Using Active Learning to Efficiently Navigate a Combinatorial Epistatic Fitness Landscape

Navneedh Maudgalya
Department of Electrical Engineering and Computer Science
University of California, Berkeley
Berkeley, CA 94704
navneedhm@berkeley.edu

Abstract

For machine learning-assisted directed evolution it is important to infer the protein fitness landscape with minimal screening burden. Variants for training must be selected optimally to be diverse and fitness-enriched to learn the topology of the rugged epistatic landscape. To avoid a combinatorial search over all variants, [11] uses zero-shot predictions to discover informative sequences for screening and model training. However, finding an effective heuristic for zero-shot predictions requires background knowledge and is not always possible for a given protein. Hence, we use active learning as an alternative iterative and adaptive data selection strategy. We show that active learning discovers fitness-enriched data efficiently, enabling machine learning models to learn the sequence-fitness mapping as effectively as using simulated zero-shot predictions.

1 Introduction

Protein design is useful for discovering and creating new proteins with desired functions such as catalytic, thermodynamic, or binding activity. Since the space of potential sequences is exponentially large, an exhaustive search to find highly functional variants is infeasible and directed protein evolution provides a more efficient alternative. In traditional directed evolution, we begin with an initial set of variants with some level of desired function. We then iterate through several rounds of mutations and experimental screening to discover mutated sequences with a sufficient level of the desired property. When only considering single-site mutations, the screening burden is minimal when using traditional directed evolution. However, many proteins are affected by epistasis for which we must increase our search space to include variants with mutations at multiple sites. To prevent exhaustive screening of this combinatorial library, machine learning approaches have shown to be effective [13][2].

Machine learning-assisted directed evolution attempts to reduce the search space when selecting new variants to discover highly functional proteins sooner. Sequence-function information can be used to train machine learning models to learn the fitness landscape of the protein, replacing experimental screening. Correctly learning the fitness landscape allows the model to accurately predict function of unscreened variants which can inform decisions about variants to select for the next iteration of evolution. However, learning the fitness landscape is difficult since epistatic effects make the fitness landscape rugged and high fitness variants are generally sparse. Several methods have been proposed to deal with these issues, but in this project I focus on [11] which trains an ensemble of regressors to learn the four-site combinatorial fitness landscape of protein G domain B1 (GB1).

Though the architecture of machine learning models influences its capacity to infer the fitness landscape, the quality of the sequence-function data trained on is equally important. To avoid training on the abundant zero fitness variants and ensure sufficient diversity in the data to learn global topology,

Preprint. Under review.

[11] suggests using zero-shot predictions to filter for fitness-enriched training data. However, zero shot learning strategies are protein-dependent and not always available. Alternatively, active learning does not require any prior information and is commonly used in deep learning models to discover useful data for training. Active learning, described in further detail in Section 2.1, adaptively selects new data to train on using a heuristic that quantifies the utility of a sample for training the model. Active learning typically reduces the number of training samples required while maintaining or improving model accuracy, reducing data storage and labeling costs [7]. In this project, I compare the efficacy of active learning and zero-shot predictions as data filtering strategies when learning the GB1 fitness landscape.

1.1 Related Work

There are several factors that must be considered for machine learning-guided directed evolution including input sequence encoding, model architecture, training data selection strategy, and the number of variants to train and test on. [1] used a partial least squares (PLS) algorithm to improve volumetric activity of halohydrin dehalogenase. At each iteration, PLS is trained on a set of sequence-function pairs and based on predicted fitness, each variant is labeled as beneficial, neutral, or deleterious. In the following iteration, deleterious mutations are removed, neutral mutations are rescreened and beneficial mutations are randomly recombined to generate new training data. In [3], two rounds of ML-assisted directed evolution are performed on proteinase K sequences using 8 different machine learning models. For the first round of evolution, selected mutations for training are derived from previously known active variants and in the second round of evolution, mutations are selected for screening based on feedback from the machine learning model.

Gaussian processes are an efficient and effective method for learning protein fitness landscapes, capturing relevant pairwise sequence covariation and explicitly representing model uncertainty to locate and train on informative samples. [5] uses Gaussian processes to accurately infer the thermostability and binary functional status of cytochrome P450s. By constructing a multivariate normal distribution over known fitness values, the posterior for a given fitness value can be used to infer its mean and variance. This probabilistic representation of the fitness landscape can elucidate the model's uncertainty which is used to adaptively select training data. To keep model architectures consistent for comparison, this project does not use Gaussian processes. However, future work should compare the benefits afforded by selecting data using Gaussian processes, zero-shot predictions and active learning for protein fitness prediction.

