Data-driven protein engineering

Jonathan Greenhalgh[1*], Apoorv Saraogee[1*], and Philip A. Romero[1,2]

1. Department of Chemical and Biological Engineering, University of Wisconsin--Madison
2. Department of Biochemistry, University of Wisconsin--Madison
*these authors contributed equally to this work

Introduction

A protein's sequence of amino acids encodes its function. This "function" could refer to a protein's natural biological function, or it could also be any other property including binding affinity toward a particular ligand, thermodynamic stability, or catalytic activity. A detailed understanding of how these functions are encoded would allow us to more accurately reconstruct the tree of life and possibly predict future evolutionary events, diagnose genetic diseases before they manifest symptoms, and design new proteins with useful properties. We know that a protein sequence folds into a three-dimensional structure, and this structure positions specific chemical groups to perform a function; however, we're missing the quantitative details of this sequence-structure-function mapping. This mapping is extraordinarily complex because it involves thousands of molecular interactions that are dynamically coupled across multiple length and time scales.

Computational methods can be used to model the mapping from sequence to structure to function. Tools such as molecular dynamics simulations or Rosetta use atomic representations of protein structures and physics-based energy functions to model structures and functions (1–3). While these models are based on well-founded physical principles, they often fail to capture a protein's overall global behavior and properties. There are numerous challenges associated with physics-based models including consideration of conformational dynamics, the requirement to make energy function approximations for the sake of computational efficiency, and the fact that, for many complex properties such as enzyme catalysis, the molecular basis is simply unknown (4). In systems composed of thousands of atoms, the propagation of small errors quickly overwhelms any predictive accuracy. Despite tremendous breakthroughs and research progress over the last century, we still lack the key details to reliably predict, simulate, and design protein function.

In this chapter, we present the emerging field of data-driven protein engineering. Instead of physically modeling the relationships between protein sequence, structure, and function, data-driven methods use ideas from statistics and machine learning to infer these complex relationships from data. This top-down modeling approach implicitly captures the numerous and possibly unknown factors that shape the mapping from sequence to function. Statistical models have been used to understand the molecular basis of protein function and provide exceptional predictive accuracy for protein design.

The data revolution in biology

The volume of biological data has exploded over the last decade. This is being driven by advances in our ability to read and write DNA, which are progressing faster than Moore's law (5). Simultaneously, we have also gained unprecedented ability to characterize biological systems with advances in automation, miniaturization, multiplex assays, and genome engineering. It is now routine to perform experiments on thousands to millions of molecules, genes, proteins, and/or cells. The resulting data provides a unique opportunity to study biological systems in a comprehensive and unbiased manner.

Protein sequence and structure databases have been growing exponentially for decades (**Fig 1bc**). Currently, the UniProt database (6) contains over 100 million unique protein sequences and the Protein Data Bank (7) contains over 100,000 experimentally determined

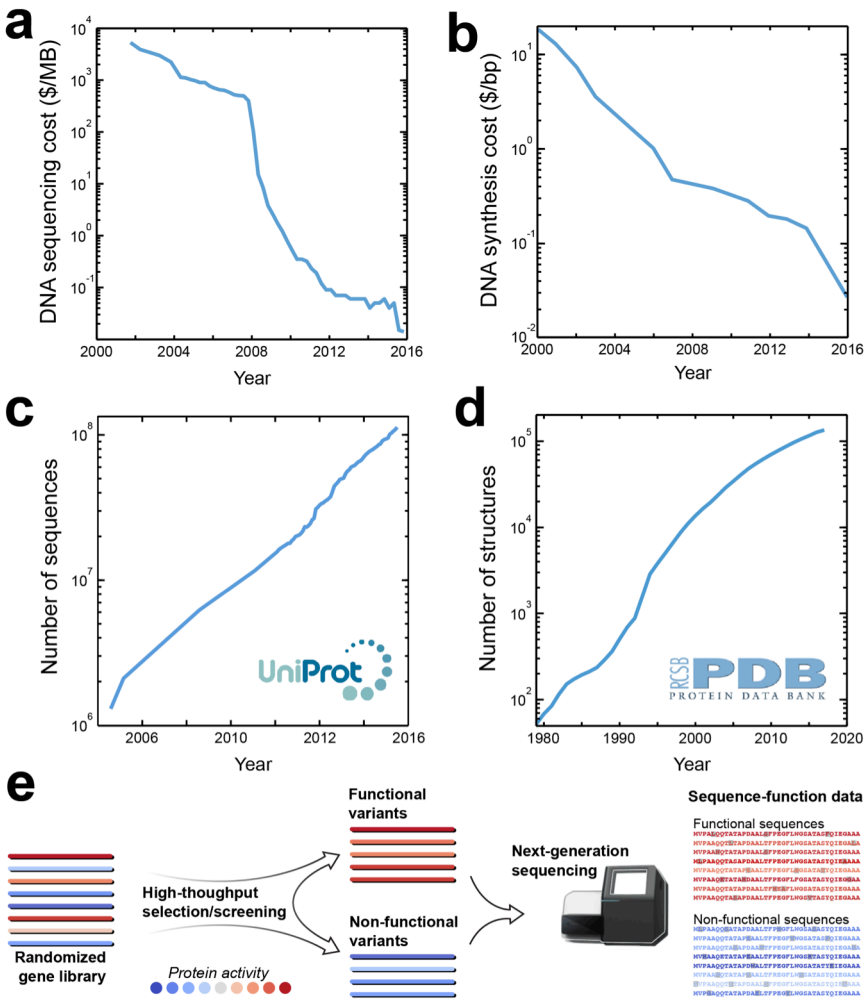

Figure 1: The growth of biological data. (a,b) DNA sequencing and synthesis technologies are advancing faster than Moore's law. As a result, costs have decreased exponentially over the last two decades. (c,d) Large-scale genomics, metagenomics, and structural genomics initiatives have resulted in exponential growth of protein sequence and structure databases. (e) Deep mutational scanning experiments combine high-throughput screens/selections with next-generation DNA sequencing to map sequence-function

