Accelerated Learning on Large-Scale Screens Using Generative Library Models

Eli N. Weinstein[⋆,1,2] **Andrei Slabodkin**[⋆,1] **Mattia G. Gollub**[⋆,1] **Elizabeth B. Wood**[1,†]
1. JURA Bio, Boston, MA, USA. 2. Technical University of Denmark, Kgs. Lyngby, DK.

Abstract

Biological machine learning is often bottlenecked by a lack of scaled data. One promising route to relieving data bottlenecks is through high-throughput screens, which can experimentally test the activity of $10^6 - 10^{12}$ protein sequences in parallel. In this article, we introduce algorithms to optimize high throughput screens for data creation and model training. We focus on the large-scale regime, where dataset sizes are limited by the cost of measurement and sequencing. We show that when active sequences are rare, we maximize information gain if we *only* collect positive examples of active sequences, i.e. x with $y > 0$. We can correct for the missing negative examples using a generative model of the library, producing a consistent and efficient estimate of the true $\mathrm{p}(y \mid x)$. We demonstrate this approach in simulation and on a large-scale screen of antibodies. Overall, co-design of experiments and inference lets us accelerate learning dramatically.

Machine learning holds dramatic potential for advancing biological discovery, with growing success in designing proteins, diagnosing genetic disease, predicting pathogen evolution and more [Watson et al., 2023, Frazer et al., 2021, Thadani et al., 2023]. But these and other potential applications are often bottlenecked by a lack of large-scale, high quality training data. In particular, there is insufficient information about biological sequences' activity and function.

One promising route to relieving these data bottlenecks is through large-scale experiments. Chemistry and biology provide tools to synthesize sequences, deliver them into cells, and measure their functional activity. Large-scale screens can test $10^6 - 10^{12}$ cells in parallel, each with a different sequence design. The result is a dataset relating sequences, x, to their activity, y. The challenge is that with large numbers of cells, dataset sizes become limited by the cost of measurement, i.e. the cost of recovering x and y from individual cells.

In this article, we introduce methods to optimize these large-scale screens for dataset creation and model training. We focus specifically on a common biological setting where we can sort cells, and then allocate measurements based on their activity. Rather than take a random sample, we propose a strategy to allocate a limited experimental budget to maximally informative measurements. We use the theory of Bayesian experimental design to analyze tradeoffs and optimal resource allocation. We develop a measurement strategy and inference approach that can accelerate model training dramatically, extracting orders of magnitude more information than standard approaches. The technique is particularly well-suited to the hardest learning and design problems, where sequences with a desired activity are exceedingly rare.

1 PROBLEM SETUP & APPROACH

Our goal is to learn a mapping from biological sequences x to functional activity y. To do so, we will synthesize different sequences x as DNA, express them as protein, and measure their activity y. Then we will train a model that predicts y from x. The question is how to accomplish this data generation and model training at scale, such that we obtain a high quality predictor at the end. We consider the following experimental approach (Figure 1a).

1. Synthesize sequences x from a distribution, $\mathrm{p}(x)$, where $\mathrm{p}(x)$ is specified by some generative sequence model. (This can be done using variational synthesis [Weinstein et al., 2022, 2026].)

⋆ equal contribution. † contact ew@jurabio.com

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

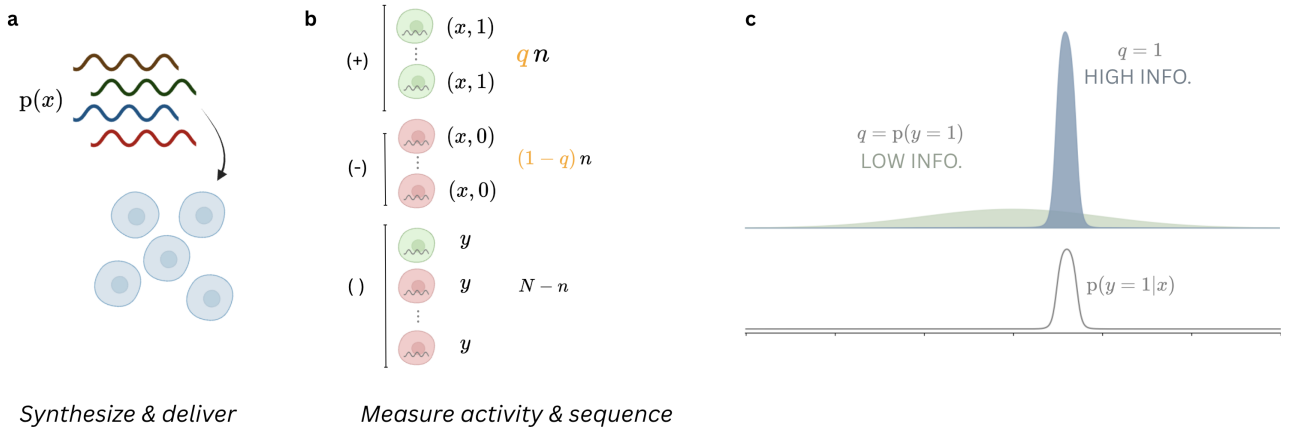


Figure 1: **Overview of LeaVS.** a. In a large-scale screen, sequences x are delivered into cells for testing activity y. Each cell receives a sample $x \sim \mathrm{p}(x)$. b. We can measure (x, y) for some cells, but only observe y for the rest. We can choose the fraction q of measurements to allocate to positive and negative examples. c. If $\mathrm{p}(y \mid x)$ is sparse and we collect a representative sample from the library ($q \approx \mathrm{p}(y = 1)$), most x will not be very informative. Allocating all measurements to positive examples ($q = 1$) yields more information about $\mathrm{p}(y \mid x)$.

2. Deliver these sequences, at random, into N different cells. (E.g. using viral vectors.)

3. Measure sequences x_i and activity y_i across cells i. (To simplify the presentation, we assume that y is binary, with $y = 1$ denoting an active sequence. Section 3.4 generalizes to the case where y is an arbitrary vector.)

The result is a training dataset of x-y pairs, $\{(x_1, y_1), ..., (x_n, y_n)\}$. We can use this data to fit a model of $\mathrm{p}(y \mid x)$, such as a neural network.

This experimental approach can reach very large scale in practice. Variational synthesis can create quadrillions of DNA sequences, each an independent sample from $\mathrm{p}(x)$. Individual sequences can then be delivered into tens of millions of different cells or more.

In this large-scale experimental setting, dataset sizes become limited by the cost of *measurement*, i.e. recovering the x-y pair from each cell. For example, with single-cell sequencing it can be intractable to collect data from more than 10-100 thousand cells. We study efficient learning algorithms and optimal resource allocation in this measurement-limited regime.

Baseline A straightforward approach to collecting data would be to measure a random sample of cells from the population. This provides a dataset of i.i.d. samples from the population distribution, $(x_i, y_i) \overset{iid}{\sim} \mathrm{p}(x, y)$ for $i \in \{1, \ldots, n\}$. We can train models on this data using standard techniques, e.g. optimizing a cross-entropy loss.

(Re)allocating measurement We propose an alternative. We begin from the observation that we typically have some experimental control over which cells we measure, as well as some side information about those cells we do not measure. Many high throughput screening technologies physically separate the initial set of N cells into pools of active and inactive cells, e.g. via a cell sorter (FACS) or pulldown, before sequencing. This has two important practical consequences.

First, we can decide how many active cells to sequence versus how many inactive cells (Figure 1b). That is, if we have a budget for sequencing n cells, we can decide the fraction q to collect with $y = 1$ versus $y = 0$. Then we obtain a dataset $\{(x_1, 1), ..., (x_{nq}, 1), (x_{nq+1}, 0), ..., (x_n, 0)\}$, where $x_i \sim \mathrm{p}(x \mid y_i)$ and $q \in [0, 1]$ is the chosen parameter.

Second, we know the number of cells that are active and inactive. This implies that we have a dataset of y values for the cells we do not measure, $\{y_{n+1}, \ldots, y_N\}$, though their x values are missing.

In short, revealing hidden x_i is costly, but we can choose how to allocate our measurement budget to active vs. inactive sequences.

Contributions First, we introduce a training algorithm that can learn the true sequence-to-activity map $\mathrm{p}(y \mid x)$ under any $q \in [0, 1]$ (Section 3). We provide evaluations to check the accuracy and calibration of the learned model. Second, we show that for biologically plausible conditional distributions $\mathrm{p}(y \mid x)$, we should measure only active sequences. That is, to maximize the information we obtain about $\mathrm{p}(y \mid x)$ for a budget n, we should set $q = 1$ (Section 4). We

demonstrate on simulated and real datasets, with an application to therapeutic antibody discovery. We call the combined experimental design and training approach *LeaVS* (Learning from Variational Synthesis), since it rests on the synthesis distribution $\mathrm{p}(x)$.

2 RELATED WORK

We work in the framework of Bayesian experimental design, aiming to maximize the amount of information we collect under a given budget [Lindley, 1956, Chaloner and Verdinelli, 1995, Rainforth et al., 2024]. We are interested in a setting where prior information is weak and we must make a single experimental design choice before seeing any data. We use asymptotic theory to derive a general decision rule that is applicable under a broad class of biological priors [Zemplenyi and Miller, 2023].

Other recent work in Bayesian experimental design instead focuses on the sequential experimentation setting, and explicitly optimizes a measure of the information gain [Foster et al., 2019, 2020, Huang et al., 2024, Oliveira et al., 2024]. Computationally, this remains intractable for highly parallelized biology experiments, which test millions of datapoints and are modeled with deep neural networks. It also requires an explicit prior, which can be hard to specify for initial experiments.

As in active learning, we are focused on experimental design for training prediction models [Balcan and Urner, 2016, Gal et al., 2017, Smith et al., 2023]. However, standard active learning chooses points x at which to measure y. Here, we have a different experimental design choice, which is about where to measure x rather than y.

Our learning procedure builds on ideas for handling missing data in supervised (deep) learning [Josse et al., 2024, Ipsen et al., 2022]. We marginalize out uncertainty in x. Unusually, many datapoints are missing x in its entirety, not just part of x.

Also related is work on learning from positive and unlabeled data (PU learning). Often, PU learning assumes all the x values are observed, but some of the y values are missing [Elkan and Noto, 2008, Song et al., 2021]. Our situation is the opposite: all the y values are observed, but there are missing x. More closely related is Kiryo et al. [2017], Plessis et al. [2015], Wu [2021], which assume access to samples from $\mathrm{p}(x)$ and from $\mathrm{p}(x \mid y = 1)$, along with knowledge of $\mathrm{p}(y = 1)$. They use a modified empirical risk minimization objective to train a classifier. We make similar assumptions when $q = 1$, but learn using the data's likelihood. This enables Bayesian experimental design, and allows the method to be extended to non-binary y.

[Song et al., 2021] develop PU methods for learning sequence-activity maps from screens. Their method is designed for smaller scale screens, where we deterministically design, synthesize and test each x_i, meaning each x_i is known. We consider larger scale screens, where sequences must be synthesized stochastically, and each cell receives a random sample from the distribution. In this case, many x_i are latent.

In practice in biology, the goal of large-scale screens is often to find active sequences, rather than to learn $\mathrm{p}(y \mid x)$ [e.g. Skora et al., 2015]. Hence, it is standard to only collect positive examples. Indeed, for assays based on selection, the negatives are destroyed. Despite concerns this data is not suitable for machine learning, we show that it is *optimal* in a specific sense.

3 METHOD

3.1 Training

Our goal is to learn a model that predicts y from x, regardless of q. To do so, we consider the joint likelihood of all the available data $\mathcal{D}$, including not just x-y pairs, $\mathcal{D}_{xy}$, but also the y values with missing x, $\mathcal{D}_y$. Let $\mathrm{p}_\theta(y \mid x)$ denote a model parameterized by $\theta \in \Theta$, such as a neural network. The data likelihood $\log \mathrm{p}_\theta(\mathcal{D})$ is

$$\sum_{i=1}^{n} \log \mathrm{p}_\theta(y_i \mid x_i) + \sum_{i=n+1}^{N} \log \int \mathrm{p}_\theta(y_i \mid x)\mathrm{p}(x)dx \quad (1)$$

$$\triangleq \mathcal{L}_{xy}(\theta) + \mathcal{L}_y(\theta) \quad (2)$$

The first term (in blue) is the standard regression likelihood; for binary y, it is the cross-entropy. The second term (in green) is the marginal likelihood of y_i. Since x_i is missing, we integrate over its distribution.

How can we compute $\mathcal{L}_y(\theta)$ in practice? We rely on access to samples from $\mathrm{p}(x)$. In particular, if we are using variational synthesis to produce a large library, we have a generative model specifying the library distribution $\mathrm{p}(x)$ [Weinstein et al., 2022]. We draw samples, $x'_1, ..., x'_M \sim \mathrm{p}(x)$ and compute,

$$\int \mathrm{p}_\theta(y_i \mid x)\mathrm{p}(x)dx \approx \frac{1}{M}\sum_{j=1}^{M} \mathrm{p}_\theta(y_i \mid x'_i). \quad (3)$$

For optimization, we can differentiate the approximation with respect to θ, as $\mathrm{p}(x)$ is independent of θ.

We refer to the log likelihood in Equation (1) as the LeaVS objective, since it incorporates the generative synthesis model $\mathrm{p}(x)$. Intuitively, the LeaVS objective is akin to data augmentation, except instead of giving generated sequences fixed labels, they are used to approximate an integral.

3.2 Learning under Alternative Allocations

The key property of the LeaVS objective is that it allows us to learn $\mathrm{p}(y \mid x)$ under different measurement allocations q. In particular, we propose to set $q = 1$. Here, we summarize the theory in Section 4.

First consider just using data on x-y pairs, $\mathcal{D}_{xy}$, and learning a model based on the likelihood $\mathcal{L}_{xy}(\theta)$. Then we will only learn the correct predictor $\mathrm{p}(y \mid x)$ if we measure a representative sample from the library, i.e. we must set $q = \mathrm{p}(y = 1)$ to obtain a consistent estimate of $\mathrm{p}(y \mid x)$ (Proposition 2).

