Neural Network Models of Learning and Generalization

Thesis by
Panteleimon Vafeidis

In Partial Fulfillment of the Requirements for the
Degree of
Doctor of Philosophy
in
Computation and Neural Systems

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2025
Defended May 8, 2025

Panteleimon Vafeidis
ORCID: 0000-0002-9768-0609

ACKNOWLEDGEMENTS

To **Thanos Siapas, Michael Dickinson, and others**, thank you for the opportunity to be part of the Caltech community.

To my advisor, **Antonio Rangel**, thank you for your trust, thoughtfulness, and for giving me academic freedom—not much more one can ask for.

To my friend and collaborator, **Aman Bhargava**, thank you for your friendship, clarity of thought, commitment to your beliefs, kindness, for the exciting intellectual journeys, and cheers to more.

To **Markus Meister**, thank you for not compromising your scientific standards, and encouraging others not to—this thesis is an example of this.

To **Pietro Perona**, thank you for your words of encouragement, at a time much needed.

To **Reza Sadri**, thank you for your mentorship, support, and for introducing me to the world of industry.

To **Richard Kempter and Tiziano D'Albis**, thank you for your patience, and for introducing me to the field of theoretical neuroscience.

To **Saif Kabariti, Arda Secme, Tom Duan,** and the rest of the crew, thank you for allowing me to explore a different side of LA, and of myself.

To all other **friends**, thank you for making my time at Caltech memorable, and for making me feel part of a community.

To **Amy Li**, thank you for making me a better person.

To my **family**, thank you for your love and support.

Finally, I would like to thank the **Onassis Foundation** for their financial support for most of my studies.

ABSTRACT

Neural networks have emerged as powerful models for understanding both biological and artificial intelligence, yet significant questions remain about how these systems develop rich, generalizable representations of the world. This thesis investigates fundamental principles of learning and generalization across four interconnected domains, bridging insights from theoretical neuroscience and artificial intelligence to advance our understanding of intelligent systems.

In Chapter I, we address a central question in associative learning: how do neural circuits learn to associate concepts with one another? We propose a recurrent neural network model incorporating two critical features of cortical architecture—mixed selectivity and compartmentalized neurons. These architectural inductive biases enable a biologically plausible learning rule that achieves stimulus substitution, where neurons respond identically to a conditioned stimulus as they would to the associated unconditioned stimulus. Our model explains a remarkable range of conditioning phenomena under conditions in which traditional associative models fail, highlighting how the cortical architecture may confer significant evolutionary advantages for flexible learning.

Chapter II pivots from the static mappings between concepts learned in Chapter I to explore how neural systems develop the precise synaptic connectivity required to establish dynamic mappings for path integration—the ability to maintain an internal sense of direction without external cues. We demonstrate that the same principles of compartmentalized learning can shape networks that accurately track angular position in darkness. Applied to the *Drosophila* head direction system, our model develops connectivity patterns strikingly similar to those observed experimentally, with continuous attractor (CAN) dynamics emerging naturally from learning. This offers a novel perspective on how precisely calibrated neural circuits can develop through experience, rather than requiring genetic pre-specification, and explains experimental findings where animals adapt their internal representation when sensory experience changes.

In Chapter III, we establish a theoretical framework explaining how disentangled representations—internal models that isolate independent factors of variation in the world—emerge from multi-task learning. We prove that any system competent at multiple related tasks must implicitly represent the underlying latent variables in a

linearly decodable form when sufficient tasks are learned. These theoretical guarantees align with experimental results showing neural networks develop generalizable representations when trained on multiple tasks simultaneously. This work reveals a fundamental connection between task diversity and representation quality, with implications for biological cognition and artificial intelligence design, particularly explaining why modern transformer models may develop human-interpretable concepts, and how brains may acquire their impressive zero-shot generalization ability.

Chapter IV proposes leveraging Large Language Models as cognitive tools for evaluating latent factor hypotheses for psychology, leveraging the theoretical insights from Chapter III. It suggests that the self-consistency of an LLM's responses given hypothesized psychological factors could serve as a metric for hypothesis evaluation. While preliminary, this approach represents a novel computational methodology for psychology that could transform how hypotheses for human cognition are developed and refined.

All chapters are supported by corresponding Appendices that go deeper in particular details, including proofs. An exemption is Chapter IV, which is work early in development (yet valuable to mention). Instead, for Appendix D we provide some considerations about the detection of Continuous Attractors (CANs), which display prominently in Chapters II and III, consideration particularly important in order to avoid confusion when it comes to these concepts, particularly within the experimental neuroscience community.

Together, these investigations reveal complementary aspects of how intelligent systems develop useful representations through learning. From biologically plausible learning rules to abstract computational principles, this thesis demonstrates how neural networks can illuminate fundamental mechanisms of intelligence across natural and artificial systems, advancing our understanding of the computational foundations that enable flexible, generalizable learning.

PUBLISHED CONTENT AND CONTRIBUTIONS

Vafidis, Pantelis, Aman Bhargava, and Antonio Rangel (2025a). "Disentangling representations through multi-task learning." In: *The Thirteenth International Conference on Learning Representations*. URL: https://openreview.net/forum?id=yVGGtsOgc7.
P.V. conceived the project, developed and implemented the neural network models, conducted all simulations, and wrote the manuscript.

– (2025b). "Multi-task learning yields disentangled world models: Impact and implications." In: *NeurIPS 2024 Workshop on Symmetry and Geometry in Neural Representations*. URL: https://openreview.net/forum?id=vqD8LEvIq3.
P.V. conceived the project, developed and implemented the neural network models, conducted all simulations, and wrote the manuscript.

– (2024a). "Multi-task learning yields disentangled world models: Impact and implications." In: *UniReps: 2nd Edition of the Workshop on Unifying Representations in Neural Models*. URL: https://openreview.net/forum?id=S0YFcYMis7.
P.V. conceived the project, developed and implemented the neural network models, conducted all simulations, and wrote the manuscript.

– (2024b). "Parallel decision-making yields disentangled world models: Impact and implications." In: *The First Workshop on NeuroAI @ NeurIPS2024*. URL: https://openreview.net/forum?id=LWPoA68TFT.
P.V. conceived the project, developed and implemented the neural network models, conducted all simulations, and wrote the manuscript.

Vafidis, Pantelis and Antonio Rangel (2024). "Stimulus-to-stimulus learning in RNNs with cortical inductive biases." arXiv: 2409.13471 [q-bio.NC]. URL: https://arxiv.org/abs/2409.13471.
P.V. conceived the project, developed and implemented the neural network models, performed all simulations, wrote the initial draft of the manuscript, and co-wrote the final version of the manuscript.

Vafidis, Pantelis et al. (June 2022). "Learning accurate path integration in ring attractor models of the head direction system." In: *eLife* 11. Ed. by S. Ostojic, R. L. Calabrese, and H. Rouault, e69841. ISSN: 2050-084X. DOI: 10.7554/eLife.69841. URL: https://doi.org/10.7554/eLife.69841.
P.V. contributed in the conception of the project, developed and implemented the neural network models, performed simulations, wrote the initial draft of the manuscript, and co-wrote the final version of the manuscript.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

NOMENCLATURE

ANN. Artificial Neural Network.

BCM. Bienenstock-Cooper-Munro.

BERT. Bidirectional Encoder Representations from Transformers.

BPTT. Backpropagation Through Time.

CAN. Continuous Attractor Network.

GPT. Generative Pre-Trained Transformer.

ID. In-distribution.

LLM. Large Language Model.

LSTM. Long Short-Term Memory.

OOD. Out-of-distribution.

RNN. Recurrent Neural Network.

"Once you get the representation right, you're often almost done, because the representation reveals the structure of the solution."

— Prof. Patrick Henry Winston (1943 – 2019)
MIT 6.034 Artificial Intelligence

INTRODUCTION

This thesis touches on several fields: theoretical neuroscience, machine learning, representation learning, (cognitive) neuroscience, and psychology. A common bridging element between all the different components is the usage of Neural Networks as elemental building blocks of intelligence, and the attempt to account for several aspects of intelligence from the low level up; in ways that often allow for precise *mechanistic interpretability*, i.e., understanding of the precise computational mechanism that leads to emergent behavior, as opposed to "word models" commonly used in the field of neuroscience, which often fail to expose competing claims, hindering progress.

Biological Inspiration

This thesis draws inspiration from several key neuroscientific observations that illuminate fundamental principles of neural computation. The first concerns the unique structure of pyramidal neurons, particularly in layer 5 of the cortex, which feature distinct compartments that integrate inputs from different sources (Larkum, Zhu, and Sakmann, 1999; Larkum, 2013; Urbanczik and Senn, 2014; Doron et al., 2020). These compartmentalized neurons implement a cellular-level predictive coding mechanism through their anatomical organization: proximal dendrites typically receive feedforward sensory inputs, while distal dendrites in layer 1 receive feedback signals carrying predictions based on internal models (Larkum, 2013). This architecture enables these neurons to compare bottom-up sensory information with top-down predictions, generating error signals when mismatches occur. The resulting dendritic plateau potentials and calcium spikes serve as coincidence detection mechanisms that drive synaptic plasticity, effectively implementing the computational principle of prediction error minimization (Rao and Ballard, 1999; Rao, 1999). Through this structural specialization, individual neurons become powerful predictive units capable of associating temporally correlated inputs arriving at different dendritic compartments, laying the foundation for cortical learning and representation formation.

The structure of neural representations across brain regions is another inspiration this thesis draws from: successful generalization requires representations that isolate underlying factors of variation (abstract, or disentangled representations (Caruana, 1997; Higgins et al., 2017; Higgins et al., 2018; Ostojic and Fusi, 2024)). In

the field of machine learning and computational neuroscience, multi-task learning approaches have demonstrated that systems trained on diverse but related tasks naturally develop such representations (Maziarka et al., 2023; Johnston and Fusi, 2023). Wet lab neuroscience research has corroborated these findings, revealing abstract, disentangled encoding of behaviorally relevant variables in prefrontal cortex (Bongioanni et al., 2021; Nogueira et al., 2023), hippocampus (Bernardi et al., 2020; Boyle et al., 2022; Courellis et al., 2024), and amygdala (Saez et al., 2015), across humans, mice and non-human primates. These representations allow neural circuits to generalize by decomposing novel stimuli into familiar components, enabling appropriate responses to previously unseen inputs—a hallmark of biological intelligence that artificial systems increasingly emulate.

Finally, mixed selectivity—the tendency of neurons to respond to diverse combinations of task variables rather than encoding single variables, is a central theme of this work. Initially, mixed selectivity appeared to be a curious property of neural coding, particularly in prefrontal regions (Rigotti et al., 2013). However, theoretical work has demonstrated that mixed selectivity is not a bug but a feature, dramatically expanding the computational capacity of neural circuits (Fusi, Miller, and Rigotti, 2016). Mixed representations enable high-dimensional encoding of complex task variables, supporting flexible computation through linear readouts that can extract task-relevant information. This reframing of mixed selectivity as a computational advantage rather than an inefficiency challenges traditional notions of neural specialization and suggests that apparent disorder in neural responses may actually reflect a sophisticated encoding strategy optimized for flexible, generalizable computation.

Gaps in the Literature

Despite these insights, the neuroscience field has neglected critical aspects of neural computation that this thesis aims to address. A significant oversight is the field's persistent focus on serial processing paradigms, where experimental designs typically isolate single tasks or functions. This approach fundamentally misaligns with how the brain operates—neural circuits process multiple streams of information simultaneously and perform numerous computations in parallel (Markram et al., 2015; Hawkins et al., 2019). The field's fixation on what might be termed the "consciousness bottleneck"—the serial nature of awareness—has led to experimental paradigms that artificially constrain neural processing to single tasks, potentially missing the rich parallel computations occurring below the threshold of conscious perception. This methodological constraint has profound consequences: it has lim-

ited our understanding of how the brain develops generalizable representations that support flexible behavior across contexts. By designing experiments that force serial task execution, we may have systematically underestimated the brain's capacity for parallel computation and obscured the very mechanisms that enable its remarkable generalization abilities.

Another critical gap exists in connecting neuronal architecture to computational capabilities. The cortex's parallel processing architecture, with its massively recurrent connectivity pattern repeated across regions, appears ideally suited for developing disentangled representations through simultaneous processing of diverse task demands (Hawkins et al., 2019; Ostojic and Fusi, 2024). Yet this connection between parallel cortical structure and the emergence of abstract representations remains theoretically underdeveloped. Similarly, mixed selectivity—while recognized as expanding computational capacity—has not been explicitly linked to flexible predictive learning capabilities. When neurons maintain mixed tuning to multiple variables, they create a high-dimensional representation space where complex, nonlinear functions become linearly separable (Rigotti et al., 2013; Fusi, Miller, and Rigotti, 2016). This property should theoretically allow neural circuits to learn increasingly sophisticated predictive models of the world, yet the mechanisms by which mixed selectivity enables predictive learning have remained elusive. The synergistic relationship between cortical architecture, mixed representations, and learning capabilities represents a fundamental gap in our understanding of intelligence. This thesis addresses these interconnected aspects of neural computation, proposing that the cortex's structural organization specifically enables both efficient disentangled representations and high-capacity predictive learning through its unique combination of parallel processing and mixed selectivity.

Contributions

This thesis makes fundamental advancements in understanding two key aspects of intelligence: learning and generalization. Through computational modeling and theoretical analysis, I demonstrate how neural architectural principles enable sophisticated learning capabilities and robust generalization across diverse contexts. The first major contribution establishes that compartmentalized neurons with mixed selectivity provide an ideal substrate for stimulus-stimulus associative learning. In Chapter I, we show how this architecture enables cortical circuits to efficiently pack multiple associations within the same neural population—a significant evolutionary advantage compared to dedicated circuit approaches. The model accounts for a

broad range of classical conditioning phenomena using biologically plausible learning rules, demonstrating how the two-compartment structure of layer-5 pyramidal neurons enables predictive learning without requiring specialized teaching signals (Vafidis and Rangel, 2024). This work suggests that the cortical architecture itself may represent an evolved specialization for flexible associative learning, allowing mammals to rapidly acquire new stimulus associations beyond what genetically pre-specified circuits could support.

Building on these insights, this thesis demonstrates that the same cellular and learning principles can support the development of sophisticated continuous attractor dynamics necessary for path integration. Chapter II shows how a biologically plausible learning rule, operating within compartmentalized neurons, shapes a head direction network capable of accurately integrating angular velocity signals in the absence of external cues (Vafidis et al., 2022). The resulting model produces connectivity patterns strikingly similar to those observed in the *Drosophila* head direction system, suggesting that precisely calibrated neural circuits for complex functions need not be genetically pre-specified but can instead emerge through self-supervised learning during development. Importantly, the continuous attractor dynamics that support path integration are found to emerge naturally from learning rather than requiring fine-tuned connectivity. This finding generalizes beyond navigation, suggesting that continuous attractors may serve as a general computational substrate for maintaining and updating internal representations across neural systems.

A central theoretical contribution of this thesis is the establishment of formal guarantees for the emergence of disentangled representations in neural systems that perform multiple tasks in parallel. Chapter III proves that any system competent at multiple related tasks must implicitly represent the underlying latent variables in its hidden state, with disentanglement increasing as the number of tasks grows (Vafidis, Bhargava, and Rangel, 2025). This theoretical result provides a fundamental explanation for why parallel processing—a hallmark of cortical computation—leads to generalizable representations. Through extensive computational experiments, I confirm that recurrent neural networks trained on multiple tasks develop disentangled representations in the form of continuous attractors, leading to zero-shot generalization to previously unseen regions of input space. These findings establish a deep connection between architectural organization (parallel processing) and representational properties (disentanglement), providing an explanatory framework for empirical observations of abstract representations across brain regions.

Finally, this thesis demonstrates the power of leveraging computational models as tools for cognitive science. Chapter IV proposes a novel framework for evaluating psychological latent factor hypotheses using Large Language Models, extending the theoretical insights from Chapter III to the domain of psychological theory development. By quantifying the self-consistency of model responses to different psychological constructs, this approach offers a computational methodology for assessing theoretical coherence in psychology. While more preliminary than the other contributions, this work illustrates how computational principles discovered in the context of neural representation learning can inform and enhance human-centered disciplines like psychology, pointing toward new paradigms for theory evaluation and refinement that leverage sophisticated computational models.

Chapters III and IV share a deep, not directly obvious methodological connection. Both chapters leverage the model's *confidence* as a primary signal rather than treating it as a mere byproduct. In Chapter III, uncertainty in multi-task classification helps uncover the quality and structure of learned representations, while Chapter IV extends this approach by using Large Language Model confidence to evaluate psychological hypotheses. This focus on extracting meaningful information from model uncertainty represents a powerful approach that can yield insights across diverse domains, with applications to model distillation and the "copying of world models" via the logits—effectively transferring the model's understanding of the world rather than just its output behavior.

Viewpoint of this Thesis

The diverse topics touched upon here reflect the diversity of the interests of the author, which ideally should be the substrate for cross-disciplinary efforts in the brain sciences. The field is in desperate need of researchers that can think beyond the specific sub-discipline they have been trained for (cognitive science, neuroscience, neural computation, artificial intelligence) which, again, is necessary for real progress. Brain science is not an extension of biology or plant science; hence it should not be pursued as such.

Central to what makes the brain interesting is the question of intelligence—how do (low-level) mechanisms in the brain lead to flexible, intelligent cognition. So while a lot of the models used in this thesis draw direct and close inspiration from biology, our primary focus is on understanding the computational principles that enable intelligence rather than replicating biological details that may or may not be important

from a computational standpoint. The question of intelligence is, unfortunately, largely ignored—or reduced to a byproduct—by contemporary neuroscience. Two exceptions to this rule, namely the investigation of mechanisms for predictive coding and the representational structure that allows flexible generalization, have sparked the interest of the author and hence appear prominent in this thesis.

Answering the question of intelligence is something definitely within the reach of neuroscience with its modern (and expensive) tools, and something that might make the field relevant beyond internal consumption. Brain science should (and can) be the catalyst for truly intelligent artificial intelligence. With its current intense focus on methods for the sake of methods, neuroscience may have lost sight of this big picture. Hence, as far as the author is concerned at least, when we are talking about neuroscience, what we really mean is brain science. With machine learning merging with psychology with the advent of Large Language Models, we are way beyond that compartmentalized approach to cognition, and the term neuroscience is archaic and out of focus (although there are several examples of intelligence down to the level of neurons, some examples in this thesis).

Similarly, terms such as "NeuroAI" assume a dichotomy that does not exist, and will not be utilized in this thesis. As reflected in the name of Caltech's "Computation and Neural Systems" program which started all the way back in 1986, the correct word is "computation"; insights on the fundamental nature and principles of computation when pairing many simple elements together can lead to better understanding of both biological and artificial systems. The author believes that being able to comfortably go back and forth between the two will pay dividends in the future. We will be seeing direct evidence of that, as the work on Disentangled Representations in Chapter III was directly inspired from the work in the *Drosophila* head direction system in Chapter II; the *Drosophila* has an explicit representation of head direction exactly because it is useful for many downstream processes.

Brain science may represent a new kind of "soft hard science," where the traditional statistical rigidity of Western scientific methods sometimes must yield to intuition and assessments of coherence to make progress. The complex, self-referential nature of studying a high-capacity, adaptable system creates unique challenges that our current epistemological frameworks struggle to accommodate. This does not mean abandoning rigor, but rather recognizing that understanding intelligence may require innovative methodological approaches that blend quantitative precision with qualitative insight in ways that traditional disciplinary boundaries often resist. This

"good taste" has been the hallmark of valuable philosophical insight, beyond the stricter confines of "natural philosophy" science stems from; a good taste increasingly missing from academic discourse often favoring formulaism and process over substance (you have to be able to write a grant to "defend" anything nowadays).

To conclude, I would like to mention two catalysts that allowed this work to materialize. One is the unrivaled scientific merit, freedom to think, and opportunity to be heard that is present in Caltech; its importance cannot be understated in fostering scientists that have the confidence and self-belief to make impact. The second lesson I learned here was the value of collaboration. I am now convinced that collaboration, not isolation, is the right way to do science, where every one brings out the best from another, and the sum is always greater than its parts.

References

Bernardi, Silvia et al. (Nov. 2020). "The geometry of abstraction in the hippocampus and prefrontal cortex." In: *Cell* 183.4, 954–967.e21. DOI: 10.1016/j.cell.2020.09.031. URL: https://doi.org/10.1016/j.cell.2020.09.031.

Bongioanni, Alessandro et al. (Jan. 2021). "Activation and disruption of a neural mechanism for novel choice in monkeys." In: *Nature* 591.7849, pp. 270–274. DOI: 10.1038/s41586-020-03115-5. URL: https://doi.org/10.1038/s41586-020-03115-5.

Boyle, Lara M. et al. (Jan. 2022). "Tuned geometries of hippocampal representations meet the demands of social memory." DOI: 10.1101/2022.01.24.477361. URL: https://doi.org/10.1101/2022.01.24.477361.

Caruana, Rich (1997). "Multitask learning." In: *Machine Learning* 28.1, pp. 41–75. ISSN: 0885-6125. DOI: 10.1023/a:1007379606734. URL: http://dx.doi.org/10.1023/A:1007379606734.

Courellis, Hristos S. et al. (Aug. 2024). "Abstract representations emerge in human hippocampal neurons during inference." In: *Nature* 632.8026, pp. 841–849. ISSN: 1476-4687. DOI: 10.1038/s41586-024-07799-x. URL: http://dx.doi.org/10.1038/s41586-024-07799-x.

Doron, Guy et al. (2020). "Perirhinal input to neocortical layer 1 controls learning." In: *Science* 370.6523, eaaz3136. ISSN: 0036-8075. DOI: 10.1126/science.aaz3136. eprint: https://science.sciencemag.org/content/370/6523/eaaz3136.full.pdf. URL: https://science.sciencemag.org/content/370/6523/eaaz3136.

Fusi, Stefano, Earl K. Miller, and Mattia Rigotti (Apr. 2016). "Why neurons mix: high dimensionality for higher cognition." In: *Current Opinion in Neurobiology* 37, pp. 66–74. ISSN: 0959-4388. DOI: 10.1016/j.conb.2016.01.010. URL: http://dx.doi.org/10.1016/j.conb.2016.01.010.

Hawkins, Jeff et al. (Jan. 2019). "A framework for intelligence and cortical function based on grid cells in the neocortex." In: *Frontiers in Neural Circuits* 12. ISSN: 1662-5110. DOI: 10.3389/fncir.2018.00121. URL: http://dx.doi.org/10.3389/fncir.2018.00121.

Higgins, Irina et al. (2017). "β-VAE: Learning basic visual concepts with a constrained variational framework." In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=Sy2fzU9gl.

Higgins, Irina et al. (2018). "Towards a definition of disentangled representations." DOI: 10.48550/ARXIV.1812.02230. URL: https://arxiv.org/abs/1812.02230.

Johnston, W. Jeffrey and Stefano Fusi (Feb. 2023). "Abstract representations emerge naturally in neural networks trained to perform multiple tasks." In: *Nature Communications* 14.1. DOI: 10.1038/s41467-023-36583-0. URL: https://doi.org/10.1038/s41467-023-36583-0.

Larkum, Matthew E. (Mar. 2013). "A cellular mechanism for cortical associations: An organizing principle for the cerebral cortex." In: *Trends in Neurosciences* 36.3, pp. 141–151. DOI: 10.1016/j.tins.2012.11.006. URL: https://doi.org/10.1016/j.tins.2012.11.006.

Larkum, Matthew E., J. Julius Zhu, and Bert Sakmann (Mar. 1999). "A new cellular mechanism for coupling inputs arriving at different cortical layers." In: *Nature* 398.6725, pp. 338–341. DOI: 10.1038/18686. URL: https://doi.org/10.1038/18686.

Markram, Henry et al. (Oct. 2015). "Reconstruction and simulation of neocortical microcircuitry." In: *Cell* 163.2, pp. 456–492. ISSN: 0092-8674. DOI: 10.1016/j.cell.2015.09.029. URL: http://dx.doi.org/10.1016/j.cell.2015.09.029.

Maziarka, Łukasz et al. (2023). "On the relationship between disentanglement and multi-task learning." In: *Machine Learning and Knowledge Discovery in Databases*. Springer International Publishing, pp. 625–641. ISBN: 9783031263873. DOI: 10.1007/978-3-031-26387-3_38. URL: http://dx.doi.org/10.1007/978-3-031-26387-3_38.

Nogueira, Ramon et al. (Jan. 2023). "The geometry of cortical representations of touch in rodents." In: *Nature Neuroscience* 26.2, pp. 239–250. DOI: 10.1038/s41593-022-01237-9. URL: https://doi.org/10.1038/s41593-022-01237-9.

Ostojic, Srjan and Stefano Fusi (July 2024). "Computational role of structure in neural activity and connectivity." In: *Trends in Cognitive Sciences* 28.7, pp. 677–690. ISSN: 1364-6613. DOI: 10.1016/j.tics.2024.03.003. URL: http://dx.doi.org/10.1016/j.tics.2024.03.003.

Rao, Rajesh P. N. (June 1999). "An optimal estimation approach to visual perception and learning." In: *Vision Research* 39.11, pp. 1963–1989. ISSN: 0042-6989. DOI: 10.1016/s0042-6989(98)00279-x. URL: http://dx.doi.org/10.1016/S0042-6989(98)00279-X.

Rao, Rajesh P. N. and Dana H. Ballard (Jan. 1999). "Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects." In: *Nature Neuroscience* 2.1, pp. 79–87. DOI: 10.1038/4580. URL: https://doi.org/10.1038/4580.

Rigotti, Mattia et al. (May 2013). "The importance of mixed selectivity in complex cognitive tasks." In: *Nature* 497.7451, pp. 585–590. DOI: 10.1038/nature12160. URL: https://doi.org/10.1038/nature12160.

Saez, Alex et al. (Aug. 2015). "Abstract context representations in primate amygdala and prefrontal cortex." In: *Neuron* 87.4, pp. 869–881. DOI: 10.1016/j.neuron.2015.07.024. URL: https://doi.org/10.1016/j.neuron.2015.07.024.

Urbanczik, Robert and Walter Senn (Feb. 2014). "Learning by the dendritic prediction of somatic spiking." In: *Neuron* 81.3, pp. 521–528. DOI: 10.1016/j.neuron.2013.11.030. URL: https://doi.org/10.1016/j.neuron.2013.11.030.

Vafidis, Pantelis, Aman Bhargava, and Antonio Rangel (2025). "Disentangling representations through multi-task learning." In: *The Thirteenth International Conference on Learning Representations*. URL: https://openreview.net/forum?id=yVGGtsOgc7.

Vafidis, Pantelis and Antonio Rangel (2024). "Stimulus-to-stimulus learning in RNNs with cortical inductive biases." arXiv: 2409.13471 [q-bio.NC]. URL: https://arxiv.org/abs/2409.13471.

Vafidis, Pantelis et al. (June 2022). "Learning accurate path integration in ring attractor models of the head direction system." In: *eLife* 11. Ed. by S. Ostojic, R. L. Calabrese, and H. Rouault, e69841. ISSN: 2050-084X. DOI: 10.7554/eLife.69841. URL: https://doi.org/10.7554/eLife.69841.

Chapter 1

STIMULUS-TO-STIMULUS LEARNING IN RECURRENT NEURAL NETWORKS WITH CORTICAL INDUCTIVE BIASES

Vafidis, Pantelis and Antonio Rangel (2024). "Stimulus-to-stimulus learning in RNNs with cortical inductive biases." arXiv: 2409.13471 [q-bio.NC]. URL: https://arxiv.org/abs/2409.13471.

The brain's remarkable ability to learn from experience is a fundamental aspect of intelligence. This chapter explores theoretical models that help explain how the cerebral cortex might learn to associate stimuli with each other, in a heteroassociative manner, with deep connections to the rich history of classical conditioning research. We show how specific architectural features (inductive biases) of cortical neurons, namely mixed selectivity in the cortex and the compartmentalization of layer-5 pyramidal neurons, provide the biological foundation for efficient learning mechanisms. Such mechanisms offer insight into why mammals with developed cortices demonstrate superior associative abilities, which might have conferred them important evolutionary advantanges. A main distinction of our work to previous theoretical work on classicial conditioning is that we associate rich stimulus population vectors to each other, instead of mapping a stimulus to an outcome (e.g., a reward), allowing for more direct, general and powerful association of concepts, in a process coined stimulus substitution. We also show that traditional synaptic plasticity models (e.g., Hebbian, BCM) fail to learn multiple associations in the mixed selectivity regime in which the cortex operates, emphasizing the importance of predictive learning rules, like the one utilized in our model.

Chapter Abstract

Animals learn to predict external contingencies from experience through a conditioning process. A natural mechanism for conditioning is stimulus substitution, in which the neuronal response to the *CS* becomes increasingly identical to that of the *US*. We propose a recurrent neural network model of stimulus substitution which leverages two forms of inductive bias pervasive in the cortex: representational inductive bias in the form of mixed stimulus representations, and architectural inductive bias in the form of two-compartment pyramidal neurons that have been shown to serve as a fundamental unit of cortical associative learning. The properties of these neurons allow for a biologically plausible learning rule that implements stimulus substitution, utilizing only information available locally at the synapses. We show that the model generates a wide array of conditioning phenomena, and can learn large numbers of associations with an amount of training commensurate with animal experiments, without relying on parameter fine-tuning for each individual experimental task. In contrast, we show that commonly used three-factor Hebbian rules fail to learn generic stimulus-stimulus associations with mixed selectivity and require task-specific parameter fine-tuning. In comparison to previous work, we directly predict the *identity* of the *US*, and not just a proxy for it (e.g., the reward associated with the *US*). Overall, our framework highlights the importance of multi-compartment neuronal processing in the cortex, and showcases how it might confer cortical animals the evolutionary edge.

1.1 Introduction

The ability to forecast important events is necessary for effective behavior. Animals are equipped with innate reflexes to tackle common threats and to exploit opportunities in their environment. However, given the complex and changing nature of the world, animals also need to acquire new reflexes by learning from experience. This process involves the association or conditioning of an initially neutral stimulus (conditioned stimulus, *CS*) with another stimulus intrinsically related to primary reward or punishment (unconditioned stimulus, *US*). If learning is successful, the *CS* can then induce the same behavioral response as the *US*. Initially proposed by Pavlov, this type of learning is known as classical conditioning.

A potential mechanism for conditioning is stimulus substitution (Jenkins and Moore, 1973). Under this mechanism, the response of the relevant population of neurons to the *CS* becomes increasingly identical to that generated by the *US*. After this, any downstream processes that are normally triggered by the *US* are also triggered by the *CS*. Behavioral evidence in favor of stimulus substitution comes from studies showing that animals display the same behavior to the *CS* as to the *US*, even when the behavior is not appropriate (e.g., consummatory response towards a light that has been associated with food), and that the behavior is reinforcer dependent (Jenkins and Moore, 1973). Furthermore, recent experiments show that during conditioning the response of S1 pyramidal neurons to the *CS* becomes increasingly similar to their response to the *US*, a phenomenon the authors termed "learning induced neuronal identity switch," and that this change correlates with learning performance (Dai and Sun, 2023).

A basic goal in computational and cognitive neuroscience is to build plausible models of neural network architectures capable of accounting for psychological phenomena. Previous work has shown that three-factor Hebbian synaptic plasticity rules accounts for a wide gamut of conditioning phenomena (Sutton and Barto, 1987; Klopf, 1988; Balkenius and Morén, 1998; Izhikevich, 2007). However, these models have some important limitations. First, they fail to capture the generality of pattern-to-pattern associations implicit in stimulus substitution, where both the *US* and the *CS* correspond to population activity patterns. Some use learning rules requiring storage of recent events at each synapse (Balkenius and Morén, 1998), while most assume that the tuning of neurons to stimuli is demixed, allowing simple reward modulated spike-timing-dependent plasticity to establish the appropriate mappings (Balkenius and Morén, 1998; Izhikevich, 2007). These assumptions are

inconsistent with the well-established fact that representations throughout the brain are high-dimensional and mixed (Rigotti et al., 2013).

In this study we propose a recurrent neural network (*RNN*) model of stimulus substitution. Critically, the model learns pattern-to-pattern associations using only biologically plausible local plasticity, and individual neurons are tuned to multiple behavioral stimuli, which gives rise to mixed representations of the *CS*s and *US*s. While subcortical (Christian and Thompson, 2003) and even single-neuron (Gershman et al., 2021) mechanisms for conditioning exist, our model is focused on stimulus-stimulus learning in the cortex, where the use of mixed stimulus representation allows learning a wide and flexible range of associations within the same neuronal network, which confers an evolutionary edge.

To achieve this goal, we leverage two forms of biological inductive bias built into the cortex: first, representational inductive bias in the form of mixed stimulus representations, that permit the efficient packing of multiple associations within the same neuronal population. To combat the additional complexity introduced by mixed representations, which requires not just the activation of the correct neurons but also the correct activity level, we leverage the second form of inductive bias: architectural inductive bias in the form of two-compartment layer-5 pyramidal neurons which are prevalent in the cortex (Nieuwenhuys, 1994).

We propose a *RNN* model of such two-compartment neurons. Recent work has shown that these neurons can learn to be predictive of a reward (Doron et al., 2020), and suggests that they could serve as a fundamental unit of associative learning in the cortex through a built-in cellular mechanism (Larkum, 2013). Hence, we refer to them as associative neurons. The term associative here does not have a strictly Hebbian interpretation; rather it refers to the *hetero*-associative capacity of these neurons to link together information originating from different streams (Shin, Doron, and Larkum, 2021), through a mechanism known as BAC firing (Larkum, Zhu, and Sakmann, 1999). The properties of these neurons allow for a biologically plausible learning rule that utilizes only information available locally at the synapses, and that is capable of inducing self-supervised predictive plasticity (Urbanczik and Senn, 2014; Urbanczik and Senn, 2009), which allows neurons to respond with the same firing rate to the *CS* as they would to the *US*, i.e., achieve stimulus substitution. Hence, a key distinction of our work with previous work (e.g., Izhikevich (2007)) is that we predict the *identity* (i.e., induced population vector of firing activity) of the *US*, and not just a proxy for it (e.g., the reward associated with the *US*).

We show that the model generates a wide array of conditioning phenomena, including delay conditioning, trace conditioning, extinction, blocking, overshadowing, saliency effects, overexpectation, contingency effects and faster reacquisition of previous learnt associations. Furthermore, it can learn large numbers of *CS-US* associations with an amount of training commensurate with animal experiments, without relying on parameter fine-tuning for each individual experimental task. In contrast, we show that Hebbian learning rules, including three-factor extensions of Oja's rule (Oja, 1982) and the BCM rule (Bienenstock, Cooper, and Munro, 1982), fail to learn generic stimulus-to-stimulus associations due to their unsupervised nature, and require task specific parameter fine-tuning.

1.2 Methods

Model setup

In classical conditioning animals learn to predict the upcoming appearance of an unconditioned stimulus (*US*, e.g., food) after the presentation of a conditioned stimulus (*CS*, e.g., bell ring). As shown in fig. 1.1A, trials start with the presentation of the *CS*, which lasts until $t_{\text{cs-off}}$. The *US* is presented at $t_{\text{us-on}}$, and lasts until the end of the trial. Each trial has a fixed duration of t_{trial} seconds. If the *US* appears before the *CS* disappears, the task involves delay conditioning. In contrast, if the *CS* disappears before the *US* is shown, the task involves trace conditioning, with $t_{\text{delay}} = t_{\text{us-on}} - t_{\text{cs-off}}$ denoting the delay between the two stimuli. In our task animals need to learn N_{stim} different *CS-US* pairs. Every trial one pair is randomly chosen, and the corresponding *CS* is shown followed by its associated *US*.

We model a *RNN* of associative neurons (fig. 1.1C, yellow background) that represents the stimuli using mixed population representations and is capable of learning all of the *CS-US* associations using only local information available at the synapses. The inputs to the model are time-dependent vectors $r_{\text{cs}}(t)$ and $r_{\text{us}}(t)$, of dimension N_{inp}, that encode the presence and identity of the *CS* and the *US*. For simplicity, these vectors are represented by unique Boolean vectors, and they take the value of the stimulus while it is shown, and zero otherwise. The vectors are randomly generated, subject to a constraint for a minimal Hamming distance H_{d} between any two vectors of the same type. This minimal separation limits the extent to which learning on any give pair impairs learning of the other associations. The output of the associative network is an estimate of the *US* vector r_{us}, denoted $\hat{r}_{\text{us}}$, which is decoded from network activity at all times (see fig. 1.1C and "*US* decoding" in Methods).

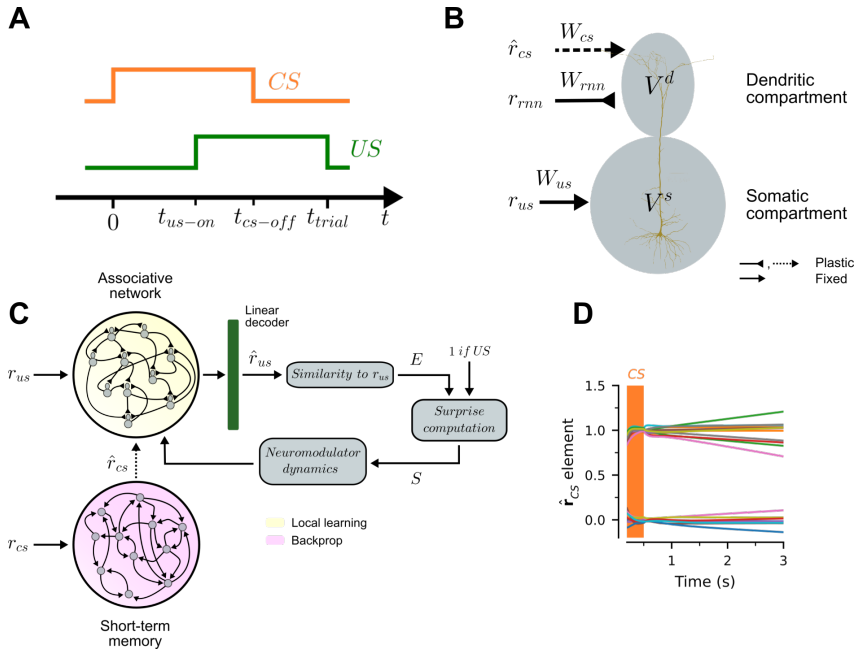

Figure 1.1: **Stimulus Substitution Model.** (A) Every trial has a duration of t_{trial} seconds. Trials start with the presentation of a *CS*, which disappears after time $t_{\text{cs-off}}$. The associated *US* appears at time $t_{\text{us-on}}$ and stays until the end of the trial. The network has to learn N_{stim} unique *CS*-*US* pairs. (B) Associative neurons are modeled as an abstraction of a layer-5 cortical pyramidal neuron. V^{s} and V^{d} denote the voltage in the somatic and dendritic compartments. The somatic compartment receives as input a Boolean vector r_{us} representing the *US*. The dendritic compartment receives as inputs a vector $\hat{r}_{\text{cs}}$ with a short-term memory representation of the *CS*, as well as recursive activity from all other neurons in the *RNN*. The matrices W_{rnn}, W_{cs} and W_{us} denote the synaptic weights for the inputs. W_{us} is fixed throughout the experiment. W_{rnn} and W_{cs} are updated over trials with training. (C) Full outline of the model. The associative network is made of N_{rnn} associative neurons. The *US* is presented directly to the associative neurons, whereas the *CS* is presented to a short-term memory circuit that produces the short-term memory representation $\hat{r}_{\text{cs}}$. Learning in the associated network is gated by a surprise signal which measures the extent to which the *US*, or its absence, was anticipated. The surprise signal is computed in three steps. First, throughout the trial a linear decoder is used to obtain an estimate $\hat{r}_{\text{us}}$ of the *US* from the population vector of the associative network, denoted by r_{rnn}. Second, an expectation E^i is formed according for each *US* based on the similarity between r^i_{us} and $\hat{r}_{\text{us}}$. These expectations determine the level of surprise S associated with the arrival or absence of the *US*, which then gives rise to neuromodulator dynamics that gate learning in the associative network. (D) Performance of the short-term memory network in a single trial when *CS*s are presented only for 500 ms. We plot the output of the memory network for several seconds. Each color denotes a different element in r_{cs}.

***RNN* of associative neurons**

The fundamental unit of computation in the associative network is the associative neuron, a two-compartment neuron modelled after layer-5 pyramidal cells in the cortex (fig. 1.1B). A crucial property of the associative neuron is that it can separate incoming "feedforward" inputs from "feedback" ones, and compare the two to drive learning. In our case, since we are modelling a primary reinforcer cortical area, *US* inputs are assumed feedforward and arrive at the somatic compartment (corresponding to the soma and proximal dendrites) through synaptic connections W_{us}, and *CS* inputs are considered feedback connections arriving to the distal dendrites from the rest of the cortex, along with local recurrent connections (W_{cs} and W_{rnn}, respectively, fig. 1.1B). This separation of inputs ultimately allows for the construction of a biologically plausible predictive learning rule, capable of achieving stimulus substitution.

The central element of the model is a *RNN* of N_{rnn} associative neurons. The goal of the network is to learn to predict the identity of the upcoming *US* from the presentation of the corresponding *CS*, by reproducing the *US* population vector when only the *CS* is presented. Each associative neuron is a two-compartment rate neuron modelled after layer-5 pyramidal cortical neurons (Larkum, Zhu, and Sakmann, 1999; Urbanczik and Senn, 2014). The somatic compartment models the activity of the soma and apical dendrites of the neuron, while the dendritic compartment models the activity of distal dendrites in cortical layer-1. As depicted in fig. 1.1B, the somatic compartment receives $r_{\text{us}}(t)$ as input, whereas the dendritic compartment receives $\hat{r}_{\text{cs}}(t)$ as well feedback activity from the all the *RNN* units, which is denoted by $r_{\text{rnn}}(t)$.

The instantaneous firing rate of the associative neurons is a sigmoidal function of the somatic voltage V^{s}:

$$r_{\text{rnn}} = \frac{f_{\max}}{1 + \exp\left[-\beta(V^{\text{s}} - V_{1/2})\right]}. \tag{1.1}$$

This activation function is applied element-wise to the vector V^{s}, which represents the instantaneous somatic voltage in each associative neuron. $f_{\max}$ sets the maximum firing rate of the neuron, β is the slope of the activation function, and $V_{1/2}$ is the voltage level at which half of the maximum firing rate is attained. We set $f_{\max}$ to a reasonable value for cortical neurons, and choose appropriate values for β and $V_{1/2}$ so that the whole dynamic range of the activation function is used and firing rates

when somatic input is present are relatively uniform. See table 1.1 for a description of all model parameters.

The somatic voltages, and thus the firing rates, are determined by the following system of differential equations:

- The associative neurons receive an input current to their dendritic compartments, denoted by I^{d}, which obey:

$$\tau_{\mathrm{s}} \frac{\mathrm{d}I^{\mathrm{d}}}{\mathrm{d}t} = -I^{\mathrm{d}} + W_{\mathrm{cs}}\, \hat{r}_{\mathrm{cs}} + W_{\mathrm{rnn}}\, r_{\mathrm{rnn}} \tag{1.2}$$

  where W_{rnn} is the matrix of synaptic weights between any pair of associative neurons (dimension: $N_{\mathrm{rnn}} \times N_{\mathrm{rnn}}$), W_{cs} is the matrix of synaptic weights for the *CS* input (dimension: $N_{\mathrm{rnn}} \times N_{\mathrm{inp}}$), and τ_{s} is the synaptic time constant.

- The dynamics of the voltage in the dendritic compartments V^{d} are given by:

$$\tau_{\mathrm{l}} \frac{\mathrm{d}V^{\mathrm{d}}}{\mathrm{d}t} = -V^{\mathrm{d}} + I^{\mathrm{d}}; \tag{1.3}$$

  i.e., it is a low-pass filtered version of the dendritic current I^{d} with the leak time constant τ_{l}. For simplicity, voltages and currents are dimensionless in our model. Therefore the leak resistance of the dendritic compartment is also dimensionless and set to unity.

- The voltages of the somatic compartments, denoted by V^{s}, are given by:

$$C \frac{\mathrm{d}V^{\mathrm{s}}}{\mathrm{d}t} = -g_L V^{\mathrm{s}} - g_D (V^{\mathrm{s}} - V^{\mathrm{d}}) + I^{\mathrm{s}} \tag{1.4}$$

  where C is the somatic membrane capacitance, g_L is the leak conductance, g_D is the conductance of the coupling from the dendritic to the somatic compartment, and I^{s} is a vector of input currents to the somatic compartments. Note that this specification assumes that the time constant for the somatic voltage is one, or equivalently, that it is included in C.

- The vector I^{s} of input currents to the somatic compartment is given by:

$$I^{\mathrm{s}} = g_{\mathrm{e}} \odot (E_{\mathrm{e}} - V^{\mathrm{s}}) + g_{\mathrm{i}} \odot (E_{\mathrm{i}} - V^{\mathrm{s}}) \tag{1.5}$$

  where g_{e} and g_{i} are vectors describing the time-varying excitatory and inhibitory conductances of the inputs, E_{e} and E_{i} are the reversal potentials for excitatory and inhibitory inputs, and $\odot$ denotes the Hadamard (element-wise) product.

- The vectors of excitatory and inhibitory conductances g_e and g_i for the somatic compartment are described, respectively, by the following two equations:

$$\tau_s \frac{dg_e}{dt} = -g_e + [W_{us}]_+ r_{us} \tag{1.6}$$

and

$$\tau_s \frac{dg_i}{dt} = -g_i + [-W_{us}]_+ r_{us} + g_{inh} \tag{1.7}$$

where W_{us} is a matrix describing the synaptic weights for the *US* inputs to the somatic compartments (dimension: $N_{rnn} \times N_{inp}$), τ_s is the same synaptic time constant used in equation 1.2, g_{inh} is a constant inhibitory conductance of all associative neurons, and $[.]_+$ is the rectification function applied element-wise.

The model implicitly assumes zero resting potentials for the somatic and dendritic compartments. In addition, we assume that there is no input to the *RNN* between trials, and that the inter-trial interval is sufficiently long so that the variables controlling activity in the associative neurons reset to zero between trials. The differential equations describing activity within trials are simulated using the forward Euler method with time setp $\Delta t = 1$ ms.

At the beginning of the experiment, all synaptic weight matrices are randomly initialised, independently for each entry, using a normal distribution with mean 0 and standard deviation $1/\sqrt{N_{rnn}}$, as is standard in the literature. Note that since associative neurons are pyramidal cells, the elements of W_{rnn} are restricted to positive values; hence we use the absolute value of those random weights.

W_{us} stays fixed for the entire experiment. W_{rnn} and W_{cs} are plastic and updated using the learning rules described next.

Synaptic plasticity rule

Specifically, to account for the ability of the associative neuron to predict its own spiking activity to somatic inputs from dendritic inputs alone (Larkum, Zhu, and Sakmann, 1999), we utilize a synaptic plasticity rule that implements local error correction at the neuronal level (Urbanczik and Senn, 2014). The learning rule modifies the connections to the dendritic compartment (i.e., W_{cs} and W_{rnn}) in order to minimize the discrepancy between the firing rate of the neuron $f(V^s)$ (where V^s is the somatic voltage, primarily controlled by *US* inputs in the beginning of learning, and f the activation function) and the prediction of the firing rate by the dendritic

Table 1.1: Model parameter values

Parameter	Value	Description
N_{stim}	16	Number of *CS*-*US*s pairs to be learnt
t_{trial}	2 s	Trial duration
$t_{\text{cs-off}}$	2 s	Time in the trial at which *CS* disappears
$t_{\text{us-on}}$	1 s	Time in the trial at which *US* appears
N_{inp}	20	Stimuli input vector length
H_{d}	8	Minimal Hamming distance between stimulus vectors
N_{rnn}	64	Number of associative neurons
$f_{\max}$	100 Hz	Maximum firing rate
β	2	Steepness of activation function
$V_{1/2}$	1.5	Input level for 50% of the maximum firing rate
τ_{s}	100 ms	Synaptic time constant
τ_{l}	20 ms	Leak time constant of dendritic compartment
C	2 ms	Capacitance of somatic compartment
g_L	0.1	Leak conductance of somatic compartment
g_D	0.2	Conductance from dendritic to somatic compartment
g_{inh}	$3/8$	Constant inhibitory conductance
E_{e}	$14/3$	Excitatory synaptic reversal potential
E_{i}	$-1/3$	Inhibitory synaptic reversal potential
a	0.95	Self-consistency adjustment constant of learning rule
τ_{r}	200 ms	Dopamine release time constant
τ_{u}	300 ms	Dopamine uptake time constant
η_0	5×10^{-3}	Baseline learning rate
Δt	1 ms	Euler integration step size

Model parameter values. These values apply to all simulations, unless otherwise stated. Note that voltages, currents, and conductances are assumed unitless in the text; therefore capacitances have the same units as time constants.

compartment $f(p'V^{\text{d}})$ (where V^{d} is the dendritic voltage, primarily controlled by *CS* inputs, and p' is a constant accounting for attenuation of V^{d} due to imperfect coupling with the somatic compartment). The synaptic weight $W_{\text{pre,post}}$ from a presynaptic neuron to a postsynaptic associative neuron is modified according to:

$$\Delta W_{\text{pre,post}} = \eta(S) \left[f(V^{\text{s}}_{\text{post}}) - f(p' V^{\text{d}}_{\text{post}}) \right] P_{\text{pre}} \tag{1.8}$$

where η is a variable learning rate which depends on a surprise signal S and P_{pre} the postsynaptic potential from the presynaptic neuron (for details, see "Synaptic plasticity rule" in Methods). In the Supplementary Information (section "Predictive coding and normative justification for the learning rule") we show how this learning rule can be derived directly from the objective of stimulus substitution.

We utilize a synaptic plasticity rule inspired by Larkum, Zhu, and Sakmann, 1999; Larkum, 2013; Doron et al., 2020, where the firing rate of the somatic compartment in the presence of the *US* acts like a target signal for learning the weights W_{rnn} and W_{cs} (see Urbanczik and Senn, 2014 for the initial spike-based learning rule, and Vafidis et al., 2022 for the rate-based formulation). The learning rule modifies these synaptic weights so that, after learning, *CS* inputs can predict the responses of the *RNN* to the *US*s.

Consider the synaptic weights from input neuron j to associative neuron i, for either the *RNN* or the *CS* inputs. The weights are updated continuously during the trial using the following rule:

$$\Delta W_{ij} = \eta(S) \left[f(V_i^{\text{s}}) - f(p' V_i^{\text{d}}) \right] P_j \tag{1.9}$$

where $\eta(S)$ is a variable learning rate that depends on the instantaneous level of a surprise signal S, p' is an attenuation constant derived below, and P_j is the postsynaptic potential in input neuron j.

The postsynpactic potential P_j has a simple closed form solution detailed in Vafidis et al., 2022. In particular, it is a low-passed filtered version of the neuron's firing rate, so that

$$P_j(t) = H(t) * r_j(t), \tag{1.10}$$

where $*$ denotes the convolution operator, and H is the transfer function given by

$$H(t) = \frac{1}{\tau_{\text{l}} - \tau_{\text{s}}} \left[\exp(-\frac{t}{\tau_{\text{l}}}) - \exp(-\frac{t}{\tau_{\text{s}}}) \right] u(t) \tag{1.11}$$

and $u(t)$ is the Heaviside step function that takes a value of 1 for $t > 0$ and a value of 0 otherwise.

As noted in Vafidis et al., 2022, for constant η the learning rule is a predictive coding extension of the classical Hebbian rule. When η is controlled by a surprise signal, as in our model, it can be thought of a predictive coding extension of a three-factor Hebbian rule (Frémaux, Sprekeler, and Gerstner, 2010; Frémaux and Gerstner, 2016).

Importantly, all of the terms in the learning rule are available at the synapses in the dendritic compartment, making this a local, biologically plausible learning rule. The firing rate of the neuron $f(V_i^{\text{s}})$ is available due to backpropagation of action potentials (Larkum, Zhu, and Sakmann, 1999). $f(p' V_i^{\text{d}})$ is a constant function of the local voltage V_i^{d} computed locally in the dendritic compartment even when the

somatic input is present. By definition, postsynaptic potentials are available at the synapse.

There are a total number of N_{train} training trials, divided among all *CS-US* pairs. After each training trial we measure the state of the *RNN* off-line by inputing one r_{cs} at a time without the *US*, keeping the network weights constant, and measuring the output produced by the model at that stage of the learning process.

***US* decoding**

Up to this point the model has been faithful to the biophysics of the brain. The next part of the model is designed to capture the variable learning rate η in equation 1.9, and thus is more conceptual in nature. Our goal here is simply to provide a plausible model of the factors affecting the learning rates for the *RNN*. As illustrated in fig. 1.1C, this part of the model involves three distinct computations: decoding the *US* from the *RNN* activity, computing expectations about upcoming *US*s, and computing the surprise signal *S*.

The brain must have a way to decode the upcoming *US*, or its presence, from the population activity in the *RNN* at any point during the trial. This prediction is represented by the time-dependent vector $\hat{r}_{\text{us}}(t)$. For the purposes of our model, we will use the optimal linear decoder D (dimension: $N_{\text{rnn}} \times N_{\text{inp}}$), so that

$$\hat{r}_{\text{us}}(t) = r_{\text{rnn}}(t)^{\intercal} D. \tag{1.12}$$

The optimal linear decoder D is constructed as follows. First, for each *US* $i = 1, ..., N_{\text{stim}}$ define the row vector ϕ_i describing the steady-state firing rate the each associative neuron that arises when it is presented alone. Then define an activity matrix Φ by stacking vertically these N_{stim} row vectors (dimension: $N_{\text{stim}} \times N_{\text{rnn}}$). Φ is built using the initial random weights W_{rnn}, before learning has taken place. Second, define a target matrix T (dimension: $N_{\text{stim}} \times N_{\text{inp}}$) to be the row-wise concatenated set of *US* input vectors r_{us}. Then, if D perfectly decodes the *US* from the *RNN* activity, when only the *US*s are presented, we must have that

$$\Phi D = T. \tag{1.13}$$

It then follows that

$$D = \Phi^{+} T, \tag{1.14}$$

where $^{+}$ denotes the Moore-Penrose matrix inverse. A desirable property of the Moore-Penrose inverse is that if equation 1.14 has more than one solutions, it provides the minimum norm solution, which results in the smoothest possible decoding.

Note that the decoder, which could be implemented in any downstream brain area requiring information about *US*s, is completely independent of the input representations of the *CS*s. Instead, it is determined before learning given only knowledge of the USs, and is kept fixed throughout training.

US expectation estimation

Since the *US*s are primary reinforcers, it is reasonable to assume that their representations, r^i_{us} for $i = 1, ..., N_{\text{stim}}$, are stored somewhere in the brain. Then an expectation for each *US* can be formed by

$$E^i(t) = \exp(-\kappa \|\hat{r}_{\text{us}}(t) - r^i_{\text{us}}\|^2), \tag{1.15}$$

where $\|\hat{r}_{\text{us}}(t) - r^i_{\text{us}}\|$ is Euclidean distance between the stored and the decoded representations for each *US* at time t, and κ controls the steepness of the Gaussian kernel. Recognizing that the ability to discriminate these patterns increases with the Hamming distance H_{d}, we set the precision to be inversely proportional to H_{d}, i.e., $\kappa = \left(\frac{8}{H_{\text{d}}}\right)^2$.

Note that E^i takes values between 0 and 1, and equals 1 only when the *US* is perfectly decoded (i.e., when $\hat{r}_{\text{us}} = r^i_{\text{us}}$). Thus, E^i can be interpreted as a probabilistic estimate for each *US* that is computed throughout the trial. To simplify the notation, we denote the expectation for the *US* associated with the trial as E.

Surprise based learning rates

The learning rule in equation 1.8 is gated by a well-documented surprise signal (Gerstner et al., 2018). This surprise signal diffuses across the brain, and activates learning in the *RNN*.

For each *US* the following surprise signal is computed throughout the trial:

$$S^i(t) = \delta(t - t_{\text{trig}}) \left(\mathbf{1}_{US^i} - E^i(t - t_{\text{syn}})\right), \tag{1.16}$$

where $\mathbf{1}_{US^i}$ is an indicator function for the presence of *US-i*, δ is the Dirac delta function and t_{trig} the time a surprise signal is triggered. In trials where the *US* appears, we set $t_{\text{trig}} = t_{\text{us-on}} + t_{\text{syn}}$, where $t_{\text{syn}} = 2 * \tau_{\text{s}} = 200\,ms$ is a synaptic transmission delay for the detection of the *US* which matches well perceptual delays (Picton, 1992). The expectation E^i also lags by the same amount, representing synaptic delays from the associative network to the surprise computation area. As can be seen in eq. (1.16), the more the *US* is expected upon its presentation, the

lower the surprise. In extinction trials, we set $t_{\text{trig}} = t_{\text{us-on}} + t_{\text{syn}} + t_{\text{wait}}$, where t_{wait} is a time after which a *US* is no longer expected to arrive. The overall surprise signal is given by:

$$S = \sum_i S^i. \tag{1.17}$$

The surprise signal S gives rise to neuromodulator release and uptake which determine the learning rate η. We assume that separate neuromodulators are at work for positive and negative surprise, and that they follow double-exponential dynamics (Cragg, Hille, and Greenfield, 2000).

Consider the case of positive surprise. The released and uptaken neuromodulator concentration C_{r}^+ and C_{u}^+ are given by:

$$\tau_{\text{r}} \frac{\mathrm{d}C_{\text{r}}^+}{\mathrm{d}t} = -C_{\text{r}}^+ + [S]_+ \tag{1.18}$$

and

$$\tau_{\text{u}} \frac{\mathrm{d}C_{\text{u}}^+}{\mathrm{d}t} = -C_{\text{u}}^+ + C_{\text{r}}^+ \tag{1.19}$$

where τ_{r} and τ_{u} are the neuromodulator release and uptake time constants, respectively, chosen to match the dopamine dynamics in fig. 1.1B in Cragg, Hille, and Greenfield, 2000.

Negative surprise is controlled by a different neuromodulator, described by the following analogous dynamics:

$$\tau_{\text{r}} \frac{\mathrm{d}C_{\text{r}}^-}{\mathrm{d}t} = -C_{\text{r}}^- + [-S]_+ \tag{1.20}$$

and

$$\tau_{\text{u}} \frac{\mathrm{d}C_{\text{u}}^-}{\mathrm{d}t} = -C_{\text{u}}^- + C_{\text{r}}^-. \tag{1.21}$$

The neuromodulator uptake concentrations control the learning rate:

$$\eta = \eta_0 \, (C_{\text{u}}^+ - C_{\text{u}}^-), \tag{1.22}$$

where η_0 is the baseline learning rate.

CS short-term memory circuit

During trace conditioning the *CS* disappears before the *US* appears, but an association is still learnt. This suggests that the brain maintains some short-term memory representation of the *CS* after it disappears. To capture this, we introduce a short-term memory *RNN* that maintains a (noisy) representation of the *CS*, denoted by $\hat{r}_{\text{cs}}$,

over time (for details, see "*CS* short-term memory circuit" in Methods). As shown in fig. 1.1D, the network is able to maintain short-term representations of the *CS* for several seconds before memory leak becomes considerable.

We now describe the short-term memory network used to maintain the $\hat{r}_{\text{cs}}$ representation that serves as input to the *RNN*.

To obtain a circuit that can maintain a short-term memory through persistent activity in the order of seconds (Wang, 2001), we train a separate recurrent neural network of point neurons using backpropagation through time (BPTT). These networks have been deemed to not be biologically plausible (although see Greedy et al., 2022). However, for the purposes of our model we are only interested in the end product of a short-term memory circuit, and not in how the brain acquired such a circuit. Thus, BPTT provides an efficient means of accomplishing this goal.

The memory circuit contains 64 neurons, and the vector of their firing rates r_{mem} obeys:

$$\tau_{\text{s}} \frac{\text{d}r_{\text{mem}}}{\text{d}t} = -r_{\text{mem}} + \left[W_{\text{mem}}\, r_{\text{mem}} + W_{\text{inp}}\, r_{\text{cs}} + b + n_{\text{mem}} \right]_{+} \tag{1.23}$$

where W_{mem} is a matrix with the connection weights between the memory neurons (dimension: 64×64), W_{inp} is a matrix of connection weights for the incoming *CS* inputs to the memory net (dimension: $64 \times N_{\text{inp}}$), τ_{s} is the same synaptic time constant described above, b is a unit-specific bias vector, and n_{mem} is a vector of IID Gaussian noise with zero mean and variance 0.01 added during training. A linear readout of the activity of the memory network provides the memory representation:

$$\hat{r}_{\text{cs}} = W_{\text{out}}\, r_{\text{mem}}, \tag{1.24}$$

where W_{out} is a readout matrix (dimension: $N_{\text{inp}} \times 64$).

The weight matrices W_{mem}, W_{inp}, and W_{out}, as well as the bias vector b, are trained as follows. Every trial lasts for 3 seconds. On trial onset, a Boolean vector r_{cs} is randomly generated and provided as input to the network. The *CS* input is provided for a random duration drawn uniformly from $[0.5, 2]$ seconds. The network is trained to output r_{cs} at all times for trials that are 3 seconds long. We train the network for a total of 10^7 trials in batches of 100. We use mean square error loss at the output, with a grace period 200 ms at the beginning of the trial where errors are not penalized. We optimize using Adam (Kingma and Ba, 2014) with default parameters (decay rates for first and second moments 0.9 and 0.999, respectively, learning rate 0.001). To facilitate BPTT, which does not scale well with the number of timepoints, we train the memory network using a time step of $10 * \Delta t$.

1.3 Network learns stimulus substitution in delay conditioning

Consider a delay conditioning experiment in which the animal needs to learn 16 *CS*-*US* pairs, and the timing of the trial is as shown in fig. 1.2A. Note that in this case the *CS* is present throughout the trial and, as a result, $\hat{r}_{\text{cs}} \approx r_{\text{cs}}$. Although the short-term memory network is not necessary in this particular experiment, we keep it in the model to maintain consistency across experiments.

We train the *RNN* for a total of 1000 trials. Figure 1.2B compares the actual representations of all the *US*s, one component at a time, with those decoded from the activity of the network in response only to the associated *CS*s. The network has accurately learnt all of the associations after 500 training trials ($\approx$ 32 per *CS*-*US* pair).

We next investigate how learning evolves with the amount of training. Figure 1.2C compares the activity of the associative neurons when presented only with the *US*, for all possible *CS*-*US* pairs, with their activity when presented only with the associated *CS*. Early in training, the associative neurons exhibit little activity in response to the *CS*s, and their responses are not correlated with the amount of activity elicited by the *US*s. By the end of training however, the neurons respond to the *CS* the same way they respond to the *US*, therefore stimulus substitution is achieved. A host of conditioning phenomena, detailed in following sections, follow from that. For further details on the trial dynamics of learning see the Supplementary Information (section "How does the *RNN* learn?"). Importantly, in the Supplements we also show that three-factor Hebbian learning rules fail at stimulus substitution in our experiments.

Figure 1.2D tracks the learning dynamics more closely. The green curve shows the average expectation E assigned to the *US*s at different stages of training. Perfect learning occurs when $E = 1$ for all *US*s. The red curve provides a measure of distance between the r_{us} and $\hat{r}_{\text{us}}$. We see that learning requires few repetitions per *CS*-*US* pair, and is substantially faster early on.

There are three sources of randomness in the model: (1) randomness in the sampling of *CS* and *US* sets, (2) randomness in the order in which the stimulus pairs are presented, and (3) randomness in the initialization of W_{rnn}, W_{cs} and W_{us}. In fig. 1.3 we explore the impact of this noise in our results by training 5 networks with different initializations and training schedules. We find that the level of random variation across training runs is small, and is mostly dominated by randomness in the sampling

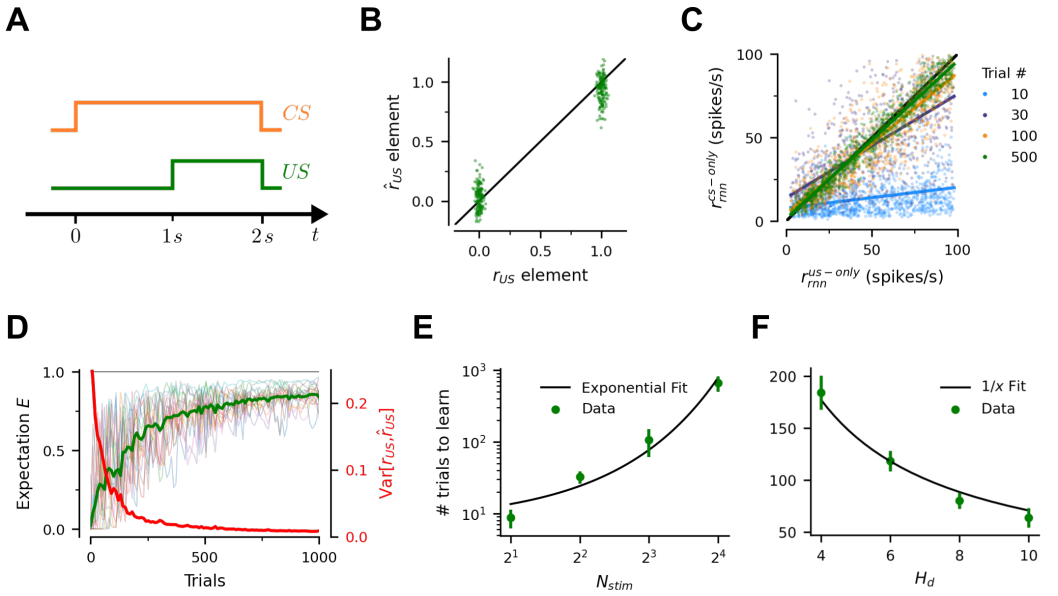

Figure 1.2: **Delay conditioning and stimulus substitution.** (A) Trial structure. The network is presented with $N_{\text{stim}} = 16$ different *CS-US* pairs, randomly selected in each trial. (B) The network learns all of the *CS-US* pairs after 500 training trials (≈ 32 per pair). r_{us} denotes the individual components of the Boolean vectors encoding each of the *US*s. $\hat{r}_{\text{us}}$ denotes the individual components of the decoded *US*s, based only on the presentation of the associated *CS*s, and measured just before the *US* appears. (C) Evolution of population responses during learning. Colors denote trial number. Each point compares the firing rate of an associate neuron at that stage of learning for a specific *CS-US* pair when only the *US*, or only the associated *CS* are presented. The colored lines are linear regression fits at each stage of learning. (D) Evolution of predicted *US* during learning. Green curve depicts the average expectation across *US*s after the network is presented only with the associated *CS*. Red curve depicts the distance between the true representation of the *US*s (r_{us}) and their decoded representation $\hat{r}_{\text{us}}$ when presented only with the associated *CS*. Individual pairs are shown in faint thin lines. (E) Number of trials required for the network to reach 80% performance for all pairs (defined as the first time at which the average expectation E across pairs exceeds 0.8) for different numbers of stimulus pairs. Performance is measured just before the *US* appears. Error bands denote $\mp$ SD computed across 5 different runs of the experiment. (F) Number of trials required to reach 80% performance for all pairs for different levels of similarity in the encoding of the *CS* and *US* input vectors. Error bands denote $\mp$ SD computed across 10 different runs of the experiment.

of the stimuli. For this reason, unless otherwise stated, we present results using only a single training run.

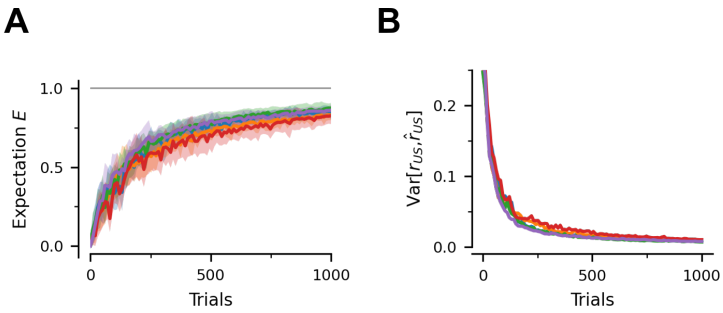

Figure 1.3: **Variation across training runs.** Each curve depicts a different training run. Bands represent the ∓ SD across stimulus pairs. (A) Expectation for each *US* after the network is presented only with the associated *CS*, averaged across all pairs at different levels of training. (B) Distance between the true representation of the *US*s (r_{us}) and their decoded representation $\hat{r}_{\text{us}}$ when presented only with the associated *CS*, averaged across all pairs at different levels of training.

Since the *RNN* uses mixed representations over the same neurons to encode the stimuli, one natural question is how does learning depend on the number of *CS*-*US* pairs in the experiment (N_{stim}) and on the similarity of their representations (r_{cs} vs r_{us}).

We explore the first question by training the model for different values of N_{stim} and then measuring the number of trials that it takes the network to reach a 80% level of maximum performance, defined as the level of training at which the average expectation E across pairs exceeds 0.8. Interestingly, the required number of trials increases exponentially with the number of *CS*-*US* pairs (fig. 1.2E). This is likely due to interference across pairs: learning of an association also results in unlearning of other associations at the single trial level. This interference gets worse as the number of stimuli N_{stim} increases (fig. 1.4), which might explain the exponential dependence. Finally, note that the network is capable of very fast learning when there are only a few pairs (about 5 presentations per pair for two pairs, fig. 1.2E).

We explore the second question by training the model for different values of the Hamming distances H_{d}, which provides a lower bound on the similarity among *US*s and, separately, among *CS*s. $N_{\text{stim}} = 8$ for these experiments. Perhaps unsurprisingly, the more dissimilar the stimulus representations, the faster the learning (fig. 1.2F). Figure 1.5 shows how smaller H_{d} naturally leads to greater interference across stimuli.

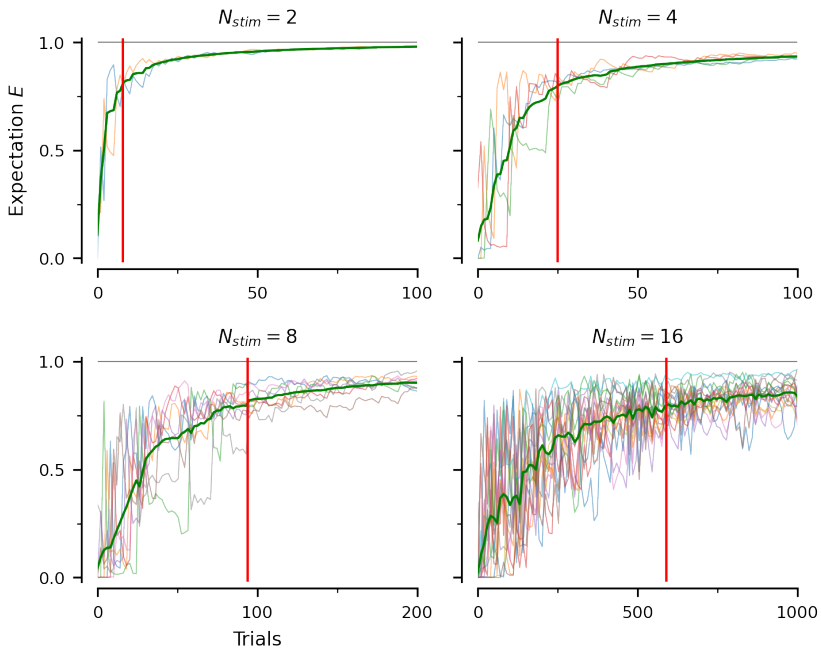

Figure 1.4: **Impact of the number of stimulus pairs on delay conditioning.** Learning paths for each *CS*-*US* pair for a single experimental run. Each thin line tracks the expectation E for a single stimulus pair. Note that the paths do not increase monotonically, which shows that there can be interference across pairs. The vertical read lines indicate the time at which the average E across pairs (thicker green line) reaches 80% performance level.