protein structures. While there is an abundance of protein sequence and structure data, there is still relatively little data mapping sequence to function. ProtaBank is a new effort to build a protein function database (8). However, function data is challenging to standardize because it is highly dependent on experimental conditions and even the particular researcher that performed the experiments. Therefore, most data-driven protein modeling approaches utilize sequence-function data for a particular protein family that is generated by a single research group. This allows a consistent definition of "function" that is not influenced by uncontrolled experimental factors.

Many sequence-function data sets are generated by protein engineering experiments that involve screening libraries of sequence variants for improved function. These variants may include natural homologs, random mutants, targeted mutants, chimeric proteins generated by homologous recombination, and computationally designed sequences. Each of these sequence diversification methods explores different features of the sequence-function mapping and varies in their information content. Important factors include the sequence diversity of a library, the

likelihood of functional vs nonfunctional sequences, and the difficulty/cost of building the desired gene sequences.

Recent advances in high-throughput experimentation have enabled researchers to map sequence-function relationships for thousands to millions of protein variants (9, 10). These "deep mutational scanning" experiments start with a large library of protein variants, and this library is passed through a high-throughput screen/selection to separate variants based on their functional properties (**Fig 1e**). The genes from these variant pools are then extracted and analyzed using next-generation DNA sequencing. Deep mutational scanning experiments generate data containing millions of sequences and how those sequences map to different functional classes (e.g. active/inactive, binds ligand 1/binds ligand and 2). The resulting data have been used to study the structure of the protein fitness landscape, discover new functional sites, improve molecular energy functions, and identify beneficial combinations of mutations for protein engineering (9, 11–13).

Statistical representations of protein sequence, structure, and function

The growing trove of biological data can be mined to understand the relationships between protein sequence, structure, and function. This complex and heterogenous protein data needs to be represented in simple, machine-readable formats to leverage advanced tools in pattern recognition and machine learning. There are many possible ways of representing proteins mathematically including simple sequence-based representations or more advanced structure/ physics-based representations. In general, a good representation is low dimensional but still captures the system's relevant degrees of freedom.

Representing protein sequences

A protein's amino acid sequence contains all the information necessary to specify its structure and function. Each position in this sequence can be modeled as a categorical variable that can take on one of twenty amino acid values. Categorical data can be represented using a one-hot encoding strategy that assigns one bit to each possible category. If a particular observation falls into one of these categories, it is assigned a "1" at that category's bit, otherwise it is assigned a "0." A protein sequence of length *l* can be represented with a vector of 20*l* bits; 20 bits for each sequence position (**Fig 2**). For example, assuming the amino acid bits are arranged in alphabetical order (A, C, D, E ... W, Y), if a protein has alanine (A) at the first position, the first bit would be 1 and the next 19 bits would be 0. If a protein has aspartic acid (D) at the first position, the first two bits would be 0, the third bit 1, and the next 17 bits 0. This encoding strategy can be applied to all amino acid positions in a protein and represent any sequence of length *l*. One-hot encoding sequence representations are widely used in machine learning because they are simple and flexible. However, they are also very high dimensional (20*l* ≈ thousands of variables for most proteins) and therefore require large quantities of data for learning.

Machine learning is widely used in the fields of text mining and natural language processing to understand sequences of characters and words. The tools word2vec and doc2vec use neural networks to learn vector representations that encode the linguistic context of words and documents (14, 15). These embeddings attempt to capture word/document "meaning" and are much lower dimensional than the original input space. Similar concepts have recently been applied to learn embedded representations of amino acid sequences (16). Each amino acid sequence is treated as a document and the sequence is broken up into k-mers of constant length to represent words. These k-mers, along with their corresponding protein sequences, are used to predict average k-mers and infer representations or 'protein embeddings'. These protein embeddings are then used to model specific properties such as thermostability. This method lowers the dimensionality of the protein sequence representation because only a subset of k-mers is required to represent the entire protein sequence.