In this project we use the machine learning-assisted directed evolution (MLDE) algorithm introduced in [12] and extended in [11]. MLDE learns to map GB1 sequences to fitness with minimal screening burden. MLDE performs one round of directed evolution where a sample of sequences are experimentally screened and used to train an ensemble of 22 machine learning models with varied architectures. The trained models are ranked according to validation error and the fitness predictions of the top performing models on the unlabeled variants are averaged. The maximum and mean true fitness of the 96 variants with the highest predicted fitness value indicate the extent to which higher fitness variants have been discovered. To navigate the hole-filled and epistatic fitness landscape, authors show that predicted $\Delta\Delta G$ of protein stability is an effective indicator of GB1 fitness and can be used to filter for fitness-enriched training data. By randomly sampling training data from varying amounts of variants with low $\Delta\Delta G$, they avoid training on zero fitness data while encouraging diversity in training data. This outperforms traditional directed evolution and is shown to be more beneficial than randomly selecting training data. Although $\Delta\Delta G$ is shown to be correlated with GB1 fitness and therefore a useful zero-shot predictor, there is no guarantee that $\Delta\Delta G$ would work for other proteins. Active learning is dismissed in this paper as "it adds an additional round of data collection to the workflow", but despite this burden active learning may assist in navigating the rugged landscape effectively without any prior protein information. To the best of my knowledge, a direct comparison between zero-shot predictions and active learning for protein fitness predictions remains to be seen.

2 Methods

2.1 Active Learning (AL) Overview

In pool-based AL, samples are iteratively selected from a larger pool of data to train the model. From a large set of unlabeled training data U, an initial batch of training data is selected randomly or using

Querying Strategy	Equation
Ensemble Upper Confidence Bound (E-UCB)	$h(v) = \mu_v + 2 * \sigma_v$
Ensemble Lower Confidence Bound (E-LCB)	$h(v) = \mu_v - 2 * \sigma_v$
Fitness Expectation (FE)	$h(v) = \mu_v$
Fitness Variance (FV)	$h(v) = \sigma_v$

Table 1: v is an unlabeled variant, μ_v is the average predicted fitness over all 22 ensemble models and σ_v is the standard deviation of the predicted fitness over all 22 ensemble models.

a heuristic and labeled for training. In the following iterations of AL, a querying strategy discovers informative data and selects these samples to train the model from scratch. The querying strategies use different approaches to quantify the usefulness of samples in U using feedback from the partially trained model. The number of iterations of AL depend on the task or data consumption constraints.

2.2 Querying Strategies

The appropriate data querying strategy depends on the machine learning task and model architecture. For image classification tasks using convolutional neural networks, the objective is to correctly classify all training images and generalize to test data. Hence, it is intuitive to train on data the classification model has trouble classifying correctly. To measure this uncertainty one may compare the least confidence or entropy of the softmax outputs of images in U [9]. Although these uncertainty-based methods for AL have been shown to be effective, outputs of deep learning models on unlabeled data do not indicate model uncertainty but may rather reflect uncertainty inherent in the data. Furthermore, our objective for protein fitness prediction is to avoid holes and discover high-fitness variants. Our AL querying strategies utilize uncertainty and fitness predictions across machine learning models. The metrics below are applied to all unlabeled variants at each iteration and the top variants are added to the training data for the next round of evolution.

Using the posterior probability from the Gaussian process, we can infer the mean and variance of the fitness for any sequence which can be used to recognize informative data for adaptive data selection. The GP-UCB and GP-LCB algorithms mentioned in [5] prioritize sequences with fitness values containing high mean and high variance and high mean and low variance respectively. With an ensemble of machine learning models, we no longer have a multivariate normal distribution over fitness values but can treat these models' predictions for a given sequence as samples from a fitness probability distribution. By computing the average and standard deviation of these samples, we can reason about the ensemble's efficacy and uncertainty to use the query by committee based strategies [4] mentioned in Table 1.