1.4 Short-term memory and trace conditioning

Next we consider trace conditioning experiments, in which there is a delay interval $t_{\text{delay}} > 0$ between the disappearance of the *CS* and the arrival of the *US* (fig. 1.6A). In this case the memory network is crucial for maintaining a memory trace of the *CS* to be associated with the *US*.

As before, we train the *RNN* for 1000 trials, with 16 different pairs, to explore how learning changes over time and how the delay $t_{\text{delay}} > 0$ affects learning. For comparison purposes, we include the case of delay conditioning in the same figures ($t_{\text{delay}} = -1$ s).

Figure 1.6B shows the quality of the decoded representation of the *US* and fig. 1.6C-D the strength of the associated expectation signal, both measured offline and in response only to the *CS*. We find that the *RNN* learns the associations well for small delays, but that the quality of the learning decays for larger delays. This pattern has been observed in animal experiments (Schneiderman and Gormezano, 1964), and

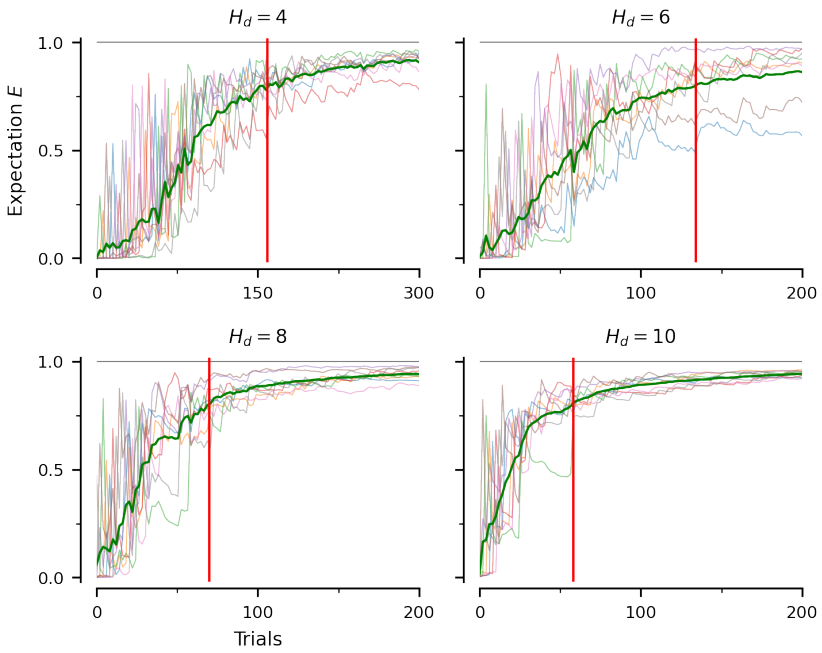

Figure 1.5: **Impact of the similarity on stimulus representation on delay conditioning.** Learning paths for each *CS*-*US* pair for a single experimental run. Each thin line tracks the expectation E for a single stimulus pair. Note that the paths do not increase monotonically, which shows that there can be interference across pairs. The vertical read lines indicate the time at which the average E across pairs (thicker green line) reaches 80% performance level.

the model provides a mechanistic explanation: conditioning worsens with increasing delays because the memory representation of the *CS* is leaky and degrades at longer delays, as shown in fig. 1.1D.

1.5 Extinction and re-acquisition

The model can also account for the phenomenon of extinction. To investigate this, we focus on the case in which the *RNN* only needs to learn a single *CS*-*US* pair in the delay conditioning task described before. We keep the same trial structure, except that the *US* is not shown at all, and the trial duration is extended (fig. 1.7A). The latter is important because in extinction, the computation of surprise in equation 1.16 is triggered t_{wait} seconds after the normal time the *US* would appear, where t_{wait} is the time after which the *US* is no longer expected. Without loss of generality, we set $t_{\text{wait}} = 5$ seconds.

As shown in fig. 1.7B, the network learns this association with a small number of trials. At this point the extinction regime is introduced by presenting the same *CS* in

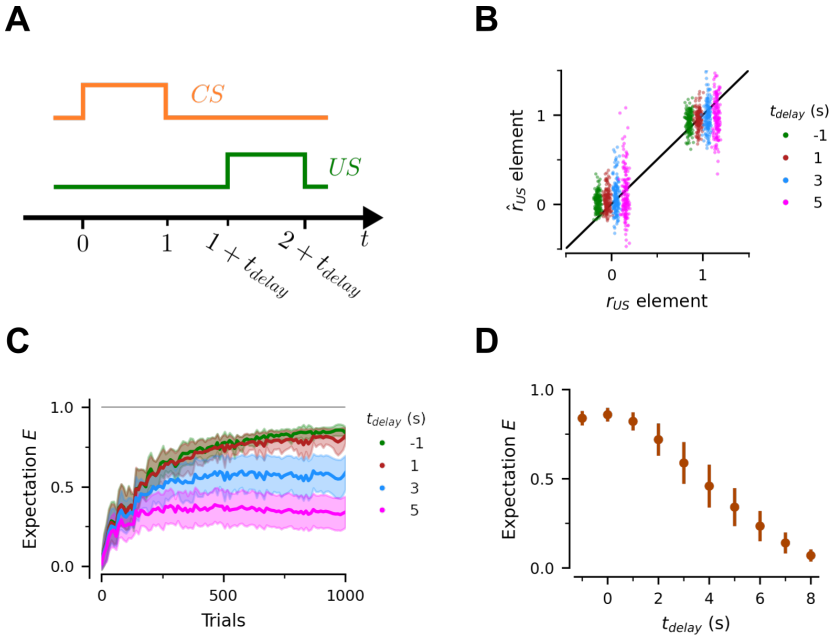

Figure 1.6: **Trace conditioning.** (A) Trial structure. The network is presented with $N_{\text{stim}} = 16$ different *CS*-*US* pairs, randomly selected in each trial. (B) After 500 training trials ($\sim$ 32 per pair), the network learns all of the *CS*-*US* pairs for short t_{delay}, but struggles for longer delays. r_{us} denotes the individual components of the Boolean vectors encoding each of the *US*s. $\hat{r}_{\text{us}}$ denotes the individual components of the decoded *US*s, based only on the presentation of the associated *CS*s. For comparison purposes, we also show results for delay conditioning ($t_{\text{delay}} = -1$) (C) Evolution of predicted *US* during learning. Each curve depicts the expectation for each *US* after the network is presented on ly with the associated *CS*. Line is the mean across all stimulus pairs. Bands represent the $\mp$ SD across stimulus pairs. (D) Network learning performance after 500 training trials for different *CS*-*US* delays. Bars denoted $\mp$ SD across stimulus pairs.

isolation, and as a result the learned association rapidly disappears from the network (fig. 1.7B,C).

Figure 1.7D looks at the phenomenon of re-acquisition where, after a period of extinction, the same *CS*-*US* pair is reintroduced in training. A common finding in many classical conditioning experiments is that re-acquisition is faster than the initial learning (Napier, Macrae, and Kehoe, 1992). To test this, we compare two cases: one in which the same *US* is used during re-acquisition (shown in blue), and one in which a different *US* is used during re-acquisition (shown in red). We find that re-learning an association to the same *US* is faster, therefore accounting for experimental findings on re-acquisition. Furthermore, our network provides a mechanistic explanation: re-acquisition is faster because the responses of the

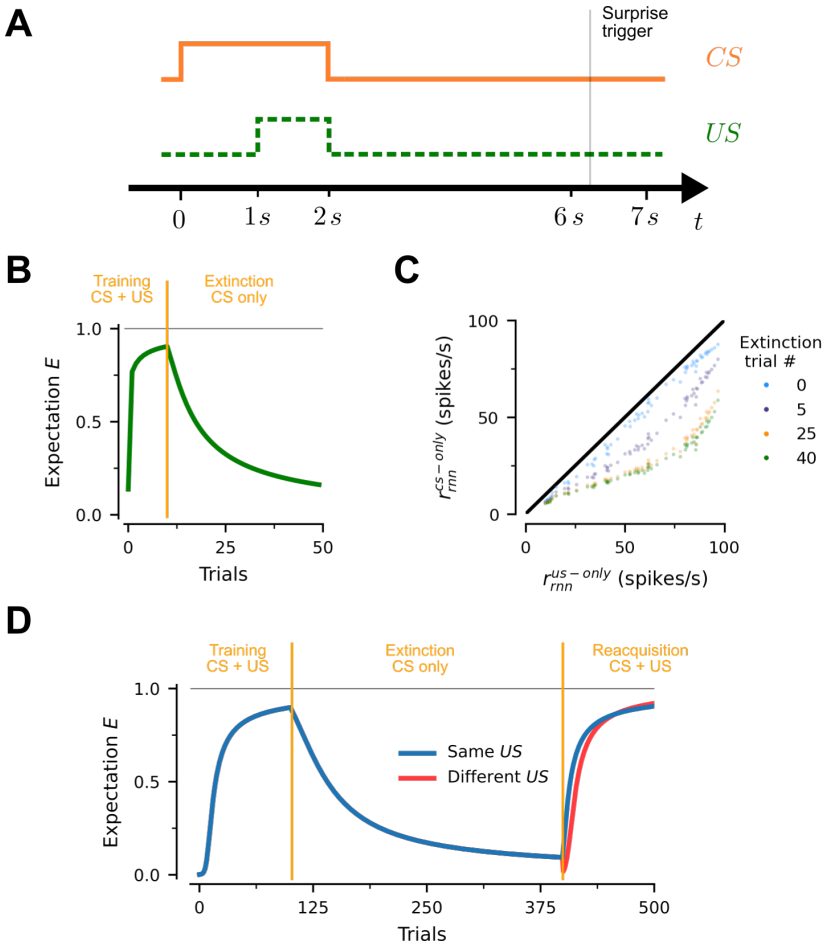

Figure 1.7: **Extinction and re-acquisition.** (A) Trial structure. In trials where there *US* is not shown, surprise is computed at $t \approx 6$ seconds. (B) Learning and extinction path for the acquisition of a single *CS-US* pair. (C) Evolution of population responses during extinction. Colors denote extinction trial number. Each point compares the firing rate of an associate neuron at that stage of learning for a specific *CS-US* pair when only the *US*, or only the associated *CS* are presented. (D) Learning, extinction and re-acquisition path. Blue line involves an experiment in which the same *CS-US* pair is used in training and re-acquisition. Red line involves an experiment in which a new *US* is used at the re-acquisition phase.

neurons in fig. 1.7C have not decayed to zero, even though the expectation almost has. Therefore, re-learning is faster to begin with, although the new pattern catches up later.

1.6 Phenomena arising from *CS* competition

So far we have focused on experiments in which the network needs to learn one-to-one *CS*-*US* pairings. However, some of the most interesting findings in conditioning arise when multiple *CS*s are associated with the same *US*.

To explore this, we extend the model to the case in which the network can be exposed to two *CS*s for each *US* (fig. 1.8A). Now there are two separate *RNN*s of associative neurons, one for each *CS*. Without loss of generality we focus on delay conditioning and therefore, for the sake of simplicity, we remove the short-term memory network and directly feed inputs for the respective *CS*s (denoted by r_{cs1} and r_{cs2}). The activity of these populations is used to decode the identity of the *US*, based on the activity generated by each *CS* separately. These predictions are then used to generate expectations E_{cs1} and E_{cs2}, which denote the predicted strength generated by each of them when shown in isolation. The total expectation for the *US* is then given by $E = E_{\text{cs1}} + E_{\text{cs2}}$. The same logic could be extended to more than two *CS*s. For all of these experiments, we learn a single association between a pair of *CS*s and a single *US*, i.e., $N_{\text{stim}} = 1$.

Figure 1.8B presents the results for a typical blocking experiment. We first present CS_1 alone for the first 100 trials, resulting in the acquisition of an expectation very close to 1. Subsequently, we start presenting both *CS*s together. However, the *US* is already well predicted from CS_1, resulting in small surprises after CS_2 is introduced, and thus an approximate zero learning rate. Thus, in this setting the model generates the well established phenomenon of blocking.

Figure 1.8C studies an overshadowing experiment. Here we present both *CS*s together from the first trial. In this case both of them develop an expectation from the *US*, but neither individually reaches 1. Instead, it is the sum of their expectations that learns the association. Thus, in this setting the model generates the well established phenomenon of overshadowing. Notice that the expectation stemming from one of the *CS*s is larger than the other, which can be attributed to randomness in the weight matrix initializations.

Figure 1.8D investigates the impact of stimulus saliency in *CS* competition. Salient stimuli receive more attention and generate stronger neural responses than similar but less salient ones (Gottlieb, Kusunoki, and Goldberg, 1998). We model relative saliency by multiplying the input vector r_{cs1} of CS_1, the high-saliency cue, by a constant $s_h = 1.2$, while keeping r_{cs2} the same. Otherwise, the task is identical to the case of overshadowing. Consistent with animal experiments, fig. 1.8D shows that

the more salient CS_1 acquires a substantially stronger association with the *US* than the less salient CS_2. This results from the fact that the more salient stimulus leads to higher firing rates, and thus to stronger pre-synaptic potentials which strengthen learning at those synapses.

Finally, fig. 1.8E presents the results for a typical overexpectation experiment. Here CS_1 is presented alone for the first 100 trials, CS_2 is then presented alone for the next 100, and starting from trial 200, both *CS*s are presented together. Since at this point the *CSs* already have expectations very close to 1, their joint expectation greatly surpasses 1. As a result, surprise is now negative, leading to unlearning of both conditioned responses, up to the point where $E_{\text{cs1}} + E_{\text{cs2}} \approx 1$.

1.7 Contingency and unconditional support

So far we have considered experiments that depend on the temporal contiguity of the *CS* and *US*. Another important variable affecting conditioning is contingency; i.e., the probability with which the *CS* and the *US* are presented together (Rescorla, 1968).

To vary the level of contingency, the *US* is shown in every trial, but the *CS*s are presented only with some probability, which we vary across experiments. Note that this is not the only way of running contingency conditioning experiments. For example, one could change the contingency by showing the *CS*s every trial and then only show the *US* with some probability. This would manipulate the degree of contingency, but also introduce an element of extinction, since there are some trials in which no *US* follows the *CS*. We favor the aforementioned experiment because it eliminates this confound.

Figure 1.9A involves experiments with a single *CS* which is shown with different probability. Consistent with the animal literature (Rescorla, 1968), we find that the strength and speed of learning increases with the *CS*-*US* contingency.

Figure 1.9B involves experiments with two independent predictive stimuli. Every trial CS_1 is shown with probability 0.8 and, independently, CS_2 is shown with probability 0.4. Unsurprisingly, we find that the *CS* with the highest contingency acquires the stronger predictive response. Note that the conditioned responses do not need to add up to 1 in this setting.

Figure 1.9C,D involves a different probabilistic structure for the *CS*s. CS_1 is shown every trial with probability 0.8, as in the previous case. But now CS_2 is only shown if CS_1 is present, and with various probability $P(CS_2|CS_1)$. When $P(CS_2|CS_1) = 0.5$,

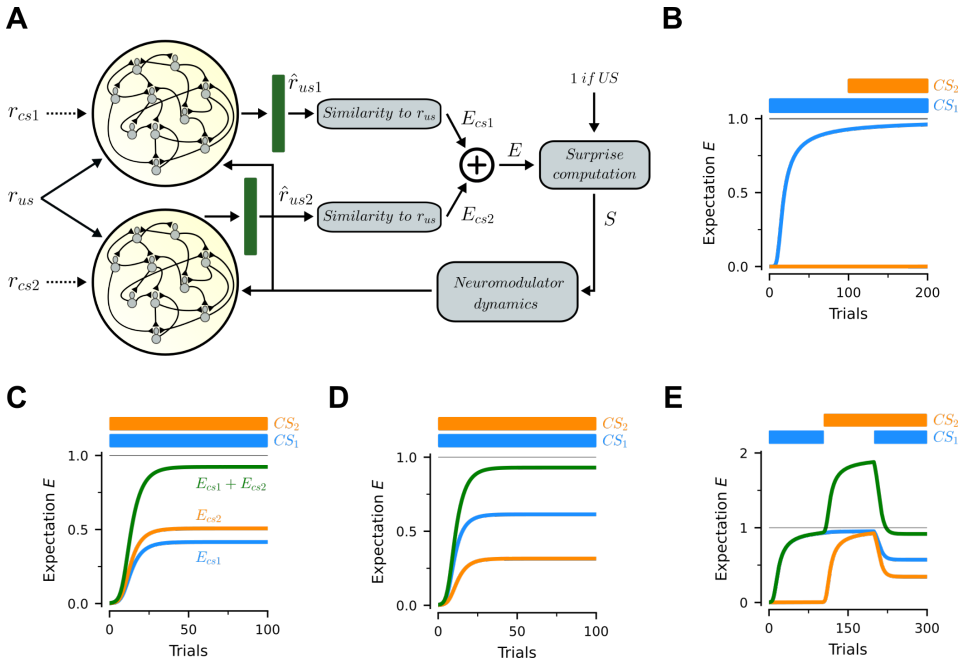

Figure 1.8: **Blocking, overshadowing, saliency and overexpectation.** (A) Model extension to allow for simultaneous presentation of two *CS*s. Associations for CS_1 and CS_2 are represented in separate populations of associative neurons. The activity of each population is used to separately decode the *US* and to construct expectations E_{cs1} and E_{cs2}. The overall expectation generated by the two *CS*s is given by $E = E_{\text{cs1}} + E_{\text{cs2}}$. Experiments assume that a single association between the *US* and both *CS*s has to be learnt. E_{cs1} is the prediction generated by CS_1 alone. E_{cs2} is the prediction generated by CS_2 alone. and $E_{\text{cs1}} + E_{\text{cs2}}$ is the prediction generated by both cues together. Since the *CS*s are present throughout the trial, we omit the short-term memory networks from this exercise. (B) Blocking: CS_1 is presented in isolation and fully learns to predict the *US* before CS_2 is introduced. In this case, CS_2 is blocked from learning to predict the *US*. (C) Overshadowing: Both *CS*s are presented from onset and none of them reaches the same conditioning level as when it was presented alone; instead, the sum E of their expectations learns the full association. (D) Saliency effects: similar to (C), but now the relative salience of CS_1 has been increased by scaling up its input vector. As a result, the final conditioning level of CS_1 is consistently higher than the one for CS_2. (D) Overexpectation: CS_1 and CS_2 are conditioned separately. When presented together, E exceeds 1, which leads to a negative learning rate and unlearning.

the unconditional probabilities of the two *CS*s are the same as in fig. 1.9B, but the associations learnt are different. After an initial acquisition phase, E_{cs2} decays monotonically to zero. More interestingly, the same effect arises if $P(CS_2|CS_1) = 0.875$, where $P(CS2) = 0.7$: even though the two *CS*s are similarly likely, E_{cs2}

decays to zero after initially going toe-to-toe with E_{cs1}. This exemplifies the heavily non-linear behavior of this phenomenon.

To explain this finding, we need to introduce the concept of *unconditional support*. A *CS* has unconditional support if there are trials when it is presented by itself, which means the network has to rely on it to predict the incoming *US*. In fig. 1.9B, both *CS*s have unconditional support, albeit CS_2's is much lower. This explains both the noisiness in E_{cs2}, which increases each time CS_2 is presented alone, and the fact that $E_{\text{cs2}} < E_{\text{cs1}}$. However, the situation drastically changes when CS_2 is only presented together with CS_1. Here CS_2 has no unconditional support. Initially, both *CS*s are conditioned, until the sum of their conditioned responses reaches 1. At that point no more positive surprise is generated for CS_2. When CS_1 is presented alone, $S > 0$ because $E_{\text{cs1}} < 1$, which leads to an increase in the E_{cs1} association. When both *CS*s are presented together, the sum of their conditioned responses is now greater than 1, and therefore $S < 0$ and both conditioned responses drop. As a result, over time E_{cs2} gradually decay to zero. This also explain why E_{cs} takes longer to decay when $P(CS_2|CS_1)$ is high.

In this task, CS_2 is a spurious predictor of the *US*, since it only appears if CS_1 is shown, and has no additional predictive value conditional on CS_1, as shown in fig. 1.9E. Essentially, the network learns to retain the predictive relationship but erase the spurious one. Importantly, we did nothing that would bias the network towards developing this strikingly non-linear effect.

A common fallacy of causal reasoning is known as the *post hoc ergo propter hoc* fallacy (Hamblin, 1970). It posits that the temporal proximity of two events is sufficient to infer that the earlier event is a contributing cause of the latter. This can lead to erroneous conclusions, when such temporal proximity is coincidental. In fig. 1.9C-E, CS_1 is predictive of both CS_2 and the *US*, but CS_2 is not predictive of the *US*, despite it preceding it temporally. Therefore, the network can recognize the lack of predictive ability (or unconditional support) of CS_2, resolving the *post hoc* fallacy in this simpler predictive setting. Similar mechanisms might allow the brain to perform more advanced forms of causal reasoning.

Finally, note that compared to other conditioning phenomena, the network takes substantially longer to learn the predictive structure of the task. Combined with the fact that real world data are scarce and often ambiguous, this might explain why such fallacies often persist.

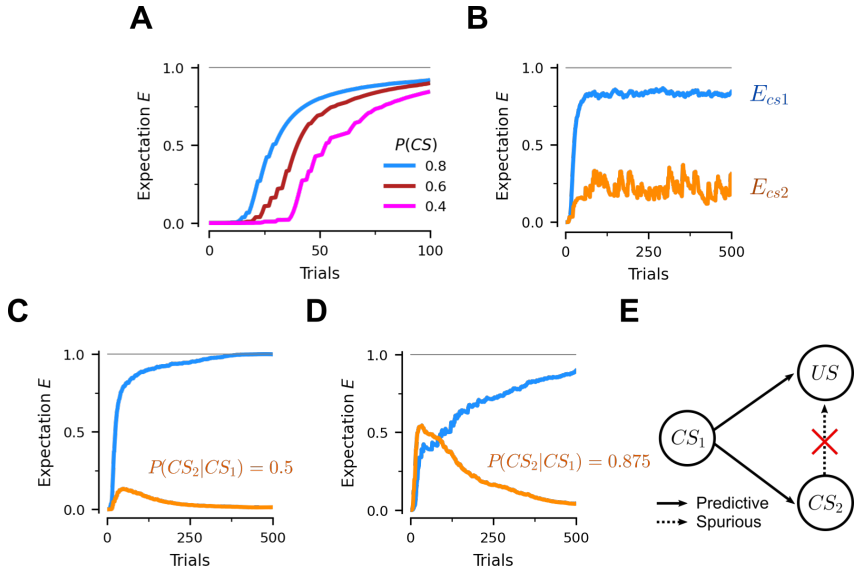

Figure 1.9: **Contingency and causality.** The *US* is shown every trial, while the contingency of the *CS*s is varied. (A) Impact of changing the probability of showing the *CS* in every trial. Each line depicts the learning path for a different experiment. (B) Experiment with two independent predictive stimuli. In every trial, CS_1 is shown with probability 0.8 and CS_2 is shown with probability 0.4. Blue curve is the expectation acquired by CS_1 when shown by itself. Orange curve is the expectation acquired by CS_2 when shown by itself. (C,D) Experiments with a conditional *CS* structure. Every trial CS_1 is shown with probability 0.8 and CS_2 is shown only if CS_1 is also present, with probability $P(CS_2|CS_1)$. (E) The network learns to ignore spurious predictors. Since CS_2 is conditionally dependent on CS_1, our network gradually phases out any explanatory power of CS_2, as more evidence that the *US* is never caused by the CS_2 by itself arrives.

1.8 Three-factor Hebbian learning fails at stimulus substitution

So far we have shown that our model with the predictive learning rule in equation 1.8 accounts for many common patterns in classical conditioning. A key feature of this rule is that learning is guided by a comparison of activity in the dendritic and somatic compartments of the associative neurons.

We now investigate whether the same network trained using previously proposed Hebbian plasticity rules is able to account for the same phenomena. To do this, we keep all of the model components unchanged except for the learning rule. We train the model with two widely used Hebbian-like plasticity rules, Oja's rule (Oja, 1982) and the BCM rule (Bienenstock, Cooper, and Munro, 1982). We find that the resulting network either cannot learn multiple associations, or requires task specific parameter tuning.

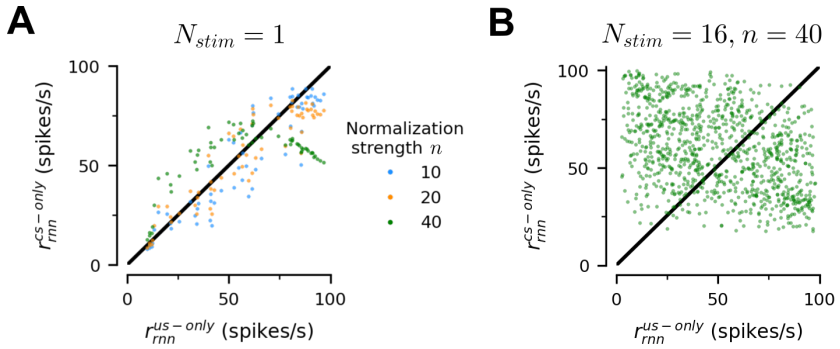

Figure 1.10: **Delay conditioning with Oja's rule.** Each point compares the firing rate of an associative neuron at that stage of learning for a specific *CS-US* pair when only the *US*, or only the associated *CS* are presented. Model is trained using Oja's rule. (A) Model learns stimulus substitution for different normalization strengths when $N_{\text{stim}} = 1$. Model trained for 100 trials with $\eta_0 = 2 * 10^{-4}$. (B) Models fails to learn after 1000 training trials (64 per *CS-US* pair) when $N_{\text{stim}} = 16$. For this experiment, we use $\eta_0 = 10^{-3}$.

Consider Oja's learning rule first. In this case the synaptic weights from input neuron j to associative neuron i are updated using the following rule:

$$\Delta W_{ij} = \eta(S) f(V_i^{\text{s}}) \left[P_j - n W_{ij} f(V_i^{\text{s}}) \right] \tag{1.25}$$

where n is the normalization strength. The normalization component is crucial, because otherwise learning would diverge. Normalization here focuses on the weights, and subjects the largest weights to the strongest normalization. We choose $n = 40$ for which the final responses to the *CS* span most of the output range of associative neurons in our model (0 – 100 spikes/s).

Figure 1.10 shows the results of training the *RNN* with this learning rule in the delay conditioning task. We find that the network can learn well when there is a single *CS-US* pair, but it fails when it has to learn multiple associations. In fact, $r_{\text{rnn}}^{\text{us-only}}$ and $r_{\text{rnn}}^{\text{cs-only}}$ are anti-correlated in this case. This occurs because normalization introduces competition between incoming synapses to the same neuron (Gerstner et al., 2014), which in turn induces competition between the associations to be stored and leads to interference. More specifically, neurons that fire strongly for one pattern will sustain the harshest normalization in their incoming weights affecting the response to all other patterns. The role of the normalization coefficient is minimal when learning a single association. This might be because final weight levels for active neurons are determined mostly by the firing rate $f(V_i^{\text{s}})$ which is constant for the same association, and hence it serves as a modulator of the learning rate.

Now consider the BCM rule, which involves an alternative normalization strategy that, instead of focusing on the weights, sets a variable potentiation threshold for the postsynaptic firing rate. The rule is given by:

$$\Delta W_{ij} = \eta(S) f(V_i^{\mathrm{s}}) \left[f(V_i^{\mathrm{s}}) - \alpha\theta_i \right] P_j \tag{1.26}$$

where θ_i is a time-varying threshold, and α is a parameter that modulates the size of the threshold. A common choice is to make the threshold a function of the average recent firing rate, which we implement by making it an exponential moving average of the firing rate though the following differential equation:

$$\tau_\theta \frac{\mathrm{d}\theta_i}{\mathrm{d}t} = -\theta_i + f(V_i^{\mathrm{s}}) \tag{1.27}$$

where the parameter τ_θ determines the temporal window of integration. In theory, this approach sounds promising, since if $r_{\mathrm{rnn},i}^{\text{cs-only}} < r_{\mathrm{rnn},i}^{\text{both}}$, then $f(V_i^{\mathrm{s}}) > \theta_i$ leading to potentiation and vice versa, with this logic converging to $r_{\mathrm{rnn},i}^{\text{cs-only}} \approx r_{\mathrm{rnn},i}^{\text{both}}$. However, as we show this is not enough to guarantee the performance of the BCM rule.

Figure 1.11 shows the results of training the *RNN* with the BCM rule and $\alpha = 1$. We find that with the same trial conditions used for our main results (as shown in fig. 1.2) the BCM rule generates intermediate amounts of conditioning, as it has a tendency to overshoot. Furthermore, when we change the time at which the *US* appears conditioning becomes even worse, and that the problem persists for different values of τ_θ. Since the BCM rule has a tendency of underestimating the impact of the *CS*, we also explored a remedy that involved amplifying the threshold by setting $\alpha = 1.05$. This can fix the problem for experiments in which $t_{us-on} = 1$ s, but the performance of the network is still highly dependent on *US* timing. This is because the threshold, determined by a moving averaging filter of the firing rate, is highly dependent on trial specifics. Therefore, we conclude that the time-dependent threshold of the BCM rule introduces sensitivity to experimental details that cannot be overcome.

Overall, the need to fine-tune the parameters of the BCM rule to specific trial details is a general problem of Hebbian learning rules, stemming from the fact that they lack supervision. A similar point has been made by Vafidis et al., 2022; Stringer et al., 2002. In contrast, predictive learning does not demonstrate such sensitivity. The network using our predictive learning rule learns the task well for a variety of *US* onset times, without any explicit parameter tuning.

These results showcase the importance of the predictive learning rule in this work, facilitated by the two-compartment nature of the associative neurons. The existence

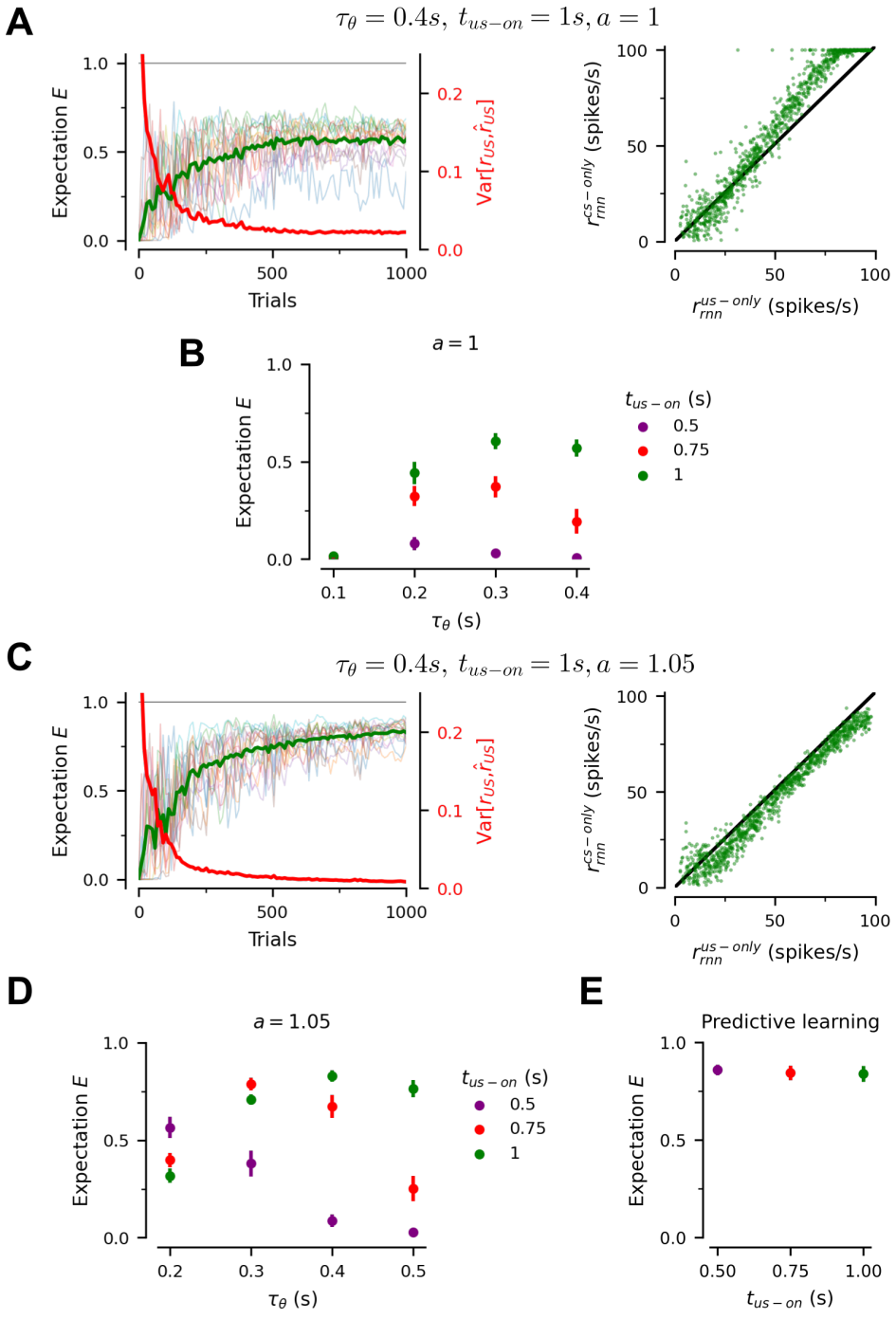

Figure 1.11: **Delay conditioning with BCM rule.** Model trained with $\eta_0 = 0.3$ for 1000 trials in order to learn $N_{\text{stim}} = 16$ associations. Model parameters and task conditions vary across panels. (A,C) Left: learning path. Green curve depicts the average expectation across *CS-US* pairs. Learned expectations for individual pairs are shown in faint thin lines. Right: firing rates of all associative neurons after training. (B,D) Network performance, as measured by E, as a function of the parameter τ_θ in the BCM rule and the timing at which the *US* is presented. (E) Learnt *US* expectations with our proposed predictive learning rule for different *US* timings. In contrast to the BCM rule, predictive learning is insensitive to experimental details.

of two compartments, which separate *CS* inputs to the dendritic compartment from *US* inputs to the somatic compartment, makes it possible for the biologically plausible learning rule in eq. 1.8 to compare the two and guide learning using only information locally available at the synapse. In this learning rule, the activity of the somatic compartment serves as a supervisory signal for learning the weights of the inputs to the dendritic compartment until they are able to fully predict their activity in response to the *US*. In contrast, we have shown that two canonical Hebbian rules struggle with this type of associative learning, in part because they do not have an analogous supervisory signal.

1.9 Discussion

The ability to engage in stimulus-stimulus associative learning provides a crucial evolutionary advantage. The cerebral cortex might contribute to this evolutionary edge by exploiting representational (Rigotti et al., 2013) and architectural (Larkum, Zhu, and Sakmann, 1999) inductive biases present in the cortical microcircuit (Nieuwenhuys, 1994). We here propose a recurrent neuronal network model of how the cortex can implement stimulus substitution, which allows the same set of neurons to encode multiple stimulus-stimulus associations. The model relies on the properties of two-compartment layer-5 pyramidal neurons, which based on recent experimental findings, we refer to as associative neurons. These neurons can act as coincidence detectors for information about the *US* arriving at their somatic compartment and information about the *CS* arriving at their dendritic compartment (Larkum, Zhu, and Sakmann, 1999; Larkum, 2013; Doron et al., 2020). Coincidence detection allows for a biologically plausible synaptic plasticity rule that, after learning, results in neurons that would normally fire in the presence of the *US* to respond in the same manner when the *CS* is presented. At the population level, this means that the pattern of neural activity corresponding to the *CS* can be morphed into the one corresponding to the *US*, leading to stimulus substitution.

Our model accounts for many of most important conditioning phenomena observed in animal experiments, including delay conditioning, trace conditioning, extinction, blocking, overshadowing, saliency effects, overexpectation and contingency effects. The model is able to learn multiple *CS*-*US* associations with a degree of training that is commensurate with animal experiments. Significantly, the model performs well across a wide variety of conditioning tasks without experiments-specific parameter fine-tuning.

We also show that some influential models of three-factor Hebbian learning rules - Oja's rule (Oja, 1982) and the BCM rule (Bienenstock, Cooper, and Munro, 1982) - fail to learn generic stimulus-stimulus associations due to their unsupervised nature. Hebbian rules have demonstrable autoassociative (Hopfield, 1982) and heteroassociative (Sompolinsky and Kanter, 1986) capabilities, and when augmented with eligibility traces they have been shown to account for neuronal-level reinforcement learning (Urbanczik and Senn, 2009; Vasilaki et al., 2009; Frémaux, Sprekeler, and Gerstner, 2010). Still, they struggle with pattern-to-pattern associations when representations are mixed. This is because Hebbian rules are purely unsupervised, and therefore provide no guarantee that the impact of the *CS* will be eventually shaped to be identical to the one of the *US*. Instead, network performance heavily depends on implementation details, like training history, task details and stimulus statistics. As a result, decoding from a population encoding several associations is hampered by the fact that activation levels for individual neurons when exposed to the *CS* will more often than not be off from those resulting from exposure to the corresponding *US*.

Related work utilized a predictive learning rule similar to the one used here to account for prospective coding of anticipated stimuli (Brea et al., 2016). While prospective coding might also be involved in conditioning, their study differs in several ways. First, their learning rule is timing-dependent; it succeeds in a delayed pair associative learning task, but it would require re-learning when the relative timing of the *US* in relation to the *CS* is variable. In contrast, our learning rule applies to arbitrary task timings. Second, their learning rule lacks gating which, unless strict conditions are met (dendritic and somatic activity conditioned on a stationary Markov chain), leads to reduced responses and even catastrophic forgetting. Furthermore, adding gating is not feasible in their model, because learning needs to bootstrap before the presentation of the delayed stimulus, and gating would inactivate learning at these times.

Several features of the model are worth emphasizing.

First, the proposed *RNN* leverages architectural inductive biases in the form of two-compartment associative neurons. These associative neurons are the most common neuron type in the mammalian cortex (Nieuwenhuys, 1994). This is likely no coincidence; once evolution stumbled upon their usefulness in predicting external contingencies, it might have favored them. While subcortical (Christian and Thompson, 2003) and even single-neuron (Gershman et al., 2021) mechanisms for

conditioning exist, the mechanism that we propose can handle mixed representations, and thus allow animals with a cerebral cortex to flexibly learn large numbers of associations.

The structure of the associative neuron is ideal for stimulus-stimulus learning. Feedforward inputs, like the *US* representations, arrive near the soma in layer-5 and directly control the neuron's firing rate. Feedback inputs, like the *CS* representations and the activity of other cortical neurons, arrive at the distal dentrites in layer-1 (Larkum, 2013). This compartmentalized structure allows the signals to travel independently, and get associated via a cellular mechanism known as BAC firing (Larkum, Zhu, and Sakmann, 1999). Specifically, it has been shown that these cells implement coincidence detection, whereby feedforward inputs trigger a spike which backpropagates to the distal dendrites and concurrently feedback input arrives at these dendrites, then plateau calcium potentials are initiated in the dendritic compartment (Larkum, Zhu, and Sakmann, 1999). These plateau potentials result in the neuron spiking multiple times subsequently and learning occurs in the distal dendrites, so that feedback inputs can elicit spikes alone in the future, without the need for external information.

Second, a prerequisite for the biological plausiblity of the learning rule used in the model is that backpropagating action potentials to be disentangled from postsynaptic potentials at the dendritic compartment. Only then can the two critical components in our learning rule, $f(V^{\mathrm{s}})$ and $f(p'V^{\mathrm{d}})$ in eq. 1.8 be compared. Since backpropagating action potentials (denoted by $f(V^{\mathrm{s}})$ in the model) do not need to travel far, they experience minimal attenuation (Larkum, Zhu, and Sakmann, 1999) and therefore they maintain some of their high-frequency components, which could be used at synapses to differentiate them from slower postsynaptic potentials (denoted by V^{d} in the model). As a result, only a static transformation of this last term is needed to compare the two signals. Consequently, the learning rule relies only on information locally available at each synapse, which is a prerequisite for biological plausibility.

Third, our model suggests multiple functional roles for gating. It limits learning to episodes that appear to have behavioral significance. Gating also prevents drifting of learned associations due to a lack of perfect self-consistency between $f(V^{\mathrm{s}})$ and $f(p'V^{\mathrm{d}})$ in the learning rule (Urbanczik and Senn, 2009), which is expected in a biological system subject to noise and approximate computation. In addition, gating provides a critical global reference signal when multiple *CS*s are available at the same time.

The model also has some limitations to be addressed in future work. Most importantly, it does not account for spontaneous recovery of previously learnt associations after extinction. In our model, extinction stems from the decay of the response of the associate neurons to the *CS*, a mechanism akin to unlearning, which erases previous learning, and thus does not allow for spontaneous recovery or faster re-acquistion. The extinction mechanism proposed here is complementary to inhibitory learning, the mechanism initially put forth by Pavlov to explain spontaneous recovery.

In the case of experiments with multiple *CS*s, the model assumes that different neuronal population implements separate *RNN*s to learn the associations for each of them. Although the two populations interact indirectly through the surprise signals, they each learn to predict the *US* on their own. The existence of separate populations might be justifiable when the *CS*s involve different sensory modalities (e.g., sound and vision), or very different spatial locations, but not necessarily when they are presented simultaneously. Extending the model to include differential routing of simultaneously presented stimuli is an open question for future work.

Another direction for future work is to account for more psychological aspects of conditioning by developing a larger model that incorporates other forms of learning and generalization like model-based strategies also thought to take place in the PFC (Wang et al., 2018), or to allow for context-dependent computation to resolve conflicts among competing stimuli (Mante et al., 2013). In these larger models, our network would model the stimulus substitution component.

The model allows to differentiate between conditioning effects that can be accounted by low-level, synaptic plasticity mechanisms, versus other high level explanations. At its core, the model performs stimulus substitution at the neuronal level, via a gradual acquisition process (Thorndike, 1898; Rescorla and Wagner, 1972; Sutton and Barto, 1981). Despite that, the model is still capable of rapid, few-shot learning, especially when the number of associations is small compared to size of the network (fig. 1.2E). Yet, for rapid learning in more complicated scenarios, fast inference based on prior knowledge might be necessary (Lake et al., 2016).

Finally, our model suggests an alternative role for representational inductive biases in the form of mixed selectivity, other than readout flexibility (Fusi, Miller, and Rigotti, 2016): it allows the efficient packing of multiple stimulus-stimulus associations within the same neuronal population, which might confer cortical animals the evolutionary edge.

References

Balkenius, Christian and Jan Morén (1998). "Computational models of classical conditioning: a comparative study." Working Paper. URL: http://www.lucs.lu.se/People/Christian.Balkenius/PostScript/LUCS62.pdf.

Bienenstock, Elie L., Leon N. Cooper, and Paul W. Munro (Jan. 1982). "Theory for the development of neuron selectivity: Orientation specificity and binocular interaction in visual cortex." In: *Journal of Neuroscience* 2.1, pp. 32–48. URL: http://www.jneurosci.org/content/2/1/32.long.

Brea, Johanni et al. (June 2016). "Prospective coding by spiking neurons." In: *PLoS Computational Biology* 12.6. Ed. by Peter E. Latham, e1005003. DOI: 10.1371/journal.pcbi.1005003. URL: https://doi.org/10.1371/journal.pcbi.1005003.

Christian, Kimberly M. and Richard F. Thompson (Nov. 2003). "Neural substrates of eyeblink conditioning: Acquisition and retention." In: *Learning & Memory* 10.6, pp. 427–455. DOI: 10.1101/lm.59603. URL: https://doi.org/10.1101/lm.59603.

Cragg, Stephanie J., Christopher J. Hille, and Susan A. Greenfield (Nov. 2000). "Dopamine release and uptake dynamics within nonhuman primate Striatum in vitro." In: *The Journal of Neuroscience* 20.21, pp. 8209–8217. DOI: 10.1523/jneurosci.20-21-08209.2000. URL: https://doi.org/10.1523/jneurosci.20-21-08209.2000.

Dai, Jiaman and Qian-Quan Sun (Sept. 2023). "Learning induced neuronal identity switch in the superficial layers of the primary somatosensory cortex." DOI: 10.1101/2023.08.30.555603. URL: http://dx.doi.org/10.1101/2023.08.30.555603.

Doron, Guy et al. (2020). "Perirhinal input to neocortical layer 1 controls learning." In: *Science* 370.6523, eaaz3136. ISSN: 0036-8075. DOI: 10.1126/science.aaz3136. eprint: https://science.sciencemag.org/content/370/6523/eaaz3136.full.pdf. URL: https://science.sciencemag.org/content/370/6523/eaaz3136.

Frémaux, Nicolas and Wulfram Gerstner (Jan. 2016). "Neuromodulated spike-timing-dependent plasticity, and theory of three-factor learning rules." In: *Frontiers in Neural Circuits* 9. DOI: 10.3389/fncir.2015.00085. URL: https://doi.org/10.3389/fncir.2015.00085.

Frémaux, Nicolas, Henning Sprekeler, and Wulfram Gerstner (Oct. 2010). "Functional requirements for reward-modulated spike-timing-dependent plasticity." In: *Journal of Neuroscience* 30.40, pp. 13326–13337. DOI: 10.1523/jneurosci.6249-09.2010. URL: https://doi.org/10.1523/jneurosci.6249-09.2010.

Fusi, Stefano, Earl K. Miller, and Mattia Rigotti (Apr. 2016). "Why neurons mix: high dimensionality for higher cognition." In: *Current Opinion in Neurobiology* 37, pp. 66–74. ISSN: 0959-4388. DOI: 10.1016/j.conb.2016.01.010. URL: http://dx.doi.org/10.1016/j.conb.2016.01.010.

Gershman, Samuel J. et al. (Jan. 2021). "Reconsidering the evidence for learning in single cells." In: *eLife* 10. DOI: 10.7554/elife.61907. URL: https://doi.org/10.7554/elife.61907.

Gerstner, Wulfram et al. (July 2014). "Neuronal dynamics." Cambridge, England: Cambridge University Press.

Gerstner, Wulfram et al. (July 2018). "Eligibility traces and plasticity on behavioral time scales: Experimental support of neo-Hebbian three-factor learning rules." In: *Frontiers in Neural Circuits* 12. DOI: 10.3389/fncir.2018.00053. URL: https://doi.org/10.3389/fncir.2018.00053.

Gottlieb, Jacqueline P., Makoto Kusunoki, and Michael E. Goldberg (Jan. 1998). "The representation of visual salience in monkey parietal cortex." In: *Nature* 391.6666, pp. 481–484. ISSN: 1476-4687. DOI: 10.1038/35135. URL: http://dx.doi.org/10.1038/35135.

Greedy, Will et al. (2022). "Single-phase deep learning in cortico-cortical networks." In: *Advances in Neural Information Processing Systems* 35, pp. 24213–24225.

Hamblin, Charles Leonard (Mar. 1970). "Fallacies." en. London, England: Methuen Young Books.

Hopfield, John J. (Apr. 1982). "Neural networks and physical systems with emergent collective computational abilities." In: *Proceedings of the National Academy of Sciences* 79, pp. 2554–2558. URL: https://www.pnas.org/content/79/8/2554.

Izhikevich, Eugene M. (Jan. 2007). "Solving the distal reward problem through linkage of STDP and dopamine signaling." In: *Cerebral Cortex* 17.10, pp. 2443–2452. DOI: 10.1093/cercor/bhl152. URL: https://doi.org/10.1093/cercor/bhl152.

Jenkins, H. M. and Bruce R. Moore (Sept. 1973). "The form of the auto-shaped response with food or water reinforcers." In: *Journal of the Experimental Analysis of Behavior* 20.2, pp. 163–181. DOI: 10.1901/jeab.1973.20-163. URL: https://doi.org/10.1901/jeab.1973.20-163.

Kingma, Diederik and Jimmy Ba (Dec. 2014). "Adam: a method for stochastic optimization." In: *International Conference on Learning Representations*.

Klopf, Harry (1988). "A neuronal model of classical conditioning." In: *Psychobiology* 16, pp. 85–125. DOI: 10.3758/BF03333113. URL: https://doi.org/10.3758/BF03333113.

Lake, Brenden M. et al. (Nov. 2016). "Building machines that learn and think like people." In: *Behavioral and Brain Sciences* 40, e253. DOI: 10.1017/s0140525x16001837. URL: https://doi.org/10.1017/s0140525x16001837.

Larkum, Matthew E. (Mar. 2013). "A cellular mechanism for cortical associations: An organizing principle for the cerebral cortex." In: *Trends in Neurosciences* 36.3, pp. 141–151. DOI: 10.1016/j.tins.2012.11.006. URL: https://doi.org/10.1016/j.tins.2012.11.006.

Larkum, Matthew E., J. Julius Zhu, and Bert Sakmann (Mar. 1999). "A new cellular mechanism for coupling inputs arriving at different cortical layers." In: *Nature* 398.6725, pp. 338–341. DOI: 10.1038/18686. URL: https://doi.org/10.1038/18686.

Mante, Valerio et al. (Nov. 2013). "Context-dependent computation by recurrent dynamics in prefrontal cortex." In: *Nature* 503.7474, pp. 78–84. DOI: 10.1038/nature12742. URL: https://doi.org/10.1038/nature12742.

Napier, Renée M., Michaela Macrae, and E. James Kehoe (1992). "Rapid reacquisition in conditioning of the rabbit's nictitating membrane response." In: *Journal of Experimental Psychology: Animal Behavior Processes* 18.2, pp. 182–192. ISSN: 0097-7403. DOI: 10.1037/0097-7403.18.2.182. URL: http://dx.doi.org/10.1037/0097-7403.18.2.182.

Nieuwenhuys, Rudolf (Oct. 1994). "The neocortex. An overview of its evolutionary development, structural organization and synaptology." In: *Anatomy and Embryology* 190.4. DOI: 10.1007/bf00187291. URL: https://doi.org/10.1007/bf00187291.

Oja, Erkki (Nov. 1982). "Simplified neuron model as a principal component analyzer." In: *Journal of Mathematical Biology* 15.3, pp. 267–273. DOI: 10.1007/bf00275687. URL: https://doi.org/10.1007/bf00275687.

Picton, Terence W. (Oct. 1992). "The P300 wave of the human event-related potential." In: *Journal of Clinical Neurophysiology* 9.4, pp. 456–479. ISSN: 0736-0258. DOI: 10.1097/00004691-199210000-00002. URL: http://dx.doi.org/10.1097/00004691-199210000-00002.

Rescorla, Robert A. (1968). "Probability of shock in the presence and absence of cs in fear conditioning." In: *Journal of Comparative and Physiological Psychology* 66.1, pp. 1–5. DOI: 10.1037/h0025984. URL: https://doi.org/10.1037/h0025984.

Rescorla, Robert A. and Allan R. Wagner (1972). "A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement." In: *Current Research and Theory*, pp. 64–99.

Rigotti, Mattia et al. (May 2013). "The importance of mixed selectivity in complex cognitive tasks." In: *Nature* 497.7451, pp. 585–590. DOI: 10.1038/nature12160. URL: https://doi.org/10.1038/nature12160.

Schneiderman, Neil and I. Gormezano (Apr. 1964). "Conditioning of the nictitating membrane of the rabbit as a function of CS-US interval." In: *Journal of Comparative and Physiological Psychology* 57.2, pp. 188–195. ISSN: 0021-9940. DOI: 10.1037/h0043419. URL: http://dx.doi.org/10.1037/h0043419.

Shin, Jiyun N., Guy Doron, and Matthew E. Larkum (Oct. 2021). "Memories off the top of your head." In: *Science* 374.6567, pp. 538–539. DOI: 10.1126/science.abk1859. URL: https://doi.org/10.1126/science.abk1859.

Sompolinsky, Haim and Ido Kanter (Dec. 1986). "Temporal association in asymmetric neural networks." In: *Physical Review Letters* 57.22, pp. 2861–2864. DOI: 10.1103/physrevlett.57.2861. URL: https://doi.org/10.1103/physrevlett.57.2861.

Stringer, S. M. et al. (May 2002). "Self-organizing continuous attractor networks and path integration: one-dimensional models of head direction cells." In: *Network: Computation in Neural Systems* 13.2, pp. 217–242. URL: https://www.ncbi.nlm.nih.gov/pubmed/12061421.

Sutton, Richard S. and Andrew G. Barto (1981). "Toward a modern theory of adaptive networks: Expectation and prediction." In: *Psychological Review* 88.2, pp. 135–170. DOI: 10.1037/0033-295x.88.2.135. URL: https://doi.org/10.1037/0033-295x.88.2.135.

– (1987). "A temporal-difference model of classical conditioning." In: *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*. Seattle, WA, pp. 355–378.

Thorndike, Edward L. (1898). "Animal intelligence: An experimental study of the associative processes in animals." In: *The Psychological Review: Monograph Supplements* 2.4, pp. i–109. DOI: 10.1037/h0092987. URL: https://doi.org/10.1037/h0092987.

Urbanczik, Robert and Walter Senn (Feb. 2009). "Reinforcement learning in populations of spiking neurons." In: *Nature Neuroscience* 12.3, pp. 250–252. DOI: 10.1038/nn.2264. URL: https://doi.org/10.1038/nn.2264.

– (Feb. 2014). "Learning by the dendritic prediction of somatic spiking." In: *Neuron* 81.3, pp. 521–528. DOI: 10.1016/j.neuron.2013.11.030. URL: https://doi.org/10.1016/j.neuron.2013.11.030.

Vafidis, Pantelis et al. (June 2022). "Learning accurate path integration in ring attractor models of the head direction system." In: *eLife* 11. Ed. by Srdjan Ostojic, Ronald L Calabrese, and Hervé Rouault, e69841. ISSN: 2050-084X. DOI: 10.7554/eLife.69841. URL: https://doi.org/10.7554/eLife.69841.

Vasilaki, Eleni et al. (Dec. 2009). "Spike-based reinforcement learning in continuous state and action space: when policy gradient methods fail." In: *PLoS Computational Biology* 5.12. Ed. by Karl J. Friston, e1000586. DOI: 10.1371/journal.pcbi.1000586. URL: https://doi.org/10.1371/journal.pcbi.1000586.

Wang, Jane X. et al. (May 2018). "Prefrontal cortex as a meta-reinforcement learning system." In: *Nature Neuroscience* 21.6, pp. 860–868. DOI: 10.1038/s41593-018-0147-8. URL: https://doi.org/10.1038/s41593-018-0147-8.

Wang, Xiao-Jing (Aug. 2001). "Synaptic reverberation underlying mnemonic persistent activity." In: *Trends in Neurosciences* 24.8, pp. 455–463. DOI: 10.1016/s0166-2236(00)01868-3. URL: https://doi.org/10.1016/s0166-2236(00)01868-3.

Chapter 2

LEARNING ACCURATE PATH INTEGRATION IN RING ATTRACTOR MODELS OF THE HEAD DIRECTION SYSTEM

Vafidis, Pantelis et al. (June 2022). "Learning accurate path integration in ring attractor models of the head direction system." In: *eLife* 11. Ed. by S. Ostojic, R. L. Calabrese, and H. Rouault, e69841. ISSN: 2050-084X. DOI: 10.7554/eLife.69841. URL: https://doi.org/10.7554/eLife.69841.

In Chapter I, we explored how associative neurons can learn static stimulus-to-stimulus mappings using a compartmentalized learning rule with surprise-based gating. Here, we extend this approach to the more complex domain of angular path integration, where neurons must learn a dynamic mapping from current head direction and angular velocity to future head direction. We employ a similar learning rule based on compartmentalized neurons, but without the gating mechanism, focusing instead on how this rule can shape a network that accurately integrate angular velocity inputs to perform angular path-integration in darkness, in the absence of external cues. Turns out that the resulting circuit is a one-dimensional continuous attractor (CAN), or ring-attractor, a circuit long hypothesized by theorists and relatively recently discovered by experimentalists in the fruit fly (Drosophila) head-direction system. The learned neural network can indeed perform (pretty) accurate integration over long time horizons, and maintain a dynamically varying estimate of the integrand, only using a pair of (well-tuned) weights, learned with biologically plausible learning rules.

Chapter Abstract

Ring attractor models for angular path integration have received strong experimental support. To function as integrators, head direction circuits require precisely tuned connectivity, but it is currently unknown how such tuning could be achieved. Here, we propose a network model in which a local, biologically plausible learning rule adjusts synaptic efficacies during development, guided by supervisory allothetic cues. Applied to the *Drosophila* head direction system, the model learns to path-integrate accurately and develops a connectivity strikingly similar to the one reported in experiments. The mature network is a quasi-continuous attractor and reproduces key experiments in which optogenetic stimulation controls the internal representation of heading, and where the network remaps to integrate with different gains in rodents. Our model predicts that path integration requires self-supervised learning during a developmental phase, and proposes a general framework to learn to path-integrate with gain-1 even in architectures that lack the physical topography of a ring.

2.1 Introduction

Spatial navigation is crucial for the survival of animals in the wild and has been studied in many model organisms (Tolman, 1948; O'Keefe and Nadel, 1978; Gallistel, 1993; Eichenbaum, 2017). To orient themselves in an environment, animals rely on external sensory cues (e.g., visual, tactile, or auditory), but such allothetic cues are often ambiguous or absent. In these cases, animals have been found to update internal representations of their current location based on idiothetic cues, a process that is termed path integration (PI, Darwin, 1873; Mittelstaedt and Mittelstaedt, 1980; McNaughton et al., 1996; Etienne, Maurer, and Séguinot, 1996; Neuser et al., 2008; Burak and Fiete, 2009). The head direction (HD) system partakes in PI by performing one of the computations required: estimating the current HD by integrating angular velocities; namely angular integration. Furthermore, head direction cells in rodents and flies provide an internal representation of orientation that can persist in darkness (Ranck, 1984; Mizumori and Williams, 1993; Seelig and Jayaraman, 2015).

In rodents, the internal representation of heading takes the form of a localized "bump" of activity in the high-dimensional neural manifold of HD cells (Chaudhuri et al., 2019). It has been proposed that such a localized activity bump could be sustained by a ring attractor network with local excitatory connections (Skaggs et al., 1995; Redish, Elga, and Touretzky, 1996; Hahnloser, 2003; Samsonovich and McNaughton, 1997; Song and Wang, 2005; Stringer et al., 2002; Xie, Hahnloser, and Seung, 2002), resembling reverberation mechanisms proposed for working memory (Wang, 2001). Ring attractor networks used to model HD cells fall in the theoretical framework of continuous attractor networks (CANs, Amari, 1977; Ben-Yishai, Bar-Or, and Sompolinsky, 1995; Seung, 1996) . In this setting, HD cells can update the heading representation in darkness by smoothly moving the bump around the ring obeying idiothetic angular-velocity cues.

Interestingly, a physical ring-like attractor network of HD cells was observed in the *Drosophila* central complex (CX, Seelig and Jayaraman, 2015; Green et al., 2017; Green et al., 2019; Franconville, Beron, and Jayaraman, 2018; Kim et al., 2019; Fisher et al., 2019; Turner-Evans et al., 2020). Notably, in *Drosophila* (from here on simply referred to as "fly"), HD cells (named E-PG neurons, also referred to as "compass" neurons) are physically arranged in a ring, and an activity bump is readily observable from a small number of cells (Seelig and Jayaraman, 2015). Moreover, as predicted by some computational models (Skaggs et al., 1995; Samsonovich and

McNaughton, 1997; Stringer et al., 2002; Song and Wang, 2005), the fly HD system also includes cells (named P-EN1 neurons) that are conjunctively tuned to head direction and head angular velocity. We refer to these neurons as head rotation (HR) cells because of their putative role in shifting the HD bump across the network according to the head's angular velocity (Turner-Evans et al., 2017; Turner-Evans et al., 2020).

A model for PI needs to both sustain a bump of activity and move it with the right speed and direction around the ring. The latter presents a great challenge, since the bump has to be "pushed" for the right amount starting from any location and for all angular velocities. Therefore, ring attractor models that act as path integrators require that synaptic connections are precisely tuned (Hahnloser, 2003). If the circuit was completely hardwired, the amount of information that an organism would need to genetically encode connection strenghts would be exceedingly high. Additionally, it would be unclear how these networks could cope with variable sensory experiences. In fact, remarkable experimental studies in rodents have shown that when animals are placed in an augmented reality environment where visual and self-motion information can be manipulated independently, PI capabilities adapt accordingly (Jayakumar et al., 2019). These findings suggest that PI networks are able to self-organize and to constantly recalibrate. Notably, in mature flies there is no evidence for such plasticity (Seelig and Jayaraman, 2015) — however, the presence of plasticity has not been tested in young animals.

Here, we propose that a simple local learning rule could support the emergence of a PI circuit during development and its re-calibration once the circuit has formed. Specifically, we suggest that accurate PI is achieved by associating allothetic and idiothetic inputs at the cellular level. When available, the allothetic sensory input (here chosen to be visual) acts as a "teacher" to guide learning. The learning rule is an example of self-supervised multimodal learning, where one sense acts as a teaching signal for the other and the need for an external teacher is obviated. It exploits the relation between the allothetic heading of the animal (given by the visual input) and the idiothetic self-motion cues (which are always available), to learn how to integrate the latter.

The learning rule is inspired by previous experimental and computational work on mammalian cortical pyramidal neurons, which are believed to associate inputs to different compartments through an in-built cellular mechanism (Larkum, 2013; Urbanczik and Senn, 2014; Brea et al., 2016). In fact, it was shown that in layer 5

pyramidal cells internal and external information about the world arrive at distinct anatomical locations, and active dendritic gating controls learning between the two (Doron et al., 2020). In a similar fashion, we propose that learning PI in the HD system occurs by associating inputs at opposite poles of compartmentalized HD neurons, which we call "associative neurons" (Urbanczik and Senn, 2014; Brea et al., 2016). Therefore, to accomplish PI the learning rule relies on structural inductive biases in terms of the morphology and arborization of HD cells.

In summary, here we show for the first time how a biologically plausible synaptic plasticity rule enables to learn and maintain the complex circuitry required for PI. We apply our framework to the fly HD system because it is well characterized; yet our model setting is general and can be used to learn PI in other animal models once more details about the HD circuit there are known (Abbott et al., 2020). We find that the learned network is a ring attractor with a connectivity that is strikingly similar to the one found in the fly CX (Turner-Evans et al., 2020) and that it can accurately path-integrate in darkness for the entire range of angular velocities that the fly displays. Crucially, the learned network accounts for several key findings in the experimental literature, and it generates predictions, including the presence of plasticity in young animals, that could be tested experimentally.

2.2 Methods

Model setup

The gross model architecture closely resembles the one found in the fly CX (fig. 2.1A). It comprises HD cells organized in a ring, and HR cells organized in two wings. One wing is responsible for leftward and the other for rightward movement of the internal heading representation. HD cells receive visual input from the so-called "ring" neurons; this input takes the form of a disinhibitory bump centered at the current HD (fig. 2.1B, Omoto et al., 2017; Fisher et al., 2019). The location of this visual bump in the network is controlled by the current head direction. We simulate head movements by sampling head-turning velocities from an Ornstein-Uhlenbeck process (Methods), and we provide the corresponding velocity input to the HR cells (fig. 2.1C). HR cells provide direct input to HD cells, and HR cells also receive input from HD cells (fig. 2.1A). Both HR and HD cells receive global inhibition, which is in line with a putative "local" model of HD network organization (Kim et al., 2017). The connections from HR to HD cells (W^{HR}) and the recurrent connections among HD cells (W^{rec}) are assumed to be plastic. The goal of learning is to tune these plastic connections so that the network can achieve PI in the absence of visual input.

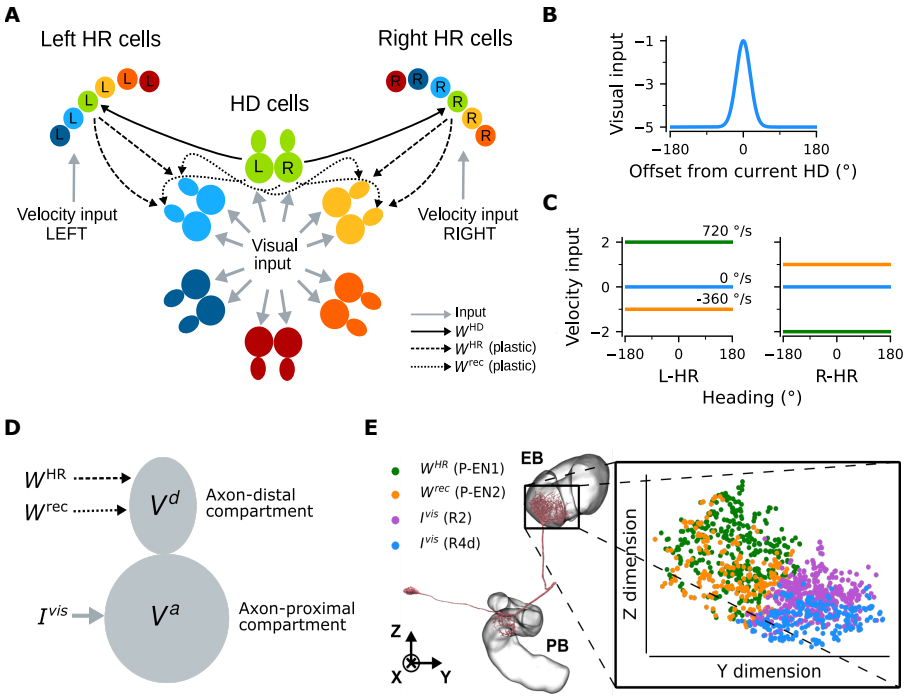

Figure 2.1: **Path-integrating network architecture.** (A) The ring of HD cells projects to two wings of HR cells, a leftward (Left HR cells, abbreviated as L-HR) and a rightward (Right HR cells, or R-HR), so that each wing receives selective connections only from a specific HD cell (L: left, R: right) for every head direction. For illustration purposes, the network is scaled-down by a factor of 5 compared to the cell numbers $N^{HR} = N^{HD} = 60$ in the model. The schema shows the outgoing connections (W^{HD} and W^{rec}) only from the green HD neurons and the incoming connections (W^{HR} and W^{rec}) only to the light blue and yellow HD neurons. Furthermore, the visual input to HD cells and the velocity inputs to HR cells are indicated. (B) Visual input to the ring of HD cells as a function of radial distance from the current head direction (see eq. (2.5)). (C) Angular-velocity input to the wings of HR cells for three angular velocities: 720 (green), 0 (blue), and −360 (orange) deg/s (see eq. (2.10)). (D) The associative neuron: V^a and V^d denote the voltage in the axon-proximal (i.e., closer to the axon initial segment) and axon-distal (i.e., further away from the axon initial segment) compartment, respectively. Arrows indicate the inputs to the compartments, as in (A), and I^{vis} is the visual input current. (E) Left: skeleton plot of an example HD (E-PG) neuron (Neuron ID = 416642425) created using neuPrint (Clements et al., 2020); the ellipsoid body (EB) and protocerebral bridge (PB) are overlayed. Right: zoomed in area in the EB indicated by the box, showing postsynaptic locations in the EB for this E-PG neuron; for details, see Methods. The neuron receives recurrent and HR input (green and orange dots, corresponding to inputs from P-EN1 and P-EN2 cells, respectively) and visual input (purple and blue dots, corresponding to inputs from visually responsive R2 and R4d cells, respectively) in distinct spatial locations.

The unit that controls plasticity in our network is an ”associative neuron.” It is inspired by pyramidal neurons of the mammalian cortex whose dendrites act, via backpropagating action potentials, as coincidence detectors for signals arriving from different layers of the cortex and targeting different compartments of the neuron (Larkum, Zhu, and Sakmann, 1999). Paired with synaptic plasticity, coincidence detection can lead to long-lasting associations between these signals (Larkum, 2013). To map the morphology of a cortical pyramidal cell to the one of a HD cell in the fly, we first point out that all relevant inputs arrive at the dendrites of HD cells within the ellipsoid body (EB) of the fly (Xu et al., 2020); moreover, the soma itself is externalized in the fly brain, and it is unlikely to contribute considerably to computations (Nathan W. Gouwens, 2009; Tuthill, 2009). We thus link the dendrites of the pyramidal associative neuron to the axon-distal dendritic compartment of the associative HD neuron in the fly, and we link the soma of the pyramidal associative neuron to the axon-proximal dendritic compartment of the associative HD neuron in the fly. Furthermore, we assume that the axon-proximal compartment is electrotonically closer to the axon initial segment, and therefore, similarly to the somatic compartment in pyramidal neurons, inputs there can more readily initiate action potentials. Note that our model does not require *active* backpropagation of action potentials — *passive* spread of voltage to the axon-distal compartment would be sufficient (for details, see Methods and Discussion). We also assume that associative HD cells receive visual input (I^{vis}) in the axon-proximal compartment, and both recurrent input (W^{rec}) and HR input (W^{HR}) in the axon-distal compartment; accordingly, we model HD neurons as two-compartment units (fig. 2.1D). The associative neuron can learn the synaptic weights of the incoming connections in the axon-distal compartment, therefore, as mentioned, we let W^{rec} and W^{HR} be plastic.

We find that the assumption of spatial segregation of postsynapses of HD cells is consistent with our analysis of EM data from the fly (Xu et al., 2020). For an example HD (E-PG) neuron, fig. 2.1E depicts that head rotation and recurrent inputs (mediated by P-EN1 and P-EN2 cells, respectively (Turner-Evans et al., 2020)) contact the E-PG cell in locations within the EB that are distinct compared to those of visually responsive neurons R2 and R4d (Omoto et al., 2017; Fisher et al., 2019), as hypothesized. The same pattern was observed for a total of 16 E-PG neurons (one for each "wedge" of the EB) that we analyzed (fig. B.1A). To further support the assumption that visual inputs are separated from recurrent and HR-to-HD inputs, we perform binary classification between the two classes, using SVMs (for details, see Methods). fig. B.1B shows that predicting class identity from spatial location alone

in held-out test data is excellent (test accuracy > 0.95 across neurons and model runs).

The connections from HD to HR cells (W^{HD}) are assumed to be fixed, and HR cells are modeled as single-compartment units. Projections are organized such that each wing neuron receives input from only one specific HD neuron for every HD (fig. 2.1A). This simple initial wiring makes HR cells conjunctively tuned to HR and HD, and we assume that it has already been formed, for example, during circuit assembly. We note that the conditions for 1-to-1 wiring and constant amplitude of the HD-to-HR connections can be relaxed, because the learning rule can balance asymmetries in the initial architecture (see Appendix B.4). In addition, the connections carrying the visual and angular velocity inputs are also assumed to be fixed. Although plasticity in the visual inputs has been shown to exist (Fisher et al., 2019; Kim et al., 2019), here we focus on how the path-integrating circuit itself originally self-organizes. Therefore, to simplify the setting and without loss of generality, we assume a fixed anchoring to environmental cues as the animal moves in the same environment (for details, see Discussion).

In our model, the visual input acts as a supervisory signal during learning (as in D'Albis and Kempter, 2020), which is used to change weights of synapses onto the axon-distal compartment of HD cells. We utilize the learning rule proposed by Urbanczik and Senn (2014) (for details, see Methods), which tunes the incoming synaptic connections in the axon-distal compartment in order to minimize the discrepancy between the firing rate of the neuron $f(V^a)$ (where V^a is the axon-proximal voltage, primarily controlled by the visual input) and the prediction of the firing rate by the axon-distal compartment in the absence of visual input, $f(pV^d)$ (where p is a constant and V^d is the axon-distal voltage, which depends on head rotation velocity). From now on, we refer to this discrepancy as "learning error," or simply "error" (eq. (2.18)) (in units of firing rate). The synaptic weight change $\Delta W_{pre,post}$ from a presynaptic (HD or HR) neuron to a postsynaptic HD neuron is then given by:

$$\Delta W_{pre,post} = \eta \left[f(V^a_{post}) - f(pV^d_{post}) \right] P_{pre} \tag{2.1}$$

where η is the constant learning rate and P_{pre} is the postsynaptic potential from the presynaptic neuron. When implementing this learning rule, we low-pass filter the prospective weight change $\Delta W_{pre,post}$ to ensure smoothness of learning.

