Exploring protein-protein interactions using high-throughput datasets and deep learning

Alyssa La Fleur

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington

2025

Reading Committee:

Georg Seelig, Chair

Jeffery Nivala

Pang Wei Koh

Program Authorized to Offer Degree:

Paul G. Allen Department of Computer Science and Engineering

University of Washington

ABSTRACT

Exploring protein-protein interactions using high-throughput datasets and deep learning

Alyssa Marie La Fleur

Chair of the Supervisory Committee:
Associate Professor Georg Seelig
Electrical and Computer Engineering
Computer Science and Engineering

Protein-protein interactions (PPIs) are fundamental to cellular function. Understanding which proteins interact—and how sequence variation alters these interactions—is essential for advancing therapeutic discovery and protein engineering. High-throughput sequencing technologies enable the large-scale measurement of PPIs, but the resulting datasets are complex and require error correction, modeling, and interpretation to yield meaningful insights. This thesis presents work across the process of designing, executing, and making use of high-throughput data, including (1) designing and modeling mutant protein libraries for large-scale PPI measurement, (2) developing PPI-specific sequencing analysis pipelines, (3) training models on limited structural features for PPI prediction for specific families, and (3) applying feature attribution techniques to interpret sequence-to-function models. Together, this work supports the continued development of experimental and computational tools to deepen our understanding of protein-protein interactions.

Acknowledgments

I want to thank my advisor Prof. Georg Seelig for the encouragement, advice, and mentorship he has provided over my years at the University of Washington in the Seelig Lab. I would also like to thank the members of my doctoral committee Jeffery Nivala, Pang Wei Koh, and my GSR Armita Nourmohammad.

To the members of the Seelig lab, both past and present: thank you all for being the finest collaborators, coworkers, and friends. I have learned so much from all of you and wish you continued success.

I would also like to thank my family (especially my parents), friends (especially my gaming group: Ashleigh, Judah, Kyle, and Nick), and pets (especially my dog, Maple) for their unwavering support and companionship.

(Additionally, thanks to Sebastian Castillo-Hair for this thesis template.)

Contents

List of Figures

Chapter 1: Introduction to PPIs and high-throughput assays

Protein-protein interaction (PPI) is a general term to describe a physical interaction where at least two proteins come together and make intentional contact to form a complex. Determining if proteins will interact – and how changes in protein sequence result in changes in interaction occurrence – is an active field of research, with applications from healthcare to synthetic biology. Our understanding of what controls interactions has exploded in the last few decades. Previously thought undruggable, misfunctioning PPIs have become drug targets: inhibitors have been developed to weaken unwanted interactions and molecular glues to strengthen interactions (1,2). Multiple approaches for engineering protein interactions and *de novo* interacting proteins exist, from physics-based methods like Rosetta (3,4) to structural-predictor based methods (5,6) have been developed. However, PPIs remain mysterious in many ways, with much work remaining to be done to be able to reliably predict when proteins will interact, and how to modify these interactions to our advantage.

1.1. PPI types and relevant concepts

1.1.1. PPI definitions and subtypes

The individual components of a PPI are known as protomers,. There are three main classifications of PPIs dependent on their composition, duration, and the stability of the protomers (7). One of the main distinctions is between oligomers which are comprised of identical subunits (homo-oligomers) versus those which have non-identical protomers (hetero-oligomers). A second distinction is if the complex is permanent or transient, with permanent complexes being those which are very stable and often considered irreversible, and transient being those which can come together then break apart. Moreover, transient interactions can exist on a scale: some are continuously coming together and coming apart, while others come together only when a significant structural change has occurred in a protomer/protomers, such as

the binding of a co-factor (7,8) Finally, oligomers can be divided by being obligate or non-obligate: when their individual components do not exist or function stably independently, they are obligate. Otherwise, they are considered non-obligate. Often, complexes which carry out a vital cellular function are obligate and permanent (7).

Besides these three common PPI groupings, there are also subtypes of PPIs based on what mediates their binding or characteristics of the protomers involved. Of particular importance to this thesis are PPIs involving domain-motif interactions, domain-domain interactions, and fuzzy interactions. Often, domain-motif interactions involve intrinsically disordered regions (IDRs). IDRs are regions of a protein which exist on a spectrum of disordered (that is, without a fixed secondary or tertiary structure) to ordered. Proteins with IDR regions involved in binding have disordered 'floppy' regions containing a short linear motif (SLiM) which become structured when bound by a domain capable of recognizing it (9). Domain-domain interactions are those mediated by two structured protein domains recognizing and binding one another (10). One way PPIs have been investigated across entire interaction networks previously is by identifying repetitive domain and motif elements by sequence similarity, and creating databases of domain-domain and domain-motif relationships (10–13). Unlike DDIs and SLiM-mediated interactions, fuzzy PPIs involve complexes which lack a single defined structure but instead may adopt multiple configurations or retain a degree of disorder even when bound. This typically occurs because one or more partners are intrinsically disordered proteins (IDPs), which retain a high amount of flexibility and disorder when bound. In general, it is thought that PPIs involving disorder and fuzziness are key to context dependent protein interactions, which is critical for vital cell functions like signaling cascades (14,15).

1.1.2. Epistasis and evolution of PPIs

The biological properties of proteins are determined by their physical properties, which in turn are determined by their amino acid sequence. The instructions for

this amino acid sequence are encoding in genes – changes in the genetic sequence can then affect the protein sequence, and the biological function of the protein. Mutations which result in a single amino acid change in protein sequence are referred to as missense mutations – though other changes in nucleotide sequence can occur which change the protein sequence, such as deletions or insertions of additional nucleotides. When multiple mutations occur, their cumulative effects can differ from the sum of their individual effects, in a phenomenon known as intramolecular epistasis (16,17).

In PPIs, there are multiple genes whose products interact with one another – so epistatic interactions can become even more complicated to disentangle. The prevalence of epistatic effects in proteins is debated, with some works claiming it is overrepresented due to the tendency to assign single sequences as the 'wild type' (18) and others claiming it is ubiquitous in proteins (17). Either way, awareness of potential epistatic effects is key when considering PPIs, as even if the effects of mutations in the individual protomers are known, their cumulative effects may result in much different PPI behaviors.

1.2. High-throughput assays for studying genetic variants

Knowing how changes in coding sequence result in protein behavioral changes is important. However, it is also necessary to understand how cis-regulatory codes govern protein production when linking genetic variation to gene expression changes for designing proteins for applications from mRNA therapy to synthetic biology. Gene expression is a multi-step process controlled by dense, frequently overlapping regulatory elements that can affect many processes. These cis-regulatory elements (CREs) are often spread out before, after, and inside the coding sequence for any given protein. Given the complexity of this code, many sequences can be necessary when determining what a CRE does in a given context - which can become highly complicated given the limited set of genomic sequences we can observe. There are a fixed number of coding genes in the human genome, and human population genetic

variation provides limited additional data due to the high degree of sequence similarity between individuals (19). Furthermore, natural sequence representation only allows us to observe sequences produced by natural selection, which results in a deficiency of extremely deleterious, potentially lethal variants for us to study.

Massively parallel reporter assays (MPRAs) are an extremely powerful method to study gene regulation and protein function that overcome natural dataset size limitations. In a recently published review (20) in which I am the first author, with co-authors YongSheng Shi and Georg Seelig, we covered the development, attributes, and common approaches of using MPRAs coupled with deep learning to investigate cis-regulatory elements. This section is partially adapted from a portion of that review—omitting its majority as this dissertation focuses on PPIs, not regulatory elements.

MPRA assays have been used to great effect to improve our understanding of gene expression**.** In an MPRA, a stretch of sequence known as the 'reporter' is believed to be relevant to the gene expression process of interest, and it is extensively varied to study the effects of what changes in that region do to expression. This set of reporter sequences is called a reporter library, which is then delivered into the system where expression is being studied (e.g., cell extract, cells, or animals), where reporter expression is connected to some quantifiable phenotype that can be measured.

There are two defining features of an MPRA:

1. Sequence variation is targeted to region(s) within a reporter gene (for example, the binding site of a protein) on a plasmid/vector to be inserted into the assay system. By limiting variation to part of a gene, it is possible to isolate its contributions to the process being studied. Often, sequence regions in the constructed plasmid/vector other than that being varied are fixed to help with this goal.
2. The phenotype of interest is read out using high-throughput sequencing. Instead of being forced to investigate variants individually or at small

> scales, we can leverage state-of-the-art sequencing technologies to simultaneously measure thousands to millions of sequences to investigate a gene expression process.

The origins of these MPRA characteristics can be traced to the 1990s, when in vitro experiments characterizing partially randomized nucleotide pools using Sanger sequencing were conducted. Due to Sanger sequencing's more limited thought, selecting a few reporters for sequencing via repeated rounds of amplification was necessary (21–23). At the same time, work was being conducted introducing the idea of coupling mutagenesis with selection to in vivo reporter assays (24,25). Some early examples of PPI focused assays combining these concepts include work where error-prone PCR was used to create a library of 7.5x10^5 mutants of a PDZ domain bound to a fragment of a binding partner, Myc (26). Of these many mutations post screening, only 15 potential mutant PDZ sequences were screened.

With the advent of next generation sequencing technologies in the 2000s, the size limits of functional library screening skyrocketed making high throughput assays possible by the early 2010s. Patwardhan et al. (27) was the first MPRA paper by the above two part definition, investigating the effects of single mutations on six promoter sequences. Many MPRAs soon followed, focused on a variety of regulatory elements. For example, some were focused on enhancers (28–30) or exon inclusion (31). Concurrent to the development of these regulatory-element focused assays, high throughput assays were developed to investigate the effects of mutations in proteins. Three papers in the early 2010s were released relatively closely by Fowler (32), Ernst (33) and Hietpas (34) introduced a high-throughput technique called Deep Mutational Scanning (DMS). In DMS studies, all possible single amino acid changes in a protein sequence are generated and measured for changes in function. DMS assay development and applications have been extensively reviewed elsewhere (35–37), and are now commonly used to study epistatic effects in protein sequences.

