Learning to Design Protein and DNA Libraries

Akosua Busia

May 8, 2023

Learning to Design Protein and DNA Libraries

by

Akosua Busia

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering - Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Michael Jordan, Co-chair
Professor Jennifer Listgarten, Co-chair
Professor David Schaffer

Spring 2023

Learning to Design Protein and DNA Libraries

Abstract

Learning to Design Protein and DNA Libraries

by

Akosua Busia

Doctor of Philosophy in Engineering - Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor Michael Jordan, Co-chair

Professor Jennifer Listgarten, Co-chair

Using next-generation sequencing, it is now possible to screen up to billions of protein or DNA sequences in parallel for a property of interest. Consequently, high-throughput sequencing has vastly accelerated the rate of biological discovery for both basic scientific inquiry and for engineering novel enzymes, therapeutics, antibodies, regulatory elements, and beyond. In such high-throughput sequencing-based screens and selections, the quality of the starting sequence library greatly influences the overall chance of successfully identifying sequences with the desired property. Generalizable *in silico* methods for designing high-quality sequence libraries promise to reduce wetlab experimental burden and improve the speed with which new, functional sequences can be discovered. Machine learning, in particular, provides a useful set of tools for implementing such methods, as it is well-suited to analyzing the large quantities of data produced by high-throughput sequencing. In this dissertation, we will discuss several aspects of machine learning-guided library design, and propose solutions to challenges posed by existing technologies.

First, we introduce a framework for machine learning-guided library design, and showcase its ability to design diverse, functional libraries in a gene therapy context. Specifically, we (i) outline a modeling approach for predicting the property selected for in a high-throughput sequencing-based selection experiment that explicitly accounts for uncertainty in the observed sequencing data, and (ii) describe a novel machine learning-guided design procedure that optimally trades off between a library's average predicted property values and its sequence diversity. We use these methods to design a clinically-relevant adeno-associated virus (AAV) peptide insertion library. AAVs hold tremendous promise as delivery vectors for clinical gene therapy, and packaging is a general prerequisite for delivering genetic material to a target tissue. Standard diversified libraries for engineering effective AAV delivery vectors contain a high proportion of variants that are unable to assemble or package their genomes,

which often limits the effectiveness of downstream selections for desired properties such as efficient infection of human tissues. Using our machine learning-guided design framework, we systematically design effective starting libraries that are as diverse as possible whilst being biased towards variants that are able to assemble and package the viral genome efficiently. Specifically, we design a library of peptide insertions into the AAV capsid that achieves five-fold higher packaging fitness than the standard insertion library—known as the "NNK" library—with negligible reduction in diversity. We further demonstrate the general utility of our designed library on a downstream task to which our design approach was agnostic: infection of primary human brain tissue. Compared to the standard NNK library, our machine learning-designed library contains approximately 10-fold more variants that successfully infect the human brain.

Next, we highlight a key shortcoming of the above predictive modeling approach—namely, its extremely limited ability to share information across related but non-identical reads—that prevents it from making effective use of sequencing data in many settings of interest. We introduce *model-based enrichment* (MBE) to overcome this shortcoming. MBE is based on a new perspective of differential sequencing analysis that uses sound theoretical principles from the density ratio estimation field in machine learning, is easy to implement, and can trivially make use of advances in modern-day machine learning classification architectures or related innovations. We evaluate MBE empirically, both in simulation and on real experimental data, and show that it improves accuracy compared to current ways of performing sequencing-based differential analyses—including the previous section's predictive modeling approach. The greater flexibility of our new approach enables effective analysis across a broader range of common experimental setups than can currently be achieved, thereby expanding the set of biological applications for which one can learn accurate predictive models to guide library design.

Finally, we highlight some remaining challenges for machine learning-guided library design, including research opportunities into combining multiple sources of biological information in the design process. In summary, this dissertation presents a number of machine learning techniques that can be brought to bear on the problem of designing improved starting libraries for biological screens and selection experiments. The insights from this work provide further motivation for researchers to combine laboratory experiments with tools from machine learning to efficiently engineer novel functional protein and DNA sequences.

Contents

List of Figures

List of Tables

Acknowledgments

Without the support of my personal and professional communities, this dissertation would not exist.

I would, first, like to thank my parents, Carol and Kofi. For as long as I can remember, my mother has been among my greatest sources of inspiration; without her, it would never have occurred to me to dream that I might find myself in a job like this, at a place like Berkeley. My father has consistently encouraged me to pursue my passions and served as an oft-needed reminder that there is no need to panic. I also want to thank my siblings, Ama and Abrefa, whose love for me and belief in my ability to succeed never seem to waver. There is no doubt in my mind that the time I spent with my family over the course of my graduate studies inspired me to be better and think more creatively—I cannot imagine completing this work without their support.

I am grateful to my advisors, Professors Michael Jordan and Jennifer Listgarten, for the freedom they provided me to pursue my own research interests, and for the patient support they offered me through the long and uncertain process of discovering those interests. I have also been fortunate to work with a number of inspiring researchers in both machine learning and bioengineering. I thank Professor David Schaffer for his enthusiasm and much-needed expertise that helped guide my research, and Professor Danqing Zhu who dedicated precious time, energy, and experimental resources to translate my machine learning ideas into real laboratory experiments. I am grateful to Professor Nilah Ioannidis for serving on my qualifying exam committee and providing thoughtful critiques of my work that helped refine my research trajectory.

To my colleagues in the ScienceML research group: thank you for all of your support and guidance. In particular, thank you to David Brookes for introducing me to the library design problem, and to Hunter Nisonoff for your generous, consistent emotional and intellectual support at all the most crucial moments. I am grateful to Clara Wong-Fannjiang, Chloe Hsu, and Junhao Xiong for all of the insightful discussions, compassion, and impromptu adventures.

Last, but certainly not least, I would like to express my appreciation for my personal support network. Caroline Frost, Jotthe Kannappan, Lisa Wolbert, Madelyn Boslough, Shelly Najera, and Zainab Taymuree continually inspire me with their brilliance, introduce me to ideas both inside and outside of my field, challenge me to adopt new perspectives, and fill my life with laughter and home-cooked meals.

Chapter 1

Introduction

Genetic or *protein engineering* refers to the process of constructing new DNA or protein sequences with particular properties or functions of interest. Such engineering techniques have successfully been used in applications spanning antibody design [28, 34]; protein binding [63, 111]; improving enzyme activity and thermostability [78, 80]; engineering gene therapy delivery vectors [12, 61, 62, 116]; regulatory element engineering [66] and beyond.

One powerful and widely-used approach to genetic and protein engineering leverages high-throughput sequencing to assay a library of up to billions of sequences for a specific property, in parallel [77, 45, 54, 109]. This approach, which we will refer to as a *high-throughput selection experiment*, typically involves:

1. constructing a starting diversified library,
2. subjecting the starting library to a round of selection for a property of interest, and
3. amplifying and sequencing both the pre- and post-selection libraries.

Such high-throughput selection experiments are frequently used for directed evolution [77, 45] (Fig. 1.1), deep mutational scanning [9, 63, 23, 54, 79, 109], and functional enrichment analysis [108], and have wide-ranging biologically-significant applications, including: improving thermostability [78]; designing antibodies and therapeutics [28, 34]; profiling pathogen proteomes for epitopes and major histocompatibility complex binding [35, 72]; and assessing binding [46, 63, 111], catalytic activity [78, 80], and packaging efficiency or infectivity of viral vectors [12, 61, 62, 116].

The overall chance of successfully identifying novel or interesting sequences in a high-throughput selection experiment depends on the quality of the starting library. Ideally, a starting library should:

- *be sufficiently diverse.* A more diverse library is more likely to contain rare beneficial mutations and to be useful for a broad variety of downstream selections [92].

- *contain mostly functional variants.* A library that contains a high proportion of non-functional variants—for example, proteins that do not fold—effectively wastes screening effort since it requires the time and resources of a large library, but yields insights that could easily have been obtained using a much smaller one [92].

- *be cheap and easy to construct.* Libraries that are expensive to construct are less widely useful. For example, the high cost of synthesizing individual genes [43] prohibits the widespread use of fully-designed libraries that must be constructed using individual synthesis. Currently, libraries produced using individual synthesis tend to contain fewer successful sequences than larger libraries that are produced using cheaper but more constrained synthesis methods [104].

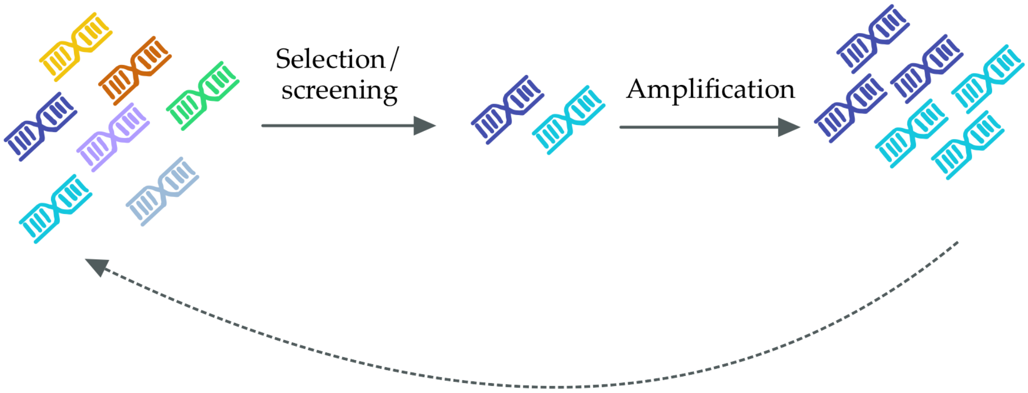


Figure 1.1: Visual representation of a high-throughput selection experiment (solid lines), and directed evolution (solid and dotted lines), which consists of high-throughput selection experiments performed iteratively.

Developing general methods for designing starting libraries with these criteria in mind promises to improve the speed with which desirable protein and DNA sequences can be discovered. *Rational design* is one potential method for producing efficient starting libraries. Rational design produces libraries using prior knowledge—such as three-dimensional structure, catalytic mechanism, and the location of active sites or variable regions—and physical models. The resulting libraries are often strongly biased towards functional sequences, which often yields an increased overall success rate and decreases the overall number of sequences that must be experimentally screened [52]. Rational design has been used successfully to improve the efficacy, immunogenicity, and pharmacokinetics of protein and antibody therapeutics [52]; to design enzymes with increased substrate specificity [106]; and to improve the thermostability of polyethylene terephthalate-degrading enzymes [85]. However, rational design's reliance on prior physical knowledge severely limits its utility; there is great need for general methods that do not require previous mechanistic understanding of desired properties. Data-driven approaches to *in silico* library design promise to produce efficient starting libraries without requiring detailed physical understanding of the properties being studied. Machine learning provides a particularly promising set of techniques for developing such data-driven methods, as it is well-suited to analyzing large quantities of sequencing data produced by high-throughput selection experiments.

Many different machine learning techniques have been applied to genetic and protein engineering: several recent works have used machine learning to analyze local fitness landscapes based on data from high-throughput selection experiments [70, 61, 63], predict fitness effects of mutations from evolutionary data [32, 75, 33, 59], and suggest new libraries and guide directed evolution experiments [113, 111, 12, 116]. This dissertation will touch on several of these techniques, and highlight how they can be used successfully to design diverse, functional libraries that can be used as high-quality starting points for downstream selection experiments. Specifically, in Chapter 2, we define a framework for machine learning-guided library design that combines a predictive model of high-throughput selection experiments with a library design objective. We apply this framework in the gene therapy context by designing a library of adeno-associated virus (AAV) capsid sequences that optimally balances predicted packaging ability and sequence diversity. The accuracy of the predictive model used to guide design is crucial to the overall chance of producing successful libraries. In Chapter 3, we introduce *model-based enrichment* (MBE), an improved predictive modeling approach for differential analysis using high-throughput sequencing data that is based on a new perspective of sequencing-based assays and selection experiments. While MBE has implications for differential sequencing analysis more generally, we demonstrate empirically, both in simulation and on real data, that it leads to more accurate predictive models of high-throughput selection experiments across a broad range of common experimental setups. Consequently, by expanding the set of experimental applications for which one can learn accurate predictive models, MBE further broadens the applicability of the machine learning-guided library design approach from Chapter 2.

In the remaining sections of this chapter, we will provide general background for the predictive modeling and library design problems.

1.1 Predictive modeling for high-throughput selection

A key desired outcome from a high-throughput selection experiment is to use the resulting sequencing data to quantify the change in relative abundance of a particular sequence after the bulk screen and selection occurs, which serves as a proxy for the sequence's *fitness* in the selection. This type of quantification is often referred to as estimating the *log-enrichment* of a sequence between the pre- and post-selection conditions [23, 54, 79]. Due to our interest in library design, we focus here on high-throughput selection experiments. However, this type of log-enrichment estimation is performed much more broadly—for example, in functional enrichment analysis [108] and differential analyses of RNA-seq and ATAC-seq experiments [94, 76, 112, 48]. In general, accurately estimating log-enrichment for large sequence libraries enables identification of sequences that are more (or less) likely to have a desired property, and has the potential to reveal insights into the sequence determinants of the property of interest.

Increasingly, log-enrichment estimates are also being used as supervised labels for training machine learning model of fitness [12, 46, 70, 111, 116, 86, 82, 25] that allow one to predict log-enrichment for previously unobserved sequences, and often do so more accurately than popular physics-based and unsupervised machine learning methods such as Rosetta and DeepSequence [25]. In Chapter 2, we present a supervised machine learning approach based on log-enrichment estimates and use it to learn a model of AAV capsid fitness from data generated by a high-throughput selection experiment. In Chapter 3, we highlight a key shortcoming of existing log-enrichment-based approaches—namely, their ineffective use of sequencing data due to their extremely limited ability to share information across related but non-identical sequences—and introduce an alternative supervised machine learning approach for differential sequencing analyses. We show that our new approach improves predictive modeling accuracy for high-throughput selection experiments compared to existing approaches like the one in Chapter 2.

1.2 Designing sequence libraries

High-throughput selection experiments are one way to approach the genetic or protein engineering problem. As such, the high-level motivation is, typically, to identify one or a few novel sequences with high fitness compared to those previously observed. However, it is often preferable to design large libraries of sequences rather than individual ones. For example, one might wish to construct a library that can be used as an effective starting point for additional iterations of high-throughput selection experiments, or to exploit cost-effective library construction techniques to generate and screen a larger number of novel sequences than can be characterized using expensive individual gene synthesis techniques.

Designing a library entails choosing a library construction technique and specifying its variable parameters. Popular techniques for constructing libraries include:

- *error-prone polymerase chain reaction (PCR)* [17, 50, 36], a random mutagenesis technique that introduces random point mutations into a DNA segment;

- *combinatorial recombination* [77, 22, 62], which generates a library of chimeras by recombining segments of parent sequences at fixed crossover positions;

- *DNA shuffling* [88, 26], which involves randomly recombining variable length segments of parent sequences; and

- *peptide insertion* [19], which introduces short, random, fixed-length peptide sequences into a fixed location in a given background sequence.

In computational library design, the parameters that control these construction techniques are set using computational strategies. For example, one can identify crossover positions suitable for combinatorial recombination by computationally minimizing disruptions to three-dimensional structure [100, 62], or use computational analysis of single-substitution fitness measurements to choose genomic locations and mutation probabilities for mutagenesis [81]. This dissertation focuses on *machine learning-guided library design* which uses a predictive model of fitness to guide the choice of parameters for a given library construction technique.

Recall that a high-quality starting library for high-throughput selection experiments is sufficiently diverse and biased towards functional sequences. One can encourage both of these qualities using machine learning-guided library design by defining a multi-objective optimization procedure that considers both the diversity of the library and the fitnesses of its sequences, based on a trained predictive model. However, these two desiderata are, generally, competing—for example, the library with the highest average predicted fitness is comprised of (many copies of) only the single sequence with the maximum predicted fitness, and therefore has the worst possible sequence diversity. In Chapter 2, we describe a machine learning-guided library design approach that explicitly considers the trade-off between a library's average predicted fitness and its diversity, and produces libraries (or parameters for the relevant library construction technique) which optimally balance these two objectives. Further, we demonstrate this framework's ability to systematically design a more effective AAV peptide insertion library that achieves a five-fold higher packaging fitness than a standard insertion library with negligible reduction in diversity.

Chapter 2

Machine Learning-Guided Library Design with Optimal Trade-off Control

This chapter contains material reproduced with permission from:
Danqing Zhu et al. “Machine learning-based library design improves packaging and diversity of adeno-associated virus (AAV) libraries”. In: *bioRxiv* (2021)

2.1 Introduction

In the previous chapter, we highlighted key qualities of starting libraries that increase the overall chance of successfully discovering desirable sequences using high-throughput selection experiments: a good starting library should be cost-effective, diverse, and biased towards sequences with high fitness. By producing starting libraries that are more likely to have these qualities, generalizable computational library design methods have the potential to increase the efficiency with which novel protein and DNA sequences are discovered. In this chapter, we describe a machine learning method for designing effective starting libraries. Our machine learning-guided design framework combines a predictive model of high-throughput selection experiments with a design optimization problem that optimally balances average predicted fitness and diversity. We demonstrate the utility of this framework by applying it to design a diverse library of insertion sequences into the adeno-associated virus (AAV) capsid with improved ability to package the viral genome. Moreover, we show that this designed library serves as a better starting library for a clinically-relevant downstream selection task—namely, infection of human brain tissue—than the current standard AAV insertion library.

This chapter is structured as follows: in Section 2.2, we describe a standard method for quantifying sequences using the data produced by a high-throughput selection experiment; in Sections 2.3 and 2.4, we introduce the predictive modeling and library optimization components of our machine learning-guided library design framework; and in Section 2.5, we apply this framework to design a library of AAV peptide insertion sequences that (i) achieves a five-fold improvement in overall packaging ability when compared to the standard insertion library and (ii) makes a better starting library for infecting human brain tissue.

2.2 High-throughput selection data

Recall from Chapter 1 that in a high-throughput selection experiment, one (1) synthesizes a starting, or pre-selection, library of sequences, (2) performs a round of selection on the starting library to produce a post-selection library, and then (3) subjects each of the pre- and post-selection libraries to high-throughput sequencing. This yields data $\mathcal{D} = \{(r_i, y_i)\}_{i=1}^{M}$ where r_i is the i^{th} read's sequence and y_i is a binary $-1/+1$ label indicating whether the read r_i arose from sequencing the pre- or post-selection library, respectively. It is often convenient to represent $\mathcal{D}$ in terms of unique sequences: $\mathcal{D}' = \{(x_i, n_i^{\text{pre}}, n_i^{\text{post}})\}_{i=1}^{M'}$ where $\{x_i\}_{i=1}^{M'} \subseteq \{r_i\}_{i=1}^{M}$ is the set of unique observed sequences, $n_i^{\text{pre}} = \sum_{(r,y)\in\mathcal{D}} \mathbb{1}\{r = x_i\}\mathbb{1}\{y = -1\}$ is the observed pre-selection read count for sequence x_i, and $n_i^{\text{post}} = \sum_{(r,y)\in\mathcal{D}} \mathbb{1}\{r = x_i\}\mathbb{1}\{y = +1\}$ is the corresponding post-selection read count (Fig. 2.1).

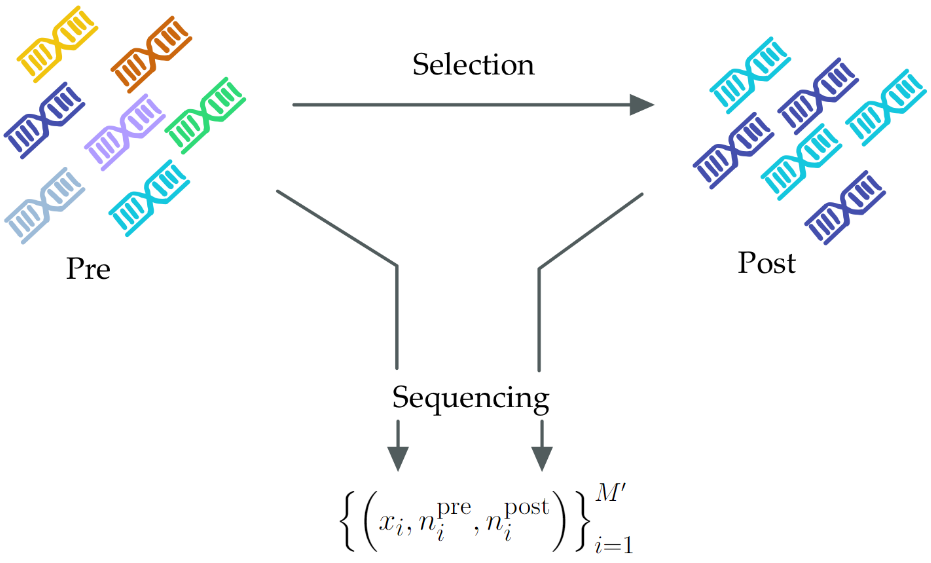

Selection
Pre
Post
Sequencing
$\left\{\left(x_i, n_i^{\text{pre}}, n_i^{\text{post}}\right)\right\}_{i=1}^{M'}$

Figure 2.1: Schematic of the sequence dataset produced by a high-throughput selection experiment.

Intuitively, the change in relative abundance of each sequence in a high-throughput selection experiment is related to the extent to which the sequence passes the selection. Thus, by accurately estimating this change for a large number of library sequences, one can identify sequences that are more (or less) likely to have the property targeted by the selection. In other words, this change in relative abundance—commonly referred to as "log-enrichment" [23, 39, 54, 79, 109]—is a proxy for sequence *fitness* with respect to the selection. As such, a key desired outcome from a high-throughput selection experiment is to use the generated data, $\mathcal{D}'$, to estimate the log-enrichment for each observed sequence.

The standard count-based log-enrichment (cLE) estimate—*i. e.,* the log-ratio of post- and pre-selection sequencing counts—for each sequence x_i in $\mathcal{D}'$ is defined as

$$\log e_i = \log\left(\left(\frac{n_i^{\text{post}}}{N^{\text{post}}}\right)\left(\frac{n_i^{\text{pre}}}{N_{\text{pre}}}\right)^{-1}\right), \tag{2.1}$$

where $N^{\text{pre}} = \sum_{i=1}^{M'} n_i^{\text{pre}}$ and $N^{\text{post}} = \sum_{i=1}^{M'} n_i^{\text{post}}$. Several previous works have studied the statistical properties of these count-based estimates [39, 79, 54], which are commonly used to measure and analyze fitness from sequencing-based assays [23]. For example, cLE has been successfully to study local fitness landscapes of binding proteins [63], to measure antibody affinity [28, 34], and to assess the infectivity of viral vectors [62]. Increasingly, cLE estimates are also being used to train machine learning models that can be used to predict the fitness of unobserved sequences [12, 46, 111, 116, 86, 25]. In the next section, we introduce a model-training procedure that leverages cLE estimates and explicitly incorporates the uncertainty in these estimates to improve predictive performance.

2.3 Predictive modeling using count-based log-enrichment

In Section 2.2, we described the sequencing data produced by a high-throughput selection experiment, $\mathcal{D}$, as well as a quantity known as log-enrichment which can be interpreted as a proxy for fitness and estimated from the sequencing counts in $\mathcal{D}$. Next, we describe a supervised machine learning approach that uses cLE estimates to learn a predictive model. Specifically, we derive a new weighted least squares regression procedure for learning a model that predicts log-enrichment from sequence.

Recall that for a given sequence, x_i, the cLE estimate, $\log e_i$, is equal to the log-ratio of normalized post- and pre-selection sequencing counts (Eq. 2.1). Intuitively, $\log e_i$ is a more reliable estimate of fitness for sequences with more observed data. For example, for fixed N^{pre} and N^{post}, a sequence with $n_i^{\text{pre}} = 1$ and $n_i^{\text{post}} = 2$ has the same $\log e_i$ as a sequence with $n_i^{\text{pre}} = 100$ and $n_i^{\text{post}} = 200$, yet the latter is supported by 100 times more evidence. Indeed, for sequences with higher counts, the cLE estimate has lower variance. To capture this heteroscedasticity in our training procedure, we compute sample weights based on the asymptotic variance of $\log e_i$:

$$\sigma_i^2 = \frac{1}{n_i^{\text{post}}}\left(1 - \frac{n_i^{\text{post}}}{N^{\text{post}}}\right) + \frac{1}{n_i^{\text{pre}}}\left(1 - \frac{n_i^{\text{pre}}}{N^{\text{pre}}}\right). \tag{2.2}$$

Although previous works have studied these asymptotic variances [39], to our knowledge, we are the first to use them to weight data points within a supervised machine learning procedure. Specifically, we train a model, f_θ with learnable parameters θ, using the weighted least squares loss

$$\ell(\mathcal{D}') = \sum_{i=1}^{M'} \frac{1}{2\sigma_i^2}(\log e_i - f_\theta(x_i))^2 \tag{2.3}$$

which accounts for the heteroscedastic noise in the observed log-enrichment estimates: when the counts n_i^{pre} and n_i^{post} are low, σ_i^2 is larger, indicating that $\log e_i$ is a noisier estimate of fitness when computed from less sequencing data. We will refer to this modeling approach as the *weighted log-enrichment regression* (wLER) approach.

In summary, we have introduced wLER, a predictive modeling approach with three steps: (1) compute cLE estimates from observed sequencing data, (2) estimate the noise in the observed log-enrichment estimates, and (3) use these two pieces of information to train a supervised machine learning model via weighted least squares regression.

2.4 Model-guided library design

In this section, we develop a general library design framework that leverages a trained predictive model of fitness to suggest an improved starting library of sequences.

Maximum entropy library design

Recall from Chapter 1 that a high-quality starting library is sufficiently diverse and biased towards sequences with high fitness. We define an optimization framework that explicitly encourages both of these qualities by considering both the average fitness of the library—according to a user-specified predictive model, such as a trained neural network—and its diversity. Our approach represents libraries as discrete probability distributions over sequence space (Fig. 2.2). This representation allows our framework to design a library by optimizing over the underlying library distribution, and to measure its diversity using statistical entropy. Entropy is a common diversity metric for probability distributions that has been used extensively in ecology to describe the diversity of populations [97]. Our library design approach aims to find *maximum entropy distributions* that optimize statistical entropy whilst also satisfying a constraint on average predicted fitness. By varying the severity of this constraint, one can use our framework to trace out an optimal trade-off curve between average predicted fitness and statistical entropy. The ability to generate and analyze this trade-off curve distinguishes our approach from related model-guided design approaches which typically analyze library diversity *post hoc* [12, 8]. To do so, our proposed framework builds on a body of recent research that explores design methods for combinatorial recombination libraries that optimally balance multiple objectives [65, 115, 99]. However, these previous works only incorporate objectives which encourage sequences in the library to differ from a given set of naturally-occurring sequences, without incorporating a notion of diversity that explicitly encourages library sequences to differ from one another. Our ap-

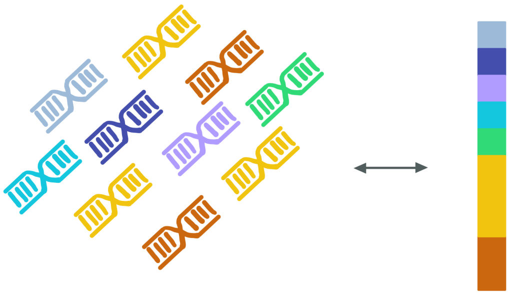


Figure 2.2: Visualization of the correspondence between a sequence library (left) and its underlying probability distribution over sequence space (right).

proach, therefore, improves upon these techniques by (i) generalizing to library construction techniques beyond combinatorial recombination, (ii) allowing any predictive model of fitness to be used, and (iii) explicitly navigating a trade-off with entropy, a measure of diversity between library sequences.

Our design approach is motivated by the maximum entropy optimization problem

$$\max_{p \in \mathcal{P}} H[p] \quad \text{s.t.} \quad \mathbb{E}_p[f_\theta(x)] \geq \alpha; \quad \alpha \in \mathbb{R}, \tag{2.4}$$

where $\mathcal{X}$ is the space of all sequences that may be included in a library (*e. g.,* all length 21 DNA sequences), $\mathcal{P}$ represents the set of all discrete probability distributions over $\mathcal{X}$, $H[p] = \mathbb{E}_p[-\log p(x)]$ is the statistical entropy of the library distribution p, f_θ is a trained predictive model of fitness, and α is the constraint on average predicted fitness. This optimization problem has a closed-form solution known as the *maximum entropy distribution* [37],

$$p_\lambda(x) = Z(\lambda)^{-1} \exp\left(\lambda^{-1} f_\theta(x)\right), \tag{2.5}$$

where the constant $\lambda > 0$ is a monotonic function of the constraint cutoff α, and $Z(\lambda) = \sum_{x \in \mathcal{X}} \exp\left(\lambda^{-1} f_\theta(x)\right)$ is a normalizing constant. Each distribution p_λ represents a point on the optimal trade-off curve between average predicted fitness and statistical entropy; for a given λ, it is not possible to specify another library distribution that simultaneously achieves higher statistical entropy and higher average predicted fitness than p_λ. The parameter λ controls the balance between diversity and average predicted fitness: when λ is large, p_λ has high diversity (i.e., $H[p_\lambda]$ is large), and as λ is decreased, the corresponding p_λ concentrates more probability mass on sequences with high f_θ (i.e., $H[p_\lambda]$ decreases and $\mathbb{E}_{p_\lambda}[f_\theta(x)]$ increases). Theoretically, the entire optimal trade-off curve can be traced out by calculating the average predicted fitness and statistical entropy of p_λ for every possible value of $\lambda > 0$. In practice, it is sufficiently useful to trace out a curve by choosing a finite set of λ values to evaluate.

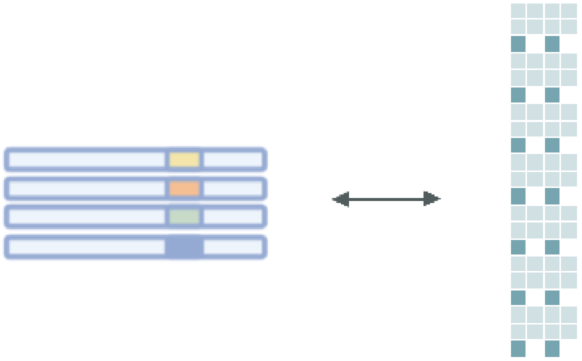


Figure 2.3: Visualization of the correspondence between a degenerate oligo peptide insertion library (left) and its underlying position-wise nucleotide probability distribution (right).

Maximum entropy design for constrained libraries

Thus far, we have described how our design framework can be used to select a library distribution, p_λ, that optimally balances average predicted fitness and statistical entropy. However, in practice, the theoretical library represented by p_λ often cannot be exactly realized experimentally. If constructing libraries using individual gene synthesis techniques, one can sample individual sequences from p_λ to design a realizable library. However, given the current high cost of individual synthesis, it is currently not cost-effective to synthesize large numbers of individual sequences [104, 43]. Instead, our design framework can be used to design libraries constructed using more affordable—but more constrained—experimental techniques that are as close as possible to the theoretical p_λ.

To apply our maximum entropy design framework to produce libraries that can be constructed using a specific experimental technique, we define a set, $\mathcal{Q}$, containing all library distributions that are realizable using the chosen construction mechanism. For example, in Section 2.5 we will design libraries that can be constructed using *degenerate oligo-synthesis* techniques that specify libraries using position-wise nucleotide probabilities (Fig. 2.3), and thus the corresponding $\mathcal{Q}$ contains distributions of the form

$$q_\phi(x) = \prod_{j=1}^{L} \sum_{k=1}^{4} q_{\phi_j}(k)\delta_k(x^j) \tag{2.6}$$

where L is the sequence length, k indexes the four possible nucleotides, $\phi \in \mathbb{R}^{L\times 4}$ is a matrix of position-wise distribution parameters, ϕ_j is the j^{th} row of ϕ, x^j is the j^{th} position of x, $\delta_k(x^j) = 1$ if the $x^j = k$ and zero otherwise, and

$$q_{\phi_j}(k) = \frac{e^{\phi_{jk}}}{\sum_{l=1}^{4} e^{\phi_{jl}}} \tag{2.7}$$

denotes the probability of observing the k^{th} nucleotide option at the j^{th} position in the sequence.

To apply the maximum entropy formulation (Eq. 2.4) to design libraries that are constrained to be in $\mathcal{Q}$, we take a variational approach: for a given λ, we find the library distribution $q \in \mathcal{Q}$ that is the best approximation to p_λ in terms of the Kullback-Leibler divergence,

$$\min_{\mathcal{Q}} D_{KL}\left[q \| p_\lambda\right] = \max_{\mathcal{Q}} \mathbb{E}_q[f_\theta(x)] + \lambda H[q]. \tag{2.8}$$

As a concrete example, for libraries specified via position-wise nucleotide probabilities, this is equivalent to solving

$$\phi_\lambda = \operatorname{argmax}_\phi \mathbb{E}_{q_\phi}[f_\theta(x)] + \lambda H[q_\phi], \tag{2.9}$$

which can be efficiently approximated using Stochastic Gradient Descent (Section A.7).

In the following section, we demonstrate our ability to design diverse, experimentally-realizable libraries of AAV insertion sequences by solving this maximum entropy design problem for constrained libraries (Eq. 2.9).