Importantly, this learning rule is biologically plausible because the firing rate of an associative neuron $f(V^a)$ is locally available at every synapse in the axon-distal

compartment due to the (passive or active) backpropagation of axonal activity to the axon-distal dendrites. The other two signals that enter the learning rule are the voltage of the axon-distal compartment V^d and the postsynaptic potential P, which are also available locally at the synapse; for details, see Methods. Furthermore, recent behavioral experiments show that conditioning in *Drosophila* (Zhao et al., 2021) is not well explained by classical correlation-based plasticity, but it can be well accounted for by predictive synaptic plasticity. The latter is in line with the learning rule utilized here.

Detailed Network Architecture

In what follows, we describe our computational model for learning a ring attractor network that accomplishes accurate angular PI, in more detail. The model described here focuses on the HD system of the fly, however the proposed computational setup is general and could be applied to other systems. Unless otherwise stated, the simulation parameter values are the ones summarized in table 2.1. Simulation results for a given choice of parameters are very consistent across runs, hence most figures are generated from a single simulation run, unless otherwise stated.

Neuronal Model

We model a recurrent neural network comprising $N^{HD} = 60$ head-direction (HD) and $N^{HR} = 60$ head-rotation (HR) cells, which are close to the number of E-PG and P-EN1 cells in the fly central complex (CX), respectively (Turner-Evans et al., 2020; Xu et al., 2020). A scaled-down version of the network for $N^{HR} = N^{HD} = 12$ is shown in fig. 2.1A. The average spiking activity of HD and HR cells is modelled by firing-rate neurons. HD cells are organized in a ring and receive visual input, which encodes the angular position of the animal's head with respect to external landmarks. We use a discrete representation of angles and we model two HD cells for each head direction, as observed in the biological system (Turner-Evans et al., 2017). Therefore the network can represent head direction with an angular resolution $\Delta\phi = 12$ deg.

Motivated by the anatomy of the fly CX (Green et al., 2017; Turner-Evans et al., 2020), HR cells are divided in two populations (fig. 2.1A): a 'leftward' (L-HR) population (with increased velocity input when the head turns leftwards) and a 'rightward' (R-HR) population (with increased velocity input when the head turns rightwards). After learning, these two HR populations are responsible to move the HD bump in the anticlockwise and clockwise directions, respectively.

Table 2.1: Parameter values.

Parameter	Value	Explanation
N^{HD}	60	Number of head direction (HD) neurons
N^{HR}	60	Number of head rotation (HR) neurons
$\Delta\phi$	12 deg	Angular resolution of network
τ_s	65 ms	Synaptic time constant
I_{inh}^{HD}	-1	Global inhibition to HD neurons
τ_l	10 ms	Leak time constant of axon-distal compartment
C	1 ms	Capacitance of axon-proximal compartment
g_L	1	Leak conductance of axon-proximal compartment
g_D	2	Conductance from distal to proximal compartment
$\boldsymbol{I}_{exc}^{HD}$	4	Input to axon-proximal compartment in light
σ_n	0	Synaptic input noise level
M	4	Visual input amplitude
M_{stim}	16	Optogenetic stimulation amplitude
σ	0.15	Visual receptive field width
σ_{stim}	0.25	Optogenetic stimulation width
I_o^{vis}	-5	Visual input baseline
f_{max}	150 Hz	Maximum firing rate
β	2.5	Steepness of activation function
$x_{1/2}$	1	Input level for 50 % of the maximum firing rate
I_{inh}^{HR}	-1.5	Global inhibition to HR neurons
k	$1/360$ s/deg	Ratio of velocity input and head angular velocity
A_{active}	2	Input range for which f has not saturated
w^{HD}	$13.\overline{3}$ ms	Constant weight from HD to HR neurons
τ_δ	100 ms	Plasticity time constant
Δt	0.5 ms	Euler integration step size
τ_v	0.5 s	Time constant of velocity decay
σ_v	450 deg/$\sqrt{s}$	Standard deviation of angular velocity noise
η	0.05 1/s	Learning rate

Parameter values. Parameter values, in the order they appear in the Methods section. These values apply to all simulations, unless otherwise stated. Note that voltages, currents, and conductances are assumed unitless in the text; therefore capacitances have the same units as time constants.

The recurrent connections among HD cells and the connections from HR to HD cells are assumed to be plastic. On the contrary, connections from HD to HR cells are assumed fixed and determined as follows: for every head direction, one HD neuron projects to a cell in the L-HR population, and the other to a cell in the R-HR population. Because HD cells project to HR cells in a 1-to-1 manner, each HR neuron is simultaneously tuned to a particular head direction and a particular head

rotation direction. The synaptic strength of the HD-to-HR projections is the same for all projections (these restrictions on the HD-to-HR connections are relaxed in Appendix B.4). Finally, HR cells do not form recurrent connections.

We assume that each HD neuron is a rate-based associative neuron (fig. 2.1D), i.e., a two-compartmental neuron comprising an axon-proximal and an axon-distal dendritic compartment (Urbanczik and Senn, 2014; Brea et al., 2016). The two compartments model the dendrites of that neuron that are closer to or further away from the axon initial segment. Note that here the axon-proximal compartment replaces the somatic compartment in the original model by Urbanczik and Senn (2014). This is because the somata of fly neurons are typically electrotonically segregated from the rest of the cell and they are assumed to contribute little to computation (Nathan W. Gouwens, 2009; Tuthill, 2009). We also note that to fully capture the input/output transformations that HD neurons in the fly perform, more compartments than two might be needed (Xu et al., 2020). Finally, only HD cells are associative neurons, whereas HR cells are simple rate-based point neurons.

HD cells receive an input current $\boldsymbol{I}^d$ to the axon-distal dendrites, which obey

$$\tau_s \frac{\mathrm{d}\boldsymbol{I}^d}{\mathrm{d}t} = -\boldsymbol{I}^d + W^{rec}\boldsymbol{r}^{HD} + W^{HR}\boldsymbol{r}^{HR} + I_{inh}^{HD} + \sigma_n \boldsymbol{n}^d \tag{2.2}$$

where $\boldsymbol{I}^d$ is a vector of length N^{HD} with each entry corresponding to one HD cell. In eq. (2.2), τ_s is the synaptic time constant, W^{rec} is a $N^{HD} \times N^{HD}$ matrix of the recurrent synaptic weights among HD cells, W^{HR} is a $N^{HD} \times N^{HR}$ matrix of the synaptic weights from HR to HD cells, $\boldsymbol{r}^{HR}$ and $\boldsymbol{r}^{HD}$ are vectors of the firing rates of HR and HD cells, respectively, I_{inh}^{HD} is a constant inhibitory input common to all HD cells, and $\boldsymbol{n}^d$ is a random noise input to the axon-distal compartment. $\boldsymbol{n}^{\boldsymbol{d}}$ is drawn IID from $N(0, 1)$, and its variance is scaled by σ_n^2. Note that in the main text we set σ_n to zero, but we explore different values for this parameter in Appendix B.2. The constant current I_{inh}^{HD} is in line with a global-inhibition model with local recurrent connectivity, as opposed to having long-range inhibitory recurrent connectivity (Kim et al., 2017). The inhibitory current I_{inh}^{HD} suppresses HD bumps in general; however the exact strength of this inhibition is not important in our model.

Since several electrophysiological parameters of the fly neurons modeled here are unknown, we use dimensionless conductance values. Therefore, in eq. (2.2), which describes the dynamics of the axon-distal input of HD cells, currents (e.g., $\boldsymbol{I}^d$, I_{inh}^{HD}, and $\boldsymbol{n}^d$) are dimensionless. Membrane voltages are also chosen to be dimensionless, and because we measure firing rates in units of 1/s, all synaptic weights (e.g., W^{rec}

and W^{HR}) then have, strictly speaking, the unit 'seconds' (s), even though we mostly suppress this unit in the text. Importantly, all time constants (e.g., τ_s), which define the time scale of dynamics, are measured in units of time (in seconds).

Our model incorporates several time scales, whose interplay is not obvious. To facilitate understanding, we summarize the parameters that define the time scales in table B.1, and discuss their relation in Appendix B.5.

The axon-distal voltage $\boldsymbol{V}^d$ of HD cells is a low-pass filtered version of the input current $\boldsymbol{I}^d$, that is,

$$\tau_l \frac{\mathrm{d}\boldsymbol{V}^d}{\mathrm{d}t} = -\boldsymbol{V}^d + \boldsymbol{I}^d \tag{2.3}$$

where τ_l is the leak time constant of the axon-distal compartment. The voltage $\boldsymbol{V}^d$ and the current $\boldsymbol{I}^d$ have the same unit (both dimensionless), which means that the leak resistance of the axon-distal compartment is also dimensionless, and we assume that it is unity for simplicity. We choose values of τ_l and τ_s (for specific values, see table 2.1) so that their sum matches the phenomenological time constant of HD neurons (E-PG in the fly), while τ_s equals to the phenomenological time constant of HR neurons (P-EN1 in the fly, Turner-Evans et al., 2017). Note that $\boldsymbol{V}^d$ is the low-frequency component of the axon-distal voltage originating from postsynaptic potentials, i.e., excluding occasional high-frequency contributions from backpropagating action potentials.

The axon-proximal voltage $\boldsymbol{V}^{\boldsymbol{a}}$ of HD cells is then given by

$$C \frac{\mathrm{d}\boldsymbol{V}^{\boldsymbol{a}}}{\mathrm{d}t} = -g_L \boldsymbol{V}^{\boldsymbol{a}} - g_D(\boldsymbol{V}^{\boldsymbol{a}} - \boldsymbol{V}^d) + \boldsymbol{I}^{vis} + \boldsymbol{I}^{HD}_{exc} + \sigma_n \boldsymbol{n}^{\boldsymbol{a}} \tag{2.4}$$

where C is the capacitance of the membrane of the axon-proximal compartment, g_L is the leak conductance, g_D is the conductance of the coupling from axon-distal to axon-proximal dendrites, $\boldsymbol{I}^{vis}$ is a vector of visual input currents to the axon-proximal compartment of HD cells, $\boldsymbol{I}^{HD}_{exc}$ is an excitatory input to the axon-proximal compartment, and $\boldsymbol{n}^{\boldsymbol{a}}$ is a random noise vector injected to the axon-proximal compartment, drawn IID from $N(0, 1)$. The excitatory current $\boldsymbol{I}^{HD}_{exc}$ is assumed to be present only in light conditions. The values of C, g_L, and g_D in the fly HD (E-PG) neurons are unknown, thus we keep these parameters unitless, and set their values to the ones in Urbanczik and Senn (2014). Note that since conductances are dimensionless here, C is effectively a time constant.

Following Hahnloser (2003), the visual input to HD cell i is a localized bump of activity at angular location θ_i:

$$I_i^{vis}(t) = M \exp\left(-\frac{1}{2\sigma^2}\sin^2\left(\frac{\theta_i + \theta_0(t)}{2}\right)\right) + I_o^{vis} \tag{2.5}$$

where M scales the bump's amplitude, σ controls the width of the bump, θ_i is the preferred orientation of HD neuron i, $\theta_0(t)$ is the position of a visual landmark at time t in head-centered coordinates, and $I_o^{vis} < 0$ is a constant inhibitory current that acts as the baseline for the visual input. We choose M so that the visual input can induce a weak bump in the network at the beginning of learning, and we choose σ so that the resulting bump after learning is ~60 deg wide. Note that the bump in the mature network has a square shape (fig. B.3B); therefore we elect to make it slightly narrower than the average full width at half maximum of the experimentally observed bump (~80 deg; Seelig and Jayaraman, 2015; Kim et al., 2017; Turner-Evans et al., 2017). In addition, the current I_o^{vis} is negative enough to make the visual input purely inhibitory, as reported (Fisher et al., 2019). The visual input is more inhibitory in the surround to suppress activity outside of the HD receptive field. Therefore the mechanism in which the visual input acts on the HD neurons is disinhibition.

The firing rate of HD cells, which is set by the voltage in the axon-proximal compartment, is given by

$$\boldsymbol{r}^{HD} = f(\boldsymbol{V}^{\boldsymbol{a}}) \tag{2.6}$$

where

$$f(x) = \frac{f_{max}}{1 + \exp(-\beta(x - x_{1/2}))} \tag{2.7}$$

is a sigmoidal activation function applied element-wise to the vector $\boldsymbol{V}^{\boldsymbol{a}}$. The variable f_{max} sets the maximum firing rate of the neuron, β is the slope of the activation function, and $x_{1/2}$ is the input level at which half of the maximum firing rate is attained. The value of f_{max} is arbitrary, while β is chosen such that the activation function has sufficient dynamic range, and $x_{1/2}$ is chosen such that for small negative inputs the activation function is non-zero.

We note that the saturation of the activation function f in eq. (2.7) is an essential feature of our model, especially for the convergence of the plasticity rule in eq. (2.12); see also the section "Synaptic Plasticity Rule." Even though, to the best of our knowledge, it is currently not known whether E-PG neurons actually reach saturation, other *Drosophila* neurons are known to reach saturation with increasing inputs,

instead of some sort of depolarization block (Wilson, 2013; Brandão, Silies, and Martelli, 2021). Saturation with increasing inputs may be due to, for instance, short-term synaptic depression: beyond a certain frequency of incoming action potentials, the synaptic input current is almost independent of that frequency (Tsodyks and Markram, 1997; Tsodyks, Pawelzik, and Markram, 1998).

The firing rates of the HR cells are given by

$$\boldsymbol{r}^{HR} = f\left(W^{HD}\boldsymbol{r}^{HD}_{LP} + \boldsymbol{I}^{vel} + I^{HR}_{inh} + \sigma_n \boldsymbol{n}^{HR}\right) \tag{2.8}$$

where $\boldsymbol{r}^{HR}$ is the vector of length N^{HR} of firing rates of HR cells, the $N^{HR} \times N^{HD}$ matrix W^{HD} encodes the fixed connections from the HD to the HR cells, $\boldsymbol{r}^{HD}_{LP}$ is a low-pass filtered version of the firing rate of the HD cells where the filter accounts for delays due to synaptic transmission in the incoming synapses from HD cells, $\boldsymbol{I}^{vel}$ is the angular velocity input, I^{HR}_{inh} is a constant inhibitory input common to all HR cells, and $\boldsymbol{n}^{HR}$ is a random noise input to the HR cells drawn IID from $N(0, 1)$. We set I^{HR}_{inh} to a value that still allows sufficient activity in the HR cell bump, even when the animal does not move. The low-pass filtered firing-rate vector $\boldsymbol{r}^{HD}_{LP}$ is given by

$$\tau_s \frac{\mathrm{d}\boldsymbol{r}^{HD}_{LP}}{\mathrm{d}t} = -\boldsymbol{r}^{HD}_{LP} + \boldsymbol{r}^{HD}\,, \tag{2.9}$$

and the angular-velocity input to HR neuron i is given by

$$I^{vel}_i(t) = q\, k\, v(t) \quad \text{with} \quad q = \begin{cases} -1 & \text{for } i \le N^{HR}/2 \\ 1 & \text{for } i > N^{HR}/2 \end{cases} \tag{2.10}$$

where k is the proportionality constant between head angular velocity and velocity input to the network, $v(t)$ is the head angular velocity at time t in units of deg/s, and the factor q is chosen such that the left (right) half of the HR cells are primarily active during leftward (rightward) head rotation. Note that the same τ_s is in both eq. (2.2) and eq. (2.9). Finally, as mentioned earlier, the matrix W^{HD} encodes the hardwired 1-to-1 HD-to-HR connections, i.e., $W^{HD}_{ij} = w^{HD}$ if HD neuron j projects to HR neuron i, and $W^{HD}_{ij} = 0$ otherwise. Specifically, for j odd, HD neuron j projects to L-HR neuron $i = \frac{j+1}{2}$, whereas for j even, HD neuron j projects to R-HR neuron $i = 30 + \frac{j}{2}$. The synaptic strength w^{HD} is chosen such that the range of the firing rates of the HD cells is mapped to the entire range of firing rates of the HR cells. Specifically, we set $w^{HD} = \frac{A_{active}}{f_{max}}$, where A_{active} is the range of inputs for which f has not saturated, i.e., the input values for which f remains between about 7% and 93% of its maximum firing rate f_{max} (see eq. (2.7)). Finally, the

proportionality constant k is set so that the firing rate of HR neurons does not reach saturation for the range of velocities relevant for the fly (approx. $[-500, 500]$ deg/s), given all other inputs they receive.

Synaptic Plasticity Rule

In our network, the associative HD neurons receive direct visual input in the axon-proximal compartment and indirect angular velocity input in the axon-distal compartment through the HR-to-HD connections (fig. 2.1D). We hypothesize that the visual input acts as a supervisory signal that controls the axon-proximal voltage $\boldsymbol{V^a}$ directly, and the latter initiates spikes. Therefore, the goal of learning is for the axon-distal voltage $\boldsymbol{V}^d$ to predict the axon-proximal voltage by changing the synaptic weights W^{rec} and W^{HR}. This change is achieved by minimizing the difference between the firing rate $f(\boldsymbol{V^a})$ in the presence of visual input and the axon-distal prediction $f(\boldsymbol{V}^{ss})$ of the firing rate in the absence of visual input. In the latter case and at steady-state, the voltage V_i^{ss} for HD neuron i is an attenuated version of the axon-distal voltage,

$$V_i^{ss} = \frac{g_D}{g_D + g_L} V_i^d \,, \tag{2.11}$$

with conductance g_D of the coupling from the axon-distal to axon-proximal dendrites and leak conductance g_L of the axon-proximal compartment, as explained in eq. (2.4), and $p = \frac{g_D}{g_D+g_L}$ in eq. (2.1). Therefore, following Urbanczik and Senn (2014), we define the plasticity-induction variable PI_{ij} for the connection between presynaptic neuron j and postsynaptic neuron i as

$$PI_{ij} = \left[f(V_i^a) - f(V_i^{ss}) \right] P_j \tag{2.12}$$

where P_j is the postsynaptic potential of neuron j, which is a low-pass filtered version of the presynaptic firing rate r_j. That is,

$$P_j(t) = H(t) * r_j(t) \tag{2.13}$$

where $*$ denotes convolution. The transfer function

$$H(t) = \frac{1}{\tau_l - \tau_s} \left[\exp\left(-\frac{t}{\tau_l}\right) - \exp\left(-\frac{t}{\tau_s}\right) \right] u(t) \tag{2.14}$$

is derived from the filtering dynamics in eq. (2.2) and eq. (2.3) and accounts for the delays introduced by the synaptic time constant τ_s and the leak time constant τ_l. In eq. (2.14), $u(t)$ denotes the Heaviside step function, i.e., $u(t) = 1$ for $t > 0$ and

$u(t) = 0$ otherwise. The plasticity-induction variable is then low-pass filtered to account for slow learning dynamics,

$$\tau_\delta \frac{\mathrm{d}\delta_{ij}}{\mathrm{d}t} = -\delta_{ij} + PI_{ij} \,, \tag{2.15}$$

and the final weight change is given by

$$\frac{\mathrm{d}W_{ij}}{\mathrm{d}t} = \eta\delta_{ij} \tag{2.16}$$

where η is the learning rate and W_{ij} is the connection weight from presynaptic neuron j to postsynaptic neuron i. Note that the synaptic weight W_{ij} is an element of either the matrix W^{rec} or the matrix W^{HR} depending on whether the presynaptic neuron j is an HD or an HR neuron, respectively. The value of the plasticity time constant τ_δ is not known, therefore we adopt the value suggested by Urbanczik and Senn (2014).

Equation (2.12) is a "delta-like" rule that can be interpreted as an extension of the Hebbian rule; compared to a generic Hebbian rule, we have replaced the postsynaptic firing rate $f(V_i^a)$ by the difference between $f(V_i^a)$ and the predicted firing rate $f(V_i^{ss})$ of the axon-distal compartment of the postsynaptic neuron. This difference drives plasticity in the model. We note that $f(V_i^a)$ is a continuous approximation of the spike train of the postsynaptic neuron, which could be available at the axon-distal compartment via back-propagating action potentials (Larkum, 2013). Furthermore, the axon-distal voltage V_i^d and postsynaptic potentials are by definition available at the synapses arriving at the axon-distal compartment. Note that even though $f(V_i^{ss})$ is the firing rate in the absence of visual input, it can still be computed at the axon-distal compartment when the visual input is available; V_i^d is the local voltage and therefore only a constant multiplicative factor (eq. (2.11)) and the static nonlinearity f need to be computed to obtain $f(V_i^{ss})$. Therefore, the learning rule is biologically plausible because all information is locally available at the synapse.

The learning rule used here differs from the one in the original work of Urbanczik and Senn (2014) because we utilize a rate-based version instead of the original spike-based version. Even though spike trains can introduce Poisson noise to $f(V_i^a)$, Urbanczik and Senn (2014) show that once learning has converged, asymmetries in the weights due the spiking noise are on average canceled out.

Another difference in our learning setup is that, unlike in Urbanczik and Senn (2014), the input to the axon-proximal compartment does not reach zero in equilibrium

(see, e.g., fig. 2.3D, and Appendix B.6). Therefore, an activation function with a saturating non-linearity, as in eq. (2.7), is crucial for convergence, which could not be achieved with a less biologically plausible threshold-linear activation function. This lack of strict convergence in our setup is responsible for the square form of the bump (fig. B.3B and Appendix B.6).

Training Protocol

We train the network with synthetically generated angular velocities, simulating head turns of the animal. W^{rec} and W^{HR} are both initialized with random connectivity drawn from a normal distribution with mean 0 and standard deviation $1/\sqrt{N^{HD}}$, as common practise in the modeling literature. In further simulations with various other initial conditions (e.g., in simulations with gain changes in fig. 2.4 or in simulations in which we randomly shuffled weights after learning, not shown), we confirmed that the final PI performance is virtually independent of the initial distribution of weights W^{rec} and W^{HR}.

The network dynamics are updated in discrete time steps Δt using forward Euler integration. The entrained angular velocities cover the range of angular velocities exhibited by the fly, which are at maximum $\sim$ 500 deg/s during walking or flying (Geurten et al., 2014; Stowers et al., 2017). The angular velocity $v(t)$ is modeled as an Ornstein-Uhlenbeck process given by

$$v(t + \Delta t) = (1 - \alpha)\, v(t) + \sigma_v \sqrt{\Delta t}\, n(t) \tag{2.17}$$

where $\alpha = \Delta t/\tau_v$ and τ_v is the time constant with which $v(t)$ decays to zero, $n(t)$ is noise drawn from a normal distribution with mean 0 and standard deviation 1 at each time step, and σ_v scales the noise strength.

We pick σ_v and τ_v so that the resulting angular velocity distribution and its time course are similar to what has been reported in flies during walking or flying (Geurten et al., 2014; Stowers et al., 2017). Finally, note that we train the network for angular velocities a little larger than what flies typically display (up to ±720 deg/s).

Quantification of the Mean Learning Error

In eq. (2.12) we have used the learning error

$$E_i = f(V_i^a) - f(V_i^{ss}) \tag{2.18}$$

which controls learning in every associative HD neuron i. To quantify the mean learning error $err(t)$ in the whole network at time t, we average E_i across all HD

neurons and across a small time interval $[t, t + t_w]$, that is,

$$err(t) = \frac{1}{t_w N^{HD}} \sum_{i=1}^{N^{HD}} \int_{t}^{t+t_w} |E_i(\tau)| \, \mathrm{d}\tau \tag{2.19}$$

with $t_w = 10\,\mathrm{s}$. In fig. 2.3D, we plot this mean error at every 1 % of the simulation, for 12 simulations, and averaged across the ensemble of the simulations. Note that individual simulations occasionally display "spikes" in the error. Large errors occur if the network happens to be driven by very high velocities that the network does not learn very well because they are rare; larger errors also occur for very small velocities, i.e., when the velocity input is not strong enough to overcome the local attractor dynamics, as seen, e.g., in fig. 2.2C. On average, though, we can clearly see that the mean learning error decreases with increasing time and settles to a small value (e.g., fig. 2.3D and fig. 2.4A).

Population Vector Average

To decode from the activity of HD neurons an average HD encoded by the network, we use the population vector average (PVA). We thus first convert the tuning direction θ_i of each HD neuron i to the corresponding complex number $e^{j\theta_i}$ on the unitary circle, where j is the imaginary unit. This complex number is multiplied by the firing rate r_i^{HD} of HD neuron i, and then averaged across neurons to yield the PVA

$$r_{av} = \frac{1}{N^{HD}} \sum_{i=1}^{N^{HD}} r_i^{HD} e^{j\theta_i} \, . \tag{2.20}$$

The PVA is a vector in the 2-D complex plane and points to the center of mass of activity in the HD network. Finally, we take the angle θ of the PVA as a measure for the current heading direction represented by the network.

Diffusion Coefficient

To quantify the variability of heading direction in the trained networks, we define the diffusion coefficient D as:

$$D = \frac{\left\langle \Delta\theta^2 \right\rangle - \langle \Delta\theta \rangle^2}{t_{sim}} \tag{2.21}$$

where $\Delta\theta$ is the change in heading direction in a time interval t_{sim}. Therefore, D is given by the variance of the distribution of displacements in a given time interval, divided by the time interval.

In the main text, we estimate D during PI, i.e., with velocity inputs only. In this setting, D is the rate at which the variance of the PI errors increases (see, e.g., fig. 2.2B). Deviations from gain-1 PI contribute to this estimate; hence, to single out the effects of noise during training on the stability of the learned attractor in Appendix B.2, we also estimate D in the presence of test noise when no inputs are received at all.

Fly Connectome Analysis

Our model assumes the segregation of visual inputs to HD (E-PG) cells from head rotation and recurrent inputs to the same cells. To test this hypothesis, we leverage on the fly hemibrain connectome (Xu et al., 2020; Clements et al., 2020). First, we randomly choose one E-PG neuron per wedge of the EB, for a total of 16 E-PG neurons. We reasoned this sample would be sufficient because the way E-PG neurons in the same wedge are innervated is expected to be similar. We then find all incoming connections to these neurons from visually responsive ring neurons R2 and R4d (Omoto et al., 2017; Fisher et al., 2019). These are the connections that arrive at the axon-proximal compartment in our model. We then find all incoming connections from P-EN1 cells, which correspond to the HR neurons, and from P-EN2 cells, which are involved in a recurrent excitatory loop from E-PG to P-EG to P-EN2 and back to E-PG (Turner-Evans et al., 2020). These are the connections that arrive at the axon-distal compartment in our model.

To further support the assumption that visual inputs are separated from recurrent and HR-to-HD inputs in the *Drosophila* EB, we perform binary classification between the two classes (R2 and R4d vs. P-EN1 and P-EN2). We use SVMs with Gaussian kernel, and perform nested 5-fold cross validation, for a total of 30 model runs for every neuron tested (fig. B.1).

Quantification of PI performance

To quantify PI performance of the network and compare to fly performance, we use the measure defined by Seelig and Jayaraman (2015) and estimate the correlation coefficient between the unwrapped PVA and true heading in darkness. We estimate the correlation in 140-second long trials and report the point estimate and 95% confidence intervals (Student's t-test, $N = 100$).

2.3 Mature network can path-integrate in darkness

Figure 2.2A shows an example of the performance of a trained network, for the light condition (i.e., when visual input is available; yellow overbars) and for PI in darkness (purple overbars); the performance is quantified by the PI error (in units of degrees) over time. PI error refers to the accumulated difference between the internal representation of heading and the true heading, and it is different from the learning error introduced previously.

A unique bump of activity is clearly present at all times in the HD network (fig. 2.2A, top), in both light and darkness conditions, and this bump moves smoothly across the network for a variable angular velocity (fig. 2.2A, bottom). The position of the bump is defined as the population vector average (PVA) of the neural activity in the HD network. The HD bump also leads to the emergence of bumps in the HR network, separately for L-HR and R-HR cells (fig. 2.2A, second and third panel from top). In light conditions (0–20 s in fig. 2.2A), the PVA closely tracks the head direction of the animal in HD, L-HR, and R-HR cells alike, which is expected because the visual input guides the network activity. Importantly, however, in darkness (20–50 s in fig. 2.2A), the self-motion input alone is enough to track the animal's heading, leading to a small PI error between the internal representation of heading and the ground truth. This error is corrected after the visual input reappears (at 50 s in fig. 2.2A). Such PI errors in darkness are qualitatively consistent with data reported in the experimental literature (Seelig and Jayaraman, 2015). The correction of the PI error also reproduces *in silico* the experimental finding that the visual input (whenever available) exerts stronger control on the bump location than the self-motion input (Seelig and Jayaraman, 2015), which suggests that even the mature network does not rely on PI when visual cues are available.

To quantify the accuracy of PI in our model, we draw 1000 trials, each 60 s long, for constant synaptic weights and in the absence of visual input. We also limit the angular velocities in these trials to retain only velocities that flies realistically display (see dashed green lines in fig. 2.2C and Methods). We then plot the distribution of PI errors every 10 s (fig. 2.2B). We find that average absolute PI errors (widths of distributions) increase with time in darkness, but most of the PI errors at 60 s are within 60 deg of the true heading. This vastly exceeds the PI performance of flies (Seelig and Jayaraman, 2015). In flies, the correlation between the PVA estimate and the true heading in darkness varied widely across animals in the range $[0.3, 0.95]$ (Seelig and Jayaraman, 2015), whereas for the model it is close to 1.

However, it should be noted that the model here corresponds to an ideal scenario that serves as a proof of principle. We will later incorporate irregularities owing to biological factors (asymmetry in the weights, biological noise) that bring the network's performance closer to the fly's behavior.

To further assess the network's ability to integrate different angular velocities, we simulate the system both with and without visual input in 5-s intervals during which the angular velocity is constant. We then compute the average movement velocity of the bump across the network, i.e., the neural velocity, and compare it to the real velocity provided as input. Figure 2.2C shows that the network achieves a PI gain (defined as the ratio between neural and real velocity) close to 1 both with and without supervisory visual input, meaning that the neural velocity matches very well the angular velocity of the animal, for all angular velocities that are observed in experiments ($|v| < 500$ deg/s for walking and flying) (Geurten et al., 2014; Stowers et al., 2017). Although expected in light conditions, the fact that gain 1 is achieved in darkness shows that the network predicts the missing visual input from the velocity input, i.e., the network path integrates accurately. Note that PI is impaired in our model for very small angular velocities (fig. 2.2C, flat purple line for $|v| < 30$ deg/s), similarly to previous hand-tuned theoretical models (Turner-Evans et al., 2017). This is a direct consequence of the fact that maintaining a stable activity bump and moving it across the network at very small angular velocities are competing goals. Crucially, it has been reported that such an impairment of PI for small angular velocities exists in flies (Seelig and Jayaraman, 2015). Note that if we increase the number of HD neurons from 60 (~50 were reported in the fly by Turner-Evans et al. (2020) and Xu et al. (2020)) to 120 or 240, this flat region is no longer observed (data not shown).

2.4 The network is a quasi-continuous attractor

A continuous attractor network (CAN) should be able to maintain a localised bump of activity in virtually a continuum of locations around the ring of HD cells. To prove that the learned network approximates this property, we seek to reproduce *in silico* experimental findings in Kim et al. (2017). There it was shown that local optogenetic stimulation of HD cells in the ring can cause the activity bump to jump to a new position and persist in that location — supported by internal dynamics alone.

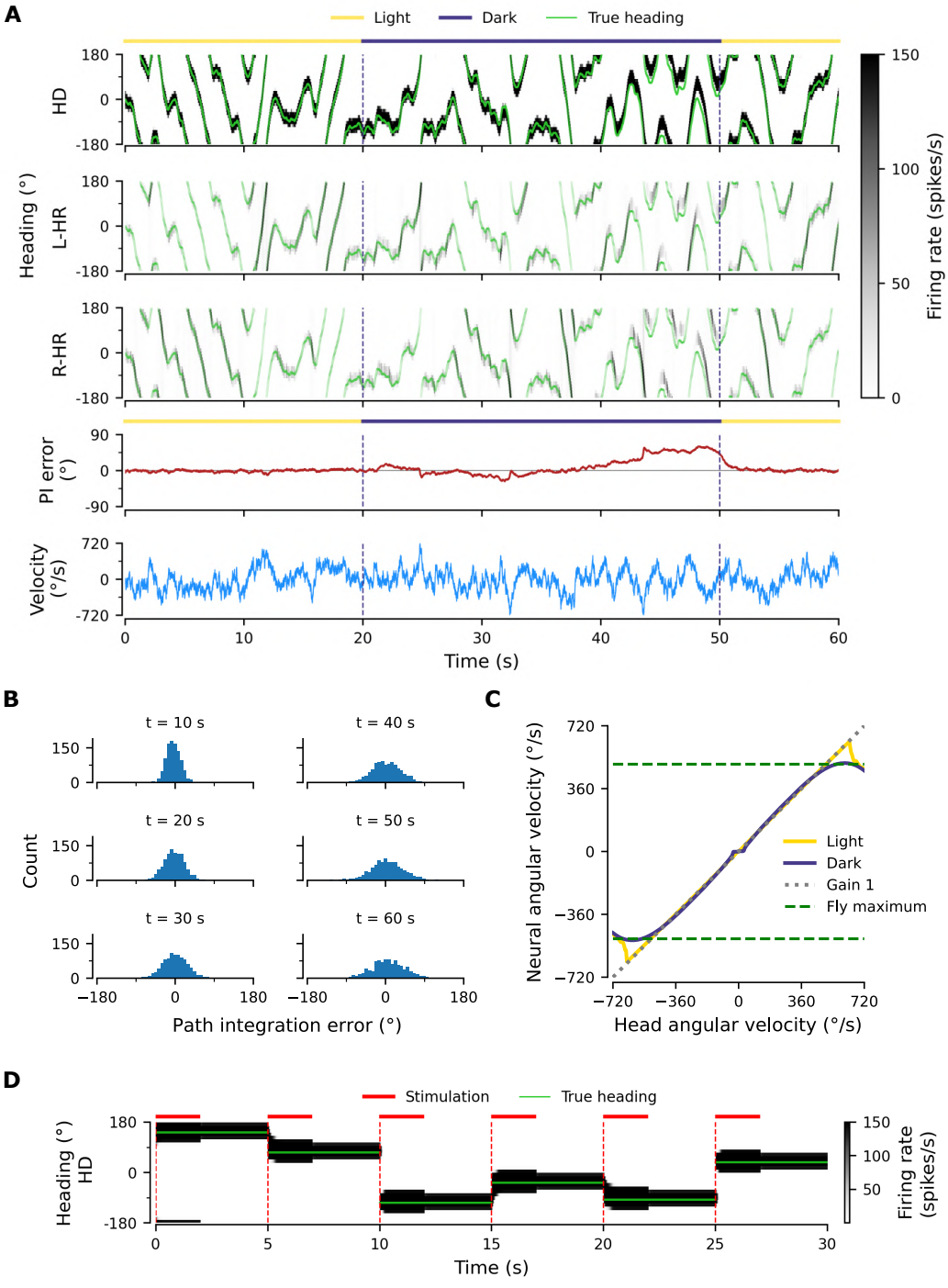
A
Light
Dark
True heading
HD
L-HR
R-HR
Heading (°)
PI error (°)
Velocity (°/s)
Firing rate (spikes/s)
Time (s)
B
t = 10 s
t = 20 s
t = 30 s
t = 40 s
t = 50 s
t = 60 s
Count
Path integration error (°)
C
Neural angular velocity (°/s)
Head angular velocity (°/s)
Light
Dark
Gain 1
Fly maximum
D
Stimulation
True heading
Heading (°)
HD
Firing rate (spikes/s)
Time (s)

Figure 2.2: **Path integration (PI) performance of the network.** (A) Example activity profiles of HD, L-HR, and R-HR neurons (firing rates gray-scale coded). Activities are visually guided (yellow overbars) or are the result of PI in the absence of visual input (purple overbar). The ability of the circuit to follow the true heading is slightly degraded during PI in darkness. The PI error, i.e., the difference between the PVA and the true heading of the animal as well as the instantaneous head angular velocity are plotted separately. (B) Temporal evolution of the distribution of PI errors in darkness, for 1000 simulations. The distribution gets wider with time, akin to a diffusion process. We estimate the diffusion coefficient to be $D = 24.5\ \mathrm{deg}^2/\mathrm{s}$ (see "Diffusion Coefficient" in Methods). Note that, unless otherwise stated, for this type of plot we limit the range of angular velocities to those normally exhibited by the fly, i.e., $|v| < 500$ deg/s. (C) Relation between head angular velocity and neural angular velocity, i.e., the speed with which the bump moves in the network. There is almost perfect (gain 1) PI in darkness for head angular velocities within the range of maximum angular velocities that are displayed by the fly (dashed green horizontal lines; see Methods). (D) Example of consecutive stimulations in randomly permeated HD locations, simulating optogenetic stimulation experiments in Kim et al. (2017). Red overbars indicate when the network is stimulated with stronger than normal visual-like input, at the location indicated by the animal's true heading (light green line), while red dashed vertical lines indicate the onset of the stimulation. The network is then left in the dark. Our simulations show that the bump remains at the stimulated positions.

To reproduce the experiments by Kim et al. (2017), we simulate optogenetic stimulation of HD cells in our network as visual input of increased strength and extent (for details, see Methods). We find that the strength and extent of the stimulation needs to be increased relative to that of the visual input; only in this case, a bump at some other location in the network can be suppressed, and a new bump emerges at the stimulated location. The stimuli are assumed to appear instantaneously at random locations, but we restrict our set of stimulation locations to the discrete angles represented by the finite number of HD neurons. Furthermore, the velocity input is set to zero for the entire simulation, signaling lack of head movement.

Figure 2.2D shows network activity in response to several stimuli, when the stimulation location changes abruptly every 5 s. During stimulation (2 s long, red overbars), the bump is larger than normal due to the use of a stronger than usual visual-like input to mimic optogenetic stimulation. The way in which the network responds to a stimulation depends on how far away from the "current" location it is stimulated: for shorter distances, the bump activity shifts to the new location, as evidenced by the transient dynamics at the edges of the bump resembling a decay from an initial

to a new location (see fig. 2.2D at {5,15,20} s). However, for larger phase shifts $\Delta\theta$ the bump first emerges in the new location and subsequently disappears at the initial location, a mechanism akin to a "jump" (fig. 2.2D, all other transitions). Similar effects have been observed in the experimental literature (Seelig and Jayaraman, 2015; Kim et al., 2017). The way the network responds to stimulation indicates that it operates in a CAN manner, and not as a winner-takes-all network where changes in bump location would always be instantaneous (Carpenter and Grossberg, 1987; Itti, Koch, and Niebur, 1998; Wang, 2002). That is to say, the network operates as expected from a quasi-continuous attractor. Furthermore, we find that the transition strategy in our model changes from predominantly smooth transitions to jumps at $\Delta\theta \approx 90$ deg, which matches experiments well (Kim et al., 2017).

Following a 2-s stimulation, the network activity has converged to the new cued location. After the stimulation has been turned off, the bump remains at the new location (within the angular resolution $\Delta\phi$ of the network), supported by internal network dynamics alone (fig. 2.2D). We confirmed in additional simulations that the bump does not drift away from the stimulated location for extended periods of time (3-minute duration tested, only 3 s shown), and for all discrete locations in the HD network (only six locations shown). Therefore, we conclude that the HD network is a quasi-continuous attractor that can reliably sustain a heading representation over time in all HD locations. Note that for the network size used ($N^{HD} = 60$) we still obtain discrete attractors with separated basins of attraction; however it is expected that with increasing N^{HD} adjacent attractors will merge when the intrinsic noise overcomes the barrier separating them. Indeed, we find that for $N^{HD} = N^{HR} = 120$ it is easier to diffuse to adjacent attractors in the presence of synaptic input noise; for the impact of noise, see fig. B.6C in Appendix B.2. In reality, the bump may drift away due to asymmetries in the connectivity of the biological circuit as well as intrinsic noise (Burak and Fiete, 2012); see also Appendix B.2. In flies, for instance, the bump can stay put only for several seconds (Kim et al., 2017).

2.5 Learning results in synaptic connectivity that matches the one in the fly

To gain more insight into how the network achieves PI and attains CAN properties, we show how the synaptic weights of the network are tuned during a developmental period (fig. 2.3). fig. 2.3A,B shows the learned recurrent synaptic weights among the HD cells, W^{rec}, and the learned synaptic weights from HR to HD cells, W^{HR}, respectively. Circular symmetry is apparent in both matrices, a crucial property for a symmetric ring attractor. Therefore we also plot the profiles of the learned

weights as a function of receptive field difference in fig. 2.3C. Note that pixelized appearance in these plots is due to the fact that two adjacent HD neurons are tuned for the same HD, and develop identical synaptic strengths.

First, we discuss the properties of the learned weights. Local excitatory connections have developed along the main diagonal of W^{rec}, similar to what is observed in the CX (Turner-Evans et al., 2020). This local excitation can be readily seen in the weight profile of W^{rec} in fig. 2.3C, and it is the substrate that allows the network to support stable activity bumps in virtually any location. In addition, we observe inhibition surrounding the local excitatory profile in both directions. This inhibition emerges despite the fact that we provide global inhibition to all HD cells (I_{inh}^{HD} parameter, Methods), in line with suggestions from previous work (Kim et al., 2017). Surrounding inhibition was a feature we observed consistently in learned networks of different sizes and for different global inhibition levels. Finally, the angular offset of the two negative sidelobes in the connectivity depends on the size and shape of the entrained HD bump (for details, see Appendix B.6).

Furthermore, we find a consistent pattern of both L-HR and R-HR populations to excite the direction for which they are selective (fig. 2.3C), which is also similar to what is observed in the CX (Turner-Evans et al., 2020). Excitation in one direction is accompanied by inhibition in the reverse direction in the learned network. As a result of the symmetry in our learning paradigm, the connectivity profiles of L-HR and R-HR cells are mirrored versions of each other, which is also clearly visible in fig. 2.3C. The inhibition of the reverse direction has a width comparable to the bump size and acts as a “break” to prevent the bump from moving in this direction. The excitation in the selective direction, on the other hand, has a wider profile, which allows the network to path integrate for a wide range of angular velocities, i.e., for high angular velocities neurons further downstream can be "primed" and activated in rapid succession. Indeed, when we remove the wide projections from the excitatory connectivity, PI performance is impaired for the higher angular velocities exclusively (fig. B.2). The even weight profile in W^{rec} and the mirror symmetry for L-HR vs. R-HR profiles in W^{HR}, together with the circular symmetry of the weights throughout the ring, guarantee that there is no side bias (i.e., tendency of the bump to favor one direction of movement versus the other) during PI. Indeed, the PI error distribution in fig. 2.2B remains symmetric throughout the 60-s simulations.

Next, we focus our attention on the dynamics of learning. For training times larger than a few hours, the absolute learning error drops and settles to a low value,

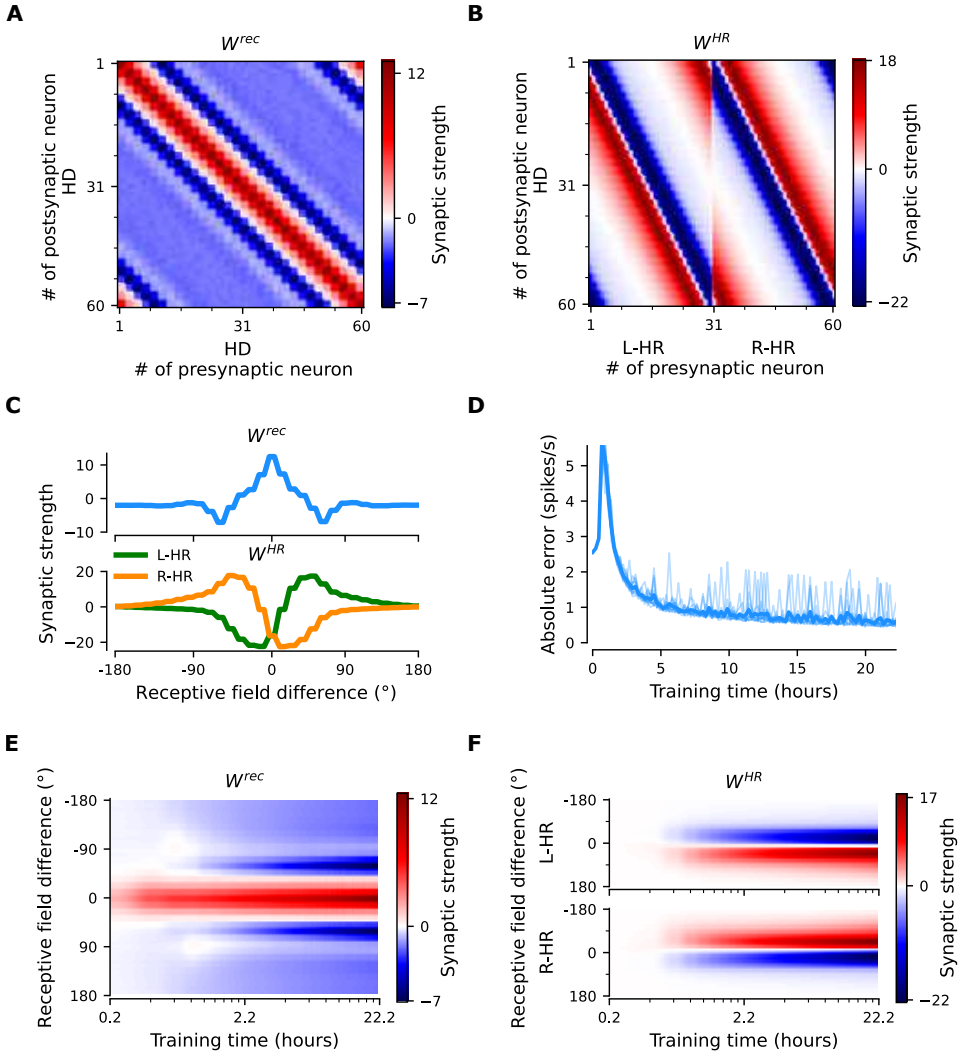

Figure 2.3: **Network connectivity during and after learning.** (A), (B) The learned weight matrices (color coded) of recurrent connections in the HD ring, W^{rec}, and of HR-to-HD connections, W^{HR}, respectively. Note the circular symmetry in both matrices. (C) Profiles of (A) and (B), averaged across presynaptic neurons. (D) Absolute learning error in the network (eq. (2.19)) for 12 simulations (transparent lines) and average across simulations (opaque line). At time $t = 0$, we initialize all the plastic weights at random and train the network for 8×10^4 s ($\sim$ 22 hours). The mean learning error increases in the beginning while a bump in W^{rec} is emerging, which is necessary to generate a pronounced bump in the network activity. For weak activity bumps, absolute errors are small because the overall network activity is low. After $\sim$1 hour of training, the mean learning error decreases with increasing training time and converges to a small value. (E), (F) Time courses of development of the profiles of W^{rec} and W^{HR}, respectively. Note the logarithmic time scale.

indicating that learning has converged after $\sim$20 hours (or 4000 cycles, each cycle lasting $1/\eta$) of training time (fig. 2.3D). The non-zero value of the final error is only

due to errors occurring at the edges of the bump (fig. B.3A, top panel). An intuitive explanation of why these errors persist is that the velocity pathway is learning to predict the visual input; as a result, when the visual input is present, the velocity pathway creates errors that are consistent with PI velocity biases in darkness.

Figure 2.3E,F shows the weight development history for the entire simulation. The first structure that emerges during learning is the local excitatory recurrent connections in W^{rec}. For these early stages of learning, the initial connectivity is controlled by the autocorrelation of the visual input, which gets imprinted in the recurrent connections by means of Hebbian co-activation of adjacent HD neurons. As a result, the width of the local excitatory profile mirrors the width of the visual input. Once a clear bump is established in the HD ring, the HR connections are learned to support bump movement, and negative sidelobes in W^{rec} emerge. To understand the shape of the learned connectivity profiles and the dynamics of their development, we study a reduced version of the full model, which follows learning in bump-centric coordinates (see Appendix B.6). The reduced model produces a connectivity strikingly similar to the full model, and highlights the important role of non-linearities in the system.

So far we have shown results in which our model far outperforms flies in terms of PI accuracy. To bridge this gap, we add noise to the weight connectivity in fig. 2.3A,B and obtain the connectivity matrices in fig. B.4A,B, respectively. This perturbation of the weights could account for irregularities in the fly HD system owning to biological factors such as uneven synaptic densities. The resulting neural velocity gain curve in fig. B.4E is impaired mainly for small angular velocities (cf. fig. 2.2C). Interestingly, it now bears greater similarity to the one observed in flies, because the previously flat area for small angular velocities is wider (flat for $|v| < 60$ deg/s, cf. extended data fig. 7G,J in Seelig and Jayaraman (2015)). This happens because the noisy connectivity is less effective in initiating bump movement. Finally, the PI errors in the network with noisy connectivity grow much faster and display a strong side bias (fig. B.4D, cf. fig. 2.2B). The latter can be attributed to the fact that the noise in the connectivity generates local minima that are easier to transverse from one direction vs. the other. Side bias can also emerge if the learning rate η in eq. (2.16) is increased, effectively forcing learning to converge faster to a local minimum, which results in slight deviations from circularly symmetric connectivity (data not shown). It is therefore expected that different animals will display different degrees and directions of side bias during PI, owning either to fast

learning or asymmetries in the underlying neurobiology. Since the exact behavior of the network with noise in the connectivity depends on the specific realization, we also generate multiple such networks and estimate the diffusion coefficient during path integration, which quantifies how fast the width of the PI error distribution in fig. B.4D increases. We find the grand average to be 82.3 ± 15.7 deg^2/s, which is considerably larger (Student's t-test, 95% conf. intervals for a total of 12 networks) than the diffusion coefficient for networks without a perturbation in the weights (24.5 deg^2/s in fig. 2.2B). Finally, in Appendix B.2 we also incorporate random Gaussian noise to all inputs, which can account for noisy percepts or stochasticity of spiking, and show that learning is not disrupted even for high noise levels.

2.6 Fast adaptation of neural velocity gain

Having shown how PI and CAN properties are learned in our model, we now turn our attention to the flexibility that our learning setup affords. Motivated by augmented-reality experiments in rodents where the relative gain of visual and self-motion inputs is manipulated (Jayakumar et al., 2019), we test whether our network can rewire to learn an arbitrary gain between the two. In other words, we attempt to learn an arbitrary gain g between the idiothetic angular velocity v sensed by the HR cells and the neural velocity $g \cdot v$ dictated by the allothetic visual input. This simulates the conditions in an augmented reality environment, where the speed at which the world around the animal rotates is determined by the experimenter, but the proprioceptive sense of head angular velocity remains the same.

Starting with the learned network shown in fig. 2.3, which displayed gain $g = 1$, we suddenly switch to a different gain, i.e., we learn weights for $g \in \{0.25, 0.5, 1.5, 2\}$. In all cases, we observe that the network readily rewires to achieve the new gain. The mean learning error after the gain switch is initially high, but reaches a lower, constant level after at most 3 hours of training (fig. 2.4A). We note that convergence is much faster compared to the time it takes for the gain-1 network to emerge from scratch (compare to fig. 2.3D), especially for the smaller gain changes. Importantly, fig. 2.4B shows that PI performance in the resulting networks is excellent for the new gains, with some degradation only for very low and very high angular velocities. There are two reasons why high angular velocities are not learned that well: limited training of these velocities, and saturation of HR cell activity. Both reasons are by design and do not reflect a fundamental limit of the network. In Appendix B.3 we show that without the aforementioned limitations the network learns to path-integrate up to an angular velocity limit set by synaptic delays and that the bump

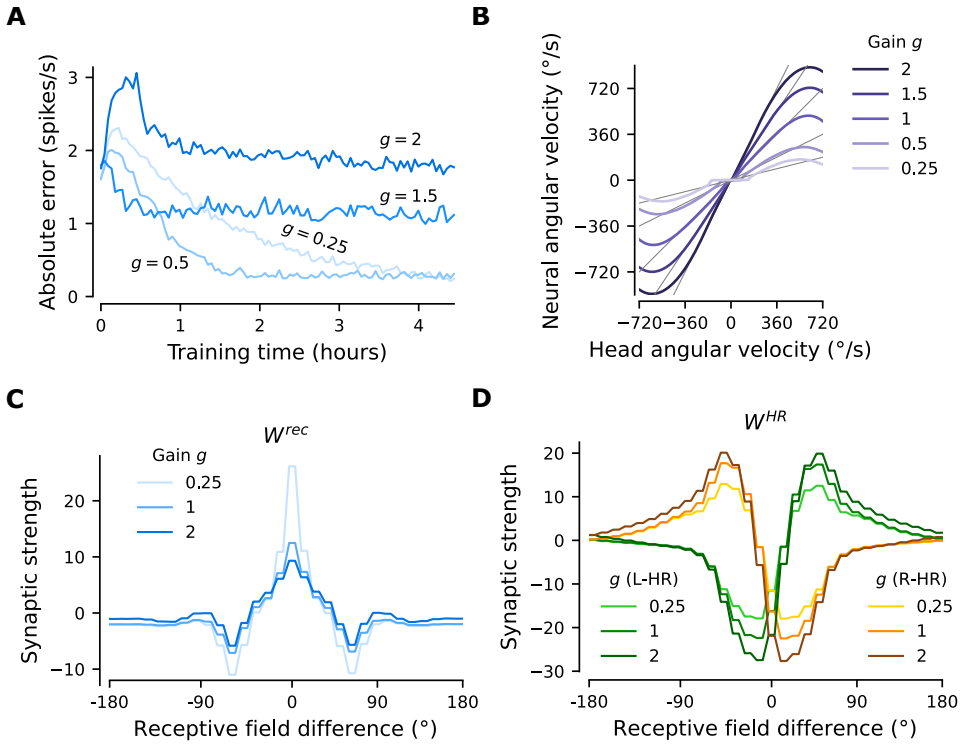

Figure 2.4: **The network rapidly adapts to new gains.** Starting from the converged network in fig. 2.3, we change the gain g between visual and self-motion inputs, akin to experiments conducted in VR in flies and rodents (Seelig and Jayaraman, 2015; Jayakumar et al., 2019). (A) The mean learning error averaged across 12 simulations for each gain. After an initial increase due to the change of gain, the errors decrease rapidly and settle to a lower value. The steady-state values depend on the gain due to the by-design impairment of high angular velocities, which affects high gains preferentially. Crucially, adaptation to a new gain is much faster than learning the HD system from scratch (cf. fig. 2.3D). (B) Velocity gain curves for different gains. The network has remapped to learn accurate PI with different gains for the entire dynamic range of head angular velocity inputs (approx. $[-500, 500]$ deg/s). (C), (D) Final profiles of W^{rec} and W^{HR}, respectively, for different gains.

width sets a trade-off between location and velocity-integration accuracy in the HD system.

Figure 2.4C,D compare the weight profiles of the circularly symmetric matrices W^{rec} and W^{HR} resulting from the initial gain $g = 1$, with the weight profiles resulting from adaptation to the most extreme gains shown in fig. 2.4, i.e., $g \in \{0.25, 2\}$. An increase in gain slightly suppresses the recurrent connections and slightly amplifies the HR-to-HD connections, while a decrease in gain substantially amplifies the recurrent connections and slightly suppresses the HR-to-HD connections. The latter explains why the flat region for small angular velocities in fig. 2.4B has

been extended for $g \in \{0.25, 0.5\}$: it is now harder for small angular velocities to overcome the attractor formed by stronger recurrent weights and move the bump.

Finally, we address the limits of the ability of the network to rewire to new gains (fig. B.5). We find that after rewiring the performance is excellent for gains between 0.25 and 4.5. The network can even reverse its gain to $g = -1$, i.e., when allothetic and idiothetic inputs are signaling movement in opposite directions. However, for larger gain changes, learning takes longer.

2.7 Summary of findings

The ability of animals to navigate in the absence of external cues is crucial for their survival. Head direction, place, and grid cells provide internal representations of space (Ranck, 1984; Moser, Kropff, and Moser, 2008) that can persist in darkness and possibly support path integration (PI) (Mizumori and Williams, 1993; Quirk, Muller, and Kubie, 1990; Hafting et al., 2005). Extensive theoretical work has focused on how the spatial navigation system might rely on continuous attractor networks (CANs) to maintain and update a neural representation of the animal's current location. Special attention was devoted to models representing orientation, with the ring attractor network being one of the most famous of these models (Amari, 1977; Ben-Yishai, Bar-Or, and Sompolinsky, 1995; Skaggs et al., 1995; Seung, 1996). So far, modelling of the HD system has been relying on hand-tuned synaptic connectivity (Zhang, 1996; Xie, Hahnloser, and Seung, 2002; Turner-Evans et al., 2017; Page, Walters, and Stringer, 2018) without reference to its origin; or has been relying on synaptic plasticity rules that either did not achieve gain-1 PI (Stringer et al., 2002) or were not biologically plausible (Hahnloser, 2003).

Inspired by the recent discovery of a ring attractor network for HD in *Drosophila* (Seelig and Jayaraman, 2015), we show how a biologically plausible learning rule leads to the emergence of a circuit that achieves gain-1 PI in darkness. The learned network features striking similarities in terms of connectivity to the one experimentally observed in the fly (Turner-Evans et al., 2020), and reproduces experiments on CAN dynamics (Kim et al., 2017) and gain changes between external and self-motion cues in rodents (Jayakumar et al., 2019). Furthermore, an impairment of PI for small angular velocities is observed in the mature network, which is a feature that has been reported in experiments (Seelig and Jayaraman, 2015). Finally, the proposed learning rule can serve to compensate deviations from circular symmetry

in the synaptic weight profiles; such deviations are expected in biological systems and — if not compensated — could lead to large PI errors.

The mature circuit displays two properties characteristic of CANs: 1) it can support and actively maintain a local bump of activity at a virtual continuum of locations, and 2) it can move the bump across the network by integrating self-motion cues. Note that we did not explicitly train the network to achieve these CAN properties, but they rather emerged in a self-organized manner.

To achieve gain-1 PI performance, our network must attribute learning errors to the appropriate weights. The learning rule we adopt in eq. (2.1) is a "delta-like" rule, with a learning error that gates learning in the network, and a Hebbian component that comes in the form of the postsynaptic potential and assigns credit to synapses that are active when errors are large. The learning rule leads to the emergence of both symmetric local connectivity between HD cells (which is required for bump maintenance and stability), and asymmetric connectivity from HR to HD cells (which is required for bump movement in darkness). The first happens because adjacent neurons are co-active due to correlated visual input; the second because only one HR population is predominantly active during rotation: the population that corresponds to the current rotation direction. Crucial to the understanding of the learning dynamics of the model was the development of a reduced model, which follows learning in bump-centric coordinates and is analytically tractable (see Appendix B.6). The reduced model can be extended to higher dimensional manifolds (Gardner et al., 2022), and therefore it offers a general framework to study how activity-dependent synaptic plasticity shapes CANs.

2.8 Relation to experimental literature

Our work comes at a time at which the fly HD system receives a lot of attention (Seelig and Jayaraman, 2015; Turner-Evans et al., 2020; Kim et al., 2017; Kim et al., 2019; Fisher et al., 2019), and suggests a mechanism of how this circuit could self-organize during development. Synaptic plasticity has been shown to be important in this circuit for anchoring the visual input to the HD neurons when the animal is exposed to a new environment (Kim et al., 2019; Fisher et al., 2019). This has also been demonstrated in models of the mammalian HD system (Skaggs et al., 1995; Zhang, 1996; Song and Wang, 2005). Here we assume that an initial anchoring of the topographic visual input to the HD neurons with arbitrary offset with respect to external landmarks already exists prior to the development of the PI circuit; such an

anchoring could even be prewired. In our model, it is sufficient that the visual-input tuning is local and topographically arranged. Once the PI circuit has developed, visual connections could be anchored to different environments, as shown by Kim et al. (2019) and Fisher et al. (2019). Alternatively, the HD system itself could come prewired with an initial gross connectivity, sufficient to anchor the visual input; in this case, our learning rule would enable fine tuning of this connectivity for gain-1 PI. In either case, for the sake of simplicity and without loss of generality, we study the development of the path-integrating circuit while the animal moves in the same environment, and keep the visual input-tuning fixed. Therefore, the present work addresses the important question of how the PI circuit itself could be formed, and it is complementary to the problem of how allothetic inputs to the PI circuit are wired (Fisher et al., 2019; Kim et al., 2019). The interplay of the two forms of plasticity during development would be of particular future interest.

A requirement for the learning rule we use is that information about the firing rate of HD neurons is available at the axon-distal compartment. There is no evidence for active backpropagation of APs in E-PG neurons in the fly, but passive backpropagation would suffice in this setting. In fact, passive spread of activity has been shown to attenuate weakly in central fly neurons (Nathan W. Gouwens, 2009). In HD neurons, the axon-proximal and axon-distal compartments belong to the same dendritic tuft (fig. 2.1E), and since we assume that the axon initial segment is close to the axon-proximal compartment, the generated AP would need to propagate only a short distance compared to the effective electrotonic length. This means that APs would not be attenuated much on their way from the axon initial segment to the axon-distal compartment, and thus would maintain some of their high-frequency component, which could be used at synapses to differentiate them from slower postsynaptic potentials.

In fig. 2.4 we show that our network can adapt to altered gains much faster than the time required to learn the network from scratch. Our simulations are akin to experiments where rodents are placed in a VR environment and the relative gain between visual and proprioceptive signals is altered by the experimenter (Jayakumar et al., 2019). In this scenario, Jayakumar et al. (2019) found that the PI gain of place cells can be recalibrated rapidly. In contrast, Seelig and Jayaraman (2015) found that PI gain in darkness is not significantly affected when flies are exposed to different gains in light conditions. We note, however, that Seelig and Jayaraman (2015) tested mature animals (8–11 days old), whereas plasticity in the main HD network is

presumably stronger in younger animals. Also note that the manipulation we use to address adaptation of PI to different gains differs from the one in (Kim et al., 2019) who used optogenetic stimulation of the HD network combined with rotation of the visual scene to trigger a remapping of the visual input to the HD cells in a Hebbian manner. The findings in Jayakumar et al. (2019) can only be reconciled by plasticity in the PI circuit, and not in the sensory inputs to the circuit.

In order to address the core mechanisms that underlie the emergence of a path integrating network, we use a model that is a simplified version of the biological circuit. For example, we did not model inhibitory neurons explicitly and omitted some of the recurrent connectivity in the circuit, whose functional role is uncertain (Turner-Evans et al., 2020). We also choose to separate PI from other complex processes that occur in the CX (Raccuglia et al., 2019). Finally, we do not force the network to obey Dale's law and do not model spiking explicitly.

Nevertheless, after learning, we obtain a network connectivity that is strikingly similar to the one of the fly HD system. Indeed, the mature model exhibits local excitatory connectivity in the HD neurons (fig. 2.3A,C), which in the fly is mediated by the excitatory loop from E-PG to P-EG to P-EN2 and back to E-PG (Turner-Evans et al., 2020), a feature that hand-tuned models of the fly HD system did not include (Turner-Evans et al., 2017). Furthermore, the HR neurons have excitatory projections towards the directions they are selective for (fig. 2.3B,C), similar to P-EN1 neurons in the fly. Interestingly, these key features that we uncover from learning have been utilized in other hand-tuned models of the system (Turner-Evans et al., 2017; Kim et al., 2017; Kim et al., 2019). Future work could endeavor to come closer to the architecture of the fly HD system and benefit from the incorporation of more neuron types and the richness of recurrent connectivity that has been discovered in the fly (Turner-Evans et al., 2020).

Compared to the fly, our network achieved better PI performance. As a simple way to match the performances, we added noise to the learned connectivity in the model; however this is not an explanation why the fly performs worse. Indeed, there could be multiple reasons why PI performance is worse in the biological circuit. For instance, a confounder that would affect performance but not necessarily learning could be the presence of inputs that are unrelated to path integration, e.g., inputs related to circadian cycles and sleep (Raccuglia et al., 2019). In the presence of such confounders, a precise tuning of the weights might be crucial in order to reach the performance of the fly. In other words, only if the model outperforms the biological

circuit in a simplified setting, it has a chance to perform as well in a realistic setting, with all the additional complexities the latter comes with.

2.9 Relation to theoretical literature

A common problem with CANs is that they require fine tuning: even a slight deviation from the optimal synaptic weight tuning leads to catastrophic drifting (Goldman, Compte, and Wang, 2009). A way around this problem is to sacrifice the continuity of the attractor states in favor of a discrete number of stable states that are much more robust to noise or weight perturbations (Kilpatrick, Ermentrout, and Doiron, 2013). In our network, the small number of HD neurons enables a coarse-grained representation of heading; the network is a CAN only in a quasi-continuous manner, and the number of discrete attractors corresponds to the number of HD neurons. This makes it harder to transition to adjacent attractors, since a "barrier" has to be overcome in the quasi-continuous case (Kilpatrick, Ermentrout, and Doiron, 2013). The somewhat counter-intuitive conclusion follows that a CAN with more neurons and, as a result, finer angular resolution, will not be as potent in maintaining activity, and diffusion to nearby attractors will be easier since the barrier will be lower. Indeed, we found that doubling the number of neurons produces a CAN that is less robust to noise. Overall, the quasi-continuous and coarse nature of the attractor shields the internal representation of heading against the ever-present biological noise, which would otherwise lead to diffusion of the bump with time. The fact that the network can still path-integrate accurately with this coarse-grained representation of heading is remarkable.

Seminal theoretical work on ring attractors has proven that in order to achieve gain-1 PI, the asymmetric component of the network connectivity (corresponding here to W^{HR}) needs to be proportional to the derivative of the symmetric component (corresponding to W^{rec}) (Zhang, 1996). However, this result rests on the assumption that asymmetric and symmetric weight profiles are mediated by the same neuronal population, as in the double-ring architecture proposed by Xie, Hahnloser, and Seung (2002) and Hahnloser (2003), but does not readily apply to the architecture of the fly HD system where HD and HR cells are separate. In our learned network we find that the HR weight profile is not proportional to the derivative of the recurrent weight profile, therefore this requirement is not necessary for gain-1 PI in our setting. Note that our learning setup can also learn gain-1 PI for a double-ring architecture, which additionally obeys Dale's law (Vafidis (2019), *Learning of a path-integrating circuit* [Unpublished master's thesis], Technical University of

Berlin). Finally, we emphasize that circular symmetry is not a necessary condition for a ring attractor (Darshan and Rivkind, 2021). Rather, symmetry in our model results from the symmetry in the architecture, the symmetrially prewired weights, and the symmetric stimulus space. If any of those were to be relaxed, the resulting network would not be circular symmetric; then, the reduced model analysis that we perform in Appendix B.6 would also not be feasible, because local asymmetries in the setup would result in non-local deviations from circular symmetry of the learned weights, which was our main assumption there. Nevertheless, we demonstrated that the full model can handle such asymmetries in the setup and learn accurate PI (see Appendix B.4).

Our learning setup, inspired by Urbanczik and Senn (2014), is similar to the one in Guerguiev, Lillicrap, and Richards (2017) in the sense that both involve compartmentalized neurons that receive "target" signals in a distinct compartment. It differs, however, in the algorithm and learning rule used. Guerguiev, Lillicrap, and Richards (2017) use local gradient descent during a "target" phase, which is separate from a forward propagation phase, akin to forward/backward propagation stages in conventional deep learning. In contrast, we use a modified Hebbian rule, and in our model "forward" computation and learning happen at the same time; time multiplexing, whose origin in the brain is unclear, is not required. Our setting would be more akin to the one in Guerguiev, Lillicrap, and Richards (2017) if an episode of PI in darkness would be required before an episode of learning in light conditions, which does not seem in line with the way animals naturally learn.

Previous theoretical work showed that head direction cells, head rotation cells, and grid cells emerge in neural networks trained for PI (Banino et al., 2018; Cueva and Wei, 2018; Cueva et al., 2020). These networks were trained with backpropagation, therefore achieving gain-1 PI was not their primary focus; rather, this work elegantly demonstrated that the aforementioned cell types are efficient representations for spatial navigation that could be learned from experience.

2.10 Testable predictions

We devote this section to discussing predictions of our model, and we suggest future experiments in flies and, potentially, other animal models. An obvious prediction of our model is that synaptic plasticity is critical for the development of the PI network for heading, and the lack of a supervisory allothetic sensory input (e.g., visual) during development should disrupt the formation of the PI system. Previous

experimental work showed that head direction cells in rat pups displayed mature properties already in their first exploration of the environment outside their nest (Langston et al., 2010), which may seem to contradict our assumption that the PI circuit wires during development; however, directional selectivity of HD cells in the absence of allothetic inputs and PI performance were not tested in this study. In addition, it has been shown that visually impaired flies were not able to learn to accurately estimate the size of their body. This type of learning also requires visual inputs and, upon consolidation, remains stable (Krause et al., 2019).

We also predict that HD neurons have a compartmental structure where idiothetic inputs are separated from allothetic sensory inputs, which initiate action potentials more readily due to being electrotonically closer to the axon initial segment. While we already demonstrate the separation of allothetic and idiothetic inputs to E-PG neurons in the fly EB (fig. 2.1E, fig. B.1), our prediction can only be tested with electrophysiological experiments. Another model prediction that can be tested only with electrophysiology is that APs backpropagate from the axon-proximal compartment (at least passively but with little attenuation) to the axon-distal compartment. Then spikes could be separated from postsynaptic potentials locally at the synapse by cellular mechanisms sensitive to the spectral density of the voltage.

Finally, similarly to place cell studies in rodents (Jayakumar et al., 2019), we predict that during development the PI system can adapt to experimenter-defined gain manipulations, and that it can do so faster than the time required for the system to develop from scratch. Therefore, a suggestion from this study would be to repeat in young flies the adaptation experiments by Seelig and Jayaraman (2015).

2.11 Outlook

The present study adds to the growing literature of potential computational abilities of compartmentalized neurons (Poirazi, Brannon, and Mel, 2003; Gidon et al., 2020; Payeur et al., 2021). The associative HD neuron used in this study is a coincidence detector, which serves to associate external and internal inputs arriving at different compartments of the cell. Coupled with memory-specific gating of internally generated inputs, coincidence detection has been suggested to be the fundamental mechanism that allows the mammalian cortex to form and update internal knowledge about external contingencies (Doron et al., 2020; Shin, Doron, and Larkum, 2021). This structured form of learning does not require engineered "hints" during training, and it might be the reason why neural circuits evolved to be so efficient at reasoning

about the world, with the mammalian cortex being the pinnacle of this achievement. Here we demonstrate that learning at the cellular level can predict external inputs (visual information) by associating firing activity with internally generated signals (velocity inputs) during training. This effect is due to the anti-Hebbian component of the learning rule in eq. (2.12), where the product of postsynaptic axon-distal and presynaptic activity comes with a negative sign. Specifically, it has previously been demonstrated that anti-Hebbian synaptic plasticity can stabilize persistent activity (Xie and Seung, 2000) and perform predictive coding (Bell et al., 1997; Hahnloser, 2003). At the population level, this provides a powerful mechanism to internally produce activity patterns that are identical to the ones induced from an external stimulus. This mechanism can serve as a way to anticipate external events or, in our case, as a way of "filling in" missing information in the absence of external inputs.

Local, Hebb-like learning rules are considered a weak form of learning, due to their inability to utilize error information in a sophisticated manner. Despite that, we show that local associative learning can be particularly successful in learning appropriate fine-tuned synaptic connectivity, when operating within a cell structured for coincidence detection. Therefore, in learning and reasoning about the environment, our study highlights the importance of inductive biases with developmental origin (e.g., allothetic and idiothetic inputs arrive in different compartments of associative neurons) (Lake et al., 2016).