E-UCB balances the trade-off between exploration and exploitation by selecting sequences with large average predicted fitness and standard deviation (SD) over all models. E-LCB prioritizes sequences with large average predicted fitness and lower SD, sequences for whose predictions are stable yet high. In [5], GP-LCB was only used for the last iteration, but for our experiments we use it across all iterations. FV and FE utilize SD or average exclusively for querying. Since the original MLDE algorithm uses the predictions of trained models with lowest validation error, we create variants of the four previously mentioned querying strategies: E-TUCB, E-TLCB, TFV and TFE. These four strategies use the top 5 trained models for computing predicted fitness mean and SD.

2.3 Simulating Zero-Shot Predictions

To compare these AL querying strategies to zero-shot predictions using $\Delta\Delta G$, I attempted to calculate $\Delta\Delta G$ using the Triad Protabit software for all 149,361 screened variants but was unable to finish in time for submission. Hence to simulate how [11] randomly samples from varying amounts of variants with low $\Delta\Delta G$, I constructed two training sets, SIM0-1 and SIM0-2. For SIM0-1, I sampled 384 variants with fitness values between 0 and 3 and for SIM0-2, I sampled 384 variants with fitness values between 0 and 6. By preserving fitness diversity and preventing a selection of a majority of zero-fitness variants, these two data sets excel to varying degrees at capturing the properties of the training data enabled by using $\Delta\Delta G$. Zero-fitness variants are not absent due to random sampling for data diversity and the potential error in using $\Delta\Delta G$. The distributions of fitness values for SIM0-1

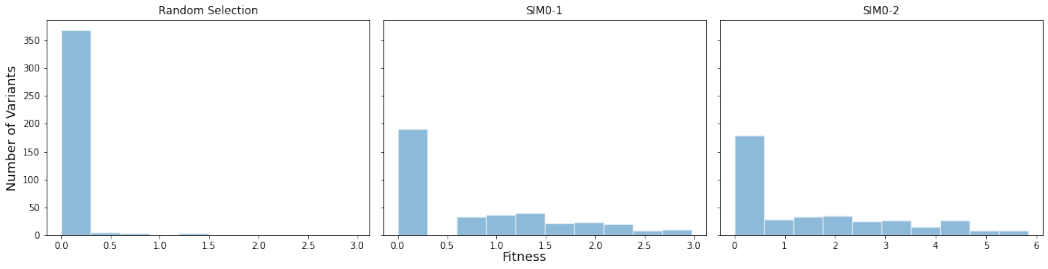

Figure 1: Fitness distributions for random data selection and simulated zero-shot prediction

and SIM0-2 are shown in Figure 1, along with a random selection of 384 variants from all screened variants.

2.4 AL Procedure for Learning GB1 Fitness Landscape

1. Select initial training data pool of 96 sequences using random selection or a relevant heuristic
2. Train all models in ensemble from scratch using cross validation on current training data pool
3. Evaluate all models in ensemble on all possible variants for which we have experimental fitness values (149,361 variants) and use a querying strategy to add new data to the training data pool
4. Repeat from step 2 for a total of four iterations

To compare my experimental results with those of [11], I select 96 labeled variants four each of the 4 AL iterations providing a total 384 sequences, equivalent to the training set size shown to produce the best performance in [11]. Although the original paper optimally selects the sequence encoding method and model hyperparameters, I use the georgiev encoding and do not perform any hyperparameter optimization in all AL experiments. Since the performance of the ensemble is dependent on the models' weight initialization and the initial data pool, 20 independent runs of AL are performed. For simulated zero-shot prediction experiments, 96 sequence-fitness pairs are randomly selected from either SIM0-1 or SIM0-2 and added to the training pool in each of the 4 iterations. Since active learning is often sensitive to the initial training pool and avoiding low-fitness regimes is important, we run experiments combining simulated zero-shot predictions with active learning. The initial training pool contains 96 variants sampled from SIM0-1 and the following three iterations use active learning. Upon training the ensemble on a data pool, the normalized mean and max true fitness of the 96 test variants with the highest average predicted fitness across the top 3 trained models are recorded.