1.2.1. High throughput assays for PPI screening

The development of high-throughput protein-protein interaction (PPI) assays has significantly advanced our ability to map interactomes at scale. Early PPI screening approaches relied on *in vitro* techniques such as arrays displaying synthetic peptides or proteins (38), or in vivo methods such as transforming thousands of individual yeast colonies on large arrays, where only a small set of selected clones were sequenced (39). These methods laid the foundation for systematic PPI discovery but were limited by low throughput and labor-intensive workflows. As the field progressed, mass spectrometry- and improved protein array–based methods were introduced, enabling more comprehensive mapping but requiring extensive protein purification (40,41). Display technologies such as yeast and phage display later harnessed next-generation sequencing (NGS) to scale up interaction profiling, though these approaches were constrained to 'several-versus-many' designs for some time (42). A more recent innovation, AlphaSeq, exploits the yeast mating pathway for library-on-library screening, improving throughput but still facing limitations related to protein folding on the yeast surface (43).

Yeast two-hybrid (Y2H) assays are an *in vivo* PPI assay introduced in the 1980s. In Y2H, one protein is fused to a DNA-binding domain (DBD) and the second to a transcriptional activation domain (AD). If the proteins interact, a functional transcription factor is reconstituted and drives the expression of a growth-essential enzyme. Initially, Y2H assays were conducted in a low-throughput, plate-based format, testing a limited number of interactions simultaneously (44). These scale limitations were partially overcome by advances in lab automation and improved pooling strategies, which allowed proteome-scale screening (45). With the advent of next-generation sequencing (NGS) technologies, Y2H assays were adapted into high-throughput formats (HT-Y2H) (46–55). Enzyme complementation assays, which use the reconstitution of an enzyme to measure PPI strength instead of the reconstitution of a transcriptional factor, allow for alternative PPI testing approaches. Custom computational workflows have further improved throughput (56,57). One issue is

that most HT-Y2H methods require protein library construction in *E. coli* and separate transformation of MATα and MATa yeast strains. High-throughput bacterial two-hybrid systems offer a noneukaryotic alternative that circumvents these steps (58). However, for proteins of eukaryotic origin, yeast remains a favorable host due to its ability to support more accurate folding, solubility, and post-translational modifications for such proteins (59). Y2H, and HT-Y2H, and other HT yeast-based assays remain critical to PPI discovery efforts.

1.2.2. Machine learning models for high-throughput assay data

Training ML models on high-throughput data has become popular because these models can learn complex data relationships and can be used to predict activities even for sequences not yet tested experimentally. Moreover, models can be applied to stratify rare or de novo variants or guide the design of synthetic sequences.

Early work in learning from MPRA data fit equations, often inspired by biophysics, to observed trends in data (60,61). Classical ML models using sequence features as input were also common, including linear and logistic regression models (30,62–65) and decision trees (66). At times, such relatively simple ML models can effectively capture the behavior being investigated, and they lend themselves to easy interpretability by examination of model weights. Regression remains a relevant modeling technique for investigating processes from stability (67) to splicing determinants (68). Likewise, ensemble models of decision trees remain popular, such as using gradient boosted regressors for predicting splicing (69).

With their many parameters, deep learning models are adept at modeling non-linearities. The development of neural network models using biological sequences as inputs (70–72) has paved the way for these approaches to be applied to high throughput data (73–77). Deep learning models for genomics (71,78) and protein applications (79,80) are reviewed in detail elsewhere, but we briefly cover a few key concepts. A network architecture often used with MPRA data is the convolutional neural network (CNN) (**Fig. 3A**, left). CNNs include convolutional layers, which

consist of pattern-detecting filters that are scanned across inputs to evaluate how well each position matches the filter. The final layers in a CNN compress prior layer information into a set number of outputs. Recent MPRA modeling work has exploited architectural changes to enhance model performance and interpretability. Recurrent neural networks (RNN) can also be used for biological sequences. In RNNs, information moves sideways through layers and forward from input to output, allowing recurrent layer nodes to have internal memory states that allow the preceding and following nucleotides in a sequence to influence the current state. Models combining CNN and RNN elements are also common with discrete biological sequence inputs (**Fig. 3A**, middle) (81–83).

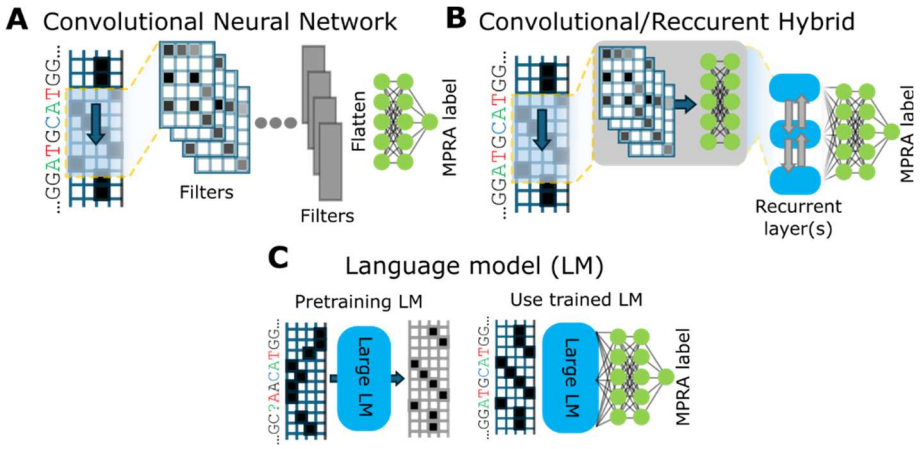


Figure 1.1. Examples of deep learning network structures for discrete biological sequence inputs. A. Convolutional neural networks (CNNs) use 1-d filters to extract patterns from inputs, which are then fed into subsequent layers. B CNN and recurrent neural network (RNN) hybrid architectures are also popular for discrete biological sequence predictions C. Language models can be pre-trained on large datasets of genomic or protein data, and then used as part of predictors for specific tasks such as MPRA label prediction.

A final class of models relevant to high-throughput assays are large language models (LLMs) (**Fig. 3A,** right**)**. LLMs have yielded promising results in protein sequence-to-function prediction tasks (84,85), where they are trained on databases of protein sequences to fill in masked positions with the most likely amino acid. These models can then be used to generate sequence embeddings for a downstream model, with

the assumption that the LLM will already contain protein sequence distribution knowledge to build off of (86). Similar foundational LLMs have begun to be trained on genomic data (87,88).

1.3. PPI specific models

Given the importance of PPIs for health and our understanding of basic biology, creating models to predict if an interaction will occur between proteins reliably without conducting expensive and time-consuming assays has been a popular research topic. Early PPI predictors included many simple machine learning models trained on protein features or protein sequence which classified two proteins as interacting or not, and quickly progressed to using deep learning models. A second common task is to predict if the effect of a mutation on a PPI: that is, if the mutant protein results in a stronger or weaker interaction than normal. As more data and more complicated models have become available for both the binarized PPI occurrence and interaction strength change prediction problems, there have been several reviews on the progress of this field (89–92). However, there are several issues in the PPI classification field stemming from poor communication in the literature and inconsistencies in train/test dataset practices which render it difficult to gauge how these models perform.

1.3.1. General PPI classifiers and train/test splitting methods

The first major problem with PPI classification stems from the lack of high-quality negative data to train with. It has been estimated that the percent of positive interactions out of all possible protein interactions is smaller than 2% (93), and may be as small as 0.1% (94) **.** Therefore, most proteins do not interact – however, while assays have undoubtedly screened many proteins with low to no interaction signal with modern PPI assays which sequence non-interactors, this data is rarely added to interaction databases (of which there are many, as reviewed here (90)). There is only one database of likely non-interacting protein pairs, the Negatome (95), which was

created by manual annotation of tested pairs which did not result in a strong PPI from large-scale interaction datasets, and later supplemented with text parsing (96). Unfortunately, the Negatome has not been updated for over a decade but remains the only specifically curated negative PPI database. Other databases contain lesser amounts of negative PPI data: for example, the Intact database (97) has a relatively small number of negative interactions. The majority of these Intact negatives come from a single paper (98) where isoforms of human proteins were screened using Y2H-methods to determine their PPI profiles when compared with the canonical isoform.

As such, it is common to generate potential negative interactions for training. One standard method to generate negative pairs includes pairing proteins from differing cellular compartments or with different annotations, with the motivation that since these proteins exist in different compartments, they would not interact. However, this approach has been heavily criticized (99–101) due to biases it introduces in the training set, which have been shown to affect the generalizability of predictors using it (99). Additionally, it has been theorized that the separation of proteins into differing compartments is part of the regulation process to prevent proteins that can interact from engaging in unwanted interactions (102). An alternative method to generating negatives is to pair proteins from the dataset uniformly at random, assuming that the low likelihood of any two random proteins interacting from a large set is small enough to lead to minimal false negatives (99). This second method of negative generation has pitfalls when employed in non-general settings, such as training predictors for a single family of binding domains, as the assumption of a low false negative rate may not hold. Various methods have been proposed to combat this: one involves selecting negatives based on the minimum path needed to traverse between the proteins in an interaction graph (100), only choosing negative examples with low sequence similarity to positive pairs (101,103), or using proteins with low degree in a protein interaction network to create negative pairs (104). However, each of these comes with its own drawbacks as they constrain the problem and introduce biases in the negative interaction data. For example, it has been shown that limiting the sequence similarity between interacting and non-interacting proteins makes the

classification problem appear easier than it is when assessing model performance (101). Often, negative interactions are generated at a 1:1 ratio with the positive interactions selected for the dataset, but it is also common to generate them at different rations (1:50, 1:100, etc.) with positive to mirror the low rate of PPIs occurring naturally.