2.5 Designing a library of AAV capsid insertion sequences

Next, we use our machine learning-guided design framework—which combines our wLER (Section 2.3) and maximum entropy design (Section 2.4) approaches—to produce improved diversified libraries of AAV capsid proteins. AAVs hold major promise as delivery vectors for gene therapy. While naturally-occurring AAVs can be clinically administered safely and, in some cases, efficaciously, they have a number of shortcomings that limit their use in many human therapeutic applications. For example, naturally-occurring AAVs do not target delivery to specific organs or cells, have limited delivery efficiency, and are susceptible to pre-existing neutralizing antibodies [50, 18, 95]. Consequently, directed evolution of the AAV capsid protein has emerged as a powerful strategy for engineering therapeutically-suitable AAV variants. Although successes have been achieved with directed evolution [18, 95, 5, 36, 16, 42], several challenges slow progress. For instance, a substantial fraction of the variants in the starting libraries for these selections are unable to assemble properly or package their payload efficiently [95, 61, 2, 14]—a basic functional requirement for any selection. Consequently, much of the library is wasted, thereby decreasing the chance of successfully achieving the desired engineering goal in downstream selections. Because of the high prevalence of non-packageable variants in starting libraries, it is typical to perform one initial round of packaging selection (*i. e.,* packaging, amplification, and recovery of viral genomes) to remove these. However, as a consequence of this step, the diversity of the library is dramatically reduced—often by as much as half [2, 14]—before the selection of primary interest has been applied. Diversity of the starting library is one of the key determinants of success in directed evolution because it increases the chances of identifying rare beneficial mutations within the library. Thus, if one could redesign the starting library to have high diversity whilst improving packaging ability, one would increase the probability of success for any general AAV directed evolution goal. Our machine learning-guided library design framework allows us to do just this. In this section, we apply this framework to design a library that balances the requirements that this library should contain AAV variants that package, while also being as diverse as possible.

Recent studies have applied machine learning models trained on experimental data to generate novel AAV variants [51, 12]. However, these studies examined diversity *post hoc* and provided no way to systematically navigate an optimal trade-off between diversity and packaging ability. In contrast, our approach (i) allows for the use of any predictive model of packaging ability, (ii) explicitly addresses and controls the diversity within the designed library, and (iii) is broadly applicable to different kinds of library constructions.

We instantiated and evaluated our library design approach by designing a library of 7 amino acid (7-mer) peptide insertions into AAV serotype 5 (AAV5) to optimally balance diversity and average packaging fitness. The choice of AAV5 was motivated by its immediate clinical relevance [102]: among the natural AAV serotypes, AAV5 has been suggested as an especially promising candidate for clinical gene delivery because of the low prevalence of pre-existing neutralizing antibodies and successful clinical development for hemophilia B [10, 101]. We focus, specifically, on peptide insertion libraries because they are both simple and highly practical, having already been translated to the clinic (*e. g.,* NCT03748784, NCT04645212, NCT04483440, NCT04517149, NCT04519749, NCT03326336, and NCT05197270) [30].

We proceed as follows: first, we use cLE (Section 2.2) to estimate the packaging fitness of variants in a standard insertion library known as the "NNK" library, which is widely-used as a starting point for experimental selections of AAV capsids. We, then, use these estimated packaging efficiencies as labels to build a predictive model from peptide insertion sequence to packaging fitness using wLER (Section 2.3). Finally, we apply the design approach from Section 2.4 to systematically trade off library diversity with packaging efficiency. We show that the resulting designed library has five-fold higher average packaging fitness than the standard NNK library with a negligible decrease in diversity, suggesting that our library will be more broadly useful. Moreover, when we subjected the NNK library to one round of packaging selection, the resulting pool of variants still had lower packaging fitness than that of our designed library while also being substantially less diverse. Finally, to demonstrate the general downstream utility of our designed library for engineering tasks for which it was not specifically designed, we show that it infected primary human brain tissue with substantially higher efficiency than the NNK library. To the best of our knowledge, this is the first machine learning-guided AAV capsid library design used for selection in human tissue.

	A	T	C	G
Position 1	0.25	0.25	0.25	0.25
2	0.25	0.25	0.25	0.25
3	0	0.5	0	0.5

Table 2.1: Table of nucleotide probabilities specified by the NNK degenerate codon.

Library preparation and packaging selection

In our study, we used AAV5 libraries with a variable 7-mer sequence inserted at positions 575-577 in the viral protein monomer, within a loop at the 3-fold symmetry axis associated with receptor binding and cell-specific entry [57, 69] (Section A.1). For the standard NNK library, each of the 7 amino acids in the insertion sequence is sampled at random from the distribution corresponding to the NNK degenerate codon, which specifies a uniform distribution over all four nucleotides (N) in the first two positions of a codon, and equal probability on nucleotides G and T (K) in the third position (Table 2.1). The K in the third position was chosen to reduce the chance of stop codons which typically render the protein non-functional. Although NNK libraries are among the most promising AAV libraries [18], a substantial fraction ($> 50\%$) of the variants in these libraries fail to package (*i. e.,* do not assemble into viable capsids) and many more have lower packaging fitness than the parental AAV5 virus [2, 14].

First, we experimentally synthesized roughly 10^7 variants from the NNK library to yield the NNK *pre-packaged library*. The plasmid library was then packaged and the resulting viral particles were harvested and purified, and their genomes extracted [117], yielding the NNK *post-packaged library* (Fig. 2.4; Section A.2). The sequences from both pre- and post-packaged libraries were PCR amplified and deep sequenced with a single read run on Illumina NovaSeq 6000. This process yielded 49,619,716 and 55,135,155 sequencing reads corresponding to the pre- and post-packaged libraries, respectively. Each read contained a fixed 21 base pair (bp) primer sequence and variable 21 bp sequence containing the nucleotide insertion sequence. We filtered the reads, removing those that either contained more than two mismatches in the primer sequences or ambiguous nucleotides. After this filtering, the pre- and post-packaged libraries contained 46,049,235 and 45,306,265 reads, respectively. The insertion sequences were, then, extracted from each read and translated to 7-mer amino acid sequences. This process yielded read counts for 8,552,729 unique 7-mer insertion sequences.

For each unique insertion sequence, we used pre- and post-packaged read counts to calculate a cLE estimate (Eq. 2.1) quantifying its effect on packaging. Note that only 218,942 of the 8,552,729 unique insertion sequences appeared in both the pre- and post-packaged libraries. A pseudo-count of 1 was added to each count prior to computing log-enrichment estimates to avoid $\log 0$ when the sequence appeared in only one of the two libraries. In all cases, the natural logarithm was used. We also estimated a variance associated with each log-enrichment estimate (Eq. 2.2).

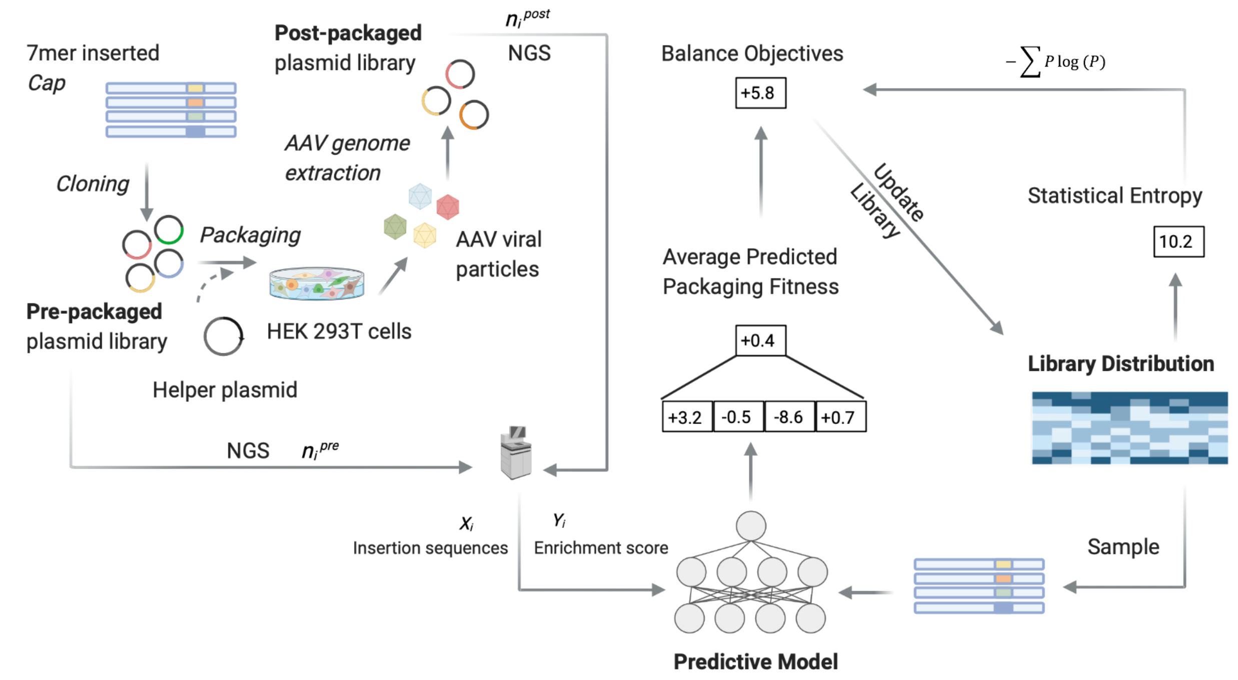


Figure 2.4: Experimental workflow for generating pre- and post-packaged AAV5 7-mer library sequencing data. Libraries were constructed by inserting a variable 7-mer sequence in the viral protein monomer. After packaging [62, 57], AAV library vectors were produced by transfection of HEK293T cells, and capsid sequences were recovered by PCR and subjected to next-generation sequencing (NGS). Experimental sequencing data were used to build a supervised regression model where the target variable reflects the packaging success of each insertion sequence. The predictive model was then systematically inverted to design libraries that optimally balance diversity (statistical entropy) and average predicted packaging fitness.

Training and evaluation of predictive models of packaging fitness

To find the best type of predictive model to use for our machine learning-guided library design, we compared seven classes of regression models: three linear models and four feed-forward neural networks (NNs). Each model was trained using the cLE estimates and corresponding variance estimates in the wLER framework (Section 2.3). The three linear models differed in the set of input features used: one used the “Independent Site” (IS) representation wherein individual amino acids in each 7-mer insertion sequence were one-hot encoded; another used a “Neighbors” representation comprised of the IS features in addition to pairwise interactions between all positions that are directly adjacent in the amino acid sequence; and the third used a ”Pairwise” representation comprised of the IS features in addition to all pairwise interactions among positions in the sequence. All NNs used the IS features, as these models have the capacity to construct higher-order interaction features from the IS features. Each NN architecture used exactly two densely-connected hidden layers with tanh activation functions. The four NN models differed in the size of the hidden layers, with each using either 100, 200, 500, or 1000 nodes in each hidden layer.

For the linear architectures, the weighted regression loss (Eq. 2.3) is a convex function which can be solved exactly for the minimizing model parameters. In order to stabilize training, we used a small amount of ℓ_2 regularization for the Neighbors and Pairwise representations (with regularization coefficients 0.001 and 0.0025, respectively). For the NNs, the loss is non-convex and we use stochastic optimization techniques to solve for suitable model parameters. We implemented these models in TensorFlow [1] and used the built-in implementation of the Adam optimization algorithm [40] to approximately solve Equation 2.3.

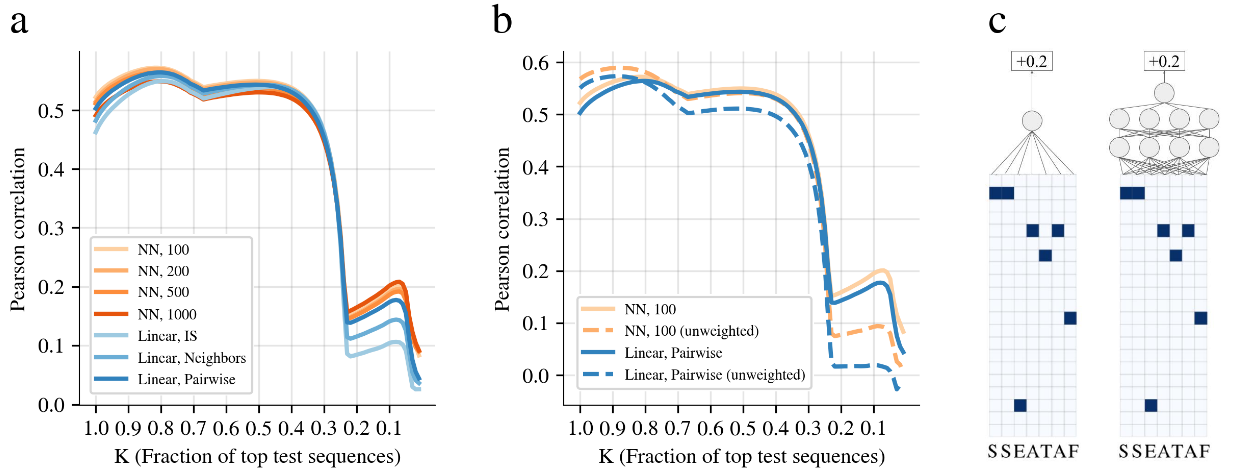


Figure 2.5: (a) Comparison of different model architectures' abilities to predict packaging log-enrichment from AAV5 7-mer insertion sequence using "top-K" Pearson correlation between model predictions and observed count-based log-enrichment estimates, where K denotes the fraction of test sequences with top-ranked observed log-enrichment used to compute correlation. (a) Comparison of seven different model architectures, including three linear models with different inputs features (Independent Sites (IS), Neighbors, or Pairwise) and four neural network (NN) architectures distinguished by the number of nodes in each hidden layer (100, 200, 500, or 1000). (b) Comparison of weighted versus unweighted least squares regression during training for the final selected "NN, 100" model and a baseline "Linear, Pairwise" model. (c) Schematic illustrations of "Linear, IS" (left) and "NN, 4" (right) predictive models.

We compared the performance of these seven models using the Pearson correlation between model predictions and observed log-enrichment estimates on a held-out test set. We randomly split the data into a training set containing 80% of the sequences and a test set containing the remaining 20%. Because our ultimate aim was to design a library of sequences that package well, we also studied how the models' predictive accuracy changed when restricted to sequences in the test set with high estimated packaging log-enrichment. Specifically, we computed the Pearson correlation on subsets of the test set restricted to the fraction K of sequences with the highest observed cLE. By varying K, we traced out a performance curve where, for lower K, the evaluation is more focused on accurate prediction of higher log-enrichment estimates rather than lower ones (Fig. 2.5a, Fig. A.2). Overall, we found that the NN models performed better than the linear models, presumably due to their capacity to capture higher-order epistatic interactions in the fitness function. We selected the NN model with 100 hidden units per layer ("NN, 100") to use for library design because it achieved relatively high predictive accuracy on the test set (Fig. A.3), and it performed similarly to the overall best-performing model ("NN, 1000") whilst using many fewer parameters.

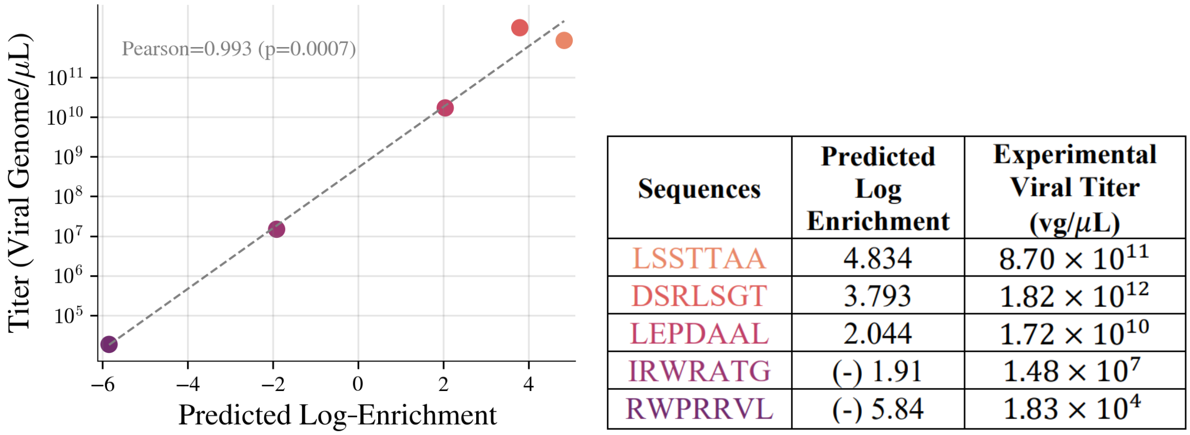
We also assessed the effect of our sequence-specific weights by retraining two of the models—the selected "NN, 100" model and the "Linear, Pairwise" model—using standard, unweighted least squares regression. We, again, used Pearson correlation to evaluate the trained models (Fig. 2.5b). Training in an unweighted manner—rather than using our wLER approach—resulted in a small performance benefit for K near 1, but substantially degraded the performance near $K < 0.25$, a regime of particular interest since it focuses on sequences with high observed cLE and our goal is to bias our designed library towards sequences with high packaging fitness. These results further supported our choice of the "NN, 100" model trained with wLER for performing library design.

Experimental validation of predictive models of packaging fitness

Before proceeding to use our predictive model to design libraries, we, first, validated the "NN, 100" model by identifying and synthesizing five individual 7-mer insertion sequences that were not present in our experimental dataset used for training. These five sequences were chosen to span a broad range of predicted log-enrichment (-5.84 to 4.83; Fig. 2.6). Each of the five variants was packaged individually into viral particles, harvested, and titered by quantifying the resulting number of genome-containing particles using digital-droplet PCR (Section A.2). High titer values indicate the variant was capable of packaging its genome properly in the assembled capsid. The agreement between model predictions and corresponding experimental viral titer measurements (between 1.83×10^4 and 8.70×10^{11} viral genomes (vg) / μL; Fig. 2.6) demonstrates that the predictive model was sufficiently accurate to be used for design. Note that the increased predictive accuracy on these five variants (Fig. 2.6) compared to the full test set (Fig. 2.5a, Fig. A.3) can be largely attributed to the choice of five sequences that spanned a large range of predicted log-enrichment (Fig. A.4).

Model-guided insertion library design

Having validated our selected predictive model, we, next, aimed to design a library that improves upon the commonly-used NNK library. In particular, our goal was to design a library that packages better than the NNK library, on average, whilst maintaining high diversity. In Section 2.4, we developed a general framework for sequence library design that can be used with our trained predictive model of packaging fitness, and is broadly applicable to different library construction mechanisms. We used this framework to design insertion libraries that can be constructed using degenerate oligo-synthesis by optimizing probabilities for each nucleotide in each position of the insertion sequence in a manner that achieves bet-



Sequences	Predicted Log Enrichment	Experimental Viral Titer (vg/μL)
LSSTTAA	4.834	8.70×10^{11}
DSRLSGT	3.793	1.82×10^{12}
LEPDAAL	2.044	1.72×10^{10}
IRWRATG	(-) 1.91	1.48×10^{7}
RWPRRVL	(-) 5.84	1.83×10^{4}

Figure 2.6: Experimental packaging titers (viral genome (vg)/μL) versus predicted packaging log-enrichment for five test variants selected to span a broad range of predicted log-enrichment. Log-enrichment scales are computed using natural logarithm.

ter overall packaging fitness than NNK, while maintaining high library diversity. A designed library will, thus, specify 84 (= 7 amino acids × 3 codon positions × 4 nucleotide options) probabilities (Fig. 2.3) that balance packaging fitness—approximated with our trained machine learning model—and sequence diversity. We refer to designed libraries specified in this manner as *position-wise nucleotide* specified.

Recall that there is an inherent trade-off between a library's diversity and its mean predicted packaging fitness. For example, mean predictive packaging fitness is maximized by a library that contains only the single variant with the highest predicted fitness, while diversity is maximized by a library that is uniformly distributed across all of sequence space irrespective of packaging fitness. The library that is most effective for downstream selection will lie between these two extremes, balancing mean predicted packaging fitness with diversity. We used our maximum entropy design framework to trace out an optimal trade-off curve (Fig. 2.7a). Note that the underlying optimization problem (Eq. 2.9) is challenging to solve exactly (*i. e.,* it is non-convex). However, the curve can be inferred approximately using a Stochastic Gradient Descent algorithm (Section A.7). Although libraries computed in this manner may not lie exactly on the theoretical optimal trade-off curve, the approximate curve, nevertheless, provides useful insights. Indeed, our trade-off curve allows us to assess what mean packaging fitness can be achieved for a given level of library diversity. As we shall see, on this basis, one can choose promising candidate libraries.

In particular, we highlight three designed libraries—D1, D2, and D3—as representative of three important regions of the trade-off curve (Fig. 2.7a). Remarkably, the NNK library has a dramatically poor mean predicted log-enrichment, much lower than any designed library. In particular, library D3 has nearly identical diversity to the NNK library, but substantially higher mean predicted packaging fitness. This observation implies that library D3 effectively dominates the NNK library—that is, it increases predicted packaging fitness without much

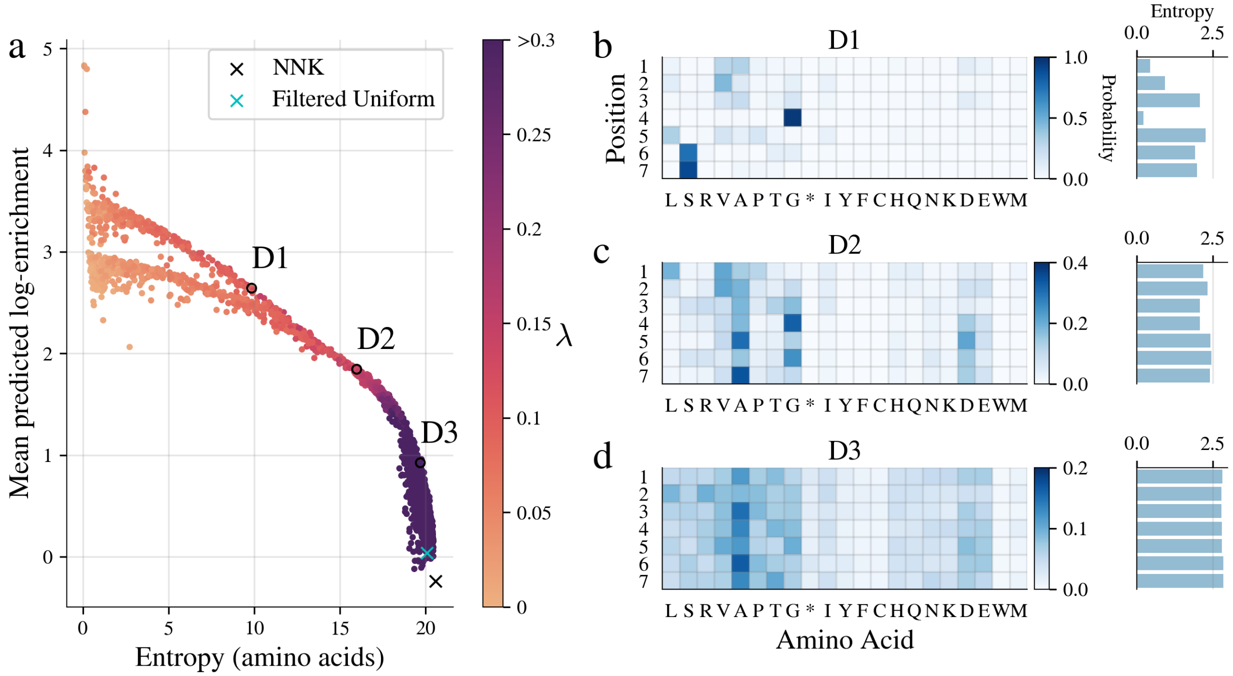


Figure 2.7: Designed AAV5 7-mer insertion libraries. Each point in (a) represents a theoretical library designed with our machine learning-guided design framework with one particular diversity constraint, λ (higher values yields more diverse library distributions). Entropy indicates diversity of the library distribution, while mean predicted log-enrichment indicates overall library packaging fitness; both quantities were computed from the theoretical library distribution. The baseline NNK library is denoted with a black "x", while a cyan "x" denotes the "filtered uniform" library that is uniform over all 21-mer nucleotide sequences except for those containing stop codons. Three designed libraries have been circled and labeled D1-3 for reference. Due to the non-convex optimization problem, some dots are suboptimal (i.e., lie strictly below or to the left of other dots) and are, therefore, further from the optimal frontier, but are displayed for completeness. (b-d) (left) designed library parameters (probability of each amino acid at each position) for the three designed libraries D1-3, respectively, and (right) the entropy of the distribution at each position.

reduction in diversity. In addition, we see that, compared to D3, library D2 is less diverse but is predicted to package better. Similarly, library D1 is less diverse than D2, but, again, is predicted to package better.

Although the original motivation for creating the NNK library was to reduce the number of stop codons, it does not eliminate them entirely. Therefore, for further comparison, we computed the mean predicted packaging fitness and diversity of the theoretical library that is uniformly distributed over all possible 7-mer sequences that do not contain any stop codons. In practice, this library—which we refer to as the "filtered uniform" library—is not realizable using a position-wise nucleotide specification strategy, but serves as a useful comparator. We find that the filtered uniform library has slightly higher mean predicted packaging fitness than the NNK library and, correspondingly, slightly lower diversity. However, these differences are negligible compared to the differences between the NNK and D3 libraries, suggesting that further removal of stop codons is not the primary mechanism by which our designed libraries achieve higher predicted packaging fitness.

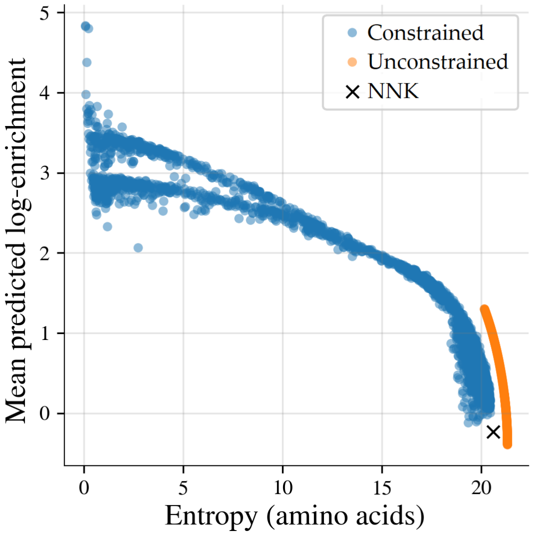


Figure 2.8: Comparison of maximum entropy unconstrained and constrained AAV5 7-mer insertion libraries. Blue points are identical to the points in Figure 2.7a with colors removed. Orange points represent unconstrained libraries designed using the same λ values as used to produce the constrained libraries.

Richer library construction methods

Recall that each position-wise nucleotide specified designed library is defined by the 84 marginal probabilities of individual nucleotides at each position in the 7 amino acid insertion (Fig. 2.7b-d, Table A.2, Table A.3). Our machine learning-guided design approach can, however, be used to guide any library construction method—for example, individual synthesis construction where we specify and synthesize individual sequences to create a library. We use the term "unconstrained" to refer to libraries that are designed with this construction method since individual synthesis offers the most control over sequences in the library.

We have focused our experiments on "constrained" position-wise nucleotide specified libraries—which cannot guarantee the inclusion of any particular sequence—because they are currently much more cost-effective and, thus, more widely used. Moreover, Weinstein *et al.* [104] showed that, for a fixed cost, the use of a constrained library constructions can yield orders of magnitude more promising leads in protein engineering than an unconstrained approach. As the cost of individual synthesis declines, it will become increasingly useful to use our design approach to specify unconstrained libraries that are both diverse and fit.

With this future in mind, we demonstrated the use of our approach for designing such libraries. Specifically, we estimated the optimal trade-off curve for unconstrained libraries using our method (Section A.6), and found that, for the same level of mean predicted fitness, unconstrained library designs are able to achieve greater diversity than the position-wise nucleotide specified designs (Fig. 2.8). Thus, if cost were no concern, it would be advantageous to use individual synthesis construction methods to realize the theoretical unconstrained libraries designed using our machine learning-guided framework.

Experimental validation of designed libraries

We synthesized two designed libraries, D2 and D3, from our optimal trade-off curve (Fig. 2.7a) to assess the accuracy of the designed libraries' trade-off between diversity and mean predicted packaging fitness. The library D2 was selected on the basis of its location near the "elbow" of the trade-off curve, which is suggestive of a good balance between diversity and predicted packaging fitness. The library D3 was selected because, as previously discussed, it dominates the NNK library by achieving much higher mean predicted packaging fitness with a negligible decrease in diversity. After assessing the packaging fitness and diversity of our designed libraries using laboratory experiments, we also test library D2 in a downstream selection task—namely, infection of primary human brain tissue.

After experimentally constructing and deep sequencing libraries D2 and D3, we, first, confirmed that the realized libraries matched the statistics of the theoretical designed library distributions. Indeed, we found that the empirical position-wise probabilities for each amino acid in each of the constructed designed libraries was within 5% of the designed specification (Table A.2, Table A.3). Further, the sequencing data demonstrated that the reduction in diversity between the NNK and designed libraries is relatively small: approximately 2.7 and 4.4 million unique variants were observed in the sequencing data for the D2 and D3 libraries, respectively (Table A.4, Fig. A.5).

Having validated that the constructed libraries were as specified, we packaged and harvested each library using the same methods as for the NNK library (Section A.1), yielding a pre- and post-packaged version of each, and corresponding viral titer measurements. Deep sequencing data for each pre- and post-packaged library confirmed that the designed libraries are substantially different from the standard NNK library (Table A.5). We also found a strong positive Pearson correlation between the mean predicted log-enrichment and experimental titer measurement for each library (Pearson=0.959; Fig. 2.9a).

It is not clear from the trade-off curve alone which of the libraries D2 and D3 is a better general-purpose starting library since each one trades off predicted packaging fitness and diversity differently. To assess this trade off, we subjected each of the D2, D3, and NNK libraries to one round of packaging selection and analyzed the diversity of each post-packaging library using deep sequencing data. When analyzing packaged libraries, the true underlying probability distributions corresponding to each library are not known and thus we cannot exactly compute entropies. Instead, we estimate the effective sample size—specifically, the effective number of variants—of each post-packaging library from the observed sequencing data (Section A.8). Effective sample size is commonly used to estimate phylogenetic diversity [96, 15], the number of non-redundant homologous sequences in multiple sequence alignments [67], cell-type specificity of transcription factor expression [68], and population

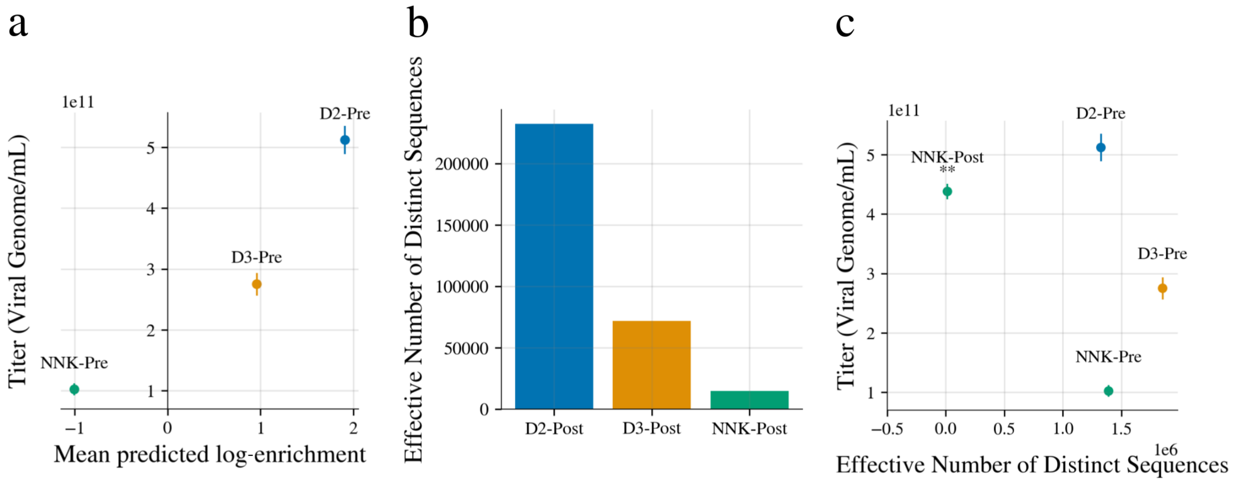


Figure 2.9: Experimental comparison of machine learning-designed libraries D2 and D3 to the NNK library. (a) Experimental packaging titers (viral genome/mL) plotted against the mean predicted log-enrichment (*** $p < 0.001$; two-sided student's t-test). (b) Comparison of the effective number of variants present in each library after packaging. (c) Experimental titers and effective number of variants for D2, D3, and NNK DNA libraries (pre-packaging selection), and the NNK post-packaged library (** $p < 0.01$; two-sided student's t-test). DNA and post-packaged libraries are distinguished using the '-Pre' and '-Post' suffixes, respectively. In all cases, experimental titers are measured on 3 replicates, and the error bars display standard deviations across replicates.

sizes in population genetics [110] because it measures the uniformity of the distribution (*i. e.*, relative abundances) of the unique observations rather than just the number of unique observations. In our context, the effective number of variants for a given library is defined as

$$N_e = \exp\left(\sum_s -p_{\text{empirical}}(s)\log p_{\text{empirical}}(s)\right)$$

where $p_{\text{empirical}}(s)$ corresponds to the empirical read frequency of the variant with sequence s in the corresponding sequencing data. Therefore, for a given library, the effective number of variants reflects not just the number of unique variants, but also relative read frequencies among variants. For example, if 100 sequencing reads are distributed among five unique variants with read counts (25, 25, 25, 20, 5), the estimated effective number of variants is

$$\exp\left(-3\frac{25}{100}\log\frac{25}{100} - \frac{20}{100}\log\frac{20}{100} - \frac{5}{100}\log\frac{5}{100}\right) \approx 4.5,$$

whereas the estimated effective number of variants if the 100 reads are instead distributed as (90, 3, 3, 2, 2) is

$$\exp\left(-\frac{90}{100}\log\frac{90}{100} - 2\frac{3}{100}\log\frac{3}{100} - 2\frac{2}{100}\log\frac{2}{100}\right) \approx 1.6.$$

A larger effective number of variants after packaging indicates that the post-packaging library is less likely to be dominated by a small number of variants, and thus that it contains more unique variants that are able to be packaged efficiently. We were also to confirm empirically that the effective number of variants is not sensitive to the small observed differences in read coverage between libraries in our study (Table A.6, Fig. A.6).