3 Results

As shown in Figure 2, other than FE, querying strategies that utilize all models in the ensemble perform similarly to random sampling and do not perform better than sampling from SIM0-1. However, the same querying strategies utilizing the top 5 models ranked by validation error are better than SIM0-1. Just as [11] computes fitness predictions by averaging predictions across the top 3 models, we find that considering the top models for active learning is valuable. Despite all querying strategies being understandably worse than SIM0-1 for the first iteration, most show improvements over SIM0-1 by just the second iteration. Finally, deriving the initial training pool from SIM0-1 boosts performance of all querying strategies and closes the gap with the fitness-enriched SIM0-2 sequences considerably. Random sampling after zero-shot initialization performs comparably to sampling from SIM0-1 throughout training, indicating the benefits of training with a small sample of relatively diverse and high fitness data. It is worth noting that mean fitness is not always monotonically increasing across training iterations, suggesting that the addition of new training data is not always beneficial. However, by recording results per iteration, we can stop early at the best performing iteration. [11] showed that ensembles trained on small training sets with 24 and 48 sequence-function pairs is not as effective as training on 384 sequences. Though more training data is advantageous,

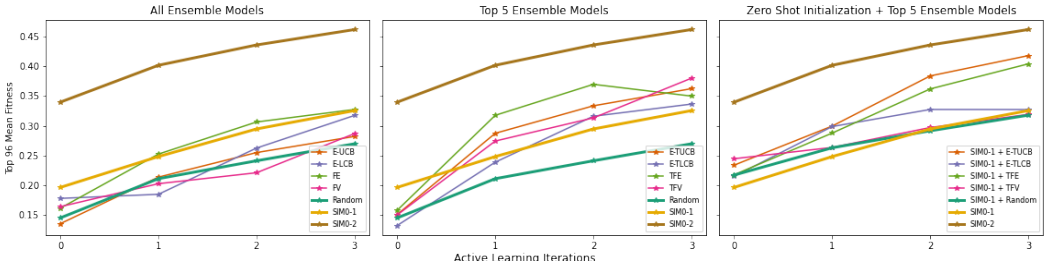

Figure 2: Average mean fitness of the top 96 predicted variants over 20 runs are shown for AL querying strategies using all models in ensemble (left), top 5 models in ensemble (middle) and combined with zero-shot predictions for initial data pool (right). Results are also shown for selecting data incrementally from SIM-0, SIM-1 and randomly from all possible variants.

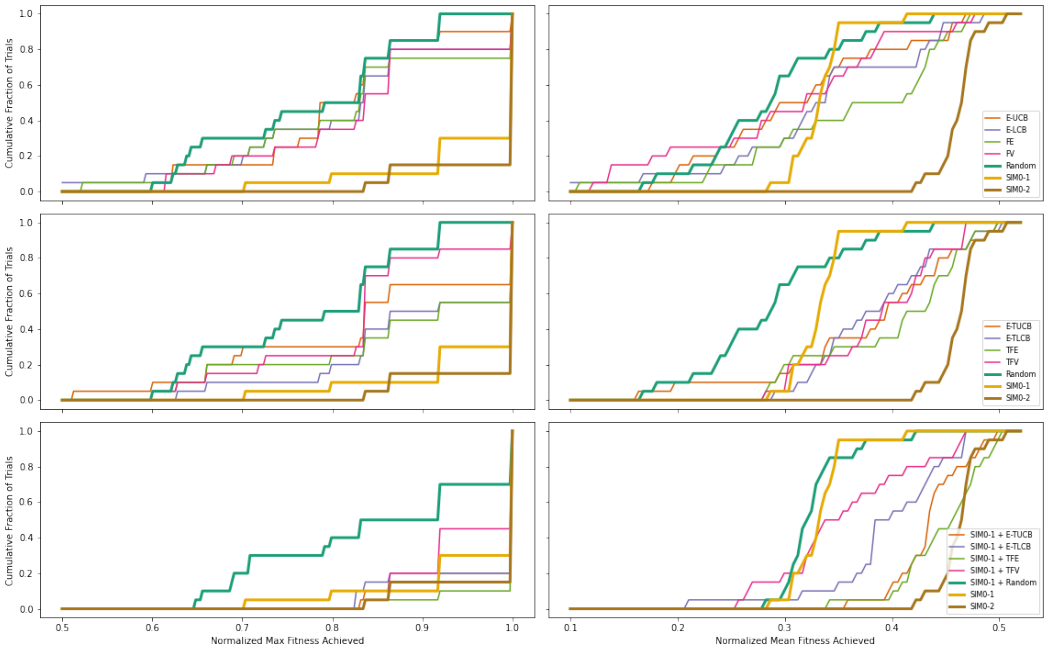

Figure 3: Empirical cumulative distribution functions (ECDFs) which for a given max (left) or mean (right) fitness value, specify the fraction of runs that reach a max or mean fitness equal to or less than that value. Results are shown for AL querying strategies using all models in ensemble (top), top 5 models in ensemble (middle) and combined with zero-shot predictions for initial data pool (bottom).

we can rely on models trained on less data to effectively guide data selection and improve ensemble performance.

