# *In vitro* continuous protein evolution empowered by machine learning and automation

Tianhao Yu[1,2,3], Aashutosh Girish Boob[1,2,4], Nilmani Singh[4], Yufeng Su[3,5], and Huimin Zhao[1,2,3,4*]

[1]Department of Chemical and Biomolecular Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, [2]Carl R. Woese Institute for Genomic Biology, [3]NSF Molecule Maker Lab Institute, [4]DOE Center for Advanced Bioenergy and Bioproducts Innovation, University of Illinois at Urbana-Champaign, Urbana, IL 61801, [5]Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL 61801

*To whom correspondence should be addressed. Phone: (217) 333-2631. Fax: (217) 333-5052. E-mail: zhao5@illinois.edu

## Summary

Directed evolution has become one of the most successful and powerful tools for protein engineering. However, the efforts required for designing, constructing, and screening a large library of variants can be laborious, time-consuming, and costly. With the recent advent of machine learning (ML) in the directed evolution of proteins, researchers can now evaluate variants in silico and guide a more efficient directed evolution campaign. Furthermore, recent advancements in laboratory automation have enabled the rapid execution of long, complex experiments for high-throughput data acquisition in both industrial and academic settings, thus providing the means to collect a large quantity of data required to develop ML models for protein engineering. In this perspective, we propose a closed-loop *in vitro* continuous protein evolution framework leveraging the best of both worlds - ML and automation and provide a brief overview of the recent developments in the field.

## Introduction

The essence of directed protein evolution lies in a two-step process designed to expedite natural evolution by iteratively creating a diverse variant library and then screening/selecting from the library the mutants which show improved phenotypes.[1] Step one aims to cover a broad spectrum of possible variant space by gene diversification tools such as random mutagenesis, focused mutagenesis, and recombination. Step two aims to discover mutations resulting in improved properties by experimentally quantifying or qualifying variants obtained from step one using high-throughput screening/selection methods. Although directed evolution has enabled many successful protein engineering studies,[2,3] the process involves laborious experiments and is often limited by the throughput of screening and selection methods.[1,4,5] Thus, many studies focused on developing *in vivo* continuous evolution[6] or increasing the throughput of screening/selection methods.[7,8] However, many of the new technologies are not versatile as they are only suitable for certain properties, and the success of one evolution campaign can be difficult to be transferred to another target protein.[1]

To address the need for efficient and versatile protein evolution processes, ML has been increasingly applied to assist protein engineering studies.[4,9,10] ML models can be broadly categorized into two groups: supervised ML and unsupervised ML. Supervised ML models are trained using examples of input-label pairs and tasked to predict the label of unseen inputs. Unsupervised ML models, on the other hand, are trained without labels and tasked to generalize and extract meaningful representations or patterns from the input. Targeting different aspects of directed protein engineering, many studies have successfully applied ML to optimize different steps of the engineering process. For example, unsupervised ML models have been applied to predict variant fitness using only sequence information enabling an informative and high-quality design of variants.[11-14] Supervised ML models can infer the properties of unseen variants based on a small library of screened variants, which can replace the necessity of screening a large number of variants.[15-17] These models can be integrated into one unified framework to further improve the efficiency of protein engineering and minimize human intervention. With algorithms intelligently navigating through the variant landscape and highly customizable automation robotics performing experiments, we envision a future of automated closed-loop protein evolution, i.e., *in vitro* continuous protein evolution (Figure 1).

The closed-loop process can be initiated with the design of an informed variant library by zero-shot ML models predicting variational effects. The proposed library will be constructed, expressed, and screened by automation robotics. After obtaining the experimental data, a supervised ML model can be trained to map variants to fitness. Subsequently, optimization algorithms can explore the variant landscape and make rational decisions to recommend the variants to be tested in the next round. Analogous to the active learning strategy,[18] such closed-loop process can be carried out iteratively until optimal variants are obtained. In this perspective, we do not intend to discuss the applications and methods of ML-assisted protein engineering, as they are covered comprehensively elsewhere.[4,5,9,19-21] Instead, the primary focus of this perspective is to summarize the recent developments in ML models and automation technologies that can be integrated to achieve our proposed framework of *in vitro* continuous protein evolution. This framework will have ML algorithms as a decision maker and robotics as an experiment executioner, which delivers a highly flexible and versatile protein engineering platform with minimal human intervention.

## Design an initial variant library using zero-shot ML models

Directed evolution utilizes various gene diversification tools like random mutagenesis or gene recombination to explore the protein variant landscape to identify the variants with the desired properties. However, randomly designed libraries by uniformly sampling from the variant landscape will be dominated by zero- or low-fitness variants, especially when the modified residue is in the region contributing to protein activity.[22-24] The enormous size of the variant landscape combined with the rare occurrence of beneficial variants makes directed evolution a challenging and inefficient task, which necessitates high-throughput screening and selection methods. Although the throughput of screening is no longer the top limiting factor under the context of machine learning-guided directed evolution (MLDE), a randomly designed variant library will still compromise the training efficacy of the ML model. Firstly, the variants with zero or close to zero fitness contain little information, which hinders the ML model to learn the meaningful topology of the variant landscape. Secondly, the domination of low fitness variants among the training data will introduce a bias toward the low fitness regime and cause the ML model to be less accurate in the high fitness regime where improved variants are located.[24] Therefore, the design of a high-quality and informative initial library is crucial for the success of initiating a closed-loop MLDE. One effective strategy to minimize the occurrences of low fitness variants is to design the initial library with “zero-shot” variational effect prediction models. The name “zero-shot” implies such prediction tools do not require any prior knowledge other than the wild-type sequence of the protein and the homologous proteins.

Sequence-based zero-shot models typically make predictions based on two different types of information: local and global evolutionary context (Figure 2a). Local evolutionary models, such as EVmutation[25] and DeepSequence[12], take advantage of the multiple sequence alignment (MSA) data of the target protein. They are essentially probabilistic models that seek to capture residue dependencies by using the statistical patterns observed in the MSA. They can make quantified zero-shot predictions to mutational effects by comparing the likelihood of certain variants with wild-type. Both EVmutation and DeepSequence have demonstrated that the predicted mutational effects can have a significant correlation with the phenotype obtained from various deep mutational scanning experiments.[12,16,25] Using a collection of 42 high-throughput mutational scanning experiments as a comparison, DeepSequence outperformed EVmutation by achieving a higher average rank correlation. However, depending on the quality of the MSA data, EVmutation could have superior performance than DeepSequence as shown by a simulated MLDE study using protein G domain B1.[24] Generally, local evolutionary context models are dependent on the quality of the MSA data. Low-quality MSA with deficient diversity makes DeepSequence more prone to failure compared with EVmutation.

On the other hand, global context models leverage large sequence databases containing billions of protein sequences not limited to the homology of the target protein. The majority of global models are language models, such as evolutionary scale modeling (ESM)[11,26] and ProtTrans[27]. They are trained on raw amino acid sequences and are tasked to translate protein sequences into semantic-rich representations. During training, amino acids are kept hidden (masked) from the model and the model’s task is to retrieve or fill in the hidden residues based on the rest of the unmasked residues. Through such a training process, the model is encouraged to learn the semantics of a protein sequence. When used as zero-shot predictors, the trained language models can be used to calculate the possibilities of a masked residue being a certain amino acid giving all other unmasked residues. Therefore, by iteratively masking all positions and interrogating the language model to calculate the possibilities of all possible amino acids, researchers can obtain the predicted fitness of all single variants. Notably, global models are not dependent on MSA data, and will not be limited by the low-quality MSA. Furthermore, the local and global models can be combined as demonstrated by the MSA transformer.[28] Instead of training on protein sequences, the MSA transformer was trained on protein MSAs. Compared with local models, the MSA transformer can make up for the lack of informative MSAs. Most importantly, the study has shown that large scale language models trained on sequence information alone can effectively capture the functional effect of sequence variation across protein families without any supervision from experimental data.[11]

The initial conceptualization of designing an informative variant library was reported by Wittmann et al. using a simulated MLDE workflow.[24] The study compared the outcomes of simulated MLDE using a library designed with or without zero-shot prediction models. They concluded that using informed library design, the simulated MLDE can obtain optimal variants much more frequently than without the use of informed library design, regardless of the choice of zero-shot models. Although the success of introducing zero-shot prediction models was demonstrated *in silico*, few experimental studies implemented such a strategy, especially for protein engineering studies with more complex phenotypes such as enzyme reactivity and selectivity.

With the recent advancement of generative models, zero-shot design can also be achieved by using generative models such as ProGen[29,30], MSA VAE/AR-VAE[31], and ProteinGAN[32]. However, the zero-shot models discussed earlier in this section are aimed to predict variant effects with a given wild-type, while the generative models are best applied to sample diverse and novel sequences and are covered in more detail in section 4. It is also worth mentioning that besides using ML, the community has demonstrated many successful directed evolution champaigns enabled by the design of smart libraries using structural bioinformatics tools.[33-36]

### Develop ML models predicting variant properties

After the variant library has been constructed, expressed, and quantified, the mutation to property pairs can be obtained. They can be used as training data for a supervised ML model, such as a simple linear regression or more complicated neural networks tasked with predicting the property of unseen variants. Therefore, the laborious and iterative high-throughput screening part of the traditional directed evolution can be replaced with the prediction of the variant landscape *in silico* hence requiring only a small fraction of the variant space needed for experimental testing. Several independent studies have also demonstrated that ML could generalize low-order variant information to higher-order variants, which implies that ML models trained on single point mutations can also capture multiple-point mutations information.[10,15-17,24] Various successful studies have been conducted, showcasing the value of ML-assisted protein engineering, such as the engineering of beta-lactamase[17], polyethylene terephthalate hydrolase[37], and more[4,5,9,19,38].