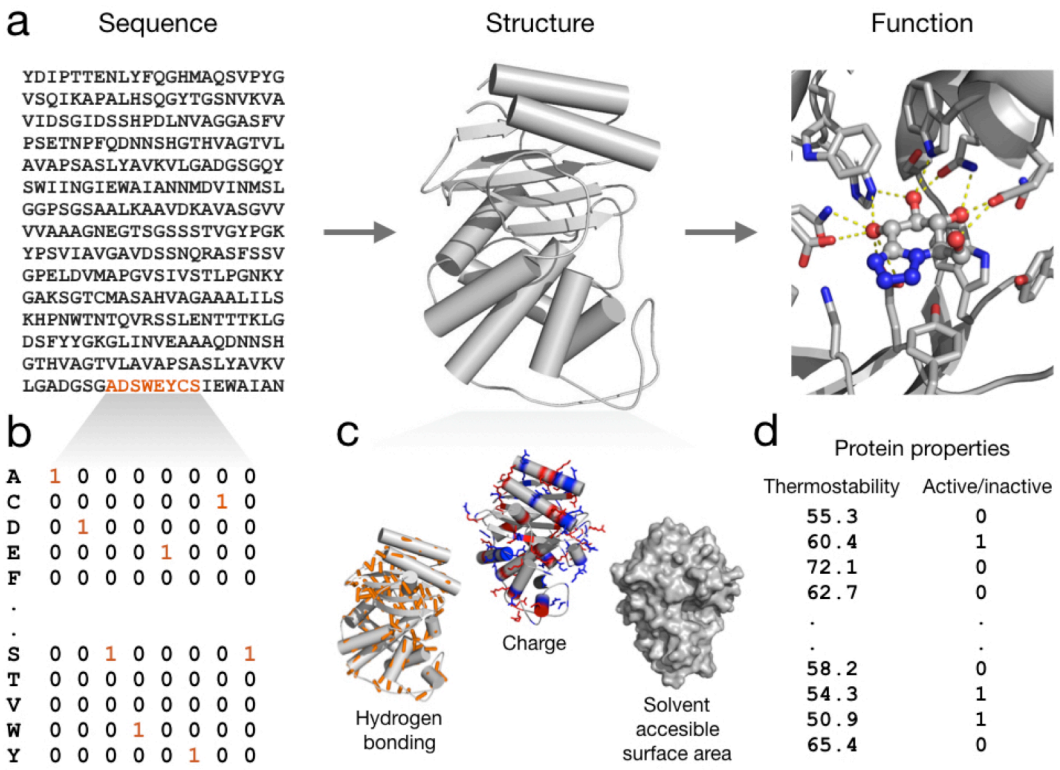

Figure 2: Sequence, structure, and function representations. (a) A protein's sequence folds into a three-dimensional structure, and this structure determines its function and properties. (b) Protein sequences can be represented using a one-hot encoding scheme that assigns 20 amino acid bits to each residue position. A bit is assigned a value of "1" if the protein has the corresponding amino acid at a particular residue position. (c) Structure-based representations use modeled protein structures to extract key physiochemical properties such as hydrogen bonds, total charge, or molecular surface areas. (d) Protein functions can be continuous properties such as thermostability or catalytic efficiency, or discrete properties such as active/inactive. Discrete properties can be represented using a binary (0 or 1) encoding.

Representing protein structures

The properties of proteins depend on sequence through their structure, therefore structure-based representations provide a more direct link to function. Experimentally determining a protein's three-dimensional structure (via crystallography, NMR, CryoEM) is significantly more challenging and time consuming than determining sequence or function. Therefore, most sequence-function data sets do not contain experimentally determined protein structures. Instead, this missing structural information can be approximated by taking advantage of the extreme conservation of structures within a family. Homologous proteins with as low as 20% sequence identity still have practically identical three-dimensional structures (17).

A protein's overall fold can be represented by specifying which residues are "contacting" in the three-dimensional structure. These contacting residues could be defined as any pair of residues that has an atom within five angstroms. Other contact definitions could include different distance cutoffs, Cα-Cα distances, or Cβ-Cβ distances. A protein's contact map specifies all pairs of contacting residues and provides a coarse-grained description of the protein's overall fold. Importantly, contact maps are highly conserved within a protein family, and therefore any two evolutionarily related proteins have practically identical contact maps. If we assume a fixed contact map for a protein family, structural information can be represented using a one-hot

encoding scheme similar to sequence encoding described above. Each pair of contacting residues can take on one of 400 (20^2) possible amino acid combinations, which can be one-hot encoded using 400 bits. Therefore, the structure of a protein with *c* contacts can be represented with 400*c* bits. In contrast to sequence-based representations, this contact-based representation can capture pairwise interactions between residues. However, this increased flexibility comes at the cost of significantly higher dimensionality.

Three-dimensional protein structures can also be predicted using molecular modeling and simulation software. Most protein sequence-function data sets can take advantage of homology modeling approaches that start with a closely related template structure, mutate differing residues to the target sequence, and run minimization methods to relax the structure into a local energy minimum. State-of-the-art homology modeling methods can reliably predict protein structures with less than 2 angstrom atomic RMSD (18). These predicted structures can be analyzed to extract key physiochemical properties such as surface areas, solvent exposure, and physical interactions (**Fig 2**). This approach was recently applied to model the kinetic properties of β-glucosidase point mutants (19). The substrate was docked into β-glucosidase homology models, and this enzyme-substrate interaction was used to extract 59 physical features such as interface energy, number of intermolecular hydrogen bonds, and change in solvent accessible surface area. A simple linear regression model could relate these physical features to β-glucosidase turnover number, Michaelis constant, and catalytic efficiency. Physics-based representations tend to be lower dimensional than the sequence and contact encodings described above. They may also have good generalization within a protein family or even across protein families because they are based on fundamental biophysical principles. However, these representations will always be limited by the resolution of structure prediction methods, and therefore may be restricted to small changes in protein sequence space (~1-2 amino acid substitutions).

Learning the sequence-function mapping from data

Advanced pattern recognition and machine learning techniques can be used to automatically identify key relationships between protein sequence, structure, and function. These tools are used for two primary tasks: supervised learning and unsupervised learning. Supervised methods, such as regression and classification, attempt to learn the mapping between a set of input variables and output variables. The term "supervised learning" arises because the algorithms are given examples of input-output mapping to guide the learning process. In contrast, unsupervised methods are not given information about the output variable, but instead try to learn relationships between the various input variables.