With the LeaVS objective $\mathcal{L}_{xy}(\theta) + \mathcal{L}_{y}(\theta)$ the situation changes markedly. We can now learn the correct predictor *regardless* of how we allocate measurement resources. That is, $\mathrm{p}_\theta(y \mid x)$ will converge to the true $\mathrm{p}(y \mid x)$ for any $q \in [0, 1]$ (Proposition 1). Notably, this range includes $q = 1$: measuring *only* positive examples. The LeaVS objective automatically accounts for the unobserved inactive sequences, through knowledge of the underlying library distribution $\mathrm{p}(x)$.

In general, the optimal q depends on the situation. But biological sequence-to-activity maps show a recurring feature: only sequences within a narrow region of sequence space $\mathcal{X}$ show non-trivial activity (Figure 1c). So, active sequences provide a large amount of information: they pin down the region's location quite precisely. Negative examples, on the other hand, tell us little about the region's location. As a result, focusing our measurements on positive examples maximizes information gain (Proposition 3).

In summary, we propose to change our experimental design, setting $q = 1$, and change our training objective, adding $\mathcal{L}_y(\theta)$, to accelerate learning of $\mathrm{p}(y \mid x)$.

3.3 Model evaluation

We have shown that it is possible to learn the sequence-to-activity map under different measurement allocations. But for reallocation to be practical, we must also be able to evaluate the models we train, to check their performance. This is non-trivial if we only have access to positive examples, $q = 1$.

Accuracy To evaluate accuracy based on positive examples, we again make use of the library distribution $\mathrm{p}(x)$. Let $t_\theta(x) = \mathbb{I}(\mathrm{p}_\theta(y = 1|x) > 0.5)$ be a classifier based on $\mathrm{p}_\theta(y|x)$. Applying the approach of Kiryo et al. [2017], we decompose the accuracy $\mathrm{A}_\mathrm{p}[t_\theta] = \mathbb{E}_\mathrm{p}[Y t_\theta(X)] + \mathbb{E}_\mathrm{p}[(1 - Y)(1 - t_\theta(X))]$ as

$$\begin{aligned}\mathrm{A}_\mathrm{p}[t_\theta] =& \mathrm{p}(y = 1)\mathbb{E}[t_\theta(X)|Y = 1] + \mathbb{E}_\mathrm{p}[1 - t_\theta(X)] \\ &- \mathrm{p}(y = 1)\mathbb{E}_p[1 - t_\theta(X)|Y = 1]\end{aligned}$$

The first term is the fraction of predictions that are true positives; the second is the fraction of negative predictions; the last is the fraction of false negatives. The first and third terms just depend on $\mathrm{p}(y)$ and $\mathrm{p}(x \mid y = 1)$, so can be approximated using heldout data. The second term depends on $\mathrm{p}(x)$, so can be approximated using the library model (Section A.1).

We can apply the same strategy to approximate false positive and false negative rates, or ROC and precision-recall curves, by plugging different thresholds into $t_\theta(x) = \mathbb{I}(\mathrm{p}_\theta(y|x) > 0.5)$ (Section A.1).

Calibration We would like models that are not only accurate, but also quantify their uncertainty. For example, many algorithms for designing and optimizing sequences require upper confidence bounds. A standard evaluation metric is the expected calibration error (ECE) [Vaicenavicius et al., 2019].

The ECE asks whether, if the model says we should see $Y = 1$ some percentage of the time, then we actually see $Y = 1$ at that percentage. $\mathrm{ECE}_\mathrm{p}[\mathrm{p}_\theta]$ is

$$\mathbb{E}_\mathrm{p}\left[d\left(\mathrm{p}\left(y = 1 \mid \mathrm{p}_\theta(y = 1|X)\right), \mathrm{p}_\theta(y = 1|X)\right)\right] \quad (4)$$

Here $d(\cdot, \cdot)$ is a distance function, such as the absolute difference. It is standard to use a histogram based estimator for the ECE. We define bins $\Phi_1, \ldots, \Phi_K$, where $\Phi_1 = [0, 1/K), \Phi_2 = [1/K, 2/K), \ldots$, then evaluate,

$$d\Big(\mathbb{E}_\mathrm{p}[Y \mid \mathrm{p}_\theta(y = 1|X) \in \Phi_k], \\ \mathbb{E}_\mathrm{p}[\mathrm{p}_\theta(y = 1|X) \mid \mathrm{p}_\theta(y = 1|X) \in \Phi_k]\Big)$$

We can estimate $\mathbb{E}_\mathrm{p}[Y \mid \mathrm{p}_\theta(y = 1 \mid X) \in \Phi_k]$ as,

$$\frac{\mathrm{p}(Y = 1)\mathbb{E}_\mathrm{p}[\mathbb{I}(\mathrm{p}_\theta(y = 1 \mid X) \in \Phi_k) \mid Y = 1]}{\mathbb{E}_\mathrm{p}[\mathbb{I}(\mathrm{p}_\theta(y = 1 \mid X) \in \Phi_k)]},$$

where again we find only quantities that depend on $\mathrm{p}(y = 1)$, $\mathrm{p}(x)$ and $\mathrm{p}(x \mid y = 1)$. Details in Section A.2.

In summary, it is possible to evaluate models using heldout positive examples. So we can critique and improve our models even when $q = 1$.

3.4 Extensions

Richer measurements Many technologies record measurements that are richer than a binary signal, such as a vector of binding counts against different targets, or a vector of gene expression levels. In this case, y is no longer binary, and instead $y \in \mathbb{R}^k$.

LeaVS generalizes. Experimentally, assume we can sort cells based on an overall measurement of activity. We choose the fraction q of our measurement budget

to allocate to cells with $y \in \mathcal{O}$. For example, if y_j records binding against target j, we may sort out cells that show binding against at least one target: $\mathcal{O} = \{y : \sum_j y_j > 0\}$. After sorting, we measure a dataset of n sequence-activity pairs $(x_1, y_1), \ldots, (x_n, y_n)$, of which $q\,n$ have $y_i \in \mathcal{O}$ and the rest have $y_i \notin \mathcal{O}$. For the remaining $N - n$ cells, we can only record whether or not they are hits, giving a dataset $\{z_{n+1}, \ldots, z_N\}$ where $z_i = \mathbb{I}(y_i \in \mathcal{O})$. In this setting, the LeaVS objective becomes,

$$\sum_{i=1}^{n} \log \mathrm{p}_\theta(y_i \mid x_i) + \sum_{i=n+1}^{N} \log \int \mathrm{p}_\theta(z_i \mid x)\mathrm{p}(x)dx, \quad (5)$$

where the second term describes the marginal likelihood that $y_i \in \mathcal{O}$ or $y_i \notin \mathcal{O}$.

We can still learn the true $\mathrm{p}(y \mid x)$ under $q = 1$, if we assume $\mathrm{p}(y \mid x)$ is exponential family given x. For example, binding or expression counts are often modeled as Poisson$(h(x))$, where $h : \mathcal{X} \to \mathbb{R}_+^k$ is a neural network (Assumption 7). Then the model $\mathrm{p}_\theta(y \mid x)$ will converge to the true $\mathrm{p}(y \mid x)$ for any $q \in [0, 1]$ (Proposition 8), and information is maximized at $q = 1$ when active sequences are rare (Proposition 9). We can evaluate by checking the accuracy of $\mathrm{p}_\theta(y \in \mathcal{O} \mid x)$ and the fit of $\mathrm{p}_\theta(y \mid x, y \in \mathcal{O})$ (Section B.7.4).

Robust library models Accurate learning of $\mathrm{p}(y \mid x)$ with LeaVS depends on an accurate model of $\mathrm{p}(x)$. Models can be quite accurate for chemical DNA synthesis, but in cases where we do not have or trust a model of $\mathrm{p}(x)$, we can incorporate data about the prescreening library [Weinstein et al., 2026].

If we sequence the library before screening, we obtain samples $x'_1, \ldots, x'_M \sim \mathrm{p}(x)$. We can use their empirical distribution to approximate $\mathrm{p}(x)$: we plug the data directly into the Monte Carlo approximation in Equation (3). We can also combine the data with an approximate model, e.g. we can plug in a mix of samples from the data and samples from the model to Equation (3). This corresponds to posterior inference under a Dirichlet process prior, $\mathrm{p}(x) \sim \mathrm{DP}(\mathrm{p}_{model}(x), \alpha)$, which lets us leverage model predictions when data is sparse or incomplete (Section B.8).

4 THEORY

We have proposed methods to train and evaluate predictive models under different measurement allocations, $q \in [0, 1]$. In this section, we ask which q to choose in practice. We consider a class of distributions that is widely found in biology. For this class, $q = 1$ maximizes the information we obtain about $\mathrm{p}(y \mid x)$.

We work in the framework of Bayesian experimental design [Lindley, 1956, Chaloner and Verdinelli, 1995, Rainforth et al., 2024]. The goal is to choose a design that maximizes the information we will gain from collecting data. Here, the experimental design parameter is q. Based on q, we obtain a (random) dataset $\mathcal{D}_q$. After observing this data, we obtain a posterior over the unknown parameter θ. Its Shannon entropy is,

$$\mathcal{H}(\mathrm{p}(\theta \mid \mathcal{D}_q)) = -\int \mathrm{p}(\theta \mid \mathcal{D}_q) \log \mathrm{p}(\theta \mid \mathcal{D}_q). \quad (6)$$

Our goal is to set q such that the entropy will be minimized, indicating that we have learned as much as possible about $\mathrm{p}(y \mid x)$:

$$\operatorname*{argmin}_{q \in [0,1]} \mathcal{H}(\mathrm{p}(\theta \mid \mathcal{D}_q)). \quad (7)$$

We present our results for binary y, and extend the results to more general y in Section B.7.

4.1 Sparse Activity

In general, the optimal choice of q will depend on the data distribution $\mathrm{p}(x, y)$ and the model $\mathrm{p}_\theta(y \mid x)$. We will focus on a class of distributions that are common in biological applications.

Empirically, often only sequences in a small region of sequence space have any biological function or activity. That is, activity is *sparse*. To formalize this mathematically, note we can write any $\mathrm{p}_\theta(y \mid x)$ as,

$$\mathrm{p}_\theta(y = 1 \mid x) = \begin{cases} h_\theta(x) & x \in S_\theta \\ 0 & x \notin S_\theta \end{cases} \quad (8)$$

for $\theta \in \Theta \subseteq \mathbb{R}^d$. Here $S_\theta \subseteq \mathcal{X}$ defines the region of sequence space where the model predicts non-zero activity. We will assume that the true data distribution falls into the model class,

Assumption 1 (Well-specified model)**.** *There exists* $\theta_0 \in \Theta$ *such that* $\mathrm{p}_{\theta_0}(y \mid x) = \mathrm{p}(y \mid x)$.

With this class of distributions, $\eta \triangleq \mathrm{p}(x \in S_{\theta_0})$ gives the probability of a sequence falling in the active region. η is a measure of how rare active sequences are across the sequence landscape. We are interested in the regime where η is small, i.e., activity is sparse. This is especially common in the most challenging molecular discovery problems.

4.2 Asymptotic consistency

We are interested in the large-scale screening regime, where we are collecting big datasets that are limited in size by the cost of measuring x. To study this regime, we first take $N \to \infty$, so that there is an excess of

y-only data. Then, we examine the behavior of the posterior as $n \to \infty$, i.e. as we collect more x-y pairs.

When $N \to \infty$, the likelihood contribution $\mathcal{L}_y(\theta)$ dominates Equation (1). Asymptotically, $\mathcal{L}_y(\theta)$ will be maximized when $\int \mathrm{p}_\theta(y \mid x)\mathrm{p}(x)dx = \mathrm{p}(y)$, i.e. the model captures the true frequency of positive and negative examples. So to study the x measurement-limited regime, we make the simplifying assumption,

Assumption 2 (Known hit rate)**.** $\int \mathrm{p}_\theta(y = 1 \mid x)\mathrm{p}(x)dx = \mathrm{p}(y = 1)$ *for all* $\theta \in \Theta$.

We are now interested in understanding the behavior of the posterior as function of n, the number of x-y pairs we observe. After setting q, we observe,

$$X_{1,\ldots,nq} \overset{iid}{\sim} \mathrm{p}(x \mid y = 1) \tag{9}$$

$$X_{nq+1,\ldots,n} \overset{iid}{\sim} \mathrm{p}(x \mid y = 0), \tag{10}$$

and we are interested in the posterior $\mathrm{p}(\theta \mid \mathcal{D}_{n,q})$

$$\propto \mathrm{p}(\theta) \prod_{i=1}^{nq} \mathrm{p}_\theta(y_i = 1 \mid x_i) \prod_{i=nq+1}^{n} \mathrm{p}_\theta(y_i = 0 \mid x_i), \tag{11}$$

under the constraint of Assumption 2.

We first show that this posterior concentrates at the true θ_0 regardless of the choice of q. We assume the distribution $\mathrm{p}(x)$ has full support and the prior covers the true parameter, $\pi(\theta_0) > 0$, and we place standard regularity conditions on the likelihood's parameterization [Miller, 2021]. Details and proof in Section B.1.

Proposition 1 (Consistent for any q)**.** *Under Assumption 1, Assumption 2, Assumption 3 and Assumption 4, we have* $\mathrm{p}(\theta \mid \mathcal{D}_{n,q}) \to \delta_{\theta_0}$ *a.s. as* $n \to \infty$*, for* $q \in [0, 1]$.

The y-only data is essential to learning θ_0 under different allocations q. If we use a model without the constraint from Assumption 2, consistency fails. Details and proof in Section B.2.

Proposition 2 (Inconsistent without y-only data)**.** *Under Assumption 1, Assumption 5 and Assumption 6, we have* $\mathrm{p}(\theta \mid \mathcal{D}_{n,q}) \to \delta_{\tilde{\theta}}$ *a.s. as* $n \to \infty$*, where* $\mathrm{p}_{\tilde{\theta}}(y \mid x) \neq \mathrm{p}(y \mid x)$ *except at* $q = \mathrm{p}(y = 1)$.

4.3 Efficiency and optimal q

We now show that when activity is sparse, posterior information is asymptotically maximized by collecting only positive examples.

First, we look more closely at the behavior of the posterior as we collect more data. The Bernstein-von Mises theorem says the posterior will approach a Gaussian, under regularity conditions on the likelihood and prior. Let $\mathrm{p}_q(y)$ denote Bernoulli(q).

