Machine learning in protein engineering

Kevin K. Yang Zachary Wu Frances H. Arnold

Presented by: Marius Păltănea, Ph.D.

March 19, 2019

Introduction

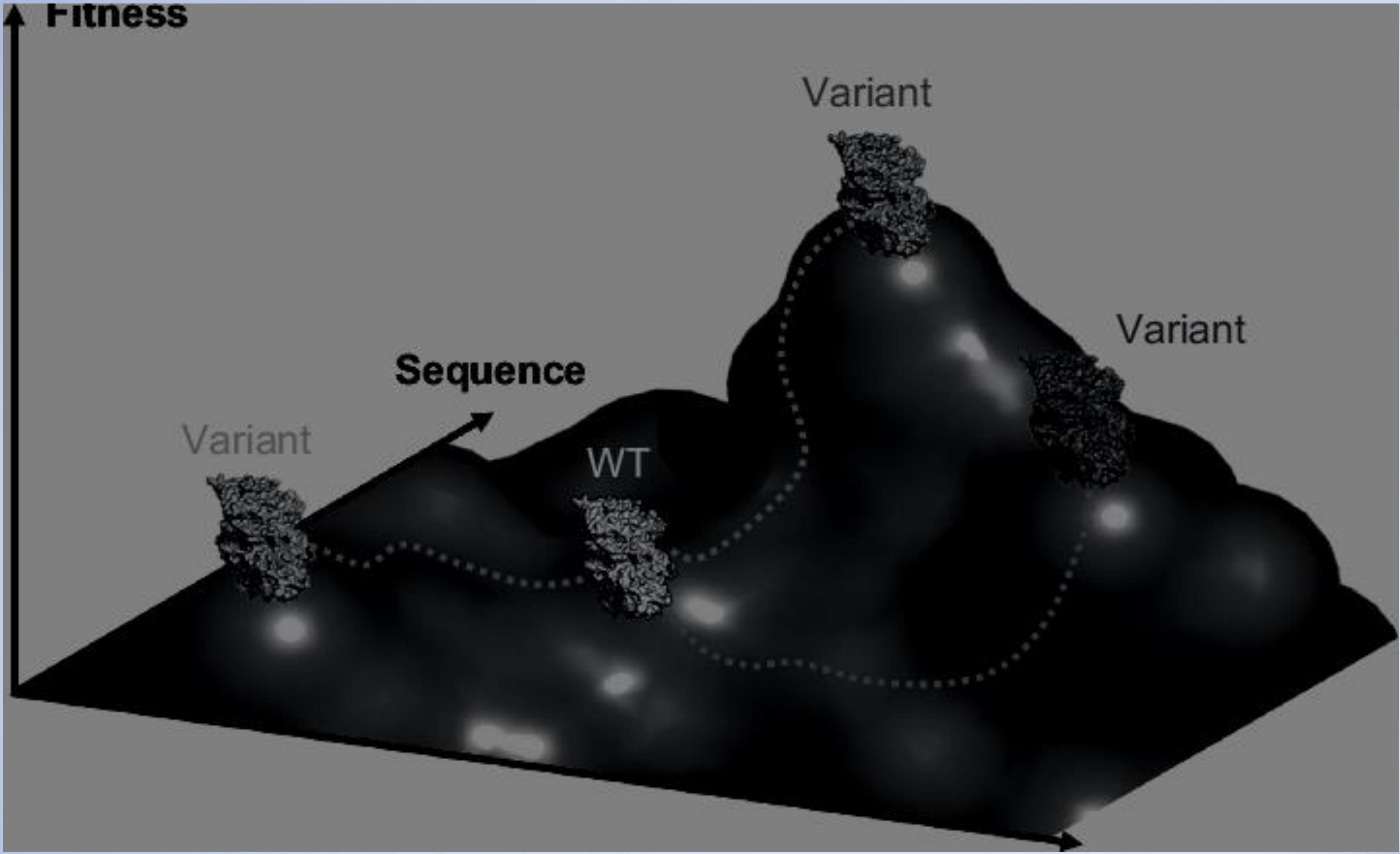
Protein engineering – design or discover proteins whose properties, useful for technological, scientific, or medical applications, have not been needed or optimized in nature.