Supervised learning (Regression/Classification)

Regression is a supervised learning technique that is used to model and predict continuous properties. Continuous protein properties could include thermostability, binding affinity, or catalytic efficiency. Regression methods span from simple linear models to advanced nonlinear models such as neural networks.

Linear regression is the simplest regression technique and applies fixed weights to each input variable. A linear model is described by the following equation:

$$y = X\beta + \epsilon,$$

where *y* is a vector of continuous output variables, *X* is a matrix of sequence/structure features (one protein variant per row), β is the weight vector, and ϵ is the model error. The model parameters (β) can be estimated by minimizing the sum of the squared error. This least-squares parameter estimate has an analytical solution:

$$\hat{\beta} = (X^T X)^{-1}(X^T y)$$

Here, $\hat{\beta}$ corresponds to an estimate of the true β. $\hat{\beta}$ can then be applied to new proteins to predict their properties:

$$\hat{y} = X_{new}\hat{\beta}$$

Linear regression provides a simple framework for relating sequence/structure to function, and predicting the properties of previously uncharacterized proteins.

Linear regression has been used to model chimeric cytochrome P450 thermostability (20). A library of chimeric P450s was generated by shuffling sequence elements from three related bacterial P450s (21). The thermostability of 184 randomly chosen chimeric P450s was determined, and a linear regression model was used to relate sequence to thermostability. Each chimeric protein's sequence was one-hot encoded by specifying which sequence elements were present. This encoding scheme is similar to the sequence-based one-hot encoding described above, but sequence "blocks" are used rather than individual amino acids. This simple regression model revealed a strong correlation between the predicted and observed thermostability (**Fig 3**). The model was applied to predict the thermostabilities of all 6,351 possible sequences in the chimeric P450 library, and the most stable predicted sequences were validated experimentally.

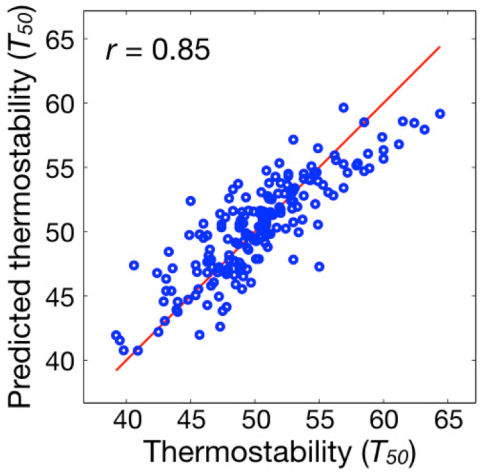

Figure 3: A linear regression model for cytochrome P450 thermostability. This model relates sequence blocks of chimeric P450s to their thermostability values. The plot shows the model's cross-validated predictions for 184 chimeric P450s.

Supervised learning methods, including linear regression, are highly susceptible to overfitting data, especially when the number of model parameters exceeds the number of observations. Overfitting occurs when the model fits spurious correlations or noise in the data, and not the true underlying signal. An overfit model will display very small error on the training data, but large prediction error on new data points. All statistical models must be evaluated for overfitting and their ability to generalize to new, unseen data points. One method for model validation involves training the model on some fraction of the data and using the remainder to evaluate the model's predictive ability. For example, one could train a model on 60% of the data and test the model on the remaining 40%. This holdout method is simple to implement, but also throws out valuable information because the model is not learning from the entire data set. Cross-validation is another method for model evaluation that more effectively utilizes the available data. Cross-validation is similar to the holdout method, but rotates through multiple training set-test set combinations. For example, ten-fold cross-validation breaks the data into ten subsets; a model is trained on nine of these subsets and used to predict the tenth subset. This process is repeated over all ten data folds (i.e. testing on all ten subsets) and the results

are averaged. Cross-validation allows all data points to be used in model training and evaluation.

Overfitting can be reduced using regularization methods that favor simpler models. Regularized parameter estimation involves minimizing the model's squared error in addition to the magnitude of the model parameters. This can be achieved by including a penalty term on the norm of the parameter vector:

$$\min_{\beta}(X\beta - y)^2 + \lambda\|\beta\|_n$$

Here, the first term corresponds to the model's squared error, the second term is the magnitude of the model parameters, and λ tunes the relative influence of these two terms. *n* determines the type of vector norm and is typically equal to 0, 1, or 2. L0 regularization (n=0) penalizes the total number of non-zero parameters in the model, L1 regularization (n=1) penalizes the sum of the parameter absolute values, and L2 regularization (n=2) penalizes the sum of the squared parameters. This minimization problem can be solved analytically if n=2 or using convex optimization if n=1. The hyperparameter λ can be determined using cross-validation. Combinations of these penalties can also be used, such as elastic net regression, which utilizes both L1 and L2 norms.

While regression methods model continuous properties, classification methods are used to model discrete protein properties such as folded/unfolded or active/inactive. Classifiers are important for modeling data generated by high-throughput methods such as deep mutational scanning because these methods often bin proteins into broad functional classes. Logistic regression is simple classification method that transforms a linear model through the logistic (sigmoid) function to produce binary outputs. Note: the name "logistic regression" is a misnomer because it actually performs classification rather than regression. Logistic regression parameters can be identified using iterative methods or convex optimization. Like the regression models discussed above, classification models can be evaluated using cross-validation and regularization can be used to prevent overfitting.