We present ECDFs of the achieved max and mean fitness in Figure 3 for 20 runs. When observing max fitness achieved, SIM0-1 and SIM0-2 perform similarly but the same cannot be said for mean fitness. Hence, both metrics must be observed to accurately compare data selection methods. For the three training paradigms, it is not always clear when certain querying strategies outperform others, but generally FE performs most effectively followed by UCB and LCB strategies. Along with the mediocre performance of FV, this indicates that uncertainty across models is not as effective as using expected fitness for data selection. Uncertainty may be ineffective because of the type of models used in the ensemble or the weight assigned to the standard deviation in the querying strategies. Using TFE with zero-shot initialization is more effective than iteratively sampling from SIM0-2 according to both metrics, encouraging further exploration of combining zero-shot predictions and active learning.

To visualize the quality of the selected data, we plot the true fitness distributions of our data pools at each training iteration. Random sampling obtains few high-fitness variants across the four iterations,

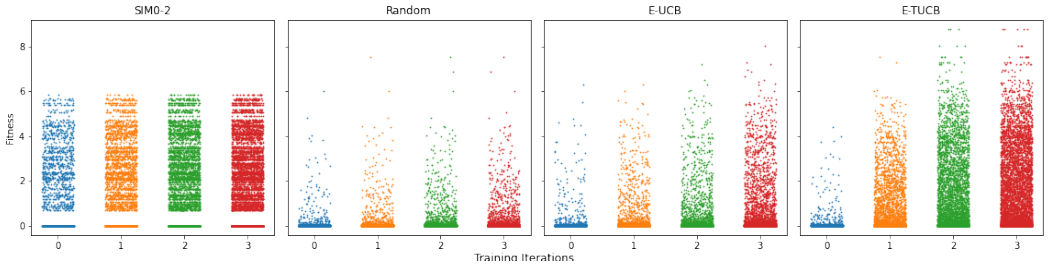

Figure 4: Distributions of fitness values for training pools across four training iterations for all 20 runs. Data is iteratively queried from SIM0-2, randomly, using E-UCB, or using E-TUCB.

Figure 5: Distributions of fitness values for training pools across four training iterations for all 20 runs when the initial data pool is randomly sampled from SIM0-1. Subsequent iterations sample from SIM0-1, randomly, using TFE, or using E-TUCB.

but E-UCB and E-TUCB obtain more fitness-enriched variants. E-TUCB's success at selecting high-fitness sequences improves the ensemble's training and ability to discover high-fitness variants among the held-out sequences. Despite only containing variants with fitness values between 0 and 6, training the ensemble on SIM0-2 is highly effective. This may be due to properties of the selected variants besides fitness and corroborates the finding from [11] that the highest fitness variants are not necessary for training effective models.

In Figure 5, we show the training pools over time when we initialize the first data pool by randomly sampling from SIM0-1. Random sampling in subsequent iterations discovers a few variants with fitness above 3, but TFE and E-TUCB perform substantially better. Compared to Figure 4, combining active learning with simulated zero-shot predictions is advantageous for quickly discovering fitness-enriched sequences.

4 Conclusion

Active learning is effective for inferring the GB1 fitness landscape with minimal screening burden. Querying strategies which utilize a diverse and accurate set of machine learning models are as effective, if not better than sampling from SIM0-1. Though sampling from SIM0-2, which contains variants with higher fitness values compared to SIM0-1, is most effective, combining active learning with an initial data pool selected from SIM0-1 is shown to be similarly effective. Hence, if zero-shot predictions are unavailable, limited or noisy for a given protein, active learning is a protein-agnostic approach for data selection that can discover fitness-enriched variants in the landscape. However, choosing the optimal querying strategy may be difficult as it could be protein-dependent and must correctly manage the exploration-exploitation trade-off during data selection. Furthermore, as seen in the results, it is difficult to isolate the impact of the querying strategy as its success may depend on the initial training pool, training data quality, or the number and types of machine learning models used. Although active learning strategies require in-silico screening of all variants per AL iteration, performing the forward pass for the top models on all variants should not be too computationally expensive.