The general framework of building an ML model for predicting variant fitness is to first represent protein sequence in a numerical form and to use the data obtained from experimental assay to train a regression model. The major consideration in designing such an ML model is the choice of protein embedding methods, which refers to the process of encoding amino acid sequences into numerical representations. The simplest embedding method is one-hot encoding where only the amino acid sequence information is preserved. In the past three years, unsupervised ML models and language models trained on large protein sequence database have been developed.[13,26,27,39-41] These methods provide much richer and denser information. Using such representation is also referred to as "semi-supervised" learning as the global representation is obtained by trained ML models in an unsupervised manner and a downstream regression model is trained in a supervised manner. Although trained on sequences from all protein families, some of these models can be further combined with the MSA data of the target protein to make the representation more task-specific (Figure 2b). For example, ESM[26] and UniRep[39] are fine-tuned using the homologs of the target proteins. Besides fine-tuning, the global representation models can also be concatenated with local features like the direct coupling model as demonstrated by ECNet.[17] With representations carrying meaningful features, even a simple linear regression model or shallow neural networks can capture the variant properties accurately.[15,16,24,42] Even though many global representation models use transformer architecture[43], several disadvantages can be observed. Specifically, the computational expense is high, and the attention-based transformer has a limit on the length of the proteins. Recently, Yang et al. reported a pre-trained model using convolutional architecture, which performed comparatively with state-of-the-art amino acid sequence language models using transformers and scales better for modeling long protein sequences.[41]

Although ML is known to be data hungry where accurate prediction requires a considerably large dataset, recent studies have been focusing on "low-*N*" scenarios where the number of training examples is limited (e.g., less than 100). Under the ideal low-*N* setup, researchers do not need to screen a large number of variants extensively. Therefore, low-*N* ML models can fundamentally avoid the need for high-throughput screening and selection methods and aid in making such a workflow universally applicable to all protein engineering tasks. The pioneering work on data-efficient protein engineering was introduced by Biswas et al.[15] The authors presented a low-*N* *in silico* directed evolution workflow consisting of five steps: 1) start with an unsupervised global representation model, UniRep; 2) fine-tune the unsupervised model using sequences homologous to target protein; 3) train a simple regression model using experimentally determined properties of less than one hundred variants; 4) perform *in silico* directed evolution, which is covered in more details in the section below; and 5) experimentally validate the predicted candidates. Following such a paradigm, they successfully located variants of the green fluorescent protein and TEM-1 β-lactamase with a several-fold increase in fluorescence intensity and ampicillin resistance, respectively. Although the authors found that the pre-trained embeddings are helpful under a low-*N* situation, recent systematic benchmarking studies found that simple strategies which do not use pre-training are competitive or better in most scenarios.[16,24,44]

Hsu et al. performed a systematic and comprehensive comparison of more than ten different ML models for variant property prediction using more than twenty different protein datasets under a low-*N* scenario.[16] Besides the existing ML models, the authors also introduced a simple, yet elegant model termed the augmented approach. It involves training a linear regression model using a combination of two features: the first feature is the evolutionary density feature such as the inferred variant's likelihood from evolutionary context models and the second is the one-hot encoding. All models included in the study can be categorized into three groups: probabilistic models trained by MSA data such as EVmutation and DeepSequence, global context models such as ESM and UniRep, and models trained by both evolutionary context and assay-labeled data such as the augmented approaches and TLmutation[45]. The study is carried out in a low-*N* scenario by using a training dataset of size ranging from 48 to 240 variants. It was concluded that the augmented version of DeepSequence achieved the best average performance at all training dataset sizes. The authors also found that all the augmented approaches outperformed the non-augmented counterparts regardless of the size of the training dataset. This study is consistent with the study by Wittmann et al., that in the low-*N* regime, small models perform competitively or better than large-scale global models.[24]

A wide variety of ML models have been developed for predicting variant properties over recent years. Many of these models are analogous to the development trajectory of the natural language processing (NLP) field in the computer science community such as Doc2Vec[46], mLSTM (multiplicative long-short term memory)[39], and transformer[26]. However, Hsu et al. showed that mainstream NLP models (using supervised data to fine-tune global context models) did not perform as well as ML models without directed NLP analog (augmented approaches).[16] Others also demonstrated non-traditional NLP models are comparative with standard mainstream ML models[41,47]. As a result, it is of importance to further develop biology-specific solutions for protein modeling and to use mainstream NLP models with caution when developing sequence-based models. On the other hand, with the recent success of protein structure prediction tools such as AlphaFold[48,49], RoseTTAFold[50], and ESMFold[51], structural information is more accessible than ever before. New variant property predictors leveraging structural information could push the current state-of-the-art to a new level, enabling even more accurate low-*N* or zero-shot predictions.

**Navigate through variant spaces**

The protein length ranges from tens to thousands of amino acids. Given that there are 20 amino acids, it is impossible to explore the entire protein landscape even with the availability of high-quality variant fitness ML models. Therefore, it is important to prioritize variants that have a higher likelihood to be viable when screened for fitness and activity. Two commonly used directed evolution approaches are site saturation mutagenesis which inspects all 20 amino acids at a few promising sites primarily selected based on literature or structural modeling and deep mutational scanning.[1] However, such methods can be laborious, result in a sparsely functional library, and are often limited by the lack of high-throughput screening and selection methods. In the past few years, researchers have exploited new MLDE approaches to cover a broader landscape and aid in the selection of functional higher-order (multi-mutation) variants. These approaches include the use of various deep generative models that can design artificial counterparts to improve the efficiency of MLDE(Figure 2c).[21]

The protein sequence is like a sentence in a foreign language and only when amino acids are put in a certain manner, it makes a viable variant. Therefore, language models from NLP literature are increasingly applied to capture complex dependencies between amino acids and learn the context of the amino acids in the protein sequence. Masked language models are particularly of importance for MLDE as their objective is to predict the probability of amino acid occurrence given all other amino acids in a protein sequence. Therefore, one can compute the likelihoods of all single-residue substitutions and suggest ‘evolutionary’ beneficial mutations in the local landscape. For directed evolution, these can be tested in a laboratory setting and the ones with the improved fitness can be iteratively fed to the trained language model to combine beneficial mutations. This approach is recently demonstrated by evolving human antibodies using an ensemble of ESM1b and ESM1v language models.[52] The authors performed two rounds of language model-guided evolution on 7 antibodies and reported 7-fold and 160-fold improvement in binding affinities on highly evolved and unmature antibodies, respectively. As the language models are general, i.e., trained on large datasets of natural protein sequences, the authors also showed that the recommended predictions can guide directed evolution of proteins belonging to diverse families.

Instead of screening variants in the laboratory after every round, one can also utilize a fitness predictor for developing an *in silico* directed evolution platform. Biswas et al. integrated the eUniRep model with a Markov chain Monte Carlo-based sequence proposal to computationally explore landscapes for the green fluorescent protein and TEM-1 β-lactamase at a scale of $10^7$–$10^8$ variants.[15] Starting with an initial sequence, random mutations are incorporated and accepted if the fitness of the new sequence is greater than the starting sequence. Schissel et al. also demonstrated the ability of *in silico* directed evolution by designing nuclear-targeting abiotic miniproteins to deliver various cargos with high efficiencies.[53] Apart from a predictor, the platform incorporated a generator trained using a nested LSTM architecture and an optimizer to perform evolution towards the most optimal sequence. A similar closed-loop approach has been demonstrated for designing highly potent antimicrobial peptides.[54]

While most of the MLDE studies are carried out by introducing mutations to the wild-type sequence, incorporating functional proteins with high sequence diversity can provide valuable insights about the global landscape. Generative models specialize in this very task as they learn the underlying high-dimensional distribution from the training data and can aid in sampling novel sequences far away from the wild-type. A variety of deep learning architectures such as variational autoencoders[31,55], generative adversarial networks[32], and autoregressive language models[29,30,56], have been developed to engineer artificial proteins. Of particular interest are conditional generative models capable of designing proteins with desired functional properties and novel attributes. Compared with generative models trained only on a large dataset of homologous protein sequences belonging to a particular family, universal models can be developed where text or sequence labels can be used to control sequence generation.[29] This can be immensely helpful in engineering rare enzymes (with very low sequence identity) as one can train a global model to learn the protein representation and condition (or fine-tune) the model based on the substrate, protein family, or organism to generate desired functional homologs. Furthermore, novel enzymes can also be sampled to enhance the substrate scope as demonstrated by the work on designer recombinases.[57] Sequence-to-sequence or neural machine translation models can also be employed for designing proteins or peptides of interest[58]  as the objective can be viewed as finding the target sequence that maximizes the conditional probability given the input.