Logistic regression was recently used to refine molecular energy functions for designing de novo miniproteins (22). Thousands miniproteins were designed using Rosetta protein design software, and these designs were screened for folding using a high-throughput yeast display assay. Each protein's structure was modeled and used to generate physical input features such as number of H-bonds, Lennard-Jones energies, and net charge. Logistic regression was then used to map these physical features to whether a design was successful or unsuccessful. The authors found a protein's buried nonpolar surface area was a dominant factor in determining design success. The logistic regression model was used to rank designs and drastically improved the rate of successful designs.

Kernel methods are another modeling approach that is widely used in machine learning and bioinformatics. In contrast to the parametric regression/classification methods described above, kernel methods do not input feature vectors, but instead define a similarity function to compare pairs of data points. This similarity function could be as simple as an inner product between feature vectors, or they can represent more complex, potentially infinite dimensional, relationships between data points (23). This flexibility allows them to learn from unstructured objects such as biological systems. Popular kernel methods include Support Vector Machines (SVMs) and Gaussian Process (GP) regression/classification.

Gaussian processes use kernel functions to define a prior probability distribution over a function space. This allows predictions of both the function mean and its confidence intervals. Gaussian processes were used to model cytochrome P450s (24). A structure-based kernel function was developed to define structural similarity between pairs of proteins. GP regression using this kernel function explained 30% more of the variation in P450 thermostability in comparison to linear regression and sequence-based kernels. The structure-based kernel was also used to model enzyme activity and binding affinity for several P450 substrates.

Unsupervised/semisupervised learning

Unlike supervised learning, where the data is labeled or categorized, in unsupervised learning there are no labels associated with each data point. Unsupervised learning can be used to find patterns such as clusters or correlations within data. Examples of unsupervised methods include principal component analysis (PCA) and clustering. The main drawback of unsupervised techniques is that the outputs are unknown, i.e. there is no mapping to protein function. However, these techniques still provide valuable information about proteins because of the massive amount of protein sequence data that is currently available.

Unsupervised methods can be used to identify patterns in multiple sequence alignments (MSAs) of evolutionarily related proteins. Statistical coupling analysis (SCA) analyzes residue coevolution by performing principal component analysis on a protein family's MSA (25). The dominant principle components consist of positions that coevolve and can reveal networks of spatially connected amino acids called protein sectors (**Fig 4**). Protein sectors have been demonstrated to play roles in protein dynamics and allostery and may represent functional modules (26, 27). EVmutation is another unsupervised method that models natural sequence variation and simultaneously considers epistasis (non-independence of mutational effects) (28). Although EVmutation is only parameterized on an MSA (i.e. it is unsupervised), it is capable of predicting the functional effects of amino acid substitutions and residue interdependencies.

Semisupervised methods learn from data sets that contain both unlabeled and labeled data points. Semisupervised approaches can be used in protein engineering to transfer knowledge across protein families. A semisupervised approach was recently developed that trained an unsupervised embedding model (doc2vec) on a large protein sequence database (16). These embeddings were then used as the inputs for supervised Gaussian process regression. This approach was used to model channelrhodopsin membrane localization, P450 thermostability, and epoxide hydrolase enantioselectivity.

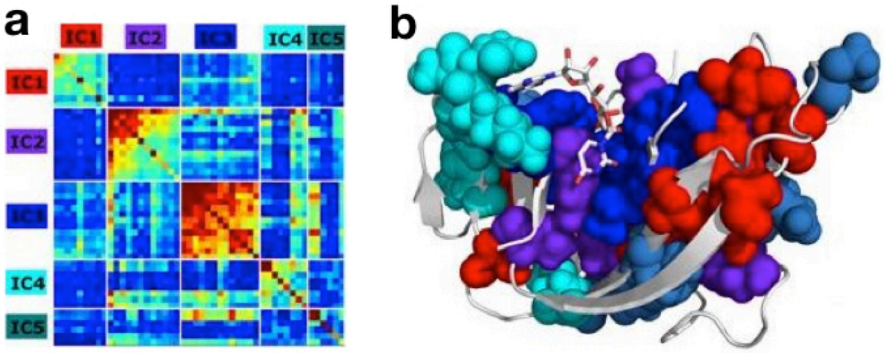

Figure 4: Unsupervised learning from protein sequences. (A) Statistical coupling analysis of the RNase superfamily reveals five independent components (ICs) that correspond to groups of coevolving residues (B) These five ICs form contiguous "sectors" in the three-dimensional protein structure. Figure was adapted from (27).

Applying statistical models to engineer proteins

Statistical modeling approaches provide unprecedented predictive accuracy for a wide variety of complex protein functions/properties. These models can be used to understand protein function and design new proteins. We discuss several protein engineering strategies that leverage the predictive power of statistical models.

The most straightforward data-driven protein engineering approach involves training a model on a data set and then extrapolating that model to design best predicted sequences. This method was applied to engineer thermostable fungal cellobiohydrolase class II (CBHII) cellulases (29). A panel of 33 chimeric CBHIIs was characterized for their thermal inactivation

half-lives at elevated temperatures. This data was used to train a linear regression model that related sequence blocks to thermal tolerance. This model was then used to design 18 chimeras that were predicted to have enhanced stability relative to the parent enzymes. Most of these designed CBHII chimeras could hydrolyze cellulose at higher temperatures than most stable parent. A key feature of this extrapolation-based design approach is a relatively small training set (<1% of possible chimeras) can be used to make predictions over a massive combinatorial sequence space. The CBHII regression model also pointed to a single sequence block that contributed over 8 °C of thermostability (30). Further analysis revealed that a single mutation in that block (C313S) was responsible for the elevated thermostability. This example highlights how statistical models can be used to uncover molecular mechanisms contributing to protein function.