An analysis of the effective number of variants revealed library D2 to be a more promising starting library that D3 (Fig. 2.9b). Moreover, designed library D2 showed a five-fold higher packaging titer (5.12×10^{11} viral genome/mL) than the NNK library (1.02×10^{11} viral genome/mL). We also measured the titer of the post-packaging NNK library ("NNK-Post"), and the resulting titer value (4.38×10^{11} viral genome/mL) was still lower than that of library D2. This suggests that an additional round of packaging selection is not sufficient to lift the NNK library's titer level to that of library D2. Note, also, that (i) the NNK-Post library contained only 1.48×10^4 effective variants compared to the 1.33×10^6 effective variants in library D2, and (ii) the designed libraries differ substantially from the observed position-wise amino acid frequencies in the NNK-Post library (Fig. A.7). Collectively, these experimental results suggest that our machine learning-guided library design procedure yielded a more useful starting library than the NNK library, which is currently the standard peptide insertion library for AAV directed evolution experiments.

Human brain tissue infection using designed library

Recall that our goal was to design a broadly useful starting library, agnostic to the downstream selection goal. As such, having demonstrated our ability to design and construct sufficiently diverse libraries with improved packaging efficiency, we next investigated the extent to which these improvements translate to better performance on downstream selection tasks for which the library has not been tailored. We proceeded to further compare the widely-used NNK library and our designed D2 library by using each to infect primary human brain tissue. Infecting such tissue with AAV is an initial step towards numerous clinical applications in the central nervous system. We specifically focus on human tissue because AAV selection using directed evolution is sensitive to the choice of experimental system—evolved variants display high specificity in the context of cell types, species, and even strains within the same species [47, 29]. Given transcriptional differences between mouse and human cell types in the brain [44, 53] and evolutionary emergence of new cell types in the human brain that are absent in rodents [49], careful selection of the starting biological material and model system are crucial. To make our work relevant for therapeutic interventions in humans, we used fresh, surgically-resected adult cortical tissues from epilepsy patients to select AAV variants that can efficiently infect and drive gene expression in the human-specific context.

We applied approximately 50 μL of each library onto a roughly 300 μm human adult brain slice (Fig. A.8) and harvested the tissues after 72 hours of infection (Fig. 2.10a; Section A.4). We evaluated the success of each library on this task by comparing the effective number of variants in each library after brain infection. Recall that a higher effective number of variants post-brain infection suggests that the starting library contained more variants that were able to successfully infect human brain tissue and is, therefore, indicative of a more useful starting library that yields a larger set of promising variants.

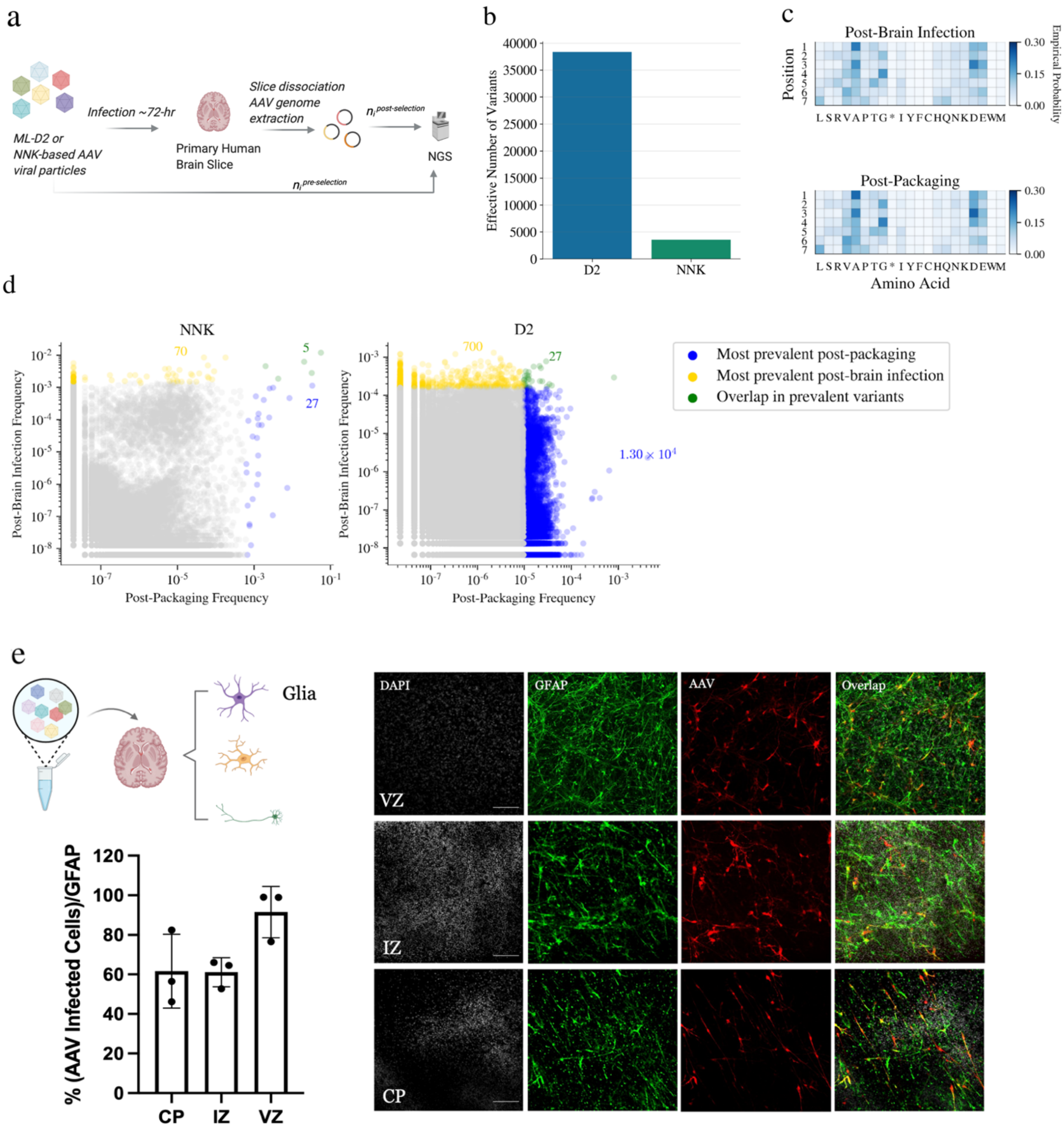


Figure 2.10: (a) Workflow of the brain infection study. (b) Effective number of variants in NNK and D2 post-brain infection libraries. (c) Empirical position-wise probabilities for D2 post-packaging and post-brain infection. (d) Read frequencies of variants after packaging and brain infection. For each library, variants occurring in the top 20% were determined by sorting unique variants by descending read count and identifying top variants accounting for 20% of total reads. Colored variants occur in the top 20% after packaging (blue), brain infection (yellow), or both (green). Colored numbers indicate the number of variants of each color. (e) Validation of variant VVKQRGD from the D2 post-brain infection library. Green represents the Glial Fibrillary acidic protein (GFAP) marker; red represents infected cells (scale bar= 100μm; CP: cortical plate; IZ: intermediate zone; VZ: ventricular zone).

We found that the designed library, D2, had a ten-fold higher effective number of variants post-brain infection than the NNK library (Fig. 2.10b; Fig. A.9)—38,350 versus 3,541 effective variants after the brain infection. Diversity can be achieved in different ways, and we sought to determine whether diversity was spread over the length of the 7-mer insertion sequence or if some positions might have "collapsed" to be more constrained as a result of the selection. Therefore, for each post-packaging and post-brain infection library, we computed the probability of each amino acid at each position in the 7-mer insertion, and found largely uniform distributions over amino acids (Fig. 2.10c; Fig. A.10). This suggests that position-wise diversity was well maintained. We, next, compared the post-packaging and post-brain infection libraries at the level of individual variants to assess some practical implications of the difference in diversity between the NNK and D2 libraries (Fig. 2.10d). We found that a small set of variants dominated the post-packaging NNK library: the 32 most prevalent variants post-packaging (blue and green points in Fig. 2.10e) accounted for 20% of the total sequencing reads. There were roughly 100-fold more unique variants in the top 20% of the post-packaging D2 library (1.32×10^4 blue and green points). This indicates that there is a much larger set of variants that package well in the D2 library and, therefore, using D2 instead of NNK as a starting library increases the chances of discovering a variant that packages and subsequently succeeds in the downstream selection. Indeed, a much smaller set of variants dominated the post-brain infection NNK library: roughly ten-fold fewer unique variants account for the top 20% of the post-brain infection NNK library compared to D2 (75 versus 727 yellow and green points). We also considered the top 50% and 80% of the post-packaging and post-brain infection libraries (Fig. A.11) and found these conclusions to be consistent. Thus, in practice, library D2's higher effective number of variants post-packaging and post-brain infection translates to a much larger set of promising individual variants after each type of selection. Collectively, the results of our brain infection study (Fig. 2.10) demonstrate that our designed library, D2, provided a more useful balance of diversity and packaging ability compared to the widely-used NNK library, thereby making it an effective, general starting library for downstream selections for which it was not specifically designed.

Finally, we validated that individual AAV variants from our designed library, D2, can successfully mediate cell-specific infection, which is a significant open challenge in AAV engineering. For example, glial cells are important regulators for many aspects of human brain function and disease, but AAVs that specifically target glial cells remain elusive [60]. To identify top variants for cell expression validation, we applied the D2 library to prenatal human brain tissues, dissociated and isolated glial cells, extracted the cells' AAV genomes, and applied deep sequencing (Section A.5). We ranked the variants in the post-glia infection D2 library by cLE (computed between the starting and post-glia infection D2 libraries) and selected the top variants for individual validation. Each of these selected AAV variants showed high titer (approximately 10^{12} viral genome/μL) when packaged with a GFP-encoding genome (Table A.7). Moreover, the AAV variant containing insertion VVKQRGD showed high levels of glial infection across multiple regions of the primary human brain tissue in immunostaining (Fig. 2.10e). This provides further evidence that our designed library makes an effective starting point for biologically-interesting and clinically-relevant downstream selections. Moreover, it will be straightforward to extend our library design approach to cell types in the brain or other tissues for a variety of therapeutic applications in future work.

2.6 Conclusion

In this chapter, we provided an end-to-end set of machine learning-guided library design solutions and validated them through laboratory experiments in a therapeutically-relevant system. Specifically, we (i) presented a predictive modeling approach for learning regression models that can predict a property of interest based on sequencing data produced by a high-throughput selection experiment, (ii) introduced a framework for machine learning-guided library design that leverages a trained predictive model to design libraries that optimally trade off diversity and the predicted property, and (iii) demonstrated the use of these methods in an end-to-end experimental and computational pipeline for machine learning-guided design of a library of insertion sequences into the AAV capsid. In accomplishing the latter, we showed that we can build predictive models for packaging fitness for AAV peptide insertion libraries that are sufficiently accurate to guide the design of libraries with improved packaging fitness. We demonstrated that the resulting designed library makes a better starting library for downstream brain infection selections than the current standard AAV insertion library. It is our hope that future work will further generalize this approach to additional downstream selection tasks for AAV, such as evasion of pre-existing neutralizing antibodies.

The machine learning-guided library design approach presented in this chapter can, in principle, be used for other library construction techniques and extended to design libraries with multiple desired properties—beyond diversity—by replacing the predictive model with one trained to simultaneously predict multiple properties or fitnesses, as we shall see in Chapter 3. Such an approach could be particularly useful for designing libraries of AAV capsids with improved specificity across multiple cell types, which is particularly challenging using conventional experimental approaches. However, predictive accuracy is critical to the success of model-guided library design. Thus, when one considers applying our design approach in more complicated settings, important concerns arise regarding the use of cLE for predictive modeling. In Chapter 3, we will highlight key limitations of count-based log-enrichment estimates which drastically reduce their utility in many practical settings of interest and limit the accuracy of predictive modeling approaches based on these estimates. We will, then, present an alternative predictive modeling approach—which leverages ideas from the rich density ratio estimation literature within machine learning—to address these limitations, thereby enabling effective analysis across a broader range of common experimental setups that can currently be achieved.

Chapter 3

A New Model-Based Enrichment Approach for High-Throughput Sequencing Experiments

This chapter contains material reproduced with permission from:
Akosua Busia and Jennifer Listgarten. "Model-based differential sequencing analysis". In: *bioRxiv* (2023)

3.1 Introduction

In Chapter 2, high-throughput sequencing enabled characterization of large diversified libraries of adeno-associated virus (AAV) capsid proteins. More generally, we can now assay up to billions of DNA or RNA sequences in parallel for an ever-expanding set of properties of interest using next-generation sequencing [45, 54, 109]. As a consequence, high-throughput sequencing has dramatically changed the landscape of biological discovery—both for basic scientific inquiry into protein behavior and evolution [109, 79], and in application areas spanning human disease and variant detection [79, 107], engineering anti-viral immunogens and therapeutics [109, 61, 62], drug and antibiotic resistance [79, 109], regulatory element engineering in synthetic biology [66] and beyond.

Recall that, across many of these scientific areas, a key desired outcome from a high-throughput sequencing experiment is to quantify the change in relative abundance between two conditions for a large number of distinct sequences, and this type of quantification is often referred to as estimating the log-enrichment of each sequence between conditions [23, 39, 79, 54, 61, 62, 63]. For example, log-enrichment estimation is performed in differential analyses of RNA-seq and ATAC-seq experiments [94, 76, 112, 48], deep mutational scanning [9, 63, 23, 54, 79, 109], functional enrichment analysis [108], and high-throughput selection experiments [46, 35, 72, 12, 61, 62, 116] like those in Chapter 2 to assess the packaging efficiency and infectivity of viral vectors. Such selection experiments have many other wide-ranging biologically-significant applications, including: antibody design and affinity maturation [28, 34]; profiling pathogen proteomes for epitopes and major histocompatibility complex binding [35, 72]; improving thermostability [78]; and assessing binding [46, 63, 111] and catalytic activity [78, 80]. In general, by accurately estimating log-enrichment for a large sequence library, one can identify sequences that are more likely to have desired properties and potentially gain insights into the sequence determinants of a property of interest. Moreover, log-enrichment estimates computed from observed sequencing counts are increasingly being used as supervised labels for training machine learning models that can predict log-enrichment for previously unobserved sequences [12, 46, 70, 111, 116, 86, 82, 25], often more accurately than popular physics-based and unsupervised machine learning methods [25].

Limitations of log-enrichment estimates

Although standard count-based log-enrichment (cLE) estimates have proven incredibly useful, they suffer from an important shortcoming: the inability to share information across non-identical reads. This inability causes a loss of important available information in a number of practical settings, including:

1. *Short reads*: when short, possibly overlapping reads are available that do not individually cover the entire *sequence of interest*—*i. e.,* the entire span of sequence which we would like to quantify.

2. *Sparse reads*: when few sequencing reads are available per library sequence, as is especially common with long-read sequencing [84, 107, 38, 74].

3. *Hybrid reads*: when a combination of long and short reads are available.

4. *Negative selection*: when the goal is to discover sequences enriched in a property that is opposite from the selection.

5. *More than two conditions*: when we seek to characterize sequences across multiple conditions/selections, such as might occur when engineering viral gene therapy vectors to selectively infect one cell type but not another.

It is well-known that cLE estimates suffer from high variance when sequencing counts are low [39, 116, 79] (*i. e.,* in the sparse reads and negative selection settings). Previous efforts to mitigate variance induced by low counts use regression to "de-noise" cLE estimates by either using a model to intelligently aggregate data across iterative selection rounds [79], or by downweighting examples with low counts (Section 2.3). While these techniques can improve analyses, they remain extremely limited in their ability to share information across non-identical reads. As a simple example, if only 10 out of 300 positions in a sequence of interest are predictive of the property of interest, better statistical power can be achieved by calculating cLE using counts defined by only the 10 relevant positions than by counting based on all 300, since the latter will cause most reads to appear to be non-identical and hence treated separately. A method that can automatically learn to share information appropriately would, therefore, be of high value.

This idea of information sharing can be generalized beyond the sparse read and negative selection settings. Ideally, to accurately estimate log-enrichment (LE), one would prefer sequencing data with high read coverage that is comprised of reads that each individually cover the full sequence of interest. However, in practice, individual reads often do not cover the entire sequence of interest—this typically arises with short-read sequencing, but could also occur when using long-read technologies to analyze large sequences of interest [74] or in the hybrid read setting. In these settings, it is not obvious how to count reads for the sequence of interest, nor how to calculate the desired cLE estimates. To tackle the LE estimation problem nonetheless, one might consider estimating read-level cLE estimates and then devising heuristics to combine them to produce a LE estimate for the full sequence of interest. However, such an approach is unlikely to account for correlations across reads (*e. g.,* linkage disequilibrium) nor partial overlap between reads. Moreover, it is not clear which out of the abundance of possible heuristics to use, and the answer is likely application-dependent. For example, in applications where there is a known reference sequence—such as in many RNA-seq and ATAC-seq experiments—the reference can help provide information about how to combine reads [94]. However, this is typically accomplished by performing alignment and assembly prior to standard cLE estimation, and thus such approaches still suffer from many of the limitations just described. Devising an alternative approach to LE estimation—one that is capable of "sewing" together partially overlapping reads and sharing across non-identical reads—would enable more efficient sharing of information.

A new approach for log-enrichment estimation

Ultimately, a method that can automatically learn to share information as appropriate across non-identical reads will improve our ability to extract important information from sequencing data in a broad range of settings, thereby providing higher statistical power given the same type and amount of sequencing data. In this chapter, we shall see that reframing the LE estimation problem as what is known as *density ratio estimation* (DRE) in machine learning [90] enables us to develop just such a method. By leveraging a machine learning approach that learns directly from sequencing reads without the need to pre-compute cLE estimates as supervised labels, we make progress on overcoming the shortcomings of existing approaches based on cLE within one cohesive framework, thereby improving performance in each of the four previously enumerated settings.

In Sections 3.2–3.6, we will: (i) review how LE estimates are currently computed; (ii) describe our proposed machine learning approach, *model-based enrichment* (MBE); (iii) provide a detailed empirical characterization of MBE using data from simulated high-throughput selection experiments; and (iv) do the same using data from real selection experiments. Overall, we demonstrate empirically that MBE enables effective analysis across a broader range of common experimental setups than can currently be achieved. While our primary motivation is to improve *predictions* of LE on new, unobserved sequences—as this is crucial to the success of our machine learning-guided library design approach from Chapter 2—our results show that MBE also enables better *estimation* of LE for observed sequences, the more classical use case. In particular, we will demonstrate that, compared to existing approaches based on cLE, MBE produces predictions that correlate better with ground truth fitness across a broad range of high-throughput selection datasets, in part due to its improved robustness to low sequencing counts. We also show that MBE enables better characterization of sequences from negative selection experiments and is, consequently, better at identifying *selective* sequences that are high in one property and simultaneously low in another—such as one might seek in studies of AAV tropism [62, 69] where the ideal viral vector selectively infects one cell type, but not others.

3.2 Current log-enrichment estimation approaches

Here, we review existing count-based LE estimation and regression approaches before describing our proposed MBE approach in Section 3.3.

Currently, count-based approaches to estimating LE (1) compute a cLE estimate for each sequence from observed sequencing counts, and, optionally, (2) train a supervised machine learning model using these observed cLE estimates as labels. To achieve (1), one subjects two sequence libraries—one for each of two conditions A and B—to high-throughput sequencing, yielding a dataset

$$\mathcal{D} = \{(r_i, y_i)\}_{i=1}^{M} \tag{3.1}$$

where r_i is the i^{th} read's sequence and y_i is a binary $-1/+1$ label indicating whether the read r_i arose from condition A or B. For high-throughput selection experiments, the conditions A and B correspond to pre- and post-selection, however, the methodology in this chapter can be applied broadly to settings with sequencing data from two conditions for which we seek to understand or predict sequence properties. In Section B.1, we also further generalize to more than two conditions.

Next, one calculates a cLE estimate [23, 39, 54, 79, 109] for each unique sequence from these data, $\mathcal{D}$, which serves as a quantitative estimate of the extent to which the sequence has the property being investigated. Recall from Chapter 2 that, for high-throughput selection experiments, the cLE estimate serves as a proxy for the fitness that drives the selection process. To compute cLE estimates, it is convenient to represent $\mathcal{D}$ in terms of unique sequences: $\mathcal{D}' = \{(x_i, n_i^A, n_i^B)\}_{i=1}^{M'}$ where $\{x_i\}_{i=1}^{M'} \subseteq \{r_i\}_{i=1}^{M}$ is the set of unique observed sequences,

$$n_i^A = \sum_{(r,y)\in\mathcal{D}} \mathbb{1}\{r = x_i\}\mathbb{1}\{y = -1\} \tag{3.2}$$

is the observed read count for sequence x_i in the sequencing data for condition A, and

$$n_i^B = \sum_{(r,y)\in\mathcal{D}} \mathbb{1}\{r = x_i\}\mathbb{1}\{y = +1\} \tag{3.3}$$

is the corresponding condition B read count. For each sequence, the cLE estimate is equal to the log-ratio of read frequencies for conditions A and B,

$$\log e_i = \log\left(\left(\frac{n_i^B}{N^B}\right)\left(\frac{n_i^A}{N^A}\right)^{-1}\right), \tag{3.4}$$

where $N^A = \sum_{i=1}^{M'} n_i^A$ and $N^B = \sum_{i=1}^{M'} n_i^B$. In practice, it is common to add a small constant to each count prior to calculating cLE estimates for mathematical convenience [54, 79]. These "pseudo-counts" stabilize the cLE estimates, and allow one to avoid division by zero for sequences observed in only one condition. In our experiments in Sections 3.5–3.6, we added a pseudo-count of 1 to each raw count.

For (2), LE regression approaches fit a model that maps from x_i to $\log e_i$. In particular, in Chapter 2, we derive a weighted log-enrichment regression (wLER) approach for fitting such a model and show that it is beneficial compared to standard unweighted supervised regression. Our wLER procedure assigns a weight, $w_i = (2\sigma_i^2)^{-1}$, to each sequence, where

$$\sigma_i^2 = \frac{1}{n_i^B}\left(1 - \frac{n_i^B}{N^B}\right) + \frac{1}{n_i^A}\left(1 - \frac{n_i^A}{N^A}\right). \tag{3.5}$$

This choice of w_i is motivated by a convergence argument: σ_i^2 is the asymptotic variance of $\log e_i$ [39, 116]. Recall that when the counts n_i^A and n_i^B are low, $\log e_i$ is a noisier estimate of fitness and the corresponding weight, w_i, is smaller. Thus, training a model using wLER accounts for the heteroscedastic noise in the observed cLE estimates. In Section 3.3, we describe our MBE method which, in contrast to wLER, does not require explicit derivation of a weighted loss; MBE naturally accounts for different levels of evidence arising from higher or lower counts.

3.3 A new approach: model-based enrichment

In Section 3.2, we saw that regression-based LE estimation (or prediction) is performed in two sequential steps:

1. compute a cLE estimate for each unique sequence [23, 39, 79, 54], and
2. train a regression model to predict these cLE estimates from the observed sequences, possibly weighting each sequence to account for its corresponding level of evidence [25, 70, 111, 116].

We introduce a new method, MBE, that performs both of these steps at once, resulting in a more powerful and more general analysis framework. We do so by reframing the LE estimation problem: in this section, we show that a cLE estimate can be viewed as an approximation to the logarithm of what is known as a *density ratio*—a ratio of probability densities of the observed sequence under each condition (Fig. 3.1). Therefore, we can estimate

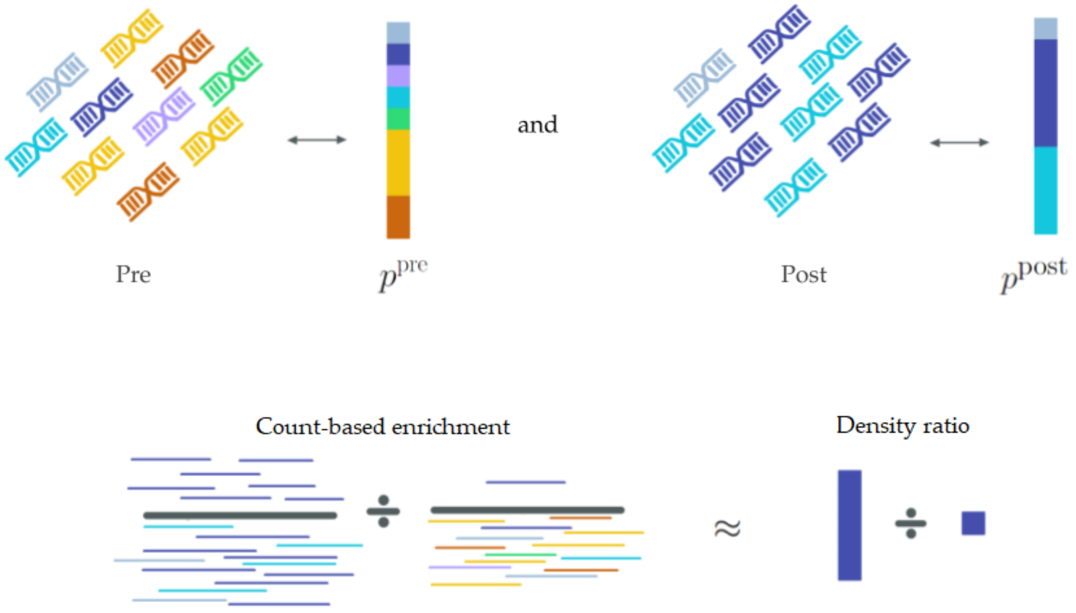


Figure 3.1: Visualization of the relationship between count-based enrichment and density ratios. (top) Each library can be viewed as a probability distribution over sequence space, and, therefore, (bottom) count-based enrichment scores computed based on observed sequencing reads can be viewed as a sample-based approximation of the underlying density ratio between library distributions.

and predict LE by solving a DRE problem. Further, DRE can be effectively and accurately performed by training a probabilistic classifier to predict which of the two densities a sample came from (*e. g.,* condition A or B) [27, 71, 91, 90, 31, 64, 56]. Specifically, the ratio of such a classifier's predicted class probabilities provably converges to the density ratio [71, 91, 90]. Through this series of theoretically-justified steps, we are able to transform the problem of estimating LE into one of training a read-level classifier to distinguish the condition from which a read came. Our classifier-based DRE approach differs substantially from several recent approaches that also make use of classification: in one, cLE estimates are thresholded and a classifier built to predict the resulting binarized labels (*e. g.,* [12]), and in another, a classifier is built to predict whether a sequence appeared at all in post-selection sequencing data (*e. g.,* [86]). Neither of these approaches address the shortcomings that we seek to resolve with MBE.

Recall from Section 3.2 that the cLE estimate, $\log e_i$, for a given sequence, x_i, is the log-ratio of the two normalized counts $\frac{n_i^A}{N^A}$ and $\frac{n_i^B}{N^B}$ (Eq. 3.4). These normalized counts are also the observed empirical frequencies of x_i in the sequencing data for conditions A and B, respectively. In particular, these two ratios are the sample-based estimates of the true population frequencies of x_i in each library, which we denote $p^A(x_i)$ and $p^B(x_i)$, where p^A and p^B are each a discrete probability distribution over sequence space (Fig. 2.2). Consequently,

$\log e_i$ can be viewed as a sample-based estimate of the population-level LE, which we denote $\log d(x_i)$. Specifically,

$$\log e_i \approx \log d(x_i) = \log \frac{p^B(x_i)}{p^A(x_i)}. \tag{3.6}$$

where d is the density ratio between the library distributions. By training a binary classifier to predict the probability that a read with sequence x_i came from the library in condition B, we can estimate $\log d(x_i)$, and hence the LE, using the log-ratio of the classifier's predicted class probabilities. It has been proven theoretically that under a correctly specified model, this classification-based density ratio estimation method is optimal among a broad class of semi-parametric estimators—that includes the wLER method—in terms of asymptotic variance [71, 90] (Section B.2).

In more detail, in our MBE approach, we train a probabilistic classifier, g_θ with learnable parameters θ, on the data $\mathcal{D}$ (Eq. 3.1) to predict y_i from r_i by minimizing the standard logistic loss,

$$\ell_{\mathrm{C}} = \sum_{i=1}^{M} \log(1 + \exp(-y_i g_\theta(r_i))). \tag{3.7}$$

This produces a model of $p(y \mid r)$ and, by Bayes' theorem, of the density ratio [71, 90]:

$$\begin{align} d(x) = \frac{p^B(x)}{p^A(x)} &= \left(\frac{p(y=+1 \mid r=x)p(x)}{p(y=+1)}\right)\left(\frac{p(y=-1 \mid r=x)p(x)}{p(y=-1)}\right)^{-1} \tag{3.8} \\ &= \frac{p(y=+1 \mid r=x)}{p(y=-1 \mid r=x)}\frac{p(y=-1)}{p(y=+1)} \approx \frac{N^A g_\theta(x)}{N^B(1-g_\theta(x))}, \tag{3.9} \end{align}$$

where N^A and N^B are the total read counts for each condition (as in Eq. 3.4).

MBE naturally accounts for heteroscedastic noise in the observed sequencing data. To see this, we can rewrite ℓ_{C} in terms of unique sequences,

$$\ell_{\mathrm{C}} = \sum_{i=1}^{M'} n_i^B \log(1 + \exp(-g_\theta(x_i))) + n_i^A \log(1 + \exp(g_\theta(x_i)), \tag{3.10}$$

where n_i^A and n_i^B are read counts (Eq. 3.2–3.3). This form of ℓ_{C} highlights the fact that sequences with higher counts make larger contributions to the loss than those with lower counts, simply by virtue of having been sequenced many times. Thus, g_θ is biased towards modeling d more accurately for sequences with more sequencing data, as desired. In this way, the MBE approach accounts for heteroscedasticity in the observed sequencing data without the need to derive a bespoke weighted loss function, unlike the wLER approach.

This MBE approach has several important practical advantages, including:

1. the underlying classifier is trained directly on reads and can automatically learn how to share information across reads;

2. there is no need to explicitly compute sample weights for each unique sequence to account for variance arising from low sequencing counts, as this is implicitly learned by the classification model;

3. it can be easily implemented using standard software packages for learning classifiers, and allows one to leverage standard techniques for supervised learning problems such as cross-validation;

4. it can leverage existing model architectures, including neural networks that have been previously established to work well for LE regression [25] and those that can handle variable-length inputs to rationally accommodate short- and hybrid-read settings;

5. it easily and efficiently generalizes to settings with more than two conditions of interest by replacing the binary classifier with a multi-class classifier (Section B.1).

The latter advantage enables us to naturally handle experiments with multiple rounds of selection or properties of interest.

3.4 Experimental setup for empirical comparisons

In this section, we describe the simulated and experimental datasets used to empirically compare and contrast our MBE approach with cLE and wLER across a broad range of settings. Then, we provide an overview of the modeling details and evaluation methods.

Simulated high-throughput selection data

Using simulated high-throughput selection experiments, we sought to understand the strengths and weaknesses of the MBE and wLER approaches as we changed following simulation settings:

1. the length of the sequence of interest, L, ranging from 21–2,253 nucleotides;

2. whether short or long reads were used (300 *vs.* 10,000 nucleotides);

3. the number of unique sequences in the theoretical pre- and post-selection libraries, M', ranging from 8.5×10^6–2.6×10^7;

4. the number of pre- and post-selection reads, N^{pre} and N^{post}—always set equal to each other—ranging from 4.6×10^3–4.6×10^7; and

5. the complexity of the functional mapping between sequence and property of interest. This complexity was characterized in terms of a summary parameter controlling the amount of epistasis, T (Section B.3).

Table 3.1: Summary of simulated datasets. For each dataset we list the: library name (Library), sequence length in nucleotides (L), number of unique library sequences (M'), epistasis hyperparameter used for fitness simulation (T), read type (short, long, or hybrid), % of the sequence of interest covered by individual reads (Cover), and number of pre-selection and post-selection reads (N^{pre} and N^{post}), which were always equal. We simulate 4.6×10^7 short reads to match the experimental data from Zhu *et al.* [116], and up to 4.6×10^5 long reads to be within the current throughput of PacBio's technologies [38, 74]. Each dataset is described in more detail in Section B.5.