While generative models are well established to design realistic proteins, the selection of variants with improved activity largely relies on scores obtained from the fitness prediction models. However, the predictor can be integrated directly with the discriminator to steer the output from the generator to be sequences possessing higher fitness.[59,60] Biasing in sequence generation can also be achieved using transfer learning as demonstrated by Antibody-GAN.[61] Chan et al. reported attribute extrapolation beyond the training dataset using a generative framework called GENhance.[62] The model utilizes an encoder-decoder framework where the encoder learns meaningful representation in the latent vector while the decoder reconstructs the original input from the latent vector in an autoregressive manner. To capture the information regarding the target attribute, the encoder is trained with a pairwise contrastive loss on a subset of dimensions from the latent vector. This results in disentanglement and storage of information primarily concerning the fitness in a few dimensions which can be perturbed during sampling to generate sequences with improved properties (in this case, protein stability). Furthermore, the model consists of smoothing and cycle-consistency learning objectives to generate feasible output when the perturbed latent vector is fed to the decoder and generate sequences that can be accurately ranked by the predictor, respectively. Combining all the objectives, the model outperforms the baseline ML models including adaptive sampling[59] in generating high-quality texts or proteins (low perplexity) with a high bias towards the sampling of strong-positive movie reviews and highly stable Angiotensin-Converting Enzyme 2 (ACE2) protein variants, respectively.

Overall, the incorporation of these deep learning models integrated with a fitness predictor will aid in exploiting and exploring the fitness landscape in the MLDE efficiently. In the future, integration of generative models to provide multiple starting points for language model-guided *in silico* directed evolution can further make evolution faster, cheaper, and easier. We refer readers to recent in-depth reviews covering language models[63] and deep generative models[21].

**Prioritize designs for subsequent experimental measurements**

Prioritizing designs for subsequent experimental measurements is an essential step in our proposed *in vitro* continuous protein evolution framework, which is achieved by designing an acquisition function to rank all variants to determine a subset that keeps evolving in the subsequent rounds of model training. The naïve approach is to preserve the variant with the highest predicted fitness in each round.[64] This straightforward greedy algorithm highly relies on the quality of the top prediction. Moreover, this will result in the selected variants being highly similar and thus make the variants list not diverse enough. Limited coverage of the protein landscape reduces the probability of finding high-quality variants. Beam search, a common strategy in text generation[65] was thus introduced for this task[14,66,67]. In this strategy, the researchers kept a predetermined number of best partial solutions. The number could be customized to achieve a balance between *in silico* running time and the quality of generated variants. To further improve the diversity of selected variants on each round, the batched-acquisition function was designed.[23] The batched-acquisition function includes an extra mutual information term encouraging the model to select diverse mutations. However, due to the cost of calculating the mutual information, the exact solution according to the batched-acquisition function is hard to calculate. Thus, an alternative approach is usually used by using the sampling method to get an estimated solution.

Besides focusing on a few variants with the highest predicted fitness, it is also necessary to enlarge the search regions on the protein landscape. To explore the candidates which might be underestimated by computational models, some methods introduced randomness in the process.[68] For example, Biswas et al. performed the Metropolis-Hastings Markov chain Monte Carlo algorithm to stochastically sample from all variants.[15] The probability of each variant is calculated according to the predicted fitness. Other approaches demonstrated that rather than random exploration, the region with higher uncertainty should be prioritized when exploring.[22,23,69] Especially when the selected variants are experimentally evaluated after each round and computational models are updated according to the new ground truth. In this active learning-like setup, experimentally evaluating the sites with the higher uncertainty could significantly increase the high confidence region of the computational model and thus improve their performance. Following this idea, it is natural to use Bayesian optimization in protein engineering. A common choice of the acquisition function is the Upper Confidence Bound (UCB) which includes both predicted fitness and uncertainty as the evaluation criterion.[70] The weights of both factors are set as hyperparameters, which can be used to guide the trade-off between exploitation and exploration. Other acquisition functions like Lower Confidence Bound (LCB) and Expected Improvement (EI) are also widely used to select "next-best" candidates.[71,72]

The Gaussian process is a popular choice of the surrogate model of Bayesian optimization as it can provide both predicted fitness and uncertainty and works well with limited data.[10,23,73,74] Gaussian process is defined by a mean function and a covariance function. In protein engineering applications, the mean function is expected to be the fitness of the variant and the covariance function is estimated by the kernel function of embedding vectors of two variants. The embedding vector could be very flexible including pre-trained language embedding, one-hot sequence embedding, and expert knowledge features vector.[22,69,74] After fitting the model with experimental data, the mean function and marginal variance are used as the predicted fitness and the uncertainty, respectively (Figure 2d). As an example, by using UCB to navigate the fitness landscape of acyl-ACP reductase, Greenhalgh et al. successfully identified a variant that increases the fatty alcohols production by more than two-fold as compared with the starting sequence.[75]

Deep learning models continue to be successfully applied to an increasingly diverse range of biology problems. The structural biology community has shown its huge advantages in understanding complex biological mechanisms.[48,50] So theoretically, the combination of a deep learning model and Bayesian optimization could be a powerful strategy for protein engineering. However, it is tricky to get a well-estimated uncertainty for ML models.[76] Among several attempts, the widely used strategy to define the uncertainty is to use the variance of prediction of models with different hyperparameters, different neural network structures, or even different kinds of ML models.[22,77] This strategy can work well, although it does not strictly follow the mathematical definition of variance of the model. Exploring a mathematically correct definition of the variance of ML models could be a challenging but interesting future research direction.[10]

### Develop an automation platform enabling library creation, protein expression, and assay screening

The wet-lab experiments for directed protein evolution are typically accomplished in a low-throughput fashion by skilled researchers, which is often tedious and labor-intensive. However, biofoundries provide an ideal integrated environment with access to hardware and software for high-throughput data generation, acquisition, and analysis.[78,79] An integrated biofoundry combines high-throughput core instruments such as liquid handlers, thermocyclers, fragment analyzer, high content screening instruments with peripherals such as plate sealers, shakers, and incubators using a robotic arm and scheduling software. Setting up new biofoundries and automation platforms is expensive, and benefits may not be clearly laid out to incentivize researchers to dedicate resources towards this. Additionally, biofoundries require skilled personnel and long-term service contracts for instrument maintenance that further adds to the hurdle. New automation platforms require careful consideration of ongoing projects, design needs for various assays and future scalability of the system. While automation platforms are not routine in academia, they are widely used in industry for multitude of biological assays providing superior speed, accuracy, and reproducibility. Further, automation instrumentation and processes can be shared among projects and laboratories to reduce costs. Collaboration with established biofoundries across the globe will further accelerate the development of automated MLDE workflows.[79] We envision that the combination of ML and an integrated biofoundry offers an ideal platform for creating an *in vitro* continuous protein evolution workflow.

Laboratory automation can lead to increased throughput, streamlined and reproducible workflows, reduced turnover time, and significant labor savings.[80,81] For example, Edinburgh Genome Foundry and Illinois Biological Foundry for Advanced Biomanufacturing (iBioFAB) have developed automated workflows that can perform thousands of DNA assemblies per week.[79,82] Recently, an end-to-end pipeline called PlasmidMaker for automated high-throughput plasmid design and construction was developed and implemented on the iBioFAB.[83] In addition, the integration of iBioFAB and ML led to an automated closed-loop system named BioAutomata for pathway engineering.[84] Diverse scientific disciplines have taken advantage of laboratory automation e.g. chemical synthesis[85], synthetic biology[86], metabolic engineering[87], and natural product discovery[88].

However, directed protein evolution is yet to take full advantage of recent developments in biofoundries. While *in vivo* continuous evolution approaches remain challenging to adapt to diverse proteins and pathways,[1] the development of an *in vitro* continuous protein evolution platform offers a promising new direction. Even with recent advances in MLDE, the bench work remains time-consuming and requires meticulous attention from skilled researchers. Additionally, automation will allow exploration of higher order variants and global protein landscapes, which is difficult due to experimental limitations. Exploration of higher order variants such as the diverse library of AAV capsids with 12-29 mutations[66] and functional GFP library with as many as 48 mutations[89] can greatly benefit from integrated laboratory automation platforms. A fully autonomous system will require minimal supervision of protocols and processes resulting in faster construction and analysis of predicted variants.

The wet-lab experiments for MLDE can be divided into repetitive execution of multiple molecular biology techniques such as PCR, mutagenesis, library creation, colony picking, miniprep, sequencing, and functional assays, the majority of which are amenable to full automation (Figure 3). Many of these protocols have been automated in previous efforts such as in PlasmidMaker[83], automated yeast genome-scale engineering,[90] and BioAutomata[84]. Following the automation of individual protocols, the high-throughput processes on different instruments are integrated to create an end-to-end pipeline using a robotic arm for plate transfers and scheduling software. For the above-mentioned case studies, the researchers used Thermo F5 robotic arm with Thermo momentum scheduler software for process integration among various core and peripheral equipment. Notably, many specialized instruments for laboratory automation are commercially available that integrate well with robotic arms and scheduling software. High-throughput robotic liquid handlers of different capacities and customizability can be purchased from various vendors such as TECAN, Hamilton, and Agilent. These liquid handlers can be customized to accommodate a variety of smaller instruments such as thermocyclers, shakers, and colony pickers on deck. Acoustic liquid handlers such as Echo can precisely dispense up to nanoliters and can be readily integrated with robotic arms for a variety of assays. Robotics-compatible thermocyclers, fragment analyzer, and sequencers can be integrated with robotic arms that can be used to design a fully automated PCR and mutagenesis protocol. High-throughput plate readers that are compatible with laboratory automation systems can perform a variety of enzymatic assays in an automated fashion. Recent developments in automated microfluidic devices with diverse applications in synthetic biology and proteomics add another exciting dimension to laboratory automation.[91,92] Microfluidic devices that are capable of automating proteomics assays[93,94] further add to the ability to design *in vitro* continuous protein evolution strategies for a diverse set of proteins. The robot scientist concept is another exciting direction for assigning the tedious, repetitive, and high precision tasks to automated systems.[95]

Development of *in vitro* continuous MLDE requires highly specialized skills at the intersection of synthetic biology, computer science, and laboratory automation and robotics. There is a need to train the next generation of synthetic biology researchers with recent developments in biofoundries and laboratory automation. While setting up new biofoundries can be difficult due to a multitude of factors (e.g., high costs, limited funding, lack of trained personnel), an alternative could be a core biofoundry facility where researchers from a variety of experimental disciplines can develop new automation routines. Biofoundries combined with ML will expand the scope for designing efficient directed protein evolution strategies. In the future, researchers can efficiently accomplish *in vitro* continuous protein evolution of almost all proteins with minimal human intervention.

