Principles of massively parallel sequencing for engineering and characterizing gene delivery

Thesis by

David Brown

In Partial Fulfillment of the Requirements for the

Degree of

Doctor of Philosophy

CALIFORNIA INSTITUTE OF TECHNOLOGY

Pasadena, California

2022

Defended August 31, 2021

David Brown
ORCID: 0000-0002-9757-1744

ACKNOWLEDGEMENTS

I would like to thank the following people that played a critical role in my journey into science, and throughout my Ph.D.:

Tatyana Dobreva, for… everything.

My mom and my brother—our stable trio—for always being proud of me and encouraging me to do what I wanted to do, and not what I was supposed to do, and my mom for pushing me to make a better life for myself.

My oma, for being the most resilient and kind person. She has been a consistent anchor of sanity in my otherwise chaos-ridden roller coaster of a life.

My therapist, for helping me accept my value as a human being.

Viviana Gradinaru, for encouraging and supporting me to do my own experiments and work towards becoming an expert in all aspects of my work.

Matt Thomson, for never saying "No," and instead nurturing a dialogue of "How?" and "Why?"

Michael Altermatt, for being the most dedicated, principled teammate I could have asked for, and for setting a precedence for rigor I aspire to achieve.

The members of my committee, Yisong Yue, Frances Arnold, and Steve Mayo, for being great sounding boards for my forays into machine learning, directed evolution, and protein engineering.

Ben Deverman, Ken Chan, Alon Greenbaum, Elisha Mackey, Spencer Kellis, and so many others for welcoming me into the world of biology.

Dr. Jane Huggins and Dr. Richard Andersen, for giving me the opportunity to take my first steps into science.

Jeff Park, for answering my 1000s of wetlab questions over the years, and for our many sci-fi musings.

The members of Viviana Gradinaru's lab that have accompanied me on this journey and helped me build my expertise of engineering gene therapy: Sripriya Ravindra Kumar, Máté Borsos, Xinhong Chen, Xiaozhe Ding, David Goertsen, Gerry Coughlin, Min, Tim Miles, Alexander Wang, Nick Goeden, and Nick Flytzanis.

The members of Matt Thomson's lab, especially the single-cell subgroup, that have been instrumental brainstorming partners: Sisi Chen, Paul Rivaud, Tiffany Tsou, Tyler Ross, and Tami Khazaei

Sovereignty Club, for providing a space where saying "I plan to own a country one day" was a normal thing to say.

TechLit, for giving me space (and pizza) to write down my dreams and cautions about the future.

NeuroTechers, and the CNS/NB student community, for so many great conversations about the brain, AI, and everything in between (or are they the same?)

My previous bosses at BIGGBY COFFEE, Mike and Bob, for making the crazy decision to entrust their technology infrastructure to a college dropout from New Mexico, and then allowing me the flexibility to pursue my dream of going back to school and building a life that I love.

Kevin Warwick, for responding to that letter from my naïve 20-something-year-old IT admin self – you may never know how much of a role you played in encouraging my future path.

The musicians and mix artists who became the background track of my Ph.D.: Wardruna, Danheim, Delta Notch, Guillaume David, Michael McCann, Hans Zimmer, Brian Tyler, Hol Baumann, Carbon Based Lifeforms, Karen Marie Garrett, JimTV, Clint Mansell.

The creators behind the universes that inspire my sense of wonder and dreams for the future: Deus Ex, The Expanse, Cyberpunk, The Witcher, and so many others.

The amalgam of amazing, unique, curious minds not yet listed above that, to me, embody the delicate balance of creativity, depth, rigor, and rebelliousness that represents Caltech and the scientific spirit that I have swapped ideas with over the years: Eduardo da Veiga Beltrame, Anand Muthusamy, Samuel Clamons, Christopher Smith, Christopher Smith, Guruprasad Raghavan, Robert Gehle, Dylan Bannon, Jeremy Bernstein, Florian Schaeffer, Anish Sarma, Aiden Aceves, Christian Klaes, and surely others that my memory has failed to recall at this moment.

And finally, for everyone not listed here, because, frankly, the idea of making a binary threshold (see Section 2.5) between who contributed to this work, this field, and my personal growth, and those who did not, is an unsolvable problem. Humanity is a hypercollective organism, whether we like to admit it or not. Thank you to the employees at Taco Bell who fed me, and to the first humans who pondered at the sky and wondered what was out there, and to the first curious creature that wandered from the sea to the land, and to the first self-replicating nucleic acid molecule that began this journey of life on Earth.

ABSTRACT

The advent of massively parallel sequencing and synthesis technologies have ushered in a new paradigm of biology, where high throughput screening of billions of nucleid acid molecules and production of libraries of millions of genetic mutants are now routine in labs and clinics. During my Ph.D., I worked to develop data analysis and experimental methods that take advantage of the scale of this data, while making the minimal assumptions necessary for deriving value from their application. My Ph.D. work began with the development of software and principles for analyzing deep mutational scanning data of libraries of engineered AAV capsids. By looking at not only the top variant in a round of directed evolution, but instead a broad distribution of the variants and their phenotypes, we were able to identify AAV variants with enhanced ability to transduce specific cells in the brain after intravenous injection. I then shifted to better understand the phenotypic profile of these engineered variants. To that end, I turned to single-cell RNA sequencing to seek to identify, with high resolution, the delivery profile of these variants in all cell types present in the cortex of a mouse brain. I began by developing infrastructure and tools for dealing with the data analysis demands of these experiments. Then, by delivering an engineered variant to the animal, I was able to use the single-cell RNA sequencing profile, coupled with a sequencing readout of the delivered genetic cargo present in each cell type, to define the variant's tropism across the full spectrum of cell types in a single step. To increase the throughput of this experimental paradigm, I then worked to develop a multiplexing strategy for delivering up to 7 engineered variants in a single animal, and obtain the same high resolution readout for each variant in a single experiment. Finally, to take a step towards translation to human diagnostics, I leveraged the tools I built for scaling single-cell RNA sequencing studies and worked to develop a protocol for obtaining single-cell immune profiles of low volumes of self-collected blood. This study enabled repeat sampling in a short period of time, and revealed an incredible richness in individual variability and time-of-day dependence of human immune gene expression. Together, my Ph.D. work provides strategies for employing massively parallel sequencing and synthesis for new biological applications, and builds towards a future paradigm where personalized, high-resolution sequencing might be coupled with modular, customized gene therapy delivery.

PUBLISHED CONTENT AND CONTRIBUTIONS

Brown, D.*, Altermatt, M.*, Dobreva, T., Chen, S., Wang, A., Thomson, M., & Gradinaru, V. (2021). Deep Parallel Characterization of AAV Tropism and AAV-Mediated Transcriptional Changes via Single-Cell RNA Sequencing. In *Frontiers in Immunology* (Vol. 12). Frontiers Media SA. https://doi.org/10.3389/fimmu.2021.730825

D.B. participated in conceiving the project, designing and performing the experiments, and writing the manuscript.

Dobreva, T.*, Brown, D.*, Park, J. H., & Thomson, M. (2020). Single cell profiling of capillary blood enables out of clinic human immunity studies. *Scientific Reports*, 10(1). https://doi.org/10.1038/s41598-020-77073-3.

D.B. participated in conceiving the project, designing the experiments, performing single-cell isolation experiments, preparing sequencing libraries, analyzing the data, and writing the manuscript.

Ravindra Kumar, S., Miles, T. F., Chen, X., Brown, D., Dobreva, T., Huang, Q., Ding, X., Luo, Y., Einarsson, P. H., Greenbaum, A., Jang, M. J., Deverman, B. E., & Gradinaru, V. (2020). Multiplexed Cre-dependent selection yields systemic AAVs for targeting distinct brain cell types. *Nature Methods*, 17(5), 541–550. https://doi.org/10.1038/s41592-020-0799-7.

D.B. developed the sequencing data processing pipelines, performed data analysis, and made visualizations.

PUBLISHED CONFERENCE ABSTRACTS

Ravindra Kumar, S., Miles, T. F., Chen, X., Brown, D., Dobreva, T., Huang, Q., Ding, X., Luo, Y., Einarsson, P.H., Greenbaum, A., Jang, J.J., Deverman, B. E., Gradinaru, V. Evolution and Investigation of Engineered AAV Capsids Exhibiting Enhanced Transduction of the Central Nervous System with or without Murine Strain Specificity. 2020 ASGCT Annual Meeting Abstracts. (2020). *Molecular Therapy*, 28(4), 1–592. https://doi.org/10.1016/j.ymthe.2020.04.019

D.B. developed software for data analysis and visualization.

Padia, U., Brown, D., Ding, X., Chen, X., Ravindra Kumar, S., Gradinaru, V. Cloud-based Software for NGS Data Management and Analysis for Directed Evolution of Peptide-Based Delivery Vectors. 2020 ASGCT Annual Meeting Abstracts. (2020). *Molecular Therapy*, 28(4), 1–592. https://doi.org/10.1016/j.ymthe.2020.04.019

D.B. conceived the project and provided mentorship.

Brown, D., Altermatt, M., Dobreva, T., Park, J.H., Ravindra Kumar, S., Chen, X., Coughlin, G.M., Pool, A.H., Thomson, M., Gradinaru, V. A Computational and

Experimental Platform for Detecting Full Transcriptome Cell Type Tropism of Lowly Expressed Barcoded and Pooled AAV Variants via Single-Cell RNA Sequencing. 2020 ASGCT Annual Meeting Abstracts. (2020). *Molecular Therapy*, 28(4), 1–592. https://doi.org/10.1016/j.ymthe.2020.04.019

D.B. participated in conceiving the project, designing and performing the experiments, and writing the manuscript.

Ravindra Kumar, S., Chen, X., Deverman, B.E., Brown, D., Dobreva, T., Huang, Q., Ding, X., Luo, Y., Einarsson, P.H., Goeden,N., Flytzanis, N., Greenbaum, A., Gradinaru, V. Multiplexed-CREATE Selection Yields AAV Vectors Targeting Different Cell Types of the Central Nervous System Following Systemic Delivery. *Molecular Therapy* 27 (4), Art. No. 99 (2019). https://doi.org/10.1016/j.ymthe.2019.04.004

D.B. developed software for data analysis and visualization.

Ravindra Kumar, S., Chan, K., Jang, M.J., Huang, Q., Brown, B., Dobreva, T., Kim H.M., Luo, Y., Hurt, R.C., Chen, X., Deverman, B.E., Gradinaru, V. Developing AAV Vectors for More Efficient and Selective Gene Expression in Specific Cell Types of the Nervous System Following Systemic Delivery. ASGCT 21st Annual Meeting Abstracts. (2018). *Molecular Therapy*, 26(5), 1–459. https://doi.org/10.1016/j.ymthe.2018.05.001

D.B. developed software for data analysis and visualization.

T. Dobreva, D. Brown, S. Kumar, Y. Luo, R. Hurt, B. E. Deverman, V. Gradinaru. "Engineering novel adeno-associated viruses for enhanced transduction and target specificity across the CNS by adopting high-throughput in vivo and in silico methods". Session No. 468. Neuroscience 2016 Abstracts. San Diego, CA: *Society for Neuroscience*, 2016. Online.

D.B. participated in conceiving the project, performing data analysis, and preparing the poster.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS AND/OR TABLES

Chapter 1

INTRODUCTION

1.1 Introduction

Massively parallel nucleic acid sequencing and synthesis technologies have enabled researchers to analyze the state of an entire complex system, such as the brain or immune system, via thousands to billions of simultaneous DNA or RNA measurements, and with relatively straightforward protocols, also design and explore the effect of DNA or RNA modifications to those systems. This paradigm is poised to dramatically transform human medicine. On the diagnostic side, it enables researchers and clinicians to perform assays that do not require selecting a specific metric, but instead give an entire suite of metrics in a single experiment. On the therapeutic side, instead of having to develop and isolate a new small molecule or protein for each therapy, the building blocks of DNA or RNA therapies packaged by engineered delivery vehicles can be designed once, and then be customized for new applications. Such highly multiplexed DNA and RNA assays are already being employed for a wide variety of research applications and have seen success in areas as diverse as protein binding prediction, antibody complementarity determination, and cell-type-specific enhancer screening (Aharon et al., 2020; Forsyth et al., 2013; Li and Samulski, 2020).

One common thread between all these applications is their reliance on massively parallel sequencing (frequently called next-generation sequencing, or NGS) data. Using NGS data as an assay for these contexts presents many challenges that are distinct from their imaging or other traditional assay analogs. By nature of the NGS assay, wherein a library of DNA or RNA molecules is first recovered, then amplified, and then sequenced, NGS data is at least three steps removed from the underlying biological phenomena. The first step, molecule recovery, is highly dependent on the specific assay, but in all applications introduces some degree of signal loss, recovery bias, and biological noise. The second step, PCR amplification, while highly accurate, introduces artifacts and amplification bias

for some molecules over others. The third step, the sequencing itself, is almost always a subsampling of the library, and is limited by the cost and scale of parallel sequencing equipment; although modern sequencers can process billions of sequences in a single experiment, this still pales in comparison to the quantity of molecules present in the source tissue of most assays. An additional challenge of NGS data is that it almost always crosses into the realm of "big data," i.e. data that is large enough that researchers are unable to analyze it by traditional, manual data analysis methods on their local machine. This is both a challenge and an opportunity, as it necessitates automation and cloud or cluster computing, and the data is at a scale that is ripe for applications of machine learning.

In transitioning traditional non-NGS methods to their higher-throughput, parallel NGS counterparts, these challenges motivate the development of methods that specifically consider the nuances of NGS data. In my Ph.D., I aimed to establish general principles and develop software and analysis methods for operating with NGS data for bioengineering applications. With these principles in hand, I then applied them to experimental workflows in viral vector screening. Next, as a main proof of concept of the accuracy and effectiveness of these NGS data strategies, I worked with members of Viviana Gradinaru's lab to develop a new paradigm for characterizing the delivery profile of viral vectors in parallel via single-cell RNA sequencing. Finally, I worked with members of Matt Thomson's lab to translate some of these principles to direct applications for human health by co-developing a method to process and analyze immune gene expression profiles from self-collected, low-volume human blood samples. Collectively, these principles and methods contribute to the new and growing paradigm of obtaining high-dimensional data on complex biological systems and engineering modular therapeutic solutions to change the state of these systems.

1.2 Massively parallel sequencing for engineering adeno-associated viruses

Adeno-associated viruses (AAVs) are widely used as gene delivery vehicles for basic research, and are being evaluated in a growing number of gene therapy clinical trials due to their broad transduction and lack of pathogenicity (Kuzmin et al., 2021). However, naturally-occurring AAVs have delivery profiles that miss important therapeutic targets, particularly less accessible tissues such as the brain

(Zincarelli et al., 2008). To meet this need, AAV engineers have developed a variety of strategies to design, screen, and select for engineered AAV variants that have higher transduction efficiency, and even specificity for organs or cell types of interest. One of these methods, CREATE, pioneered by Ben Deverman and other researchers in Viviana Gradinaru's lab, takes advantage of the availability of engineered mouse lines expressing Cre recombinase in specific cell types and applies selective pressure to libraries of AAV variants (Deverman et al., 2016). This directed evolution approach has already yielded several highly transducing and cell-type-specific viral variants that are capable of crossing the blood-brain barrier and delivering genetic cargo to regions and cell types of interest (Chan et al., 2017; Deverman et al., 2016).

However, one of the remaining limitations of current directed evolution approaches is their inability to apply negative or combinatorial selective pressure to variant libraries. While methods like CREATE can find variants that are highly effective at transducing a certain cell type, it does not guarantee that the variant is exclusive to those cell types. To address this, researchers in Viviana Gradinaru's lab, led by Sripriya Ravindra Kumar, turned to NGS to gain insight into the virus mutants that are present in or absent from different recovered tissues. In Chapter 2, I discuss my work on processing and analyzing the data that emerges from these experiments. In Sections 2.3-2.4, I discuss some challenges that arise from analyzing data from such an NGS-based screen. In Section 2.5, I describe a computationally efficient method implemented in Python for correcting errors that arise during PCR amplification and sequencing. In Section 2.6, I document a Python package I developed that provides these and other convenience and visualization functions that are broadly useful for NGS and deep mutational scanning datasets. Finally, in Section 2.7, I show some of the results from the use of this software and analysis in discovering several AAV variants with enhanced tropism for vascular cells in the mouse brain (Ravindra Kumar et al., 2020).

One of the other major challenges facing both directed evolution and rational design approaches for engineering novel proteins for *in vivo* applications, such as AAV capsids for tropism specificity, is the combinatorial explosion of possible sequences to explore in a screen. Unlike protein engineering in bacterial or *in vitro* contexts where systems can be designed for continuous evolution, to date, *in*

vivo selective pressure experiments start with a library of fixed size, and perform selection in discrete experimental steps, often involving weeks to months of labwork. This constraint means that there is high value in knowing what sequences to include in a round of evolution.

These AAV NGS experiments have yielded rich datasets of hundreds of thousands of viral variants per experiment with quantitative measures of their tropism, and have revealed new variants with enhanced tropism for specific cell types. This field, and, more generally, NGS-based *in vivo* protein selection strategies, will likely produce an increasing data stream of mutant proteins and their measured effectiveness at a variety of therapeutically-relevant functions, as soon as such functions can be reduced to a sequencing readout. I hope that the analysis and principles presented here will have relevance to this exciting new paradigm of directed evolution.

1.3 Single-cell RNA sequencing for characterizing engineered adeno-associated viruses

As custom AAV and other gene therapy delivery vehicles are being developed, it is critical to understand their delivery and expression profile. Off-target delivery of gene therapies can lead to diminished therapeutic efficacy, immune response, and toxicity. Gene therapy trials, while already showing great promise in a variety of disease areas (Papanikolaou and Bosio, 2021), have also had several high-profile clinical failures that have resulted from unnecessary levels of off-target delivery or expression (Lehrman, 1999; Paulk, 2020; Servick, 2021).

One of the bottlenecks in gene therapy development and characterization is the full profiling of on- and off-target delivery and expression. While selection and screening methods as described above have become highly parallelizable in terms of the number of variants, they have not become highly parallelizable in terms of the number of cell types or tissues screened; each new cell type or tissue of interest requires additional animals or lab work to characterize. Thus, when selecting or characterizing new variants, vector developers are forced to choose a limited set of cell types and tissues. On the side of gene therapy users, researchers looking to deliver gene therapies for diseases or to answer research questions often have specific on and off-target needs for their application. However, the probability that a particular vector has been characterized by the vector developer in

exactly the cell types or tissues of interest to the gene therapy researcher becomes smaller as the number of cell types and tissues that are considered increases.

Parallel single-cell RNA sequencing (scRNA-seq) is a recently developed class of methods that allows researchers to interrogate the expression of thousands of genes in thousands to millions of cells in a single experiment (Klein et al., 2015; Macosko et al., 2015; Rosenberg et al., 2018; Zheng et al., 2017). These methods have given unprecedented insight into the diversity of cell types and states, and are proving an invaluable tool in defining cell type hierarchies and in understanding the role of cell subpopulations in the context of systematic function, disease, and response to perturbations. However, this increased level of understanding has a concomitant increase in expectations and needs; if a cell type implicated in a particular disease is revealed to contain several cell subtypes, it is only a matter of time before researchers will explore which roles each subtype plays within the greater disease context.

Thus, characterizing newly developed gene therapy vectors in all relevant cell types becomes untenable unless it can scale with the discovery of new cell types. Ideally, gene therapy vector developers could perform a scRNA-seq experiment with their newly designed vectors to characterize their performance in all cell types in a single experiment. Unfortunately, there are many obstacles precluding the rapid adoption of scRNA-seq as a characterization method. scRNA-seq datasets are impressively large; samples from a single, small region yield at least as much data as full 30X coverage whole genome sequencing (90 gigabases). Extending this to multiple regions, or multiple samples, can quickly outpace the data storage, processing and bioinformatics capabilities of manual workflows and non-cluster computing. In Section 3.2, I lay out some software abstractions that I believe are appropriate for thinking about and developing software for scRNA-seq workflows. Using these abstractions, in Sections 3.3-3.4, I describe software I developed to enable rapid access to scRNA-seq data across many samples, and a web-hosted software infrastructure for automating the most common pipeline elements of scRNA-seq samples.

Since scRNA-seq is still a relatively new paradigm, and has a constantly evolving technology landscape, an additional class of problems arises in translating scRNA-seq to new applications: identifying the cell types and states among the heterogeneity of cells and noise within a sample. While a multitude of methods exist for attempting to address these issues, they often involve setting somewhat arbitrary thresholds, make assumptions of homogeneity in the sample, and do not consider the biological knowledge a researcher has about their tissue of interest. In Section 5.7.14, I elaborate on a guided machine-learning-based approach to distinguish cell types from confounding noise, and identify cell subtypes confidently based on both known gene markers and automated discovery of cell type markers.

One final nuance in translating scRNA-seq for characterizing vector delivery is that, unlike most scRNA-seq applications which seek to discover cell types or states that can be measured by multiple genes or gene pathways, vector characterization requires a high fidelity recovery and readout of the presence or absence of the specific delivered gene. In Chapter 4, I discuss strategies for increasing the fidelity of identifying individual gene transcripts. In Section 4.2, I show why these strategies are critical to avoid misinterpretation due to PCR artifacts.

With all of these pieces in place, in Chapter 5, I discuss a new method for characterizing the unbiased cell type tropism of AAVs via scRNA-seq that I developed with members of Viviana Gradinaru and Matt Thomson's labs. In Section 5.3.1-5.3.2, I detail the pipeline we developed and show that it produces results in accordance with existing immunohistochemistry-based characterization. In Sections 5.3.3 and 5.3.5, I discuss cell-subtype-specific biases we uncovered from our unbiased approach. And, in Sections 5.3.4 and 5.3.7, I show how this method can scale not only to multiple cell types, but also to multiple AAV variants in parallel via a barcoding strategy. By providing principles, software, and a workflow that enables confident readouts of viral tropism, I hope that future vector development efforts can benefit from the parallelizability of NGS-based assays and report comprehensive on- and off-target delivery efficiency of their tools.

1.4 Translating single-cell RNA sequencing to human immune studies

The technology development of scRNA-seq has seen an impressive rate of improvement since its inception in 2009, and the number of cells interrogated per study has been growing exponentially (Svensson et al., 2018). This increase is primarily driven by technological developments that increase the usability of scRNA-seq protocols, and cost reductions on a per-cell basis. Unsurprisingly, these technology improvements have yielded steady growth in the number of scRNA-seq studies, with 412 studies published and curated in 2020, compared to only 54 in 2015 (Svensson et al., 2020). However, there is a curious additional trend: despite this explosive growth in technology adoption and scale, there is not a concomitant growth in the number of subjects included per study in human blood (**Figure 1**). This suggests that while the technology is scaling, the ability to acquire samples may present an additional bottleneck.

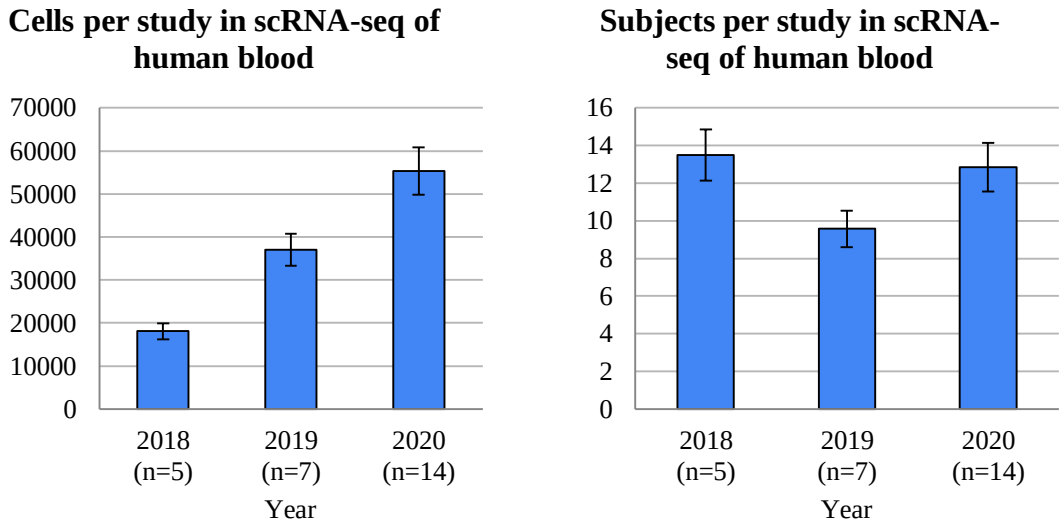


Figure 1. Growth of single-cell RNA sequencing studies. Growth of single-cell RNA sequencing in terms of number of cells per study (left) vs. number of subjects (right).

In collaboration with Tatyana Dobreva and Jong Hwee Park in Matt Thomson's lab, we explored the field of scRNA-seq on human blood, specifically peripheral blood mononuclear cells (PBMCs),

and found that the default method of sample collection is via venous blood draws, administered by a licensed phlebotomist. However, typical scRNA-seq experiments process on the order of 10^3-10^4 cells per sample, while whole blood contains approximately $4.5x10^3 – 1.1x10^4$ white blood cells per µL (Dean, 2005). Thus, in a typical venous blood draw of 10-15mL, less than 1% of the collected cells are used for eventual scRNA-seq workflows. This suggests the possibility of performing scRNA-seq not on phlebotomist-administered venous blood draws, but instead on less invasive and cumbersome procedures, such as capillary blood extraction which can be performed by researchers directly, or even the subjects themselves. In Chapter 6, we describe a protocol for extracting and isolating PBMCs from such small volumes of self-collected capillary blood. In Sections 6.3.1-6.3.2, we show how capillary blood can be used to obtain the same cell types as in traditional venous blood draws. With easy access to subjects and the low burden of capillary blood collection, we were able to perform a time-course study with two samples per day. In Sections 6.3.3-6.3.5, I discuss our findings on cell-type-specific gene expression that is dependent on the time of day, and showcase the incredible level of immune-relevant individuality in gene expression profiles between subjects. Our hope is that using protocols that can operate on self-collected, low-volume capillary blood will allow single-cell studies to scale in number of subjects in tandem with the technology, and help society gain a more robust understanding of the individuality present in the human immune system: a critical component to developing increasingly personalized molecular, cellular, and genetic therapies.

Chapter 2

PRINCIPLES FOR ANALYZING DEEP MUTATIONAL SCANNING DATASETS

2.1 Summary

Applying directed evolution to AAV engineering has resulted in substantial improvements in efficacy of gene therapy delivery to a variety of cell types and tissues. Of particular relevance to neuroscience and brain disorders, the Cre-dependent system developed by Deverman et al. (2016) has yielded AAV-PHP.B, AAV-PHP.eB, and AAV-PHP.S, three variants with enhanced transduction of different subpopulations of cells in the central nervous system via systemic injection. This system works by inverting a segment of the delivered viral cargo sequence and flanking it with Lox sites and then injecting into Cre-expressing animals, whereafter amplification protocols selectively extract only viruses that have successfully transduced the cell types expressing Cre.

While incredibly powerful, this molecular selection strategy is limited to applying positive selective pressure for transduction of specific cell types, and cannot apply selective pressure to detarget other cell types. Furthermore, directed evolution that operates in such discrete experimental steps as opposed to continuous mutagenesis under selective pressure is limited by the scale of the input library and the throughput of the final readout, and is dependent on the effectiveness of the selective pressure.

The use of next-generation sequencing data for screening and analyzing libraries of engineered mutants, a paradigm referred to as deep mutational scanning, can provide a high-throughput readout of the results of a directed evolution selection round (Forsyth et al., 2013; Fowler and Fields, 2014; Starita and Fields, 2015; Wrenbeck et al., 2017). In many cases, deep mutational scanning can provide a significant boost to the expected efficiency of the output of a screen. However, given that

deep mutational screens cost additional labwork and sequencing reagents, it is important to understand in what contexts and applications the increased efficiency outweighs the costs. In this work, I present a framework for assessing the probability that incorporating deep mutational scans into a directed evolution paradigm will lead to improved variants.

From the perspective of this framework, there are several properties of Cre-dependent directed evolution of capsids that suggest that deep mutational scanning is beneficial and even necessary for evolving certain properties of capsids. In this vein, I worked with Sripriya Ravindra Kumar and colleagues to incorporate an NGS readout on the pre- and post-selection Cre-dependent AAV libraries with the goal of engineering capsids with enhanced specificity for particular cell types. In this work, I describe the metrics that we determined to screen for capsids with specific properties, and elaborate on the unique data analysis elements of deep mutational scanning experiments, in particular with regards to the subsampled nature of NGS data. In addition, there are some considerations for the particular paradigm of performing saturation mutagenesis on small regions, such as the 7-mer amino acid insertion at the 588-589 loop region of the AAV capsid we explored. Finally, I developed a suite of software tools to incorporate this analysis framework and make the analysis of deep mutation scanning experiments more user-friendly, while helping to manage the large number of samples and variants considered in such multiplexed screens. As proof of the utility of deep mutational scanning and these analysis methods for screening AAVs, we show that several novel engineered capsids arose from this screen with useful cell-type-specific tropisms that otherwise would likely not have been detected using a traditional directed evolution approach alone.

2.2 Directed evolution vs. deep mutational scanning

As evidenced by the Nobel Prize in Chemistry in 2018, directed evolution—i.e., the use of selective pressure combined with mutagenesis to evolve proteins or cells for desired function—has had a profound impact on science, ranging from evolution of enzymes for increased catalytic function, to optimization of binding affinity for antibodies via phage display, to evolution of AAVs for enhanced delivery of gene therapy to the brain (Arnold, 1998; Chen and Arnold, 1993; Deverman et al., 2016;

Smith, 1985). When successful, the products of directed evolution can have orders of magnitude of increased effectiveness above their parent. Thus, when faced with a protein engineering challenge, directed evolution is often an appealing first candidate strategy.

However, while directed evolution relieves the engineer of the burden of having to rationally design the optimal protein or cell, it burdens the engineer with a new task: rationally designing the selective pressure applied to the library. This task is so important, its implications have been dubbed the first law of directed evolution: "you get what you screen for" (Arnold, 1998).

A directed evolution selection strategy can be thought of in terms of several metrics that collectively determine the expected increase in fitness obtained in a round of selection. Determining, or at least estimating, these metrics can be useful in deciding between possible selection strategies, including whether or not to add a deep mutational scanning step to each round of selection. These metrics are:

- The number of mutants in the input library, *N;*
- The number of molecules in the output library, *M;*
- The number of mutants that can be isolated in a round of evolution, *n.*

If we model the effectiveness of the selective pressure by the distribution of the mutant counts post-selection, F(X), we can use this distribution and the above metrics to predict the probability of picking a mutant that is among the top mutants in an output library.

As a simple example, let us assume uniform selective pressure, meaning each mutant is equally likely to be selected at each opportunity. In this case, the distribution of the number of times we see a particular mutant in our output library, *M*, can be modeled by the Poisson distribution, where the rate parameter, $\lambda = \frac{M}{N}$. In the case of directed evolution, typically M >> N, therefore $\lambda >> 1$, since there are many copies of each mutant due to some amplification as part of the selection process.

For example, in a simple case of N=1000 mutants with an output library size of M=100000, we expect the familiar Poisson distribution centered near λ (**Figure 2. Example mutant distribution under uniform selection pressure.** Poisson distribution conditioned on X>0 for an estimated mutant library size 100-fold smaller than the output library.Figure 2).

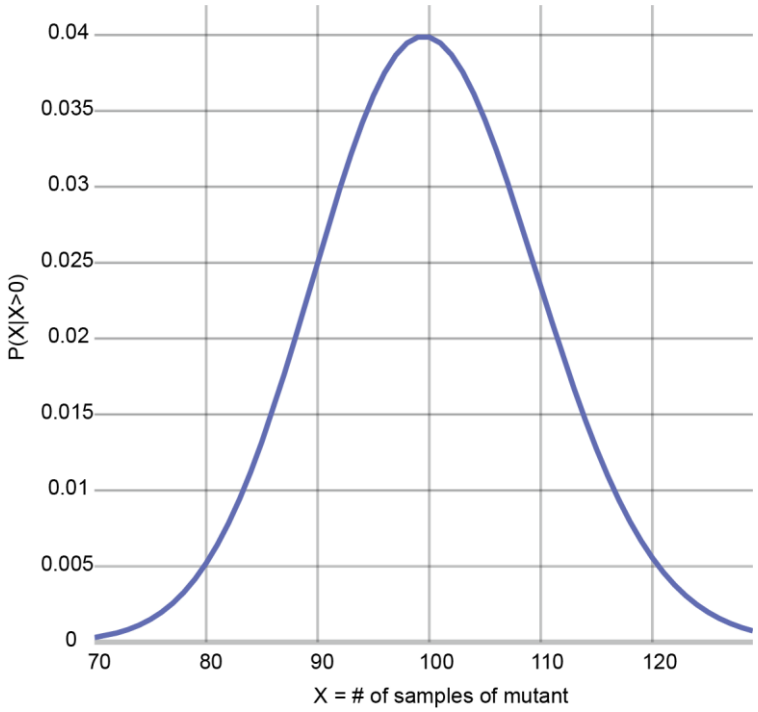


Figure 2. Example mutant distribution under uniform selection pressure. Poisson distribution conditioned on X>0 for an estimated mutant library size 100-fold smaller than the output library.

We can then think of a non-uniform selective pressure function, F, as the same process, but with increased selective pressure strength that leads to overdispersion of the Poisson. Thus, we can make a simple 2-parameter model of selective pressure via the negative binomial distribution, where the mean, μ, represents the same value, λ, as it did in the case of the Poisson distribution; that is, the ratio between our sample size post-selection and our initial mutant library size, but with a new dispersion parameter, α, that represents the strength of the selective pressure. This is a common parameterization of the negative binomial, and can be converted to the traditional number of successes, n, and probability of success, p, with the following transformations:

$$\sigma^2 = \mu + \alpha\mu^2$$

$$p = \frac{\mu}{\sigma^2}$$

$$n = \frac{\mu^2}{\sigma^2 - \mu}$$

It can then be seen that as the selective pressure strength, α converges to 0, the variance approaches the mean, representing the Poisson:

$$\lim_{\alpha \to 0} \sigma^2 = \lim_{\alpha \to 0} \mu + \alpha\mu^2$$

$$\lim_{\alpha \to 0} \sigma^2 = \mu$$

Now that we can represent the strength of the selective pressure, it would be helpful to know if there are different regimes of selective pressure strength that lend themselves to different selection strategies. In particular, under what regimes is there an advantage in performing deep mutational scanning. For this, we can turn to simulations of these negative binomials. For a given negative binomial with $\mu = \frac{M}{N}$ and selection strength α, we can generate a theoretical output library that represents the counts of each mutant before sampling. If we assume that the mutants have multiplied many times over such that the count of each variant, $m_i >> n$ (our sampling depth) for all mutants i, we can then sample from these variants, with replacement, weighted by their relative count. To know whether we sampled one of the top k variants within the pool, if we let K = the set of top k variants, we can calculate:

$$P(K|n) = 1 - \left(1 - \frac{\sum_{i \in K} m_i}{\sum_i m_i}\right)^n$$

The question we ultimately want to know for screens of different scales, is what is the probability that we will uncover our mutant of interest if we pursue the larger screen. We can calculate this

probability of getting at least one of the top *K* variants in a larger sample size, n_1, but not in a smaller sample size, n_2, as:

$$P(K|n_1) * (1 - P(K|n_2))$$

When modeled like this, we can explore the different regimes in which deep mutational scanning, which samples on the order of 10^6 variants per experiment, will yield one of the desired variants, whereas a smaller screen, like picking clones, which is on the order of 10 variants per experiment, will not.

For a fixed estimated library size of N=10000 mutants, I explored the probability of finding a top *k* variant across a variety of dispersion parameters and different values for *k* (Figure 3). Unsurprisingly, two trends are revealed: as selective pressure increases and as *k* increases, the value of a deep mutational scan decreases. Next, I explored the effect of library size on this phenomenon. Again, unsurprisingly, as library size increases, the value of a deep mutational scan increases; however, this effect goes away as the library size becomes unmanageable for either strategy. Together, these simulations suggest a particular regime of directed evolution that benefits from the extremely large number of samples obtained from deep mutational scanning; however, there are scenarios, such as in regimes of very high selective pressure or small library sizes, where deep mutational scanning will have very little benefit. For Cre-dependent AAV engineering applications, where library sizes are estimated to be on the order of 1e6 variants, and we are interested in capturing the select top few variants that maximize transduction, deep mutational scanning is likely to confer a large benefit across all but the strongest selective pressures.

Finally, it is worth noting that there is another case where deep mutational scanning is critical: namely, when the desired property of the protein to engineer does not have a selective pressure strategy available. For this, engineers can use the data that comes from a deep mutational screen to explore additional, analytical phenotypes of their mutants.

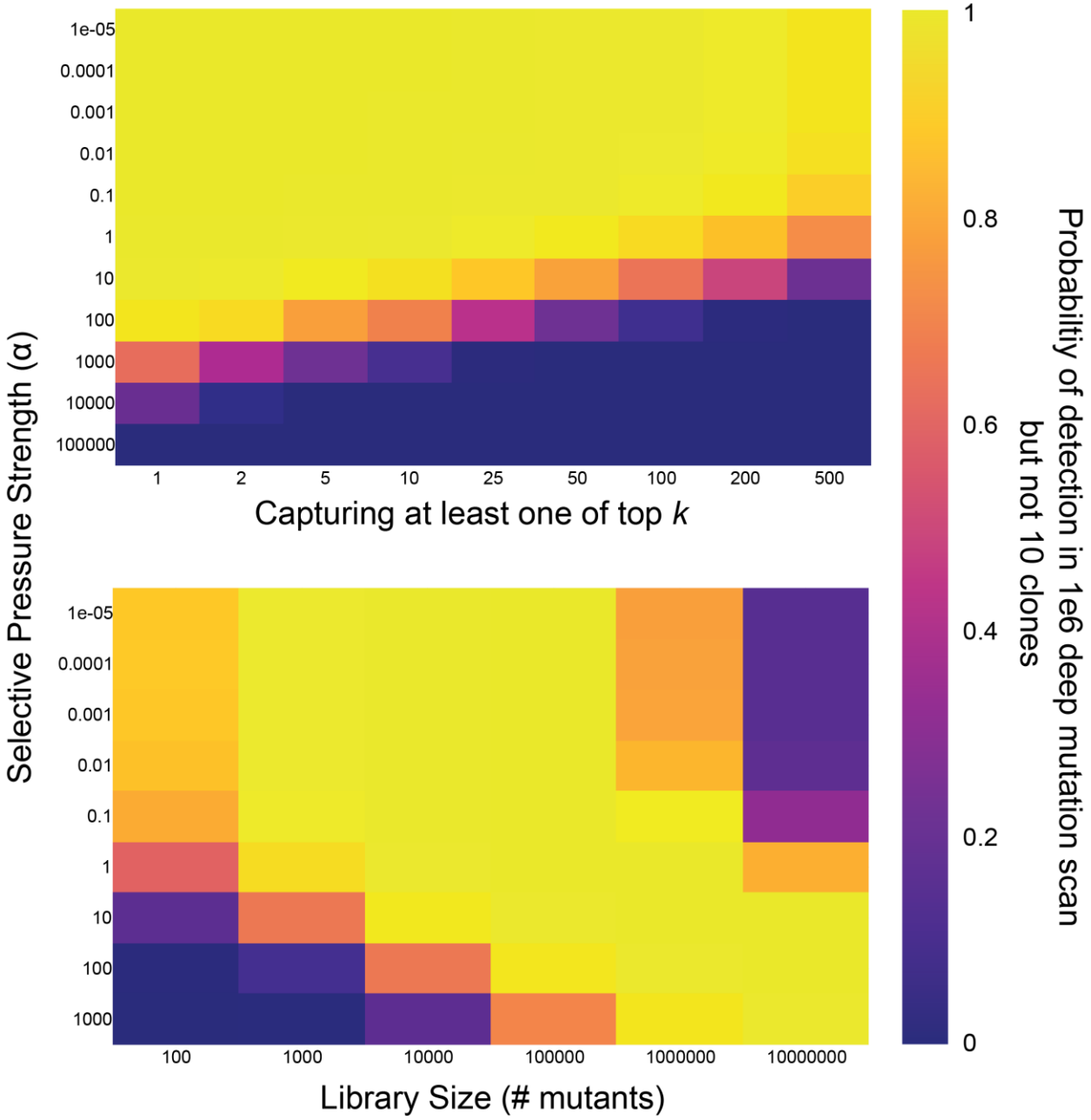


Figure 3. Estimated boost in probability of detecting a top *k* variant in a deep mutational screen vs. traditional colony picking. (top) For a fixed library size of 10000 variants, the probability of improvement for a variety of selection strengths (y) and values of top *k* (x). (bottom) For a fixed *k*=10, the probability of improvement for a variety of selection strengths (y) and library sizes (x).

2.3 Calculating enrichment and specificity

Deep mutational scanning data has several unique components in the data analysis workflow that need to be considered when analyzing the fitness of variants. Arguably the most critical of these is determining the metric that will be used to score and rank the variants. A first approach for ranking variants for tropism in a recovered tissue or cell type might be to count the number of times a variant is detected in the deep mutational scanning data. These counts are a direct measure of the selective pressure applied to the library, and thus can act as a proxy for the selective pressure. Choosing variants from a library based solely on the abundance of their recovered counts is equivalent to picking colonies from a dish of variants that have survived a selective pressure round. For some applications, this may be enough, and, as shown in Section 2.2, already provides value in many experimental regimes. However, one of the advantages of deep mutational scanning is that researchers can create *in silico* selective pressure analogs through analysis. The choice and implementation of these metrics are critically important, as they will provide the selective pressure for downstream rounds. Analogous to the first law of directed evolution, a first law of deep mutational scanning might be "you get what you analyze for."

In the case of AAV tropism evolution, AAV variants recovered from tissue have already undergone three distinct rounds of selection by the time they are recovered. The first round of selection happens during DNA synthesis and cloning, where there might be a bias from PCR amplification or bacterial production. The second round of selection happens during virus production, where AAV variants that fail to form capsids, or form capsids that do not survive the purification procedure, are filtered. The final round of selection is from the effectiveness of the variant at navigating to the cell type and tissue of interest. Unlike some other deep mutational scanning contexts, the selective pressure that we are primarily interested in is only the final step—transduction. If a variant has high effectiveness at transducing a cell type of interest, but only moderate ability to produce functional variants, variant manufacturing can usually scale up to meet the demand. Therefore, we are interested in the change in distribution of variants not from our starting DNA pool, but rather our post-production virus pool.

This suggests our first metric of interest for AAV transduction: tissue enrichment. Tissue enrichment can be calculated as the relative change in abundance of a variant between the post-production virus pool and the variant pool in the sample of interest. Given $c_{i,j}$ as the count of variant i in sample j, we can then define abundance, p, as:

$$p_{i,j} = \frac{c_{i,j}}{\sum_k c_{k,j}}$$

And from here define enrichment, e, over the post-production sample, v, as:

$$e_{i,j} = \log \frac{p_{i,j}}{p_{i,v}}$$

Note that $e_{i,j}$ is undefined if the variant has 0 abundance in the virus production pool. I discuss workarounds for this below (see Section 2.4 The zero count problem).

In addition to having high efficiency at targeting a tissue or cell type of interest, one of the properties of interest for AAV capsids is to have specificity; that is, low transduction of off-target tissue types. More generally, for a set of on-target tissues, J, and off-target tissues, K, we can calculate the specificity, s, of variant i, as:

$$s_{i,J,K} = \log \frac{\sum_{j \in J} p_{i,j}}{\left(\frac{\sum_{j \in J} p_{i,j} + \sum_{k \in K} p_{i,k}}{|J| + |K|} \right)}$$

An example of these metrics is given in Figure 4. With these two metrics in hand, researchers implementing a multiplexed AAV directed evolution paradigm can now choose any combination of on- and off-target tissues to fine-tune their desired AAV tropism.

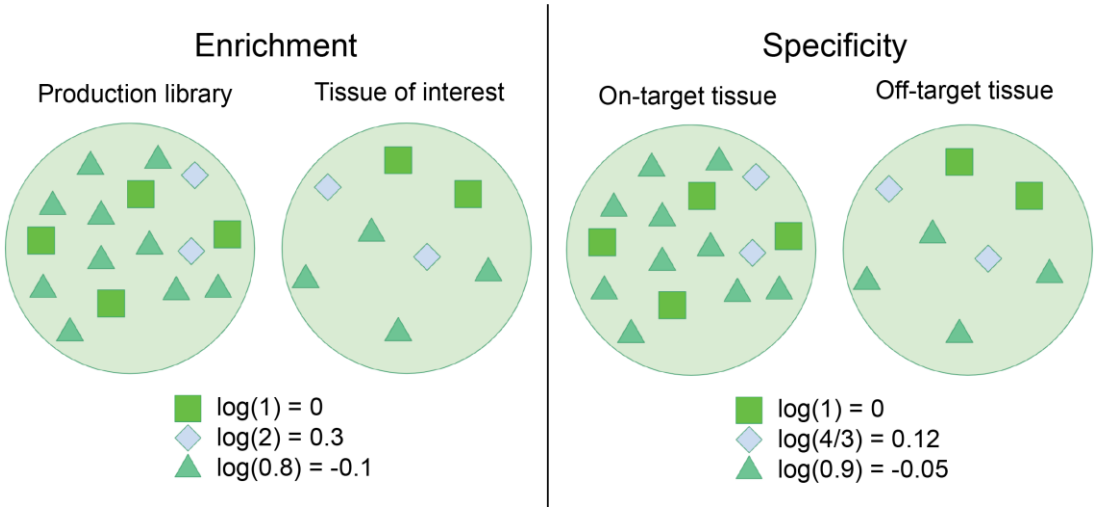


Figure 4. An example of enrichment (left) and tissue specificity (right)

2.4 The zero count problem

Variant read counts derived from NGS data in most deep mutational scanning contexts are a substantial subsampling of the total pool of recovered molecules. Thus, it cannot be assumed that the absence of a mutant in the NGS data means the variant was not present in the pool; mathematically, it means the absence of a variant in a pool cannot be treated as a 0. This is additionally important for calculating fitness metrics (see Section 2.3 Calculating enrichment and specificity) where the denominator of the metric contains a read count; if the read count is 0, the fitness metric is undefined.

One common strategy for dealing with zero counts in count data is an additive method; that is, for calculations that require the use of counts that may be 0, such as the log transform, every count c is replaced with a count $c' = c + k$, where k is some manually defined constant, often 1 (Lindstone, 1920). The problem with this approach is that count data is not normally distributed, so adding a constant to each value skews the distribution. Also, there are many factors involved that change the meaning of a 0 in different datasets, such as sequencing depth, and the variance of the count distribution. As an extreme example, seeing a variant 0 times in a dataset of only 1 sample does not carry the same weight as seeing a variant 0 times in a dataset of millions of samples; the latter gives

more confidence that the variant might not be present in the library. Despite these many shortcomings, the simple additive transform continues to be used in many applications, risking misinterpretation of substantial scientific findings (Booeshaghi and Pachter, 2021).

Instead of using an additive method, to estimate the probability of an unseen variant, I perform a two-step procedure. The first step is to estimate the probability that, if we were to sequence 1 more read, the read would originate from a previously unseen variant. This calculation is inspired by the first step of estimation in the Good-Turing frequency estimation procedure (Good, 1953), and is:

$$p_0 = \frac{N_1}{N},$$

where N is the number of variants in the dataset, and N_1 is the number of variants detected with a count of 1. Since the single read counts in sequencing data can often be attributed to mutations of existing variants, I perform an additional correction step to this estimate. I first calculate the expected number of erroneous sequencing reads, E, based on a cumulative sum of the per-base read errors as reported in the FASTQ file. Then, I simulate a large number of point mutations from the data and calculate the frequency with which these mutations result in a sequence that is already present in the data; the complement of this is an estimate of the probability, p_e that any given sequencing read error results in a sequence that does not otherwise exist in the dataset. I then use this to estimate how many single count reads can be accounted for by mutation and sequencing errors by multiplying the number of expected error reads by the probability, $N1' = E*p_e$. Finally, I subtract this from the expected number of unseen, and obtain:

$$p_0' = \frac{N_1}{N} - \frac{N_1'}{N}$$

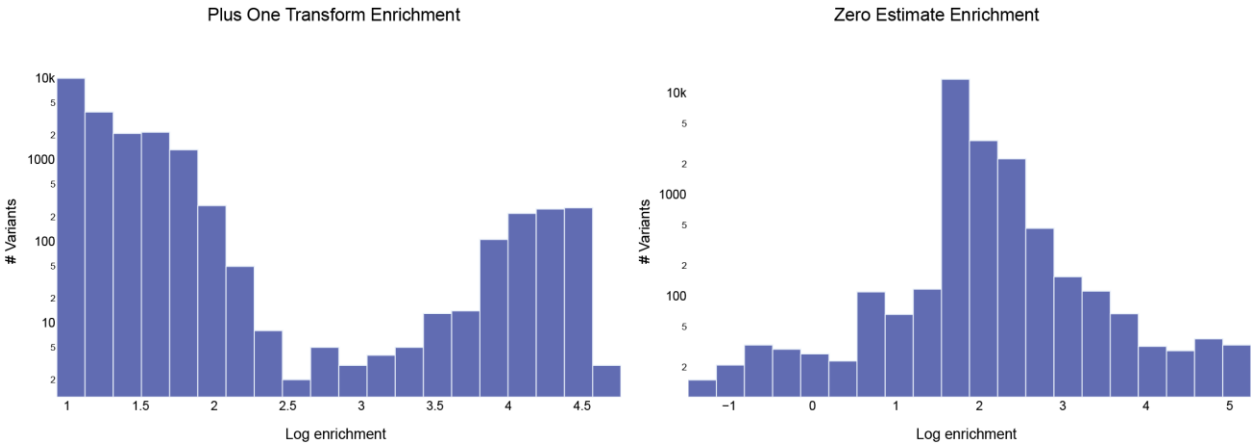


Figure 5. Distribution of enrichments under different transforms. Using log plus 1 transform (left) vs empirical zero estimate (right).

This number now replaces zeros when encountered in enrichment and specificity calculations. Typical values for this estimate in our AAV variant libraries range from 1e-6 to 1e-1, and can be shown to reduce artifacts that otherwise show up in traditional log plus 1 transforms. For example, performing the log plus one transform on missing variants in enrichment of tissue recovery counts over virus production counts creates an artifact where every variant has a positive log fold enrichment, despite clear examples where the variant is present in the virus production library at moderate counts, but depleted in the recovered tissue, representing negative enrichment (Figure **5**, left). The zero count transformation reduces this artifact and produces negative enrichment values for variants in this regime (Figure **5**, right).

Looking at different values of p_0' in relation to the histograms of counts shows some expected trends. Large sequencing depth libraries with read counts of mostly 1 or 2 have a very high p_0' estimate (0.7), whereas similarly high sequencing depth libraries with a stretched out distribution have much lower estimates (Figure 6A, B). Libraries with low sequencing depth but moderate skew have p_0' estimates in the middle, whereas a low sequencing depth library dominated by a single sequence has an extremely low p_0' (Figure 6**C, D**).

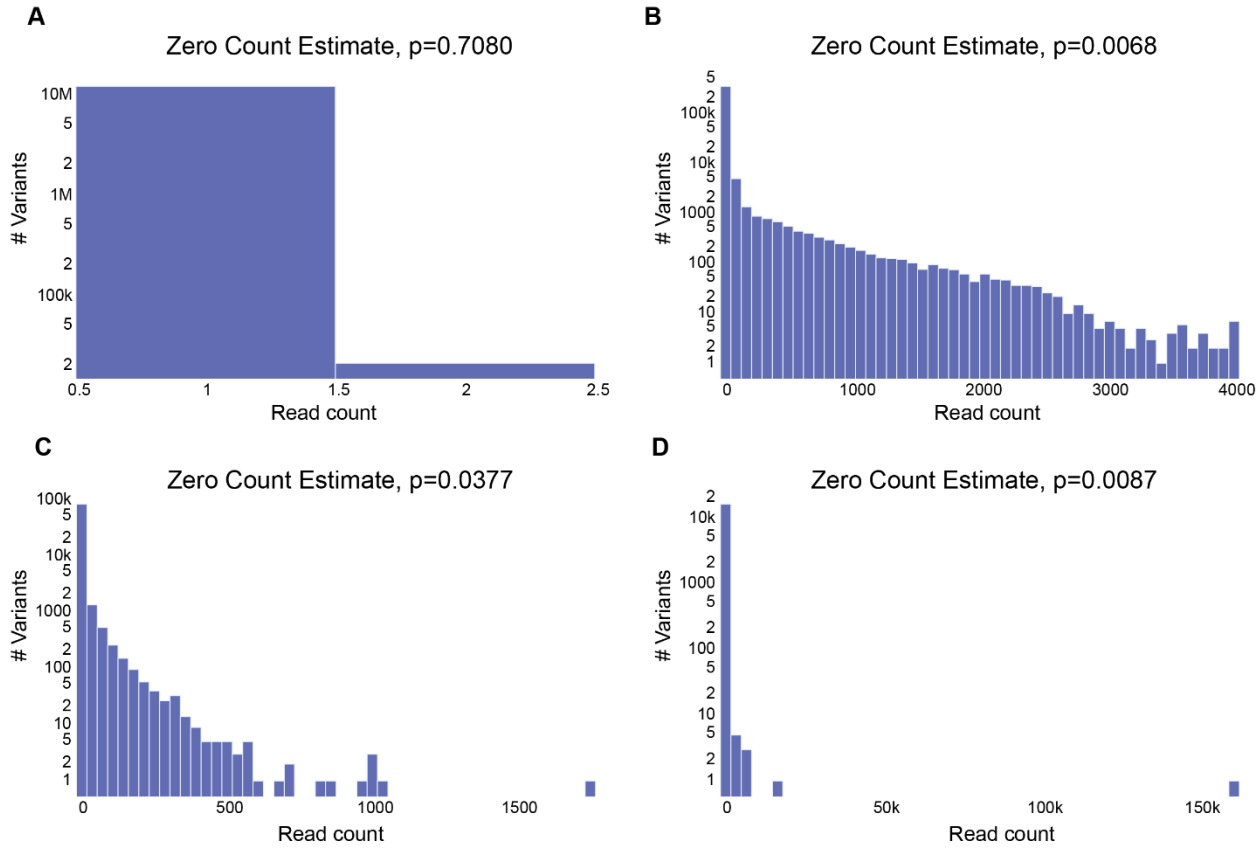


Figure 6. Estimated non-zero probabilities in different empirical data regimes. (A) An undersampled library, with read counts of only 1 or 2. **(B)** A diverse, high variance mutant library. **(C)** A moderately undersampled library with some diversity. **(D)** An oversampled library with only 1or 2 dominant sequences.

While these zero count estimates do provide a seemingly better transform than the traditional additive transformations, there are likely more principled estimates possible, especially given the scale of the data. For example, one could fit a distribution based on the known read counts, and then use this distribution, coupled with an estimate of the library size, to better estimate the probability of a variant being present in the library, but just not sequenced. Additionally, one could incorporate mechanistically-inspired models of PCR, sequencing errors, or other library amplification steps to better understand the distribution of low counts, and whether they constitute undersampled library as opposed to technical artifacts. However, the method described herein provide a better estimate

than transforms that do not depend on the distribution of the sequencing input data, is straightforward to implement, and is available as part of the pepars Python package (see Section 2.7.2 protfarm).

2.5 Correcting PCR and sequencing errors

Errors during early rounds of PCR amplification that occur during preparation of deep mutational scanning libraries can introduce high-count artifacts in the data that are mutants of their higher-count parents. In a selective pressure context, where the top-performing variants may have orders of magnitude higher performance than lower performers, these artifacts can have counts as high as or even higher than low performing, non-artifact variants. In directed evolution library protocols such as site-saturation mutagenesis or error-prone PCR on large fragments, these artifacts can be easier to identify by their deviation from the expected types of mutants present in the data. For example, in site-saturation mutagenesis, a variant that contains two or more mutations at low count is likely to be an artifact, since only point mutations are expected. Similarly, in error-prone PCR on large fragments, the probability that an error on a parent sequence yields a mutant that could have come from another parent is extremely unlikely, and becomes increasingly unlikely the larger the fragment that undergoes error-prone PCR.

In the directed evolution paradigm of using degenerate primers to produce large libraries with multiple mutations in a small region, however, it is likely that several steps of errors from a parent variant could yield a sequence that is identical to the sequence of an erroneous mutation from another variant in the pool.

To address these artifacts, we developed a strategy to correct for such PCR and sequencing errors. The underlying assumption of our strategy is that counts of child mutants will have less than half the counts than their associated parent, and this assumption holds as long as each variant has more than one copy in the pre-amplified library, and the PCR error rate is less than 50% (much greater than typical PCR error rates).

The procedure is as follows:

1. Sort the sequences in ascending order, by count.
2. Pick the next candidate child sequence from the sorted list. Find if there are any sequences that are hamming distance one away with at least twice the count of the candidate child sequence.
    a. If so, distribute the counts of the candidate child sequence into the parent sequences, proportionally to the parent read counts.
3. Repeat step 2 for all sequences.

A visualization of an example of this procedure is presented in Figure 7.