fitness landscape – protein's performance in terms of expression level, catalytic activity, binding or other properties of interest to the protein engineer

Introduction

Space of possible protein sequences is too large to be searched exhaustively naturally, in the laboratory, or computationally → there is no known polynomial-time method for searching this space (e.g for a 100 amino-acid protein, there are 20^100 possible sequences)

Functional proteins are extremely scarce in this space, even more so as the threshold level of fitness is increased


Fitness
Variant
Variant
Sequence
Variant
WT

Rational design

- uses physics-based models (which contain an atomic structural representation of a protein) to guide the search for improved sequences

- useful when a single stable structure dictates function

- design an idealized active site for the desired reaction, matching the active site residues to stable backbones, and then apply molecular dynamics simulations (extremely costly in terms of computational resources)

Directed evolution

- Inspired by natural evolution - accumulates beneficial mutations in an iterative protocol of mutation and selection

- first step:sequence diversification using random mutagenesis, site-saturation mutagenesis, or recombination

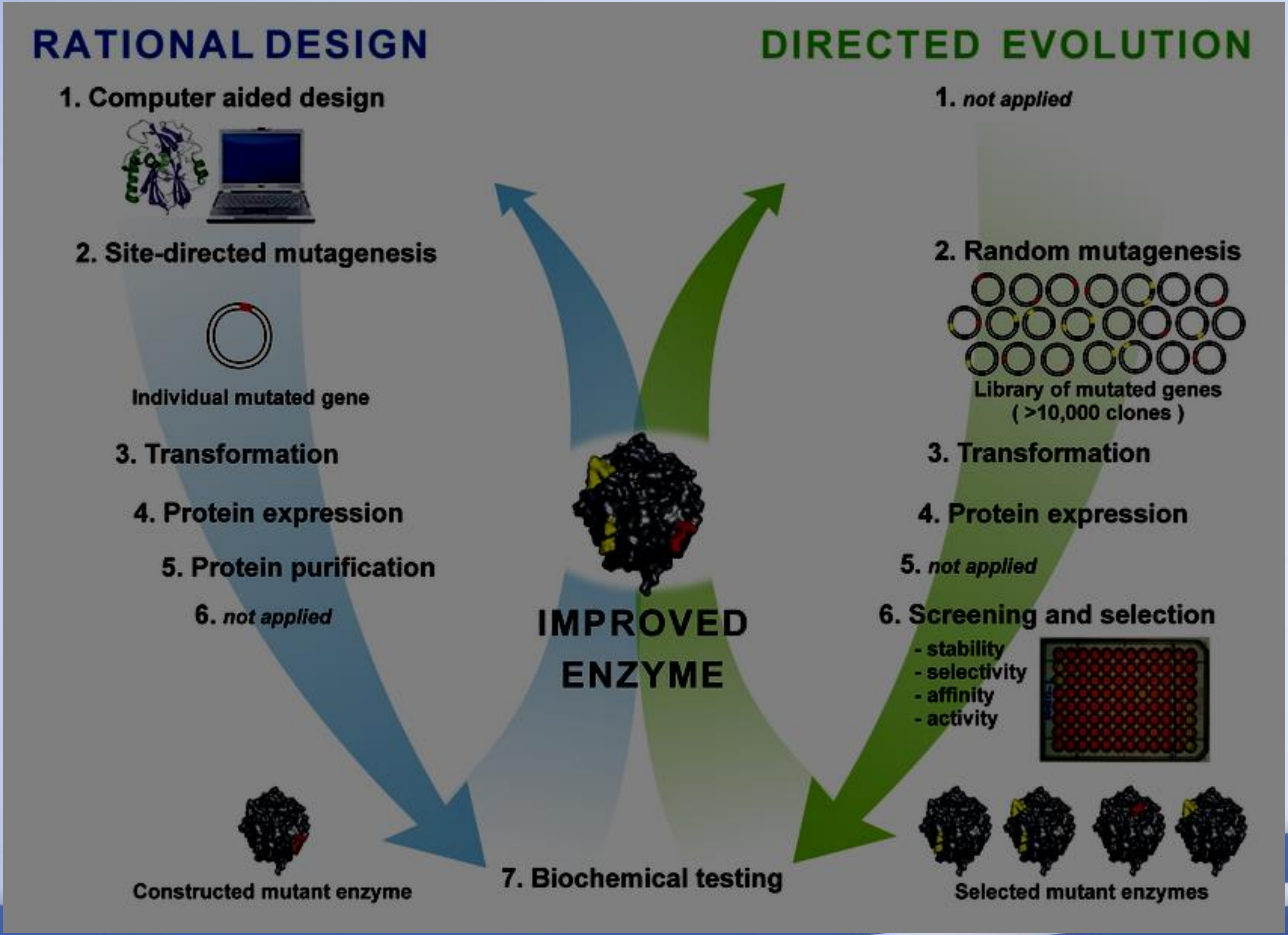
- second step: screening or selection to identify variants with improved properties for the next round of diversification