The structure of the generated graph of interactions has been theorized to lead to predictors which have high performance not because they learn generalizable patterns from their inputs, but because they have learned about the topology of the underlying network (105,106). This may be partially due to the natural structure of naturally occurring PPI networks: the majority of proteins have few partners, while some are known as 'hub' proteins which can have many interactions. Intertwined with this network structure issue is the most critical dataset issue for PPI classifiers: that the partitioning method of the dataset for training leads to vastly different model performance.

There is a significant breakdown in communication in the literature about the partitioning problem, with it having been repeatedly 'rediscovered' over the last few decades (107–109). One of the earliest recognitions of this problem was a letter to the editor by Park and Marcotte in 2012 (107), which showed that 50 studies at the time had the same performance breakdown. If interactions are randomly chosen from the network to be assigned to train/validation/test sets, model performance is high on the test set. However, if the PPI network is partitioned on the protein level to generate three classes of test set interactions, performance drops drastically on novel protein predictions. These classes are PPIs where both proteins are found in the training set (class 1/C1), PPIs where only one protein is found in the training set (class 2/C2), and PPIs where neither test set protein is found in the training set (class3/C3). As the end-goal for PPI classifier models are tools which can be used for *in silico* screening of how new proteins will behave, a random partitioning approach cannot reliably predict how the network will behave in its likely use case, which corresponds to C2 or C3 PPIs. Despite this early recognition of this issue,

many models continued to be trained with random partitioning (89,92), creating a narrative that general PPI classification was a solved problem with high accuracy classifiers available.

A smaller number of works give a more accurate state of general PPI classification. These include several recurrent-network based DL models, and those incorporating pLM and other embedding types (110–112). Recent high C3 performances have reached as high as 0.81 AUPRC (111). Datasets and train/test splits other than the C1/C2/C3 approach have been created to combat this problem. Some of these take a less stringent approach to protein partitioning, where the test set is created by selecting starting proteins from which to traverse the graph (108). A breadth-first search traversal leads to the BFS test set, which mimics the use-case of predicting how a new set of proteins will interact with each other, as well as the existing proteins in the training network. Alternatively, a breadth-first search traversal leads to a BFS test set, which mimics the use-case of a set of proteins that do not interact heavily with one another but have many interactions with the training network. These datasets have been used multiple times (113–115) however, the sets are rarely partitioned further when test performance breakdowns are shown which means that comparisons to BFS/DFS split and C1/C2/C3 protein partitioning splits are difficult to make

This disconnect in train/test splitting procedures is especially relevant for the next section, which discusses the problem of classifying PPIs using predicted complex structures.

1.3.2. PPI prediction with structural predictors

The goal of the critical assessment of structure prediction (CASP) since its inception in 1994 was to encourage the development of predictors of protein structure from amino acid sequence with a bi-annual competition for the best performing models. In 2018, this goal was partially realized with the entry of Alphafold (AF) (116), and subsequent years have seen improvements in protein structure performance with new

AF models. In particular, the AlphaFold2 (AF2) greatly improved performance, with some considering the prediction problem 'solved' (117). Additional structure predictors based off AF versions have been developed – such as AlphaFold-Multimer (AF-M) (118), trRosetta (119), and RosettaFold (120). In general, these models can be divided into two approaches: those which require multiple sequence alignment (MSA) inputs, and those which do not.

Making an MSA, which is an alignment of similar sequences across large databases of protein sequences, condenses information about the conservation of certain positions in the protein sequence. Conservation information can indicate which positions are important for structural integrity. By taking in this information, sequence information, and structural alignments, structural predictors can be used to predict an output structure and per-residue confidence values in said structure. An alternative approach are models which do not require a MSA input, but instead attempt to extract conservation information using pLMs such as ESMFold (85).

While these models have many fascinating applications, the most relevant here is that they can be used to predict protein complexes. These complexes can then be used to predict PPI information by examining its predicted error metrics or as inputs to additional predictors. Early versions of structural predictors such as the original AF model required linker sequences between the proteins to be complexed in order to output a complex structure. However, later AF versions have been specifically adapted to allow complex predictions, such as AF-M (118) and the latest AF model AF-3 (121). Complex predictors have been used to make predictions across interactomes for organisms– mostly by predicting many thousands of complexes and ranking them by their predicted structure quality (122,123). Structural predictor-based pipelines and approaches have also been developed to run large all-by-all complex predictions (124,125). Many of these works use these *in silico* screens to rank likely interactions using metrics for predicted model quality. One of these values is the predicted local distance difference test (DTT), which is popularly used to

determine whether a predicted structure is high-quality. Another metric of AF performance is the predicted template modeling (PTM) score, which attempts to give an aggregate score of model quality. A complex specific score that can be calculated is interface PTM (iPTM), which focuses on predicted interface quality. While such metrics have been shown to be useful for decoy selection (the task of selecting the correct complex for an interaction from a pool of potential candidate complexes for the interaction), less work has been done to determine their ability to separate interacting from non-interacting protein pairs (126,127).

Of the work done assessing AF metrics for classifying proteins as interacting, some claim predicted error metrics are insufficient for separating non-interacting complexes from interacting (128), while other works argue they have a good ability to do so (129). Models and composite metrics going beyond metrics like pLDDT, PTM, and iPTM have been made to use inputs of predicted structures to classify two proteins as interacting (130). However, these approaches construct train/test datasets without any acknowledgement of the held-out protein prediction issue or employed dataset filtering methods which largely ignored the body of past PPI prediction work (128–131). For example, a DL network taking in structural features of AF-M predicted complexes only used a dataset of 600 positive and 600 negative PPIs between 375 proteins for training with random 5-fold CV (131). Another recent PPI classifier work using engineered features from AF2 predicted structures based their dataset on heterodimers with complexes in the PDB, filtered such that no protein had >30% sequence similarity. They created 'compelling decoys' from the remaining 1,481 heterodimers remaining by looking for structurally similar protein chains to randomly combine for negative interactions. These positive and negative interactions were then randomly split into train/validation/test sets (129). However, the source of structures for dataset creation heavily biases the datasets towards only including proteins which are amenable to crystallization and so its behavior as a general classifier is unclear, as the PDB is biased towards structured domain-domain interactions (132).

Other non-predicted structure input PPI classifiers and decoy rankers exist for structure-based prediction. However, these also seem to not take into account past PPI prediction literature for train/test split designs, or focus primarily on decoy selection (133–137). Therefore, it is unclear how current structure predictors actually perform in classification of proteins as interacting or non-interacting, as they have not been evaluated on more stringent PPI train/test datasets with C1/C2/C3 test sets, or BFS and DFS train/test splits.

Finally, there has been work which suggests that while AF and other models learn to predict structures – this should not be conflated with learning protein folding. Work comparing multiple structure predictors, including AF2, to experimental folding pathways found that worse than trivial performance (138). Additional works found little to no correlation with $\Delta\Delta G$ and pLDDT (139).

1.4. Outline

In Chapter 2, I will cover work on a HT-Y2H assay published in (140). This assay was extensively validated on literature datasets and used to measure large all-by-all assays of designed heterodimers. Work exploring large datasets to extract interesting subsets of orthogonal proteins will be discussed. Orthogonal proteins sets are those where there is a set of 'on-target' interactions which are strong, and very little crosstalk between the proteins in the set otherwise. It will also cover training simple linear regression and logistic regression models on AF error metrics and simulated energy terms for predicted complexes.

Chapter 3 will cover work that was done chronologically before the MP3-seq paper, using a variant of the MP3-seq assay to collect a DMS PPI dataset for human septin proteins. As such, it is currently in the process of being updated for publishing. It will cover the design of the dataset to be collected, and the past modeling of this dataset.

Chapter 4 will cover work on a feature attribution method known as scrambling neural networks, published in (141). In particular, the use of the feature attribution method for exploring complex regulatory logic will be covered. Additionally, a example of using scramblers to explore the determinants of designed protein interactions will be explored.

Finally, in Chapter 5, I will cover current projects. Part of this chapter will focus on a human immune cell data resource I compiled for cytokine-receptor interaction downstream effects, as part of a collaborative project analyzing a very large scRNA-seq dataset. Septin modeling update plans will be briefly covered, as will closing remarks about PPI predictor dataset design.

Chapter 2: Massively parallel protein-protein interaction measurement by sequencing

This chapter covers a high throughput Y2H-based PPI assay developed in the Seelig Lab, known as massively parallel protein-protein interaction measurement by sequencing (MP3-seq), which is published in (140). The assay method was initially conceived by Benjamin Groves but was fully developed by Alexandr Baryshev throughout his PhD, with final data collection handled by Cirstyn Michel after Alex's graduation. My involvement in the project was to devise a data cleaning pipeline, formalize the comparisons to literature datasets, write the manuscript, and explore the data for interesting patterns. This exploration involved finding ways to reduce PPI networks generated by large all-by-all MP3-seq screens to extract interesting protein sets. Several of the screened sets involved alpha helix-based protein binders designed by Ajasja Ljubetič. Focused assays were run to determine the binding determinants of a particularly successful design. Additionally, Ajasja ran complex predictions for two sets of the screened proteins with multiple versions of AF, and energetic simulations in Rosetta for said predicted complexes. I did extensive work using the metrics he collected from these simulations to train predictors for MP3-Seq output values.

2.1. Assay motivation and experimental method overview

There are many varieties of HT PPI assay methods, as covered in section **1.2.1**, many of which involve Y2H or yeast surface display methods. In MP3-seq the identity of each protein is encoded in a DNA barcode, and the abundance of a barcode pair before and after selection provides a proxy value for interaction strength. Homologous recombination in yeast is used to assemble plasmids encoding the protein pairs of interest, their barcodes, and all other required elements for selection, which circumnavigates the requirements of some other HT-Y2H workflows for plasmid cloning in E. coli or yeast mating. Additionally, the proteins fold and interact inside the cell instead of on the surface. While it is like other assays, a

diverse set of PPI measurement methods is necessary to validate interactions, as discussed previously, since their interaction can be context-dependent and individual proteins may not fold correctly or interact correctly in an environment sufficiently different from their native context.