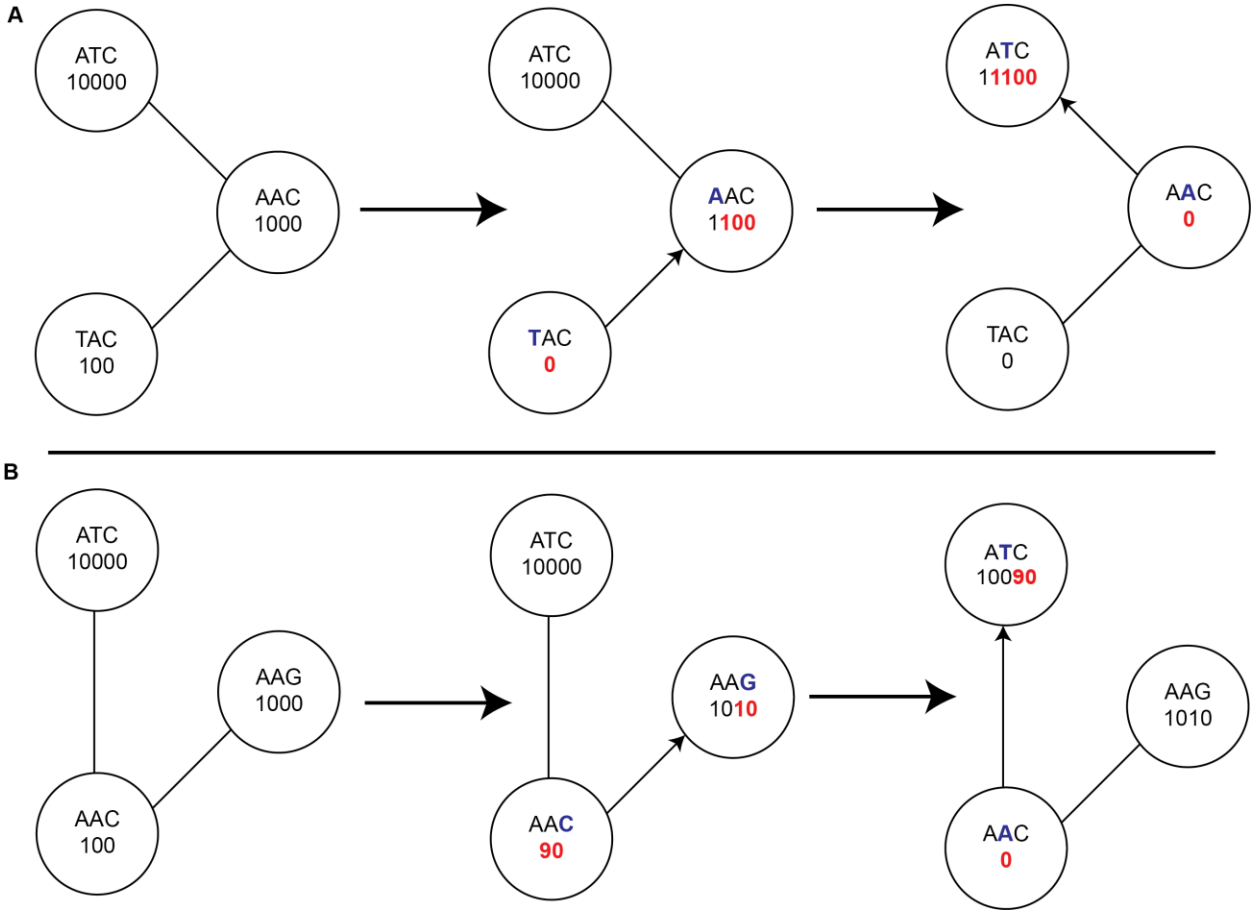


Figure 7. Variant collapse schematic. The procedure in the sequential case **(A)**, and the multiple parent case **(B)**.

Performing this procedure naively by searching for all possible hamming distance 1 parent strings in a list is an extremely computationally expensive procedure with run time $O(mn^2)$, where m is the number of nucleotides in the sequence, and n is the number of sequences in the dataset. For even small datasets, such as the 10^5 21-nucleotide sequences we encounter in our AAV selections, this can take hours to days to run on a modern computer.

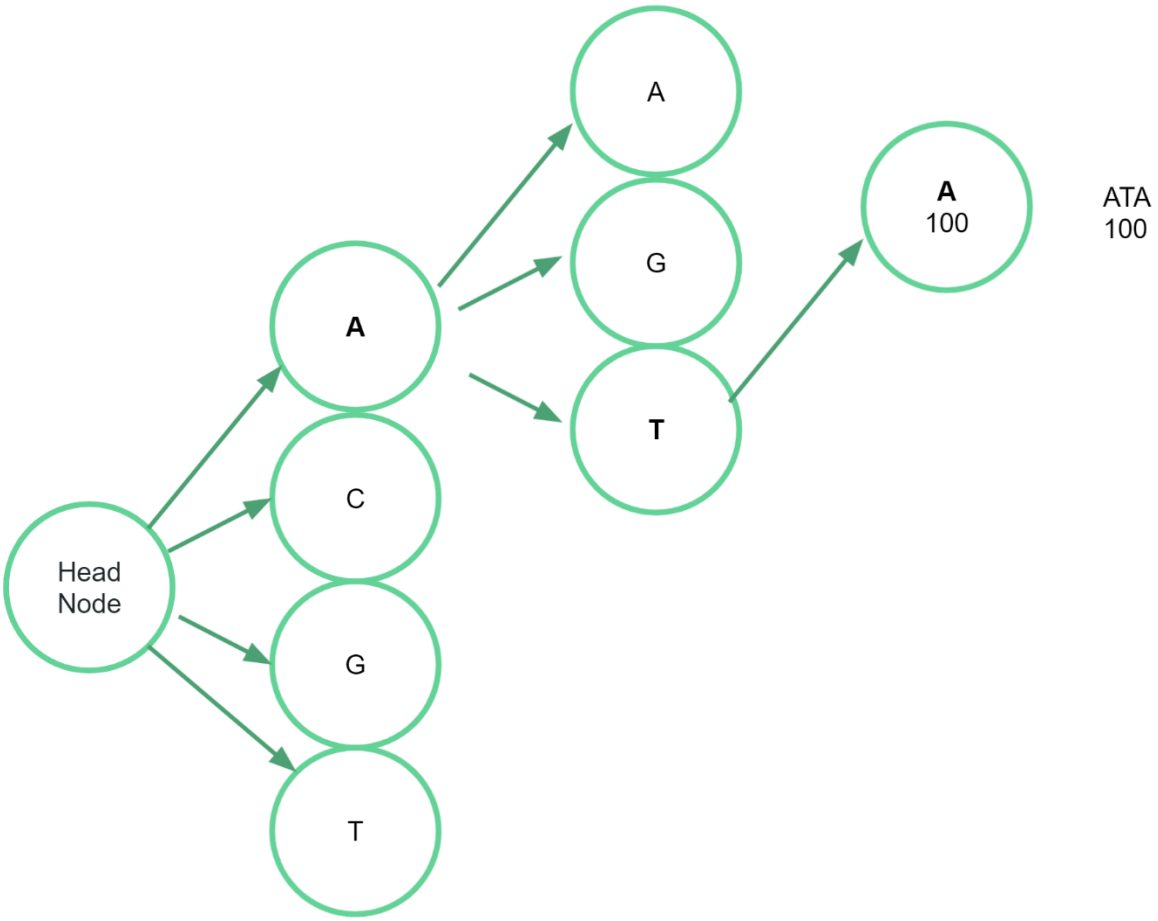


Figure 8. Schematic of a Sequence Trie and the nodes traversed for a 3-nucleotide lookup.

Fortunately, there is a computer science data structure that is optimally suited for this problem: tries. Tries are data structures that represent data as a tree, with each node in the trie containing a pointer to all existing subsequent characters. Structured for nucleotide data, an example trie would look like Figure 8. The benefit of a trie for collapsing similar sequences is that rather than looking for a possible parent sequence with hamming distance 1, it generates all possible hamming distance 1 sequences from the candidate child sequence, and queries the trie to see if it exists. A similar procedure could be accomplished by storing sequence counts in a hash table; however, sequence tries enable the additional optimization of short-circuiting all subsequent lookups if a parent node in the trie does not exist.

For example, imagine we are looking for all possible parent sequences of AAA. The exhaustive list of hamming distance one possible parents is: AAG, AAC, AAT, AGA, ACA, ATA, GAA, CAA, TAA. In a hash table implementation of the lookup, each lookup requires a search through the hash table from scratch; however, with a trie, the algorithm can start at the current node (AAA), traverse to its parent (AA), and immediately access the exhaustive list of possible hamming distance one parents at this node (AAG, AAC, AAT). Although modern hash table implementations are impressively fast and boast worst case lookup times of *O(k)* and average lookup times of *O(1),* the pruning of large numbers of nodes can make up for the theoretical *O(k)* average lookup times.

This collapse procedure using a Sequence Trie is implemented in the pepars Python package (see Section 2.6 pepars: A Python package for manipulating NGS data). There has been a more optimized Java implementation of a similar collapsing procedures released and described in a recent article (Liu, 2019), but for quick usage in Python notebooks, this implementation may be of use, and can collapse variant count datasets of 10^5-10^7 variants in seconds-minutes.

2.6 pepars: A Python package for manipulating NGS data

While several tools now exist for analyzing deep mutational scanning experiments for different contexts (Bloom, 2015; Rubin et al., 2017), they were either not yet developed or lacked critical features at the time of the first AAV deep mutational scanning datasets originating from Viviana Gradinaru's lab. Thus, Tatyana Dobreva, Pétur Helgi Einarsson, and I developed a suite of software for managing the large number of samples generated by multiplexed deep mutational scans of AAV variants. There are 3 main software components:

- **pepars** is a suite of helper, visualization, and analysis functions needed for many deep mutational scanning contexts.

- **protfarm** is a package for storing, organizing, and querying the raw and aligned data coming from deep mutational scanning experiments, particularly useful in the case when there are 10s to 100s of different experiments and FASTQ files.

- **protfarm-gui** is a user interface built in tkinter that interfaces with the protfarm package to provide user-friendly access to its functions. This was built primarily by Pétur Helgi Einarsson, and documentation can be found at https://github.com/GradinaruLab/protfarm-gui/blob/master/doc/manual.tex.

2.7.1 pepars: Protein Engineering via Parallel Sequencing

pepars (**P**rotein **E**ngineering via **Par**allel **S**equencing) is a Python package containing various utilities for dealing with parallel sequencing data (e.g. NGS FASTQ files) for protein engineering contexts.

2.7.1.1. Installation

You can install pepars via pip:

pip install git+https://github.com/GradinaruLab/pepars.git

2.7.1.2 Functionality

pepars is broken up into packages roughly based on functionality. The main packages are:

- alignment
- analysis
- fileio
- plotting
- utils
- simulation

Details about the functionality of each package are below.

2.7.1.3 Alignment

The alignment package contains functions for taking FASTQ files and extracting variant regions. The main element of the alignment package is the Aligner class, which is an abstract class that defines how alignment should function.

To perform alignment on a set of FASTQ files, instantiate one of the subclasses of Aligner (e.g. Perfect_Match_Aligner or Bowtie_Aligner), and then call the align function. See examples/alignment.ipynb for a typical use case.

The alignment parameters vary by aligner, as below:

2.7.1.3.1 Perfect_Match_Aligner Parameters

```
variant_sequence_quality_threshold=0
mismatch_quality_threshold=0
```

2.7.1.3.2 Bowtie_Aligner Parameters

```
working_directory=os.getcwd()
is_local=False
output_frequency=1e5
approach=None
allow_insertions_deletions=False
quality_threshold=0.0
working_directory=os.getcwd()
is_local=False
output_frequency=1e5
approach=None
allow_insertions_deletions=False
quality_threshold=0.0
```

2.1.1.3.2 Subclassing Aligner

To subclass the Aligner class, your class needs to implement the internal _align method.

2.7.1.4 Analysis

The analysis package has a variety of analyses relevant to parallel sequencing-based protein engineering experiments. Some examples are:

- amino_acids.get_amino_acid_codon_biases: Get the expected bias of amino acids for a degenerate nucleotide sequence.
- sequencing_reads.get_nucleotide_distribution: Get the distribution of nucleotides in a FASTQ file.
- confidence.get_sequence_confidences: Get the normalized confidences of a list of sequences and their counts, based on a few different confidence metrics.

2.7.1.5 Fileio

Some simple file wrappers, designed to make reading/writing CSV and sequence count files easier.

2.7.1.6 Plotting

A set of wrappers around Plotly, designed to generate plots useful for massively parallel sequencing with only one or a few lines of code, either interactively or exported. Most of the plotting functions in here internally call plotting.generate_plotly_plot, which is a catch-all Plotly wrapper that prints to screen, or writes to a file, or both. To have plots generate interactively, make sure to start your notebook with plotting.init_notebook_mode().

Some useful plots:

- plotting.plot_histogram: A simple Plotly histogram wrapper
- plotting.plot_scatter: A simple Plotly scatter plot wrapper
- plotting.plot_count_distribution: Plot the count distribution of variant sequences over multiple samples

- DNA.plot_amino_acid_bias: Plot a heatmap of amino acid bias, given sequence counts and a template

2.7.1.7 Utils

A variety of potentially useful utility functions. Some highlights:

- DNA: All the typical bioinformatics stuff: IUPAC grammar, the genetic code, translation/complement functions.
- FASTQ_File: An easy-to-use FASTQ file iterator. Operates seemlessly on both gzipped and raw FASTQ files, and lets you iterate by sequences, quality scores, or both.
- FASTQ_File_Set: A convenient way to iterate line-by-line along multiple FASTQ files in parallel - e.g. for paired end reads.
- Sequence_Trie: A fast storage and query data structure for sequence information. Takes advantage of a trie structure to rapidly search for/iterate over one-off or two-off mutants.
- AminoAcid: A class for each Amino acid—useful for extracting physical properties.

2.7.1.8 Simulation

Currently, this is just some functions for generating random mutants and their counts based on a template. Useful for testing.

2.7.2 protfarm

Protfarm is a Python package designed for managing large amounts of parallel sequencing protein engineering datasets. It relies heavily on the functionality of the pepars package, and mostly just provides a database and API for storing, querying, and analyzing multiple protein engineering experiments, with multiple samples per experiment.

2.7.2.1 Prerequisites

Protfarm relies on the pepars package, so this must be installed before installing protfarm. Refer to the pepars repo for installation instructions.

2.7.2.2 Installation

You can then install protfarm via pip:

pip install git+https://github.com/GradinaruLab/protfarm.git

Note: use pip or pip3, whichever is associated with your Python3

2.7.2.3 GUI

If you prefer to work with a GUI instead of in Jupyter notebooks, Protfarm also has a graphical user interface, which is available as a separate package here: https://github.com/GradinaruLab/protfarm-gui.

2.7.2.4 Terminology

Before diving into the functionality, it is important to establish some terminology that will be used throughout the instructions and the package.

- **Data Path**: Protfarm expects all the data for all experiments to be contained within a single folder. This folder will largely be managed by Protfarm, so requires little user interaction (other than dropping raw FASTQ files in, or getting exported data out). It is recommended to create a folder just for this purpose, and not manually put any data/analysis here. **There are no guarantees that Protfarm will not delete/manipulate data in this folder!**

- **Experiment**: Protfarm organizes data into "Experiments." An experiment can consist of any number of samples and rounds of data, but every sample in an experiment is expected to have

the same mutation strategy. For example, if you mutate two separate regions of a protein as part of two different libraries, all the data associated with these two regions should be bundled together into two separate experiments.

- **Sample/Library**: The terms sample and library are used interchangeably, and refer to a set of FASTQ files that should be bundled together for determining variant counts. Typically, a sample/library has a one-to-one correlation with a particular index in a sequencing run.

2.7.2.5 Functionality

The easiest way to see the functionality of Protfarm is to follow the examples in the following order:

1. workspace/workspace_initialization.ipynb: Set up a new workspace from scratch.
2. alignment/alignment.ipynb: Do an alignment of some FASTQ files against a template.
3. alignment/export_alignment.ipnyb: Export the variant counts to a CSV file.
4. analysis/sequence_counts.ipynb: Investigate the sequence counts to see library diversity.
5. analysis/export_enrichment.ipynb: Calculate the enrichment of one sample over another, and export it.
6. analysis/amino_acid_heatmap.ipynb: Look at the distribution of amino acids, normalized by their intrinsic bias.
7. analysis/collapsing_sequences.ipynb: Collapse sequences that are likely to be PCR errors, and compare the library diversity.

2.7 Multiplexed Cre-dependent Selection (M-CREATE) yields systemic AAVs for targeting distinct brain cell types

Adapted from:

Ravindra Kumar, S., Miles, T. F., Chen, X., **Brown, D.**, Dobreva, T., Huang, Q., Ding, X., Luo, Y., Einarsson, P. H., Greenbaum, A., Jang, M. J., Deverman, B. E., & Gradinaru, V. (2020). Multiplexed

Cre-dependent selection yields systemic AAVs for targeting distinct brain cell types. Nature Methods, 17(5), 541–550. https://doi.org/10.1038/s41592-020-0799-7

2.7.1 Summary

Recombinant adeno-associated viruses (rAAVs) are efficient gene delivery vectors via intravenous delivery; however, natural serotypes display a finite set of tropisms. To expand their utility, we evolved AAV capsids to efficiently transduce specific cell types in adult mouse brains. Building upon our Cre-recombination-based AAV targeted evolution (CREATE) platform, we developed Multiplexed-CREATE (M-CREATE) to identify variants of interest in a given selection landscape through multiple positive and negative selection criteria. M-CREATE incorporates next-generation sequencing, synthetic library generation, and a dedicated analysis pipeline. We have identified capsid variants that can transduce the central nervous system broadly, exhibit bias toward vascular cells and astrocytes, target neurons with greater specificity, or cross the blood–brain barrier across diverse murine strains. Collectively, the M-CREATE methodology accelerates the discovery of capsids for use in neuroscience and gene-therapy applications.

2.7.2 Introduction

Recombinant adeno-associated viruses (rAAVs) are widely used as gene delivery vectors in scientific research and therapeutic applications due to their ability to transduce both dividing and non-dividing cells, their long-term persistence as episomal DNA in infected cells, and their low immunogenicity (Daya and Berns, 2008; Deverman et al., 2018; Gaj et al., 2016; Naso et al., 2017; Wu et al., 2006). However, gene delivery by natural AAV serotypes is limited by dose-limiting safety constraints and largely overlapping tropisms. AAV capsids engineered by rational design (Bartlett et al., 1999; Davidsson et al., 2019; Lee et al., 2018; Sen, 2014) or directed evolution (Bedbrook et al., 2018; Dalkara et al., 2013; Excoffon et al., 2009; Grimm et al., 2008; Kotterman and Schaffer, 2014; Maheshri et al., 2006; Müller et al., 2003; Ogden et al., 2019; Pekrun et al., 2019; Pulicherla et al., 2011; Ying et al., 2010)have yielded vectors with improved efficiencies for select cell populations (Chan et al., 2017; Davis et al., 2015; Deverman et al., 2016; Körbelin et al.,

2016a; Ojala et al., 2018a; Tervo et al., 2016; Tordo et al., 2018), yet much work remains to identify a complete toolbox of efficient and specific vectors. Previously, we evolved the AAV-PHP.B and AAV-PHP.eB variants from AAV9 using a selection method called CREATE (Deverman et al., 2016). This method applies positive selective pressure for capsids capable of infecting a target cell population by pairing a viral genome containing lox sites with *in vivo* selection in transgenic mice expressing Cre in the cell type of interest. This combination allows a Cre–Lox recombination-dependent PCR amplification of only those capsids which successfully deliver their genomes to the nuclei of the target cell type.

To more efficiently expand the AAV toolbox, we developed Multiplexed-CREATE (M-CREATE) (Figure 9), which compares the enrichment profiles of thousands of capsid variants across multiple cell types and organs within a single experiment. This method improves upon its predecessor by capturing the breadth of capsid variants at every stage of the selection process. M-CREATE supports: (1) the calculation of an enrichment score for each variant by using next-generation sequencing (NGS) to correct for biases in viral production prior to selection, (2) reduced propagation of bias in successive rounds of selection through the creation of a post-round one synthetic pool library with equal variant representation, and (3) the reduction of false positives by including codon replicates of each selected variant in the pool. These improvements allow interpretation of variants' relative infection efficiencies across a broad range of enrichments in multiple positive selections and enable post-hoc negative screening by comparing capsid libraries recovered from multiple target cell types or organs. Collectively, these features allow prioritization of capsid variants for validation and characterization.

To demonstrate the ability of M-CREATE to reveal useful variants missed by its predecessor (CREATE), we used the capsid library design that yielded AAV-PHP.B, and identified several AAV9 variants with distinct tropisms including variants that have biased transduction of brain vascular cells or that can cross the blood–brain barrier (BBB) without mouse-strain specificity.

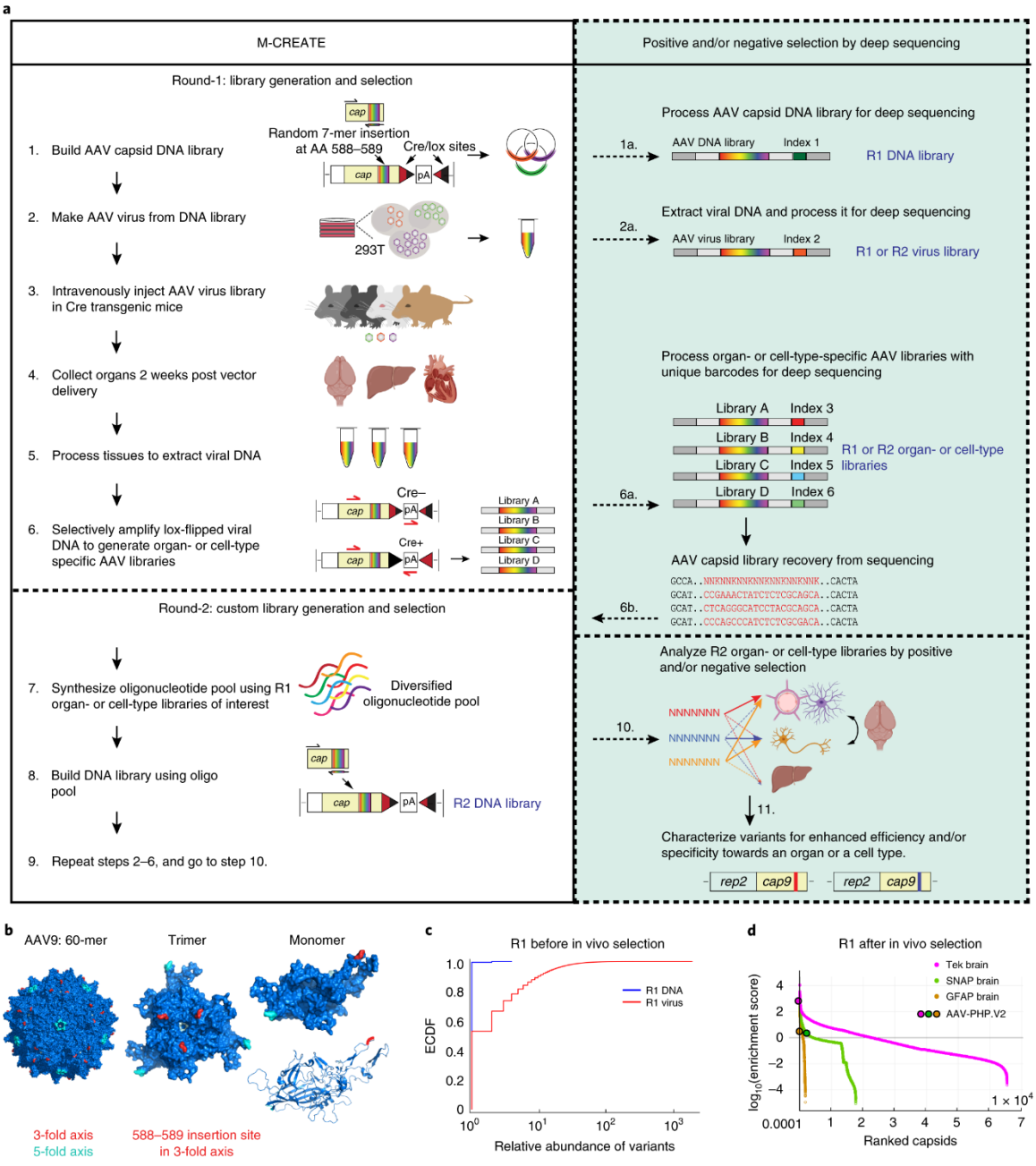


Figure 9 Workflow of M-CREATE and analysis of 7-mer-i selection in round-1. **(a)**, A multiplexed selection approach to identify capsids with specific and broad tropisms. Steps 1–6 describe the workflow in round-1 (R1) selection, steps 7–9 describe round-2 (R2) selection using the synthetic-pool method, steps 1a, 2a and 6a,b show the incorporation of deep sequencing to recover capsids after R1 and R2 selection, and steps 10–11 describe positive and/or negative selection criteria followed by variant characterization. The genes rep2 and cap9 in step 11 refers to rep from AAV2 and cap from AAV9, respectively, and the colored

bar within cap9 represents the targeted mutation. **(b)**, Structural model of the AAV9 capsid (PDB 3UX1) with the insertion site for the 7-mer-i library highlighted in red in the 60-meric (left), trimeric (middle) and monomeric (right) forms. **(c)**, Empirical cumulative distribution frequency (ECDF) of R1 DNA and virus libraries that were recovered by deep sequencing post Gibson assembly and virus production, respectively. **(d)**, Distributions of variants recovered from three R1 libraries from Tek-Cre, SNAP25-Cre and GFAP-Cre brain tissue (n = 2 per Cre line) are shown with capsid libraries, sorted by decreasing order of the enrichment score. The enrichment scores of the AAV-PHP.V2 variant are mapped as well.

2.7.3 Results

2.7.3.1 Analysis of capsid libraries during round-1 selection

M-CREATE was developed to enable the analysis of capsid variants' behavior within and across *in vivo* selections. By doing so, we aimed to identify capsids with diverse tropisms, as well as reveal the capsid sequence diversity within a given tropism. M-CREATE achieves these aims by incorporating NGS and a synthetic capsid library for round-2 *in vivo* selection along with a dedicated analysis pipeline to assign capsid enrichment values.

During DNA- and virus-library generation there is potential for biased accumulation and over-representation of certain capsid variants, obscuring their true enrichment during *in vivo* selection. These deviations may result from PCR amplification bias in the DNA library or sequence bias in the efficiency of virus production across various steps such as capsid assembly, genome packaging and purification. We investigated this with a *7-mer-i* (i for insertion) library, in which a randomized 7-amino acids (AA) library is inserted between AA 588 and 589 of AAV9 (Figure 9***a,b***) in the rAAV-ΔCap9-in-cis-Lox2 plasmid (theoretical library size, 3.4×10^{10} unique nucleotide sequences, and an estimated $\sim 1 \times 10^{8}$ nucleotide sequences upon transfection). We sequenced the libraries after DNA assembly and after virus purification to a depth of 10–20 million (M) reads, which was adequate to capture the bias among variants during virus production (Figure 9***c***). The DNA library had a uniform distribution of 9.6 M unique variants within ~10 M total reads (read count (RC) mean = 1.0, s.d. = 0.074), indicating minimal bias. In contrast, the virus library had 3.6 M unique variants within ~20 M depth (RC mean = 4.59, s.d. = 11.15) indicating enrichment of a subset of variants during viral

production. Thus, even permissive sites like 588–589 will impose biological constraints on sampled sequence space.

For *in vivo* selection, we intravenously administered the *7-mer-i* viral library in transgenic mice expressing Cre in astrocytes (GFAP-Cre), neurons (SNAP25-Cre) or endothelial cells (Tek-Cre) at a dose of 2×10^{11} vector genomes (vg) per adult mouse ($n = 2$ mice per Cre transgenic line). Two weeks after intravenous (i.v.) injection, we collected brain, spinal-cord and liver tissues. We extracted the rAAV genomes from tissues and selectively amplified the capsids that transduced Cre-expressing cells. Upon deep sequencing, we observed ~8×10^4 unique nucleotide variants in brain tissue samples (~48% of which were identified in the sequenced portion of the virus library) and <50 variants in spinal-cord samples across the transgenic lines, and each variant was represented with an enrichment score that reflects the change in relative abundance between the brain and the starting virus library (2.7.5 Methods and Figure 9***d***).

Two features of this dataset stand out. First, the variants recovered from brain tissue were disproportionately represented in the sequenced fraction of the viral library, demonstrating how production biases can skew selection results. Second, the distribution of capsid read counts reveals that more than half of the unique variants recovered after selection appear at low read counts. These variants may either have arisen spontaneously from errors during experimental manipulation or retain AAV9's basal levels of central nervous system (CNS) transduction.

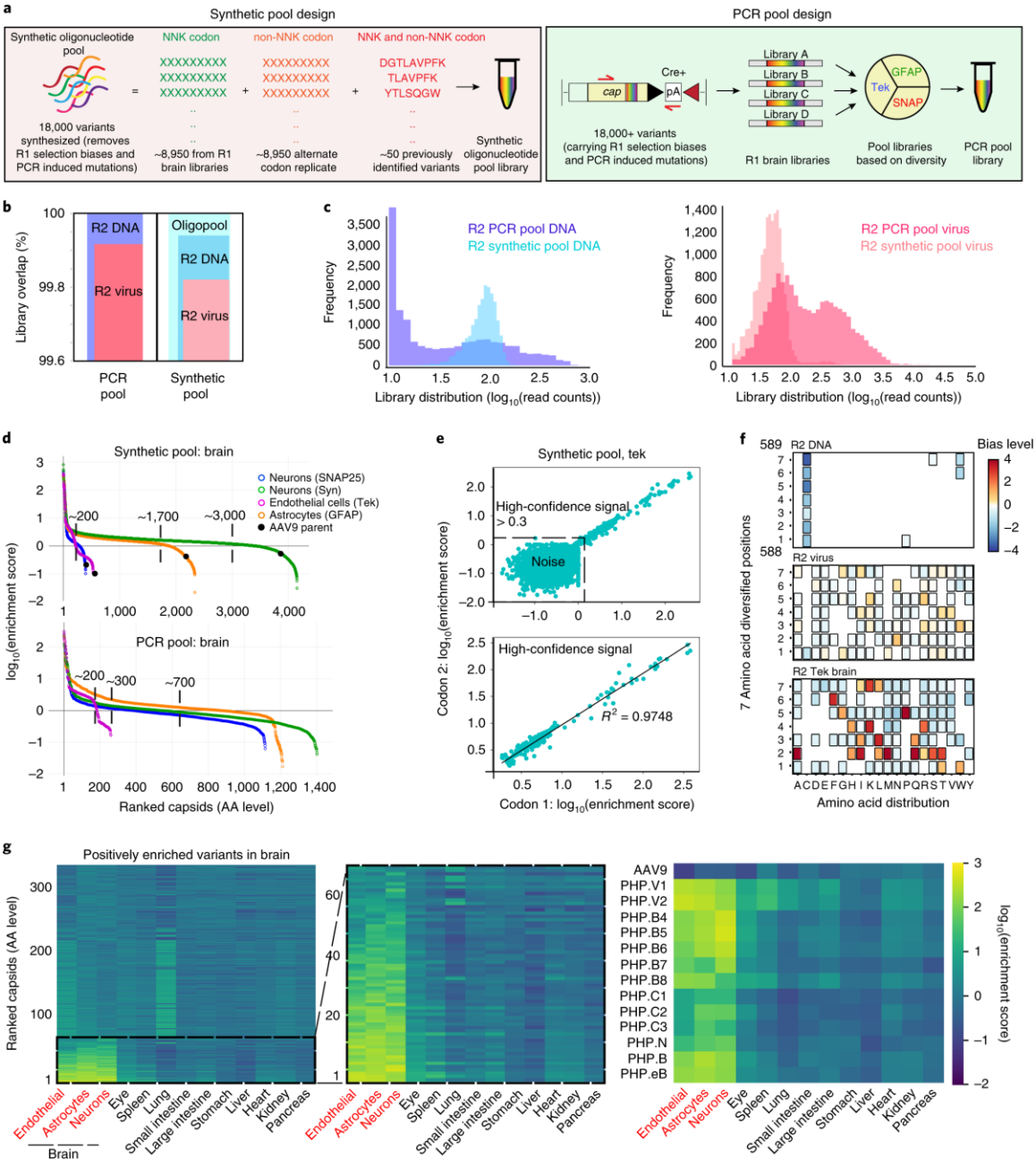


Figure 10 ***Round-2 capsid selections by synthetic pool and PCR pool methods.*** **(a)**, Schematic of R2 synthetic pool (left) and PCR pool (right) library design. **(b)**, Overlapping bar chart showing the percentage of library overlap between the mentioned libraries and their theoretical composition. **(c)**, Histograms of DNA and virus libraries from the two methods, where the variants in a library are binned by their read counts (in $\log_{10}$ scale) and the height of the histogram is proportional to their frequency. **(d)**, Distributions of R2 brain libraries from all Cre transgenic lines ($n = 2$ mice per Cre Line, mean is plotted) and both methods, in which the libraries are sorted in decreasing order of enrichment score ($\log_{10}$ scale). The total number of positively enriched variants from these libraries

are highlighted by dotted straight lines and AAV9's relative enrichment is mapped on the synthetic pool plot. **(e)**, Comparison of the enrichment scores (log_{10} scale) of two alternate codon replicates for 8,462 variants from the Tek-Cre brain library ($n = 2$ mice, mean is plotted). The broken line separates the high-confidence signal (>0.3) from noise. For the high-confidence signal (below), a linear least-squares regression is determined between the two codons and the regression line (best fit). The coefficient of determination r^2 is shown. **(f)**, Heat maps representing the magnitude (log_2(fold change)) of a given amino acid's relative enrichment or depletion at each position given statistical significance is reached (boxed if $P \leq 0.0001$, two-sided, two-proportion *z*-test, *P* values corrected for multiple comparisons using Bonferroni correction). R2 DNA normalized to oligopool (top, ~9,000 sequences), R2 virus normalized to R2 DNA (middle, n = ~9,000 sequences), R2 Tek brain library with enrichment over 0.3 (high-confidence signal) from synthetic pool method normalized to R2 virus (bottom, 154 sequences) are shown ($n = 2$ for brain library, one per mouse; all other libraries, $n = 1$). **(g)**, Heat map of Cre-independent relative enrichment across organs ($n = 2$ mice per Cre line, mean across 6 samples from 3 Cre lines is plotted) for variants enriched in the brain tissue of at least one Cre-dependent synthetic pool selection (red text, $n = 2$ mice per cell-type, mean is plotted) (left). Zoom-in of the most CNS-enriched variants (middle), and of the variants that are characterized in the current study along with spike-in library controls (right) are shown.

2.7.3.2 Analysis of capsid libraries after round-2 selection

Whereas the amino acid distribution of the DNA library closely matched the Oligopool design, virus production selected for a motif within the hepta AA diversified insertion (between AA 588 and AA 589), with Asn at position 2, β-branched amino acids (I, T, V) at position 4 and positively charged amino acids (K, R) at position 5 (Figure 10***f***. Fitness for BBB crossing resulted in a different pattern. For instance, variants highly enriched after recovery from brain tissue (across all Cre lines) shared preferences for Pro in position 5, and Phe in position 6.

By assessing enrichment score reproducibility within the synthetic pool design, we next determined the brain enriched variants' distribution across peripheral organs (Figure 10***g***, left). About 60 variants that are highly enriched in brain are comparatively depleted across other organs (Figure 10***g***, middle). Encouraged by the expected behavior of spike-in control variants (AAV9, PHP.B, PHP.eB), we chose eleven additional variants for further validation (Figure 10***g***, right), including several that would have been overlooked if the choice had been based on PCR pool or CREATE (**Table 1**).

AAV Variants	***Synthetic pool* enrichment rank**	***PCR pool* enrichment rank**	***PCR pool* read count rank**
PHP.V1	1	4	3

PHP.V2	2	1	1
PHP.B4	4	10	56
PHP.B7	6	13	36
PHP.B8	3	7	23
PHP.C1	13	34	74
PHP.C2	12	20	293
PHP.C3	16	Not recovered	Not recovered

Table 1. ***Ranking of AAV-PHP capsids across methods.*** Ranks of selected variants among all capsids recovered from R2 Tek-Cre selection by *synthetic pool* enrichment score (representing M-CREATE), *PCR pool* enrichment score (representing closer to M-CREATE), or *PCR pool* read counts (representing CREATE), the highest ranks of which starts from 1, and "Not recovered" represent absence of the variant from R2 sequencing data.

We chose these variants due to their enrichments and where they fall in sequence space. We noticed that the enriched variants cluster into distinct families based on sequence similarity. The most enriched variants form a distinct family across selections with a common motif: T in position 1, L in position 2, P in positive 5, F in position 6 and K or L in position 7 (Figure 11***a***). This amino acid pattern closely resembles the TLAVPFK motif in the previously identified variant AAV-PHP.B (Deverman et al., 2016). Given the sequence similarity among members of this family, we next tested whether selected variants can cross the BBB and target the CNS with similar efficiency and tropism.

2.7.3.3 The dominance of PHP.B-like motif

The ability to twice recover the AAV-PHP.B sequence family from completely independently constructed and selected libraries confirms that the viral library's sequence space coverage was broad enough to recover a family of variants sharing a common motif. Unlike CREATE which identified only one variant, AAV-PHP.B, M-CREATE yielded a diverse PHP.B-like family that hints toward important chemical features of this motif. The sequence diversity within this family suggests that isolating AAV-PHP.B was not simply good fortune in our prior study (considering a theoretical starting library size of ~1.3 billion), and that this is a dominant family for this particular experiment.

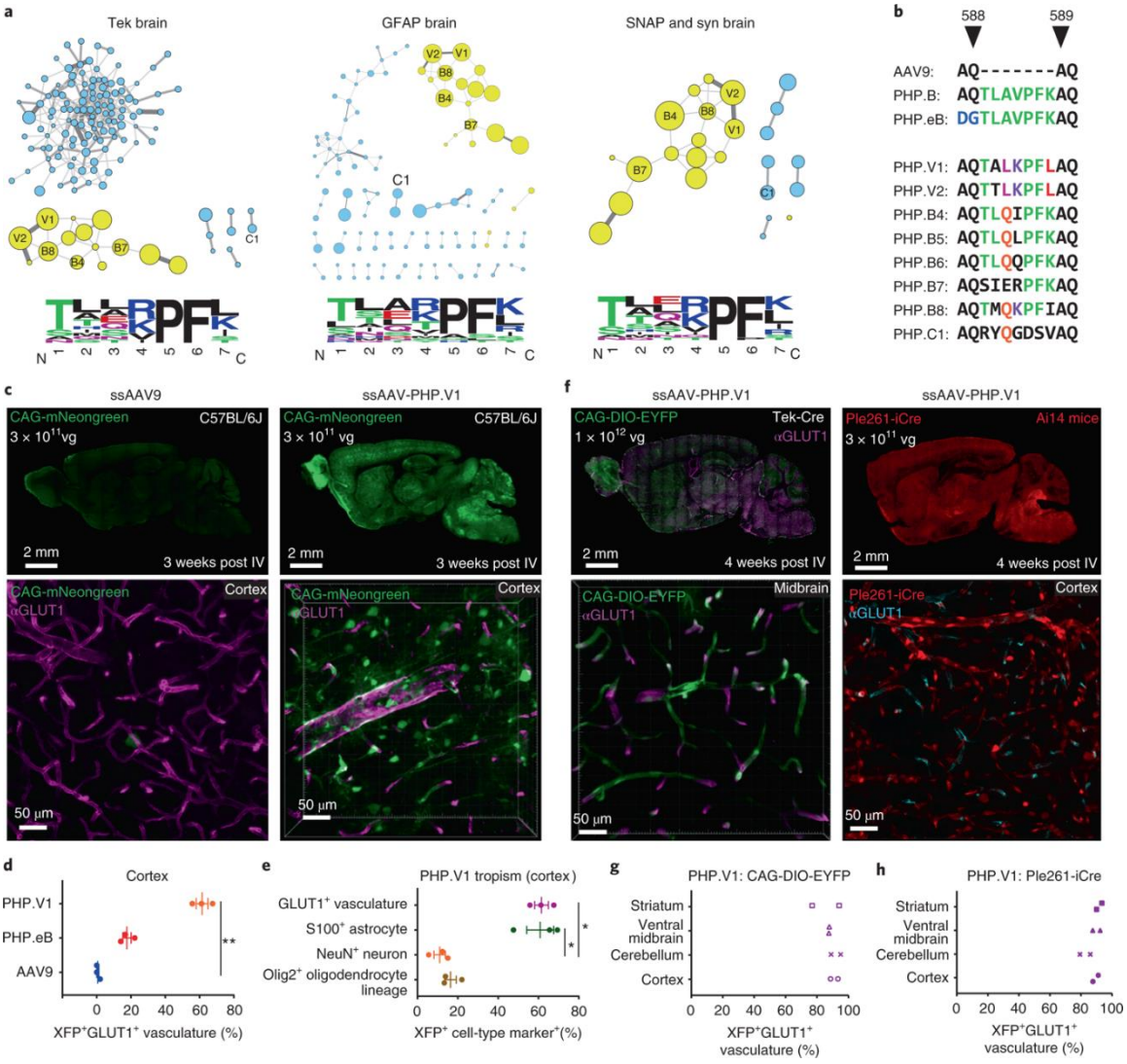


Figure 11. Selected AAV capsids form sequence families and include variants for brain-wide transduction of vasculature. **(a)**, Clustering analysis of variants from synthetic pool brain libraries after enrichment in Tek-Cre (left), GFAP-Cre (middle) and combined SNAP-Cre and Syn-Cre (right) selections. The size of the nodes represents relative enrichment in the brain. Thickness of the edges (connecting lines) represents the degree of relatedness. Distinct families (yellow) with the corresponding AA frequency logos (AA size represents prevalence and color encodes AA properties) are shown. **(b)**, The hepta AA insertion peptide sequences of AAV-PHP variants between AA positions 588–589 of AAV9 capsid are shown. AAs are colored by shared identity to AAV-PHP.B and eB (green) or among new variants (unique color per position). **(c)**, AAV9 (left) and AAV-PHP.V1 (right) mediated expression using ssAAV:CAG-mNeongreen genome (green, n = 3, 3 weeks of expression in C57BL/6J adult mice with 3 × 1011 vg i.v. dose per mouse, imaged under the same settings) in sagittal sections of brain (top) with higher-magnification image from cortex (bottom). Magenta, αGLUT1 antibody staining for vasculature. **(d)**, Percentage of vasculature stained with αGLUT1 that

overlaps with mNeongreen (XFP) expression in cortex. One-way analysis of variance (ANOVA) non-parametric Kruskal–Wallis test (P = 0.0036), and follow-up multiple comparisons using uncorrected Dunn's test (P = 0.0070 for AAV9 versus PHP.V1) are reported. **P ≤ 0.01 is shown, P > 0.05 is not shown; data are mean ± s.e.m., n = 3 mice per AAV variant, cells quantified from 2–4 images per mouse per cell type. **(e)**, Percentage of cells stained with each cell-type specific marker (αGLUT1, αS100 for astrocytes, αNeuN for neurons, and αOlig2 for oligodendrocyte lineage cells) that overlaps with mNeongreen (XFP) expression in cortex. Kruskal–Wallis test (P = 0.0078), and uncorrected Dunn's test (P = 0.0235 for neuron versus vascular cells, and 0.0174 for neuron versus astrocyte) are reported. *P ≤ 0.05 is shown, and P > 0.05 is not shown; data are mean ± s.e.m., n = 3 mice, cells quantified from 2–4 images per mouse per cell type. **(f)**, Vascular transduction by ssAAV-PHP.V1:CAG-DIO-EYFP in Tek-Cre adult mice (left) (n = 2, 4 weeks of expression, 1 × 1012 vg i.v. dose per mouse), and by ssAAV-PHP.V1:Ple261-iCre in Ai14 reporter mice (right) (n = 2, 3 weeks of expression, 3 × 1011 vg i.v. dose per mouse). Tissues are stained with αGLUT1 (magenta (left) and cyan (right)). **(g)**, Efficiency of vascular transduction (as described in (d)) in Tek-Cre mice (n = 2, mean from 3 images per mouse per brain region). **(h)**, Efficiency of vascular transduction in Ai14 mice (n = 2, a mean from 4 images per mouse per brain region).

2.7.3.4 AAV9 variants with enhanced BBB entry and CNS transduction

Given the dominance of the PHP.B family in the R2 selection, we characterized its most enriched member, harboring a TALKPFL motif and henceforth referred to as AAV-PHP.V1 (Figure 11***a,b***). Despite its sequence similarity to AAV.PHP.B, the tropism of AAV-PHP.V1 is biased toward transducing brain vascular cells (Figure 11***c***). When delivered intravenously, AAV-PHP.V1 carrying a fluorescent reporter under the control of the ubiquitous CAG promoter transduces ~60% of $GLUT1^+$ cortical brain vasculature, compared with ~20% with AAV-PHP.eB and almost no transduction with AAV9 (Figure 11***c,d***). In addition to the vasculature, AAV-PHP.V1 also transduced ~60% of cortical $S100^+$ astrocytes (Figure 11***e***).

We next investigated a series of variants selected to verify M-CREATE's predictive power outside this family. A highly enriched variant with an unrelated sequence, AAV-PHP.C1 harboring RYQGDSV (Figure 11***a,b*** and

Figure 12*a,b*), transduced astrocytes at a similar efficiency and neurons at lower efficiency compared to other tested variants from the B family (

Figure 12*b*).

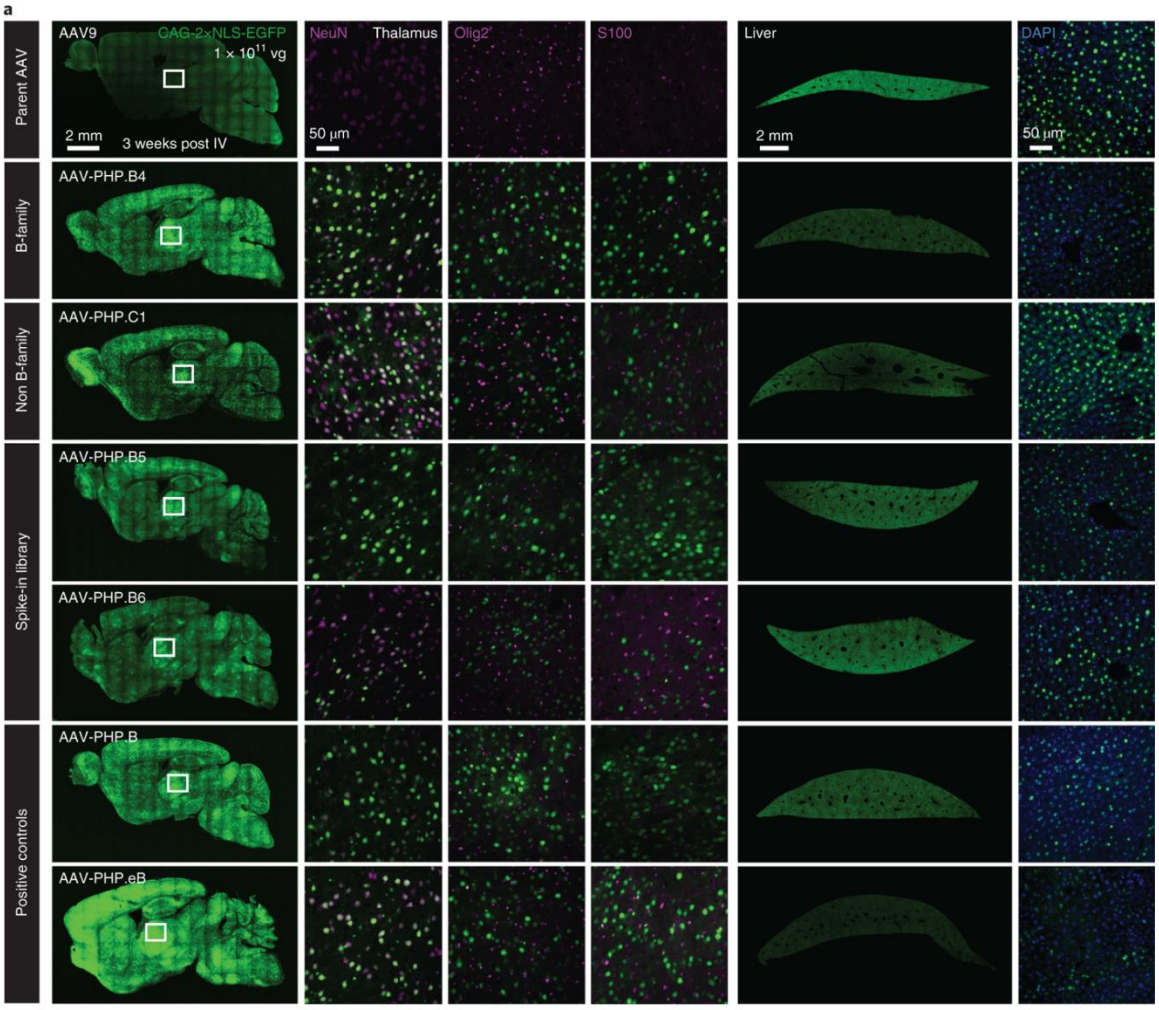

a
Parent AAV
B-family
Non B-family
Spike-in library
Positive controls
AAV9
CAG-2×NLS-EGFP
1 × 10^11 vg
2 mm
3 weeks post IV
NeuN
Thalamus
Olig2
S100
50 μm
Liver
DAPI
AAV-PHP.B4
AAV-PHP.C1
AAV-PHP.B5
AAV-PHP.B6
AAV-PHP.B
AAV-PHP.eB
b
Neurons
Astrocytes
Oligodendrocyte lineage cells
NeuN+ EGFP+ (%)
S100+ EGFP+ (%)
Olig2+ EGFP+ (%)
Cortex
Ventral midbrain
Striatum
Thalamus
PHP.eB
PHP.B
PHP.B4
PHP.B5
PHP.C1
PHP.B6
AAV9

Figure 12. Characterization of round-2 brain libraries and identification of capsids with broad CNS tropism. (a), Transduction by AAV-PHP.B4-B6 and C1 variants, as well as B, eB and AAV9 controls in sagittal brain and liver sections (each column was imaged under the same settings). White box, thalamus (this is not the precise region of the figures to the right). Vectors are packaged with ssAAV:CAG-2xNLS-EGFP genome ($n = 3$ per group, 1×10^{11} vg i.v. dose per adult C57BL/6J mouse, 3 weeks of expression). Tissues are stained with cell-type specific markers (magenta): αNeuN for neurons, αS100 for astrocytes and αOlig2 for oligodendrocyte lineage cells. Liver tissues are stained with DAPI (blue). **(b)**, The percentage of $\alpha NeuN^+$, $\alpha S100^+$ and $\alpha Olig2^+$ cells with detectable nuclear-localized EGFP in the indicated brain regions are shown ($n = 3$ per group, 1×10^{11} vg dose). A two-way ANOVA with correction for multiple comparisons using Tukey's test is reported with adjusted *P* values (****$P \leq 0.0001$, ***$P \leq 0.001$, **$P \leq 0.01$, *$P \leq 0.05$, is shown, and $P > 0.05$ is not shown on the plot; 95% confidence interval (CI), data are mean ± s.e.m. The dataset comprises a mean of two images per region per cell-type marker per mouse).

Collectively, our characterization of these AAV variants suggests several key points. First, within a diverse sequence family, there is room for both functional redundancy and the emergence of alternative tropisms. Second, highly enriched sequences outside the dominant family are also likely to possess enhanced function. Third, buoyed by codon replicate agreement in the synthetic pool, a variant's enrichment across tissues may be predictive. Fourth, while the synthetic pool R2 library contains a subset of the sequences that are in the PCR pool R2 and may thereby lack some enhanced variants, those variants found exclusively within the PCR pool library are more likely to be false positives.

The ability to confidently predict *in vivo* transduction from a pool of 18,000 nucleotide variants in R2 across multiple mice and Cre-lines is a substantial advance in the selection process and demonstrates the power of M-CREATE for the evolution of individual vectors.

2.7.3.4 An AAV9 variant that specifically transduces neurons

Using NGS, we re-investigated a *3-mer-s* (s for substitution) PHP.B library generated by the prior CREATE methodology and that yielded AAV-PHP.eB (Chan et al., 2017) (Figure 13***a***). Briefly, the re-investigated *3-mer-s* PHP.B library diversified positions 587-597 of the AAV-PHP.B capsid (equivalent of 587-590 AA on AAV9) in portions of three consecutive AAs, (~40,000 total variants) (Figure 13***a***). Selections were performed in three Cre-transgenic lines: Vglut2-IRES-Cre for glutamatergic neurons, Vgat-IRES-Cre for GABAergic neurons, and GFAP-Cre for astrocytes.

We deep sequenced the libraries recovered from brain (using Cre-dependent PCR) and a R2 library from the livers of wild-type mice (processed via PCR for all capsid sequences regardless of Cre-mediated inversion) and identified 150–200 capsids enriched in brain tissue (Figure 13***b***).

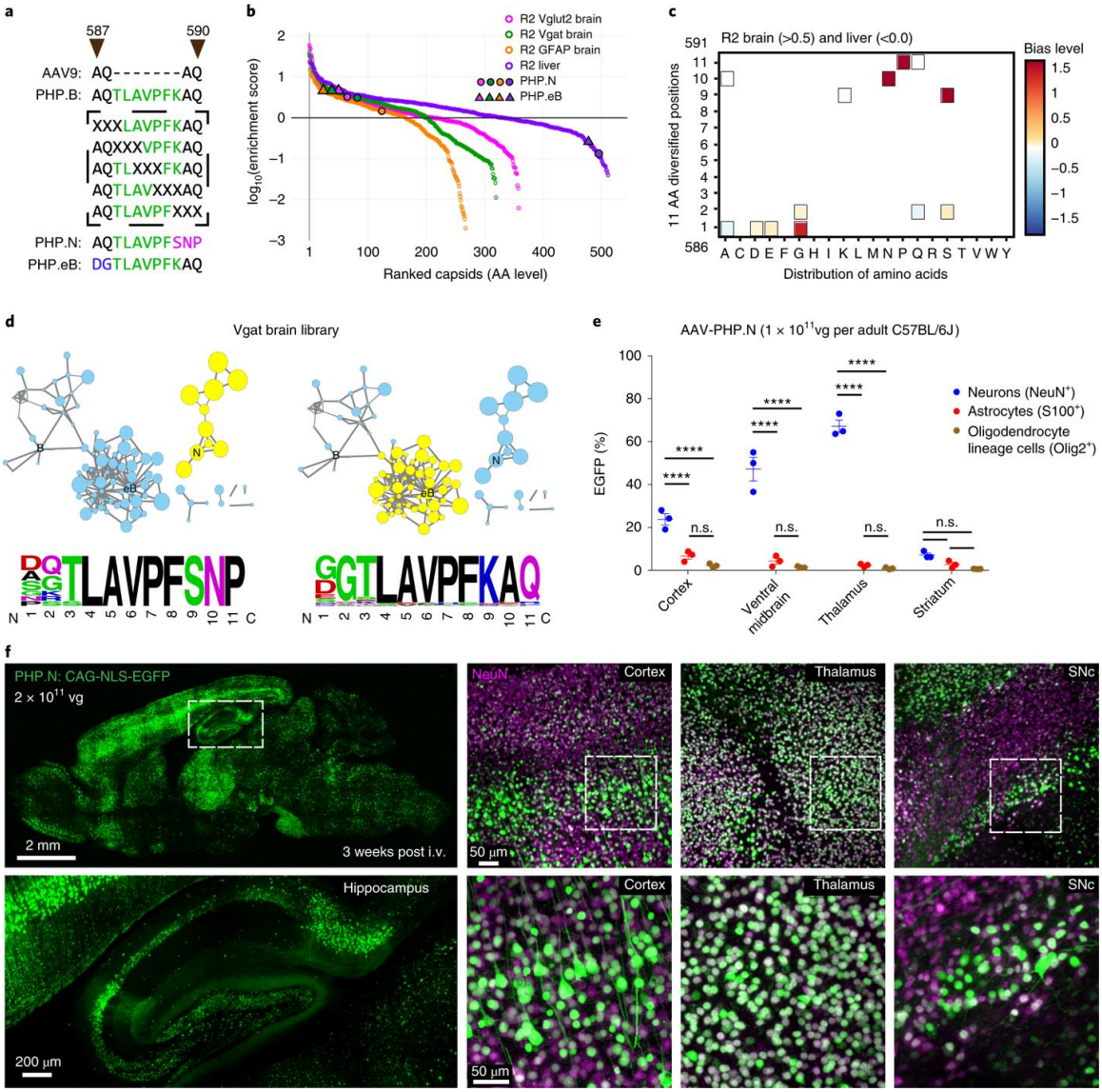


Figure 13. Recovery of AAV-PHP.B variants including one with high specificity for neurons.