Directed evolution drawbacks

- even the most high-throughput screening / selection only sample an insignificant fraction of the sequences

- recombination methods may allow for bigger jumps in sequence space while retaining function, but sequences designed using recombination are restricted to exploring combinations of previously-explored mutations

- directed evolution requires at least one minimally-functional parent



RATIONAL DESIGN

1. Computer aided design
2. Site-directed mutagenesis

Individual mutated gene

3. Transformation
4. Protein expression
5. Protein purification
6. *not applied*

Constructed mutant enzyme

DIRECTED EVOLUTION

1. *not applied*
2. Random mutagenesis

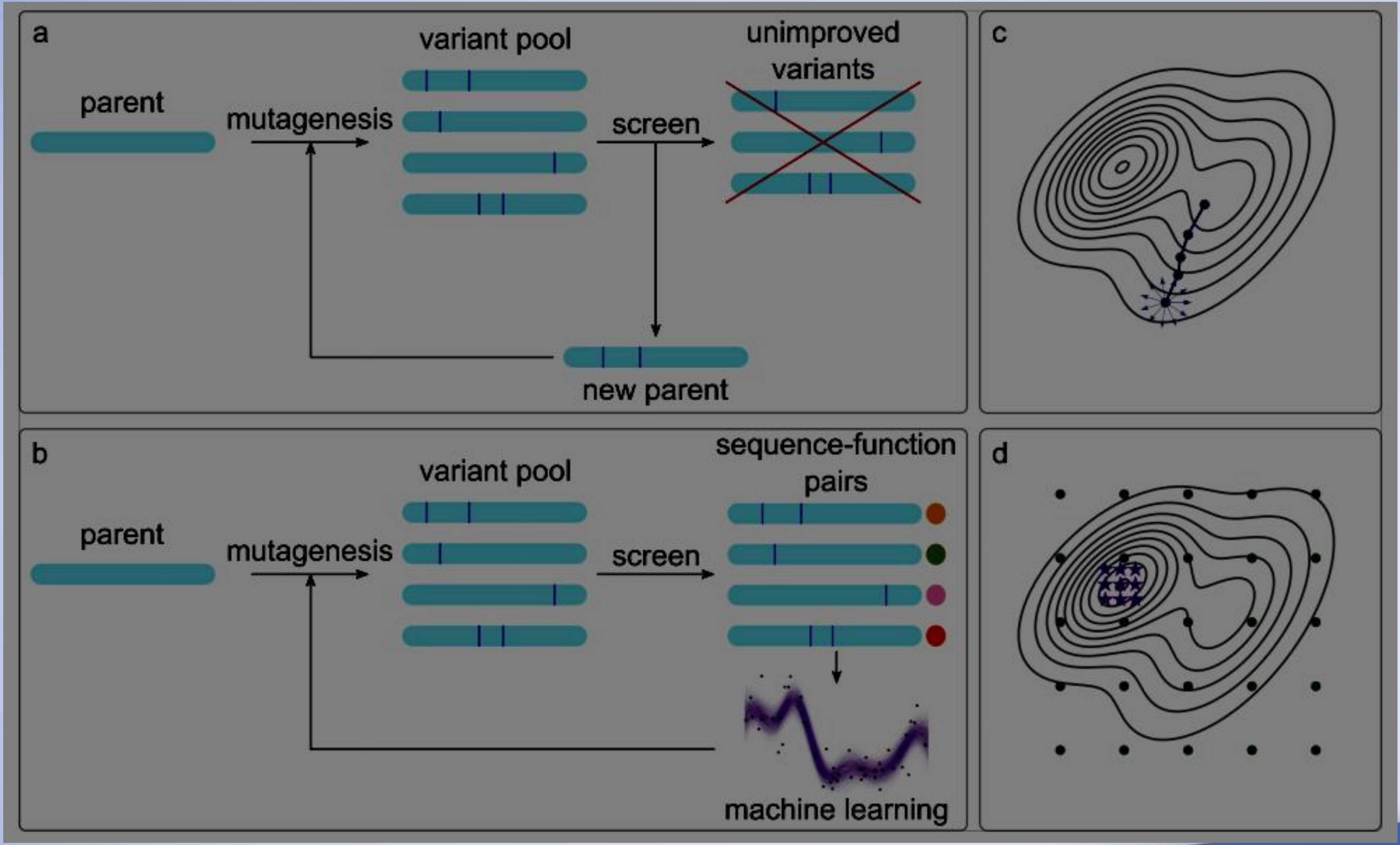
Library of mutated genes
(>10,000 clones)

