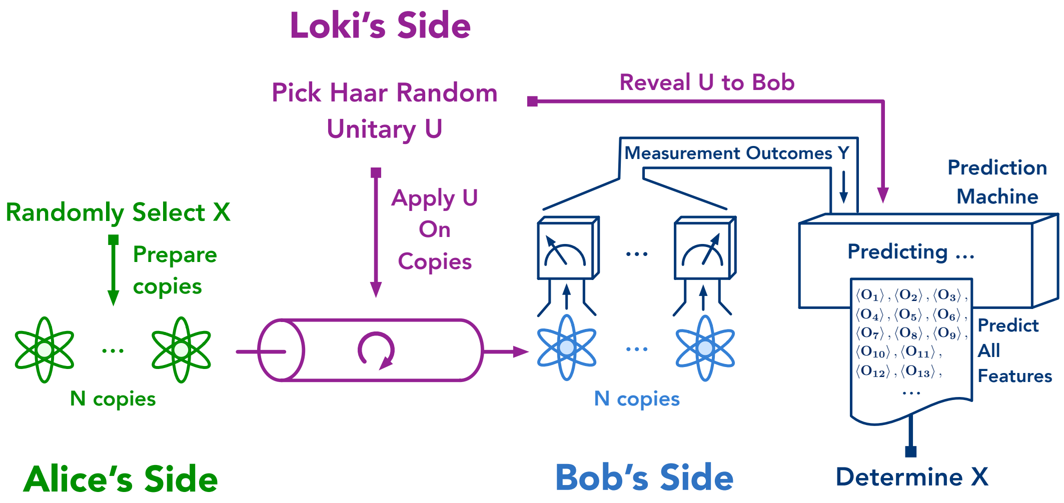


Figure 4.10: *Illustration of the communication protocol behind Theorem 15 and Theorem 16.* Two parties (Alice and Bob) devise a protocol that allows them to communicate classical bit strings: Alice encodes a bit string X in a quantum state and sends N independent copies of the state to Bob. Bob performs quantum measurements and uses a black box device (e.g. classical shadows) to decode Alice's original message. An unpredictable trickster (Loki) tampers with this procedure by randomly rotating Alice's quantum states en route to Bob. Loki reveals his actions only after Bob has completed the measurement stage of his protocol.

Bob is now apparent:

1. Alice randomly selects an integer X from $\{1, \ldots, M\}$.
2. Alice prepares N copies of the code-state ρ_X associated to X and sends them to Bob.
3. Bob performs POVMs $F^{(i)}$ on individual states and receives a string of measurement outcomes Y.
4. Bob inputs Y into the feature prediction machine to estimate $\mathrm{tr}(O_i \rho_X), \forall i = 1, \ldots, M$.
5. Bob finds $\overline{X}$ that has the largest $\mathrm{tr}(O_{\overline{X}} \rho_X)$.

The working assumption is that the feature prediction machine can estimate

$$\mathrm{tr}(O_1 \rho_X), \ldots, \mathrm{tr}(O_M \rho_X) \tag{4.156}$$

within ϵ-error and high success probability. This in turn ensures that this plain communication protocol is mostly successful, i.e. $\overline{X} = X$ with high probability. In words: Alice can transmit information to Bob, when no adversary is present.

We now show how they can still communicate safely in the presence of an adversary (Loki) who randomly rotates the transmitted code states en route: $\rho_X \mapsto U\rho_X U^\dagger$ and U is a Haar-random unitary.

This random rotation affects the measurement outcome statistics associated with the fixed POVMs $F^{(1)}, \ldots, F^{(N)}$. Each element of $Y = \left[Y^{(1)}, \ldots, Y^{(N)}\right]$ is now a random variable that depends on both X and U. After Bob has performed the quantum measurements to obtain Y, the adversary confesses to Bob and reveals the random unitary U. While Bob no longer has any copies of ρ_X, he can still incorporate precise knowledge of U by instructing the machine to predict linear features $UO_1U^\dagger, \ldots, UO_MU^\dagger$, instead of the original $O_1, \ldots, O_M$. This reverses the effect of the original unitary transformation, because $\mathrm{tr}(UO_iU^\dagger U\rho_X U^\dagger) = \mathrm{tr}(O_i\rho_X)$. This modification renders the original communication protocol stable with respect to Loki's actions. Alice can still send any integer in $\{1, \ldots, M\}$ to Bob with high probability.

Information-theoretic analysis

The following arguments use properties of Shannon entropy and mutual information, which can be found in standard textbooks on information theory, such as (Cover and Thomas, 2006).

The communication protocol is guaranteed to work with high probability, ensuring that Bob's recovered message $\bar{X}$ equals Alice's input X with high probability. Moreover, we assume that Alice selects her message uniformly at random. Fano's inequality then implies

$$I(X : \overline{X}) = H(X) - H(X|\overline{X}) \geq \Omega(\log(M)), \tag{4.157}$$

where $I(X : \overline{X})$ is the mutual information, and $H(X)$ is the Shannon entropy. By assumption, Loki chooses the unitary roatation U uniformly at random, regardless of the message X. This implies $I(X : U) = 0$ and, in turn

$$I(X : \overline{X}) \leq I(X : \overline{X}, U) = I(X : U) + I(X : \overline{X}|U) = I(X : \overline{X}|U). \tag{4.158}$$

For fixed U, $\overline{X}$ is the output of the machine that only takes into account the measurement outcomes Y. The data processing inequality then yields

$$I(X : Y|U) \geq I(X : \overline{X}|U) \geq I(X : \overline{X}) \geq \Omega(\log(M)). \tag{4.159}$$

Recall that Y is the measurement outcome of the N POVMs $F_1, \dots, F_N$. We denote the measurement outcome of F_k as Y_k. Because $Y_1, \dots, Y_N$ are random variables that depend on X and U,

$$\begin{aligned} I(X:Y|U) &= H(Y_1, \dots, Y_N|U) - H(Y_1, \dots, Y_N|X, U) \\ &\leq H(Y_1|U) + \dots + H(Y_N|U) - H(Y_1, \dots, Y_N|X, U) \\ &= \sum_{k=1}^{N} \Big(H(Y_k|U) - H(Y_k|X, U) \Big) = \sum_{k=1}^{N} I(X : F_k \text{ on } U\rho_X U^\dagger|U). \end{aligned} \tag{4.160}$$

The second to last equality uses the fact that when X, U are fixed, $Y_1, \dots, Y_N$ are independent. To obtain the best lower bound, we should choose Alice's signal states $\{\rho_i\}$ such that $I(X : F_k \text{ on } U\rho_X U^\dagger|U)$ is as small as possible. In Sec. 4.11, we will see that, no matter how Bob chooses his measurements $\{F_1, F_2, \dots, F_N\}$, there are signal states satisfying (4.155) such that

$$I(X : F_k \text{ on } U\rho_X U^\dagger|U) \leq \frac{36\epsilon^2}{B}, \forall k. \tag{4.161}$$

Assuming that this relation holds, we have established a connection between M and N: $\Omega(\log(M)) \leq I(X : Y|U) \leq 36N\epsilon^2/B$ and, therefore, $N \geq \Omega\Big(B\log(M)/\epsilon^2\Big)$. This establishes the claim in Theorem 15.

Detailed construction of quantum encoding and linear prediction decoding

We now construct a codebook $\rho_1, \dots, \rho_M$ and linear features $0 \leq O_1, O_2, \dots, O_M \leq \mathbb{I}$ with $\max_i \|O_i\|_2^2 \leq B$ that obey two key properties:

1. the code states $\rho_1, \dots, \rho_M$ obey the requirement displayed in Eq. (4.161).
2. the linear features $O_1, \dots, O_M$ are capable of identifying a unique code state:

$$\mathrm{tr}(O_i\rho_i) \geq \max_{j\neq i} \mathrm{tr}(O_j\rho_i) + 3\epsilon \quad \text{for all} \quad 1 \leq i \leq M. \tag{4.162}$$

The second condition requires each ρ_i to be distinguishable from $\rho_1, \dots, \rho_M$ via linear features O_i. The first condition, on the contrary, requires ρ_X to convey as little information about X as possible. The general idea would then be to create distinguishable quantum states that are, at the same time, very similar to each other.

In order to achieve these two goals, we choose M rank-$B/4$ subspace projectors $\Pi_1, \dots, \Pi_M$ that obey $\mathrm{tr}(\Pi_i\Pi_j)/r < 1/2$ for all $i \neq j$. The probabilistic method asserts that such a projector configuration exists; see Lemma 13 below. Now, we set

$$\rho_i = (1 - 3\epsilon)\frac{\mathbb{I}}{d} + 3\epsilon\frac{4\Pi_i}{B}, \quad \text{and} \quad O_i = 2\Pi_i, \quad \text{for all} \quad 1 \leq i \leq M. \tag{4.163}$$

It is easy to check that this construction meets the requirement in Eq. (4.162). The other condition – Eq. (4.161) is verified in Lemma 14 below.

Lemma 13. *If $M \leq \exp(rd/32)$ and $d \geq 4r$, then $\exists M$ rank-r subspace projectors $\Pi_1, \dots, \Pi_M$ such that*

$$\mathrm{tr}(\Pi_i\Pi_j)/r < 1/2, \forall i \neq j. \tag{4.164}$$

Proof. We find the subspace projectors using a probabilistic argument. We randomly choose M rank-r subspaces according to the unitarily invariant measure in the Hilbert space, the Grassmannian, and bound the probability that the randomly chosen subspaces do not satisfy the condition. For a pair of fixed $i \neq j$, we have

$$\Pr\left[\frac{1}{r}\,\mathrm{tr}(\Pi_i\Pi_j) \geq \frac{1}{2}\right] \leq \exp\left(-r^2 f\left(\frac{d}{2r} - 1\right)\right) < \exp\left(-\frac{rd}{16}\right), \tag{4.165}$$

where we make use of (Haah et al., 2017, Lemma 6) in the first inequality and $f(z) = z - \log(1 + z) > z/4$ for all $z \geq 1$ in the second inequality. A union bound then asserts

$$\Pr\left[\exists i \neq j, \frac{1}{r}\,\mathrm{tr}(\Pi_i\Pi_j) \geq \frac{1}{2}\right] < M^2 \exp\left(-\frac{rd}{16}\right) \leq 1. \tag{4.166}$$

Because the probability is less than one, there must exist $\Pi_1, \dots, \Pi_M$ that satisfy the desired property. □

Lemma 14. *Consider a set of d-dimensional quantum states $\{\rho_1, \dots, \rho_M\}$ such that $\rho_i = (1 - \alpha)\frac{\mathbb{I}}{d} + \alpha\frac{\Pi_i}{r}$, where Π_i is a rank-r subspace projector. Consider U sampled from Haar measure, and X sampled from $\{1, \dots, M\}$ uniformly at random. Consider any POVM measurement F. Then the information gain regarding X, conditioned on U, obtained from the measurement F performed on the state $U\rho_X U^\dagger$ satisfies*

$$I(X : F \text{ on } U\rho_X U^\dagger | U) \leq \frac{\alpha^2}{r}. \tag{4.167}$$

Note that we can obtain the statement (4.161) by choosing $\alpha = 3\epsilon$ and $r = B/4$, hence completing the proof of Theorem 15.

Proof. First of all, let us decompose all POVM elements $\{F_1, \dots, F_l\}$ to rank-1 elements $F' = \left\{w_i d \left|v_i\right\rangle\left\langle v_i\right|\right\}_{i=1}^{l'}$, where $l \leq l'$. We can perform measurement F by performing measurement with F': when we measure a rank-1 element, we return the original POVM element the rank-1 element belongs to. Using data processing inequality, we have $I(X : F \text{ on } U\rho_X U^\dagger|U) \leq I(X : \tilde{F} \text{ on } U\rho_X U^\dagger|U)$. From now on, we can consider the POVM $\mathbf{F}$ to be $\left\{w_i d \left|v_i\right\rangle\left\langle v_i\right|\right\}_{i=1}^{l}$. Normalization demands

$$\operatorname{tr}\Big(\sum_i w_i d \left|v_i\right\rangle\left\langle v_i\right|\Big) = \operatorname{tr}(\mathbb{I}) = d \quad \text{and therefore} \quad \sum_i w_i = 1. \tag{4.168}$$

Let us define the probability vector $\mathbf{p} = \operatorname{tr}(U\rho_1 U^\dagger \mathbf{F})$, so $p_i = w_i d \left\langle v_i\right| U\rho_1 U^\dagger \left|v_i\right\rangle.$ And the expression we hope to bound satisfies $I(X : F \text{ on } U\rho_X U^\dagger|U) = I(X, U : F \text{ on } U\rho_X U^\dagger) - I(U : F \text{ on } U\rho_X U^\dagger) \leq I(X, U : F \text{ on } U\rho_X U^\dagger)$ using the chain rule and the nonnegativity of mutual information. We now bound

$$\begin{aligned}
&I(X, U : F \text{ on } U\rho_X U^\dagger) \qquad (4.169)\\
=&H\Big(\sum_{X=1}^{M} \frac{1}{M} \mathop{\mathbb{E}}_U[\operatorname{tr}(U\rho_X U^\dagger \mathbf{F})]\Big) - \sum_{X=1}^{M} \frac{1}{M} \mathop{\mathbb{E}}_U \Big[H\Big(\operatorname{tr}(U\rho_X U^\dagger \mathbf{F})\Big)\Big]\\
=&H\Big(\operatorname{tr}(\mathop{\mathbb{E}}_U[U\rho_1 U^\dagger]\mathbf{F})\Big) - \mathop{\mathbb{E}}_U\Big[H\Big(\operatorname{tr}(U\rho_1 U^\dagger \mathbf{F})\Big)\Big]\\
=&\sum_i -(\mathop{\mathbb{E}}_U p_i)\log(\mathop{\mathbb{E}}_U p_i) + \mathop{\mathbb{E}}_U[p_i \log p_i]\\
\leq&\sum_i -(\mathop{\mathbb{E}}_U p_i)\log(\mathop{\mathbb{E}}_U p_i) + \mathop{\mathbb{E}}_U\Big[p_i \log(\mathop{\mathbb{E}}_U p_i) + p_i \frac{p_i - \mathbb{E}_U p_i}{\mathbb{E}_U p_i}\Big]\\
=&\sum_i \frac{\mathbb{E}_U[p_i^2] - \mathbb{E}_U[p_i]^2}{\mathbb{E}_U[p_i]}. \qquad (4.170)
\end{aligned}$$

The second equality uses the fact that $\mathbb{E}_U f(U\rho_X U^\dagger) = E_U f(U\rho_1 U^\dagger), \forall X$ which follows from the fact that $\forall X, \exists U_X, \rho_X = U_X \rho_1 U_X^\dagger$. The inequality uses the fact that $\log(x)$ is concave, so $\log(x) \leq \log(y) + \frac{x-y}{y}$. Using properties of Haar random unitary $d \times d$ matrices, we conclude

$$\mathop{\mathbb{E}}_U[p_i] = w_i, \ \mathop{\mathbb{E}}_U[p_i^2] = w_i^2 \frac{d}{(d+1)}\left(1 + \frac{1}{d} + \alpha^2\Big(\frac{1}{r} - \frac{1}{d}\Big)\right). \tag{4.171}$$

Therefore we have

$$\frac{\mathbb{E}_U[p_i^2] - \mathbb{E}_U[p_i]^2}{\mathbb{E}_U[p_i]} = w_i \alpha^2 \frac{d}{d+1}\Big(\frac{1}{r} - \frac{1}{d}\Big) \leq \frac{w_i \alpha^2}{r}, \tag{4.172}$$

which establishes the claim:

$$I(X : F \text{ on } U\rho_X U^\dagger|U) \leq \sum_i \frac{\mathbb{E}_U[p_i^2] - \mathbb{E}_U[p_i]^2}{\mathbb{E}_U[p_i]} \leq \frac{\alpha^2}{r}. \tag{4.173}$$

□

4.12 Information-theoretic bounds on predicting local observables

In Theorem 15, we have shown that if a procedure can predict arbitrary observables with $\mathrm{tr}(O_i^2) \leq B$, then it must use at least $\Omega(B \log(M)/\epsilon^2)$ single-copy measurements (as long as M is not extraordinarily large). A similar argument can be used to show that if a procedure can predict arbitrary k-local observables, then it requires at least $\Omega(2^k \log(M)/\epsilon^2)$ single-copy measurements (when M is not too large). This is because if we focus on a k-qubit subsystem, then the guarantee allows us to predict arbitrary observables $0 \leq O_i \leq \mathbb{I}$ with $\mathrm{tr}(O_i^2) \leq 2^k$. In the following theorem, we show a stronger lower bound by focusing on local measurements. A local measurement is a POVM $\{w_i d \lvert v_i\rangle\langle v_i\rvert\}_i$ where $\lvert v_i\rangle = \lvert v_i^{(1)}\rangle \otimes \ldots \otimes \lvert v_i^{(n)}\rangle$, $\sum_i w_i = 1$, and $d = 2^n$. This is the same as not performing any entangling gates when implementing the measurement. (Random) Pauli basis measurements are a prominent example.

Theorem 16 (Detailed restatement of Theorem 8 for exponential scaling in locality)**.** *Fix a sequence of local measurements $F_1, \ldots, F_N$ on n-qubit system, i.e., $F_j = \{w_{j,i} d \lvert v_{j,i}\rangle\langle v_{j,i}\rvert\}_i$ where $\lvert v_{j,i}\rangle = \lvert v_{j,i}^{(1)}\rangle \otimes \ldots \otimes \lvert v_{j,i}^{(n)}\rangle$, $\sum_i w_{j,i} = 1$, and $d = 2^n$. Suppose that given any M k-local observables $-\mathbb{I} \leq O_1, O_2, \ldots, O_M \leq \mathbb{I}$, there exists a machine (with arbitrary runtime as long as it always terminates) that can use the measurement outcomes of $F_1, \ldots, F_N$ on N copies of an unknown quantum state ρ to ϵ-accurately predict* $\mathrm{tr}(O_1\rho), \ldots, \mathrm{tr}(O_M\rho)$ *with high probability. Assuming $M \leq 3^k \binom{n}{k}$, then necessarily*

$$N \geq \Omega\left(\frac{3^k \log(M)}{\epsilon^2}\right). \tag{4.174}$$

Proof. The proof uses a quantum communication protocol between Alice and Bob, with Loki interfering in the middle. Alice would encode some classical information in the quantum state and send to Bob. Bob would then use the prediction procedure to decode the encoded classical information. In the middle, Loki will alter the quantum state by applying a random unitary. Loki would then reveal the random unitary to Bob after Bob performed quantum measurements on the quantum states. An illustration of the communication protocol can be found in Supplementary Figure 4.10. The quantum state Alice encodes, the unitary applied by Loki, and the features predicted by Bob are considerably simplified in this result compared to the previous proof.

We define $\rho_i = (\mathbb{I} + 3\epsilon P_i)/2^n, \forall i = 1, \ldots, M$. P_i is the i-th Pauli observable acting on k qubits in the n-qubit system. Any ordering of the Pauli observables is fine. Note that there are at most $3^k \binom{n}{k}$ such Pauli observables. This is the reason why we assume $M \leq 3^k \binom{n}{k}$. The corresponding linear functions chosen by Bob are $O_i = P_i, \forall i = 1, \ldots, M$. This guarantees the following relation:

$$\mathrm{tr}(O_i \rho_j) = 3\epsilon \delta_{ij} \quad \text{for all } 1 \leq i, j \leq M, \tag{4.175}$$

where δ_{ij} is the Kronecker-delta ($\delta ij = 1$ if $i = j$ and $\delta_{ij} = 0$ otherwise). The random unitary applied by Loki consists of random single-qubit unitary rotations, i.e. $U = U^{(1)} \otimes \ldots \otimes U^{(n)}$. The complete communication protocol works as follows.

1. Alice randomly selects an integer X from $\{1, \ldots, M\}$.
2. Alice prepares N copies of the code-state ρ_X according associated to X and sends them to Bob.
3. Loki intercepts the N copies, samples a random unitary $U = U^{(1)} \otimes \ldots \otimes U^{(n)}$, applies U on all copies of $\rho_X \to U \rho_X U^\dagger$, and sends to Bob.
4. Bob performs local measurements F_j on individual states and receives a string of measurement outcomes Y.
5. Loki reveals the random unitary U to Bob. Now Bob would have to predict the expectation value of $UO_1U^\dagger, \ldots, UO_MU^\dagger$ instead of the original $O_1, \ldots, O_M$.
6. Since $UO_1U^\dagger, \ldots, UO_MU^\dagger$ are still k-local observables, Bob can input Y into the feature prediction machine to estimate $\langle UO_iU^\dagger \rangle_{U\rho_X U^\dagger} = \mathrm{tr}(O_i \rho_X), \forall i = 1, \ldots, M$.
7. Bob finds $\overline{X} \in \{1, \ldots, M\}$ that has the largest $\mathrm{tr}(O_{\overline{X}} \rho_X)$.

Because $\mathrm{tr}(O_i \rho_X)$ are predicted to ϵ additive error, and $\mathrm{tr}(O_i \rho_X) = 3\epsilon \delta_{iX}$, if the prediction procedure works as guaranteed, Bob's decoded information $\hat{X}$ would be equal to Alice's encoded information X with high probability. Moreover, we assume that Alice selects her message uniformly at random. Fano's inequality then implies

$$I(X : \overline{X}) = H(X) - H(X|\overline{X}) \geq \Omega(\log(M)), \tag{4.176}$$

where $I(X : \overline{X})$ is the mutual information, and $H(X)$ is the Shannon entropy. By assumption, Loki chooses the random unitary U regardless of the message X. This implies $I(X:U)=0$ and, in turn

$$I(X:\overline{X}) \leq I(X:\overline{X},U) = I(X:U) + I(X:\overline{X}|U) = I(X:\overline{X}|U). \tag{4.177}$$

For fixed U, $\overline{X}$ is the output of the machine that only takes into account the measurement outcomes Y. The data processing inequality then implies

$$I(X:Y|U) \geq I(X:\overline{X}|U) \geq I(X:\overline{X}) \geq \Omega(\log(M)). \tag{4.178}$$

Recall that Y is the measurement outcome of the N POVMs $F_1,\ldots,F_N$. We denote the measurement outcome of F_j as Y_j. Because $Y_1,\ldots,Y_N$ are random variables that depend on X and U,

$$\begin{aligned} I(X:Y|U) &= H(Y_1,\ldots,Y_N|U) - H(Y_1,\ldots,Y_N|X,U) \\ &\leq H(Y_1|U)+\ldots+H(Y_N|U) - H(Y_1,\ldots,Y_N|X,U) \\ &= \sum_{j=1}^{N}\Big(H(Y_j|U)-H(Y_j|X,U)\Big) = \sum_{j=1}^{N} I(X:F_j \text{ on } U\rho_X U^\dagger|U). \end{aligned} \tag{4.179}$$

The second to last equality uses the fact that when X,U are fixed, $Y_1,\ldots,Y_N$ are independent. This part of the derivation is exactly the same as in Section 4.11. All that is left is to properly upper bound $I(X:F_j \text{ on } U\rho_X U^\dagger|U)$. First, by definition,

$$\begin{aligned} I(X:F_j \text{ on } U\rho_X U^\dagger|U) &= \mathop{\mathbb{E}}_{U}\left[H(F_j \text{ on } U\rho_X U^\dagger) - H(X, F_j \text{ on } U\rho_X U^\dagger)\right] \\ &= \mathop{\mathbb{E}}_{U}\left[H\left(\mathop{\mathbb{E}}_{X}\operatorname{tr}(U\rho_X U^\dagger \mathbf{F}_j)\right) - \mathop{\mathbb{E}}_{X} H\left(\operatorname{tr}(U\rho_X U^\dagger \mathbf{F}_j)\right)\right] \\ &\leq H\left(\mathop{\mathbb{E}}_{X}\mathop{\mathbb{E}}_{U}\operatorname{tr}(U\rho_X U^\dagger \mathbf{F}_j)\right) - \mathop{\mathbb{E}}_{X}\mathop{\mathbb{E}}_{U} H\left(\operatorname{tr}(U\rho_X U^\dagger \mathbf{F}_j)\right). \end{aligned} \tag{4.180}$$

The last inequality exploits concavity of the Shannon entropy $H(\cdot)$. By assumption, the F_j's must be local measurements, i.e. $F_j = \{w_{j,i} d \,|v_{k,i}\rangle\langle v_{k,i}|\}_i$ where $|v_{k,i}\rangle = |v_{k,i}^{(1)}\rangle \otimes \ldots \otimes |v_{k,i}^{(n)}\rangle$, $\sum_i w_i = 1$, and $d=2^n$. We define the probability of measuring i-th outcome using POVM F_j as

$$p_{j,i} = w_{j,i} d \,\langle v_{j,i}|\, U\rho_X U^\dagger \,|v_{j,i}\rangle\,, \tag{4.181}$$

which is a random number depending on X and U. Using Equation (4.180) and the definition of $H(\cdot)$, we have

$$I(X:F_j \text{ on } U\rho_X U^\dagger|U) \tag{4.182}$$

$$
\begin{aligned}
&\leq H\left(\mathop{\mathbb{E}}_{X}\mathop{\mathbb{E}}_{U}\mathrm{tr}(U\rho_X U^\dagger \mathbf{F}^{(k)})\right) - \mathop{\mathbb{E}}_{X}\mathop{\mathbb{E}}_{U} H\left(\mathrm{tr}(U\rho_X U^\dagger \mathbf{F}^{(k)})\right) \\
&= \sum_i \big(\mathop{\mathbb{E}}_{X,U}[p_{j,i}\log(p_{j,i})] - \mathop{\mathbb{E}}_{X,U}[p_{j,i}]\log(\mathop{\mathbb{E}}_{X,U}[p_{j,i}])\big) \\
&\leq \sum_i -(\mathop{\mathbb{E}}_{X,U} p_{j,i})\log(\mathop{\mathbb{E}}_{X,U} p_{j,i}) + \mathop{\mathbb{E}}_{X,U}\left[p_{j,i}\log(\mathop{\mathbb{E}}_{X,U} p_{j,i}) + p_{j,i}\frac{p_{j,i} - \mathbb{E}_{X,U}\, p_{j,i}}{\mathbb{E}_{X,U}\, p_{j,i}}\right] \\
&= \sum_i \frac{\mathbb{E}_{X,U}[p_{j,i}^2] - \mathbb{E}_{X,U}[p_{j,i}]^2}{\mathbb{E}_{X,U}[p_{j,i}]}. \qquad (4.183)
\end{aligned}
$$

The second inequality uses the fact that $\log(x)$ is concave, so $\log(x) \leq \log(y) + \frac{x-y}{y}$. We now compute $\mathbb{E}_{X,U}[p_{j,i}]$ and $\mathbb{E}_{X,U}[p_{j,i}^2]$ by using the following relation for single-qubit random unitary:

$$
\begin{aligned}
\mathop{\mathbb{E}}_{U^{(j)}}\left[U^{(j)}\,|v_{k,i}^{(j)}\rangle\langle v_{k,i}^{(j)}|\,(U^{(j)})^\dagger\right] &= \frac{\mathbb{I}^{(j)}}{2}, \qquad (4.184)\\
\mathop{\mathbb{E}}_{U^{(j)}}\left[\left(U^{(j)}\,|v_{k,i}^{(j)}\rangle\langle v_{k,i}^{(j)}|\,(U^{(j)})^\dagger\right)^{\otimes 2}\right] &= \frac{\mathbb{I}^{(j)}\otimes\mathbb{I}^{(j)} + S^{(j)}}{3}, \qquad (4.185)
\end{aligned}
$$

where j refers to the j-th qubit, and S is the two qubit swap operator ($|\psi\rangle\otimes|\phi\rangle = |\phi\rangle\otimes|\psi\rangle$). Recall the definition of $p_{j,i}$ in Equation (4.181). Together with the above relation, we have

$$
\begin{aligned}
\mathop{\mathbb{E}}_{X,U}[p_{j,i}] &= \mathop{\mathbb{E}}_{X}\left[w_{j,i} d\,\mathrm{tr}\left(\rho_X\frac{\mathbb{I}}{2^n}\right)\right] = \mathop{\mathbb{E}}_{X}\left[w_{j,i}2^n\,\mathrm{tr}\left(\frac{\mathbb{I}+3\epsilon P_X}{2^n}\frac{\mathbb{I}}{2^n}\right)\right] = w_{j,i} \quad \text{and}\\
\mathop{\mathbb{E}}_{X,U}[p_{j,i}^2] &= \mathop{\mathbb{E}}_{X}\left[w_{j,i}^2 d^2\,\mathrm{tr}\left(\rho_X^{\otimes 2}\bigotimes_{j=1}^{n}\left(\frac{\mathbb{I}^{(j)}\otimes\mathbb{I}^{(j)}+S^{(j)}}{3}\right)\right)\right] = w_{j,i}^2\left(1+\frac{9\epsilon^2}{3^k}\right). \qquad (4.186)
\end{aligned}
$$

Putting this computation into Inequality (4.183), we have obtained

$$
I(X : F_j \text{ on } U\rho_X U^\dagger | U) \leq \sum_i w_{j,i}\frac{9\epsilon^2}{3^k} = \frac{9\epsilon^2}{3^k}. \qquad (4.187)
$$

Combining the above result with Inequality (4.178) and (4.179), we have

$$
\frac{9N\epsilon^2}{3^k} \geq I(X:Y|U) \geq \Omega(\log(M)) \quad \text{which implies} \quad N \geq \Omega\left(\frac{3^k\log(M)}{\epsilon^2}\right). \qquad (4.188)
$$

□

Chapter 5

SOLVING QUANTUM MANY-BODY PROBLEMS

Solving quantum many-body problems, such as finding ground states of quantum systems, has far-reaching consequences for physics, materials science, and chemistry. While classical computers have facilitated many profound advances in science and technology, they often struggle to solve such problems. Powerful methods, such as density functional theory (P. Hohenberg and W. Kohn, 1964; W. Kohn, 1999), quantum Monte Carlo (Ceperley and Alder, 1986; Sandvik, 1999; Becca and Sorella, 2017) and density-matrix renormalization group (Steven R. White, 1992; Steven R. White, 1993b), have enabled solutions to certain restricted instances of many-body problems, but many general classes of problems remain outside the reach of even the most advanced classical algorithms.

Scalable fault-tolerant quantum computers will be able to solve a broad array of quantum problems, but are unlikely to be available for years to come. Meanwhile, how can we best exploit our powerful classical computers to advance our understanding of complex quantum systems? Recently, classical machine learning (ML) techniques have been adapted to investigate problems in quantum many-body physics (Carleo, Cirac, et al., 2019; Carrasquilla, 2020), with promising results (Deng, Xiaopeng Li, and Das Sarma, 2017; Carrasquilla and Roger G. Melko, 2017b; Carleo and Troyer, 2017b; Torlai and Roger G. Melko, 2016; Nomura et al., 2017; Nieuwenburg, Y.-H. Liu, and Sebastian D. Huber, 2017; Wang, 2016; Gilmer et al., 2017; Torlai, Mazzola, et al., 2018; Vargas-Hernández et al., 2018; Schütt et al., 2019; Glasser et al., 2018; Rodriguez-Nieva and Scheurer, 2019; Qiao et al., 2020; Choo, Mezzacapo, and Carleo, 2020; Kawai and Nakagawa, 2020; Moreno, Carleo, and Georges, 2020; Kottmann et al., 2021). So far, these approaches are mostly heuristic, reflecting the general paucity of rigorous theory in ML. While shown to be effective in some intermediate-size experiments (Bohrdt et al., 2019; Rem et al., 2019; Torlai, Timar, et al., 2019), these methods are generally not backed by convincing theoretical arguments to ensure good performance, particularly for problem instances where traditional classical algorithms falter.

In general, simulating quantum many-body physics is hard for classical computers because accurately describing an n-qubit quantum system may require an amount of classical data that is exponential in n. In Chapter 4, we addressed this bottleneck using *classical shadows* — succinct classical descriptions of quantum many-body states that can be used to accurately predict a wide range of properties with rigorous performance guarantees (Huang, Richard Kueng, and Preskill, 2020; Paini and Kalev, 2019). Furthermore, this quantum-to-classical conversion technique can be readily implemented in various existing quantum experiments (Struchalin et al., 2021; Andreas Elben, Richard Kueng, et al., 2020b; J. Choi et al., 2021). Classical shadows open new opportunities for addressing quantum problems using classical methods such as ML. In this chapter, we build on the classical shadow formalism and devise polynomial-time classical ML algorithms for quantum many-body problems, which are supported by rigorous theory.

We consider two applications of classical ML, indicated in Figure 5.1. The first application we examine is learning to predict classical representations of quantum many-body ground states. We consider a family of Hamiltonians, where the Hamiltonian $H(x)$ depends smoothly on m real parameters (denoted by x). The ML algorithm is trained on a set of training data consisting of sampled values of x, each accompanied by the corresponding classical shadow for the ground state $\rho(x)$ of $H(x)$. This training data could be obtained from either classical simulations or quantum experiments. During the prediction phase, the ML algorithm predicts a classical representation of $\rho(x)$ for new values of x different from those in the training data. Ground state properties can then be estimated using the predicted classical representation.

This learning algorithm is efficient, provided that the ground state properties to be predicted do not vary too rapidly as a function of x. Indeed, sufficient upper bounds on the gradient can be derived for any family of gapped geometrically-local Hamiltonians in any finite spatial dimension if the property of interest is the expectation value of a sum of few-body observables. The conclusion is that any such property can be predicted with a small average error, where the amount of training data and the classical computation time are polynomial in m and linear in the system size. Furthermore, we show that classical algorithms that do not learn from data cannot provide the same rigorous guarantee without violating widely accepted complexity-theoretic conjectures. This is a manifestation of the advantage of ML algorithms with data over those without data (Huang, Broughton, Masoud Mohseni, Babbush, Boixo, Neven, and Jarrod R McClean, 2021a) as discussed in Section 3.1.

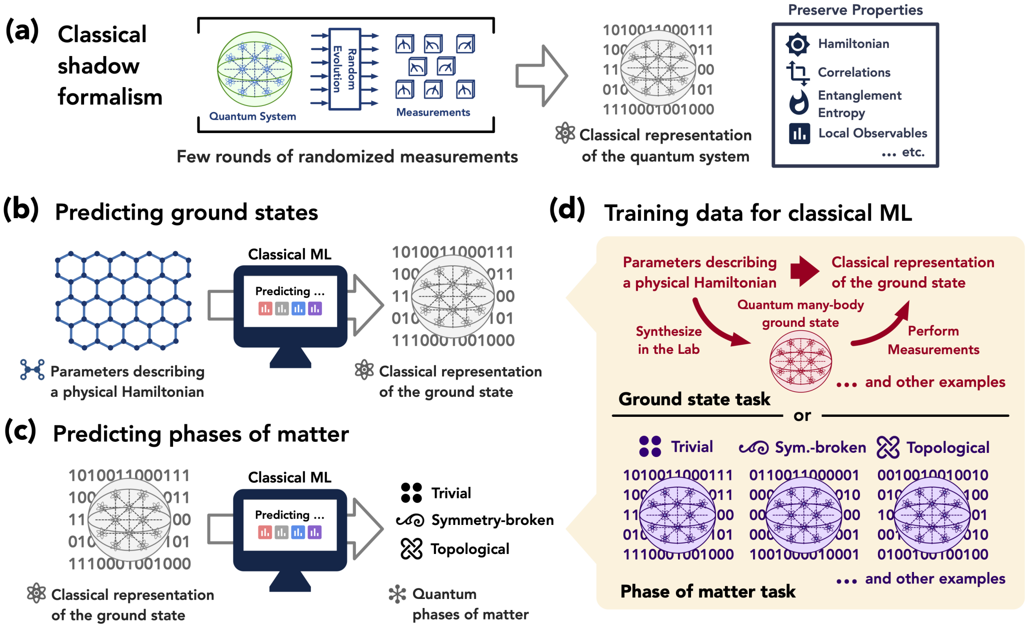


Figure 5.1: (a) Efficient quantum-to-classical conversion. The classical shadow of a quantum state, constructed by measuring very few copies of the state, can be used to predict many properties of the state with a rigorous performance guarantee. (b) Predicting ground state properties. After training on data obtained in quantum experiments, a classical ML model predicts a classical representation of the ground state $\rho(x)$ of the Hamiltonian $H(x)$ for parameters x spanning the entire phase. This representation yields estimates of the properties of $\rho(x)$, avoiding the need to run exhaustive classical computations or quantum experiments. (c) Classifying quantum phases. After training, a classical ML receives a classical representation of a quantum state and predicts the phase from which the state was drawn. (d) Training data. For predicting ground states, the classical ML receives a classical representation of $\rho(x)$ for each value of x sampled during training. For predicting quantum phases of matter, the training data consists of classical representations of quantum states accompanied by labels identifying the phase to which each state belongs.

If the training data are obtained from quantum experiments, then one might choose to learn about properties of $\rho(x)$ for a new input x by conducting new experiments rather than by using the classical ML to generalize from the training data. However, ML could be far more convenient in some cases, especially when changing some parameters may even require costly re-engineering of the entire experiment. ML algorithms open up the possibility of efficiently and accurately predicting properties of quantum states that are extremely challenging to prepare and measure in the laboratory.

Classical ML could be used to generalize from training data that are obtained from either quantum experiments or classical simulations; the same rigorous performance guarantees apply in either case. Even if the training data are generated classically, it could be more efficient and more accurate to use ML to predict properties for new values of the input x, rather than doing new simulations which could be computationally very demanding and of unverified reliability. Promising insights into quantum many-body physics are already being obtained using classical ML based on classical simulation data (Deng, Xiaopeng Li, and Das Sarma, 2017; Nomura et al., 2017; Carleo and Troyer, 2017b; Y. Zhang, Roger G Melko, and E.-A. Kim, 2017; Y. Zhang, Ginsparg, and E.-A. Kim, 2020; Gilmer et al., 2017; Vargas-Hernández et al., 2018; Schütt et al., 2019; Qiao et al., 2020; Choo, Mezzacapo, and Carleo, 2020; Kawai and Nakagawa, 2020). Our rigorous analysis identifies general conditions that guarantee the success of classical ML models, and elucidates the advantages of classical ML models over non-ML algorithms. These results enhance the prospects for interpretable ML techniques (Ribeiro, Singh, and Guestrin, 2016; Arrieta et al., 2020; Y. Zhang, Ginsparg, and E.-A. Kim, 2020) to further shed light on quantum many-body physics.

In the second application we examine, the goal is to classify quantum states of matter into phases (Read, 2012) in a supervised learning scenario. Suppose that during training, we are provided with sample quantum states which carry labels indicating whether each state belongs to phase A or phase B. Our goal is to predict the phase label for new quantum states that were not encountered during training. We assume that, during both the learning and prediction stages, each quantum state is represented by its classical shadow, which could be obtained either from a classical computation or from an experiment on a quantum device. The classical ML, then, trains on labeled classical shadows, and learns to predict labels for new classical shadows.

We assume that the A and B phases can be distinguished by a nonlinear function of marginal density operators of subsystems of constant size. This assumption is reasonable because we expect the phase to be revealed in subsystems that are larger than the correlation length but independent of the total system size. We show that if such a function exists, a classical ML can learn to distinguish the phases using an amount of training data and classical processing, which are polynomial in the system size. We do not need to know anything about this nonlinear function in advance, apart from its existence.

In this chapter, we briefly review the classical shadow formalism (Huang, Richard Kueng, and Preskill, 2020) for readers that skipped Chapter 4 and use this formalism to derive rigorous guarantees for ML algorithms in predicting ground state properties and classifying quantum phases of matter. We also describe numerical experiments in a wide range of physical systems to support our theoretical results.

5.1 A brief review of classical shadow tomography

The classical shadows formalism uses randomized (single-shot) measurements to predict many properties of an unknown quantum state ρ at once (Huang, Richard Kueng, and Preskill, 2020). The underlying idea dates back to (Ohliger, Nesme, and Eisert, 2013b) and also features prominently in (Enk and Beenakker, 2012; A. Elben et al., 2019b; Vermersch et al., 2018). In particular, the classical shadows formalism comes with rigorous performance guarantees in terms of approximation accuracy, classical storage, as well as data processing. Here, we focus on randomized single-qubit Pauli measurements and repeat the following procedure a total of T times: (i) prepare an independent copy of ρ; (ii) select n single-qubit Pauli measurements uniformly at random (Z, X and Y occur with probability $1/3$ each) and (iii) perform the associated measurement to obtain n classical bits (+1 if we measure 'up' and -1 if we measure 'down'). Subsequently, we store the associated post-measurement state

$$|s_1^{(t)}\rangle \otimes \cdots \otimes |s_n^{(t)}\rangle \quad \text{with} \quad |s_1^{(t)}\rangle, \ldots, |s_n^{(t)}\rangle \in \{|0\rangle, |1\rangle, |+\rangle, |-\rangle, |\mathrm{i}+\rangle, |\mathrm{i}-\rangle\} \subset \mathbb{C}^2 \tag{5.1}$$

in classical memory. This is very cheap because there are only six possibilities for each qubit. Randomized measurements can be performed in actual physical experiments or through classical simulations. After T repetitions, we obtain an entire collection of nT single-qubit states that we arrange in a two-dimensional array:

$$S_T(\rho) = \left\{|s_i^{(t)}\rangle : \; i \in \{1, \ldots, n\}, t \in \{1, \ldots, T\}\right\} \in \{|0\rangle, |1\rangle, |+\rangle, |-\rangle, |\mathrm{i}+\rangle, |\mathrm{i}-\rangle\}^{n\times T} \tag{5.2}$$

The distribution of product states contains valuable information about the underlying n-qubit density matrix ρ. In fact, we can use $S_T(\rho)$ to approximate ρ via

$$\rho \approx \sigma_T(\rho) = \frac{1}{T}\sum_{t=1}^{T} \sigma_1^{(t)} \otimes \cdots \otimes \sigma_n^{(t)} \quad \text{where} \quad \sigma_i^{(t)} = 3|s_i^{(t)}\rangle\langle s_i^{(t)}| - \mathbb{I}, \tag{5.3}$$

and $\mathbb{I}$ denotes the identity matrix (here, a 2-by-2 identity). It is instructive to view this as the empirical average of T independent and identically (*iid*) random matrices.

Each random matrix is an *iid* copy of $\sigma_1(\rho) = \big(3|s_1\rangle\langle s_1| - \mathbb{I}\big) \otimes \cdots \otimes \big(3|s_n\rangle\langle s_n| - \mathbb{I}\big)$. Each tensor factor is guaranteed to have eigenvalues $\lambda_+ = 2$ and $\lambda_- = -1$. This ensures that

$$\begin{aligned} \operatorname{tr}(\sigma_1(\rho)) =& \operatorname{tr}(|s_1\rangle\langle s_1| - \mathbb{I}) \cdots \operatorname{tr}(|s_n\rangle\langle s_n| - \mathbb{I}) = 1 \quad \text{and} && \text{(5.4a)} \\ \|\sigma_1(\rho)\|_p =& \, \|3|s_1\rangle\langle s_1| - \mathbb{I}\|_p \cdots \|3|s_n\rangle\langle s_n| - \mathbb{I}\|_p && \text{(5.4b)} \\ =& \, (|\lambda_+|^p + |\lambda_-|^p)^{n/p} = (2^p + 1^p)^{n/p}, && \text{(5.4c)} \end{aligned}$$

regardless of the concrete realization (and the underlying quantum state ρ). The most relevant Schatten-p norms are $\|\sigma_1(\rho)\|_1 = 3^n$, $\|\sigma_1(\rho)\|_2 = 5^{n/2}$ and $\|\sigma_1(\rho)\|_\infty = 2^n$. Note, however, that the matrix $\sigma_1(\rho)$ is never positive semidefinite.

The random matrix $\sigma_1(\rho)$ is a highly structured tensor product that can assume a total of 6^n values. Each of them reflects the outcome of performing randomly selected single-qubit Pauli measurements on the n-qubit state ρ. Let us denote these Pauli matrices by $W_1, \ldots, W_n \in \{X, Y, Z\}$ and let $o_1, \ldots, o_n \in \{\pm 1\}$ be the observed outcomes (+1 if we measure 'spin up' and −1 if we measure 'spin down'). Elementary reformulations and Born's rule then imply

$$\begin{aligned} \sigma_1(\rho) =& \, \frac{1}{2}(\mathbb{I} + 3o_1 W_1) \otimes \cdots \otimes \frac{1}{2}(\mathbb{I} + 3o_n W_n) && \text{(5.5)} \\ \text{with prob.} \quad & \frac{1}{3^n}\operatorname{tr}\left(\frac{1}{2}(\mathbb{I} + o_1 W_1) \otimes \cdots \otimes \frac{1}{2}(\mathbb{I} + o_n W_n)\rho\right). && \text{(5.6)} \end{aligned}$$

This construction ensures that $\sigma_1(\rho)$ exactly reproduces the underlying quantum state ρ in expectation. That is, if we average over all 3^n choices of Pauli measurements and the associated (single-shot) outcomes $o_i \in \{\pm 1\}$, we obtain

$$\begin{aligned} & \mathop{\mathbb{E}}_{s_1,\ldots,s_n} [\sigma_1(\rho)] && \text{(5.7a)} \\ &= \mathop{\mathbb{E}}_{s_1,\ldots,s_n} \left[\frac{1}{2}(\mathbb{I} + 3o_1 W_1) \otimes \cdots \otimes \frac{1}{2}(\mathbb{I} + 3o_n 3W_n)\right] && \text{(5.7b)} \\ &= \sum_{W_1,\ldots,W_n = X,Y,Z} \sum_{o_1,\ldots,o_n = \pm 1} \frac{1}{3^n}\operatorname{tr}\left(\frac{1}{2}(\mathbb{I} + o_1 W_1) \otimes \cdots \otimes \frac{1}{2}(\mathbb{I} + o_n W_n)\rho\right) && \text{(5.7c)} \\ &\qquad \times \frac{1}{2}(\mathbb{I} + o_1 3W_1) \otimes \cdots \otimes \frac{1}{2}(\mathbb{I} + o_n 3W_n) && \text{(5.7d)} \\ &= \rho. && \text{(5.7e)} \end{aligned}$$

We refer the readers to Ref. (Huang, Richard Kueng, and Preskill, 2020) for a more detailed derivation and context.

The classical shadow (5.3) attempts to approximate this expectation value by an empirical average over T independent samples, much like Monte Carlo sampling approximates an integral. The accuracy of the approximation increases with T, but insisting on accurate approximations of the global state ρ is prohibitively expensive. Known fundamental lower bounds (Steven T Flammia et al., 2012; Haah et al., 2017) state that classical shadows of exponential size (at least) $T = \Omega\left(2^n/\epsilon^2\right)$ are required to ϵ-approximate ρ in trace distance. This quickly becomes intractable in terms of both measurement budget, as well as classical storage and processing.

This bleak picture lightens up considerably if we restrict our attention to subsystem approximations. The classical shadow size required to accurately approximate *all* reduced r-body density matrices scales exponentially in subsystem size r, but is independent of the total number of qubits n.

Lemma 15. *Fix $\epsilon, \delta \in (0,1)$, a subsystem size $r \leq n$ and let $\sigma_T(\rho)$ be a classical shadow (5.3) of an n-qubit quantum state ρ with size*

$$T = (8/3)12^r \left(r\left(\log(n) + \log(12)\right) + \log(1/\delta)\right)/\epsilon^2 = O\left(r12^r \log(n/\delta)/\epsilon^2\right). \tag{5.8}$$

Then, with probability at least $1-\delta$,

$$\|\mathrm{tr}_{\neg A}\left(\sigma_T(\rho)\right) - \mathrm{tr}_{\neg A}\left(\rho\right)\|_1 \leq \epsilon \tag{5.9}$$

for all subsystems $A \subset \{1,\ldots,n\}$ with size $|A| \leq r$.

Proof. Let us start by considering a fixed subsystem $A = \{i_1,\ldots,i_r\}$ comprised of (at most) r qubits. Use linearity to exchange partial trace with expectation value to obtain

$$\mathop{\mathbb{E}}_{s_{i_1}^{(t)},\ldots,s_{i_r}^{(t)}} \left[\left(3|s_{i_1}^{(t)}\rangle\langle s_{i_1}^{(t)}| - \mathbb{I}\right) \otimes \cdots \otimes \left(3|s_{i_r}^{(t)}\rangle\langle s_{i_r}^{(t)}| - \mathbb{I}\right)\right] \tag{5.10a}$$

$$= \mathrm{tr}_{\neg A}\left(\mathop{\mathbb{E}}_{s_1^{(t)},\ldots,s_n^{(t)}} \left[\left(3|s_1^{(t)}\rangle\langle s_1^{(t)}| - \mathbb{I}\right) \otimes \cdots \otimes \left(3|s_n^{(t)}\rangle\langle s_n^{(t)}| - \mathbb{I}\right)\right]\right) \tag{5.10b}$$

$$= \mathrm{tr}_{\neg A}(\rho), \tag{5.10c}$$

according to Eq. (5.7). In words, each reduced tensor product is an independent random matrix that reproduces the r-qubit state $\mathrm{tr}_{\neg A}(\rho)$ exactly in expectation. Empirical averages of T such independent and identically distributed (*iid*) random matrices tend to concentrate sharply around this expectation value. The matrix Bernstein inequality, see e.g. (Tropp, 2012), provides powerful tail bounds in terms of operator norm deviation. Let $X_1, \ldots, X_T$ be *iid* random D-dimensional matrices that obey $\|X_t - \mathbb{E}\, X_t\|_\infty \leq R$ almost surely. Then, for $\tilde{\epsilon} > 0$

$$\Pr\left[\left\|\tfrac{1}{T}\sum_{t=1}^{T}(X_t - \mathbb{E}\, X_t)\right\|_\infty \geq \tilde{\epsilon}\right] \leq 2D \exp\left(-\frac{T\tilde{\epsilon}^2/2}{\sigma^2 + R\tilde{\epsilon}/3}\right) \tag{5.11}$$

$$\text{where} \quad \sigma^2 = \left\|\tfrac{1}{T}\sum_t \mathbb{E}\, X_t^2\right\|_\infty . \tag{5.12}$$

Let us apply this tail bound to classical shadow concentration. We have $D \leq 2^r$ (at most r qubits) and set $X_t = \left(3|s_{i_1}^{(t)}\rangle\langle s_{i_1}^{(t)}| - \mathbb{I}\right) \otimes \cdots \otimes \left(3|s_{i_r}^{(t)}\rangle\langle s_{i_r}^{(t)}| - \mathbb{I}\right)$, such that $\mathbb{E}\, X_t = \mathrm{tr}_{\neg A}(\rho)$. Eq. (5.4) then implies $\|X_t - \mathbb{E}\, X_t\|_\infty \leq \|X_t\| + \|\mathbb{E}\, X_t\|_\infty \leq 2^r + 1 =: R$. Accurately bounding σ^2 is somewhat more involved, and we turn to existing literature. A computation detailed in (Guţă et al., 2020a, Appendix C.3) yields $\sigma^2 = 3^r$. We are now ready to apply the matrix Bernstein inequality. For $\tilde{\epsilon} > 0$,

$$\Pr\left[\|\mathrm{tr}_{\neg A}\left(\sigma_T(\rho)\right) - \mathrm{tr}_{\neg A}(\rho)\|_\infty \geq \tilde{\epsilon}\right] \tag{5.13}$$

$$\leq 2^{r+1}\exp\left(-\frac{T\tilde{\epsilon}^2/2}{3^r + (2^r+1)\tilde{\epsilon}/3}\right) \leq 2^{r+1}\exp\left(-\frac{3T\tilde{\epsilon}^2}{8\times 3^r}\right), \tag{5.14}$$

for $\tilde{\epsilon} \in (0,1)$. This is a powerful concentration statement in the operator norm. We can use the equivalence relation between trace- and operator norm, $\|X\|_\infty \leq \|X\|_1 \leq D\|X\|_\infty$, to obtain a tail bound for trace norm deviations:

$$\Pr\left[\|\mathrm{tr}_{\neg A}\left(\sigma_T(\rho)\right) - \mathrm{tr}_{\neg A}(\rho)\|_\infty \geq \epsilon\right] \tag{5.15}$$

$$\leq \Pr\left[\|\mathrm{tr}_{\neg A}\left(\sigma_T(\rho)\right) - \mathrm{tr}_{\neg A}(\rho)\|_1 \geq \epsilon/2^r\right] \leq 2^{r+1}\exp\left(-\frac{3T\epsilon^2}{8\times 12^r}\right). \tag{5.16}$$

We see that for a fixed subsystem $A = \{i_1, \ldots, i_r\}$, the probability of an ϵ-deviation in trace distance is exponentially suppressed in the size T of the classical shadow. A union bound allows us to extend this assertion to *all* subsystems comprised of (at most) r qubits:

$$\Pr\left[\max_{A\subset\{1,\ldots,n\},|A|\leq r}\|\mathrm{tr}_{\neg A}\left(\sigma_T(\rho)\right) - \mathrm{tr}_{\neg A}(\rho)\|_1 \geq \epsilon\right] \tag{5.17a}$$

$$\leq \sum_{A\subset\{1,\ldots,n\},|A|\leq r}\Pr\left[\|\mathrm{tr}_{\neg A}\left(\sigma_T(\rho)\right) - \mathrm{tr}_{\neg A}(\rho)\|_1 \geq \epsilon\right] \tag{5.17b}$$

$$\leq n^r 2^{r+1}\exp\left(-\frac{3T\epsilon^2}{8\times 12^r}\right). \tag{5.17c}$$

Setting

$$T = (8/3)12^r\left(\log(n^r 12^r) + \log(1/\delta)\right)/\epsilon^2 \tag{5.18}$$

$$= (8/3)12^r \left(r\left(\log(n)+\log(12)\right)+\log(1/\delta)\right)/\epsilon^2 \tag{5.19}$$

ensures that this upper bound on failure probability does not exceed δ. □

This *classical shadow* representation (Huang, Richard Kueng, and Preskill, 2020; Paini and Kalev, 2019) exactly reproduces the global density matrix in the limit $T \to \infty$, but $T = \mathcal{O}(\mathrm{const}^r \log(n)/\epsilon^2)$ already provides an ϵ-accurate approximation of *all* reduced r-body density matrices (in trace distance). This, in turn, implies that we can use $\sigma_T(\rho)$ to predict any function that depends on only reduced density matrices, such as expectation values of (sums of) local observables and (sums of) entanglement entropies of small subsystems. Classical storage and postprocessing cost also remain tractable in this regime. To summarize, the classical shadow formalism equips us with an efficient quantum-to-classical converter that allows classical machines to efficiently and reliably estimate subsystem properties of any quantum state ρ.

5.2 Predicting ground states of quantum many-body systems

We consider the task of predicting ground state representations of quantum many-body Hamiltonians in finite spatial dimensions. Suppose that a family of geometrically local, n-qubit Hamiltonians $\{H(x) : x \in [-1,1]^m\}$ is parametrized by a classical variable x. That is, $H(x)$ smoothly maps a bounded m-dimensional vector x (parametrization) to a Hermitian matrix of size $2^n \times 2^n$ (n-qubit Hamiltonian). We do not impose any additional structure on this mapping; in particular, we do not assume knowledge about how the physical Hamiltonian depends on the parameterization. The goal is to learn a model $\hat{\sigma}(x)$ that can predict properties of the ground state $\rho(x)$ associated with Hamiltonian. This problem arises in many practical scenarios. Suppose diligent experimental effort has produced experimental data for ground state properties of various physical systems. We would like to use this data to train an ML model that predicts ground state representations of hitherto unexplored physical systems.

An ML algorithm with rigorous guarantee

We will prove that a classical ML algorithm can predict classical representations of ground states after training on data belonging to the same quantum phase of matter. Formally, we consider a smooth family of Hamiltonians $H(x)$ with a constant spectral gap. During the training phase of the ML algorithm, many values of x are randomly sampled, and for each sampled x, the classical shadow of the corresponding ground state $\rho(x)$ of $H(x)$ is provided, either by classical simulations or quantum experiments. The full training data of size N is given by $\left\{x_\ell \to \sigma_T(\rho(x_\ell))\right\}_{\ell=1}^N$, where T is the number of randomized measurements in the construction of the classical shadows at each value of x_ℓ.

We train classical ML models using the size-N training data, such that when given the input x_ℓ, the ML model can produce a classical representation $\hat{\sigma}(x)$ that approximates $\sigma_T(\rho(x_\ell))$. During prediction, the classical ML produces $\hat{\sigma}(x)$ for new values of x different from those in the training data. While $\hat{\sigma}(x)$ and $\sigma_T(\rho(x_\ell))$ classically represent exponentially large density matrices, the training and prediction can be done efficiently on a classical computer using various existing classical ML models, such as neural networks with large hidden layers (Jacot, Gabriel, and Hongler, 2018; Z. Li et al., 2019; Du et al., 2019; Novak, L. Xiao, Hron, J. Lee, Alexander A. Alemi, et al., 2020) and kernel methods (Cortes and Vapnik, 1995; Chang and C.-J. Lin, 2011a). In particular, the predicted output of the trained classical ML models can be written as the extrapolation of the training data using a learned metric $\kappa(x, x_\ell) \in \mathbb{R}$,

$$\hat{\sigma}(x) = \frac{1}{N}\sum_{\ell=1}^{N} \kappa(x, x_\ell)\sigma_T(\rho(x_\ell)). \tag{5.20}$$

For example, prediction using a trained neural network with large hidden layers (Jacot, Gabriel, and Hongler, 2018) is equivalent to using the metric $\kappa(x, x_\ell) = \sum_{\ell'=1}^{N} f^{(\mathrm{NTK})}(x, x_{\ell'})(F^{-1})_{\ell'\ell}$, where $f^{(\mathrm{NTK})}(x, x')$ is the neural tangent kernel (Jacot, Gabriel, and Hongler, 2018) corresponding to the neural network and $F_{\ell'\ell} = f^{(\mathrm{NTK})}(x_{\ell'}, x_\ell)$; see Appendix 5.11 for more discussion. The ground state properties are then estimated using these predicted classical representations $\hat{\sigma}(x)$. Specifically, $f_O(x) = \mathrm{tr}\,(O\rho(x))$ can be predicted efficiently whenever O is a sum of few-body operators.

To derive a provable guarantee, we consider the simple metric

$$\kappa(x, x_\ell) = \sum_{k\in\mathbb{Z}^m, \|k\|_2\le\Lambda} \cos(\pi k\cdot(x - x_\ell)) \tag{5.21}$$

with cutoff Λ, which we refer to as the l_2-Dirichlet kernel. We prove that the prediction will be accurate and efficient if the function $f_O(x)$ does not vary too rapidly when x changes in any direction. Indeed, sufficient upper bounds on the gradient magnitude of $f_O(x)$ can be derived using quasi-adiabatic continuation (Matthew B Hastings and Wen, 2005; Bachmann, Michalakis, et al., 2012).

Under the l_2-Dirichlet kernel, the classical ML model is equivalent to learning a truncated Fourier series to approximate the function $f_O(x)$. The parameter Λ is a cutoff for the wavenumber k that depends on (upper bounds on) the gradient of $f_O(x)$. Using statistical analysis, one can guarantee that $\mathbb{E}_x\,|\operatorname{tr}(O\hat{\sigma}(x)) - f_O(x)|^2 \leq \epsilon$ as long as the amount of training data obeys $N = m^{O(1/\epsilon)}$ in the $m \to \infty$ limit. The conclusion is that any such $f_O(x)$ can be predicted with a small *constant* average error, where the amount of training data and the classical computation time are polynomial in m and at most linear in the system size n. Moreover, the training data need only contain a *single* classical shadow snapshot at each point x_ℓ in the parameter space (i.e., $T = 1$). An informal statement of the theorem is given below; we explain the proof strategy in Appendix 5.3, and provide more details in Appendix 5.4. We also discuss how one could generalize the proof to long-range interacting systems, electronic Hamiltonians, and other settings in Appendix 5.3.

Theorem 17 (Learning to predict ground state representations; informal)**.** *For any smooth family of Hamiltonians $\{H(x) : x \in [-1,1]^m\}$ in a finite spatial dimension with a constant spectral gap, the classical machine learning algorithm can learn to predict a classical representation of the ground state $\rho(x)$ of $H(x)$ that approximates few-body reduced density matrices up to a constant error ϵ when averaged over x. The required training data size N and computation time are polynomial in m and linear in the system size n.*

Though formally "efficient" in the sense that N scales polynomially with m for any fixed approximation error ϵ, the required amount of training data scales badly with ϵ. This unfortunate scaling is not a shortcoming of the considered ML algorithm, but a necessary feature. In Appendix 5.5, we show that the data size and time complexity cannot be improved further without making stronger assumptions about the class of gapped local Hamiltonians. However, in cases of practical interest, the Hamiltonian may obey restrictions such as translational invariance or graph structure that can be exploited to obtain better results. Incorporating these restrictions can be achieved by using a suitable $\kappa(x, x_\ell)$, such as one that corresponds to a large-width convolutional neural network (Z. Li et al., 2019) or a graph neural network (Du et al., 2019). Rigorously establishing that neural-network-based ML algorithms can achieve improved prediction performance and efficiency for particular classes of Hamiltonians is a goal for future work.

Computational hardness for non-ML algorithms

In the following proposition, we show that a classical algorithm that does not learn from data cannot achieve the same guarantee in estimating ground state properties without violating the widely believed conjecture that NP-complete problems cannot be solved in randomized polynomial time. This proposition is a corollary of standard complexity-theoretic results (Lichtenstein, 1982; L. Valiant and V. Vazirani, 1986). See Appendix 5.6 for the detailed statement and proof.

Proposition 8 (Informal)**.** *Consider a randomized polynomial-time classical algorithm $\mathcal{A}$ that does not learn from data. Suppose for any smooth family of two-dimensional Hamiltonians $\{H(x) : x \in [-1,1]^m\}$ with a constant spectral gap, $\mathcal{A}$ can efficiently compute expectation values of one-body observables in the ground state $\rho(x)$ of $H(x)$ up to a constant error when averaged over x. Then there is a randomized classical algorithm that can solve NP-complete problems in polynomial time.*

It is instructive to observe that a classical ML algorithm with access to data can perform tasks that cannot be achieved by classical algorithms which do not have access to data. This phenomenon is studied in (Huang, Broughton, Masoud Mohseni, Babbush, Boixo, Neven, and Jarrod R McClean, 2021a), where it is shown that the complexity class defined by classical algorithms that can learn from data is strictly larger than the class of classical algorithms that do not learn from data. (The data can be regarded as a restricted form of randomized advice string.) We caution that obtaining the data to train the classical ML model could be challenging. However, if we focus only on data that could be efficiently generated by quantum-mechanical processes, it is still possible that a classical ML that learns from data could be more powerful than classical computers. In Appendix 5.6, we present a contrived family of Hamiltonians that establishes this claim based on the (classical) computational hardness of factoring.

5.3 Proof idea for the efficiency in predicting ground states

In order to illustrate the proof of Theorem 17, let us begin by looking at a simpler task: training a machine learning model to predict a specified ground state property instead of the classical representation of the ground state. Consider the property $\mathrm{tr}(O\rho)$, where ρ is the ground state and O is a local observable. In this simpler task, we consider the training data to be

$$\big\{x_1 \to \mathrm{tr}(O\rho(x_1)), \quad \ldots, \quad x_N \to \mathrm{tr}(O\rho(x_N))\big\}, \tag{5.22}$$

where $x_\ell \in [-1,1]^m$ is a classical description of the Hamiltonian $H(x_\ell)$ and $\rho(x_\ell)$ is the ground state of $H(x)$. Intuitively, in a quantum phase of matter, the ground state property $\mathrm{tr}(O\rho(x))$ changes smoothly as a function of the input parameter x. The smoothness condition can be rigorously established as an upper bound on the average magnitude of the gradient of $\mathrm{tr}(O\rho(x))$ using quasi-adiabatic evolution (Matthew B Hastings and Wen, 2005; Bachmann, Michalakis, et al., 2012), assuming that the spectral gap of $H(x)$ is bounded below by a nonzero constant throughout the parameter space. The upper bound on the average gradient magnitude enables us to design a simple classical ML model based on an l_2-Dirichlet kernel for generalizing from the training set to a new input $x \in [-1,1]^m$:

$$\hat{O}_N(x) = \frac{1}{N}\sum_{\ell=1}^{N} \kappa(x, x_\ell)\, \mathrm{tr}(O\rho(x_\ell)) \tag{5.23}$$

$$\text{with } \kappa(x, x_\ell) = \sum_{k\in\mathbb{Z}^m, \|k\|_2\leq\Lambda} \cos(\pi k\cdot(x-x_\ell)) \in \mathbb{R}. \tag{5.24}$$

The l_2-Dirichlet kernel is often used in the study of high-dimensional Fourier series (Weisz, 2012) and the proposed ML model is equivalent to learning a truncated Fourier series to approximate the function $\mathrm{tr}(O\rho(x))$, where the parameter Λ is a cutoff on the wavenumber k that depends on the upper bound on the gradient of $\mathrm{tr}(O\rho(x))$. Using statistical analysis, one can guarantee that $\mathbb{E}_x\, |\hat{O}_N(x) - \mathrm{tr}(O\rho(x))|^2 \leq \epsilon$ as long as the amount of training data $N = m^{O(1/\epsilon)}$ where our big-O notation is with respect to the $m \longrightarrow \infty$ limit. Hence, we can achieve a small *constant* prediction error with an amount of training data and computational time that are both polynomial in the number m of input parameters. The training is efficient because the number of modes needed for the truncated Fourier series to provide an accurate approximation to $\mathrm{tr}(O\rho)$ scales polynomially with m.

The key to the statistical analysis is to bound the model complexity of the above machine learning model. In particular, the model complexity depends on the number of wave vectors we consider in the l_2-Dirichlet kernel. The more wave vectors k we include, the higher the model complexity; and we would have to use more data to train the ML model to achieve good generalization performance. Furthermore, one could show that the amount of data is proportional to the number of wave vectors we consider. In order to achieve a prediction error $\mathbb{E}_x\, |\hat{O}_N(x) - \mathrm{tr}(O\rho(x))|^2 \leq \epsilon$, we would need to select Λ to be of order $\sqrt{1/\epsilon}$. Hence, the number of wave vectors is proportional to the number of lattice points in an m-dimensional l_2 ball of radius Λ. The volume of an m-dimensional l_2 ball with radius Λ is proportional to $\Lambda^m = (1/\epsilon)^{m/2}$. If the number of lattices points is proportional to the volume, then this would imply an exponential scaling in the number of parameters m. However, through a proper combinatorial analysis, we show that the number of lattices points is actually proportional to $m^{O(\Lambda^2)} = m^{O(1/\epsilon)}$, which is only polynomial in the number of parameters m.

We can build on this idea to address the task of predicting ground state representations. Now instead of predicting $\mathrm{tr}(O\rho)$ for a new input x, the goal is to predict the classical shadow of the ground state $\rho(x)$. We consider the training data to be $\left\{x_\ell \to \sigma_1(\rho(x_\ell))\right\}_{\ell=1}^N$, where $\sigma_1(\rho(x_\ell))$ is the classical shadow representation of $\rho(x_\ell)$ obtained from just a *single* randomized Pauli measurement of the state (the $T = 1$ case of Eq. (5.3)). Following the same approach as outlined above for the case of predicting a single property, the predicted ground state representation is now given by

$$\hat{\sigma}_N(x) = \frac{1}{N}\sum_{\ell=1}^{N} \kappa(x, x_\ell)\sigma_1(\rho(x_\ell)) \tag{5.25}$$

$$\text{with } \kappa(x, x_\ell) = \sum_{k\in\mathbb{Z}^m, \|k\|_2\leq\Lambda} \cos(\pi k\cdot(x - x_\ell)) \in \mathbb{R}. \tag{5.26}$$

One can then guarantee that this representation accurately predicts expectation values for a wide range of observables.

The fact that only a single snapshot σ_1 per parameter point is required for our protocol may be surprising. However, since the snapshots depends on the parameters, sampling over training data indirectly samples over different snapshots, and is thus sufficient for a reasonable estimate of properties of the phase. The estimate can of course be further improved if multiple snapshots are used for each parameter point, and we leave proving such improved bounds as an exciting goal for future work.

Generalization to other systems and settings

In this subsection, we discuss how one could generalize the proof of Theorem 17 to various different scenarios.

Prediction based on other quantum measurements Throughout this chapter, we considered classical shadows based on randomized Pauli measurements (Huang, Richard Kueng, and Preskill, 2020). However, it may be difficult to perform randomized Pauli measurements in some experimental systems. Theorem 17 can be directly generalized to other kinds of measurement procedures. Consider a restricted setting where the experimentalist can only obtain training data of the form

$$\{x_\ell \to \mathrm{tr}(O\rho(x_\ell))\}_{\ell=1}^N, \tag{5.27}$$

for a single observable O (that can be written as a sum of local observables). In this case, the classical ML model can no longer predict a classical representation of $\rho(x)$ for a new x. Nevertheless, the classical ML model can still predict $\mathrm{tr}(O\rho(x))$ accurately for a new x by following the proof sketch in Appendix 5.3.

More generally, suppose the experimentalist can construct some classical representation of the ground state $\rho(x_\ell)$ through the available measurements, such as classical shadows based on another random unitary ensemble (H.-Y. Hu and You, 2021), or simply a list of properties of $\rho(x_\ell)$. And suppose that the classical representation allows us to predict the expectation values of observables $O_1, O_2, \ldots, O_M$ in the ground state $\rho(x_\ell)$. Then for a new x, the classical ML model can predict $\mathrm{tr}(O_i\rho(x))$ accurately for $i = 1, \ldots, M$.

A variable number of parameters So far, we have considered the input vector x to be of a fixed dimension m. Here we briefly discuss how to generalize Theorem 17 to a setting where the input is not a fixed dimensional vector. We can think of the input as $\xi = (m, x)$, where $m \in \mathbb{N}$ is a discrete variable specifying the number of parameters, and $x \in \mathbb{R}^m$ is an m-dimensional vector with continuous entries. The number of parameters m may range from $m_{\min}$ to $m_{\max}$. We consider a class of Hamiltonians $H(\xi) = H((m, x))$ that depends on both the discrete parameter m and the continuous vector x. For example, we may have

$$m = 1: \qquad H((m,x)) = \sum_{i=1}^{n} x_1(X_iX_{i+1} + Y_iY_{i+1}), \tag{5.28}$$

$$m = 2: \qquad H((m,x)) = \sum_{i=1}^{n} x_1(X_iX_{i+1} + Y_iY_{i+1}) + x_2(Z_iZ_{i+1}), \tag{5.29}$$

where x_1, x_2 denote the first and second entry of the vector x. In order the train the ML model, we can consider training data to be of the form

$$\left\{\xi_\ell \to \sigma_T(\rho(\xi_\ell))\right\}_{\ell=1}^{N}, \tag{5.30}$$

where $\rho(\xi_\ell)$ is the ground state of the Hamiltonian $H(\xi_\ell)$ (and $\xi_\ell = (m_\ell, x_\ell)$). In this most general case, we can now simply train a distinct ML model for each $m \in [m_{\min}, m_{\max}]$. Using this direct method, we only need a training data size N that is $(m_{\max} - m_{\min} + 1)$ times larger than the training data size when m is fixed.

Systems with long-range interactions For simplicity, the proof for our main theorem (Theorem 17) focuses on Hamiltonians that can be written as a sum of geometrically local terms,

$$H(x) = \sum_j h_j(x), \tag{5.31}$$

where $h_j(x)$ acts on a constant number of constituents that are contained in a ball of constant size in a finite-dimensional space. Our proof can be generalized to some physical systems where $h_j(x)$ acts on constituents that are geometrically non-local. The main condition we must impose is that the evolution under the Hamiltonian $H(x)$ in the ground state $\rho(x)$ has a bounded speed of information spreading. In the study of quantum many-body systems (C.-F. Chen and Lucas, 2019; Kuwahara and Saito, 2020; Tran et al., 2020) this assumption is described as a *linear light cone*, meaning that if a perturbation is applied at a point P at time zero, then the effects of that perturbation at a later time t are mostly confined to a region centered at P with radius vt; here $v > 0$ is called the Lieb-Robinson velocity.

To be more precise, consider two few-body operators, O_A acting on a set of constituents A, and O_B acting on a set of constituents B; the sets A and B need not be geometrically local. We denote by $d(O_A, O_B)$ the minimum Euclidean distance between constituents in A and constituents in B. Recall that in the Heisenberg picture, operators evolve according to $O(t) = \mathrm{e}^{itH(x)} O \mathrm{e}^{-itH(x)}$, where $H(x)$ is the Hamiltonian. We require that the expectation value in the ground state $\rho(x)$ of the commutator of O_A with $O_B(t)$ is highly suppressed when $d(O_A, O_B)$ is small compared to vt, i.e.,

$$|\mathrm{tr}\left([O_A, O_B(t)]\, \rho(x)\right)| \leq \frac{c|t|^\beta}{\max(0, d(O_A, O_B) - v|t|)^\alpha} \left\|O_A\right\|_\infty \left\|O_B\right\|_\infty, \tag{5.32}$$

where c is a constant, and $\alpha > \beta > 0$ are constants that determine the decay,

Such Lieb-Robinson bounds were proven for geometrically local Hamiltonians decades ago, but linear light cones in physical systems with non-local interactions had not been studied until comparatively recently (C.-F. Chen and Lucas, 2019; Kuwahara and Saito, 2020; Tran et al., 2020). It has now been established that, for many long-range interacting systems, Eq. (5.32) applies, where α is sufficiently large compared to β for our arguments to apply. Specifically, in the proof given in Appendix 5.4, we can replace Eq. (5.107) by

$$|\,\mathrm{tr}([O, D_{\hat{u}}(x)]\rho(x))| \tag{5.33}$$

$$\leq \sum_i \int_{-\infty}^{\infty} W_\gamma(t) \sum_j \left|\mathrm{tr}\left(\left[O_i, \mathrm{e}^{\mathrm{i}tH(x)} \frac{\partial h_j}{\partial \hat{u}}(x) \mathrm{e}^{-\mathrm{i}tH(x)}\right] \rho(x)\right)\right| \mathrm{d}t, \tag{5.34}$$

and also replace the Lieb-Robinson bound in Eq. (5.100) by the bound in Eq. (5.32). When α is sufficiently large compared to β in Eq. (5.32), we can guarantee that the right hand side of Eq. (5.33) is upper bounded by

$$\text{const} \times \sum_i \|O_i\|_\infty , \tag{5.35}$$

using an analysis similar to that given in Appendix 5.4. After establishing such an upper bound on $|\operatorname{tr}([O, D_{\hat{u}}(x)]\rho(x))|$, we can follow exactly the same proof given in the other sections in Appendix 5.4 to show that the classical ML model can accurately predict the classical representation of the ground state for long-range interacting systems with a similar guarantee as Theorem 17, assuming that the Lieb-Robinson velocity v is bounded above by a constant.

Fermionic systems We can also generalize the proof of Theorem 17 to fermionic systems, such as those arising in studies of electronic structure; see for example (Helgaker, Jorgensen, and Olsen, 2014). We consider second quantization, also known as the occupation number representation, and use the abstract Fock space to represent the Hamiltonians of fermionic systems. Given a system of n spin orbitals, the Fock space is a 2^n-dimensional space spanned by $|c_0, c_1, \ldots, c_{n-1}\rangle$, where $c_j = 1$ indicates that mode j is occupied and $c_j = 0$ indicates that mode j is unoccupied. A vector in the Fock space is a linear combination of these 2^n basis states. Given a mode $j \in \{1, \ldots, n\}$, a fermionic *creation operator* A_j is defined by

$$\begin{aligned} A_j^\dagger |c_0, c_1, \ldots, 0_j, \ldots, c_{n-1}\rangle &= (-1)^{\sum_{k=0}^{j-1} c_k} |c_0, c_1, \ldots, 1_j, \ldots, c_{n-1}\rangle , \\ A_j^\dagger |c_0, c_1, \ldots, 1_j, \ldots, c_{n-1}\rangle &= 0, \end{aligned} \tag{5.36}$$

whereas the fermionic *annihilation operator* A_j is defined by

$$\begin{aligned} A_j |c_0, c_1, \ldots, 0_j, \ldots, c_{n-1}\rangle &= 0, \\ A_j |c_0, c_1, \ldots, 1_j, \ldots, c_{n-1}\rangle &= (-1)^{\sum_{k=0}^{j-1} c_k} |c_0, c_1, \ldots, 0_j, \ldots, c_{n-1}\rangle . \end{aligned} \tag{5.37}$$

For a fermionic system, each local term $h_j(x)$ in the Hamiltonian $H(x) = \sum_j h_j(x)$ is a Hermitian matrix that can be expressed as a product of an even number of fermionic creation and annihilation operators; we refer to such a Hermitian matrix as an *even fermionic observable*. For example, we could have $h_{pqrs}(x) = U_{pqrs}(x) A_p^\dagger A_q^\dagger A_r A_s + \overline{U_{pqrs}}(x) A_s^\dagger A_r^\dagger A_q A_p$, where $U_{pqrs}(x)$ is a complex-valued number. (This particular term conserves the total fermion number, but fermion number conservation is not actually required for our arguments to work.) Two *even fermionic observables* acting on disjoint sets of spin orbitals commute with one another, just as two local observables acting on disjoint sets of qubits commute. As a result, several results in qubit systems based on the commutation relations of disjoint local observables can be easily generalized to *even fermionic observables* in fermionic systems. In particular, one can generalize the proof of Theorem 17 as follows.

- First, we construct a classical shadow representation for fermionic systems. An efficient approach for constructing such a representation is given in (Zhao, Rubin, and Miyake, 2021). This work rigorously analyzes how to predict a large number of properties using outcomes of measurements performed after randomized fermionic Gaussian unitaries. We can replace the classical shadow based on randomized Pauli measurements with the fermionic partial tomography introduced in (Zhao, Rubin, and Miyake, 2021).

- Secondly, we establish a bounded speed of information spreading under evolution governed by $H(x)$ in the ground state $\rho(x)$. Intuitively, we would like the "diameter" of the support (by "support" we mean the set of spin orbitals that an observable acts on substantially) of an *even fermionic observable* under Heisenberg evolution to grow at most linearly in time. As for qubit systems, this growth rate is known as the Lieb-Robinson velocity. Because two even fermionic observables acting on disjoint sets of spin orbitals commute with one another, one can establish an upper bound on the Lieb-Robinson velocity in fermionic systems by following the argument used for qubit systems (Bru and Siqueira Pedra, 2016; Nachtergaele, Sims, and Young, 2018). This argument does not work for arbitrary fermionic systems, but it does work if the interaction graph of the spin orbitals is suitably sparse.

After these replacements, the rest of the proof follows immediately, yielding a version of Theorem 17 for fermionic systems. As we noted, the argument used to bound the Lieb-Robinson velocity does not work for some fermionic systems; for example it fails in models where orbitals have all-to-all connectivity without any geometrical constraints (the same is true for qubit systems). But the proof of Theorem 17 does go through for tight-binding models, including the Fermi-Hubbard model. Since computing ground state properties of the Fermi-Hubbard model is notoriously difficult for classical computers, it is encouraging to find that our classical ML algorithm can compute these properties efficiently when provided with polynomial-size training data.

5.4 Proof of efficiency for predicting ground states

This section contains a detailed proof for one of our main contributions. Namely, a rigorous performance guarantee for learning to predict ground state representations.

Theorem 18 (Theorem 17, detailed restatement)**.** *Consider any family of n-qubit geometrically-local Hamiltonians $\{H(x) : x \in [-1,1]^m\}$ in a finite spatial dimension, such that each local term in $H(x)$ depends smoothly on x, and the smallest eigenvalue and the next smallest eigenvalues have a constant gap $\gamma \geq \Omega(1)$ between them. Let $\rho(x)$ be the ground state of $H(x)$, that is*

$$\rho(x) = \lim_{\beta\to\infty} e^{-\beta H(x)}/\mathrm{tr}(e^{-\beta H(x)}) \in (\mathbb{H}_2)^{\otimes n} \quad \textit{(ground state of Hamiltonian } H(x)\textit{)} \tag{5.38}$$

where $\mathbb{H}_2$ is the vector space of 2×2 Hermitian matrices. Suppose that we are interested in learning to predict a sum $O = \sum_{i=1}^{L} O_i$ of L local observables that satisfies $\sum_{i=1}^{L} \|O_i\| \leq B$ (bounded norm). Then, classical shadow data $\{x_\ell \to \sigma_1(\rho(x_\ell))\}_{\ell=1}^{N}$, with $x_\ell \sim \mathrm{Unif}[-1,1]^m$ and

$$N = B^2 m^{O(B^2/\epsilon)} \qquad \textit{(training data size)}, \tag{5.39}$$

suffices to produce a ground state prediction model

$$\hat{\sigma}_N(x) = \frac{1}{N}\sum_{\ell=1}^{N} \kappa(x, x_\ell)\rho(x_\ell) \;\; \textit{with} \;\; \kappa(x, x_\ell) = \sum_{k\in\mathbb{Z}^m, \|k\|_2\leq\Lambda} \cos(\pi k \cdot (x - x_\ell)) \in \mathbb{R}, \tag{5.40}$$

that achieves

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |\,\mathrm{tr}(O\hat{\sigma}_N(x)) - \mathrm{tr}(O\rho(x))|^2 \leq \epsilon, \tag{5.41}$$

with high probability. The classical training time for constructing $\hat{\sigma}_N(x)$ and the prediction time for computing $\mathrm{tr}(O\hat{\sigma}(x))$ are both upper bounded by $O((n + L)B^2 m^{O(B^2/\epsilon)})$.

Theorem 18 can be generalized to the following statement about learning a family of quantum states. In particular, we will prove the following theorem and use it to derive Theorem 18.

Theorem 19. *Consider a parametrized family of n-qubit states*

$$\{\rho(x) : x \in [-1,1]^m\} \tag{5.42}$$

and a sum $O = \sum_{i=1}^{L} O_i$ of L local observables that obey

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} \|\nabla_x \operatorname{tr}(O\rho(x))\|_2^2 \leq C \quad \textit{(smoothness condition)}, \tag{5.43a}$$

$$\sum_i \|O_i\| \leq B \quad \textit{(bounded norm)}. \tag{5.43b}$$

Then, classical shadow data $\{x_\ell \to \sigma_1(\rho(x_\ell))\}_{\ell=1}^N$, with $x_\ell \sim \mathrm{Unif}[-1,1]^m$ and

$$N = B^2 m^{O(C/\epsilon)} \qquad \textit{(training data size)}, \tag{5.44}$$

suffices to produce a state prediction model we can learn from classical data $\{x_\ell \to \sigma_1(\rho(x_\ell))\}_{\ell=1}^N$ to produce a model

$$\hat{\sigma}_N(x) = \frac{1}{N}\sum_{\ell=1}^{N} \kappa(x, x_\ell) \operatorname{tr}(O\rho(x_\ell)) \tag{5.45}$$

$$\textit{with} \;\; \kappa(x, x_\ell) = \sum_{k\in\mathbb{Z}^m, \|k\|_2\leq\Lambda} \cos(\pi k\cdot(x - x_\ell)) \in \mathbb{R}, \tag{5.46}$$

that achieves

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |\operatorname{tr}(O\hat{\sigma}_N(x)) - \operatorname{tr}(O\rho(x))|^2 \leq \epsilon, \tag{5.47}$$

with high probability. The classical training time for constructing $\hat{\sigma}_N(x)$ and the prediction time for computing $\operatorname{tr}(O\hat{\sigma}(x))$ *are both upper bounded by $O((n + L)B^2 m^{O(C/\epsilon)})$.*

The following sections are structured as follows. In Section 5.4, we provide an overview to illustrate the proof of the sample complexity upper bound. The first step, given in Section 5.4, bounds the truncation error when approximating the quantum state function $\rho(x)$ using a truncated Fourier series. The second step, given in Section 5.4, bounds the generalization error for learning the Fourier approximation to the quantum state function $\rho(x)$. Then, in Section 5.4, we analyze the training and prediction time of the proposed classical machine learning model. These three sections establish Theorem 19. Finally, in Section 5.4, we use Theorem 19 and nice properties about ground states of Hamiltonians to prove Theorem 18.

Overview for sample complexity upper bound

The key intermediate step is to construct a truncated Fourier series of the quantum state function $\rho(x)$. The Fourier series of the matrix-valued function $\rho(x)$ is given as

$$\rho(x) = \sum_{k\in\mathbb{Z}^m} e^{\mathrm{i}\pi k\cdot x} A_k, \tag{5.48}$$

where A_k are matrix-valued Fourier coefficients

$$A_k = \frac{1}{2^m} \int_{[-1,1]^m} e^{-\mathrm{i}\pi k \cdot x} \rho(x) \mathrm{d}^m x. \tag{5.49}$$

We define the truncated Fourier series as

$$\rho_\Lambda(x) = \sum_{k \in \mathbb{Z}^m, \|k\|_2 \leq \Lambda} e^{\mathrm{i}\pi k \cdot x} A_k, \tag{5.50}$$

where $\Lambda > 0$ is a pre-specified cutoff value. Given an observable O that can be written as a sum of local observables $O = \sum_i O_i$ with $\sum_i \|O_i\|_\infty \leq B$ and $\mathbb{E}_{x\sim[-1,1]^m} \|\nabla_x \operatorname{tr}(O\rho(x))\|_2^2 \leq C$, the proof of Theorem 18 consists of two parts.

First, we bound the error between the truncated Fourier series $\rho_\Lambda(x)$ and the true quantum state function $\rho(x)$ in Section 5.4 giving

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |\operatorname{tr}(O\rho(x)) - \operatorname{tr}(O\rho_\Lambda(x))|^2 \leq O\left(\frac{C}{\Lambda^2}\right), \tag{5.51}$$

We choose the truncation $\Lambda = \Theta(\sqrt{C/\epsilon})$ such that the error between truncated Fourier series and the true quantum state function obeys

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |\operatorname{tr}(O\rho(x)) - \operatorname{tr}(O\rho_\Lambda(x))|^2 \leq \frac{\epsilon}{4}. \tag{5.52}$$

In the second part, we bound the error between the machine learning model $\hat{\sigma}(x)$ and the truncated Fourier series $\rho_\Lambda(x)$ in Section 5.4. With high probability over the randomness in generating the training data, we have

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |\operatorname{tr}(O\hat{\sigma}(x)) - \operatorname{tr}(O\rho_\Lambda(x))|^2 \leq \frac{B^2 m^{O(\Lambda^2)}}{N}. \tag{5.53}$$

The training data contains two sources of randomness, one from the sampling of x_ℓ and the other from the local randomized measurement to construct approximate classical representation for $\rho(x_\ell)$ that could be feed into the classical machine learning model. We choose the training data size

$$N = \frac{2B^2 m^{O(C/\epsilon)}}{\epsilon} \leq B^2 m^{O(C/\epsilon)+\log(1/\epsilon)+1} = B^2 m^{O(C/\epsilon)}, \tag{5.54}$$

such that the error between the machine learning model and the truncated Fourier series obeys

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |\operatorname{tr}(O\hat{\sigma}(x)) - \operatorname{tr}(O\rho_\Lambda(x))|^2 \leq \epsilon/4, \tag{5.55}$$

with high probability. The two parts can be combined by a triangle inequality to yield

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |\mathrm{tr}(O\hat{\sigma}(x)) - \mathrm{tr}\,(O\rho(x))|^2 \tag{5.56a}$$

$$\leq \left(\sqrt{\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |\mathrm{tr}(O\hat{\sigma}(x)) - \mathrm{tr}\,(O\rho_\Lambda(x))|^2} \right. \tag{5.56b}$$

$$\left. + \sqrt{\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |\mathrm{tr}(O\rho(x)) - \mathrm{tr}\,(O\rho_\Lambda(x))|^2}\right)^2 = \epsilon, \tag{5.56c}$$

with high probability over the randomness in the training data. This establishes the sample complexity upper bound for Theorem 18.

When the Hamiltonians $H(x)$ have spectral gap $\geq \Omega(1)$ in the domain $x \in [-1,1]^m$, for any observable $O = \sum_i O_i$ that can be written as a sum of local observables with $\sum_i \|O_i\|_\infty \leq B$, we have

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} \|\nabla_x \,\mathrm{tr}(O\rho(x))\|_2^2 \leq \mathcal{O}(B^2). \tag{5.57}$$

Hence, we can prove the sample complexity upper bound in Theorem 19 by utilizing Theorem 18 and the fact that $C = \mathcal{O}(B^2)$.

Controlling the truncation error

For a fixed observable O, we can define a function

$$f(x) = \mathrm{tr}(O\rho(x)) = \sum_{k\in\mathbb{Z}^m} e^{\mathrm{i}\pi k\cdot x}\,\mathrm{tr}(OA_k). \tag{5.58}$$

And the truncated Fourier series of the function $f(x)$ is given by

$$f_\Lambda(x) = \mathrm{tr}(O\rho_\Lambda(x)) = \rho_\Lambda(x) = \sum_{k\in\mathbb{Z}^m, \|k\|_2\leq\Lambda} e^{\mathrm{i}\pi k\cdot x}\,\mathrm{tr}(OA_k). \tag{5.59}$$

Lemma 16 (truncation error)**.** *Let*

$$f(x) = \sum_{k\in\mathbb{Z}^m} \alpha_k e^{\mathrm{i}\pi k\cdot x} \tag{5.60}$$

and

$$f_\Lambda(x) = \sum_{k\in\mathbb{Z}^m, \|k\|_2\leq\Lambda} \alpha_k e^{\mathrm{i}\pi k\cdot x}. \tag{5.61}$$

Then

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |f(x) - f_\Lambda(x)|^2 \leq \frac{1}{\pi^2\Lambda^2} \mathop{\mathbb{E}}_{x\sim[-1,1]^m} \|\nabla_x f(x)\|_2^2 \quad \textit{for any cutoff} \quad \Lambda > 0. \tag{5.62}$$

Proof. The claim follows from standard Harmonic analysis arguments. More precisely, we combine *orthogonality* ($\int_{[-1,1]^m} \mathrm{e}^{\mathrm{i}(\pi(k-k')x}\mathrm{d}^m x = \delta_{(k,k')}$) with the fact that the Fourier transform exchanges differentials ("momentum") with multiplications ("position"):

$$\nabla_x f(x) = \sum_{k\in\mathbb{Z}^m} \alpha_k \nabla_x \mathrm{e}^{\mathrm{i}\pi kx} = \mathrm{i}\pi \sum_{k\in\mathbb{Z}^m} \alpha_k k \mathrm{e}^{\mathrm{i}\pi kx}. \tag{5.63}$$

Use orthogonality to rewrite the truncation error as

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |f(x) - f_\Lambda(x)|^2 = \int_{[-1,1]^m} \Big| \sum_{k\in\mathbb{Z}^m:\|k\|>\Lambda} \mathrm{e}^{\mathrm{i}\pi kx}\alpha_k \Big|^2 \mathrm{d}^m x \tag{5.64a}$$

$$= \sum_{k:\|k\|_2>\Lambda} \sum_{k':\|k'\|_2>\Lambda} \Big(\int_{[-1,1]^m} \mathrm{e}^{\mathrm{i}\pi(k-k')x}\mathrm{d}^m x \Big) \overline{\alpha_k}\alpha_k \tag{5.64b}$$

$$= \sum_{k:\|k\|_2>\Lambda} |\alpha_k|^2 . \tag{5.64c}$$

Conversely, we use orthogonality and Rel. (5.63) to rephrase this upper bound. Let $\langle k', k\rangle$ be the Euclidean inner product between two vectors $k, k' \in \mathbb{Z}^m$. Then,

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} \|\nabla_x f(x)\|_2^2 = \int_{[-1,1]^m} \Big\| \sum_{k\in\mathbb{Z}^m} \pi k \mathrm{e}^{\mathrm{i}\pi kx}\alpha_k \Big\|_2^2 \mathrm{d}^m x \tag{5.65a}$$

$$= \sum_{k,k'\in\mathbb{Z}^m} \pi^2 \langle k', k\rangle \int_{[-1,1]^m} \mathrm{e}^{\mathrm{i}\pi(k-k')x}\mathrm{d}^m x \overline{\alpha_{k'}}\alpha_k \tag{5.65b}$$

$$=\pi^2 \sum_{k\in\mathbb{Z}^m} \langle k, k\rangle |\alpha_k|^2 = \pi^2 \sum_{k\in\mathbb{Z}^m} \|k\|_2^2 \,|\alpha_k|^2 . \tag{5.65c}$$

In words, the upper bound from Eq. (5.64c) can be rephrased as the Euclidean norm $\|\nabla_x f(x)\|_2^2$ of the vector $\nabla_x f(x)$. The advertised claim readily follows from comparing these two reformulations:

$$\sum_{k:\|k\|_2>\Lambda} |\alpha_k|^2 \leq \frac{1}{\Lambda^2} \sum_{k:\|k\|_2>\Lambda} \|k\|_2^2 \,|\alpha_k|^2 \leq \frac{1}{\pi^2\Lambda^2}\Big(\pi^2 \sum_{k\in\mathbb{Z}^m} \|k\|_2^2 \,|\alpha_k|^2 \Big). \tag{5.66}$$

This concludes the proof. □

Using Lemma 16 and the condition that $\mathbb{E}_{x\sim[-1,1]^m} \|\nabla_x \operatorname{tr}(O\rho(x))\|_2^2 \leq C$, we can obtain the desired inequality for bounding the truncation error,

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |\operatorname{tr}(O\rho(x)) - \operatorname{tr}(O\rho_\Lambda(x))|^2 \leq O\left(\frac{C}{\Lambda^2}\right). \tag{5.67}$$

Controlling generalization errors from using the training data

This section is devoted to a practical issue regarding training data based on classical shadows. Each label is obtained by performing a single-shot quantum measurement of a parametrized quantum state $\rho(x_i)$. We can use Eq. (5.3) to convert the single-shot outcome into $\sigma_1(\rho) = \bigotimes_{i=1}^n (3|s_i\rangle\langle s_i| - \mathbb{I})$. Such a classical shadow approximation reproduces the underlying state in expectation, i.e., $\mathbb{E}_{s_1,\ldots,s_n}[\sigma_1(\rho)] = \rho$. Recall that the training data $\mathcal{T} = \{x_\ell \to \sigma_1(\rho(x_\ell))\}_{\ell=1}^N$ consists of such classical shadow approximations. The machine learning model makes predictions based on a truncated Fourier kernel for future predictions. For new input $x \in [-1,1]^n$, we predict

$$\hat{\sigma}(x) = \frac{1}{N}\sum_{\ell=1}^{N} \kappa(x, x_\ell)\sigma_1\left(\rho(x_\ell)\right) \quad \text{with} \tag{5.68a}$$

$$\kappa(x, x_\ell) = \sum_{k\in\mathbb{Z}^m, \|k\|_2\leq\Lambda} \mathrm{e}^{\mathrm{i}\pi k\cdot(x-x_\ell)} = \sum_{k\in\mathbb{Z}^m, \|k\|_2\leq\Lambda} \cos(\pi k\cdot(x-x_\ell)). \tag{5.68b}$$

In the following, we will show that machine learning model $\hat{\sigma}(x)$ is equal to the truncated Fourier series $\rho_\Lambda(x)$ of the true target quantum state if we take the expectation over the training data, which includes the sampled inputs $x_1,\ldots,x_N$ and the randomized measurement outcomes $S_1(\rho(x_\ell)) = \{s_i\}_{i=1}^n$ for each input x_ℓ. Moreover, statistical flucutations due to shot noise will be small provided that we are interested in predicting an observable that decomposes nicely as a sum of local terms. These observations are the content of the following statement.

Lemma 17 (Statistical properties of the predicted quantum state $\hat{\sigma}(x)$)**.** *Let $\mathcal{T} = \{x_\ell \to \sigma_1(\rho(x_\ell))\}_{\ell=1}^N$ be a training set featuring uniformly random inputs $x_\ell \overset{unif}{\sim} [-1,1]^m$ and classical shadows of the associated quantum states as labels. Then, the machine learning model obeys*

$$\mathbb{E}_{\mathcal{T}}[\hat{\sigma}(x)] = \rho_\Lambda(x) = \sum_{k\in\mathbb{Z}^m, \|k\|_2\leq\Lambda} e^{\mathrm{i}\pi k\cdot x} A_k. \tag{5.69}$$

Moreover, suppose that an observable $O = \sum_i O_i$ decomposes into a sum of q-local terms. Then, with probability at least $1-\delta$, we have

$$\mathbb{E}_{x\sim[-1,1]^m} |\mathrm{tr}(O\hat{\sigma}(x)) - \mathrm{tr}\left(O\rho_\Lambda(x)\right)|^2 \tag{5.70}$$

$$\leq \frac{1}{N}9^q\Big(\sum_i \|O_i\|_\infty\Big)^2(2m+1)^{\Lambda^2}\left(\Lambda^2\log(2m+1) + \log\left(4/\delta\right)\right). \tag{5.71}$$

The advertised bound can be further streamlined if the observable locality q and confidence level δ are constant. Assuming $q, \delta = \mathcal{O}(1)$ ensures the following simplified scaling:

$$\underset{x\sim[-1,1]^m}{\mathbb{E}} |\mathrm{tr}(O\hat{\sigma}(x)) - \mathrm{tr}\,(O\rho_\Lambda(x))|^2 \tag{5.72}$$

$$= \mathcal{O}\left(\frac{1}{N}\left(\sum_i \|O_i\|\right)^2 (2m+1)^{\Lambda^2+\log(\Lambda^2)+1}\right) \tag{5.73}$$

$$= \frac{\left(\sum_i \|O_i\|\right)^2 m^{\mathcal{O}(\Lambda^2)}}{N}. \tag{5.74}$$

Using the condition that $\sum_i \|O_i\| \leq B$, we have

$$\underset{x\sim[-1,1]^m}{\mathbb{E}} |\mathrm{tr}(O\hat{\sigma}(x)) - \mathrm{tr}\,(O\rho_\Lambda(x))|^2 = \frac{B^2 m^{\mathcal{O}(\Lambda^2)}}{N}, \tag{5.75}$$

which controls the generalization error from quantum measurements. The argument is based on fundamental properties of classical shadows that have been reviewed in Appendix 5.1.

Proof of Lemma 17. We begin by condensing notation somewhat. Here, we only consider classical shadows of size $T = 1$. Hence, we may replace the superscript (t) by (x_ℓ) to succinctly keep track of classical input parameters. More precisely, we let $|s_i^{(x_\ell)}\rangle$ be the randomized Pauli measurement outcome for the i-th qubit when measuring the quantum state $\rho(x_\ell)$. The training data $\mathcal{T} = \{x_\ell \to \sigma_1(\rho(x_\ell))\}_{\ell=1}^N$ is determined by the following random variables

$$x_\ell \in [-1,1]^m, \qquad \text{for } \ell \in \{1, \ldots, N\}, \tag{5.76a}$$

$$s_i^{(x_\ell)} \in \{|0\rangle, |1\rangle, |+\rangle, |-\rangle, |\mathrm{i}+\rangle, |\mathrm{i}-\rangle\}, \quad \text{for } i \in \{1, \ldots, n\} \text{ and } \ell \in \{1, \ldots, N\}. \tag{5.76b}$$

The first claim is an immediate consequence of Eq. (5.7):

$$\underset{\mathcal{T}}{\mathbb{E}}[\hat{\sigma}(x)] = \frac{1}{N}\sum_{\ell=1}^N \underset{x_\ell\sim[-1,1]^m}{\mathbb{E}}\left[\kappa(x, x_\ell) \underset{s_1^{(x_\ell)},\ldots,s_n^{(x_\ell)}}{\mathbb{E}}[\sigma_1(\rho(x_\ell))]\right] \tag{5.77a}$$

$$= \frac{1}{N}\sum_{\ell=1}^N \underset{x_\ell\sim[-1,1]^m}{\mathbb{E}}[\kappa(x, x_\ell)\rho(x_\ell)] \tag{5.77b}$$

$$= \underset{x_1\sim[-1,1]^m}{\mathbb{E}}[\kappa(x, x_1)\rho(x_1)] \tag{5.77c}$$

$$= \sum_{k\in\mathbb{Z}^m, \|k\|_2\le\Lambda} e^{\mathrm{i}\pi k\cdot x} \mathop{\mathbb{E}}_{x_1\sim[-1,1]^m} \left[e^{-\mathrm{i}\pi k\cdot x_1}\rho(x_1)\right] \tag{5.77d}$$

$$= \sum_{k\in\mathbb{Z}^m, \|k\|_2\le\Lambda} e^{\mathrm{i}\pi k\cdot x} \frac{1}{2^m}\int_{[-1,1]^m} e^{-\mathrm{i}\pi k\cdot x_1}\rho(x)\mathrm{d}^m x_1 \tag{5.77e}$$

$$= \sum_{k\in\mathbb{Z}^m, \|k\|_2\le\Lambda} e^{\mathrm{i}\pi k\cdot x} A_k \tag{5.77f}$$

$$= \rho_\Lambda(x). \tag{5.77g}$$

Here, we have also used the fact that each x_ℓ is sampled independently and uniformly from $[-1,1]^m$.

The second result is contingent on the training data for predicting the ground state representation $\mathcal{T} = \{x_\ell \to \sigma_1(\rho(x_\ell))\}_{\ell=1}^N$. We begin with using the definitions of $\hat{\sigma}(x)$ (5.68) and $\rho_\Lambda(x)$ to rewrite the expression of interest as

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |\mathrm{tr}(O\hat{\sigma}(x)) - \mathrm{tr}\left(O\rho_\Lambda(x)\right)|^2$$

$$= \frac{1}{2^m}\int_{[-1,1]^m} \mathrm{d}^m x \left| \sum_{k\in\mathbb{Z}^m, \|k\|_2\le\Lambda} \mathrm{e}^{\mathrm{i}\pi k\cdot x}\left(\frac{1}{N}\sum_{\ell=1}^N \mathrm{e}^{-\mathrm{i}\pi k\cdot x_\ell}\,\mathrm{tr}\left(O\sigma_1\left(\rho(x_\ell)\right)\right) - \mathrm{tr}\left(OA_k\right)\right)\right|^2 \tag{5.78a}$$

$$= \sum_{k\in\mathbb{Z}^m, \|k\|_2\le\Lambda} \left|\frac{1}{N}\sum_{\ell=1}^N \mathrm{e}^{-\mathrm{i}\pi k\cdot x_\ell}\,\mathrm{tr}\left(O\sigma_1\left(\rho(x_\ell)\right)\right) - \mathrm{tr}\left(OA_k\right)\right|^2 , \tag{5.78b}$$

$$\equiv \sum_{k\in\mathbb{Z}^m, \|k\|_2\le\Lambda} D_{(k)}(\mathcal{T})^2 , \tag{5.78c}$$

where we have evaluated the Fourier integral over x and introduced shorthand notation $D_{(k)}(\mathcal{T})^2$ for each summand.

The next key step is to notice that each A_k is an expectation value over both the parameters and the shadows. Writing out A_k and expressing ρ in terms of shadows using Eq. (5.7),

$$\mathrm{tr}\left(OA_k\right) = \frac{1}{2^m}\int_{[-1,1]^m} \mathrm{e}^{-\mathrm{i}\pi k\cdot x_\ell}\,\mathrm{tr}\left(O\rho(x_\ell)\right)\mathrm{d}^m x_\ell \tag{5.79a}$$

$$= \mathop{\mathbb{E}}_{x_\ell\sim[-1,1]^m} \mathrm{e}^{-\mathrm{i}\pi k\cdot x_\ell}\,\mathrm{tr}\left(O\rho(x_\ell)\right) \tag{5.79b}$$

$$= \mathop{\mathbb{E}}_{x_\ell \text{ and } s_1^{(x_\ell)},\ldots,s_n^{(x_\ell)}} \mathrm{e}^{-\mathrm{i}\pi k\cdot x_\ell}\,\mathrm{tr}\left(O\sigma_1\left(\rho(x_\ell)\right)\right) . \tag{5.79c}$$

Plugging this back into the summand in Eq. (5.78c) yields

$$D_{(k)}(\mathcal{T})^2 \tag{5.80}$$

$$= \left| \frac{1}{N} \sum_{\ell=1}^{N} \mathrm{e}^{-\mathrm{i}\pi k \cdot x_\ell} \operatorname{tr}\left(O\sigma_1\left(\rho(x_\ell)\right)\right) - \mathop{\mathbb{E}}_{x_\ell \text{ and } s_1^{(x_\ell)},\ldots,s_n^{(x_\ell)}} \mathrm{e}^{-\mathrm{i}\pi k \cdot x_\ell} \operatorname{tr}\left(O\sigma_1\left(\rho(x_\ell)\right)\right) \right|^2 . \tag{5.81}$$

Therefore, each $D_{(k)}(\mathcal{T})^2$ is the (square-)deviation of an empirical average from the true expectation value A_k. Hence, we can use Hoeffding's inequality to bound it, provided that O is local and bounded. This may come as a surprise, as the empirical average samples only different parameters x_ℓ and not different shadows σ. However, the shadows depend on the parameters, so sampling only over the parameters turns out to be sufficient for a reasonable estimate.

In order to apply Hoeffding's inequality, we first have to make sure the expectation value is bounded. Recall that $O = \sum_i O_i$ decomposes nicely into a sum of q-body terms. More formally, $\operatorname{supp}(O_j) \subset \{1, \ldots, n\}$ contains at most q qubits. We also know trace and trace norm of each single-qubit contribution to $\sigma_1(\rho(x_\ell))$, $\operatorname{tr}\big(3|s_j^{(x_\ell)}\rangle\langle s_j^{(x_\ell)}| - \mathbb{I}\big) = 1$, and Eq. (5.4) asserts $\left\|3|s_j^{(x_\ell)}\rangle\langle s_j^{(x_\ell)}| - \mathbb{I}\right\|_1 = 3$. The matrix Hoelder inequality then implies, for every $x_\ell \in [-1, 1]^m$,

$$\left|\mathrm{e}^{\mathrm{i}\pi k \cdot x_\ell} \operatorname{tr}\left(O\sigma_1\left(\rho(x_\ell)\right)\right)\right| \leq \sum_i \left|\operatorname{tr}\left(O_i \sigma_1(\rho(x_\ell))\right)\right| \tag{5.82a}$$

$$= \sum_i \left|\operatorname{tr}\left(O_{A_i} \operatorname{tr}_{\neg A_i}\left(\sigma_1\left(\rho(x_\ell)\right)\right)\right)\right| \tag{5.82b}$$

$$\leq \sum_i \left\|O_{A_i}\right\|_\infty \left\|\operatorname{tr}_{\neg A_i}\left(\sigma_1\left(\rho(x_\ell)\right)\right)\right\|_1 \tag{5.82c}$$

$$= \sum_i \|O_i\|_\infty \prod_{j \in \operatorname{supp}(O_i)} \left\|3|s_j^{(x_\ell)}\rangle\langle s_j^{(x_\ell)}| - \mathbb{I}\right\|_1 \tag{5.82d}$$

$$= \sum_i \|O_i\|_\infty 3^{|\operatorname{supp}(O_j)|} \leq 3^q \sum_i \|O_i\|_\infty . \tag{5.82e}$$

Thus, the expectation value is bounded.

We are now ready to bound the likelihood of a large deviation $D_{(k)}(\mathcal{T})^2$. To recap, for each $k \in \mathbb{Z}^m$ obeying $\|k\|_2 \leq \Lambda$, we face a contribution that collects the (square-)deviation of a sum of *iid* and *bounded* random variables around their expectation value. These variables are complex, but one can analyze their real and imaginary parts separately and collect them into a complex version of Hoeffding's inequality:

$$\Pr\left[D_{(k)}(\mathcal{T})^2 \geq \tau^2\right] = \Pr\left[D_{(k)}(\mathcal{T}) \geq \tau\right] \tag{5.83a}$$

$$\leq 2\exp\left(-\frac{2N\tau^2}{9^q\left(\sum_i \|O_i\|\right)^2}\right) \quad \text{for all} \quad \tau > 0. \tag{5.83b}$$

This concentration bound connects training data size $N = |\mathcal{T}|$ with the size of a (fixed, but arbitrary) contribution to the expected deviation (5.78). For fixed magnitude τ and confidence δ, there is always a (finite) training data size $N = N(\tau, \delta)$ that ensures $D_{(k)}(\mathcal{T})^2 \leq \tau$ with probability at least $1 - \delta$. We can extend this reasoning to the entire sum in Eq. (5.78) by exploiting that Rel. (5.83) is independent of k, and the summation only ranges over finitely many terms. Introduce $K_\Lambda = |\{k \in \mathbb{Z}^m : \ \|k\|_2 \leq \Lambda\}|$ — the number of wave-vectors $k \in \mathbb{Z}^m$ whose Euclidean norm is bounded by Λ — and apply a union bound to conclude

$$\Pr\left[\sum_{k\in\mathbb{Z}^m, \|k\|_2\leq\Lambda} D_{(k)}(\mathcal{T})^2 \geq K_\Lambda \tau^2\right] \tag{5.84a}$$

$$\leq \Pr\left[\exists k \in \mathbb{Z}^m : \|k\|_2 \leq \Lambda, \ \text{s.t. } D_{(k)}(\mathcal{T})^2 \geq \tau^2\right] \tag{5.84b}$$

$$\leq \sum_{k\in\mathbb{Z}^m, \|k\|_2\leq\Lambda} \Pr\left[D_{(k)}(\mathcal{T})^2 \geq \tau^2\right] \tag{5.84c}$$

$$\leq 2K_\Lambda \exp\left(-\frac{2N\tau^2}{9^q \left(\sum_i \|O_i\|\right)^2}\right) \tag{5.84d}$$

for all $\tau > 0$. To finish the argument, we take guidance from Eq. (5.83). Fix a confidence level $\delta \in (0, 1)$ and set

$$\tau^2 = \frac{1}{N} 9^q \left(\sum_i \|O_i\|\right)^2 \log(2K_\Lambda/\delta) \tag{5.85}$$

to ensure

$$\Pr\left[\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |\mathrm{tr}(O\hat{\sigma}(x)) - \mathrm{tr}\,(O\rho_\Lambda(x))|^2 \geq K_\Lambda \tau^2\right] \leq \delta. \tag{5.86}$$

The advertised bound follows from inserting an explicit bound on the number of relevant wavevectors:

$$K_\Lambda = |\{k \in \mathbb{Z}^m : \ \|k\|_2 \leq \Lambda\}| \leq (2m+1)^{\Lambda^2}. \tag{5.87}$$

To see this, note that $\|k\|_2^2 = \sum_{i=1}^n |k_i|^2 \geq \sum_{i=1}^n |k_i| = \|k\|_1$, because $k_i \in \mathbb{Z}$. Conversely, every $k \in \mathbb{Z}^m$ that obeys $\|k\|_2 \leq \Lambda$ also obeys $\|k\|_1 \leq \Lambda^2$. Next, we enumerate all wave-vectors that obey the relaxed condition $\|k\|_1 \leq \Lambda^2$. To this end, we consider a simple process: select an index $i \in [m]$, and update the associated wave number by $+1$ (increment), 0 (do nothing) or -1 (decrement). Repeating this process a total of Λ^2 times allows us to generate no more than $(2m+1)^{\Lambda^2}$ different wavevectors. But, at the same time, every wave vector $k \in \mathbb{Z}^m$ that obeys $\|k\|_2 \leq \Lambda^2$ can be reached in this fashion. Hence, we conclude $K_\Lambda \leq \left|\left\{k \in \mathbb{Z}^m : \ \|k\|_1 \leq \Lambda^2\right\}\right| \leq (2m+1)^{\Lambda^2}$. □

Computational time for training and prediction

We have proposed a very simple prediction model that is based on approximating a truncated Fourier series (l_2-Dirichlet kernel). The training time is equivalent to loading the training data $\mathcal{T} = \{x_\ell \to \sigma_1(\rho(x_\ell))\}_{\ell=1}^{N}$. Only a single snapshot is provided for each sampled parameter x_ℓ (i.e., $T = 1$), so we relabel $s^{(t)} \longrightarrow s^{(x_\ell)}$. The training data is given by the collection of x_ℓ and shadows $\{s_i^{x_\ell}\}_{i=1}^{n}$, following Eq. (5.76). Therefore, one only needs

$$\mathcal{O}\left((n+m)N\right) = \mathcal{O}\left((n+m)B^2 m^{\mathcal{O}(C/\epsilon)}\right) = \mathcal{O}\left(nB^2 m^{\mathcal{O}(C/\epsilon)}\right) \quad \text{(training time)} \tag{5.88}$$

computational time to load the relevant data into a classical memory. Next, suppose that $O = \sum_{i=1}^{L} O_i$ is comprised of L q-local terms. Then, we can compute the associated expectation value for the predicted quantum state $\hat{\sigma}(x)$ by evaluating

$$\operatorname{tr}(O\hat{\sigma}(x)) = \frac{1}{N}\sum_{\ell=1}^{N}\sum_{i=1}^{L}\kappa(x, x_\ell)\operatorname{tr}(O_i\sigma_1(\rho(x_\ell))). \tag{5.89}$$

Recall that the kernel function is defined as

$$\kappa(x, x_\ell) = \sum_{k\in\mathbb{Z}^m, \|k\|_2\leq\Lambda} \mathrm{e}^{\mathrm{i}\pi k\cdot(x-x_\ell)} = \sum_{k\in\mathbb{Z}^m, \|k\|_2\leq\Lambda} \cos(\pi k\cdot(x-x_\ell)). \tag{5.90}$$

This can be computed in time $\mathcal{O}\left(K_\Lambda\right)$, where $K_\Lambda = |\{k \in \mathbb{Z}^m : \|k\|_2 \leq \Lambda\}| \leq (2m+1)^{\Lambda^2}$, according to Rel. (5.87) above. Because we have chosen $\Lambda = \Theta(\sqrt{C/\epsilon})$, the runtime to evaluate one kernel function is upper bounded by $m^{\mathcal{O}(C/\epsilon)}$.

On the other hand, the computation of each $\operatorname{tr}(O_j\sigma_1(\rho(x_\ell)))$ can be performed in constant time after storing the data in a classical memory. This is a consequence of the tensor product structure of $\sigma_1(\rho(x_\ell)) = \bigotimes_{i=1}^{n}\left(3|s_i^{(x_\ell)}\rangle\langle s_i^{(x_\ell)}| - \mathbb{I}\right)$ which ensures

$$\operatorname{tr}(O_j\sigma_1(\rho(x_\ell))) = \operatorname{tr}\left(O_j \bigotimes_{i\in\operatorname{supp}(O_j)}\left(3|s_i^{(x_\ell)}\rangle\langle s_i^{(x_\ell)}| - \mathbb{I}\right)\right), \tag{5.91}$$

where $\operatorname{supp}(O_j)$ is the set of qubits in $\{1, \ldots, n\}$ the local observable O_j acts on. Because $|\operatorname{supp}(O_j)| \leq q = \mathcal{O}(1)$, computing $\operatorname{tr}(O_j\sigma_1(\rho(x_\ell)))$ takes only constant time. However, the computation time does scale exponentially in $|\operatorname{supp}(O_j)|$. This can become a problem if $|\operatorname{supp}(O_j)|$ ceases to be a *small* constant. Putting everything together implies that $\operatorname{tr}(O\hat{\sigma}(x))$ can be computed in time (at most)

$$\mathcal{O}\left(NLm^{\mathcal{O}(C/\epsilon)}\right) = \mathcal{O}\left(LB^2 m^{\mathcal{O}(C/\epsilon)}\right) \quad \text{(prediction time).} \tag{5.92}$$

We conclude that both classical training time and prediction time for $\mathrm{tr}(O\hat{\sigma}(x))$ are upper bounded by

$$\mathcal{O}((n+L)B^2 m^{\mathcal{O}(C/\epsilon)}). \tag{5.93}$$

This concludes the proof of all statements given in Theorem 19.

Spectral gap implies smooth parametrizations

We attempt to deduce Theorem 18 from Theorem 19. The key step involves showing that the ground state $\rho(x)$ in a quantum phase of matter satisfies the following condition: For any observable $O = \sum_i O_i$ that can be written as a sum of local observables with $\sum_i \|O_i\|_\infty \leq B$, we have

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} \|\nabla_x \mathrm{tr}(O\rho(x))\|_2^2 \leq \mathcal{O}(B^2). \tag{5.94}$$

Then we can apply Theorem 19 with $C = \mathcal{O}(B^2)$ to derive Theorem 18.

The average gradient magnitude $\mathbb{E}_{x\sim[-1,1]^m} \|\nabla_x \mathrm{tr}(O\rho(x))\|_2^2$ depends on the observable O in question, but also on the parametrization $x \mapsto H(x) \mapsto \rho(x)$. This section provides a useful smoothness bound based on physically meaningful assumptions:

(a) *Physical system:* We consider n finite-dimensional quantum many-body systems that are arranged at locations, or sites, in a d-dimensional space, e.g., a spin chain ($d = 1$), a square lattice ($d = 2$), or a cubic lattice ($d = 3$). Unless specified otherwise, our big-$\mathcal{O}, \Omega, \Theta$ notation will be with respect to the thermodynamic limit $n \longrightarrow \infty$.

(b) *Hamiltonian:* $H(x)$ decomposes into a sum of geometrically local terms $H(x) = \sum_j h_j(x)$, each of which only acts on an $\mathcal{O}(1)$ number of sites in a ball of $\mathcal{O}(1)$ radius. Individual terms $h_j(x)$ obey $\left\|h_j(x)\right\|_\infty \leq 1$ and also have bounded directional derivative: $\left\|\partial h_j/\partial \hat{u}\right\|_\infty \leq 1$, where $\hat{u}$ is a unit vector in parameter space. However, each term $h_j(x)$ can depend on the entire input vector $x \in [-1,1]^m$.

(c) *Ground-state subspace:* We consider "the" ground state $\rho(x)$ for the Hamiltonian $H(x)$ to be defined as $\rho(x) = \lim_{\beta\to\infty} \mathrm{e}^{-\beta H(x)}/\mathrm{tr}(\mathrm{e}^{-\beta H(x)})$. This is equivalent to a uniform mixture over the eigenspace of $H(x)$ with the minimum eigenvalue.

(d) *Observable:* O decomposes into a sum of few-body observables $O = \sum_i O_i$, each of which only acts on an $\mathcal{O}(1)$ number of sites. Each few-body observables O_i can act on geometrically-nonlocal sites.

Assumptions (a)–(c) should be viewed as mild technical assumptions that are often met in practice. The main result of this section bounds the smoothness condition based on an additional requirement.

Lemma 18 (Spectral gap implies smoothness condition)**.** *Consider a class of local Hamiltonians*

$$\{H(x) : \ x \in [-1,1]^m\} \tag{5.95}$$

and an observable $O = \sum_i O_i$ that obey the technical requirements (a)–(c) above. Moreover, suppose that the spectral *gap of each $H(x)$ is lower bounded by (constant) $\gamma > \Omega(1)$. Then,*

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} \|\nabla_x \operatorname{tr}(O\rho(x))\|_2^2 \leq c_{\mathrm{all}} \Big(\sum_i \|O_i\|_\infty \Big)^2. \tag{5.96}$$

Here, $c_{\mathrm{all}} > 0$ is a constant that depends on spatial dimension d, spectral gap γ, as well as the Lieb-Robinson velocities.

The proof is based on combining two powerful techniques from quantum many body physics. Namely, Lieb-Robinson bounds (Elliott H. Lieb and Robinson, 1972) to exploit locality and the spectral flow formalism (Bachmann, Michalakis, et al., 2012), also referred to as quasi-adiabatic evolution or continuation (Matthew B Hastings and Wen, 2005; Osborne, 2007), to exploit the spectral gap.

Quasi-adiabatic continuation for gapped Hamiltonians (Matthew B Hastings and Wen, 2005; Osborne, 2007; Bachmann, Michalakis, et al., 2012): Given a quantum system satisfying the above assumptions (a)-(c), it is reasonable to expect that small changes in x only lead to small changes in the associated ground state $\rho(x)$. Spectral flow makes this intuition precise. Let the spectral gap of $H(x)$ be lower bounded by a constant γ over $[-1,1]^m$. Then, the directional derivative of an associated ground state, in the direction defined by the parameter unit vector $\hat{u}$, obeys

$$\frac{\partial \rho}{\partial \hat{u}}(x) = \mathrm{i}[D_{\hat{u}}(x), \rho(x)] \quad \text{where} \quad D_{\hat{u}}(x) = \int_{-\infty}^{\infty} W_\gamma(t) \mathrm{e}^{\mathrm{i}tH(x)} \frac{\partial H}{\partial \hat{u}}(x) \mathrm{e}^{-\mathrm{i}tH(x)} \mathrm{d}t. \tag{5.97}$$

Here, $W_\gamma(t)$ is a fast decaying weight function that obeys $\sup_t |W_\gamma(t)| = 1/2$ and only depends on the spectral gap. More precisely,

$$|W_\gamma(t)| \leq \begin{cases} \frac{1}{2} & 0 \leq \gamma|t| \leq \theta, \\ 35\mathrm{e}^2(\gamma|t|)^4 \mathrm{e}^{-\frac{2}{7}\frac{\gamma|t|}{\log(\gamma|t|)^2}} & \gamma|t| > \theta. \end{cases} \tag{5.98}$$

The constant θ is chosen to be the largest real solution of $35\mathrm{e}^2\theta^4\exp(-\frac{2}{7}\frac{\theta}{\log(\theta)^2}) = 1/2$.

Lieb-Robinson bounds for local Hamiltonians/observables (Elliott H. Lieb and Robinson, 1972; Matthew B Hastings, 2010): Let $\mathrm{supp}(X)$ denote the sites on which a many-body operator X acts nontrivially. Furthermore, for any two operators X_1, X_2, we define the distance $\Delta(X_1, X_2)$ to be the minimum distance between all pairs of sites acted on by X_1 and X_2, respectively, in the d-dimensional space. We also consider the number of local terms in a ball of radius r. For any operator X acting on a single site, this ball contains $O(r^d)$ local terms in d-dimensional space,

$$\sum_{j:\Delta(X,h_j)\leq r} 1 \leq b_d + c_d r^d, \forall r \geq 0, \tag{5.99}$$

where we recall the definition that $H = \sum_j h_j$ is a sum of local terms h_j. The bound on the number of local terms in a ball of radius r implies the existence of a Lieb-Robinson bound (Bravyi, Matthew B Hastings, and Verstraete, 2006; Matthew B Hastings, 2010). It states that for any two operator X_1, X_2 and any $t \in \mathbb{R}$, we have

$$\|[\exp(\mathrm{i}tH(x))X_1\exp(-\mathrm{i}tH(x)), X_2]\|_\infty \tag{5.100}$$

$$\leq c_{\mathrm{lr}}\|X_1\|_\infty\|X_2\|_\infty|\mathrm{supp}(X_1)|\mathrm{e}^{-a_{\mathrm{lr}}(\Delta(X_1,X_2)-v_{\mathrm{lr}}|t|)}, \tag{5.101}$$

for some constants $a_{\mathrm{lr}}, c_{\mathrm{lr}}, v_{\mathrm{lr}} = \Theta(1)$.

Apart from these two concepts, we will also need a bound on integrals of certain fast-decaying functions.

Lemma 19 (Lemma 2.5 in (Bachmann, Michalakis, et al., 2012))**.** *Fix $a > 0$ and define the function $u_a(x) = \exp(-ax/\log(x)^2)$ on the domain $x \in (1, \infty)$. Then,*

$$\int_t^\infty x^k u_a(x)\mathrm{d}x \leq \frac{2k+3}{a}t^{2k+2}u_a(t) \quad \textit{for all } t > \mathrm{e}^4 \textit{ and } k \in \mathbb{N} \tag{5.102}$$

that obey $2k + 2 \leq \frac{at}{\log(t)^2}$.

Proof of Lemma 18. Fix an input $x \in [-1, 1]^n$ and a unit vector $\hat{u} \in \mathbb{R}^n$ (direction). We may then rewrite the associated directional derivative of $\rho(x)$ in two ways, namely

$$\frac{\partial\rho}{\partial\hat{u}}(x) = \hat{u}\cdot\nabla_x\rho(x), \quad \text{and} \tag{5.103a}$$

$$\frac{\partial \rho}{\partial \hat{u}}(x) = -\mathrm{i}\,[D_{\hat{u}}(x), \rho(x)] \quad \text{with} \quad D_{\hat{u}}(x) = \int_{-\infty}^{\infty} \mathrm{d}t W_\gamma(t) \mathrm{e}^{\mathrm{i}tH(x)} \frac{\partial H}{\partial \hat{u}}(x) \mathrm{e}^{-\mathrm{i}tH(x)}. \tag{5.103b}$$

When evaluated on an observable O, this establishes the following correspondence:

$$\hat{u} \cdot \nabla_x \operatorname{tr}(O\rho(x)) = \operatorname{tr}\left(O\,[D_{\hat{u}}(x), \rho(x)]\right) = \operatorname{tr}\left([O, D_{\hat{u}}(x)]\,\rho(x)\right), \tag{5.104}$$

for any $\hat{u}$. Choosing $\hat{u} = \hat{u}(x, O) = \frac{\nabla_x \operatorname{tr}(O\rho(x))}{\|\nabla_x \operatorname{tr}(O\rho(x))\|_2}$ implies

$$\|\nabla_x \operatorname{tr}(O\rho(x))\|_2^2 = \left|\operatorname{tr}([O, D_{\hat{u}(x,O)}(x)]\rho(x))\right|^2. \tag{5.105}$$

The left hand side is the magnitude of steepest slope in a phase for the particular observable O. The average slope over the entire domain $[-1, 1]^m$ is thus given as

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} \|\nabla_x \operatorname{tr}(O\rho(x))\|_2^2 = \frac{1}{2^m} \int_{[-1,1]^m} \left|\operatorname{tr}([O, D_{\hat{u}(x,O)}(x)]\rho(x))\right|^2 \mathrm{d}^m x. \tag{5.106}$$

Intuitively, thermodynamic observables should not change too rapidly within a phase. Making this intuition precise will allow us to upper bound the average slope by a constant C.

We first expand $D_{\hat{u}}(x)$ and apply a triangle inequality to obtain

$$|\operatorname{tr}([O, D_{\hat{u}}(x)]\rho(x))| \leq \sum_i \int_{-\infty}^{\infty} W_\gamma(t) \sum_j \left\| \left[O_i, \mathrm{e}^{\mathrm{i}tH(x)} \frac{\partial h_j}{\partial \hat{u}}(x) \mathrm{e}^{-\mathrm{i}tH(x)} \right] \right\|_\infty \mathrm{d}t. \tag{5.107}$$

For fixed t, we can separate local Hamiltonian terms into two groups, defines using the constants in the Lieb-Robinson bound (5.100). The first group contains all terms h_j that obey $\Delta(O_i, h_j) \leq v_{\mathrm{lr}}|t|$. The second group contains all h_j that obey $\Delta(O_i, h_j) > v_{\mathrm{lr}}|t|$ instead. Equation (5.99) above provides a useful bound on the size of the first group. It contains at most $|\mathrm{supp}(O_i)|(b_d + c_d(v_{\mathrm{lr}}|t|)^d) \leq c_O(b_d + c_d(v_{\mathrm{lr}}|t|)^d)$ local terms h_j, for some constant $c_O \leq \mathrm{supp}(O)$. We can bound the summation over these terms using $\|[A, B]\|_\infty \leq 2\|A\|_\infty \|B\|_\infty$ to obtain

$$\sum_{j:\Delta(O_i,h_j)\leq v_{\mathrm{lr}}t} \left\| \left[O_i, \mathrm{e}^{\mathrm{i}tH(x)} \frac{\partial h_j}{\partial \hat{u}}(x) \mathrm{e}^{-\mathrm{i}tH(x)} \right] \right\|_\infty \tag{5.108a}$$

$$\leq c_O(b_d + c_d(v_{\mathrm{lr}}|t|)^d) \times 2\|O_i\|_\infty \left\| \frac{\partial h_j}{\partial \hat{u}} \right\|_\infty \tag{5.108b}$$

$$\leq 2c_O \|O_i\|_\infty (b_d + c_d(v_{\mathrm{lr}}|t|)^d). \tag{5.108c}$$

The second inequality follows from technical assumption (b): $\left\|\partial h_j / \partial \hat{u}\right\|_\infty \leq 1$.

The contributions from the second group can be controlled via the Lieb-Robinson bound from Eq. (5.100). For every h_j that obeys $\Delta(O_i, h_j) > v_{\mathrm{lr}}|t|$, we have

$$\begin{aligned}
&\left\| \left[O_i, \mathrm{e}^{\mathrm{i}tH(x)} \frac{\partial h_j}{\partial \hat{u}}(x) \mathrm{e}^{-\mathrm{i}tH(x)} \right] \right\|_\infty && \text{(5.109a)}\\
&\leq c_{\mathrm{lr}} \left\| O_i \right\|_\infty \left\| \partial h_j / \partial \hat{u} \right\|_\infty |\mathrm{supp}(h_j)| \mathrm{e}^{-a_{\mathrm{lr}}(\Delta(O_i,h_j)-v_{\mathrm{lr}}|t|)} && \text{(5.109b)}\\
&\leq c_{\mathrm{lr}} c_h \left\| O_i \right\|_\infty \mathrm{e}^{-a_{\mathrm{lr}}(\Delta(O_i,h_j)-v_{\mathrm{lr}}|t|)}. && \text{(5.109c)}
\end{aligned}$$

Reusing Eq. (5.99), we conclude that there are at most $|\mathrm{supp}(O_i)|(b_d + c_d(v_{\mathrm{lr}}|t| + r + 1)^d)$ local terms h_j with $\Delta(O_i, h_j) \in [v_{\mathrm{lr}}|t| + r, v_{\mathrm{lr}}|t| + r + 1]$. This ensures

$$\begin{aligned}
&\sum_{j:\Delta(O_i,h_j)>v_{\mathrm{lr}}|t|} \left\| \left[O_i, \mathrm{e}^{\mathrm{i}tH(x)} \frac{\partial h_j}{\partial \hat{u}}(x) \mathrm{e}^{-\mathrm{i}tH(x)} \right] \right\|_\infty \\
&\leq \sum_{r=0}^{\infty} \sum_{j:\Delta(O_i,h_j)\in[v_{\mathrm{lr}}|t|+r, v_{\mathrm{lr}}|t|+r+1]} \left\| \left[O_i, \mathrm{e}^{\mathrm{i}tH(x)} \frac{\partial h_j}{\partial \hat{u}}(x) \mathrm{e}^{-\mathrm{i}tH(x)} \right] \right\|_\infty && \text{(5.110a)}\\
&\leq \int_{r=0}^{\infty} \mathrm{d}r c_{\mathrm{lr}} c_h \left\| O_i \right\|_\infty \mathrm{e}^{-a_{\mathrm{lr}} r} \times \mathrm{supp}(O_i)(b_d + c_d(v_{\mathrm{lr}}|t| + r + 1)^d) && \text{(5.110b)}\\
&\leq c_{\mathrm{lr}} c_h c_O \left\| O_i \right\|_\infty \int_{r=0}^{\infty} \mathrm{d}r \mathrm{e}^{-a_{\mathrm{lr}} r} (b_d + c_d(v_{\mathrm{lr}}|t| + r + 1)^d) && \text{(5.110c)}\\
&\leq c_{\mathrm{lr}} c_h c_O \left\| O_i \right\|_\infty \left(\frac{b_d}{a_{\mathrm{lr}}} + c_d \sum_{p=0}^{d} \frac{d!}{p! a_{\mathrm{lr}}^{d-p+1}} (v_{\mathrm{lr}}|t| + 1)^p \right). && \text{(5.110d)}
\end{aligned}$$

We can now combine the two bounds into a single statement:

$$\sum_j \left\| \left[O_i, \mathrm{e}^{\mathrm{i}tH(x)} \frac{\partial h_j}{\partial \hat{u}}(x) \mathrm{e}^{-\mathrm{i}tH(x)} \right] \right\|_\infty \leq \left\| O_i \right\|_\infty \sum_{p=0}^{d} C_p |t|^p. \qquad (5.111)$$

Here, we have implicitly defined a new set of constants C_p that depend on the constants $c_O, c_h, c_{\mathrm{lr}}, c_d, a_{\mathrm{lr}}, v_{\mathrm{lr}}, d$ that had already featured before. Plugging the above into Eq. (5.107) and substituting the spectral flow weight function W (5.98) for its absolute value allows us to bound the maximum slope of $\mathrm{tr}(O\rho(x))$ when the Hamiltonian moves from $H(x)$ to $H(x + \mathrm{d}\hat{u})$. Indeed,

$$|\,\mathrm{tr}([O, D_{\hat{u}}(x)]\rho(x))| \leq \Big(\sum_i \left\| O_i \right\|_\infty \Big) \sum_{p=0}^{d} C_p \int_{-\infty}^{\infty} |W_\gamma(t)||t|^p \mathrm{d}t \,. \qquad (5.112)$$

To bound the resulting integral, we recall that $W_\gamma(t)$ obeys $\sup_t |W_\gamma(t)| = 1/2$, define $t^* = \max(\mathrm{e}^4, 7(d+5), \theta)/\gamma$, and split up the integration into two parts, $t \in [-t^*, t^*]$ and $t \notin [-t^*, t^*]$. Symmetry then ensures

$$\int_{-\infty}^{\infty} \mathrm{d}t |W_\gamma(t)||t|^p \leq \frac{1}{2} \int_{-t^*}^{t^*} \mathrm{d}t |t|^p + 2 \int_{t^*}^{\infty} \mathrm{d}t \; 35 \mathrm{e}^2 (\gamma t)^4 \mathrm{e}^{-\frac{2}{7} \frac{\gamma t}{\log(\gamma t)^2}} t^p \qquad (5.113a)$$

$$= \int_0^{t^*} \mathrm{d}t\, t^p + 70\mathrm{e}^2\gamma^{-p-1} \int_{x=\gamma t^*}^{\infty} \mathrm{d}x\, x^{p+4} \mathrm{e}^{-\frac{2}{7}\frac{x}{\log(x)^2}}. \tag{5.113b}$$

The first integral is straightforward, and the second integral can be bounded using Lemma 19. Set $a = 2/7$, $k = p + 4$ and note that we have chosen t^* such that all assumptions are valid. Applying Lemma 19 ensures

$$\int_{-\infty}^{\infty} \mathrm{d}t |W_\gamma(t)||t|^p \mathrm{d}t \le \frac{|t^*|^{p+1}}{p+1} + 70\mathrm{e}^2\gamma^{-p-1}\frac{2k+3}{a}(\gamma t^*)^{2k+2}\mathrm{e}^{-\frac{2\gamma t^*}{7\log(\gamma t^*)^2}} \tag{5.114a}$$

$$= \frac{|t^*|^{p+1}}{p+1} + 35\mathrm{e}^2\gamma^{-p-1}7(2p+11)(\gamma t^*)^{2p+10}\mathrm{e}^{-\frac{2\gamma t^*}{7\log(\gamma t^*)^2}}, \tag{5.114b}$$

for any integer $0 \le p \le d$. Inserting these bounds into the sum (5.111) implies

$$|\operatorname{tr}([O, D_{\hat{u}}(x)]\rho(x))| \tag{5.115}$$

$$\le \Big(\sum_i \|O_i\|_\infty\Big) \sum_{p=0}^{d} C_p \left(\frac{|t^*|^{p+1}}{p+1} + 35\mathrm{e}^2\gamma^{-p-1}7(2p+11)(\gamma t^*)^{2p+10}\mathrm{e}^{-\frac{2\gamma t^*}{7\log(\gamma t^*)^2}}\right). \tag{5.116}$$

Recall that $t^* = \max(\mathrm{e}^4, 7(d+5), \theta)/\gamma$ is a constant that only depends on d and γ, and the C_p's are also constants that depend on on $c_O, c_h, c_{\mathrm{lr}}, c_d, a_{\mathrm{lr}}, v_{\mathrm{lr}}, d$. We may subsume all of these constant contributions in a new constant c_{all} and conclude

$$|\operatorname{tr}([O, D_{\hat{u}}(x)]\rho(x))| \le c_{\mathrm{all}}\Big(\sum_i \|O_i\|_\infty\Big). \tag{5.117}$$

Inserting this upper bound into Eq. (5.106) completes the proof of Lemma 18. □

5.5 Sample complexity lower bound for predicting ground states

This section establishes an information-theoretic lower bound for the task of predicting ground state approximations. It highlights that, without further assumptions on the Hamiltonians, the training data size required in Theorem 19 is tight.

Theorem 20. *Fix a prediction error tolerance ϵ, a number m of parameters, as well as constants $C, B > 0$ such that $C/(9\epsilon) \le m^{0.99}$. Consider a quantum ML model that learns from quantum data $\{x_\ell \to \rho(x_\ell)\}_{\ell=1}^N$ of size N to generate ground state predictions $\hat{\sigma}(x)$, where $x \in [-1,1]^m$. Suppose the quantum ML model can achieve*

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} |\operatorname{tr}(O\hat{\sigma}(x)) - \operatorname{tr}(O\rho(x))|^2 \le \epsilon, \tag{5.118}$$

with high probability, for every class of Hamiltonians $H(x)$ and for every observable O given as a sum of local observables $\sum_i O_i$ that obey

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} \|\nabla_x \operatorname{tr}(O\rho(x))\|_2^2 \leq C \qquad \textit{(smoothness condition)}, \tag{5.119a}$$

$$\sum_i \|O_i\| \leq B \qquad \textit{(bounded norm)}. \tag{5.119b}$$

Then, the (quantum) training data size must obey

$$N \geq B^2 m^{\Omega(C/\epsilon)}/\log(B). \tag{5.120}$$

This is also a lower bound on quantum computational time associated with the quantum ML model.

The assumption $C/(9\epsilon) \leq m^{0.99}$ is required for technical reasons outlined below. It is equivalent to demanding that the prediction error tolerance is large enough compared to the inverse of m, i.e., $\epsilon \geq C/(9m^{0.99})$. If the quantum ML model can achieve an even smaller prediction error, such that $\mathbb{E}_{x\sim[-1,1]^m} |\operatorname{tr}(O\hat{\sigma}(x)) - \operatorname{tr}(O\rho(x))|^2 < C/(9m^{0.99})$, then we choose $\epsilon = C/(9m^{0.99})$. For such a choice of ϵ, the training data size lower bound becomes $N \geq B^2 m^{\Omega(m^{0.99})}/\log(B)$, which is exponential in $m^{0.99}$. Hence, in all cases, we need ϵ to be a constant for any (quantum or classical) machine learning algorithm to obtain a sample complexity that scales polynomially in m.

We prove Theorem 20 by means of an information-theoretic analysis. Conceptually, it resembles arguments developed in prior work (Huang, Richard Kueng, and Preskill, 2021) (sample complexity lower bound for general quantum machine learning models). Section 5.5 formulates a learning problem that involves predicting ground state properties of a certain class of Hamiltonians. Subsequently, Section 5.5 incorporates a hypothetical (quantum ML) solution to this learning problem as a decoding procedure in a communication protocol. Information-theoretic bottlenecks then beget fundamental restrictions on the sample complexity of any ML model that solves the learning problem, see Section 5.5.

Learning problem formulation

We consider a family of single-qubit Hamiltonians, i.e. $n = 1$, that is parametrized by m degrees of freedom. We first map $x \in [-1, 1]^m$ to a real number by evaluating a truncated Fourier series f_a. Fix a cutoff $\Lambda = \sqrt{C/(9\epsilon)}$ and let

$$K_\Lambda = \left|\left\{k \in \mathbb{Z}^m : \|k\|_2 \leq \Lambda = \sqrt{C/(9\epsilon)}\right\}\right| \tag{5.121}$$

denote the number of n-dimensional wave-vectors with Euclidean norm at most Λ. We equip each of these wave vectors k with a sign $a_k \in \{\pm 1\}$ and define the function

$$f_a(x) = \sqrt{\frac{9\epsilon}{K_\Lambda}} \sum_{k \in \mathbb{Z}^m, \|k\|_2 \leq \Lambda} a_k \cos\left(\pi k \cdot x\right), \quad \text{where} \quad a \in \{\pm 1\}^{K_\Lambda}, \tag{5.122}$$

subsumes all sign choices involved. We use this function to define a single-qubit Hamiltonian. For Pauli matrices X and Z, we set

$$H_a(x) = \exp\left(+\tfrac{\mathrm{i}}{2} \arcsin\left(f_a(x)/B\right) X\right) (-Z) \exp\left(-\tfrac{\mathrm{i}}{2} \arcsin\left(f_a(x)/B\right) X\right), \tag{5.123}$$

where B is a constant reflecting the size of the target observable, see Eq. (5.119b). To summarize, each choice of $a \in \{\pm 1\}^{K_\Lambda}$ yields an entire class of single-qubit Hamiltonians $H_a(x)$ that is parametrized by m-dimensional inputs $x \in [-1, 1]^m$. These stylized Hamiltonians are simple enough to compute their (nondegenerate) ground state explicitly:

$$\rho_a(x) = |\psi_a(x)\rangle\langle\psi_a(x)| \quad \text{with} \quad |\psi_a(x)\rangle = \begin{pmatrix} \cos\left(\frac{1}{2}\arcsin(f_a(x)/B)\right) \\ \mathrm{i}\sin\left(\frac{1}{2}\arcsin(f_a(x)/B)\right) \end{pmatrix} \in \mathbb{C}^2. \tag{5.124}$$

Finally, we fix the single-qubit observable O to be a scaled version of Pauli Y. Setting $O = BY$ yields a 1-local observable. And, more importantly,

$$\operatorname{tr}\left(O\rho_a(x)\right) = B\langle\psi_a|Y|\psi_a\rangle = B\left(-\mathrm{i}\overline{\langle 0|\psi_a(x)\rangle}\langle 1|\psi_a(x)\rangle + \mathrm{i}\langle 0|\psi_a(x)\rangle\overline{\langle 1|\psi_a(x)\rangle}\right) \tag{5.125a}$$

$$= 2B\cos\left(\tfrac{1}{2}\arcsin\left(f_a(x)/B\right)\right)\sin\left(\tfrac{1}{2}\arcsin\left(f_a(x)/B\right)\right) \tag{5.125b}$$

$$= B\sin\left(\arcsin\left(f_a(x)/B\right)\right) = f_a(x). \tag{5.125c}$$

By construction, the expectation value $\operatorname{tr}\left(O\rho_a(x)\right)$ exactly reproduces the function $f_a(x)$ defined in Eq. (5.122). Being able to accurately predict it will be equivalent to accurately learning this function – regardless of the underlying sign parameter $a \in \{\pm 1\}^{K_\Lambda}$.

To complete the formulation of the learning problem, we recall that the training parameters are sampled from the uniform distribution over the hypercube, Unif $[-1, 1]^m$, and that we will evaluate the expectation $\mathbb{E}$ over x with respect to this distribution from now on. This choice of distribution implies a nice closed-form expression for the average squared distance of two functions f_a, f_b. For $a, b \in \{\pm 1\}^{K_\Lambda}$,

$$\mathop{\mathbb{E}}_{x} |f_a(x) - f_b(x)|^2 \tag{5.126a}$$

$$= \frac{9\epsilon}{K_\Lambda} \sum_{k,l \in \mathbb{Z}^m, \|k\|_2, \|l\|_2 \leq \Lambda} (a_k - b_k)(a_l - b_l) \int_{[-1,1]^m} \cos(\pi k \cdot x) \cos(\pi l \cdot x) \, \mathrm{d}^m x \tag{5.126b}$$

$$= \frac{9\epsilon}{K_\Lambda} \sum_{k \in \mathbb{Z}^m, \|k\|_2 \leq \Lambda} (a_k - b_k)^2 \tag{5.126c}$$

$$= \frac{9\epsilon}{K_\Lambda} \sum_{k \in \mathbb{Z}^m, \|k\|_2 \leq \Lambda} 4 \times \mathbf{1}\{a_k \neq b_k\} \tag{5.126d}$$

$$= \frac{36\epsilon}{K_\Lambda} d_H(a, b) \, , \tag{5.126e}$$

where we have used orthonormality of the Fourier basis $\cos(\pi k \cdot x)$, and $d_H(a, b) = \sum_k \mathbf{1}\{a_k \neq b_k\}$ is the *Hamming distance* on $\{\pm 1\}^{K_\Lambda}$.

We conclude this expository section by examining whether the construction fulfills the requirement stated in the theorem and presenting a technical lemma. First of all, we have

$$\|O\| = B \, \|Y\| = B, \tag{5.127}$$

which satisfies the bounded norm constraint in Eq. (5.119b). Furthermore, we can use the orthonormality of $\cos(\pi k \cdot t)$ to find that

$$\mathop{\mathbb{E}}_{x \sim [-1,1]^m} \|\nabla_x \operatorname{tr}(O \rho_a(x))\|_2^2 = \frac{9\epsilon}{K_\Lambda} \sum_{k \in \mathbb{Z}^m : \|k\|_2 \leq \Lambda} \|k\|_2^2 \, |a_k|^2 \leq \frac{9\epsilon}{K_\Lambda} K_\Lambda \Lambda^2 = C. \tag{5.128}$$

Thus the smoothness condition in Eq. (5.119a) is also satisfied. Now, we turn our attention to the ground state (5.124). The following technical lemma exposes the function $f_a(x)/B$ directly in the amplitudes of ground states.

Lemma 20. *Let $|\psi_a(x)\rangle$ be the ground state of H_a defined in Eq.* (5.124)*. Then, we have*

$$\rho_a(x) = |\psi_a(x)\rangle\langle\psi_a(x)| = \frac{1}{2} \begin{pmatrix} 1 + \sqrt{1 - (f_a(x)/B)^2} & -\mathrm{i} f_a(x)/B \\ \mathrm{i} f_a(x)/B & 1 - \sqrt{1 - (f_a(x)/B)^2} \end{pmatrix}. \tag{5.129}$$

Proof. The proof is based on double-angle and half-angle trigonometric identities.

Suppressing x dependence, the first diagonal entry becomes

$$\langle 0|\rho_a|0\rangle = |\langle 0|\psi_a\rangle|^2 = \cos^2\left(\tfrac{1}{2}\arcsin\left(f_a/B\right)\right) = \tfrac{1}{2}\left(1+\cos\left(\arcsin\left(f_a/B\right)\right)\right) \tag{5.130a}$$

$$= \tfrac{1}{2}\left(1+\sqrt{1-\sin^2\left(\arcsin\left(f_a/B\right)\right)}\right) = \tfrac{1}{2}\left(1+\sqrt{1-\left(f_a/B\right)^2}\right), \tag{5.130b}$$

and normalization implies that $\langle 1|\rho_a|1\rangle = 1 - \langle 0|\rho_a|0\rangle$. The off-diagonal entries are given by

$$\overline{\langle 1|\rho_a|0\rangle} = \langle 0|\rho_a|1\rangle = \langle 0|\psi_a\rangle\langle\psi_a|1\rangle \tag{5.131a}$$

$$= -\mathrm{i}\cos\left(\tfrac{1}{2}\arcsin\left(f_a/B\right)\right)\sin\left(\tfrac{1}{2}\arcsin\left(f_a/B\right)\right) \tag{5.131b}$$

$$= -\tfrac{\mathrm{i}}{2}\sin\left(\arcsin\left(f_a/B\right)\right) = -\tfrac{\mathrm{i}}{2}f_a/B\,. \tag{5.131c}$$

This concludes the proof. □

Communication protocol

Consider the learning problem introduced in the previous section. Suppose that a quantum ML model can use training data $\mathcal{T} = \{x_\ell, \rho_a(x_\ell)\}_{\ell=1}^N$ to learn a function $f^Q(x)$ that (on average) predicts $\mathrm{tr}\left(O\rho_a(x)\right) = f_a(x)$ for a particular unknown $a \in \{\pm 1\}^{K_\Lambda}$, up to some accuracy ϵ,

$$\mathop{\mathbb{E}}_x \left|f^Q(x) - f_a(x)\right|^2 \leq \epsilon\,. \tag{5.132}$$

Such a model will not fare as well in estimating the expectation value associated with $b \neq a$, whenever b is sufficiently far away from a. Using the triangle inequality and Eq. (5.126),

$$\mathop{\mathbb{E}}_x \left|f^Q(x) - f_b(x)\right|^2 \geq \mathop{\mathbb{E}}_x \left|f_a(x) - f_b(x)\right|^2 - \mathop{\mathbb{E}}_x \left|f^Q(x) - f_a(x)\right|^2 \tag{5.133}$$

$$\geq \frac{36\epsilon}{K_\Lambda} d_H(a,b) - \epsilon\,. \tag{5.134}$$

The model's accuracy significantly worsens at $d_H(a,b) > K_\Lambda/18$, where we recall $K_\Lambda = |\{k \in \mathbb{Z}^m : \|k\|_2 \leq \Lambda\}|$ from Eq. (5.121). In other words, a good quantum ML model would allow us to use training data $\mathcal{T}$ in order to recover the underlying parameter $a \in \{\pm 1\}^{K_\Lambda}$ up to resolution $K_\Lambda/18$ in Hamming distance.

We can use this assertion as an effective decoding procedure in a two-way communication protocol involving Alice and Bob. To accommodate imperfect resolution, Alice and Bob agree on a dictionary of sign vectors $\left\{a^{(1)}, \ldots, a^{(M)}\right\} \subset \{\pm 1\}^{K_\Lambda}$

whose pairwise Hamming distance is large enough: $d_H(a_i, a_j) > K_\Lambda/18$ for all $i \neq j$. Let M denote the cardinality of this dictionary. Alice and Bob use this dictionary and the ML procedure to transmit integers up to size M over a quantum channel. Alice samples an integer $j \in \{1, \dots, M\}$ and sets $a = a^{(j)} \in \{\pm 1\}^{K_\Lambda}$. Subsequently, she uses a to generate (quantum) training data $\mathcal{T} = \{(x_\ell, \rho_a(x_\ell))\}_{\ell=1}^{N}$ with $x_1, \dots, x_N \sim \text{Unif}\,[-1,1]^m$ which she passes on to Bob. Subsequently, Bob uses $\mathcal{T}$ to train a quantum ML model to predict the underlying function $\operatorname{tr}(O\rho_a(x)) = f_a(x)$. By checking $\mathbb{E}_x \left| f_{\bar{a}}(x) - f^Q(x) \right|^2 \leq \epsilon$ for every possible dictionary element $\bar{a}$, he will retrieve the correct message with high probability, i.e., $\bar{a} = a$.

This is a protocol that conveys classical information via a quantum dataset. It is subject to fundamental constraints from information theory. These will allow us to deduce a lower bound on the required training data size $N = |\mathcal{T}|$. An important figure of merit in this argument is the cardinality M of the dictionary. That is, the number of different integers that can be communicated. The larger M, the more powerful the communication protocol, and following result, sometimes attributed to Gilbert and Varshamov (Gilbert, 1952), is a lower bound on how many bits one can “pack” into the space of L-bit strings while maintaining the required distance.

Lemma 21 (Lemma 5.12 in (Rigollet and Hütter, 2015))**.** *There exists a dictionary $\left\{a^{(1)}, \dots, a^{(M)}\right\} \in \{\pm 1\}^{K_\Lambda}$ of cardinality $M \geq \lfloor \exp(K_\Lambda/32) \rfloor$ that achieves $d_H\left(a^{(i)}, a^{(j)}\right) \geq K_\Lambda/4$ whenever $i \neq j$.*

Information-theoretic analysis

Let us now take a closer look at the communication protocol introduced above by bounding the correlation between Alice’s original randomly chosen message a and Bob’s decoded signal $\bar{a}$. Up to now, we have stablished the following. Per the bound in Lemma 21, the dictionary of available a’s can be chosen to be rather large: $M = \lfloor \exp(K_\Lambda/32) \rfloor$. Moreover, the existence of a good quantum ML procedure, in the sense of Proposition 20, ensures that $\bar{a} = a$ with high probability.

Correlations between Alice’s and Bob’s variables are quantified by the (classical) mutual information

$$I(a : \bar{a}) \geq \Omega\left(\log(M)\right) = \Omega(K_\Lambda), \tag{5.135}$$

which we have bounded from below using Fano’s inequality (B. Yu, 1997). Our task now is to provide an upper bound on $I(a : \bar{a})$, in terms of N, B and ϵ, in order to relate those parameters to K_Λ and obtain the desired result in Theorem 20.

Since the parameters $x_1, \dots, x_N$ are sampled independently from a, we have $I(a : x_1, \dots, x_N) = 0$ and $a|_{x_1,\dots,x_N} = a$. Therefore, we can upper bound the mutual information as follows,

$$\begin{aligned} I\left(a : \bar{a}\right) &\le I\left(a : \bar{a}, x_1, \dots, x_N\right) && (5.136a)\\ &= I\left(a : x_1, \dots, x_N\right) + I\left(a : \bar{a} | x_1, \dots, x_N\right) && (5.136b)\\ &= I\left(a : \bar{a} | x_1, \dots, x_N\right) && (5.136c)\\ &= \mathop{\mathbb{E}}_{x_1,\dots,x_N} I\left(a|_{x_1,\dots,x_N} : \bar{a}|_{x_1,\dots,x_N}\right) && (5.136d)\\ &= \mathop{\mathbb{E}}_{x_1,\dots,x_N} I\left(a : \bar{a}|_{x_1,\dots,x_N}\right) \ , && (5.136e) \end{aligned}$$

where $Q|_x$ denotes the random variable Q conditioned on the random variable x.

Next, recall that Bob reconstructs the classical $\bar{a}$ by performing quantum operations on the training data $\mathcal{T} = \{(x_\ell, \rho_a(x_\ell)\}_{\ell=1}^N$. For each instance of randomly chosen parameters $x_1, \dots, x_N \sim \mathrm{Unif}[-1, 1]^m$, Bob performs a quantum measurement on the state $\bigotimes_{\ell=1}^N \rho_a(x_\ell)$ and uses the measurement outcomes to reconstruct $\bar{a}$. Bob's procedure is equivalent to performing the quantum ML algorithm that we have been promised in Sec. 5.5. Thus we can use Holevo's theorem (A. S. Holevo, 1973)[Wilde, 2013, Sec. 11.6.1] to write

$$I\left(a : \bar{a}|_{x_1,\dots,x_N}\right) \le \chi\left(a : \bigotimes_{\ell=1}^{N} \rho_a\left(x_\ell\right)\Big|_{x_1,\dots,x_N}\right) \ , \qquad (5.137)$$

where the Holevo information χ quantifies correlations between a random variable z and a quantum state ρ_z,

$$\chi(z : \rho_z) = S\left(\mathop{\mathbb{E}}_{z} \rho_z\right) - \mathop{\mathbb{E}}_{z} S(\rho_z) \ , \qquad (5.138)$$

and $S(\rho) = -\operatorname{tr}(\rho \log \rho)$ is the von Neumann entropy. In other words, for each instance of parameters, the correlation between a and $\bar{a}$ is bounded by the Holevo information of Bob's ensemble of quantum states.

Next, we use the subadditivity of von Neumann entropy, $S(\mathbb{E}_z \rho_z \otimes \sigma_z) \le S(\mathbb{E}_z \rho_z) + S(\mathbb{E}_z \sigma_z)$, and the additivity of entropy for independent systems, $S(\rho \otimes \sigma) = S(\rho) + S(\sigma)$, to obtain

$$\chi\left(a : \bigotimes_{\ell=1}^{N} \rho_a\left(x_\ell\right)\Big|_{x_1,\dots,x_N}\right) \le \sum_{\ell=1}^{N} \chi\left(a : \rho_a\left(x_\ell\right)\big|_{x_1,\cdots,x_N}\right) \ . \qquad (5.139)$$

Plugging Eqs. (5.137) and (5.139) into Eq. (5.136) and using the fact that $\rho_a(x_\ell)$ is independent to $x_{\ell'}$ for any $\ell' \neq \ell$, we obtain

$$I\left(a : \bar{a}\right) \leq \sum_{\ell=1}^{N} \mathop{\mathbb{E}}_{x_1,\cdots,x_N} \chi\left(a : \rho_a\left(x_\ell\right)\big|_{x_1,\cdots,x_N}\right) \tag{5.140a}$$

$$= \sum_{\ell=1}^{N} \mathop{\mathbb{E}}_{x_\ell} \chi\left(a : \rho_a\left(x_\ell\right)\big|_{x_\ell}\right) \tag{5.140b}$$

$$= N \mathop{\mathbb{E}}_{x} \chi\left(a : \rho_a\left(x\right)\right) \ . \tag{5.140c}$$

The last equality follows from the fact that each $(x_\ell, \rho_a(x_\ell))$ is generated independently and in an identical fashion for all $\ell = 1, \ldots, N$.

We have thus reduced the problem of bounding the correlations between classical variables a and $\bar{a}$ to that of bounding the Holevo information of the ensemble of states ρ_a — a much simpler problem because ρ_a is a two-by-two matrix. In Lemma 22 at the end of section, we obtain the bound

$$\mathop{\mathbb{E}}_{x} \chi\left(a : \rho_a(x)\right) \leq \frac{9\epsilon}{4B^2} \log\left(\frac{4\mathrm{e}B^2}{9\epsilon}\right) \ . \tag{5.141}$$

Using this bound, the first claim in Theorem 20 readily follows, provided that we are allowed to choose

$$K_\Lambda = m^{\Omega(C/\epsilon)} \ . \tag{5.142}$$

This assumption, combined with Eqs. (5.135-5.141) ensures that

$$N \frac{9\epsilon}{4B^2} \log\left(\frac{4\mathrm{e}B^2}{9\epsilon}\right) \geq \Omega(K_\Lambda) = m^{\Omega(C/\epsilon)} \quad \text{which implies} \quad N \geq \frac{B^2 m^{\Omega(C/\epsilon)}}{\log(B)}. \tag{5.143}$$

Because the quantum ML has to process quantum training data of size $N \geq \frac{B^2 m^{\Omega(C/\epsilon)}}{\log(B)}$, the runtime of the quantum ML has to be lower bounded by that amount as well.

Let us now verify the assumption (5.142) on the number of Fourier modes K_Λ available for estimating the quantum state. While we have already determined that $K_\Lambda \leq m^{O(C/\epsilon)}$ in Eq. (5.87), here we need a lower bound. We utilize the assumption that $C/(9\epsilon) \leq m^{0.99}$, which implies $\lfloor C/(9\epsilon) \rfloor \leq m^{0.99}$. To establish Eq. (5.142), we restrict our attention to binary wavevectors $k \in \{0,1\}^m$, such that the number of ones is exactly equal to $\lfloor C/(9\epsilon) \rfloor$. Clearly, every such wavevector obeys $\|k\|_2 \leq \sqrt{C/(9\epsilon)}$, so the number of such wavevectors lower bounds K_Λ. This observation, along with some combinatorics, yields

$$K_\Lambda \geq \left| \left\{ k \in \{0,1\}^m : \sum_{j=1}^{m} k_j = \lfloor C/(9\epsilon) \rfloor \right\} \right| \tag{5.144a}$$

$$= \binom{m}{\lfloor C/(9\epsilon)\rfloor} \geq \frac{m^{\lfloor C/(9\epsilon)\rfloor}}{(\lfloor C/9\epsilon\rfloor)^{\lfloor C/(9\epsilon)\rfloor}} \tag{5.144b}$$

$$= m^{\lfloor C/(9\epsilon)\rfloor - (\lfloor C/9\epsilon\rfloor)\log(\lfloor C/(9\epsilon)\rfloor)/\log(m)} \geq m^{0.01\lfloor C/(9\epsilon)\rfloor} = m^{\Omega(C/\epsilon)}. \tag{5.144c}$$

We now prove the upper bound (5.141) on the mutual information. It follows from analyzing the ground state representations provided by Lemma 20.