5 Discussion

A limitation to this project was that simulated zero-shot predictions were used, as opposed to $\Delta\Delta G$ Triad calculations, which should be used in future work. This project also only compared active learning with zero-shot predictions for the GB1 protein. To understand the general effectiveness of active learning for navigating epistatic fitness landscapes, the two data selection paradigms must be evaluated on different proteins and different zero-shot prediction heuristics. Also, as we develop a better understanding of different query by committee strategies, we can use different querying strategies per iteration as was shown to be effective in [5].

The diversity of ensemble models is important for effective data selection ([4]), so future work should explore how to quantify and enforce decorrelated models that are accurate. Although our querying strategies emphasize selecting high-fitness variants, diverse sequences should also be prioritized to better learn the landscape's topology which can be accounted for when designing new querying strategies. Finally, since each model is trained using cross validation, we have access to multiple trained versions of each model. New querying strategies should be developed that incorporate individual model uncertainty captured by variation in performance over different held-out data during training.

References

[1] Richard Fox, Christopher Davis, Emily C Mundorff, Lisa M Newman, Vesna Gavrilovic, Steven K Ma, Loleta M Chang, Sarena Tam, Sheela Muley, John Grate, John Gruber, John C Whitman, Roger A Sheldon, and Gjalt W Huisman. Improving catalytic function by prosar-driven enzyme evolution. *Nature Biotechnology*, 25:338–344, 2 2007.

[2] Jonathan Greenhalgh, Apoorv Saraogee, and Philip A. Romero. Data-driven protein engineering. *Protein Engineering: Tools and Applications*, 2020.

[3] Jun Liao, Manfred K. Warmuth, Sridhar Govindarajan, Jon E. Ness, Rebecca P. Wang, Claes Gustafsson, and Jeremy Minshull. Engineering proteinase k using machine learning and synthetic genes. *BMC Biotechnology*, 7, 3 2007.

[4] Prem Melville and Raymond J. Mooney. Diverse ensembles for active learning. In *Proceedings of the Twenty-First International Conference on Machine Learning*, ICML '04, page 74, New York, NY, USA, 2004. Association for Computing Machinery.

[5] Philip A. Romero, Andreas Krause, and Frances H. Arnold. Navigating the protein fitness landscape with gaussian processes. *Proceedings of the National Academy of Sciences*, 110(3):E193–E201, 2013.

[6] Yutaka Saito, Misaki Oikawa, Hikaru Nakazawa, Teppei Niide, Tomoshi Kameda, Koji Tsuda, and Mitsuo Umetsu. Machine-learning-guided mutagenesis for directed evolution of fluorescent proteins. *ACS Synthetic Biology*, 7(9):2014–2022, 2018. PMID: 30103599.

[7] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[8] H. S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, page 287–294, New York, NY, USA, 1992. Association for Computing Machinery.

[9] Keze Wang, Dongyu Zhang, Ya Li, Ruimao Zhang, and Liang Lin. Cost-effective active learning for deep image classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(12):2591–2600, Dec 2017.

[10] Bruce J. Wittmann, Kadina E. Johnston, Zachary Wu, and Frances H. Arnold. Advances in machine learning for directed evolution. *Current Opinion in Structural Biology*, 69:11–18, 2021.

[11] Bruce J. Wittmann, Yisong Yue, and Frances H. Arnold. Machine learning-assisted directed evolution navigates a combinatorial epistatic fitness landscape with minimal screening burden. *bioRxiv*, 2020.

[12] Zachary Wu, S. B. Jennifer Kan, Russell D. Lewis, Bruce J. Wittmann, and Frances H. Arnold. Machine learning-assisted directed protein evolution with combinatorial libraries. *Proceedings of the National Academy of Sciences*, 116(18):8852–8858, 2019.

[13] Kevin K. Yang, Zachary Wu, and Frances H. Arnold. Machine-learning-guided directed evolution for protein engineering. *Nature Methods*, 16:687–694, 7 2019.