In brief, the MP3-seq workflow consists of constructing a plasmid library through homologous recombination in Haploid MATa-type yeast to measure all possible interactions for a set of proteins. The yeast are transformed with a mixture of DNA fragments to do this: a backbone carrying a centromere sequence, the selection marker (the growth-essential enzyme his3), a DBD (the Cys(2)His(2) zinc-finger domain of the mouse transcription factor Zif26842), an AD (the herpes simplex virus-derived protein domain VP16), and fragments containing one of the proteins of interest and their associated barcode of interest separated by a terminator. After transformation, an outgrowth and selection step in media without Tryptophan (TRP) ensures plasmid maintenance in the yeast. At the same time, the centromere sequence means that each cell will contain, on average, one assembled plasmid, essential to linking the growth of transfected yeast to barcode counts. The promoter corresponding to the DBD is used to drive the expression of the selection marker. The cells are then transferred to media lacking histidine (HIS). By amplifying and sequencing the barcode regions from samples from after TRP but before HIS and after HIS selection, their enrichment can be calculated from barcode counts to measure interaction strength (an overview of the experimental pipeline can be seen in Figure 2.1).

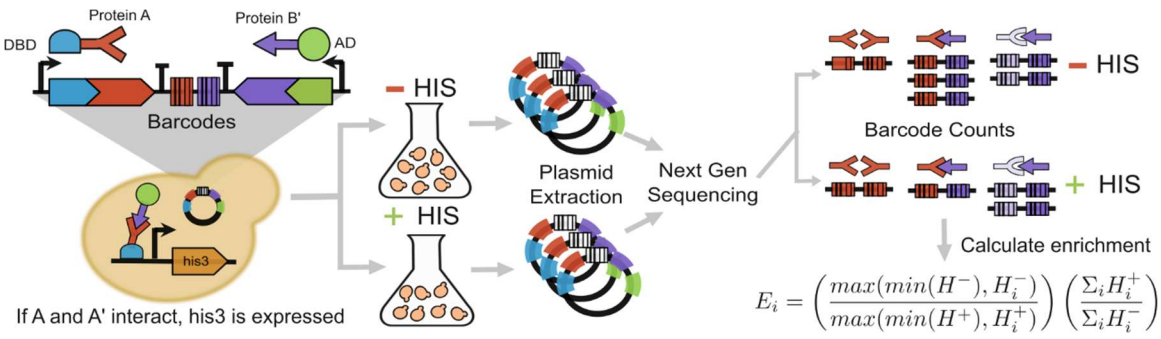


Figure 2.1. Overview of the MP3-seq experimental pipeline. If protein A and A' interact, the growth-essential gene his3 will be expressed. By selecting in HIS- media, only cells with plasmids encoding interacting pairs will survive. Plasmids can be extracted for sequencing from HIS- and HIS+ samples, the barcode amplified and then sequenced to get counts corresponding to each PPI i. Barcodes can be used to calculate enrichment (E) using library size-normalized read counts with a pseudo count of the minimum detected value per condition.

2.2. Development of data pipeline

One of the main challenges I faced when cleaning the MP3-Seq data was the lack of sophisticated PPI-focused sequencing analysis methods I could use for the analysis. Y2H all-by-all assay data has distinct structural considerations and potential error modes to be considered during analysis. For example, all-by-all PPI assays such as MP3-Seq can yield symmetrical data. In our case, every library replicate retained for the study was carried out with every protein partner fused to both the AD and DBD (that is, for every PPI testing protein A and B, both AD-A and B-DBD and AD-B and A-DBD were screened). Also important is an error mode that can occur in Y2H known as autoactivation: in it, the fusion of the assayed proteins to the assay domains results in unspecific activation of the HIS gene. While this error mode can easily be seen visually when analyzing the data as rows or columns of strong interactions without symmetry, developing a standardized way to detect and 'correct' autoactivation was necessary when developing the pipeline.

Unfortunately, there were (and still are) only a handful of papers focused on HT-Y2H assay analysis, which are focused on mating-based assays (56,57). One of these works (57) applied a differential analysis program, DESeq2, to their collected data to conduct a more rigorous analysis to identify strong interactions. Differential analysis programs are used to identify genes from sequencing data whose expression level changes significantly between two or more biological conditions (ex: healthy vs treated samples) but can also be applied to a wide variety of applications with pre- and post- selection read counts like the read counts from the pre and post HIS treatments in MP3-seq by treating each PPI like a gene (142). DESeq2 takes raw

read counts across multiple replicates, performs normalization to account for variables like sequencing depth and other technical biases, models the variance in counts for the genes, and calculates log fold changes (LFC) in expression between conditions. A particularly useful component of DESeq2 is that it performs statistical testing and generates Hochberg-adjusted Wald test p-values which can be used to identify genes with LFCs significantly different from 0. However, it is highly recommended to use raw sequencing counts when modeling the read counts with DESeq2. Therefore, it would be prudent to preserve the raw sequencing accounts as much as possible when creating input tables for DESeq2 for MP3-Seq data, while correcting for autoactivation.

To detect autoactivators but avoid flagging proteins as autoactivators which are strong interactors in general, a measure of dispersion which focuses on the middle of the distribution could be useful. One such measure is the trimmed interquartile mean (IQM) (143), the mean of values only falling in the middle 50% of a distribution. By calculating the IQM for all assayed proteins when fused to either MP3-SEQ domain, we are given two distributions of IQMs: one for the DBD, and one for the AD. Extreme positive outliers for each of these distributions can then be found, (those greater than 3* IQR) and marked as potential autoactivators. See Figure 2.2 for an example of the IQM distributions for DBD and AD for one of the replicates of an all by all assay from processing the MP3-seq data. This heuristic identification approach could theoretically be tuned to be more specific use cases (i.e., instead of an IQM, a trimmed mean of 90% of the data could be used).

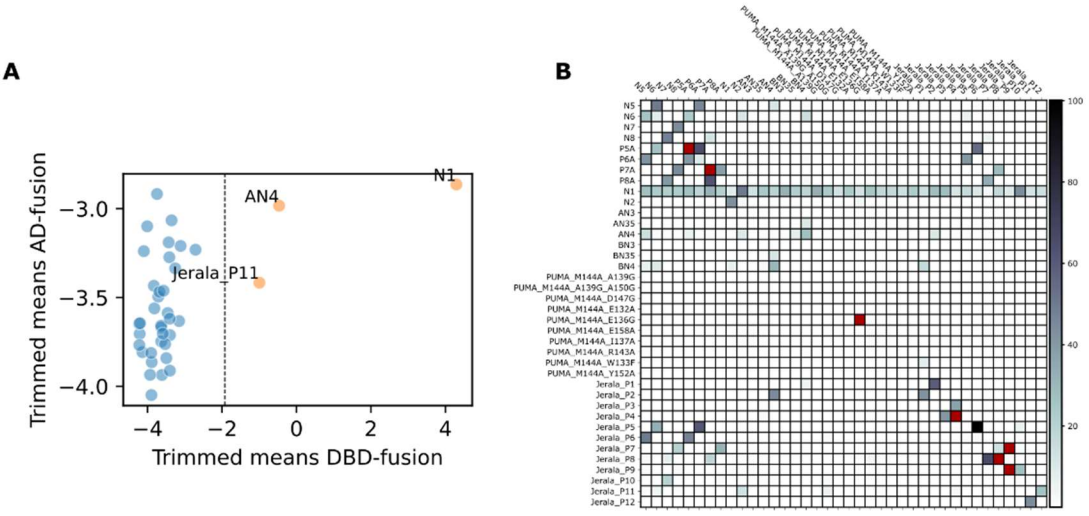


Figure 2.2. Example of trimmed mean identification of autoactivators for library L68. A. Trimmed means for proteins fused to the AD and DBD, with those identified as potential autoactivators shown in orange with annotations. The dotted line is the 3*IQR identification cutoff. B. The heatmap of the enrichment values for the L68 replicates. DBD are the rows, AD are the columns. Red squares indicated missing interactions. Note the autoactivator N1 visible as a dark row without a symmetrical dark column.

After identifying all autoactivators, a second problem develops. It is still recommended that DESeq2 have close to raw data counts as inputs to best model the distributions of the reads for log fold change and p-value calculations. Including all replicates possible when fitting the models for correcting read counts and calculating LFC is also ideal. One issue, however, is that DESeq2 does not permit missing data points during the modeling process. An immediate option is to drop all interactions for which there is autoactivation in any replicate. However, since the autoactivation degree was inconsistent between replicates, the same proteins would not act as autoactivators in every experiment. Another approach was to try and infill the autoactivation measurements with a value derived from the non-autoactivating fusion order. Infilling would allow DESeq2 to run on the data successfully, retaining the bulk of the raw counts for most replicates. To keep with the musical naming theme, this correction approach was called Autotune and is given by Equations 2.1-2.2, where i is the interaction being infilled FNA_i being the non-

autoactivating fusion order values for interaction i, FA being the autoactivating fusion order, and E_{FAi}being the enrichment of the non-autoactivating fusion order for interaction i.

$$HIS_i^+ = \frac{HIS_{FNAi}^+}{\Sigma_{FNA} HIS^+}(\Sigma_{FA} HIS^+) \quad (2.1)$$

$$HIS_i^- = \frac{\Sigma_{FA} HIS^-}{\Sigma_{FA} HIS^+} E_{FAi}(HIS_i^+) \quad (2.2)$$

A final interesting characteristic of the data for the MP3-Seq assays, and indeed, any all-by-all PPI assay where the protomers can be fused to two different portions of a reporter, is that it is symmetric. Assuming the data for an interaction is represented by a matrix where rows represent proteins being fused to the DBD and columns fusions to the AD, each non-homodimer PPI has two symmetrical entries along the diagonal of the matrix, with homodimers being the diagonal entries. Since each non-homodimer has two entries, we can treat these similar to separate measurements of the same PPI. I chose to call these 'pseudoreplicates' as they are not truly independent experiments. However, it is convenient to treat them as such because it allows us to simplify the analysis of MP3-seq datasets greatly. Instead of having two values per PPI, creating input tables for DESeq2 using pseudoreplicates can produce one value per interaction, which yields what we refer to as a pseudoreplicates-LFC (P-LFC). P-LFCs can be helpful when comparing MP3-seq LFCs with other values, such as dissociation constants, where only one Kd exists per PPI. An important note is that homodimers could not be recovered with normal Autotune, but they still needed to be infilled when making pseudoreplicates. Instead, Equation 2.3 was used to infill the HIS- values for both pseudoreplicates per replicate. An overview of the entire computational pipeline can be seen in Figure 2.3.