Theorem 1 (Bernstein-von Mises)**.** *[e.g. Miller, 2021, Thm. 3.2]. Let Assumption 1, Assumption 2, Assumption 3 and Assumption 4 hold. Let* $r_n(\tilde{\theta})$ *be the density of* $\sqrt{n}(\theta - \theta_n)$*, where* θ_n *is the maximum likelihood estimate. Then,* $\mathrm{p}(\tilde{\theta} \mid \mathcal{D}_{q,n})$ *converges a.s. in total variation to a normal distribution with mean 0 and inverse covariance,*

$$H_q = -\mathbb{E}_{\mathrm{p}(x|y)\mathrm{p}_q(y)}[\nabla^2_\theta \log \mathrm{p}_\theta(Y \mid X)|_{\theta=\theta_0}] \tag{12}$$

So, to maximize the amount of information we collect, we will minimize the entropy of the asymptotic posterior distribution (D-optimality [Chaloner and Verdinelli, 1995, Huan et al., 2024]),

$$\operatorname*{argmin}_{q \in [0,1]} \mathcal{H}(\mathcal{N}(0, H_q^{-1})). \tag{13}$$

We find that when the size η of the active region is sufficiently small, the asymptotic posterior entropy is minimized at $q = 1$. Let $S_0 \triangleq S_{\theta_0}$ and define information matrices over S_0,

$$\mathcal{I}_y \triangleq -\mathbb{E}_{\mathrm{p}(x|S_0,y)}[\nabla^2_\theta \log \mathrm{p}_\theta(Y \mid X)|_{\theta_0}] \tag{14}$$

for $y \in \{0, 1\}$. Note $\mathcal{I}_0$ and $\mathcal{I}_1$ can be positive definite because the true θ_0 is identified from $\mathrm{p}(x \mid y = 1)$ or, by symmetry, from $\mathrm{p}(x \mid y = 0)$.

Proposition 3. *Assume* $\mathcal{I}_0$ *and* $\mathcal{I}_1$ *are both positive definite. Then if,*

$$\det(\mathcal{I}_1 - \frac{\eta}{1-\eta}\mathcal{I}_0) > 0 \tag{15}$$

we have $\operatorname{argmax}_q \mathcal{H}(\mathcal{N}(0, H_q)) = 1$*. This holds for any* η *sufficiently small.*

Proof in Section B.3. The key idea is that, when η is small, most of the negative examples will come from outside the active region S_0 (Figure 1c). These provide very little information about $\mathrm{p}(y \mid x)$. Positive examples, meanwhile, provide large amounts of information, since they come from within the active region. By measuring only positive examples ($q = 1$), we concentrate our data collection efforts in the region of sequence space that provides the most information.

When should we set $q = 1$ in practice? *A priori* we do not know $\mathcal{I}_1$, $\mathcal{I}_0$ or η. For a back-of-the-envelope calculation, assume that within S_0, positive and negative examples are roughly equally common and equally informative, such that $\mathcal{I}_1 \approx \mathcal{I}_0$ and $\eta \approx 2\mathrm{p}(y = 1)$. Then, if $\mathrm{p}(y = 1) < 1/4$, Equation (15) will hold. So, back-of-the-envelope, we should collect only positive examples if we expect hit rates below about 25%. In practice, we can often estimate $\mathrm{p}(y = 1)$ before selecting q and measuring x, e.g. after sorting but before sequencing, so this decision rule is actionable.

How much information will we gain once we set $q = 1$? Back-of-the-envelope, we gain more than $-\frac{d}{2}\log 3\mathrm{p}(y=1)$ nats, where d is the dimension of the parameter space θ (Section B.4). This corresponds to the same information gain, asymptotically, as increasing the sample size from n to $n/3\mathrm{p}(y=1)$. Empirically, in challenging therapeutic discovery problems, $\mathrm{p}(y=1)$ can be one in a billion [Skora et al., 2015]. This suggests that in practice, we could increase our effective dataset size by a factor of hundreds of millions.

4.4 Robustness

Finally, we study the robustness of learning under $q = 1$. Learning with LeaVS requires an accurate model of $\mathrm{p}(x)$ to achieve identification and consistency. We examine robustness when the model of $\mathrm{p}(x)$ is wrong, and the true distribution is contaminated by the presence of outliers [Huber and Ronchetti, 2009].

Proposition 4 (Robust to ϵ contamination of $\mathrm{p}(x)$)**.** *Assume the data is generated from* $\mathrm{p}_\epsilon(x) = \epsilon\delta_{x'}(x) + (1-\epsilon)\mathrm{p}(x)$ *where* $x' \notin S_0$ *is outside the active region, while the model assumes* $\mathrm{p}(x)$ *is true (Assumption 2). Assume there exists a* θ_ϵ *such that*

$$\mathrm{p}_{\theta_\epsilon}(y \mid x) = (1-\epsilon)\mathrm{p}(y \mid x). \tag{16}$$

Then, under Assumption 3 and Assumption 4, $\mathrm{p}(\theta \mid \mathcal{D}_{n,q}) \to \delta_{\theta_\epsilon}$ *a.s. as* $n \to \infty$*, for* $q = 1$*.*

Proof in Section B.5. This result shows the learned model will be the same as the true $\mathrm{p}(y \mid x)$ up to rescaling. It will correctly rank sequences by expected activity, and provide the optimal ROC curve AUC.

Learning with LeaVS also requires an accurate estimate of $\mathrm{p}(y)$ from the y-only data. If this data is unreliable, we will again learn a model that is only wrong by a constant factor (Proposition 7). The model will still correctly rank sequences by expected activity.

5 EMPIRICAL RESULTS

We examine the empirical performance of LeaVS on both synthetic and experimental data. Our key finding is that the right combination of experimental design (measure only positive examples) and training algorithm (the LeaVS objective) yields better predictions.

Our theoretical results are based on an asymptotic analysis of Bayesian learning with a parametric model. In practice, non-Bayesian, deep learning models are widely used for sequence-to-activity maps. Here, we show empirically that our key theoretical findings carry over to this setting. We train transformer and CNN-based models, and approximate the LeaVS objective using stochastic minibatching (Section C.1).

5.1 Synthetic data

We first examined a synthetic data setting, where the true $\mathrm{p}(y \mid x)$ is known. We set $\mathrm{p}(x)$ to be a variational synthesis model of antibody CDRH3 loops [Weinstein et al., 2026]. Following previous studies of antibody binding, we set $\mathrm{p}(y \mid x)$ such that y depends on the presence of specific amino acid motif in x [Akbar et al., 2021, Pavlović et al., 2021]. The overall activity rate was small, at $\mathrm{p}(y=1) = 0.015$. Details in Section C.2.

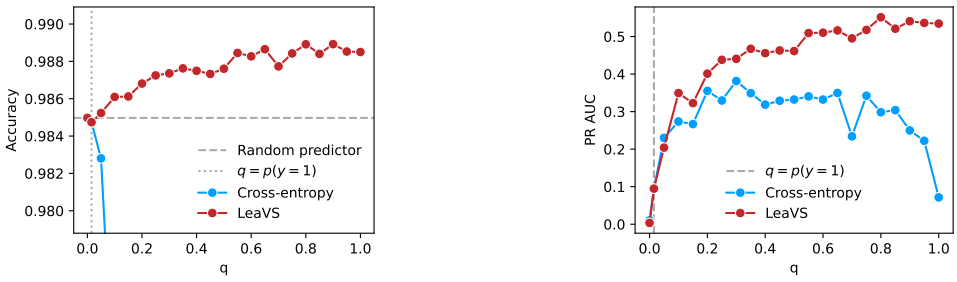
We generated datasets under different measurement allocations q, and trained transformer-based models using cross-entropy and the LeaVS objective. Examining model accuracy, we see very different performance as a function of q (Figure 2a). Standard cross-entropy peaks around $q = \mathrm{p}(y=1)$, i.e. when we take a representative sample from the distribution. But even at this peak, it does not surpass random predictions with frequency $\mathrm{p}(y=1)$. Training with LeaVS, we find similar performance at $q = \mathrm{p}(y=1)$. But now, as we turn up q, the model improves substantially, peaking near $q = 1$, even as the cross-entropy model's accuracy drops to near zero (Figure S1). In short, by modifying both the experimental design and the training procedure, we achieve substantial performance gains, while either modification on its own is insufficient.

As an additional performance metric, we also considered the area under the precision-recall curve. This metric is commonly used for imbalanced datasets, and is, unlike accuracy, insensitive to rescaling the prediction $\mathrm{p}_{\hat{\theta}}(y \mid x)$. We find qualitatively similar behavior. Cross-entropy peaks at low values of q, while LeaVS peaks near $q = 1$, achieving much better performance.

We checked the reliability of model evaluation in the absence of negative examples. We see a close match between estimated and true accuracy, sufficient to choose a high-quality model based on held-out data (Figure S2a). Calibration error estimates are similarly reliable (Figure S2b).

We repeated the simulations with count-valued rather than binary y, using a negative binomial distribution (Section 3.4). Again, performance is optimized by setting $q = 1$ and training with LeaVS (Figure S3).

We ran simulations with higher activity rates. At $\mathrm{p}(y=1) = 0.25$, LeaVS with $q = 1$ still outperforms cross-entropy (PR AUC 0.93 versus 0.70), while at $\mathrm{p}(y=1) = 0.5$, LeaVS underperforms (PR AUC 0.69 versus 0.76). This suggests the guidance to use LeaVS only when $\mathrm{p}(y=1) < 0.25$ (Section 4.3) can be conservative in practice.



(a) Accuracy.

(b) Area under the precision-recall curve.

Figure 2: **Performance on synthetic data.**

5.2 Experimental data from TCRs

We next sought to evaluate LeaVS on real experimental data, where $\mathrm{p}(x, y)$ is given by nature. We used a dataset screening human T cell receptors (TCRs), x, for binding against an influenza antigen, y [Genomics, 2019]. Here we have measurements of (x, y) for each cell, obtained via single-cell sequencing. Starting from this complete dataset, we generated smaller datasets with different values of q, by subsampling cells. To obtain samples from $\mathrm{p}(x)$ for training, we drew samples of x from heldout cells. Details in Section C.3. Note that because we must subsample a full dataset, n is relatively small, 170 cells, which increases variability.

The results on this experimental data, shown in Figure 3, show a similar pattern to the synthetic data. Combining high q with the LeaVS objective boosts performance over cross-entropy at $q = \mathrm{p}(y = 1)$. Alone, increasing q or using the LeaVS objective does not help; the experiment and inference must be modified together. Accuracy and calibration error estimates continue to track the true values, though they are sometimes conservative, under-estimating accuracy and over-estimating calibration (Figure S4). Overall, we find LeaVS produces better models of a real sequence-to-activity relationship.

5.3 Large-scale demonstration on antibodies

Finally, we deploy LeaVS at scale to learn therapeutically important sequence-activity relationships. We focus on TCR-mimicking antibodies (TCRm) and their binding against a challenging oncology target [Klebanoff et al., 2023]. Previously, we used variational synthesis to synthesize 10^{16} samples from a generative model of human antibody CDRH3s. The DNA was assembled into scFv CAR cell therapy constructs and delivered into 22.5 million human cells. The library was screened against a panel of fluorescent and DNA-barcoded peptide-HLA (pHLA) oncology targets. We obtained y values for the full population of cells based on sorting (FACS). Single-cell sequencing was used to recover synthesized antibody sequence, x, along with counts measuring binding strength, y (Section C.4 and Weinstein et al. [2026]). Given the low proportion of active sequences estimated from sorting (below 1%), we chose $q = 1$ and allocated our entire single-cell sequencing budget to active sequences.

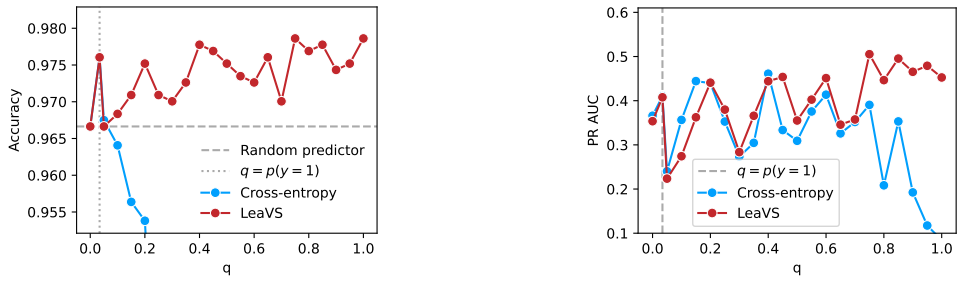
Using the LeaVS objective, we trained a model to predict binding to the cancer-testis antigen MAGE-A4. We trained on binding counts using the generalized LeaVS objective (Equation (5)) and a negative binomial activity distribution, binarizing y with a threshold for evaluation. We used the antibody variational synthesis model for $\mathrm{p}(x)$. We find that the LeaVS-trained predictor successfully generalizes to heldout data, surpassing a random predictor (Figure 4a). Indeed, we see precision values more than an order of magnitude larger than the baseline hit rate (dashed), implying that if we select sequences with high values of $\mathbb{E}[Y \mid x]$, we can enrich for MAGE-A4 hits by 10-100x. This suggests the model is capable of being used for iterative sequence design.

To confirm the model's predictive ability was not an artifact of misspecification of $\mathrm{p}(x)$, we used sequencing data from the pre-screen DNA library. The model's estimate of $\mathbb{E}[Y \mid x]$ still distinguishes this distribution from the distribution of hits (Figure 4b), with similar predictive performance (Figure S5). These results are robust to changes in model architecture, e.g. using a CNN in place of a transformer (Figure S6).

In short, we can use LeaVS to train therapeutically relevant predictive models, describing interactions between generative model-designed antibodies and challenging oncology targets.

6 DISCUSSION

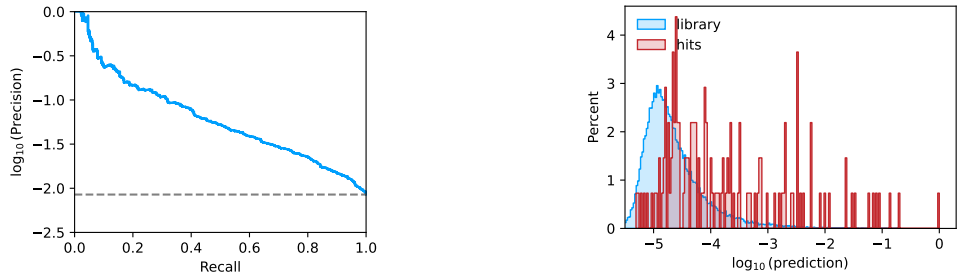
We have proposed LeaVS, an approach to scale up biological machine learning. LeaVS rests on the co-



(a) Accuracy.