Library	L	M'	T	Read Type	Cover	$N^{\text{pre}} = N^{\text{post}}$
21-mer insertion	21	8.5×10^6	140	Short	100	4.6×10^7
150-mer insertion	150	8.5×10^6	1000	Short	100	4.6×10^7
300-mer insertion	300	8.5×10^6	2000	Short	100	4.6×10^7
avGFP mutagenesis	714	2.5×10^7	4760	Long	100	4.6×10^5
avGFP mutagenesis	714	2.5×10^7	4760	Short	42	4.6×10^7
AAV recombination	2253	2.6×10^7	15020	Long	100	4.6×10^5
AAV recombination	2253	2.6×10^7	15020	Long	100	4.6×10^4
AAV recombination	2253	2.6×10^7	15020	Long	100	4.6×10^3
AAV recombination	2253	2.6×10^7	15020	Short	13	4.6×10^7
AAV recombination	2253	2.6×10^7	15020	Hybrid	100 long + 13 short	4.6×10^3 long + 4.5×10^7 short

We simulated libraries that correspond to three types of experimental library constructions:

(a) *Insertion* of a fixed-length sequence into a background sequence at a fixed position. This library construction is motivated by our work in AAV capsid engineering in Chapter 2, which aims to understand sequence determinants of AAV properties such as packaging using libraries of 21-mer nucleotide insertion sequences into the capsid. In this Chapter, we simulate insertion libraries with varying lengths (21, 150, and 300 nucleotides). The pre-selection library is generated to be roughly uniform in nucleotide space (technically, the NNK degenerate codon distribution).

(b) *Random mutagenesis*—motivated by a study to understand the fitness landscape of a green fluorescent protein of length 714 nucleotides [82]. Herein, we mutagenise the green fluorescent protein across all positions using a 10% mutation rate to generate the pre-selection library.

(c) *Recombination*—motivated by an AAV directed evolution study [62], wherein several AAV serotypes are recombined using seven crossovers separating eight recombination blocks. We generate library sequences by recombining nine AAV serotypes using eight equally-sized blocks. The total length of all eight blocks is 2253 nucleotides.

A summary of the simulated sequencing datasets is provided in Table 3.1.

To simulate selection, we must simulate the ground truth fitness function for each of these libraries. We did so as a linear function all independent amino acid sites, and T higher-order epistatic features drawn randomly from all possible such effects in a manner that re-capitulates the distribution of these effects in a real protein fitness landscape (Section B.3). Combining insights from several recent works [6, 98, 11], we assumed that T scaled linearly with the length of the sequence of interest with a fixed coefficient based on Poelwijk *et al.* [70].

Next, we simulated reads from the pre- and post-selection libraries as follows (Section B.4): first, we generate library sequences using one of the three previously described library construction simulations. Then, we randomly perturb the empirical distribution of the library sequences (which simulates slight distributional perturbations that may occur with PCR amplification) to generate a pre-selection probability distribution. Next, the corresponding post-selection probability distribution is determined by scaling the pre-selection distribution according to the simulated fitness of the library sequences. Finally, we sample reads that cover the full sequence of interest from the pre- and post-selection distributions. When simulating short reads, we randomly truncate each of these reads to 300 nucleotides.

We also perform negative selection simulations, which were motivated by experiments wherein one seeks to identify sequences with a property, such as low-binding affinity, for which the only available assay enriches for the opposite, such as high-binding. We, therefore, aimed to estimate the accuracy of wLER and MBE to negatively select against an undesirable fitness and, moreover, to identify sequences of interest that are selective—meaning that they are simultaneously high in one fitness (the *positive fitness*) and low in a second (the *negative fitness*). To do so, we simulated two independent fitness functions and used each, separately, on the same pre-selection library to simulate two post-selection libraries and corresponding reads.

Although most of our simulations did not include sequencing errors, we constructed versions of two of the aforementioned datasets that did (Section B.5). For one of the insertion datasets, we used a uniform random substitution error rate of 1%, consistent with observed error rates of Illumina's next-generation sequencers [24]. For one of the recombination datasets, we used SimLoRD [89] to simulate PacBio SMRT sequencing errors. As shall be seen, the sequencing noise had little effect on our results.

Experimental high-throughput selection data

We used five experimental datasets—each comprised of sequencing data from a pre-selection library and after one or more selections on that library—to compare the MBE and wLER approaches. For our evaluations, we also used low-throughput experimental property measurements corresponding to the selected property for each of the five sequencing datasets. Each of the following experimental datasets and its corresponding property measurements are summarized in Table 3.2:

1. A library of 21-mer nucleotide insertions into a fixed AAV background sequence subjected to a round of packaging selection, and packaging titer measurements for five sequences not present in the library [116] (Chapter 2).

2. A library containing every 15 amino acid peptide in the SARS-CoV-2 proteome (which has 14,439 amino acids) subjected to four rounds of selection for binding to human major histocompatibility complex (MHC). For ground truth, there are IC_{50} measurements for 24 peptides held out from the LE analysis [35].

3. A site saturation mutagenesis library containing all single and double amino acid mutations within the 168 nucleotide IgG-binding domain of protein G (GB1) subjected to selection for binding to IgG-FC. For ground truth, the are $\Delta\ln(K_A)$ measurements for 11 individual variants held out from the sequencing data [63].

4. A library containing natural chorismate mutase homologs and designed sequences sampled from a direct coupling analysis model. All sequences are of length 288 nucleotides. For ground truth there are biochemical measurements for 11 variants held out from the sequencing data [80].

5. A β-glucosidase enzyme (Bgl3) error-prone PCR random mutagenesis library subjected to a heat challenge and high-throughput droplet-based microfluidic screening. All sequences are of length 1506 nucleotides. For ground truth, there are T_{50} (temperature where half of the protein is inactivated in ten minutes) measurements for six mutants held out from the sequencing data [78].

Table 3.2: Summary of experimental datasets. For each dataset we list the: library description (Library); sequence length in nucleotides (L); number of unique library sequences after holding out experimentally-validated variants, if needed (M'); number of experimentally-validated variants (n); % of the sequence of interest covered by individual reads (Cover); number of pre-selection reads (N^{pre}); and number of post-selection reads (N^{post}). For the dataset from Huisman *et al.* [35], the number of reads for each round of selection is presented on a separate row.

Library	L	M'	n	Cover	N^{pre}	N^{post}
AAV5 insertion [116]	21	8,552,729	5	100	46,049,235	45,306,265
SARS-CoV-2-derived peptide [35]	45	167,841	24	100	44,073	88,032
SARS-CoV-2-derived peptide [35]	45	167,841	24	100	88,032	169,730
SARS-CoV-2-derived peptide [35]	45	167,841	24	100	169,730	235,787
SARS-CoV-2-derived peptide [35]	45	167,841	24	100	235,787	160,863
GB1 double site saturation [63]	168	536,953	11	100	324,434,913	262,112,210
Chorismate mutase homolog [80]	288	3,063	11	100	1,228,687	1,929,212
Bgl3 random mutagenesis [78]	1506	468,194	6	100	1,177,842	710,555

Model architectures and training

We implemented wLER and MBE using several model architectures. To enable direct comparison of the two methods, we kept the set of architectures and allowed hyper-parameters the same for both approaches, excluding the final layer and loss which dictate whether the model is for regression (wLER) or classification (MBE). Specifically, we used eleven different model architectures: the seven architectures from Section 2.5—three linear models and four fully-connected neural networks (NNs)—and four additional convolutional neural network (CNN) architectures that operate on variable-length sequences, allowing us to train on short reads and make predictions on the full-length sequences of interest. Recall from Section 2.5 that the linear models each use one of three input representations: (1) an "independent site" (IS) representation, (2) a "neighbor" representation, and (3) a "pairwise" representation. All NNs use IS input features and have two hidden layers, and differ by the number of hidden units: 100, 200, 500, or 1000 units per layer. The CNNs differ in the number of convolutional layers used (2, 4, 8, or 16), but all use IS input features, convolutions with a window of size 5 and 100 filters, residual and skip connections, and a global max pooling layer as the penultimate layer. As the linear and NN architectures and hyper-parameters are from our study that used wLER, to the extent the selected architectures may favor one of the two approaches, they would favor wLER.

Several of our experiments simulate negative selection against an undersirable fitness, and selectivity experiments that select for sequences that are simultaneously high in a desirable positive fitness and low in an undesirable negative fitness. For simplicity, in these experiments we used only one model architecture—the smallest NN architecture—as a two-output model, one for the positive fitness and one for the negative (Section B.1). We used this architecture because it was the simplest non-linear model architecture we explored—meaning it is capable of capturing higher-order epistasis whilst being relatively parsimonious. Based on the results of our initial simulation experiments, this choice of architecture does not systematically benefit either of the wLER or MBE approaches (Fig. B.1).

For all real experimental datasets (except for the Bgl3 dataset), we similarly used the smallest NN architecture because it tended to achieve better cross-validation performance than the linear architectures, and comparable performance to the larger NN and CNN architectures whilst being more parsimonious (Fig. B.8a-l). For the Bgl3 dataset, we used a simpler linear model because overfitting was observed with the NNs (Extended Data Fig. B.8m-o). For the one dataset that had multiple rounds (Huisman *et al.* [35]), we used a multi-output model with one output per round and took the final prediction to be the average of the predictions for each round.

All models were trained using the AMSGrad Adam optimizer [73] with default learning rate (10^{-3}) for ten epochs. For the linear models and NNs, we used the default value for Adam's ϵ parameter (10^{-7}); for the CNNs, we set $\epsilon = 1^{-4}$ and applied gradient clipping with a threshold of 1 to stabilize training.

Evaluation methods

Using both the simulated and experimental datasets, we compared three approaches, as appropriate: standard cLE, wLER, and MBE. wLER and MBE can be used to both (i) make predictions on sequences not observed in the training data, and (ii) make predictions on sequences in the training data itself to yield LE estimates—which can be thought of as "de-noising" cLE estimates. We refer to these two tasks, respectively, as *prediction* and *estimation.* cLE can only be used for estimation.

To compare wLER to MBE on any given dataset, we used all model architectures and hyper-parameters for both methods, and then selected the best combination for each of wLER and MBE separately. No model or hyper-parameter selection is required for cLE since it does not use any model or have any parameters.

An important point to appreciate throughout is that we cannot use straightforward cross-validation to assess the accuracy of each method because we cannot use ground truth fitness values during training, but rather only to evaluate performance. We also cannot use, say, cLE estimates for standard cross-validation, as these are not ground truth values. Hence, in simulated settings, we perform slightly modified cross-validation where we use only sequencing data to train, and evaluate performance on each fold by comparing predictions to the held-out sequences' ground truth fitness values. For the real experimental datasets where ground truth fitness values for the library sequences are unknown, we use available low-throughput (non-sequencing-based) experimental fitness measurements for validation. These low-throughput measurements may still be corrupted by noise, but are more direct measurements of the property of interest than the sequencing-based assays.

In our simulations, we use three-fold cross-validation to compute the Spearman correlation between ground truth fitness and predicted LE to compare the performance of each method. Additionally, we use a generalized Spearman correlation that focuses on sequences that have the highest ground truth fitness—the focusing is controlled by a threshold on true fitness, which we sweep through a range of values, such that at one extreme, we compute the Spearman of all sequences in the test set, and on the other, of only the most truly fit sequences (as in Section 2.5). The test fold is always comprised of full sequences of interest, even when the training data contained reads that were shorter. We averaged the Spearman correlations computed on each fold to produce one cross-validated correlation value, and we use William's t-test [87] to assess statistical significance of the difference between the cross-validated Spearman correlations.

Each selectivity simulation is defined by two different simulated fitnesses, a positive fitness and negative fitness. For the positive fitness, we use the generalized Spearman correlation described above to evaluate predictive performance. For the negative fitness, we use a similar generalized Spearman correlation that focuses on sequences with *lowest*—instead of highest—ground truth fitness. We also define the selectivity of a sequence as the difference between its positive and negative fitness values, and compare how well wLER and MBE identify test sequences with high selectivity. To do so, for each method, we (i) rank the sequences in each test fold according to predicted selectivity—the difference between predictions for each fitness—and take the top ten, and then (ii) compare the two ground truth fitness values of each of the chosen sequences to the fitness values of a theoretical optimally-selective sequence that has the maximum true positive fitness and minimum true negative fitness observed in the given dataset. We use McNemar's test [55] to assess the statistical significance of the difference between the methods' accuracy at identifying the 1% of test sequences with highest selectivity.

Using the real experimental data, we compare the wLER and MBE approaches by computing Spearman correlation between predicted LE and low-throughput experimental property measurements. We use a paired t-test to assess statistical significance of the performance difference between wLER and MBE aggregated across all five experimental datasets.

3.5 Results on simulated high-throughput selection data

Across all simulated datasets, our MBE approach made significantly more accurate LE predictions than wLER (Fig. 3.2a) according to standard Spearman correlation ($p < 10^{-10}$). The improvements of MBE over wLER in terms of Spearman correlation values ranged from 0.005 to 0.561, with an average of 0.177. In no cases did MBE perform worse than wLER. We also found that our MBE method performed better when faced with both Illumina- and PacBio-like sequencing error (Fig. 3.2, Fig. B.5). In addition, MBE was less sensitive to the choice of model architecture, to such an extent that even the worst-performing MBE model performed better than the best-performing wLER model on several datasets (Fig. B.1a). Similarly, for the estimation task, MBE outperformed wLER across all simulated datasets (Fig. 3.2b, Fig. B.1b). Collectively, these results demonstrate a clear win for MBE over wLER across a broad range of settings. In the subsequent sections, we examine each of the following settings in more detail for a more comprehensive view of the strengths and weaknesses of each method: sparse reads, short reads, hybrid long and short reads, negative selection, and selection for sequence selectivity.

Sparse read setting

We define the sparse read setting as occurring when the average number of sequencing reads per library sequence was lower than 0.02. In our experiments, this includes the simulated long-read datasets for the avGFP mutagenesis and AAV recombination libraries. We hypothesized that the MBE approach would have a particular advantage in this setting because of its improved ability to combine information across similar but non-identical reads. Indeed, on the prediction task, MBE maintains comparable accuracy to wLER on test sequences with high ground truth fitness, while improving accuracy in the other regimes (Fig. B.2a-b, Fig. B.3). Additionally, MBE had lower variance than wLER across the different test folds (Fig. B.3). We also note that the longer the sequence of interest, the more MBE outperforms wLER—this nicely matches our intuition as the longer the read, the more sparse the

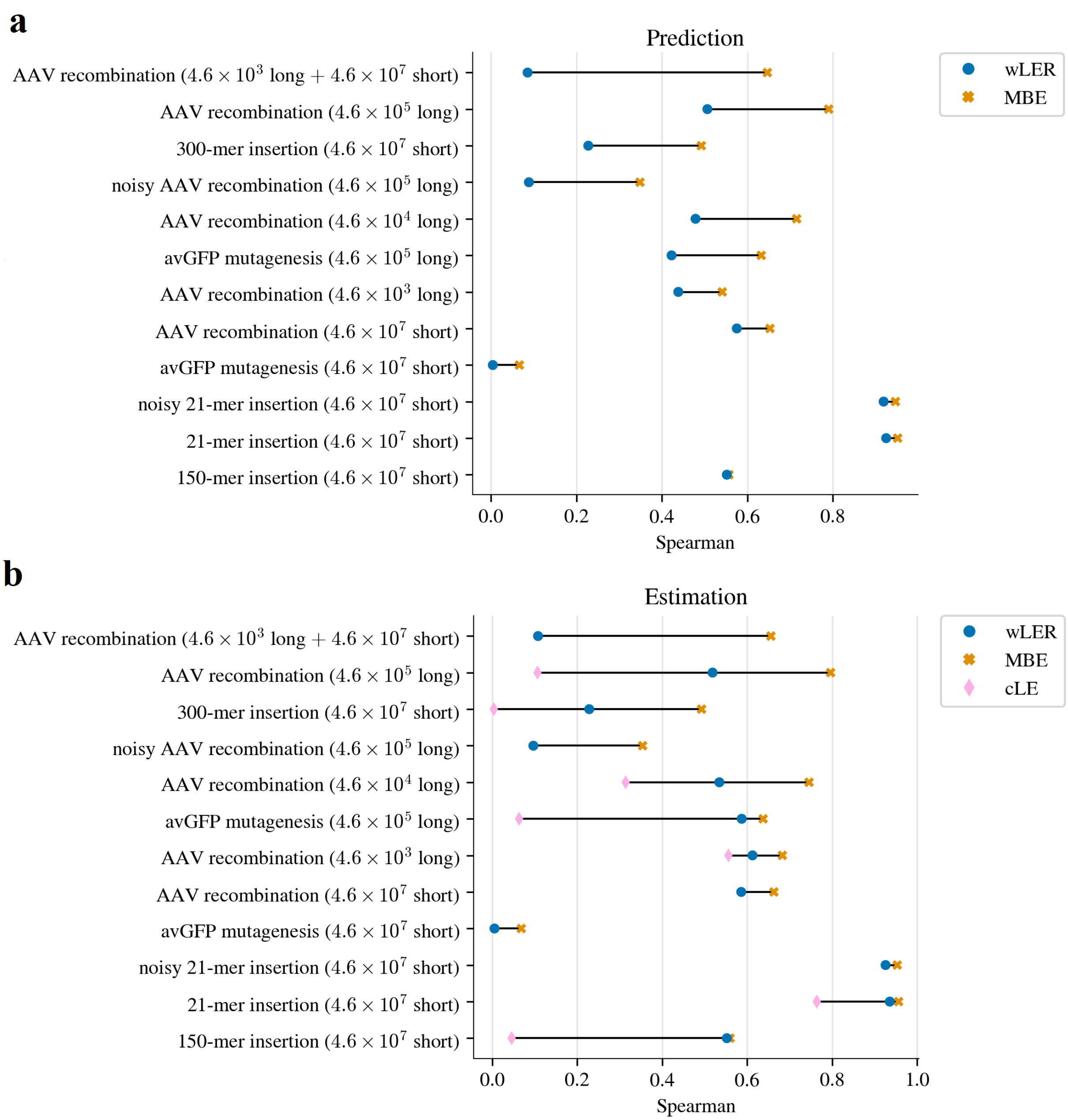


Figure 3.2: Simulated library results. Spearman correlation between ground truth fitness and cLE, wLER, and MBE estimates on full-length sequences of interest for the tasks of (a) prediction and (b) estimation. For wLER and MBE, each panel displays the Spearman correlation achieved by the best-performing model architecture for each method on each simulated dataset. The cLE approach can only be used for estimation (not prediction), and additionally, only for experiments where the sequencing reads were long enough to cover the sequences of interest ("Cover=100"). Thus, cLE is missing from some experiments. All differences are statistically significant ($p < 10^{-10}$). Results shown are for the best architecture for each approach, as described in main text. For comprehensive results across all model architectures see Fig. B.1.

setting (Fig. 3.2a, Fig. B.3d-e, Fig. B.4). We observed similar trends for the estimation task (Fig. 3.2b, Fig. B.1b, Fig. B.6).

We also explored how predictive performance was affected by the number of sequencing reads collected. When we increase the total number of long reads for the AAV recombination library (from 4.6×10^3 to 4.6×10^5), more unique sequences with low counts occur in the data (Extended Data Fig. B.4). Consequently, wLER is particularly challenged because it is trained using cLE estimates that cannot share data across non-identical reads to mitigate the effects of low sequencing counts. In fact, wLER is so challenged that, for many model architectures, its performance degrades when provided with more long-read sequencing data (Extended Data Fig. B.2a-c). In contrast, MBE follows a more intuitive pattern: more training data always either maintained or improved performance, but never hurt the overall performance metrics (Fig. 3.2, Extended Data Fig. B.2).

Short- and hybrid-read settings

In practice, experimenters often offset the sparsity of long-read sequencing by augmenting with higher-throughput short-read sequencing, thereby creating hybrid-read datasets. Motivated by this idea, we compared performance of the wLER and MBE methods when trained on short- and hybrid-read datasets. Again, our results follow our intuition: in both settings, MBE outperformed wLER (Fig. 3.2a, Fig. B.2d-f). In fact, because wLER cannot leverage partial overlap between reads, its accuracy actually decreased when long-read data was supplemented with additional short reads, despite the fact that this creates a larger overall training set. In contrast, MBE, again, behaved more intuitively: its accuracy improved with this larger dataset.

Negative selection

In negative selection experiments, the property being selected for is opposite from the property of interest. Thus, a key goal is to produce accurate predictions for sequences with low ground truth fitness. The post-selection counts for such low-fitness sequences are, by definition, low, making these estimates extremely challenging. To analyze this specific use case, we compared wLER and MBE predictive accuracy using generalized Spearman correlation focused on sequences with low ground truth fitness. MBE achieved higher predictive accuracy, not only overall, but also specifically on the subset of the test sequences with lowest true fitness (Fig. 3.3).

Selection for sequence selectivity

A key reason to seek high predictive accuracy for the negative selection task is so that we can leverage this task to perform a selectivity experiment, wherein we seek to identify sequences that are simultaneously high in a positive fitness and low in the negative fitness. We found that MBE yielded better predictive accuracy on both fitnesses than wLER (Fig. B.7a-b,

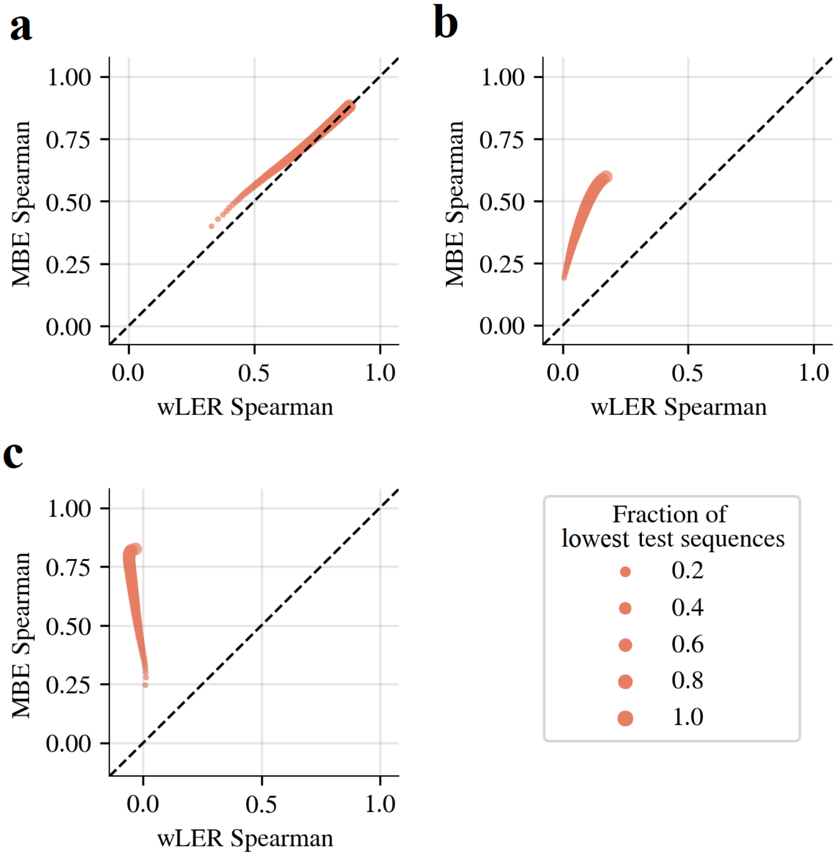


Figure 3.3: Simulated negative selection prediction results. Comparison of wLER and MBE predictive accuracy for simulated negative selection using the 100-unit NNs on the (a) 21-mer insertion (4.6 × 10^7 short reads), (b) avGFP mutagenesis (4.6 × 10^5 long reads), and (c) AAV recombination (4.6 × 10^5 long reads) datasets. Dot size represents the fraction of test sequences with lowest ground truth fitness used to compute Spearman correlation. In these experiments, we focus on sequences with lower ground truth fitness, which are the smaller dots. The dashed black line represents equal performance of the two approaches.

d-e, and g-h). Moreover, MBE was better than wLER at identifying selective sequences, where we measured a sequence's selectivity using the difference between its positive and negative fitness values—the larger this difference, the more selective the sequence is for the positive selection relative to the negative selection. MBE was more accurate than wLER in identifying top selective sequences: the best sequences identified by MBE were, on average, closer to a theoretical optimally-selective sequence, compared to wLER (Fig. 3.4, Fig. B.7c, f, and i). Overall, for each of dataset, MBE was significantly better than wLER at identifying the 1% of test sequences with highest true selectivity ($p < 10^{-3}$).

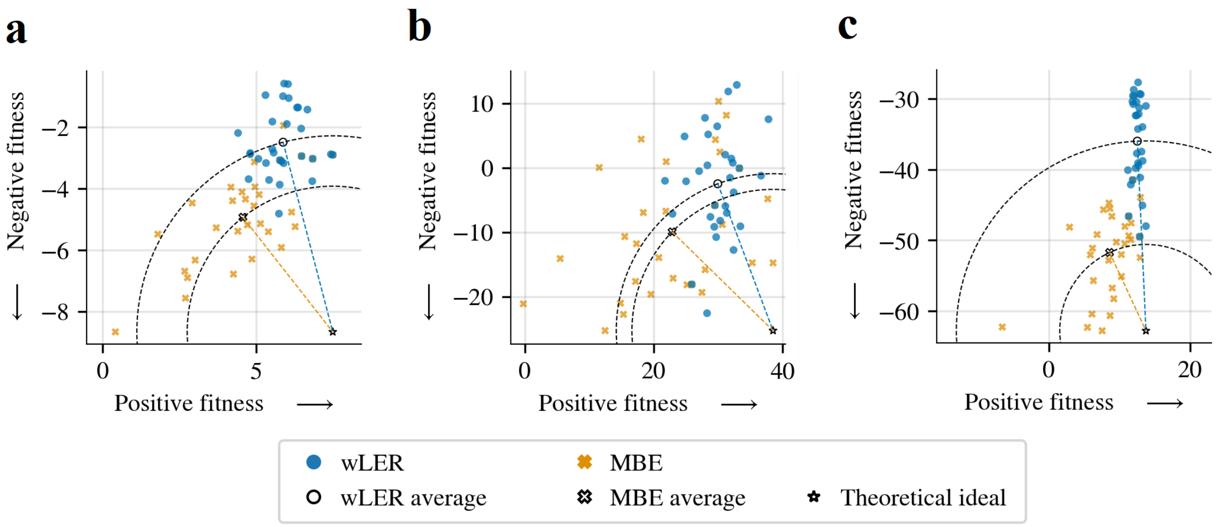


Figure 3.4: Simulated sequence selectivity prediction results. Comparison of wLER and MBE (using 100-unit NNs) for identifying selective test sequences over three simulated datasets: (a) 21-mer insertion (4.6×10^7 short reads), (b) avGFP mutagenesis (4.6×10^5 long reads), and (c) AAV recombination (4.6×10^5 long reads). Colored points show the true positive and negative fitness of the top ten test sequences identified from each of three test folds from three-fold cross-validation according to each model's predicted selectivity (*i. e.,* difference in predicted positive and negative fitness values). To gauge overall performance, the average point from each method is also plotted in black-and-white, as is a theoretical optimally-selective sequence (star) with the maximum positive fitness and minimum negative fitness among all sequences in the relevant dataset. Distance from optimal to average is conveyed by a circular contour line through the average point for each method.

3.6 Results on experimental high-throughput selection data

Having characterized the behavior of wLER and MBE in a broad range of simulated settings, we applied these methods on real experimental data. Across all the experimental datasets, MBE achieved better predictive accuracy than wLER (Fig. 3.5, Fig. B.9). For the SARS-CoV-2 dataset from Huisman *et al.* [35], we also found that predictions of experimental IC_{50} by MBE were more accurate than the predictions by NetMHCIIpan4.0, a model specifically devised to predict peptide binding to MHC II molecules (Table B.1). Recall that an important challenge with experimental data is that, to obtain the best ground truth experimental values possible, we require access to detailed biophysical assays rather than sequencing-based proxies. Consequently, the validation data we have access to have extremely limited sample sizes (ranging from 5–24 test points), thereby limiting our our ability to detect statistical significance on each dataset individually. Nevertheless, the trends that we observed on the simulated data continue on each dataset, and when performance over all of them is considered jointly, the improvement of MBE over wLER is statistically significant ($p < 0.03$) (Fig. 3.5). Thus, our experimental results (Fig. 3.5) demonstrate that switching from a LE-based approach to MBE leads to comparable or improved results across a range of

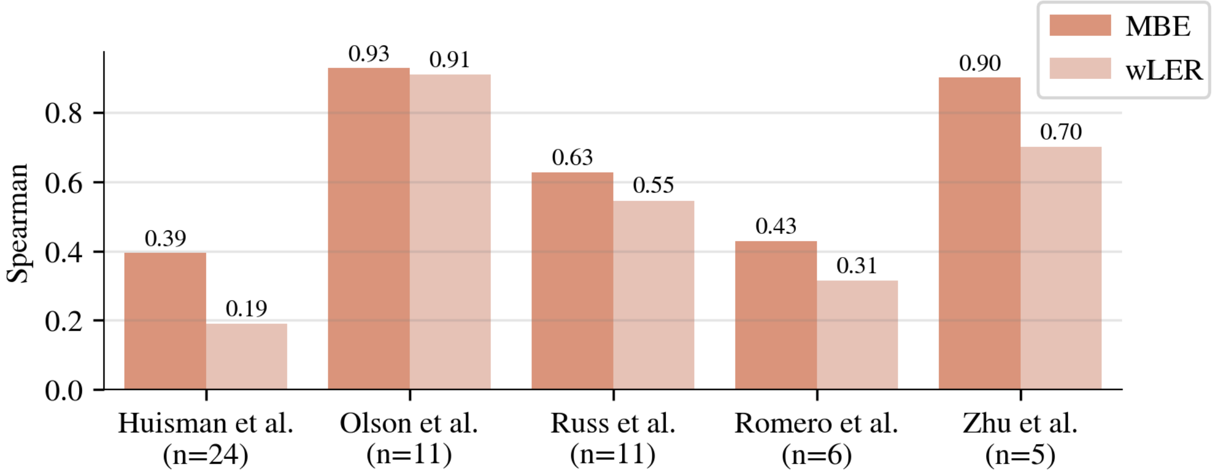


Figure 3.5: Real experimental prediction results. Comparison of Spearman correlation between wLER or MBE predictions and n experimental property measurements from Zhu *et al.*[116], Huisman *et al.*[35], Olson *et al.*[63], Russ *et al.*[80], and Romero *et al.*[78]. Each method is trained on real pre- and post-selection sequencing data, then used to predict the fitness of the n unobserved test sequences. The 100-unit NN model architecture is used for all datasets except that from Romero *et al.*[78], for which the linear architecture with IS features is used. The average performance improvement of MBE over wLER over all five experimental datasets, jointly, is statistically significant ($p < 0.03$).

laboratory applications, including: clinically-relevant selection experiments in gene therapy and immunology, a yeast-display assay, an mRNA display screen for protein binding, and both an *in vivo* complementation assay and microfluidic screen for enzyme activity.

3.7 Conclusion

Quantitatively characterizing the difference in sequence abundances between two conditions using high-throughput sequencing data—as occurs in high-throughput selection experiments like those in Chapter 2—is a key component in answering a large range of scientific questions. Not only do we wish to quantify the differences in observed data, but we also often want to predict the difference for sequences not yet observed—for example, in order to design improved starting libraries for further rounds of experimentation. Until now, such quantification was accomplished by counting the number of times a sequence occurred in each condition and taking the ratio of these counts (after normalization). Optionally, one may then have constructed a regression model to predict these count-based log-enrichment ratios. A key issue underlying this approach is the inability of count-based estimates to share any information across sequences that are not identical, when such sharing of information can be extremely valuable. In this Chapter, we introduced and evaluated a framework that overcomes this key limitation. This new framework is based on a reformulation of the log-enrichment estimation problem that uses density ratio estimation, implemented using any standard machine learning classifier. Our new method, model-based enrichment, improves performance over competing approaches based on either raw counts or weighted regression on count-based log-enrichment. In particular, we show this improvement holds across a broad range of simulated and real experimental data from high-throughput selection experiments on large sequence libraries. Our method enables estimation of log-enrichment in challenging experimental setups comprised of, for example, short reads spanning a sequence of interest; long reads with poor coverage; a mixture of both short and long reads; and settings with more than two conditions, such as when we seek to find sequences enriched by one selection and negatively selected by another—as occurs in designing libraries of gene therapy viral vectors that selectively infect one cell type but not another. In general, our approach also helps to mitigate poor estimates arising from relatively little sequencing data. It will be valuable to perform further validation of these results as more experimental data become available.

Our newly-developed method can immediately leverage any advances in general machine learning classifiers, and naturally handles sequencing reads of variable lengths within a given experiment whenever the classifier itself does so—as we demonstrated using convolutional neural networks. The predictive performance of such variable-length classifiers can potentially be further improved by incorporating other informative inputs in addition to read sequence. For example, in applications where it is possible to align to a known reference sequence prior to modeling, one may incorporate the mapped position for each read as an additional input. We anticipate that, as high-throughput selection experiments and sequencing-based assays continue to become more varied in their applications, the full potential of model-based enrichment to improve predictive performance and guide challenging downstream tasks such as library design and protein engineering will be further revealed.

Chapter 4

Concluding Remarks

In this dissertation, we have discussed several key problems relating to the application of machine learning to designing libraries of biological sequences, including:

- **Chapter 2**: optimally balancing the competing goals of library diversity and fitness within a principled computational framework; and
- **Chapter 3**: improving the accuracy and robustness of fitness estimation and prediction from high-throughput sequencing-based assays and selection experiments.

In the coming years, as new experimental technologies are introduced that lower the cost of synthesizing large, designed libraries and enable higher-throughput characterization of protein, DNA, and RNA structures and properties, we can anticipate that the set of technical problems related to machine learning-guided library design will only continue to grow in scope.