In conclusion, Chapter I addresses the age-old question of how to develop a CAN that performs accurate, gain-1 PI in the absence of external sensory cues. We show that this feat can be achieved in a network model of the HD system by means of a biologically plausible learning rule at the cellular level. Even though our network architecture is tailored to the one of the fly CX, the learning setup where idiothetic and allothetic cues are associated at the cellular level is general and can be applied to other PI circuits. Of particular interest is the rodent HD system: despite the lack of evidence for a topographically-organized recurrent HD network in rodents, a one-dimensional HD manifold was extracted in an unsupervised way (Chaudhuri et al., 2019). Therefore, our work lays the path to study the development of ring-like neural manifolds in mammals. Finally, it has been shown that grid cells in mammals form a continuous attractor manifold with toroidal topology (Gardner et al., 2022). It would be interesting to see if a similar mechanism underlies the emergence of PI in place and grid cells. Our model can be extended to higher dimensional CAN manifolds and provides a framework to interrogate this assumption.

References

Abbott, Larry F. et al. (Sept. 2020). "The mind of a mouse." In: *Cell* 182.6, pp. 1372–1376. DOI: 10.1016/j.cell.2020.08.010. URL: https://doi.org/10.1016/j.cell.2020.08.010.

Amari, Shun-ichi (1977). "Dynamics of pattern formation in lateral-inhibition type neural fields." In: *Biological Cybernetics* 27.2, pp. 77–87. DOI: 10.1007/bf00337259. URL: https://doi.org/10.1007/bf00337259.

Banino, Andrea et al. (May 2018). "Vector-based navigation using grid-like representations in artificial agents." In: *Nature* 557.7705, pp. 429–433. DOI: 10.1038/s41586-018-0102-6. URL: https://doi.org/10.1038/s41586-018-0102-6.

Bell, Curtis C. et al. (May 1997). "Synaptic plasticity in a cerebellum-like structure depends on temporal order." In: *Nature* 387.6630, pp. 278–281. DOI: 10.1038/387278a0. URL: https://doi.org/10.1038/387278a0.

Ben-Yishai, Rani, R. Lev Bar-Or, and Haim Sompolinsky (Apr. 1995). "Theory of orientation tuning in visual cortex." In: *Proceedings of the National Academy of Sciences of the United States of America* 92.9, pp. 3844–3848. DOI: 10.1073/pnas.92.9.3844. URL: https://doi.org/10.1073/pnas.92.9.3844.

Brandão, Sofia C., Marion Silies, and Carlotta Martelli (Jan. 2021). "Adaptive temporal processing of odor stimuli." In: *Cell and Tissue Research* 383.1, pp. 125–141. DOI: 10.1007/s00441-020-03400-9. URL: https://doi.org/10.1007/s00441-020-03400-9.

Brea, Johanni et al. (June 2016). "Prospective coding by spiking neurons." In: *PLoS Computational Biology* 12.6. Ed. by Peter E. Latham, e1005003. DOI: 10.1371/journal.pcbi.1005003. URL: https://doi.org/10.1371/journal.pcbi.1005003.

Burak, Yoram and Ila R. Fiete (Feb. 2009). "Accurate path integration in continuous attractor network models of grid cells." In: *PLoS Computational Biology* 5.2. Ed. by Olaf Sporns, e1000291. DOI: 10.1371/journal.pcbi.1000291. URL: https://doi.org/10.1371/journal.pcbi.1000291.

– (Oct. 2012). "Fundamental limits on persistent activity in networks of noisy neurons." In: *Proceedings of the National Academy of Sciences of the United States of America* 109.43, pp. 17645–17650. DOI: 10.1073/pnas.1117386109. URL: https://doi.org/10.1073/pnas.1117386109.

Carpenter, Gail A. and Stephen Grossberg (Jan. 1987). "A massively parallel architecture for a self-organizing neural pattern recognition machine." In: *Computer Vision, Graphics, and Image Processing* 37.1, pp. 54–115. DOI: 10.1016/s0734-189x(87)80014-2. URL: https://doi.org/10.1016/s0734-189x(87)80014-2.

Chaudhuri, Rishidev et al. (Aug. 2019). "The intrinsic attractor manifold and population dynamics of a canonical cognitive circuit across waking and sleep." In: *Nature Neuroscience* 22.9, pp. 1512–1520. DOI: 10.1038/s41593-019-0460-x. URL: https://doi.org/10.1038/s41593-019-0460-x.

Clements, Jody et al. (Jan. 2020). "neuPrint: Analysis tools for EM connectomics." In: *bioRxiv*. DOI: 10.1101/2020.01.16.909465. URL: https://doi.org/10.1101/2020.01.16.909465.

Cueva, Christopher J. and Xue-Xin Wei (2018). "Emergence of grid-like representations by training recurrent neural networks to perform spatial localization." In: *arXiv preprint arXiv:1803.07770*.

Cueva, Christopher J. et al. (May 2020). "Emergence of functional and structural properties of the head direction system by optimization of recurrent neural networks." In: *arXiv:1912.10189 [cs, q-bio, stat]*. arXiv: 1912.10189. URL: http://arxiv.org/abs/1912.10189 (visited on 10/21/2021).

D'Albis, Tiziano and Richard Kempter (Oct. 2020). "Recurrent amplification of grid-cell activity." In: *Hippocampus* 30.12, pp. 1268–1297. DOI: 10.1002/hipo.23254. URL: https://doi.org/10.1002/hipo.23254.

Darshan, Ran and Alexander Rivkind (June 2021). "Learning to represent continuous variables in heterogeneous neural networks." In: *bioRxiv*. DOI: 10.1101/2021.06.01.446635. URL: https://doi.org/10.1101/2021.06.01.446635.

Darwin, Charles (Apr. 1873). "Origin of certain instincts." In: *Nature* 7.179, pp. 417–418. DOI: 10.1038/007417a0. URL: https://doi.org/10.1038/007417a0.

Doron, Guy et al. (2020). "Perirhinal input to neocortical layer 1 controls learning." In: *Science* 370.6523, eaaz3136. ISSN: 0036-8075. DOI: 10.1126/science.aaz3136. eprint: https://science.sciencemag.org/content/370/6523/eaaz3136.full.pdf. URL: https://science.sciencemag.org/content/370/6523/eaaz3136.

Eichenbaum, Howard (Apr. 2017). "The role of the hippocampus in navigation is memory." In: *Journal of Neurophysiology* 117.4, pp. 1785–1796. DOI: 10.1152/jn.00005.2017. URL: https://doi.org/10.1152/jn.00005.2017.

Etienne, Ariane S., Roland Maurer, and Valérie Séguinot (Jan. 1996). "Path integration in mammals and its interaction with visual landmarks." In: *Journal of Experimental Biology* 199.1, pp. 201–209. URL: https://www.ncbi.nlm.nih.gov/pubmed/8576691.

Fisher, Yvette E. et al. (Nov. 2019). "Sensorimotor experience remaps visual input to a heading-direction network." In: *Nature* 576.7785, pp. 121–125. DOI: 10.1038/s41586-019-1772-4. URL: https://doi.org/10.1038/s41586-019-1772-4.

Franconville, Romain, Celia Beron, and Vivek Jayaraman (2018). "Building a functional connectome of the Drosophila central complex." In: *eLife* 7, e37017. DOI: 10.7554/elife.37017.001. URL: https://doi.org/10.7554/elife.37017.001.

Gallistel, Charles R. (Jan. 1993). "The organization of learning." Bradford Books/MIT Press. ISBN: 9780262570985.

Gardner, Richard J. et al. (Jan. 2022). "Toroidal topology of population activity in grid cells." In: *Nature* 602.7895, pp. 123–128. DOI: 10.1038/s41586-021-04268-7. URL: https://doi.org/10.1038/s41586-021-04268-7.

Geurten, Bart R. H. et al. (Oct. 2014). "Saccadic body turns in walking Drosophila." In: *Frontiers in Behavioral Neuroscience* 8, p. 365. DOI: 10.3389/fnbeh.2014.00365. URL: https://doi.org/10.3389/fnbeh.2014.00365.

Gidon, Albert et al. (2020). "Dendritic action potentials and computation in human layer 2/3 cortical neurons." In: *Science* 367.6473, pp. 83–87.

Goldman, Mark S., Albert Compte, and Xiao-Jing Wang (2009). "Neural integrator models." In: *Encyclopedia of Neuroscience*. Elsevier, pp. 165–178. DOI: 10.1016/b978-008045046-9.01434-0. URL: https://doi.org/10.1016/b978-008045046-9.01434-0.

Green, Jonathan et al. (May 2017). "A neural circuit architecture for angular integration in Drosophila." In: *Nature* 546.7656, pp. 101–106. DOI: 10.1038/nature22343. URL: https://doi.org/10.1038/nature22343.

Green, Jonathan et al. (July 2019). "A neural heading estimate is compared with an internal goal to guide oriented navigation." In: *Nature Neuroscience* 22.9, pp. 1460–1468. DOI: 10.1038/s41593-019-0444-x. URL: https://doi.org/10.1038/s41593-019-0444-x.

Guerguiev, Jordan, Timothy P. Lillicrap, and Blake A. Richards (Dec. 2017). "Towards deep learning with segregated dendrites." In: *eLife* 6. DOI: 10.7554/eLife.22901.

Hafting, Torkel et al. (June 2005). "Microstructure of a spatial map in the entorhinal cortex." In: *Nature* 436.7052, pp. 801–806. URL: https://www.nature.com/articles/nature03721.

Hahnloser, Richard H. R. (Sept. 2003). "Emergence of neural integration in the head-direction system by visual supervision." In: *Neuroscience* 120.3, pp. 877–891. DOI: 10.1016/s0306-4522(03)00201-x. URL: https://doi.org/10.1016/s0306-4522(03)00201-x.

Itti, Laurent, Christof Koch, and Ernst Niebur (1998). "A model of saliency-based visual attention for rapid scene analysis." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20.11, pp. 1254–1259. DOI: 10.1109/34.730558. URL: https://doi.org/10.1109/34.730558.

Jayakumar, Ravikrishnan P. et al. (Feb. 2019). "Recalibration of path integration in hippocampal place cells." In: *Nature* 566.7745, pp. 533–537. DOI: 10.1038/s41586-019-0939-3. URL: https://doi.org/10.1038/s41586-019-0939-3.

Kilpatrick, Zachary P., Bard Ermentrout, and Brent Doiron (Nov. 2013). "Optimizing working memory with heterogeneity of recurrent cortical excitation." In: *Journal of Neuroscience* 33.48, pp. 18999–19011. DOI: 10.1523/jneurosci.1641-13.2013. URL: https://doi.org/10.1523/jneurosci.1641-13.2013.

Kim, Sung Soo et al. (May 2017). "Ring attractor dynamics in the Drosophila central brain." In: *Science* 356.6340, pp. 849–853. DOI: 10.1126/science.aal4835. URL: https://doi.org/10.1126/science.aal4835.

Kim, Sung Soo et al. (Nov. 2019). "Generation of stable heading representations in diverse visual scenes." In: *Nature* 576.7785, pp. 126–131. DOI: 10.1038/s41586-019-1767-1. URL: https://doi.org/10.1038/s41586-019-1767-1.

Krause, Tammo et al. (June 2019). "Drosophila acquires a long-lasting body-size memory from visual feedback." In: *Current Biology* 29.11, 1833–1841.e3. DOI: 10.1016/j.cub.2019.04.037. URL: https://doi.org/10.1016/j.cub.2019.04.037.

Lake, Brenden M. et al. (Nov. 2016). "Building machines that learn and think like people." In: *Behavioral and Brain Sciences* 40, e253. DOI: 10.1017/s0140525x16001837. URL: https://doi.org/10.1017/s0140525x16001837.

Langston, Rosamund F. et al. (June 2010). "Development of the spatial representation system in the rat." In: *Science* 328.5985, pp. 1576–1580. DOI: 10.1126/science.1188210. URL: https://doi.org/10.1126/science.1188210.

Larkum, Matthew E. (Mar. 2013). "A cellular mechanism for cortical associations: An organizing principle for the cerebral cortex." In: *Trends in Neurosciences* 36.3, pp. 141–151. DOI: 10.1016/j.tins.2012.11.006. URL: https://doi.org/10.1016/j.tins.2012.11.006.

Larkum, Matthew E., J. Julius Zhu, and Bert Sakmann (Mar. 1999). "A new cellular mechanism for coupling inputs arriving at different cortical layers." In: *Nature* 398.6725, pp. 338–341. DOI: 10.1038/18686. URL: https://doi.org/10.1038/18686.

McNaughton, Bruce L. et al. (1996). "Deciphering the hippocampal polyglot: The hippocampus as a path integration system." In: *Journal of Experimental Biology* 199.1, pp. 173–185.

Mittelstaedt, Marie Luise and Holst Mittelstaedt (Nov. 1980). "Homing by path integration in a mammal." In: *Naturwissenschaften* 67.11, pp. 566–567. DOI: 10.1007/bf00450672. URL: https://doi.org/10.1007/bf00450672.

Mizumori, Sheri J. and John D. Williams (Sept. 1993). "Directionally selective mnemonic properties of neurons in the lateral dorsal nucleus of the thalamus of rats." In: *Journal of Neuroscience* 13.9, pp. 4015–4028. DOI: 10.1523/jneurosci.13-09-04015.1993. URL: https://doi.org/10.1523/jneurosci.13-09-04015.1993.

Moser, Edvard I., Emilio Kropff, and May-Britt Moser (July 2008). "Place cells, grid cells, and the brain's spatial representation system." In: *Annual Review of Neuroscience* 31.1, pp. 69–89. DOI: 10.1146/annurev.neuro.31.061307.090723. URL: https://doi.org/10.1146/annurev.neuro.31.061307.090723.

Nathan W. Gouwens, Rachel I. Wilson (May 2009). "Signal propagation in Drosophila central neurons." In: *Journal of Neuroscience* 29.19, pp. 6239–6249. DOI: 10.1523/jneurosci.0764-09.2009. URL: https://doi.org/10.1523/jneurosci.0764-09.2009.

Neuser, Kirsa et al. (May 2008). "Analysis of a spatial orientation memory in Drosophila." In: *Nature* 453, pp. 1244–1247. DOI: 10.1038/nature07003. URL: https://doi.org/10.1038/nature07003.

O'Keefe, John and Lynn Nadel (Jan. 1978). "The hippocampus as a cognitive map." Oxford University Press. ISBN: 9780198572060.

Omoto, Jaison Jiro et al. (Apr. 2017). "Visual input to the Drosophila central complex by developmentally and functionally distinct neuronal populations." In: *Current Biology* 27.8, pp. 1098–1110. DOI: 10.1016/j.cub.2017.02.063. URL: https://doi.org/10.1016/j.cub.2017.02.063.

Page, Hector J. I., Daniel Walters, and Simon M. Stringer (Oct. 2018). "A speed-accurate self-sustaining head direction cell path integration model without recurrent excitation." In: *Network: Computation in Neural Systems* 29.1-4, pp. 37–69. DOI: 10.1080/0954898x.2018.1559960. URL: https://doi.org/10.1080/0954898x.2018.1559960.

Payeur, Alexandre et al. (May 2021). "Burst-dependent synaptic plasticity can coordinate learning in hierarchical circuits." In: *Nature Neuroscience* 24.7, pp. 1010–1019. DOI: 10.1038/s41593-021-00857-x. URL: https://doi.org/10.1038/s41593-021-00857-x.

Poirazi, Panayiota, Terrence Brannon, and Bartlett W. Mel (Mar. 2003). "Pyramidal neuron as two-layer neural network." In: *Neuron* 37.6, pp. 989–999. DOI: 10.1016/s0896-6273(03)00149-1. URL: https://doi.org/10.1016/s0896-6273(03)00149-1.

Quirk, Gregory J., Robert U. Muller, and John L. Kubie (June 1990). "The firing of hippocampal place cells in the dark depends on the rat's recent experience." In: *Journal of Neuroscience* 10.6, pp. 2008–2017. URL: https://www.ncbi.nlm.nih.gov/pubmed/2355262.

Raccuglia, Davide et al. (Nov. 2019). "Network-specific synchronization of electrical slow-wave oscillations regulates sleep drive in Drosophila." In: *Current Biology* 29.21, 3611–3621.e3. DOI: 10.1016/j.cub.2019.08.070. URL: https://doi.org/10.1016/j.cub.2019.08.070.

Ranck, James B. (1984). "Head direction cells in the deep layer of dorsal presubiculum in freely moving rats." In: *Society of Neuroscience Abstract* 10, p. 599. URL: https://ci.nii.ac.jp/naid/10022341517/en/.

Redish, A. David, Adam N. Elga, and David S. Touretzky (Jan. 1996). "A coupled attractor model of the rodent head direction system." In: *Network: Computation in Neural Systems* 7.4, pp. 671–685. DOI: 10.1088/0954-898x_7_4_004. URL: https://doi.org/10.1088/0954-898x_7_4_004.

Samsonovich, Alexei and Bruce L. McNaughton (Aug. 1997). "Path integration and cognitive mapping in a continuous attractor neural network model." In: *Journal of Neuroscience* 17.15, pp. 5900–5920. URL: https://www.ncbi.nlm.nih.gov/pubmed/9221787.

Seelig, Johannes D. and Vivek Jayaraman (May 2015). "Neural dynamics for landmark orientation and angular path integration." In: *Nature* 521.7551, pp. 186–191. DOI: 10.1038/nature14446. URL: https://doi.org/10.1038/nature14446.

Seung, H. Sebastian (Nov. 1996). "How the brain keeps the eyes still." In: *Proceedings of the National Academy of Sciences of the United States of America* 93.23, pp. 13339–13344. DOI: 10.1073/pnas.93.23.13339. URL: https://doi.org/10.1073/pnas.93.23.13339.

Shin, Jiyun N., Guy Doron, and Matthew E. Larkum (Oct. 2021). "Memories off the top of your head." In: *Science* 374.6567, pp. 538–539. DOI: 10.1126/science.abk1859. URL: https://doi.org/10.1126/science.abk1859.

Skaggs, William E. et al. (1995). "A model of the neural basis of the rat's sense of direction." In: *Advances in Neural Information Processing Systems* 7, pp. 173–180.

Song, Pengcheng and Xiao-Jing Wang (Jan. 2005). "Angular path integration by moving "hill of activity": A spiking neuron model without recurrent excitation of the head-direction system." In: *Journal of Neuroscience* 25.4, pp. 1002–1014. DOI: 10.1523/jneurosci.4172-04.2005. URL: https://doi.org/10.1523/jneurosci.4172-04.2005.

Stowers, John R. et al. (Aug. 2017). "Virtual reality for freely moving animals." In: *Nature Methods* 14.10, pp. 995–1002. DOI: 10.1038/nmeth.4399. URL: https://doi.org/10.1038/nmeth.4399.

Stringer, S. M. et al. (May 2002). "Self-organizing continuous attractor networks and path integration: one-dimensional models of head direction cells." In: *Network: Computation in Neural Systems* 13.2, pp. 217–242. URL: https://www.ncbi.nlm.nih.gov/pubmed/12061421.

Tolman, Edward C. (1948). "Cognitive maps in rats and men." In: *Psychological Review* 55.4, pp. 189–208. DOI: 10.1037/h0061626. URL: https://doi.org/10.1037/h0061626.

Tsodyks, Misha and Henry Markram (Jan. 1997). "The neural code between neocortical pyramidal neurons depends on neurotransmitter release probability." In: *Proceedings of the National Academy of Sciences* 94.2, pp. 719–723. DOI: 10.1073/pnas.94.2.719. URL: https://doi.org/10.1073/pnas.94.2.719.

Tsodyks, Misha, Klaus Pawelzik, and Henry Markram (May 1998). "Neural networks with dynamic synapses." In: *Neural Computation* 10.4, pp. 821–835. DOI: 10.1162/089976698300017502. URL: https://doi.org/10.1162/089976698300017502.

Turner-Evans, Daniel B. et al. (May 2017). "Angular velocity integration in a fly heading circuit." In: *eLife* 6. Ed. by Alexander Borst, e23496. ISSN: 2050-084X. DOI: 10.7554/eLife.23496. URL: https://doi.org/10.7554/eLife.23496.

Turner-Evans, Daniel B. et al. (2020). "The neuroanatomical ultrastructure and function of a biological ring attractor." In: *Neuron* 108.1, 145–163.e10. ISSN: 0896-6273. DOI: https://doi.org/10.1016/j.neuron.2020.08.006. URL: https://www.sciencedirect.com/science/article/pii/S0896627320306139.

Tuthill, John C. (Sept. 2009). "Lessons from a compartmental model of a Drosophila neuron." In: *Journal of Neuroscience* 29.39, pp. 12033–12034. DOI: 10.1523/jneurosci.3348-09.2009. URL: https://doi.org/10.1523/jneurosci.3348-09.2009.

Urbanczik, Robert and Walter Senn (Feb. 2014). "Learning by the dendritic prediction of somatic spiking." In: *Neuron* 81.3, pp. 521–528. DOI: 10.1016/j.neuron.2013.11.030. URL: https://doi.org/10.1016/j.neuron.2013.11.030.

Wang, Xiao-Jing (Aug. 2001). "Synaptic reverberation underlying mnemonic persistent activity." In: *Trends in Neurosciences* 24.8, pp. 455–463. DOI: 10.1016/s0166-2236(00)01868-3. URL: https://doi.org/10.1016/s0166-2236(00)01868-3.

– (Dec. 2002). "Probabilistic decision making by slow reverberation in cortical circuits." In: *Neuron* 36.5, pp. 955–968. DOI: 10.1016/s0896-6273(02)01092-9. URL: https://doi.org/10.1016/s0896-6273(02)01092-9.

Wilson, Rachel I. (July 2013). "Early olfactory processing in Drosophila: Mechanisms and principles." In: *Annual Review of Neuroscience* 36.1, pp. 217–241. DOI: 10.1146/annurev-neuro-062111-150533. URL: https://doi.org/10.1146/annurev-neuro-062111-150533.

Xie, Xiaohui, Richard H. R. Hahnloser, and H. Sebastian Seung (Oct. 2002). "Double-ring network model of the head-direction system." In: *Physical Review E* 66.4, p. 041902. DOI: 10.1103/physreve.66.041902. URL: https://doi.org/10.1103/physreve.66.041902.

Xie, Xiaohui and H. Sebastian Seung (2000). "Spike-based learning rules and stabilization of persistent neural activity." In: *Advances in Neural Information Processing Systems* 12, pp. 199–208.

Xu, C. Shan et al. (Jan. 2020). "A connectome of the adult Drosophila central brain." In: *bioRxiv*. DOI: 10.1101/2020.01.21.911859. URL: https://doi.org/10.1101/2020.01.21.911859.

Zhang, Kechen (Mar. 1996). "Representation of spatial orientation by the intrinsic dynamics of the head-direction cell ensemble: A theory." In: *Journal of Neuroscience* 16.6, pp. 2112–2126. DOI: 10.1523/jneurosci.16-06-02112.1996. URL: https://doi.org/10.1523/jneurosci.16-06-02112.1996.

Zhao, Chang et al. (Mar. 2021). "Predictive olfactory learning in Drosophila." In: *Scientific Reports* 11.1. DOI: 10.1038/s41598-021-85841-y. URL: https://doi.org/10.1038/s41598-021-85841-y.

Chapter 3

DISENTANGLING REPRESENTATIONS THROUGH MULTI-TASK LEARNING

Vafidis, Pantelis, Aman Bhargava, and Antonio Rangel (2025). "Disentangling representations through multi-task learning." In: *The Thirteenth International Conference on Learning Representations*. URL: https://openreview.net/forum?id=yVGGtsOgc7.

While Chapters I and II explored biologically plausible learning mechanisms based on compartmentalized neurons and local learning rules, this chapter takes a different approach focusing on representation learning without specific claims about mechanism. Here, we show that neural networks develop generalizable, disentangled representations when trained to perform multiple related tasks in parallel. Although we employ backpropagation through time (BPTT) rather than biologically plausible learning rules, our work addresses fundamental principles that may underlie generalization in both biological and artificial systems: the emergence of robust representations through competence at multiple interrelated tasks.

Remarkably, we find that networks trained on noisy evidence accumulation tasks develop multi-dimensional continuous attractor dynamics similar to those explored in Chapter II. We also provide theoretical guarantees that any system that optimally solves multiple such tasks must encode a disentangled representation of the underlying latent variables in its hidden state. Our experiments with recurrent networks confirm these predictions, demonstrating zero-shot generalization to unseen regions of the input space. These findings suggest that parallel processing of diverse tasks may be a general principle through which both biological and artificial agents develop generalizable world models, irrespective of the learning mechanism that leads to competence at such tasks. Finally, we posit that ideal substrate for such parallel processing in the brain is the cortex, with its massively parallel, largely invariant structure.

Chapter Abstract

Intelligent perception and interaction with the world hinges on internal representations that capture its underlying structure ("disentangled" or "abstract" representations). Disentangled representations serve as world models, isolating latent factors of variation in the world along approximately orthogonal directions, thus facilitating feature-based generalization. We provide experimental and theoretical results guaranteeing the emergence of disentangled representations in agents that optimally solve multi-task evidence accumulation classification tasks, canonical in the neuroscience literature. The key conceptual finding is that, by producing accurate multi-task classification estimates, a system implicitly represents a set of coordinates specifying a disentangled representation of the underlying latent state of the data it receives. The theory provides conditions for the emergence of these representations in terms of noise, number of tasks, and evidence accumulation time, when the classification boundaries are affine in the latent space. Surprisingly, the theory also produces closed-form expressions for extracting the disentangled representation from the model's latent state $\mathbf{Z}(t)$. We experimentally validate these predictions in RNNs trained on multi-task classification, which learn disentangled representations in the form of continuous attractors, leading to zero-shot out-of-distribution (OOD) generalization in predicting latent factors. We demonstrate the robustness of our framework across autoregressive architectures, decision boundary geometries and in tasks requiring classification confidence estimation. We find that transformers are particularly suited for disentangling representations, which might explain their unique world understanding abilities. Overall, our framework establishes a formal link between competence at multiple tasks and the formation of disentangled, interpretable world models in both biological and artificial systems, and helps explain why ANNs often arrive at human-interpretable concepts, and how they both may acquire exceptional zero-shot generalization capabilities.

3.1 Introduction

The ability to construct representations that capture the underlying structure of the world from data, is a hallmark of intelligence. Humans and animals leverage their experiences to construct such faithful representations of the world ("world models," "cognitive maps"), resulting in a near-effortless ability to generalize to new settings (Lake, Salakhutdinov, and Tenenbaum, 2015; Lake et al., 2016). Modern foundation models also display emergent out-of-distribution (OOD) generalization abilities, in the form of zero- or few-shot learning (Brown et al., 2020; Pham et al., 2021; Jia et al., 2021; Oquab et al., 2023); however whether artificial systems learn world models remains unclear. Understanding the conditions under which that occurs is bound to lead to better generalizable systems, and explain why artificial systems often converge to human interpretable, aligned representations of the world (Templeton et al., 2024).

A promising direction towards understanding the construction of world models is abstract, or disentangled representations (Higgins et al., 2017; Kim and Mnih, 2018; Johnston and Fusi, 2023). These two concepts are interrelated yet somewhat distinct. Since there exists some ambivalence about their meaning in the literature, we would like to strictly define them here. We will be using definitions adapted from Ostojic and Fusi (2024):

- An abstract representation of latent factors $x_1, \ldots, x_n$ represents each x_i linearly and approximately mutually orthogonally. Thus, abstractness ensures a simple linear map can decode each x_i regardless of variation in $x_{j \neq i}$.
- Disentangled representations of $x_1, \ldots, x_n$ encode each x_i orthogonally, without the necessity of linearity.

Both abstact and disentangled representations preserve the latent structure present in the world in their geometry by isolating *factors of variation* in the data, which facilitates downstream generalization. When a representation is abstract, a linear decoder (i.e., downstream neuron) trained to discriminate between two categories can readily generalize to stimuli not observed in training, due to the structure of the representation. Furthermore, the more disentangled the representation is, the lower the interference from other variables and hence the better the performance. This corresponds to decomposing a novel stimulus into its familiar features, and performing feature-based generalization. For instance, imagine you are at a grocery

store, deciding whether a fruit is ripe or not. If the brain's internal representation of food attributes (ripeness, caloric content, etc.) is disentangled, then learning to perform this task for bananas would lead to zero-shot generalization to other fruit (e.g., mangos, fig. 3.1a). Crucially, the visual representation of a mango is high-dimensional, non-linear and noisy, making it particularly challenging to extract a low dimensional latent like "ripeness."

Several brain areas including the amygdala, prefrontal cortex, and hippocampus encode variables of interest in an abstract format (Saez et al., 2015; Bernardi et al., 2020; Boyle et al., 2022; Nogueira et al., 2023; Courellis et al., 2024). This raises the question of under which conditions do such representations emerge in biological and artificial agents alike. Previous work showed that feedforward neural networks develop abstract representations when trained to multitask (Johnston and Fusi, 2023). However, real-world decisions typically rely on imperfect, noisy information, evolving dynamically over time (Britten et al., 1992; Krajbich, Armel, and Rangel, 2010). To account for this important feature of the world, we train autoregressive models (RNNs, LSTMs, transformers) to multitask canonical neuroscience tasks involving the accumulation of evidence over noisy streams. The tasks tie closely to Bayesian filtering theory, and should be solved by any agent that deals with a noisy world.

Contributions. Our main contributions are the following:

- We prove that any optimal multi-task classifier is guaranteed to learn an abstract representation of the ground truth contained in the noisy measurements in its latent state, if the classification boundary normal vectors span the input space (Appendix C.6). Furthermore, the representations are guaranteed to be disentangled as the number of tasks N_{task} greatly exceeds the input dimensionality D. Intriguingly, noise in the observations is necessary to guarantee the latent state would compute a disentangled representation of the ground truth.

- We confirm that RNNs trained to multitask develop abstract representations that zero-shot generalize OOD, when $N_{\text{task}} \geq D$, and orthogonal, disentangled representations for greater N_{task}. The computational substrate of these representations is a 2D continuous attractor (Amari, 1977) storing a ground truth estimate in a product space of the latent factors. In addition, the representations are sparse and mixed, attributes of biological neural networks.

- We reproduce these findings in GPT-2 transformers, which generalize better due to them learning disentangled representations already from $N_{\text{task}} \geq D$, confirming their appropriateness for constructing disentangled world models.
- We demonstrate that our setting is robust to a number of manipulations, including correlated inputs, interleaved learning of tasks and free reaction-time tasks canonical in the cognitive neuroscience literature (Britten et al., 1992; Krajbich, Armel, and Rangel, 2010).
- Finally, we discuss implications for generalizable representation learning in biological and artificial systems, and demonstrate the strong advantage of multi-task learning over previously proposed mechanisms of representation learning in the brain (Mante et al., 2013).

Although framed in the context of canonical neuroscience tasks, our results are general; they apply to any system aggregating noisy evidence. While our experiments focus on supervised multi-task learning for tractability, the theory only assumes **competence** at multiple tasks, thus enabling alternative methods of acquiring such competence, such as self-supervised or unsupervised pre-training.

Related work

Disentanglement has long been recognized as a promising strategy for generalization (Bengio, Courville, and Vincent, 2012) (although note Locatello et al. (2019) and Montero et al. (2020) for a contrarian view), yet most classic work focuses on feedforward architectures (Higgins et al., 2017; Kim and Mnih, 2018; Whittington et al., 2022; Maziarka et al., 2023). In autoregressive models, Hsu, Zhang, and Glass (2017) and Li and Mandt (2018) showed that variational LSTMs disentagle representations of underlying factors in sequential data allowing style transfer; however the underlying representational geometry was not characterised. Other work focuses on fitting RNNs to behavioral data while enforcing disentanglement for interpretability (Dezfouli et al., 2019; Miller et al., 2023). Work on context-dependent decision making has shown that RNNs re-purpose learned representations in a compositional manner when trained in related tasks (Yang et al., 2019; Driscoll, Shenoy, and Sussillo, 2022); however, the abstractness of the resulting representations was not established. Finally John et al. (2018) show that multitasking results in disentanglement, however unlike us they directly enforce latent factor separation through their adversarial optimization objectives. Our approach is most closely related to

weakly supervised disentanglement, without comparing across samples (Shu et al., 2019).

Our work relates to previous work on linear identifiability. Geoffrey Roeder (2021) show that representations of models trained on the same distribution must be linear transformations of each other; yet we go beyond their results to show that abstract representations are **guaranteed** to emerge under moderate conditions, irrespectively of the dimensionality of the input and model architecture. Lachapelle et al. (2023) proved that disentangled representations emerge in feedforward architectures from multitask learning in sparse tasks when a sparsity regularization constraint is placed on the predictors; we place no such constraints and still uncover disentangled representations.

Previous neuroscience-inspired work showed that multitasking feedforward networks learn abstract representations, as quantified by regression generalization (Johnston and Fusi, 2023). We expand upon these findings in several ways. First, we extend the framework to autoregressive architectures (RNNs, LSTMs, transformers) that can update their representations as further information arrives. Second, we prove theorems that guarantee the emergence of abstract representations **in any optimal multitask classifier** if the number of tasks exceeds the input dimensionality D, and showcase disentanglement in our trained networks. Third, we rigorously analyze the role of noise in forming disentangled representations, extending the noise-free regime studied in Johnston and Fusi (2023). Finally, we explore a range of values for D, providing experimental validation of our theory.

3.2 Problem formulation

Multi-Task Classification with Evidence-Aggregation: We study the evidence accumulation multi-task classification paradigm shown in Figure 3.1b. An agent with latent state $\mathbf{Z}(t)$ receives noisy, non-linearly mapped observations $\{f\left(\mathbf{X}(t)\right)\}_{t=1}^{T}$ where each $\mathbf{X}(t) = \mathbf{x}^* + \sigma\,\mathcal{N}(0, I_D)$ is a noisy measurement of unknown ground truth vector $\mathbf{x}^* \in \mathbb{R}^D (x_i^* \sim \mathit{Uniform}(-0.5, 0.5))$, $\mathcal{N}$ being Gaussian noise. The noisy measurements are transformed by an injective observation map f, which can be non-linear and high dimensional, representing the wide range of sensory transformations found in real-world scenarios. The agent is tasked with simultaneously solving N_{task} classification problems by accumulating information over time (a canonical neuroscience task (Britten et al., 1992)), each defined by a random linear decision

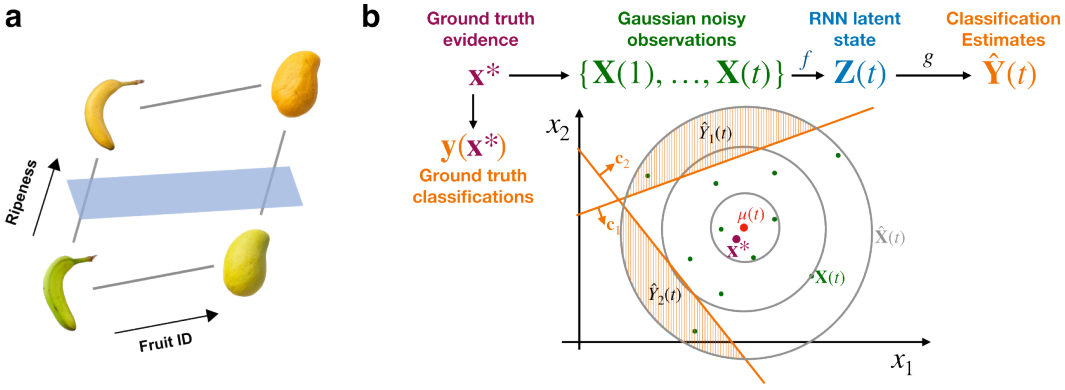

Figure 3.1: **Disentangled representations and a framework to learn them. (a)** A disentangled representation directly lends itself to OOD generalization: a downstream linear decoder that can differentiate ripe from unripe bananas can readily generalize to mangos, even though it has never been trained on mangos. **(b)** Overview of our multi-task classification framework. A ground truth $\mathbf{x}^*$ is sampled and Gaussian noise is added to arrive at observations $\{\mathbf{X}(1), ..., \mathbf{X}(t)\}$. These observations are processed by the filter-based model illustrated graphically in Figure C.6, maintaining a latent state $\mathbf{Z}(t)$. The latent state $\mathbf{Z}(t)$ is then used to produce classification outputs $\hat{Y}_1(t), \hat{Y}_2(t)$. Theorem C.6.6 proves that $\mathbf{Z}(t)$ must encode the optimal estimator of $\mathbf{x}^*$ given the noisy observations, $\mu(t)$.

boundary[1] in the ground truth space $\mathbb{R}^D$, i.e.,

$$y_i(\mathbf{x}^*) = \begin{cases} 1 & \text{if } \mathbf{c}_i^\top \mathbf{x}^* > b_i \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

where $\mathbf{y}(\mathbf{x}^*) \in \{0,1\}^{N_{task}}$ represents the N_{task} classifications of $\mathbf{x}^*$, $\{(\mathbf{c}_i, b_i)\}_{i=1}^{N_{task}}$ are the classification boundary normal vectors and offsets, and let $\hat{\mathbf{Y}}(t) = g(\mathbf{Z}(t)) \in [0,1]^{N_{task}}$ represent the agent's predicted likelihood of $y_i(\mathbf{x}^*) = 1$ over each of the binary classifications i at time t. The classification lines reflect criteria based on which decisions will be made. Imagine for example that x_1 corresponds to food and x_2 to water reward. Depending on the agent's internal state, one takes precedence over the other, and the degree of preference is reflected in the slope of the line.

Criterion for Disentangled Representation Learning: We investigate how solving the multi-task classification problem (Figure 3.1b) leads to agents learning disentangled representations of the latent ground truth $\mathbf{x}^*$ in its internal state $\mathbf{Z}(t)$. Specifically, we ask whether there exists a linear-affine transformation $(\mathbf{A}, \mathbf{b})$ such

[1]Due to the observation map f, the tasks may appear non-linear from the perspective of the multi-task classification agent.

that $\mathbf{x}^* = \mathbf{A}\mathbf{Z}(t) + \mathbf{b}$. Such a mapping would imply $\mathbf{Z}(t)$ linearly represents $\mathbf{x}^*$. If the rows of $\mathbf{A}$ are approximately orthogonal, the representation is disentangled.

3.3 Theoretical results

Here we provide conditions and guarantees for the emergence of disentangled representations in optimal multi-task classifiers with latent state $\mathbf{Z}(t)$ in the paradigm described in Section 3.2 and Figure 3.1. By "optimal multi-task classifier," we refer to any agent or system whose outputs $\hat{\mathbf{Y}}(t)$ correspond to the correct posterior classification probabilities given the noisy, non-linearly transformed observations; that is, for each task $i = 1, \ldots, N_{task}$

$$\hat{Y}_i(t) = \Pr\left(y_i(\mathbf{x}^*) = 1 \mid f(\mathbf{X}(1)), \ldots, f(\mathbf{X}(t))\right). \tag{3.2}$$

The notion of optimality allows us to make precise statements about the informational content of the agent's internal state since $\hat{\mathbf{Y}}(t) = g(\mathbf{Z}(t))$. Let $\mathbf{C} \in \mathbb{R}^{N_{task} \times D}$ be a matrix where each row is a decision boundary normal vector. Then

Theorem 3.3.1 (Disentangled Representation Theorem)**.** *If $\mathbf{C} \in \mathbb{R}^{N_{task} \times D}$ is a full-rank matrix and $N_{task} \geq D$ and noise $\sigma > 0$, then*

1. *Any optimal estimator of $\mathbf{y}(\mathbf{x}^*)$* ***must encode a finite-sample, maximum likelihood estimate*** *$\mu(t)$ of the ground truth evidence variable $\mathbf{x}^*$ in its latent state $\mathbf{Z}(t)$.*
2. *If the activation function is sigmoid-like, $\mu(t)$ will be* ***linearly decodable from*** *$\mathbf{Z}(t)$, thus implying that $\mathbf{Z}(t)$ contains an abstract representation of $\mu(t)$ (Ostojic and Fusi, 2024).*
3. *The representation is guaranteed to be disentangled (orthogonal) as $N_{task} \gg D$ for random decision boundaries.*

Specifically, $\mu(t)$ is the maximum likelihood estimate (MLE) of $\mathbf{x}^*$ given observations $f(\mathbf{X}(1)), \ldots, f(\mathbf{X}(t))$. A closed-form expression for extracting $\mu(t)$ from $\mathbf{Z}(t)$ if $N_{task} \geq D$ is:

$$\mu(t) = (\mathbf{C}^\top \mathbf{C})^{-1} \mathbf{C}^\top \left(\frac{\sigma}{\sqrt{t}} \Phi^{-1}\big(g(\mathbf{Z}(t))\big) + \mathbf{b} \right) \tag{3.3}$$

where Φ is the CDF of the normal distribution, σ is the noise magnitude and t the trial duration. Furthermore, if the activation function g is of the sigmoid family

*of functions (*tanh, sigmoid*), then the term* $\Phi^{-1}(g(\cdot))$ *approximately cancels out, leading to:*

$$\mu(t) \approx \underbrace{\frac{a_g\,\sigma}{\sqrt{t}}(\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\mathbf{Z}(t)}_{\textit{Linear Function of } \mathbf{Z}(t)} + \underbrace{(\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\mathbf{b}}_{\textit{Affine Term}} \tag{3.4}$$

where we have approximated $a_{\tanh} = \frac{2\sqrt{3}}{\pi}$ *for* $g = \tanh$ *and* $a_\sigma = 0.5886$ *for* $g = \text{sigmoid}$*. For Gaussian IID noise,* $\mu(t)$ *is the sample mean of* $\{\mathbf{X}(t)\}_{t=1}^{T}$*, i.e., with non-linearity* f *removed.*

Proof. Point 1 and Equation 3.3 are proven in Appendix C.6 in Theorem C.6.6. Point 2 and Equation 3.4 are proven in Corollary C.6.8 for tanh and Corollary C.6.9 for sigmoid. Point 3 is proven in Corollary C.6.10. □

The key conceptual insight in the proof of Theorem 3.3.1 is that each of the multi-task classification probability estimates $\hat{Y}_i(t)$ represents an estimated projection distance between the MLE $\mu(t)$ and the given classification boundary $(\mathbf{c}_i, b_i)$. Once distances to classification boundaries are recovered, $\mu(t)$ can be inferred if the N_{task} classification boundaries span the D-dimensional space of $\mathbf{x}^*$.

Robustness of results While theorem 3.3.1 applies to optimal multi-task classifiers, Corollary C.6.7 shows that a sub-optimal multi-classifier with zero-mean independent errors will represent $\tilde{\mu}(t)$ in state $\mathbf{Z}(t)$ (Equation 3.3) with residual errors w.r.t. optimal $\mu(t)$ expected to decrease at a rate of approximately $O(1/\sqrt{N_{task}})$. See Appendix C.6, C.6.9 for extensions of the theory to anisotropic and non-Gaussian noise distributions (Elliptical, t-distribution, Laplace distributions). The linear approximation for decoding $\mu(t)$ from $\mathbf{Z}(t)$ in Equation 3.4 is enabled by the remarkable similarity between sigmoid functions and the Gaussian CDF Φ (Corollary C.6.8). The sigmoid-like structure of Φ suggests many similar activations g (e.g., softmax) would exhibit approximate linear decodability.

More general decision boundaries Decision boundaries y_i on latents $\mathbf{x}^*$ may appear non-linear in the image of observation map f, but theorem 3.3.1 applies to linear boundaries y_i on latent space (Eq. 3.1). Our results extend naturally to smooth manifold decision boundaries through local linearization when the manifold y_i's reach τ_i[2] is much larger than the noise scale σ. Intriguingly, classification

[2] "Reach": maximum distance at which each point on a manifold has a unique closest point on the manifold

boundary distances are only guaranteed to be recoverable when there is non-zero noise $\sigma > 0$ such that $\hat{Y}_i(t)$ does not saturate to 1 or 0, and thus still carries useful decision boundary distance information (see Lemma C.6.3) [3]. An intriguing open question is what conditions on manifolds $\{y_i\}_{i=1}^{N_{\text{task}}}$ are necessary and sufficient to preserve the decodability of $\mu(t)$. We leave a complete characterization of representation learning with multiple manifold decision boundaries for future work.

3.4 Methods

Network architecture

We trained autoregressive models (RNNs, LSTMs, GPT-2 transformers) with latent state $\mathbf{z}(t)$, to output multi-task classifications $\mathbf{y}(\mathbf{x}^*) \in \{0, 1\}^{N_{task}}$ given noisy and non-linearly mapped inputs $f(\mathbf{X}(1)), \ldots, f(\mathbf{X}(t))$ (fig. 3.2). We subsequently trained linear probes $\mathbf{A}$ on $\mathbf{z}(t)$ to estimate $\mathbf{x}^*$, denoted $\hat{\mu}(t) = \mathbf{A}\,\mathbf{z}(t)$. We here focus on leaky RNNs, representing a brain area making decisions; for more details on GPT-2 experiments see Section 3.4. The networks contain N_{neu} neurons, and their activations $\mathbf{z}(t)$ obey:

$$\tau \dot{\mathbf{z}} = -\mathbf{z} + [\, \mathbf{W}_{rec}\, \mathbf{z} + \mathbf{W}_{in}\, \mathbf{x}_{in} + \mathbf{b}\,]_+ \tag{3.5}$$

where $\mathbf{W}_{rec}$ is the recurrent weight matrix, $\mathbf{W}_{in}$ is the matrix carrying the input vector $\mathbf{x}_{in}$, $\mathbf{b}$ is a unit-specific bias vector, τ is the neuronal time constant, $[.]_+$ is the ReLU applied element-wise and time dependencies have been dropped for brevity. We discretize Equation 3.5 using the forward Euler method for $T = 20$ timesteps of duration $\Delta t = \tau = 100\,\text{ms}$, which we find to be stable. The RNN's output $\hat{\mathbf{y}}(t) \in \mathbb{R}^{N_{task}}$ is given by $\hat{\mathbf{y}}(t) = g(\mathbf{W}_{out}\, \mathbf{z}(t))$, where $\mathbf{W}_{out}$ is a readout matrix and g = sigmoid the output activation function applied elementwise. The encoder f is a 3-layer MLP with hidden dimensions 100, 100, 40 and ReLU non-linearities, and it is randomly initialized and kept fixed during training as it represents a static mapping from latents to observations (observation map). An additional fixation input is directly passed to the hidden layer. It is 1 during the trial and turns 0 at the end of the trial, indicating that the network should report its decisions (fig. 3.2b). The fixation input is concatenated with $f(\mathbf{X}(t))$ to form $\mathbf{x}_{in}$, and it precludes the RNN from learning a specific timing in its response. We refer to this kind of tasks as fixed reaction-time (RT). The network is trained with a cross-entropy loss and

[3] In fact, Equations 3.3 and 3.4 do not hold for $\sigma \to 0$, as they were derived by means of Bayesian estimation which assumes the presence of noise.

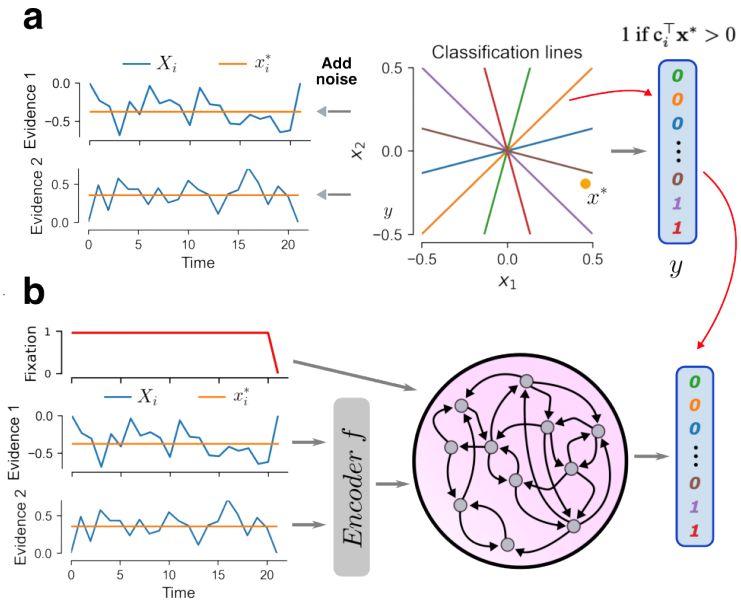

Figure 3.2: **Data generation and architecture.** **(a)** For each trial, we sample a ground truth vector $\mathbf{x}^*$, and add IID noise to arrive at $\mathbf{X}(t)$. The task is to report whether $\mathbf{x}^*$ lies above (1) or below (0) each of the classification lines (color matches corresponding boolean variable in y), given the noisy and non-linearly transformed samples $f(\mathbf{X}(1)), \ldots, f(\mathbf{X}(t))$. **(b)** Models (RNN depicted) are trained to report the outcome of all the binary classifications in **a** at the end of the trial (indicated by the fixation input turning 0).

Adam default settings, except learning rate $\eta_0 = 10^{-3}$, to produce the target outputs $\mathbf{y}(\mathbf{x}^*)$. By minimizing loss across trials, the network is incentivized to estimate $\hat{\mathbf{Y}}(t) = \Pr\{y_i(\mathbf{x}^*) = 1\}$. Table 3.1 summarizes all hyperparameters and their values, which are shared across all architectures.

Quantification of generalization performance

To assess OOD generalization performance, we keep the trained networks fixed and train a linear decoder $\mathbf{A}$ to predict the ground truth $\mathbf{x}^*$ from network activity at the end of the trial. We train the decoder in 3 out of 4 quadrants and test OOD in the remaining quadrant, repeating this process 5 times for each quadrant, which results in a total of 20 OOD r^2 values for each network. To account for randomness in initialization and sythetic generation of datasets, we train 5 networks for each combination of number of tasks N_{task} and dimensionality D, resulting in a total of 100 OOD r^2 values for each pair of (N_{task}, D). We report the 25, 50 (median) and 75 percentiles of those values in fig. 3.3a,b and throughout the text. For ID generalization performance, we train on all quadrants and test in one quadrant at a

Table 3.1: Hyperparameter values for RNN training

Parameter	Value	Explanation
Δt	100 ms	Euler integration step size
τ	100 ms	Neuronal time constant
N_{neu}	64	Number of hidden neurons
σ	0.2	Input noise standard deviation
T	20	Trial duration (in Δts)
η_0	0.001/0.003	Adam learning rate fixed/free RT
B	16	Batch size
N_{batch}	10^5	Number of training batches
D	2	Dimensionality of latent space
N_{layer}	1	RNN/LSTM number of layers

Hyperparameter values for RNN training. These values apply to all simulations, unless otherwise stated. $\tau = 100ms$ was chosen as a conservative estimate of membrane time constant. σ was varied in some simulations (e.g., fig. 3.7c). We found that free RT tasks benefited from a higher learning rate. Other hyperparameters worked out of the box.

time. For input dimensionality $D > 2$, we keep the same logic by choosing every 4-th quadrant to be sampled only in testing, repeating the process for every $mod\,4$ group of quadrants.

Estimation of angles between latent factors

To estimate the angles between latent factors in the representation, we obtain the normal vectors of the decoders $\mathbf{A}$ for each of the latents, and compute pairwise angles for all of them. To account for variability in the decoder fits, we repeat the decoder fit 5 times for each out-of-distribution region (see Section 3.4 for details). We also repeat this process across 5 trained networks for each combination of (N_{task}, D), and report the 25, 50 (median) and 75 percentiles of all values for each (N_{task}, D) combination in fig. 3.3c and fig. 3.8.

Derivation of theoretical r^2 for optimal multi-task classifiers

Here we derive the theoretical r^2 for the estimation of ground truth $\mathbf{x}^*$ from noisy data for a discrete time optimal multi-task classifier at time t. r^2 is defined as:

$$r^2 = 1 - \frac{MSE(\mathbf{x}^*, \mu)}{Var(\mathbf{x}^*)} \tag{3.6}$$

where μ, the mean of $\mathbf{X}(1), \ldots, \mathbf{X}(t)$, is the prediction of the multi-task classifier (see Appendix C.6). The optimal estimator of $\mathbf{x}^*$ given observations $\mathbf{X}(1), \ldots, \mathbf{X}(t)$ is denoted $\hat{\mathbf{X}}(t) \sim \mathcal{N}(\mu(t), t^{-1}\sigma^2 I_D)$ where σ is the noise strength. Note that $\mu(t) \to \mathbf{x}^*$ as $t \to \infty$ by the central limit theorem, and $\mu(t)$ is the optimal estimator of $\mathbf{x}^*$ given Gaussian-noised observations. Since the dimensions in both noise and ground truth are independent, we can focus on one dimension at a time, i.e.:

$$r^2 = 1 - \frac{MSE(x_i^*, \mu_i(t))}{Var(x_i^*)} = 1 - \frac{\mathbb{E}[(x_i^* - x_i^* + \mathcal{N}(0, t^{-1}\sigma^2))^2]}{Var(x_i^*)} = 1 - \frac{\sigma^2}{t\, Var(x_i^*)}. \quad (3.7)$$

Remembering that $x_i^* \sim Uniform(-0.5, 0.5)$ it follows that $Var(x_i^*) = \frac{2}{3}0.5^3$, and replacing $\sigma = 0.2$ from table 2.1 we arrive to $r^2 = 1 - \frac{0.48}{T}$ for given trial duration T which we compare to RNN OOD generalization performance in fig. 3.7a.

GPT-2 experiments

We train GPT-2 causal transformers with $d_{model} = N_{neu} = 64$, $N_{layer} = 1$, $N_{head} = 8$ in the multi-task classification task of the main text. The networks receive continuous, noisy and non-linearly mapped inputs $f(\mathbf{X}(1)), \ldots, f(\mathbf{X}(t))$, and should output multi-task classifications $\mathbf{y}(\mathbf{x}^*) \in \{0, 1\}^{N_{task}}$. The output of the network is $\hat{\mathbf{Y}}(t) := g(\mathbf{Z}(t))$, where $\mathbf{Z}(t)$ is the last embedding of the sequence in the last layer and g = sigmoid. Since the input is continuous, we omit the tokenization and embedding steps, and project the input directly to the hidden state with a linear map. Furthermore, since the inputs are IID, we do not include positional encodings. The networks are trained with binary cross-entropy loss for $N_{batch} = 2 * 10^4$ batches, while the rest of the parameters are identical to the fixed RT networks of the main text (table 2.1).

Finding fixed points and linearization of dynamics

To find approximate fixed points of RNN dynamics after training, we follow a standard procedure outlined in Sussillo and Barak (2013). Specifically, we keep network weights fixed, provide no inputs to the network, and instead optimize over hidden activity. Specifically, we penalize any changes in the hidden activity, motivating the network to find stable states of the dynamics in the absence of input, i.e., attractors of the dynamics. This process finds all states of accumulated evidence that can be stored in this network as short-term memory. Network dynamics could then leverage these states to maintain and update the internal representation of the ground truth $\mathbf{x}^*$ on a single trial level, and drive downstream decisions.

Then for every approximate fixed point $\mathbf{z}^f$, we linearize RNN dynamics around it and estimate the eigenmodes which describe how the system behaves in a small region $\delta\mathbf{z}$ around $\mathbf{z}^f$. Specifically, following Sussillo and Barak (2013) and Mante et al. (2013) we take the difference system $\delta\mathbf{z}(t+t_0) = \mathbf{z}(t+t_0) - \mathbf{z}(t_0)$ and linearize it, i.e.,

$$\dot{\delta\mathbf{z}} = \mathbf{F}'(\mathbf{z}^f)\,\delta\mathbf{z} \tag{3.8}$$

where $\dot{\mathbf{z}} = \mathbf{F}(\mathbf{z})$ is the function describing the RNN dynamics and $\mathbf{M} \equiv \mathbf{F}'(\mathbf{z}^f)$ is its Jacobian at $\mathbf{z}^f$. To estimate $\mathbf{F}'(\mathbf{z}^f)$, we let network dynamics run in the absence of inputs for one time step Δt starting from $\mathbf{z}^f$, i.e., $\delta\mathbf{z}(\Delta t) = \mathbf{z}(\Delta t) - \mathbf{z}^f$, and autodifferentiate $\delta\mathbf{z}(\Delta t)$. We then perform eigendecomposition of $\mathbf{M}$ and report the eigenvalues around each approximate fixed point. Eigenvalues near 0 indicate that the difference system $\delta\mathbf{z}(t) = \mathbf{z}(t) - \mathbf{z}^f$ changes slowly over time, i.e., they correspond to "slow" dimensions in network dynamics which can integrate inputs and maintain them over time (continuous attractors) (Amari, 1977; Mante et al., 2013).

3.5 Multi-task learning leads to disentangled representations

We train RNNs to do simultaneous classifications for N_{task} linear partitions of the latent space for $D = 2$ (fig. 3.2a, 6 partitions shown). To quantify the disentanglement of the representations after learning, we evaluate regression generalization by training a linear decoder to predict the ground truth $\mathbf{x}^*$ while network weights are frozen. We perform out-of-distribution 4-fold crossvalidation, i.e., train the decoder on 3 out of 4 quadrants and test in the remaining quadrant (Section 3.4 for details). We also evaluate in-distribution (ID) performance by training the decoder in all quadrants. An example of train and test losses is shown in fig. 3.5f. We find that the network's OOD and ID generalization performance are excellent (median $r^2 = 0.96, 0.97$, respectively, across 5 example networks); therefore the network has learned an abstract representation that zero-shot generalizes OOD. In addition, ID performance increases with the number of tasks N_{task}, and the OOD generalization gap decreases (fig. 3.3a). Performance is identical when choosing a more nonlinear, power-law nonlinearity for the encoder (fig. 3.4). Therefore we conclude that multi-task learning leads to abstract representations in the RNN's hidden layer, when tasks span the latent space.

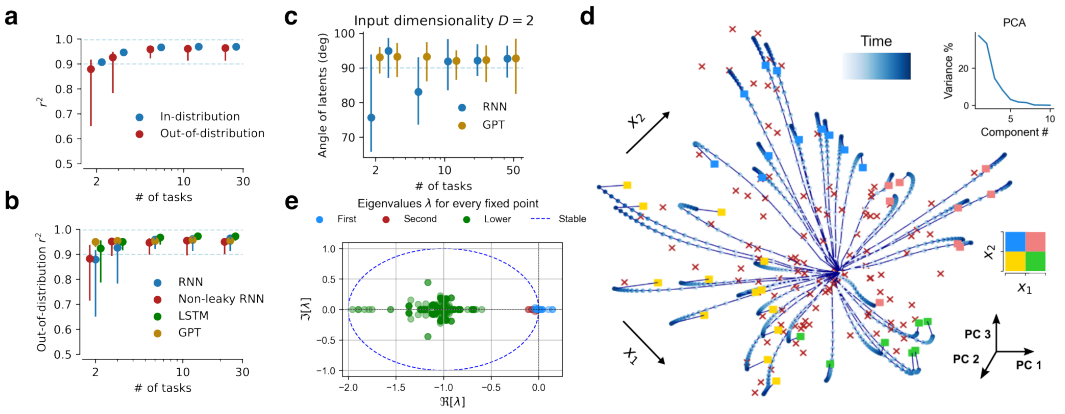

Figure 3.3: **Learning disentangled representations.** **(a)** ID and OOD generalization performance for networks trained in different number of tasks N_{task}. We report the 25, 50 and 75 percentile of r^2 for each network size (see Section 3.4). ID and OOD performance increase with N_{task}, and the generalization gap decreases, indicating that the networks have learned abstract representations. **(b)** The results hold for other autoregressive architectures, including LSTMs and GPT-2 transformers. **(c)** Angles between latent factor decoders (see Section 3.4 for how they were estimated). The angles approach 90 degrees as $N_{\text{task}} \gg D$ for RNNs, but already fror $N_{\text{task}} \geq D$ for transformers. Remaining errors around 90 degrees are attributed to variability in the linear decoder fits. **(d)** Top 3 PCs of RNN activity ($N_{task} = 24$, $D = 2$), capturing 85% of variance (see inset). Each line is a trial, while color saturation indicates time. All trials start from the center and move outwards, towards the location of $\mathbf{x}^*$ in state space. We color the last timepoint in each trial (squares) according to the quadrant this trial was drawn from. Red x's correspond to attractors (see Section 3.4). Here we remove input noise so that trajectories can be visualized easier. The network learns a two-dimensional continuous attractor that provides a disentangled representation of the 2D state space. **(e)** Spectral plot resulting from linearizing RNN dynamics around every fixed point (Section 3.4). First two eigenvalues of the difference system are near 0, while the rest decay much faster, indicating marginal stability across two dimensions for every fixed point, a signature of a 2D continuous attractor.

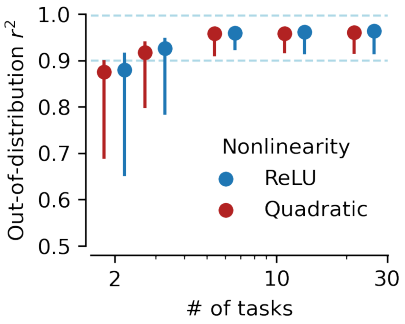

Figure 3.4: **OOD generalization is robust to choice of encoder nonlinearity.** We find that replacing ReLUs in the encoder with a quadratic nonlinearity results in virtually identical OOD generalization performance compared to fig. 3.3a. Therefore we conclude that our setting is robust to the choice of encoder nonlinearity, even when the nonlinearity is not injective, going beyond our theoretical proofs (Appendix C.6).

Since $\mathbf{x}^*$ can be decoded by this representation in unseen (by the decoder) parts of the state space, it follows that the representation can be used to solve **any** task involving the same latent variables, without requiring further pretraining. In other words, to solve any other task we do not need to deal with the denoising and unmixing of the latent factors x_1, x_2; we would just need to learn the (potentially non-linear) mapping from x_1, x_2 to task output. Furthermore, the representation scales linearly with input dimensionality D (see fig. 3.7b). This marks a significant improvement from previously proposed models for representation learning in the brain where one task is executed at a time (Mante et al., 2013; Yang et al., 2019), which scale linearly with N_{task}, and exponentially with D (see Appendix C.1 for details). Crucially, these findings are architecture-agnostic: they hold for non-leaky ("vanilla") RNNs, which outperform leaky ones for small N_{task}, LSTMs which perform the best, and GPT-2 transformers (details in Section 3.4) which have excellent performance already from $N_{task} = 2$ (fig. 3.3b). Note that state-space models have superior asymptotic performance, which is expected due to the nature of the task. We focus on leaky RNNs because of their closer correspondence to biological neurons, which have a membrane voltage that decays over time.

So far we showcased abstractness, but not disentanglement. For disentanglement, it is crucial that the latents lie in orthogonal subspaces. Looking at the angles between the decoders of the latents, we find that they become orthogonal as $N_{\text{task}} \gg D$ for RNNs (fig. 3.3c), as predicted by our theory. Intriguingly, this already occurs from $N_{\text{task}} \geq D$ in transformers, showcasing their superior ability to separate latent

factors. Furthermore, orthogonality strongly correlates with OOD generalization performance, which emphasizes the close link between disentanglement and abstractness: the more orthogonal the representations are, the cleaner the readout of the latent factors by linear decoders.

We further demonstrate the robustness of our setting by showing that abstract representations emerge for different noise distributions and correlated inputs (Appendix C.2), non-linear boundaries (Appendix C.3), and for cognitive neuroscience integrate-to-bound tasks where the agent can make their decision whenever confident enough, not at a fixed time (Krajbich, Armel, and Rangel, 2010) (Appendix C.4).

3.6 Representational structure in RNNs and Transformers

In this section, we open the black box and investigate the representations learned by the networks, starting with RNNs. Figure 3.3d shows the top 3 PCs (capturing ~ 85 % of the variance) of network activity after training (final accuracy $\sim 95\%$) for multiple trials, along with the fixed points of network dynamics. To find the fixed points, we follow a standard procedure outlined in Sussillo and Barak (2013) (see Section 3.4 for details). Looking at fig. 3.3d the fixed points span the entire two-dimensional manifold that the trials evolve in, which corresponds to a continuous attractor with stable states across a 2D "sheet." Linearizing the dynamics around each fixed point and computing the eigenvalues of the linearized system (Section 3.4 for details), reveals marginal stability across two eigenvectors, i.e., near-0 eigenvalues which correspond to slow, integration dimensions in network dynamics, therefore confirming the continuousness of the attractor (fig. 3.3e). This implies that the network can store a short-term memory (Wang, 2001) of the current amount of accumulated evidence in a product space of the latent variables, and update it as further evidence arrives.

Furthermore, compared to the representations after the encoder which are non-linearly mixed, high-dimensional and overlapping (fig. 3.5a), the representation in fig. 3.3d looks disentangled as we would expect from the theory and metrics above. Individual trials with noise show how the representation maintains a sense of metric distances in the RNN representation space (fig. 3.5b). Figure 3.5c demonstrates how this representation comes about during learning, and fig. 3.5d that the short-term memory persists when a delay period is included before the decision. Therefore, multi-task learning has led to disentangled, persistent representations of the latent variables. Importantly, and in line with our theory, this only happens when noise

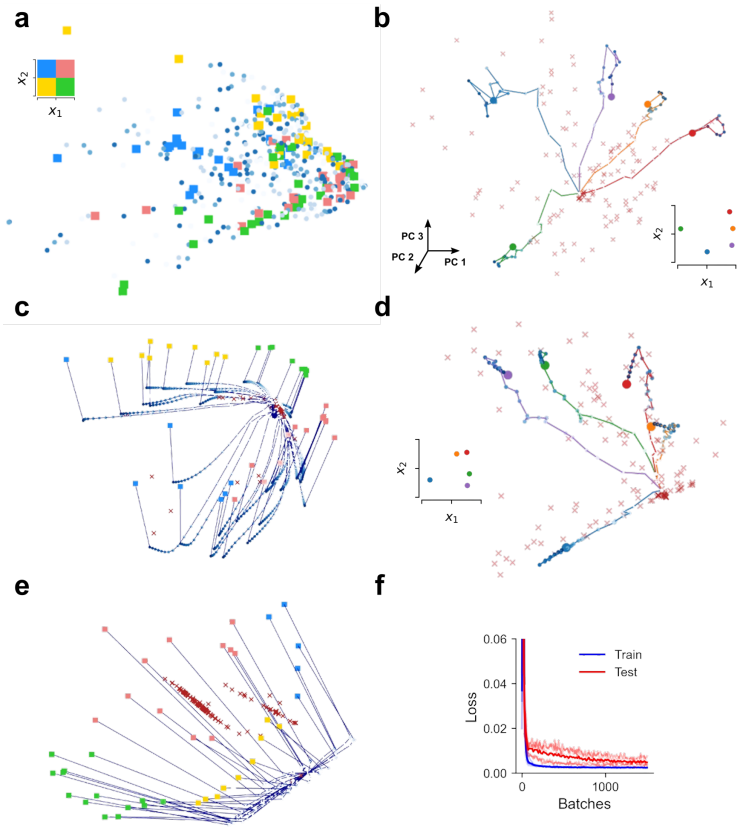

Figure 3.5: **Details of learned representations and of learning. (a)** Representation after the decoder. Compared to fig. 3.3d the representations wrap around non-linearly, and the quadrants are overlapping. The RNN needs to invert the non-linear mapping and denoise to arrive at disentangled representations. **(b)** Individual trial examples from network in fig. 3.2b. Plotting conventions same as in fig. 3.3d, except here every trial has its own color. The ground truth $\mathbf{x}^*$ for all trials is shown in the bottom right. As can be seen, the network maintains a sense of metric distances in the 2D space: examples close in state space are also close in representation space. **(c)** Representation early in learning, for a network trained with 1/4 of the examples compared to fig. 3.3d. The representation is not disentangled yet, however it is visible how the quadrants start separating and the attractors start spreading in the 2D manifold. **(d)** Same as in **b**, but for network with a delay period of 500 ms (5 darker dots at the end of trajectories). Activity remains localized after the removal of the evidence streams, maintaining a short-term memory of the joint evidence with only minor leaks. **(e)** Representation learned in a network trained without input noise ($\sigma = 0$). Trajectories separate from the beginning, and there is no pressure to learn a 2D continuous attractor anymore. **(f)** Train and test errors for linear decoder for the OOD generalization task. Transparent lines correspond to different quadrants while opaque lines to the average across quadrants for one network.

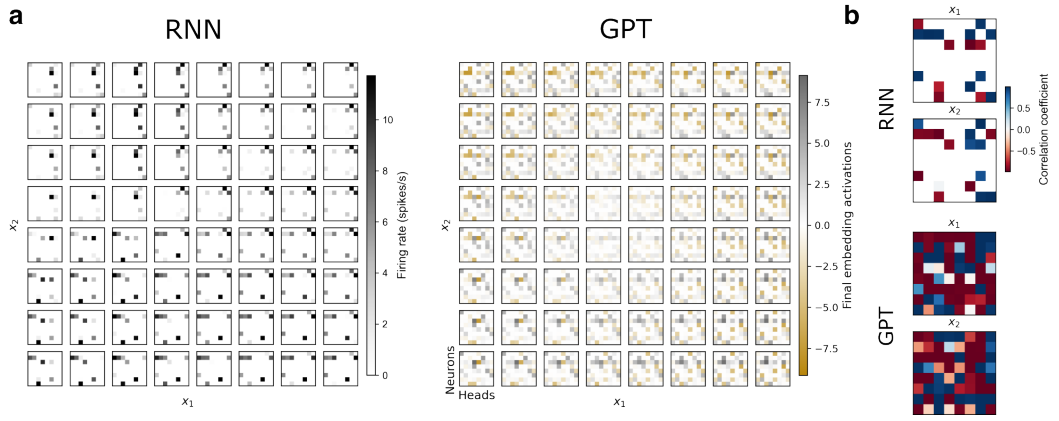

Figure 3.6: **RNN and GPT representations and relation to latent variables. (a)** Hidden layer activations of RNN in fig. 3.2b (left) and GPT-2 transformer (right), while systematically varying the latent factors x_1 and x_2 from -0.5 to 0.5. Activations are plotted in 8*8 grids, one for each value of x_1 and x_2. Each grid contains firing rates for a total of 64 neurons for the RNN, and activations for 8 units for each of the 8 heads from the final embedding of the sequence for GPT-2. **(b)** Correlation coefficient of activations for both models with x_1 and x_2, respectively.

is present in the input, which forces the network to learn a notion of distance from classification boundaries (Lemma C.6.3). Indeed, when the network is trained without input noise, it does not learn a 2D continuous attractor (fig. 3.5e).

Finally, we examined RNN and GPT unit activations, and their relation to the latent variables. In fig. 3.6a we plot activations for all 64 units for both networks, while regularly sampling x_1 and x_2. RNNs representations are sparse, with only $\sim 10\,\%$ of neurons active at any time, which is in line with sparse coding in the brain (see Appendix C.5 for quantification of sparsity as a function of N_{task}, D and RNN architecture). In addition, the average firing rate is ~ 1 spike/s, which is surprisingly close to cortical values. Transformers on the other hand, do not have these features, shared by RNNs and their biological counterparts. Furthermore, we find that both networks display mixed selectivity, i.e., neurons are tuned to both variables, which is a known property of cortical neurons (Rigotti et al., 2013) (fig. 3.6b). This suggests that metrics of disentanglement that assume that individual neurons encode distinct factors of variation (Higgins et al., 2017; Kim and Mnih, 2018; Chen et al., 2018; Eastwood et al., 2022; Hsu et al., 2023) might be insufficient in detecting disentanglement in networks that generalize well. While recent work incorporates such axis-alignment in the definition of disentaglement, our work along with others

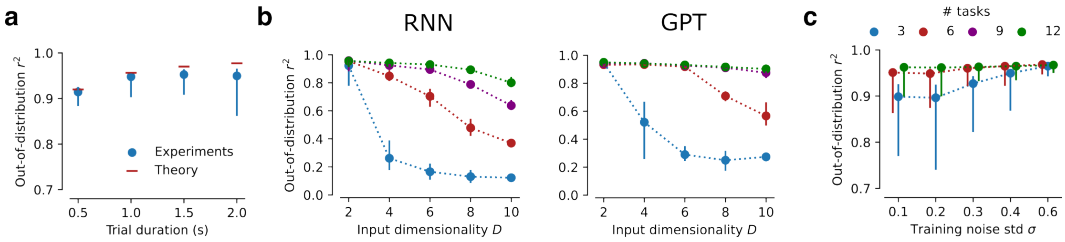

Figure 3.7: **Experiments confirm theoretical predictions.** **(a)** OOD r^2 for free RT RNN required to report its estimate of $\mathbf{x}^*$ at different times (see Appendix C.4 for details, $N_{task} = 24$, $D = 2$). Maximum network r^2 matches optimal multi-task classifier theory predictions (Equation 3.7 in Section 3.4). **(b)** OOD r^2 as a function of input dimensionality D and number of tasks N_{task}. Good values of r^2 are obtained when $N_{task} \geq D$, especially for GPT models, confirming our theoretical results. **(c)** Increasing amounts of noise in pretraining results in better OOD generalization ($D = 2$).