## Conclusion

Directed protein evolution aims to evolve protein variants with properties desirable for scientific and industrial applications. The community has a long history of optimizing the evolutionary process to make it more efficient and comprehensive. By combining ML and automation, *in vitro* continuous protein evolution could greatly expand researchers' ability to evolve and improve proteins. Even though most of the applications of ML in directed protein evolution are focused on using supervised learning to predict variant properties, the past few years have seen an increased use of unsupervised learning to intelligently design variant libraries and explore fitness landscapes. More importantly, the addition of automation can further unleash ML to its full potential by delivering high-quality data acquisition and rapid, reproducible iterative cycles of the experiment. While ML for protein engineering and automation in biotechnology have received early attention, the seamless combination of the two strategies can continuously advance and push the boundaries of directed protein evolution.

## Acknowledgements

This work was supported by the Molecule Maker Lab Institute: An AI Research Institute program supported by U.S. National Science Foundation under grant no. 2019897 (H.Z.) and U.S. Department of Energy award DE-SC0018420 (H.Z.). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect those of the National Science Foundation or Department of Energy. The online tool BioRender (biorender.com) was used to create Figures 1, 2 and 3.

**Figure 1**. An overview of the steps involved in *in vitro* continuous protein evolution. An informed variant library is first designed by zero-shot predictors. The library is then constructed, expressed, and screened by a fully automated robotic platform. The acquired fitness data is then used to train an ML model tasked to predict the fitness landscape. Based on the predicted landscape, subsequent rounds of variant libraries are proposed. Such steps are performed iteratively until the optimal variant is identified.

**Figure 2**. Role of machine learning in *in vitro* continuous protein evolution. (A) Using global or local evolutionary context, an informed variant library can be designed using two types of zero-shot variant effect predictors. While the global model is trained on amino acid sequences from all protein families, the local model calculates the probability based on residue dependencies inferred from multiple sequence alignment. (B) Variant sequences are represented using sequence embeddings obtained from pre-trained global language models. The representation can be further fine-tuned by homologous sequences to be more task-specific. The obtained variant-fitness data can then be used to train a regression model such as a neural network or a simple linear regressor. (C) The protein sequence space can be navigated by incorporating deep generative models, masked language models, or an in silico directed evolution platform. While the last two approaches can navigate the local landscape efficiently, deep generative models can create diverse/highly mutated functional variants to aid in exploration of the global protein landscape. (D) The process of Bayesian optimization with upper confidence bound (UCB) acquisition function. UCB acquisition function not only considers the predicted fitness (mean prediction) but also includes an estimated uncertainty (95% confidence interval used in the figure) to encourage the model to explore a broader region of the protein landscape.

**Figure 3**. The biofoundry can serve as an ideal platform for *in vitro* continuous protein evolution. (A) A framework for continuous MLDE, all experiments can be automated over a biofoundry and combined with machine learning to create a closed-loop protein evolution platform; (B) Layout of an automated biofoundry with various core and peripheral instruments, connected via a central robotic arm and controlled through scheduling software.

## Box 1. Glossary
## A glossary of common technical terms used throughout this manuscript.

Zero-shot:
Zero-shot learning refers to the extension of prediction to a new set of classes which are not observed during model training.

Masking:
Masking is a method used in ML to skip and exclude certain residues from the input amino acid sequences.

Representations:
Representations are a numerical form of the input amino acid sequences which represents their features.

One-hot encoding:
One-hot encoding is one simplest way to represent amino acids sequences. The method represents an amino acid sequence of length L with a matrix whose dimension is L by 20. The matrix is filled with either ones or zeros. Ones are filled in the matrix where the certain position of the amino acid sequence is the certain amino acid at the row, and zeros otherwise.

Natural language processing (NLP):
NLP is a branch of ML studies aiming to let machine understand and makes sense of human languages. Common tasks include spell checking, language translation, topic detection and sentence completion.

Latent vector/features:
Meaningful representation of the input in a vector space obtained generally from an encoder-decoder architecture.

Disentanglement:
Creating latent features independent of one another.

## References

1. Wang, Y., Xue, P., Cao, M., Yu, T., Lane, S.T., and Zhao, H. (2021). Directed Evolution: Methodologies and Applications. Chem. Rev. *121*, 12384-12444. 10.1021/acs.chemrev.1c00260.
2. Bornscheuer, U.T., Hauer, B., Jaeger, K.E., and Schwaneberg, U. (2019). Directed Evolution Empowered Redesign of Natural Proteins for the Sustainable Production of Chemicals and Pharmaceuticals. Angew. Chem. Int. Ed. *58*, 36-40. 10.1002/anie.201812717.
3. Zeymer, C., and Hilvert, D. (2018). Directed Evolution of Protein Catalysts. Annu. Rev. Biochem. *87*, 131-157. 10.1146/annurev-biochem-062917-012034.
4. Wittmann, B.J., Johnston, K.E., Wu, Z., and Arnold, F.H. (2021). Advances in machine learning for directed evolution. Curr. Opin. Struct. Biol. *69*, 11-18. 10.1016/j.sbi.2021.01.008.
5. Mazurenko, S., Prokop, Z., and Damborsky, J. (2020). Machine Learning in Enzyme Engineering. ACS Catal. *10*, 1210-1223. 10.1021/acscatal.9b04321.
6. Molina, R.S., Rix, G., Mengiste, A.A., Álvarez, B., Seo, D., Chen, H., Hurtado, J.E., Zhang, Q., García-García, J.D., Heins, Z.J., et al. (2022). In vivo hypermutation and continuous evolution. Nat Rev Methods Primers *2*, 1-22. 10.1038/s43586-022-00119-5.
7. Ibrahim, S.F., and van den Engh, G. (2007). Flow Cytometry and Cell Sorting. In Adv. Biochem. Eng. Biotechnol., 19-39. 10.1007/10_2007_073.
8. Li, M. (2000). Applications of display technology in protein analysis. Nat Biotechnol *18*, 1251-1256. 10.1038/82355.
9. Yang, K.K., Wu, Z., and Arnold, F.H. (2019). Machine-learning-guided directed evolution for protein engineering. Nat Methods *16*, 687-694. 10.1038/s41592-019-0496-6.
10. Hie, B.L., and Yang, K.K. (2022). Adaptive machine learning for protein engineering. Current Opinion in Structural Biology *72*, 145-152. 10.1016/j.sbi.2021.11.002.
11. Meier, J., Rao, R., Verkuil, R., Liu, J., Sercu, T., and Rives, A. (2021/11/17/). Language models enable zero-shot prediction of the effects of mutations on protein function. bioRxiv. 10.1101/2021.07.09.450648.
12. Riesselman, A.J., Ingraham, J.B., and Marks, D.S. (2018). Deep generative models of genetic variation capture the effects of mutations. Nat Methods *15*, 816-822. 10.1038/s41592-018-0138-4.
13. Rao, R., Bhattacharya, N., Thomas, N., Duan, Y., Chen, X., Canny, J., Abbeel, P., and Song, Y.S. (2019). Evaluating Protein Transfer Learning with TAPE. Adv Neural Inf Process Syst *32*, 9689-9701.
14. Wittmann, B.J., Yue, Y., and Arnold, F.H. (2020/12/04/). Machine Learning-Assisted Directed Evolution Navigates a Combinatorial Epistatic Fitness Landscape with Minimal Screening Burden. bioRxiv 10.1101/2020.12.04.408955.
15. Biswas, S., Khimulya, G., Alley, E.C., Esvelt, K.M., and Church, G.M. (2021). Low-N protein engineering with data-efficient deep learning. Nat Methods *18*, 389-396. 10.1038/s41592-021-01100-y.
16. Hsu, C., Nisonoff, H., Fannjiang, C., and Listgarten, J. (2022). Learning protein fitness models from evolutionary and assay-labeled data. Nat Biotechnol, 1-9. 10.1038/s41587-021-01146-5.
17. Luo, Y., Jiang, G., Yu, T., Liu, Y., Vo, L., Ding, H., Su, Y., Qian, W.W., Zhao, H., and Peng, J. (2021). ECNet is an evolutionary context-integrated deep learning framework for protein engineering. Nat Commun *12*, 5743. 10.1038/s41467-021-25976-8.