Looking to this future, it will be critical for researchers tackling these problems to recognize the utility that statistical and machine learning theory can have in guiding both computational and experimental approaches to design. Although advances in machine learning model architectures often outpace the development of accompanying theory, theoretical insights from even simplified systems can facilitate more efficient collection and use of biological sequence data. For example, in Chapter 3, theoretical knowledge of the asymptotic efficiency of classification-based density ratio estimators guided the development of a new perspective of high-throughput selection, and a more statistically efficient approach for estimating and predicting fitness based on sequencing-based assay data. Additionally, recent work combining density ratio estimation techniques with theory of statistical uncertainty and robustness led to a novel computational design approach that can take into account models' predictive uncertainty, ultimately resulting in more reliable design decisions and quantitative risk estimates for practitioners [21]. In the last few years, several studies have successfully leveraged signal processing and compressed sensing theory to help gain understanding into the role of epistasis in sequence-fitness landscapes [11, 3, 20]. Such insights from compressed sensing theory can also be used to (i) inform both experimental and computational decisions regarding the type and amount of assay data required for modeling [11], and (ii) decipher epistatic signals in machine learning models of fitness to potentially guide experimentalists towards new biological insights or interpretations [20].

An important advantage of data-driven design approaches is that, unlike rational design, a detailed physical understanding of the sequence properties being studied is not a prerequisite. However, to further improve the overall success of machine learning-guided design, an important challenge will be to efficiently combine information from large-scale data with the type of domain knowledge used for rational design, when available. Intuitively, modeling approaches that can intelligently combine multiple sources of data—and automatically learn which sources are most useful in a given application—promise to be more powerful than computational approaches based on single sources. For example, design approaches that coherently combine high-throughput assay data, structural information, evolutionary information from known homologous sequences, and prior domain knowledge regarding active sites and mechanisms of action promise to be much more informative than approaches utilizing only one of these information sources. Indeed, recent work has clearly shown that combining assay and evolutionary data when training machine learning models can improve predictive accuracy compared to using either type of data alone [33]. Combining structural models with knowledge of important protein symmetries or interfaces has also been shown to improve design success rates [103]. Future modeling techniques for sequence and library design should strive to efficiently combine multiple types of information—for example, by developing new loss and objective functions that capture multiple information sources—to enable more accurate predictive and generative performance.

Computational approaches that intelligently combine multiple sources of information will be of critical importance in making progress on the problem of designing protein active sites—one that is, currently, particularly challenging to solve with machine learning techniques. Often, machine learning-guided design approaches are used to design areas on the protein surface (*e. g.,* Chapter 2) or parts of the protein around the active site while preserving critical active site residues (*e. g.,* by designing around a fixed active site motif [103]). In the last few years, model-guided design approaches have emerged that incorporate evolutionary and molecular structure information with domain knowledge about active sites, binding conformations, and stabilizing mechanisms in order to produce libraries of functional active site designs [105, 114]. Success rates for these approaches remain low at present (as low as 0.03% [114]). What is clear, however, is that further development of modeling and optimization frameworks that incorporate both large-scale data and application-specific domain expertise in principled ways will be important in improving our ability to design novel, functional protein active sites using machine learning techniques.

In the short-term future, innovations in the space of machine learning-guided library design are likely to be driven by the invention and refinement of laboratory techniques. Computational researchers continuing work in this area will find it advantageous to study the complex biology underlying relevant experimental techniques, and use this knowledge to guide their research directions. At the same time, experimentalists will benefit from close relationships with computational scientists; an understanding of which data are likely to be most useful for modeling purposes will inform decisions regarding which specific laboratory techniques to adopt or develop. For example, such symbiotic relationships between computational and experimental researchers motivated the study in Chapter 2, which was performed in tandem with experimentalist in Professor David Schaffer's research group from project conception through final experimental validation of cell-specific viral vectors. It is our hope that, by having demonstrated the clear advantages of such collaborations, the work presented here will encourage computational and experimental researchers to form fruitful collaborations in the design space. Ultimately, it will be the insights gained from such relationships that will accelerate progress towards the ultimate goal of generalizable sequence design and engineering, to the benefit of the scientific community as a whole.

Bibliography

[1] Martin Abadi et al. “{TensorFlow}: a system for {Large-Scale} machine learning”. In: *12th USENIX symposium on operating systems design and implementation (OSDI 16)*. 2016, pp. 265–283.

[2] Kei Adachi et al. “Drawing a high-resolution functional map of adeno-associated virus capsid by massively parallel sequencing”. In: *Nature communications* 5.1 (2014), pp. 1–14.

[3] Amirali Aghazadeh et al. “Epistatic Net allows the sparse spectral regularization of deep neural networks for inferring fitness functions”. In: *Nature communications* 12.1 (2021), p. 5225.

[4] Uri Arad. “Modified Hirt procedure for rapid purification of extrachromosomal DNA from mammalian cells”. In: *Biotechniques* 24.5 (1998), pp. 760–762.

[5] MA Bartel, JR Weinstein, and DV Schaffer. “Directed evolution of novel adeno-associated viruses for therapeutic gene delivery”. In: *Gene therapy* 19.6 (2012), pp. 694–700.

[6] Lisa Bartoli et al. “The pros and cons of predicting protein contact maps”. In: *Protein Structure Prediction* (2008), pp. 199–217.

[7] Pascal Batard, Martin Jordan, and Florian Wurm. “Transfer of high copy number plasmid into mammalian cells by calcium phosphate transfection”. In: *Gene* 270.1-2 (2001), pp. 61–68.

[8] Surojit Biswas et al. “Low-N protein engineering with data-efficient deep learning”. In: *Nature methods* 18.4 (2021), pp. 389–396.

[9] Jesse D Bloom. “Software for the analysis and visualization of deep mutational scanning data”. In: *BMC bioinformatics* 16.1 (2015), pp. 1–13.

[10] Sylvie Boutin et al. “Prevalence of serum IgG and neutralizing factors against adeno-associated virus (AAV) types 1, 2, 5, 6, 8, and 9 in the healthy population: implications for gene therapy using AAV vectors”. In: *Human gene therapy* 21.6 (2010), pp. 704–712.

[11] David H Brookes, Amirali Aghazadeh, and Jennifer Listgarten. “On the sparsity of fitness functions and implications for learning”. In: *Proceedings of the National Academy of Sciences* 119.1 (2022), e2109649118.

[12] Drew H Bryant et al. "Deep diversification of an AAV capsid protein by machine learning". In: *Nature Biotechnology* 39.6 (2021), pp. 691–696.

[13] Akosua Busia and Jennifer Listgarten. "Model-based differential sequencing analysis". In: *bioRxiv* (2023).

[14] Leah C Byrne et al. "In vivo–directed evolution of adeno-associated virus in the primate retina". In: *JCI insight* 5.10 (2020).

[15] Anne Chao, Chun-Huo Chiu, and Lou Jost. "Phylogenetic diversity measures and their decomposition: a framework based on Hill numbers". In: *Biodiversity Conservation and Phylogenetic Systematics* 14 (2016).

[16] Sourav R Choudhury et al. "In vivo selection yields AAV-B1 capsid for central nervous system and muscle gene therapy". In: *Molecular Therapy* 24.7 (2016), pp. 1247–1257.

[17] Patrick C Cirino, Kimberly M Mayer, and Daisuke Umeno. "Generating mutant libraries using error-prone PCR". In: *Directed evolution library creation.* Springer, 2003, pp. 3–9.

[18] Deniz Dalkara et al. "In vivo–directed evolution of a new adeno-associated virus for therapeutic outer retinal gene delivery from the vitreous". In: *Science translational medicine* 5.189 (2013), 189ra76–189ra76.

[19] Benjamin E Deverman et al. "Cre-dependent selection yields AAV variants for widespread gene transfer to the adult brain". In: *Nature biotechnology* 34.2 (2016), pp. 204–209.

[20] Yigit Efe Erginbas et al. "Efficiently Computing Sparse Fourier Transforms of q-ary Functions". In: *arXiv preprint arXiv:2301.06200* (2023).

[21] Clara Fannjiang and Jennifer Listgarten. "Autofocused oracles for model-based design". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 12945–12956.

[22] Mary F Farrow and Frances H Arnold. "Combinatorial recombination of gene fragments to construct a library of chimeras". In: *Current Protocols in Protein Science* 62.1 (2010), pp. 26–2.

[23] Douglas M Fowler et al. "Enrich: software for analysis of protein function by enrichment and depletion of variants". In: *Bioinformatics* 27.24 (2011), pp. 3430–3431.

[24] Edward J Fox et al. "Accuracy of next generation sequencing platforms". In: *Next generation, sequencing & applications* 1 (2014).

[25] Sam Gelman et al. "Neural networks to learn protein sequence–function relationships from deep mutational scanning data". In: *Proceedings of the National Academy of Sciences* 118.48 (2021), e2104878118.

[26] Lori Giver and Frances H Arnold. "Combinatorial protein design by in vitro recombination". In: *Current opinion in chemical biology* 2.3 (1998), pp. 335–338.

[27] Michael Gutmann and Jun-ichiro Hirayama. "Bregman divergence as general framework to estimate unnormalized statistical models". In: *arXiv preprint arXiv:1202.3727* (2012).

[28] Edward P. Harvey et al. "An in silico method to assess antibody fragment polyreactivity". In: *bioRxiv* (2022). DOI: `10.1101/2022.01.12.476085`. eprint: `https://www.biorxiv.org/content/early/2022/01/13/2022.01.12.476085.full.pdf`. URL: `https://www.biorxiv.org/content/early/2022/01/13/2022.01.12.476085`.

[29] Ting He et al. "The influence of murine genetic background in adeno-associated virus transduction of the mouse brain". In: *Human Gene Therapy Clinical Development* 30.4 (2019), pp. 169–181.

[30] National Institutes of Health et al. "US National Library of Medicine. ClinicalTrials. gov". In: *Clinical study to investigate the effect of the combination of psychotropic drugs and an opioid on ventilation. https://clinicaltrials. gov/ct2/show/NCT04310579. Accessed May* 15 (2020).

[31] Olivier Henaff. "Data-efficient image recognition with contrastive predictive coding". In: *International conference on machine learning*. PMLR. 2020, pp. 4182–4192.

[32] Thomas A Hopf et al. "Mutation effects predicted from sequence co-variation". In: *Nature biotechnology* 35.2 (2017), pp. 128–135.

[33] Chloe Hsu et al. "Learning protein fitness models from evolutionary and assay-labeled data". In: *Nature Biotechnology* (2022), pp. 1–9.

[34] Dongmei Hu et al. "Effective optimization of antibody affinity by phage display integrated with high-throughput DNA synthesis and sequencing technologies". In: *PloS one* 10.6 (2015), e0129125.

[35] Brooke D Huisman et al. "A high-throughput yeast display approach to profile pathogen proteomes for MHC-II binding". In: *eLife* 11 (July 2022). Ed. by Satyajit Rath and Evan W Newell, e78589. ISSN: 2050-084X. DOI: `10.7554/eLife.78589`. URL: `https://doi.org/10.7554/eLife.78589`.

[36] Jae-Hyung Jang et al. "An evolved adeno-associated viral variant enhances gene delivery and gene targeting in neural stem cells". In: *Molecular Therapy* 19.4 (2011), pp. 667–675.

[37] Edwin T Jaynes. "Information theory and statistical mechanics". In: *Physical review* 106.4 (1957), p. 620.

[38] Nisha Kanwar et al. "PacBio sequencing output increased through uniform and directional fivefold concatenation". In: *Scientific reports* 11.1 (2021), pp. 1–13.

[39] DJSM Katz et al. "Obtaining confidence intervals for the risk ratio in cohort studies". In: *Biometrics* (1978), pp. 469–474.

[40] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[41] Jack PC Kleijnen and Reuven Y Rubinstein. “Optimization and sensitivity analysis of computer simulation models by the score function method”. In: *European Journal of Operational Research* 88.3 (1996), pp. 413–427.

[42] James T Koerber et al. “Molecular evolution of adeno-associated virus for enhanced glial gene delivery”. In: *Molecular Therapy* 17.12 (2009), pp. 2088–2095.

[43] Sriram Kosuri and George M Church. “Large-scale de novo DNA synthesis: technologies and applications”. In: *Nature methods* 11.5 (2014), pp. 499–507.

[44] Gioele La Manno et al. “Molecular diversity of midbrain development in mouse, human, and stem cells”. In: *Cell* 167.2 (2016), pp. 566–580.

[45] MD Lane and B Seelig. “Directed evolution of novel proteins”. In: *Curr Opin Chem Biol* 22 (2014), pp. 129–126.

[46] Katherine S Lim et al. “Machine learning on DNA-encoded library count data using an uncertainty-aware probabilistic loss function”. In: *Journal of Chemical Information and Modeling* (2022).

[47] Rui Lin et al. “Directed evolution of adeno-associated virus for efficient gene delivery to microglia”. In: *Nature Methods* 19.8 (2022), pp. 976–985.

[48] Michael I Love, Wolfgang Huber, and Simon Anders. “Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2”. In: *Genome biology* 15.12 (2014), pp. 1–21.

[49] Jan H Lui, David V Hansen, and Arnold R Kriegstein. “Development and evolution of the human neocortex”. In: *Cell* 146.1 (2011), pp. 18–36.

[50] Narendra Maheshri et al. “Directed evolution of adeno-associated virus yields enhanced gene delivery vectors”. In: *Nature biotechnology* 24.2 (2006), pp. 198–204.

[51] Andrew D Marques et al. “Applying machine learning to predict viral assembly for adeno-associated virus capsid libraries”. In: *Molecular Therapy-Methods & Clinical Development* 20 (2021), pp. 276–286.

[52] Shannon A Marshall et al. “Rational design and engineering of therapeutic proteins”. In: *Drug discovery today* 8.5 (2003), pp. 212–221.

[53] Sophie N Mathiesen et al. “CNS transduction benefits of AAV-PHP. eB over AAV9 are dependent on administration route and mouse strain”. In: *Molecular Therapy-Methods & Clinical Development* 19 (2020), pp. 447–458.

[54] Sebastian Matuszewski et al. “A statistical guide to the design of deep mutational scanning experiments”. In: *Genetics* 204.1 (2016), pp. 77–87.

[55] Quinn McNemar. “Note on the sampling error of the difference between correlated proportions or percentages”. In: *Psychometrika* 12.2 (1947), pp. 153–157.

[56] Shakir Mohamed and Balaji Lakshminarayanan. “Learning in implicit generative models”. In: *arXiv preprint arXiv:1610.03483* (2016).

[57] Oliver J Müller et al. "Random peptide libraries displayed on adeno-associated virus to select for targeted gene therapy vectors". In: *Nature biotechnology* 21.9 (2003), pp. 1040–1046.

[58] Whitney K Newey and Daniel McFadden. "Large sample estimation and hypothesis testing". In: *Handbook of econometrics* 4 (1994), pp. 2111–2245.

[59] Pascal Notin et al. "Tranception: protein fitness prediction with autoregressive transformers and inference-time retrieval". In: *International Conference on Machine Learning*. PMLR. 2022, pp. 16990–17017.

[60] Simon J O'Carroll, William H Cook, and Deborah Young. "AAV targeting of glial cell types in the central and peripheral nervous system and relevance to human gene therapy". In: *Frontiers in Molecular Neuroscience* 13 (2021), p. 618020.

[61] Pierce J Ogden et al. "Comprehensive AAV capsid fitness landscape reveals a viral gene and enables machine-guided design". In: *Science* 366.6469 (2019), pp. 1139–1143.

[62] David S Ojala et al. "In vivo selection of a computationally designed SCHEMA AAV library yields a novel variant for infection of adult neural stem cells in the SVZ". In: *Molecular Therapy* 26.1 (2018), pp. 304–319.

[63] C Anders Olson, Nicholas C Wu, and Ren Sun. "A comprehensive biophysical description of pairwise epistasis throughout an entire protein domain". In: *Current biology* 24.22 (2014), pp. 2643–2651.

[64] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. "Representation learning with contrastive predictive coding". In: *arXiv preprint arXiv:1807.03748* (2018).

[65] Andrew S Parker, Karl E Griswold, and Chris Bailey-Kellogg. "Optimization of combinatorial mutagenesis". In: *Journal of computational biology* 18.11 (2011), pp. 1743–1756.

[66] Rupali P Patwardhan et al. "High-resolution analysis of DNA regulatory elements by synthetic saturation mutagenesis". In: *Nature biotechnology* 27.12 (2009), pp. 1173–1175.

[67] Jian Peng and Jinbo Xu. "Low-homology protein threading". In: *Bioinformatics* 26.12 (2010), pp. i294–i300.

[68] Yi-Rong Peng et al. "Molecular classification and comparative taxonomics of foveal and peripheral cells in primate retina". In: *Cell* 176.5 (2019), pp. 1222–1237.

[69] Luca Perabo et al. "In vitro selection of viral vectors with modified tropism: the adeno-associated virus display". In: *Molecular therapy* 8.1 (2003), pp. 151–157.

[70] Frank J Poelwijk, Michael Socolich, and Rama Ranganathan. "Learning the pattern of epistasis linking genotype and phenotype in a protein". In: *Nature communications* 10.1 (2019), pp. 1–11.

[71] Jing Qin. "Inferences for case-control and semiparametric two-sample density ratio models". In: *Biometrika* 85.3 (1998), pp. 619–630.

[72] C Garrett Rappazzo, Brooke D Huisman, and Michael E Birnbaum. "Repertoire-scale determination of class II MHC peptide binding via yeast display improves antigen prediction". In: *Nature communications* 11.1 (2020), pp. 1–14.

[73] Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. "On the convergence of adam and beyond". In: *arXiv preprint arXiv:1904.09237* (2019).

[74] Anthony Rhoads and Kin Fai Au. "PacBio sequencing and its applications". In: *Genomics, proteomics & bioinformatics* 13.5 (2015), pp. 278–289.

[75] Adam J Riesselman, John B Ingraham, and Debora S Marks. "Deep generative models of genetic variation capture the effects of mutations". In: *Nature methods* 15.10 (2018), pp. 816–822.

[76] Mark D Robinson, Davis J McCarthy, and Gordon K Smyth. "edgeR: a Bioconductor package for differential expression analysis of digital gene expression data". In: *bioinformatics* 26.1 (2010), pp. 139–140.

[77] Philip A Romero and Frances H Arnold. "Exploring protein fitness landscapes by directed evolution". In: *Nature reviews Molecular cell biology* 10.12 (2009), pp. 866–876.

[78] Philip A Romero, Tuan M Tran, and Adam R Abate. "Dissecting enzyme function with microfluidic-based deep mutational scanning". In: *Proceedings of the National Academy of Sciences* 112.23 (2015), pp. 7159–7164.

[79] Alan F Rubin et al. "A statistical framework for analyzing deep mutational scanning data". In: *Genome biology* 18.1 (2017), pp. 1–15.

[80] William P Russ et al. "An evolution-based model for designing chorismate mutase enzymes". In: *Science* 369.6502 (2020), pp. 440–445.

[81] Jorge Santiago-Ortiz et al. "AAV ancestral reconstruction library enables selection of broadly infectious viral variants". In: *Gene therapy* 22.12 (2015), pp. 934–946.

[82] Karen S Sarkisyan et al. "Local fitness landscape of the green fluorescent protein". In: *Nature* 533.7603 (2016), pp. 397–401.

[83] Pauline F Schmit et al. "Cross-packaging and capsid mosaic formation in multiplexed AAV libraries". In: *Molecular Therapy-Methods & Clinical Development* 17 (2020), pp. 107–121.

[84] Bo Segerman. "The most frequently used sequencing technologies and assembly methods in different time segments of the bacterial surveillance and RefSeq genome databases". In: *Frontiers in Cellular and Infection Microbiology* 10 (2020), p. 527102.

[85] Hyeoncheol Francis Son et al. "Rational protein engineering of thermo-stable PETase from Ideonella sakaiensis for highly efficient PET degradation". In: *ACS Catalysis* 9.4 (2019), pp. 3519–3526.

[86] Hyebin Song et al. "Inferring protein sequence-function relationships with large-scale positive-unlabeled learning". In: *Cell systems* 12.1 (2021), pp. 92–101.

[87] James H Steiger. “Tests for comparing elements of a correlation matrix.” In: *Psychological bulletin* 87.2 (1980), p. 245.

[88] Willem P Stemmer. “DNA shuffling by random fragmentation and reassembly: in vitro recombination for molecular evolution.” In: *Proceedings of the National Academy of Sciences* 91.22 (1994), pp. 10747–10751.

[89] Bianca K Stöcker, Johannes Köster, and Sven Rahmann. “SimLoRD: simulation of long read data”. In: *Bioinformatics* 32.17 (2016), pp. 2704–2706.

[90] Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. *Density ratio estimation in machine learning*. Cambridge University Press, 2012.

[91] Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. “Density-ratio matching under the Bregman divergence: a unified framework of density-ratio estimation”. In: *Annals of the Institute of Statistical Mathematics* 64.5 (2012), pp. 1009–1044.

[92] Kang Lan Tee and Tuck Seng Wong. “Polishing the craft of genetic diversity creation in directed evolution”. In: *Biotechnology advances* 31.8 (2013), pp. 1707–1721.

[93] Jonathan T Ting et al. “A robust ex vivo experimental platform for molecular-genetic dissection of adult human neocortical cell types and circuits”. In: *Scientific reports* 8.1 (2018), pp. 1–13.

[94] Cole Trapnell et al. “Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks”. In: *Nature protocols* 7.3 (2012), pp. 562–578.

[95] Longping Victor Tse et al. “Structure-guided evolution of antigenically distinct adeno-associated virus variants for immune evasion”. In: *Proceedings of the National Academy of Sciences* 114.24 (2017), E4812–E4821.

[96] Caroline M Tucker et al. “A guide to phylogenetic metrics for conservation, community ecology and macroecology”. In: *Biological Reviews* 92.2 (2017), pp. 698–715.

[97] Hanna Tuomisto. “A diversity of beta diversities: straightening up a concept gone awry. Part 1. Defining beta diversity as a function of alpha and gamma diversity”. In: *Ecography* 33.1 (2010), pp. 2–22.

[98] Michele Vendruscolo, Edo Kussell, and Eytan Domany. “Recovery of protein structure from contact maps”. In: *Folding and Design* 2.5 (1997), pp. 295–306.

[99] Deeptak Verma, Gevorg Grigoryan, and Chris Bailey-Kellogg. “Pareto optimization of combinatorial mutagenesis libraries”. In: *IEEE/ACM transactions on computational biology and bioinformatics* 16.4 (2018), pp. 1143–1153.

[100] Christopher A Voigt et al. “Protein building blocks preserved by recombination”. In: *Nature structural biology* 9.7 (2002), pp. 553–558.

[101] Annette Von Drygalski et al. “Etranacogene dezaparvovec (AMT-061 phase 2b): normal/near normal FIX activity and bleed cessation in hemophilia B”. In: *Blood advances* 3.21 (2019), pp. 3241–3247.

[102] Yuqiu Wang et al. “Directed evolution of adeno-associated virus 5 capsid enables specific liver tropism”. In: *Molecular Therapy-Nucleic Acids* 28 (2022), pp. 293–306.

[103] Joseph L Watson et al. “Broadly applicable and accurate protein design by integrating structure prediction networks and diffusion generative models”. In: *bioRxiv* (2022), pp. 2022–12.

[104] Eli N Weinstein et al. “Optimal design of stochastic DNA synthesis protocols based on generative sequence models”. In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2022, pp. 7450–7482.

[105] Jonathan Yaacov Weinstein et al. “Designed active-site library reveals thousands of functional GFP variants”. In: *bioRxiv* (2022), pp. 2022–10.

[106] James A Wells et al. “Designing substrate specificity by protein engineering of electrostatic interactions.” In: *Proceedings of the National Academy of Sciences* 84.5 (1987), pp. 1219–1223.

[107] Aaron M Wenger et al. “Accurate circular consensus long-read sequencing improves variant detection and assembly of a human genome”. In: *Nature biotechnology* 37.10 (2019), pp. 1155–1162.

[108] Kaumadi Wijesooriya et al. “Urgent need for consistent standards in functional enrichment analysis”. In: *PLoS computational biology* 18.3 (2022), e1009935.

[109] Emily E Wrenbeck, Matthew S Faber, and Timothy A Whitehead. “Deep sequencing methods for protein engineering and design”. In: *Current opinion in structural biology* 45 (2017), pp. 36–44.

[110] Sewall Wright. “Evolution in Mendelian populations”. In: *Genetics* 16.2 (1931), p. 97.

[111] Zachary Wu et al. “Machine learning-assisted directed protein evolution with combinatorial libraries”. In: *Proceedings of the National Academy of Sciences* 116.18 (2019), pp. 8852–8858.

[112] Feng Yan et al. “From reads to insight: a hitchhiker’s guide to ATAC-seq data analysis”. In: *Genome biology* 21 (2020), pp. 1–16.

[113] Kevin K Yang, Zachary Wu, and Frances H Arnold. “Machine-learning-guided directed evolution for protein engineering”. In: *Nature methods* 16.8 (2019), pp. 687–694.

[114] Andy Hsien-Wei Yeh et al. “De novo design of luciferases using deep learning”. In: *Nature* 614.7949 (2023), pp. 774–780.

[115] Wei Zheng, Alan M Friedman, and Chris Bailey-Kellogg. “Algorithms for joint optimization of stability and diversity in planning combinatorial libraries of chimeric proteins”. In: *Journal of Computational Biology* 16.8 (2009), pp. 1151–1168.

[116] Danqing Zhu et al. “Machine learning-based library design improves packaging and diversity of adeno-associated virus (AAV) libraries”. In: *bioRxiv* (2021).

[117] S Zolotukhin et al. “Recombinant adeno-associated virus purification using novel methods improves infectious titer and yield”. In: *Gene therapy* 6.6 (1999), pp. 973–985.

Appendix A

Supplementary Information: Designing Adeno-Associated Virus Capsid Insertion Libraries

A.1 Construction of insertion libraries

In Chapter 2, we used libraries with a variable 7 amino acid (7-mer) insertion region flanked by amino acid linkers (TGGLS) introduced at position 575-577 in the viral protein monomer. Each 7-mer NNK oligo was synthesized (Elim) and introduced to the 5' end of the right fragment by a primer overhang (7mer_F; Table A.1). For machine learning-designed libraries, instead of using NNK, we specified position-specific nucleotide probabilities (Table A.2) at the time of synthesis (GeneWiz) to be incorporated at the 5' end of the 7mer_F primer. Left and right fragments were each PCR amplified by primers Seq_F/Seq_R and 7mer_F/7mer_R, respectively (Table A.1). PCR products of the two fragments were then purified individually and subjected to overlap extension PCR (using HindIII_F and NotI_R primers) with Vent DNA polymerase (Thermo Fisher) with equimolar amounts of the left and right fragments for a total of 250ng DNA templates. The resulting library was then digested with HindIII and NotI (New England Biolabs, Inc.) and ligated into replication incompetent AAV packaging plasmid pSub2repKO [62] for library construction. The resulting ligation reaction was electroporated (Bio-Rad) into *Escherichia coli* for plasmid production and purification. HEK293T cells were originally obtained from the American Type Culture Collection (Manassas, VA, USA) and cultured in DMEM (GIBCO) with 10% fetal bovine serum (Invitrogen) and 1% penicillin/streptomycin (GIBCO) at 37°C and 5% CO_2. The passage number of 293T for packaging AAV libraries was between 10–15.

Primer	Sequence (5' - 3')
Seq_F	GGTGGAGCATGAATTCTACGTC
Seq_R	GCTCTGGTTGTTGGTGGCC
7mer_F	GGCCACCAACAACCAGAGCACCGGTNNKNNKNNKNNKNN KNNKNNKGGCTTAAGTTCCACCACTGCCC
7mer_R	GCTCTGGTTGTTGGTGGCC
Vg_F	GCGGAAGCTTCGATCAACTACG
Vg_R	CGCAGAGACCAAAGTTCAACTGA
HindIII_F	TTCCACGTCTTTATATGGTGCCCAGTC
NotI_R	CGCAGAGACCAAAGTTCAACTGA

Table A.1: Primer sequences for PCR reactions.

A.2 Vector packaging and production

AAV library vectors were packaged as described previously [62, 57] with transfection of HEK293T cells. Specifically, in a 75–80% confluent density 15 cm dish of HEK293T cells, 13.5 μg of pHelper, 9 μg of pBluescript (Addgene), 70 ng of the capsid plasmid library, and 5 μg of pRepHelper were co-transfected by the polyethyleneimine (PEI) method. This 1:2 $\times$ 10^{-4} molar ratio was calculated such that $>$ 90% of cells received approximately one or zero members of the capsid plasmid library to minimize occurrences of cross-packaging, assuming each cell receives approximately 50,000 total plasmids [7]. Cells were harvested 72 hours later, and the supernatant was collected. The cell pellet was resuspended in a lysis buffer (50mM Tris, 150mM NaCl, pH 8.5) and frozen/thawed three times using dry ice/ethanol. The lysate was then incubated at 37°C for 30 minutes with an addition of 10 U/mL of Benzonase (Invitrogen). Then, the lysate was first spun at 2000 rpm for 2 minutes, followed by a 10,000 rpm spin for 10 minutes, before the supernatant was all collected for purification. Collected virus was purified via iodixanol density centrifugation and buffer exchanged into PBS by Amicon (Ultra-15, Millipore) filtration.

This packaging process has the potential to be confounded by cross-packaging, in which viral particles are composed of viral genomes and capsid proteins derived from different library variants. To minimize cross-packaging, we diluted the plasmid library to a previously determined concentration that minimizes the event of multiple members of the capsid plasmid library entering into the same cell [50, 83]. To quantify capsid cross-packaging in a given library, we used green fluorescent protein (GFP) plasmid mixed with capsid libraries in 1:7 molar ratio and determined correctly-packaged versus cross-packaged viral particles using either Cap-specific or GFP-specific primers, respectively (Fig. A.1, Table A.8). These findings quantitatively characterized cross-packaging and provided experimental evidence of similar but minimal levels (less than 2%) of cross-packaging in all libraries in our study in Chapter 2.

To characterize the packaging ability of individual sequence plasmids (Fig. 2.6), each plasmid was packaged separately, and its titer was measured separately. Specifically, in a 75–80% confluent density 15 cm dish of HEK293T cells, 12 μg of pHelper, 10 μg of the pRepCap (AAV capsid variant), and 6 μg of GFP-encoding AAV vector plasmid were co-transfected by the PEI method. 72 hours later, collected virus was purified and buffer exchanged into PBS. We, then, measured the packaged viral titers using ddPCR with GFP-probe (CGCGATCACATGGTCCTGCTGG).

A.3 AAV viral genome extraction and titer

Packaged AAV vectors were first combined with equal volume of 10X DNase buffer (New England Biolabs, B0303S) and 0.5 μL 10 U/μL DNase I (New England Biolabs, M0303L) incubate for 30 minutes at 37°C. Then, equal volume of 2x Proteinase K Buffer was added with sample to break open capsid. After heat inactivating for 20 minutes at 95°C, the sample was further diluted at 1:1000 and 1:10,000 and use as templates for titer. DNase-resistant viral genomic titers were measured using digital-droplet PCR (ddPCR) (BioRad) using with Hex-ITR probes (CACTCCCTCTCTGCGCGCTCG) tagging the conserved regions of encapsidated viral genome of AAV. After primary tissue infection, capsid sequences were recovered by PCR from harvested cells using primers HindIII_F and NotI_R (Table A.1). A 75–85 base pair region containing the 7-mer insertion was PCR amplified from harvested DNA. Primers included the Illumina adapter sequences containing unique barcodes to allow for multiplexing of amplicons from multiple libraries. PCR amplicons were purified and sequenced with a single read run on Illumina NovaSeq 6000.

A.4 Primary adult human brain slice culture, library infection, and extraction

UCSF consent statement. De-identified tissue samples were collected with previous patient consent in strict observance of the legal and institutional ethical regulations. Sample use was approved by the Institutional Review Board at UCSF and experiments conform to the principles set out in the WMA Declaration of Helsinki and the Department of Health and Human Services Belmont Report.

Adult surgical specimens from epilepsy cases were obtained from the UCSF medical center in collaboration with neurosurgeons with previous patient consent. Surgically excised specimens were immediately placed in a sterile container filled with N-methyl-D-glucamine (NMDG) substituted artificial cerebrospinal fluid (aCSF) of the following composition (in mM): 92 NMDG, 2.5 KCl, 1.25 NaH_2PO_4, 30 $NaHCO_3$, 20 4-(2-hydroxyethyl)-1-piperazineethanesulfonic acid (HEPES), 25 glucose, 2 thiourea, 5 Na-ascorbate, 3 Napyruvate, 0.5 $CaCl_2 \cdot 4H_2O$ and 10 $MgSO_4 \cdot 7H_2O$. The pH of the NMDG aCSF was titrated pH to 7.3-–7.4 with 1M Tris-Base at pH8, and the osmolality was 300—305 mOsmoles/Kg. The solution was pre-chilled to 2—4°C and thoroughly bubbled with carbogen (95% O_2/5%

CO_2) gas prior to collection. The tissue was transported from the operating room to the laboratory for processing within 40—60 minutes. Blood vessels and meninges were removed from the cortical tissue, and then the tissue block was secured for cutting using superglue and sectioned perpendicular to the cortical plate to 300 μm using a Leica VT1200S vibrating blade microtome in aCSF. The slices were then transferred into a container of sterile-filtered NMDG aCSF that was pre-warmed to 32—34°C and continuously bubbled with carbogen gas. After a 12 minute recovery incubation, slices were transferred to slice culture inserts (Millicell, PICM03050) on six-well culture plates (Corning) and cultured in adult brain slice culture medium containing 840 mg MEM Eagle medium with Hanks salts and 2mM L-glutamine (Sigma, M4642), 18 mg ascorbic acid (Sigma, A7506), 3 mL HEPES (1M stock) (Sigma, H3537), 1.68 mL $NaHCO_3$ (892.75 mM solution, Gibco, 25080-094), 1.126 mL D-glucose, (1.11M solution, Gibco, A24940-01), 0.5 mL penicillin/streptomycin, 0.25 mL GlutaMax (at 400x, Gibco, 35050-061), 100 μL 2M stock $MgSO_4 \cdot 7H_2O$ (Sigma, M1880), 50 μL 2M stock $CaCl_2 \cdot 2H_2O$ (Sigma, C7902), 50 μL insulin from bovine pancreas, (10 mg/mL, Sigma, I0516), 20 mL horse serum-heat inactivated, 95 mL MilliQ H_2O (as previously described [93]). The following day after plating, adult human brain slices were infected with the viral library at an estimated of 10,000 MOI (N=3 per group) based on the number of cells estimated per slice. Slices were cultured at the liquid–air interface created by the cellculture insert in a 37°C incubator at 5% CO_2 for 72 hours post infection.

