LEARNING PROTEIN FAMILY MANIFOLDS WITH SMOOTHED ENERGY-BASED MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

We resolve difficulties in training and sampling from discrete energy-based models (EBMs) by learning a smoothed energy landscape, sampling the smoothed data manifold with Langevin Markov chain Monte Carlo, and projecting back to the true data manifold with one-step denoising. Our formalism combines the attractive properties of EBMs and improved sample quality of score-based models, while simplifying training and sampling by requiring only a single noise scale. We demonstrate the robustness of our approach on generative modeling of antibody proteins.

1 INTRODUCTION

Discrete sequence generation poses a number of challenges to gradient-based generative models. Generative models must be expressive enough to faithfully capture the underlying data distribution, while also having controllable outputs that are novel, unique, diverse, and respect the constraints of the problem space. Energy-based models (EBMs) (Hinton & Sejnowski, 1986; LeCun et al., 2006) fit an energy function that specifies a probability distribution over data analogous to the Boltzmann distribution from statistical physics. Giving access to an easily computable energy is an advantage of EBMs, but on the flip-side they can be difficult to train and sample from. Denoising objectives based on score matching (Hyvärinen, 2005; Vincent, 2011) and the related advancements in diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020) overcome these issues, but these either model the energy gradient or only provide access to an empirical lower-bound of the likelihood.

Protein design is an instance of the discrete sequence generation problem, wherein the challenge is to find useful proteins in the large, discrete, and sparsely functional space (Romero & Arnold, 2009) of dimension 20^L for proteins of length L. Here, we consider the specific problem of deep generative modeling of antibody molecules, a class of proteins with highly conserved structure that are of immense interest for therapeutics. In addition to the qualities mentioned above, generative models for antibodies must be sample-efficient because of the relatively small size of datasets with therapeutic antibodies. Antibodies consist of well-conserved domains and high entropy variable regions, so leveraging evolutionary information from pre-trained protein language models is not an immediate solution. To this end, we introduce *Walk-jump discrete energy-based models* (WJDEBMs), a method building on the neural empirical Bayes (NEB) (Saremi & Hyvärinen, 2019) formalism that addresses the brittleness of discrete EBMs and in doing so, provides a robust and general framework for protein design.

2 RELATED WORK

Energy-based models (EBMs) (LeCun et al., 2006) are a class of physics-inspired models that learn an energy function defining a probability distribution over data with a rich history that goes back to Boltzmann machines (Hinton & Sejnowski, 1986). Contrastive divergence (Hinton, 2002) training using Markov-Chain Monte Carlo (MCMC) was proposed to estimate the gradient of the log partition function, wherein input data is usually discrete and MCMC chains are initialized from training data, leading to long mixing times in high dimensions. Using continuous inputs and gradient-based MCMC (Langevin dynamics) initialized from uniform noise with a replay buffer of past samples, efficient training was achieved (Du & Mordatch, 2019). The Langevin MCMC approach to sampling and maximum likelihood training yield advantages in simplicity (only one network is trained),

flexibility (no constraints imposed by a prior distribution), and compositionality (energy functions can be summed).

Estimating unnormalized densities can also be formulated using score matching (Hyvärinen, 2005). This formulation led to probabilistic models for denoising autoencoders (Vincent, 2011; Alain & Bengio, 2014; Saremi et al., 2018), but also has an empirical Bayes interpretation that is most related to this work. The neural empirical Bayes (NEB) (Saremi & Hyvärinen, 2019) formalism unifies kernel density estimation (Parzen, 1962) and empirical Bayes (Robbins, 1956) to transform the unsupervised learning problem into a more tractable form where a neural network energy function is parameterized to capture a *smoothed* data distribution. NEB gives rise to the "walk-jump" sampling (WJS) machinery, wherein sampling from a generative model is performed by "walking" (Langevin MCMC) in the smoothed landscape and "jumping" (empirical Bayes) to the true data distribution by denoising.