(a), Design of the 3-mer-s PHP.B library with combinations of three AA diversification between AA 587–597 of AAV-PHP.B (corresponding to AA 587–590 of AAV9). Shared amion acid identity with the parent AAV-PHP.B (green) is shown along with unique motifs for AAV-PHP.N (pink) and AAV-PHP.eB (blue). **(b)**, Distributions of R2 brain and liver libraries (at the amino acid level) by enrichment score (normalized to R2 virus library, with variants sorted in decreasing order of enrichment score). The enrichment of AAV-PHP.eB and AAV-PHP.N across all libraries is mapped on the plot. **(c)**, Heat map represents the magnitude ($\log_2$(fold change)) of a given amino acid's relative enrichment or depletion at each position across the diversified region, only if statistical significance is reached on fold change (boxed if $P \leq 0.0001$, two-sided, two-proportion *z*-test, *P* corrected for multiple comparisons using Bonferroni correction). Plot includes variants that were highly enriched in brain (>0.5 mean enrichment score, where mean is drawn across Vglut2, Vgat and GFAP, $n = 1$ library per mouse line (sample pooled from 2 mice per line)) and underrepresented in liver (<0.0) (32 amino acid sequences). **(d)**, Clustering analysis of enriched variants from Vgat brain library is shown. Node size represents the degree of depletion in liver. Thickness of edges (connecting lines) represents degree of relatedness between nodes. Two distinct families are highlighted in yellow and their corresponding amino acid frequency logos are shown below (amino acid size represents prevalence, and color encodes amino acid properties). **(e)**, The percentage of neurons, astrocytes and oligodendrocyte lineage cells with ssAAV-PHP.N:CAG-2xNLS-EGFP in the indicated brain regions is shown ($n = 3$, 1×10^{11} vg i.v. dose per adult C57BL/6J mouse, 3 weeks of expression, data is mean ± s.e.m., 6–8 images for cortex, thalamus and striatum, and 2 images for ventral midbrain, per mouse per cell-type marker using ×20 objective covering the entire regions). A two-way ANOVA with correction for multiple comparisons using Tukey's test gave adjusted *P* values reported as ****$P \leq 0.0001$, n.s. for $P > 0.05$, 95% CI. **(f)**, Transduction by ssAAV-PHP.N:CAG-NLS-EGFP ($n = 2$, 2×10^{11} vg i.v. dose per adult C57BL/6J mouse, 3 weeks of expression) is shown with NeuN staining (magenta) across three brain areas (cortex, SNc (substantia nigra pars compacta) and thalamus).

Variants that were enriched in brain and underrepresented in liver show a significant bias towards certain amino acids such as G, D and E at position 1; G and S at position 2 (which includes the AAV-PHP.eB motif, DG); and S, N and P at position 9, 10 and 11 (Figure 13***c*** and

Supplementary Figure 1*c*; $P \leq 0.0001$, two-sided, two-proportion *z*-test, *P* values were corrected for multiple comparisons using Bonferroni correction). We clustered variants that were enriched in the brain according to their sequence similarities and ranked them by their underrepresentation in liver (represented by node size in clusters). A distinct family referred to as N emerged with the common motif SNP at positions 9–11 in the PHP.B backbone (Figure 13*d* and

Supplementary Figure 1*d*).

The core variant of the N-family cluster, with the AQTLAVPFSNP motif, was highly abundant in R1 and R2 selections, had higher enrichment score in Vglut2 and Vgat brain tissues compared to GFAP, and was underrepresented in liver tissue (Figure 13***b*** and

Supplementary Figure 1*a*–*d*). Unlike AAV-PHP.eB, this variant (AAV-PHP.N) specifically transduced $NeuN^+$ neurons even when packaged with a ubiquitous CAG promoter, although the transduction efficiency varied across brain regions (from ~10–70% in $NeuN^+$ neurons, including both $VGLUT1^+$ excitatory and $GAD1^+$ inhibitory neurons; Figure 13***e,f*** and **Supplementary Figure 1*e,f***).

Thus, by re-examining the *3-mer-s* library we identified several useful variants, including one with notable cell-type-specific tropism. While Vglut2-Cre and Vgat-Cre mice were used for *in* vivo selection, we didn't find variants that stood out for neuronal subtype-specific transduction of excitatory and inhibitory populations from our initial investigations on the NGS dataset. It is possible that a biological solution to this (stringent) selection was not present in this library.

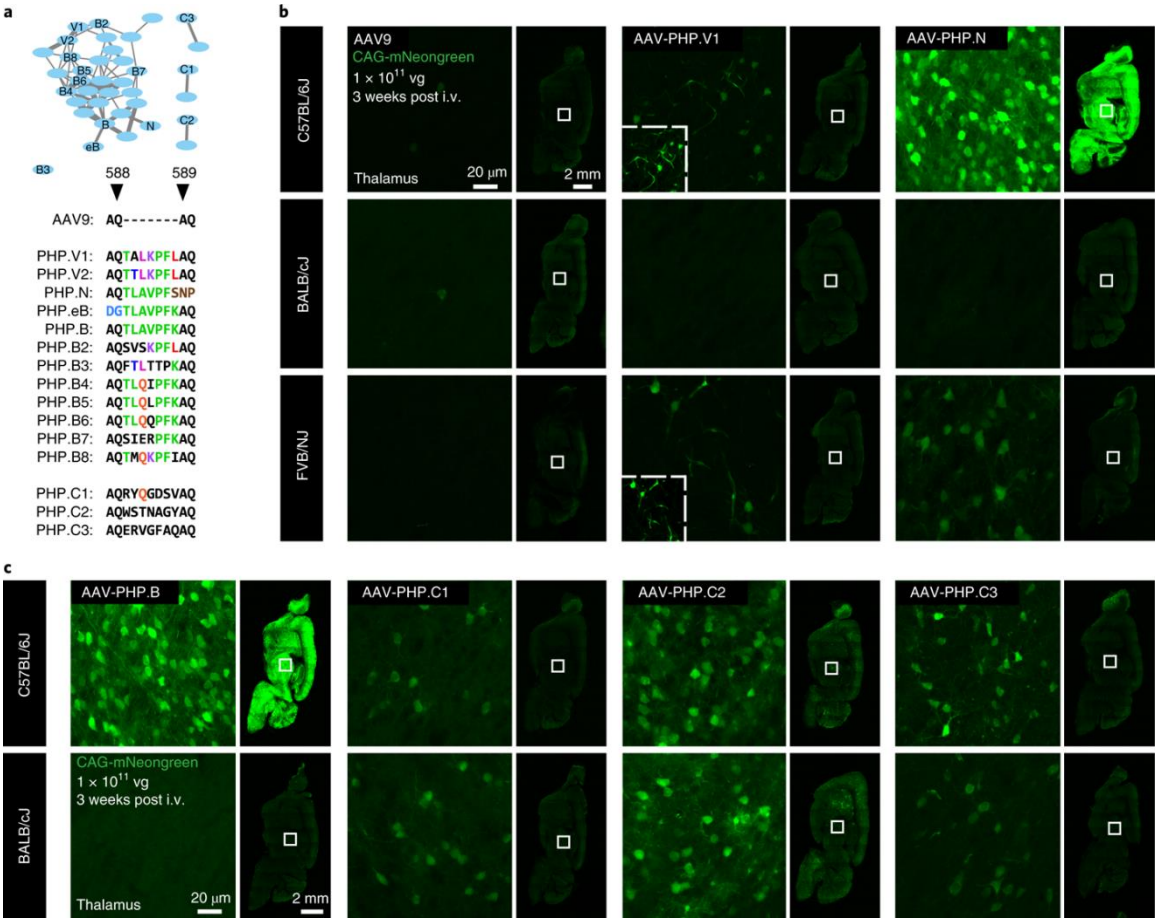


Figure 14. Tropism of variants from distinct families across mouse strains. (a), Clustering analysis showing the brain-enriched sequence families of variants identified in prior studies (PHP.B-B3, PHP.eB) or in the current study (PHP.B4-B8, PHP.V1-2, PHP.C1-3). Thickness of edges (connecting lines) represents degree of relatedness between nodes. The amino acid sequences inserted between 588–589 (of AAV9 capsid) for all the variants discussed are shown below. **(b)**, Transduction of AAV9, AAV-PHP.V1 and AAV-PHP.N across the mouse strains C57BL/6J, BALB/cJ and FVB/NJ are shown in sagittal brain sections (right), along with a higher-magnification image of the thalamus brain region (left). **(c)**, Transduction by AAV-PHP.B, AAV-PHP.C1-C3 in C57BL/6J and BALB/cJ mice are shown in sagittal brain sections (right), along with a higher-magnification image of the thalamus brain region (left). **(b**,**c)**, White box, thalamus (this is not the precise area that is zoomed-in on the figure to the left). All sagittal sections and thalamus regions were acquired under same image settings. The insets in AAV-PHP.V1 are zoom-ins with enhanced brightness. The indicated capsids were used to package ssAAV:CAG-mNeongreen ($n = 2$–3 per group, 1×10^{11} vg i.v. dose per 6- to 8-week-old adult mouse, 3 weeks of expression. The data reported in **(b)** and **(c)** are from one experiment where all viruses were freshly prepared and titered in the same assay for dosage consistency. AAV-PHP.C2 and AAV-PHP.C3 were further validated in an independent experiment for BALB/cJ, $n = 2$ per group).

2.7.3.5 Investigation of capsid families beyond the C57BL/6J mouse strain

The enhanced CNS tropism of AAV-PHP.eB and AAV-PHP.B relative to AAV9 is absent in a subset of mouse strains. Their CNS transduction is highly efficient in C57BL/6J, FVB/NCrl, DBA/2 and SJL/J, with intermediate enhancement in 129S1/SvimJ, and no apparent enhancement over AAV9 in BALB/cJ and several additional strains (Batista et al., 2020; Challis et al., 2019; Hordeaux et al., 2018, 2019; Huang et al., 2019; Matsuzaki et al., 2019). This pattern holds for the two variants from the PHP.B family that we characterized further, AAV-PHP.V1 and AAV-PHP.N (Figure 14***a***). These variants did not transduce the CNS in BALB/cJ, yet transduced the FVB/NJ strain (Figure 14***b***).

Notably, M-CREATE revealed many non-PHP.B-like sequence families that enriched through selection for transduction of cells in the CNS. We tested the previously mentioned AAV-PHP.C1 (RYQGDSV), as well as AAV-PHP.C2 (WSTNAGY), and AAV-PHP.C3 (ERVGFAQ) (Figure 14***a***). These showed enhanced BBB crossing irrespective of mouse strain, with roughly equal CNS transduction in BALB/cJ and C57BL/6J (Figure 14***c***). Collectively, these studies suggest that M-CREATE is capable of finding capsid variants with diverse mechanisms of BBB entry that do not exhibit strain specificity.

2.7.4 Discussion

This work outlines the development and validation of the M-CREATE platform for multiplexed viral capsid selection. M-CREATE incorporates multiple internal controls to monitor sequence progression, minimize bias and accelerate the discovery of capsid variants with useful tropisms. Utilizing M-CREATE, we have identified both individual capsids and distinct families of capsids that are biased toward different cell-types of the adult brain when delivered intravenously. The outcome from 7-mer-i selection demonstrates the possibility of finding AAV capsids with improved efficiency and specificity towards one or more cell types. Patterns of CNS infectivity across mouse strains suggest that M-CREATE may also identify capsids with distinct mechanisms of BBB crossing. With additional rounds of evolution as shown in the *3-mer-s* selection, the specificity or

efficiency of 7-mer-i library variants may be improved, as was observed with AAV-PHP.N or AAV-PHP.eB (Chan et al., 2017).

We believe that the variants tested *in vivo* and their families will find broad application in neuroscience, including studies involving the BBB (Sweeney et al., 2019), neural circuits (Betley and Sternson, 2011), neuropathologies (Sweeney et al., 2018), and therapeutics (Lykken et al., 2018). AAV-PHP.V1 or AAV-PHP.N are well-suited for studies requiring gene delivery for optogenetic or chemogenetic manipulations (Vlasov et al., 2018), or in rare monogenic disorders (targeting brain endothelial cells, for example GLUT1-deficiency syndrome, NLS1-microcephaly (Sweeney et al., 2018), or targeting neurons, for example mucopolysaccharidosis type IIIC (Tordo et al., 2018)).

The outcomes from our experiments employing M-CREATE opens several promising lines of inquiry, such as the assessment of identified capsid families across species, the investigation of the mechanistic properties that underlie the ability to cross specific barriers (such as the BBB) or target specific cell populations and further evolution of the identified variants for improved efficiency and specificity. In addition, the datasets generated by M-CREATE could be used as training sets for in silico selection by machine-learning models. M-CREATE is presently limited by the low throughput of vector characterization *in vivo*; however, RNA-sequencing technologies (Hwang et al., 2018) offer hope in this regard. In summary, M-CREATE will serve as a next-generation capsid-selection platform that can open directions in vector engineering and potentially broaden the AAV toolbox for various applications in science and in therapeutics.

2.7.5 Methods

2.7.5.1 NGS data alignment and processing

The raw fastq files from NGS runs were processed with custom-built scripts that align the data to AAV9 template DNA fragment containing the diversified region 7xNNK (for R1) or 11xNNN (for R2 since it was synthesized as 11xNNN).

The pipeline to process these datasets involved filtering the dataset to remove the low-quality reads by using the deep sequencing quality score for each sequence. The variant sequences were then recovered from the sequencing reads by searching for the flanking template sequences, and extracting the nucleotides of the diversified region (perfect string match algorithm). The quality of the aligned data was further investigated to remove any erroneous sequences (such as ones with stop codons). The raw data was plotted to study the quality of recovery across every library. Based on the RC distribution, we adapted a thresholding method to remove plausible erroneous mutants that may have resulted from PCR or NGS based errors. The assumption is that if there is a PCR mutation or NGS error on the recovered parent sequence, the parent must have existed at least one round earlier than the erroneous sequence, and thus a difference in RCs should exist.

For R1 tissue libraries, we observed a steep drop in the slope of the distribution curve following a long tail of low count sequences, and were found to be rich in sequences that are variations of the parents in the higher counts range. We manually setup a threshold for RCs to remove such erroneous mutants. The thresholded data were then processed differently based on the experimental needs as described elsewhere using custom Python based scripts.

For R2 tissue libraries from *PCR pool* and *synthetic pool,* given the smaller library size compared to R1, we thresholded the data in two steps. We only considered the tissue recovered sequences that were present in the respective input DNA and virus library (after removing lower count variants from input libraries following the same principle as R1 tissue libraries). This step partially removed the long tail of low count reads. As a second step, we applied the thresholding that was described for R1 tissue libraries.

While it is plausible that true variants may be lost during thresholding, this method minimized false positives as the low count mutants in tissue and virus libraries often seemed to have very high enrichment score (as RCs are normalized to input library). In other words, thresholding allowed selective investigation on enriched variants that had a higher-confidence in their NGS RCs.

As an alternative to our manual thresholding method, an optional error correction method called "Collapsing" was built to further validate the outcome from filtered datasets (see Section 2.5 Correcting PCR and sequencing errors). This method starts at the lowest count variants (variants of count 1) and searches for potential parent variants that are off by one nucleotide but have at least 2-fold higher counts (fold change = ($2^{\Delta CT}$) where CT is PCR cycle threshold). This error correction method then transfers the counts of these potential erroneous sequences to their originating sequences and repeats recursively until all sequences have been considered. On applying this error correction to our thresholded data, an additional ~0.002-0.03% of sequences were captured (compared to >19% captured by thresholding), confirming that our thresholding strategy was largely successful.

2.7.5.2 NGS data analysis

The aligned data were then further processed via a custom data-processing pipeline, with scripts written in Python.

The enrichment scores of variants (total, *N*) across different libraries were calculated from the read counts (RCs) according to the following formula:

Enrichment score = $\log_{10}$((variant 1 RC in tissue library1 / sum of variants N RC in library1) / (variant 1 RC in virus library / sum of variants N RC in virus library)).

To consistently represent library recovery between R1 and R2 selected variants, we estimated the enrichment score of the variants in R1 selection.

Since the DNA and virus libraries were not completely sampled unlike the tissue libraries, we assigned an estimated RC for variants that were not present in the input library but were present in the output library. For instance, R1 virus library is the input library to the R1 tissue libraries. The estimated RC is defined as a number that is lower than the lowest RC in the library with the assumption that these variants were found at a relatively lower abundance than the variants recovered

from the deep sequencing. In virus libraries, since RC of 1.0 was the lowest, we assigned all missing variants an estimated RC of 0.9. We use this method to calculate the enrichment score of the R1 tissue libraries which is normalized to R1 virus library (Figure 9***d***). This was done to represent libraries across two selection rounds consistently. Although, the individual enrichment score among R1 variants didn't add a significant value to the variants selected for R2 selection as described in the criteria to separate signal vs noise in R1 using the RCs.

The standard score of variants in a specific library was calculated using this formula:

Standard score = (read count_i – mean) / s.d.

Read count_i is raw copy number of a variant i. Mean is the mean of read counts of all variants across a specific library. The s.d. is the s.d. of read counts of all variants across a specific library.

The plots generated in this article were using the following software: Plotly, GraphPad PRISM 7.05, Matplotlib, Seaborn and Microsoft Excel 2016. The AAV9 capsid structure (PDB 3UX1)(DiMattia et al., 2012) was modeled in PyMOL.

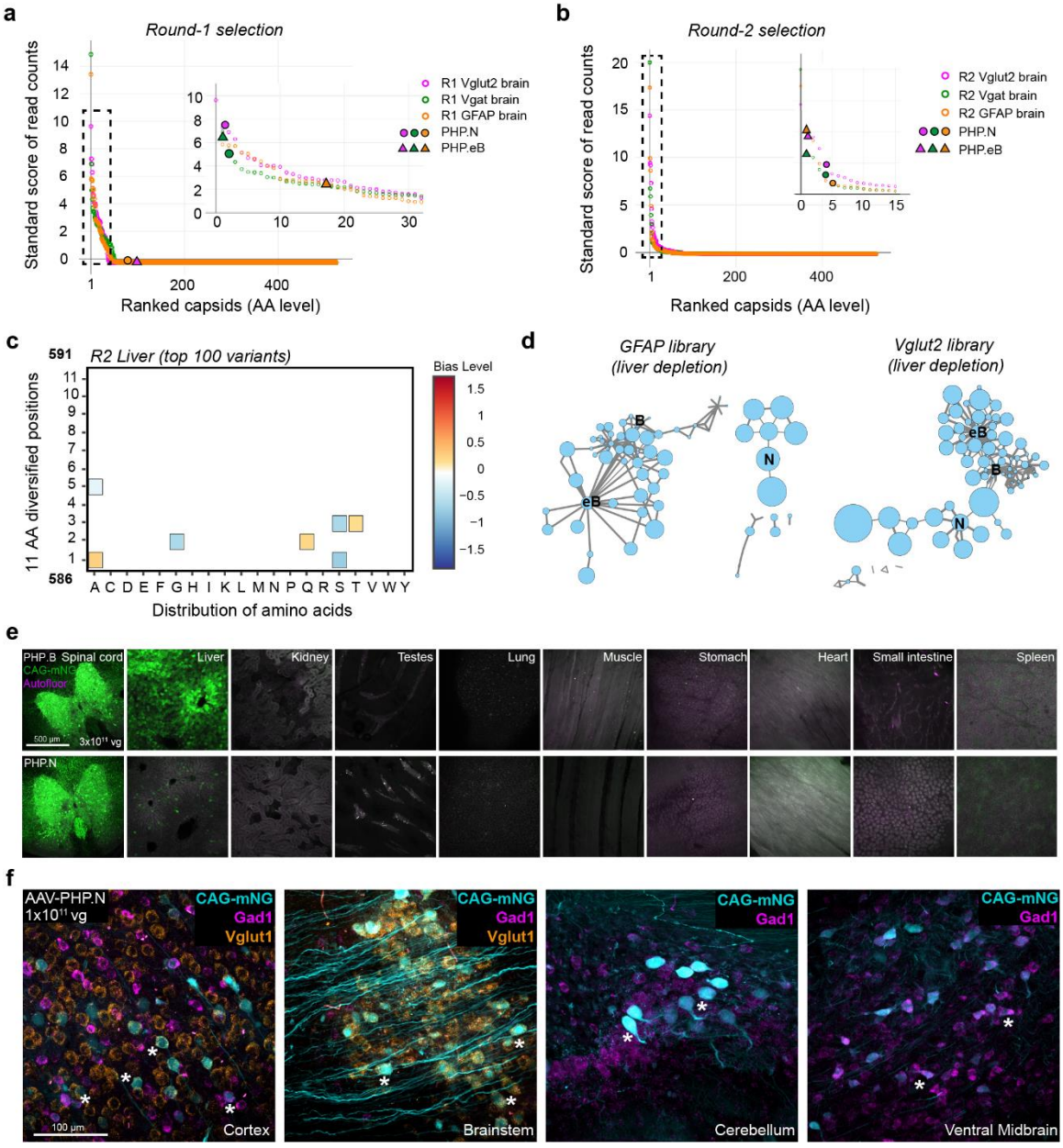
2.7.5.3 Heat map generation

The relative amino acid distributions of the diversified regions are plotted as heat maps. The plots were generated using the Python Plotly plotting library. The heat map values were generated from custom scripts written in Python, using functions in the custom "pepars" Python package (see 2.6 pepars: A Python package for manipulating NGS data).

Each heatmap uses both an expected (input) distribution of amino acid sequences and an output distribution. The output distribution must be a list of sequences and their count, and the input distribution can be either a list of sequences and their count, or an expected amino acid frequency from a template, such as NNK. For both input and output, the total count of amino acids in each position is tallied in accordance to each sequence's count and then divided by the total sum of counts, giving a frequency of each amino acid at each position. Then, the log2 fold change is calculated

between the output and the input. For amino acids with a count of 0 in either the input or output, no calculation is performed. In order to distinguish between statistically significant amino acid biases, a statistical test was performed using the statsmodels Python library. For the case where there are two amino acid counts, a two-sided, two-proportion z-test was performed; for comparing the output amino acid count to an expected input frequency from a template, a one-proportion z-test was performed. All p-values were then corrected for multiple comparisons using Bonferroni correction. Only bias differences below a significance threshold of 1e-4 are then outlined on the heatmap; all other (insignificant) squares are left empty.

2.7.6 Supplemental Figures



Supplementary Figure 1. Evolution Of The AAV-PHP.B Capsid By Diversifying Amino Acid Positions 587-597. (a), Distributions of R1 and **(b)**, R2 brain libraries (at AA level, SS of RCs sorted in decreasing order of scores) is shown. The SS for AAV-PHP.N and AAV-PHP.eB across libraries are mapped on the zoomed-in view of this plot (dotted line box). **(c)**, Heatmap of AA distributions across the diversified region of the enriched variants from R2 liver library (top 100 sequences) normalized to the R2 virus (input library). **(d)**, Clustering analysis of enriched variants from GFAP and Vglut2 brain libraries are shown with size of nodes representing their relative depletion in liver, and the thickness of edges (connecting lines) representing their relative identity between nodes. **(e)**, Expression of AAV-PHP.B (above) and AAV-PHP.N (below) packaged with ssAAV:CAG-mNeonGreen across all organs is shown (n = 3, $3x10^{11}$ vg i.v. dose per adult C57BL/6J mouse, 3 weeks of expression). The background auto fluorescence is in magenta. **(f)**, Transduction of mouse brain by the AAV-PHP.N variant, carrying the CAG promoter that drives the expression of mNeonGreen (n = 3, $1x10^{11}$ vg i.v. dose per C57BL/6J adult mouse, 3 weeks of expression) is shown. Fluorescence *in situ* hybridization chain reaction (FITC-HCR) was used to label excitatory neurons with Vglut1 and inhibitory neurons with Gad1. Few cells where EGFP expression co-localized with specific cell markers are highlighted by asterisks symbol.

2.8 Estimating M-CREATE selection pressure

Given the success of M-CREATE in discovering several novel AAV variants with enhanced transduction of specific cell types, we can return to the question of directed evolution vs. deep mutational scanning from the context of this data. Specifically, given that we have deep mutational scanning data, we can treat the read counts as proxies of what we would see in a traditional cloning experiment and ask, what is the probability that we would see a top k variant from within the deep mutational scanning data after picking a number of clones. Note that this is a slightly different formulation than in section 2.2, since we only have access to the deep mutational scanning data, not the ground truth. Therefore, instead of measuring the probability of improvement with a deep mutational scan, we are measuring the probability of missing out on a top variant from within the deep mutational scan data.

For example, in the case of PHP.V1 and PHP.V2, which were ranked 3 and 1, respectively, in terms of raw read counts in round 2 of selection, we can consider sampling from the read counts and simulating picking colonies and seeing the probability that we find PHP.V1, PHP.V2, or either (Figure 15). Despite a full 2^{nd} round of evolution, it is surprising to see that the probability of recovering these valuable variants from a lower throughput, read-count-based screen is quite low.

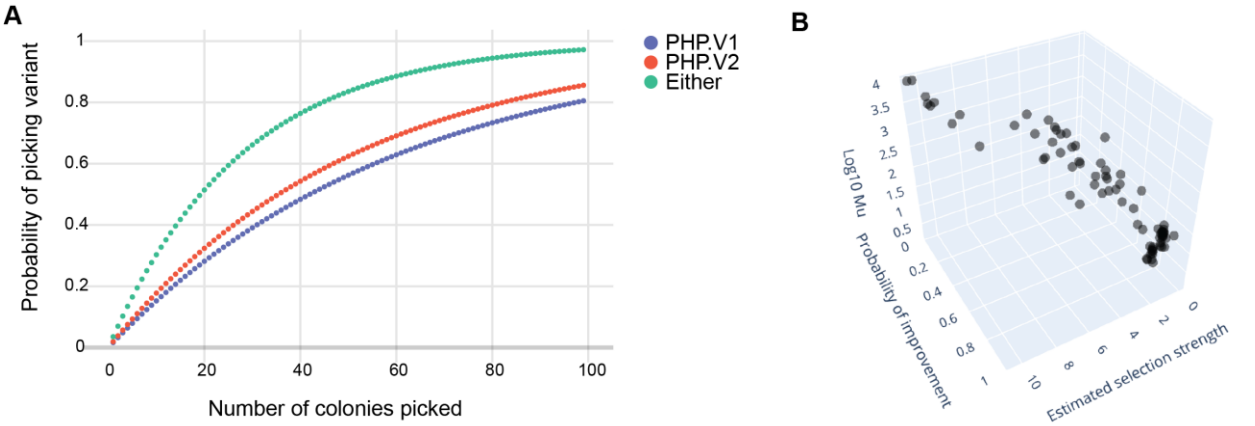


Figure 15. Exploration of selection pressure strength. (A) The probability PHP.V1 or PHP.V2 would have been found under a traditional low-throughput selection. **(B)** The parameters of fitted negative binomial models to variant read count data, and their correlation with probability of improvement of a 1e6 deep mutational scan over an n=10 low throughput screen.

Another thing that we can explore is the overdispersion of negative binomials in the raw read counts in different recovered libraries to see if overdispersion and library size are reasonable interpretations of a negative binomial of post-selection mutant counts. To do this, I fit a negative binomial using the package diffxpy (https://github.com/theislab/diffxpy) to the variant counts in all of our recovered libraries, and converted the learned parameters to the λ=M/N and estimated selective pressure strength, α, in accordance with the formulas in Section 2.2. Additionally, for each of these libraries, we can calculate an empirical estimate of the probability of improvement from a deep mutational scan by taking random samples from the data proportional to the read counts, with replacement, for both the n_1=10 colonies and n_2=1e6 sequencing depth case. We can then see what is the probability that each of these respective screens recovered one of the top k=3 variants. Similarly to the earlier simulations, lower estimated selection strength is correlated with a probability of improvement, as long as it is in a high library count (low M/N) regime (Figure 15B). The learned overdispersion parameters for all libraries ranged between 1 and 10. If we assume deep mutational scanning data is an adequate representation of the underlying distribution of variants, then we can use this range of overdispersion to make a lookup table of when deep mutational scanning will yield better results for a given library size, N and top k (Figure 16).

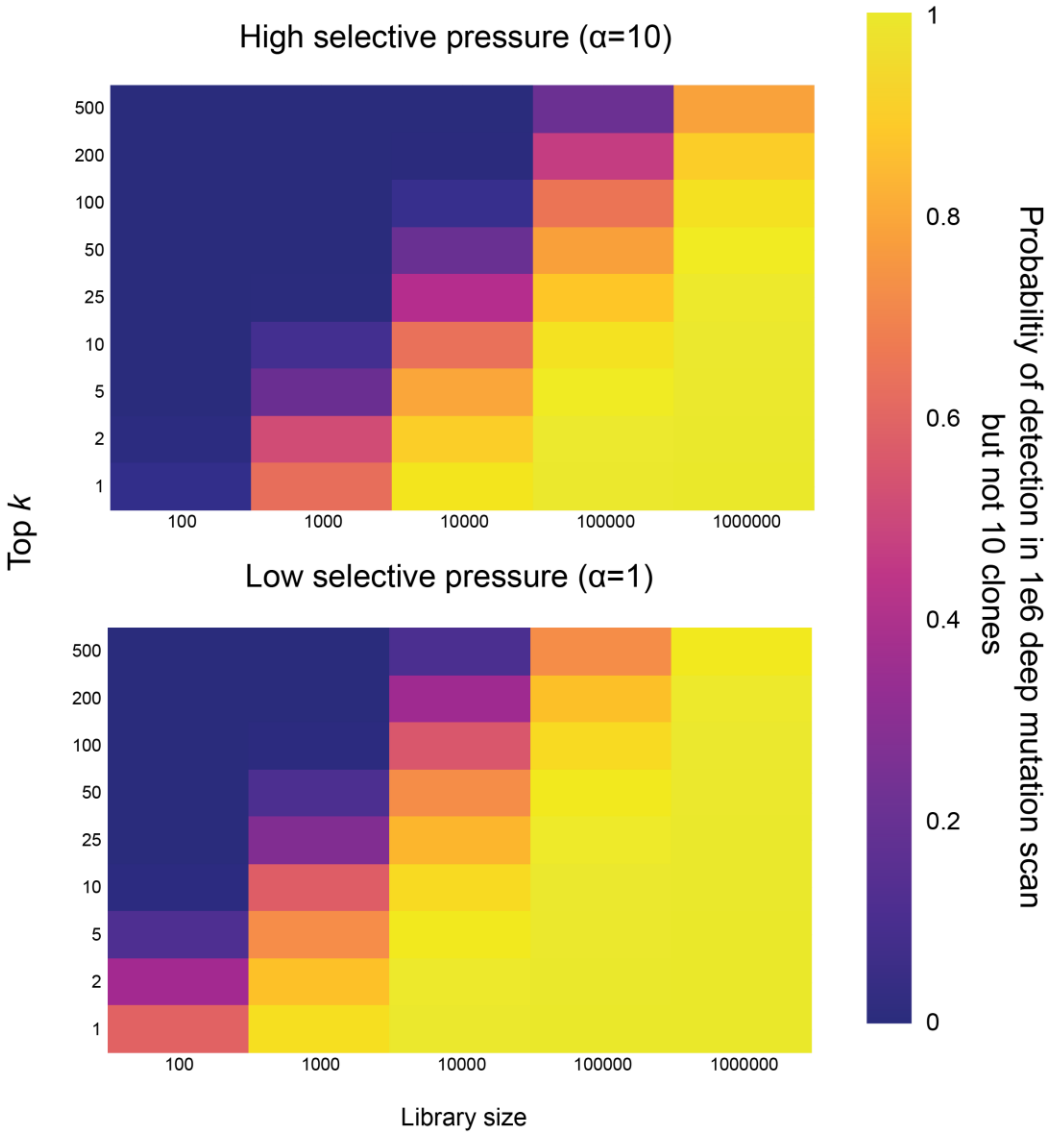


Figure 16. Relevant library sizes for deep mutational scanning based on AAV 588-589 mutation data. Lookup table of high (top) and low (bottom) selective pressure regimes to determine for what library sizes a deep mutational scan (i.e. 1e6 readout) will be more likely to yield a top *k* variant than a low throughput (1e1) screen.

Chapter 3

Tools for Single-cell Analysis

3.1 Summary

Single-cell RNA sequencing, although growing rapidly, is still a relatively new field, with many open questions, unsolved challenges, and opportunities for technological development. One area of active development are the data analysis methods, pipelines, and infrastructure for managing data across single-cell experiments. While many principles of scRNA-seq began as transformations of bulk RNA-seq workflows to a single-cell context, it quickly became evident that performing data analysis across scRNA-seq experiments is sufficiently distinct from the task of combining bulk RNA-seq experiments such that it necessitates new data structures, analysis workflows, and statistical modeling methods.

In my work towards applying scRNA-seq analysis to studies of the profile of gene delivery vehicles and the variability in immune signatures, several needs arose that were not met by current solutions. In this chapter, I discuss a framework for thinking about scRNA-seq data that can meet the growing complexity of experimental scRNA-seq paradigms. I then discuss the unique data structure needs of large, sparse scRNA-seq datasets, and propose a simple, extensible data format that exceeds the performance of current gold standard scRNA-seq formats. Finally, I discuss software developed in Matt Thomson's lab to handle the growing scale of single-cell studies. Together, these tools afforded us the ability to rapidly iterate our hypotheses and explore our data across large numbers of samples, and they may, in part or in whole, be of use to future researchers seeking to perform large-scale single-cell studies.

3.2 Abstractions for single-cell analysis

Different software packages, experimental protocols, and data analysis methods among the scRNA-seq community use a variety of terminology to refer to the different elements of the scRNA-seq field.

However, subsets of these terms (such as “library,” “barcode,” “index,” “read,” “transcript,” “UMI,” “sample,” “batch”) are sometimes used interchangeably, or to refer to different concepts in different contexts, which can make it difficult to understand what is meant in a specific workflow, paper, or protocol. Furthermore, the variety of protocols that involve multiplexing and demultiplexing different components at different stages makes it difficult to decide what the right abstraction is. For example, if 3 tissue samples are pooled together prior to loading into a single lane of 10X, is this 1 sample or 3? Does this change if the sample is able to be demultiplexed computationally later? What if the samples are tagged with lipid-anchored oligos prior to loading them into the 10X lane? What if, after merging the samples together, all 3 samples are split among 8 10X lanes? Is this one sample, or 3, or 8, or 24, or somewhere in between?

Before discussing some tools and methods I have developed for scRNA-seq analysis, I will describe the nomenclature and abstractions that I think are a good way to represent the different elements of scRNA-seq experiments, and that capture the possible complexity of scRNA-seq workflows. At its core, these abstractions, and all single-cell sequencing workflows, revolve around the idea of taking nucleic acid molecules, and grouping them into sets. The most common set grouping is a “Cell,” but from a data perspective, there is nothing different about a cell vs. an exosome, or a debris-filled droplet. The abstractions that arise from this fundamental molecular ground truth are:

- Nucleic Acid Molecule: A single molecule of DNA or RNA.

- Nucleic Acid Set (a “Cell”): An (unordered) grouping of multiple Nucleic Acid Molecules. While I will refer to this as a Cell, it is important to note that non-cell groupings of Nucleic Acid Molecules form a large part of many single-cell datasets, such as exosomes or debris.

- Cell Set: An (unordered) grouping of multiple Nucleic Acid Sets. This can be within a single sample, or across any number of samples. This is the fundamental grouping in scRNA-seq data, and applies to many concepts: cells that are part of a single Tissue Sample, cells that are part of the same Batch, or computationally-derived groupings made post facto.

- Read: A contiguous sequence of a digital readout of a Nucleic Acid.

- Read Set: An (unordered) set of Reads. Commonly implemented as a FASTQ file. A Read Set originates from a Sequencing Run.

- Barcode: A generic term referring to any number of possible DNA sequences that identifies something about a Nucleic Acid. This could be a cell barcode (as in the 10X cell barcode), an Illumina sample index, a cell hashing tag, a variant sequence, a feature barcode, and many more.

With this abstraction, many common scRNA-seq experimental terms can be thought of in this context:

- Sequencing Library: A Nucleic Acid Set, with sequencing adapters and one or more sample indices attached to the end of the molecules.

- Sequencing Run: A single processing of one or more Sequencing Libraries to obtain associated Read Sets.

- Tissue Source: A Cell Set consisting of cells from single organism, well, or person, extracted at a particular time.

- Tissue Sample: A Cell Set consisting of a single physical sample, from a single Tissue Source. Multiple Tissue Samples can originate from a single Tissue Source, e.g. in the case of extracting multiple regions from a sample, or sorting cells via FACS prior to single-cell isolation.

- Batch: A Cell Set consisting of all cells processed as a single scRNA-seq reaction. This is a lane of 10X, or a full split-pool kit.

Using this abstraction, most scRNA-seq experimental and computational protocols can be thought of in terms of assignment operations of these core elements. The core operations, then, are:

1. Associating Reads with Nucleic Acids

    a) Alignment: Determining the origin Nucleic Acid Molecule associated with a Read.

    b) UMI Collapsing: Assign multiple Reads to the same Nucleic Acid.

2. Assigning Nucleic Acid Molecules to Nucleic Acid Sets. For example:

    a) Gene Counting: Assigning UMIs to Cells based on cell barcodes.

    b) Cell Demultiplexing: Demultiplexing reads from Cell multiplets into separate Cells.

3. Assigning Nucleic Acid Sets to Cell Sets. For example:

    a) Cell Calling: The process of determining whether a Nucleic Acid Set is a true "Cell," debris, or something else.

    b) Cell Typing: The process of assigning multiple Cells to named Sets (the "types").

    c) Sample Demultiplexing: The process of assigning multiple Cells to named Sets associated with a particular Tissue Sample.

In most cases, application-specific terminology of each of these concepts will be used; for example:

- Referring to a Nucleic Acid Molecule as a "Transcript" in RNA applications or a "UMI," generally.

- Referring to a Nucleic Acid Set as a “Cell Barcode”
- Referring to a Cell Set as a “Cell Type” or “Cell State”
- Referring to a Barcode identifying a Nucleic Acid Molecule’s originating Sequencing Library as a “Sample Index.”

With these abstractions, it becomes clear that many steps in single-cell sequencing workflows are, in fact, the same fundamental information transformation operations, just performed with different protocols, parameters, or algorithms. With such abstractions in place, it is my hope that data structures, software packages, and analysis workflows can be cross-applied to more applications than their original source application, decreasing the redundancies of reinventing the same algorithms and software procedures for new applications.

3.3 sparsedat: An on-disk data format for sparse data

3.3.1. Summary

There are two typical workflows of scRNA-seq file management:

1. **Aggregated file for all samples**: In this workflow, sample gene count matrices are combined into a single file, typically by adding an additional field or modifying the cell identifier to indicate the originating sample.
2. **Per-sample matrices**: In this workflow, each sample has its own data matrix containing gene counts. This is the default output of single-cell RNA sequencing processing software, such as Cell Ranger, kallisto, and salmon.

There are benefits and drawbacks to each workflow. Having an aggregated file for all samples makes it easy to keep data associated with a single experiment together, and if all data can fit into memory, makes it easy to work with data interactively. The major drawbacks are that as new samples arrive, this file must be updated with the new data, and even if a researcher wants to access only one sample,

they have to open the file for the entire experiment. Alternatively, having a separate matrix per sample has the advantage of easily loading only the necessary samples, and adding samples as additional files to workflows.

Given the growth in the scale of single-cell RNA sequencing studies, the model of having an aggregated file with all data from an experiment will become increasingly cumbersome and reach beyond the memory limits of data scientist workstations, and even cloud cluster nodes. However, data analysis across hundreds or more samples is an exciting prospect, and data structures will need to accommodate these demands.

Fortunately, one of the most common use cases of single-cell RNA sequencing data analysis is to consider only subsets of cells or genes within the matrix for downstream processing. For example, although a 10X Chromium v3 data matrix has 6.7M entries for all possible cell barcodes, typically a researcher will only be interested in the cell barcodes that are likely to contain single intact cells (on the order of 10^4 per sample), or even specific subpopulations of cells corresponding to a single cell type (on order 10^2-10^3 per sample). As another example, researchers are often only interested in a subset of the genes (columns) present in a sample, focusing on genes that are relevant to their area of research, or that they have particular hypotheses on.

Thus, for the overwhelming number of use cases, data scientists do not need the entire gene count matrix for any particular analysis workflow, and are instead interested in slicing the matrix along subsets of rows (cells) and genes (columns).

This presents an opportunity for a data structure that would solve the above use cases and provide a workflow that could scale to hundreds or more samples with smaller memory footprints. To this end, I developed the sparsedat (Sparse Data Table, file extension .sdt) file format specification and accompanying Python interface. By giving the ability to access rows and columns of a sparse matrix directly from disk without having to load the full matrix, sparesedat dramatically improves performance for the use case of users wanting random access to rows and/or columns of a sparsely encoded matrix. sparsedat can also be used like a traditional sparse matrix format that is loaded fully

into memory by toggling a switch during initialization. Thus, sparsedat offers the speed of access to specific entries in a sparse matrix, without sacrificing the performance boosts associated with working in memory. As single-cell RNA sequencing studies continue to grow in scale in terms of number of samples, I hope that this format, or the principles that make it possible, will enable new, interactive workflows that speed up the scientific discovery process of these large datasets.

3.3.2 File format specification

Adapted from: https://github.com/thomsonlab/sparsedat

3.3.2.1 Overview

A semi-minimalist sparse data format that attempts to strike a balance between generalizability, functionality, and efficiency. The main focus is to allow both row- and column-indexed slicing of sparse matrices on demand (i.e. without having to load the full sparse matrix from disk).

3.3.2.2 Key features

- Simultaneous row and column sparse matrices stored in the same file;
- Binary encoding to minimize data storage and allow byte access to row or column elements;
- Built-in row and column names, as well as an expandable metadata specification.

3.3.2.3 File sections

The SDT file is broken down into 4 primary sections:

1. Header information
2. Metadata
3. Row and column indices

4. Data

3.3.2.4 Header information

The header information is a fixed number of bytes and should contain all the information needed to know:

- How to calculate the bytes to skip directly to the other 3 sections (metadata, row and column indices, and data)
- Which parser(s) are needed to process the data (version and data type)

Num bytes	**Description**	**Format**	**Example**	**Notes**
8	Version tag	SDTv[VERSION]	SDTv0001	[VERSION] is a 4-byte left padded ASCII representation of the version of the format
1	Data type id	[DATA_TYPE_ID]	1	[DATA_TYPE_ID] is an unsigned integer representing the data type of data contained in this table (See Data Types)
1	Data size	[NUM_BYTES]	8	[NUM_BYTES] is a long unsigned integer representing the number of bytes per data element
4	Number of rows	[NUM_ROWS]	500	[NUM_ROWS] is an unsigned integer representing the number of rows in the table

4	Number of columns	[NUM_COLUMNS]	500	[NUM_COLUMNS] is an unsigned integer representing the number of columns in the table
8	Metadata size	[METADATA_SIZE]	4096	[METADATA_SIZE] is an unsigned integer stating how many bytes makes up the metadata section

Note that by knowing the metadata size, one can skip directly to the indexes, and knowing the number of rows and columns, one can skip over the indexes to get to the data.

3.3.2.5 Metadata

The metadata section starts with a metadata index so that parsing what metadata is available is quick and does not require loading the entirety of the metadata into memory

Num bytes	**Description**	**Format**	**Example**	**Notes**
4	Number of metadata entries	[NUM_METADATA_ENTRIES]	2	[NUM_METADATA_ENTRIES] is an unsigned int representing how many metadata entries there are

Now, for each metadata entry, we have:

Num bytes	Description	Format	Example	Notes
4	Metadata type id	[METADATA_TYPE_ID]	0	[METADATA_TYPE_ID] is an unsigned integer representing the metadata type of this entry (See 3.3.2.9 Metadata Types)
4	Metadata start byte	[METADATA_START_BYTE]	240	[METADATA_START_BYTE] is a byte offset relative to the end of the metadata index of where the data for this metadata entry begins

Then, for each metadata entry, the bytes associated with that metadata are stored sequentially.

3.3.2.6 Row and Column Indices

This section contains a full sorted list for each row and column. Each entry is the start byte (relative to the first data entry) of where this row's data is contained. This can be used to jump directly on disk to the data contained in a particular row or column.

3.3.2.7 Data

The data section contains the default value, row-index data, and column-indexed data subsequently.

Num bytes	Description	Format	Example	Notes
DATA_SIZE	Default value	[DEFAULT_VALUE]	0	[DEFAULT_VALUE] is of the same type and size as specified by DATA_TYPE_ID and

				DATA_SIZE and specifies the default value of a table if the index is not specified in the sparse table.

3.3.2.7.1 Row Data

Num bytes	Description	Format	Example	Notes
4	Column index	[ROW_COLUMN_INDEX]	0	[ROW_COLUMN_INDEX] specifies the column index of this data entry in the current row
DATA_SIZE	Value	[DATA_VALUE]	15	[DATA_VALUE] is the value of the entry at the current row-column index

3.3.2.7.2 Column Data

Num bytes	Description	Format	Example	Notes
4	Row index	[COLUMN_ROW_INDEX]	0	[COLUMN_ROW_INDEX] specifies the row index of this data entry in the current column
DATA_SIZE	Value	[DATA_VALUE]	15	[DATA_VALUE] is the value of the entry at the current row-column index

3.3.2.8 Data Types

Data type id	Data type	Notes

0	Unsigned Int	Can be 2, 4, or 8 bytes
1	Int	Can be 2, 4, or 8 bytes
2	Float	Always stored as double precision (8 bytes)

3.3.2.9 Metadata Types

Metadata type id	**Data type**	**Notes**
0	Row Names	A list of UTF-8 encoded strings (max length 256), separated by a single length byte
1	Column Names	A list of UTF-8 encoded strings (max length 256), separated by a single length byte

3.3.3 Usage

Adapted from:

https://github.com/thomsonlab/sparsedat-py

3.3.3.1 Importing

To use the Sparse_Data_Table object, import via:

from sparsedat import Sparse_Data_Table

3.3.3.2 Workflow

In general, the workflow of sparsedat proceeds like this:

1. Create an SDT file from existing data/file

2. Initialize a Sparse_Data_Table object from an SDT file path (optionally without loading the whole file from disk)

3. View and manipulate the object

4. If changes want to be saved, save the object.

3.3.3.3 Key Considerations

- sparsedat will not save any changes to files without you explicitly calling the save function!
- sparsedat will by default save back to the same file path as created, so be careful to change file paths if you do not want to overwrite!
- Indexing a sparsedat object returns a new sparsedat object, without a file path specified. You must specify a file path to save subsampled data
- There are no operations built in—you must convert a sparsedat object to numpy or pandas in order to do calculations.

3.3.3.4 Creating an SDT file

There are currently 3 ways to create an SDT file. The first two, from row-column values and sparse representation, do not consider row and column names. If you want to add row and column names, you can do that separately.

3.3.3.4.1 From row-column values

If you have a list of row and column indices and their values, you can use Sparse_Data_Table.from_row_column_values(). Example usage:

```
from sparsedat import Sparse_Data_Table
```

```
row_column_values = [
    (0, 1, 5),
    (1, 2, 15),
    (5, 4, 2)
]

sdt = Sparse_Data_Table()
sdt.from_row_column_values(
    row_column_values,
    num_rows=8,
    num_columns=8,
    default_value=0
)

sdt.save(file_path="test.sdt")
```

3.3.3.4.2 From sparse row or sparse column representation

This operates the same as scipy's csr and csc initialization functions, as in scipy.sparse.csr_matrix

```
from sparsedat import Sparse_Data_Table

# Specify the starting index of each row you have data for
row_start_indices = [0, 2, 3]

# Specify the column index of each row entry
row_column_indices = [1, 5, 1, 1, 2, 4]

# The data values
values = [10, 10, 2, 1, 1, 8]

sdt = Sparse_Data_Table()

sdt.from_sparse_row_entries(
    (
        values,
        row_column_indices,
        row_start_indices
    ),
    num_rows=len(row_start_indices),
    num_columns=8,
    default_value=0
)

sdt.save(file_path="test.sdt")
```

The same can be done in sparse column format with Sparse_Data_Table.from_sparse_column_entries

3.3.3.4.3 Adding row/column names

To add row and/or column names to a loaded file, before saving:

```
...

sdt.row_names = ["Row 1", "Row 2", "Row 3"]
sdt.column_names = ["Col %i" % (i + 1) for i in range(8)]

sdt.save(file_path="test.sdt")
```

3.3.3.4.4 From mtx

Finally, there is a wrapper function for creating an SDT file from MTX format.

Example:

```
import os
test_data_directory = os.path.join("test", "data")

from sparsedat import wrappers as sparsedat_wrappers

sdt = sparsedat_wrappers.load_mtx(
    os.path.join(test_data_directory, "features.tsv"),
    os.path.join(test_data_directory, "barcodes.tsv"),
    os.path.join(test_data_directory, "matrix.mtx")
)

sdt.save("test_mtx.sdt")
```

3.3.3.5 Using an SDT file

3.3.3.5.1 Loading

To load an SDT file:

```
from sparsedat import Sparse_Data_Table

sdt = Sparse_Data_Table("test_mtx.sdt")
Optionally, you can load an SDT file without loading it all to memory. This will
reduce the memory footprint, but will require reading from disk each time you
access it.
from sparsedat import Sparse_Data_Table
```

```
sdt = Sparse_Data_Table("test.sdt", load_on_demand=True)
```

3.3.3.5.2 Indexing

Sparsedat-py supports several types of indexing: location, boolean, or name-based. Values can be either slices, lists, or individual values. Using the test_mtx.sdt from above, here are some examples:

```
from sparsedat import Sparse_Data_Table

sdt = Sparse_Data_Table("test_mtx.sdt")

# Get value by direct location
sdt[0, 0]

# Get an entire row
sdt[42, :]

# Boolean indexing
sdt[:, [True, True, True, False, False, False, False, True, True, False]]

# Named indexing
sdt["ENSG00000187608\tISG15\tGene Expression", :]
```

3.3.3.5.3 Conversion

By default, indexing a Sparse_Data_Table returns another Sparse_Data_Table. However, you may want to do arithmetic or other actions using numpy or pandas objects. A Sparse_Data_Table object can be converted as follows:

```
from sparsedat import Sparse_Data_Table

sdt = Sparse_Data_Table("test_mtx.sdt")

# To a numpy array
sdt.to_array()

# To a pandas object
sdt.to_pandas()
```

3.3.3.5.4 Adding rows and columns

Currently, directly updating values is not supported. However, adding rows and columns is supported by providing all non-default values for the new row/column, either by name or index. Using the test.sdt, with column and row names added from above:

```
from sparsedat import Sparse_Data_Table

sdt = Sparse_Data_Table("test.sdt")

new_values = [
    (0, 5),
    (3, 5),
    (5, 2)
]

sdt.add_row(new_values, row_name="New Row")

# Or by name

new_values = {
    "Col 1": 2,
    "Col 2": 3,
    "Col 6": 1
}

sdt.add_row(new_values.items(), row_name="New Row by Name")

# Let's see what we added
sdt[["New Row", "New Row by Name"], :]
```

3.3.4 Performance

I compared the performance of sparsedat to 5 other dominant scRNA-seq data formats: Anndata, Loom, mtx, Cell Ranger H5, and CSV on a typically-sized scRNA-seq dataset of 8,891 cells by 32,738 genes. However, the mtx, Cell Ranger H5, and CSV formats were so slow for this large of a matrix (loading times of 2-30s), that the read access simulations would have taken weeks to run, so I did not continue looking at mtx, Cell Ranger H5, or CSV (Figure 17**A**). First, I ran a test of random read access to individual elements in the matrix in batches of lookups. Unsurprisingly, since the Anndata format is stored in a comparable row-oriented sparse matrix, it closely follows the performance of the preloaded version of sparsedat. However, for small numbers of lookups (<100

per file open), sparsedat's on-demand version and Loom outperform Anndata and sparsedat's preloaded version (Figure 17**C**). This is as intended—for small numbers of lookups, sparsedat does not need to read in the entire sparse matrix, significantly reducing the overhead. Interestingly, Loom outperforms sparsedat's on-demand version after about 10 lookups. However, if a user knows they will be performing large numbers of lookups (1000 or more), preloading the sparsedat matrix will yield a boost in performance over Loom.

Next, I looked at the more common use case than a single entry: extracting entire rows or columns. Again, in the row case, sparsedat's preloaded performance matches Anndata; however, sparsedat far exceeds Anndata when extracting full columns (Figure 17**D, E**). This is likely due to the fact that the data was stored in row-oriented sparse format for the Anndata. This could easily be worked around by storing two separate Anndata files; one in row-oriented format and one in column-oriented format; however, this is not a native feature of Anndata.

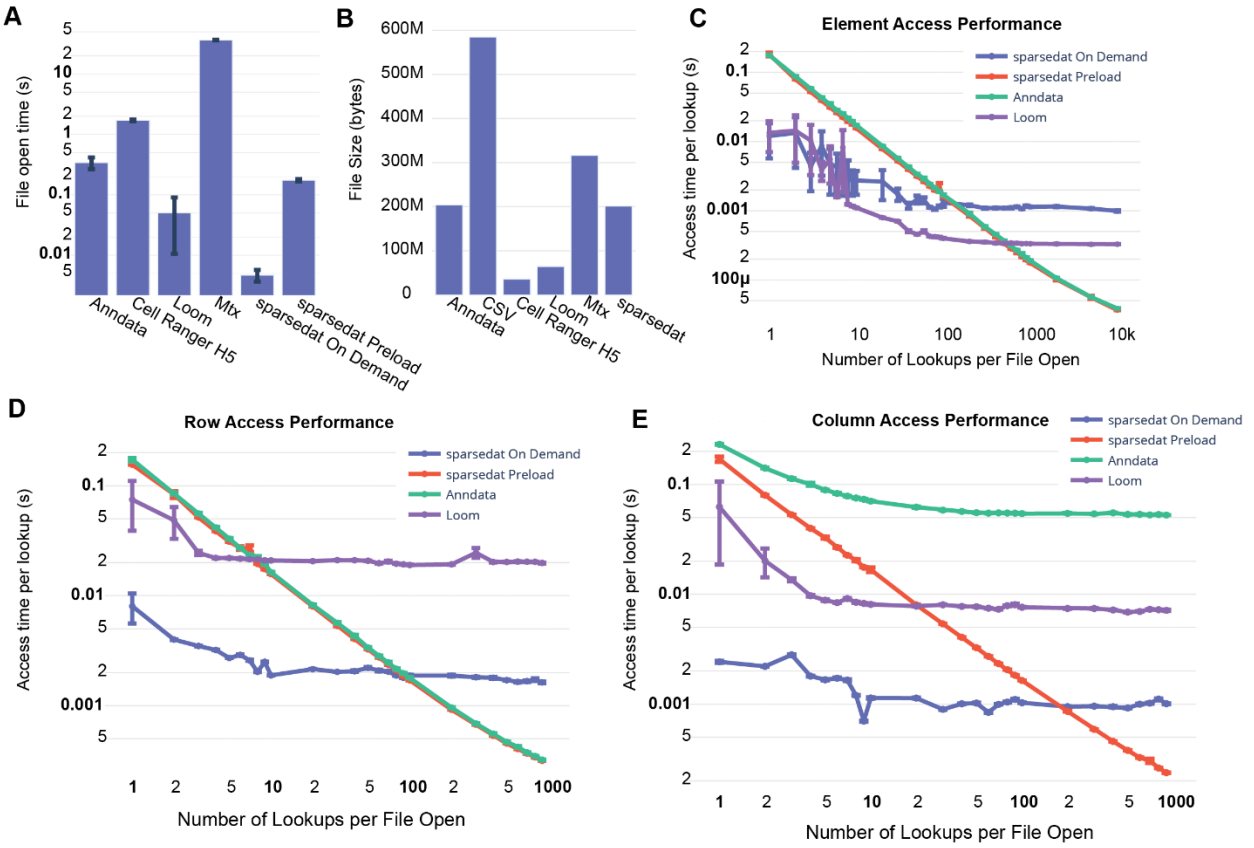


Figure 17. Sparsedat performance metrics. (**A**) Average file open times (n=3) for an 8,891x32,738 sparse matrix. (**B**) File sizes for each format for the same data. (**C**) Single element access performance. (**D**) Per-row access performance. (**E**) Full column access performance.

The performance of sparsedat is particularly striking in the regime of 1–100 row or column lookups using sparsedat's on-demand version. Compared to all other methods, even Loom, which was quite competitive in the single element lookup case, sparsedat can extract rows and columns 10–100x faster.

The performance boost of sparsedat primarily comes from two principles: 1) simultaneous storage of row and column-oriented data, with automatic detection of the access pattern to determine which orientation to use, and 2) optional direct on-disk access to full rows and columns. Currently, a user must know what regime they will be operating in, and set it with a flag, but future versions of sparsedat could incorporate automatic detection of user behavior and swap to fully loaded matrices

when user access patterns suggest a benefit. While these principles are currently implemented in the sparsedat Python package, future, more well-tested and scalable implementations in a lower-level language such as C would likely confer additional performance increases. Finally, this performance boost does come at a cost; sparsedat has a larger footprint on disk—comparable to Anndata, but 4 times greater than Loom (Figure 17**B**)—and has some regimes where it can be outperformed by Loom or Anndata; nonetheless, the performance boost for accessing rows and columns across many samples make it a compelling alternative to existing scRNA-seq standards.

3.4 Cloud-based infrastructure for managing single-cell RNA sequencing data

3.4.1 Summary

Single-cell RNA sequencing data has been growing in scale dramatically since the first single-cell RNA isolation in 2009, with studies in 2021 often reporting hundreds of thousands to millions of cells across 10s to 100s of samples. With data storage requirements on the order of 100 GB per sample and per-sample compute times for some operations, such as alignment, on the order of 256-core hours, largescale studies quickly become unmanageable for local computing, and require constant attention to organize data, optimize speed, and minimize cost.

To meet the needs for labs or single-cell sequencing cores that aim to host repositories of large numbers of samples, we have developed SCRAP, a modular, cloud-based architecture that takes advantage of a variety of data storage options, caching, virtual computing, and modern interactive visualization tools, and gives access to single-cell sequencing data to users via web-based interfaces and APIs. Such an architecture will be critical for having consistent, easy-to-use data analysis tools for a broad audience.