Seventy-two hours after infection with the viral library, cultured brain tissue slices were first rinsed with DPBS (Gibco, 14190250) twice and detached from the filters. Then mechanically minced to $1mm^2$ pieces and enzymatically digested with papain digestion kit (Worthington, LK003163) with the addition of DNase for 1 hour at 37°C. After the enzymatic digestion, tissue was mechanically triturated using fire-polished glass pipettes (Fisher Scientific, cat#13-678-6A), filtered through a 40 μm cell strainer (Corning 352340), pelleted at 300xg for 5 minutes and washed twice with DBPS. Following mechanical digestion, the slices were first treated with lysis buffer (10% SDS, 1M Tris-HCL, pH 7.4–8.0, and 0.5M EDTA, pH 8.0) with the addition of RNase A (Thermo Scientific, EN0531) for 60 minutes at 37°C and proteinase K (New England Biolabs, P8107S) for 3 hours at 55°C. The enzymatically-digested tissue homogenate was then proceeded to the Hirt column protocol as previously published [4].

A.5 Primary prenatal human brain slice library infection and cell purification

De-identified primary tissue samples were collected with previous patient consent in strict observance of the legal and institutional ethical regulations (see Section A.4). Cortical brain tissue was immediately placed in a sterile conical tube filled with oxygenated artificial cerebrospinal fluid (aCSF) containing 125 mM NaCl, 2.5 mM KCl, 1mM $MgCl_2$, 1 mM $CaCl_2$, and 1.25 mM NaH_2PO_4 bubbled with carbogen (95% O_2/5% CO_2). Blood vessels and meninges were removed from the cortical tissue, and then the tissue block was embedded in 3.5% low-melting-point agarose (Thermo Fisher, BP165-25) and sectioned perpendicular to the ventricle to 300 μm using a Leica VT1200S vibrating blade microtome in a sucrose protective aCSF containing 185 mM sucrose, 2.5 mM KCl, 1 mM MgCl2, 2 mM CaCl2, 1.25 mM NaH2PO4, 25 mM NaHCO3, 25 mM d-(+)-glucose. Slices were transferred to slice culture inserts (Millicell, PICM03050) on six-well culture plates (Corning) and cultured in prenatal brain slice culture medium containing 66% (vol/vol) Eagle's basal medium, 25% (vol/vol) HBSS, 2% (vol/vol) B27, 1% N2 supplement, 1% penicillin/streptomycin and GlutaMax (Thermo Fisher). Slices were cultured in a 37°C incubator at 5% CO_2, 8% O_2 at the liquid–air interface created by the cell-culture insert.

Cultured brain slices were washed twice with DPBS (Gibco, 14190250), detached from the filters and enzymatically digested with papain digestion kit (Worthington, LK003163) with the addition of DNase for 30 minutes at 37°C. Following enzymatic digestion, slices were mechanically triturated using a fire-polished glass pipette, filtered through a 40 μm cell strainer test tube (Corning 352235), pelleted at 300xg for 5 minutes and washed twice with DBPS.

Dissociated cells were resuspended in MACS buffer (DPBS with 1 mM EGTA and 0.5% BSA) with addition of DNAse and incubated with CD11b antibody (microglia) for 15 minutes on ice. After the incubation, cells were washed in a 10 ml of MACS buffer and loaded on LS columns (Miltenyi Biotec, 130-042-401) on the magnetic stand. Cells were washed 3 times with 3 ml of MACS buffer, then the column was removed from the magnetic field and microglia cells were eluted using 5 ml of MACS buffer. The flow-through cells were then gently prepared to separate out neurons using polysialylated-neural cell adhesion molecule (PSA-NCAM), and the flow-through cell population was used as glial-cell type. Cells were pelleted, re-suspended in 1 ml of culture media and counted.

A.6 Maximum entropy design of unconstrained libraries

In Section 2.5, we consider constrained library designs, where one specifies the marginal probabilities of observing each amino acid at each position. In contrast, unconstrained libraries are specified by listing all the oligonucleotide sequences that comprise the library. Unconstrained libraries are more flexible than constrained libraries because they provide more control over the contents of the library, but this increased flexibility comes at a substantially higher monetary cost per oligonucleotide. Therefore, when considering constrained versus unconstrained libraries, one must trade off flexibility and library size.

Although we did not experimentally construct any constrained libraries in our study in Section 2.5, demonstrate that it is possible to apply our maximum entropy formulation to the design of unconstrained libraries. For the purposes of comparison, we exactly compute the entropy and mean predicted log-enrichment of the maximum entropy library defined by

Equation 2.5 by enumerating all possible 7-mer insertion sequences and evaluating the same predictive model, $f(x)$, used to design the constrained libraries (Fig. 2.7) for each sequence. In general, even when it is not possible to fully enumerate the relevant sequence space, it is conceptually straightforward to build a list of sequences that approximates the maximum entropy library by sampling from this distribution with, for instance, Markov Chain Monte Carlo algorithms. This resulting set of samples represents a particle-based approximation to p_λ and thus will approximately respect the Pareto optimal property of the maximum entropy library.

We computed the entropy and mean predicted log-enrichment for unconstrained libraries corresponding to 404 different settings of λ (Fig. 2.8). We can see that, compared to constrained libraries, the unconstrained library construction allows one to build a library with greater diversity at the same level of mean predicted fitness. As oligonucleotide synthesis becomes cheaper, unconstrained library synthesis will became correspondingly cheaper. Therefore, our results suggest that at some point, it is likely that unconstrained libraries will become the libraries of choice.

A.7 Stochastic gradient descent for optimizing nucleotide probabilities

In Section 2.5, we design position-wise specified libraries by optimizing the individual probabilities of each nucleotide in each position according to the objective in Equation 2.9. Here, we describe the Stochastic Gradient Descent (SGD) algorithm we used to solve this nonconvex objective. We used a variant of SGD based on the score function estimator [41] to solve Equation 2.9. We randomly initialized a parameter matrix, $\phi^{(0)}$, with independent Normal samples, and then updated the parameters according to

$$\phi^{(t)} = \phi^{(t-1)} + \beta \nabla_\phi F\left(\phi^{(t-1)}\right) \tag{A.1}$$

for $t = 1, \dots, T$, where we define $F(\phi) = \mathbb{E}_{q_\phi}[f_\theta(x)] + \lambda H[q_\phi]$ to be the objective function in Equation 2.9. For the experiments in Section 2.5, we used $\beta = 0.01$ and the number of iterations was set to $T = 2000$ as we observed convergence of the objective function values in most runs of the optimization within this number of iterations. After T iterations, we assumed that we had reached a near-optimal solution (*i.e.*, that $\phi^{(T)}$ can be used as a decent approximation of ϕ_λ).

We now derive the gradient in Equation A.1. First, we recognize that the gradient of the entropy is given by

$$\begin{aligned}
\nabla_\phi H[q_\phi] &= -\nabla_\phi \mathbb{E}_{q_\phi}[\log q_\phi(x)] \\
&= -\sum_{x\in\mathcal{X}} \nabla_\phi q_\phi(x) \log q_\phi(x) \\
&= -\sum_{x\in\mathcal{X}} \left(\log q_\phi(x) \nabla_\phi q_\phi(x) + q_\phi(x)\nabla_\phi \log q_\phi(x)\right) \\
&= -\sum_{x\in\mathcal{X}} \left(\log q_\phi(x) q_\phi(x) \nabla_\phi \log q_\phi(x) + q_\phi(x)\nabla_\phi \log q_\phi(x)\right) \\
&= -\sum_{x\in\mathcal{X}} q_\phi(x)\left(1 + \log q_\phi(x)\right) \nabla_\phi \log q_\phi(x) \\
&= -\mathbb{E}_{q_\phi}\left[(1+\log q_\phi(x))\nabla_\phi \log q_\phi(x)\right]
\end{aligned}$$

where in the third line we have used the equality $\nabla_\phi q_\phi(x) = q_\phi(x)\nabla_\phi \log q_\phi(x)$. We, then, have

$$\begin{aligned}
\nabla_\phi F(\phi) &= \nabla_\phi \mathbb{E}_{q_\phi}[f_\theta(x)] + \lambda \nabla_\phi H[q_\phi] && \text{(A.2)} \\
&= \mathbb{E}_{q_\phi}[f_\theta(x)\nabla \log q_\phi(x)] - \lambda \mathbb{E}_{q_\phi}\left[(1+\log q_\phi(x))\nabla_\phi \log q_\phi(x)\right] && \text{(A.3)} \\
&= \lambda \mathbb{E}_{q_\phi}[(f_\theta(x) - \lambda(1+\log q_\phi(x)))\nabla_\phi \log q_\phi(x)] && \text{(A.4)} \\
&= \mathbb{E}_{q_\phi}[w_\lambda(x) \nabla_\phi \log q_\phi(x)], && \text{(A.5)}
\end{aligned}$$

where $w_\lambda(x) := f_\theta(x) - \lambda(1+\log q_\phi(x))$. Using the notation from Section 2.4, the individual components of $\nabla_\phi \log q_\phi(x)$ are given by

$$\begin{aligned}
\frac{\partial}{\partial \phi_{jk}} \log q_\phi(x) &= \frac{\partial}{\partial \phi_{jk}} \log q_{\phi_j}(x^j) && \text{(A.6)} \\
&= \frac{\partial}{\partial \phi_{jk}} \log \frac{e^{\phi_{jx^j}}}{\sum_{l=1}^4 e^{\phi_{jl}}} && \text{(A.7)} \\
&= \frac{\partial}{\partial \phi_{jk}} \phi_{jx^j} - \frac{\partial}{\partial \phi_{jk}} \log \sum_{l=1}^4 e^{\phi_{jl}} && \text{(A.8)} \\
&= \delta_k(x^j) - \frac{1}{\sum_{l=1}^4 e^{\phi_{jl}}} \sum_{l=1}^4 \frac{\partial}{\partial \phi_{jk}} e^{\phi_{jl}} && \text{(A.9)} \\
&= \delta_k(x^j) - \frac{e^{\phi_{jk}}}{\sum_{l=1}^4 e^{\phi_{jl}}} && \text{(A.10)} \\
&= \delta_k(x^j) - q_{\phi_j}(k). && \text{(A.11)}
\end{aligned}$$

Using Equation A.6 in Equation A.2 gives a expression for the components of the gradient in the SGD algorithm (Eq. A.1):

$$\frac{\partial}{\partial \phi_{jk}} F(\phi) = \mathbb{E}_{q_\phi}\left[w_\lambda(x)\left(\delta_k(x^j) - q_{\phi_j}(k)\right)\right]. \tag{A.12}$$

The expectation in Equation A.12 cannot be solved exactly, so we used a Monte Carlo approximation:

$$\frac{\partial}{\partial \phi_{jk}} F(\phi) \approx \frac{1}{M} \sum_{i=1}^{M} w_\lambda(x_i) \left(\delta_k(x_i^j) - q_{\phi_j}(k) \right), \quad x_i \sim q_\phi(x), \tag{A.13}$$

where M is the number of samples used. In the experiments in Section 2.5, we used $M = 1000$ samples to perform library optimizations for $2,238$ different settings of λ.

In practice, the nucleotide sequences sampled from q_ϕ must be translated to amino acid sequences before being passed to the predictive model f, which is a model trained to predict log-enrichment from amino acid sequence. For notational simplicity, we have omitted this translation step from the above equations.

A.8 Comparison of constructed libraries via effective number of variants

Statistical entropy is closely related to another notion of diversity known as *effective sample size*: the effective sample size of a library with entropy H is defined as $N_e = e^H$ and is equal to the number of unique variants required to construct a library with entropy H under the constraint that equal probability mass is placed on each variant. This can be seen by noting that $H = \log N_e = -\sum_{i=1}^{N_e} \frac{1}{N_e} \log \frac{1}{N_e}$. This interpretation of statistical entropy is commonly used in the population genetics literature, first introduced by Wright in 1931 [110].

In Section 2.5, we were able to compare designed theoretical library distributions by computing the statistical entropy of each exactly in terms of its position-wise nucleotide probabilities. However, when analyzing post-selection libraries, there is no known underlying probability distribution for which we can exactly compute entropy. Consequently, we, instead, estimated and compared the effective sample size of the empirically observed distribution in each post-selection library. Specifically, we estimates the effective number of variants in a library using the observed sequencing data,

$$N_e = \exp\left(-\sum_{s \in S} p_{\text{empirical}}(s) \log p_{\text{empirical}}(s) \right), \tag{A.14}$$

where $p_{\text{empirical}}(s)$ corresponds to the empirical frequency of the sequence s appearing in the post-selection sequencing data, S.

A.9 Supplementary results

Library D2	A	T	C	G
Position 1	0.12	0.04	0.39	0.45
2	0.18	0.47	0.3	0.05
3	0.21	0.19	0.28	0.32
4	0.14	0.02	0.19	0.65
5	0.23	0.33	0.29	0.15
6	0.28	0.24	0.25	0.23
7	0.35	0	0.14	0.51
8	0.13	0.17	0.36	0.34
9	0.21	0.31	0.31	0.17
10	0.13	0	0.06	0.81
11	0.26	0.12	0.22	0.4
12	0.16	0.29	0.36	0.19
13	0.09	0	0.08	0.83
14	0.36	0.12	0.37	0.15
15	0.13	0.49	0.24	0.14
16	0.22	0	0.13	0.65
17	0.29	0.08	0.24	0.39
18	0.1	0.42	0.34	0.14
19	0.16	0.01	0.09	0.74
20	0.28	0.11	0.47	0.14
21	0.17	0.35	0.3	0.18

Library D3	A	T	C	G
Position 1	0.21	0.09	0.43	0.27
2	0.22	0.25	0.37	0.16
3	0.25	0.28	0.27	0.2
4	0.27	0.12	0.13	0.48
5	0.2	0.35	0.27	0.18
6	0.22	0.23	0.22	0.33
7	0.22	0.08	0.16	0.54
8	0.32	0.19	0.34	0.15
9	0.25	0.17	0.27	0.31
10	0.24	0.07	0.43	0.26
11	0.34	0.35	0.2	0.11
12	0.22	0.26	0.28	0.24
13	0.28	0.06	0.17	0.49
14	0.45	0.14	0.29	0.12
15	0.2	0.27	0.3	0.23
16	0.32	0.08	0.19	0.41
17	0.23	0.2	0.35	0.22
18	0.23	0.27	0.36	0.14
19	0.2	0.04	0.32	0.44
20	0.39	0.17	0.3	0.14
21	0.26	0.25	0.24	0.25

Table A.2: Marginal nucleotide probabilities of machine learning-designed library distributions D2 and D3.

Library D2	A	T	C	G
Position 1	0.13	0.05	0.40	0.42
2	0.18	0.50	0.27	0.05
3	0.20	0.21	0.27	0.32
4	0.14	0.03	0.18	0.65
5	0.22	0.36	0.28	0.14
6	0.27	0.27	0.24	0.22
7	0.35	0	0.14	0.51
8	0.13	0.19	0.36	0.32
9	0.20	0.34	0.30	0.16
10	0.13	0	0.06	0.81
11	0.27	0.14	0.21	0.38
12	0.16	0.31	0.35	0.18
13	0.09	0	0.08	0.83
14	0.37	0.14	0.35	0.14
15	0.12	0.53	0.22	0.13
16	0.22	0	0.12	0.66
17	0.30	0.10	0.24	0.36
18	0.1	0.45	0.33	0.14
19	0.16	0.01	0.09	0.74
20	0.29	0.14	0.44	0.14
21	0.16	0.41	0.28	0.18

Library D3	A	T	C	G
Position 1	0.22	0.10	0.42	0.25
2	0.22	0.29	0.34	0.15
3	0.24	0.32	0.25	0.19
4	0.26	0.14	0.12	0.48
5	0.19	0.39	0.25	0.17
6	0.21	0.26	0.21	0.32
7	0.22	0.09	0.15	0.54
8	0.32	0.22	0.32	0.14
9	0.24	0.20	0.25	0.31
10	0.25	0.08	0.41	0.26
11	0.33	0.38	0.18	0.11
12	0.21	0.29	0.26	0.24
13	0.28	0.07	0.16	0.49
14	0.45	0.16	0.27	0.12
15	0.19	0.31	0.28	0.22
16	0.32	0.09	0.18	0.41
17	0.23	0.24	0.33	0.20
18	0.22	0.30	0.35	0.13
19	0.2	0.04	0.32	0.44
20	0.38	0.19	0.29	0.14
21	0.25	0.30	0.23	0.22

Table A.3: Marginal nucleotide probabilities of synthesized D2 and D3 libraries approximated using deep sequencing. 193,228 and 212,388 total sequencing reads were assessed for library D2 and D3, respectively.

Library	Condition	Number of Filtered Reads	Number of Variants	Effective Number of Variants
NNK	Pre-packaging	46,046,268	6,439,964	1,391,453
Library D2	Pre-packaging	32,906,886	2,730,606	1,325,880
Library D3	Pre-packaging	58,980,102	4,438,600	1,852,644
NNK	Post-packaging	45,303,374	2,326,627	14,774
Library D2	Post-packaging	37,940,372	1,603,734	232,221
Library D3	Post-packaging	54,340,339	1,670,527	71,958
NNK	Post-brain infection	152,436,128	4,113,029	3,541
Library D2	Post-brain infection	147,317,910	5,021,387	38,350

Table A.4: Number of total sequencing reads (after filtering based on mismatches in primer sequences; see Section 2.5), number of unique variants, and effective number of variants in each sequencing pool. Each row corresponds to a sequencing pool, which is indicated by the Library and Condition columns.

Libraries	Condition	Number of Common Variants	Percent Common Variants
NNK and Library D2	Pre-packaging	4,772	0.17
NNK and Library D3	Pre-packaging	17,899	0.40
Library D2 and Library D3	Pre-packaging	111,556	4.09
NNK, Library D2, and Library D3	Pre-packaging	1,016	0.04
NNK and Library D2	Post-packaging	5,035	0.31
NNK and Library D3	Post-packaging	8,327	0.50
Library D2 and Library D3	Post-packaging	41,263	2.57
NNK, Library D2, and Library D3	Post-packaging	1,095	0.07

Table A.5: Analysis of overlap between sequence pools. Each row represents two or three pools of sequences indicated in the 'Libraries' and 'Condition' columns, and the two rightmost columns show the number and percentage of common variants that appear in all of the row's pools.

Library	Condition	Number of Filtered Reads	Number of Variants	Effective Number of Variants
NNK	Pre-packaging	32,906,886	5,311,946	1,336,116
Library D2	Pre-packaging	32,906,886	2,730,606	1,325,880
Library D3	Pre-packaging	32,906,886	3,387,033	1,783,804
NNK	Post-packaging	37,940,372	2,014,332	14,651
Library D2	Post-packaging	37,940,372	1,603,734	232,221
Library D3	Post-packaging	37,940,372	1,262,483	71,217
NNK	Post-brain infection	147,317,910	4,000,976	3,537
Library D2	Post-brain infection	147,317,910	5,021,387	38,350

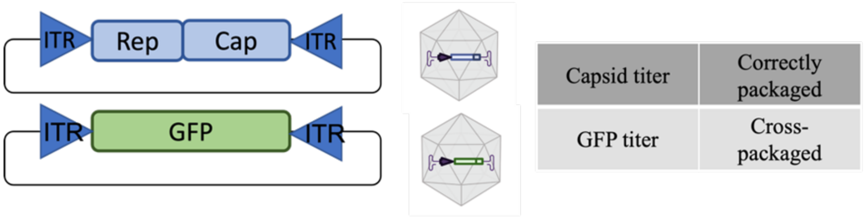
Table A.6: Library diversity analysis with artificially equalized sequencing depth. This table displays equivalent analyses as Table A.4, when each of the pre-packaging, post-packaging, and post-brain infection sequencing pools for Library D2, D3 are subsampled to have exactly the same number of total reads as the corresponding pool of NNK prior analysis.

Insertion Sequence	Experimental Titer (vg/μL)
VTNVVRA	2.84×10^{12}
KVSNAAN	5.97×10^{12}
VVKQRGD	9.14×10^{12}

Table A.7: Experimental packaging titers (viral genome (vg)/μL) for glia-infecting AAV variants selected from machine learning-design library D2.

Library	Capsid titer (vg/mL)	GFP titer (vg/mL)
NNK	1.12×10^{11}	2.43×10^{9}
Library D2	6.81×10^{11}	8.96×10^{9}
Library D3	2.39×10^{11}	5.23×10^{9}

Table A.8: Quantification of cross-packaging in three AAV insertion libraries: NNK, Library D2, and Library D3.



Capsid titer	Correctly packaged
GFP titer	Cross-packaged

Figure A.1: Schematic illustrating the use of green fluorescent protein (GFP) to identify cross-packaging.

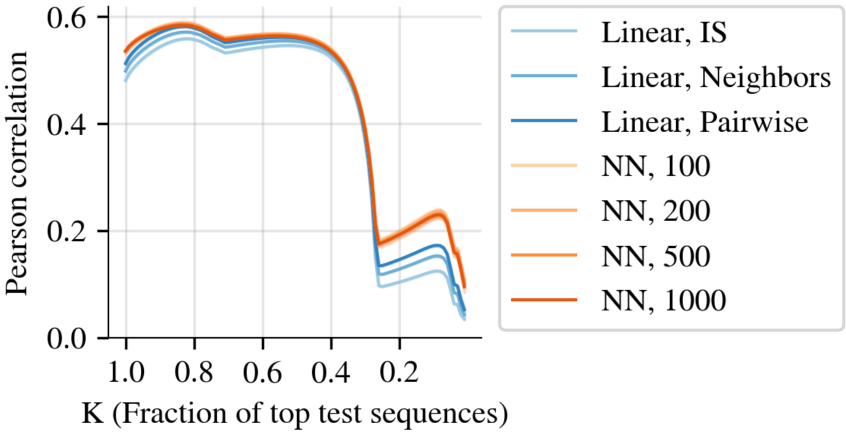


Figure A.2: Comparison of models for predicting AAV5 7-mer insertion packaging log-enrichment in the same format as Figure 2.5: models are compared using “top-K” Pearson correlation between predicted and observed log-enrichment, where K denotes what fraction of top test sequences according to observed log-enrichment are used to compute correlation. Curves and error bands are computed using three-fold cross-validation.

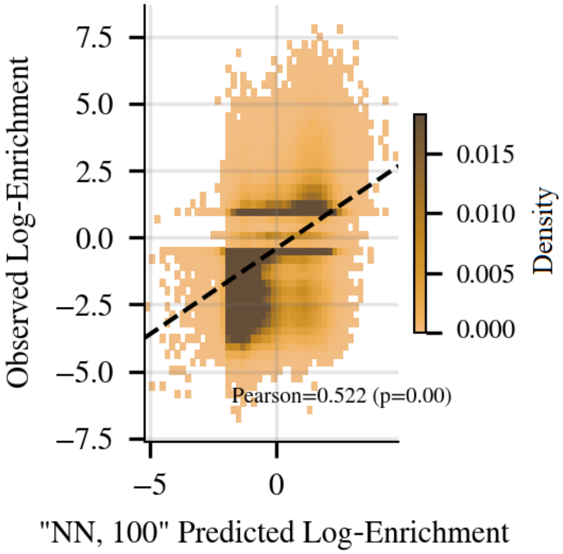


Figure A.3: Paired plot comparing predicted and observed log-enrichment of each sequence in the held-out test set for the 100-unit neural network (NN) model.

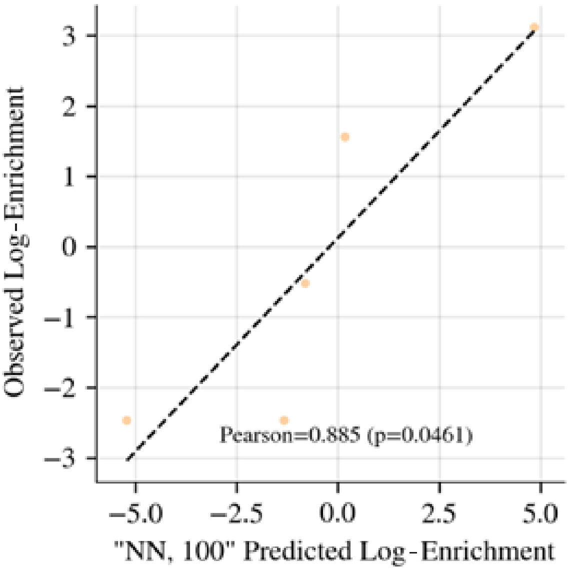


Figure A.4: Correlation between predicted and observed log-enrichment for five variants whose predicted log-enrichment values are near the quartiles of all predictions on the test set. This choice of variants mimics the process of choosing the variants that were tested in Figure 2.6.

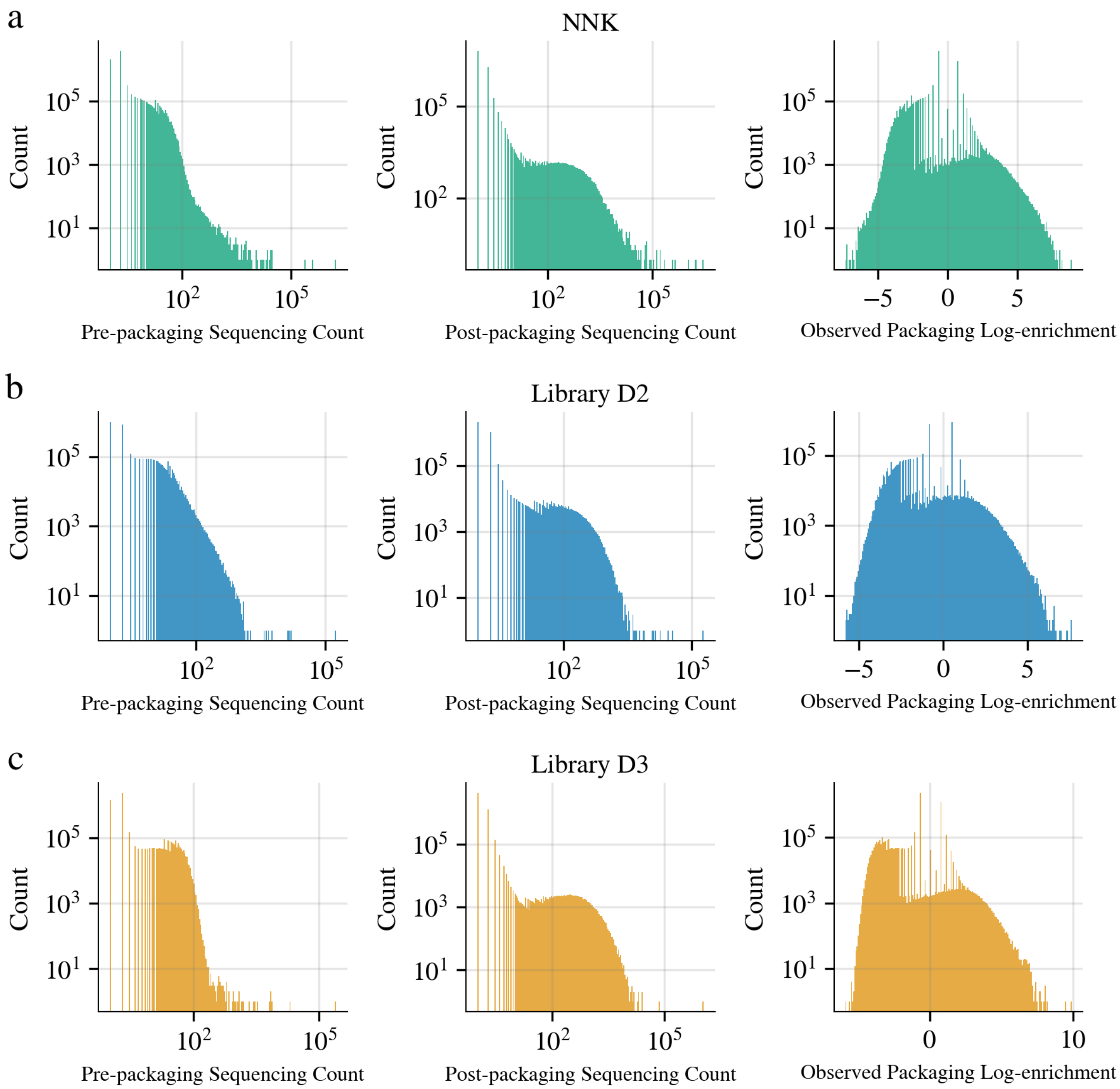


Figure A.5: Histograms showing the empirical distributions of read counts and count-based log-enrichment estimates in the (a) NNK, (b) D2, and (c) D3 libraries. The first and second columns of histograms show the distribution of pre- and post-packaging read counts, respectively. The final column shows the distribution of observed log-enrichment estimates.

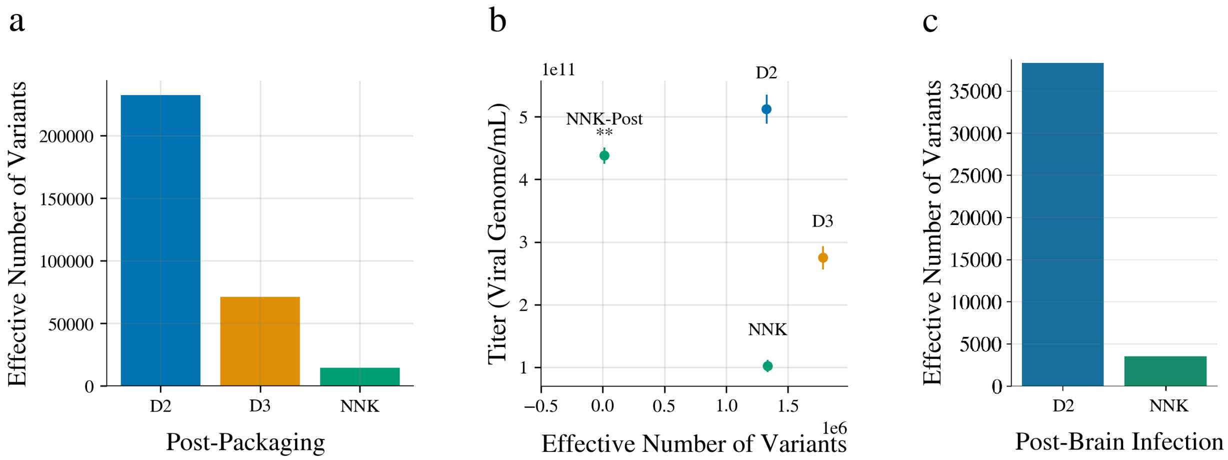


Figure A.6: Library diversity analysis with artificially equalized sequencing depth. Analyses are equivalent to (a) Figure 2.9a, (b) Figure 2.9c, and (c) Figure 2.10b when each of the pre-packaging, post-packaging, and post-brain infection sequencing pools for Library D2 and Library D3 are subsampled to have exactly the same number of total reads as the corresponding NNK sequencing pool prior to analysis.

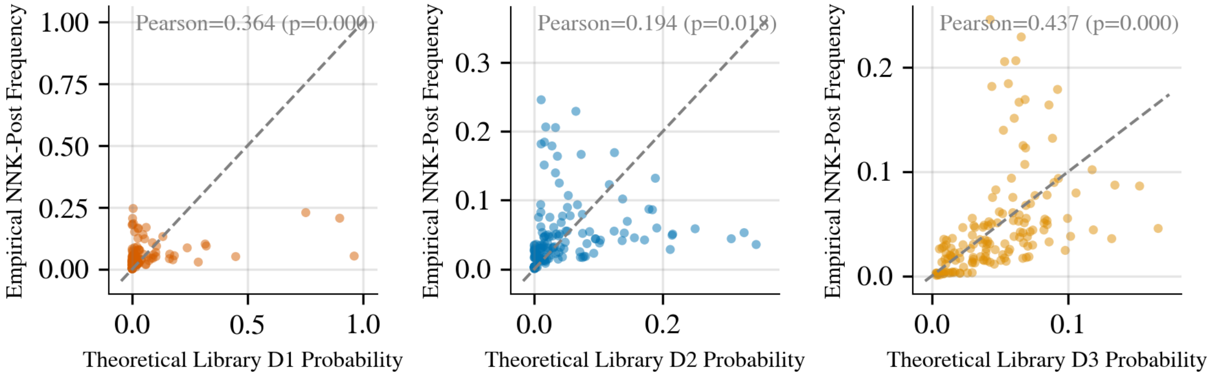


Figure A.7: Comparison between empirical frequencies of amino acids in the post-packaged NNK library and amino acid probabilities in designed libraries (left) D1, (center) D2, and (right) D3. Each point represents a unique amino acid at one of the seven insertion positions: its position on the vertical axis represents the empirical frequency of the specified amino acid at that position in the post-packaged NNK library, while its position on the horizontal axis represents the probability of the specified amino acid at that position in the relevant designed library.