Lemma 22. *The learning problem from Section 5.5 is set up to obey*

$$\mathop{\mathbb{E}}_{x\sim\mathrm{Unif}[-1,1]^m} \chi\left(a : \rho_a(x)\right) \leq \frac{9\epsilon}{4B^2}\log\left(\frac{4\mathrm{e}B^2}{9\epsilon}\right). \tag{5.145}$$

Proof. Using the definition (5.138) of the Holevo information and the von Neumann entropy,

$$\mathop{\mathbb{E}}_{x\sim\mathrm{Unif}[-1,1]^m} \chi\left(a : \rho_a(x)\right) \tag{5.146a}$$

$$= \mathop{\mathbb{E}}_{x}\left[\mathop{\mathbb{E}}_{a}[\mathrm{tr}(\rho_a(x)\log\rho_a(x))] - \mathrm{tr}\left(\left(\mathop{\mathbb{E}}_{a}\rho_a(x)\right)\log\left(\mathop{\mathbb{E}}_{a}\rho_a(x)\right)\right)\right] \tag{5.146b}$$

$$= -\mathop{\mathbb{E}}_{x}\mathrm{tr}\left[\left(\mathop{\mathbb{E}}_{a}\rho_a(x)\right)\log\left(\mathop{\mathbb{E}}_{a}\rho_a(x)\right)\right]\ . \tag{5.146c}$$

The second equality follows from the fact that $\rho_a(x)$ is a pure state, so we have $\mathrm{tr}(\rho_a(x)\log\rho_a(x)) = 0$. We also consider $\mathbb{E}_x$ to be $\mathbb{E}_{x\sim\mathrm{Unif}[-1,1]^m}$. Recalling Lemma 20 yields

$$\mathop{\mathbb{E}}_{a}\rho_a(x) = \frac{1}{2}\mathop{\mathbb{E}}_{a}\begin{pmatrix} 1+\sqrt{1-(f_a(x)/B)^2} & -\mathrm{i}f_a(x)/B \\ \mathrm{i}f_a(x)/B & 1-\sqrt{1-(f_a(x)/B)^2}\end{pmatrix}. \tag{5.147}$$

The eigenvalues $\lambda_\pm$ of $\mathbb{E}_a\,\rho_a(x)$, like those of any two-by-two matrix, can be expressed in terms of the trace and determinant. Using the formula for the eigenvalues and evaluating the trace and determinant yield

$$\lambda_\pm = \frac{1}{2}\mathrm{tr}\left[\mathop{\mathbb{E}}_{a}\rho_a(x)\right] \pm \frac{1}{2}\sqrt{\left(\mathrm{tr}\left[\mathop{\mathbb{E}}_{a}\rho_a(x)\right]\right)^2 - 4\det\left[\mathop{\mathbb{E}}_{a}\rho_a(x)\right]} \tag{5.148a}$$

$$= \frac{1}{2} \pm \frac{1}{2}\sqrt{\left(\mathop{\mathbb{E}}_{a}f_a(x)\right)^2/B^2 + \left(\mathop{\mathbb{E}}_{a}\sqrt{1-f_a(x)^2/B^2}\right)^2}\ . \tag{5.148b}$$

We will use following lower bound for λ_+

$$\lambda_+ \geq \frac{1}{2} + \frac{1}{2}\mathop{\mathbb{E}}_{a}\sqrt{1-f_a(x)^2/B^2} \tag{5.149a}$$

$$\geq \frac{1}{2} + \frac{1}{2}(1 - \mathop{\mathbb{E}}_{a}f_a(x)^2/B^2) \tag{5.149b}$$

$$= 1 - \frac{1}{2} \mathop{\mathbb{E}}_{a} f_a(x)^2 / B^2 \geq \frac{1}{2}. \tag{5.149c}$$

The first inequality follows from dropping the term $(\mathbb{E}_a f_a(x))^2/B^2$. The second inequality follows from the fact that $\sqrt{1-z} \geq 1 - z$ for all $z \in [0, 1]$.

We now proceed to bounding the von Neumann entropy of $\mathbb{E}_a \rho_a(x)$,

$$\begin{aligned} -\operatorname{tr}\left(\left(\mathop{\mathbb{E}}_{a} \rho_a(x)\right) \log\left(\mathop{\mathbb{E}}_{a} \rho_a(x)\right)\right) &= -\lambda_+ \log \lambda_+ - \lambda_- \log \lambda_- = H(\lambda_+) && (5.150a) \\ &\leq H\left(1 - \frac{1}{2} \mathop{\mathbb{E}}_{a} f_a(x)^2 / B^2\right) && (5.150b) \\ &= H\left(\frac{1}{2} \mathop{\mathbb{E}}_{a} f_a(x)^2 / B^2\right) && (5.150c) \\ &\equiv H(g(x)) \leq g(x) \log(\mathrm{e}/g(x)), && (5.150d) \end{aligned}$$

where $H(x) = -x \log x - (1-x)\log(1-x)$ is the binary entropy, and $g(x) = \frac{1}{2} \mathbb{E}_a f_a(x)^2/B^2$. The first inequality follows from the fact that $H(x) \leq H(x')$ for all $1/2 \leq x' \leq x$. Going back to Eq. (5.146),

$$\begin{aligned} \mathop{\mathbb{E}}_{x} \chi\,(a : \rho_a(x) | x) &= -\mathop{\mathbb{E}}_{x} \operatorname{tr}\left[\left(\mathop{\mathbb{E}}_{a} \rho_a(x)\right) \log\left(\mathop{\mathbb{E}}_{a} \rho_a(x)\right)\right] && (5.151a) \\ &\leq \mathop{\mathbb{E}}_{x}[g(x) \log(\mathrm{e}/g(x))] && (5.151b) \\ &\leq \left(\mathop{\mathbb{E}}_{x} g(x)\right) \log\left(\frac{\mathrm{e}}{\mathbb{E}_x\, g(x)}\right) && (5.151c) \\ &= \frac{\mathbb{E}_{x,a} f_a(x)^2}{2B^2} \log\left(\frac{2\mathrm{e}B^2}{\mathbb{E}_{x,a} f_a(x)^2}\right). && (5.151d) \end{aligned}$$

The first inequality follows from Eq. (5.150). The second inequality follows Jensen's inequality and the fact that $z \log(\mathrm{e}/z)$ is concave for all $z \geq 0$. Orthogonality of the $\cos(\pi k \cdot x)$ terms in f_a (5.122) yields

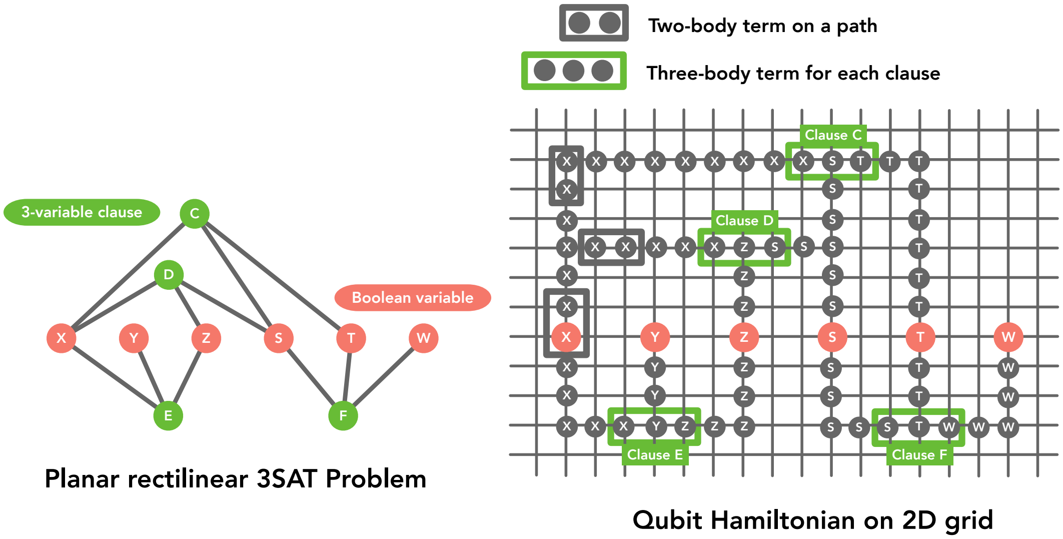
$$\mathop{\mathbb{E}}_{x,a} f_a(x)^2 = \frac{1}{2} \times \frac{9\epsilon}{L} \sum_{k \in \mathbb{Z}^m, \|k\|_2 \leq \Lambda} \mathop{\mathbb{E}}_{a} |a_k|^2 = \frac{9\epsilon}{2}. \tag{5.152}$$

Plugging the above into Eq. (5.151d), we obtain the advertised bound. □

5.6 Hardness for non-ML algorithms to predict ground state properties

NP-hardness for estimating one-body observables in the ground state of 2D Hamiltonians

We begin by showing that the task of estimating one-body observables in the ground state of any smooth class of two-dimensional Hamiltonians with a constant spectral gap is NP-hard. The task is hard even if we consider the computation to yield a small error averaged over the smooth class of Hamiltonians.



Planar rectilinear 3SAT Problem

Qubit Hamiltonian on 2D grid

Figure 5.2: Reduction of planar rectilinear 3SAT (LEFT) to a qubit Hamiltonian on a 2D grid (RIGHT). Each pair (i, j) of nearby grid points on a path (originating from variable X, Y, Z, S, T, W) contains a two-body local term $-Z_i Z_j$ (illustrated by boxes with gray stroke). Each clause (C, D, E, F) corresponds to a three-body local term that imposes the Boolean constraint, e.g., $X \vee Z \vee S$ would correspond to $-\sum_{x,z,s\in\{0,1\}} \mathbb{1}[x \vee z \vee s = 1] \cdot |x\rangle\langle x| \otimes |z\rangle\langle z| \otimes |s\rangle\langle s|$. Every empty grid point (the irrelevant qubits) contain a single body term $-Z_i$.

Proposition 9 (Detailed restatement of Proposition 8; a variant of Lemma 1.4 in (Abrahamsen, 2020))**.** *Consider a randomized polynomial-time classical algorithm $\mathcal{A}(H, i, r)$ whose inputs are the description of a Hamiltonian H, an index i that enumerates the qubits in the Hamiltonian, and a random bit string r. Suppose that for any smooth class of Hamiltonians on a two-dimensional grid with a spectral gap ≥ 1 and a unique ground state,*

$$H(x) = \sum_a h_a(x) \;\text{ with }\; \rho(x) \,:\, \textit{the ground state of } H(x), \tag{5.153}$$

where $x \in [-1, 1]^m$ is a parameter and $h_a(x)$ is a three-qubit geometrically-local observable, and for each one-body Pauli-Z observable Z_i, the randomized classical algorithm $\mathcal{A}$ outputs $\mathcal{A}(H, i, r)$ that approximates $\mathrm{tr}(Z_i \rho(x))$ up to an average error $\mathbb{E}_{x\sim[-1,1]^m} |\mathbb{E}_r\, \mathcal{A}(H, i, r) - \mathrm{tr}(Z_i \rho(x))| \leq 1/4$. Then RP = NP.

Proof. From standard results in complexity theory (Lichtenstein, 1982; Knuth and Raghunathan, 1992; L. Valiant and V. Vazirani, 1986), it is known that if there is a randomized polynomial-time classical algorithm that can find the solution for any planar rectilinear 3SAT problem with a unique solution with probability at least 1/2, then RP = NP. (RP, also known as Randomized Polynomial Time, is the class of decision problems such that there is a polynomial-time randomized classical algorithm that outputs YES with probability at least 1/2 when the correct answer is YES, and outputs NO with probability one when the correct answer is NO. RP is contained in BPP, the class of decision problems that can be solved efficiently by a randomized classical computer.) The planar rectilinear 3SAT problem is a constrained version of 3SAT, where all the Boolean variables $x_1, \ldots, x_n$ are vertices on the x-axis and all the clauses containing three variables are vertices that lie above or below the x-axis. Each clause is connected by an edge to each of the the variables that the clause contains. The vertices and the edges form a planar graph; see Figure 5.2 (left) for an illustration.

We can embed such a planar graph in a two-dimensional grid with a single qubit on each grid point; see Figure 5.2 (right) for an illustration of the embedding. First, we distinguish between the variable vertices and the clause vertices in the planar graph. Variable vertices lie on the x-axis of the two-dimensional qubit grid, and clause vertices lie above or below the x-axis. Edges of the planar graph become embedded paths on the the 2D grid connecting clause vertices to variable vertices. Because the original graph is planar, we can ensure that the paths corresponding to each edge on the planar graph do not overlap (except when they terminate at the same variable) by choosing a large enough spacing between the variable vertices on the x-axis. For each path on the 2D grid, we add a $-Z_i Z_j$ term to the Hamiltonian for every pair of nearest neighbors along the path. The two body $-Z_i Z_j$ term ensures that, in the unique ground state, the qubits on the path must be either all $|0\rangle$'s or all $|1\rangle$'s. Then, for every clause vertex on the planar graph, we add a three-body geometrically-local term (diagonal in the Z-basis) to the Hamiltonian enforcing that in the ground state the endpoints of the three corresponding paths satisfy the Boolean constraint of the corresponding clause. For example, the Boolean clause $X \vee Z \vee S$ would correspond to the three body local term $-\sum_{x,z,s\in\{0,1\}} \mathbb{1}[x \vee z \vee s = 1] \cdot |x\rangle\langle x| \otimes |z\rangle\langle z| \otimes |s\rangle\langle s|$, where $\mathbb{1}[A]$ is 1 if A is true and 0 otherwise. The qubits on paths are called the "relevant" qubits, and the rest of the qubits are called "irrelevant." We add a $-Z_i$ term to the Hamiltonian for all the irrelevant qubits, fixing these qubits to be $|0\rangle$ in the ground state.

Moreover, the eigenstates of the Hamiltonian are computational basis states, because all the local terms are diagonal in the Z-basis. We can also see that there are no terms connecting the relevant and irrelevant qubits, hence the ground state space of the constructed Hamiltonian must be the tensor product of the ground state space for the relevant qubits and the ground state space for the irrelevant qubits. The unique ground state for the irrelevant qubits is the all-zero state $|0\rangle \otimes \cdots \otimes |0\rangle$ due to the $-Z_i$ term. Because the original planar rectilinear 3SAT problem has a unique solution, the ground state for the relevant qubits is also unique. We denote the ground state by $|b\rangle\langle b|, b \in \{0,1\}^n$, where n is the total number of qubits in the two dimensional grid. In this ground state, all variable vertices are fixed at the values that solve the 3SAT problem. Furthermore, because all eigenvalues of the Hamiltonian are integers, the spectral gap is at least one.

Let us define $\sum_a h_a$ to be the Hamiltonian constructed from a planar rectilinear 3SAT problem. Note that h_a is diagonal in the Z-basis and acts on at most three geometrically-local qubits. We define a trivial class of two-dimensional Hamiltonians with a spectral gap ≥ 1,

$$H(x) = \sum_a h_a(x) = \sum_a h_a = H, \tag{5.154}$$

where $x \in [-1,1]^m$ is the parameter, and $H(x)$ does not depend on x. Let $\rho(x)$ be the unique ground state of $H(x)$. We have $\rho(x) = |b\rangle\langle b|, b \in \{0,1\}^n$, where b encodes the solution to the planar rectilinear 3SAT problem.

We apply the randomized classical algorithm to provide estimates for all the expectation values of Pauli-Z observables in the ground state space $\rho(x)$ of $H(x)$. Let $\mathcal{A}$ be the randomized classical algorithm. By the assumption that the randomized classical algorithm could output an estimate of $\mathrm{tr}(Z_i\rho(x))$ up to an additive error $1/4$ averaged uniformly over $x \in [-1,1]^m$, we have

$$\mathop{\mathbb{E}}_{x\sim[-1,1]^m} \left| \mathop{\mathbb{E}}_{r} \mathcal{A}(H(x), i, r) - \mathrm{tr}(Z_i\rho(x)) \right| \leq 1/4, \quad \forall i = 1, \ldots, n. \tag{5.155}$$

Using Jensen's inequality, we have the following bound,

$$\left| \mathop{\mathbb{E}}_{x\sim[-1,1]^m} \mathop{\mathbb{E}}_{r} \mathcal{A}(H(x), i, r) - \mathop{\mathbb{E}}_{x\sim[-1,1]^m} \mathrm{tr}(Z_i\rho(x)) \right| \leq 1/4, \quad \forall i = 1, \ldots, n. \tag{5.156}$$

We can see that $\mathbb{E}_{x\sim[-1,1]^m} \mathrm{tr}(Z_i\rho(x)) = \langle b_i| Z_i |b_i\rangle$, where b_i is the i-th bit in the n-bit string b that encodes the solution to the planar rectilinear 3SAT problem.

We sample random x uniformly from $[-1,1]^m$ and sample the random string r, obtaining the output value $\mathcal{A}(H(x), i, r)$ using the randomized classical algorithm $\mathcal{A}$. As a result of the above analysis, by sampling $O(\log(n))$ times and computing the average over the output $\mathcal{A}(H(x), i, r)$, we can obtain an estimate for $\langle b_i| Z_i |b_i\rangle$ up to an additive error $1/2$ with probability at least $1 - \frac{1}{2n}$, where n is the total number of qubits in the 2D grid. Because $\langle b_i| Z_i |b_i\rangle \in \{-1, 1\}$, an estimate for $\langle b_i| Z_i |b_i\rangle$ up to an additive error $1/2$ allows us to obtain $b_i \in \{0, 1\}$. Using the union bound, with probability at least $1/2$, we can obtain b_i, for all $i = 1, \ldots, n$. This implies that we can obtain the bit string b with probability at least $1/2$. Hence, we can use the randomized classical algorithm $\mathcal{A}$ to find the unique solution for the planar rectilinear 3SAT problem with probability at least $1/2$. Therefore, RP=NP if such an algorithm exists. □

We remark that a similar argument still applies if we replace the constant Hamiltonian $H(x)$ considered above by a suitably chosen class of Hamiltonians $\{H(x) = H : x \in [-1, 1]^m\}$ with nontrivial dependence on x. For example, we can consider $H(x) = \sum_a h_a(x) = \sum_a (U_1(x_1) \otimes \ldots \otimes U_n(x_n))\, h_a\, (U_1(x_1) \otimes \ldots \otimes U_n(x_n))^\dagger$, where n is the number of qubits in the Hamiltonian, $U_i(x_i) = \exp(-\mathrm{i}(\pi/4)X_i x_i)$ is a single-qubit rotation, X_i is the Pauli-X matrix on the i-th qubit, and $m = n$. It is not hard to see that $h_a(x)$ still acts on at most three geometrically-local qubits. Furthermore, one can adapt the proof to show that predicting ground state properties averaged over x for this nonconstant class of Hamiltonians is still hard.

Computational hardness for a class of Hamiltonians based on factoring

Theorem 17 and Proposition 9 together implies that an NP-hard problem could be solved by performing single-qubit measurements on a modest number of copies of the ground state of a two-dimensional local Hamiltonian, and then performing an efficient classical computation with the measurement outcomes as input. We may therefore conclude that, in hard instances, the preparation of the ground state is itself an NP-hard task. Because we do not expect any NP-hard task to be performed efficiently in the physics lab, or in any other physically realizable process, Proposition 9 does not usefully characterize the computational power of data under realistic conditions.

In contrast, it is reasonable to expect that simple measurements performed on quantum states that are efficiently prepared by quantum computers, combined with classical processing, suffice for solving computational problems that are beyond the reach of classical processing alone. Indeed, proposals for using variation quantum eigensolvers to study quantum chemistry and materials (Peruzzo et al., 2014; Jarrod R McClean, Romero, et al., 2016) are motivated by this expectation. Theorem 17 is of potential practical interest for a class of Hamiltonians $\{H(x)\}$ such that the ground state of $H(x)$ can be prepared efficiently by a feasible quantum process, yet cannot be efficiently prepared classically.

The rest of this subsection outlines a stylized example that illustrates this idea. Leveraging the efficient quantum algorithm for factoring large numbers, and the assumption that factoring is classically hard, we construct a smooth class of local Hamiltonians whose ground states are easy to prepare quantumly, such that expectation values of one-local observables can be learned efficiently from training data, yet are hard to learn by any classical procedure without access to data.

The first step is to construct two-dimensional Hamiltonians such that computing expectation values of one-local observables in the ground state is equivalent to solving a factoring problem. This can be done by noting a series of well-known facts in complexity theory.

1. The following task is expected to be hard for classical computers. Given a n-bit number R guaranteed to be a product of two prime numbers $p < q$, find p, q. When R is large, all known classical algorithms scale superpolynomially with n. Solving this problem suffices to break the RSA encryption (Rivest, A. Shamir, and Adleman, 1978).

2. We can represent p, q using at most $2n$ binary variables (bits), and we can write down a propositional formula for these $2n$ variables, which corresponds to a logical circuit that computes the multiplication of p, q and checks if the product equals R. The propositional formula can be written without any additional Boolean variable. This yields a SAT problem with $2n$ Boolean variables whose unique solution is equal to the two prime numbers p, q.

3. A SAT problem with a unique solution can be efficiently mapped to a 3SAT problem with a unique solution; see (Kozen, 1992).

4. A 3SAT problem with a unique solution can be efficiently mapped to a planar rectilinear 3SAT problem with a unique solution; see (Lichtenstein, 1982; Knuth and Raghunathan, 1992).

5. A planar rectilinear 3SAT problem with a unique solution can be efficiently mapped to a two-dimensional 3-local Hamiltonian with a spectral gap of one and a unique ground state, such that estimating one-local observables in the

ground state of the Hamiltonian to a constant error with a constant probability is sufficient to find the unique solution for the planar rectilinear 3SAT problem; see the proof of Proposition 9.

We now focus on any smooth class of two-dimensional Hamiltonians $H^{\mathrm{RSA}}(x)$ with a constant spectral gap such that there exists $x^{\mathrm{RSA}} \in [-1, 1]^m$ such that $H^{\mathrm{RSA}}(x^{\mathrm{RSA}})$ can be written as a two-dimensional Hamiltonian that is mapped from a factoring problem. We refer to such a class of Hamiltonians as an RSA-based two-dimensional gapped Hamiltonian class.

For any RSA-based Hamiltonian class H^{RSA}, we can efficiently obtain the training data from a quantum experiment. We first prepare the ground state for $H^{\mathrm{RSA}}(x^{\mathrm{RSA}})$ by applying Shor's algorithm. Then we can adiabatically evolve the ground state for $H^{\mathrm{RSA}}(x^{\mathrm{RSA}})$ to obtain the ground state for $H^{\mathrm{RSA}}(x), \forall x \in [-1, 1]$ due to the existence of a constant spectral gap (Aharonov, Van Dam, et al., 2008; Wan and I. Kim, 2020). Hence, according to Theorem 17, for any RSA-based Hamiltonian class, a classical ML algorithm trained from data obtained in quantum experiments can efficiently predict expectation values of one-local observables in the ground state. In contrast, a classical algorithm that does not learn from training data is unable to efficiently estimate 1-body observables in the ground state, assuming that classical computers cannot break RSA encryption.

5.7 Classifying quantum phases of matter

Classifying quantum phases of matter is another important application of machine learning to physics. We will consider this classification problem in the case where quantum states are succinctly represented by their classical shadows. For simplicity, we consider the classification of two phases (denoted A and B), but the analysis naturally generalizes to classifying any number of phases.

ML algorithms

We envision training a classical ML with classical shadows, where each classical shadow carries a label y indicating whether it represents a quantum state ρ from phase A ($y(\rho) = 1$) or phase B ($y(\rho) = -1$). We want to show that a suitably chosen classical ML can learn to efficiently predict the phase for new classical shadows beyond those encountered during training. Following a strategy which is standard in learning theory, we consider a classical ML that maps each classical shadow to a corresponding feature vector in a high-dimensional feature space, and then attempts to find a hyperplane that separates feature vectors in the A phase from feature vectors in the B phase. The learning is efficient if the geometry of the feature space is efficiently computable, and if the feature map is sufficiently expressive. Thus, our task is to construct a feature map with the desired properties.

In the simpler task of classifying symmetry-breaking phases, there is typically a local order parameter $O = \sum_i O_i$ given as a sum of r-body observables for some $r > 0$ that satisfies

$$\mathrm{tr}(O\rho) \geq 1, \forall \rho \in \text{phase } A, \qquad \mathrm{tr}(O\rho) \leq -1, \forall \rho \in \text{phase } B. \tag{5.157}$$

Under this criterion, the classification function may be chosen to be $y(\rho) = \mathrm{sign}(\mathrm{tr}(O\rho))$. Hence, classifying symmetry-breaking phases can be achieved by finding a hyperplane that separates the two phases in the high-dimensional feature space that subsumes all r-body reduced density matrices of the quantum state ρ. The feature vector consisting of all r-body reduced density matrices of the quantum state ρ can be accurately reconstructed from the classical shadow representation $S_T(\rho)$ when T is sufficiently large.

Finding a suitable choice of hyperplane in the feature space can be cast as a convex optimization problem known as the soft-margin support vector machine, discussed in more detail in Appendix 5.9. With a sufficient amount of training data, the hyperplane found by the classical ML model will generalize so the phase $y(\rho)$ can be predicted accurately for a previously unseen quantum state ρ. The classical ML is not merely a black box; it exhibits the order parameter (encoded by the hyperplane), guiding physicists toward a deeper understanding of the phase structure.

For more exotic quantum phases of matter, such as topologically ordered phases, the above classical ML model no longer suffices. The topological phase of a state is invariant under a constant-depth quantum circuit, and a phase containing the product state $|0\rangle^{\otimes n}$ is called the trivial phase. Using these notions, we can prove that no observable — not even one that acts on the entire system — can be used to distinguish between two topological phases. The proof, given in Appendix 5.8, uses the observation that random single-qubit unitaries can confuse any global or local order parameter.

Proposition 10. *Consider two distinct topological phases A and B (one of the phases could be the trivial phase). No observable O exists such that*

$$\mathrm{tr}(O\rho) > 0, \forall \rho \in \textit{phase } A, \qquad \mathrm{tr}(O\rho) \leq 0, \forall \rho \in \textit{phase } B. \tag{5.158}$$

While this proposition implies that no linear function $\mathrm{tr}(O\rho)$ can be used to classify topologically ordered phases, it does not exclude nonlinear functions, such as quadratic functions $\mathrm{tr}(O\rho \otimes \rho)$, degree-$d$ polynomials $\mathrm{tr}(O\rho^{\otimes d})$ and more general analytic functions. For example, it is known that the topological entanglement entropy (A. Y. Kitaev and Preskill, 2006; Levin and Wen, 2006), a nonlinear function of ρ, can be used to classify a wide variety of topologically ordered phases. For this purpose, it suffices to consider a subsystem whose size is large compared to the correlation length of the state, but is independent of the total size of the system. The correlation length in the ground state of a local Hamiltonian increases when the spectral gap between the ground state and the first excited state becomes smaller (Matthew B Hastings and Koma, 2006). On the other hand, a linear function on the full system will fail even with constant correlation length.

To learn nonlinear functions, we need a more expressive ML model. For this purpose we devise a powerful feature map that takes the classical shadow $S_T(\rho)$ of the quantum state ρ to a feature vector that includes arbitrarily-large r-body reduced density matrices, as well as an arbitrarily-high-degree polynomial expansion,

$$\begin{aligned}&\phi^{(\mathrm{shadow})}(S_T(\rho)) \qquad (5.159)\\ &= \lim_{D,R\to\infty} \bigoplus_{d=0}^{D} \sqrt{\frac{\tau^d}{d!}} \left(\bigoplus_{r=0}^{R} \sqrt{\frac{1}{r!}\left(\frac{\gamma}{n}\right)^r} \bigoplus_{i_1=1}^{n} \ldots \bigoplus_{i_r=1}^{n} \mathrm{vec}\left[\frac{1}{T} \sum_{t=1}^{T} \bigotimes_{\ell=1}^{r} \sigma_{i_\ell}^{(t)} \right] \right)^{\otimes d}, \qquad (5.160)\end{aligned}$$

where $\tau, \gamma > 0$ are hyper-parameters. The direct sum $\bigoplus_{r=0}^{R}$ is a concatenation of all r-body reduced density matrices, and the other direct sum $\bigoplus_{d=0}^{D}$ subsumes all degree-d polynomial expansions. The computational cost of finding a hyperplane in feature space that separates the training data into two classes is dominated by the cost of computing inner products between feature vectors. The inner product $\langle \phi^{(\mathrm{shadow})}(S_T(\rho)), \phi^{(\mathrm{shadow})}(S_T(\tilde{\rho})) \rangle$ can be analytically computed by reorganizing the direct sums, writing it as a double series, and wrapping both series into an exponential, which gives

$$k^{(\mathrm{shadow})}\left(S_T(\rho), S_T(\tilde{\rho})\right) = \exp\left(\frac{\tau}{T^2} \sum_{t,t'=1}^{T} \exp\left(\frac{\gamma}{n} \sum_{i=1}^{n} \mathrm{tr}\left(\sigma_i^{(t)} \tilde{\sigma}_i^{(t')} \right) \right) \right), \qquad (5.161)$$

where $S_T(\rho)$ and $S_T(\tilde{\rho})$ are classical shadow representations of ρ and $\tilde{\rho}$, respectively. The computation time for the inner product is $O(nT^2)$, linear in the system size n and quadratic in T, the number of copies of each quantum state which are measured to construct the classical shadow.

Rigorous guarantee

By statistical analysis, we can establish a rigorous guarantee for the classical ML model $\langle \alpha, \phi^{(\text{shadow})}(S_T(\rho)) \rangle$, where α is the trainable vector defining the classifying hyperplane. The result is the following theorem proven in Appendix 5.9.

Theorem 21 (Classifying quantum phases of matter; informal)**.** *If there is a nonlinear function of few-body reduced density matrices that classifies phases, then the classical algorithm can learn to classify these phases accurately. The required amount of training data and computation time scale polynomially in system size.*

If there is an efficient procedure based on *few-body reduced density matrices* for classifying phases, the proposed ML algorithm is guaranteed to find the procedure efficiently. This includes local order parameters for classifying symmetry breaking phases, and topological entanglement entropy in a sufficiently large local region for partially classifying topological phases (A. Y. Kitaev and Preskill, 2006; Levin and Wen, 2006). We expect that, to classify topological phases accurately, the classical ML will need access to local regions that are sufficiently large compared to the correlation length, and as we approach the phase boundary, the correlation length increases. As a result, the classifying function for topological phases may depend on r-body subsystems with a larger r, and the amount of training data and computation time required would increase accordingly. Note that the classical ML not only classifies phases accurately, but also constructs a classifying function explicitly.

Our classical ML model may also be useful for classifying and understanding symmetry-protected topological (SPT) phases. SPT phases are characterized much like topological phases, but with the additional constraint that all structures involved (states, Hamiltonians, and quantum circuits) respect a particular symmetry. It is reasonable to expect that an SPT phase can be identified by examining reduced density matrices on constant-size regions (H. Li and F. D. M. Haldane, 2008; Pollmann, Ari M. Turner, et al., 2010; Pollmann and Ari M Turner, 2012; Haegeman et al., 2012; Shapourian, Shiozaki, and Ryu, 2017; Dehghani et al., 2021), where the size of the region is large compared to the correlation length. The existence of classifying functions based on reduced matrices have been rigorously established in some cases (Alexei Yu. Kitaev, 2006; Y. Zhang and E.-A. Kim, 2017; Matthew B. Hastings and Michalakis, 2015; Kapustin and Sopenko, 2020; Bachmann, Bols, et al., 2020; Bachmann and Nachtergaele, 2014; Tasaki, 2018; Tasaki, 2020). In Appendix 5.10, we prove that the ML algorithm is guaranteed to efficiently classify a class of gapped spin-1 chains in one dimension. For more general SPT phases, the ML algorithm should be able to corroborate known classification schemes, determine new and potentially more compact classifiers, and shed light on interacting SPT phases in two or more dimensions for which complete classification schemes have not yet been firmly established.

The hypothesis of Theorem 21, stating that phases can be recognized by inspecting regions of constant size independent of the total system size, is particularly plausible for gapped phases, but might apply to some gapless phases as well. Our classical ML model would be able to efficiently classify such gapless phases. On the other hand, the contrapositive of Theorem 21 asserts that if the classical ML is not able to distinguish between two distinct gapless phases, then nonlocal data is required to characterize at least one of those phases.

5.8 No observable can classify topological phases

Recall that ground states of two Hamiltonians are in the same topological phase if there exists a constant-depth geometrically-local quantum circuit that can transform one state to another (Zeng et al., 2019). The goal of this section is to establish the following proposition.

Proposition 11. *Consider two distinct topological phases A and B (one of the phases could be the trivial phase). No observable O exists such that*

$$\mathrm{tr}(O\rho) > 0, \forall \rho \in \textit{phase } A, \qquad \mathrm{tr}(O\rho) \leq 0, \forall \rho \in \textit{phase } B. \tag{5.162}$$

Proof. We consider depth-1 quantum circuits consisting of single-qubit unitaries $U_1, \ldots, U_n$. We let $|\psi_A\rangle, |\psi_B\rangle$ be the signature quantum state for phase A and B. Suppose there is an observable such that

$$\mathrm{tr}(O\rho) > 0, \forall \rho \in \text{phase } A, \qquad \mathrm{tr}(O\rho) \leq 0, \forall \rho \in \text{phase } B. \tag{5.163}$$

Then, by definition, we have

$$\langle\psi_A| (U_1^\dagger \otimes \ldots \otimes U_n^\dagger) O (U_1 \otimes \ldots \otimes U_n) |\psi_A\rangle > 0, \forall U_1, \ldots, U_n \in U(2), \tag{5.164a}$$

$$\langle\psi_B| (U_1^\dagger \otimes \ldots \otimes U_n^\dagger) O (U_1 \otimes \ldots \otimes U_n) |\psi_B\rangle \leq 0, \forall U_1, \ldots, U_n \in U(2), \tag{5.164b}$$

However, from Lemma 23, no such observable exists. Hence no observable exists that can be used to classify two topologically ordered phases. □

The key lemma utilized in the above proof is the following.

Lemma 23. *For any two n-qubit states $|\psi_1\rangle, |\psi_2\rangle$, no observable O exists such that*

$$\langle\psi_1| (U_1^\dagger \otimes \ldots \otimes U_n^\dagger) O (U_1 \otimes \ldots \otimes U_n) |\psi_1\rangle > 0, \forall U_1, \ldots, U_n \in U(2), \quad (5.165a)$$

$$\langle\psi_2| (U_1^\dagger \otimes \ldots \otimes U_n^\dagger) O (U_1 \otimes \ldots \otimes U_n) |\psi_2\rangle \leq 0, \forall U_1, \ldots, U_n \in U(2), \quad (5.165b)$$

where $U(2)$ is the unitary group of 2×2 unitary matrices.

Proof. We will prove this result by contradiction. Assume the existence of an observable O such that Eq. (5.165a) and (5.165b) both hold. Consider $U_1, \ldots, U_n$ to be independent random matrices that follows the Haar measure on the unitary group $U(2)$. Then using the identity for the first order moment of Haar integration,

$$\mathop{\mathbb{E}}_{U \sim \mathrm{Haar}(U(d))} U X U^\dagger = \mathrm{tr}(X) \frac{\mathbb{I}}{d}, \quad (5.166)$$

we can obtain the following identity,

$$\mathop{\mathbb{E}}_{U_1, \ldots, U_n \sim \mathrm{Haar}(U(2))} \left[(U_1 \otimes \ldots \otimes U_n) |\psi_1\rangle\!\langle\psi_1| (U_1^\dagger \otimes \ldots \otimes U_n^\dagger) \right] \quad (5.167)$$

$$= \mathrm{tr}(|\psi_1\rangle\!\langle\psi_1|) \frac{\mathbb{I}}{2^n} = \frac{\mathbb{I}}{2^n}. \quad (5.168)$$

The key property is the compactness of the unitary group $U(2)$. Consider the following infimum,

$$o_1 = \inf_{U_1, \ldots, U_n \in U(2)} \langle\psi_1| (U_1^\dagger \otimes \ldots \otimes U_n^\dagger) O (U_1 \otimes \ldots \otimes U_n) |\psi_1\rangle. \quad (5.169)$$

Because the infimum is always attained by an element in the compact set,

$$\exists U_1^{\mathrm{inf}}, \ldots, U_n^{\mathrm{inf}} \in U(2) \quad (5.170)$$

such that

$$o_1 = \langle\psi_1| ((U_1^{\mathrm{inf}})^\dagger \otimes \ldots \otimes (U_n^{\mathrm{inf}})^\dagger) O (U_1^{\mathrm{inf}} \otimes \ldots \otimes U_n^{\mathrm{inf}}) |\psi_1\rangle. \quad (5.171)$$

Therefore, we have $o_1 > 0$ from Eq. (5.165a). Using the property of infimum, we have

$$\langle\psi_1| (U_1^\dagger \otimes \ldots \otimes U_n^\dagger) O (U_1 \otimes \ldots \otimes U_n) |\psi_1\rangle \geq o_1, \forall U_1, \ldots, U_n \in U(2), \quad (5.172)$$

we have the following inequality,

$$\mathop{\mathbb{E}}_{U_1, \ldots, U_n \sim \mathrm{Haar}(U(2))} \langle\psi_1| (U_1^\dagger \otimes \ldots \otimes U_n^\dagger) O (U_1 \otimes \ldots \otimes U_n) |\psi_1\rangle \geq o_1 > 0. \quad (5.173)$$

By the linearity of expectation and Eq. (5.167), we have

$$\mathop{\mathbb{E}}_{U_1,\ldots,U_n\sim\mathrm{Haar}(U(2))} \langle\psi_1|\,(U_1^\dagger\otimes\ldots\otimes U_n^\dagger)O(U_1\otimes\ldots\otimes U_n)\,|\psi_1\rangle \tag{5.174}$$
$$= \mathrm{tr}\left(O \mathop{\mathbb{E}}_{U_1,\ldots,U_n\sim\mathrm{Haar}(U(2))}\left[(U_1\otimes\ldots\otimes U_n)|\psi_1\rangle\!\langle\psi_1|(U_1^\dagger\otimes\ldots\otimes U_n^\dagger)\right]\right) = \frac{\mathrm{tr}(O)}{2^n}.$$

Together, we have

$$\frac{\mathrm{tr}(O)}{2^n} \geq o_1 > 0. \tag{5.175}$$

The argument for $|\psi_2\rangle$ is slightly simpler. Consider the following supremum,

$$o_2 = \sup_{U_1,\ldots,U_n\in U(2)} \langle\psi_2|\,(U_1^\dagger\otimes\ldots\otimes U_n^\dagger)O(U_1\otimes\ldots\otimes U_n)\,|\psi_2\rangle\,. \tag{5.176}$$

From Eq. (5.165b), we have $o_2 \leq 0$. Using the fact that

$$\langle\psi_2|\,(U_1^\dagger\otimes\ldots\otimes U_n^\dagger)O(U_1\otimes\ldots\otimes U_n)\,|\psi_2\rangle \leq o_2, \forall U_1,\ldots,U_n\in U(2), \tag{5.177}$$

we have the following inequality,

$$\mathop{\mathbb{E}}_{U_1,\ldots,U_n\sim\mathrm{Haar}(U(2))} \langle\psi_2|\,(U_1^\dagger\otimes\ldots\otimes U_n^\dagger)O(U_1\otimes\ldots\otimes U_n)\,|\psi_2\rangle \leq o_2 \leq 0. \tag{5.178}$$

By the linearity of expectation and Eq. (5.167), we have

$$\mathop{\mathbb{E}}_{U_1,\ldots,U_n\sim\mathrm{Haar}(U(2))} \langle\psi_2|\,(U_1^\dagger\otimes\ldots\otimes U_n^\dagger)O(U_1\otimes\ldots\otimes U_n)\,|\psi_2\rangle \tag{5.179}$$
$$= \mathrm{tr}\left(O \mathop{\mathbb{E}}_{U_1,\ldots,U_n\sim\mathrm{Haar}(U(2))}\left[(U_1\otimes\ldots\otimes U_n)|\psi_2\rangle\!\langle\psi_2|(U_1^\dagger\otimes\ldots\otimes U_n^\dagger)\right]\right) = \frac{\mathrm{tr}(O)}{2^n}.$$

Together, we have

$$\frac{\mathrm{tr}(O)}{2^n} \leq o_2 \leq 0. \tag{5.180}$$

From Eq. (5.175) and (5.180), we have derived the following result

$$\frac{\mathrm{tr}(O)}{2^n} \leq o_2 \leq 0 < o_1 \leq \frac{\mathrm{tr}(O)}{2^n}, \tag{5.181}$$

which is a contradiction. Therefore, no such observable O exists. □

5.9 Proof of efficiency for classifying phases of matter

This section contains a detailed proof for another one of our main contributions. Namely, a rigorous performance guarantee for learning to predict quantum phases of matter.

Training support vector machines

Let us start by reviewing the textbook framework for reasoning about supervised learning tasks: support vector machines (SVMs). The underlying idea is simple and intuitive. Suppose that we have N data points $\mathbf{x}_\ell \in \mathbb{R}^D$ with binary labels $y_\ell \in \{\pm 1\}$ that form two well separated clusters. Then, we may try to separate these training clusters with a linear hyperplane $\mathsf{H}_\alpha = \left\{\mathbf{x} \in \mathbb{R}^D : \langle \alpha, \mathbf{x} \rangle = 0\right\} \subset \mathbb{R}^D$, defined using any vector α that is perpendicular to all vectors in the hyperplane. Here, we implicitly assume that the hyperplane H_α must contain the origin $\mathbf{0} \in \mathbb{R}^D$. This simplifies exposition and will suffice for our purposes, but also constitutes an actual restriction (linear SVMs typically also allow for affine shifts). Such a hyperplane divides $\mathbb{R}^D$ up into two half-spaces. For linear classification, we want that these half-spaces perfectly capture the labels of training data: $\langle \alpha, \mathbf{x}_\ell \rangle > 0$ whenever $y_\ell = +1$ and $\langle \alpha, \mathbf{x}_\ell \rangle < 0$ whenever $y_\ell = -1$. The hope is that this simple linear classification strategy generalizes to data we haven't yet seen. When we get a new data point, we simply check which halfspace it belongs to and assign the label accordingly. In the training stage, the main question is: how do we find a suitable hyperplane? Several strategies are known in the literature. One of them is the *soft margin* problem:

$$\underset{\alpha \in \mathbb{R}^D}{\text{minimize}} \quad \sum_{\ell=1}^{N} \max\{0, 1 - y_\ell \langle \alpha, \mathbf{x}_\ell \rangle\} \tag{5.182a}$$

$$\text{subject to} \quad \langle \alpha, \alpha \rangle \leq \Lambda^2. \tag{5.182b}$$

For both label values, a positive product $y_\ell \langle \alpha, \mathbf{x}_\ell \rangle$ is theoretically sufficient. However, numerical precision considerations warrant a nonzero separation between the clusters, so the product is optimized to be at least as large as a positive number (here, 1). Otherwise, a hyperplane defined by α does not perfectly classify the data, yielding the training error $\mathrm{E}_{\mathrm{tr}}(\alpha) = \sum_{\ell=1}^{N} \max\{0, 1 - y_\ell \langle \alpha, \mathbf{x}_\ell \rangle\}$. The task is to find $\alpha_\sharp$ that achieves the smallest training error: $\mathrm{E}_{\mathrm{tr}}(\alpha_\sharp) \leq \mathrm{E}_{\mathrm{tr}}(\alpha)$ for all vectors that obey $\langle \alpha, \alpha \rangle \leq \Lambda^2$. This is a convex optimization problem that can be solved in polynomial time and we refer to Figure 5.3 for a visual illustration. The most interesting situation occurs if we manage to achieve an optimal objective value of 0. This corresponds to zero training error. In this case, we have found a hyperplane $\mathsf{H}_{\alpha_\sharp}$ that perfectly separates training data. What is more, the constraint $\langle \alpha_\sharp, \alpha_\sharp \rangle \leq \Lambda^2$ ensures that the margin of separation is strictly positive. Let $\hat{\alpha} = \alpha / \|\alpha\|$ be the unit vector that characterizes a hyperplane. Then, zero training error implies $\langle \hat{\alpha}, \mathbf{x}_\ell \rangle \geq 1/\|\alpha\| \geq 1/\Lambda$ for all $\mathbf{x}_\ell$ with $y_\ell = +1$ and $\langle \hat{\alpha}, \mathbf{x}_\ell \rangle < -1/\Lambda$ for all $\mathbf{x}_\ell$ with $y_\ell = -1$. In turn, the

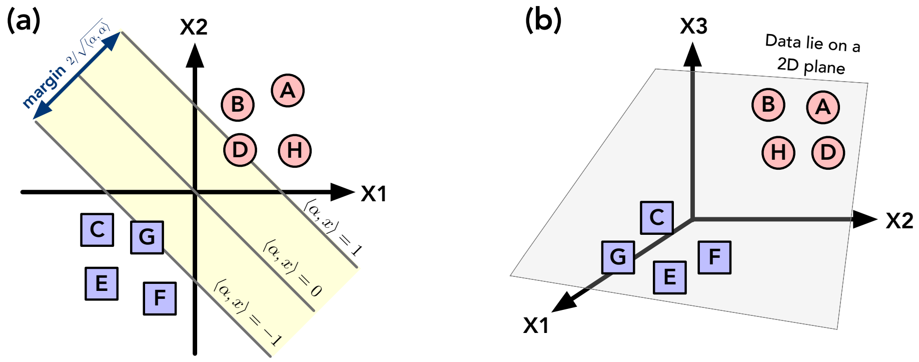

(a)
X2
X1
margin $2/\sqrt{\langle \alpha, \alpha \rangle}$
A
B
D
H
C
G
E
F
$\langle \alpha, x \rangle = 1$
$\langle \alpha, x \rangle = 0$
$\langle \alpha, x \rangle = -1$
(b)
X3
X2
X1
Data lie on a 2D plane
B
A
H
D
C
G
E
F

Figure 5.3: (a) Geometric intuition behind support vector machines (SVMs). The idea is to separate clusters of labeled data with a linear hyperplane. The separation margin (yellow) is inversely proportional to the length $\sqrt{\langle \alpha, \alpha \rangle}$ of the hyperplane normal vector. During the training stage we try to find a hyperplane that separates points with label +1 (blue) from points with label -1 (red) such that the margin is as large as possible (left). This hyperplane separates the data space into two halfspaces. In order to predict the label of a new data point, we simply check which halfspace it belongs to. (b) Geometric intuition behind the representer theorem. When trying to find a separating hyperplane, the total dimension of the data space does not matter. We can without loss restrict our attention to the smallest subspace that contains all the data points. This is because orthogonal directions don't matter during training and has two implications: (i) the cost of finding a separating hyperplane depends on the training data size N, not feature space dimension and (ii) we can express the hyperplane vector as a linear combination of training data points.

minimal margin amounts to $2/\Lambda$.

However, it should not come as a surprise that such linear classification strategies are often inadequate. Most labeled collections of data simply cannot be separated by a linear hyperplane. However, it has been observed that this drawback can be overcome by first transforming data into a (usually much larger) feature space $\mathbf{x}_\ell \mapsto \phi(\mathbf{x}_\ell)$ and trying to find a separating hyperplane there. This transformation is typically nonlinear and increases the expressiveness of hyperplane classification. Although the separating hyperplane is linear in feature space, it may be highly nonlinear in the original data space. Denote the feature space by $\mathcal{F}$ and suppose that it comes with an inner product $\langle \cdot, \cdot \rangle_{\mathcal{F}}$ and dual space $\mathcal{F}^*$. We can then formally phrase the search for a linear classifier in feature space as

$$\begin{aligned} \underset{\alpha \in \mathcal{F}^*}{\text{minimize}} \quad & \sum_{\ell=1}^{N} \max\left\{0, 1 - y_\ell \langle \alpha, \phi(\mathbf{x}_\ell) \rangle_{\mathcal{F}}\right\} && (5.183\text{a}) \\ \text{subject to} \quad & \langle \alpha, \alpha \rangle_{\mathcal{F}} \leq \Lambda^2. && (5.183\text{b}) \end{aligned}$$

This problem looks more daunting than its linear counterpart, especially because the feature space $\mathcal{F}$ may have an exceedingly large – perhaps even infinite – dimension. But we are still interested in identifying a hyperplane that separates a total of N transformed data points $\phi(\mathbf{x}_1), \ldots, \phi(\mathbf{x}_N) \in \mathcal{F}$ in a linear fashion: $\langle \alpha, \phi(\mathbf{x}_\ell)\rangle_{\mathcal{F}} > 0$ if $y_\ell = +1$ and $\langle \alpha, \phi(\mathbf{x}_\ell)\rangle_{\mathcal{F}} < 0$ else if $y_\ell = -1$. And in order to achieve this, we can without loss restrict ourselves to the N-dimensional subspace span $\{\phi(\mathbf{x}_1), \ldots, \phi(\mathbf{x}_N)\} \subset \mathcal{F}$ that is spanned by the data points themselves (all other directions are orthogonal to *all* data points and do not play a role for classification). For finite dimensional feature spaces $(\mathcal{F}, \langle \cdot, \cdot \rangle_{\mathcal{F}})$, this is an intuitive observation that follows from basic orthogonality arguments. We refer to Figure 5.3 for a visual illustration. For infinite-dimensional feature spaces it is the content of the celebrated generalized representer theorem (Schölkopf, Herbrich, and Alex J Smola, 2001). More formally, this insight allows us to decompose every (relevant) hyperplane normal vector α in the optimization problem (5.183a) as $\alpha = \sum_{\ell=1}^N \alpha_\ell \phi(\mathbf{x}_\ell)$. Linearity then ensures $\langle \alpha, \phi(\mathbf{x}_{\ell'})\rangle_{\mathcal{F}} = \sum_{\ell=1}^N \alpha_\ell \langle \phi(\mathbf{x}_\ell), \phi(\mathbf{x}_{\ell'})\rangle_{\mathcal{F}}$ for each $\ell' \in \{1, \ldots, N\}$ and also $\langle \alpha, \alpha \rangle_{\mathcal{F}} = \sum_{\ell,\ell'=1}^N \alpha_\ell \alpha_{\ell'} \langle \phi(\mathbf{x}_\ell), \phi(\mathbf{x}_{\ell'})\rangle_{\mathcal{F}}$. These expressions only depend on the elements of a $N \times N$ Gram matrix in feature space:

$$[\mathbf{K}]_{\ell\ell'} = \langle \phi(\mathbf{x}_\ell), \phi(\mathbf{x}_{\ell'})\rangle_{\mathcal{F}} =: k\left(\mathbf{x}_\ell, \mathbf{x}_{\ell'}\right) \quad \text{for } \ell, \ell' \in \{1, \ldots, N\}. \tag{5.184}$$

The expression $k(\mathbf{x}_\ell, \mathbf{x}_{\ell'})$ is called the *kernel* associated with the feature map ϕ and the matrix $\mathbf{K}$ is the *kernel matrix*. Kernels are a measure of similarity between (training) data points that is often easier to compute than performing the underlying feature map $\phi : \mathbb{R}^D \to \mathcal{F}$. But, for linear classification (in feature space), both contain exactly the same amount of information. Indeed, we may re-express the optimization problem (5.183a) as

$$\underset{\alpha \in \mathbb{R}^N}{\text{minimize}} \quad \sum_{\ell=1}^N \max\left\{0, 1 - y_\ell \alpha^T \mathbf{K} \mathbf{e}_\ell\right\} \tag{5.185a}$$

$$\text{subject to} \quad \alpha^T \mathbf{K} \alpha \leq \Lambda^2. \tag{5.185b}$$

We can also collect the classification labels in a diagonal matrix

$$\mathbf{Y} = \text{diag}\left(y_1, \ldots, y_N\right) \tag{5.186}$$

of compatible dimension and linearize the loss function by means of an entry-wise nonnegative slack variable $\beta \geq \mathbf{0}$. Let $\mathbf{1} = (1, \ldots, 1)^T$ denote the vector of ones. Then, problem (5.185a) is equivalent to solving

$$\underset{\alpha, \beta \in \mathbb{R}^N}{\text{minimize}} \quad \langle \mathbf{1}, \beta \rangle \tag{5.187a}$$

$$\text{subject to} \quad \beta \geq \mathbf{1} - \mathbf{YK}\alpha \tag{5.187b}$$

$$\beta \geq \mathbf{0}, \ \alpha^* \mathbf{K} \alpha \leq \Lambda^2. \tag{5.187c}$$

Similar to before, the optimal function value denotes the minimal *training error* $\mathrm{E}_{\mathrm{tr}}(\alpha_\sharp) = \langle \mathbf{1}, \beta_\sharp \rangle$. Apart from a single quadratic constraint ($\alpha^* \mathbf{K} \alpha \leq \Lambda^2$), this optimization problem looks like a linear program in $2N$ dimensions. It is a convex instance of a quadratically constrained quadratic program (QCQP) and can be solved in time at most polynomial in the training data size N (S. Boyd, S. P. Boyd, and Vandenberghe, 2004). In practice, one could use existing software packages, such as scikit-learn (Pedregosa et al., 2011) or LIBSVM (Chang and C.-J. Lin, 2011a). If the time to compute the kernel function $k(\mathbf{x}_\ell, \mathbf{x}_{\ell'})$ is t_{kernel}, then the time complexity for training a support vector machine is given by

$$O(t_{\mathrm{kernel}} N^2 + \mathrm{poly}(N)) \qquad \text{(training time)}. \tag{5.188}$$

Hence, for support vector machines with efficiently computable kernel functions $k(\mathbf{x}_\ell, \mathbf{x}_{\ell'})$, small training data sizes N directly ensure a short training time. The polynomial scaling in training data size depends on the type of algorithm. Dedicated solvers for the soft margin problem (Joachims, 1999; Chang and C.-J. Lin, 2011a; Hazan, Koren, and Srebro, 2011) require (at most) $O\left(N^3 + \Lambda^2 N/\epsilon^2\right)$ arithmetic operations to produce a solution $\alpha_{\sharp,\epsilon}$ that is ϵ-close to optimal: $\mathrm{E}_{\mathrm{tr}}(\alpha_{\sharp,\epsilon}) \leq \mathrm{E}_{\mathrm{tr}}(\alpha_\sharp) + \epsilon$. For the concrete training problems considered here, such an approximation is good enough and the associated runtime bound simplifies to $O\left(t_{\mathrm{kernel}} N^2 + N^3\right)$. Interior point methods offer an alternative that scale worse in training data size, but much better in the approximation error ϵ, see e.g. (S. Boyd, S. P. Boyd, and Vandenberghe, 2004).

Prediction using support vector machines

In the last section, we have explained how feature maps and kernels can considerably boost the expressiveness of initially linear classifiers. We have also explained how to use labeled training data of size N to find a separating hyperplane in feature space by solving a quadratic program (5.187a) that depends on the kernel matrix (5.184). Ideally, $\mathrm{E}_{\mathrm{tr}}(\alpha_\sharp) = 0$ (zero training error) and the optimal solution $\alpha_\sharp \in \mathbb{R}^N$ parametrizes a separating hyperplane with minimal margin $2/\Lambda$ in feature space:

$$h_\sharp(\mathbf{x}_{\ell'}) = \sum_{\ell=1}^{N} \left[\alpha_\sharp\right]_\ell \langle \phi(\mathbf{x}_\ell), \phi(\mathbf{x}_{\ell'}) \rangle_{\mathcal{F}} \tag{5.189}$$

$$= \sum_{\ell=1}^{N} \left[\alpha_\sharp\right]_\ell k\left(\mathbf{x}_\ell, \mathbf{x}_{\ell'}\right) \begin{cases} > +1/\Lambda & \text{if } y_{\ell'} = +1, \\ < -1/\Lambda & \text{else if } y_{\ell'} = -1, \end{cases} \tag{5.190}$$

for all (labeled) training data points $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$.The sign of this classifier, in turn, correctly reproduces training labels:

$$y_\sharp(\mathbf{x}_{\ell'}) := \operatorname{sign}\left(h_\sharp(\mathbf{x}_{\ell'})\right) = y_{\ell'} \quad \text{for each } \ell \in \{1, \ldots, N\}. \tag{5.191}$$

In the prediction stage, we use this function to assign a label $y_\sharp(\mathbf{x}) \in \{\pm 1\}$ to a new (and unlabeled) data point $\mathbf{x}$. The cost of evaluating $y_\sharp(\mathbf{x}_{\ell'})$ is dominated by the cost of evaluating N kernel functions. If the time to compute the kernel function is t_{kernel}, then the prediction time for a new input vector $\mathbf{x}$ is bounded by

$$\mathcal{O}(t_{\text{kernel}} N) \qquad\qquad \text{(prediction time).} \tag{5.192}$$

Similar to the training time (5.188), a small training data size N translates into a fast prediction time.

The hope is that training with an adequate kernel uncovers latent structure that generalizes beyond training data. Typically, larger training data sizes N also increase the chance for learning good classifiers (5.191). But generalization beyond training data often only makes sense if the new data point $\mathbf{x}$ is somewhat related to the training data (e.g. training a SVM on labeled cat-vs-dog images does not necessarily produce a classifier that can distinguish apples from oranges). Extra assumptions that address similarity of training and prediction data are important when one aims at establishing rigorous bounds on the probability of making a wrong prediction, i.e. $y_\sharp(\mathbf{x}) = -y(\mathbf{x})$. A common assumption is that both the training data and new data points are generated independently from the same distribution: $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N), (\mathbf{x}, y) \sim \mathcal{D}$. The data distribution $\mathcal{D}$ is a joint distribution over both the input vector $\mathbf{x}$ and the label y. Such an assumption encompasses the intuition that the label y is correlated with the input vector $\mathbf{x}$, but is not necessarily a function of $\mathbf{x}$. Flexibility of this form is useful for describing situations where the data points $\mathbf{x}$ are corrupted by noise. This is often the case in quantum mechanics due to the inherent randomness in quantum measurements. The underlying data distribution should be taken into account when reasoning about false predictions, motivating the probability

$$\Pr_{(\mathbf{x},y)\sim\mathcal{D}}\left[y_\sharp(\mathbf{x}) \neq y\right] \in [0, 1] \qquad \text{(average-case prediction error)} \tag{5.193}$$

as a good figure of merit. Noting that there are in general many approaches to bounding the prediction error, we present a user-friendly theorem that bounds the average-case prediction error in terms of the training error $\mathrm{E}_{\mathrm{tr}}(\alpha_\sharp)$ and training data size N.

Theorem 22 (Prediction error for support vector machines)**.** *Fix a data distribution* $(\mathbf{x}, y) \sim \mathcal{D}$*, a kernel function* $k(\cdot,\cdot)$*, a minimal margin* $2/\Lambda$ *and a training data size* N*. Assume* $k(\mathbf{x},\mathbf{x}) \leq R^2$ *almost surely. Then, with probability (at least)* $1-\delta$*,*

$$\mathrm{Pr}_{(\mathbf{x},y)\sim\mathcal{D}}\left[y_\sharp(\mathbf{x}) \neq y\right] \leq \frac{1}{N}\mathrm{E}_{\mathrm{tr}}(\alpha_\sharp) + 7(\Lambda R + 1)\sqrt{\frac{\log(2/\delta)}{N}}, \tag{5.194}$$

where $y_\sharp(\mathbf{x})$ *is the classifier* (5.191) *obtained from solving the training problem* (5.187a) *on independently sampled training data* $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N) \sim \mathcal{D}$*, and* E_{tr} *denotes the associated training error.*

This rigorous statement bounds the average prediction error in terms of the training error plus an error term that decays as $1/\sqrt{N}$ in training data size. The core assumption is that training and prediction data is sampled in an independent and identically distributed (*iid*) fashion. The proof is based on specializing a standard result from high dimensional probability theory to the task at hand.

Theorem 23 (Theorem 3.3 in (Mohri, Rostamizadeh, and Talwalkar, 2018))**.** *Fix a probability distribution* $\mathcal{D}$ *over elements in a set* X*, a family of functions* $\mathcal{G}$ *from* X *to the interval* $[0,\gamma_{\max}]$*, as well as* $\delta \in (0,1)$ *and* $N \in \mathbb{N}$*. Then, with probability* $1-\delta$*, the following bound is valid for* all *functions* $g \in \mathcal{G}$ *simultaneously:*

$$\underset{x\sim\mathcal{D}}{\mathbb{E}}[g(x)] \tag{5.195}$$

$$\leq \frac{1}{N}\sum_{\ell=1}^{N} g(x_\ell) + 3\gamma_{\max}\sqrt{\frac{\log(2/\delta)}{2N}} + \frac{2}{\sqrt{N}}\underset{\varepsilon_1,\ldots,\varepsilon_N}{\mathbb{E}}\left[\sup_{g\in\mathcal{G}}\frac{1}{\sqrt{N}}\sum_{\ell=1}^{N}\varepsilon_\ell g(x_\ell)\right]. \tag{5.196}$$

Here, $x_1,\ldots,x_N \overset{iid}{\sim} \mathcal{D}$ *are sampled from* X *and* $\varepsilon_1,\ldots,\varepsilon_N \overset{iid}{\sim} \{\pm 1\}$ *are Rademacher random variables (the failure probability* $\leq \delta$ *addresses these random selections).*

The right hand side of this upper bound contains three qualitatively different contributions. The first term describes an empirical average over N independent samples. It approximates the true expectation value by Monte Carlo sampling, and can underestimate the true average. As N increases, the approximation accuracy becomes better and, simultaneously, the probability of sampling a poor approximation diminishes exponentially. This is precisely the content of the second term. Larger sampling rates N suppress it and also allow for insisting on ever smaller failure probabilities δ. However, these two terms are still not enough for an upper bound because we would like to have a bound for *all functions* $g \in \mathcal{G}$. This is where the third term comes into play. It contains the empirical width, a statistical summary parameter for the extent of the function set $\mathcal{G}$, see e.g. (Vershynin, 2018a). Suppose, for instance, that $\mathcal{G} = \{g\}$ contains only a single function. Then, we can ignore the supremum (over a single element) and the contribution vanishes entirely (Rademacher random variables have zero expectation). The empirical width parameter can, however, grow with the size of the function set $g \in \mathcal{G}$.

In the context of bounding the performance of support vector machines, the domain variable x becomes $(\mathbf{x}, y)$, and the function family consists of the training error g_α from Eq. (5.183a), indexed by α. The third term in Theorem 23 can then be bounded by the largest norm of the feature vectors.

Lemma 24. *Fix a feature map* $\phi : \mathbb{R}^D \times \{\pm 1\} \to \mathcal{F}$ *and define* $g_\alpha(\mathbf{x}, y) = \max\{0, 1 - y\langle \alpha, \phi(\mathbf{x})\rangle_{\mathcal{F}}\}$ *for* $\alpha \in \mathcal{F}^*$. *Then,*

$$\mathop{\mathbb{E}}_{\varepsilon_1,\ldots,\varepsilon_N}\left[\sup_{\langle\alpha,\alpha\rangle_{\mathcal{F}}\leq\Lambda^2} \frac{1}{\sqrt{N}}\sum_{\ell=1}^N \varepsilon_\ell g_\alpha\left(\mathbf{x}_\ell, y_\ell\right)\right] \leq \Lambda \max_{1\leq\ell\leq N}\sqrt{\langle\phi(\mathbf{x}_\ell), \phi(\mathbf{x}_\ell)\rangle_{\mathcal{F}}} \tag{5.197}$$

for any collection $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N) \in \mathbb{R}^D \times \{\pm 1\}$.

Proof. Let us abbreviate the expectation over all N Rademacher random variables by $\mathbb{E}_\varepsilon$. Note that the empirical width is invariant under a constant shift of the hinge loss function: $\max\{0, 1 - z\} \mapsto \max\{0, 1 - z\} - 1$. In turn, the shifted loss function $L(z) = \max\{0, 1 - z\} - 1$ describes a contraction, i.e. $L(0) = 0$ and $|L(z_1) - L(z_2)| \leq |z_1 - z_2|$ for all $z_1, z_2 \in \mathbb{R}$. Such contractions can only decrease the empirical width. More precisely, the Rademacher comparison principle (Ledoux and Talagrand, 2013, Eq. (4.20)) asserts

$$\mathop{\mathbb{E}}_{\varepsilon}\left[\sup_{\langle\alpha,\alpha\rangle_{\mathcal{F}}\leq\Lambda^2} \frac{1}{\sqrt{N}}\sum_{\ell=1}^N \varepsilon_\ell g_\alpha(\mathbf{x}_\ell, y_\ell)\right] \tag{5.198a}$$

$$= \mathop{\mathbb{E}}_{\varepsilon}\left[\sup_{\langle\alpha,\alpha\rangle_{\mathcal{F}}\leq\Lambda^2} \frac{1}{\sqrt{N}}\sum_{\ell=1}^N \varepsilon_\ell\left(\max\{0, 1 - y_\ell\langle\alpha, \phi(\mathbf{x}_\ell)\rangle_{\mathcal{F}}\} - 1\right)\right] \tag{5.198b}$$

$$\leq \mathop{\mathbb{E}}_{\varepsilon}\left[\sup_{\langle\alpha,\alpha\rangle_{\mathcal{F}}\leq\Lambda^2} \frac{1}{\sqrt{N}}\sum_{\ell=1}^N \varepsilon_\ell y_\ell\langle\alpha, \phi(\mathbf{x}_\ell)\rangle_{\mathcal{F}}\right] \tag{5.198c}$$

$$= \mathop{\mathbb{E}}_{\varepsilon}\left[\sup_{\langle\alpha,\alpha\rangle_{\mathcal{F}}\leq\Lambda^2} \langle\alpha, h_\varepsilon\rangle_{\mathcal{F}}\right]. \tag{5.198d}$$

In the last step, we have introduced the short-hand notation

$$h_\varepsilon = \tfrac{1}{\sqrt{N}} \sum_{\ell=1}^{N} \varepsilon_\ell y_\ell \phi(\mathbf{x}_\ell) \in \mathcal{F}. \tag{5.199}$$

Applying a Cauchy-Schwarz inequality in feature space allows us to separate the supremum from the Rademacher randomness:

$$\mathop{\mathbb{E}}_{\varepsilon}\left[\sup_{\langle\alpha,\alpha\rangle_{\mathcal{F}}\leq\Lambda^2} \langle \alpha, h_\varepsilon\rangle_{\mathcal{F}}\right] \leq \sup_{\langle\alpha,\alpha\rangle_{\mathcal{F}}\leq\Lambda^2} \sqrt{\langle\alpha,\alpha\rangle_{\mathcal{F}}}\ \mathop{\mathbb{E}}_{\varepsilon}\left[\sqrt{\langle h_\varepsilon, h_\varepsilon\rangle_{\mathcal{F}}}\right] \leq \Lambda\sqrt{\mathop{\mathbb{E}}_{\varepsilon}\langle h_\varepsilon, h_\varepsilon\rangle_{\mathcal{F}}}. \tag{5.200}$$

The last inequality is Jensen's. We complete the argument using $\mathbb{E}_\varepsilon\left[\varepsilon_\ell\varepsilon_{\ell'}\right] = \delta_{\ell,\ell'}$ and $y_\ell^2 = 1$:

$$\mathop{\mathbb{E}}_{\varepsilon}\langle h_\varepsilon, h_\varepsilon\rangle_{\mathcal{F}} = \frac{1}{N}\sum_{\ell,\ell'=1}^{N} \mathop{\mathbb{E}}_{\varepsilon}\left[\varepsilon_\ell\varepsilon_{\ell'}\right] y_\ell y_{\ell'} \langle\phi(\mathbf{x}_\ell), \phi(\mathbf{x}_{\ell'})\rangle_{\mathcal{F}} \tag{5.201}$$

$$= \frac{1}{N}\sum_{\ell=1}^{N}\langle\phi(\mathbf{x}_\ell), \phi(\mathbf{x}_\ell)\rangle_{\mathcal{F}} \leq \max_{1\leq\ell\leq N}\langle\phi(\mathbf{x}_\ell), \phi(\mathbf{x}_\ell)\rangle_{\mathcal{F}}. \tag{5.202}$$

This establishes the claim. □

We are now ready to prove the general connection between average prediction (5.193) and training error.

Proof of Theorem 22. We consider functions $y_\alpha(\mathbf{x}) = \mathrm{sign}\left(\langle\alpha, \phi\left(\mathbf{x}\right)\rangle_{\mathcal{F}}\right) \in \{\pm 1\}$, such that $\alpha \in \mathcal{F}^*$ obeys $\langle\alpha,\alpha\rangle_{\mathcal{F}} \leq \Lambda^2$. This family of functions includes all classifiers that are feasible points in the training stage (5.187a) of our support vector machine. For α fixed, but otherwise arbitrary, we want to compare the corresponding classifier $y_\alpha\left(\mathbf{x}\right)$ to the true data label $y \in \{\pm 1\}$. Elementary reformulations then allow us to re-express the failure probability as

$$\Pr\nolimits_{(\mathbf{x},y)\sim\mathcal{D}}\left[y_\alpha(\mathbf{x}) \neq y\right] \tag{5.203}$$

$$= \Pr\nolimits_{(\mathbf{x},y)\sim\mathcal{D}}\left[\mathrm{sign}\left(\langle\alpha,\phi(\mathbf{x})\rangle_{\mathcal{F}}\right) \neq y\right] = \Pr\nolimits_{(\mathbf{x},y)\sim\mathcal{D}}\left[y\langle\alpha,\phi(\mathbf{x})\rangle_{\mathcal{F}} < 0\right], \tag{5.204}$$

because the sign is negative if and only if the number itself is. Next, we rewrite this probability as the expectation value of the associated indicator function and use $\mathbf{1}\left\{z \leq 0\right\} \leq \max\left\{0, 1 - z\right\}$ for all $z \in \mathbb{R}$ to obtain

$$\Pr\nolimits_{(\mathbf{x},y)\sim\mathcal{D}}\left[y\langle\alpha,\phi(\mathbf{x})\rangle_{\mathcal{F}} < 0\right] = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\mathbf{1}\left\{y\langle\alpha,\phi(\mathbf{x}_\ell)\rangle_{\mathcal{F}} < 0\right\}\right] \tag{5.205}$$

$$\leq \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \left[\max\{0, 1 - y\langle\alpha, \phi(\mathbf{x})\rangle_{\mathcal{F}}\}\right]. \tag{5.206}$$

This upper bound is the expected value of a certain hinge loss function

$$g_\alpha(\mathbf{x}, y) = \max\{0, 1 - y\langle\alpha, \phi(\mathbf{x})\rangle_{\mathcal{F}}\} \quad \text{with} \quad \langle\alpha, \alpha\rangle_{\mathcal{F}} \leq \Lambda^2. \tag{5.207}$$

The function is a specific element of an entire family, namely

$$\mathcal{G} = \left\{g_\alpha(\cdot, \cdot) : \langle\alpha, \alpha\rangle_{\mathcal{F}} \leq \Lambda^2\right\} : \mathbb{R}^D \times \{\pm 1\} \to [0, \infty). \tag{5.208}$$

The associated function values are always nonnegative and bounded. Indeed, the Cauchy-Schwarz inequality in feature space asserts

$$g_\alpha(\mathbf{x}, y) \leq \tag{5.209}$$

$$|y\langle\alpha, \phi(\mathbf{x})\rangle_{\mathcal{F}}| + 1 \leq \sqrt{\langle\alpha, \alpha\rangle_{\mathcal{F}}\langle\phi(\mathbf{x}), \phi(\mathbf{x})\rangle_{\mathcal{F}}} + 1 \tag{5.210}$$

$$\leq \Lambda\sqrt{k(\mathbf{x}, \mathbf{x})} + 1 \leq \Lambda R + 1 =: \gamma_{\max}. \tag{5.211}$$

We are now in a position to use Theorem 23. With probability (at least) $1 - \delta$,

$$\mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} [g_\alpha(\mathbf{x}, y)] \leq \tag{5.212}$$

$$\frac{1}{N}\sum_{\ell=1}^N g_\alpha(\mathbf{x}_\ell, y_\ell) + 3\gamma_{\max}\sqrt{\frac{\log(2/\delta)}{2N}} + \frac{2}{\sqrt{N}}\mathop{\mathbb{E}}_{\varepsilon}\left[\sup_{\langle\alpha,\alpha\rangle_{\mathcal{F}}\leq\Lambda^2} \frac{1}{\sqrt{N}}\sum_{\ell=1}^N \varepsilon_\ell g_\alpha(\mathbf{x}_\ell, y_\ell)\right], \tag{5.213}$$

is true for *all* dual vectors $\alpha \in \mathcal{F}^*$ that obey $\langle\alpha, \alpha\rangle_{\mathcal{F}} \leq \Lambda^2$. Here,

$$(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N) \sim \mathcal{D} \tag{5.214}$$

is a randomly sampled (but fixed) collection of labeled data points. We now use $\sqrt{k(\mathbf{x}_\ell, \mathbf{x}_\ell)} \leq R$ almost surely to apply Lemma 24 and control the empirical width term:

$$\mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} [g_\alpha(\mathbf{x}, y)] \leq \frac{1}{N}\sum_{\ell=1}^N g_\alpha(\mathbf{x}_\ell, y_\ell) + 3(\Lambda R + 1)\sqrt{\frac{\log(2/\delta)}{2N}} + \frac{2\Lambda R}{\sqrt{N}} \tag{5.215a}$$

$$\leq \frac{1}{N}\sum_{\ell=1}^N g_\alpha(\mathbf{x}_\ell, y_\ell) + 7(\Lambda R + 1)\sqrt{\frac{\log(2/\delta)}{N}}. \tag{5.215b}$$

With probability (at least) $1 - \delta$, this bound is valid for all hyperplane vectors $\alpha \in \mathcal{F}$. The tightest bound is achieved for minimizing the right hand side. This is precisely what training a support vector machine does, as the first term is precisely the training error that is minimized in the training stage (5.187a). The optimal solution $\alpha^\sharp$ to this problem simultaneously produces the actual classifier $y_\sharp(\mathbf{x})$ on the left hand side and the (minimal) training error on the right hand side. □

Kernel functions for classical shadows

We have reviewed the classical shadow formalism in Appendix 5.1. For randomized single-qubit Pauli measurements, a classical shadow approximates a n-qubit state ρ by means of T elementary tensor products. Each shadow raw data corresponds to a two-dimensional array

$$S_T(\rho) = S_T(\rho) = \left\{ |s_i^{(t)}\rangle : \; i \in \{1, \ldots, n\}, t \in \{1, \ldots, T\} \right\} \tag{5.216}$$

$$\in \{|0\rangle, |1\rangle, |+\rangle, |-\rangle, |\mathrm{i}+\rangle, |\mathrm{i}-\rangle\}^{n \times T} \tag{5.217}$$

and is combined into an approximator of the state as

$$\sigma_T(\rho) == \frac{1}{T} \sum_{t=1}^{T} \sigma_1^{(t)} \otimes \cdots \otimes \sigma_n^{(t)}, \tag{5.218}$$

where we have introduced the short-hand notation $\sigma_i^{(t)} = 3|s_i^{(t)}\rangle\langle s_i^{(t)}| - \mathbb{I}$. For these quantum state representations, we fix parameters $\tau, \gamma > 0$ and introduce a suggestive, yet finite-dimensional feature map. For large, but finite, integers $D, R > 0$ we define

$$\phi^{(\text{finite})}\left(S_T(\rho)\right) = \tag{5.219}$$

$$\bigoplus_{d=0}^{D} \sqrt{\frac{\tau^d}{d!}} \Big(\bigoplus_{r=0}^{R} \sqrt{\frac{1}{r!} \left(\frac{\gamma}{n}\right)^r} \bigoplus_{i_1=1}^{r} \cdots \bigoplus_{i_r=1}^{r} \frac{1}{T} \sum_{t=1}^{T} \mathrm{vec}\left(\sigma_{i_1}^{(t)}\right) \otimes \cdots \otimes \mathrm{vec}\left(\sigma_{i_r}^{(t)}\right) \Big)^{\otimes d}, \tag{5.220}$$

Here, $\mathrm{vec}(\cdot)$ denotes an appropriate vectorization operation that maps the real-valued vector space $\mathbb{H}_2$ of Hermitian 2×2 matrices to $\mathbb{R}^4$ such that the Hilbert-Schmidt inner product is preserved: $\langle \mathrm{vec}(A), \mathrm{vec}(B) \rangle = \mathrm{tr}(AB)$.

This feature map embeds classical shadows in a very large-dimensional, real-valued feature space $\mathcal{F}^{(\text{finite})}$. This feature space arises from taking direct sums and tensor products of $\mathrm{vec}(\mathbb{H}_2) \simeq \mathbb{R}^4$. We can extend the standard inner product $\langle \cdot, \cdot \rangle$ on $\mathbb{R}^4$ to this feature space by setting $\langle x_1 \oplus x_2, y_1 \oplus y_2 \rangle = \langle x_1, y_1 \rangle + \langle x_2, y_2 \rangle$ (direct sums), as well as $\langle x_1 \otimes x_2, y_1 \otimes y_2 \rangle = \langle x_1, y_1 \rangle \langle x_2, y_2 \rangle$ (tensor products) and extend these definitions linearly. Doing so equips the feature space $\mathcal{F}^{(\text{finite})}$ with a well-defined inner product $\langle \cdot, \cdot \rangle_{\mathcal{F}^{(\text{finite})}}$. The inner product and feature map induce a kernel function on pairs of classical shadows of equal size T:

$$k^{(\text{finite})}\left(S_T(\rho_1), \tilde{S}_T(\rho_2)\right) = \left\langle \phi^{(\text{finite})}\left(S_T(\rho_1)\right), \phi^{(\text{finite})}\left(\tilde{S}_T(\rho_2)\right) \right\rangle_{\mathcal{F}^{(\text{finite})}} \tag{5.221a}$$

$$= \textstyle\sum_{d=0}^{D} \frac{\tau^d}{d!} \left(\sum_{r=0}^{R} \frac{1}{r!} \left(\frac{\gamma}{n}\right)^r \sum_{i_1=1}^{n} \cdots \sum_{i_r=1}^{n} \frac{1}{T^2} \sum_{t,t'=1}^{T} \left\langle \mathrm{vec}\left(\sigma_{i_1}^{(t)}\right), \mathrm{vec}\left(\tilde{\sigma}_{i_1}^{(t')}\right) \right\rangle \cdots \left\langle \mathrm{vec}\left(\sigma_{i_r}^{(t)}\right), \mathrm{vec}\left(\tilde{\sigma}_{i_r}^{(t')}\right) \right\rangle \right)^d \tag{5.221b}$$

$$=\textstyle\sum_{d=0}^{D} \frac{\tau^d}{d!} \left(\sum_{r=0}^{R} \frac{1}{r!} \left(\frac{\gamma}{n}\right)^r \sum_{i_1=1}^{n} \cdots \sum_{i_r=1}^{n} \mathrm{tr}\left(\left(\frac{1}{T} \sum_{t=1}^{T} \sigma_{i_1}^{(t)} \otimes \cdots \otimes \sigma_{i_r}^{(t)}\right)\right)\left(\frac{1}{T} \sum_{t'=1}^{T} \tilde{\sigma}_{i_1}^{(t')} \otimes \cdots \otimes \tilde{\sigma}_{i_r}^{(t')}\right)\right)^d \tag{5.221c}$$

$$=\textstyle\sum_{d=0}^{D} \frac{1}{d!} \left(\frac{\tau}{T^2} \sum_{t,t'=1}^{T} \sum_{r=0}^{R} \frac{1}{r!} \left(\frac{\gamma}{n}\right)^r \sum_{i=1}^{n} \cdots \sum_{i_r=1}^{r} \mathrm{tr}\left(\sigma_{i_1}^{(t)} \tilde{\sigma}_{i_1}^{(t')}\right) \cdots \mathrm{tr}\left(\sigma_{i_r}^{(t)} \tilde{\sigma}_{i_r}^{(t')}\right)\right)^d \tag{5.221d}$$

$$=\textstyle\sum_{d=0}^{D} \frac{1}{d!} \left(\frac{\tau}{T^2} \sum_{t,t'=1}^{T} \sum_{r=0}^{R} \frac{1}{r!} \left(\frac{\gamma}{n} \sum_{i=1}^{n} \mathrm{tr}\left(\sigma_{i}^{(t)} \tilde{\sigma}_{i}^{(t')}\right)\right)^r\right)^d. \tag{5.221e}$$

This kernel function still looks somewhat complicated, but it simplifies considerably if we first take $R \longrightarrow \infty$ and then $D \longrightarrow \infty$:

$$k^{(\mathrm{shadow})}\left(S_T(\rho_1), \tilde{S}_T(\rho_2)\right) \tag{5.222a}$$

$$:= \lim_{D\to\infty} \lim_{R\to\infty} k^{(\mathrm{finite})}\left(S_T(\rho_1), \tilde{S}_T(\rho_2)\right) \tag{5.222b}$$

$$= \lim_{D\to\infty} \sum_{d=0}^{D} \frac{1}{d!} \left(\frac{\tau}{T^2} \sum_{t,t'=1}^{T} \lim_{R\to\infty} \sum_{r=0}^{R} \frac{1}{r!} \left(\frac{\gamma}{n} \sum_{i=1}^{n} \mathrm{tr}\left(\sigma_i^{(t)} \tilde{\sigma}_i^{(t')}\right)\right)^r\right)^d \tag{5.222c}$$

$$= \exp\left(\frac{\tau}{T^2} \sum_{t,t'=1}^{T} \exp\left(\frac{\gamma}{n} \sum_{i=1}^{n} \mathrm{tr}\left(\sigma_i^{(t)} \tilde{\sigma}_i^{(t')}\right)\right)\right) \tag{5.222d}$$

We call this kernel function a *shadow kernel*. In contrast to its finite approximations, this kernel function can be computed very efficiently. Trace inner products between single-qubit shadow constituents assume one out of 3 values only:

$$\mathrm{tr}\left(\sigma_i^{(t)} \tilde{\sigma}_i^{(t)}\right) = \qquad \mathrm{tr}\left((3|s_i^{(t)}\rangle\!\langle s_i^{(t)}| - \mathbb{I})(3|\tilde{s}_i^{(t)}\rangle\!\langle \tilde{s}_i^{(t)}| - \mathbb{I})\right) \tag{5.223}$$

$$= 9\left|\langle s_i^{(t)} | \tilde{s}_i^{(t)}\rangle\right|^2 - 4 \in \{-4, 1/2, 5\}. \tag{5.224}$$

And we need to compute exactly nT^2 of them to unambiguously characterize the shadow kernel (5.222b). The total cost for evaluating shadow kernels also amounts to

$$O\left(nT^2\right) \qquad \text{(shadow kernel evaluation cost)} \tag{5.225}$$

arithmetic operations. As long as T is not too large, this is extremely efficient, given that we combine classical approximations of n-qubit quantum states ρ_1, ρ_2 which way well have $(4^n - 1)$ degrees of freedom. Eq. (5.224) also ensures that shadow kernels remain bounded functions:

$$0 \leq k^{(\mathrm{shadow})}\left(S_T(\rho_1), \tilde{S}_T(\rho_2)\right) \leq \exp\left(\tau \exp\left(5\gamma\right)\right), \tag{5.226}$$

because exponential functions are nonnegative and monotonic.

While easy to evaluate and conceptually appealing, the shadow kernel does have its downsides. By construction, the associated feature space is not finite-dimensional anymore. This can complicate a thorough analysis of support vector machines substantially. In particular, it is a priori not clear if powerful results, like Theorem 22, cover the shadow kernel as well. Fortunately, we can bypass such mathematical subtleties by approximating $k^{(\text{shadow})}(\cdot,\cdot)$ with $k^{(\text{finite})}(\cdot,\cdot)$, where D and R are large, but finite, numbers. This incurs an additional approximation error, but allows us to formulate theoretical prediction and training guarantees exclusively for finite-dimensional feature spaces. What is more, elementary approximation results from calculus ensure that we can make this additional approximation error arbitrarily small by making the cutoffs sufficiently large. Taylor's approximation theorem, for instance, shows that $D = \mathrm{e}^2\tau\exp(5\gamma) + \log(1/\eta) - 1$, as well as $R = 5\mathrm{e}^2\gamma + \tau\exp(5\gamma) + \log(\tau/\eta) - 1$ ensure

$$\left| k^{(\text{shadow})}\left(S_T(\rho_1), \tilde{S}_T(\rho_2)\right) - k^{(\text{finite})}\left(S_T(\rho_1), \tilde{S}_T(\rho_2)\right) \right| \leq 2\eta \tag{5.227}$$

for all pairs of classical shadows with compatible size T. Properly tuning γ and τ would yield better prediction performance in practice. Nevertheless, for simplicity, we will assume $\gamma = \tau = 1$ in the following theoretical analysis.

Finite-dimensional feature space approximations also allow us to highlight the expressiveness behind the shadow kernel (5.222b). It describes (the limit of) a feature map that extracts *all* tensor powers of *all* subsystem operators $X_A = \mathrm{tr}_{\neg A}(X) \in \mathbb{H}_2^{\otimes |A|}$, where $A \subset [n] = \{1, \ldots, n\}$. In particular, any function that can be written as a finite power series, of degree at most d_p, in reduced subsystem operators, of size at most r, becomes a *linear* function in feature space, represented by the dual vector α_f:

$$f\left(S_T(\rho)\right) \tag{5.228a}$$

$$= \sum_{d=0}^{d_p} \frac{1}{d!} \sum_{A_1 \ldots A_d \subset \{1,\ldots,n\}, |A_i| \leq r} \mathrm{tr}\left(O_{A_1,\ldots,A_d} \mathrm{tr}_{\neg A_1}\left(\sigma_T(\rho)\right) \otimes \cdots \otimes \mathrm{tr}_{\neg A_d}\left(\sigma_T(\rho)\right)\right) \tag{5.228b}$$

$$= \langle \alpha_f, \phi^{(\text{finite})}(S_T(\rho)) \rangle_{\mathcal{F}^{(\text{finite})}}, \tag{5.228c}$$

provided that $d_p \leq D, r \leq R$. The (extended) Euclidean norm of α_f is also bounded. Use Eq. (5.220) (with tuning parameters $\gamma, \tau = 1$) to compute

$$\langle \alpha_f, \alpha_f \rangle_{\mathcal{F}^{(\text{finite})}} \tag{5.229a}$$

$$\leq \sum_{d=0}^{d_p} \frac{(r!n^r)^d}{d!} \sum_{A_1,\ldots,A_d \subset \{1,\ldots,n\}, |A_i| \leq r} \operatorname{tr}\left(O_{A_1,\ldots,A_d}^2\right) \tag{5.229b}$$

$$\leq \sum_{d=0}^{d_p} \frac{(r!n^r)^d}{d!} \sum_{A_1,\ldots,A_d \subset \{1,\ldots,n\}, |A_i| \leq r} 2^{rd} \|O_{A_1,\ldots,A_d}\|_\infty^2 \tag{5.229c}$$

$$\leq (2nr)^{rd_p} \max_{\substack{d \leq d_p, A_1,\ldots,A_d \\ \subset \{1,\ldots,n\}, |A_i| \leq r}} \left\|O_{A_1,\ldots,A_d}\right\|_\infty \sum_{d=0}^{d_p} \frac{1}{d!} \sum_{\substack{A_1,\ldots,A_d \subset \{1,\ldots,n\}, \\ |A_i| \leq r}} \|O_{A_1,\ldots,A_d}\|_\infty \tag{5.229d}$$

$$\leq (2nr)^{rd_p} d_p^{d_p} \left(\sum_{d=0}^{d_p} \frac{1}{d!} \sum_{A_1,\ldots,A_d \subset \{1,\ldots,n\}, |A_i| \leq r} \|O_{A_1,\ldots,A_d}\|_\infty \right)^2 \tag{5.229e}$$

Here, we have used the fundamental Schatten-p norm relation

$$\|X\|_2 \leq \sqrt{\dim(X)} \|X\|_\infty, \tag{5.230}$$

as well as the assumption that each $O_{A_1,\ldots,A_d}$ is supported on a total tensor product space with dimension 2^{rd} (a tensor product of d subsystems comprised of at most r qubits each). The second to last inequality follow from using $\sum_i x_i^2 \leq \max_i |x_i| \sum_i |x_i|$, and Stirling's formula. The final simplifications uses Stirling's formula again as well as the fact that $\sum_i |x_i| \geq \max_i |x_i|$.

Physical assumptions about classifying quantum phases of matter

We want to learn how to classify two phases of n-qubit states: either ρ belongs to phase A ($y(\rho) = +1$) or ρ belongs to phase B ($y(\rho) = -1$). We assume that we have access to labeled classical shadows: $\left\{ \left(S_T(\rho_\ell), y(\rho_\ell) \right) : \ell \in \{1, \ldots, N\} \right\}$, where each $S_T(\rho_\ell)$ is classical shadow data obtained from performing T randomized single-qubit measurements on independent copies of ρ_ℓ. We can use this raw data to form classical representations $\sigma_T(\rho_\ell)$ of the underlying quantum state ρ_ℓ, see Eq. (5.3). The number T determines the resolution of these approximations. Note that $\sigma_T(\rho_\ell) \approx \rho_\ell$ can only become exact for $T \geq \exp(\Omega(n))$ (Guţă et al., 2020a; Haah et al., 2017). This would be far too costly for experimental implementations and efficient data processing. For instance, recall from Eq. (5.225) that a single shadow kernel evaluation scales quadratically in T. In this section, we show that we can choose much coarser resolutions if the underlying phase can be classified by a nice analytic function on reduced density matrices.

Assumption 1 (well-conditioned phase separation)**.** *Consider two phases among n-qubit states. For $\epsilon > 0$, we assume that there exists a function f on reduced r-body* *density matrices $\rho_A = \mathrm{tr}_{\neg A}(\rho)$ that can distinguish the two phases in question. In particular,*

$$f(\rho) = f\left(\{\rho_A : \ A \subset \{1,\ldots,n\}, |A| \leq r\}\right) \quad \textit{satisfies} \tag{5.231a}$$

$$f(\rho) \begin{cases} > +1 & \textit{for all } \rho \textit{ that belong to phase } A \ (y(\rho) = +1), \\ < -1 & \textit{for all } \rho \textit{ that belong to phase } B \ (y(\rho) = -1). \end{cases} \tag{5.231b}$$

Moreover, we assume that $f(\rho)$ can be approximated by a truncated power series

$$f^{(d_p)}(\rho) = \sum_{d=0}^{d_p} \frac{1}{d!} \sum_{A_1,\ldots,A_d \subset \{1,\ldots,n\}, |A_i| \leq r} \mathrm{tr}\left(O_{A_1,\ldots,A_d} \rho_{A_1} \otimes \cdots \otimes \rho_{A_d}\right), \tag{5.232}$$

up to constant accuracy: $\left|f(\rho) - f^{(d_p)}(\rho)\right| \leq 0.25$ for all n-qubit quantum states ρ. We refer to d_p as the truncation degree and define the normalization constant

$$C = \sum_{d=0}^{d_p} \frac{1}{d!} \Big(\sum_{A_1,\ldots,A_d \subset \{1,\ldots,n\}, |A_i| \leq r} \|O_{A_1,\ldots,A_d}\|_\infty \Big). \tag{5.233}$$

We don't need to know the normalization constant exactly. An upper bound is fully adequate for the theoretical analysis presented in this section.

Morally, the second part of Assumption 1 requires that the phase classication function can be well-approximated by a degree-d_p-polynomial in reduced density matrices. The actual formulation is general enough to encompass most physically relevant functions. Let us illustrate this by means of three popular examples.

Subsystem purity: Fix a subsystem $A \subset \{1,\ldots,n\}$ comprised of $|A| = r$ qubits and let $\rho_A = \mathrm{tr}_{\neg A}(\rho)$ be the associated r-body density matrix. The subsystem purity $f(\rho) = \mathrm{tr}(\rho_A^2)$ is a quadratic polynomial in this reduced density matrix. We can rewrite this as $f^{(2)}(\rho) = \mathrm{tr}(S_A \rho_A \otimes \rho_A)$, where S_A denotes the swap operator between two copies of the subsystem A. This reformulation is also an *exact* approximation of $f(\rho)$ with degree $d_p = 2$ and normalization constant $C = \frac{1}{2!}\|S_A\|_\infty = \frac{1}{2}$. These arguments readily extend to averages of multiple subsystem purities.

Subsystem Rényi entropy: Let us consider the subsystem Rényi entropy of order two $H_2(\rho_A) = -\log\left(\mathrm{tr}(\rho_A^2)\right)$ (the argument will generalize straightforwardly to higher order entropies). This function is closely related to the subsystem purity but also features a logarithm. And, although the logarithm is *not* a polynomial,

$-\log(1-x)$ can be accurately approximated by the truncated Mercator series. A crude but sufficient bound ensures

$$l^{(d_p)}(x) = \sum_{d=1}^{d_p} \frac{1}{d} x^d \tag{5.234}$$

$$\text{obeys} \quad \left| l^{(d_p)}(x) - \log(1-x) \right| \leq x^{d_p} \log\left(1/(1-x)\right) \quad \text{for } x \in (-1,1). \tag{5.235}$$

We can now approximate $H_2(\rho_A) = -\log\left(1 - (1 - \mathrm{tr}(\rho_A^2))\right)$ by $l^{(d_p)}(1 - \mathrm{tr}(\rho_A^2))$. Subsystem purities necessarily obey $\mathrm{tr}(\rho_A^2) \geq 2^{-|A|} = 2^{-r}$. This allows us to conclude

$$\left| l^{(d_p)}\left(1 - \mathrm{tr}(\rho_A^2)\right) - H_2(\rho_A) \right| \leq (1 - 2^{-r})^{d_p} r \log(2) \leq \log(2) r \exp\left(-d_p/2^r\right) \tag{5.236}$$

which drops beneath 0.25 if we set $d_p = \log(4\log(2)r)2^r = \mathcal{O}\left(\log(r)2^r\right)$. This degree scales exponentially in the subsystem size r, but is independent of total dimension. We can also use $1 = \mathrm{tr}(\rho_A)^2 = \mathrm{tr}\left(\mathbb{I}_A^{\otimes 2} \rho_A^{\otimes 2}\right)$ and $\mathrm{tr}(X)\mathrm{tr}(Y) = \mathrm{tr}(X \otimes Y)$ to bring this polynomial approximation onto the form advertised in Eq. (5.232). Indeed,

$$l^{(d_p)}\left(1 - \mathrm{tr}(\rho_A^2)\right) \tag{5.237a}$$

$$= l^{(d_p)}\left(\mathrm{tr}\left((\mathbb{I}_A^{\otimes 2} - S_A)\rho_A^{\otimes 2}\right)\right) \tag{5.237b}$$

$$= \sum_{d=1}^{d_p} \frac{1}{d!} \mathrm{tr}\left((d-1)!\left(\mathbb{I}_A^{\otimes 2} - S_A\right)^{\otimes d} \rho_A^{\otimes 2d}\right) \quad \text{and} \tag{5.237c}$$

$$C = \sum_{d=1}^{d_p} \frac{1}{d!} \left\| (d-1)!(\mathbb{I}_A^{\otimes 2} - S_A)^{\otimes d} \right\|_\infty = \sum_{d=1}^{d_p} \frac{1}{d} \left\| \mathbb{I}_A^{\otimes 2} - S_A \right\|_\infty^d = \sum_{d=1}^{d_p} \frac{1}{d} \approx \log(d_p). \tag{5.237d}$$

This analysis readily extends to higher order Rényi entropies, as well as averages over multiple subsystems.

Entanglement entropy: This is where things start to get somewhat interesting, because the entanglement (von Neumann) entropy $H(\rho_A) = -\mathrm{tr}\left(\rho_A \log(\rho_A)\right) \in [0, r\log(2)]$ of a r-body subsystem is notoriously difficult to accurately approximate with a polynomial (Fawzi, Saunderson, and Parrilo, 2019). Fortunately, Assumption 1 does not require an accurate approximation – a constant error of size $1/4$ is fine. To achieve this goal, we make the following polynomial ansatz in the reduced density matrix ρ_A:

$$H^{(d_p)}(\rho_A) = -\operatorname{tr}\left((\rho_A - \mathbb{I}_A) + \sum_{k=2}^{d_p} \frac{(\mathbb{I}_A - \rho_A)^k}{k(k-1)} \right) \tag{5.238}$$

Let λ_i denote the eigenvalues of a subsystem density matrix ρ_A and note that there are 2^r eigenvalues in ρ_A. We can rewrite the entanglement entropy and the polynomial ansatz as

$$H(\rho_A) = -\sum_{i=1}^{2^r} \lambda_i \log(\lambda_i) \quad \text{and} \tag{5.239a}$$

$$H^{(d_p)}(\rho_A) = -\sum_{i=1}^{2^r} \left((\lambda_i - 1) + \sum_{k=2}^{d_p} \frac{(1-\lambda_i)^k}{k(k-1)} \right), \tag{5.239b}$$

respectively. Using Taylor's theorem in the interval $[0, 1]$, we have

$$x \log(x) = (x-1) + \left(\sum_{k=2}^{\infty} \frac{(1-x)^k}{k(k-1)} \right). \tag{5.240}$$

Note that at $x = 0$, $x \log x = 0$ and the infinite sum comprising the second term on the right hand side also converges to 1. This ensures that the above equality is valid for the closed interval $[0, 1]$. We shall also use the following identity

$$\sum_{k=2}^{n} \frac{1}{k(k-1)} = 1 - \frac{1}{n}, \tag{5.241}$$

which remains valid even in the limit $n \longrightarrow \infty$. We combine Eq. (5.240) and (5.241) to obtain an approximation error for our polynomial ansatz function. For all $x \in [0, 1]$, we have

$$\left| x \log(x) - \left((x-1) + \left(\sum_{k=2}^{d_p} \frac{(1-x)^k}{k(k-1)} \right) \right) \right| \tag{5.242}$$

$$\leq \sum_{k=d_p+1}^{\infty} \frac{(1-x)^k}{k(k-1)} \leq \sum_{k=d_p+1}^{\infty} \frac{1}{k(k-1)} = \frac{1}{d_p}. \tag{5.243}$$

This allows us to bound the approximation error for each individual eigenvalue $\lambda_i \in [0, 1]$ of ρ_A. There are in total 2^r eigenvalues and a triangle inequality asserts

$$|H(\rho_A) - H^{(d_p)}(\rho_A)| \leq \sum_{i=1}^{2^r} \left| \lambda_i \log(\lambda_i) - \left((\lambda_i - 1) + \left(\sum_{k=2}^{d_p} \frac{(1-\lambda_i)^k}{k(k-1)} \right) \right) \right| \leq \frac{2^d}{d_p}. \tag{5.244}$$

By choosing $d_p = 2^{r+2}$, we can approximate the entanglement entropy in r-body subsystem by a polynomial function. As long as the subsystem size r is a constant independent of total system size n, the polynomial approximation degree d_p is also a constant. And it is not hard to check that the same is true for the normalization constant C. This analysis readily extends to averages of multiple entanglement entropies.

Training with shadow kernels

We are now ready to dive into the main results of this section: converting Assumption 1 into a statement about classical shadows and their expressiveness when it comes to training a support vector machine. Our measure of similarity is the *shadow kernel* (5.222b) evaluated on classical shadows. The kernel matrix is

$$[\mathbf{K}]_{\ell\ell'} = k^{(\text{shadow})} \left(S_T(\rho_\ell), S_T(\rho_{\ell'})\right) \quad \text{for } \ell, \ell' \in \{1, \ldots, N\}, \tag{5.245}$$

and implicitly specifies the feature map, as well as the nonlinear geometry with respect to which we want to find classifiers for phases. We begin by approximating the true classifier, given as a nonlinear function $f(\rho)$ in Assumption 1, by a finite power series $f^{(d_p)}(\rho)$ with degree-d_p. We will then use $f^{(d_p)}(\rho)$ as an approximate phase classifier. Recalling Eq. (5.228c), a finite power series $f^{(d_p)}(S_T(\rho))$ is linear in feature space, with its corresponding dual vector α_f defining a candidate hyperplane for separating the two phases. To complete the connection to the support vector machines from Section 5.9, we must ensure that $f^{(d_p)}(S_T(\rho))$ does not differ substantially from the approximate phase classifier $f^{(d_p)}(\rho)$ from Assumption 1. This is the content of the following auxiliary statement.

Lemma 25. *Suppose that Assumption 1 is valid for a function on reduced r-body density matrices with the two constants $C \geq 1$ and $d_p \in \mathbb{N}$. For any $0 < \epsilon < 1$, classical shadows of size*

$$T = (32/3) d_p^2 C^2 12^r \left(r\left(\log(n) + \log(12)\right) + \log(1/\delta)\right)/\epsilon^2 \tag{5.246}$$

suffice to ϵ-approximate $f^{(d_p)}(\rho)$ with high probability. In particular, for any density matrix $\rho \in \mathbb{H}_2^{\otimes n}$,

$$\left| f^{(d_p)}(S_T(\rho)) - f^{(d_p)}(\rho) \right| \leq \epsilon \tag{5.247}$$

with probability at least $1 - \delta$ (over the randomized measurement settings and outcomes producing $S_T(\rho)$).

A proof can be found at the end of this subsection. With high probability, this statement ensures that existence of a well-conditioned phase separation implies the existence of a separating hyperplane in shadow feature space. This, in turn, is enough to ensure that the SVM training stage can be executed perfectly: solving the training problem (5.187a) efficiently yields a separating hyperplane parametrization $\alpha_\sharp$ that (1) lies in the subspace $\mathbb{R}^N$ of $\mathcal{F}^{(\text{shadow})}$ spanned by the N training vectors, and (2) performs at least as well as α_f. Since we are guaranteed that α_f separates training data perfectly and achieves zero training error, $\alpha_\sharp$ must be at least as good: $\mathrm{E}_{\mathrm{tr}}(\alpha_\sharp) \leq \mathrm{E}_{\mathrm{tr}}(\alpha_f) = 0$ with high probability. The main result of this section formalizes this observation.

Proposition 12. *Suppose that Assumption 1 is valid for some function on reduced r-body density matrices with normalization constant C and truncation degree d_p. Then, for $\delta \in (0,1)$, a (joint) classical shadow size*

$$T = (512/3) d_p^2 C^2 12^r \left(r \left(\log(n) + \log(12) \right) + \log(N/\delta) \right) \tag{5.248}$$

ensures that we can achieve zero training error when solving (5.187a) *with squared margin constant $\Lambda^2 = 4\,(2rn)^{rd_p}\, d_p^{d_p} C^2$.*

The extra constraint $\Lambda^2 \geq \langle \alpha_f, \alpha_f \rangle_{\mathcal{F}^{(\text{finite})}}$ ensures that the ideal separating hyperplane is a feasible point of the training problem (5.187a).

Proof of Proposition 12. We establish the claim not for the shadow kernel itself ($k^{(\text{shadow})}(\cdot,\cdot)$), but for large finite-dimensional approximations ($k^{(\text{finite})}(\cdot,\cdot)$) thereof. We begin by utilizing Eq. (5.232) that approximates the nonlinear function $f(\rho)$ by a finite power series $f^{(d_p)}(\rho)$ with the approximation error,

$$|f(\rho) - f^{(d_p)}(\rho)| \leq 0.25. \tag{5.249}$$

For each $\ell \in \{1, \ldots, N\}$, we invoke Lemma 25 using the truncated Taylor series to conclude

$$\Pr\left[|f^{(d_p)}(\rho_\ell) - f^{(d_p)}\left(S_T(\rho_\ell)\right)| \geq 0.25 \right] \leq \delta/N, \tag{5.250}$$

provided that $T = (512/3) d_p^2 C^2 12^r \left(r \left(\log(n) + \log(12) \right) + \log(N/\delta) \right)$. Triangle inequality and a union bound allows us to combine these approximation guarantees into a single statement:

$$\max_{1 \leq \ell \leq N} \left| f(\rho_\ell) - f^{(d_p)}\left(S_T(\rho_\ell)\right) \right| \leq 0.5 \quad \text{with probability (at least) } 1 - \delta. \tag{5.251}$$

Let us condition on this desirable event and also assume hat the cutoff values of our finite kernel approximation are large enough, i.e. $D \geq d_p$, $R \geq r$). Then, the function

$$2f^{(d_p)}(S_T(\rho_\ell)) = \langle \alpha_f, \phi^{(\mathrm{finite})}(S_T(\rho_\ell))\rangle \tag{5.252}$$

describes a linear function in feature space $\mathcal{F}^{(\mathrm{finite})}$ that is guaranteed to achieve zero training error. Indeed, combine Eq. (5.231) and Eq. (5.251) to ensure

$$\left|2f^{(d_p)}(S_T(\rho_\ell))\right| \geq 2(|f(\rho_\ell)| - \left|f(\rho_\ell) - f^{(d_p)}(S_T(\rho_\ell))\right|) \geq 2(1-0.5) = 1 \tag{5.253}$$

and, moreover,

$$\mathrm{sign}\left(f^{(d_p)}(S_T(\rho_\ell))\right) = \mathrm{sign}\left(f(\rho_\ell)\right) = y(\rho_\ell) \in \{\pm 1\} \tag{5.254}$$

for all $\ell \in \{1, \ldots, N\}$. In turn,

$$\sum_{\ell=1}^{N} \max\left\{0, 1 - y(\rho_\ell)\langle \alpha_f, \phi^{(\mathrm{finite})}\left(S_T(\rho_\ell)\right)\rangle\right\} \tag{5.255a}$$

$$= \sum_{\ell=1}^{N} \max\left\{0, 1 - \mathrm{sign}\left(f^{(d_p)}(S_T(\rho_\ell))\right) 2f^{(d_p)}\left(S_T(\rho_\ell)\right)\right\} \tag{5.255b}$$

$$= \sum_{\ell=1}^{N} \max\left\{0, 1 - \left|f^{(d_p)}\left(S_T(\rho_\ell)\right)\right|\right\} = 0. \tag{5.255c}$$

Since zero is the smallest possible training error, the minimizer of the original training problem (5.187a) must also achieve zero, provided that α_f is actually a feasible point of this optimization. We can, however, ensure this by choosing the squared margin constant large enough. Eq. (5.229) and Assumption 1 ensures

$$\langle \alpha_f, \alpha_f\rangle_{\mathcal{F}^{(\mathrm{finite})}} \leq 4\,(2rn)^{rd_p}\, d_p^{d_p} C^2. \tag{5.256}$$

Choosing a squared margin size Λ^2 that exceeds this bound ensures that α_f is indeed a feasible point of the training problem (5.187a) and the claim follows. □

We conclude our discussion on training with shadow kernels by providing a rigorous proof of the auxiliary statement.

Proof of Lemma 25. It suffices to analyze implications of Lemma 15: for $\eta, \delta \in (0,1)$

$$T \geq (8/3)12^r\left(r\left(\log(n) + \log(12)\right) + \log(1/\delta)\right)/\eta^2 \tag{5.257}$$

$$\Rightarrow \max_{A\subset\{1,\dots,n\},|A|\leq r} \|\mathrm{tr}_{\neg A}\left(\sigma_T(\rho)\right) - \mathrm{tr}_{\neg A}(\rho)\|_1 \leq \eta \tag{5.258}$$

with probability at least $1-\delta$. Here, $\|\cdot\|_1$ denotes the trace norm. Abbreviate $\mathrm{tr}_{\neg A_i}\left(\sigma_T(\rho)\right)$ and $\mathrm{tr}_{\neg A_i}(\rho)$ as σ_{A_i} and ρ_{A_i}, respectively. A combination of triangle inequalities and Matrix Hoelder ($\mathrm{tr}(XY) \leq \|X\|_\infty \|Y\|_1$) asserts

$$\left|f^{(d_p)}(\rho) - f^{(d_p)}\left(S_T(\rho)\right)\right| \tag{5.259a}$$

$$\leq \sum_{d=0}^{d_p} \frac{1}{d!} \sum_{A_1,\dots,A_d\subset\{1,\dots,n\},|A_i|\leq r} \left|\mathrm{tr}\left(O_{A_1,\dots,A_r}\left(\rho_{A_1}\otimes\cdots\otimes\rho_{A_d} - \sigma_{A_1}\otimes\cdots\otimes\sigma_{A_d}\right)\right)\right| \tag{5.259b}$$

$$\leq \sum_{d=0}^{d_p} \frac{1}{d!} \sum_{A_1,\dots,A_d\subset\{1,\dots,n\},|A_i|\leq r} \left\|O_{A_1,\dots,A_d}\right\|_\infty \left\|\rho_{A_1}\otimes\cdots\otimes\rho_{A_d} - \sigma_{A_1}\otimes\cdots\otimes\sigma_{A_d}\right\|_1 . \tag{5.259c}$$

Next, we fix a trace norm contribution and use a telescoping trick ($A_1\otimes A_2 - B_1\otimes B_2 = (A_1-B_1)\otimes A_2 + B_1\otimes(A_2-B_2)$), as well as a triangle inequality and $\|\rho_{A_i}\|_1 = \mathrm{tr}(\rho_{A_i}) = 1$ to infer

$$\left\|\rho_{A_1}\otimes\ldots\otimes\rho_{A_d} - \sigma_{A_1}\otimes\ldots\otimes\sigma_{A_d}\right\|_1 \tag{5.260a}$$

$$= \left\|\left(\rho_{A_1}-\sigma_{A_1}\right)\otimes\rho_{A_2}\otimes\ldots\otimes\rho_{A_d} + \sigma_{A_1}\otimes\left(\rho_{A_2}\otimes\rho_{A_3}\ldots - \sigma_{A_2}\otimes\sigma_{A_3}\ldots\right)\right\|_1 \tag{5.260b}$$

$$\leq \|\rho_{A_1}-\sigma_{A_1}\|_1\|\rho_{A_2}\|_1\ldots\|\rho_{A_d}\|_1 + \|\sigma_{A_1}\|_1\left\|\rho_{A_2}\otimes\rho_{A_3}\ldots - \sigma_{A_1}\otimes\sigma_{A_3}\ldots\right\|_1 \tag{5.260c}$$

$$\leq \|\rho_{A_1}-\sigma_{A_1}\|_1 + \left(1+\|\rho_{A_1}-\sigma_{A_1}\|_1\right)\left\|\rho_{A_2}\otimes\ldots\otimes\rho_{A_d} - \sigma_{A_1}\otimes\ldots\otimes\sigma_{A_d}\right\|_1 \tag{5.260d}$$

$$\leq \eta + (1+\eta)\|\rho_{A_2}\otimes\ldots\otimes\rho_{A_d} - \sigma_{A_1}\otimes\sigma_{A_d}\|_1. \tag{5.260e}$$

The last line follows from Rel. (5.258). Iterating this simplification procedure ensures

$$\left\|\rho_{A_1}\otimes\cdots\otimes\rho_{A_d} - \sigma_{A_1}\otimes\cdots\otimes\sigma_{A_d}\right\|_1 \leq \eta\sum_{k=0}^{d-1}(1+\eta)^k = (1+\eta)^d - 1. \tag{5.261}$$

According to Rel. (5.258), such an upper bound is valid for every trace norm contribution in Eq. (5.259). This allows us to obtain

$$\left|f^{(d_p)}(\rho) - f^{(d_p)}\left(S_T(\rho)\right)\right| \tag{5.262a}$$

$$\leq \sum_{d=0}^{d_p} \frac{1}{d!} \sum_{A_1,\ldots,A_d \subset \{1,\ldots,n\}, |A_i| \leq r} \left\| O_{A_1,\ldots,A_d} \right\|_\infty \left[(1+\eta)^d - 1 \right] \tag{5.262b}$$

$$\leq \left[(1+\eta)^{d_p} - 1 \right] \sum_{d=0}^{d_p} \frac{1}{d!} \sum_{A_1,\ldots,A_d \subset \{1,\ldots,n\}, |A_i| \leq r} \left\| O_{A_1,\ldots,A_d} \right\|_\infty \tag{5.262c}$$

$$= C \left[(1+\eta)^{d_p} - 1 \right]. \tag{5.262d}$$

Here, we have used Assumption 1. Finally, by choosing $\eta = \epsilon/(2Cd_p)$, we can see that

$$\left| f^{(d_p)}(\rho) - f^{(d_p)}\left(S_T(\rho)\right) \right| \leq C \left[\left(1 + \frac{\epsilon}{2Cd_p} \right)^{d_p} - 1 \right] \leq C[\exp(\epsilon/2C) - 1] \leq \epsilon. \tag{5.263}$$

The second inequality follows from $(1 + x/n)^n \leq \exp(x), \forall |x| \leq n, n \geq 1$. The third inequality utilizes $\exp(x) \leq 1 + 2x, \forall x \in [0, 1]$. The claim of Lemma 25 now follows from inserting this specific choice of η into Rel. (5.258). □

Prediction based on shadow kernels

We now have all pieces in place to prove strong bounds on the prediction error of a SVM based on shadow kernels. The main result of this section will be a consequence of Theorem 22. For fixed parameters $\tau, \gamma = 1$, the shadow kernel (5.222b) (and finite approximations thereof) is always bounded when applied to classical shadows. Eq. (5.226) (under $\tau = \gamma = 1$) asserts

$$k^{(\text{shadow})}\left(S_T(\rho_1), \tilde{S}_T(\rho_2)\right) \leq \exp\left(\exp\left(5\right)\right) \tag{5.264}$$

for any T and quantum states ρ_1, ρ_2. This bound readily extends to finite dimensional approximations $k^{(\text{finite})}(\cdot,\cdot)$. Next, we need to specify a distribution. We assume that $\tilde{\mathcal{D}}$ is a distribution over n-qubit quantum states ρ that either belong to phase A or phase B. We sample quantum states $\rho_\ell \sim \tilde{\mathcal{D}}$ accordingly, but are not permitted to process them directly. Instead, we obtain a (randomly generated) classical shadow of size T. Denote the raw data by $S_T(\rho_\ell)$ which allows us to produce a state approximation $\sigma_T(\rho_\ell)$. We do, however, require that we have direct access to the label $y(\rho_\ell) \in \{\pm 1\}$ associated with the phase of ρ_ℓ. This produces a joint distribution over input data $S_T(\rho_\ell)$ and the label $y(\rho_\ell)$ which we call $\mathcal{D}$. In summary, we assume that training data and new data are generated independently from this data distribution: $\left(S_T(\rho_1), y(\rho_1)\right), \ldots, \left(S_T(\rho_N), y(\rho_N)\right), \left(S_T(\rho), y\right) \sim \mathcal{D}$. We are now ready to combine Theorem 22 (the prediction error is bounded by the training error) and Proposition 12 (the training error vanishes if a good phase classifier exists) to obtain a powerful result about generalization.

Corollary 1. *Fix* $\delta, \epsilon \in (0,1)$ *and suppose there exists an analytic function on reduced* r*-body density matrices that can distinguish phases:* $f(\rho) > 1$ *if* $\rho \in$ *phase* A *and* $f(\rho) < -1$ *else if* $\rho \in$ *phase* B*. Let* C *be the normalization constant and* d_p *be the truncation degree given in Assumption 1. Suppose that we obtain identically distributed training data* $(S_T(\rho_1), y(\rho_1)), \ldots, (S_T(\rho_N), y(\rho_N)) \sim \mathcal{D}$ *such that*

$$T \geq (512/3) d_p^2 C^2 12^r \left(r\left(\log(n) + \log(12)\right) + \log(N/\delta)\right) \quad \textit{and} \tag{5.265a}$$

$$N \geq 256\,(2rn)^{rd_p}\, d_p^{d_p} C^2 \exp(\exp(5)) \log(4/\delta)/\epsilon^2. \tag{5.265b}$$

Then, solving the training problem (5.187a) *for the shadow kernel with squared margin constant* $\Lambda^2 = 4\,(2rn)^{rd_p}\, d_p^{d_p} C^2$ *will produce a hyperplane* $\alpha_\sharp \in \mathbb{R}^N$ *in shadow feature space that achieves zero training error with probability (at least)* $1 - \delta/2$*. Conditioned on perfect training, the resulting classifier*

$$y_\sharp\left(S_T(\rho)\right) = \operatorname{sign}\Big(\sum_{\ell=1}^{N} \left[\alpha_\sharp\right]_\ell k^{(\text{shadow})}(S_T(\rho_\ell), S_T(\rho))\Big) \in \{\pm 1\} \tag{5.266}$$

achieves, with probability (at least) $1 - \delta/2$,

$$\Pr_{(S_T(\rho), y(\rho))}\left[y_\sharp\left(S_T(\rho)\right) \neq y(\rho)\right] \leq \epsilon\,. \tag{5.267}$$

The total probability of success is (at least) $1-\delta$ and follows from a union bound over either desirable event failing. Theorem 1 is contingent on four core assumptions:

1. It must be possible to distinguish phases A and B by evaluating a well-conditioned analytical function on reduced r-body density matrices. The coefficients in the power series of the analytical function should also be bounded, but explicit knowledge is *not* necessary. This is the content of Assumption 1.

2. We use classical shadow raw data to read-in training data ($\rho_\ell \mapsto S_T(\rho_\ell)$) and process new states in the prediction phase ($\rho \mapsto S_T(\rho)$). We assume that each classical shadow arises from T randomized single-qubit Pauli measurements on independent state copies. The larger T, the more accurate these representations become. Theorem 1 requires

$$T \geq (512/3) d_p^2 C^2 12^r \left(r\left(\log(n) + \log(12)\right) + \log(N/\delta)\right) \tag{5.268}$$

$$= O\left(r12^r d_p^2 C^2 \log(nN/\delta)\right). \tag{5.269}$$

If r, C, d_p are constants, this resolution only scales polylogarithmically in system size n because N scales polynomially in n; see the next bullet point.

3. The training data size must not be too small either. We need to have a training data size N of order at least $(2rn)^{rd_p} d_p^{d_p} C^2 \exp(\exp(5)) \log(4/\delta)/\epsilon^2$. As long as r, C, d_p are constants (independent of system size n), this requirement simplifies to $N = O\left(n^{rd_p} \log(1/\delta)/\epsilon^2\right)$. Hence, the number scales polynomially in system size n.

4. The squared margin constant also scales polynomially with system size n: $\Lambda^2 = 4\,(2rn)^{rd_p} d_p^{d_p} C^2 = O\left(n^{rd_p}\right)$ if $r, C, d_p = \text{const}$. This is equivalent to demanding that the minimal margin $2/\Lambda$ scales inverse polynomially in system size n.

Corollary 1 does not only bound a hypothetical training error. The required shadow size T and training data size N both scale favorably in the number of qubits n. This also ensures that the numerical costs behind this procedure remain tractable for a wide range of system sizes. The costs associated with storage (classical shadows are sums of T elementary tensor products), training (can be reduced to a QCQP in N dimensions per Section 5.9) and prediction (execute Formula (5.191)) all scale polynomially in system size n, shadow size T, and training data size N.

Proof of Corollary 1. Again, we establish the claim for large, but finite-dimensional, approximations to the shadow kernel ($1 \leq d_p \ll D < \infty$ and $1 \leq r \ll R < \infty$). Fix $\delta \in (0,1)$ (probability of failure) and $\epsilon \in (0,1)$ (bound on average prediction error). Consider the data distribution $(S_T(\rho), y(\rho)) \sim \mathcal{D}$, the kernel $k^{(\text{finite})}(\cdot,\cdot)$ – which obeys $k^{(\text{finite})}\left(S_T(\rho), S_T(\rho)\right) \leq \exp\left(\exp(5)\right)$ – and a squared margin constant Λ^2 to be specified later. Assume $\Lambda^2 \exp(\exp(5)) \geq 1$ for simplicity (the other case is similar). Then, for training data size N, Theorem 22 asserts

$$\Pr_{(S_T(\rho), y(\rho)) \sim \mathcal{D}}\left[y_\sharp\left(S_T(\rho)\right) \neq y(\rho)\right] \leq \frac{1}{N}\mathrm{E}_{\mathrm{tr}}(\alpha_\sharp) + 8\sqrt{\Lambda^2 \exp(\exp(5))\frac{\log(4/\delta)}{N}}, \tag{5.270}$$

with probability (at least) $1 - \delta/2$. Choosing N large enough allows us to suppress the second contribution beneath the desired approximation error bound:

$$N \geq 64\Lambda^2 \exp(\exp(5)) \log(4/\delta)/\epsilon^2 \tag{5.271}$$

$$\Rightarrow \quad \Pr_{(S_T(\rho),y(\rho))\sim\mathcal{D}}\left[y_\sharp\left(S_T(\rho)\right) \neq y(\rho)\right] \leq \frac{1}{N}\mathrm{E}_{\mathrm{tr}}(\alpha_\sharp) + \epsilon, \tag{5.272}$$

with probability (at least) $1 - \delta/2$. Here, $\mathrm{E}_{\mathrm{tr}}(\alpha_\sharp)$ is the training error obtained from solving problem (5.187a) for N independently sampled training data points $(S_T(\rho_1), y(\rho_1)), \ldots, (S_T(\rho_N), y(\rho_N)) \sim \mathcal{D}$. Proposition 12 asserts that this training error can vanish with high probability, provided that a well-conditioned analytical function on reduced r-body density matrices exists that can distinguish the phases (see Assumption 1). The classical shadow size T and the squared margin constant Λ^2 depend on the number of body r, the normalization constant C, and the truncation degree d_p of this classifier:

$$\left.\begin{array}{rcl} T & \geq & (512/3)d_p^2C^2 12^r\left(r\left(\log(n)+\log(12)\right)+\log(N/\delta)\right) \\ \Lambda & \geq & 4\left(2rn\right)^{rd_p} d_p^{d_p} C^2 \end{array}\right\} \quad \Rightarrow \mathrm{E}_{\mathrm{tr}}(\alpha_\sharp) = 0 \tag{5.273}$$

with probability (at least) $1-\delta/2$. The claim now follows from inserting this squared margin size into the expression (5.272) for training data size. □

5.10 Classifying SPT phases with O(2) symmetry

Symmetry-protected topological phases

We consider a scenario similar to that of Section 5.4, namely, a family of Hamiltonians $H(x)$ parameterized by x. We additionally enforce that $H(x)$ be invariant under certain symmetry transformations, which can include tensor products of on-site rotations, "spatial" transformations permuting the sites, or antiunitary maps characterizing time-reversal. These additional symmetry constraints allow for a fine-grained characterization of $H(x)$ into various symmetry-protected topological (SPT) phases. Removing said constraints reduces this characterization to the coarser one involving purely topological phases. Similar to the coarser characterization, ground states of $H(x)$ remain in a particular SPT when the parameters x are varied continuously, as long as the spectral gap of the Hamiltonian remains finite. In other words, the gap has to close at some x in order for the ground states to transition into another phase. When there is a constant spectral gap, it is expected that an operator acting on a local region larger than some constant size independent of the full system size n can classify different SPT phases. The existence of a classifying function of local density matrices has been rigorously established for a handful of cases: $U(1)$-symmetric systems in two dimensions (either noninteracting fermionic (Alexei Yu. Kitaev, 2006; Y. Zhang and E.-A. Kim, 2017) or interacting (Matthew B. Hastings and Michalakis, 2015; Kapustin and Sopenko, 2020; Bachmann, Bols, et al., 2020)), and certain spin-1 chains in one dimension (Bachmann and Nachtergaele, 2014; Tasaki, 2018; Tasaki, 2020).

SPT phases of one-dimensional spin chains with unique ground states, symmetric under tensor-product unitaries forming a symmetry group G, are in one-to-one correspondence with the various projective representations realized by G (X. Chen, Z.-C. Gu, and Wen, 2011). Projective representations are those in which the group's multiplication table is decorated with phases in a way that is consistent with associativity (Arovas, n.d.). A genuine (i.e., linear) representation corresponds to the unique trivial projective representation.

Consider, for example, spin chains symmetric under $G = SO(3)$. This group admits two distinct classes of projective representations: one class corresponds to integer spin, and one corresponds to half-integer spin. Thus, there are two different phases for such chains — the trivial phase and the "Haldane phase" (F. Haldane, 1983; X. Chen, Z.-C. Gu, and Wen, 2011).

Relaxing the symmetry group down to its $O(2)$ subgroup maintains the two-phase classification, because $O(2)$ also admits two projective representations (X. Chen, Z.-C. Gu, Z.-X. Liu, et al., 2013). In fact, one can relax the symmetry all the way down to the simplest dihedral subgroup $Z_2 \times Z_2$ (Z.-C. Gu and Wen, 2009; Pollmann, Ari M. Turner, et al., 2010); such a classification is similar to that of the model in Appendix 5.13. We investigate systems admitting the larger $O(2)$ symmetry below, noting that the work we rely on (Tasaki, 2018; Tasaki, 2020) also studies symmetry groups that include spatial inversion and time reversal.

$O(2)$-symmetric qutrit spin chains

The representative states for each of the two $O(2)$-symmetric phases for qutrit spin chains are the product state, representing the trivial phase, and the valence-bond-solid (VBS) state (Affleck, Kennedy, et al., 1988), admitting a projective representation of the symmetry (Tasaki, 2020) and thus representing the Haldane phase. It has long been known that the expectation value of a nonlocal "twist" operator O_L (Totsuka and Suzuki, 1995; Nakamura and Todo, 2002) distinguishes these two representative states: $\text{sign}(\langle O_L \rangle)$ is +1 for the product state, and -1 for the VBS state. We will see later that, by continuity arguments, this sign will stay constant for other states within the same phase.

In order to work efficiently, our phase classification algorithms require a *local* operator whose expectation value (a) has the same sign as that of O_L; and (b) is above or below a margin (here, $1/2$), in order to determine the required accuracy of the classical shadows. Recently, criterion (a) was explicitly demonstrated by Tasaki (Tasaki, 2018; Tasaki, 2020) using a local version O_ℓ of the twist, see Eq. (5.278) below. We collect relevant parts of his results to prove both criteria in the theorem below. Due to the existence of a local operator for classifying the SPT phases, our ML algorithms are guaranteed to predict the SPT phases accurately based on the proof given in Appendix 5.9.

Theorem 24. *Consider the triple $\{H(x), |\psi(x)\rangle, \Delta(x)\}$ containing $(2L+2)$-site spin-one chains with periodic boundary conditions*

$$H(x) = \sum_{j=-(L-r)}^{L-r+1} h_j(x) + h_{-L}(x) + h_{L+1}(x) \tag{5.274}$$

that admit corresponding unique ground states $|\psi(x)\rangle$ and spectral gaps $\Delta(x) \geq \gamma = \Omega(1)$, bounded interaction strength $\left\|h_j(x)\right\|_\infty \leq R = O(1)$, and whose terms $h_j(x)$ are supported on sites k such that $|j-k| \leq r = O(1)$. Assume that $H(x)$ is $O(2)$-symmetric, with the symmetry group generated by

1. *a collective z-axis rotation by any angle, and*
2. *an x-axis rotation by π.*

There exists a few-body observable A, such that for all x, we have

$$\operatorname{sign}(\langle\psi(x)| A |\psi(x)\rangle) = \operatorname{sign}(\langle\psi(x)| O_L |\psi(x)\rangle), \quad \textit{as well as} \tag{5.275a}$$

$$|\langle\psi(x)| A |\psi(x)\rangle| \geq 1/2\,. \tag{5.275b}$$

Proof. We use spin-one operators $S^{(\alpha)}$ with $\alpha \in \{x, y, z\}$ that have eigenvalues $\{0, \pm 1\}$ and satisfy angular-momentum commutation relations $[S^{(x)}, S^{(y)}] = \mathrm{i}S^{(z)}$. Eigenstates of $S^{(z)}$ are denoted by $|\sigma\rangle$ with $\sigma \in \{0, \pm 1\}$. A rotation around axis α is a unitary operator generated by the corresponding $S^{(\alpha)}$. The two symmetry group generators are, for $\theta \in [0, 2\pi)$,

$$U(\theta) = \bigotimes_{j=-L}^{L+1} e^{-\mathrm{i}\theta S_j^{(z)}} \qquad \text{and} \qquad V = \bigotimes_{j=-L}^{L+1} e^{-\mathrm{i}\pi S_j^{(x)}}\,. \tag{5.276}$$

By assumption, both symmetries commute with each Hamiltonian term h_j; we will explicitly use both to prove the theorem. We will also need superimposed versions

$S^{(\pm)} = S^{(x)} \pm \mathrm{i}S^{(y)}$, which satisfy

$$e^{\mathrm{i}\phi S^{(z)}} S^{(\pm)} e^{-\mathrm{i}\phi S^{(z)}} = S^{(\pm)} e^{\pm \mathrm{i}\phi} \,. \tag{5.277}$$

The family of unitary twist operators (Affleck and Elliott H. Lieb, 1986), acting on an interval of 2ℓ spins centered at the origin, is

$$O_\ell = \bigotimes_{k\,,\left|k-\frac{1}{2}\right|\le \ell+\frac{1}{2}} \exp\left(-\mathrm{i}2\pi \frac{k+\ell}{2\ell+1} S_k^{(z)}\right) \,. \tag{5.278}$$

Each site's rotation is by a multiple of $2\pi/(2\ell+1)$ that is proportional to the site index, forming the namesake twist pattern. The $\ell = L$ case reduces to the aforementioned nonlocal twist operator O_L, while $\ell \ll L$ are its local versions.

Suppressing x dependence, the key property is that the twisted ground state $O_\ell|\psi\rangle$ has energy close to that of the ground state. In particular, there exists $C_0, C_1 > 0$, such that for all $\ell \ge C_0$, Lemma 26 below yields

$$\langle\psi|O_\ell H O_\ell^\dagger|\psi\rangle - \langle\psi|H|\psi\rangle \le \frac{C_1}{\ell} \,. \tag{5.279}$$

The ground state is unique by our assumption of a gap, so the twisted ground state must then become proportional to the ground state as $\ell \to \infty$. In other words, the magnitude of their overlap must be close to one as long as $\ell \ge C_0$,

$$|\langle\psi|O_\ell|\psi\rangle|^2 \ge 1 - \frac{C_1}{\Delta\ell} \,; \tag{5.280}$$

see Lemma 27 below. The phase of this overlap is either 0 or π because the π-rotation V leaves the ground state invariant:

$$\langle\psi|O_\ell|\psi\rangle = \langle\psi|V^\dagger O_\ell V|\psi\rangle = \langle\psi|O_\ell^\dagger|\psi\rangle = \overline{\langle\psi|O_\ell|\psi\rangle} \in \mathbb{R} \,. \tag{5.281}$$

Hence, the few-body Hermitian observable $A = (O_\ell + O_\ell^\dagger)/2$ with

$$\ell = \max(4\gamma/(3C_1), C_0) \tag{5.282}$$

satisfies

$$|\langle\psi|A|\psi\rangle| = |\langle\psi|O_\ell|\psi\rangle| \ge \sqrt{1 - \frac{C_1}{\Delta\ell}} \ge \frac{1}{2} \,, \tag{5.283}$$

proving Eq. (5.275b). Note that the required value of ℓ depends on the gap, and thus also on x.

To prove Eq. (5.275a), we need to show that the sign of the twist's expectation value remains the same for any $\ell \ge \max(4\gamma/(3C_1), C_0)$. To do this, first notice that, when ℓ is relaxed to be a nonnegative real, the twist (5.278) is *continuous* in ℓ. (This can be verified, e.g., by studying the twist's eigenvalues.) Continuity implies that the expectation value cannot change sign; otherwise, it would have to cross zero, thus violating Eq. (5.283). Therefore, the sign remains the same, confirming Eq. (5.275a). Similarly, by continuity in ℓ and x, the expectation value maintains its sign within each phase. □

The above argument is contingent on two auxiliary statements, which we now prove.

Lemma 26 (Vanishing energy difference (Tasaki, 2018); Eq. (5.279))**.** *For constants C_0, C_1, as long as $\ell \geq C_0$, we have*

$$\langle\psi|O_\ell H O_\ell^\dagger|\psi\rangle - \langle\psi|H|\psi\rangle \leq \frac{C_1}{\ell}. \tag{5.284}$$

Proof. Using the variational principle (which says that the difference in energy between any state and the ground state is nonnegative), plugging in O_ℓ and H, applying $\langle\psi|O|\psi\rangle \leq \|O\|_\infty$, and distributing the norm over the sum yields

$$\begin{aligned}\langle\psi|O_\ell H O_\ell^\dagger|\psi\rangle - \langle\psi|H|\psi\rangle &\leq \langle\psi|\left(O_\ell H O_\ell^\dagger + O_\ell^\dagger H O_\ell - 2H\right)|\psi\rangle && (5.285\text{a})\\ &= \sum_{j=-(\ell+r)}^{\ell+r+1} \langle\psi|\left(O_\ell h_j O_\ell^\dagger + O_\ell^\dagger h_j O_\ell - 2h_j\right)|\psi\rangle && (5.285\text{b})\\ &\leq \sum_{j=-(\ell+r)}^{\ell+r+1} \left\|O_\ell h_j O_\ell^\dagger + O_\ell^\dagger h_j O_\ell - 2h_j\right\|_\infty && (5.285\text{c})\end{aligned}$$

Next, we use the finite support and rotational invariance of h_j from Eq. (5.276) to rotate the twist O_ℓ,

$$\begin{aligned}O_\ell h_j O_\ell^\dagger &= O_\ell U\left(\theta_j\right) h_j U^\dagger\left(\theta_j\right) O_\ell^\dagger && (5.286\text{a})\\ &= \left(\bigotimes_{|k-j|\leq r} e^{-\mathrm{i}\left(\frac{2\pi}{2\ell+1}[k+\ell]+\theta_j\right)S_k^{(z)}}\right) h_j \left(\bigotimes_{|k-j|\leq r} e^{\mathrm{i}\left(\frac{2\pi}{2\ell+1}[k+\ell]+\theta_j\right)S_k^{(z)}}\right), && (5.286\text{b})\end{aligned}$$

where we pick $\theta_j = -\frac{2\pi}{2\ell+1}\left(j+\ell\right)$ for each j. That way, the twist does not affect site j, with

$$O_\ell h_j O_\ell^\dagger = e^{\mathrm{i}\frac{2\pi}{2\ell+1}M_j} h_j e^{-\mathrm{i}\frac{2\pi}{2\ell+1}M_j}, \qquad \text{and} \qquad M_j = \sum_{|k-j|\leq r}\left(j-k\right)S_k^{(z)}. \tag{5.287}$$

We now expand h_j as a polynomial in $\{S_k^{(z)}, S_k^{(\pm)}\}$. This can be done because products of powers of these operators form a matrix basis for any operator on the chain. For a single site, the set $\{S^{(z)}S^{(\pm)}, (S^{(+)})^2\}$, along with their complex conjugates and some powers of $S^{(z)}$, form the basis of nine matrix units for all 3×3 operators on the site. Tensor products of these operators therefore form a matrix-unit basis for all sites. The conjugation property (5.277) and Eq. (5.287) imply that each term in the expansion of h_j, upon conjugation by O_ℓ, will be imparted with a phase that is some multiple μ of $2\pi/(2\ell+1)$. Combining all terms with the same phase into $h_{j,\mu}$, we have

$$e^{\mathrm{i}\frac{2\pi}{2\ell+1}M_j} h_{j,\mu} e^{-\mathrm{i}\frac{2\pi}{2\ell+1}M_j} = h_{j,\mu} e^{\mathrm{i}\frac{2\pi}{2\ell+1}\mu} \,. \tag{5.288}$$

Moreover, $|\mu| \leq 2\mu_{\max}$, where $\mu_{\max} = \sum_{|k-j|\leq r} |j-k| = r\,(r+1)$ is the largest eigenvalue of M_j. Plugging this in and expanding the resulting cosine yields

$$\left\| O_\ell h_j O_\ell^\dagger + O_\ell^\dagger h_j O_\ell - 2h_j \right\|_\infty = 2 \left\| \sum_{|\mu|\leq 2r(r+1)} \left[\cos\left(\frac{2\pi}{2\ell+1}\mu\right) - 1 \right] h_{j,\mu} \right\|_\infty \tag{5.289a}$$

$$\leq \left(\frac{2\pi}{2\ell+1}\right)^2 \sum_{|\mu|\leq 2r(r+1)} \mu^2 \left\| h_{j,\mu} \right\|_\infty \,. \tag{5.289b}$$

Since the spin operators form a matrix-unit basis, each $h_{j,\mu}$ is simply h_j with some entries removed. Therefore, the norm of $h_{j,\mu}$ is bounded by R. Applying that and performing the remaining sum (5.285c) over j yields

$$\langle\psi|O_\ell H O_\ell^\dagger|\psi\rangle - \langle\psi|H|\psi\rangle \leq \frac{\ell+r+1}{(2\ell+1)^2} 4\pi^2 R \left(\sum_{|\mu|\leq 2r(r+1)} \mu^2 \right) . \tag{5.290}$$

Thus, for $\ell \geq C_0$, the difference in energies between the ground state and twisted ground state will be bounded by C_1/ℓ, where C_0, C_1 are two constants depending on the interaction range r and norm bound R of the Hamiltonian terms. □

Lemma 27 (High overlap (Tasaki, 2020); Eq. (5.280))**.** *For constants C_0, C_1, as long as $\ell \geq C_0$, we have*

$$|\langle\psi|O_\ell|\psi\rangle|^2 \geq 1 - \frac{C_1}{\Delta\ell} \,. \tag{5.291}$$

Proof. All eigenvalues of H are bounded below by the sum of the ground state energy $E_{\mathrm{gnd}} = \langle\psi|H|\psi\rangle$ and spectral gap Δ,

$$H \geq E_{\mathrm{gnd}}|\psi\rangle\langle\psi| + \left(E_{\mathrm{gnd}} + \Delta\right)\left(\mathbb{I} - |\psi\rangle\langle\psi|\right) = E_{\mathrm{gnd}}\mathbb{I} + \Delta\left(\mathbb{I} - |\psi\rangle\langle\psi|\right) \,. \tag{5.292}$$

Conjugating by O_ℓ and evaluating the result in the ground state yields

$$\langle\psi|O_\ell H O_\ell^\dagger|\psi\rangle \geq E_{\mathrm{gnd}} + \Delta\left(1 - |\langle\psi|O_\ell|\psi\rangle|^2\right). \tag{5.293}$$

Rearranging this and plugging in Lemma 26 yields the desired result. □

5.11 Neural networks with classical shadow for quantum problems

Imposing inductive biases in the ML model is a common technique for boosting the prediction performance of ML models. One approach is to enhance the proposed ML algorithms with neural networks, such as convolutional or graph neural networks. These neural networks could better capture structure of the underlying function we are trying to learn and hence may require significantly less data than the very expressive ML model given in the main text. We leave the proof that neural network enhancements can lead to better prediction performance as a goal for future work.

There are multiple ways of combining classical shadows and neural networks. Here, we will only showcase one such approach by utilizing the theory of neural tangent kernels (Jacot, Gabriel, and Hongler, 2018). Remarkably, this theory allows us to efficiently train various types of neural networks (convolutional/graph/etc.) with an infinite number of neurons in each hidden layer (*infinite width*). As such, this line of work has gained a lot of attention (Du et al., 2019; Arora et al., 2019; Novak, L. Xiao, Hron, J. Lee, Alexander A. Alemi, et al., 2020) in recent years. In the limit of infinite width, one can analytically solve for the neural network after training on a set of data $\{x_\ell, y_\ell\}_{\ell=1}^N$, where x_ℓ and y_ℓ are vectors of some size. For example, consider training a neural network that takes in a vector x and produces a vector $f_\theta^{\mathrm{NN}}(x)$ through the following optimization problem using gradient descent,

$$\min_\theta \sum_{\ell=1}^N \left\|f_\theta^{\mathrm{NN}}(x_\ell) - y_\ell\right\|_2^2, \tag{5.294}$$

where we begin on a randomly initialized θ. Note that due to the infinite number of neurons, θ is a vector of infinite dimension. The trained neural network $f_{\theta^*}^{\mathrm{NN}}(x)$ can always be written in the following form

$$f_{\theta^*}^{\mathrm{NN}}(x) = \sum_{\ell=1}^N \sum_{\ell'=1}^N k^{(\mathrm{NTK})}(x, x_\ell)(K^{-1})_{\ell\ell'} y_{\ell'}, \tag{5.295}$$

where $k^{(\mathrm{NTK})}(x, x')$ is a function called the neural tangent kernel (Jacot, Gabriel, and Hongler, 2018), and $K_{\ell,\ell'} = k^{(\mathrm{NTK})}(x_\ell, x_{\ell'})$ is the kernel matrix of the neural tangent kernel. One can see that the infinite-dimensional vector θ^* does not appear on the right hand side of Eq. (5.295). And as long as we can efficiently evaluate the neural tangent kernel $k^{(\mathrm{NTK})}(x, x')$, we can evaluate the infinite-dimensional neural network in polynomial time. This is the main contribution of (Jacot, Gabriel, and Hongler, 2018), which enables one to efficiently train infinite-width neural networks. For a given neural network architecture, one can compute $k^{(\mathrm{NTK})}(x, x')$ efficiently using open-source software, such as (Novak, L. Xiao, Hron, J. Lee, Alexander A. Alemi, et al., 2020). In Appendix 5.13, we give the code for training infinite-width neural networks using the open-source software: Neural Tangents (Novak, L. Xiao, Hron, J. Lee, Alexander A. Alemi, et al., 2020).

Predicting ground state representation

For the task of predicting ground state representation, we consider the training data to be

$$\left\{x_\ell \to \sigma_T(\rho(x_\ell))\right\}_{\ell=1}^{N}, \tag{5.296}$$

where $\sigma_T(\rho(x_\ell))$ is the classical shadow representation of $\rho(x_\ell)$ given in Eqs. (5.3) based on T randomized Pauli measurements. Recall that $\sigma_T(\rho(x_\ell))$ is a $2^n \times 2^n$ matrix that reproduces $\rho(x_\ell)$ in expectation over the randomized Pauli measurements. Suppose we now train an infinite-width neural network parameterized by θ that takes in an input x and produces an exponential-size matrix $\sigma_\theta^{\mathrm{NN}}(x)$, by solving the optimization problem

$$\min_\theta \quad \sum_{\ell=1}^{N} \left\| \sigma_\theta^{\mathrm{NN}}(x_\ell) - \sigma_T(\rho(x_\ell)) \right\|_F^2. \tag{5.297}$$

The squared Frobenius difference between two matrices is equal to the squared Euclidean norm of their vectorizations (flattenings). In turn, the theory of infinite-width neural networks (Jacot, Gabriel, and Hongler, 2018) shows that the trained neural network $\sigma_{\theta^*}^{\mathrm{NN}}(x)$ could be written in the form

$$\sigma_{\theta^*}^{\mathrm{NN}}(x) = \sum_{\ell=1}^{N} \sum_{\ell'=1}^{N} k^{(\mathrm{NTK})}(x, x_\ell)(K^{-1})_{\ell\ell'} \sigma_T(\rho(x_{\ell'})). \tag{5.298}$$

The kernel function $k^{(\mathrm{NTK})}(x, x')$ depends on the neural network architecture and could be calculated utilizing existing open-source software (Novak, L. Xiao, Hron, J. Lee, Alexander A. Alemi, et al., 2020). This also falls into the general form shown in the main text; see Eq. (5.20). Hence, training an infinite-width neural network to predict an exponentially large density matrix can be done efficiently on a classical computer. For a given neural network architecture, all one has to do is compute the kernel function $k^{(\mathrm{NTK})}(x, x')$. Then the neural network optimized using the training data could be analytically solved as given in Eq. (5.298). To estimate a property on the predicted ground state using the neural network is as simple as evaluating

$$\operatorname{tr}(O\sigma^{\mathrm{NN}}_{\theta^*}(x)) = \sum_{\ell=1}^{N}\sum_{\ell'=1}^{N} k^{(\mathrm{NTK})}(x, x_\ell)(K^{-1})_{\ell\ell'} \operatorname{tr}(O\sigma_T(\rho(x_{\ell'}))), \tag{5.299}$$

which can be done by first computing $\operatorname{tr}(O\sigma_T(\rho(x_\ell))), \forall \ell = 1, \ldots, N$ and compute the linear interpolation.

Classifying phases of matter

We want to learn how to classify two phases of n-qubit states. A fully classical training set would simply consist of N labeled classical representations of quantum states $\{\rho_\ell \to y_\ell\}_{\ell=1}^{N}$, where $y_\ell = +1$ (-1) if ρ_ℓ belongs to phase A (B). However, insisting on perfect knowledge of each ρ_ℓ is impractical for a variety of reasons. Instead, we assume that we have access to classical shadows of ρ_ℓ. The raw data $S_T(\rho_\ell)$ behind each classical shadow is a 2-dimensional array,

$$S_T(\rho_\ell) = \left\{ |s_i^{(t)}\rangle \, : \; i \in \{1, \ldots, n\}, t \in \{1, \ldots, T\} \right\} \tag{5.300}$$

$$\text{where} \quad |s_i^{(t)}\rangle \in \{|0\rangle, |1\rangle, |{+}\rangle, |{-}\rangle, |\mathrm{i}{+}\rangle, |\mathrm{i}{-}\rangle\}. \tag{5.301}$$

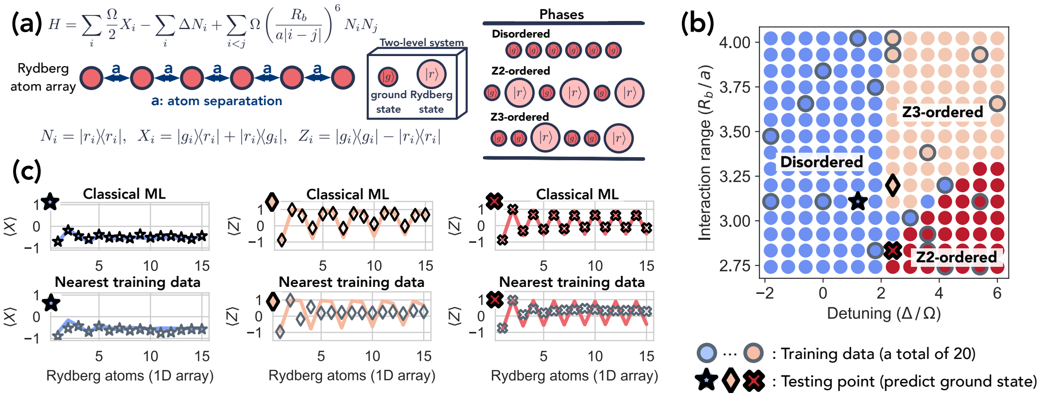
In the main text, we propose to use this data to train a support vector machine based on the shadow kernel

$$k^{(\mathrm{shadow})}\left(S_T(\rho_\ell), \tilde{S}_T(\rho_{\ell'})\right) \tag{5.302}$$

$$= \exp\left(\frac{\tau}{T^2}\sum_{t,t'=1}^{T} \exp\left(\frac{\gamma}{n}\sum_{i=1}^{n} \operatorname{tr}\left(\left(3|s_i^{(t)}\rangle\!\langle s_i^{(t)}| - \mathbb{I}\right)\left(3|\tilde{s}_i^{(t)}\rangle\!\langle \tilde{s}_i^{(t)}| - \mathbb{I}\right)\right)\right)\right). \tag{5.303}$$

This specific choice of (deterministic) kernel function allows us to carry out a thorough theoretical analysis of the entire learning procedure; see Appendix 5.9.

But there are other sensible kernels that may perform even better in practice. For instance, we could feed the two-dimensional data array (5.301) into a neural network architecture, e.g. a convolutional neural network. In the limit of an infinite number of neurons in each hidden layer, this produces the neural tangent kernel $k^{(\mathrm{NTK})}\left(S_T(\rho_\ell), \tilde{S}_T(\rho_{\ell'})\right)$ (Jacot, Gabriel, and Hongler, 2018). This kernel is positive-semidefinite and should be viewed as a measure of similarity induced by the trained neural network. Mercer's theorem (Mercer, 1909) allows us to make this intuition precise by reformulating the neural tangent kernel as a Gram matrix in



*Solid lines in the six line plots indicate exact values from DMRG

Figure 5.4: Numerical experiment for predicting ground-state properties in a 1D Rydberg atom system with 51 atoms. (a) Hamiltonian. Illustration of the Rydberg array geometry, Hamiltonian, and phases. (b) Phase diagram. The system's three distinct phases (Bernien et al., 2017) are characterized by two order parameters (for Z_2 and Z_3 orders). Training data are enclosed by gray circles, and three specific testing points are indicated by the star, diamond, and cross, respectively. (c) Local expectation values. We use classical ML (the best model is selected from a set of ML models) to predict the expectation values of Pauli operators X_i and Z_i for each atom at the three testing points. We compare with "predictions" obtained from the training data nearest to the testing points. The markers denote predicted values, while the solid lines denote exact values obtained from DMRG. Additional predictions are shown in Appendix 5.13.

feature space:

$$k^{(\mathrm{NTK})}\left(S_T(\rho_\ell), \tilde{S}_T(\rho_{\ell'})\right) = \left\langle \phi^{(\mathrm{NTK})}\left(S_T(\rho_\ell)\right), \phi^{(\mathrm{NTK})}\left(\tilde{S}_T(\rho_{\ell'})\right) \right\rangle . \tag{5.304}$$

Hence, any infinite-width neural network with input array $S_T(\rho)$ induces a feature map $\phi^{(\mathrm{NTK})}$ that can be used instead of the doubly-infinite feature map $\phi^{(\mathrm{shadow})}$ (5.159) that is associated with the shadow kernel (5.303).

5.12 Numerical experiments

We have conducted numerical experiments assessing the performance of classical ML algorithms in some practical settings. The results demonstrate that our theoretical claims carry over to practice, with the results sometimes turning out even better than our guarantees suggest.

Predicting ground state properties

For predicting ground states, we consider classical ML models encompassed by Eq. (5.20). We examine various metrics $\kappa(x, x_\ell)$ equivalent to training neural networks with large hidden layers (Jacot, Gabriel, and Hongler, 2018; Novak, L. Xiao, Hron, J. Lee, Alexander A. Alemi, et al., 2020) or training kernel methods (Cortes and Vapnik, 1995; Murphy, 2012). We find the best ML model and the hyperparameters using a validation set to minimize root-mean-square error (RMSE) and report the predictions on a test set. The full details of the models and hyperparameters, as well as their comparison, are given in Appendix 5.13 and 5.13.

Rydberg atom chain — Our first example is trapped Rydberg atoms (Fendley, Sengupta, and Sachdev, 2004; Browaeys and Lahaye, 2020), a programmable and highly controlled platform for Ising-type quantum simulations (Schauß et al., 2015; Endres et al., 2016; Bernien et al., 2017; Labuhn et al., 2016; Ebadi et al., 2020; Scholl et al., 2020). Following (Bernien et al., 2017), we consider a one-dimensional array of $n = 51$ atoms, with each atom effectively described as a two-level system composed of a ground state $|g\rangle$ and a highly-excited Rydberg state $|r\rangle$. The atomic chain is characterized by a Hamiltonian $H(x)$ (given in Figure 5.4(a)) whose parameters are the laser detuning $x_1 = \Delta/\Omega$ and the interaction range $x_2 = R_b/a$. The phase diagram (shown in Figure 5.4(b)) features a disordered phase and several broken-symmetry phases, stemming from the competition between the detuning and the Rydberg blockade (arising from the repulsive Van der Waals interactions).

We trained a classical ML model using 20 randomly chosen values of the parameter $x = (x_1, x_2)$; these values are indicated by gray circles in Figure 5.4(b). For each such x, an approximation to the exact ground state was found using DMRG (Steven R. White, 1992) based on the formalism of matrix product states (MPS) (Schollwoeck, 2011). For each MPS, we performed $T = 500$ randomized Pauli measurements to construct a classical shadow. The classical ML then predicted classical representations at the testing points in the parameter space, and these predicted classical representations were used to estimate expectation values of local observables at the testing points.

Predictions for expectation values of Pauli operators Z_i and X_i at the testing points are shown in Figure 5.4(c), and found to agree well with exact values obtained from the DMRG computation of the ground state at the testing points. Additional predictions can be found in Appendix 5.13. Also shown are results from a more naive procedure, in which properties are predicted using only the data at the point in the training set which is closest to the testing point. The naive procedure predicts poorly, illustrating that the considered classical ML model effectively leverages the data from multiple points in the training set.

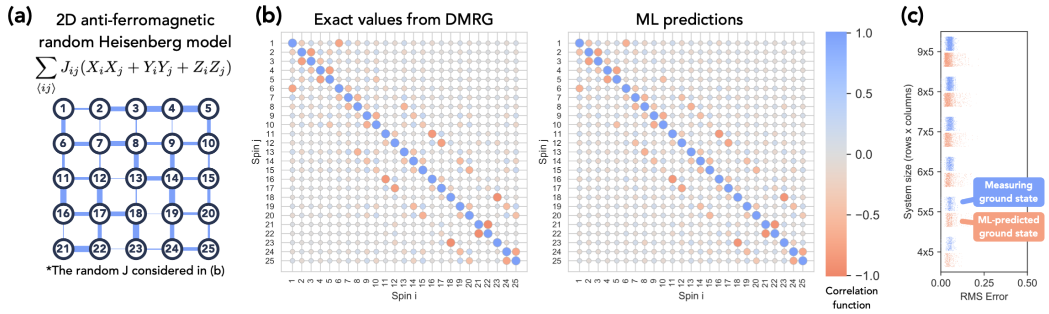


Figure 5.5: Numerical experiment for predicting ground state properties in the 2D antiferromagnetic Heisenberg model. (a) Hamiltonian. Illustration of the Heisenberg model geometry and Hamiltonian. We consider random couplings J_{ij}, sampled uniformly from $[0, 2]$. A particular instance is shown, with coupling strength indicated by the thickness of the edges connecting lattice points. (b) Two-point correlator. Exact values and ML predictions of the expectation value of the correlation function $C_{ij} = \frac{1}{3}(X_iX_j + Y_iY_j + Z_iZ_j)$ for all spin pairs (ij) in the lattice, for the Hamiltonian instance shown in (a). The absolute value of C_{ij} is represented by the size of each circle, while the circle's color indicates the actual value. (c) Prediction error. Each blue point indicates the root-mean-square error (averaged over Heisenberg model instances) of the correlation function for a particular pair (ij), where the estimate of C_{ij} is obtained using a classical shadow with $T = 500$ randomized Pauli measurements of the true ground state. Red points indicate errors in ML predictions for C_{ij}.

This example corroborates our expectation that classical machines can learn to efficiently predict ground state representations. An important caveat is that the rigorous guarantee in Theorem 17 applies only when the training points and the testing points are sampled from the same phase, while in this example the training data includes values of x from three different phases. Nevertheless, our numerics show that classical machines can still learn to predict well.

2D antiferromagnetic Heisenberg model — Our next example is the two-dimensional antiferromagnetic Heisenberg model. Spin-$\frac{1}{2}$ particles (i.e. qubits) occupy sites on a square lattice, and for each pair (ij) of neighboring sites the Hamiltonian contains a term $J_{ij}\left(X_iX_j + Y_iY_j + Z_iZ_j\right)$ where the couplings $\{J_{ij}\}$ are uniformly sampled from the unit interval $[0, 2]$. The parameter x is a list of all J_{ij} couplings; hence in this case the dimension of the parameter space is $m = O(n)$, where n is the number of qubits. The Hamiltonian $H(x)$ on a 5×5 lattice is shown in Figure 5.5(a).

We trained a classical ML model using 90 randomly chosen values of the parameter $x = \{J_{ij}\}$. For each such x, the exact ground state was found using DMRG, and we simulated $T = 500$ randomized Pauli measurements to construct a classical shadow. The classical ML predicted the classical representation at new values of x, and we used the predicted classical representation to estimate a two-body correlation function, the expectation value of $C_{ij} = \frac{1}{3}\left(X_i X_j + Y_i Y_j + Z_i Z_j\right)$, for each pair of qubits (ij). In Figure 5.5(b), the predicted and actual values of the correlation function are displayed for a particular value of x, showing reasonable agreement.

Figure 5.5(c) shows the prediction performance for all pairs of spins and for variable system size. Each red point in the plot represents the RMSE in the correlation function estimated using our predicted classical representation, for a particular pair of spins and averaged over sampled values of x. For comparison, each blue point is the RMSE when the correlation function is predicted using the classical shadow obtained by measuring the actual ground state $T = 500$ times. For most correlation functions, the prediction error achieved by the best classical ML model is comparable to the error achieved by measuring the actual ground state.

Classifying quantum phases of matter

For classifying quantum phases of matter, we consider an unsupervised classical ML model that constructs an infinite-dimensional *nonlinear* feature vector for each quantum state ρ by applying the map $\phi^{(\text{shadow})}$ in Eq. (5.159) with $\tau, \gamma = 1$ to the classical shadow $S_T(\rho)$ of the quantum state ρ. We then perform a principal component analysis (PCA) (Pearson, 1901) in the infinite-dimensional *nonlinear* feature space. The low-dimensional subspace found by PCA in the nonlinear feature space corresponds to a nonlinear low-dimensional manifold in the original quantum state space. This method is efficient using the shadow kernel $k^{(\text{shadow})}$ given in Eq. (5.161) and the kernel PCA procedure (Schölkopf, A. Smola, and Müller, 1998). Details are given in Appendix 5.13 and 5.13.

Bond-alternating XXZ model — We begin by considering the bond-alternating XXZ model with $n = 300$ spins. The Hamiltonian is given in Figure 5.6(a); it encompasses the bond-alternating Heisenberg model ($\delta = 1$) and the bosonic version of the Su-Schrieffer-Heeger model ($\delta = 0$) (Su, Schrieffer, and Heeger, 1979). The phase diagram in Figure 5.6(b) is obtained by evaluating the partial reflection many-body topological invariant (Pollmann and Ari M Turner, 2012; Andreas Elben, J. Yu, et al., 2020). There are three different phases: trivial, symmetry-protected topological, and symmetry broken.

For each value of J and δ considered, we construct the exact ground state using

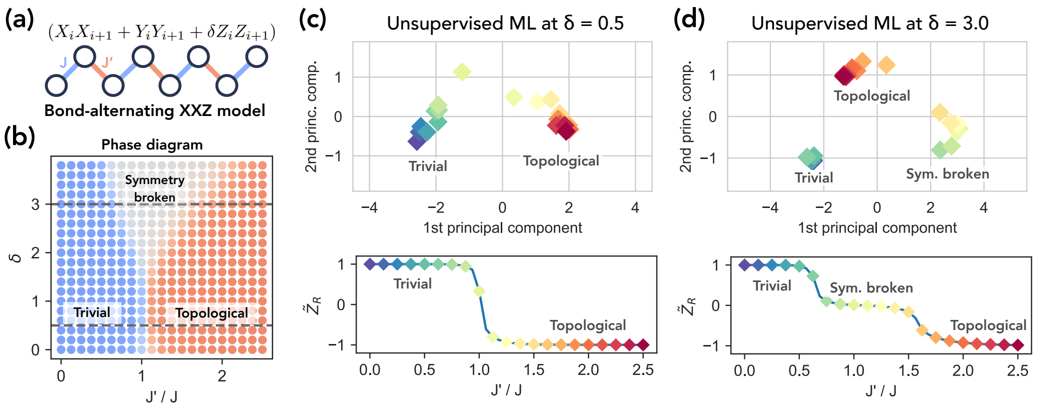


Figure 5.6: Numerical experiments for classifying quantum phases in the bond-alternating XXZ model. (a) Hamiltonian. Illustration of the model — a one-dimensional qubit chain, where the coefficient of $(X_iX_{i+1} + Y_iY_{i+1} + \delta Z_iZ_{i+1})$ alternates between J and J'. (b) Phase diagram. The system's three distinct phases are characterized by the many-body topological invariant $\tilde{Z}_R$ discussed in Refs. (Pollmann and Ari M Turner, 2012; Andreas Elben, J. Yu, et al., 2020). Blue denotes $\tilde{Z}_R = 1$, red denotes $\tilde{Z}_R = -1$, and gray denotes $\tilde{Z}_R \approx 0$. (c, d) Unsupervised phase classification. Bottom panels: $\tilde{Z}_R$ vs. J'/J at cross sections (c) $\delta = 0.5$ and (d) $\delta = 3.0$ of the phase diagram. Top panels: visualization of the quantum states projected to two dimensions using the unsupervised ML (nonlinear PCA with shadow kernel). In all panels, colors of the points indicate the value of J'/J, indicating that the two phases naturally cluster in the expressive feature space.