4.4.2 SCRAP: Single-cell RNA Analysis Platform

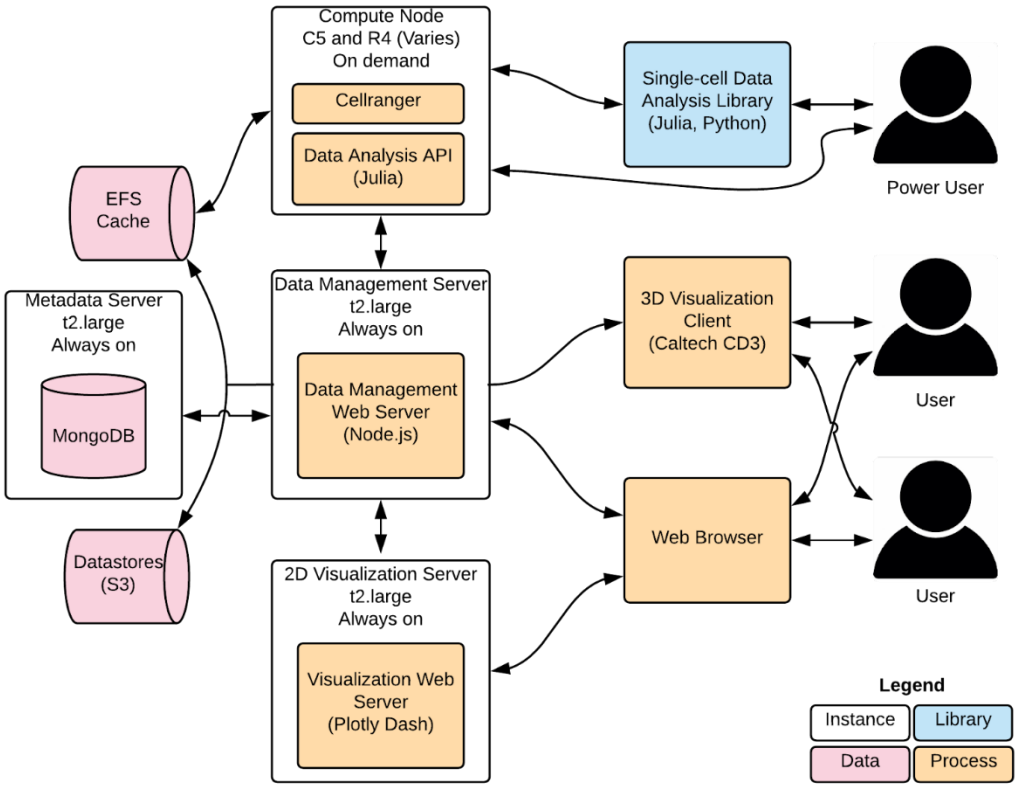


Figure 18. SCRAP architecture overview. An always-on data management server interfaces with a Mongo database, manages spinning up instances for on-demand compute needs, and serves up a web API for accessing data for visualization and analysis.

3.4.2.1 Architecture Overview

SCRAP (**S**ingle-**c**ell **R**NA **a**nalysis **p**latform) is a modular, Node.js-based web infrastructure that compartmentalizes its functions into several distinct components:

Data Management Server (scrap-dm): An always-on web server responding to data access and computation requests. This is a Node.js server serving up both an API and a web interface. This server handles shuttling data into the EFS cache, and starting up instances for compute jobs that are too demanding for the server to handle.

Metadata Server: An always-on web server hosting a Mongo database that contains metadata about what samples and transformations of data are available.

Datastores: Location of raw single-cell sequencing data, stored in S3. This data is not conducive for a database due to its large-scale, sparse nature.

EFS Cache: Recently-accessed data, or data needed for active compute operations, stored on EFS. Used for seamless access to data by on-demand instances.

Compute Nodes: Resource-heavy (either memory or CPU, depending on computation) instances that can either be started by power users manually, or will be automatically started by the Data Management Server. They will listen for API requests to perform compute operations using Julia.

3.4.2.2 User features

FASTQ file import: Ingest scRNA-seq data starting at its original source: the FASTQ files. Users can either upload local BCL files directly from a MiSeq, or can import data via FTP by supplying their user credentials, and choosing which samples to import.

Merging Read Sets: Merge read sets in the event that reads from a sample are split across multiple sequencing runs.

Alignment: Align read sets to a reference genome. Supports alignment via both Cell Ranger or kallisto. When an alignment is requested, the scrap instance monitor queues up a new instance that processes the alignment request, then shuttles data to the common data store when complete, and shuts down.

3.4.2.3 Back-end features

On-demand transfers from S3: Since in the typical scRNA-seq workflow, the raw FASTQ files are typically only processed once, SCRAP stores these files on an S3 infrequent access bucket. Users

can request files at any time via the web interface, and SCRAP will automatically shuttle data from S3 to a local EFS cache for temporary storage and serving it up to the user.

Automated queuing system: The only processes running at all times are the SCRAP management web serers; all other computationally expensive tasks such as transformations and alignments are queued up on user demand, and executed on ephemeral instances that are destroyed after their task is completed. This is accomplished by the SCRAP instance monitor, which is constantly checks the database for any unprocessed tasks, creates instances and assigns them to tasks, and then destroys instances when their assigned tasks are completed.

Self-managed compute nodes: Compute nodes, when launched, communicate with the database server to query whether they have any tasks assigned to them. This eliminates having to configure compute nodes to receive push connections from a central server, and makes task allocations more robust, since each compute node can operate independently.

3.4.2.4 Data structures

SCRAP implements many of the abstractions outlined in Section 3.2 Abstractions for single-cell analysis.

Chapter 4

METHODS FOR SINGLE-CELL SEQUENCING OF DELIVERED MUTANT TRANSCRIPTS

4.1 Summary

One promising area in the field of single-cell RNA sequencing (scRNA-seq) is the prospect of using individual cells in a sample as testbeds for perturbations. Unlike traditional perturbation experiments, which operate at the scale of individual animals or *in vitro* cultures, an experimental paradigm where each cell receives and responds to a perturbation independently could yield orders of magnitude improvements in scale. The key to making such a paradigm functional is that the phenotype and the perturbation must be captured via sequencing, and the two must be reliably linked. RNA and DNA therapies delivered to cells are excellent candidates for such a paradigm, due to the ease with which libraries of DNA or RNA mutants can be produced and delivered. However, current single-cell RNA sequencing assays are typically unable to confidently call the presence of specific, individual RNA transcripts in a sample due to a variety of noise sources, such as subsampling, amplification, and template switching.

One of the key distinguishing features of modern single-cell RNA sequencing workflows is that they are broad, high-dimensional assays. Rather than capturing particular targeted metrics, they capture a wide class of metrics in a single experiment. The most common single-cell RNA sequencing workflows use a poly(dT) oligo capture probe such that they are able to broadly capture all polyadenylated RNAs, but other workflows exist with a similar breadth-first perspective, such as chromatin accessibility profiling (Lareau et al., 2019), methylation status (Li et al., 2019), T- and B-cell receptor sequencing (Singh et al., 2019), and total RNA capture (Hayashi et al., 2018). In all cases, however, the experimental goal is to capture large subsets of DNA, rather than specific transcripts.

Unfortunately, such breadth-first experimental protocols have a drawback: since the eventual readout of any scRNA-seq protocol is a next-generation sequencing run, and the scale of sequencing depth currently available to researchers is significantly below the biological scale of the numbers of RNA molecules in a sample, the data will necessarily be a subsampling. For many applications, such as determining cell type or state, or for differential gene expression between samples or subpopulations of cells, this subsampling can be sufficient, since cells can be combined in the data for determining statistically significant differences. However, in other applications, such as analyzing rare transcripts, identifying the allele status of individual cells, or determining delivery of mutants to individual cells, this subsampling means that the transcript or region of interest may not be reliably captured by standard workflows. To this end, a variety of protocols have been developed to add to the standard scRNA-seq workflow, such as capturing targeted panels of genes of interest (Saikia et al., 2019), amplifying specific cells within a sample (Riemondy et al., 2019), or identifying mutant alleles (Rodriguez-Meira et al., 2019, 2020).

Ideally, in order to use each cell in a scRNA-seq experiment as an independent test of mutants in a library of delivered therapies, researchers need a highly accurate and specific readout of which therapy was delivered to each cell. Described herein is a series of molecular biology and computational techniques that enable more robust quantification of the presence of specific transcripts in the presence of technical noise.

4.2 Amplification of transcripts

One straightforward strategy to counteract the undersampling of transcripts of interest relative to other captured molecules in an scRNA-seq nucleic acid library is to selectively amplify the transcripts of interest in an additional, parallel PCR amplification step on an aliquot of the library. In the droplet-based 10X Chromium scRNA-seq protocol, for example, each mRNA molecule is captured with a poly(dT) oligo that contains a common sequence on its 5’ end. After mRNA capture and reverse transcription, this yields a cDNA library that contains full-length mRNA transcripts with a common sequence after the polyadenylation region. By using a primer pair that binds to this

common region and to a region unique to the transcript of interest, this transcript can be selectively amplified above the level of the rest of the cDNA library. Additionally, by choosing a primer on the transcript of interest that is upstream of any target regions of interest, these regions can be sequenced with high accuracy short-read sequencing platforms, such as Illumina's next-generation sequencers.

To test this targeted amplification strategy, we performed such an amplification on a cDNA library of single-cell suspensions of brain cells from a mouse that received an intravenous injection of AAV-PHP.eB carrying mNeonGreen (see Section 5.3.2 Single-cell RNA sequencing recapitulates AAV capsid cell-type-specific tropisms). For the amplification, we used primers specific to the delivered (see Section 5.7.6 Viral library construction).

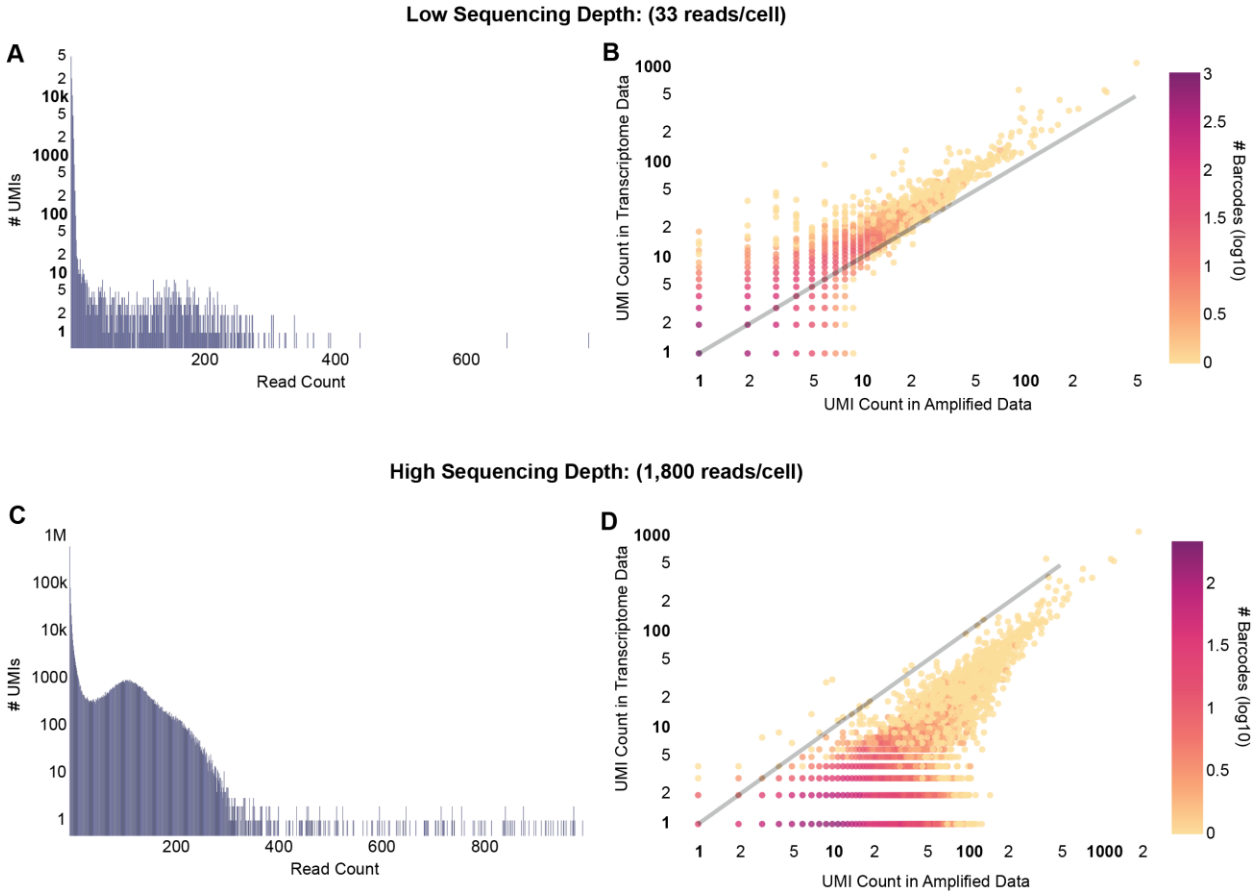


Figure 19. Comparison of sequencing depth of selectively amplified transcripts. (A, C) The read count distribution of UMIs in a low **(A)** and high **(C)** sequencing depth amplification library. **(B, D)** A comparison between UMI counts per cell in the original transcriptome data (y) as compared to the amplified library (x).

For our first sequencing attempt, we sequenced the reads at a sequencing depth of 326,236 reads. Given our expected cell count of 10,000 cells and a manufacturer-recommended 30,000 reads/cell for a full transcriptome library, this read depth would give about 0.1% of the recommended total, which, we estimated, was a reasonable estimate for the abundance for our gene of interest. However, when we inspected the distribution of read counts for each unique cell barcode/UMI combo, we saw read counts that were much higher (up to 793 reads) than the typical 10 or so reads per UMI in full transcriptome data (Figure 19**A**). We suspect that this originates from additional rounds of PCR

amplification, increasing the variance of the distribution due to the compounding exponential amplification.

We then inspected the UMI counts of these amplified reads as compared to the UMI counts of the delivered transgene in the transcriptome data, which had one less round of PCR amplification (Figure 19**B**). The overwhelming majority of the barcodes had very low UMI counts (1 or 2), and there were UMI counts in both the amplified data and the transcriptome data that were not present in the other data. Additionally, at higher UMI counts, the amplified data underrepresented the UMI counts in the transcriptome data. Together, this suggests that the sequencing depth was too low to recover all the transcripts of interest, and that there is substantial disagreement between the two data sources, either due to noise or both methods undersampling the transcripts of interest.

To explore this further, we sequenced the amplified DNA library again to a much greater sequencing depth (18M reads, or 1,800 reads per cell). This time, while the distribution was still skewed (up to 45k reads for the largest UMI), inspection of the lower counts of the histogram revealed a potential multimodal distribution (Figure 19**C**). We again investigated the agreement between the UMI counts in cell barcodes in this amplified data compared to the transcriptome data, and discovered a different effect from before; the amplified UMI counts were now consistently higher than the UMI counts in the transcriptome data, with some rare exceptions (Figure 19**D**). Additionally, there were almost no UMIs in the transcriptome data that did not exist in the amplified data, and there were large numbers of cell barcodes that had UMI counts in the amplified data but not the transcriptome. Together with the low sequencing depth experiment, this suggests two possibilities: either the high sequencing depth amplified data recovers large numbers of transcripts that were not present in the original transcriptome library, or the amplification introduces significant artifacts.

To determine whether these transcripts are likely to be artifacts, we investigated to see how many cell barcodes had at least one delivered transcript across different cell types and found that all cell types had transcripts detected in 97.8% or more of their cells. This is in direct contradiction to the repeated finding that both wild-type and engineered AAV variants have low transduction of

microglia (Bartlett et al., 1998; Deverman et al., 2016; Foust et al., 2009). This suggests that the UMI count increase in the amplified data originates at least partially from an amplification artifact.

4.3 Maximum likelihood estimation for reducing PCR amplification noise

Given the likelihood of an artifact in the amplified data and the presence of a multimodal distribution in the read counts of amplified transcripts, I hypothesized that the different modes of the distribution represented a combined distribution of signal and noise. Unfortunately, even with the high sequencing depth of 1,800 reads per cell, the distribution did not have a clear location where a binary threshold could cleanly separate the signal from the noise. Therefore, I decided to fit a mixture model to the read count data using Maximum Likelihood Estimation. Based on the shape of the distribution and the commune usage of the negative binomial for read count data, I started with a mixture of two negative binomials. However, initial attempts to fit two negative binomials resulted in models that failed to capture the region of counts between the low counts and the first peak. Despite the expressiveness of the negative binomial distribution, there was no parameter regime that could account for the early steep decrease in the distribution, but still have a tail that overlaps a second negative binomial. Upon further inspection of the count distribution, there appeared to be nearly exponential drop in UMI counts in the low regime—i.e. from 1 to 2, 2 to 3, etc. Given that there are at least two unique biological sources of errors—i.e. point mutation errors during sequencing, where the probability of errors can be modeled exponentially, and more complicated PCR-based errors, which might propagate over rounds of PCR—I then moved to modeling the distribution as a mixture model of three distributions—one binomial and two negative binomials (Figure 20**A**). Given the long tail of the signal distribution and the possibility of mode collapse for the two negative binomials, there were some additional considerations when performing the MLE procedure. The complete procedure is outlined below:

1. Find the first trough (inverse peak) in the data using signal.find_peaks, and choosing the most prominent trough. Set this to be the threshold, *t*.

2. Perform MLE to fit parameters *n* and *p* of a negative binomial distribution, conditioned on a count greater than the threshold, that minimizes the negative log likelihood of all counts above the threshold.

$$P(n, p|X: X > t, X > 0)$$

3. Perform MLE to fit the parameters n_1, p_1 of a binomial distribution and n_2, p_2 of a negative binomial distribution, and the weight, *w*, between them, both conditioned on counts between 1 and the threshold. Additionally, we constrain the noise negative binomial to have a larger *p* than the signal negative binomial to prevent mode collapse.

$$P\big(n_1, n_p, n_2, p_2, w\big|X: X < t, X > 0, p_2 > p\big)$$

4. Finally, we relax the constraints of the threshold counts, and perform a final MLE to fit the weights of the three distributions

$$P(w_1, w_2, w_3|X: X > 0, n, p, n_1, n_p, n_2, p_2).$$

After fitting the mixture model, we can then use Bayes' rule to calculate the probability whether a given read count originated from the signal distribution or one of the noise distributions (Figure 20**B**):

$$P(Real|X) = \frac{P(X|Real)P(Real)}{P(X)}$$

After fitting the overamplified read counts using this procedure, we investigated its effect on the UMI counts per cell, and discovered that, remarkably, the procedure recovered the original counts strongly in accordance with the original transcriptome counts, suggesting that this fitting procedure performs well at recovering true molecular counts in the presence of strong amplification noise (Figure 20**C**).

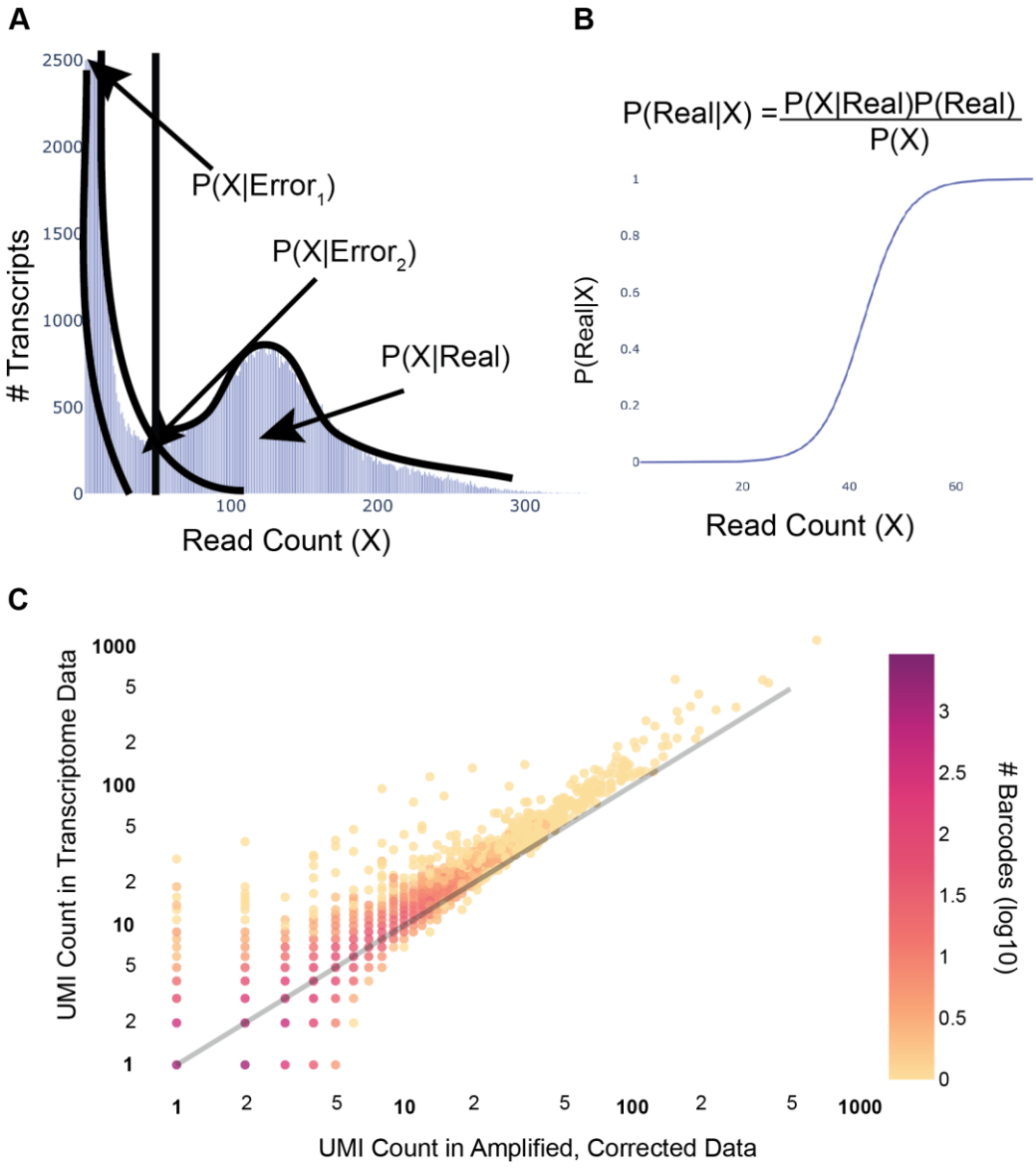


Figure 20. Maximum likelihood estimation for correcting PCR amplification noise. (A) The multimodal read count distribution is fitted with a signal (real) distribution and 2 noise (noise) distributions. **(B)** Bayes' rule is used to calculate the probability of a read count originating from the signal or noise. **(C)** A comparison between UMI counts per cell in the original transcriptome data (y) as compared to the amplified post-correction library (x).

4.4 Template switching artifacts

After correcting for PCR errors across all our datasets, we then performed downstream analysis on the distribution of transcripts present in different cell types in our data. Similarly to the unexpectedly high level of microglial transduction in the amplified, uncorrected counts, we uncovered an additional potential artifact: high estimated number of transcripts in oligodendrocytes in a few samples. In order to determine whether this was an artifact or a real signal, we explored the distribution of read counts of the amplified transcripts in cell barcodes belonging to oligodendrocytes. Interestingly, we found that the distribution of read counts was, in fact, different between oligodendrocytes and other cell types, such as neurons. Similarly, when we performed the MLE error correction procedure on UMIs that were restricted to oligodendrocytes and neurons separately, we found that the procedure learned a significantly different probability mapping, with higher read counts needed in oligodendrocytes to reach the same probability of being a real transcript as neurons (Figure 21**A,B**).

Template switching has been previously identified as a source of noise in NGS datasets (Kebschull and Zador, 2015). Thus, we hypothesized that there might be a cell-type-dependent template switching happening between our delivered and amplified cargo transcript, CAG-mNeonGreen-WPRE, and some transcript within the cell. This could happen if there is a region of significant homology between CAG-mNeonGreen-WPRE and a transcript that is expressed in higher levels in oligodendrocytes (Figure 21**C**). To investigate whether this was the case, we looked at the native reads in the transcriptome, and calculated what percent of Cell Barcode/UMIs that were associated with a delivered transcript also appeared in another transcript (**Table 2**). There was a surprising amount of Cell Barcode/UMIs, with at least 22.4% of viral transcripts having a corresponding transcript in the transcriptome with the same Cell Barcode/UMI across all cell types. However, the overlap was significantly higher (up to 51%) for both subtypes of oligodendrocytes, as well as microglia, suggesting a strong cell-type dependence of this artifact.

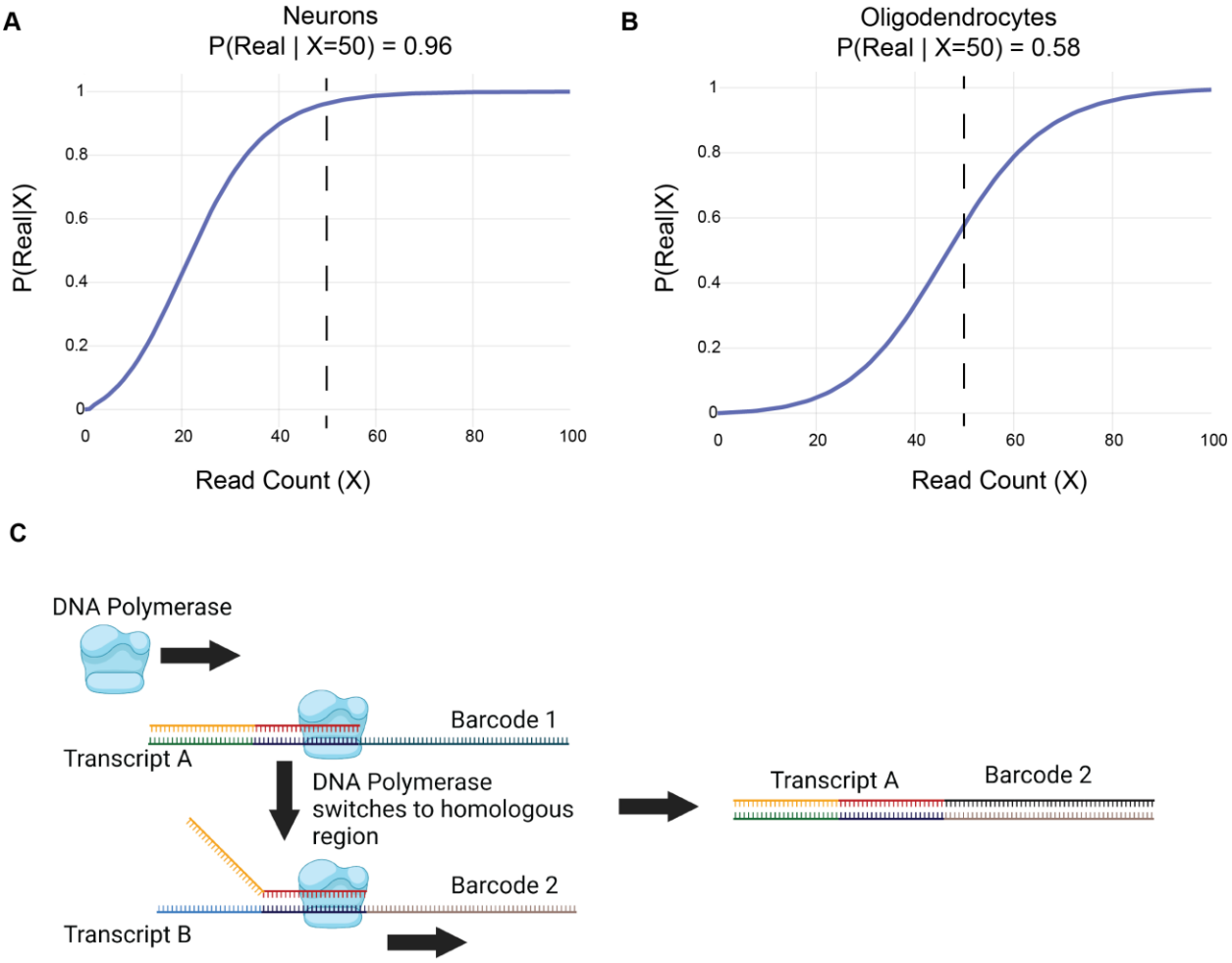


Figure 21. Fitted read count probabilities vary by cell type. (A, B) A given read count in neurons **(A)** has a much higher predicted probability of being a real transcript than in oligodendrocytes **(B)**. **(C)** A schematic of a possible template switching that may associate transcripts with the wrong barcode.

4.5 Conclusion

In general, the MLE method for PCR amplification error correction is an effective strategy for recovering accurate transcript counts, and worked across a variety of samples, amplicons, and sequencing depths we tested, as long as the sequencing depth is sufficient to reveal a detectable

trough separating the distributions. However, it is important to note that in our explorations, we uncovered an alarmingly high degree of template switching that we were able to detect due to the high sequencing depth of our targeted amplification libraries, and that this template switching can happen on a cell-type-dependent basis. Thus, it is important to keep in mind that error correction would ideally be performed on a per-cell-type basis for any cell types under study.

For use cases where the purpose of the targeted amplification is not to amplify a signal, but instead identify a region of interest on the 5' end, an alternative strategy for identifying the source transcripts of targeted amplifications would be to only keep amplified transcripts that have a corresponding Cell Barcode/UMI in the transcriptome data. This is the strategy we employed for our targeted amplification of multiplexed viral transcripts (see Section 5.7.10 Constructing the variant lookup table).

Although these artifacts were uncovered during our exploration of targeted amplification of transcripts of interest, it is possible that similar levels of template switching happen between native transcripts with some homology in standard scRNA-seq workflows. With a similar methodology and thought process as presented here, future work could uncover new, important artifacts, and develop computational methods to improve the confidence of single-cell transcriptome data.

Table 2. Comparison of UMI overlaps between CAG-mNeonGreen transcripts and native transcriptome transcripts across different cell types.

Cell Type	Percent UMI Overlap
Microglia	44.70%
Astrocytes	33.25%
Vascular Cells	26.39%
OPCs	48.87%
Mature Oligodendrocytes	51.06%
Pericytes	27.37%
Inhibitory Neurons	22.35%
Excitatory Neurons	25.23%

Chapter 5

DEEP PARALLEL CHARACTERIZATION OF AAV TROPISM AND AAV-MEDIATED TRANSCRIPTIONAL CHANGES VIA SINGLE-CELL RNA SEQUENCING

Adapted from:

Brown, D.*, Altermatt, M.*, Dobreva, T., Chen, S., Wang, A., Thomson, M., & Gradinaru, V. (2021). Deep parallel characterization of AAV tropism and AAV-mediated transcriptional changes via single-cell RNA sequencing. Cold Spring Harbor Laboratory. https://doi.org/10.1101/2021.06.25.449955

Updated version published at:

Brown, D.*, Altermatt, M.*, Dobreva, T., Chen, S., Wang, A., Thomson, M., & Gradinaru, V. (2021). Deep Parallel Characterization of AAV Tropism and AAV-Mediated Transcriptional Changes via Single-Cell RNA Sequencing. In Frontiers in Immunology (Vol. 12). Frontiers Media SA. https://doi.org/10.3389/fimmu.2021.730825

5.1 Summary

Engineered variants of recombinant adeno-associated viruses (rAAVs) are being developed rapidly to meet the need for gene-therapy delivery vehicles with particular cell-type and tissue tropisms. While high-throughput AAV engineering and selection methods have generated numerous variants, subsequent tropism and response characterization have remained low throughput and lack resolution across the many relevant cell and tissue types. To fully leverage the output of these large screening paradigms across multiple targets, we have developed an experimental and computational single-cell RNA sequencing (scRNA-seq) pipeline for in vivo characterization of barcoded rAAV pools at

unprecedented resolution. Using our platform, we have corroborated previously reported viral tropisms and discovered unidentified AAV capsid targeting biases. As expected, we observed that the tropism profile of AAV.CAP-B10 in mice was shifted toward neurons and away from astrocytes when compared with AAV-PHP.eB. Our transcriptomic analysis revealed that this neuronal bias is mainly due to increased targeting efficiency for glutamatergic neurons, which we confirmed by RNA fluorescence in situ hybridization. We further uncovered cell subtype tropisms of AAV variants in vascular and glial cells, such as low transduction of pericytes and Myoc+ astrocytes. Additionally, we have observed cell-type-specific responses to systemic AAV-PHP.eB administration, such as upregulation of genes involved in p53 signaling in endothelial cells three days post-injection, which return to control levels by day twenty-five. Such ability to parallelize the characterization of AAV tropism and simultaneously measure the transcriptional response of transduction will facilitate the advancement of safe and precise gene delivery vehicles.

5.2 Introduction

Recombinant AAVs (rAAVs) have become the preferred gene delivery vehicles for many clinical and research applications (Bedbrook et al., 2018; Samulski and Muzyczka, 2014) owing to their broad viral tropism, ability to transduce dividing and non-dividing cells, low immunogenicity, and stable persistence as episomal DNA ensuring long-term transgene expression (Daya and Berns, 2008; Deverman et al., 2018; Gaj et al., 2016; Hirsch and Samulski, 2014; Naso et al., 2017; Wu et al., 2006). However, current systemic gene therapies using AAVs have a relatively low therapeutic index (Mével et al., 2020). High doses are necessary to achieve sufficient transgene expression in target cell populations, which can lead to severe adverse effects from off-target expression (Hinderer et al., 2018; Srivastava, 2020; Wilson and Flotte, 2020). Increased target specificity of rAAVs would reduce both the necessary viral dose and off-target effects: thus, there is an urgent need for AAV gene delivery vectors that are optimized for cell-type-specific delivery (Paulk, 2020). Lower viral doses would also alleviate demands on vector manufacturing and minimize the chances of undesirable immunological responses (Calcedo et al., 2018; Gao et al., 2009; Mingozzi and High, 2013). Capsid-specific T-cell activation was reported to be dose-dependent in vitro (Finn et al., 2010;

Pien et al., 2009) and in humans (Mingozzi et al., 2009; Nathwani et al., 2011). Shaping the tropism of existing AAVs to the needs of a specific disease has the potential to reduce activation of the immune system by detargeting cell types, such as dendritic cells, that have an increased ability to activate T-cells (Herzog et al., 2019; Rogers et al., 2017; Rossi et al., 2019; Somanathan et al., 2010; Vandenberghe et al., 2006; Zhu et al., 2009).

Several studies have demonstrated that the transduction efficiency and specificity of natural AAVs can be improved by engineering their capsids using rational design (Bartlett et al., 1999; Davidsson et al., 2019; Davis et al., 2015; Lee et al., 2018; Sen, 2014) or directed evolution (Chan et al., 2017; Dalkara et al., 2013; Deverman et al., 2016; Excoffon et al., 2009; Grimm et al., 2008; Körbelin et al., 2016b; Kotterman and Schaffer, 2014; Maheshri et al., 2006; Müller et al., 2003; Ogden et al., 2019; Ojala et al., 2018b; Pekrun et al., 2019; Pulicherla et al., 2011; Ravindra Kumar et al., 2020; Tervo et al., 2016; Ying et al., 2010). These engineering methods yield diverse candidates that require thorough, preferably high-throughput, in vivo vector characterization to identify optimal candidates for a particular clinical or research application. Toward this end, conventional immunohistochemistry (IHC) and various in situ hybridization (ISH) techniques are commonly employed to profile viral tropism by labeling proteins expressed by the viral transgene or viral nucleic acids, respectively (Arruda et al., 2001; Chan et al., 2017; Deleage et al., 2016, 2018; Deverman et al., 2016; Grabinski et al., 2015; Hinderer et al., 2018; Hunter et al., 2019; Miao et al., 2000; Polinski et al., 2015, 2016; Puray-Chavez et al., 2017; Ravindra Kumar et al., 2020; Wang et al., 2020; Zhang et al., 2016; Zhao et al., 2020).

Although these histological approaches preserve spatial information, current technical challenges limit their application to profiling the viral tropism of just one or two AAV variants across a few gene markers, thus falling short of efficiently characterizing multiple AAVs across many complex cell types characteristic of tissues in the central nervous system (CNS). The reliance on known marker genes also prevents the unbiased discovery of tropisms since such marker genes need to be chosen *a priori*. Choosing marker genes is particularly challenging for supporting cell types, such as pericytes in the CNS microvasculature and oligodendrocytes, which often have less established cell

type identification strategies (Liu et al., 2020; Marques et al., 2016). The advent of single-cell RNA sequencing (scRNA-seq) has enabled comprehensive transcriptomic analysis of entire cell-type hierarchies, and brought new appreciation to the role of cell subtypes in disease (Berto et al., 2020; Gokce et al., 2016; Tasic et al., 2016, 2018; Zeisel et al., 2018). However, experimental and computational challenges, such as the sparsity of RNA capture and detection, strong batch effects between samples, and the presence of ambient RNA in droplets, reduce the statistical confidence of claims about individual gene expression (Lähnemann et al., 2020; Yang et al., 2020; Zheng et al., 2017). Computational methods have been developed to address some of these challenges, such as identifying contaminating RNA (Yang et al., 2020), accounting for or removing batch effects (Korsunsky et al., 2019; Lin et al., 2019; Lopez et al., 2018), and distinguishing intact cells from empty droplets (Lun et al., 2019; Macosko et al., 2015; Zheng et al., 2017). However, strategies for simultaneously processing transcripts from multiple delivery vehicles and overcoming the computational challenges of confidently detecting individual transcripts have not yet been developed for probing the tropism of AAVs in complex, heterogeneous cell populations.

Collecting the entire transcriptome of injected and non-injected animals offers an opportunity to study the effects of AAV transduction on the host cell transcriptome. A similar investigation has been conducted with G-deleted rabies virus (Huang and Sabatini, 2020). This study demonstrated that virus infection led to the downregulation of genes involved in metabolic processes and neurotransmission in host cells, whereas genes related to cytokine signaling and the adaptive immune system were upregulated. At present, no such detailed examination of transcriptome changes upon systemic AAV injection has been conducted. High-throughput single-cell transcriptomic analysis could provide further insight into the ramifications of AAV capsid and transgene modifications with regard to innate (Duan, 2018; Hösel et al., 2012; Martino et al., 2011; Shao et al., 2018; Zaiss et al., 2008) and adaptive immune recognition (George et al., 2017; Manno et al., 2006; Mingozzi et al., 2007; Nathwani et al., 2011, 2014). Innate and adaptive immune responses to AAV gene delivery vectors and transgene products constitute substantial hurdles to their clinical development(Colella et al., 2018; Shirley et al., 2020). The study of brain immune response to viral gene therapy has been limited to antibody staining and observation of brain tissue slices post direct injection. In particular,

prior studies have shown that intracerebral injection of rAAV vectors in rat brains does not induce leukocytic infiltration or gliosis (Chamberlin et al., 1998; McCown et al., 1996); however, innate inflammatory responses were observed (Lowenstein et al., 2007). Results reported by these methods are rooted in single-marker staining and thus prevent the discovery of unexpected cell-type-specific responses. A comprehensive understanding of the processes underlying viral vector or transgene-mediated responses is critical for further optimizing AAV gene delivery vectors and treatment modalities that mitigate such immune responses.

Here, we introduce an experimental and bioinformatics workflow capable of profiling the viral tropism and response of multiple barcoded AAV variants in a single animal across numerous complex cell types by taking advantage of the transcriptomic resolution of scRNA-seq techniques (Figure 22**A**). For this proof-of-concept study, we profile the tropism of previously-characterized AAV variants that emerged from directed evolution with the CREATE (AAV-PHP.B, AAV-PHP.eB) (Chan et al., 2017; Deverman et al., 2016) or M-CREATE (AAV-PHP.C1, AAV-PHP.C2, AAV-PHP.V1, AAV.CAP-B10) (Flytzanis et al., 2020; Ravindra Kumar et al., 2020) platforms. We selected the AAV variants based on their unique CNS tropism following intravenous injection. AAV-PHP.B and AAV-PHP.eB are known to exhibit overall increased targeting of the CNS compared with AAV9 and preferential targeting of neurons and astrocytes. Despite its sequence similarity to AAV-PHP.B, the tropism of AAV-PHP.V1 is known to be biased toward transducing brain vascular cells. AAV-PHP.C1 and AAV-PHP.C2 have both demonstrated enhanced blood–brain barrier (BBB) crossing relative to AAV9 across two mouse strains (C57BL/6J and BALB/cJ). Finally, AAV.CAP-B10 is a recently-developed variant with a bias toward neurons compared to AAV-PHP.eB (Flytzanis et al., 2020).

In our initial validation experiment, we quantify the transduction biases of AAV-PHP.eB and AAV-CAP-B10 across major cell types using scRNA-seq, and our results correlate well with both published results and our own conventional IHC-based quantification. We then demonstrate the power of our transcriptomic approach by going beyond the major cell types to reveal significant differences in sub-cell-type transduction specificity. Compared with AAV-CAP-B10, AAV-PHP.eB

displays biased targeting of inhibitory neurons, and both variants transduce Sst+ or Pvalb+ inhibitory neurons more efficiently than Vip+ inhibitory neurons. We validate these results with fluorescent in situ hybridization – hybridization chain reaction (FISH-HCR). We then develop and validate a barcoding strategy to investigate the tropism of AAV-PHP.V1 relative to AAV-PHP.eB in non-neuronal cells and reveal that pericytes, a subclass of vascular cells, evade transduction by this and other variants. We further use scRNA-seq to profile cell-type-specific responses to AAV.PHP-eB at 3 and 25 days post-injection (DPI), finding, for example, numerous genes implicated in the p53 pathway in endothelial cells to be upregulated at 3 DPI. While most upregulated genes across cell types return to control levels by day twenty-five, excitatory neurons show a persistent upregulation of genes involved in MAPK signaling extending to 25 days. Finally, we showcase the capabilities of parallel characterization by verifying the preceding findings in a single animal with seven co-injected AAV variants and reveal the unique non-neuronal tropism bias of AAV-PHP.C2.

5.3 Results

5.3.1 Multiplexed single-cell RNA sequencing-based AAV profiling pipeline

To address the current bottleneck in AAV tropism profiling, we devised an experimental and computational workflow (Figure 22**A**) that exploits the transcriptomic resolution of scRNA-seq to profile the tropism of multiple AAV variants across complex cell-type hierarchies. In this workflow, single or multiple barcoded rAAVs are injected into the retro-orbital sinus of mice followed by tissue dissociation, single-cell library construction using the 10X Genomics Chromium system, and sequencing with multiplexed Illumina next-generation sequencing (NGS) (Zheng et al., 2017). The standard mRNA library construction procedure includes an enzymatic fragmentation step that truncates the cDNA amplicon such that its final size falls within the bounds of NGS platforms (Figure 22**B**). These cDNA fragments are only approximately 450 bp in length and, due to the stochastic nature of the fragmentation, sequencing from their 5’ end does not consistently capture any particular region. The fragment length limit and heterogeneity pose a problem for parallelizing AAV tropism profiling, which requires reliable recovery of regions of the transgene that identify the

originating AAV capsid. For example, posttranscriptional regulatory elements, such as the 600 bp Woodchuck hepatitis virus posttranscriptional regulatory element (WPRE), are commonly placed at the 3' end of viral transgenes to modulate transgene expression. The insertion of such elements pushes any uniquely identifying cargo outside the 450 bp capture range, making them indistinguishable based on the cDNA library alone (Supplementary Figure 2**A**). An alternative strategy of adding barcodes in the 3' polyadenylation site also places the barcode too distant for a 5' sequencing read, and reading from the 3' end would require sequencing through the homopolymeric polyA tail, which is believed to be unreliable in NGS platforms (Chang et al., 2014; Shin and Park, 2016).

We circumvented these limitations in viral cargo identification by taking an aliquot of the intact cDNA library and adding standard Illumina sequencing primer recognition sites to the viral transcripts using PCR amplification such that the identifying region is within the two Illumina primer target sequences (e.g. Figure 23**B**). The cell transcriptome aliquots undergoing the standard library construction protocol and the amplified viral transcripts are then sequenced as separate NGS libraries. We sequence shorter viral transcripts in the same flow cell as the cell transcriptomes and longer viral transcripts on the Illumina MiSeq, which we found to be successful at sequencing cDNAs up to 890 bp long. The sequencing data undergoes a comprehensive data processing pipeline (see Methods). Using a custom genome reference, reads from the cell transcriptome that align to the viral cargo plasmid sequences are counted as part of the standard 10X Cell Ranger count pipeline (see Methods and Supplementary Figure 2**C**). In parallel, reads from the amplified viral transcripts are used to count the abundance of each viral barcode associated with each cell barcode and unique molecular identifier (UMI). The most abundant viral barcode for each cell barcode and UMI is assumed to be the correct viral barcode, and is used to construct a variant lookup table. This lookup table approach identifies an originating capsid in 67.6 ± 2.0% of viral transcripts detected in the cell transcriptome aliquots (Table S 4).

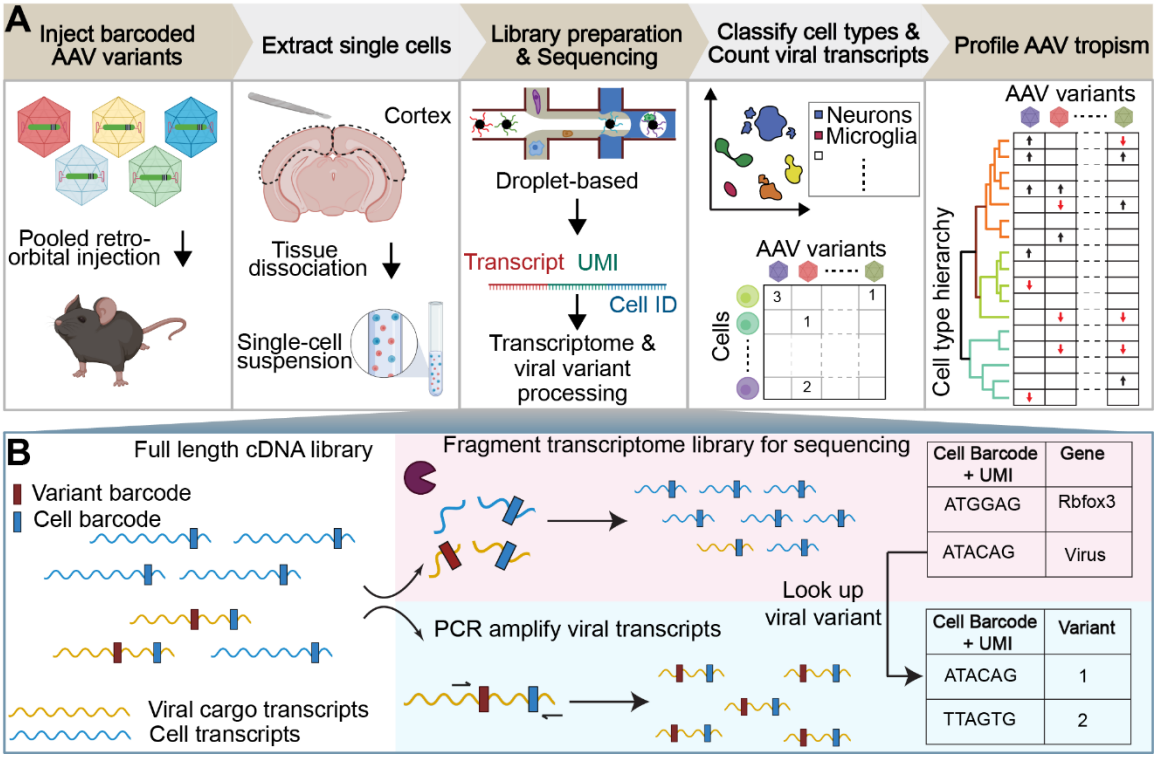


Figure 22. Workflow of AAV tropism characterization by scRNA-seq. (A) (I) Injection of a single AAV variant or multiple barcoded AAV variants into the retro-orbital sinus. (II) After 3–4 weeks post-injection, the brain region of interest is extracted and the tissue is dissociated into a single-cell suspension. (III) The droplet-based 10x Genomics Chromium system is used to isolate cells and build transcriptomic libraries (see **B**). (IV) Cells are assigned a cell-type annotation and a viral transcript count. (V) AAV tropism profiling across numerous cell types. **(B)** The full length cDNA library is fragmented for sequencing as part of the single-cell sequencing protocol (top). To enable viral tropism characterization of multiple rAAVs in parallel, an aliquot of the intact cDNA library undergoes further PCR amplification of viral transcripts (bottom). During cDNA amplification, Illumina sequencing primer targets are added to the viral transcripts such that the sequence in between the Illumina primer targets contains the AAV capsid barcode sequence. Viral cargo in the cell transcriptome is converted to variant barcodes by matching the corresponding cell barcode + UMI in the amplified viral transcript library (right).

For determining viral cell-type tropism, we developed a method to estimate the fraction of cells within a cell type that express viral transcripts. Viral RNA expression levels depend on both the multiplicity of infection and the transcription rate of the delivered cargo. Thus, directly using viral RNA counts to determine tropism is confounded by differences in transcription rate between cell types, limiting comparison with imaging-based tropism quantification methods. As evidence of this, we detected that viral RNA expression levels can vary by cell type but are not perfectly rank

correlated with the percent of cells detected as expressing that transcript (Supplemental Figure 2 B). An additional confound arises from the ambient RNA from cellular debris co-encapsulated with cell-containing droplets, which can lead to false positives, i.e., detecting viral RNA in droplets containing a cell that was not expressing viral RNA. For example, we detected low levels of viral transcripts in large percentages of cells, even in cell types suspected to evade transduction, such as immune cells (Supplementary Figure 3**A**). To reduce the effect of both variability in expression and ambient RNA, we developed an empirical method to estimate the percentage of cells expressing transcripts above the noise, wherein the distribution of viral transcript counts in a set of cells of interest is compared to a background distribution of cell-free (empty) droplets (see Methods, Supplemental Figure 2 C). In simulation, this method accurately recovers the estimated number of cells expressing transcripts above background across a wide range of parameterizations of negative binomial distributions (see Methods, Supplementary Figure 3**D**).

To address several additional technical problems in default single-cell pipelines, we developed a simultaneous quality control (QC) and droplet identification pipeline. Our viral transduction rate estimation method described above relies on having an empirical background distribution of viral transcript counts in empty droplets to compare against the cell type of interest. However, the default cell vs. empty droplet identification method provided by the 10X Cell Ranger software, which is based on the EmptyDrops method (Lun et al., 2019), yielded unexpectedly high numbers of cells and clusters with no recognizable marker genes, suggesting they may consist of empty droplets of ambient RNA or cellular debris (Supplementary Figure 4**A, B**). Additionally, we sought to remove droplets containing multiple cells (multiplets) from our data due to the risk of falsely attributing viral tropism of one cell type to another. However, using Scrublet (Wolock et al., 2019), an established method for identifying droplets containing multiplets, failed to identify multiplets in some of our samples and only identified small proportions of clusters positive for known non-overlapping marker genes, such as Cldn5 and Cx3cr1 (Supplementary Figure 4**C**). To address both the empty droplet and multiplet detection issues, we built a droplet classification pipeline based on scANVI, a framework for classifying single-cell data via neural-network-based generative models (Xu et al., 2021). Using clusters with a high percentage of predicted multiplets from Scrublet as training

examples of multiplets, and clusters positive for known neuronal and non-neuronal marker genes as training examples of neurons and non-neuronal cells, we trained a predictive model to classify each droplet as a neuron, non-neuron, multiplet, or empty droplet (see Methods, Supplementary Figure 5**A**). This model performed with 97.6% accuracy on 10% of cells held out for testing, and yielded a database of 270,982 cortical cells (Supplementary Figure 5**B**). Inspection of the cells classified as empty droplets reveals that these droplets have lower transcript counts and higher mitochondrial gene ratios, consistent with other single-cell quality control pipelines (Supplementary Figure 5**D**). Critically, we discovered that non-neuronal clusters contained significantly more cells that had been previously removed by the Cell Ranger filtering method as compared to neuronal clusters ($P = 0.02$, 2-sided student t-test). In some clusters, such as $Gpr17^{+}$ $C1ql1^{+}$ oligodendrocytes and $Gper^{+}$ $Myl9^{+}$ vascular cells, we identified up to 85% more cells than what were recovered via Cell Ranger in some samples.

Using our combined experimental and computational pipeline for viral transcript recovery and droplet identification, we can recover a lower bound on the expected number of cells expressing each unique viral cargo within groups of cells in heterogeneous samples.

5.3.2 Single-cell RNA sequencing recapitulates AAV capsid cell-type-specific tropisms

As a first step, we validated our method by comparing the quantification of AAV transduction of major cell types via scRNA-seq to conventional IHC. For this purpose, we characterized the tropism of two previously reported AAV variants, AAV-PHP.eB (Chan et al., 2017) and AAV-CAP-B10 (Flytzanis et al., 2020) (Figure 23**A**). In total, four animals received single or dual retro-orbital injections of AAV-PHP.eB and/or AAV-CAP-B10 with 1.5×10^{11} viral genomes (vg) per variant. Co-injection of both variants served to test the ability of our approach to parallelize tropism profiling. By having each variant package a distinct fluorophore, tropism could be simultaneously assessed via multi-channel fluorescence and mRNA expression of the distinct transgene. After 3–4 weeks of expression, we harvested the brains and used one hemisphere for IHC and one hemisphere for scRNA-seq. To recover viral transcripts, we chose primers such that enough of the XFP sequence

was contained within the Illumina primer target sequences to differentiate the two variants (Table S **1**). For this comparison, we focused on the transduction rate for neurons (Rbfox3), astrocytes (S100b), and oligodendrocytes (Olig2). For IHC, a cell was classified as positive for the marker gene on the basis of antibody staining, and was classified as transduced on the basis of expression of the delivered fluorophore. For scRNA-seq, all cells that passed our QC pipeline were projected into a joint scVI latent space and clustered. To most closely match our imaging quantification, we considered all clusters that were determined to be positive for the respective marker gene as belonging to the corresponding cell type (see Methods). All clusters of the same marker gene were grouped together, and the transduction rate of the combined group of cells was determined using our viral transduction rate estimation method.

Our analysis of the scRNA-seq data demonstrates that the viral tropism biases across the three canonical marker genes are consistent with previous reports (Figure 23**C**) (Chan et al., 2017; Flytzanis et al., 2020). In contrast to AAV-PHP.eB, AAV-CAP-B10 preferentially targets neurons over astrocytes and oligodendrocytes. No marked discrepancies in viral tropism characterization were observed with single versus dual injections.

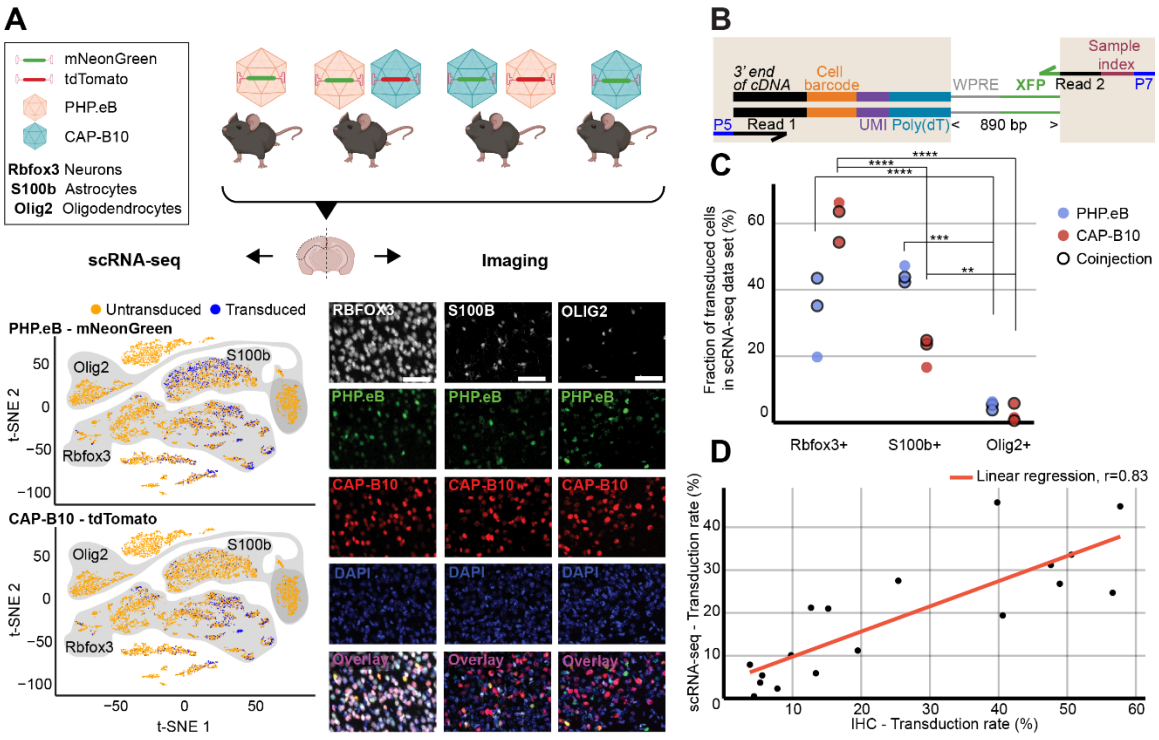


Figure 23. Comparison of viral tropism profiling with traditional IHC and scRNA-seq. (A) Overview of the experiment. Four animals were injected with 1.5×10^{11} viral genomes (vg) packaged in AAV-PHP.eB and/or AAV-CAP-B10. The bottom panels show a representative dataset collected from an animal that was co-injected with AAV-PHP.eB and AAV-CAP-B10. The left side displays the scRNA-seq data set in the lower dimensional t-SNE space, with cells colored according to transduction status. The shaded areas indicate clusters with high expression of the corresponding gene marker. The right side shows representative confocal images of cortical tissue labeled with IHC. Scale bar, 50 µm. **(B)** Viral transcript recovery strategy. The shaded areas highlight sequences added during library construction. **(C)** The fraction of the total number of transduced cells labeled as expressing the corresponding marker gene. For each AAV variant, the results of a two-way ANOVA with correction for multiple comparisons using Sidak's test are reported with adjusted P-values (****P ≤ 0.0001, ***P ≤ 0.001, and **P ≤ 0.01 are shown; P > 0.05 is not shown). **(D)** Comparison of transduction rates based on quantification via scRNA-seq or IHC. Transduction rate was calculated as (number of transduced cells in the group)/(total number of cells in the group). Each dot represents the transduction rate of neurons/Rbfox3+, astrocytes/S100b+, or oligodendrocytes/Olig2+ by AAV-PHP.eB or AAV-CAP-B10 in one animal. Histology data are averages across three brain slices per gene marker and animal. r indicates the Pearson correlation coefficient.