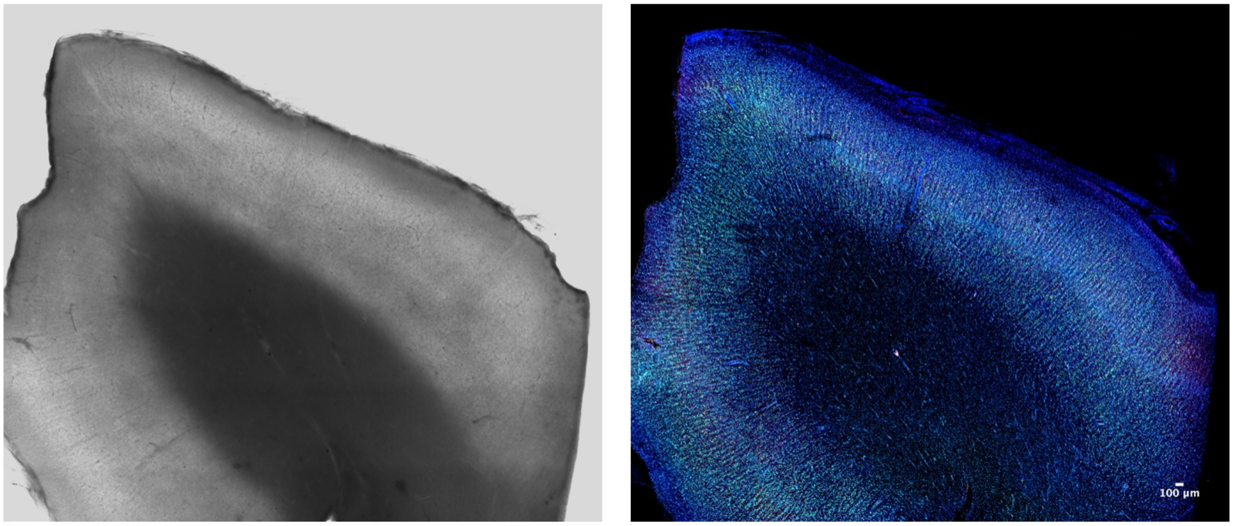


Figure A.8: Characterization of primary adult human brain section in culture. Brightfield (left); Immunostaining (right): DAPI (blue), Nissl (Cyan), and NeuN (magenta). The middle area is white matter, and the surrounding area is grey matter.

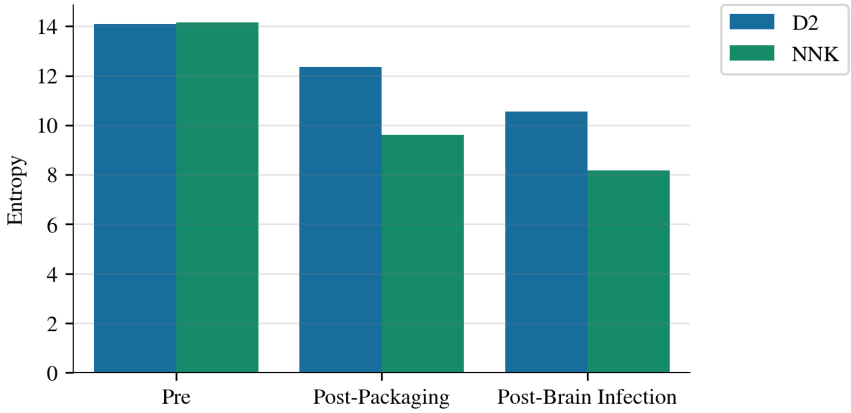


Figure A.9: Comparison of diversity (entropy) between synthesized NNK and machine learning-designed D2 libraries before and after packaging and infection of primary adult human brain tissue. Library D2 present a comparable level of initial diversity (pre) to that of the NNK library, but has substantially higher diversity than the NNK library after both packaging (post-packaging) and primary human brain infection (post-brain infection).

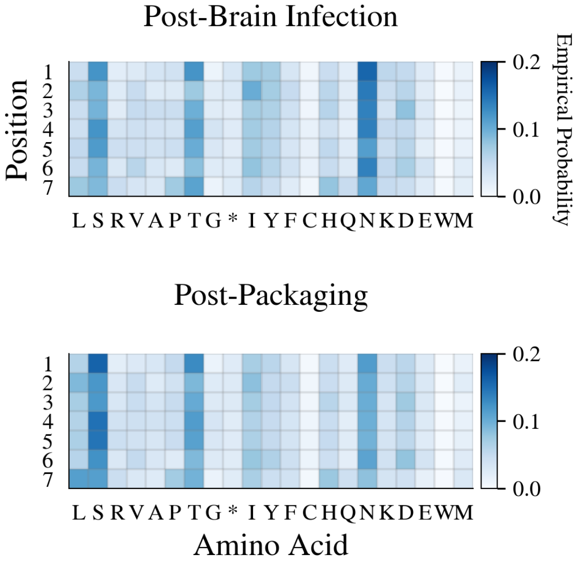


Figure A.10: Empirical marginal probabilities of amino acids at each insertion position in NNK library after packaging and primary brain infection based on deep sequencing data.

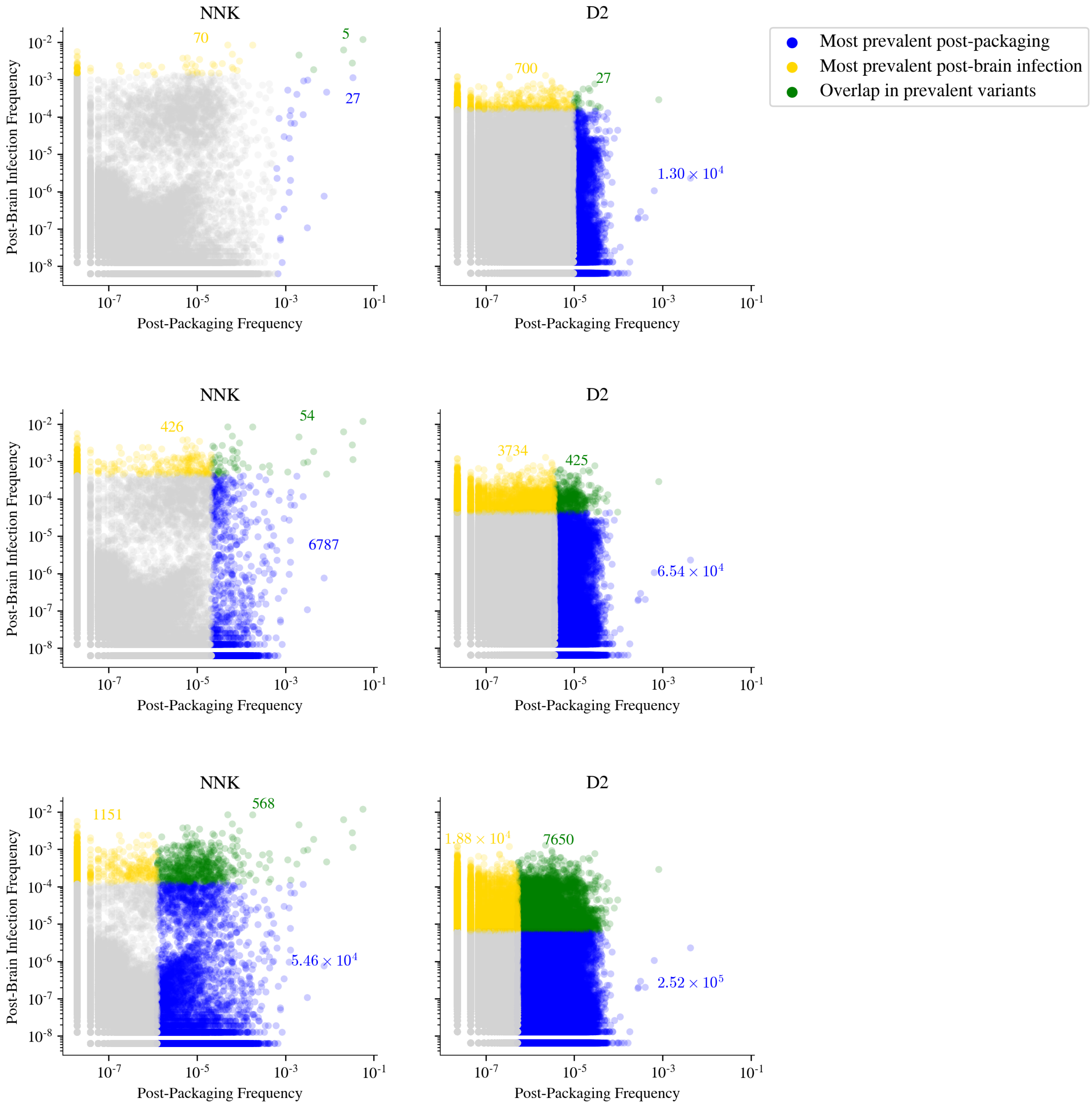


Figure A.11: Scatterplots illustrating the behavior of individual variants in the NNK and machine learning-designed D2 libraries over packaging and primary human brain infection selections. Each row is identical to Figure 2.10d except for the top percentage of reads that is assigned colors. From top to bottom, variants that are in the top 20%, 50%, and 80% of total reads are assigned colors.

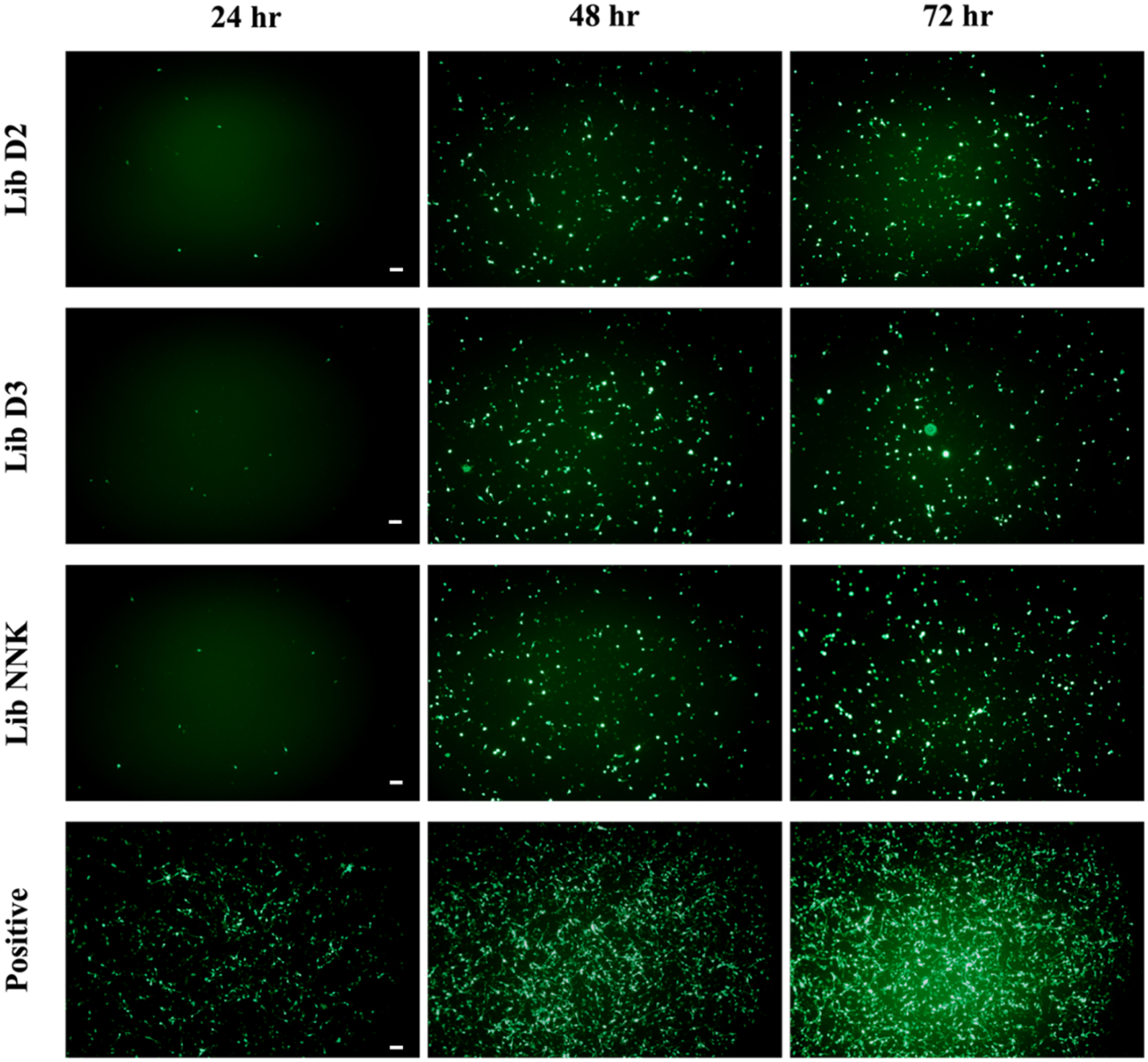


Figure A.12: GFP fluorescence signals across three libraries (Library D2, Library D3, and NNK) with positive control (100% GFP-ITR plasmid) at 24, 48, and 72 hours after transfection in HEK293T cells (Scale bar = 40 μm). During early times (24 hours after transfection), the amount of GFP signal is similar across libraries and is significantly less than the positive control.

Appendix B

Supplementary Information: A New Model-Based Enrichment Approach for High-Throughput Sequencing Experiments

B.1 Multi-output modeling

In Chapter 3, we presented weighted log-enrichment regression (wLER) and model-based enrichment (MBE) methods for estimating and predicting log-enrichment given sequencing data from two conditions. In practice, one often aims to compare sequences across more than two conditions. For example, one may wish to perform multiple rounds of selection for a property of interest (*e. g.,* [35]) or to select for multiple different properties (*e. g.,* [62]). Here, we describe generalizations of the MBE and wLER approaches that can be used to model high-throughput sequencing data collected from more than two conditions.

In the multi-condition setting, one has sequencing data $\mathcal{D}'' = \{(r_i, y_i)\}_{i=1}^{M}$ where r_i is the i^{th} read's sequence and y_i is a categorical label indicating the condition from which the read r_i arose. For example, if one runs an experiment selecting for $k \in \mathbf{N}$ different properties, one can define $y \in \{0, 1, \ldots, k\}$ where $y_i = 0$ indicates a read from the pre-selection sequencing data, and $y_i = j$ for $j \in \{1, \ldots, k\}$ indicates a read from the post-selection sequencing data for the j^{th} property.

It is straightforward to handle multiple conditions using the MBE approach: instead of using a binary classifier, one trains a multi-class classification model, g_θ, to predict the categorical label y_i from read sequence r_i using a standard categorical cross-entropy loss. This produces a model of $p(y \mid r)$ which can be used to estimate the density ratios $d^j = \frac{p^j}{p^0} \approx \frac{N^0 g_\theta^j}{N^j g_\theta^0}$, where p^j denotes the true probability distribution corresponding to the library in the j^{th} condition and g_θ^j denotes the predicted class probability for $y = j$.

For the wLER approach, the data $\mathcal{D}''$ can be converted into cLE estimates for each unique sequence:

$$\log e_i^j = \log \left(\left(\frac{n_i^j}{N^j} \right) \left(\frac{n_i^0}{N^0} \right)^{-1} \right),$$

where n_i^j is the number of times the sequence x_i appeared in the sequencing data for the j^{th} condition and N^j is the total number of reads from the j^{th} condition. One can, then, fit a multi-output regression model that jointly predicts the cLE estimates for each condition from sequence. The overall loss for training a such a multi-output model, f_θ, using wLER is

$$\sum_{j=1}^{k} \sum_{i=1}^{M'} w_i^j (\log e_i^j - f_\theta^j(x_i))^2$$

where w_i^j is the weight for the i^{th} sequence and j^{th} condition, and f_θ^j denotes the j^{th} model output.

B.2 Asymptotic optimality of model-based enrichment

In this section, we review key parametric convergence results which imply that, under the assumption of a correctly specified parametric model, the proposed model-based enrichment (MBE) estimator is optimal among a broad class of semi-parametric density ratio estimators—including the weighted log-enrichment regression (wLER) method [116]—in terms of asymptotic variance.

We begin by recalling some notation: let p^A and p^B be two probability distributions, $d = \frac{p^B}{p^A}$ be their density ratio, and

$$\mathcal{D} = \{(r_i, y_i)\}_{i=1}^{M} \tag{B.1}$$

be a dataset of observed samples where y_i is a binary label indicating whether the sample r_i is from p^A ($y_i = -1$) or p^B ($y_i = +1$). Further, let N^A and N^B be the number of samples from p^A and p^B, respectively. Recall that the MBE approach uses logistic regression to learn a classifier that predicts $p(y_i \mid r_i)$, and these predicted class probabilities give an estimate of the density ratio (Online Methods). In other words, the MBE approach estimates the density ratio using the parametric model

$$\log d_\theta(r) = \theta_0 + \phi_{\theta_1}(r) \tag{B.2}$$

where $\theta_0 \in \mathbf{R}$, $\theta = (\theta_0, \theta_1) \in \mathbf{R}^b$ is a b-dimensional parameter, and ϕ_{θ_1} is a real-valued function (*e. g.,* defined by the choice of model architecture).

Whenever correctly-specified density models for both p^A and p^B are unavailable, direct density ratio estimation of $\frac{p^B}{p^A}$—as performed by the MBE approach—is preferable compared to separate density estimation of p^A and p^B in terms of asymptotic unnormalized

Kullback–Leibler divergence to the true density ratio, d [90]. Moreover, Qin [71] showed that, if the logistic regression model is correctly specified—that is, if the true density ratio d is realized by d_{θ^*} in the parametric model—then the MBE approach is optimal among a large class of semi-parametric density ratio estimators in the sense that it has the smallest asymptotic variance. Specifically, the class of semi-parametric estimators in Qin's analysis is a class of generalized moment-matching estimators:

$$\begin{aligned}\{\hat{\theta}_\eta \ \mid \eta_\theta(r) \in \mathbb{R}^b, &\operatorname{Var}_{p^A}[\eta_\theta(r)] \text{ and } \operatorname{Var}_{p^B}[\eta_\theta(r)] \text{ are finite,} \\ &\frac{1}{N^A}\sum_{(r_i,y_i)\in\mathcal{D}} \eta_{\hat{\theta}_\eta}(r_i) d_{\hat{\theta}_\eta}(r_i)\mathbb{1}\{y_i=-1\} = \frac{1}{N^B}\sum_{(r_i,y_i)\in\mathcal{D}} \eta_{\hat{\theta}_\eta}(r_i)\mathbb{1}\{y_i=+1\}\}.\end{aligned}$$

This class of estimators contains several popular density ratio estimators, including the Kullback-Leibler (KL) importance estimation procedure [90, 91] that learns a density ratio model by minimizing empirical KL divergence between $d \cdot p^A$ and p^B. Other estimation techniques, including weighted and non-linear least squares regression, can also be cast in terms of generalized moment-matching optimization [58] and, therefore, the wLER approach is included in Qin's class of estimators, as are several other existing log-enrichment regression approaches [12, 70]. Thus, under a correctly specified parametric model, the MBE approach is the preferred density ratio estimation technique—and, in the context of this work, the preferred technique for quantifying sequences based on sequencing data from a high-throughput screen or selection—in terms of asymptotic variance.

B.3 Simulating ground truth fitness

Recall from Section 3.4 that we constructed several simulated datasets to help analyze the strengths and weaknesses of MBE, wLER, and cLE across different practical settings. These simulations were motivated by high-throughput selection experiments [116, 82, 62] which perform a selection on large sequence libraries for a property of interest, such as fluorescence [82]. To simulate such selection experiments, we first simulate the ground truth fitness function that maps sequence to property, then use this fitness to simulate selection. In the remainder of this section, we describe the process used to simulated fitness as a linear function of independent amino acid sites and randomly selected higher-order epistatic interactions. In Section B.4, we describe the procedure to simulate selection using this simulated fitness.

First, we give a brief overview of the process used to simulate ground truth fitness before providing the technical details. For a given sequence of interest, we first constructed a set containing all independent amino acid sites and a user-specified number of combinations of sites—such as an epistatic combination of the second, third, and tenth positions—drawn randomly from among all possible higher-order epistatic interactions between positions. The degree of each epistatic effect (2 up to the sequence length) is drawn randomly based on an empirical estimate of this degree distribution. The fitness function is, then, taken to be a linear function of all the independent sites and epistatic terms in this constructed set with random coefficients.

In more detail, for a sequence x of length L_a amino acids, we simulated the fitness function, $F_T(x)$ as

$$F_T(x) = \sum_{J \in \mathcal{E}_T} \beta_J \cdot \phi(x[J]), \tag{B.3}$$

where T is the hyper-parameter controlling the maximum number of epistatic terms included in F_T; $\mathcal{E}_T \subseteq 2^{\{1,\ldots,L_a\}}$, is a set of index sets—each of which represents an independent site or a particular higher-order epistatic combination—whose construction is described below; $x[J]$ is the subsequence of x at the positions in the index set J; ϕ denotes standard one-hot encoding; and the coefficients are sampled according to $\beta_J \sim \mathcal{N}(\mathbf{0}, 2^{-|J|}\mathbf{I})$.

We constructed $\mathcal{E}_T$ (the specific set of first-order and higher-order epistatic terms to include in the simulated fitness function) to contain all singleton sets $(\{\{i\} \mid i \in \{1, \ldots, L_a\}\} \subseteq \mathcal{E}_T)$, so that F_T includes terms for all independent sites. In addition, $\mathcal{E}_T$ contains T randomly-chosen non-singleton index sets, each generated by:

1. randomly choosing the order of epistasis, R, by sampling $\tilde{R} \sim \mathrm{N}(3, 1/2)$ (based on visual inspection of the empirical bell-shaped distribution of the orders of statistically significant epistatic terms in Poelwijk *et al.*. [70]), and taking $R = \mathrm{round}(\tilde{R})$; and

2. choosing the specific positions included in the epistatic term by sampling R times without replacement from $\{1, \ldots, L_a\}$.

To guide our choice of T, we combined the following insights: (i) for a fluorescent protein with 13 amino acids, 260 epistatic terms are sufficient for an accurate model of fitness [70]; (ii) the number of contacts in a protein scales linearly with sequence length [6, 98]; and (iii) recent work suggests that the sparsity of higher-order epistatic interactions in fitness landscapes is closely related to structural contact information [11]. We, therefore, hypothesized that the linear scaling $T = \frac{260 L_a}{13}$ provides a reasonable starting point for analyses.

B.4 Simulating pre- and post-selection sequencing data

The wLER and MBE approaches both aim to accurately quantify sequences of interest based on high-throughput sequencing data. In Section 3.5, we used simulated high-throughput selection datasets to compare each method's ability to quantify sequences accurately using sequencing data, which requires simulating sequencing reads from pre- and post-selection libraries. Here, we detail the process of simulating sequencing reads given library sequences and a ground truth fitness function. Then, in Section B.5, we describe how we combined this process with three specific approaches for simulating library sequences to construct our datasets.

Let $\{(x_i, c_i)\}_{i=1}^{M'}$ be pairs of, respectively, a unique library sequence and its true count—as generated, for example, by one of the three library construction simulations described in the subsequent sections. In addition, let F_T be a ground truth fitness function simulated as in the previous section. Briefly, the process to simulate sequencing reads from pre- and post-selection libraries proceeds as follows: first, we generate a pre-selection library distribution by adding a small random perturbation to the empirical distribution $\left\{c_i / \sum_{i=1}^{M'} c_i\right\}_{i=1}^{M'}$. This step simulates slight distributional perturbations that may occur with PCR amplification, and also has the nice side-effect of allowing one to generate multiple replicates with slightly different pre- and post-selection library distributions for the same set of unique sequences $\{x_i\}_{i=1}^{M'}$. Next, we simulate selection according to the fitness F_T: the post-selection library distribution is determined by scaling the pre-selection distribution using $\{\exp(F_T(x_i))\}_{i=1}^{M'}$, which ensures that the ground truth log-density ratio is proportional to the specified fitness $\left(\log d = \log \frac{p^{\text{post}}}{p^{\text{pre}}} \propto F_T\right)$. Finally, we sample from the pre- and post-selection distributions to simulate sequencing reads, optionally truncating each read to 100 amino acids uniformly at random to generate short reads.

In more detail, we simulated pre- and post-selection sequencing data by:

1. sampling $(p^{\text{pre}}(x_i))_{i=1}^{M'} \sim \text{Dirichlet}(c_1, \ldots, c_{M'})$;
2. setting
$$p^{\text{post}}(x_i) = Z \exp(F_T(x_i)) p^{\text{pre}}(x_i)$$
where $Z = \sum_{i=1}^{M'} \exp(F_T(x_i)) p^{\text{pre}}(x_i)$ is a normalization constant;
3. sampling pre- and post-selection sequencing counts according to
$$\begin{aligned} (n_i^{\text{pre}})_{i=1}^{M'} &\sim \text{Multinomial}(N^{\text{pre}}, (p^{\text{pre}}(x_i))_{i=1}^{M'}) \quad \text{and} \\ (n_i^{\text{post}})_{i=1}^{M'} &\sim \text{Multinomial}(N^{\text{post}}, (p^{\text{post}}(x_i))_{i=1}^{M'}) \end{aligned}$$
for some desired number of sequencing reads, N^{pre} and N^{post}; and, if simulating short reads, additionally
4. sampling n_i^{pre} and n_i^{post} contiguous 100-mers from x_i uniformly at random.

B.5 Simulated dataset details

To empirically compare and contrast our MBE approach to the wLER approach in practical settings, we sought to simulate realistic sequence libraries motivated by experimental constructions from recent studies. Here, we describe three specific approaches for simulating library constructions, and detail how we used each to simulate datasets from high-throughput selection experiments.

Peptide insertion libraries

We simulated diversified libraries of peptide insertion sequences motivated by our work in adeno-associated virus (AAV) capsid engineering (Chapter 2). Recall from Section 2.5 that, in this study, we used a library of 21-mer nucleotide insertion sequences, where each codon was independently sampled from the distribution defined by the NNK degenerate codon: "NN" denotes a uniform distribution over all four nucleotides in the first two positions of a codon and "K" denotes equal probability on nucleotides G and T in the third codon position. Here, we sampled sequences from this NNK distribution to simulate three insertion libraries containing length 21, 150, and 300 nucleotide sequences, respectively. Specifically, each sequence is generated by sampling either 7, 50, or 100 codons independently from the NNK distribution. To keep each of our simulated insertion datasets as similar as possible to the experimental data from Chapter 2, we sampled sequences in this manner until we obtained a set of 8.5×10^6 unique library sequences. We take the set $\{(x_i, c_i)\}_{i=1}^{8.5\times 10^6}$ to be the simulated library, where x_i is the i^{th} unique insertion sequence and c_i is the number of times it was sampled from the NNK distribution before 8.5×10^6 unique sequences were generated. We used $T = 140, 1000$, and 2000 to simulate ground truth epistatic fitness for the 21-mer, 150-mer, and 300-mer insertion libraries, respectively, and simulated $N^{\text{pre}} = N^{\text{post}} = 4.6 \times 10^7$ sequencing reads for each library using the process described in the previous Section B.4.

To gain insight into the effect of sequencing error on MBE and wLER, we also constructed a noisy version of the sequencing data for the 21-mer insertion library containing simulated sequencing errors in both the pre- and post-selection sequencing reads. Because Illumina's next-generation sequencers have an approximately 1% error rate and predominantly produce substitution errors [24], we added substitution errors to each position of each simulated read uniformly at random with probability 0.01.

avGFP mutagenesis library

Motivated by a recent study of the fitness landscape of the green fluorescent protein from *Aequorea victoria* [82], we generated an avGFP library by mutating positions of the avGFP reference sequence from Sarkisyan et al. [82] (238 amino acids long) uniformly at random. We used a mutation rate of 10% to generate 2.5×10^7 unique library sequences. Specifically, we generated mutated avGFP sequences—by mutating each position independently with probability 0.01—until we obtained a set $\{(x_i, c_i)\}_{i=1}^{2.5\times 10^7}$, where each x_i a unique library sequence and c_i is the number of times it was generated before 2.5×10^7 unique sequences were obtained.

To simulate selection and sequencing, we used $T = 4,760$ to simulate ground truth fitness, and generated both long-read ($N^{\text{pre}} = N^{\text{post}} = 4.6 \times 10^5$ to be within PacBio's throughput [38, 74]) and short-read ($N^{\text{pre}} = N^{\text{post}} = 4.6 \times 10^7$ to match the dataset from Chapter 2) sequencing data.

AAV recombination library

We simulated a recombination library of AAV capsid sequences motivated by an AAV directed evolution study [62], wherein several AAV serotypes are recombined using seven crossovers separating eight recombination blocks. We generated library sequences by recombining AAV serotypes 1-9 with seven uniformly-spaced crossovers. This library contains 26,873,856 unique library sequences that are 2,253 nucleotides long. We simulated epistatic fitness with $T = 15,020$.

To assess the effects of the type and amount of sequencing data, we generated multiple datasets: three long-read datasets with $N^{\text{pre}} = N^{\text{post}} = 4.6 \times 10^3$, 4.6×10^4, and 4.6×10^5, respectively; one short-read dataset with $N^{\text{pre}} = N^{\text{post}} = 4.6 \times 10^7$; and one hybrid dataset containing 4.6×10^3 long reads and 4.5×10^7 short reads for both pre- and post-selection. To help gain insights into the effects of sequencing error, we also constructed a noisy AAV recombination dataset that incorporated simulated sequencing errors into the 4.6×10^5 pre- and post-selection sequencing reads using SimLoRD [89] to simulate PacBio SMRT sequencing errors.

B.6 Supplementary results

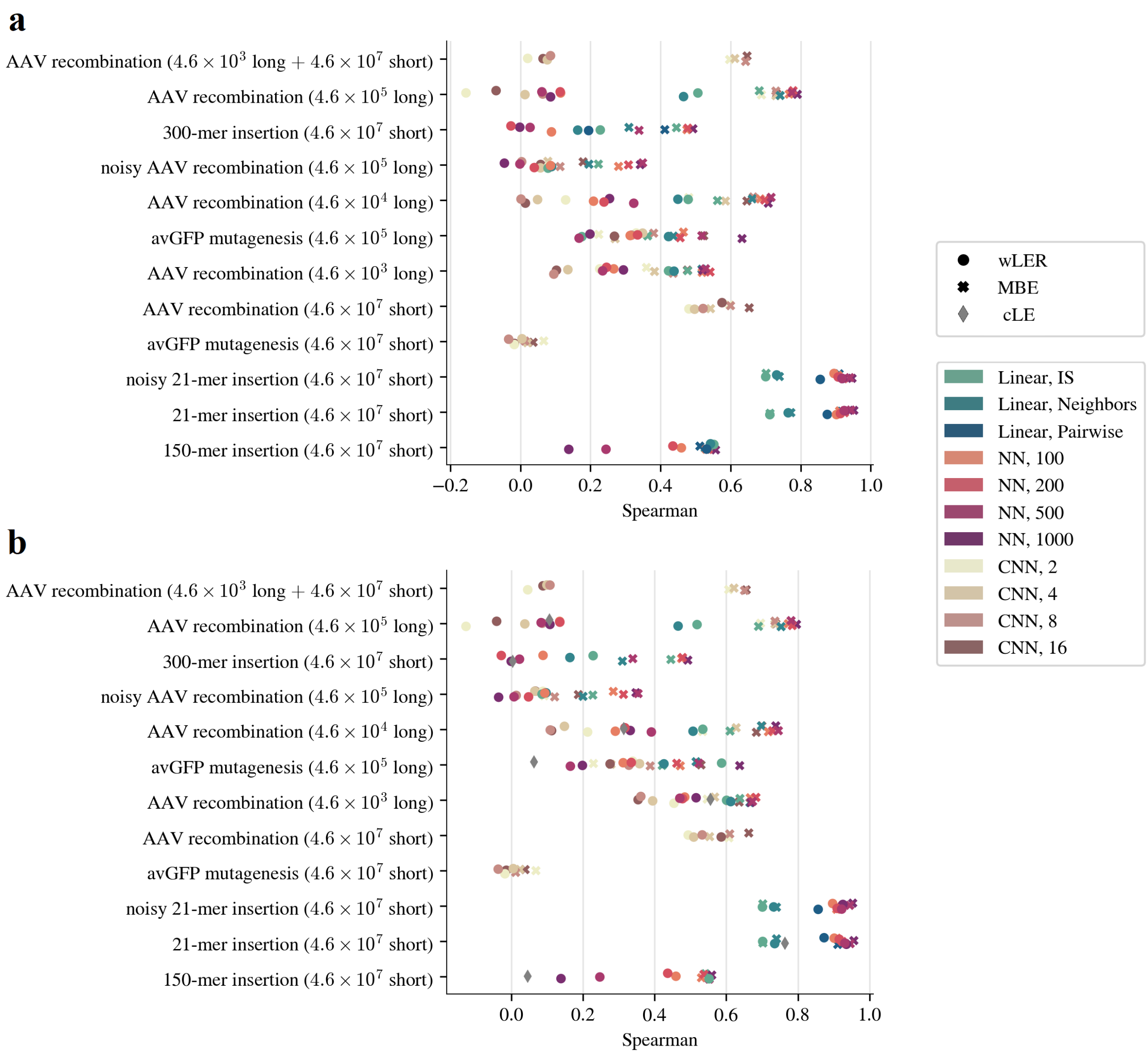


Figure B.1: Simulation results for all model architectures. (a) and (b) are the same as Fig. 3.2a and b, respectively, but display the Spearman correlation between model predictions and ground truth fitness for all model architectures.