DMRG, and find its classical shadow by performing randomized Pauli measurement $T = 500$ times. We then consider a two-dimensional principal subspace of the infinite-dimensional nonlinear feature space found by the unsupervised ML based on the shadow kernel, which is visualized in Figure 5.6(c, d). We can clearly see that the different phases are well separated in the principal subspace. This shows that even without any phase labels on the training data, the ML model can already classify the phases accurately. Hence, when trained with only a small amount of labeled data, the ML model will be able to correctly classify the phases as guaranteed by Theorem 21.

Distinguishing a topological phase from a trivial phase — We consider the task of distinguishing the toric code topological phase from the trivial phase in a system of $n = 200$ qubits. Figure 5.7(a) illustrates the sampled topological and trivial states. We generate representatives of the nontrivial topological phase by applying low-depth geometrically local random quantum circuits to Kitaev's toric code state

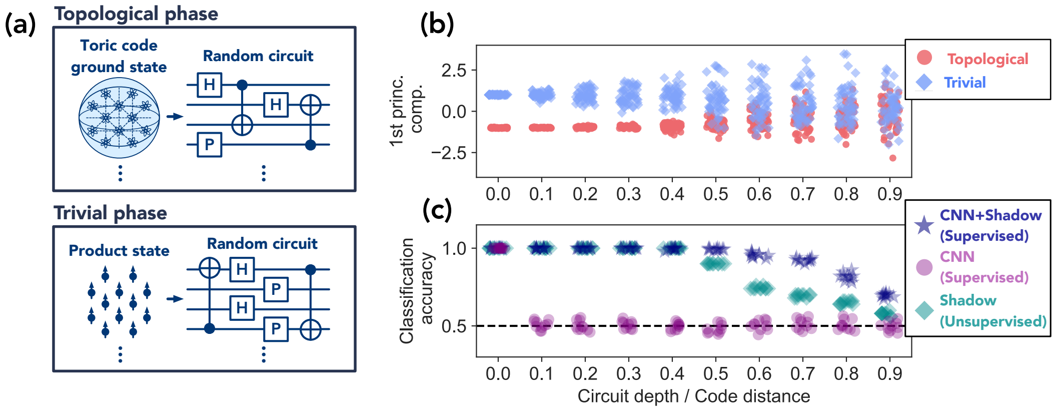


Figure 5.7: Numerical experiments for distinguishing between trivial and topological phases. (a) State generation. Trivial or topological states are generated by applying local random quantum circuits of some circuit depth to a product state or exactly-solved topological state, respectively. (b) Unsupervised phase classification. visualization of the quantum states projected to one dimension using the unsupervised ML (nonlinear PCA with shadow kernel), shown for varying circuit depth (divided by the "code distance" 10, which quantifies the depth at which the topological properties are washed out). The feature space is sufficiently expressive to resolve the phases for a small enough depth without training, with classification becoming more difficult as the depth increases. (c) Classification accuracy for three ML algorithms described in Section 5.12.

(A Yu Kitaev, 2003) with code distance 10, and we generate representatives of the trivial phase by applying random circuits to a product state.

Randomized Pauli measurements are performed $T = 500$ times to convert the states to their classical shadows, and these classical shadows are mapped to feature vectors in the high-dimensional feature space using the feature map $\phi^{(\text{shadow})}$. Figure 5.7(b) displays a one-dimensional projection of the feature space using the unsupervised classical ML for various values of the circuit depth, indicating that the phases become harder to distinguish as the circuit depth increases. In Figure 5.7(c), we show the classification accuracy of the unsupervised classical ML model. We also compare to training convolutional neural networks (CNN) that use measurement outcomes from the Pauli-6 POVM (Carrasquilla, Torlai, et al., 2019) as input to learn an observable for classifying the phases. Since Proposition 10 establishes that no observable (even a global one) can classify topological phases, this CNN approach is doomed to fail. On the other hand, if the CNN takes classical shadow representations as input, then it can learn nonlinear functions and successfully classify the phases.

5.13 Details regarding numerical experiments

In this appendix, we provide additional numerical experiments as well as more details about the numerical experiments described in the main text.

Additional numerical experiments

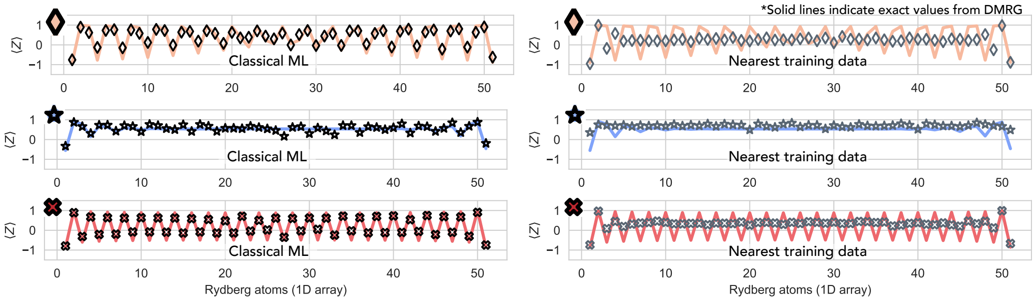


Figure 5.8: Numerical experiment for predicting ground state properties (Pauli-Z in each atom) in a 1D Rydberg atom system with 51 atoms. We use classical ML to predict the ground state properties at the three testing points. Also shown are "predictions" obtained from the training data nearest to the testing points. The markers denote predicted values, while the solid lines denote exact values obtained from DMRG.

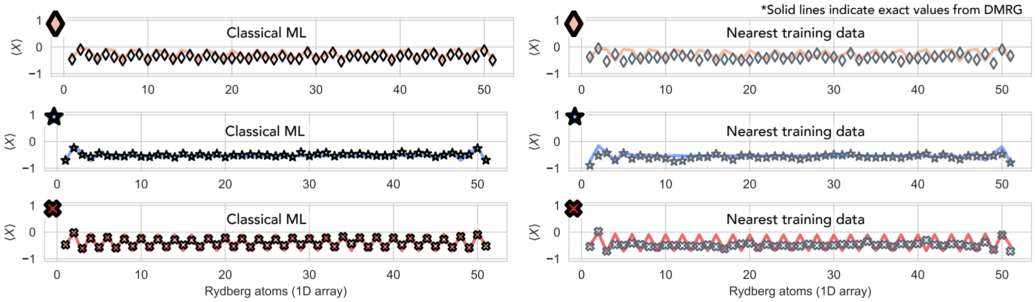


Figure 5.9: Numerical experiment for predicting ground state properties (Pauli-X in each atom) in a 1D Rydberg atom system with 51 atoms. We use classical ML to predict the ground state properties at the three testing points. Also shown are "predictions" obtained from the training data nearest to the testing points. The markers denote predicted values, while the solid lines denote exact values obtained from DMRG.

Rydberg atom chain — In the main text, we have provided partial prediction outcomes for a one-dimensional chain of $n = 51$ Rydberg atoms; see Figure 5.4. Here, we supply predictions of expectation values of Pauli operators Z_i and X_i on all 51

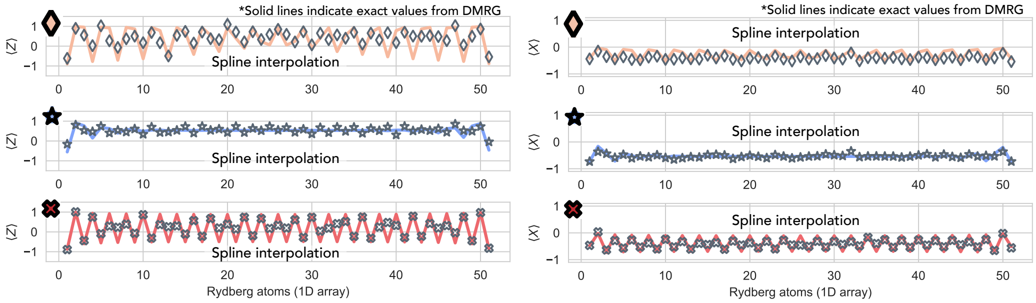


Figure 5.10: “Predictions” obtained by performing bivariate B-spline interpolation using the training data. The markers denote interpolated values, while the solid lines denote exact values obtained from DMRG.

atoms at the testing points marked in Figure 5.4(b). These are shown in Figure 5.8 and Figure 5.9, respectively. These extend the more restricted presentation in the main text to all qubits. In Figure 5.10, we show a different baseline considering bivariate B-spline interpolation from the training data.

Distinguishing an SPT phase from a trivial phase — We consider a one-dimensional chain of $n = 50$ qubits with $Z_2 \times Z_2$ symmetry. The 1D cluster state is in the nontrivial SPT phase. We generate other representatives of the nontrivial SPT phase by applying symmetric depth-3 geometrically local random quantum circuits to the cluster state, and we generate representatives of the trivial phase by applying symmetric depth-3 random circuits to a product state.

Randomized Pauli measurements are performed $T = 500$ times to convert the states to their classical shadows, and these classical shadows are mapped to feature vectors in the infinite-dimensional feature space using the feature map $\phi^{(\text{shadow})}$ (5.159). In Figure 5.11(a), inner products of feature vectors (matrix elements of the shadow kernel) are displayed. Figure 5.11(b) shows the feature vectors projected onto a two-dimensional subspace using *nonlinear* principal component analysis (PCA) based on the shadow kernel $k^{(\text{shadow})}$. Both figures show that feature vectors representing distinct phases can be distinguished easily. Correspondingly, the classical ML efficiently learns how to classify phases accurately, even if the training data is unlabeled.

Distinguishing a topologically-ordered phase from a trivial phase — We consider the task of distinguishing the toric code (A Yu Kitaev, 2003) topologically-ordered phase from the trivial phase in a system of $n = 200$ qubits. We generate other

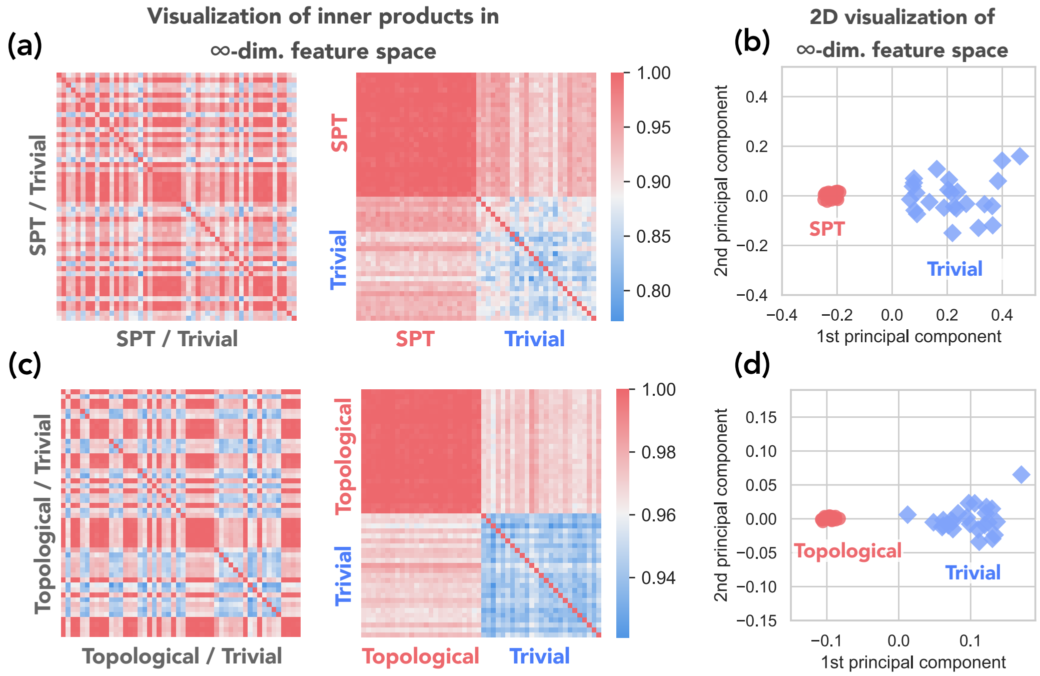


Figure 5.11: Numerical experiments for distinguishing trivial and topological phases. Trivial or topological states are generated by applying low-depth local random quantum circuits to a product state or exactly solved topological state respectively. (a) Kernel matrix for SPT/trivial phases The exactly solved topological state is the cluster state. The (i, j)-entry denotes the inner product of the i-th and j-th feature vectors in the infinite-dimensional feature space defined by the classical shadow representation. To the left, states from the two phases are randomly mixed. To the right, the two phases are ordered. (b) Kernel matrix for topologically-ordered/trivial phases. The exactly solved topological state is the toric code ground state.

representatives of the topologically-ordered phase by applying two-dimensional depth-3 geometrically local random quantum circuits to the toric code state, and we generate representatives of the trivial phase by applying two-dimensional depth-3 random circuits to a product state.

Randomized Pauli measurements are performed $T = 500$ times to convert the states to their classical shadows, and these classical shadows are mapped to feature vectors in the infinite-dimensional feature space using the feature map $\phi^{(\text{shadow})}$. In Figure 5.11(c, d), inner products of feature vectors (matrix elements of the shadow kernel) and the projection of feature space data onto the two-dimensional subspace spanned by the largest principal components is shown. Once more, one can clearly see that feature vectors representing distinct phases can be distinguished easily. Correspondingly, the classical ML efficiently learns how to classify phases accurately, even if the training data is unlabeled.

Ground state properties of the Rydberg atom chain

Our first example is a one-dimensional chain of $n = 51$ Rydberg atoms (Fendley, Sengupta, and Sachdev, 2004; Browaeys and Lahaye, 2020; Bernien et al., 2017). Each atom can be in either its ground state or a highly excited Rydberg state. Such systems can effectively be regarded as a qubit, where the basis state $|0\rangle$ is the ground state $|g\rangle$ and the basis state $|1\rangle$ is the Rydberg state $|r\rangle$. The Hamiltonian of the atomic chain is

$$H = \frac{\Omega}{2}\sum_i X_i - \Delta\sum_i N_i + \Omega\sum_{i<j}\left(\frac{R_b}{a|i-j|}\right)^6 N_i N_j \ , \tag{5.305}$$

where Ω is the (fixed) Rabi frequency, Δ is the laser detuning, N_i is the Rydberg occupation number operator, a is the separations of the atoms, and R_b is the so called Rydberg blockade radius. For large and negative Δ, the ground state of H is a vacuum state, where all atoms are in the ground state $|g\rangle$. In contrast, for large and positive Δ, different broken-symmetry ground states can be engineered depending on the value of R_b.

Approximations of the exact ground states of the Rydberg chain were found using the density-matrix renormalization group (DMRG) based on matrix product states (MPS). Starting from a random MPS with bond dimension $\chi = 10$, we variationally optimize the MPS using a singular value decomposition (SVD) cutoff of 10^{-9}. We perform a number of DMRG sweeps until the change in energy is below $\epsilon = 10^{-6}$. Upon convergence, we perform randomized Pauli measurements simply by performing local rotations into the corresponding Pauli bases, and sampling the resulting state (Ferris and Vidal, 2012).

In Figure 5.4(b), the color in the phase diagram corresponds to the phase obtained by two order parameters for characterizing Z_2 and Z_3 order. For Z_2 order, where the atoms are in $|rgrgrg\ldots\rangle$ or $|grgrgr\ldots\rangle$, we consider the order parameter,

$$O_{Z_2} = \frac{1}{n-1}\sum_{i=1}^{n-1}\left(|r_i g_{i+1}\rangle\langle r_i g_{i+1}| + |g_i r_{i+1}\rangle\langle g_i r_{i+1}|\right) . \tag{5.306}$$

For Z_3 order, where the atoms are in $|rggrgg\ldots\rangle$ or $|grggrg\ldots\rangle$ or $|ggrggr\ldots\rangle$, we consider the order parameter,

$$O_{Z_3} = \tag{5.307}$$

$$\tfrac{1}{n-2}\sum_{i=1}^{n-2} \left(|r_i g_{i+1} g_{i+2}\rangle\langle r_i g_{i+1} g_{i+2}| + |g_i r_{i+1} g_{i+2}\rangle\langle g_i r_{i+1} g_{i+2}| + |g_i g_{i+1} r_{i+2}\rangle\langle g_i g_{i+1} r_{i+2}| \right). \tag{5.308}$$

We estimate the two order parameters of the ground state ρ. First we check which order parameter (O_{Z_2} or O_{Z_3}) yields a larger expectation value. Then, we check if that expectation value is larger than the threshold value 0.8. If $O_{Z_2} > O_{Z_3}$ and $O_{Z_2} > 0.8$, we associate the state with the Z_2-order phase (red color). Else if $O_{Z_3} > O_{Z_2}$ and $O_{Z_3} > 0.8$, we say that the state is in the Z_3-order phase (vanilla color). If neither of these conditions is satisfied (both expectation values are less than 0.8), we assign the disordered phase (blue color) to this state.

For the Rydberg atom experiment, the input parameter vector x is two-dimensional. We first normalize the values to lie within a square $[-1, 1]^2$. Then we consider classical machine learning models given by

$$\hat{\sigma}_N(x) = \sum_{\ell=1}^{N} \kappa(x, x_\ell)\sigma_T(x_\ell) = \sum_{\ell=1}^{N} \underbrace{\left(\sum_{\ell'=1}^{N} k(x, x_{\ell'})(K + \lambda I)^{-1}_{\ell'\ell} \right)}_{\kappa(x,x_\ell)} \sigma_T(x_\ell), \tag{5.309}$$

where $\lambda > 0$ is a parameter to regularize the model when K is not invertible, $\sigma_T(x_\ell)$ is shorthand for $\sigma_T(\rho_\ell)$ and denotes the classical shadow representation of the ground state $\rho_\ell = \rho(x_\ell)$ under T randomized Pauli measurements. Moreover, $K_{ij} = k(x_i, x_j)$ is the kernel matrix, $k(x, x')$ is a kernel function, and $\kappa(x, x_\ell)$ is a function that depends on the kernel function, the kernel matrix K, and λ. We consider a set of different regularization parameters,

$$\lambda \in \{0.0125, 0.025, 0.05, 0.125, 0.25, 0.5, 1.0, 2.0, 4.0, 8.0\}, \tag{5.310}$$

and we also consider a set of different kernel functions

$$k(x, x') = \tilde{k}(x, x')/\sqrt{\tilde{k}(x, x)\tilde{k}(x', x')}, \tag{5.311}$$

where the kernels $\tilde{k}$ are

$$\tilde{k}(x, x') = \exp(-\gamma \, \|x - x'\|_2^2), \qquad \text{(Gaussian)}, \tag{5.312a}$$

$$\tilde{k}(x,x') = \sum_{k_1=-3}^{3}\sum_{k_2=-3}^{3} \cos\left(\pi(k_1(x_1 - x_1') + k_2(x_2 - x_2'))\right), \qquad \text{(Dirichlet)}, \tag{5.312b}$$

$$\tilde{k}(x,x') = k^{(\mathrm{NTK})}(x,x'), \qquad \text{(Neural tangent)}. \tag{5.312c}$$

The hyperparameter $\gamma > 0$ in the Gaussian kernel is chosen to be equal to

$$N^2 / \sum_{i=1}^{N}\sum_{j=1}^{N} \left\|x_i - x_j\right\|_2^2, \tag{5.313}$$

the inverse of the average distance between x_i and x_j. We consider the neural tangent kernel $k^{(\mathrm{NTK})}(x,x')$ (Jacot, Gabriel, and Hongler, 2018; Novak, L. Xiao, Hron, J. Lee, Alexander A. Alemi, et al., 2020) that is equivalent to an infinite-width feed-forward neural network with $2,3,4,5$ hidden layers and that uses the rectified linear unit (ReLU) as the activation function. Computing the neural tangent kernel can be implemented easily using the open-source software Neural Tangents (Novak, L. Xiao, Hron, J. Lee, Alexander A. Alemi, et al., 2020). Suppose that the input data $\{x_\ell\}_{\ell=1}^{N}$ is stored in a `numpy` array of size $N \times m$, denoted as `dataX` in the following code. We can use then use following code to generate the neural tangent kernel matrix. The imported package `neural_tangents` can be downloaded from `https://github.com/google/neural-tangents`.

```python
import jax
import numpy as np
from neural_tangents import stax

init_fn, apply_fn, kernel_fn = stax.serial(
    stax.Dense(32), stax.Relu(),
    stax.Dense(32), stax.Relu(),
    stax.Dense(1)
)
kernel_NN2 = kernel_fn(dataX, dataX, 'ntk')

init_fn, apply_fn, kernel_fn = stax.serial(
    stax.Dense(32), stax.Relu(),
    stax.Dense(32), stax.Relu(),
    stax.Dense(32), stax.Relu(),
    stax.Dense(1)
)
kernel_NN3 = kernel_fn(dataX, dataX, 'ntk')

init_fn, apply_fn, kernel_fn = stax.serial(
    stax.Dense(32), stax.Relu(),
    stax.Dense(32), stax.Relu(),
    stax.Dense(32), stax.Relu(),
    stax.Dense(32), stax.Relu(),
    stax.Dense(1)
)
kernel_NN4 = kernel_fn(dataX, dataX, 'ntk')

init_fn, apply_fn, kernel_fn = stax.serial(
    stax.Dense(32), stax.Relu(),
    stax.Dense(32), stax.Relu(),
    stax.Dense(32), stax.Relu(),
    stax.Dense(32), stax.Relu(),
    stax.Dense(32), stax.Relu(),
    stax.Dense(1)
)
kernel_NN5 = kernel_fn(dataX, dataX, 'ntk')

list_kernel_NN = [kernel_NN2, kernel_NN3, kernel_NN4, kernel_NN5]

# Normalization of the kernel matrix
for r in range(len(list_kernel_NN)):
    for i in range(len(list_kernel_NN[r])):
        for j in range(len(list_kernel_NN[r])):
            list_kernel_NN[r][i][j] /= (list_kernel_NN[r][i][i] \
                                        * list_kernel_NN[r][j][j]) ** 0.5
```

In order to predict the expectation value $\mathrm{tr}(O\hat{\sigma}_N(x))$ of an observable O for a new ground state $\hat{\sigma}_N(x)$, we utilize the following property of expectation values,

$$\mathrm{tr}(O\hat{\sigma}_N(x)) = \sum_{\ell=1}^{N} \kappa(x, x_\ell)\, \mathrm{tr}(O\sigma_T(x_\ell)). \tag{5.314}$$

Hence, we first compute $\mathrm{tr}(O\sigma_T(x_\ell))$, which can be done efficiently for r-body observables that factorize nicely into tensor products. Indeed, an $O = O_{i_1} \otimes \ldots \otimes O_{i_r}$ ensures

$$\mathrm{tr}(O\sigma_T(x_\ell)) = \frac{1}{T}\sum_{t=1}^{T} \mathrm{tr}\left(O\sigma_1^{(t)}(x_\ell) \otimes \cdots \otimes \sigma_n^{(t)}(x_\ell)\right) \tag{5.315}$$

$$= \frac{1}{T}\sum_{t=1}^{T} \mathrm{tr}\left(O_{i_1}\sigma_{i_1}^{(t)}(x_\ell)\right) \ldots \mathrm{tr}\left(O_{i_r}\sigma_{i_r}^{(t)}(x_\ell)\right), \tag{5.316}$$

and the right hand side can be computed with $O(Tn)$ arithmetic operations. Then, we can compute $\mathrm{tr}(O\hat{\sigma}_N(x))$ by extrapolating $\mathrm{tr}(O\sigma_T(x_\ell))$ using $\kappa(x, x_\ell)$. We utilize scikit-learn, a Python package (Pedregosa et al., 2011), for the training of these machine learning models.

Due to the different classical ML models one could consider (corresponding to different regularization parameters λ and kernel functions $k(x, x')$), we have to perform model selection to find an appropriate ML model. Typically, the prediction performance will be quite sensitive to these parameters, so one has to select them carefully. To evaluate the ML models, we consider 100 different points $x \in [-1, 1]^2$ in parameter space. Among these 100 points, we select $N = 20$ to be training data. These are the circled points in Figure 5.4(b). For each property we would like to predict, we choose one of the the three kernels and the different values of λ such that the prediction error is minimized on a validation set containing $80 - 3$ inputs of x. The validation set is disjoint from the 20 training points and the 3 testing points for evaluating the prediction performances (special markers in Figure 5.4(a)). Their purpose is to perform model selection. Finally, we test on the three input x's shown by the special markers (cross, diamond and star) in Figure 5.4(b).

We found that for each property we would like to predict, the prediction performance for different classical ML model varies moderately. When we have sufficiently large training data size N, most choices of λ and the kernel function should yield good prediction performance. However, we are using a very small number of training data in our experiments, hence the choice of these options becomes more important. In particular, the best choice of λ can differ quite significantly over the different properties we would like to predict.

For completeness, we include a set of experiments where we vary the training data size N or the classical shadow size T, where by "shadow size" we mean the number of randomized Pauli measurements used to approximate each state. The result in given

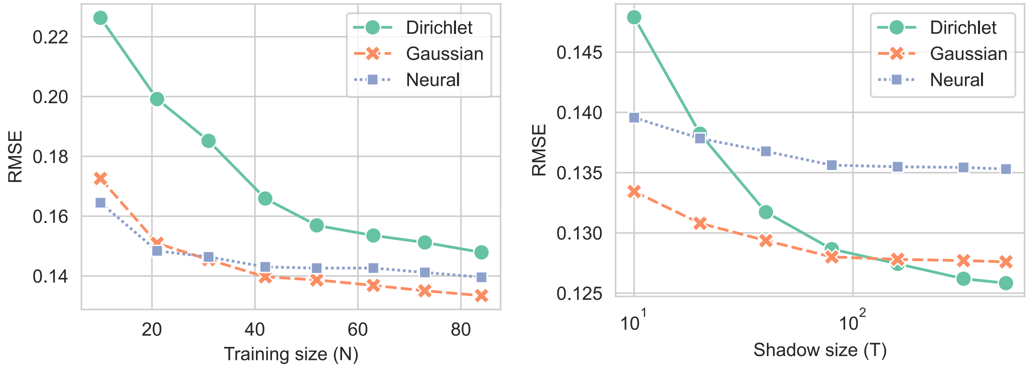


Figure 5.12: Numerical experiment for predicting ground state properties (Pauli-X and Z in each atom) in a 1D Rydberg atom system with 51 atoms under different hyperparameters. (Left) The prediction error (root-mean-square error) over different training sizes N with a fixed number $T = 10$ of randomized Pauli measurements, also referred to as the shadow size. (Right) The prediction error over different shadow sizes T with a fixed training data size $N = 31$.

in Figure 5.12. For this set of experiments, we consider a fixed set of 70 validation points in the phase space. Recall that we are using the ML model to predict ground state properties. Here, we consider the properties to be the expectation values of single-site Pauli-X and Pauli-Z operators. Because there are a total of 51 atoms, there are a total of $51 \times 2 = 102$ properties. For each property, we randomly draw 10 different points in the phase space (not in the training set or the validation set). Therefore, the test set is of size 1020, where each instance in the test set corresponds to a property of a point in the phase space. The prediction error is given by the root-mean-square error over the 1020 instances in the test set. We can see that as training set size N increases, the prediction becomes better. However, we see that as the training size increases, the slope of the prediction error (RMSE) over N flattens. This is expected from the theorem we established showing that $N = m^{O(1/\epsilon)}$, where N is the training set size, m is the number of parameters, and ϵ is the prediction error. While the theorem only provides an upper bound on N, if we assume the upper bound is saturated, then we can use elementary calculus to derive

$$\frac{d\epsilon}{dN} \text{ is proportional to } -\frac{\epsilon^2}{N\log(m)}. \tag{5.317}$$

Hence, the analysis is compatible with the observation that the slope of RMSE over N flattens as N becomes larger. While we proved a rigorous result using the Dirichlet kernel, other commonly used ML models may yield a better prediction performance in practice. Proving rigorous prediction guarantees and understanding the limitations and strengths for other more commonly used ML models are important future directions.

Ground state properties of the 2D antiferromagnetic Heisenberg model

Our next example is the two-dimensional antiferromagnetic Heisenberg model. Spin-$\frac{1}{2}$ particles (i.e. qubits) occupy sites on a square lattice, and for each pair (ij) of neighboring sites the Hamiltonian contains a term $J_{ij}\left(X_iX_j + Y_iY_j + Z_iZ_j\right)$ where the couplings $\{J_{ij}\}$ are uniformly sampled from the interval $[0,2]$. The parameter x is a list of all J_{ij} couplings; hence in this case the dimension of the parameter space is $m = O(n)$, where n is the number of qubits. The Hamiltonian $H(x)$ on a 5×5 lattice is shown in Figure 5.5(a). The exact ground state was found using DMRG. Analogously to the Rydberg atoms experiments, we fixed the SVD cutoff to 10^{-8} and stopped the DMRG runs when the difference in energy was below 10^{-4}.

The classical ML models we considered are the same as the Rydberg atom chain experiment. The only difference is that we slightly modify the Dirichlet kernel (5.312b) to

$$k(x,x') = \sum_{i \neq j} \sum_{k_i=-3}^{3} \sum_{k_j=-3}^{3} \cos\left(\pi(k_i(x_i - x'_i) + k_j(x_j - x'_j))\right), \quad \text{(Dirichlet kernel).} \tag{5.318}$$

We trained the classical ML model using a training set containing $N = 90$ randomly chosen values of the parameter $x = \{J_{ij}\}$. Then, for each property we would like to predict, we find the top-performing ML model setting (out of all λ parameters and kernel functions $k(x,x')$) on a validation set containing 100 parameters x distinct from the training set. Finally, we test on 10 newly sampled parameters x to estimate the prediction error. Figure 5.5(b) shows the prediction outcome from one of the input parameter x. Figure 5.5(c) shows the RMSE from all 10 input parameters.

Similar to the Rydberg atom experiment, the best-performing ML model setting differs across the properties we would like to predict. The three kernels perform similarly at larger training data size N and larger number of randomized Pauli measurements T. But neural networks and Gaussian kernel methods tend to perform better in most cases. The best choice of λ differs substantially across the different properties: there is no single choice of λ that performs uniformly better than the other choices.

To showcase these effects, we also include a set of experiments where we vary

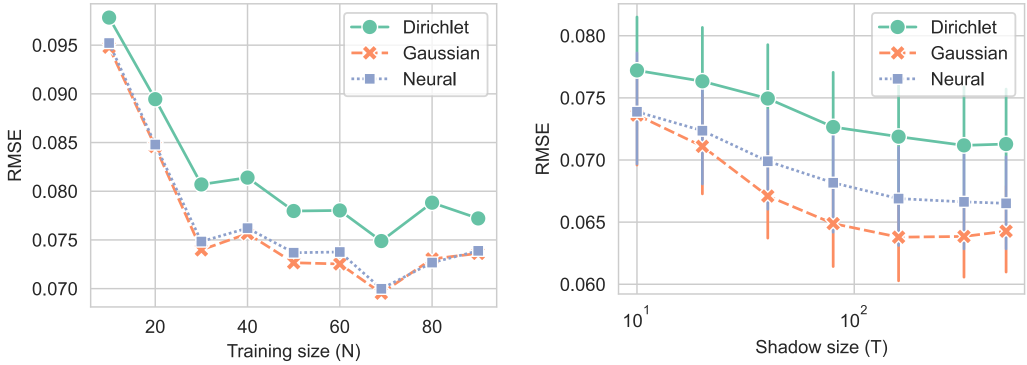


Figure 5.13: Numerical experiment for predicting ground state properties (two-point correlation functions) in a 2D antiferromagnetic Heisenberg model with 5×5 spins under different hyperparameters. (Left) The predict error (root-mean-square error) over different training size N with a fixed number of randomized Pauli measurements $T = 10$, also referred to as the shadow size. (Right) The prediction error (root-mean-square error) over different shadow size T with a fixed training data size $N = 90$.

the training data size N or the classical shadow size T, that is, the number of randomized Pauli measurements used to approximate each state. The numerical results are summarized in Figure 5.13. For this set of experiments, we consider fixed sets of 100 validation points. For this set of experiments, we consider a fixed set of 70 validation points in the phase space. Recall that we are using the ML model to predict ground state properties. Here, we consider the properties to be the two-point correlation functions over every pair of the 25 spins. This results in a total of $25 \times 25 = 625$ properties. For each property, we randomly draw 10 testing points in the $m = O(n)$ dimensional parameter space (not in the training set or the validation set). Therefore, the test set is of size 6250, where each instance in the test set corresponds to a property of a point in the parameter space. The prediction error is given by the root-mean-square error over the 6250 instances in the test set. The results resemble what was found in the Rydberg atom experiments, but with one notable difference — in the Rydberg experiments, but not for the 2D antiferromagnet, the Dirichlet kernel has the best performance for the largest shadow size T we considered. This may be because the dimension m of the parameter space is much lower in the Rydberg case.

Classifying phases of the bond-alternating XXZ model

To illustrate our classical ML for classifying quantum phases of matter, we consider the bond-alternating XXZ model with $n = 300$ spin-$\frac{1}{2}$ particles (i.e. qubits). The Hamiltonian is given by

$$\sum_{i:\text{odd}} J(X_i X_{i+1} + Y_i Y_{i+1} + \delta Z_i Z_{i+1}) + \sum_{i:\text{even}} J'(X_i X_{i+1} + Y_i Y_{i+1} + \delta Z_i Z_{i+1}), \quad (5.319)$$

and encompasses the bond-alternating Heisenberg model ($\delta = 1$), as well as the bosonic version of the Su-Schrieffer-Heeger model (Su, Schrieffer, and Heeger, 1979) ($\delta = 0$). The phase diagram in Figure 5.6(b) is obtained by evaluating the partial reflection many-body topological invariant (Pollmann and Ari M Turner, 2012; Andreas Elben, J. Yu, et al., 2020). It is given by

$$\tilde{\mathcal{Z}}_{\mathcal{R}} = \frac{\mathcal{Z}_{\mathcal{R}}}{\sqrt{[\mathrm{tr}(\rho_{I_1}^2) + \mathrm{tr}(\rho_{I_2}^2)]/2}}, \quad \text{where} \quad \mathcal{Z}_{\mathcal{R}} = \mathrm{tr}(\rho_{I_1 \cup I_2} \mathcal{R}_{I_1 \cup I_2}), \quad (5.320)$$

and we consider I_1 with 6 spins: the 145-th spin to the 150-th spin. Likewise, we fix I_2 to also contain 6 spins: the 151-th spin to the 156-th spin. Hence, the union $I_1 \cup I_2$ contains 12 spins. The symbols ρ_{I_1}, ρ_{I_2} and $\rho_{I_1 \cup I_2}$ denote the reduced density matrices associated with each local region. The reflection operator $\mathcal{R}_{I_1 \cup I_2}$ acts on the local region $I_1 \cup I_2$ and is given by

$$\mathcal{R}_{I_1 \cup I_2} \left| s_1, \ldots, s_{|I_1 \cup I_2|} \right\rangle = \left| s_{|I_1 \cup I_2|}, \ldots, s_1 \right\rangle, \quad \forall s_1, \ldots, s_{|I_1 \cup I_2|} \in \{0, 1\}. \quad (5.321)$$

The partial reflection many body-topological invariant can resolve three phases: trivial ($\tilde{\mathcal{Z}}_{\mathcal{R}} = +1$), symmetry-protected topological (SPT) ($\tilde{\mathcal{Z}}_{\mathcal{R}} = -1$) and symmetry broken ($\tilde{\mathcal{Z}}_{\mathcal{R}} = 0$). In Figure 5.6(b), we use the colors blue (trivial), red (SPT) and gray (symmetry broken) to visualize these different types of phases.

For each value of J'/J and δ considered, we construct the exact ground state using DMRG, and find its classical shadow by performing randomized single-qubit Pauli measurements a total of $T = 500$ times. To simulate this experiment, we follow the same setting for DMRG used in (Andreas Elben, J. Yu, et al., 2020). We limit the maximum number of sweeps to 100 and set the DMRG cutoff to 10^{-9}. We initialize the state to be the Néel state $|0101\ldots\rangle$. To pin one of the degenerate ground states in the symmetry-broken phase, we include a penalty term given by $0.1 J Z_1$ in the Hamiltonian.

After obtaining the classical shadow representation $S_T(\rho_\ell)$ for each quantum state ρ_ℓ, we compute the kernel matrix $K \in \mathbb{R}^{N \times N}$, where each entry is given by the shadow kernel $k^{(\text{shadow})}(S_T(\rho_\ell), S_T(\rho_{\ell'}))$. Recall that the shadow kernel is defined as

$$k^{(\text{shadow})}(S_T(\rho), S_T(\tilde{\rho})) = \exp\left(\frac{1}{T^2}\sum_{t,t'=1}^{T} \exp\left(\frac{1}{n}\sum_{i=1}^{n} \operatorname{tr}\left(\sigma_i^{(t)}\tilde{\sigma}_i^{(t')}\right)\right)\right), \tag{5.322}$$

$$\text{where } \sigma_i^{(t)} = 3|s_i^{(t)}\rangle\langle s_i^{(t)}| - \mathbb{I}, \tag{5.323}$$

and the classical shadow representation is given by

$$S_T(\rho) = \left\{|s_i^{(t)}\rangle : \ i \in \{1, \ldots, n\}, \ t \in \{1, \ldots, T\}\right\}, \tag{5.324}$$

$$\text{where} \quad |s_i^{(t)}\rangle \in \{|0\rangle, |1\rangle, |+\rangle, |-\rangle, |\mathrm{i}+\rangle, |\mathrm{i}-\rangle\}. \tag{5.325}$$

Care should be taken when computing diagonal elements of the kernel matrix K. The problem is that for $\rho = \tilde{\rho}$ and $t = t'$, we necessarily have $\operatorname{tr}\left(\sigma_i^{(t)}\tilde{\sigma}_i^{(t)}\right) = 5$ for all $1 \leq i \leq n$. And the double exponential will amplify this already substantial contribution enormously. We found that counteracting this blow-up improves the numerical stability of the kernel method substantially. When $\ell = \ell'$, when we compute $k^{(\text{shadow})}(S_T(\rho_\ell), S_T(\rho_\ell))$, we sum over $t \neq t'$ instead of all t, t'. In particular, when $\rho = \tilde{\rho}$, we consider a slight modification to the kernel definition,

$$k^{(\text{shadow})}(S_T(\rho), S_T(\rho)) = \exp\left(\frac{1}{T(T-1)}\sum_{t \neq t'} \exp\left(\frac{1}{n}\sum_{i=1}^{n} \operatorname{tr}\left(\sigma_i^{(t)}\sigma_i^{(t')}\right)\right)\right), \tag{5.326}$$

This modification also seems to slightly improve the classification performance.

After evaluating the kernel matrix K, we renormalize the entries to obtain the standardized kernel matrix

$$\overline{K}_{\ell\ell'} = \frac{K_{\ell\ell'}}{\sqrt{K_{\ell\ell}K_{\ell'\ell'}}} \quad \text{for} \quad \ell, \ell' \in \{1, \ldots, N\}. \tag{5.327}$$

Subsequently, we perform kernel principal component analysis (PCA) on $\overline{K}$. The implementation we used for kernel PCA is based on scikit-learn (Buitinck et al., 2013). The output of kernel PCA is a list of low-dimensional vectors (the dimension can be chosen arbitrarily, but we choose two dimensions for this experiment). Each low-dimensional vector corresponds to a quantum state. In Figure 5.6(c, d), we can see that the low-dimensional vectors are clustered into different quantum phases of matter.

Distinguishing an SPT phase from a trivial phase — We consider a one-dimensional chain of $n = 50$ qubits with $Z_2 \times Z_2$ symmetry. The 1D cluster state is in the nontrivial SPT phase. We generate other representatives of the nontrivial SPT phase by applying symmetric low-depth geometrically local random quantum circuits to the cluster state, and we generate representatives of the trivial phase by applying symmetric random circuits to a product state. We simulate the application of symmetric low-depth geometrically local random quantum circuits to the cluster state through matrix product states (MPS). Each circuit layer consists of patterns of random two-qubit gates acting on next-to-nearest neighbors sites. We generate the random gates in a block-sparse structure in the parity symmetry sectors. This choice, together with the choice of connectivity, guarantees that the $Z_2 \times Z_2$ symmetry is conserved during the circuit evolution.

Randomized Pauli measurements are performed $T = 500$ times to convert the states to their classical shadows. We perform kernel PCA to find low-dimensional representation for the quantum states using exactly the same method as the experiment on bond-alternating XXZ model.

Distinguishing a topological phase from a trivial phase

We consider the task of distinguishing the toric code topological phase from the trivial phase in a system of $n = 200$ qubits. Kitaev's toric code state (A Yu Kitaev, 2003) is in the nontrivial topologically-ordered phase, while a product state represents the trivial phase. To populate both phases, we apply low-depth geometrically local random Clifford circuits (Aaronson and Gottesman, 2004) to Kitaev's toric code state (A Yu Kitaev, 2003) with code distance 10, and we generate representatives of the trivial phase by applying random Clifford circuits to a product state. We utilize Clifford circuits to ensure efficient simulation of in total $n = 200$ qubits (and with a depth up to 9) by means of the Gottesman-Knill theorem. We again perform kernel PCA to find low-dimensional representations for the quantum states using exactly the same method as the experiment on bond-alternating XXZ model. This is used to generate the plot in Figure 5.7(b) for a one-dimensional projection of the feature space, as well as the plot in Figure 5.11(d) for a two-dimensional projection.

For the unsupervised ML model shown in Figure 5.7(c), we consider a combination of kernel PCA and randomized projections (Karnin et al., 2012). First we perform kernel PCA to map the data to a six-dimensional subspace of the infinite-dimensional feature space. Then we repeat the following procedure 500 times. We select a one-dimensional subspace uniformly at random in the six-dimensional subspace. We project all the quantum states to the one-dimensional subspace. Then, we find the center point (according to median instead of mean) to split up the quantum states into two phases. We also record the sum of the absolute values from all points to the center point in the one-dimensional subspace. Finally, we consider the classification obtained from the random one-dimensional projection that results in the largest sum of the absolute values.

For the convolutional neural network (CNN) approach shown in Figure 5.7(c), we consider the following CNN built from Keras (Chollet et al., 2015).

```
import tensorflow as tf
from tensorflow.keras import datasets, layers, models

model = models.Sequential()
model.add(layers.Conv2D(32, (2, 2), activation='relu', padding='same',
            input_shape=(2*L, L, 6)))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(32, (2, 2), activation='relu', padding='same'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(32, (2, 2), activation='relu', padding='same'))
model.add(layers.Flatten())
model.add(layers.Dense(32, activation='relu'))
model.add(layers.Dense(2))
```

In the above code, L is the code distance for the toric code and is equal to 10 in this experiment (recall that toric code ground state has $n = 2L^2$ qubits). This CNN model is supervised and requires a training data with a corresponding label for indicating which phase the training data point is in. We first perform the Pauli-6 POVM on each qubit (Carrasquilla, Torlai, et al., 2019) to transform the quantum state into a array of size n where each entry has six outcomes. We perform one-hot encoding to yield a classical vector of size $6n$, where each entry in the classical vector is either 0 or 1. Because the toric code ground state is two-dimensional ($2L \times L$), we restructure the classical vector into a three-dimensional tensor of size $2L \times L \times 6$. The first two dimensions corresponds to the spatial dimension of the toric code ground state. The last dimension corresponds to the one-hot encoded vector for the six-outcome POVM. We then train the above model using the Adam optimizer (Kingma and Ba, 2014) with the categorical cross entropy as the loss function. The code is given below.

```
model.compile(optimizer='adam',
    loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
    metrics=['accuracy'])
```

We train the convolutional neural network using 100 training points (half are topologically-ordered states, and the other half are trivial states). Then we use a validation set of 100 points to perform early stopping. This is because the longer we train, the more likely the neural network is going to overfit. Hence, it is a good practice to perform model selection by choosing which model to use at different time points (during the training process). We choose the model that performs the best on the validation set. Then we test the classification accuracy (the percentage that the prediction of the phases is correct) on a testing set consisting of 100 points.

The performance of the above ML model is not substantially different from random guessing. Hence, we also consider a very simple CNN enhanced with classical shadow under $T = 500$ randomized Pauli measurements. In particular, we compute the local reduced density matrix using the classical shadow. Then for each qubit, we represent it with the local reduced density matrix. For simplicity, we consider the i-th qubit to be represented by a vector of size 16, which includes the 2-body reduced density matrix for the subsystem consisting of the i-th and the $i+1$-th qubit. Hence, each quantum state is now represented by a classical vector of dimension $2L^2 \times 16$. We reshape the classical vector into a three-dimensional tensor of size $2L \times L \times 16$. The classical vector is feed into the convolutional neural network structured as follows. We also apply the Adam optimizer (Kingma and Ba, 2014) with the categorical cross entropy as the loss function. The evaluation process is exactly the same as the CNN approach based on the Pauli-6 POVM.

```
import tensorflow as tf
from tensorflow.keras import datasets, layers, models

model = models.Sequential()
model.add(layers.Conv2D(16, (1, 1), activation='relu',\
            padding='same', input_shape=(2*L, L, 16)))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(16, (2, 2), activation='relu',\
            padding='same'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(16, (2, 2), activation=’relu’,\
            padding=’same’))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Flatten())
model.add(layers.Dense(32, activation=’relu’))
model.add(layers.Dense(2))

model.compile(optimizer=’adam’,
    loss=tf.keras.losses.SparseCategoricalCrossentropy(
            from_logits=True),
    metrics=[’accuracy’])
```

Chapter 6

LEARNING TO PREDICT QUANTUM DYNAMICS

Learning complex quantum dynamics is a fundamental problem at the intersection of machine learning (ML) and quantum physics. Given an unknown n-qubit completely positive trace preserving (CPTP) map $\mathcal{E}$ that represents a physical process happening in nature or in a laboratory, we consider the task of learning to predict functions of the form

$$f(\rho, O) = \mathrm{tr}(O\mathcal{E}(\rho)), \tag{6.1}$$

where ρ is an n-qubit state and O is an n-qubit observable. Related problems arise in many fields of research, including quantum machine learning (Biamonte et al., 2017; Schuld and Killoran, 2019; Havlicek et al., 2019; Caro, Huang, Marco Cerezo, et al., 2022; Schreiber, Jens Eisert, and Meyer, 2022; Jarrod R McClean, Boixo, et al., 2018; Caro, Huang, Ezzell, et al., 2023; Huang, Broughton, J. Cotler, et al., 2022; Farhi and Neven, 2018; Arunachalam and Wolf, 2017), variational quantum algorithms (Gibbs et al., 2022; Cirstoiu et al., 2020; Peruzzo et al., 2014; Kandala et al., 2017; Kokail et al., 2019; Marco Cerezo et al., 2021; Grimsley et al., 2019), machine learning for quantum physics (Carleo and Troyer, 2017a; Sharir et al., 2020; Van Nieuwenburg, Y.-H. Liu, and Sebastian D Huber, 2017; Z. Zhou, Xiaocheng Li, and Zare, 2017; Carrasquilla and Roger G Melko, 2017a; Parr, 1980; Car and Parrinello, 1985; Becke, 1993; Steven R White, 1993a; Gilmer et al., 2017; Huang, Richard Kueng, Torlai, et al., 2022; Huang, Broughton, Masoud Mohseni, Babbush, Boixo, Neven, and Jarrod R McClean, 2021a), and quantum benchmarking (Masoud Mohseni, Ali T Rezakhani, and Daniel A Lidar, 2008; Scott, 2008; J. L. O'Brien et al., 2004; Levy, Luo, and Clark, 2021; Huang, Steven T Flammia, and Preskill, 2022; Merkel et al., 2013; Blume-Kohout et al., 2017). As an example, for predicting outcomes of quantum experiments (Huang, Richard Kueng, and Preskill, 2021; Melnikov et al., 2018; Huang, Broughton, J. Cotler, et al., 2022), we consider ρ to be parameterized by a classical input x, $\mathcal{E}$ is an unknown process happening in the lab, and O is an observable measured at the end of the experiment. Another example is when we want to use a quantum ML algorithm to learn a model of a complex quantum evolution with the hope that the learned model can be faster (Cirstoiu et al., 2020; Gibbs et al., 2022; Caro, Huang, Ezzell, et al., 2023).

As an n-qubit CPTP map $\mathcal{E}$ consists of exponentially many parameters, prior works, including those based on covering number bounds (Caro, Huang, Marco Cerezo, et al., 2022; Caro, Huang, Ezzell, et al., 2023; Huang, Richard Kueng, and Preskill, 2021; Huang, Broughton, J. Cotler, et al., 2022), classical shadow tomography (Levy, Luo, and Clark, 2021; Kunjummen et al., 2021), or quantum process tomography (Masoud Mohseni, Ali T Rezakhani, and Daniel A Lidar, 2008; Scott, 2008; J. L. O'Brien et al., 2004), require an exponential number of data samples to guarantee a small constant error for predicting outcomes of an arbitrary evolution $\mathcal{E}$ under a general input state ρ. To improve upon this, recent works (K.-M. Chung and H.-H. Lin, 2018; Caro, Huang, Marco Cerezo, et al., 2022; Caro, Huang, Ezzell, et al., 2023; Huang, Richard Kueng, and Preskill, 2021; Huang, Broughton, J. Cotler, et al., 2022) have considered quantum processes $\mathcal{E}$ that can be generated in polynomial-time and shown that a polynomial amount of data samples suffices to learn $\mathrm{tr}(O\mathcal{E}(\rho))$ in this restricted class. However, these results still require exponential computation time.

In this chapter, we present a computationally-efficient ML algorithm that can learn a model of an arbitrary unknown n-qubit process $\mathcal{E}$, such that given ρ sampled from a wide range of distributions over arbitrary n-qubit states and any O in a large physically-relevant class of observables, the ML algorithm can accurately predict $f(\rho, O) = \mathrm{tr}(O\mathcal{E}(\rho))$. The ML model can predict outcomes for highly entangled states ρ after learning from a training set that only contains data for random product input states and randomized Pauli measurements on the corresponding output states. The training and prediction of the proposed ML model are both efficient even if the unknown process $\mathcal{E}$ is a Hamiltonian evolution over an exponentially long time, a quantum circuit with exponentially many gates, or a quantum process arising from contact with an infinitely large environment for an arbitrarily long time. Furthermore, given few-body reduced density matrices (RDMs) of the input state ρ, the ML algorithm uses only classical computation to predict output properties $\mathrm{tr}(O\mathcal{E}(\rho))$.

The proposed ML model is a combination of efficient ML algorithms for two learning problems: (1) predicting $\mathrm{tr}(O\rho)$ given a known observable O and an unknown state ρ, and (2) predicting $\mathrm{tr}(O\rho)$ given an unknown observable O and a known state ρ. We give sample- and computationally-efficient learning algorithms for both problems. Then we show how to combine the two learning algorithms to address the problem of learning to predict $\mathrm{tr}(O\mathcal{E}(\rho))$ for an arbitrary unknown n-qubit quantum process $\mathcal{E}$. Together, the sample and computational efficiency of the two

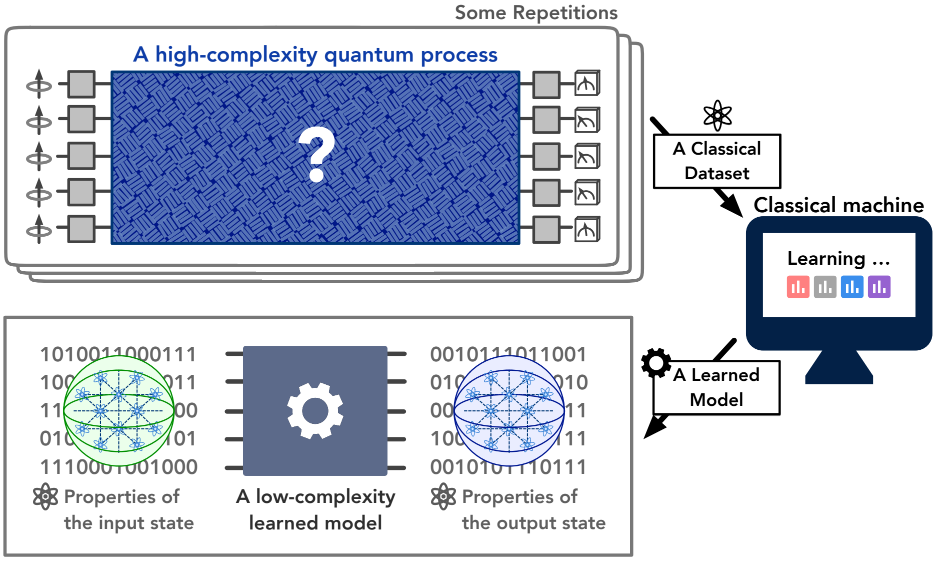


Figure 6.1: *Learning to predict an arbitrary unknown quantum process $\mathcal{E}$.* Given an unknown quantum process $\mathcal{E}$ with arbitrarily high complexity, and a classical dataset obtained from evolving random product states under $\mathcal{E}$ and performing randomized Pauli measurements on the output states. We give an algorithm that can learn a low-complexity model for predicting the local properties of the output states given the local properties of the input states.

learning algorithms implies the efficiency of the combined ML algorithm.

In order to establish the rigorous guarantee for the proposed ML algorithms, we consider a different task: optimizing a k-local Hamiltonian $H = \sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \alpha_P P$. We present an improved approximate optimization algorithm that finds either a maximizing/minimizing state $|\psi\rangle$ with a rigorous lower/upper bound guarantee on the energy $\langle\psi| H |\psi\rangle$ in terms of the Pauli coefficients α_P of H. The rigorous bounds improve upon existing results on optimizing k-local Hamiltonians (Dinur et al., 2006; Barak et al., 2015; Aram W Harrow and Montanaro, 2017a; Anshu, Gosset, et al., 2021). We then use the improved optimization algorithm to give a constructive proof of several useful norm inequalities relating the spectral norm $||O||$ of an observable O and the ℓ_p-norm of the Pauli coefficients α_P associated to the observable O. The proof resolves a recent conjecture in (Rouzé, Wirth, and Haonan Zhang, 2022) about the existence of quantum Bohnenblust-Hille inequalities. These norm inequalities are then used to establish the efficiency of the proposed ML algorithms.

6.1 Learning quantum states, observables, and processes

Before proceeding to state our main results in greater detail, we describe informally the learning tasks discussed in this paper: what do we mean by learning a quantum state, observable, and process?

Learning an unknown state

It is possible in principle to provide a complete classical description of an n-qubit quantum state ρ. However this would require an exponential number of experiments, which is not at all practical. Therefore, we set a more modest goal: to learn enough about ρ to predict many of its physically relevant properties. We specify a family of target observables $\{O_i\}$ and a small target accuracy ϵ. The learning procedure is judged to be successful if we can predict the expectation value $\mathrm{tr}(O_i \rho)$ of every observable in the family with error no larger than ϵ.

Suppose that ρ is an arbitrary and unknown n-qubit quantum state, and suppose that we have access to N identical copies of ρ. We acquire information about ρ by measuring these copies. In principle, we could consider performing collective measurements across many copies at once. Or we might perform single-copy measurements sequentially and *adaptively*; that is, the choice of measurement performed on copy j could depend on the outcomes obtained in measurements on copies $1, 2, 3, \ldots j-1$. The target observables we consider are *bounded-degree observables*. A bounded-degree n-qubit observable O is a sum of local observables (each with support on a constant number of qubits independent of n) such that only a constant number (independent of n) of terms in the sum act on each qubit. Most thermodynamic quantities that arise in quantum many-body physics can be written as a bounded-degree observable O, such as a geometrically-local Hamiltonian or the average magnetization.

In the learning protocols discussed in this paper, the measurements are neither collective nor adaptive. Instead, we fix an ensemble of possible single-copy measurements, and for each copy of ρ we independently sample from this ensemble and perform the selected measurement on that copy. Thus there are two sources of randomness in the protocol — the randomly chosen measurement on each copy, and the intrinsic randomness of the quantum measurement outcomes. If we are unlucky, the chosen measurements and/or the measurement outcomes might not be sufficiently informative to allow accurate predictions. We will settle for a protocol that achieves the desired prediction task with a high success probability.

For the protocol to be practical, it is highly advantageous for the sampled measurements to be easy to perform in the laboratory and easy to describe in classical language. The measurements we consider, *random Pauli measurements*, meet both of these criteria. For each copy of ρ and for each of the n qubits, we choose uniformly at random to measure one of the three single-qubit Pauli observables X, Y, or Z. This learning method, called *classical shadow tomography*, was analyzed in (Huang, Richard Kueng, and Preskill, 2020), where an upper bound on the sample complexity (the number N of copies of ρ needed to achieve the task) was expressed in terms of a quantity called the *shadow norm* of the target observables.

In this chapter, using a new norm inequality derived here, we improve on the result in (Huang, Richard Kueng, and Preskill, 2020) by obtaining a tighter upper bound on the shadow norm for bounded degree observables. The upshot is that, for a fixed target accuracy ϵ, we can predict all bounded-degree observables with spectral norm less than B by performing random Pauli measurements on

$$N = \mathcal{O}\left(\log(n)B^2/\epsilon^2\right) \tag{6.2}$$

copies of ρ. This result improves upon the previously known bound of

$$\mathcal{O}(n\log(n)B^2/\epsilon^2). \tag{6.3}$$

Furthermore, we derive a matching lower bound on the number of copies required for this task, which applies even if collective measurements across many copies are allowed.

Learning an unknown observable

Now suppose that O is an arbitrary and unknown n-qubit observable. We also consider a distribution $\mathcal{D}$ on n-qubit quantum states. This distribution, too, need not be known, and it may include highly entangled states. Our goal is to find a function $h(\rho)$ which predicts the expectation value $\mathrm{tr}(O\rho)$ of the observable O on the state ρ with a small mean squared error:

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} |h(\rho) - \mathrm{tr}(O\rho)|^2 \leq \epsilon.$$

To define this learning task, it is convenient to assume that we can access training data of the form

$$\{\rho_\ell, \mathrm{tr}\left(O\rho_\ell\right)\}_{\ell=1}^{N}, \tag{6.4}$$

where ρ_ℓ is sampled from the distribution $\mathcal{D}$. In practice, though, we cannot directly access the exact value of the expectation value $\mathrm{tr}\left(O\rho_\ell\right)$; instead, we might measure

O multiple times in the state ρ_ℓ to obtain an accurate estimate of the expectation value. Furthermore, we don't necessarily need to sample states from $\mathcal{D}$ to achieve the task. We might prefer to learn about O by accessing its expectation value in states drawn from a different ensemble.

A crucial idea is that we can learn O efficiently if the distribution $\mathcal{D}$ has suitably nice features. Specifically, we consider distributions that are invariant under single-qubit Clifford gates applied to any one of the n qubits. We say that such distributions are *locally flat*, meaning that the probability weight assigned to an n-qubit state is unmodified (i.e., the distribution appears flat) when we locally rotate any one of the qubits.

An arbitrary observable O can be expanded in terms of the Pauli operator basis:

$$O = \sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \alpha_P P. \tag{6.5}$$

Though there are 4^n Pauli operators, if the distribution $\mathcal{D}$ is locally flat and O has a constant spectral norm, we can approximate the sum over P by a truncated sum

$$O^{(k)} = \sum_{P \in \{I,X,Y,Z\}^{\otimes n} : |P| \leq k} \alpha_P P. \tag{6.6}$$

including only the Pauli operators P with weight $|P|$ up to k, those acting nontrivially on no more than k qubits. The mean squared error incurred by this truncation decays exponentially with k. Therefore, to learn O with mean squared error ϵ it suffices to learn this truncated approximation to O, where $k = \mathcal{O}(\log(1/\epsilon))$. Furthermore, using norm inequalities derived in this paper, we show that for the purpose of predicting the expectation value of this truncated operator it suffices to learn only a few relatively large coefficients α_P, while setting the rest to zero. The upshot is that, for a fixed target error ϵ, an observable with constant spectral norm can be learned from training data with size $\mathcal{O}(\log n)$, where the classical computational cost of training and predicting is $n^{\mathcal{O}(k)}$.

Usually, in machine learning, after learning from a training set sampled from a distribution $\mathcal{D}$, we can only predict new instances sampled from the same distribution $\mathcal{D}$. We find, though, that for the purpose of learning an unknown observable, there is a particular locally flat distribution $\mathcal{D}'$ such that learning to predict under $\mathcal{D}'$ suffices for predicting under any other locally flat distribution. Namely, we samples from the n-qubit state distribution $\mathcal{D}'$ by preparing each one of the n qubits in one of the six Pauli operator eigenstates $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle, |y+\rangle, |y-\rangle\}$, chosen uniformly at random. Pleasingly, preparing samples from $\mathcal{D}'$ is not only sufficient for our task, but also easy to do with existing quantum devices.

After training is completed, to predict $\mathrm{tr}(O\rho)$ for a new state ρ drawn from the distribution $\mathcal{D}$, we need to know some information about ρ. The state ρ, like the operator O, can be expanded in terms of Pauli operators, and when we replace O by its weight-k truncation, only the truncated part of ρ contributes to its expectation value. Thus if the k-body reduced density matrices (RDMs) for states drawn from $\mathcal{D}$ are known classically, then the predictions can be computed classically. If the states drawn from $\mathcal{D}$ are presented as unknown quantum states, then we can learn these k-body RDMs efficiently (for small k) using classical shadow tomography, and then proceed with the classical computation to obtain a predicted value of $\mathrm{tr}(O\rho)$.

Learning an unknown process

Now suppose that $\mathcal{E}$ is an arbitrary and unknown quantum process mapping n qubits to n qubits. Let $\{O_i\}$ be a family of target observables, and $\mathcal{D}$ be a distribution on quantum states. We assume the ability to repeatedly access $\mathcal{E}$ for a total of N times. Each time we can apply $\mathcal{E}$ to an input state of our choice, and perform the measurement of our choice on the resulting output. In principle we could allow input states that are entangled across the N channel uses, and allow collective measurements across the N channel outputs. But here we confine our attention to the case where the N inputs are unentangled, and the channel outputs are measured individually. Our goal is to find a function $h(\rho, O)$ which predicts, with a small mean squared error, the expectation value of O_i in the output state $\mathcal{E}(\rho)$ for every observable O_i in the family $\{O_i\}$:

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} |h(\rho, O_i) - \mathrm{tr}(O_i\mathcal{E}(\rho))|^2 \leq \epsilon. \tag{6.7}$$

Our main result is that this task can be achieved efficiently if O_i is a bounded-degree observable and $\mathcal{D}$ is locally flat. That is, N, the number of times we access $\mathcal{E}$, and the computational complexity of training and prediction, scale reasonably with the system size n and the target accuracy ϵ.

To prove this result, we observe that the task of learning an unknown quantum process can be reduced to learning unknown states and learning unknown observables. If ρ_ℓ is sampled from the distribution $\mathcal{D}$, then, since $\mathcal{E}$ is unknown, $\mathcal{E}(\rho_\ell)$ should be regarded as an unknown quantum state. Suppose we learn this state; that is, after preparing and measuring $\mathcal{E}(\rho_\ell)$ sufficiently many times we can accurately predict the expectation value $\mathrm{tr}(O_i\mathcal{E}(\rho_\ell))$ for each target observable O_i.

Now notice that $\mathrm{tr}(O_i\mathcal{E}(\rho_\ell)) = \mathrm{tr}(\mathcal{E}^\dagger(O_i)\rho_\ell)$, where $\mathcal{E}^\dagger$ is the (Heisenberg-picture) map dual to $\mathcal{E}$. Since $\mathcal{E}^\dagger$ is unknown, $\mathcal{E}^\dagger(O_i)$ should be regarded as an unknown observable. Suppose we learn this observable; that is, using the dataset $\{\rho_\ell, \mathrm{tr}(\mathcal{E}^\dagger(O_i)\rho_\ell)\}$ as training data, we can predict $\mathrm{tr}(\mathcal{E}^\dagger(O_i)\rho)$ for ρ drawn from $\mathcal{D}$ with a small mean squared error. This achieves the task of learning the process $\mathcal{E}$ for state distribution $\mathcal{D}$ and target observable O_i.

Having already shown that arbitrary quantum states can be learned efficiently for the purpose of predicting expectation values of bounded-degree observables, and that arbitrary observables can be learned efficiently for locally flat input state distributions, we obtain our main result. Since the distribution $\mathcal{D}$ is locally flat, it suffices to learn the low-degree truncated approximation to the unknown operator $\mathcal{E}^\dagger(O_i)$, incurring only a small mean squared error. To predict $\mathrm{tr}(\mathcal{E}^\dagger(O_i)\rho)$, then, it suffices to know only the few-body RDMs of the input state ρ. For any input state ρ, these few-body density matrices can be learned efficiently using classical shadow tomography.

As noted above in the discussion of learning observables, the states ρ_ℓ in the training data need not be sampled from $\mathcal{D}$. To learn a low-degree approximation to $\mathcal{E}^\dagger(O_i)$, it suffices to sample from a locally flat distribution on product states. Even if we sample only product states during training, we can make accurate predictions for highly entangled input states. We also emphasize again that the unknown process $\mathcal{E}$ is arbitrary. Even if $\mathcal{E}$ has quantum computational complexity exponential in n, we can learn to predict $\mathrm{tr}(O\mathcal{E}(\rho))$ accurately and efficiently, for bounded-degree observables O and for any locally flat distribution on the input state ρ.

6.2 Algorithm for learning an unknown quantum process

Consider an unknown n-qubit quantum process $\mathcal{E}$ (a CPTP map). Suppose we have obtained a classical dataset by performing N randomized experiments on $\mathcal{E}$. Each experiment prepares a random product state $|\psi^{(\mathrm{in})}\rangle = \bigotimes_{i=1}^n |s_i^{(\mathrm{in})}\rangle$, passes through $\mathcal{E}$, and performs a randomized Pauli measurement (Huang, Richard Kueng, and Preskill, 2020; Andreas Elben, Steven T Flammia, et al., 2022) on the output state. Recall that a randomized Pauli measurement measures each qubit of a state in a random Pauli basis (X, Y or Z) and produces a measurement outcome of $|\psi^{(\mathrm{out})}\rangle = \bigotimes_{i=1}^n |s_i^{(\mathrm{out})}\rangle$, where $|s_i^{(\mathrm{out})}\rangle \in \mathrm{stab}_1 \triangleq \{|0\rangle, |1\rangle, |+\rangle, |-\rangle, |y+\rangle, |y-\rangle\}$.

We denote the classical dataset of size N to be

$$S_N(\mathcal{E}) \triangleq \left\{ |\psi_\ell^{(\text{in})}\rangle = \bigotimes_{i=1}^{n} |s_{\ell,i}^{(\text{in})}\rangle,\ |\psi_\ell^{(\text{out})}\rangle = \bigotimes_{i=1}^{n} |s_{\ell,i}^{(\text{out})}\rangle \right\}_{\ell=1}^{N}, \tag{6.8}$$

where $|s_{\ell,i}^{(\text{in})}\rangle, |s_{\ell,i}^{(\text{out})}\rangle \in \text{stab}_1$. Each product state is represented classically with $\mathcal{O}(n)$ bits. Hence, the classical dataset $S_N(\mathcal{E})$ is of size $\mathcal{O}(nN)$ bits. The classical dataset can be seen as one way to generalize the notion of classical shadows of quantum states (Huang, Richard Kueng, and Preskill, 2020) to quantum processes. Our goal is to design an ML algorithm that can learn an approximate model of $\mathcal{E}$ from the classical dataset $S_N(\mathcal{E})$, such that for a wide range of states ρ and observables O, the ML model can predict a real value $h(\rho, O)$ that is approximately equal to $\text{tr}(O\mathcal{E}(\rho))$.

ML algorithm

We are now ready to state the proposed ML algorithm. At a high level, the ML algorithm learns a low-degree approximation to the unknown n-qubit CPTP map $\mathcal{E}$. Despite the simplicity of the ML algorithm, several ideas go into the design of the ML algorithm and the proof of the rigorous performance guarantee. These ideas are presented in Section 6.3.

Let O be an observable with $\|O\| \leq 1$ that is written as a sum of few-body observables, where each qubit is acted by $\mathcal{O}(1)$ of the few-body observables. We denote the Pauli representation of O as $\sum_{Q\in\{I,X,Y,Z\}^{\otimes n}} a_Q Q$. By definition of O, there are $\mathcal{O}(n)$ nonzero Pauli coefficients a_Q. We consider a hyperparameter $\tilde{\epsilon} > 0$; roughly speaking $\tilde{\epsilon}$ will scale inverse polynomially in the dataset size N from Eq. (6.13). For every Pauli observable $P \in \{I, X, Y, Z\}^{\otimes n}$ with $|P| \leq k = \Theta(\log(1/\epsilon))$, the algorithm computes an empirical estimate for the corresponding Pauli coefficient α_P via

$$\hat{x}_P(O) = \frac{1}{N}\sum_{\ell=1}^{N} \text{tr}\left(P \bigotimes_{i=1}^{n} |s_{\ell,i}^{(\text{in})}\rangle\!\langle s_{\ell,i}^{(\text{in})}| \right) \text{tr}\left(O \bigotimes_{i=1}^{n} \left(3|s_{\ell,i}^{(\text{out})}\rangle\!\langle s_{\ell,i}^{(\text{out})}| - I \right) \right), \tag{6.9}$$

$$\hat{\alpha}_P(O) = \begin{cases} 3^{|P|}\hat{x}_P(O), & \left(\frac{1}{3}\right)^{|P|} > 2\tilde{\epsilon} \text{ and } |\hat{x}_P(O)| > 2\cdot 3^{|P|/2}\sqrt{\tilde{\epsilon}}\sum_{Q:a_Q\neq 0}|a_Q|, \\ 0, & \text{otherwise.} \end{cases} \tag{6.10}$$

The computation of $\hat{x}_P(O)$ and $\hat{\alpha}_P(O)$ can both be done classically. The basic idea of $\hat{\alpha}_P(O)$ is to set the coefficient $3^{|P|}\hat{x}_P(O)$ to zero when the influence of Pauli observable P is negligible. Given an n-qubit state ρ, the algorithm outputs

$$h(\rho, O) = \sum_{P:|P|\leq k} \hat{\alpha}_P(O)\, \mathrm{tr}(P\rho). \tag{6.11}$$

With a proper implementation, the computational time is $O(kn^k N)$. Note that, to make predictions, the ML algorithm only needs the k-body reduced density matrices (k-RDMs) of ρ. The k-RDMs of ρ can be efficiently obtained by performing randomized Pauli measurement on ρ and using the classical shadow formalism (Huang, Richard Kueng, and Preskill, 2020; Andreas Elben, Steven T Flammia, et al., 2022). Except for this step, which may require quantum computation, all other steps of the ML algorithm only requires classical computation. Hence, if the k-RDMs of ρ can be computed classically, then we have a classical ML algorithm that can predict an arbitrary quantum process $\mathcal{E}$ after learning from data.

Rigorous guarantee

To measure the prediction error of the ML model, we consider the average-case prediction performance under an arbitrary n-qubit state distribution $\mathcal{D}$ invariant under single-qubit Clifford gates, which means that the probability distribution $f_{\mathcal{D}}(\rho)$ of sampling a state ρ is equal to $f_{\mathcal{D}}(U\rho U^\dagger)$ of sampling $U\rho U^\dagger$ for any single-qubit Clifford gate U. We call such a distribution locally flat.

Theorem 25 (Learning an unknown quantum process)**.** *Given $\epsilon, \epsilon' = \Theta(1)$ and a training set $S_N(\mathcal{E})$ of size $N = O(\log n)$ as specified in Eq.* (6.8)*. With high probability, the ML model can learn a function $h(\rho, O)$ from $S_N(\mathcal{E})$ such that for any distribution $\mathcal{D}$ over n-qubit states invariant under single-qubit Clifford gates, and for any bounded-degree observable O with $\|O\| \leq 1$,*

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} |h(\rho, O) - \mathrm{tr}(O\mathcal{E}(\rho))|^2 \leq \epsilon + \max\left(\|O'\|^2, 1\right)\epsilon', \tag{6.12}$$

where O' is the low-degree truncation (of degree $k = \lceil \log_{1.5}(1/\epsilon) \rceil$) of the observable O after the Heisenberg evolution under $\mathcal{E}$. The training and prediction time of $h(\rho, O)$ are both polynomial in n. When ϵ is small and $\epsilon' = 0$, the data size N and computational time scale as $2^{O(\log(\frac{1}{\epsilon})\log(n))}$.

The detailed theorem statement and the proof of the theorem are given in Section 6.9. An interesting aspect of the above theorem is that the states sampled from the distribution $\mathcal{D}$ can be highly entangled, even though the training data $S_N(\mathcal{E})$ only contains information about random product states. From the theorem, we can see that if $\|O'\| = O(1)$, then we only need $O(\log(n))$ samples to obtain a constant prediction error. Otherwise, $O(\log(n))$ samples is still enough to guarantee a constant prediction error relative to $\|O'\|^2$. The precise scaling is given as follows. Consider data size

$$N = \log(n)\ \min\left(2^{O\left(\log(\frac{1}{\epsilon})\left(\log\log(\frac{1}{\epsilon})+\log(\frac{1}{\epsilon'})\right)\right)}, 2^{O(\log(\frac{1}{\epsilon})\log(n))}\right). \tag{6.13}$$

The computational time to learn and predict $h(\rho, O)$ is bounded above by $O(kn^k N)$ and the prediction error is bounded as

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} |h(\rho, O) - \mathrm{tr}(O\mathcal{E}(\rho))|^2 \leq \epsilon + \max\left(\|O'\|^2, 1\right)\epsilon'. \tag{6.14}$$

As we take ϵ' to be zero, we can remove the dependence on the low-degree truncation O'. In this setting, N and computation time both become $2^{O(\log(\frac{1}{\epsilon})\log(n))}$, which is polynomial in n if $\epsilon = \Theta(1)$.

6.3 Proof ideas

The proof of the rigorous performance guarantee for the proposed ML algorithm consists of five parts. The first two parts presented in Section 6.5 and Section 6.6 are a detour to establish a few fundamental and useful norm inequalities about Hamiltonians/observables. The latter three parts given in Section 6.7, Section 6.8, and Section 6.9 apply the newly-established norm inequalities to three learning tasks. In the following, we present the basic ideas in each part.

Improved approximation algorithms for optimizing local Hamiltonians

We begin with a different task, namely optimizing local Hamiltonians. We are given an n-qubit k-local Hamiltonian

$$H = \sum_{P\in\{I,X,Y,Z\}^{\otimes n}:|P|\leq k} \alpha_P P, \tag{6.15}$$

where $|P|$ is the weight of the Pauli operator P, the number of qubits upon which P acts nontrivially. Our goal is to find a state $|\psi\rangle$ that maximizes/minimizes $\langle\psi| H |\psi\rangle$. This task is related to solving ground states (Kempe, A. Kitaev, and Regev, 2006; Sakurai and Napolitano, 2017) when we consider minimizing $\langle\psi| H |\psi\rangle$ and quantum optimization (Farhi, Goldstone, and Gutmann, 2014a; Farhi, Goldstone, and Gutmann, 2014b; Aram W Harrow and Montanaro, 2017a; Parekh and Thompson, 2020; Hallgren, E. Y. Lee, and Parekh, 2020; Anshu, Gosset, et al., 2021; Matthew B Hastings and O'Donnell, 2022) when we consider maximizing $\langle\psi| H |\psi\rangle$.

We give a general randomized approximation algorithm in Section 6.5 for producing a random product state $|\psi\rangle$ that either approximately minimizes or approximately maximizes a k-local Hamiltonian H with a rigorous upper/lower bound based on the Pauli coefficients α_P of H. The proposed optimization algorithm applies to various classes of Hamiltonians and is inspired by the proofs of Littlewood's 4/3 inequality (Littlewood, 1930) and the Bohnenblust-Hille inequality (Bohnenblust and Hille, 1931). For classes that have been studied previously (Dinur et al., 2006; Barak et al., 2015; Aram W Harrow and Montanaro, 2017a; Anshu, Gosset, et al., 2021), the proposed algorithm obtains an improved bound. Our improvement crucially stems from our construction for the random state $|\psi\rangle$. (Dinur et al., 2006; Barak et al., 2015; Aram W Harrow and Montanaro, 2017a) utilize a random restriction approach, where some random subset of qubits are fixed with some random values and the rest of the qubits are optimized. On the other hand, we utilize a polarization approach, where we replicate each qubit many times, randomly fix all except the last replica, optimize the last replica, and combine using a random-signed averaging. A detailed comparison is given in Section 6.5 and 6.5.

Two classes of Hamiltonians used in our learning applications are general k-local Hamiltonians and bounded-degree k-local Hamiltonians. A k-local Hamiltonian with degree at most d is a Hermitian operator that can be written as a sum of k-qubit observables, where each qubit is acted on by at most d of the k-qubit observables.

Corollary 2 (Optimizing general k-local Hamiltonian)**.** *Given an n-qubit k-local Hamiltonian*

$$H = \sum_{P:|P|\leq k} \alpha_P P. \tag{6.16}$$

There is a randomized algorithm that runs in time $O(n^k)$ and produces either a random maximizing state $|\psi\rangle = |\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle$ satisfying

$$\mathop{\mathbb{E}}_{|\psi\rangle} \left[\langle\psi| H |\psi\rangle \right] \geq \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}} \left[\langle\phi| H |\phi\rangle \right] + C(k) \left(\sum_{P\neq I} |\alpha_P|^{2k/(k+1)} \right)^{(k+1)/(2k)}, \tag{6.17}$$

or a random minimizing state $|\psi\rangle = |\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle$ satisfying

$$\mathop{\mathbb{E}}_{|\psi\rangle} \left[\langle\psi| H |\psi\rangle \right] \leq \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}} \left[\langle\phi| H |\phi\rangle \right] - C(k) \left(\sum_{P\neq I} |\alpha_P|^{2k/(k+1)} \right)^{(k+1)/(2k)}, \tag{6.18}$$

where $C(k) = 1/\exp(\Theta(k \log k))$.

Corollary 3 (Optimizing bounded-degree k-local Hamiltonian)**.** *Given an n-qubit k-local Hamiltonian $H = \sum_{P:|P|\leq k} \alpha_P P$ with bounded degree d, $|\alpha_P| \leq 1$ for all P, and $k = O(1)$. There is a randomized algorithm that runs in time $O(nd)$ and produces either a random maximizing state $|\psi\rangle = |\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle$ satisfying*

$$\mathop{\mathbb{E}}_{|\psi\rangle} \left[\langle\psi| H |\psi\rangle \right] \geq \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}} \left[\langle\phi| H |\phi\rangle \right] + \frac{C}{\sqrt{d}} \sum_{P\neq I} |\alpha_P|, \tag{6.19}$$

or a random minimizing state $|\psi\rangle = |\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle$ satisfying

$$\mathop{\mathbb{E}}_{|\psi\rangle} \left[\langle\psi| H |\psi\rangle \right] \leq \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}} \left[\langle\phi| H |\phi\rangle \right] - \frac{C}{\sqrt{d}} \sum_{P\neq I} |\alpha_P| \tag{6.20}$$

for some constant C.

We note that in the above results, we cannot control whether our algorithm outputs an approximate maximizer or minimizer. This caveat stems from the use of polarization, where the random-signed averaging only guarantees improvement in one of the two directions. Modifying our approach to address this issue is an interesting direction for future work.

Norm inequalities from approximate optimization algorithms

The bridge that connects the optimization of k-local Hamiltonians and efficient learning of quantum states and processes is a set of norm inequalites. A norm that characterizes the efficiency of learning is the Pauli-p norm, defined as the ℓ_p-norm on the Pauli coefficients of an observable/Hamiltonian $H = \sum_P \alpha_P P$,

$$||H||_{\text{Pauli},p} \triangleq \left(\sum_{P\in\{I,X,Y,Z\}^{\otimes n}} |\alpha_P|^p \right)^{1/p}. \tag{6.21}$$

The rigorous guarantees from the previous section, namely on finding a state $|\psi\rangle$ whose energy is higher/lower than a Haar-random state by a margin that depends on the Pauli coefficients α_P, give an algorithmic proof that the spectral norm $||H||$ and the Pauli coefficients α_P are related. The proof of this relation is given in Section 6.6. In particular, for general and bounded-degree k-local Hamiltonian, we can use the rigorous guarantee from the approximation algorithms to obtain the following norm inequalites. Corollary 4 proves the conjecture given in (Rouzé, Wirth, and Haonan Zhang, 2022).

Corollary 4 (Norm inequality for general k-local Hamiltonian)**.** *Given an n-qubit k-local Hamiltonian H. We have*

$$\frac{1}{3} C(k) \, \|H\|_{\mathrm{Pauli}, \frac{2k}{k+1}} \leq \|H\| \, , \tag{6.22}$$

where $C(k) = 1/\exp(\Theta(k \log k))$.

Corollary 5 (Norm inequality for bounded-degree local Hamiltonian)**.** *Given an n-qubit k-local Hamiltonian H with a bounded degree d. We have*

$$\frac{1}{3} C(k, d) \, \|H\|_{\mathrm{Pauli}, 1} \leq \|H\| \, , \tag{6.23}$$

where $C(k, d) = 1/(\sqrt{d} \exp(\Theta(k \log k)))$.

Sample-optimal algorithm for predicting bounded-degree observables

As the first application of the above norm inequalities to learning, we consider the basic problem of predicting many properties of an unknown n-qubit state ρ. Given M observables $O_1, \ldots, O_M$, after performing measurements on multiple copies of ρ, we would like to predict $\mathrm{tr}(O_i \rho)$ to ϵ error for all $i \in \{1, \ldots, M\}$. This is the task known as shadow tomography (Aaronson, 2018; Aaronson and Rothblum, 2019; Huang, Richard Kueng, and Preskill, 2020). One approach for obtaining practically-efficient algorithms for shadow tomography is via the classical shadow formalism (Huang, Richard Kueng, and Preskill, 2020).

We consider a physically-relevant class of observables, where the observable $O_i = \sum_j O_{ij}$ is a sum of few-body observables O_{ij} and each qubit is acted on by $\mathcal{O}(1)$ of the few-body observables. Despite significant recent progress in shadow tomography (Levy, Luo, and Clark, 2021; Zhao, Rubin, and Miyake, 2021; H.-Y. Hu and You, 2021; Koh and Grewal, 2020; Senrui Chen, W. Yu, et al., 2021; Hadfield et al., 2020; Aaronson, 2018; Struchalin et al., 2021; Huang, Sitan Chen, and Preskill, 2023; O'Gorman, 2022; Wan, Huggins, et al., 2022; Bu et al., 2022; Huang, Broughton, J. Cotler, et al., 2022; Sitan Chen, J. Cotler, et al., 2021b; Coopmans, Kikuchi, and Benedetti, 2022), the sample complexity (number of copies of ρ) for predicting this class of observables has not been established. The central challenge is the appearance of the Pauli-1 norm $\|O_i\|_{\mathrm{Pauli},1}$ when characterizing the sample complexity. In particular, one can bound the shadow norm $\|O_i\|_{\mathrm{shadow}}$ (Huang, Richard Kueng, and Preskill, 2020), which gives an upper bound on the sample complexity in terms of the Pauli-1 norm $\|O_i\|_{\mathrm{Pauli},1}$ up to a constant factor. Using the new norm inequality established in this chapter, we give a sample-optimal algorithm for predicting bounded-degree observables.

The sample-optimal algorithm is equivalent to performing classical shadow tomography based on randomized Pauli measurements (Huang, Richard Kueng, and Preskill, 2020; Andreas Elben, Steven T Flammia, et al., 2022), and is essentially the ML algorithm given in Section 6.2 with a fixed input state. Consider an unknown n-qubit state ρ. After performing N randomized Pauli measurements on N copies of ρ, we have a classical dataset denoted as

$$S_N(\rho) \triangleq \left\{ |\psi_\ell^{(\text{out})}\rangle = \bigotimes_{i=1}^{n} |s_{\ell,i}^{(\text{out})}\rangle \right\}_{\ell=1}^{N}, \tag{6.24}$$

where $|s_{\ell,i}^{(\text{out})}\rangle \in \text{stab}_1$ is a single-qubit stabilizer state. Given an observable O, the algorithm predicts

$$h(O) = \frac{1}{N} \sum_{\ell=1}^{N} \operatorname{tr}\left(O \bigotimes_{i=1}^{n} \left(3|s_{\ell,i}^{(\text{out})}\rangle\!\langle s_{\ell,i}^{(\text{out})}| - I \right) \right). \tag{6.25}$$

It is not hard to see that computing $h(O)$ only requires $\mathcal{O}(nN)$ classical computation time. Hence, as we show later that $N = \mathcal{O}(\log(n)/\epsilon^2)$, the learning algorithm is very efficient. Using the norm inequality for bounded-degree local Hamiltonian $\|H\|_{\text{Pauli},1} \leq C\|H\|$ for a constant C in Corollary 5, and the classical shadow formalism (Huang, Richard Kueng, and Preskill, 2020; Andreas Elben, Steven T Flammia, et al., 2022), we obtain the following performance guarantee.

Theorem 26 (Sample complexity upper bound)**.** *Given an unknown n-qubit state ρ and any n-qubit observables $O_1, \ldots, O_M$ with $\|O_i\| \leq B_\infty$. Suppose each observable O_i is a sum of few-body observables, where each qubit is acted on by $\mathcal{O}(1)$ of the few-body observables. Using a classical dataset $S_N(\rho)$ of size*

$$N = \mathcal{O}\left(\frac{\log\left(\min(M, n)\right) B_\infty^2}{\epsilon^2} \right), \tag{6.26}$$

we have $|h(O_i) - \operatorname{tr}(O_i\rho)| \leq \epsilon, \forall i \in \{1, \ldots, M\}$ with high probability. The constant factor in the $\mathcal{O}(\cdot)$ notation above scales polynomially in the degree and exponentially in the locality of the observables.

The following theorem shows that the above algorithm achieves the optimal sample complexity among any algorithms that can perform collective measurement on many copies of ρ.

Theorem 27 (Sample complexity lower bound)**.** *Consider the following task. There is an unknown n-qubit state ρ, and we are given M observables $O_1, \ldots, O_M$ with* $\max_i \|O_i\| \leq B_\infty$*. Each observable O_i is a sum of few-body observables, where every qubit is acted on by $O(1)$ of the few-body observables. We would like to estimate* $\mathrm{tr}(O_i\rho)$ *to ϵ error for all $i \in [M]$ with high probability by performing arbitrary collective measurements on N copies of ρ. The number of copies N must be at least*

$$N = \Omega\left(\frac{\log\left(\min(M, n)\right) B_\infty^2}{\epsilon^2}\right) \tag{6.27}$$

for any algorithm to succeed in this task.

The detailed proofs of the sample complexity stated in the above theorems are given in Section 6.7.

Efficient algorithms for learning an unknown observable from $\log(n)$ samples

As a second learning application of the norm inequalities, we consider the task of learning an unknown n-qubit observable $O^{(\mathrm{unk})} = \sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \alpha_P P$. We can think of this unknown observable as $\mathcal{E}^\dagger(O)$, i.e., the observable O after Heisenberg evolution under the unknown process $\mathcal{E}$. Suppose we are given a training dataset of $\{\rho_\ell, \mathrm{tr}\left(O^{(\mathrm{unk})}\rho_\ell\right)\}_{\ell=1}^N$, where ρ_ℓ is sampled from an arbitrary distribution $\mathcal{D}$ over n-qubit states that is invariant under single-qubit Clifford gates. Given an integer $k > 0$, we define the weight-k truncation of $O^{(\mathrm{unk})}$ to be the following Hermitian operator

$$O^{(\mathrm{unk},k)} \triangleq \sum_{P \in \{I,X,Y,Z\}^{\otimes n} : |P| \leq k} \alpha_P P, \tag{6.28}$$

where $|P|$ is the number of qubits upon which P acts nontrivially. For a small k, we can think of $O^{(\mathrm{unk},k)}$ as a low-weight approximation of the unknown observable $O^{(\mathrm{unk})}$. By definition, $O^{(\mathrm{unk},k)}$ is a k-local Hamiltonian, hence the norm inequality in Corollary 4 shows that

$$\frac{1}{3}C(k)\left\|O^{(\mathrm{unk},k)}\right\|_{\mathrm{Pauli},\frac{2k}{k+1}} = \frac{1}{3}C(k)\left(\sum_{P \in \{I,X,Y,Z\}^{\otimes n} : |P| \leq k} |\alpha_P|^r\right)^{1/r} \leq \left\|O^{(\mathrm{unk},k)}\right\|, \tag{6.29}$$

where $r = 2k/(k+1) \in [1, 2)$. An ℓ_r norm bound ($r < 2$) on the Pauli coefficients implies that we can remove most of the small Pauli coefficients without incurring too much change under the ℓ_2 norm. As an example, consider an M-dimensional vector x with $\|x\|_r \leq 1$. Given $\widetilde{\epsilon} > 0$, let $\widetilde{x}$ be the M-dimensional vector with $\widetilde{x}_i = x_i$

if $|x_i| > \widetilde{\epsilon}$ and $\widetilde{x}_i = 0$ if $|x_i| \leq \widetilde{\epsilon}$. We have

$$\|x - \widetilde{x}\|_2^2 = \sum_{i:|x_i|\leq\widetilde{\epsilon}} |x_i|^2 \leq \widetilde{\epsilon}^{2-r} \sum_{i:|x_i|\leq\widetilde{\epsilon}} |x_i|^r \leq \widetilde{\epsilon}^{2-r} \sum_i |x_i|^r \leq \widetilde{\epsilon}^{2-r}. \tag{6.30}$$

In Section 6.8, we show that the average error (both the mean squared error and the mean absolute error) is characterized by the ℓ_2 norm. Hence, Eq. (6.29) implies that we can set most of the Pauli coefficients in $O^{(\text{unk},k)}$ to zero without incurring too much error on average.

Using the above reasoning, learning the low-weight truncation $O^{(\text{unk},k)}$ amounts to learning the large Pauli coefficients of $O^{(\text{unk},k)}$ and setting all small Pauli coefficients to zero. This ensures that the learning can be done very efficiently. This approach is presented in Section 6.8 with the main result stated in Lemma 45. It is inspired by the learning algorithm of (Eskenazis and Ivanisvili, 2022) that achieves a logarithmic sample complexity for learning classical low-degree functions.

The last step in the proof is to argue that the low-weight truncation $O^{(\text{unk},k)}$ is a good surrogate for the unknown observable $O^{(\text{unk})}$ when the goal is to predict $\text{tr}(O^{(\text{unk})}\rho)$. The key insight here is that for distributions $\mathcal{D}$ that are invariant under single-Clifford gates, the contribution of any Pauli term P in $O^{(\text{unk})}$ to $\mathbb{E}_{\rho\sim\mathcal{D}}[\text{tr}(O^{(\text{unk})}\rho)^2]$ is *exponentially decaying* in the the weight $|P|$. This allows us to prove that $\mathbb{E}_{\rho\sim\mathcal{D}}[\text{tr}((O^{(\text{unk})} - O^{(\text{unk},k)})\rho)^2]$ is small.

Putting these ingredients together, we arrive at the following theorem. As stated in the theorem, the learning algorithm is computationally efficient.

Theorem 28 (Learning an unknown observable)**.** *Given* $\epsilon, \epsilon', \delta > 0$. *Let* $k = \lceil \log_{1.5}(1/\epsilon) \rceil$ *and* $r = \frac{2k}{k+1} \in [1, 2)$. *From training data* $\{\rho_\ell, \text{tr}\left(O^{(\text{unk})}\rho_\ell\right)\}_{\ell=1}^N$ *of size*

$$N = \log(n/\delta)\ \min\left(2^{\mathcal{O}\left(\log(\frac{1}{\epsilon})\left(\log\log(\frac{1}{\epsilon})+\log(\frac{1}{\epsilon'})\right)\right)}, 2^{\mathcal{O}(\log(\frac{1}{\epsilon})\log(n))}\right), \tag{6.31}$$

where ρ_ℓ *is sampled from* $\mathcal{D}$*, we can learn a function* $h(\rho)$ *such that*

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \left|h(\rho) - \text{tr}(O^{(\text{unk})}\rho)\right|^2 \leq (\epsilon + \epsilon') \left\|O^{(\text{unk})}\right\|^2 + \epsilon' \left\|O^{(\text{unk},k)}\right\|^r \left\|O^{(\text{unk})}\right\|^{2-r} \tag{6.32}$$

with probability at least $1-\delta$*. The training and prediction time of* $h(\rho)$ *are* $\mathcal{O}(Nn^k)$*.*

The factor of $\left\|O^{(\text{unk})}\right\|^2$ in the prediction error is the natural scale of the squared error. From the theorem, we can see that we only need $\mathcal{O}(\log(n))$ samples to obtain a constant prediction error relative to $\left\|O^{(\mathrm{unk})}\right\|^2 + \left\|O^{(\mathrm{unk},k)}\right\|^r \left\|O^{(\mathrm{unk})}\right\|^{2-r}$. The proof the the theorem and the detailed description of the ML algorithm are given in Section 6.8.

Learning an unknown quantum process

The ML algorithm for learning an unknown n-qubit quantum process $\mathcal{E}$ is essentially the combination of the two learning applications described above with a few modifications. At a high level, we consider the following. There is an n-qubit state ρ sampled from an unknown distribution $\mathcal{D}$, as well as an observable O that can be written as a sum of few-body observables, where each qubit is acted on by a constant number of the few-body observables. In the first stage, we use the sample-optimal algorithm for predicting the bounded-degree observable O, where $\mathcal{E}(\rho_\ell)$ is an unknown quantum state, thus transforming the classical dataset $S_N(\mathcal{E})$ in Eq. (6.8) into a dataset,

$$\left\{\rho_\ell \triangleq |\psi_\ell^{(\mathrm{in})}\rangle\langle\psi_\ell^{(\mathrm{in})}|,\ \mathrm{tr}\left(O\mathcal{E}\left(\rho_\ell\right)\right)\right\}_{\ell=1}^{N} \tag{6.33}$$

that maps quantum states to real numbers. In the second stage, we apply the efficient algorithm for learning an unknown observable $O^{(\mathrm{unk})} = \mathcal{E}^\dagger(O)$, regarding Eq. (6.33) as the training data for this task, thus predicting $\mathrm{tr}\left(\mathcal{E}^\dagger(O)\rho\right) = \mathrm{tr}\left(O\mathcal{E}\left(\rho\right)\right)$ for the state ρ drawn from the distribution $\mathcal{D}$. Because both stages of the algorithm run in time polynomial in n, the overall runtime for this procedure is polynomial in n.

In our actual proofs, there are a few deviations from the above high-level design, stemming from the fact that the input states ρ_ℓ are tensor products of random single-qubit stabilizer states. This specific setting allows a few simplifications to be made. With the simplifications, we can remove an additive factor of ϵ' in the prediction error. Furthermore, a surprising fact is that learning from random product states is sufficient to predict highly-entangled states sampled from any distribution $\mathcal{D}$ invariant under single-qubit Clifford unitaries. This surprising fact is a result of the characterization of the prediction error given in Lemma 41 based on a modified purity on subsystems of an input quantum state $\rho \sim \mathcal{D}$.

By combining the five parts together, we can establish Theorem 25, the precise sample complexity scaling in Eq. (6.13), and the prediction error bound in Eq. (6.14). The full proof is given in Section 6.9.

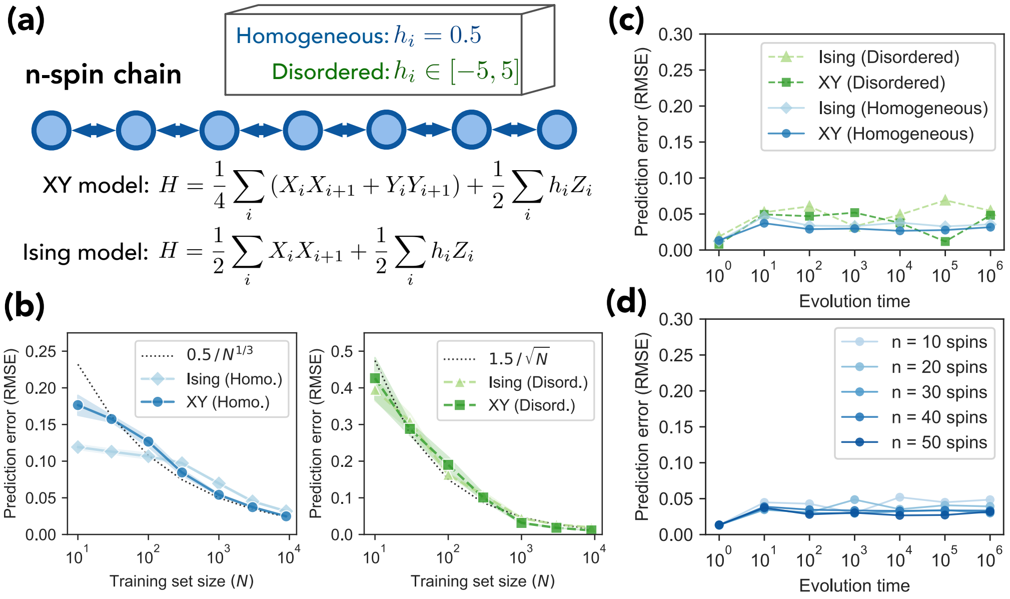


Figure 6.2: *Prediction performance of ML models for learning $\mathcal{E}(\rho) = e^{-itH}\rho e^{itH}$ for a large time t.* (A) HAMILTONIANS. We consider XY/Ising model with a homogeneous/disordered Z field on an n-spin open chain. (B) ERROR SCALING WITH TRAINING SET SIZE (N). We show the root-mean-square error (RMSE) for predicting the Pauli-Z operator Z_i on the output state $\mathcal{E}(\rho)$ for random product states ρ. (C, D) ERROR SCALING WITH EVOLUTION TIME (t) AND SYSTEM SIZE (n). (d) shows the RMSE for the XY model with a homogeneous Z field. The prediction error remains similar as we exponentially increase t and Hilbert space dimension 2^n.

6.4 Numerical experiments

We have conducted numerical experiments to assess the performance of ML models in learning the dynamics of several physical systems. The results corroborate our theoretical claims that long-time evolution over a many-body system can be learned efficiently. While our theorem only guarantees good performance for randomly sampled input states, we also find that the ML models work very well for structured input states that could be of practical interest. The source code can be found on a public GitHub repository[1].

We focus on training ML models to predict output state properties after the time dynamics of 1D n-spin XY/Ising chains with homogeneous/disordered Z fields. Let H be the many-body Hamiltonian. The quantum processes $\mathcal{E}$ is given by $\mathcal{E}(\rho) = e^{-itH}\rho e^{itH}$ for a significantly long evolution time $t = 10^6$. We consider

[1] https://github.com/hsinyuan-huang/learning-quantum-process

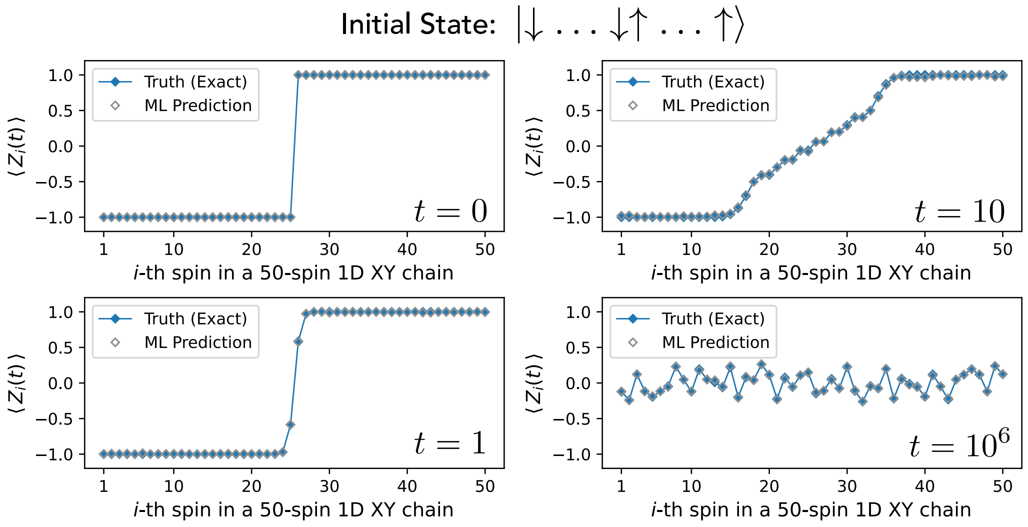


Figure 6.3: *Visualization of ML model's prediction for an initial state $\rho = |\psi\rangle\langle\psi|$ with a domain wall.* We consider the 1D 50-spin XY chain with a homogeneous Z field. We show the expectation value of $Z_i(t) = e^{itH} Z_i e^{-itH}$ for all the 50 spins on the initial state $|\psi\rangle = |\downarrow \ldots \downarrow\uparrow \ldots \uparrow\rangle$. The ML model is trained on 10000 random product states. We see that the ML model performs accurately for a significantly large range of time t.

the ML models described by Eq. (6.11). While we utilize the very simple sparsity-enforcing strategy of setting small values to zero to prove Theorem 25, the standard sparsity-enforcing approach is through ℓ_1 regularization (R. Tibshirani, 1996). A detailed description of applying ℓ_1 regularization to enforce sparsity in $\alpha_P(O)$ is given in Section 6.10. We find the best hyperparameters using four-fold cross-validation to minimize root-mean-square error (RMSE) and report the predictions on a test set.

Fig. 6.2 considers the performance for predicting the expectation of the Pauli-Z operator Z_i on the output state for randomly sampled product input states not in the training data. Fig. 6.2(a) illustrates the many-body Hamiltonian H. Fig. 6.2(b) shows the dependence of the error on training set size N. We can clearly see that as training set size N increases, the prediction error notably decreases. This observation confirms our theoretical claim that long-time quantum dynamics could be efficiently learned. In Fig. 6.2(c), we consider how evolution time t affects prediction performance. From the figure, we can see that even when we exponentially increase t, the prediction performance remains similar. This matches with our theorem stating that no matter what the quantum process $\mathcal{E}$ is, even if $\mathcal{E}$ is an exponentially long-time dynamics, the ML model can still predict accurately and efficiently. In Fig. 6.2(d), we consider the dependence on system size n. As n increases linearly, the Hilbert space dimension 2^n grows exponentially. Despite the exponential growth, even for 50-spin systems, the ML model still predicts well. This matches with the logarithmic scaling on n given in Theorem 25.

In Fig. 6.3, we consider predicting properties of the final state after long-time dynamics for a highly structured input product state:

$$|\psi\rangle = |\downarrow \ldots \downarrow\uparrow \ldots \uparrow\rangle, \tag{6.34}$$

which has a single domain wall in the middle. We focus on predicting the expected value for $Z_i(t) = e^{itH} Z_i e^{-itH}$ on every spin in the 1D 50-spin XY chain with a homogeneous Z field $h_i = 0.5$ and consider evolution time t from 0 to 10^6. We train the ML model using $N = 10000$ random input product states. We can see that the ML model predicts very well for this highly structured product state. The ML model accurately predicts the collapse of the domain wall despite only seeing outcomes from random unstructured product states. This numerical experiment suggests that the performance of the ML model goes beyond Theorem 25, which only guarantees accurate prediction on average.

Theorem 25 states that the ML model can predict well on highly-entangled input states after learning only from random product state inputs. We test this claim in Fig. 6.4 by considering an entangled input state

$$|\psi_e\rangle = \sum_{\substack{s \in \{\leftarrow, \rightarrow\}^{n/2} \\ \text{w/ even \# of} \rightarrow}} \frac{1}{\sqrt{2^{(n/2)-1}}} |s\rangle \otimes |{\rightarrow}{\downarrow}{\leftarrow}{\uparrow}{\rightarrow}{\downarrow}{\leftarrow}{\uparrow} \ldots\rangle. \tag{6.35}$$

The left $n/2$ spins of the state $|\psi_e\rangle$ exhibit GHZ-like entanglement, which requires a linear-depth 1D quantum circuit to prepare. The right $n/2$ spins of $|\psi_e\rangle$ form a product state with spins rotating clockwise from left to right. Combining the left and right spins together, the state $|\psi_e\rangle$ cannot be generated by a short-depth 1D quantum circuit. We can see that for this entangled input state, the ML model trained on random product states still predicts very well across a broad range of the evolution time t.

6.5 Optimizing k-local Hamiltonian with random product states

While our goal is to design a good machine learning (ML) algorithm with low sample complexity, this section is a detour to a different task on the optimization

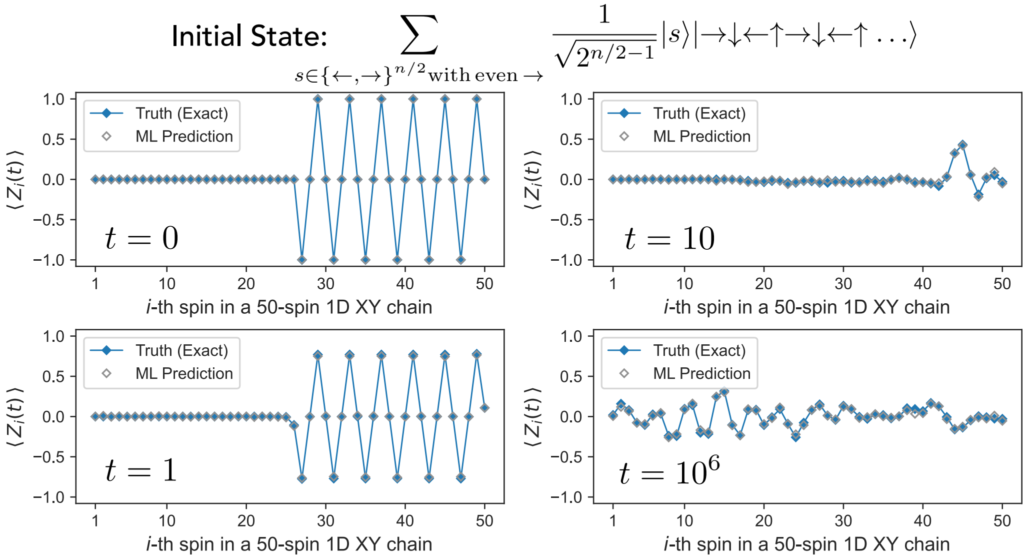


Figure 6.4: *Visualization of ML model's prediction for a highly-entangled initial state* $\rho = |\psi\rangle\langle\psi|$. We consider the expected value of $Z_i(t) = e^{itH} Z_i e^{-itH}$, where H corresponds to the 1D 50-spin XY chain with a homogeneous Z field. The initial state $|\psi\rangle$ has a GHZ-like entanglement over the left-half chain and is a product state with spins rotating clockwise over the right-half chain. To prepare $|\psi\rangle$ with 1D circuits, a depth of at least $\Omega(n)$ is required. Even though the ML model is trained only on random product states (a total of $N = 10000$), it still performs accurately in predicting the highly-entangled state over a wide range of evolution time t.

of a k-local Hamiltonian. We present an improved approximation algorithm for optimizing any k-local Hamiltonian. The central result in this detour will become useful for showing the low sample complexity of several ML algorithms.

Task description and main theorem

Task 1 (Optimizing quantum Hamiltonian)**.** *Given $n, k \geq 1$ and an n-qubit k-local Hamiltonian*

$$H = \sum_{P\in\{I,X,Y,Z\}^{\otimes n}:|P|\leq k} \alpha_P P, \tag{6.36}$$

where $|P|$ is the number of non-identity components in P. Find a state $|\psi\rangle$ that maximizes/minimizes $\langle\psi| H |\psi\rangle$.

The task given above is related to solving ground states (Kempe, A. Kitaev, and Regev, 2006; Sakurai and Napolitano, 2017) when we consider minimizing $\langle\psi| H |\psi\rangle$ and quantum optimization (Farhi, Goldstone, and Gutmann, 2014a; Farhi, Goldstone, and Gutmann, 2014b; Aram W Harrow and Montanaro, 2017a; Parekh and Thompson, 2020; Hallgren, E. Y. Lee, and Parekh, 2020; Anshu, Gosset, et al., 2021; Matthew B Hastings and O'Donnell, 2022) when we consider maximizing $\langle\psi| H |\psi\rangle$. The maximization and minimization are often the same problem since maximizing $\langle\psi| H |\psi\rangle$ is the same as minimizing $\langle\psi| (-H) |\psi\rangle$. Without further constraints, even for $k = 2$, finding the optimal state $|\psi^*\rangle$ maximizing $\langle\psi| H |\psi\rangle$ is known to be QMA-hard (Piddock and Montanaro, 2015), hence it is expected to have no polynomial-time algorithm even on a quantum computer. Most existing works consider deterministic or randomized constructions of $|\psi\rangle$ with rigorous upper/lower bound guarantees on $\langle\psi| H |\psi\rangle$ for minimization/maximization. Some of these lower bounds (Parekh and Thompson, 2020; Hallgren, E. Y. Lee, and Parekh, 2020; Matthew B Hastings and O'Donnell, 2022) are based on the optimal value $\mathrm{OPT} = \sup_{|\psi\rangle} \langle\psi| H |\psi\rangle$, while some (Farhi, Goldstone, and Gutmann, 2014b; Aram W Harrow and Montanaro, 2017a; Anshu, Gosset, et al., 2021) are based on the Pauli coefficients α_P.

Definition of expansion

In this section, we present a random product state construction for the optimization problem, where the rigorous upper/lower bound is based on the Pauli coefficients α_P and the expansion property defined below. The expansion property is defined for any Hamiltonian H.

Definition 3 (Expansion property)**.** *Given an n-qubit Hamiltonian $H = \sum_P \alpha_P P$. We say H has an expansion coefficient c_e and expansion dimension d_e if for any $\Upsilon \subseteq \{1, \ldots, n\}$ with $|\Upsilon| = d_e$,*

$$\sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \mathbb{1}\Big[\alpha_P \neq 0 \text{ and } \big(\Upsilon \subseteq \mathsf{dom}(P) \text{ or } \mathsf{dom}(P) \subseteq \Upsilon\big)\Big] \leq c_e, \qquad (6.37)$$

where $\mathsf{dom}(P)$ is the set of qubits that P acts nontrivially on.

The expansion property captures the connectivity of the Hamiltonian. We give two examples, general k-local Hamiltonian and geometrically-local Hamiltonian, to provide more intuition on the expansion property.

Fact 2 (Expansion property for general k-local Hamiltonian)**.** *Any Hamiltonian given by a sum of k-qubit observables has expansion coefficient 4^k and expansion dimension k.*

Proof. Let $H = \sum_P \alpha_P P$. All the Pauli observables P with nonzero α_P act at most on k qubits. For any Υ with $|\Upsilon| = k$, all the Pauli observables with nonzero α_P must have a domain contained in Υ. There are at most 4^k such Pauli observables. Hence, the claim follows. □

Fact 3 (Expansion property for bounded-degree k-local Hamiltonian)**.** *Any Hamiltonian given by a sum of k-qubit observables $H = \sum_j h_j$, where each qubit is acted on by at most d of the k-qubit observables h_j, has expansion coefficient $c_e = 4^k d$ and expansion dimension $d_e = 1$.*

Proof. For every Υ with $|\Upsilon|$, $\Upsilon = \{i\}$ for some qubit i. For each qubit i (corresponding to $\Upsilon = \{i\}$), we have at most d k-qubit observables acting on i. Each of the k-qubit observables can be expanded into at most 4^k Pauli terms. Hence we can set $c_e = 4^k d$ and $d_e = 1$. □

Fact 4 (Expansion property for geometrically-local Hamiltonian)**.** *Any Hamiltonian given by a sum of geometrically-local observables has expansion coefficient $c_e = O(1)$ and expansion dimension* 1.

Proof. For a geometrically-local Hamiltonian $H = \sum_P \alpha_P P$, each qubit i is acted by at most a constant number $c_i = O(1)$ of P with non-zero α_P. Hence for any qubit i, $\sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \mathbb{1}[\alpha_P \neq 0 \text{ and } (\Upsilon \subseteq \mathsf{dom}(P) \text{ or } \mathsf{dom}(P) \subseteq \Upsilon)] = c_i$. Thus, we can set $d_e = 1$ and $c_e = \max_i c_i = O(1)$. □

Main theorem

With the expansion property defined, we can state the rigorous guarantee on the performance of the proposed randomized approximation algorithm on optimizing an n-qubit k-local Hamiltonian H. We compare with the average energy $\mathbb{E}_{|\phi\rangle:\text{Haar}}\left[\langle\phi| H |\phi\rangle\right] = \alpha_I$ over Haar random state. The randomized approximation algorithm uses an optimization over a single-variable polynomial that guarantees improvement in at least one direction (minimization or maximization).

Theorem 29 (Random product states for optimizing k-local Hamiltonian)**.** *Given an n-qubit k-local Hamiltonian $H = \sum_{P:|P|\leq k} \alpha_P P$ with expansion coefficient/dimension c_e, d_e. Let $r = 2d_e/(d_e+1) \in [1,2)$ and $\mathsf{nnz}(H) \triangleq |\{P : \alpha_P \neq 0\}|$. There is a randomized algorithm that runs in time $O(nk + \mathsf{nnz}(H)2^k)$ and produces either* *a random maximizing state $|\psi\rangle = |\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle$ satisfying*

$$\mathop{\mathbb{E}}_{|\psi\rangle} \left[\langle\psi| H |\psi\rangle \right] \geq \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}} \left[\langle\phi| H |\phi\rangle \right] + C(c_e, d_e, k) \left(\sum_{P \neq I} |\alpha_P|^r \right)^{1/r}, \tag{6.38}$$

or a random minimizing state $|\psi\rangle = |\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle$ satisfying

$$\mathop{\mathbb{E}}_{|\psi\rangle} \left[\langle\psi| H |\psi\rangle \right] \leq \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}} \left[\langle\phi| H |\phi\rangle \right] - C(c_e, d_e, k) \left(\sum_{P \neq I} |\alpha_P|^r \right)^{1/r}. \tag{6.39}$$

The constant $C(c_e, d_e, k)$ is given by

$$C(c_e, d_e, k) = \frac{\sqrt{2(k!)}}{c_e^{1/(2d_e)} k^{k+1.5+1/r} (\sqrt{6} + 2\sqrt{3})^k} = \Theta_k \left(\frac{1}{c_e^{1/(2d_e)}} \right), \tag{6.40}$$

where Θ_k considers the asymptotic scaling when k is a constant.

Some observations can be made. First, the improvement over Haar random states in Theorem 29 becomes larger when the expansion coefficient c_e is smaller. Second, $(\sum_{P \neq I} |\alpha_P|^r)^{1/r}$ is the ℓ_r-norm on the non-identity Pauli coefficients, so by monotonicity of ℓ_r-norms, $(\sum_{P \neq I} |\alpha_P|^r)^{1/r}$ becomes smaller as r becomes larger (corresponding to larger d_e). Hence, the improvement is greater for smaller expansion dimension d_e. In particular, it is helpful to contrast Eqs. (6.38) and (6.39) with the following basic estimate corresponding to $r = 2$ which holds regardless of c_e, d_e, k:

$$\sup_{|\psi\rangle} \left| \langle\psi| H |\psi\rangle - \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}} \left[\langle\phi| H |\phi\rangle \right] \right| \geq \left(\sum_{P \neq I} |\alpha_P|^2 \right)^{1/2}. \tag{6.41}$$

This holds for any Hamiltonian $H = \sum_P \alpha_P P$ because

$$\left(\sum_{P \neq I} |\alpha_P|^2 \right)^{1/2} = \frac{1}{2^{n/2}} \|H - \alpha_I I\|_F \leq \|H - \alpha_I I\|_\infty, \tag{6.42}$$

where $\|\cdot\|$ denotes spectral norm, and $\alpha_I = \mathbb{E}_{|\psi\rangle:\text{Haar}} \left[\langle\phi| H |\phi\rangle \right]$. This basic estimate shows that we can always find a state that improves by at least the ℓ_2-norm of α_P, although the optimization process can be computationally hard.

An alternative version of the main theorem

By following the proof of Theorem 29 and replacing the use of Corollary 10 by Lemma 32, we can establish the following alternative theorem statement that does not utilize the expansion property.

Theorem 30 (Random product states for optimizing k-local Hamiltonian; alternative)**.** *Given an n-qubit k-local Hamiltonian* $H = \sum_{P:|P|\leq k} \alpha_P P$ *with* $k = O(1)$*. Let* $\mathsf{nnz}(H) \triangleq |\{P : \alpha_P \neq 0\}|$*. There is a randomized algorithm that runs in time* $O(nk + \mathsf{nnz}(H)2^k)$ *and produces a random state* $|\psi\rangle = |\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle$ *satisfying*

$$\left| \mathop{\mathbb{E}}_{|\psi\rangle} \left[\langle\psi| H |\psi\rangle \right] - \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}} \left[\langle\phi| H |\phi\rangle \right] \right| \geq D \sum_{i\in[n],p\in\{X,Y,Z\}} \sqrt{\sum_{P:P_i=p} \alpha_P^2}, \tag{6.43}$$

for some constant D.

We can compare the above theorem with a closely related result in (Aram W Harrow and Montanaro, 2017a). The following is a restatement of the approximation guarantee from Theorem 2 and Lemma 3 in (Aram W Harrow and Montanaro, 2017a), which is a corollary of a powerful result in Boolean function analysis (Dinur et al., 2006; Barak et al., 2015) relating the maximum influence and the ability to sample a bitstring from the Boolean hypercube with a large magnitude in the function value. We can define the influence of qubit i under Pauli matrix $p \in \{X, Y, Z\}$ as $I(i, p) = \sum_{P:P_i=p} \alpha_P^2$.

Theorem 31 (Approximation guarantee from (Aram W Harrow and Montanaro, 2017a) for optimizing k-local Hamiltonian)**.** *Given an n-qubit k-local Hamiltonian*

$$H = \sum_{P:|P|\leq k} \alpha_P P \text{ with } k = O(1). \tag{6.44}$$

There is a polynomial-time randomized algorithm that produces a random state $|\psi\rangle = |\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle$ *satisfying*

$$\left| \mathop{\mathbb{E}}_{|\psi\rangle} \left[\langle\psi| H |\psi\rangle \right] - \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}} \left[\langle\phi| H |\phi\rangle \right] \right| \geq D \sum_{i\in[n],p\in\{X,Y,Z\}} \frac{\sum_{P:P_i=p} \alpha_P^2}{\max_{j,q} \sqrt{\sum_{P:P_j=q} \alpha_P^2}}, \tag{6.45}$$

for some constant D.

The guarantee from (Aram W Harrow and Montanaro, 2017a) is asymptotically optimal when the influence $I(i, p)$ are of a similar magnitude for different qubit i and Pauli matrix p. However, the approximation guarantee can be far from optimal when there is a large variation in the influence $I(i, p)$ over different qubits i, p. As an example, consider a 1D n-qubit nearest-neighbor chain, where $|\alpha_P| = 1$ for only a constant number of Pauli observables P and $|\alpha_P| = 1/\sqrt{n}$ for the rest of the Pauli observables. The improvements over Haar random state by our algorithm and the algorithm in (Aram W Harrow and Montanaro, 2017a) are respectively given by

$$\Theta\left(\sum_{i\in[n],p\in\{X,Y,Z\}} \sqrt{\sum_{P:P_i=p} \alpha_P^2}\right) = \Theta\left(\sqrt{n}\right), \tag{6.46}$$

$$\Theta\left(\sum_{i\in[n],p\in\{X,Y,Z\}} \frac{\sum_{P:P_i=p} \alpha_P^2}{\max_{j,q} \sqrt{\sum_{P:P_j=q} \alpha_P^2}}\right) = \Theta\left(1\right). \tag{6.47}$$

Hence, when there is large variation in the influence, our guarantee improves over that of (Aram W Harrow and Montanaro, 2017a). For our machine learning applications, the removal of the dependence on the maximum influence is central. By removing the ratio $\sqrt{I(i,p)}/\max_{j,q}\sqrt{I(j,q)}$, we can obtain the ℓ_r norm dependence for an $r < 2$ as given in Theorem 29. We will later see that having the ℓ_r norm bound (for $r < 2$) allows a substantial reduction in the sample complexity in training machine learning models for predicting properties.

We do want to mention that the improvement comes at a cost of a slightly worse dependence on $k = O(1)$. In Theorem 31 from (Aram W Harrow and Montanaro, 2017a) based on Boolean function analysis (Dinur et al., 2006; Barak et al., 2015), the dependence on D is $1/2^{\Theta(k)}$. However, our result in Theorem 30 is $D = 1/2^{\Theta(k \log k)}$. This difference stems from the construction for the random state $|\psi\rangle$. (Dinur et al., 2006; Barak et al., 2015; Aram W Harrow and Montanaro, 2017a) utilize a random restriction approach, where some random subset of variables are fixed with some random values and the rest of the variables are optimized. On the other hand, we utilize a polarization approach, where we replicate each variable many times, randomly fix all except the last replica, optimize the last replica, and combine using a random-signed averaging.

Corollaries of the main theorem

Here, we consider how the main theorem applies to certain classes of k-local Hamiltonians and discuss the relations of the corollaries to related works.

Optimizing arbitrary k-local Hamiltonians

The first corollary considers a general k-local Hamiltonian $H = \sum_{P:|P|\leq k} \alpha_P P$. We can combine Fact 2 and the main theorem to obtain the following corollary.

Corollary 6 (Optimizing arbitrary k-local Hamiltonian)**.** *Given an n-qubit k-local Hamiltonian $H = \sum_{P:|P|\leq k} \alpha_P P$. There is a randomized algorithm that runs in time*

$O(n^k)$ and produces a random product state $|\psi\rangle = |\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle$ with

$$\left| \mathop{\mathbb{E}}_{|\psi\rangle} \left[\langle\psi| H |\psi\rangle \right] - \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}} \left[\langle\phi| H |\phi\rangle \right] \right| \geq C(k) \left(\sum_{P \neq I} |\alpha_P|^{2k/(k+1)} \right)^{(k+1)/(2k)}, \tag{6.48}$$

where $C(k) = \frac{\sqrt{2(k!)}}{2k^{k+1.5+(k+1)/(2k)}(\sqrt{6}+2\sqrt{3})^k}$.

For $k = 2$, we have $2k/(k+1) = 4/3$ and the above result resembles Littlewood's $4/3$ inequality. Recall that Littlewood's $4/3$ inequality states that given $\{\beta_{i,j} \in \mathbb{C}\}_{i,j}$,

$$\sup \left\{ \left| \sum_{i,j} \beta_{i,j} x_i^{(1)} x_j^{(2)} \right| : x_i^{(k)} \in \mathbb{C}, \left| x_i^{(k)} \right| \leq 1, \forall i \in \mathbb{N}, k \in \{1,2\} \right\} \tag{6.49}$$

$$\geq \frac{1}{\sqrt{2}} \left(\sum_{i,j} |\beta_{i,j}|^{4/3} \right)^{3/4}. \tag{6.50}$$

For $k > 2$, the above result resembles Bohnenblust-Hille inequality, which states that given $\{\beta_{i_1,\ldots,i_k} \in \mathbb{C}\}_{i_1,\ldots,i_k}$,

$$\sup \left\{ \left| \sum_{i_1,\ldots,i_k} \beta_{i_1,\ldots,i_k} x_{i_1}^{(1)} \ldots x_{i_k}^{(k)} \right| : x_{i_\kappa}^{(\kappa)} \in \mathbb{C}, \left| x_{i_\kappa}^{(\kappa)} \right| \leq 1, \forall i_\kappa \in \mathbb{N}, \kappa \in [k] \right\} \tag{6.51}$$

$$\geq D_k \left(\sum_{i_1,\ldots,i_k} |\beta_{i_1,\ldots,i_k}|^{2k/(k+1)} \right)^{(k+1)/(2k)}, \tag{6.52}$$

for some constant D_k that depends on k. For optimizing general k-local Hamiltonian, the design of the randomized approximation algorithm is inspired by the original proof (Bohnenblust and Hille, 1931) of Bohnenblust-Hille inequality from 1931, which is used to study the absolute convergence of Dirichlet series.

Optimizing bounded-degree k-local Hamiltonians

Here, we consider a Hamiltonian given by a sum of k-qubit observables, where each qubit is acted on by at most d of the k-qubit observables. This is often referred to as a k-local Hamiltonian with a bounded degree d. We can combine Fact 3 and the main theorem to obtain the following corollary.

Corollary 7 (Optimizing bounded-degree k-local Hamiltonian)**.** *Given an n-qubit k-local Hamiltonian $H = \sum_{P:|P| \leq k} \alpha_P P$ with bounded degree d, $|\alpha_P| \leq 1$ for all P, and $k = O(1)$. There is a randomized algorithm that runs in time $O(nd)$ and* *produces either a random maximizing state* $|\psi\rangle = |\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle$ *satisfying*

$$\mathop{\mathbb{E}}_{|\psi\rangle}\left[\langle\psi| H |\psi\rangle\right] \geq \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}}\left[\langle\phi| H |\phi\rangle\right] + \frac{C}{\sqrt{d}}\sum_{P\neq I}|\alpha_P|, \tag{6.53}$$

or a random minimizing state $|\psi\rangle = |\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle$ *satisfying*

$$\mathop{\mathbb{E}}_{|\psi\rangle}\left[\langle\psi| H |\psi\rangle\right] \leq \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}}\left[\langle\phi| H |\phi\rangle\right] - \frac{C}{\sqrt{d}}\sum_{P\neq I}|\alpha_P| \tag{6.54}$$

for some constant C.

The task of optimizing bounded-degree k-local Hamiltonians has been considered in previous work (Anshu, Gosset, et al., 2021).

Theorem 32 (Approximation guarantee from (Anshu, Gosset, et al., 2021))**.** *Given an* n*-qubit* 2*-local Hamiltonian* $H = \sum_{P:|P|\leq 2}\alpha_P P$ *with bounded degree* d*, and* $|\alpha_P| \leq 1$ *for all* P*. There is a polynomial-time randomized algorithm that produces a quantum circuit that generates a random maximizing state* $|\psi\rangle$ *satisfying*

$$\mathop{\mathbb{E}}_{|\psi\rangle}\left[\langle\psi| H |\psi\rangle\right] \geq \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}}\left[\langle\phi| H |\phi\rangle\right] + \frac{C}{d}\left(\sum_{P\neq I}|\alpha_P|^2\right)\cdot\frac{\sum_{P\neq I}|\alpha_P|^2}{\sum_{P\neq I}\mathbb{1}[\alpha_P\neq 0]}, \tag{6.55}$$

as well as a random minimizing state $|\psi\rangle$ *satisfying*

$$\mathop{\mathbb{E}}_{|\psi\rangle}\left[\langle\psi| H |\psi\rangle\right] \leq \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}}\left[\langle\phi| H |\phi\rangle\right] - \frac{C}{d}\left(\sum_{P\neq I}|\alpha_P|^2\right)\cdot\frac{\sum_{P\neq I}|\alpha_P|^2}{\sum_{P\neq I}\mathbb{1}[\alpha_P\neq 0]} \tag{6.56}$$

for some constant C.

The result from (Anshu, Gosset, et al., 2021) considers a single-step gradient descent using a shallow quantum circuit on an initial random product state. Because $\sum_{P\neq I}|\alpha_P|^2 \leq \sum_{P\neq I}\mathbb{1}[\alpha_P \neq 0]$ and $\sum_{P\neq I}|\alpha_P| \geq \sum_{P\neq I}|\alpha_P|^2$, our result in Corollary 7 improves either the maximization problem or the minimization problem over Theorem 32. For example, if we consider $\alpha_P = \Theta(1/d)$, which sets the total interaction strength on each qubit to be $\Theta(1)$, then the improvement over Haar random state by our algorithm and the algorithm in (Anshu, Gosset, et al., 2021) is given by

$$\Theta\left(\frac{1}{\sqrt{d}}\sum_{P\neq I}|\alpha_P|\right) = \Theta\left(\frac{n}{d^{1.5}}\right), \Theta\left(\frac{1}{\sqrt{d}}\left(\sum_{P\neq I}|\alpha_P|^2\right)\cdot\frac{\sum_{P\neq I}|\alpha_P|^2}{\sum_{P\neq I}\mathbb{1}[\alpha_P\neq 0]}\right) = \Theta\left(\frac{n}{d^{4.5}}\right). \tag{6.57}$$

We can see that our algorithm gives a larger improvement for the scaling with the degree d. As another example, consider a 1D n-qubit nearest-neighbor chain (hence

$d = 2$), where $|\alpha_P| = 1$ for only a constant number of Pauli observables P and $|\alpha_P| = 1/\sqrt{n}$ for the rest of the Pauli observables. The improvement over Haar random state by our algorithm and the algorithm in (Anshu, Gosset, et al., 2021) is given by

$$\Theta\left(\frac{1}{\sqrt{d}}\sum_{P\neq I}|\alpha_P|\right) = \Theta\left(\sqrt{n}\right), \; \Theta\left(\frac{1}{\sqrt{d}}\left(\sum_{P\neq I}|\alpha_P|^2\right)\cdot\frac{\sum_{P\neq I}|\alpha_P|^2}{\sum_{P\neq I}\mathbb{1}[\alpha_P\neq 0]}\right) = \Theta\left(\frac{1}{n}\right). \tag{6.58}$$

We can see that our algorithm gives a larger improvement for the scaling with the number n of qubits.

Description of the randomized approximation algorithm

There are a few steps in the proposed randomized algorithm. The first step is to choose the best slice of the k-local Hamiltonian by splitting the k-local Hamiltonian $H = \sum_{P:|P|\leq k}\alpha_P P$ as follows,

$$H = \alpha_I I + \sum_{\kappa=1}^{k} H_\kappa, \quad H_\kappa \triangleq \sum_{P:|P|=\kappa}\alpha_P P. \tag{6.59}$$

We choose $\kappa^* \in \{1,\ldots,k\}$ to be the κ that maximizes $\sum_{P:|P|=\kappa}|\alpha_P|^r$, where $r = 2d_e/(d_e+1)$. This step can be performed in time $\mathcal{O}(\mathsf{nnz}(H)k)$.

In the second step, the algorithm samples $(\kappa^*-1)n$ Haar-random single-qubit pure states,

$$|\psi_{(s,j)}\rangle \in \mathbb{C}^2, \quad \forall s \in \{1,\ldots,\kappa^*-1\}, j \in \{1,\ldots,n\}. \tag{6.60}$$

This step can be performed in time $\mathcal{O}(nk)$.

The third step is a local optimization on each qubit based on $|\psi_{(s,j)}\rangle$. For each qubit i and Pauli matrix $p \in \{X,Y,Z\}$, we define an $(n-1)$-qubit homogeneous (κ^*-1)-local Hermitian operator,

$$H_{\kappa^*,i,p} \triangleq \sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_i=p}} \alpha_P\left(\bigotimes_{j\neq i} P_j\right), \tag{6.61}$$

For each qubit i and $p \in \{X,Y,Z\}$, the algorithm computes the real value given as follows,

$$\beta_{i,p} \triangleq \tag{6.62}$$

$$\mathop{\mathbb{E}}_{\sigma\in\{\pm1\}^{\kappa^*-1}}\left[\sigma_1\cdots\sigma_{\kappa^*-1}\,\mathrm{tr}\left[H_{\kappa^*,i,p}\bigotimes_{j\neq i}\left[\frac{I}{2}+\frac{1}{\kappa^*-1}\sum_{s=1}^{\kappa^*-1}\sigma_s\left(|\psi_{(s,j)}\rangle\!\langle\psi_{(s,j)}|-\frac{I}{2}\right)\right]\right]\right] \tag{6.63}$$

Then for each qubit j, we consider a single-qubit local optimization

$$|\psi_{(\kappa^*,j)}\rangle \triangleq \underset{|\phi\rangle\text{: 1-qubit state}}{\arg\max} \langle\phi| \left(\sum_{p\in\{X,Y,Z\}} \beta_{j,p} p \right) |\phi\rangle = \frac{I + n_X X + n_Y Y + n_Z Z}{2}, \quad (6.64)$$

where $n_p = \beta_{j,p} / \sqrt{\sum_q \beta_{j,q}^2}$ for $p \in \{X, Y, Z\}$. After the optimization, the algorithm samples random numbers $\sigma_s \in \{\pm 1\}, \forall s \in \{1, \ldots, \kappa^*\}$ to define a one-dimensional parameterized family of n-qubit product states,

$$\rho\left(t; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right) \triangleq \bigotimes_{j=1}^{n} \left(\frac{I}{2} + \frac{t}{\kappa^*} \sum_{s=1}^{\kappa^*} \sigma_s \left(|\psi_{(s,j)}\rangle\langle\psi_{(s,j)}| - \frac{I}{2} \right) \right), \quad \forall t \in [-1, 1]. \quad (6.65)$$

We will denote this by $\rho(t)$ when $|\psi_{(\cdot,\cdot)}\rangle, \sigma$ are clear from context. This concludes the third step. The third step can be performed in time $O(\mathsf{nnz}(H)2^k)$.

The fourth step performs a polynomial optimization over the one-dimensional family,

$$\max_{t\in[-1,1]} \left| \mathrm{tr}\left(H\rho\left(t; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right) \right) - \alpha_I \right|. \quad (6.66)$$

The function $f(t) = \mathrm{tr}(H\rho(t))$ is a polynomial of degree at most k. We can compute the function $f(t)$ efficiently in time $O(\mathsf{nnz}(H)k)$ as $\rho(t)$ is a product state. The optimization can thus be performed efficiently by sweeping through all possible values of t on a sufficiently fine grid. Let t^* be the optimal t.

The final step considers the sampling of a random pure state $|\psi\rangle = |\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle$ from the distribution that corresponds to the mixed state $\rho\left(t^*; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)$. If $\mathrm{tr}(H\rho\left(t^*; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)) - \alpha_I > 0$, then the random product state $|\psi\rangle$ is a maximizing state satisfying Eq. (6.38). Otherwise, the random product state $|\psi\rangle$ is a minimizing state satisfying Eq. (6.39). This step can be performed in time $O(n)$.

Proof of Theorem 29

The first step of the algorithm considers splitting the k-local Hamiltonian H into homogeneous κ-local Hamiltonians H_κ defined below. In particular, a homogeneous κ^*-local H_{κ^*} is chosen.

Definition 4 (Homogeneous k-local)**.** *A Hermitian operator H is homogeneous k-local if $H = \sum_{P:|P|=k} \alpha_P P$.*

The second step is a random sampling that generates a single-qubit pure state $|\psi_{(s,j)}\rangle$ for each qubit j and each copy $s \in \{1, \ldots, \kappa^* - 1\}$. The third step is the most important part of the proof. We will devote Section 6.5, 6.5, and 6.5 to establish the first inequality given below (Corollary 10).

$$\mathop{\mathbb{E}}_{|\psi_{(\cdot,\cdot)}\rangle}\mathop{\mathbb{E}}_{\sigma\in\{\pm1\}^{\kappa^*}}\left|\operatorname{tr}\left(H_{\kappa^*}\rho\left(t=1;|\psi_{(\cdot,\cdot)}\rangle,\sigma\right)\right)\right| \tag{6.67}$$

$$\geq \frac{\sqrt{2(k!)}}{c_e^{1/(2d_e)}k^{k+1.5}\sqrt{6}^k}\left(\sum_{P:|P|=\kappa^*}|\alpha_P|^r\right)^{1/r} \tag{6.68}$$

$$\geq \frac{\sqrt{2(k!)}}{c_e^{1/(2d_e)}k^{k+1.5+1/r}\sqrt{6}^k}\left(\sum_{P\neq I}|\alpha_P|^r\right)^{1/r}. \tag{6.69}$$

The second inequality follows from

$$k\sum_{P:|P|=\kappa^*}|\alpha_P|^r \geq \sum_{\kappa=1}^{k}\sum_{P:|P|=\kappa}|\alpha_P|^r = \sum_{P\neq I}|\alpha_P|^r. \tag{6.70}$$

For the fourth step, the analysis of polynomial optimization given in Section 6.5 (Corollary 11) can be combined with the above inequality to obtain

$$\mathop{\mathbb{E}}_{|\psi_{(\cdot,\cdot)}\rangle}\mathop{\mathbb{E}}_{\sigma\in\{\pm1\}^{\kappa^*}}\left|\operatorname{tr}\left(H\rho\left(t^*;|\psi_{(\cdot,\cdot)}\rangle,\sigma\right)\right)-\alpha_I\right| \tag{6.71}$$

$$\geq \frac{\sqrt{2(k!)}}{c_e^{1/(2d_e)}k^{k+1.5+1/r}(\sqrt{6}(1+\sqrt{2}))^k}\left(\sum_{P\neq I}|\alpha_P|^r\right)^{1/r}. \tag{6.72}$$

For the final step of the algorithm, using $\mathbb{E}_{|\psi\rangle}|\psi\rangle\langle\psi| = \rho(t^*;\rho_{(s,j)},\sigma_s)$ and convexity, we have

$$\mathop{\mathbb{E}}_{|\psi_{(\cdot,\cdot)}\rangle}\mathop{\mathbb{E}}_{\sigma\in\{\pm1\}^{\kappa^*}}\mathop{\mathbb{E}}_{|\psi\rangle}\left|\langle\psi|H|\psi\rangle-\alpha_I\right| \tag{6.73}$$

$$\geq \mathop{\mathbb{E}}_{|\psi_{(\cdot,\cdot)}\rangle}\mathop{\mathbb{E}}_{\sigma\in\{\pm1\}^{\kappa^*}}\left|\operatorname{tr}\left(H\mathop{\mathbb{E}}_{|\psi\rangle}|\psi\rangle\langle\psi|\right)-\alpha_I\right| \tag{6.74}$$

$$\geq \frac{\sqrt{2(k!)}}{c_e^{1/(2d_e)}k^{k+1.5+1/r}(\sqrt{6}+2\sqrt{3})^k}\left(\sum_{P\neq I}|\alpha_P|^r\right)^{1/r}. \tag{6.75}$$

The theorem follows by noting that $\mathbb{E}_{|\phi\rangle:\text{Haar}}\left[\langle\phi|H|\phi\rangle\right] = \alpha_I$.

Polarization

We justify the definition of $\beta_{i,p}$ using polarization. Given an n-qubit homogeneous k-local observable $O = \sum_{P:|P|=k}\alpha_P P$, consider the following nk-qubit observable. First, we will index the set $[nk]$ using ordered tuples (s,i) where $s\in[k]$ and $i\in[n]$. For every Pauli operator P on n qubits with $|P|=k$, suppose that it acts nontrivially on qubits $i_1 < \cdots < i_k$ via Pauli matrices $P_{i_1}, \ldots, P_{i_k}$. Then for any permutation $\pi \in \mathcal{S}_k$, consider the nk-qubit observable $\mathsf{pol}_\pi(P)$ which acts on the $(\pi(s), i_s)$-th qubit via P_{i_s} for all $s \in [k]$. Then define

$$\mathsf{pol}(P) \coloneqq \frac{1}{k!} \sum_{\pi \in \mathcal{S}_k} \mathsf{pol}_\pi(P). \tag{6.76}$$

We can extend $\mathsf{pol}(\cdot)$ linearly and define $\mathsf{pol}(O) \triangleq \sum_P \alpha_P \mathsf{pol}(P)$. We refer to $\mathsf{pol}(O)$ as the *polarization* of O. The squared Frobenius norm of O and $\mathsf{pol}(O)$ are related by

$$\operatorname{tr}(O^2) = k! \operatorname{tr}(\mathsf{pol}(O)^2). \tag{6.77}$$

We prove the following operator analogue of the classical polarization identity:

Lemma 28 (Polarization identity)**.** *For any nk-qubit product state*

$$\rho = \bigotimes_{s \in [k]} \left[\bigotimes_{i \in [n]} \rho_{(s,i)} \right] \tag{6.78}$$

and any n-qubit homogeneous k-local observable O and any $t \in \mathbb{R}$, we have the following identity

$$t^k \operatorname{tr}(\mathsf{pol}(O)\rho) \tag{6.79}$$

$$= \frac{k^k}{k!} \mathop{\mathbb{E}}_{\sigma \in \{\pm 1\}^k} \left[\sigma_1 \cdots \sigma_k \cdot \operatorname{tr}\left(O \bigotimes_{i \in [n]} \left\{ \frac{I}{2} + \frac{t}{k} \sum_{s=1}^{k} \sigma_s \left(\rho_{(s,i)} - \frac{I}{2} \right) \right\} \right) \right], \tag{6.80}$$

where the expectation is with respect to the uniform measure on $\{\pm 1\}^k$.

Proof. Let $O = \sum_{P:|P|=k} \alpha_P P$. By the multinomial theorem, we can expand the right-hand side to get

$$\frac{t^k}{k!} \sum_{P:|P|=k} \alpha_P \mathop{\mathbb{E}}_{\sigma} \Bigg[\sigma_1 \cdots \sigma_k \sum_{0 \le s_1, \ldots, s_n \le k} \tag{6.81}$$

$$\operatorname{tr}\left(P \bigotimes_{i=1}^{n} \left\{ \frac{I}{2} \cdot \mathbb{1}[s_i = 0] + \sigma_{s_i} \left(\rho_{s_i,i} - \frac{I}{2} \right) \cdot \mathbb{1}[s_i > 0] \right\} \right) \Bigg]. \tag{6.82}$$

For a given Pauli operator P, note that the only terms in the inner summation that are nonzero are given by $(s_1, \ldots, s_n)$ satisfying that if $s_i > 0$, then P acts nontrivially on the i-th qubit, because otherwise $\operatorname{tr}(\rho_{s_i,i} - I/2) = 0$ and the corresponding summand vanishes. Furthermore, for $(s_1, \ldots, s_n)$ satisfying this property, if $\{1, \ldots, k\}$ do not each appear exactly once, then

$$\sigma_1 \cdots \sigma_k \cdot \bigotimes_{i=1}^{n} \left\{ \frac{I}{2} \cdot \mathbb{1}[s_i = 0] + \sigma_{s_i} \left(\rho_{s_i,i} - \frac{I}{2} \right) \cdot \mathbb{1}[s_i > 0] \right\} \tag{6.83}$$

$$= \sigma_1^{c_1} \cdots \sigma_k^{c_k} \cdot \bigotimes_{i=1}^{n} \left\{ \frac{I}{2} \cdot \mathbb{1}[s_i = 0] + \left(\rho_{s_i,i} - \frac{I}{2} \right) \cdot \mathbb{1}[s_i > 0] \right\} \tag{6.84}$$

for $0 \leq c_1, \ldots, c_k \leq k$ such that $c_s = 1$ for some $s \in [k]$. In this case, the expectation of this term with respect to σ vanishes. Altogether, we conclude that for P which acts via $P_1, \ldots, P_k$ on qubits $1 \leq i_1 < \cdots < i_k \leq n$ and via identity elsewhere, the corresponding expectation over σ in Eq. (6.82) is given by

$$\sum_{\pi \in \mathcal{S}_k} \mathrm{tr} \left(\bigotimes_{s=1}^{k} P_j \left(\rho_{\pi(s),i_s} - \frac{I}{2} \right) \right) = \sum_{\pi \in \mathcal{S}_k} \mathrm{tr} \left(\bigotimes_{s=1}^{k} P_s \rho_{\pi(s),i_s} \right) = \sum_{\pi} \mathrm{tr}(\mathsf{pol}_\pi(P)\rho), \tag{6.85}$$

from which the lemma follows. □

Using the polarization identity, we can obtain the following corollary, which shows that $\beta_{i,p}$ is defined to be proportional to the expection of the polarization $\mathsf{pol}(H_{\kappa^*,i,p})$ of the homogeneous κ^*-local observable $H_{\kappa^*,i,p}$ on the tensor product of $n(\kappa^* - 1)$ single-qubit Haar-random states. We will later study the expectation value of the polarized observable on random product states.

Corollary 8. *From the definitions given in Section 6.5, we have*

$$\mathrm{tr} \left(\mathsf{pol}(H_{\kappa^*,i,p}) \bigotimes_{s \in [\kappa^* - 1], i \in [n]} |\psi_{(s,j)}\rangle\langle\psi_{(s,j)}| \right) = \frac{(\kappa^* - 1)^{\kappa^* - 1}}{(\kappa^* - 1)!} \beta_{i,p}. \tag{6.86}$$

Proof. The claim follows from the polarization identity in Lemma 28 and the definition of $\beta_{i,p}$ in Eq. (6.62). □

Khintchine inequality for polarized observables

We recall the following basic result in high-dimensional probability.

Lemma 29 (Standard Khintchine inequality (Haagerup, 1981))**.** *Consider $\varepsilon_1, \ldots, \varepsilon_n$ to be i.i.d. random variables with $P(\varepsilon_i = \pm 1) = 1/2$. For any $a_1, \ldots, a_n \in \mathbb{R}$, we have*

$$\frac{1}{\sqrt{2}} \left(\sum_{i=1}^{n} a_i^2 \right)^{1/2} \leq \mathop{\mathbb{E}}_{\varepsilon_1, \ldots \varepsilon_n} \left| \sum_{i=1}^{n} a_i \varepsilon_i \right| \leq \left(\sum_{i=1}^{n} a_i^2 \right)^{1/2}. \tag{6.87}$$

We prove an analogue of the Khintchine inequality when we replace the random ±1 variables with random product states and replace $a_1, \dots, a_n$ with a homogeneous 1-local observable.

Lemma 30 (Khintchine inequality for homogeneous 1-local observables)**.** *Let $n \geq 1$. Consider $|\psi\rangle = \bigotimes_{i=1}^n |\psi_i\rangle$ where $|\psi_i\rangle$ is a single-qubit Haar-random pure state. For any homogeneous 1-local n-qubit observable O,*

$$\frac{1}{\sqrt{6}}\sqrt{\operatorname{tr}(O^2)/2^n} \leq \mathop{\mathbb{E}}_{|\psi\rangle}\left[|\langle\psi|\, O\, |\psi\rangle|\right] \leq \frac{1}{\sqrt{3}}\sqrt{\operatorname{tr}(O^2)/2^n}. \tag{6.88}$$

Proof. A homogeneous 1-local observable O is $\sum_{i=1}^n \sum_{j=1}^3 \alpha_{ij} P_i^j$, where P_i^j is the Pauli matrix $\sigma_j \in \{X, Y, Z\}$ on the i-th qubit. Given n single-qubit unitaries $U_1, \dots, U_n$, we consider O under the rotated Pauli basis

$$O = \sum_{i=1}^n \sum_{j=1}^3 \alpha_{ij}^U U_i^\dagger P_i^j U_i. \tag{6.89}$$

Using the orthogonality of Pauli matrices, we have

$$\sqrt{\operatorname{tr}(O^2)/2^n} = \left(\sum_{i=1}^n \sum_{j=1}^3 (a_{ij}^U)^2\right)^{1/2} \tag{6.90}$$

under any rotated Pauli basis. We will utilize the rotated Pauli basis to establish the claimed results.

A single-qubit Haar-random pure state $|\psi_i\rangle$ can be sampled as follows. First, we sample a random single-qubit unitary U_i. Then, we consider $|\psi_i\rangle$ to be sampled uniformly from the set of 8 pure states,

$$\Upsilon^{U_i} = \left\{ \frac{I + \frac{1}{\sqrt{3}}(s_i^X U_i X U_i^\dagger + s_i^Y U_i Y U_i^\dagger + s_i^Z U_i Z U_i^\dagger)}{2} \,\middle|\, s_i^X, s_i^Y, s_i^Z \in \{\pm 1\} \right\}. \tag{6.91}$$

Using this sampling formulation and the rotated Pauli basis representation for O, we have

$$\begin{aligned}\mathop{\mathbb{E}}_{|\psi\rangle}\left[|\langle\psi|\, O\, |\psi\rangle|\right] &= \mathop{\mathbb{E}}_{U_i} \mathop{\mathbb{E}}_{|\psi_i\rangle\sim\Upsilon^{U_i}} \left| \sum_{i=1}^n \sum_{j=1}^3 \alpha_{ij}^U \operatorname{tr}\left(U_i^\dagger P_i^j U_i |\psi_i\rangle\langle\psi_i|\right)\right| &(6.92)\\ &= \frac{1}{\sqrt{3}} \mathop{\mathbb{E}}_{U_i} \mathop{\mathbb{E}}_{s_i^X, s_i^Y, s_i^Z \sim\{\pm 1\}} \left| \sum_{i=1}^n \alpha_{i1}^U s_i^X + \alpha_{i2}^U s_i^Y + \alpha_{i3}^U s_i^Z \right|. &(6.93)\end{aligned}$$

Using the standard Khintchine inequality given in Lemma 29, we have

$$\frac{1}{\sqrt{2}}\left(\sum_{i=1}^{n}\sum_{j=1}^{3}(a_{ij}^{U})^2\right)^{1/2} \tag{6.94}$$

$$\leq \mathop{\mathbb{E}}_{s_i^X,s_i^Y,s_i^Z\sim\{\pm 1\}}\left|\sum_{i=1}^{n}\alpha_{i1}^{U}s_i^X+\alpha_{i2}^{U}s_i^Y+\alpha_{i3}^{U}s_i^Z\right| \leq \left(\sum_{i=1}^{n}\sum_{j=1}^{3}(a_{ij}^{U})^2\right)^{1/2}. \tag{6.95}$$

Using Eq. (6.90), we can obtain

$$\frac{1}{\sqrt{6}}\mathop{\mathbb{E}}_{U_i}\sqrt{\mathrm{tr}(O^2)/2^n} \leq \mathop{\mathbb{E}}_{|\psi\rangle}\left[\left|\langle\psi|\,O\,|\psi\rangle\right|\right] \leq \frac{1}{\sqrt{3}}\mathop{\mathbb{E}}_{U_i}\sqrt{\mathrm{tr}(O^2)/2^n}, \tag{6.96}$$

which implies the claimed result. □

We prove the left half of Khintchine inequality for polarized observables. The right half can be shown using a similar proof, but we are only going to use the left half stated below.

Lemma 31 (Khintchine inequality for polarized observables)**.** *Given $n, k > 0$. Consider an nk-qubit observable $O = \mathsf{pol}(O')$, which is the polarization of an n-qubit homogeneous k-local observable O'. Consider $|\psi\rangle = \bigotimes_{s\in[k],i\in[n]}|\psi_{(s,i)}\rangle$ where $|\psi_{(s,i)}\rangle$ is a single-qubit Haar-random pure state. We have*

$$\left(\frac{1}{\sqrt{6}}\right)^k\sqrt{\mathrm{tr}(O^2)/2^n} \leq \mathop{\mathbb{E}}_{|\psi\rangle}\left[\left|\langle\psi|\,O\,|\psi\rangle\right|\right]. \tag{6.97}$$

Proof. For $\ell \in [3n]$, define $P^{(\ell)}$ to be an n-qubit observable equal to the Pauli matrix $\sigma_{1+(\ell \bmod 3)} \in \{X, Y, Z\}$ acting on the $\lceil \ell/3 \rceil$-th qubit. From the definition of polarization, we can represent O as

$$O = \sum_{\ell_1,\ldots,\ell_k\in[3n]}\alpha_{\ell_1,\ldots,\ell_k}P^{(\ell_1)}\otimes\ldots\otimes P^{(\ell_k)}. \tag{6.98}$$

For arbitrary coefficients $\alpha_{\ell_1,\ldots,\ell_k} \in \mathbb{R}$, we prove the following claim by induction on k,

$$\left(\frac{1}{\sqrt{6}}\right)^k\left(\sum_{\ell_1,\ldots,\ell_k\in[3n]}\alpha_{\ell_1,\ldots,\ell_k}^2\right)^{1/2} \tag{6.99}$$

$$\leq \mathop{\mathbb{E}}_{|\psi\rangle}\left[\left|\langle\psi|\sum_{\ell_1,\ldots,\ell_k\in[3n]}\alpha_{\ell_1,\ldots,\ell_k}P^{(\ell_1)}\otimes\ldots\otimes P^{(\ell_k)}\,|\psi\rangle\right|\right]. \tag{6.100}$$

It is not hard to see that the left-hand side of Eq. (6.99) is $\left(\frac{1}{\sqrt{6}}\right)^k \sqrt{\operatorname{tr}(O^2)/2^n}$ and the right-hand side of Eq. (6.99) is $\mathbb{E}_{|\psi\rangle}\left[|\langle\psi| O |\psi\rangle|\right]$. Hence, the lemma follows from Eq. (6.99).

We now prove the base case and the inductive step. The base case of $k = 1$ follows from Khintchine inequality for homogeneous 1-local observables given in Lemma 30. Assume by induction hypothesis that the claim holds for $k - 1$. By denoting $|\psi^{(k)}\rangle$ to be a product of n Haar-random single-qubit states, we can then apply Khintchine inequality for homogeneous 1-local observables (Lemma 30) to obtain

$$\left(\frac{1}{\sqrt{6}}\right)^k \left(\sum_{\ell_1,\ldots,\ell_k \in [3n]} \alpha^2_{\ell_1,\ldots,\ell_k} \right)^{1/2} \tag{6.101}$$

$$= \left(\sum_{\ell_1,\ldots,\ell_{k-1} \in [3n]} \left(\left(\sum_{\ell_k \in [3n]} \alpha^2_{\ell_1,\ldots,\ell_k} \right)^{1/2} \right)^2 \right)^{1/2} \tag{6.102}$$

$$\leq \left(\sum_{\ell_1,\ldots,\ell_{k-1} \in [3n]} \left(\mathop{\mathbb{E}}_{|\psi^{(k)}\rangle} \left| \langle\psi^{(k)}| \sum_{\ell_k \in [3n]} \alpha_{\ell_1,\ldots,\ell_k} P^{(\ell_k)} |\psi^{(k)}\rangle \right| \right)^2 \right)^{1/2}. \tag{6.103}$$

We can then apply Minkowski's integral inequality to upper bound the above and yield

$$\left(\frac{1}{\sqrt{6}}\right)^k \left(\sum_{\ell_1,\ldots,\ell_k \in [3n]} \alpha^2_{\ell_1,\ldots,\ell_k} \right)^{1/2} \tag{6.104}$$

$$\leq \mathop{\mathbb{E}}_{|\psi^{(k)}\rangle} \left(\sum_{\ell_1,\ldots,\ell_{k-1} \in [3n]} \left(\langle\psi^{(k)}| \sum_{\ell_k \in [3n]} \alpha_{\ell_1,\ldots,\ell_k} P^{(\ell_k)} |\psi^{(k)}\rangle \right)^2 \right)^{1/2} \tag{6.105}$$

$$\leq \mathop{\mathbb{E}}_{|\psi^{(k)}\rangle} \mathop{\mathbb{E}}_{|\psi^{(1,\ldots,k-1)}\rangle} \left| \langle\psi^{(1,\ldots,k-1)}| \langle\psi^{(k)}| \right. \tag{6.106}$$

$$\left. \sum_{\ell_1,\ldots,\ell_k \in [3n]} \alpha_{\ell_1,\ldots,\ell_k} P^{(\ell_1)} \otimes \ldots \otimes P^{(\ell_k)} |\psi^{(1,\ldots,k-1)}\rangle |\psi^{(k)}\rangle \right|. \tag{6.107}$$

The last line considers $\langle\psi^{(k)}| \sum_{\ell_k \in [3n]} \alpha_{\ell_1,\ldots,\ell_k} P^{(\ell_k)} |\psi^{(k)}\rangle$ to be a scalar indexed by $\ell_1, \ldots, \ell_{k-1}$ and uses the induction hypothesis. We have thus established the induction step. The claim in Eq. (6.99) follows. □

Khintchine inequality for polarized observable allows us to show that the average magnitude of $\mathsf{pol}(H_{\kappa^*,i,p})$ for the tensor product of single-qubit Haar-random states is at least as large as the Frobenius norm of $H_{\kappa^*,i,p}$ up to a constant depending on κ^*. Using the definitions from the design of the approximate optimization algorithm, we can obtain the following corollary.

Corollary 9. *From the definitions given in Section 6.5, we have*

$$\mathop{\mathbb{E}}_{|\psi_{(\cdot,\cdot)}\rangle}\left|\mathrm{tr}\left(\mathsf{pol}(H_{\kappa^*,i,p})\bigotimes_{s\in[\kappa^*-1],i\in[n]}|\psi_{(s,j)}\rangle\langle\psi_{(s,j)}|\right)\right| \geq \left(\frac{1}{\sqrt{6}}\right)^{\kappa^*-1}\sqrt{\frac{\mathrm{tr}(H^2_{\kappa^*,i,p})}{2^n(\kappa^*-1)!}}. \tag{6.108}$$

Proof. The claim follows immediately from Lemma 31 and Eq. (6.77). □

Characterization of the locally optimized random state

Recall that $\rho\left(1;|\psi_{(\cdot,\cdot)}\rangle,\sigma\right)$ is created by sampling random product states and performing local single-qubit optimizations. The locally optimized random state satisfies the following inequality.