To quantify the similarity of the AAV tropism characterizations obtained with IHC and scRNA-seq, we directly compared the transduction rate of each AAV variant for every cell type and its corresponding marker gene (i.e., Rbfox3, S100b, or Olig2) as determined by each technique and noticed a good correlation (Figure 23**D**). Despite the different underlying biological readouts–protein

expression in IHC and RNA molecules in labeled cell types for scRNA-seq–the two techniques reveal similar viral tropisms.

5.3.3 Tropism profiling at transcriptomic resolution reveals AAV variant biases for neuronal subtypes

After validating our approach against the current standard of AAV tropism characterization (IHC imaging), we scrutinized the tropism of AAV-PHP.eB and AAV-CAP-B10 beyond the major cell types (Figure 24). Since AAV-CAP-B10 has increased neuronal bias relative to AAV-PHP.eB, we first sought to understand if there were neuronal subtypes that were differentially responsible for this bias. However, in-depth cell typing of transcriptomes collected from tissues with numerous and complex cell types, such as neurons in the brain, requires expert knowledge of the tissue composition, time to manually curate the data, and the availability of large datasets (Zeisel et al., 2018). To minimize the burden of manual annotation, computational tools have been developed that use previously-annotated single-cell databases to predict the cell type of cells in new, unannotated single-cell experiments, even across single-cell platforms (Cao et al., 2020; Tan and Cahan, 2019; Xu et al., 2021). We decided to leverage these tools and expanded our marker gene-based cell typing approach by having more complicated or well-established cell types be assigned based on annotations in a reference dataset (Supplementary Figure 5**A**). To this end, we again employed scANVI to construct a joint model of cells from our samples and cells from an annotated reference database. For this model, we used the Mouse Whole Cortex and Hippocampus 10x v2 dataset available from the Allen Brain Institute (Yao et al., 2021). Since this is a neuron-enriched dataset, we constructed the model using only the 109,992 cells in our dataset classified as neurons from our marker-based QC pipeline combined with the 561,543 neuronal cells from cortical regions from the reference database. We trained this model to predict to which of 14 neuron subtype groupings each cell belonged. We held out 10% of the data for testing: the model performed with 97.9% classification accuracy on the held-out data. We then applied the model to predict the neuron subtypes of our cells.

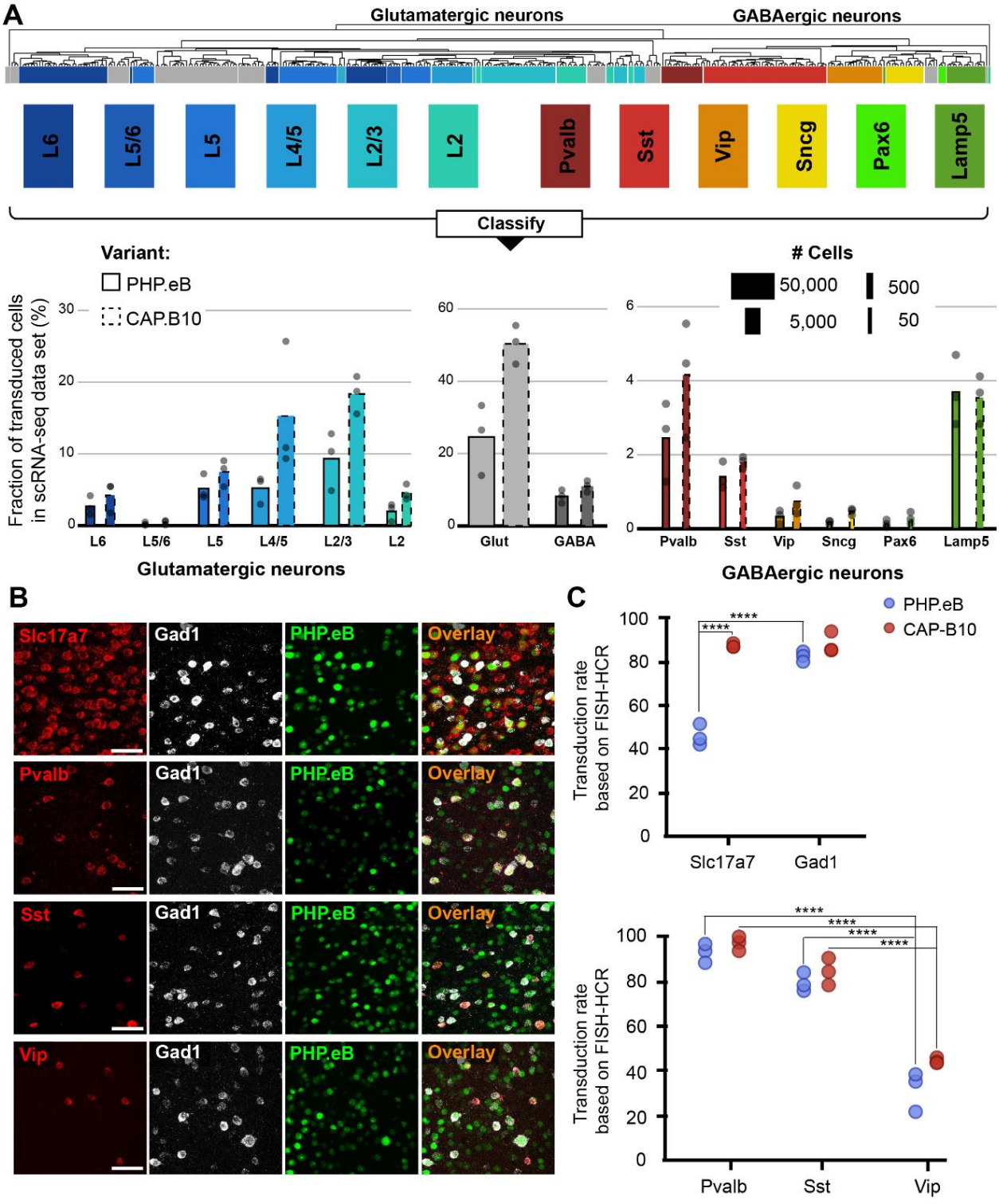

A
Glutamatergic neurons
GABAergic neurons
L6
L5/6
L5
L4/5
L2/3
L2
Pvalb
Sst
Vip
Sncg
Pax6
Lamp5
Classify
Variant:
PHP.eB
CAP.B10
Cells
50,000
5,000
500
50
Fraction of transduced cells in scRNA-seq data set (%)
Glutamatergic neurons
GABAergic neurons
B
Slc17a7
Gad1
PHP.eB
Overlay
Pvalb
Sst
Vip
C
PHP.eB
CAP-B10
Transduction rate based on FISH-HCR
Slc17a7
Gad1
Pvalb
Sst
Vip
****

Figure 24. In-depth AAV tropism characterization of neuronal subtypes at transcriptomic resolution. (A) Viral tropism profiling across neuronal sub types. Neuronal subtype annotations are predicted by a model learned from the Allen Institute reference dataset using scANVI (Xu et al., 2021; Yao et al., 2021). Each dot represents data from one animal injected with AAV-PHP.eB and/or AAV-CAP-B10. Bar width indicates the total number of cells of a particular cell type present in our dataset. **(B)** Representative confocal images of cortical tissue from an animal injected with 1.5×10^{11} vg of AAV-PHP.eB. Tissue was labeled with FISH-HCR for gene markers of glutamatergic neurons (Slc17a7) and GABAergic neurons (Gad1, Pvalb, Sst, Vip). AAV-PHP.eB shows the endogenous fluorescence of mNeonGreen. Scale bar, 50 µm. **(C)** Confirmation of viral tropism biases across neuronal subtypes using FISH-HCR (3 mice per AAV variant, 1.5×10^{11} vg dose). Dots represent the average values across three brain slices from one animal. Results from a two-way ANOVA with correction for multiple comparisons using Tukey's test is reported with adjusted P-values (****$P \leq 0.0001$; and $P > 0.05$ is not shown on the plot).

During our in-depth characterization, we discovered several previously unnoticed sub-cell-type biases for AAV-PHP.eB and AAV-CAP-B10 (Figure 24**A**). Starting at the top of our neuronal hierarchy, the fraction of transduced cells that were glutamatergic neurons was markedly reduced for AAV-PHP.eB compared with AAV-CAP-B10 ($P = 0.03$, 2-sided student t-test, corrected for 2 neuron subtype comparisons). Furthermore, Pvalb+ and Sst+ inhibitory neurons both represented a larger fraction of transduced cells than Vip+ inhibitory neurons with both variants (adjusted $P < 0.0001$, $P = 0.10$, respectively, two-way ANOVA with multiple comparison correction for inhibitory neuron subtypes using Tukey's method).

To confirm these tropism biases in neuronal subtypes with a traditional technique, we performed FISH-HCR for glutamatergic and GABAergic gene markers (Figure 24**B**) (Choi et al., 2014; Patriarchi et al., 2018). As indicated by our scRNA-seq data, AAV-CAP-B10, when compared with AAV-PHP.eB, has increased transduction efficiency of glutamatergic neurons (SLC17A7). Furthermore, FISH-HCR verified the downward trend in transduction efficiency from Pvalb+, to Sst+, to Vip+ neurons in both AAV variants (Figure 24**C**).

5.3.4 Pooled AAVs packaging barcoded cargo recapitulate the non-neuronal tropism bias of PHP.V1

To enable profiling viral variants in parallel without needing distinct transgenes per variant, we established a barcoding strategy whereby we package AAV variants with the same transgene and

regulatory elements but with short, distinguishing nucleotide sequences within the 3' UTR (Figure 25**A**). To verify that this barcoding strategy can recover tropisms consistent with our previous transgene-based capsid-identification strategy, we performed a set of experiments to re-characterize the tropism of AAV-PHP.eB in parallel with that of the recently developed AAV-PHP.V1, which has increased specificity for vascular cells over AAV-PHP.eB (Ravindra Kumar et al., 2020).

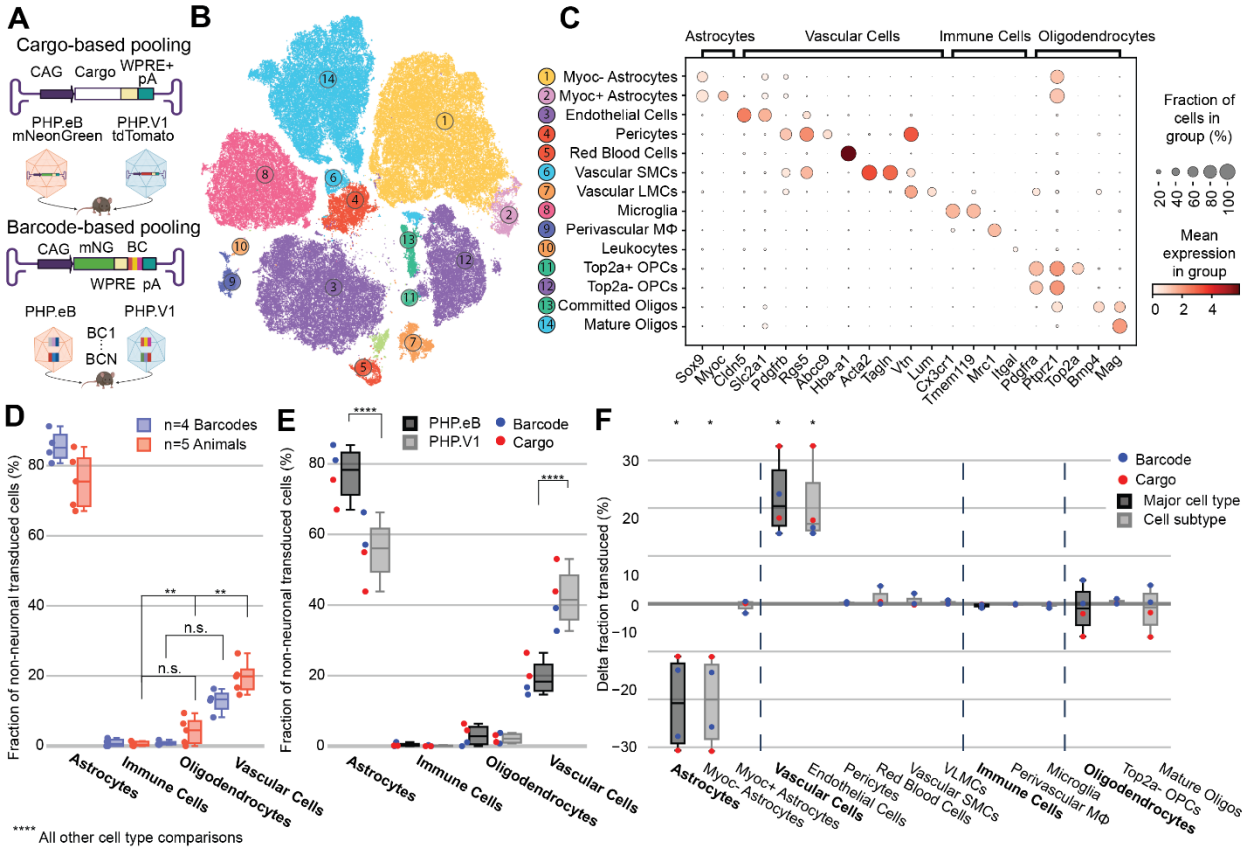


Figure 25. Barcoded co-injected rAAVs reveal the non-neuronal tropism bias of AAV-PHP.V1. (A) Experimental design for comparing barcode vs cargo-based tropism profiling. Animals received dual injections of AAV-PHP.eB and AAV-PHP.V1, carrying either distinct fluorophores (cargo) or the same fluorophore with distinct barcodes. **(B)** t-SNE projection of the single-cell Variational Inference (scVI) latent space of cells and their cell type classification of the 169,265 non-neuronal cells across all our samples. Each number corresponds to the cell type labeled in **C**. **(C)** Marker genes used to identify non-neuronal cell types. Darker colors indicate higher mean expression, and dot size correlates with the abundance of the gene in that cell type. **(D)** The distribution of non-neuronal cells expressing transcripts from AAV-PHP.eB across 4 barcodes within one animal (blue) and across 5 animals (red). All animals received dual injections, with one of the vectors being 1.5 x 1011 vg of PHP.eB carrying CAG-mNeonGreen. The y-axis represents the fraction of transduced non-neuronal cells that are of the specified cell type. Only the non-significant

comparisons between cell types in a two-way ANOVA with correction for multiple comparisons using Tukey's test are reported. All other cell-type comparisons within a paradigm were significant at $P \leq 0.0001$. **(E)** The distribution of non-neuronal cells expressing transcripts from AAV-PHP.eB (black) and AAV-PHP.V1 (gray). Results from the different experimental paradigms are combined. Results shown are from a two-way ANOVA with correction for multiple comparisons using Sidak's test comparing transduction by AAV-PHP.eB to AAV-PHP.V1 for each cell type, with adjusted P-values (****$P \leq 0.0001$ is shown; $P > 0.05$ is not shown). **(F)** Within-animal difference in the fraction of cells transduced with AAV-PHP.V1 relative to AAV-PHP.eB across four animals, two from each experimental paradigm. For each cell type in each sample, the combined 2-proportion z score for the proportion of that cell type transduced by AAV-PHP.V1 vs AAV-PHP.eB is reported. Cell types with fewer than 2 cells transduced by both variants were discarded. Z scores were combined across multiple animals using Stouffer's method and corrected for multiple comparisons. Cell-type differences with an adjusted P-value below 0.05 are indicated with *.

We produced AAV-PHP.eB carrying CAG-mNeonGreen and AAV-PHP.V1 carrying either CAG-mRuby2 or CAG-tdTomato. Additionally, we produced AAV-PHP.eB and AAV-PHP.V1 both carrying CAG-mNeonGreen with 7-nucleotide barcodes 89 bp upstream of the polyadenylation start site such that they did not interfere with the WPRE. We ensured each barcode had equal G/C content, and that all barcodes were Hamming distance 3 from each other (Table S 5). Each of the barcoded variants was packaged with multiple barcodes that were pooled together during virus production. Four animals received a retro-orbital co-injection of 1.5×10^{11} vg/each of AAV-PHP.V1 and AAV-PHP.eB. Two animals received viruses carrying separate fluorophores (cargo-based), and two animals received viruses carrying the barcoded cargo (barcode-based). For amplification of the viral cDNA in the animals receiving the barcoded cargo, we used primers closer to the polyA region such that the sequencing read covered the barcoded region (Table S 1). During the single-cell sequencing dissociation and recovery, one of our dissociations resulted in low recovery of neurons (Supplementary Figure 5**C**); thus, we investigated only non-neuronal cells for this experiment.

Despite variability in the total transgene RNA content between barcodes of the same variant (Supplementary Figure 6**A**), the estimated percent of cells expressing the transgene within each cell type was consistent between barcodes within a single animal, with standard deviations ranging from 0.003 to 0.058 (Supplementary Figure 7**A**). Our analysis of both the barcode-based animals and cargo-based animals shows the same bias in non-neuronal tropism, with AAV-PHP.eB significantly preferring astrocytes over oligodendrocytes, vascular cells, and immune cells (Figure 25**D**). Interestingly, our analysis also revealed that the variance between barcodes within an animal was

less than the variance between animals, even when controlling for cargo and dosage (P = 0.021, Bartlett's test, P-values combined across all variants and cell types using Stouffer's method, weighted by transduced cell type distribution).

Next, we investigated the distribution of cells transduced by AAV-PHP.eB vs AAV-PHP.V1 in the major non-neuronal cell types across both barcode-based and cargo-based paradigms (Figure 25**E**). The single-cell tropism data confirms the previously-established finding that AAV-PHP.V1 has a bias toward vascular cells relative to AAV-PHP.eB. Additionally, we uncovered that this is coupled with a bias away from astrocytes relative to AAV-PHP.eB, but that transduction of oligodendrocytes and immune cells did not differ between the variants. To investigate for a specific effect of the barcoding strategy, we performed a three-way ANOVA across the variant, cell type, and experimental paradigm factors. We found that the cell type factor accounted for 89.25% of the total variation, the combined cell type + variant factor accounted for 7.7% of the total variation, and the combined cell type + experimental paradigm factor accounted for only 2.0% of the total variation, confirming our hypothesis that barcoded pools can recover tropism with minimal effect.

5.3.5 Relative tropism biases reveal non-neuronal subtypes with reduced AAV transduction

To further characterize the tropism biases of AAV-PHP.V1 and expand our method to less well-established cell hierarchies, we explored the non-neuronal cell types in our dataset. Since the Allen Brain Institute reference database that we used to investigate neuronal tropism was enriched for neurons, it does not contain enough non-neuronal cells to form a robust non-neuronal cell atlas. Our combined dataset consists of 169,265 non-neuronal cells, making it large enough to establish our own non-neuronal cell clustering. Thus, we performed an additional round of automatic clustering on the cells classified as non-neuronal in our combined dataset, and identified 12 non-neuronal cell subtypes based on previously established marker genes (Figure 25**B, C**, Table S 2).

Most cell subtypes had multiple clusters assigned to them, which suggested there may be additional subtypes of cells for which we did not find established marker genes. To determine whether any of these clusters delineated cell types with distinct transcriptional profiles, we investigated the

probability of gene expression in each cluster compared to the other clusters of the same cell subtype (see Methods). Our approach determined two subclusters of pericytes, astrocytes, and oligodendrocyte precursor cells (OPCs). Both clusters of pericytes had strong expression of canonical pericytes marker genes Rgs5, Abcc9, and Higd1b. However, one of the clusters had no marker genes that made it distinct from the other pericyte cluster, nor from endothelial cells. Consistent with previous reports, this suggests that this cluster could be pericytes contaminated with endothelial cell fragments, and thus was not considered for further analysis (He et al., 2016; Vanlandewijck et al., 2018; Yang et al., 2021). Two distinct groups of astrocytes were detected, one of which had unique expression of Myoc and Fxyd6. Finally, one of the clusters of OPCs were uniquely expressing Top2a, Pbk, Spc24, Smc2, and Lmnb1. Using these new marker genes, we expanded our non-neuronal cell taxonomy to 14 cell types, now including Myoc+ and Myoc- astrocytes, and Top2a+ and Top2a- OPCs.

Given our finding that inter-sample variability exceeds intra-sample variability, we established a normalization method for comparing transduction biases between variants co-injected into the same animal. This normalization–calculating the difference in the fraction of transduced cells between variants–captures the relative bias between variants, instead of the absolute tropism of a single variant (see Methods). By considering the relative bias between variants, we are able to interrogate tropism in a way that is more robust to inter-sample variability that arises from different distributions of recovered cells, expression rate of delivered cargo, and success of the injection. Using this normalization method, we evaluated the non-neuronal cell type bias of AAV-PHP.V1 relative to AAV-PHP.eB in both the cargo-based animals and the barcode-based animals across our non-neuronal cell-type taxonomy (Figure 25**F**). We discovered that the bias of AAV-PHP.V1 for vascular cells is driven by an increase in transduction of endothelial cells, but not pericytes. Similarly, AAV-PHP.V1's bias away from astrocytes is driven by a decrease in transduction of Myoc- astrocytes, but not Myoc+ astrocytes. Further inspection of the transduction of pericytes and Myoc+ astrocytes revealed that pericytes are not highly transduced by any of the AAVs tested in this work, and that Myoc+ astrocytes have both lower viral transcript expression and lower abundance than Myoc-

astrocytes, and thus do not contribute significantly to tropism (Supplementary Figure 5**B**, Supplementary Figure 8**B**).

5.3.6 Single-cell RNA sequencing reveals early cell-type-specific responses to IV administration of AAV-PHP.eB that return to baseline by 3.5 weeks

To investigate the temporal cell-type-specific transcriptional effects of systemic AAV delivery and cargo expression, we performed a single-cell profiling experiment comparing animals injected with AAV to saline controls. We injected four male mice with AAV-PHP.eB (1.5×10^{11} vg) carrying mNeonGreen, and performed single-cell sequencing on two mice three days post-injection (3 DPI) and two mice twenty-five days post-injection (25 DPI). These time points were chosen based on previous work showing MHC presentation response peaking around day seven and transgene response peaking around day 30 (Lowenstein et al., 2007). The two saline control mice were processed 3 DPI. We then analyzed differential gene expression for each cell type between injected animals and controls using DESeq2 (Table S 7). Of note, we excluded cell types with less than 50 cells in each sample, and excluded leukocytes and red blood cells given the risk of their presence due to dissociation rather than chemokine mediated infiltration. Additionally, we collapsed subtypes of excitatory neurons, inhibitory neurons, and OPCs to have greater than 50 cells for differential analysis. We estimated viral transduction rate of AAV-PHP.eB using its delivered cargo, mNeonGreen, across cell types and time points. We identified that Myoc- Astrocytes have significantly higher estimated transduction rate at 25 DPI compared to 3DPI (adjusted P-value = 0.0003, two-way ANOVA with multiple comparison correction using Sidak's method). It is also worth noting that endothelial cells have a similar transduction rate between the time points in both animals, while one of the animals at 25 DPI exhibited higher transduction in neurons (Figure 26**A**). The number of statistically relevant genes between the injected and control group (adjusted P-value < 0.05, DESeq2) were highest in pericytes (26 genes), endothelial cells (76 genes), and excitatory neurons (45 genes) at 3 DPI (Figure 26**B**). At day twenty-five, only excitatory neurons had greater than 10 genes (14 genes total) differentially expressed (adjusted P-value < 0.05, DESeq2).

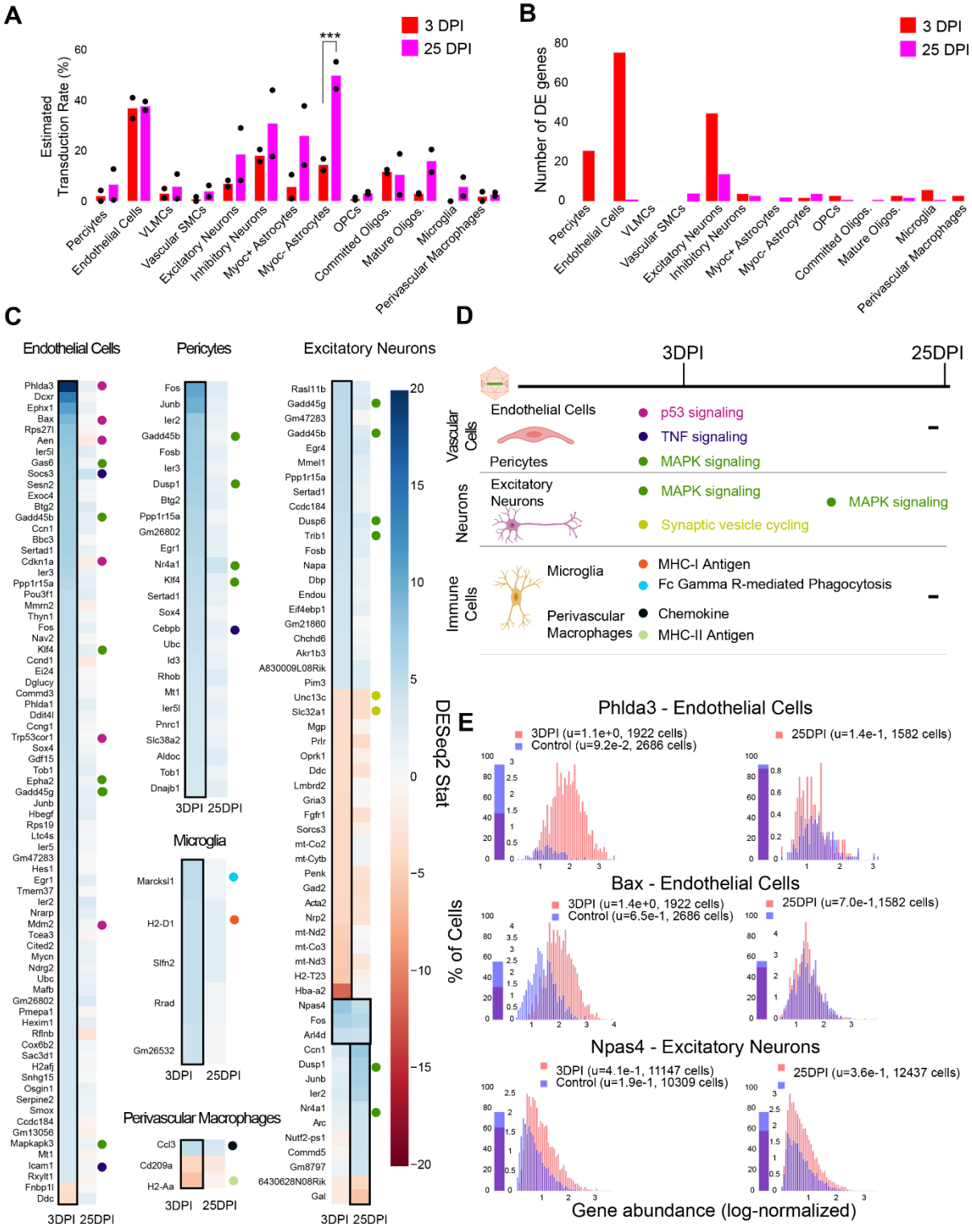

A
3 DPI
25 DPI
Estimated Transduction Rate (%)
***
Pericytes
Endothelial Cells
VLMCs
Vascular SMCs
Excitatory Neurons
Inhibitory Neurons
Myoc+ Astrocytes
Myoc- Astrocytes
OPCs
Committed Oligos.
Mature Oligos.
Microglia
Perivascular Macrophages
B
Number of DE genes
C
Endothelial Cells
Pericytes
Excitatory Neurons
Microglia
Perivascular Macrophages
3DPI 25DPI
DESeq2 Stat
% of Cells
D
3DPI
25DPI
Vascular Cells
Endothelial Cells
Pericytes
p53 signaling
TNF signaling
MAPK signaling
Neurons
Excitatory Neurons
MAPK signaling
Synaptic vesicle cycling
MAPK signaling
Immune Cells
Microglia
Perivascular Macrophages
MHC-I Antigen
Fc Gamma R-mediated Phagocytosis
Chemokine
MHC-II Antigen
E
Phlda3 - Endothelial Cells
3DPI (u=1.1e+0, 1922 cells)
Control (u=9.2e-2, 2686 cells)
25DPI (u=1.4e-1, 1582 cells)
Bax - Endothelial Cells
3DPI (u=1.4e+0, 1922 cells)
Control (u=6.5e-1, 2686 cells)
25DPI (u=7.0e-1,1582 cells)
Npas4 - Excitatory Neurons
3DPI (u=4.1e-1, 11147 cells)
Control (u=1.9e-1, 10309 cells)
25DPI (u=3.6e-1, 12437 cells)
Gene abundance (log-normalized)

Figure 26. Single-cell gene expression profiling finds cell-type-specific responses to AAV transduction in vascular cells and excitatory neurons. (A) Estimated transduction rate (%) of mNeonGreen cargo at three and twenty-five days post-injection (DPI). Results from a two-way ANOVA with correction for multiple comparisons using Sidak's method is reported with adjusted P-values (***$P \leq 0.001$; and $P > 0.05$ is not shown on the plot). **(B)** Number of differentially expressed genes (adjusted P-value < 0.05, DESeq2) at 3 DPI and 25 DPI across 2 animals. **(C)** Differentially expressed genes across the two time points in endothelial cells, pericytes, microglia, perivascular macrophages, and excitatory neurons. Color indicates DESeq2 test statistic with red representing downregulation and blue representing upregulation. Genes outlined by a black rectangle are determined to have statistically significant differential expression compared to controls (adjusted P-value < 0.05, DESeq2). Colored circles adjacent to each gene indicate the corresponding pathway presented in D. **(D)** A summary of corresponding pathways in which the differentially regulated genes in (C) are involved across the time points. **(E)** Distribution of p53 signaling transcripts in endothelial cells (animals are combined) and an example of a gene upregulated in both 3 and 25 DPI in excitatory neurons.

We found that endothelial cells had the most acute response at 3 DPI with pathways such as p53, MAPK, and TNF signaling notably impacted. A significant upregulation of Phlda3 and its effectors Bax, Aen, Mdm2, and Cdkn1a, all involved in the p53/Akt signaling pathway, was present (Figure 26**C,E**) (Ferreira and Nagai, 2019; Ghouzzi et al., 2016). Of relevance, we also detected Trp53cor1/LincRNA-p21, responsible for negative regulation of gene expression (Amirinejad et al., 2020), upregulated in endothelial cells at 3 DPI. Other examples of upregulated genes relevant to inflammation and stress response in vascular cells include the suppressor of cytokine signaling protein Socs3 (Baker et al., 2009), and Mmrn2, responsible for regulating angiogenesis in endothelial cells (Lorenzon et al., 2012). Expression of Socs3 and Icam1, which are upregulated in endothelial cells at 3 DPI, and Cepbp, which is upregulated in pericytes at 3 DPI, have all been linked to TNF signaling (Burger et al., 1997; Cao et al., 2018; Li et al., 2020). We have also observed genes linked to MAPK signaling upregulated in endothelial cells, such as Gas6, Epha2, and Mapkapk3, and Klf4 in both endothelial cells and pericytes (Chen et al., 1997; Macrae et al., 2005; Riverso et al., 2017).

In brain immune cells, we observe a few substantial changes in genes pertaining to immune regulation at 3DPI which vanish at 25 DPI. For example, we observe an upregulation of MHC-I gene H2-D1 at 3 DPI in microglia, which then stabilizes back to control levels at 25 DPI (Figure 26**C**). Marcksl1, previously reported as a gene marker for neuroinflammation induced by alpha-synuclein (Sarkar et al., 2020), also shows upregulation at 3 DPI. We did not observe significant differences in pro-inflammatory chemokines, Ccl2 and Ccl5, which are related to breakdown of the blood-brain

barrier via regulation of tight-junction proteins and recruitment of peripheral leukocytes (Gralinski et al., 2009). Ccl3, responsible for infiltration of leukocytes and CNS inflammation (Chui and Dorovini-Zis, 2010), was upregulated in perivascular macrophages in 3 DPI and diminished back to control levels at 25 DPI (Figure 26**C**). In contrast, Cd209a, a gene previously identified as critical for attracting and activating naïve T Cells (Franchini et al., 2019), was downregulated at 3 DPI.

Interestingly, we found that excitatory neurons had changes in genes across both 3 DPI and 25 DPI. MHC-Ib H2-T23, which is involved in the suppression of CD4+ T cell responses (Ohtsuka et al., 2008), is downregulated at 3 DPI. Meanwhile, the growth arrest genes, Gadd45g and Gadd45b (Vairapandi et al., 2002), are upregulated. Genes involved in synaptic vesicle cycling, such as Unc13c (Palfreyman and Jorgensen, 2017) and Slc32a1 (Taoufiq et al., 2020), are also downregulated at 3 DPI. Some genes remain upregulated throughout the study, such as Npas4, responsible for regulating excitatory-inhibitory balance (Spiegel et al., 2014). Genes implicated in MAPK signaling were upregulated – such as Gadd45b/g, Dusp6, and Trib1 at 3 DPI, and Dusp1 and Nr4a1 at 25 DPI (Muhammad et al., 2018; Ollila et al., 2012; Pérez-Sen et al., 2019; Salvador et al., 2013; Zhang and Yu, 2018). Gadd45b, Dusp1, Nr4a1 were also upregulated in pericytes and Gadd45b/g in endothelial cells (Figure 26**C**).

Immediate early genes such as Ier2 (Kodali et al., 2020) were upregulated across pericytes, endothelial cells, inhibitory neurons, and OPCs at 3 DPI, while Fos, Ier2, Junb, and Arc were prominent in excitatory neurons at 25 DPI.

By investigating the gene expression differences in subpopulations of cells post-injection, we found that vascular cells such as endothelial cells and pericytes upregulate genes linked to p53, MAPK, and TNF signaling pathways at 3 DPI (Figure 26**D**). Immune cells such as microglia and perivascular macrophages upregulate genes involved in chemokine signaling, MHCI/II antigen processing, and Fc Gamma R-Mediated Phagocytosis (Zhang et al., 2021) at 3 DPI (Figure 26**D**). Excitatory neurons are the only cell type with genes implicated in the same pathway (MAPK signaling) upregulated across both of the time points (3 DPI, 25 DPI).

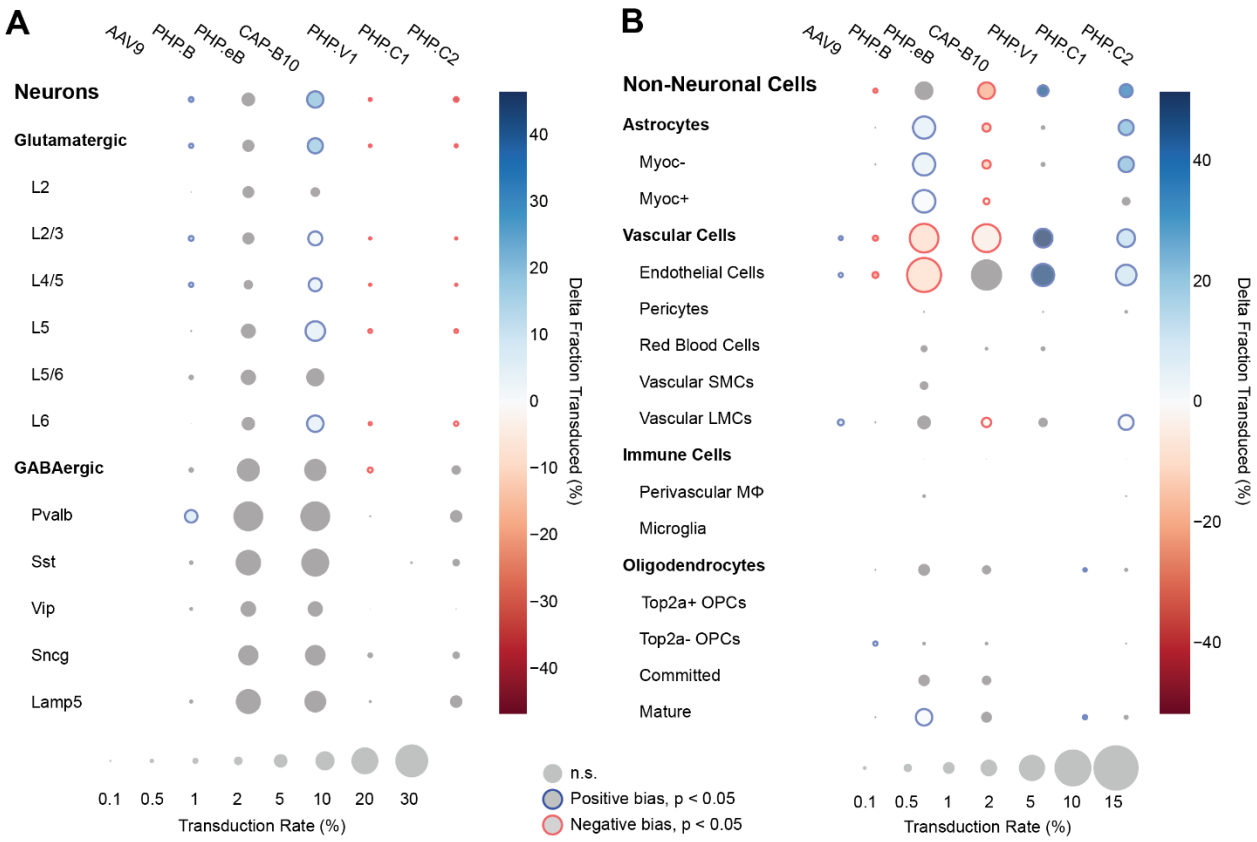


Figure 27. Single animal injections of multiple barcoded rAAVs enables deep, parallel characterization. (A, B) Relative cell type tropism of 7 co-injected rAAVs for neuronal **(A)** and non-neuronal **(B)** cell types. The color scale indicates the difference in transduction bias of a variant relative to all other variants in the pool. The area of each circle scales linearly with the fraction of cells of that type with viral transcripts above background. For each variant and cell type, a 2-proportion z score was calculated to compare the number of cells of that type transduced by that variant relative to all other variants combined. Z scores were combined across two single-cell sequencing aliquots using Stouffer's method, and corrected for multiple comparisons. Cell types with fewer than 10 transduced cells in either the variant or variants compared against were discarded. Only cell-type biases at an adjusted P-value < 0.05 are colored; otherwise they are grayed out.

5.3.7 Larger pools of barcoded AAVs recapitulate complex tropism within a single animal

To showcase the capabilities of parallel characterization, we next designed a 7-variant barcoded pool that included the three previously characterized variants (AAV-PHP.eB, AAV-CAP-B10, and AAV-PHP.V1), AAV9 and AAV-PHP.B controls, and two additional variants, AAV-PHP.C1 and AAV-PHP.C2. For simplification of cloning and virus production, we designed a plasmid, UBC-mCherry-

AAV-cap-in-cis, that contained both the barcoded cargo, UBC-mCherry, and the AAV9 capsid DNA (Supplementary Figure 2**B**). We assigned three distinct 24 bp barcodes to each variant (Table S 5). Each virus was produced separately to control the dosage, and 1.5 x 10^{11} vg of each variant was pooled and injected into a single animal.

After 3 weeks of expression, we performed single-cell sequencing on extracted cortical tissue. To increase the number of cells available for profiling, we processed two aliquots of cells, for a total of 36,413 recovered cells. To amplify the viral transcripts, we used primers that bind near the 3' end of mCherry such that the barcode was captured in sequencing (Table S 1).

Using our cell typing and viral transcript counting methods, we investigated the transcript counts and transduction bias of the variants in the pool. Compared with our previous profiling experiments, the number of UBC-mCherry viral transcripts detected per cell was significantly lower than CAG-mNeonGreen-WPRE and CAG-tdTomato (adjusted $P < 0.0001$, $P=0.0445$, respectively, two-way ANOVA with multiple comparison correction using Tukey's method) and shifted towards vascular cells (adjusted $P < 0.0001$, $P=0.0008$, respectively, two-way ANOVA with multiple comparison correction using Tukey's method) (Supplementary Figure 6**B, C**). Next, we looked at the transduction rate difference for each variant compared with the rest of the variants in the pool for each cell type in our taxonomy (Figure 27**A, B**). Despite the lower expression rate and bias shift, the transduction rate difference metric captured the same tropism biases for AAV-CAP-B10 and AAV-PHP.V1 as determined from our previous experiments. AAV-CAP-B10 showed enhanced neuronal targeting relative to other variants in the pool, with this bias coming specifically from an increase in the transduction of glutamatergic neurons. All five variants with transcripts detected in neurons showed a decreased transduction rate in Vip+ neurons relative to other GABAergic neuronal subtypes (Supplementary Figure 8**C**). AAV-PHP.eB showed enhanced targeting of astrocytes (+6.2%, $P = 1.4 \times 10^{-8}$, 2-proportion z-test, multiple comparison corrected with Benjamini/Hochberg correction), and AAV-PHP.V1 showed strong bias for vascular cells (+51.6%, $p = 1.7 \times 10^{-43}$). In addition to confirming all our existing hypotheses, we were able to identify biases for the previously reported AAV-PHP.C2, which has not been characterized in depth. This variant, which was reported

as having a non-neuronal bias similar to AAV-PHP.V1, showed significant transduction bias not only toward vascular cells (+13.6%, $P = 8.3 \times 10^{-6}$), but also toward astrocytes (+24.0%, $P = 1.6^{-30}$), and a bias away from neurons (−38%, $p = 4.1 \times 10^{-32}$).

5.4 Discussion

The advent of NGS has enabled screening of large libraries of AAV capsids in vivo by extracting viral DNA from relevant tissue followed by sequencing of capsid gene inserts or DNA barcodes corresponding to defined capsids. To date, NGS-based screening has been successfully applied to libraries created by peptide insertions (Davidsson et al., 2019; Körbelin et al., 2016a), DNA shuffling of capsids (De Alencastro et al., 2020; Herrmann et al., 2019; Paulk et al., 2018), and site-directed mutagenesis (Adachi et al., 2014). Although these NGS-based strategies allow the evolution of new AAV variants with diverse tissue tropisms, it has been difficult to obtain a comprehensive profiling for multiple variants across cell types, which is of utmost importance in organs with complex cell-type compositions, such as the brain (Deverman et al., 2016; Ravindra Kumar et al., 2020; Tasic et al., 2016, 2018; Zeisel et al., 2018). Towards this end, techniques such as IHC, fluorescent in situ RNA hybridization (Chen et al., 2015; Choi et al., 2014; Femino et al., 1998; Lubeck et al., 2014; Shah et al., 2016a, 2016b) or in situ RNA sequencing (Ke et al., 2013; Lee et al., 2014; Wang et al., 2018) can be employed. Several limitations make it challenging to apply these techniques as high-throughput, post-selection AAV tropism profiling methods. First, the limits of optical resolution and the density of transcripts in single cells pose challenges for full in situ transcriptome analysis and, until recently, have restricted the total number of simultaneously measured genes in single cells within tissue to several hundred (Ke et al., 2013; Lee et al., 2014; Liao et al., 2020; Shah et al., 2016a; Wang et al., 2018). By contrast, scRNA-seq with the 10x Genomics Chromium system enables detection of over 4000 genes per cell (Yao et al., 2021), fast transcriptomic analysis, and multiplexing across different tissue types (McGinnis et al., 2019; Stoeckius et al., 2018). Furthermore, the method is already widely used by the research community which can help with adoption of our proposed pipelines. Although droplet-based scRNA-seq methods lose spatial information during the dissociation procedure, analysis packages have been developed that can infer

single-cell localization by combining scRNA-seq data with pre-existing information from ISH-based labeling for specific marker genes (Achim et al., 2015; Durruthy-Durruthy et al., 2015; Halpern et al., 2017; Nitzan et al., 2019; Satija et al., 2015; Stuart et al., 2019). Therefore, scRNA-seq techniques have great potential to rapidly profile the tropism of multiple AAV variants in parallel across several thousand cells defined by their entire transcriptome.

Here, we established an experimental and data-analysis pipeline that leverages the capabilities of scRNA-seq to achieve simultaneous characterization of several AAV variants across multiplexed tissue cell types within a single animal. To differentiate multiple AAV capsid variants in the sequencing data, we packaged variants with unique transgenes or the same transgene with unique barcodes incorporated at the 3' end. We added standard Illumina sequencing primer recognition sites (Read 2) to the viral transcripts using PCR amplification such that the barcoded region could be consistently read out from the Illumina sequencing data. Our computational pipeline demultiplexes viral reads found in the transcriptome according to which matching sequence is most abundant in a separate amplified viral transgene library. Comparing the distribution of viral transcripts by cell type to a null model of empty droplets, we could then determine the cell-type biases.

Our platform has corroborated the tropism of several previously characterized AAV variants and has provided more detailed tropism information beyond the major cell types. The fraction of transduced cells that are glutamatergic neurons was found to be markedly reduced for AAV-PHP.eB when compared with AAV-CAP-B10. Furthermore, within all the variants we tested, both Pvalb+ and Sst+ inhibitory neurons have greater transduction rates than Vip+ neurons. This bodes well for delivery to Pvalb+ neurons, which have been implicated in a wide range of neuro-psychiatric disorders (Ruden et al., 2021), and suggests Vip+ interneurons, which have recently been identified as being a sufficient delivery target for induction of Rett syndrome-like symptoms, as a target for optimization (Mossner et al., 2020). Awareness of neuronal subtype biases in delivery vectors is critical both for neuroscience researchers and for clinical applications. Dissection of neural circuit function requires understanding the roles of neuronal subtypes in behavior and disease and relies on successful and sometimes specific delivery of transgenes to the neuronal types under study (Bedbrook et al., 2018).

We further discovered that the vascular bias of AAV-PHP.V1 originates from its transduction bias towards endothelial cells. Interestingly, this is the only cell type we detected expressing Ly6a (Supplementary Figure 9), a known surface receptor for AAV variants in the PHP.B family (Batista et al., 2020; Hordeaux et al., 2019; Huang et al., 2019). Given AAV-PHP.V1's sequence similarity to AAV-PHP.B and its tropism across mouse strains, this pattern suggests that AAV-PHP.V1 transduction may also be Ly6a-mediated. Finding such associations between viral tropism and cell-surface membrane proteins also suggests that full transcriptome sequencing data may hold a treasure trove of information on possible mechanisms of transduction of viral vectors.

We also revealed that AAV-PHP.C2 has a strong, broad non-neuronal bias toward both vascular cells and astrocytes. AAV-PHP.C2 also transduces BALB/cJ mice, which do not contain the Ly6a variant that mediates transduction by PHP.B family variants (Hordeaux et al., 2019). This suggests that PHP.C2 may be the most promising candidate from this pool for researchers interested in delivery to non-neuronal cells with minimal neuronal transduction both in C57BL/6J mice and in strains and organisms that do not have the Ly6a variant.

All our tested variants with non-neuronal transduction have lower expression in Myoc+ astrocytes and pericytes. Astrocytes expressing Myoc and Gfap, which intersect in our data (Supplementary Figure 9), have been previously identified as having reactive behavior in disease contexts, making them a target of interest for research on neurological diseases (Perez-Nievas and Serrano-Pozo, 2018; Wu et al., 2017). Similarly, pericytes, whose dysfunction has been shown to contribute to multiple neurological diseases, may be an important therapeutic target (Blanchard et al., 2020; Liu et al., 2020; Montagne et al., 2020). Both of these cell types may be good candidates for further AAV optimization, but may have been missed with marker gene-based approaches. In both AAV characterization and neuroscience research efforts, different marker genes are often used for astrocyte classification – sometimes more restrictive genes such as Gfap, and other times more broadly expressing genes such as S100b or Aldh1l1 (Yang et al., 2011; Zhang et al., 2019). Similarly, defining marker genes for pericytes is still an active field (He et al., 2016; Yang et al., 2021). Given the constraints of having to choose specific marker genes, it is difficult for staining-based

characterizations to provide tropism profiles that are relevant for diverse and changing research needs. This highlights the importance of using unbiased, full transcriptome profiling for vector characterization.

We have shown that our combined experimental and computational platform is able to recover transduction biases and profile multiple variants in a single animal, even amidst the noise of ambient RNA. We have further shown that our method is robust to the variability inherent in delivery and extraction from different animals, with different transgenes, and with different regulatory elements. For example, we discovered lower overall expression from vectors carrying UBC-mCherry compared with CAG-mNeonGreen-WPRE. Such differences are not surprising since the WPRE is known to increase RNA stability and therefore transcript abundance (Johansen et al., 2003). Furthermore, the shift in cell-type bias may come from the UBC promoter, as even ubiquitous promoters such as CAG and UBC have been shown to have variable levels of expression in different cell types (Qin et al., 2010). Despite these biases, looking at the differences in transduction between variants delivering the same construct within an individual animal reveals the strongest candidate vectors for on-target and off-target cell types of interest. While we show that our method can profile AAVs carrying standard fluorescent cargo, caution is needed when linking differences in absolute viral tropism to changes in capsid composition alone without considering the contribution of the transgene and regulatory elements. Therefore, for more robust and relative tropism between variants, we found it beneficial to use small barcodes and co-injections of pools of vectors. Our scRNA-seq-based approach is not restricted to profiling capsid variants, but can be expanded in the future to screen promoters (Chuah et al., 2014; Jüttner et al., 2019; Rincon et al., 2015), enhancers (Hrvatin et al., 2019; Mich et al., 2020), or transgenes (Gustafsson et al., 2004; Shirley et al., 2020), all of which are essential elements requiring optimization to improve gene therapy.

Finally, we have used scRNA-seq to understand how intra-orbital administration of AAV-PHP.eB affects the host cell transcriptome across distinct time points. Results from our study show genes pertaining to the p53 pathway in endothelial cells are differentially expressed 3 days after injection. The highest number of differentially expressed genes being in endothelial cells suggests that vascular

cells could be the initial responders to viral transduction and expression of the transgene. This is supported by Kodali et al., who have shown that endothelial cells are the first to elicit a response to peripheral inflammatory stimulation by transcribing genes for proinflammatory mediators and cytokines (Kodali et al., 2020). With regards to p53 differentially expressed genes, Ghouzzi, et al. have also shown that the genes Phdla3, Aen, and Cdkn1a were upregulated in cells infected with ZIKA virus, signifying genotoxic stress and apoptosis induction (Ghouzzi et al., 2016). Upregulation of genes such as Bax and Cdkn1a in our data hint at an initiation of apoptosis and cell cycle arrest, respectively, in response to cellular stress induced by viral transduction (Ferreira and Nagai, 2019; Zamagni et al., 2020). The reduction in the number of differential expressed genes across all cells (Figure 26**B**) at day twenty-five imparts that the initial inflammatory responses did not escalate. Downregulation of Cd209a gene in perivascular macrophages in our data further implies that the AAV-PHP.eB infection did not necessitate a primary adaptive immune response. Additionally, antigen presenting genes, such as H2-D1, returning back to control expression levels and a lack of proinflammatory cytokines being upregulated supports that the event of infiltration of peripheral leukocytes is unlikely, in agreement with prior studies (Chamberlin et al., 1998; McCown et al., 1996). Upregulation of genes such as Gadd45g, Gadd45b, and Ppp1r15a suggest that neurons are turning on stress-related programs as an early response to encountering the virus. Genes such as Nr4a1 and Dusp1, which play a role in the MAPK pathway, indicate sustained stress response even at day 25. Based on prior studies, we speculate that the genes that are differentially expressed at day 25 in the excitatory neurons are due to transgene expression and not due to the virion (Lowenstein et al., 2007). It is important to note that the findings discussed here are specific to the rAAV, transgene, and dosage. Our results highlight the power of single-cell profiling in being able to ascertain cell-type-specific responses at an early time point post-injection.

In summary, our platform could aid the gene therapy field by allowing more thorough characterization of existing and emerging recombinant AAVs by helping uncover cellular responses to rAAV-mediated gene therapy, and by guiding the engineering of novel AAV variants.

5.5 Acknowledgements

We thank the Gradinaru and Thomson labs for helpful discussions, Allan-Hermann Pool for advice on the mouse brain tissue dissociation procedure, Jeff Park for advice on 10X Genomics Chromium single-cell library preparation, Min Jee Jang for help in designing probes and troubleshooting FISH-HCR, and Ben Deverman and Ken Chan for early discussions on strategy. This work was supported by the NIH Pioneer DP1OD025535, Beckman Institute for CLARITY, Optogenetics and Vector Engineering Research at Caltech, the Single-Cell Profiling and Engineering Center (SPEC) in the Beckman Institute at Caltech, and the Curci Foundation. V.G. and M.T. are Heritage Principal Investigators supported by the Heritage Medical Research Institute.

5.6 Author contributions

D.B., M.A., T.D., and V.G. conceived the project and designed the experiments. S.C. and M.T. provided critical single-cell RNA sequencing expertise. T.D., M.A., and D.B. prepared the DNA constructs and produced virus. M.A. performed the injections, tissue dissociation, histology, imaging and image quantification. D.B. and T.D. performed the single-cell library preparation and prepared samples for sequencing. D.B. and M.A. built the data processing pipeline. D.B., M.A., T.D., and A.W. performed the analysis. All authors contributed to the MS as drafted by D.B., M.A., and V.G.. M.T. supervised single-cell RNA sequencing computational pipelines while V.G. supervised the overall project.

5.7 Methods

5.7.1 Animals

Animal husbandry and all experimental procedures involving animals were performed in accordance with the California Institute of Technology Institutional Animal Care and Use Committee (IACUC) guidelines and approved by the Office of Laboratory Animal Resources at the California Institute of Technology (animal protocol no. 1650). Male C57BL/6J mice (Stock No: 000664) used in this study

were purchased from the Jackson Laboratory (JAX). AAV variants were injected i.v. into the retro-orbital sinus of 6–7 week old mice.

5.7.2 Plasmids

In vivo vector characterization of AAV variant capsids was conducted using single-stranded (ss) rAAV genomes. pAAV:CAG-NLS-mNeonGreen, pAAV:CAG-NLS-mRuby2, pAAV:CAG-tdTomato, and pAAV:CAG-NLS-tdTomato constructs were adapted from previous publications (Chan et al., 2017; Ravindra Kumar et al., 2020). To introduce barcodes into the polyA region of CAG-NLS-mNeonGreen, we digested the plasmid with BglII and EcoRI, and performed Gibson assembly (E2611, NEB) to insert synthesized fragments with 7bp degenerate nucleotide sequences 89 bp upstream of the polyadenylation site. We then seeded bacterial colonies and selected and performed Sanger sequencing on the resulting plasmids to determine the corresponding barcode.

The UBC-mCherry-AAV-cap-in-cis plasmid was adapted from the rAAV-Cap-in-cis-lox plasmid from a previous publication (Deverman et al., 2016). We performed a restriction digest on the plasmid with BsmbI and SpeI to remove UBC-mCherry and retain the AAV9 cap gene and remaining backbone. We then circularized the digested plasmid using a gblock joint fragment to get a plasmid containing AAV2-Rep, AAV9-Cap, and the remaining backbone via T4 ligation. In order to insert UBC-mCherry with the desired orientation and location, we amplified its linear segment from the original rAAV-Cap-in-cis-lox plasmid. The linear UBC-mCherry-polyA segment and circularized AAV2-Rep,AAV9-cap plasmid were then both digested with HindIII and ligated using T4 ligation. In order to get the SV40 PolyA element in the proper orientation with respect to the inserted UBC-mCherry, we removed the original segment from the plasmid using AvrII and AccI enzymes and inserted AvrII, AccI treated SV40 gblock using T4 ligation to get the final plasmid.

To insert barcodes into UBC-mCherry-AAV-cap-in-cis, we obtained 300 bp DNA fragments containing the two desired capsid mutation regions for each variant and the variant barcode, flanked by BsrGI and XbaI cut sites. The three segments of the fragment were separated by BsaI Type I restriction sites. We digested the UBC-mCherry-AAV-cap-in-cis plasmid with BsrGI and XbaI, and

ligated each variant insert to this backbone. Then, to reinsert the missing regions, we performed Golden Gate assembly with two inserts and BsaI-HF.

5.7.3 Viral production

To produce viruses carrying *in trans* constructs, we followed established protocols for the production of rAAVs (Challis et al., 2019). In short, HEK293T cells were triple transfected using polyethylenimine (PEI) with three plasmids: pAAV (see Plasmids), pUCmini-iCAP-PHP.eB (Chan et al., 2017), pUCmini-iCAP-CAP-B10 (Flytzanis et al., 2020), or pUCmini-iCAP-PHP.V1 (Ravindra Kumar et al., 2020), and pHelper. After 120 h, virus was harvested and purified using an iodixanol gradient (Optiprep, Sigma). For our 7-variant pool, we modified the protocol to be a double transfection using PEI with two plasmids: UBC-mCherry-AAV-cap-in-cis and pHelper.