Although generative modeling is widely adopted in image and natural language generation, successful applications of generative modeling in the sciences are few and far between, due to the over-representation of image and text datasets, challenges in evaluation, and the need for generating samples that are novel and diverse while respecting the underlying symmetries and structure of a particular domain. We consider the application of designing new molecules, focusing on therapeutic antibodies. Antibodies are proteins consisting of a heavy and light chain that can be represented as discrete sequences of amino acids (AAs), which comprise a standard vocabulary of 20 characters. Approaches borrowing from traditional ML generative modeling have been used to model antibodies (Shuai et al., 2021; Gligorijević et al., 2021; Ferruz & Höcker, 2022; Tagasovska et al., 2022), but typical natural-language-based methods struggle to capture the data distribution of antibodies, for which there is limited training data ($\sim$100K high-quality sequences) and additional challenges due to the high-entropy variable regions of the sequence. Here, we address the above challenges with training and sampling discrete sequences from EBMs using a formulation of WJS.

3 METHODS

3.1 ENERGY-BASED MODELS

EBMs are a class of models that learn an energy function $f_\theta : \mathcal{X} \to \mathbb{R}$ mapping inputs x (in $\mathbb{R}^d$) to a scalar "energy" value. The data distribution is approximated by the Boltzmann distribution

$$p_\theta(x) \propto e^{-f_\theta(x)}.$$

EBMs are typically trained via contrastive divergence (Hinton, 2002) and new samples are drawn from $p_\theta(x)$ by Markov-Chain Monte Carlo (MCMC). Details of the loss function used in this work are given in Section A.1. In Langevin MCMC, samples are initialized from a known data point or random noise and refined with (discretized) Langevin diffusion

$$x_{k+1} = x_k - \delta \, \nabla f_\theta(x_k) + \sqrt{2\delta} \, \varepsilon_k, \; \varepsilon_k \sim \mathcal{N}(0, I_d), \tag{1}$$

where ∇ denotes the gradient of the energy function with respect to inputs, k is the sampling step, δ is the (discretization) step size, and the noise ε_k is drawn from the normal distribution at each step.

3.2 NEURAL EMPIRICAL BAYES AND WALK-JUMP SAMPLING

In NEB, the random variable X is transformed with additive Gaussian noise $Y = X + \mathcal{N}(0, \sigma^2 I_d)$. The least-squares estimator of X given $Y = y$ is given by (Robbins, 1956; Miyasawa, 1961)

$$\widehat{x}(y) = y + \sigma^2 \nabla \log p(y), \tag{2}$$

where $p(y) = \int p(y|x)p(x)dx$ is the probability distribution function of the smoothed density.[1] The estimator is expressed purely in terms of $g(y) = \nabla \log p(y)$ known as the score function Hyvärinen (2005) which is parametrized with a neural network denoted by $g_\phi : \mathbb{R}^d \to \mathbb{R}^d$. The least-squares estimator then takes the following parametric form:

$$\widehat{x}_\phi(y) = y + \sigma^2 g_\phi(y). \tag{3}$$

[1] We follow the convention $p(x) \coloneqq p_X(x), p(y) \coloneqq p_Y(y)$, etc.

Putting this all together leads to the following learning objective

$$\mathcal{L}(\phi) = \mathbb{E}_{x \sim p(x), y \sim p(y|x)} \|x - \widehat{x}_\phi(y)\|^2, \tag{4}$$

which is optimized with stochastic gradient descent. Notably, no MCMC sampling is required during learning. In short, the objective is "learning to denoise" with an empirical Bayes formulation. Following learning one can sample noisy data using the learned score function $g_\phi(y)$ with Langevin MCMC (replace $-\nabla f$ with g in Eq. 1). For any such draws y_k, clean samples from the true data manifold $\mathcal{M}$ are obtained by "jumping" back to $\mathcal{M}$ with the least-squares estimator $\widehat{x}_\phi(y_k) = y_k + \sigma^2 g_\phi(y_k)$. This is the walk-jump sampling (WJS) scheme. A key property of WJS is the fact that the least-squares estimation (jump) is *decoupled* from the Langevin MCMC (walk).

Here, we take advantage of this decoupling to train an EBM with maximum likelihood estimation on the smoothed distribution of noisy sequences, generate noisy samples with Langevin MCMC, and denoise samples with a separately trained neural network, the least-squares estimator.