Lemma 32 (Characterization of $\rho(t)$ for $t = 1$)**.** *From the definitions given in Section 6.5, we have*

$$\mathop{\mathbb{E}}_{|\psi_{(\cdot,\cdot)}\rangle}\mathop{\mathbb{E}}_{\sigma\in\{\pm1\}^{\kappa^*}}\left|\mathrm{tr}\left(H_{\kappa^*}\rho\left(1;|\psi_{(\cdot,\cdot)}\rangle,\sigma\right)\right)\right| \tag{6.109}$$

$$\geq \frac{\sqrt{2(\kappa^*!)}}{(\kappa^*)^{\kappa^*+1.5}\sqrt{6}^{\kappa^*}}\sum_{i\in[n],p\in\{X,Y,Z\}}\sqrt{\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\|P|=\kappa^*,P_i=p}}\alpha_P^2}. \tag{6.110}$$

Proof. From the polarization identity given in Lemma 28, we have

$$\frac{(\kappa^*)^{\kappa^*}}{\kappa^*!}\mathop{\mathbb{E}}_{\sigma\in\{\pm1\}^{\kappa^*}}\left[\sigma_1\cdots\sigma_{\kappa^*}\,\mathrm{tr}\left(H_{\kappa^*}\rho\left(1;|\psi_{(\cdot,\cdot)}\rangle,\sigma\right)\right)\right] \tag{6.111}$$

$$= \mathrm{tr}\left(\mathsf{pol}(H_{\kappa^*})\bigotimes_{s\in[\kappa^*],j\in[n]}|\psi_{(s,j)}\rangle\langle\psi_{(s,j)}|\right). \tag{6.112}$$

Next, using the definition of $H_{\kappa^*,i,p}$ in Eq. (6.61), we have

$$\mathsf{pol}(H_{\kappa^*}) = \left(\frac{1}{\kappa^*}\right)^2\sum_{i\in[n]}\sum_{p\in\{X,Y,Z\}}\mathsf{pol}(H_{\kappa^*,i,p})\otimes(I^{\otimes i-1}\otimes p\otimes I^{\otimes n-i}). \tag{6.113}$$

We can see this by considering the case when H_{κ^*} is a single Pauli observable $P \in \{I, X, Y, Z\}^{\otimes n}$ with $|P| = \kappa^*$, and then extending linearly to any homogeneous κ^*-local Hamiltonian H_{κ^*}. Eq. (6.111) and (6.113) give

$$\frac{(\kappa^*)^{\kappa^*}}{\kappa^*!} \mathop{\mathbb{E}}_{\sigma\in\{\pm 1\}^{\kappa^*}} \left[\sigma_1 \cdots \sigma_{\kappa^*} \operatorname{tr}\left(H_{\kappa^*}\rho\left(1; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right)\right] \tag{6.114}$$

$$= \frac{1}{(\kappa^*)^2} \sum_{\substack{i\in[n],\\ p\in\{X,Y,Z\}}} \langle\psi_{(\kappa^*,i)}| \, p \, |\psi_{(\kappa^*,i)}\rangle \operatorname{tr}\left(\mathsf{pol}(H_{\kappa^*,i,p}) \bigotimes_{s\in[\kappa^*-1], j\in[n]} |\psi_{(s,j)}\rangle\!\langle\psi_{(s,j)}|\right). \tag{6.115}$$

From Corollary 8, we can rewrite the right hand side as

$$\frac{1}{(\kappa^*)^2} \frac{(\kappa^*-1)^{\kappa^*-1}}{(\kappa^*-1)!} \sum_{i\in[n]} \langle\psi_{(\kappa^*,i)}| \left(\sum_{p\in\{X,Y,Z\}} \beta_{i,p} p\right) |\psi_{(\kappa^*,i)}\rangle \,. \tag{6.116}$$

From the local optimization of $|\psi_{(\kappa^*,i)}\rangle$ given in Eq. (6.64), we have that for every $i \in [n]$,

$$\langle\psi_{(\kappa^*,i)}| \left(\sum_{p\in\{X,Y,Z\}} \beta_{i,p} p\right) |\psi_{(\kappa^*,i)}\rangle = \sqrt{\sum_{p\in\{X,Y,Z\}} \beta_{i,p}^2} \geq \frac{1}{\sqrt{3}} \sum_{p\in\{X,Y,Z\}} \left|\beta_{i,p}\right|. \tag{6.117}$$

Using Corollary 8 yields the following lower bound,

$$\frac{(\kappa^*)^{\kappa^*}}{\kappa^*!} \mathop{\mathbb{E}}_{|\psi_{(\cdot,\cdot)}\rangle} \mathop{\mathbb{E}}_{\sigma\in\{\pm 1\}^{\kappa^*}} \left[\sigma_1 \cdots \sigma_{\kappa^*} \operatorname{tr}\left(H_{\kappa^*}\rho\left(1; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right)\right] \tag{6.118}$$

$$\geq \frac{1}{\sqrt{3}(\kappa^*)^2} \sum_{i\in[n], p\in\{X,Y,Z\}} \mathop{\mathbb{E}}_{|\psi_{(\cdot,\cdot)}\rangle} \left|\operatorname{tr}\left(\mathsf{pol}(H_{\kappa^*,i,p}) \bigotimes_{s\in[\kappa^*-1], j\in[n]} |\psi_{(s,j)}\rangle\!\langle\psi_{(s,j)}|\right)\right|. \tag{6.119}$$

From Corollary 9, we can further obtain

$$\frac{(\kappa^*)^{\kappa^*}}{\kappa^*!} \mathop{\mathbb{E}}_{|\psi_{(\cdot,\cdot)}\rangle} \mathop{\mathbb{E}}_{\sigma\in\{\pm 1\}^{\kappa^*}} \left[\sigma_1 \cdots \sigma_{\kappa^*} \operatorname{tr}\left(H_{\kappa^*}\rho\left(1; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right)\right] \tag{6.120}$$

$$\geq \frac{1}{\sqrt{3}(\kappa^*)^2} \sum_{i\in[n], p\in\{X,Y,Z\}} \left(\frac{1}{\sqrt{6}}\right)^{\kappa^*-1} \sqrt{\frac{\operatorname{tr}(H_{\kappa^*,i,p}^2)}{2^n(\kappa^*-1)!}}. \tag{6.121}$$

The definition of $H_{\kappa^*,i,p}$, the above inequality, and the following inequality

$$\mathop{\mathbb{E}}_{|\psi_{(\cdot,\cdot)}\rangle} \mathop{\mathbb{E}}_{\sigma\in\{\pm 1\}^{\kappa^*}} \left|\operatorname{tr}\left(H_{\kappa^*}\rho\left(1; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right)\right| \tag{6.122}$$

$$\geq \mathop{\mathbb{E}}_{|\psi_{(\cdot,\cdot)}\rangle} \mathop{\mathbb{E}}_{\sigma\in\{\pm 1\}^{\kappa^*}} \left[\sigma_1 \cdots \sigma_{\kappa^*} \operatorname{tr}\left(H_{\kappa^*}\rho\left(1; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right)\right], \tag{6.123}$$

can be used to establish the claim. □

Given the expansion property, we are going to use the following implication, which considers an arbitrary ordering π of the n qubits. The inequality allows us to control the growth for the number of Pauli observables that act on qubits before the i-th qubit under the ordering π. The precise statement is given below.

Lemma 33 (A characterization of expansion)**.** *Given an n-qubit Hamiltonian $H = \sum_P \alpha_P P$ with expansion coefficient c_e and expansion dimension d_e. Consider any permutation $\pi \in S_n$ over n qubits. For any $i \in [n]$,*

$$\sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \mathbb{1}[\alpha_P \neq 0] \mathbb{1}[P_{\pi(i)} \neq I] \mathbb{1}[P_{\pi(j)} = I, \forall j > i] \leq c_e i^{d_e - 1}, \tag{6.124}$$

Proof. Given a permutation $\pi \in S_n$ over n qubits and an $i \in [n]$. We separately consider two cases: (1) $i < d_e$ and (2) $i \geq d_e$. For the first case, let $\Upsilon = \{\pi(1), \ldots, \pi(d_e)\}$, we have

$$\sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \mathbb{1}[\alpha_P \neq 0] \mathbb{1}[P_{\pi(i)} \neq I] \mathbb{1}[P_{\pi(j)} = I, \forall j > i] \tag{6.125}$$

$$\leq \sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \mathbb{1}\Big[\alpha_P \neq 0 \text{ and } \big(\mathsf{dom}(P) \subseteq \Upsilon\big)\Big] \leq c_e. \tag{6.126}$$

The second inequality follows from the definition of the expansion coefficient c_e. For the second case, we consider all subset $\Upsilon \subseteq \pi([i]) \triangleq \{\pi(1), \pi(2), \ldots, \pi(i)\}$ with $|\Upsilon| = d_e - 1$ and $\pi(i) \in \Upsilon$,

$$\sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \mathbb{1}[\alpha_P \neq 0] \mathbb{1}[P_{\pi(i)} \neq I] \mathbb{1}[P_{\pi(j)} = I, \forall j > i] \tag{6.127}$$

$$\leq \sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \sum_{\substack{\Upsilon \subseteq \pi([i]),\\ |\Upsilon| = d_e, \pi(i) \in \Upsilon}} \mathbb{1}\Big[\alpha_P \neq 0 \text{ and } \big(\mathsf{dom}(P) \subseteq \Upsilon \text{ or } \Upsilon \subseteq \mathsf{dom}(P)\big)\Big] \tag{6.128}$$

$$\leq \sum_{\substack{\Upsilon \subseteq \pi([i]),\\ |\Upsilon| = d_e, \pi(i) \in \Upsilon}} c_e \leq c_e (i-1)^{d_e - 1} \leq c_e i^{d_e - 1}. \tag{6.129}$$

The second inequality again follows from the definition of c_e. □

Using the above implication of the expansion property, we can obtain the following inequality relating two norms. Basically, we can use the limit on the growth of the number of Pauli observables to turn the sum of ℓ_2-norm into an ℓ_r-norm, where r depends on the expansion dimension d_e.

Lemma 34 (Norm inequality using expansion property)**.** *Given an n-qubit Hamiltonian $H = \sum_P \alpha_P P$ with an expansion coefficient c_e and expansion dimension d_e. Let $r = 2d_e/(d_e+1)$. For any $\kappa^* \geq 1$, we have*

$$\sum_{i\in[n]} \left(\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_i\neq I}} \alpha_P^2 \right)^{1/2} \geq \frac{1}{c_e^{1/(2d_e)}} \left(\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*}} |\alpha_P|^r \right)^{1/r}. \tag{6.130}$$

Proof. We begin by considering a permutation π over n qubits, such that

$$\sqrt{\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_{\pi(i)}\neq I}} \alpha_P^2} \leq \sqrt{\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_{\pi(j)}\neq I}} \alpha_P^2}, \quad \forall i < j \in [n]. \tag{6.131}$$

The permutation π can be obtained by sorting the n qubits. The above ensures that for all $i \in [n]$,

$$i \sqrt{\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_{\pi(i)}\neq I}} \alpha_P^2} \leq \sum_{j\in[n]} \sqrt{\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_{\pi(j)}\neq I}} \alpha_P^2}. \tag{6.132}$$

By going through the n qubits based on the permutation π, we have the following identity,

$$\sum_{P:|P|=\kappa^*} |\alpha_P|^r = \sum_{i=1}^{n} \sum_{p\in\{X,Y,Z\}} \sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_{\pi(i)}=p}} |\alpha_P|^r \, \mathbb{1}[\alpha_P \neq 0] \mathbb{1}[P_{\pi(j)} = I, \forall j > i]. \tag{6.133}$$

Holder's inequality and $1/(d_e+1) = 1 - r/2$ allows us to obtain the following upper bound on $\sum_{P:|P|=\kappa^*} |\alpha_P|^r$,

$$\sum_{i=1}^{n} \left(\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_{\pi(i)}\neq I}} \alpha_P^2 \right)^{r/2} \times \tag{6.134}$$

$$\left(\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*}} \mathbb{1}[\alpha_P \neq 0] \mathbb{1}[P_{\pi(i)} \neq I] \mathbb{1}[P_{\pi(j)} = I, \forall j > i] \right)^{1/(d_e+1)}. \tag{6.135}$$

We can then use Lemma 33 to obtain

$$\sum_{P:|P|=\kappa^*} |\alpha_P|^r \leq \sum_{i=1}^{n} \left(\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_{\pi(i)}\neq I}} \alpha_P^2 \right)^{r/2} \left(c_e i^{d_e - 1} \right)^{1/(d_e+1)}. \tag{6.136}$$

Using $r - 1 = (d_e - 1)/(d_e + 1) \geq 0$, we have

$$\sum_{P:|P|=\kappa^*} |\alpha_P|^r \leq c_e^{1/(d_e+1)} \sum_{i=1}^{n} \left(i \sqrt{\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_{\pi(i)}\neq I}} \alpha_P^2} \right)^{r-1} \sqrt{\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_{\pi(i)}\neq I}} \alpha_P^2} \tag{6.137}$$

The choice of π ensures Eq. (6.132), which gives rise to

$$\sum_{P:|P|=\kappa^*} |\alpha_P|^r \leq c_e^{1/(d_e+1)} \left(\sum_{i\in[n]} \left(\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_i\neq I}} \alpha_P^2 \right)^{1/2} \right)^r. \tag{6.138}$$

The claim follows from $1/(r(d_e + 1)) = 1/(2d_e)$. □

Together, we can obtain the ℓ_r-norm lower bound for the expectation of the homogeneous κ^*-local Hamiltonian H_{κ^*} on the product state $\rho\left(1; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)$.

Corollary 10. *From the definitions given in Section 6.5, we have*

$$\mathop{\mathbb{E}}_{|\psi_{(\cdot,\cdot)}\rangle} \mathop{\mathbb{E}}_{\sigma\in\{\pm1\}^{\kappa^*}} \left| \mathrm{tr}\left(H_{\kappa^*} \rho\left(1; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right) \right| \tag{6.139}$$

$$\geq \frac{\sqrt{2(\kappa^*!)}}{c_e^{1/(2d_e)} (\kappa^*)^{\kappa^*+1.5} \sqrt{6}^{\kappa^*}} \left(\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*}} |\alpha_P|^r \right)^{1/r} \tag{6.140}$$

$$\geq \frac{\sqrt{2(k!)}}{c_e^{1/(2d_e)} k^{k+1.5} \sqrt{6}^{k}} \left(\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*}} |\alpha_P|^r \right)^{1/r}. \tag{6.141}$$

Proof. From Lemma 32, we have

$$\mathop{\mathbb{E}}_{|\psi_{(\cdot,\cdot)}\rangle} \mathop{\mathbb{E}}_{\sigma\in\{\pm1\}^{\kappa^*}} \left| \mathrm{tr}\left(H_{\kappa^*} \rho\left(1; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right) \right| \tag{6.142}$$

$$\geq \frac{\sqrt{2(\kappa^*!)}}{(\kappa^*)^{\kappa^*+1.5} \sqrt{6}^{\kappa^*}} \sum_{i\in[n], p\in\{X,Y,Z\}} \sqrt{\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_i=p}} \alpha_P^2}. \tag{6.143}$$

By the elementary inequality $\sqrt{x} + \sqrt{y} + \sqrt{z} \geq \sqrt{x+y+z}$ for nonnegative x, y, z,

$$\sum_{i\in[n], p\in\{X,Y,Z\}} \sqrt{\sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_i=p}} \alpha_P^2} \geq \sum_{i\in[n]} \sqrt{\sum_{p\in\{X,Y,Z\}} \sum_{\substack{P\in\{I,X,Y,Z\}^{\otimes n}:\\ |P|=\kappa^*, P_i=p}} \alpha_P^2}. \tag{6.144}$$

Combining with Lemma 34 and the fact that $k \geq \kappa^*$ yields the stated result. □

Homogeneous to inhomogeneous through polynomial optimization

We need the following basic result from real analysis:

Lemma 35 (Markov brothers' inequality, see e.g. p. 248 of (Borwein and Erdélyi, 1995))**.** *For any real polynomial $p(t) = \sum_{\kappa=1}^{k} a_\kappa x^\kappa$,*

$$|a_\kappa| \leq (1+\sqrt{2})^k \sup_{|t|\leq 1} |p(t)| \tag{6.145}$$

for all $1 \leq \kappa \leq k$.

Using the Markov brothers' inequality, we can show that performing the one-dimensional polynomial optimization over t achieves a good advantage over

$$\alpha_I = \mathop{\mathbb{E}}_{|\psi\rangle:\text{Haar}} \langle\psi| H |\psi\rangle, \tag{6.146}$$

the average energy.

Corollary 11. *From the definitions given in Section 6.5, we have*

$$\left|\operatorname{tr}\left(H\rho\left(t^*; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right) - \alpha_I\right| \geq \frac{1}{(1+\sqrt{2})^k}\left|\operatorname{tr}\left(H_{\kappa^*}\rho\left(1; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right)\right|. \tag{6.147}$$

Proof. Recall that $H = \alpha_I I + \sum_{\kappa=1}^{k} H_\kappa$ from Eq. (6.59). We can use the polarization identity given in Lemma 28 to see that the function $f(t) = \operatorname{tr}\left(H\rho\left(t; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right)$ is a polynomial,

$$\operatorname{tr}\left(H\rho\left(t; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right) = \alpha_I + \sum_{\kappa=1}^{k} \operatorname{tr}\left(H_\kappa\rho\left(1; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right) t^\kappa. \tag{6.148}$$

Recall that t^* is chosen based on the optimization

$$\max_{t\in[-1,1]} \left|\operatorname{tr}\left(H\rho\left(t; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right) - \alpha_I\right|. \tag{6.149}$$

By considering Lemma 35 with $a_\kappa = \operatorname{tr}\left(H_\kappa\rho\left(1; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right)$, we have

$$(1+\sqrt{2})^k \left|\operatorname{tr}\left(H\rho\left(t^*; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right) - \alpha_I\right| \geq \left|\operatorname{tr}\left(H_\kappa\rho\left(1; |\psi_{(\cdot,\cdot)}\rangle, \sigma\right)\right)\right|. \tag{6.150}$$

This concludes the proof of this corollary. □

6.6 Norm inequalities from approximate optimization algorithm

The approximate optimization algorithm described in the previous section is not used directly in the ML algorithm, but used to derive norm inequalities, i.e., inequalities relating different norms over Hermitian operators. An important norm that we will use in the ML algorithms is the Pauli-p norm defined below. The Pauli-p norm is equivalent to the vector-p norm on the Pauli coefficient of an observable H.

Definition 5 (Pauli-p norm)**.** *Given $H = \sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \alpha_P P$ and $p \geq 1$. The Pauli-p norm of H is*

$$\|H\|_{\text{Pauli},p} = \left(\sum_P |\alpha_P|^p \right)^{1/p}. \tag{6.151}$$

Recall that the spectral norm $\|H\| = \max_{|\psi\rangle} | \langle\psi| H |\psi\rangle | = \max_\rho |\operatorname{tr}(H\rho)|$. In this section, we will use the approximate optimization algorithm to derive several norm inequalities relating the Pauli-p norm $\|\cdot\|_{\text{Pauli},p}$ to the spectral norm $\|\cdot\|$ for common classes of observables.

We begin with a well-known fact that equates the Frobenius norm and the Pauli-2 norm. This proposition follows directly from the orthonormality of the Pauli observables $\{I, X, Y, Z\}^{\otimes n}$.

Proposition 13 (Frobenius norm)**.** *Given any n-qubit Hermitian operator H. We have*

$$\frac{1}{\sqrt{2^n}} \|H\|_F = \|H\|_{\text{Pauli},2} \leq \|H\|. \tag{6.152}$$

Proof. Let n be the number of qubits H act on and $\lambda_1, \ldots, \lambda_{2^n}$ be the eigenvalues of O. From the fact that $\operatorname{tr}(PQ) = 2^n \delta_{P=Q}$, we have

$$\|H\|_F^2 = \operatorname{tr}(H^2) = \sum_P |\alpha_P|^2 2^n = 2^n \|H\|_{\text{Pauli},2}^2. \tag{6.153}$$

Since $\|H\|_F^2 = \sum_{i=1}^{2^n} |\lambda_i|^2 \leq 2^n \max_i |\lambda_i|^2 = 2^n \|H\|_\infty^2$, we have

$$\sum_P |\alpha_P|^2 = \|H\|_F^2 / 2^n \leq \|H\|_\infty^2, \tag{6.154}$$

which establishes the claim. □

We now utilize Theorem 29 to obtain the following useful norm inequality.

Theorem 33 (Norm inequality from Theorem 29)**.** *Given an n-qubit k-local Hamiltonian H with expansion coefficient/dimension c_e, d_e. Let $r = 2d_e/(d_e+1) \in [1, 2)$. We have*

$$\frac{1}{3} C(c_e, d_e, k) \left\|H\right\|_{\text{Pauli},r} \leq \left\|H\right\|, \tag{6.155}$$

where $C(c_e, d_e, k) = \frac{\sqrt{2(k!)}}{c_e^{1/(2d_e)} k^{k+1.5+1/r} (\sqrt{6}+2\sqrt{3})^k}$ is the same as Theorem 29.

Proof. Consider the Pauli representation $H = \sum_{P:|P|\leq k} \alpha_P P$. If we consider $\rho = I/2^n$, then we have

$$\left\|H\right\| \geq |\text{tr}(H)/2^n| \geq \left| \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}} \left[\langle\phi| H |\phi\rangle \right] \right| = |\alpha_I|. \tag{6.156}$$

If we consider the random product state $|\psi\rangle$ from Theorem 29, then we have

$$\mathop{\mathbb{E}}_{|\psi\rangle} \left| \langle\psi| H |\psi\rangle - \mathop{\mathbb{E}}_{|\phi\rangle:\text{Haar}} \left[\langle\phi| H |\phi\rangle \right] \right| \geq C(c_e, d_e, k) \left(\sum_{P\neq I} |\alpha_P|^r \right)^{1/r}. \tag{6.157}$$

Using $\mathbb{E}_{|\phi\rangle:\text{Haar}} \left[\langle\phi| H |\phi\rangle \right] = \alpha_I$ and $\mathbb{E}_{|\psi\rangle} |\langle\psi| H |\psi\rangle - \alpha_I| \leq \mathbb{E}_{|\psi\rangle} |\langle\psi| H |\psi\rangle| + |\alpha_I|$, we have

$$\left\|H\right\| \geq \mathop{\mathbb{E}}_{|\psi\rangle} |\langle\psi| H |\psi\rangle| \geq C(c_e, d_e, k) \left(\sum_{P\neq I} |\alpha_P|^r \right)^{1/r} - |\alpha_I|. \tag{6.158}$$

Next, we utilize the following inequality

$$\max(x_1, cx_2 - x_1) \geq \frac{c}{c+2}(x_1 + x_2), \forall x_1, x_2, c \geq 0, \tag{6.159}$$

which can be shown by considering the two cases: $x_1 \geq (c/2)x_2$ and $x_1 < (c/2)x_2$, as well as the lower bounds on $\left\|H\right\|$ to show that

$$\begin{aligned} \left\|H\right\| &\geq \frac{C(c_e, d_e, k)}{C(c_e, d_e, k) + 2} \left(|\alpha_I| + \left(\sum_{P\neq I} |\alpha_P|^r \right)^{1/r} \right) &(6.160) \\ &\geq \frac{C(c_e, d_e, k)}{3} \left(|\alpha_I| + \left(\sum_{P\neq I} |\alpha_P|^r \right)^{1/r} \right). &(6.161) \end{aligned}$$

The second inequality uses $k, c_e, d_e \geq 1$, which implies $C(c_e, d_e, k) \in [0, 1]$. Finally, the inequality

$$|\alpha_I| + \left(\sum_{P\neq I} |\alpha_P|^r \right)^{1/r} \geq \left(\sum_{P} |\alpha_P|^r \right)^{1/r}, \tag{6.162}$$

can be used to establish the claim. □

Using Fact 2 and Fact 3 that characterize the expansion property for general k-local Hamiltonians and bounded degree k-local Hamiltonians (i.e., each qubit is acted on by at most d of the k-qubit observables), we can establish the following corollaries.

Corollary 12 (Norm inequality for k-local Hamiltonian)**.** *Given an n-qubit k-local Hamiltonian H. We have*

$$\frac{1}{3}C(k)\left\|H\right\|_{\text{Pauli},\frac{2k}{k+1}} \leq \left\|H\right\|, \tag{6.163}$$

where $C(k) = \frac{\sqrt{2(k!)}}{2k^{k+1.5+(k+1)/(2k)}(\sqrt{6}+2\sqrt{3})^k}$ is the same as Corollary 6.

Corollary 13 (Norm inequality for bounded-degree Hamiltonian)**.** *Given an n-qubit k-local Hamiltonian H with a bounded degree d. We have*

$$\frac{1}{3}C(k,d)\left\|H\right\|_{\text{Pauli},1} \leq \left\|H\right\|, \tag{6.164}$$

where $C(k,d) = \frac{\sqrt{2(k!)}}{\sqrt{d}k^{k+2.5}(2\sqrt{6}+4\sqrt{3})^k}$.

6.7 Sample-optimal algorithms for predicting bounded-degree observables

In this section, we consider one of the most basic learning problems in quantum information theory: predicting properties of an unknown n-qubit state ρ. This has been studied extensively in the literature on shadow tomography (Aaronson, 2018; Aaronson and Rothblum, 2019) and classical shadows (Huang, Richard Kueng, and Preskill, 2020).

Review of classical shadow formalism

We recall the following definition and theorem from classical shadow tomography (Huang, Richard Kueng, and Preskill, 2020) based on randomized Pauli measurements. Each randomized Pauli measurement is performed on a single copy of ρ and measures each qubit of ρ in a random Pauli basis (X, Y, or Z).

Definition 6 (Shadow norm from randomized Pauli measurements)**.** *Given an n-qubit observable O. Let $\mathcal{U}$ be the distribution over the tensor product of n single-qubit random Clifford unitary, and $\mathcal{M}_P^{-1} = \bigotimes_{i=1}^n \mathcal{M}_1^{-1}$ with $\mathcal{M}_1^{-1}(A) = 3A - \mathrm{tr}(A)I$. The shadow norm of O is defined as*

$$\left\|O\right\|_{\text{shadow}} = \max_{\sigma:\text{state}}\left(\mathop{\mathbb{E}}_{U\sim\mathcal{U}}\sum_{b\in\{0,1\}^n}\langle b|\,U\sigma U^\dagger\,|b\rangle\,\langle b|\,U\mathcal{M}_P^{-1}(O)U^\dagger\,|b\rangle^2\right)^{1/2}. \tag{6.165}$$

Theorem 34 (Classical shadow tomography using randomized Pauli measurements (Huang, Richard Kueng, and Preskill, 2020))**.** *Given an unknown n-qubit state ρ and M observables $O_1, \ldots, O_M$ with $B_{\text{shadow}} = \max_{i \in [M]} \|O_i\|_{\text{shadow}}$. After N randomized Pauli measurements on copies of ρ satisfying*

$$N = \mathcal{O}\left(\frac{\log(M) B_{\text{shadow}}^2}{\epsilon^2}\right), \tag{6.166}$$

we can estimate $\mathrm{tr}(O_i \rho)$ *to ϵ error for all $i \in [M]$ with high probability.*

We can see that the sample complexity for predicting many properties of an unknown quantum state ρ depends on the shadow norm $\|\cdot\|_{\text{shadow}}$. The larger $\|\cdot\|_{\text{shadow}}$ is, the more experiments is needed to estimate properties of ρ accurately. From the original classical shadow paper (Huang, Richard Kueng, and Preskill, 2020), we can obtain the following shadow norm bounds for Pauli observables and for few-body observables.

Lemma 36 (Shadow norm for Pauli observables (Huang, Richard Kueng, and Preskill, 2020))**.** *For any $P \in \{I, X, Y, Z\}^{\otimes n}$, we have*

$$\|P\|_{\text{shadow}} = 3^{|P|/2}. \tag{6.167}$$

Lemma 37 (Shadow norm for few-body observables (Huang, Richard Kueng, and Preskill, 2020))**.** *For any observable O that acts nontrivially on at most k qubits, we have*

$$\|O\|_{\text{shadow}} \leq 2^k \|O\|. \tag{6.168}$$

Combining the above lemmas and Theorem 34, we can see that Pauli observables and few-body observables can both be predicted efficiently under very few number of randomized Pauli measurements.

Upper bound for predicting bounded-degree observables

Consider an n-qubit observable O given as a sum of k-qubit observables $O = \sum_j O_j$, where each qubit is acted on by at most d of these k-qubit observables ${}_{O}j$. We focus on $k = \mathcal{O}(1)$ and $d = \mathcal{O}(1)$, and refer to such an observable as a bounded-degree observable. These bounded-degree observables arise frequently in quantum many-body physics and quantum information. For example, the Hamiltonian in a quantum spin system can often be described by a geometrically-local Hamiltonian, which is an instance of bounded-degree observables. For these observables, the shadow norm is related to the Pauli-1 norm of the observable,

$$\|O\|_{\mathrm{shadow}} \leq \sum_{P:|P|\leq k} |\alpha_P| \, \|P\|_{\mathrm{shadow}} \leq 3^{k/2} \sum_{P:|P|\leq k} |\alpha_P| = 3^{k/2} \|O\|_{\mathrm{pauli},1} \,. \tag{6.169}$$

If we consider the norm inequality between ℓ_1-norm and ℓ_2-norm and use the standard result relating Frobenius norm and spectral norm (Proposition 13), we would obtain the following upper bound on shadow norm.

$$\|O\|_{\mathrm{shadow}} \leq 3^{k/2} \|O\|_{\mathrm{pauli},1} \leq (2\sqrt{3})^k \sqrt{nd} \, \|O\|_{\mathrm{pauli},2} = \mathcal{O}\left(\sqrt{n} \, \|O\|\right). \tag{6.170}$$

Using Theorem 34, this shadow norm bound would give rise to a number of measurements scaling as

$$N = \mathcal{O}\left(\frac{n \log(M) B_\infty^2}{\epsilon^2}\right), \tag{6.171}$$

where $B_\infty = \max_{i\in[M]} \|O_i\|_\infty$ is an upper bound on the spectral norm $\|\cdot\|$. Due to the linear dependence on the number n of qubits in the unknown quantum state, this scaling is not ideal. Furthermore, we will later show that this scaling is actually far from optimal.

To improve the sample complexity, we will use the improved approximate optimization algorithm presented in Section 6.5, and the corresponding norm inequality presented in Section 6.6. Using the norm inequality relating Pauli-1 norm and the spectral norm (Corollary 13), we can obtain the following shadow norm bound.

Lemma 38 (Shadow norm for bounded-degree observables)**.** *Given $k, d = \mathcal{O}(1)$ and an n-qubit observable O that is a sum of k-qubit observables, where each qubit is acted on by at most d of these k-qubit observables.*

$$\|O\|_{\mathrm{shadow}} \leq C \|O\|, \tag{6.172}$$

for some constant $C > 0$.

Combining the above lemma with Theorem 34 allows us to establish the following theorem. Compared to Eq. (6.171), the following theorem uses n times fewer measurements.

Theorem 35 (Classical shadow for bounded-degree observables)**.** *Given an unknown n-qubit state ρ and M observables $O_1, \ldots, O_M$ with $B_\infty = \max_i \|O_i\|_\infty$. Suppose each observable O_i is a sum of few-body observables $O_i = \sum_j O_{ij}$, where* *every qubit is acted on by a constant number of the few-body observables O_{ij}. After N randomized Pauli measurements on copies of ρ with*

$$N = \mathcal{O}\left(\frac{\log\left(\min(M, n)\right) B_\infty^2}{\epsilon^2}\right), \tag{6.173}$$

we can estimate $\mathrm{tr}(O_i\rho)$ *to ϵ error for all $i \in [M]$ with high probability.*

Proof. The upper bound of $N = \mathcal{O}\left(\log(M) \max_{i\in[M]} \left\|O_i\right\|_\infty^2 / \epsilon^2\right)$ follows immediately from Theorem 34 and Lemma 38. We can also establish an upper bound of $N = \mathcal{O}\left(\log(n) \max_{i\in[M]} \left\|O_i\right\|_\infty^2 / \epsilon^2\right)$. To see this, consider the task of predicting all k-qubit Pauli observables $P \in \{I, X, Y, Z\}^{\otimes n}$ with $|P| \leq k$. There are at most $\mathcal{O}(n^k)$ such Pauli observables. To predict all of the k-qubit Pauli observables to ϵ' error under the unknown state ρ, we can combine Theorem 34 and Lemma 36 to see that we only need

$$N = \mathcal{O}\left(\log(n) \max_{i\in[M]} \left\|O_i\right\|_\infty^2 / (\epsilon')^2\right) \tag{6.174}$$

randomized Pauli measurements. Now, given any observable $O_i = \sum_P \alpha_P P$ that is a sum of few-body observables $O_i = \sum_j O_{ij}$, where every qubit is acted on by a constant number of the few-body observables O_{ij}, we can predict $\mathrm{tr}(O_i\rho)$ using the following identity

$$\mathrm{tr}(O_i\rho) = \sum_{P:|P|\leq k} \alpha_P \,\mathrm{tr}(P\rho), \tag{6.175}$$

which incurs a prediction error of at most $\sum_P |\alpha_P| \epsilon'$. Using the norm inequality in Corollary 13, we have

$$\left\|O_i\right\|_{\text{Pauli},1} = \sum_P |\alpha_P| \leq C \left\|O_i\right\|, \tag{6.176}$$

for a constant C. Hence, by setting $\epsilon' = \epsilon/C$, we can predict O_i to ϵ error. Thus we can also establish an upper bound of $N = \mathcal{O}\left(\log(n) \max_{i\in[M]} \left\|O_i\right\|_\infty^2 / \epsilon^2\right)$. The claim follows by considering the corresponding prediction algorithm (use the standard classical shadow when $M < n$, and use the above algorithm when $M \geq n$). □

Optimality of Theorem 35

Here we prove the following lower bound on the sample complexity of shadow tomography for bounded-degree observables demonstrating that Theorem 35 is optimal. The optimality holds even when we considered collective measurement procedure on many copies of ρ. This is in stark contrast to other sets of observables, such as the collection of high-weight Pauli observables, where single-copy measurements (e.g., classical shadow tomography) require exponentially more copies than collective measurements.

Theorem 36 (Lower bound for predicting bounded-degree observables)**.** *Consider the following task. Given any unknown n-qubit state ρ and any M observables $O_1, \ldots, O_M$ with $B_\infty = \max_i \|O_i\|$. Each observable O_i is a sum of few-body observables $O_i = \sum_j O_{ij}$, where every qubit is acted on by a constant number of the few-body observables O_{ij}. We would like to estimate $\mathrm{tr}(O_i\rho)$ to ϵ error for all $i \in [M]$ with high probability by performing arbitrary collective measurements on N copies of ρ. The number of copies needs to be at least*

$$N = \Omega\left(\frac{\log\left(\min(M,n)\right)B_\infty^2}{\epsilon^2}\right), \tag{6.177}$$

for any algorithm to succeed in this task.

To show Theorem 36, we show a lower bound for the following *distinguishing task*, from which the lower bound for shadow tomography will follow readily. Given $i \in [n]$, let P_i denote the n-body Pauli operator that acts as Z on the i-th qubit and trivially elsewhere, and define the mixed state

$$\rho^i \triangleq \frac{1}{2^n}\left(I + \frac{\epsilon}{B_\infty} \cdot P_i\right). \tag{6.178}$$

We will show a lower bound for distinguishing whether ρ is maximally mixed or of the form ρ^i for some i.

Lemma 39 (Lower bound for a distinguishing task)**.** *Let $0 \le \epsilon \le 1$ and $\delta \ge 2\epsilon$. Let $\mathcal{A}$ be an algorithm that, given access to N copies of a mixed state ρ which is either the maximally mixed state or ρ^i for some $i \in [\min(M,n)]$, correctly determines whether or not ρ is maximally mixed with probability at least $3/4$. Then $N = \Omega(\log(\min(M,n))B_\infty^2/\epsilon^2)$.*

Proof of Theorem 36. Let $\mathcal{A}$ be an algorithm that solves the task in Theorem 36 to error $\epsilon/3$. We can use this to give an algorithm for the task in Lemma 39: applying $\mathcal{A}$ to the following $\min(M,n)$ observables,

$$O_1 \triangleq B_\infty P_1, \quad \ldots, \quad O_{\min(M,n)} \triangleq B_\infty P_{\min(M,n)}, \tag{6.179}$$

we can produce $\epsilon/3$-accurate estimates for $\mathrm{tr}(\rho P_j)$ for all $j \in [\min(M,n)]$. Note that if ρ is maximally mixed, $\mathrm{tr}(\rho O_j) = 0$ for all j, whereas if $\rho = \rho^i$, then $\mathrm{tr}(\rho O_j) = \epsilon \cdot \mathbb{1}[i = j]$. In particular, by checking whether there is a j for which $\mathrm{tr}(\rho P_j) > 2\epsilon/3$, we can determine whether ρ is maximally mixed or equal to some ρ^i. The lower bound in Lemma 39 thus implies the lower bound in Theorem 36. □

For convenience, define $n' \triangleq \min(M,n)$. Note that for any $i \in [n]$, $(\rho^i)^{\otimes N}$ is diagonal, so we can assume without loss of generality that $\mathcal{A}$ simply makes N independent measurements in the computational basis. Proving Lemma 39 thus amounts to showing a lower bound for a classical distribution testing task.

Note that the distribution π^i over outcomes of a single measurement of ρ^i in the computational basis places

$$\frac{1 + (-1)^{x_i}\epsilon}{2^n} \tag{6.180}$$

mass on each string $x \in \{0,1\}^n$. The distribution π over outcomes of a single measurement of the maximally mixed state in the computational basis is uniform over all strings $x \in \{0,1\}^n$. The following basic result in binary hypothesis testing lets us reduce proving Lemma 39 to upper bounding

$$d_{\mathrm{TV}}\left(\mathop{\mathbb{E}}_i[(\pi^i)^{\otimes N}], \pi^{\otimes N}\right). \tag{6.181}$$

Lemma 40 (Le Cam's two-point method (LeCam, 1973))**.** *Let p_0, p_1 be distributions over a domain Ω for which there exists a distribution D such that $d_{\mathrm{TV}}(p_0, p_1) < 1/3$. Then there is no algorithm $\mathcal{A}$ that maps elements of Ω to $\{0,1\}$ for which $\Pr_{x\sim p_i}[\mathcal{A}(x) = i] \geq 2/3$ for both $i = 0, 1$.*

Proof of Lemma 39. To bound the expression in Eq. (6.181), it suffices to bound the chi-squared divergence $\chi^2(\mathbb{E}_i[(\pi^i)^{\otimes N}]\|\pi^{\otimes N})$ because for any distributions p, q, we have $d_{\mathrm{TV}}(p,q) \leq 2\sqrt{\chi^2(p\|q)}$. For convenience, let us define the likelihood ratio perturbation

$$\eta^i(x) \triangleq \frac{\mathrm{d}\pi^i}{\mathrm{d}\pi}(x) - 1 = (-1)^{x_i}\epsilon \tag{6.182}$$

and observe that for any $i, j \in [n]$,

$$\mathop{\mathbb{E}}_{x\sim\pi}[\eta^i(x) \cdot \eta^j(x)] = \epsilon^2 \cdot \mathbb{1}[i = j]. \tag{6.183}$$

Also given strings $x^1, \ldots, x^N \in \{0,1\}^n$ and $S \subseteq [N]$, denote

$$\eta^i(x^S) \triangleq \prod_{j\in S} \eta^i(x_j). \tag{6.184}$$

We then have the standard calculation, see e.g. (Wu, 2017, Lemma 22.1):

$$1+\chi^2\left(\mathop{\mathbb{E}}_{i\sim[n']}[(\pi^i)^{\otimes N}]\|\pi^{\otimes N}\right) = \mathop{\mathbb{E}}_{x^1,\ldots,x^N\sim\pi^{\otimes N}}\left[\mathop{\mathbb{E}}_{i\sim[n']}\left[\prod_{j=1}^{N}(1+\eta^{i,t}(x^j))\right]^2\right] \tag{6.185}$$

$$= \mathop{\mathbb{E}}_{i,i'\sim[n']}\left[\mathop{\mathbb{E}}_{x^1,\ldots,x^N\sim\pi^{\otimes N}}\left[\sum_{S,T\subseteq[N]}\eta^i(x^S)\eta^{i'}(x^T)\right]\right] \tag{6.186}$$

$$= \mathop{\mathbb{E}}_{i,i'\sim[n']}\left[\mathop{\mathbb{E}}_{x^1,\ldots,x^N\sim\pi^{\otimes N}}\left[\sum_{S\subseteq[N]}\eta^i(x^S)\eta^{i'}(x^S)\right]\right] \tag{6.187}$$

$$= \mathop{\mathbb{E}}_{i,i'\sim[n']}\left[\mathop{\mathbb{E}}_{x^1,\ldots,x^N\sim\pi^{\otimes N}}\left[\prod_{j=1}^{N}(1+\eta^i(x^j)\eta^{i'}(x^j))\right]\right] \tag{6.188}$$

$$= \mathop{\mathbb{E}}_{i,i'\sim[n']}\left[(1+\mathop{\mathbb{E}}_{x\sim\pi}[\eta^i(x)\eta^{i'}(x)])^N\right] \tag{6.189}$$

$$= \frac{1}{n'}(1+\epsilon^2)^N + \frac{n'-1}{n'} \tag{6.190}$$

We conclude that

$$\chi^2(\mathop{\mathbb{E}}_{i\sim[n']}[(\pi^i)^{\otimes N}]\|\pi^{\otimes N}) \le \frac{1}{n'}((1+\epsilon^2)^N-1), \tag{6.191}$$

so for $N = c\log(n')/\epsilon^2$ for sufficiently small constant $c > 0$, this quantity is less than $1/3$. By applying Lemma 40 to $p_0 = \pi^{\otimes N}$ and $p_1 = \mathbb{E}_{i\sim[n']}[(\pi^i)^{\otimes N}]$, we obtain the claimed lower bound. □

6.8 Learning to predict an unknown observable

We begin with a definition of invariance for distribution over quantum states.

Definition 7 (Invariance under a unitary). *A probability distribution $\mathcal{D}$ over quantum states is invariant under a unitary U if the probability density remains unchanged after the action of U, i.e.,*

$$f_{\mathcal{D}}(\rho) = f_{\mathcal{D}}(U\rho U^\dagger) \tag{6.192}$$

for any state ρ.

In this section, we will utilize the norm inequalities in Section 6.6 to give a learning algorithm that achieves the following guarantee. The learning algorithm can learn any unknown n-qubit observable $O^{(\mathrm{unk})}$ even if the scale $\left\|O^{(\mathrm{unk})}\right\|$ is unknown. The mean squared error $\mathbb{E}_{\rho\sim\mathcal{D}}\left|h(\rho) - \mathrm{tr}\left(O^{(\mathrm{unk})}\rho\right)\right|^2$ scales quadratically with the scale of the unknown observable $O^{(\mathrm{unk})}$. We can see that the sample complexity N has a quasi-polynomial dependence on the error ϵ, ϵ' relative to the scale of the unknown observable $O^{(\mathrm{unk})}$, and only depends on the system size n and the failure probability δ logarithmically.

Theorem 37 (Learning to predict an unknown observable)**.** *Given $n, \epsilon, \epsilon', \delta > 0$. Consider any unknown n-qubit observable $O^{(\mathrm{unk})} = \sum_P \alpha_P P$ and any unknown n-qubit state distribution $\mathcal{D}$ that is invariant under single-qubit H and S gates. Given training data $\{\rho_\ell, \mathrm{tr}(O^{(\mathrm{unk})}\rho_\ell))\}_{\ell=1}^N$ of size*

$$N = \log\left(\frac{n}{\delta}\right)\, \min\left(2^{\mathcal{O}\left(\log(\frac{1}{\epsilon})\left(\log\log(\frac{1}{\epsilon}) + \log(\frac{1}{\epsilon'})\right)\right)}, 2^{\mathcal{O}(\log(\frac{1}{\epsilon})\log(n))}\right). \tag{6.193}$$

Let $k = \lceil \log_{1.5}(1/\epsilon)\rceil$, $O^{(\mathrm{low})} = \sum_{|P|\leq k}\alpha_P P$ be the low-degree approximation of $O^{(\mathrm{unk})}$, and $r = \frac{2k}{k+1} \in [1, 2)$. The algorithm can learn a function $h(\rho) =$ $\max(-\hat{\Theta}, \min(\hat{\Theta}, \mathrm{tr}(\hat{O}\rho)))$ *for an observable $\hat{O}$ and a real number $\hat{\Theta}$ that achieves a prediction error*

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}}\left|h(\rho) - \mathrm{tr}\left(O^{(\mathrm{unk})}\rho\right)\right|^2 \leq \left(\epsilon + \epsilon'\left[1 + \left(\frac{\|O^{(\mathrm{low})}\|}{\|O^{(\mathrm{unk})}\|}\right)^r\right]\right)\left\|O^{(\mathrm{unk})}\right\|^2 \tag{6.194}$$

with probability at least $1 - \delta$.

Low-degree approximation under mean squared error

In order to characterize the mean squared error $\mathbb{E}_{\rho\sim\mathcal{D}}\,\mathrm{tr}(O_1\rho) - \mathrm{tr}(O_2)\rho$ between two observables O_1, O_2, we need the following definition of a modified purity for quantum states.

Definition 8 (Non-identity purity)**.** *Given a k-qubit state ρ. The non-identity purity of ρ is*

$$\gamma^\star(\rho) \triangleq \frac{1}{2^k}\sum_{Q\in\{X,Y,Z\}^{\otimes k}} \mathrm{tr}(Q\rho)^2. \tag{6.195}$$

Non-identity purity is bounded by purity,

$$\gamma^\star(\rho) \leq \gamma(\rho) = \mathrm{tr}(\rho^2) = \frac{1}{2^k}\sum_{Q\in\{I,X,Y,Z\}^{\otimes k}} \mathrm{tr}(Q\rho)^2. \tag{6.196}$$

Lemma 41 (Mean squared error)**.** *Given two n-qubit observables O_1, O_2 with*

$$O_1 - O_2 = \sum_{P\in\{I,X,Y,Z\}^{\otimes n}} \Delta\alpha_P P, \tag{6.197}$$

and a distribution $\mathcal{D}$ over quantum states that is invariant under single-qubit H and S gates. We have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} |\mathrm{tr}(O_1\rho) - \mathrm{tr}(O_2\rho)|^2 = \sum_{P\in\{I,X,Y,Z\}^{\otimes n}} \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \left[\gamma^\star\left(\rho_{\mathsf{dom}(P)}\right)\right] \left(\frac{2}{3}\right)^{|P|} |\Delta\alpha_P|^2 . \tag{6.198}$$

Proof. Consider $U_1, \ldots, U_n$ to be independent random single-qubit Clifford unitaries. Because $\mathcal{D}$ is invariant under single-qubit Hadamard and phase gates, $\mathcal{D}$ is invariant under any tensor product of single-qubit Clifford unitaries. This implies that the distribution of the random state ρ is the same as the distribution of the random state $(U_1 \otimes \ldots \otimes U_n)\rho(U_1 \otimes \ldots \otimes U_n)^\dagger$. Using this fact, we expand the mean squared error as

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} |\mathrm{tr}(O_1\rho) - \mathrm{tr}(O_2\rho)|^2 \tag{6.199}$$

$$= \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \mathop{\mathbb{E}}_{U_1,\ldots,U_n} \sum_{P,Q} \Delta\alpha_P\Delta\alpha_Q \,\mathrm{tr}\left(\left(\bigotimes_{i=1}^n U_i^\dagger P_i U_i\right) \otimes \left(\bigotimes_{i=1}^n U_i^\dagger Q_i U_i\right)(\rho\otimes\rho)\right). \tag{6.200}$$

Using the unitary 2-design property of random Clifford unitary and $\mathrm{SWAP} = \frac{1}{2}\sum_{P\in\{I,X,Y,Z\}} P \otimes P$, we have

$$\mathop{\mathbb{E}}_{U_i}\left[U_i^\dagger P_i U_i \otimes U_i^\dagger Q_i U_i\right] = \begin{cases} I \otimes I, & P_i = Q_i = I, \\ \frac{1}{3}\left(X\otimes X + Y \otimes Y + Z \otimes Z\right), & P_i = Q_i \neq I, \\ 0, & P_i \neq Q_i. \end{cases} \tag{6.201}$$

We can now write the target value as

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} |\mathrm{tr}(O_1\rho) - \mathrm{tr}(O_2\rho)|^2 \tag{6.202}$$

$$= \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \sum_{P\in\{I,X,Y,Z\}^{\otimes n}} \frac{1}{3^{|P|}} |\Delta\alpha_P|^2 \sum_{Q\in\{X,Y,Z\}^{\otimes|P|}} \mathrm{tr}(Q\rho_{\mathsf{dom}(P)})^2. \tag{6.203}$$

The claim follows from Definition 8 on non-identity purity $\gamma^\star$. □

The following lemma tells us that the mean absolute error can be upper bounded by the root mean squared error. Hence, both the mean absolute error and the mean squared error are characterized by the ℓ_2 distance between the Pauli coefficients (as well as the average non-identity purity). Due to the following relation, we will focus on the mean squared error throughout the text.

Lemma 42 (Mean absolute error)**.** *Given two n-qubit observables O_1, O_2 with*

$$O_1 - O_2 = \sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \Delta\alpha_P P, \tag{6.204}$$

and a distribution $\mathcal{D}$ over quantum states that is invariant under single-qubit H and S gates. We have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} |\mathrm{tr}(O_1\rho) - \mathrm{tr}(O_2\rho)| \leq \left(\sum_{P\in\{I,X,Y,Z\}^{\otimes n}} \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \left[\gamma^\star\left(\rho_{\mathsf{dom}(P)}\right)\right] \left(\frac{2}{3}\right)^{|P|} |\Delta\alpha_P|^2 \right)^{1/2}. \tag{6.205}$$

Proof. Jensen's inequality gives

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} |\mathrm{tr}(O_1\rho) - \mathrm{tr}(O_2\rho)| \leq \left(\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} |\mathrm{tr}(O_1\rho) - \mathrm{tr}(O_2\rho)|^2\right)^{1/2}. \tag{6.206}$$

Combining with Lemma 41 yields the stated result. □

From Lemma 41, we can construct a low-degree approximation by removing all high-weight Pauli terms for any observable O. The approximation error decays exponentially with the weight of the Pauli terms.

Corollary 14 (Low-degree approximation)**.** *Given an n-qubit observable $O = \sum_{P\in\{I,X,Y,Z\}^{\otimes n}} \alpha_P P$ and a distribution $\mathcal{D}$ over quantum states that is invariant under single-qubit H and S gates. For $k > 0$, consider $O^{(k)} = \sum_{P:|P|<k} \alpha_P P$. We have*

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \left|\mathrm{tr}(O\rho) - \mathrm{tr}(O^{(k)}\rho)\right|^2 \leq \left(\frac{2}{3}\right)^k \|O\|^2. \tag{6.207}$$

Proof. Using Lemma 41 and the fact that $\gamma^\star(\varrho) \leq \gamma(\varrho) \leq 1$ for any state ϱ, we have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \left|\mathrm{tr}(O\rho) - \mathrm{tr}(O^{(k)}\rho)\right|^2 \leq \sum_{P:|P|\geq k} \left(\frac{2}{3}\right)^{|P|} |\alpha_P|^2 \leq \left(\frac{2}{3}\right)^k \sum_P |\alpha_P|^2. \tag{6.208}$$

The norm inequality given in Prop. 13 establishes the claim. □

Tools for extracting and filtering Pauli coefficients

In order to learn the low-degree approximation of an arbitrary observable O, we need to be able to extract the relevant α_P. Furthermore, we will impose criteria for filtering out uninfluential Pauli observables P to prevent them from increasing the noise and leading to a higher prediction error.

Extracting Pauli coefficient

Lemma 43 (Extracting Pauli coefficient)**.** *Given an n-qubit observable*

$$O = \sum_{P \in \{I,X,Y,Z\}^{\otimes n}} \alpha_P P \tag{6.209}$$

and a distribution $\mathcal{D}$ over quantum states that is invariant under single-qubit H and S gates. For any Pauli observable $P \in \{I, X, Y, Z\}^{\otimes n}$, we have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \operatorname{tr}(O\rho)\operatorname{tr}(P\rho) = \left(\frac{2}{3}\right)^{|P|} \alpha_P \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \gamma^*(\rho_{\mathsf{dom}(P)}). \tag{6.210}$$

Proof. Using the invariance of $\mathcal{D}$, we have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \operatorname{tr}(O\rho)\operatorname{tr}(P\rho) \tag{6.211}$$

$$= \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \mathop{\mathbb{E}}_{U_1,\ldots,U_n} \sum_{Q\in\{I,X,Y,Z\}^{\otimes n}} \alpha_Q \operatorname{tr}\left(\left(\left(\bigotimes_{i=1}^{n} U_i^\dagger P_i U_i\right) \otimes \left(\bigotimes_{i=1}^{n} U_i^\dagger Q_i U_i\right) (\rho\otimes\rho)\right)\right). \tag{6.212}$$

Using Eq. (6.201), we can rewrite the above expression as

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \operatorname{tr}(O\rho)\operatorname{tr}(P\rho) = \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \frac{1}{3^{|P|}}\alpha_P \sum_{Q\in\{X,Y,Z\}^{\otimes|P|}} \operatorname{tr}(Q\rho_{\mathsf{dom}(P)})^2. \tag{6.213}$$

The claim follows from the definition of the non-identity purity γ^*. □

For each Pauli observable $P \in \{I, X, Y, Z\}^{\otimes n}$, define the quantity we can extract using the lemma to be

$$x_P = \left(\frac{2}{3}\right)^{|P|} \alpha_P \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \gamma^*(\rho_{\mathsf{dom}(P)}). \tag{6.214}$$

We can obtain an estimate $\hat{x}_P$ for x_P by averaging $\operatorname{tr}(O\rho)\operatorname{tr}(P\rho)$ over the training data. However, to obtain an estimate $\hat{\alpha}_P$ for α_P, we need to divide $\hat{x}$ by $\left(\frac{2}{3}\right)^{|P|} \mathbb{E}_{\rho\sim\mathcal{D}} \gamma^*(\rho_{\mathsf{dom}(P)})$. The error in the estimate $\hat{\alpha}_P$ could be arbitrarily large if $\left(\frac{2}{3}\right)^{|P|} \mathbb{E}_{\rho\sim\mathcal{D}} \gamma^*(\rho_{\mathsf{dom}(P)})$ is close to zero. Hence, we present a filter in Section 6.8 to handle this issue. In addition to this filter, the norm inequalities given in Section 6.6 show that most α_P would be close to zero. Hence, when α_P is small, we could simply set them to zero to avoid noise build-up. This gives rise to the second filtering layer given in Section 6.8.

Filtering small weight factor

The first filter sets the estimate $\hat{\alpha}_P$ to be zero when the average non-identity purity $\mathbb{E}_{\rho\sim\mathcal{D}}\, \gamma^*(\rho_{\mathsf{dom}(P)})$ is close to zero. We define the weight factor for a Pauli observable P to be

$$\beta_P = \left(\frac{2}{3}\right)^{|P|} \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \gamma^*(\rho_{\mathsf{dom}(P)}). \tag{6.215}$$

The weight factor β_P depends on the distribution $\mathcal{D}$, which may be unknown. Hence, we can only obtain an estimate $\hat{\beta}_P$ for β_P by utilizing the training data. Recall from Lemma 43, we can only obtain an estimate $\hat{x}_P$ for $x_P = \alpha_P\beta_P$. The mean squared error (Lemma 41) shows that the contribution from error in $\hat{\alpha}_P$ is

$$\beta_P \left|\hat{\alpha}_P - \alpha_P\right|^2. \tag{6.216}$$

The presence of β_P in the mean squared error is very useful since it counteracts the fact that we cannot estimate $\hat{\alpha}_P$ accurately when β_P is close to zero. The following lemma shows that estimates for β_P and x_P are sufficient to perform filtering and achieve a small mean squared error.

Lemma 44 (Filtering small weight factor)**.** *Given $\tilde{\epsilon}, \eta > 0$. Consider $\alpha \in [-\eta, \eta]$, and $\beta \in [0, 1]$. Let $x = \alpha\beta \in [-\eta, \eta]$. Given estimates $\hat{x}$ and $\hat{\beta}$ with $|\hat{x} - x| < \eta\tilde{\epsilon}$ and $|\hat{\beta} - \beta| < \tilde{\epsilon}$. If we define the estimate*

$$\hat{\alpha} = \begin{cases} 0, & \hat{\beta} \leq 2\tilde{\epsilon}, \\ \hat{x}/\hat{\beta}, & \hat{\beta} > 2\tilde{\epsilon}, \end{cases} \tag{6.217}$$

then we have $\beta|\hat{\alpha} - \alpha|^2 \leq 3\eta^2\tilde{\epsilon}$.

Proof. Consider the first case of $\hat{\beta} \leq 2\tilde{\epsilon}$. We have

$$\beta|\hat{\alpha} - \alpha|^2 = \beta\alpha^2 \leq \eta^2\beta \leq \eta^2\hat{\beta} + \eta^2\tilde{\epsilon} \leq 3\eta^2\tilde{\epsilon}. \tag{6.218}$$

For the second case of $\hat{\beta} > 2\tilde{\epsilon}$, we have $\beta > \tilde{\epsilon}$. By applying triangle inequality, we have

$$\left|\sqrt{\beta}\hat{\alpha} - \sqrt{\beta}\alpha\right| \leq \frac{\sqrt{\beta}}{\hat{\beta}}\left|\hat{x} - x\right| + \left|\sqrt{\beta}x\right|\left|\frac{1}{\hat{\beta}} - \frac{1}{\beta}\right|. \tag{6.219}$$

The first term can be bounded as $\frac{\sqrt{\beta}}{\hat{\beta}}\left|\hat{x} - x\right| \leq \eta\frac{\sqrt{\beta}}{\hat{\beta}}\tilde{\epsilon}$. The second term can be bounded by the same expression

$$\left|\sqrt{\beta}x\right|\left|\frac{1}{\hat{\beta}} - \frac{1}{\beta}\right| = \beta^{3/2}|\alpha|\frac{|\hat{\beta} - \beta|}{\hat{\beta}\beta} \leq \eta\frac{\sqrt{\beta}}{\hat{\beta}}\tilde{\epsilon}. \tag{6.220}$$

Using the fact that $\sqrt{z+\tilde{\epsilon}}/z$ is monotonically decreasing for $z > 0$, we have

$$\frac{\sqrt{\beta}}{\hat{\beta}}\tilde{\epsilon} \leq \frac{\sqrt{\hat{\beta}+\tilde{\epsilon}}}{\hat{\beta}}\tilde{\epsilon} \leq \sqrt{\frac{3}{4}\tilde{\epsilon}}. \tag{6.221}$$

Together, $\left|\sqrt{\beta}\hat{\alpha} - \sqrt{\beta}\alpha\right|^2 \leq 3\eta^2\tilde{\epsilon}$ and the claim is established. □

Filtering uninfluential Pauli observables

Consider a set $S \subseteq \{I, X, Y, Z\}^{\otimes n}$ that contains the Pauli observables of interest. For example, we will later consider S to be the set of all few-body Pauli observables. Using the norm inequalities given in Section 6.6, we can filter out more α_P to achieve an improved mean squared error. Below is the filtering lemma that combines both the filtering of Pauli observables with a small weight factor (Lemma 44) and the filtering of those with a small contribution (characterized by $|x_P|/\beta_P^{1/2}$).

Lemma 45 (Filtering lemma)**.** *Given $\tilde{\epsilon}, \eta > 0$, and a set $S \subseteq \{I, X, Y, Z\}^{\otimes n}$. Consider $\alpha_P \in [-\eta, \eta]$, $\beta_P \in [0, 1]$, $x_P = \alpha_P\beta_P \in [-\eta, \eta]$ for all $P \in S$. Suppose there exists $A > 0$ and $1 \leq r < 2$, such that*

$$\sum_{P \in S} |\alpha_P|^r \leq A^r. \tag{6.222}$$

Given $\hat{x}_P$ and $\hat{\beta}_P$ with $|\hat{x}_P - x_P| < \eta\tilde{\epsilon}$ and $|\hat{\beta}_P - \beta_P| < \tilde{\epsilon}$ for all $P \in S$. If we define

$$\hat{\alpha}_P = \begin{cases} 0, & \hat{\beta}_P \leq 2\tilde{\epsilon}, \\ 0, & \hat{\beta}_P > 2\tilde{\epsilon},\ |\hat{x}_P|/\hat{\beta}_P^{1/2} \leq 2\eta\sqrt{\tilde{\epsilon}}, \\ \hat{x}_P/\hat{\beta}_P, & \hat{\beta}_P > 2\tilde{\epsilon},\ |\hat{x}_P|/\hat{\beta}_P^{1/2} > 2\eta\sqrt{\tilde{\epsilon}}, \end{cases} \tag{6.223}$$

then we have $\sum_{P \in S} \beta_P |\hat{\alpha}_P - \alpha_P|^2 \leq 6A^r\eta^{2-r}\tilde{\epsilon}^{1-(r/2)}$. We also have $\beta_P|\hat{\alpha}_P - \alpha_P|^2 \leq 9\eta^2\tilde{\epsilon}, \forall P \in S$.

Proof. We first define $S^u \subseteq S$ to be the set of Pauli observables P with $\hat{\beta}_P > 2\tilde{\epsilon}, |\hat{x}_P|/\hat{\beta}_P^{1/2} > 2\eta\sqrt{\tilde{\epsilon}}$. The set S^u contains all the unfiltered Pauli observables. We define S^f to be $S \setminus S^u$, which contains all the filtered Pauli observables. We separate the contribution of S^u and S^f in the mean squared error $\sum_{P \in S} \beta_P |\hat{\alpha}_P - \alpha_P|^2$,

$$\sum_{P \in S} \beta_P |\hat{\alpha}_P - \alpha_P|^2 = \sum_{P \in S^u} \beta_P |\hat{\alpha}_P - \alpha_P|^2 + \sum_{P \in S^f} \beta_P |\hat{\alpha}_P - \alpha_P|^2. \tag{6.224}$$

A key quantity for the analysis is $\beta_P^{1/2}\alpha_P = x_P/\beta_P^{1/2}$. For Pauli P with $\hat{\beta}_P \leq 2\tilde{\epsilon}$, we have

$$|\beta_P^{1/2}\alpha_P| \leq \eta\sqrt{\hat{\beta}_P + \tilde{\epsilon}} \leq \eta\sqrt{3\tilde{\epsilon}}. \tag{6.225}$$

For Pauli P with $\hat{\beta}_P > 2\tilde{\epsilon}$, we have

$$\left|\frac{\hat{x}_P}{\hat{\beta}_P^{1/2}} - \frac{x_P}{\beta_P^{1/2}}\right| \leq \frac{1}{\hat{\beta}_P^{1/2}}|\hat{x}_P - x_P| + |x_P|\left|\frac{1}{\hat{\beta}_P^{1/2}} - \frac{1}{\beta_P^{1/2}}\right| \leq \eta\sqrt{\frac{\tilde{\epsilon}}{2}} + \eta\left|\frac{\beta_P}{\hat{\beta}_P^{1/2}} - \beta_P^{1/2}\right| \leq \eta\sqrt{\tilde{\epsilon}}. \tag{6.226}$$

The last inequality uses the fact that $\beta_P > \tilde{\epsilon}$, $\hat{\beta}_P/\beta_P > 2$, and hence

$$\left|\frac{\beta_P}{\hat{\beta}_P^{1/2}} - \beta_P^{1/2}\right| = \frac{\left|\hat{\beta}_P - \beta_P\right|}{\hat{\beta}_P^{1/2}\left(1 + \left(\frac{\hat{\beta}_P}{\beta_P}\right)^{1/2}\right)} \leq \frac{\sqrt{\tilde{\epsilon}}}{2+\sqrt{2}}. \tag{6.227}$$

We are now ready to analyze the contributions of S^u and S^f.

For the unfiltered Pauli observables (those in set S^u), we can use Lemma 44 to obtain

$$\sum_{P\in S^u} \beta_P|\hat{\alpha}_P - \alpha_P|^2 \leq 3\eta^2\tilde{\epsilon}|S^u|. \tag{6.228}$$

Eq. (6.226) shows that for Pauli observable P with $\hat{\beta}_P > 2\tilde{\epsilon}$ and $|\hat{x}_P|/\hat{\beta}_P^{1/2} > 2\eta\sqrt{\tilde{\epsilon}}$, we have $|x_P|/\beta_P^{1/2} > 2\eta\sqrt{\tilde{\epsilon}} - \eta\sqrt{\tilde{\epsilon}}$. We will use this fact to bound the size of the set $|S^u|$,

$$|S^u| \leq \sum_{P\in S^u} \frac{(|x_P|/\beta_P^{1/2})^r}{\left(2\eta\sqrt{\tilde{\epsilon}} - \eta\sqrt{\tilde{\epsilon}}\right)^r} = \frac{1}{\eta^r\tilde{\epsilon}^{r/2}}\sum_{P\in S^u}|\alpha_P|^r\beta_P^{r/2} \leq \frac{1}{\eta^r\tilde{\epsilon}^{r/2}}\sum_{P\in S}|\alpha_P|^r = \frac{A^r}{\eta^r\tilde{\epsilon}^{r/2}}. \tag{6.229}$$

Together, we have the following upper bound,

$$\sum_{P\in S^u} \beta_P|\hat{\alpha}_P - \alpha_P|^2 \leq 3\eta^{2-r}A^r\tilde{\epsilon}^{1-(r/2)}. \tag{6.230}$$

For the filtered Pauli observables (those in set S^f), we have

$$\sum_{P\in S^f} \beta_P|\hat{\alpha}_P - \alpha_P|^2 = \sum_{P\in S^f}\left|\beta_P^{1/2}\alpha_P\right|^r\left|\beta_P^{1/2}\alpha_P\right|^{2-r}. \tag{6.231}$$

There are two types of Pauli observables in S^f.

1. For P with $\hat{\beta}_P \leq 2\tilde{\epsilon}$, we have $\left|\beta_P^{1/2}\alpha_P\right| \leq \eta\sqrt{3\tilde{\epsilon}}$ from Eq. (6.225).

2. For P with $\hat{\beta}_P > 2\tilde{\epsilon}$ and $|\hat{x}_P|/\hat{\beta}_P^{1/2} \leq \eta 2\sqrt{\tilde{\epsilon}}$, we have $\left|\beta_P^{1/2}\alpha_P\right| = |x_P|/\beta_P^{1/2} \leq 2\eta\sqrt{\tilde{\epsilon}} + \eta\sqrt{\tilde{\epsilon}}$ from Eq. (6.226).

Together, we have the following upper bound,

$$\sum_{P \in S^f} \beta_P |\hat{\alpha}_P - \alpha_P|^2 \leq (3\eta\sqrt{\tilde{\epsilon}})^{2-r} \sum_{P \in S^f} \beta_P^{r/2} |\alpha_P|^r \leq A^r (3\eta\sqrt{\tilde{\epsilon}})^{2-r} \leq 3A^r \eta^{2-r} \tilde{\epsilon}^{1-(r/2)}. \tag{6.232}$$

Combining the contribution of S^u and S^f yields

$$\sum_{P \in S} \beta_P |\hat{\alpha}_P - \alpha_P|^2 \leq 6A^r \eta^{2-r} \tilde{\epsilon}^{1-(r/2)}. \tag{6.233}$$

Thus we have established the first statement of the lemma.

We now focus on the second statement of the lemma. For Pauli observable P that satisfies the first and the third cases of Eq. (6.223), we can use Lemma 44 to obtain $\beta_P |\hat{\alpha}_P - \alpha_P|^2 \leq 3\eta^2\tilde{\epsilon} < 9\eta^2\tilde{\epsilon}$. For the second case of Eq. (6.223), we can use Eq. (6.226) to see that

$$\beta_P |\hat{\alpha}_P - \alpha_P|^2 = \left(\frac{x_P}{\beta_P^{1/2}}\right)^2 \leq \left(\frac{|\hat{x}_P|}{\hat{\beta}_P^{1/2}} + \left|\frac{\hat{x}_P}{\hat{\beta}_P^{1/2}} - \frac{x_P}{\beta_P^{1/2}}\right|\right)^2 \leq 9\eta^2\tilde{\epsilon}. \tag{6.234}$$

Hence, for all $P \in S$, we have $\beta_P |\hat{\alpha}_P - \alpha_P|^2 \leq 9\eta^2\tilde{\epsilon}$. □

Learning algorithm

In this section, we present a learning algorithm satisfying the guarantee given in Theorem 37. Consider the full training data $\{\rho_\ell, y_\ell = \mathrm{tr}(O^{(\mathrm{unk})}\rho_\ell)\}_{\ell=1}^N$ of size N. The learning algorithm splits the full data into a smaller training set of size N_{tr} and a validation set of size N_{val} with $N = N_{\mathrm{tr}} + N_{\mathrm{val}}$. The training set is used to extract Pauli coefficients and perform filtering with a hyperparameter η. The validation set is used to choose the best hyperparameter η. We can set $N_{\mathrm{tr}} = (4/5)N$ and $N_{\mathrm{val}} = (1/5)N$.

We consider two slightly different learning algorithms for the sample complexity scaling of

$$N = \log\left(\frac{n}{\delta}\right) 2^{\mathcal{O}\left(\log(\frac{1}{\epsilon})\left(\log\log(\frac{1}{\epsilon}) + \log(\frac{1}{\epsilon'})\right)\right)} \quad \text{and} \quad N = \log\left(\frac{n}{\delta}\right) 2^{\mathcal{O}(\log(\frac{1}{\epsilon})\log(n))}. \tag{6.235}$$

We can simply look at which sample complexity is smaller and select the corresponding learning algorithm.

We begin with the learning algorithm for achieve the sample complexity on the left of Eq. (6.235). First, the algorithm computes the sample maximum over the training set,

$$\hat{\Theta} = \max_{\ell \in \{1,\ldots,N_{\mathrm{tr}}\}} |y_\ell| = \max_{\ell \in \{1,\ldots,N_{\mathrm{tr}}\}} \left|\mathrm{tr}(O^{(\mathrm{unk})}\rho_\ell))\right| \leq \left\|O^{(\mathrm{unk})}\right\|. \tag{6.236}$$

to obtain a scale for the function value. Let $C(k)$ be the constant from Corollary 12. We define

$$\tilde{\epsilon} \triangleq \left(\frac{\epsilon'}{12}\right)^{k+1} \left(\frac{C(k)}{3}\right)^{2k}. \tag{6.237}$$

Next, we consider the following grid of hyperparameters,

$$\eta \in \left\{2^0\hat{\Theta}, 2^1\hat{\Theta}, 2^2\hat{\Theta} \ldots, 2^R\hat{\Theta}\right\}, \tag{6.238}$$

where $R = \log_2\lceil 1/\tilde{\epsilon}\rceil$. For each hyperparameter η, the learning algorithm runs the following. The learning algorithm considers every Pauli observable $P \in \{I, X, Y, Z\}^{\otimes n}$ with $|P| \leq \log_{1.5}(1/\epsilon)$. We define the set that contains the Pauli observables of interest,

$$S = \left\{P : |P| \leq \log_{1.5}(1/\epsilon)\right\}, \tag{6.239}$$

and $k = \lceil\log_{1.5}(1/\epsilon)\rceil$. For each $P \in S$, the algorithm computes

$$\hat{x}_P = \frac{1}{N_{\mathrm{tr}}} \sum_{\ell=1}^{N_{\mathrm{tr}}} \mathrm{tr}(P\rho_\ell) y_\ell, \tag{6.240}$$

$$\hat{\beta}_P = \frac{1}{N_{\mathrm{tr}}} \sum_{\ell=1}^{N_{\mathrm{tr}}} \mathrm{tr}(P\rho_\ell)\,\mathrm{tr}(P\rho_\ell), \tag{6.241}$$

using the training set $\{(\rho_\ell, y_\ell = \mathrm{tr}(O^{\mathrm{unk}\rho_\ell}))\}_{\ell=1}^{N_{\mathrm{tr}}}$. By definition of $\hat{x}_P$ and $\hat{\Theta}$, we have

$$|\hat{x}_P| \leq \hat{\Theta}, \quad \forall P \in S. \tag{6.242}$$

Then, for each $P \in S$, the algorithm computes

$$\hat{\alpha}_P(\eta) = \begin{cases} 0, & \hat{\beta}_P \leq 2\tilde{\epsilon}, \\ 0, & \hat{\beta}_P > 2\tilde{\epsilon},\ |\hat{x}_P|/\hat{\beta}_P^{1/2} \leq 2\eta\sqrt{\tilde{\epsilon}}, \\ \hat{x}_P/\hat{\beta}_P, & \hat{\beta}_P > 2\tilde{\epsilon},\ |\hat{x}_P|/\hat{\beta}_P^{1/2} > 2\eta\sqrt{\tilde{\epsilon}}, \end{cases} \tag{6.243}$$

The algorithm considers the function $h(\rho; \eta) = \max(-\hat{\Theta}, \min(\hat{\Theta}, \mathrm{tr}(\hat{O}(\eta)\rho)))$, where the observable $\hat{O}(\eta)$ is defined as follows,

$$\hat{O}(\eta) = \sum_{P \in S} \hat{\alpha}_P(\eta) P. \tag{6.244}$$

The best η is selected using the validation set,

$$\eta^* = \underset{\eta \in \{2^0\hat{\Theta}, \ldots, 2^R\hat{\Theta}\}}{\arg\min} \frac{1}{N_{\text{val}}} \sum_{\ell=N_{\text{tr}}+1}^{N_{\text{tr}}+N_{\text{val}}} |h(\rho_\ell; \eta) - y_\ell|^2 . \tag{6.245}$$

The learning algorithm outputs $h(\rho; \eta^*)$ as the learned function.

We now present the learning algorithm for achieving the sample complexity on the right of Eq. (6.235). We define the set that contains the Pauli observables of interest,

$$S' = \left\{ P : |P| \leq \log_{1.5}(2/\epsilon) \right\}, \tag{6.246}$$

and $k' = \lceil \log_{1.5}(2/\epsilon) \rceil$. For each $P \in S'$, the algorithm computes

$$\hat{x}'_P = \frac{1}{N} \sum_{\ell=1}^{N} \mathrm{tr}(P\rho_\ell) y_\ell, \tag{6.247}$$

$$\hat{\beta}'_P = \frac{1}{N} \sum_{\ell=1}^{N} \mathrm{tr}(P\rho_\ell)\, \mathrm{tr}(P\rho_\ell), \tag{6.248}$$

using the full dataset $\{(\rho_\ell, y_\ell = \mathrm{tr}(O^{\mathrm{unk}\rho_\ell}))\}_{\ell=1}^N$. The algorithm uses the following hyperparameter

$$\tilde{\epsilon}' \triangleq \frac{\epsilon}{6n^{k'}}. \tag{6.249}$$

Then, for each $P \in S'$, the algorithm computes

$$\hat{\alpha}'_P = \begin{cases} 0, & \hat{\beta}'_P \leq 2\tilde{\epsilon}', \\ \hat{x}'_P / \hat{\beta}'_P, & \hat{\beta}'_P > 2\tilde{\epsilon}'. \end{cases} \tag{6.250}$$

The algorithm outputs the function $h'(\rho) = \mathrm{tr}(\hat{O}'\rho)$, where the observable $\hat{O}'$ is defined as $\hat{O}' = \sum_{P \in S'} \hat{\alpha}'_P P$.

Here, we assume that $\mathrm{tr}(P\rho_\ell)$ can be obtained from the training data. However, for each $\mathrm{tr}(P\rho_\ell)$, we only need to be able to obtain an unbiased estimator for $\mathrm{tr}(P\rho_\ell)$ and for $\mathrm{tr}(P\rho_\ell)^2$. Recall that an unbiased estimator for a is a random variable with expectation value equal to a. For example, an unbiased estimator for $\mathrm{tr}(P\rho_\ell)^2$ can be obtained by performing two quantum measurements on two individual copies of ρ_ℓ using the observable P and multiplying the results, or by utilizing classical shadow formalism (Huang, Richard Kueng, and Preskill, 2020) and randomized measurement (Andreas Elben, Steven T Flammia, et al., 2022).

Rigorous performance guarantee

In this section, we prove that the learning algorithm presented in the last section satisfies Theorem 37. We separate the proof for achieving the sample complexity on the left and right of Eq. (6.235).

The proof for the sample complexity stated on the left of Eq. (6.235) consists of three parts: (1) a characterization of the prediction error, (2) the existence of a good hyperparamter $\eta^{\triangle}$ that achieves a small prediction error, (3) the hyperparameter η^* found through grid search on the validation set has a small prediction error.

The proof for the sample complexity stated on the right of Eq. (6.235) is simpler and is given at the end.

Characterization of the prediction error

We begin with a lemma about the sample maximum.

Lemma 46 (Sample maximum)**.** *Given* $1 > \epsilon, \delta > 0$. *Consider an arbitrary real-valued random variable* X. *Let* $X_1, \ldots, X_N$ *be* N *independent samples of* X *with* $N = \lceil \log(1/\delta)/\epsilon \rceil$ *and let* $\hat{\Theta} = \max_i X_i$. *Then*

$$\Pr\left[X \leq \hat{\Theta}\right] \geq 1 - \epsilon. \tag{6.251}$$

with probability at least $1 - \delta$.

Proof. Recall that the cumulative distribution function is defined as

$$F(\theta) = \Pr\left[X \leq \theta\right]. \tag{6.252}$$

We define the approximate maximum as follows,

$$\Theta \triangleq \inf_{\theta : F(\theta) \geq 1 - \epsilon} \theta. \tag{6.253}$$

Using the right-continuity of $F(\theta) = \Pr\left[X \leq \theta\right]$, we have

$$F(\Theta) = \Pr\left[X \leq \Theta\right] \geq 1 - \epsilon. \tag{6.254}$$

Furthermore, from the definition of Θ, we have

$$\Pr\left[X \geq \Theta\right] \geq \epsilon. \tag{6.255}$$

To see the above inequality, suppose that $\Pr\left[X \geq \Theta\right] < \epsilon$. Then from the left-continuity of $F'(\theta) = \Pr\left[X \geq \theta\right]$, we can find $\Theta' < \Theta$, such that $\Pr\left[X \geq \Theta'\right] \leq \epsilon$.

Thus, there exists $\Theta' < \Theta$ with $\Pr\left[X \leq \Theta'\right] \geq 1 - \epsilon$, which is a contradiction to the definition of Θ. Together, we have

$$\Pr\left[X_i < \Theta, \forall i \in [N]\right] \leq (1 - \epsilon)^N. \tag{6.256}$$

By choosing $N = \lceil \log(1/\delta)/\epsilon \rceil$, we have

$$\Pr\left[\max_i X_i \geq \Theta\right] \geq 1 - (1 - \epsilon)^{\log(1/\delta)/\epsilon} \geq 1 - \delta. \tag{6.257}$$

Thus with probability at least $1 - \delta$, we have $\hat{\Theta} \geq \Theta$. Using the monotonicity of $F(\theta)$, we have

$$\Pr\left[X \leq \hat{\Theta}\right] = F(\hat{\Theta}) \geq F(\Theta) \geq 1 - \epsilon, \tag{6.258}$$

which establishes this lemma. □

Using the above lemma, we can show that given a training set of size

$$N_{\mathrm{tr}} \geq \frac{12 \log(3/\delta)}{\epsilon'}, \tag{6.259}$$

the real value $\hat{\Theta} \leq \left\|O^{(\mathrm{unk})}\right\|$ obtained by the algorithm satisfies

$$\Pr_{\rho \sim \mathcal{D}}\left[\left|\mathrm{tr}(O^{(\mathrm{unk})}\rho)\right| \leq \hat{\Theta}\right] \geq 1 - \frac{\epsilon'}{12} \tag{6.260}$$

with probability at least $1 - (\delta/3)$. Hence, with probability at least $1 - (\delta/3)$, we have

$$\mathop{\mathbb{E}}_{\rho \sim \mathcal{D}}\left|h(\rho;\eta) - \mathrm{tr}(O^{(\mathrm{unk})}\rho)\right|^2 \leq \mathop{\mathbb{E}}_{\rho \sim \mathcal{D}}\left|\mathrm{tr}(\hat{O}(\eta)\rho) - \mathrm{tr}(O^{(\mathrm{unk})}\rho)\right|^2 + \frac{\epsilon'}{12}\left|\hat{\Theta} + \left\|O^{(\mathrm{unk})}\right\|\right|^2. \tag{6.261}$$

Using Lemma 41 on mean squared error and Corollary 14 on low-degree approximation, we have

$$\mathop{\mathbb{E}}_{\rho \sim \mathcal{D}}\left|h(\rho;\eta) - \mathrm{tr}(O^{(\mathrm{unk})}\rho)\right|^2 \tag{6.262}$$

$$\leq \underbrace{(2/3)^k \left\|O^{(\mathrm{unk})}\right\|^2}_{\leq \left\|O^{(\mathrm{unk})}\right\|^2 \epsilon} + \sum_{P \in S} \mathop{\mathbb{E}}_{\rho \sim \mathcal{D}}\left[\gamma^*(\rho_{\mathsf{dom}(P)})\right]\left(\frac{2}{3}\right)^{|P|} \left|\hat{\alpha}_P(\eta) - \alpha_P\right|^2 + \frac{\epsilon'}{3}\left\|O^{(\mathrm{unk})}\right\|^2 \tag{6.263}$$

with probability at least $1 - (\delta/3)$.

Let us define the following variables,

$$x_P \triangleq \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}}\left[\gamma^*(\rho_{\mathsf{dom}(P)})\right]\left(\frac{2}{3}\right)^{|P|}\alpha_P, \quad \beta_P \triangleq \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}}\left[\gamma^*(\rho_{\mathsf{dom}(P)})\right]\left(\frac{2}{3}\right)^{|P|}, \quad \forall P \in S. \tag{6.264}$$

Then, with probability at least $1-(\delta/3)$ over the sampling of the training set, we have the following characterization of the prediction error for all $\eta > 0$,

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}}\left|h(\rho;\eta)-\mathrm{tr}(O^{(\mathrm{unk})}\rho)\right|^2 \leq \epsilon\left\|O^{(\mathrm{unk})}\right\|^2+\frac{\epsilon'}{3}\left\|O^{(\mathrm{unk})}\right\|^2+\sum_{P\in S}\beta_P\left|\hat{\alpha}_P(\eta)-\alpha_P\right|^2. \tag{6.265}$$

We will utilize this form to show the existence of a good hyperparameter $\eta^{\triangle}$.

Existence of a good hyperparamter $\eta^{\triangle}$

By considering the training set size to be

$$N_{\mathrm{tr}} = \Omega\left(\frac{\log(1/\delta)}{\epsilon'}+\frac{\log(|S|/\delta)}{\tilde{\epsilon}^2}\right), \tag{6.266}$$

we can guarantee Eq. (6.265) with probability at least $1-(\delta/3)$. Furthermore, utilizing Hoeffding's inequality and union bound, we could also guarantee that

$$|\hat{x}_P - x_P| \leq \left\|O^{(\mathrm{unk})}\right\|\tilde{\epsilon}, \quad \left|\hat{\beta}_P-\beta_P\right| \leq \tilde{\epsilon}, \quad \forall P\in S \tag{6.267}$$

with probability at least $1-(\delta/3)$. The norm inequality given in Corollary 12 shows that

$$\sum_{P\in S}|\alpha_P|^r \leq \left(\frac{3}{C(k)}\right)^r\left\|O^{(\mathrm{low})}\right\|^r \tag{6.268}$$

for a constant given by

$$C(k) = \frac{\sqrt{2(k!)}}{2k^{k+1.5+(k+1)/(2k)}(\sqrt{6}+2\sqrt{3})^k}. \tag{6.269}$$

We now condition on the event that Eq. (6.265) and Eq. (6.267) both hold, which happens with probability at least $1-(2/3)\delta$. We are now ready to define the good hyperparameter $\eta^{\triangle}$.

Let hyperparameter $\eta^{\triangle}$ belonging to the grid in Eq. (6.238) be defined as follows,

$$\eta^{\triangle} = 2^{\min\left(R,\left\lceil\log_2\left(\|O^{(\mathrm{unk})}\|/\hat{\Theta}\right)\right\rceil\right)}\hat{\Theta}. \tag{6.270}$$

We separately consider two cases: (1) $\eta^{\triangle} = 2^R\hat{\Theta}$, (2) $\eta^{\triangle} < 2^R\hat{\Theta}$. For the first case $\eta^{\triangle} = 2^R\hat{\Theta}$, we can use $|\hat{x}_P| \leq \hat{\Theta}$ in Eq. (6.242) and the definition of R to see that

$$\hat{\alpha}_P(\eta^{\triangle}) = 0, \quad \forall P\in S. \tag{6.271}$$

Since $\eta^{\triangle} = 2^R \hat{\Theta}$, we have $R \leq \left\lceil \log_2 \left(\left\| O^{(\mathrm{unk})} \right\| / \hat{\Theta} \right) \right\rceil$. This yields $\eta^{\triangle} \leq 2 \left\| O^{(\mathrm{unk})} \right\|$, which implies that

$$\hat{\alpha}_P \left(2 \left\| O^{(\mathrm{unk})} \right\| \right) = 0, \quad \forall P \in S. \tag{6.272}$$

Hence, the reconstructed Pauli coefficients $\hat{\alpha}_P(\cdot)$ are the same for $\eta^{\triangle}$ and $2 \left\| O^{(\mathrm{unk})} \right\|$. The filtering lemma given in Lemma 45 shows that

$$\sum_{P \in S} \mathop{\mathbb{E}}_{\rho \sim \mathcal{D}} \left[\gamma^*(\rho_{\mathsf{dom}(P)}) \right] \left(\frac{2}{3} \right)^{|P|} \left| \hat{\alpha}_P(\eta^{\triangle}) - \alpha_P \right|^2 \tag{6.273}$$

$$= \sum_{P \in S} \mathop{\mathbb{E}}_{\rho \sim \mathcal{D}} \left[\gamma^*(\rho_{\mathsf{dom}(P)}) \right] \left(\frac{2}{3} \right)^{|P|} \left| \hat{\alpha}_P \left(2 \left\| O^{(\mathrm{unk})} \right\| \right) - \alpha_P \right|^2 \tag{6.274}$$

$$\leq 12 \left(\frac{3}{C(k)} \right)^r \left\| O^{(\mathrm{unk})} \right\|^{2-r} \left\| O^{(\mathrm{low})} \right\|^r \tilde{\epsilon}^{1-(r/2)}. \tag{6.275}$$

For the second case $\eta^{\triangle} < 2^R \hat{\Theta}$, we have the following bound on $\eta^{\triangle}$,

$$\eta^{\triangle} = 2^{\lceil \log_2 (\| O^{(\mathrm{unk})} \| / \hat{\Theta}) \rceil} \hat{\Theta} \in \left[\left\| O^{(\mathrm{unk})} \right\|, 2 \left\| O^{(\mathrm{unk})} \right\| \right]. \tag{6.276}$$

The filtering lemma given in Lemma 45 shows that

$$\sum_{P \in S} \mathop{\mathbb{E}}_{\rho \sim \mathcal{D}} \left[\gamma^*(\rho_{\mathsf{dom}(P)}) \right] \left(\frac{2}{3} \right)^{|P|} \left| \hat{\alpha}_P(\eta^{\triangle}) - \alpha_P \right|^2 \tag{6.277}$$

$$\leq 6 (\eta^{\triangle})^r \left(\frac{3}{C(k)} \right)^r \left\| O^{(\mathrm{low})} \right\|^r \tilde{\epsilon}^{1-(r/2)} \tag{6.278}$$

$$\leq 12 \left(\frac{3}{C(k)} \right)^r \left\| O^{(\mathrm{unk})} \right\|^{2-r} \left\| O^{(\mathrm{low})} \right\|^r \tilde{\epsilon}^{1-(r/2)}. \tag{6.279}$$

In both case (1) and case (2), using the definition $r = 2k/(k+1)$ and $\tilde{\epsilon} = \left(\frac{\epsilon'}{12} \right)^{k+1} \left(\frac{C(k)}{3} \right)^{2k}$, we have

$$\sum_{P \in S} \mathop{\mathbb{E}}_{\rho \sim \mathcal{D}} \left[\gamma^*(\rho_{\mathsf{dom}(P)}) \right] \left(\frac{2}{3} \right)^{|P|} \left| \hat{\alpha}_P(\eta^{\triangle}) - \alpha_P \right|^2 \leq \epsilon' \left\| O^{(\mathrm{unk})} \right\|^{2-r} \left\| O^{(\mathrm{low})} \right\|^r. \tag{6.280}$$

Combining with Eq. (6.265), we have

$$\mathop{\mathbb{E}}_{\rho \sim \mathcal{D}} \left| h(\rho; \eta^{\triangle}) - \mathrm{tr}(O^{(\mathrm{unk})} \rho) \right|^2 \leq \epsilon \left\| O^{(\mathrm{unk})} \right\|^2 + \frac{\epsilon'}{3} \left\| O^{(\mathrm{unk})} \right\|^2 + \epsilon' \left\| O^{(\mathrm{unk})} \right\|^{2-r} \left\| O^{(\mathrm{low})} \right\|^r \tag{6.281}$$

with probability at least $1 - (2/3)\delta$.

The prediction performance of the hyperparameter η^*

From the definition of $h(\rho;\eta)$, for any quantum state ρ, we have

$$\left|h(\rho;\eta) - \mathrm{tr}(O^{(\mathrm{unk})}\rho)\right|^2 \leq \left|\hat{\Theta} + \left\|O^{(\mathrm{unk})}\right\|\right|^2 \leq 4\left\|O^{(\mathrm{unk})}\right\|^2 \tag{6.282}$$

Using Hoeffding's inequality and union bound, we can show that given a validation set of size

$$N_{\mathrm{val}} = \Omega\left(\frac{\log(R/\delta)}{(\epsilon')^2}\right), \tag{6.283}$$

with probability at least $1 - (\delta/3)$, we have

$$\left|\frac{1}{N_{\mathrm{val}}}\sum_{\ell=N_{\mathrm{tr}}+1}^{N_{\mathrm{tr}}+N_{\mathrm{val}}}\left|h(\rho_\ell;\eta) - \mathrm{tr}(O^{(\mathrm{unk})}\rho_\ell)\right|^2 - \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}}\left|h(\rho;\eta) - \mathrm{tr}(O^{(\mathrm{unk})}\rho)\right|^2\right| \tag{6.284}$$

$$\leq \left\|O^{(\mathrm{unk})}\right\|^2\frac{\epsilon'}{3}, \tag{6.285}$$

for all $\eta \in \{2^0\hat{\Theta},\ldots,2^R\hat{\Theta}\}$. Using the definition of η^* and $\eta^\triangle$, we have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}}\left|h(\rho;\eta^*) - \mathrm{tr}(O^{(\mathrm{unk})}\rho)\right|^2 \tag{6.286}$$

$$\leq \frac{1}{N_{\mathrm{val}}}\sum_{\ell=N_{\mathrm{tr}}+1}^{N_{\mathrm{tr}}+N_{\mathrm{val}}}\left|h(\rho_\ell;\eta^*) - \mathrm{tr}(O^{(\mathrm{unk})}\rho_\ell)\right|^2 + \left\|O^{(\mathrm{unk})}\right\|^2\frac{\epsilon'}{3} \tag{6.287}$$

$$\leq \frac{1}{N_{\mathrm{val}}}\sum_{\ell=N_{\mathrm{tr}}+1}^{N_{\mathrm{tr}}+N_{\mathrm{val}}}\left|h(\rho_\ell;\eta^\triangle) - \mathrm{tr}(O^{(\mathrm{unk})}\rho_\ell)\right|^2 + \left\|O^{(\mathrm{unk})}\right\|^2\frac{\epsilon'}{3} \tag{6.288}$$

$$\leq \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}}\left|h(\rho;\eta^\triangle) - \mathrm{tr}(O^{(\mathrm{unk})}\rho)\right|^2 + \left\|O^{(\mathrm{unk})}\right\|^2\frac{2\epsilon'}{3} \tag{6.289}$$

with probability at least $1-(\delta/3)$ over the sampling of the validation set. Combining with Eq. (6.281) and employing union bound, we have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}}\left|h(\rho;\eta^*) - \mathrm{tr}(O^{(\mathrm{unk})}\rho)\right|^2 \tag{6.290}$$

$$\leq \epsilon\left\|O^{(\mathrm{unk})}\right\|^2 + \epsilon'\left\|O^{(\mathrm{unk})}\right\|^2 + \epsilon'\left\|O^{(\mathrm{unk})}\right\|^{2-r}\left\|O^{(\mathrm{low})}\right\|^r \tag{6.291}$$

with probability at least $1 - \delta$, as claimed in Eq. (6.194).

Finally, by noting that $|S| = O(n^k)$ and $k = \log_{1.5}(1/\epsilon)$ and recalling the definition of $\tilde{\epsilon}$ in Eq .(6.237) on the right-hand side of Eq. (6.266), we have

$$\frac{\log(1/\delta)}{\epsilon'} + \frac{\log(|S|/\delta)}{\tilde{\epsilon}^2} = \log\left(\frac{n}{\delta}\right)\left(\frac{1}{\epsilon'}\right)^{k+1}2^{O(k\log k)} \tag{6.292}$$

$$= \log\left(\frac{n}{\delta}\right) 2^{\mathcal{O}\left(\log(\frac{1}{\epsilon})\left(\log\log(\frac{1}{\epsilon})+\log(\frac{1}{\epsilon'})\right)\right)}. \tag{6.293}$$

So it suffices to have

$$N_{\text{val}} = \log\left(\frac{n}{\delta}\right) 2^{\Omega\left(\log(\frac{1}{\epsilon})\left(\log\log(\frac{1}{\epsilon})+\log(\frac{1}{\epsilon'})\right)\right)}. \tag{6.294}$$

Furthermore, by noting that $R = \log_2\lceil 1/\tilde{\epsilon}\rceil = \mathcal{O}(k\log(\epsilon') + k\log^2 k)$ in Eq. (6.283), we see that it suffices to have

$$N_{\text{val}} = \Omega\left(\frac{\log\log(\epsilon) + \log\log(\epsilon') + \log(1/\delta)}{(\epsilon')^2}\right). \tag{6.295}$$

Recall that the full data size $N = N_{\text{tr}} + N_{\text{val}}$, and the quantity in Eq. (6.295) is dominated by the one in Eq. (6.294), yielding one argument in the minimum of the sample complexity claimed in Theorem 37.

Establishing sample complexity on the right of Eq. (6.235)

By considering the full dataset size to be

$$N = \Omega\left(\frac{\log(|S'|/\delta)}{(\tilde{\epsilon}')^2}\right), \tag{6.296}$$

Hoeffding's inequality and union bound can be used to guarantee that

$$\left|\hat{x}'_P - x_P\right| \leq \left\|O^{(\text{unk})}\right\|\tilde{\epsilon}', \quad \left|\hat{\beta}'_P - \beta_P\right| \leq \tilde{\epsilon}', \quad \forall P \in S' \tag{6.297}$$

with probability at least $1-\delta$. Using Lemma 44 on filtering small-weight factor, we have

$$\beta_P\left|\hat{\alpha}'_P - \alpha_P\right|^2 \leq 3\left\|O^{(\text{unk})}\right\|^2\tilde{\epsilon}'. \tag{6.298}$$

Using Lemma 41 on mean squared error and Corollary 14 on low-degree approximation, we have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}}\left|\operatorname{tr}(\hat{O}'\rho) - \operatorname{tr}(O^{(\text{unk})}\rho)\right|^2 \leq (2/3)^k\left\|O^{(\text{unk})}\right\|^2 + \sum_{P\in S'}\beta_P\left|\hat{\alpha}'_P - \alpha_P\right|^2 \tag{6.299}$$

$$\leq \left\|O^{(\text{unk})}\right\|^2\frac{\epsilon}{2} + 3n^{k'}\left\|O^{(\text{unk})}\right\|^2\tilde{\epsilon}'. \tag{6.300}$$

From the definition of $\tilde{\epsilon}'$ in Eq. (6.249), we have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}}\left|\operatorname{tr}(\hat{O}'\rho) - \operatorname{tr}(O^{(\text{unk})}\rho)\right|^2 \leq \epsilon\left\|O^{(\text{unk})}\right\|^2. \tag{6.301}$$

The sample complexity is

$$N = \mathcal{O}\left(\frac{\log(|S'|/\delta)}{(\tilde{\epsilon}')^2}\right) = \log(n/\delta)\, 2^{\mathcal{O}(\log(1/\epsilon)\log(n))}, \tag{6.302}$$

which completes the sample complexity claimed in Theorem 37.

6.9 Learning quantum evolutions from randomized experiments

We recall the following definitions pertaining to classical shadows for quantum states and quantum evolutions, based on randomized Pauli measurements and random input states.

Definition 9 (Single-qubit stabilizer state)**.** *We define*

$$\mathrm{stab}_1 \triangleq \{|0\rangle, |1\rangle, |+\rangle, |-\rangle, |y+\rangle, |y-\rangle\} \tag{6.303}$$

to be the set of single-qubit stabilizer states.

We define randomized Pauli measurements as follows.

Definition 10 (Randomized Pauli measurement)**.** *Given $n > 0$. A randomized Pauli measurement on an n-qubit state is given by a 6^n-outcome POVM*

$$\mathcal{F}^{(\mathrm{Pauli})} \triangleq \left\{ \frac{1}{3^n} \bigotimes_{i=1}^{n} |s_i\rangle\!\langle s_i| \right\}_{s_1,\ldots,s_n \in \mathrm{stab}_1}, \tag{6.304}$$

which corresponds to measuring every qubit under a random Pauli basis (X, Y, Z). The outcome of $\mathcal{F}^{(\mathrm{Pauli})}$ is an n-qubit state $|\psi\rangle = \bigotimes_{i=1}^{n} |s_i\rangle$, where $|s_i\rangle \in \mathrm{stab}_1$ is a single-qubit stabilizer state.

In the following, we define the classical shadow of a quantum state based on randomized Pauli measurements. Classical shadows could also be defined based on other randomized measurements (Huang, Richard Kueng, and Preskill, 2020).

Definition 11 (Classical shadow of a quantum state)**.** *Given $n, N > 0$. Consider an n-qubit state ρ. A size-N classical shadow $S_N(\rho)$ of quantum state ρ is a random set given by*

$$S_N(\rho) \triangleq \{|\psi_\ell\rangle\}_{\ell=1}^{N}, \tag{6.305}$$

where $|\psi_\ell\rangle = \bigotimes_{i=1}^{n} |s_{\ell,i}\rangle$ is the outcome of the ℓ-th randomized Pauli measurement on a single copy of ρ.

We can generalize classical shadows from quantum states to quantum processes by considering random product input states and randomized Pauli measurements. A similar generalization has been studied in (Levy, Luo, and Clark, 2021).

Definition 12 (Classical shadow of a quantum process)**.** *Given an n-qubit CPTP map $\mathcal{E}$. A size-N classical shadow $S_N(\mathcal{E})$ of quantum evolution $\mathcal{E}$ is a random set given by*

$$S_N(\mathcal{E}) \triangleq \left\{ |\psi_\ell^{(\mathrm{in})}\rangle \, , |\psi_\ell^{(\mathrm{out})}\rangle \right\}_{\ell=1}^{N}, \tag{6.306}$$

where $|\psi_\ell^{(\mathrm{in})}\rangle = \bigotimes_{i=1}^n |s_{\ell,i}^{(\mathrm{in})}\rangle$ is a random input state with $|s_{\ell,i}^{(\mathrm{in})}\rangle \in \mathrm{stab}_1$ sampled uniformly, and $|\psi_\ell^{(\mathrm{out})}\rangle = \bigotimes_{i=1}^n |s_{\ell,i}^{(\mathrm{out})}\rangle$ is the outcome of performing randomized Pauli measurement on $\mathcal{E}(|\psi_\ell^{(\mathrm{in})}\rangle\!\langle\psi_\ell^{(\mathrm{in})}|)$.

After obtaining the outcome from N randomized experiments, we can design a learning algorithm that learns a model of the unknown CPTP map $\mathcal{E}$, such that given an input state ρ and an observable O, the algorithm could predict $\mathrm{tr}(O\mathcal{E}(\rho))$. The rigorous guarantee is given in the following theorem.

Theorem 38 (Learning to predict a quantum evolution)**.** *Given $n, \epsilon, \epsilon', \delta > 0$. Consider any unknown n-qubit CPTP map $\mathcal{E}$. Given a classical shadow $S_N(\mathcal{E})$ of $\mathcal{E}$ obtained by N randomized experiments with*

$$N = \log\left(\frac{n}{\delta}\right) \min\left(2^{\mathcal{O}\left(\log(\frac{1}{\epsilon})\left(\log\log(\frac{1}{\epsilon})+\log(\frac{1}{\epsilon'})\right)\right)},\ 2^{\mathcal{O}(\log(1/\epsilon)\log(n))}\right). \tag{6.307}$$

With probability $\geq 1-\delta$, the algorithm learns a function h, s.t. for any n-qubit state distribution $\mathcal{D}$ invariant under single-qubit H and S gates, and any observable O given as a sum of few-body observables, where each qubit is acted on by $\mathcal{O}(1)$ of the few-body observables,

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} |h(\rho, O) - \mathrm{tr}\left(O\mathcal{E}(\rho)\right)|^2 \leq \left(\epsilon + \epsilon' \left[\frac{\|O^{(\mathrm{low})}\|}{\|O\|}\right]^{\frac{2\lceil\log_{1.5}(1/\epsilon)\rceil}{\lceil\log_{1.5}(1/\epsilon)\rceil+1}}\right) \|O\|^2. \tag{6.308}$$

Here, $O^{(\mathrm{low})}$ is the low-degree approximation of O after Heisenberg evolution under $\mathcal{E}$.

The scaling given in the main text corresponds to the additional assumption that $\|O\| \leq 1$. By noting that $\frac{2\lceil\log_{1.5}(1/\epsilon)\rceil}{\lceil\log_{1.5}(1/\epsilon)\rceil+1} \in [1, 2)$, we have

$$\left[\frac{\|O^{(\mathrm{low})}\|}{\|O\|}\right]^{\frac{2\lceil\log_{1.5}(1/\epsilon)\rceil}{\lceil\log_{1.5}(1/\epsilon)\rceil+1}} \|O\|^2 \leq \left\|O^{(\mathrm{low})}\right\|^{\frac{2\lceil\log_{1.5}(1/\epsilon)\rceil}{\lceil\log_{1.5}(1/\epsilon)\rceil+1}} \leq \max\left(\left\|O^{(\mathrm{low})}\right\|^2, 1\right). \tag{6.309}$$

Theorem 25 follows by considering $\epsilon' \to 0$.

Learning algorithm

Recall that a size-N classical shadow $S_N(\mathcal{E})$ of the CPTP map $\mathcal{E}$ is a set given by

$$S_N(\mathcal{E}) \triangleq \left\{ |\psi_\ell^{(\mathrm{in})}\rangle = \bigotimes_{i=1}^{n} |s_{\ell,i}^{(\mathrm{in})}\rangle \,, |\psi_\ell^{(\mathrm{out})}\rangle = \bigotimes_{i=1}^{n} |s_{\ell,i}^{(\mathrm{out})}\rangle \right\}_{\ell=1}^{N}. \tag{6.310}$$

Given an observable O that can be written as a sum of κ-qubit observables, where each qubit is acted on by at most d of the κ-qubit observables with $\kappa, d = \mathcal{O}(1)$. We have

$$O = \sum_{Q \in \{I,X,Y,Z\}^{\otimes n} : |Q| \leq \kappa} a_Q Q, \tag{6.311}$$

where $\sum_{Q:|Q|\leq\kappa} \mathbb{1}[a_Q \neq 0] = \mathcal{O}(n)$. The algorithm creates a dataset,

$$\left\{ \rho_\ell = |\psi_\ell^{(\mathrm{in})}\rangle\!\langle\psi_\ell^{(\mathrm{in})}|, \quad y_\ell(O) = \sum_{Q:|Q|\leq\kappa} a_Q \operatorname{tr}\left(Q \bigotimes_{i=1}^{n} \left(3|s_{\ell,i}^{(\mathrm{out})}\rangle\!\langle s_{\ell,i}^{(\mathrm{out})}| - I \right) \right) \right\}_{\ell=1}^{N} \tag{6.312}$$

from the classical shadow $S_N(\mathcal{E})$, which requires $\mathcal{O}(nN)$ computational time. We also define the parameter

$$\eta \triangleq \sum_{Q:|Q|\leq\kappa} |a_Q| = \|O\|_{\mathrm{Pauli},1} \tag{6.313}$$

based on the given observable O.

The sample complexity in Eq. (6.307) is the minimum of two arguments. Each of the two corresponds to a hyperparameter setting for k and $\tilde{\epsilon}$. Let $C(k)$ be the function from Corollary 12 and $C(k,d)$ be the function from Corollary 13. The first hyperparameter setting considers

$$k = \lceil \log_{1.5}(1/\epsilon) \rceil, \quad \tilde{\epsilon} = \left(\frac{\epsilon'}{6 \cdot 2^k} \right)^{k+1} \left(\frac{C(\kappa,d)}{3} \right)^2 \left(\frac{C(k)}{3} \right)^{2k}. \tag{6.314}$$

The second hyperparameter setting considers

$$k = \lceil \log_{1.5}(2/\epsilon) \rceil, \quad \tilde{\epsilon} = \frac{\epsilon}{9 \cdot 2^{k+1} \cdot n^k} \left(\frac{C(\kappa,d)}{3} \right)^2. \tag{6.315}$$

For every Pauli observable $P \in \{I,X,Y,Z\}^{\otimes n}$ with $|P| \leq k$, the algorithm computes

$$\hat{x}_P(O) = \frac{1}{N} \sum_{\ell=1}^{N} \operatorname{tr}(P\rho_\ell) y_\ell(O), \tag{6.316}$$

$$\hat{\beta}_P = \left(\frac{1}{3} \right)^{|P|}, \tag{6.317}$$

$$\hat{\alpha}_P(O) = \begin{cases} 0, & \hat{\beta}_P \leq 2\tilde{\epsilon}, \\ 0, & \hat{\beta}_P > 2\tilde{\epsilon},\ |\hat{x}_P(O)|/\hat{\beta}_P^{1/2} \leq 2\eta\sqrt{\tilde{\epsilon}}, \\ \hat{x}_P(O)/\hat{\beta}_P, & \hat{\beta}_P > 2\tilde{\epsilon},\ |\hat{x}_P(O)|/\hat{\beta}_P^{1/2} > 2\eta\sqrt{\tilde{\epsilon}}, \end{cases} \tag{6.318}$$

which requires $\mathcal{O}(kN)$ time per Pauli observable P. Finally, given an n-qubit state ρ, the algorithm outputs

$$h(\rho, O) \triangleq \sum_{P:|P|\leq k} \hat{\alpha}_P(O)\,\mathrm{tr}(P\rho), \tag{6.319}$$

which uses a computational time of $\mathcal{O}(n^k)$.

Rigorous performance guarantee

In this section, we prove that the learning algorithm presented in the last section satisfies Theorem 38. The proof uses the tools presented in Section 6.8 and is similar to the proof of Theorem 37.

Definitions

For a given observable that is a sum of κ-qubit observables, where $\kappa = \mathcal{O}(1)$ and each qubit is acted on by $d = \mathcal{O}(1)$ of the κ-qubit observables, we can write

$$O = \sum_{Q\in\{I,X,Y,Z\}^{\otimes n}:|Q|\leq\kappa} a_Q Q. \tag{6.320}$$

We define a few variables based on O as follows. We consider the unknown observable to be

$$O^{(\mathrm{unk})} \triangleq \mathcal{E}^\dagger(O) \triangleq \sum_{P\in\{I,X,Y,Z\}^{\otimes n}} \alpha_P(O) P, \tag{6.321}$$

and the low-degree approximation of $O^{(\mathrm{unk})}$ to be

$$O^{(\mathrm{low})} \triangleq \sum_{P\in\{I,X,Y,Z\}^{\otimes n}:|P|\leq k} \alpha_P(O) P. \tag{6.322}$$

Then for all Pauli observables $P \in \{I, X, Y, Z\}^{\otimes n}$, we define

$$x_P(O) \triangleq \left(\frac{1}{3}\right)^{|P|} \alpha_P(O), \quad \beta_P \triangleq \left(\frac{1}{3}\right)^{|P|}. \tag{6.323}$$

We also define the standard n-qubit input state distribution $\mathcal{D}^0$ to be the uniform distribution over the tensor product of n single-qubit stabilizer states. A nice property of $\mathcal{D}^0$ is that for any state ρ in the support of $\mathcal{D}^0$, the non-identity purity for a subsystem A of size L is

$$\gamma^*(\rho_A) = \frac{1}{2^L}. \tag{6.324}$$

Using this property and Lemma 43 on extracting Pauli coefficients, we have the following identities

$$x_P(O) = \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0} \operatorname{tr}(P\rho) \operatorname{tr}\left(\mathcal{E}^\dagger(O)\rho\right), \tag{6.325}$$

$$\beta_P = \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0} \operatorname{tr}(P\rho)^2 = \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0} \left[\gamma^*(\rho_{\mathsf{dom}(P)})\right] \left(\frac{2}{3}\right)^{|P|}. \tag{6.326}$$

We are now ready to prove Theorem 38.

Prediction error under standard distribution $\mathcal{D}^0$ (first set of hyperparameters)

We begin the proof by considering the first set of hyperparameters $k, \tilde{\epsilon}$ as given in Eq. (6.314). For a Pauli observable $Q \in \{I, X, Y, Z\}^{\otimes n}$ with $|Q| \leq \kappa = O(1)$, we consider the random variable

$$\hat{x}_P(Q) = \frac{1}{N}\sum_{\ell=1}^{N} \operatorname{tr}(P\rho_\ell) y_\ell(Q) = \frac{1}{N}\sum_{\ell=1}^{N} \operatorname{tr}(P\rho_\ell) \operatorname{tr}\left(Q \bigotimes_{i=1}^{n} \left(3|s_{\ell,i}^{(\mathrm{out})}\rangle\langle s_{\ell,i}^{(\mathrm{out})}| - I\right)\right). \tag{6.327}$$

Because $|Q| = O(1)$, we have $\left|\operatorname{tr}\left(Q \bigotimes_{i=1}^{n} \left(3|s_{\ell,i}^{(\mathrm{out})}\rangle\langle s_{\ell,i}^{(\mathrm{out})}| - I\right)\right)\right| = O(1)$ with probability one. By considering the size of the classical shadow $S_N(\mathcal{E})$ to be

$$N = \Omega\left(\frac{\log(n^{k+\kappa}/\delta)}{\tilde{\epsilon}^2}\right), \tag{6.328}$$

we can utilize Hoeffding's inequality and union bound to guarantee that

$$|\hat{x}_P(Q) - x_P(Q)| \leq \tilde{\epsilon}, \quad \forall P, Q \in \{I, X, Y, Z\}^{\otimes n}, |P| \leq k, |Q| \leq \kappa \tag{6.329}$$

with probability at least $1 - \delta$. In the following proof, we will condition on the above event.

Using triangle inequality, we have

$$|\hat{x}_P(O) - x_P(O)| \leq \|O\|_{\mathrm{Pauli},1}\, \tilde{\epsilon} = \eta\tilde{\epsilon}, \quad \left|\hat{\beta}_P - \beta_P\right| = 0, \quad \forall P : |P| \leq k. \tag{6.330}$$

The norm inequality given in Corollary 12 shows that

$$\sum_{P:|P|\leq k} |\alpha_P(O)|^r \leq \left(\frac{3}{C(k)}\right)^r \left\|O^{(\mathrm{low})}\right\|^r \tag{6.331}$$

for the constant $C(k)$ defined in (6).

The filtering lemma given in Lemma 45 shows that

$$\sum_{P:|P|\leq k} \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0} \left[\gamma^*(\rho_{\mathsf{dom}(P)})\right] \left(\frac{2}{3}\right)^{|P|} |\hat{\alpha}_P(O) - \alpha_P(O)|^2 \tag{6.332}$$

$$\leq 6\eta^{2-r} \left(\frac{3}{C(k)}\right)^r \left\|O^{(\mathrm{low})}\right\|^r \tilde{\epsilon}^{1-(r/2)}. \tag{6.333}$$

From the norm inequality and the constant $C(k, d)$ given in Corollary 13, we have

$$\eta = \|O\|_{\mathrm{Pauli},1} \leq \frac{3}{C(\kappa, d)} \|O\|\,. \tag{6.334}$$

Combining with the definition of $\tilde{\epsilon}$ given in Eq. (6.314), we have

$$\sum_{P:|P|\leq k} \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0} \left[\gamma^*(\rho_{\mathsf{dom}(P)})\right] \left(\frac{2}{3}\right)^{|P|} |\hat{\alpha}_P(O) - \alpha_P(O)|^2 \leq \left[\frac{\|O^{(\mathrm{low})}\|}{\|O\|}\right]^r \frac{\epsilon'}{2^k} \cdot \|O\|^2\,. \tag{6.335}$$

Using Lemma 41 on mean squared error and Corollary 14 on low-degree approximation, we have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0} \left|h(\rho, O) - \mathrm{tr}(O^{(\mathrm{unk})}\rho)\right|^2 \tag{6.336}$$

$$\leq \underbrace{(2/3)^k \left\|O^{(\mathrm{unk})}\right\|^2}_{\leq \left\|O^{(\mathrm{unk})}\right\|^2 \epsilon} + \sum_{P:|P|\leq k} \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0} \left[\gamma^*(\rho_{\mathsf{dom}(P)})\right] \left(\frac{2}{3}\right)^{|P|} |\hat{\alpha}_P(O) - \alpha_P(O)|^2\,. \tag{6.337}$$

Using the definition of $O^{(\mathrm{unk})}$, we have $O^{(\mathrm{unk})} = \mathcal{E}^\dagger(O)$ and $\left\|O^{(\mathrm{unk})}\right\| \leq \|O\|$. Hence

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0} |h(\rho, O) - \mathrm{tr}(O\mathcal{E}(\rho))|^2 \leq \left(\epsilon + \frac{\epsilon'}{2^k} \left[\frac{\|O^{(\mathrm{low})}\|}{\|O\|}\right]^r\right) \|O\|^2\,, \tag{6.338}$$

which establishes a prediction error bound for distribution $\mathcal{D}^0$.

Prediction error under general distribution $\mathcal{D}$ (first set of hyperparameters)

We now consider an arbitrary n-qubit state distribution $\mathcal{D}$ invariant under single-qubit H and S gates. Using Lemma 41 on mean squared error and Corollary 14 on low-degree approximation, we have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \left|h(\rho, O) - \mathrm{tr}(O^{(\mathrm{unk})}\rho)\right|^2 \tag{6.339}$$

$$\leq \epsilon \left\|O\right\|^2 + \sum_{P:|P|\leq k} \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \left[\gamma^*(\rho_{\mathsf{dom}(P)})\right] \left(\frac{2}{3}\right)^{|P|} \left|\hat{\alpha}_P(O) - \alpha_P(O)\right|^2. \tag{6.340}$$

Recall that $\gamma^*(\rho_{\mathsf{dom}(P)}) \leq 1$, hence

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \left[\gamma^*(\rho_{\mathsf{dom}(P)})\right] \left(\frac{2}{3}\right)^{|P|} \leq 2^k \left(\frac{1}{3}\right)^{|P|}, \quad \forall P \in \{I, X, Y, Z\}^{\otimes n}, |P| \leq k. \tag{6.341}$$

Furthermore, we have $\mathbb{E}_{\rho\sim\mathcal{D}_0} \left[\gamma^*(\rho_{\mathsf{dom}(P)})\right] \left(\frac{2}{3}\right)^{|P|} = (1/3)^{|P|}$. Together, we have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \left|h(\rho, O) - \mathrm{tr}(O^{(\mathrm{unk})}\rho)\right|^2 \tag{6.342}$$

$$\leq \epsilon \left\|O\right\|^2 + 2^k \sum_{P:|P|\leq k} \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0} \left[\gamma^*(\rho_{\mathsf{dom}(P)})\right] \left(\frac{2}{3}\right)^{|P|} \left|\hat{\alpha}_P(O) - \alpha_P(O)\right|^2. \tag{6.343}$$

Combining the above with Eq. (6.335), we have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \left|h(\rho, O) - \mathrm{tr}(O\mathcal{E}(\rho))\right|^2 \leq \left(\epsilon + \epsilon' \left[\frac{\|O^{(\mathrm{low})}\|}{\|O\|}\right]^r\right) \left\|O\right\|^2, \tag{6.344}$$

which is the prediction error under distribution $\mathcal{D}$.

Putting everything together (first set of hyperparameters)

From Eq. (6.314), we have set the parameter $\tilde{\epsilon}$ to be

$$\tilde{\epsilon} = \left(\frac{\epsilon'}{6}\right)^{k+1} \left(\frac{C(\kappa, d)}{3}\right)^2 \left(\frac{C(k)}{3}\right)^{2k}. \tag{6.345}$$

Furthermore, given the classical shadow $S_N(\mathcal{E})$ of size

$$N = \mathcal{O}\left(\frac{\log(n^{k+\kappa}/\delta)}{\tilde{\epsilon}^2}\right) = \log\left(\frac{n}{\delta}\right) 2^{\mathcal{O}\left(\log(\frac{1}{\epsilon})\left(\log\log(\frac{1}{\epsilon}) + \log(\frac{1}{\epsilon'})\right)\right)}, \tag{6.346}$$

we can guarantee that with probability at least $1 - \delta$, the following holds. For any observable O that is a sum of κ-qubit observables, where $\kappa = \mathcal{O}(1)$ and each qubit is acted on by $d = \mathcal{O}(1)$ of the κ-qubit observables, and any n-qubit state distribution $\mathcal{D}$ invariant under single-qubit H and S gates, we have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}} \left|h(\rho, O) - \mathrm{tr}(O\mathcal{E}(\rho))\right|^2 \leq \left(\epsilon + \epsilon' \left[\frac{\|O^{(\mathrm{low})}\|}{\|O\|}\right]^r\right) \left\|O\right\|^2. \tag{6.347}$$

This establishes one of the argument for the sample complexity stated in Theorem 38.

Prediction error under standard distribution $\mathcal{D}^0$ (second set of hyperparameters)

In the following proof, we consider the second set of hyperparameters $k, \tilde{\epsilon}$ as given in Eq. (6.315). By considering the size of the classical shadow $S_N(\mathcal{E})$ to be

$$N = \Omega\left(\frac{\log(n^{k+\kappa}/\delta)}{\tilde{\epsilon}^2}\right), \tag{6.348}$$

we can utilize Hoeffding's inequality and union bound to guarantee that

$$|\hat{x}_P(Q) - x_P(Q)| \leq \tilde{\epsilon}, \quad \forall P, Q \in \{I, X, Y, Z\}^{\otimes n}, |P| \leq k, |Q| \leq \kappa \tag{6.349}$$

with probability at least $1-\delta$. In the following proof, we will condition on the above event. Using triangle inequality, we have

$$|\hat{x}_P(O) - x_P(O)| \leq \|O\|_{\mathrm{Pauli},1}\, \tilde{\epsilon} = \eta\tilde{\epsilon}, \quad \left|\hat{\beta}_P - \beta_P\right| = 0, \quad \forall P : |P| \leq k. \tag{6.350}$$

The filtering lemma given in Lemma 45 shows that

$$\sum_{P:|P|\leq k} \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0}\left[\gamma^*(\rho_{\mathsf{dom}(P)})\right]\left(\frac{2}{3}\right)^{|P|} |\hat{\alpha}_P(O) - \alpha_P(O)|^2 \leq 9\eta^2\tilde{\epsilon}^2. \tag{6.351}$$

From the norm inequality and the function $C(k,d)$ given in Corollary 13, we have

$$\eta = \|O\|_{\mathrm{Pauli},1} \leq \frac{3}{C(\kappa,d)}\|O\|. \tag{6.352}$$

Combining with the definition of $\tilde{\epsilon}$ given in Eq. (6.315), we have

$$\sum_{P:|P|\leq k} \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0}\left[\gamma^*(\rho_{\mathsf{dom}(P)})\right]\left(\frac{2}{3}\right)^{|P|} |\hat{\alpha}_P(O) - \alpha_P(O)|^2 \leq \frac{\epsilon}{2^{k+1}} \cdot \|O\|^2. \tag{6.353}$$

Using Lemma 41 on mean squared error and Corollary 14 on low-degree approximation, we have

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0}\left|h(\rho,O) - \mathrm{tr}(O^{(\mathrm{unk})}\rho)\right|^2 \tag{6.354}$$

$$\leq (2/3)^k \left\|O^{(\mathrm{unk})}\right\|^2 + \sum_{P:|P|\leq k} \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0}\left[\gamma^*(\rho_{\mathsf{dom}(P)})\right]\left(\frac{2}{3}\right)^{|P|} |\hat{\alpha}_P(O) - \alpha_P(O)|^2 \tag{6.355}$$

$$\leq \frac{\epsilon}{2}\left\|O^{(\mathrm{unk})}\right\|^2 + \sum_{P:|P|\leq k} \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0}\left[\gamma^*(\rho_{\mathsf{dom}(P)})\right]\left(\frac{2}{3}\right)^{|P|} |\hat{\alpha}_P(O) - \alpha_P(O)|^2. \tag{6.356}$$

Using the definition of $O^{(\mathrm{unk})}$, we have $O^{(\mathrm{unk})} = \mathcal{E}^\dagger(O)$ and $\left\|O^{(\mathrm{unk})}\right\| \leq \|O\|$. Hence

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}^0}|h(\rho,O) - \mathrm{tr}(O\mathcal{E}(\rho))|^2 \leq \frac{1}{2}\left(\epsilon + \frac{\epsilon}{2^k}\right)\|O\|^2, \tag{6.357}$$

which establishes a prediction error bound for distribution $\mathcal{D}^0$.

Prediction error under general distribution $\mathcal{D}$ (second set of hyperparameters)

We now consider an arbitrary n-qubit state distribution $\mathcal{D}$ invariant under single-qubit H and S gates. Using Lemma 41 on mean squared error, Corollary 14 on low-degree approximation, $k = \lceil \log_{1.5}(2/\epsilon) \rceil$, the fact that $\gamma^*(\rho_{\mathsf{dom}(P)}) \leq 1$, and $\mathbb{E}_{\rho \sim \mathcal{D}_0} \left[\gamma^*(\rho_{\mathsf{dom}(P)}) \right] \left(\frac{2}{3}\right)^{|P|} = (1/3)^{|P|}$, we have

$$\underset{\rho \sim \mathcal{D}}{\mathbb{E}} \left| h(\rho, O) - \mathrm{tr}(O^{(\mathrm{unk})} \rho) \right|^2 \tag{6.358}$$

$$\leq \frac{\epsilon}{2} \|O\|^2 + 2^k \sum_{P : |P| \leq k} \underset{\rho \sim \mathcal{D}^0}{\mathbb{E}} \left[\gamma^*(\rho_{\mathsf{dom}(P)}) \right] \left(\frac{2}{3} \right)^{|P|} \left| \hat{\alpha}_P(O) - \alpha_P(O) \right|^2 . \tag{6.359}$$

Combining the above with Eq. (6.353), we have

$$\underset{\rho \sim \mathcal{D}}{\mathbb{E}} \left| h(\rho, O) - \mathrm{tr}(O \mathcal{E}(\rho)) \right|^2 \leq \epsilon \|O\|^2 , \tag{6.360}$$

which is the prediction error under distribution $\mathcal{D}$.

Putting everything together (second set of hyperparameters)

From Eq. (6.315), we have set the parameter $\tilde{\epsilon}$ to be

$$\tilde{\epsilon} = \frac{\epsilon}{9 \cdot 2^{k+1} \cdot n^k} \left(\frac{C(\kappa, d)}{3} \right)^2 . \tag{6.361}$$

Furthermore, given the classical shadow $S_N(\mathcal{E})$ of size

$$N = \mathcal{O}\left(\frac{\log(n^{k+\kappa}/\delta)}{\tilde{\epsilon}^2} \right) = \log\left(\frac{n}{\delta} \right) 2^{\mathcal{O}\left(\log(\frac{1}{\epsilon}) \log(n) \right)}, \tag{6.362}$$

we can guarantee that with probability at least $1 - \delta$, the following holds. For any observable O that is a sum of κ-qubit observables, where $\kappa = \mathcal{O}(1)$ and each qubit is acted on by $d = \mathcal{O}(1)$ of the κ-qubit observables, and any n-qubit state distribution $\mathcal{D}$ invariant under single-qubit H and S gates, we have

$$\underset{\rho \sim \mathcal{D}}{\mathbb{E}} \left| h(\rho, O) - \mathrm{tr}(O \mathcal{E}(\rho)) \right|^2 \leq \epsilon \|O\|^2 . \tag{6.363}$$

This completes the proof of Theorem 38.

6.10 Details of numerical experiments

In the numerical experiments, we consider the two classes of Hamiltonians,

$$H = \frac{1}{4}\sum_i (X_i X_{i+1} + Y_i Y_{i+1}) + \frac{1}{2}\sum_i h_i Z_i, \qquad \text{(XY model)} \tag{6.364}$$

$$H = \frac{1}{2}\sum_i X_i X_{i+1} + \frac{1}{2}\sum_i h_i Z_i, \qquad \text{(Ising model)} \tag{6.365}$$

where $h_i = 0.5$ for the homogeneous Z field, and h_i is sampled uniformly at random from $[-5, 5]$ for the disordered Z field. We solve for the time-evolved properties using the Jordan-Wigner transform to map the spin chains to a free fermion model and the technique described in (E. Lieb, Schultz, and Mattis, 1961) to solve the free fermion model.

We consider the training set to be a collection of N random product states $|\psi_\ell\rangle, \ell = 1, \ldots, N$ and their associated measured properties y_ℓ corresponding to measuring an observable O after evolving under $U(t) = \exp(-itH)$. The measured properties are averaged over 500 measurements. Hence y_ℓ is a noisy estimate of the true expectation value $\mathrm{tr}(OU(t)|\psi_\ell\rangle\langle\psi_\ell|U(t)^\dagger)$. We consider essentially the same ML algorithm as described in Section 6.2, but utilize a more sophisticated approach to enforce sparsity in $\hat{\alpha}_P$. We also consider α_P for Pauli operator P that is geometrically local. For ease of analysis, we consider a simple strategy of setting small values to zero. The standard approach that is often used in practice is LASSO (R. Tibshirani, 1996).

In the numerical experiments, we perform a simple grid search for the two hyperparameters using two-fold cross-validation on the training set:

$$k = 1, 2, 3, 4, \tag{6.366}$$

$$a = 2^{-15}, 2^{-14}, 2^{-13}, \ldots, 2^{-4}, 2^{-3}, \tag{6.367}$$

where k corresponds to the maximum number of qubits that the Pauli operators P act on, and a is a hyperparameter corresponding to the strength of the ℓ_1 regularization term in LASSO. In particular, the optimization problem of LASSO is given by

$$\min_{\hat{\alpha}_P} \frac{1}{2N}\sum_{\ell=1}^{N}\left| y_\ell - \sum_{P:|P|\leq k} \hat{\alpha}_P \,\mathrm{tr}(P|\psi_\ell\rangle\langle\psi_\ell|)\right|^2 + a \sum_{P:|P|\leq k} |\hat{\alpha}_P|, \tag{6.368}$$

where $|P|$ is the number of qubits that the Pauli observable P acts nontrivially on. We then use the values $\hat{\alpha}_P$ found by the above optimization to form a succinct

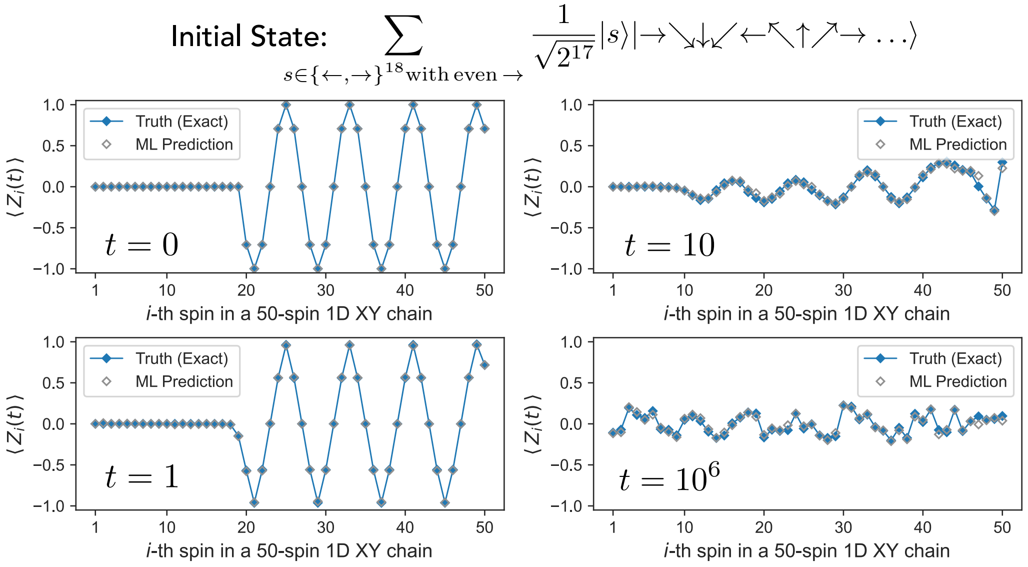


Figure 6.5: *Visualization of ML model's prediction for a highly-entangled initial state* $\rho = |\psi\rangle\langle\psi|$. We consider the expected value of $Z_i(t) = e^{itH} Z_i e^{-itH}$, where H corresponds to the 1D 50-spin XY chain with a homogeneous Z field. The initial state $|\psi\rangle$ has a GHZ-like entanglement over the first 18-spin chain and is a product state with spins rotating clockwise over the latter 32-spin chain. To prepare $|\psi\rangle$ with 1D circuits, a depth of at least $\Omega(n)$ is required. Even though the ML model is trained only on random product states (a total of $N = 10000$), it still performs accurately in predicting the highly-entangled state over a wide range of evolution time t.

approximate model

$$\sum_{P:|P|\leq k} \hat{\alpha}_P P \tag{6.369}$$

of the time-evolved observable $O(t) = U(t)^\dagger O U(t)$. Given a new initial state ρ, we would predict the time-evolved property $\mathrm{tr}(O(t)\rho) = \mathrm{tr}(OU(t)\rho U(t)^\dagger)$ using

$$\sum_{P:|P|\leq k} \hat{\alpha}_P \, \mathrm{tr}(P\rho). \tag{6.370}$$

In addition to the figures given in the main text, Fig. 6.5 shows another example for predicting a highly entangled initial state. Even though the ML model is trained with random product states, it still performs very well on a structured entangled initial state.

Part III

Learning with quantum machines

Chapter 7

INFORMATION-THEORETIC BOUNDS ON QUANTUM ADVANTAGE

The widespread applications of machine learning (ML) to problems of practical interest have fueled interest in machine learning using quantum platforms (Biamonte et al., 2017; Schuld and Killoran, 2019; Havlicek et al., 2019). Though many potential applications of quantum ML have been proposed, so far the prospect for quantum advantage in solving purely classical problems remains unclear (E. Tang, 2019; E. Tang, 2018; Gilyén, Lloyd, and E. Tang, 2018; Arrazola et al., 2019). On the other hand, it seems plausible that quantum ML can be fruitfully applied to problems faced by quantum scientists, such as characterizing the properties of quantum systems and predicting the outcomes of quantum experiments (Carleo and Troyer, 2017a; Van Nieuwenburg, Y.-H. Liu, and Sebastian D Huber, 2017; Carrasquilla and Roger G Melko, 2017a; Gilmer et al., 2017; Melnikov et al., 2018; Sharir et al., 2020; Aharonov, J. S. Cotler, and Qi, 2021).

Here we focus on an important class of learning problems motivated by quantum mechanics. Namely, we are interested in predicting functions of the form

$$f(x) = \mathrm{tr}(O\mathcal{E}(|x\rangle\langle x|)), \tag{7.1}$$

where x is a classical input, $\mathcal{E}$ is an arbitrary (possibly unknown) completely positive and trace preserving (CPTP) map, and O is a known observable. Equation (7.1) encompasses *any* physical process that takes a classical input and produces a real number as output. The goal is to construct a function $h(x)$ that accurately approximates $f(x)$ after accessing the physical process $\mathcal{E}$ as few times as possible.

A particularly important special case of setup (7.1) is training an ML model to predict what would happen in physical experiments (Melnikov et al., 2018). Such experiments might explore, for instance, the outcome of a reaction in quantum chemistry (Z. Zhou, Xiaocheng Li, and Zare, 2017), ground state properties of a novel molecule or material (Parr, 1980; Car and Parrinello, 1985; Becke, 1993; Steven R White, 1993a; Peruzzo et al., 2014; Kandala et al., 2017; Gilmer et al., 2017), or the behavior of neutral atoms in an analog quantum simulator (Buluta and

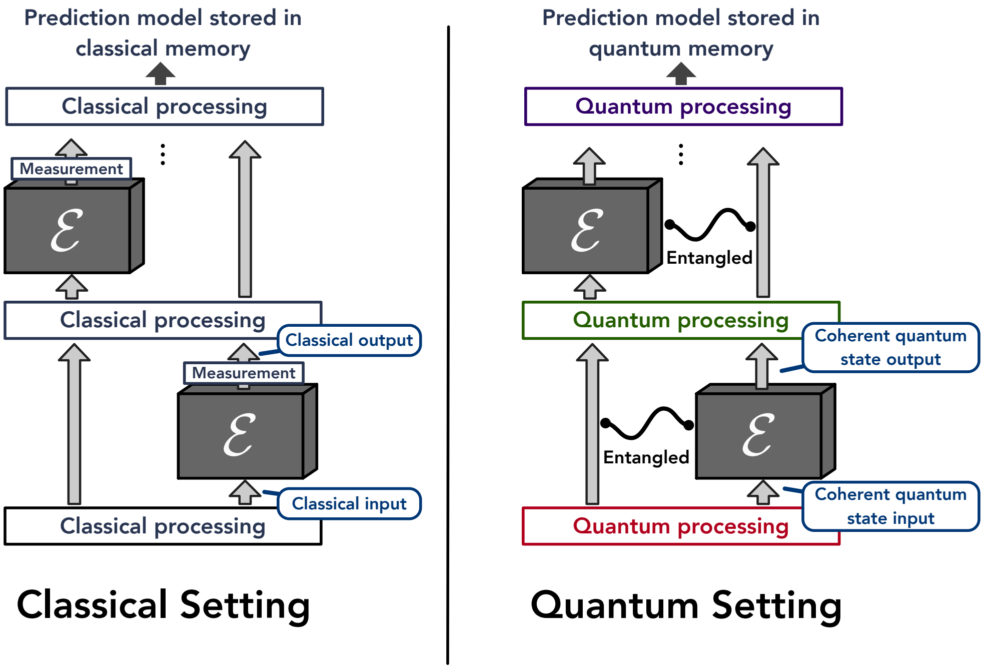


Figure 7.1: *Illustration of classical and quantum machine learning settings:* The goal is to learn about an unknown CPTP map $\mathcal{E}$ by performing physical experiments. (Left) In the learning phase of the classical ML setting, a measurement is performed after each query to $\mathcal{E}$; the classical measurement outcomes collected during the learning phase are consulted during the prediction phase. (Right) In the learning phase of the quantum ML setting, multiple queries to $\mathcal{E}$ may be included in a single coherent quantum circuit, yielding an output state stored in a quantum memory; this stored quantum state is consulted during the prediction phase.

Nori, 2009; Levine et al., 2018; Bernien et al., 2017). In these cases, the input x subsumes parameters that characterize the process, e.g., chemicals involved in the reaction, a description of the molecule, or the intensity of lasers that control the neutral atoms. The map $\mathcal{E}$ characterizes a quantum evolution happening in the lab. Depending on the parameter x, it produces the quantum state $\mathcal{E}(|x\rangle\langle x|)$. Finally, the experimentalist measures a certain observable O at the end of the experiment. The goal is to predict the measurement outcome for new physical experiments with new values of x that have not been encountered during the training process.

Motivated by these concrete applications, we want to understand the power of classical and quantum ML models in predicting functions of the form given in Equation (7.1). On the one hand, we consider classical ML models that can gather classical measurement data $\{(x_i, o_i)\}_{i=1}^{N_C}$, where o_i is the outcome when we perform a POVM measurement on the state $\mathcal{E}(|x_i\rangle\langle x_i|)$. We denote by N_C the number of such experiments performed during training in the classical ML setting. On the other hand, we consider quantum ML models in which multiple runs of the CPTP map $\mathcal{E}$ can be composed coherently to collect quantum data, and predictions are produced by a quantum computer with access to the quantum data. We denote by N_Q the number of times $\mathcal{E}$ is used during training in the quantum setting. The classical and quantum ML settings are illustrated in Figure 7.1.

We focus on the question of whether quantum ML can have a large advantage over classical ML: to achieve a small prediction error, can the optimal N_Q in the quantum ML setting be much less than the optimal N_C in the classical ML setting? For the purpose of this comparison, we disregard the runtime of the classical or quantum ML models that generate the predictions; we are only interested in how many times the process $\mathcal{E}$ must run during the learning phase in the quantum and classical settings.

The main result of this chapter addresses small *average* prediction error, i.e., the prediction error $|h(x) - f(x)|^2$ averaged over some specified input distribution $\mathcal{D}(x)$. We rigorously show that, for any $\mathcal{E}$, O, and $\mathcal{D}$, and for any quantum ML model, one can always design a classical ML model achieving a similar average prediction error such that N_C is larger than N_Q by at worst a small polynomial factor. Hence, there is no exponential advantage of quantum ML over classical ML if the goal is to achieve a small average prediction error and if the efficiency is quantified by the number of times $\mathcal{E}$ is used in the learning process. This statement holds for existing quantum ML models running on near-term devices (Havlicek et al., 2019; Schuld and Killoran, 2019; Huang, Broughton, Masoud Mohseni, Babbush, Boixo, Neven, and Jarrod R McClean, 2021a) and future quantum ML models yet to be conceived. We note, though, that while there is no large advantage in query complexity, a substantial quantum advantage in computational complexity is possible (Servedio and Gortler, 2004).

7.1 Machine learning settings

We assume that the observable O (with $\|O\| \leq 1$) is known and the physical experiment $\mathcal{E}$ is an unknown CPTP map that belongs to a set of CPTP maps $\mathcal{F}$. Apart from $\mathcal{E} \in \mathcal{F}$, the process can be arbitrary — a common assumption in statistical learning theory (Leslie G Valiant, 1984; Blumer et al., 1989; Bartlett and Mendelson, 2002; Vapnik, 2013; Arunachalam and Wolf, 2017). For the sake of concreteness, we assume that $\mathcal{E}$ is a CPTP map from a Hilbert space of n qubits to a Hilbert space of m qubits. Regarding inputs, we consider bit-strings of size n: $x \in \{0,1\}^n$. This is not a severe restriction, since floating-point representations of continuous parameters can always be truncated to a finite number of digits. We now give precise definitions for classical and quantum ML settings; see Fig. 7.1 for an illustration.

Classical (C) ML: The ML model consists of two phases: learning and prediction. During the learning phase, a randomized algorithm selects classical inputs x_i and we perform a (quantum) experiment that results in an outcome o_i from performing a POVM measurement on $\mathcal{E}(|x_i\rangle\langle x_i|)$. A total of N_{C} experiments give rise to the classical training data $\{(x_i, o_i)\}_{i=1}^{N_{\mathrm{C}}}$. After obtaining this training data, the ML model executes a randomized algorithm $\mathcal{A}$ to learn a prediction model

$$s_{\mathrm{C}} = \mathcal{A}\left(\{(x_1, o_1), \dots (x_{N_{\mathrm{C}}}, o_{N_{\mathrm{C}}})\}\right), \tag{7.2}$$

where s_{C} is stored in the classical memory. In the prediction phase, a sequence of new inputs $\tilde{x}_1, \tilde{x}_2, \ldots \in \{0,1\}^n$ is provided. The ML model will use s_{C} to evaluate predictions $h_{\mathrm{C}}(\tilde{x}_1), h_{\mathrm{C}}(\tilde{x}_2), \ldots$ that approximate $f(\tilde{x}_1), f(\tilde{x}_2), \ldots$ up to small errors.

Restricted classical ML: We will also consider a restricted version of the classical setting. Rather than performing arbitrary POVM measurements, we restrict the ML model to measure the target observable O on the output state $\mathcal{E}|x_i\rangle\langle x_i|$ to obtain the measurement outcome o_i. In this case, we always have $o_i \in \mathbb{R}$ and $\mathbb{E}[o_i] = \mathrm{tr}(O\mathcal{E}(|x_i\rangle\langle x_i|))$.

Quantum (Q) ML: During the learning phase, the model starts with an initial state ρ_0 in a Hilbert space of arbitrarily high dimension. Subsequently, the quantum ML model accesses the unknown CPTP map $\mathcal{E}$ a total of N_{Q} times. These queries are interleaved with quantum data processing steps:

$$\rho_{\mathcal{E}} = C_{N_{\mathrm{Q}}}(\mathcal{E} \otimes \mathcal{I})C_{N_{\mathrm{Q}}-1} \ldots C_1(\mathcal{E} \otimes \mathcal{I})(\rho_0), \tag{7.3}$$

where each C_i is an arbitrary but known CPTP map, and we write $\mathcal{E} \otimes \mathcal{I}$ to emphasize that $\mathcal{E}$ acts on an n-qubit subsystem of a larger quantum system. The final state $\rho_{\mathcal{E}}$, encoding the prediction model learned from the queries to the unknown CPTP map $\mathcal{E}$, is stored in a quantum memory. In the prediction phase, a sequence of new inputs $\tilde{x}_1, \tilde{x}_2, \ldots \in \{0,1\}^n$ is provided. A quantum computer with access to the stored quantum state $\rho_{\mathcal{E}}$ executes a computation to produce prediction values $h_{\mathrm{Q}}(\tilde{x}_1), h_{\mathrm{Q}}(\tilde{x}_2), \ldots$ that approximate $f(\tilde{x}_1), f(\tilde{x}_2), \ldots$ up to small errors[1].

The quantum ML setting is strictly more powerful than the classical ML setting. During the prediction phase, classical ML models are restricted to processing classical data, albeit data obtained by measuring a quantum system during the learning phase. In contrast, quantum ML models can work directly with the quantum data and perform quantum data processing. A quantum ML model can have an exponential advantage relative to classical ML models for some tasks, as we demonstrate in Chapter 9.

7.2 Average-case prediction error

For a prediction model $h(x)$, we consider the average-case prediction error

$$\sum_{x\in\{0,1\}^n} \mathcal{D}(x)|h(x) - \mathrm{tr}(O\mathcal{E}(|x\rangle\!\langle x|))|^2, \tag{7.4}$$

with respect to a fixed distribution $\mathcal{D}$ over inputs. This could, for instance, be the uniform distribution.

Although learning from quantum data is strictly more powerful than learning from classical data, there are fundamental limitations. The following rigorous statement limits the potential for quantum advantage.

Theorem 39. *Fix an n-bit probability distribution $\mathcal{D}$, an m-qubit observable O ($\|O\| \leq 1$) and a set $\mathcal{F}$ of CPTP maps with n input qubits and m output qubits. Suppose there is a quantum ML model which accesses the map $\mathcal{E} \in \mathcal{F}$ N_{Q} times, producing with high probability a function $h_{\mathrm{Q}}(x)$ that achieves*

$$\sum_{x\in\{0,1\}^n} \mathcal{D}(x) \left|h_{\mathrm{Q}}(x) - \mathrm{tr}(O\mathcal{E}(|x\rangle\!\langle x|))\right|^2 \leq \epsilon. \tag{7.5}$$

Then there is an ML model in the restricted classical setting which accesses $\mathcal{E}$ $N_{\mathrm{C}} = \mathcal{O}(mN_{\mathrm{Q}}/\epsilon)$ times and produces with high probability a function h_{C} that achieves

$$\sum_{x\in\{0,1\}^n} \mathcal{D}(x) \left|h_{\mathrm{C}}(x) - \mathrm{tr}(O\mathcal{E}(|x\rangle\!\langle x|))\right|^2 = \mathcal{O}(\epsilon). \tag{7.6}$$

[1] Due to non-commutativity of quantum measurements, the ordering of new inputs matters. For instance, the two lists $\tilde{x}_1, \tilde{x}_2$ and $\tilde{x}_2, \tilde{x}_1$ can lead to different outcome predictions $h_{\mathrm{Q}}(\tilde{x}_i)$. Our main results do not depend on this subtletey — they are valid, irrespective of prediction input ordering.

Proof sketch. The proof consists of two parts. First, we cover the entire set of CPTP maps $\mathcal{F}$ with a maximal packing net, i.e. the largest subset $\mathcal{S} = \{\mathcal{E}_s\}_{s=1}^{|\mathcal{S}|} \subset \mathcal{F}$ such that the functions $f_{\mathcal{E}_s}(x) = \mathrm{tr}(O\mathcal{E}_s(|x\rangle\langle x|))$ obey

$$\sum_{x\in\{0,1\}^n} \mathcal{D}(x) \left|f_{\mathcal{E}_s}(x) - f_{\mathcal{E}_{s'}}(x)\right|^2 > 4\epsilon \tag{7.7}$$

whenever $s \neq s'$. We then set up a communication protocol as follows. Alice chooses an element s of the packing net uniformly at random, records her choice s, and then applies $\mathcal{E}_s$ N_Q times to prepare a quantum state $\rho_{\mathcal{E}_s}$ as in Eq. (7.3). Alice's random ensemble of quantum states is thus given by

$$\rho_{\mathcal{E}_s} \text{ with probability } p_s = \tfrac{1}{|\mathcal{S}|} \tag{7.8}$$

for $s = 1, \ldots, |\mathcal{S}|$. Alice then sends the randomly sampled quantum state $\rho_{\mathcal{E}_s}$ to Bob, hoping that Bob can decode the state $\rho_{\mathcal{E}_s}$ to recover her chosen message s. Using the quantum ML model, Bob can produce the function $h_{Q,s}(x)$. Because by assumption the function $h_{Q,s}(x)$ achieves a small average-case prediction error with high probability, and because the packing net has been constructed so that the functions $\{f_{\mathcal{E}_s}\}$ are sufficiently distinguishable, Bob can determine s successfully with high probability. Because Alice chose from among $|\mathcal{S}|$ possible messages, the mutual information of the chosen message s and Bob's measurement outcome must be at least of order $\log|\mathcal{S}|$ bits. According to Holevo's theorem, the Holevo χ quantity of Alice's ensemble Eq. (7.8) upper bounds this mutual information, and therefore must also be $\chi = \Omega(\log|\mathcal{S})|$. Furthermore, we can analyze how χ depends on N_Q, finding that each additional application of $\mathcal{E}_s$ can increase χ by at most $O(m)$. We conclude that $\chi = O(mN_\mathrm{Q})$, yielding the lower bound $N_\mathrm{Q} = \Omega(\log(|\mathcal{S}|)/m)$. The lower bound applies to any quantum ML model, where the size $|\mathcal{S}|$ of the packing net depends on the average-case prediction error ϵ. This completes the first part of the proof.

In the second part, we explicitly construct an ML model in the restricted classical setting that achieves a small average-case prediction error using a modest number of experiments. In this ML model, an input x_i is selected by sampling from the probability distribution $\mathcal{D}$, and an experiment is performed in which the observable O is measured in the output quantum state $\mathcal{E}(|x_i\rangle\langle x_i|)$, obtaining measurement outcome o_i which has expectation value $\mathrm{tr}(O\mathcal{E}(|x_i\rangle\langle x_i|))$. A total of N_C such experiments are conducted. Then, the ML model minimizes the least-squares error to find the best fit within the aforementioned maximal packing net $\mathcal{S}$:

$$h_{\mathrm{C}} = \arg\min_{f \in \mathcal{S}} \frac{1}{N_{\mathrm{C}}} \sum_{i=1}^{N_{\mathrm{C}}} |f(x_i) - o_i|^2. \tag{7.9}$$

Because the measurement outcome o_i fluctuates about the expectation value of O, it may be impossible to achieve zero training error. Yet it is still possible for h_{C} to achieve a small average-case prediction error, potentially even smaller than the training error. We use properties of maximal packing nets and of quantum fluctuations of measurement outcomes to perform a tight statistical analysis of the average-case prediction error, finding that with a high probability,

$$\sum_{x \in \{0,1\}^n} \mathcal{D}(x) \left| h_{\mathrm{C}}(x) - \mathrm{tr}(O\mathcal{E}(|x\rangle\!\langle x|)) \right|^2 = \mathcal{O}(\epsilon), \tag{7.10}$$

provided that N_{C} is of order $\log(|\mathcal{S}|)/\epsilon$.

Finally, we combine the two parts to conclude $N_{\mathrm{C}} = \mathcal{O}(mN_{\mathrm{Q}}/\epsilon)$. The full proof is in Appendix 7.3. □

Theorem 39 shows that all problems that are approximately learnable by a quantum ML model are also approximately learnable by some restricted classical ML model which executes the quantum process $\mathcal{E}$ a comparable number of times. This applies in particular, to predicting outputs of quantum-mechanical processes. The relation $N_{\mathrm{C}} = \mathcal{O}(mN_{\mathrm{Q}}/\epsilon)$ is tight. We give an example in Appendix 7.4 with $N_{\mathrm{C}} = \Omega(mN_{\mathrm{Q}}/\epsilon)$.

For the task of learning classical Boolean circuits, fundamental limits on quantum advantage have been established in previous work (Servedio and Gortler, 2004; C. Zhang, 2010; Arunachalam and Wolf, 2016; Arunachalam and Wolf, 2017; K.-M. Chung and H.-H. Lin, 2018; Arunachalam, Grilo, and Yuen, 2020). Theorem 39 generalizes these existing results to the task of learning outcomes of quantum processes.

7.3 Proof of the information-theoretic bounds

This section contains a thorough treatment of *average* prediction errors. We consider related setups for the classical and quantum learning settings.

The learning problem is defined by a set of CPTP maps $\mathcal{F}$, an input distribution $\mathcal{D}$, and an observable O with $\|O\| \leq 1$. Each CPTP map $\mathcal{E} \in \mathcal{F}$ maps a n-qubit quantum state to m-qubit state. This collection defines a function

$$f_{\mathcal{E}}(x) = \operatorname{tr}(O\mathcal{E}(|x\rangle\langle x|)) : \{0,1\}^n \to \mathbb{R} \tag{7.11}$$

The goal is to learn a function $f : \{0,1\}^n \to \mathbb{R}$ such that with high probability

$$\mathbb{E}_{x\sim\mathcal{D}}|f(x) - f_{\mathcal{E}}(x)|^2 \quad \text{is small.} \tag{7.12}$$

A bit of additional context is appropriate here: we are studying the existence of learning algorithms under a fixed learning problem defined by input distribution $\mathcal{D}$, observable O, and set of CPTP maps $\mathcal{F}$. In turn, the actual learning algorithms may and, in general, will depend on these mathematical objects.

One of our main technical contributions – Theorem 39 – highlights that a substantial quantum advantage is impossible for this setting (small *average-case* prediction error). This is in stark contrast to the setting of achieving small *worst-case* prediction error. The proof consists of two parts. Section 7.3 establishes a lower bound for the query complexity of *any* quantum ML model. Subsequently, Section 7.3 provides an upper bound for the query complexity achieved by *certain* classical ML models. Finally, a combination of these two results establishes Theorem 39, see Section 7.3.

Information-theoretic lower bound for quantum machine learning models

The quantum machine learning model consists of learning phase and prediction phase. In the learning phase, the quantum ML model accesses the quantum experiment characterized by the CPTP map $\mathcal{E}$ for N_{Q} times to learn a model. We consider the quantum ML model to be a mixed state quantum computation algorithm (a generalization of unitary quantum computation). Starting point is an initial state ρ_0 on any number of qubits. Subsequently, arbitrary quantum operations C_t (CPTP maps) are interleaved with in total N_{Q} invocations $\mathcal{E}\otimes\mathcal{I}$ of the unknown (black box) CPTP map and produce a final state

$$\rho_{\mathcal{E}} = C_{N_{\mathrm{Q}}}(\mathcal{E}\otimes\mathcal{I})C_{N_{\mathrm{Q}}-1}\ldots C_1(\mathcal{E}\otimes\mathcal{I})(\rho_0). \tag{7.13}$$

In this model, we can assume without loss that $\mathcal{E}$ always acts on the first n qubits, because the quantum operations C_t are unrestricted. In particular, they could contain certain SWAP operations that permute the qubits around. The final state $\rho_{\mathcal{E}}$ is the quantum memory that stores the prediction model learned from the CPTP map $\mathcal{E}$ using the quantum ML algorithm. Obtaining $\rho_{\mathcal{E}}$ concludes the quantum learning phase.

In the prediction phase, we assume that new inputs are provided as part of a sequence

$$x_1, x_2, x_3, \ldots \in \{0,1\}^n. \tag{7.14}$$

For each sequence member x_i, the quantum ML model accesses the input x_i, as well as the current quantum memory. It produces an outcome by performing a POVM measurement on the quantum memory $\rho_{\mathcal{E}}$. We emphasize that this can, and in general will, affect the quantum memory nontrivially. The quantum ML outputs $h_{\mathrm{Q}}(x_i)$ depend on the entire sequence $x_1, \ldots, x_i$. And different sequence orderings will produce different predictions. For example, when $n = 2$, the following ordering may result in the prediction

$$\begin{aligned} x_1 = 00, x_2 = 01, x_3 = 10, x_4 = 11 & \qquad (7.15)\\ \rightarrow \;\; h_{\mathrm{Q}}(00) = -0.3, h_{\mathrm{Q}}(01) = 0.5, h_{\mathrm{Q}}(10) = 0.2, h_{\mathrm{Q}}(11) = -0.7, & \qquad (7.16) \end{aligned}$$

but a different ordering may result in a slightly different prediction, such as

$$\begin{aligned} x_1 = 11, x_2 = 01, x_3 = 10, x_4 = 00 & \qquad (7.17)\\ \rightarrow \;\; h_{\mathrm{Q}}(00) = -0.2, h_{\mathrm{Q}}(01) = 0.5, h_{\mathrm{Q}}(10) = 0.2, h_{\mathrm{Q}}(11) = -0.6. & \qquad (7.18) \end{aligned}$$

Also, note that $h_{\mathrm{Q}}(x_i)$ can be randomized because a quantum measurement is performed to produce the prediction outcome. The ordering does not affect the theorem we want to prove. In the following, we will fix the input ordering to be an arbitrary ordering. For example, we can use the input ordering such that the quantum ML model has the smallest prediction error.

After fixing an input ordering, we can treat the entire prediction phase (taking a sequence of inputs $x_1, x_2, \ldots$ and producing $h_{\mathrm{Q}}(x_1), h_{\mathrm{Q}}(x_2), \ldots$) as an enormous POVM measurement on the output state $\rho_{\mathcal{E}}$ obtained from the learning phase. Each outcome a from the enormous POVM measurement on the output state $\rho_{\mathcal{E}}$ corresponds to a function $h_{\mathrm{Q},a}(x) : \{0,1\}^n \rightarrow \mathbb{R}$. Using Naimark's dilation theorem, every POVM measurement is a projective measurements on a larger Hilbert space. Since the quantum memory that the quantum ML model can operate on contains an arbitrary amount of qubits, we can use Naimark's dilation theorem to restrict the enormous POVM measurement to a projective measurement $\{P_a\}_a$. Hence, for any CPTP map $\mathcal{E} \in \mathcal{F}$, when we ask the quantum ML model to produce the prediction for an ordering of inputs $x_1, x_2, \ldots$, the output values $h_{\mathrm{Q}}(x_1), h_{\mathrm{Q}}(x_2), \ldots$ will be given by

$$h_{\mathrm{Q},a}(x) \text{ w. pr. } \operatorname{tr}(P_a \rho_{\mathcal{E}}) = \tag{7.19}$$

$$\mathrm{tr}(P_a C_{N_\mathrm{Q}}(\mathcal{E} \otimes \mathcal{I}) C_{N_\mathrm{Q}-1} \dots C_2(\mathcal{E} \otimes \mathcal{I}) C_1(\mathcal{E} \otimes \mathcal{I}) C_0(\rho_0)), \tag{7.20}$$

for a projective measurement $\{P_a\}_a$ with $\sum_a P_a = I$.

Finally, we will assume that the produced function $h_\mathrm{Q}(x)$ achieves small prediction error

$$\mathop{\mathbb{E}}_{x \sim \mathcal{D}} \left| h_\mathrm{Q}(x) - \mathrm{tr}(O\mathcal{E}(|x\rangle\langle x|)) \right|^2 \leq \epsilon \quad \text{with probability at least } 2/3. \tag{7.21}$$

for any CPTP map $\mathcal{E} \in \mathcal{F}$. This assumption asserts that

$$\sum_{a=1}^{A} \mathrm{tr}(P_a \rho_\mathcal{E}) \mathbb{1} \left[\mathop{\mathbb{E}}_{x \sim \mathcal{D}} \left| h_{\mathrm{Q},a}(x) - \mathrm{tr}(O\mathcal{E}(|x\rangle\langle x|)) \right|^2 \leq \epsilon \right] \geq 2/3, \tag{7.22}$$

where $\mathbb{1}[z]$ denotes the indicator function of event z. That is, $\mathbb{1}[z] = 1$ if z is true and $\mathbb{1}[z] = 0$ otherwise.

Maximal packing net

We emphasize that Rel. (7.22) must be valid for *any* $\mathcal{E} \in \mathcal{F}$. Because we only need to output a function $h_\mathrm{Q}(x)$ that approximates $f(x) = \mathrm{tr}(O\mathcal{E}(|x\rangle\langle x|))$ on average, the task will not be hard when there are only a few qualitatively different CPTP maps in $\mathcal{F}$. However, the problem could become harder when $\mathcal{F}$ contains a large amount of very different CPTP maps. The task is now to transform this requirement into a stringent lower bound on N_Q – the number of black-box uses of the unknown CPTP map $\mathcal{E} \otimes \mathcal{I}$ within the quantum computation (7.13). As a starting point, we equip the set of target functions $\mathcal{F}_f = \{f_\mathcal{E}(x) = \mathrm{tr}(O\mathcal{E}(|x\rangle\langle x|)) | \mathcal{E} \in \mathcal{F}\}$ with a packing net. Packing nets are discrete subsets whose elements are guaranteed to have a certain minimal pairwise distance (think of spheres that must not overlap with each other). We choose points (functions) $f_{\mathcal{E}_i} \in \mathcal{F}_f$ and demand

$$\mathop{\mathbb{E}}_{x \sim \mathcal{D}} |f_{\mathcal{E}_i}(x) - f_{\mathcal{E}_j}(x)|^2 > 4\epsilon \quad \text{whenever} \quad i \neq j. \tag{7.23}$$

We denote the resulting packing net of $\mathcal{F}_f$ by $M^p_{4\epsilon}(\mathcal{F}_f)$ and note that every such set has finitely many elements ($\mathcal{F}_f$ is a compact set). We also assume that $M^p_{4\epsilon}(\mathcal{F}_f)$ is maximal in the sense that no other 4ϵ-packing net can contain more points (functions).

It is possible to utilize packing nets to derive a query complexity lower bound for the quantum machine learning model. In fact, we will present two different proof strategies. The first proof is inspired by (Steven T Flammia et al., 2012; Haah et al., 2017; Huang, Richard Kueng, and Preskill, 2020) and analyzes a communication protocol. The second proof uses a proof technique that depends on an analysis of polynomials similar to (Beals et al., 2001). While it is somewhat weaker than the information-theoretic bound in the first proof, we include the derivation for completeness as we believe that it may be insightful for the interested reader.

Proof strategy I: mutual information analysis

Let us define a communication protocol between two parties, say Alice and Bob. They use the packing net $M^p_{4\epsilon}(\mathcal{F}_f)$ as a dictionary to communicate randomly selected classical messages. More precisely, Alice samples an integer X uniformly at random from $1, 2, \ldots, |M^p_{4\epsilon}(\mathcal{F}_f)|$ and chooses the corresponding CPTP map $\mathcal{E}_X \in M^p_{4\epsilon}(\mathcal{F}_f)$. When Bob wants to access the unknown CPTP map $\mathcal{E}_X$, he will ask Alice to apply the CPTP map $\mathcal{E}_X$. Bob will then execute the quantum machine learning model (7.13) to obtain a prediction model $h_{\mathrm{Q},a}(x)$, where a parameterizes the prediction model. Subsequently, Bob solves the following optimization problem

$$\tilde{X} = \underset{X'=1,\ldots,|M^p_{4\epsilon}(\mathcal{F}_f)|}{\arg\min} \mathop{\mathbb{E}}_{x\sim\mathcal{D}} \left| h_{\mathrm{Q},a}(x) - \mathrm{tr}(O\mathcal{E}_{X'}(|x\rangle\langle x|)) \right|^2 \tag{7.24}$$

to obtain an integer $\tilde{X}$. This decoding procedure seems adequate, provided that the prediction model h_{Q} approximately reproduces the true underlying function. More precisely, assumption (7.21) asserts

$$\mathop{\mathbb{E}}_{x\sim\mathcal{D}} \left| h_{\mathrm{Q},a}(x) - \mathrm{tr}(O\mathcal{E}_X(|x\rangle\langle x|)) \right|^2 \leq \epsilon \quad \text{with probability at least } 2/3. \tag{7.25}$$

Here is where the choice of dictionary matters: $M^p_{4\epsilon}(\mathcal{F}_f)$ is a packing net, see Equation (7.23). For $X' \neq X$ this necessarily implies

$$\mathop{\mathbb{E}}_{X\sim\mathcal{D}} \left| h_{\mathrm{Q},a}(x) - f_{\mathcal{E}_{X'}}(x) \right|^2 \tag{7.26}$$

$$\geq \left(\left(\mathop{\mathbb{E}}_{X\sim\mathcal{D}} \left| f_{\mathcal{E}_X}(x) - f_{\mathcal{E}_{X'}}(x) \right|^2 \right)^{1/2} - \left(\mathop{\mathbb{E}}_{X\sim\mathcal{D}} \left| h_{\mathrm{Q},a}(x) - f_{\mathcal{E}_X}(x) \right|^2 \right)^{1/2} \right)^2 \tag{7.27}$$

$$> \left(2\sqrt{\epsilon} - \sqrt{\epsilon}\right)^2 = \epsilon. \tag{7.28}$$

This allows us to conclude that Bob's decoding strategy (7.24) succeeds perfectly if

$$\mathop{\mathbb{E}}_{x\sim\mathcal{D}} \left| h_{\mathrm{Q},a}(x) - \mathrm{tr}(O\mathcal{E}_X(|x\rangle\langle x|)) \right|^2 \leq \epsilon. \tag{7.29}$$

In turn, Assumption 7.21 ensures $\tilde{X} = X$ (perfect decoding) with probability at least 2/3.