5.7.4 Tissue processing for single-cell suspension

Three to four weeks after the injection, mice (9-10 weeks old) were briefly anesthetized with isoflurane (5%) in an isolated plexiglass chamber followed by i.p. injection of euthasol (100 mg/kg). The following dissociation procedure of cortical tissue into a single-cell suspension was adapted with modifications from a previous report (Pool et al., 2020). Animals were transcardially perfused with ice-cold carbogenated (95% O_2 and 5% CO_2) NMDG-HEPES-ACSF (93 mM NMDG, 2.5 mM KCl, 1.2 mM NaH_2PO_4, 30 mM $NaHCO_3$, 20 mM HEPES, 25 mM glucose, 5 mM Na L-ascorbate, 2 mM thiourea, 3 mM Na-pyruvate, 10 mM $MgSO_4$, 1 mM $CaCl_2$, 1 mM kynurenic acid Na salt, pH adjusted to 7.35 with 10N HCl, osmolarity range 300–310 mOsm). Brains were rapidly extracted and cut in half along the anterior-posterior axis with a razor blade. Half of the brain was used for IHC histology while the second half of the brain was used for scRNA-seq. Tissue used for scRNA-seq was immersed in ice-cold NMDG-HEPES-ACSF saturated with carbogen. The brain was sectioned into 300-µm slices using a vibratome (VT-1200, Leica Biosystems, IL, USA). Coronal sections from Bregma −0.94 mm to −2.80 mm were collected in a dissection dish on ice containing NMDG-HEPES-ACSF. Cortical tissue from the dorsal surface of the brain to ~3.5 mm ventral was cut out and further sliced into small tissue pieces. NMDG-HEPES-ACSF was replaced by trehalose-

HEPES-ACSF (92 mM NaCl, 2.5 mM KCl, 1.2 mM NaH_2PO_4, 30 mM $NaHCO_3$, 20 mM HEPES, 25 mM glucose, 2 mM $MgSO_4$, 2 mM $CaCl_2$, 1 mM kynurenic acid Na salt, 0.025 mM D-(+)-trehalose dihydrate*$2H_2O$, pH adjusted to 7.35, osmolarity ranging 320–330 mOsm) containing papain (60 U/ml; P3125, Sigma Aldrich, pre-activated with 2.5 mM cysteine and a 0.5–1 h incubation at 34°C, supplemented with 0.5 mM EDTA) for the enzymatic digestion. Under gentle carbogenation, cortical tissue was incubated at 34°C for 50 min with soft agitation by pipetting every 10 min. 5 µl 2500 U/ml DNase I (04716728001 Roche, Sigma Aldrich) was added to the single-cell suspension 10 min before the end of the digestion. The solution was replaced with 200 µl trehalose-HEPES-ACSF containing 3 mg/ml ovomucoid inhibitor (OI-BSA, Worthington) and 1 µl DNase I. At room temperature, the digested cortical tissue was gently triturated with fire-polished glass Pasteur pipettes for three consecutive rounds with decreasing pipette diameters of 600, 300, and 150 µm. 800 µl of trehalose-HEPES-ACSF with 3 mg/ml ovomucoid inhibitor was added. The uniform single-cell suspension was pipetted through a 40 µm cell strainer (352340, Falcon) into a new microcentrifuge tube followed by centrifugation at 300 g for 5 min at 4°C. The supernatant was discarded and cell pellet was resuspended in 1 ml of trehalose-HEPES-ACSF. After mixing using a Pasteur pipette with a 150 µm tip diameter, the single-cell suspension was centrifuged again. Supernatant was replaced with fresh trehalose-HEPES-ACSF and the resuspended cell pellet was strained with a 20 µm nylon net filter (NY2004700, Millipore). After resuspension in trehalose-HEPES-ACSF, cells were pelleted again and resuspended in 100 µl of ice-cold resuspension-ACSF (117 mM NaCl, 2.5 mM KCl, 1.2 mM NaH_2PO_4, 30 mM $NaHCO_3$, 20 mM HEPES, 25 mM glucose, 1 mM $MgSO_4$, 2 mM $CaCl_2$, 1 mM kynurenic acid Na salt and 0.05% BSA, pH adjusted to 7.35 with Tris base, osmolarity range 320–330 mOsm). Cells were counted with a hemocytometer and the final cell densities were verified to be in the range of 400–2,500 cells/µl. The density of single-cell suspension was adjusted with resuspension-ACSF if necessary.

5.7.5 Transcriptomic library construction

Cell suspension volumes containing 16,000 cells–expected to retrieve an estimated 10,000 single-cell transcriptomes–were added to the 10x Genomics RT reaction mix and loaded to the 10x Single

Cell Chip A (230027, 10x Genomics) for 10x v2 chemistry or B (2000168, 10x Genomics) for 10x v3 chemistry per the manufacturer's protocol (Document CG00052, Revision F, Document CG000183, Revision C, respectively). We used the Chromium Single Cell 3' GEM and Library Kit v2 (120237, 10x genomics) or v3 (1000075, 10x Genomics) to recover and amplify cDNA, applying 11 rounds of amplification. We took 70 ng to prepare Illumina sequencing libraries downstream of reverse transcription following the manufacturer's protocol, applying 13 rounds of sequencing library amplification.

5.7.6 Viral library construction

We selectively amplified viral transcripts from 15 ng of cDNA using a cargo-specific primer binding to the target of interest and a primer binding the partial Illumina Read 1 sequence present on the 10x capture oligos (Table S 1). For animals injected with a single cargo, amplification was performed only once using the primer for the delivered cargo; for animals with distinct cargo sequences per variant, amplification was performed in parallel reactions from the same cDNA library using different cargo-specific primers for each reaction. We performed the amplification using 2x KAPA HiFi HotStart ReadyMix (KK2600) for 28 cycles at an annealing temperature of 53°C. Afterwards, we performed a left-sided SPRI cleanup with a concentration dependent on the target amplicon length, in accordance with the manufacturer's protocol (SPRISelect, Beckman Coulter B23318). We then performed an overhang PCR on 100 ng of product with 15 cycles using primers that bind the cargo and the partial Illumina Read 1 sequence and appending the P5/P7 sequences and Illumina sample indices. We performed another SPRI cleanup, and analyzed the results via an Agilent High Sensitivity DNA Chip (Agilent 5067-4626).

5.7.7 Sequencing

Transcriptome libraries were pooled together in equal molar ratios according to their DNA mass concentration and their mean transcript size as determined via bioanalyzer. Sequencing libraries were processed on Novaseq 6000 S4 300-cycle lanes. The run was configured to read 150 bp from each

end. Sequencing was outsourced to Fulgent Genetics and the UCSF Center for Advanced Technology.

All viral transcript libraries except barcoded UBC-mCherry were pooled together in equal molar ratios into a 4 nM sequencing library, then diluted and denatured into a 12 pM library as per the manufacturer's protocol (Illumina Document #15039740v10). The resulting library was sequenced using a MiSeq v3 150-cycle reagent kit (MS-102-3001), configured to read 91 base pairs for Read 2 and 28 base pairs for Read 1. To characterize the effect of sequencing depth, one viral transcript library was additionally processed independently on a separate MiSeq run.

The UBC-mCherry viral transcript library, which was recovered with primers near the polyadenylation site, consisted of fragments ~307 bp long. Since this length is within the common range for an Illumina NovaSeq run, this viral transcript library was pooled and included with the corresponding transcriptome library.

5.7.8 Transcriptome read alignment

For transcriptome read alignment and gene expression quantification, we used 10x Cell Ranger v5.0.1 with default options to process the FASTQ files from the transcriptome sequencing library. The reads were aligned against the mus musculus reference provided by Cell Ranger (mm10 v2020-A, based on Ensembl release 98).

To detect viral transcripts in the transcriptome, we ran an additional alignment using 10x Cell Ranger v5.0. 1 with a custom reference genome based on mm10 v2020-A. We followed the protocol for constructing a custom Cell Ranger reference as provided by 10x Genomics. This custom reference adds a single gene containing all the unique sequences from our delivered plasmids in the study, delineated as separate exons. Sequences that are common between different cargo are provided only once, and annotated as alternative splicings.

5.7.9 Viral transcript read alignment

For viral read alignment, we aligned each Read 2 to a template derived from the plasmid, excluding barcodes. The template sequence was determined by starting at the ATG start site of the XFP cargo and ending at the AATAAA polyadenylation stop site. We used a Python implementation of the Striped Smith-Waterman algorithm from scikit-bio to calculate an alignment score for each read, and normalized the score by dividing by the maximum possible alignment score for a sequence of that length, minus the length of the barcode region. For each Read 2 that had a normalized alignment score of greater than 0.7, we extracted the corresponding cell barcode and UMI from Read 1, and any insertions into the template from Read 2.

5.7.10 Constructing the variant lookup table

For co-injections with multiple templates and injections of barcoded templates, we constructed a lookup table to identify which variant belongs to each cell barcode/UMI. For each template, we counted the number of reads for each cell barcode/UMI. For reads of barcoded cargo, we only counted reads where the detected insertion in the barcode region unambiguously aligned to one of the pre-defined variant barcodes. Due to sequencing and PCR amplification errors, most cell barcode/UMI combinations had reads associated with multiple variants. Thus, we identified the variant with the largest count for each cell barcode/UMI. We discarded any cell barcode/UMIs that had more than one variant tied for the largest count. Finally, each cell barcode/UMI that was classified as a viral transcript in the transcriptome (see Transcriptome read alignment) was converted into the virus detected in the variant lookup table, or was discarded if it did not exist in the variant lookup table.

5.7.11 Estimating transduction rate

To determine an estimate of the percent of cells within a group expressing viral cargo above background, we compared the viral transcript counts in that group of cells to a background distribution of viral transcript counts in debris (see Droplet type classification). First, we obtained

the empirical distribution of viral transcript counts by extracting the viral counts for that variant in cell barcodes classified as the target cell type as well as cell barcodes classified as debris. Next, we assumed a percentage of cells containing debris. For each viral transcript count, starting at 0, we calculated the number of cells that would contain this transcript count, if the assumed debris percentage was correct. We then calculated an error between this estimate and the number of cells with this transcript count in the cell type of interest. We tallied this error over all the integer bins in the histogram, allowing the error in a previous bin to roll over to the next bin. We repeated this for all possible values of percentage of debris from 0 to 100 in increments of 0.25, and the value that minimized the error was the estimated percentage of cells whose viral transcript count could be accounted for by debris. The inverse of this was our estimate of the number of cells expressing viral transcripts above background.

To validate that this method reliably recovers an estimate of transduction rate, we performed a series of simulations using models of debris viral transcript counts added to proposed cell type transcript count distributions across a range of parameterizations. To get estimates of the background distribution of debris, we used diffxpy (https://github.com/theislab/diffxpy) to fit the parameters of a negative binomial distribution to the viral transcript counts in debris droplets within a sample. We then postulated 1,000 different parameterizations of the negative binomial representing transcript counts in groups of cells, with 40 values of r ranging from 0.1 to 10, spaced evenly apart, and 25 values of p ranging from 0.001 to 0.99, spaced evenly apart. For each proposed negative binomial model, we drew 1,000 random samples of viral counts from the learned background distribution, and 1,000 random samples from the proposed cell distribution, and summed the two vectors. This summed vector was then used in our transduction rate estimation function, along with a separate 1,000 random samples of background viral transcripts for the function to use as an estimate of the background signal. We calculated the true probability of non-zero expression in our proposed cell negative binomial model ($1 - P(X = 0)$), and compared this value with the estimated value from the transduction rate estimation method.

5.7.12 Calculating viral tropism

For each variant v_n and cell type of interest c_i, we estimated the percentage of cells expressing viral cargo. To calculate tropism bias, we used this estimated expression rate, t_{c_i,v_n}, to estimate the number of cells expressing viral transcripts in that cell type, T_{c_i,v_n} out of the total number of cells of that type, N_{c_i}. $T_{c_i,v_n} = t_{c_i,v_n} N_{c_i}$. Cell type bias, $b_{c_i,v_n,}$ within a sample was then calculated as the ratio of the number of cells of interest divided by the total number of transduced cells, $b_{c_i,v_n} = \frac{T_{c_i,v_n}}{\sum_j T_{c_j v_n}}$. Finally, to calculate the difference in transduction bias for a particular variant relative to other variants in the sample, δ_{c_i,v_n}, we subtracted the bias of the variant from the mean bias across all other variants, $\delta_{c_i,v_n} = \frac{T_{c_i,v_n}}{\sum_j T_{c_j v_n}} - \frac{\sum_{m \neq n} T_{c_i,v_m}}{\sum_{m \neq n} \sum_j T_{c_j,v_m}}$.

5.7.13 Histology

5.7.13.1 Immunohistochemistry

The immunohistochemistry procedure was adapted from a previous publication (Oikonomou et al., 2019). Brain tissue was fixed in 4% paraformaldehyde (PFA) at 4°C overnight on a shaker. Samples were immersed in 30% sucrose in 1x phosphate buffered saline (PBS) solution for >2 days and then embedded in Tissue-Tek O.C.T. Compound (102094-104, VWR) before freezing in dry ice for 1 h. Samples were sectioned into 50 μm coronal slices on a cryostat (Leica Biosystems). Brain slices were washed once with 1x phosphate buffered saline (PBS) to remove O.C.T. Compound. Samples were then incubated overnight at 4°C on a shaker in a 1x PBS solution containing 0.1% Triton X-100, 10% normal goat serum (NGS; Jackson ImmunoResearch, PA, USA), and primary antibodies. Sections were washed three times for 15 min each in 1x PBS. Next, brain slices were incubated at 4°C overnight on a shaker in a 1x PBS solution containing 0.1% Triton X-100, 10% NGS, and secondary antibodies. Sections were washed again three times for 15 min each in 1x PBS. Finally, slices were mounted on glass microscope slides (Adhesion Superfrost Plus Glass Slides, #5075-Plus, Brain Research Laboratories, MA, USA). After the brain slices dried, DAPI-containing mounting

media (Fluoromount G with DAPI, 00-4959-52, eBioscience, CA, USA) was added before protecting the slices with a cover glass (Cover glass, #4860-1, Brain Research Laboratories, MA, USA). Confocal images were acquired on a Zeiss LSM 880 confocal microscope (Zeiss, Oberkochen, Germany). The following primary antibodies were used: rabbit monoclonal to NeuN (Rbfox3) (1:500; ab177487; Abcam, MA, USA), rabbit monoclonal to S100 beta (1:500; ab52642; Abcam, MA, USA), and rabbit monoclonal to Olig2 (1:500; ab109186; Abcam, MA, USA). The following secondary antibody was used: goat anti-rabbit IgG H&L Alexa Fluor 647 (1:500; ab150079; Abcam, MA, USA).

5.7.13.2 Fluorescent in situ hybridization chain reaction

FISH-HCR was conducted as previously reported (Patriarchi et al., 2018). Probes targeting neuronal markers were designed using custom-written software (https://github.com/GradinaruLab/HCRprobe). Probes contained a target sequence of 20 nucleotides, a spacer of 2 nucleotides, and an initiator sequence of 18 nucleotides. Criteria for the target sequences were: (1) a GC content between 45%–60%, (2) no nucleotide repeats more than three times, (3) no more than 20 hits when blasted, and (4) the ΔG had to be above –9 kcal/mol to avoid self-dimers. Last, the full probe sequence was blasted and the Smith-Waterman alignment score was calculated between all possible pairs to prevent the formation of cross-dimers. In total, we designed 26 probes for *Gad1*, 20 probes for *Vip*, 22 probes for *Pvalb*, 18 probes for *Sst*, and 28 probes for *Slc17a7*. Probes were synthesized by Integrated DNA Technologies.

5.7.14 Droplet type identification

scRNA-seq datasets were analyzed with custom-written scripts in Python 3.7.4 using a custom fork off of scVI v0.8.1, and scanpy v1.6.0. To generate a training dataset for classifying a droplet as debris, multiplets, neuronal, or non-neuronal cells, we randomly sampled cells from all 27 cortical tissue samples. We sampled a total of 200,000 cells, taking cells from each tissue sample proportional to the expected number of cells loaded into the single-cell sequencing reaction. Within each sample, cells were drawn randomly, without replacement, weighted proportionally by their total

number of detected UMIs. For each sample, we determined a lower bound on the cutoff between cells and empty droplets by constructing a histogram of UMI counts per cell from the raw, unfiltered gene count matrix. We then found the most prominent trough preceding the first prominent peak, as implemented by the scipy peak_prominences function. We only sampled from cells above this lower bound. Using these sampled cells, we trained a generative neural network model via scVI with the following parameters: 20 latent features, 2 layers, and 256 hidden units. These parameters were chosen from a coarse hyperparameter optimization centered around the scVI default values (Table S 3). We included the sample identifier as the batch key so that the model learned a latent representation with batch correction.

After training, Leiden clustering was performed on the learned latent space as implemented by scanpy. We used default parameters except for the resolution, which we increased to 2 to ensure isolation of small clusters of cell multiplets. Using the learned generative model, we draw 5000 cells from the posterior distribution based on random seed cells in each cluster. We draw an equal number conditioned on each batch. From these samples, we then calculated a batch-corrected probability of each cluster expressing a given marker gene (see Cluster marker gene determination). For this coarse cell typing, we chose a single marker gene for major cell types expected in the cortex (Table S 2). If a cluster was expressing the neuron marker gene Rbfox3, it was labeled as “Neurons”. If a cluster was expressing any of the other non-neuronal marker genes, it was labelled as “Non-neurons”. Next, we ran Scrublet on the training cells to identify potential multiplets. Scrublet was run on each sample independently, since it is not designed to operate on combined datasets with potential batch-specific confounds. We then calculated the percentage of droplets in each cluster of the combined data that were identified as multiplets by Scrublet. We found a percentage threshold for identifying a cluster as containing predominantly multiplets by using Otsu’s threshold, as implemented by scikit-image. All droplets in any cluster above the multiplet percentage threshold were labelled as “Multiplets”. All other clusters were labelled as “Debris”.

Next, we trained a cell-type classifier using scANVI on the droplets labeled as training data. We used the weights from the previously trained scVI model as the starting weights for scANVI. Rather

than using all cells for every epoch of the trainer, we implemented an alternative sampling scheme that presented each cell type to the classifier in equal proportions. Once the model was trained, all cells above the UMI lower noise bound were run through the classifier to obtain their cell-type classification. Droplets classified as "Neurons" or "Non-neurons" were additionally filtered by their scANVI-assigned probability. We retained only cells above an FDR threshold of 0.05, corrected for multiple comparisons using the Benjamini-Hochberg procedure. Finally, since the original run of Scrublet for multiplet detection was performed on only the training data, and thus did not take advantage of all the cells available, we ran Scrublet on all droplets classified as cells, and removed any identified multiplets.

5.7.15 Cluster marker gene determination

To identify which clusters are expressing marker genes, we determined an estimated probability of a marker gene being expressed by a random cell in that cluster. For each cluster, we randomly sampled 5,000 cells, with replacement. We used scVI to project each cell into its learned latent space, and then used scVI's posterior predictive sampling function to generate an example cell from this latent representation, and tallied how many times the gene is expressed. We repeated this for each batch, conditioning the posterior sample on that batch, to account for technical artifacts such as sequencing depth. Once we obtained a probability of expression of a marker gene for each cluster, we find a threshold for expression using Otsu's method, as implemented by scikit-image. Clusters that have a probability of expression above the threshold are considered positive for that marker gene.

5.7.16 Neuronal subtype classification

Cells classified as neurons were further subtyped using annotations from a well-curated reference dataset. We used the Mouse Whole Cortex and Hippocampus 10x dataset from the Allen Institute for Brain Science as our reference dataset (Yao et al., 2021). First, we filtered the reference dataset to contain only cell types that are found within the brain regions collected for our experiments. To ensure that, overall, enough cells per cell type were present in our datasets, we merged cell types

with common characteristics, such as expression of key marker genes. We re-aligned our cell transcriptome reads to the same pre-mRNA reference used to construct the reference dataset, so that the gene count matrices had a 1:1 mapping. We then trained a joint scANVI model with all cells identified as neurons from our samples and the reference database to learn a common latent space between them. The model was trained to classify cells based on the labels provided in the reference dataset. Cells were sampled from each class in equal proportions during training. After the model was trained, all neurons from our sample were run through the model to obtain their cell type classification.

5.7.17 Non-neuronal subtype classification

Cells classified as non-neuronal were further subtyped using automatic clustering and marker gene identification. We trained an scVI model using only the non-neuronal cells and performed Leiden clustering as implemented by scanpy on the latent space. We determined which clusters were expressing each of 31 marker genes across 13 cell subtypes. Marker genes were identified from a review of existing scRNA-seq, bulk RNA-seq, or IHC studies of mouse brain non-neuronal subtypes (Table S 2). Each cluster was assigned to a cell subtype if it was determined positive for all the marker genes for that cell subtype (see Cluster marker gene determination). If a cluster contained all the marker genes for multiple cell subtypes, the cluster was assigned to the cell subtype with the greatest number of marker genes. Clusters that did not express all the marker genes for any cell subtype were labeled as “Unknown”. Clusters that expressed all the marker genes for multiple cell subtypes with the same total number of marker genes were labeled as “Multiplets”. For cell types that contained multiple clusters, we then calculated the probability of every gene being zero in each cluster (see Cluster marker gene determination). We then compared gene presence between clusters of the same cell type to see if there were any subclusters that had a dominant marker gene (present in > 50% of samples), that was not present in any of the other clusters (< 10% of samples). For the three cell types that had unique marker genes, we named the cluster after the gene with the highest 2-proportion z-score between the sampled gene counts in that cluster vs the rest.

5.7.18 Quantification of images

Quantitative data analysis of confocal images was performed blind with regard to AAV capsid variant. Manual quantification was performed using the Cell Counter plugin, present in the Fiji distribution of ImageJ (National Institutes of Health, Bethesda, MD) (Schindelin et al., 2012). Transduction rate was calculated as the total number of double positive cells (i.e. viral transgene and cell type marker) divided by the total number of cell type marker labeled cells. For each brain slice, at least 100 cells positive for the gene markers of interest were counted in the cortex.

5.7.19 Differential expression

To calculate differential expression within cell types between groups of animals, we used the DESeq2 R package (Love et al., 2014). For each cell type, the gene counts are summed across all cells of that type and treated as a pseudo-bulk sample. The summed gene counts from each animal are then included as individual columns for a DESeq2 differential expression analysis. We performed 3 DPI DE and 25 DPI separately, testing each sample against saline-injected controls. For each cell type, only genes that were present in all samples of at least one condition are included.

5.7.20 Marker gene dot plots

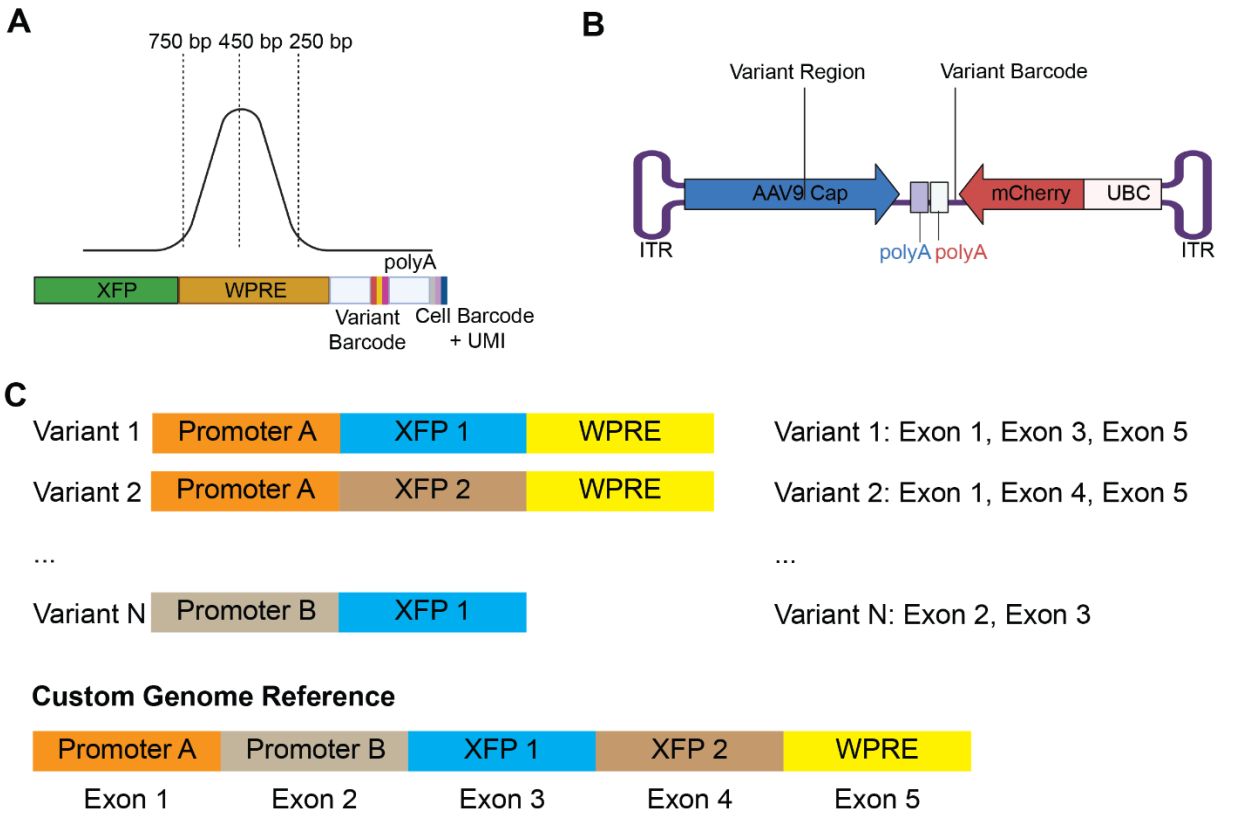
To generate dot plots for marker genes, we used scanpy's dotplot function (Wolf et al., 2018). Gene counts were normalized to the sum of the total transcript counts per cell using scanpy's normalize_total function. Normalized gene expression values are log-transformed as part of the plotting function.

5.7.21 Statistics

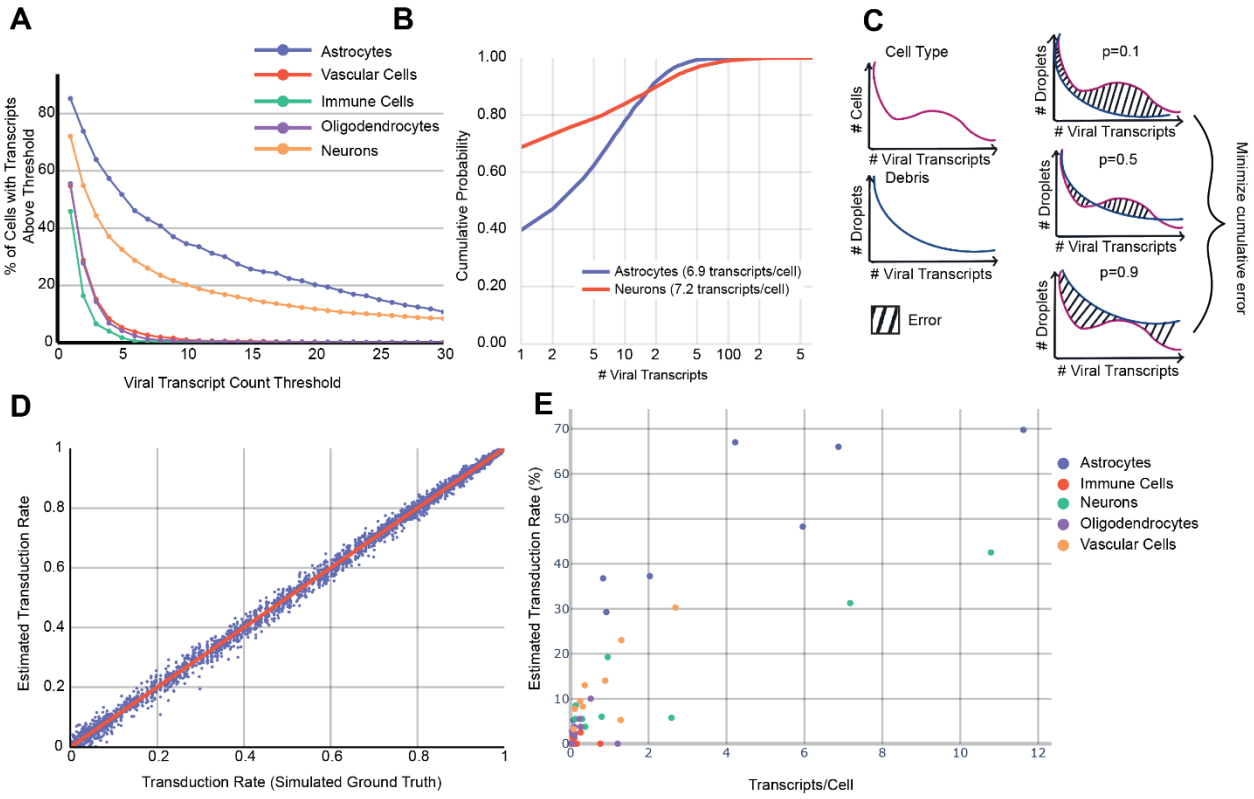
Statistical analyses comparing the fraction of transduced cells in different cell types for Figures 2, 3, and 4 C were conducted using GraphPad Prism 9. Statistical analyses comparing proportions of transduced cells within an animal in Figure 25**E** Figure Figure 27 were performed using the Python statsmodels library v0.12.1. No statistical methods were used to predetermine sample sizes. The

statistical test applied, sample sizes, and statistical significant effects are reported in each figure legend. The significance threshold was defined as a = 0.05.

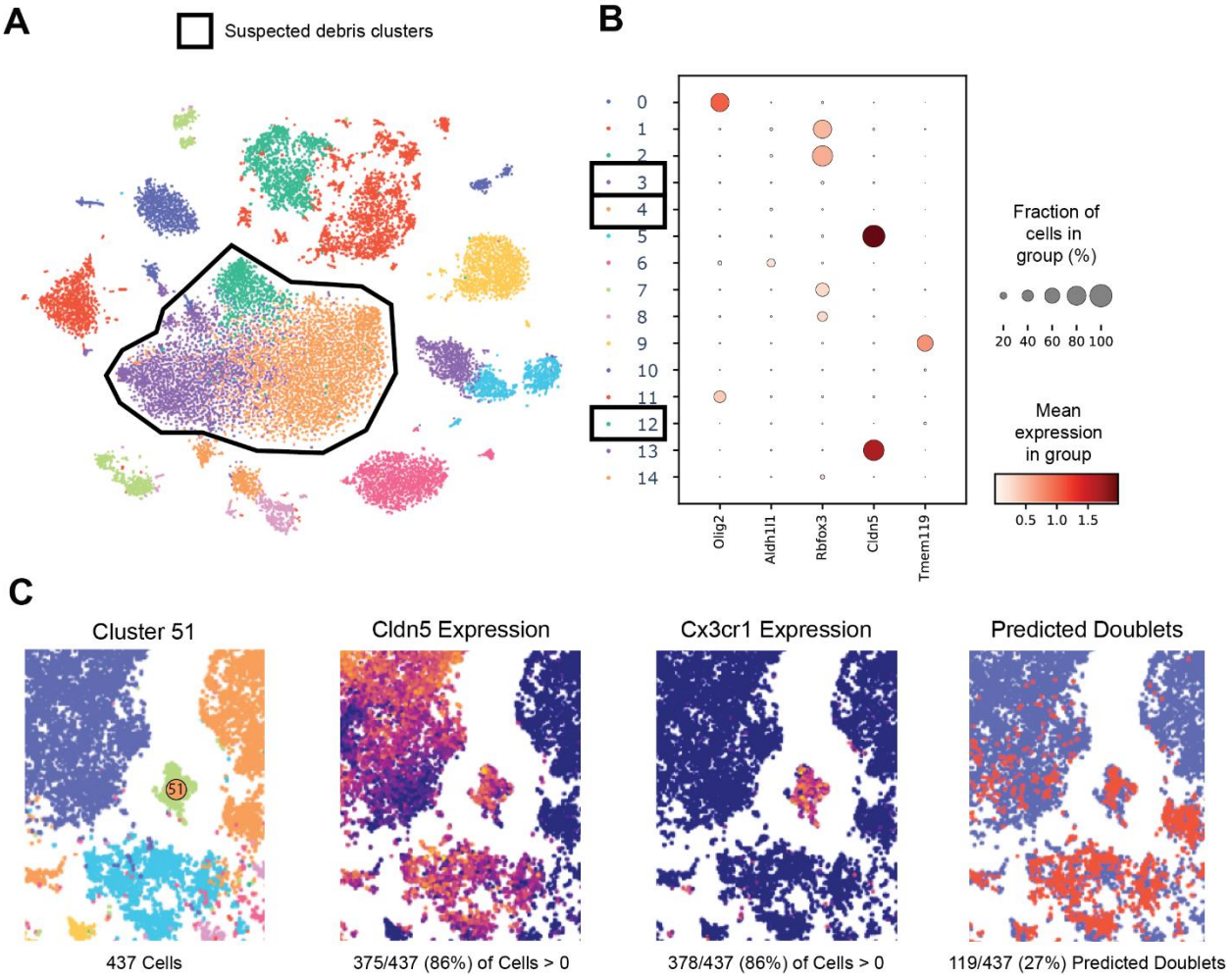
6.8 Supplemental Figures



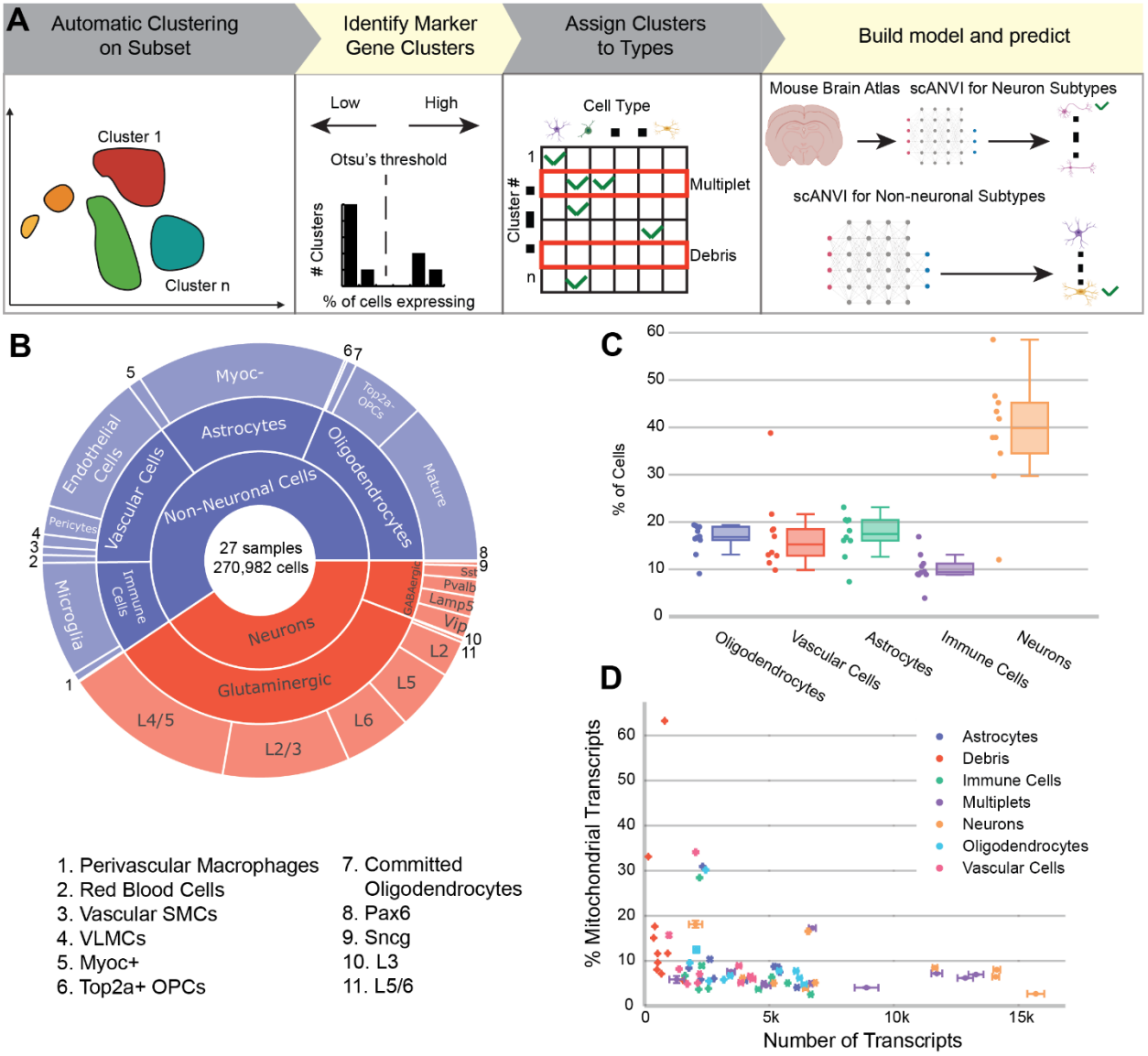
Supplementary Figure 2. Plasmid details. (A) Size of typical transcriptome cDNA library post-fragmentation. Both distinguishing XFPs and variant barcodes fall outside the typical capture region of single-cell RNA sequencing workflows. (B) UBC-mCherry-AAV-cap-in-cis plasmid used for 7-variant barcoded pool. (C) Visualization of the construction procedure for the custom genome reference. Variant cargos are segmented into common and uncommon regions, and each unique segment is concatenated together as a contiguous gene. Variants are defined as different splicings of the custom AAV gene.



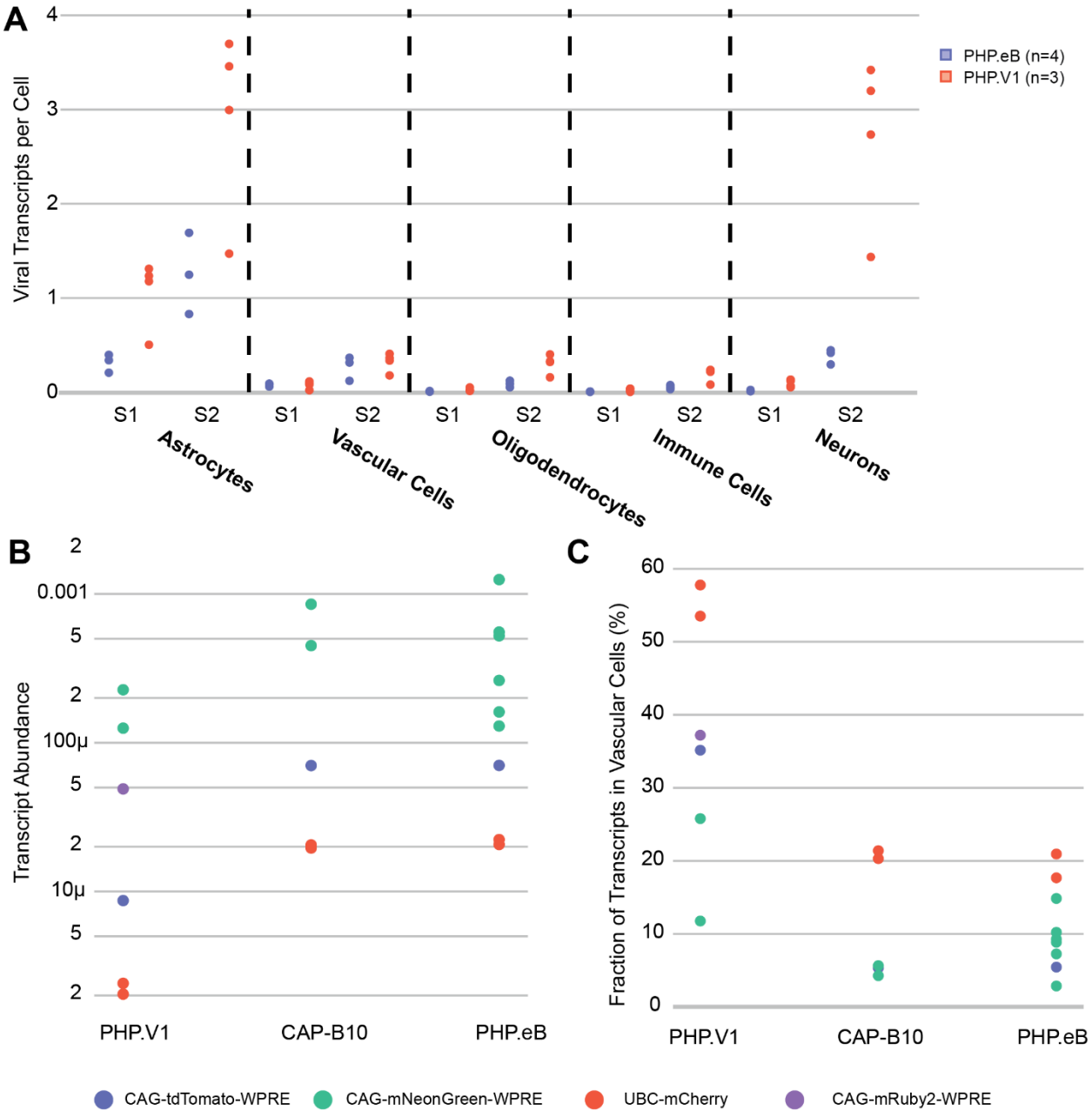
Supplementary Figure 3. Expression rate estimation. (A) Percent of cells expressing AAV-PHP.eB cargo transcripts above a fixed threshold in a single sample. **(B)** An example of the distribution of viral transcript counts in a single animal from AAV-PHP.eB carrying CAG-mNeonGreen-WPRE in neurons and astrocytes. **(C)** Visualization of our expression-rate estimation algorithm. The distribution of the cell type of interest and background debris is obtained. An error is calculated for different estimates of the percent of the cells that express background levels of transcripts. This error is minimized to find the best fit. **(D)** Performance of the expression rate estimation algorithm on simulated data consisting of negative binomial distributions with parameters r between 0.1 and 10 and p between 0.001 and 0. 99, spaced evenly apart. **(E)** Comparison between mean transcripts/cell (x) and the estimated transduction rate (y) in major cell types for AAV-PHP.eB across 9 samples.



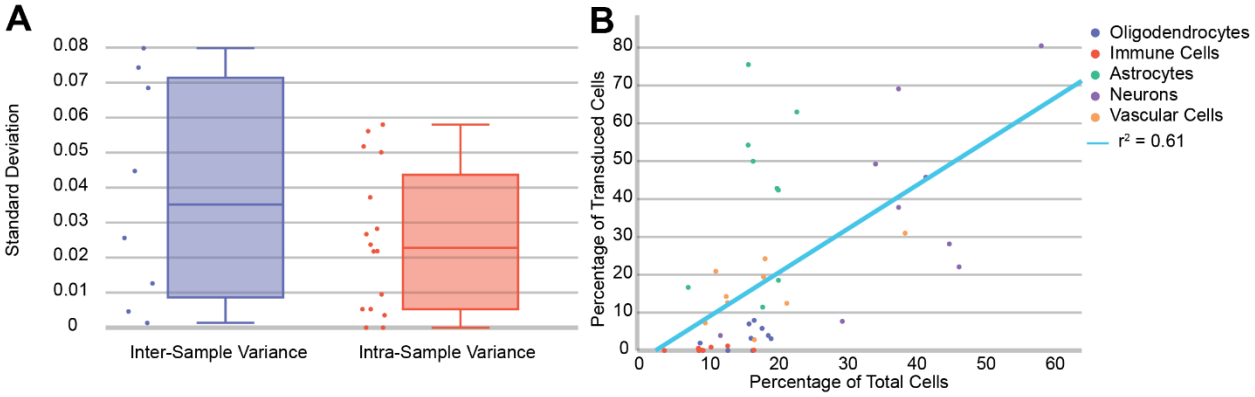
Supplementary Figure 4. Noise from debris and doublets. (A) An example of a Cell Ranger filtered dataset. This is a t-SNE projection of the log-normalized gene expression space. Suspected debris clusters are outlined. **(B)** Marker gene expression for the major cell types in the brain—Oligodendrocytes/Olig2, Astrocytes/Aldh1l1, Neurons/Rbfox3, Vascular Cells/Cldn5, Immune Cells/Tmem119—for each cluster. Darker colors indicate higher mean expression, and dot size correlates with the abundance of the gene in that cluster. **(C)** An example of a multiplet cluster from the joint scVI space of all training samples, projected via t-SNE. Cluster 51 is annotated, and raw gene expression of Cldn5 and Cx3cr1 are shown. The percentage of cells in cluster 51 expressing each marker gene is displayed. (right) Predicted doublets from Scrublet are overlaid in red.



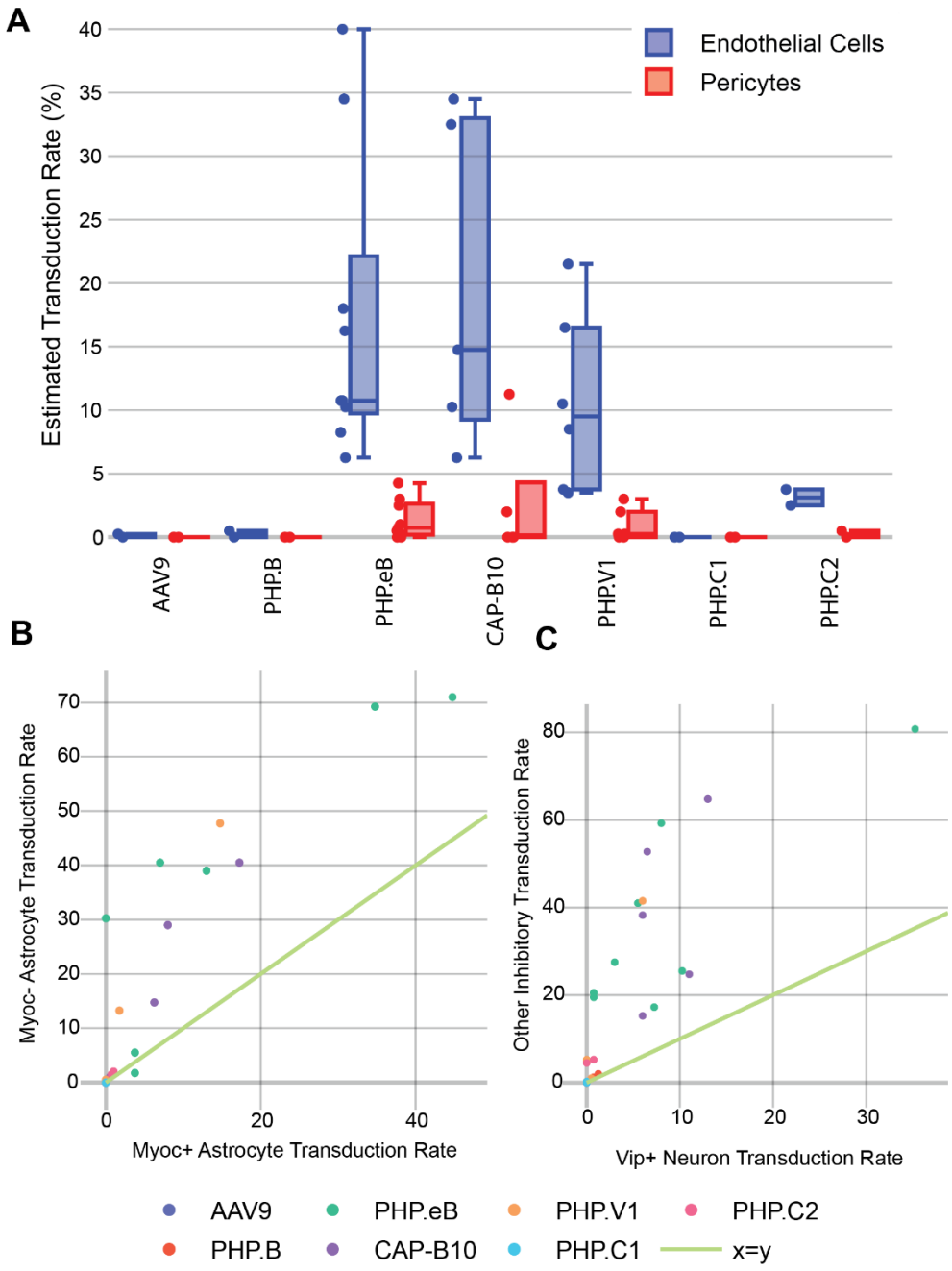
Supplementary Figure 5. Cell typing. (A) Cell typing workflow. A subset of cells are used for training. For each marker gene, clusters expressing that marker gene are identified. Clusters that have no marker genes (debris) or are determined to be multiplets via Scrublet are marked for removal. Training data used to train a scANVI model to predict the remaining cells. A reference database can be used instead of manually labeled cells, as we did for neuronal subtypes. **(B)** Cell-type distribution of all identified cells from our combined cell-type taxonomy. This includes samples described in the study as well as additional controls and animals used for troubleshooting and prototyping. **(C)** Cell-type percentages across the major cell types in the ten samples used for AAV tropism characterization. One of the samples, BC1, had dramatically fewer neurons than any other sample and correspondingly higher percentages of non-neurons. **(D)** Mitochondrial gene ratio and total transcript counts of the major cell type clusters in the ten samples used for tropism characterization.



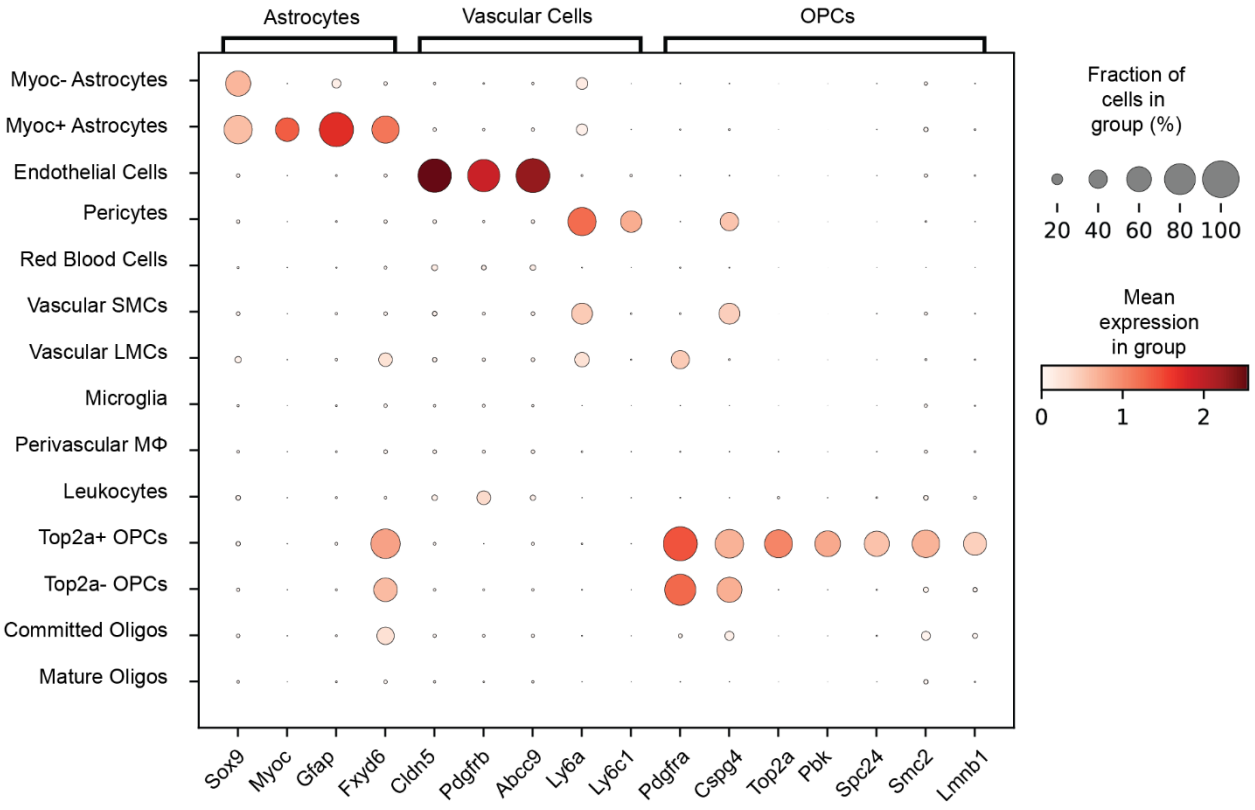
Supplementary Figure 6. Transcript expression. (A) Viral transcript expression of different barcodes across two samples (S1, S2). Each point is a distinct barcode. **(B)** Viral transcript abundance in entire samples (viral transcripts / total transcripts) across different variants carrying different cargo. **(C)** Fraction of transcripts detected in vascular cells vs all other cell types.



Supplementary Figure 7. Inter-sample variability. (A) The standard deviations between measurements of the fraction of transduced cells in all major non-neuronal cell types in AAV-PHP.V1 and AAV-PHP.eB. Inter-sample variance (left) refers to the standard deviation between animals, and intra-sample variance (right) refers to the standard deviation between barcodes within the same animal. **(B)** The distribution of recovered cell types compared to the distribution of transduced cells across nine samples injected with AAV-PHP.eB.



Supplementary Figure 8. Cell subtype inspection. (A) Estimated transduction rate of endothelial cells vs pericytes across all samples and variants. **(B)** Pairwise transduction rate of Myoc+ and Myoc- astrocytes across all variants and samples. Each point is a single variant in a different sample. **(C)** Pairwise transduction rate of Vip+ neurons vs all other inhibitory neurons across all variants and samples.



Supplementary Figure 9. Cell subtype markers. Gene expression of additional marker genes for astrocyte and OPC subtypes.

Table S 1. Primers. Primers used for round 1 and round 2 amplification of viral transcripts. Primers with TC1 and TC2 in the amplicon name indicate they were used only for those samples.

Amplicon	Read	Round	Sequence (Ns indicate Illumina sample index)
All Viruses	1	1	CTACACGACGCTCTTCCGATCT
All Viruses	1	2	AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGCTCTTCCGAT

mNeonGreen TC1	2	1	TTCAAGGAGTGGCAAAAGGCCTTTACCGATGTGAT
mRuby2	2	1	CAACGGGAACATGCAGTTGCCAAGTTTGCTGG
mNeonGreen	2	1	TAACTATCTGAAGAACCAGCCGATGTAC
tdTomato TC2	2	1	AGGACTACACAATTGTCGAACAGTATGAG
tdTomato	2	1	ACAACGAGGACTACACCATCGTGG
mCherry	2	1	CATCGTGGAACAGTACGAACG
WPRE	2	2	CAAGCAGAAGACGGCATACGAGATNNNNNNNNGT GACTGGAGTTCAGACGTGTGCTCTTCCGATCTAGA CGAGTCGGATCTCCCT
mNeonGreen	2	2	CAAGCAGAAGACGGCATACGAGATNNNNNNNNGT GACTGGAGTTCAGACGTGTGCTCTTCCGATCTTTC AAGGAGTGGCAAAAGGC
mRuby2	2	2	CAAGCAGAAGACGGCATACGAGATNNNNNNNNGT GACTGGAGTTCAGACGTGTGCTCTTCCGATCTCAA CGGGAACATGCAGTTGC
tdTomato TC2	2	2	CAAGCAGAAGACGGCATACGAGATNNNNNNNNGT GACTGGAGTTCAGACGTGTGCTCTTCCGATCTGCA TGGACGAGCTGTACAAG

tdTomato	2	2	CAAGCAGAAGACGGCATACGAGATNNNNNNNNGT GACTGGAGTTCAGACGTGTGCTCTTCCGATCTCCT CTTTCTCTATGGGATGGATGA
mCherry	2	2	CAAGCAGAAGACGGCATACGAGATNNNNNNNNGT GACTGGAGTTCAGACGTGTGCTCTTCCGATCTCGG CATGGACGAGCTGT

Table S 2. Marker Genes.

Cell Type	Marker Gene(s)
Astrocytes	Aldh1l1 *(Cahoy et al., 2008)*, Sox9 *(Sun et al., 2017)*
Neurons	Rbfox3 *(Lin et al., 2016)*
Vascular Cells	Cldn5 *(Song et al., 2020)*
Endothelial Cells	Slc2a1 *(Veys et al., 2020)*
Pericytes	Pdgfrb *(Winkler et al., 2010)*, Rgs5, Abcc9 *(He et al., 2016)*
Red Blood Cells	Hba-a1, Hba-a2 *(Capellera-Garcia et al., 2016)*
Vascular SMCs	Acta2, Myh11, Tagln *(Chasseigneaux et al., 2018)*
Vascular LMCs	Fam180a, Slc6a13, Dcn, Ptgds *(Marques et al., 2016)*
Microglia	Cx3cr1, Tmem119 *(Jordão et al., 2019)*

Leukocytes	Itgal, Gzma *(Huang and Sabatini, 2020)*
Perivascular Macropages	Mrc1 *(Jordão et al., 2019)*
Oligodendrocytes	Olig2 *(Dai et al., 2015)*
OPCs	Pdgfra, Cspg4 *(Suzuki et al., 2017)*
Mature Oligos	Mog, Mbp *(Miron et al., 2011)*
Committed Oligos	Ptprz1, Bmp4, Nkx2-2, Vcan *(Marques et al., 2016)*

Table S 3. scVI Hyperparameter Tuning

Dispersion	Latent Lib Size	# Latent	# Layers	# Hidden	Test KL Divergence
Gene	False	10	1	128	5366.4
Gene-batch	True	10	1	128	5406.1
Gene	True	10	1	128	5391.0
Gene-batch	True	50	2	512	5362.6
Gene-batch	False	10	1	128	5378.7
Gene	False	25	1	128	5354.6
Gene	False	25	2	256	5337.8

Gene	False	20	2	256	5336.7
Gene	False	40	4	1024	5338.0

Table S 4. Sample Metadata. Supplemental file contains the following fields.