$$HIS_F^- = \frac{\Sigma_F HIS^-}{\Sigma_F HIS^+} E_{FAi}(HIS_{Fi}^+) \quad (2.3)$$

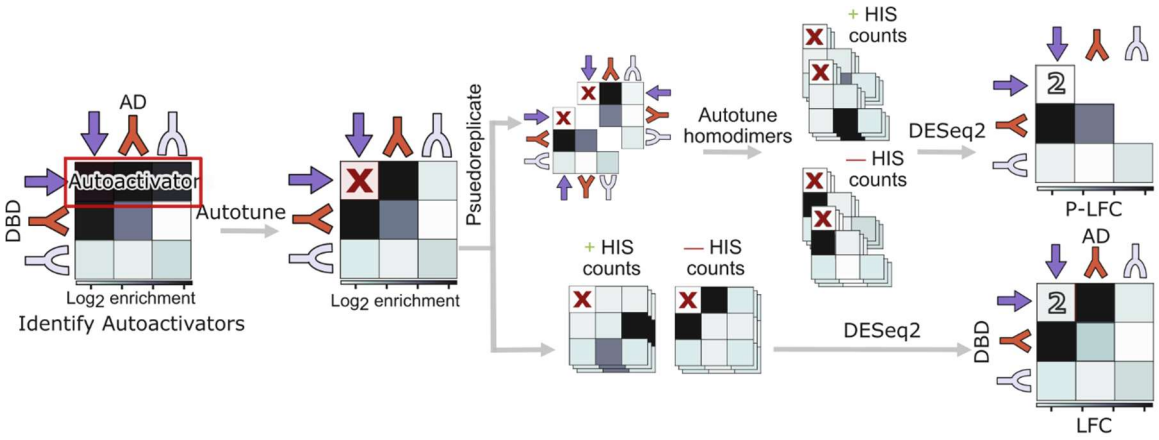


Figure 2.3. Overview of the MP3-seq computational pipeline. After enrichment calculation, each replicate is screened for autoactivators and corrected with Autotune Equations 2.1 and 2.2. Replicate pre- and post-selection barcode counts are merged directly with DESeq2 or split into pseudoreplicates (with homodimers corrected with Equation 2.3). Replicates are then merged to obtain the log fold change (LFC) or pseudoreplicate log fold change (P-LFC).

2.3. Validating MP3-seq with literature interactions

Several datasets were used to validate MP3-seq, spanning various PPI types. For the purposes of this thesis, I will only discuss some of the tested protein sets: a set of simple, short, coiled-coil orthogonal binders, a set of domain-peptide mediated interactions, and a set of disorder-mediated interactions.

2.3.1. Simple orthogonal interactions

The most extensively measured set of PPIs to validate MP3-seq were 144 pairwise interactions between six orthogonal single coil protomers, the National Institute of Chemistry Peptides (NICP) set (144) (Figure 2.4A). LFCs for both orientations were calculated from five experimental replicates, with interactions occurring almost exclusively between designed (on-target) partners (that is, P1:P2, P3:P4, etc., Figure 2.4B), as was seen in the original design paper. Moreover, there was a good correlation ($r^2 = 0.74$) of MP3-seq LFCs with luciferase expression assay interaction measurements in HEK293T cells from the NICP design study (Figure 2.4C). An

additional coiled-coil set of protomers was designed by (145) to have a range of interaction strengths instead of orthogonal behavior. This set was also screened with MP3-seq, and P-LFC values agreed very well with the design study Kd values (r^2 = 0.91, Figure 2.4D).

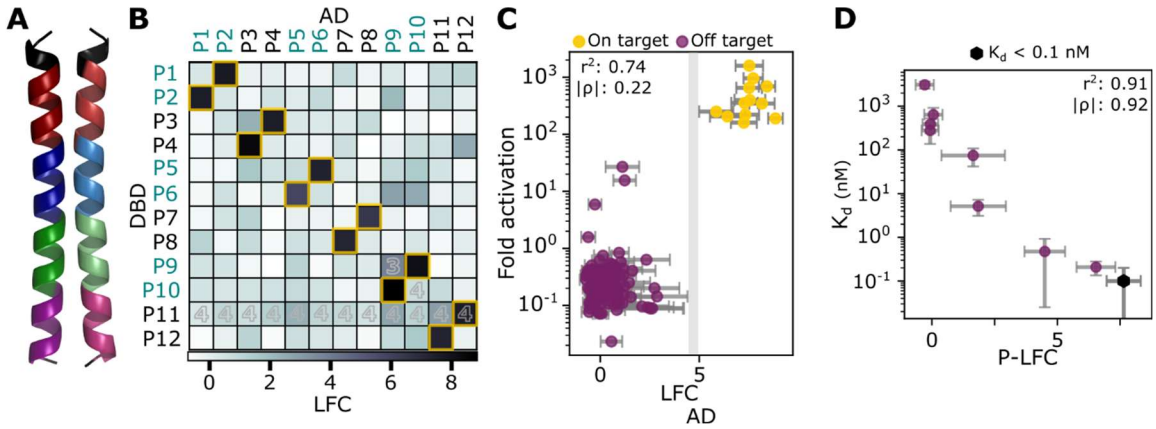


Figure 2.4. Coiled-coil designed dimer validations sets. A. An example of one of the NICP series, P1-P2. B MP3-seq LFC of the NICP series interactions. All MP3-seq values were calculated from five biological replicates except for those labeled, where labels indicate the number of replicates available. Outlines denote designed on-target interactions. C. Correlation of the on-target and off-target NICP series MP3-seq LFCs with average fold activation fluorescence values (144) (n = 141 PPIs; three homodimers had insufficient reads or were autotuned and were omitted). The gray bar is the gap separating on-target and off-target interactions in the orthogonal set. D. Correlation of n = 9 PPI MP3-seq P-LFCs with Kd values from (145).

2.3.2. Domain-peptide mediated interactions

To validate MP3-seq with non-coiled proteins, we tested a set of proteins characterized by biolayer interferometry (146) and Alpha-seq (43) composed of nine *de novo* designed inhibitors of six homologous proteins from the human BCL2 family (Bcl-2, Bcl-xL, Bcl-w, Mcl-1, Bfl-1, Bcl-B) (146). A crystal structure of one of the inhibitors bound to its BCL2 target is shown in Figure 2.5A**.** The P-LFCs of two replicas of all BCL2 homologs against the inhibitors are shown in Figure 2.5B and agreed well with dissociation constants obtained from biolayer interferometry from the original study (r^2=0.61, n=43) and Alpha-Seq percent survival for their low-throughput pairwise and high-throughput batched assays on the same interactions

(batched r^2: 0.45, paired r^2: 0.61, n = 43). MP3-seq interactions are measured in yeast, while biolayer interferometry uses purified proteins in a specialized instrument, and Alpha-seq displays proteins on the yeast surface. These different measurement methods may partially explain the variation between our results and those published earlier.

One important aspect of this protein set is that some BCL2 inhibitors failed to produce biolayer interferometry Kd measurements, likely because the interactions were below detection limits. We examined distributions of detected (n = 43) and undetected (n=11) interactions. We found that the mean P-LFC value of detected PPIs was significantly greater than that of undetected PPIs (one-tailed independent t-test, H1: $\mu_{detected} > : \mu_{undetected}$, t: 4.51, p: 1.858e-5, Figure 2.5C). Pairwise Alpha-Seq also had a significantly greater mean of detected interactions than undetected (H1: $\mu_{detected} > : \mu_{undetected}$, t: 1.91, p: 0.0307), though the high-throughput batched Alpha-Seq did not (H1: $\mu_{detected} > : \mu_{undetected}$, t: 1.36, p: 0.0904).

2.3.3. Disorder mediated interactions

PUMA is an intrinsically disordered protein, which becomes ordered and coiled when bound to Mcl-1 at the BH3 pocket. A set of PUMA mutants was developed to investigate the effects of helicity degree on this IDR-dependent interaction by (147). We screened the peptides against a truncated Mcl-1 protein. We found a good correlation between P-LFC measurements with stopped-flow fluorescence Kd measurements for PUMA peptides interacting with the full Mcl-1 protein (r^2: 0.66, n = 13) (Figure 2.5D).

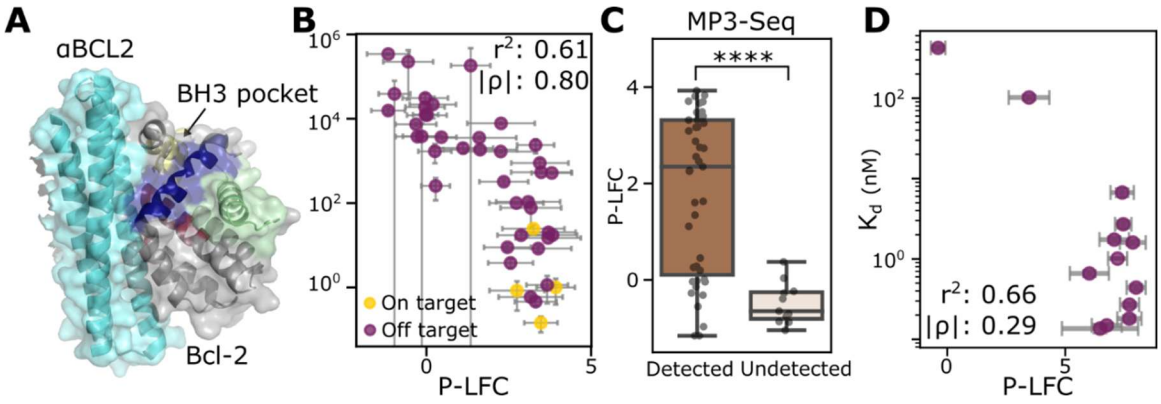


Figure 2.5. Validation of MP3-Seq with BCL2 proteins. A. Colored crystal structure of Bcl-2 and its designed BH3 binding inhibitor49 (PDB: 5JSN). B. Correlations with Kd measurements from biolayer interferometry C. MP3-Seq P-LFC distributions for interactions that were undetected by biolayer interferometry due to instrument detection limits (Kd $\geq$ 25 mM). D. Correlation of PUMA peptide and Mcl-1 P-LFCs from two experimental replicates with PUMA peptide and full Mcl-1 Kd.