(b) Area under the precision-recall curve.

Figure 3: **Performance on experimental TCR data.**



(a) Precision-recall curve. AUC: 0.12.

(b) Predictive scores on library sequences (blue) versus heldout positive examples (red).

Figure 4: **Predicting binding of TCRm scFv CAR therapeutic candidates to MAGE-A4, a challenging oncology target.**

design of experiments and training algorithms. It produces more informative datasets by focusing limited measurement budgets on positive examples. Then, it adjusts for missing data during training, using a generative model of the experiment. By shifting the distribution of training data, then correcting for this shift post-hoc, we accelerate learning.

Limitations and assumptions LeaVS assumes access to samples from $\mathrm{p}(x)$. These samples can come from a variational synthesis model of the library, but this model may not be perfect. Samples can also come from sequencing part of the library before screening, but this is costly, and provides only a limited number of samples.

The choice of the synthesis distribution $\mathrm{p}(x)$ to use in the experiment is critical in practice, since it defines the region of sequence space to explore. Our method does not provide guidance on how to set $\mathrm{p}(x)$, but rather takes it as given.

The optimal q depends on quantities that are unobserved. We have relied on back-of-the-envelope estimates to suggest that we set $q = 1$ when we observe $\mathrm{p}(y = 1) < 0.25$ (Section 4.3). But if, for example, the sequence-to-activity map is not sparse, this may not be the optimal choice.

Future directions The LeaVS objective includes a marginal likelihood, which we approximate with a nested Monte Carlo estimate (Equation (3)) [Rainforth et al., 2018]. This estimate might be improved using importance sampling or multilevel Monte Carlo methods [Goda et al., 2022].

Experiments can be performed sequentially to gather more information. In this setting, the library distribution $\mathrm{p}(x)$ can be updated given current knowledge of $\mathrm{p}(y \mid x)$. It could be steered toward successful designs, or toward regions of sequence space where uncertainty remains high [Yang et al., 2025, Smith et al., 2023]. This naturally leads to higher values of $\mathrm{p}(y = 1)$, raising the question of how to adaptively set q as information accumulates [Rainforth et al., 2024].

A fundamental lesson of modern machine learning is that scale is essential to unlocking new model capabilities. In biological machine learning, laboratory experiments are a core part of the AI stack. Our work emphasizes the importance of designing experiments and training in tandem, to scale up models and accelerate learning.

Acknowledgements

We thank Alan N. Amin for useful discussions.

References

Rahmad Akbar, Philippe A Robert, Milena Pavlović, Jeliazko R Jeliazkov, Igor Snapkov, Andrei Slabodkin, Cédric R Weber, Lonneke Scheffer, Enkelejda Miho, Ingrid Hobæk Haff, Dag Trygve Tryslew Haug, Fridtjof Lund-Johansen, Yana Safonova, Geir K Sandve, and Victor Greiff. A compact vocabulary of paratope-epitope interactions enables predictability of antibody-antigen binding. *Cell Rep.*, 34(11):108856, March 2021.

Maria-Florina Balcan and Ruth Urner. Active learning – modern learning theory. In *Encyclopedia of Algorithms*, pages 8–13. Springer New York, New York, NY, 2016.

Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Stat. Sci.*, 10(3): 273–304, 1995.

Charles Elkan and Keith Noto. Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, August 2008. ACM.

Adam Foster, Martin Jankowiak, Eli Bingham, Paul Horsfall, Yee Whye Teh, Tom Rainforth, and Noah Goodman. Variational bayesian optimal experimental design. In *Neural Information Processing Systems (NeurIPS)*, March 2019.

Adam Foster, Martin Jankowiak, Matthew O'Meara, Yee Whye Teh, and Tom Rainforth. A unified stochastic gradient approach to designing Bayesian-optimal experiments. In *International Conference on Artificial Intelligence and Statistics*, pages 2959–2969. PMLR, June 2020.

Jonathan Frazer, Pascal Notin, Mafalda Dias, Aidan Gomez, Joseph K Min, Kelly Brock, Yarin Gal, and Debora S Marks. Disease variant prediction with deep generative models of evolutionary data. *Nature*, 599(7883):91–95, 2021.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *International conference on machine learning (ICML)*, 2017.

10x Genomics. A new way of exploring immunity - linking highly multiplexed antigen recognition to immune repertoire and phenotype, 2019.

Takashi Goda, Tomohiko Hironaka, Wataru Kitade, and Adam Foster. Unbiased MLMC stochastic gradient-based optimization of bayesian experimental designs. *SIAM J. Sci. Stat. Comput.*, 44(1): A286–A311, 2022.

Xun Huan, Jayanth Jagalur, and Youssef Marzouk. Optimal experimental design: Formulations and computations. *Acta Numer.*, 33:715–840, July 2024.

Daolang Huang, Yujia Guo, Luigi Acerbi, and Samuel Kaski. Amortized bayesian experimental design for decision-making. In *Neural Information Processing Systems*, November 2024.

Peter J Huber and Elvezio M Ronchetti. *Robust Statistics*. 2009.

Niels Bruun Ipsen, Pierre-Alexandre Mattei, and Jes Frellsen. How to deal with missing data in supervised deep learning? In *International Conference on Learning Representations (ICLR)*, 2022.

Julie Josse, Jacob M Chen, Nicolas Prost, Gaël Varoquaux, and Erwan Scornet. On the consistency of supervised learning with missing values. *Stat. Pap.*, 65(9):5447–5479, December 2024.

Ryuichi Kiryo, Gang Niu, Marthinus C du Plessis, and Masashi Sugiyama. Positive-unlabeled learning with non-negative risk estimator. In *Neural Information Processing Systems (NeurIPS)*, March 2017.

Christopher A Klebanoff, Smita S Chandran, Brian M Baker, Sergio A Quezada, and Antoni Ribas. T cell receptor therapeutics: immunological targeting of the intracellular cancer proteome. *Nat. Rev. Drug Discov.*, pages 1–22, October 2023.

D V Lindley. On a measure of the information provided by an experiment. *Ann. Math. Stat.*, 27(4): 986–1005, December 1956.

Jeffrey W Miller. Asymptotic normality, concentration, and coverage of generalized posteriors. *J. Mach. Learn. Res.*, 22(168):1–53, 2021.

Rafael Oliveira, Dino Sejdinovic, David Howard, and Edwin V Bonilla. Bayesian adaptive calibration and optimal design. In *Neural Information Processing Systems (NeurIPS)*, November 2024.

Milena Pavlović, Lonneke Scheffer, Keshav Motwani, Chakravarthi Kanduri, Radmila Kompova, Nikolay Vazov, Knut Waagan, Fabian L M Bernal, Alexandre Almeida Costa, Brian Corrie, Rahmad Akbar, Ghadi S Al Hajj, Gabriel Balaban, Todd M Brusko, Maria Chernigovskaya, Scott Christley, Lindsay G Cowell, Robert Frank, Ivar Grytten, Sveinung Gundersen, Ingrid Hobæk Haff, Eivind Hovig, Ping-Han Hsieh, Günter Klambauer, Marieke L Kuijjer, Christin Lund-Andersen, Antonio Martini, Thomas Minotto, Johan Pensar, Knut Rand, Enrico Riccardi, Philippe A Robert, Artur Rocha, Andrei Slabodkin, Igor Snapkov, Ludvig M Sollid, Dmytro

Titov, Cédric R Weber, Michael Widrich, Gur Yaari, Victor Greiff, and Geir Kjetil Sandve. The immuneML ecosystem for machine learning analysis of adaptive immune receptor repertoires. *Nature Machine Intelligence*, 3(11):936–944, November 2021.

Marthinus Du Plessis, Gang Niu, and Masashi Sugiyama. Convex formulation for learning from positive and unlabeled data. In *International Conference on Machine Learning (ICML)*, 2015.

Tom Rainforth, Robert Cornish, Hongseok Yang, Andrew Warrington, and Frank Wood. On nesting monte carlo estimators. In *International Conference on Machine Learning (ICML)*, 2018.

Tom Rainforth, Adam Foster, Desi R Ivanova, and Freddie Bickford Smith. Modern bayesian experimental design. *Stat. Sci.*, 2024.

Andrew D Skora, Jacqueline Douglass, Michael S Hwang, Ada J Tam, Richard L Blosser, Sandra B Gabelli, Jianhong Cao, Luis A Diaz, Jr, Nickolas Papadopoulos, Kenneth W Kinzler, Bert Vogelstein, and Shibin Zhou. Generation of MANAbodies specific to HLA-restricted epitopes encoded by somatically mutated genes. *Proc. Natl. Acad. Sci. U. S. A.*, 112(32):9967–9972, August 2015.

Freddie Bickford Smith, Andreas Kirsch, Sebastian Farquhar, Yarin Gal, Adam Foster, and Tom Rainforth. Prediction-oriented bayesian active learning. In *International Conference on Artificial Intelligence and Statistics*, 2023.

Hyebin Song, Bennett J Bremer, Emily C Hinds, Garvesh Raskutti, and Philip A Romero. Inferring protein sequence-function relationships with large-scale positive-unlabeled learning. *Cell Syst.*, 12(1): 92–101.e8, January 2021.

Y W Teh. Dirichlet process. *Encyclopedia of machine learning*, 1063:280–287, 2010.

Nicole N Thadani, Sarah Gurev, Pascal Notin, Noor Youssef, Nathan J Rollins, Daniel Ritter, Chris Sander, Yarin Gal, and Debora S Marks. Learning from prepandemic data to forecast viral escape. *Nature*, 622(7984):818–825, October 2023.

Juozas Vaicenavicius, David Widmann, Carl Andersson, Fredrik Lindsten, Jacob Roll, and Thomas B Schön. Evaluating model calibration in classification. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, February 2019.

Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, Basile I M Wicky, Nikita Hanikel, Samuel J Pellock, Alexis Courbet, William Sheffler, Jue Wang, Preetham Venkatesh, Isaac Sappington, Susana Vázquez Torres, Anna Lauko, Valentin De Bortoli, Emile Mathieu, Sergey Ovchinnikov, Regina Barzilay, Tommi S Jaakkola, Frank DiMaio, Minkyung Baek, and David Baker. De novo design of protein structure and function with RFdiffusion. *Nature*, 620(7976):1089–1100, August 2023.

Eli N Weinstein, Alan N Amin, Will Grathwohl, Daniel Kassler, Jean Disset, and Debora S Marks. Optimal design of stochastic DNA synthesis protocols based on generative sequence models. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*. PMLR, 2022.

Eli N Weinstein, Andrei Slabodkin, Mattia G Gollub, Kerry Dobbs, Xiao-Bing Cui, Fang Zhang, Kristina Gurung, and Elizabeth B Wood. Lifting biomolecular data acquisition. *arXiv [q-bio.BM]*, December 2025.

Eli N Weinstein, Mattia G Gollub, Andrei Slabodkin, Kerry Dobbs, Xiao-Bing Cui, Cameron L Gardner, Ryan J Grant, Kristina Gurung, Amira Bailey, Alan N Amin, George M Church, and Elizabeth B Wood. Manufacturing-aware generative models enable petascale synthesis of designed DNA. *Nat. Biotechnol.*, pages 1–9, March 2026.

Yifan Wu. *Learning to Predict and Make Decisions under Distribution Shift*. PhD thesis, Carnegie Mellon University, 2021.

Jason Yang, Wenda Chu, Daniel Khalil, Raul Astudillo, Bruce James Wittmann, Frances H Arnold, and Yisong Yue. Steering generative models with experimental data for protein fitness optimization. In *Neural Information Processing Systems (NeurIPS)*, October 2025.

Michele Zemplenyi and Jeffrey W Miller. Bayesian optimal experimental design for inferring causal structure. *Bayesian Anal.*, 18(3):929–956, 2023.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [**Yes**/No/Not Applicable]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [**Yes**/No/Not Applicable]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [**Yes**/No/Not Applicable]

2. For any theoretical claim, check if you include:

(a) Statements of the full set of assumptions of all theoretical results. [**Yes**/No/Not Applicable]
(b) Complete proofs of all theoretical results. [**Yes**/No/Not Applicable]
(c) Clear explanations of any assumptions. [**Yes**/No/Not Applicable]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [**Yes**/No/Not Applicable]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [**Yes**/No/Not Applicable]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [**Yes**/No/Not Applicable]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [**Yes**/No/Not Applicable]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [**Yes**/No/Not Applicable]
   (b) The license information of the assets, if applicable. [**Yes**/No/Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [**Yes**/No/Not Applicable]
   (d) Information about consent from data providers/curators. [Yes/No/**Not Applicable**]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Yes/No/**Not Applicable**]
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Yes/No/**Not Applicable**]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Yes/No/**Not Applicable**]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Yes/No/**Not Applicable**]

Accelerated Learning on Large-Scale Screens: Supplementary Materials

Code demonstrating LeaVS is available at `https://github.com/jurabio/leavs-demo`.

A Evaluation Details

A.1 Accuracy Approximation

To approximate the accuracy $\mathrm{A_p}[t_\theta]$, let $(\tilde{x}_1, 1), ..., (\tilde{x}_{\tilde{n}}, 1), y_{\tilde{n}+1}, ..., y_{\tilde{N}}$ denote heldout data, and draw samples $x'_1, ..., x'_M \sim \mathrm{p}(x)$. We obtain,

$$\mathrm{A_p}[t_\theta] = \mathrm{p}(y=1)\mathbb{E}[t_\theta(X)|Y=1] + \mathbb{E}_\mathrm{p}[1 - t_\theta(X)] - \mathrm{p}(y=1)\mathbb{E}_p[1 - t_\theta(X)|Y=1] \tag{17}$$

$$\approx (\frac{1}{\tilde{N}}\sum_{i=1}^{\tilde{N}} \tilde{y}_i)\frac{1}{\tilde{n}}\sum_{i=1}^{\tilde{n}} t_\theta(\tilde{x}_i) + \frac{1}{M}\sum_{j=1}^{M}(1 - t_\theta(x'_i)) - (\frac{1}{\tilde{N}}\sum_{i=1}^{\tilde{N}} \tilde{y}_i)\frac{1}{\tilde{n}}\sum_{i=1}^{\tilde{n}}(1 - t_\theta(\tilde{x}_i)) \tag{18}$$

Precision and Recall We can apply the same strategy as used for accuracy to approximate the precision and recall, along with other standard measures of classifier performance. The precision admits the decomposition

$$\frac{\mathbb{E}_\mathrm{p}[Y t_\theta(X)]}{\mathbb{E}_\mathrm{p}[t_\theta(X)]} = \frac{\mathrm{p}(y=1)\mathbb{E}[t_\theta(X)|Y=1]}{\mathbb{E}_\mathrm{p}[t_\theta(X)]} \tag{19}$$

The numerator can be estimated as for accuracy; the denominator can be estimated using samples from $\mathrm{p}(x)$. The recall is the same, but with a different denominator,

$$\frac{\mathbb{E}_\mathrm{p}[Y t_\theta(X)]}{\mathrm{p}(y=1)}. \tag{20}$$

We can estimate the denominator using y-only data, as for accuracy.