3. Transformation
4. Protein expression
5. *not applied*
6. Screening and selection
   - stability
   - selectivity
   - affinity
   - activity

Selected mutant enzymes

IMPROVED ENZYME

7. Biochemical testing


a
variant pool
unimproved variants
parent
mutagenesis
screen
new parent
c
b
variant pool
sequence-function pairs
parent
mutagenesis
screen
machine learning
d

Protein function datasets

1) Databases;

- Protein sequences: *UniProt*

- Protein structure: *Protein Data Bank*

- Protein sequence-function: *ProTherm*

2) Datasets derived from protein engineering experiments – small and focused on high-performing variants,bisased by the intent of the study

3) Datasets of natural variants – large,with exponents from many families of proteins, biased by evolution itself

Vector representations

Protein **sequence**: a string of length *L* where each position is chosen from an alphabet of size 20

One-hot encoding:

- each of the *L* positions as 19 zeros and one 1
- single mutations: 18 zeroes, -1 as the original amino-acid, 1 as the new mutated amino-acid
- sparse, memory-inefficient, high-dimensional, provide no notion of similarity
- require that all sequence variants of interest are aligned

Vector representations

Encoding by **physical properties**:

- by representing each amino acid with a collection of physical properties, such as its volume, charge or hydrophobicity, and each protein with a combination of those properties
- by predicted secondary structures
- difficult to know what properties (from the large number of available properties) will be predictive for a particular task

Vector representations

Only a tiny fraction of the vast number of known protein sequences are labeled with measured properties

Unlabeled sequences contain information about the frequency and patterns of amino-acids selected by evolution to compose proteins

Embedded representations: BLOSUM / AAIndex substitution matrix – based on relative amino-acid frequencies

Models for protein data

Supervised learning

- the training data consist of inputs and their associated output values (labels)
- learn a mapping from input space to output space that enables to accurately predict outputs from new inputs
- regression - predict real-valued outputs
- classification – predict class membership

Models for protein data

Linear models - apply a linear transformation of the input (the amino acid at each position, the presence or absence of a mutation)

Classification and regression trees - often encountered in protein engineering experiments of small datasets (<10^4 training examples) – successfully used to predict thermostability

k-nearest-neighbor - quality of the predictions can be affected by setting the neighborhood size *k* as well as the distance metric used to identify the nearest neighbors → are not commonly applied to protein datasets

Models for protein data

Kernel methods – employ a kernel function, which calculates similarities between pairs of inputs, to implicitly project the input features into a high-dimensional feature space without explicitly calculating the coordinates in this new space

Gaussian process – rigorously capture uncertainty, and can provide principled ways to guide experimental design in optimizing protein properties, unsuitable for large (> 10^3) datasets – used to predict thermostability, fluorescence, membrane localization

Models for protein data

Neural networks - multiple linear layers connected by non-linear activation functions, allowing to extract high-level features from structured inputs