Now, we use the fact that Alice samples her message X uniformly at random from a total of $|M^{p}_{4\epsilon}(\mathcal{F}_f)|$ integers. Because $\tilde{X} = X$ (perfect decoding) with probability at least $2/3$, Fano's inequality implies that

$$H(X|\tilde{X}) \leq H(1/3) + \log(|M^{p}_{4\epsilon}(\mathcal{F}_f)|)/3, \tag{7.30}$$

where $H(x) = -x\log x - (1-x)\log(1-x)$ is the binary entropy. This gives a lower bound on the mutual information between sent and decoded message, namely

$$I(X:\tilde{X}) = H(X) - H(X|\tilde{X}) \geq \frac{2}{3}\log(|M^{p}_{4\epsilon}(\mathcal{F}_f)|) - H(1/3) = \Omega\left(\log(|M^{p}_{4\epsilon}(\mathcal{F}_f)|)\right). \tag{7.31}$$

Next, note that $\tilde{X}$ is obtained by classically processing a measurement outcome a of the quantum state $\rho_{\mathcal{E}_X}$ The data processing inequality and Holevo's theorem (Alexander Semenovich Holevo, 1973; Horodecki et al., 2009; Bengtsson and Życzkowski, 2017; Araki and Elliott H Lieb, 2002) then imply

$$I(X:\tilde{X}) \leq I(X:a) \leq \chi(X:\rho_{\mathcal{E}_X}). \tag{7.32}$$

The Holevo χ quantity between the classical random variable X and the quantum state $\rho_{\mathcal{E}_X}$ is

$$\chi(X:\rho_{\mathcal{E}_X}) = S\left(\mathop{\mathbb{E}}_{X}\rho_{\mathcal{E}_X}\right) - \mathop{\mathbb{E}}_{X} S\left(\rho_{\mathcal{E}_X}\right), \tag{7.33}$$

where $S(\rho) = \mathrm{tr}(-\rho\log\rho)$ is the von Neumann entropy. Throughout this work, we refer to log with base e. Recall that Bob produces $\rho_{\mathcal{E}_X}$ by utilizing a total of N_{Q} channel copies obtained from Alice. We can use the specific layout (7.13) of Bob's quantum computation to produce an upper bound on the Holevo-χ:

$$\chi(X:\rho_{\mathcal{E}_X}) \leq O(mN_{\mathrm{Q}}) \tag{7.34}$$

This bound follows from induction over a sample-resolved variant of Bob's quantum computation. For $t = 0, 1, \ldots, N_{\mathrm{Q}}$, we will show that

$$\rho^{t}_{\mathcal{E}} = C_t(\mathcal{E}\otimes\mathcal{I})C_{t-1}\ldots C_1(\mathcal{E}\otimes\mathcal{I})C_0(\rho_0) \quad \text{obeys} \quad \chi(X:\rho^{t}_{\mathcal{E}_X}) \leq (2\log 2)mt. \tag{7.35}$$

Bound (7.34) then follows from recognizing that setting $t = N_{\mathrm{Q}}$ reproduces Bob's complete computation, see Equation (7.13).

The base case ($t = 0$) is simple, because $\rho^{0}_{\mathcal{E}_X} = C_0(\rho_0)$ does not depend on X at all. This ensures

$$\chi\left(X:\rho^{0}_{\mathcal{E}_X}\right) = S\left(\mathop{\mathbb{E}}_{X}\rho^{0}_{\mathcal{E}_X}\right) - \mathop{\mathbb{E}}_{X} S\left(\rho^{0}_{\mathcal{E}_X}\right) \tag{7.36}$$

$$= S\left(\rho^0_{\mathcal{E}_X}\right) - S\left(\rho^0_{\mathcal{E}_X}\right) = 0 \leq (2\log 2)mt \quad (t=0). \tag{7.37}$$

Now, let us move to the induction step ($t > 0$). The induction hypothesis provides us with

$$\chi(X : \rho^{t-1}_{\mathcal{E}_X}) \leq (2\log 2)m(t-1) \tag{7.38}$$

and we must relate $\chi(X : \rho^t_{\mathcal{E}_X})$ to $\chi(X : \rho^{t-1}_{\mathcal{E}_X})$. To achieve this goal, we use the fact that the Holevo-χ is closely related to the quantum relative entropy $D(\rho||\sigma) = \mathrm{tr}\,(\rho(\log\rho - \log\sigma))$ (Horodecki et al., 2009; Bengtsson and Życzkowski, 2017; Araki and Elliott H Lieb, 2002). Indeed,

$$\chi(X : \rho^t_{\mathcal{E}_X}) = \mathop{\mathbb{E}}_X\left[\mathrm{tr}\left(\rho^t_{\mathcal{E}_X}\log\rho^t_{cE_X} - \rho^t_{\mathcal{E}_X}\log\left(\mathop{\mathbb{E}}_{X'}\rho^t_{cE_{X'}}\right)\right)\right] = \mathop{\mathbb{E}}_X D\left(\rho^t_{\mathcal{E}_X}||\mathop{\mathbb{E}}_{X'}\rho^t_{cE_{X'}}\right), \tag{7.39}$$

and monotonicity of the quantum relative entropy asserts

$$\begin{aligned}
\mathop{\mathbb{E}}_X D\left(\rho^t_{\mathcal{E}_X}||\mathop{\mathbb{E}}_{X'}\rho^t_{cE_{X'}}\right) =& \mathop{\mathbb{E}}_X D\left(C_t\left((\mathcal{E}_X\otimes\mathcal{I})\left(\rho^{t-1}_{\mathcal{E}_X}\right)\right) \ || \ C_t\left(\mathop{\mathbb{E}}_{X'}(\mathcal{E}_{X'}\otimes\mathcal{I})\left(\rho^{t-1}_{\mathcal{E}_{X'}}\right)\right)\right) \\
\leq& \mathop{\mathbb{E}}_X D\left((\mathcal{E}_X\otimes\mathcal{I})\left(\rho^{t-1}_{\mathcal{E}_X}\right) \ || \ \mathop{\mathbb{E}}_{X'}(\mathcal{E}_{X'}\otimes\mathcal{I})\left(\rho^{t-1}_{\mathcal{E}_{X'}}\right)\right) \\
=& S\left(\mathop{\mathbb{E}}_X(\mathcal{E}_X\otimes\mathcal{I})\,(\rho^{t-1}_{\mathcal{E}_X})\right) - \mathop{\mathbb{E}}_X S\left((\mathcal{E}_X\otimes\mathcal{I})\,(\rho^{t-1}_{\mathcal{E}_X})\right).
\end{aligned}$$

This effectively allows us to ignore the t-th quantum operation C_t and instead exposes the t-th invocation of $\mathcal{E}\otimes\mathcal{I}$.

We analyze the two remaining terms separately. Let use define the notation $\mathrm{tr}_{\leq m}$ as the partial trace over the first m qubits, and $\mathrm{tr}_{>m}$ as the partial trace over the rest of the qubits. Subadditivity of the von Neumann entropy $S(\rho)$ (Horodecki et al., 2009; Bengtsson and Życzkowski, 2017; Araki and Elliott H Lieb, 2002) implies

$$S\left(\mathop{\mathbb{E}}_X(\mathcal{E}_X\otimes\mathcal{I})\,(\rho^{t-1}_{\mathcal{E}_X})\right) \tag{7.40}$$

$$\leq S\left(\mathrm{tr}_{\leq m}\mathop{\mathbb{E}}_X(\mathcal{E}_X\otimes\mathcal{I})\,(\rho^{t-1}_{\mathcal{E}_X})\right) + S\left(\mathrm{tr}_{>m}\mathop{\mathbb{E}}_X(\mathcal{E}_X\otimes\mathcal{I})\,(\rho^{t-1}_{\mathcal{E}_X})\right), \tag{7.41}$$

$$\leq S\left(\mathrm{tr}_{\leq m}\mathop{\mathbb{E}}_X(\mathcal{E}_X\otimes\mathcal{I})\,(\rho^{t-1}_{\mathcal{E}_X})\right) + m\log 2 \tag{7.42}$$

$$= S\left(\mathrm{tr}_{\leq n}\mathop{\mathbb{E}}_X\rho^{t-1}_{\mathcal{E}_X}\right) + m\log 2. \tag{7.43}$$

The second inequality uses the fact that the maximum entropy for an m-qubit system is at most $m\log 2$. The last equality is due to the following technical observation (the action of a CPTP map can be traced out).

Lemma 47. *Fix a CPTP map $\mathcal{E}$ from n qubits to m qubits and let $\mathcal{I}$ denote the identity map on $n' \geq 0$ qubits. Then,* $\mathrm{tr}_{\leq m}[(\mathcal{E} \otimes \mathcal{I})\rho] = \mathrm{tr}_{\leq n}[\rho]$ *for any $(n+n')$-qubit state ρ.*

Proof. Let $\mathcal{E}(\rho) = \sum_i K_i \rho K_i^\dagger$ be a Kraus representation of the CP (completely-positive) map $\mathcal{E}$. TP (trace-preserving) moreover implies $\sum_i K_i^\dagger K_i = I$. For any input state ρ, Linearity and (partial) cyclicity of the partial trace then ensure

$$\begin{aligned}\mathrm{tr}_{\leq m}\left((\mathcal{E} \otimes \mathcal{I})\rho\right) &= \sum_i \mathrm{tr}_{\leq m}\left(K_i \otimes I \rho K_i^\dagger \otimes I\right) \\ = \sum_i \mathrm{tr}_{\leq m}\left(\rho (K_i^\dagger K_i) \otimes I\right) &= \mathrm{tr}_{\leq m}\left(\rho I \otimes I\right) = \mathrm{tr}_{\leq m}\left(\rho\right).\end{aligned}$$

This concludes the proof of the lemma. □

Similarly, the second term can be lower bounded by

$$\begin{aligned}&\mathop{\mathbb{E}}_X S\left((\mathcal{E}_X \otimes \mathcal{I})(\rho_{\mathcal{E}_X}^{t-1})\right) && (7.44)\\ &\geq \mathop{\mathbb{E}}_X S\left(\mathrm{tr}_{\leq m}(\mathcal{E}_X \otimes \mathcal{I})(\rho_{\mathcal{E}_X}^{t-1})\right) - \mathop{\mathbb{E}}_X S\left(\mathrm{tr}_{>m}(\mathcal{E}_X \otimes \mathcal{I})(\rho_{\mathcal{E}_X}^{t-1})\right), && (7.45)\\ &\geq \mathop{\mathbb{E}}_X S\left(\mathrm{tr}_{\leq m}(\mathcal{E}_X \otimes \mathcal{I})(\rho_{\mathcal{E}_X}^{t-1})\right) - m\log 2, && (7.46)\\ &= \mathop{\mathbb{E}}_X S\left(\mathrm{tr}_{\leq n}\rho_{\mathcal{E}_X}^{t-1}\right) - m\log 2. && (7.47)\end{aligned}$$

We can combine these two bounds with the monotonicity of the quantum relative entropy to obtain

$$\begin{aligned}\chi(X:\rho_{\mathcal{E}_X}^{t}) &\leq S\left(\mathop{\mathbb{E}}_X (\mathcal{E}_X \otimes \mathcal{I})(\rho_{\mathcal{E}_X}^{t-1})\right) - \mathop{\mathbb{E}}_X S\left((\mathcal{E}_X \otimes \mathcal{I})(\rho_{\mathcal{E}_X}^{t-1})\right) && (7.48)\\ &\leq S\left(\mathrm{tr}_{\leq n}\mathop{\mathbb{E}}_X \rho_{\mathcal{E}_X}^{t-1}\right) - \mathop{\mathbb{E}}_X S\left(\mathrm{tr}_{\leq n}\rho_{\mathcal{E}_X}^{t-1}\right) + (2\log 2)m && (7.49)\\ &= \mathop{\mathbb{E}}_X D\left(\mathrm{tr}_{\leq n}\rho_{\mathcal{E}_X}^{t-1} \,||\, \mathrm{tr}_{\leq n}\mathop{\mathbb{E}}_{X'}\rho_{\mathcal{E}_{X'}}^{t-1}\right) + (2\log 2)m && (7.50)\\ &\leq \mathop{\mathbb{E}}_X D\left(\rho_{\mathcal{E}_X}^{t-1} \,||\, \mathop{\mathbb{E}}_{X'}\rho_{\mathcal{E}_{X'}}^{t-1}\right) + (2\log 2)m && (7.51)\\ &= \chi(X:\rho_{\mathcal{E}_X}^{t-1}) + (2\log 2)m. && (7.52)\end{aligned}$$

Plug in the induction hypothesis (7.38) to complete the argument:

$$\begin{aligned}\chi(X:\rho_{\mathcal{E}_X}^{t}) &\leq \chi(X:\rho_{\mathcal{E}_X}^{t-1}) + (2\log 2)m && (7.53)\\ &\leq (2\log 2)m(t-1) + (2\log 2)m = (2\log 2)mt. && (7.54)\end{aligned}$$

It is worthwhile to pause and recapitulate the main insights from this section: i.) a lower bound on the mutual information in terms of packing net cardinality, see Equation (7.31); ii.) Holevo's theorem, see Equation (7.32); and, (iii) an upper bound on the Holevo-χ in terms of query complexity, see Equation (7.54) for $t = N_{\mathrm{Q}}$. Combining all of them yields

$$\Omega\left(\log(|M^p_{4\epsilon}(\mathcal{F}_f)|)\right) \leq I\left(X : \tilde{X}\right) \leq \chi\left(X : \rho_{\mathcal{E}_X}\right) \leq (2\log 2) m N_{\mathrm{Q}}.$$

Rearranging this display yields a lower bound on the minimal query complexity in terms of packing net size:

$$N_{\mathrm{Q}} \geq \Omega\left(\frac{\log(|M^p_{4\epsilon}(\mathcal{F}_f)|)}{m}\right). \tag{7.55}$$

Proof strategy II: polynomial method

The second proof uses a proof technique that depends on analysis of polynomials (Beals et al., 2001). It leads to somewhat weaker results that only apply if $m \leq n$. We include this derivation for completeness as we believe that it may be insightful for the interested reader.

Let us start by recalling that we may embed a 4ϵ-packing net $M^p_{4\epsilon}(\mathcal{F}_f)$ within the set of target functions $\mathcal{F}_f$. Geometrically, this means that each $\mathcal{E} \in M^p_{4\epsilon}(\mathcal{F}_f)$ describes the center of a 2ϵ-ball (this radius is defined with respect to average prediction error squared). And, according to the defining property Equation (7.23), these balls do not overlap. We can use these disjoint balls to cluster different quantum machine learning solutions. Define

$$\mathcal{F}^{\mathrm{Q}}_{\mathcal{E}} = \left\{ a \in A \ : \ \mathop{\mathbb{E}}_{x\sim\mathcal{D}} \left|h_{\mathrm{Q},a}(x) - f_{\mathcal{E}}(x)\right|^2 \leq \epsilon \right\}, \tag{7.56}$$

where A is a placeholder for all possible answers the quantum machine learning model can provide. See the definition given in Equation (7.19). The packing net condition (7.23) ensures that different clusters are completely disjoint. For distinct $\mathcal{E}_1, \mathcal{E}_2 \in \mathcal{F}$ and $a_1 \in \mathcal{F}^{\mathrm{Q}}_{\mathcal{E}_1}$, $a_2 \in \mathcal{F}^{\mathrm{Q}}_{\mathcal{E}_2}$, two triangle inequalities and Equation (7.23) yield

$$\sqrt{\mathop{\mathbb{E}}_{x\sim\mathcal{D}} \left|h_{\mathrm{Q},a_1}(x) - h_{\mathrm{Q},a_2}(x)\right|^2} \tag{7.57}$$

$$\geq \sqrt{\mathop{\mathbb{E}}_{x\sim\mathcal{D}} \left|f_{\mathcal{E}_1}(x) - h_{\mathrm{Q},a_2}(x)\right|^2} - \sqrt{\mathop{\mathbb{E}}_{x\sim\mathcal{D}} \left|h_{\mathrm{Q},a_1}(x) - f_{\mathcal{E}_1}(x)\right|^2} \tag{7.58}$$

$$\geq \sqrt{\mathop{\mathbb{E}}_{x\sim\mathcal{D}} \left|f_{\mathcal{E}_1}(x) - f_{\mathcal{E}_2}(x)\right|^2} - \sqrt{\mathop{\mathbb{E}}_{x\sim\mathcal{D}} \left|h_{\mathrm{Q},a_2}(x) - f_{\mathcal{E}_2}(x)\right|^2} \tag{7.59}$$

$$- \sqrt{\mathop{\mathbb{E}}_{x\sim\mathcal{D}} \left|h_{\mathrm{Q},a_1}(x) - f_{\mathcal{E}_1}(x)\right|^2} \tag{7.60}$$

$$> 2\sqrt{\epsilon} - \sqrt{\epsilon} - \sqrt{\epsilon} = 0. \tag{7.61}$$

This implies $f_{a_1}^Q \neq f_{a_2}^Q$ and, more importantly, $\mathcal{F}_{\mathcal{E}_1}^Q \cap \mathcal{F}_{\mathcal{E}_2}^Q = \emptyset$ whenever $f_{\mathcal{E}_1} \neq f_{\mathcal{E}_2}$.

We will use this insight to reason about an auxiliar matrix P of size $|M_{4\epsilon}^p(\mathcal{F}_f)| \times |M_{4\epsilon}^p(\mathcal{F}_f)|$. We label rows and columns by packing net elements $f_{\mathcal{E}_i}$ with $i = 1$ (rows) or $i = 2$ (columns). For each pair $f_{\mathcal{E}_1}, f_{\mathcal{E}_2}$, let $P_{\mathcal{E}_1,\mathcal{E}_2}$ denote the probability of a mix-up between $\mathcal{E}_1$ and $\mathcal{E}_2$. Such mix-ups occur if the underlying CPTP map is $\mathcal{E}_2$, but the quantum ML model outputs an answer $a \in \mathcal{F}_{\mathcal{E}_1}^{\mathrm{Q}}$ that belongs to the cluster associated with $\mathcal{E}_1$:

$$P_{\mathcal{E}_1,\mathcal{E}_2} = \sum_{a=1}^{A} \mathrm{tr}(P_a \rho_{\mathcal{E}_2}) \mathbb{1}\left[\mathop{\mathbb{E}}_{x\sim\mathcal{D}} \left|h_{\mathrm{Q},a}(x) - f_{\mathcal{E}_1}(x)\right|^2 \leq \epsilon\right] = \sum_{a:h_{\mathrm{Q},a}\in\mathcal{F}_{\mathcal{E}_1}^Q} \mathrm{tr}(P_a \rho_{\mathcal{E}_2}). \tag{7.62}$$

Here, ρ_{ϵ_2} is the outcome state of the quantum ML model (trained on CPTP map $\mathcal{E}_2$) and P_a is the POVM element associated with predicting a_1. Recall that the main assumption on the quantum ML model is that it predicts accurately with probability at least $2/3$. This implies

$$P_{\mathcal{E}_1,\mathcal{E}_1} \geq 2/3 \quad \text{for each} \quad \mathcal{E}_1 \in M_{4\epsilon}^p(\mathcal{F}_f), \tag{7.63}$$

while each row sum over off-diagonal matrix elements is strictly smaller. For $f_{\mathcal{E}_2} \in M_{4\epsilon}^p(\mathcal{F}_f)$,

$$\sum_{\substack{f_{\mathcal{E}_1}\in M_{4\epsilon}^p(\mathcal{F}_f)\\ f_{\mathcal{E}_1}\neq f_{\mathcal{E}_2}}} P_{\mathcal{E}_1,\mathcal{E}_2} = \sum_{\substack{f_{\mathcal{E}_1}\in M_{4\epsilon}^p(\mathcal{F}_f)\\ f_{\mathcal{E}_1}\neq f_{\mathcal{E}_2}}} \sum_{a:h_{\mathrm{Q},a}\in\mathcal{F}_{\mathcal{E}_1}^Q} \mathrm{tr}(P_a \rho_{\mathcal{E}_2}) \tag{7.64}$$

$$= \sum_{f_{\mathcal{E}_1}\in M_{4\epsilon}^p(\mathcal{F}_f)} \sum_{a:h_{\mathrm{Q},a}\in\mathcal{F}_{\mathcal{E}_1}^Q} \mathrm{tr}(P_a \rho_{\mathcal{E}_2}) - \sum_{a:h_{\mathrm{Q},a}\in\mathcal{F}_{\mathcal{E}_2}^Q} \mathrm{tr}(P_a \rho_{\mathcal{E}_2}) \tag{7.65}$$

$$= \sum_{\substack{a:\exists f_{\mathcal{E}_1}\in M_{4\epsilon}^p(\mathcal{F}_f)\\ h_{\mathrm{Q},a}\in\mathcal{F}_{\mathcal{E}_1}^Q}} \mathrm{tr}(P_a \rho_{\mathcal{E}_2}) - \sum_{a:h_{\mathrm{Q},a}\in\mathcal{F}_{\mathcal{E}_2}^Q} \mathrm{tr}(P_a \rho_{\mathcal{E}_2}) \tag{7.66}$$

$$\leq \sum_{a=1}^{A} \mathrm{tr}(P_a \rho_{\mathcal{E}_2}) - \sum_{a:h_{\mathrm{Q},a}\in\mathcal{F}_{\mathcal{E}_2}^Q} \mathrm{tr}(P_a \rho_{\mathcal{E}_2}) \tag{7.67}$$

$$= 1 - P_{\mathcal{E}_2,\mathcal{E}_2} \leq 1/3. \tag{7.68}$$

The first equality uses the definition of matrix P. The third equality follows from the observation that distinct clusters are also disjoint ($\mathcal{F}^Q_{\mathcal{E}_1} \cap \mathcal{F}^Q_{\mathcal{E}_2} = \emptyset$). We conclude that the $|M^p_{4\epsilon}(\mathcal{F}_f)| \times |M^p_{4\epsilon}(\mathcal{F}_f)|$-matrix P is diagonally dominant. Such matrices are guaranteed to be non-singular, i.e. they have full rank.

This is a suitable starting point for analyzing the probability $\mathrm{tr}(P_a \rho_{\mathcal{E}})$ via a polynomial method (Beals et al., 2001). Let $\{K_i^{\mathcal{E}}\}_{i=1}^{2^n 2^m}$,

$$\mathcal{E}(\rho) = \sum_i K_i^{\mathcal{E}} \rho (K_i^{\mathcal{E}})^\dagger, \tag{7.69}$$

with $K_i^{\mathcal{E}} \in \mathbb{C}^{2^m \times 2^n}$ be the Kraus representation of a fixed CPTP map $\mathcal{E} \in \mathcal{F}$. This representation is parametrized by (at most) $2^n 2^m \times 2^m 2^n = 2^{2(n+m)}$ complex parameters:

$$(K_i^{\mathcal{E}})_{jk} = z^{\mathcal{E}}_{i \times 2^{n+m} + j \times 2^n + k} \quad \text{defines} \quad z^{\mathcal{E}} \in \mathbb{C}^{2^{2(n+m)}}.$$

On a high level, we parametrize inputs to the quantum ML model by vectors. After training, the probability of obtaining answer $a \in A$ corresponds to a homogeneous polynomial of degree N_{Q} in $z^{\mathcal{E}}$ and of degree N_{Q} in $\bar{z}^{\mathcal{E}}$:

$$\mathrm{tr}(P_a \rho_{\mathcal{E}}) = \mathrm{tr}(P_a \mathcal{C}_{N_{\mathrm{Q}}} (\mathcal{E} \otimes \mathcal{I}) \mathcal{C}_{N_{\mathrm{Q}}-1} \ldots \mathcal{C}_2 (\mathcal{E} \otimes \mathcal{I}) \mathcal{C}_1 (\mathcal{E} \otimes \mathcal{I}) \mathcal{C}_0(\rho_0)) \tag{7.70}$$

$$= w_a^\dagger \underbrace{(z^{\mathcal{E}} \otimes \bar{z}^{\mathcal{E}}) \otimes \cdots \otimes (z^{\mathcal{E}} \otimes \bar{z}^{\mathcal{E}})}_{N_{\mathrm{Q}} \text{ times}}, \tag{7.71}$$

where $w_a^\dagger$ is a dual tensor product vector with compatible dimension

$$N = 2^{2(n+m)2N_{\mathrm{Q}}} = 2^{4(n+m)N_{\mathrm{Q}}}. \tag{7.72}$$

Every matrix element $P_{\mathcal{E}_1,\mathcal{E}_2}$ of P defined in Equation (7.62) can be expressed as a sum of homogeneous polynoials in $(z^{\mathcal{E}_2} \otimes \bar{z}^{\mathcal{E}_2}) \otimes \cdots \otimes (z^{\mathcal{E}_2} \otimes \bar{z}^{\mathcal{E}_2})$. Collecting all $M = |M^p_{4\epsilon}(\mathcal{F}_f)|$ possible tensor products as rows of the matrix

$$Z = \begin{bmatrix} (z^{\mathcal{E}_1} \otimes \bar{z}^{\mathcal{E}_1}) \otimes \cdots \otimes (z^{\mathcal{E}_1} \otimes \bar{z}^{\mathcal{E}_1}) & \cdots & (z^{\mathcal{E}_M} \otimes \bar{z}^{\mathcal{E}_M}) \otimes \cdots \otimes (z^{\mathcal{E}_M} \otimes \bar{z}^{\mathcal{E}_M}) \end{bmatrix} \tag{7.73}$$

$$\in \mathbb{C}^{N \times M} \tag{7.74}$$

allows us to present the multilinear characterization of all entries of P in a single display:

$$P = WZ \quad \text{with} \quad W = \begin{bmatrix} \sum_{h_{\mathrm{Q},a} \in \mathcal{F}^Q_{\mathcal{E}_1}} w_a^\dagger \\ \vdots \\ \sum_{h_{\mathrm{Q},a} \in \mathcal{F}^Q_{\mathcal{E}_M}} w_a^\dagger \end{bmatrix} \in \mathbb{C}^{M \times 2^N}$$

Above, we have shown that the $M \times M$-matrix P must have full column rank. This is only possible if

$$|M^p_{4\epsilon}(\mathcal{F}_f)| = M \leq N = 2^{2(n+m)2N_{\mathrm{Q}}} = 2^{4(n+m)N_{\mathrm{Q}}}. \tag{7.75}$$

Rearranging these terms and assuming $n \leq m$ implies the following lower bound on quantum query complexity N_{Q}:

$$N_{\mathrm{Q}} \geq \frac{\log(|M^p_{4\epsilon}(\mathcal{F}_f)|)}{4(n+m)} \geq \frac{\log(|M^p_{4\epsilon}(\mathcal{F}_f)|)}{8m}. \tag{7.76}$$

Information-theoretic upper bound for restricted classical machine learning models

We will focus on restricted classical ML models that can select inputs $x_i \in \{0,1\}^n$ and obtain the corresponding outcome $o_i \in \mathbb{R}$. This outcome is obtained by performing a single-shot measurement (the projective measurement given by the eigenbasis of O) of observable O on the output quantum state $\mathcal{E}(|x_i\rangle\langle x_i|)$. This ensures

$$\mathbb{E}[o_i] = \mathrm{tr}(O\mathcal{E}(|x_i\rangle\langle x_i|) = f_{\mathcal{E}}(x_i) \quad \text{and, moreover,} \quad |o_i| \leq 1 \text{ with probability one,} \tag{7.77}$$

because observables are bounded in spectral norm ($\|O\| \leq 1$). By using the obtained training data $\{(x_i, o_i)\}_i$, the restricted classical ML model will produce a prediction model $h_{\mathrm{C}}(x)$ that allows accurate prediction of $f_{\mathcal{E}}(x) = \mathrm{tr}(O\mathcal{E}(|x\rangle\langle x|))$. The restricted classical ML model should provide accurate prediction model for any CPTP map $\mathcal{E} \in \mathcal{F}$.

Classical machine learning model for a given learning problem

We consider the following classical machine learning model. First, we sample N classical inputs $x_1, \ldots, x_N$ according to the distribution $\mathcal{D}$. Then, we obtain an associated quantum measurement outcome o_i for each input x_i.That is, o_i is a random variable that reproduces the target function in expectation only, see Equation (7.77). We denote the underlying distribution by $\mathcal{D}_o(O, \mathcal{E}(|x_i\rangle\langle x_i|))$ to delineate dependence on input x_i and CPTP map $\mathcal{E}$. After obtaining the training data $\{(x_i, o_i)\}_{i=1}^N$, the model performs the following optimization to minimize the *empirical* training error

$$f_* = \operatorname*{arg\,min}_{f \in M^p_{4\epsilon}(\mathcal{F}_f)} \frac{1}{N} \sum_{i=1}^{N} |f(x_i) - o_i|^2. \tag{7.78}$$

Here, $M^p_{4\epsilon}(\mathcal{F}_f)$ is the maximal packing net defined in Section 7.3. The packing net is a subset of the set $\mathcal{F}_f$ that contains functions that are sufficiently different from one another. By "empirical training error" we mean the deviation of the function $f(x_i)$ from the actual measurement outcome o_i, averaged over N data points: $\frac{1}{N}\sum_{i=1}^{N}|f(x_i)-o_i|^2$. In the later discussion, we will also refer to the *ideal* training error, meaning the average deviation of the function $f(x_i)$ from the expectation value $f_{\mathcal{E}}(x_i)=\mathrm{tr}(O\mathcal{E}(|x_i\rangle\langle x_i|))$: $\frac{1}{N}\sum_{i=1}^{N}|f(x_i)-f_{\mathcal{E}}(x_i)|^2$. This distinction is important; the ideal training error could be close to zero as long as the maximal packing net $M^p_{4\epsilon}(\mathcal{F}_f)$ is closely packed, but because of the statistical fluctuation in the quantum measurements, the outcomes $\{o_i\}$ can deviate substantially from the expectation value $f_{\mathcal{E}}(x_i)$. Therefore we might not be able to achieve small empirical training error even if $f=f_{\mathcal{E}}$.

In the following, we will provide a tight statistical analysis for bounding the prediction error

$$\mathop{\mathbb{E}}_{x\sim\mathcal{D}}|f_*(x)-f_{\mathcal{E}}(x)|^2. \tag{7.79}$$

The statistical analysis relies crucially on the distance measure used to define the packing net $M^p_{4\epsilon}(\mathcal{F}_f)$. Recall that this is the average squared distance over the input distribution $\mathcal{D}$, and the statistical fluctuation in performing quantum measurements to obtain o_i, see Equation (7.23). In particular, we will show that a data size of $N=\Theta(\log(|M^p_{4\epsilon}(\mathcal{F}_f)|)/\epsilon)$ suffices to achieve prediction errors of order $O(\epsilon)$ only.

We find it worthwhile to point out that this scaling is better than one might expect. Standard results in statistical learning theory (Bartlett and Mendelson, 2002; Mohri, Rostamizadeh, and Talwalkar, 2018) usually yield a data size of order $\log(|M^p_{4\epsilon}(\mathcal{F}_f)|)/\epsilon^2$, which is worse than our result by an additional $1/\epsilon$ factor.

Concentration results I: Ideal training error

We begin by considering the concentration of the *ideal* training error for an arbitrary function f from the maximal packing net $M^p_{4\epsilon}(\mathcal{F}_f)$:

$$\frac{1}{N}\sum_{i=1}^{N}|f(x_i)-f_{\mathcal{E}}(x_i)|^2, \tag{7.80}$$

which only depends on the inputs $x_1,\ldots,x_N$ and is independent of the observable measurement outcome o_i. We use the quantifier *ideal* because we compare directly with the expectation value $f_{\mathcal{E}}(x_i)$ rather than the measurement outcome o_i.

As a first step, view $|f(x)-f_{\mathcal{E}}(x)|^2$ with $x\overset{\mathcal{D}}{\sim}\{0,1\}^n$ as a random variable and check that it is bounded: $|f(x)-f_{\mathcal{E}}(x)|^2\leq(|f(x)|+|f_{\mathcal{E}}(x)|)^2\leq 4$ for all $x\in\{0,1\}^n$.

This implies the following bound on the variance:

$$\mathrm{Var}[|f(x) - f_{\mathcal{E}}(x)|^2] \leq \mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^4 \leq 4 \mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2. \tag{7.81}$$

We see that the ideal training error is a sum of independent random variables with bounded variance. Bernstein's inequality implies for $t > 0$

$$\Pr\left[\left|\frac{1}{N}\sum_{i=1}^{N} |f(x_i) - f_{\mathcal{E}}(x_i)|^2 - \mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2\right| \geq t\right] \tag{7.82}$$

$$\leq 2\exp\left(-\frac{1}{2}\frac{Nt^2}{4\,\mathbb{E}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2 + \frac{8}{3}t}\right). \tag{7.83}$$

Assigning $t = \frac{1}{4}\mathbb{E}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2$ allows us to conclude

$$\Pr\left[\left|\frac{1}{N}\sum_{i=1}^{N} |f(x_i) - f_{\mathcal{E}}(x_i)|^2 - \mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2\right| \geq \frac{1}{4}\mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2\right] \tag{7.84}$$

$$\leq 2\exp\left(-\frac{3}{448}N\mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2\right) \tag{7.85}$$

On the other hand, if $\mathbb{E}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2 \leq 4\epsilon$, we instead assign $t = \epsilon$ to obtain

$$\Pr\left[\left|\frac{1}{N}\sum_{i=1}^{N} |f(x_i) - f_{\mathcal{E}}(x_i)|^2 - \mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2\right| \geq \epsilon\right] \leq 2\exp\left(-\frac{3}{112}N\epsilon\right). \tag{7.86}$$

These two tail bounds (that cover different regimes) and a union bound then imply

$$\Pr\Bigg[\forall f \in M^p_{4\epsilon}(\mathcal{F}_f), \left|\frac{1}{N}\sum_{i=1}^{N} |f(x_i) - f_{\mathcal{E}}(x_i)|^2 - \mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2\right| \tag{7.87}$$

$$\geq \frac{1}{4}\max\left(4\epsilon, \mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2\right)\Bigg]$$

$$\leq 2\sum_{f\in M^p_{4\epsilon}(\mathcal{F}_f)} \exp\left(-\frac{3}{448}N\max\left(4\epsilon, \mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2\right)\right) \tag{7.88}$$

$$\leq 2\left|M^p_{4\epsilon}(\mathcal{F}_f)\right|\exp\left(-\frac{3}{112}N\epsilon\right). \tag{7.89}$$

Intuitively, this can be understood as follows. The functions f close to $f_{\mathcal{E}}$ will be distorted by at most ϵ, while the functions f that are further away from $f_{\mathcal{E}}$ will be distorted by a value proportional to the distance $\mathbb{E}_{x\sim\mathcal{D}} \left|f(x) - f_{\mathcal{E}}(x)\right|^2$. For $\delta \in (0,1)$ (confidence), we set

$$N \geq \frac{38 \log(2 \left|M^p_{4\epsilon}(\mathcal{F}_f)\right| / \delta)}{\epsilon}. \tag{7.90}$$

(Throughout this paper, log has base e unless otherwise indicated.) Then, with probability at least $1 - \delta$, we have

$$\forall f \in M^p_{4\epsilon}(\mathcal{F}_f), \left|\frac{1}{N}\sum_{i=1}^{N} |f(x_i) - f_{\mathcal{E}}(x_i)|^2 - \mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2\right| \tag{7.91}$$

$$< \frac{1}{4}\max\left(4\epsilon, \mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2\right). \tag{7.92}$$

We note that the training data size N in Equation (7.90) scales as $1/\epsilon$ rather than $1/\epsilon^2$, an improvement over the standard scaling typically encountered in statistical learning theory (Bartlett and Mendelson, 2002; Mohri, Rostamizadeh, and Talwalkar, 2018). The $1/\epsilon^2$ comes naturally when we sample over the different inputs x_i and apply a concentration inequality on the ideal training error $\frac{1}{N}\sum_{i=1}^{N} |f(x_i) - f_{\mathcal{E}}(x_i)|^2$ to guarantee an $O(\epsilon)$ statistical fluctuation around the prediction error $\mathbb{E}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2$. The main reason for the improved scaling is that any function f with a small prediction error $\mathbb{E}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2 \leq 4\epsilon$ also has a small variance (e.g., a highly biased coin that almost always come out heads has a variance close to zero, which is much smaller than for an unbiased coin), so we need only $N = O(1/\epsilon)$ to achieve $O(\epsilon)$ statistical fluctuations. Furthermore, by examining Equation (7.91), we see that if function f has a large prediction error then the statistical fluctuations in the training data may also be large. This is a price we pay to avoid a training set with size N scaling as $1/\epsilon^2$. The increased statistical fluctuations for a function f with a large prediction error are not problematic, because the statistical fluctuation are still smaller than the prediction error in that case. We find that functions with small prediction error have small ideal training error, while functions with large prediction error have large ideal training error, which is adequate for our purposes.

We condition on the event that the display Equation (7.91) holds true and proceed to the second step.

Concentration results II: Shifted empirical training error

In the second step, we will condition on a set of inputs $x_1, \ldots, x_N$ and study the concentration of statistical fluctuations in the observable measurement outcome o_i.

Let us define a new quantity, which we call the *shifted* empirical training error:

$$\frac{1}{N}\sum_{i=1}^{N}\left[|f(x_i)-o_i|^2-|f_{\mathcal{E}}(x_i)-o_i|^2\right], \tag{7.93}$$

where f can be any function in the packing net $M^p_{4\epsilon}(\mathcal{F}_f)$. The expectation value of the shifted empirical training error can be computed by means of direct expansion. Use $f_{\mathcal{E}}(x_i)=\mathbb{E}_{o\sim\mathcal{D}_o(O,\mathcal{E}(|x_i\rangle\langle x_i|))}[o]$ to rewrite

$$|f(x_i)-f_{\mathcal{E}}(x_i)|^2=\mathop{\mathbb{E}}_{o\sim\mathcal{D}_o(O,\mathcal{E}(|x_i\rangle\langle x_i|))}|f(x_i)-o|^2-|f_{\mathcal{E}}(x_i)-o|^2, \tag{7.94}$$

and for fixed input x_i, we can also bound the variance:

$$\mathrm{Var}_{o\sim\mathcal{D}_o(O,\mathcal{E}(|x_i\rangle\langle x_i|))}|f(x_i)-o|^2-|f_{\mathcal{E}}(x_i)-o|^2 \tag{7.95}$$

$$=\mathop{\mathbb{E}}_{o\sim\mathcal{D}_o(O,\mathcal{E}(|x_i\rangle\langle x_i|))}4(f(x_i)-f_{\mathcal{E}}(x_i))^2(o-f_{\mathcal{E}}(x_i))^2 \tag{7.96}$$

$$=4(f(x_i)-f_{\mathcal{E}}(x_i))^2\,\mathrm{Var}_{o\sim\mathcal{D}_o(O,\mathcal{E}(|x_i\rangle\langle x_i|))}[o] \tag{7.97}$$

$$\leq 4|f(x_i)-f_{\mathcal{E}}(x_i)|^2. \tag{7.98}$$

The last inequality is contingent on $\|O\|\leq 1$ which implies $o\in[-1,1]$ with probability one. Now, we apply Bernstein's inequality again. The o_i's are independent, bounded random variables with small variance. So, we obtain

$$\Pr\left[\left|\frac{1}{N}\sum_{i=1}^{N}\left[|f(x_i)-o_i|^2-|f_{\mathcal{E}}(x_i)-o_i|^2\right]-\frac{1}{N}\sum_{i=1}^{N}|f(x_i)-f_{\mathcal{E}}(x_i)|^2\right|\geq t\right] \tag{7.99}$$

$$\leq 2\exp\left(-\frac{1}{2}\frac{Nt^2}{\frac{4}{N}\sum_{i=1}^{N}|f(x_i)-f_{\mathcal{E}}(x_i)|^2+\frac{8}{3}t}\right)\quad\text{for } t>0. \tag{7.100}$$

Assigning $t=\frac{1}{4N}\sum_{i=1}^{N}|f(x_i)-f_{\mathcal{E}}(x_i)|^2$ ensures

$$\Pr\left[\left|\frac{1}{N}\sum_{i=1}^{N}\left[|f(x_i)-o_i|^2-|f_{\mathcal{E}}(x_i)-o_i|^2\right]-\frac{1}{N}\sum_{i=1}^{N}|f(x_i)-f_{\mathcal{E}}(x_i)|^2]\right| \tag{7.101}$$

$$\geq\frac{1}{4N}\sum_{i=1}^{N}|f(x_i)-f_{\mathcal{E}}(x_i)|^2\Bigg] \tag{7.102}$$

$$\leq 2\exp\left(-\frac{3}{448}\sum_{i=1}^{N}|f(x_i)-f_{\mathcal{E}}(x_i)|^2\right). \tag{7.103}$$

If $\frac{1}{N}\sum_{i=1}^{N}|f(x_i)-f_{\mathcal{E}}(x_i)|^2\leq 4\epsilon$, we instead assign $t=\epsilon$ to obtain

$$\Pr\left[\left|\frac{1}{N}\sum_{i=1}^{N}\left[|f(x_i)-o_i|^2-|f_{\mathcal{E}}(x_i)-o_i|^2\right]-\frac{1}{N}\sum_{i=1}^{N}|f(x_i)-f_{\mathcal{E}}(x_i)|^2]\right|\geq\epsilon\right] \tag{7.104}$$

$$\leq 2\exp\left(-\frac{3}{112}N\epsilon\right). \tag{7.105}$$

These results are conditioned on $x_1, \ldots, x_N$ already being sampled (the result of first step) where the event given in Equation (7.91) holds.

Prediction error for functions in the maximal packing net

Before bounding the prediction error of $f_\star$ obtained by the restricted classical ML model, we need to show that the following events happen simultaneously with high probability.

- *Event 1:* There exists a function $\tilde{f} \in M^p_{4\epsilon}(\mathcal{F}_f)$ with small prediction error that results in an empirical training error that is upper bounded by a certain threshold. In particular, we will set out to show that
$$\frac{1}{N}\sum_{i=1}^{N} |\tilde{f}(x_i) - o_i|^2 \leq \frac{1}{N}\sum_{i=1}^{N} |f_{\mathcal{E}}(x_i) - o_i|^2 + \frac{25}{4}\epsilon. \tag{7.106}$$
- *Event 2:* All functions $f \in M^p_{4\epsilon}(\mathcal{F}_f)$ that have a large prediction error will result in an empirical training error lower bounded by a certain threshold. In particular, we define the event to be: for all models $f \in M^p_{4\epsilon}(\mathcal{F}_f)$ such that $\mathbb{E}_{x\sim D} |f(x) - f_{\mathcal{E}}(x)|^2 \geq 12\epsilon$ will have:
$$\frac{1}{N}\sum_{i=1}^{N} |f(x_i) - o_i|^2 \geq \frac{1}{N}\sum_{i=1}^{N} |f_{\mathcal{E}}(x_i) - o_i|^2 + \frac{27}{4}\epsilon. \tag{7.107}$$

Then, we can combine these statements to obtain a bound on the prediction error of $f_\star$. Let us first relate the packing net to another useful concept.

Lemma 48 (Maximal packing nets are covering nets)**.** *For all $f_{\mathcal{E}} \in \mathcal{F}_f$, there exists $\tilde{f} \in M^p_{4\epsilon}(\mathcal{F}_f)$, such that*

$$\mathop{\mathbb{E}}_{x\sim D} |f(x) - f_{\mathcal{E}}(x)|^2 \leq 4\epsilon. \tag{7.108}$$

The proof is standard, see e.g. (Vershynin, 2018a), and based on contradicting the assumption that $M^p_{4\epsilon}(\mathcal{F}_f)$ is a maximal packing net. Since it is short and insightful, we include the full proof for completeness.

Proof of Lemma 48. If there exists $f_{\mathcal{E}} \in \mathcal{F}$, such that for all $\tilde{f} \in M^p_{4\epsilon}(\mathcal{F}_f)$, we have

$$\mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f(x) - f_{\mathcal{E}}(x)|^2 > 4\epsilon, \tag{7.109}$$

then we can add $f_{\mathcal{E}}$ into the packing net $M^p_{4\epsilon}(\mathcal{F}_f)$. Hence $M^p_{4\epsilon}(\mathcal{F}_f)$ is not the maximal packing net. □

For Event 1 in Equation (7.106), we want to show the existence of a function $\tilde{f}$ that has a small prediction error as well as an empirical training error upper bounded by a threshold. Because $M^p_{4\epsilon}(\mathcal{F}_f)$ is a maximal packing net, using Lemma 48, there exists a function $\tilde{f}$ such that the prediction error

$$\mathop{\mathbb{E}}_{x\sim D} |\tilde{f}(x) - f_{\mathcal{E}}(x)|^2 \leq 4\epsilon. \tag{7.110}$$

We now condition on Equation (7.91) being true, which happens with probability at least $1-\delta$. Therefore, we have the following bound on the ideal training error

$$\frac{1}{N}\sum_{i=1}^{N} |\tilde{f}(x_i) - f_{\mathcal{E}}(x_i)|^2 \leq 5\epsilon. \tag{7.111}$$

We can now use this insight to control the shifted empirical training error. Use a combination of Eqs. (7.103) and (7.105) to conclude

$$\Pr\left[\left|\frac{1}{N}\sum_{i=1}^{N}\left[|\tilde{f}(x_i) - o_i|^2 - |f_{\mathcal{E}}(x_i) - o_i|^2\right] - \frac{1}{N}\sum_{i=1}^{N}|\tilde{f}(x_i) - f_{\mathcal{E}}(x_i)|^2]\right| \geq \frac{5}{4}\epsilon\right] \tag{7.112}$$

$$\leq 2\exp\left(-\frac{3}{112}N\epsilon\right) \leq \frac{\delta}{|M^p_{4\epsilon}(\mathcal{F}_f)|}. \tag{7.113}$$

The first inequality comes from separately analyzing the two cases: $\frac{1}{N}\sum_{i=1}^{N}|\tilde{f}(x_i) - f_{\mathcal{E}}(x_i)|^2 \leq 4\epsilon$ or $4\epsilon < \frac{1}{N}\sum_{i=1}^{N}|\tilde{f}(x_i) - f_{\mathcal{E}}(x_i)|^2 \leq 5\epsilon$, then take the looser statement. The second inequality arises from inserting the (lower bound) on training data size N from Equation (7.90). Therefore, if the display from Equation (7.91) is true (which happens with probability at least $1-\delta$), then

$$\frac{1}{N}\sum_{i=1}^{N}|\tilde{f}(x_i) - o_i|^2 \tag{7.114}$$

$$\leq \frac{1}{N}\sum_{i=1}^{N}|f_{\mathcal{E}}(x_i) - o_i|^2 + \frac{1}{N}\sum_{i=1}^{N}|\tilde{f}(x_i) - f_{\mathcal{E}}(x_i)|^2 + \frac{5}{4}\epsilon \tag{7.115}$$

$$\leq \frac{1}{N}\sum_{i=1}^{N} |f_{\mathcal{E}}(x_i) - o_i|^2 + \frac{25}{4}\epsilon \quad \text{with probability at least } 1 - \delta/|M^{p}_{4\epsilon}(\mathcal{F}_f)|. \tag{7.116}$$

The second inequality is contingent on using Equation (7.111). This is Event 1 that we have set out to establish. And it is guaranteed to happen with high probability.

We now move on to Event 2 given in Equation (7.107). For any $f \in M^{p}_{4\epsilon}(\mathcal{F}_f)$ with a large prediction error

$$\mathop{\mathbb{E}}_{x\sim D} |f(x) - f_{\mathcal{E}}(x)|^2 \geq 12\epsilon, \tag{7.117}$$

we want to show that the training error $\frac{1}{N}\sum_{i=1}^{N} |f(x_i) - o_i|^2$ will also be large. We again condition on the event displayed by Equation (7.91) (which happens with probability at least $1 - \delta$). This relation implies the following bound on the ideal training error

$$\frac{1}{N}\sum_{i=1}^{N} |f(x_i) - f_{\mathcal{E}}(x_i)|^2 \geq \frac{3}{4}\mathop{\mathbb{E}}_{x\sim D} |f(x) - f_{\mathcal{E}}(x)|^2 \geq 9\epsilon. \tag{7.118}$$

Using the concentration result from Equation (7.103), we have

$$\Pr\left[\left|\frac{1}{N}\sum_{i=1}^{N}\left[|f(x_i) - o_i|^2 - |f_{\mathcal{E}}(x_i) - o_i|^2\right] - \frac{1}{N}\sum_{i=1}^{N}|f(x_i) - f_{\mathcal{E}}(x_i)|^2\right]\right| \tag{7.119}$$

$$\left.\geq \frac{1}{4N}\sum_{i=1}^{N}|f(x_i) - f_{\mathcal{E}}(x_i)|^2\right] \tag{7.120}$$

$$\leq 2\exp\left(-\frac{3}{448}\sum_{i=1}^{N}|f(x_i) - f_{\mathcal{E}}(x_i)|^2\right) \leq 2\exp\left(-\frac{3}{448}N(9\epsilon)\right) \leq \frac{\delta}{|M^{p}_{4\epsilon}(\mathcal{F}_f)|}. \tag{7.121}$$

The last inequality uses the training data size bound from Equation (7.90). This ensures

$$\frac{1}{N}\sum_{i=1}^{N}|f(x_i) - o_i|^2 \tag{7.122}$$

$$\geq \frac{1}{N}\sum_{i=1}^{N}|f_{\mathcal{E}}(x_i) - o_i|^2 + \frac{3}{4}\frac{1}{N}\sum_{i=1}^{N}|f(x_i) - f_{\mathcal{E}}(x_i)|^2 \tag{7.123}$$

$$\geq \frac{1}{N}\sum_{i=1}^{N}|f_{\mathcal{E}}(x_i) - o_i|^2 + \frac{9}{16}\mathop{\mathbb{E}}_{x\sim D}|f(x) - f_{\mathcal{E}}(x)|^2 \tag{7.124}$$

$$\geq \frac{1}{N}\sum_{i=1}^{N} |f_{\mathcal{E}}(x_i) - o_i|^2 + \frac{27}{4}\epsilon \quad \text{with probability at least } 1 - \delta/|M^p_{4\epsilon}(\mathcal{F}_f)|. \tag{7.125}$$

We can combine these insights by applying union bound to obtain that all the desired events given in Equation (7.106) and (7.107) happen simultaneously with probability at least $1 - \delta$ if we condition on Equation (7.91) to be true. Furthermore, because the event in display (7.91) happens with probability $1 - \delta$, we can guarantee that the the desired events given in Equation (7.106) and (7.107) happen simultaneously with a probability at least $(1-\delta)^2 \geq 1-2\delta$. This statement uses the following basic fact from elementary probability theory: Let $p(A|B)$ be the probability of event A when we condition on event B. And let $p(B)$ be the probability of event B. Then the probability of event A, $p(A)$, is larger or equal to the probability that A, B both happens, $p(A, B) = p(A|B)p(B)$.

To conclude, we have shown that using a training data of size

$$N \geq 38\log(2\left|M^p_{4\epsilon}(\mathcal{F}_f)\right|/\delta)/\epsilon \tag{7.126}$$

guarantees that the relations given in Equation (7.106) and (7.107) happen with probability at least $1 - 2\delta$.

Prediction error for functions produced by restricted classical ML

Let us choose a data of size $N \geq 38\log(4\left|M^p_{4\epsilon}(\mathcal{F}_f)\right|/\delta)/\epsilon$ such that the two relations in Equation (7.106) and (7.107) are both true with probability $1-\delta$. We now combine these with two other concepts from the previous subsections. Let $f_{\mathcal{E}}(x)$ be the actual target function and recall that (at least) one packing net element $\tilde{f} \in M^p_{4\epsilon}(\mathcal{F}_f)$ is guaranteed to be close ($\mathbb{E}_{x\sim\mathcal{D}} |\tilde{f}(x) - f_{\mathcal{E}}(x)|^2 \leq 4\epsilon$ according to Lemma 48). The restricted classical ML model tries to identify such a packing net element by minimizing the empirical training error: $f_* = \arg\min_{f\in M^p_{4\epsilon}(\mathcal{F}_f)} \frac{1}{N}\sum_{i=1}^{N} |f(x_i) - o_i|^2$ according to Equation (7.78). This setup ensures

$$\frac{1}{N}\sum_{i=1}^{N} |f_*(x_i) - o_i|^2 = \min_{f\in M^p_{4\epsilon}(\mathcal{F}_f)} \frac{1}{N}\sum_{i=1}^{N} |f(x_i) - o_i|^2 \leq \frac{1}{N}\sum_{i=1}^{n} \left|\tilde{f}(x_i) - o_i\right|^2. \tag{7.127}$$

The first relation in Equation (7.106) allows us to take it from there. Indeed,

$$\frac{1}{N}\sum_{i=1}^{n} \left|\tilde{f}(x_i) - o_i\right|^2 \leq \frac{1}{N}\sum_{i=1}^{N} |f_{\mathcal{E}}(x_i) - o_i|^2 + \frac{25}{4}\epsilon < \frac{1}{N}\sum_{i=1}^{N} |f_{\mathcal{E}}(x_i) - o_i|^2 + \frac{27}{4}\epsilon,$$

where the strict inequality is completely trivial. Apply the second relation in Equation (7.107) to complete the chain of arguments:

$$\frac{1}{N}\sum_{i=1}^{N}|f_*(x_i)-o_i|^2 < \min_{f\in M_{4\epsilon}^{p}(\mathcal{F}_f):\ \mathbb{E}_{x\sim\mathcal{D}}|f(x)-f_{\mathcal{E}}(x)|^2\geq 12\epsilon}\frac{1}{N}\sum_{i=1}^{N}|f(x_i)-o_i|^2\,. \tag{7.128}$$

In words, this display implies that the empirical training error achieved by f_* – the output of the restricted classical ML model – is strictly smaller than any empirical training error that could be achieved by any packing net function that has a comparatively large prediction error (at least 12ϵ). Therefore if f_* has a prediction error of at least 12ϵ, then this leads to a contradiction that $\frac{1}{N}\sum_{i=1}^{N}|f_*(x_i)-o_i|^2 < \frac{1}{N}\sum_{i=1}^{N}|f_*(x_i)-o_i|^2$. By contradiction, this claim implies that the prediction error achieved by f_* cannot be too bad.

Proposition 14. *Let $f_*:\{0,1\}^n\to\mathbb{R}$ be the packing net element that minimizes the empirical training error in Equation* (7.78)*. Then, for $\delta\in(0,1)$, training data of size $N\geq 38\log(4\left|M_{4\epsilon}^{p}(\mathcal{F}_f)\right|/\delta)/\epsilon$ implies:*

$$\mathop{\mathbb{E}}_{x\sim\mathcal{D}}|f^*(x)-f_{\mathcal{E}}(x)|^2<12\epsilon \quad \textit{with probability at least } 1-\delta. \tag{7.129}$$

Combining the upper and lower bound

If a quantum ML model produces a prediction h_{Q} achieving average prediction error

$$\mathop{\mathbb{E}}_{x\sim\mathcal{D}}\left|h_{\mathrm{Q}}(x)-\mathrm{tr}(O\mathcal{E}(|x\rangle\langle x|))\right|^2\leq\epsilon, \tag{7.130}$$

with probability at least $2/3$ for any CPTP map $\mathcal{E}\in\mathcal{F}$, then, as proven in Equation (7.55), the quantum ML must access the map $\mathcal{E}$ at least N_{Q} times, where

$$N_{\mathrm{Q}}=\Omega\left(\frac{\log(|M_{4\epsilon}^{p}(\mathcal{F}_f)|)}{m}\right). \tag{7.131}$$

On the other hand, from Proposition 14, we know there is a restricted classical ML model producing prediction h_{C} achieving average prediction error

$$\mathop{\mathbb{E}}_{x\sim\mathcal{D}}|h_{\mathrm{C}}(x)-\mathrm{tr}(O\mathcal{E}(|x\rangle\langle x|))|^2\leq 12\epsilon=O(\epsilon), \tag{7.132}$$

with high probability for any CPTP map $\mathcal{E}\in\mathcal{F}$, such that the restricted classical ML accesses the map N_{C} times, where

$$N_{\mathrm{C}}=O\left(\frac{\log(|M_{4\epsilon}^{p}(\mathcal{F}_f)|)}{\epsilon}\right)=O\left(\frac{mN_{\mathrm{Q}}}{\epsilon}\right). \tag{7.133}$$

This concludes the proof of Theorem 39.

7.4 Examples saturating the maximum information-theoretic advantage

Proposition 15. *For any $\epsilon \in (0, 1/3)$, and positive integer m, there exists a learning problem (7.1) – specified by an m-qubit observable O, a set $\mathcal{F}$ of CPTP maps and a distribution $\mathcal{D}$ on n-bit inputs, where $n = m-1$, – with the following property: Any restricted classical ML model that can learn a function $h_{\mathrm{C}}(x)$ that achieves*

$$\mathop{\mathbb{E}}_{x\sim\mathcal{D}} |h_{\mathrm{C}}(x) - \mathrm{tr}(O\mathcal{E}(|x\rangle\langle x|))|^2 \leq \epsilon, \tag{7.134}$$

must use classical training data of size $N_{\mathrm{C}} = \Omega(mN_{\mathrm{Q}}/\epsilon)$, where N_{Q} is the number of queries in the best quantum ML model.

This statement follows from constructing a stylized learning problem that admits the largest possible separation (albeit only a small polynomial factor). We first introduce the problem and discuss quantum and classical strategies (and their limitations) afterwards. We will focus on restricted classical ML models, because Theorem 39 also considers restricted classical ML models. We leave open the question of whether the separation between unrestricted classical ML and quantum ML is tight or not.

Learning problem formulation Fix $\epsilon \in (0, 1/3)$, let m be the integer in the statement of Proposition 15 and set $n = m - 1$. We consider a set of CPTP maps $\mathcal{F} = \{\mathcal{E}_a : a \in \{0, 1\}^n\}$ containing 2^n elements, where each map in the set takes an n-qubit input to an $(n+1)$-qubit output. The map $\mathcal{E}_a$, labeled by bit string $a \in \{0, 1\}^n$, is comprised of 2×2^n Kraus operators:

$$\mathcal{E}_a(\rho) = \sum_{z\in\{0,1\}^n} \sum_{i=1}^{2} K_a^{z,i} \rho (K_a^{z,i})^\dagger \text{ with } \begin{cases} K_a^{z,1} = & \sqrt{\frac{1+\sqrt{3\epsilon}}{2}} (I \otimes I^{\otimes n})|a \odot z, a\rangle\langle z|, \\ K_a^{z,2} = & \sqrt{\frac{1-\sqrt{3\epsilon}}{2}} (X \otimes I^{\otimes n})|a \odot z, a\rangle\langle z|. \end{cases} \tag{7.135}$$

Here, X is a single-qubit bit flip and $a \odot z \in \{0, 1\}$ denotes the inner product of bit-strings in $\mathbb{Z}_2$. We also choose the $(n+1)$-qubit observable $O = Z \otimes I^{\otimes n}$, i.e. we measure the first qubit in the Z-basis and trace out the rest of the system. By construction, the resulting function admits a closed-form expression:

$$f_a(x) = \mathrm{tr}\left(O\mathcal{E}_a(|x\rangle\langle x|)\right) = \sqrt{3\epsilon}\left(1 - 2a \odot x\right). \tag{7.136}$$

We consider $\mathcal{D}$ to be the uniform distribution over the n-bit inputs.

Upper bound on the quantum query complexity The above learning problem is easy to solve in the quantum realm. Since the set of CPTP maps $\mathcal{F} = \{\mathcal{E}_a : a \in \{0,1\}^n\}$ is known, it suffices to extract the label $a \in \{0,1\}^n$ of the underlying CPTP map. Once a is known, the closed-form expression (7.136) allows to predict future function values $f_a(x)$ efficiently with perfect accuracy – regardless of the input $x \in \{0,1\}^n$.

Quantum computers are well equipped to extract the label a. In fact, a single query of the unknown CPTP map $\mathcal{E}_a$ suffices to extract the label by executing the following simple procedure:

1. prepare the all-zero state on n qubits: $\rho_0 = |0,\ldots,0\rangle\langle 0,\ldots,0|$;
2. query $\mathcal{E}$ and apply it to ρ_0: $\rho_1 = \frac{1}{2}(I + \sqrt{3\epsilon}Z) \otimes |a\rangle\langle a|$, according to Equation (7.135);
3. throw away (trace out) the first qubit to obtain the n remaining ones: $\rho_2 = |a\rangle\langle a|$;
4. perform a computational basis measurement to extact $a \in \{0,1\}^n$ with probability one.

We see that a single quantum query ($N_{\mathrm{Q}} = 1$) suffices to extract the label a with certainty. Subsequently, we can make efficient and perfect predictions via the closed-form expression (7.136):

$$\mathbb{E}_{x\sim\mathcal{D}} \left| h_{\mathrm{Q}}(x) - \operatorname{tr}\left(O\mathcal{E}_a(|x\rangle\langle x|)\right)\right|^2 = 0 \leq \epsilon \quad \Leftarrow \quad N_{\mathrm{Q}} = 1. \tag{7.137}$$

In words, $N_{\mathrm{Q}} = 1$ allows for training a quantum ML model $h_{\mathrm{Q}}(x)$ that achieves zero prediction error for all input distributions (perfect prediction). This concrete ML model is also optimal, because $N_{\mathrm{Q}} = 1$ is the smallest number of queries conceivable ($N_{\mathrm{Q}} = 0$ would not reveal any information about the underlying CPTP map).

Lower bound on the classical query complexity Let us now turn to potential classical strategies for solving the above learning problem. In contrast to the previous paragraph, we will not construct an explicit strategy. Instead, we will use ideas similar to Appendix 7.3 to establish a fundamental lower bound.

Recall that the input distribution $\mathcal{D}$ is taken to be the uniform distribution. Also, for each $\mathcal{E}_a \in \mathcal{F}$, the underlying function $f_a(x) = \operatorname{tr}\left(O\mathcal{E}_a(|x\rangle\langle x|)\right)$ admits a closed form expression, see Equation (7.136). For $a, b \in \{0,1\}^n$,

$$\mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f_a(x) - f_b(x)|^2 = \frac{1}{2^n} \sum_{x\in\{0,1\}^n} \left|\sqrt{3\epsilon} 2(a-b)\odot x\right|^2 = \begin{cases} 0 & \text{if } a = b, \\ 6\epsilon & \text{else,} \end{cases} \tag{7.138}$$

because $2^{-n}\sum_{x\in\{0,1\}^n} |c\odot x|^2 = 1/2$ for all n-bit strings $c \neq (0,\dots,0)$. Now, suppose that a restricted classical ML model can utilize training data $\mathcal{T} = \{(x_i, o_i)\}_{i=1}^{N_\mathrm{C}}$ to learn a function $h_\mathrm{C}(x)$ that obeys $\mathbb{E}_{x\sim\mathcal{D}} |h_\mathrm{C}(x) - \mathrm{tr}\,(O\mathcal{E}_a(|x\rangle\langle x|))|^2 \leq \epsilon$ with high probability for any label $a \in \{0,1\}^n$. Then, this model would also allow us to identify the underlying label. Indeed $\mathbb{E}_{x\sim\mathcal{D}} |h_\mathrm{C}(x) - f_b|^2 \leq \epsilon$ if $b = a$, while for $b \neq a$, by the triangle inequality,

$$\mathop{\mathbb{E}}_{x\sim\mathcal{D}} |h_\mathrm{C}(x) - f_b(x)|^2 \geq \left(\sqrt{\mathop{\mathbb{E}}_{x\sim\mathcal{D}} |f_a(x) - f_b(x)|^2} - \sqrt{\mathop{\mathbb{E}}_{x\sim\mathcal{D}} |h_\mathrm{C}(x) - f_a(x)|^2}\right)^2 \tag{7.139}$$

$$\geq \left(\sqrt{6\epsilon} - \sqrt{\epsilon}\right)^2 > \epsilon. \tag{7.140}$$

By checking $\mathbb{E}_{x\sim D} |h_\mathrm{C}(x) - f_b(x)|^2 \leq \epsilon$ for every possible value $b \in \{0,1\}^n$, the restricted classical ML model allows us to recover the underlying bit-string label $a \in \{0,1\}^n$ with high probability. For this part of the argument, what's essential is that the right-hand-side of the inequality Equation (7.139) is greater than ϵ. If we replace 3ϵ in Equation (7.135) by $\alpha\epsilon$, where α is a constant, we require $\sqrt{2\alpha} - 1 > 1$, or $\alpha > 2$. We chose $\alpha = 3$ merely for convenience.

For any random hidden bitstring $a \in \{0,1\}^n$, we can use the restricted classical ML to obtain the training data $\{(x_i, o_i)\}_{i=1}^{N_\mathrm{C}}$ and determine the underlying bitstring a. We assume that the restricted classical ML first query x_1 obtains o_1, then query x_2 obtains o_2, and so on. We also have

$$o_i = \begin{cases} +1 & \text{with probability } p_+ = \frac{1}{2}\left(1 + \sqrt{3\epsilon}\,(1 - 2a\odot x_i)\right), \\ -1 & \text{with probability } p_- = \frac{1}{2}\left(1 - \sqrt{3\epsilon}\,(1 - 2a\odot x_i)\right), \end{cases} \tag{7.141}$$

which is a single-shot outcome for measuring the observable O on the state

$$\mathcal{E}_a(|x_i\rangle\langle x_i|) \tag{7.142}$$

in the eigenbasis of $O = Z \otimes I^{\otimes n}$. Because we can use the training data $\{(x_i, o_i)\}_{i=1}^{N_\mathrm{C}}$ to determine a with high probability (by the assumption of the restricted classical ML model), Fano's inequality and the data processing inequality then imply a bound on the mutual information between the training data and the CPTP map label a:

$$I\big(a : \{(x_i, o_i)\}_{i=1}^{N_\mathrm{C}}\big) = \Omega(n). \tag{7.143}$$

Next, using chain rule of mutual information based on conditional mutual information, we have

$$I\big(a : \{(x_i, o_i)\}_{i=1}^{N_\mathrm{C}}\big) = \sum_{i=1}^{N_\mathrm{C}} I(a : (x_i, o_i) | \{(x_j, o_j)\}_{j=1}^{i-1}) \tag{7.144}$$

$$= \sum_{i=1}^{N_\mathrm{C}} I(a : o_i | \{(x_j, o_j)\}_{j=1}^{i-1}, x_i). \tag{7.145}$$

The second equality follows from the fact that x_i is chosen by the restricted classical ML using only the information of $\{(x_j, o_j)\}_{j=1}^{i-1}$, hence the input x_i does not provide any additional information about a, i.e., $I(a : x_i | \{(x_j, o_j)\}_{j=1}^{i-1}) = 0$. We now upper bound each term:

$$I(a : o_i | \{(x_j, o_j)\}_{j=1}^{i-1}, x_i) = H(o_i | \{(x_j, o_j)\}_{j=1}^{i-1}, x_i) - H(o_i | \{(x_j, o_j)\}_{j=1}^{i-1}, x_i, a) \tag{7.146}$$

Because o_i is a two-outcome random variable, $H(o_i | \{(x_j, o_j)\}_{j=1}^{i-1}, x_i) \leq H(o_i) \leq \log_2(2) = 1$. We now consider the distribution of o_i when we condition on $\{(x_j, o_j)\}_{j=1}^{i-1}, x_i, a$. A closer inspection of Equation (7.141) reveals that the probability of one outcome is $p = \frac{1}{2}\left(1 + \sqrt{3\epsilon}\right)$ and the other is $1 - p$ (the value $a \odot x_i \in \{0, 1\}$ only ever permutes the outcome sign). This ensures

$$H(o_i | \{(x_j, o_j)\}_{j=1}^{i-1}, x_i, a) = -p \log_2(p) - (1-p)\log_2(1-p) \geq \log_2(2) - (2p-1)^2, \tag{7.147}$$

and we conclude

$$I(a : o_i | \{(x_j, o_j)\}_{j=1}^{i-1}, x_i) \leq (2p - 1)^2 = 3\epsilon. \tag{7.148}$$

Finally, we combine Eqs. (7.143), (7.145) and (7.148) to conclude

$$\Omega(n) \leq I\big(a : \{(x_i, o_i)\}_{i=1}^{N_\mathrm{C}}\big) \leq \sum_{i=1}^{N_\mathrm{C}} I(a : o_i | \{(x_j, o_j)\}_{j=1}^{i-1}, x_i) \leq 3\epsilon N_\mathrm{C}.$$

Therefore, recalling that the output size of our set of maps is $m = n+1$, we have for a restricted classical ML model with small average prediction error:

$$\mathop{\mathbb{E}}_{x\sim\mathcal{D}} |h_\mathrm{C}(x) - \mathrm{tr}\left(O\mathcal{E}_a(|x\rangle\!\langle x|)\right)|^2 \leq \epsilon \text{ with high probability} \Rightarrow N_\mathrm{C} = \Omega\left(m/\epsilon\right). \tag{7.149}$$

Proposition 15 follows from combining this assertion with the fact that the underlying learning problem does admit a perfect quantum solution with $N_\mathrm{Q} = 1$, see Equation (7.137).

Chapter 8

POWER OF DATA AND QUANTUM ADVANTAGE

As quantum technologies continue to advance rapidly, it becomes increasingly important to understand which applications can benefit from the power of these devices. At the same time, machine learning on classical computers has made great strides, revolutionizing applications in image recognition, text translation, and even physics applications, with more computational power leading to ever-increasing performance Halevy, Norvig, and Pereira, 2009. As such, if quantum computers could accelerate machine learning, the potential for impact is enormous.

At least two paths towards quantum enhancement of machine learning have been considered. First, motivated by quantum applications in optimization Grover, 1996; Durr and Hoyer, 1996; Farhi, Goldstone, Gutmann, et al., 2001, the power of quantum computing could, in principle, be used to help improve the training process of existing classical models Neven et al., 2009; Rebentrost, Masoud Mohseni, and Lloyd, 2014, or enhance inference in graphical models Leifer and Poulin, 2008. This could include finding better optima in a training landscape or finding optima with fewer queries. However, without more structure known in the problem, the advantage along these lines may be limited to quadratic or small polynomial speedups Aaronson and Ambainis, 2009; Jarrod R McClean, Harrigan, et al., 2020.

The second vein of interest is the possibility of using quantum models to generate correlations between variables that are inefficient to represent through classical computation. The recent success both theoretically and experimentally for demonstrating quantum computations beyond classical tractability can be taken as evidence that quantum computers can sample from probability distributions that are exponentially difficult to sample from classically Boixo et al., 2018; Arute et al., 2019. If these distributions were to coincide with real-world distributions, this would suggest the potential for significant advantage. This is typically the type of advantage that has been sought in recent work on both quantum neural networks Peruzzo et al., 2014; Jarrod R McClean, Romero, et al., 2016; Farhi and Neven, 2018, which seek to parameterize a distribution through some set of adjustable parameters, and quantum kernel methods Havlicek et al., 2019 that use quantum computers to define a feature map that maps classical data into the quantum Hilbert space. The justification for

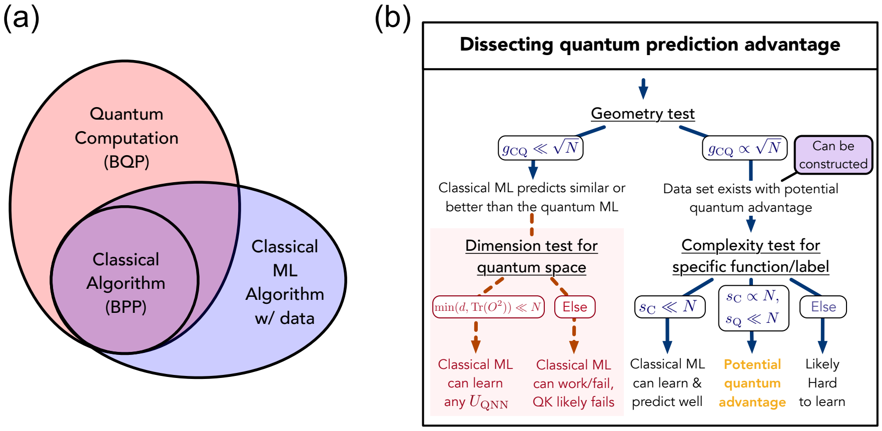

(a)
Quantum Computation (BQP)
Classical Algorithm (BPP)
Classical ML Algorithm w/ data
(b)
Dissecting quantum prediction advantage
Geometry test
$g_{\mathrm{CQ}} \ll \sqrt{N}$
$g_{\mathrm{CQ}} \propto \sqrt{N}$
Can be constructed
Classical ML predicts similar or better than the quantum ML
Data set exists with potential quantum advantage
Dimension test for quantum space
Complexity test for specific function/label
$\min(d, \mathrm{Tr}(O^2)) \ll N$
Else
$s_{\mathrm{C}} \ll N$
$s_{\mathrm{C}} \propto N,$ $s_{\mathrm{Q}} \ll N$
Else
Classical ML can learn any U_{QNN}
Classical ML can work/fail, QK likely fails
Classical ML can learn & predict well
Potential quantum advantage
Likely Hard to learn

Figure 8.1: Illustration of the relation between complexity classes and a flowchart for understanding and pre-screening potential quantum advantage. (a) We cartoon the separation between problem complexities that are created by the addition of data to a problem. Classical algorithms that can learn from data define a complexity class that can solve problems beyond classical computation (BPP), but it is still expected that quantum computation can efficiently solve problems that classical ML algorithm with data cannot. Rigorous definition and proof for the separation between classical algorithms that can learn from data and BPP / BQP is given in Section 3.1. (b) The flowchart we develop for understanding the potential for quantum prediction advantage. N samples of data from a potentially infinite depth QNN made with encoding and function circuits U_{enc} and U_{QNN} are provided as input along with quantum and classical methods with associated kernels. Tests are given as functions of N to emphasize the role of data in the possibility of a prediction advantage. One can first evaluate a geometric quantity g_{CQ} that measures the possibility of an advantageous quantum/classical prediction separation without yet considering the actual function to learn. We show how one can efficiently construct an adversarial function that saturates this limit if the test is passed, otherwise the classical approach is guaranteed to match performance for any function of the data. To subsequently consider the actual function provided, a label/function specific test may be run using the model complexities s_C and s_Q. If one specifically uses the quantum kernel (QK) method, the red dashed arrows can evaluate if all possible choices of U_{QNN} lead to an easy classical function for the chosen encoding of the data.

the capability of these methods to exceed classical models often follows similar lines as Refs Boixo et al., 2018; Arute et al., 2019 or quantum simulation results. That is, if the model leverages a quantum circuit that is hard to sample results from classically, then there is potential for a quantum advantage.

In this work, we show quantitatively how this picture is incomplete in machine learning (ML) problems where some training data is provided. The provided data can elevate classical models to rival quantum models, even when the quantum circuits generating the data are hard to compute classically. We begin with a motivating example and complexity-theoretic argument showing how classical algorithms with data can match quantum output. Following this, we provide rigorous prediction error bounds for training classical and quantum ML methods based on kernel functions Cortes and Vapnik, 1995; Schölkopf, Alexander J Smola, Bach, et al., 2002; Mohri, Rostamizadeh, and Talwalkar, 2018; Jacot, Gabriel, and Hongler, 2018; Novak, L. Xiao, Hron, J. Lee, Alexander A Alemi, et al., 2019; Arora et al., 2019; Havlicek et al., 2019; Blank et al., 2020; Bartkiewicz et al., 2020; Y. Liu, Arunachalam, and Temme, 2020 to learn quantum mechanical models. We focus on kernel methods, as they not only provide provable guarantees, but are also very flexible in the functions they can learn. For example, recent advancements in theoretical machine learning show that training neural networks with large hidden layers is equivalent to training an ML model with a particular kernel, known as the neural tangent kernel Jacot, Gabriel, and Hongler, 2018; Novak, L. Xiao, Hron, J. Lee, Alexander A Alemi, et al., 2019; Arora et al., 2019. Throughout, when we refer to classical ML models related to our theoretical developments, we will be referring to ML models that can be easily associated with a kernel, either explicitly as in kernel methods, or implicitly as in the neural tangent kernels. However, in the numerical section, we will also include performance comparisons to methods where direct association of a kernel is challenging, such as random forest methods. In the quantum case, we will also show how quantum ML based on kernels can be made equivalent to training an infinite depth quantum neural network.

We use our prediction error bounds to devise a flowchart for testing potential quantum prediction advantage, the separation between prediction errors of quantum and classical ML models for a fixed amount of training data. The most important test is a geometric difference between kernel functions defined by classical and quantum ML. Formally, the geometric difference is defined by the closest efficient classical ML model. In practice, one should consider the geometric difference with respect to a suite of optimized classical ML models. If the geometric difference is small, then a classical ML method is guaranteed to provide similar or better performance in prediction on the data set, independent of the function values or labels. Hence this represents a powerful, function independent pre-screening that allows one to evaluate if there is any possibility of better performance. On the other hand, if the geometry differs greatly, we show both the existence of a data set that exhibits large prediction advantage using the quantum ML model and how one can construct it efficiently. While the tools we develop could be used to compare and construct hard classical models like hash functions, we enforce restrictions that allow us to say something about a quantum separation. In particular, the feature map will be white box, in that a quantum circuit specification is available for the ideal feature map, and that feature map can be made computationally hard to evaluate classically. A constructive example of this is a discrete log feature map, where a provable separation for our kernel is given in Section 8.13. Additionally, the minimum over classical models means that classical hash functions are reproduced formally by definition.

Moreover, application of these tools to existing models in the literature rules many of them out immediately, providing a powerful sieve for focusing development of new data encodings. Following these constructions, in numerical experiments, we find that a variety of common quantum models in the literature perform similarly or worse than classical ML on both classical and quantum data sets due to a small geometric difference. The small geometric difference is a consequence of the exponentially large Hilbert space employed by existing quantum models, where all inputs are too far apart. To circumvent the setback, we propose an improvement, which enlarges the geometric difference by projecting quantum states embedded from classical data back to approximate classical representation Huang, Richard Kueng, and Preskill, 2020; J. Cotler and Wilczek, 2020b; Paini and Kalev, 2019. With the large geometric difference endowed by the projected quantum model, we are able to construct engineered data sets to demonstrate large prediction advantage over *common* classical ML models in numerical experiments up to 30 qubits. Despite our constructions being based on methods with associated kernels, we find empirically that the prediction advantage remains robust across tested classical methods, including those without an easily determined kernel. This opens the possibility to use a small quantum computer to generate efficiently verifiable machine learning problems that could be challenging for classical ML models.

8.1 Setup and motivating example

We begin by setting up the problems and methods of interest for classical and quantum models, and then provide a simple motivating example for studying how data can increase the power of classical models on quantum data. The focus will be a supervised learning task with a collection of N training examples $\{(x_i, y_i)\}$, where x_i is the input data and y_i is an associated label or value. We assume that x_i are sampled independently from a data distribution $\mathcal{D}$.

In our theoretical analysis, we will consider $y_i \in \mathbb{R}$ to be generated by some quantum model. In particular, we consider a continuous encoding unitary that maps a classical input data x_i into quantum state $|x_i\rangle = U_{\text{enc}}(x_i)\,|0\rangle^{\otimes n}$ and refer to the corresponding density matrix as $\rho(x_i)$. The expressive power of these embeddings have been investigated from a functional analysis point of view Lloyd, Schuld, et al., 2020; Schuld, Sweke, and Meyer, 2020, however the setting where data is provided requires special attention. The encoding unitary is followed by a unitary $U_{\text{QNN}}(\theta)$. We then measure an observable O after the quantum neural network. This produces the label/value for input x_i given as $y_i = f(x_i) = \langle x_i|\, U_{\text{QNN}}^{\dagger} O U_{\text{QNN}}\, |x_i\rangle$. The quantum model considered here is also referred to as a quantum neural network (QNN) in the literature Farhi and Neven, 2018; Jarrod R McClean, Boixo, et al., 2018. The goal is to understand when it is easy to predict the function $f(x)$ by training classical/quantum machine learning models.

With notation in place, we turn to a simple motivating example to understand how the availability of data in machine learning tasks can change computational hardness. Consider data points $\{\mathbf{x}_i\}_{i=1}^N$ that are p-dimensional classical vectors with $\|\mathbf{x}_i\|_2 = 1$, and use amplitude encoding Grant et al., 2019; Schuld, Bocharov, et al., 2020; LaRose and Coyle, 2020 to encode the data into an n-qubit state $|\mathbf{x}_i\rangle = \sum_{k=1}^{p} x_i^k\,|k\rangle$, where x_i^k is the individual coordinate of the vector $\mathbf{x}_i$. If U_{QNN} is a time-evolution under a many-body Hamiltonian, then the function $f(\mathbf{x}) = \langle \mathbf{x}|\, U_{\text{QNN}}^{\dagger} O U_{\text{QNN}}\, |\mathbf{x}\rangle$ is in general hard to compute classically Aram W Harrow and Montanaro, 2017b , even for a single input state. In particular, we have the following proposition showing that if a classical algorithm can compute $f(\mathbf{x})$ efficiently, then quantum computers will be no more powerful than classical computers; see Section 3.1 for proof.

Proposition 16. *If a classical algorithm without training data can compute $f(\mathbf{x})$ efficiently for any U_{QNN} and O, then BPP=BQP.*

Nevertheless, it is incorrect to conclude that training a classical model from data to learn this evolution is hard. To see this, we write out the expectation value as

$$\begin{aligned} f(x_i) &= \left(\sum_{k=1}^{p} x_i^{k*}\,\langle k|\right) U_{\text{QNN}}^{\dagger} O U_{\text{QNN}} \left(\sum_{l=1}^{p} x_i^{l}\,|l\rangle\right) \\ &= \sum_{k=1}^{p}\sum_{l=1}^{p} B_{kl}\, x_i^{k*} x_i^{l}, \end{aligned} \tag{8.1}$$

which is a quadratic function with p^2 coefficients $B_{kl} = \langle k|\, U_{\text{QNN}}^{\dagger} O U_{\text{QNN}}\, |l\rangle$. Using the theory developed later in this work, we can show that, for any U_{QNN} and O, training a specific classical ML model on a collection of N training examples $\{(\mathbf{x}_i, y_i = f(\mathbf{x}_i))\}$ would give rise to a prediction model $h(\mathbf{x}_i)$ with

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}} |h(\mathbf{x}) - f(\mathbf{x})| \leq c\sqrt{\frac{p^2}{N}}, \tag{8.2}$$

for a constant $c > 0$. We refer to Section 3.1 for the proof of this result. Hence, with $N \propto p^2/\epsilon^2$ training data, one can train a classical ML model to predict the function $f(\mathbf{x})$ up to an additive prediction error ϵ. This elevation of classical models through some training samples is illustrative of the power of data. In Section 3.1, we give a rigorous complexity-theoretic argument on the computational power provided by data. A cartoon depiction of the complexity separation induced by data is provided in Figure 8.1(a).

While this simple example makes the basic point that sufficient data can change complexity considerations, it perhaps opens more questions than it answers. For example, it uses a rather weak encoding into amplitudes and assumes one has access to an amount of data that is on par with the dimension of the model. The more interesting cases occur if we strengthen the data encoding, include modern classical ML models, and consider number of data N much less than the dimension of the model. These more interesting cases are the ones we quantitatively answer.

Our primary interest will be ML algorithms that are much stronger than fitting a quadratic function and the input data is provided in more interesting ways than an amplitude encoding. In this work, we focus on both classical and quantum ML models based on kernel functions $k(x_i, x_j)$. At a high level, a kernel function can be seen as a measure of similarity, if $k(x_i, x_j)$ is large when x_i and x_j are close. When considered for finite input data, a kernel function may be represented as a matrix $K_{ij} = k(x_i, x_j)$ and the conditions required for kernel methods are satisfied when the matrix representation is Hermitian and positive semi-definite.

A given kernel function corresponds to a nonlinear feature mapping $\phi(x)$ that maps x to a possibly infinite-dimensional feature space, such that $k(x_i, x_j) = \phi(x_i)^\dagger \phi(x_j)$. This is the basis of the so-called "kernel trick" where intricate and powerful maps $\phi(x_i)$ can be implemented through the evaluation of relatively simple kernel functions k. As a simple case, in the example above, using a kernel of $k(x_i, x_j) = |\langle x_i\rangle x_j|^2$ corresponds to a feature map $\phi(x_i) = \sum_{kl} x_i^{k*} x_i^l |k\rangle \otimes |l\rangle$ which is capable of learning quadratic functions in the amplitudes. In kernel based ML algorithms, the trained model can always be written as $h(x) = \mathbf{w}^\dagger \phi(x)$ where $\mathbf{w}$

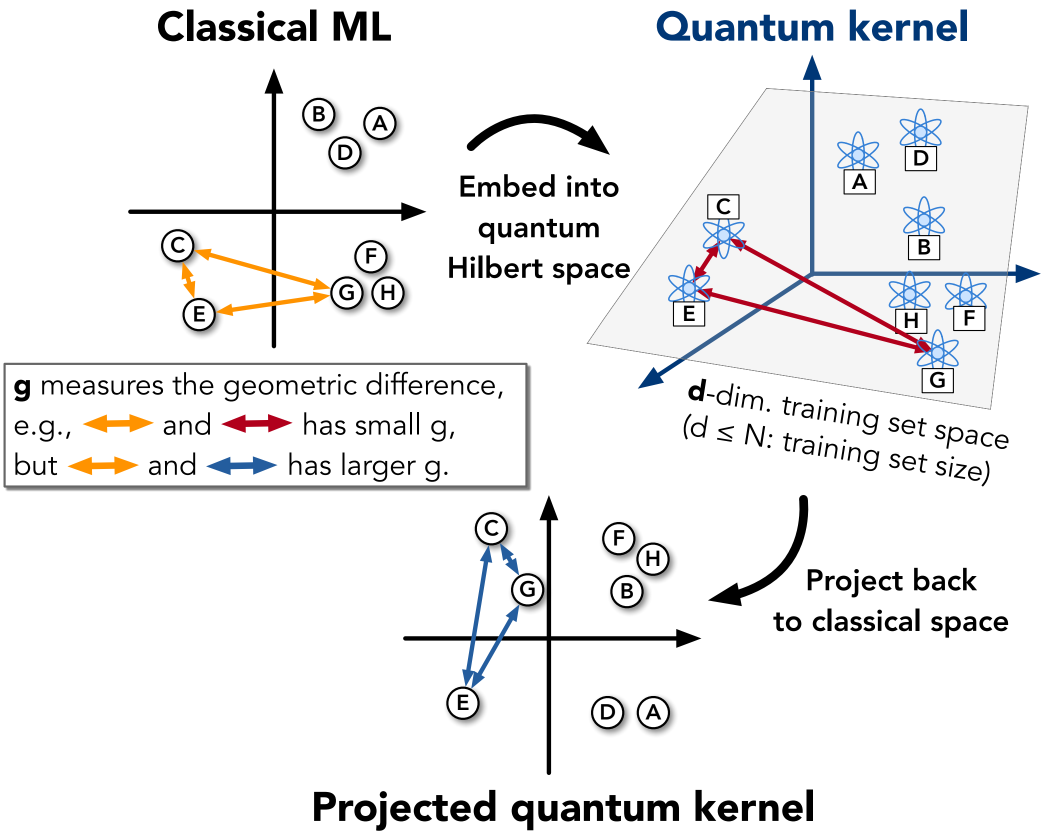

Classical ML
Quantum kernel
B
A
D
Embed into
quantum
Hilbert space
C
F
G
H
E
A
D
C
B
E
H
F
G
g measures the geometric difference,
e.g., ⟷ and ⟷ has small g,
but ⟷ and ⟷ has larger g.
d-dim. training set space
(d ≤ N: training set size)
C
F
H
G
B
Project back
to classical space
E
D
A
Projected quantum kernel

Figure 8.2: Cartoon of the geometry (kernel function) defined by classical and quantum ML models. The letters A, B, ... represent data points $\{x_i\}$ in different spaces with arrows representing the similarity measure (kernel function) between data. The geometric difference g is a difference between similarity measures (arrows) in different ML models and d is an effective dimension of the data set in the quantum Hilbert space.

is a vector in the feature space defined by the kernel. For example, training a convolutional neural network with large hidden layers Jacot, Gabriel, and Hongler, 2018; Z. Li et al., 2019 is equivalent to using a corresponding neural tangent kernel k^{CNN}. The feature map ϕ^{CNN} for the kernel k^{CNN} is a nonlinear mapping that extracts all local properties of x Z. Li et al., 2019. In quantum mechanics, similarly a kernel function can be defined using the native geometry of the quantum state space $|x\rangle$. For example, we can define the kernel function as $\langle x_i \rangle\, x_j$ or $|\langle x_i \rangle\, x_j|^2$. Using the output from this kernel in a method like a classical support vector machine Cortes and Vapnik, 1995 defines the quantum kernel method.

A wide class of functions can be learned with a sufficiently large amount of data by using the right kernel function k. For example, in contrast to the perhaps more natural kernel, $\langle x_i \rangle x_j$, the quantum kernel $k^{\mathrm{Q}}(x_i, x_j) = |\langle x_i \rangle x_j|^2 = \mathrm{tr}(\rho(x_i)\rho(x_j))$ can learn arbitrarily deep quantum neural network U_{QNN} that measures any observable O (shown in Section 8.5), and the Gaussian kernel, $k^{\gamma}(x_i, x_j) = \exp(-\gamma||x_i - x_j||^2)$ with hyper-parameter γ, can learn any continuous function in a compact space Micchelli, Xu, and Haizhang Zhang, 2006, which includes learning any QNN. Nevertheless, the required amount of data N to achieve a small prediction error could be very large in the worst case. Although we will work with other kernels defined through a quantum space, due both to this expressive property and terminology of past work, we will refer to $k^{\mathrm{Q}}(x_i, x_j) = \mathrm{tr}\left[\rho(x_i)\rho(x_j)\right]$ as the quantum kernel method throughout this work, which is also the definition given in Havlicek et al., 2019.

8.2 Testing quantum advantage

We now construct our more general framework for assessing the potential for quantum prediction advantage in a machine learning task. Beginning from a general result, we build both intuition and practical tests based on the geometry of the learning spaces. This framework is summarized in Figure 8.1.

Our foundation is a general prediction error bound for training classical/quantum ML models to predict some quantum model defined by $f(x) = \mathrm{tr}(O^U \rho(x))$ derived from concentration inequalities, where $O^U = U^\dagger_{\mathrm{QNN}} O U_{\mathrm{QNN}}$. Suppose we have obtained N training examples $\{(x_i, y_i = f(x_i))\}$. After training on this data, there exists an ML algorithm that outputs $h(x) = \mathbf{w}^\dagger \phi(x)$ using kernel $k(x_i, x_j) = K_{ij} = \phi(x_i)^\dagger \phi(x_j)$ which has a simplified prediction error bounded by

$$\mathbb{E}_{x \sim \mathcal{D}} |h(x) - f(x)| \leq c \sqrt{\frac{s_K(N)}{N}} \tag{8.3}$$

for a constant $c > 0$ and N independent samples from the data distribution $\mathcal{D}$. We note here that this and all subsequent bounds have a key dependence on the quantity of data N, reflecting the role of data to improve prediction performance. Due to a scaling freedom between $\alpha\phi(x)$ and $\mathbf{w}/\alpha$, we have assumed $\sum_{i=1}^{N} \phi(x_i)^\dagger \phi(x_i) = \mathrm{tr}(K) = N$. A derivation of this result is given in Section 8.6.

Given this core prediction error bound, we now seek to understand its implications. The main quantity that determines the prediction error is

$$s_K(N) = \sum_{i=1}^{N} \sum_{j=1}^{N} (K^{-1})_{ij} \, \mathrm{tr}(O^U \rho(x_i)) \, \mathrm{tr}(O^U \rho(x_j)). \tag{8.4}$$

The quantity $s_K(N)$ is equal to the model complexity of the trained function $h(x) = \mathbf{w}^\dagger \phi(x)$, where $s_K(N) = \|\mathbf{w}\|^2 = \mathbf{w}^\dagger \mathbf{w}$ after training. A smaller value of $s_K(N)$ implies better generalization to new data x sampled from the distribution $\mathcal{D}$. Intuitively, $s_K(N)$ measures whether the closeness between x_i, x_j defined by the kernel function $k(x_i, x_j)$ matches well with the closeness of the observable expectation for the quantum states $\rho(x_i), \rho(x_j)$, recalling that a larger kernel value indicates two points are closer. The computation of $s_K(N)$ can be performed efficiently on a classical computer by inverting an $N \times N$ matrix K after obtaining the N values $\mathrm{tr}(O^U \rho(x_i))$ by performing order N experiments on a physical quantum device. The time complexity scales at most as order N^3. Due to the connection between $\mathbf{w}^\dagger \mathbf{w}$ and the model complexity, a regularization term $\mathbf{w}^\dagger \mathbf{w}$ is often added to the optimization problem during the training of $h(x) = \mathbf{w}^\dagger \phi(x)$, see e.g., Krogh and Hertz, 1992; Cortes and Vapnik, 1995; Suykens and Vandewalle, 1999. Regularization prevents $s_K(N)$ from becoming too large at the expense of not completely fitting the training data. A detailed discussion and proof under regularization is given in Section 8.6 and 8.8.

The prediction error upper bound can often be shown to be asymptotically tight by proving a matching lower bound. As an example, when $k(x_i, x_j)$ is the quantum kernel $\mathrm{tr}(\rho(x_i)\rho(x_j))$, we can deduce that $s_K(N) \leq \mathrm{tr}(O^2)$ hence one would need a number of data N scaling as $\mathrm{tr}(O^2)$. In Section 8.10, we give a matching lower bound showing that a scaling of $\mathrm{tr}(O^2)$ is unavoidable if we assume a large Hilbert space dimension. This lower bound holds for any learning algorithm and not only for quantum kernel methods. The lower bound proof uses mutual information analysis and could easily extend to other kernels. This proof strategy is also employed extensively in a follow-up work Huang, Richard Kueng, and Preskill, 2021 to devise upper and lower bounds for classical and quantum ML in learning quantum models. Furthermore, not only are the bounds asymptotically tight, in numerical experiments given in Section 8.15 we find that the prediction error bound also captures the performance of other classical ML models not based on kernels where the constant factors are observed to be quite modest.

Given some set of data, if $s_K(N)$ is found to be small relative to N after training for a classical ML model, this quantum model $f(x)$ can be predicted accurately even if $f(x)$ is hard to compute classically for any given x. In order to formally evaluate the potential for quantum prediction advantage generally, one must take $s_K(N)$ to be the minimal over efficient classical models. However, we will be more focused on minimally attainable values over a reasonable set of classical methods with tuned hyperparameters. This prescribes an effective method for evaluating potential quantum advantage in practice, and already rules out a considerable number of examples from the literature.

From the bound, we can see that the potential advantage for one ML algorithm defined by K^1 to predict better than another ML algorithm defined by K^2 depends on the largest possible separation between s_{K^1} and s_{K^2} for a data set. The separation can be characterized by defining an asymmetric geometric difference that depends on the dataset, but is independent of the function values or labels. Hence evaluating this quantity is a good first step in understanding if there is a potential for quantum advantage, as shown in Figure 8.1. This quantity is defined by

$$g_{12} = g(K^1||K^2) = \sqrt{\left\| \sqrt{K^2}(K^1)^{-1}\sqrt{K^2} \right\|_\infty}, \tag{8.5}$$

where $||.||_\infty$ is the spectral norm of the resulting matrix and we assume $\mathrm{tr}(K^1) = \mathrm{tr}(K^2) = N$. One can show that $s_{K^1} \leq g_{12}^2 s_{K^2}$, which implies the prediction error bound $c\sqrt{s_{K^1}/N} \leq cg_{12}\sqrt{s_{K^2}/N}$. A detailed derivation is given in Section 8.8 and an illustration of g_{12} can be found in Figure 8.2. The geometric difference $g(K^1||K^2)$ can be computed on a classical computer by performing a singular value decomposition of the $N \times N$ matrices K^1 and K^2. Standard numerical analysis packages E. Anderson et al., 1999 provide highly efficient computation of a singular value decomposition in time at most order N^3. Intuitively, if $K^1(x_i, x_j)$ is small/large when $K^2(x_i, x_j)$ is small/large, then the geometric difference g_{12} is a small value ~ 1, where g_{12} grows as the kernels deviate.

To see more explicitly how the geometric difference allows one to make statements about the possibility for one ML model to make different predictions from another, consider the geometric difference $g_{\mathrm{CQ}} = g(K^{\mathrm{C}}||K^{\mathrm{Q}})$ between a classical ML model with kernel $k^{\mathrm{C}}(x_i, x_j)$ and a quantum ML model, e.g., with $k^{\mathrm{Q}}(x_i, x_j) = \mathrm{tr}(\rho(x_i)\rho(x_j))$. If g_{CQ} is small, because

$$s_{\mathrm{C}} \leq g_{\mathrm{CQ}}^2 s_{\mathrm{Q}}, \tag{8.6}$$

the classical ML model will always have a similar or better model complexity $s_K(N)$ compared to the quantum ML model. This implies that the prediction performance for the classical ML will likely be competitive or better than the quantum ML model, and one is likely to prefer using the classical model. This is captured in the first step of our flowchart in Figure 8.1.

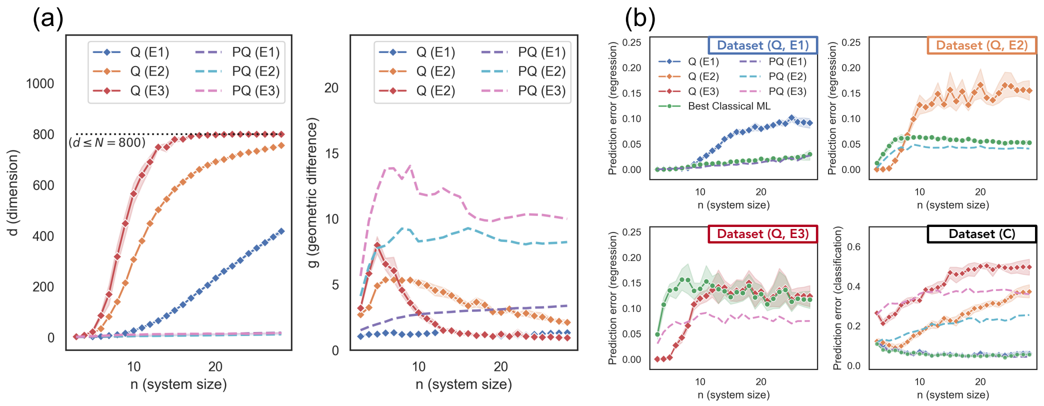


Figure 8.3: Relation between dimension d, geometric difference g, and prediction performance. The shaded regions are the standard deviation over 10 independent runs and n is the number of qubits in the quantum encoding and dimension of the input for the classical encoding. (a) The approximate dimension d and the geometric difference g with classical ML models for quantum kernel (Q) and projected quantum kernel (PQ) under different embeddings and system sizes n. (b) Prediction error (lower is better) of the quantum kernel method (Q), projected quantum kernel method (PQ), and classical ML models on classical (C) and quantum (Q) data sets with number of data $N = 600$. As d grows too large, the geometric difference g for quantum kernel becomes small. We see that small geometric difference g always results in classical ML being competitive or outperforming the quantum ML model. When g is large, there is a potential for improvement over classical ML. For example, projected quantum kernel improves upon the best classical ML in Dataset (Q, E3).

In contrast, if g_{CQ} is large we show that there exists a data set with $s_{\mathrm{C}} = g_{\mathrm{CQ}}^2 s_{\mathrm{Q}}$ with the quantum model exhibiting superior prediction performance. An efficient method to explicitly construct such a maximally divergent data set is given in Section 8.9 and a numerical demonstration of the stability of this separation is provided in the next section. While a formal statement about classical methods generally requires defining it over all efficient classical methods, in practice, we consider g_{CQ} to be the minimum geometric difference among a suite of optimized classical ML models. Our engineered approach minimizes this value as a hyperparameter search to find the best classical adversary, and shows remarkable robustness across classical methods including those without an associated kernel, such as random forests Breiman, 2001.

In the specific case of the quantum kernel method with

$$K_{ij}^{Q} = k^{\mathrm{Q}}(x_i, x_j) = \mathrm{tr}(\rho(x_i)\rho(x_j)), \tag{8.7}$$

we can gain additional insights into the model complexity s_K, and sometimes make conclusions about classically learnability for all possible U_{QNN} for the given encoding of the data. Let us define $\text{vec}(X)$ for a Hermitian matrix X to be a vector containing the real and imaginary part of each entry in X. In this case, we find $s_Q = \text{vec}(O^U)^T P_Q \text{vec}(O^U)$, where P_Q is the projector onto the subspace formed by $\{\text{vec}(\rho(x_1)), \ldots, \text{vec}(\rho(x_N))\}$. We highlight

$$d = \dim(P_Q) = \text{rank}(K^{\text{Q}}) \leq N, \tag{8.8}$$

which defines the effective dimension of the quantum state space spanned by the training data. An illustration of the dimension d can be found in Figure 8.1. Because P_Q is a projector and has eigenvalues 0 or 1, $s_{\text{Q}} \leq \min(d, \text{vec}(O^U)^T \text{vec}(O^U)) = \min(d, \text{tr}(O^2))$ assuming $\|O\|_\infty \leq 1$. Hence in the case of the quantum kernel method, the prediction error bound may be written as

$$\mathbb{E}_{x \in \mathcal{D}} |h(x) - f(x)| \leq c \sqrt{\frac{\min(d, \text{tr}(O^2))}{N}}. \tag{8.9}$$

A detailed derivation is given in Section 8.7. We can also consider the approximate dimension d, where small eigenvalues in K^{Q} are truncated, by incurring a small training error. After obtaining K^{Q} from a quantum device, the dimension d can be computed efficiently on a classical machine by performing a singular value decomposition on the $N \times N$ matrix K^{Q}. Estimation of $\text{tr}(O^2)$ can be performed by sampling random states $|\psi\rangle$ from a quantum 2-design, measuring O on $|\psi\rangle$, and performing statistical analysis on the measurement data Huang, Richard Kueng, and Preskill, 2020. This prediction error bound shows that a quantum kernel method can learn any U_{QNN} when the dimension of the training set space d or the squared Frobenius norm of observable $\text{tr}(O^2)$ is much smaller than the amount of data N. In Section 8.10, we show that quantum kernel methods are optimal for learning quantum models with bounded $\text{tr}(O^2)$ as they saturate the fundamental lower bound. However, in practice, most observables, such as Pauli operators, will have exponentially large $\text{tr}(O^2)$, so the central quantity is the dimension d. Using the prediction error bound for the quantum kernel method, if both g_{CQ} and $\min(d, \text{tr}(O^2))$ are small, then a classical ML would also be able to learn any U_{QNN}. In such a case, one must conclude that the given encoding of the data is classically easy, and this cannot be affected by an arbitrarily deep U_{QNN}. This constitutes the bottom left part of our flowchart in Figure 8.1.

Ultimately, to see a prediction advantage in a particular data set with specific function values/labels, we need a large separation between s_{C} and s_{Q}. This happens when the inputs x_i, x_j considered close in a quantum ML model are actually close in the target function $f(x)$, but are far in classical ML. This is represented as the final test in Figure 8.1 and the methodology here outlines how this result can be achieved in terms of its more essential components.

8.3 Projected quantum kernels

In addition to analyzing existing quantum models, the analysis approach introduced also provides suggestions for new quantum models with improved properties, which we now address here. For example, if we start with the original quantum kernel, when the effective dimension d is large, kernel $\mathrm{tr}(\rho(x_i)\rho(x_j))$, which is based on a fidelity-type metric, will regard all data to be far from each other and the kernel matrix K^{Q} will be close to identity. This results in a small geometric difference g_{CQ} leading to classical ML models being competitive or outperforming the quantum kernel method. In Section 8.11, we present a simple quantum model that requires an exponential amount of samples to learn using the quantum kernel $\mathrm{tr}(\rho(x_i)\rho(x_j))$, but only needs a linear number of samples to learn using a classical ML model.

To circumvent this setback, we propose a family of projected quantum kernels as a solution. These kernels work by projecting the quantum states to an approximate classical representation, e.g., using reduced physical observables or classical shadows Gosset and J. Smolin, 2019; Aaronson, 2020; Aaronson and Rothblum, 2019; Paini and Kalev, 2019; Huang, Richard Kueng, and Preskill, 2020. Even if the training set space has a large dimension $d \sim N$, the projection allows us to reduce to a low-dimensional classical space that can generalize better. Furthermore, by going through the exponentially large quantum Hilbert space, the projected quantum kernel can be challenging to evaluate without a quantum computer. In numerical experiments, we find that the classical projection increases rather than decreases the geometric difference with classical ML models. These constructions will be the foundation of our best performing quantum method later.

One of the simplest forms of projected quantum kernel is to measure the one-particle reduced density matrix (1-RDM) on all qubits for the encoded state, $\rho_k(x_i) = \mathrm{tr}_{j\neq k}[\rho(x_i)]$, then define the kernel as

$$k^{\mathrm{PQ}}(x_i, x_j) = \exp\left(-\gamma \sum_k \left\|\rho_k(x_i) - \rho_k(x_j)\right\|_F^2\right). \tag{8.10}$$

This kernel defines a feature map function in the 1-RDM space that is capable of expressing arbitrary functions of powers of the 1-RDMs of the quantum state. From

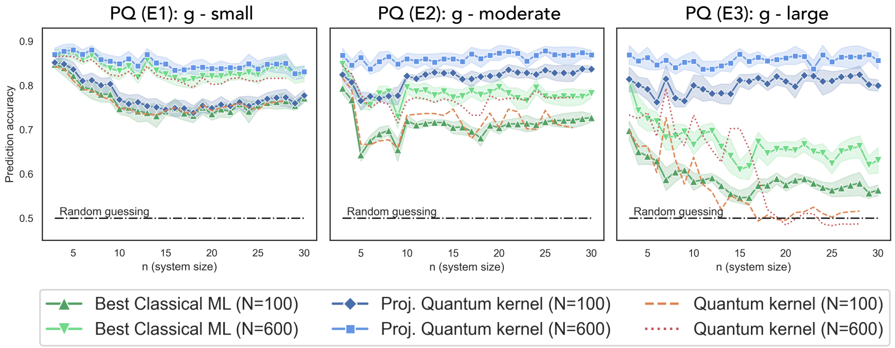


Figure 8.4: Prediction accuracy (higher the better) on engineered data sets. A label function is engineered to match the geometric difference $g(\mathrm{C}||\mathrm{PQ})$ between projected quantum kernel and classical approaches, demonstrating a significant gap between quantum and the best classical models up to 30 qubits when g is large. We consider the best performing classical ML models among Gaussian SVM, linear SVM, Adaboost, random forest, neural networks, and gradient boosting. We only report the accuracy of the quantum kernel method up to system size $n = 28$ due to the high simulation cost and the inferior performance.

non-intuitive results in density functional theory, we know even one body densities can be sufficient for determining exact ground state Pierre Hohenberg and Walter Kohn, 1964 and time-dependent Runge and E. K. Gross, 1984 properties of many-body systems under modest assumptions. In Section 8.12, we provide examples of other projected quantum kernels. This includes an efficient method for computing a kernel function that contains all orders of RDMs using local randomized measurements and the formalism of classical shadows Huang, Richard Kueng, and Preskill, 2020. The classical shadow formalism allows efficient construction of RDMs from very few measurements. In Section 8.13, we show that projected versions of quantum kernels lead to a simple and rigorous quantum speed-up in a recently proposed learning problem based on discrete logarithms Y. Liu, Arunachalam, and Temme, 2020.

8.4 Numerical experiments

We now provide numerical evidence up to 30 qubits that supports our theory on the relation between the dimension d, the geometric difference g, and the prediction performance. Using the projected quantum kernel, the geometric difference g is much larger and we see the strongest empirical advantage of a scalable quantum model on quantum data sets to date. These are the largest combined simulation and analysis in digital quantum machine learning that we are aware of, and make use of the Tensor-

Flow and TensorFlow-Quantum package Broughton, Verdon, McCourt, Antonio J Martinez, et al., 2020, reaching a peak throughput of up to 1.1 quadrillion floating point operations per second (petaflop/s). Trends of approximately 300 teraflop/s for quantum simulation and 800 teraflop/s for classical analysis were observed up to the maximum experiment size with the overall floating point operations across all experiments totalling approximately 2 quintillion (exaflop).

In order to mimic a data distribution that pertains to real-world data, we conduct our experiments around the fashion-MNIST data set H. Xiao, Rasul, and Vollgraf, 2017, which is an image classification for distinguishing clothing items, and is more challenging than the original digit-based MNIST source LeCun, Cortes, and Burges, 2010. We pre-process the data using principal component analysis Jolliffe, 1986 to transform each image into an n-dimensional vector. The same data is provided to the quantum and classical models, where in the classical case the data is the n-dimensional input vector, and the quantum case uses a given circuit to embed the n-dimensional vector into the space of n qubits. For quantum embeddings, we explore three options, E1 is a separable rotation circuit Schuld and Killoran, 2019; Schuld, Bocharov, et al., 2020; Skolik et al., 2020, E2 is an IQP-type embedding circuit Havlicek et al., 2019, and E3 is a Hamiltonian evolution circuit, with explicit constructions in Section 8.14.

For the classical ML task (C), the goal is to correctly identify the images as shirts or dresses from the original data set. For the quantum ML tasks, we use the same fashion-MINST source data and embeddings as above, but take as function values the expectation value of a local observable that has been evolved under a quantum neural network resembling the Trotter evolution of 1D-Heisenberg model with random couplings. In these cases, the embedding is taken as part of the ground truth, so the resulting function will be different depending on the quantum embedding. For these ML tasks, we compare against the best performing model from a list of standard classical ML algorithms with properly tuned hyper-parameters (see Section 8.14 for details).

In Figure 8.3, we give a comparison between the prediction performance of classical and quantum ML models. One can see that not only do classical ML models perform best on the original classical dataset, the prediction performance for the classical methods on the quantum datasets is also very competitive and can even outperform existing quantum ML models despite the quantum ML models having access to the training embedding while the classical methods do not. The performance of the classical ML model is especially strong on Dataset (Q, E1) and Dataset (Q, E2). This elevation of the classical performance is evidence of the power of data. Moreover, this intriguing behavior and the lack of quantum advantage may be explained by considering the effective dimension d and the geometric difference g following our theoretical constructions. From Figure 8.3a, we can see that the dimension d of the original quantum state space grows rather quickly, and the geometric difference g becomes small as the dimension becomes too large ($d \propto N$) for the standard quantum kernel. The saturation of the dimension coincides with the decreasing and statistical fluctuations in performance seen in Figure 8.4. Moreover, given poor ML performance a natural instinct is to throw more resources at the problem, e.g. more qubits, but as demonstrated here, doing this for naïve quantum kernel methods is likely to lead to tiny inner products and even worse performance. In contrast, the projected quantum space has a low dimension even when d grows, and yields a higher geometric difference g for all embeddings and system sizes. Our methodology predicts that, when g is small, classical ML model will be competitive or outperform the quantum ML model. This is verified in Figure 8.3b for both the original and projected quantum kernel, where a small geometric difference g leads to a very good performance of classical ML models and no large quantum advantage can be seen. Only when the geometric difference g is large (projected kernel method with embedding E3) can we see some mild advantage over the best classical method. This result holds disregarding any detail of the quantum evolution we are trying to learn, even for ones that are hard to simulate classically.

In order to push the limits of separation between quantum and classical approaches in a learning setting, we now consider a set of engineered data sets with function values designed to saturate the geometric inequality $s_\mathrm{C} \leq g(K^\mathrm{C} || K^\mathrm{PQ})^2 s_\mathrm{PQ}$ between classical ML models with associated kernels and the projected quantum kernel method. In particular, we design the data set such that $s_\mathrm{PQ} = 1$ and $s_\mathrm{C} = g(K^\mathrm{C} || K^\mathrm{PQ})^2$. Recall from Eq. (8.3), this data set will hence show the largest separation in the prediction error bound $\sqrt{s(N)/N}$. The engineered data set is constructed via a simple eigenvalue problem with the exact procedure described in Section 8.9 and the results are shown in Figure8.4. As the quantum nature of the encoding increases from E1 to E3, corresponding to increasing g, the performance of both the best classical methods and the original quantum kernel decline precipitously. The advantage of projected quantum kernel closely follows the geometric difference g and reaches more than 20% for large sizes. Despite the optimization of g only being possible for classical methods with an associated kernel, the performance advantage remains stable across other common classical methods. Note that we also constructed engineered data sets saturating the geometric inequality between classical ML and the original quantum kernel, but the small geometric difference g presented no empirical advantage at large system size (see Section 8.15).

In keeping with our arguments about the role of data, when we increase the number of training data N, all methods improve, and the advantage will gradually diminish. While this data set is engineered, it shows the strongest empirical separation on the largest system size to date. We conjecture that this procedure could be used with a quantum computer to create challenging data sets that are easy to learn with a quantum device, hard to learn classically while still being easy to verify classically given the correct labels. Moreover, the size of the margin implies that this separation may even persist under moderate amounts of noise in a quantum device.

8.5 Relation between quantum kernels and quantum neural networks

In this section, we demonstrate the formal equivalence of an arbitrary depth neural network with a quantum kernel method built from the original quadratic quantum kernel. This connection helps demonstrate the feature map induced by this kernel to motivate its use as opposed to the simpler inner product. While this equivalence shows the flexibility of this quantum kernel, it does not imply that it allows learning with a parsimonious amount of data. Indeed, in many cases, it requires both an exponential amount of data and exponential precision in the evaluation due to the fidelity type metric. In later sections, we show simple cases where it fails for illustration purposes.

Proposition 17. *Training an arbitrarily deep quantum neural network U_{QNN} with a trainable observable O is equivalent to training a quantum kernel method with kernel $k_Q(x_i, x_j) = \operatorname{tr}(\rho(x_i)\rho(x_j))$.*

Proof. Let us define $\rho_i = \rho(x_i) = U_{\text{enc}}(x_i) |0^n\rangle\langle 0^n| U_{\text{enc}}(x_i)^\dagger$ to be the corresponding quantum states for the classical input x_i. The training of a quantum neural network can be written‘ as

$$\min_{U \in C \subset U(2^n)} \sum_{i=1}^{N} l(\operatorname{tr}(OU\rho_i U^\dagger), y_i), \tag{8.11}$$

where $l(\tilde{y}, y)$ is a loss function that measures how close the prediction $\tilde{y}$ is to the true label y, C is the space of all possible unitaries considered by the parameterized quantum circuit, O is some predefined observable that we measure after evolving with U. Let us denote the optimal U to be U^*, then the prediction for a new input x is given by $\mathrm{tr}(OU^*\rho(x)(U^*)^\dagger)$.

On the other hand, the training of the quantum kernel method under the implied feature map is equivalent to training $W \in \mathbb{C}^{2^n\times 2^n}$ under the optimization

$$\min_{W\in\mathbb{C}^{2^n\times 2^n}} \sum_{i=1}^{N} l(\mathrm{tr}(W\rho_i), y_i) + \lambda\, \mathrm{tr}(W^\dagger W), \tag{8.12}$$

where $\lambda \geq 0$ is the regularization parameter and $l(\tilde{y}, y)$ is the loss function. Let us denote the optimal W to be W^*, then the prediction for a new input x is given by $\mathrm{tr}(W^*\rho(x))$. The well-known kernel trick allows efficient implementation of this machine learning model, and connects the original quantum kernel to the derivation here. Using the fact that ρ_i is Hermitian and set $\lambda = 0$, the quantum kernel method can be expressed as

$$\min_{\substack{U\in U(2^n),\\ O\in\mathbb{C}^{2^n\times 2^n}, O=O^\dagger}} \sum_{i=1}^{N} l(\mathrm{tr}(OU\rho_i U^\dagger), y_i). \tag{8.13}$$

This is equivalent to training an arbitrarily deep quantum neural network U with a trainable observable O. □

8.6 Proof of a general form of prediction error bound

This section is dedicated to deriving the precise statement for the core prediction error bound from which we base our methodology: $\mathbb{E}_x|h(x) - f(x)| \leq O(\sqrt{s/N})$ given by the first inequality in Equation (8.3). We will provide a detailed proof for the following general theorem when we include the regularization parameter λ. The regularization parameter λ will be used to improve prediction performance by limiting the complexity of the machine learning model.

Theorem 40. *Consider an observable O with $\|O\|_\infty \leq 1$, a quantum unitary U (e.g., a quantum neural network or a general Hamiltonian evolution), a mapping of classical input x to quantum system $\rho(x)$, and a training set of N data $\{(x_i, y_i = \mathrm{tr}(O^U\rho(x_i)))\}_{i=1}^N$, with $O^U = U^\dagger O U$ being the Heisenberg evolved observable. The training set is sampled from some unknown distribution over the input x. Suppose that $k(x, x')$ can be evaluated efficiently, and the kernel function is re-scaled to satisfy $\sum_{i=1}^N k(x_i, x_i) = N$. Define the Gram matrix $K_{ij} = k(x_i, x_j)$. For any $\lambda \geq 0$, with probability at least $1 - \delta$ over the sampling of the training data, we can learn a model $h(x)$ from the training data, such that the expected prediction error is* *bounded by*

$$\mathbb{E}_x|h(x) - \mathrm{tr}(O^U \rho(x))| \tag{8.14}$$

$$\leq O\left(\sqrt{\frac{\mathrm{tr}(A_{\mathrm{tra}} O^U \otimes O^U)}{N}} + \sqrt{\frac{\mathrm{tr}(A_{\mathrm{gen}} O^U \otimes O^U)}{N}} + \sqrt{\frac{\log(1/\delta)}{N}}\right), \tag{8.15}$$

where the two operators $A_{\mathrm{tra}}, A_{\mathrm{gen}}$ are given as

$$A_{\mathrm{tra}} = \lambda^2 \sum_{i=1}^{N} \sum_{j=1}^{N} ((K + \lambda I)^{-2})_{ij} \rho(x_i) \otimes \rho(x_j), \tag{8.16}$$

$$A_{\mathrm{gen}} = \sum_{i=1}^{N} \sum_{j=1}^{N} ((K + \lambda I)^{-1} K (K + \lambda I)^{-1})_{ij} \rho(x_i) \otimes \rho(x_j). \tag{8.17}$$

This is a data-dependent bound as A_{tra} and A_{gen} both depend on the N training data.

When we take the limit of $\lambda \longrightarrow 0$, we have $A_{\mathrm{tra}} = 0$ and

$$A_{\mathrm{gen}} = \sum_{i=1}^{N} \sum_{j=1}^{N} (K^{-1})_{ij} \rho(x_i) \otimes \rho(x_j). \tag{8.18}$$

Thus with probability at least $0.99 = 1 - \delta$, we have

$$\mathbb{E}_x|h(x) - \mathrm{tr}(O^U \rho(x))| \leq O\left(\sqrt{\frac{s_K(N)}{N}}\right), \tag{8.19}$$

where $s_K(N) = \sum_{i=1}^{N} \sum_{j=1}^{N} (K^{-1})_{ij} \mathrm{tr}(O^U \rho(x_i)) \mathrm{tr}(O^U \rho(x_j))$. This is the formula stated in the main text. However, in practice, we would recommend the use of regularization $\lambda > 0$ to prevent numerical instability and to obtain prediction error bound when we use a regularized ML model.

In Section 8.6, we will present the definition of the machine learning models used to prove Theorem 40. In Section 8.6 and 8.6, we will analyze the training error and generalization error of the machine learning models we consider to prove the prediction error bound given in Theorem 40.

Definition and training of machine learning models

We consider a class of machine learning models, including Gaussian kernel regression, infinite-width neural networks, and quantum kernel methods. These models are equivalent to training a linear function mapping from a (possibly infinite-dimensional) Hilbert space $\mathcal{H}$ to $\mathbb{R}$. The linear function can be written as $\langle \mathbf{w}, \phi(x) \rangle$, where $\mathbf{w}$ parameterizes the linear function, $\langle \cdot, \cdot \rangle : \mathcal{H} \times \mathcal{H} \to \mathbb{R}$ is an inner product, and $\phi(x)$ is a nonlinear mapping from the classical input x to the Hilbert space $\mathcal{H}$. For example, in quantum kernel method, we use the space of $2^n \times 2^n$ Hermitian matrices as the Hilbert space $\mathcal{H}$. This yields a natural definition of inner product $\langle \rho, \sigma \rangle = \mathrm{tr}(\rho\sigma) \in \mathbb{R}$.

Because the output $y = \mathrm{tr}(U^\dagger O U \rho(x))$ of the quantum model satisfies $y \in [-1, 1]$, we confine the output of the machine learning model to the interval $[-1, 1]$. The resulting machine learning model would be

$$h_w(x) = \min(1, \max(-1, \langle \mathbf{w}, \phi(x) \rangle)). \tag{8.20}$$

For efficient optimization of $\mathbf{w}$, we consider minimization of the following loss function

$$\min_{\mathbf{w}} \lambda \langle \mathbf{w}, \mathbf{w} \rangle + \sum_{i=1}^{N} \Big(\langle \mathbf{w}, \phi(x) \rangle - \mathrm{tr}(U^\dagger O U \rho(x_i)) \Big)^2, \tag{8.21}$$

where $\lambda \geq 0$ is a hyper-parameter. We define $\Phi = (\phi(x_1), \ldots, \phi(x_N))$. The kernel matrix $K = \Phi^\dagger \Phi$ is an $N \times N$ matrix that defines the geometry between all pairs of the training data. We see that $K_{ij} = \langle \phi(x_i), \phi(x_j) \rangle = k(x_i, x_j) \in \mathbb{R}$. Without loss of generality, we consider $\mathrm{tr}(K) = N$, which can be done by rescaling $k(x_i, x_j)$. The optimal $\mathbf{w}$ can be written down explicitly as

$$\mathbf{w} = \sum_{i=1}^{N} \sum_{j=1}^{N} \phi(x_i)((K + \lambda I)^{-1})_{ij} \, \mathrm{tr}(U^\dagger O U \rho(x_j)). \tag{8.22}$$

Hence the trained machine learning model would be

$$h_w(x) = \min\left(1, \max\left(-1, \sum_{i=1}^{N} \sum_{j=1}^{N} k(x_i, x)((K + \lambda I)^{-1})_{ij} \, \mathrm{tr}(U^\dagger O U \rho(x_j))\right)\right). \tag{8.23}$$

This is an analytic representation for various trained machine learning models, including least-square support vector machine Suykens and Vandewalle, 1999, kernel regression Nadaraya, 1964; N. S. Altman, 1992, and infinite-width neural networks Jacot, Gabriel, and Hongler, 2018. We will now analyze the prediction error of these machine learning models:

$$\epsilon_w(x) = |h_w(x) - \mathrm{tr}(U^\dagger O U \rho(x))|, \tag{8.24}$$

which is uniquely determined by the kernel matrix K and the hyper-parameter λ. In particular, we will focus on providing an upper bound on the expected prediction error

$$\mathbb{E}_x \ \epsilon_w(x) = \underbrace{\frac{1}{N}\sum_{i=1}^{N}\epsilon_w(x_i)}_{\text{Training error}} + \underbrace{\mathbb{E}_x \ \epsilon_w(x) - \frac{1}{N}\sum_{i=1}^{N}\epsilon_w(x_i)}_{\text{Generalization error}}, \tag{8.25}$$

which is the sum of training error and generalization error.

Training error

We will now relate the training error to the optimization problem, i.e., Equation (8.21), for obtaining the machine learning model $h_w(x)$. Because $\|O\| \leq 1$, we have $\mathrm{tr}(U^\dagger O U \rho(x)) \in [-1, 1]$, and hence $\epsilon_w(x) = |h_w(x) - \mathrm{tr}(U^\dagger O U \rho(x))| \leq |\langle \mathbf{w}, \phi(x)\rangle - \mathrm{tr}(U^\dagger O U \rho(x))|$. Using the convexity of x^2 and Jensen's inequality, we obtain

$$\frac{1}{N}\sum_{i=1}^{N}\epsilon_w(x_i) \leq \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\langle \mathbf{w}, \phi(x)\rangle - \mathrm{tr}(U^\dagger O U \rho(x_i))\right)^2}. \tag{8.26}$$

We can plug in the expression for the optimal $\mathbf{w}$ given in Equation (8.22) to yield

$$\frac{1}{N}\sum_{i=1}^{N}\epsilon_w(x_i) \leq \sqrt{\frac{\mathrm{tr}(A_{\mathrm{tra}}(U^\dagger O U) \otimes (U^\dagger O U))}{N}}, \tag{8.27}$$

where $A_{\mathrm{tra}} = \lambda^2 \sum_{i=1}^{N}\sum_{j=1}^{N}((K + \lambda I)^{-2})_{ij}\rho(x_i) \otimes \rho(x_j)$. When K is invertible and $\lambda = 0$, we can see that the training error is zero. However, in practice, we often set $\lambda > 0$.

Generalization error

A basic theorem in statistics and learning theory is presented below. This theorem provides an upper bound on the largest (one-sided) deviation from expectation over a family of functions. The following theorem has been introduced in Chapter 2.2. Here, we restate the theorem for convenience.

Theorem 41 (See Theorem 3.3 in Mohri, Rostamizadeh, and Talwalkar, 2018)**.** *Let $\mathcal{G}$ be a family of function mappings from a set $\mathcal{Z}$ to $[0, 1]$. Then for any $\delta > 0$, with probability at least $1 - \delta$ over identical and independent draw of N samples from $\mathcal{Z}$: $z_1, \ldots, z_N$, we have for all $g \in \mathcal{G}$,*

$$\mathbb{E}_z[g(z)] \leq \frac{1}{N}\sum_{i=1}^{N} g(z_i) + 2\mathbb{E}_\sigma\left[\sup_{g\in\mathcal{G}}\frac{1}{N}\sum_{i=1}^{N}\sigma_i g(z_i)\right] + 3\sqrt{\frac{\log(2/\delta)}{2N}}, \tag{8.28}$$

where $\sigma_1, \ldots \sigma_N$ are independent and uniform random variables over ± 1.

For our purpose, we will consider $\mathcal{Z}$ to be the space of classical input with $z_i = x_i$ drawn from some input distribution. Each function g would be equal to $\epsilon_w/2$ for some $\mathbf{w}$, where ϵ_w is defined in Equation (8.24). The reason that we divide by 2 is because the range of ϵ_w is $[0,2]$. And $\forall \gamma = 1,2,3,\ldots$, we define $\mathcal{G}_\gamma$ to be $\{\epsilon_w/2 \mid \forall \|\mathbf{w}\| \leq \gamma\}$. The definition of an infinite sequence of family of functions $\mathcal{G}_\gamma$ is useful for proving a prediction error bound for an unbounded class of machine learning models $h_w(x)$, where $\|\mathbf{w}\|$ could be arbitrarily large. Using Theorem 3 and multiplying the entire inequality by 2, we can show that the following inequality holds for any $\mathbf{w}$ with $\|\mathbf{w}\| \leq \gamma$,

$$\mathbb{E}_x[\epsilon_w(x)] - \frac{1}{N}\sum_{i=1}^{N}\epsilon_w(x_i) \leq 2\mathbb{E}_\sigma\left[\sup_{\|\mathbf{v}\|\leq\gamma}\frac{1}{N}\sum_{i=1}^{N}\sigma_i\epsilon_{\mathbf{v}}(x_i)\right] + 6\sqrt{\frac{\log(4\gamma^2/\delta)}{2N}}, \quad (8.29)$$

with probability at least $1 - \delta/2\gamma^2$. This probabilistic statement holds for any $\gamma = 1,2,3,\ldots$, but this does not yet guarantee that the inequality holds for all γ with high probability. We need to apply a union bound over all γ to achieve this, which shows that Inequality (8.29) holds for all γ with probability at least $1 - \sum_{\gamma=1}^{\infty}\delta/2\gamma^2 \geq 1 - \delta$.

Together we have shown that, for any $\mathbf{w} \in \mathcal{H}$, the generalization error $\mathbb{E}_x[\epsilon_w(x)] - \frac{1}{N}\sum_{i=1}^{N}\epsilon_w(x_i)$ is upper bounded by

$$2\mathbb{E}_\sigma\left[\sup_{\|\mathbf{v}\|\leq\lceil\|\mathbf{w}\|\rceil}\frac{1}{N}\sum_{i=1}^{N}\sigma_i\epsilon_{\mathbf{v}}(x_i)\right] + 6\sqrt{\frac{\log(4\lceil\|\mathbf{w}\|\rceil^2/\delta)}{2N}}, \quad (8.30)$$

with probability at least $1 - \delta$, where we consider the particular inequality with $\gamma = \lceil\|\mathbf{w}\|\rceil$. We will now analyze the above inequality using Talagrand's contraction lemma.

Lemma 49 (Talagrand's contraction lemma; See Lemma 5.7 in Mohri, Rostamizadeh, and Talwalkar, 2018)**.** *Let $\mathcal{G}$ be a family of function from a set $\mathcal{Z}$ to $\mathbb{R}$. Let $l_1,\ldots,l_N$ be Lipschitz-continuous function from $\mathbb{R} \to \mathbb{R}$ with Lipschitz constant L. Then*

$$\mathbb{E}_\sigma\left[\sup_{g\in\mathcal{G}}\frac{1}{N}\sum_{i=1}^{N}\sigma_i l_i(g(z_i))\right] \leq L\mathbb{E}_\sigma\left[\sup_{g\in\mathcal{G}}\frac{1}{N}\sum_{i=1}^{N}\sigma_i g(z_i)\right]. \quad (8.31)$$

We consider $l_i(s) = |\min(1,\max(-1,s)) - \mathrm{tr}(U^\dagger O U\rho(x_i))|$, $z_i = x_i$, and $\mathcal{G} = \{g_v(z_i) = \langle\mathbf{v}, z_i\rangle \mid \|\mathbf{v}\| \leq \lceil\|\mathbf{w}\|\rceil\}$. This choice of functions gives $\epsilon_v(x_i) = l_i(g(z_i))$.

Furthermore, l_i is Lipschitz-continuous with Lipschitz constant 1. Talagrand's contraction lemma then allows us to bound the formula in Equation (8.30) by

$$2\mathbb{E}_\sigma\left[\sup_{\|\mathbf{v}\|\leq\lceil\|\mathbf{w}\|\rceil}\frac{1}{N}\sum_{i=1}^N\sigma_i\langle\mathbf{v},\phi(x_i)\rangle\right]+6\sqrt{\frac{\log(4\lceil\|\mathbf{w}\|\rceil^2/\delta)}{2N}} \tag{8.32}$$

$$\leq 2\mathbb{E}_\sigma\left[\sup_{\|\mathbf{v}\|\leq\lceil\|\mathbf{w}\|\rceil}\frac{1}{N}\|\mathbf{v}\|\left\|\sum_{i=1}^N\sigma_i\phi(x_i)\right\|\right]+6\sqrt{\frac{\log(4\lceil\|\mathbf{w}\|\rceil^2/\delta)}{2N}} \tag{8.33}$$

$$\leq 2\lceil\|\mathbf{w}\|\rceil\mathbb{E}_\sigma\left[\frac{1}{N}\left\|\sum_{i=1}^N\sigma_i\phi(x_i)\right\|\right]+6\sqrt{\frac{\log(4\lceil\|\mathbf{w}\|\rceil^2/\delta)}{2N}} \tag{8.34}$$

$$\leq 2\frac{\lceil\|\mathbf{w}\|\rceil}{N}\sqrt{\mathbb{E}_\sigma\sum_{i=1}^N\sum_{j=1}^N\sigma_i\sigma_j k(x_i,x_j)}+6\sqrt{\frac{\log(4\lceil\|\mathbf{w}\|\rceil^2/\delta)}{2N}} \tag{8.35}$$

$$\leq 2\frac{\sqrt{\lceil\|\mathbf{w}\|\rceil^2\operatorname{tr}(K)}}{N}+6\sqrt{\frac{\log(4\lceil\|\mathbf{w}\|\rceil^2/\delta)}{2N}} \tag{8.36}$$

$$\leq 2\sqrt{\frac{\lceil\|\mathbf{w}\|\rceil^2}{N}}+6\sqrt{\frac{\log(\lceil\|\mathbf{w}\|\rceil)}{N}}+6\sqrt{\frac{\log(4/\delta)}{2N}} \tag{8.37}$$

$$\leq 8\sqrt{\frac{\lceil\|\mathbf{w}\|\rceil^2}{N}}+6\sqrt{\frac{\log(4/\delta)}{2N}}. \tag{8.38}$$

The first inequality uses Cauchy's inequality. The second inequality uses the fact that $\|\mathbf{v}\|\leq\lceil\|\mathbf{w}\|\rceil$. The third inequality uses a Jensen's inequality to move $\mathbb{E}_\sigma$ into the square-root. The fourth inequality uses the fact that σ_i are independent and uniform random variable taking $+1,-1$. The fifth inequality uses $\sqrt{x+y}\leq\sqrt{x}+\sqrt{y}, \forall x,y\geq 0$ and our assumption that we rescale K such that $\operatorname{tr}(K)=N$. The sixth inequality uses the fact that $x^2\geq\log(x), \forall x\in\mathbb{N}$.

Finally, we plug in the optimal $\mathbf{w}$ given in Equation (8.22). This allows us to obtain an upper bound of the generalization error:

$$\mathbb{E}_x[\epsilon_w(x)]-\frac{1}{N}\sum_{i=1}^N\epsilon_w(x_i)\leq 8\frac{\lceil\sqrt{\operatorname{tr}(A_{\text{gen}}(U^\dagger OU)\otimes(U^\dagger OU))}\rceil}{\sqrt{N}}+6\sqrt{\frac{\log(4/\delta)}{2N}}, \tag{8.39}$$

where $A_{\text{gen}}=\sum_{i=1}^N\sum_{j=1}^N((K+\lambda I)^{-1}K(K+\lambda I)^{-1})_{ij}\rho(x_i)\otimes\rho(x_j)$. When K is invertible and $\lambda=0$, we have $A_{\text{gen}}=\sum_{i=1}^N\sum_{j=1}^N(K^{-1})_{ij}\rho(x_i)\otimes\rho(x_j)$.

8.7 Prediction error bound based on dimension and geometric difference

In this section, we will show that for quantum kernel methods, we have

$$\mathbb{E}_x|h^{\text{Q}}(x)-\operatorname{tr}(O^U\rho(x))|\leq O\left(\sqrt{\frac{\min(d,\operatorname{tr}(O^2))}{N}}\right), \tag{8.40}$$

where d is the dimension of the training set space $d = \dim(\mathrm{span}(\rho(x_1), \ldots, \rho(x_N)))$. If we use the quantum kernel method as a reference point, then the prediction error of another machine learning algorithm that produces $h(x)$ using kernel matrix K can be bounded by

$$\mathbb{E}_x |h(x) - \mathrm{tr}(O^U \rho(x))| \leq O\left(g\sqrt{\frac{\min(d, \mathrm{tr}(O^2))}{N}}\right), \tag{8.41}$$

where $g = \sqrt{\left\|\sqrt{K^{\mathrm{Q}}} K^{-1} \sqrt{K^{\mathrm{Q}}}\right\|_\infty}$ assuming the normalization condition $\mathrm{tr}(K^{\mathrm{Q}}) = \mathrm{tr}(K) = N$.

Quantum kernel method

In quantum kernel method, the kernel function that will be used to train the model is defined using the quantum Hilbert space $k_Q(x, x') = \mathrm{tr}(\rho(x)\rho(x'))$. Correspondingly, we define the kernel matrix $K^Q_{ij} = k_Q(x_i, x_j)$. We will focus on $\rho(x)$ being a pure state, so the scaling condition $\mathrm{tr}(K^{\mathrm{Q}}) = \sum_{i=1}^N k_Q(x_i, x_i) = N$ is immediately satisfied. We also denote the trained model as h^{Q} for the quantum kernel method. We now consider an orthonormal basis $\{\sigma_1, \ldots, \sigma_d\}$ for the d-dimensional quantum state space formed by the training data $\mathrm{span}\{\rho(x_1), \ldots, \rho(x_N)\}$ under the inner product $\langle \rho, \sigma \rangle = \mathrm{tr}(\rho\sigma)$. We have σ_p is Hermitian, $\mathrm{tr}(\sigma_p^2) = 1$, but σ_p may not be positive semi-definite.

We consider an expansion of $\rho(x_i)$ in terms of σ_p:

$$\rho(x_i) = \sum_{p=1}^d \alpha_{ip} \sigma_p, \tag{8.42}$$

where $\alpha \in \mathbb{R}^{N \times d}$. The coefficient α is real as the vector space of Hermitian matrices is over real numbers. Note that multiplying a Hermitian matrix with an imaginary number will not generally result in a Hermitian matrix, hence Hermitian matrices are not a vector space over complex numbers. We can perform a singular value decomposition on $\alpha = U\Sigma V^\dagger$, where $U \in \mathbb{C}^{N \times d}, \Sigma, V \in \mathbb{C}^{d \times d}$ with $U^\dagger U = I$, Σ is diagonal and $\Sigma \succ 0$, $V^\dagger V = VV^\dagger = I$. Then $K^{\mathrm{Q}} = \alpha\alpha^\dagger = U\Sigma^2 U^\dagger$. This allows us to explicitly evaluate A_{tra} and A_{gen} given in Equation (8.16) and (8.17):

$$A_{\mathrm{tra}} = \lambda^2 \sum_{p=1}^d \sum_{q=1}^d \left(V \frac{\Sigma^2}{(\Sigma^2 + \lambda I)^2} V^\dagger\right)_{pq} \sigma_p \otimes \sigma_q, \tag{8.43}$$

$$A_{\mathrm{gen}} = \sum_{p=1}^d \sum_{q=1}^d \left(V \frac{\Sigma^4}{(\Sigma^2 + \lambda I)^2} V^\dagger\right)_{pq} \sigma_p \otimes \sigma_q, \tag{8.44}$$

which can be done by expanding $\rho(x_i)$ in terms of σ_p. Because $\Sigma \succ 0$, when we take the limit of $\lambda \to 0$, we have $A_{\text{tra}} = 0$ and $A_{\text{gen}} = \sum_{p=1}^d \sum_{q=1}^d \delta_{pq} \sigma_p \otimes \sigma_q = \sum_{p=1}^d \sigma_p \otimes \sigma_p$. Hence $\operatorname{tr}(A_{\text{tra}} O^U \otimes O^U) = 0$ and $\operatorname{tr}(A_{\text{gen}} O^U \otimes O^U) = \sum_{p=1}^d \operatorname{tr}(\sigma_p O^U)^2$. From Equation (8.14) with $\lambda \to 0$, we have

$$\mathbb{E}_x |h^Q(x) - \operatorname{tr}(O^U \rho(x))| \leq O\left(\sqrt{\frac{\sum_{p=1}^d \operatorname{tr}(O^U \sigma_p)^2}{N}} + \sqrt{\frac{\log(1/\delta)}{N}} \right). \tag{8.45}$$

Because $\{\sigma_1, \ldots, \sigma_k\}$ forms an orthonormal set in the space of $2^n \times 2^n$ Hermitian matrices, $\sum_{p=1}^d \operatorname{tr}(O^U \sigma_p)^2$ is the Frobenius norm of the observable O^U restricted to the subspace $\operatorname{span}\{\sigma_1, \ldots, \sigma_k\}$.

We now focus on obtaining an informative upper bound on how large

$$\sum_{p=1}^d \operatorname{tr}(O^U \sigma_p)^2 \tag{8.46}$$

could be. First, because we can extend the subspace $\operatorname{span}\{\sigma_1, \ldots, \sigma_k\}$ to the full Hilbert space $\operatorname{span}\{\sigma_1, \ldots \sigma_{4^n}\}$, we have

$$\sum_{p=1}^d \operatorname{tr}(O^U \sigma_p)^2 \leq \sum_{p=1}^{4^n} \operatorname{tr}(O^U \sigma_p)^2 = \operatorname{tr}((O^U)^2) = \left\| O^U \right\|_F^2. \tag{8.47}$$

Next, we will show that $\sum_{p=1}^d \operatorname{tr}(O^U \sigma_p)^2 \leq d \left\| O^U \right\|_\infty^2 \leq d$, where $\left\| O^U \right\|_\infty$ is the spectral norm of the observable O^U. We pick a linearly-independent set of $\{\rho_1, \ldots, \rho_k\}$ from $\{\rho(x_1), \ldots \rho(x_N)\}$. We assume that all the quantum states are pure, hence we have $\rho_i = |\psi_i\rangle\langle\psi_i|, \forall i = 1, \ldots, d$. The pure states $\{|\psi_1\rangle, \ldots, |\psi_k\rangle\}$ may not be orthogonal, so we perform a Gram-Schmidt process to create an orthonormal set of quantum states $\{|\phi_1\rangle, \ldots, |\phi_k\rangle\}$. Because ρ_i are linear combination of $|\phi_q\rangle \langle\phi_r|, \forall q, r = 1, \ldots, d$, we have

$$\sigma_p = \sum_{q=1}^d \sum_{r=1}^d s_{pqr} |\phi_q\rangle\langle\phi_r|, \forall p = 1, \ldots, d. \tag{8.48}$$

The condition $\operatorname{tr}(\sigma_p \sigma_{p'}) = \delta_{pp'}$ implies that $\sum_{q=1}^d \sum_{r=1}^d s_{pqr} s_{p'qr} = \delta_{pp'}$. If we view s as a vector $\mathbf{s}$ of size d^2, then $\langle \mathbf{s}_p, \mathbf{s}_{p'} \rangle = \delta_{pp'}$. Thus $\{\mathbf{s}_1, \ldots, \mathbf{s}_k\}$ forms a set of orthonormal vectors in $\mathbb{R}^{d^2}$, which implies $\sum_{p=1}^d \mathbf{s}_p \mathbf{s}_p^\dagger \preceq I$. Let us define the projection operator $P = \sum_{q=1}^d |\phi_q\rangle\langle\phi_q|$. We will also consider a vector $\mathbf{o} \in \mathbb{R}^{d^2}$, where $\mathbf{o}_{qr} = \langle\phi_r| O^U |\phi_q\rangle$. We have

$$\sum_{p=1}^d \operatorname{tr}(O^U \sigma_p)^2 = \sum_{p=1}^d \left(\sum_{q=1}^d \sum_{r=1}^d s_{pqr} \langle\phi_r| O^U |\phi_q\rangle \right)^2 = \sum_{p=1}^d \mathbf{o}^\dagger \mathbf{s}_p \mathbf{s}_p^\dagger \mathbf{o} \tag{8.49}$$

$$\leq \mathbf{o}^\dagger \mathbf{o} = \left(\sum_{q=1}^{d} \sum_{r=1}^{d} \langle \phi_r | \, O^U \, | \phi_q \rangle \right)^2 = \left\| P O^U P \right\|_F^2 . \tag{8.50}$$

The inequality comes from the fact that $\sum_{p=1}^{d} \mathbf{s}_p \mathbf{s}_p^\dagger \leq I$. With a proper choice of basis, one could view PO^UP as an $d \times d$ matrix. Hence $\left\|PO^UP\right\|_F \leq \sqrt{d} \left\|PO^UP\right\|_\infty \leq \sqrt{d} \left\|O^U\right\|_\infty$. This established the fact that $\sum_{p=1}^{d} \operatorname{tr}(O^U \sigma_p)^2 \leq d \left\|O^U\right\|_\infty^2 \leq d$. Combining with Equation (8.45), we have

$$\mathbb{E}_x |h^Q(x) - \operatorname{tr}(O^U \rho(x))| \leq O\left(\sqrt{\frac{\min(d, \left\|O^U\right\|_F^2)}{N}} + \sqrt{\frac{\log(1/\delta)}{N}} \right). \tag{8.51}$$

This elucidates the fact that the prediction error of a quantum kernel method is bounded by minimum of the dimension of the quantum subspace formed by the training set and the Frobenius norm of the observable O^U.

Choosing a small but non-zero λ allows us to consider an approximate space of $\operatorname{span}\{\rho(x_1), \ldots, \rho(x_N)\}$ formed by the training set. The training error

$$\sqrt{\operatorname{tr}(A_{\text{tra}} O^U \otimes O^U)/N} \tag{8.52}$$

would increase slightly, and the generalization error $\sqrt{\operatorname{tr}(A_{\text{gen}} O^U \otimes O^U)/N}$ would reflect the Frobenius norm of O^U restricted to a smaller subspace, which only contains the principal components of the space formed by the training set. This would be a better choice when most states lie in low-dimensional subspace with small random fluctuations. One may also consider training a machine learning model with truncated kernel matrix K_λ, where all singular values below λ are truncated. This makes the act of restricting to an approximate subspace more explicit.

Another machine learning method compared to quantum kernel

We now consider an upper bound on the prediction error using the quantum kernel method as a reference point for some machine learning algorithm. For the following discussion, we consider classical neural networks with large hidden sizes. The function generated by a classical neural network with large hidden size after training is equivalent to the function $h(x)$ given in Equation (8.22) with $\lambda = 0$ and with a special kernel function $k_{\text{NTK}}(x, x')$ known as the neural tangent kernel (NTK) Jacot, Gabriel, and Hongler, 2018. The precise definition of $k_{\text{NTK}}(x, x')$ depends on the architecture of the neural network. For example, a two-layer feedforward neural network (FNN), a three-layer FNN, or some particular form of convolutional neural network (CNN) all correspond to different $k_{\mathrm{NTK}}(x, x')$. Given the kernel $k_{\mathrm{NTK}}(x, x')$, we can define the kernel matrix $\tilde{K}_{ij} = k_{\mathrm{NTK}}(x_i, x_j)$. For neural tangent kernel, the scaling condition $\mathrm{tr}(\tilde{K}) = \sum_{i=1}^N k_{\mathrm{NTK}}(x_i, x_i) = N$ may not be satisfied. Hence, we define a normalized kernel matrix $K = N\tilde{K}/\mathrm{tr}(\tilde{K})$. When $\lambda = 0$, the trained machine learning model (given in Equation (8.22)) under the normalized matrix K and the original matrix $\tilde{K}$ are the same. In order to apply Theorem 40, we will use the normalized kernel matrix K for the following discussion. From Equation (8.14) with $\lambda = 0$, we have

$$\mathbb{E}_x |h(x) - \mathrm{tr}(O^U \rho(x))| \leq \mathcal{O}\left(\sqrt{\frac{\mathrm{tr}(AO^U \otimes O^U)}{N}} + \sqrt{\frac{\log(1/\delta)}{N}}\right), \tag{8.53}$$

where $A = \sum_{i=1}^N \sum_{j=1}^N (K^{-1})_{ij} \rho(x_i) \otimes \rho(x_j)$. Using Equation (8.42) on the expansion of $\rho(x_i)$, we have

$$A = \sum_{p=1}^d \sum_{q=1}^d \sum_{i=1}^N \sum_{j=1}^N (K^{-1})_{ij} \alpha_{ip} \alpha_{jq} \sigma_p \otimes \sigma_q \tag{8.54}$$

$$= \sum_{p=1}^d \sum_{q=1}^d (\alpha^\dagger K^{-1} \alpha)_{pq} \sigma_p \otimes \sigma_q. \tag{8.55}$$

Using the definition of spectral norm, we have

$$\mathrm{tr}(AO^U \otimes O^U) = \sum_{p=1}^d \sum_{q=1}^d (\alpha^\dagger K^{-1} \alpha)_{pq} \,\mathrm{tr}(\sigma_p O^U) \,\mathrm{tr}(\sigma_q O^U) \tag{8.56}$$

$$\leq \left\| \alpha^\dagger K^{-1} \alpha \right\|_\infty \sum_{p=1}^d \mathrm{tr}(O^U \sigma_p)^2. \tag{8.57}$$

Recall from the definition below Equation (8.42), we have

$$\alpha = U\Sigma V^\dagger, K^{\mathrm{Q}} = \alpha\alpha^\dagger = U\Sigma^2 U^\dagger. \tag{8.58}$$

Using the fact that orthogonal transformation do not change the spectral norm,

$$\left\| \alpha^\dagger K^{-1} \alpha \right\|_\infty = \left\| \Sigma U^\dagger K^{-1} U \Sigma \right\|_\infty = \left\| U\Sigma U^\dagger K^{-1} U \Sigma U^\dagger \right\|_\infty = \left\| \sqrt{K^{\mathrm{Q}}} K^{-1} \sqrt{K^{\mathrm{Q}}} \right\|_\infty. \tag{8.59}$$

Hence

$$\mathrm{tr}(AO^U \otimes O^U) \leq \left\| \sqrt{K^{\mathrm{Q}}} K^{-1} \sqrt{K^{\mathrm{Q}}} \right\|_\infty \sum_{p=1}^d \mathrm{tr}(O^U \sigma_p)^2. \tag{8.60}$$

Together with Equation (8.53), we have the following prediction error bound

$$\mathbb{E}_x|h(x) - \operatorname{tr}(O^U\rho(x))| \leq \mathcal{O}\left(g\sqrt{\frac{\sum_{p=1}^d \operatorname{tr}(O^U\sigma_p)^2}{N}} + \sqrt{\frac{\log(1/\delta)}{N}}\right), \tag{8.61}$$

where $g = \sqrt{\left\|\sqrt{K^{\mathrm{Q}}}K^{-1}\sqrt{K^{\mathrm{Q}}}\right\|_\infty}$. The scalar g measures the closeness of the geometry between the training data points defined by classical neural network and quantum state space. Note that without the geometric scalar g, this prediction error bound is the same as Equation (8.45) for the quantum kernel method. Hence, if g is small, classical neural network could predict as well (or potentially better) as the quantum kernel method. The same analysis in Section 8.7 allows us to arrive at the following result

$$\mathbb{E}_x|h(x) - \operatorname{tr}(O^U\rho(x))| \leq \mathcal{O}\left(g\sqrt{\frac{\min(d, \|O^U\|_F^2)}{N}} + \sqrt{\frac{\log(1/\delta)}{N}}\right). \tag{8.62}$$

The same analysis holds for other machine learning algorithms, such as Gaussian kernel regression.

8.8 Detailed discussion on the relevant quantities s, d, and g

There are some important aspects on the three relevant quantities s, d, g that were not fully discussed in the main text, including the limit when we have infinite amount of data and the effect of regularization. While in practice one always has a finite amount of data, constructing these formal limits both clarifies the construction and provides another perspective through which to understand the finite data constructions. This section will provide a detailed discussion of these aspects.

Model complexity s

While we have used $s_K(N) = \sum_{i=1}^N \sum_{j=1}^N (K^{-1})_{ij} \operatorname{tr}(O^U\rho(x_i)) \operatorname{tr}(O^U\rho(x_j))$ in the main text, this is a simplified quantity when we do not apply regularization. The model complexity $s_K(N)$ under regularization is given by

$$s_K(N) = \|\mathbf{w}\|^2 = \operatorname{tr}(A_{\mathrm{gen}}O^U \otimes O^U) \tag{8.63}$$

$$= \sum_{i=1}^N \sum_{j=1}^N ((K+\lambda I)^{-1}K(K+\lambda I)^{-1})_{ij} \operatorname{tr}(O^U\rho(x_i)) \operatorname{tr}(O^U\rho(x_j)) \tag{8.64}$$

$$= \sum_{i=1}^N \sum_{j=1}^N (\sqrt{K}(K+\lambda I)^{-2}\sqrt{K})_{ij} \operatorname{tr}(O^U\rho(x_i)) \operatorname{tr}(O^U\rho(x_j)). \tag{8.65}$$

Training machine learning model with regularization is often desired when we have a finite number N of training data. $\|\mathbf{w}\|^2$ has been used extensively in regularizing machine learning models, see e.g., Krogh and Hertz, 1992; Cortes and Vapnik, 1995; Suykens and Vandewalle, 1999. This is because we can often significantly reduce generalization error $\sqrt{\mathrm{tr}(A_{\mathrm{gen}} O^U \otimes O^U)/N}$ by slightly increasing the training error $\sqrt{\mathrm{tr}(A_{\mathrm{tra}} O^U \otimes O^U)/N}$. In practice, we should choose the regularization parameter λ to be a small number such that the training error plus the generalization error is minimized.

The model complexity $s_K(N)$ we have been calculating can be seen as an approximation to the true model complexity when we have a finite number N of training data. If we have exact knowledge about the input distribution given as a probability measure μ_x, we can also write down the precise model complexity in the reproducing kernel Hilbert space $\phi(x)$ where $k(x, y) = \phi(x)^\dagger \phi(y)$. Starting from

$$\min_{\mathbf{w}} \lambda \mathbf{w}^\dagger \mathbf{w} + \int \left\| \mathbf{w}^\dagger \phi(x) - \mathrm{tr}(O^U \rho(x)) \right\|^2 d\mu_x, \tag{8.66}$$

we can obtain

$$\mathbf{w} = \left(\lambda I + \int \phi(x)\phi(x)^\dagger d\mu_x \right)^{-1} \int \mathrm{tr}(O^U \rho(x)) \phi(x) d\mu_x. \tag{8.67}$$

Hence the true model complexity is

$$\|\mathbf{w}\|^2 \tag{8.68}$$

$$= \int \int d\mu_{x_1} d\mu_{x_2} \, \mathrm{tr}(O^U \rho(x_1)) \, \mathrm{tr}(O^U \rho(x_2)) \tag{8.69}$$

$$\phi(x_1)^\dagger \left(\lambda I + \int \phi(\xi)\phi(\xi)^\dagger d\mu_\xi \right)^{-2} \phi(x_2) \tag{8.70}$$

$$= \mathrm{tr}(A_{\mathrm{gen}} O^U \otimes O^U), \tag{8.71}$$

where the operator

$$A_{\mathrm{gen}} = \int \int d\mu_{x_1} d\mu_{x_2} \phi(x_1)^\dagger \left(\lambda I + \int \phi(\xi)\phi(\xi)^\dagger d\mu_\xi \right)^{-2} \phi(x_2) \; \rho(x_1) \otimes \rho(x_2). \tag{8.72}$$

If we replace the integration over the probability measure with N random samples and apply the fact that $k(x, y) = \phi(x)^\dagger \phi(y)$, then we can obtain the original expression given in Equation (8.17).

Dimension d

The dimension we considered in the main text is the effective dimension of the training set quantum state space. This can be seen as the rank of the quantum kernel matrix $K^Q_{ij} = \mathrm{tr}(\rho(x_i)\rho(x_j))$. However, it will often be the case that most of the states lie in some low-dimensional subspace, but have negligible contributions in a much higher dimensional subspace. In this case, the dimension of the low-dimensional subspace is the better characterization. More generally, we can perform a singular value decomposition of K^{Q}

$$K^{\mathrm{Q}} = \sum_{i=1}^{N} t_i u_i u_i^\dagger, \tag{8.73}$$

with $t_1 \geq t_2 \geq \ldots \geq t_N$. We define $\sigma_i = \sum_{j=1}^N u_{ij}\rho(x_j) / \left\| \sum_{j=1}^N u_{ij}\rho(x_j) \right\|_F$, where $\|\cdot\|_F$ is the Frobenius norm. σ_i is the i-th principal component of the quantum state space. Recall the normalization condition $\mathrm{tr}(K^{\mathrm{Q}}) = N$, so $\sum_{i=1}^N t_i = N$. If the training set quantum state space is one-dimensional ($d = 1$), then

$$t_1 = N, t_i = 0, \forall i > 1. \tag{8.74}$$

If all the quantum states in the training set are orthogonal ($d = N$), then

$$t_i = 1, \forall i = 1, \ldots, N. \tag{8.75}$$

By the Eckart-Young-Mirsky theorem, for any $k \geq 1$, the first k principal components $\sigma_1, \ldots, \sigma_k$ form the best k-dimensional subspace for approximating

$$\mathrm{span}\{\rho(x_1), \ldots, \rho(x_N)\}. \tag{8.76}$$

The approximation error is given by

$$\sum_{i=1}^{N} \left\| \rho(x_i) - \sum_{j=1}^{k} \sqrt{t_j} u_{ji} \sigma_j \right\|_F^2 = \sum_{l=k+1}^{N} t_l. \tag{8.77}$$

As we can see, when the spectrum is flatter, the dimension is larger. The error decreases at most as $\sum_{l=k}^N t_l \leq N - k$, where the equality holds when all states are orthogonal. In the numerical experiment, we choose the following measure as the approximate dimension

$$1 \leq \sum_{k=1}^{N} \left(\frac{1}{N-k} \sum_{l=k}^{N} t_l \right) \leq N \tag{8.78}$$

due to the independence to any hyperparameter. Alternatively, we can also define approximate dimension by choosing the smallest k such that $\sum_{l=k+1}^{N} t_l/N < \epsilon$ for some $\epsilon > 0$. Both give similar trend, but the actual value of the dimension would be different.

From the discussion, we can see that in the above definitions, the dimension will always be upper bounded by the number N of training data. Similar to the case of model complexity, we can also define the dimension d when we have the exact knowledge about the input distribution given by probability measure μ_x. For a quantum state space representing n qubits, we simply consider the spectrum $t_1 \geq t_2 \geq \ldots \geq t_{2^n}$ of the following operator

$$\int \mathrm{vec}(\rho(x))\mathrm{vec}(\rho(x))^T d\mu_x. \tag{8.79}$$

When we replace the integration by a finite number of training samples, the spectrum would be equivalent to the spectrum given in Equation (8.73) except for the additional zeros.