18. Gazut, S., Martinez, J.-M., Dreyfus, G., and Oussar, Y. (2008). Towards the Optimal Design of Numerical Experiments. IEEE Transactions on Neural Networks *19*, 874-882. 10.1109/TNN.2007.915111.
19. Li, G., Dong, Y., and Reetz, M.T. (2019). Can Machine Learning Revolutionize Directed Evolution of Selective Enzymes? Advanced Synthesis & Catalysis *361*, 2377-2386. 10.1002/adsc.201900149.
20. Siedhoff, N.E., Schwaneberg, U., and Davari, M.D. (2020). Chapter Twelve - Machine learning-assisted enzyme engineering. In Methods in Enzymology, D.S. Tawfik, ed. (Academic Press), pp. 281-315.
21. Strokach, A., and Kim, P.M. (2022). Deep generative modeling for protein design. Current Opinion in Structural Biology *72*, 226-236. 10.1016/j.sbi.2021.11.008.
22. Hie, B., Bryson, B.D., and Berger, B. (2020). Leveraging Uncertainty in Machine Learning Accelerates Biological Discovery and Design. Cell Systems *11*, 461-477.e469. 10.1016/j.cels.2020.09.007.
23. Romero, P.A., Krause, A., and Arnold, F.H. (2013). Navigating the protein fitness landscape with Gaussian processes. Proceedings of the National Academy of Sciences *110*, E193-E201. 10.1073/pnas.1215251110.
24. Wittmann, B.J., Yue, Y., and Arnold, F.H. (2021). Informed training set design enables efficient machine learning-assisted directed protein evolution. Cell Systems *12*, 1026-1045.e1027. 10.1016/j.cels.2021.07.008.
25. Hopf, T.A., Ingraham, J.B., Poelwijk, F.J., Schärfe, C.P.I., Springer, M., Sander, C., and Marks, D.S. (2017). Mutation effects predicted from sequence co-variation. Nat Biotechnol *35*, 128-135. 10.1038/nbt.3769.
26. Rives, A., Meier, J., Sercu, T., Goyal, S., Lin, Z., Liu, J., Guo, D., Ott, M., Zitnick, C.L., Ma, J., and Fergus, R. (2021). Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. Proceedings of the National Academy of Sciences *118*, e2016239118. 10.1073/pnas.2016239118.
27. Elnaggar, A., Heinzinger, M., Dallago, C., Rehawi, G., Wang, Y., Jones, L., Gibbs, T., Feher, T., Angerer, C., Steinegger, M., et al. (2021). ProtTrans: Towards Cracking the Language of Lifes Code Through Self-Supervised Deep Learning and High Performance Computing. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1-1. 10.1109/TPAMI.2021.3095381.
28. Rao, R., Liu, J., Verkuil, R., Meier, J., Canny, J.F., Abbeel, P., Sercu, T., and Rives, A. (2021/08/27/). MSA Transformer. bioRxiv. 10.1101/2021.02.12.430858.
29. Madani, A., Krause, B., Greene, E.R., Subramanian, S., Mohr, B.P., Holton, J.M., Olmos, J.L., Xiong, C., Sun, Z.Z., Socher, R., et al. (2023). Large language models generate functional protein sequences across diverse families. Nat Biotechnol. 10.1038/s41587-022-01618-2.
30. Nijkamp, E., Ruffolo, J., Weinstein, E.N., Naik, N., and Madani, A. (2022/06/27/). ProGen2: Exploring the Boundaries of Protein Language Models. bioRxiv. 10.48550/arXiv.2206.13517.
31. Hawkins-Hooker, A., Depardieu, F., Baur, S., Couairon, G., Chen, A., and Bikard, D. (2021). Generating functional protein variants with variational autoencoders. PLOS Computational Biology *17*, e1008736. 10.1371/journal.pcbi.1008736.
32. Repecka, D., Jauniskis, V., Karpus, L., Rembeza, E., Rokaitis, I., Zrimec, J., Poviloniene, S., Laurynenas, A., Viknander, S., Abuajwa, W., et al. (2021). Expanding functional protein sequence spaces using generative adversarial networks. Nat Mach Intell *3*, 324-333. 10.1038/s42256-021-00310-5.
33. Nobili, A., Gall, M.G., Pavlidis, I.V., Thompson, M.L., Schmidt, M., and Bornscheuer, U.T. (2013). Use of 'small but smart' libraries to enhance the enantioselectivity of an

esterase from Bacillus stearothermophilus towards tetrahydrofuran-3-yl acetate. The FEBS Journal *280*, 3084-3093. 10.1111/febs.12137.

34. Jochens, H., and Bornscheuer, U.T. (2010). Natural Diversity to Guide Focused Directed Evolution. ChemBioChem *11*, 1861-1866. 10.1002/cbic.201000284.

35. Hulley, M.E., Toogood, H.S., Fryszkowska, A., Mansell, D., Stephens, G.M., Gardiner, J.M., and Scrutton, N.S. (2010). Focused Directed Evolution of Pentaerythritol Tetranitrate Reductase by Using Automated Anaerobic Kinetic Screening of Site-Saturated Libraries. ChemBioChem *11*, 2433-2447. 10.1002/cbic.201000527.

36. Gustafsson, C., Govindarajan, S., and Minshull, J. (2003). Putting engineering back into protein engineering: bioinformatic approaches to catalyst design. Current Opinion in Biotechnology *14*, 366-370. 10.1016/S0958-1669(03)00101-0.

37. Lu, H., Diaz, D.J., Czarnecki, N.J., Zhu, C., Kim, W., Shroff, R., Acosta, D.J., Alexander, B.R., Cole, H.O., Zhang, Y., et al. (2022). Machine learning-aided engineering of hydrolases for PET depolymerization. Nature *604*, 662-667. 10.1038/s41586-022-04599-z.

38. Xu, Y., Verma, D., Sheridan, R.P., Liaw, A., Ma, J., Marshall, N.M., McIntosh, J., Sherer, E.C., Svetnik, V., and Johnston, J.M. (2020). Deep Dive into Machine Learning Models for Protein Engineering. J. Chem. Inf. Model. *60*, 2773-2790. 10.1021/acs.jcim.0c00073.

39. Alley, E.C., Khimulya, G., Biswas, S., AlQuraishi, M., and Church, G.M. (2019). Unified rational protein engineering with sequence-based deep representation learning. Nat Methods *16*, 1315-1322. 10.1038/s41592-019-0598-1.

40. Melidis, D.P., and Nejdl, W. (2021). Capturing Protein Domain Structure and Function Using Self-Supervision on Domain Architectures. Algorithms *14*, 28. 10.3390/a14010028.

41. Yang, K.K., Lu, A.X., and Fusi, N. (2022/05/25/). Convolutions are competitive with transformers for protein sequence pretraining. bioRxiv. 10.1101/2022.05.19.492714.

42. Ma, E.J., Siirola, E., Moore, C., Kummer, A., Stoeckli, M., Faller, M., Bouquet, C., Eggimann, F., Ligibel, M., Huynh, D., et al. (2021). Machine-Directed Evolution of an Imine Reductase for Activity and Stereoselectivity. ACS Catal. *11*, 12433-12445. 10.1021/acscatal.1c02786.

43. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is All you Need. Advances in Neural Information Processing Systems, 2017. (Curran Associates, Inc.).

44. Shanehsazzadeh, A., Belanger, D., and Dohan, D. (2020/10/31). Is Transfer Learning Necessary for Protein Landscape Prediction? arXiv. 10.48550/arXiv.2011.03443.

45. Shamsi, Z., Chan, M., and Shukla, D. (2020). TLmutation: Predicting the Effects of Mutations Using Transfer Learning. J. Phys. Chem. B *124*, 3845-3854. 10.1021/acs.jpcb.0c00197.

46. Yang, K.K., Wu, Z., Bedbrook, C.N., and Arnold, F.H. (2018). Learned protein embeddings for machine learning. Bioinformatics *34*, 2642-2648. 10.1093/bioinformatics/bty178.

47. Lu, A.X., Zhang, H., Ghassemi, M., and Moses, A. (2020/11/10/). Self-Supervised Contrastive Learning of Protein Representations By Mutual Information Maximization. bioRxiv. 10.1101/2020.09.04.283929.

48. Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. (2021). Highly accurate protein structure prediction with AlphaFold. Nature *596*, 583-589. 10.1038/s41586-021-03819-2.

49. Varadi, M., Anyango, S., Deshpande, M., Nair, S., Natassia, C., Yordanova, G., Yuan, D., Stroe, O., Wood, G., Laydon, A., et al. (2022). AlphaFold Protein Structure Database:

massively expanding the structural coverage of protein-sequence space with high-accuracy models. Nucleic Acids Research *50*, D439-D444. 10.1093/nar/gkab1061.