4 RESULTS

We report results for our method, walk-jump discrete energy based model (WJDEBM) (Fig. 1), on modeling the data distribution of antibodies and alleviating the difficulties of training and sampling from EBMs for discrete sequences. We also present surprising results related to EBM parameter optimization with maximum likelihood training on noisy data that suggest a more general and robust training procedure for EBMs. Details related to model architectures, training, and sequence sampling are in Section A.1

4.1 WJDEBM CAPTURES THE ANTIBODY DISTRIBUTION MEASURED BY SYNTHETIC BENCHMARKS

We measure generative model performance with a suite of "antibody likeness" (ab-likeness) metrics including labels derived from the AA sequence with Biopython (Cock et al., 2009), a sequence similarity score from sequence alignments with DIAMOND (Buchfink et al., 2021), Levenshtein edit distances calculated with Edlib (Šošić & Šikić, 2017), and a naturalness metric (Bachas et al., 2022) computed from the likelihoods of a masked language model pre-trained on antibody sequences. Sequence property metrics are condensed into a single scalar metric by computing the normalized average Wasserstein distance W_{property} between the property distributions of samples and a validation set. The average total edit distance E_{dist} summarizes the novelty and diversity of samples compared to the validation set. The results summarized in Table 1 show that with increasing noise level, σ, better agreement is reached between the sample property distributions and the validation set. The average total edit distance also increases monotonically with increasing σ, reflecting the improved sequence novelty/diversity and mode exploration. Distributions of DIAMOND similarity (Fig. A.2) and naturalness (Fig. A.2) metrics indicate that natural sequences are generated with reasonable similarity to the training set, while maintaining diversity and novelty.

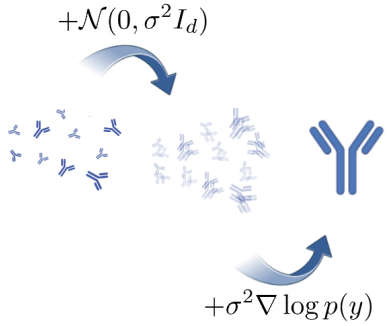

Figure 1: Walk-jump discrete energy based model samples from the distribution of noisy sequences and jumps back to the clean data manifold.

4.2 WJDEBM STABILIZES TRAINING

We observe that the WJDEBM formalism prevents instabilities during maximum likelihood training. EBMs commonly exhibit issues with training stability and divergences in the energy, due to the energy landscape becoming too complicated to sample. Noising the data provides strong regularization that prevents overfitting and instabilities. This is seen over a range of noise levels $\sigma \in (0.1, 1.0]$. For $\sigma = 0$, samples have very few edits (<3.5 on average) compared to training data, but they are highly nonphysical changes that yield disagreement between the biophysical property distributions (Table 1).

Table 1: Ablikeness metrics. Heavy chain (light chain) metrics are reported separately. The score function implemented in the EBM training, denoted by `score`, is taken to be either as $\pm\nabla f_\theta(y)$

σ	$W_{\text{property}} \downarrow$	$E_{\text{dist}} \uparrow$
`score` $\leftarrow -\nabla f_\theta(y)$		
0	0.31 (0.31)	2.3 (3.3)
0.1	0.17 (0.20)	4.9 (4.8)
0.5	0.08 (0.10)	16.5 (16.3)
1.0	**0.07 (0.08)**	**33.5 (40.3)**
`score` $\leftarrow +\nabla f_\theta(y)$		
0	0.31 (0.31)	2.3 (3.3)
0.1	0.17 (0.18)	5.0 (5.0)
0.5	0.10 (0.11)	16.5 (17.4)
1.0	**0.07 (0.10)**	**30.9 (40.2)**