Remark 3. *The same definition of dimension can be used for any kernels, such as projected quantum kernels or neural tangent kernels (under the normalization* $\mathrm{tr}(K) = N$*).*

Geometric difference g

The geometric difference is defined between two kernel functions K^1, K^2 and the corresponding reproducing kernel Hilbert space $\phi_1(x), \phi_2(x)$. If we have a function represented by the first kernel $\mathbf{w}^\dagger \phi_1(x)$, what would be the model complexity for the second kernel? We consider the ideal case where we know the input distribution μ_x exactly. The optimization for training the first kernel method with regularization $\lambda > 0$ is

$$\min_{\mathbf{v}} \lambda \mathbf{v}^\dagger \mathbf{v} + \int \left\|\mathbf{v}^\dagger \phi_2(x) - \mathbf{w}^\dagger \phi_1(x)\right\|^2 d\mu_x. \tag{8.80}$$

The solution is given by

$$\mathbf{v} = \left(\lambda I + \int \phi_2(x)\phi_2(x)^\dagger d\mu_x\right)^{-1} \int \mathbf{w}^\dagger \phi_1(x)\phi_2(x) d\mu_x. \tag{8.81}$$

Hence the model complexity for the optimized $\mathbf{v}$ is

$$\|\mathbf{v}\|^2 \tag{8.82}$$

$$= \mathbf{w}^\dagger \int\int d\mu_{x_1} d\mu_{x_2} \phi_1(x_1)\phi_2(x_1)^\dagger \left(\lambda I + \int \phi_2(\xi)\phi_2(\xi)^\dagger d\mu_\xi\right)^{-2} \phi_2(x_2)\phi_1(x_2)^\dagger \mathbf{w} \tag{8.83}$$

$$\leq g_{\text{gen}}^2 \left\|\mathbf{w}\right\|^2, \tag{8.84}$$

where the geometric difference is

$$g_{\text{gen}} = \tag{8.85}$$

$$\sqrt{\left\| \int \int d\mu_{x_1} d\mu_{x_2} \phi_1(x_1)\phi_2(x_1)^\dagger \left(\lambda I + \int \phi_2(\xi)\phi_2(\xi)^\dagger d\mu_\xi\right)^{-2} \phi_2(x_2)\phi_1(x_2)^\dagger \right\|_\infty}. \tag{8.86}$$

The subscript in g_{gen} is added because when $\lambda > 0$, there will also be a contribution from training error. When we only have a finite number N of training samples, we can use the fact that $k(x, y) = \phi(x)^\dagger\phi(y)$ and the definition that $K_{ij} = k(x_i, x_j)$ to obtain

$$g_{\text{gen}} = \sqrt{\left\|\sqrt{K^1}\sqrt{K^2}\left(K^2 + \lambda I\right)^{-2}\sqrt{K^2}\sqrt{K^1}\right\|_\infty}. \tag{8.87}$$

This formula differs from the main text due to the regularization parameter λ. If $\lambda = 0$, then the above formula for g_{gen} reduces to the formula

$$g_{\text{gen}} = \sqrt{\left\|\sqrt{K^1}(K^2)^{-1}\sqrt{K^1}\right\|_\infty}. \tag{8.88}$$

When λ is non-zero, the geometric difference can become much smaller. This is the same as the discussion on model complexity s in Section 8.8. However, a nonzero λ induces a small amount of training error. For a finite number N of samples, the training error can always be upper bounded:

$$\frac{1}{N}\sum_{i=1}^{N}\left\|\mathbf{v}^\dagger\phi_2(x_i) - \mathbf{w}^\dagger\phi_1(x_i)\right\|^2 \leq \lambda^2\left\|\sqrt{K^1}(K^2 + \lambda I)^{-2}\sqrt{K^1}\right\|_\infty \left\|\mathbf{w}\right\|^2 = g_{\text{tra}}^2\left\|\mathbf{w}\right\|^2, \tag{8.89}$$

where $g_{\text{tra}} = \lambda\sqrt{\left\|\sqrt{K^1}(K^2 + \lambda I)^{-2}\sqrt{K^1}\right\|_\infty}$. This upper bound can be obtained by plugging the solution for $\mathbf{v}$ in Equation (8.80) under finite samples into the training error $\frac{1}{N}\sum_{i=1}^{N}\left\|\mathbf{v}^\dagger\phi_2(x_i) - \mathbf{w}^\dagger\phi_1(x_i)\right\|^2$ and utilizing the fact that $\mathbf{w}^\dagger A\mathbf{w} \leq \left\|A\right\|_\infty\left\|\mathbf{w}\right\|^2$. In the numerical experiment, we report g_{gen} given in Equation (8.87) with the largest λ such that the training error $g_{\text{tra}} \leq 0.045$.

8.9 Constructing dataset to separate quantum and classical model

In the main text, our central quantity of interest is the geometric difference g, which provides a quantification for a given data set, how large the prediction gap can be for possibly function or labels associated with that data. Here we detail how one can efficiently construct a function that saturates this bound for a given data set. This is the approach that is used in the main text to engineer the data set with maximal performance.

Given a (projected) quantum kernel $k^{\mathrm{Q}}(x_i, x_j) = \phi^{\mathrm{Q}}(x_i)^\dagger \phi^{\mathrm{Q}}(x_j)$ and a classical kernel $k^{\mathrm{C}}(x_i, x_j) = \phi^{\mathrm{C}}(x_i)^\dagger \phi^{\mathrm{C}}(x_j)$, our goal is to construct a dataset that would best separate the two models. Consider a dataset $(x_i, y_i), \forall i = 1, \ldots, N$. We use the model complexity $s = \sum_{i=1}^N \sum_{j=1}^N (K^{-1})_{ij} y_i y_j$ to quantify the generalization error of the model. The model complexity has been introduced in the main text, where a detailed proof relating s to prediction error is given in Section 8.6. To separate between quantum and classical model, we consider $s_{\mathrm{Q}} = 1$ and s_{C} is as large as possible for a particular choice of targets $y_1, \ldots, y_N$. To achieve this, we solve the optimization

$$\min_{y \in \mathbb{R}^N} \frac{\sum_{i=1}^N \sum_{j=1}^N ((K^{\mathrm{C}})^{-1})_{ij} y_i y_j}{\sum_{i=1}^N \sum_{j=1}^N ((K^{\mathrm{Q}})^{-1})_{ij} y_i y_j} \tag{8.90}$$

which has an exact solution given by a generalized eigenvalue problem. The solution is given by $y = \sqrt{K^{\mathrm{Q}}}\mathbf{v}$, where $\mathbf{v}$ is the eigenvector of $\sqrt{K^{\mathrm{Q}}}(K^{\mathrm{C}})^{-1}\sqrt{K^{\mathrm{Q}}}$ corresponding to the eigenvalue $g^2 = \left\|\sqrt{K^{\mathrm{Q}}}(K^{\mathrm{C}})^{-1}\sqrt{K^{\mathrm{Q}}}\right\|_\infty$. This guarantees that $s_{\mathrm{C}} = g^2 s_{\mathrm{Q}} = g^2$, and note that by definition of g, $s_{\mathrm{C}} \leq g^2 s_{\mathrm{Q}}$. Hence this dataset fully utilized the geometric difference between the quantum and classical space.

We should also include regularization parameter λ when constructing the dataset. Detailed discussion on model complexity s and geometric difference g with regularization is given in Section 8.8. Recall that for $\lambda > 0$,

$$s_C^\lambda = y^\dagger (\sqrt{K^{\mathrm{C}}}\left(K^{\mathrm{C}} + \lambda I\right)^{-2}\sqrt{K^{\mathrm{C}}})_{ij} y, \tag{8.91}$$

which is the model complexity that we want to maximize. Similar to the unregularized case, we consider the (unregularized) model complexity $s_{\mathrm{Q}} = y^\dagger (K^{\mathrm{Q}})^{-1} y$ to be one. Solving the generalized eigenvector problem yields the target $y = \sqrt{K^{\mathrm{Q}}}\mathbf{v}$, where $\mathbf{v}$ is the eigenvector of

$$\sqrt{K^{\mathrm{Q}}}\sqrt{K^{\mathrm{C}}}\left(K^{\mathrm{C}} + \lambda I\right)^{-2}\sqrt{K^{\mathrm{C}}}\sqrt{K^{\mathrm{Q}}} \tag{8.92}$$

with the corresponding eigenvalue

$$g_{\mathrm{gen}}^2 = \left\|\sqrt{K^{\mathrm{Q}}}\sqrt{K^{\mathrm{C}}}\left(K^{\mathrm{C}} + \lambda I\right)^{-2}\sqrt{K^{\mathrm{C}}}\sqrt{K^{\mathrm{Q}}}\right\|_\infty. \tag{8.93}$$

The larger λ is, the smaller g_{gen}^2 would be. In practice, one should choose a λ such that the training error bound $g_{\mathrm{tra}}^2 s_{\mathrm{Q}} = \lambda^2 \left\|\sqrt{K^{\mathrm{Q}}}(K^{\mathrm{C}} + \lambda I)^{-2}\sqrt{K^{\mathrm{Q}}}\right\|_\infty$ for the classical

ML model is small enough. In the numerical experiment, we choose a λ such that the training error bound $g_{\mathrm{tra}}^2 s_{\mathrm{Q}} \leq 0.002$ and g_{gen} is as large as possible. Finally, we can turn this dataset, which maps input x to a real value y, into a classification task by replacing y with +1 if $y > \operatorname{median}(y_1, \ldots, y_N)$ and -1 if $y \leq \operatorname{median}(y_1, \ldots, y_N)$.

The constructed dataset will yield the largest separation between quantum and classical models from a learning-theoretic sense, as the model complexity fully saturates the geometric difference. If there is no quantum advantage in this dataset, there will likely be none. We believe this construction procedure will eventually lead to the first quantum advantage in machine learning problems (classification problems to be more specific).

8.10 Lower bound on learning quantum models

In this section, we will prove a fundamental lower bound for learning quantum models stated in Theorem 42. This result says that in the worst case, the number N of training data has to be at least $\Omega(\mathrm{tr}(O^2)/\epsilon^2)$ when the input quantum state can be distributed across a sufficiently large Hilbert space. Quantum kernel method matches this lower bound. When the data spans over the entire Hilbert space, the dimension d will be large and the prediction error of the quantum kernel method given in Equation (8.40) becomes

$$\mathbb{E}_x |h^{\mathrm{Q}}(x) - \mathrm{tr}(O^U \rho(x))| \leq O\left(\sqrt{\frac{\mathrm{tr}(O^2)}{N}}\right). \tag{8.94}$$

Hence we can achieve ϵ error using $N \leq O(\mathrm{tr}(O^2)/\epsilon^2)$ matching the fundamental lower bound.

Theorem 42. *Consider any learning algorithm $\mathcal{A}$. Suppose for any unknown unitary evolution U, any unknown observable O with bounded Frobenius norm $\mathrm{tr}(O^2) \leq B$, and any distribution D over the input quantum states, the learning algorithm $\mathcal{A}$ could learn a function h such that*

$$\mathop{\mathbb{E}}_{\rho \sim D} |h(\rho) - \mathrm{tr}(OU\rho U^\dagger)| \leq \epsilon, \tag{8.95}$$

from N training data $(\rho_i, \mathrm{tr}(OU\rho_i U^\dagger)), \forall i = 1, \ldots, N$ with high probability. Then we must have

$$N \geq \Omega(B/\epsilon^2). \tag{8.96}$$

Proof. We select a Hilbert space with dimension $d = B/4\epsilon^2$ (this could be a subspace of an exponentially large Hilbert space). We define the distribution D to be the uniform distribution over the basis states $|x\rangle\langle x|$ of the d-dimensional Hilbert space. Then we consider the unknown unitary U to always be the identity, while the possible observables are

$$O_v = 2\epsilon \sum_{x=1}^{d} v_x \, |x\rangle\langle x| \, , \tag{8.97}$$

with $v_x \in \{\pm 1\}, \forall x = 1, \ldots, d$. There are hence 2^d different choices of observables O_v.

We now set up a simple communication protocol to prove the lower bound on the number of data needed. This is a simplified version of the proofs found in Refs.Haah et al., 2017; Huang, Richard Kueng, and Preskill, 2020. Alice samples an observable O_v uniformly at random from the 2^d possible choices. We can treat v as a bit-string of d entries. Then she samples N quantum states $|x_i\rangle\langle x_i| \, , \forall i = 1, \ldots, N$. Alice then gives Bob the following training data $\mathcal{T} = \{(|x_i\rangle\langle x_i| \, , \langle x_i| \, O_v \, |x_i\rangle) = v_{x_i}, \forall i = 1, \ldots, N\}$. Notice that the mutual information $I(v, \mathcal{T})$ between v and the training data $\mathcal{T}$ satisfies

$$I(s, \mathcal{T}) \leq N, \tag{8.98}$$

because the training data contains at most N values of s.

With high probability, the following is true by the requirement of the learning algorithm $\mathcal{A}$. Using the training data $\mathcal{T}$, Bob can apply the learning algorithm $\mathcal{A}$ to obtain a function f such that

$$\mathop{\mathbb{E}}_{\rho \sim D} |h(\rho) - \mathrm{tr}(OU\rho U^\dagger)| \leq \epsilon. \tag{8.99}$$

Using Markov's inequality, we have

$$\Pr[|h(\rho) - \mathrm{tr}(OU\rho U^\dagger)| < 2\epsilon] > \frac{1}{2}. \tag{8.100}$$

For all $x = 1, \ldots, d$, if $|h(|x\rangle\langle x|) - \mathrm{tr}(OU \, |x\rangle\langle x| \, U^\dagger)| < 2\epsilon$, we have $|h(|x\rangle\langle x|)/2\epsilon - v_x| < 1$. This means that if $h(|x\rangle\langle x|) > 0$, then $v_x = 1$ and if $h(|x\rangle\langle x|) < 0$, then $v_x = -1$. Hence Bob can construct a bit-string $\tilde{v}$ given as $\tilde{v}_x = \mathrm{sign}(h(|x\rangle\langle x|)), \forall x = 1, \ldots, d$. Using Equation (8.100), we know that at least $d/2$ bits in $\tilde{v}$ will be equal to v.

Because with high probability, $\tilde{v}$ and v has at least $d/2$ bits in common. Fano's inequality tells us that $I(v, \tilde{v}) \geq \Omega(d)$. Because the bit-string $\tilde{v}$ is constructed solely from the training data $\mathcal{T}$. Data processing inequality tells us that $I(v, \tilde{v}) \leq I(v, \mathcal{T})$. Together with Equation (8.98), we have

$$N \geq I(v, \mathcal{T}) \geq I(v, \tilde{v}) \geq \Omega(d). \tag{8.101}$$

Recall that $d = B/4\epsilon^2$, we have hence obtained the desired result $N \geq \Omega(B/\epsilon^2)$. □

8.11 Limitations of quantum kernel methods

Even though the quantum kernel method saturates the fundamental lower bound $\Omega(\mathrm{tr}(O^2)/\epsilon^2)$ and can be made formally equivalent to infinite depth quantum neural networks it has a number of limitations that hinder its practical applicability. In this section we construct a simple example where the overhead for using the quantum kernel method is exponential in comparison to trivial classical methods.

Specifically, it has the limitation of closely following this lower bound for any unitary U and observable O. This is not true for other machine learning methods, such as classical neural networks or projected quantum kernel methods. It is possible for classical machine learning methods to learn quantum models with exponentially large $\mathrm{tr}(O^2)$, which is not learnable by the quantum kernel method. This can already be seen in the numerical experiments given in the main text. In this section, we provide a simple example that allows theoretical analysis to illustrate this limitation.

We consider a simple learning task where the input vector $\mathbf{x} \in \{0, \pi\}^n$. The encoding of the input vector $\mathbf{x}$ to the quantum state space is given as

$$|\mathbf{x}\rangle = \prod_{k=1}^{n} \exp(\mathrm{i} X_k x_k) |0^n\rangle . \tag{8.102}$$

The quantum state $|\mathbf{x}\rangle$ is a computational basis state. We define $\rho(\mathbf{x}) = |\mathbf{x}\rangle\langle\mathbf{x}|$. The quantum model applies a unitary $U = I$, and measures the observable $O = I \otimes \ldots \otimes I \otimes Z$. Hence $f(\mathbf{x}) = \mathrm{tr}(O\rho(\mathbf{x})) = (2x_n - \pi)$. Notice that for this very simple quantum model, the function $f(\mathbf{x})$ is an extremely simple linear model. Hence a linear regression or a single-layer neural network can learn the function $f(\mathbf{x})$ from training data of size n with high probability.

Despite being a very simple quantum model, the Frobenius norm of the observable $\mathrm{tr}(O^2)$ is exponentially large, i.e., $\mathrm{tr}(O^2) = 2^n$. We now show that a quantum kernel method will need a training data of size $N \geq \Omega(2^n)$ to learn this simple function $f(\mathbf{x})$. Suppose we have obtained a training set given as $\{(\mathbf{x}_i, \mathrm{tr}(O\rho(\mathbf{x}_i))\}_{i=1}^{N}$ where each $\mathbf{x}_i$ is selected uniformly at random from $\{0, \pi\}^n$. Recall from the analysis in Section 8.6, the function learned by the quantum kernel method will be

$$h^{\mathrm{Q}}(\mathbf{x}) = \min\left(1, \max\left(-1, \sum_{i=1}^{N}\sum_{j=1}^{N} \mathrm{tr}(\rho(\mathbf{x}_i)\rho(\mathbf{x}))((K^{\mathrm{Q}} + \lambda I)^{-1})_{ij}\, \mathrm{tr}(O\rho(\mathbf{x}_j))\right)\right), \tag{8.103}$$

where $K^Q_{ij} = k^{\mathrm{Q}}(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{tr}(\rho(\mathbf{x}_i)\rho(\mathbf{x}_j))$. The main problem of the quantum kernel method comes from the precise definition of the kernel function $k(\mathbf{x}_i, \mathbf{x}) = \mathrm{tr}(\rho(\mathbf{x}_i)\rho(\mathbf{x}))$. For at least $2^n - N$ choices of $\mathbf{x}$, we have $\mathrm{tr}(\rho(\mathbf{x}_i)\rho(\mathbf{x})) = 0, \forall i = 1, \ldots, N$. This means that for at least $2^n - N$ choices of $\mathbf{x}$, $h^{\mathrm{Q}}(\mathbf{x}) = 0$. However, by construction, $f(\mathbf{x}) \in \{1, -1\}$. Hence the prediction error can be lower bounded by

$$\frac{1}{2^n} \sum_{\mathbf{x} \in \{0, \pi\}^n} |h^{\mathrm{Q}}(\mathbf{x}) - f(\mathbf{x})| \geq 1 - \frac{N}{2^n}. \tag{8.104}$$

Therefore, if $N < (1 - \epsilon)2^n$, then the prediction error will be greater than ϵ. Hence we need a training set of size $N \geq (1 - \epsilon)2^n$ to achieve a prediction error $\leq \epsilon$.

In general, when we place the classical vectors $\mathbf{x}$ into an exponentially large quantum state space, the quantum kernel function $\mathrm{tr}(\rho(\mathbf{x}_i)\rho(\mathbf{x}_j))$ will be exponentially close to zero for $\mathbf{x}_i \neq \mathbf{x}_j$. In this case K^{Q} will be close to the identity matrix, but $\mathrm{tr}(\rho(\mathbf{x}_i)\rho(\mathbf{x}))$ will be exponentially small. For a training set of size $N \ll 2^n$, $h^{\mathrm{Q}}(\mathbf{x})$ will be exponentially close to zero similar to the above example. Despite $h^{\mathrm{Q}}(\mathbf{x})$ being exponentially close to zero, if we can distinguish > 0 and < 0, then h^{Q} could still be useful in classification tasks. However, due to the inherent quantum measurement error in evaluating the kernel function $\mathrm{tr}(\rho(\mathbf{x}_i)\rho(\mathbf{x}_j))$ on a quantum computer, we will need an exponential number of measurements to resolve such an exponentially small difference.

8.12 Projected quantum kernel methods

In the main text, we argue that projection back from the quantum space to a classical one in the projected quantum kernel can greatly improve the performance of such methods. There we focused on the simple case of a squared exponential based on reduced 1-particle observables, however this idea is far more general. In this section we explore some of these generalizations including a novel scheme for calculating functions of all powers of RDMs efficiently.

From discussions on the quantum kernel method, we have seen that using the native quantum state space to define the kernel function, e.g., $k(x_i, x_j) = \mathrm{tr}(\rho(x_i)\rho(x_j))$ can fail to learn even a simple function when the full exponential quantum state space is being used. We have to utilize the entire exponential quantum state space otherwise the quantum machine learning model could be simulated efficiently classically and a large advantage could not be found. In this section, we will detail a set of solutions that project the quantum states back to approximate classical representations and define the kernel function using the classical representation. We refer to these modified quantum kernels as projected quantum kernels. The projected quantum kernels are defined in a classical vector space to circumvent the hardness of learning due to the exponential dimension in quantum Hilbert space. However, projected quantum kernels still use the exponentially large quantum Hilbert space for evaluation and can be hard to simulate classically.

Some simple choices based on reduced density matrices (RDMs) of the quantum state are given below.

1. A linear kernel function using 1-RDMs

   $$Q_l^1(x_i, x_j) = \sum_k \mathrm{Tr}\left[\mathrm{Tr}_{m \neq k}[\rho(x_i)]\mathrm{Tr}_{n \neq k}[\rho(x_j)]\right], \tag{8.105}$$

   where $\mathrm{tr}_m \neq k(\rho)$ is the partial trace of the quantum state ρ over all qubits except for the k-th qubit. It could learn any observable that can be written as a sum of one-body terms.

2. A Gaussian kernel function using 1-RDMs

   $$Q_g^1(x_i, x_j) = \exp\left(-\gamma \sum_k \left(\mathrm{Tr}_{m \neq k}[\rho(x_i)] - \mathrm{Tr}_{n \neq k}[\rho(x_j)]\right)^2\right), \tag{8.106}$$

   where $\gamma > 0$ is a hyper-parameter. It could learn any nonlinear function of the 1-RDMs.

3. A linear kernel using $k-$RDMs

   $$Q_l^k(x_i, x_j) = \sum_{K \in S_k(n)} \mathrm{Tr}\left[\mathrm{Tr}_{n \notin K}[\rho(x_i)]\mathrm{Tr}_{m \notin K}[\rho(x_j)]\right] \tag{8.107}$$

   where $S_k(n)$ is the set of subsets of k qubits from n, $\mathrm{Tr}_{n \notin K}$ is a partial trace over all qubits not in subset K. It could learn any observable that can be written as a sum of k-body terms.

The above choices have a limited function class that they can learn, e.g., Q_l^1 can only learn observables that are sum of single-qubit observables. It is desirable to define a kernel that can learn any quantum models (e.g., arbitrarily deep quantum neural networks) with sufficient amount of data similar to the original quantum kernel $k^{\mathrm{Q}}(x_i, x_j) = \mathrm{tr}(\rho(x_i)\rho(x_j))$ as discussed in Section 8.5.

We now define a projected quantum kernel that contains all orders of RDMs. Since all quantum models $f(x) = \mathrm{tr}(OU\rho(x)U^\dagger)$ are linear functions of the full quantum state, this kernel can learn any quantum models with sufficient data. A k-RDM of a quantum state $\rho(x)$ for qubit indices $(p_1, p_2, \ldots, p_k)$ can be reconstructed by local randomized measurements using the formalism of classical shadows Huang, Richard Kueng, and Preskill, 2020:

$$\rho^{(p_1,p_2,\ldots,p_k)}(x) = \mathbb{E}\left[\otimes_{r=1}^{k}(3\,|s_{p_r}, b_{p_r}\rangle\langle s_{p_r}, b_{p_r}| - I)\right], \tag{8.108}$$

where b_{p_r} is a random Pauli measurement basis X, Y, Z on the p_r-th qubit, and s_{p_r} is the measurement outcome ± 1 on the p_r-th qubit of the quantum state $\rho(x)$ under Pauli basis b_{p_r}. The expectation is taken with respect to the randomized measurement on $\rho(x)$. The inner product of two k-RDMs is equal to

$$\mathrm{Tr}\left[\rho^{(p_1,p_2,\ldots,p_k)}(x_i)\rho^{(p_1,p_2,\ldots,p_k)}(x_j)\right] = \mathbb{E}\left[\Pi_{r=1}^{k}(9\delta_{s^i_{p_r} s^j_{p_r}}\delta_{b^i_{p_r} b^j_{p_r}} - 4)\right], \tag{8.109}$$

where we used the fact that the randomized measurement outcomes for $\rho(x_i)$ and $\rho(x_j)$ are independent. We extend this equation to the case where some indices p_r, p_s coincide. This would only introduce additional features in the feature map $\phi(x)$ that defines the kernel $k(x_i, x_j) = \phi(x_i)^\dagger \phi(x_j)$. The sum of all possible k-RDMs can be written as

$$Q^k(\rho(x_i), \rho(x_j)) = \sum_{p_1=1}^{n} \cdots \sum_{p_k=1}^{n} \mathrm{Tr}\left[\rho^{(p_1,p_2,\ldots,p_k)}(x_i)\rho^{(p_1,p_2,\ldots,p_k)}(x_j)\right] \tag{8.110}$$

$$= \mathbb{E}\left[\left(\sum_{p=1}^{n}(9\delta_{s^i_p s^j_p}\delta_{b^i_p b^j_p} - 4)\right)^k\right], \tag{8.111}$$

where we used Equation (8.109) and linearity of expectation. A kernel function that contains all orders of RDMs can be evaluated as

$$Q^\infty_\gamma(\rho(x_i), \rho(x_j)) = \sum_{k=0}^{\infty} \frac{\gamma^k}{k! n^k} Q^k(\rho(x_i), \rho(x_j)) = \mathbb{E}\exp\left(\frac{\gamma}{n}\sum_{p=1}^{n}(9\delta_{s^i_p s^j_p}\delta_{b^i_p b^j_p} - 4)\right), \tag{8.112}$$

where γ is a hyper-parameter. The kernel function $Q^\infty_\gamma(\rho(x_i), \rho(x_j))$ can be computed by performing local randomized measurement on the quantum states $\rho(x_i)$ and $\rho(x_j)$ independently. First, we collect a set of randomized measurement data for $\rho(x_i), \rho(x_j)$ independently:

$$\rho(x_i) \to \{((s_1^{i,r}, b_1^{i,r}), \ldots, (s_n^{i,r}, b_n^{i,r})), \forall r = 1, \ldots, N_s\}, \tag{8.113}$$

$$\rho(x_j) \to \{((s_1^{j,r}, b_1^{j,r}), \ldots, (s_n^{j,r}, b_n^{j,r})), \forall r = 1, \ldots, N_s\}, \tag{8.114}$$

where N_s is the number of repetition for each quantum state. For each repetition, we will randomly sample a Pauli basis for each qubit and measure that qubit to obtain an outcome ±1. For the r-th repetition, the Pauli basis in the k-th qubit is given as $b_k^{i,r}$ and the measurement outcome ±1 is given as $s_k^{i,r}$. Then we compute

$$\frac{1}{N_s(N_s-1)} \sum_{r_1=1}^{N_s} \sum_{\substack{r_2=1 \\ r_2 \neq r_1}}^{N_s} \exp\left(\frac{\gamma}{n} \sum_{p=1}^{n} (9\delta_{s_p^{i,r_1} s_p^{j,r_2}} \delta_{b_p^{i,r_1} b_p^{j,r_2}} - 4) \right) \approx Q_\gamma^\infty(\rho(x_i), \rho(x_j)). \tag{8.115}$$

We reuse all pairs of data r_1, r_2 to reduce variance when estimating $Q_\gamma^\infty(\rho(x_i)$, since the resulting estimator would still be equal to the desired quantity in expectation. This technique is known as U-statistics, which is often used to create minimum-variance unbiased estimators. U-statistics is also applied in Huang, Richard Kueng, and Preskill, 2020 for estimating Renyi entanglement entropy with high accuracy.

8.13 Simple and rigorous quantum advantage

In Ref.Y. Liu, Arunachalam, and Temme, 2020, the authors proposed a machine learning problem based on discrete logarithm which is assumed to be hard for any classical machine learning algorithm, complementing existing work studying learnability in the context of discrete logs Servedio and Gortler, 2004; Sweke et al., 2020. Much of the challenge in their construction Sweke et al., 2020 was related to technicalities involved in the original quantum kernel approach. Here we present a simple quantum machine learning algorithm using the projected quantum kernel method. The problem is defined as follows, where p is an exponentially large prime number and g is chosen such that computing $\log_g(x)$ in $\mathbb{Z}_p^*$ is classically hard and $\log_g(x)$ is one-to-one.

Definition 13 (Discrete logarithm-based learning problem)**.** *For all input $x \in \mathbb{Z}_p^*$, where $n = \lceil \log_2(p) \rceil$, the output is*

$$y(x) = \begin{cases} +1, & \log_g(x) \in [s, s + \frac{p-3}{2}], \\ -1, & \log_g(x) \notin [s, s + \frac{p-3}{2}], \end{cases} \tag{8.116}$$

for some $s \in \mathbb{Z}_p^*$. The goal is to predict $y(x)$ for an input x sampled uniformly from $\mathbb{Z}_p^*$.

Let us consider the most straight-forward feature mapping that maps the classical input x into the quantum state space $|\log_g(x)\rangle$ using Shor's algorithm for computing discrete logarithms Michael A Nielsen and Isaac L Chuang, n.d.

Training the original quantum kernel method using this feature mapping will require training data $\{(x_i, y_i)\}_{i=1}^N$ with N being exponentially large to yield a small prediction error. This is because for a new $x \in \mathbb{Z}_p^*$, such that $\log_g(x) \neq \log_g(x_i), \forall i = 1, \ldots, N$, quantum kernel method will be equivalent to random guessing. Hence the quantum kernel method has to see most of the values in the range of $\log_g(x)$ ($\mathbb{Z}_p^*$) to make accurate predictions. This is the same as the example to demonstrate the limitation of quantum kernel methods in Section 8.11. Since $\mathbb{Z}_p^*$ is exponentially large, the quantum kernel method has to use an exponentially amount number of data N for this straight-forward feature map. The central problem is that all the inputs x are maximally far apart from one another, and this impedes the ability for quantum kernel methods to generalize.

On the other hand, we can project the quantum feature map $|\log_g(x)\rangle$ back to a classical space, which is now just a number $\log_g(x) \in \mathbb{Z}_p^*$. Recall that $\mathbb{Z}_p^*$ contains all number from $0, \ldots, p-1$, thus we consider mapping x to a real number $z = \log_g(x)/p \in [0, 1)$. Let us define $t = s/p$. In this projected space, we are learning a simple classification problem where $y(z) = +1$ if $z \in [t, t + \frac{p-3}{2p}]$, and $y(z) = -1$ if $z \notin [t, t + \frac{p-3}{2p}]$. We are using a periodic boundary where 0 and 1 are the same point. If $t + \frac{p-3}{2p} < 1$, then there exists some $a, b \in [0, 1)$ and $a < b$, such that $y(z) = +1$, if $a \leq z \leq b$, and $y(z) = -1$, otherwise. In this case we have $y(z) = \mathrm{sign}((b - z)(z - a))$, where $\mathrm{sign}(t) = +1$ if $t \geq 0$, otherwise $\mathrm{sign}(t) = -1$. If $t + \frac{p-3}{2p} \geq 1$, then there exists some $a, b \in [0, 1)$ and $a < b$, such that $y(z) = -1$, if $a \leq z \leq b$, and $y(z) = +1$, otherwise. In this case we have $y(z) = \mathrm{sign}((a - z)(z - b))$. Through this analysis, we can see that we only need to learn a simple quadratic function to perform accurate classification. Hence one could simply define a projected quantum kernel as

$$k^{\mathrm{PQ}}(x_i, x_j) = \Big((\log_g(x_i)/p)(\log_g(x_j)/p) + 1 \Big)^2, \tag{8.117}$$

where the division in $(\log_g(x_i)/p)$ is performed as real number in $\mathbb{R}$. This projected quantum kernel can efficiently learn any quadratic function $az^2 + bz + c$ with $z = \log_g(x_i)/p$, hence solving the above learning problem.

Theorem 43 (Corollary 3.19 in Mohri, Rostamizadeh, and Talwalkar, 2018)**.** *Let $\mathcal{H}$ be a class of functions taking values in $\{+1, -1\}$ with VC-dimension d. Then with probability $\geq 1 - \delta$ over the sampling of $z_1, \ldots z_N$ from some distribution $\mathcal{D}$,* *we have*

$$\mathop{\mathbb{E}}_{z\sim\mathcal{D}} I[h(z) \neq y(z)] \leq \frac{1}{N}\sum_{i=1}^{N} I[h(z_i) \neq y(z_i)] + \sqrt{\frac{2d\log(\mathrm{e}N/d)}{N}} + \sqrt{\frac{\log(1/\delta)}{N}}, \tag{8.118}$$

for all $h \in \mathcal{H}$, where $I[Statement] = 1$ if Statement is true, otherwise $I[Statement] =$ 0.

A simple and rigorous statement could be made by noticing that the VC-dimension Blumer et al., 1989; Mohri, Rostamizadeh, and Talwalkar, 2018 for the function class $\{\mathrm{sign}(az^2 + bz + c) | a, b, c \in \mathbb{R}\}$ is 3. Let us apply Theorem 43 with

$$z = \log_g(x)/p \text{ and } \mathcal{H} = \{\mathrm{sign}(az^2 + bz + c) | a, b, c \in \mathbb{R}\}. \tag{8.119}$$

This theorem bounds the prediction error for new inputs z coming from the same distribution as how the training data is sampled. For a given set of training data $(z_i, y(z_i))_{i=1}^{N}$, we perform a minimization over $a, b, c \in \mathbb{R}$ such that the training error $\frac{1}{N}\sum_{i=1}^{N} I[h(z_i) \neq y(z_i)]$ is zero. This can be achieved by applying a standard support vector machine algorithm Chang and C.-J. Lin, 2011b using the above kernel k^{PQ}, because $y(z_i) \in \mathcal{H}$, so one can always fit the training data perfectly. Using Eq. (8.118) with $\delta = 0.01$, we can provide a prediction error bound for the trained projected quantum kernel method

$$f_*(x) = h_*(\log_g(x)/p) = h_*(z) = \mathrm{sign}(a_* z^2 + b_* z + c_*). \tag{8.120}$$

Because we fit the training data perfectly, we have

$$\frac{1}{N}\sum_{i=1}^{N} I[h_*(z_i) \neq y(z_i)] = 0. \tag{8.121}$$

With probability at least 0.99, a projected quantum kernel method

$$f_*(x) = h_*(\log_g(x)/p) \tag{8.122}$$

that perfectly fit a data set of size $N = \mathcal{O}(\log(1/\epsilon)/\epsilon^2)$ has a prediction error

$$\mathop{\mathbb{P}}_{x\sim\mathbb{Z}_p^*}[f(x) \neq y(x)] \leq \epsilon. \tag{8.123}$$

This concludes the proof showing that the discrete logarithm-based learning problem can be solved with a projected quantum kernel method using a sample complexity independent of the input size n.

Despite the limitations of the quantum kernel method, the authors in Y. Liu, Arunachalam, and Temme, 2020 have shown that a clever choice of feature mapping $x \longrightarrow \rho(x)$ would also allow quantum kernels $\mathrm{tr}(\rho(x_i)\rho(x_j))$ to predict well in this learning problem.

8.14 Details of numerical experiments

Here we give the complete details for the numerical studies presented in the main text. For the input distribution, we focused on the fashion MNIST dataset H. Xiao, Rasul, and Vollgraf, 2017. We use principal component analysis (PCA) provided by scikit-learn Buitinck et al., 2013 to map each image (28×28 grayscale) into classical vectors $\mathbf{x}_i \in \mathbb{R}^n$, where n is the number of principal components. After PCA, we normalize the vectors $\mathbf{x}_i$ such that each dimension is centered at zero and the standard deviation is one. Finally, we sub-sample 800 data points from the dataset without replacement.

Embedding classical data into quantum states

The three approaches for embedding classical vectors $\mathbf{x}_i \in \mathbb{R}^n$ into quantum states $|\mathbf{x}_i\rangle$ are given below.

- **E1**: Separable encoding or qubit rotation circuit. This is a common choice in literature, e.g., see Schuld and Killoran, 2019; Skolik et al., 2020.

$$|\mathbf{x}_i\rangle = \bigotimes_{j=1}^{n} \mathrm{e}^{-\mathrm{i}X_j x_{ij}} |0^n\rangle, \tag{8.124}$$

  where x_{ij} is the j-th entry of the n-dim. vector $\mathbf{x}_i$, X_j is the Pauli-X operator acting on the j-th qubit.

- **E2**: IQP-style encoding circuit. This is an embedding proposed in Havlicek et al., 2019 that suggests a quantum advantage.

$$|\mathbf{x}_i\rangle = U_Z(\mathbf{x}_i) H^{\otimes n} U_Z(\mathbf{x}_i) H^{\otimes n} |0^n\rangle, \tag{8.125}$$

  where $H^{\otimes n}$ is the unitary that applies Hadamard gates on all qubits in parallel, and

$$U_Z(\mathbf{x}_i) = \exp\left(\sum_{j=1}^{n} x_{ij} Z_j + \sum_{j=1}^{n}\sum_{j'=1}^{n} x_{ij} x_{ij'} Z_j Z_{j'}\right), \tag{8.126}$$

  with Z_j defined as the Pauli-Z operator acting on the j-th qubit. In the original proposal Havlicek et al., 2019, $x \in [0, 2\pi]^n$, and they used $U_Z(\mathbf{x}_i) =$

$\exp\left(\sum_{j=1}^{n} x_{ij} Z_j + \sum_{j=1}^{n} \sum_{j'=1}^{n} (\pi - x_{ij})(\pi - x_{ij'}) Z_j Z_{j'}\right)$ instead. Here, due to the data pre-processing steps, $\mathbf{x}$ will be centered around 0 with a standard deviation of 1, hence we made the equivalent changes to the definition of $U_Z(\mathbf{x}_i)$.

- **E3**: A Hamiltonian evolution ansatz. This ansatz has been explored in the literature Wecker, Matthew B Hastings, and Troyer, 2015; Cade et al., 2019; Wiersema et al., 2020 for quantum many-body problems. We consider a Trotter formula with T Trotter steps (we choose $T = 20$) for evolving an 1D-Heisenberg model with interactions given by the classical vector $\mathbf{x}_i$ for a time t proportional to the system size (we choose $t = n/3$).

$$|\mathbf{x}_i\rangle = \left(\prod_{j=1}^{n} \exp\left(-\mathrm{i} \frac{t}{T} x_{ij} \left(X_j X_{j+1} + Y_j Y_{j+1} + Z_j Z_{j+1} \right) \right) \right)^{T} \bigotimes_{j=1}^{n+1} |\psi_j\rangle , \quad (8.127)$$

where X_j, Y_j, Z_j are the Pauli operators for the j-th qubit and $|\psi_j\rangle$ is a Haar-random single-qubit quantum state. We sample and fix the Haar-random quantum states $|\psi_j\rangle$ for every qubit.

Definition of original and projected quantum kernels

We use Tensorflow-Quantum Broughton, Verdon, McCourt, Antonio J Martinez, et al., 2020 for implementing the original/projected quantum kernel methods. This is done by performing quantum circuit simulation for the above embeddings and computing the kernel function $k(\mathbf{x}_i, \mathbf{x}_j)$. For quantum kernel, we store the quantum states $|\mathbf{x}_i\rangle$ as explicit amplitude vectors and compute the squared inner product

$$k^{\mathrm{Q}}(\mathbf{x}_i, \mathbf{x}_j) = |\langle \mathbf{x}_i \rangle \, \mathbf{x}_j|^2. \quad (8.128)$$

On actual quantum computers, we obtain the quantum kernel by measuring the expectation of the observable $|0^n\rangle\langle 0^n|$ on the quantum state $U_{\mathrm{emb}}(\mathbf{x}_j)^{\dagger} U_{\mathrm{emb}}(\mathbf{x}_i) |0^n\rangle$. For projected quantum kernel, we use the kernel function

$$k^{\mathrm{PQ}}(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \sum_{k} \sum_{P \in \{X,Y,Z\}} \left(\mathrm{tr}(P\rho(\mathbf{x}_i)_k) - \mathrm{tr}(P\rho(\mathbf{x}_j)_k) \right)^2 \right), \quad (8.129)$$

where P is a Pauli matrix and $\gamma > 0$ is a hyper-parameter chosen to maximize prediction accuracy. We compute the kernel matrix $K \in \mathbb{R}^{N \times N}$ with $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ using the sub-sampled dataset with $N = 800$ for both the original/projected quantum kernel.

Dimension and geometric difference

Following the discussion in Section 8.8, the approximate dimension of the original/projected quantum space is computed by

$$\sum_{k=1}^{N}\left(\frac{1}{N-k}\sum_{l=k}^{N} t_l\right), \tag{8.130}$$

where $N = 800$ and $t_1 \geq t_2 \geq \ldots \geq t_N$ are the singular values of the kernel matrix $K \in \mathbb{R}^{N\times N}$. Based on the discussion in Section 8.8, we report the minimum geometric difference g of the original/projected quantum space (we refer to both the original/projected quantum kernel matrix as $K^{\mathrm{P/Q}}$)

$$g_{\mathrm{gen}} = \sqrt{\left\| \sqrt{K^{\mathrm{P/Q}}}\sqrt{K^{\mathrm{C}}}\left(K^{\mathrm{C}} + \lambda I\right)^{-2}\sqrt{K^{\mathrm{C}}}\sqrt{K^{\mathrm{P/Q}}} \right\|_{\infty}}, \tag{8.131}$$

under a condition for having a small training error

$$g_{\mathrm{tra}} = \lambda\sqrt{\left\| \sqrt{K^{\mathrm{P/Q}}}(K^{\mathrm{C}} + \lambda I)^{-2}\sqrt{K^{\mathrm{P/Q}}} \right\|_{\infty}} < 0.045. \tag{8.132}$$

The actual value of g will depend on the list of choices for λ and classical kernels K^{C}. We consider the following list of λ

$$\lambda \in \{0.00001, 0.0001, 0.001, 0.01, 0.025, 0.05, 0.1\}, \tag{8.133}$$

and classical kernel matrix K^{C} being the linear kernel $k^{\ell}(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^{\dagger}\mathbf{x}_j$ or the Gaussian kernel $k^{\gamma}(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2)$ with hyper-parameter γ from the list

$$\gamma \in \{0.25, 0.5, 1.0, 2.0, 4.0, 8.0, 16.0, 32.0, 64.0\}/(n\,\mathrm{Var}[x_{ik}]) \tag{8.134}$$

for estimating the minimum geometric difference. $\mathrm{Var}[x_{ik}]$ is the variance of all the coordinates $k = 1, \ldots, n$ from all the data points $x_1, \ldots, x_N$. One could add more choices of regularization parameters λ or classical kernel functions, such as using polynomial kernels or neural tangent kernels, which are equivalent to training neural networks with large hidden layers (a package, called Neural Tangents Novak, L. Xiao, Hron, J. Lee, Alexander A. Alemi, et al., 2020, is available for use). This will provide a smaller geometric difference with the quantum state space, but all theoretical predictions remain unchanged.

Datasets

We include a variety of classical and quantum data sets.

1. **Dataset (C)**: For the original classical image recognition data set, i.e., Dataset (C) in Figure 8.3(b), we choose two classes, dresses (class 3) and shirts (class 6), to form a binary classification task. The prediction error (between 0.0 and 1.0) is equal to the portion of data that are incorrectly labeled.

2. **Dataset (Q, E1/E2/E3)**: For the quantum data sets in Figure 8.3(b), we consider the following quantum neural network

   $$U_{\text{QNN}} = \left(\prod_{j=1}^{n} \exp\left(-\mathrm{i} \frac{t}{T} J_j \left(X_j X_{j+1} + Y_j Y_{j+1} + Z_j Z_{j+1} \right) \right) \right)^T, \tag{8.135}$$

   where we choose $T = t = 10$ and $J_j \in \mathbb{R}$ are randomly sampled from the Gaussian distribution with mean 0 and standard deviation 1. We measure Z_1 after the quantum neural network, hence the resulting function is

   $$f(\mathbf{x}) = \mathrm{tr}(Z_1 U_{\text{QNN}} |\mathbf{x}\rangle\langle\mathbf{x}| U_{\text{QNN}}^{\dagger}). \tag{8.136}$$

   The mapping from $\mathbf{x}$ to $|\mathbf{x}\rangle$ depends on the feature embedding (E1, E2, or E3) discussed in Section 8.14. A different embedding $|\mathbf{x}\rangle$ corresponds to a different funtion $f(\mathbf{x})$, and hence would result in a different dataset. The prediction error for these datasets are the average absolute error with $f(\mathbf{x})$.

3. **Engineered datasets**: In Figure 8.4, we consider datasets that are engineered to saturate the potential of a quantum ML model. Given the choice of classical kernel K^{C} that has the smallest geometric difference g with a quantum ML model K^{Q}, we can create a data set that saturates $s_{\text{C}} = g^2 s_{\text{Q}}$ following the procedure in Section 8.9. In particular, we construct the dataset such that $s_{\text{Q}} = 1$ and $s_{\text{C}} = g^2$. We compute the eigenvector $\mathbf{v}$ corresponding to the maximum eigenvalue of

   $$\sqrt{K^{\text{Q}}}\sqrt{K^{\text{C}}} \left(K^{\text{C}} + \lambda I \right)^{-2} \sqrt{K^{\text{C}}}\sqrt{K^{\text{Q}}} \tag{8.137}$$

   and construct $\mathbf{y}' = \sqrt{K^{\text{Q}}}\mathbf{v} \in \mathbb{R}^N$. y_i' corresponds to a real number for data point $\mathbf{x}_i$. Finally we define the label of input data point $\mathbf{x}_i$ as

   $$y_i = \begin{cases} \mathrm{sign}(y_i'), & \text{with prob. } 0.9, \\ \text{random} \pm 1, & \text{with prob. } 0.1. \end{cases} \tag{8.138}$$

   This data set will show the maximal separation between quantum and classical ML model. The plots in Figure 8.4 uses engineered datasets generated by saturating the geometric difference of classical ML models and quantum ML models based on projected quantum kernels in Equation (8.129) under different embeddings (E1, E2, and E3). In Figure 8.5, we show the results for quantum ML models based on the original quantum kernels.

Classical machine learning models

We present the list of classical machine learning models that we compared with. We used scikit-learn Buitinck et al., 2013 for training the classical ML models.

- Neural network: We perform a grid search over two-layer feedforward neural networks with hidden layer size

  $$h \in \{10, 25, 50, 75, 100, 125, 150, 200\}. \tag{8.139}$$

  For classification, we use MLPClassifier.
  For regression, we use MLPRegressor.

- Linear kernel method: We perform a grid search over the regularization parameter

  $$C \in \{0.006, 0.015, 0.03, 0.0625, 0.125, 0.25, 0.5, 1.0, \tag{8.140}$$
  $$2.0, 4.0, 8.0, 16.0, 32.0, 64.0, 128.0, 256, 512, 1024\}. \tag{8.141}$$

  For classification, we use SVC with a linear kernel. For regression, we choose the best between SVR and KernelRidge (both using linear kernel).

- Gaussian kernel method: We perform a grid search over the regularization parameter

  $$C \in \{0.006, 0.015, 0.03, 0.0625, 0.125, 0.25, 0.5, 1.0, \tag{8.142}$$
  $$2.0, 4.0, 8.0, 16.0, 32.0, 64.0, 128.0, 256, 512, 1024\}. \tag{8.143}$$

  and kernel hyper-parameter

  $$\gamma \in \{0.25, 0.5, 1.0, 2.0, 3.0, 4.0, 5.0, 20.0\}/(n\,\mathrm{Var}[x_{ik}]). \tag{8.144}$$

  $\mathrm{Var}[x_{ik}]$ is the variance of all the coordinates $k = 1, \ldots, n$ from all the data points $\mathbf{x}_1, \ldots, \mathbf{x}_N$. For classification, we use SVC with RBF kernel (equivalent to Gaussian kernel). For regression, we choose the best between SVR and KernelRidge (both using RBF kernel).

- Random forest: We perform a grid search over the individual tree depth

  $$\text{max\_depth} \in \{2, 3, 4, 5\}, \tag{8.145}$$

  and number of trees

  $$\text{n\_estimators} \in \{25, 50, 100, 200, 500\}. \tag{8.146}$$

  For classification, we use RandomForestClassifier. For regression, we use RandomForestRegressor.

- Gradient boosting: We perform a grid search over the individual tree depth

  $$\text{max\_depth} \in \{2, 3, 4, 5\}, \tag{8.147}$$

  and number of trees

  $$\text{n\_estimators} \in \{25, 50, 100, 200, 500\}. \tag{8.148}$$

  For classification, we use GradientBoostingClassifier. For regression, we use GradientBoostingRegressor.

- Adaboost: We perform a grid search over the number of estimators

  $$\text{n\_estimators} \in \{25, 50, 100, 200, 500\}. \tag{8.149}$$

  For classification, we use AdaBoostClassifier.
  For regression, we use AdaBoostRegressor.

Quantum machine learning models

For training quantum kernel methods, we use the kernel function $k^{\mathrm{Q}}(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{tr}(\rho(\mathbf{x}_i)\rho(\mathbf{x}_j))$. For classification, we use SVC with the quantum kernel. For regression, we choose the best between SVR and KernelRidge (both using the quantum kernel). We perform a grid search over

$$C \in \{0.006, 0.015, 0.03, 0.0625, 0.125, 0.25, 0.5, 1.0, \tag{8.150}$$

$$2.0, 4.0, 8.0, 16.0, 32.0, 64.0, 128.0, 256, 512, 1024\}. \tag{8.151}$$

For training projected quantum kernel methods, we use the kernel function

$$k^{\mathrm{PQ}}(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \sum_k \sum_{P \in \{X,Y,Z\}} \big(\mathrm{tr}(P\rho(\mathbf{x}_i)_k) - \mathrm{tr}(P\rho(\mathbf{x}_j)_k)\big)^2\right), \tag{8.152}$$

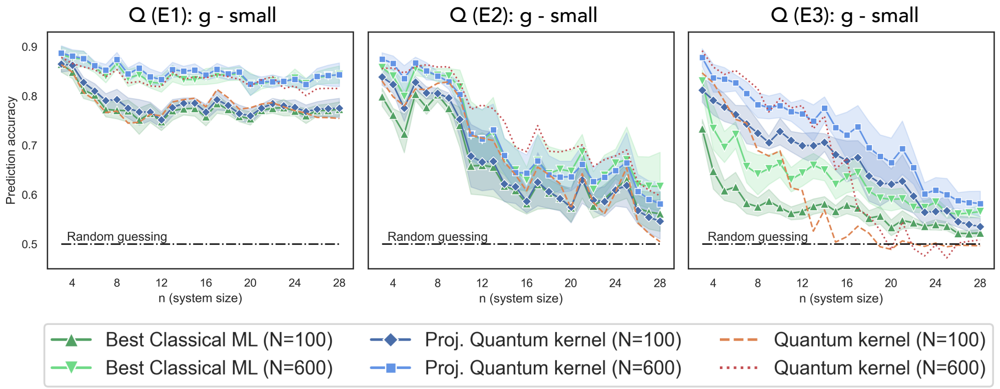


Figure 8.5: Prediction accuracy (higher the better) on engineered data sets. A label function is engineered to match the geometric difference $g(\mathrm{C}||\mathrm{QK})$ between the original quantum kernel and classical approaches. No substantial advantage is found using quantum kernel methods at large system size due to the small geometric difference $g(\mathrm{C}||\mathrm{QK})$. We consider the best performing classical ML models among Gaussian SVM, linear SVM, Adaboost, random forest, neural networks, and gradient boosting.

where P is a Pauli matrix. For classification, we use SVC with the projected quantum kernel $k^{\mathrm{PQ}}(\mathbf{x}_i, \mathbf{x}_j)$. For regression, we choose the best between SVR and KernelRidge (both using the projected quantum kernel $k^{\mathrm{PQ}}(\mathbf{x}_i, \mathbf{x}_j)$). We perform a grid search over

$$C \in \{0.006, 0.015, 0.03, 0.0625, 0.125, 0.25, 0.5, 1.0, \tag{8.153}$$

$$2.0, 4.0, 8.0, 16.0, 32.0, 64.0, 128.0, 256, 512, 1024\}. \tag{8.154}$$

and kernel hyper-parameter

$$\gamma \in \{0.25, 0.5, 1.0, 2.0, 3.0, 4.0, 5.0, 20.0\}/(n\,\mathrm{Var}[\mathrm{tr}(P\rho(\mathbf{x}_i)_k)]). \tag{8.155}$$

$\mathrm{Var}[\mathrm{tr}(P\rho(\mathbf{x}_i)_k)]$ is the variance of $\mathrm{tr}(P\rho(\mathbf{x}_i)_k)$ for all $P \in \{X, Y, Z\}$, all coordinates $k = 1, \ldots, n$, and all data points $x_1, \ldots, x_N$. We report the prediction performance under the best hyper-parameter for all classical and quantum machine learning models.

8.15 Additional numerical experiments

In the main text, we have presented engineered data sets to saturate the geometric inequality $s_{\mathrm{C}} \leq g(\mathrm{C}||\mathrm{PQ})^2 s_{\mathrm{PQ}}$ between classical ML and projected quantum kernel. As an additional experiment to see if the same approach can work with the original quantum kernel method, we can create similar engineered data sets that saturate the

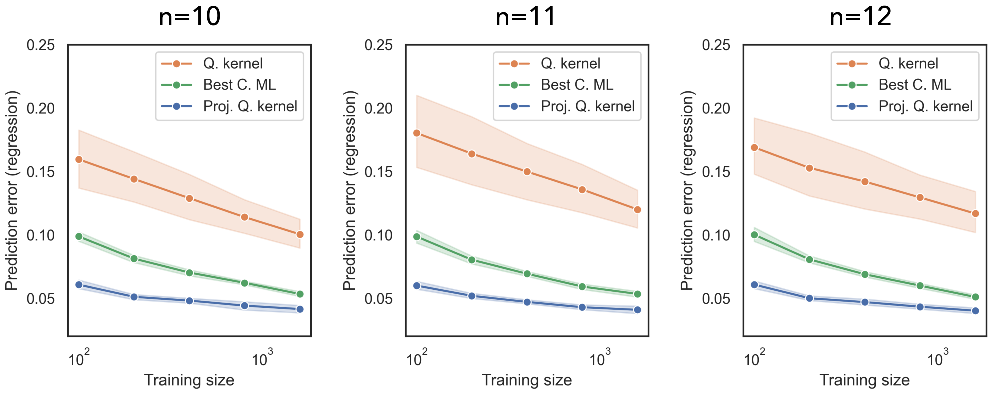


Figure 8.6: Prediction error (lower the better) on quantum data set (E2) over different training set size N. We can see that as the number of data increases, every model improves and the separation between them decreases.

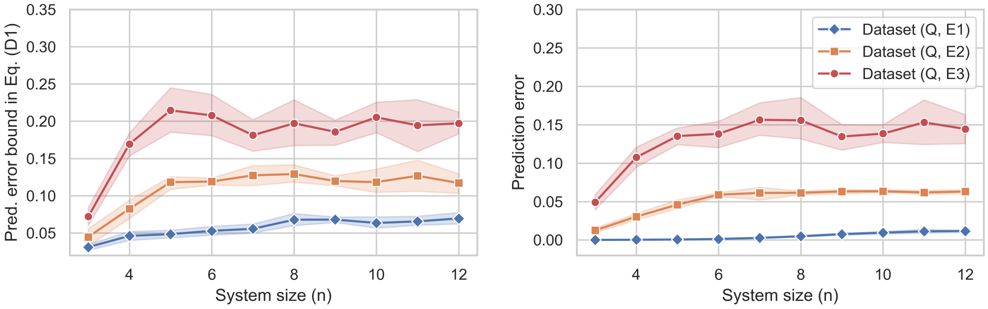


Figure 8.7: A comparison between the prediction error bound based on classical kernel methods (see Eq. (8.14)) and the prediction performance of the best classical ML model on the three quantum datasets. We consider the best-performing classical ML models among Gaussian SVM, linear SVM, Adaboost, random forest, neural networks, and gradient boosting. While the prediction error bound is an upper bound to the actual prediction error, the trends are very similar (a large prediction error bound gives a large prediction error).

geometric inequality between classical ML and quantum kernel The result is given in Figure 8.5. We can see that due to the large dimension d and small geometric difference $g(\mathrm{C}||\mathrm{Q})$ between classical ML and quantum kernel at large system size, there are no obvious advantage even for this best-case scenario. Interestingly, we see some advantage of projected quantum kernel over classical ML even when this data set is not constructed for projected quantum kernel.

In Figure 8.6, we show the prediction performance for learning a quantum neural network under a wide range for the number of training data N. We can see that there is a non-trivial advantage for small training size $N = 100$ when comparing projected quantum kernel and the best classical ML model. However, as training size N increases, every model will improve and the prediction advantage will shrink.

In Figure 8.7, we compare the prediction error bound $s_K(N)$ for classical kernel methods and the prediction performance of the best classical ML model (including a variety of classical ML models in Section 8.14). To be more precise, we consider different classical kernel functions and different regularization parameter λ. Then we compute

$$s_{K,\lambda}(N) = \sqrt{\frac{\lambda^2 \sum_{i=1}^N \sum_{j=1}^N ((K+\lambda I)^{-2})_{ij} y_i y_j}{N}} \tag{8.156}$$

$$+ \sqrt{\frac{\sum_{i=1}^N \sum_{j=1}^N ((K+\lambda I)^{-1} K (K+\lambda I)^{-1})_{ij} y_i y_j}{N}}. \tag{8.157}$$

This is a generalization of $s_K(N)$ described in the main text, where we consider regularized classical kernel methods with a regularization parameter λ to improve generalization performance (setting $\lambda = 0$ reduces to $s_K(N)$ given in the main text). See Section 8.6 for a detailed proof of an upper bound to the prediction error (note that the output label $y_i = \mathrm{tr}(O^U \rho(\mathbf{x}_i))$). We can see that while the prediction error bound and the actual prediction error has a non-negligible gap, the two figures follow a similar trend. When the prediction error bound is small, the prediction error of the best classical ML is also fairly small (and vice versa). It shows that $s_{K,\lambda}(N)$ is a good predictive metric for whether a classical ML model can learn to predict outputs from a quantum computation model.

Chapter 9

QUANTUM ADVANTAGE IN LEARNING FROM EXPERIMENTS

Humans learn about nature by doing experiments, but up until now, our ability to acquire knowledge has been hindered by viewing the quantum world through a classical lens. The rapid advance of quantum technology portends an opportunity to observe the world in a fundamentally different and more powerful way. Instead of measuring physical systems and then processing the classical measurement outcomes to infer properties of the physical systems, quantum sensors (Degen, Reinhard, and Cappellaro, 2017) will eventually be able to transduce (Lauk et al., 2020) quantum information in physical systems directly to a quantum memory (Lvovsky, Sanders, and Tittel, 2009; Dennis et al., 2002b), where it can be processed by a quantum computer. Figure 9.1A illustrates the distinction between conventional and quantum-enhanced experiments. For example, in a quantum-enhanced experiment, multiple photons might be captured and stored coherently at each node of a quantum network and then processed coherently to extract an informative signal (Gottesman, Jennewein, and Croke, 2012; Bland-Hawthorn, Sellars, and Bartholomew, 2021; Giovannetti, Lloyd, and Maccone, 2011). A key distinction between conventional and quantum-enhanced settings in the language of entanglement measurements is that the conventional setting may take arbitrary measurements, including entangled measurements, but is restricted to a single copy of a state at a time. In contrast, the quantum-enhanced setting may make entangled measurements between copies and hence requires sufficient memory to hold at least two copies of a state.

Recently, we have found that there exist properties of an n-qubit system that a quantum machine can learn efficiently, while the required number of conventional experiments to achieve the same task is exponential in n (Huang, Richard Kueng, and Preskill, 2021; Aharonov, J. S. Cotler, and Qi, 2021). This exponential advantage contrasts sharply with the quadratic advantage achieved in many previously proposed strategies for improving sensing using quantum technology (Degen, Reinhard, and Cappellaro, 2017). In this chapter, we analyze three classes of learning tasks with exponential quantum advantage and report on proof-of-principle experiments using up to 40 qubits on a Google Sycamore processor (Arute et al., 2019). These experiments confirm that a substantial quantum advantage can be realized even

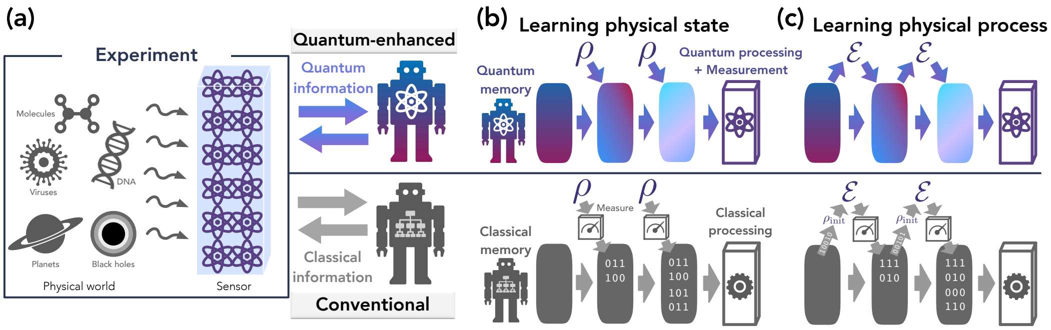


Figure 9.1: *Illustration of quantum-enhanced and conventional experiments.(a) Quantum-enhanced experiments versus conventional experiments.* Quantum-enhanced / conventional experiments interface with a quantum / classical machine running a quantum / classical learning algorithm that can store and process quantum / classical information. *(b) Learning physical state* ρ*.* Each experiment produces a physical state ρ. In the conventional setting, we measure each ρ to obtain classical data (the measurement could depend on prior measurement outcomes) and store the data in a classical memory. In the quantum-enhanced setting, ρ can coherently alter the quantum information stored in the memory of the quantum machine (illustrated by the change in color). With large enough quantum memory, the quantum machine can simply store each copy of ρ. After multiple rounds of experiments, quantum processing followed by a measurement is performed on the quantum memory. *(c) Learning physical process* $\mathcal{E}$*.* Each experiment is an evolution under $\mathcal{E}$. In the conventional setting, the classical machine specifies the input state to $\mathcal{E}$ using a classical bitstring and obtains classical measurement data (M. Mohseni, A. T. Rezakhani, and D. A. Lidar, 2008). In the quantum-enhanced setting, the evolution $\mathcal{E}$ coherently alters the memory of the quantum machine: the input state to $\mathcal{E}$ is entangled with the quantum memory in the quantum machine, and the output state is retrieved coherently by the quantum machine.

when the quantum memory and processor are both noisy.

To be more concrete, suppose that each experiment generates an n-qubit state ρ, and our goal is to learn some property of the quantum state ρ (Fig. 9.1). We depict *conventional* and *quantum-enhanced* experiments for this scenario in Fig. 9.1(b). In conventional experiments, each copy of ρ is measured separately, the measurement data is stored in a classical memory, and a classical computer outputs a prediction for the property after processing the classical data. In quantum-enhanced experiments, each copy of ρ is stored in a quantum memory, and then the quantum machine outputs the prediction after processing the quantum data in the quantum memory. We proved that for some tasks, the number of experiments needed to learn a desired property is exponential in n using the conventional strategy, but only polynomial in n using the quantum-enhanced strategy. While collective and entangled measurements strategies have been used to show separations in state identification tasks (Bennett et al., 1999), previous work focused on cases where single copies were separated and hence actually represent a weakened form of what we consider as conventional experiments here. This chapter further differentiates itself by ruling out even arbitrarily adaptive single-copy strategies and making connections to more complex learning tasks. We provide a more detailed comparison to these related works in Section 9.3. For suitably defined tasks, we could achieve exponential quantum advantage using a protocol as simple as storing two copies of ρ in quantum memory and performing an entangling measurement. We also showed that quantum-enhanced experiments have a similar exponential advantage in a related scenario shown in Fig. 9.1(c), in which the goal is to learn about a quantum process $\mathcal{E}$ rather than a quantum state ρ.

We proved that for a task that entails acquiring information about a large number of non-commuting observables, quantum-enhanced experiments could have an exponential advantage even when the measured quantum state is unentangled. This chapter here distinguishes itself from previous work by eliminating all dependencies on some exponential resource, a key requirement for enabling experimental demonstration. Conceptually, this also helps to distill the physical source of quantum advantage by proving an exponential advantage for simpler tasks than considered previously. Proving that an advantage exists for the simplest tasks one might consider enables exploration under experiment in more realistic conditions and bolsters confidence in future applications of these techniques. By performing experiments with up to 40 superconducting qubits, we show that this quantum advantage persisted even when using currently available quantum processors. We also demonstrated quantum advantage in learning the symmetry class of a physical evolution operator, inspired by recent theoretical advances (Aharonov, J. S. Cotler, and Qi, 2021; Sitan Chen, J. Cotler, et al., 2021b). Finally, in a theoretical contribution, we rigorously proved that quantum-enhanced experiments have an exponential advantage in learning about the principal component of a noisy state, as previously indicated in (Lloyd, Masoud Mohseni, and Rebentrost, 2014).

In our proof-of-principle experiments, we directly executed the state preparation or process to be learned within the quantum processor. In an actual application, the quantum data analyzed by the learning algorithm might be produced by an analog quantum simulator or a gate-based quantum computer. We also envision future applications in which quantum sensors equipped with quantum processors interact coherently with the physical world. The robustness of quantum advantage with respect to noise, validated by our experiments using a noisy superconducting device, boosts our confidence that the quantum-enhanced strategies described here can be exploited someday to achieve a substantial advantage in realistic applications.

9.1 Provable exponential quantum advantage

Here we present three classes of learning tasks and the associated quantum-enhanced experiments, each yielding a provable exponential advantage over conventional experiments. Each result is encapsulated by a theorem that we state informally. Precise statements and proofs are presented later in this chapter. Our experimental demonstrations are discussed in section Demonstrations of Quantum Advantage. The proofs proceed by representing a classical algorithm with a decision tree depicted at the center of the gray robot in Fig. 9.1. The tree representation encodes how the classical memory changes as we obtain more experimental data. We then analyzed how the transitions on the tree differ for distinct measured physical systems to provide rigorous information-theoretic lower bounds.

The first task concerns learning about a physical system described by an n-qubit state ρ. We suppose that each experiment generates one copy of ρ. In the conventional setting, we measure each copy of ρ to obtain classical data. The procedure can be adaptive, that is, each measurement can depend on the data obtained in earlier measurements. In the quantum-enhanced setting, a quantum computer can store each copy of ρ in a quantum memory, and act jointly on multiple copies of ρ. In both scenarios, we require all quantum data to be measured at the end of the learning phase of the procedure, so that only classical data survives. After the learning is completed, the learner is asked to provide an accurate prediction for the expectation value of one observable drawn from a set $\{O_1, O_2, \ldots\}$ where the number of observables in the set is exponentially large in n. The observables in the set can be highly incompatible, that is, each observable may fail to commute with many others in the set.

In prior work by some of the authors (Huang, Richard Kueng, and Preskill, 2021; Sitan Chen, J. Cotler, et al., 2021b), we required the learner to predict exponentially many observables, which is not possible in practice if the system size is large. In order to demonstrate the advantage in an actual device, we proved that predicting just the absolute value of one observable requires exponentially many copies in the conventional scenario. In contrast, predicting the entire set of observables can be achieved with a polynomial number of copies in the quantum-enhanced scenario. We thereby established the following constant versus exponential separation. The proof is given in Section 9.5.

Theorem 44 (Predicting observables)**.** *There exists a distribution over n-qubit states and a set of observables such that in the conventional scenario, at least order* 2^n *experiments are needed to predict the absolute value of one observable selected from the set, while a constant number of experiments suffice in the quantum-enhanced scenario.*

The exponential quantum advantage can occur even if the state ρ is unentangled. For example, in our experiments we consider $\rho \propto (I + \alpha P)$ where P is an n-qubit Pauli operator and $\alpha \in (-1, 1)$. This state can be realized as a probabilistic ensemble of product states, each of which is an eigenstate of P with eigenvalue α. Even if the state is known to be of this form, but P, α are unknown, the exponential separation between conventional and quantum-enhanced experiments persists. Moreover, the quantum advantage can be achieved by performing simple entangling measurements on pairs of copies of ρ. That the quantum advantage applies even when correlations among the n qubits are classical leads us to believe that the quantum-enhanced strategy will be beneficial in a broad class of sensing applications. In Section 9.10, we extended this theorem, showing that a sufficiently large quantum memory is needed to achieve this task in the quantum-enhanced scenario.

Our second machine learning task with a quantum advantage is quantum principal component analysis (PCA) (Lloyd, Masoud Mohseni, and Rebentrost, 2014). In this task, each experiment produces one copy of ρ, and our goal is to predict properties of the (first) principal component of ρ, namely the eigenstate $|\psi\rangle$ of ρ with the largest eigenvalue. For example, we may want to predict the expectation values of a few observables in the state $|\psi\rangle$. This task may become a valuable ingredient in future quantum-sensing applications. If an imperfect quantum sensor transduces a detected quantum state into quantum memory, the state is likely to be corrupted by noise. But it is reasonable to expect that properties of the principal component are relatively robust with respect to noise (Koczor, 2021b), and therefore highly informative about the uncorrupted state. To perform quantum PCA, a learning algorithm was introduced in Ref. (Lloyd, Masoud Mohseni, and Rebentrost, 2014) based on phase estimation which requires fault-tolerant quantum computers. One can also obtain information about the principal component of ρ using more near-term algorithms, such as virtual cooling (J. Cotler and Wilczek, 2020a), virtual distillation (Huggins et al., 2020; Koczor, 2021a), and variational algorithms (LaRose, Tikku, et al., 2019; Marco Cerezo et al., 2021).

Although the quantum PCA algorithm in (Lloyd, Masoud Mohseni, and Rebentrost, 2014) is exponentially faster than known algorithms based on conventional experiments, this advantage was not proven against all possible algorithms in the conventional scenario. Here, we rigorously established the exponential quantum advantage for performing quantum PCA. The exponential quantum advantage also holds in some of the near-term proposals (J. Cotler and Wilczek, 2020a; Huggins et al., 2020). The proofs are in Section 9.6.

Theorem 45 (Performing quantum PCA)**.** *In the conventional scenario, at least order $2^{n/2}$ experiments are needed to learn a fixed property of the principal component of an unknown n-qubit quantum state, while a constant number of experiments suffice in the quantum-enhanced scenario.*

It is worth commenting on recent results in Refs. (E. Tang, 2021; Chia, Gilyen, et al., 2020) showing that quantum PCA can be achieved by polynomial-time classical algorithms, which may seem to contradict Theorem 45. Those works assume the ability to access any entry of the exponentially large matrix ρ to exponentially high precision in polynomial time. Such a capability permits solving problems that even a quantum computer are not believed to efficiently solve (J. Cotler, Huang, and Jarrod R McClean, 2021), and in this case, we showed accurate evaluation of elements of ρ^k on single copies requires a precision growing with the dimension of the space. Achieving such a high precision requires measuring exponentially many copies of ρ, which takes an exponential number of experiments and exponential time. Hence, the assumptions of (E. Tang, 2021; Chia, Gilyen, et al., 2020) do not hold here. See Ref. (J. Cotler, Huang, and Jarrod R McClean, 2021) which provides a detailed exposition of these matters.

Another core task in quantum mechanics is understanding physical processes rather than states. Here, each experiment implements a physical process $\mathcal{E}$, and we can interface with $\mathcal{E}$ through a quantum / classical machine in the quantum-enhanced / conventional setting; see Fig. 9.1(c). We showed that a quantum machine can learn an approximate model of any polynomial-time quantum process $\mathcal{E}$ from only a polynomial number of experiments. Given a distribution on input states, the approximate model can predict the output state from $\mathcal{E}$ accurately on average. In

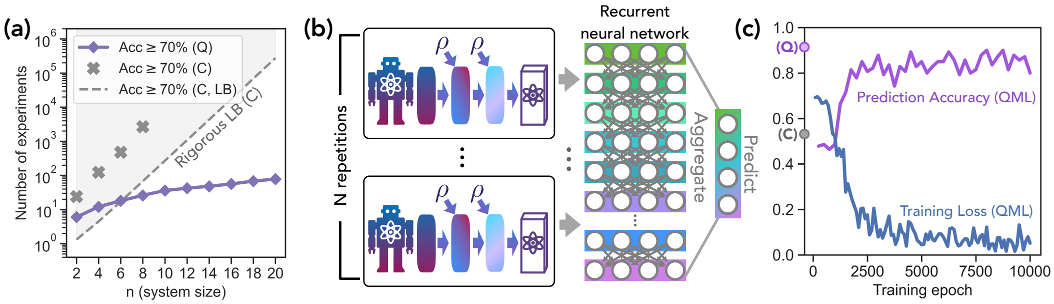


Figure 9.2: Quantum advantage in learning physical states. *(a) Quantum advantage in the number of experiments needed to achieve* $\geq 70\%$ *accuracy.* Here, (Q) corresponds to results running the best known strategy described in Section 9.5 for quantum-enhanced experiments and (C) corresponds to results running the best known conventional strategy. The dotted line is a lower bound for any conventional strategy (C, LB) as proven in Section 9.5. Even running on a noisy quantum processor, quantum-enhanced experiments are seen to vastly outperform the best theoretically achievable conventional results (C, LB). *(b) Supervised machine learning (ML) model based on quantum-enhanced experiments.* N repetitions of quantum-enhanced experiments are performed and the data is fed into a gated recurrent neural network (GRU) (J. Chung et al., 2014; D. Tang, Qin, and T. Liu, 2015). The neurons in the GRU are aggregated to predict an output. *(c) Training process of the supervised ML model.* We train the supervised ML model to determine which of two n-qubit Pauli operators has a larger magnitude for the expectation value in an unknown state ρ using noiseless simulation for small system sizes ($n < 8$). We consider the cross entropy (Murphy, 2012) as the training loss. Then we use the supervised ML model to make predictions using data from noisy quantum-enhanced experiments running on the Sycamore processor (Arute et al., 2019) for larger system sizes ($8 \leq n \leq 20$). We consider the probability to predict correctly as the prediction accuracy. The purple (Q) and gray (C) dots on the y-axis are the accuracy of the best known quantum-enhanced and conventional strategy considered in (a). Random guessing yields a prediction accuracy of 0.5.

contrast, we would need an exponential number of experiments to achieve the same task in the conventional setting. The proof for general quantum processes is given in Section 9.8.

Theorem 46 (Learning quantum processes)**.** *Suppose we are given a polynomial-time physical process* $\mathcal{E}$ *acting on* n *qubits and a probability distribution over* n*-qubit input states. In the conventional scenario, at least order* 2^n *experiments are needed to learn an approximate model of* $\mathcal{E}$ *that predicts output states accurately on average, while a polynomial number of experiments suffice in the quantum-enhanced scenario.*

9.2 Demonstrations of quantum advantage

The exponential quantum advantage captured by Theorems 44, 45, and 46 applies no matter how much classical processing power is leveraged in the conventional experiments. The conventional strategy fails because there is just no way to access enough classical data to perform the specified tasks, if the number of experiments is subexponential in n. But these exponential separations apply in an idealized setting where quantum states are stored and processed perfectly. Will access to quantum memory unlock a substantial quantum advantage under more realistic conditions?

For two different tasks, we have investigated the robustness of the quantum advantage by conducting experiments using a superconducting quantum processor. We consider specialized tasks that maintain exponential quantum advantage and have better noise robustness than the general tasks described in the previous section. The first task we studied pertains to Theorem 44. The task is to approximately estimate the magnitude for the expectation value of Pauli observables. The unknown state is an unentangled n-qubit state $\rho = 2^{-n}(I + \alpha P)$, where $\alpha = \pm 0.95$, P is a Pauli operator, and both α, P are unknown. After all measurements are completed and learning is terminated, two distinct Pauli operators Q_1 and Q_2 are announced, one of which is P and the other of which is not equal to P. We ask the machine to determine which of $|\mathrm{tr}(Q_1\rho)|$ and $|\mathrm{tr}(Q_2\rho)|$ is larger.

In the conventional scenario, where copies of ρ are measured one by one, the best known strategy is to use randomized Clifford measurements requiring an exponential number of copies to achieve the task with reasonable success probability (Huang, Richard Kueng, and Preskill, 2020; Huang, Richard Kueng, and Preskill, 2021). In the quantum-enhanced scenario, copies of ρ are deposited in quantum memory two at a time, and a Bell measurement across the two copies is performed to extract a snapshot of the state. We consider two data analysis approaches. The first approach considers a specialized formula for estimating $|\mathrm{tr}(Q\rho)|$ given in Section 9.5. Figure 9.2A depicts, as a function of the system size n, the number of experiments needed in each scenario to achieve 70% prediction accuracy. We show the experimental results when using conventional and quantum-enhanced experiments, along with a theoretical lower bound on the number of experiments needed in the conventional scenario as proven in Section 9.5. The first approach is explicitly tailored to the problem structure, which can limit its applicability to other problems. Instead of having a specialized formula for analyzing the snapshot, the second approach simply feeds the snapshot to a supervised machine learning (ML)

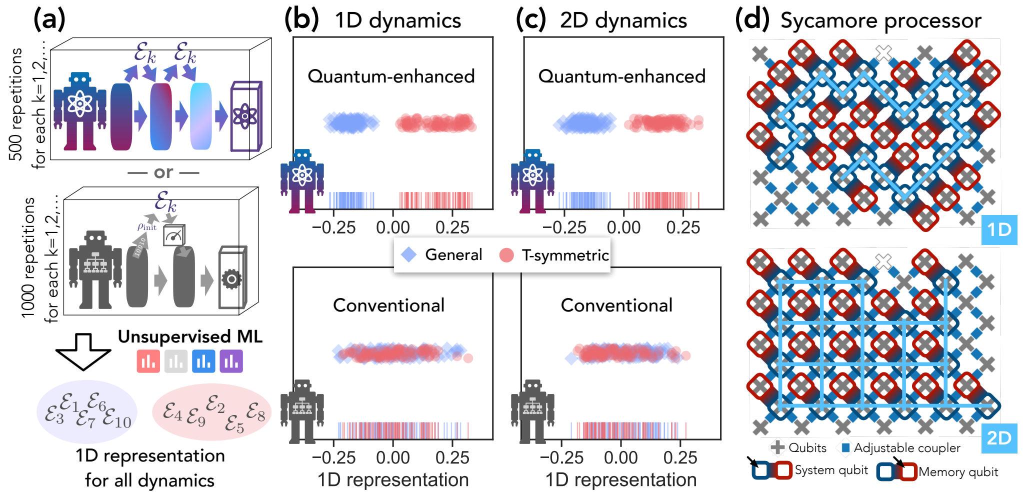


Figure 9.3: Quantum advantage in learning physical dynamics. *(a) Unsupervised machine learning (ML) model.* We perform 500 repetitions of quantum-enhanced experiments (each accessing $\mathcal{E}_k$ twice) for every physical process $\mathcal{E}_k$, and feed the data into an unsupervised ML model (Gaussian kernel PCA (Schölkopf, A. Smola, and Müller, 1998)) to learn a one-dimensional representation for describing distinct physical dynamics $\mathcal{E}_1, \mathcal{E}_2, \ldots$. Similarly, we also consider applying unsupervised ML to data obtained from 1000 repetitions of the best known conventional experiments (each accessing $\mathcal{E}_k$ once) for every physical process $\mathcal{E}_k$. *(b) Representation learned by unsupervised ML for 1D dynamics.* Each point corresponds to a distinct physical process $\mathcal{E}_k$. The vertical line at the bottom shows the exact 1D representation of each $\mathcal{E}_k$. Half of the processes satisfy time-reversal symmetry (blue diamonds) while the other half of them do not (red circles). When fed with data from quantum-enhanced experiments, the ML model accurately discovers the underlying symmetry pattern. In contrast, the ML model fails to do so when fed with data from conventional experiments. *(c) Representation learned by unsupervised ML for 2D dynamics. (d) The geometry implemented on the Sycamore processor (Arute et al., 2019).* We consider two different classes of connectivity geometry for implementing 1D (top) and 2D (bottom) dynamics.

model based on a recurrent neural network (J. Chung et al., 2014; D. Tang, Qin, and T. Liu, 2015; Goodfellow, Bengio, and Courville, 2016) to make a prediction, as depicted in Figure 9.2B. The use of ML methods is designed to highlight the fact that while a specialized formula is available for this problem, knowing or precisely tuning to that solution is not required. Indeed, the use of an RNN trained at smaller sizes shows that the data is so clear, the task can even be learned at smaller sizes and automatically generalized to larger sizes, even in the presence of experimental noise. We train the neural network using noiseless simulation data for small system sizes ($n < 8$). Then we use the neural network to make predictions when we are provided with experimental data for large system sizes ($8 \leq n \leq 20$). We report the prediction accuracy, which is equal to the probability for correctly answering whether $|\mathrm{tr}(Q_1\rho)|$ or $|\mathrm{tr}(Q_2\rho)|$ is larger. Figure 9.2C shows the performance of the ML model as we train the neural network. Despite the noisy storage and processing in the experimental device, we observed a significant quantum advantage using both the specialized and the machine learning approaches.

The second task we studied, which pertains to Theorem 46, was inspired by the recent observation that quantum-enhanced experiments can efficiently identify the symmetry class of a quantum evolution operator, while conventional experiments cannot (Aharonov, J. S. Cotler, and Qi, 2021; Sitan Chen, J. Cotler, et al., 2021b). An unknown n-qubit quantum evolution operator is presented, drawn either from the class of all unitary transformations, or from the class of time-reversal-symmetric unitary transformations (i.e., real orthogonal transformations). We consider whether an unsupervised ML can learn to recognize the symmetry class of the unknown evolution operator based on data obtained from either quantum-enhanced experiments or conventional experiments. An illustration is shown in Figure 9.3A.

In the conventional scenario, we repeatedly apply the unknown evolution operator to the initial state $|0\rangle^{\otimes n}$, and then measure each qubit of the output state in the Y-basis. Under T-symmetric evolution the output state has purely real amplitudes; hence the expectation value of any purely imaginary observable, such as the Pauli-Y operator, is always zero. In contrast, the expectation value of Y after general unitary evolution is generically nonzero, but may be exponentially small and hence hard to distinguish from zero. In the quantum-enhanced scenario, we make use of n additional memory qubits. We prepare an initial state in which the n system qubits are entangled with the n memory qubits, evolve the system qubits under the unknown evolution operator, swap the system and memory qubits, evolve the system qubits again, and finally perform n Bell measurements, each acting on one system qubit and one memory qubit.

Each evolution operator is a one-dimensional or two-dimensional n-qubit quantum circuit as shown in Fig. 9.3(d). After sampling many different evolution operators from both symmetry classes (and obtaining data from each sampled evolution multiple times), we used an unsupervised ML model (kernel PCA (Schölkopf, A. Smola, and Müller, 1998)) to find a one-dimensional representation of the evolution operators. The representations learned by the unsupervised ML model are shown in Figures 9.3(b, c). The use and success of an unsupervised ML model for this highlights that the data from these generic two-copy measures are clear enough to discover a previously unknown phenomena, where specialized measurements or even training labels are not required. The desired data in the quantum-enhanced case is so clear that it is analogous to comparing a picture of a cat to a white background whereas the conventional scenario has access to at best blurred cat images. In Section 9.4, we present results using the best known specialized data-processing approach, which yield the same conclusions.

Using the quantum-enhanced data, the ML model discovers a clean separation between the two symmetry classes, while there is no discernable separation into classes when using data from conventional experiments. The signal from the quantum-enhanced experiments was strong enough that the two classes were easily recognized without access to any labeled training data.

9.3 Related works

We begin with existing works that study the separation between conventional and quantum-enhanced strategies for learning physical systems and dynamics. In (Bubeck, Sitan Chen, and J. Li, 2020), they establish a polynomial separation between conventional and quantum-enhanced strategies for testing if a state is maximally mixed or not. In (Huang, Richard Kueng, and Preskill, 2021; Aharonov, J. S. Cotler, and Qi, 2021; Sitan Chen, J. Cotler, et al., 2021b), exponential separations between conventional and quantum-enhanced strategies are established for tasks regarding the learning of physical systems and dynamics. However, all the tasks studied in (Huang, Richard Kueng, and Preskill, 2021; Aharonov, J. S. Cotler, and Qi, 2021; Sitan Chen, J. Cotler, et al., 2021b) contain components that require exponential resources. In order to perform a demonstration in a physical experiment, this work shows that many of these exponential resources can actually be improved to polynomial. For predicting highly-incompatible properties, (Huang, Richard Kueng, and Preskill, 2021; Sitan Chen, J. Cotler, et al., 2021b) require checking an exponential number of observables to demonstrate the exponential advantage. In this thesis, we show that checking a constant number of carefully chosen observables is sufficient to establish the exponential advantage. For purity testing, (Aharonov, J. S. Cotler, and Qi, 2021; Sitan Chen, J. Cotler, et al., 2021b) requires that the target state one would like to learn about is generated by an exponentially deep random quantum circuit. In this thesis, we establish the exponential advantage for states that are only of polynomial complexity by using techniques from pseudo-random state construction. Furthermore, this work provides a new reduction from purity testing to quantum principal component analysis (PCA), hence establishing the exponential advantage in quantum PCA. We also provide a new task on approximate learning of polynomial-time quantum processes that yield exponential advantage.

We also mention some relevant works that study other classes of strategies for learning or characterizing physical systems and dynamics. It was shown in Ref. (M. Mohseni and D. A. Lidar, 2007) that the dynamics of open quantum systems with dimension d^n, where d is a prime, can be fully reconstructed with a quadratically fewer experiments over conventional quantum process tomography, with a quantum-enhanced strategy consisting of n auxiliary systems of same dimensions d and performing generalized Bell-sate preparations and generalized Bell-state measurements. The results in (Haah et al., 2017) give a polynomial separation between a restricted class of conventional strategies and quantum-enhanced strategies for learning the complete description of a quantum state. The results in (Huang, Richard Kueng, and Preskill, 2020) give an exponential separation between a restricted class of conventional strategies and quantum-enhanced strategies for learning to predict properties of a quantum state. Ref. (Senrui Chen, S. Zhou, et al., 2021) establishes an exponential separation between ancilla-free strategies and ancilla-assisted strategies for learning the eigenvalues in Pauli channels. Ref. (Sitan Chen, J. Cotler, et al., 2021a) gives an exponential separation between restricted quantum-enhanced strategies and quantum-enhanced strategies for learning about a quantum state. Ref. (Anshu, Landau, and Y. Liu, 2021) considers a problem on learning two spatially separated quantum states using local quantum learning algorithms and give an exponential separation between having a quantum or a classical communication channel between the local quantum learning algorithms. In Refs. (Coudron and Menda, 2020; Chia, K.-M. Chung, and Lai, 2020), an exponential separation between two bounded-depth quantum learning algorithms are given for learning about an exponential-time quantum process.

We briefly discuss how the learning problems considered in this thesis relate to concepts in machine learning. Supervised learning, unsupervised learning, and PAC learning (Leslie G Valiant, 1984) consider the setting when the data has already been gathered. In this thesis, we consider a learning algorithm that can perform new experiments to actively gather data in order to maximize its information about the physical world. This learning setting is closer to active learning (Settles, 2009) and reinforcement learning (Sutton and Barto, 2018). Active learning considers learning agents that can actively gather new data (such as by doing new experiments), while reinforcement learning considers learning agents that can perform action to gather information from the environment in order to maximize some reward function (such as its knowledge about the world). Both aspects are relevant in the learning problems we consider. For example, the exponential lower bound we established for conventional strategies shows that any learning agent that can actively conduct new conventional experiments can only learn about the unknown state or dynamics after an exponential number of experiments. We believe existing techniques in active learning and reinforcement learning will also be relevant for the future development of this line of research.

There is also a large body of works that consider the separation between various classes of restricted protocols for learning states and unitaries, e.g., see (Bennett et al., 1999; Bisio et al., 2010; Slussarenko et al., 2017; Laneve et al., 2021; Sentis, Martinez-Vargas, and Munoz-Tapia, 2022) and the references therein. (Bennett et al., 1999) shows that given a *bipartite* state from a special class of states, a measurement on the state can identify the state accurately, but any sequence of local operations and classical communications (LOCC) between two separate observers residing the two parts of the system cannot identify the state accurately. The result in (Bennett et al., 1999) demonstrates the advantage of having quantum network that can bring a multipartite quantum state that is distributed across different physical locations to a single place and perform a measurement on the state. An experimental demonstration is given in (Laneve et al., 2021). In this chapter, we do not assume that the quantum state is distributed across different physical locations. Hence, we consider conventional experiments to be those that can conduct arbitrary measurements on the unknown state to extract classical information. In this point of view, the LOCC strategy is a more restricted class of conventional experiments, and (Bennett et al., 1999) separates the more restricted class from the general class. (Slussarenko et al., 2017) studies the ability to better distinguish different quantum states by performing entangled measurements on multiple copies of an unknown state. (Slussarenko et al., 2017) shows that entangled measurements (similar to our quantum-enhanced experiments) can be better than some incoherent measurement procedures (similar to our conventional experiments). However, (Slussarenko et al., 2017) did not establish a provable advantage over all possible incoherent measurement procedures. (Bisio et al., 2010) considers learning unitary and shows that incoherent strategies (similar to our conventional experiments) are optimal when the unknown unitary is sampled uniformly from a group. So for this setting, coherent strategies (similar to our quantum-enhanced experiments) do not provide an advantage. The argument in (Bisio et al., 2010) relies on the group structure and does not apply when the unknown unitary is not uniformly sampled from a group. This work provides new techniques to rigorously establish the exponential advantage of coherent strategies (quantum-enhanced experiments) over incoherent strategies (conventional experiments). We believe that these techniques could also be helpful for the development of these existing research programs.

9.4 Experimental details

In this section, we present the details of the physical experiments run on the superconducting processor as well as the supervised/unsupervised machine learning models used to analyze the data.

General description

The experiments were performed on a Google Sycamore processor containing up to 53 superconducting transmon qubits. The largest error source in the Sycamore processor (Arute et al., 2019) is qubit readout error, which ranges from 3% to 7%. The second largest error source is the two-qubit gate with an error around 0.5% to 1.5%. Single-qubit gates have the smallest error around 0.05% to 0.5%. The Sycamore chip was introduced in Ref. (Arute et al., 2019), where additional details concerning the hardware implementation and performance can be found. The Sycamore chip was controlled remotely using an internal cloud interface programmed using Cirq (Developers, 2021) and TensorFlow Quantum (Broughton, Verdon, McCourt, Antonio J. Martinez, et al., 2021). The layout of the chip including connectivity is depicted in Supp. Fig. 9.4(a).

For all our experiments on learning about states and dynamics, the total number of qubits was varied from 4 to 40 qubits (where in each case half of the qubits are used to simulate a physical system). As the system size was varied, the subset of qubits used for the experiments was varied in order to maximize experimental performance and minimize any overhead related to the 2D connectivity. That is, the largest contiguous patches with low gate and measurement error rates were selected, and swap operations were used to meet connectivity requirements when necessary. The experimental requirements for learning states, 1D dynamics, and 2D dynamics differ considerably in their experimental complexity.

In all experiments, the implementations of the unknown states or dynamics are performed in the quantum processor, where the learning algorithms do not know about them. While this is only an emulation of the process of data collection from a physical system in an actual sensing experiment, it allows us to examine the proposed pipeline for quantum data processing in a situation where data collection is imperfect.

Experiments on learning physical states

We separate this subsection into the concrete procedure for generating the unknown states, the conventional experiments we run, the quantum-enhanced experiments we run, and the supervised neural network model for making prediction based on data from quantum-enhanced experiments.

Procedure for generating the class of unknown states

The state preparation we consider is relatively simple in that the unknown state ρ is unentangled, but has strong non-local classical correlations. In the experimental demonstration we consider states of the form $\rho = 2^{-n}(I + \alpha P)$, where $\alpha \in \{-0.95, 0.95\}$, $P = \bigotimes_{i=1}^{n} P_i$ is an n-qubit Pauli operator, and both α and P are unknown. The state $\rho = 2^{-n}(I + \alpha P)$ is prepared by a randomized constant-depth circuit described in the following. To generate one copy of ρ, we introduce a parameter $\eta = \mathrm{sign}(\alpha)$. Then for each qubit $i = 1, \ldots, n$, we do the following.

1. If $P_i = I$, then we set qubit i to be $|0\rangle\langle 0|$ with probability $1/2$, and be $|1\rangle\langle 1|$ with probability $1/2$.
2. If $P_i \neq I$ and there exists $j > i$ such that $P_j \neq I$, then we set qubit i to be one of the two eigenstates of P_i with equal probability. We multiply η by the eigenvalue (+1 or −1) of the selected eigenstate of P_i.
3. If $P_i \neq I$ and there does not exist $j > i$ such that $P_j \neq I$, then we use the following procedure.
   a) With probability 0.05, we set qubit i to be either $|0\rangle\langle 0|$ or $|1\rangle\langle 1|$ with equal probability.
   b) With probability 0.95, we set qubit i to be the positive eigenstate of P_i if $\eta = +1$, and set qubit i to be the negative eigenstate of P_i if $\eta = -1$.

By construction, the density operator prepared by this procedure is realized as an ensemble of pure states, where each pure state is a tensor product of Pauli operator eigenstates. Therefore, there is no quantum entanglement across different qubits. Furthermore, step 3 is designed to assure that $|\,\mathrm{tr}\,(P\rho)\,| = 0.95$.

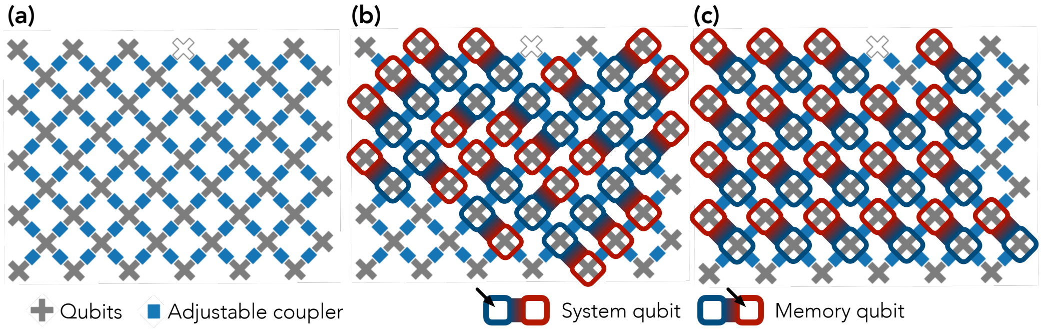


Figure 9.4: *(a) Layout of a Google Sycamore processor.* There is a total of 53 superconducting transmon qubits (the qubit corresponding to an empty cross is out of order). The blue rectangles show the adjustable couplers that can apply the entangling two-qubit gate SYC to neighboring qubits. *(b) Layout used for learning states and for learning 1D dynamics.* We partition the 40 qubits into 20 system qubits and 20 memory qubits. Either an unknown state of the system qubits is prepared, or an unknown process is applied to the system qubits. *(c) Layout used for learning 2D dynamics.*

Conventional experiments

In the conventional setting, the optimal strategy (up to logarithmic factors) for estimating expectation values of high-weight n-qubit Pauli observables uses classical shadow tomography based on randomized Clifford measurements (Huang, Richard Kueng, and Preskill, 2020). Using this strategy, in each experiment we randomly sample a unitary transformation from the Clifford group, apply the sampled transformation to the unknown state ρ, and then measure in the computational basis. Although such randomized Clifford measurements can be executed using quantum circuits of polynomial size, the required circuits are too large to be performed accurately with today's noisy quantum devices except for quite modest values of n. Furthermore, the classical post-processing of the measurement results has complexity exponential in n.

In our conventional experiments, because randomized Clifford measurements are infeasible we instead use classical shadow tomography based on randomized Pauli measurements (Huang, Richard Kueng, and Preskill, 2020). Using this strategy, in each experiment, we randomly sample from depth-1 Clifford circuits, apply the sampled circuit to ρ, and then measure in the computational basis. That is, for each of the n qubits, we decide uniformly at random to measure one of the three Pauli observables X, Y, or Z. Many such measurements are performed, each time on a new copy of ρ, and the classical data collected is post-processed to predict expectation values of observables in the state ρ.

Classical shadow tomography based on randomized Pauli measurements is a powerful technique that enables classical ML models to predict quantum many-body ground states and quantum phases of matter with rigorous guarantees (Huang, Richard Kueng, Torlai, et al., 2022). However, for the task of estimating the expectation value of an n-qubit Pauli observable that is announced after all measurements are completed, both randomized Clifford and randomized Pauli measurements require a number of experiments that scale exponentially in n, as we have proven in Section 9.5. Likewise, exponentially many experiments are needed to perform Task 2 defined in Section 9.5 with high success probability. In Fig. 9.2(a), we report the number of experiments required to achieve a prediction accuracy of at least 70% for different system sizes. We consider a maximum of 5000 experiments. For system size $n \geq 10$, we are unable to achieve at least 70% prediction accuracy with 5000 experiments. Hence we only show system size $n = 2, 4, 6, 8$ for conventional experiments in Fig. 9.2(a). In Fig. 9.2(c), we report the average prediction accuracy (the probability of performing Task 2 successfully) over system sizes $n = 8, 10, 12, 14, 16, 18, 20$. For each n-qubit state ρ, we conduct 1000 experiments to obtain the measurement data. The prediction accuracy is indicated by the gray point shown on the vertical axis, which is only slightly better than random guessing (0.5). For system sizes $n \geq 10$, the prediction accuracy is very close to 0.5. Classical shadow tomography based on randomized Pauli measurements is a statistical estimation procedure that has no training phase. Therefore, the prediction accuracy for conventional experiments is a single point Fig. 9.2(c). The training epoch on the horizontal axis in Fig. 9.2(c) is only for quantum-enhanced strategy.

Quantum-enhanced experiments

Quantum-enhanced experiments are executed by performing an entangling Bell measurement across two copies of ρ. We prepare the state ρ on the system qubits (marked blue in Supp. Fig. 9.4), swap the state to the memory qubits (marked red), prepare another state ρ on the system qubits, then perform an entangling Bell measurement across the two copies of ρ. Note that every preparation of ρ generates a random product state according to a classical probability distribution described in Section 9.4.

For each system size $n = 2, \ldots, 20$, we choose n qubits from among the 20 pairs of qubits shown in Supp. Fig. 9.4(b); these pairs are selected to minimize errors in the state preparations, gates, and measurements. Because the state ρ is not entangled, no entangling gates are used during the state preparation; therefore there is no advantage in choosing the pairs of qubits to be in proximity to one another. For each system size n, we use the same qubits for the conventional experiments as for the quantum-enhanced experiments, except that in conventional experiments we prepare the unknown state ρ only on the system qubits, and then perform a randomized Pauli measurement of the system immediately after the state preparation.

While a Bell measurement on a pair of qubits can be performed via a simple circuit containing one Hadamard gate, one CNOT gate, and two Z-basis measurements, we instead compile these operations into operations better suited for the Sycamore processor. In particular, the native two-qubit entangling gate is the Sycamore Gate, which has a unitary matrix representation given by

$$\mathrm{SYC} = \mathrm{iSWAP}^{\dagger}\ \mathrm{CPHASE}(-\pi/6) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -i & 0 \\ 0 & -i & 0 & 0 \\ 0 & 0 & 0 & e^{-i\pi/6} \end{pmatrix}. \tag{9.1}$$

Using this gate, the Bell state measurements may be performed by the gate sequence needed to unprepare a Bell state, which when compiled to Sycamore's native gates is expressed as a product of SYC gates and phased XZ gates (PhXZ_i). The phased XZ gate PhXZ_i on the i-th qubit is a native gate on the Sycamore device that can be expressed as

$$\mathrm{PhXZ}(a, x, z)_i = Z_i^z Z_i^a X_i^x Z_i^{-a}. \tag{9.2}$$

where X_i and Z_i are the standard Pauli operators acting on qubit i, and the exponents a, x, z are real numbers. The particular angles (a, x, z) for each of the gates used in our experiments were compiled numerically via a variational optimization. As the inverse Sycamore is not a native gate of the architecture, the compilations to hardware for inverse gates have to compensate for this difference, which we do numerically. The full decomposition of all the gates and circuits we reference are provided as Cirq circuits in additional supplemental material.

Each quantum-enhanced experiment generates a classical bitstring of size $2n$. We collect the bitstrings from all experiments and feed them into two data processing approaches. The first approach uses the specialized formula described in Section 9.5 to estimate $|\operatorname{tr}(Q_1\rho)|$ and $|\operatorname{tr}(Q_2\rho)|$. The second approach is a general approach that simply feeds the bitstrings into a neural network model. In addition to the experimental data, the neural network model also takes in two Pauli strings Q_1, Q_2 and predicts which one of $|\operatorname{tr}(Q_1\rho)|$ and $|\operatorname{tr}(Q_2\rho)|$ is larger. In both the specialized and the neural network approaches, we perform a basic form of measurement error mitigation described at the end of Section 9.4. In Fig. 9.2(a), we use the first approach (specialized formula) and repeat the quantum-enhanced experiments for a maximum of 500 times over different system sizes from $n = 2, 4, 6, 8, 10, 12, 14, 16, 18, 20$. In Fig. 9.2(c), we show the training process of the second approach (neural network model) and repeat each quantum-enhanced experiments 500 times. This provides a fair comparison with conventional experiments because two copies of the unknown state ρ are used in each quantum-enhanced experiment; therefore a total of 1000 copies are consumed in both our conventional and quantum-enhanced experiments.

A supervised neural network model using data from quantum-enhanced experiments

We train a supervised neural network model using noiseless simulation data from small system sizes. Then we use the trained neural network model on the noisy experimental data obtained from performing quantum-enhanced experiments. The neural network model has three layers. Each of the outer layers runs the preceding inner layer multiple times. In the following, we describe each layer of the neural network model.

1. The *inner layer* is a recurrent neural network based on gated recurrent unit (GRU) (J. Chung et al., 2014; D. Tang, Qin, and T. Liu, 2015; Goodfellow, Bengio, and Courville, 2016). The recurrent neural network takes in a size-$2n$ bitstring, corresponding to the measurement outcome from a single quantum-enhanced experiment, and an n-qubit Pauli operator Q, which can be represented as a size-$2n$ bitstring. The recurrent neural network outputs a two-dimensional real vector. Other popular choices of recurrent units, such as long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997) or transformer (Vaswani et al., 2017), could be used instead of GRU.

2. The *intermediate layer* is an aggregation layer. This layer runs the inner layer for all the bitstrings obtained from each of the quantum-enhanced experiments. For example, if we run the quantum-enhanced experiments for 100 times, we would obtain 100 size-$2n$ bitstring and we would run the inner layer for 100

times over each bitstring. The intermediate layer outputs the average of the two-dimensional output vectors from the multiple runs of the inner layers.

3. The *outer layer* is inspired by a Siamese neural network (twin neural network) (Koch, Zemel, Salakhutdinov, et al., 2015). This layer runs the intermediate layer twice, one for each of the two Pauli operators Q_1 and Q_2. Each intermediate layer generates a real-valued vector x of dimension two, which we map to a single real value by considering $x_1 - x_2$. The outer layer compares the real value from the two intermediate layers and outputs Q_1 or Q_2 based on which one of them has a higher real value.

The specific details of the above neural network structure is given in the accompanying code repository at Ref. (Huang, Richard Kueng, Torlai, et al., 2021).

In the inner layer, we create a recurrent neural network with an encoding layer that maps an integer between 0 and 15 to a vector of dimension 30, a GRU with 30 neurons, and a decoding layer that maps 30 neurons to 2 neurons. Only the inner layer contains trainable parameters. The intermediate layer and the outer layer are both fixed operations based on outputs from the inner layer, which will not be updated.

Next, we discuss the process for training the neural network model. We use noiseless simulation data (for small system sizes $n < 8$) to train the recurrent neural network. During training, we pick a state $\rho = 2^{-n} (I + \alpha P)$ where $\alpha \in \{-0.95, 0.95\}$ and P is an n-qubit Pauli operator, and pick an n-qubit Pauli operator Q that is equal to P with probability $1/2$ and is not equal to P with probability $1/2$. We encode the training data into two tensors, `inp` and `target`. The encoding is defined by the following.

- The tensor `inp` is of size $b \times n$, where b is the number of quantum-enhanced experiments we performed, n is the number of qubits, and each entry of `inp` is an integer from 0 to 15. The (t, i)-th entry of `inp` encodes the component of Q on qubit i (a choice of 4 for I, X, Y, Z) and the Bell measurement outcome on qubit i from the t-th quantum-enhanced experiment (also a choice of 4). Each entry takes a total of 16 possible values.

- The tensor `target` is of size 1. The entry in `target` is equal to 1 if $P = Q$, and is equal to 0 if $P \neq Q$.

We update the neural network model once using `inp` of size $b \times n$ and `target` of size 1. We are using the cross entropy loss and employ the Adam optimizer (Kingma and Ba, 2014), which is a gradient-based optimization algorithm that adaptively estimates lower-order moments. We generate multiple different states ρ and Q corresponding to different `inp` and `target` to train the neural network model.

During the training process, we are not using the *outer layer*. Also, we simultaneously run the b repetitions of the inner layer for each outcome from a single quantum-enhanced experiment by leveraging parallel computing. Then, we average over the b repetitions of the inner layer. Also, the output of the neural network model is a two-dimensional real vector, denoted as $v = (v_0, v_1)$. When `target` is $a \in \{0, 1\}$, the loss function is given by

$$-\log\left(\frac{\mathrm{e}^{v_a}}{\mathrm{e}^{v_0} + \mathrm{e}^{v_1}}\right). \tag{9.3}$$

The two real values v_0, v_1 are combined to produce a probability distribution

$$\frac{\mathrm{e}^{v_0}}{\mathrm{e}^{v_0} + \mathrm{e}^{v_1}} = 1 - \frac{1}{\mathrm{e}^{v_0 - v_1} + 1}, \quad \frac{\mathrm{e}^{v_1}}{\mathrm{e}^{v_0} + \mathrm{e}^{v_1}} = \frac{1}{\mathrm{e}^{v_0 - v_1} + 1}, \tag{9.4}$$

indicating which of $a = 0$ and $a = 1$ is more likely. If $v_0 - v_1$ is large, then $a = 0$ corresponding to $P \neq Q$ is more likely. On the other hand, if $v_0 - v_1$ is small, then $a = 1$ corresponding to $P = Q$ is more likely. We compute the gradient through back-propagation and update the model using the Adam optimizer (Kingma and Ba, 2014).

Finally, we discuss the prediction process in the neural network model. Due to the significant amount of measurement errors, we employ a form of measurement error mitigation. We first characterize the measurement errors for every qubit assuming the zero state preparations and X-gates are perfect. For each qubit i, we obtain a 2×2 matrix specifying the probability to measure 0 or 1 if the qubit is in $|0\rangle\langle 0|$ or $|1\rangle\langle 1|$. We store that as a list of 2×2 matrices called `calib_2x2`. We then expand the data, referred to as `data` in the pseudo-code, obtained from the quantum-enhanced experiments, which is a two-dimensional array of size $b \times (2n)$. Basically, we expand each measurement to 20 measurements with a real-valued coefficient associated to each of the expanded measurements. Therefore, `data_expanded` is a two-dimensional array of size $(20b) \times (2n)$ and `coefficients` is a one-dimensional array of size $20b$.

```
def noise_inversion(data, calib_2x2, inverse_cnt=20):
    Set data_expanded as an empty array
    Set list_of_coefficients as an empty array

    for t from 0 to b-1:
        for r from 0 to inverse_cnt-1:
            Set single_data as an empty array
            Set coefficient as 1.0

            for i from 0 to 2n-1:
                Set p as calib_2x2[i][1, 1] if data[t][i] = 0
                Set p as calib_2x2[i][0, 0] if data[t][i] = 1

                With probability 1-p do:
                    single_data.append(1-data[t][i])
                    coefficient *= -1
                Else do:
                    single_data.append(data[t][i])

            Append single_data to data_expanded
            Append coefficient to list_of_coefficients

    return data_expanded, list_of_coefficients
```

After obtaining `data_expanded`, we construct two tensors `inp1` and `inp2` corresponding to the same experimental data `data_expanded`, but different Pauli operators Q_1, Q_2. Both `inp1` and `inp2` are tensors of size $(20b) \times n$, where b is the number of quantum-enhanced experiments we performed, n is the number of qubits, and each entry of `inp1` and `inp2` is an integer from 0 to 15 similar to the training process. Then the neural network make a prediction using the two input tensors `inp1` and `inp2`. In the *outer layer*, the neural network model runs the *intermediate layer* (as well as the multiple repetitions of *inner layer*) for each of the two input tensors to obtain two 2D vectors denoted as $u = (u_0, u_1), v = (v_0, v_1)$. From the discussion given around Eq. (9.4). If $u_0 - u_1$ is small, then it is more likely that $P = Q_1$. If $v_0 - v_1$ is small, then it is more likely that $P = Q_2$. The neural network hence compare $u_0 - u_1$ and $v_0 - v_1$ to predict whether $P = Q_1$ or $P = Q_2$.

Experiments on learning physical dynamics

For the task of learning about physical dynamics in 1D and 2D, we considered unitary transformations implemented by 1D and 2D random quantum circuits. We generated many random circuits, half of which are time-reversal symmetric (i.e., real orthogonal), and half of which are general unitary circuits without any symmetry. For each of these circuits, we performed both conventional and quantum-enhanced experiments to generate classical measurement data. This data was fed to an unsupervised machine learning model to learn a low-dimensional classical representation of the physical dynamics. We wished to see whether the unsupervised ML model could recognize the difference between time-reversal symmetric dynamics and general dynamics. The results summarized in Fig. 9.3 were obtained in experiments analyzing 180 different circuits in each of the the two classes, using methods described below. The largest quantum circuits we ran on the Sycamore processor are presented in Table 9.1.

	Number of qubits	**Number of gates**	**Circuit depth**
1D dynamics	**40**	**842**	**40**
2D dynamics	**40**	**1388**	**54**

Table 9.1: Circuit information for the experiments on learning physical dynamics.

In (Aharonov, J. S. Cotler, and Qi, 2021), a restricted subclass of conventional strategies was shown to require an exponential number of experiments to distinguish between general unitary dynamics and time-reversal-symmetric dynamics. In (Sitan Chen, J. Cotler, et al., 2021b), some of the authors of the present work have shown that an exponential number of experiments are required for this task even when arbitrary conventional strategies are allowed. Furthermore, it is plausible that under appropriate cryptographic assumptions, the superpolynomial difficulty of characterizing quantum dynamics in conventional experiments would persist even for psuedo-random dynamical processes that can be efficiently generated on a quantum computer. However, at present, explicit constructions of cryptographically-secure pseudo-random unitaries are not known. In light of this, in our experiments we resort to studying random quantum circuits similar to those used for demonstrating quantum computational supremacy (Arute et al., 2019).

In this subsection, we provide further details regarding how our samples of 1D dynamics and 2D dynamics are generated, how our experiments are conducted, and how our unsupervised machine learning model works. As we will see, the unsupervised ML successfully learns to classify quantum circuits into symmetry classes when provided with data from quantum-enhanced experiments, but not when provided with data from conventional experiments.

1D dynamics

We use the layout provided in Supp. Fig. 9.4(b). A 1D circuit is implemented along the 1D line connecting all the qubits circled in blue. For system size $n < 20$, we consider a contiguous region on the 1D line with the smallest gate/measurement error. For general unitary dynamics, we use a 1D version of the quantum supremacy circuit (Arute et al., 2019). The quantum supremacy circuit in 1D interleaves between a layer of random single-qubit gates and a layer of two-qubit entangling gates, namely SYC gates applied to neighboring qubits. We alternate the partitioning of the two-qubit entangling gates, e.g., $(1, 2), (3, 4), (5, 6) \longleftrightarrow (2, 3), (4, 5)$ for $n = 6$.

For T-symmetric (time-reversal-symmetric) dynamics, the single-qubit gates are real orthogonal 2×2 matrices of the form e^{-itY}, where Y is the Pauli-Y matrix and t is a randomly chosen real number. In addition, we replace the two-qubit entangling gate SYC by a T-symmetric two-qubit entangling gate

$$V = (U_3 \otimes U_4)\mathrm{SYC}(U_1 \otimes U_2), \tag{9.5}$$

where U_1, U_2, U_3, U_4 are appropriately chosen single-qubit gates. In order to find a suitable choice of $U_1, \ldots, U_4$ such that the two-qubit gate V is time-reversal symmetric, we employ a numerical optimization. We parameterize each of U_i as $\exp(\mathrm{i}(a_i X + b_i Y + c_i Z))$, where $a_i, b_i, c_i \in \mathbb{R}$ are initialized randomly. There are a total of 12 variables. Next, we define the loss function to be equal to the Frobenius norm of the imaginary part of the matrix V (after fixing the top left entry of V to be real). We then perform gradient descent to minimize the loss function and terminate once we have found that the loss function is below 10^{-9}. We then replace SYC by V.

In the conventional experiment, we begin with $|0^n\rangle\langle 0^n|$ on the system qubits, evolve under the 1D dynamics, and measure in the Y-basis. We also considered using randomized Pauli measurement at the end, but the performance for measuring in the Y-basis is slightly better. The rationale is that the output state under T-symmetric evolution has purely real amplitudes; hence the expectation value of any purely imaginary observable, such as the Pauli-Y operator, is always zero. In

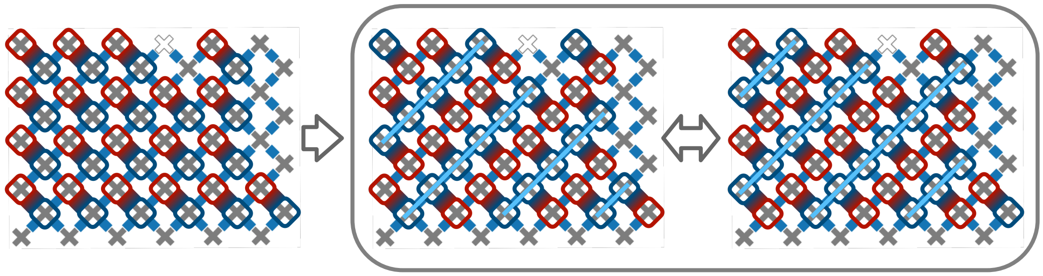


Figure 9.5: To implement the 2D dynamics, we first move from the layout on the left to the layout in the middle (by swapping some pairs of system and memory qubits). Then, we iterate between the layout in the middle and the layout in the right (by swapping all pairs of system and memory qubits).

contrast, the expectation value of Y after a general unitary evolution is non-zero. T-symmetric unitaries are nevertheless hard to distinguish from general unitaries in the conventional setting because the expectation value of Y is exponentially small for general unitaries; therefore an exponentially large number of experiments are needed to discern its nonzero value. Indeed, the result in (Aharonov, J. S. Cotler, and Qi, 2021; Sitan Chen, J. Cotler, et al., 2021b) shows that conventional strategies require an exponential number of experiments for distinguishing T-symmetric unitaries from general unitaries.

In the quantum-enhanced experiment, we prepare a Bell state $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ for every pair of system and memory qubits. Then we evolve the system qubits under the unknown dynamics. After the evolution, we swap the system and the memory qubits. Then we evolve the system qubits under the unknown dynamics again. Finally, we measure every pair of system and memory qubits in the Bell basis. Each quantum-enhanced experiment generates a $2n$-bit string. We perform gradient descent to find our implementation of the Bell state preparation, swap operation, and Bell measurement using the native gates in the Sycamore processor.

2D dynamics

For our 2D circuits we use the layout provided in Supp. Fig. 9.4(c) which is also shown as the leftmost layout in Supp. Fig. 9.5. In the leftmost layout, none of the system qubits (circled blue) are connected to one another. In order to implement 2D dynamics, we first swap some pairs of the system and memory qubits to obtain the layout shown in the middle. In the middle layout, we can see that many of the system qubits are connected (the light blue line). We implement a depth-4 1D random quantum circuit for each light blue line. Each depth-4 circuit corresponds to 1 layer of single-qubit gates, 1 layer of two-qubit gates, 1 layer of single-qubit gates, and 1 layer of two-qubit gates. The partitioning of the two layers of two-qubit gates are different. Then, we swap all pairs of qubits to obtain the layout shown on the right. The right-most layout connects a different set of system qubits (the light blue line). We again implement a depth-4 1D random quantum circuit for each light blue line. After that, we move back to the middle layout and repeat for multiple rounds. After sufficiently many repetitions, the n qubits become globally entangled.

An unsupervised machine learning model

For each circuit, we create a feature vector by obtaining statistics for each bit in the measurement outcome bitstring. In conventional experiments, each experiment produces an n-bit measurement outcome. In quantum-enhanced experiments, each experiment produces a $2n$-bit measurement outcome. In the following, we consider $\ell = n$ or $2n$ depending on whether we are running conventional or quantum-enhanced experiments. For an ℓ-bit measurement outcome, we obtain a feature vector of size 2ℓ including the first and second moment of each bit. After constructing a feature vector for each circuit, we map the feature vector to an infinite-dimensional reproducing kernel Hilbert space (corresponding to a Gaussian kernel) that includes all the polynomial expansions of the feature vector. Then, we find a low-dimensional subspace in the infinite-dimensional Hilbert space using principal component analysis (PCA) (Schölkopf, A. Smola, and Müller, 1998). The entire procedure can be performed efficiently using kernel PCA (Schölkopf, A. Smola, and Müller, 1998). Kernel PCA is implemented using scikit-learn (Buitinck et al., 2013).

In Fig. 9.3 of the main text, we show a one-dimensional subspace found by the unsupervised ML model, for both 1D and 2D random quantum circuits. We can use this one-dimensional representation to classify the circuits into two classes (by splitting the one-dimensional representation in the middle). Then, we can evaluate the accuracy of the unsupervised ML model by checking the percentage of circuits that are correctly classified as general circuits or as T-symmetric circuits. A two-dimensional subspace found by the unsupervised ML model, and an assessment of classification accuracy, are discussed in Section 9.4

Additional experimental results

In Supp. Fig. 9.6, we provide the one-dimensional representations using the best

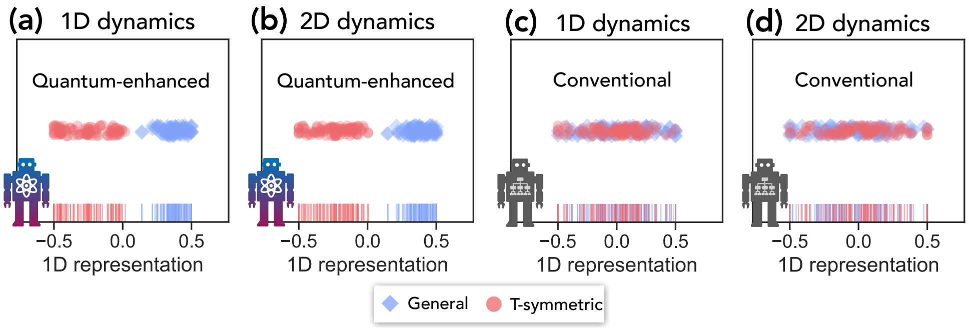


Figure 9.6: *The best known human-designed one-dimensional representation for 1D and 2D dynamics.* Each point in the one-dimensional space corresponds to a distinct physical process. Half of the processes have time-reversal symmetry (blue diamonds) while the other half do not (red circles).

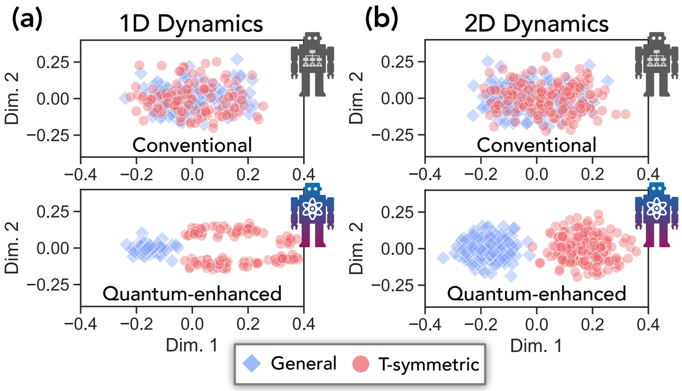


Figure 9.7: *Two-dimensional representation learned by unsupervised ML for (a) 1D dynamics and (b) 2D dynamics.* Each point in the two-dimensional plane corresponds to a distinct physical process. Half of the processes have time-reversal symmetry (blue diamonds) while the other half do not (red circles). When fed with data from quantum-enhanced experiments, the ML model accurately discovers the underlying symmetry pattern. In contrast, the ML model fails to do so when fed with data from conventional experiments.

known specialized data-processing approach for the various random quantum circuits investigated in our conventional and quantum-enhanced experiments. We design the 1D representation based on the following facts. In noiseless quantum enhanced experiments, the measurement outcome should be 0 for all qubits when the evolution satisfies T-symmetry, and the measurement outcome should be a random 0 or 1 when the evolution is a general unitary dynamics. In noiseless conventional experiments, the expectation value of single-qubit Y observable should be zero for all qubits when the evolution satisfies T-symmetry, and the expectation value of single-qubit Y observable should be nonzero when the evolution is a general unitary dynamics. Using these facts, we consider the representation for the quantum-enhanced experiments to be the fraction of bits that are measured to be 0. On the other hand, the representation for the conventional experiments is the absolute value of the expectation value of single-qubit Y observable averaged over all qubits. In both cases, we linearly scale the 1D representation to be between -0.5 and 0.5. We can see that the highly-specialized representations are similar to the one-dimensional representations learned by unsupervised ML (see Fig. 9.3 in the main text).

In Supp. Fig. 9.7, we provide the two-dimensional representations learned by unsupervised ML for the various random quantum circuits investigated in our conventional and quantum-enhanced experiments. (One-dimensional representations are presented in Fig. 9.3 in the main text.) We see that in the second dimension found by unsupervised ML using quantum-enhanced experiments for 1D dynamics, T-symmetric dynamics are clustered into two groups. Further inspection shows that the unsupervised ML model has learned substructure corresponding to the parity of the depth of the evolution (recall that the depth is always an integer). In principle, the unsupervised ML model should be able to learn a wide variety of structures in the dynamics. Notably, we see that it places the distinction between general unitary dynamics and T-symmetric dynamics as the major axis (the first dimension), and places less prominent structure as the second major axis (the second dimension).

In Supp. Fig. 9.8, we provide the accuracy of the unsupervised ML model for distinguishing between general unitary dynamics and T-symmetric dynamics. We see a substantial advantage for using the quantum-enhanced strategy in both the physical experiments and the noiseless simulation. We perform brute-force noiseless simulation for conventional experiments because the system size is at most 20. The noiseless simulation for quantum-enhanced experiments uses the fact that $(U \otimes U)\frac{1}{\sqrt{2^n}}\sum_{i=0}^{2^n-1}|ii\rangle = (UU^T \otimes I)\frac{1}{\sqrt{2^n}}\sum_{i=0}^{2^n-1}|ii\rangle$, hence we can effectively reduce the simulation to a system size at most 20.

Performance and characterization data

The performance of the device was characterized before each run. The measurement data is collected explicitly and used for measurement error mitigation in the prediction process of the supervised neural network model (see discussion in the last part of Section 9.4). The task of learning quantum states is largely limited by the

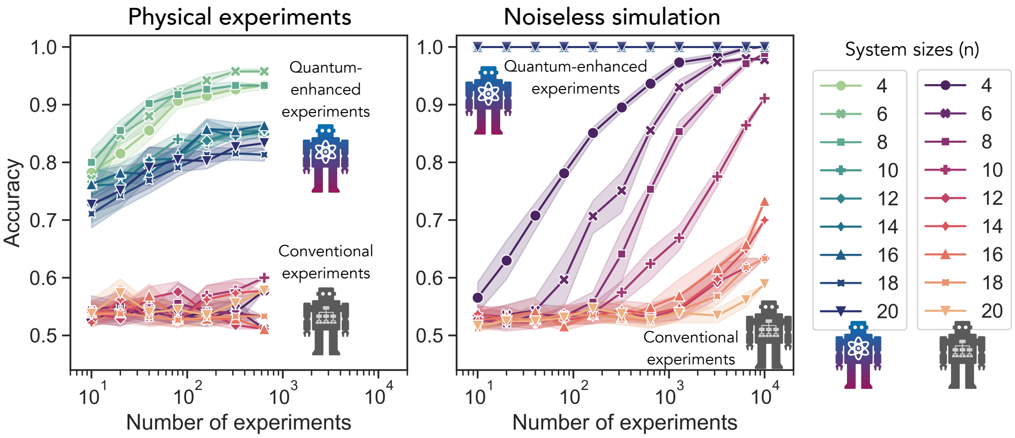


Figure 9.8: *Accuracy of the unsupervised ML model for classifying general unitary and T-symmetric dynamics.* For each system size, we generate 100 different circuits for each of the two classes (general and T-symmetric). The one-dimensional representation found by the unsupervised ML model is used to classify the 200 circuits into two classes. We consider both physical experiments and noiseless simulations. Accuracy is plotted as a function of the number of experiments in both the conventional and quantum-enhanced settings. In the noiseless simulation, quantum-enhanced experiments has an accuracy of 1.0 for all system sizes and all numbers of experiments we considered.

qubit measurement fidelities. A representative sample of the data from the device is reported in Fig. 9.9, where one can see that typical readout errors (conflated with errors in preparing zero and one states) range from 3% to 7%. For transmons, the $|0\rangle$ preparation has a small error; hence Supp. Fig. 9.9(a) is dominated by the readout error. Furthermore, the single-qubit gate error (shown in Supp. Fig. 9.10) is much smaller than the error shown in Fig. 9.9(b), hence the error shown in Supp. Fig. 9.9(b) is mostly due to readout errors rather than gate errors. During the actual run of the experiments, we avoid using qubits with the worst readout errors by checking the measurement errors before the experiment and selecting the layout accordingly.

The task of learning quantum dynamics involves circuits of higher complexity and hence is limited by both measurement errors and errors in two-qubit gates. For these experiments, we report in Supp. Fig. 9.10 the quality of the single-qubit and two-qubit gates across the device. This data was obtained via parallel cross-entropy benchmarking and single-qubit randomized benchmarking. The typical single-qubit gate error is around 0.001 to 0.005, while the typical two-qubit gate error is around 0.01 to 0.05.

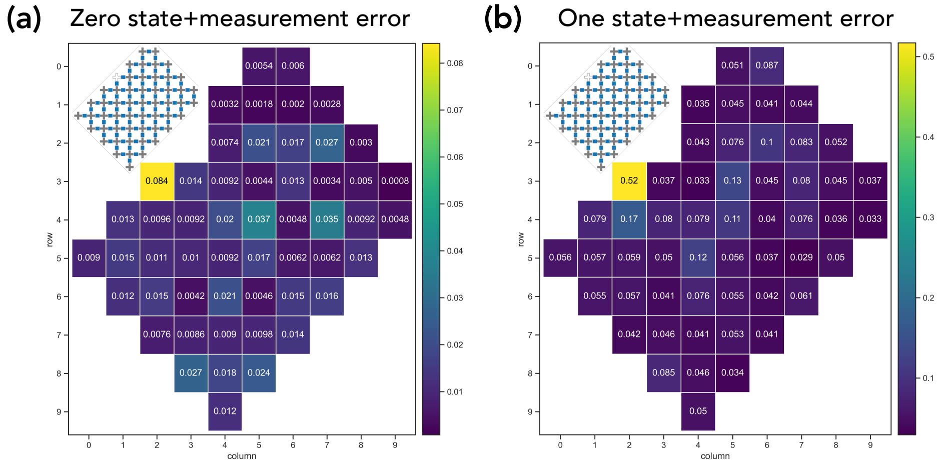


Figure 9.9: *Sycamore state preparation and measurement error data.*
(a) $|0\rangle$ *state preparation and measurement error.* We prepare a noisy zero state $|0\rangle$ and measure in the computational basis using the noisy single-qubit readout. We show the probability of measuring $|1\rangle$ in the qubit readout.
(b) $|1\rangle$ *state preparation and measurement error.* We prepare a noisy one state $|1\rangle$ and measure in the computational basis using the noisy single-qubit readout. We show the probability of measuring $|0\rangle$ in the qubit readout.
While these values change over time, we present here a representative sample of the error. One can see that in accordance with physical expectations based on T_1 errors, the readout in the physical 1 state is substantially higher than the 0 state.

9.5 Quantum advantage in predicting highly-incompatible observables

The first task we study using the framework of the previous section involves learning about a physical system represented by an n-qubit state ρ. We provide an illustration of the task in Supp. Fig. 9.11.

- In conventional experiments, we consider algorithms that can measure each copy of ρ one at a time. The algorithm can choose to perform any POVM measurement on each copy, where the POVM measurement can be chosen adaptively based on the outcomes of previous experiments.

- In quantum-enhanced experiments, we consider algorithms that can use a quantum computer to act collectively on multiple copies of ρ to obtain entangled measurement data.

In both scenarios, we consider all quantum data to be used during the learning phase, and we are left only with classical measurement data. After this learning phase, the learner is then asked to provide accurate predictions for the expectation value

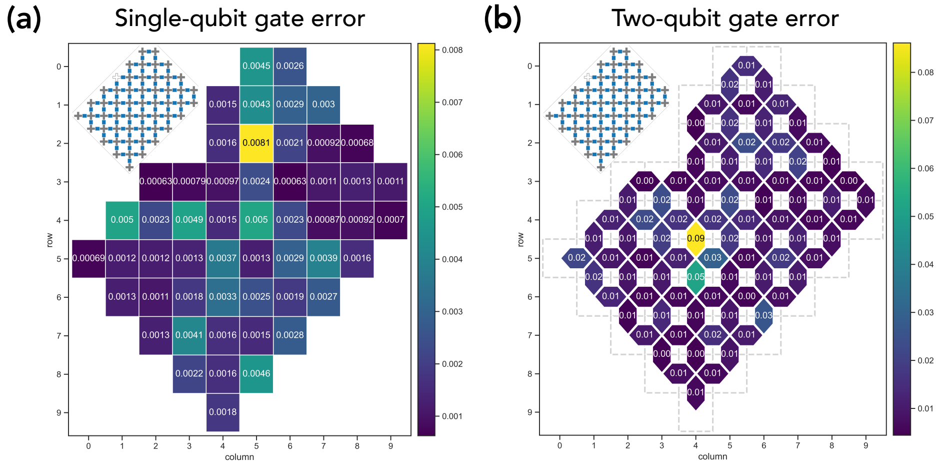


Figure 9.10: *Sycamore single- and two-qubit gate error data.*
(a) Single-qubit gate error. The figure shows the error of single-qubit gates across the chip using parallel single-qubit randomized benchmarking.
(b) Two-qubit gate error. The figure shows the error across the chip of two-qubit gates being executed in parallel, as to account for errors that occur during simultaneous operation of qubits. We can see that the distribution of errors varies across the chip couplers, showing the extent to which performance is non-uniform.

of an observable O, using the classical data obtained from the experiments. The observable O is selected from an exponentially large set $\{O_1, O_2, \dots O_M\}$, where $O_1, \dots, O_M$ may not be mutually commuting and M is exponential in n.

Note that when the observables in the set are not mutually commuting, it is impossible to measure all of them simultaneously. Hence, a naïve algorithm in the conventional scenario would be to measure the exponential number of observables individually, which would result in exponential sample complexity.

Exponential advantage in predicting absolute value of a single observable

We will prove that even predicting the absolute value of just a single observable requires exponentially many copies in the conventional scenario. In contrast, an algorithm with quantum memory can predict the expectation values for M arbitrary observables from only $O(n\log(M)/\epsilon^4)$ copies of ρ through the procedure known as shadow tomography (Aaronson, 2019; Aaronson and Rothblum, 2019; Bădescu and O'Donnell, 2020). Hence, even if we would like to predict an exponential number of observables, an algorithm with quantum memory only needs a polynomial number of copies.

In fact, for certain natural instances, we can show an even more dramatic separation.

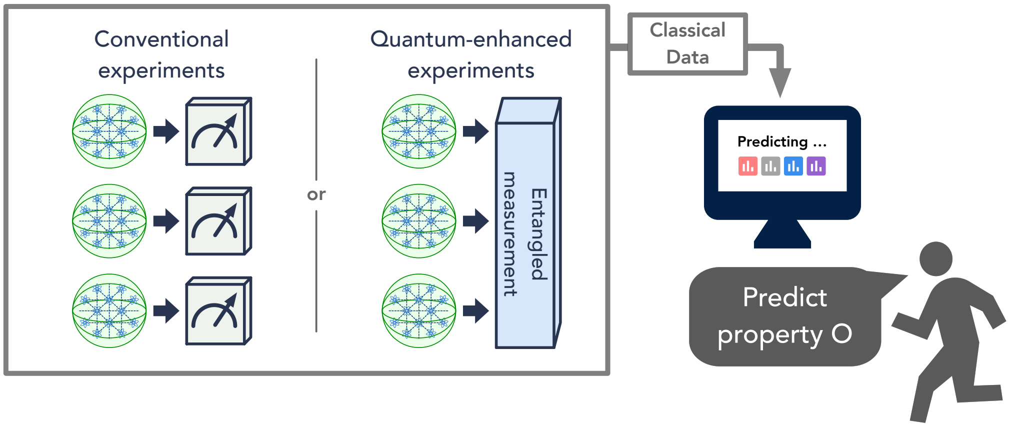


Figure 9.11: *Illustration for the task of predicting highly-incompatible observables.* The unknown quantum state ρ is represented by the green sphere. Conventional experiments measure each copy of state ρ individually, and the measurements can depend adaptively on previous measurements. Quantum-enhanced experiments store many copies of ρ in a quantum memory, process the copies with a quantum computer, and produce an entangled measurement outcome. The classical data obtained from the experiments are used to predict a property of ρ.

Specifically, for the following states and observables, we will show how to achieve an exponential versus *constant* separation.

Definition 14 (Separation instance)**.** *Consider a distribution $\mathcal{D}$ over n-qubit state ρ and observable O.*

1. *With probability $1/2$, the state is $\rho = I/2^n$ and $O \in \{I, X, Y, Z\}^{\otimes n} \setminus \{I^{\otimes n}\}$ is chosen uniformly at random.*
2. *With probability $1/2$, the state is $\rho = (I + 0.9sP)/2^n$ and $O = P$, where $s = \{\pm 1\}$ with equal probability and $P \in \{I, X, Y, Z\}^{\otimes n} \setminus \{I^{\otimes n}\}$ is chosen uniformly at random.*

Note that while 0.9 is used in the definition $\rho = (I + 0.9sP)/2^n$, any constant value smaller than 1 is sufficient to obtain the exponential separation. A technical difficulty arises when we consider $(I + sP)/2^n$, and it is unclear whether this difficulty is fundamental. Interestingly, the n-qubit state ρ considered in the above definition does not contain any quantum entanglement. The state ρ can be written as a classical probability distribution over tensor products of single-qubit states. Despite the lack of quantum entanglement, we can still achieve an exponential versus constant separation. This result is a substantial improvement over the result established in (Huang, Richard Kueng, and Preskill, 2021). In (Huang, Richard Kueng, and Preskill, 2021), some of the authors showed an $\Omega(2^{n/3})$ versus $O(n)$ separation between conventional and quantum-enhanced strategies, but the task was to predict an exponential number of observables, which can only be verified using an exponential amount of time.

Theorem 47 (Exponential advantage in predicting highly-incompatible observables)**.** *We sample an n-qubit state* ρ *and an observable* O *according to* $\mathcal{D}$ *given in Definition 14; both of these are unknown to the algorithm. The algorithm then learns about* ρ *through conventional or quantum-enhanced experiments. After the learning phase, we ask the learning algorithm to predict* $|\mathrm{tr}(O\rho)|$.

- Upper bound: *There is an algorithm in the* ***quantum-enhanced*** *scenario using only* $O(1)$ *copies of* ρ *to predict up to* 0.25 *additive error with probability at least* 0.8.
- Lower bound: *For any algorithm in the* ***conventional*** *scenario, it needs at least* $\Omega(2^n)$ *copies of* ρ *to predict up to* 0.25 *additive error with probability at least* 0.8.

Here, we are using the standard Big-O and Big-Ω notations: $f = \Omega(g)$ if there is an $n_0, C > 0$ such that $\forall n > n_0$, $f(n) \geq Cg(n)$; and $f = O(g)$ if there is an $n_0, M > 0$ such that $\forall n > n_0$, $|f(n)| \leq Mg(n)$. We separate the proof of Theorem 47 into the following two subsections. In Section 9.5, we prove a constant upper bound for quantum-enhanced experiments for this task. In Section 9.5, we prove an exponential lower bound for conventional experiments for the same task.

A constant upper bound for quantum-enhanced experiments

The learning algorithm in the quantum-enhanced scenario builds on results presented in (Huang, Richard Kueng, and Preskill, 2021). We separate the protocol into the learning phase, where entangled measurements are performed, and the prediction phase, where we predict the desired properties.

Learning phase

Consider N_{Q} rounds of two-copy entangled measurements. In round

$$t \in \{1, \ldots, N_{\mathrm{Q}}\}, \tag{9.6}$$

for every $k \in \{1, \ldots, n\}$ we measure the k-th qubit from the first and second copies of ρ in the Bell basis to obtain

$$S_k^{(t)} \in \left\{ |\Psi^+\rangle\langle\Psi^+|, |\Psi^-\rangle\langle\Psi^-|, |\Phi^+\rangle\langle\Phi^+|, |\Phi^-\rangle\langle\Phi^-| \right\}, \tag{9.7}$$

where the Bell basis encompasses four maximally entangled two-qubit states. Here, $|\Omega\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ is the Bell state, and we additionally have

$$|\Psi^+\rangle = I \otimes I\, |\Omega\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle), \quad |\Psi^-\rangle = I \otimes Z\, |\Omega\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle),$$
$$|\Phi^+\rangle = I \otimes X\, |\Omega\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle), \quad |\Phi^-\rangle = \mathrm{i} I \otimes Y\, |\Omega\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle).$$

Then, we efficiently store the measurement data $S_k^{(t)}, \forall k = 1, \ldots, n, \forall t = 1, \ldots, N_{\mathrm{Q}}$ in a classical memory with $2nN_{\mathrm{Q}}$ classical bits.

Prediction phase

Given an observable O drawn from $\{I, X, Y, Z\}^{\otimes n} \setminus \{I^{\otimes n}\}$, we can use the block of classical memory obtained in the learning phase to estimate $|\operatorname{tr}(O\rho)|$. First let us consider the case where ρ is a single-qubit state. When we measure $\rho \otimes \rho$ in the Bell basis, the measurement outcome S is a projector onto one of the four Bell states given in Eq. (9.7). Let $\sigma \in \{I, X, Y, Z\}$ be any Pauli matrix. Each Bell state is an eigenstate of $\sigma \otimes \sigma$ with an eigenvalue ± 1. The probability that the Bell measurement outcome S is an eigenstate of $\sigma \otimes \sigma$ with eigenvalue $+1$ is

$$\mathrm{Prob}(+) = \frac{1}{2} \operatorname{tr}\left((I \otimes I + \sigma \otimes \sigma)(\rho \otimes \rho)\right), \tag{9.8}$$

while the -1 eigenvalue occurs with probability

$$\mathrm{Prob}(-) = \frac{1}{2} \operatorname{tr}\left((I \otimes I - \sigma \otimes \sigma)(\rho \otimes \rho)\right). \tag{9.9}$$

Therefore, we have

$$\mathbb{E}\left[\operatorname{tr}\left((\sigma \otimes \sigma) S\right)\right] = \mathrm{Prob}(+) - \mathrm{Prob}(-) = \operatorname{tr}\left((\sigma \otimes \sigma)(\rho \otimes \rho)\right) = |\operatorname{tr}(\sigma\rho)|^2, \tag{9.10}$$

where $\mathbb{E}$ denotes the expectation with respect to the probability distribution over Bell measurement outcomes. We see that the entangling Bell measurement enables us to estimate the absolute value $|\operatorname{tr}(\sigma\rho)|$ for any Pauli matrix $\sigma \in \{I, X, Y, Z\}$.

We can generalize this observation to the case where ρ is an n-qubit state, and each pair of qubits in $\rho \otimes \rho$ is measured in the Bell basis to yield the outcomes

$\{S_k, k = 1, 2, \ldots, n\}$. If $O = \sigma_1 \otimes \cdots \otimes \sigma_n$ is a Pauli observable, then as in the $n = 1$ case the Bell state S_k is an eigenstate of $\sigma_k \otimes \sigma_k$ with eigenvalue ±1 for each k. This implies that $\bigotimes_{k=1}^n S_k$ is an eigenstate of $O \otimes O$ with an eigenvalue ±1. In particular, let us consider the product

$$\prod_{k=1}^{n} \mathrm{tr}\left((\sigma_k \otimes \sigma_k) S_k\right) = \pm 1. \tag{9.11}$$

This product is equal to +1 when the tensor product of the Bell measurement outcomes $\bigotimes_{k=1}^n S_k$ is an eigenstate of $O \otimes O$ with eigenvalue +1, and it is −1 when $\otimes_{k=1}^n S_k$ is an eigenstate of $O \otimes O$ with eigenvalue −1. We conclude that

$$\begin{aligned}\mathbb{E}\left[\prod_{k=1}^{n} \mathrm{tr}\left((\sigma_k \otimes \sigma_k) S_k\right)\right] &= \mathbb{E}\left[\mathrm{tr}\left((O \otimes O) \bigotimes_{k=1}^{n} S_k\right)\right] \\ &= \mathrm{Prob}(O \otimes O = +1) - \mathrm{Prob}(O \otimes O = -1) \\ &= \mathrm{tr}\left((O \otimes O)(\rho \otimes \rho)\right) \\ &= |\,\mathrm{tr}(O\rho)|^2, \end{aligned} \tag{9.12}$$

where $\mathbb{E}$ denotes the expectation with respect to the probability distribution of Bell measurement outcomes. The above derivation shows that the n-qubit entangling Bell measurement enables us to estimate the absolute value $|\,\mathrm{tr}(O\rho)|$ for any O considered in Definition 14.

Because Equation (9.12) relates the probability distribution of Bell measurement outcomes to the absolute value $|\,\mathrm{tr}(O\rho)|$, we can estimate $|\,\mathrm{tr}(O\rho)|$ accurately by repeatedly making entangling Bell measurements on successive pairs of copies of ρ sufficiently many times. Specifically, in the learning phase, we perform the entangling Bell measurement on N_{Q} pairs of copies of ρ, and collect the measurement data $\{S_k^{(t)}\}$ in the classical memory, where $k = 1, 2, \ldots, n$ labels the qubit pairs, and $t = 1, 2, \ldots, N_{\mathrm{Q}}$ labels the different rounds of measurements. For any given n-qubit Pauli observable $O = \sigma_1 \otimes \cdots \otimes \sigma_n$, we consider the following estimator

$$\hat{a}(O) = \frac{1}{N_Q} \sum_{t=1}^{N_Q} \prod_{k=1}^{n} \mathrm{tr}\left((\sigma_k \otimes \sigma_k) S_k^{(t)}\right), \tag{9.13}$$

which can be computed efficiently in time $O(nN_{\mathrm{Q}})$.

Using the expectation evaluated in Equation (9.12), we can apply Hoeffding's inequality to show that the estimate $\hat{a}(O)$ is equal to the expectation value $\mathrm{tr}((O \otimes O)(\rho \otimes \rho)) = |\,\mathrm{tr}(O\rho)|^2$ up to a small error with high probability. The formal statement is given below.

Lemma 50. *Given $N_\mathrm{Q} = \Theta(\log(1/\delta)/\epsilon^2)$. For any observable O considered in Definition 14, we have*

$$\left|\hat{a}(O) - |\operatorname{tr}(O\rho)|^2\right| \leq \epsilon, \tag{9.14}$$

with probability at least $1 - \delta$.

To obtain an estimate for the absolute value $|\operatorname{tr}(O\rho)|$, we consider the estimate

$$\hat{b} = \sqrt{\max(0, \hat{a})}. \tag{9.15}$$

We can show the inequalities

$$|\operatorname{tr}(O\rho)|^2 - \epsilon \leq \hat{a} \leq |\operatorname{tr}(O\rho)|^2 + \epsilon \tag{9.16}$$

$$\implies \max(0, \sqrt{|\operatorname{tr}(O\rho)|^2} - \sqrt{\epsilon}) \leq \hat{b} \leq \sqrt{|\operatorname{tr}(O\rho)|^2} + \sqrt{\epsilon}, \tag{9.17}$$

using the fact that $\sqrt{x+y} \leq \sqrt{x} + \sqrt{y}$.

In the final step of the upper bound proof, we use Lemma 50 to obtain the following result. As long as $N_\mathrm{Q} = \mathcal{O}(1)$, we can estimate the absolute value of $\operatorname{tr}(O\rho)$ for any observable O given in Definition 14 to an error 0.25 with probability at least 0.8.

Corollary 15. *Let $N_\mathrm{Q} = \Theta(1)$. For any observable O considered in Definition 14, we have*

$$\left|\hat{b} - |\operatorname{tr}(O\rho)|\right| \leq 0.25, \tag{9.18}$$

with probability at least 0.8.

This concludes the constant upper bound for quantum-enhanced experiments in Theorem 47.

An exponential lower bound for conventional experiments

The proof begins with a reduction to the partially-revealed many-versus-one distinguishing task followed by bounding the total variation distance.

Reduction to partially-revealed many-versus-one distinguishing task

We consider the following partially-revealed many-versus-one distinguishing task discussed in Section 3.2, namely where:

- The null hypothesis is $I/2^n$.

- The alternative hypothesis is $(I + 0.9sP)/2^n$.

The partially revealed information is the Pauli operator P. Recall the following from Definition 14,

1. With probability $1/2$, the state is $\rho = I/2^n$ and $O \in \{I, X, Y, Z\}^{\otimes n} \setminus \{I^{\otimes n}\}$ is sampled uniformly at random. (Corresponds to the null hypothesis)
2. With probability $1/2$, the state is $\rho = (I + 0.9sP)/2^n$ and $O = P$, where $s = \{\pm 1\}$ with equal probability and $P \in \{I, X, Y, Z\}^{\otimes n} \setminus \{I^{\otimes n}\}$ uniformly. (Corresponds to the alternative hypothesis)

For $\rho = I/2^n$, we have $|\operatorname{tr}(O\rho)| = 0$. For $\rho = (I + 0.9sP)/2^n$, we have $|\operatorname{tr}(O\rho)| = 0.9$. Therefore, if an algorithm could predict $|\operatorname{tr}(O\rho)|$ to 0.25 error with probability at least $1 - \delta$, it could be used to distinguish between the null and alternative hypotheses with success probability at least $1 - \delta$.

Total variation distance

From the information-theoretic lower bound for partially-revealed many-versus-one distinguishing task given in Section 3.2, if we let $p_\rho(\ell)$ be the leaf probability distribution under ρ, then

$$\mathop{\mathbb{E}}_{P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}} \mathrm{TV}\left(p_{I/2^n}, \mathop{\mathbb{E}}_{s \in \{\pm 1\}} p_{(I+0.9sP)/2^n}\right) \geq 1 - 2\delta. \tag{9.19}$$

For each leaf node ℓ, we consider the path from the root to ℓ,

$$u_0 = r \xrightarrow{s_1} u_1 \xrightarrow{s_2} u_2 \xrightarrow{s_3} \ldots \xrightarrow{s_{T-1}} u_{T-1} \xrightarrow{s_T} u_T = \ell. \tag{9.20}$$

At each node u, we perform a POVM measurement $\{w_s^u |\phi_s^u\rangle\langle\phi_s^u|\}_s$ on ρ to obtain an outcome s with probability

$$w_s^u \langle\phi_s^u| \rho |\phi_s^u\rangle. \tag{9.21}$$

Hence, we can write down the probability to arrive at the leaf ℓ as

$$p_\rho(\ell) = \prod_{t=1}^{T} w_{s_t}^{u_{t-1}} \langle\phi_{s_t}^{u_{t-1}}| \rho |\phi_{s_t}^{u_{t-1}}\rangle. \tag{9.22}$$

Recalling the definition of total variation distance, note that for any probability distributions p_A, p_B for which $p_A(\ell) > 0$ whenever $p_B(\ell) > 0$,

$$\mathrm{TV}(p_A, p_B) = \frac{1}{2} \sum_\ell |p_A(\ell) - p_B(\ell)| = \sum_\ell \max(0, p_A(\ell) - p_B(\ell)) \tag{9.23}$$

$$= \sum_{\ell} p_A(\ell) \cdot \max\left(0, 1 - \frac{p_B(\ell)}{p_A(\ell)}\right), \tag{9.24}$$

where the last equality follows from $\max(ax, ay) = a \max(x, y)$ for all $x, y \in \mathbb{R}$ and all $a \geq 0$.

Observe that for leaf ℓ,

$$\frac{\mathbb{E}_{s\in\{\pm1\}}\, p_{(I+0.9sP)/2^n}(\ell)}{p_{I/2^n}(\ell)} = \frac{\mathbb{E}_{s\in\{\pm1\}} \prod_{t=1}^{T} w_{s_t}^{u_{t-1}} \langle\phi_{s_t}^{u_{t-1}}|\, \frac{I+0.9sP}{2^n}\, |\phi_{s_t}^{u_{t-1}}\rangle}{\prod_{t=1}^{T} w_{s_t}^{u_{t-1}} \langle\phi_{s_t}^{u_{t-1}}|\, \frac{I}{2^n}\, |\phi_{s_t}^{u_{t-1}}\rangle} \tag{9.25}$$

$$= \mathop{\mathbb{E}}_{s\in\{\pm1\}} \prod_{t=1}^{T} \left(1 + 0.9s\, \langle\phi_{s_t}^{u_{t-1}}|\, P\, |\phi_{s_t}^{u_{t-1}}\rangle\right). \tag{9.26}$$

Combining (9.24) and (9.26), we can express the total variation distance inside the expectation in (9.19) as

$$\mathrm{TV}\left(p_{I/2^n}, \mathop{\mathbb{E}}_{s\in\{\pm1\}} p_{(I+0.9sP)/2^n}\right) \tag{9.27}$$

$$= \sum_{\ell} p_{I/2^n}(\ell)\ \max\left(0,\ 1 - \mathop{\mathbb{E}}_{s\in\{\pm1\}} \prod_{t=1}^{T} \left(1 + 0.9s\, \langle\phi_{s_t}^{u_{t-1}}|\, P\, |\phi_{s_t}^{u_{t-1}}\rangle\right)\right) \tag{9.28}$$

Upper bound for total variation distance

We analyze one of the terms in the total variation distance using Jensen's inequality (note that $\exp(x)$ is a convex function in x).

$$\mathop{\mathbb{E}}_{s\in\{\pm1\}} \prod_{t=1}^{T} \left(1 + 0.9s\, \langle\phi_{s_t}^{u_{t-1}}|\, P\, |\phi_{s_t}^{u_{t-1}}\rangle\right) \tag{9.29}$$

$$= \mathop{\mathbb{E}}_{s\in\{\pm1\}} \exp\left[\sum_{t=1}^{T} \log\left(1 + 0.9s\, \langle\phi_{s_t}^{u_{t-1}}|\, P\, |\phi_{s_t}^{u_{t-1}}\rangle\right)\right] \tag{9.30}$$

$$\geq \exp\left[\mathop{\mathbb{E}}_{s\in\{\pm1\}} \sum_{t=1}^{T} \log\left(1 + 0.9s\, \langle\phi_{s_t}^{u_{t-1}}|\, P\, |\phi_{s_t}^{u_{t-1}}\rangle\right)\right] \tag{9.31}$$

$$= \exp\left[\sum_{t=1}^{T} \frac{1}{2} \log\left(1 - 0.81\, \langle\phi_{s_t}^{u_{t-1}}|\, P\, |\phi_{s_t}^{u_{t-1}}\rangle^2\right)\right] \tag{9.32}$$

$$= \prod_{t=1}^{T} \sqrt{1 - 0.81\, \langle\phi_{s_t}^{u_{t-1}}|\, P\, |\phi_{s_t}^{u_{t-1}}\rangle^2}\,. \tag{9.33}$$

We can then upper bound the total variation distance as

$$\mathrm{TV}\left(p_{I/2^n}, \mathop{\mathbb{E}}_{s\in\{\pm1\}} p_{(I+0.9sP)/2^n}\right) \tag{9.34}$$

$$\leq \sum_{\ell} p_{I/2^n}(\ell) \, \max\left(0, \; 1 - \prod_{t=1}^{T} \sqrt{1 - 0.81 \, \langle \phi_{s_t}^{u_{t-1}} | \, P \, | \phi_{s_t}^{u_{t-1}} \rangle^2}\right) \tag{9.35}$$

$$= \sum_{\ell} p_{I/2^n}(\ell) \left(1 - \prod_{t=1}^{T} \sqrt{1 - 0.81 \, \langle \phi_{s_t}^{u_{t-1}} | \, P \, | \phi_{s_t}^{u_{t-1}} \rangle^2}\right). \tag{9.36}$$

The last equality follows from the fact that all eigenvalues of P are ±1, hence $1 \geq \prod_{t=1}^{T} \sqrt{1 - 0.81 \, \langle \phi_{s_t}^{u_{t-1}} | \, P \, | \phi_{s_t}^{u_{t-1}} \rangle^2}$.

Lower bound for the number of measurements

We can combine Eq. (9.36) and Eq. (9.19) to find

$$\mathop{\mathbb{E}}_{P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}} \sum_{\ell} p_{I/2^n}(\ell) \left(1 - \prod_{t=1}^{T} \sqrt{1 - 0.81 \, \langle \phi_{s_t}^{u_{t-1}} | \, P \, | \phi_{s_t}^{u_{t-1}} \rangle^2}\right) \geq 1 - 2\delta. \tag{9.37}$$

By linearity of expectation, we have

$$\sum_{\ell} p_{I/2^n}(\ell) \left(1 - \mathop{\mathbb{E}}_{P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}} \prod_{t=1}^{T} \sqrt{1 - 0.81 \, \langle \phi_{s_t}^{u_{t-1}} | \, P \, | \phi_{s_t}^{u_{t-1}} \rangle^2}\right) \geq 1 - 2\delta. \tag{9.38}$$

We analyze the expectation value term in the summand as follows:

$$\mathop{\mathbb{E}}_{P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}} \prod_{t=1}^{T} \sqrt{1 - 0.81 \, \langle \phi_{s_t}^{u_{t-1}} | \, P \, | \phi_{s_t}^{u_{t-1}} \rangle^2} \tag{9.39}$$

$$= \mathop{\mathbb{E}}_{P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}} \exp\left[\frac{1}{2} \sum_{t=1}^{T} \log\left(1 - 0.81 \, \langle \phi_{s_t}^{u_{t-1}} | \, P \, | \phi_{s_t}^{u_{t-1}} \rangle^2\right)\right] \tag{9.40}$$

$$\geq \exp\left[\frac{1}{2} \sum_{t=1}^{T} \mathop{\mathbb{E}}_{P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}} \log\left(1 - 0.81 \, \langle \phi_{s_t}^{u_{t-1}} | \, P \, | \phi_{s_t}^{u_{t-1}} \rangle^2\right)\right] \tag{9.41}$$

$$\geq \exp\left[\frac{1}{2} \sum_{t=1}^{T} \mathop{\mathbb{E}}_{P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}} -1.701 \, \langle \phi_{s_t}^{u_{t-1}} | \, P \, | \phi_{s_t}^{u_{t-1}} \rangle^2\right] \tag{9.42}$$

$$= \exp\left[-0.8505 \sum_{t=1}^{T} \frac{1}{2^n + 1}\right] = \exp\left(-\frac{0.8505T}{2^n + 1}\right). \tag{9.43}$$

The second line follows from Jensen's inequality because $\exp(x)$ is convex in x. The third line uses $\log(1 - x) \geq -2.1x, \forall x \in [0, 0.82]$. The fourth line uses the fact that

$$\mathop{\mathbb{E}}_{P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}} P \otimes P = \frac{2^n \mathrm{SWAP} - I \otimes I}{4^n - 1}, \tag{9.44}$$

hence $\mathbb{E}_{P\in\{I,X,Y,Z\}^{\otimes n}\setminus\{I^{\otimes n}\}} \langle\phi_{s_t}^{u_{t-1}}|\, P\, |\phi_{s_t}^{u_{t-1}}\rangle^2 = \frac{2^n-1}{4^n-1} = \frac{1}{2^n+1}$.

Combining the analysis with Eq. (9.38), we find that

$$\sum_{\ell} p_{I/2^n}(\ell) \left(1 - \exp\left(-\frac{0.8505T}{2^n+1}\right)\right) \geq 1 - 2\delta. \tag{9.45}$$

Because $\sum_\ell p_{I/2^n}(\ell) = 1$, we have

$$1 - \exp\left(-\frac{0.8505T}{2^n+1}\right) \geq 1 - 2\delta. \tag{9.46}$$

Together, the following lower bound on the number of experiments can be obtained:

$$T \geq \frac{2^n+1}{0.8505} \log\left(\frac{1}{2\delta}\right). \tag{9.47}$$

After setting $\delta = 0.2$ (corresponding to a success probability of at least 0.8), we conclude the exponential lower bound for conventional experiments in Theorem 47.

An exponential lower bound for comparing absolute values

In the physical experiment presented in the main text, we considered a slightly different task that also yields an exponential lower bound for conventional experiments. This slightly different task has two main differences when compared with the task described in Section 9.5. First, we do not consider the maximally mixed state $I/2^n$. Second, we ask the learner to predict which of two observables O_1, O_2 has a larger absolute value. The task description is given below.

Task 2 (Comparing absolute values)**.** *There is an unknown state $\rho = (I + 0.9sP)/2^n$ where $s = \{\pm 1\}$ and $P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}$ are both sampled uniformly at random. The algorithm learns about ρ through conventional or quantum-enhanced strategies. The algorithm transforms all quantum data to classical data. After learning, we present the learning algorithm with*

$$O_1 = P,\ O_2 = Q \quad \text{or} \quad O_1 = Q,\ O_2 = P, \tag{9.48}$$

with equal probability, where $Q \neq P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}$ is sampled uniformly. The learning algorithm succeeds if it correctly classifies whether $|\operatorname{tr}(O_1\rho)| > |\operatorname{tr}(O_2\rho)|$ or $|\operatorname{tr}(O_1\rho)| < |\operatorname{tr}(O_2\rho)|$.

Using the procedure presented in Section 9.5, it is not hard to show that quantum-enhanced strategies could accomplish the above task with classification accuracy (i.e., the probability that the classification is correct) at least $1-\delta$ from only $\mathcal{O}(\log(1/\delta))$ experiments. In contrast, we have the following theorem for conventional experiments.