50. Baek, M., DiMaio, F., Anishchenko, I., Dauparas, J., Ovchinnikov, S., Lee, G.R., Wang, J., Cong, Q., Kinch, L.N., Schaeffer, R.D., et al. (2021). Accurate prediction of protein structures and interactions using a three-track neural network. Science *373*, 871-876. 10.1126/science.abj8754.
51. Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Costa, A.d.S., Fazel-Zarandi, M., Sercu, T., Candido, S., and Rives, A. (2022/07/21/). Language models of protein sequences at the scale of evolution enable accurate structure prediction. bioRxiv. 10.1101/2022.07.20.500902.
52. Hie, B.L., Xu, D., Shanker, V.R., Bruun, T.U.J., Weidenbacher, P.A., Tang, S., and Kim, P.S. (2022/04/11/). Efficient evolution of human antibodies from general protein language models and sequence information alone. bioRxiv. 10.1101/2022.04.10.487811.
53. Schissel, C.K., Mohapatra, S., Wolfe, J.M., Fadzen, C.M., Bellovoda, K., Wu, C.-L., Wood, J.A., Malmberg, A.B., Loas, A., Gómez-Bombarelli, R., and Pentelute, B.L. (2021). Deep learning to design nuclear-targeting abiotic miniproteins. Nat. Chem. *13*, 992-1000. 10.1038/s41557-021-00766-3.
54. Yoshida, M., Hinkley, T., Tsuda, S., Abul-Haija, Y.M., McBurney, R.T., Kulikov, V., Mathieson, J.S., Galiñanes Reyes, S., Castro, M.D., and Cronin, L. (2018). Using Evolutionary Algorithms and Machine Learning to Explore Sequence Space for the Discovery of Antimicrobial Peptides. Chem *4*, 533-543. 10.1016/j.chempr.2018.01.005.
55. Giessel, A., Dousis, A., Ravichandran, K., Smith, K., Sur, S., McFadyen, I., Zheng, W., and Licht, S. (2022). Therapeutic enzyme engineering using a generative neural network. Sci Rep *12*, 1536. 10.1038/s41598-022-05195-x.
56. Ferruz, N., Schmidt, S., and Höcker, B. (2022). ProtGPT2 is a deep unsupervised language model for protein design. Nat Commun *13*, 4348. 10.1038/s41467-022-32007-7.
57. Schmitt, L.T., Paszkowski-Rogacz, M., Jug, F., and Buchholz, F. (2022). Prediction of designer-recombinases for DNA editing with generative deep learning. Nat Commun *13*, 7966. 10.1038/s41467-022-35614-6.
58. Wu, Z., Yang, K.K., Liszka, M.J., Lee, A., Batzilla, A., Wernick, D., Weiner, D.P., and Arnold, F.H. (2020). Signal Peptides Generated by Attention-Based Neural Networks. ACS Synth. Biol. *9*, 2154-2161. 10.1021/acssynbio.0c00219.
59. Brookes, D.H., and Listgarten, J. (2020/02/10/). Design by adaptive sampling. arXiv. 10.48550/arXiv.1810.03714.
60. Gupta, A., and Zou, J. (2019). Feedback GAN for DNA optimizes protein functions. Nat Mach Intell *1*, 105-111. 10.1038/s42256-019-0017-4.
61. Amimeur, T., Shaver, J.M., Ketchem, R.R., Taylor, J.A., Clark, R.H., Smith, J., Van Citters, D., Siska, C.C., Smidt, P., Sprague, M., et al. (2020/04/13/). Designing Feature-Controlled Humanoid Antibody Discovery Libraries Using Generative Adversarial Networks. bioRxiv.
62. Chan, A., Madani, A., Krause, B., and Naik, N. (2021/10/25/). Deep Extrapolation for Attribute-Enhanced Generation. arXiv. 10.48550/arXiv.2107.02968.
63. Bepler, T., and Berger, B. (2021). Learning the protein language: Evolution, structure, and function. Cell Systems *12*, 654-669.e653. 10.1016/j.cels.2021.05.017.
64. Fox, R.J., Davis, S.C., Mundorff, E.C., Newman, L.M., Gavrilovic, V., Ma, S.K., Chung, L.M., Ching, C., Tam, S., Muley, S., et al. (2007). Improving catalytic function by ProSAR-driven enzyme evolution. Nat Biotechnol *25*, 338-344. 10.1038/nbt1286.

65. Wiseman, S., and Rush, A.M. (2016/11/09/). Sequence-to-Sequence Learning as Beam-Search Optimization. arXiv. 10.48550/arXiv.1606.02960.

66. Bryant, D.H., Bashir, A., Sinai, S., Jain, N.K., Ogden, P.J., Riley, P.F., Church, G.M., Colwell, L.J., and Kelsic, E.D. (2021). Deep diversification of an AAV capsid protein by machine learning. Nat Biotechnol *39*, 691-696. 10.1038/s41587-020-00793-4.

67. Wu, Z., Kan, S.B.J., Lewis, R.D., Wittmann, B.J., and Arnold, F.H. (2019). Machine learning-assisted directed protein evolution with combinatorial libraries. Proceedings of the National Academy of Sciences *116*, 8852-8858. 10.1073/pnas.1901979116.

68. Osadchy, M., and Kolodny, R. (2021). How Deep Learning Tools Can Help Protein Engineers Find Good Sequences. J. Phys. Chem. B *125*, 6440-6450. 10.1021/acs.jpcb.1c02449.

69. Bedbrook, C.N., Yang, K.K., Rice, A.J., Gradinaru, V., and Arnold, F.H. (2017). Machine learning to design integral membrane channelrhodopsins for efficient eukaryotic expression and plasma membrane localization. PLOS Computational Biology *13*, e1005786. 10.1371/journal.pcbi.1005786.

70. Snoek, J., Larochelle, H., and Adams, R.P. (2012/06/13). Practical bayesian optimization of machine learning algorithms. arXiv *25*. 10.48550/arXiv.1206.2944.

71. Shmilovich, K., Mansbach, R.A., Sidky, H., Dunne, O.E., Panda, S.S., Tovar, J.D., and Ferguson, A.L. (2020). Discovery of Self-Assembling π-Conjugated Peptides by Active Learning-Directed Coarse-Grained Molecular Simulation. J. Phys. Chem. B *124*, 3873-3891. 10.1021/acs.jpcb.0c00708.

72. Lamparth, M., Bestehorn, M., and Märkisch, B. (2022/05/27/). Gaussian Processes and Bayesian Optimization for High Precision Experiments. arXiv. 10.48550/arXiv.2205.07625.

73. Bedbrook, C.N., Yang, K.K., Robinson, J.E., Mackey, E.D., Gradinaru, V., and Arnold, F.H. (2019). Machine learning-guided channelrhodopsin engineering enables minimally invasive optogenetics. Nat Methods *16*, 1176-1184. 10.1038/s41592-019-0583-8.

74. Moss, H., Leslie, D., Beck, D., González, J., and Rayson, P. (2020). BOSS: Bayesian Optimization over String Spaces. Advances in Neural Information Processing Systems, 2020. (Curran Associates, Inc.), pp. 15476-15486.

75. Greenhalgh, J.C., Fahlberg, S.A., Pfleger, B.F., and Romero, P.A. (2021). Machine learning-guided acyl-ACP reductase engineering for improved in vivo fatty alcohol production. Nat Commun *12*, 5825. 10.1038/s41467-021-25831-w.

76. Neal, R.M. (2012). Bayesian Learning for Neural Networks (Springer Science & Business Media).

77. Lakshminarayanan, B., Pritzel, A., and Blundell, C. (2017/11/03/). Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. arXiv. 10.48550/arXiv.1612.01474.

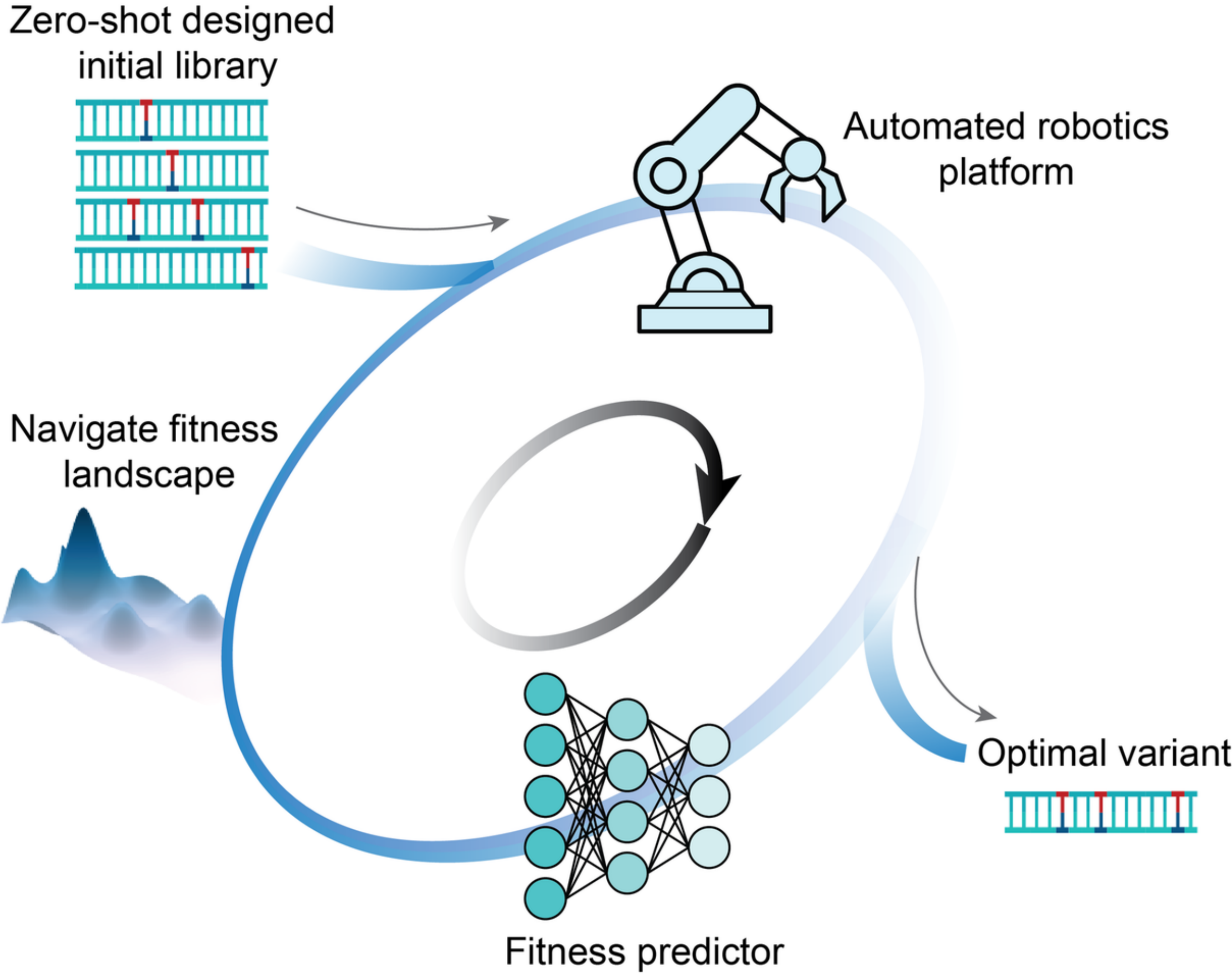
78. Chao, R., Mishra, S., Si, T., and Zhao, H. (2017). Engineering biological systems using automated biofoundries. Metabolic Engineering *42*, 98-108. 10.1016/j.ymben.2017.06.003.