(Johnston and Fusi, 2023) showcases the advantages of approaching disentanglement from a mixed representations perspective. Importantly, these properties were not imposed during training, nor was there any parameter fine tuning involved; they emerged from task and optimization objectives.

3.7 Experiments confirm and extend theoretical predictions

Here we expore the relation between the theory in Section 3.3 and Appendix C.6 and the experiments in the previous sections in more depth. First, we wondered why performance saturates in our networks to a high yet non-1 r^2. The central limit theorem predicts that the estimate of the ground truth $\mathbf{x}^*$ in any optimal multi-task classifier becomes more accurate with $\sqrt{t}$, providing a theoretical maximum r^2 given trial duration T (Section 3.4). Since the RNNs trained on the free reaction-time (free RT) task in Appendix C.4 are required to output their decision confidence at any time in the trial, we can compute OOD r^2 of free RT network predictions at any timepoint t, and compare that to the theoretical prediction. fig. 3.7a shows that indeed the highest RNN r^2 falls in the vicinity of or just short of the theoretical maximum. This indicates that RNNs trained with BPTT on these tasks behave like near-optimal multi-task classifiers that create increasingly accurate predictions with time, tightening the relation between our theoretical and experimental results.

An important prediction of our theory is that to learn abstract representations, N_{task} should exceed D. To test this, we increase D (adding more inputs to fig. 3.2b), while varying N_{task}. Sampling classification hyperplanes homogeneously (similar

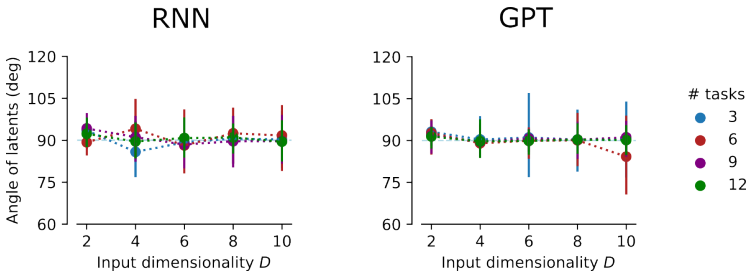

Figure 3.8: **Angles between latent factor decoders in higher dimensions.** Angles converge to 90 degrees as $N_{\text{task}} \gg D$ for RNNs, and as early as $N_{\text{task}} \geq D$ for transformers (see Section 3.4 for angle estimation details). This confirms that multi-task learning leads to orthogonal, disentangled representations, in some cases even earlier than our theoretical guarantees.

to fig. 3.2a, center) in high-dimensional spaces is non-trivial; therefore we resort to randomly sampling them. Figure 3.7b shows OOD generalization performance for various combinations of D and N_{task}. We observe that performance is bad when the $N_{task} < D$, but it increases when $N_{task} \geq D$. For RNNs, this increase is abrupt for smaller D and more gradual for higher, which is in line with remarks by Johnston and Fusi (2023) that it is easier to learn abstract representations when D is high. Transformers on the other hand display higher generalization performance than RNNs, and always perform almost perfectly when $N_{task} \geq D$, demonstrating their superior performance in learning abstract representations. Looking at the angles between latents for higher D (fig. 3.8), we find that transformers have excellent disentanglement as long as $N_{task} \geq D$, which might explain their superior generalization performance to RNNs for lower N_{task}. These results, together with fig. 3.3c demonstrate the superior ability of transformers in disentangling latent factors. Overall, our findings confirm our theory that abstract representations emerge when $N_{task} \geq D$, and even go beyond to suggest that disentangled representations emerge earlier than the theoretical condition $N_{task} \gg D$, as long as the architecture is appropriate. These results are remarkable, especially for high D, because they go against our intuition that N_{task} should scale exponentially with D to fill up the space adequately; instead it need only scale linearly.

Importance of noise for generalization Our theory and experiments provide insight on the importance of noise for developing efficient, abstract representations (fig. 3.5e). The closer to a classification boundary the ground truth $\mathbf{x}^*$ is, the

more likely noise will cross over the boundary. Since, as our theory shows, any optimal multi-task classifier has to estimate $\Pr\{y_i(\mathbf{x}^*) = 1\}$, and said probability directly relates to the actual distance from the boundary, it follows that noise allows the model to learn distances from boundaries (Lemma C.6.3)), leading to efficient localization. We reasoned that additional noise might be even more beneficial, as it would allow more accurate estimation of $\Pr\{y_i(\mathbf{x}^*) = 1\}$, especially when $\mathbf{x}^*$ is far from the boundary. To test this, we increase noise strength σ when pretraining RNNs, while testing with the same $\sigma = 0.2$. Indeed, increasing amounts of noise consistently result in better OOD generalization (fig. 3.7c). This benefit comes for smaller numbers of tasks, allowing us to consider less supervised tasks (e.g., 3), train on them with more noise, and achieve the same performance as more tasks (e.g., 12). So even though networks with more noise perform worse in pretraining (low 90%s classification accuracy), they learn more abstract representations. These findings are highly non-trivial, and have informed our thinking about generalization and inherent variability of the underlying latent factors.

3.8 Implications for representation learning

In this Chapter, we proved that disentangled, generalizable representations **must** emerge in agents optimally solving multi-task evidence accumulation tasks canonical in the neuroscience literature. We also conducted experiments in a suite of autoregressive models (RNNs, LSTMs, transformers) which confirmed all of the main theoretical predictions. A key takeaway is that transformers more readily disentangle representations, which may explain their unique world understanding abilities. Here we discuss the broader impact of this work for representation learning, followed by implications for neuroscience alike, limitations of this study and how it can be extended in the future.

Topology-preserving representation learning Our work has profound implications for learning representations that inherit the topological structure of the world. We prove this naturally happens as long as there are enough tasks to uniquely identify the location of $\mathbf{x}^*$. Crucially, the constraints from different tasks should be placed simultaneously on the representation, which explains why representations from context-dependent computation (Mante et al., 2013) are typically not disentangled.

Representational alignment across individuals Our results provide a new perspective on the Platonic representation hypothesis (Huh et al., 2024), which suggests that the convergence in deep neural network representations is driven by a shared statistical model of reality, like Plato's concept of an ideal "Platonic" reality. Theorem 3.3.1 suggests that the key factor driving convergence is the diversity and comprehensiveness of the tasks being learned. As long as individuals are faced with similar day-to-day tasks that collectively span the space of the underlying data representation, convergence to a shared, reality-aligned representation can occur. This could explain why for example modern LLMs come to encode high-level, human-interpretable concepts (Templeton et al., 2024).

Manifold hypothesis While our problem is framed in terms of arbitrary injective observation map f, the formulation encompasses many scenarios relevant to the manifold hypothesis (Fefferman, Mitter, and Narayanan, 2013). The function f can represent a smooth manifold embedded in a high dimensional space, directly modelling the manifold hypothesis of deep learning. In neuroscience, f could be a non-linear encoding of stimuli in a neural population response, connecting our work to neural manifold research (Langdon, Genkin, and Engel, 2023). By developing and testing theoretical guarantees for the emergence of disentangled representations in this multi-task problem formulation, we provide insight on how neural networks can inherently discover and linearize low-dimensional manifolds within high-dimensional, non-linear observations, enhancing our understanding of how complex data structures are captured and represented in deep learning models and biological systems alike.

Interplay between number of tasks and fine-grainness of representations Finally, the theorem and experimental results presented here are not a one-way-street from dimensionality D of the latents to how many tasks N_{task} are required to uncover them. Instead, there is a fundamental interplay between richness of tasks performed and detail of the representation learned. In a high-dimensional world, the richness of the tasks at hand directly affects the dimensionality D of the latents that can be extracted, allowing for "ground truths" $\mathbf{x}^*$ at different levels of granularity to be explored. The richer the label information available, the more fine-grained the resulting world model will be.

3.9 Implications for neuroscience

The brain encodes variables of interest in a disentangled format, in processes as disparate as memory (Boyle et al., 2022), emotion (Saez et al., 2015), and decision making (Bongioanni et al., 2021). Furthermore, performance in tasks has been shown to degrade once abstract representations collapse (Saez et al., 2015), supporting their role in guiding generalizable behavior. Our findings put forth **parallel processing** as a unifying mechanism for generalization in brains. The cortex, with its massively parallel architecture (Markram et al., 2015; Hawkins et al., 2019), is a prime candidate area for the construction of disentangled, generalizable world models. Another candidate area is the thalamus; it is posited that thalamocortical loops operate in parallel, and combined with internal state-dependent mechanisms lead to state-dependent action selection (e.g., prioritizing water when thirsty), while evidence integration occurs in corticostriatal circuits (Rubin et al., 2020). The representations discovered here (continuous attractors, CANs) have been widely found in the brain when solving similar tasks, highlighting their role as a general computational substrate for cognitive functions in the brain. Notably, the receipt of rich supervisory signals from the environment is not a requirement for our setting, as it can leverage the output of previously learned tasks (see Section 3.10 on the biological plausibility of multi-task learning).

The algorithmic efficiency of multi-task learning compared to alternatives ("context-dependent computation"(Mante et al. (2013), Appendix C.1)), makes us think that it is no coincidence that the cortex can support parallel processing; all the pieces are there, and we feel that the brain has to leverage this feature to construct faithful models of the world, as it does.

Relation to neuroscience literature

An ongoing debate in the brain sciences is whether to solve tasks the brain learns abstracts representations, or simple input-output mappings. Here we show that training RNNs to multitask results in shared, disentangled representations of the latent variables, in the form of continuous attractors. In this multitask setting, one task acts as a regularizer for the others, by not letting the representation collapse, or overfit, to specific tasks (Zhang and Yang, 2017).

Our findings directly link to two important neuroscientific findings: spatial cognition and value-based decision-making. First, the tasks here bear close resemblance to path-integration, i.e., the ability of animals to navigate space only relying on

their proprioceptive sense of linear and angular velocity (Mittelstaedt and Mittelstaedt, 1980; Burak and Fiete, 2009; Vafidis et al., 2022; Sorscher et al., 2023). In path-integration animals integrate velocity signals to get location, while here we integrate noisy evidence to get rid of the noise. In path-integration, networks have to explicitly report distances, while in our setting distances are estimated implicitly (Lemma C.6.3)). We learn abstract representations in the form of a 2D "sheet" continuous attractor, while the computational substrate for path integration is a 2D toroidal attractor (Gardner et al., 2022; Sorscher et al., 2023)—not an abstract representation. The conditions under which a 2D sheet vs. toroidal continuous attractor is learned is a potential area of future research. Second, decision making experiments in monkeys result in a 2D abstract representation in the medial frontal cortex, which supports novel inferential decisions (Bongioanni et al., 2021). Likewise, context-dependent decision-making experiments in humans also resulted in orthogonal, abstract representations (Flesch et al., 2022).

3.10 Biological plausibility of multi-task learning

While our theory stems from parallel processing, i.e., multi-task learning, it is not contingent upon the parallel *execution* of multiple tasks, i.e., multitasking, or the receipt of rich supervisory feedback from the environment in parallel. Behaviorally, the agent need only perform one action, the one most appropriate to its current internal state (e.g., its level of thirst vs. hunger might control the slope of the decision boundary in the 2D latent space of water & food). What we posit is that tasks that have been performed by the agent before and rely on the same input are still resolved somewhere in the brain, by the brain circuits (e.g., cortical columns Hawkins et al. (2019)) previously responsible for them, instead of the entire decision-making brain area focusing only on the current task (Mante et al., 2013). Therefore, the output of these tasks is still placing pressure on the representation, even though they are not actively driving behavior. In other words, our theory assumes **competence** at N_{task} tasks, independently of when and how that competence was achieved. We feel that this is a more natural way of thinking about how the brain manages different tasks, with older tasks still leaving traces somewhere in the brain (Losey et al., 2024); after all, biological agents are remarkable *because* they achieve high performance on many tasks. This theory is also closely related to the widely observed phenomenon of memory replay (Foster and Wilson, 2006), or mental simulation of counterfactuals (Jensen, Hennequin, and Mattar, 2024). A future direction to further enhance the biological relevance of our work would be to investigate the relation between multi-

task learning and slow, interleaved learning (see Appendix C.3), in a continual learning setting.

3.11 Limitations and future directions

A limitation of the present work is that factorization is assumed. Yet not all problems are factorizable, or should be factorized. For instance, a more coarse-grained understanding of the world, that does not disentangle all factors, might be more suitable in many cases, and that might be reflected in the nature of the tasks. Furthermore, we focus on canonical cognitive neuroscience tasks which are somewhat removed from standard ML benchmarks. Normally, disentanglement methods would be tested against a benchmark such as dSprites (Matthey et al., 2017); however to the best of our knowledge no such benchmark exists for sequential tasks where evidence has to be aggregated over time. Future work could endeavor to apply our setting to richer tasks, like extracting latent item attributes from item embeddings when sequential decisions are made in online retailer settings.

Our theory is agnostic to the way by which competence at multiple tasks is achieved. Thus, a natural next step is to investigate whether disentangled representations exist in a wider range of models capable of solving multiple tasks. A prime example is large language models that display excellent zero- and few-shot generalization capabilities, with progress already made in that direction (Templeton et al., 2024). Moreover, the pre-training objective for LLMs (cross entropy loss/likelihood maximization) fits well within our theoretical framing on (approximately) optimal multi-classifiers. Another application area, as already mentioned, is neuroscience; animals are naturally competent at multiple tasks, thus our work provides theoretical justification for why disentangled representations have been found in many brain areas, and motivates looking for more.

Our experiments showed parsimony of our theoretical results under conditions not covered by our theory, including non-injective observation maps (Appendix C.3) and decision boundaries (Appendix C.4) which is encouraging for testing our findings on settings beyond what is strictly covered by the theory. It would be interesting to see how the theoretical insights generalize to different task geometries, for example those implied by self-supervised learning applications (e.g., image patch-filling, next-token prediction, iterative de-noising). The connection between our framework and self-supervised learning is deep and promising. Both frameworks share a common structure, where an underlying latent truth (e.g., objects in an image and their

relationships) is inferred. Each objective (e.g., filling in missing image patches) contributes synergistically to understanding the latent space as a whole. A similar logic applies to predicting words, where the latent "meaning" of a sentence is shared, whether in a causal (e.g., LLMs) or masked setting (e.g., BERT). Our study is a first effort towards understanding such parallel learning, and providing guarantees for its performance.

References

Amari, Shun-ichi (1977). "Dynamics of pattern formation in lateral-inhibition type neural fields." In: *Biological Cybernetics* 27.2, pp. 77–87. DOI: 10.1007/bf00337259. URL: https://doi.org/10.1007/bf00337259.

Bengio, Yoshua, Aaron Courville, and Pascal Vincent (2012). "Representation learning: A review and new perspectives." DOI: 10.48550/ARXIV.1206.5538. URL: https://arxiv.org/abs/1206.5538.

Bernardi, Silvia et al. (Nov. 2020). "The geometry of abstraction in the hippocampus and prefrontal cortex." In: *Cell* 183.4, 954–967.e21. DOI: 10.1016/j.cell.2020.09.031. URL: https://doi.org/10.1016/j.cell.2020.09.031.

Bongioanni, Alessandro et al. (Jan. 2021). "Activation and disruption of a neural mechanism for novel choice in monkeys." In: *Nature* 591.7849, pp. 270–274. DOI: 10.1038/s41586-020-03115-5. URL: https://doi.org/10.1038/s41586-020-03115-5.

Boyle, Lara M. et al. (Jan. 2022). "Tuned geometries of hippocampal representations meet the demands of social memory." DOI: 10.1101/2022.01.24.477361. URL: https://doi.org/10.1101/2022.01.24.477361.

Britten, Kenneth H. et al. (Dec. 1992). "The analysis of visual motion: A comparison of neuronal and psychophysical performance." In: *Journal of Neuroscience* 12.12, pp. 4745–4765. DOI: 10.1523/jneurosci.12-12-04745.1992. URL: https://doi.org/10.1523/jneurosci.12-12-04745.1992.

Brown, Tom B. et al. (2020). "Language models are few-shot learners." DOI: 10.48550/ARXIV.2005.14165. URL: https://arxiv.org/abs/2005.14165.

Burak, Yoram and Ila R. Fiete (Feb. 2009). "Accurate path integration in continuous attractor network models of grid cells." In: *PLoS Computational Biology* 5.2. Ed. by Olaf Sporns, e1000291. DOI: 10.1371/journal.pcbi.1000291. URL: https://doi.org/10.1371/journal.pcbi.1000291.

Chen, Ricky T. Q. et al. (2018). "Isolating sources of disentanglement in variational autoencoders." In: *CoRR* abs/1802.04942. arXiv: 1802.04942. URL: http://arxiv.org/abs/1802.04942.

Courellis, Hristos S. et al. (Aug. 2024). "Abstract representations emerge in human hippocampal neurons during inference." In: *Nature* 632.8026, pp. 841–849. ISSN: 1476-4687. DOI: 10.1038/s41586-024-07799-x. URL: http://dx.doi.org/10.1038/s41586-024-07799-x.

Dezfouli, Amir et al. (2019). "Disentangled behavioural representations." In: *Advances in Neural Information Processing Systems* 32.

Driscoll, Laura, Krishna Shenoy, and David Sussillo (2022). "Flexible multitask computation in recurrent networks utilizes shared dynamical motifs." In: *bioRxiv*, pp. 2022–08.

Eastwood, Cian et al. (2022). "DCI-ES: An extended disentanglement framework with connections to identifiability." DOI: 10.48550/ARXIV.2210.00364. URL: https://arxiv.org/abs/2210.00364.

Fefferman, Charles, Sanjoy Mitter, and Hariharan Narayanan (2013). "Testing the manifold hypothesis." DOI: 10.48550/ARXIV.1310.0425. URL: https://arxiv.org/abs/1310.0425.

Flesch, Timo et al. (2022). "Modelling continual learning in humans with Hebbian context gating and exponentially decaying task signals." DOI: 10.48550/ARXIV.2203.11560. URL: https://arxiv.org/abs/2203.11560.

Foster, David J. and Matthew A. Wilson (Feb. 2006). "Reverse replay of behavioural sequences in hippocampal place cells during the awake state." In: *Nature* 440.7084, pp. 680–683. ISSN: 1476-4687. DOI: 10.1038/nature04587. URL: http://dx.doi.org/10.1038/nature04587.

Gardner, Richard J. et al. (Jan. 2022). "Toroidal topology of population activity in grid cells." In: *Nature* 602.7895, pp. 123–128. DOI: 10.1038/s41586-021-04268-7. URL: https://doi.org/10.1038/s41586-021-04268-7.

Geoffrey Roeder Luke Metz, Diederik P. Kingma (July 2021). "On linear identifiability of learned representations." In: *Proceedings of the 38th International Conference on Machine Learning*. Vol. 139. Proceedings of Machine Learning Research. PMLR, pp. 9030–9039. URL: https://proceedings.mlr.press/v139/roeder21a.html.

Hawkins, Jeff et al. (Jan. 2019). "A framework for intelligence and cortical function based on grid cells in the neocortex." In: *Frontiers in Neural Circuits* 12. ISSN: 1662-5110. DOI: 10.3389/fncir.2018.00121. URL: http://dx.doi.org/10.3389/fncir.2018.00121.

Higgins, Irina et al. (2017). "β-VAE: Learning basic visual concepts with a constrained variational framework." In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=Sy2fzU9gl.

Hsu, Kyle et al. (2023). "Disentanglement via latent quantization." DOI: 10.48550/ARXIV.2305.18378. URL: https://arxiv.org/abs/2305.18378.

Hsu, Wei-Ning, Yu Zhang, and James Glass (2017). "Unsupervised learning of disentangled and interpretable representations from sequential data." DOI: 10.48550/ARXIV.1709.07902. URL: https://arxiv.org/abs/1709.07902.

Huh, Minyoung et al. (2024). "The platonic representation hypothesis." In: *arXiv preprint arXiv:2405.07987*.

Jensen, Kristopher T., Guillaume Hennequin, and Marcelo G. Mattar (June 2024). "A recurrent network model of planning explains hippocampal replay and human behavior." In: *Nature Neuroscience* 27.7, pp. 1340–1348. ISSN: 1546-1726. DOI: 10.1038/s41593-024-01675-7. URL: http://dx.doi.org/10.1038/s41593-024-01675-7.

Jia, Chao et al. (2021). "Scaling up visual and vision-language representation learning with noisy text supervision." DOI: 10.48550/ARXIV.2102.05918. URL: https://arxiv.org/abs/2102.05918.

John, Vineet et al. (2018). "Disentangled representation learning for non-parallel text style transfer." DOI: 10.48550/ARXIV.1808.04339. URL: https://arxiv.org/abs/1808.04339.

Johnston, W. Jeffrey and Stefano Fusi (Feb. 2023). "Abstract representations emerge naturally in neural networks trained to perform multiple tasks." In: *Nature Communications* 14.1. DOI: 10.1038/s41467-023-36583-0. URL: https://doi.org/10.1038/s41467-023-36583-0.

Kim, Hyunjik and Andriy Mnih (July 2018). "Disentangling by factorising." In: *Proceedings of the 35th International Conference on Machine Learning*. Vol. 80. Proceedings of Machine Learning Research. PMLR, pp. 2649–2658. URL: https://proceedings.mlr.press/v80/kim18b.html.

Krajbich, Ian, Carrie Armel, and Antonio Rangel (2010). "Visual fixations and the computation and comparison of value in simple choice." In: *Nature Neuroscience* 13.10, pp. 1292–1298. DOI: 10.1038/nn.2635. URL: https://doi.org/10.1038/nn.2635.

Lachapelle, Sebastien et al. (July 2023). "Synergies between disentanglement and sparsity: Generalization and identifiability in multi-task learning." In: *Proceedings of the 40th International Conference on Machine Learning*. Vol. 202. Proceedings of Machine Learning Research. PMLR, pp. 18171–18206. URL: https://proceedings.mlr.press/v202/lachapelle23a.html.

Lake, Brenden M., Ruslan Salakhutdinov, and Joshua B. Tenenbaum (Dec. 2015). "Human-level concept learning through probabilistic program induction." In: *Science* 350.6266, pp. 1332–1338. DOI: 10.1126/science.aab3050. URL: https://doi.org/10.1126/science.aab3050.

Lake, Brenden M. et al. (Nov. 2016). "Building machines that learn and think like people." In: *Behavioral and Brain Sciences* 40, e253. DOI: 10.1017/s0140525x16001837. URL: https://doi.org/10.1017/s0140525x16001837.

Langdon, Christopher, Mikhail Genkin, and Tatiana A. Engel (Apr. 2023). "A unifying perspective on neural manifolds and circuits for cognition." In: *Nature Reviews Neuroscience* 24.6, pp. 363–377. ISSN: 1471-0048. DOI: 10.1038/s41583-023-00693-x. URL: http://dx.doi.org/10.1038/s41583-023-00693-x.

Li, Yingzhen and Stephan Mandt (2018). "Disentangled sequential autoencoder." DOI: 10.48550/ARXIV.1803.02991. URL: https://arxiv.org/abs/1803.02991.

Locatello, Francesco et al. (2019). "Challenging common assumptions in the unsupervised learning of disentangled representations." In: *International Conference on Machine Learning*. Best Paper Award. URL: http://proceedings.mlr.press/v97/locatello19a.html.

Losey, Darby M. et al. (Apr. 2024). "Learning leaves a memory trace in motor cortex." In: *Current Biology* 34.7, 1519–1531.e4. ISSN: 0960-9822. DOI: 10.1016/j.cub.2024.03.003. URL: http://dx.doi.org/10.1016/j.cub.2024.03.003.

Mante, Valerio et al. (Nov. 2013). "Context-dependent computation by recurrent dynamics in prefrontal cortex." In: *Nature* 503.7474, pp. 78–84. DOI: 10.1038/nature12742. URL: https://doi.org/10.1038/nature12742.

Markram, Henry et al. (Oct. 2015). "Reconstruction and simulation of neocortical microcircuitry." In: *Cell* 163.2, pp. 456–492. ISSN: 0092-8674. DOI: 10.1016/j.cell.2015.09.029. URL: http://dx.doi.org/10.1016/j.cell.2015.09.029.

Matthey, Loic et al. (2017). "dSprites: Disentanglement testing sprites dataset."

Maziarka, Łukasz et al. (2023). "On the relationship between disentanglement and multi-task learning." In: *Machine Learning and Knowledge Discovery in Databases*. Springer International Publishing, pp. 625–641. ISBN: 9783031263873. DOI: 10.1007/978-3-031-26387-3_38. URL: http://dx.doi.org/10.1007/978-3-031-26387-3_38.

Miller, Kevin J. et al. (June 2023). "Cognitive model discovery via disentangled RNNs." DOI: 10.1101/2023.06.23.546250. URL: http://dx.doi.org/10.1101/2023.06.23.546250.

Mittelstaedt, Marie Luise and Holst Mittelstaedt (Nov. 1980). "Homing by path integration in a mammal." In: *Naturwissenschaften* 67.11, pp. 566–567. DOI: 10.1007/bf00450672. URL: https://doi.org/10.1007/bf00450672.

Montero, M. Llera et al. (2020). "The role of disentanglement in generalisation." In: *International Conference on Learning Representations*.

Nogueira, Ramon et al. (Jan. 2023). "The geometry of cortical representations of touch in rodents." In: *Nature Neuroscience* 26.2, pp. 239–250. DOI: 10.1038/s41593-022-01237-9. URL: https://doi.org/10.1038/s41593-022-01237-9.

Oquab, Maxime et al. (2023). "DINOv2: Learning robust visual features without supervision." DOI: 10.48550/ARXIV.2304.07193. URL: https://arxiv.org/abs/2304.07193.

Ostojic, Srjan and Stefano Fusi (July 2024). "Computational role of structure in neural activity and connectivity." In: *Trends in Cognitive Sciences* 28.7, pp. 677–690. ISSN: 1364-6613. DOI: 10.1016/j.tics.2024.03.003. URL: http://dx.doi.org/10.1016/j.tics.2024.03.003.

Pham, Hieu et al. (2021). "Combined scaling for zero-shot transfer learning." DOI: 10.48550/ARXIV.2111.10050. URL: https://arxiv.org/abs/2111.10050.

Rigotti, Mattia et al. (May 2013). "The importance of mixed selectivity in complex cognitive tasks." In: *Nature* 497.7451, pp. 585–590. DOI: 10.1038/nature12160. URL: https://doi.org/10.1038/nature12160.

Rubin, Jonathan E et al. (May 2020). "The credit assignment problem in cortico-basal ganglia-thalamic networks: A review, a problem and a possible solution." In: *European Journal of Neuroscience* 53.7, pp. 2234–2253. ISSN: 1460-9568. DOI: 10.1111/ejn.14745. URL: http://dx.doi.org/10.1111/ejn.14745.

Saez, Alex et al. (Aug. 2015). "Abstract context representations in primate amygdala and prefrontal cortex." In: *Neuron* 87.4, pp. 869–881. DOI: 10.1016/j.neuron.2015.07.024. URL: https://doi.org/10.1016/j.neuron.2015.07.024.

Shu, Rui et al. (2019). "Weakly supervised disentanglement with guarantees." DOI: 10.48550/ARXIV.1910.09772. URL: https://arxiv.org/abs/1910.09772.

Sorscher, Ben et al. (Jan. 2023). "A unified theory for the computational and mechanistic origins of grid cells." In: *Neuron* 111.1, 121–137.e13. DOI: 10.1016/j.neuron.2022.10.003. URL: https://doi.org/10.1016/j.neuron.2022.10.003.

Sussillo, David and Omni Barak (Mar. 2013). "Opening the black box: Low-dimensional dynamics in high-dimensional recurrent neural networks." In: *Neural Computation* 25.3, pp. 626–649. DOI: 10.1162/neco_a_00409. URL: https://doi.org/10.1162/neco_a_00409.

Templeton, Adly et al. (2024). "Scaling monosemanticity: Extracting interpretable features from Claude 3 Sonnet." In: *Transformer Circuits Thread*. URL: https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html.

Vafidis, Pantelis et al. (June 2022). "Learning accurate path integration in ring attractor models of the head direction system." In: *eLife* 11. Ed. by Srdjan Ostojic, Ronald L Calabrese, and Hervé Rouault, e69841. ISSN: 2050-084X. DOI: 10.7554/eLife.69841. URL: https://doi.org/10.7554/eLife.69841.

Wang, Xiao-Jing (Aug. 2001). "Synaptic reverberation underlying mnemonic persistent activity." In: *Trends in Neurosciences* 24.8, pp. 455–463. DOI: 10.1016/s0166-2236(00)01868-3. URL: https://doi.org/10.1016/s0166-2236(00)01868-3.

Whittington, James C. R. et al. (2022). "Disentanglement with biological constraints: A theory of functional cell types." DOI: 10.48550/ARXIV.2210.01768. URL: https://arxiv.org/abs/2210.01768.

Yang, Guangyu Robert et al. (Feb. 2019). “Task representations in neural networks trained to perform many cognitive tasks.” In: *Nature Neuroscience* 22.2, pp. 297–306. ISSN: 1546-1726. DOI: 10.1038/s41593-018-0310-2. URL: https://doi.org/10.1038/s41593-018-0310-2.

Zhang, Yu and Qiang Yang (Sept. 2017). “An overview of multi-task learning.” In: *National Science Review* 5.1, pp. 30–43. ISSN: 2095-5138. DOI: 10.1093/nsr/nwx105. eprint: https://academic.oup.com/nsr/article-pdf/5/1/30/31567358/nwx105.pdf. URL: https://doi.org/10.1093/nsr/nwx105.

Chapter 4

EVALUATING PSYCHOLOGICAL LATENT FACTOR HYPOTHESES THROUGH SELF-CONSISTENCY

This Chapter is a little bit different than the previous ones. While all previous chapters contain mature work, this is a more theoretical idea with little experimental exploration, but potentially big impact in the way we approach the field of psychology.

The exploratory idea discussed in this Chapter is to leverage Large Language Models as computational tools for the field of psychology. While several ideas exist in the realm of prompting, this theoretical framework here proposes a deeper level of exploration, leveraging insights from Chapter III. LLMs are sophisticated models, trained on an immense wealth of data, often rivaling human performance when it comes to performing intensive tasks and finding relations between concepts (from language to mathematics). The core idea is to leverage their associative and "world understanding" (loosely defined) abilities to evaluate psychological latent factor hypotheses (i.e., latent dimensions that can describe someone's psyche), along certain axes. Specifically, I hypothesize that an LLM's uncertainty when evaluating psychological constructs could potentially reflect something about the coherence (or Self-Consistency) of those constructs themselves. Of course, this comes with a lot of practical and conceptual roadblocks that will be discussed throughout the chapter.

The hope of the author is that these ideas form the seed for the serious usage of LLMs as tools and facilitators in the field of psychology, and beyond. This Chapter should be read as an invitation to consider novel methodological possibilities rather than as established research.

4.1 Introduction: Self-Consistency in Philosophical Inquiry

Since the dawn of philosophical inquiry, thinkers have sought to develop theories that embody self-consistency—frameworks of understanding that can coherently explain a multitude of phenomena without breaking down when extended beyond their original domain. This search for self-consistent explanations of reality represents one of the foundational aims of philosophical thought. From Aristotle's systematic categorization of knowledge to Kant's transcendental idealism, to Nietzsche's Will to Power and modern analytical philosophy, the mark of theoretical excellence has consistently been the ability to maintain coherence across varied contexts and applications.

The concept of self-consistency in philosophy extends beyond mere logical non-contradiction. A truly self-consistent theory demonstrates an internal harmony where its principles reinforce rather than undermine one another, and where its explanatory power extends gracefully to new domains. Philosophers have long held that the value of a philosophical theory rests in how accurately and comprehensively it can explain disparate experienced phenomena while maintaining its own internal integrity.

As expressed by Ludwig Wittgenstein in his *Tractatus Logico-Philosophicus*, "The limits of my language mean the limits of my world" (Wittgenstein, 1922). This profound insight suggests that our conceptual frameworks, embodied in language, shape our understanding of reality itself. The quality of these frameworks—their clarity, distinctiveness, and self-consistency—determines their utility in making sense of our experience. And that language is the prime and only mediator of these concepts.

The pursuit of self-consistency has always been essential to philosophical progress, yet it has remained largely qualitative, relying on the careful reasoning and intuition of individual thinkers. The challenge has been a lack of standardized, quantifiable methods for evaluating the self-consistency of theoretical frameworks. Today, with advances in computational methods and artificial intelligence, we stand at a unique historical juncture where we can begin to operationalize and quantify this fundamental philosophical principle. In particular, we nowadays have extremely sophisticated models that operate in the realm of psychology and philosophy—language—and that excel at understanding relations between concepts: Large Language Models (LLMs).

This chapter introduces a novel computational framework for evaluating the self-consistency of psychological latent factor hypotheses, bridging centuries of philosophical inquiry with contemporary computational capabilities. By leveraging Large Language Models (LLMs) as tools for psychological inquiry, we propose a method to quantitatively assess the coherence, distinctiveness, and relevance of psychological latent factor hypetheses—creating a new intersection between philosophy, psychology, and computational science. Promising as it might be, the framework proposed here remains theoretical and would require substantial experimental validation; a direction of current active research.

4.2 From Philosophy to Psychological Latent Factors

While philosophy aims to understand fundamental principles governing reality broadly, psychology focuses more specifically on the nature of human experience, behavior, and mental processes. Within psychology, a central challenge has been developing frameworks that can adequately capture the multidimensional nature of human personality and psychological functioning.

Psychological Latent Factor Hypotheses

Psychological latent factor hypotheses represent attempts to systematically categorize and understand human psychology through a defined set of underlying dimensions or traits. These hypotheses posit that a finite set of latent factors—unobservable psychological constructs—underlie and explain the vast diversity of human psychological experience. Such factors are typically derived through a combination of theoretical reasoning and statistical methods applied to behavioral, cognitive, or self-report data.

A psychological latent factor hypothesis, in our context, consists of:

- A set of latent attributes (factors or traits) that can collectively characterize the psychological makeup of individuals across a population,
- Clear descriptions and definitions of these attributes, and
- An implicit claim that these factors are sufficient, necessary, and optimal for understanding human psychology.

The central question we aim to explore is: How might we potentially evaluate the quality, validity, and utility of different latent factor hypotheses? More specifically,

could we develop methods to assess whether a given set of psychological factors represents a self-consistent and comprehensive framework for understanding human psychology?

Prominent Psychological Latent Factor Models

Several influential latent factor models have been developed in psychology. These models vary in their theoretical foundations, the number of dimensions they propose, and their applications. Among the most prominent are:

The Five-Factor Model (NEO)

The Five-Factor Model, often operationalized through the NEO Personality Inventory (NEO-PI), represents one of the most empirically supported frameworks for understanding personality (McCrae and John, 1992). The model proposes five broad dimensions:

- **Neuroticism**: Tendency toward negative emotions, psychological distress, and emotional instability;
- **Extraversion**: Tendency toward sociability, assertiveness, and positive emotionality;
- **Openness to Experience**: Appreciation for art, emotion, adventure, unusual ideas, imagination, and curiosity;
- **Agreeableness**: Tendency toward compassion, cooperation, and prosocial behavior;
- **Conscientiousness**: Tendency toward organization, self-discipline, and achievement orientation.

The NEO model emerged from lexical studies of personality traits in natural language and was refined through extensive factor analysis. Its strength lies in its robust empirical foundation and cross-cultural validation.

Myers-Briggs Type Indicator (MBTI)

The MBTI, based on Carl Jung's theory of psychological types, identifies four dichotomies (Myers, 1962):

- **Extraversion (E) vs. Introversion (I)**: Orientation of energy toward the external world or the inner world;
- **Sensing (S) vs. Intuition (N)**: Preference for concrete information versus abstract patterns;
- **Thinking (T) vs. Feeling (F)**: Preference for logical analysis versus value-based decision making;
- **Judging (J) vs. Perceiving (P)**: Preference for structure and closure versus flexibility and openness.

Though widely used in professional and educational settings, the MBTI has received criticism regarding its test-retest reliability and construct validity (Pittenger, 1993). Nevertheless, its cultural impact and intuitive appeal have made it a persistent framework in applied psychology.

Alternative Models

Other influential models include:

- **HEXACO**: A six-factor model adding Honesty-Humility to the five factors;
- **Minnesota Multiphasic Personality Inventory (MMPI)**: Focused on clinical dimensions of personality;
- **Eysenck's PEN model**: Proposing Psychoticism, Extraversion, and Neuroticism as core dimensions;
- **Cattell's 16PF**: Offering a more granular 16-dimension approach to personality.

Each model emerges from different methodological approaches and theoretical traditions, leading to variation in both the number and nature of the proposed factors. This diversity raises a critical question: How can we determine which model offers the most useful, accurate, and self-consistent account of human psychology?

4.3 Measuring Self-Consistency: A Thought Experiment

To address the challenge of evaluating the coherence of a psychological latent factor hypothesis, I propose a thought experiment that operationalizes the concept of self-consistency in a measurable way.

The Human Rating Paradigm

Imagine the following scenario: We provide a group of human raters with comprehensive information about a participant—including detailed narratives about their life experiences, behavioral tendencies, cognitive patterns, emotional reactions, and interpersonal dynamics, or even responses to standardized questionnaires (like the NEO). This information is sufficiently rich to paint a complete psychological portrait.

We then ask these raters to evaluate the participant on each dimension of a given psychological latent factor hypothesis (e.g., the five factors of the NEO). Specifically, raters must assign a value on a defined scale (e.g., $1 - 10$) for each latent factor based on their interpretation of the participant's psychological profile.

For a well-defined, self-consistent psychological factor, we would expect high agreement among raters—the distribution of ratings would be sharply concentrated around a specific value. In contrast, for a poorly defined or inconsistent factor, ratings would be widely dispersed, reflecting uncertainty and ambiguity in how the factor applies to the individual, a poorly defined/ambiguous concept, or combinations thereof.

Crucially, when this procedure is repeated across many diverse participants, patterns emerge that inform us not just about individual participants but about the latent factors themselves. The key insight is that while rating distributions for a single participant tell us primarily about how well we can characterize that individual, aggregate patterns across many, diverse participants reveal a fundamental property of the psychological factors themselves—namely, the Self-Consistency of the concept across a diverse group of individuals.

Limitations of Human Raters

While conceptually elegant, implementing this approach with human raters presents significant challenges:

- **Expertise requirements**: Accurate ratings demand sophisticated understanding of psychological constructs;
- **Scale limitations**: Conducting the experiment across hundreds of participants and multiple factor models would require enormous human resources;
- **Consistency challenges**: Human raters may themselves show inconsistency in their application of psychological constructs;

- **Implicit biases**: Human judgments are susceptible to various cognitive biases that may distort ratings;
- **Domain-specific knowledge**: Raters may have uneven familiarity with different psychological domains.

These practical constraints have historically limited our ability to systematically evaluate psychological latent factor hypotheses at scale. However, recent advances in artificial intelligence, particularly in the domain of Large Language Models (LLMs), offer a promising alternative approach.

4.4 Large Language Models as Tools for Psychological Latent Factor Hypothesis Evaluation

Large Language Models (LLMs) present a unique opportunity to operationalize the evaluation of psychological latent factor hypotheses. These computational systems, trained on vast corpora of text, have developed sophisticated capabilities for processing and generating language-based content, including psychological descriptions and assessments.

Why LLMs Are Suited for Psychological Latent Factor Hypothesis Evaluation
Several properties suggest that LLMs might be well-suited for exploring psychological latent factor hypotheses, though limitations exist:

- **Knowledge breadth**: LLMs are trained on extensive psychological literature, including different theoretical traditions;
- **Distributional outputs**: They produce probability distributions over potential responses rather than single-point estimates;
- **Consistent application**: They apply the same inference process across all evaluations;
- **Scalability**: They can process thousands of evaluations efficiently;
- **Linguistic sophistication**: They can interpret nuanced psychological descriptions and apply abstract constructs.

Most importantly, LLMs are fundamentally designed to model probability distributions over language. This property aligns perfectly with our need to assess uncertainty in psychological construct application—the key to measuring self-consistency.

Such uncertainty will be quantified by the sharpness of the resulting distributions over the ratings $(1 - 10)$ for a specific latent factor.

From LLM Uncertainty to Psychological Self-Consistency

The central insight of our approach is that an LLM's uncertainty when rating an individual on a psychological factor can serve as a proxy for that factor's self-consistency as a theoretical construct.

When presented with a comprehensive description of an individual and asked to rate them on a psychological dimension, an LLM produces a probability distribution over possible ratings. The entropy of this distribution directly reflects the model's certainty about how the factor applies to the individual. Low entropy (highly concentrated distribution) indicates high certainty, while high entropy (diffuse distribution) signals uncertainty. Importantly, since the LLM is queried with multiple sources of information, our theoretical findings from Chapter III provide theoretical justification to trust the LLM responses—with sufficient information, we have localization guarantees when it comes to psychological latent factors (see Trilateration Theorem, Theorem C.6.4).

By aggregating these entropy measurements across diverse individuals, we obtain a metric of the factor's overall self-consistency. A truly self-consistent psychological factor should yield consistently low-entropy distributions across many individuals, indicating that the concept is well-defined and can be applied with high certainty across diverse psychological profiles.

This approach leverages what LLMs do best—generating probability distributions over next tokens while drawing on extensive knowledge—to interrogate models of psychological latent factors.

Relation to Existing LLM Literature

Our approach connects to several important developments in the LLM literature while formalizing what has traditionally been a highly empirical domain. The concept of "self-consistency" has already emerged as a technique in prompt engineering and LLM evaluation, though in different contexts from our application.

Wang et al. (2023) introduced self-consistency for mathematical reasoning, showing that sampling multiple reasoning paths and selecting the most consistent answer improves performance. Similarly, Kadavath et al. (2022) demonstrated that LLMs

can evaluate the correctness of their own reasoning. These approaches, however, have been largely empirical and task-specific, lacking theoretical formalization.

Our framework elevates these empirical techniques into a formalized methodology with rigorous mathematical foundations specifically designed for psychological latent factor hypothesis evaluation. While previous work used self-consistency pragmatically to improve single-point answers, we provide a principled formalization through information-theoretic metrics like entropy and KL divergence. This transforms what was previously a heuristic approach into a quantifiable, systematic framework.

The use of LLMs to evaluate coherence of concepts also aligns with recent work on evaluating conceptual understanding in language models (Shanahan, McDonell, and Reynolds, 2023), suggesting that these models have developed sophisticated capabilities for assessing theoretical coherence. Our contribution extends this work on calibration and uncertainty in LLMs (Jiang et al., 2021), showing that model uncertainty can be meaningfully harnessed to evaluate psychological constructs.

By explicitly defining rigorous quantitative metrics, we move beyond subjective assessments of hypothesis quality toward a formalized evaluative framework that can be consistently applied across different psychological theories.

4.5 Formalizing Metrics for Latent Factor Hypothesis Evaluation

To operationalize our approach, I propose a formal mathematical framework for evaluating psychological latent factor hypotheses. This framework includes three core metrics: Self-Consistency Entropy Metric (SCEM), Factor Distinctiveness Metric (FDM), and Relevance Metric (RM), but could be easily extended to include more, or tweeked to use different computational quantities for a specific metric (e.g., correlation for the Distinctiveness metric).

Mathematical Preliminaries

Before introducing the metrics, let us establish notation:

- Let z represent a set of latent psychological factors $\{z_1, z_2, ..., z_M\}$ that constitute a psychological latent factor hypothesis.
- Let $i \in \{1, 2, ..., N\}$ index individuals whose psychological profiles are being evaluated.
- Let $j \in \{1, 2, ..., M\}$ index the latent factors in the hypothesis.

- Let $P^z_{i,j}$ represent the probability distribution over ratings (1 to 10) for factor j applied to individual i, as determined by an LLM.

We now introduce the key measure of information theory that underlies our metrics.

Entropy

The entropy of a probability distribution measures its uncertainty or dispersion. For a discrete distribution P over values $\{1, 2, ..., 10\}$, the entropy is given by:

$$H(P) = -\sum_{k=1}^{10} P(k) \log P(k). \tag{4.1}$$

Lower entropy indicates a more concentrated distribution, reflecting higher certainty.

Kullback-Leibler Divergence

The Kullback-Leibler (KL) Divergence measures the difference between two probability distributions. For distributions P and Q, the KL Divergence is defined as:

$$D_{KL}(P\|Q) = \sum_{k} P(k) \log \frac{P(k)}{Q(k)}. \tag{4.2}$$

Higher KL Divergence indicates greater differentiation between distributions.

Self-Consistency Entropy Metric (SCEM)

The Self-Consistency Entropy Metric quantifies how consistently and precisely a psychological latent factor hypothesis's latent factors can be applied across individuals. It is calculated as the average entropy of rating distributions across all factors and individuals:

$$\text{SCEM}(z) = \frac{1}{N \times M} \sum_{i=1}^{N} \sum_{j=1}^{M} H(P^z_{i,j}) \tag{4.3}$$

where $H(P^z_{i,j})$ is the entropy of the rating distribution for individual i on factor j.

Lower SCEM values indicate higher self-consistency—the psychological factors can be applied with greater precision and certainty across diverse individuals. This metric directly operationalizes the philosophical concept of self-consistency in psychology and philosophy.

Factor Distinctiveness Metric (FDM)

The Factor Distinctiveness Metric evaluates how uniquely distinguishable the different factors within a hypothesis are from one another. It uses the Kullback-Leibler Divergence to measure the differentiation between factors:

$$\text{FDM}(z) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\binom{M}{2}}\sum_{j\neq k} D_{KL}(P_{i,j}^{z} \| P_{i,k}^{z}) \tag{4.4}$$

where $D_{KL}(P_{i,j}^{z} \| P_{i,k}^{z})$ is the KL Divergence between the distributions for factors j and k for individual i, and $\binom{M}{2}$ is the number of factor pairs.

Higher FDM values indicate greater distinctiveness among factors. This metric captures how well a psychological latent factor hypothesis differentiates between its constituent factors, avoiding redundancy and overlap. Distinctiveness relates to the principle of parsimony (Occam's razor) in scientific theory—a theory should use the minimal necessary number of distinct factors to explain phenomena.

Relevance Metric (RM)

The Relevance Metric assesses how applicable the latent factors are across different individuals. It quantifies how many factors have high expected values for each individual:

$$\text{RM}_i(z) = \sum_{j=1}^{M} I\left(\mathbb{E}[P_{i,j}^{z}] > V_{\text{threshold}}\right) \tag{4.5}$$

where $\mathbb{E}[P_{i,j}^{z}]$ is the expected value of the rating distribution, $V_{\text{threshold}}$ is a predefined threshold, and $I(\cdot)$ is an indicator function that equals 1 if the condition is met and 0 otherwise.

The aggregate Relevance Metric is the average across individuals:

$$\text{RM}(z) = \frac{1}{N}\sum_{i=1}^{N}\text{RM}_i(z). \tag{4.6}$$

Higher RM values indicate that the latent factors are consistently relevant across diverse individuals, capturing salient aspects of human psychology. This metric ensures that the psychological latent factor hypothesis includes factors that are meaningfully applicable to a wide range of individuals.

Implementation with Large Language Models

To implement these metrics using LLMs, we:

1. Provide the LLM with comprehensive descriptions of diverse individuals.
2. Ask the model to rate each individual on each factor z_j in a given psychological latent factor hypothesis z.
3. Extract the probability distributions over ratings from the model's output.
4. Calculate SCEM, FDM, and RM using these distributions.

Mathematically, we can represent the LLM as a function $\mathcal{L}$ that maps a description of individual i and a psychological factor j to a probability distribution over ratings:

$$P_{i,j}^{z} = \mathcal{L}(D_i, z_j) \tag{4.7}$$

where D_i is the description of individual i and z_j is the definition of factor j.

The probability distribution is extracted from the logits of the LLM's output layer, representing the model's uncertainty in applying the psychological factor to the given individual.

4.6 Comprehensive Hypothesis Evaluation Framework

While each metric provides valuable information about specific aspects of a psychological latent factor hypothesis z, a comprehensive evaluation requires considering all three metrics together. Rather than collapsing these metrics into a single score, we propose reporting them as a tuple:

$$\text{Evaluation}(z) = (\text{SCEM}(z), \text{FDM}(z), \text{RM}(z)). \tag{4.8}$$

This multi-dimensional approach respects the complexity of psychological phenomena while offering potential comparative standards between different latent factor hypotheses. The practical utility of these metrics for psychological research remains to be seen.

Under this framework, a strong psychological latent factor hypothesis z should demonstrate:

- Low SCEM (high self-consistency);
- High FDM (strong factor distinctiveness);
- High RM (broad relevance).

However, trade-offs between these dimensions may exist. For example, a hypothesis might achieve high self-consistency by sacrificing relevance, or high distinctiveness by sacrificing self-consistency. The tuple representation makes these trade-offs explicit and enables researchers to select theories based on priorities relevant to their specific applications.

It is important to emphasize that our proposed metrics focus primarily on external validity—how well psychological constructs can be consistently applied across diverse individuals and contexts. This approach differs from traditional internal consistency measures that assess reliability within the constructs themselves. What makes our framework valuable is its ability to evaluate how psychological theories perform "in the wild" when applied to varied human experiences.

In clinical psychology, for instance, predictive validity remains crucial—a psychological latent factor hypothesis should effectively forecast treatment outcomes or symptom trajectories. Our metrics complement these predictive approaches by assessing how clearly delineated and applicable the theoretical constructs themselves are. Additional metrics like test-retest reliability, cross-cultural generalizability, and developmental sensitivity provide further dimensions for evaluation. By incorporating our self-consistency framework alongside these established validation techniques, researchers can develop a more comprehensive understanding of a psychological latent factor hypothesis's real-world utility and conceptual strength.

Example Application: Comparing NEO and MBTI

To illustrate the utility of our framework, we can consider how it might be applied to compare prominent psychological theories such as the NEO Five-Factor Model and the Myers-Briggs Type Indicator.

The NEO model, with its empirical foundation in factor analysis, might be expected to show high self-consistency (low SCEM) and high factor distinctiveness (high FDM), as its dimensions were explicitly derived to be orthogonal. However, its relevance (RM) might vary across different populations.

In contrast, the MBTI, with its dichotomous structure, might show lower self-consistency (higher SCEM) due to the forced binary nature of its dimensions. Its factor distinctiveness (FDM) might also be lower due to correlations between dimensions. However, its cultural popularity might contribute to high relevance scores (RM) in certain populations.

Our framework would allow for a systematic, quantitative comparison of these theories based on their performance across all three metrics, potentially revealing strengths and weaknesses that have not been apparent in traditional validation approaches. It could also detect and report correlations between extraneous concepts in lengthier psychological latent factor hypotheses, reflecting potential existing criticisms of such hypotheses in the psychological literature, which would be a good validation step for our theoretical metrics.

4.7 Future Directions and Limitations

The approach presented in this chapter represents a novel theoretical framework for evaluating psychological latent factor hypotheses. While conceptually promising, it would still require empirical validation, and several important directions for future development and limitations must be acknowledged.

Expanding the Framework

Future work could enhance the framework in several ways:

- **Incorporating additional metrics**: Beyond self-consistency, distinctiveness, and relevance, other dimensions such as predictive utility, developmental stability, and cross-cultural applicability could be formalized.
- **Human-guided iterative refinement**: The metrics can be used by psychological experts to iteratively refine theories in a quantitatively guided process—identifying which factors show poor self-consistency, excessive correlation with other factors, or limited relevance, and refining their definitions accordingly. Unsupervised learning in the rich $N \times M$ matrices of the proposed measures would be immensely helpful in that direction, to detect outliers and examples that cannot be covered by a certain hypothesis, and come up with new proposals.

- **Integration with empirical methods**: Combining our computational approach with traditional empirical validation methods could provide a more comprehensive evaluation methodology.

- **Cross-cultural validation**: Testing whether the same latent factor hypotheses maintain self-consistency across different cultural contexts.

Limitations and Potential Mitigation Strategies

Several important limitations warrant consideration, along with potential approaches to address them:

- **LLM limitations and biases**: Current models have inherent limitations in their understanding of complex psychological phenomena and may reflect biases present in their training corpora, which can skew evaluations. These can be mitigated through fine-tuning on balanced, diverse psychological literature and careful prompt engineering.

- **Simplicity bias**: The framework may favor simpler theories that are easier to apply consistently, potentially missing valuable complexity. This can be addressed by explicitly incorporating metrics that reward appropriate complexity where justified.

- **Domain and cultural specificity**: LLMs may have uneven performance across different psychological domains and primarily reflect Western psychological frameworks. Human-in-the-loop verification by experts from diverse backgrounds can help identify and correct these imbalances.

- **Validation challenges**: Determining ground truth for comparison remains challenging. Cross-model validation using different model architectures and training sets can help establish convergent validity.

- **Evolving model capabilities**: As LLMs continue to develop, their evaluations of psychological theories may change, requiring periodic recalibration of the framework.

Ethical Considerations

The application of computational methods to psychological hypothesis evaluation raises important ethical considerations:

- **Algorithmic determinism**: An over-reliance on computational evaluation could lead to reductive views of human psychology.
- **Cultural sensitivity**: Psychological frameworks vary across cultures, and computational evaluations must respect this diversity.
- **Reinforcement of dominant paradigms**: LLMs may reinforce already dominant psychological frameworks in their evaluations.
- **Informed consent**: Use of psychological profiles for evaluation raises questions about consent and privacy.

These considerations underscore the importance of using our framework as one component within a broader, human-centered approach to psychological hypothesis development and evaluation. It is crucial to emphasize that our approach positions LLMs as tools for psychological inquiry, not as replacements for human psychological expertise. The ultimate interpretation of results and hypothesis refinement remains the domain of trained psychologists and researchers.

4.8 Discussion

This chapter has introduced a novel computational framework for evaluating psychological latent factor hypotheses through the lens of self-consistency. By leveraging Large Language Models as tools for psychological inquiry, we have operationalized the philosophical concept of self-consistency in a quantifiable way that can be applied systematically to diverse (psychological) concepts.

The proposed metrics—Self-Consistency Entropy Metric (SCEM), Factor Distinctiveness Metric (FDM), and Relevance Metric (RM)—provide a multidimensional approach to psychological latent factor hypothesis evaluation that respects the complexity of psychological phenomena while offering rigorous comparative standards. Yet, the practical value and implementation of these metrics in LLMs remains to be seen.

This approach represents a new intersection between philosophy, psychology, and computational science, potentially opening new avenues for psychological hypothesis development and refinement. By quantifying aspects of theoretical quality that have historically been evaluated through more subjective means, our framework offers a complementary methodology that may accelerate progress in psychological science.

Importantly, the framework positions Large Language Models not as replacements for human psychological expertise but as tools that extend our capabilities for systematic psychological latent factor hypothesis evaluation. The ultimate interpretation and application of psychological theories remain firmly in the domain of human understanding.

As computational capabilities continue to advance and our understanding of both human psychology and artificial intelligence deepens, frameworks like the one proposed here may help bridge the gap between qualitative theoretical insights and quantitative evaluation methodologies, potentially catalyzing new developments in our understanding of the human mind.

References

Jiang, Zhengbao et al. (2021). "How can we know when language models know? On the calibration of language models for question answering." arXiv: 2012.00955 [cs.CL]. URL: https://arxiv.org/abs/2012.00955.

Kadavath, Saurav et al. (2022). "Language models (mostly) know what they know." arXiv: 2207.05221 [cs.CL]. URL: https://arxiv.org/abs/2207.05221.

McCrae, Robert R. and Oliver P. John (1992). "An introduction to the five-factor model and its applications." In: *Journal of Personality* 60.2, pp. 175–215. DOI: https://doi.org/10.1111/j.1467-6494.1992.tb00970.x. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-6494.1992.tb00970.x. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-6494.1992.tb00970.x.

Myers, Isabel B. (1962). "The Myers-Briggs type indicator: Manual (1962)." Consulting Psychologists Press. DOI: 10.1037/14404-000. URL: http://dx.doi.org/10.1037/14404-000.

Pittenger, David J. (Jan. 1993). "Measuring the MBTI ... and coming up short." In: *Journal of Career Planning and Employment* 54.

Shanahan, Murray, Kyle McDonell, and Laria Reynolds (Nov. 2023). "Role play with large language models." In: *Nature* 623.7987, pp. 493–498. ISSN: 1476-4687. DOI: 10.1038/s41586-023-06647-8. URL: http://dx.doi.org/10.1038/s41586-023-06647-8.

Wang, Xuezhi et al. (2023). "Self-consistency improves chain of thought reasoning in language models." arXiv: 2203.11171 [cs.CL]. URL: https://arxiv.org/abs/2203.11171.

Wittgenstein, Ludwig (1922). "Tractatus logico-philosophicus." In: *London: Routledge, 1981*. Ed. by D. F. Pears. URL: http://scholar.google.de/scholar.bib?q=info:1G2GoIkyCZIJ:scholar.google.com/&output=citation&hl=de&ct=citation&cd=0.

CONCLUSION

This thesis has investigated learning and generalization in neural networks, drawing parallels between biological and artificial systems while seeking to explain cognitive processes from the bottom up, with a particular focus on cortical mechanisms. By examining the computational principles underlying intelligence across multiple levels—from individual neurons to network dynamics and representational structure—the work presented here advances our understanding of how neural systems develop rich, generalizable internal models of the world.

A central focus has been cortical learning and the unique advantages conferred by the architectural and representational inductive biases present in the cerebral cortex. In Chapter I, we demonstrated how the compartmentalized structure of pyramidal neurons enables efficient packing of multiple associations within the same neural population—a capability that explains the exceptional associative learning capacity of mammals with developed cortices. Importantly, we addressed a significant gap in the literature by explicitly testing whether traditional three-factor Hebbian learning rules could achieve similar performance in networks with mixed selectivity. The results conclusively showed that these rules fail at stimulus substitution in mixed representation regimes, highlighting the necessity of predictive learning mechanisms like those enabled by compartmentalized neurons. This finding provides a mechanistic explanation for why evolution may have converged on the specific morphology of cortical pyramidal cells. Furthermore, the model's ability to account for sophisticated conditioning phenomena, including forms of causal reasoning like unconditional support, demonstrates how these architectural features contribute to cognitive capabilities previously thought to require more complex, top-down mechanisms.

This work also establishes how the cortex's massively parallel processing architecture serves as a powerful inductive bias driving the emergence of generalizable representations. Chapter III provided theoretical guarantees that systems competent at multiple related tasks must develop abstract, disentangled representations of underlying latent variables, with representational quality directly tied to task diversity. This insight has profound implications for computational neuroscience, potentially revolutionizing how we understand cortical function. The theoretical finding aligns with empirical observations that performance in cognitive tasks degrades when abstract representations collapse (Saez et al., 2015), suggesting that representational

disentanglement is not merely a computational convenience but a requirement for flexible cognition. Furthermore, the consistent finding that neural networks arrive at similar representations when solving similar tasks, regardless of their specific architecture, points to fundamental principles governing representation formation in both biological and artificial systems.

These insights extend beyond neuroscience to inform artificial intelligence research, particularly in understanding why large language models, which are inherently multitask learners (Brown et al., 2020), develop human-interpretable concepts. In Chapter IV, we exploit this multitask nature of LLMs by proposing a novel framework that leverages their representational properties to disentangle psychological latent factors from behavioral reports of the participants, demonstrating how computational principles discovered in neural systems can inform traditionally human-centered disciplines. This cross-disciplinary application illustrates the broad impact of the theoretical advances presented in this thesis.

Predictive learning objectives emerged throughout this thesis as a powerful mechanism for developing rich, structured representations. The compartmentalized learning rules explored in Chapters I and II implement a form of self-supervised prediction where neurons learn to anticipate their own activity patterns. Looking beyond the mammalian cortex, Chapter II demonstrated that the same predictive learning principles can be applied to understand neural circuit development in evolutionarily distant organisms like the fruit fly. The resulting model not only explained how precise connectivity for path integration could develop through experience rather than genetic pre-specification, but also made novel predictions about the plasticity of the head direction system that align with experimental findings (Jayakumar et al., 2019). This predictive mechanism does not just apply to biological neurons; the same computational principle drives large language models where next-token prediction serves as a remarkably effective unsupervised learning objective (Brown et al., 2020). The parallel between these domains suggests that prediction may represent a universal principle for developing rich internal models, regardless of whether the substrate is biological or artificial.

Integration and evidence accumulation emerged as central computational themes across chapters, with continuous attractor dynamics serving as a common substrate for these functions. First proposed by Amari (Amari, 1977), continuous attractors were long theorized in computational neuroscience but received limited experimental attention until compelling demonstrations in the fly head direction system (Seelig

and Jayaraman, 2015; Green et al., 2017; Green et al., 2019). Chapter II showed how these dynamics emerge naturally from learning in the path integration context, while Chapter III demonstrated the spontaneous formation of higher-dimensional, disentangled continuous attractors in networks trained on noisy evidence accumulation tasks. The recurrent appearance of this computational motif across diverse contexts suggests that continuous attractors represent a general neural solution for maintaining and updating continuous variables. The recent proliferation of claimed continuous attractor discoveries in the experimental literature also prompted the development of Appendix D, which provides a principled approach to identifying and characterizing these dynamics, helping to prevent confusion and false positive detection of continuous attractors in experimental work.

The contributions of this thesis span multiple levels of analysis—from biophysically detailed models of neuronal learning to abstract computational principles governing representation formation. Throughout, a consistent theme has been the emergence of sophisticated computational capabilities from the interaction of relatively simple elements. This perspective bridges traditionally separate domains of neuroscience and artificial intelligence, demonstrating how insights from one can inform the other, and breaking a duality that has been ever-present, in particular in neuroscience research. The predictive learning mechanisms explored in cortical neurons may inspire new approaches to training artificial neural networks, while the theoretical guarantees for representation quality through multitask learning offer a principled framework for understanding both biological and artificial intelligence.

Overall, by focusing on the computational principles that enable learning and generalization across different neural systems, this thesis contributes to a unified science of Neural Computation. The findings presented here suggest that, irrespective of their specific implementations, biological and artificial neural networks converge on similar solutions when faced with similar computational challenges. This convergence points to fundamental principles of information processing that transcend the specific substrate in which they are implemented. As we continue to develop more sophisticated computational models and gather more detailed experimental data, the conceptual frameworks established in this thesis may help guide our understanding of how interconnected networks of simple processing elements—whether biological neurons or artificial units—give rise to the remarkable flexibility and generalization capabilities that characterize intelligent behavior; the mammalian cortex being our prime example and inspiration for such intelligence.

References

Amari, Shun-ichi (1977). "Dynamics of pattern formation in lateral-inhibition type neural fields." In: *Biological Cybernetics* 27.2, pp. 77–87. DOI: 10.1007/bf00337259. URL: https://doi.org/10.1007/bf00337259.

Brown, Tom B. et al. (2020). "Language models are few-shot learners." DOI: 10.48550/ARXIV.2005.14165. URL: https://arxiv.org/abs/2005.14165.

Green, Jonathan et al. (May 2017). "A neural circuit architecture for angular integration in Drosophila." In: *Nature* 546.7656, pp. 101–106. DOI: 10.1038/nature22343. URL: https://doi.org/10.1038/nature22343.

Green, Jonathan et al. (July 2019). "A neural heading estimate is compared with an internal goal to guide oriented navigation." In: *Nature Neuroscience* 22.9, pp. 1460–1468. DOI: 10.1038/s41593-019-0444-x. URL: https://doi.org/10.1038/s41593-019-0444-x.

Jayakumar, Ravikrishnan P. et al. (Feb. 2019). "Recalibration of path integration in hippocampal place cells." In: *Nature* 566.7745, pp. 533–537. DOI: 10.1038/s41586-019-0939-3. URL: https://doi.org/10.1038/s41586-019-0939-3.

Saez, Alex et al. (Aug. 2015). "Abstract context representations in primate amygdala and prefrontal cortex." In: *Neuron* 87.4, pp. 869–881. DOI: 10.1016/j.neuron.2015.07.024. URL: https://doi.org/10.1016/j.neuron.2015.07.024.

Seelig, Johannes D. and Vivek Jayaraman (May 2015). "Neural dynamics for landmark orientation and angular path integration." In: *Nature* 521.7551, pp. 186–191. DOI: 10.1038/nature14446. URL: https://doi.org/10.1038/nature14446.

Appendix A

SUPPLEMENTARY MATERIAL FOR CHAPTER I

A.1 How Does the RNN Learn?

In Chapter I, we have shown that the network is able to learn complex delay conditioning tasks using relatively few trials. In this section we explore in more detail the mechanisms through which the network solves the problem.

Figure A.1 shows how the activity of the associative neurons changes with training. It compares firing rates in response only to the *CS*s (top), in response only to the associated *US*s (middle), or in response to the full trial in which both are presented (bottom). Several things are worth noticing.

First, the right column depicts the activity of the network after it has learnt the delay conditioning task. At this stage, the activity patterns in response to only the *CS* or only the *US* are very similar. This makes it possible to decode the upcoming *US* using only the activity in the network in response to the associated *CS*.

Second, the network learns mixed stimulus representations. This is important since there is evidence that the associative areas of the prefrontal cortex use this type of mixed coding (Rigotti et al., 2013).

Third, the pattern of activity in response to only the *US* is unchanged by learning. This follows from the fact that in this case the response of the associative neurons is driven only by the input r_{us} to the somatic compartment and the synaptic weights W_{us} are not updated with training.

Fourth, the activity pattern in response to both the *CS* and the *US*, is very similar to the response to the *US* alone, irrespective of the stage of learning. This is because the firing rate in our model is mainly controlled by the *US*, while later in learning the *CS* would induce the same response anyway. Overall, the learning rule modifies the *CS* weights so that the *CS* inputs are able to generate the representation of the *US* both when the *CS* is presented by itself, and when presented together with the *US*.

Figure A.2 provides further insight into the inner workings of the model. Each panel depicts the dynamics of a model component within a training trial. Columns denote different stages of training. Recall that the learning rule between associative neuron i and input neuron j is the product of three terms: a surprise modulated learning rate $\eta(S)$, the presynaptic potential in the input neuron P_j, and the neuron-specific firing rate error term $\left[\, f(V_i^{\text{s}}) - f(p'\, V_i^{\text{d}}) \,\right]$.

Consider the last term first. $f(V_i^{\text{s}})$ is the firing rate of associative neuron i, which is determined by its somatic voltage V_i^{s}. $f(p'\, V_i^{\text{d}})$ is the (approximate) counterfactual

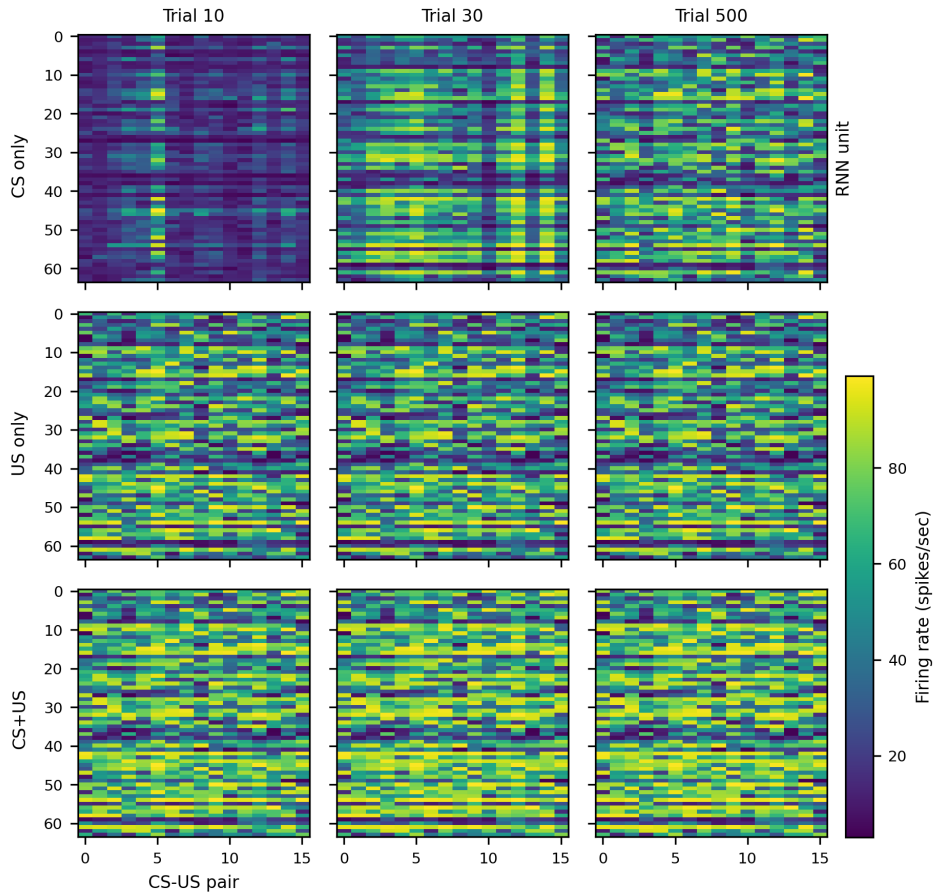

Figure A.1: ***RNN* activity during learning.** Firing rates in response to each stimulus pair at different stages of learning. Top row shows activity in response only to the associated *CS*. Middle row shows activity in response only to the associated US. Bottom row shows activity in response to the presentation of both. Activity is measured off-line (i.e., between learning trials).

firing rate that would occur if the *CS* were presented by itself. When the *US* is presented it dominates the activity of the associative neurons and thus the firing rate in the presence of both stimuli is similar to what would have been in the presence of only the *US*. As a result, for the *RNN* to be able to predict the *US* in response to only the *CS*, it has to be the case that $f(p' V_i^{\mathrm{d}}) \approx f(V_i^{\mathrm{s}})$. The learning rule implements a gradient like rule by increasing the *CS* input weights when $f(p' V_i^{\mathrm{d}}) < f(V_i^{\mathrm{s}})$, and decreasing them when the opposite is true. As shown in the third row of fig. A.2, these two variables are unrelated early in training, but converge to the same pattern as learning progresses.