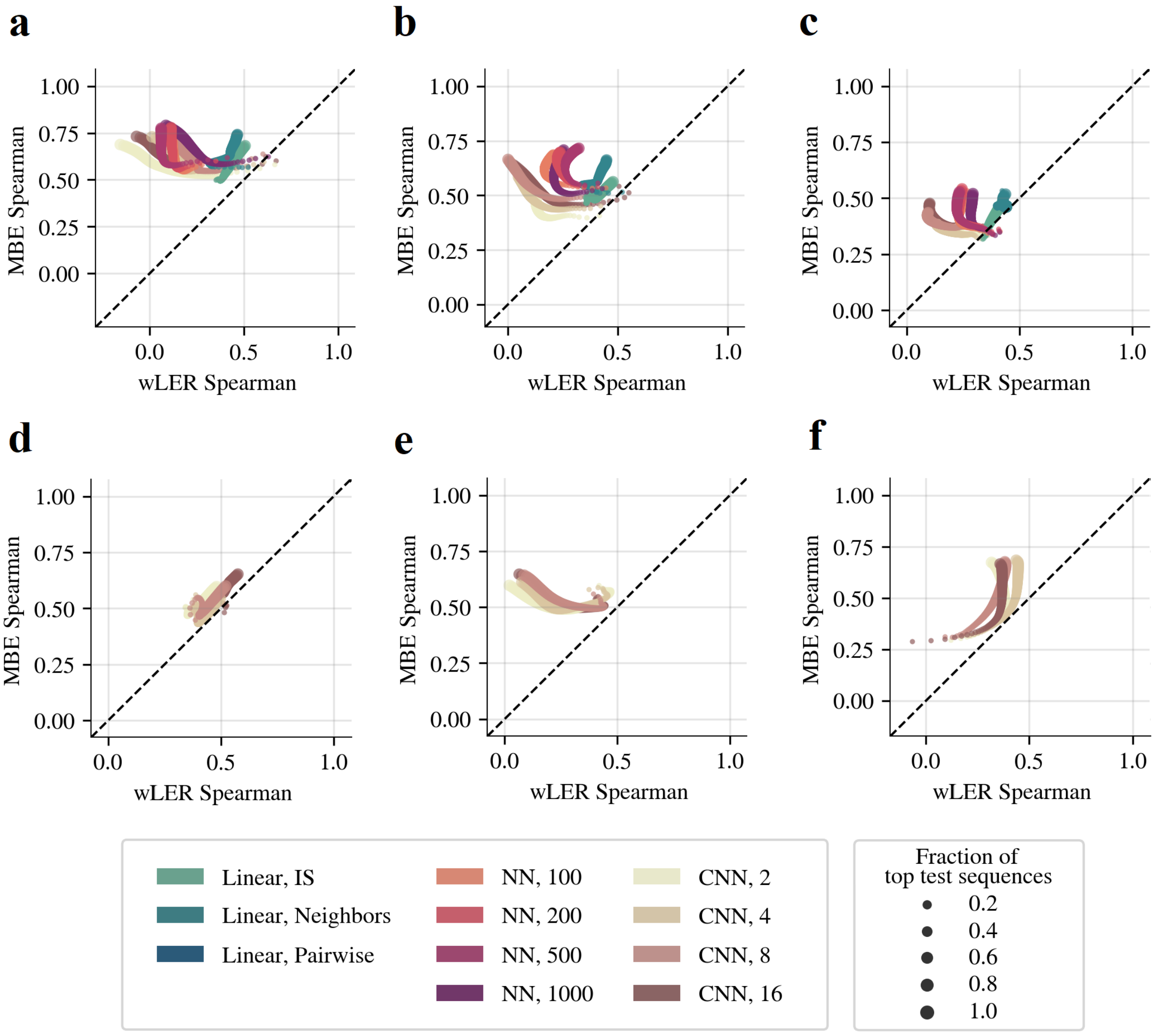


Figure B.2: Simulated library results with increasing long read sparsity and short reads. Compares Spearman correlation between simulated ground truth fitness and wLER or MBE predictions on held-out sequences of interest when models are trained using the simulated AAV recombination datasets with (a) 4.6×10^5 long reads, (b) 4.6×10^4 long reads, (c) 4.6×10^3 long reads, (d) 4.6×10^7 short reads, and (e) a combination of 4.6×10^3 long and 4.6×10^7 short reads, and (f) the avGFP mutagenesis dataset with 4.6×10^7 short reads. Each panel compares the Spearman correlation achieved by the wLER and MBE approaches using the same model architecture and hyper-parameters. Dot size represents the fraction of test sequences with highest ground truth fitness used to compute Spearman correlation. Only CNNs are included in d-f since the linear and NN models cannot operate on variable-length sequences.

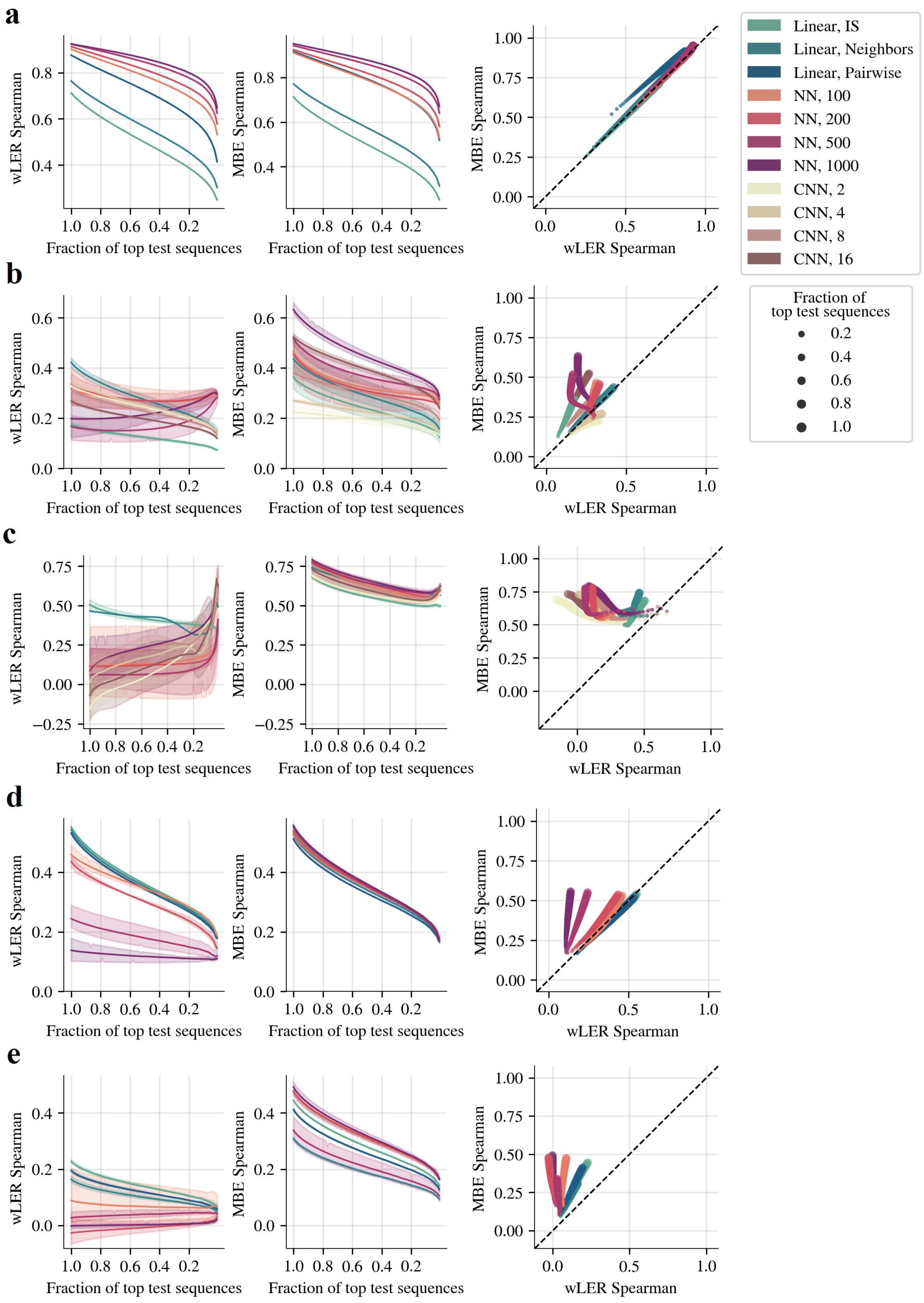


Figure B.3: Comparison of generalized Spearman correlation between simulated ground truth fitness and wLER or MBE predictions on held-out sequences for the simulated (a) 21-mer insertion (4.6×10^7 short reads), (b) avGFP mutagenesis (4.6×10^5 long reads), (c) AAV recombination (4.6×10^5 long reads), (d) 150-mer insertion (4.6×10^7 short reads), and (e) 300-mer insertion (4.6×10^7 short reads) datasets. In each row, the left panel displays the performance of wLER for each model architecture, the center panel is the same as the left panel MBE, and the rightmost panel is a paired plot version of the left and center panels.

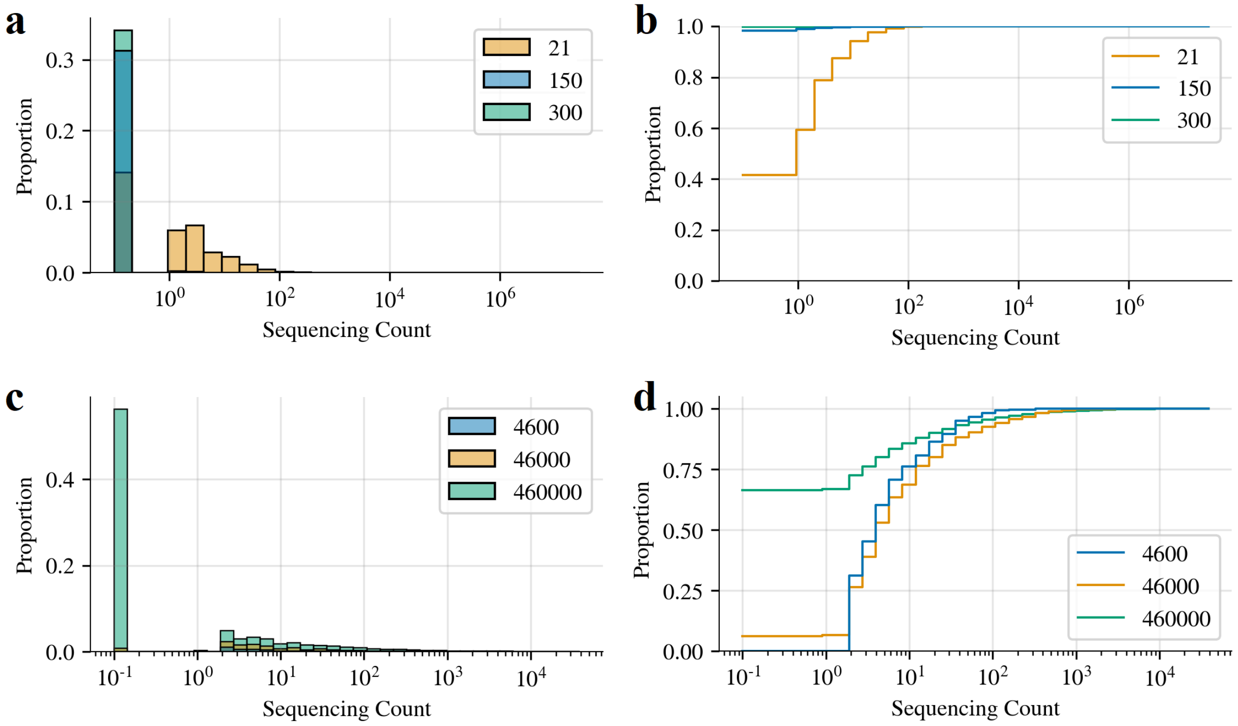


Figure B.4: Sequencing count histograms for simulated insertion and recombination libraries. Histogram (left) and cumulative histogram (right) of simulated post-selection sequencing counts for the (a-b) 21-mer, 150-mer, and 300-mer insertion datasets, and (c-d) AAV recombination datasets with 4.6×10^5, 4.6×10^4, and 4.6×10^3 long reads.

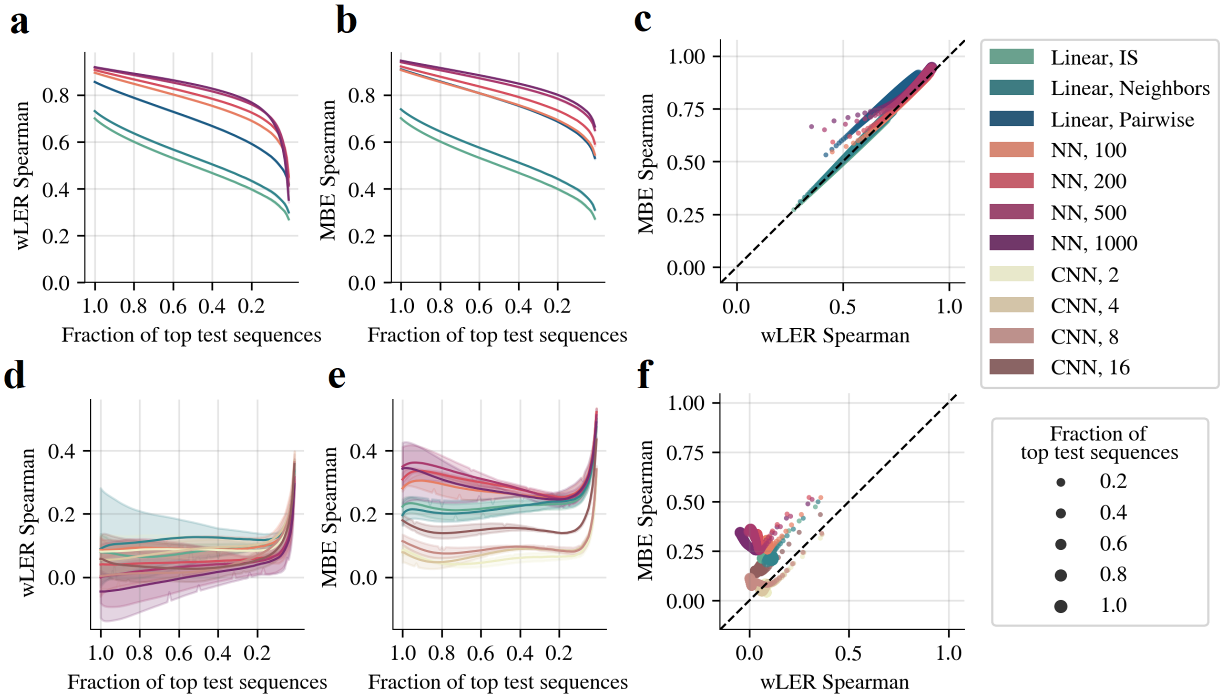


Figure B.5: Generalized Spearman for prediction with simulated sequencing errors. Comparison of the Spearman correlation between simulated ground truth fitness and wLER or MBE predictions on held-out full-length library sequences when models are trained using the simulated (a-c) noisy 21-mer insertion (4.6×10^7 short reads) and (d-f) noisy AAV recombination (4.6×10^5 long) datasets. The noisy 21-mer insertion dataset includes substitution errors added to the training set at a uniform error rate of 1%, consistent with Illumina's next-generation sequencers [24]. The noisy AAV recombination dataset contains simulated PacBio SMRT sequencing errors added to the training set using SimLoRD [89]. In each row, the leftmost panel compares the performance of wLER for each model architecture, the center panel is the same as the left panel for MBE, and the rightmost panel is a paired plot version of the left and center plots.

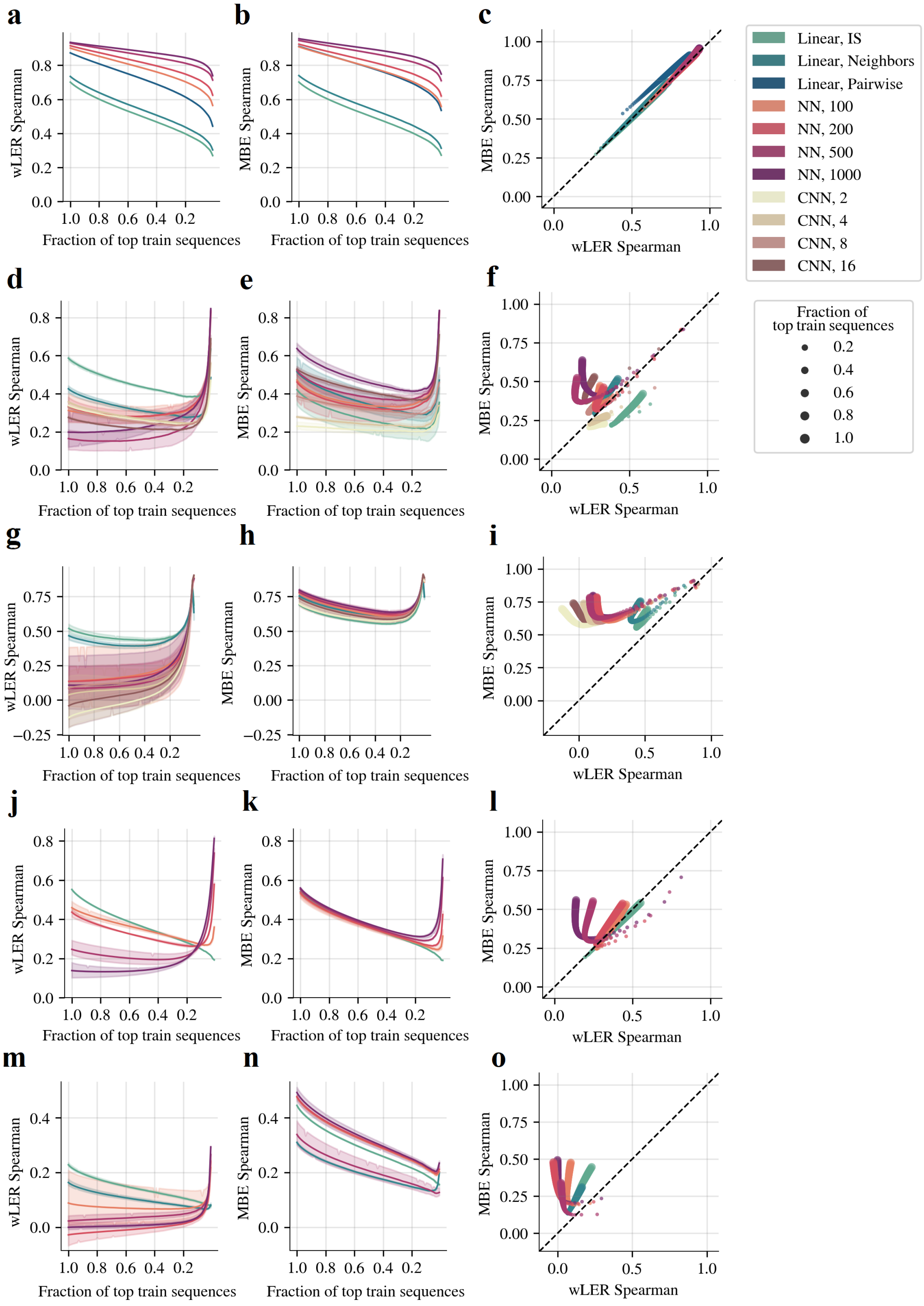


Figure B.6: Comparison of the correlation between simulated ground truth fitness and wLER or MBE estimates for sequences observed during training for the simulated (a-c) 21-mer insertion (4.6×10^7 short reads), (d-f) avGFP mutagenesis (4.6×10^5 long reads), (g-i) AAV recombination (4.6×10^5 long reads), (j-l) 150-mer insertion (4.6×10^7 short reads), and (m-o) 300-mer insertion (4.6×10^7 short reads) datasets. In each row, the left panel compares the performance of wLER for each model architecture, the center panel is the same as the leftmost panel for MBE, and the right panel is a paired plot version of the left and center plots.

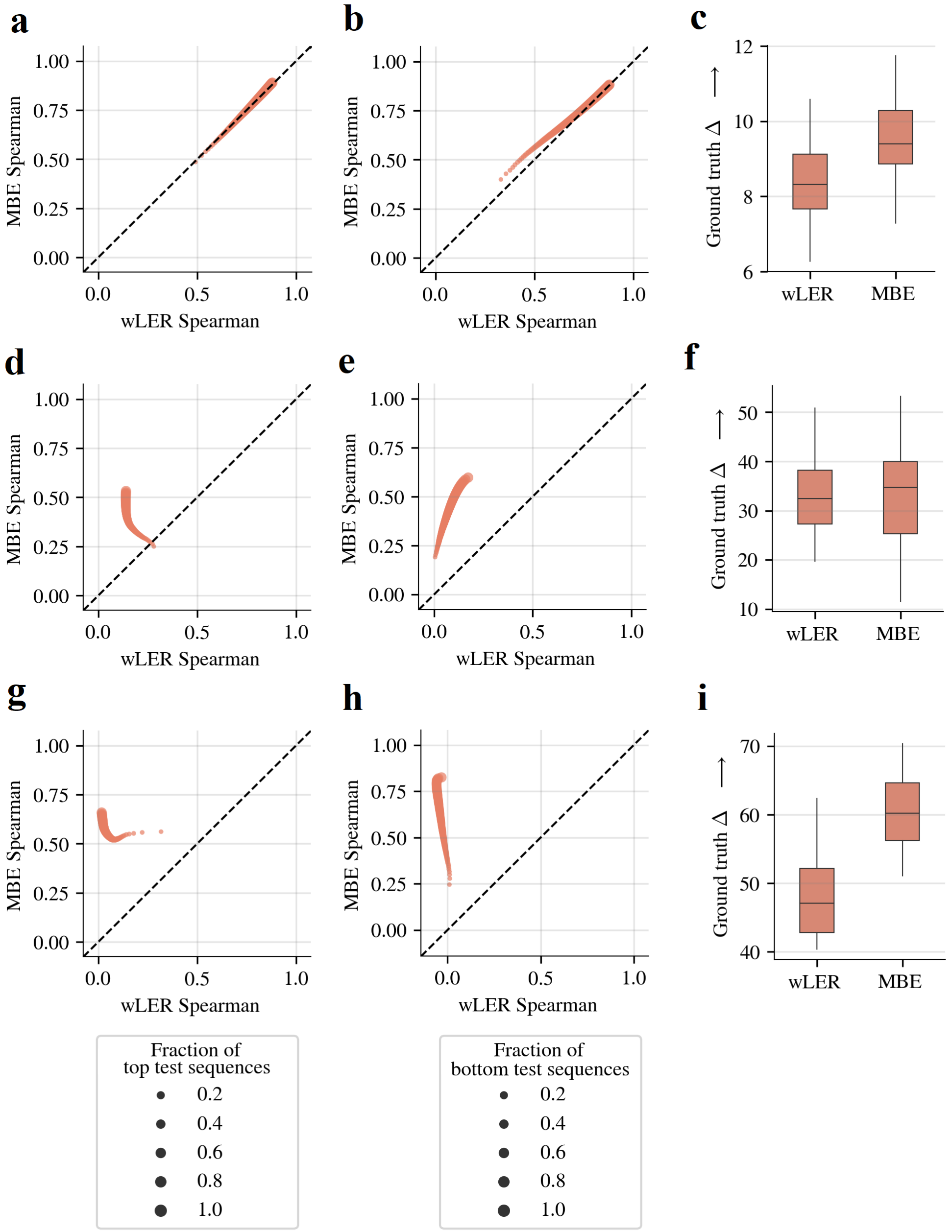


Figure B.7: Simulated positive, negative, and selectivity selection results. Comparison of wLER and MBE on (left) prediction for sequences with high ground truth positive fitness, (center) prediction for sequences with low ground truth negative fitness, and (right) identification of selective sequences for the simulated (a-c) 21-mer insertion (4.6×10^7 short reads), (d-f) avGFP mutagenesis (4.6×10^5 long reads), and (g-i) AAV recombination (4.6×10^5 long reads) datasets. For positive fitness, dot size represents the fraction of top test sequences according to highest ground truth fitness. For negative fitness, dot size represents the fraction of test sequences with lowest ground truth fitness. In each row, the rightmost panel displays ground truth selectivity (the difference between positive and negative fitness values, Δ) for the top ten test sequences according to each model's predicted selectivity (the difference between predicted fitness values) for each of the three test folds.

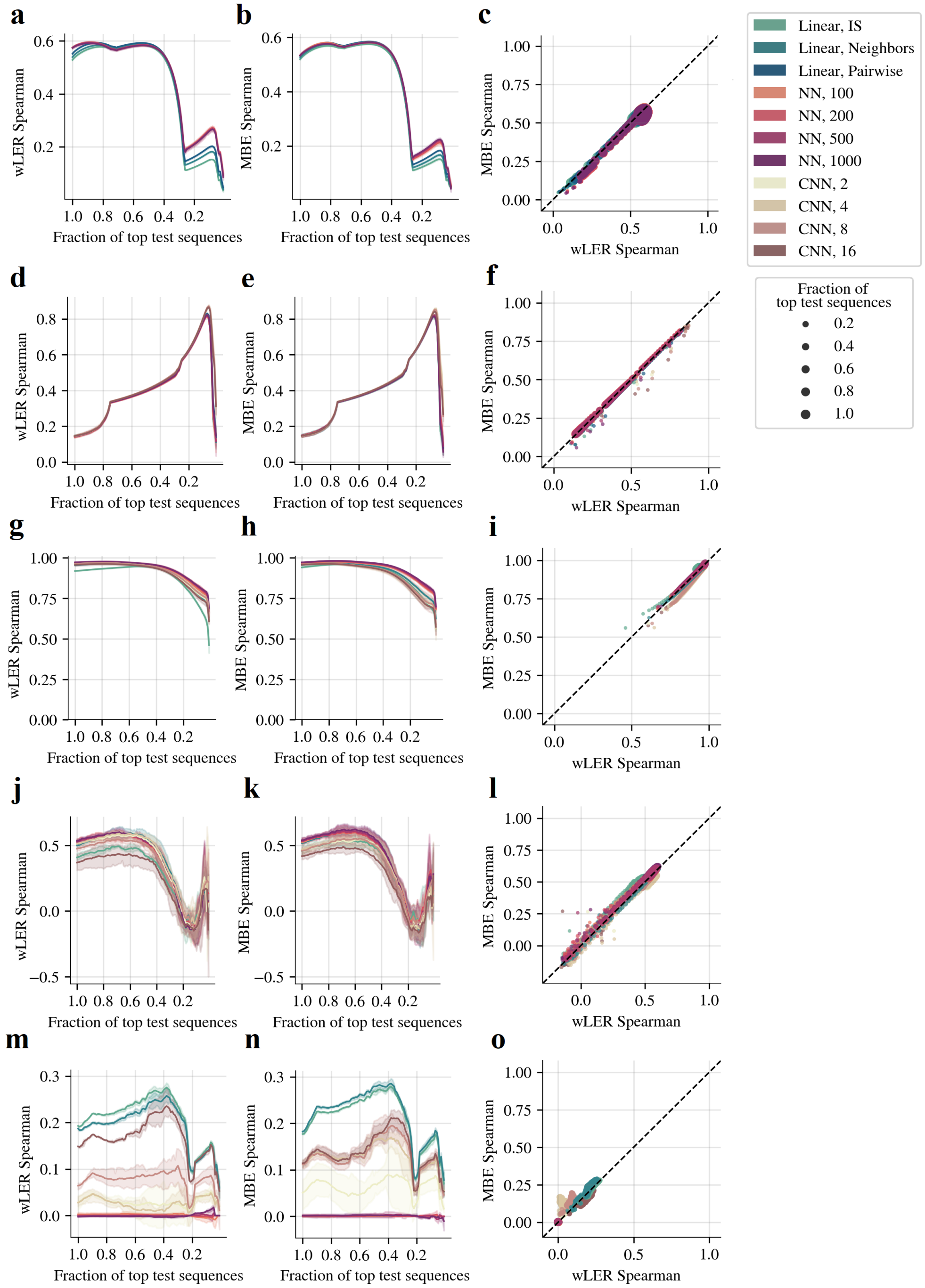


Figure B.8: Comparison of the Spearman correlation between wLER or MBE predictions and observed cLE estimates on experimental sequencing datasets from the (a-c) AAV5 insertion library from Zhu *et al.* [116], (d-f) SARS-CoV-2 tiled peptide library from Huisman *et al.* [35], (g-i) GB1 double site saturation mutagenesis library from Olson *et al.* [63], (j-l) chorismate mutase homolog library from Russ *et al.* [80], and (m-o) Bgl3 random mutagenesis library from Romero *et al.* [78]. In each row, the left panel compares the performance of wLER for each model architecture restricted to a given top fraction of test sequences with highest observed cLE, the center panel is the same as the left panel for MBE, and the right panel is a paired plot version of the left and center panels.

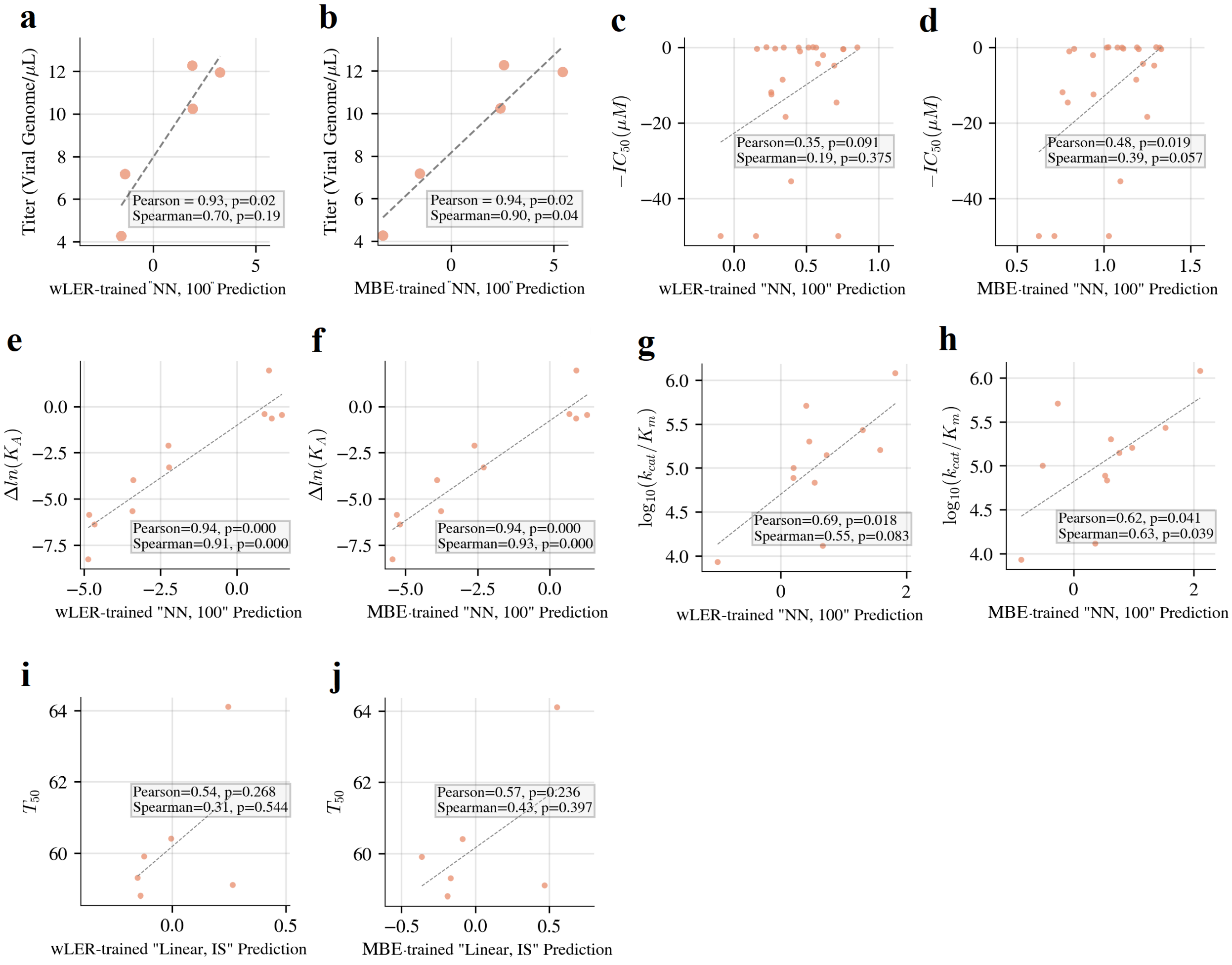


Figure B.9: Low-throughput experimental property measurement predictions. Comparison of wLER and MBE predictions and experimental property measurements from (a-b) Zhu *et al.* [116] (packaging titer), (c-d) Huisman *et al.* [35] (IC_{50}, half maximal inhibitory concentration), (e-f) Olson *et al.* [63] ($\Delta\ln(K_A)$, change in log-binding constant), (g-h) Russ *et al.* [80] ($\log_{10}(k_{cat}/K_m)$, log-second-order reaction rate constant), and (i-j) Romero *et al.* [78] (T_{50}, temperature where half of the protein is inactivated in ten minutes).

Table B.1: Comparison of Spearman correlation between experimental IC_{50} measurements from Huisman *et al.* [35] and wLER predictions, MBE predictions, or reported NetMHCIIpan4.0 predictions from Huisman *et al.* [35]. The 100-unit NN architecture is used for the wLER and MBE methods.

	Spearman	p-value
MBE	0.394	0.057
wLER	0.190	0.375
NetMHCIIpan4.0 %Rank	0.275	0.193