It is important to consider the space of sequences that a statistical model can make valid predictions on. This prediction domain is highly dependent on the model's sequence/structure representation. For example, consider a model that uses one-hot encoding to represent protein sequences. This model can only learn the effect of amino acids that are observed in the training set, and therefore can only make predictions about sequences composed of combinations of these observed amino acids. Representations that include information about amino acid properties and/or protein structure can broaden a model's prediction domain. Representations that use three-dimensional structural models to extract key physiochemical properties have potential to generalize well within a protein family and even across protein families.

Statistical models can be incorporated into an iterative directed evolution framework. ProSAR uses a statistical model to guide the search for beneficial mutations (31). This model consists of a one-hot encoded sequence representation and a partial least squares linear regression model to relate sequence to function. A mutational library is screened, and the model classifies each amino acid substitution as deleterious, neutral, beneficial, or underdetermined (i.e. needing more information). Substitutions that are beneficial or underdetermined are combined with new substitutions in the next round, and this screen-and-learn process is repeated over multiple rounds. The ProSAR method was used to engineer bacterial halohydrin dehalogenases (HHDH) to perform a cyanation reaction important for the synthesis of the cholesterol-lowering drug Lipitor (31). 18 rounds of ProSAR yielded HHDH variants with over 35 mutations and increased the volumetric productivity of target reaction by ~4,000-fold. More recently, ProSAR-driven evolution was used to evolve ultra-stable carbonic anhydrase variants (107 ˚C thermostability at pH 10 in 4.2 M solvent) that enhanced the rate of CO_2 capture by 25-fold over the natural enzyme (32).

Statistical models can also be used in an active learning setting that iteratively explores protein sequence space. The goal of active learning in protein engineering is to identify optimal sequences while minimizing the total number of required experiments. This can be accomplished by iterating over multiple design-test-learn cycles (**Fig 5a**). At each design step in this cycle, the active learning algorithm must decide between (1) designing informative sequences that will improve the model or (2) designing optimal sequences predicted by the model. This "exploration-exploitation dilemma" is encountered in diverse application domains such as online advertising, robotic control, and clinical trials. Upper confidence bound (UCB) algorithms provide a principled framework for trading off between exploration and exploitation modes (33). UCB algorithms iteratively select the point with the largest upper confidence bound (predicted mean plus confidence interval) and therefore encourage sampling of points that are simultaneously optimized and uncertain (**Fig 5b**). A UCB search algorithm was combined with a Gaussian process regression model to optimize cytochrome P450 thermostability (24). Eight rounds of UCB optimization identified thermostable P450s that were more stable than variants made by rational design, recombination or directed evolution.

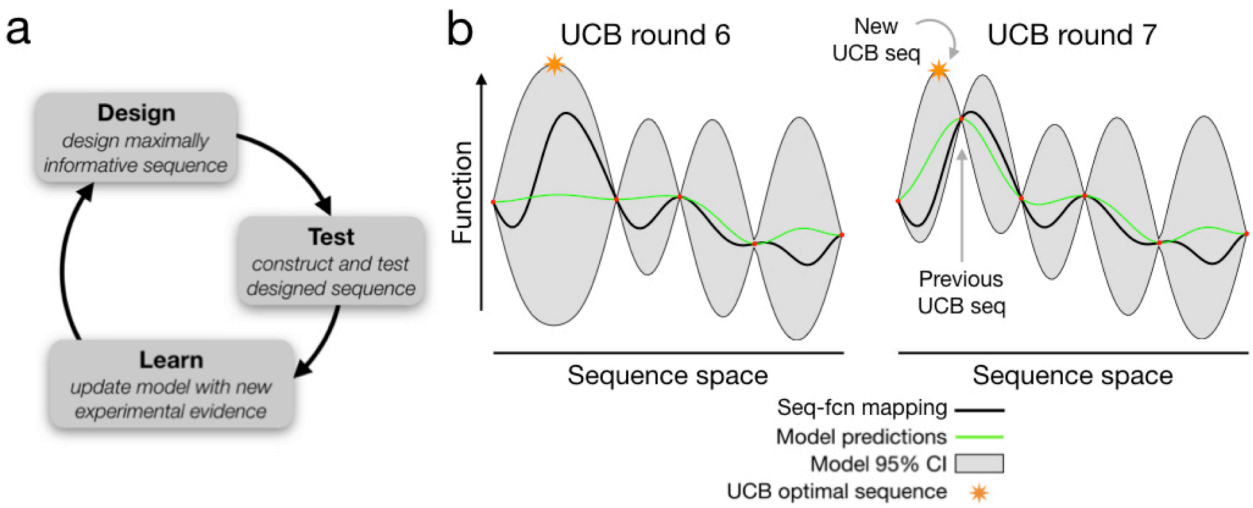

Figure 5: Active machine learning. (a) Active learning involves designing maximally informative sequences, experimentally characterizing these sequences, learning from the resulting data, and repeating this process over multiple iterations. (b) Upper-confidence bound (UCB) optimization involves iteratively selecting the sequence with the largest upper confidence bound (mean + confidence interval). The schematic illustrates sequence space in one dimension and the true mapping from sequence to function as a black line. Characterized sequences (small red dots) have accurate model predictions and small confidence intervals. The first panel shows five characterized sequences, which cause the model to propose one UCB optimal sequence (marked with a star). The second panel shows the results after this UCB optimal sequence is characterized—this causes a new UCB sequence to be proposed. This iterative process is guaranteed to efficiently converge to the optimal point.