2.4. Exploring PPI assays with graph representations

One of the benefits of using a differential analysis program like DESeq2 for data processing is that it calculates an adjusted p-value (p_{adj})_ of the significance for an LFC against a null LFC of zero (142). Therefore, we can differentiate significant interactions using the p_{adj}. One valuable application of this is that we can immediately significantly reduce the complexity of the datasets for visualization.

By only retaining the most significant interactions, we can augment the classic PPI visualization method where each protein is represented as a node and interactions as edges with MP3-seq data. By making each edge's thickness correspond to its P-LFC value, it becomes easy to view assays to see strong, significant interactors and other patterns of note. For example, the orthogonal NICP PPI set graph shows the six designed interactions (Figure 2.6). In addition, a second work created variations of the NICP proteins, modifying the original sequence to confer increased thermodynamic stability to their on-target interactions (148). By visualizing only the strong, significant interactions of these protomers in an all-by-all assay, the

shared binding behavior between variants of the original NICP proteins can be seen (for example, P5A and N5 are variations of P5, and interact with P6, P6A, and N6, Figure 2.6).

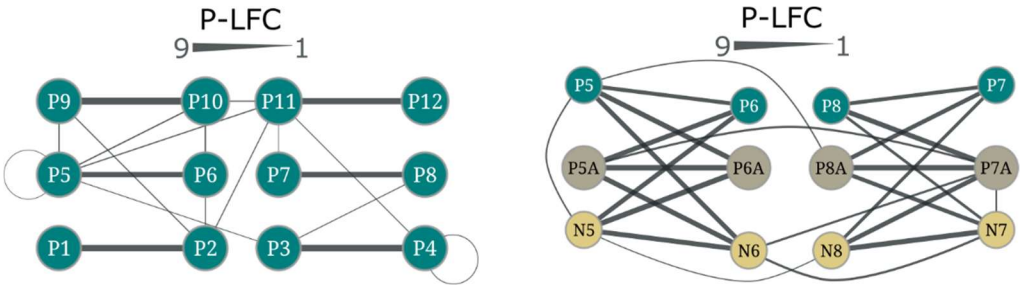


Figure 2.6. Validation of MP3-seq with coiled-coil heterodimers. Filtering NICP-series (left) interactions and E. P, PA, and N series (right) designed coil interactions47 to include only those with $p_{adj} \leq 0.05$. Line weights correspond to MP3-Seq P-LFCs.

2.4.1. Extracting potential orthogonal subsets from large-scale assays

The paper's largest set of assayed proteins consisted of several sets of coiled-coil-based heterodimers (DHDs) mainly designed using the approach of (149). These heterodimers consisted of four groups: a set of 35 dimers (DHD0), a set of 100 heterodimers (DHD1), a set of 21 more dimers including truncated designs or modified loop designs (DHD2), and a series of 5 dimers derived from a common parent design (mALb). For each on-target pair, one protomer is designated 'A,' and the other is designated 'B.' These designs were measured in several overlapping assays and two all-by-all assays, and the designs were evaluated for success (defined as any PPI with $p_{adj} \leq 0.01$ and P-LFC ≥ 4). The design sets had an approximately 20% success rate in the largest screen, consistent with past 22% success rates for α-helical bundles(150). While determining the success of designed pairs was the primary goal of the assay, the on-target interactions comprised only a small portion of the collected dataset. Most measured interactions consisted of "off-target" non-designed interactions between protomers. In fact, in the largest all-by-all screens of the dataset, only 33 of 914 PPIs with $p_{adj} \leq 0.01$ and P-LFC ≥ 4 were on target.

Given the potential applications of orthogonal protein sets, one of the use cases for this dataset is the extraction of potentially orthogonal heterodimer sets. The graph-based visualization technique above offers one avenue to do so: we used significant ($\alpha = 0.01$) positive P-LFC values to construct a weighted undirected graph as in Figure 2.6. Then, the problem of finding a set of heterodimers could be rephrased as finding a subgraph of the high-significance graph of only degree-one vertices without self-edges, such that the sum of the remaining edges would be maximized. The orthogonality gap is a metric when defining orthogonal protein sets: the difference between the weakest on-target interaction and the strongest off-target interaction (58). Making sure that a potential orthogonal subgraph possessed a sufficiently large orthogonal gap was also necessary.

$$score(G,e,c) = \begin{cases} w_{v_1,v_2}, & \text{if } deg(v_1) = deg(v_2) = 1 \\ w_{v_1,v_2}, & \text{if } deg(v_1) = 1 \text{ and } deg(v_2) \neq 1 \\ & \text{ and } max(v_i \ni N(v_2) \setminus v_1 : w_{v_i,v_2}) \leq cw_{v_1,v_2} \\ -w_{v_1,v_2}, & \text{if } deg(v_1) = 1 \text{ and } deg(v_2) \neq 1 \\ & \text{ and } max(v_i \ni N(v_2) \setminus v_1 : w_{v_i,v_2}) > cw_{v_1,v_2} \\ w_{v_1,v_2}, & \text{if } deg(v_2) = 1 \text{ and } deg(v_1) \neq 1 \\ & \text{ and } max(v_i \ni N(v_1) \setminus v_2 : w_{v_i,v_1}) \leq cw_{v_1,v_2} \\ -w_{v_1,v_2}, & \text{if } deg(v_2) = 1 \text{ and } deg(v_1) \neq 1 \\ & \text{ and } max(v_i \ni N(v_1) \setminus v_2 : w_{v_i,v_1}) > cw_{v_1,v_2} \\ w_{v_1,v_2}, & \text{if } deg(v_1) \neq 1 \text{ and } deg(v_2) \neq 1 \\ & \text{ and } max(v_i \ni N(v_2) \setminus v_1 : w_{v_i,v_2}) \leq cw_{v_1,v_2} \\ & \text{ and } max(v_i \ni N(v_1) \setminus v_2 : w_{v_i,v_1}) \leq cw_{v_1,v_2} \\ -w_{v_1,v_2}, & \text{if } deg(v_1) \neq 1 \text{ and } deg(v_2) \neq 1 \\ & \text{ and } max(v_i \ni N(v_2) \setminus v_1 : w_{v_i,v_2}) > cw_{v_1,v_2} \\ & \text{ or } max(v_i \ni N(v_1) \setminus v_2 : w_{v_i,v_1}) > cw_{v_1,v_2} \\ 0 & \text{else} \end{cases} \quad \text{where } v_1, v_2 \ni e \tag{2.4}$$

A greedy approach that would consider the entire graph score was devised to meet this goal. To find a graph satisfying our constraints, we developed a simple scoring function that rewards graphs based on existing orthogonal edges or those over a desired orthogonality gap and punishes graphs for nonorthogonal edges. A larger gap generally results in smaller sets (see Equation 2.4). This scoring function was used in a greedy graph reduction method, Deleting Undirected Edges Thoughtfully

(DUET), which removes a vertex and its associated edges each iteration from the graph until a one-regular graph remains (Figure 2.7A). The results of DUET versus a simpler graph reduction method, removing the highest degree node at each iteration, can be seen in Figure 2.7B with the corresponding scores of the approaches in Figure 2.7C, with DUET reducing the all-by-all design screen high significance graph from 1,562 edges between 270 vertices to 36 DUET pairs. An additional undesirable behavior for potentially orthogonal sets would be if DUET is biased toward selecting proteins with missing interaction data (and, therefore, fewer edges in the graph). To check if this was occurring, we ran permutation tests, summing the number of missing MP3-seq interactions between the proteins in the initial DUET results and uniform randomly sampled protein sets of the same size from the start graph. We did not find that DUET results had a significantly higher number of missing interactions (Figure 2.7D).