Field Name	*Description*
10X Version	Whether the sample was processed using 10X V2 or V3 chemistry
Animal ID	A unique animal identifier. Some animals provided multiple samples
Target # Cells	The target number of cells for extraction. 1.6X this number is loaded into the 10X Chromium instrument
# Recovered Cells	The number of cells recovered, after debris and multiplet filtering
Cell Ranger # Cells	The number of cells as predicted by Cell Ranger
Predicted Multiplets	The number of predicted multiplets
Transcriptome Sequencing Depth	The number of reads
Transcriptome Reads/Cell	The number of reads divided by the number of recovered cells
Median UMIs/Cell	Of the recovered cells, the median total UMI count

Median Genes/Cell	Of the recovered cells, the median number of genes detected with at least one transcript
Variants Recovered	Which variants were recovered from this sample. Samples labeled "Cell Typing Only" were not used for tropism analysis, but were included in the cell type classifier
Virus Sequencing Depth	The number of reads of the amplified viral transcripts across all templates
Virus Reads/Cell	The read depth of the amplified viral transcripts
Age at Extraction (Days)	The age of the animal at extraction time
Virus Incubation Time (Days)	How many days prior to extraction the animal was injected
Percent of Virus UMIs Determined	What percent of transcriptome reads that aligned to the virus gene were disambiguated from the amplified lookup table

Table S 5. Variant Barcodes

Variant	**Cargo**	**Barcodes**
AAV-PHP.eB	pAAV:CAG-NLS-mNeonGreen	
AAV-PHP.V1	pAAV:CAG-NLS-mRuby2	

AAV-PHP.eB	pAAV:CAG-NLS-mNeonGreen	CCTGACA, GGACAGA, GCACAGA, CGAGAGA
AAV-PHP.V1	pAAV:CAG-tdTomato	
AAV-PHP.V1	pAAV:CAG-NLS-mNeonGreen	CAGTGTC, GAGAGTG, GTGTGAG
AAV-CAP-B10	pAAV:CAG-NLS-mNeonGreen	
AAV-PHP.eB	pAAV:CAG-NLS-tdTomato	
AAV-CAP-B10	pAAV:CAG-NLS-tdTomato	
AAV9	UBC-mCherry-AAV-cap-in-cis	CGTCTCAGCTATAACTTCCAA CGAGGTCGTAAGGTCGGCATT TGATTATCATGCCTGCTCAGG
AAV-PHP.B	UBC-mCherry-AAV-cap-in-cis	TATACCCAACCACTCAGTCCC CGGTTTTAGCACGGCCATAGA AAGCGATGTCTCTACACGATA
AAV-PHP.eB	UBC-mCherry-AAV-cap-in-cis	TACAGCTTTTTGACTGGAGGT CTGGCATTAATACGCGGGTCA TACAGGTCCTAGACAGGTGAT
AAV-CAP-B10	UBC-mCherry-AAV-cap-in-cis	GCTGGGCGTTAAAGTACTCGC

		GCAACTGGGATAATCGTAGTC AACGGAGTGAACGGACCCTAG
AAV-PHP.V1	UBC-mCherry-AAV-cap-in-cis	GTGGCGGGTTTCCGAAAAAGT TCGTCGGCACTCTCTTAGAGC CATGTGATAGTGAAGCACGCC
AAV-PHP.C1	UBC-mCherry-AAV-cap-in-cis	TCTGTGCTGCTCTTCTAACAA TCTGACGGCGGGTAAACACTG TGGCCACCCGCAGAGTATACT
AAV-PHP.C2	UBC-mCherry-AAV-cap-in-cis	GACTAGGGTAAGTGAGCTATG CGAATTTCTTCCATACCTCCT TAGTGCCAACAACGGAGAAGA

Table S 6. Differentially Expressed Genes. Supplemental file contains one tab for astrocytes, pericytes, and OPCs, with the following fields.

Field Name	Description
Gene ID	The Ensembl Gene ID
Gene name	The canonical gene name

P Non Zero	The probability of a cell expressing this gene in this cluster
P Non Zero Rest	The probability of a cell expressing this gene in the other cell subtype clusters

Table S 7. Differentially Expressed Genes Across Time Points. Supplemental file contains one tab per cell type, with the following fields.

Field Name	Description
Gene ID	The Ensembl Gene ID
Gene name	The canonical gene name
Mean expression	The mean expression of this gene in this group
L2FC	The log fold change of this gene
L2FC SE	The standard error of the L2FC
Stat	The stat, as reported by DESeq2
P-value	The unadjusted P-value
Adjusted P-value	The adjusted P-value

Chapter 6

SINGLE CELL PROFILING OF CAPILLARY BLOOD ENABLES OUT OF CLINIC HUMAN IMMUNITY STUDIES

Adapted from:

Dobreva, T.*, Brown, D.*, Park, J. H., & Thomson, M. (2020). Single cell profiling of capillary blood enables out of clinic human immunity studies. Scientific Reports, 10(1). https://doi.org/10.1038/s41598-020-77073-3

6.1 Summary

An individual's immune system is driven by both genetic and environmental factors that vary over time. To better understand the temporal and inter-individual variability of gene expression within distinct immune cell types, we developed a platform that leverages multiplexed single-cell sequencing and out-of-clinic capillary blood extraction to enable simplified, cost-effective profiling of the human immune system across people and time at single-cell resolution. Using the platform, we detect widespread differences in cell type-specific gene expression between subjects that are stable over multiple days.

6.2 Introduction

Increasing evidence implicates the immune system in an overwhelming number of diseases, and distinct cell types play specific roles in their pathogenesis (Farh et al., 2015; Gate et al., 2020). Studies of peripheral blood have uncovered a wealth of associations between gene expression, environmental factors, disease risk, and therapeutic efficacy (De Jager et al., 2015; Fairfax and Knight, 2014; Sumitomo et al., 2018). For example, in rheumatoid arthritis, multiple mechanistic paths have been found that lead to disease, and gene expression of specific immune cell types can

be used as a predictor of therapeutic non-response (Sumitomo et al., 2018). Furthermore, vaccines, drugs, and chemotherapy have been shown to yield different efficacy based on time of administration, and such findings have been linked to the time-dependence of gene expression in downstream pathways (Kobayashi et al., 2002; Lévi et al., 2007; Long et al., 2016). However, human immune studies of gene expression between individuals and across time remain limited to a few cell types or time points per subject, constraining our understanding of how networks of heterogeneous cells making up each individual's immune system respond to adverse events and change over time. The advent of single-cell RNA sequencing (scRNA-seq) has enabled the interrogation of heterogeneous cell populations in blood without cell type isolation and has already been employed in the study of myriad immune-related diseases (Gate et al., 2020; Kazer et al., 2020; the Accelerating Medicines Partnership Rheumatoid Arthritis and Systemic Lupus Erythematosus (AMP RA/SLE) Consortium et al., 2019; Uniken Venema et al., 2019). Recent studies employing scRNA-seq to study the role of immune cell subpopulations between healthy and ill patients, such as those for Crohn's disease (Martin et al., 2019), Tuberculosis (Cai et al., 2020), and COVID-19 (Lee et al., 2020), have identified cell type-specific disease relevant signatures in peripheral blood immune cells; however, these types of studies have been limited to large volume venous blood draws which can tax already ill patients, reduce the scope of studies to populations amenable to blood draws, and often require larger research teams to handle the patient logistics and sample processing costs and labor. In particular, getting repeated venous blood draws within a single day and/or multiple days at the subject's home has been a challenge for older people with frail skin and those on low dosage Acetylsalicylic acid(Bennett, 2020). This dependence on venous blood dramatically impacts our ability to understand the high temporal dynamics of health and disease.

Capillary blood sampling is being increasingly used in point-of-care testing and has been advised for obese, elderly, and other patients with fragile or inaccessible veins (Blicharz et al., 2018; Lenicek Krleza et al., 2015; Robison et al., 2009; Tang et al., 2017). The reduction of patient burden via capillary blood sampling could enable researchers to perform studies on otherwise difficult or inaccessible populations, and at greater temporal resolution. Additionally, capillary blood is being shown to be comparable to traditional venous blood draws for a variety of applications. For example,

Catala et al. have shown that 39 out of 45 clinically relevant metabolites had overlapping ranges between capillary blood vs traditional venous blood draws (Catala et al., 2018), and Toma et al. have shown strong correlation (Spearman correlation coefficient ≥ 0.95) between bulk RNA sequencing data between capillary and venous blood from the same donor (Toma et al., 2020). However, to date, scRNA-seq of human capillary blood has not yet been validated nor applied to study the immune system. In order to make small volumes of capillary blood (100 ul) amenable to scRNA-seq we have developed a platform which consists of a painless vacuum-based blood collection device, sample de-multiplexing leveraging commercial genotype data, and an analysis pipeline used to identify time-of-day and subject specific genes. The potential of our platform is rooted in enabling large scale studies of immune state variation in health and disease across people. The high-dimensional temporal transcriptome data could be paired with computational approaches to predict and understand emergence of pathological immune states. Most importantly, our platform makes collection and profiling of human immune cells less invasive, less expensive and as such more scalable than traditional methods rooted in large venous blood draws.

6.3 Results

6.3.1 Platform for low-cost interrogation of single-cell immune gene expression profiles

Our platform is comprised of a protocol for isolating capillary peripheral blood mononuclear cells (CPBMCs) using a touch activated phlebotomy device (TAP) (Blicharz et al., 2018), pooling samples to reduce per-sample cost using genome-based demultiplexing (Kang et al., 2018), and a computational package that leverages repeated sampling to identify genes that are differentially expressed in individuals or between time points, within subpopulations of cells (Figure 28**a**). Using a painless vacuum-based blood collection device such as the commercial FDA-approved TAP to collect capillary blood makes it convenient to perform at-home self-collected sampling and removes the need for a trained phlebotomist, increasing the ease of acquiring more samples. The isolation of CPBMCs is done using gradient centrifugation and red blood cells are further removed via a red blood cell lysis buffer. The cells from the different subjects are pooled, sequenced via scRNA-seq

using a single reagent kit, and demultiplexed (Kang et al., 2018) via each subject's single-nucleotide polymorphisms (SNPs), reducing the per-sample processing cost. By pooling the data across all 6 time points, and using a genotype-free demultiplexing software (popscle), we were able to identify which cells belonged to which subject across time points, removing the need for a separate genotyping assay to link subjects together across batches.

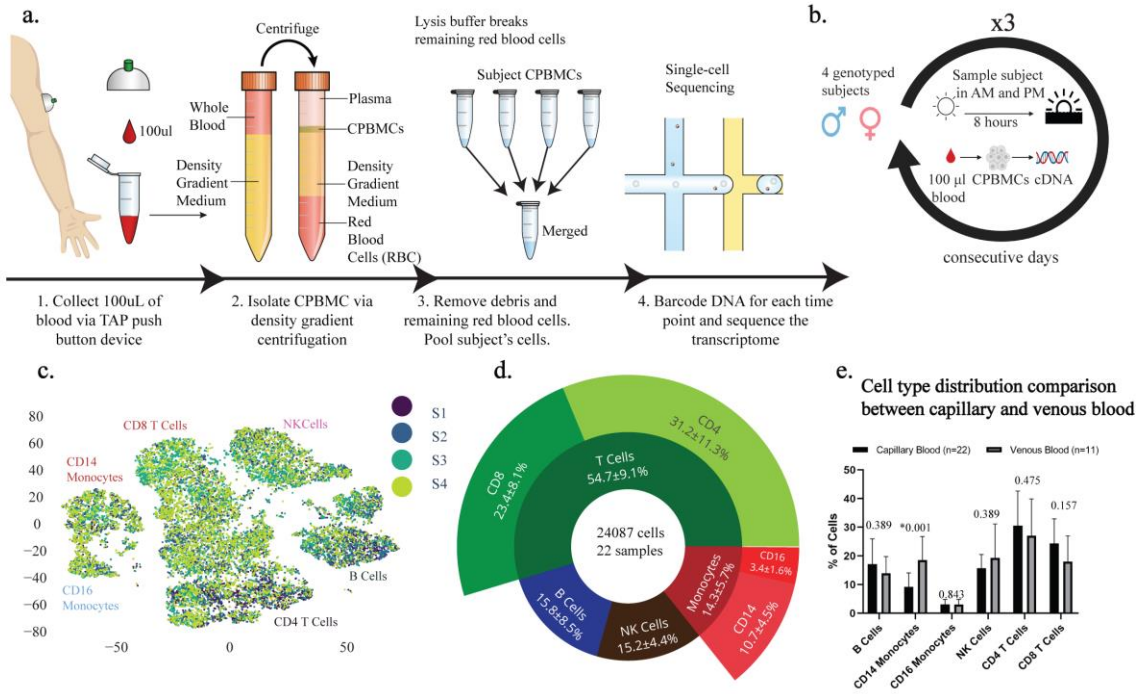


Figure 28. Experimental workflow and consistency of capillary blood sampling. (a) Experimental workflow for capillary blood immune profiling. 1. Blood is collected using the TAP device from the deltoid. 2. Capillary peripheral blood mononuclear cells (CPBMCs) are separated via centrifugation. 3. Red blood cells are lysed and removed, and samples from different subjects are pooled together. 4. Cell transcriptomes are sequenced using single-cell sequencing. **(b)** Time-course study design. CPBMCs are collected and profiled from 4 subjects (2 male, 2 female) each morning (AM) and afternoon (PM) for 3 consecutive days. **(c)** 2-dimensional t-SNE projection of the transcriptomes of all cells in all samples. Cells appear to cluster by major cell type (Fig. S6) **(d)** Immune cell type percentages across all samples shows stable cell type abundances (includes cells without subject labels). **(e)** Cell type ratios between capillary blood from this study, and venous blood from 3 other studies were the same, with the exception of CD14+ Monocytes, which are more abundant in venous blood (FDR < 0.05, 2-sided student t-test, multiple comparison corrected) The q-values are displayed for each cell type comparison.

6.3.2 Single-cell RNA sequencing (scRNA-seq) of low volume capillary blood recovers distinct immune cell populations stably across time

As a proof-of-concept, we leveraged our scRNA-seq of capillary blood platform to identify genes that exhibit diurnal behavior in subpopulations of cells and find subject-specific immune relevant gene signatures. We performed a three-day study in which we processed capillary blood from four subjects in the morning and afternoon, totaling 24,087 cells across 22 samples (Figure 28**b**). Major immune cell types such as T cells (CD4+, CD8+), Natural Killer cells, Monocytes (CD14+, CD16+), and B cells are present in all subjects and time points with stable expression of key marker genes (Figure 28**d**, Supplementary Figure 10), demonstrating that these signals are robust to technical and biological variability of CPBMC sampling (Figure 28**c**). In order to compare cell type distributions derived from our method with venous blood draws, we used data from 11 healthy subjects provided by three independent studies7,16,17 (Table S 11). CD14+ Monocytes make up a higher percentage of PBMCs in venous blood ($n = 11$) versus capillary blood ($n = 22$) ($FDR < 0.05$, 2-sided student t-test, multiple comparison corrected), while other cell types do not have a significant difference in distributions (Figure 28**e**).

6.3.3 High frequency scRNA-seq unveils new diurnal cell type-specific genes

Genes driven by time-of-day expression, such as those involved in leukocyte recruitment (He et al., 2018) and regulation of oxidative stress (Zhao et al., 2017), have been determined to play an important role in both innate and adaptive immune cells (Keller et al., 2009). Medical conditions such as atherosclerosis, parasite infection, sepsis, and allergies display distinct time-of-day immune responses in leukocytes (Pick et al., 2019), suggesting the presence of diurnally expressing genes that could be candidates for optimizing therapeutic efficacy via time-of-day dependent administration. However, studies examining diurnal gene expression in human blood have been limited to whole blood gene panels via qPCR, or bulk RNA-seq (Braun et al., 2018; Kusanagi et al., 2008; Lech et al., 2016).

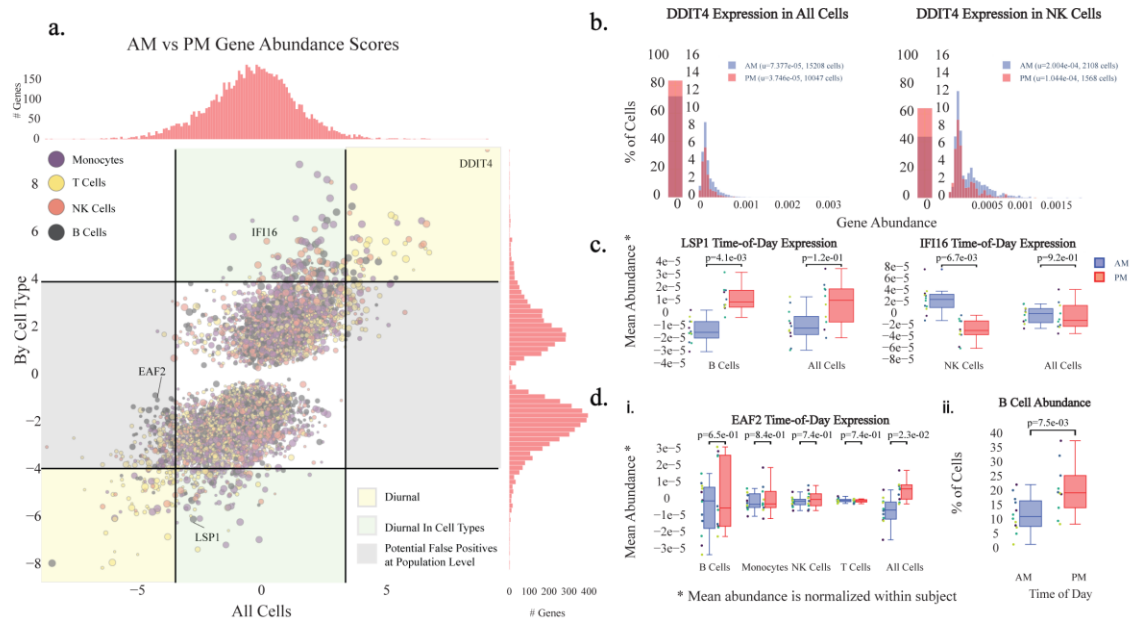


Figure 29. Diurnal variability in subpopulations of capillary blood. (a) Magnitude (Z-score) of the difference in AM vs PM gene expression across the whole population of cells (x) vs the cell type with the largest magnitude Z-score (y). Points above or below the significance lines (FDR < 0.05, multiple comparison correction) display different degrees of diurnality. The size of each marker indicates the abundance of the gene (the largest percent of cells in a subpopulation that express this gene). **(b)** Distribution of expression of DDIT4, a previously identified circadian rhythm gene(Braun et al., 2018), shows diurnal signal across all cells, as well as individual cell types, such as natural killer (NK) cells. u indicates the mean fraction of transcripts per cell (gene abundance). **(c)** Example of newly identified diurnal genes, LSP1 and IFI16 that could be missed if analyzed at the population level **(d)** Example of a gene, EAF2, that could be falsely classified as diurnal (i) without considering cell type subpopulations due to a diurnal B cell abundance shift (ii).

Leveraging our platform, which enables single-cell studies of temporal human immune gene expression, we detected 395 genes (FDR < 0.05, multiple comparison corrected) exhibiting diurnal activity within at least one cell subpopulation (Figure 29**a**). Among the 20 top diurnally classified genes, we found that 35% of those genes were previously correlated with circadian behavior (**Table S 8**), such as DDIT4 (Braun et al., 2018) (Figure 29**b**), SMAP225, and PCPB126. However, only 119/395 (30.1%) of these genes are detected as diurnal at the whole population level (FDR < 0.05, multiple comparison corrected), suggesting there may be many more diurnally-varying genes than previously discovered. For example, IFI16 and LSP1 (Figure 29**c**) have diurnal expression only in

NK cells and B cells, respectively, and display previously unreported transcriptional diurnal patterns. In particular, LSP1 has been implicated in numerous leukemias and lymphomas of B cell origin (Pulford et al., 1999). Given previous evidence of increased efficacy of time-dependent chemotherapy administration (Hermida et al., 2009; Lévi et al., 2007) and tumor cells exhibiting out-of-sync behavior compared to normal cells (Ramsey and Ellisen, 2011), understanding LSP1's diurnal expression pattern can potentially guide timely administration of candidate therapeutics. Out of the identified 395 diurnally-varying genes, 114 (29%) are considered druggable under the drug gene interaction database (https://www.dgidb.org/).

6.3.4 scRNA-seq profiling distinguishes diurnal gene expression from cell type abundance changes

We also detected 406 genes (FDR < 0.05, multiple comparison corrected) exhibiting diurnal behavior when analyzed at the population level, such as EAF2, that do not display diurnal variation in any of our major cell types (Figure 29**d.i**). Such false positives may come from diurnal shifts in cell type abundance rather than up- or down-regulation of genes. In the case of EAF2, which is most abundant in B cells, we hypothesized that the diurnality detected at the population level was a result of an increase of B cell abundance in the afternoon, and verified this in our data ($p = 7.5 \times 10^{-3}$, one-sided student-t test) (Figure 29**d.ii**). This finding highlights the importance of looking at expression within multiple cell types to avoid potentially misleading mechanistic hypotheses.

6.3.5 Individuals exhibit robust cell type-specific differences in genes and pathways relevant to immune function

Gene expression studies of isolated cell subpopulations across large cohorts of people have revealed a high degree of variability between individuals that cannot be accounted for by genetics alone, with environmental effects that vary over time likely playing a critical role (Thomas, 2010; Ye et al., 2014). Furthermore, these transcriptomic differences have been linked to a wide range of therapeutic responses, such as drug-induced cardiotoxicity (Matsa et al., 2016). However, while immune system composition and expression has been shown to be stable over long time periods within an individual,

acute immune responses generate dramatic immune system changes, meaning that large single time point population studies are unable to establish whether variability between individuals is stable or the result of dynamic response to stimuli (Brodin and Davis, 2017).

To probe the stability of individual gene expression signatures at the single-cell level, we used our pipeline to identify genes whose variation in gene expression is most likely caused by intrinsic intersubject differences rather than high frequency immune system variability. We compared the mean gene expressions of all time points between subjects in all cell types and identified 1284 genes ($FDR < 0.05$, multiple comparison corrected) that are differentially expressed in at least one subpopulation of cells. Like Whitney et al., we found MHC class II genes, such as HLA-DRB1 and HLA-DRA (Figure 30**a**) to be among the largest sources of variation between subjects (Whitney et al., 2003). Additionally, we found that DDX17, which was classified by Whitney et al. as a gene with high intersubject variability, but low intrasubject variability via repeat sampling over longer time scales, may be a new class of temporally varying gene that varies by day of week, having consistently increasing expression each subsequent sampling day. This stresses the importance of high frequency sampling for identifying genes with the most intrinsic interindividual variability.

6.3.6 Numerous subject-specific genes are revealed in specific immune cell types

Within the 1284 genes with intrinsic interindividual variability, we found myriad disease-relevant genes for all subjects and cell types, which can be explored at our interactive online portal (https://capblood-seq.caltech.edu). As just one example, subject S1's monocytes have a consistent downregulation ($p = 9.1 \times 10^{-7}$, two-sided student t-test) of LIPA, a gene that is implicated in Lysosomal Acid Lipase Deficiency (Figure 30**c**). Given the low abundance of monocytes in blood samples, such findings would typically only be discovered from a targeted blood test or RNA sequencing of isolated monocytes, either of which would only be performed if the disease was already suspected; this showcases how automated discovery in heterogeneous cell populations can be leveraged for personalized, preventative care.

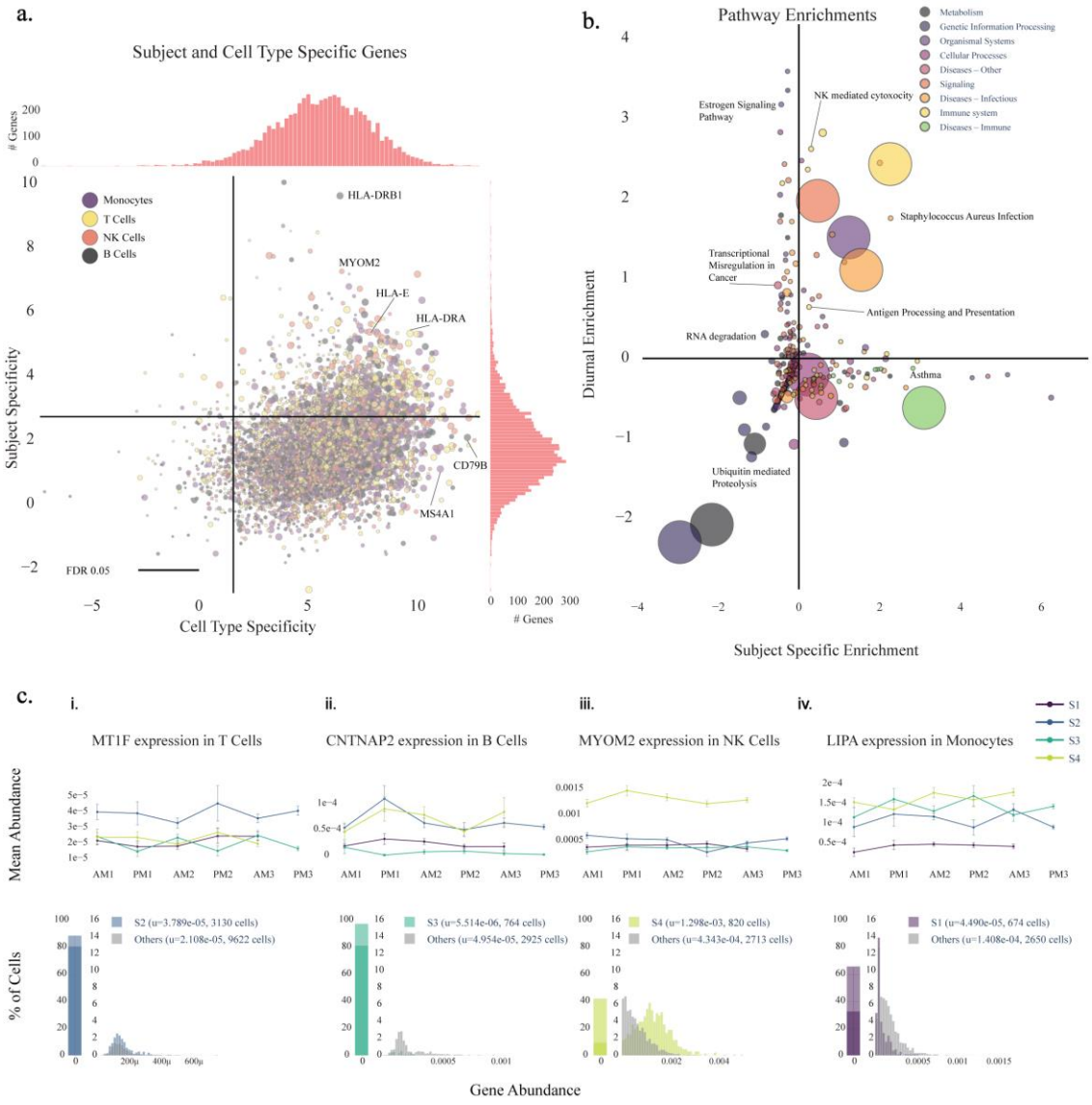


Figure 30. Subject variability in immune and disease-relevant genes and pathways. (a) Magnitude ($\log_2$ F statistic) of the variability in expression of genes between different cell types (x) and between subjects (y). 1284/7034 (18.3%) of genes are above the subject specificity significance line (FDR < 0.05, multiple comparison correction) and are classified as subject-specific. Several MHC class II genes (HLA-X) are strongly subject-specific, consistent with previous findings(Whitney et al., 2003). **(b)** KEGG pathways grouped into categories and their enrichment (Z-score from 2-proportion Z-test) among the top 250 diurnally and subject-varying genes vs all genes. Immune system and disease pathways are significantly enriched ($p = 0.029$), supportive of the conclusion that immune and disease-related genes are highly subject dependent. The large circles indicate the enrichment of the category overall, and the sizes of the smaller pathway points indicate the number of genes associated with the pathway. **(c)** Subject

and cell type specific gene examples for each subject and cell type with the upper row displaying the trace of mean gene expression across time-points and the bottom row showing gene abundance shifts for the subjects of interest.

6.3.7 Immune function and disease pathways are enriched in subject-specific genes

Given that genes do not act alone, we also found cell type-specific pathway differences among subjects. In particular, Subject 2's S100A8, S100A9, and S100A12 genes, calcium-binding proteins that play an important role in macrophage inflammation, are significantly downregulated in monocytes ($p_{S100A8} = 1.3 \times 10^{-5}$, $p_{S100A9} = 9.0 \times 10^{-5}$, $p_{S100A12} = 3.0 \times 10^{-4}$, two-sided student t-test) compared to other subjects (Supplementary Figure 11). We further explored our findings by inspecting the pathways that are most enriched in individual and time-varying genes, and found that genes that are implicated in immune system function ($p = 0.085$) and immune diseases ($p = 0.029$) are more present in subject-specific genes (Figure 30**b**). This stands in contrast to pathways of core cellular functions such as genetic information processing ($p = 0.029$) and metabolism ($p = 0.095$), which are less present in subject-specific genes.

6.4 Discussion

Genome and transcriptome sequencing projects have unveiled millions of genetic variants and associated gene expression traits in humans (Farh et al., 2015; Lappalainen et al., 2013). However, large-scale studies of their functional effects performed through venous blood draws require tremendous effort to undertake, and this is exacerbated by the cost and complexity of single-cell transcriptome sequencing. Efforts such as the Immune Cell Census (The Immune Cell Census) are already underway to perform single-cell profiling of large cohorts, but reliance on venous blood draws of PBMCs will likely limit the diversity and temporal resolution of their sample pool. Our platform gives researchers direct, scalable access to high resolution immune system transcriptome information of human subjects, lowering the barrier of entry for myriad new research avenues. Examples of such studies include: 1. tracking vulnerable populations over time, such as monitoring clonal expansion of CD8+ T cells in Alzheimer's disease progression (Gate et al., 2020), 2. profiling of individuals who are under home care to track disease progression and therapeutic response, such

as transplant patients and people under quarantine, and 3. tracking how stress, diet, and environmental conditions impact the immune system at short and long time scales, particularly in underrepresented populations who do not have easy access to hospitals or research institutions, such as people in rural or underdeveloped areas. Larger, more diverse subject pools coupled with time course studies of cell type gene expression in health and disease will have a dramatic impact on our ability to understand the baseline and variability of immune function.

6.5 Online content

Online web portal is available to explore data presented in the main figures for study summary, diurnal and subject specific genes via https://capblood-seq.caltech.edu.

6.6 Methods

6.6.1 Human study cohort

This study was conducted at Caltech. Four healthy adults (2 male, 2 female) were recruited (Supplementary Figure 12). All participants provided written informed consent. The study was approved by the Institutional Review Board (IRB) at Caltech and all methods were performed in compliance with relevant guidelines and regulations. The blood collection took place in a non-BSL room to make sure the subjects were not exposed to pathogens. Subject blood was collected roughly 8 h apart over three consecutive days.

6.6.2 CPBMC isolation

100 µl of capillary blood was collected via push-button collection device (TAP from Seventh Sense Biosystems). For each blood draw, the site of collection was disinfected with an alcohol wipe and the TAP device was placed on the deltoid of the subject per device usage instructions. The button was pushed, and then blood was collected for 2–7 min until the indicator turned red. Blood was extracted from the TAP device by gently breaking the seal foil, and mixed with PBS + 2% FBS to 1 ml. The mixture was slowly added to the side of a SepMate tube (SepMate-15 IVD, Stem Cell

Technologies) containing 4.5 ml of Lymphoprep (#07811, Stem Cell Technologies) and centrifuged for 20 min at 800 RPM. Approximately 900 µl of CPBMC layer was extracted below the plasma layer. To further remove red blood cells, 100 µl of red blood cell lysis buffer (eBioscience 10× RBC Lysis Buffer, #00-4300-54) was added to the CPBMCs and incubated at RT for 15 min. The CPBMC pellet was washed twice with PBS and centrifuged at 400 rpm for 5 min. Cells were counted using trypan blue via an automated detector (Countess II Automated Cell Counter) and subjects' cells were pooled together for subsequent single-cell RNA sequencing.

6.6.3 Single-cell RNA sequencing

Subject pooled single-cell suspensions were loaded onto a Chromium Single Cell Chip (10X Genomics) based on manufacturer's instructions (targeted 10,000 cells per sample, 2500 cells per person per time point). Captured mRNA was barcoded during cDNA synthesis and pooled for Illumina sequencing (Chromium Single Cell 3′ solution—10X Genomics). Each time point was barcoded with a unique Illumina sample index, and then pooled together for sequencing in a single Illumina flow cell. The libraries were sequenced with an 8-base index read, 26-base read 1 containing cell-identifying barcodes and unique molecular identifiers (UMIs), and a 91-base read 2 containing transcript sequences on a NovaSeq 6000.

6.6.4 Single-cell dataset generation

FASTQ files from Illumina were demultiplexed and aligned using Cell Ranger v3.0 (https://support.10xgenomics.com/single-cell-gene-expression/software/pipelines/latest/what-is-cell-ranger) and the hg19 reference genome with all options set to their defaults.

6.6.5 Sample demultiplexing

FASTQ files from the single-cell sequencing Illumina libraries were aligned against the hg19 (human) reference genome using Cellranger v3.0 count function. SNPs were detected in the aligned data using freebayes (https://github.com/ekg/freebayes), which creates a combined variant call

format (VCF) file, one per sample. SNPs were then grouped by cell barcode using popscle dsc-pileup (https://github.com/statgen/popscle). The SNP files for all samples were then merged into a single dsc-pileup file, and cell barcodes were disambiguated by providing a unique identifier per sample. Freemuxlet (popscle freemuxlet) was then run with default parameters to group cells into 4 subjects. This generates a probability of whether each cell barcode belongs to each subject, given the detection of single nucleotide polymorphism (SNPs) in reads associated with that cell barcode. Each cell was then assigned to the subject with the highest probability. Cells with low confidence (ambiguous cells) and high confidence in more than one subject (multiplets) were discarded, using popscle's default confidence thresholds. See the README at https://github.com/thomsonlab/capblood-seq for detailed instructions.

6.6.6 Debris removal

The raw cell gene matrix provided by Cell Ranger contains gene counts for all barcodes present in the data. To remove barcodes representing empty or debris-containing droplets, a debris removal step was performed. First, a UMI count threshold was determined that yielded more than the expected number of cells based on original cell counts (15,000). All barcodes below this threshold were discarded. For the remaining barcodes, principal component analysis (PCA) was performed on the log-transformed cell gene matrix, and agglomerative clustering was used to cluster the cells. The number of clusters was automatically determined by minimizing the silhouette score among a range of numbers of clusters (6 to 15). For each cluster, a barcode dropoff trace was calculated by determining the number of barcodes remaining in the cluster for all thresholds in increments of 50. These cluster traces were then clustered into two clusters using agglomerative clustering—the two clusters representing "debris" with high barcode dropoff rates and "cells" with low barcode drop-off rates. All clusters categorized as "debris" were then removed from the data.

6.6.7 Gene filtering

Before cell typing, genes that have a maximum count less than 3 are discarded. Furthermore, after cell typing, any genes that are not present in at least 10% of one or more cell types are discarded.

6.6.8 Data normalization

Gene counts were normalized by dividing the number of times a particular gene appears in a cell (gene cell count) by the total gene counts in that cell. Furthermore, for visualization only, the gene counts were multiplied by a constant factor (5000), and a constant value of 1 was added to avoid zeros and then log transformed.

6.6.9 Cell typing

We used single cell Variational Inference (scVI) to transform the raw cell gene expression data into a 10-dimensional variational autoencoder latent space (Lopez et al., 2018). The variational autoencoder is conditioned on sample batch, creating a latent space which is independent of any batch-specific effects. The variational auto-encoder parameters: learning rate = 1e−3, number of epochs = 50.

Agglomerative clustering (sci-kit learn) was used to generate clusters from the latent cell gene expression data. These clusters were then annotated based on known cell type marker genes (Supplementary Figure 10).

In order to resolve specific cell subtypes, such as those of T cells and Monocytes, we specified 13–15 clusters as an input for agglomerative clustering. For each study, we started at 13 clusters and incremented until all 4 major cell types and 2 subtypes were separable. In cases where agglomerative clustering yielded multiple clusters of the same cell type, these clusters were merged into a single cell type for analysis.

6.6.10 Venous and capillary blood comparison

In order to compare venous blood cell type distributions to capillary blood, raw gene count data was downloaded from each of the respective studies, and we performed the same cell typing pipeline as for our capillary data, first projecting the data into a latent space via scVI, followed by agglomerative clustering and manual annotation based on known cell type marker genes.

6.6.11 Diurnal gene detection

To identify genes that exhibit diurnal variation in distinct cell types, we developed a statistical procedure that detects robust gene expression differences between morning (AM) and evening (PM) samples. Given that gene expression is different between subjects, we first normalize the mean gene expression within each subject for each cell type.

$$\mu'_{g_i,s_j,c_n,k} = \mu_{g_i,s_j,c_n,k} - \left(\frac{\sum_{k=1}^{N_{s_j}} 1_{k\in AM}\mu_{g_i,s_j,c_n,k}}{2\sum_{k=1}^{N_{s_j}} 1_{k\in AM}} + \frac{\sum_{k=1}^{N_{s_j}} 1_{k\in PM}\mu_{g_i,s_j,c_n,k}}{2\sum_{k=1}^{N_{s_j}} 1_{k\in PM}}\right) \quad (1)$$

We take the mean gene expression μ for each gene g_i in all samples k for cell type c_n and subject s_j and renormalize it into μ' by subtracting the equally weighted mean of AM and PM samples (Eq. (1)). We then split the mean gene values into an AM group and a PM group and perform a statistical test (two-tailed student-t test) to determine whether to reject the null hypothesis that gene expression in AM and PM samples come from the same distribution. We then perform Benjamini–Hochberg multiple comparison correction at an FDR of 0.05 on all gene and cell type p-values to determine where to plot the significance threshold. For plotting the genes, we choose the Z-statistic corresponding to the minimum p-value among cell types for that gene. To determine diurnality at the population level, we repeated the procedure above with all cells pooled into a single cell type.

6.6.12 Subject and cell type specific gene detection

To classify genes as subject specific, we detect genes with mean gene expression levels that are robustly different between subjects in at least one cell type. For each cell type cn and gene gi, we create subject groups containing the mean gene expression values from each sample. To determine whether the gene expression means from the different subjects do not originate from the same distribution, we perform an ANOVA one-way test to get an F-statistic and p-value for each gene. We then perform Benjamini–Hochberg multiple comparison correction at an FDR of 0.05 on all gene and cell type p-values. For plotting the genes, we chose the F-statistic corresponding to the minimum p-value among cell types for that gene.

For determining gene cell type specificity, we performed a similar procedure. In particular, for each gene gi, we create cell type groups containing the mean gene expression values for that cell type from each sample. We then perform a one-way ANOVA, and Benjamini–Hochberg multiple comparison correction at an FDR of 0.05.

6.6.13 Pathway enrichment analysis

Pathways from the KEGG database (Python bioservices package) were used to calculate pathway enrichment for genes that were among the top 250 most diurnal and individual specific. All remaining genes present in the data were considered background. In order to normalize for gene presence across pathways, each gene was weighted by dividing the number of pathways in which that gene appears. For each KEGG pathway (Kanehisa, 2000, 2019; Kanehisa et al., 2019), the test statistic for a two-proportion z-test (Python statsmodel v0.11.1) is used to determine pathway enrichment. From the top level pathway classes, we broke out “Diseases” into “Other”, “Immune Diseases”, and “Infectious Diseases” and separated “Immune System” from “Organismal System” to understand diurnal and subject-specific genes in an immune relevant context.

6.6.14 Figure art

All drawings (Figure 28**a,b**, Supplementary Figure 11) are generated using BioRender.com. Figure 28**e** was generated using GraphPad Prism 8.3.1.

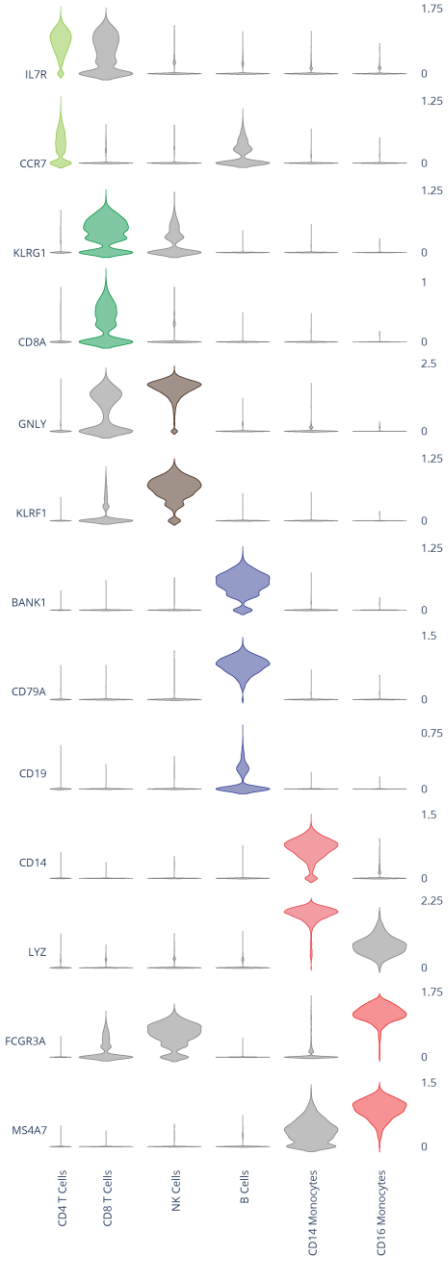
6.7 Data availability

Gene expression matrix and relevant metadata are available on https://data.caltech.edu/records/1407. FASTQ files are not being released to protect the identity of the subjects.

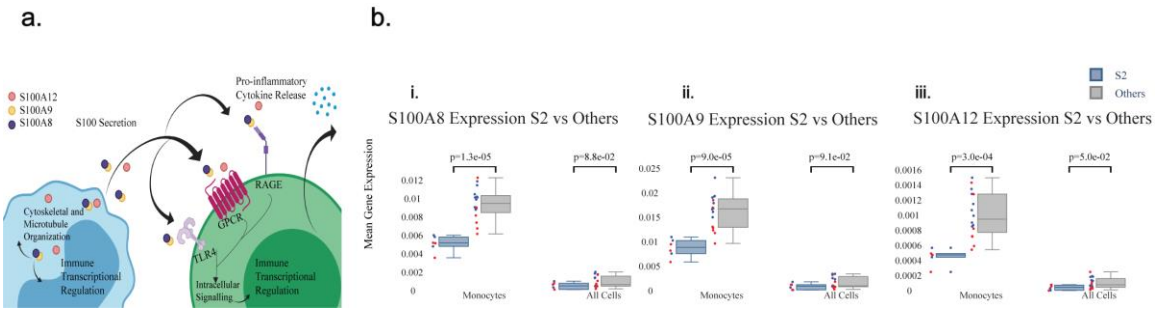
6.8 Code availability

Custom code made for diurnal and subject specific gene detection is available on https://github.com/thomsonlab/capblood-seq.

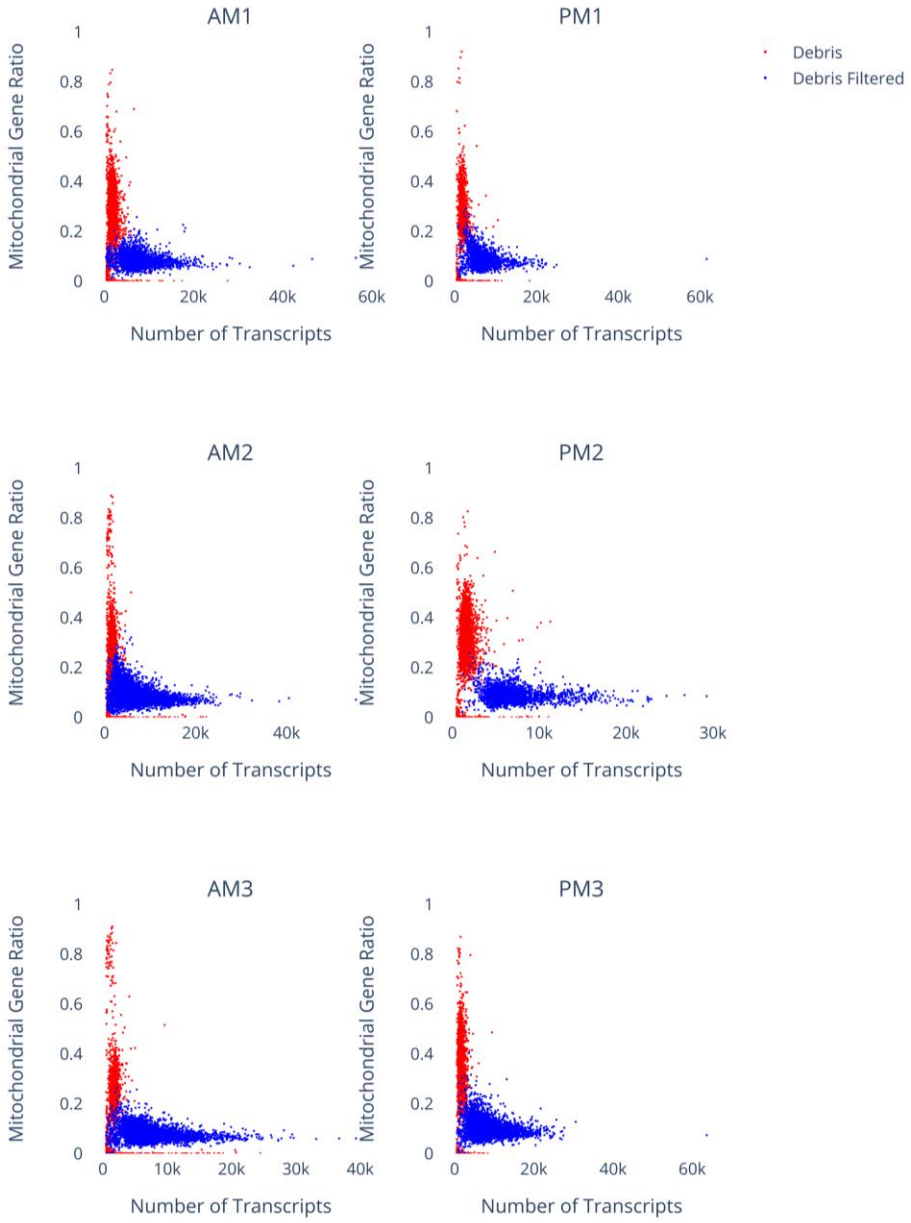
6.9 Supplemental Figures



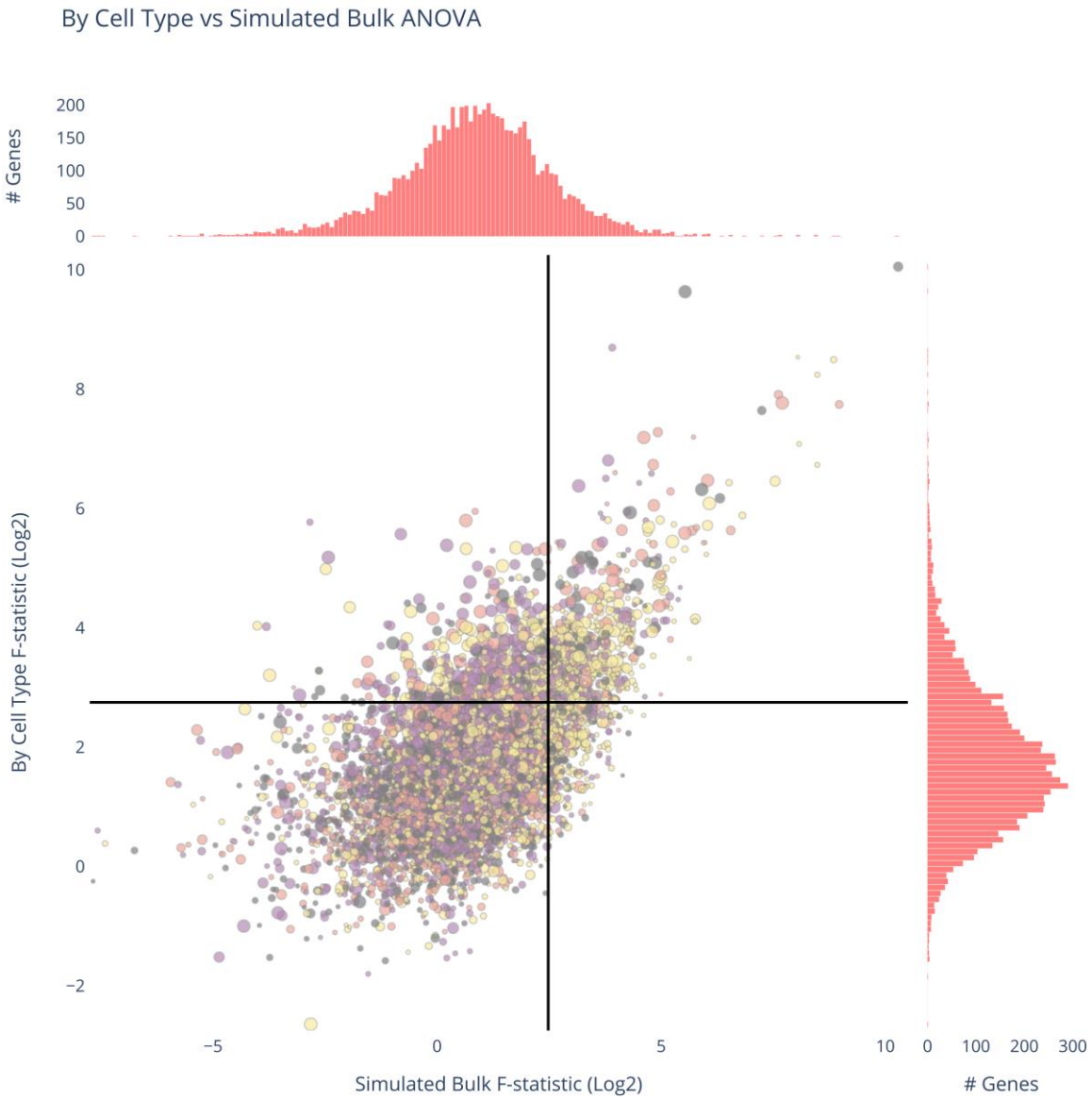
Supplementary Figure 10. Cell type marker gene expression in cell clusters. Violin plots of log-normalized gene expression (y-axis, right hand side) for cell type markers (y-axis, left hand side) used to annotate cell clusters (x-axis) for known cell types. The colors correlate to clusters from **Figure 28**.d.



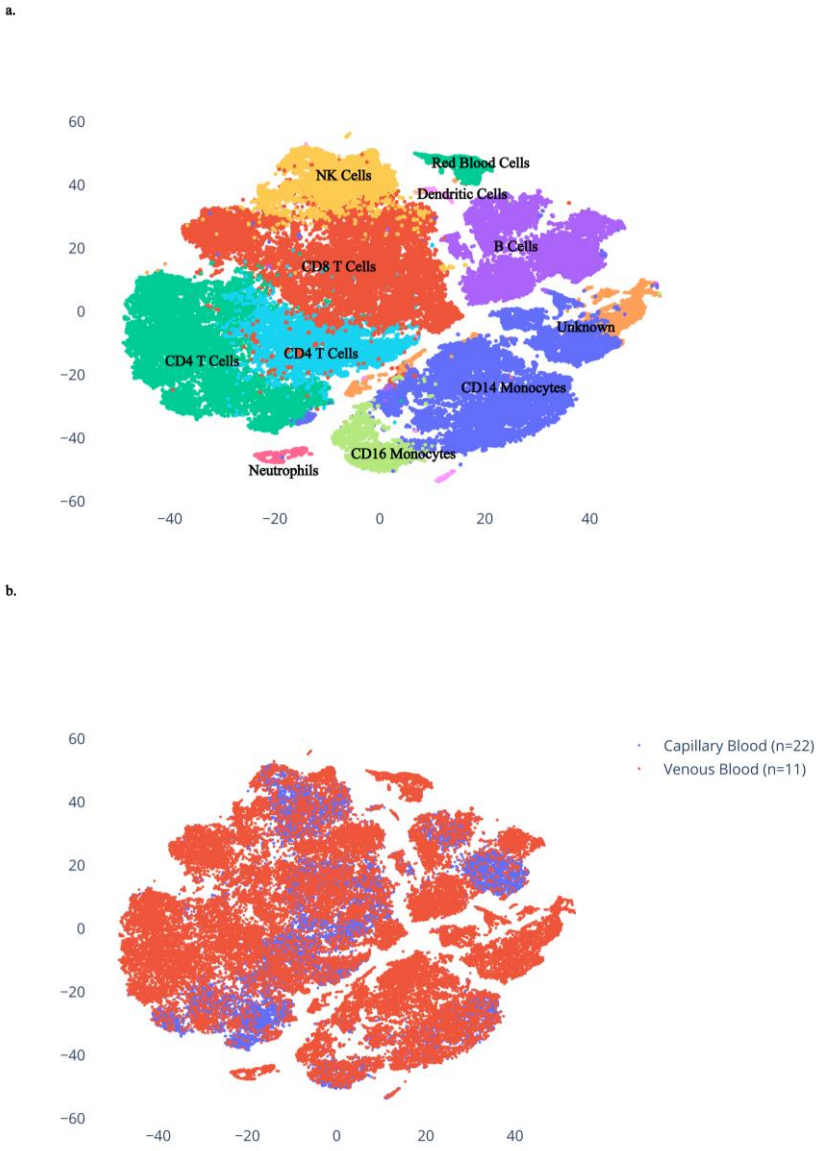
Supplementary Figure 11. S100 pathway exhibits individual-specific regulation. (a) Simple schematic illustrating the role of S100A8, S100A9, and S100A12 genes in immune regulation. **(b)** Normalized mean gene expression of S100A8, S100A9, and S100A12 genes for S2 showing significant downregulation in monocytes as compared to all cells.



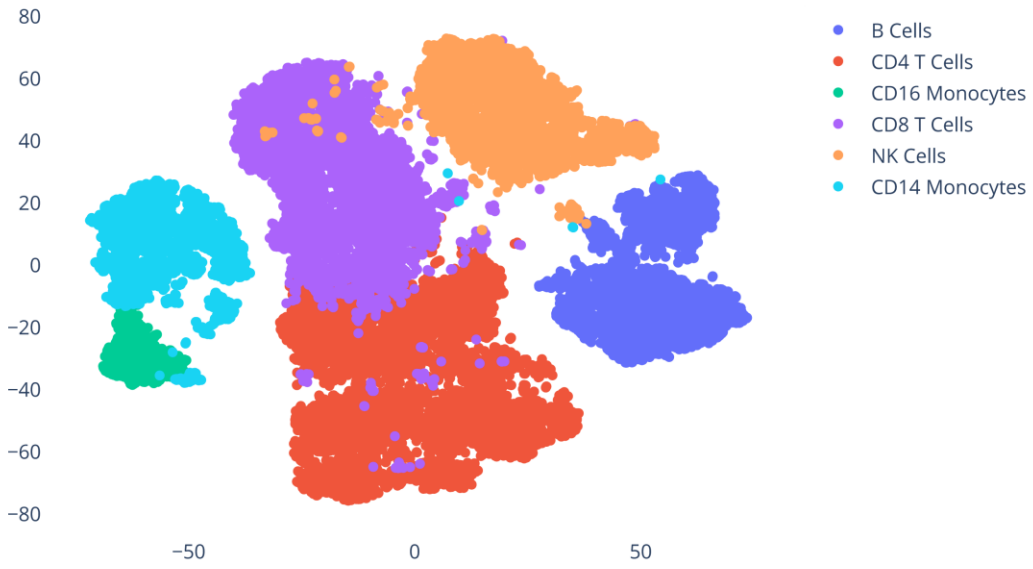
Supplementary Figure 12. Characterization of debris removal pipeline across each time sample. Scatter plots of the total number of transcripts (UMIs) detected for each barcode (x-axis), and the ratio of transcripts that are mitochondrial (y-axis). These barcodes are the union of barcodes called by 10X Cellranger and our debris filtering pipeline. Barcodes colored red were flagged as debris and removed. The debris filtering pipeline appears to detect barcodes that have both a low transcript count, and a high mitochondrial gene ratio, or a rare number of cells that appear to have 0 mitochondrial genes. The counts of barcodes removed for each sample are in Table S6.



Supplementary Figure 13. Comparison of individual specificity by cell type vs in simulated bulk data. Magnitude (log2 F statistic) of the variability in expression of genes between subjects, accounting for each cell type separately (y) and in simulated bulk (x). 1284/7034 (18.3%) of genes are above the subject specificity significance line (FDR < 0.05, multiple comparison corrected) and are classified as subject-specific. Of these, only 637/1284 (49.6%) are also detected as subject-specific when simulating bulk RNA reads, despite the significantly lower multiple comparison correction burden (7034 tests as compared to 28,136 tests in the cell type case).