A.2 Calibration Approximation

We form an estimate of the histogrammed ECE,

$$\mathrm{ECE_p}[\mathrm{p}_\theta] \approx \sum_{k=1}^{K} \mathrm{p}(\mathrm{p}_\theta(y=1|X) \in \Phi_k) d\left(\mathbb{E}_\mathrm{p}[Y \mid \mathrm{p}_\theta(y=1|X) \in \Phi_k], \mathbb{E}_\mathrm{p}[\mathrm{p}_\theta(y=1|X) \mid \mathrm{p}_\theta(y=1|X) \in \Phi_k]\right) \tag{21}$$

Let $(\tilde{x}_1, 1), ..., (\tilde{x}_{\tilde{n}}, 1), y_{\tilde{n}+1}, ..., y_{\tilde{N}}$ denote heldout data, and draw samples $x'_1, ..., x'_M \sim \mathrm{p}(x)$. We obtain,

$$\mathrm{p}(\mathrm{p}_\theta(y=1 \mid X) \in \Phi_k) \approx \frac{1}{M}\sum_{j=1}^{M} \mathbb{I}(\mathrm{p}_\theta(y=1 \mid x'_j) \in \Phi_k) \tag{22}$$

$$\mathbb{E}_\mathrm{p}[Y \mid \mathrm{p}_\theta(y=1 \mid X) \in \Phi_k] = \frac{\mathbb{E}_\mathrm{p}[Y \mathbb{I}(\mathrm{p}_\theta(y=1 \mid X) \in \Phi_k)]}{\mathbb{E}_\mathrm{p}[\mathbb{I}(\mathrm{p}_\theta(y=1 \mid X) \in \Phi_k)]} \tag{23}$$

$$\approx \frac{\left(\frac{1}{\tilde{N}}\sum_{i=1}^{\tilde{N}} \tilde{y}_i\right)\frac{1}{\tilde{n}}\sum_{i=1}^{\tilde{n}} \mathbb{I}(\mathrm{p}_\theta(y=1 \mid \tilde{x}_i) \in \Phi_k)}{\frac{1}{M}\sum_{j=1}^{M} \mathbb{I}(\mathrm{p}_\theta(y=1 \mid x'_j) \in \Phi_k)} \tag{24}$$

$$\mathbb{E}_\mathrm{p}[\mathrm{p}_\theta(y=1 \mid X) \mid \mathrm{p}_\theta(y=1 \mid X) \in \Phi_k] = \frac{\mathbb{E}_\mathrm{p}[\mathrm{p}_\theta(y=1 \mid X)\mathbb{I}(\mathrm{p}_\theta(y=1 \mid X) \in \Phi_k)]}{\mathbb{E}_\mathrm{p}[\mathbb{I}(\mathrm{p}_\theta(y=1 \mid X) \in \Phi_k)]} \tag{25}$$

$$\approx \frac{\frac{1}{M}\sum_{j=1}^{M} \mathrm{p}_\theta(y=1 \mid x'_j)\mathbb{I}(\mathrm{p}_\theta(y=1 \mid x'_i) \in \Phi_k)}{\frac{1}{M}\sum_{j=1}^{M} \mathbb{I}(\mathrm{p}_\theta(y=1 \mid x'_j) \in \Phi_k)} \tag{26}$$

B Proofs

B.1 Proof of Proposition 1 and Theorem 1

We assume the distribution $\mathrm{p}(x)$ has support over all possible values of x, i.e. we will only be able to learn $\mathrm{p}(y \mid x)$ for x that can actually occur.

Assumption 3 (Full support)**.** $\mathrm{p}(x) > 0$ *for all* $x \in \mathcal{X}$.

Our results build on the observation that we can nonparametrically identify $\mathrm{p}(y \mid x)$ from $\mathrm{p}(x)$ (the library distribution), $\mathrm{p}(x \mid y = 1)$ (the distribution of positive examples) and $\mathrm{p}(y = 1)$ (the activity/hit rate).

Lemma 1 (Nonparametric identification)**.** *Let Assumption 3 hold. Given* $\mathrm{p}(x)$*,* $\mathrm{p}(x \mid y = 1)$ *and* $\mathrm{p}(y = 1)$*, we can compute* $\mathrm{p}(y \mid x)$*.*

Proof.

$$\mathrm{p}(y = 1 \mid x) = \frac{\mathrm{p}(x \mid y = 1)\mathrm{p}(y = 1)}{\mathrm{p}(x)} \tag{27}$$

□

To prove Proposition 1 and Theorem 1, we make use of the average log likelihood of the model,

$$L_{q,n}(\theta) \triangleq \frac{1}{n}\sum_{i=1}^{nq} \log \mathrm{p}_\theta(y = 1 \mid x_i) + \frac{1}{n}\sum_{i=nq+1}^{n} \log \mathrm{p}_\theta(y = 0 \mid x_i). \tag{28}$$

By the strong law of large numbers, $L_{q,n}(\theta)$ converges pointwise a.s. to the expected log likelihood as $n \to \infty$,

$$\begin{aligned} L_q(\theta) &\triangleq q\mathbb{E}_{\mathrm{p}(x|y=1)}[\log \mathrm{p}_\theta(y = 1 \mid X)] + (1 - q)\mathbb{E}_{\mathrm{p}(x|y=0)}[\log \mathrm{p}_\theta(y = 0 \mid X)] && (29)\\ &= \mathbb{E}_{\mathrm{p}(x|y)\mathrm{p}_q(y)}[\log \mathrm{p}_\theta(Y \mid X)], && (30)\end{aligned}$$

where $\mathrm{p}_q(y)$ is defined as the Bernoulli distribution with parameter q.

Our key intermediate result is that the expected log likelihood of the model under the data distribution is maximized at θ_0, for any q.

Proposition 5 (θ_0 is optimal)**.** *Let Assumption 1, Assumption 2 and Assumption 3 hold. Then for all* $q \in [0, 1]$*,* $L_q(\theta)$ *is maximized at* θ_0*.*

Proof. Let $\mathrm{p}_\theta(x \mid y) \propto \mathrm{p}_\theta(y \mid x)\mathrm{p}(x)$ denote the conditional distribution of y under the model and the true x distribution. We have

$$\begin{aligned} \operatorname*{argmax}_{\theta\in\Theta} L_q(\theta) &= \operatorname*{argmax}_{\theta\in\Theta} \mathbb{E}_{\mathrm{p}(x|y)\mathrm{p}_q(y)}[\log \mathrm{p}_\theta(Y \mid X) + \log \mathrm{p}(X)] && (31)\\ &= \operatorname*{argmax}_{\theta\in\Theta} \mathbb{E}_{\mathrm{p}(x|y)\mathrm{p}_q(y)}[\log \mathrm{p}_\theta(X \mid Y) + \log \mathrm{p}(Y)] && (32)\\ &= \operatorname*{argmax}_{\theta\in\Theta} \mathbb{E}_{\mathrm{p}(x|y)\mathrm{p}_q(y)}[\log \mathrm{p}_\theta(X \mid Y)] && (33)\\ &= \operatorname*{argmin}_{\theta\in\Theta} q\mathrm{KL}(\mathrm{p}(x \mid y = 1)\|\mathrm{p}_\theta(x \mid y = 1)) + (1 - q)\mathrm{KL}(\mathrm{p}(x \mid y = 0)\|\mathrm{p}_\theta(x \mid y = 0)) && (34)\end{aligned}$$

where in the first line we use the fact that $\mathrm{p}(x)$ does not depend on θ, for the second line we use the chain rule for conditional entropy, for the third line we use Assumption 2.

When $q > 0$, the minimum of the first term in Equation (34) is reached when $\mathrm{p}_\theta(x \mid y = 1) = \mathrm{p}(x \mid y = 1)$. This occurs if and only if $\mathrm{p}_\theta(y \mid x) = \mathrm{p}(y \mid x)$, by Equation (27), since $\mathrm{p}(x) > 0$ for all x and $\mathrm{p}_\theta(y) = \mathrm{p}(y)$ (Assumption 2). So, since the model is well specified (Assumption 1) θ_0 must be a minimizer of the first term in Equation (34). The same argument holds for the second term of Equation (34), so θ_0 must minimize Equation (34) overall. When $q = 0$, the same argument holds with $y = 0$ switched for $y = 1$. □

For this result to translate into posterior consistency and asymptotic normality, we must place regularity conditions on the model's parameterization. We use the conditions of Theorem 3.2 in Miller [2021], which provides a version of the Bernstein-von Mises theorem (BvM) which handles settings where the model's distribution does not match the data distribution, as is the case here.

Assumption 4 (Bernstein-von Mises assumptions)**.** *Let $\Theta \subseteq \mathbb{R}^d$. Assume the prior $\mathrm{p}(\theta)$ is continuous at θ_0 and $\pi(\theta_0) > 0$. Let $E \subseteq \Theta$ be a convex bounded open set that contains θ_0. Assume $L_{n,q}(\theta)$ has continuous third derivatives on E. Assume $-\nabla^2_\theta L_q(\theta_0)$ is positive definite. Assume the set $\{\big(\sum_{i,j,k=1}^d \left(\frac{\partial L_{n,q}}{\partial\theta_i\partial\theta_j\partial\theta_k}(\theta)\right)^2\big)^{1/2} : \theta \in E, n \in \mathbb{N}\}$ is bounded. Assume that either (a) $L_q(\theta) < L_q(\theta_0)$ for all $\theta \in K\backslash\{\theta_0\}$ and $\limsup_n \sup_{\theta\in\Theta\backslash K} L_{n,q}(\theta) < L_q(\theta_0)$ for a compact $K \subseteq E$ with θ_0 in the interior of K or (b) $-L_{n,q}(\theta)$ is convex for each n and $\nabla L(\theta_0) = 0$.*

Crucially, the reason Assumption 4 can be satisfied here is because of Proposition 5, that is, θ_0 is a maximizer of the expected log likelihood, and hence we have $\nabla L(\theta_0) = 0$ and $-\nabla^2_\theta L_q(\theta_0)$ positive definite. With Assumption 4 in place, Proposition 1 and Theorem 1 follow from Theorem 3.2 in Miller [2021].

B.2 Proof of Proposition 2

If we do not have the constraint of Assumption 2, the learned model will reflect the shifted distribution $\mathrm{p}(x \mid y)\mathrm{p}_q(y)$ rather than the target distribution $\mathrm{p}(x, y)$. We assume first that the model includes this shifted distribution. Let $\mathrm{p}_q(x) = \sum_{y=0}^1 \mathrm{p}(x \mid y)\mathrm{p}_q(y)$ and $\mathrm{p}_q(y \mid x) = \mathrm{p}(x \mid y)\mathrm{p}_q(y)/\mathrm{p}_q(x)$ denote the marginal and conditional under the shifted distribution.

Assumption 5 (Specified under shifts)**.** *Assume there exists a $\tilde\theta \in \Theta$ such that $\mathrm{p}_{\tilde\theta}(y \mid x) = \mathrm{p}_q(y \mid x)$ for all x, y.*

For this assumption to hold at $q \neq \mathrm{p}(y = 1)$, Assumption 2 must be violated, since $\int \mathrm{p}_q(y = 1 \mid x)\mathrm{p}(x)dx = \mathrm{p}_q(y = 1) = q$. This means, moreover, that $\mathrm{p}_{\tilde\theta}(y \mid x) \neq \mathrm{p}(y \mid x)$. On the other hand, when $q = \mathrm{p}(y = 1)$, Assumption 5 coincides with Assumption 1, and we can set $\tilde\theta = \theta_0$.

Now, the expected likelihood is maximized at $\mathrm{p}_q(y \mid x)$.

Proposition 6 ($\tilde\theta$ is optimal)**.** *Let Assumption 5 and Assumption 3 hold. Then $L_q(\theta)$ is maximized at $\tilde\theta$.*

Proof. We have,

$$\operatorname*{argmax}_{\theta\in\Theta} L_q(\theta) = \operatorname*{argmax}_{\theta\in\Theta} \mathbb{E}_{\mathrm{p}(x|y)\mathrm{p}_q(y)}[\log \mathrm{p}_\theta(Y \mid X)] \tag{35}$$

$$= \operatorname*{argmin}_{\theta\in\Theta} \mathbb{E}_{\mathrm{p}_q(x)}[\mathrm{KL}(\mathrm{p}_q(Y \mid X)\|\mathrm{p}_\theta(Y \mid X))] \tag{36}$$

The KL divergence is minimized whenever $\mathrm{p}_q(Y \mid X) = \mathrm{p}_\theta(Y \mid X)$. □

Now, assume the model is sufficiently regular around $\tilde\theta$.

Assumption 6 (BvM assumptions)**.** *Assumption 4 holds with θ_0 replaced by $\tilde\theta$.*

As before, this assumption can only be satisfied because of Proposition 6. Then, Proposition 2 follows from Theorem 3.2 in Miller [2021].