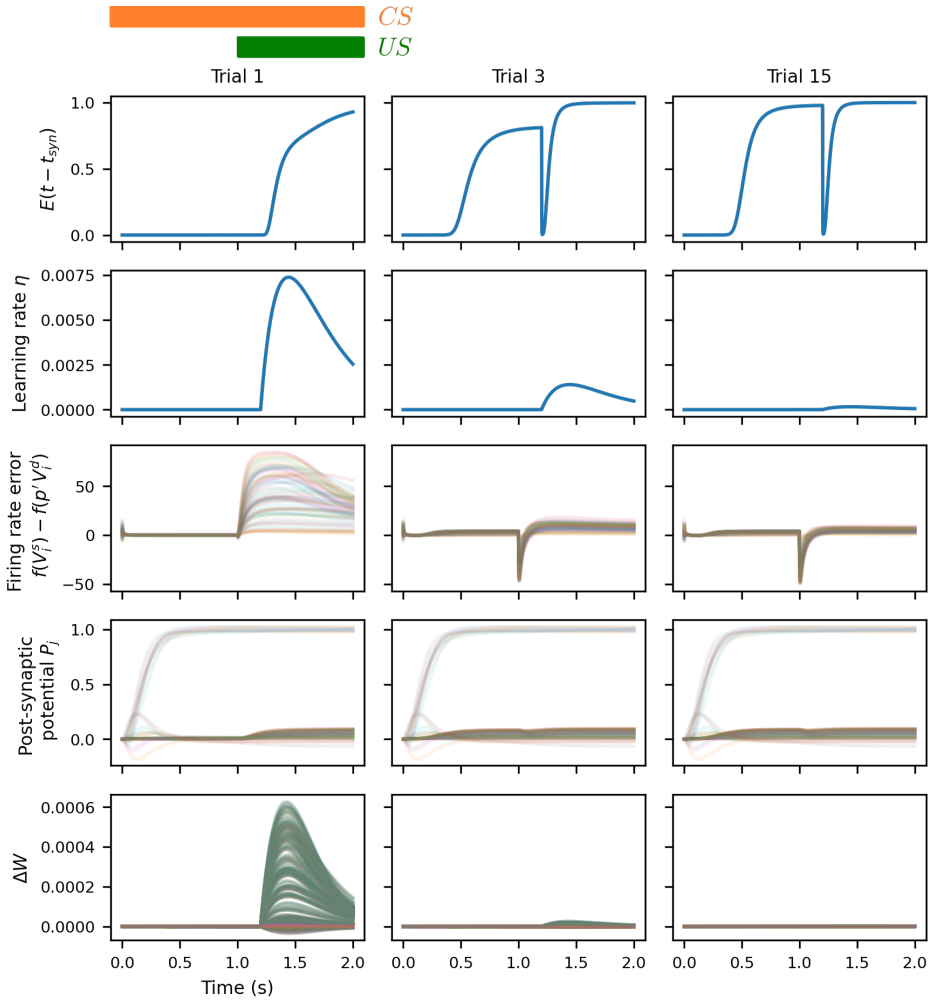

Figure A.2: **Within trial dynamics of model components.** Each panel depicts the dynamics of a model component within a training trial. Columns denote the level of training. Rows denote model variables. $f(V_i^s)$ is the firing rate of associative neuron i, which is determined by its somatic voltage V_i^s. $f(p'V_i^d)$ is the (approximate) counterfactual firing rate of the neuron when only the *CS* is presented. E is the expectation signal for the *US* shown in the trial. $\eta(S)$ is the surprise-modulated learning rate. P_j is the presynaptic potentials of input neuron j. ΔW is the incremental weight change for elements of each element in W_{rnn} and W_{cs}.

Next consider the surprise modulated learning rate $\eta(S)$. Gating the learning rate by surprise is critical, as it provides a global reference signal crucial when there are more than one predictive *CS*s available. Furthermore, biological neurons may not be able to compute $f(p' V_i^d)$ exactly at the dendritic compartment, resulting in potential mismatches between $f(V_i^s)$ and $f(p' V_i^d)$. If the learning rate η were constant across training, these mismatches would result in slow unlearning when

nothing behaviorally significant is happening. In contrast, when the learning rate is gated by surprise, the learning rate $\eta = 0$ most of the times, and any mismatch between when $f(V_i^{\mathrm{s}})$ and $f(p'V_i^{\mathrm{d}})$ does not result in unlearning.

Finally consider the presynaptic potential P_j. This term is present in most learning rules and reflects the old Hebbian dictum that "neurons that fire together wire together." In particular, other things being equal, the weights of more active synapse are updated more since they have a potentially stronger influence on the postsynaptic firing rate.

We emphasize again that the fact that associative neurons are two-compartment neurons is important for the biological plausibility of the model. The gradient like term $\left[f(V_i^{\mathrm{s}}) - f(p' V_i^{\mathrm{d}}) \right]$ depends only on information available at the synapse, since it is based only on variables associated with that neuron. By definition, the presynaptic potential P_j is also available at the synapse. Finally, the learning rate is implemented by neuromodulators that are diffused to the synapses of the associative network. As a result, all of the variables required to implement the learning rule are locally available at each synapse.

Having explored the mechanisms through which the network learns, we now provide a more rigorous mathematical analysis of why the learning rule converges to the desired solution.

A.2 Convergence of Synaptic Plasticity Rule

To understand how and why the learning rule works, it is useful to characterize the somatic voltages, and thus their associated firing rates, in different trial conditions.

Consider first the case in which only the *CS* is presented, so the associative neurons only receive dendritic input. In this case the somatic voltages converge to a steady-state given by

$$V^{\mathrm{ss}} = \frac{g_D}{g_D + g_L} V^{\mathrm{d}}. \tag{A.1}$$

In other words, the somatic voltages converge simply to an attenuated level of the dendritic voltages, with the level of attenuation given by $p = \frac{g_D}{g_D + g_L}$. In this case, the firing rates of the associative neurons converge to

$$r_{\mathrm{rnn}}^{\text{cs-only}} = f(V^{\mathrm{ss}}). \tag{A.2}$$

This follows from the fact that the dendritic voltage is determined only by equations 1.2 and 1.3, and thus is not affected by the state of the somatic compartment, and by

the fact that in the absence of *US* input $I^{\mathrm{s}} = 0$. The result then follows immediately from equation 1.4.

Next consider the case in which only the *US* is presented. In this case equations 1.2 and 1.3 imply that $V^{\mathrm{d}} = 0$, and it then follows from equations 1.4 and 1.5 that the steady-state somatic voltage, when $I^{\mathrm{s}} = 0$, is given by

$$V^{\mathrm{eq}}(t) = \frac{g_{\mathrm{e}}E_{\mathrm{e}} + g_{\mathrm{i}}E_{\mathrm{i}}}{g_{\mathrm{e}} + g_{\mathrm{i}}} \tag{A.3}$$

and that the firing rates of the associative neurons become

$$r_{\mathrm{rnn}}^{\text{us-only}} = f(V^{\mathrm{eq}}). \tag{A.4}$$

Finally consider the case in which the associative neurons receive input from both the *CS* and the *US*. We follow Brea et al., 2016 to derive the steady-state solution for the somatic voltage in this case. Provided inputs to the circuit, which are in behavioral timescales, change slower than the membrane time constant (C/g_L = 20 ms), equation 1.4 reaches a steady-state given by

$$V^{\mathrm{s}}(t) \approx \kappa V^{\mathrm{ss}} + (1 - \kappa)V^{\mathrm{eq}}, \tag{A.5}$$

where $\kappa(t) = \frac{g_D+g_L}{g_D+g_L+g_{\mathrm{e}}+g_{\mathrm{i}}} \in (0, 1]$ performs a linear interpolation between the steady-state levels reached where only the *CS* or the *US* are presented.

Practically, when there is no *US*-input, V^{ss} slightly precedes V^{s} due to the non-zero dendritic-to-somatic coupling delays, resulting in slight overestimation of the firing rate upon *CS* presentation. This can be accounted for by introducing an additional small attenuation, so that $p' = a\frac{g_D}{g_D+g_L} = ap$ in equation 1.9, with $a = 0.95$.

Learning is driven by a comparison of the firing rates of the associative neurons in the presence of both the *CS* and the *US*, and the firing rates if they only receive input from the *CS*. Importantly, this can happen online and without the need for separate learning phases, because an estimate of the latter can be formed in the dendritic compartment at all times. Learning is achieved by modifying W_{rnn} and W_{cs} to minimize this difference. We can use the expressions derived in the previous paragraphs to see why the synaptic learning rule converges to synaptic weights for which $r_{\mathrm{rnn}}^{\text{cs-only}} = r_{\mathrm{rnn}}^{\text{both}}$.

Take the case in which associative neurons underestimate the activity generated by the *US* inputs when exposed only to the *CS* (i.e., $V^{\mathrm{ss}} < V^{\mathrm{eq}}$). In this case,

$V^{ss} < V^{s} < V^{eq}$ and $I^{s} > 0$. Then from equation 1.9 we find that $\Delta w > 0$, leading to a futures increase in associative neuron activity in response to the *CS*.

The same logic applies in opposite case, where the associative neurons overestimate the activity generated by the *US* inputs when exposed only to the *CS*. In this case, $V^{ss} > V^{s} > V^{eq}$ and $I^{s} < 0$, which leads to a future decrease in associative neuron activity in response to the *CS*.

Given enough training, this leads to a state where $V^{ss} \approx V^{eq}$ and at which learning stops ($\Delta w \approx 0$). When this happens, we have that

$$r_{\text{rnn}}^{\text{cs-only}} = f(V^{ss}) \approx f(V^{eq}) = r_{\text{rnn}}^{\text{both}}, \tag{A.6}$$

so that the *RNN* responses to the *CS* become fully predictive of the activity generated by the *US*, when presented by themselves.

The previous section demonstrated how the learning rule converges to a state where the CS response matches the US response. We now show how this learning rule can be derived directly from the objective of stimulus substitution.

A.3 Predictive Coding and Normative Justification for the Learning Rule

In this section we provide further insight into the learning rule used in our model by showing that it follows directly from the objective of stimulus substitution.

Stimulus substitution states that synaptic connections change during learning so that the activity of the associative network induced by the *CS* ($r_{\text{rnn}}^{\text{cs-only}}$) becomes identical to the response induced by the *US* ($r_{\text{rnn}}^{\text{us-only}}$). It follows that the objective of stimulus substitution is to minimize the discrepancy or loss $\mathcal{L}$ between the two:

$$\mathcal{L} = \frac{1}{2}(r_{\text{rnn}}^{\text{cs-only}} - r_{\text{rnn}}^{\text{us-only}})^2. \tag{A.7}$$

We assume that the synaptic weights for *US* inputs are fixed, since these are primary reinforcers. The synaptic weights for the *CS* inputs are plastic, and they are shaped so that the *CS* elicits the same response as the *US*, essentially becoming predictive of the latter. Assuming a rectified linear (ReLU) activation function, $r_{\text{rnn}}^{\text{cs-only}}$ will obey

$$r_{\text{rnn}}^{\text{cs-only}} = [\, W^{\top} P \,]_{+} \tag{A.8}$$

where W are the plastic synaptic weights for the *CS* inputs, and P are the postsynaptic potentialsof the input *CS* neurons, low-pass filtered by synaptic delays.

To minimize the loss $\mathcal{L}$, we perform local gradient descent with respect to W, which leads to the following update rule:

$$\frac{\partial W}{\partial t} = -\eta \frac{\partial \mathcal{L}}{\partial W}. \tag{A.9}$$

This results in the following update rule between input neuron j and associative neuron i from presynaptic neuron j:

$$\Delta W_{ij} = \eta \left(r_{\text{rnn},i}^{\text{us-only}} - r_{\text{rnn},i}^{\text{cs-only}} \right) P_j. \tag{A.10}$$

Here, $r_{\text{rnn},i}^{\text{us-only}}$ acts as a "teacher" signal, in a setting that resembles self-supervised learning. Specifically, $r_{\text{rnn},i}^{\text{cs-only}}$ is compared to $r_{\text{rnn},i}^{\text{us-only}}$, and the discrepancy determines the sign and magnitude of weight change. However, only synapses from presynaptic neurons that have recently been active ($P_j > 0$) are modified. This learning rule is said to perform predictive coding, because *CS* inputs should predict (or anticipate) the response to the *US*.

An implicit requirement of the learning rule is that there has to be a way to tell apart $r_{\text{rnn},i}^{\text{cs-only}}$ and $r_{\text{rnn},i}^{\text{us-only}}$, in order to compare them. However, a neuron only has a single output at a given time. Therefore, in principle it is unclear how the two firing rates could be compared in an online fashion and within the same neuron. The 2-compartment associative neurons resolve this because the activity in the somatic compartment $f(V_i^{\text{s}})$ provides a measure of $r_{\text{rnn},i}^{\text{us-only}}$ [1], $f(p' V_i^{\text{d}})$ provides a measure of $r_{\text{rnn},i}^{\text{cs-only}}$, and the information available to compute the former term is available in the dendritic compartment due to backpropagating action potentials (Larkum, Zhu, and Sakmann, 1999). Thus, the associative neurons contain all of the information needed to implement the learning rule that yields stimulus substitution.

[1] In reality, as we show in eq. (A.5) V_i^{s} is affected by both somatic and dendritic inputs, however as we explain in the same section the influence of the dendritic inputs can never change the sign of $\left[f(V_i^{\text{s}}) - f(p' V_i^{\text{d}}) \right]$, and the resulting weight changes are always in the correct direction.

References

Brea, Johanni et al. (June 2016). "Prospective coding by spiking neurons." In: *PLoS Computational Biology* 12.6. Ed. by Peter E. Latham, e1005003. DOI: 10.1371/journal.pcbi.1005003. URL: https://doi.org/10.1371/journal.pcbi.1005003.

Larkum, Matthew E., J. Julius Zhu, and Bert Sakmann (Mar. 1999). "A new cellular mechanism for coupling inputs arriving at different cortical layers." In: *Nature* 398.6725, pp. 338–341. DOI: 10.1038/18686. URL: https://doi.org/10.1038/18686.

Rigotti, Mattia et al. (May 2013). "The importance of mixed selectivity in complex cognitive tasks." In: *Nature* 497.7451, pp. 585–590. DOI: 10.1038/nature12160. URL: https://doi.org/10.1038/nature12160.

Appendix B

SUPPLEMENTARY MATERIAL FOR CHAPTER II

B.1 Supplementary figures

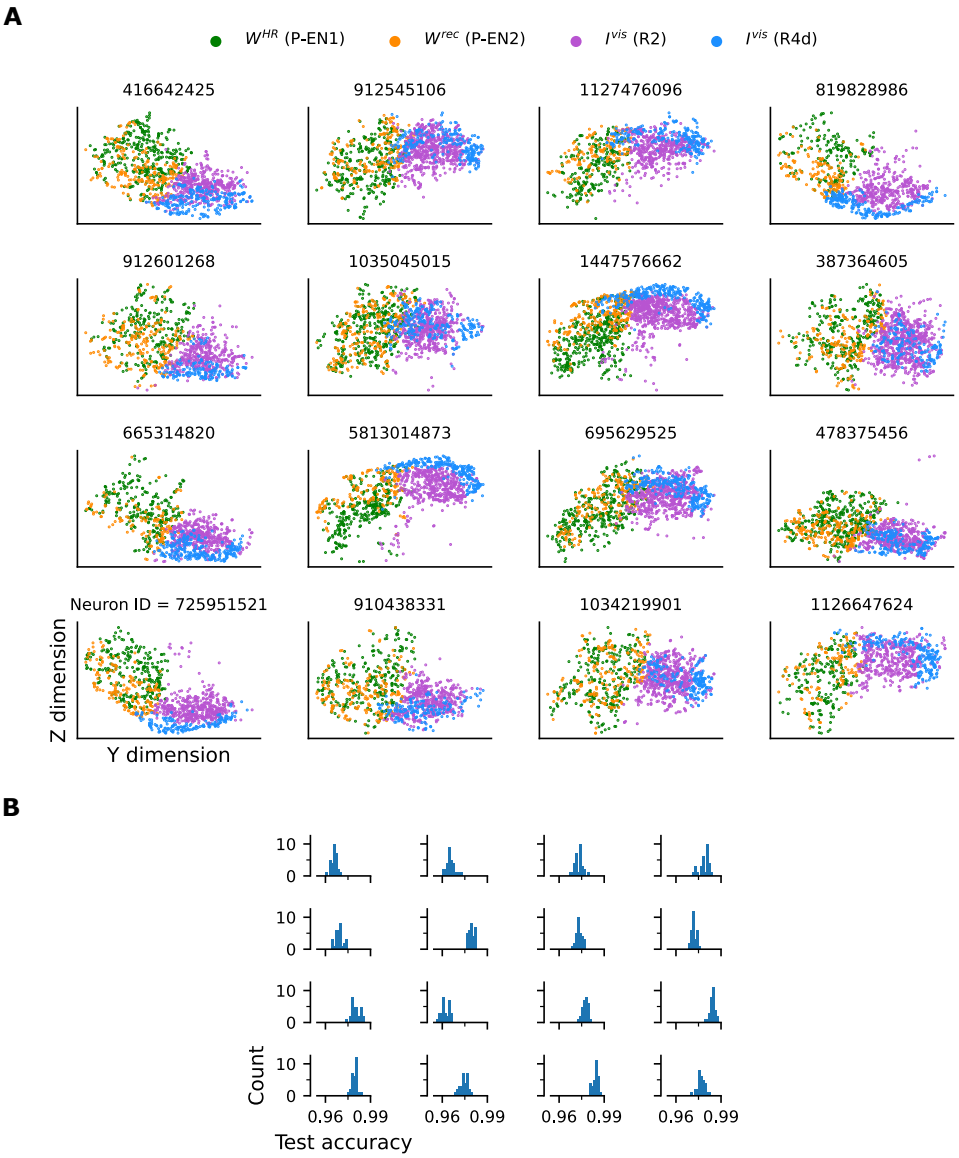

Figure B.1: **Separation of axon-proximal and axon-distal inputs to HD (E-PG) neurons in the *Drosophila* EB.** (A) Synaptic locations in the EB where visual (R2 and R4d) and recurrent and HR-to-HD (P-EN1 and P-EN2) arrive, for a total of 16 HD neurons tested (Neuron ID above each panel). Similarly to the example in Fig. 1E (repeated here in the top left panel, Neuron ID 416642425), these two sets of inputs appear to arrive in separate locations. (B) Binary classification between the two classes (R2 and R4d vs. P-EN1 and P-EN2) using SVMs with Gaussian kernel. Nested 5-fold cross validation was performed 30 times for every neuron tested, and the test accuracy histograms per neuron are plotted. The two classes can be separated with a test accuracy > 0.95 for every neuron.

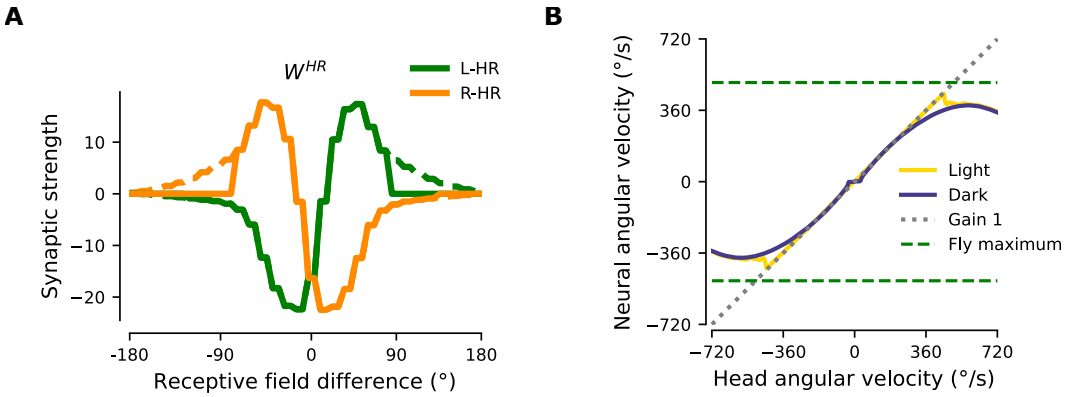

Figure B.2: **Removal of long-range excitatory projections impairs PI for high angular velocities.** (A) Profiles of the HR-to-HD weight matrix W^{HR} from fig. 2.3C (dashed lines), and the same profiles after the long-range excitatory projections have been removed (solid lines). (B) PI in the resulting network is impaired for high angular velocities, compared to fig. 2.2C.

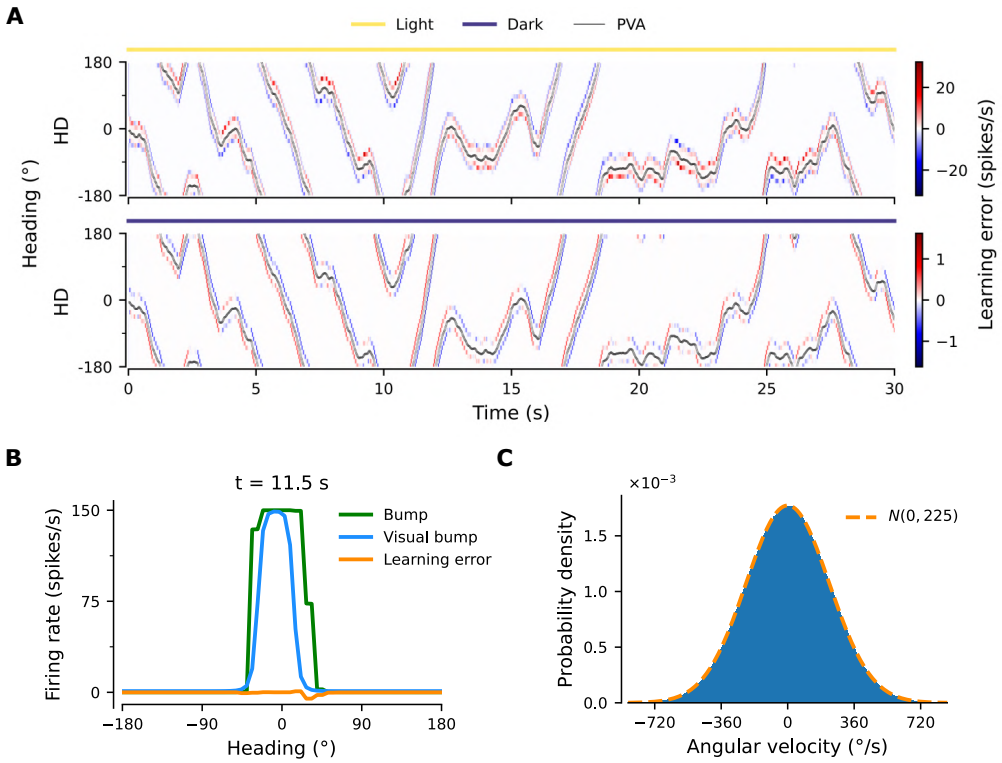

Figure B.3: **Details of learning.** (A) Learning errors (eq. (2.18)) in the converged network in light conditions (yellow overbar) or during PI in darkness (purple overbar). Note the difference in scale. In light conditions, the error is zero in all positions apart from the edges of the bump, where the error is substantial. Such errors occur because the velocity pathway, which implements PI, cannot move the bump for very small angular velocities, and tends to move it slightly faster for intermediate velocities, and slower for large ones (see fig. 2.2C). The velocity pathway is active and affects network activity even in the presence of visual input; hence, in light conditions, it creates errors at the edges of the bump, and the sign of the errors is consistent with the aforementioned PI velocity biases. Other than that, the angular velocity input predicts the visual input near-perfectly, as evidenced by the near-zero error everywhere else in the network. During PI in darkness, the network operates in a self-consistent manner, merely integrating the angular velocity input, and the learning error is much smaller. (B) Snapshot of the bump and the errors at $t = 11.5$ s in light conditions from (A). Also overlaid is the hypothetical form of the bump if only the visual input was present in the axon-proximal compartment of the HD neurons, termed "Visual bump." Notice that the errors are due to the fact that the visual bump is trailing in relation to the bump in the network. As a result, at the front of the bump the subthreshold visual input is actually inhibiting the bump. Also note that the bump in the network has a square form, in contrast to the smoother form that would be expected from visual input alone. This is because the learning rule in eq. (2.12) only converges when HD neurons reach saturation (see also fig. B.13 panel A2). (C) Histogram of entrained velocities.

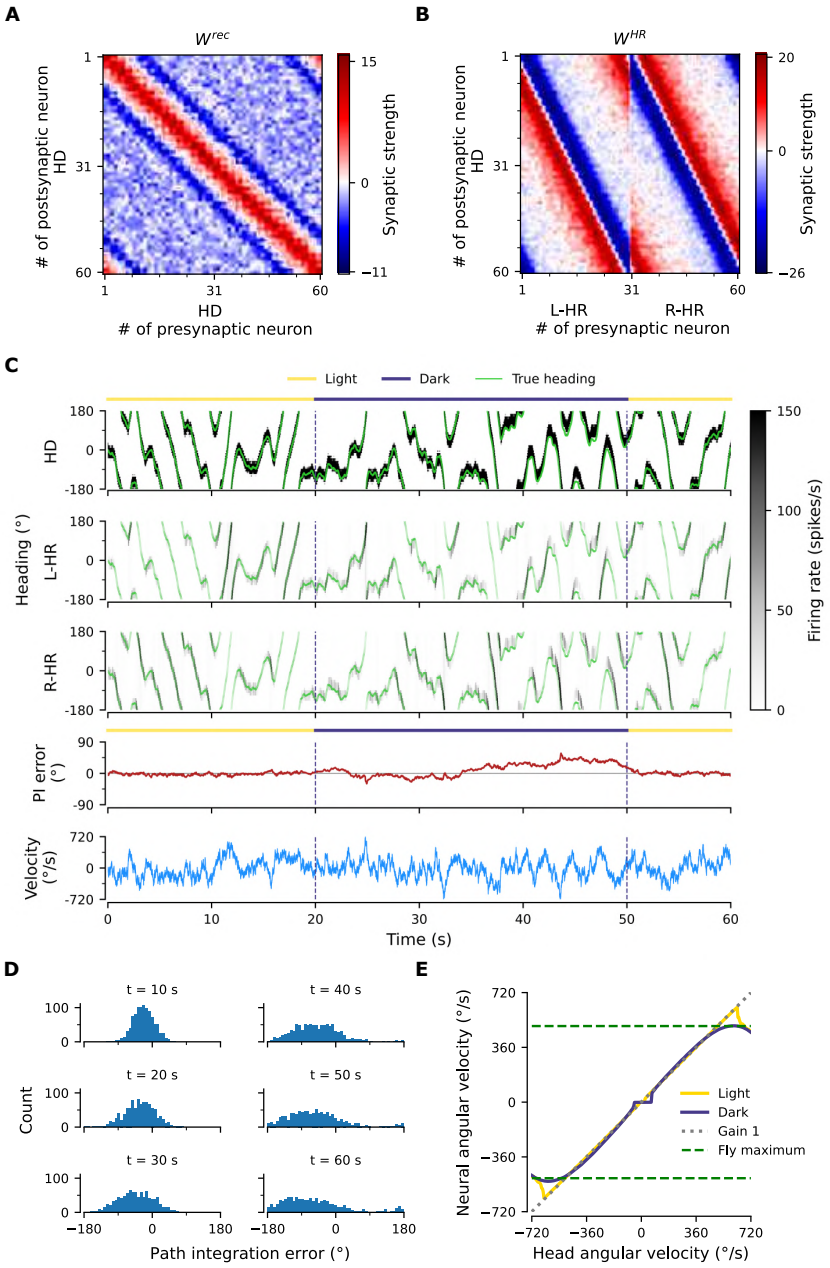

Figure B.4: **PI performance of a perturbed network.** After learning, the synaptic connections in fig. 2.3A,B have been perturbed with Gaussian noise with standard deviation ~1.5. (A), (B) Synaptic weight matrices after noise addition. (C) Example of PI. The activity of HD, L-HR, and R-HR neurons along with the PI error and instantaneous angular velocity are displayed, as in fig. 2.2A. (D) Temporal evolution of distribution of PI errors during PI in darkness. Compared to fig. 2.2B the distribution widens faster, and also exhibits side bias. (E) PI is impaired compared to fig. 2.2C, particularly for small angular velocities. Note that the exact form of the PI gain curve at very small angular velocities may vary slightly depending on the noise realization, but the findings mentioned in the Results (middle part of last paragraph of section "Learning results in synaptic connectivity that matches the one in the fly") remain consistent.

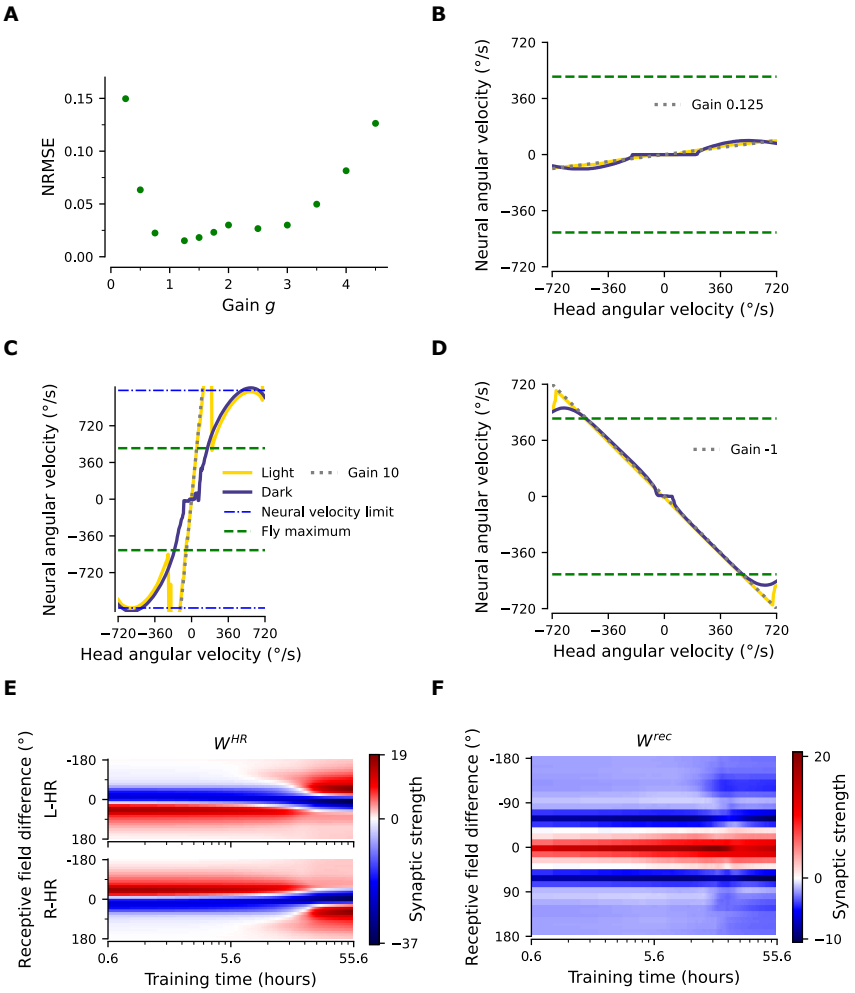

Figure B.5: **Limits of PI gain adaptation.** (A) Normalized root mean square error (NRMSE) between neural and head angular velocity, for gain-1 networks that subsequently have been rewired to learn different gains. We estimate RMSE in a range of head angular velocities set by the maximum neural angular velocity v_{max} (e.g., blue dot-dashed line in C; see also fig. B.7A); this range is given by this maximum neural angular velocity divided by the gain g. Then, to obtain the NRMSE we divide the RMSE by that range. For instance, in (C), $g = 10$ and $v_{max} = 1150$ deg/s, and we only test for the range of head angular velocities $[-115, 115]$ deg/s. We find that rewiring performance is excellent for gains g between 0.25 and 4.5, for which NRMSE < 0.15. Note that the more a new gain differs from original gain 1, the longer it takes for the network to rewire. (B), (C) PI performance plots for a small ($g = 0.125$) and a large ($g = 10$) gain. The NRMSE is 0.31 and 0.46, respectively. Performance is impaired because the flat area for small angular velocities gets enlarged in (B), whereas the network struggles to keep up with the desired gain in (C). (D) PI performance plot for a network that has been instructed to reverse its gain (from +1 to −1), i.e., when the visual and self-motion inputs are signaling movement in opposite directions. Performance is excellent, indicating that there is nothing special about negative gains; albeit learning takes considerably more time. (E), (F) Weight history for HR-to-HD and for recurrent connections, respectively, for the network trained to reverse its gain. The directionality of the asymmetric HR-to-HD connections in (E) reverses only after $\sim$ 20 hours, while the recurrent weights in (F) remain largely unaltered.

B.2 Robustness to noise

In Chapter II, the only source of stochasticity in the network came from the angular velocity noise in the Ornstein-Uhlenbeck process (Methods, eq. (2.17)). Biological HD systems, however, are subject to other forms of biological noise like randomness of ion channels. To address that, we include Gaussian IID synaptic current noise to every location in the network where inputs arrive: the axon-proximal and axon-distal compartments of HD cells and the HR cells (see Methods, parameterized by σ_n in eq. (2.2), eq. (2.4), and eq. (2.8)). We then ask how robustly can the network learn in the presence of such additional stochasticity.

To quantify the network's robustness to noise, we need to define a comparative measure of useful signals vs. noise in the network. By "signals" we refer to the velocity/visual inputs and any network activity resulting from them, whereas "noise" is the aforementioned Gaussian IID variables. We thus define the signal-to-noise ratio (*SNR*) as the squared ratio of the active range A_{active} of the activation function f (defined in eq. (2.7)) over two times the std of the Gaussian noise, σ_n, i.e.,

$$SNR = \left(\frac{A_{active}}{2\sigma_n} \right)^2 . \tag{B.1}$$

This definition is motivated by the fact that A_{active} determines the useful range that signals in the network can have. If any of the signals exceed this range, they cannot impact the network in any meaningful way because the neuronal firing rate has saturated, unless they are counterbalanced by other signals reliably present. The factor 2 in the denominator is due to the fact that the noise can extend to both positive and negative values, whereas A_{active} denotes the entire range of useful inputs.

Here we vary the *SNR* and observe its impact on learning and network performance. fig. B.6A–D shows the performance of a network that has been trained with $SNR \approx 2$. The resulting network connectivity remains circularly symmetric and maintains the required asymmetry in the HR-to-HD connections for L- and R-HR cells (data not shown). Therefore we plot only the profiles in fig. B.6B, which look very similar to the ones in fig. 2.3C trained with $SNR = \infty$. The peak of the local excitatory connectivity in W^{rec} is not as pronounced. This happens because the noise corrupts auto-correlations of firing during learning.

The network activity still displays a clear bump that smoothly follows the ground truth in the absence of visual input (fig. B.6A). There are only minor differences compared to the network without noise (fig. 2.2A). The presence of the noise is

most obvious in the HR cells, since HD cells that do not participate in the bump are deep into inhibition, and therefore synaptic input noise does not affect as much their activity. We note that the network can no longer sustain a bump in darkness when $SNR = 1$, i.e., when the standard deviation of the noise covers the full active range of inputs (data not shown).

Finally, the neural velocity slightly overestimates the head angular velocity (fig. B.6C compared to fig. 2.2C), and the PI errors diffuse faster in the network with noise (88.1 deg^2/s in fig. B.6D compared to 24.5 deg^2/s in fig. 2.2B); these values are also indicated in fig. B.6E (triangles). Importantly, we find that the diffusion assumption holds, because the estimation of the diffusion coefficient when varying simulation time (between 10 and 60 s) is consistent.

So far we have addressed the diffusivity of PI in networks that receive velocity input, and we have compared the performance of networks that were trained with and without synaptic input noise. However, in mature networks, i.e., during testing, it is unclear how large the impact of synaptic input noise is, compared to noise that originates from imperfect PI. To disentangle these two noise contributions during testing, we study diffusivity in networks that do not receive velocity inputs at all (i.e., without PI). In the absence of velocity inputs, we vary the level of synaptic input noise during testing (called "test noise"), and estimate diffusion coefficients. Specifically, for each test noise magnitude σ_n we run 1000 simulations of $t_{sim} = 10$ s, each time randomly initializing the network at one of the angular locations θ_i for which HD neurons are tuned for. For $\sigma_n = 0$, we run one simulation per θ_i, since the simulation is deterministic.

Our results in fig. B.6E show that the diffusion coefficients obtained in networks without velocity inputs (dots) are always much smaller that those with velocity inputs (triangles). Thus, imperfect integration of velocity inputs is by far the dominating source of noise in trained, mature networks. Omitting the velocity inputs, we detected small differences between networks that were trained without synaptic input noise (blue dots) and with such noise (orange dots, train noise $\sigma_n = 0.7$). The network trained with noise is sligthly more diffusive up to the level of test noise at which is was trained ($\sigma_n = 0 - 0.7$), and it is slightly less diffusive beyond that level ($\sigma_n > 0.7$).

We conclude that learning PI is robust to synaptic input noise during learning, and that synaptic input noise during testing degrades performance much less than errors due to deviations from perfect gain-1 PI, which are already quite small.

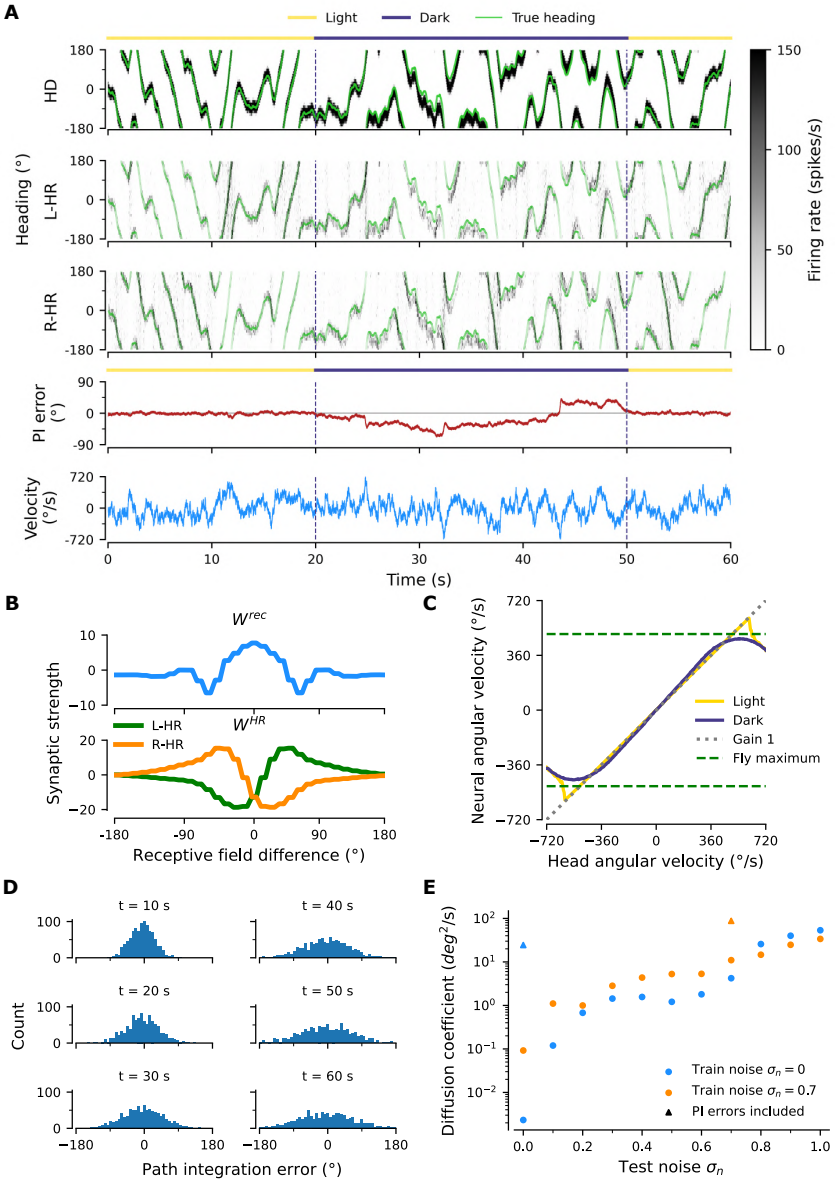

Figure B.6: **Robustness to injected noise.** (A) PI example in a network trained with noise ($SNR \approx 2$, train noise $\sigma_n = 0.7$). Panels are organized as in fig. 2.2A, which shows the activity in a network trained without noise ($SNR = \infty$, $\sigma_n = 0$). (B) Profiles of learned weights. Both W^{rec} and W^{HR} are circularly symmetric. Panel is organized as in fig. 2.3C, which shows weight profiles in a network trained without noise ($SNR = \infty$, $\sigma_n = 0$). (C) The network achieves almost perfect gain-1 PI, despite noisy inputs. Compared to fig. 2.2C the performance is only slightly impaired. (D) Temporal evolution of distribution of PI errors during PI in darkness. Compared to fig. 2.2B the distribution widens faster, however it also does not exhibit side bias. (E) Diffusion coefficient for networks as a function of the level of test noise (for details, see section "Diffusion Coefficient" in Methods). We distinguish between networks that experienced noise during training ($\sigma_n = 0.7$, orange) and networks that were trained without injected noise ($\sigma_n = 0$, blue), which were studied in the Results of Chapter II. Diffusion coefficients that include contributions from PI errors, estimated from (D) and fig. 2.2B, are also plotted (triangles).

B.3 Synaptic delays set a neural velocity limit during path integration

In Chapter II we trained networks for a set of angular velocities that cover the full range exhibited by the fly ($|v| < 500$ deg/s), and we showed that the mature network can account for several key experimental findings. However, the ability of any continuous attractor network to path-integrate is naturally limited for high angular velocities, due to the synaptic delays inherent in any such network (Zhong et al., 2020). To evaluate the ability of our network to integrate angular velocities, we sought to identify a limit of what velocities could be learned.

The width of the HD bump in our network is here termed *BW*, and it is largely determined by the width σ of the visual receptive field. This is because during training we force the network to produce a bump with a width matching that of the visual input, and this width is then maintained when the latter is not present. The reason for this behavior is that the width of the learned local excitatory connectivity profile in W^{rec} that guarantees such stable bumps of activity will be similar to the width of the bump, because recurrent connections during learning are only drawn from active neurons (non-zero P_j in eq. (2.12)). As mentioned in Chapter II, this emphasizes the Hebbian component of our learning rule (fire together — wire together). As a result, the width of local excitatory recurrent connections should be approximately *BW*.

In fig. B.2 we show that the higher angular velocities are served by the long-range excitatory connections in W^{HR}. However, these connections might not be strong enough to move the bump by themselves; a contribution from HD cells might still be needed to move the bump at such high angular velocities. In that case, the width of the connectivity bump in W^{rec} might limit how *far away* from the current location the bump can be moved. In addition, there is a limitation in how *quickly* the bump can be moved: the learning rule in eq. (2.1) tries to predict the next state of the network from the current state; but to activate the next HD neurons in line, current HD and HR cell activity must go through the synaptic delay τ_s. Therefore, the maximum velocity that the network can achieve without external guidance (i.e., without visual input) should be inversely proportional to τ_s, i.e.,

$$v_{max} = \frac{b(\sigma, \Delta\phi)}{\tau_s} \tag{B.2}$$

where we assume that b might reflect an effective HD connectivity bump width, which depends on σ but also on the angular resolution of the HD network $\Delta\phi$, due to

discretization effects. In reality, the HR-to-HD connectivity profiles in W^{HR} likely also have a bearing on b.

We then systematically vary τ_s and test what velocities the network can learn. We indeed find that networks can path-integrate all angular velocities up to a limit, but not higher than that. As predicted, this limit is inversely proportional to τ_s, for a wide range of delays (fig. B.7A). Furthermore, b matches BW reasonably well. Fitting eq. (B.2) to the data we obtain $b(0.25, 6\,\text{deg}) \approx BW = 96\,\text{deg}$ for $N^{HD} = N^{HR} = 120$ and $b(0.15, 12\,\text{deg}) = 75\,\text{deg}$, $BW = 60\,\text{deg}$ for $N^{HD} = N^{HR} = 60$.

As mentioned in Chapter II, there are two limitations other than synaptic delays why the network could not learn high angular velocities: limited training of these velocities, and saturation of HR cell activity. These limitations kick in for $\tau_s < 150$ ms, for which v_{max} matches the maximum velocity the fly displays (500 deg/s). Therefore to create fig. B.7A for these delays, we increased the standard deviation of the velocity noise in the Ornstein-Uhlenbeck process to $\sigma_v = 800\,\text{deg/s}$ to address the first limitation, and we increased the dynamic range of angular velocity inputs by decreasing the proportionality constant in eq. (2.10) to $k = 1/540\,\text{s/deg}$ to address the second.

The velocity gain plot for an example network with high synaptic delays ($\tau_s = 190$ ms) is shown in fig. B.7B. Interestingly, we notice that the performance drop at the velocity limit is not gradual; instead, the neural velocity abruptly drops to a near-zero value once past the velocity limit. Further investigation reveals that for velocities higher than this limit, the network can no longer sustain a bump (fig. B.7C). This happens because the HD network cannot activate neurons downstream fast enough to keep the bump propagating, and therefore the bump disappears and the velocity gain plot becomes flat.

Equation (B.2) is similar to a relationship reported in Turner-Evans et al. (2017) (their page 35, 1st paragraph), where it was demonstrated that the phase shift of the HR population bump compared to the HD bump limits angular velocity. Our result hence generalizes this finding in the case where recurrent connections between HD neurons are also allowed.

Finally, we note that so far we only tested the limits of network performance when increasing τ_s. To demonstrate that smaller delays also work, as an extreme example we show PI performance in a network where $\tau_s = 1$ ms in fig. B.7D,E. A potential issue with such small synaptic delays is that the network would not be able to

distinguish rightward from leftward rotation for small angular velocities, because the motion direction offset of the HR bumps would be small, and the activity in the two HR populations comparable. In such a setting it is harder to learn the asymmetries in the HR-to-HD connections required to differentiate leftward from rightward movement. Indeed, this effect is visible in fig. B.7E where the amplitude of HR-to-HD connections has been suppressed, and in fig. B.7D where the flat region for small angular velocities has been extended compared to fig. 2.2C.

Overall, these results indicate that the network learns to path-integrate angular velocities up to a fundamental limit imposed by the architecture of the HD system in the fly. Furthermore, we conclude that the phenomenological delays observed in the fly HD system in Turner-Evans et al. (2017) are not fundamentally limiting the system's performance, since they can support PI for angular velocities much higher than the ones normally displayed by the fly.

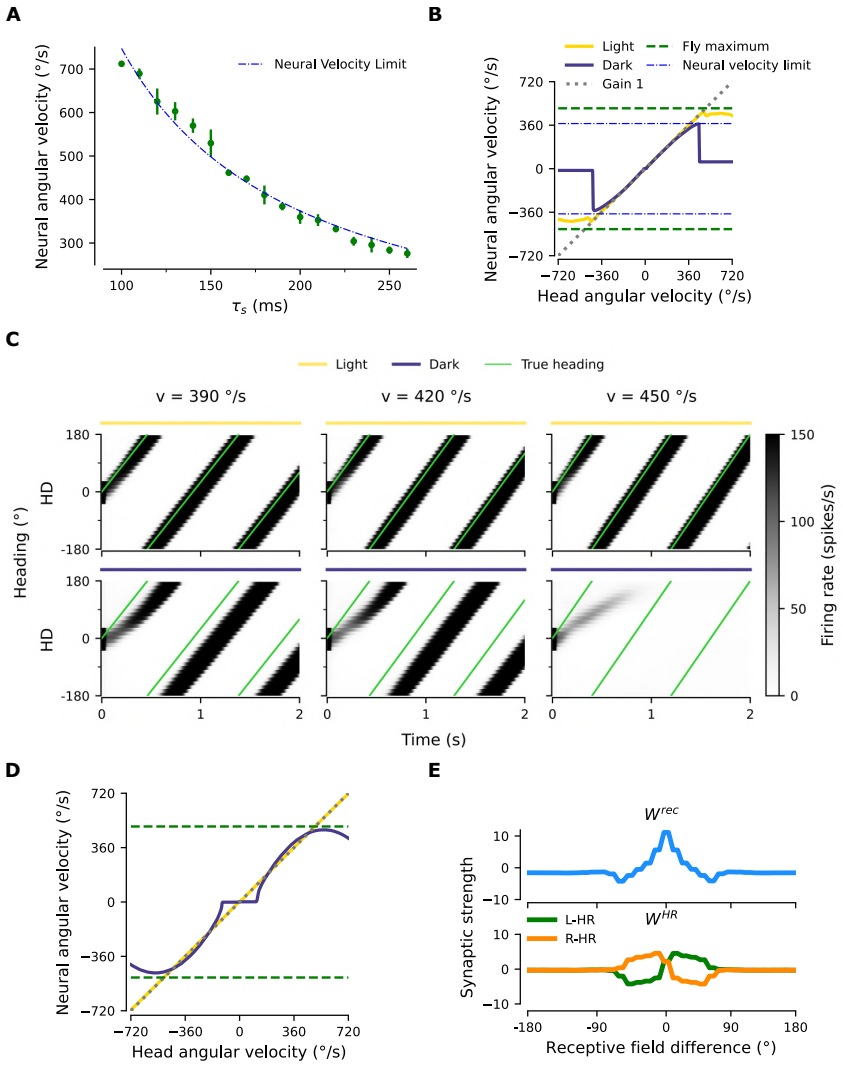

Figure B.7: **Limits of network performance when varying synaptic delays.** (A) Maximum neural angular velocity learned is inversely proportional to the synaptic delay τ_s in the network, with constant $b = 75$ deg in eq. (B.2) (blue dot-dashed line). Green dots: point estimate of maximum neural velocity learned, green bars: 95 % confidence intervals (Student's t-test, $N = 5$). (B) Example neural velocity gain plot (as in fig. 2.2C) in a network with increased synaptic delays (τ_s increased from the "standard" value 65 ms to the new value 190 ms). (C) Behavior of the activity of HD cells in the network with parameters as in (B) near the velocity limit. The example network is driven by a single velocity in every column, in light (top row) and darkness (bottom row) conditions. In darkness, near and below the limit (left and middle column), there is a delay in the appearance of the bump, which then path-integrates with gain 1; above the limit observed in (B), however, the bump cannot stabilize, resulting in the dip in neural velocity (right column). (D) PI performance for a network with drastically reduced synaptic delays ($\tau_s = 1$ms). Compared to fig. 2.2C, PI performance is worse for small angular velocities. This occurs because for small angular velocities, the offset of the HR bump in leftward vs. rightward movement is not as pronounced. As a result, it is harder to differentiate leftward from rightward movement. (E) For the same reason, the asymmetries in the learned HR-to-HD connectivity are not as prevalent as in fig. 2.3C.

B.4 Robustness to architectural asymmetries

The networks we have trained in the Results had a circular symmetric initial architecture, including the hardwired HD-to-HR connections W^{HD}. However, such symmetry is unrealistic for any biological system that is assembled by imperfect processes; deviations from symmetry should be expected. Therefore, in this Appendix we let W^{HD} vary randomly, and observe how PI performance is affected.

First, we remind the reader that the magnitude w^{HD} of the HD-to-HR connections is chosen so that we take advantage of the full dynamic range of HR neurons; however, the exact magnitude should not be critical for our model. Homeostatic plasticity could adjust the magnitude, but for simplicity we have not incorporated such plasticity rules in our model. Instead, to see whether the exact values of synaptic weights, their circular symmetry, and the 1-to-1 nature of the HD-to-HR connections is crucial for our model, we draw connection strengths randomly.

In a first approach, we let HD neurons project also to adjacent HR neurons. Specifically, if $U(a, b)$ denotes the uniform distribution in the interval (a, b), we sample the magnitude of weights from $U\left(\frac{w^{HD}}{2}, w^{HD}\right)$ for the main diagonal and $U\left(0, \frac{3\,w^{HD}}{4}\right)$ for the side diagonals of W^{HD} (fig. B.8A). We then adjust the network connectivity (W^{rec} and W^{HR}) in a learning phase, similar to the one illustrated in fig. 2.3E,F. After learning, as shown in fig. B.8C–E, PI is still excellent because the learning rule can balance out any deviations from circular symmetry in W^{HD}. It does so by introducing deviations from circular symmetry in the learned weights, mainly in W^{HR} (fig. B.8B). Thus, small deviations from circular symmetry in the learned weights are essential for PI.

We illustrate the necessity to counterbalance small deviations of circular symmetry again in a second example that is based on the symmetric network studied in the Results. Here we use the connectivity of the network illustrated in fig. 2.3A–C and also preserve the 1-to-1 nature of HD-to-HR connections, but now we randomly vary their magnitude, while maintaining the same average connection strength; specifically, we sample the magnitude of weights from $U\left(\frac{w^{HD}}{2}, \frac{3w^{HD}}{2}\right)$. PI performance in this network is considerably impaired compared to the original network (compare fig. B.9 to fig. 2.2A,C). This is a further argument in favor of synaptic plasticity operating to fine-tune connectivity, because as mentioned we expect that such anatomical asymmetries are indeed present in the biological circuit. Therefore, even if the circular symmetric synaptic weights were passed down genetically with great

accuracy, PI performance in flies should be considerably degraded for a biological circuit with anatomical asymmetries when no learning is involved.

To better quantify the effect of anatomical asymmetries, we incorporate both noise in the learned connectivity as in fig. B.4A,B and noise in the HD-to-HR connections as in fig. B.8A. We tune the noise independently for each weight matrix: for W^{rec} and W^{HR} we set the variance of the Gaussian noise to p times the variance of the individual weight matrices, while for W^{HD} we draw the connections connections from $U\left((1-p)\,w^{HD}, (1+p)\,w^{HD}\right)$ for the main diagonal and $U\left(0, p\,w^{HD}\right)$ for the side diagonals. We find that for $p = 0.3$ the correlation between the PVA and true heading in darkness drops 0.27 ± 0.09 which is below reported fly PI performance (mean correlation across animals ~ 0.5 in Seelig and Jayaraman (2015)), while the structure of the weights is preserved. We observed a steep decline in the correlation coefficient between $p = 0.25$ (for which the correlation is 0.92 ± 0.04) and $p = 0.3$. Furthermore, we study the effect of perturbing individual weight matrices, and find that perturbing only W^{HD} with $p = 0.3$ considerably affects performance (correlation 0.39 ± 0.09) while perturbation of W^{rec} and W^{HR} together has a much smaller effect on performance. This again argues in favor of learning W^{rec} and W^{HR} to counterbalance asymmetries in W^{HD} (fig. B.8). Furthermore, note that confounders other than imperfect weights might be responsible for the degradation of PI performance in the fly, which further argues in favor of learning.

As a final test for the capability of the learning rule to balance anatomical asymmetries, in fig. B.10 we use a completely random connectivity for HD-to-HR connections, drawing weights from a folded Gaussian distribution. We find that even then, PI performance of the converged network is great, albeit for a smaller range of velocities. In addition, bumps are not clearly visible in the HR populations anymore; in the main network, HR bumps were inherited from the HD bump due to the sparseness of the HD-to-HR connections. However, when HD-to-HR connections are random, HR cells are no longer mapped to a topographic state space.

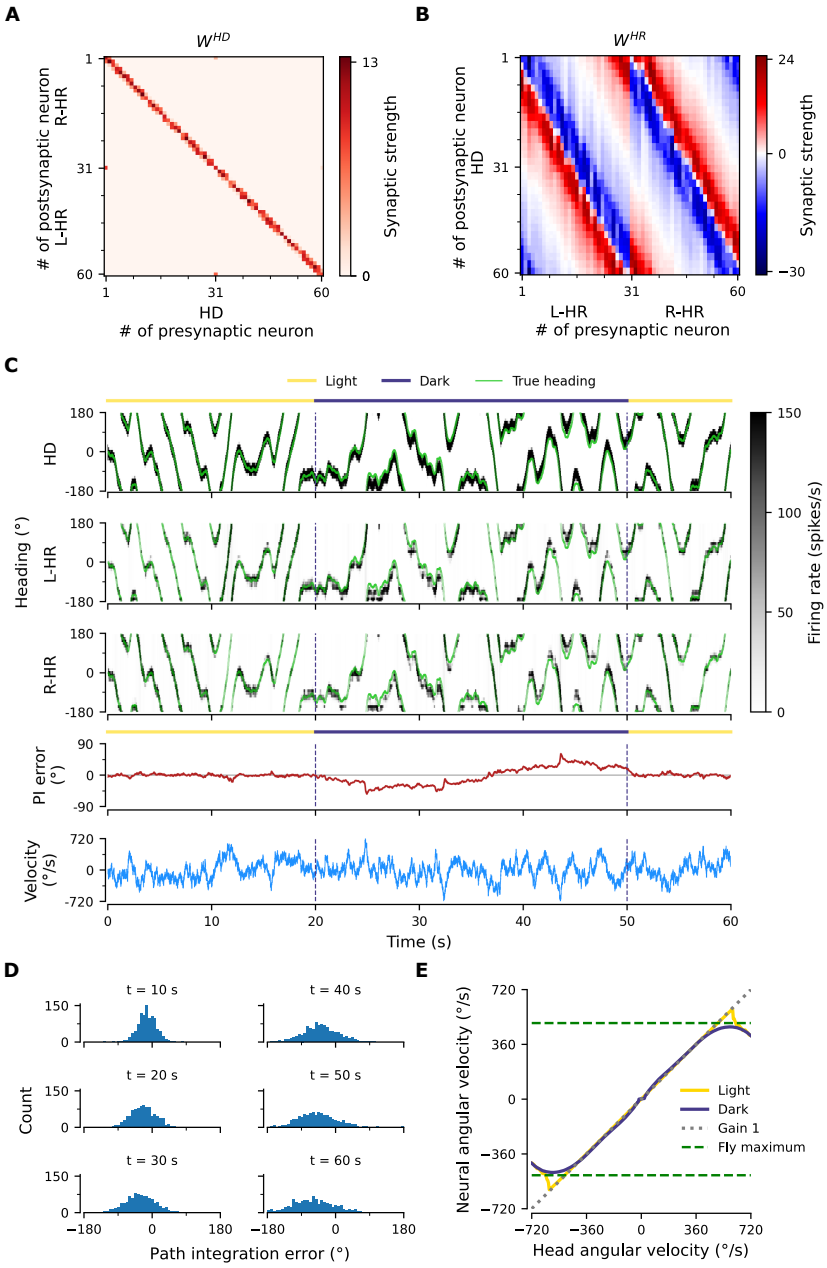

Figure B.8: **Performance of a network where HD-to-HR connection weights are allowed to vary randomly**, and HD neurons are projecting to HR neurons also adjacent to the ones they correspond to, respecting the topography of the protocerebral bridge (PB). (A) The HD-to-HR connectivity matrix, W^{HD}. Note that, compared to what is described in the Methods (final paragraph of "Neuronal Model"), the order of HD neurons is rearranged: we have grouped HD neurons that project to the same wing of the PB together, so that the diagonal structure of the connections is clearly visible. (B) The learned HR-to-HD connections, W^{HR}, depart from circular symmetry (as, e.g., in fig. 2.3B), so that asymmetries in W^{HD} could be counteracted. The recurrent connections W^{rec} (not shown) remain largely unaltered compared to the ones shown in fig. 2.3A. (C)–(E) Despite the randomization and lack of 1-to-1 nature of HD-to-HR connections, PI in the converged network remains excellent (cf. fig. 2.2A–C).

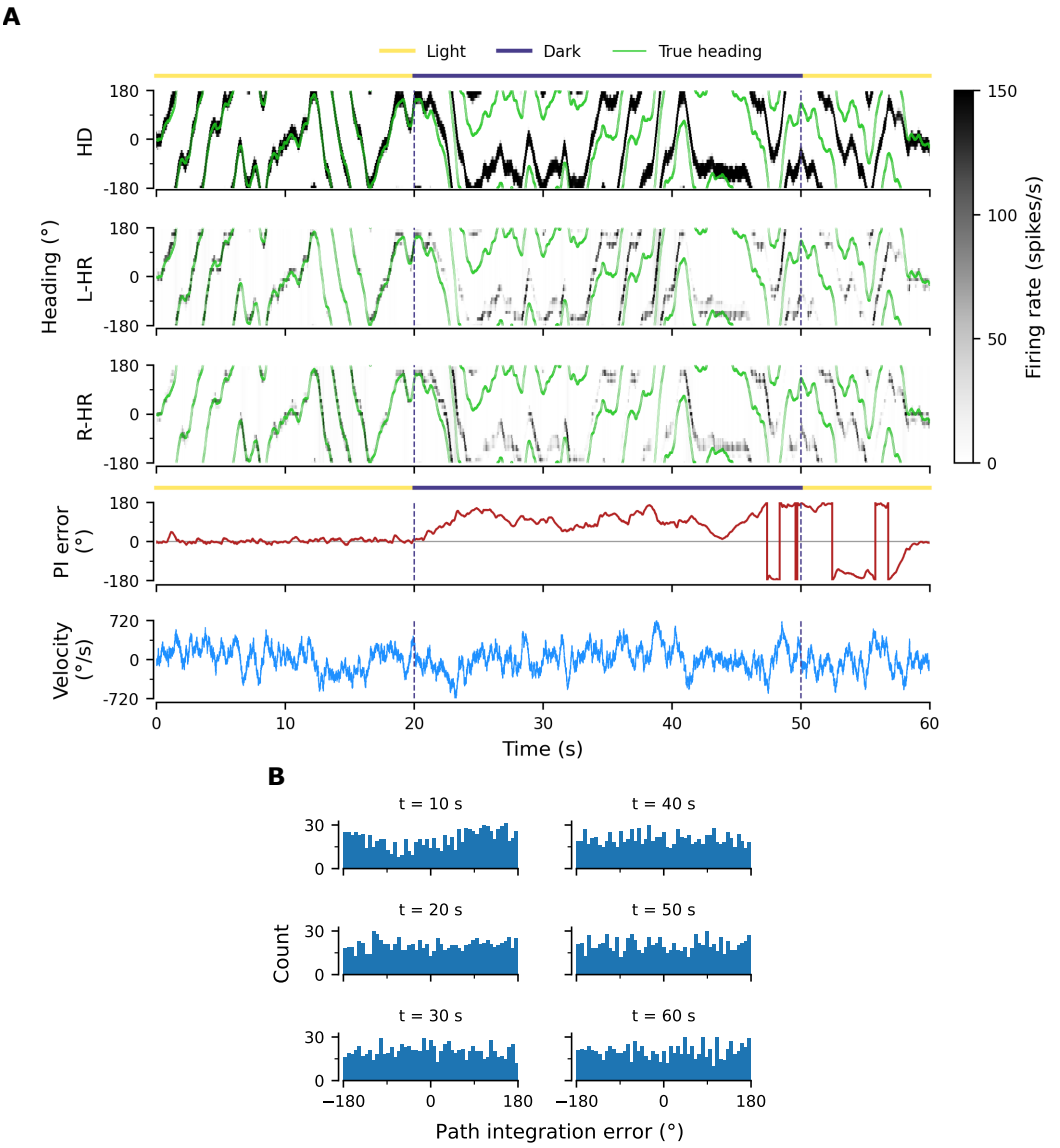

Figure B.9: **PI performance in a network with random HD-to-HR connection strengths** and learned weights from network in fig. 2.3. Here we vary the magnitude of the main diagonal HD-to-HR connections but preserve the 1-to-1 nature of the connections. We assume that W^{rec} and W^{HR} are passed down genetically (i.e., there is no further learning of these connections), and therefore the same, circular symmetric profiles apply to every location in the circuit. We choose these (assumed here to be genetically stored) profiles to be the ones we learned in the network outlined in fig. 2.3A,B. (A) Example that shows that PI is impaired, because the circular symmetric profiles passed down genetically cannot counteract small asymmetries in the architecture that are likely to be present in any biological system. Notice that it can even take several seconds for the large PI error to be corrected by the visual input. (B) PI errors grow fast (compare to, e.g., fig. 2.2B). Already by 20 sec of PI the heading estimate is random.

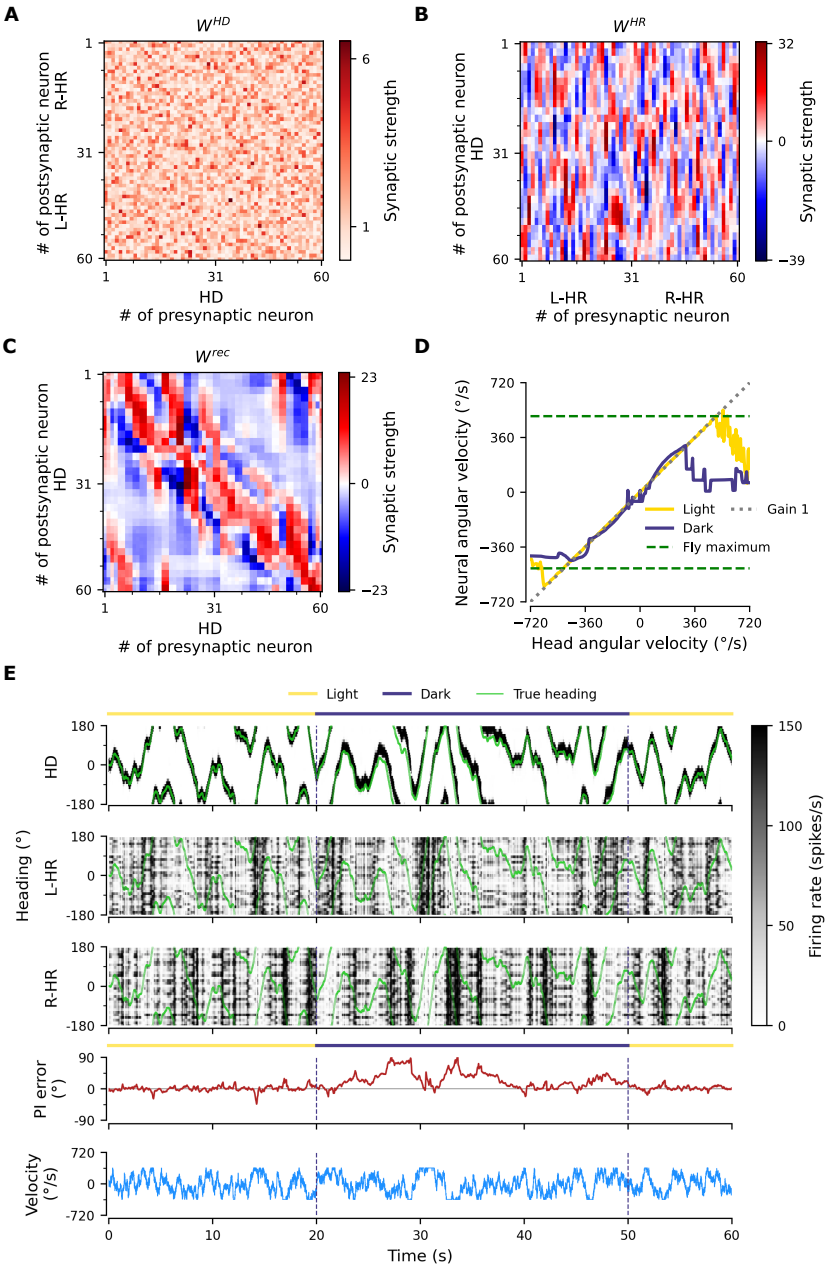

Figure B.10: **PI performance of a network where HD-to-HR connection weights are completely random.** (A) The HD-to-HR weights are drawn from a folded normal distribution, originating from a normal distribution with 0 mean and $\pi(w^{HD})^2/200$ variance. (B) As a result, the learned HR-to-HD connections have also lost their structure. (C) The recurrent connections preserve some structure, since adjacency in the HD network is still important. (D) Impressively, the converged network can still PI with a gain close to 1, but for a reduced range of angular velocities compared to, e.g., the network in fig. B.8. (E) A bump still appears in the HD network and gets integrated in darkness, albeit with larger errors. Note that bumps no longer appear in the HR populations; HR bumps are inherited from the HD bump only when adjacencies in the HD population are carried over to the HR populations by the HD-to-HR connections. Note that we have restricted angular velocities to the interval $[-360, 360]$ deg/s for this example, to showcase that PI is still accurate within this interval.

B.5 Requirements on time scales

We devote this Appendix to discuss requirements for the time scales involved in our model (see table B.1). Several of these time scales are well constrained by biology, and thus we chose to keep them constant. These include the membrane time constants of the axon-proximal and axon-distal compartments, C/g_L and τ_l, respectively, which should be in the order of milliseconds; and the velocity decay time constant τ_v, for which we choose a value in the same order of magnitude (0.5 s) as experimentally reported (Turner-Evans et al., 2017).

In general, the learning time scale given by $1/\eta$ should be the slowest one in our network model. The time scale should be large enough so that the network samples the input statistics for a long enough time. Varying $1/\eta$ from 2 s to 20 s to 200 s, we find that the final learned weights are virtually identical (not shown). However, $1/\eta$ should not be too large to enable fast enough learning.

The synaptic time constant τ_s is determined from phenomenological delays in the network (Turner-Evans et al., 2017). Nevertheless, we also addressed the impact of varying τ_s in Appendix B.3 and found that learning PI was robust in a wide range of values.

In additional simulations, we varied the weight update filtering time constant τ_δ from 0 (effectively removing filtering) to 1000 s (which is four orders of magnitude larger than the default value). We observed almost no effect on learning dynamics, and the performance of the final networks was almost identical (not shown). Since the specific value of τ_δ is of little consequence in our network (if $1/\eta$ is large enough), there are hardly any limitations on its value compared to other time scales. Therefore, this justifies ignoring τ_δ in the derivation of a reduced model without noise in Appendix B.6.

Table B.1: Default values for time scales in the model

Time scale	Expression	Value
Membrane time constant of axon-proximal compartment	C/g_L	1 ms
Membrane time constant of axon-distal compartment	τ_l	10 ms
Synaptic time constant	τ_s	65 ms
Weight update filtering time constant	τ_δ	100 ms
Velocity decay time constant	τ_v	0.5 s
Learning time scale	$1/\eta$	20 s

Default values for time scales in the model, ordered by their magnitude.

B.6 Reduced theoretical model for a circular symmetric learned network

In this section we derive a reduced model for the dynamics of the synaptic weights during learning. The goal is to gain an intuitive understanding of the structure obtained in the full model (fig. 2.3 of Chapter II). Such a model reduction is obtained by 1) exploiting the circular symmetry in the system; 2) averaging weight changes across different speeds and moving directions; 3) writing dynamical equations in terms of convolutions and cross-correlations. With these methods, we derive a non-linear dynamical system for the weight changes as a function of head direction. Finally, we simulate this dynamical system and inspect how the different variables interact to obtain the final weights. We find that the reduced model results in nearly identical connectivity and learning dynamics to the full network in Chapter II, and explains how the latter assigns learning errors to the correct weights. Furthermore, it drastically reduces simulation times by two orders of magnitude.

Note that in this section we use slightly different notation compared to Chapter II. Notably, we refer to the recurrent head direction weight matrix simply as W (omitting the superscript *rec*), and use capital letters for functions of time and small letters for functions of heading direction.

We study the learning equation (see eqs. (2.12)–(2.16) where the low-pass filtering with time constant τ_δ has been ignored, since we find that the value of τ_δ is not important for learning (see Appendix B.5))

$$\frac{\mathrm{d}}{\mathrm{d}t} W_{ij}(t) = \eta\, E_i(t)\, P_j(t) \tag{B.3}$$

where

$$E_i(t) = f[V_i^a(t)] - f[V_i^{ss}(t)] \tag{B.4}$$

is the pre-synaptic error at cell i and

$$P_j(t) = \int_0^\infty \mathrm{d}s\, H(s) f[V_j^a(t-s)] \tag{B.5}$$

is the post-synaptic potential at HD cell j, and H is a temporal filter (with time constants τ_s and τ_l, see eq. (2.14)).