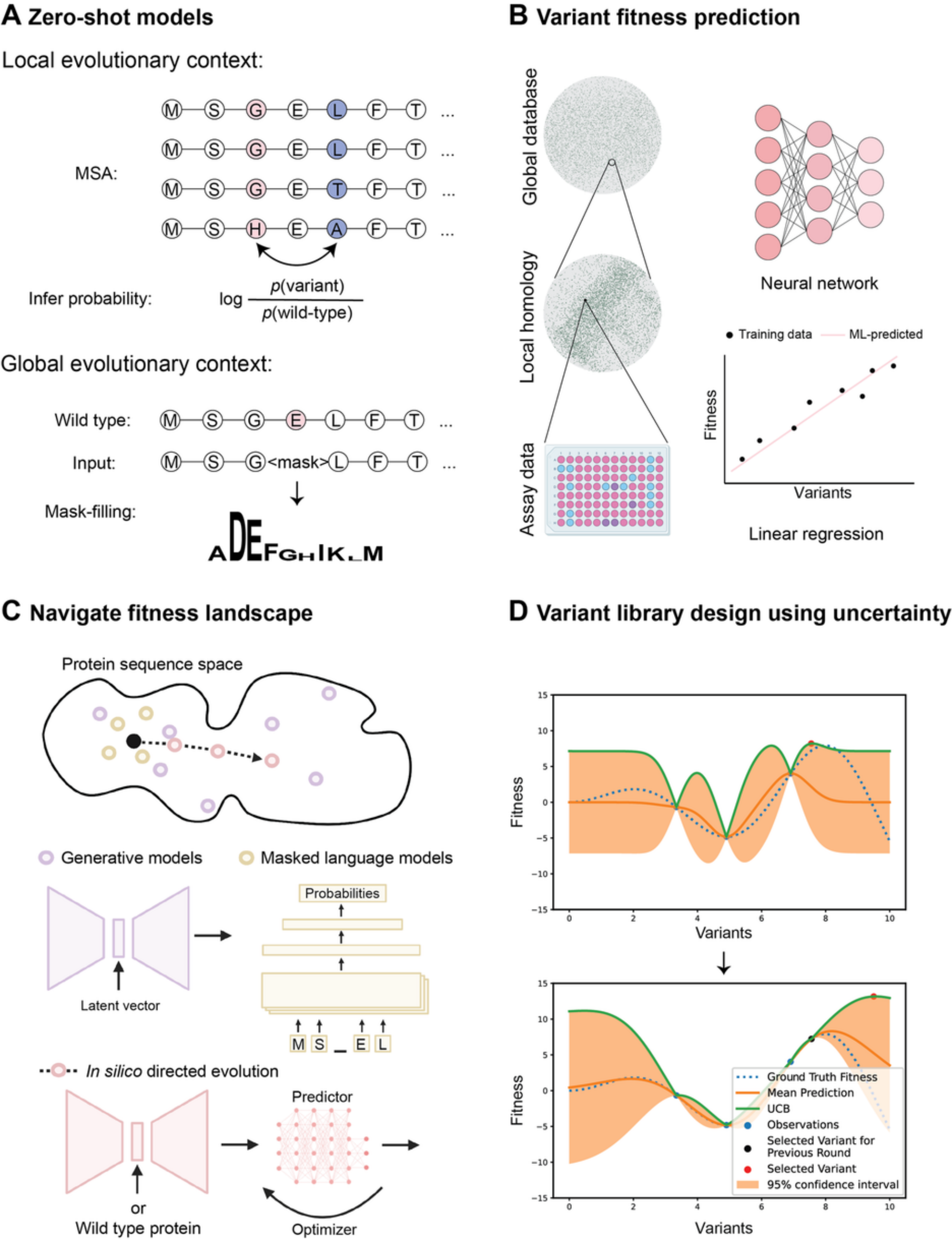
79. Hillson, N., Caddick, M., Cai, Y., Carrasco, J.A., Chang, M.W., Curach, N.C., Bell, D.J., Le Feuvre, R., Friedman, D.C., Fu, X., et al. (2019). Building a global alliance of biofoundries. Nat Commun *10*, 2040. 10.1038/s41467-019-10079-2.

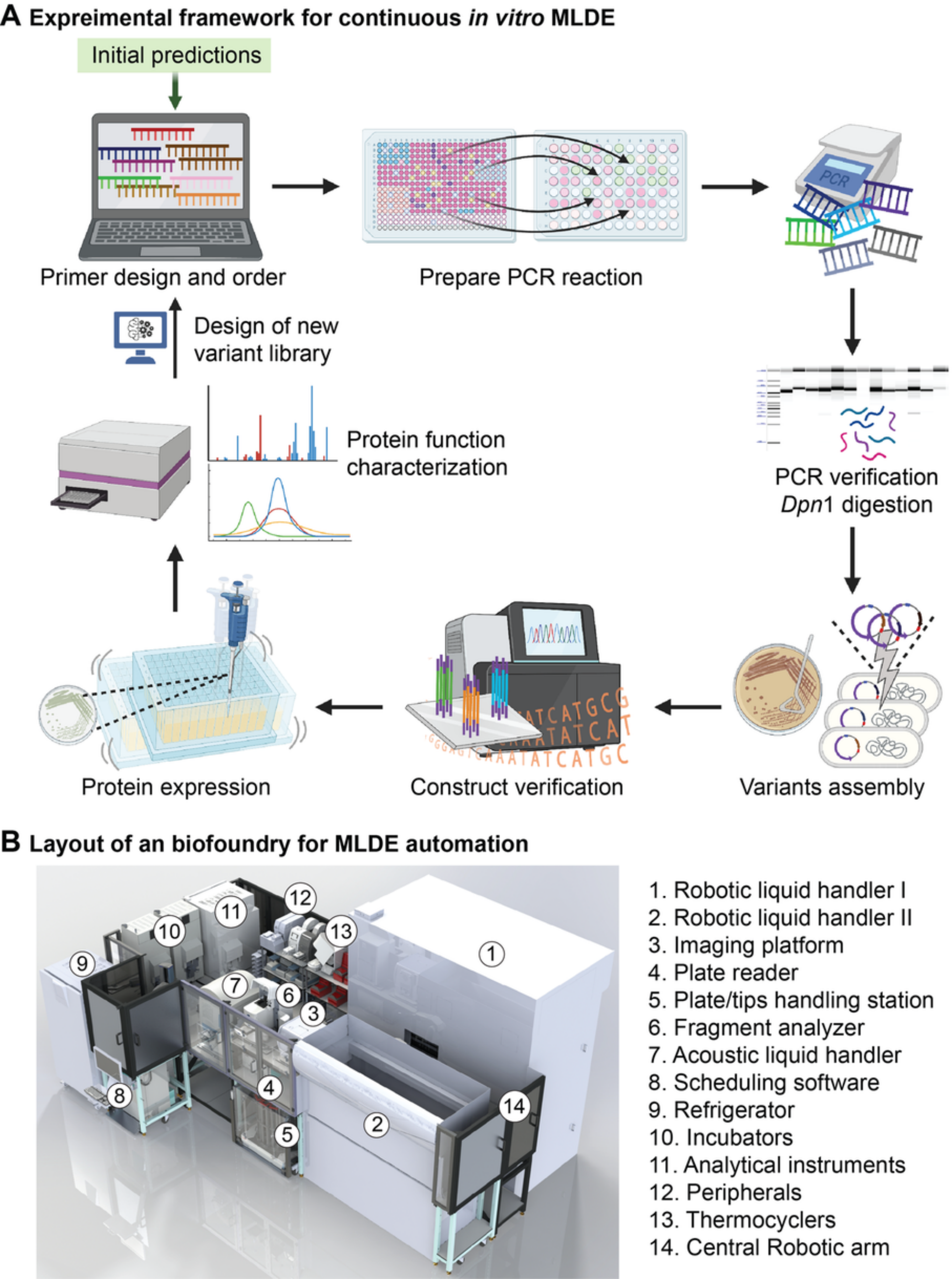
80. Christensen, M., Yunker, L.P.E., Shiri, P., Zepel, T., Prieto, P.L., Grunert, S., Bork, F., and Hein, J.E. (2021). Automation isn't automatic. Chem. Sci. *12*, 15473-15490. 10.1039/D1SC04588A.