B.3 Proof of Proposition 3

Proof. Using the formula for the entropy of a Gaussian, we can see

$$\operatorname*{argmin}_{q\in[0,1]} \mathcal{H}(\mathcal{N}(0, H_q^{-1})) = \operatorname*{argmax}_{q\in[0,1]} \det H_q \tag{37}$$

We now have

$$H_q = -q\mathbb{E}_{\mathrm{p}(x|y=1)}[\nabla^2_\theta \log \mathrm{p}_\theta(y = 1 \mid X)|_{\theta_0}] - (1 - q)\mathbb{E}_{\mathrm{p}(x|y=0)}[\nabla^2_\theta \log \mathrm{p}_\theta(y = 0 \mid X)|_{\theta_0}] \tag{38}$$

$$= q\mathcal{I}_1 + (1 - q)\mathrm{p}(S_0 \mid y = 0)\mathcal{I}_0 \tag{39}$$

where in the second line we have used that $\mathrm{p}(x \mid y=1) = \mathrm{p}(x \mid S_0, y=1)$ by Equation (8). This implies $H_1 = \mathcal{I}_1$ so we have,

$$H_q = H_1 + (1-q)(\mathrm{p}(S_0 \mid y=0)\mathcal{I}_0 - \mathcal{I}_1). \tag{40}$$

Recall some properties of the determinant: $\det(A+B) \geq \det A + \det B$, and $\det aB = a^d \det B$ for scalar a. We obtain,

$$\det H_1 \geq \det H_q + (1-q)^d \det(\mathcal{I}_1 - \mathrm{p}(S_0 \mid y=0)\mathcal{I}_0) \tag{41}$$

$$\geq \det H_q + (1-q)^d \det(\mathcal{I}_1 - \frac{\eta}{1-\eta}\mathcal{I}_0) + (1-q)^d \left(\frac{\eta}{1-\eta} - \mathrm{p}(S_0 \mid y=0)\right)^d \det \mathcal{I}_0 \tag{42}$$

Now we have,

$$\mathrm{p}(S_0 \mid y=0) = \frac{\mathrm{p}(S_0, y=0)}{\mathrm{p}(y=0)} = \frac{\eta \mathrm{p}(y=0 \mid S_0)}{1-\eta+\eta\mathrm{p}(y=0 \mid S_0)} \leq \frac{\eta}{1-\eta}. \tag{43}$$

where we have used Equation (8) for the second equality, and $0 \leq \mathrm{p}(y=0 \mid S_0) \leq 1$ for the inequality. Since $\mathcal{I}_0$ is positive definite by assumption, we have

$$\det H_1 \geq \det H_q + (1-q)^d \det(\mathcal{I}_1 - \frac{\eta}{1-\eta}\mathcal{I}_0) \tag{44}$$

Applying the assumption on the second term, we find that for any $0 \leq q < 1$,

$$\det H_1 > \det H_q. \tag{45}$$

Note $\mathcal{I}_0$ and $\mathcal{I}_1$ do not depend on η, as they condition on S_0. So, since $\mathcal{I}_1$ and $\mathcal{I}_0$ are positive definite, there always exists η sufficiently small such that $\det(\mathcal{I}_1 - \frac{\eta}{1-\eta}\mathcal{I}_0) > 0$. □

An interesting feature of this result is the optimal choice of q exhibits a threshold effect as a function of η. Naively, one may hypothesize that the optimal q converges to 1 as η converges to 0. But the result here is stronger: for sufficiently small η, the optimal choice of q becomes *exactly* 1. This means that if we expect the optimal region to be small, we should collect *only* positive examples.

B.4 Back-of-the-envelope information gain

We can compute the information gain as the difference in Shannon entropy of the asymptotic posterior distribution, when setting $q=1$ versus $q = \mathrm{p}(y=1)$ (or, equivalently, drawing samples i.i.d. from $\mathrm{p}(x,y)$):

$$\mathcal{H}(\mathcal{N}(0, n^{-1}H_q^{-1})) - \mathcal{H}(\mathcal{N}(0, n^{-1}H_1^{-1})) \tag{46}$$

Plugging in Equation (39), and using $\mathcal{I} \triangleq \mathcal{I}_0 = \mathcal{I}_1$ under our back-of-the-envelope assumptions, we have

$$= \frac{1}{2}\log\det\mathcal{I} - \frac{1}{2}\log\det[q\mathcal{I} + (1-q)\mathrm{p}(S_0 \mid y=0)\mathcal{I}] \tag{47}$$

$$= -\frac{d}{2}\log(q + (1-q)\mathrm{p}(S \mid y=0)) \tag{48}$$

Applying Equation (43), the information gain is at least

$$\geq -\frac{d}{2}\log(q + (1-q)\frac{\eta}{1-\eta}) \tag{49}$$

Plugging in the back-of-the-envelope assumption $q = \mathrm{p}(y=1) = \eta/2$, we have

$$= -\frac{d}{2}\log q - \frac{d}{2}\log(1 + 2\frac{1-q}{1-2q}) \tag{50}$$

$$\geq -\frac{d}{2}\log 3q. \tag{51}$$

We can compare this information gain from increasing the sample size from n to cn,

$$\mathcal{H}(\mathcal{N}(0, n^{-1}H_q^{-1})) - \mathcal{H}(\mathcal{N}(0, (cn)^{-1}H_q^{-1})) = \frac{1}{2}\log[(cn)^d] - \frac{1}{2}\log[n^d] \tag{52}$$

$$= \frac{d}{2}\log c \tag{53}$$

So, back-of-the envelope, collecting positive examples provides at least as much information as increasing the sample size from n to $\frac{n}{3\mathrm{p}(y=1)}$.

B.5 Proof of Proposition 4

Proof. To prove Proposition 4, we show that θ_ϵ minimizes the expected log likelihood of the model. Let $\mathrm{p}_\epsilon(x, y) = \mathrm{p}_\epsilon(x)\mathrm{p}(y \mid x)$ denote the joint distribution of the data, and let $\mathrm{p}_\theta(x, y) = \mathrm{p}(x)\mathrm{p}(y \mid x)$ denote the joint distribution of the model. Under Assumption 2, the true hit rate is known, so $\mathrm{p}_\epsilon(y = 1) = \int \mathrm{p}_\theta(y = 1 \mid x)\mathrm{p}(x)dx = \int \mathrm{p}(y = 1 \mid x)\mathrm{p}_\epsilon(x)dx$. Following Section B.1 the expected log likelihood is,

$$\operatorname*{argmax}_{\theta\in\Theta} L_q(\theta) = \operatorname*{argmax}_{\theta\in\Theta} \mathbb{E}_{\mathrm{p}_\epsilon(x|y)\mathrm{p}_q(y)}[\log \mathrm{p}_\theta(y \mid x)] \tag{54}$$

$$= \operatorname*{argmax}_{\theta\in\Theta} \mathbb{E}_{\mathrm{p}_\epsilon(x|y)\mathrm{p}_q(y)}[\log \mathrm{p}_\theta(y \mid x) + \log \mathrm{p}(x)] \tag{55}$$

$$= \operatorname*{argmax}_{\theta\in\Theta} \mathbb{E}_{\mathrm{p}_\epsilon(x|y)\mathrm{p}_q(y)}[\log \mathrm{p}_\theta(x \mid y) + \log \mathrm{p}_\epsilon(y)] \tag{56}$$

$$= \operatorname*{argmax}_{\theta\in\Theta} \mathbb{E}_{\mathrm{p}_\epsilon(x|y)\mathrm{p}_q(y)}[\log \mathrm{p}_\theta(x \mid y)] \tag{57}$$

$$= \operatorname*{argmin}_{\theta\in\Theta} q\mathrm{KL}(\mathrm{p}_\epsilon(x \mid y = 1)\|\mathrm{p}_\theta(x \mid y = 1)) + (1 - q)\mathrm{KL}(\mathrm{p}_\epsilon(x \mid y = 0)\|\mathrm{p}_\theta(x \mid y = 0)) \tag{58}$$

When $q = 1$, the minimum of zero is reached when $\mathrm{p}_\theta(x \mid y = 1) = \mathrm{p}_\epsilon(x \mid y = 1)$, implying

$$\frac{\mathrm{p}_\theta(y = 1 \mid x)\mathrm{p}(x)}{\mathrm{p}_\epsilon(y = 1)} = \frac{\mathrm{p}(y = 1 \mid x)\mathrm{p}_\epsilon(x)}{\mathrm{p}_\epsilon(y = 1)} \tag{59}$$

$$\mathrm{p}_\theta(y = 1 \mid x) = \frac{\mathrm{p}_\epsilon(x)}{\mathrm{p}(x)}\mathrm{p}(y = 1 \mid x). \tag{60}$$

Plugging in Equation (8), and using the fact that $x' \notin S_0$,

$$\mathrm{p}_\theta(y = 1 \mid x) = \begin{cases} (1 - \epsilon)h_{\theta_0}(x) & x \in S_0 \\ 0 & x \notin S_0 \end{cases} = (1 - \epsilon)\mathrm{p}(y \mid x) \tag{61}$$

We can confirm that this satisfies $\int \mathrm{p}_\theta(y = 1 \mid x)\mathrm{p}(x)dx = \mathrm{p}_\epsilon(y = 1) = \int \mathrm{p}(y = 1 \mid x)\mathrm{p}_\epsilon(x)dx = (1 - \epsilon)\int \mathrm{p}(y = 1 \mid x)\mathrm{p}(x)dx$. So θ_ϵ minimizes the expected log likelihood. The conclusion follows from Theorem 3.2 in Miller [2021], as in Section B.1 but with θ_ϵ replacing θ_0. □

Note the result holds even when $\epsilon < 0$, corresponding to the situation where a sequence drops out of the library, e.g. during a cloning or transformation step. However, the result does not necessarily hold for LeaVS when $q < 1$, since in the second term of Equation (58),

$$\mathrm{p}_\epsilon(x \mid y = 0) \propto \mathrm{p}(y = 0 \mid x)\mathrm{p}_\epsilon(x) = \begin{cases} (1 - \epsilon)\mathrm{p}(x)(1 - h_{\theta_0}(x)) & x \in S_0 \\ (1 - \epsilon)\mathrm{p}(x) + \epsilon\delta_{x'}(x) & x \notin S_0, \end{cases} \tag{62}$$

which, unlike $\mathrm{p}_\epsilon(x \mid y = 1)$, is not independent of the location of the distortion, x'. This re-emphasizes the value of setting $q = 1$.

Another source of misspecification of $\mathrm{p}(x)$ is when multiple sequences are delivered to the same cell, due to high multiplicity of infection. We characterize this situation in Weinstein et al. [2025].

B.6 Robustness to $\mathrm{p}(y)$

We examine robustness if the y-only data is unreliable. As with distortions of $\mathrm{p}(x)$, LeaVS learns the true model up to a factor that is constant in x.

Proposition 7 (Robustness to ϵ contamination of $\mathrm{p}(y)$)**.** *Assume the true marginal of the data is* $\mathrm{p}_\epsilon(y=1) = \mathrm{p}(y=1) + \epsilon$*, but the model assumes the marginal* $\mathrm{p}(y=1)$ *in Assumption 2. Assume there exists a* θ_ϵ *such that*

$$\mathrm{p}_{\theta_\epsilon}(y \mid x) = \frac{1}{1 + \frac{\epsilon}{\mathrm{p}(y=1)}} \mathrm{p}(y \mid x). \tag{63}$$

Then, under Assumption 3 and Assumption 4, $\mathrm{p}(\theta \mid \mathcal{D}_{n,q}) \to \delta_{\theta_\epsilon}$ *a.s. as* $n \to \infty$*, for* $q = 1$*.*

Proof. We show that θ_ϵ minimizes the expected log likelihood of the model. The model assumes $\int \mathrm{p}_\theta(y=1 \mid x)\mathrm{p}(x)dx = \mathrm{p}(y=1)$ whereas the actual data satisfies $\int \mathrm{p}(y=1 \mid x)\mathrm{p}(x)dx = \mathrm{p}_\epsilon(y=1)$. Following Section B.5, we have for $q = 1$,

$$\operatorname*{argmax}_{\theta \in \Theta} L_q(\theta) = \operatorname*{argmin}_{\theta \in \Theta} \mathrm{KL}(\mathrm{p}_\epsilon(x \mid y=1) \| \mathrm{p}_\theta(x \mid y=1)). \tag{64}$$

This gives a solution at,

$$\frac{\mathrm{p}_\theta(y=1 \mid x)\mathrm{p}(x)}{\mathrm{p}(y=1)} = \frac{\mathrm{p}(y=1 \mid x)\mathrm{p}(x)}{\mathrm{p}_\epsilon(y=1)} \tag{65}$$

$$\mathrm{p}_\theta(y=1 \mid x) = \frac{\mathrm{p}(y=1)}{\mathrm{p}_\epsilon(y=1)} \mathrm{p}(y=1 \mid x). \tag{66}$$

We confirm this has the specified marginal: $\int \mathrm{p}_\theta(y=1 \mid x)\mathrm{p}(x)dx = \int \frac{\mathrm{p}(y=1)}{\mathrm{p}_\epsilon(y=1)} \mathrm{p}(y=1 \mid x)\mathrm{p}(x)dx = \mathrm{p}(y=1)$. The result follows as in Section B.5. □

B.7 Extension to richer measurements

We generalize to a setting where y is not binary, but instead is in a more general space $\mathcal{Y} \subseteq \mathbb{R}^k$, following the setup in Section 3.4. We generalize the sparse activity model (Equation (8)) to an exponential family form.

Assumption 7 (Exponential family sparse activity)**.** *The conditional distribution of* y *takes the form,*

$$\mathrm{p}_\theta(y \mid x) = \begin{cases} \nu(y) \exp\left[h_\theta(x) \cdot T(y) - A(h_\theta(x))\right] & x \in S_\theta \\ \delta_{\vec{0}}(y) & x \notin S_\theta, \end{cases} \tag{67}$$

for $S_\theta \subseteq \mathcal{X}$ *and* $\vec{0}$ *the zero vector in* $\mathbb{R}^k$*. Here the exponential family has natural parameter* $h_\theta(x)$*, sufficient statistic* $T(y)$*, and log partition function* $A(h_\theta(x))$*. We assume* $\vec{0} \notin \mathcal{O}$*.*

Since we can take $S_\theta = \mathcal{X}$, this function class covers any distribution that is conditionally an exponential family distribution. The inactive level $\vec{0}$ is some arbitrary point outside the hit region $\mathcal{O}$. We assume the model is well-specified (Assumption 1), so there is a true $S_0 \triangleq S_{\theta_0}$ and $h_0(x) \triangleq h_{\theta_0}(x)$.

B.7.1 Identification

To establish identification, we require that there is sufficient information about the distribution $\mathrm{p}(y \mid x)$ within the observed region $\mathcal{O} \subseteq \mathcal{Y}$.