Theorem 48 (Exponential lower bound for Task 2)**.** *A learning algorithm in the conventional setting (without quantum memory) requires at least*

$$\frac{(2^n+1)}{0.8505}\log\left(\frac{2}{1+2\delta}\right) \tag{9.49}$$

experiments to accomplish Task 2 with an accuracy of $1-\delta$*, for a given* $\delta > 0$*.*

Lower bound for total variation distance

Here we begin the proof of Theorem 48. Task 2 is closely related to the partially-revealed many-versus-one distinguishing task, but is not exactly the same. We will use a slightly different information-theoretic bound for this task. Let us define the following notation

$$\rho_{sP} \equiv \frac{I+0.9sP}{2^n}. \tag{9.50}$$

We consider a learning algorithm in the conventional setting. We consider the probability distribution $p_\rho(\ell)$ over the leaf node ℓ when the underlying state is ρ. Recall that the leaf node ℓ is the final memory state of the learning algorithm. Any procedure that makes the prediction based on the final memory state of the learning algorithm must have a classification accuracy upper bounded by

$$\frac{1}{(4^n-1)(4^n-2)}\sum_{P\neq Q\in\{I,X,Y,Z\}^{\otimes n}\setminus\{I^{\otimes n}\}}\left[\frac{1}{2}+\frac{1}{2}\mathrm{TV}\left(\mathop{\mathbb{E}}_{s\in\{\pm 1\}} p_{\rho_{sP}}(\ell), \mathop{\mathbb{E}}_{s\in\{\pm 1\}} p_{\rho_{sQ}}(\ell)\right)\right]. \tag{9.51}$$

To understand why the above inequality holds, consider a fixed $P \neq Q$. There is an equal probability that the underlying state is $\rho_{+P}, \rho_{-P}, \rho_{+Q}$, or ρ_{-Q} because $s \in \{\pm 1\}$ with equal probability and $O_1 = P, O_2 = Q$ or $O_1 = Q, O_2 = P$ with probability $1/2$. In order to distinguish the event of ρ_{+P}, ρ_{-P} from the event ρ_{+Q}, ρ_{-Q} based on the leaf node ℓ, we need the two distributions $\mathbb{E}_{s\in\{\pm 1\}} p_{\rho_{sP}}(\ell)$ and $\mathbb{E}_{s\in\{\pm 1\}} p_\ell(\rho_{sQ})$ to be sufficiently distinct. Formally, one can show that the success probability is upper bounded by

$$\frac{1}{2}+\frac{1}{2}\mathrm{TV}\left(\mathop{\mathbb{E}}_{s\in\{\pm 1\}} p_{\rho_{sP}}(\ell), \mathop{\mathbb{E}}_{s\in\{\pm 1\}} p_{\rho_{sQ}}(\ell)\right) \tag{9.52}$$

using LeCam's two-point method, see e.g. Lemma 1 in (B. Yu, 1997). To achieve the above success probability, one can use the maximum likelihood protocol that outputs P if $\mathbb{E}_{s\in\{\pm1\}} p_{\rho_{sP}}(\ell) > \mathbb{E}_{s\in\{\pm1\}} p_{\rho_{sQ}}(\ell)$ and outputs Q otherwise. Because P, Q are both chosen uniformly at random (but distinct), the average classification accuracy is given in Eq. (9.51). If the learning algorithm could achieve an accuracy of $1-\delta$, we would have

$$(1-\delta) \leq \frac{1}{2} + \frac{1}{2(4^n-1)(4^n-2)} \sum_{P\neq Q \in \{I,X,Y,Z\}^{\otimes n}\setminus\{I^{\otimes n}\}} \tag{9.53}$$

$$\mathrm{TV}\left(\mathop{\mathbb{E}}_{s\in\{\pm1\}} p_{\rho_{sP}}(\ell), \mathop{\mathbb{E}}_{s\in\{\pm1\}} p_{\rho_{sQ}}(\ell)\right). \tag{9.54}$$

This implies that

$$1-2\delta \leq \frac{1}{(4^n-1)(4^n-2)} \sum_{P\neq Q \in \{I,X,Y,Z\}^{\otimes n}\setminus\{I^{\otimes n}\}} \tag{9.55}$$

$$\mathrm{TV}\left(\mathop{\mathbb{E}}_{s\in\{\pm1\}} p_{\rho_{sP}}(\ell), \mathop{\mathbb{E}}_{s\in\{\pm1\}} p_{\rho_{sQ}}(\ell)\right). \tag{9.56}$$

Upper bound for total variation distance

We now perform triangle inequalities and reuse inequalities in Section 9.5 to upper bound the total variation distance:

$$\mathrm{TV}\left(\mathop{\mathbb{E}}_{s\in\{\pm1\}} p_{\rho_{sP}}(\ell), \mathop{\mathbb{E}}_{s\in\{\pm1\}} p_{\rho_{sQ}}(\ell)\right) \tag{9.57}$$

$$\leq \mathrm{TV}\left(p_{I/2^n}(\ell), \mathop{\mathbb{E}}_{s\in\{\pm1\}} p_{\rho_{sP}}(\ell)\right) + \mathrm{TV}\left(p_{I/2^n}(\ell), \mathop{\mathbb{E}}_{s\in\{\pm1\}} p_{\rho_{sQ}}(\ell)\right) \tag{9.58}$$

$$\leq \sum_\ell p_{I/2^n}(\ell)\left(1-\prod_{t=1}^T \sqrt{1-0.81\,\langle\phi_{s_t}^{u_{t-1}}|\,P\,|\phi_{s_t}^{u_{t-1}}\rangle^2}\right)$$

$$+\sum_\ell p_{I/2^n}(\ell)\left(1-\prod_{t=1}^T \sqrt{1-0.81\,\langle\phi_{s_t}^{u_{t-1}}|\,Q\,|\phi_{s_t}^{u_{t-1}}\rangle^2}\right) \tag{9.59}$$

$$= 1-\sum_\ell p_{I/2^n}(\ell)\prod_{t=1}^T \sqrt{1-0.81\,\langle\phi_{s_t}^{u_{t-1}}|\,P\,|\phi_{s_t}^{u_{t-1}}\rangle^2}$$

$$+1-\sum_\ell p_{I/2^n}(\ell)\prod_{t=1}^T \sqrt{1-0.81\,\langle\phi_{s_t}^{u_{t-1}}|\,Q\,|\phi_{s_t}^{u_{t-1}}\rangle^2}. \tag{9.60}$$

The first line is triangle inequality. The second inequality uses Eq. (9.36) for both P and Q. The equality step uses $\sum_\ell p_{I/2^n}(\ell) = 1$. Therefore, we have

$$\frac{1}{(4^n-1)(4^n-2)} \sum_{P\neq Q \in \{I,X,Y,Z\}^{\otimes n}\setminus\{I^{\otimes n}\}} \mathrm{TV}\left(\mathop{\mathbb{E}}_{s\in\{\pm1\}} p_{\rho_{sP}}(\ell), \mathop{\mathbb{E}}_{s\in\{\pm1\}} p_{\rho_{sQ}}(\ell)\right) \tag{9.61}$$

$$\leq 1 - \sum_{\ell} p_{I/2^n}(\ell) \underset{P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}}{\mathbb{E}} \prod_{t=1}^{T} \sqrt{1 - 0.81 \langle \phi_{s_t}^{u_{t-1}} | P | \phi_{s_t}^{u_{t-1}} \rangle^2}$$

$$+ 1 - \sum_{\ell} p_{I/2^n}(\ell) \underset{Q \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}}{\mathbb{E}} \prod_{t=1}^{T} \sqrt{1 - 0.81 \langle \phi_{s_t}^{u_{t-1}} | Q | \phi_{s_t}^{u_{t-1}} \rangle^2} \tag{9.62}$$

$$= 1 - \sum_{\ell} p_{I/2^n}(\ell) \exp\left(-\frac{0.8505T}{2^n + 1}\right) + 1 - \sum_{\ell} p_{I/2^n}(\ell) \exp\left(-\frac{0.8505T}{2^n + 1}\right) \tag{9.63}$$

$$= 2 - 2\exp\left(-\frac{0.8505T}{2^n + 1}\right). \tag{9.64}$$

In the above inequalities, the first inequality uses Eq. (9.60). The equality thereafter uses the following analysis,

$$\frac{1}{(4^n - 1)(4^n - 2)} \sum_{P \neq Q \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}} f(P) \tag{9.65}$$

$$= \frac{1}{4^n - 1} \sum_{P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}} f(P) \left(\frac{1}{4^n - 2} \sum_{\substack{Q \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\} \\ \text{s.t.}, Q \neq P}} 1 \right) \tag{9.66}$$

$$= \frac{1}{4^n - 1} \sum_{P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}} f(P) \tag{9.67}$$

$$= \underset{P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}}{\mathbb{E}} f(P), \tag{9.68}$$

where $f(P) = \sum_{\ell} p_{I/2^n}(\ell) \prod_{t=1}^{T} \sqrt{1 - 0.81 \langle \phi_{s_t}^{u_{t-1}} | Q | \phi_{s_t}^{u_{t-1}} \rangle^2}$, as well as linearity of expectation, i.e.

$$\underset{P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}}{\mathbb{E}} \sum_{\ell} p_{I/2^n}(\ell) = \sum_{\ell} p_{I/2^n}(\ell) \underset{P \in \{I,X,Y,Z\}^{\otimes n} \setminus \{I^{\otimes n}\}}{\mathbb{E}}. \tag{9.69}$$

The third step in Eq. (9.63) uses Eq. (9.39) to (9.43), and the final step uses $\sum_{\ell} p_{I/2^n}(\ell) = 1$.

Combining upper and lower bounds

We can combine the lower bound obtained in Eq. (9.55) and the upper bound in Eq. (9.64) to find

$$1 - 2\delta \leq 2 - 2\exp\left(-\frac{0.8505T}{2^n + 1}\right). \tag{9.70}$$

Basic algebraic manipulations give

$$\frac{0.8505T}{2^n + 1} \geq -\log\left(\frac{1}{2} + \delta\right) = \log\left(\frac{2}{1 + 2\delta}\right). \tag{9.71}$$

We have thus concluded the desired lower bound

$$T \geq \frac{(2^n + 1)}{0.8505} \log\left(\frac{2}{1 + 2\delta}\right) \tag{9.72}$$

stated in Theorem 48.

9.6 Quantum advantage in principal component analysis

The (first) principal component of a nonnegative Hermitian matrix A is the eigenvector of A with the largest eigenvalue. Here, we consider a well-known task called quantum principal component analysis (PCA), which can be achieved efficiently using the quantum algorithm given in (Lloyd, Masoud Mohseni, and Rebentrost, 2014). The formal definition of the quantum PCA task is given in the following definition.

Task 3 (Quantum principal component analysis task)**.** *Let ρ be an unknown n-qubit mixed state whose top eigenvector $|\phi\rangle$ has eigenvalue larger than all other eigenvalues by a constant factor independent of n. Given a fixed observable O, we would like to predict $\langle\phi| O |\phi\rangle$ up to a small additive error.*

We can accomplish this task using the quantum PCA algorithm in (Lloyd, Masoud Mohseni, and Rebentrost, 2014). In this algorithm, multiple copies of ρ are used in a protocol that approximates the unitary operator $\sum_t |t\rangle\langle t| \otimes \exp(-i\rho t)$, where t is the "time" stored in an auxiliary register. By applying this conditional $\exp(-i\rho t)$ operation to the initial state $\rho = \sum_i \lambda_i |\phi_i\rangle\langle\phi_i|$, performing the quantum Fourier transform and measuring the auxiliary register, we read out an eigenvalue λ_i and prepare the corresponding eigenstate $|\phi_i\rangle$ with probability λ_i. The eigenvalue can be measured with constant accuracy, and the eigenstate prepared with constant fidelity, using a constant number of copies of ρ. Once $|\phi_i\rangle$ has been prepared, we can measure the observable O in this state.

By assumption, the largest eigenvalue of ρ is a constant independent of n, and furthermore is greater than all other eigenvalues by a constant. Hence by repeating the above procedure a constant number of times, we can estimate $\langle\phi|O|\phi\rangle$ to constant accuracy, where $|\phi\rangle$ is the eigenstate of ρ with the largest eigenvalue. In contrast, in Section 9.6 we show that algorithms that can only learn about ρ through conventional experiments require an exponential number of copies of ρ. Bringing these arguments all together, we establish the following theorem.

Theorem 49 (Exponential advantage for quantum principal component analysis)**.** *Let ρ be an unknown n-qubit mixed state (for $n > 1$) whose top eigenvector $|\phi\rangle$ has eigenvalue larger than all the other eigenvalues by a constant, and let Z_1 be an observable which is equal to the Pauli-Z operator on the first qubit. Algorithms learn about ρ through conventional or quantum-enhanced experiments.*

- Upper bound: *There is an algorithm in the* ***quantum-enhanced*** *scenario using only $O(1)$ copies of ρ to predict $\langle\phi| Z_1 |\phi\rangle$ up to* 0.25 *error with probability at least* 0.8.
- Lower bound: *Any algorithm in the* ***conventional*** *scenario needs at least $\Omega(2^{n/2})$ copies of ρ to predict $\langle\phi| Z_1 |\phi\rangle$ up to* 0.25 *error with probability at least* 0.8.

Instead of estimating $\langle\phi| Z_1 |\phi\rangle$, we can also consider the near-term proposals (J. Cotler and Wilczek, 2020a; Huggins et al., 2020) to obtain some information about the principal component $|\psi\rangle$ of an unknown state ρ as follows. The proposals consider $\rho^M/\mathrm{tr}(\rho^M)$, which approaches $|\psi\rangle\langle\psi|$ when M is large. In particular, (Huggins et al., 2020) shows that one can efficiently estimate $\mathrm{tr}(Z_1\rho^2)/\mathrm{tr}(\rho^2)$ by performing entangling Bell measurements over at most two copies of ρ at a time. By the analysis in (Huggins et al., 2020), if the eigenvalue associated to the principal component of ρ is a constant, then $\mathrm{tr}(Z_i\rho^2)/\mathrm{tr}(\rho^2)$ can be estimated to any constant error by performing quantum-enhanced experiments over a constant number of copies of ρ. In contrast, we show that if one can only measure a single copy of ρ at a time, exponentially many copies are necessary to estimate $\mathrm{tr}(Z_i\rho^2)/\mathrm{tr}(\rho^2)$.

Theorem 50 (Exponential advantage for near-term quantum principal component analysis)**.** *Suppose we are given an observable Z_1 which is equal to the Pauli-Z operator on the first qubit, as well as an n-qubit state ρ (for $n > 1$) where an eigenvector $|\phi\rangle$ of ρ has an eigenvalue that is larger than all the other eigenvalues by a constant. We consider algorithms which learn about ρ through conventional or quantum-enhanced experiments. Then we have the following bounds:*

- Upper bound: *There is an algorithm in the* ***quantum-enhanced*** *scenario using only $O(1)$ copies of ρ to predict* $\mathrm{tr}(Z_1\rho^2)/\mathrm{tr}(\rho^2)$ *up to* 0.25 *error with probability at least* 0.8.

- Lower bound: *Any algorithm in the* ***conventional*** *scenario needs at least* $\Omega(2^{n/2})$ *copies of* ρ *to predict* $\mathrm{tr}(Z_1\rho^2)/\mathrm{tr}(\rho^2)$ *up to* 0.25 *error with probability at least* 0.8.

An exponential lower bound for conventional experiments

The lower bound proofs for both Theorem 49 and Theorem 50 are essentially the same. We first reduce quantum PCA (or near-term analogs thereof) to a many-versus-many distinguishing task. Then we bound the total variation distance to arrive at the exponential lower bound.

Reduction to many-versus-many distinguishing task

We begin by considering a many-versus-many distinguishing task, as discussed in Section 3.2. The two hypotheses are given below.

- Hypothesis A: The unknown n-qubit state ρ is given by

$$\rho_A(|\psi\rangle) = \frac{1}{2}|0\rangle\langle 0| \otimes |\psi\rangle\langle\psi| + \frac{1}{2}|1\rangle\langle 1| \otimes \frac{I}{2^{n-1}}, \tag{9.73}$$

  where $|\psi\rangle$ is an fixed $(n-1)$-qubit pure state, sampled at the outset from the Haar measure.

- Hypothesis B: The unknown n-qubit state ρ is given by

$$\rho_B(|\psi\rangle) = \frac{1}{2}|1\rangle\langle 1| \otimes |\psi\rangle\langle\psi| + \frac{1}{2}|0\rangle\langle 0| \otimes \frac{I}{2^{n-1}}, \tag{9.74}$$

  where $|\psi\rangle$ is again a fixed $(n-1)$-qubit pure state, sampled at the outset from the Haar measure.

It is not hard to see that in hypothesis A, the principal component (largest eigenvector) is $|\phi\rangle = |0\rangle \otimes |\psi\rangle$. On the other hand, in hypothesis B, the principal component (largest eigenvector) is $|\phi\rangle = |1\rangle \otimes |\psi\rangle$. Hence, $\langle\phi| Z_1 |\phi\rangle = 1$ in hypothesis A, but $\langle\phi| Z_1 |\phi\rangle = -1$ in hypothesis B. If an algorithm in the conventional scenario can predict $\langle\psi| Z_1 |\psi\rangle$ up to 0.25 error with probability at least 0.8, then we can use the output from the algorithm to distinguish between hypotheses A and B with a success probability of at least 0.8.

Similarly, we have $\mathrm{tr}(Z_1\rho^2)/\mathrm{tr}(\rho^2) = (2^{n-1} - 1)/(2^{n-1} + 1)$ in hypothesis A and $\mathrm{tr}(Z_1\rho^2)/\mathrm{tr}(\rho^2) = -(2^{n-1} - 1)/(2^{n-1} + 1)$ in hypothesis B. If an algorithm in the conventional scenario can predict $\mathrm{tr}(Z_1\rho^2)/\mathrm{tr}(\rho^2)$ up to 0.25 error with probability at least 0.8, then we can use the output from the algorithm to distinguish between hypothesis A and B with a success probability of at least 0.8.

Together, a lower bound for distinguishing hypotheses A and B using conventional experiments immediately implies a lower bound for both Theorem 49 and Theorem 50.

Total variation distance

As in previous sections, let $p_\rho(\ell)$ denote the probability to arrive at the leaf node ℓ using the learning algorithm in the conventional setting when the unknown state is ρ. If the algorithm can distinguish between hypotheses A and B with success probability 0.8, then using Eq. (3.22) we have

$$\mathrm{TV}\left(\mathop{\mathbb{E}}_{|\psi\rangle} p_{\rho_A(|\psi\rangle)}, \mathop{\mathbb{E}}_{|\psi\rangle} p_{\rho_B(|\psi\rangle)}\right) \geq 0.6. \tag{9.75}$$

From the triangle inequality, we have

$$\mathrm{TV}\left(\mathop{\mathbb{E}}_{|\psi\rangle} p_{\rho_A(|\psi\rangle)}, p_{I/2^n}\right) + \mathrm{TV}\left(\mathop{\mathbb{E}}_{|\psi\rangle} p_{\rho_B(|\psi\rangle)}, p_{I/2^n}\right) \geq 0.6. \tag{9.76}$$

For each leaf node ℓ, we consider the path from the root to ℓ,

$$u_0 = r \xrightarrow{s_1} u_1 \xrightarrow{s_2} u_2 \xrightarrow{s_3} \ldots \xrightarrow{s_{T-1}} u_{T-1} \xrightarrow{s_T} u_T = \ell. \tag{9.77}$$

At each node u, we perform a POVM measurement $\{w_s^u |\phi_s^u\rangle\langle\phi_s^u|\}_s$ on ρ to obtain an outcome s with probability

$$w_s^u \langle\phi_s^u| \rho |\phi_s^u\rangle . \tag{9.78}$$

Hence, we can write down the probability to arrive at the leaf ℓ as

$$p_\rho(\ell) = \prod_{t=1}^{T} w_{s_t}^{u_{t-1}} \langle\phi_{s_t}^{u_{t-1}}| \rho |\phi_{s_t}^{u_{t-1}}\rangle . \tag{9.79}$$

We will use $\rho(|\psi\rangle)$ to denote either $\rho_A(|\psi\rangle)$ or $\rho_B(|\psi\rangle)$. Then recalling (9.24), we have

$$\mathrm{TV}\left(\mathop{\mathbb{E}}_{|\psi\rangle} p_{\rho(|\psi\rangle)}, p_{I/2^n}\right) \tag{9.80}$$

$$= \sum_{\ell} p_{I/2^n}(\ell) \max\left(0,\ 1 - \mathop{\mathbb{E}}_{|\psi\rangle} \prod_{t=1}^{T} 2^n \langle\phi_{s_t}^{u_{t-1}}| \rho(|\psi\rangle) |\phi_{s_t}^{u_{t-1}}\rangle\right). \tag{9.81}$$

Upper bound for total variation distance

The central quantity to control in our bound on the total variation distance is

$$\mathop{\mathbb{E}}_{|\psi\rangle} \prod_{t=1}^{T} 2^n \langle \phi_{s_t}^{u_{t-1}} | \rho(|\psi\rangle) | \phi_{s_t}^{u_{t-1}} \rangle . \tag{9.82}$$

Without loss of generality, let us consider $\rho(|\psi\rangle) = \rho_A(|\psi\rangle) = \frac{1}{2}|0\rangle\langle 0| \otimes |\psi\rangle\langle\psi| + \frac{1}{2}|1\rangle\langle 1| \otimes \frac{I}{2^{n-1}}$. Suppose each $|\phi_{s_t}^{u_{t-1}}\rangle$ takes the form

$$|\phi_{s_t}^{u_{t-1}}\rangle = \alpha_{s_t}^{u_{t-1}} |0\rangle \otimes |\phi_{s_t}^{u_{t-1},0}\rangle + \beta_{s_t}^{u_{t-1}} |1\rangle \otimes |\phi_{s_t}^{u_{t-1},1}\rangle , \tag{9.83}$$

where $\alpha_{s_t}^{u_{t-1}}, \beta_{s_t}^{u_{t-1}} \in \mathbb{C}$ and $\left|\alpha_{s_t}^{u_{t-1}}\right|^2 + \left|\beta_{s_t}^{u_{t-1}}\right|^2 = 1$. Then we have

$$\begin{aligned}
&\mathop{\mathbb{E}}_{|\psi\rangle} \prod_{t=1}^{T} 2^n \langle \phi_{s_t}^{u_{t-1}} | \rho(|\psi\rangle) | \phi_{s_t}^{u_{t-1}} \rangle && (9.84)\\
&= \mathop{\mathbb{E}}_{|\psi\rangle} \prod_{t=1}^{T} \left(2^{n-1} \left|\alpha_{s_t}^{u_{t-1}}\right|^2 \left|\langle \psi | \phi_{s_t}^{u_{t-1},0}\rangle\right|^2 + \left|\beta_{s_t}^{u_{t-1}}\right|^2 \right) && (9.85)\\
&= \sum_{S \subseteq \{1,\ldots,T\}} \prod_{t \notin S} \left|\beta_{s_t}^{u_{t-1}}\right|^2 \left[\mathop{\mathbb{E}}_{|\psi\rangle} \prod_{t \in S} 2^{n-1} \left|\alpha_{s_t}^{u_{t-1}}\right|^2 \left|\langle \psi | \phi_{s_t}^{u_{t-1},0}\rangle\right|^2 \right]. && (9.86)
\end{aligned}$$

We need to lower bound the above quantity in order to upper bound the total variation distance. In order to do so, we utilize the following lemma. The proof of the lemma is based on Haar integration. For readers unfamiliar with Haar integration, we would suggest skipping the proof of this lemma.

Lemma 51 (High moment bound for Haar-random state)**.** *Consider any m-qubit pure states* $|\phi_1\rangle, \ldots, |\phi_K\rangle$ *and an m-qubit pure state* $|\psi\rangle$ *sampled from the Haar measure, we have*

$$\mathop{\mathbb{E}}_{|\psi\rangle} \prod_{k=1}^{K} |\langle \psi | \phi_k \rangle|^2 \geq \frac{1}{(2^m + K - 1) \ldots (2^m + 1)(2^m)}. \tag{9.87}$$

Proof. The Haar integration over states shows that

$$\mathop{\mathbb{E}}_{|\psi\rangle} |\psi\rangle\langle\psi|^{\otimes K} = \frac{1}{(2^m + K - 1) \ldots (2^m + 1)(2^m)} \sum_{\pi \in \mathcal{S}_K} \pi, \tag{9.88}$$

where $\mathcal{S}_K$ is the permutation group of K items, and π is the permutation operator over the K tensor-product space. From Lemma 5.12 in (Sitan Chen, J. Cotler, et al., 2021b), we have

$$\sum_{\pi \in \mathcal{S}_K} \operatorname{tr} \left(\pi \bigotimes_{k=1}^{K} |\phi_k\rangle\langle\phi_k| \right) \geq 1. \tag{9.89}$$

Therefore, we find

$$\mathop{\mathbb{E}}_{|\psi\rangle} \prod_{k=1}^{K} |\langle\psi|\phi_k\rangle|^2 \geq \frac{1}{(2^m + K - 1)\dots(2^m+1)(2^m)}. \tag{9.90}$$

This concludes the proof. □

We apply this lemma with

$$m \equiv n-1, \quad K \equiv |S|, \quad |\phi_k\rangle \equiv |\phi_{s_t}^{u_{t-1},0}\rangle \tag{9.91}$$

to obtain the following lower bound,

$$\mathop{\mathbb{E}}_{|\psi\rangle} \prod_{t\in S} 2^{n-1} \left|\alpha_{s_t}^{u_{t-1}}\right|^2 \left|\langle\psi|\phi_{s_t}^{u_{t-1},0}\rangle\right|^2 \geq \prod_{t\in S} \frac{\left|\alpha_{s_t}^{u_{t-1}}\right|^2}{(1+\frac{|S|-1}{2^{n-1}})\dots(1+\frac{1}{2^{n-1}})(1)} \tag{9.92}$$

$$\geq \prod_{t\in S} \frac{\left|\alpha_{s_t}^{u_{t-1}}\right|^2}{\left(1+\frac{|S|-1}{2^{n-1}}\right)^{|S|-1}} \tag{9.93}$$

$$\geq \left(1+\frac{T-1}{2^{n-1}}\right)^{-(T-1)} \prod_{t\in S} \left|\alpha_{s_t}^{u_{t-1}}\right|^2. \tag{9.94}$$

Combining with Eq. (9.86), we have

$$\mathop{\mathbb{E}}_{|\psi\rangle} \prod_{t=1}^{T} 2^n \langle\phi_{s_t}^{u_{t-1}}|\,\rho(|\psi\rangle)\,|\phi_{s_t}^{u_{t-1}}\rangle \tag{9.95}$$

$$\geq \left(1+\frac{T-1}{2^{n-1}}\right)^{-(T-1)} \sum_{S\subseteq\{1,\dots,T\}} \prod_{t\notin S} \left|\beta_{s_t}^{u_{t-1}}\right|^2 \prod_{t\in S} \left|\alpha_{s_t}^{u_{t-1}}\right|^2 \tag{9.96}$$

$$= \left(1+\frac{T-1}{2^{n-1}}\right)^{-(T-1)} \prod_{t=1}^{T} \left(\left|\beta_{s_t}^{u_{t-1}}\right|^2 + \left|\alpha_{s_t}^{u_{t-1}}\right|^2\right) \tag{9.97}$$

$$= \left(1+\frac{T-1}{2^{n-1}}\right)^{-(T-1)}. \tag{9.98}$$

Next we leverage Eq. (9.81) to obtain

$$\mathrm{TV}\left(\mathop{\mathbb{E}}_{|\psi\rangle} p_{\rho(|\psi\rangle)}, p_{I/2^n}\right) \leq \sum_{\ell} p_{I/2^n}(\ell) \max\left(0,\ 1-\left(1+\frac{T-1}{2^{n-1}}\right)^{-(T-1)}\right) \tag{9.99}$$

$$= 1-\left(1+\frac{T-1}{2^{n-1}}\right)^{-(T-1)}. \tag{9.100}$$

The second line follows from $\sum_\ell p_{I/2^n}(\ell) = 1$ because $p_{I/2^n}(\ell)$ is a probability distribution.

Lower bound for the number of measurements

We can now utilize the lower bound on the total variation distance given in Eq. (9.76) and the upper bound obtained above to find

$$2\left(1-\left(1+\frac{T-1}{2^{n-1}}\right)^{-(T-1)}\right) \tag{9.101}$$

$$\geq \mathrm{TV}\left(\mathop{\mathbb{E}}_{|\psi\rangle} p_{\rho_A(|\psi\rangle)}, p_{I/2^n}\right)+\mathrm{TV}\left(\mathop{\mathbb{E}}_{|\psi\rangle} p_{\rho_B(|\psi\rangle)}, p_{I/2^n}\right) \geq 0.6. \tag{9.102}$$

Hence, we have the inequality

$$0.7 \geq \left(1+\frac{T-1}{2^{n-1}}\right)^{-(T-1)}, \tag{9.103}$$

$$\implies (T-1)\log\left(1+\frac{T-1}{2^{n-1}}\right) \geq \log(10/7). \tag{9.104}$$

Because $\log(1+x) \leq x$ for all $x > -1$, we have

$$(T-1)^2 \geq 2^{n-1}\log(10/7) \implies T \geq 1+\sqrt{\frac{\log\left(\frac{10}{7}\right)}{2}}\, 2^{n/2}. \tag{9.105}$$

Finally, we have established the lower bound $T = \Omega(2^{n/2})$ stated in Theorem 49.

An exponential lower bound using pseudorandomness

In our exponential lower bound in the previous subsection, we relied on a state that was in part constructed using a Haar-random $(n-1)$-qubit state $|\psi\rangle$. However, preparing a Haar-random state from a simple initial state (say, a product state) requires circuit depth exponential in n. As such, we can not prepare Haar-random states in practice. Accordingly we cannot prepare either $\rho_A(|\psi\rangle)$ or $\rho_B(|\psi\rangle)$ in realistic circumstances.

However, we could instead efficiently construct pseudorandom states $|\psi\rangle$ which are (very plausibly) indistinguishable from Haar-random states if we probe with any POVM instantiated by a poly(n)-time quantum algorithm (Ji, Y.-K. Liu, and Song, 2018). We will elaborate on this shortly. The parenthetical caveat 'very plausibly' is due to the fact that the construction we use relies on cryptographic assumptions which are not proven, but are widely believed. In particular, we need to suppose the existence of quantum-secure one-way functions (Ji, Y.-K. Liu, and Song, 2018); these are functions which are efficient to evaluate but are hard to invert even with a quantum computer. Making such an assumption is standard practice in computational complexity theory, and so we proceed apace.

Let us recall the definition of a pseudorandom quantum state:

Definition 15 (Pseudorandom quantum states; paraphrased from Definition 3 of (Ji, Y.-K. Liu, and Song, 2018))**.** *Given a set $\mathcal{K}$, a family of pseudorandom quantum states on n qubits is a family of states $\{|\phi_k\rangle\}_{k\in\mathcal{K}}$ and a probability distribution $\mathcal{D}$ over $\mathcal{K}$ such that:*

- *There is a $poly(n)$-time quantum algorithm that samples a single element k from $\mathcal{K}$ according to $\mathcal{D}$ and generates the corresponding state $|\phi_k\rangle$;*
- *For any polynomial $t(n)$ and any $poly(n)$-time quantum algorithm $\mathcal{A}$ with outputs in $\{0, 1\}$, we have*

$$\Big| Pr_{k\leftarrow\mathcal{K}}\big[\mathcal{A}(|\phi_k\rangle^{\otimes t(n)}) = 1\big] - Pr_{|\psi\rangle\leftarrow Haar}\big[\mathcal{A}(|\psi\rangle^{\otimes t(n)}) = 1\big]\Big| \leq negl(n). \tag{9.106}$$

Here $negl(n)$ is a function such that for all constants $c > 0$ we have $negl(n) < n^{-c}$ for n sufficiently large.

We can interpret the above definition as follows. Eq. (9.106) says that if we are given a polynomial $t(n)$ number of copies of either (i) a fixed pseudorandom state, or (ii) a fixed Haar random state, any polynomial time quantum algorithm with binary outputs cannot distinguish between the two cases. Since an exponential depth quantum algorithm can distinguish between the two cases, (9.106) describes a notion of *computational indistinguishability*, i.e. we cannot distinguish with polynomial time computational resources.

A key question is: do pseudorandom quantum states exist? We recall the following, contingent result.

Lemma 52 (Existence of pseudorandom quantum states (Ji, Y.-K. Liu, and Song, 2018))**.** *If there exist quantum-secure one-way functions, then they can be used to construct pseudorandom quantum states.*

Explicit details of the construction can be found in (Ji, Y.-K. Liu, and Song, 2018); there have also been refinements and generalizations in follow-up work (see e.g. (Brakerski and Shmueli, 2019; Brakerski and Shmueli, 2020)).

Before stating the main result of this section, we require the following definition:

Definition 16 (Polynomial-time algorithms in the conventional scenario)**.** *A poly-time algorithm $\mathcal{A}$ in the conventional scenario is constructed as follows. We consider a learning tree $\mathcal{T}$ in the* ***conventional scenario*** *with leaves ℓ, and require that the protocol described by the learning tree can be implemented by an at most $\mathrm{poly}(n)$-time quantum algorithm. (As such, the depth of $\mathcal{T}$ is at most polynomial in n.) Then let $\mathcal{D}$ be a $\mathrm{poly}(n)$-time classical algorithm which, given the transcript of measurements encoded into the leaves ℓ of $\mathcal{T}$, provides a binary output 0 or 1. We let $\mathcal{A}$ be a map from n-qubit density matrices ρ to $\{0, 1\}$, corresponding to instantiating the learning tree $\mathcal{T}$ on copies of ρ, followed by using $\mathcal{D}$ on the measurement transcript to determine a binary outcome.*

We can now leverage the putative pseudorandom states and the above definition to establish the following result:

Theorem 51 (Lower bound many-versus-many distinguishing task using pseudorandom states)**.** *Let $\{|\phi_k\rangle\}_{k\in\mathcal{K}}$ be a family of pseudorandom states on $n-1$ qubits. Then any polynomial-time algorithm $\mathcal{A}$ in the* ***conventional*** *scenario with binary output cannot distinguish $\rho_A(|\phi_k\rangle)$ from $\rho_B(|\phi_k\rangle)$ for k sampled from the probability distribution over $\mathcal{K}$. That is:*

$$\left|\, Pr_{k\leftarrow\mathcal{K}}\big[\mathcal{A}(\rho_A(|\phi_k\rangle)) = 1\big] - Pr_{k\leftarrow\mathcal{K}}\big[\mathcal{A}(\rho_B(|\phi_k\rangle)) = 1\big]\right| \leq \; negl(n)\,. \quad (9.107)$$

Proof. Using the triangle inequality several times we have

$$\begin{aligned}
&\left|\, \mathrm{Pr}_{k\leftarrow\mathcal{K}}\big[\mathcal{A}(\rho_A(|\phi_k\rangle)) = 1\big] - \mathrm{Pr}_{k\leftarrow\mathcal{K}}\big[\mathcal{A}(\rho_B(|\phi_k\rangle)) = 1\big]\right| \qquad (9.108)\\
&\quad \leq \left|\, \mathrm{Pr}_{|\psi\rangle\leftarrow\mathrm{Haar}}\big[\mathcal{A}(\rho_A(|\psi\rangle)) = 1\big] - \mathrm{Pr}_{|\psi\rangle\leftarrow\mathrm{Haar}}\big[\mathcal{A}(\rho_B(|\psi\rangle)) = 1\big]\right|\\
&\quad + \left|\, \mathrm{Pr}_{k\leftarrow\mathcal{K}}\big[\mathcal{A}(\rho_A(|\phi_k\rangle)) = 1\big] - \mathrm{Pr}_{|\psi\rangle\leftarrow\mathrm{Haar}}\big[\mathcal{A}(\rho_A(|\psi\rangle)) = 1\big]\right|\\
&\quad + \left|\, \mathrm{Pr}_{k\leftarrow\mathcal{K}}\big[\mathcal{A}(\rho_B(|\phi_k\rangle)) = 1\big] - \mathrm{Pr}_{|\psi\rangle\leftarrow\mathrm{Haar}}\big[\mathcal{A}(\rho_B(|\psi\rangle)) = 1\big]\right|\,.
\end{aligned}$$

Let $\mathcal{T}$ be the learning tree corresponding to $\mathcal{A}$, and let $\mathcal{D}$ be the binary decision function mapping leaves of $\mathcal{T}$ to $\{0, 1\}$. The depth T of $\mathcal{T}$ is necessarily at most polynomial in n; let us denote the depth by $T(n)$. Then the first term on the right-hand side of (9.108) is upper bounded by

$$\begin{aligned}
&\left|\, \mathrm{Pr}_{|\psi\rangle\leftarrow\mathrm{Haar}}\big[\mathcal{A}(\rho_A(|\psi\rangle)) = 1\big] - \mathrm{Pr}_{|\psi\rangle\leftarrow\mathrm{Haar}}\big[\mathcal{A}(\rho_B(|\psi\rangle)) = 1\big]\right| \qquad (9.109)\\
&= \left|\sum_{\ell\,\in\,\mathrm{leaf}(\mathcal{T})\,:\,\mathcal{D}(\ell)=1}\left(\mathop{\mathbb{E}}_{|\psi\rangle} p_{\rho_A(|\psi\rangle)}(\ell) - \mathop{\mathbb{E}}_{|\psi\rangle} p_{\rho_B(|\psi\rangle)}(\ell)\right)\right|
\end{aligned}$$

$$\begin{aligned}&\leq \sum_{\ell \in \operatorname{leaf}(\mathcal{T})} \left| \mathop{\mathbb{E}}_{|\psi\rangle} p_{\rho_A(|\psi\rangle)}(\ell) - \mathop{\mathbb{E}}_{|\psi\rangle} p_{\rho_B(|\psi\rangle)}(\ell) \right| \\ &\leq 2\,\mathrm{TV}\left(\mathop{\mathbb{E}}_{|\psi\rangle} p_{\rho_A(|\psi\rangle)}, p_{I/2^n} \right) + 2\,\mathrm{TV}\left(\mathop{\mathbb{E}}_{|\psi\rangle} p_{\rho_B(|\psi\rangle)}, p_{I/2^n} \right) \\ &\leq 4\left(1 - \left(1 + \frac{T(n)-1}{2^{n-1}}\right)^{-(T(n)-1)}\right) \end{aligned} \tag{9.110}$$

where the last inequality comes from (9.101).

Next we turn to bounding the second term on the right-hand side of the inequality in (9.108), namely

$$\left| \Pr_{k\leftarrow\mathcal{K}}\big[\mathcal{A}(\rho_A(|\phi_k\rangle)) = 1\big] - \Pr_{|\psi\rangle\leftarrow\mathrm{Haar}}\big[\mathcal{A}(\rho_A(|\psi\rangle)) = 1\big]\right| .$$

First, we observe that every $\mathcal{A}$ takes in $T(n)$ copies of ρ_A. Accordingly, there is a polynomial-time algorithm $\widetilde{\mathcal{A}}$ such that $\widetilde{\mathcal{A}}(\rho_A^{\otimes T(n)}) = \mathcal{A}(\rho_A)$ for all inputs ρ_A; this just amounts to a slightly different way of notating the domain of the algorithm $\mathcal{A}$. Then we can rewrite our term of interest as

$$\left| \Pr_{k\leftarrow\mathcal{K}}\big[\widetilde{\mathcal{A}}(\rho_A(|\phi_k\rangle)^{\otimes T(n)}) = 1\big] - \Pr_{|\psi\rangle\leftarrow\mathrm{Haar}}\big[\widetilde{\mathcal{A}}(\rho_A(|\psi\rangle)^{\otimes T(n)}) = 1\big]\right| . \tag{9.111}$$

But now observe that for any state $|\omega\rangle$, there is a polynomial-time quantum algorithm which takes $|\omega\rangle$ as input and produces $\rho_A(|\omega\rangle)$ as output. Accordingly, repeating this algorithm on $T(n)$ copies of $|\omega\rangle$, we produce $T(n)$ copies of $\rho_A(|\omega\rangle)$; let us denote this $T(n)$-copy algorithm by $\mathcal{B}$. We have $\mathcal{B}(|\omega\rangle^{\otimes T(n)}) = \rho_A(|\omega\rangle)^{\otimes T(n)}$, where $\mathcal{B}$ runs in polynomial time (recalling that $T(n)$ is polynomial in n). Then we can write (9.111) as

$$\left| \Pr_{k\leftarrow\mathcal{K}}\big[(\widetilde{\mathcal{A}} \circ \mathcal{B})(|\phi_k\rangle^{\otimes T(n)}) = 1\big] - \Pr_{|\psi\rangle\leftarrow\mathrm{Haar}}\big[(\widetilde{\mathcal{A}} \circ \mathcal{B})(|\psi\rangle^{\otimes T(n)}) = 1\big]\right| . \tag{9.112}$$

Since $\widetilde{\mathcal{A}} \circ \mathcal{B}$ is itself a polynomial-time quantum algorithm, Definition 15 tells us that (9.112) above is upper bounded by $\mathrm{negl}(n)$. So in summary, we find

$$\left| \Pr_{k\leftarrow\mathcal{K}}\big[\mathcal{A}(\rho_A(|\phi_k\rangle)) = 1\big] - \Pr_{|\psi\rangle\leftarrow\mathrm{Haar}}\big[\mathcal{A}(\rho_A(|\psi\rangle)) = 1\big]\right| \leq \mathrm{negl}(n) . \tag{9.113}$$

In an identical manner we can show that the third term on the right-hand side of (9.108) satisfies the bound

$$\left| \Pr_{k\leftarrow\mathcal{K}}\big[\mathcal{A}(\rho_B(|\phi_k\rangle)) = 1\big] - \Pr_{|\psi\rangle\leftarrow\mathrm{Haar}}\big[\mathcal{A}(\rho_B(|\psi\rangle)) = 1\big]\right| \leq \mathrm{negl}(n) . \tag{9.114}$$

Putting together the inequalities in (9.108), (9.109), (9.113) and (9.114), as well as observing that $T(n)$ is at most polynomially large in n, we achieve the desired bound. □

The above Theorem immediately implies that the exponential advantage for quantum principal component analysis in Theorem 49 has a counterpart for pseudorandom states. We note that in the pseudorandom context the advantage is not strictly exponential; rather, the advantage holds for arbitrary *polynomial*-time quantum learning algorithms in the quantum-enhanced scenario versus in the conventional scenario.

9.7 Quantum advantage in testing the purity of a quantum state

An exponential lower bound for conventional experiments

In this subsection, we provide an exponential lower bound for testing if a quantum state is pure or maximally mixed. In this section, we will denote $d = 2^n$ to be the Hilbert space dimension.

Theorem 52 (Purity testing lower bound)**.** *Any learning algorithm without quantum memory requires*

$$T \geq \Omega\left(2^{n/2}\right) \tag{9.115}$$

copies of $\rho \in \mathbb{H}^{2^n \times 2^n}$ *to distinguish between whether* ρ *is a pure state or a maximally mixed state with probability at least* $2/3$.

Proof. Let $\mathcal{T}$ be the tree corresponding to any given learning algorithm for this distinguishing task. By Lemma 4 it suffices to lower bound $\mathbb{E}[v] * p^{|v\rangle\langle v|}(\ell)/p^{\rho_{\mathsf{mm}}}(\ell)$ for all leaves ℓ. If $\{e_{u_t,s_t}\}_{t=1}^T$ are the edges on the path from root to the leaf ℓ, then

$$\mathbb{E}[v] * \frac{p^{|v\rangle\langle v|}(\ell)}{p^{\rho_{\mathsf{mm}}}(\ell)} = \mathbb{E}[v] * \prod_{t=1}^{T} d\,\langle \psi_{s_t}^{u_t} | v \rangle^2 \tag{9.116}$$

$$= \frac{d^T}{d(d+1)\cdots(d+T-1)} \cdot \sum_{\pi \in \mathcal{S}_T} \operatorname{tr}\left(\pi \bigotimes_{t=1}^{T} |\psi_{s_t}^{u_t}\rangle\langle\psi_{s_t}^{u_t}|\right). \tag{9.117}$$

The second equality follows from Lemma 53 where π is the permutation operator. By Lemma 54 below, the sum in (9.117) is lower bounded by 1, so

$$\mathbb{E}[v] * \frac{p^{|v\rangle\langle v|}(\ell)}{p^{\rho_{\mathsf{mm}}}(\ell)} \geq \frac{d^T}{d(d+1)\cdots(d+T-1)} \geq \prod_{t=0}^{T-1}\left(1 - \frac{t}{d}\right)^{-1} \geq \left(1 - \frac{T}{d}\right)^T. \tag{9.118}$$

Using Lemma 4, we have the probability that the given learning algorithm successfully distinguishes the two settings is upper bounded by $1 - \left(1 - \frac{T}{d}\right)^T$. Therefore, $2/3 \leq 1 - \left(1 - \frac{T}{d}\right)^T$ implying that $T \geq \Omega(\sqrt{d})$. □

Lemma 53 (Haar integration over states, see e.g. (Aram W. Harrow, 2013))**.** *Consider the uniform (Haar) measure over n-qubit pure states $|\psi\rangle$, then*

$$\mathop{\mathbb{E}}_{|\psi\rangle}[|\psi\rangle\langle\psi|^{\otimes T}] = \binom{2^n+T-1}{T}^{-1}\sum_{\pi\in S_T}\pi, \tag{9.119}$$

where π is a permutation operator on a tensor product space of T n-qubit pure states and S_T is the symmetric group of degree T.

The key step in the above proof is the following technical lemma that lower bounds the norm of the projection of any tensor product of pure states to the symmetric subspace.

Lemma 54. *For any collection of pure states $|\psi_1\rangle,\ldots,|\psi_T\rangle \in \mathbb{H}^d$,*

$$\sum_{\pi\in\mathcal{S}_T}\operatorname{tr}\left(\pi\bigotimes_{t=1}^{T}|\psi_t\rangle\langle\psi_t|\right)\geq 1. \tag{9.120}$$

Proof. Let Π denote the projector to the symmetric subspace in $(\mathbb{C}^{2^n})^{\otimes T}$. Note that (9.120) is equivalent to the statement that $\operatorname{tr}\left(\Pi\bigotimes_{t=1}^{T}|\psi_t\rangle\langle\psi_t|\right)\geq 1/T!$. This is clearly true for $T=1$; we proceed by induction on T. Let $\widetilde{\Pi}$ denote the projector to the symmetric subspace in $(\mathbb{C}^{2^n})^{\otimes T-1}$, and define the (unnormalized) state $|\widetilde{\psi}\rangle \triangleq \widetilde{\Pi}\bigotimes_{t=2}^{T}|\psi_t\rangle$.

As $\widetilde{\Pi}$ is a projector, we have

$$\langle\widetilde{\psi}|\widetilde{\psi}\rangle = \left\langle\bigotimes_{t=2}^{T}\psi_t\widetilde{\Pi}\middle|\widetilde{\Pi}\bigotimes_{t=2}^{T}\psi_t\right\rangle = \operatorname{tr}\left(\widetilde{\Pi}\bigotimes_{t=2}^{T}|\psi_t\rangle\langle\psi_t|\right)\geq\frac{1}{(T-1)!}, \tag{9.121}$$

where the last step follows by the inductive hypothesis.

We can rewrite the left-hand side of (9.120) as $\sum_{\pi\in\mathcal{S}_T}\operatorname{tr}\left(\pi|\psi_1\rangle\langle\psi_1|\otimes|\widetilde{\psi}\rangle\langle\widetilde{\psi}|\right)$ and decompose this sum into π for which $\pi(1)=1$ and all other π. Note that

$$\sum_{\pi\in\mathcal{S}_T:\pi(1)=1}\operatorname{tr}\left(\pi|\psi_1\rangle\langle\psi_1|\otimes|\widetilde{\psi}\rangle\langle\widetilde{\psi}|\right)=\sum_{\widetilde{\pi}\in\mathcal{S}_{T-1}}\operatorname{tr}\left(\widetilde{\pi}|\widetilde{\psi}\rangle\langle\widetilde{\psi}|\right)=(T-1)!\,\langle\widetilde{\psi}|\widetilde{\psi}\rangle\geq 1. \tag{9.122}$$

It remains to argue that $\sum_{\pi\in\mathcal{S}_T:\pi(1)\neq 1}\operatorname{tr}\left(\pi|\psi_1\rangle\langle\psi_1|\otimes|\widetilde{\psi}\rangle\langle\widetilde{\psi}|\right)\geq 0$.

Consider the map which sends any $\pi\in\mathcal{S}_T$ for which $\pi(1)\neq 1$ to $\breve{\pi}\in\mathcal{S}_{T-1}$ defined as follows. For any $2\leq i\leq T$, for which $\pi(i)\neq 1$, $\breve{\pi}(i-1)\triangleq\pi(i)-1$ and for $i=\pi^{-1}(1)>1$, $\breve{\pi}(i-1)\triangleq\pi(1)$. Then for any $\pi\in\mathcal{S}_T$,

$$\operatorname{tr}\left(\pi|\psi_1\rangle\langle\psi_1|\otimes|\widetilde{\psi}\rangle\langle\widetilde{\psi}|\right) \tag{9.123}$$

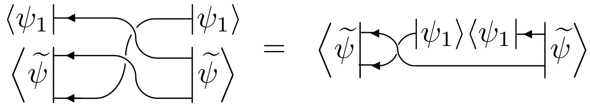

$\langle\psi_1|$
$|\psi_1\rangle$
$\langle\widetilde{\psi}|$
$|\widetilde{\psi}\rangle$
=
$\langle\widetilde{\psi}|$
$|\psi_1\rangle\langle\psi_1|$
$|\widetilde{\psi}\rangle$

Figure 9.12: Illustration of the equality (9.124) for $T = 3$, $\pi = (123)$. The corresponding permutation $\breve{\pi} = (12)$ can be seen on the right-hand side.

$$= \operatorname{tr}\left(\breve{\pi}(\underbrace{\mathrm{Id}\otimes\cdots\otimes\mathrm{Id}}_{\pi(1)-2}\otimes|\psi_1\rangle\langle\psi_1|\otimes\underbrace{\mathrm{Id}\otimes\cdots\otimes\mathrm{Id}}_{T-\pi(1)})|\widetilde{\psi}\rangle\langle\widetilde{\psi}|\right) \tag{9.124}$$

$$= \operatorname{tr}\left((\mathrm{Id}\otimes\cdots\otimes\mathrm{Id}\otimes|\psi_1\rangle\langle\psi_1|\otimes\mathrm{Id}\otimes\cdots\otimes\mathrm{Id})|\widetilde{\psi}\rangle\langle\widetilde{\psi}|P^\dagger_{\breve{\pi}}\right) \tag{9.125}$$

$$= \operatorname{tr}\left((\mathrm{Id}\otimes\cdots\otimes\mathrm{Id}\otimes|\psi_1\rangle\langle\psi_1|\otimes\mathrm{Id}\otimes\cdots\otimes\mathrm{Id})|\widetilde{\psi}\rangle\langle\widetilde{\psi}|\right) \geq 0, \tag{9.126}$$

as claimed, where the first step (9.124) is illustrated in Figure 9.12. □

A constant upper bound for quantum-enhanced experiments

Here we give a simple algorithm for the above distinguishing task that matches the lower bound in Theorem 52 up to constant factors.

Theorem 53. *There is a learning algorithm without quantum memory which takes $T = O(2^{n/2})$ copies of ρ to distinguish between whether ρ is a pure state or maximally mixed.*

To prove this, we will use the following well-known result from classical distribution testing:

Theorem 54 ((Chan et al., 2014; Diakonikolas, Kane, and Nikishkin, 2014; Canonne et al., 2018))**.** *Given $0 < \epsilon < 1$ and sample access to a distribution q over $[d]$, there is an algorithm TestUniformityL2(q, d, ϵ) that uses $T = O(\sqrt{d}/\epsilon^2)$ samples from q and with probability $9/10$ distinguishes whether q is the uniform distribution over $[d]$ or $\epsilon/\sqrt{d}$-far in L_2 distance from the uniform distribution.*

We will also need the following standard moment calculation:

Lemma 55 (Lemma 6.4 in (Sitan Chen, J. Li, and O'Donnell, 2021))**.** *For Haar-random $\mathbf{U} \in U(2^n)$ and $\rho \in \mathbb{H}^{2^n\times 2^n}$, let Z denote $\sum_{i=1}^{2^n}\left(\langle i|\,\mathbf{U}^\dagger\mathbf{M}\mathbf{U}\,|i\rangle\right)^2$. Then*

$$\mathbb{E}\,Z = \frac{1}{2^n+1}\left(\operatorname{tr}(\mathbf{M})^2 + \|\mathbf{M}\|^2_{\mathrm{HS}}\right). \tag{9.127}$$

If in addition we have that $\operatorname{tr}(\mathbf{M}) = 0$*, then*

$$\mathbb{E}\,Z^2 \leq \frac{1+o(1)}{4^n}\,\|\mathbf{M}\|^4_{\mathrm{HS}}. \tag{9.128}$$

We are now ready to prove Theorem 53.

Proof of Theorem 53. Sample a Haar-random basis $\{\mathbf{U}\,|i\rangle\}_{i\in[2^n]}$ and measure every copy of ρ in this basis. If ρ is maximally mixed, note that the distribution over outcomes from a single measurement is the uniform distribution u over $[2^n]$. On the other hand, if ρ is a pure state, let Z denote the random variable $\left\|q^{\mathbf{U}} - u\right\|_2^2$, where $q^{\mathbf{U}}$ is the distribution over outcomes from a single measurement. Note that Z is precisely the random variable Z defined in Lemma 55 for $\mathbf{M} = \rho - \rho_{\mathsf{mm}}$, so we conclude that $\mathbb{E}\,Z = \frac{1}{2^n+1} \cdot \|\rho - \rho_{\mathsf{mm}}\|_{\mathrm{HS}}^2$ and $\mathbb{E}\,Z^2 \le \frac{1+o(1)}{16^n} \cdot \|\rho - \rho_{\mathsf{mm}}\|_{\mathrm{HS}}^4$, so by Paley-Zygmund, there is an absolute constant $c > 0$ for which $\Pr|Z| \ge c\,\|\rho - \rho_{\mathsf{mm}}\|_{\mathrm{HS}}^2 \ge 9/10$. Note that $\|\rho - \rho_{\mathsf{mm}}\|_{\mathrm{HS}}^2 = 1 - 1/2^n$, so with probability at least $9/10$ over the randomness of $\mathbf{U}$, $\left\|q^{\mathbf{U}} - u\right\|_2 \ge \Omega(2^{-n/2})$.

So by Theorem 54, TestUniformityL2$(q^{\mathbf{U}}, 2^n, \Theta(2^{-n/2}))$ will take $T = O(2^{n/2})$ samples from q and correctly distinguish whether q is uniform or far from uniform with probability at least $4/5$ over the randomness of the algorithm and of $\mathbf{U}$. □

9.8 Quantum advantage in learning a polynomial-time quantum process

Here we consider the problem of learning a polynomial-time quantum process and provide a rigorous exponential separation between the conventional and quantum-enhanced learning settings.

Problem setting

We consider an unknown quantum process $\mathcal{E}$ on n qubits generated as follows.

- An n-qubit input state σ is accompanied by an m ancillary qubits initialized at $|0^m\rangle\!\langle 0^m|$.
- p unknown two-qubit unitary gates are applied on the $(n+m)$-qubit system $\sigma \otimes |0^m\rangle\!\langle 0^m|$.
- The ancillary qubits are hidden, resulting in an n-qubit mixed state $\mathcal{E}(\sigma)$

When $m, p = \mathrm{poly}(n)$, we refer to $\mathcal{E}$ as a polynomial-time quantum process. Next, we consider an input probability distribution $\mathcal{D}$ over n-qubit mixed states σ. The goal is to learn an approximate model $\tilde{\mathcal{E}}$ of $\mathcal{E}$, such that we can accurately predict the output state on average:

$$\mathop{\mathbb{E}}_{\sigma\sim\mathcal{D}} \left\|\tilde{\mathcal{E}}(\sigma) - \mathcal{E}(\sigma)\right\|_1 \le \epsilon. \tag{9.129}$$

In the above, $\|X\|_1 = \max_{O:\|O\|_\infty \leq 1} |\operatorname{tr}(OX)|$ is the trace norm.

Rigorous statements

We have the following theorem for quantum-enhanced experiments showing that they can efficiently learn a polynomial-time quantum process. We will prove the theorem later in Section 9.8.

Theorem 55 (Approximate learning of quantum processes – polynomial upper bound)**.** *For any distribution $\mathcal{D}$ and any $\epsilon, \delta > 0$, there exists a learning algorithm in the quantum-enhanced setting that can learn an approximate model $\tilde{\mathcal{E}}$ such that with probability at least $1-\delta$,*

$$\mathop{\mathbb{E}}_{\sigma\sim\mathcal{D}} \left\|\tilde{\mathcal{E}}(\sigma) - \mathcal{E}(\sigma)\right\|_1 \leq \epsilon \tag{9.130}$$

from at most $\widetilde{O}(\mathrm{poly}(n)\log(1/\delta)/\epsilon^4)$ accesses to $\mathcal{E}$, where $\widetilde{O}(\cdot)$ hides factors of $\log(1/\epsilon)$.

In contrast, our hardness results for predicting properties of physical states in the conventional setting (see Theorem 47 in Section 9.5) immediately implies the following exponential lower bound.

Corollary 16 (Approximate learning of quantum processes – exponential lower bound)**.** *Let $\mathcal{D}$ be any distribution and $\mathcal{E}$ be the quantum process that always generates a state ρ considered in Def. 14. Any algorithm in the conventional setting that learns an approximate model $\tilde{\mathcal{E}}$ such that*

$$\mathop{\mathbb{E}}_{\sigma\sim\mathcal{D}} \left\|\tilde{\mathcal{E}}(\sigma) - \mathcal{E}(\sigma)\right\|_1 \leq 0.25, \tag{9.131}$$

must use at least $\Omega(2^n)$ accesses to $\mathcal{E}$.

Proof. We consider $m = 2n$. The two-qubit gates swap the input state σ to the first n ancillary qubits. Then we use the rest of the n ancillary qubits and the n system qubits (i.e. the qubits in the input state ρ) to prepare a state ρ considered in Def. 14. To prepare the maximally mixed state $I/2^n$, we entangle each of the system qubits with the corresponding ancillary qubit to prepare a Bell state $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. To prepare the alternative state $(I + 0.9sP)/2^n$, we perform the following procedure.

1. For qubit i, where P_i is *not* the last non-identity Pauli operator (i.e. last in terms of having the largest index i), we entangle the i-th system qubit with the corresponding ancillary qubit to prepare a Bell state $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.

2. For the last remaining qubit i, which corresponds to the index i such that P_i is the last non-identity Pauli operator, we apply a sequence of two-qubit gates entangling qubit i to qubit j with $P_j \neq I$. The sequence of two-qubit gates stores the parity (or $1 -$ parity) of all qubits j with $P_j \neq I$. After this step, when we trace over the ancillary qubits, we have generated $(I + sP^{(Z)})/2^n$, where $P^{(Z)} = \bigotimes_{i=1}^n F(P_i)$ and $F(I) = I, F(X) = Z, F(Y) = I, F(Z) = I$. Then, we rotate the corresponding ancillary qubit for qubit i from $|0\rangle$ to $\sqrt{0.95}\,|0\rangle + \sqrt{0.05}\,|1\rangle$ and apply a controlled-not gate from the ancillary qubit (control) to qubit i. The system qubits are now in the state $(I + 0.9sP^{(Z)})/2^n$.

3. Finally, for each qubit i with $P_i = X$, we rotate the system qubit from $|0\rangle$ to $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and from $|1\rangle$ to $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. For each qubit i with $P_i = Y$, we rotate from 0 to $y+$ and from 1 to $y-$. Tracing over the ancillary qubits, the system qubits are now in the state $(I + 0.9sP)/2^n$.

We can see that the number of gates p is $O(n)$. Furthermore, no matter what the input state σ is, the above quantum process always produces a state ρ considered in Def. 14.

When an algorithm in the conventional setting has learned an approximate model $\tilde{\mathcal{E}}$ with

$$\mathop{\mathbb{E}}_{\sigma\sim\mathcal{D}} \left\|\tilde{\mathcal{E}}(\sigma) - \mathcal{E}(\sigma)\right\|_1 \leq 0.25, \tag{9.132}$$

we can use Jensen's inequality to conclude

$$\left\|\mathop{\mathbb{E}}_{\sigma\sim\mathcal{D}} \tilde{\mathcal{E}}(\sigma) - \rho\right\|_1 \leq \mathop{\mathbb{E}}_{\sigma\sim\mathcal{D}} \left\|\tilde{\mathcal{E}}(\sigma) - \mathcal{E}(\sigma)\right\|_1 \leq 0.25. \tag{9.133}$$

Because $\|X\|_1 = \max_{O:\|O\|_\infty\leq 1} |\operatorname{tr}(OX)|$, the above implies that the algorithm can predict $\operatorname{tr}(O\rho)$ up to an error of 0.25. From Theorem 47, we conclude that the learning algorithm must use at least $\Omega(2^n)$ copies of ρ, which corresponds to $\Omega(2^n)$ accesses to $\mathcal{E}$. □

Proof of polynomial upper bound in Theorem 55

Theorem 55 establishes an upper bound on the number of times we must access the unknown process $\mathcal{E}$ in the quantum-enhanced setting to construct an approximate model of $\mathcal{E}$. Note that the theorem only concerns the number of times we run the process $\mathcal{E}$; it does not address the computational complexity of the learning procedure. Our strategy for proving the theorem is as follows. First we find an upper bound on the number of elements of an ϵ'-covering net for the set of all

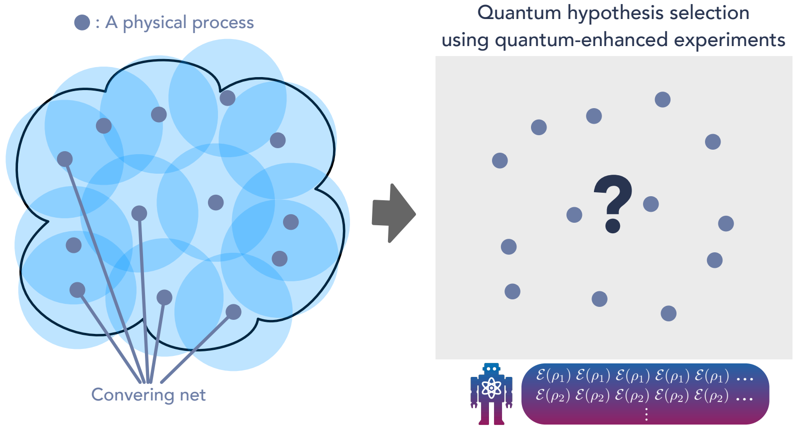

● : A physical process
Convering net
Quantum hypothesis selection
using quantum-enhanced experiments
?
$\mathcal{E}(\rho_1)$ $\mathcal{E}(\rho_1)$ $\mathcal{E}(\rho_1)$ $\mathcal{E}(\rho_1)$ $\mathcal{E}(\rho_1)$ ...
$\mathcal{E}(\rho_2)$ $\mathcal{E}(\rho_2)$ $\mathcal{E}(\rho_2)$ $\mathcal{E}(\rho_2)$ $\mathcal{E}(\rho_2)$ ...
⋮

Figure 9.13: *Illustration for the proof of Theorem 55 on learning polynomial-time quantum processes.* We first form a covering net (all dark blue dots) for the space of all polynomial-time quantum processes (the cloud shape). Any polynomial-time quantum process is close to an element in the covering net. Then we perform quantum hypothesis selection (Bădescu and O'Donnell, 2020) using a quantum dataset stored in the quantum memory to find the approximate physical process.

quantum processes that can be constructed using up to p two-qubit quantum gates, with distance defined by the diamond norm. Next we explain how to use a quantum hypothesis testing algorithm to find a process $\tilde{\mathcal{E}}$ in the covering net that approximates $\mathcal{E}$ as specified in (9.130), if ϵ' is appropriately chosen. This quantum hypothesis testing method can be carried out in the quantum-enhanced setting, but not in the conventional setting. The number of times we must access $\mathcal{E}$ depends on the size of the covering net, and can be shown to scale polynomially with the number of gates p, proving the theorem. An illustration is given in Supp. Fig. 9.13.

Covering net

First, we construct the covering net for the set $\mathcal{S}$ of quantum processes with a fixed n, m, p. An ϵ-covering net of a set $\mathcal{S}$ is a subset $\mathcal{N}_\epsilon \subseteq \mathcal{S}$ such that for every point $x \in \mathcal{S}$, there exists a point $y \in \mathcal{N}$ with $\|x - y\| \leq \epsilon$ in an appropriate norm.

Recall that a unitary U corresponds to a unitary channel $\mathcal{U}$ defined as

$$\mathcal{U}(\rho) = U\rho U^\dagger. \tag{9.134}$$

Because two-qubit unitary channels form a bounded set in a finite-dimensional space, the $\tilde{\epsilon}$-covering net for two-qubit unitary channels has a size of at most

$$\left(\frac{c_1}{\tilde{\epsilon}}\right)^{c_2}, \tag{9.135}$$

where c_1, c_2 are two constants (see, e.g., Section 4.2 in (Vershynin, 2018b)). Here, we consider the norm to be the diamond norm $\|\cdot\|_\diamond$ (see Section 3.3 in (Watrous, 2018)). The bound in (9.135) only pertains to the covering net size when the unitary acts on a *fixed* set of two qubits. Let us now consider two-qubit unitary channels that can act on any two of the $n+m$ qubits. Because there are $\binom{n+m}{2}$ pairs of qubits that the unitary could act on, the size of the $\tilde{\epsilon}$-covering net $\mathcal{N}_{\tilde{\epsilon},n+m}$ of all two-qubit gates on an $(n+m)$-qubit system is upper bounded as follows,

$$\left|\mathcal{N}_{\tilde{\epsilon},n+m}\right| \leq \binom{n+m}{2}\left(\frac{c_1}{\tilde{\epsilon}}\right)^{c_2}. \tag{9.136}$$

To construct an ϵ-covering net for the composed quantum process $\mathcal{E}$, we need to consider $\tilde{\epsilon} = \epsilon'/p$ in $\mathcal{N}_{\tilde{\epsilon},n+m}$. Consider any sequence of two-qubit unitary channels $\mathcal{U}_1, \ldots, \mathcal{U}_p$ on an $(n+m)$-qubit system. For each U_i in the sequence, we find the closest unitary channel $\tilde{\mathcal{U}}_i$ in $\mathcal{N}_{\tilde{\epsilon},n+m}$, hence $\left\|\mathcal{U}_i - \tilde{\mathcal{U}}_i\right\|_\diamond \leq \tilde{\epsilon}$. Then we can use a telescoping sum and the triangle inequality to see that

$$\left\|\mathrm{tr}_{n+1,\ldots,n+m}\left(\mathcal{U}_p \ldots \mathcal{U}_1(\rho \otimes |0^m\rangle\langle 0^m|)\right) - \mathrm{tr}_{n+1,\ldots,n+m}\left(\tilde{\mathcal{U}}_p \ldots \tilde{\mathcal{U}}_1(\rho \otimes |0^m\rangle\langle 0^m|)\right)\right\|_1 \tag{9.137}$$

$$\leq \left\|\mathcal{U}_p \ldots \mathcal{U}_1(\rho \otimes |0^m\rangle\langle 0^m|) - \tilde{\mathcal{U}}_p \ldots \tilde{\mathcal{U}}_1(\rho \otimes |0^m\rangle\langle 0^m|)\right\|_1 \tag{9.138}$$

$$\begin{aligned}\leq &\left\|\mathcal{U}_p \ldots \mathcal{U}_1(\rho \otimes |0^m\rangle\langle 0^m|) - \mathcal{U}_p\tilde{\mathcal{U}}_{p-1} \ldots \tilde{\mathcal{U}}_1(\rho \otimes |0^m\rangle\langle 0^m|)\right\|_1 \\ &+ \left\|\mathcal{U}_p\tilde{\mathcal{U}}_{p-1} \ldots \mathcal{U}_1(\rho \otimes |0^m\rangle\langle 0^m|) - \tilde{\mathcal{U}}_p\tilde{\mathcal{U}}_{p-1} \ldots \tilde{\mathcal{U}}_1(\rho \otimes |0^m\rangle\langle 0^m|)\right\|_1\end{aligned} \tag{9.139}$$

$$\leq \left\|\mathcal{U}_{p-1} \ldots \mathcal{U}_1(\rho \otimes |0^m\rangle\langle 0^m|) - \tilde{\mathcal{U}}_{p-1} \ldots \tilde{\mathcal{U}}_1(\rho \otimes |0^m\rangle\langle 0^m|)\right\|_1 + \left\|\mathcal{U}_p - \tilde{\mathcal{U}}_p\right\|_\diamond \tag{9.140}$$

$$\leq \ldots \tag{9.141}$$

$$\leq \sum_{i=1}^{p} \left\|\mathcal{U}_i - \tilde{\mathcal{U}}_i\right\|_\diamond \tag{9.142}$$

$$\leq p\tilde{\epsilon} = \epsilon'. \tag{9.143}$$

The first inequality uses the fact that taking partial trace does not increase the trace norm. The second inequality uses $\|A-B\| \leq \|A-C\| + \|C-B\|$. The third inequality uses $\|\mathcal{E}(X)\|_1 \leq \|X\|_1$ for any CPTP map $\mathcal{E}$, and $\|\mathcal{E}(\rho)\|_1 \leq \|\mathcal{E}\|_\diamond \|\rho\|_1 = \|\mathcal{E}\|_\diamond$. The fourth inequality considers the same steps taken in the second and third inequality. Then, using induction, we obtain the formula given in the second-to-last line. The last line uses the fact that $\left\|\mathcal{U}_i - \tilde{\mathcal{U}}_i\right\|_\diamond \leq \tilde{\epsilon}$ for all i and $\tilde{\epsilon} = \epsilon'/p$.

From the above analysis, we can see that we can find an ϵ'-covering net $\mathcal{N}_{\epsilon',n,m,p}$ for the space of $\mathcal{E}$ with an n-qubit input state, m ancillary qubits, and p two-qubit gates that satisfies

$$\left|\mathcal{N}_{\epsilon',n,m,p}\right| \leq \left[\binom{n+m}{2}\left(\frac{pc_1}{\epsilon'}\right)^{c_2}\right]^p. \tag{9.144}$$

For any $\mathcal{E}$ in the space, we can find an $\tilde{\mathcal{E}} \in \mathcal{N}_{\epsilon',n,m,p}$ such that for all n-qubit input states ρ we have

$$\left\|\mathcal{E}(\rho) - \tilde{\mathcal{E}}(\rho)\right\|_1 \leq \epsilon'. \tag{9.145}$$

We will then utilize the ϵ-covering net $\mathcal{N}_{\epsilon',n,m,p}$ in the subsequent proof. An ϵ-covering net of quantum processes have also been used in (Huang, Richard Kueng, and Preskill, 2021) to establish an information-theoretic bound on quantum advantage in (Caro, Huang, Cerezo, et al., 2021) to analyze generalization performance of quantum neural networks.

Learning via Hypothesis Selection: Protocol and Analysis

We will sample N_{in} input states $\rho_1, \ldots, \rho_{N_{\text{in}}}$ from the distribution $\mathcal{D}$. For each $i \in [N_{\text{in}}]$ and every $\widetilde{\mathcal{E}}_k \in \mathcal{N}_{\epsilon',n,m,p}$ (for ϵ' to be tuned later), we will access the true process $\mathcal{E}$ a number of times N_{out} using ρ_i as the input state, obtaining N_{out} copies of $\mathcal{E}(\rho_i)$. We will store these $N_{\text{in}} \cdot N_{\text{out}}$ states in the quantum memory and run a known algorithm for *quantum hypothesis selection* (Bădescu and O'Donnell, 2020) to determine for which k the product state $\bigotimes_{i=1}^{N_{\text{in}}} \widetilde{\mathcal{E}}_k(\rho_i)$ is approximately closest to $\bigotimes_{i=1}^{N_{\text{in}}} \mathcal{E}(\rho_i)$. We will argue that if ϵ' is sufficiently small and N_{in} sufficiently large, then the index k that we find will satisfy

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}}\left[\left\|\mathcal{E}(\rho) - \widetilde{\mathcal{E}}_k(\rho)\right\|_1\right] \leq \epsilon, \tag{9.146}$$

as desired.

We now proceed to the analysis of this protocol. We begin with an estimate for the distance between product states whose components are pairwise far from each other.

Lemma 56. *If $\rho_1, \ldots, \rho_N$ and $\rho'_1, \ldots, \rho'_N$ satisfy $\frac{1}{N}\sum_{i=1}^N \left\|\rho_i - \rho'_i\right\|_1 \geq \epsilon$, then*

$$\left\|\bigotimes_i \rho_i - \bigotimes_i \rho'_i\right\|_1 \geq 2\left(1 - (1-\epsilon^2/4)^{N/2}\right). \tag{9.147}$$

Proof. For convenience, denote $\epsilon_i := \left\|\rho_i - \rho_i'\right\|_1$. We have that $F(\rho_i, \rho_i') \leq 1 - \epsilon_i^2/4$, so

$$\left\|\bigotimes_i \rho_i - \bigotimes_i \rho_i'\right\|_1 \geq 2\left(1 - \sqrt{F\left(\bigotimes_i \rho_i, \bigotimes_i \rho_i'\right)}\right) \tag{9.148}$$

$$\geq 2\left(1 - \sqrt{\prod_{i=1}^{N}(1 - \epsilon_i^2/4)}\right) \tag{9.149}$$

$$\geq 2\left(1 - \left(\frac{1}{N}\sum_i (1 - \epsilon_i^2/4)\right)^{N/2}\right) \tag{9.150}$$

$$= 2\left(1 - (1 - \epsilon^2/4)^{N/2}\right) \tag{9.151}$$

where the first step follows by the standard inequality $\|\rho - \rho'\|_1 \geq 2\sqrt{1 - F(\rho, \rho')}$, the second step follows by tensorization of fidelity, the third step follows by AM-GM and the fact that $\epsilon_i = \left\|\rho_i - \rho_i'\right\|_1 \leq 2$, and the last step follows from the assumption. □

Next, we elaborate on how to select N_{in}. Consider any $\widetilde{\mathcal{E}} \in \mathcal{N}_{\epsilon',n,m,p}$. By Hoeffding's inequality, because $\rho_1, \ldots, \rho_{N_{\text{in}}}$ are sampled independently and identically distribued from the distribution $\mathcal{D}$, and $\|\rho - \rho'\| \leq 2$ for all density matrices ρ, ρ' we have

$$\left|\frac{1}{N_{\text{in}}}\sum_{i=1}^{N_{\text{in}}}\left\|\widetilde{\mathcal{E}}(\rho_i) - \mathcal{E}(\rho_i)\right\|_1 - \mathop{\mathbb{E}}_{\rho\sim\mathcal{D}}\left\|\widetilde{\mathcal{E}}(\rho) - \mathcal{E}(\rho)\right\|_1\right| \leq \epsilon/2 \tag{9.152}$$

with probability at least $1 - \delta'$ provided $N_{\text{in}} = \Omega(\log(1/\delta')/\epsilon^2)$. By a union bound over $\mathcal{N}_{\epsilon',n,m,p}$, the above bound holds simultaneously for all $\widetilde{\mathcal{E}} \in \mathcal{N}_{\epsilon',n,m,p}$ with probability at least $1 - |\mathcal{N}_{\epsilon',n,m,p}|\delta'$. Henceforth, we condition on this event holding.

The following shows that it suffices to find $\widetilde{\mathcal{E}} \in \mathcal{N}_{\epsilon',n,m,p}$ for which the product state $\bigotimes_{i=1}^{N_{\text{in}}} \widetilde{\mathcal{E}}(\rho_i)$ is sufficiently close to $\bigotimes_{i=1}^{N_{\text{in}}} \mathcal{E}(\rho_i)$.

Lemma 57. *If* (9.152) *holds for all* $\widetilde{\mathcal{E}} \in \mathcal{N}_{\epsilon',n,m,p}$, *then if*

$$\left\|\bigotimes_{i=1}^{N_{\text{in}}} \widetilde{\mathcal{E}}(\rho_i) - \bigotimes_{i=1}^{N_{\text{in}}} \mathcal{E}(\rho_i)\right\|_1 \leq 2\left(1 - (1 - \epsilon^2/16)^{N_{\text{in}}/2}\right) \tag{9.153}$$

for some $\widetilde{\mathcal{E}} \in \mathcal{N}_{\epsilon',n,m,p}$, *we have that* $\mathbb{E}_{\rho\sim\mathcal{D}}\left\|\widetilde{\mathcal{E}}(\rho) - \mathcal{E}(\rho)\right\|_1 \leq \epsilon$.

Proof. We prove the contrapositive. Suppose $\mathbb{E}_{\rho\sim\mathcal{D}}\left\|\widetilde{\mathcal{E}}(\rho)-\mathcal{E}(\rho)\right\|_1 > \epsilon$. Then by (9.152), we have

$$\frac{1}{N_{\text{in}}}\sum_{i=1}^{N_{\text{in}}}\left\|\widetilde{\mathcal{E}}(\rho_i)-\mathcal{E}(\rho_i)\right\|_1 \geq \epsilon/2. \tag{9.154}$$

The lemma follows from Lemma 56. □

As we show in the next lemma, there exists an $\widetilde{\mathcal{E}} \in \mathcal{N}_{\epsilon',n,m,p}$, namely the process in the covering net which is closest to $\mathcal{E}$, for which (9.153) holds, provided ϵ' is sufficiently small.

Lemma 58. *For any $\epsilon' > 0$, there exists an $\widetilde{\mathcal{E}} \in \mathcal{N}_{\epsilon',n,m,p}$ for which*

$$\left\|\bigotimes_{i=1}^{N_{\text{in}}}\widetilde{\mathcal{E}}(\rho_i)-\bigotimes_{i=1}^{N_{\text{in}}}\mathcal{E}(\rho_i)\right\|_1 \leq N_{\text{in}}\epsilon'. \tag{9.155}$$

Proof. Take $\widetilde{\mathcal{E}} \in \mathcal{N}_{\epsilon',n,m,p}$ satisfying $\left\|\widetilde{\mathcal{E}}(\rho)-\mathcal{E}(\rho)\right\|_1 \leq \epsilon'$ for all ρ. For convenience, let $\sigma_i := \mathcal{E}(\rho_i)$, $\sigma_i' := \widetilde{\mathcal{E}}(\rho_i)$, and $\delta_i := \sigma_i' - \sigma_i$ for all $i \in \{1,\ldots,N_{\text{in}}\}$. Then

$$\bigotimes_{i=1}^{N_{\text{in}}}\sigma_i' - \bigotimes_{i=1}^{N_{\text{in}}}\sigma_i = \sum_{i=1}^{N_{\text{in}}}\left[\left(\bigotimes_{j=1}^{i-1}\sigma_j'\right)\otimes(\sigma_i'-\sigma_i)\otimes\left(\bigotimes_{j=i+1}^{N_{\text{in}}}\sigma_j\right)\right], \tag{9.156}$$

so by the triangle inequality we conclude that $\left\|\bigotimes_i\sigma_i' - \bigotimes_i\sigma_i\right\|_1 \leq N_{\text{in}}\epsilon'$ as claimed. □

Lemma 57 and Lemma 58 guarantee the existence of a process in $\mathcal{N}_{\epsilon',n,m,p}$ satisfying the desired bound of (9.146). To complete the proof of Theorem 55, we will use the following special case of a result from (Bădescu and O'Donnell, 2020) to find a process in the covering net which performs comparably.

Theorem 56 ((Bădescu and O'Donnell, 2020), Theorem 1.5)**.** *Suppose we are given m fixed hypothesis states $\sigma_1,\ldots,\sigma_M \in \mathbb{C}^{d\times d}$, parameters $0 < \epsilon, \delta < 1/2$, and access to copies of a state $\rho \in \mathbb{C}^{d\times d}$. Then there is an algorithm that uses*

$$N = O\left(\frac{1}{\epsilon^2}\left(\log^3 M + \alpha\log M\right)\cdot\alpha\right) \tag{9.157}$$

copies of ρ for $\alpha := \log(\log(1/\eta)/\delta)$ and $\eta := \min_i\|\rho-\sigma_i\|_1$ such that with probability at least $1-\delta$ the algorithm outputs a $k \in \{1,\ldots,M\}$ for which $\|\rho-\sigma_k\|_1 \leq 4\eta$.

We can now put together the ingredients assembled in this section to complete the proof of Theorem 55.

Proof of Theorem 55. We first prove the theorem for constant δ. Take $\epsilon' = c/N_{\text{in}}$ for a sufficiently small constant $c > 0, \delta' = 1/(10|\mathcal{N}_{\epsilon',n,m,p}|)$, and $N_{\text{in}} = \widetilde{O}(p/\epsilon^2)$, where $\widetilde{O}(\cdot)$ hides factors of $\log n, \log m, \log p, \log 1/\epsilon$, so that $N_{\text{in}} \geq \Omega(\log(1/\delta')/\epsilon^2)$ and (9.152) holds for all $\widetilde{\mathcal{E}} \in \mathcal{N}_{\epsilon',n,m,p}$ with probability at least 4/5. Note that for some absolute constant $C > 0$,

$$1 - (1-\epsilon^2/16)^{N_{\text{in}}/2} \geq 1 - e^{-N_{\text{in}}\epsilon^2/32} = 1 - \delta'^C \geq \Omega(1)\,. \tag{9.158}$$

In contrast, by Lemma 58 and our choice of ϵ', there exists some $\widetilde{\mathcal{E}} \in \mathcal{N}_{\epsilon',n,m,p}$ for which

$$\left\| \bigotimes_{i=1}^{N_{\text{in}}} \widetilde{\mathcal{E}}(\rho_i) - \bigotimes_{i=1}^{N_{\text{in}}} \mathcal{E}(\rho_i) \right\|_1 \leq c\,. \tag{9.159}$$

Applying Theorem 56 to $\sigma_k = \bigotimes_i \widetilde{\mathcal{E}}_k(\rho_i)$ where $\widetilde{\mathcal{E}}_k$ is the k-th element of $\mathcal{N}_{\epsilon',n,m,p}$, using N_{out} copies of $\rho = \bigotimes_i \mathcal{E}(\rho_i)$ where

$$N_{\text{out}} = O\left(\frac{1}{\epsilon^2}\left(\log^3|\mathcal{N}_{\epsilon',n,m,p}| + \alpha\log|\mathcal{N}_{\epsilon',n,m,p}|\right)\cdot\alpha\right), \quad \alpha := O(\log\log(1/c)), \tag{9.160}$$

we can output a k for which $\|\sigma_k - \rho\|_1 \leq 4c$ with probability 4/5. By taking the constant c sufficiently small, we can leverage (9.158) and Lemma 57 to conclude that

$$\mathop{\mathbb{E}}_{\rho\sim\mathcal{D}}\left[\left\|\mathcal{E}(\rho) - \widetilde{\mathcal{E}}_k(\rho)\right\|_1\right] \leq \epsilon\,. \tag{9.161}$$

Note that N_{out} in (9.160) is dominated by the $\epsilon^{-2}\log|\mathcal{N}_{\epsilon',n,m,p}|$ term since $\alpha = O(1)$, and so recalling (9.144) and our choice of $\epsilon' = c/N_{\text{in}}$ we obtain

$$N_{\text{out}} = O\left(\frac{p^3}{\epsilon^2}\log^3((n+m)pN_{\text{in}})\right) = O\left(\frac{p^3}{\epsilon^2}\log^3((n+m)pN_{\text{in}})\right) = \widetilde{O}\left(\frac{p^3}{\epsilon^2}\right), \tag{9.162}$$

where again $\widetilde{O}(\cdot)$ hides logarithmic factors in $n, m, p, 1/\epsilon$.

As we require N_{out} copies of $\bigotimes_{i=1}^{N_{\text{in}}}\mathcal{E}(\rho_i)$, we must make $N_{\text{out}}\cdot N_{\text{in}} = \widetilde{O}(p^4/\epsilon^4)$ accesses to $\mathcal{E}$. By union bounding over (9.152) holding for all $\widetilde{\mathcal{E}} \in \mathcal{N}_{\epsilon',n,m,p}$ and over the success of the algorithm in Theorem 56, we obtain Theorem 55 for $\delta = 2/5$ from the assumption that $p = \text{poly}(n)$.

We now describe how to extend this result to general δ by a standard clustering argument. We can run $r := \Theta(\log(1/\delta))$ independent copies of the above protocol, resulting in indices $k_1, \ldots, k_r$ into $\mathcal{N}_{\epsilon',n,m,p}$ such that for any fixed $i \in [r]$,

$\left\|\sigma_{k_i} - \rho\right\|_1 \leq 4c$ with probability at least $3/5$. If $S \subset [r]$ denotes the set of $i \in [r]$ for which $\left\|\sigma_{k_i} - \rho\right\|_1 \leq 4c$, then by a Chernoff bound, $|S| \geq r/2$ with probability at least $1 - \delta$ provided the constant factor in the definition of r is sufficiently large. Condition on the event that $|S| \geq r/2$.

Let k be an index into $\mathcal{N}_{\epsilon',n,m,p}$ for which there are at least $r/2$ indices $i \in [r]$ for which $\left\|\sigma_k - \sigma_{k_i}\right\|_1 \leq 8c$, and output the channel $\widetilde{\mathcal{E}}_k$. Such a k certainly exists: take any $i \in S$ and note that by triangle inequality, for any other $j \in S$ we have

$$\left\|\sigma_{k_i} - \sigma_{k_j}\right\|_1 \leq \left\|\sigma_{k_i} - \rho\right\|_1 + \left\|\sigma_{k_j} - \rho\right\|_1 \leq 8c. \tag{9.163}$$

Now observe that regardless of which k we choose that meets the criterion that at least $r/2$ indices $i \in [r]$ satisfy $\left\|\sigma_k - \sigma_{k_i}\right\|_1 \leq 8c$, we must have

$$\|\sigma_k - \rho\|_1 \leq 12c. \tag{9.164}$$

Indeed, suppose to the contrary. Then for any $i \in S$,

$$\left\|\sigma_k - \sigma_{k_i}\right\|_1 \geq \|\sigma_k - \rho\|_1 - \left\|\sigma_{k_i} - \rho\right\|_1 > 12c - 4c = 8c, \tag{9.165}$$

where the second step is by the definition of S and the assumption that (9.164) does not hold. As $|S| \geq r/2$, this yields a contradiction of the fact that there are at least $r/2$ indices $i \in [r]$ for which $\left\|\sigma_k - \sigma_{k_i}\right\|_1 \leq 8c$.

If we take the constant c sufficiently small, then (9.164) together with (9.158) and Lemma 57 allow us to conclude that $\mathbb{E}_{\rho\sim\mathcal{D}}\left[\left\|\mathcal{E}(\rho) - \widetilde{\mathcal{E}}_k(\rho)\right\|_1\right] \leq \epsilon$. As we ran $r := \Theta(\log(1/\delta))$ independent copies of the protocol that we used for constant failure probability δ, we merely incur an additional $\Theta(\log(1/\delta))$ multiplicative overhead in the number of access to $\mathcal{E}$ we must make for general failure probability δ. □

9.9 Quantum advantage in testing properties of quantum channels

Prerequisites

We begin by generalizing the tree representation for learning quantum states to the setting of learning quantum channels. First let us state the idea of the definition intuitively before delving into its technical description. There is some quantum channel C which we wish to learn about; we have the ability to apply the channel to a state of our choice and then to completely measure the resulting state. The resulting measurement outcome can be recorded in a classical memory. The procedure of preparing a state, applying the channel, and then making a measurement is repeated over multiple rounds, wherein the measurement outcomes of previous rounds can inform the states prepared in future rounds, as well as the choice of measurement in future rounds. That is, the protocol is adaptive. At the end, we have gained a list of measurement outcomes with which we can judiciously infer properties of the channel C under investigation.

Now we provide the full technical definition:

Definition 17 (Tree representation for learning channels)**.** *Consider a fixed quantum channel C acting on an n-qubit subsystem of a Hilbert space $\mathcal{H} \simeq \mathcal{H}_{main} \otimes \mathcal{H}_{aux}$ where $\mathcal{H}_{main} \simeq (\mathbb{C}^2)^{\otimes n}$ is the 'main system' comprising n qubits and $\mathcal{H}_{aux} \simeq (\mathbb{C}^2)^{\otimes n'}$ is an 'auxiliary system' of n' qubits. It is convenient to define $d = 2^n$ and $d' = 2^{n'}$. A learning algorithm without quantum memory can be represented as a rooted tree $\mathcal{T}$ of depth T such that each node encodes all measurement outcomes the algorithm has received thus far. The tree has the following properties:*

- *Each node u has an associated probability $p^C(u)$.*
- *The root of the tree r has an associated probability $p^C(r) = 1$.*
- *At each non-leaf node u, we prepare a state $|\phi_u\rangle$ on $\mathcal{H}$, apply the channel C onto the n-qubit subsystem, and measure a rank-1 POVM $\{\sqrt{w_s^u dd'}\, |\psi_s^u\rangle\langle\psi_s^u|\}_s$ (which can depend on u) on the entire system to obtain a classical outcome s. Each child node v of the node u corresponds to a particular POVM outcome s and is connected by the edge $e_{u,s}$. We refer to the set of child node of the node u as $\mathrm{child}(u)$. Accordingly, we can relabel the POVM as $\{\sqrt{w_v dd'}\, |\psi_v\rangle\langle\psi_v|\}_{v\in\mathrm{child}(u)}$.*
- *If v is a child node of u, then*

$$p^C(v) = p^C(u)\, w_v dd'\, \langle\psi_v|\, (C \otimes \mathcal{I}_{aux})[|\phi_u\rangle\langle\phi_u|]\, |\psi_v\rangle\,. \tag{9.166}$$

- *Each root-to-leaf path is of length T. For a leaf of corresponding to node ℓ, $p^C(\ell)$ is the probability that the classical memory is in state ℓ after the learning procedure.*

Each node u in the tree represents the state of the classical memory at one time step of the learning process. The associated probability $p^C(u)$ for a node u is the probability that the classical memory enters the state u during the learning process.

Each time we perform one experiment, we transition from a node u to a child node of u.

There are several features of the definition which we will remark on. First and foremost, an important feature of the definition is that we have access to an auxiliary Hilbert space $\mathcal{H}_{\text{aux}}$ for each state preparation and measurement. In particular, even though the channel C only acts on n qubits, we can apply it to the first n qubits of a state $|\phi\rangle \in \mathcal{H}_{\text{main}} \otimes \mathcal{H}_{\text{aux}}$ which can be entangled between the n qubits and the auxiliary system. Moreover, we can measure the resulting state $(C \otimes \mathcal{I}_{\text{aux}})[|\phi\rangle\langle\phi|]$ using POVM's which are entangled between the n qubits and the auxiliary system. The presence of the auxiliary system will render our proofs somewhat elaborate; moreover, the presence of the auxiliary system renders our results stronger than previous ones for adaptive incoherent access QUALMs (Aharonov, J. S. Cotler, and Qi, 2021). In this particular QUALM setting, the notion of learning algorithm is similar, except that there is no auxiliary system.

We consider quantum channel learning tasks which were first studied in the QUALM setting without an auxiliary Hilbert space (Aharonov, J. S. Cotler, and Qi, 2021). They are:

Definition 18 (Fixed unitary task)**.** *Suppose that an n-qubit quantum channel C is one of the following with equal probability:*

- *C is the completely depolarizing channel $\mathcal{D}$.*
- *C is the unitary channel $C[\rho] = U\rho U^\dagger$ for U a fixed, Haar-random unitary.*

The fixed unitary task is to distinguish between the two above possibilities. We can also consider analogous versions of the problem where U is instead a Haar-random orthogonal matrix, or a Haar-random symplectic matrix.

Note that instead of considering the completely depolarizing channel $\mathcal{D}$ can be thought of in a different way which makes the fixed unitary task more illuminating. Specifically, we can equivalently think of $\mathcal{D}$ as an n-qubit unitary channel which applies an i.i.d. random Haar unitary each time the channel is applied. From this perspective, $\mathcal{D}$ implements time-dependent random unitary dynamics (i.e. a new unitary is selected for each application of the channel); the task is then to distinguish this from time-*independent* random unitary dynamics wherein the channel applies a single fixed random unitary. Said more simply, from this point of view the task is to distinguish a type of time-dependent dynamics from a type of time-independent dynamics.

We also consider another task from (Aharonov, J. S. Cotler, and Qi, 2021) with a slightly different flavor:

Definition 19 (Symmetry distinction task)**.** *Suppose that an n-qubit quantum channel C is one of the following with equal probability:*

- $C[\rho] = U\rho U^\dagger$ *for U a fixed, Haar-random unitary matrix.*
- $C[\rho] = O\rho O^\dagger$ *for O a fixed, Haar-random orthogonal matrix.*
- $C[\rho] = S\rho S^\dagger$ *for S a fixed, Haar-random symplectic matrix.*

The symmetry distinction task is to distinguish between the three above possibilities.

Unitary, orthogonal, and symplectic matrices manifest three different forms of what is called time-reversal symmetry (Dyson, 1962). In this terminology, the symmetry distinction task is to determine the time-reversal symmetry class of C. The task belongs to a class of problems of determining the symmetries of an uncharacterized system, which are important in experimental physics.

In the above distinguishing tasks, we will always reduce them to two-hypothesis distinguishing problem. We define a two-hypothesis distinguishing problem as follows.

Definition 20 (Two-hypothesis channel distinction task)**.** *The following two events happen with equal probability:*

- *The channel C is sampled from a probability distribution D_A over channels.*
- *The channel C is sampled from a probability distribution D_B over channels.*

The goal is to distinguish whether C is sampled from D_A or D_B.

For any two-hypothesis distinguishing problem, we can always apply the two-point method similar to Lemma 3 in learning quantum states.

Lemma 59 (Le Cam's two-point method, see e.g. Lemma 1 in (B. Yu, 1997))**.** *Consider a learning algorithm without quantum memory that is described by a rooted tree $\mathcal{T}$. The probability that the learning algorithm solves the two-hypothesis channel distinction task correctly is upper bounded by*

$$\frac{1}{2} \sum_{\ell \in \mathrm{leaf}(\mathcal{T})} \left| \left(\mathop{\mathbb{E}}_{C \sim \mathcal{D}_A} p^C(\ell) \right) - \left(\mathop{\mathbb{E}}_{C \sim \mathcal{D}_B} p^C(\ell) \right) \right|. \tag{9.167}$$

Review of the Weingarten calculus

Here we will review Haar measures on unitary, orthogonal, and symplectic matrices, and present several key lemmas that we will leverage in the later proofs. First we recall the definitions of orthogonal and symplectic matrices. Denoting the set of unitary matrices on $(\mathbb{C}^2)^{\otimes n}$ by

$$U(d) = \{U \in \mathrm{Mat}_{d\times d}(\mathbb{C}) \,:\, U^\dagger = U^{-1}\}\,, \tag{9.168}$$

the set of orthogonal matrices on is given by

$$O(d) = \{O \in U(d) \,:\, O^t = O^{-1}\}\,, \tag{9.169}$$

and the set of symplectic matrices is given by

$$\mathrm{Sp}(d/2) = \{S \in U(d) \,:\, JS^t J^{-1} = S^{-1}\}\,. \tag{9.170}$$

Here J is the symplectic form

$$J = \begin{bmatrix} \mathbf{0}_{d/2\times d/2} & \mathbb{1}_{d/2\times d/2} \\ -\mathbb{1}_{d/2\times d/2} & \mathbf{0}_{d/2\times d/2} \end{bmatrix}, \tag{9.171}$$

where $\mathbf{0}_{d/2\times d/2}$ is the $d/2 \times d/2$ matrix of all zeroes and $\mathbb{1}_{d/2\times d/2}$ is the $d/2 \times d/2$ identity matrix. The quantity JS^tJ^{-1} is sometimes called the "symplectic transpose" and denoted by S^D.

Note that $\mathrm{Sp}(d/2)$ is sometimes called the "symplectic unitary group" to distinguish it from the group of symplectic matrices which need not be unitary. For the orthogonal group, the matrices O will necessarily be real. For the symplectic unitary group, the matrices S could be complex numbers. For our purposes, we will adopt standard terminology by dropping the word 'unitary' since the context is clear.

Since $U(d), O(d), \mathrm{Sp}(d/2)$ are each compact Lie groups, they admit canonical Haar measures which are right and left-invariant under group multiplication. For instance, for $U(d)$, the Haar measure satisfies

$$\int_{U(d)} dU\, f(U) = \int_{U(d)} dU\, f(VU) = \int_{U(d)} dU\, f(UV) \tag{9.172}$$

for any $V \in U(d)$ and any $f(U)$. Analogous expressions hold for the Haar measures corresponding to $O(d)$ and $\text{Sp}(d/2)$. Such Haar integrals will be essential for our proofs, and so here we catalog important properties.

Now we turn to discussing more detailed properties of the Haar integrals. Our short overview will be based on (Collins and Matsumoto, 2017; Y. Gu, 2013; Matsumoto, 2013; Aharonov, J. S. Cotler, and Qi, 2021). Instead of using the integral notation, we will often use expectation values $\mathbb{E}_{U\sim\text{Haar}}[\,\cdot\,]$.

Haar averaging over $U(d)$

For our purposes it will be useful to study moments of the Haar ensemble, in particular

$$\mathbb{E}_{U\sim\text{Haar}}\left[U_{i_1 j_1} U_{i_2 j_2} \cdots U_{i_k j_k} \overline{U_{i'_1 j'_1} U_{i'_2 j'_2} \cdots U_{i'_k j'_k}}\right] \tag{9.173}$$

$$= \sum_{\sigma,\tau\in S_k} \delta_{\sigma(I),I'} \delta_{\tau(J),J'} \text{Wg}^U(\sigma\tau^{-1}, d)\,. \tag{9.174}$$

This equation requires some unpacking. On the left-hand side, the bar denotes complex conjugation; for instance $\overline{U}_{ij} = U^\dagger_{ji}$. On the right-hand side, S_k is the symmetric group on k elements, I is a multi-index $I = (i_1, ..., i_k)$ and similarly for I', J, J', and

$$\delta_{\sigma(I),I'} := \delta_{i_{\sigma(1)},i'_1} \delta_{i_{\sigma(2)},i'_2} \cdots \delta_{i_{\sigma(k)},i'_k}\,. \tag{9.175}$$

Finally, $\text{Wg}^U(\,\cdot\,, d)$ is a map $S_k \longrightarrow \mathbb{R}$ called the *unitary Weingarten function* to be specified shortly. To further compress notation, it will be convenient to fully commit to multi-index notation and write $U^{\otimes k}_{IJ} = U_{i_1 j_1} U_{i_2 j_2} \cdots U_{i_k j_k}$ so that (9.173) becomes

$$\mathbb{E}_{U\sim\text{Haar}}\left[U^{\otimes k}_{IJ} U^{\dagger\,\otimes k}_{J'I'}\right] = \sum_{\sigma,\tau\in S_k} \delta_{\sigma(I),I'} \delta_{\tau(J),J'} \text{Wg}^U(\sigma\tau^{-1}, d)\,. \tag{9.176}$$

The utility of (9.176) is that it allows us to compute $\mathbb{E}_{\text{Haar}}[U^{\otimes k} \otimes U^{\dagger\otimes k}]$, and matrix elements thereof, in terms of data of the symmetric group on T elements. We remark that $\mathbb{E}_{\text{Haar}}[U^{\otimes k} \otimes U^{\dagger\otimes \ell}]$ vanishes for $k \neq \ell$, so (9.176) covers all non-trivial cases.

It still remains to specify the unitary Weingarten function $\text{Wg}^U(\,\cdot\,, d)$. In fact, it can be regarded as the inverse of an easily specified matrix. To this end, in a slight abuse of notation, we will let permutations τ, σ in S_k also label their representations on $\mathcal{H}^{\otimes k}$. That is, τ will denote a unitary $d^k \times d^k$ matrix on $\mathcal{H}^{\otimes k}$ which permutes the k copies of $\mathcal{H}$ according to the permutation specified by the label τ. Now we can readily define

$$G^U(\sigma\tau^{-1}, d) := \mathrm{tr}(\sigma\tau^{-1}) = d^{\#(\sigma\tau^{-1})} \tag{9.177}$$

such that $\#(\sigma\tau^{-1})$ counts the number of cycles of $\sigma\tau^{-1}$. Now $\mathrm{Wg}^U(\,\cdot\,, d)$ is defined by the identity

$$\sum_{\tau \in S^k} \mathrm{Wg}^U(\sigma^{-1}\tau, d)\, G^U(\tau^{-1}\pi, d) = \delta_{\sigma,\pi} \tag{9.178}$$

for all $\sigma, \pi \in S_k$. This equation expresses that Wg^U and G^U are in fact inverses as $k! \times k!$ matrices. To see this more readily, we can use the notation $\mathrm{Wg}^U(\sigma^{-1}\tau, d) = \mathrm{Wg}^U_{\sigma,\tau}$ and $G^U(\tau^{-1}\pi, d) = G^U_{\tau,\pi}$ so that (9.178) is simply $\sum_\tau \mathrm{Wg}^U_{\sigma,\tau}\, G^U_{\tau,\pi} = \delta_{\sigma,\pi}$.

We conclude by presenting a theorem, corollary, and lemma which will be used in our proofs.

Theorem 3.2 of (Collins and Matsumoto, 2017). *For any $\sigma \in S_k$ and $d > \sqrt{6}k^{7/4}$,*

$$\frac{1}{1 - \frac{k-1}{d^2}} \leq \frac{(-1)^{k-\#(\sigma)} d^{2k-\#(\sigma)}\ Wg^U(\sigma, d)}{\prod_i \frac{(2\ell_i - 2)!}{(\ell_i - 1)!\ell_i!}} \leq \frac{1}{1 - \frac{6k^{7/2}}{d^2}} \tag{9.179}$$

where the left-hand side inequality is valid for any $d \geq k$. Note that $\sigma \in S_k$ has cycle type $(\ell_1, \ell_2, \ldots)$.

An immediate corollary is:

Corollary 17. $|\ Wg^U(\mathbb{1}, d) - d^{-k}| \leq \mathcal{O}(k^{7/2} d^{-(k+2)})$.

We also recapitulate a useful result from (Aharonov, J. S. Cotler, and Qi, 2021):

Lemma 6 of (Aharonov, J. S. Cotler, and Qi, 2021). $\sum_{\tau \in S_k} |\ Wg^U(\tau, d)| = \frac{(d-k)!}{d!}$.

Haar averaging over $O(d)$

Just as Haar averaging over $U(d)$ is intimately related to the permutation group S_k, Haar averaging over $O(d)$ (and likewise $\mathrm{Sp}(d/2)$) is intimately related to pair partitions $P_2(2k)$. Accordingly, we will begin with discussing pair partitions.

Informally, a pair partition on $2k$ is a way of pairing off $2k$ elements (e.g., pairing off people in a ballroom dance class). There are in fact $\frac{(2k)!}{2^k k!} = (2k-1)!!$ possible pairings of $2k$ elements. More formally, a pair partition $\mathfrak{m} \in P_2(2k)$ is a function $\mathfrak{m} : [2k] \to [2k]$ satisfying $\mathfrak{m}(2i-1) < \mathfrak{m}(2i)$ for $1 \le i \le k$ and $\mathfrak{m}(1) < \mathfrak{m}(3) < \cdots < \mathfrak{m}(2k-1)$. The pair permutation is often notated as

$$\mathfrak{m} = \{\mathfrak{m}(1), \mathfrak{m}(2)\}\{\mathfrak{m}(3), \mathfrak{m}(4)\} \cdots \{\mathfrak{m}(2k-1), \mathfrak{m}(2k)\} \tag{9.180}$$

where the brackets denote individual pairs, and the constraints on $\mathfrak{m}$ order the pairs in a canonical (and unique) manner. In words, within each pair the ‘left’ element is always less than the ‘right’ element, and the pairs themselves are ordered according to the ‘left’ element of each pair.

It is natural to endow pair permutations with a group structure so that they form a subgroup M_{2k} of the permutation group S_{2k}. We simply define M_{2k} by

$$M_{2k} := \{\sigma \in S_{2k} \,:\, \sigma(2i-1) < \sigma(2i) \text{ for } 1 \le i \le k\,, \tag{9.181}$$

$$\sigma(1) < \sigma(3) < \cdots < \sigma(2k-1)\} \tag{9.182}$$

and it is readily checked that M_{2k} forms a group. We will often leverage the natural bijection between $P_2(2k)$, namely $\mathfrak{m} \mapsto \sigma_{\mathfrak{m}}$ where $\mathfrak{m}(i) = \sigma_{\mathfrak{m}}(i)$ for all i. The identity pairing is denoted by $\mathfrak{e}$, and by the above bijection maps to the identity permutation $\sigma_{\mathfrak{e}} = \mathbb{1}$.

Pair permutations also have a notion of cycles, which differs from that of the symmetric group. That is, the cycle type of $\sigma_{\mathfrak{m}}$ as an element of M_{2k} is different from the cycle type of $\sigma_{\mathfrak{m}}$ thought of as an element of S_{2k} . To construct the pair partition cycles $\sigma_{\mathfrak{m}}$ (we will also refer to this as the cycle type of $\mathfrak{m}$), consider the function $f_{\mathfrak{m}} : [2k] \to [2k]$ defined by

$$f_{\mathfrak{m}}(i) = \begin{cases} \mathfrak{m}(2j) & \text{if } i = \mathfrak{m}(2j-1) \\ \mathfrak{m}(2j-1) & \text{if } i = \mathfrak{m}(2j) \end{cases} . \tag{9.183}$$

This function maps i to the integer it is paired with under $\mathfrak{m}$. The function $f_{\mathfrak{e}}$ corresponds to the identity pairing. We can construct the cycles of $\mathfrak{m}$ as follows. Consider the sequence

$$(1, f_{\mathfrak{m}}(1), f_{\mathfrak{e}} \circ f_{\mathfrak{m}}(1), f_{\mathfrak{m}} \circ f_{\mathfrak{e}} \circ f_{\mathfrak{m}}(1), \ldots)\,. \tag{9.184}$$

This sequence is periodic, and so we truncate it at its period so that no element is repeated. We call this truncated list B_1, and view it is a cyclically ordered list (i.e., the list is regarded as the same if it is cyclically permuted). If B_1 contains all of

$[2k]$, then $\mathfrak{m}$ contains only one cycle, namely B_1. Otherwise, let j be the smallest integer in $[2k]$ with is not in B_1, and construct

$$(j, f_{\mathfrak{m}}(j), f_{\mathfrak{e}} \circ f_{\mathfrak{m}}(j), f_{\mathfrak{m}} \circ f_{\mathfrak{e}} \circ f_{\mathfrak{m}}(j), ...)\,. \tag{9.185}$$

This is likewise periodic, and we truncate it at its period to get the cyclically ordered list B_2. If B_1 and B_2 do not contain all of $[2k]$, then we construct a B_3, etc. When the procedure terminates, we have $B_1, B_2, ...$ which contain all of $[2k]$. The $B_1, B_2, ...$ are the pair partition cycles of $\mathfrak{m}$. Their corresponding lengths $b_1, b_2, ...$ are all even, and the cycle type (also called the coset type) of $\mathfrak{m}$ is given by

$$(\mu_1, \mu_2, ...) = (b_1/2, b_2/2, ...) \tag{9.186}$$

which is often listed in descending order of cycle size.

Using our notation for pair partitions as well as the multi-index notation we established previously, we have the following formula for a Haar integral over the orthogonal group

$$\mathbb{E}_{O\sim\text{Haar}}\big[O_{IJ}O^t_{J'I'}\big] = \sum_{\mathfrak{m},\mathfrak{n}\in P_2(2k)} \Delta_{\mathfrak{m}}(II')\,\Delta_{\mathfrak{n}}(JJ')\,\text{Wg}^O(\sigma_{\mathfrak{m}}^{-1}\sigma_{\mathfrak{n}}) \tag{9.187}$$

where II' merges the multi-indices I and I' as $II' = (i_1, i'_1, ..., i_k, i'_k)$ and likewise for JJ'. Letting $II' = \mathbf{I} = (\mathbf{i}_1, \mathbf{i}_2, ..., \mathbf{i}_{2k})$ we define

$$\Delta_{\mathfrak{m}}(\mathbf{I}) := \prod_{s=1}^{k} \delta_{\mathbf{i}_{\mathfrak{m}(2s-1)},\mathbf{i}_{\mathfrak{m}(2s)}}\,. \tag{9.188}$$

Similar to before, $\text{Wg}^O(\,\cdot\,, d)$ is a map $M_{2k} \longrightarrow \mathbb{R}$ called the orthogonal Weingarten function. Although we have written (9.188) in a way that emulates (9.176), the formula (9.188) has a different character to it. Specifically, examining the left-hand side, we can equivalently write it as $\mathbb{E}_{\text{Haar}}[O_{II',JJ'}]$ where we have simply used the equivalence $O^t_{J'I'} = O_{I'J'}$. That is, (9.188) tells us how to compute $\mathbb{E}_{\text{Haar}}[O^{\otimes 2k}]$ and matrix elements thereof for $2k$ even; this integral vanishes if we replace $2k$ by an odd number. By contrast with the Haar unitary setting where we needed as many U's as $U^\dagger$'s to get a non-trivial integral, here we just need an even number of O's, essentially because $O^\dagger = O^t$.

Analogous to the unitary setting discussed above, we can define the orthogonal Weingarten function $\text{Wg}^O(\,\cdot\,, d)$ in terms of a simpler function

$$G^O(\sigma_{\mathfrak{m}}^{-1}\sigma_{\mathfrak{n}}, d) := d^{\#^O(\sigma_{\mathfrak{m}}^{-1}\sigma_{\mathfrak{n}})} \tag{9.189}$$

where $\#^O(\sigma_{\mathfrak{m}}^{-1}\sigma_{\mathfrak{n}})$ counts the number of M_{2k}-cycles of $\sigma_{\mathfrak{m}}^{-1}\sigma_{\mathfrak{n}}$. We have the identity

$$\sum_{\mathfrak{n}\in P_2(2k)} \mathrm{Wg}^O(\sigma_{\mathfrak{m}}^{-1}\sigma_{\mathfrak{n}}, d)\, G^O(\sigma_{\mathfrak{n}}^{-1}\sigma_{\mathfrak{p}}, d) = \delta_{\mathfrak{m},\mathfrak{p}}\,. \tag{9.190}$$

which expresses that Wg^O and G^O are inverses as $(2k-1)!! \times (2k-1)!!$ matrices.

Finally, we present a theorem, corollary, and lemma which we will leverage in our proofs about the symmetry distinction problem:

Theorem 4.11 of (Collins and Matsumoto, 2017). *For any $\sigma_{\mathfrak{m}} \in M_{2k}$ and $d > 12k^{7/2}$,*

$$\frac{1-\frac{24k^{7/2}}{d}}{1-\frac{144k^7}{d^2}} \leq \frac{(-1)^{k-\#^O(\sigma_{\mathfrak{m}})} d^{2k-\#^O(\sigma_{\mathfrak{m}})}\ Wg^O(\sigma_{\mathfrak{m}}, d)}{\prod_i \frac{(2\mu_i-2)!}{(\mu_i-1)!\mu_i!}} \leq \frac{1}{1-\frac{144k^7}{d^2}} \tag{9.191}$$

where $\sigma_{\mathfrak{m}}$ has M_{2k}-cycle type $(\mu_1, \mu_2, \ldots)$.

We have the immediate corollary

Corollary 18. $|\, Wg^O(\sigma_{\mathfrak{e}}, d) - d^{-k}| \leq O(k^7 d^{-(k+2)})\,.$

Analogous to Lemma 6 of (Aharonov, J. S. Cotler, and Qi, 2021) written above, we have (Aharonov, J. S. Cotler, and Qi, 2021):

Lemma 8 of (Aharonov, J. S. Cotler, and Qi, 2021). $\sum_{\mathfrak{m}\in P_2(2k)} |\, Wg^O(\sigma_{\mathfrak{m}}, d)| = \frac{(d-2k)!!}{d!!}\,.$

Haar averaging over $\mathrm{Sp}(d/2)$

Haar averaging over the $\mathrm{Sp}(d/2)$ is very similar to the orthogonal setting. We have the identity

$$\mathbb{E}_{S\sim\mathrm{Haar}}\left[S_{IJ}S^t_{J'I'}\right] = \sum_{\mathfrak{m},\mathfrak{n}\in P_2(2k)} \Delta'_{\mathfrak{m}}(II')\Delta'_{\mathfrak{n}}(JJ')\, \mathrm{Wg}^{\mathrm{Sp}}(\sigma_{\mathfrak{m}}^{-1}\sigma_{\mathfrak{n}}, d/2) \tag{9.192}$$

where

$$\Delta'_{\mathfrak{m}}(\mathbf{I}) := \prod_{s=1}^{k} J_{\mathbf{i}_{\mathfrak{m}(2s-1)},\mathbf{i}_{\mathfrak{m}(2s)}}\,. \tag{9.193}$$

Here J is the canonical symplectic form defined in (9.171). The symplectic Weingarten function $\mathrm{Wg}^{\mathrm{Sp}}(\,\cdot\,, d/2)$ taking $M_{2k} \to \mathbb{R}$ is a small modification of the orthogonal Weingarten function, namely

$$\mathrm{Wg}^{\mathrm{Sp}}(\sigma_{\mathfrak{m}}, d/2) = (-1)^k \epsilon(\sigma_{\mathfrak{n}})\, \mathrm{Wg}^O(\sigma_{\mathfrak{m}}, -d) \tag{9.194}$$

where $\epsilon(\sigma_\mathfrak{m})$ is the signature of $\sigma_\mathfrak{m}$ thought of as an element of S_{2k}.

We give a theorem, corollary and lemma analogous to the ones for the orthogonal group above.

Theorem 4.10 of (Collins and Matsumoto, 2017). *For any $\sigma_\mathfrak{m} \in M_{2k}$ and $d > 6k^{7/2}$, we have*

$$\frac{1}{1-\frac{k-1}{(d/2)^2}} \leq \frac{d^{2k-\#^{Sp}(\sigma_\mathfrak{m})}|\,Wg^{Sp}(\sigma_\mathfrak{m}, d/2)|}{\prod_i \frac{(2\mu_i-2)!}{(\mu_i-1)!\mu_i!}} \leq \frac{1}{1-\frac{6k^{7/2}}{(d/2)^2}} \tag{9.195}$$

where $\sigma_\mathfrak{m}$ has M_{2k}-cycle type $(\mu_1, \mu_2, ...)$ and $\#^{Sp}(\sigma_\mathfrak{m}) = \#^O(\sigma_\mathfrak{m})$.

This has the direct corollary

Corollary 19. $|\,Wg^{Sp}(\sigma_\mathfrak{e}, d/2) - d^{-k}| \leq O(k^{7/2}d^{-(k+2)})$.

From (Aharonov, J. S. Cotler, and Qi, 2021) we borrow the useful lemma:

Lemma 10 of (Aharonov, J. S. Cotler, and Qi, 2021).

$$\sum_{\mathfrak{m}\in P_2(2k)} |\,Wg^{Sp}(\sigma_\mathfrak{m}, d/2)| = \prod_{j=0}^{k-1} \frac{1}{d+2j}. \tag{9.196}$$

Depolarizing channel versus random unitary

In this subsection, we look at the task of distinguishing between the depolarizing channel and a random unitary.

Lower bound without quantum memory

We are now prepared to establish the following results:

Theorem 57 (Exponential hardness of fixed unitary task without quantum memory)**.** *Any learning algorithm without quantum memory requires*

$$T \geq \Omega\left(d^{1/3}\right), \tag{9.197}$$

to correctly distinguish between the completely depolarizing channel $\mathcal{D}$ on n qubits from a fixed, Haar-random unitary channel $\mathcal{U}[\rho] = U\rho U^\dagger$ on n qubits with probability at least $2/3$.

Theorem 58 (Exponential hardness of fixed orthogonal matrix task without quantum memory)**.** *Any learning algorithm without quantum memory requires*

$$T \geq \Omega\left(d^{2/7}\right), \tag{9.198}$$

to correctly distinguish between the completely depolarizing channel $\mathcal{D}$ on n qubits from a fixed, Haar-random orthogonal matrix channel $\mathcal{U}[\rho] = O\rho O^t$ on n qubits with probability at least 2/3.

Theorem 59 (Exponential hardness of fixed symplectic matrix task without quantum memory)**.** *Any learning algorithm without quantum memory requires*

$$T \geq \Omega\left(d^{1/3}\right), \tag{9.199}$$

to correctly distinguish between the completely depolarizing channel $\mathcal{D}$ on n qubits from a fixed, Haar-random symplectic matrix channel $\mathcal{U}[\rho] = S\rho S^D$ on n qubits with probability at least 2/3.

Hence, we established an exponential lower bound when the algorithms do not have a quantum memory. The proofs of Theorems 57, 58, and 59 have many similarities; however, the first involves heavy use of the combinatorics of permutations, whereas the latter two involve heavy use of the combinatorics of pair permutations. As such, we will prove Theorems 57 first, followed by a simultaneous proof of Theorems 58 and 59.

We now turn to a proof of Theorem's 57, 58, and 59 which are our main result about the fixed unitary problem. We note that a special case of these theorems were proved in (Aharonov, J. S. Cotler, and Qi, 2021), namely where the learning protocol (see Definition 17) does not have an auxiliary system, i.e. $\mathcal{H}_{\text{aux}}$ is a trivial Hilbert space. The inclusion of an $\mathcal{H}_{\text{aux}}$ of arbitrary size $d' = 2^{n'}$ is our main technical contribution here; our proof will follow the same contours as that of (Aharonov, J. S. Cotler, and Qi, 2021), but with substantive modifications and generalizations. Indeed, the original proof strategy leads to T bounds like (9.197), (9.198), and (9.199), but reduced by factors of d'^T, rendering the bounds useless even when $d' \sim O(1)$ (but non-zero).

Proof of Theorem 57

Let us begin with a proof of Theorem 57:

Proof. The proof begins by utilizing Lemma 59, which gives

$$\frac{2}{3} \le \frac{1}{2} \sum_{\ell \in \mathrm{leaf}(\mathcal{T})} \left| p^{\mathcal{D}}(\ell) - \left(\mathop{\mathbb{E}}_{\mathcal{U}} p^{\mathcal{U}}(\ell) \right) \right|. \tag{9.200}$$

The goal now would be to obtain an upper bound right hand side. It is convenient to establish some notation. Each root-to-leaf path in $\mathcal{T}$ is specified by a sequence of vertices $v_0, v_1, ..., v_T$ where $v_0 = r$ is the root and $v_T = \ell$ is a leaf. Moreover, the leaf ℓ determines the entire root-to-leaf path: it is the shortest path from that leaf to the root. So knowing ℓ is the same as knowing the entire path $v_0 = r, v_1, ..., v_{T-1}, v_T = \ell$. Recall from Eq. (9.166) and following the root-to-leaf path $v_0 = r, v_1, ..., v_{T-1}, v_T = \ell$, the probability of the leaf ℓ under the channel C is

$$p^C(\ell) = \prod_{t=1}^{T} \left(w_{v_t} dd' \langle \psi_{v_t} | (C \otimes \mathcal{I}_{\mathrm{aux}})[|\phi_{v_{t-1}}\rangle\langle\phi_{v_{t-1}}|] |\psi_{v_t}\rangle \right), \tag{9.201}$$

where each $|\phi_{v_{t-1}}\rangle, |\psi_{v_{t-1}}\rangle$ lives in (dd')-dimensional Hilbert space $\mathcal{H} \simeq \mathcal{H}_{\mathrm{main}} \otimes \mathcal{H}_{\mathrm{aux}}$ from Definition 17. To analyze the probability, we let

$$|\Phi_\ell\rangle = |\phi_{v_0}\rangle \otimes |\phi_{v_1}\rangle \otimes \cdots \otimes |\phi_{v_{T-1}}\rangle \tag{9.202}$$

$$|\Psi_\ell\rangle = |\psi_{v_1}\rangle \otimes |\psi_{v_2}\rangle \otimes \cdots \otimes |\psi_{v_T}\rangle \tag{9.203}$$

$$W_\ell = w_{v_1} w_{v_2} \cdots w_{v_T} \tag{9.204}$$

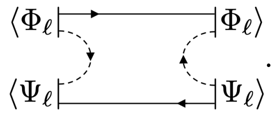
where $v_0 = r$ and $v_T = \ell$. Notice that $|\Phi_\ell\rangle$ and $|\Psi_\ell\rangle$ each live in $\mathcal{H}^{\otimes T}$. Recall from Definition 17, for any node u, the set $\{\sqrt{w_v dd'} |\psi_v\rangle\langle\psi_v|\}_{v \in \mathrm{child}(u)}$ is a POVM. Hence, we have $\sum_{v \in \mathrm{child}(u)} w_v dd' |\psi_v\rangle\langle\psi_v| = \mathbb{1}_{\mathrm{main}} \otimes \mathbb{1}_{\mathrm{aux}}$. It is not hard to use this fact to derive the following identity

$$\sum_{\ell \,\in\, \mathrm{leaf}(\mathcal{T})} (dd')^T W_\ell |\Psi_\ell\rangle\langle\Psi_\ell| = (\mathbb{1}_{\mathrm{main}} \otimes \mathbb{1}_{\mathrm{aux}})^{\otimes T} \tag{9.205}$$

and so accordingly $\sum_{\ell \,\in\, \mathrm{leaf}(\mathcal{T})} W_\ell = 1$.

With these notations at hand, we can write $p^{\mathcal{D}}(\ell)$ diagrammatically as

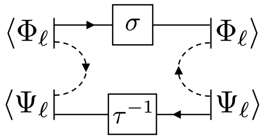
$$p^{\mathcal{D}}(\ell) = d'^T\, W_\ell$$



. (9.206)

For $\mathbb{E}_{\mathrm{Haar}}[p^{\mathcal{U}}(\ell)]$, we utilize the Haar averaging of unitary group discussed in Section 9.9 to obtain

$$\mathbb{E}_{\mathrm{Haar}}[p^{\mathcal{U}}(\ell)] = (dd')^T \sum_{\sigma,\tau \in S_T} W_\ell$$



$$\mathrm{Wg}^U(\tau\sigma^{-1}, d) \tag{9.207}$$

where the solid lines correspond to $\mathcal{H}_{\text{main}}^{\otimes T}$ and the dotted lines correspond to $\mathcal{H}_{\text{aux}}^{\otimes T}$. It is convenient to let $p_{\sigma,\tau}(\ell)$ denote the summand of (9.207). We now use the triangle inequality in combination with the Cauchy-Schwarz inequality to write

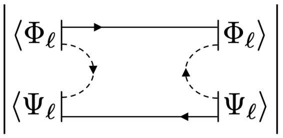
$$\begin{aligned}
&\frac{1}{2}\sum_{\ell\,\in\,\text{leaf}(\mathcal{T})} |p^{\mathcal{D}}(\ell) - \mathbb{E}_{\text{Haar}}[p^{\mathcal{U}}(\ell)]| \\
&\le \frac{1}{2}\sum_{\ell\,\in\,\text{leaf}(\mathcal{T})} |p^{\mathcal{D}}(\ell) - p_{\mathbb{1},\mathbb{1}}(\ell)| + \frac{1}{2}\sum_{\ell\,\in\,\text{leaf}(\mathcal{T})}\sum_{\sigma\neq\mathbb{1}} |p_{\sigma,\mathbb{1}}(\ell)| \qquad (9.208)\\
&\qquad + \frac{1}{2}\sum_{\ell\,\in\,\text{leaf}(\mathcal{T})}\sum_{\tau\neq\mathbb{1},\,\sigma} |p_{\sigma,\tau}(\ell)|\,. \qquad (9.209)
\end{aligned}$$

We will bound each term in turn.

First term

Applying Cauchy-Schwarz to the first term in (9.208) we find that it is less than or equal to

$$\frac{1}{2}\sum_{\ell\,\in\,\text{leaf}(\mathcal{T})} (dd')^T\, W_\ell$$



$$\left|\mathrm{Wg}^U(\mathbb{1},d) - \frac{1}{d^T}\right|\,. \qquad (9.210)$$

We can remove the absolute values on the diagrammatic term since it is strictly positive; this enables us to perform the sum over leaves $\sum_{\ell\in\text{leaf}(\mathcal{T})}$ and via the identity given in (9.205). The first term in (9.208) can now be written as

$$\frac{d^T}{2}\left|\mathrm{Wg}^U(\mathbb{1},d) - \frac{1}{d^T}\right|\,. \qquad (9.211)$$

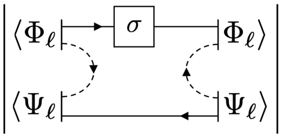
Since by Corollary 17 we have $|\mathrm{Wg}^U(\mathbb{1},d) - \frac{1}{d^T}| \le O(T^{7/2}/d^{T+2})$ for $T < \left(\frac{d}{\sqrt{6}}\right)^{4/7}$, Eqn. (9.211) is $O(T^{7/2}/d^2)$ and so

$$\frac{1}{2}\sum_{\ell\,\in\,\text{leaf}(\mathcal{T})} |p^{\mathcal{D}}(\ell) - p_{\mathbb{1},\mathbb{1}}(\ell)| \le O\!\left(\frac{T^{7/2}}{d^2}\right)\,. \qquad (9.212)$$

Second term

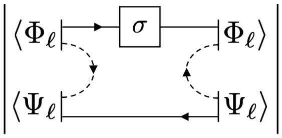
Next we treat the second term in (9.208), namely $\frac{1}{2}\sum_{\ell\,\in\,\text{leaf}(\mathcal{T})}\sum_{\sigma\neq\mathbb{1}} |p_{\sigma,\mathbb{1}}(\ell)|$. Utilizing the Cauchy-Schwarz inequality, this term is less than or equal to

$$\frac{1}{2}\sum_{\ell\,\in\,\text{leaf}(\mathcal{T})}\sum_{\sigma\neq\mathbb{1}} (dd')^T\, W_\ell$$

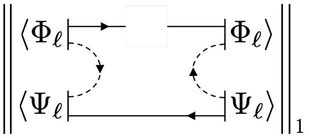


$$|\mathrm{Wg}^U(\sigma^{-1},d)|\,. \qquad (9.213)$$

We can dissect the middle term in the summand using the Hölder inequality

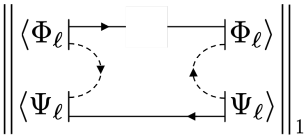


$$\leq$$

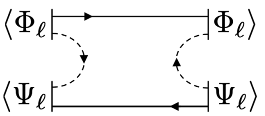


$$\|\sigma\|_\infty \tag{9.214}$$

and notice that $\|\sigma\|_\infty = 1$. Furthermore, using the fact that the matrix inside the 1-norm on the right-hand side is positive semi-definite, we can replace the 1-norm with the trace and find the bound



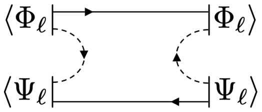
$$=$$



$$. \tag{9.215}$$

Thus (9.213) is upper bounded by

$$\frac{1}{2}\sum_{\ell\,\in\,\mathrm{leaf}(\mathcal{T})}\sum_{\sigma\neq\mathbb{1}}(dd')^T\,W_\ell$$



$$|\mathrm{Wg}^U(\sigma^{-1}, d)|\,. \tag{9.216}$$

Then applying the identity (9.205) we are left with

$$\frac{d^T}{2}\sum_{\sigma\neq\mathbb{1}}|\mathrm{Wg}^U(\sigma^{-1}, d)| \tag{9.217}$$

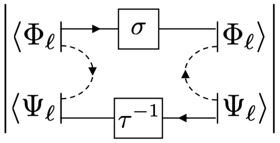
which is less than or equal to $O(T^2/d)$ by Lemma 6 of (Aharonov, J. S. Cotler, and Qi, 2021). In summary, we have

$$\frac{1}{2}\sum_{\ell\,\in\,\mathrm{leaf}(\mathcal{T})}\sum_{\sigma\neq\mathbb{1}}|p_{\sigma,\mathbb{1}}(\ell)| \leq O\left(\frac{T^2}{d}\right). \tag{9.218}$$

Third term

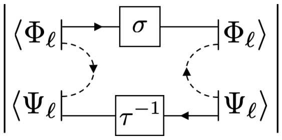
Finally we treat the third term in (9.208), namely $\frac{1}{2}\sum_{\ell\,\in\,\mathrm{leaf}(\mathcal{T})}\sum_{\tau\neq\mathbb{1},\,\sigma}|p_{\sigma,\tau}(\ell)|$, which is the most difficult case. Leveraging the Cauchy-Schwarz inequality, this term is upper bounded by

$$\frac{1}{2}\sum_{\ell\,\in\,\mathrm{leaf}(\mathcal{T})}\sum_{\tau\neq\mathbb{1},\sigma}(dd')^T\,W_\ell$$

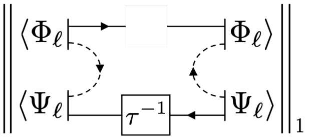


$$|\mathrm{Wg}^U(\tau\sigma^{-1}, d)|\,. \tag{9.219}$$

Applying the Hölder inequality to the second term in the summand as before, we find

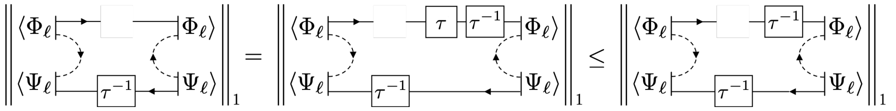


$$\leq$$



$$\|\sigma\|_\infty \tag{9.220}$$

where $\|\sigma\|_\infty = 1$. We also use the convenient inequality

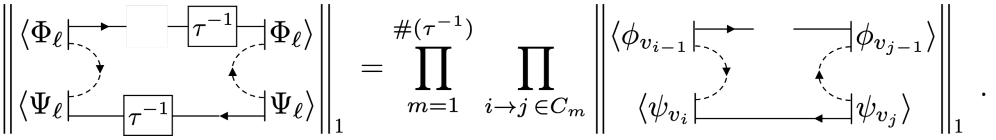


(9.221)

to reorganize the order of the tensor legs; we have used the Hölder inequality to go from the middle term to the last term, and the fact that $\|\tau\|_\infty = 1$.

For a fixed permutation τ^{-1}, we can decompose it into cycles $C_1 C_2 \cdots C_{\#(\tau^{-1})}$. We say that $i \to j$ is in the mth cycle C_m if $C_m = (\cdots ij \cdots)$. Using this notation, for fixed $\tau^{-1} = C_1 C_2 \cdots C_{\#(\tau^{-1})}$ and letting $v_0 = r, v_1, ..., v_{T-1}, v_T = \ell$ be the root-to-leaf path terminating in ℓ, we have

Lemma 60. *We have the following identity represented using tensor network diagrams.*

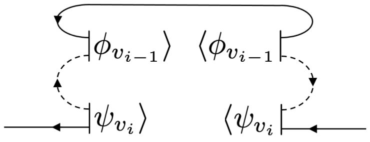


(9.222)

Proof. Take any cycle C_m and any $i \to j \in C_m$. The solid leg of $\langle\phi_{v_{i-1}}|$ is dangling and the dotted leg connects to $\langle\psi_{v_i}|$. Similarly, the solid leg of $|\phi_{v_{j-1}}\rangle$ is dangling and the dotted leg connects to $|\psi_{v_j}\rangle$. Lastly, the solid leg of $\langle\psi_{v_i}|$ connects to $|\psi_{v_j}\rangle$. This accounts for all legs among $\phi_{v_{i-1}}$, ψ_{v_i}, $\phi_{v_{j-1}}$, and ψ_{v_j}, so the part of the diagram on the left of (9.222) that corresponds to these four states is not connected to the rest of the diagram. In this fashion, we conclude that the diagram on the left is a tensor product of the diagrams on the left for all m and $i \to j \in C_m$, from which the lemma follows. □

We would like to convert the product of trace norms in (9.222) into a product of traces. We do so via the following basic estimate. First, for any $i \in [T]$, define the unnormalized density operator $\widetilde{\rho}_{v_i} \in \mathrm{Mat}_{d\times d}(\mathbb{C})$ by

$\widetilde{\rho}_{v_i} :=$

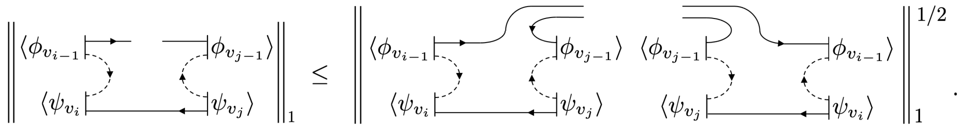


(9.223)

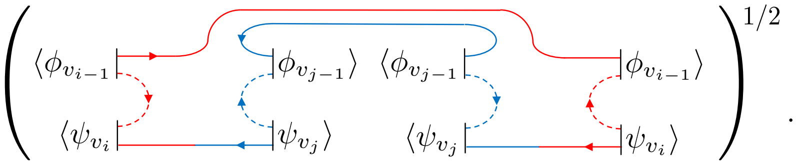
Lemma 61. *For any $i \to j \in C_m$, the corresponding term on the right-hand side of* (9.222) *is upper bounded by* $\mathrm{tr}(\widetilde{\rho}_{v_i}\widetilde{\rho}_{v_j})$. *In particular,* (9.222) *is at most*

$$\prod_{m=1}^{\#(\tau^{-1})} \prod_{i\to j\in C_m} \sqrt{\mathrm{tr}(\widetilde{\rho}_{v_i}\widetilde{\rho}_{v_j})}. \tag{9.224}$$

Proof. Using the relations $\|A\|_1 = \|A \otimes A^\dagger\|_1^{1/2} \leq \|\text{SWAP} \cdot (A \otimes A^\dagger)\|_1^{1/2}$, we find



Since the operator inside the 1-norm on the right-hand side is clearly positive semi-definite, we can replace the 1-norm with a trace, namely



(9.225)

Observe that we can equivalently rewrite (9.225) as $\sqrt{\text{tr}(\widetilde{\rho}_{v_i}\widetilde{\rho}_{v_j})}$, where $\widetilde{\rho}_{v_i}$ is the component of the diagram denoted in red and $\widetilde{\rho}_{v_j}$ is the one in blue, yielding the first part of the lemma. The second follows from plugging this into the right-hand side of Eqn. (9.222). □

To bound (9.224), it is convenient to develop some notation for cycles C_m. The usual notation for a cycle of length p (which for us is less than or equal to T) is $C_m = (a_1 a_2 \cdots a_p)$ where in our setting $\{a_1, a_2, ..., a_p\} \subseteq \{1, 2, ..., T\}$. We will decorate each a_i by an additional subscript as $a_{m,i}$ to remember that it that belongs to the mth cycle C_m. Similarly, we will sometimes write $p = |C_m|$ to remind ourselves that it depends on m. In this notation, we can write (9.224) as a product of

$$\sqrt{\text{tr}(\widetilde{\rho}_{v_{a_{m,1}}}\widetilde{\rho}_{v_{a_{m,2}}})}\sqrt{\text{tr}(\widetilde{\rho}_{v_{a_{m,2}}}\widetilde{\rho}_{v_{a_{m,3}}})}\cdots\sqrt{\text{tr}(\widetilde{\rho}_{v_{a_{m,p-1}}}\widetilde{\rho}_{v_{a_{m,p}}})}\sqrt{\text{tr}(\widetilde{\rho}_{v_{a_{m,p}}}\widetilde{\rho}_{v_{a_{m,1}}})} \tag{9.226}$$

over the different $m = 1, \ldots, \#(\tau^{-1})$. We can further process the above expression for a given m. We proceed by analyzing two cases: (i) $p = |C_m|$ is even, and (ii) $p = |C_m|$ is odd.

Case 1: *p is even.* Each term in Eqn. (9.226) has the form $\sqrt{\text{tr}(\widetilde{\rho}_{v_{a_{m,i}}}\widetilde{\rho}_{v_{a_{m,i+1}}})}$ except for the last term; however, if we treat the i subscripts of $a_{m,i}$ modulo p, then we can write the last term as $\sqrt{\text{tr}(\widetilde{\rho}_{v_{a_{m,p}}}\widetilde{\rho}_{v_{a_{m,p+1}}})}$. We elect to use this notation. Then we can rearrange and group the terms in Eqn. (9.226) as follows

$$\left(\prod_{i\text{ odd}}\sqrt{\text{tr}(\widetilde{\rho}_{v_{a_{m,i}}}\widetilde{\rho}_{v_{a_{m,i+1}}})}\right)\left(\prod_{j\text{ even}}\sqrt{\text{tr}(\widetilde{\rho}_{v_{a_{m,j}}}\widetilde{\rho}_{v_{a_{m,j+1}}})}\right). \tag{9.227}$$

Using the inequality $ab \leq \frac{1}{2}(a^2 + b^2)$, the above is upper bounded by

$$\frac{1}{2}\prod_{i\text{ odd}} \mathrm{tr}(\widetilde{\rho}_{v_{a_{m,i}}}\widetilde{\rho}_{v_{a_{m,i+1}}}) + \frac{1}{2}\prod_{j\text{ even}} \mathrm{tr}(\widetilde{\rho}_{v_{a_{m,j}}}\widetilde{\rho}_{v_{a_{m,j+1}}})\,. \tag{9.228}$$

We will call the first term $\frac{1}{2}\,R_{m,-}$ and the second term $\frac{1}{2}\,R_{m,+}$.

Case 2: *p is odd.* Again consider the product in Eqn. (9.226). We can rearrange and group terms as

$$\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,p}}}\widetilde{\rho}_{v_{a_{m,1}}})}\left(\prod_{\substack{i\text{ odd}\\1\leq i\leq p-2}}\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,i}}}\widetilde{\rho}_{v_{a_{m,i+1}}})}\right)\left(\prod_{j\text{ even}}\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,j}}}\widetilde{\rho}_{v_{a_{m,j+1}}})}\right). \tag{9.229}$$

If two matrices A, B are Hermitian and positive semi-definite, then using Cauchy-Schwarz combined with operator norm inequalities we have $\mathrm{tr}(AB) \leq \|A\|_2\,\|B\|_2 \leq \|A\|_1\,\|B\|_1 \leq \mathrm{tr}(A)\,\mathrm{tr}(B)$. Accordingly, we have

$$\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,p}}}\widetilde{\rho}_{v_{a_{m,1}}})} \leq \sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,p}}})\,\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,1}}})} \tag{9.230}$$

and so (9.229) is upper bounded by

$$\left(\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,p}}})}\prod_{\substack{i\text{ odd}\\1\leq i\leq p-2}}\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,i}}}\widetilde{\rho}_{v_{a_{m,i+1}}})}\right)\left(\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,1}}})}\prod_{j\text{ even}}\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,j}}}\widetilde{\rho}_{v_{a_{m,j+1}}})}\right). \tag{9.231}$$

Again using the inequality $ab \leq \frac{1}{2}(a^2 + b^2)$, we have the upper bound

$$\frac{1}{2}\,\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,p}}})\prod_{\substack{i\text{ odd}\\1\leq i\leq p-2}}\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,i}}}\widetilde{\rho}_{v_{a_{m,i+1}}}) + \frac{1}{2}\,\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,1}}})\prod_{j\text{ even}}\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,j}}}\widetilde{\rho}_{v_{a_{m,j+1}}}) \tag{9.232}$$

where we analogously call the first term $\frac{1}{2}\,R_{m,-}$ and the second term $\frac{1}{2}\,R_{m,+}$.

In both cases, note that $R_{m,-}$ and $R_{m,+}$ implicitly depend on the leaf ℓ, so we will denote them by $R^{\ell}_{m,-}$ and $R^{\ell}_{m,+}$ when we wish to make this dependence explicit.

Putting together our notation and bounds from Case 1 and Case 2, we find that Eqn. (9.224) is upper bounded by

$$\prod_{m=1}^{\#(\tau^{-1})}\prod_{i\to j\,\in C_m}\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_i}\widetilde{\rho}_{v_j})} \leq \frac{1}{2^{\#(\tau^{-1})}}\prod_{m=1}^{\#(\tau^{-1})}\left(R_{m,-} + R_{m,+}\right)$$

$$= \frac{1}{2^{\#(\tau^{-1})}} \sum_{i_1,\dots,i_{\#(\tau^{-1})}=\pm} R_{1,i_1} R_{2,i_2} \cdots R_{\#(\tau^{-1}),i_{\#(\tau^{-1})}} \,. \tag{9.233}$$

Each term in the sum in the last line of (9.233) is a product of terms like $\text{tr}(\widetilde{\rho}_{v_i})$ and $\text{tr}(\widetilde{\rho}_{v_j}\widetilde{\rho}_{v_k})$. But the key point is that we have arranged the equations so that each term in the sum has $\widetilde{\rho}_{v_i}$ for each $i = 1, \dots, T$ appear *exactly once*. This has the following highly useful consequence.

Lemma 62. *Fix any* $i_1, \dots, i_{\#(\tau^{-1})} \in \{+, -\}$. *Then*

$$\sum_{\ell \in \textit{ leaf}(\mathcal{T})} (dd')^T \, W_\ell \, R^\ell_{1,i_1} R^\ell_{2,i_2} \cdots R^\ell_{\#(\tau^{-1}),i_{\#(\tau^{-1})}} \leq d^{\,T - \left\lfloor \frac{L(\tau^{-1})}{2} \right\rfloor} \tag{9.234}$$

where $L(\tau^{-1})$ *is the length of the longest cycle in* τ^{-1}.

Proof. For ease of notation, we will let $R^\ell_j \triangleq R^\ell_{j,i_j}$. Recall from (9.204) that $W_\ell = w_{v_1} w_{v_2} \cdots w_{v_T}$ and note that

$$\sum_{v:\,\text{depth}(v)=i} dd' \, w_v \, \widetilde{\rho}_v = \langle \phi_{\text{parent}(v)} | \phi_{\text{parent}(v)} \rangle \, \mathbb{1}_{d\times d} = \mathbb{1}_{d \times d} \,. \tag{9.235}$$

Accordingly we have that for any $\rho \in \text{Mat}_{d\times d}(\mathbb{C})$,

$$\sum_{v:\,\text{depth}(v)=i} dd' \, w_v \, \text{tr}(\widetilde{\rho}_v \, \rho) = \text{tr}(\rho) \,, \tag{9.236}$$

and in particular, for $\rho = \mathbb{1}$,

$$\sum_{v:\,\text{depth}(v)=i} dd' \, w_v \, \text{tr}(\widetilde{\rho}_v) = d. \tag{9.237}$$

We now turn to bounding the left-hand side of (9.234). Recalling that $\prod_j R^\ell_j$ as a product of terms like $\text{tr}(\widetilde{\rho}_{v_i})$ and $\text{tr}(\widetilde{\rho}_{v_i}\widetilde{\rho}_{v_{i'}})$, we will define some sets of indices encoding this data. Let $S_1^{(T)} \subseteq [T]$ denote the indices i for which $\text{tr}(\widetilde{\rho}_{v_i})$ appears in $\prod_j R^\ell_j$, and let $S_2^{(T)} \subseteq [T] \times [T]$ denote the set of (unordered) pairs (i, i') for which $\text{tr}(\widetilde{\rho}_{v_i}\widetilde{\rho}_{v_{i'}})$ appears, so that for any root-to-leaf path in $\mathcal{T}$ consisting of nodes $v_1, \dots, v_T = \ell$, we have

$$\prod_j R^\ell_j = \prod_{i \in S_1^{(T)}} \text{tr}(\widetilde{\rho}_{v_i}) \cdot \prod_{(i,i') \in S_2^{(T)}} \text{tr}(\widetilde{\rho}_{v_i}\widetilde{\rho}_{v_{i'}}) \tag{9.238}$$

by definition. Now construct $S_1^{(t)} \subseteq [t]$ and $S_2^{(t)} \subseteq [t] \times [t]$ for $1 \leq t < T$ inductively as follows. If $t \in S_1^{(t)}$ then define $S_1^{(t-1)} \triangleq S_1^{(t)} \backslash \{t\}$ and $S_2^{(t-1)} \triangleq S_2^{(t)}$.

Otherwise if $(t, t') \in S_2^{(t)}$ for some $t' < t$, then define $S_1^{(t-1)} \triangleq S_1^{(t-1)} \cup \{t'\}$ and $S_2^{(t-1)} \triangleq S_2^{(t)} \backslash \{(t, t')\}$. We collect some basic observations about these two set sequences:

Observation 1. *For every $i \in S_1^{(T)}$, we have that $i \in S_1^{(i)}$.*

Observation 2. *For every $(i, i') \in S_2^{(T)}$, if $i \leq i'$ then $i \in S_1^{(i)}$ while $i' \notin S_1^{(i')}$.*

The reason for defining these set sequences is that we can extract $\widetilde{\rho}_{v_T}$ from the product on the right-hand side of (9.238) and apply (9.237) (resp. (9.236)) if $T \in S_1^{(T)}$ (resp. $(T, t') \in S_2^{(T)}$ for some $t' < T$) to obtain

$$\sum_{\ell \in \mathrm{leaf}(\mathcal{T})} (dd')^T W_\ell \prod_j R_j^\ell \tag{9.239}$$

$$= d^{\mathbb{1}[T \in S_1^{(T)}]} \sum_{u\colon \mathrm{depth}(u) = T-1} (dd')^{T-1} W_u \prod_{i \in S_1^{(T-1)}} \mathrm{tr}(\widetilde{\rho}_{v_i}) \cdot \prod_{(i,i') \in S_2^{(T-1)}} \mathrm{tr}(\widetilde{\rho}_{v_i} \widetilde{\rho}_{v_{i'}}), \tag{9.240}$$

where $W_u = w_{v_1} \cdots w_{v_{T-1}}$ if the path from root to u in $\mathcal{T}$ consists of $v_1, \ldots, v_{T-1} = u$. Proceeding inductively, we can express the right-hand side of (9.239) as

$$d^{\sum_{t=1}^T \mathbb{1}[t \in S_1^{(t)}]}. \tag{9.241}$$

By Observations 1 and 2, $\sum_{t=1}^T \mathbb{1}[t \in S_1^{(t)}] = |S_1^{(T)}| + |S_2^{(T)}|$. Because every even cycle C_m of τ^{-1} contributes $|C_m|/2$ pairs to $S_2^{(T)}$, and every odd cycle C_m contributes $\lfloor |C_m|/2 \rfloor$ pairs to $S_2^{(T)}$ and one element to $S_1^{(T)}$, we conclude that

$$\begin{aligned}\sum_{\ell \in \mathrm{leaf}(\mathcal{T})} (dd')^T W_\ell \prod_j R_j^\ell &= d^{\sum_{m=1}^{\#(\tau^{-1})} \left\lceil \frac{|C_m|}{2} \right\rceil} \\ &= d^{T - \sum_{m=1}^{\#(\tau^{-1})} \left\lfloor \frac{|C_m|}{2} \right\rfloor} \\ &\leq d^{T - \left\lfloor \frac{L(\tau^{-1})}{2} \right\rfloor}\end{aligned} \tag{9.242}$$

as claimed. □

Putting our previous analysis together, in particular by combining Eqn.'s (9.220), (9.221), (9.222), (9.224), (9.233), and (9.234), we arrive at

$$\sum_{\ell \,\in\, \mathrm{leaf}(\mathcal{T})} \sum_{\tau \neq \mathbb{1}, \sigma} |p_{\sigma,\tau}(\ell)| \leq d^T \sum_\sigma |\mathrm{Wg}^U(\sigma^{-1}, d)| \sum_{\tau \neq \mathbb{1}} d^{-\left\lfloor \frac{L(\tau^{-1})}{2} \right\rfloor}. \tag{9.243}$$

The first sum on the right-hand side is bounded by

$$d^T \sum_{\sigma} |\mathrm{Wg}^U(\sigma^{-1}, d)| \leq 1 + O\left(\frac{T^2}{d}\right) . \tag{9.244}$$

Considering the second sum on the right-hand side, let $N(T, \ell)$ be the number of permutations in S_T where the length of the longest cycle is ℓ. Then the second sum can be written as

$$\sum_{\ell=2}^{T} N(T, \ell)\, d^{-\lfloor \frac{\ell}{2} \rfloor} \tag{9.245}$$

where we omit $k = 1$ from the sum since it corresponds to the identity permutation. Since $N(T, \ell) \leq \binom{T}{\ell}\, \ell! = \frac{T!}{(T-\ell)!} < T^\ell$, (9.245) is upper bounded by

$$\sum_{\ell=2}^{\infty} T^\ell\, d^{-\lfloor \frac{\ell}{2} \rfloor} = \frac{(1+T)\frac{T^2}{d}}{1 - \frac{T^2}{d}} = \frac{T^3}{d} + \frac{T^2}{d} + O\left(\frac{T^5}{d^2}\right) . \tag{9.246}$$

Now if $T \leq o(d^{1/3})$, then this quantity is $o(1)$ for some absolute constant $c > 0$. Altogether, we find

$$\frac{1}{2} \sum_{\ell \in \mathrm{leaf}(\mathcal{T})} \sum_{\tau \neq \mathbb{1}, \sigma} |p_{\sigma,\tau}(\ell)| \leq o(1) . \tag{9.247}$$

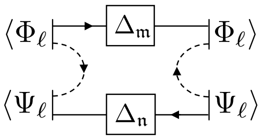
□

Proof of Theorems 58 and 59

As discussed above, we will present proof of Theorems 58 and 59, making heavy use of pair partitions.

Proof. The probability distribution $p^{\mathcal{D}}(\ell)$ is notated the same way as before. We can depict $\mathbb{E}_{\mathrm{Haar}}[p^O(\ell)]$ diagrammatically by

$$\mathbb{E}_{\mathrm{Haar}}[p^{\mathcal{O}}(\ell)] = (dd')^T \sum_{\mathfrak{m},\mathfrak{n} \in P_2(2T)} W_\ell$$

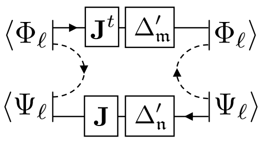


$$\mathrm{Wg}^{O}(\sigma_{\mathfrak{n}} \sigma_{\mathfrak{m}}^{-1}, d)$$

(9.248)

where $p^O_{\mathfrak{m},\mathfrak{n}}(\ell)$ denotes the summand. Similarly $\mathbb{E}_{\mathrm{Haar}}[p^S(\ell)]$ is given by

$$\mathbb{E}_{\mathrm{Haar}}[p^{\mathcal{S}}(\ell)] = (dd')^T \sum_{\mathfrak{m},\mathfrak{n} \in P_2(2T)} W_\ell$$



$$\mathrm{Wg}^{\mathrm{Sp}}(\sigma_{\mathfrak{n}} \sigma_{\mathfrak{m}}^{-1}, d/2)$$

(9.249)

where here $p^{\mathcal{S}}_{\mathfrak{m},\mathfrak{n}}(\ell)$ denotes the summand. Moreover, $\mathbf{J} := J^{\otimes t}$. As before, we use the triangle inequality in combination with the Cauchy-Schwarz inequality to write the two inequalities

$$\begin{aligned}
&\frac{1}{2}\sum_{\ell \in \mathrm{leaf}(\mathcal{T})} |p^{\mathcal{D}}(\ell) - \mathbb{E}_{\mathrm{Haar}}[p^{\mathcal{O}}(\ell)]| \\
&\leq \frac{1}{2}\sum_{\ell \in \mathrm{leaf}(\mathcal{T})} |p^{\mathcal{D}}(\ell) - p^{\mathcal{O}}_{\mathfrak{e},\mathfrak{e}}(\ell)| + \frac{1}{2}\sum_{\ell \in \mathrm{leaf}(\mathcal{T})}\sum_{\mathfrak{m}\neq\mathfrak{e}} |p^{\mathcal{O}}_{\mathfrak{m},\mathfrak{e}}(\ell)| \qquad (9.250)\\
&\qquad + \frac{1}{2}\sum_{\ell \in \mathrm{leaf}(\mathcal{T})}\sum_{\mathfrak{m}\neq\mathfrak{e},\,\mathfrak{n}} |p^{\mathcal{O}}_{\mathfrak{m},\mathfrak{n}}(\ell)|\,. \qquad (9.251)
\end{aligned}$$

and

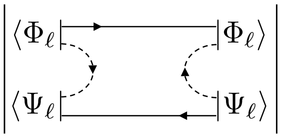
$$\begin{aligned}
&\frac{1}{2}\sum_{\ell \in \mathrm{leaf}(\mathcal{T})} |p^{\mathcal{D}}(\ell) - \mathbb{E}_{\mathrm{Haar}}[p^{\mathcal{S}}(\ell)]| \\
&\leq \frac{1}{2}\sum_{\ell \in \mathrm{leaf}(\mathcal{T})} |p^{\mathcal{D}}(\ell) - p^{\mathcal{S}}_{\mathfrak{e},\mathfrak{e}}(\ell)| + \frac{1}{2}\sum_{\ell \in \mathrm{leaf}(\mathcal{T})}\sum_{\mathfrak{m}\neq\mathfrak{e}} |p^{\mathcal{S}}_{\mathfrak{m},\mathfrak{e}}(\ell)| \qquad (9.252)\\
&\qquad + \frac{1}{2}\sum_{\ell \in \mathrm{leaf}(\mathcal{T})}\sum_{\mathfrak{m}\neq\mathfrak{e},\,\mathfrak{n}} |p^{\mathcal{S}}_{\mathfrak{m},\mathfrak{n}}(\ell)|\,. \qquad (9.253)
\end{aligned}$$

We will bound the right-hand sides of (9.250) and (9.252) term by term.

First term for $O(d)$ case

We apply the Cauchy-Schwarz inequality to the first term in (9.250) to find the upper bound

$$\frac{1}{2}\sum_{\ell \in \mathrm{leaf}(\mathcal{T})} (dd')^T\, W_\ell$$



$$\left|\mathrm{Wg}^{\mathcal{O}}(\sigma_{\mathfrak{e}}, d) - \frac{1}{d^T}\right|\,. \qquad (9.254)$$

As in the unitary setting, we remove the absolute values on the diagrammatic term by virtue of its positivity, and sum over leaves to get

$$\frac{d^T}{2}\left|\mathrm{Wg}^{\mathcal{O}}(\sigma_{\mathfrak{e}}, d) - \frac{1}{d^T}\right|\,. \qquad (9.255)$$

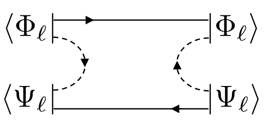
Using Corollary 18 which gives us $|\mathrm{Wg}^{\mathcal{O}}(\sigma_{\mathfrak{e}}, d) - \frac{1}{d^T}| \leq O(T^7/d^{T+2})$ for $T < \left(\frac{d}{12}\right)^{2/7}$, we immediately find

$$\frac{1}{2}\sum_{\ell \in \mathrm{leaf}(\mathcal{T})} |p^{\mathcal{D}}(\ell) - p^{\mathcal{O}}_{\mathfrak{e},\mathfrak{e}}(\ell)| \leq O\!\left(\frac{T^7}{d^2}\right)\,. \qquad (9.256)$$

First term for $\mathrm{Sp}(d/2)$ case

We recapitulate the same manipulations in the symplectic case. Applying the Cauchy-Schwarz inequality to the first term in (9.252) gives us the upper bound

$$\frac{1}{2}\sum_{\ell\,\in\,\mathrm{leaf}(\mathcal{T})}(dd')^T\,W_\ell\,\Bigg|$$



$$\Bigg|\left|\mathrm{Wg}^{\mathrm{Sp}}(\sigma_{\mathfrak{e}},d/2)-\frac{1}{d^T}\right|\,. \tag{9.257}$$

We again remove the absolute values around the diagrammatic term, and sum over leaves to find

$$\frac{d^T}{2}\left|\mathrm{Wg}^{\mathrm{Sp}}(\sigma_{\mathfrak{e}},d/2)-\frac{1}{d^T}\right|\,. \tag{9.258}$$

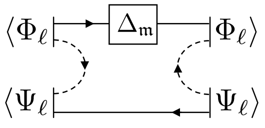
Using the analogous Corollary 19 which provides the bound $|\mathrm{Wg}^{O}(\sigma_{\mathfrak{e}},d)-\frac{1}{d^T}|\leq O(T^{7/2}/d^{T+2})$ for $T<\left(\frac{d}{6}\right)^{2/7}$, we have

$$\frac{1}{2}\sum_{\ell\,\in\,\mathrm{leaf}(\mathcal{T})}|p^{\mathcal{D}}(\ell)-p^{\mathcal{S}}_{\mathfrak{e},\mathfrak{e}}(\ell)|\leq O\!\left(\frac{T^{7/2}}{d^2}\right)\,. \tag{9.259}$$

Second term for $O(d)$ case

Applying Cauchy-Schwarz to the second term in (9.250), we find the upper bound

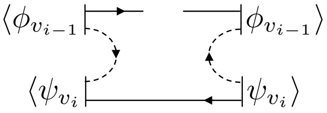
$$\frac{1}{2}\sum_{\ell\,\in\,\mathrm{leaf}(\mathcal{T})}\sum_{\mathfrak{m}\neq\mathfrak{e}}(dd')^T\,W_\ell\,\Bigg|$$



$$\Bigg|\,|\mathrm{Wg}^{O}(\sigma_{\mathfrak{m}}^{-1},d)|\,. \tag{9.260}$$

Let us define the matrix

$$C(2i-1,2i):= \tag{9.261}$$



where the $2i-1$ and $2i$ are just labels (i.e. they are not matrix indices). We will further define $C(2i,2i-1):=C(2i-1,2i)^t$. Then the diagrammatic term in (9.260) is equivalent to

$$\left|\mathrm{tr}\left(\Delta_{\mathfrak{m}}\bigotimes_{i=1}^{T}C(2i-1,2i)\right)\right|\,. \tag{9.262}$$

This can be expressed more explicitly as the absolute value of a product of traces of the C's, namely

$$\Big|\mathrm{tr}\big(C(2,1)C(f_{\mathfrak{m}}(1),f_{\mathfrak{e}}\circ f_{\mathfrak{m}}(1))C(f_{\mathfrak{m}}\circ f_{\mathfrak{e}}\circ f_{\mathfrak{m}}(1),f_{\mathfrak{e}}\circ f_{\mathfrak{m}}\circ f_{\mathfrak{e}}\circ f_{\mathfrak{m}}(1))\cdots \tag{9.263}$$

$$C(f_{\mathfrak{e}} \circ f_{\mathfrak{m}^{-\mathbb{6}}}(2), f_{\mathfrak{m}^{-\mathbb{6}}}(2)))$$
$$\cdot \operatorname{tr}(\cdots)\cdots\operatorname{tr}(\cdots)\Big| \tag{9.264}$$

where we have used the definition of $f_{\mathfrak{m}}$ and $f_{\mathfrak{e}}$ as per (9.183). Each trace corresponds to a particular M_{2T}-cycle of $\mathfrak{m}$. Using the 1-norm inequality

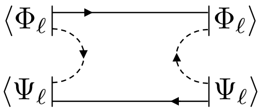
$$\|A_1 A_2 \cdots A_k\|_1 \le \prod_{i=1}^{k} \|A_i\|_1, \tag{9.265}$$

Eqn. (9.263) is upper bounded by

$$\prod_{i=1}^{T} \|C(2i-1, 2i)\|_1 \tag{9.266}$$

where we have used $\|C(2i-1,2i)\|_1 = \|C(2i, 2i-1)\|_1$. Since each $C(2i-1,2i)$ is positive semi-definite, $\|C(2i-1,2i)\|_1 = \operatorname{tr}(C(2i-1,2i))$ and so (9.260) has the upper bound

$$\frac{1}{2}\sum_{\ell\,\in\,\mathrm{leaf}(\mathcal{T})}\ \sum_{\mathfrak{m}\neq\mathfrak{e}} (dd')^T\, W_\ell$$



$$|\mathrm{Wg}^{O}(\sigma_{\mathfrak{m}}^{-1}, d)|\,. \tag{9.267}$$

Summing over leaves we arrive at

$$\frac{d^T}{2}\sum_{\mathfrak{m}\neq\mathfrak{e}} |\mathrm{Wg}^{O}(\sigma_{\mathfrak{m}}^{-1}, d)| \tag{9.268}$$

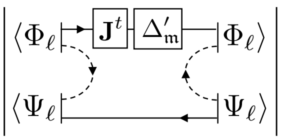
which is upper bounded by $O(T^7/d^2) + O(T^2/d)$ using Corollary 18 in combination with Lemma 8 of (Aharonov, J. S. Cotler, and Qi, 2021). The ultimate result is

$$\frac{1}{2}\sum_{\ell\,\in\,\mathrm{leaf}(\mathcal{T})}\ \sum_{\mathfrak{m}\neq\mathbb{1}} |p^{O}_{\mathfrak{m},\mathfrak{e}}(\ell)| \le O\!\left(\frac{T^7}{d^2}\right) + O\!\left(\frac{T^2}{d}\right). \tag{9.269}$$

Second term for $\mathrm{Sp}(d/2)$ case

A similar proof holds in the symplectic setting. We likewise apply Cauchy-Schwarz to the second term in (9.252) to obtain the upper bound

$$\frac{1}{2}\sum_{\ell\,\in\,\mathrm{leaf}(\mathcal{T})}\ \sum_{\mathfrak{m}\neq\mathfrak{e}} (dd')^T\, W_\ell$$



$$|\mathrm{Wg}^{\mathrm{Sp}}(\sigma_{\mathfrak{m}}^{-1}, d/2)|\,. \tag{9.270}$$

Using the same notation for $C(2i-1,2i)$ as in Eqn. (9.263) above, we further define

$$\widetilde{C}(2i-1,2i) := C(2i-1,2i)\cdot J^t \tag{9.271}$$

and similarly $\widetilde{C}(2i, 2i-1) := \widetilde{C}(2i-1, 2i)^t$. Then the diagrammatic term in Eqn. (9.270) can be written as

$$\left|\operatorname{tr}\left(\Delta'_{\mathfrak{m}} \bigotimes_{i=1}^{T} \widetilde{C}(2i-1, 2i)\right)\right| . \tag{9.272}$$

This can be expanded analogously to (9.263), namely as

$$\Big|\operatorname{tr}\big(\widetilde{C}(2,1)\, J\, C(f_{\mathfrak{m}}(1), f_{\mathfrak{e}} \circ f_{\mathfrak{m}}(1))\, J \cdots J\, \widetilde{C}(f_{\mathfrak{e}} \circ f_{\mathfrak{m}^{-6}}(2), f_{\mathfrak{m}^{-6}}(2))\, J\big) \tag{9.273}$$

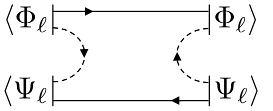
$$\operatorname{tr}\big(\cdots\big) \cdots \operatorname{tr}\big(\cdots\big)\Big| . \tag{9.274}$$

Using the same 1-norm inequality as the orthogonal case, Eqn. (9.273) is upper bounded by

$$\begin{aligned}\prod_{i=1}^{T} \|\widetilde{C}(2i-1, 2i)\, J\|_1 &\leq \prod_{i=1}^{T} \|C(2i-1, 2i)\|_1\, \|J\|_\infty\, \|J^t\|_\infty \\ &= \prod_{i=1}^{T} \operatorname{tr}(C(2i-1, 2i)) . \end{aligned} \tag{9.275}$$

In the first line we have used the Hölder inequality, and in the second line we used $\|J\|_\infty = \|J^t\|_\infty = 1$ as well as $\|C(2i-1, 2i)\|_1 = \operatorname{tr}(C(2i-1, 2i))$. Thus we arrive at an upper bound for (9.270):

$$\frac{1}{2} \sum_{\ell \,\in\, \mathrm{leaf}(\mathcal{T})} \sum_{\mathfrak{m} \neq \mathfrak{e}} (dd')^T\, W_\ell$$



$$|\mathrm{Wg}^{\mathrm{Sp}}(\sigma_{\mathfrak{m}}^{-1}, d/2)| . \tag{9.276}$$

As before we sum over leaves, giving us

$$\frac{d^T}{2} \sum_{\mathfrak{m} \neq \mathfrak{e}} |\mathrm{Wg}^{\mathrm{Sp}}(\sigma_{\mathfrak{m}}^{-1}, d/2)| . \tag{9.277}$$

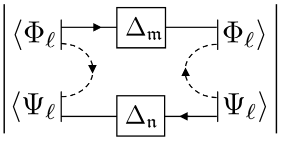
This is is upper bounded by $O(T^{7/2}/d^2) + O(T^2/d)$ using Corollary 19 in combination with Lemma 10 of (Aharonov, J. S. Cotler, and Qi, 2021), and so in the end we obtain

$$\frac{1}{2} \sum_{\ell \,\in\, \mathrm{leaf}(\mathcal{T})} \sum_{\mathfrak{m} \neq \mathbb{1}} |p^{\mathcal{S}}_{\mathfrak{m},\mathfrak{e}}(\ell)| \leq O\left(\frac{T^{7/2}}{d^2}\right) + O\left(\frac{T^2}{d}\right) . \tag{9.278}$$

Third term for $O(d)$ case

The third term in (9.250) is $\frac{1}{2} \sum_{\ell \,\in\, \mathrm{leaf}(\mathcal{T})} \sum_{\mathfrak{n} \neq \mathfrak{e},\, \mathfrak{m}} |p^{O}_{\mathfrak{m},\mathfrak{n}}(\ell)|$. Applying the Cauchy-Schwarz inequality we obtain the upper bound

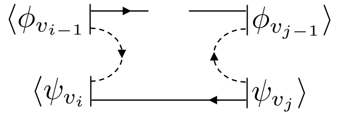
$$\frac{1}{2} \sum_{\ell \,\in\, \mathrm{leaf}(\mathcal{T})} \sum_{\mathfrak{n} \neq \mathfrak{e},\, \mathfrak{m}} (dd')^T\, W_\ell$$



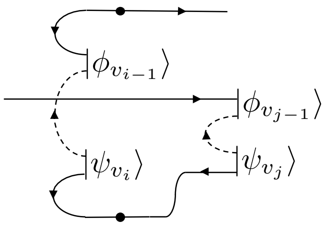
$$|\mathrm{Wg}^{O}(\sigma_{\mathfrak{n}} \sigma_{\mathfrak{m}}^{-1}, d)| . \tag{9.279}$$

Here we generalize our notation for the C matrices by writing

$$C(2i-1,2j) :=$$

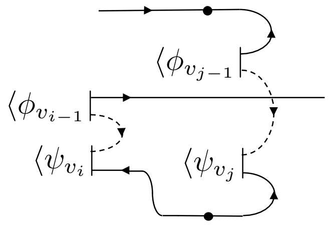


$$C(2i,2j) :=$$



$$, \quad i<j \tag{9.280}$$

$$C(2i-1,2j-1) :=$$



$$, \quad i<j$$

where as before $C(j,i) := C(i,j)^t$. Then the diagrammatic term in (9.279) can be written as

$$\Big|\mathrm{tr}\big(C(f_{\mathfrak{n}}\circ f_{\mathfrak{e}}(1),1)C(f_{\mathfrak{m}}(1),f_{\mathfrak{n}}\circ f_{\mathfrak{e}}\circ f_{\mathfrak{m}}(1)) \tag{9.281}$$

$$C(f_{\mathfrak{m}}\circ f_{\mathfrak{n}}\circ f_{\mathfrak{e}}\circ f_{\mathfrak{m}}(1), f_{\mathfrak{n}}\circ f_{\mathfrak{e}}\circ f_{\mathfrak{m}}\circ f_{\mathfrak{n}}\circ f_{\mathfrak{e}}\circ f_{\mathfrak{m}}(1))\cdots\big)$$

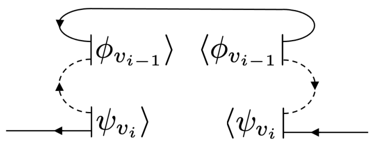
$$\cdot\,\mathrm{tr}\big(\cdots\big)\cdots\mathrm{tr}\big(\cdots\big)\Big| \tag{9.282}$$

and so using the same 1-norm bound as before we obtain the upper bound

$$\prod_{i=1}^{T}\|C(\mathfrak{n}(2i),\mathfrak{n}(2i-1))\|_1\,. \tag{9.283}$$

To simplify (9.283), we define the unnormalized density operator $\widetilde{\rho}_{v_i}\in\mathrm{Mat}_{d\times d}(\mathbb{C})$:

$$\widetilde{\rho}_{v_i} := \tag{9.284}$$



for any $i\in[T]$. Then we have the following Lemma:

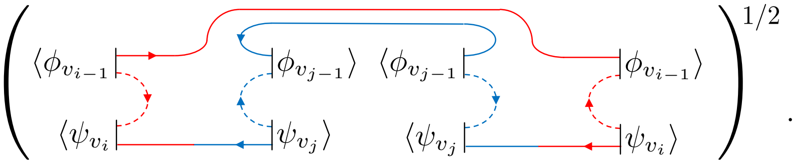
Lemma 63. *For any $i,j\in[T]$,*

$$\|C(2i-1,2j)\|_1 \le \sqrt{tr(\widetilde{\rho}_{v_i}\widetilde{\rho}_{v_j})} \tag{9.285}$$

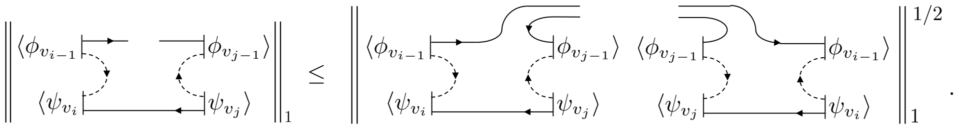
$$\|C(2i,2j)\|_1 \leq \sqrt{tr(\widetilde{\rho}_{v_i}\widetilde{\rho}^t_{v_j})} \tag{9.286}$$

$$\|C(2i-1,2j-1)\|_1 \leq \sqrt{tr(\widetilde{\rho}_{v_i}\widetilde{\rho}^t_{v_j})}\,. \tag{9.287}$$

Proof. First consider $\|C(2i-1,2j)\|_1$. Since $\|A\|_1 = \|A\otimes A^\dagger\|_1^{1/2} \leq \|\mathrm{SWAP}\cdot(A\otimes A^\dagger)\|_1^{1/2}$, we have



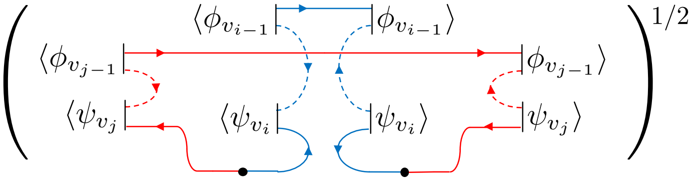
Since the tensor network in the trace is positive semi-definite, we can compute the 1-norm by taking to trace and find



(9.288)

We have colored the tensor diagram suggestively so that it is transparent how to rewrite it as $\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_i}\widetilde{\rho}_{v_j})}$.