Clockwise movement

Assuming that the head turns clockwise (which equals to rightward rotation, i.e., rotation towards decreasing angles) and anti-clockwise (leftward, i.e., towards increasing angles) with equal probability, we can approximate the weight dynamics by

summing the average weight change W_{ij}^{+} for clockwise movement and the average weight change W_{ij}^{-} for anti-clockwise movement:

$$\frac{\mathrm{d}}{\mathrm{d}t}W_{ij}(t) = \frac{\mathrm{d}}{\mathrm{d}t}W_{ij}^{+}(t) + \frac{\mathrm{d}}{\mathrm{d}t}W_{ij}^{-}(t). \tag{B.6}$$

We start by assuming head movement at constant speed and we later generalize the results for multiple speeds. We compute the expected weight change $\frac{\mathrm{d}}{\mathrm{d}t}W_{ij}^{+}$ for one lap in the clockwise direction at speed $v^{+} > 0$:

$$\frac{\mathrm{d}}{\mathrm{d}t}W_{ij}^{+}(t) = \frac{\eta v^{+}}{2\pi}\int_{0}^{2\pi/v^{+}} \mathrm{d}\tau\, E_i^{+}(\tau)\, P_j^{+}(\tau) \tag{B.7}$$

where

$$P_j^{+}(t) = \int_{0}^{\infty} \mathrm{d}s\, H(s) f[V_j^{a+}(t-s)] \tag{B.8}$$

is the post-synaptic potential for clockwise movement, and

$$E_i^{+}(t) = f[V_i^{a+}(t)] - f[V_i^{ss+}(t)] \tag{B.9}$$

is the error for a clockwise movement. Assuming that the axon-proximal voltage is at steady state (eq. (2.4) with the l.h.s. set to zero and I_{exc}^{HD} absorbed into I_{vis}), the clockwise axon-proximal voltage reads

$$V_i^{a+}(t) = V_i^{ss+}(t) + \frac{I_i^{vis}(t)}{g_D + g_L} \tag{B.10}$$

where (see eq. (2.11))

$$V_i^{ss+}(t) = \frac{g_D}{g_D + g_L} V_i^{d+}\,. \tag{B.11}$$

From eqs. (2.2) and (2.3), we can write the axon-distal voltage V_i^{d+} as a low-pass filtered version of the total axon-distal current D_i^{+} for clockwise movement (see also eq. (2.14)):

$$V_i^{d+} = \int_{0}^{\infty} \mathrm{d}s\, H(s) D_i^{+}(t-s)\,, \tag{B.12}$$

which yields

$$V_i^{ss+}(t) = \frac{g_D}{g_D + g_L}\int_{0}^{\infty} \mathrm{d}s\, H(s) D_i^{+}(t-s)\,. \tag{B.13}$$

Importantly, the visual input I^{vis} is translation invariant:

$$I_j^{vis}(t) = I_i^{vis}\left(t + \frac{\theta_j - \theta_i}{v^{+}}\right) \tag{B.14}$$

where θ_j and θ_i are the preferred head directions of cells j and i, respectively. As a result of this translation invariance, the recurrent weight matrix W develops circular symmetry:

$$W_{ij} = W_{0,(j-i) \bmod N_{\mathrm{HD}}} \tag{B.15}$$

where N_{HD} is the number of HD cells in the system. Consequently, the post-synaptic potential P_j^+ is also translation invariant:

$$P_j^+(\tau) = P_i\left(\tau + \frac{\theta_j - \theta_i}{v^+}\right) = P_0\Big(\tau + \frac{\overbrace{\theta_j - \theta_0}^{\theta :=}}{v^+}\Big)\,. \tag{B.16}$$

In this case, without loss of generality, we can rewrite eq. (B.7) for a single row of the matrix $\frac{\mathrm{d}}{\mathrm{d}t}W_{ij}^+$ as a function of the angle difference $\theta := \theta_j - \theta_0$:

$$\frac{\mathrm{d}}{\mathrm{d}t}W_{ij}^+(t) = \frac{\mathrm{d}}{\mathrm{d}t}W_{0,(j-i) \bmod N_{\mathrm{HD}}}^+(t) = \quad \frac{\eta v^+}{2\pi}\int_0^{2\pi/v^+} \mathrm{d}\tau\, E_0^+(\tau)\, P_{(j-i) \bmod N_{\mathrm{HD}}}^+(\tau) \tag{B.17}$$

$$= \quad \frac{\eta v^+}{2\pi}\int_0^{2\pi/v^+} \mathrm{d}\tau\, E_0^+(\tau)\, P_0^+(\tau + \theta/v^+) \tag{B.18}$$

$$= \quad \frac{\eta}{2\pi}\int_0^{2\pi} \mathrm{d}\varphi\, E_0^+(\varphi/v^+)\, P_0^+[(\varphi + \theta)/v^+] \tag{B.19}$$

$$= \quad \frac{\eta}{2\pi}\int_0^{2\pi} \mathrm{d}\varphi\, \epsilon^+(\varphi)\, p^+(\varphi + \theta) \tag{B.20}$$

$$= \quad \frac{\eta}{2\pi}(\epsilon^+ \star p^+)(\theta) \tag{B.21}$$

$$=: \quad \frac{\mathrm{d}}{\mathrm{d}t}w^+(\theta)\,. \tag{B.22}$$

where we defined $\epsilon^+(\varphi) := E_0^+(\varphi/v^+)$ and $p^+(\varphi) := P_0^+(\varphi/v^+)$, and $\star$ denotes circular cross-correlation.

From eq. (B.8), we derive

$$p^+(\varphi) := P_0^+(\varphi/v^+) = \int_0^\infty \mathrm{d}s\, H(s) f[V_0^{a+}(\varphi/v^+ - s)] \tag{B.23}$$

$$\approx \int_0^{2\pi} \mathrm{d}\beta \underbrace{\frac{1}{|v^+|} H(\beta/v^+)}_{=:\, h^+(\beta)} f[\underbrace{V_0^{a+}((\varphi-\beta)/v^+)}_{=:\, v^{a+}(\varphi-\beta)}] \tag{B.24}$$

$$= [h^+ * f(v^{a+})](\varphi)\,. \tag{B.25}$$

The approximation in eq. (B.24) holds when a bump exists in the network and moves with a velocity below the velocity limit, and it is valid if the temporal filter H is shorter than $2\pi/v^+$, that is for $H(t) \ll 1$ for $t > 2\pi/v^+$, which holds for the filtering time constants and velocity distribution we assumed (fig. B.11). Therefore, plugging eq. (B.25) into eq. (B.22), we obtain:

$$\frac{\mathrm{d}}{\mathrm{d}t} w^+(\theta) \approx \frac{\eta}{2\pi} \{\epsilon^+ \star [h^+ * f(v^{a+})]\}(\theta)\,. \tag{B.26}$$

By using the definition of $\epsilon(\varphi)^+$ we derive

$$\epsilon^+(\varphi) := E_0^+(\varphi/v^+) = f[\underbrace{V_0^{a+}(\varphi/v^+)}_{=\, v^{a+}(\varphi)}] - f[\underbrace{V_0^{ss+}(\varphi/v^+)}_{=:\, v^{ss+}(\varphi)}] \tag{B.27}$$

with (eq. (B.10))

$$v^{a+}(\varphi) = v^{ss+}(\varphi) + \underbrace{\frac{I_0^{vis}(\varphi/v^+)}{g_D + g_L}}_{=:\, \bar{I}_{vis}(\varphi)} \tag{B.28}$$

and (eq. (B.13))

$$v^{ss+}(\varphi) = \frac{g_D}{g_D + g_L} \int_0^\infty \mathrm{d}s\, H(s) D_0^+(\varphi/v^+ - s) \tag{B.29}$$

$$\approx \frac{g_D}{g_D + g_L} \int_0^{2\pi} \mathrm{d}\beta \underbrace{\frac{1}{|v^+|} H(\beta/v^+)}_{=\, h^+(\beta)} \underbrace{D_0^+\left(\frac{\varphi-\beta}{v^+}\right)}_{=:\, d^+(\varphi-\beta)} \tag{B.30}$$

$$\approx \frac{g_D}{g_D + g_L} (h^+ * d^+)(\varphi). \tag{B.31}$$

The approximation in eq. (B.31) is valid if the temporal filter H is shorter than $2\pi/v^+$, which again holds true for our parameter choices (fig. B.11).

Calculation of the axon-distal input

Let us compute the axon-distal current D_i^+ to neuron i for clockwise movement. From eq. (2.2), setting the l.h.s. to zero, and splitting the rotation-cell activities in the two populations (L-HR and R-HR), we derive

$$D_i^+(t) = \underbrace{\sum_j W_{ij}(t) f[V_j^{a+}(t)]}_{:= D_i^{rec+}(t)} + \underbrace{\sum_j W_{ij}^R(t) f[V_j^{R+}(t)]}_{:= D_i^{R+}(t)} + \underbrace{\sum_j W_{ij}^L(t) f[V_j^{L+}(t)]}_{:= D_i^{L+}(t)} + I_{inhib}^{\mathrm{HD}} . \tag{B.32}$$

where W_{ij}^{R} (W_{ij}^{L}) are the weights from the right (left) rotation cells, and V_j^{R+} (V_j^{L+}) are the voltages of the right (left) rotation cells (see eqs. (2.8)–(2.10)):

$$V_j^{R+}(t) = \frac{A_{\mathrm{active}}}{f_{\max}} \int_0^\infty \mathrm{d}s\, H_{\mathrm{s}}(s) f[V_j^{a+}(t-s)] + \bar{I}_{vel} + I_{inhib}^{\mathrm{HR}} \tag{B.33}$$

$$V_j^{L+}(t) = \frac{A_{\mathrm{active}}}{f_{\max}} \int_0^\infty \mathrm{d}s\, H_{\mathrm{s}}(s) f[V_j^{a+}(t-s)] - \bar{I}_{vel} + I_{inhib}^{\mathrm{HR}} . \tag{B.34}$$

The function $H_S(t) := \exp(-t/\tau_s)/\tau_s$ is a temporal low pass filter with time constant τ_s and the velocity input reads (eq. (2.10))

$$\bar{I}_{vel} := v^+/(2\pi) . \tag{B.35}$$

Equation (B.33) and eq. (B.34) show that the rotation-cell voltages are re-scaled and filtered versions of the corresponding HD-cell firing rates with a baseline shift $\bar{I}_{vel}$ that is differentially applied to right and left rotation cells.

From eq. (B.32), we derive

$$d^+(\varphi) := D_0^+\left(\frac{\varphi}{v^+}\right) = D_0^{rec+}\left(\frac{\varphi}{v^+}\right) + D_0^{R+}\left(\frac{\varphi}{v^+}\right) + D_0^{L+}\left(\frac{\varphi}{v^+}\right) + I_{inhib}^{\mathrm{HD}} . \tag{B.36}$$

Assuming a large number N_{HD} of HD cells evenly spaced around the circle, the recurrent axon-distal input reads

$$D_0^{rec+}\left(\frac{\varphi}{v^+}\right) = \sum_j W_{0j} f[V_j^{a+}(\varphi/v^+)] + I_{inhib}^{\mathrm{HD}} \tag{B.37}$$

$$= \rho_{\mathrm{HD}} \int_0^{2\pi} \mathrm{d}\theta\, w(\theta)\, f\Big[\underbrace{V_0^{a+}\left(\frac{\varphi+\theta}{v^+}\right)}_{=:\, v^{a+}(\varphi+\theta)} \Big] + I_{inhib}^{\mathrm{HD}} \tag{B.38}$$

$$= \rho_{\mathrm{HD}} \left[w \star f(v^{a+})\right](\varphi) + I_{inhib}^{\mathrm{HD}} \tag{B.39}$$

where $\rho_{\mathrm{HD}} = N_{\mathrm{HD}}/2\pi$ is the density of the HD neurons around the circle and we used the fact that the axon-proximal voltage is translation invariant (see also eq. (B.16)):

$$V_j^{a+}(\tau) = V_0^{a+}(\tau + \theta/v^+)\,. \tag{B.40}$$

Following a similar procedure for D_0^{R+} and D_0^{L+}, we obtain:

$$d^+(\theta) = \left[\rho_{\mathrm{HD}} w \star f(v^{a+}) + \rho_{\mathrm{HR}} w^R \star f(v^{R+}) + \rho_{\mathrm{HR}} w^L \star f(v^{L+})\right](\theta) + I_{inhib}^{\mathrm{HD}} \tag{B.41}$$

where $\rho_{\mathrm{HR}} = N_{\mathrm{HR}}/2\pi$ is the density of the HR neurons for one particular turning direction (note that we assumed $\rho_{\mathrm{HR}} = 2\rho_{\mathrm{HD}}$ in Chapter II). In deriving eq. (B.41) we defined

$$v^{R+}(\theta) := \quad V_0^{R+}(t/v^+) \approx \frac{A_{\mathrm{active}}}{f_{\max}}[h_s^+ * f(v^{a+})](\theta) + \bar{I}_{vel} + I_{inhib}^{\mathrm{HR}} \tag{B.42}$$

$$v^{L+}(\theta) := \quad V_0^{L+}(t/v^+) \approx \frac{A_{\mathrm{active}}}{f_{\max}}[h_s^+ * f(v^{a+})](\theta) - \bar{I}_{vel} + I_{inhib}^{\mathrm{HR}} \tag{B.43}$$

where we defined the filter $h_s^+(\varphi) := \frac{1}{|v^+|} H_s(t/v^+)$, and the approximations are valid if $H_s(t/v^+) \ll 1$ for $t > 2\pi/v^+$, which holds true for the time contant and velocity distribution assumed in Chapter II.

Finally, we compute the rotation-cells' weights change. For these weights, the learning rule is the same as the one for the recurrent connections, except that the post-synaptic HD input is replaced by the post-synaptic HR input. Therefore, following the same procedure as in eq. (B.17)–eq. (B.25), the rotation weight changes are given by:

$$\frac{\mathrm{d}}{\mathrm{d}t} w^{R+}(\theta) = \quad \frac{\eta}{2\pi}\{\epsilon^+ \star [h^+ * f(v^{R+})]\}(\theta) \tag{B.44}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} w^{L+}(\theta) = \quad \frac{\eta}{2\pi}\{\epsilon^+ \star [h^+ * f(v^{L+})]\}(\theta)\,. \tag{B.45}$$

In summary, for clockwise movement, we obtain the following system of equations:

$$\begin{cases} d^+(\theta) & = [\rho_{\mathrm{HD}} w \star f(v^{a+}) + \rho_{\mathrm{HR}} w^R \star f(v^{R+}) + \rho_{\mathrm{HR}} w^L \star f(v^{L+})](\theta) + I_{inhib}^{\mathrm{HD}} \\ v^{ss+}(\theta) & = \frac{g_D}{g_D+g_L}(h^+ * d^+)(\theta) \\ v^{a+}(\theta) & = v^{ss+}(\theta) + \bar{I}_{vis}(\theta) \\ v^{R+}(\theta) & = \frac{A_{\mathrm{active}}}{f_{\max}}[h_s^+ * f(v^{a+})](\theta) + \bar{I}_{vel} + I_{inhib}^{\mathrm{HR}} \\ v^{L+}(\theta) & = \frac{A_{\mathrm{active}}}{f_{\max}}[h_s^+ * f(v^{a+})](\theta) - \bar{I}_{vel} + I_{inhib}^{\mathrm{HR}} \\ \epsilon^+(\theta) & = f[v^{a+}(\theta)] - f[v^{ss+}(\theta)] \\ \frac{\mathrm{d}}{\mathrm{d}t} w^+(\theta) & = \frac{\eta}{2\pi}\{\epsilon^+ \star [\underbrace{h^+ * f(v^{a+})}_{=:\, p^+}]\}(\theta) \\ \frac{\mathrm{d}}{\mathrm{d}t} w^{R+}(\theta) & = \frac{\eta}{2\pi}\{\epsilon^+ \star [\underbrace{h^+ * f(v^{R+})}_{=:\, p^{R^+}}]\}(\theta) \\ \frac{\mathrm{d}}{\mathrm{d}t} w^{L+}(\theta) & = \frac{\eta}{2\pi}\{\epsilon^+ \star [\underbrace{h^+ * f(v^{L+})}_{=:\, p^{L+}}]\}(\theta)\,. \end{cases} \tag{B.46}$$

Anti-clockwise movement

We now consider anticlockwise movements with speed $v^- = -v^+$. First we note that the temporal filter

$$h^-(\theta) := \frac{1}{|v^-|} H(\theta/v^-) = \frac{1}{|v^+|} H(-\theta/v^+) = h^+(-\theta) \tag{B.47}$$

is a mirrored version about the origin of its clockwise counterpart h^+, whereas the visual input is unchanged because it is symmetric around the origin (see eq. (2.5))

$$I_0^{vis}(\theta/v^-) = I_0^{vis}(\theta/v^+)\,. \tag{B.48}$$

Let us first assume that

$$d^-(\theta) = d^+(-\theta)\,, \tag{B.49}$$

we shall verify the validity of this assumption self-consistently at the end of this section. From eq. (B.47)–eq. (B.49) it follows that $f(v^{a-}) = f\left[\frac{g_D}{g_D+g_L}(h^- * d^-) + \bar{I}^{vis}\right]$ is a mirrored version of $f(v^{a+})$, that is,

$$f[v^{a-}(\theta)] = f[v^{a+}(-\theta)]\,, \tag{B.50}$$

and, as a result,

$$\epsilon^{-}(\theta) = \epsilon^{+}(-\theta)\,. \tag{B.51}$$

We now compute the anticlockwise weight change for the recurrent weights

$$\frac{\mathrm{d}}{\mathrm{d}t}w^{-}(\theta) = \frac{\eta}{2\pi}\{\epsilon^{-} \star [h^{-} * f(v^{a-})]\}(\theta)\,. \tag{B.52}$$

The r.h.s. of eq. (B.52), without the $\eta/(2\pi)$ pre-factor reads:

$$\{\epsilon^{-} \star [h^{-} * f(v^{a-})]\}(\theta) = \int_0^{2\pi} \mathrm{d}\tau\, \epsilon^{-}(\tau) \int_0^{2\pi} \mathrm{d}s\, h^{-}(s) f[v^{a-}(\tau+\theta-s)] \tag{B.53}$$

$$= \int_0^{2\pi} \mathrm{d}\tau\, \epsilon^{+}(-\tau) \int_0^{2\pi} \mathrm{d}s\, h^{+}(-s) f[v^{a+}(-\tau-\theta+s)] \tag{B.54}$$

$$= \int_0^{2\pi} \mathrm{d}\tau\, \epsilon^{+}(\tau) \int_0^{2\pi} \mathrm{d}s\, h^{+}(s) f[v^{a+}(\tau-\theta-s)] \tag{B.55}$$

$$= \{\epsilon^{+} \star [h^{+} * f(v^{a+})]\}(-\theta) \tag{B.56}$$

where from eq. (B.54) to eq. (B.55) we used variable substitution. Therefore, the weight change for clockwise movement is the mirrored version around the origin of the weight change for anticlockwise movement:

$$\frac{\mathrm{d}}{\mathrm{d}t}w^{-}(\theta) = \frac{\mathrm{d}}{\mathrm{d}t}w^{+}(-\theta)\,, \tag{B.57}$$

meaning that, with learning, the recurrent weights develop into an even function:

$$w(\theta) = w(-\theta)\,. \tag{B.58}$$

Let us now study the anticlockwise weight change for the rotation weights. The rotation-cell voltages during anticlockwise movement read:

$$v^{R-}(\theta) = \frac{A_{\text{active}}}{f_{\max}}[h_s * f(v^{a-})](\theta) - \bar{I}_{vel} + I_{inhib}^{\text{HR}} \tag{B.59}$$

$$v^{L-}(\theta) = \frac{A_{\text{active}}}{f_{\max}}[h_s * f(v^{a-})](\theta) + \bar{I}_{vel} + I_{inhib}^{\text{HR}}\,. \tag{B.60}$$

Using eq. (B.50) in eq. (B.59) and eq. (B.60) we find

$$v^{R-}(\theta) = v^{L+}(-\theta) \tag{B.61}$$

$$v^{L-}(\theta) = v^{R+}(-\theta)\,. \tag{B.62}$$

Therefore, applying the same procedure outlined in eq. (B.52)–eq. (B.56), to the anticlockwise change in the rotation weights yields

$$\frac{\mathrm{d}}{\mathrm{d}t}w^{R-}(\theta) = \frac{\mathrm{d}}{\mathrm{d}t}w^{L+}(-\theta) \tag{B.63}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}w^{L-}(\theta) = \frac{\mathrm{d}}{\mathrm{d}t}w^{R+}(-\theta)\,, \tag{B.64}$$

meaning that, during learning, the right and left rotation weights develop mirror symmetry:

$$w^{R}(\theta) = w^{L}(-\theta)\,. \tag{B.65}$$

To verify that our original assumption in eq. (B.49) holds, we compute the axon-distal input for anticlockwise movement:

$$d^{-}(\theta) = [\rho_{\mathrm{HD}} w \star f(v^{a-}) + \rho_{\mathrm{HR}} w^{R} \star f(v^{R-}) + \rho_{\mathrm{HR}} w^{L} \star f(v^{L-})](\theta) + I^{\mathrm{HD}}_{inhib}\,. \tag{B.66}$$

Using Eqs. B.50, B.58, B.59, B.60, B.65 in eq. (B.66), yields

$$d^{-}(\theta) = \rho_{\mathrm{HD}} w \star f(v^{a+}) + \rho_{\mathrm{HR}} w^{L} \star f(v^{L+}) + \rho_{\mathrm{HR}} w^{R} \star f(v^{R+})](-\theta) + I^{\mathrm{HD}}_{inhib} = d^{+}(-\theta)\,. \tag{B.67}$$

Finally, using Eqs. B.57, B.63, and B.64, the total synaptic weight changes for both clockwise and anticlockwise movement read

$$\begin{cases} \frac{\mathrm{d}}{\mathrm{d}t}w(\theta) & = \frac{\mathrm{d}}{\mathrm{d}t}w^{+}(\theta) + \frac{\mathrm{d}}{\mathrm{d}t}w^{+}(-\theta) \\ \frac{\mathrm{d}}{\mathrm{d}t}w^{R}(\theta) & = \frac{\mathrm{d}}{\mathrm{d}t}w^{R+}(\theta) + \frac{\mathrm{d}}{\mathrm{d}t}w^{L+}(-\theta) \\ \frac{\mathrm{d}}{\mathrm{d}t}w^{L}(\theta) & = \frac{\mathrm{d}}{\mathrm{d}t}w^{R}(-\theta)\,. \end{cases} \tag{B.68}$$

Averaging across speeds

So far, we have only considered head turnings at a fixed speed v^{+} (clockwise) and $v^{-} = -v^{+}$ (anticlockwise). However, in the full model described in Chapter II, velocities are sampled stochastically from an OU process. This random process generates a half-normal distribution of speeds with spread $\sigma_v/2$ (fig. B.11, left, see also table 2.1). We thus compute the expected weight changes with respect to this speed distribution:

$$\begin{cases} \frac{\mathrm{d}}{\mathrm{d}t}\langle w\rangle_v(\theta) & := \int_0^\infty \mathrm{d}v\, p(v) \frac{\mathrm{d}}{\mathrm{d}t} w_v(\theta) \\ \frac{\mathrm{d}}{\mathrm{d}t}\langle w^R\rangle_v(\theta) & := \int_0^\infty \mathrm{d}v\, p(v) \frac{\mathrm{d}}{\mathrm{d}t} w^R_v(\theta) \\ \frac{\mathrm{d}}{\mathrm{d}t}\langle w^L\rangle_v(\theta) & := \int_0^\infty \mathrm{d}v\, p(v) \frac{\mathrm{d}}{\mathrm{d}t} w^L_v(\theta) \end{cases} \tag{B.69}$$

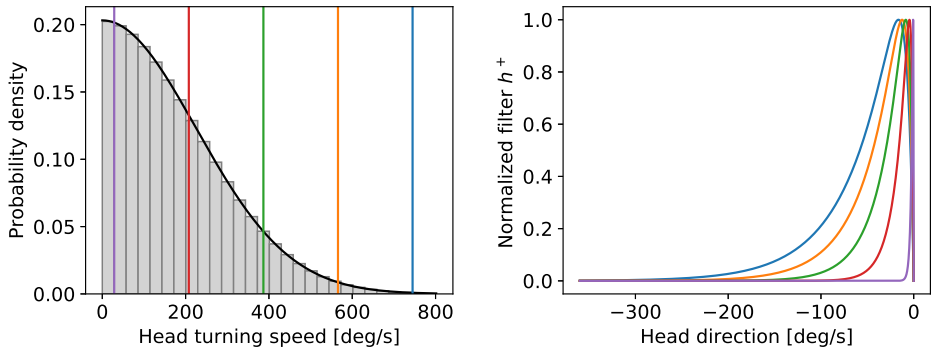

Figure B.11: **Speed distribution and impact on spatiotemporal filter.** Left: assumed distribution of head-turning speeds (black) and discrete approximation used for the simulations. The colored vertical lines indicate speeds for which the filter h^+ is plotted in the right panel. Right: temporal filter $h^+(\theta)$ for several example speeds (see vertical lines in the left panel). Note that even for the largest speeds (blue curve) the filter decays within one turn around the circle.

where w_v is the weight change for speed $|v^+| = |v^-| = v$ and $p(v)$ is an half-normal distribution with spread $\sigma_v/2$.

Simulation of the reduced model

In this section, we show the dynamics of the reduced model numerically simulated according to Eqs. B.46, B.68, and B.69. Weight changes are computed at discrete time steps and integrated using the forward Euler method. At each time step we compute the weight changes for each speed v (eq. (B.46) and eq. (B.68)) and we estimate the expected weight change according to eq. (B.69). We then update the weights and proceed to the next step of the simulation. Note that eq. (B.46) requires the firing rates of HD and HR cells at the previous time step (recurrent input, first line of eq. (B.46)). Therefore, at each time step, we save the HD and HR firing rates for every speed value v and provide them as input to the next iteration of the simulation.

Figure B.12 shows the evolution of the reduced system for 400 time steps, starting from an initial condition where all weights are zero. One can see that from time steps 75 to 100 the system switches from a linear regime (HD firing rates below saturation, see top panel) to a non-linear regime (saturated HD rates). Such a switch is accompanied by peaks in the average absolute error (third panel from the top). Notably, the rotation weights start developing a structure only after such switch has

occurred (see two bottom panels)—a feature that has been observed also in the full model (fig. 2.3E).

Development of the recurrent weights

Figure B.13 provides an intuitive explanation for the shape of the recurrent-weight profiles w that emerge during learning. The first column shows the evolution of the recurrent weights in the linear regime ($t = 25$), i.e., before the HD rates reach saturation. In this regime, both recurrent and rotation weights are small, and the steady-state axon-distal rate

$$f(v^{ss}) \approx f\left(\frac{g_D}{g_D + g_L} I_{inhib}\right) \tag{B.70}$$

is flat and close to zero. Therefore, the HD output rate $f(v^a)$ is dominated by the visual input $\bar{I}_{vis}$ (eq. (B.46), third line), which has the shape of a localized bump (panel A1). Thus the error ϵ has also the shape of a bump (B1). Additionally, the post-synaptic inputs p^+ and p^- are shifted and filtered versions of this bump (eq. (B.46), seventh line). The recurrent weight changes $\mathrm{d}w^+$ and $\mathrm{d}w^-$ for clockwise and anticlockwise movement are given by the cross-correlation of the errors ϵ^+ and ϵ^- with the post-synaptic inputs p^+ and p^- (panel C1; see eq. (B.46) seventh line and eq. (B.52)). Note that because $a(x) \star b(x) = a(-x) * b(x)$, the operation of cross-correlation can be understood graphically as a convolution between the mirrored first function a and the second function b. Such a mirroring is irrelevant in C1 (linear regime) because the error is an even function, but becomes important in C2 (non-linear regime). As a result of this cross-correlation, the recurrent recurrent-weight changes $\mathrm{d}w^+$ and $\mathrm{d}w^-$ are shifted bumps (colored lines in C1), which merge into a single central bump after summing clockwise and anticlockwise contributions (black line in C1). Therefore, in the linear regime, the recurrent weights develop a single central peak in the origin (panel D1).

The second column of fig. B.13 shows the development of the recurrent weights in the non-linear regime (time step 350). Panel A2 shows that in this scenario the HD firing-rate bumps are broader and approach saturation due to the strong recurrent input. The coupling between the axon-distal and axon-proximal compartment acts as a self-amplifying signal during learning which results in the activity of all active neurons participating in the bump reaching saturation. Additionally, because the recurrent input is filtered in time (eq. (B.46), second line), such bumps are also shifted towards the direction of movement. Importantly, due to the lack of visual

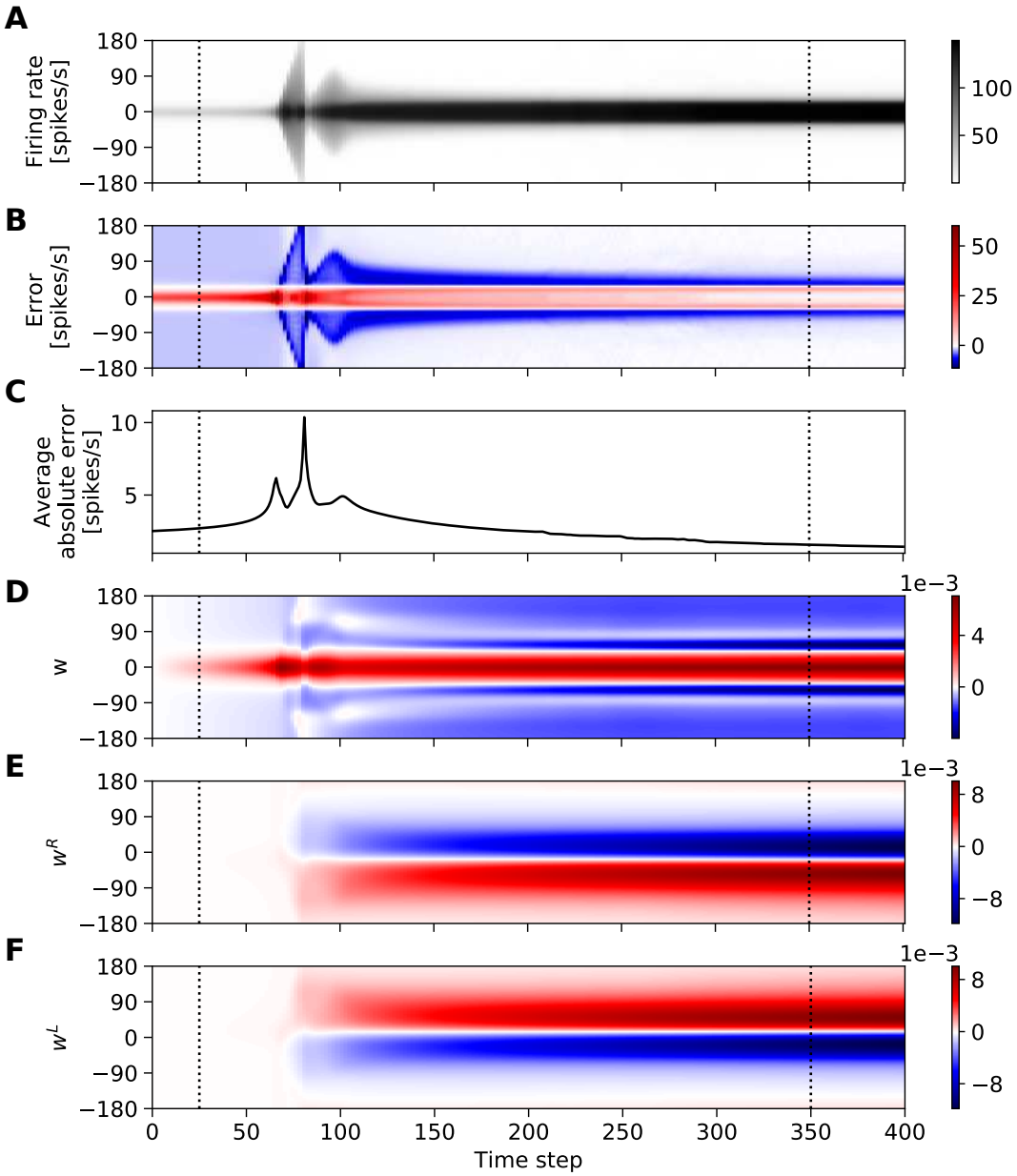

Figure B.12: **Training evolution of the reduced model.** The figure shows from top to bottom: A) the HD-cells' firing rate $f(v^{a+})$; B) the error ϵ; C) the average absolute error; D) the recurrent weights w; E-F) the rotation weights w^R and w^L. The HD firing rate and the errors (panels A-C) are averaged across speeds and and both movement directions. The vertical dashed lines denote the time points shown in fig. B.13 and fig. B.14.

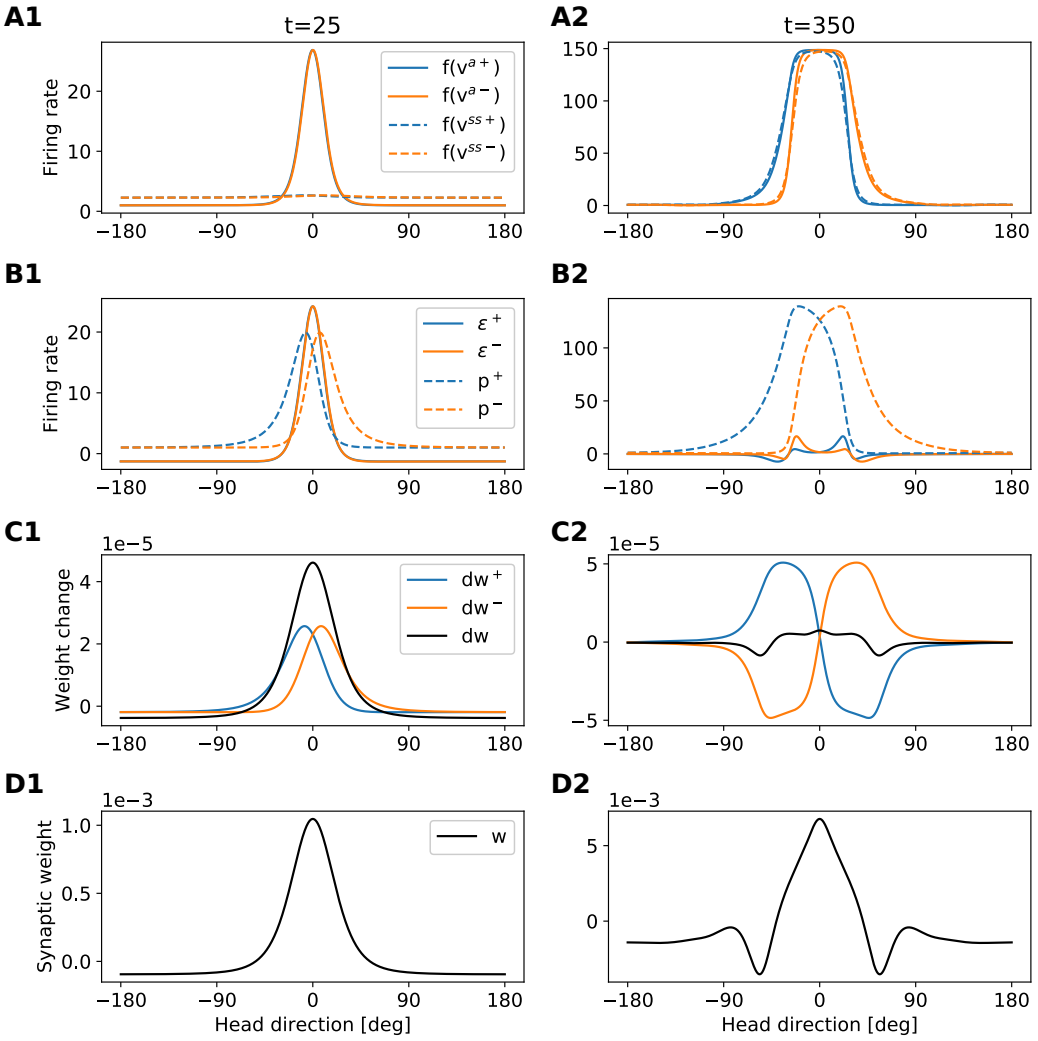

Figure B.13: **Development of the recurrent weights.** The figure provides an intuition for the shape of the recurrent-weights profiles that emerge during learning. Each column refers to a different time step (see also dashed lines in fig. B.12). Each row shows a different set of variables of the model (see legends in the first column). The figure is to be read from top to bottom, because variables in the lower rows are computed from variables in the upper rows. Blue (orange) lines always refer to clockwise (anticlockwise) motion. Black lines in C show the total weight changes for both clockwise and anti-clockwise motion, i.e., $\mathrm{d}w = \mathrm{d}w^{+} + \mathrm{d}w^{-}$.

input, within the receptive field the steady-state axon-distal rates are always smaller than the firing rates. As a result, the errors ϵ^+ and ϵ^- show small negative bumps in the direction of movement, and small positive bumps in the opposite direction (panel B2). Additionally, the post-synaptic inputs p^+ and p^- shift further apart from the origin. Consequently, the total weight change dw develops negative peaks around 60 deg (black line in C2, contrast to panel C1), and these peaks get imprinted in the final recurrent weights' profiles (panel D2).

Development of the rotation weights

Figure B.14 provides an intuitive explanation for the shape of the rotation-weights profiles w^R and w^L that emerge during learning. The first column shows the evolution of the rotation weights in the linear regime ($t = 25$), i.e., before the HD rates reach saturation. In this regime, the rotation-cell firing rates are filtered versions of the HD bumps but re-scaled by a factor $A_{\text{active}}/f_{\max} \approx 0.013$ and baseline-shifted by an amount $\pm\bar{I}_{vel} + I^{\text{HR}}_{inhib}$ (eq. (B.46) lines 4 and 5; panel A1, compare to fig. B.13 panel A1. This baseline shift acts as a switch that determines from which rotation cells population connections will be mainly drawn from, depending on the direction of motion. Panel B1 shows that the errors ϵ^+ and ϵ^- overlap and have the shape of a bump centered at the origin (same curves as in fig. B.13 panel B1). Additionally, the post-synaptic potentials $p^{R\pm}$ and $p^{L\pm}$ in B1 are filtered versions of the curves in A1 (eq. (B.46), lines 7 and 8). As a result, the weight changes d$w^{R\pm}$ and d$w^{L\pm}$, i.e., the errors cross-correlated by the post-synaptic potentials, appear similar to the bumps in A1, but they are smoother and further apart from the origin (panel C1). Finally, such weight changes get imprinted in the rotation weights (panel D1).

The second column shows the evolution of the rotation weights in the non-linear regime ($t = 350$), i.e., after the HD rates reach saturation. In this case, the large recurrent input gives rise to larger rotation rates (A2, compare to A1) and larger post-synaptic potentials (B2, compare to B1). In panel B2, we can see that the errors ϵ^+ and ϵ^- show positive and negatives peaks shifted from the origin (same curves as in fig. B.13 panel B2), which generate weight changes with both positive and negative lobes (panel C2). Such weight changes get finally imprinted in the rotation weights (panel D2).

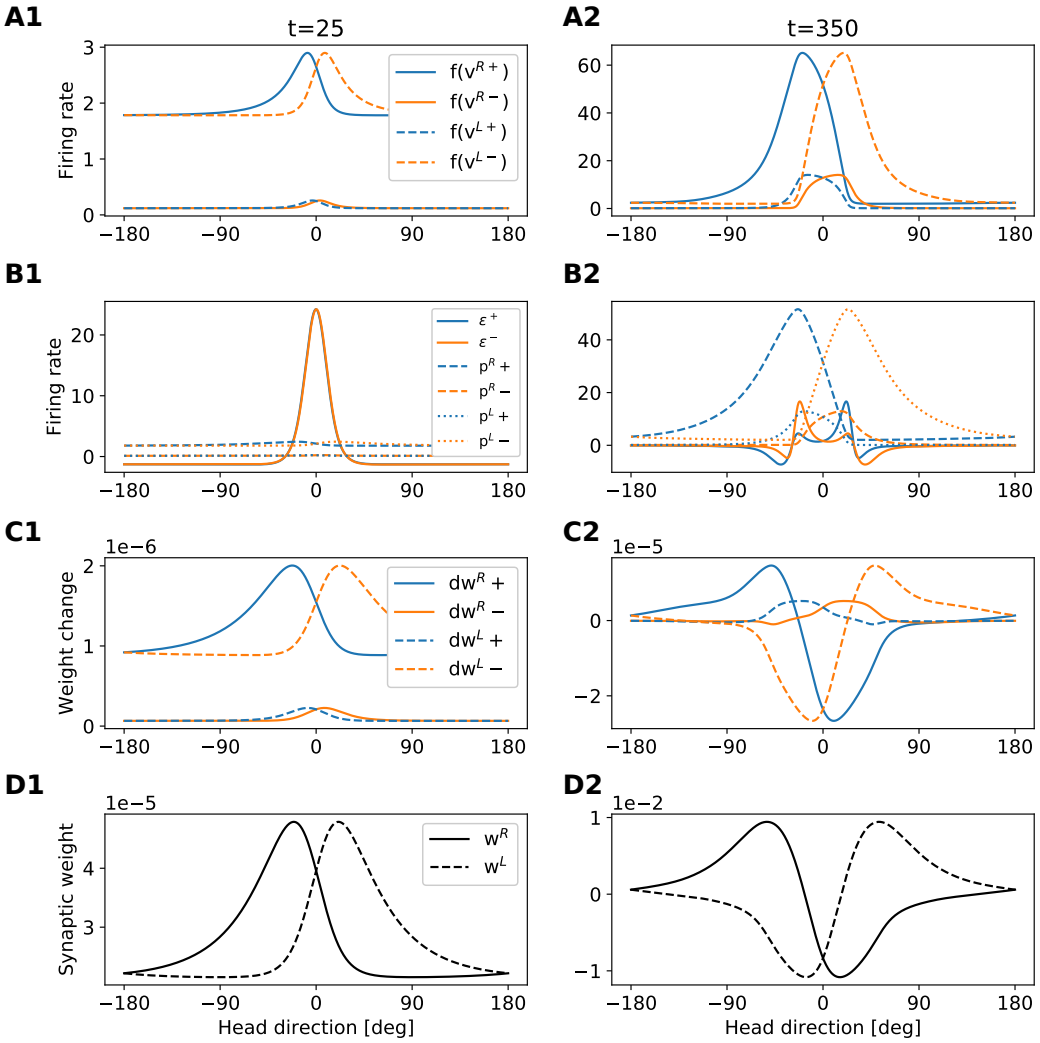

Figure B.14: **Development of the rotation weights.** The figure provides an intuition for the shape of the rotation-weights profiles that emerge during learning. Each column refers to a different time step (see also dashed lines in fig. B.12). Each row shows a different set of variables of the model (see legends in the first column). The figure is to be read from top to bottom, because variables in the lower rows are computed from variables in the upper rows. Blue (orange) lines always refer to clockwise (anticlockwise) motion.

References

Seelig, Johannes D. and Vivek Jayaraman (May 2015). "Neural dynamics for landmark orientation and angular path integration." In: *Nature* 521.7551, pp. 186–191. DOI: 10.1038/nature14446. URL: https://doi.org/10.1038/nature14446.

Turner-Evans, Daniel B. et al. (May 2017). "Angular velocity integration in a fly heading circuit." In: *eLife* 6. Ed. by Alexander Borst, e23496. ISSN: 2050-084X. DOI: 10.7554/eLife.23496. URL: https://doi.org/10.7554/eLife.23496.

Zhong, Weishun et al. (June 2020). "Nonequilibrium statistical mechanics of continuous attractors." In: *Neural Computation* 32.6, pp. 1033–1068. DOI: 10.1162/neco_a_01280. URL: https://doi.org/10.1162/neco_a_01280.

Appendix C

SUPPLEMENTARY MATERIAL FOR CHAPTER III

C.1 Context-dependent computation is not state-space efficient

The workhorse model for computational neuroscience has traditionally been context-dependent computation, where tasks are carried out one at a time and task identity is cued to the RNN by a one-hot vector (Mante et al., 2013). However, this approach can be algorithmically inefficient, because as we show here it scales linearly with the number of tasks N_{task}, and exponentially with input dimensionality D. This is because context-dependent computation utilizes different parts of the state space for different tasks, and the resulting representations collapse to what is minimally required for each task (also see (Mante et al., 2013; Yang and Wang, 2020)). This can be detrimental for brains, which need to pack a lot of computation within a large yet limited neural substrate. In contrast, abstract representations are general, compact (Ma, Tsao, and Shum, 2022), can be used for **any** downstream task involving the same variables, scale linearly with D, and as we show readily emerge from relatively simple tasks.

To compare context-dependent decision making, where one task is performed at a time (Mante et al., 2013), to multitasking, in terms of state-space usage efficiency, we train RNNs to perform context-dependent decisions on the same tasks encountered in Chapter III. Compared to the network in Chapter III (fig. 3.2b), the RNN now also receives a one-hot task rule vector indicating the current task, and it outputs the decision for that task only (fig. C.1b). We have also omitted the non-linear encoder, making the tasks easier. We train the RNN for two tasks, one task at a time, in interleaved batches (fig. C.1a). In one task, the RNN is required to decide which stream has more evidence, and at the other whether the sum of evidence across streams exceeds a certain decision threshold (here 0).

We find that in this setting the network is not incentivized to learn abstract representations. Instead a separate line attractor is present in the dynamics for each task (red and orange x's in fig. C.1c); one of them is presumably tracking the difference of evidence (similar to Yang and Wang (2020) but for independent evidence streams) and the other the sum of evidence. That is to say, the task rule biases the network to learn different computations in separate regions of state space, as in Mante et al. (2013). As a result, the 2D latent space has collapsed and cannot be decoded from network activity; therefore generalization to any task that involves these two variables is not possible.

It follows that the network can be inefficient in terms of state space usage, because instead of compressing all of its activity around the same region, it spreads it across

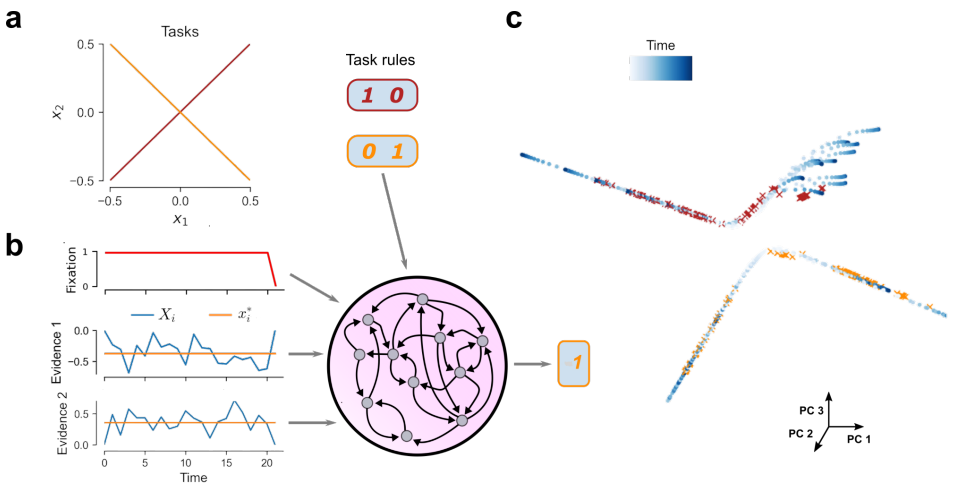

Figure C.1: **Representations in an RNN trained in context-dependent decision making.** **(a)** We trained RNNs for two classification tasks: two-alternative forced choice (where the decision boundary is the $x_1 = x_2$ line) and evidence integration (corresponding to the $x_1 = -x_2$ line). Each task corresponds to a different one-hot task rule vector. **(b)** Network architecture. In addition to the inputs in fig. 3.2b, the network also a one-hot vector indicating the current task. **(c)**, Top 3 PCs of RNN activity example trials (40 in total). The task rule biases the network towards learning separate solutions in different parts of the state space for different tasks, in the form of separate line attractors; red x's for two-alternative forced choice and orange x's for evidence integration.

multiple regions, one for each task, which scales badly (linearly with the number of tasks N_{task} and exponentially with input dimensionality D). To demonstrate the latter, imagine a family of tasks with classification boundaries of the form $\oplus x_1 \oplus x_2 \oplus .. \oplus x_D = 0$, where $\oplus \in \{+1, -1, 0\}$ is an operator indicating contribution with a positive sign, negative sign or absence of contribution for a factor to a specific task, respectively. As just shown, each one of this tasks will require its own line attractor, resulting in a total of 3^D line attractors lying in separate regions of the state space, just for this simple family of tasks. As mentioned in Chapter III, such inefficiency can be detrimental for brains, which need to pack a lot of computation within a large yet limited neural substrate. Compare that to multitasking, which builds representations that can serve any task that involves the same latent variables, scaling linearly with D (as we saw that we only need $N_{task} \geq D$ to learn them). Note that context-dependent computation can still be efficient, if tasks have a compositional structure where the solution for one task is

part of the solution for another (Yang et al., 2019; Driscoll, Shenoy, and Sussillo, 2022); in this case, representations developed for the former can act as a scaffold for representations for the latter.

Overall, we believe that multitasking may present a paradigm swift for generalizable representation learning in biological and artificial systems alike. That is not to say that context-dependent representations are not useful; they are great at leveraging the compositional structure of tasks (Yang et al., 2019; Driscoll, Shenoy, and Sussillo, 2022), but tend to overfit to the specifics of the task, while multitask representations serve as world models applicable to various scenarios. Both types of representations are likely to be found in the brain. One possibility is that context-dependent representations may emerge as a first quick solution to a task, while disentangled representations come about with more experience or when more tasks are performed over time to support better generalization.

C.2 Robustness to other noise distributions and correlated inputs

We here show that our setting is robust to Gaussian anisotropic and autocorrelated noise, and other asymmetric distributions of noise (Gumbel) whose CDF no longer matches the sigmoid functions in shape, with almost no drop in performance, and correlated inputs. This demonstrates that abstract representations are also learned outside of the specific assumptions made by our theory.

Starting from anisotropic noise, we observe that doubling the standard deviation of noise across one dimension ($\sigma = 0.4$) does not result in a reduction in OOD generalization performance (median $r^2 = 0.96$ for $N_{task} = 24$, $D = 2$). This is in line with our theory that can be extended to cover anisotropic noise (Lemma C.6.11). Non-IID noise should not be a problem either, since we are training our network for many samples and the effects of correlations will cancel out over long ensembles. Indeed, we find that including autocorrelated AR(1) noise with an AR coefficient of 0.7 results in only minor reduction in performance (median $r^2 = 0.95$ for $N_{task} = 24$, $D = 2$).

We were also curious to see the impact of correlated inputs. A problem with high correlations is that they render parts of the state space virtually invisible to the network (fig. C.2a). Surprisingly, OOD generalization performance is very weakly affected by input correlations, even though the state space is sampled uniformly in test (fig. C.2b). The behavior is highly non-linear: performance is great until $\rho = 0.97$, but for perfectly correlated inputs ($\rho = 1$), the performance drop is sharp.

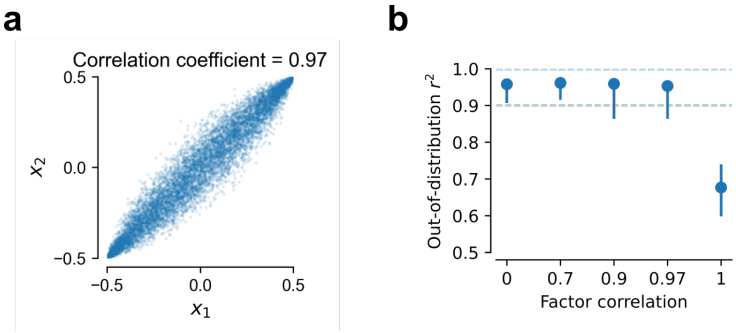

Figure C.2: **Disentanglement and factor correlations.** **(a)** We introduce strong correlations in the latent factors, rendering parts of the state space virtually invisible to the network during pre-training (trained for a total of 24 classification tasks). **(b)** Despite that, generalization performance is excellent for correlations very close to 1. Once the factors are perfectly correlated, performance drops significantly. This implies that the network can learn an abstract representation from correlated inputs, as long as there is some signal about the factors independently. This finding goes beyond (Johnston and Fusi, 2023) to show that the multi-task learning setting allows OOD generalization when the distribution during training the RNN itself is vastly different that the one during testing.

Finally, our theory pointed out sigmoid functions as a choice for activation function because of their close resemblance to the Gaussian CDF, resulting in the best OOD r^2. Still we find that for an asymmetric noise distribution (Gumbel) whose CDF does not match sigmoid functions well, there is only a slight drop in performance (median $r^2 = 0.95$ from 0.96). Therefore the conditions for the activation function/CDF should be quite lax; any monotonic bijective function should work with small performance drop. This drop in performance is because the representation would be "stretched out" and "compressed" in a non-linear manner in regions where there is discrepancy between the noise CDF and the activation function. But this nonlinear squishing (determined by the term $\Phi^{-1}(g(\mathbf{Z}(t)))$) would be geometrically inoffensive — no cutting or gluing together would be required to map from $\mathbf{Z}(t)$ to a linear representation of $\mu(t)$. As a result, the representations would remain approximately linearly decodable. Monotonicity and bijectivity are quite mild assumptions for the activation function used by neurons in the brain.

C.3 Nonlinear classification boundaries and interleaved learning

In Chapter III we trained networks on linear classification boundaries. The tasks are still non-linear, since the encoder renders these boundaries non-linear to the network.

However, there are cases where the latent factors themselves might need to be combined non-linearly, to make decisions. For instance, if the two factors represent the amount and probability of reward, respectively, an agent needs to multiply the two and decide whether the expected value exceeds a certain (metabolic) cost γ of performing an action to obtain said reward. Figure C.3a shows the classification lines for the multiplicative task, where the network should decide whether the ground truth $\mathbf{x}^*$ lies above or below the curve $x_1\, x_2 = \gamma$, for multiple values of γ. This family of tasks is not covered by theorem C.6.6, because they violate the injectivity condition. Hence, we wondered how the representation would look like if the network was trained on both the linear and multiplicative boundaries, as animals do.

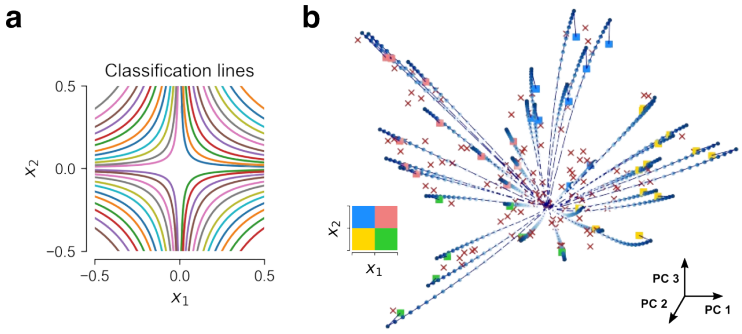

Figure C.3: **Interleaved learning of linear and non-linear boundaries. (a)** Classification lines for the multiplicative task. There is a total of 48 classification lines, 12 per quadrant. **(b)** The network learns an abstract representation when trained for the linear and multiplicative boundaries in interleaved batches.

For that we perform interleaved training of both tasks (i.e., train in batches sampled from one of the tasks at a time), a setting where neural networks excel at, compared to humans who excel at blocked training, where tasks are learned sequentially (but see Flesch et al. (2022)). Figure C.3b shows that the network still learns an abstract, two-dimensional continuous attractor. OOD generalization for this network is excellent, and almost identical to ID performance (median $r^2 = 0.94, 0.97$, respectively). Overall, we conclude that our framework extends to interleaved learning of a mixture of linear and nonlinear boundaries, which better reflects the challenges encountered by agents in the real world. Note that during interleaved training, linear and non-linear tasks are not performed simultaneously; yet they are in immediate succession which can also place pressure to the network to gradually learn representations

that satisfy all tasks. The relation between multi-task and interleaved learning is a promising topic for future research.

C.4 Abstract representations are learned for a free reaction time, integrate to bound task

In Chapter III we trained networks to produce a response at the end of the trial. However, in many situations agents are free to make a decision whenever they are certain enough. Therefore, we here seek to extend our framework to free reaction time (RT) decisions. A canonical model accounting for choices and reaction times in humans and animals is the drift-diffusion model (Krajbich, Armel, and Rangel, 2010; Brunton, Botvinick, and Brody, 2013). It is composed of an accumulator that integrates noisy evidence over time, until a certain amount of certainty, represented by a bound, is reached, triggering a decision. In the linear classification task setting, the accumulated amount of evidence at time t for a line with slope α, $A_\alpha(t)$ is given by:

$$A_\alpha(t) = A_\alpha(t-1) + X_1(t) - \alpha\, X_2(t). \tag{C.1}$$

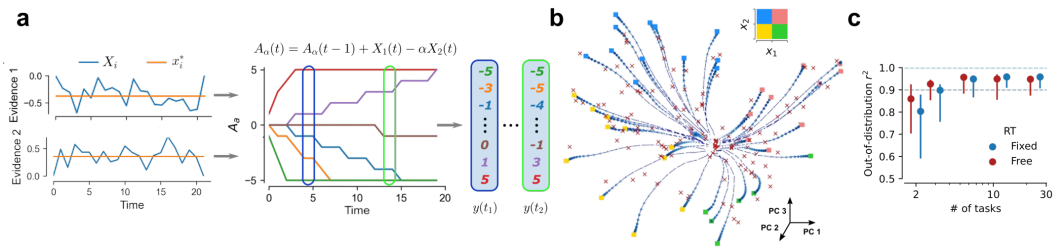

Figure C.4: **Free reaction time task.** **(a)** Data generating process. Every classification line from fig. 3.2a now corresponds to an accumulator (see corresponding colors), and the desired output for the RNN is the accumulator values for the entire trial. The accumulator is quantized to integer values between ±5. **(b)** Representation for RNN trained on free reaction time task. The network learns a two-dimensional continuous attractor, similar to fig. 3.3d. A 3D rotating figure to better visualize this representation is provided in the Supplementary Material. **(c)** OOD generalization performance for the free reaction time (RT) task. Free RT outperforms fixed RT for a small number of tasks.

Intuitively, $A_\alpha(t)$ reflects the amount of **confidence** at time t that the ground truth $\mathbf{x}^*$ lies above or below the classification line with slope α. Essentially, the network has to explicitly report distance from the classification lines, not just in which side of the line $\mathbf{x}^*$ lies for that trial. We set the decision bound to ±5, and plot the accumulators

A_α for all lines in fig. 3.2a. Note that once the bound is reached a decision is effectively made and A_α is kept constant. Also, instead of using continuous values, we quantize A_α, because it is going to be used as target signal to train the network, and we do not want to introduce a strong inductive bias towards integrating the evidence streams.

We then train the RNN to reproduce confidence estimates from fig. C.4a for the entire trial. Compared to previous experiments, the fixation input is no longer available to determine when to produce a decision. Instead, decisions evolve dynamically throughout the trial. We also use a MSE loss, change the activation function to $g = 5\tanh$, and the Adam learning rate $\eta_0 = 3 * 10^{-3}$, but all other parameters remain the same as in Chapter III. To have a closer correspondence to the free RT experiments here, we also train the fixed RT task from Chapter III with MSE loss, symmetric labels $\mathbf{y}(\mathbf{x}^*) \in \{-1, +1\}^{N_{task}}$ and output non-linearity $g = \tanh$. We find that the change of objective and loss only has minor effects on generalization performance.

Figure C.4b shows that in this setting the network still learns a two-dimensional continuous attractor of the latent space. Furthermore, the free RT outperforms the fixed RT network from Chapter III (fig. C.4c) for a small number of tasks, since it is explicitly required to report distance from the classification lines. However, as our theory shows (Lemma C.6.3)) the fixed RT network is also implicitly reporting distance from the boundaries, when behaving like an optimal multi-task classifier, which explains the similar performance for a larger number of tasks. Overall, we showed that our setting accounts for naturalistic free RT decisions, and provides theoretical justification for the importance of confidence signals in the brain (Rutishauser et al., 2018; Masset et al., 2020).

The importance of the confidence (i.e., calibrated likelihoods) of a network's output, is a recurring theme in machine learning too (e.g., knowledge distillation (Bhargava et al., 2024)). **We here show that confidence fundamentally connects to how neural networks construct world models**, either directly (integrate-to-bound task here) or indirectly (classification tasks in Chapter III). Under this framework, knowledge distillation can be cast as smaller models directly copying the world models (logits) of larger ones.

C.5 Quantification of sparsity

In Chapter III, we observed that RNN representations are sparse. We here seek to more precisely quantify the sparsity in these networks, and investigate how it is affected by the number of tasks N_{task}, latent dimensionality D, and specific recurrent architecture. To do so, we sample $n = 1000$ ground truth vectors $\mathbf{x}^*$ randomly for every network, and compute the sparseness (Vinje and Gallant, 2000) of a neuron in the hidden layer as:

$$S = \frac{1 - \left(\frac{\sum(z_i/n)^2}{\sum(z_i^2/n)}\right)}{1 - \frac{1}{n}} * 100\,\% \tag{C.2}$$

where z_i is the steady-state response of the neuron to ground-truth stimulus i. Sparseness ranges from 0 to 100 %, with greater sparseness indicating greater selectivity of the neuron to stimuli. Then, the sparsity of a network is given as the average of the sparseness of all its neurons.

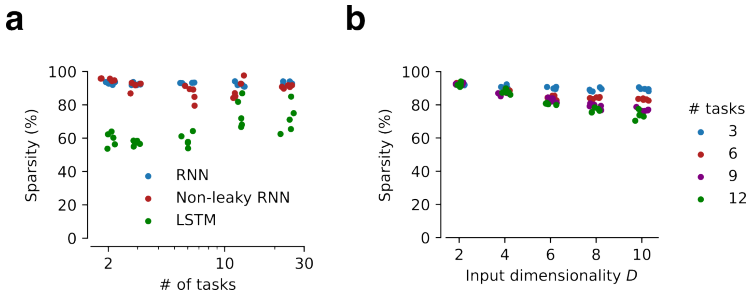

Figure C.5: **Quantification of sparsity as a function of N_{task}, D and recurrent architecture choice. (a)** Sparsity of a recurrent network as a function of number of tasks and network architecture. Five networks trained for each network configuration. Greater levels of sparsity indicate that the network activations are more sparse. **(b)** Sparsity of RNNs as a function of number of tasks and latent dimensionality D. Five network are trained for each combination of (N_{task}, D).

Figure C.5a shows that RNNs and non-leaky RNNs are very sparse, with sparsity values around 90 % for different values of N_{task}, supporting the claim in Chapter III. LSTMs on the other hand, which are less brain-like[1], have lower sparsity values, although interestingly sparsity increases with N_{task}. Notably, we did not do anything

[1]LSTMs architecturally enforce intricate, high-capacity multiplicative gating mechanisms, while in biological neural networks gating has to be learned. For other aspects of biological implausibility of LSTMs compared to RNNs, see Appendix B in (Soo, Goudar, and Wang, 2023).

to promote sparsity (e.g., regularization) in these networks. Therefore we conclude that sparsity naturally emerges from the optimization objective of multitask learning, particularly in architectures that are more brain-like.

Next we wondered how latent space dimensionality D would affect sparsity in our trained RNNs. Figure C.5b shows that networks remain very sparse for the whole range of dimensionality D tested in Chapter III, with sparsity values above 75 %. Greater dimensionality results in less sparsity on average, which is expected since $N_{neu} = 64$ in our networks, therefore a significant amount of their capacity must be used as D increases. This effect plays in only as N_{task} increases, as networks will only learn to disentangle the input dimensions that are spanned by the tasks, as our theory predicts. Overall, there seems to be a proportional relationship between the number of active neurons and dimensionality D, as long as there are enough tasks to uncover the D latents.

C.6 Theoretical Derivations

Here we prove our main theoretical result outlined in Section 3.3.

High-level summary of proof We prove that competence at N_{task} tasks guarantees linear decodability when $N_{\text{task}} \geq D$ for non-degenerate tasks (theorem C.6.5), and orthogonality when $N_{\text{task}} \gg D$ and task boundaries are sampled randomly (Corollary C.6.10). To that end, we first show that optimal evidence aggregation in a multi-task classification framework enforces the multi-task classifier to encode a notion of distances from classification lines (Lemma C.6.3). Given a suitable set of distances from classification lines, we show that one can uniquely identify an optimal estimate of the latents given noisy data in closed form (Trilateration Theorem, Theorem C.6.4). In addition, if the readout function of the multi-task classifier is sigmoid-like, the optimal estimate will be approximately linearly decodable from the representation (Corrolary C.6.8,C.6.9). We then prove by contradiction that all the above results hold when an arbitrary injective observation map f is applied to the input after noising (theorem C.6.6). The theory generalizes to sub-optimal classifiers via least-mean squares approximation and the Moore-Penrose Pseudoinverse (theorem C.6.7), and to different noise distributions (Section C.6). Finally, we discuss implications of the theorem for representation learning, manifold learning and the Platonic representation hypothesis, and future directions (Section C.6).

Notation: *lower case variables denote scalars (e.g., x), upper case variables denote random variables (e.g., X), and boldfaced variables denote vector quantities (e.g., $\mathbf{x}, \mathbf{X}$). We denote the $D \times D$ identity matrix as $\mathbf{I}_D$.*

Variable Glossary:

- $\mathbf{x}^* \in \mathbb{R}^D$: Ground truth (un-noised) input variable of dimension D.
- $\mathbf{X}(t) \sim \mathbf{x}^* + \sigma\mathcal{N}(\mathbf{0}, \mathbf{I}_D)$ are i.i.d. noisy measurements of $\mathbf{x}^*$, where
  - σ is the amount of equivariant Gaussian noise, and
  - t is the discrete time index within a trial.
- $f : \mathbb{R}^D \to \mathcal{Z}$: An injective observation map that transforms the noisy measurements $\mathbf{X}(t)$ before they reach the latent state $\mathbf{Z}(t)$ of the optimal estimator. The map f is injective, meaning that it preserves the uniqueness of

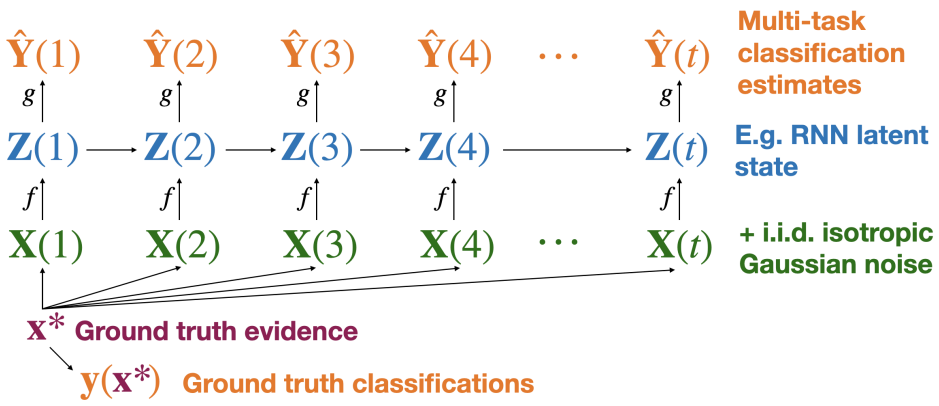

Figure C.6: **Bayesian graphical model framework representing our theoretical framework for multi-task classification.** The agent with latent state $\mathbf{Z}(t)$ estimates the ground truth decision output $\mathbf{y}(\mathbf{x}^*) \in \{0,1\}^{N_{\text{task}}}$ from noisy observations $\mathbf{X}(t)$ transformed by injective observation map f. We prove that latent state $\mathbf{Z}(t)$ must encode an optimal, linearly decodable estimate of the de-noised environment state $\mathbf{x}^*$ when the decision boundary normal vectors $\{\mathbf{c}_i\}_{i=1}^{N_{\text{task}}}$ span $\mathbb{R}^D$.

the input, i.e., if $f(\mathbf{x}_1) = f(\mathbf{x}_2)$, then $\mathbf{x}_1 = \mathbf{x}_2$. The codomain $\mathcal{Z}$ can be any suitable space, such as $\mathbb{C}^M$, $\mathbb{R}^\infty$, or other spaces.

- N_{task} is the number of classification tasks,
- $\{(\mathbf{c}_i, b_i)\}_{i=1}^{N_{task}}$ are the classification boundary normal vectors and offsets respectively, with $\mathbf{c}_i \in \mathbb{R}^D$ and $b_i \in \mathbb{R}$. We assume each $\|\mathbf{c}_i\| = 1$.
- $(\mathbf{C}, \mathbf{b})$ are a matrix and vector representing each of the N_{task} classification tasks where $\mathbf{C} \in \mathbb{R}^{N_{task} \times D}$
- $\mathbf{y}(\mathbf{x}^*) \in \{0,1\}^{N_{task}}$: Ground truth classification outputs, where each ground truth classification $y_i(\mathbf{x}^*)$ is given by

$$y_i(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{c}_i^\top \mathbf{x} > b_i \\ 0 & \text{otherwise} \end{cases} \tag{C.3}$$

- $\mathbf{Z}(t)$: Latent variable of a multi-task classification model, conditional on $\mathbf{X}(1), \ldots, \mathbf{X}(t)$.
- g : Map from latent state $\mathbf{Z}(t)$ to multi-task classification estimates $\hat{\mathbf{Y}}(t)$. For most of our experiments, readout map g = sigmoid, for instance.

- $\hat{\mathbf{Y}}(t) := g(\mathbf{Z}(t)) \in [0,1]^{N_{task}}$: Output vector of the multi-task classification model at time t, where each $\hat{Y}_i(t)$ is a Bernoulli random variable estimator, estimating the conditional probability $\Pr\{y_i(\mathbf{x}^*) = 1\}$ given the noisy observations (via latent variable $\mathbf{Z}(t)$ – see Equation C.4).

- $\hat{\mathbf{X}}(t) = \mathcal{N}(\mu(t), \Sigma(t))$: Optimal estimate of $\mathbf{x}^*$ given measurements $\mathbf{X}(1), \ldots, \mathbf{X}(t)$, derived in Lemma C.6.1.

Problem Statement: We consider optimal estimators of $\mathbf{y}(\mathbf{x}^*)$ in the multi-classification paradigm in Equation C.4, shown graphically in Figure C.6.

$$\mathbf{x}^* \to \mathbf{X}(1), \ldots, \mathbf{X}(t) \xrightarrow{f} \mathbf{Z}(t) \xrightarrow{g} \hat{\mathbf{Y}}(t) \tag{C.4}$$

Contribution: We prove in Theorem C.6.6 ("Optimal Representation Theorem") that any optimal estimator of $\mathbf{y}(\mathbf{x}^*)$ described above will represent an optimal estimate of $\mathbf{x}^*$ in latent state $\mathbf{Z}(t)$. We begin by proving results on optimal estimators in Sections C.6, C.6 with identity observation map f, developing the linear case of the optimal representation theorem (Theorem C.6.5) showing that the latent state $\mathbf{Z}(t)$ must encode an estimate of $\mathbf{x}^*$ (visualized in Figure C.7). We generalize this result any injective observation map f in Section C.6 and derive closed-form solutions for extracting the estimate of $\mathbf{x}^*$ from $\mathbf{Z}(t)$. We derive approximation results for $g = \tanh$ in Corollary C.6.8 and $g =$ sigmoid in Corollary C.6.9 that show the representation of $\mathbf{x}^*$ in $\mathbf{Z}(t)$ will be linear-affine decodable if g is in the sigmoid family of functions.

Single Decision Boundary

First, we will derive $\hat{Y}(t)$ for a single decision boundary with parameters $(\mathbf{c}, b)$. We focus on $P(\hat{Y}(t)|\mathbf{X}(1), \ldots, \mathbf{X}(t))$, reintroducing the latent variable $\mathbf{Z}(t)$ later on.

Since $y(\mathbf{x}^*)$ is a deterministic function of non-random variable $\mathbf{x}^*$, we will derive the probability distribution over $P(\mathbf{x}^*|\mathbf{X}(1), \ldots, \mathbf{X}(t))$ – denoted $\hat{\mathbf{X}}(t)$ – to determine $\hat{Y} = y(\hat{\mathbf{X}}(t))$. [2]

Lemma C.6.1. *Assuming no prior on* $\mathbf{x}^*$*, the conditional probability distribution* $\hat{\mathbf{X}}(t) \sim P(\mathbf{x}^*|\mathbf{X}(1), \ldots, \mathbf{X}(t))$ *is given by*

$$\hat{\mathbf{X}}(t) = \mathcal{N}(\mu(t), \Sigma(t)) \tag{C.5}$$

[2] Note that the intermediate computation of $\hat{\mathbf{X}}(t)$ does not imply that a system *must* compute this value to predict $\hat{Y}$, as the full computation of $\hat{\mathbf{X}}(t)$ may not be necessary to determine $\hat{Y}(t)$.