Assumption 8 (Rich hit region)**.** *Assume there exist* $\tilde{y}_1, \ldots, \tilde{y}_C \in \mathcal{O}$ *such that (a)* $\mathrm{p}(\tilde{y}_c \mid x) > 0$ *for all* $x \in S_0$ *and* $c \in \{1, \ldots, C\}$*, and (b) the square matrix* $\tilde{T}$ *with rows* $(T(\tilde{y}_c), 1)$ *is invertible.*

For example, consider the case where $y \in \mathbb{Z}_+^2$, the distribution $\mathrm{p}(y \mid x)$ is independently Poisson in both dimensions and the hit region is $\mathcal{O} = \{y : y_1 + y_2 > 0\}$. For the two-dimensional Poisson, the sufficient statistic is $T(y) = (y_1, y_2)$, so we can take $\tilde{y}_1 = (1,0)$, $\tilde{y}_2 = (0,1)$, and $\tilde{y}_3 = (1,1)$ to obtain an invertible $\tilde{T}$. More broadly, Assumption 8 admits the common biological setting where $\mathcal{O} = \{y : \sum_j y_j > \tau\}$ and the outcome distribution is Poisson, Gaussian or negative Binomial.

We can now establish identification.

Lemma 2 (Nonparametric identification)**.** *Assume* $\mathrm{p}(y \mid x)$ *satisfies Assumption 7 and Assumption 8, and let Assumption 3 hold. Given* $\mathrm{p}(x), \mathrm{p}(x, y \mid y \in \mathcal{O})$ *and* $\mathrm{p}(y \in \mathcal{O})$*, we can compute* $\mathrm{p}(y \mid x)$*.*

Proof. We first identify S_0. From Assumption 3 we can compute for all $x \in \mathcal{X}$,

$$\mathrm{p}(y \in \mathcal{O} \mid x) = \frac{\mathrm{p}(x \mid y \in \mathcal{O})\mathrm{p}(y \in \mathcal{O})}{\mathrm{p}(x)}. \tag{68}$$

From Assumption 8a, $\mathrm{p}(y \in \mathcal{O} \mid x) > 0$ for $x \in S_0$, and from Assumption 7, $\mathrm{p}(y \in \mathcal{O} \mid x) = 0$ for $x \notin S_0$. So S_0 is identified, as is $\mathrm{p}(y \mid x)$ for $x \notin S_0$.

It remains to identify $\mathrm{p}(y \mid x)$ for $x \in S_0$. We can compute for $x \in S_0$ and $y \in \mathcal{O}$,

$$\mathrm{p}(y \mid x) = \frac{\mathrm{p}(x, y \mid y \in \mathcal{O})\mathrm{p}(y \in \mathcal{O})}{\mathrm{p}(x)}. \tag{69}$$

From this, we can identify the distribution $\mathrm{p}(y \mid x)$ over all y using Assumption 8, as follows. Let $\tilde{L}$ denote the length C vector with entries $\log[\mathrm{p}(\tilde{y}_c \mid x)/\nu(\tilde{y}_c)]$, and note these values are finite by Assumption 8a. Rewrite the exponential family description, $\mathrm{p}(\tilde{y} \mid x) = \nu(\tilde{y}) \exp[h_0(x) \cdot T(\tilde{y}) - A(h_0(x))]$, to obtain the linear equation,

$$\tilde{L} = \tilde{T} \cdot \begin{pmatrix} h_0(x) \\ -A(h_0(x))) \end{pmatrix} \tag{70}$$

From Assumption 8b, we can solve to find $h_0(x) = (\tilde{T}^{-1}\tilde{L})_{1:C-1}$. Since the parameter of the exponential family, $h_0(x)$, is identified, the distribution $\mathrm{p}(y \mid x)$ is identified. □

B.7.2 Asymptotic consistency

We next establish asymptotic consistency. As before, we assume based on the the large N limit of the LeaVS objective that the hit rate is known.

Assumption 9 (Known hit rate)**.**

$$\int \mathrm{p}_\theta(y \in \mathcal{O} \mid x)\mathrm{p}(x) = \mathrm{p}(y \in \mathcal{O}) \tag{71}$$

After setting q, we observe,

$$(X_1, Y_1), \ldots, (X_{nq}, Y_{nq}) \overset{iid}{\sim} \mathrm{p}(x, y \mid y \in \mathcal{O}) \tag{72}$$

$$(X_{nq+1}, Y_{nq+1}), \ldots, (X_n, Y_n) \overset{iid}{\sim} \mathrm{p}(x, y \mid y \notin \mathcal{O}), \tag{73}$$

and we are interested in the posterior,

$$\mathrm{p}(\theta \mid \mathcal{D}_{n,q}) \propto \mathrm{p}(\theta) \prod_{i=1}^{n} \mathrm{p}_\theta(y_i \mid x_i), \tag{74}$$

under the constraint of Assumption 9. We show the posterior is asymptotically consistent under different q. Note the conditions for consistency under $q = 1$ are weaker than those for more general q.

Proposition 8 (Consistent under any q)**.** *Assume the model (Assumption 7) is well-specified (Assumption 1), the hit rate is known (Assumption 9),* $\mathrm{p}(x)$ *has full support (Assumption 3) and the prior and likelihood are well-behaved (Assumption 4). If the hit region is rich (Assumption 8), we have* $\mathrm{p}(\theta \mid \mathcal{D}_{n,q}) \to \delta_{\theta_0}$ *a.s. as* $n \to \infty$*, for* $q = 1$*. If, in addition, the complement of the hit region is rich, such that Assumption 8 holds with* $\mathcal{O}^c \triangleq \mathcal{Y} \setminus \mathcal{O}$ *in place of* $\mathcal{O}$*, then* $\mathrm{p}(\theta \mid \mathcal{D}_{n,q}) \to \delta_{\theta_0}$ *a.s. as* $n \to \infty$*, for any* $q \in [0, 1]$*.*

Proof. The average log likelihood of the data is

$$L_{q,n}(\theta) \triangleq \frac{1}{n} \sum_{i=1}^{n} \log \mathrm{p}_\theta(y_i \mid x_i). \tag{75}$$

By the strong law of large numbers, this function converges pointwise a.s. to the expected log likelihood

$$L_q(\theta) \triangleq q\mathbb{E}_{\mathrm{p}(x,y|y\in\mathcal{O})}[\log \mathrm{p}_\theta(Y \mid X)] + (1-q)\mathbb{E}_{\mathrm{p}(x,y|y\notin\mathcal{O})}[\log \mathrm{p}_\theta(Y \mid X)] \tag{76}$$

$$= \mathbb{E}_{\mathrm{p}(x,y|z)\mathrm{p}_q(z)}[\log \mathrm{p}_\theta(Y \mid X)] \tag{77}$$

where z is an indicator variable for $y \in \mathcal{O}$ and $\mathrm{p}_q(z)$ is the Bernoulli distribution with parameter q.

We next show the expected log likelihood is maximized at θ_0.

$$\operatorname*{argmax}_{\theta\in\Theta} L_q(\theta) = \operatorname*{argmax}_{\theta\in\Theta} \mathbb{E}_{\mathrm{p}(x,y|z)\mathrm{p}_q(z)}[\log \mathrm{p}_\theta(Y \mid X) + \log p(X)] \tag{78}$$

$$= \operatorname*{argmax}_{\theta\in\Theta} \mathbb{E}_{\mathrm{p}(x,y|z)\mathrm{p}_q(z)}[\log \mathrm{p}_\theta(X, Y \mid Z) + \log \mathrm{p}(Z)] \tag{79}$$

$$= \operatorname*{argmax}_{\theta\in\Theta} \mathbb{E}_{\mathrm{p}(x,y|z)\mathrm{p}_q(z)}[\log \mathrm{p}_\theta(X, Y \mid Z)] \tag{80}$$

$$= \operatorname*{argmax}_{\theta\in\Theta} q\mathbb{E}_{\mathrm{p}(x,y|y\in\mathcal{O})}[\log \mathrm{p}_\theta(X, Y \mid y \in \mathcal{O})] + (1-q)\mathbb{E}_{\mathrm{p}(x,y|y\notin\mathcal{O})}[\log \mathrm{p}_\theta(X, Y \mid y \notin \mathcal{O})] \tag{81}$$

$$= \operatorname*{argmin}_{\theta\in\Theta} q\mathrm{KL}(\mathrm{p}(x, y \mid y \in \mathcal{O})\|\mathrm{p}_\theta(x, y \mid y \in \mathcal{O})) + (1-q)\mathrm{KL}(\mathrm{p}(x, y \mid y \notin \mathcal{O})\|\mathrm{p}_\theta(x, y \mid y \notin \mathcal{O})) \tag{82}$$

where in the first line we use that $\mathrm{p}(x)$ does not depend on θ, the second line is the chain rule, and the third line uses Assumption 9 which implies $p_\theta(y \in \mathcal{O}) = \mathrm{p}(y \in \mathcal{O})$ is independent of θ.

When $q = 1$ only the first term appears. Its minimum is reached when $\mathrm{p}(x, y \mid y \in \mathcal{O}) = \mathrm{p}_\theta(x, y \mid y \in \mathcal{O})$, which by Lemma 2 implies $\mathrm{p}_\theta(y \mid x) = \mathrm{p}(y \mid x)$. By Assumption 1, this occurs at $\theta = \theta_0$.

When $q < 1$, the second term of Equation (82) appears, which is minimized at $\mathrm{p}(x, y \mid y \notin \mathcal{O}) = \mathrm{p}_\theta(x, y \mid y \notin \mathcal{O})$. We can apply the same argument as in the proof Lemma 2 to show that $\mathrm{p}(y \mid x)$ is identified. Briefly, exchanging $\mathcal{O}$ with $\mathcal{O}^c$ in Equation (68), S_0 is identified from the fact that $\mathrm{p}(y \in \mathcal{O}^c \mid x) < 1$ for $x \in S_0$ and $\mathrm{p}(y \in \mathcal{O}^c \mid x) = 1$ for $x \notin S_0$. Then, exchanging $\mathcal{O}$ with $\mathcal{O}^c$ in Equation (69), we can identify $\mathrm{p}(y \mid x)$ within $x \in S_0$ by the same exponential family argument. So the minimizer is again obtained at $\theta = \theta_0$.

We conclude that $L_q(\theta)$ is minimized at θ_0. The conclusion then follows from Theorem 3.2 in [Miller, 2021]. □

B.7.3 Efficiency

We next characterize efficiency. Under the conditions of Proposition 8, the Bernstein-von Mises theorem applies (Theorem 3.2 in [Miller, 2021]), with the information matrix

$$H_q = -\mathbb{E}_{p(x,y|z)p_q(z)}[\nabla^2_\theta \log \mathrm{p}_\theta(Y \mid X)|_{\theta=\theta_0}] \tag{83}$$

where again we use $p(x, y \mid z)$ as shorthand for $p(x, y \mid \mathbb{I}(y \in \mathcal{O}) = z)$, i.e. z is an indicator variable for whether or not $y \in \mathcal{O}$.

As in the binary y case, we find that when activity is sparse, i.e. η is small, then the asymptotic posterior entropy is minimized at $q = 1$. So, we maximize information gain by just collecting samples with $y \in \mathcal{O}$.

Proposition 9. *Define*

$$\mathcal{I}_z \triangleq -\mathbb{E}_{\mathrm{p}(x,y|S_0,z)}[\nabla^2_\theta \log \mathrm{p}_\theta(Y \mid X)|_{\theta=\theta_0}]. \tag{84}$$

Then Proposition 3 holds with H_q defined as in Equation (83).

Proof. The proof is essentially the same as that of Proposition 3 (Section B.3). We have

$$H_q = -q\mathbb{E}_{\mathrm{p}(x,y|y\in\mathcal{O})}[\nabla^2_\theta \log \mathrm{p}_\theta(Y \mid X)|_{\theta=\theta_0}] - (1-q)\mathbb{E}_{\mathrm{p}(x,y|y\notin\mathcal{O})}[\nabla^2_\theta \log \mathrm{p}_\theta(Y \mid X)|_{\theta=\theta_0}] = q\mathcal{I}_1 + (1-q)\mathrm{p}(S_0 \mid y \notin \mathcal{O})\mathcal{I}_0. \tag{85}$$

Following the same logic as in Equation (43), using the sparse activity model (Assumption 7), we have $\mathrm{p}(S_0 \mid y \notin \mathcal{O}) \leq \eta/(1-\eta)$. Then, using the same logic as in the proof of Proposition 3, we find that $\det H_1 > \det H_q$ for $0 \leq q < 1$ when Equation (15) holds. The conclusion follows. □

B.7.4 Model evaluation

We suggest a two stage procedure for checking a model $\mathrm{p}_\theta(y \mid x)$.

First, evaluate $\mathrm{p}_\theta(y \in \mathcal{O} \mid x)$. This is a classifier, and so can be evaluated using the tools in Section 3.3, using both the (x, y) pair data and the z data.

Second, evaluate $\mathrm{p}_\theta(y \mid x, y \in \mathcal{O})$. This requires checking the model's fit just on the collected (x, y) pairs, setting aside the z data. Here we can apply tools for evaluating regression models, e.g. we can compute an R^2 on held-out (x, y) pairs, look at a Q-Q plot of the residuals, or apply a full nonparametric goodness-of-fit test.

The procedure is valid because if both checks pass, the model must match $\mathrm{p}(y \mid x)$, at least asymptotically. In particular, if $\mathrm{p}_\theta(y \in \mathcal{O} \mid x) = \mathrm{p}(y \in \mathcal{O} \mid x)$ for $x \in \mathcal{X}$, and if $\mathrm{p}_\theta(y \mid x, y \in \mathcal{O}) = \mathrm{p}(y \mid x, y \in \mathcal{O})$ for $x \in \mathcal{X}$ and $y \in \mathcal{O}$, then we have the correct $\mathrm{p}(y \mid x)$ for $x \in \mathcal{X}$ and $y \in \mathcal{O}$. This implies that $\mathrm{p}_\theta(y \mid x) = \mathrm{p}(y \mid x)$ for all $y \in \mathcal{Y}$ under Assumption 8, using the identification argument in Lemma 2.

B.8 Learning from x-only data

In this section we describe a general method to combine data sampled from $\mathrm{p}(x)$ and an approximate model to estimate $\int \mathrm{p}_\theta(y \mid x)\mathrm{p}(x)dx$ when $\mathrm{p}(x)$ is unknown. The method bottoms out in approximating the integral with a mix of samples from the model and samples from the data. The tradeoff is determined by how much data is available, and the scientist's prior beliefs about how much to trust the model.