Now consider $\|C(2i,2j)\|_1$. Using the same inequality $\|A\|_1 = \|A\otimes A^\dagger\|_1^{1/2} \leq \|\mathrm{SWAP}\cdot(A\otimes A^\dagger)\|_1^{1/2}$, we find the upper bound



(9.289)

which is clearly equal to $\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_i}\widetilde{\rho}^t_{v_j})}$. The upper bound on $\|C(2i-1,2j-1)\|_1$ is given by an identical argument. □

It will be convenient to define the underline notation

$$\underline{i} := \begin{cases} \frac{i+1}{2} & \text{for } i \text{ odd} \\ \frac{i}{2} & \text{for } i \text{ even} \end{cases} \tag{9.290}$$

and analogously for $\underline{j}$. We will say that i,j have same parity if they are equal modulo 2, and have different parity otherwise. Then using the above Lemma, we can upper bound (9.283) by

$$\prod_{m=1}^{\#^O(\mathfrak{n})} \left(\prod_{\substack{i \leftrightarrow j \in B_m \\ i,j \text{ opposite parity}}} \sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}} \widetilde{\rho}_{v_{\underline{j}}})} \prod_{\substack{k \leftrightarrow \ell \in B_m \\ k,\ell \text{ same parity}}} \sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{k}}} \widetilde{\rho}^{t}_{v_{\underline{\ell}}})} \right) . \tag{9.291}$$

Now we turn to bounding (9.291). We begin by developing some notation for cycles B_m . Standard notation for an M_{2T} cycle of "length p" is $B_m = (a_1 a_2 \cdots a_{2p})$ where for us $\{a_1, a_2, ..., a_{2p}\} \subseteq [2T]$. Since the number of elements in an M_{2T}-cycle is always even, by convention we define the length as half the number of elements (i.e. p instead of $2p$). We decorate each a_i by an additional subscript as $a_{m,i}$ to remember that it that belongs to the mth cycle B_m. We sometimes write $p = |B_m|$ to remind ourselves that it depends on m. Let us also define

$$\mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}} \widetilde{\rho}_{v_{\underline{j}}}^{t(i,j)}) := \begin{cases} \mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}} \widetilde{\rho}_{v_{\underline{j}}}) & \text{if } i, j \text{ have different parities} \\ \mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}} \widetilde{\rho}^{t}_{v_{\underline{j}}}) & \text{if } i, j \text{ have the same parity} \end{cases} . \tag{9.292}$$

With these notations in mind, we can write (9.291) as a product over

$$\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,1}}}} \widetilde{\rho}_{v_{\underline{a_{m,2}}}}^{t(a_{m,1},a_{m,2})})} \cdots \sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,2p-1}}}} \widetilde{\rho}_{v_{\underline{a_{m,2p}}}}^{t(a_{m,2p-1},a_{m,2p})})} \sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,2p}}}} \widetilde{\rho}_{v_{\underline{a_{m,1}}}}^{t(a_{m,2p},a_{m,1})})} \tag{9.293}$$

over $m = 1, \ldots, \#(\mathfrak{n})$; here we drop the O superscript on $\#^O$ since we are only discussing pair permutation here. We will further analyze the above for fixed m in two cases: (i) $p = |B_m|$ is even, and (ii) $p = |B_m|$ is odd.

It will be convenient to prove a Lemma which is slightly more general than what we need for the orthogonal case; the advantage of this generality is that it will immediately apply to the symplectic case. The Lemma is as follows:

Lemma 64. *Let $tr(\widetilde{\rho}_{v_{\underline{i}}} \widetilde{\rho}_{v_{\underline{j}}}^{f(i,j)})$ equal either $tr(\widetilde{\rho}_{v_{\underline{i}}} \widetilde{\rho}_{v_{\underline{j}}})$, $tr(\widetilde{\rho}_{v_{\underline{i}}} \widetilde{\rho}^{t}_{v_{\underline{j}}})$, $tr(\widetilde{\rho}_{v_{\underline{i}}} \boldsymbol{J} \widetilde{\rho}^{t}_{v_{\underline{j}}} \boldsymbol{J}^{-1})$, depending on the value of i, j. Defining*

$$R_{m,-} := tr(\widetilde{\rho}_{v_{\underline{a_{m,p}}}}) \prod_{\substack{i \ odd \\ 1 \le i \le 2p-2}} tr(\widetilde{\rho}_{v_{\widetilde{a}_{m,i}}} \widetilde{\rho}_{v_{\underline{a_{m,i+1}}}}^{f(a_{m,i},a_{m,i+1})}) \tag{9.294}$$

$$R_{m,+} := tr(\widetilde{\rho}_{v_{\underline{a_{m,1}}}}) \prod_{j \ even} tr(\widetilde{\rho}_{v_{\underline{a_{m,j}}}} \widetilde{\rho}_{v_{\underline{a_{m,j+1}}}}^{f(a_{m,j},a_{m,j+1})}) , \tag{9.295}$$

we have the inequality

$$\prod_{m=1}^{\#(\mathfrak{n})} \prod_{i \leftrightarrow j \in B_m} \sqrt{tr(\widetilde{\rho}_{v_{\underline{i}}} \widetilde{\rho}_{v_{\underline{j}}}^{f(i,j)})} \le \frac{1}{2^{\#(\mathfrak{n})}} \sum_{i_1, \ldots, i_{\#(\mathfrak{n})} = \pm} R_{1,i_1} R_{2,i_2} \cdots R_{\#(\mathfrak{n}),i_{\#(\mathfrak{n})}} . \tag{9.296}$$

Proof. Similar to the unitary setting, the argument proceeds in two cases.

Case 1: *p is even.* Every term in Eqn. (9.293) has the form

$$\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,i}}}}\widetilde{\rho}_{v_{\underline{a_{m,i+1}}}}^{f(\underline{a_{m,i}},\underline{a_{m,i+1}})})} \tag{9.297}$$

except the last term. The i subscripts of $\widetilde{a}_{m,i}$ will be treated modulo $2p$, and $\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,2p}}}}\widetilde{\rho}_{v_{\underline{a_{m,2p+1}}}}^{f(\underline{a_{m,2p}},\underline{a_{m,2p+1}})})}$. With this notation at hand, we organize Eqn. (9.293) as

$$\left(\prod_{i\text{ odd}}\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,i}}}}\widetilde{\rho}_{v_{\underline{a_{m,i+1}}}}^{f(\underline{a_{m,i}},\underline{a_{m,i+1}})})}\right)\left(\prod_{j\text{ even}}\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,j}}}}\widetilde{\rho}_{v_{\underline{a_{m,j+1}}}}^{f(\underline{a_{m,j}},\underline{a_{m,j+1}})})}\right). \tag{9.298}$$

Since $ab \leq \frac{1}{2}(a^2+b^2)$, the expression above is upper bounded by

$$\frac{1}{2}\prod_{i\text{ odd}}\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,i}}}}\widetilde{\rho}_{v_{\underline{a_{m,i+1}}}}^{f(\underline{a_{m,i}},\underline{a_{m,i+1}})})+\frac{1}{2}\prod_{j\text{ even}}\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,j}}}}\widetilde{\rho}_{v_{\underline{a_{m,j+1}}}}^{f(\underline{a_{m,j}},\underline{a_{m,j+1}})}). \tag{9.299}$$

We will call the first term $\frac{1}{2}R_{m,-}$ and the second term $\frac{1}{2}R_{m,+}$.

Case 2: *p is odd.* In this setting we can arrange the terms in Eqn. (9.226) as

$$\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,p}}}}\widetilde{\rho}_{v_{\underline{a_{m,1}}}}^{f(a_{m,p},a_{m,1})})}\left(\prod_{\substack{i\text{ odd}\\1\leq i\leq 2p-2}}\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,i}}}}\widetilde{\rho}_{v_{\underline{a_{m,i+1}}}}^{f(a_{m,i},a_{m,i+1})})}\right) \tag{9.300}$$

$$\left(\prod_{j\text{ even}}\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,j}}}}\widetilde{\rho}_{v_{\underline{a_{m,j+1}}}}^{f(a_{m,j},a_{m,j+1})})}\right). \tag{9.301}$$

Since for A, B Hermitian and positive semi-definite we have $\mathrm{tr}(AB) \leq \|A\|_2\,\|B\|_2 \leq \|A\|_1\,\|B\|_1 \leq \mathrm{tr}(A)\,\mathrm{tr}(B)$ it follows that

$$\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,p}}}}\widetilde{\rho}_{v_{\underline{a_{m,1}}}}^{f(a_{m,p},a_{m,1})})} \leq \sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,p}}}})\,\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,1}}}}^{f(a_{m,p},a_{m,1})})} = \sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,p}}}})\,\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,1}}}})}. \tag{9.302}$$

Using the above inequality, (9.300) has the upper bound

$$\left(\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,p}}}})}\prod_{\substack{i\text{ odd}\\1\leq i\leq 2p-2}}\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,i}}}}\widetilde{\rho}_{v_{\underline{a_{m,i+1}}}}^{f(a_{m,i},a_{m,i+1})})}\right) \tag{9.303}$$

$$\left(\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,1}}}})}\prod_{j\text{ even}}\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,j}}}}\widetilde{\rho}_{v_{\underline{a_{m,j+1}}}}^{f(a_{m,j},a_{m,j+1})})}\right). \tag{9.304}$$

Further using $ab \leq \frac{1}{2}(a^2 + b^2)$, we find the upper bound

$$\frac{1}{2}\,\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,p}}}}) \prod_{\substack{i \text{ odd} \\ 1 \leq i \leq 2p-2}} \mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,i}}}} \widetilde{\rho}_{v_{\underline{a_{m,i+1}}}}^{f(a_{m,i},a_{m,i+1})}) + \frac{1}{2}\,\mathrm{tr}(\widetilde{\rho}_{v_{a_{m,1}}}) \tag{9.305}$$

$$\prod_{j \text{ even}} \mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,j}}}} \widetilde{\rho}_{v_{\underline{a_{m,j+1}}}}^{f(a_{m,j},a_{m,j+1})}) \tag{9.306}$$

where the first term is called $\frac{1}{2}\,R_{m,\,-}$ and the second term is called $\frac{1}{2}\,R_{m,\,+}$.

Since $R_{m,\,-}$ and $R_{m,\,+}$ implicitly depend on the leaf ℓ, sometimes we will denote them by $R^{\ell}_{m,\,-}$ and $R^{\ell}_{m,\,+}$ to be explicit.

Taking Case 1 and Case 2 together, we find that Eqn. (9.291) has the upper bound

$$\begin{aligned}\prod_{m=1}^{\#(\mathfrak{n})} \prod_{i \leftrightarrow j\, \in B_m} \sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}} \widetilde{\rho}_{v_{\underline{j}}}^{f(i,j)})} &\leq \frac{1}{2^{\#(\mathfrak{n})}} \prod_{m=1}^{\#(\mathfrak{n})} \left(R_{m,-} + R_{m,+}\right) \\ &= \frac{1}{2^{\#(\mathfrak{n})}} \sum_{i_1,\ldots,i_{\#(\mathfrak{n})}=\pm} R_{1,i_1} R_{2,i_2} \cdots R_{\#(\mathfrak{n}),i_{\#(\mathfrak{n})}}\,. \end{aligned} \tag{9.307}$$

This is the desired bound. □

Observe that each term in the sum in the last line of (9.307) is a product of terms like $\mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}})$ and $\mathrm{tr}(\widetilde{\rho}_{v_{\underline{j}}} \widetilde{\rho}_{v_{\underline{k}}}^{f(j,k)})$. As before, the key point is that we arranged the equations so that each term in the sum has $\widetilde{\rho}_{v_i}$ for each $i = 1, ..., T$ appear *exactly once*. This allows us to prove the following lemma, which is akin to Lemma 62 above:

Lemma 65. *Fix any* $i_1, ..., i_{\#(\mathfrak{n})} \in \{+, -\}$. *Then*

$$\sum_{\ell \,\in\, \mathit{leaf}(\mathcal{T})} (dd')^T\, W_\ell\, R^{\ell}_{1,i_1} R^{\ell}_{2,i_2} \cdots R^{\ell}_{\#(\tau^{-1}),i_{\#(\mathfrak{n})}} \leq d^{T-\left\lfloor \frac{L(\mathfrak{n})}{2} \right\rfloor} \tag{9.308}$$

where $L(\mathfrak{n})$ *is the length of the longest cycle in* $\mathfrak{n}$ *(where we recall that the length of an* M_{2T} *cycle is defined as half the number of integers in that cycle).*

Proof. To ease notation, let $R^{\ell}_j \triangleq R^{\ell}_{j,i_j}$. Recall that $W_\ell = w_{v_1} w_{v_2} \cdots w_{v_T}$ and note that

$$\sum_{v:\,\mathrm{depth}(v)=i} dd'\, w_v\, \widetilde{\rho}_v = \langle \phi_{\mathrm{parent}(v)} | \phi_{\mathrm{parent}(v)} \rangle\, \mathbb{1}_{d\times d} = \mathbb{1}_{d\times d} \tag{9.309}$$

$$\sum_{v:\,\mathrm{depth}(v)=i} dd'\, w_v\, \widetilde{\rho}_v^{\,t} = \langle \phi_{\mathrm{parent}(v)} | \phi_{\mathrm{parent}(v)} \rangle\, \mathbb{1}_{d\times d} = \mathbb{1}_{d\times d} \tag{9.310}$$

$$\sum_{v:\,\mathrm{depth}(v)=i} dd'\,w_v\,J\widetilde{\rho}_v^{\,t}J^{-1} = \langle \phi_{\mathrm{parent}(v)}|\phi_{\mathrm{parent}(v)}\rangle\,\mathbb{1}_{d\times d} = \mathbb{1}_{d\times d}\,. \tag{9.311}$$

Taking the traces of the above equations against any $\rho \in \mathrm{Mat}_{d\times d}(\mathbb{C})$, we find

$$\sum_{v:\,\mathrm{depth}(v)=i} dd'\,w_v\,\mathrm{tr}(\rho\,\widetilde{\rho}_v) = \mathrm{tr}(\rho) \tag{9.312}$$

$$\sum_{v:\,\mathrm{depth}(v)=i} dd'\,w_v\,\mathrm{tr}(\rho\,\widetilde{\rho}_v^{\,t}) = \mathrm{tr}(\rho) \tag{9.313}$$

$$\sum_{v:\,\mathrm{depth}(v)=i} dd'\,w_v\,\mathrm{tr}(\rho\,J\widetilde{\rho}_v^{\,t}J^{-1}) = \mathrm{tr}(\rho)\,. \tag{9.314}$$

In particular, for $\rho = \mathbb{1}$ we have

$$\sum_{v:\,\mathrm{depth}(v)=i} dd'\,w_v\,\mathrm{tr}(\widetilde{\rho}_v) = d. \tag{9.315}$$

With these various identities in mind, we can now turn to bounding (9.308). As we discussed, above $\prod_j R^\ell_j$ is a product of terms like $\mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}})$ and $\mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}}\widetilde{\rho}_{v_{\underline{i'}}}^{f(i,i')})$. It is convenient to define some sets of indices to encode this data. Let $S_1^{(T)} \subseteq [T]$ denote the indices $\underline{i}$ for which $\mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}})$ appears in $\prod_j R^\ell_j$, and also let $S_2^{(T)} \subseteq [T]\times[T]$ denote the set of (unordered) pairs $(\underline{i},\underline{i'})$ for which $\mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}}\widetilde{\rho}_{v_{\underline{i'}}}^{f(i,i')})$ appears, so that for any root-to-leaf path in $\mathcal{T}$ consisting of nodes $v_1,\ldots,v_T = \ell$, we have

$$\prod_j R^\ell_j = \prod_{\underline{i}\in S_1^{(T)}} \mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}}) \cdot \prod_{(\underline{i},\underline{i'})\in S_2^{(T)}} \mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}}\widetilde{\rho}_{v_{\underline{i'}}}^{f(i,i')})\,. \tag{9.316}$$

Next we construct $S_1^{(t)} \subseteq [t]$ and $S_2^{(t)} \subseteq [t]\times[t]$ for $1 \le t < T$ by the following inductive procedure. If $t \in S_1^{(t)}$ then we define $S_1^{(t-1)} \triangleq S_1^{(t)}\backslash\{t\}$ and $S_2^{(t-1)} \triangleq S_2^{(t)}$. Otherwise if $(t,t') \in S_2^{(t)}$ for some $t' < t$, then we define $S_1^{(t-1)} \triangleq S_1^{(t-1)} \cup \{t'\}$ and $S_2^{(t-1)} \triangleq S_2^{(t)}\backslash\{(t,t')\}$. We recall two key observations about such sequences, which we also leveraged in Lemma 62:

Observation 3. *For every $i \in S_1^{(T)}$, we have that $i \in S_1^{(i)}$.*

Observation 4. *For every $(i,i') \in S_2^{(T)}$, if $i \le i'$ then $i \in S_1^{(i)}$ while $i' \notin S_1^{(i')}$.*

We have defined this set of sequences in order to extract $\widetilde{\rho}_{v_T}$ from the product on the right-hand side of (9.316) and apply (9.315) (respectively (9.312)) if $T \in S_1^{(T)}$

(respectively $(T, t') \in S_2^{(T)}$ for some $t' < T$) to obtain

$$\sum_{\ell \in \mathrm{leaf}(\mathcal{T})} (dd')^T W_\ell \prod_j R_j^\ell \tag{9.317}$$

$$= d^{\mathbb{1}[T \in S_1^{(T)}]} \sum_{u\colon \mathrm{depth}(u) = T-1} (dd')^{T-1} W_u \prod_{\underline{i} \in S_1^{(T-1)}} \mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}}) \cdot \prod_{(i,i') \in S_2^{(T-1)}} \mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}} \widetilde{\rho}_{v_{\underline{i'}}}^{f(i,i')}) \,, \tag{9.318}$$

with $W_u = w_{v_1} \cdots w_{v_{T-1}}$ if the path from root to u in $\mathcal{T}$ is given by $v_1, \ldots, v_{T-1} = u$. We can proceed inductively by expressing the right-hand side of (9.317) as

$$d^{\sum_{t=1}^T \mathbb{1}[t \in S_1^{(t)}]}. \tag{9.319}$$

By virtue of Observations 3 and 4, $\sum_{t=1}^T \mathbb{1}[t \in S_1^{(t)}] = |S_1^{(T)}| + |S_2^{(T)}|$. Since every M_{2T}-cycle B_m of $\mathfrak{n}$ contributes $|B_m|/2$ pairs to $S_2^{(T)}$, and every M_{2T}-cycle B_m with $|B_m|$ even contributes $\lfloor |B_m|/2 \rfloor$ pairs to $S_2^{(T)}$ and one element to $S_1^{(T)}$, we conclude the formula

$$\begin{aligned} \sum_{\ell \in \mathrm{leaf}(\mathcal{T})} (dd')^T W_\ell \prod_j R_j^\ell &= d^{\sum_{m=1}^{\#(\mathfrak{n})} \left\lceil \frac{|B_m|}{2} \right\rceil} \\ &= d^{T - \sum_{m=1}^{\#(\mathfrak{n})} \left\lfloor \frac{|B_m|}{2} \right\rfloor} \\ &\leq d^{T - \left\lfloor \frac{L(\mathfrak{n})}{2} \right\rfloor} \end{aligned} \tag{9.320}$$

as we desired. □

Putting all our previous bounds together, in particular Eqn.'s (9.279), (9.291), (9.307), and (9.308), we arrive at

$$\sum_{\ell \in \mathrm{leaf}(\mathcal{T})} \sum_{\mathfrak{n} \neq \mathfrak{e},\, \mathfrak{m}} |p^O_{\sigma,\tau}(\ell)| \leq d^T \sum_{\mathfrak{m}} |\mathrm{Wg}^O(\sigma_{\mathfrak{m}}^{-1}, d)| \sum_{\mathfrak{n} \neq \mathfrak{e}} d^{-\left\lfloor \frac{L^O(\mathfrak{n})}{2} \right\rfloor}. \tag{9.321}$$

The first sum on the right-hand side is bounded by $d^T \sum_{\mathfrak{m}} |\mathrm{Wg}^O(\sigma_{\mathfrak{m}}^{-1}, d)| \leq 1 + O\left(\frac{T^2}{d}\right)$. Considering the second sum on the right-hand side, let $N^O(T, \ell)$ be the number of permutations in S_T where the length of the longest M_{2T}-cycle is 2ℓ. Then the second sum can be written as

$$\sum_{\ell=2}^T N^O(T, \ell)\, d^{-\lfloor \frac{\ell}{2} \rfloor} \tag{9.322}$$

where we omit $\ell = 1$ from the sum since it corresponds to the identity pair permutation. Since $N^O(T,\ell) \leq \binom{2T}{2\ell}(2\ell-1)!! = \frac{(2T)!}{(2T-2\ell)!\,2^\ell \ell!} < T^\ell$, (9.322) is upper bounded by

$$\sum_{\ell=2}^{\infty} T^\ell\, d^{-\lfloor \frac{\ell}{2} \rfloor} = \frac{(1+T)\frac{T^2}{d}}{1-\frac{T^2}{d}} = \frac{T^3}{d} + \frac{T^2}{d} + O\!\left(\frac{T^5}{d^2}\right) . \tag{9.323}$$

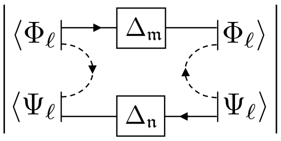
Now if $T \leq o(d^{1/3})$, then this quantity is $o(1)$ for some absolute constant $c > 0$. Altogether, we find

$$\frac{1}{2} \sum_{\ell \,\in\, \mathrm{leaf}(\mathcal{T})} \sum_{\mathfrak{n} \neq \mathfrak{e},\, \mathfrak{m}} |p^{O}_{\mathfrak{m},\mathfrak{n}}(\ell)| \leq o(1) \ . \tag{9.324}$$

Third term for $\mathrm{Sp}(d/2)$ **case**

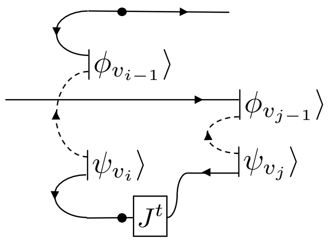
The third term in (9.252) is $\frac{1}{2}\sum_{\ell \,\in\, \mathrm{leaf}(\mathcal{T})} \sum_{\mathfrak{n} \neq \mathfrak{e},\, \mathfrak{m}} |p^{S}_{\mathfrak{m},\mathfrak{n}}(\ell)|$. Applying the Cauchy-Schwarz inequality we obtain the upper bound

$$\frac{1}{2} \sum_{\ell \,\in\, \mathrm{leaf}(\mathcal{T})} \sum_{\mathfrak{n} \neq \mathfrak{e},\, \mathfrak{m}} (dd')^T\, W_\ell \qquad |\mathrm{Wg}^O(\sigma_{\mathfrak{n}}\sigma_{\mathfrak{m}}^{-1}, d)| \, . \tag{9.325}$$



To bound the diagrammatic term, we introduce the tilde notation $\widetilde{C}(2i-1,2j) := C(2i-1,2j)$, $\widetilde{C}(2i-1,2j-1) := C(2i-1,2j-1)$ as well as

$$\widetilde{C}(2i,2j) := \qquad , \quad i < j \tag{9.326}$$



where as usual $\widetilde{C}(j,i) = \widetilde{C}(i,j)^t$. Then the diagrammatic term in (9.325) can be written as

$$\Big|\mathrm{tr}\big(C(f_{\mathfrak{n}} \circ f_{\mathfrak{e}}(1), 1)\, J\, C(f_{\mathfrak{m}}(1), f_{\mathfrak{n}} \circ f_{\mathfrak{e}} \circ f_{\mathfrak{m}}(1)) \tag{9.327}$$

$$J\, C(f_{\mathfrak{m}} \circ f_{\mathfrak{n}} \circ f_{\mathfrak{e}} \circ f_{\mathfrak{m}}(1), f_{\mathfrak{n}} \circ f_{\mathfrak{e}} \circ f_{\mathfrak{m}} \circ f_{\mathfrak{n}} \circ f_{\mathfrak{e}} \circ f_{\mathfrak{m}}(1))\, J \cdots\big)$$

$$\cdot\, \mathrm{tr}\big(\cdots\big) \cdots \mathrm{tr}\big(\cdots\big)\Big| \tag{9.328}$$

which is upper bounded in the 1-norm by

$$\prod_{i=1}^{T} \|\widetilde{C}(\mathfrak{n}(2i), \mathfrak{n}(2i-1)\, J)\|_1 \leq \prod_{i=1}^{T} \|\widetilde{C}(\mathfrak{n}(2i), \mathfrak{n}(2i-1))\|_1\, \|J\|_\infty$$

$$\leq \prod_{i=1}^{T} \|\widetilde{C}(\mathfrak{n}(2i), \mathfrak{n}(2i-1))\|_1 . \tag{9.329}$$

We now have the following Lemma, analogous to Lemma 63:

Lemma 66. *For any $i, j \in [T]$,*

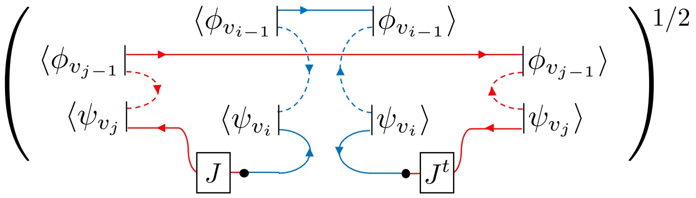
$$\|\widetilde{C}(2i-1, 2j)\|_1 \leq \sqrt{tr(\widetilde{\rho}_{v_i}\widetilde{\rho}_{v_j})} \tag{9.330}$$

$$\|\widetilde{C}(2i, 2j)\|_1 \leq \sqrt{tr(\widetilde{\rho}_{v_i}\widetilde{\rho}_{v_j}^{D})} \tag{9.331}$$

$$\|\widetilde{C}(2i-1, 2j-1)\|_1 \leq \sqrt{tr(\widetilde{\rho}_{v_i}\widetilde{\rho}_{v_j}^{t})} \tag{9.332}$$

where we recall that $A^D := JA^tJ^{-1}$ is the symplectic transpose.

Proof. The first and third inequalities follow from Lemma 63 since $\widetilde{C}(2i-1, 2j) = C(2i-1, 2j)$ and $\widetilde{C}(2i-1, 2j-1) = C(2i-1, 2j-1)$. For the second inequality, we again use $\|A\|_1 = \|A \otimes A^\dagger\|_1^{1/2} \leq \|\mathrm{SWAP} \cdot (A \otimes A^\dagger)\|_1^{1/2}$ to find the upper bound



which evidently equals $\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_i}\widetilde{\rho}_{v_j}^{D})}$. □

Leveraging the above Lemma, we can upper bound (9.329) by

$$\prod_{m=1}^{\#^O(\mathfrak{n})} \left(\prod_{\substack{i \leftrightarrow j \in B_m \\ i,j \text{ opposite parity}}} \sqrt{\mathrm{tr}(\widetilde{\rho}_{v_i}\widetilde{\rho}_{v_j})} \prod_{\substack{k \leftrightarrow \ell \in B_m \\ k,\ell \text{ both odd}}} \sqrt{\mathrm{tr}(\widetilde{\rho}_{v_k}\widetilde{\rho}_{v_\ell}^{t})} \prod_{\substack{p \leftrightarrow q \in B_m \\ p,q \text{ both even}}} \sqrt{\mathrm{tr}(\widetilde{\rho}_{v_p}\widetilde{\rho}_{v_q}^{D})} \right). \tag{9.333}$$

Defining

$$\mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}}\widetilde{\rho}_{v_{\underline{j}}}^{D(i,j)}) := \begin{cases} \mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}}\widetilde{\rho}_{v_{\underline{j}}}) & \text{if } i, j \text{ have different parities} \\ \mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}}\widetilde{\rho}_{v_{\underline{j}}}^{t}) & \text{if } i, j \text{ are both odd} \\ \mathrm{tr}(\widetilde{\rho}_{v_{\underline{i}}}\widetilde{\rho}_{v_{\underline{j}}}^{D}) & \text{if } i, j \text{ are both even} \end{cases}, \tag{9.334}$$

we can write (9.333) as a product over

$$\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,1}}}}\widetilde{\rho}_{v_{\underline{a_{m,2}}}}^{D(a_{m,1},a_{m,2})})} \cdots \sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,p-1}}}}\widetilde{\rho}_{v_{\underline{a_{m,p}}}}^{D(a_{m,p-1},a_{m,p})})} \tag{9.335}$$

$$\sqrt{\mathrm{tr}(\widetilde{\rho}_{v_{\underline{a_{m,p}}}}\widetilde{\rho}^{D(a_{m,p},a_{m,p-1})}_{v_{\underline{a_{m,1}}}})} \tag{9.336}$$

over $m = 1, \ldots, \#^{\mathrm{Sp}}(\mathfrak{n})$. Leveraging Lemmas 64 and 65 in the same exact was as in the orthogonal case, we obtain

$$\sum_{\ell \in \mathrm{leaf}(\mathcal{T})} \sum_{\mathfrak{n}\neq\mathfrak{e},\,\mathfrak{m}} |p^{S}_{\sigma,\tau}(\ell)| \leq d^T \sum_{\mathfrak{m}} |\mathrm{Wg}^{\mathrm{Sp}}(\sigma_{\mathfrak{m}}^{-1}, d/2)| \sum_{\mathfrak{n}\neq\mathfrak{e}} d^{-\left\lfloor \frac{L(\mathfrak{n})}{2} \right\rfloor}. \tag{9.337}$$

The first sum on the right-hand side is bounded by $d^T \sum_{\mathfrak{m}} |\mathrm{Wg}^{\mathrm{Sp}}(\sigma_{\mathfrak{m}}^{-1}, d/2)| \leq 1 + O\left(\frac{T^2}{d}\right)$. Considering the second sum on the right-hand side, let $N^{\mathrm{Sp}}(T, \ell)$ be the number of permutations in S_T where the length of the longest M_{2T}-cycle is 2ℓ. Since $N^{\mathrm{Sp}}(T, \ell) = N^{O}(T, \ell)$, the second sum can be written as

$$\sum_{\ell=2}^{T} N^{\mathrm{Sp}}(T, \ell)\, d^{-\lfloor \frac{\ell}{2} \rfloor} \leq \frac{T^3}{d} + \frac{T^2}{d} + O\left(\frac{T^5}{d^2}\right). \tag{9.338}$$

Now if $T \leq o(d^{1/3})$, then this quantity is $o(1)$ for some absolute constant $c > 0$. Altogether, we find

$$\frac{1}{2} \sum_{\ell \in \mathrm{leaf}(\mathcal{T})} \sum_{\mathfrak{n}\neq\mathfrak{e},\,\mathfrak{m}} |p^{S}_{\mathfrak{m},\mathfrak{n}}(\ell)| \leq o(1)\,. \tag{9.339}$$

This concludes the proof. □

Corollaries involving state distinction

Here we remark on some immediate corollaries of Theorems 57, 58, and 59. We begin with a corollary essentially identical to one from (Aharonov, J. S. Cotler, and Qi, 2021):

Corollary 20. *Any learning algorithm without quantum memory requires*

$$T \geq \Omega\left(2^{n/3}\right), \tag{9.340}$$

to correctly distinguish between the maximally mixed state $\mathbb{1}/2^n$ *on* n *qubits and a fixed, Haar-random state* $|\Psi\rangle\langle\Psi|$ *on* n *qubits with probability at least* $2/3$.

Proof. Suppose by contradiction that a learning algorithm could distinguish between $\mathbb{1}/d$ and a fixed, Haar-random $|\Psi\rangle\langle\Psi|$ with probability at least $2/3$ using $T < O(d^{1/3})$. Then we could use this learning algorithm to solve the unitary distinction problem by taking the channel C and applying it to $|0\rangle^{\otimes n}$; if $C = \mathcal{D}$ then we would get the maximally mixed state, and if $C = \mathcal{U}$ we would get a fixed Haar random state. Moreover we could distinguish the two cases in using $T < O(d^{1/3})$ by running the alleged state distinction algorithm. But this is impossible since it contradicts Theorem 57, so such a state distinction algorithm can not exist. □

While the above corollary is weaker than Theorem 52 for which $T \geq \Omega(d^{1/2})$, it is emblematic of a general strategy for using learning bounds on channel distinction problems to derive corresponding learning bounds on state distinction problems. Further leveraging this strategy, we can prove the following two additional corollaries:

Corollary 21. *Any learning algorithm without quantum memory requires*

$$T \geq \Omega\left(2^{n/3}\right), \tag{9.341}$$

to correctly distinguish between the maximally mixed state $\mathbb{1}/2^n$ *on n qubits and a fixed, real Haar random state* $|\Psi\rangle\langle\Psi|$ *on n qubits with probability at least* $2/3$.

To prove this corollary, we observe that $|\Psi\rangle\langle\Psi| = O\left(|0\rangle\langle 0|\right)^{\otimes n} O^t$ for a fixed, Haar-random orthogonal matrix; alternatively, it is also the case that $|\Psi\rangle\langle\Psi| = S\left(|0\rangle\langle 0|\right)^{\otimes n} S^D$ for a fixed, Haar-random symplectic matrix. Leveraging the symplectic version (in particular since Theorem 58 has a stronger bound than Theorem 59), the corollary follows by the same arguments as the proof of Corollary 20 in combination with Theorem 59. Corollary 21 also follow from the results in (Aharonov, J. S. Cotler, and Qi, 2021), although the corollary was not stated there.

Upper bound without quantum memory

The upper bound without quantum memory can be obtained by reducing the problem to purity testing and utilizing Theorem 53. This results in the following corollary.

Corollary 22. *There is a learning algorithm without quantum memory which takes* $T = O(2^{n/2})$ *accesses to the unknown quantum channel* C *to distinguish between whether* C *is a fixed Haar-random unitary channel or a completely depolarizing channel* $\mathcal{D}$.

Proof. We perform T repeated experiments given by the following. Input the all-zero state $|0^n\rangle$ to the unknown quantum channel C to obtain the output state ρ_{out}.

When C is a scrambling unitary channel, ρ_{out} is a fixed pure state. When C is a completely depolarizing channel, ρ_{out} is the completely mixed state. We then measure the output state ρ_{out} in the computational basis to obtain the classical data. The collection of classical data given by the computational basis measurements can be used to classify if ρ_{out} is a fixed pure state or the completely mixed state. Theorem 53 tells us that $T = O(2^{n/2})$ is sufficient to distinguish between the two cases. Hence, we can distinguish between whether C is a scrambling unitary channel or a completely depolarizing channel using $T = O(2^{n/2})$. □

Upper bound with quantum memory

The exponential lower bound for algorithms without quantum memory is in stark contrast to those with quantum memory. The following result from (Aharonov, J. S. Cotler, and Qi, 2021) states that a linear number of channel applications T and quantum gates suffices if we use a learning algorithm with quantum memory.

Theorem 60 (Fixed unitary task is easy with an n qubit quantum memory (Aharonov, J. S. Cotler, and Qi, 2021))**.** *There exists a learning algorithm with n qubits of quantum memory which, with constant probability, can distinguish a completely depolarizing channel $\mathcal{D}$ from a fixed, Haar-random $\mathcal{U}$ (either unitary, orthogonal, or symplectic) using only $T = O(1)$ applications of the channel. Moreover, the algorithm is gate efficient and has $O(n)$ gate complexity.*

The protocol is simply a swap test; the basic idea is that for a pure state $|\phi\rangle$ on n qubits we have $\text{tr}(\mathcal{D}[|\phi\rangle\langle\phi|]^2) = \frac{1}{d}$ whereas $\text{tr}(\mathcal{U}[|\phi\rangle\langle\phi|]^2) = 1$. The ability to obtain quantum interference between $\mathcal{D}[|\phi\rangle\langle\phi|]$ and a copy of itself $\mathcal{D}[|\phi\rangle\langle\phi|]$ (or $\mathcal{U}[|\phi\rangle\langle\phi|]$ and a copy of itself $\mathcal{U}[|\phi\rangle\langle\phi|]$) is enabled by the n qubit quantum memory which can store a single copy of the state.

Symmetry distinction problem

Lower bound

Using Theorem 57, 58, and 59, we can show the hardness of distinguishing between unitary channel, orthogonal matrix channel, and symplectic matrix channel for learning algorithms without quantum memory. This multiple-hypothesis distinguishing task is equivalent to uncovering what symmetry is encoded in a quantum evolution. Orthogonal matrix channel and symplectic matrix channel are quantum evolutions with different type of time-reversal symmetry, while unitary channel is a general evolution without additional symmetry.

Theorem 61. *Any learning algorithm without quantum memory requires*

$$T \geq \Omega\left(2^{2n/7}\right), \tag{9.342}$$

to correctly distinguish between a fixed, Haar-random unitary channel C^U, orthogonal matrix channel C^O, or symplectic matrix channel C^S on n qubits with probability at least $2/3$.

Proof. Given a tree representation $\mathcal{T}$ of the learning algorithm without quantum memory. The probability that the algorithm correctly identifies the class of channels is equal to

$$\frac{1}{3} \sum_{C \in \{C^U, C^O, C^S\}} \sum_{\ell \in \mathrm{leaf}(\mathcal{T})} p^C(\ell)\ \mathrm{I}[\Upsilon(\ell) = C], \tag{9.343}$$

where $\Upsilon(\ell)$ is an element in the set $\{C^U, C^O, C^S\}$ equal to the output when the algorithm arrives at the leaf ℓ. It is not hard to see that the success probability is upper bounded by

$$\frac{1}{3} \sum_{\ell \in \mathrm{leaf}(\mathcal{T})} \max\left(p^{C^U}(\ell), p^{C^O}(\ell), p^{C^S}(\ell)\right) \tag{9.344}$$

$$\leq \frac{1}{3} \sum_{\ell \in \mathrm{leaf}(\mathcal{T})} p^{\mathcal{D}}(\ell) + \max_{C \in \{C^U, C^O, C^S\}} \left(\left|p^C(\ell) - p^{\mathcal{D}}(\ell)\right|\right) \tag{9.345}$$

$$\leq \frac{1}{3} + \frac{1}{3} \sum_{C \in \{C^U, C^O, C^S\}} \left(\sum_{\ell \in \mathrm{leaf}(\mathcal{T})} \left|p^C(\ell) - p^{\mathcal{D}}(\ell)\right| \right). \tag{9.346}$$

From the proof of Theorem 57, 58, and 59, we have when $T = o(2^{2n/7})$, we have

$$\sum_{\ell \in \mathrm{leaf}(\mathcal{T})} \left|p^C(\ell) - p^{\mathcal{D}}(\ell)\right| = o(1), \quad \forall C \in \{C^U, C^O, C^S\}. \tag{9.347}$$

Therefore, when $T = o(2^{2n/7})$, the success probability will be upper bounded by $1/3 + o(1)$. Hence to achieve a success probability of at least $2/3$, we must have $T = \Omega(2^{2n/7})$. □

Upper bound

We present an upper bound for algorithms without quantum memory. There is still a gap between the lower and upper bounds, which we leave as an open question.

Theorem 62. *There is an algorithm without quantum memory that uses*

$$T = O\left(2^n\right), \tag{9.348}$$

to correctly distinguish between a fixed, Haar-random unitary channel C^U, orthogonal matrix channel C^O, or symplectic matrix channel C^S on n qubits with probability at least 2/3.

We will perform three sets of experiments. Each set of experiments is a quantum state tomography based on random Clifford measurements (Richard Kueng, Rauhut, and Terstiege, 2017; Guţă et al., 2020b) on the output state $C(|\psi_i\rangle\langle\psi_i|)$ for an input pure state $|\psi_i\rangle$, for $i = 1, 2, 3$. We will take $|\psi_1\rangle = |0^n\rangle$, $|\psi_2\rangle = \frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)\otimes|0^{n-1}\rangle$, and $|\psi_3\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \otimes |0^{n-1}\rangle$. Note that for each i, $C(|\psi_i\rangle\langle\psi_i|)$ is a pure state, which we will denote by $\phi_i \in \mathbb{C}^{2^n}$.

We will use the following guarantee:

Lemma 67 (State tomography, see (Richard Kueng, Rauhut, and Terstiege, 2017; Guţă et al., 2020b))**.** *There is an algorithm which for any $\epsilon > 0$, given copies of an unknown pure state $|\phi\rangle$ with density matrix $\rho \in \mathbb{H}^{2^n \times 2^n}$, makes $O(2^n/\epsilon^2)$ random Clifford measurements and outputs the density matrix $\widehat{\rho}$ of some pure state for which $\|\rho - \widehat{\rho}\|_{\mathrm{tr}} \leq \epsilon$ with probability at least* 14/15.

Corollary 23. *Suppose $\widehat{\rho_i} = |\widehat{\phi_i}\rangle\langle\widehat{\phi_i}|$ is the output of applying the algorithm in Lemma 67 to $\rho = |\phi_i\rangle\langle\phi_i|$. Then provided the algorithm succeeds, we have that for any matrix $M \in \mathbb{C}^{2^n \times 2^n}$ and any $i, j \in \{1, 2, 3\}$*

$$\left\|\widehat{\phi}_i^t M \widehat{\phi}_j\right| - \left|\phi_i^t M \phi\right\| \leq 2\epsilon \, \|M\|_\infty . \tag{9.349}$$

Proof. Because $\|\rho_i - \widehat{\rho}_i\|_F \leq \|\rho_i - \widehat{\rho}_i\|_{\mathrm{tr}} \leq \epsilon$, we have that $\left\|\phi_i - \zeta \cdot \widehat{\phi}_i\right\| \leq \epsilon$ for some choice of phase $\zeta_i \in \mathbb{C}$. As $|\zeta_i\zeta_j \cdot \widehat{\phi}_i^t M \widetilde{\phi}_j| = |\widehat{\phi}_i^t M \widetilde{\phi}_j|$, we may assume without loss of generality that $\zeta_i, \zeta_j = 1$. Now note that

$$|\widehat{\phi}_i^t M \widehat{\phi}_j| \leq |\phi_i^t M \widehat{\phi}_j| + \epsilon \, \|M\|_\infty \leq |\phi_i^t M \phi_j| + 2\epsilon \, \|M\|_\infty \tag{9.350}$$

by triangle inequality, from which the claim follows. □

Henceforth, let $\widehat{\phi}_i$ denote the pure state obtained by applying state tomography to copies of ϕ_i with $\epsilon = 1/5$. We collect the following basic facts about $\widehat{\phi}_i$, which will allow us to distinguish among the three types of channels.

Lemma 68 (See e.g. (G. W. Anderson, Guionnet, and Zeitouni, 2009), Corollary 4.4.28)**.** *For $G = O(d), U(d)$, let $f : G \longrightarrow \mathbb{R}$ be L-Lipschitz with respect to the Frobenius norm. There is an absolute constant $c > 0$ such that for x sampled from the Haar measure on G,* $\Pr |f(x) - \mathbb{E} f(x)| > c \cdot L\sqrt{\log(1/\delta)/d} \leq \delta$.

Lemma 69. *Condition on the outcome of Lemma 67. If C is a Haar-random symplectic matrix channel, then $|\widehat{\phi}_2^t J \widehat{\phi}_3| > 1/2$. If C is a Haar-random unitary or orthogonal channel, then $|\widehat{\phi}_2^t J \widehat{\phi}_3| < 1/2$ with probability at least $14/15$ over the randomness of the channel.*

Proof. By definition of the symplectic matrix channel, $\phi_2^t J \phi_3 = \psi_2^t S^t J S \psi_3 = \psi_2^t J \psi_3 = -1$. The first part of the lemma follows by Corollary 23 and the fact that $\|J\|_\infty = 1$. For the second part, note that the joint distribution on (ϕ_2, ϕ_3) is invariant under the transformation $(\phi_2, -\phi_3)$ when C is a Haar-random unitary (resp. orthogonal) transformation, because conditioned on ϕ_2, ϕ_3 is a Haar-random in the subspace in $\mathbb{C}^{2^n}$ (resp. $\mathbb{R}^{2^n}$) orthogonal to ϕ_2. So in either case,

$$\mathbb{E} *\phi_2^t J \phi_3 = \frac{1}{2} \mathbb{E} *\phi_2^t J \phi_3 + \mathbb{E} *\phi_2^t J(-\phi_3) = 0. \tag{9.351}$$

Note that the function $F : O \mapsto \psi_2^t O^t J O \psi_3$ is 2-Lipschitz:

$$|F(O_1) - F(O_2)| \leq \left|\psi_2^t (O_1 - O_2)^t J O_1 \psi_3\right| + \left|\psi_2^t O_2^t J (O_1 - O_2) \psi_3\right| \leq 2 \left\|O_1 - O_2\right\|_F . \tag{9.352}$$

So by Lemma 68, with probability at least $9/10$ over the randomness of the channel, we have that $|\phi_2^t J \phi_3| \leq O(1/2^{n/2})$. The second part of the lemma then follows from Corollary 23. □

Lemma 70. *Condition on the outcome of Lemma 67. If C is a Haar-random orthogonal matrix channel, then $|\widehat{\phi}_1^t \widehat{\phi}_1| = 1$. If C is a Haar-random unitary matrix channel, then $|\widehat{\phi}_1^t \widehat{\phi}_1| < 1/2$ with probability at least $14/15$ over the randomness of the channel.*

Proof. Because ϕ_1 is a unit vector with only real entries, $\widehat{\phi}_1^t \widehat{\phi}_1 = 1$, so the first part of the claim follows by Corollary 23. For the second part, if the channel is Haar-random unitary, then ϕ_1 is a Haar-random complex unit vector, so $\mathbb{E}\, \widehat{\phi}_1^t \widehat{\phi}_1 = 0$. The function $F : U \mapsto \psi_1^t U^t U \psi_1$ is 2-Lipschitz by a calculation completely analogous to (9.352). So by Lemma 68, with probability at least $9/10$ over the randomness of the channel, we have that $|\widehat{\phi}_1^t \widehat{\phi}_1| \leq O(1/2^{n/2})$. The second part of the lemma then follows from Corollary 23. □

We are now ready to prove Theorem 62.

Proof of Theorem 62. The algorithm will be to apply the state tomography algorithm of Lemma 67 to the outputs of $\{|\psi_i\rangle\}$ under the channel, yielding pure states $\{|\widehat{\phi}_i\rangle\}$. By a union bound, with probability $2/3$ we have that the state tomography algorithm succeeds for all $i = 1, 2, 3$, and Lemmas 69 and 70 hold. We form the quantity $|\widehat{\phi}_2^t J \widehat{\phi}_3|$ and check whether it exceeds $1/2$. If so, we conclude by Lemma 69 that C is symplectic. Otherwise, we form the quantity $|\widehat{\phi}_1^t \widehat{\phi}_1|$ and check whether it exceeds $1/2$. If so, we conclude by Lemma 70 that C is orthogonal, otherwise it is unitary. □

9.10 Predicting observables with bounded quantum memory

Here we will substantively generalize our results for predicting highly-incompatible observables, given in Section 9.5. We show an exponential lower bound on the number of experiments when the size of the additional quantum memory is not large enough.

Background and statement of results

Let us first recapitulate the setting of our previous results so as to draw a contrast with our generalization. We have so far considered an experimentalist who is given sequential access to copies of an unknown state ρ. In each measurement round, the experimentalist receives a copy of ρ and can measure with a POVM. The residual post-measurement state is then discarded, and only the classical data of the POVM outcome is kept. This classical data can be used to inform the choice of POVM measurement employed in subsequent rounds, i.e. the protocol can be adaptive. This kind of protocol is emblematic of most contemporary and historical experiments in physics.

Note that the information maintained and processed from round to round in the protocols described above is solely classical. With the advent of quantum computers and more flexible quantum memory architectures, a new possibility emerges. Suppose that the unknown state ρ in question is an n-qubit state. Moreover, suppose we have $n + k$ qubit registers under our control. Then we can use those registers however we please, including performing arbitrary quantum information processing. Our only constraint is that each time we receive a new copy of ρ, we must necessarily use n qubits of our registers to hold it. It is thus appropriate to say that we have k qubits of quantum memory, since even when we receive a new state ρ

we can still maintain k qubits worth of quantum data. We further allow ourselves to maintain and process an arbitrary amount of classical data, thought of as being stored in a classical device external to our quantum system.

A new question immediately presents itself: are there experimental tasks which are exponentially hard with only k qubits of quantum memory, but easy with $k' > k$ qubits of quantum memory? In (Sitan Chen, J. Cotler, et al., 2021b) this question was answered in the affirmative, but only in the sense of query complexity. That is, it was shown that there is an experimental task that requires $\Omega(2^{(n-k)/3})$ copies of ρ if there are only k qubits of quantum memory; however, the gate complexity of achieving the task is *always* exponential regardless of the size of k. Let us unpack this result. Note that if $k = n$, the bound $\Omega(2^{(n-k)/3})$ becomes trivial; indeed, it can be shown that one only requires a modest number of copies of ρ to achieve the specified task if $n = k$. But even in this case, the total number of quantum operations required is exponentially large in n. Nonetheless, the result is an interesting one: it means that unless k goes as $n - c\ \log(n)$ (i.e. unless k is logarithmically close to n), the task is exponentially hard. When $k \sim n - c\ \log(n)$, the task is only polynomially hard in the sense of query complexity.

While the aforementioned result is theoretically interesting, it does not correspond to a quantum memory advantage that could be realized by a quantum device on account of the exponential gate complexity required to achieve the task for a quantum memory of *any* size. Here we ameliorate this issue and present the first example of a quantum memory advantage in the sense of both query *and* gate complexity, and as such, it can be realized on a quantum device. Moreover, we have realized this quantum advantage in our experiments on the Sycamore quantum computer.

Our experimental task has the form of a partially-revealed many-versus-one distinguishing task, closely related to the one in Section 9.5. A statement of the new task is as follows:

Task 4 (Expectation value with bounded quantum memory)**.** *There is an unknown state ρ which is either*

1. *A maximally mixed state $I/2^n$ on n qubits, or*
2. *The state $(I + P)/2^n$ where $P \in \{I, X, Y, Z\}^{\otimes n} \setminus \{I^{\otimes n}\}$ is a random but fixed Pauli string.*

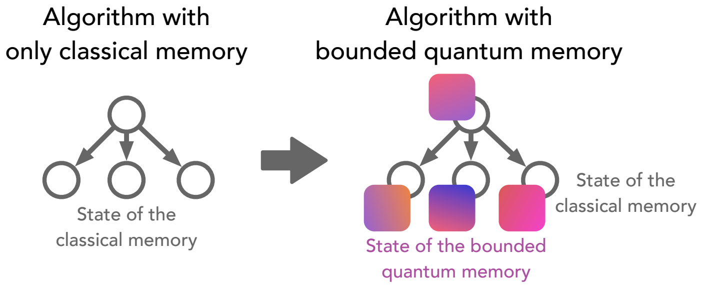


Figure 9.14: *Illustration of learning tree representation for algorithms with bounded quantum memory.* We consider algorithms with unlimited classical memory and a bounded quantum memory consisting of k qubits. To each node in the tree (corresponding to the state of the classical memory), we associate the k-qubit state of the bounded quantum memory.

The choice of whether case 1 or case 2 is instantiated is made with equal probability at the outset, and is not revealed. The experimentalist is given access to T copies of the unknown state ρ for a T decided by the experimentalist, and after this an observable O is revealed. The task of the experimentalist is to determine the value of $|\operatorname{tr}(O\rho)|$ using the final state of the $n+k$ qubit registers, along with any classical information that has been stored or processed along the way. In case 1 the operator O is chosen uniformly at random from the non-identity Pauli strings; in case 2 the O is chosen to be the Pauli operator P if the state ρ is $(I+P)/2^n$.

Note that if $k = n$ so that the total number of registers is $2n$, then the task can be readily solved using the algorithm given in Section 9.5. This algorithm is both query and gate efficient: we only require a constant number of copies of ρ (i.e. the number of copies is independent of n) and $O(n)$ gate complexity.

What is difficult is to show that if $k < n$, then the number of copies of ρ we require to determine $|\operatorname{tr}(O\rho)|$ as per the task above is $\Theta(2^{(n-k)/3})$. We will establish this in the subsections which follow below.

Review of learning tree framework for bounded quantum memories

Here we provide an exposition of the learning tree framework for bounded quantum memories in (Sitan Chen, J. Cotler, et al., 2021b). As explained above, suppose we have $n+k$ qubit registers, where we designate k as the quantum memory. Suppose for the moment that $k < n$. At each round in the protocol, we receive an n-qubit state ρ, which we must hold on the n non-memory registers. (Note that we cannot receive more than one copy of ρ, since we do not have enough registers to hold additional copies on account of $k < n$.) Then upon receiving and holding ρ, the state of all of our registers can be written as $\rho \otimes \mathbf{\Sigma}$, where $\mathbf{\Sigma}$ is the density matrix of the k-qubit quantum memory. The most general operation we can perform on the joint system is a quantum process, followed by a POVM measurement; we can then apply another quantum process followed by another POVM measurement, and so on. However, we can rewrite an alternating sequence of quantum processes and POVM measurements as a single POVM measurement, which we denote by $\{F_i\}_{i=1}^{N-1}$. Writing $F_i = M_i^\dagger M_i$, suppose our measurement outputs the ith POVM element. Defining

$$A_{M_i}^{\rho}(\mathbf{\Sigma}) := \mathrm{tr}_{1,\ldots,n}\Big(M_i(\rho \otimes \mathbf{\Sigma})M_i^\dagger\Big)\ , \tag{9.353}$$

the reduced density matrix of our quantum memory is

$$\frac{A_{M_i}^{\rho}(\mathbf{\Sigma})}{\mathrm{tr}(A_{M_i}^{\rho}(\mathbf{\Sigma}))} \tag{9.354}$$

with probability $\mathrm{tr}(A_{M_i}^{\rho}(\mathbf{\Sigma}))$. In the next round, we can leverage our measurement output i to adaptively inform our choice of POVM on

$$\rho \otimes \frac{A_{M_i}^{\rho}(\mathbf{\Sigma})}{\mathrm{tr}(A_{M_i}^{\rho}(\mathbf{\Sigma}))}\ . \tag{9.355}$$

Indeed, we can use the information of *all* of our previous POVM outcomes to inform the choice of our next POVM. An illustration is given in Supp. Fig. 9.14.

The above description is ripe for being cast in the learning tree framework, which we presently articulate. The definition below is the same as Definition 6.1 of (Sitan Chen, J. Cotler, et al., 2021b), albeit with slightly different notation.

Definition 6.1 of (Sitan Chen, J. Cotler, et al., 2021b) (Tree representation of learning states with bounded quantum memory)**.** *Let ρ be a fixed, unknown quantum density matrix on n qubits. Suppose we have access to $n + k$ qubit registers. A learning algorithm with a quantum memory of size k can be expressed as a rooted tree $\mathcal{T}$ of depth T, where each node encodes the current state of the quantum memory in addition to the transcript of measurement outcomes the algorithm has seen so far. In particular, the tree satisfies the following properties:*

1. *Each note u is associated with a k-qubit unnormalized density matrix $\mathbf{\Sigma}^{\rho}(u)$ corresponding to the current state of the quantum memory.*

2. *For the root r of the tree, $\mathbf{\Sigma}^{\rho}(r)$ is an initial state denoted by $\mathbf{\Sigma}_0$.*

3. *At each note u, we apply a POVM measurement $\{F_s^u\}_s$ on $\rho \otimes \mathbf{\Sigma}^\rho(u)$ to obtain a classical outcome s. Each child node v of u is connected through the edge $e_{u,s}$.*

4. *If v is the child note of u connected through the edge $e_{u,s}$, then letting $F_s^u = M_s^{u\,\dagger} M_s^u$ we have*
$$\mathbf{\Sigma}^\rho(v) := A_{M_s^u}(\mathbf{\Sigma}^\rho(u)) \,. \tag{9.356}$$

5. *Note that for any node u we have that $p^\rho(u) := tr(\mathbf{\Sigma}^\rho(u))$ is the probability that the transcript of measurement outcomes observed by the learning algorithm is described by u. Moreover, $\mathbf{\Sigma}^\rho(u)/p^\rho(u)$ is the (normalized) state of the k-qubit memory at the node u.*

Let us unpack the ingredients of this definition. The initial state of our quantum memory is $\mathbf{\Sigma}_0$, and we apply some initial choice of POVM $\{F_s^r\}_s$ (where r denotes the 'root' of the tree). If we measure the sth POVM outcome, then the quantum memory is in the unnormalized state $A_{M_s^r}(\mathbf{\Sigma}_0)$ with probability $\mathrm{tr}(A_{M_s^r}(\mathbf{\Sigma}_0))$. Each outcome s of the POVM corresponds to a child note of the root; thus at the next level of the tree, each node is labeled by the POVM outcome s and the corresponding state of the quantum memory $A_{M_s^r}(\mathbf{\Sigma}_0) := \mathbf{\Sigma}^\rho(s)$. For the next measurement, we can leverage our knowledge of the previous POVM to craft a new POVM to be applied to the present state of the quantum memory. This type of procedure is repeated for many rounds.

To be explicit, suppose that the present state of the quantum memory is $\Sigma^\rho(u)$, where the node ρ reflects a sequence or *transcript* of POVM outcomes which brought us to the present state by an adaptive protocol. We can use this transcript of previous outcomes to choose a new POVM $\{F_s^u\}_s$ that we use to measure $\rho \otimes \mathbf{\Sigma}^u(\rho)$, which will result in the output $A_{M_s^u}(\mathbf{\Sigma}^\rho(u))$ with probability $\mathrm{tr}(A_{M_s^u}(\mathbf{\Sigma}^\rho(u)))$. The nodes v in the next layer encode the data of the previous measurement outcomes and the latest outcome (i.e., determined by the location of v in the tree), as well as the new (conveniently unnormalized) state of the quantum memory $A_{M_s^u}(\mathbf{\Sigma}^\rho(u)) := \mathbf{\Sigma}^\rho(v)$, where here v is connected to u by an edge $e_{u,s}$ (designating that v is the consequence of the sth measurement outcome starting from the configuration in u).

Hardness result for small quantum memories

We will prove the following result using the learning tree framework:

Theorem 63 (Shadow tomography with partial reveal using a bounded quantum memory)**.** *Consider Task 4 for learning an expectation value with a bounded quantum memory. Any learning algorithm with $n + k$ qubit registers needs $T \geq \Omega(2^{(n-k)/3})$ copies of ρ to determine $|\,tr(O\rho)|$ with probability at least* $2/3$.

On account of (3.27), it suffices to upper bound $\mathbb{E}_P\left[\mathrm{TV}(p_{I/2^n}, p_{(I+P)/2^n})\right]$. To do so, we will leverage a key technical result coming from Theorem 1.4 of (Sitan Chen, J. Cotler, et al., 2021b). But in order to state this technical result, we first need to introduce the notion of *good Paulis* and *bad Paulis*. While details are provided in Definition 6.4 of (Sitan Chen, J. Cotler, et al., 2021b), here we describe the essential intuition and key properties.

In the learning protocol, we are trying to distinguish between the maximally mixed state $I/2^n$ and states of the form $(I + P)/2^n$. The intuition is that if the size of our quantum memory k is small relative to n, then it is hard to tell the two kinds of states apart. If this was the case, then any round of the protocol should only reveal a very small amount of distinguishing information. In particular, suppose we are at node u in the learning tree, and so the state of the quantum memory at that node is either $\mathbf{\Sigma}_{I/2^n}(u)$ or $\mathbf{\Sigma}_{(I+P)/2^n}(u)$ for some Pauli string P. If we consider a POVM $\{F_s^u\}_s$ where $F_s^u = M_s^{u\,\dagger} M_s^u$, then if we measure some fixed outcome s the new state of the quantum memory will be either $A_{M_s^u}^{I/2^n}(\mathbf{\Sigma}_{I/2^n}(u))$ or $A_{M_s^u}^{(I+P)/2^n}(\mathbf{\Sigma}_{(I+P)/2^n}(u))$. We would like for $\left\|A_{M_s^u}^{I/2^n}(\mathbf{\Sigma}_{I/2^n}(u)) - A_{M_s^u}^{(I+P)/2^n}(\mathbf{\Sigma}_{(I+P)/2^n}(u))\right\|_1$ to be exponentially small in $n-k$, in particular relative to some distinguishing measure between $\mathbf{\Sigma}_{I/2^n}(u)$ and $\mathbf{\Sigma}_{(I+P)/2^n}(u)$. This would mean that starting from node u, the next POVM measurement will not significantly change our ability to distinguish the two possibilities for the resulting memory registers. While we cannot guarantee that such a property holds for all states $(I + P)/2^n$, such a property will hold for some P's. Given a node u, the set of good Paulis $P[u]$ is the set of all Pauli operators satisfying a particular version of the above property for *all* edges from the root of the tree to u (see Definition 6.4 of (Sitan Chen, J. Cotler, et al., 2021b) for details). The residual Paulis are called the set of bad Paulis. In other words, the good Paulis $P[u]$ designate the states $(I + P)/2^n$ which are hard to distinguish from $I/2^n$ for a particular instantiation of the learning tree, specifically for the sequence of POVMs that get us from the root of said learning tree to the node u. By contrast, the bad Paulis reveal too much information.

We have the following useful Lemma about bad Paulis, which we will soon leverage in the proof of Theorem 63:

Lemma 71 (Fact 6.5 of (Sitan Chen, J. Cotler, et al., 2021b))**.** *For any edge $e_{u,s}$, there are at most $2^{-(n-k)/3} \cdot (4^n - 1)$ bad Paulis P. In particular, along any root-to-leaf path of the learning tree, there are at most $T \cdot 2^{-(n-k)/3} \cdot (4^n - 1)$ Paulis which are bad for some edge along the path.*

Equipped with our discussion of good and bad Paulis, we can now state the following technical result from (Sitan Chen, J. Cotler, et al., 2021b):

Lemma 72 (Following from the proof of Theorem 1.4 of (Sitan Chen, J. Cotler, et al., 2021b))**.** *We have the inequality*

$$\frac{1}{4^n - 1} \sum_{\ell \in \, leaf(\mathcal{T})} \sum_{P \in P[\ell]} \left\| \boldsymbol{\Sigma}_{I/2^n}(\ell) - \boldsymbol{\Sigma}_{(I+P)/2^n}(\ell) \right\|_1 \leq T \cdot 2^{-(n-k)/3} \cdot \sqrt{\frac{2^{2n}}{2^{2n} - 1}} \,. \tag{9.357}$$

We are finally ready to prove Theorem 63.

Proof of Theorem 63. Let us upper bound $\mathbb{E}_P\left[\mathrm{TV}(p_{I/2^n}, p_{(I+P)/2^n})\right]$. We have the inequalities

$$\mathbb{E}_P\left[\mathrm{TV}(p_{I/2^n}, p_{(I+P)/2^n})\right] \tag{9.358}$$

$$\leq \mathbb{E}_P\left[\sum_{\ell} \max\big(0, p_{I/2^n}(\ell) - p_{(I+P)/2^n}(\ell)\big)\right] \tag{9.359}$$

$$\leq \mathbb{E}_P\left[\sum_{\ell} \min\big(p_{I/2^n}(\ell), |p_{I/2^n}(\ell) - p_{(I+P)/2^n}(\ell)|\big)\right] \tag{9.360}$$

$$\leq \mathbb{E}_P\left[\sum_{\ell} \min\Big(p_{I/2^n}(\ell), \left\| \boldsymbol{\Sigma}_{I/2^n}(\ell) - \boldsymbol{\Sigma}_{(I+P)/2^n}(\ell) \right\|_1\Big)\right] \tag{9.361}$$

$$\leq \sum_{\ell} \Pr[P \notin P[\ell]] \, p_{I/2^n}(\ell) + \frac{1}{4^n - 1} \sum_{P \in P[\ell]} \left\| \boldsymbol{\Sigma}_{I/2^n}(\ell) - \boldsymbol{\Sigma}_{(I+P)/2^n}(\ell) \right\|_1 \,. \tag{9.362}$$

In the first line, we have used that $\mathrm{TV}(p, q) = \frac{1}{2}\sum_i |p_i - q_i| = \sum_{i\,:\,p_i \geq q_i}(p_i - q_i) = \sum_i \max(0, p_i - q_i)$. To go from (9.359) to (9.360) we used $\max(0, a - b) \leq \min(a, |a - b|)$. In going from (9.360) to (9.361) we leveraged that $|p_{I/2^n}(\ell) - p_{(I+P)/2^n}(\ell)| = |\mathrm{tr}(\boldsymbol{\Sigma}_{I/2^n}(\ell) - \boldsymbol{\Sigma}_{(I+P)/2^n}(\ell))| \leq \|\boldsymbol{\Sigma}_{I/2^n}(\ell) - \boldsymbol{\Sigma}_{(I+P)/2^n}(\ell)\|_1$. Finally, to go from (9.361) to (9.362) we used $\sum_{i \in S} \min(a_i, b_i) \leq \sum_{i \in S \setminus R} a_i + \sum_{i \in R} b_i$ for any $R \subseteq S$.

By Lemma 71 and the fact that $\sum_\ell p_{I/2^n}(\ell) = 1$, we have the simple bound

$$\sum_\ell \Pr[P \notin P[\ell]]\, p_{I/2^n}(\ell) \leq T \cdot 2^{-(n-k)/3} \tag{9.363}$$

and Lemma 72 gives us

$$\frac{1}{4^n - 1} \sum_{P \in P[\ell]} \left\| \mathbf{\Sigma}_{I/2^n}(\ell) - \mathbf{\Sigma}_{(I+P)/2^n}(\ell) \right\|_1 \leq T \cdot 2^{-(n-k)/3} \cdot \sqrt{\frac{2^{2n}}{2^{2n} - 1}}\,. \tag{9.364}$$

Then in total, we have

$$\mathbb{E}_P \left[\mathrm{TV}(p_{I/2^n}, p_{(I+P)/2^n}) \right] \leq T \cdot 2^{-(n-k)/3} \left(1 + \sqrt{\frac{2^{2n}}{2^{2n} - 1}} \right). \tag{9.365}$$

If the left-hand side is $\Omega(1)$, then we must thus have $T \geq \Omega(2^{(n-k)/3})$, as claimed.

□

BIBLIOGRAPHY

Aaronson, Scott (2018). “Shadow Tomography of Quantum States”. In: *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2018. Los Angeles, CA, USA: ACM, pp. 325–338. ISBN: 978-1-4503-5559-9. DOI: 10.1145/3188745.3188802. URL: http://doi.acm.org/10.1145/3188745.3188802.

– (2019). “Shadow tomography of quantum states”. In: *SIAM Journal on Computing* 49.5, STOC18–368. DOI: 10.1137/18M120275X. URL: https://epubs.siam.org/doi/abs/10.1137/18M120275X.

– (2020). “Shadow tomography of quantum states”. In: *SIAM Journal on Computing* 0, STOC18–368. URL: https://epubs.siam.org/doi/abs/10.1137/18M120275X.

Aaronson, Scott and Andris Ambainis (2009). “The need for structure in quantum speedups”. In: *arXiv preprint arXiv:0911.0996*. URL: https://arxiv.org/abs/0911.0996.

Aaronson, Scott and Daniel Gottesman (2004). “Improved simulation of stabilizer circuits”. In: *Phys. Rev. A* 70.5, p. 052328.

Aaronson, Scott and Guy N. Rothblum (2019). “Gentle Measurement of Quantum States and Differential Privacy”. In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2019. Phoenix, AZ, USA: Association for Computing Machinery, pp. 322–333. ISBN: 9781450367059. DOI: 10.1145/3313276.3316378. URL: https://doi.org/10.1145/3313276.3316378.

Abrahamsen, Nilin (2020). “Sub-exponential algorithm for 2D frustration-free spin systems with gapped subsystems”. In: *arXiv preprint arXiv:2004.02850*.

Acín, A. et al. (July 2001). “Classification of Mixed Three-Qubit States”. In: *Phys. Rev. Lett.* 87 (4), p. 040401. DOI: 10.1103/PhysRevLett.87.040401. URL: https://link.aps.org/doi/10.1103/PhysRevLett.87.040401.

Affleck, Ian, Tom Kennedy, et al. (1988). “Valence bond ground states in isotropic quantum antiferromagnets”. In: *Commun. Math. Phys.* 115.3, pp. 477–528. ISSN: 0010-3616. DOI: 10.1007/BF01218021. URL: http://link.springer.com/10.1007/BF01218021.

Affleck, Ian and Elliott H. Lieb (1986). “A proof of part of Haldane’s conjecture on spin chains”. In: *Lett. Math. Phys.* 12.1, pp. 57–69. ISSN: 0377-9017. DOI: 10.1007/BF00400304. URL: http://link.springer.com/10.1007/BF00400304.

Aharonov, Dorit, Jordan S Cotler, and Xiao-Liang Qi (2021). “Quantum Algorithmic Measurement”. In: *arXiv preprint arXiv:2101.04634*.

Aharonov, Dorit, Wim Van Dam, et al. (2008). “Adiabatic quantum computation is equivalent to standard quantum computation”. In: *SIAM review* 50.4, pp. 755–787.

Alon, Noga and Joel H. Spencer (2008). *The Probabilistic Method, Third Edition*. Wiley-Interscience series in discrete mathematics and optimization. Wiley. ISBN: 978-0-470-17020-5.

Altman, Naomi S (1992). “An introduction to kernel and nearest-neighbor nonparametric regression”. In: *The American Statistician* 46.3, pp. 175–185.

Anderson, E. et al. (1999). *LAPACK Users’ Guide*. Third. Philadelphia, PA: Society for Industrial and Applied Mathematics. ISBN: 0-89871-447-8 (paperback).

Anderson, Greg W., Alice Guionnet, and Ofer Zeitouni (2009). *An Introduction to Random Matrices*. Cambridge Studies in Advanced Mathematics. Cambridge University Press. DOI: 10.1017/CBO9780511801334.

Anshu, Anurag, David Gosset, et al. (2021). “Improved approximation algorithms for bounded-degree local hamiltonians”. In: *Physical Review Letters* 127.25, p. 250502.

Anshu, Anurag, Zeph Landau, and Yunchao Liu (2021). “Distributed quantum inner product estimation”. In: *arXiv:2111.03273*.

Araki, Huzihiro and Elliott H Lieb (2002). “Entropy inequalities”. In: *Inequalities*. Springer, pp. 47–57.

Arora, Sanjeev et al. (2019). “On exact computation with an infinitely wide neural net”. In: *NeurIPS*, pp. 8139–8148.

Arovas, D. P. (n.d.). *Lecture Notes on Group Theory in Physics*. Available at https://courses.physics.ucsd.edu/2016/Spring/physics220/LECTURES/GROUP_THEORY.pdf. online notes. URL: https://courses.physics.ucsd.edu/2016/Spring/physics220/LECTURES/GROUP_THEORY.pdf.

Arrazola, Juan Miguel et al. (2019). “Quantum-inspired algorithms in practice”. In: *arXiv preprint arXiv:1905.10415*.

Arrieta, Alejandro Barredo et al. (2020). “Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI”. In: *Information Fusion* 58, pp. 82–115.

Arunachalam, Srinivasan, Alex B Grilo, and Aarthi Sundaram (2019). “Quantum hardness of learning shallow classical circuits”. In: *arXiv:1903.02840*.

Arunachalam, Srinivasan, Alex B Grilo, and Henry Yuen (2020). “Quantum statistical query learning”. In: *arXiv preprint arXiv:2002.08240*.

Arunachalam, Srinivasan and Ronald de Wolf (2016). “Optimal quantum sample complexity of learning algorithms”. In: *arXiv preprint arXiv:1607.00932*.

Arunachalam, Srinivasan and Ronald de Wolf (2017). “Guest column: A survey of quantum learning theory”. In: *ACM SIGACT News* 48.2, pp. 41–67.

Arute, Frank et al. (2019). “Quantum supremacy using a programmable superconducting processor”. In: *Nature* 574.7779, pp. 505–510. URL: `https://doi.org/10.1038/s41586-019-1666-5`.

Bachmann, Sven, Alex Bols, et al. (2020). “A Many-Body Index for Quantum Charge Transport”. In: *Commun. Math. Phys.* 375.2, pp. 1249–1272. ISSN: 0010-3616. DOI: `10.1007/s00220-019-03537-x`. URL: `http://link.springer.com/10.1007/s00220-019-03537-x`.

Bachmann, Sven, Spyridon Michalakis, et al. (2012). “Automorphic equivalence within gapped phases of quantum lattice systems”. In: *Commun. Math. Phys.* 309.3, pp. 835–871.

Bachmann, Sven and Bruno Nachtergaele (2014). “On Gapped Phases with a Continuous Symmetry and Boundary Operators”. In: *J. Stat. Phys.* 154.1-2, pp. 91–112. ISSN: 0022-4715. DOI: `10.1007/s10955-013-0850-5`. URL: `http://link.springer.com/10.1007/s10955-013-0850-5`.

Bădescu, Costin and Ryan O’Donnell (2020). “Improved quantum data analysis”. In: *arXiv:2011.10908*.

Barak, Boaz et al. (2015). “Beating the Random Assignment on Constraint Satisfaction Problems of Bounded Degree”. In: *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2015)*. Ed. by Naveen Garg et al. Vol. 40. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, pp. 110–123. ISBN: 978-3-939897-89-7. DOI: `10.4230/LIPIcs.APPROX-RANDOM.2015.110`. URL: `http://drops.dagstuhl.de/opus/volltexte/2015/5298`.

Bartkiewicz, Karol et al. (2020). “Experimental kernel-based quantum machine learning in finite feature space”. In: *Scientific Reports* 10.1, pp. 1–9. URL: `https://www.nature.com/articles/s41598-020-68911-5`.

Bartlett, Peter L and Shahar Mendelson (2002). “Rademacher and Gaussian complexities: Risk bounds and structural results”. In: *J Mach Learn Res* 3.Nov, pp. 463–482.

Beals, Robert et al. (2001). “Quantum lower bounds by polynomials”. In: *J. ACM* 48.4, pp. 778–797.

Becca, Federico and Sandro Sorella (2017). *Quantum Monte Carlo Approaches for Correlated Systems*. Cambridge University Press. DOI: `10.1017/9781316417041`.

Becke, Axel D (1993). “A new mixing of Hartree–Fock and local density-functional theories”. In: *J. Chem. Phys.* 98.2, pp. 1372–1377.

Bengtsson, Ingemar and Karol Życzkowski (2017). *Geometry of quantum states: an introduction to quantum entanglement*. Cambridge university press.

Bennett, Charles H et al. (1999). “Quantum nonlocality without entanglement”. In: *Physical Review A* 59.2, p. 1070. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.59.1070.

Bernien, Hannes et al. (2017). “Probing many-body dynamics on a 51-atom quantum simulator”. In: *Nature* 551.7682, pp. 579–584.

Biamonte, Jacob et al. (2017). “Quantum machine learning”. In: *Nature* 549.7671, pp. 195–202.

Bisio, Alessandro et al. (2010). “Optimal quantum learning of a unitary transformation”. In: *Physical Review A* 81.3, p. 032324. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.81.032324.

Bland-Hawthorn, Joss, Matthew J. Sellars, and John G. Bartholomew (July 2021). “Quantum memories and the double-slit experiment: implications for astronomical interferometry”. In: *Journal of the Optical Society of America B* 38.7, A86–A98. DOI: 10.1364/JOSAB.424651. URL: http://www.osapublishing.org/josab/abstract.cfm?URI=josab-38-7-A86.

Blank, Carsten et al. (2020). “Quantum classifier with tailored quantum kernel”. In: *npj Quantum Information* 6.1, pp. 1–7. URL: https://www.nature.com/articles/s41534-020-0272-6.

Blum, Avrim, Adam Kalai, and Hal Wasserman (July 2003). “Noise-Tolerant Learning, the Parity Problem, and the Statistical Query Model”. In: *J. ACM* 50.4, pp. 506–519. ISSN: 0004-5411. DOI: 10.1145/792538.792543. URL: https://doi.org/10.1145/792538.792543.

Blume-Kohout, Robin et al. (2017). “Demonstration of qubit operations below a rigorous fault tolerance threshold with gate set tomography”. In: *Nature communications* 8.1, pp. 1–13.

Blumer, Anselm et al. (1989). “Learnability and the Vapnik-Chervonenkis dimension”. In: *J. ACM* 36.4, pp. 929–965.

Bohnenblust, H. F. and Einar Hille (1931). “On the Absolute Convergence of Dirichlet Series”. In: *Annals of Mathematics* 32.3, pp. 600–622. ISSN: 0003486X. URL: http://www.jstor.org/stable/1968255 (visited on 08/26/2022).

Bohrdt, Annabelle et al. (2019). “Classifying snapshots of the doped Hubbard model with machine learning”. In: *Nat. Phys.* 15.9, pp. 921–924. ISSN: 1745-2481. DOI: 10.1038/s41567-019-0565-x. URL: https://doi.org/10.1038/s41567-019-0565-x.

Boixo, Sergio et al. (2018). “Characterizing quantum supremacy in near-term devices”. In: *Nature Physics* 14.6, pp. 595–600. URL: https://www.nature.com/articles/s41567-018-0124-x.

Bonet-Monroig, Xavier, Ryan Babbush, and Thomas E O’Brien (2019). “Nearly optimal measurement scheduling for partial tomography of quantum states”. In: *arXiv preprint arXiv:1908.05628*.

Borwein, Peter and Tamás Erdélyi (1995). *Polynomials and polynomial inequalities*. Vol. 161. Springer Science & Business Media.

Boyd, Stephen, Stephen P Boyd, and Lieven Vandenberghe (2004). *Convex optimization*. Cambridge University Press.

Brakerski, Zvika and Omri Shmueli (2019). "(Pseudo) Random Quantum States with Binary Phase". In: *Theory of Cryptography Conference*. Springer, pp. 229–250.

– (2020). "Scalable Pseudorandom Quantum States". In: *Annual International Cryptology Conference*. Springer, pp. 417–440.

Brandão, Fernando GSL et al. (2019). "Models of quantum complexity growth". In: *arXiv preprint arXiv:1912.04297*.

Bravyi, Sergey, Matthew B Hastings, and Frank Verstraete (2006). "Lieb-Robinson bounds and the generation of correlations and topological quantum order". In: *Phys. Rev. Lett.* 97.5, p. 050401.

Breiman, Leo (2001). "Random forests". In: *Machine learning* 45.1, pp. 5–32. URL: `https://link.springer.com/article/10.1023/A:1010933404324`.

Bridgeman, Jacob C and Christopher T Chubb (2017). "Hand-waving and interpretive dance: an introductory course on tensor networks". In: *Journal of physics A: Mathematical and theoretical* 50.22, p. 223001.

Briegel, H. J. et al. (Jan. 2009). "Measurement-based quantum computation". In: *Nat. Phys.* 5, pp. 19–26. URL: `https://doi.org/10.1038/nphys1157`.

Broughton, Michael, Guillaume Verdon, Trevor McCourt, Antonio J Martinez, et al. (2020). "Tensorflow quantum: A software framework for quantum machine learning". In: *arXiv preprint arXiv:2003.02989*. URL: `https://arxiv.org/abs/2003.02989`.

Broughton, Michael, Guillaume Verdon, Trevor McCourt, Antonio J. Martinez, et al. (2021). *TensorFlow Quantum: A Software Framework for Quantum Machine Learning*. arXiv: `2003.02989 [quant-ph]`.

Browaeys, Antoine and Thierry Lahaye (2020). "Many-body physics with individually controlled Rydberg atoms". In: *Nat. Phys.* 16.2, pp. 132–142. ISSN: 1745-2481. DOI: `10.1038/s41567-019-0733-z`. URL: `https://doi.org/10.1038/s41567-019-0733-z`.

Bru, J-B and Walter de Siqueira Pedra (2016). *Lieb-Robinson bounds for multi-commutators and applications to response theory*. Vol. 13. Springer.

Brydges, Tiff et al. (2019). "Probing Rényi entanglement entropy via randomized measurements". In: *Science* 364.6437, pp. 260–263.

Bu, Kaifeng et al. (2022). "Classical shadows with Pauli-invariant unitary ensembles". In: *arXiv preprint arXiv:2202.03272*.

Bubeck, Sebastien, Sitan Chen, and Jerry Li (2020). “Entanglement is Necessary for Optimal Quantum Property Testing”. In: *arXiv preprint arXiv:2004.07869*.

Buitinck, Lars et al. (2013). “API design for machine learning software: experiences from the scikit-learn project”. In: *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pp. 108–122. URL: https://arxiv.org/abs/1309.0238.

Buluta, Iulia and Franco Nori (2009). “Quantum simulators”. In: *Science* 326.5949, pp. 108–111.

Cade, Chris et al. (2019). “Strategies for solving the Fermi-Hubbard model on near-term quantum computers”. In: *arXiv preprint arXiv:1912.06007*. URL: https://arxiv.org/abs/1912.06007.

Canonne, Clément L et al. (2018). “Testing shape restrictions of discrete distributions”. In: *Theory of Computing Systems* 62.1, pp. 4–62.

Car, Richard and Mark Parrinello (1985). “Unified approach for molecular dynamics and density-functional theory”. In: *Phys. Rev. Lett.* 55.22, p. 2471.

Carleo, Giuseppe, Ignacio Cirac, et al. (2019). “Machine learning and the physical sciences”. In: *Rev. Mod. Phys.* 91 (4), p. 045002. DOI: 10.1103/RevModPhys.91.045002. URL: https://link.aps.org/doi/10.1103/RevModPhys.91.045002.

Carleo, Giuseppe and Matthias Troyer (2017a). “Solving the quantum many-body problem with artificial neural networks”. In: *Science* 355.6325, pp. 602–606.

– (2017b). “Solving the quantum many-body problem with artificial neural networks”. In: *Science* 355.6325, pp. 602–606. ISSN: 0036-8075. DOI: 10.1126/science.aag2302.

Caro, Matthias C, Hsin-Yuan Huang, M Cerezo, et al. (2021). “Generalization in quantum machine learning from few training data”. In: *arXiv:2111.05292*.

Caro, Matthias C, Hsin-Yuan Huang, Marco Cerezo, et al. (2022). “Generalization in quantum machine learning from few training data”. In: *Nature communications* 13.1, pp. 1–11.

Caro, Matthias C, Hsin-Yuan Huang, Nicholas Ezzell, et al. (2023). “Out-of-distribution generalization for learning quantum dynamics”. In: *Nature Communications* 14.1, p. 3751.

Carrasquilla, Juan (2020). “Machine learning for quantum matter”. In: *Adv. Phys.: X* 5.1, p. 1797528. DOI: 10.1080/23746149.2020.1797528. eprint: https://doi.org/10.1080/23746149.2020.1797528. URL: https://doi.org/10.1080/23746149.2020.1797528.

Carrasquilla, Juan and Roger G Melko (2017a). “Machine learning phases of matter”. In: *Nature Physics* 13.5, pp. 431–434.

Carrasquilla, Juan and Roger G. Melko (2017b). "Machine learning phases of matter". In: *Nat. Phys.* 13, p. 431. URL: `https://doi.org/10.1038/nphys4035`.

Carrasquilla, Juan, Giacomo Torlai, et al. (2019). "Reconstructing quantum states with generative models". In: *Nat. Mach. Intell.* 1.3, p. 155.

Ceperley, David and Berni Alder (1986). "Quantum Monte Carlo". In: *Science* 231.4738, pp. 555–560. ISSN: 0036-8075. DOI: `10.1126/science.231.4738.555`. eprint: `https://science.sciencemag.org/content/231/4738/555.full.pdf`. URL: `https://science.sciencemag.org/content/231/4738/555`.

Cerezo, Marco et al. (2021). "Variational quantum algorithms". In: *Nature Reviews Physics* 3.9, pp. 625–644.

Chan, Siu-On et al. (2014). "Optimal algorithms for testing closeness of discrete distributions". In: *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*. SIAM, pp. 1193–1203.

Chang, Chih-Chung and Chih-Jen Lin (2011a). "LIBSVM: A library for support vector machines". In: *ACM Transactions on Intelligent Systems and Technology* 2 (3). Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`, 27:1–27:27.

– (2011b). "LIBSVM: a library for support vector machines". In: *ACM transactions on intelligent systems and technology (TIST)* 2.3, pp. 1–27.

Chen, Chi-Fang and Andrew Lucas (2019). "Finite speed of quantum scrambling with long range interactions". In: *Physical review letters* 123.25, p. 250605.

Chen, Senrui, Wenjun Yu, et al. (2021). "Robust shadow estimation". In: *PRX Quantum* 2.3, p. 030348.

Chen, Senrui, Sisi Zhou, et al. (2021). "Quantum advantages for Pauli channel estimation". In: *arXiv preprint arXiv:2108.08488*.

Chen, Sitan, Jordan Cotler, et al. (2021a). "A Hierarchy for Replica Quantum Advantage". In: *arXiv:2111.05874*.

– (2021b). "Exponential separations between learning with and without quantum memory". In: *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, pp. 574–585. DOI: `10.1109/FOCS52979.2021.00063`.

Chen, Sitan, Jerry Li, and Ryan O'Donnell (2021). "Toward Instance-Optimal State Certification With Incoherent Measurements". In: *arXiv:2102.13098*.

Chen, Xie, Zheng-Cheng Gu, Zheng-Xin Liu, et al. (2013). "Symmetry protected topological orders and the group cohomology of their symmetry group". In: *Phys. Rev. B* 87.15, p. 155114. ISSN: 1098-0121. DOI: `10.1103/PhysRevB.87.155114`. URL: `https://link.aps.org/doi/10.1103/PhysRevB.87.155114`.

Chen, Xie, Zheng-Cheng Gu, and Xiao-Gang Wen (2011). “Classification of gapped symmetric phases in one-dimensional spin systems”. In: *Phys. Rev. B* 83.3, p. 035107. ISSN: 1098-0121. DOI: 10.1103/PhysRevB.83.035107. URL: https://link.aps.org/doi/10.1103/PhysRevB.83.035107.

Chia, Nai-Hui, Kai-Min Chung, and Ching-Yi Lai (2020). “On the need for large quantum depth”. In: *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 902–915.

Chia, Nai-Hui, Andras Gilyen, et al. (2020). “Sampling-based sublinear low-rank matrix arithmetic framework for dequantizing quantum machine learning”. In: *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 387–400.

Childs, Andrew M et al. (2018). “Toward the first quantum simulation with quantum speedup”. In: *Proc. Natl. Acad. Sci. U.S.A.* 115.38, pp. 9456–9461.

Choi, Joonhee et al. (2021). *Emergent Randomness and Benchmarking from Many-Body Quantum Chaos*. arXiv: 2103.03535 [quant-ph].

Chollet, Francois et al. (2015). “Keras”. In: Available at https://github.com/fchollet/keras. URL: https://github.com/fchollet/keras.

Choo, Kenny, Antonio Mezzacapo, and Giuseppe Carleo (May 2020). “Fermionic neural-network states for ab-initio electronic structure”. In: *Nat. Commun.* 11.1, p. 2368. ISSN: 2041-1723. DOI: 10.1038/s41467-020-15724-9. URL: https://doi.org/10.1038/s41467-020-15724-9.

Chung, Junyoung et al. (2014). “Empirical evaluation of gated recurrent neural networks on sequence modeling”. In: *arXiv:1412.3555*.

Chung, Kai-Min and Han-Hsuan Lin (2018). “Sample Efficient Algorithms for Learning Quantum Channels in PAC Model and the Approximate State Discrimination Problem”. In: *arXiv preprint arXiv:1810.10938*.

Cirstoiu, Cristina et al. (2020). “Variational fast forwarding for quantum simulation beyond the coherence time”. In: *npj Quantum Information* 6.1, pp. 1–10.

Collins, Benoit and Sho Matsumoto (2017). “Weingarten calculus via orthogonality relations: new applications”. In: *arXiv:1701.04493*.

Coopmans, Luuk, Yuta Kikuchi, and Marcello Benedetti (2022). “Predicting Gibbs State Expectation Values with Pure Thermal Shadows”. In: *arXiv:2206.05302*.

Cortes, Corinna and Vladimir Vapnik (1995). “Support-vector networks”. In: *Mach. Learn.* 20.3, pp. 273–297.

Cotler, Jordan, Hsin-Yuan Huang, and Jarrod R McClean (2021). “Revisiting dequantization and quantum advantage in learning tasks”. In: *arXiv 2112.00811*. URL: https://arxiv.org/abs/2112.00811.

Cotler, Jordan and Frank Wilczek (Mar. 2020a). “Quantum Overlapping Tomography”. In: *Phys. Rev. Lett.* 124 (10), p. 100401. DOI: 10.1103/PhysRevLett.124.100401. URL: https://link.aps.org/doi/10.1103/PhysRevLett.124.100401.

– (2020b). “Quantum overlapping tomography”. In: *Physical review letters* 124.10, p. 100401.

Cotler, Jordan S et al. (2021). “Emergent quantum state designs from individual many-body wavefunctions”. In: *arXiv preprint arXiv:2103.03536*.

Coudron, Matthew and Sanketh Menda (2020). “Computations with greater quantum depth are strictly more powerful (relative to an oracle)”. In: *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 889–901.

Cover, Thomas M. and Joy A. Thomas (2006). *Elements of information theory*. Second. Wiley-Interscience [John Wiley & Sons], Hoboken, NJ, pp. xxiv+748.

Cramer, Marcus et al. (2010). “Efficient quantum state tomography”. In: *Nat. Commun.* 1, p. 149.

Dasgupta, Chandan and Shang-keng Ma (1980). “Low-temperature properties of the random Heisenberg antiferromagnetic chain”. In: *Phys. Rev. B* 22.3, p. 1305.

Degen, Christian L, F Reinhard, and Paola Cappellaro (2017). “Quantum sensing”. In: *Rev. Mod. Phys.* 89.3, p. 035002. DOI: 10.1103/RevModPhys.89.035002. URL: https://journals.aps.org/rmp/abstract/10.1103/RevModPhys.89.035002.

Dehghani, Hossein et al. (2021). “Extraction of the many-body Chern number from a single wave function”. In: *Phys. Rev. B* 103.7, p. 075102. ISSN: 2469-9950. DOI: 10.1103/PhysRevB.103.075102. URL: https://link.aps.org/doi/10.1103/PhysRevB.103.075102.

Deng, Dong-Ling, Xiaopeng Li, and S. Das Sarma (2017). “Machine learning topological states”. In: *Phys. Rev. B* 96 (19), p. 195145. DOI: 10.1103/PhysRevB.96.195145. URL: https://link.aps.org/doi/10.1103/PhysRevB.96.195145.

Dennis, Eric et al. (2002a). “Topological quantum memory”. In: *Journal of Mathematical Physics* 43.9, pp. 4452–4505.

– (2002b). “Topological quantum memory”. In: *Journal of Mathematical Physics* 43.9, pp. 4452–4505. URL: https://aip.scitation.org/doi/abs/10.1063/1.1499754.

Developers, Cirq (Aug. 2021). *Cirq*. Version v0.12.0. See full list of authors on Github: https://github .com/quantumlib/Cirq/graphs/contributors. DOI: 10.5281/zenodo.5182845. URL: https://doi.org/10.5281/zenodo.5182845.

Diakonikolas, Ilias, Daniel M Kane, and Vladimir Nikishkin (2014). “Testing identity of structured distributions”. In: *Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*. SIAM, pp. 1841–1854.

Dinur, Irit et al. (2006). “On the Fourier tails of bounded functions over the discrete cube”. In: *Israel Journal of Mathematics* 160, pp. 389–412.

Du, Simon S et al. (2019). “Graph neural tangent kernel: Fusing graph neural networks with graph kernels”. In: *arXiv preprint arXiv:1905.13192*.

Durr, Christoph and Peter Hoyer (1996). “A quantum algorithm for finding the minimum”. In: *arxiv preprint arXiv:quant-ph/9607014*. URL: `https://arxiv.org/abs/quant-ph/9607014`.

Dyson, Freeman J (1962). “The threefold way. Algebraic structure of symmetry groups and ensembles in quantum mechanics”. In: *Journal of Mathematical Physics* 3.6, pp. 1199–1215.

Ebadi, Sepehr et al. (2020). “Quantum Phases of Matter on a 256-Atom Programmable Quantum Simulator”. In: *arXiv e-prints*, arXiv:2012.12281. arXiv: `2012.12281 [quant-ph]`.

Efron, Bradley and Robert J. Tibshirani (1993). *An introduction to the bootstrap*. Vol. 57. Monographs on Statistics and Applied Probability. Chapman and Hall, New York, pp. xvi+436. ISBN: 0-412-04231-2. DOI: `10.1007/978-1-4899-4541-9`. URL: `https://doi.org/10.1007/978-1-4899-4541-9`.

Elben, A. et al. (May 2019a). “Statistical correlations between locally randomized measurements: A toolbox for probing entanglement in many-body quantum states”. In: *Phys. Rev. A* 99.5, p. 052323. DOI: `10.1103/PhysRevA.99.052323`. URL: `https://link.aps.org/doi/10.1103/PhysRevA.99.052323`.

– (2019b). “Statistical correlations between locally randomized measurements: A toolbox for probing entanglement in many-body quantum states”. In: *Phys. Rev. A* 99.5, p. 052323. DOI: `10.1103/PhysRevA.99.052323`. URL: `https://link.aps.org/doi/10.1103/PhysRevA.99.052323`.

Elben, Andreas, Steven T Flammia, et al. (2022). “The randomized measurement toolbox”. In: *arXiv preprint arXiv:2203.11374*.

Elben, Andreas, Richard Kueng, et al. (Nov. 2020a). “Mixed-State Entanglement from Local Randomized Measurements”. In: *Phys. Rev. Lett.* 125 (20), p. 200501. DOI: `10.1103/PhysRevLett.125.200501`. URL: `https://link.aps.org/doi/10.1103/PhysRevLett.125.200501`.

– (2020b). “Mixed-State Entanglement from Local Randomized Measurements”. In: *Phys. Rev. Lett.* 125 (20), p. 200501. DOI: `10.1103/PhysRevLett.125.200501`. URL: `https://link.aps.org/doi/10.1103/PhysRevLett.125.200501`.

Elben, Andreas, Jinlong Yu, et al. (2020). “Many-body topological invariants from randomized measurements in synthetic quantum matter”. In: *Science advances* 6.15, eaaz3666.

Emerson, Joseph, Robert Alicki, and Karol Życzkowski (2005). “Scalable noise estimation with random unitary operators”. In: *J. Opt. B Quantum Semiclass. Opt.* 7.10, S347–S352. ISSN: 1464-4266. DOI: 10.1088/1464-4266/7/10/021. URL: https://doi.org/10.1088/1464-4266/7/10/021.

Endres, Manuel et al. (2016). “Atom-by-atom assembly of defect-free one dimensional cold atom arrays”. In: *Science* 354.6315, pp. 1024–1027. ISSN: 0036-8075. DOI: 10.1126/science.aah3752. URL: http://www.sciencemag.org/lookup/doi/10.1126/science.aah3752.

Enk, S. J. van and C. W. J. Beenakker (2012). “Measuring Trρ^n on Single Copies of ρ Using Random Measurements”. In: *Phys. Rev. Lett.* 108 (11), p. 110503. DOI: 10.1103/PhysRevLett.108.110503. URL: https://link.aps.org/doi/10.1103/PhysRevLett.108.110503.

Eskenazis, Alexandros and Paata Ivanisvili (2022). “Learning low-degree functions from a logarithmic number of random queries”. In: *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 203–207.

Evans, Tim J, Robin Harper, and Steven T Flammia (2019). “Scalable Bayesian Hamiltonian learning”. In: *arXiv preprint arXiv:1912.07636*.

Farhi, Edward, Jeffrey Goldstone, and Sam Gutmann (2014a). “A quantum approximate optimization algorithm”. In: *arXiv preprint arXiv:1411.4028*.

– (2014b). “A Quantum Approximate Optimization Algorithm Applied to a Bounded Occurrence Constraint Problem”. In: *arXiv: Quantum Physics*.

Farhi, Edward, Jeffrey Goldstone, Sam Gutmann, et al. (2001). “A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem”. In: *Science* 292.5516, pp. 472–475. URL: https://science.sciencemag.org/content/292/5516/472.

Farhi, Edward and Hartmut Neven (2018). “Classification with quantum neural networks on near term processors”. In: *arXiv preprint arXiv:1802.06002*.

Fawzi, Hamza, James Saunderson, and Pablo A. Parrilo (Apr. 2019). “Semidefinite Approximations of the Matrix Logarithm”. In: *Found. Comput. Math.* 19.2, pp. 259–296. ISSN: 1615-3375. DOI: 10.1007/s10208-018-9385-0. URL: http://link.springer.com/10.1007/s10208-018-9385-0.

Fendley, Paul, K. Sengupta, and Subir Sachdev (2004). “Competing density-wave orders in a one-dimensional hard-boson model”. In: *Phys. Rev. B* 69 (7), p. 075106. DOI: 10.1103/PhysRevB.69.075106. URL: https://link.aps.org/doi/10.1103/PhysRevB.69.075106.

Ferris, Andrew J. and Guifre Vidal (2012). "Perfect sampling with unitary tensor networks". In: *Phys. Rev. B* 85 (16), p. 165146. DOI: 10.1103/PhysRevB.85.165146. URL: https://link.aps.org/doi/10.1103/PhysRevB.85.165146.

Flammia, Steven T et al. (2012). "Quantum tomography via compressed sensing: error bounds, sample complexity and efficient estimators". In: *New J. Phys.* 14.9, p. 095022.

Flammia, Steven T. and Yi-Kai Liu (June 2011). "Direct Fidelity Estimation from Few Pauli Measurements". In: *Phys. Rev. Lett.* 106 (23), p. 230501. DOI: 10.1103/PhysRevLett.106.230501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.106.230501.

Friis, Nicolai et al. (2019). "Entanglement certification from theory to experiment". In: *Nat. Rev. Phys.* 1.1, pp. 72–87. ISSN: 2522-5820. DOI: 10.1038/s42254-018-0003-5. URL: https://doi.org/10.1038/s42254-018-0003-5.

Gibbs, Joe et al. (2022). "Dynamical simulation via quantum machine learning with provable generalization". In: *arXiv preprint arXiv:2204.10269*.

Gilbert, Edgar N (1952). "A comparison of signalling alphabets". In: *The Bell system technical journal* 31.3, pp. 504–522.

Gilmer, Justin et al. (2017). "Neural message passing for quantum chemistry". In: *arXiv preprint arXiv:1704.01212*.

Gilyén, András, Seth Lloyd, and Ewin Tang (2018). "Quantum-inspired low-rank stochastic regression with logarithmic dependence on the dimension". In: *arXiv preprint arXiv:1811.04909*.

Giovannetti, Vittorio, Seth Lloyd, and Lorenzo Maccone (2011). "Advances in quantum metrology". In: *Nature photonics* 5.4, pp. 222–229. URL: https://www.nature.com/articles/nphoton.2011.35.

Glasser, Ivan et al. (2018). "Neural-Network Quantum States, String-Bond States, and Chiral Topological States". In: *Phys. Rev. X* 8 (1), p. 011006. DOI: 10.1103/PhysRevX.8.011006. URL: https://link.aps.org/doi/10.1103/PhysRevX.8.011006.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (2016). *Deep Learning*. The MIT Press.

Gosset, David and John Smolin (2019). "A Compressed Classical Description of Quantum States". In: *14th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2019)*. Vol. 135. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 8:1–8:9. ISBN: 978-3-95977-112-2. DOI: 10.4230/LIPIcs.TQC.2019.8. URL: http://drops.dagstuhl.de/opus/volltexte/2019/10400.

Gottesman, Daniel (1997). “Stabilizer codes and quantum error correction. Caltech Ph. D”. PhD thesis. Thesis, eprint: quant-ph/9705052.

Gottesman, Daniel, Thomas Jennewein, and Sarah Croke (Aug. 2012). “Longer-Baseline Telescopes Using Quantum Repeaters”. In: *Phys. Rev. Lett.* 109 (7), p. 070503. DOI: 10.1103/PhysRevLett.109.070503. URL: https://link.aps.org/doi/10.1103/PhysRevLett.109.070503.

Grant, Edward et al. (2019). “An initialization strategy for addressing barren plateaus in parametrized quantum circuits”. In: *Quantum* 3, p. 214. URL: https://quantum-journal.org/papers/q-2019-12-09-214/.

Greenberger, Daniel M., Michael A. Horne, and Anton Zeilinger (1989). “Going Beyond Bell’s Theorem”. In: *Bell’s Theorem, Quantum Theory and Conceptions of the Universe*. Dordrecht: Springer Netherlands, pp. 69–72. ISBN: 978-94-017-0849-4. DOI: 10.1007/978-94-017-0849-4_10. URL: https://doi.org/10.1007/978-94-017-0849-4_10.

Grimsley, Harper R et al. (2019). “An adaptive variational algorithm for exact molecular simulations on a quantum computer”. In: *Nat. Commun.* 10.1, pp. 1–9.

Gross, D., F. Krahmer, and R. Kueng (2015). “A partial derandomization of PhaseLift using spherical designs”. In: *J. Fourier Anal. Appl.* 21.2, pp. 229–266. ISSN: 1069-5869. DOI: 10.1007/s00041-014-9361-2. URL: https://doi.org/10.1007/s00041-014-9361-2.

Grover, Lov K (1996). “A fast quantum mechanical algorithm for database search”. In: *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pp. 212–219. URL: https://dl.acm.org/doi/10.1145/237814.237866.

Gu, Yinzheng (2013). “Moments of random matrices and weingarten functions”. PhD thesis. Queen’s University.

Gu, Zheng-Cheng and Xiao-Gang Wen (2009). “Tensor-entanglement-filtering renormalization approach and symmetry-protected topological order”. In: *Phys. Rev. B* 80.15, p. 155131. ISSN: 1098-0121. DOI: 10.1103/PhysRevB.80.155131. URL: https://link.aps.org/doi/10.1103/PhysRevB.80.155131.

Gühne, Otfried and Géza Tóth (2009). “Entanglement detection”. In: *Phys. Rep.* 474.1, pp. 1–75. ISSN: 0370-1573. DOI: https://doi.org/10.1016/j.physrep.2009.02.004. URL: http://www.sciencedirect.com/science/article/pii/S0370157309000623.

Guta, Madalin et al. (2020). “Fast state tomography with optimal error bounds”. In: *J. Phys. A*. URL: http://iopscience.iop.org/10.1088/1751-8121/ab8111.

Guţă, M et al. (2020a). “Fast state tomography with optimal error bounds”. In: *Journal of Physics A: Mathematical and Theoretical* 53.20, p. 204001.

– (2020b). “Fast state tomography with optimal error bounds”. In: *Journal of Physics A: Mathematical and Theoretical* 53.20, p. 204001.

Haagerup, Uffe (1981). “The best constants in the Khintchine inequality”. eng. In: *Studia Mathematica* 70.3, pp. 231–283. URL: http://eudml.org/doc/218383.

Haah, Jeongwan et al. (2017). “Sample-optimal tomography of quantum states”. In: *IEEE T. Inform. Theory* 63.9, pp. 5628–5641.

Hadfield, Charles et al. (2020). “Measurements of Quantum Hamiltonians with Locally-Biased Classical Shadows”. In: *preprint arXiv:2006.15788*.

Haegeman, Jutho et al. (2012). “Order Parameter for Symmetry-Protected Phases in One Dimension”. In: *Phys. Rev. Lett.* 109.5, p. 050402. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.109.050402. URL: https://link.aps.org/doi/10.1103/PhysRevLett.109.050402.

Haldane, F.D.M. (1983). “Continuum dynamics of the 1-D Heisenberg antiferromagnet: Identification with the O(3) nonlinear sigma model”. In: *Phys. Lett. A* 93.9, pp. 464–468. ISSN: 03759601. DOI: 10.1016/0375-9601(83)90631-X. URL: https://linkinghub.elsevier.com/retrieve/pii/037596018390631X.

Halevy, Alon, Peter Norvig, and Fernando Pereira (2009). “The unreasonable effectiveness of data”. In: *IEEE Intelligent Systems* 24.2, pp. 8–12. URL: https://www.computer.org/csdl/magazine/ex/2009/02/mex2009020008/13rRUy0HYOb.

Hallgren, Sean, Eun Young Lee, and Ojas Parekh (2020). “An Approximation Algorithm for the MAX-2-Local Hamiltonian Problem”. In: *APPROX-RANDOM*.

Harrow, Aram W and Ashley Montanaro (2017a). “Extremal eigenvalues of local Hamiltonians”. In: *Quantum* 1, p. 6.

– (2017b). “Quantum computational supremacy”. In: *Nature* 549.7671, pp. 203–209. URL: https://www.nature.com/articles/nature23458.

Harrow, Aram W. (2013). “The church of the symmetric subspace”. In: *arXiv 1308.6595*.

Hastings, Matthew B (2010). “Locality in quantum systems”. In: *arXiv 1008.5137*.

Hastings, Matthew B and Tohru Koma (2006). “Spectral gap and exponential decay of correlations”. In: *Communications in mathematical physics* 265.3, pp. 781–804.

Hastings, Matthew B and Ryan O’Donnell (2022). “Optimizing strongly interacting fermionic Hamiltonians”. In: *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 776–789.

Hastings, Matthew B and Xiao-Gang Wen (2005). “Quasiadiabatic continuation of quantum states: The stability of topological ground-state degeneracy and emergent gauge invariance”. In: *Phys. Rev. B* 72.4, p. 045141.

Hastings, Matthew B. and Spyridon Michalakis (2015). “Quantization of Hall Conductance for Interacting Electrons on a Torus”. In: *Commun. Math. Phys.* 334.1, pp. 433–471. ISSN: 0010-3616. DOI: `10.1007/s00220-014-2167-x`. URL: `http://link.springer.com/10.1007/s00220-014-2167-x`.

Havlicek, Vojtěch et al. (2019). “Supervised learning with quantum-enhanced feature spaces”. In: *Nature* 567.7747, pp. 209–212.

Hazan, Elad, Tomer Koren, and Nati Srebro (2011). “Beating SGD: Learning SVMs in Sublinear Time.” In: *NIPS*. Citeseer, pp. 1233–1241.

Helgaker, Trygve, Poul Jorgensen, and Jeppe Olsen (2014). *Molecular electronic-structure theory*. John Wiley & Sons.

Hochreiter, Sepp and Jürgen Schmidhuber (Nov. 1997). “Long Short-Term Memory”. In: *Neural Computation* 9.8, pp. 1735–1780. ISSN: 0899-7667. DOI: `10.1162/neco.1997.9.8.1735`.

Hoeffding, Wassily (1992). “A class of statistics with asymptotically normal distribution”. In: *Breakthroughs in Statistics*. Springer, pp. 308–334.

Hohenberg, P. and W. Kohn (1964). “Inhomogeneous Electron Gas”. In: *Phys. Rev.* 136 (3B), B864–B871. DOI: `10.1103/PhysRev.136.B864`. URL: `https://link.aps.org/doi/10.1103/PhysRev.136.B864`.

Hohenberg, Pierre and Walter Kohn (1964). “Inhomogeneous electron gas”. In: *Physical review* 136.3B, B864. URL: `https://journals.aps.org/pr/abstract/10.1103/PhysRev.136.B864`.

Holevo, A. S. (1973). “Some estimates of the information transmitted by quantum communication channels”. In: *Probl. Inf. Transm.* 9.3, pp. 177–183. URL: `http://www.ams.org/mathscinet-getitem?mr=456936`.

Holevo, Alexander Semenovich (1973). “Bounds for the quantity of information transmitted by a quantum communication channel”. In: *Problemy Peredachi Informatsii* 9.3, pp. 3–11.

Horodecki, Ryszard et al. (2009). “Quantum entanglement”. In: *Rev. Mod. Phys.* 81.2, p. 865.

Hradil, Z. (Mar. 1997). “Quantum-state estimation”. In: *Phys. Rev. A* 55 (3), R1561–R1564. DOI: `10.1103/PhysRevA.55.R1561`. URL: `https://link.aps.org/doi/10.1103/PhysRevA.55.R1561`.

Hu, Hong-Ye and Yi-Zhuang You (2021). “Hamiltonian-Driven Shadow Tomography of Quantum States”. In: *arXiv preprint arXiv:2102.10132*.

Huang, Hsin-Yuan, Michael Broughton, Jordan Cotler, et al. (2022). “Quantum advantage in learning from experiments”. In: *Science* 376.6598, pp. 1182–1186. DOI: `10.1126/science.abn7293`.

Huang, Hsin-Yuan, Michael Broughton, Masoud Mohseni, Ryan Babbush, Sergio Boixo, Hartmut Neven, and Jarrod R McClean (2021a). “Power of data in quantum machine learning”. In: *Nat. Commun.* 12.1, pp. 1–9.

– (May 2021b). “Power of data in quantum machine learning”. In: *Nature Communications* 12.1, p. 2631. DOI: 10.1038/s41467-021-22539-9.

Huang, Hsin-Yuan, Sitan Chen, and John Preskill (2023). “Learning to predict arbitrary quantum processes”. In: *PRX Quantum* 4.4, p. 040337. DOI: 10.1103/PRXQuantum.4.040337.

Huang, Hsin-Yuan, Steven T Flammia, and John Preskill (2022). “Foundations for learning from noisy quantum experiments”. In: *arXiv preprint arXiv:2204.13691*.

Huang, Hsin-Yuan and Richard Kueng (2019). “Predicting Features of Quantum Systems using Classical Shadows”. In: *arXiv preprint arXiv:1908.08909*.

Huang, Hsin-Yuan, Richard Kueng, and John Preskill (2020). “Predicting many properties of a quantum system from very few measurements”. In: *Nature Physics*. DOI: 10.1038/s41567-020-0932-7.

– (2021). “Information-theoretic bounds on quantum advantage in machine learning”. In: *Phys. Rev. Lett.* 126 (19), p. 190505. DOI: 10.1103/PhysRevLett.126.190505.

Huang, Hsin-Yuan, Richard Kueng, Giacomo Torlai, et al. (Dec. 2021). “Code for Provably efficient machine learning for quantum many-body problems”. In: *Github*, https://github.com/hsinyuan-huang/provable-ml–quantum. URL: https://github.com/hsinyuan-huang/provable-ml-quantum.

– (2022). “Provably efficient machine learning for quantum many-body problems”. In: *Science* 377.6613, eabk3333. DOI: 10.1126/science.abk3333.

Huggins, William J et al. (2020). “Virtual Distillation for Quantum Error Mitigation”. In: *arXiv preprint arXiv:2011.07064*.

Jacot, Arthur, Franck Gabriel, and Clément Hongler (2018). “Neural tangent kernel: Convergence and generalization in neural networks”. In: *NeurIPS*, pp. 8571–8580.

Jerrum, Mark R., Leslie G. Valiant, and Vijay V. Vazirani (1986). “Random generation of combinatorial structures from a uniform distribution”. In: *Theoret. Comput. Sci.* 43.2-3, pp. 169–188. ISSN: 0304-3975. DOI: 10.1016/0304-3975(86)90174-X. URL: https://doi.org/10.1016/0304-3975(86)90174-X.

Ji, Zhengfeng, Yi-Kai Liu, and Fang Song (2018). “Pseudorandom quantum states”. In: *Annual International Cryptology Conference*. Springer, pp. 126–152.

Joachims, Thorsten (1999). “Making Large-Scale Support Vector Machine Learning Practical”. In: *Advances in Kernel Methods: Support Vector Learning*. Cambridge, MA, USA: MIT Press, pp. 169–184. ISBN: 0262194163.

Jolliffe, Ian T (1986). “Principal components in regression analysis”. In: *Principal component analysis*. Springer, pp. 129–155. URL: https://link.springer.com/chapter/10.1007/978-1-4757-1904-8_8.

Kandala, Abhinav et al. (2017). “Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets”. In: *Nature* 549.7671, pp. 242–246.

Kapustin, Anton and Nikita Sopenko (2020). “Hall conductance and the statistics of flux insertions in gapped interacting lattice systems”. In: *J. Math. Phys.* 61.10, p. 101901. ISSN: 0022-2488. DOI: 10.1063/5.0022944. URL: http://aip.scitation.org/doi/10.1063/5.0022944.

Karnin, Zohar et al. (2012). “Unsupervised SVMs: On the Complexity of the Furthest Hyperplane Problem”. In: *Proceedings of the 25th Annual Conference on Learning Theory*. Ed. by Shie Mannor, Nathan Srebro, and Robert C. Williamson. Vol. 23. Proceedings of Machine Learning Research. Edinburgh, Scotland: JMLR Workshop and Conference Proceedings, pp. 2.1–2.17. URL: http://proceedings.mlr.press/v23/karnin12.html.

Kawai, Hiroki and Yuya O Nakagawa (2020). “Predicting excited states from ground state wavefunction by supervised quantum machine learning”. In: *Machine Learning: Science and Technology* 1.4, p. 045027.

Kempe, Julia, Alexei Kitaev, and Oded Regev (2006). “The complexity of the local Hamiltonian problem”. In: *Siam journal on computing* 35.5, pp. 1070–1097.

Kingma, Diederik P and Jimmy Ba (2014). “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980*.

Kitaev, A Yu (2003). “Fault-tolerant quantum computation by anyons”. In: *Annals of Physics* 303.1, pp. 2–30.

Kitaev, A. Y. and John Preskill (2006). “Topological entanglement entropy”. In: *Phys. Rev. Lett.* 96.11, p. 110404.

Kitaev, Alexei Yu. (2006). “Anyons in an exactly solved model and beyond”. In: *Ann. Phys.* 321.1, pp. 2–111. ISSN: 00034916. DOI: 10.1016/j.aop.2005.10.005. URL: http://linkinghub.elsevier.com/retrieve/pii/S0003491605002381.

Knill, Emanuel et al. (2008). “Randomized benchmarking of quantum gates”. In: *Physical Review A* 77.1, p. 012307.

Knuth, Donald E and Arvind Raghunathan (1992). “The problem of compatible representatives”. In: *SIAM Journal on Discrete Mathematics* 5.3, pp. 422–427.

Koch, Gregory, Richard Zemel, Ruslan Salakhutdinov, et al. (2015). “Siamese neural networks for one-shot image recognition”. In: *ICML deep learning workshop*. Vol. 2. Lille.

Koczor, Bálint (2021a). "Exponential error suppression for near-term quantum devices". In: *Physical Review X* 11.3, p. 031057. URL: https://journals.aps.org/prx/abstract/10.1103/PhysRevX.11.031057.

– (2021b). "The dominant eigenvector of a noisy quantum state". In: *New Journal of Physics* 23.12, p. 123047. URL: https://iopscience.iop.org/article/10.1088/1367-2630/ac37ae/meta.

Koenig, Robert and John A. Smolin (2014). "How to efficiently select an arbitrary Clifford group element". In: *J. Math. Phys.* 55.12, pp. 122202, 12. ISSN: 0022-2488. DOI: 10.1063/1.4903507. URL: https://doi.org/10.1063/1.4903507.

Koh, Dax Enshan and Sabee Grewal (2020). "Classical Shadows with Noise". In: *arXiv preprint arXiv:2011.11580*.

Kohn, W. (1999). "Nobel Lecture: Electronic structure of matter—wave functions and density functionals". In: *Rev. Mod. Phys.* 71 (5), pp. 1253–1266. DOI: 10.1103/RevModPhys.71.1253. URL: https://link.aps.org/doi/10.1103/RevModPhys.71.1253.

Kokail, Christian et al. (2019). "Self-verifying variational quantum simulation of lattice models". In: *Nature* 569.7756, pp. 355–360.

Kottmann, Korbinian et al. (May 2021). "Unsupervised mapping of phase diagrams of 2D systems from infinite projected entangled-pair states via deep anomaly detection". In: arXiv: 2105.09089. URL: http://arxiv.org/abs/2105.09089.

Kozen, Dexter C (1992). *The design and analysis of algorithms*. Springer Science & Business Media.

Krogh, Anders and John A Hertz (1992). "A simple weight decay can improve generalization". In: *Advances in neural information processing systems*, pp. 950–957.

Kueng, Richard and David Gross (2015). "Qubit stabilizer states are complex projective 3-designs". In: *arXiv preprint arXiv:1510.02767*.

Kueng, Richard, Holger Rauhut, and Ulrich Terstiege (2017). "Low rank matrix recovery from rank one measurements". In: *Applied and Computational Harmonic Analysis* 42.1, pp. 88–116.

Kunjummen, Jonathan et al. (2021). "Shadow process tomography of quantum channels". In: *arXiv preprint arXiv:2110.03629*.

Kuwahara, Tomotaka and Keiji Saito (2020). "Strictly linear light cones in long-range interacting systems of arbitrary dimensions". In: *Physical Review X* 10.3, p. 031010.

Labuhn, Henning et al. (2016). "Tunable two-dimensional arrays of single Rydberg atoms for realizing quantum Ising models". In: *Nature* 534. URL: https://doi.org/10.1038/nature18274.

Landsberg, Joseph M (2012). "Tensors: geometry and applications". In: *Representation theory* 381.402, p. 3.

Laneve, Alessandro et al. (2021). "Experimental multi-state quantum discrimination through a Quantum network". In: *arXiv preprint arXiv:2107.09968*. URL: https://arxiv.org/abs/2107.09968.

Lanyon, B. P. et al. (Sept. 2017). "Efficient tomography of a quantum many-body system". In: *Nat. Phys.* 13. URL: https://doi.org/10.1038/nphys4244.

LaRose, Ryan and Brian Coyle (2020). "Robust data encodings for quantum classifiers". In: *Physical Review A* 102 (3), p. 032420. DOI: 10.1103/PhysRevA.102.032420. URL: https://link.aps.org/doi/10.1103/PhysRevA.102.032420.

LaRose, Ryan, Arkin Tikku, et al. (2019). "Variational quantum state diagonalization". In: *npj Quantum Information* 5.1, pp. 1–10.

Lauk, Nikolai et al. (2020). "Perspectives on quantum transduction". In: *Quantum Science and Technology* 5.2, p. 020501. URL: https://iopscience.iop.org/article/10.1088/2058-9565/ab788a/meta.

LeCam, L. (1973). "Convergence of Estimates Under Dimensionality Restrictions". In: *The Annals of Statistics* 1.1, pp. 38–53. ISSN: 00905364. URL: http://www.jstor.org/stable/2958155 (visited on 08/27/2022).

LeCun, Yann, Corinna Cortes, and CJ Burges (2010). "MNIST handwritten digit database". In: *ATT Labs [Online]* 2. URL: http://yann.lecun.com/exdb/mnist.

Ledoux, Michel and Michel Talagrand (2013). *Probability in Banach Spaces: isoperimetry and processes*. Springer Science & Business Media.

Leifer, Matthew S and David Poulin (2008). "Quantum graphical models and belief propagation". In: *Annals of Physics* 323.8, pp. 1899–1946. URL: https://linkinghub.elsevier.com/retrieve/pii/S0003491607001509.

Levin, Michael and Xiao-Gang Wen (2006). "Detecting topological order in a ground state wave function". In: *Phys. Rev. Lett.* 96.11, p. 110405.

Levine, Harry et al. (2018). "High-fidelity control and entanglement of rydberg-atom qubits". In: *Phys. Rev. Lett.* 121.12, p. 123603.

Levy, Ryan, Di Luo, and Bryan K Clark (2021). "Classical shadows for quantum process tomography on near-term quantum computers". In: *arXiv preprint arXiv:2110.02965*.

Lewis, Laura et al. (2024). “Improved machine learning algorithm for predicting ground state properties”. In: *nature communications* 15.1, p. 895. DOI: 10.1038/s41467-024-45014-7.

Li, Hui and F. D. M. Haldane (2008). “Entanglement Spectrum as a Generalization of Entanglement Entropy: Identification of Topological Order in Non-Abelian Fractional Quantum Hall Effect States”. In: *Phys. Rev. Lett.* 101.1, p. 010504. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.101.010504. URL: https://link.aps.org/doi/10.1103/PhysRevLett.101.010504.

Li, Zhiyuan et al. (2019). “Enhanced convolutional neural tangent kernels”. In: *arXiv preprint arXiv:1911.00809*.

Lichtenstein, D. (1982). “Planar Formulae and Their Uses”. In: *SIAM J. Comput.* 11, pp. 329–343.

Lieb, Elliott, Theodore Schultz, and Daniel Mattis (1961). “Two soluble models of an antiferromagnetic chain”. In: *Annals of Physics* 16.3, pp. 407–466.

Lieb, Elliott H. and Derek W. Robinson (Sept. 1972). “The finite group velocity of quantum spin systems”. In: *Commun. Math. Phys.* 28.3, pp. 251–257. ISSN: 0010-3616. DOI: 10.1007/BF01645779. URL: http://link.springer.com/10.1007/BF01645779.

Littlewood, John E (1930). “On bounded bilinear forms in an infinite number of variables”. In: *The Quarterly Journal of Mathematics* 1, pp. 164–174.

Liu, Yunchao, Srinivasan Arunachalam, and Kristan Temme (2020). “A rigorous and robust quantum speed-up in supervised machine learning”. In: arXiv: 2010.02174 [quant-ph].

Lloyd, Seth (1996). “Universal quantum simulators”. In: *Science*, pp. 1073–1078.

Lloyd, Seth, Masoud Mohseni, and Patrick Rebentrost (2014). “Quantum principal component analysis”. In: *Nat. Phys.* 10.9, pp. 631–633.

Lloyd, Seth, Maria Schuld, et al. (2020). “Quantum embeddings for machine learning”. In: *arXiv preprint arXiv:2001.03622*. URL: https://arxiv.org/abs/2008.08605.

Lvovsky, Alexander I, Barry C Sanders, and Wolfgang Tittel (2009). “Optical quantum memory”. In: *Nature photonics* 3.12, pp. 706–714. URL: https://www.nature.com/articles/nphoton.2009.231?message=remove&lang=en.

Ma, Shang-keng, Chandan Dasgupta, and Chin-kun Hu (1979). “Random antiferromagnetic chain”. In: *Phys. Rev. Lett.* 43.19, p. 1434.

Magesan, Easwar, J. M. Gambetta, and Joseph Emerson (May 2011). “Scalable and Robust Randomized Benchmarking of Quantum Processes”. In: *Phys. Rev. Lett.* 106 (18), p. 180504. DOI: 10.1103/PhysRevLett.106.180504. URL: https://link.aps.org/doi/10.1103/PhysRevLett.106.180504.

Matsumoto, Sho (2013). “Weingarten calculus for matrix ensembles associated with compact symmetric spaces”. In: *arXiv:1301.5401*.

McClean, Jarrod R, Sergio Boixo, et al. (2018). “Barren plateaus in quantum neural network training landscapes”. In: *Nature communications* 9.1, pp. 1–6.

McClean, Jarrod R, Matthew P Harrigan, et al. (2020). “Low depth mechanisms for quantum optimization”. In: *arXiv preprint arXiv:2008.08615*. URL: https://arxiv.org/abs/2008.08615.

McClean, Jarrod R, Jonathan Romero, et al. (2016). “The theory of variational hybrid quantum-classical algorithms”. In: *New Journal of Physics* 18.2, p. 023023.

Melnikov, Alexey A et al. (2018). “Active learning machine learns to create new quantum experiments”. In: *Proc. Natl. Acad. Sci. U.S.A.* 115.6, pp. 1221–1226.

Mercer, James (Jan. 1909). “Functions of positive and negative type, and their connection the theory of integral equations”. In: *Philos. T. Roy. Soc. A* 209.441-458, pp. 415–446. ISSN: 0264-3952. DOI: 10.1098/rsta.1909.0016. URL: https://royalsocietypublishing.org/doi/10.1098/rsta.1909.0016.

Merkel, Seth T et al. (2013). “Self-consistent quantum process tomography”. In: *Physical Review A* 87.6, p. 062119.

Micchelli, Charles A, Yuesheng Xu, and Haizhang Zhang (2006). “Universal kernels”. In: *Journal of Machine Learning Research* 7.Dec, pp. 2651–2667. URL: https://www.jmlr.org/papers/volume7/micchelli06a/micchelli06a.pdf.

Mohri, Mehryar, Afshin Rostamizadeh, and Ameet Talwalkar (2018). *Foundations of machine learning*. The MIT Press.

Mohseni, M. and D. A. Lidar (June 2007). “Direct characterization of quantum dynamics: General theory”. In: *Phys. Rev. A* 75 (6), p. 062331. DOI: 10.1103/PhysRevA.75.062331. URL: https://link.aps.org/doi/10.1103/PhysRevA.75.062331.

Mohseni, M., A. T. Rezakhani, and D. A. Lidar (Mar. 2008). “Quantum-process tomography: Resource analysis of different strategies”. In: *Phys. Rev. A* 77 (3), p. 032322. DOI: 10.1103/PhysRevA.77.032322. URL: https://link.aps.org/doi/10.1103/PhysRevA.77.032322.

Mohseni, Masoud, Ali T Rezakhani, and Daniel A Lidar (2008). “Quantum-process tomography: Resource analysis of different strategies”. In: *Phys. Rev. A* 77.3, p. 032322.

Moreno, Javier Robledo, Giuseppe Carleo, and Antoine Georges (2020). “Deep learning the hohenberg-kohn maps of density functional theory”. In: *Physical Review Letters* 125.7, p. 076402.

Motwani, Rajeev and Prabhakar Raghavan (1995). *Randomized Algorithms*. Cambridge University Press. ISBN: 0-521-47465-5.

Murphy, Kevin P (2012). *Machine learning: a probabilistic perspective*. MIT press.

Nachtergaele, Bruno, Robert Sims, and Amanda Young (2018). "Lieb–Robinson bounds, the spectral flow, and stability of the spectral gap for lattice fermion systems". In: *Mathematical Problems in Quantum Physics* 717.

Nadaraya, Elizbar A (1964). "On estimating regression". In: *Theory of Probability & Its Applications* 9.1, pp. 141–142.

Nakamura, Masaaki and Synge Todo (2002). "Order Parameter to Characterize Valence-Bond-Solid States in Quantum Spin Chains". In: *Phys. Rev. Lett.* 89.7, p. 077204. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.89.077204. URL: https://link.aps.org/doi/10.1103/PhysRevLett.89.077204.

Nandkishore, Rahul and David A Huse (2015). "Many-body localization and thermalization in quantum statistical mechanics". In: *Annu. Rev. Condens. Matter Phys.* 6.1, pp. 15–38.

Nemirovsky, A. S. and D. B. and Yudin (1983). *Problem complexity and method efficiency in optimization*. A Wiley-Interscience Publication. Translated from the Russian and with a preface by E. R. Dawson, Wiley-Interscience Series in Discrete Mathematics. John Wiley & Sons, Inc., New York, pp. xv+388. ISBN: 0-471-10345-4.

Neven, Hartmut et al. (2009). "Training a large scale classifier with the quantum adiabatic algorithm". In: *arXiv preprint arXiv:0912.0779*. URL: https://arxiv.org/abs/0912.0779.

Nielsen, Michael A and Isaac L Chuang (n.d.). "Quantum computation and quantum information". In: ().

– (2000). *Quantum computation and quantum information*. Cambridge University Press, Cambridge, pp. xxvi+676. ISBN: 0-521-63503-9.

Nieuwenburg, Evert P. L. van, Ye-Hua Liu, and Sebastian D. Huber (2017). "Learning phase transitions by confusion". In: *Nat. Phys.* 13, p. 435. URL: https://doi.org/10.1038/nphys4037.

Nomura, Yusuke et al. (2017). "Restricted Boltzmann machine learning for solving strongly correlated quantum systems". In: *Phys. Rev. B* 96 (20), p. 205152. DOI: 10.1103/PhysRevB.96.205152. URL: https://link.aps.org/doi/10.1103/PhysRevB.96.205152.

Novak, Roman, Lechao Xiao, Jiri Hron, Jaehoon Lee, Alexander A Alemi, et al. (2019). "Neural tangents: Fast and easy infinite neural networks in python". In: *arXiv preprint arXiv:1912.02803*. URL: https://arxiv.org/abs/1912.02803.

Novak, Roman, Lechao Xiao, Jiri Hron, Jaehoon Lee, Alexander A. Alemi, et al. (2020). "Neural Tangents: Fast and Easy Infinite Neural Networks in Python". In: *International Conference on Learning Representations*. URL: `https://github.com/google/neural-tangents`.

O'Brien, Jeremy L et al. (2004). "Quantum process tomography of a controlled-NOT gate". In: *Physical review letters* 93.8, p. 080502.

O'Donnell, Ryan and John Wright (2016). "Efficient Quantum Tomography". In: *Proceedings of the Forty-eighth Annual ACM Symposium on Theory of Computing*. STOC '16. Cambridge, MA, USA: ACM, pp. 899–912. ISBN: 978-1-4503-4132-5. DOI: `10.1145/2897518.2897544`. URL: `http://doi.acm.org/10.1145/2897518.2897544`.

O'Gorman, Bryan (2022). "Fermionic tomography and learning". In: *arXiv preprint arXiv:2207.14787*.

Ohliger, M, V Nesme, and J Eisert (Jan. 2013a). "Efficient and feasible state tomography of quantum many-body systems". In: *New J. Phys.* 15.1, p. 015024. ISSN: 1367-2630. DOI: `10.1088/1367-2630/15/1/015024`. URL: `http://dx.doi.org/10.1088/1367-2630/15/1/015024`.

– (Jan. 2013b). "Efficient and feasible state tomography of quantum many-body systems". In: *New Journal of Physics* 15.1, p. 015024. ISSN: 1367-2630. DOI: `10.1088/1367-2630/15/1/015024`. URL: `http://dx.doi.org/10.1088/1367-2630/15/1/015024`.

Osborne, Tobias J (2007). "Simulating adiabatic evolution of gapped spin systems". In: *Phys. Rev. A* 75.3, p. 032321.

Paini, Marco and Amir Kalev (2019). "An approximate description of quantum states". In: *preprint arXiv:1910.10543*.

Parekh, Ojas and Kevin Thompson (2020). "Beating random assignment for approximating quantum 2-local Hamiltonian problems". In: *arXiv:2012.12347*.

Parr, Robert G (1980). "Density functional theory of atoms and molecules". In: *Horizons of quantum chemistry*. Springer, pp. 5–15.

Pearson, Karl (1901). "LIII. On lines and planes of closest fit to systems of points in space". In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2.11, pp. 559–572.

Pedregosa, F. et al. (2011). "Scikit-learn: Machine Learning in Python". In: *J. Mach. Learn. Res.* 12, pp. 2825–2830.

Peruzzo, Alberto et al. (2014). "A variational eigenvalue solver on a photonic quantum processor". In: *Nat. Commun.* 5.1, p. 4213. DOI: `10.1038/ncomms5213`. URL: `https://doi.org/10.1038/ncomms5213`.

Piddock, Stephen and Ashley Montanaro (2015). "The complexity of antiferromagnetic interactions and 2D lattices". In: *arXiv preprint arXiv:1506.04014*.

Pollmann, Frank and Ari M Turner (2012). "Detection of symmetry-protected topological phases in one dimension". In: *Phys. Rev. B* 86.12, p. 125441.

Pollmann, Frank, Ari M. Turner, et al. (2010). "Entanglement spectrum of a topological phase in one dimension". In: *Phys. Rev. B* 81.6, p. 064439. ISSN: 1098-0121. DOI: 10.1103/PhysRevB.81.064439. URL: https://link.aps.org/doi/10.1103/PhysRevB.81.064439.

Preskill, John (2018). "Quantum Computing in the NISQ era and beyond". In: *Quantum* 2, p. 79.

Qiao, Zhuoran et al. (2020). "OrbNet: Deep learning for quantum chemistry using symmetry-adapted atomic-orbital features". In: *J. Chem. Phys.* 153.12, p. 124111.

Raghavan, Prabhakar (1988). "Probabilistic construction of deterministic algorithms: approximating packing integer programs". In: *Journal of Computer and System Sciences* 37.2, pp. 130–143.

Raussendorf, Robert and Hans J. Briegel (May 2001). "A One-Way Quantum Computer". In: *Phys. Rev. Lett.* 86 (22), pp. 5188–5191. DOI: 10.1103/PhysRevLett.86.5188. URL: https://link.aps.org/doi/10.1103/PhysRevLett.86.5188.

Read, Nicholas (2012). "Topological phases and quasiparticle braiding". In: *Phys. Today* 65.7, p. 38. ISSN: 00319228. DOI: 10.1063/PT.3.1641. URL: http://scitation.aip.org/content/aip/magazine/physicstoday/article/65/7/10.1063/PT.3.1641.

Rebentrost, Patrick, Masoud Mohseni, and Seth Lloyd (Sept. 2014). "Quantum Support Vector Machine for Big Data Classification". In: *Phys. Rev. Lett.* 113 (13), p. 130503. DOI: 10.1103/PhysRevLett.113.130503. URL: https://link.aps.org/doi/10.1103/PhysRevLett.113.130503.

Refael, Gil and Ehud Altman (2013). "Strong disorder renormalization group primer and the superfluid–insulator transition". In: *C. R. Phys.* 14.8, pp. 725–739.

Regev, Oded (Sept. 2009). "On Lattices, Learning with Errors, Random Linear Codes, and Cryptography". In: *J. ACM* 56.6. ISSN: 0004-5411. DOI: 10.1145/1568318.1568324. URL: https://doi.org/10.1145/1568318.1568324.

– (2010). "The learning with errors problem". In: *Invited survey in CCC* 7.30, p. 11.

Rem, Benno S. et al. (2019). "Identifying quantum phase transitions using artificial neural networks on experimental data". In: *Nat. Phys.* 15.9, pp. 917–920. ISSN: 1745-2473. DOI: 10.1038/s41567-019-0554-0. URL: http://www.nature.com/articles/s41567-019-0554-0.

Renes, Joseph M. et al. (2004). "Symmetric informationally complete quantum measurements". In: *J. Math. Phys.* 45.6, pp. 2171–2180. ISSN: 0022-2488. DOI: 10.1063/1.1737053. URL: https://doi.org/10.1063/1.1737053.

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin (2016). “" Why should i trust you?" Explaining the predictions of any classifier”. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144.

Rigollet, Phillippe and Jan-Christian Hütter (2015). “High dimensional statistics”. In: *Lecture notes for course 18S997* 813. Available at `http://www-math.mit.edu/~rigollet/PDFs/RigNotes17.pdf`, p. 814.

Rivest, Ronald L, Adi Shamir, and Leonard Adleman (1978). “A method for obtaining digital signatures and public-key cryptosystems”. In: *Communications of the ACM* 21.2, pp. 120–126.

Rodriguez-Nieva, Joaquin F and Mathias S Scheurer (2019). “Identifying topological order through unsupervised machine learning”. In: *Nat. Phys.* 15.8, pp. 790–795.

Roth, I. et al. (Oct. 2018). “Recovering Quantum Gates from Few Average Gate Fidelities”. In: *Phys. Rev. Lett.* 121 (17), p. 170502. DOI: `10.1103/PhysRevLett.121.170502`. URL: `https://link.aps.org/doi/10.1103/PhysRevLett.121.170502`.

Rouzé, Cambyse, Melchior Wirth, and Haonan Zhang (2022). “Quantum Talagrand, KKL and Friedgut’s theorems and the learnability of quantum Boolean functions”. In: DOI: `10.48550/ARXIV.2209.07279`. URL: `https://arxiv.org/abs/2209.07279`.

Runge, Erich and Eberhard KU Gross (1984). “Density-functional theory for time-dependent systems”. In: *Physical Review Letters* 52.12, p. 997. URL: `https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.52.997`.

Sakurai, J. J. and Jim Napolitano (2017). *Modern Quantum Mechanics*. 2nd ed. Cambridge University Press. DOI: `10.1017/9781108499996`.

Sandvik, Anders W. (1999). “Stochastic series expansion method with operator-loop update”. In: *Phys. Rev. B* 59 (22), R14157–R14160. DOI: `10.1103/PhysRevB.59.R14157`. URL: `https://link.aps.org/doi/10.1103/PhysRevB.59.R14157`.

Schauß, P. et al. (2015). “Crystallization in Ising quantum magnets”. In: *Science* 347.6229, pp. 1455–1458. ISSN: 0036-8075. DOI: `10.1126/science.1258351`.

Schölkopf, Bernhard, Ralf Herbrich, and Alex J Smola (2001). “A generalized representer theorem”. In: *International conference on computational learning theory*. Springer, pp. 416–426.

Schölkopf, Bernhard, Alexander Smola, and Klaus-Robert Müller (1998). “Nonlinear component analysis as a kernel eigenvalue problem”. In: *Neural computation* 10.5, pp. 1299–1319.

Schölkopf, Bernhard, Alexander J Smola, Francis Bach, et al. (2002). *Learning with kernels: support vector machines, regularization, optimization, and beyond.*

Scholl, Pascal et al. (2020). “Programmable quantum simulation of 2D antiferromagnets with hundreds of Rydberg atoms”. In: *arXiv e-prints*, arXiv:2012.12268, arXiv:2012.12268. arXiv: 2012.12268 [quant-ph].

Schollwoeck, Ulrich (2011). “The density-matrix renormalization group in the age of matrix product states”. In: *Ann. Phys.* 326.1. January 2011 Special Issue, pp. 96–192. ISSN: 0003-4916. DOI: https://doi.org/10.1016/j.aop.2010.09.012. URL: http://www.sciencedirect.com/science/article/pii/S0003491610001752.

Schreiber, Franz J, Jens Eisert, and Johannes Jakob Meyer (2022). “Classical surrogates for quantum learning models”. In: *arXiv preprint arXiv:2206.11740*.

Schuld, Maria, Alex Bocharov, et al. (2020). “Circuit-centric quantum classifiers”. In: *Physical Review A* 101.3, p. 032308. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.101.032308.

Schuld, Maria and Nathan Killoran (2019). “Quantum machine learning in feature Hilbert spaces”. In: *Phys. Rev. Lett.* 122.4, p. 040504.

Schuld, Maria, Ryan Sweke, and Johannes Jakob Meyer (2020). “The effect of data encoding on the expressive power of variational quantum machine learning models”. In: *arXiv preprint arXiv:2008.08605*. URL: https://arxiv.org/abs/2008.08605.

Schütt, KT et al. (2019). “Unifying machine learning and quantum chemistry with a deep neural network for molecular wavefunctions”. In: *Nat. Commun.* 10.1, pp. 1–10.

Scott, A. J. (2008). “Optimizing quantum process tomography with unitary 2-designs”. In: *J. Phys.* A41, p. 055308. DOI: 10.1088/1751-8113/41/5/055308. arXiv: 0711.1017 [quant-ph].

Sentis, Gael, Esteban Martinez-Vargas, and Ramon Munoz-Tapia (2022). “Online identification of symmetric pure states”. In: *Quantum* 6, p. 658. URL: https://quantum-journal.org/papers/q-2022-02-21-658/.

Servedio, Rocco A and Steven J Gortler (2004). “Equivalences and separations between quantum and classical learnability”. In: *SIAM J. Comput.* 33.5, pp. 1067–1092.

Settles, Burr (2009). “Active learning literature survey”. In.

Shalev-Shwartz, Shai, Ohad Shamir, and Shaked Shammah (Aug. 2017). “Failures of Gradient-Based Deep Learning”. In: *Proceedings of the 34th International Conference on Machine Learning*. Vol. 70. Proceedings of Machine Learning Research. International Convention Centre, Sydney, Australia: PMLR, pp. 3067–3075. URL: http://proceedings.mlr.press/v70/shalev-shwartz17a.html.

Shapourian, Hassan, Ken Shiozaki, and Shinsei Ryu (2017). “Many-Body Topological Invariants for Fermionic Symmetry-Protected Topological Phases”. In: *Phys. Rev. Lett.* 118.21, p. 216402. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.118.216402. URL: http://link.aps.org/doi/10.1103/PhysRevLett.118.216402.

Sharir, Or et al. (2020). “Deep autoregressive models for the efficient variational simulation of many-body quantum systems”. In: *Phys. Rev. Lett.* 124.2, p. 020503.

Skolik, Andrea et al. (2020). “Layerwise learning for quantum neural networks”. In: *arXiv preprint arXiv:2006.14904*. URL: https://arxiv.org/abs/2006.14904.

Slussarenko, Sergei et al. (2017). “Quantum state discrimination using the minimum average number of copies”. In: *Physical review letters* 118.3, p. 030502. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.118.030502.

Spencer, Joel (1994). *Ten lectures on the probabilistic method*. Second. Vol. 64. CBMS-NSF Regional Conference Series in Applied Mathematics. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, pp. vi+88. ISBN: 0-89871-325-0. DOI: 10.1137/1.9781611970074. URL: https://doi.org/10.1137/1.9781611970074.

Struchalin, G.I. et al. (2021). “Experimental Estimation of Quantum State Properties from Classical Shadows”. In: *PRX Quantum* 2 (1), p. 010307. DOI: 10.1103/PRXQuantum.2.010307. URL: https://link.aps.org/doi/10.1103/PRXQuantum.2.010307.

Su, W_P, JR Schrieffer, and Ao J Heeger (1979). “Solitons in polyacetylene”. In: *Phys. Rev. Lett.* 42.25, p. 1698.

Sugiyama, Takanori, Peter S. Turner, and Mio Murao (Oct. 2013). “Precision-Guaranteed Quantum Tomography”. In: *Phys. Rev. Lett.* 111 (16), p. 160406. DOI: 10.1103/PhysRevLett.111.160406. URL: https://link.aps.org/doi/10.1103/PhysRevLett.111.160406.

Sutton, Richard S and Andrew G Barto (2018). *Reinforcement learning: An introduction*. MIT press.

Suykens, Johan AK and Joos Vandewalle (1999). “Least squares support vector machine classifiers”. In: *Neural processing letters* 9.3, pp. 293–300.

Sweke, Ryan et al. (2020). “On the quantum versus classical learnability of discrete distributions”. In: *arXiv preprint arXiv:2007.14451*.

Tang, Duyu, Bing Qin, and Ting Liu (2015). “Document modeling with gated recurrent neural network for sentiment classification”. In: *Proceedings of the 2015 conference on empirical methods in natural language processing*, pp. 1422–1432.

Tang, Ewin (2018). “Quantum-inspired classical algorithms for principal component analysis and supervised clustering”. In: *arXiv preprint arXiv:1811.00414*.

Tang, Ewin (2019). "A quantum-inspired classical algorithm for recommendation systems". In: *STOC*, pp. 217–228.

– (2021). "Quantum Principal Component Analysis Only Achieves an Exponential Speedup Because of Its State Preparation Assumptions". In: *Phys. Rev. Lett.* 127.6, p. 060503.

Tasaki, Hal (2018). "Topological Phase Transition and Z2 Index for S=1 Quantum Spin Chains". In: *Phys. Rev. Lett.* 121 (14), p. 140604. DOI: 10.1103/PhysRevLett.121.140604. URL: https://link.aps.org/doi/10.1103/PhysRevLett.121.140604.

– (2020). *Physics and Mathematics of Quantum Many-Body Systems*. Graduate Texts in Physics. Cham: Springer International Publishing. ISBN: 978-3-030-41264-7. DOI: 10.1007/978-3-030-41265-4. URL: http://link.springer.com/10.1007/978-3-030-41265-4.

Tibshirani, Robert (1996). "Regression shrinkage and selection via the lasso". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1, pp. 267–288.

Torlai, Giacomo, Guglielmo Mazzola, et al. (2018). "Neural-network quantum state tomography". In: *Nat. Phys.* 14.5, p. 447.

Torlai, Giacomo and Roger G. Melko (2016). "Learning thermodynamics with Boltzmann machines". In: *Physical Review B* 94.16, p. 165134. DOI: 10.1103/PhysRevB.94.165134. URL: http://link.aps.org/doi/10.1103/PhysRevB.94.165134 (visited on 01/20/2017).

Torlai, Giacomo, Brian Timar, et al. (2019). "Integrating Neural Networks with a Quantum Simulator for State Reconstruction". In: *Phys. Rev. Lett.* 123 (23), p. 230504. DOI: 10.1103/PhysRevLett.123.230504. URL: https://link.aps.org/doi/10.1103/PhysRevLett.123.230504.

Totsuka, K and M Suzuki (1995). "Matrix formalism for the VBS-type models and hidden order". In: *J. Phys. Condens. Matter* 7.8, pp. 1639–1662. ISSN: 0953-8984. DOI: 10.1088/0953-8984/7/8/012. URL: https://iopscience.iop.org/article/10.1088/0953-8984/7/8/012.

Tran, Minh C et al. (2020). "Hierarchy of linear light cones with long-range interactions". In: *Physical Review X* 10.3, p. 031009.

Tropp, Joel A. (2012). "User-Friendly Tail Bounds for Sums of Random Matrices". In: *Found. Comput. Math* 12.4, pp. 389–434. DOI: 10.1007/s10208-011-9099-z. URL: https://doi.org/10.1007/s10208-011-9099-z.

Valiant, L.G. and V.V. Vazirani (1986). "NP is as easy as detecting unique solutions". In: *Theoretical Computer Science* 47, pp. 85–93. ISSN: 0304-3975. DOI: https://doi.org/10.1016/0304-3975(86)90135-0. URL: https://www.sciencedirect.com/science/article/pii/0304397586901350.

Valiant, Leslie G (1984). “A theory of the learnable”. In: *Commun. ACM* 27.11, pp. 1134–1142.

Van Nieuwenburg, Evert PL, Ye-Hua Liu, and Sebastian D Huber (2017). “Learning phase transitions by confusion”. In: *Nat. Phys.* 13.5, pp. 435–439.

Vapnik, Vladimir (2013). *The nature of statistical learning theory*. Springer science & business media.

Vargas-Hernández, Rodrigo A et al. (2018). “Extrapolating quantum observables with machine learning: inferring multiple phase transitions from properties of a single phase”. In: *Physical review letters* 121.25, p. 255702.

Vaswani, Ashish et al. (2017). “Attention is all you need”. In: *Advances in neural information processing systems*, pp. 5998–6008.

Vazirani, Vijay V. (2001). *Approximation algorithms*. Springer. ISBN: 978-3-540-65367-7. URL: `http://www.springer.com/computer/theoretical+computer+science/book/978-3-540-65367-7`.

Vermersch, B. et al. (2018). “Unitary n -designs via random quenches in atomic Hubbard and spin models: Application to the measurement of Rényi entropies”. In: *Phys. Rev. A* 97.2, p. 023604. DOI: `10.1103/PhysRevA.97.023604`. URL: `https://link.aps.org/doi/10.1103/PhysRevA.97.023604`.

Vershynin, Roman (2018a). *High-dimensional probability*. Vol. 47. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, Cambridge, pp. xiv+284. ISBN: 978-1-108-41519-4. DOI: `10.1017/9781108231596`. URL: `https://doi.org/10.1017/9781108231596`.

– (2018b). *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge University Press.

Wan, Kianna, William J Huggins, et al. (2022). “Matchgate Shadows for Fermionic Quantum Simulation”. In: *arXiv preprint arXiv:2207.13723*.

Wan, Kianna and Isaac Kim (2020). “Fast digital methods for adiabatic state preparation”. In: *arXiv preprint arXiv:2004.04164*.

Wang, Lei (2016). “Discovering phase transitions with unsupervised learning”. In: *Phys. Rev. B* 94 (19), p. 195105. DOI: `10.1103/PhysRevB.94.195105`. URL: `https://link.aps.org/doi/10.1103/PhysRevB.94.195105`.

Watrous, John (2018). *The Theory of Quantum Information*. Cambridge University Press. DOI: `10.1017/9781316848142`.

Webb, Zak (2016). “The Clifford group forms a unitary 3-design”. In: *Quantum Information & Computation* 16.15-16, pp. 1379–1400.

Wecker, Dave, Matthew B Hastings, and Matthias Troyer (2015). “Progress towards practical quantum variational algorithms”. In: *Physical Review A* 92.4, p. 042303. URL: `https://journals.aps.org/pra/abstract/10.1103/PhysRevA.92.042303`.

Weisz, Ferenc (2012). “Summability of multi-dimensional trigonometric Fourier series”. In: *arXiv preprint arXiv:1206.1789*.

White, Steven R (1993a). “Density-matrix algorithms for quantum renormalization groups”. In: *Phys. Rev. B* 48.14, p. 10345.

– (1992). “Density matrix formulation for quantum renormalization groups”. In: *Phys. Rev. Lett.* 69 (19), pp. 2863–2866. DOI: 10.1103/PhysRevLett.69.2863. URL: https://link.%20aps.org/doi/10.1103/PhysRevLett.69.2863.

– (1993b). “Density-matrix algorithms for quantum renormalization groups”. In: *Phys. Rev. B* 48 (14), pp. 10345–10356. DOI: 10.1103/PhysRevB.48.10345. URL: https://link.aps.org/doi/10.1103/PhysRevB.48.10345.

Wiersema, Roeland et al. (2020). “Exploring entanglement and optimization within the Hamiltonian Variational Ansatz”. In: *arXiv preprint arXiv:2008.02941*. URL: https://arxiv.org/abs/2008.02941.

Wigderson, Avi and David Xiao (2008). “Derandomizing the Ahlswede-Winter matrix-valued Chernoff bound using pessimistic estimators, and applications”. In: *Theory Comput.* 4, pp. 53–76. DOI: 10.4086/toc.2008.v004a003. URL: https://doi.org/10.4086/toc.2008.v004a003.

Wilde, Mark M. (2013). *Quantum Information Theory*. 2nd. Cambridge: Cambridge University Press. ISBN: 9781139525343. DOI: 10.1017/CBO9781139525343. URL: http://ebooks.cambridge.org/ref/id/CBO9781139525343.

Wu, Yihong (2017). “Lecture notes on information-theoretic methods for high-dimensional statistics”. In: *Lecture Notes for ECE598YW (UIUC)* 16.

Xiao, Han, Kashif Rasul, and Roland Vollgraf (2017). “Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms”. In: *arXiv preprint arXiv:1708.07747*. URL: https://arxiv.org/abs/1708.07747.

Yu, Bin (1997). “Assouad, fano, and le cam”. In: *Festschrift for Lucien Le Cam*. Springer, pp. 423–435.

Zeng, Bei et al. (2019). *Quantum information meets quantum matter*. Springer.

Zhang, Chi (2010). “An improved lower bound on query complexity for quantum PAC learning”. In: *Inf. Process. Lett.* 111.1, pp. 40–45.

Zhang, Yi, Paul Ginsparg, and Eun-Ah Kim (2020). “Interpreting machine learning of topological quantum phase transitions”. In: *Physical Review Research* 2.2, p. 023283.

Zhang, Yi and Eun-Ah Kim (2017). “Quantum loop topography for machine learning”. In: *Phys. Rev. Lett.* 118.21, p. 216401.

Zhang, Yi, Roger G Melko, and Eun-Ah Kim (2017). “Machine learning Z 2 quantum spin liquids with quasiparticle statistics”. In: *Physical Review B* 96.24, p. 245119.

Zhao, Andrew, Nicholas C Rubin, and Akimasa Miyake (2021). “Fermionic partial tomography via classical shadows”. In: *Physical Review Letters* 127.11, p. 110504.

Zhou, Zhenpeng, Xiaocheng Li, and Richard N Zare (2017). “Optimizing chemical reactions with deep reinforcement learning”. In: *ACS Cent. Sci.* 3.12, pp. 1337–1344.

Zhu, Huangjun (Dec. 2017). “Multiqubit Clifford groups are unitary 3-designs”. In: *Phys. Rev. A* 96 (6), p. 062336. DOI: `10.1103/PhysRevA.96.062336`. URL: `https://link.aps.org/doi/10.1103/PhysRevA.96.062336`.

INDEX