81. Pavan, M. (2021). Setting Up an Automated Biomanufacturing Laboratory. Methods Mol Biol *2229*, 137-155. 10.1007/978-1-0716-1032-9_5.

82. Chao, R., Liang, J., Tasan, I., Si, T., Ju, L., and Zhao, H. (2017). Fully Automated One-Step Synthesis of Single-Transcript TALEN Pairs Using a Biological Foundry. ACS Synth. Biol. *6*, 678-685. 10.1021/acssynbio.6b00293.
83. Enghiad, B., Xue, P., Singh, N., Boob, A.G., Shi, C., Petrov, V.A., Liu, R., Peri, S.S., Lane, S.T., Gaither, E.D., and Zhao, H. (2022). PlasmidMaker is a versatile, automated, and high throughput end-to-end platform for plasmid construction. Nat Commun *13*, 2697. 10.1038/s41467-022-30355-y.
84. HamediRad, M., Chao, R., Weisberg, S., Lian, J., Sinha, S., and Zhao, H. (2019). Towards a fully automated algorithm driven platform for biosystems design. Nat Commun *10*, 5150. 10.1038/s41467-019-13189-z.
85. Angello, N.H., Rathore, V., Beker, W., Wołos, A., Jira, E.R., Roszak, R., Wu, T.C., Schroeder, C.M., Aspuru-Guzik, A., Grzybowski, B.A., and Burke, M.D. (2022). Closed-loop optimization of general reaction conditions for heteroaryl Suzuki-Miyaura coupling. Science *378*, 399-405. 10.1126/science.adc8743.
86. Radivojević, T., Costello, Z., Workman, K., and Garcia Martin, H. (2020). A machine learning Automated Recommendation Tool for synthetic biology. Nat Commun *11*, 4879. 10.1038/s41467-020-18008-4.
87. Otero-Muras, I., and Carbonell, P. (2021). Automated engineering of synthetic metabolic pathways for efficient biomanufacturing. Metabolic Engineering *63*, 61-80. 10.1016/j.ymben.2020.11.012.
88. Ayikpoe, R.S., Shi, C., Battiste, A.J., Eslami, S.M., Ramesh, S., Simon, M.A., Bothwell, I.R., Lee, H., Rice, A.J., Ren, H., et al. (2022). A scalable platform to discover antimicrobials of ribosomal origin. Nat Commun *13*, 6135. 10.1038/s41467-022-33890-w.
89. Gonzalez Somermeyer, L., Fleiss, A., Mishin, A.S., Bozhanova, N.G., Igolkina, A.A., Meiler, J., Alaball Pujol, M.-E., Putintseva, E.V., Sarkisyan, K.S., and Kondrashov, F.A. (2022). Heterogeneity of the GFP fitness landscape and data-driven protein design. eLife *11*, e75842. 10.7554/eLife.75842.
90. Si, T., Chao, R., Min, Y., Wu, Y., Ren, W., and Zhao, H. (2017). Automated multiplex genome-scale engineering in yeast. Nat Commun *8*, 15187. 10.1038/ncomms15187.
91. Bowman, E.K., and Alper, H.S. (2020). Microdroplet-Assisted Screening of Biomolecule Production for Metabolic Engineering Applications. Trends in Biotechnology *38*, 701-714. 10.1016/j.tibtech.2019.11.002.
92. Linshiz, G., Jensen, E., Stawski, N., Bi, C., Elsbree, N., Jiao, H., Kim, J., Mathies, R., Keasling, J.D., and Hillson, N.J. (2016). End-to-end automated microfluidic platform for synthetic biology: from design to functional analysis. Journal of Biological Engineering *10*, 3. 10.1186/s13036-016-0024-5.
93. Chen, Y., Guenther, J.M., Gin, J.W., Chan, L.J.G., Costello, Z., Ogorzalek, T.L., Tran, H.M., Blake-Hedges, J.M., Keasling, J.D., Adams, P.D., et al. (2019). Automated “Cells-To-Peptides” Sample Preparation Workflow for High-Throughput, Quantitative Proteomic Assays of Microbes. J. Proteome Res. *18*, 3752-3761. 10.1021/acs.jproteome.9b00455.
94. Diefenbach, X.W., Farasat, I., Guetschow, E.D., Welch, C.J., Kennedy, R.T., Sun, S., and Moore, J.C. (2018). Enabling Biocatalysis by High-Throughput Protein Engineering Using Droplet Microfluidics Coupled to Mass Spectrometry. ACS Omega *3*, 1498-1508. 10.1021/acsomega.7b01973.
95. King, R.D., Schuler Costa, V., Mellingwood, C., and Soldatova, L.N. (2018). Automating Sciences: Philosophical and Social Dimensions. IEEE Technology and Society Magazine *37*, 40-46. 10.1109/MTS.2018.2795097.


Zero-shot designed initial library
Automated robotics platform
Navigate fitness landscape
Optimal variant
Fitness predictor

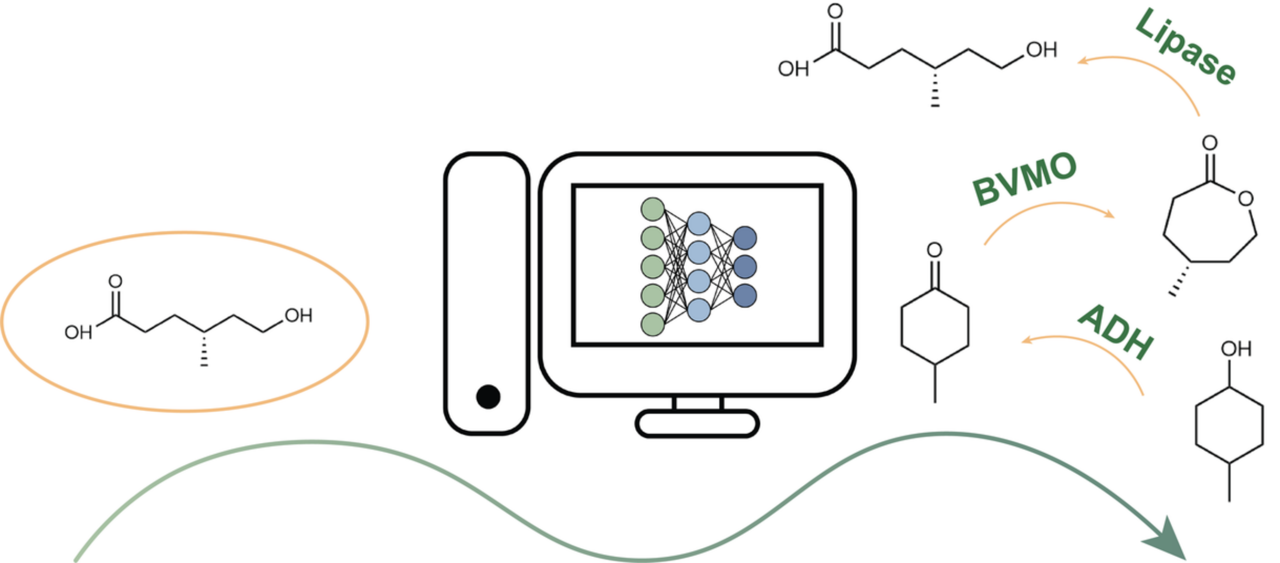

A Zero-shot models
Local evolutionary context:
MSA:
M S G E L F T ...
M S G E L F T ...
M S G E T F T ...
M S H E A F T ...
Infer probability:
$\log \frac{p(\text{variant})}{p(\text{wild-type})}$
Global evolutionary context:
Wild type: M S G E L F T ...
Input: M S G <mask> L F T ...
Mask-filling:
A D E F G H I K L M
B Variant fitness prediction
Global database
Local homology
Assay data
Neural network
Training data
ML-predicted
Fitness
Variants
Linear regression
C Navigate fitness landscape
Protein sequence space
Generative models
Masked language models
Probabilities
Latent vector
M S _ E L
In silico directed evolution
Predictor
or
Wild type protein
Optimizer
D Variant library design using uncertainty
15
10
5
0
−5
−10
−15
Fitness
0
2
4
6
8
10
Variants
Ground Truth Fitness
Mean Prediction
UCB
Observations
Selected Variant for Previous Round
Selected Variant
95% confidence interval



**A** Expreimental framework for continuous *in vitro* MLDE

**B** Layout of an biofoundry for MLDE automation

1. Robotic liquid handler I
2. Robotic liquid handler II
3. Imaging platform
4. Plate reader
5. Plate/tips handling station
6. Fragment analyzer
7. Acoustic liquid handler
8. Scheduling software
9. Refrigerator
10. Incubators
11. Analytical instruments
12. Peripherals
13. Thermocyclers
14. Central Robotic arm


Lipase
BVMO
ADH
OH
O
OH
OH
O
O
O
O
OH
OH
O
OH