4.3 WJDEBM SIMPLIFIES AND GENERALIZES CONTRASTIVE TRAINING

We investigate the effects of discarding many of the techniques for improved EBM training that, while introduced to ameliorate challenges with EBMs, also introduce complexities that make EBMs brittle, inflexible, and difficult to optimize. In particular, we discard the replay buffer, Metropolis rejection sampling, and time step annealing. We use the Langevin MCMC algorithm from (Sachs et al., 2017) and eliminate the need for careful hyperparameter finetuning; σ is the only free hyperparameter in WJDEBM. Similarly, we use a simplified 1D CNN architecture for our EBM that further eliminates complexity and training difficulties. Simplifying training in this way greatly accelerates convergence, leading to converged models in $\sim$50 training steps, with no hyperparameter tuning or adjustments needed to train on different data distributions. Our method achieves strong ablikeness results (Table 1, Figs A.2 and A.2), simply by increasing σ to an optimal value.

Surprisingly, even changing the sign in the score function for Langevin MCMC (Table 1) during contrastive training does not affect the robustness of the results. We observe that changing the sign leads to some instability early in training, but does not affect the convergence behavior or ablikeness results. This suggests that further investigation is needed to uncover the truly essential components of maximum likelihood training with Langevin MCMC.

5 CONCLUSIONS AND FUTURE WORK

We proposed the "Walk-jump discrete energy based model" (WJDEBM), a generative model for discrete sequences that uses Langevin Markov chain Monte Carlo to sample from smoothed data distributions. We evaluate our approach on the antibody design problem, showing the capability of our method to generate novel, diverse, and functional antibodies as measured by synthetic biophysical property distributions and similarity metrics. The strong regularization provided by fitting the energy function to noisy data completely prevents overfitting, even on small datasets of $<$1000 samples, resulting in fast and efficient retraining and sampling with low compute requirements. WJDEBM discards many of the commonly used techniques for improving EBM training with Langevin MCMC (replay buffers, simulated annealing, rejection sampling, etc.) and reduces the engineering complexity of training EBMs and diffusion-based models to a single hyperparameter choice: the noise level, σ. Intriguingly, we find that even inverting the sign of the score function in Langevin MCMC does not deteriorate sample quality or impede training. Altogether, our results suggest a simplified, more general and robust framework for training and sampling from discrete energy based models with applications to therapeutic molecule design. Future work will probe the generality of our results to other classes of molecules and even other data modalities (e.g., images), as well as theoretical investigation into the results presented here.

REFERENCES

Guillaume Alain and Yoshua Bengio. What regularized auto-encoders learn from the data-generating distribution. *Journal of Machine Learning Research*, 15(1):3563–3593, 2014.

Sharrol Bachas, Goran Rakocevic, David Spencer, Anand V Sastry, Robel Haile, John M Sutton, George Kasun, Andrew Stachyra, Jahir M Gutierrez, Edriss Yassine, et al. Antibody optimization enabled by artificial intelligence predictions of binding affinity and naturalness. *bioRxiv*, 2022.

Benjamin Buchfink, Klaus Reuter, and Hajk-Georg Drost. Sensitive protein alignments at tree-of-life scale using diamond. *Nature methods*, 18(4):366–368, 2021.

Peter JA Cock, Tiago Antao, Jeffrey T Chang, Brad A Chapman, Cymon J Cox, Andrew Dalke, Iddo Friedberg, Thomas Hamelryck, Frank Kauff, Bartek Wilczynski, et al. Biopython: freely available python tools for computational molecular biology and bioinformatics. *Bioinformatics*, 25(11):1422–1423, 2009.

Yilun Du and Igor Mordatch. Implicit generation and modeling with energy based models. *Advances in Neural Information Processing Systems*, 32, 2019.

James Dunbar and Charlotte M Deane. Anarci: antigen receptor numbering and receptor classification. *Bioinformatics*, 32(2):298–300, 2016.

Noelia Ferruz and Birte Höcker. Controllable protein design with language models. *Nature Machine Intelligence*, 4(6):521–532, 2022.

Vladimir Gligorijević, Daniel Berenberg, Stephen Ra, Andrew Watkins, Simon Kelow, Kyunghyun Cho, and Richard Bonneau. Function-guided protein design by deep manifold sampling. *bioRxiv*, 2021.

Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

Geoffrey E Hinton and Terrence J Sejnowski. Learning and relearning in Boltzmann machines. *Parallel distributed processing: Explorations in the microstructure of cognition*, 1(282-317):2, 1986.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Annemarie Honegger and Andreas PluÈckthun. Yet another numbering scheme for immunoglobulin variable domains: an automatic modeling and analysis tool. *Journal of molecular biology*, 309 (3):657–670, 2001.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(Apr):695–709, 2005.

Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aaron van den Oord, Alex Graves, and Koray Kavukcuoglu. Neural machine translation in linear time. *arXiv preprint arXiv:1610.10099*, 2016.

Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, and Fujie Huang. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Koichi Miyasawa. An empirical Bayes estimator of the mean of a normal population. *Bulletin of the International Statistical Institute*, 38(4):181–188, 1961.

Tobias H Olsen, Fergus Boyles, and Charlotte M Deane. Observed antibody space: A diverse database of cleaned, annotated, and translated unpaired and paired antibody sequences. *Protein Science*, 31(1):141–146, 2022.

Emanuel Parzen. On estimation of a probability density function and mode. *The annals of mathematical statistics*, 33(3):1065–1076, 1962.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

Herbert Robbins. An empirical Bayes approach to statistics. In *Proc. Third Berkeley Symp.*, volume 1, pp. 157–163, 1956.

Philip A Romero and Frances H Arnold. Exploring protein fitness landscapes by directed evolution. *Nature reviews Molecular cell biology*, 10(12):866–876, 2009.

Matthias Sachs, Benedict Leimkuhler, and Vincent Danos. Langevin dynamics with variable coefficients and nonconservative forces: from stationary states to numerical methods. *Entropy*, 19(12): 647, 2017.

Saeed Saremi and Aapo Hyvärinen. Neural empirical Bayes. *Journal of Machine Learning Research*, 20:1–23, 2019.

Saeed Saremi, Arash Mehrjou, Bernhard Schölkopf, and Aapo Hyvärinen. Deep energy estimator networks. *arXiv preprint arXiv:1805.08306*, 2018.

Richard W Shuai, Jeffrey A Ruffolo, and Jeffrey J Gray. Generative language modeling for antibody design. *bioRxiv*, 2021.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pp. 2256–2265. PMLR, 2015.

Martin Šošić and Mile Šikić. Edlib: a c/c++ library for fast, exact sequence alignment using edit distance. *Bioinformatics*, 33(9):1394–1395, 2017.

Nataša Tagasovska, Nathan C Frey, Andreas Loukas, Isidro Hötzel, Julien Lafrance-Vanasse, Ryan Lewis Kelly, Yan Wu, Arvind Rajpal, Richard Bonneau, Kyunghyun Cho, et al. A pareto-optimal compositional energy-based model for sampling and optimization of protein sequences. *arXiv preprint arXiv:2210.10838*, 2022.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.

A Appendix

A.1 Additional details on experiments

Discrete sequence generation We represent antibody protein molecules as $x = (x_1, \ldots, x_d)$, where $x_l \in \{1, \ldots, 20\}$ corresponds to the AA type at position l. Sequences from the Observed Antibody Space (OAS) database (Olsen et al., 2022) are one-hot encoded and aligned according to the AHo numbering scheme (Honegger & PluÈckthun, 2001) using the ANARCI (Dunbar & Deane, 2016) package. Aligning sequences in this way is a simple solution to handling insertions and deletions, which are otherwise troublesome for models that require fixed length inputs and outputs; alignment introduces a "gap" token that can be introduced or removed during sampling to effectively change the length of sequences. This allows the model to capture the distribution of lengths present in natural antibodies. An EBM is trained via contrastive divergence on the noisy manifold of one-hot encodings. A separate denoising model is trained with the objective in Eq. 4. The EBM is a simple 1D convolutional neural network (CNN) and the denoising model is a ByteNet (Kalchbrenner et al., 2016) architecture trained from scratch. New antibody sequences are generated by sampling noisy samples with Langevin MCMC following gradients from the EBM, denoising with the least-squares estimator, and taking $argmax\ x$ to recover a one-hot encoding.