Conclusions and future outlook

The protein sequence-structure-function mapping involves thousands of interacting atoms, a practically infinite number of dynamic conformational states, and physical processes that span multiple length and time scales. This mapping is extremely difficult to model from a physical perspective. In contrast, statistical methods are able to learn complex interrelationships directly from experimental data. This top-down understanding of complex systems allows discovery of new functional mechanisms and provides exceptional predictive accuracy.

This chapter provides an overview of emerging data-driven approaches to model and engineer proteins. We have described statistical representations of proteins, how these representations can be used to learn from data, and practical protein engineering applications of these models. As a relatively new field, there is still significant room for improving these methods, especially in the area of sequence and structure representations. Ideal representations would be sparse, but still have a broad prediction domain. These representations may integrate different sources of information (evolutionary, biochemical, and physical) into a single unified model. Advanced machine learning methods such as dictionary learning and deep learning attempt to learn new representations directly from data and could play an important role in protein modeling.

In addition to proteins, statistical approaches can be used to model genotype-phenotype relationships across all levels of biological organization. For example, linear regression was used to model product titers in a multi-enzyme biosynthetic pathway; this model was then used to optimize enzyme expression levels to maximize overall product production (34). Another example used compressed sensing methods to model a protein's DNA-binding specificity (35). Statistical methods have been widely used in genetics relate phenotypes to genetic loci using quantitative trait locus (QTL) mapping (36).

Data-driven approaches are transforming every field of science and engineering. This revolution has been triggered by the confluence of advances in data generation, data access, and data analysis/interpretation. Advanced experimental technologies are allowing us to analyze biological systems on an unprecedented scale and resolution. The resulting data is also becoming readily accessible through large, public biological databases and repositories. At the same time, there have been tremendous advances in artificial intelligence and pattern recognition. Widespread interest in machine learning has also driven improvements in software packages such as the Scikit-learn and Keras deep learning Python libraries. Data-driven approaches leverage the continuously expanding sea of data and will play an increasingly important role in biological discovery and engineering.

References

1. Lazaridis T, Karplus M (2000) Effective energy functions for protein structure prediction. *Curr Opin Struct Biol* 10(2):139–145.
2. Li Z, Yang Y, Zhan J, Dai L, Zhou Y (2013) Energy functions in de novo protein design: current challenges and future prospects. *Annu Rev Biophys* 42:315–35.
3. Bradley P, Misura KM, Baker D (2005) Toward high-resolution de novo structure prediction for small proteins. *Science (80-)* 309(5742):1868–1871.
4. Baker D (2010) An exciting but challenging road ahead for computational enzyme design. *Protein Sci* 19(10):1817–1819.
5. Kosuri S, Church GM (2014) Large-scale de novo DNA synthesis: Technologies and applications. *Nat Methods* 11(5):499–507.
6. Apweiler R, Bairoch A, Wu CH, Barker WC, Boeckmann B, Ferro S, Gasteiger E, Huang H, Lopez R, Magrane M, Martin MJ, Natale D a, O'Donovan C, Redaschi N, Yeh L-SL (2017) UniProt: the universal protein knowledgebase. *Nucleic Acids Res* 45:D158–D169.
7. Rose PW, Prlic A, Altunkaya A, Bi C, Bradley AR, Christie CH, Di Costanzo L, Duarte JM, Dutta S, Feng Z, Green RK, Goodsell DS, Hudson B, Kalro T, Lowe R, Peisach E, Randle C, Rose AS, Shao C, Tao Y-P, Valasatava Y, Voigt M, Westbrook JD, Woo J, Yang H, Young JY, Zardecki C, Berman HB, Burley SK (2017) The RCSB protein data bank: integrative view of protein, gene and 3D structural information. *Nucleic Acids Res* 45:D271–D281.
8. Wang CY, Chang PM, Ary ML, Allen BD, Chica RA, Mayo SL, Olafson BD (2018) ProtaBank: A repository for protein design and engineering data. *Protein Sci*. doi:10.1002/pro.3406.
9. Hietpas RT, Jensen JD, Bolon DNA (2011) Experimental illumination of a fitness landscape. *Proc Natl Acad Sci U S A* 108(19):7896–7901.
10. Araya CL, Fowler DM (2011) Deep mutational scanning: Assessing protein function on a massive scale. *Trends Biotechnol* 29(9):435–442.
11. Romero PA, Tran TM, Abate AR (2015) Dissecting enzyme function with microfluidic-based deep mutational scanning. *Proc Natl Acad Sci U S A* 112(23):7159–7164.
12. Whitehead TA, Chevalier A, Song Y, Dreyfus C, Fleishman SJ, De Mattos C, Myers CA, Kamisetty H, Blair P, Wilson IA, Baker D (2012) Optimization of affinity, specificity and function of designed influenza inhibitors using deep sequencing. *Nat Biotechnol* 30(May):1–9.
13. Bloom JD (2014) An experimentally determined evolutionary model dramatically improves phylogenetic fit. *Mol Biol Evol* 31(8):1956–1978.
14. Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient Estimation of Word Representations in Vector Space. *CoRR* abs/1301.3. doi:10.1162/153244303322533223.
15. Le Q V, Mikolov T (2014) Distributed Representations of Sentences and Documents. *CoRR* abs/1405.4. doi:10.1145/2740908.2742760.
16. Yang KK, Wu Z, Bedbrook CN, Arnold FH (2018) Learned protein embeddings for