We consider the posterior mean of the marginal likelihood of $y = 1$ under the model,

$$\mathrm{p}(x) \sim \mathrm{DirichletProcess}(\mathrm{p}_{model}(x), \alpha) \tag{86}$$

$$x'_1, \ldots, x'_M \overset{iid}{\sim} \mathrm{p}(x) \tag{87}$$

By conjugacy, the posterior over $\mathrm{p}(x)$ is $\mathrm{DP}(\frac{\alpha}{\alpha+M}\mathrm{p}_{model}(x) + \frac{M}{\alpha+M}\hat{\mathrm{p}}_M(x), \alpha + M)$ where $\hat{\mathrm{p}}_M(x) = \frac{1}{M}\sum_{i=1}^{M}\delta_{x'_i}(x)$ is the empirical distribution of the data [Teh, 2010]. We can compute the posterior mean of the marginal likelihood as

$$\mathbb{E}[\int \mathrm{p}_\theta(y \mid x)\mathrm{p}(x)dx \mid x'_{1:M}] = \int \mathrm{p}_\theta(y \mid x)\mathbb{E}[\mathrm{p}(x) \mid x'_{1:M}]dx \tag{88}$$

$$= \int \mathrm{p}_\theta(y \mid x)\left(\frac{\alpha}{\alpha+M}\mathrm{p}_{model}(x) + \frac{M}{\alpha+M}\hat{\mathrm{p}}_M(x)\right)dx \tag{89}$$

where we have used the fact that the marginal likelihood is a linear functional of $\mathrm{p}(x)$, and plugged in the posterior mean of the Dirichlet process. We can approximate this integral by Monte Carlo, drawing a fraction $\frac{\alpha}{\alpha+M}$ of the samples from the model and the rest from the sequencing data. With no data, we just use samples from the model, and we recover the standard LeaVS approach. As data accumulates, the posterior will concentrate at the true value of $\mathrm{p}(x)$, and the approximation will correspondingly be dominated by the contribution of the data.

C Empirical Results

C.1 Models and training

In all three datasets, we used a shallow single-layer CNN with a kernel size of 5, stride 1, 32 channels, and a 16-dimensional initial embedding for the individual amino acids. Where possible (in the synthetic results and the large-scale demonstration on antibodies) we used a batch size of 128 positive examples; for the TCR dataset, due to low n, we used a batch size of 16 positive examples. For the LeaVS marginal likelihood correction we draw $M = 19 \times$ `batch-size` samples from $\mathrm{p}(x)$, i.e. $M = 2{,}432$ for the synthetic and antibody demonstrations, and $M = 304$ for the TCR dataset. The models were trained for 1000 epochs on a single NVIDIA H100 GPU (about 6 minutes per dataset). For the large-scale demonstration on antibodies, we additionally trained a transformer-based model: encoder-only, depth 8, attention dimension 16, 16 eight-dimensional attention heads.

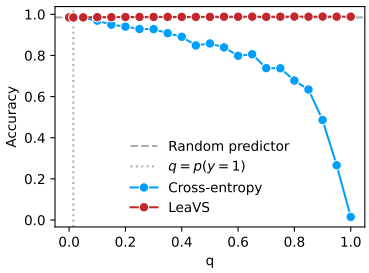


Figure S1: Accuracy on synthetic data (Figure 2a) with y-axis extended to zero.

C.2 Synthetic data

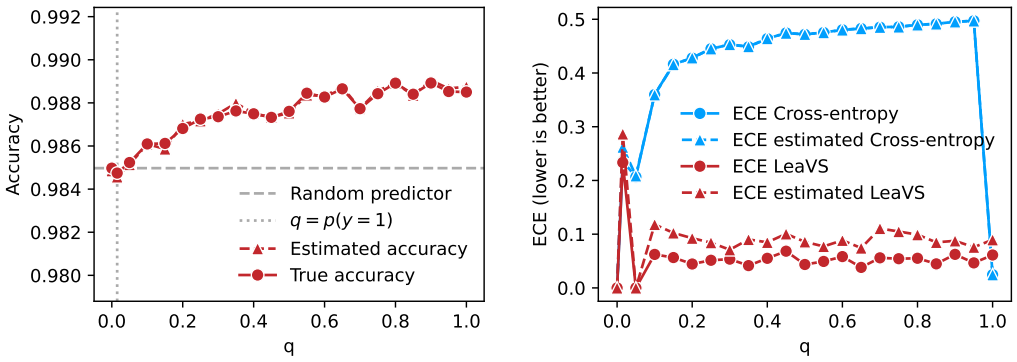
In the synthetic data setting, we set $\mathrm{p}(x)$ to be a variational synthesis model of antibody CDRH3 amino acid sequences [Weinstein et al., 2026]. We set $\mathrm{p}(y \mid x)$ so that y depends on the presence of specific amino acid sequences at specific positions. Namely, $\mathrm{p}(y \mid x)$ is defined as follows: given a sequence x, we set its 'base strength' $s(x)$ to 1. Then, we check for the presence of the following substrings: (i) `"P"` or `"C"` at position 3 (zero-indexed), (ii) `"N"` or `"C"` at position 5, and (iii) `"PC"` or `"SS"` starting at position 6. The presence of each substring increases the strength $s(x)$ by a multiplier of 10. We then add noise by sampling the final $c(x)$ value from a negative binomial distribution with a mean of $s(x)$ and variance $s(x) + \frac{s(x)^2}{\phi}$ (where $\phi = 2.28$), and set $y = \mathbb{I}[c(x) > 30]$ — so that a sequence is required to have at least two of the substrings for its expected label to be positive, but some of the sequences that only have a single pattern may be labeled positive as well due to added stochasticity. In this setting $\mathrm{p}(y = 1) = 0.015$. This process, including the choice of ϕ, was designed based on the experimental conditions in the antibody demonstration. We used $n = 2000$ observed sequences for training, with q ranging from 0 to 1 with a step of 0.05, with an additional $q = \mathrm{p}(y = 1)$. For testing, we used a set of 400,000 sequences sampled from $\mathrm{p}(x)$. For evaluating the estimated accuracy and calibration, we split this set into halves, and used only the positive examples from one half to represent our observed set, while the other half represented the unlabeled set (the library). We used 10 equal-sized bins for evaluation of the the estimated calibration error.

We repeated the simulations with non-binarized y, by instead setting $y = c(x)$. We define a hit as $y \in \mathcal{O} = \{y : y > 30\}$. We replace the logit output of the model with a negative binomial distribution. We compute $\mathrm{p}(y \notin \mathcal{O} \mid x)$ in closed form to obtain the marginal likelihood. To compute precision and recall for a non-binary outcome (Figure S3), we use a sliding threshold: we replace Y with $\mathbb{I}(Y > \tau)$ in Section A.1, where τ is the same threshold as for the predictor, $t_\theta(x) = \mathbb{I}(\mathbb{E}_\theta[Y \mid x] > \tau)$.

To explore higher values of $\mathrm{p}(y = 1)$, we simulated data as follows. Let $\tilde{\mathrm{p}}(x)$ denote the distribution over peptides defined by first sampling the length uniformly $10 \leq l \leq 30$, then sampling each position independently from the uniform distribution over the 20 amino acids. Let $s(x)$ be an indicator for whether x has any of the following 3-mers: `"ACD"`, `"KLM"`, `"QRS"`, `"FGH"`, `"VWY"`. For a given value of $\mathrm{p}(y = 1)$ (0.25 or 0.5 in practice) we set $\mathrm{p}(x, y) = \tilde{\mathrm{p}}(x \mid s(x) = y)\mathrm{p}(y)$. We draw samples from $\tilde{\mathrm{p}}(x \mid s(x) = y)$ by rejection sampling. We use $n = 100$ observed sequences for training and 10,000 for the test set. For these particular experiments we trained a single-layer CNN with 60 filters, a kernel size of 3, and global max pooling. We use a batch size of 20 observed sequences, and draw $M = 320$ samples from $\mathrm{p}(x)$ to estimate the LeaVS marginal likelihood.

C.3 Experimental TCR data

For the experimental data setting, we used a dataset of T cell receptors (TCRs) from human CD8+ T cells, screened for binding against multiple targets, publicly available at `https://www.10xgenomics.com/datasets/cd-8-plus-t-cells-of-healthy-donor-1-1-standard-3-0-2` [Genomics, 2019]. We focused on a single target – the influenza antigen. In this dataset, we have measurements of (x, y) for each example but we do not have control over $\mathrm{p}(x)$, as the sequences come from a human patient. Instead, we split the dataset into halves and used the empirical distribution of one half as an approximation of $\mathrm{p}(x)$ (note this strategy would also be feasible in practice, since sequencing TCRs is substantially easier than obtaining TCR-binding activity measurements). We used the other half of the complete dataset to generate smaller datasets with varying values of q. Note that



(a) Accuracy.

(b) Expected calibration error.

Figure S2: Evaluation estimates using heldout positives-only data, on the synthetic data.

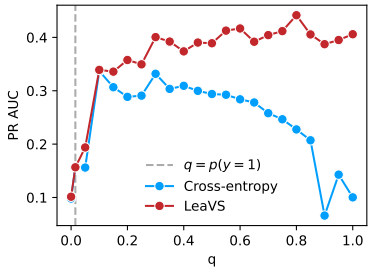
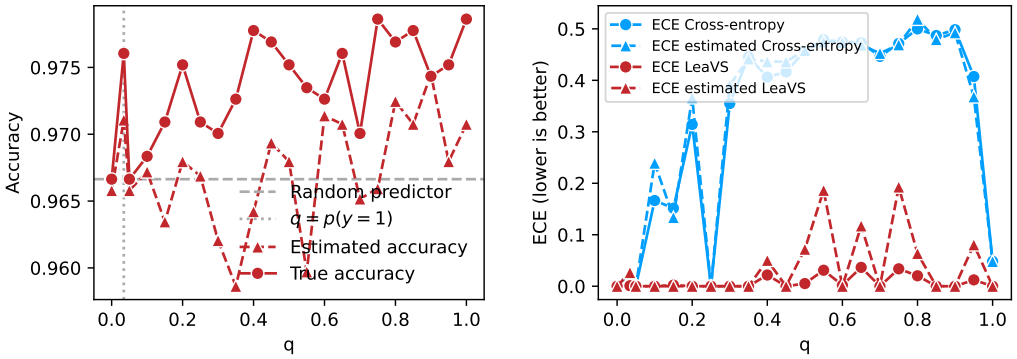


Figure S3: Area under the precision recall curve on synthetic data, training on counts rather than a binarized value (as in Figure 2).

because we must subsample a full dataset and $p(y) = 0.034$, n is relatively small, which increases variability. In total, there were 400 positive cells, 10% (40 positive cells) of the full dataset is reserved for the heldout set, which leaves 360 cells. The full training set was split in halves (without stratifying by label) as described above — which left us with $n = 170$ positive cells in the training set.

C.4 Large-scale demonstration

For the large-scale demonstration, we used an experimental dataset of 9000 observed antibody CDRH3 sequences, paired with integer-valued binding measurements against 9 antigens – these sequences are the outcome of a high-throughput screen of a variational synthesis library of antibody CDRH3 amino acid sequences [Weinstein et al., 2026]. Each observed sequences is positive against at least one of the targets. We used 10% of the data for the heldout set, and we used the corresponding variational synthesis model as $\mathrm{p}(x)$. We use a negative binomial likelihood in $\mathrm{p}_\theta(y \mid x)$. To compute precision and recall for a non-binary outcome Y, we use a sliding threshold: we replace Y with $\mathbb{I}(Y > \tau)$ in Section A.1, where τ is the same threshold as for the predictor, $t_\theta(x) = \mathbb{I}(\mathrm{p}_\theta(y = 1 \mid x) > \tau)$.



(a) Accuracy.

(b) Expected calibration error.

Figure S4: Evaluation estimates using heldout positives-only data, on the experimentally measured TCR data.

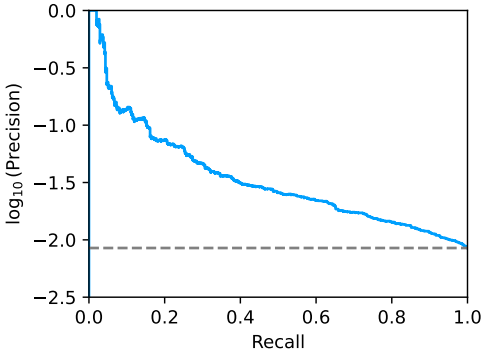
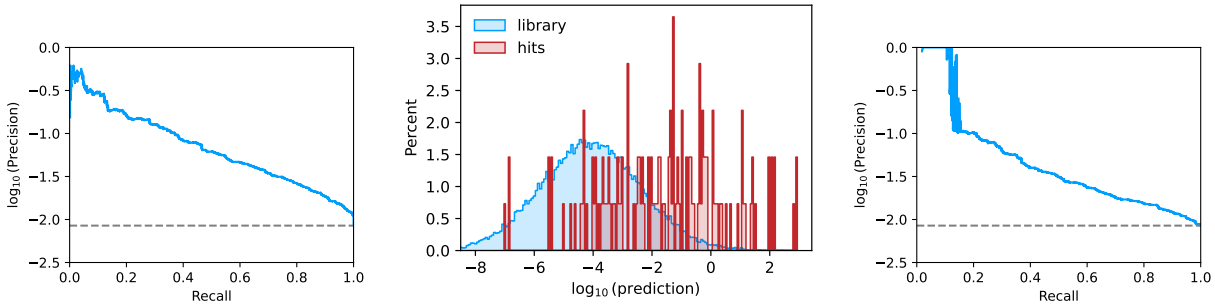


Figure S5: Predicting binding of TCR mimicking scFv CAR therapeutic candidates to MAGE-A4, a challenging oncology target. Here we plot the precision recall curve using sequenced samples from the variational synthesis library as samples from p(x), rather than samples from the computational library model as in Figure 4a. The AUC is 0.07.



(a) Precision-recall. AUC: 0.10.

(b) Predictive scores on library sequences (blue) versus heldout positive examples (red).

(c) Precision-recall using sequenced x in place of computational samples from the synthesis model. AUC: 0.14.

Figure S6: Predicting binding of TCR mimicking scFv CAR therapeutic candidates to MAGE-A4, a challenging oncology target. Here we use a CNN-based architecture, rather than a transformer as in Figure 4.