Supplementary Figure 14. Merged projection of capillary and venous blood cells. Capillary blood cells from this study (n=22) and venous blood cells from 3 other studies (n=11) were projected into a joint latent space using scVI. (a) Agglomerative clustering with n=13 clusters was performed to identify cell types, and annotated using known cell type markers (b) Capillary blood cells cluster together with venous blood cells, with the exception of one cluster of B cells unique to capillary cells, as well as 3 cell types unique to the venous blood sample: red blood cells, dendritic cells, and neutrophils, which are likely filtered out via laboratory procedures and the computational debris filtering pipeline.



Supplementary Figure 15. Immune cell type clusters detected in capillary blood. 2-dimensional t-SNE projection of the transcriptomes of all cells in all samples obtained from agglomerative clustering of latent gene expression. Cell clusters were annotated and grouped based on the markers presented in Table S2. Small unidentifiable clusters were not included in the figure.

Table S 8. Genes that ranked in top 20 that had pre-existing literature tying to circadian/diurnal expression

Gene	DOI Reference
DDIT4	10.7554/eLife.20214.001, 10.1073/pnas.1800314115
SMAP2	10.1038/s41398-019-0671-7
RPL19	10.1128/MCB.00701-15
RPS9	10.1073/pnas.1515308112
PCPB1	10.1038/s41556-019-0441-z
RPS2	10.1073/pnas.1601895113
RBM3	10.1038/srep02054
COX5B	10.1152/physiolgenomics.00066.2007

Table S 9. Marker genes used to annotate clusters with specified cell population identity.

Cells	Marker Genes
CD14 Monocytes	CD14, LYZ
CD16 Monocytes	FCGR3A, MS4A7
CD4 T Cells	IL7R,CCR7
CD8 T Cells	KLRG1, CD8A, CD8B
Natural Killer (NK) Cells	GNLY, KLRF1, KLRD1
B Cells	BANK1, CD79A, CD79B, CD19

Table S 10. Subject age and demographics. All subjects indicated to be healthy during the study.

Subject	Age	Gender
S1	32	M
S2	41	M
S3	34	F
S4	26	F

Table S 11. Details of studies used to get healthy venous blood single-cell RNA sequencing dataset for comparison with capillary blood.

Subject	Age	Gender	Corresponding DOI	Corresponding Study Identification
S1	21	M	https://doi.org/10.1038/s41598-020-59827-1	Pre-THC-S1
S2	21	M	https://doi.org/10.1038/s41598-020-59827-1	Pre-THC-S2
S3	63	F	https://doi.org/10.1126/sciimmunol.abd1554	Sample 5_Normal 1 scRNA-seq [SW107]
S4	54	F	https://doi.org/10.1126/sciimmunol.abd1554	Sample 13_Normal 2 scRNA-seq [SW115]
S5	67	F	https://doi.org/10.1126/sciimmunol.abd1554	Sample 14_Normal 3 scRNA-seq [SW116]
S6	63	M	https://doi.org/10.1126/sciimmunol.abd1554	Sample 19_Normal 4 scRNA-seq [SW121]
S7	50	M	https://doi.org/10.1073/pnas.1907883116	CT1
S8	70	F	https://doi.org/10.1073/pnas.1907883116	CT2
S9	60	F	https://doi.org/10.1073/pnas.1907883116	CT3
S10	70	F	https://doi.org/10.1073/pnas.1907883116	CT4
S11	80	M	https://doi.org/10.1073/pnas.1907883116	CT5

Table S 12. Number of genes in different cell types that is specific to each subject.

	B Cells	Monocytes	NK Cells	T Cells	Any
S1	55	67	58	269	400
S2	24	94	49	58	190
S3	55	149	70	150	353
S4	49	36	34	44	131

Table S 13. Statistics for debris removal pipeline.

	Cellranger Called	Removed	Added	Final # Cells	% Removed
AM1	5808	2662	21	3167	45.83
PM1	3144	1302	12	1854	41.41
AM2	8772	2037	20	6755	23.22
PM2	6172	3587	0	2585	58.12
AM3	6684	1408	10	5286	21.07
PM3	7974	2370	4	5608	29.72

	Description	File Name
Table S7	Differential expression analysis for each cluster and cell type of the combined capillary blood (n=22) dataset	cluster_differential_expression.xlsx
Table S8	Differential expression analysis for all clusters between capillary blood (n=22, this study), and venous blood (n=11, external studies)	capillary_vs_venous_differential_expression.xlsx

BIBLIOGRAPHY

[1] Achim, K. et al. (2015). High-throughput spatial mapping of single-cell RNA-seq data to tissue of origin. *Nat. Biotechnol. 33*, 503–509.

[2] Adachi, K., Enoki, T., Kawano, Y., Veraz, M., and Nakai, H. (2014). Drawing a high-resolution functional map of adeno-associated virus capsid by massively parallel sequencing. *Nat. Commun. 5*, 3075.

[3] Aharon, L., Aharoni, S.-L., Radisky, E.S., and Papo, N. (2020). Quantitative mapping of binding specificity landscapes for homologous targets by using a high-throughput method. *Biochem. J. 477*, 1701–1719.

[4] Amirinejad, R., Rezaei, M., and Shirvani-Farsani, Z. (2020). An update on long intergenic noncoding RNA p21: a regulatory molecule with various significant functions in cancer. *Cell Biosci. 10*, 82.

[5] Arnold, F.H. (1998). Design by Directed Evolution. *Acc. Chem. Res. 31*, 125–131.

[6] Arruda, V.R. et al. (2001). Lack of germline transmission of vector sequences following systemic administration of recombinant AAV-2 vector in males. *Mol. Ther. J. Am. Soc. Gene Ther. 4*, 586–592.

[7] Baker, B.J., Akhtar, L.N., and Benveniste, E.N. (2009). SOCS1 and SOCS3 in the control of CNS immunity. *Trends Immunol. 30*, 392–400.

[8] Bartlett, J.S., Samulski, R.J., and McCown, T.J. (1998). Selective and Rapid Uptake of Adeno-Associated Virus Type 2 in Brain. *Hum. Gene Ther. 9*, 1181–1186.

[9] Bartlett, J.S., Kleinschmidt, J., Boucher, R.C., and Samulski, R.J. (1999). Targeted adeno-associated virus vector transduction of nonpermissive cells mediated by a bispecific F(ab'gamma)2 antibody. *Nat. Biotechnol. 17*, 181–186.

[10] Batista, A.R. et al. (2020). *Ly6a* Differential Expression in Blood–Brain Barrier Is Responsible for Strain Specific Central Nervous System Transduction Profile of AAV-PHP.B. *Hum. Gene Ther. 31*, 90–102.

[11] Bedbrook, C.N., Deverman, B.E., and Gradinaru, V. (2018). Viral strategies for targeting the central and peripheral nervous systems. *Annu. Rev. Neurosci. 41*, 323–348.

[12] Bennett, D. (2020). Personal Communication.

[13] Berto, S., Liu, Y., and Konopka, G. (2020). Genomics at cellular resolution: insights into cognitive disorders and their evolution. *Hum. Mol. Genet. 29*, R1–R9.

[14] Betley, J.N., and Sternson, S.M. (2011). Adeno-Associated Viral Vectors for Mapping, Monitoring, and Manipulating Neural Circuits. *Hum. Gene Ther. 22*, 669–677.

[15] Blanchard, J.W. et al. (2020). Reconstruction of the human blood-brain barrier in vitro reveals a pathogenic mechanism of APOE4 in pericytes. *Nat. Med. 26*, 952–963.

[16] Blicharz, T.M. et al. (2018). Microneedle-based device for the one-step painless collection of capillary blood samples. *Nat. Biomed. Eng. 2*, 151–157.

[17] Bloom, J.D. (2015). Software for the analysis and visualization of deep mutational scanning data. *BMC Bioinformatics 16*, 168.

[18] Booeshaghi, A.S., and Pachter, L. (2021). Normalization of single-cell RNA-seq counts by $\log(x + 1)$ or $\log(1 + x)$. *Bioinformatics 37*, 2223–2224.

[19] Braun, R. et al. (2018). Universal method for robust detection of circadian state from gene expression. *Proc. Natl. Acad. Sci. 115*, E9247–E9256.

[20] Brodin, P., and Davis, M.M. (2017). Human immune system variation. *Nat. Rev. Immunol. 17*, 21–29.

[21] Burger, D., Lou, J., Dayer, J.-M., and Grau, G.E. (1997). Both soluble and membrane-associated TNF activate brain microvascular endothelium: relevance to multiple sclerosis. *Mol. Psychiatry 2*, 113–116.

[22] Cahoy, J.D. et al. (2008). A Transcriptome Database for Astrocytes, Neurons, and Oligodendrocytes: A New Resource for Understanding Brain Development and Function. *J. Neurosci. 28*, 264–278.

[23] Cai, Y. et al. (2020). Single-cell transcriptomics of blood reveals a natural killer cell subset depletion in tuberculosis. *EBioMedicine 53*, 102686.

[24] Calcedo, R., Chichester, J.A., and Wilson, J.M. (2018). Assessment of humoral, innate, and T-cell immune responses to adeno-associated virus vectors. *Hum. Gene Ther. Methods 29*, 86–95.

[25] Cao, L., Wang, Z., and Wan, W. (2018). Suppressor of Cytokine Signaling 3: Emerging Role Linking Central Insulin Resistance and Alzheimer's Disease. *Front. Neurosci. 12*, 417.

[26] Cao, Z.-J., Wei, L., Lu, S., Yang, D.-C., and Gao, G. (2020). Searching large-scale scRNA-seq databases via unbiased cell embedding with Cell BLAST. *Nat. Commun. 11*, 3458.

[27] Capellera-Garcia, S. et al. (2016). Defining the Minimal Factors Required for Erythropoiesis through Direct Lineage Conversion. *Cell Rep. 15*, 2550–2562.

[28] Catala, A., Culp-Hill, R., Nemkov, T., and D'Alessandro, A. (2018). Quantitative metabolomics comparison of traditional blood draws and TAP capillary blood collection. *Metabolomics 14*, 100.

[29] Challis, R.C. et al. (2019). Systemic AAV vectors for widespread and targeted gene delivery in rodents. *Nat. Protoc. 14*, 379–414.

[30] Chamberlin, N.L., Du, B., de Lacalle, S., and Saper, C.B. (1998). Recombinant adeno-associated virus vector: use for transgene expression and anterograde tract tracing in the CNS. *Brain Res. 793*, 169–175.

[31] Chan, K.Y. et al. (2017). Engineered AAVs for efficient noninvasive gene delivery to the central and peripheral nervous systems. *Nat. Neurosci. 20*, 1172–1179.

[32] Chang, H., Lim, J., Ha, M., and Kim, V.N. (2014). TAIL-seq: Genome-wide Determination of Poly(A) Tail Length and 3′ End Modifications. *Mol. Cell 53*, 1044–1052.

[33] Chasseigneaux, S. et al. (2018). Isolation and differential transcriptome of vascular smooth muscle cells and mid-capillary pericytes from the rat brain. *Sci. Rep. 8*, 12272.

[34] Chen, K., and Arnold, F.H. (1993). Tuning the activity of an enzyme for unusual environments: sequential random mutagenesis of subtilisin E for catalysis in dimethylformamide. *Proc. Natl. Acad. Sci. 90*, 5618–5622.

[35] Chen, J., Carey, K., and Godowski, P.J. (1997). Identification of Gas6 as a ligand for Mer, a neural cell adhesion molecule related receptor tyrosine kinase implicated in cellular transformation. *Oncogene 14*, 2033–2039.

[36] Chen, K.H., Boettiger, A.N., Moffitt, J.R., Wang, S., and Zhuang, X. (2015). Spatially resolved, highly multiplexed RNA profiling in single cells. *Science 348*.

[37] Choi, H.M.T., Beck, V.A., and Pierce, N.A. (2014). Next-generation in situ hybridization chain reaction: Higher gain, lower cost, greater durability. *ACS Nano 8*, 4284–4294.

[38] Chuah, M.K. et al. (2014). Liver-specific transcriptional modules identified by genome-wide in silico analysis enable efficient gene therapy in mice and non-human primates. *Mol. Ther. J. Am. Soc. Gene Ther. 22*, 1605–1613.

[39] Chui, R., and Dorovini-Zis, K. (2010). Regulation of CCL2 and CCL3 expression in human brain endothelial cells by cytokines and lipopolysaccharide. *J. Neuroinflammation 7*, 1.

[40] Colella, P., Ronzitti, G., and Mingozzi, F. (2018). Emerging issues in AAV-mediated in vivo gene therapy. *Mol. Ther. - Methods Clin. Dev. 8*, 87–104.

[41] Dai, J., Bercury, K.K., Ahrendsen, J.T., and Macklin, W.B. (2015). Olig1 Function Is Required for Oligodendrocyte Differentiation in the Mouse Brain. *J. Neurosci. 35*, 4386–4402.

[42] Dalkara, D. et al. (2013). In vivo-directed evolution of a new adeno-associated virus for therapeutic outer retinal gene delivery from the vitreous. *Sci. Transl. Med. 5*, 189ra76.

[43] Davidsson, M. et al. (2019). A systematic capsid evolution approach performed in vivo for the design of AAV vectors with tailored properties and tropism. *Proc. Natl. Acad. Sci. 116*, 27053–27062.

[44] Davis, A.S. et al. (2015). Rational design and engineering of a modified adeno-associated virus (AAV1)-based vector system for enhanced retrograde gene delivery. *Neurosurgery 76*, 216–225.

[45] Daya, S., and Berns, K.I. (2008). Gene therapy using adeno-associated virus vectors. *Clin. Microbiol. Rev. 21*, 583–593.

[46] De Alencastro, G. et al. (2020). Tracking adeno-associated virus capsid evolution by high-throughput sequencing. *Hum. Gene Ther. 31*, 553–564.

[47] De Jager, P.L. et al. (2015). ImmVar project: Insights and design considerations for future studies of "healthy" immune variation. *Semin. Immunol. 27*, 51–57.

[48] Dean, L. (2005). Chapter 1: Blood and the cells it contains. In Blood Groups and Red Cell Antigens, (National Center for Biotechnology Information), p.

[49] Deleage, C. et al. (2016). Defining HIV and SIV reservoirs in lymphoid tissues. *Pathog. Immun. 1*, 68–106.

[50] Deleage, C., Chan, C.N., Busman-Sahay, K., and Estes, J.D. (2018). Next-generation in situ hybridization approaches to define and quantify HIV and SIV reservoirs in tissue microenvironments. *Retrovirology 15*, 4.

[51] Deverman, B.E. et al. (2016). Cre-dependent selection yields AAV variants for widespread gene transfer to the adult brain. *Nat. Biotechnol. 34*, 204–209.

[52] Deverman, B.E., Ravina, B.M., Bankiewicz, K.S., Paul, S.M., and Sah, D.W.Y. (2018). Gene therapy for neurological disorders: Progress and prospects. *Nat. Rev. Drug Discov. 17*, 641–659.

[53] DiMattia, M.A. et al. (2012). Structural Insight into the Unique Properties of Adeno-Associated Virus Serotype 9. *J. Virol. 86*, 6947–6958.

[54] Duan, D. (2018). Systemic AAV micro-dystrophin gene therapy for Duchenne muscular dystrophy. *Mol. Ther. 26*, 2337–2356.
[55] Durruthy-Durruthy, R., Gottlieb, A., and Heller, S. (2015). 3D computational reconstruction of tissues with hollow spherical morphologies using single-cell gene expression data. *Nat. Protoc. 10*, 459–474.
[56] Excoffon, K.J.D.A. et al. (2009). Directed evolution of adeno-associated virus to an infectious respiratory virus. *Proc. Natl. Acad. Sci. 106*, 3865–3870.
[57] Fairfax, B.P., and Knight, J.C. (2014). Genetics of gene expression in immunity to infection. *Curr. Opin. Immunol. 30*, 63–71.
[58] Farh, K.K.-H. et al. (2015). Genetic and epigenetic fine mapping of causal autoimmune disease variants. *Nature 518*, 337–343.
[59] Femino, A.M., Fay, F.S., Fogarty, K., and Singer, R.H. (1998). Visualization of single RNA transcripts in situ. *Science 280*, 585–590.
[60] Ferreira, M.P., and Nagai, M.A. (2019). PHLDA3 (Pleckstrin Homology-Like Domain, family A, member 3). *Atlas Genet. Cytogenet. Oncol. Haematol.*
[61] Finn, J.D. et al. (2010). Proteasome inhibitors decrease AAV2 capsid derived peptide epitope presentation on MHC class I following transduction. *Mol. Ther. 18*, 135–142.
[62] Flytzanis, N.C. et al. (2020). Broad gene expression throughout the mouse and marmoset brain after intravenous delivery of engineered AAV capsids. *BioRxiv* 2020.06.16.152975.
[63] Forsyth, C.M. et al. (2013). Deep mutational scanning of an antibody against epidermal growth factor receptor using mammalian cell display and massively parallel pyrosequencing. *MAbs 5*, 523–532.
[64] Foust, K.D. et al. (2009). Intravascular AAV9 preferentially targets neonatal neurons and adult astrocytes. *Nat. Biotechnol. 27*, 59–65.
[65] Fowler, D.M., and Fields, S. (2014). Deep mutational scanning: a new style of protein science. *Nat. Methods 11*, 801–807.
[66] Franchini, A.M., Myers, J.R., Jin, G.-B., Shepherd, D.M., and Lawrence, B.P. (2019). Genome-Wide Transcriptional Analysis Reveals Novel AhR Targets That Regulate Dendritic Cell Function during Influenza A Virus Infection. *ImmunoHorizons 3*, 219–235.
[67] Gaj, T., Epstein, B.E., and Schaffer, D.V. (2016). Genome engineering using adeno-associated virus: Basic and clinical research applications. *Mol. Ther. J. Am. Soc. Gene Ther. 24*, 458–464.
[68] Gao, G. et al. (2009). Adeno-associated virus-mediated gene transfer to nonhuman primate liver can elicit destructive transgene-specific T cell responses. *Hum. Gene Ther. 20*, 930–942.
[69] Gate, D. et al. (2020). Clonally expanded CD8 T cells patrol the cerebrospinal fluid in Alzheimer's disease. *Nature 577*, 399–404.
[70] George, L.A. et al. (2017). Hemophilia B gene therapy with a high-specific-activity factor IX variant. *N. Engl. J. Med. 377*, 2215–2227.
[71] Ghouzzi, V.E. et al. (2016). ZIKA virus elicits P53 activation and genotoxic stress in human neural progenitors similar to mutations involved in severe forms of genetic microcephaly and p53. *Cell Death Dis. 7*, e2440–e2440.

[72] Gokce, O. et al. (2016). Cellular Taxonomy of the Mouse Striatum as Revealed by Single-Cell RNA-Seq. *Cell Rep. 16*, 1126–1137.
[73] Good, I.J. (1953). THE POPULATION FREQUENCIES OF SPECIES AND THE ESTIMATION OF POPULATION PARAMETERS. *Biometrika 40*, 237–264.
[74] Grabinski, T.M., Kneynsberg, A., Manfredsson, F.P., and Kanaan, N.M. (2015). A method for combining RNAscope in situ hybridization with immunohistochemistry in thick free-floating brain sections and primary neuronal cultures. *PLoS One 10*, e0120120.
[75] Gralinski, L.E., Ashley, S.L., Dixon, S.D., and Spindler, K.R. (2009). Mouse Adenovirus Type 1-Induced Breakdown of the Blood-Brain Barrier. *J. Virol. 83*, 9398–9410.
[76] Grimm, D. et al. (2008). In vitro and in vivo gene therapy vector evolution via multispecies interbreeding and retargeting of adeno-associated viruses. *J. Virol. 82*, 5887–5911.
[77] Gustafsson, C., Govindarajan, S., and Minshull, J. (2004). Codon bias and heterologous protein expression. *Trends Biotechnol. 22*, 346–353.
[78] Halpern, K.B. et al. (2017). Single-cell spatial reconstruction reveals global division of labour in the mammalian liver. *Nature 542*, 352–356.
[79] Hayashi, T. et al. (2018). Single-cell full-length total RNA sequencing uncovers dynamics of recursive splicing and enhancer RNAs. *Nat. Commun. 9*, 619.
[80] He, L. et al. (2016). Analysis of the brain mural cell transcriptome. *Sci. Rep. 6*, 35108.
[81] He, W. et al. (2018). Circadian Expression of Migratory Factors Establishes Lineage-Specific Signatures that Guide the Homing of Leukocyte Subsets to Tissues. *Immunity 49*, 1175-1190.e7.
[82] Hermida, R.C., Ayala, D.E., Chayán, L., Mojón, A., and Fernández, J.R. (2009). Administration-Time-Dependent Effects of Olmesartan on the Ambulatory Blood Pressure of Essential Hypertension Patients. *Chronobiol. Int. 26*, 61–79.
[83] Herrmann, A.-K. et al. (2019). A robust and all-inclusive pipeline for shuffling of adeno-associated viruses. *ACS Synth. Biol. 8*, 194–206.
[84] Herzog, R.W. et al. (2019). Regulatory T cells and TLR9 activation shape antibody formation to a secreted transgene product in AAV muscle gene transfer. *Cell. Immunol. 342*, 103682.
[85] Hinderer, C. et al. (2018). Severe toxicity in nonhuman primates and piglets following high-dose intravenous administration of an adeno-associated virus vector expressing human SMN. *Hum. Gene Ther. 29*, 285–298.
[86] Hirsch, M.L., and Samulski, R.J. (2014). AAV-Mediated Gene Editing via Double-Strand Break Repair. In Gene Correction: Methods and Protocols, F. Storici, ed. (Totowa, NJ: Humana Press), pp. 291–307.
[87] Hordeaux, J. et al. (2018). The Neurotropic Properties of AAV-PHP.B Are Limited to C57BL/6J Mice. *Mol. Ther. 26*, 664–668.
[88] Hordeaux, J. et al. (2019). The GPI-Linked Protein LY6A Drives AAV-PHP.B Transport across the Blood-Brain Barrier. *Mol. Ther. 27*, 912–921.
[89] Hösel, M. et al. (2012). Toll-like receptor 2–mediated innate immune response in human nonparenchymal liver cells toward adeno-associated viral vectors. *Hepatology 55*, 287–297.
[90] Hrvatin, S. et al. (2019). A scalable platform for the development of cell-type-specific viral drivers. *ELife 8*, e48089.

[91] Huang, K.W., and Sabatini, B.L. (2020). Single-cell analysis of neuroinflammatory responses following intracranial injection of G-deleted rabies viruses. *Front. Cell. Neurosci. 14*, 65.

[92] Huang, Q. et al. (2019). Delivering genes across the blood-brain barrier: LY6A, a novel cellular receptor for AAV-PHP.B capsids. *PLOS ONE 14*, e0225206.

[93] Hunter, J.E., Gurda, B.L., Yoon, S.Y., Castle, M.J., and Wolfe, J.H. (2019). In situ hybridization for detection of AAV-mediated gene expression. *Methods Mol. Biol. Clifton NJ 1950*, 107–122.

[94] Hwang, B., Lee, J.H., and Bang, D. (2018). Single-cell RNA sequencing technologies and bioinformatics pipelines. *Exp. Mol. Med. 50*, 96.

[95] Johansen, J., Tornøe, J., Møller, A., and Johansen, T.E. (2003). Increased *in vitro* and *in vivo* transgene expression levels mediated through *cis* -acting elements: *Cis* Elements Increased *Ex Vivo* Gene Expression. *J. Gene Med. 5*, 1080–1089.

[96] Jordão, M.J.C. et al. (2019). Single-cell profiling identifies myeloid cell subsets with distinct fates during neuroinflammation. *Science 363*, eaat7554.

[97] Jüttner, J. et al. (2019). Targeting neuronal and glial cell types with synthetic promoter AAVs in mice, non-human primates and humans. *Nat. Neurosci. 22*, 1345–1356.

[98] Kanehisa, M. (2000). KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res. 28*, 27–30.

[99] Kanehisa, M. (2019). Toward understanding the origin and evolution of cellular organisms. *Protein Sci. 28*, 1947–1951.

[100] Kanehisa, M., Sato, Y., Furumichi, M., Morishima, K., and Tanabe, M. (2019). New approach for understanding genome variations in KEGG. *Nucleic Acids Res. 47*, D590–D595.

[101] Kang, H.M. et al. (2018). Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. *Nat. Biotechnol. 36*, 89–94.

[102] Kazer, S.W. et al. (2020). Integrated single-cell analysis of multicellular immune dynamics during hyperacute HIV-1 infection. *Nat. Med. 26*, 511–518.

[103] Ke, R. et al. (2013). In situ sequencing for RNA analysis in preserved tissue and cells. *Nat. Methods 10*, 857–860.

[104] Kebschull, J.M., and Zador, A.M. (2015). Sources of PCR-induced distortions in high-throughput sequencing data sets. *Nucleic Acids Res.* gkv717.

[105] Keller, M. et al. (2009). A circadian clock in macrophages controls inflammatory immune responses. *Proc. Natl. Acad. Sci. 106*, 21407–21412.

[106] Klein, A.M. et al. (2015). Droplet Barcoding for Single-Cell Transcriptomics Applied to Embryonic Stem Cells. *Cell 161*, 1187–1201.

[107] Kobayashi, M., Wood, P.A., and Hrushesky, W.J.M. (2002). Circadian chemotherapy for gynecological and genitourinary cancers. *Chronobiol. Int. 19*, 237–251.

[108] Kodali, M.C., Chen, H., and Liao, F.-F. (2020). Temporal unsnarling of brain's acute neuroinflammatory transcriptional profiles reveals panendothelitis as the earliest event preceding microgliosis. *Mol. Psychiatry*.

[109] Körbelin, J. et al. (2016b). Pulmonary targeting of adeno-associated viral vectors by next-generation sequencing-guided screening of random capsid displayed peptide libraries. *Mol. Ther. 24*, 1050–1061.

[110] Körbelin, J. et al. (2016a). A brain microvasculature endothelial cell-specific viral vector with the potential to treat neurovascular and neurological diseases. *EMBO Mol. Med. 8*, 609–625.
[111] Korsunsky, I. et al. (2019). Fast, sensitive and accurate integration of single-cell data with Harmony. *Nat. Methods 16*, 1289–1296.
[112] Kotterman, M.A., and Schaffer, D.V. (2014). Engineering adeno-associated viruses for clinical gene therapy. *Nat. Rev. Genet. 15*, 445–451.
[113] Kusanagi, H. et al. (2008). Expression profiles of 10 circadian clock genes in human peripheral blood mononuclear cells. *Neurosci. Res. 61*, 136–142.
[114] Kuzmin, D.A. et al. (2021). The clinical landscape for AAV gene therapies. *Nat. Rev. Drug Discov. 20*, 173–174.
[115] Lähnemann, D. et al. (2020). Eleven grand challenges in single-cell data science. *Genome Biol. 21*, 31.
[116] Lappalainen, T. et al. (2013). Transcriptome and genome sequencing uncovers functional variation in humans. *Nature 501*, 506–511.
[117] Lareau, C.A. et al. (2019). Droplet-based combinatorial indexing for massive-scale single-cell chromatin accessibility. *Nat. Biotechnol. 37*, 916–924.
[118] Lech, K. et al. (2016). Dissecting Daily and Circadian Expression Rhythms of Clock-Controlled Genes in Human Blood. *J. Biol. Rhythms 31*, 68–81.
[119] Lee, E.J., Guenther, C.M., and Suh, J. (2018). Adeno-associated virus (AAV) vectors: Rational design strategies for capsid engineering. *Curr. Opin. Biomed. Eng. 7*, 58–63.
[120] Lee, J.H. et al. (2014). Highly multiplexed subcellular RNA sequencing in situ. *Science 343*, 1360–1363.
[121] Lee, J.S. et al. (2020). Immunophenotyping of COVID-19 and influenza highlights the role of type I interferons in development of severe COVID-19. *Sci. Immunol. 5*, eabd1554.
[122] Lehrman, S. (1999). Virus treatment questioned after gene therapy death. *Nature 401*, 517–518.
[123] Lenicek Krleza, J., Dorotic, A., Grzunov, A., and Maradin, M. (2015). Capillary blood sampling: national recommendations on behalf of the Croatian Society of Medical Biochemistry and Laboratory Medicine. *Biochem. Medica* 335–358.
[124] Lévi, F. et al. (2007). Implications of circadian clocks for the rhythmic delivery of cancer therapeutics. *Adv. Drug Deliv. Rev. 59*, 1015–1035.
[125] Li, C., and Samulski, R.J. (2020). Engineering adeno-associated virus vectors for gene therapy. *Nat. Rev. Genet. 21*, 255–272.
[126] Li, G. et al. (2019). Joint profiling of DNA methylation and chromatin architecture in single cells. *Nat. Methods 16*, 991–993.
[127] Li, J., Lv, H., and Che, Y. (2020). microRNA-381-3p Confers Protection Against Ischemic Stroke Through Promoting Angiogenesis and Inhibiting Inflammation by Suppressing Cebpb and Map3k8. *Cell. Mol. Neurobiol. 40*, 1307–1319.
[128] Liao, J., Lu, X., Shao, X., Zhu, L., and Fan, X. (2020). Uncovering an organ's molecular architecture at single-cell resolution by spatially resolved transcriptomics. *Trends Biotechnol. 39*, 43–58.

[129] Lin, Y. et al. (2019). scMerge leverages factor analysis, stable expression, and pseudoreplication to merge multiple single-cell RNA-seq datasets. *Proc. Natl. Acad. Sci. 116*, 9775–9784.
[130] Lin, Y.-S. et al. (2016). Neuronal Splicing Regulator RBFOX3 (NeuN) Regulates Adult Hippocampal Neurogenesis and Synaptogenesis. *PLOS ONE 11*, e0164164.
[131] Lindstone, G.J. (1920). Note on the General Case of the Bayes-Laplace Formula for Inductive or a Posteriori Probabilities. *Trans. Oo Fac. Actuar.*
[132] Liu, D. (2019). Algorithms for efficiently collapsing reads with Unique Molecular Identifiers. *PeerJ 7*, e8275.
[133] Liu, Q., Yang, Y., and Fan, X. (2020). Microvascular pericytes in brain-associated vascular disease. *Biomed. Pharmacother. 121*, 109633.
[134] Long, J.E. et al. (2016). Morning vaccination enhances antibody response over afternoon vaccination: A cluster-randomised trial. *Vaccine 34*, 2679–2685.
[135] Lopez, R., Regier, J., Cole, M.B., Jordan, M.I., and Yosef, N. (2018). Deep generative modeling for single-cell transcriptomics. *Nat. Methods 15*, 1053–1058.
[136] Lorenzon, E. et al. (2012). MULTIMERIN2 impairs tumor angiogenesis and growth by interfering with VEGF-A/VEGFR2 pathway. *Oncogene 31*, 3136–3147.
[137] Love, M.I., Huber, W., and Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biol. 15*, 550.
[138] Lowenstein, P., Mandel, R., Xiong, W., Kroeger, K., and Castro, M. (2007). Immune Responses to Adenovirus and Adeno-Associated Vectors Used for Gene Therapy of Brain Diseases: The Role of Immunological Synapses in Understanding the Cell Biology of Neuroimmune Interactions. *Curr. Gene Ther. 7*, 347–360.
[139] Lubeck, E., Coskun, A.F., Zhiyentayev, T., Ahmad, M., and Cai, L. (2014). Single-cell in situ RNA profiling by sequential hybridization. *Nat. Methods 11*, 360–361.
[140] Lun, A.T.L. et al. (2019). EmptyDrops: Distinguishing cells from empty droplets in droplet-based single-cell RNA sequencing data. *Genome Biol. 20*, 63.
[141] Lykken, E.A., Shyng, C., Edwards, R.J., Rozenberg, A., and Gray, S.J. (2018). Recent progress and considerations for AAV gene therapies targeting the central nervous system. *J. Neurodev. Disord. 10*, 16.
[142] Macosko, E.Z. et al. (2015). Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. *Cell 161*, 1202–1214.
[143] Macrae, M. et al. (2005). A conditional feedback loop regulates Ras activity through EphA2. *Cancer Cell 8*, 111–118.
[144] Maheshri, N., Koerber, J.T., Kaspar, B.K., and Schaffer, D.V. (2006). Directed evolution of adeno-associated virus yields enhanced gene delivery vectors. *Nat. Biotechnol. 24*, 198–204.
[145] Manno, C.S. et al. (2006). Successful transduction of liver in hemophilia by AAV-Factor IX and limitations imposed by the host immune response. *Nat. Med. 12*, 342–347.
[146] Marques, S. et al. (2016). Oligodendrocyte heterogeneity in the mouse juvenile and adult central nervous system. *Science 352*, 1326–1329.

[147] Martin, J.C. et al. (2019). Single-Cell Analysis of Crohn's Disease Lesions Identifies a Pathogenic Cellular Module Associated with Resistance to Anti-TNF Therapy. *Cell 178*, 1493-1508.e20.

[148] Martino, A.T. et al. (2011). The genome of self-complementary adeno-associated viral vectors increases Toll-like receptor 9–dependent innate immune responses in the liver. *Blood 117*, 6459–6468.

[149] Matsa, E. et al. (2016). Transcriptome Profiling of Patient-Specific Human iPSC-Cardiomyocytes Predicts Individual Drug Safety and Efficacy Responses In Vitro. *Cell Stem Cell 19*, 311–325.

[150] Matsuzaki, Y. et al. (2019). Neurotropic Properties of AAV-PHP.B Are Shared among Diverse Inbred Strains of Mice. *Mol. Ther. 27*, 700–704.

[151] McCown, T.J., Xiao, X., Li, J., Breese, G.R., and Jude Samulski, R. (1996). Differential and persistent expression patterns of CNS gene transfer by an adeno-associated virus (AAV) vector. *Brain Res. 713*, 99–107.

[152] McGinnis, C.S. et al. (2019). MULTI-seq: Sample multiplexing for single-cell RNA sequencing using lipid-tagged indices. *Nat. Methods 16*, 619–626.

[153] Mével, M. et al. (2020). Chemical modification of the adeno-associated virus capsid to improve gene delivery. *Chem. Sci. 11*, 1122–1131.

[154] Miao, C.H. et al. (2000). Nonrandom transduction of recombinant adeno-associated virus vectors in mouse hepatocytes in vivo: Cell cycling does not influence hepatocyte transduction. *J. Virol. 74*, 3793–3803.

[155] Mich, J.K. et al. (2020). Functional enhancer elements drive subclass-selective expression from mouse to primate neocortex. *BioRxiv* 555318.

[156] Mingozzi, F., and High, K.A. (2013). Immune responses to AAV vectors: Overcoming barriers to successful gene therapy. *Blood 122*, 23–36.

[157] Mingozzi, F. et al. (2007). CD8 + T-cell responses to adeno-associated virus capsid in humans. *Nat. Med. 13*, 419–422.

[158] Mingozzi, F. et al. (2009). AAV-1–mediated gene transfer to skeletal muscle in humans results in dose-dependent activation of capsid-specific T cells. *Blood 114*, 2077–2086.

[159] Miron, V.E., Kuhlmann, T., and Antel, J.P. (2011). Cells of the oligodendroglial lineage, myelination, and remyelination. *Biochim. Biophys. Acta BBA - Mol. Basis Dis. 1812*, 184–193.

[160] Montagne, A. et al. (2020). APOE4 leads to blood–brain barrier dysfunction predicting cognitive decline. *Nature 581*, 71–76.

[161] Mossner, J.M., Batista-Brito, R., Pant, R., and Cardin, J.A. (2020). Developmental loss of MeCP2 from VIP interneurons impairs cortical function and behavior. *ELife 9*, e55639.

[162] Muhammad, K.A., Nur, A.A., Nurul, H.S., Narazah, M.Y., and Siti, R.A.R. (2018). Dual-specificity phosphatase 6 (DUSP6): a review of its molecular characteristics and clinical relevance in cancer. *Cancer Biol. Med. 15*, 14.

[163] Müller, O.J. et al. (2003). Random peptide libraries displayed on adeno-associated virus to select for targeted gene therapy vectors. *Nat. Biotechnol. 21*, 1040–1046.

[164] Naso, M.F., Tomkowicz, B., Perry, W.L., and Strohl, W.R. (2017). Adeno-associated virus (AAV) as a vector for gene therapy. *Biodrugs 31*, 317–334.
[165] Nathwani, A.C. et al. (2011). Adenovirus-associated virus vector–mediated gene transfer in hemophilia B. *N. Engl. J. Med. 365*, 2357–2365.
[166] Nathwani, A.C. et al. (2014). Long-term safety and efficacy of factor IX gene therapy in hemophilia B. *N. Engl. J. Med. 371*, 1994–2004.
[167] Nitzan, M., Karaiskos, N., Friedman, N., and Rajewsky, N. (2019). Gene expression cartography. *Nature 576*, 132–137.
[168] Ogden, P.J., Kelsic, E.D., Sinai, S., and Church, G.M. (2019). Comprehensive AAV capsid fitness landscape reveals a viral gene and enables machine-guided design. *Science 366*, 1139–1143.
[169] Ohtsuka, M., Inoko, H., Kulski, J.K., and Yoshimura, S. (2008). Major histocompatibility complex (Mhc) class Ib gene duplications, organization and expression patterns in mouse strain C57BL/6. *BMC Genomics 9*, 178.
[170] Oikonomou, G. et al. (2019). The serotonergic raphe promote sleep in zebrafish and mice. *Neuron 103*, 686-701.e8.
[171] Ojala, D.S. et al. (2018a). In Vivo Selection of a Computationally Designed SCHEMA AAV Library Yields a Novel Variant for Infection of Adult Neural Stem Cells in the SVZ. *Mol. Ther. J. Am. Soc. Gene Ther. 26*, 304–319.
[172] Ojala, D.S. et al. (2018b). In vivo selection of a computationally designed SCHEMA AAV library yields a novel variant for infection of adult neural stem cells in the SVZ. *Mol. Ther. 26*, 304–319.
[173] Ollila, H.M. et al. (2012). TRIB1 constitutes a molecular link between regulation of sleep and lipid metabolism in humans. *Transl. Psychiatry 2*, e97–e97.
[174] Palfreyman, M.T., and Jorgensen, E.M. (2017). Unc13 Aligns SNAREs and Superprimes Synaptic Vesicles. *Neuron 95*, 473–475.
[175] Papanikolaou, E., and Bosio, A. (2021). The Promise and the Hope of Gene Therapy. *Front. Genome Ed. 3*, 618346.
[176] Patriarchi, T. et al. (2018). Ultrafast neuronal imaging of dopamine dynamics with designed genetically encoded sensors. *Science 360*, eaat4422.
[177] Paulk, N. (2020). Gene Therapy: It Is Time to Talk about High-Dose AAV: The deaths of two children with X-linked myotubular myopathy in the ASPIRO trial prompts a reexamination of vector safety. *Genet. Eng. Biotechnol. News 40*, 14–16.
[178] Paulk, N.K. et al. (2018). Bioengineered AAV capsids with combined high human liver transduction in vivo and unique humoral seroreactivity. *Mol. Ther. 26*, 289–303.
[179] Pekrun, K. et al. (2019). Using a barcoded AAV capsid library to select for clinically relevant gene therapy vectors. *JCI Insight 4*, e131610.
[180] Perez-Nievas, B.G., and Serrano-Pozo, A. (2018). Deciphering the Astrocyte Reaction in Alzheimer's Disease. *Front. Aging Neurosci. 10*, 114.
[181] Pérez-Sen et al. (2019). Dual-Specificity Phosphatase Regulation in Neurons and Glial Cells. *Int. J. Mol. Sci. 20*, 1999.

[182] Pick, R., He, W., Chen, C.-S., and Scheiermann, C. (2019). Time-of-Day-Dependent Trafficking and Function of Leukocyte Subsets. *Trends Immunol. 40*, 524–537.
[183] Pien, G.C. et al. (2009). Capsid antigen presentation flags human hepatocytes for destruction after transduction by adeno-associated viral vectors. *J. Clin. Invest. 119*, 1688–1695.
[184] Polinski, N.K. et al. (2015). Recombinant adenoassociated virus 2/5-mediated gene transfer is reduced in the aged rat midbrain. *Neurobiol. Aging 36*, 1110–1120.
[185] Polinski, N.K. et al. (2016). Impact of age and vector construct on striatal and nigral transgene expression. *Mol. Ther. Methods Clin. Dev. 3*, 16082.
[186] Pool, A.-H. et al. (2020). The cellular basis of distinct thirst modalities. *Nature 588*, 112–117.
[187] Pulford, Jones, Banham, Haralambieva, and Mason (1999). Lymphocyte-specific protein 1: a specific marker of human leucocytes. *Immunology 96*, 262–271.
[188] Pulicherla, N. et al. (2011). Engineering liver-detargeted AAV9 vectors for cardiac and musculoskeletal gene transfer. *Mol. Ther. 19*, 1070–1078.
[189] Puray-Chavez, M. et al. (2017). Multiplex single-cell visualization of nucleic acids and protein during HIV infection. *Nat. Commun. 8*, 1882.
[190] Qin, J.Y. et al. (2010). Systematic Comparison of Constitutive Promoters and the Doxycycline-Inducible Promoter. *PLoS ONE 5*, e10611.
[191] Ramsey, M.R., and Ellisen, L.W. (2011). Circadian function in cancer: Regulating the DNA damage response. *Proc. Natl. Acad. Sci. 108*, 10379–10380.
[192] Ravindra Kumar, S. et al. (2020). Multiplexed Cre-dependent selection yields systemic AAVs for targeting distinct brain cell types. *Nat. Methods 17*, 541–550.
[193] Riemondy, K.A. et al. (2019). Recovery and analysis of transcriptome subsets from pooled single-cell RNA-seq libraries. *Nucleic Acids Res. 47*, e20–e20.
[194] Rincon, M.Y. et al. (2015). Genome-wide computational analysis reveals cardiomyocyte-specific transcriptional Cis-regulatory motifs that enable efficient cardiac gene therapy. *Mol. Ther. J. Am. Soc. Gene Ther. 23*, 43–52.
[195] Riverso, M., Montagnani, V., and Stecca, B. (2017). KLF4 is regulated by RAS/RAF/MEK/ERK signaling through E2F1 and promotes melanoma cell growth. *Oncogene 36*, 3322–3333.
[196] Robison, E.H. et al. (2009). Whole genome transcript profiling from fingerstick blood samples: a comparison and feasibility study. *BMC Genomics 10*, 617.
[197] Rodriguez-Meira, A. et al. (2019). Unravelling Intratumoral Heterogeneity through High-Sensitivity Single-Cell Mutational Analysis and Parallel RNA Sequencing. *Mol. Cell 73*, 1292-1305.e8.
[198] Rodriguez-Meira, A., O'Sullivan, J., Rahman, H., and Mead, A.J. (2020). TARGET-Seq: A Protocol for High-Sensitivity Single-Cell Mutational Analysis and Parallel RNA Sequencing. *STAR Protoc. 1*, 100125.
[199] Rogers, G.L. et al. (2017). Plasmacytoid and conventional dendritic cells cooperate in crosspriming AAV capsid-specific CD8+ T cells. *Blood 129*, 3184–3195.
[200] Rosenberg, A.B. et al. (2018). Single-cell profiling of the developing mouse brain and spinal cord with split-pool barcoding. *Science 360*, 176–182.

[201] Rossi, A. et al. (2019). Vector uncoating limits adeno-associated viral vector-mediated transduction of human dendritic cells and vector immunogenicity. *Sci. Rep. 9*, 3631.
[202] Rubin, A.F. et al. (2017). A statistical framework for analyzing deep mutational scanning data. *Genome Biol. 18*, 150.
[203] Ruden, J.B., Dugan, L.L., and Konradi, C. (2021). Parvalbumin interneuron vulnerability and brain disorders. *Neuropsychopharmacology 46*, 279–287.
[204] Saikia, M. et al. (2019). Simultaneous multiplexed amplicon sequencing and transcriptome profiling in single cells. *Nat. Methods 16*, 59–62.
[205] Salvador, J.M., Brown-Clay, J.D., and Fornace, A.J. (2013). Gadd45 in Stress Signaling, Cell Cycle Control, and Apoptosis. In Gadd45 Stress Sensor Genes, D.A. Liebermann, and B. Hoffman, eds. (New York, NY: Springer New York), pp. 1–19.
[206] Samulski, R.J., and Muzyczka, N. (2014). AAV-mediated gene therapy for research and therapeutic purposes. *Annu. Rev. Virol. 1*, 427–451.
[207] Sarkar, S. et al. (2020). Molecular Signatures of Neuroinflammation Induced by αSynuclein Aggregates in Microglial Cells. *Front. Immunol. 11*, 33.
[208] Satija, R., Farrell, J.A., Gennert, D., Schier, A.F., and Regev, A. (2015). Spatial reconstruction of single-cell gene expression data. *Nat. Biotechnol. 33*, 495–502.
[209] Schindelin, J. et al. (2012). Fiji: An open-source platform for biological-image analysis. *Nat. Methods 9*, 676–682.
[210] Sen, D. (2014). Improving clinical efficacy of adeno associated vectors by rational capsid bioengineering. *J. Biomed. Sci. 21*.
[211] Servick, K. (2021). Gene therapy clinical trial halted as cancer risk surfaces. *Science*.
[212] Shah, S., Lubeck, E., Zhou, W., and Cai, L. (2016a). In situ transcription profiling of single cells reveals spatial organization of cells in the mouse hippocampus. *Neuron 92*, 342–357.
[213] Shah, S. et al. (2016b). Single-molecule RNA detection at depth by hybridization chain reaction and tissue hydrogel embedding and clearing. *Development 143*, 2862–2867.
[214] Shao, W. et al. (2018). Double-stranded RNA innate immune response activation from long-term adeno-associated virus vector transduction. *JCI Insight 3*.
[215] Shin, S., and Park, J. (2016). Characterization of sequence-specific errors in various next-generation sequencing systems. *Mol. Biosyst. 12*, 914–922.
[216] Shirley, J.L., Jong, Y.P. de, Terhorst, C., and Herzog, R.W. (2020). Immune Responses to Viral Gene Therapy Vectors. *Mol. Ther. 28*, 709–722.
[217] Singh, M. et al. (2019). High-throughput targeted long-read single cell sequencing reveals the clonal and transcriptional landscape of lymphocytes. *Nat. Commun. 10*, 3120.
[218] Smith, G. (1985). Filamentous fusion phage: novel expression vectors that display cloned antigens on the virion surface. *Science 228*, 1315–1317.
[219] Somanathan, S., Breous, E., Bell, P., and Wilson, J.M. (2010). AAV vectors avoid inflammatory signals necessary to render transduced hepatocyte targets for destructive T cells. *Mol. Ther. 18*, 977–982.
[220] Song, H.W. et al. (2020). Transcriptomic comparison of human and mouse brain microvessels. *Sci. Rep. 10*, 12358.

[221] Spiegel, I. et al. (2014). Npas4 Regulates Excitatory-Inhibitory Balance within Neural Circuits through Cell-Type-Specific Gene Programs. *Cell 157*, 1216–1229.
[222] Srivastava, A. (2020). AAV vectors: Are they safe? *Hum. Gene Ther. 31*, 697–699.
[223] Starita, L.M., and Fields, S. (2015). Deep Mutational Scanning: A Highly Parallel Method to Measure the Effects of Mutation on Protein Function: Figure 1. *Cold Spring Harb. Protoc. 2015*, pdb.top077503.
[224] Stoeckius, M. et al. (2018). Cell Hashing with barcoded antibodies enables multiplexing and doublet detection for single cell genomics. *Genome Biol. 19*, 224.
[225] Stuart, T. et al. (2019). Comprehensive Integration of Single-Cell Data. *Cell 177*, 1888-1902.e21.
[226] Sumitomo, S. et al. (2018). Transcriptome analysis of peripheral blood from patients with rheumatoid arthritis: a systematic review. *Inflamm. Regen. 38*, 21.
[227] Sun, W. et al. (2017). SOX9 Is an Astrocyte-Specific Nuclear Marker in the Adult Brain Outside the Neurogenic Regions. *J. Neurosci. 37*, 4493–4507.
[228] Suzuki, N. et al. (2017). Differentiation of Oligodendrocyte Precursor Cells from Sox10-Venus Mice to Oligodendrocytes and Astrocytes. *Sci. Rep. 7*, 14133.
[229] Svensson, V., Vento-Tormo, R., and Teichmann, S.A. (2018). Exponential scaling of single-cell RNA-seq in the past decade. *Nat. Protoc. 13*, 599–604.
[230] Svensson, V., da Veiga Beltrame, E., and Pachter, L. (2020). A curated database reveals trends in single-cell transcriptomics. *Database 2020*, baaa073.
[231] Sweeney, M.D., Sagare, A.P., and Zlokovic, B.V. (2018). Blood-brain barrier breakdown in Alzheimer disease and other neurodegenerative disorders. *Nat. Rev. Neurol. 14*, 133–150.
[232] Sweeney, M.D., Zhao, Z., Montagne, A., Nelson, A.R., and Zlokovic, B.V. (2019). Blood-Brain Barrier: From Physiology to Disease and Back. *Physiol. Rev. 99*, 21–78.
[233] Tan, Y., and Cahan, P. (2019). SingleCellNet: A Computational Tool to Classify Single Cell RNA-Seq Data Across Platforms and Across Species. *Cell Syst. 9*, 207-213.e2.
[234] Tang, R. et al. (2017). Capillary blood for point-of-care testing. *Crit. Rev. Clin. Lab. Sci. 54*, 294–308.
[235] Taoufiq, Z. et al. (2020). Hidden proteome of synaptic vesicles in the mammalian brain. *Proc. Natl. Acad. Sci. 117*, 33586–33596.
[236] Tasic, B. et al. (2016). Adult mouse cortical cell taxonomy revealed by single cell transcriptomics. *Nat. Neurosci. 19*, 335–346.
[237] Tasic, B. et al. (2018). Shared and distinct transcriptomic cell types across neocortical areas. *Nature 563*, 72–78.
[238] Tervo, D.G. et al. (2016). A designer AAV variant permits efficient retrograde access to projection neurons. *Neuron 92*, 372–382.
[239] the Accelerating Medicines Partnership Rheumatoid Arthritis and Systemic Lupus Erythematosus (AMP RA/SLE) Consortium et al. (2019). Tubular cell and keratinocyte single-cell transcriptomics applied to lupus nephritis reveal type I IFN and fibrosis relevant pathways. *Nat. Immunol. 20*, 915–927.
[240] The Immune Cell Census The Immune Cell Census.

[241] Thomas, D. (2010). Gene–environment-wide association studies: emerging approaches. *Nat. Rev. Genet. 11*, 259–272.
[242] Toma, R. et al. (2020). A clinically validated human capillary blood transcriptome test for global systems biology studies. *BioTechniques 69*, 289–301.
[243] Tordo, J. et al. (2018). A novel adeno-associated virus capsid with enhanced neurotropism corrects a lysosomal transmembrane enzyme deficiency. *Brain 141*, 2014–2031.
[244] Uniken Venema, W.T. et al. (2019). Single-Cell RNA Sequencing of Blood and Ileal T Cells From Patients With Crohn's Disease Reveals Tissue-Specific Characteristics and Drug Targets. *Gastroenterology 156*, 812-815.e22.
[245] Vairapandi, M., Balliet, A.G., Hoffman, B., and Liebermann, D.A. (2002). GADD45b and GADD45g are cdc2/cyclinB1 kinase inhibitors with a role in S and G2/M cell cycle checkpoints induced by genotoxic stress. *J. Cell. Physiol. 192*, 327–338.
[246] Vandenberghe, L.H. et al. (2006). Heparin binding directs activation of T cells against adeno-associated virus serotype 2 capsid. *Nat. Med. 12*, 967–971.
[247] Vanlandewijck, M. et al. (2018). A molecular atlas of cell types and zonation in the brain vasculature. *Nature 554*, 475–480.
[248] Veys, K. et al. (2020). Role of the GLUT1 Glucose Transporter in Postnatal CNS Angiogenesis and Blood-Brain Barrier Integrity. *Circ. Res. 127*, 466–482.
[249] Vlasov, K., Van Dort, C.J., and Solt, K. (2018). Chapter Eleven - Optogenetics and Chemogenetics. In Methods in Enzymology, R.G. Eckenhoff, and I.J. Dmochowski, eds. (Academic Press), pp. 181–196.
[250] Wang, S.K., Lapan, S.W., Hong, C.M., Krause, T.B., and Cepko, C.L. (2020). In situ detection of adeno-associated viral vector genomes with SABER-FISH. *Mol. Ther. - Methods Clin. Dev. 19*, 376–386.
[251] Wang, X. et al. (2018). Three-dimensional intact-tissue sequencing of single-cell transcriptional states. *Science 361*, eaat5691.
[252] Whitney, A.R. et al. (2003). Individuality and variation in gene expression patterns in human blood. *Proc. Natl. Acad. Sci. 100*, 1896–1901.
[253] Wilson, J.M., and Flotte, T.R. (2020). Moving forward after two deaths in a gene therapy trial of myotubular myopathy. *Hum. Gene Ther. 31*, 695–696.
[254] Winkler, E.A., Bell, R.D., and Zlokovic, B.V. (2010). Pericyte-specific expression of PDGF beta receptor in mouse models with normal and deficient PDGF beta receptor signaling. *Mol. Neurodegener. 5*, 32.
[255] Wolf, F.A., Angerer, P., and Theis, F.J. (2018). SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol. 19*, 15.
[256] Wolock, S.L., Lopez, R., and Klein, A.M. (2019). Scrublet: Computational identification of cell doublets in single-cell transcriptomic data. *Cell Syst. 8*, 281-291.e9.
[257] Wrenbeck, E.E., Faber, M.S., and Whitehead, T.A. (2017). Deep sequencing methods for protein engineering and design. *Curr. Opin. Struct. Biol. 45*, 36–44.
[258] Wu, Y.E., Pan, L., Zuo, Y., Li, X., and Hong, W. (2017). Detecting Activated Cell Populations Using Single-Cell RNA-Seq. *Neuron 96*, 313-329.e6.

[259] Wu, Z., Asokan, A., and Samulski, R.J. (2006). Adeno-associated virus serotypes: Vector toolkit for human gene therapy. *Mol. Ther. 14*, 316–327.
[260] Xu, C. et al. (2021). Probabilistic harmonization and annotation of single-cell transcriptomics data with deep generative models. *Mol. Syst. Biol. 17*.
[261] Yang, A.C. et al. (2021). A human brain vascular atlas reveals diverse cell mediators of Alzheimer's disease risk (Neuroscience).
[262] Yang, S. et al. (2020). Decontamination of ambient RNA in single-cell RNA-seq with DecontX. *Genome Biol. 21*, 57.
[263] Yang, Y. et al. (2011). Molecular comparison of GLT1+ and ALDH1L1+ astrocytes in vivo in astroglial reporter mice. *Glia 59*, 200–207.
[264] Yao, Z. et al. (2021). A taxonomy of transcriptomic cell types across the isocortex and hippocampal formation. *Cell 184*, 3222-3241.e26.
[265] Ye, C.J. et al. (2014). Intersection of population variation and autoimmunity genetics in human T cell activation. *Science 345*, 1254665–1254665.
[266] Ying, Y. et al. (2010). Heart-targeted adeno-associated viral vectors selected by in vivo biopanning of a random viral display peptide library. *Gene Ther. 17*, 980–990.
[267] Zaiss, A.K. et al. (2008). Complement is an essential component of the immune response to adeno-associated virus vectors. *J. Virol. 82*, 2727–2740.
[268] Zamagni, A. et al. (2020). CDKN1A upregulation and cisplatin-pemetrexed resistance in non-small cell lung cancer cells. *Int. J. Oncol.*
[269] Zeisel, A. et al. (2018). Molecular architecture of the mouse nervous system. *Cell 174*, 999-1014.e22.
[270] Zhang, Z., and Yu, J. (2018). NR4A1 Promotes Cerebral Ischemia Reperfusion Injury by Repressing Mfn2-Mediated Mitophagy and Inactivating the MAPK–ERK–CREB Signaling Pathway. *Neurochem. Res. 43*, 1963–1977.
[271] Zhang, X. et al. (2016). In situ analysis of intrahepatic virological events in chronic hepatitis B virus infection. *J. Clin. Invest. 126*, 1079–1092.
[272] Zhang, X. et al. (2021). Identification of key genes and evaluation of immune cell infiltration in vitiligo. *Math. Biosci. Eng. 18*, 1051–1062.
[273] Zhang, Z. et al. (2019). The Appropriate Marker for Astrocytes: Comparing the Distribution and Expression of Three Astrocytic Markers in Different Mouse Cerebral Regions. *BioMed Res. Int. 2019*, 1–15.
[274] Zhao, J. et al. (2020). High-resolution histological landscape of AAV DNA distribution in cellular compartments and tissues following local and systemic injection. *Mol. Ther. - Methods Clin. Dev. 18*, 856–868.
[275] Zhao, Y. et al. (2017). Uncovering the mystery of opposite circadian rhythms between mouse and human leukocytes in humanized mice. *Blood 130*, 1995–2005.
[276] Zheng, G.X.Y. et al. (2017). Massively parallel digital transcriptional profiling of single cells. *Nat. Commun. 8*, 14049.
[277] Zhu, J., Huang, X., and Yang, Y. (2009). The TLR9-MyD88 pathway is critical for adaptive immune responses to adenoassociated virus gene therapy vectors in mice. *J. Clin. Invest. 119*, 2388–2398.

[278] Zincarelli, C., Soltys, S., Rengo, G., and Rabinowitz, J.E. (2008). Analysis of AAV Serotypes 1–9 Mediated Gene Expression and Tropism in Mice After Systemic Injection. *Mol. Ther. 16*, 1073–1080.