machine learning. *Bioinformatics* 34(15):2642–2648.
17. Chothia C, Lesk AM (1986) The relation between the divergence of sequence and structure in proteins. *EMBO J* 5(4):823–826.
18. Misura KM, Chivian D, Rohl CA, Kim DE, Baker D (2006) Physically realistic homology models built with ROSETTA can be more accurate than their templates. *Proc Natl Acad Sci* 103(14):5361–5366.
19. Carlin DA, Caster RW, Wang X, Betzenderfer SA, Chen CX, Duong VM, Ryklansky C V., Alpekin A, Beaumont N, Kapoor H, Kim N, Mohabbot H, Pang B, Teel R, Whithaus L, Tagkopoulos I, Siegel JB (2016) Kinetic characterization of 100 glycoside hydrolase mutants enables the discovery of structural features correlated with kinetic constants. *PLoS One*. doi:10.1371/journal.pone.0147596.
20. Li Y, Drummond DA, Sawayama AM, Snow CD, Bloom JD, Arnold FH (2007) A diverse family of thermostable cytochrome P450s created by recombination of stabilizing fragments. *Nat Biotechnol* 25(9):1051–1056.
21. Otey CR, Landwehr M, Endelman JB, Hiraga K, Bloom JD, Arnold FH (2006) Structure-Guided Recombination Creates an Artificial Family of Cytochromes P450. *PLoS Biol* 4(5):e112.
22. Rocklin GJ, Chidyausiku TM, Goreshnik I, Ford A, Houliston S, Lemak A, Carter L, Ravichandran R, Mulligan VK, Chevalier A, Arrowsmith CH, Baker D (2017) Global analysis of protein folding using massively parallel design, synthesis, and testing. *Science (80-)* 357(6347):168–175.
23. Rasmussen CE, Williams C (2006) *Gaussian Processes for Machine Learning* (MIT Press, Cambridge, MA).
24. Romero PA, Krause A, Arnold FH (2013) Navigating the protein fitness landscape with Gaussian processes. *Proc Natl Acad Sci U S A* 110(3):E193–E201.
25. Lockless SW, Ranganathan R (1999) Evolutionarily conserved pathways of energetic connectivity in protein families. *Science (80-)* 286(5438):295–299.
26. Halabi N, Rivoire O, Leibler S, Ranganathan R (2009) Protein sectors: evolutionary units of three-dimensional structure. *Cell* 138(4):774–786.
27. Narayanan C, Gagné D, Reynolds KA, Doucet N (2017) Conserved amino acid networks modulate discrete functional properties in an enzyme superfamily. *Sci Rep*. doi:10.1038/s41598-017-03298-4.
28. Hopf TA, Ingraham JB, Poelwijk FJ, Schärfe CPI, Springer M, Sander C, Marks DS (2017) Mutation effects predicted from sequence co-variation. *Nat Biotechnol* 35(2). doi:10.1038/nbt.3769.
29. Heinzelman P, Snow CD, Wu I, Nguyen C, Villalobos A, Govindarajan S, Minshull J, Arnold FH (2009) A family of thermostable fungal cellulases created by structure-guided recombination. *Proc Natl Acad Sci U S A* 106(14):5610–5615.
30. Heinzelman P, Snow CD, Smith MA, Yu X, Kannan A, Boulware K, Villalobos A, Govindarajan S, Minshull J, Arnold FH (2009) SCHEMA recombination of a fungal cellulase uncovers a single mutation that contributes markedly to stability. *J Biol Chem* 284(39):26229–26233.
31. Fox RJ, Davis SC, Mundorff EC, Newman LM, Gavrilovic V, Ma SK, Chung LM, Ching C, Tam S, Muley S, Grate J, Gruber J, Whitman JC, Sheldon RA, Huisman GW (2007) Improving catalytic function by ProSAR-driven enzyme evolution. *Nat Biotechnol* 25(3):338–344.
32. Alvizo O, Nguyen LJ, Savile CK, Bresson JA, Lakhapatri SL, Solis EOP, Fox RJ, Broering JM, Benoit MR, Zimmerman SA, Novick SJ, Liang J, Lalonde JJ (2014) Directed evolution of an ultrastable carbonic anhydrase for highly efficient carbon capture from flue gas. *Proc Natl Acad Sci* 111(46):16436–16441.
33. Auer P (2002) Using Confidence Bounds for Exploitation-Exploration Trade-offs. *J Mach*

Learn Res 3(3):397–422.
34. Lee ME, Aswani A, Han AS, Tomlin CJ, Dueber JE (2013) Expression-level optimization of a multi-enzyme pathway in the absence of a high-throughput assay. *Nucleic Acids Res* 41(22):10668–10678.
35. AlQuraishi M, McAdams HH (2011) Direct inference of protein-DNA interactions using compressed sensing methods. *Proc Natl Acad Sci* 108(36):14819–14824.
36. Doerge RW (2002) Mapping and analysis of quantitative trait loci in experimental populations. *Nat Rev Genet*. doi:10.1038/nrg703.