Well-suited for tasks with large labeled datasets, with examples from many protein families:

protein-nucleic acid binding, binding site prediction, thermostability, secondary structure, solubility

Model training and evaluation

Hyperparameter - configuration that is external to the model and whose value cannot be estimated from data; often specified by the practitioner (e.g type of kernel, learning rate)

Test set – 20% of the data to be set aside until the absolute end for model evaluation

The training - used to learn model parameters

Validation set – used to choose between models with different hyperparameters

Model interpretation

Learned weights in a linear model indicate which mutations or sequence blocks are beneficial for a function of interest

Splits in a decision tree naturally map to human-interpretable information about the features used to make predictions

For non-parametric models – local or global linear approximations

Convolution weights indicate the relative importance of sequence motifs to the property predicted (a convolution layer scans across a sequence looking for the presence of a learned motif)

ML as a guide to directed evolution

ML methods can use the information discarded by directed evolution in order to expedite evolution by intelligently selecting new variants to screen → careful choice of mutations to test decreases the screening burden and improves outcomes

A ML-guided evolution strategy requires a method for generating diversity, a screen to evaluate diversity, a ML model that learns the relationship between sequence and function, and a method to use the model to choose mutations for the next round of evolution

Generating diversity

Random mutations throughout the length of the protein by error-prone polymerase chain reaction (PCR) → linear models can be used to classify mutations as beneficial, detrimental or neutral

Site-saturation mutagenesis randomizes selected locations within the sequence determined to be most responsible for function or most likely to tolerate mutation

Recombination methods make larger jumps in sequence space while preserving a large fraction of functional sequences by only considering diversity from within a set of related proteins

Future directions

Biggest obstacle to future applications of machine learning to protein engineering is a **lack of high-quality data** → can be augmented with computationally-generated examples

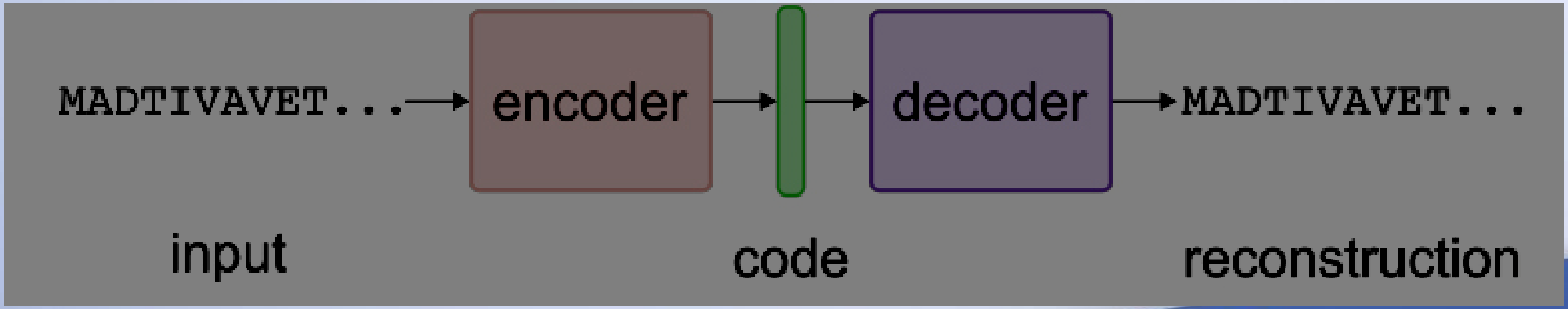
Deep mutational scanning - combines a high-throughput screen with next-generation sequencing to generate large sequence-function datasets → test beds for ML methods that learn to predict the effects of small numbers of mutations

Large quantities of **unlabeled sequence data** may enable ML models to generate artificial protein diversity leading to novel protein functions.

Future directions

Generative models learn to generate examples that are similar to those in the training set but are not found in the training set.

Autoencoder – consists of an encoder and a decoder model. The encoder converts the input to a lowdimensional vector (code). The decoder reconstructs the input from this code. Typically, the encoder and decoder are both neural networks


MADTIVAVET...
encoder
decoder
MADTIVAVET...
input
code
reconstruction