EBM architecture For all experiments we use an identical architecture consisting of three Conv1D layers with kernel sizes 15, 5, and 3 and padding 1, ReLU non-linearities and an output linear layer of size 128. All EBMs were trained with contrastive training and the AdamW (Loshchilov & Hutter, 2017) optimizer in PyTorch (Paszke et al., 2019).

EBM loss The EBM is trained by maximizing the log-likelihood of *noisy* data under the model:

$$\arg\max_{\theta} \mathbb{E}_{y\sim p_Y}[\log p_\theta(y)] = \arg\max_{\theta} \left(\mathbb{E}_{y^-\sim p_\theta(y)}[f_\theta(y^-)] - \mathbb{E}_{y^+\sim p_Y}[f_\theta(y^+)]\right) \tag{5}$$

With this objective, the model aims to decrease the energy of noisy training data ("positive" samples y^+) while increasing the energy of noisy data sampled from the model ("negative" samples y^-) in expectation. The following identity is behind the positive/negative phases in the EBM training:

$$\begin{aligned}
\nabla_\theta \log p_\theta(y) &= -\nabla_\theta f_\theta(y) - \nabla_\theta \log Z(\theta) \\
&= -\nabla_\theta f_\theta(y) + \frac{\int \nabla_\theta f_\theta(y) e^{-f_\theta(y)} dy}{Z(\theta)} \\
&= -\nabla_\theta f_\theta(y) + \int \nabla_\theta f_\theta(y) \cdot p_\theta(y) dy \\
&= -\nabla_\theta f_\theta(y) + \mathbb{E}_{y\sim p_\theta(y)}[\nabla_\theta f_\theta(y)],
\end{aligned} \tag{6}$$

where $Z(\theta) = \int e^{-f_\theta(y)} dy$ is the partition function. We add an additional ℓ_2 norm penalty to the loss to regularize the energies.

A.2 Additional ablikeness metrics

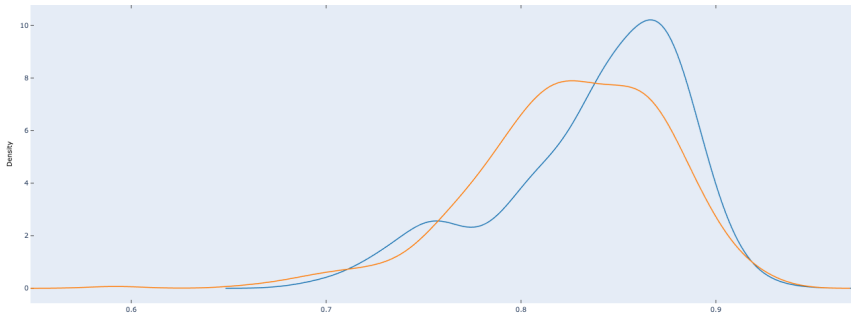

Figure 2: DIAMOND sequence similarity distributions of heavy (light) chain in blue (orange). Samples between 0.8-0.9 indicate good similarity to training set while maintaining novelty and diversity.

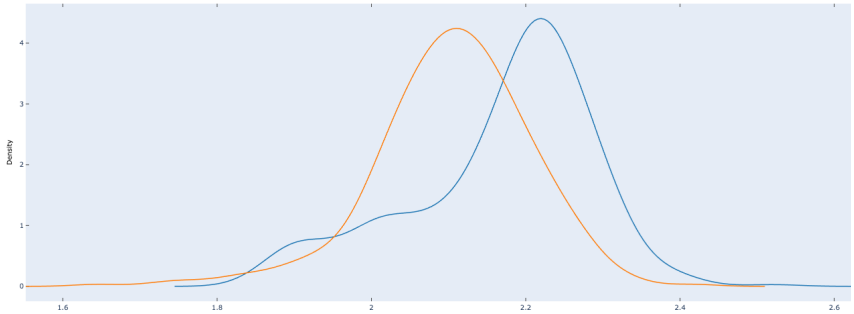

Figure 3: Naturalness distributions of heavy (light) chain in blue (orange). Samples above 2 indicate sequences with high likelihood.