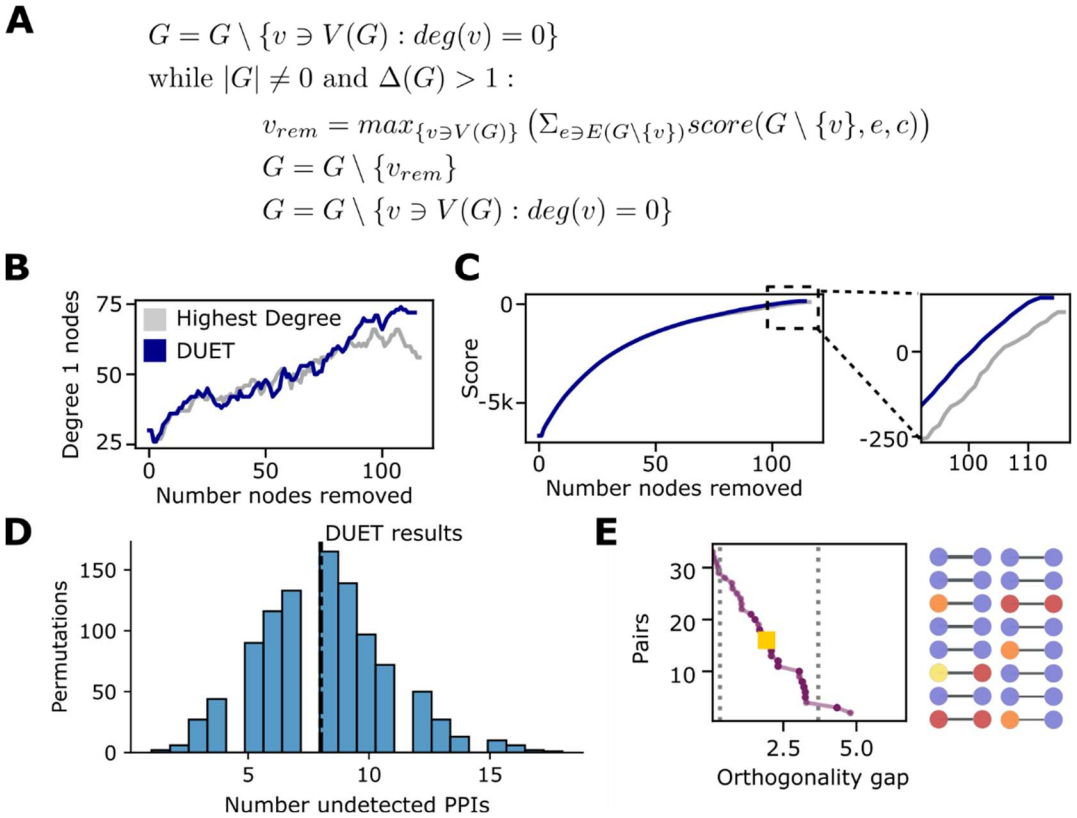


Figure 2.7. Orthogonal protein set search. A. DUET pseudocode, c = 4. B. Number of degree 1 nodes in the graph for DUET versus a simple greedy approach removing the highest degree node from the graph at each iteration. C. The *score* of the graph at each iteration of

node removal for the two approaches. Inset shows final iterations, and the gap between the scores. D. Permutation test for n = 10000 samples of 72 proteins from the initial graph for the total undetected PPIs between the proteins. E. Orthogonality gaps of the DUET final networks without significance filtering. Left and right dashed lines show the MP3-seq orthogonality gaps for BCL2 family inhibitors and the NCIP-series, respectively. Yellow squares correspond to half of the starting networks remaining. Number of DUET pairs at half the starting network size shown to right of graph.

The outcome of this process is a set of potentially orthogonal proteins. To better evaluate these sets, we used all P-LFC > 0 between DUET pair protomers, regardless of significance, for a more conservative analysis. As the graph G for DUET is created using only strong, significant interactions, it ignores all non-significant, weak interactions. First, we removed interactions with protomers for which the DUET pair P-LFC was lower than the highest non-DUET pair P-LFC. Then, we reduced the remaining DUET pairs by removing whichever pair had the largest non-DUET P-LFC one by one to yield smaller potentially orthogonal sets. Orthogonal gaps can be calculated for the P1-P12 and Bcl2 inhibitor designs to show the gap of DUET pair results throughout this process. It can be seen that the DUET sets reach orthogonality gaps and sizes comparable to orthogonally designed protein sets in the literature (Figure 2.7E). While these are not experimentally validated with additional assays, the DUET graph reduction algorithm demonstrates one of the many potential use cases of the scale of the MP3-SEQ method when used for large, all-by-all applications.

2.5. Modeling PPI interaction using predicted structure characteristics

To assess AF's ability to predict orthogonal interactions, we used AF2 and three AF-M versions (v1–v3) to predict complexes for all 144 NICP PPIs compared to coiled-coil predictors from the literature (58,151,152) . We evaluated performance metrics (predicted local distance difference test, IPTM, etc.) for predicting MP3-seq

LFC values and classifying on-target and off-target PPIs (Fig 2.7A-C). Both the top predicted model metrics and the average of model metrics were assessed, and it was found that the best-performing metric was the interface predicted TM (iPTM) score averaged across five predicted complexes. Also, AF-M was much better at NICP complex prediction (particularly for AUCROC and AUPRC of on- vs. off-target classification, Figure 2.7D). However, compared to the performance of coiled-coil binding predictors, the specialized predictors performed much better in correlation with LFC values classifying on/off target interactions AUPRC (Figure 2.7E).

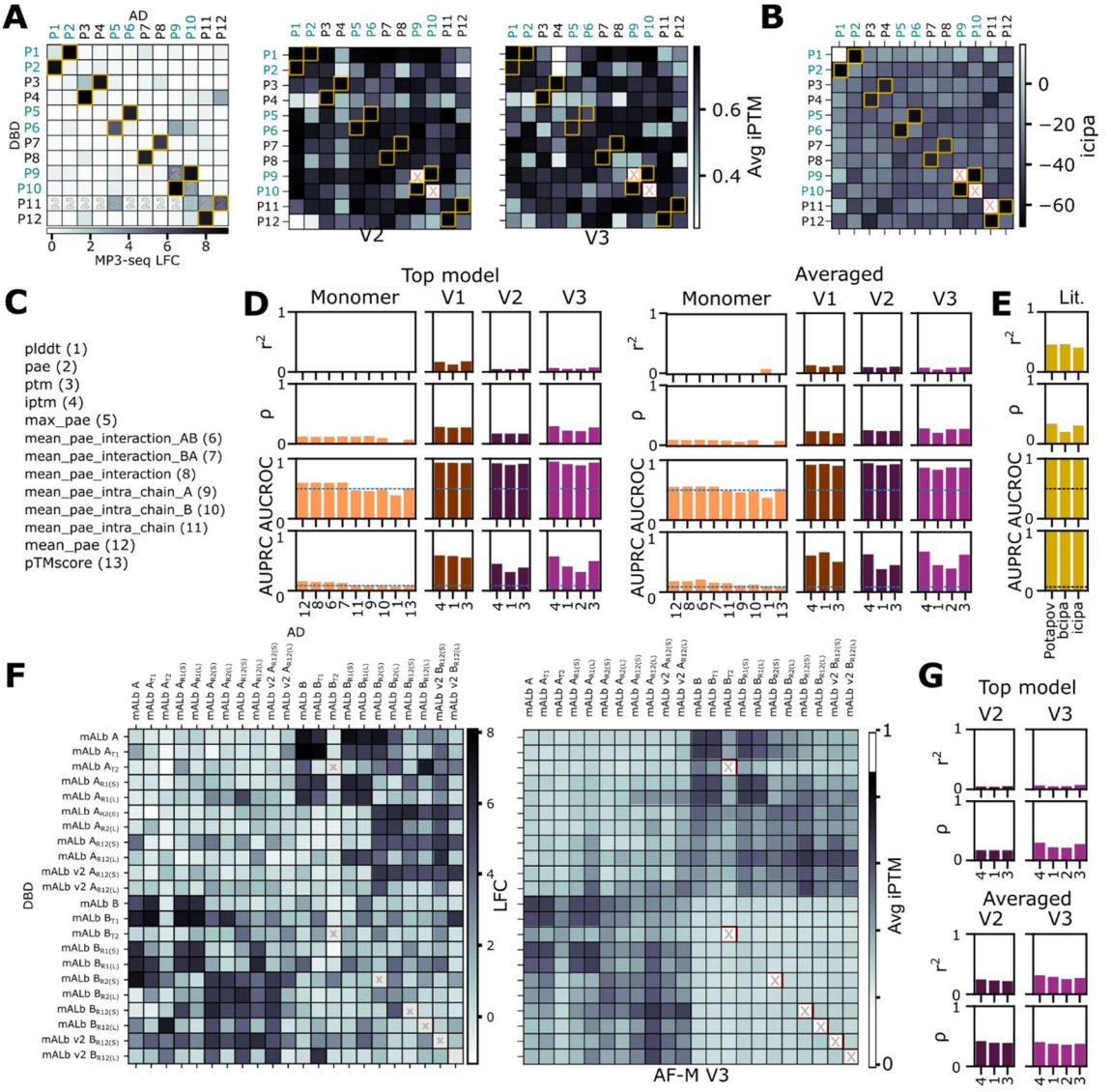


Figure 2.8. AlphaFold error metrics for the NICP and mALb interactions. A. Average iPTM for the NICP series, with LFC plot shown for comparison. B. iCipa values for NICP series. C. Complex metrics examined D. Correlation and classification of PNIC LFCs and interactions for top ranked models and averaged models. E. Correlation and classification of PNIC interactions using coiled-coil PPI predictors (Potapov:(151), bCIPA: (152), iCIPA(58)). F. LFC and average iPTM values for the mALb interactions for AF-M v3 G. Correlation of AF-M metrics with mALb series LFCs for the top rank model and averages of five models.

Following the NICP predictions, we wanted to assess the generalizability of AF2 on more complex protein heterodimers. Complexes were predicted using AF-M V2-V3 for a set of mALb coiled-coil designs based on an initial heterodimer consisting of mALb A and mALb B coil-loop-coil protomers investigated for binding determinants in MP3-seq. The AF-M v3 average iPTM and LFC values for the mALb8 interactions are shown in Figure 2.7F. Heterodimeric interactions (PPIs with A and B proteins) had higher values than homodimeric complexes (PPIs with A and A, or B and B proteins), which matched the MP3-seq measurements. Still, the overall correlation between AF metrics and LFC values was low (Figure 2.7G).

While these are useful metrics, as outlined in Chapter 1.3.2, their reliability in PPI prediction tasks has not been satisfactorily tested. Therefore, while they demonstrated some correlation with MP3-seq assay values and some classification abilities when considering the AUPRC metric (AUCROC is unreliable here due to the unbalanced interaction to non-interaction label ratio) we set out to investigate if energetic terms from Rosetta simulations and AF quality output metrics could be used to predict LFC values more accurately. Rosetta was used to collect physics-based metrics (energy of interaction, interface surface, shape complementarity, etc.) for each simulated dimer complex - the list of retained features after removing features with less than three unique values can be seen in Fig 2.8A. However, many metrics and energy terms collected were highly related (such as linear and logistic regression). Reducing the amount of highly correlated features would be beneficial to understanding the inputs needed for model performance. To do so, features were

grouped with agglomerative hierarchical clustering using Spearman's correlation coefficient with one another, and dendrograms were created with the cluster linkage values. Since the height at which features are joined together in a dendrogram is calculated based on their linkage distance, selecting features based on height thresholds is one way to choose features to include for smaller, less related input sets. To create sets of features with lower similarity, thresholds at 10% of the total height of the dendrogram were used to select feature groups until only two features remained at a threshold. A representative was selected for each group with a line intersecting the threshold (see Figure 2.8B-C). For example, the 50% threshold set for the NICP set monomer features included only the mean pLDDT and the Rosetta total score. This selection process resulted in multiple sets of features for each model version; the membership of the reduced sets can be seen in (Figure 2.8D). One interesting relationship seen in the dendrograms and selected feature sets is that the AF metrics (the bolded feature numbers in the dendrograms) cluster close together for both the NICP and mALb protein complexes, which suggests that these metrics are highly similar from a modeling perspective.