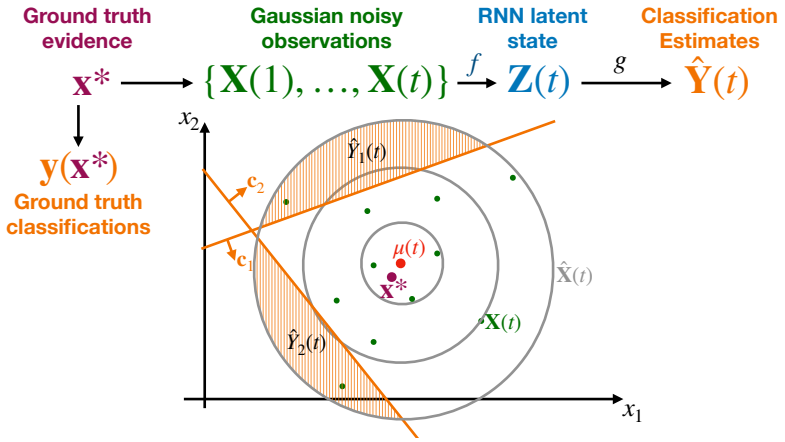

Figure C.7: **An overview of the classification process using an RNN with Gaussian noisy observations.** The ground truth $\mathbf{x}^*$ generates the noisy observations $\{\mathbf{X}(1), ..., \mathbf{X}(t)\}$. These observations are processed by the filter-based model illustrated graphically in Figure C.6, maintaining a latent state $\mathbf{Z}(t)$. The latent state $\mathbf{Z}(t)$ is then used to produce classification outputs $\hat{Y}_1(t)$ and $\hat{Y}_2(t)$. Theorem C.6.6 proves that $\mathbf{Z}(t)$ must encode an estimate of $\mathbf{x}^*$, visualized in this figure, shown as $\hat{\mathbf{X}}^*$, including its mean $\mu(t)$, which is the optimal estimator for $\mathbf{x}^*$ given the noisy observations.

where $\mu(t) = mean(\mathbf{X}(1), \ldots, \mathbf{X}(t))$ *and* $\Sigma(t) = t^{-1}\sigma^2\mathbf{I}_D$.

Proof. Since $\mathbf{X}(1), \ldots, \mathbf{X}(t)$ are i.i.d. from a Gaussian distribution with mean $\mathbf{x}^*$ and identity covariance, the sample mean is known to be distributed normally centered at the ground truth $\mathbf{x}^*$. We apply the known standard deviation of the underlying distribution (identity covariance scaled by σ) to arrive at $\Sigma(t) = t^{-1}\sigma^2\mathbf{I}_D$ as the variance on the sample mean (derived from the central limit theorem). □

We can use estimator $\hat{\mathbf{X}}(t)$ to construct $\hat{\mathbf{Y}}(t)$ by expanding $\hat{\mathbf{Y}}(t) = y(\hat{\mathbf{X}}(t))$ via Equation C.3.

In essence, we are interested in the amount of the probability density of $\hat{\mathbf{X}}$ that lies on each side of the decision boundary. Deriving this probability is simplified by the fact that $\hat{\mathbf{X}}$ is isotropic—i.e., it inherits the spherical covariance of the underlying data generation process (Lemma C.6.2).

Lemma C.6.2. $\hat{\mathbf{X}}(t) = \mathcal{N}(\mu(t), \Sigma(t))$ *with isotropic covariance* $\Sigma(t) = t^{-1}\sigma^2\mathbf{I}_D$ *and mean* $\mu(t) \in \mathbb{R}^D$. *The probability density of* $\hat{\mathbf{X}}(t)$ *on the positive side of the*

decision boundary $\{\mathbf{x} : \mathbf{c}^\top \mathbf{x} > b\}$ *can be expressed as*

$$\hat{Y}(t) \triangleq \Pr\{\mathbf{c}^\top \mathbf{x}^* > b\} = \Phi(k\sqrt{t}/\sigma) \tag{C.6}$$

where Φ *is the CDF of the normal distribution and* $k = \mathbf{c}^\top \mu(t) - b$ *is the signed projection distance between the decision boundary and the mean* $\mu(t)$ *of* $\hat{\mathbf{X}}(t)$.

Proof. Since the $\hat{\mathbf{X}}(t)$ is isotropic, the variance on every axis is equal and independent. We may rotate our coordinate system such that the projection line between the plane and the mean of $\hat{\mathbf{X}}(t)$ aligns with an axis we denote as "axis 0." The rest of the axes must be orthogonal to the plane. Since each component of an isotropic Gaussian is independent, the marginal distribution of $\hat{\mathbf{X}}(t)$ on axis 0 is a univariate Gaussian with variance $t^{-1}\sigma^2$ mean at distance k from the boundary. Equation C.6 applies the normal distribution CDF Φ to determine the probability mass on the positive side of the boundary. □

Observe that $\hat{Y}(t)$ in Equation C.6 **monotonically scales** with the signed distance k between the hyperplane and $\mu(t)$ (CDFs are monotonic).

Lemma C.6.3. *Knowledge of time t and optimal classification estimate* $\hat{Y}(t)$ *is sufficient to determine the projection distance k between* $\mu(t) = mean\big(\mathbf{X}(1), \ldots, \mathbf{X}(t)\big)$ *and the decision boundary* $(\mathbf{c}, b)$.

Proof. Recall Equation C.6 from Lemma C.6.2. We may solve for projection distance k separating the decision boundary and the mean $\mu(t)$ of observations $\mathbf{X}(1), \ldots, \mathbf{X}(t)$ as

$$k = \frac{\sigma}{\sqrt{t}} \Phi^{-1}(\hat{Y}(t)) \tag{C.7}$$

Since Φ is the CDF of the normal distribution, and the normal distribution is not zero except at $\pm\infty$, the inverse Φ^{-1} is well-defined. □

Note that non-zero noise is required for Lemma C.6.3 to hold, as zero noise would yield zero probability mass on one side of each decision boundary, meaning that no distance information would be recoverable from $\hat{Y}(t)$ (and eq. (C.7) would lead to a $0 \cdot \infty$ indeterminacy).

Trilateration via Multiple Decision Boundaries

To recap Section C.6 : We derived an optimal estimator of $\mathbf{x}^*$ (denoted $\hat{\mathbf{X}}(t)$) based on noisy i.i.d. measurements $\mathbf{X}(1), \ldots, \mathbf{X}(t) \sim \mathcal{N}(\mathbf{x}^*, \sigma^2 \mathbf{I}_D)$ in Lemma C.6.1. In Lemma C.6.2 we derived the equation for Bernoulli variable estimator $\hat{Y}(t)$ to estimate a single classification output $y(\mathbf{x}^*)$ based on the same noisy measurements via $\hat{\mathbf{X}}(t)$. Finally, we showed in Lemma C.6.3 that the uncertainty in $\hat{Y}(t)$ and the time t is sufficient to determine the projection distance between the decision boundary and $\mu(t) = \text{mean}(\mathbf{X}(1), \ldots, \mathbf{X}(t))$ via Equation C.7.

Let $\hat{\mathbf{Y}}(t)$ denote the vector of classification estimates $\hat{Y}(t)$ from Equation C.7. We now have the tools to prove our final result via **trilateration**. Much like distance information from cell towers can be used to trilaterate[3] one's position, we will leverage Lemma C.6.3 and use distances from decision boundaries $\{(\mathbf{c}_i, b_i)\}_{i \in [N_{task}]}$ to constrain the positions.

Theorem C.6.4 (Trilateration Theorem)**.** *If* $\mathbf{C} \in \mathbb{R}^{N_{task} \times D}$ *is full-rank and* $N_{task} \geq D$*, then* $\hat{\mathbf{Y}}(t)$*,* t*,* $\mathbf{b}$*, and* $\mathbf{C}$ *are sufficient to reconstruct the exact value of* $\mu(t)$*, the mean of* $\mathbf{X}(1), \ldots, \mathbf{X}(t)$*, which is also the optimal estimator for* $\mathbf{x}^*$*.*

Proof. We may prove this claim by providing an algorithm to reconstruct $\mu(t) = \text{mean}(\mathbf{X}(1), \ldots, \mathbf{X}(t))$ from $\hat{\mathbf{Y}}(t), \mathbf{C}$, and t. Invoke Lemma C.6.3 to compute the signed projection distance between $\mu(t)$ and each decision plane $(\mathbf{c}_i, b_i)$. Let $\mathbf{k} = [k_1, \ldots, k_{N_{task}}]^\top$ where each k_i corresponds to decision boundary $\mathbf{c}_i$. Then the mean $\mu(t)$ must satisfy

$$\mathbf{C}\mu(t) = \mathbf{k} + \mathbf{b} \tag{C.8}$$

Thus, for full rank $\mathbf{C}$ and $N_{task} \geq D$, we will have a uniquely determined $\mu(t)$ value. □

Sufficient statistics and optimal estimators: "A statistic $\mu(t)$ is called sufficient for $\mathbf{x}^*$ if it contains all the information in $\mathbf{X}(1), \ldots, \mathbf{X}(t)$ about $\mathbf{x}^*$." (from Cover and Thomas' Elements of Information Theory, 1999, Section 2.10, substituting variable names).

[3] Trilateration differs from triangulation, and it is more frequently used in practice. Triangulation is when one has angle information w.r.t. the cell towers. Usually, this is not available – so one **trilaterates** their position Oguejiofor et al., 2013. This more closely matches our setting, where we just have distances information w.r.t. the decision boundaries and must determine the position.

More formally, "A function $T(\mathbf{X}(1), \ldots, \mathbf{X}(t))$ is said to be a sufficient statistic relative to the family [of probability density functions indexed by $\mathbf{x}^*$] $f(\mathbf{X}(1), \ldots, \mathbf{X}(t)|\mathbf{x}^*)$ if $\mathbf{X}(1), \ldots, \mathbf{X}(t)$ is independent of $\mathbf{x}^*$ given $T(\mathbf{X}(1), \ldots, \mathbf{X}(t))$, i.e.,

$$\mathbf{x}^* \rightarrow T\big(\mathbf{X}(1), \ldots, \mathbf{X}(t)\big) \rightarrow \mathbf{X}(1), \ldots, \mathbf{X}(t)$$

forms a Markov chain. This is the same as the condition for equality in the data processing inequality,

$$I(\mathbf{x}^*; \mathbf{X}(1), \ldots, \mathbf{X}(t)) = I(\mathbf{x}^*; \mu(t))$$

for all distributions on $\mathbf{x}^*$. Hence sufficient statistics preserve mutual information and conversely." (Cover and Thomas' Elements of Information Theory, 1999, Section 2.10, substituting variable names)

$\mu(t) = \text{mean}(\mathbf{X}(1), \ldots, \mathbf{X}(t))$ is a sufficient statistic for $\mathbf{x}^*$ given measurements $\mathbf{X}(i) \sim \mathbf{x}^* + \sigma\mathcal{N}(0, \mathbf{I}_D)$: For Gaussian noise, it is a well known result that the sufficient statistic for the underlying mean given i.i.d. samples is the sample mean of the observations (Cover and Thomas, Elements of Information Theory, 1999, Section 2.10).

Theorem C.6.5 (Optimal Representation Theorem, Linear Case)**.** *Any system that optimally estimates classification probabilities* $\hat{\mathbf{Y}}(t)$ *based on noisy measurements* $\{\mathbf{X}(1), \ldots, \mathbf{X}(t)\}$ *must implicitly encode a representation of* $\mu(t) = mean(\mathbf{X}(1), \ldots, \mathbf{X}(t))$ *in its latent state* $\mathbf{Z}(t)$ *if decision boundary matrix* $\mathbf{C}$ *is full rank and* $N_{task} \geq D$.

Proof. We showed in Theorem B.4 (Trilateration Theorem) that if $\mathbf{C} \in \mathbb{R}^{N_{task} \times D}$ is full-rank and $N_{task} \geq D$, then $\hat{\mathbf{Y}}(t)$, t, $\mathbf{b}$, and $\mathbf{C}$ are sufficient to reconstruct the exact value of $\mu(t)$, the mean of $\mathbf{X}(1), \ldots, \mathbf{X}(t)$. Rearranging Equation 16 and applying Equation 15,

$$\mu(t) = (\mathbf{C}^\top \mathbf{C})^{-1} \mathbf{C}^\top \left(\frac{\sigma}{\sqrt{t}} \Phi^{-1}(\hat{\mathbf{Y}}(t)) + \mathbf{b}\right)$$

Replacing $\hat{\mathbf{Y}}(t) = g(\mathbf{Z}(t))$ from our problem setup reveals that $\mu(t)$ is a deterministic function of $\mathbf{Z}(t)$.

Therefore, optimal multi-task classifier latent state $\mathbf{Z}(t)$ contains a sufficient statistic $\mu(t)$ of $\mathbf{x}^*$, which implies that $\mathbf{Z}(t)$ must also contain all information about $\mathbf{x}^*$ given noisy measurements $\mathbf{X}(1), \ldots, \mathbf{X}(t)$ if $\mathbf{C} \in \mathbb{R}^{N_{task} \times D}$ is full-rank and $N_{task} \geq D$. □

Theorem C.6.5 boils down to the observation that the confidence associated with each $\hat{Y}_i$ in $\hat{\mathbf{Y}}(t)$ are measures of distance between an implied estimate of $\mathbf{x}^*$ (denoted $\mu(t)$) and classification boundary i (denoted $(\mathbf{c}_i, b)$). $\hat{\mathbf{Y}}$ specifies the position of $\hat{\mathbf{X}} = \mu$ via "coordinates" defined by decision boundary normal vectors $\mathbf{c}_1, \ldots, \mathbf{c}_{N_{task}}$.

For sub-optimal estimators of $\hat{\mathbf{Y}}$, we may still obtain an understanding of the implied estimate $\hat{\mathbf{X}}$ using the same methods. In fact, the machinery of least-squares estimation for $\mathbf{Ax} = \mathbf{b}$ provides a readily accessible formula for $\tilde{\mu}$ in sub-optimal estimators of $\hat{\mathbf{Y}}$ (Equation C.8) in the form of the Moore-Penrose pseudoinverse:

$$\tilde{\mu} = (\mathbf{C}^\top \mathbf{C})^{-1} \mathbf{C}^\top (\mathbf{k} + \mathbf{b}) \tag{C.9}$$

Conveniently, if the estimation errors in sub-optimal $\hat{\mathbf{Y}}$ have a mean of zero, additional decision boundaries in $\mathbf{C}$ (e.g., beyond the minimum D linearly independent boundaries) result in improved estimation of $\mathbf{x}^*$ by the central limit theorem, thus generalizing our results to sub-optimal estimators (see Corollary C.6.7).

Optimal Representation Theorem (General Case)

We extend the results from the linear case (Theorem C.6.5) to the general case where observations are transformed by an injective observation map f in Theorem C.6.6.

Theorem C.6.6 (Optimal Representation Theorem)**.** *Let $\mathbf{x}^* \in \mathbb{R}^D$ be a latent representation for linear binary classification task $\mathbf{y}(\mathbf{x}^*) \in \{0,1\}^{N_{task}}$ and $\mathbf{X}(t) = f(\mathbf{x}^* + \sigma\mathcal{N}(\mathbf{0}, \mathbf{I}_D))$ be noisy observations transformed by an injective observation map f.*

If $\mathbf{C} \in N_{task} \times D$ is a full-rank matrix representing the decision boundary normal vectors in $\mathbb{R}^D$ and $N_{task} \geq D$, ***then any optimal estimator of*** *$\mathbf{y}(\mathbf{x}^*)$* ***must encode an optimal estimator*** *$\mu(t)$* ***of the latent variable*** *$\mathbf{x}^*$* ***in its latent state*** *$\mathbf{Z}(t)$****.*** *Furthermore, $\mu(t)$ is a sufficient statistic of $\mathbf{x}^*$, ensuring that all the information about $\mathbf{x}^*$ contained in $\{\mathbf{X}(t)\}$ is also contained in $\mathbf{Z}(t)$. Consequently, $\mu(t)$ – the optimal estimate of $\mathbf{x}^*$ based on $f(\mathbf{X}(1)), \ldots, f(\mathbf{X}(t))$ – can be written as a deterministic function (Equation C.10) of latent state $\mathbf{Z}(t)$.*

$$\mu(t) = (\mathbf{C}^\top \mathbf{C})^{-1} \mathbf{C}^\top \left(\frac{\sigma}{\sqrt{t}} \Phi^{-1}\big(g(\mathbf{Z}(t))\big) + \mathbf{b} \right) \tag{C.10}$$

Proof. We use proof by contradiction to extend the linear case of the general representation theorem to account for injective observation maps f that map $\mathbf{X}(t)$ before they are input to $\mathbf{Z}(t)$. Assume toward a contradiction that there exists a superior way of computing $\hat{Y}$ based on injectively mapped $f(\mathbf{X}(t))$ other than learning f^{-1} and following the same procedure as when $\mathbf{X}(t)$ was fed in directly (which we derived the optimal estimator for in Lemma C.6.1 and Lemma C.6.2). This assumption implies there is some additional information in $f(\mathbf{X}(t))$ that is not in $\mathbf{X}(t)$, violating the data processing inequality.

Formally, consider the following Markov chain:

$$\mathbf{x}^* \to \{\mathbf{X}(1), \ldots, \mathbf{X}(t)\} \xrightarrow{f} \mathbf{Z}(t) \to \hat{\mathbf{Y}}(t) \to \mu(t). \tag{C.11}$$

Since f is injective, f^{-1} exists, making $f(\mathbf{X}(t)) \to \mathbf{X}(t)$ an equivalent transformation in terms of information content. Hence, any optimal estimator that processes $f(\mathbf{X}(t))$ can only perform as well as if it had directly processed $\mathbf{X}(t)$.

To complete the proof, we show that $\mu(t)$ can be reconstructed from $\mathbf{Z}(t)$. Given the full-rank matrix $\mathbf{C}$, we can use the same trilateration process as in the linear case. The optimal estimate $\mu(t)$ can be written as:

$$\mu(t) = (\mathbf{C}^\top \mathbf{C})^{-1} \mathbf{C}^\top \left(\tfrac{\sigma}{\sqrt{t}} \Phi^{-1}\big(g(\mathbf{Z}(t))\big) + \mathbf{b} \right), \tag{C.12}$$

where $g(\mathbf{Z}(t))$ represents the transformation from the latent state to the classification probabilities.

Since $\mu(t)$ is the optimal estimator (and sufficient statistic) for $\mathbf{x}^*$ given measurements $\{\mathbf{X}(1), \ldots, \mathbf{X}(t)\}$, it contains all information about $\mathbf{x}^*$ contained in the measurements (Cover and Thomas, 1991). In other words, $\mu(t)$ is a deterministic function of $\mathbf{Z}(t)$, implying that $\mathbf{Z}(t)$ will contain all information about $\mathbf{x}^*$ contained in the measurements $\{\mathbf{X}(t)\}$. □

Corollary C.6.7 (Recovery of $\mu(t)$ for Sub-Optimal Classifiers)**.** *Let $\hat{\mathbf{Y}}(t) \in [0,1]^{N_{task}}$ represent the output of a sub-optimal classifier with zero-mean independent errors, i.e., $\hat{Y}_i(t) = \Pr\{y_i(\mathbf{x}^*) = 1\} + \epsilon_i$, where $\mathbb{E}[\epsilon_i] = 0$ and $Var[\epsilon_i] = \sigma_\epsilon^2$ for all $i \in \{1, \ldots, N_{task}\}$.*

If $\mathbf{C} \in \mathbb{R}^{N_{task} \times D}$ is a full-rank and well-conditioned matrix of decision boundary normal vectors with $N_{task} \geq D$, the estimated mean $\tilde{\mu}(t)$ of $\mathbf{x}^*$ can be recovered using the Moore-Penrose pseudoinverse:

$$\tilde{\mu}(t) = (\mathbf{C}^\top \mathbf{C})^{-1} \mathbf{C}^\top (\mathbf{k} + \mathbf{b}),$$

where $\mathbf{k} = \frac{\sigma}{\sqrt{t}}\Phi^{-1}(\hat{\mathbf{Y}}(t))$ *and* Φ^{-1} *is the inverse CDF of the standard normal distribution.*

For sub-optimal classifiers, as the number of tasks N_{task} *increases:*

- *The redundancy in* $\mathbf{C}$ *reduces sensitivity to classification errors.*
- *Under the assumption of independent, zero-mean errors in* $\hat{\mathbf{Y}}(t)$*, the residual error in* $\tilde{\mu}(t)$ *is expected to decrease at a rate of approximately* $O(1/\sqrt{N_{task}})$*, driven by the averaging effect of least-squares estimation.*

Motivated by the similarity between $\Phi(z)$ and sigmoid-like activation functions $g(z)$, we show that the two can be approximately canceled in Equation C.10, implying that $\mu(t)$ can be reconstructed with high accuracy with a linear-affine transformation (e.g., linear decoding) when $g = \tanh$ or $g = \text{sigmoid}$. This implies that $\mathbf{Z}(t)$ contains an abstract representation of $\mu(t)$ (Ostojic and Fusi, 2024).

Corollary C.6.8. *If the readout function* g *is* tanh*, then the reconstruction equation for* $\mu(t)$ *from* $\mathbf{Z}(t)$ *can be simplified using the approximation* $\Phi(z) \approx \frac{1}{2}\tanh(\frac{\pi}{2\sqrt{3}}z) + \frac{1}{2}$*. Consequently,* $\mu(t)$ *can be expressed directly in terms of* $\mathbf{Z}(t)$ *without the need for the inverse CDF.*

$$\mu(t) \approx \frac{2\sqrt{3}\sigma}{\pi\sqrt{t}}(\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\mathbf{Z}(t) + (\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\mathbf{b}. \tag{C.13}$$

Proof. Consider the readout function g given by $g(\mathbf{Z}(t)) = \hat{\mathbf{Y}}(t) = \frac{1}{2}\tanh(\mathbf{Z}(t)) + \frac{1}{2}$. To show that this function allows for linear decoding of $\mathbf{x}^*$ from $\mathbf{Z}(t)$, we need to leverage the similarity between tanh and Φ.

The normal distribution CDF $\Phi(z)$ and the function $\frac{1}{2}\tanh(z) + \frac{1}{2}$ are known to be very similar, as both functions are sigmoid-like, centered at zero, and asymptotically approach 0 and 1 (Choudhury, 2014).

Page (1977) proposed a simple approximation of Φ via tanh. Eliminating higher order terms, their approximation is $\Phi(x) \approx \frac{1}{2}\tanh(\sqrt{\frac{2}{\pi}}x) + \frac{1}{2}$. We found the following approximation yielded a superior mean squared error:

$$\Phi(z) \approx \frac{1}{2}\tanh\left(\frac{\pi}{2\sqrt{3}}z\right) + \frac{1}{2}.$$

Using this approximation, we can express Φ^{-1} in terms of tanh:

$$\Phi^{-1}\left(\frac{1}{2}\tanh(z)+\frac{1}{2}\right) \approx \frac{2\sqrt{3}}{\pi}z.$$

Substituting this approximation into the reconstruction equation for $\mu(t)$ from Theorem C.6.6:

$$\begin{aligned}\mu(t) &= (\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\left(\tfrac{\sigma}{\sqrt{t}}\Phi^{-1}\left(\tfrac{1}{2}\tanh(\mathbf{Z}(t))+\tfrac{1}{2}\right)+\mathbf{b}\right)\\ &\approx (\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\left(\tfrac{\sigma}{\sqrt{t}}\left(\tfrac{2\sqrt{3}}{\pi}\mathbf{Z}(t)\right)+\mathbf{b}\right)\\ &= \frac{2\sqrt{3}\sigma}{\pi\sqrt{t}}(\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\mathbf{Z}(t)+(\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\mathbf{b}.\end{aligned}$$

Therefore, we have shown that $\mu(t)$ can be expressed as a linear transformation of $\mathbf{Z}(t)$ when the readout function g is sigmoid-like. This confirms the corollary.

□

Corollary C.6.9. *For g = sigmoid, linear scaling by $a_\sigma = 0.5886$ yields a mean absolute error of 0.0038699 from $Z(t)$ in the range $[-10, 10]$, enabling the following accurate linear-affine approximation of $\mu(t)$ from $\mathbf{Z}(t)$ given g = sigmoid:*

$$\mu(t) \approx \frac{a_\sigma\,\sigma}{\sqrt{t}}(\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\mathbf{Z}(t)+(\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\mathbf{b}. \tag{C.14}$$

Proof. We found $a_\sigma = 0.5886$ by computationally minimizing the mean squared error between $\Phi(a_\sigma z)$ and $\sigma(z)$ (see `sigmoid_approx_gaussianCDF.m` in supporting code). Upon computing a_σ on successively larger optimization bounds $T = 1, 10, 100, ...$, we found that a_σ converged to 0.5886. We note that sigmoid approximations to the Gaussian distribution CDF have existed in the literature for some time (Waissi and Rossin, 1996). □

Corollary C.6.8 and C.6.9 are visualized in Figure C.8, showing the close approximations to the Gaussian CDF.

Corollary C.6.10. *$N_{task} \gg D$ implies orthogonal representations in latent $Z(t)$.*

Proof. Recall eq. (3.3) from the disentangled representation theorem:

$$\mu(t) = (\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\left(\tfrac{\sigma}{\sqrt{t}}\,\Phi^{-1}\big(g(\mathbf{Z}(t))\big)+\mathbf{b}\right).$$

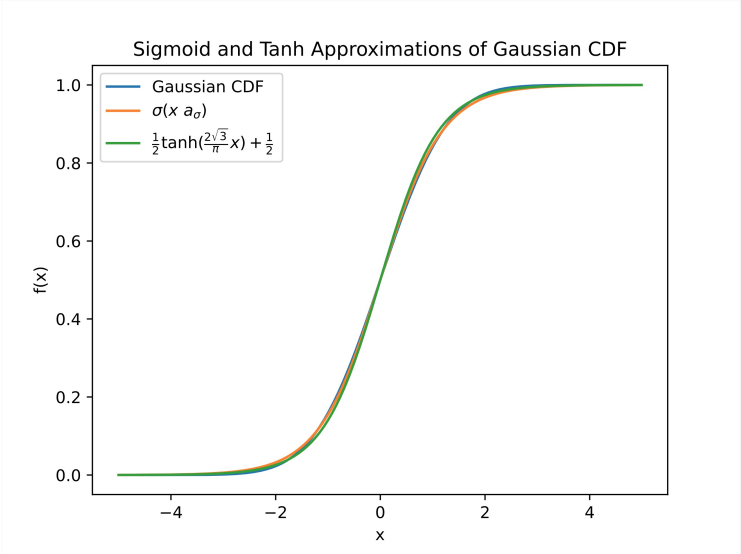

Figure C.8: **Sigmoid ($\sigma(\cdot)$) and** tanh **approximations of the normal distribution CDF Φ** via horizontal scaling.

For sigmoid-like g, we can approximate

$$\mu(t) \approx \frac{a_g\,\sigma}{\sqrt{t}}(\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\mathbf{Z}(t) + (\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\mathbf{b}.$$

The orthogonality of the representations in $\mathbf{Z}(t)$ is therefore governed by the orthogonality of the rows in the matrix $\mathbf{A} := (\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top \in \mathbb{R}^{D\times N_{task}}$. If $\mathbf{A}\mathbf{A}^\top \in \mathbb{R}^{D\times D}$ is diagonal, then the rows of $\mathbf{A}$ are orthogonal.

$$\begin{aligned}\mathbf{A}\mathbf{A}^\top &= \big((\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\big)\big((\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\big)^\top \\ &= \big((\mathbf{C}^\top\mathbf{C})^{-1}\mathbf{C}^\top\big)\big(\mathbf{C}((\mathbf{C}^\top\mathbf{C})^{-1})^\top\big) \\ &= (\mathbf{C}^\top\mathbf{C})^{-1}(\mathbf{C}^\top\mathbf{C})((\mathbf{C}^\top\mathbf{C})^{-1})^\top \\ &= ((\mathbf{C}^\top\mathbf{C})^{-1})^\top\end{aligned}$$

$B^\top B$ is a symmetric matrix for any matrix B. Recall that the inverse of a symmetric matrix is also symmetric. So $(\mathbf{C}^\top\mathbf{C})^{-1}$ is also symmetric. Therefore

$$\mathbf{A}\mathbf{A}^\top = (\mathbf{C}^\top\mathbf{C})^{-1}.$$

As the columns of $\mathbf{C}$ are high-dimensional randomly sampled vectors, their probability of being non-orthogonal vanishes as the dimensionality N_{task} increases. We can also state the condition in terms of the singular value decomposition (SVD) of

$\mathbf{C} = \mathbf{U}\Sigma\mathbf{V}^T$ where $\mathbf{U} \in \mathbb{R}^{N_{task} \times D}$, $\Sigma = \text{diag}(\sigma_1, ..., \sigma_D)$, $\mathbf{V} \in \mathbb{R}^{D \times D}$, and σ_i is the ith singular value of $\mathbf{C}$ and $\mathbf{U}, \mathbf{V}$ are orthonormal. Then

$$\mathbf{A}\mathbf{A}^\top = \mathbf{V}\Sigma^{-2}\mathbf{V}^T.$$

If the singular values are approximately uniform $\sigma_1 \dots \sigma_D \approx \sigma$ then

$$\mathbf{A}\mathbf{A}^\top \approx \mathbf{V}(\frac{1}{\sigma^2}\mathbf{I}_D)\mathbf{V}^T$$

$$\mathbf{A}\mathbf{A}^\top \approx \frac{1}{\sigma^2}\mathbf{V}\mathbf{V}^T = \frac{1}{\sigma^2}\mathbf{I}_D.$$

Therefore, uniform singular values in $\mathbf{C}$ is a sufficient condition to guarantee an orthogonal, disentangled representation of $\mu(t)$ in $\mathbf{Z}(t)$[4]. □

As a sidenote, we would like to point out that the setting here relates to the Marchenko-Pastur law while noting important caveats: while the law typically applies to matrices with i.i.d. entries $N(0, \sigma)$, our $\mathbf{C}$ matrix consists of random row vectors in $\mathbb{R}^D$ of unit norm. This structure, while not strictly meeting the law's conditions, still supports our conclusions about orthogonalization.

Suitable Noise Distributions

While the original proof leverages Gaussian noise due to its mathematical convenience, the key property required for the proof is more general. Specifically, the essential requirement is that the **marginal posterior distributions along the decision boundary normals $\mathbf{c}_i$ have invertible cumulative distribution functions (CDFs)**, allowing us to recover the distances from the observed classification probabilities.

We will now provide a precise mathematical description of the class of noise distributions where this key property holds, generalizing the disentangled representation theorem beyond Gaussian noise.

Key Noise Property Required for Proof: Invertibility of the Marginal Posterior CDFs Along Decision Boundary Normals. For each decision boundary normal vector $\mathbf{c}_i$, the marginal posterior distribution of $\mathbf{x}^*$ projected onto $\mathbf{c}_i$ must have an invertible CDF. This allows us to map the observed classification probabilities to unique distances between the estimated mean $\mu(t)$ and the decision boundaries.

[4]This uniformity condition in the singular value decomposition is analogous to the outcome of the LM damping technique (Levenberg, 1944; Marquardt, 1963), used for least squares inversion problems in various applications.

Mathematical Description of Suitable Noise Distributions: Let us define a class of noise distributions $\epsilon(t)$ where the key property holds and is straight forward to solve analytically.

Definition: The proof is immediately generalizable to noise distribution $\epsilon(t)$ if it satisfies the following conditions:

1. **Additive noise model**:
$$\mathbf{X}(t) = \mathbf{x}^* + \epsilon(t)$$
where $\epsilon(t)$ are i.i.d. random vectors.

2. **Posterior Distribution Tractability**: The posterior distribution $P(\mathbf{x}^*|\{\mathbf{X}(s)\}_{s=1}^t$ must be analytically tractable or well-approximated, allowing us to compute the posterior mean or maximum a posteriori estimate $\mu(t)$ of $\mathbf{x}^*$ and understand its properties.

3. **Existence of Invertible Marginal Posterior CDFs**: For each decision boundary normal vector $\mathbf{c}_i$, the marginal posterior distribution of $\mathbf{c}_i^\top \mathbf{x}^*$ has a continuous and strictly increasing CDF $F_i(k)$, which is invertible.

4. **Support over $\mathcal{X}$**: Let $\mathcal{X} \subseteq \mathbb{R}^D$ be the connected subset of allowable $\mathbf{x}^*$ values. The noise distribution must have full support over $\mathcal{X}$, ensuring that any real-valued $\mathbf{x}^*$ is possible to trilaterate. For our proof, we assume support over the maximally permissible $\mathbb{R}^D$ is used.

Implications: Any suitable noise distribution allows classification task probability $\hat{Y}_i(t)$ to be expressed as

$$\hat{Y}_i(t) = \Pr\{\mathbf{c}_i^\top \mathbf{x}^* > b_i | [\mathbf{X}(s)]_{s=1}^T\}$$

$$= 1 - F_i(b_i - \mathbf{c}_i^\top \mu(t))$$

where F_i is the marginal distribution of $\mathbf{c}_i^\top \mathbf{x}^*$.

Since F_i is invertible, we can solve for distance $k_i = \mathbf{c}_i^\top \mu(t) - b_i$ as

$$k_i = F_i^{-1}(1 - \hat{Y}_i(t)).$$

This equation allows us to reconstruct decision boundary distances k_i from optimal classification probabilities $\hat{Y}_i(t)$. Thus the proof via trilateration for the disentangled

representation theorem is feasible for any suitable noise distribution $\epsilon(t)$ as described above.

Examples of Suitable Noise Distributions

Elliptical Distributions

Definition: A multivariate distribution (Fang, 2018) is elliptical if its density function $f(\mathbf{x})$ can be expressed as:

$$f(\mathbf{x}) = |\mathbf{\Sigma}|^{-1/2} g\left((\mathbf{x}-\boldsymbol{\mu})^\top \mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

where $g : [0,\infty) \to [0,\infty)$ is a non-negative function, $\mu \in \mathbb{R}^D$ is the location parameter, and $\Sigma \in \mathbb{R}^{D\times D}$ is the scale matrix.

Properties:

- Symmetric and unimodal around $\boldsymbol{\mu}$.
- Projections onto any direction $\mathbf{c}_i$ yield univariate elliptical distributions.
- Marginal distributions along $\mathbf{c}_i$ have invertible CDFs if g leads to such marginals.

Examples of Suitable Elliptical Distributions:

- **Multi-variate Gaussians with Full-Rank Covariance Matrix** (Lemma C.6.11).
- **Multi-variate T-distributions with Full-Rank Scale Matrix**: Heavy-tailed alternative to the Gaussian.
- **Multivariate Laplace Distribution With Full-Rank Scale Matrix**: Has exponential tails.

Exponential Power Distributions

Definition: Also known as the generalized Gaussian distribution, defined by the density:

$$f(\mathbf{x}) = \frac{\beta}{2\alpha\Gamma(1/\beta)} \exp\left(-\left(\frac{|\mathbf{x}-\boldsymbol{\mu}|}{\alpha}\right)^\beta\right),$$

where $\beta > 0$ controls the kurtosis (Box and Tiao, 2011). Properties:

- For $\beta = 2$, it reduces to the Gaussian distribution.
- For $\beta = 1$, it becomes the Laplace distribution.
- Symmetric and unimodal.

- Marginal distributions are of the same form and have invertible CDFs.

Generalizing the Proof

Given the above, the proof can be generalized to any noise distribution $\epsilon(t)$ satisfying the conditions stated. The key steps are as follows:

1. Compute the Posterior Distribution:
   - Since $\epsilon(t)$ is i.i.d., the likelihood function is:
   $$P(\{\mathbf{X}(s)\}_{s=1}^{t} \mid \mathbf{x}^*) = \prod_{s=1}^{t} f_\epsilon(\mathbf{X}(s) - \mathbf{x}^*).$$
   - Without a prior (uniform prior), the posterior is proportional to the likelihood.
   - The posterior distribution $P(\mathbf{x}^* \mid \{\mathbf{X}(s)\}_{s=1}^{t})$ can be found (or approximated) based on the noise distribution.
2. Marginalize Along Decision Boundary Normals:
   - For each $\mathbf{c}_i$, compute the marginal posterior distribution of $\mathbf{c}_i^\top \mathbf{x}^*$.
   - Due to the symmetry and unimodality of the noise distribution, this marginal will also be symmetric and unimodal.
3. Compute Classification Probabilities: The classification probability is:
   $$\hat{Y}_i(t) = \Pr\{\mathbf{c}_i^\top \mathbf{x}^* > b_i \mid \{\mathbf{X}(s)\}_{s=1}^{t}\} = 1 - F_i(b_i - \mathbf{c}_i^\top \mu(t)),$$
   where F_i is the marginal posterior CDF of $\mathbf{c}_i^\top \mathbf{x}^*$.
4. Invert Marginal CDFs to Find Distances: Since F_i is invertible, we can solve for k_i:
   $$k_i = F_i^{-1}(1 - \hat{Y}_i(t)).$$
5. Set Up Linear System to Recover $\mu(t)$:
   - The distances k_i relate to $\mu(t)$ via:
   $$\mathbf{c}_i^\top \mu(t) = k_i + b_i.$$
   - Collecting all N_{task} equations:
   $$\mathbf{C}\mu(t) = \mathbf{k} + \mathbf{b}.$$

6. Solve for $\mu(t)$: If $\mathbf{C}$ is full rank, we can solve for $\mu(t)$:

$$\mu(t) = (\mathbf{C}^\top \mathbf{C})^{-1} \mathbf{C}^\top (\mathbf{k} + \mathbf{b}).$$

Implications

- Optimal Estimator Must Encode $\mu(t)$.
- The latent state $\mathbf{Z}(t)$ must contain sufficient information to recover $\mu(t)$, as it is essential for optimal classification across all tasks.
- The theorem holds for any noise distribution satisfying the stated conditions, not just Gaussian noise.

Example with Multivariate t-Distribution

Suppose $\epsilon(t)$ follows a multivariate t-distribution (Kotz and Nadarajah, 2004) with degrees of freedom $\nu > 2$:

1. Posterior Distribution: The posterior $P(\mathbf{x}^* \mid \{\mathbf{X}(s)\}_{s=1}^t)$ is also a multivariate t-distribution.
2. Marginal Posterior Distributions: Projections onto $\mathbf{c}_i$ yield univariate t-distributions.
3. Invertible Marginal CDFs: The CDF of the t-distribution is known and invertible.
4. Recover Distances: Use the inverse t-CDF to find k_i:

$$k_i = s_t \cdot T_\nu^{-1}(1 - \hat{Y}_i(t)),$$

   where s_t is the scale parameter, and T_ν^{-1} is the inverse CDF of the t-distribution with ν degrees of freedom.
5. Proceed with the proof: Follow the same steps as before to reconstruct $\mu(t)$.

Example with Anisotropic Gaussian Noise

Lemma C.6.11. *Suppose $\epsilon(t)$ follows an anisotropic multi-variate Gaussian distribution with full-rank covariance matrix Σ and zero mean. Then we can update Equation C.6 from Lemma C.6.2 as*

$$\hat{Y}(t) \triangleq \Pr(\mathbf{c}^\top \mathbf{x}^* > b) \tag{C.15}$$

$$= \Phi\left(\frac{k\sqrt{t}}{\sqrt{\mathbf{c}^\top \Sigma \mathbf{c}}}\right). \tag{C.16}$$

Proof. Anisotropic noise results in the quadratic form $\mathbf{c}^\top \Sigma \mathbf{c}$ in the denominator representing the variance of the marginalized anisotropic noise distribution along decision boundary normal vector $\mathbf{c}_i$. As long as Σ is non-singular, the remainder of the disentangled representation proof may proceed substituting Equation C.6 with Equation C.16. □

Conclusion

The key property enabling us to recover distances from classification probabilities is the invertibility of the marginal posterior CDFs along the decision boundary normals. This property is not exclusive to Gaussian noise but is shared by a broader class of noise distributions, including but not limited to:

- Elliptical distributions (e.g., Laplace, multivariate t-distributions).
- Exponential power distributions.
- Other symmetric and unimodal distributions with invertible marginals.

Therefore, the proof of the Disentangled Representation Theorem generalizes to any noise distribution satisfying the conditions outlined above. The essential requirement is that we can uniquely map the observed classification probabilities to distances along the normals, allowing us to reconstruct the posterior mean $\mu(t)$ and establish that any optimal estimator must encode this information in its latent state $\mathbf{Z}(t)$.

Correspondence in the structure of noise distribution CDF along marginals F_i and point-wise activation functions g (the activation function $\hat{\mathbf{Y}}(t) = g(\mathbf{Z}(t))$).

Discussion

The theoretical results presented in this appendix, particularly the Optimal Representation Theorem (Theorem C.6.6), provide insights into the factors driving representational convergence and alignment in neural networks and, more generally, any optimal multi-task classifier in the setup shown in Figure C.6. This theorem establishes a clear connection between the latent representations learned by optimal multi-task classifiers and the true underlying data representation, offering a principled explanation for the emergence of disentangled representations aligned with the intrinsic structure of the data.

Connection to Manifold Hypothesis: Our theoretical results have important implications for the manifold hypothesis, which posits that real-world high-dimensional

data tend to lie on or near low-dimensional manifolds embedded in the high-dimensional space (Fefferman, Mitter, and Narayanan, 2016; Olah, 2014). The key insight is that our proofs show an optimal multi-task classifier must encode an estimate of the disentangled coordinates of the true underlying environment state in its latent representation. Consider the disentangled space in which $\mathbf{x}^*$ resides, denoted $\mathcal{X}^*$. The injective observation map $f : \mathcal{X}^* \to \mathcal{X}$, where decision boundaries $y_i : \mathcal{X}^* \to \{0, 1\}$ are linear. Our results imply that an optimal classifier's latent state $\mathbf{Z}(t)$ must encode disentangled coordinates in $\mathcal{X}^*$ rather than ambient coordinates $\mathcal{X}$.

The injective observation map f aligns closely to the typical conception of a data manifold (e.g., if $f \in C^1$ or $f \in C^n$, as described in Tu, 2017). The disentangled space $\mathcal{X}^*$ can be seen as the intrinsic coordinate system of the manifold, while f maps these coordinates to the high-dimensional observation space $\mathcal{X}$. Our findings suggest that an optimal classifier will implicitly learn to invert this mapping to recover the disentangled coordinates. Moreover, for natural data where the manifold hypothesis holds, the learned latent representation would plausibly capture the manifold structure, as this is essential for disambiguating noisy observations and estimating the true underlying state. The low-dimensional manifold structure is a key prior that an optimal classifier can (and in our case must) exploit to improve its performance.

Relation to Autoregressive Language & Multi-Modal Transformers: Consider an analogy with masked autoencoder vision foundation models, where $\mathbf{x}^*$ is the "ground truth" of a scene (objects, positions, states, and relationships), the measurement variable $\mathbf{X}$ is an image with missing patches (Dosovitskiy et al., 2020; He et al., 2021), and the model predicts the missing patch data $y(\mathbf{x}^*)$. The model's latent variable $\mathbf{Z}$ exhibits some "understanding" of $\mathbf{x}^*$ in the form of abstract representations useful for downstream tasks. This analogy extends to masked language models (Devlin et al., 2018) and autoregressive language models (Radford et al., 2019), where $\mathbf{x}^*$ is "meaning" in a semantic space, $\mathbf{X}(t)$ are words, and y is the next word. Localizing $\mathbf{x}^*$ from $\mathbf{Z}(t)$ relates to constructing a world model, showing that $\mathbf{Z}$ represents $\mathbf{x}^*$ abstractly and with high fidelity.

Ordering of Noise and Observation Map: The ordering of the noise and the non-linear observation map matters for the latent space representation. When the noise is applied before the observation map, the noisy observations are constrained

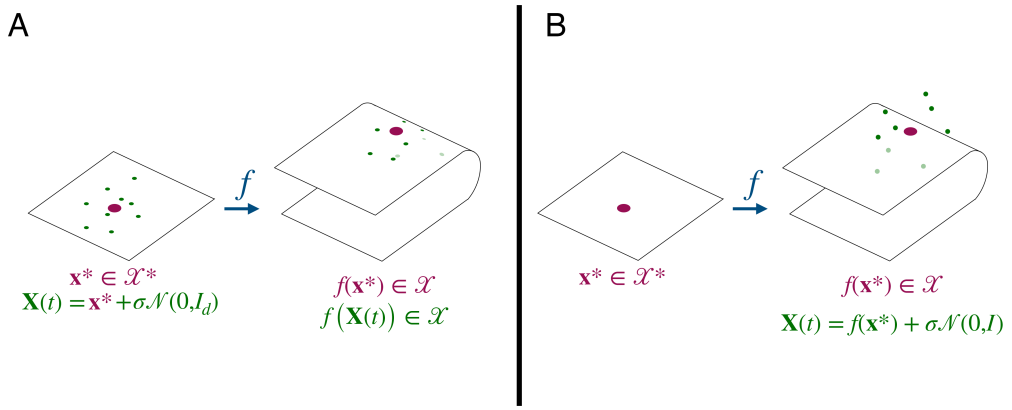

Figure C.9: **The impact of noise and non-linear transformation order.** (A) $\mathbf{x}^*$ is noised before being transformed by injective observation map f, resulting in observations $f(\mathbf{X}(t))$ lying on the image of f (here f is a 2D folded surface). (B) $\mathbf{x}^*$ is first transformed by injective observation map f and noise is added afterward, resulting in observations $f(\mathbf{x}^*) + \sigma\mathcal{N}(0, I)$ that do not lie on the image of f.

to lie on a manifold with the same intrinsic dimension as the true latent space $\mathcal{X}^*$. In contrast, when the noise is applied after the observation map, the noisy observations can deviate from the low-dimensional manifold, potentially introducing degeneracy where two noised observations arising from different $\mathbf{x}^*$ may appear identical (i.e., non-injective). Imagine $\mathcal{X}^*$ as a 2D piece of paper. An injective, smooth, continuous observation map $f : \mathcal{X}^* \to \mathcal{X}$ where $\mathcal{X}$ is a 3-dimensional space "crumples" the sheet of paper $\mathcal{X}^*$ into a crumpled ball in $\mathcal{X}$. If you add noise after the mapping, a point on one corner of the paper could get "popped out" of the 2D manifold by the noise and end up very far away on the crumpled surface if you were to examine it flattened out (illustrated in Figure C.9).

The curvature of the observation map f and the level of noise σ are fundamental factors influencing the extent of the degeneracy introduced by the noise after the observation map. High curvature in f can make the intrinsic geometry of the data more challenging to identify (e.g., more tightly crumpled paper). Large noise levels can push observations further from the underlying manifold, similarly worsening the potential degeneracy in the observations. The reach of the manifold f (Aamari et al., 2019) can be used as an immediate loose bound for post-observation map noise $\epsilon_2(t)$ to ensure that the derived theorems still hold.

Connection to the Platonic Representation Hypothesis: Our results provide a new perspective on the Platonic representation hypothesis (Huh et al., 2024). The

Platonic representation hypothesis suggests that the convergence in deep neural network representations is driven by a shared statistical model of reality, like Plato's concept of an ideal reality. Convergence of representations is analyzed in terms of similarity of distances between embedded datapoints among AI models trained on various modalities. While the authors of the hypothesis argue that energy constraints might lead to divergence from a shared representation for specialized tasks, our Optimal Representation Theorem suggests that the key factor driving convergence is the diversity and comprehensiveness of the tasks being learned. As long as the tasks collectively span the space of the underlying data representation, convergence to a shared, reality-aligned representation can occur, even in the presence of energy or computational limitations. Our theoretical results amount to a necessary condition for optimal multi-task classifiers to represent a disentangled representation of the data within their latent state. With energy constraints, extraneous network activity may be regularized out of the model, resulting in greater alignment between disentangled representations in energy constrained models that "understand" the Platonic nature of reality. The very energy constraints Huh et al., 2024 suggest may lead to divergence could actually facilitate convergence of the platonic representations, as they may encourage the learning of simple, generalizable features that capture the essential structure of the data. This insight opens up interesting avenues for future research on the interplay between task diversity, energy constraints, and the emergence of shared representations. Finally, energy constraints have been shown to naturally lead to predictive coding (Rao and Ballard, 1999; Ali et al., 2022), tightening the relationship between energy efficiency, prediction, and cognitive map learning. A relationship between predictive coding and optimal Bayesian estimation has also been established (Rao, 1999).

Implications and Future Directions: The theoretical analysis presented in this appendix sheds light on the factors driving the emergence of disentangled representations in neural networks and their alignment with the intrinsic structure of the data. By formalizing the conditions under which learned representations recover the true underlying data manifold, our work provides a foundation for understanding the remarkable success of representation learning across diverse domains. Avenues for future research include exploring the sample complexity of learning under different observation maps and noise levels, and empirically validating the convergence of representations across models and modalities in the context of task diversity and energy constraints.

References

Aamari, Eddie et al. (2019). "Estimating the reach of a manifold." arXiv: 1705.04565 [math.ST]. URL: https://arxiv.org/abs/1705.04565.

Ali, Abdullahi et al. (Dec. 2022). "Predictive coding is a consequence of energy efficiency in recurrent neural networks." In: *Patterns* 3.12, p. 100639. ISSN: 2666-3899. DOI: 10.1016/j.patter.2022.100639. URL: http://dx.doi.org/10.1016/j.patter.2022.100639.

Bhargava, Aman et al. (2024). "Prompt baking." arXiv: 2409.13697 [cs.CL]. URL: https://arxiv.org/abs/2409.13697.

Box, George E.P. and George C. Tiao (2011). "Bayesian inference in statistical analysis." John Wiley & Sons.

Brunton, Bingni W., Matthew M. Botvinick, and Carlos D. Brody (Apr. 2013). "Rats and humans can optimally accumulate evidence for decision-making." In: *Science* 340.6128, pp. 95–98. DOI: 10.1126/science.1233912. URL: https://doi.org/10.1126/science.1233912.

Choudhury, Amit (2014). "A simple approximation to the area under standard normal curve." In: *Mathematics and Statistics* 2.3, pp. 147–149.

Cover, Thomas M. and Joy A. Thomas (1991). "Information theory and the stock market." In: *Elements of Information Theory. Wiley Inc., New York*, pp. 543–556.

Devlin, Jacob et al. (2018). "BERT: Pre-training of deep bidirectional transformers for language understanding." In: *CoRR* abs/1810.04805. arXiv: 1810.04805. URL: http://arxiv.org/abs/1810.04805.

Dosovitskiy, Alexey et al. (2020). "An image is worth 16x16 words: Transformers for image recognition at scale." DOI: 10.48550/ARXIV.2010.11929. URL: https://arxiv.org/abs/2010.11929.

Driscoll, Laura, Krishna Shenoy, and David Sussillo (2022). "Flexible multitask computation in recurrent networks utilizes shared dynamical motifs." In: *bioRxiv*, pp. 2022–08.

Fang, Kai Wang (2018). "Symmetric multivariate and related distributions." Chapman and Hall/CRC.

Fefferman, Charles, Sanjoy Mitter, and Hariharan Narayanan (2016). "Testing the manifold hypothesis." In: *Journal of the American Mathematical Society* 29.4, pp. 983–1049.

Flesch, Timo et al. (2022). "Modelling continual learning in humans with Hebbian context gating and exponentially decaying task signals." DOI: 10.48550/ARXIV.2203.11560. URL: https://arxiv.org/abs/2203.11560.

He, Kaiming et al. (2021). "Masked autoencoders are scalable vision learners." arXiv: 2111.06377 [cs.CV].

Huh, Minyoung et al. (2024). "The platonic representation hypothesis." In: *arXiv preprint arXiv:2405.07987*.

Johnston, W. Jeffrey and Stefano Fusi (Feb. 2023). "Abstract representations emerge naturally in neural networks trained to perform multiple tasks." In: *Nature Communications* 14.1. DOI: 10.1038/s41467-023-36583-0. URL: https://doi.org/10.1038/s41467-023-36583-0.

Kotz, Samuel and Saralees Nadarajah (2004). "Multivariate t-distributions and their applications." Cambridge University Press.

Krajbich, Ian, Carrie Armel, and Antonio Rangel (2010). "Visual fixations and the computation and comparison of value in simple choice." In: *Nature Neuroscience* 13.10, pp. 1292–1298. DOI: 10.1038/nn.2635. URL: https://doi.org/10.1038/nn.2635.

Levenberg, Kenneth (1944). "A method for the solution of certain non-linear problems in least squares." In: *Quarterly of Applied Mathematics* 2.2, pp. 164–168. ISSN: 1552-4485. DOI: 10.1090/qam/10666. URL: http://dx.doi.org/10.1090/qam/10666.

Ma, Yi, Doris Tsao, and Heung-Yeung Shum (2022). "On the principles of parsimony and self-consistency for the emergence of intelligence." DOI: 10.48550/ARXIV.2207.04630. URL: https://arxiv.org/abs/2207.04630.

Mante, Valerio et al. (Nov. 2013). "Context-dependent computation by recurrent dynamics in prefrontal cortex." In: *Nature* 503.7474, pp. 78–84. DOI: 10.1038/nature12742. URL: https://doi.org/10.1038/nature12742.

Marquardt, Donald W. (June 1963). "An algorithm for least-squares estimation of nonlinear parameters." In: *Journal of the Society for Industrial and Applied Mathematics* 11.2, pp. 431–441. ISSN: 2168-3484. DOI: 10.1137/0111030. URL: http://dx.doi.org/10.1137/0111030.

Masset, Paul et al. (July 2020). "Behavior- and modality-general representation of confidence in orbitofrontal cortex." In: *Cell* 182.1, 112–126.e18. DOI: 10.1016/j.cell.2020.05.022. URL: https://doi.org/10.1016/j.cell.2020.05.022.

Oguejiofor, Obinna Samuel et al. (2013). "Trilateration based localization algorithm for wireless sensor network." In: *International Journal of Science and Modern Engineering (IJISME)* 1.10, pp. 2319–6386.

Olah, Chris (Apr. 2014). "Neural networks, manifolds, and topology." URL: https://colah.github.io/posts/2014-03-NN-Manifolds-Topology/.

Ostojic, Srjan and Stefano Fusi (July 2024). "Computational role of structure in neural activity and connectivity." In: *Trends in Cognitive Sciences* 28.7, pp. 677–690. ISSN: 1364-6613. DOI: 10.1016/j.tics.2024.03.003. URL: http://dx.doi.org/10.1016/j.tics.2024.03.003.

Page, E. (1977). "Approximations to the cumulative normal function and its inverse for use on a pocket calculator." In: *Journal of the Royal Statistical Society Series C: Applied Statistics* 26.1, pp. 75–76.

Radford, Alec et al. (2019). "Language models are unsupervised multitask learners." In: *OpenAI blog* 1.8, p. 9.

Rao, Rajesh P. N. (June 1999). "An optimal estimation approach to visual perception and learning." In: *Vision Research* 39.11, pp. 1963–1989. ISSN: 0042-6989. DOI: 10.1016/s0042-6989(98)00279-x. URL: http://dx.doi.org/10.1016/S0042-6989(98)00279-X.

Rao, Rajesh P. N. and Dana H. Ballard (Jan. 1999). "Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects." In: *Nature Neuroscience* 2.1, pp. 79–87. DOI: 10.1038/4580. URL: https://doi.org/10.1038/4580.

Rutishauser, Ueli et al. (Jan. 2018). "Single-neuron representation of memory strength and recognition confidence in left human posterior parietal cortex." In: *Neuron* 97.1, 209–220.e3. DOI: 10.1016/j.neuron.2017.11.029. URL: https://doi.org/10.1016/j.neuron.2017.11.029.

Soo, Wayne W. M., Vishwa Goudar, and Xiao-Jing Wang (Oct. 2023). "Training biologically plausible recurrent neural networks on cognitive tasks with long-term dependencies." DOI: 10.1101/2023.10.10.561588. URL: http://dx.doi.org/10.1101/2023.10.10.561588.

Tu, Loring W. (2017). "Differential geometry: Connections, curvature, and characteristic classes." Vol. 275. Springer.

Vinje, William E. and Jack L. Gallant (Feb. 2000). "Sparse coding and decorrelation in primary visual cortex during natural vision." In: *Science* 287.5456, pp. 1273–1276. ISSN: 1095-9203. DOI: 10.1126/science.287.5456.1273. URL: http://dx.doi.org/10.1126/science.287.5456.1273.

Waissi, Gary R. and Donald F. Rossin (1996). "A sigmoid approximation of the standard normal integral." In: *Applied Mathematics and Computation* 77.1, pp. 91–95.

Yang, Guangyu Robert and Xiao-Jing Wang (Sept. 2020). "Artificial neural networks for neuroscientists: A primer." In: *Neuron* 107.6, pp. 1048–1070. DOI: 10.1016/j.neuron.2020.09.005. URL: https://doi.org/10.1016/j.neuron.2020.09.005.

Yang, Guangyu Robert et al. (Feb. 2019). "Task representations in neural networks trained to perform many cognitive tasks." In: *Nature Neuroscience* 22.2, pp. 297–306. ISSN: 1546-1726. DOI: 10.1038/s41593-018-0310-2. URL: https://doi.org/10.1038/s41593-018-0310-2.

Appendix D

A PRIMER ON DETECTING AND QUANTIFYING CONTINUOUS ATTRACTORS (CANS)

Continuous attractors played a central role in this thesis, notably in Chapters II and III; however, their definitions often reside in somewhat ambiguous territory. This Appendix aims to clarify these ambiguities, particularly to prevent unnecessary confusion within the experimental community, leading to wasted efforts and time, and potentially reduced trust for computational methods.

D.1 Defining Attractors

Attractors are stable configurations toward which dynamical systems evolve over time. They represent equilibrium states in an energy landscape, typically characterized by an energy function $E(\mathbf{x})$, where the system state $\mathbf{x}$ evolves to minimize this energy. Formally, the dynamics of the system with state $\mathbf{x}$ are defined as:

$$\frac{d\mathbf{x}}{dt} = -\nabla E(\mathbf{x}). \tag{D.1}$$

In neural networks, Hopfield networks (Hopfield, 1982) are classic examples of discrete attractors, where stable points represent stored memories, and $\mathbf{x}$ is the population vector of the firing rates of the neurons that evolves over time. More generally, lets define an arbitrary recurrent neural network with recurrent matrix weight W connecting its neurons. Its dynamics are governed by:

$$\frac{d\mathbf{x}}{dt} = F(\mathbf{x}, W), \quad \text{where} \quad F(\mathbf{x}, W) = -\nabla E(\mathbf{x}). \tag{D.2}$$

Here, the function F describes how the state $\mathbf{x}$ evolves continuously over time, and the explicit inclusion of the recurrent weight matrix W emphasizes its central role in the dynamics. The recurrent weight matrix W governs how activity propagates and evolves through the neural population, determining the stability and dynamics of network states. Each neuron's activation x_i influences others through this connectivity, enabling complex patterns of sustained and evolving activity. Hence, W is going to be the center of our attention going forward, when we define continuous attractors.

D.2 Extension to Continuous Attractors

Continuous attractors (CANs), introduced notably by Amari (1977), extend the concept of discrete attractors to represent a continuum of stable states. Imagine a perfectly flat energy valley rather than isolated energy wells. A useful analogy is a multi-stable pool table where a cue ball can stably rest at any point along a line or plane. Continuous attractors are useful to store continuous variables, instead of discrete memories. However, continuous attractors in neuroscience typically require very precise connectivity, to avoid leakage while being at the "edge" of stability: stable, but also reactive to minimal perturbations that might update their state (i.e., the value of the stored variable). In addition, maintaining exact continuous attractors in biological neural networks is challenging due to inherent noise and

imprecisions. Hence, biological continuous attractors are often leaky, exhibiting slight drift. Furthermore, in order to have perfectly continuous attractors, an infinite number of neurons is needed, as the number of stable states scales linearly with the number of available neurons supporting these states. Therefore, every continuous attractor in neuroscience is an approximate, or *quasi-continuous* attractor. The word *approximate* has been used erroneously in recent neuroscience literature to relax the conditions a quasi-continuous attractor must meet to qualify as such, however the correct definition is the one just mentioned; more on that later. For now, the reader should keep in mind that when it comes to neuroscience, when we are talking about a continuous attractor we are effectively referring to a quasi-continuous attractor. Yet, in order to qualify as a quasi-continuous attractor, an attractor still has to satisfy some (quite hard to meet) requirements (e.g., separation of timescales, see next).

A quasi-continuous attractor can take several forms, including a set of discrete attractors that are close enough and have low energy barriers between them so that they qualify as a continuous attractor, or a set of slow points that, even though not really stable, are slow enough to act as a continuous memory with regards to the rest of the circuit (separation of timescales argument) (Seung, 1996; Khona and Fiete, 2022). Examples of quasi-continuous attractors in neuroscience include head-direction (Seelig and Jayaraman, 2015; Kim et al., 2017; Chaudhuri et al., 2019; Vafidis et al., 2022) and grid cell representations (Gardner et al., 2022; Banino et al., 2018; Sorscher et al., 2023). The former are implemented by a 1D ring attractor, while the latter by 2D toroidal attractors. An accurate estimation of location in space is crucial for animals to navigate the world in the absence of external cues (see Chapter II of this thesis), justifying the precise synaptic weights required for the implementation of continuous attractors that form the substrate of these representations.

D.3 Continuous vs. Discrete: Microstructure Matters

To determine whether a network implements a truly continuous attractor or merely a set of discrete ones, it is essential to analyze the microstructure of the attractor. Continuous attractors should possess multiple stable (or near-stable) states separated by small energy barriers, facilitating smooth transitions within the attractor manifold (Vafidis et al., 2022; Khona and Fiete, 2022). Critically, continuous attractors function as memory systems, necessitating a separation of timescales between dynamics within the attractor dimension (slow) and external circuit dynamics (fast).

This separation of timescales can be practically assessed by analyzing the eigenstructure of the recurrent connectivity matrix of the neural system, as demonstrated in Chapter III of this thesis (Vafidis, Bhargava, and Rangel, 2025). The continuousness of the attractor can also be judged by the stimulation experiments performed in vivo in Kim et al. (2017) and in silico in Vafidis et al. (2022) (Chapter II of this thesis), where continuous attractors have small energy barriers between stable states, and as a result transitions between adjacent stable states are "smooth" as opposed to "jumpy" for far away states (see fig. 2.2D).

D.4 From Local Eigenvalues to Local Time Constants

Mathematically, a neural system's discrete-time linearized dynamics around a fixed point can be approximated by:

$$\mathbf{x}_{t+1} - \mathbf{x}_t \approx W\mathbf{x}_t \tag{D.3}$$

where W represents the effective recurrent weight matrix. To analyze local stability and dynamics, we calculate eigenvalues λ_i of matrix W:

$$W\mathbf{v}_i = \lambda_i \mathbf{v}_i. \tag{D.4}$$

These eigenvalues were computed in Chapter III of this thesis. Eigenvalues near 0 indicate that the difference system $\mathbf{x}_{t+1} - \mathbf{x}_t$ changes slowly over time, i.e., they correspond to "slow" dimensions in network dynamics which can integrate inputs and maintain them over time (continuous attractors) (Amari, 1977; Mante et al., 2013). From these eigenvalues, local time constants (τ_i) describing the speed at which the system returns to equilibrium along the eigen-directions can also be derived as:

$$\tau_i = \frac{1}{|\log(|1 + \lambda_i|)|} \tag{D.5}$$

where time constants are assumed to be positive. Note that the notation here follows the difference system $\mathbf{x}_{t+1} - \mathbf{x}_t$. For the time evolution system $\mathbf{x}_{t+1} \approx W'\mathbf{x}_t$, the formula for the time constants would be $\tau_i = \frac{1}{|\log|\lambda'_i||}$, where W' and λ'_i are the effective recurrent weight matrix and corresponding eigenvalue for the time evolution system, where $W = I + W'$ and $\lambda_i = 1 + \lambda'_i$ (Maheswaranathan et al., 2019).

D.5 From Local Time Constants to Quasi-Continuous Attractors

The presence of a continuous attractor dimension manifests as one or more directions in state space with very long (large) time constants relative to other directions. Thus, a continuous attractor can be identified as a set of discrete attractors with similarly high time constants forming a continuous manifold. To quantify the ability of an (approximate) fixed point to maintain its state, we can define the time constant amplification ratio A as:

$$A = \frac{\tau_{\text{network}}}{\tau_{\text{neuronal}}}. \tag{D.6}$$

Here, the numerator, τ_{network}, represents the time constant of the network dynamics, while the denominator, τ_{neuronal}, reflects the intrinsic timescale of individual neurons (a combination of membrane and synaptic time constants). Thus, this ratio captures how much the network dynamics have been amplified beyond the intrinsic neural timescales due to the presence of the attractor.

When multiple fixed points with significantly amplified time constants exist within a continuous manifold, we can define *directional time constants* that characterize the temporal dynamics along specific dimensions of the network state space. These directional time constants emerge as compositions of individual point time constants within that manifold, typically dominated by the slowest individual attractor states.

If such a continuum of stable or quasi-stable states exists, characterized by directional time constants substantially larger than intrinsic neuronal timescales (indicating clear separation of timescales), we conclude that the network exhibits a quasi-continuous attractor along that particular direction.

Indeed, this is exactly what we observed in Chapter III, where a continuum of approximately fixed points was observed (fig. 3.3d), with eigenvalues close to 0 across two directions for each point (fig. 3.3e). This paints the picture of a 2D continuous attractor, similar to the pool table mentioned before. Furthermore, we can quantify the time constant amplification ratio A for all of these fixed points. We observe that most amplification ratios are in the range of 10-100 (fig. D.1), which satisfies the condition for separation of timescales (typically, there should be an order of magnitude or more difference between the slow and fast timescales). While ideally we would like these time constant amplification factors to be larger, we have to remember that the circuit is just composed of 64 neurons, that have to perform a lot of other operations apart from memory (invert nonlinear mapping, denoise, etc.).

Furthermore, we can obtain directional time constants by quantifying the average directional amplification ratio A_d as a result of traversing along a certain direction in this continuous attractor (dominated by higher individuals ratios A).

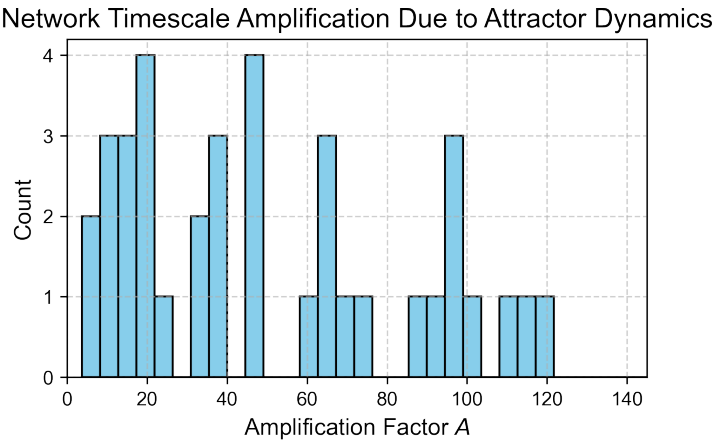

Figure D.1: **Time Constant Amplification Ratios** for individual approximate fixed points in Continuous Attractor in Chapter III (fig. 3.3d). Network time constants τ_{network} were derived from the eigenvalues in fig. 3.3e, and the time constant amplification ratio was computed as $A = \frac{\tau_{\text{network}}}{\tau}$, where τ is the neuronal time constant from table 3.1.

Finally, note that in the case of the continuous attractor in Chapter II, the fixed points are stable, therefore the amplification factor is technically infinity. However, in that case we still have to show that the energy barrier between adjacent fixed point is low, i.e., show that the continuous attractor can update itself with minimal perturbation, a requirement to maintain a memory of a continuous variable. This is exactly what we showed in fig. 2.2D. This further demonstrates the dynamic balance between stability and flexibility that a continuous attractor has to demonstrate, a feat particularly challenging to achieve, requiring exquisite synaptic weight balance.

References

Amari, Shun-ichi (1977). "Dynamics of pattern formation in lateral-inhibition type neural fields." In: *Biological Cybernetics* 27.2, pp. 77–87. DOI: 10.1007/bf00337259. URL: https://doi.org/10.1007/bf00337259.

Banino, Andrea et al. (May 2018). "Vector-based navigation using grid-like representations in artificial agents." In: *Nature* 557.7705, pp. 429–433. DOI: 10.1038/s41586-018-0102-6. URL: https://doi.org/10.1038/s41586-018-0102-6.

Chaudhuri, Rishidev et al. (Aug. 2019). "The intrinsic attractor manifold and population dynamics of a canonical cognitive circuit across waking and sleep." In: *Nature Neuroscience* 22.9, pp. 1512–1520. DOI: 10.1038/s41593-019-0460-x. URL: https://doi.org/10.1038/s41593-019-0460-x.

Gardner, Richard J. et al. (Jan. 2022). "Toroidal topology of population activity in grid cells." In: *Nature* 602.7895, pp. 123–128. DOI: 10.1038/s41586-021-04268-7. URL: https://doi.org/10.1038/s41586-021-04268-7.

Hopfield, John J. (Apr. 1982). "Neural networks and physical systems with emergent collective computational abilities." In: *Proceedings of the National Academy of Sciences* 79, pp. 2554–2558. URL: https://www.pnas.org/content/79/8/2554.

Khona, Mikail and Ila R. Fiete (Nov. 2022). "Attractor and integrator networks in the brain." In: *Nature Reviews Neuroscience* 23.12, pp. 744–766. ISSN: 1471-0048. DOI: 10.1038/s41583-022-00642-0. URL: http://dx.doi.org/10.1038/s41583-022-00642-0.

Kim, Sung Soo et al. (May 2017). "Ring attractor dynamics in the Drosophila central brain." In: *Science* 356.6340, pp. 849–853. DOI: 10.1126/science.aal4835. URL: https://doi.org/10.1126/science.aal4835.

Maheswaranathan, Niru et al. (2019). "Reverse engineering recurrent networks for sentiment classification reveals line attractor dynamics." In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2019/file/d921c3c762b1522c475ac8fc0811bb0f-Paper.pdf.

Mante, Valerio et al. (Nov. 2013). "Context-dependent computation by recurrent dynamics in prefrontal cortex." In: *Nature* 503.7474, pp. 78–84. DOI: 10.1038/nature12742. URL: https://doi.org/10.1038/nature12742.

Seelig, Johannes D. and Vivek Jayaraman (May 2015). "Neural dynamics for landmark orientation and angular path integration." In: *Nature* 521.7551, pp. 186–191. DOI: 10.1038/nature14446. URL: https://doi.org/10.1038/nature14446.

Seung, H. Sebastian (Nov. 1996). "How the brain keeps the eyes still." In: *Proceedings of the National Academy of Sciences of the United States of America* 93.23, pp. 13339–13344. DOI: 10.1073/pnas.93.23.13339. URL: https://doi.org/10.1073/pnas.93.23.13339.

Sorscher, Ben et al. (Jan. 2023). "A unified theory for the computational and mechanistic origins of grid cells." In: *Neuron* 111.1, 121–137.e13. DOI: 10.1016/j.neuron.2022.10.003. URL: https://doi.org/10.1016/j.neuron.2022.10.003.

Vafidis, Pantelis, Aman Bhargava, and Antonio Rangel (2025). "Disentangling representations through multi-task learning." In: *The Thirteenth International Conference on Learning Representations*. URL: https://openreview.net/forum?id=yVGGtsOgc7.

Vafidis, Pantelis et al. (June 2022). "Learning accurate path integration in ring attractor models of the head direction system." In: *eLife* 11. Ed. by Srdjan Ostojic, Ronald L Calabrese, and Hervé Rouault, e69841. ISSN: 2050-084X. DOI: 10.7554/eLife.69841. URL: https://doi.org/10.7554/eLife.69841.