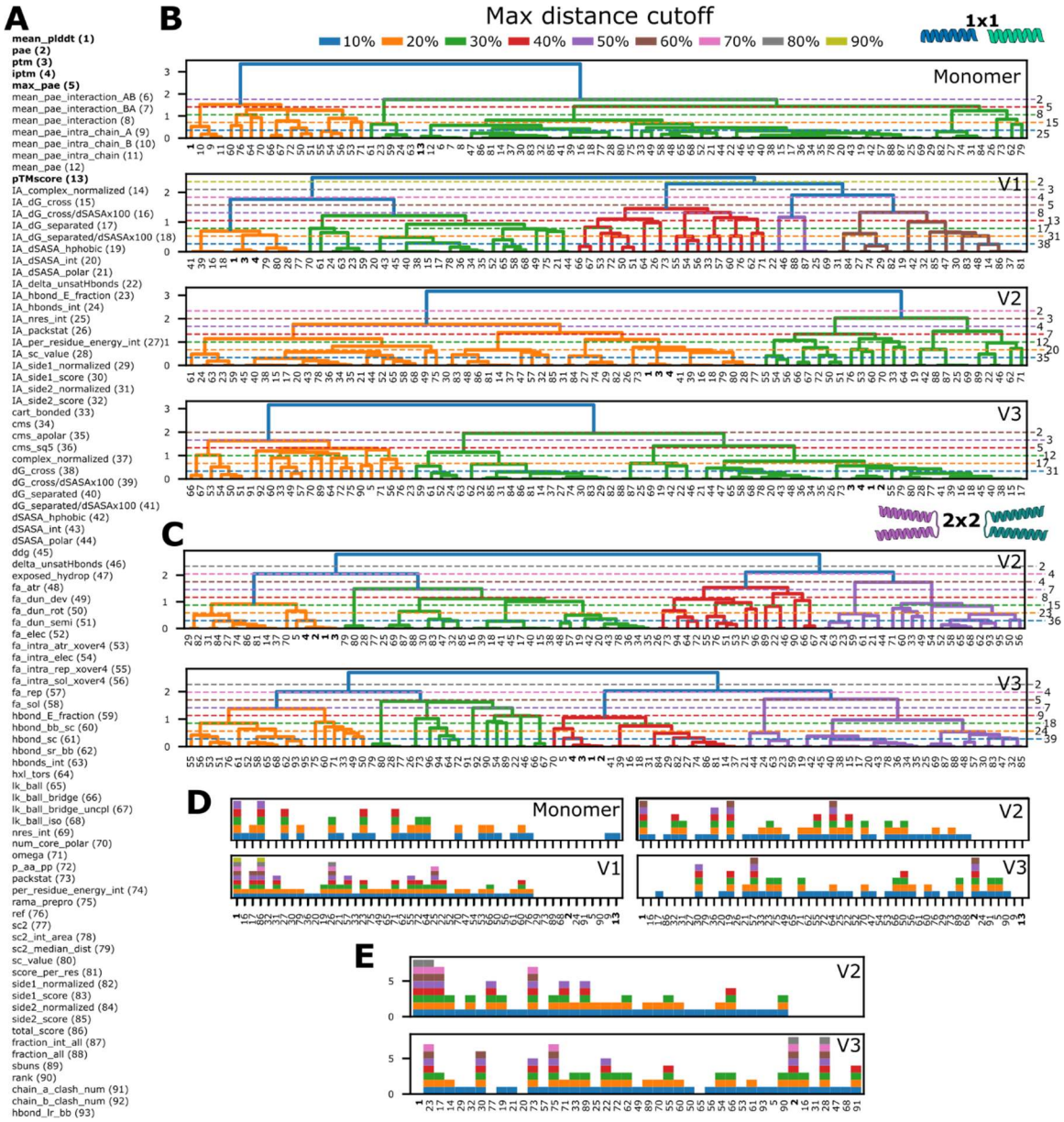


Figure 2.9. AlphaFold error metrics for the NICP and mALb interactions. A. List of all features retained from AF metrics and the Rosetta simulations. Bolded features are AF metrics. B. The dendrograms and inter-feature relationships for the NCIP proteins for all four structure predictors. C. The dendrograms and inter-feature relationships for the mALb proteins. Note that only AF-M V2 and AF-M V3 were used for these proteins. D. The membership of the actual features used in model training sets for the NCIP data. The colors correspond to the cutoffs on the dendrogram charts. E. The membership of the actual features used in model training sets for the mALb data.

Linear least square ridge regression models and logistic ridge regression classifiers were trained on all feature sets created, along with a set including only AF metrics. Two train-test splits were used: (1) p_{adj}-ranked interaction-based splits, where high-confidence interactions were held out to test the model's ability to recover missing data, and (2) protein-based splits, where all interactions involving held-out proteins were used as the test set to evaluate generalization to unseen proteins. For the first, we ranked all data points by their) p_{adj}values and partitioned the dataset into high-quality test set interactions and a mix of high-quality and low-quality training set interactions. In the other approach, a subset of proteins was selected, and all interactions involving those proteins were assigned to the test set. In particular, for the two protein sets:

- *NCIP series model training:*
  For the held-out interaction task, after ranking interactions with p_{adj} every other interaction was assigned to test until the desired test set size was reached, with the rest of the interactions assigned to the train set. Test set sizes of n=23 and n=44 were created. Five-fold cross-validation using stratified k-fold splits was used on the remaining interaction for training to explore different L2 weights and oversampling of high interactions so that those with an LFC > 5 make up the same fraction of the dataset as those below 5 was used due to the low number of strong interactions. For the held-out protein task, all 12 proteins were used as the test proteins, and all six possible designed pairs were used as test sets (P1&P2, P3&P4, …, P11&P2). The other interactions were assigned to the train set. Five-fold cross-validation was used to explore different L2 weights again. L2 weights were selected from the range 10^{-5} to 10^{5}.
- *MALb models training*
  The modeling process for the mALb proteins was generally the same as the NCIP proteins - however, the held-out interaction task used test set sizes of 43 PPIs, 84 PPIs, and 123 PPIs. These sizes were selected for comparison to the held-out protein train-test split. There were no clear pairs or groups of proteins for the held-out protein task to sample for the test sets, as the

mALb sets were not designed to be orthogonal. Instead, one, two, or three proteins were randomly drawn to make up an individual test set. Twelve samples of one, two, or three proteins were drawn uniformly from the mALb set for test sets to understand training set size effects on model performance.

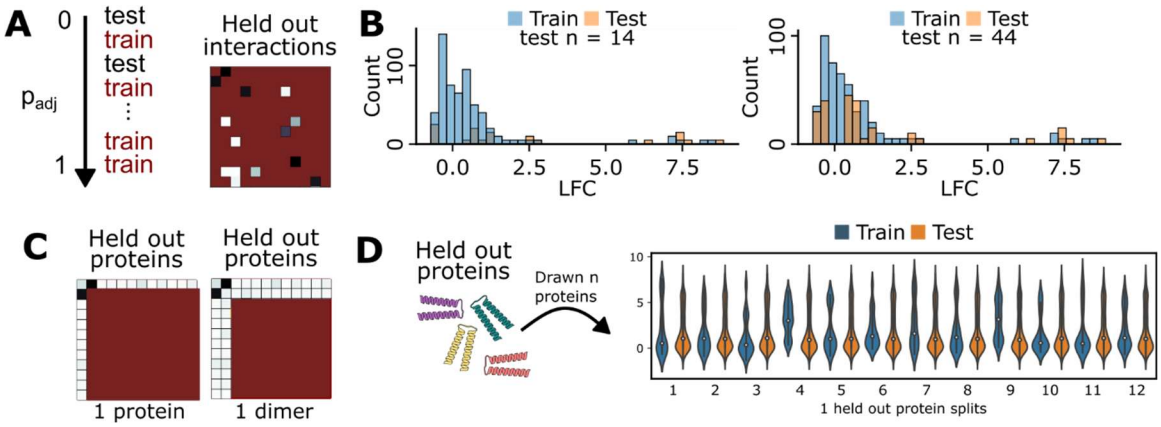


Figure 2.10. Train test split approaches for NIPC and mALb MP3-seq prediction. A. Interactions were ranked by their padj value, and then split into test and train sets for the held-out interaction approach. B. Examples of n=23 and n=44 size test sets created with the held-out interaction approach. C. Held out protein approach example splits for 1 protein or 1 dimer for the NICP proteins. D. Held out protein splits for the mALb data were created by drawing proteins uniformly at random without replacement. An example of the twelve 1 held out protein train/test sets is shown.

The NICP models reached decent regression performance for models using large numbers of features in the held-out interaction split, using AF-M V2 for the smaller test set (Figure 2.10A). However, performance dropped steeply for the larger test set, - and models in general struggled to reach the performance levels in the held-out protein task as in the held-out interaction ask for both 1 held out and 2 held out splits. Therefore, even for the extremely simple NICP interactions, simple models using these metrics struggle to learn to rank strong and weak interactions. However, logistic regression performance was extremely good, reaching literature predictor levels, for classifying on-target vs off-target interactions (Figure 2.10C). These models continued to perform well even in the held-out protein task (Figure 2,10D).

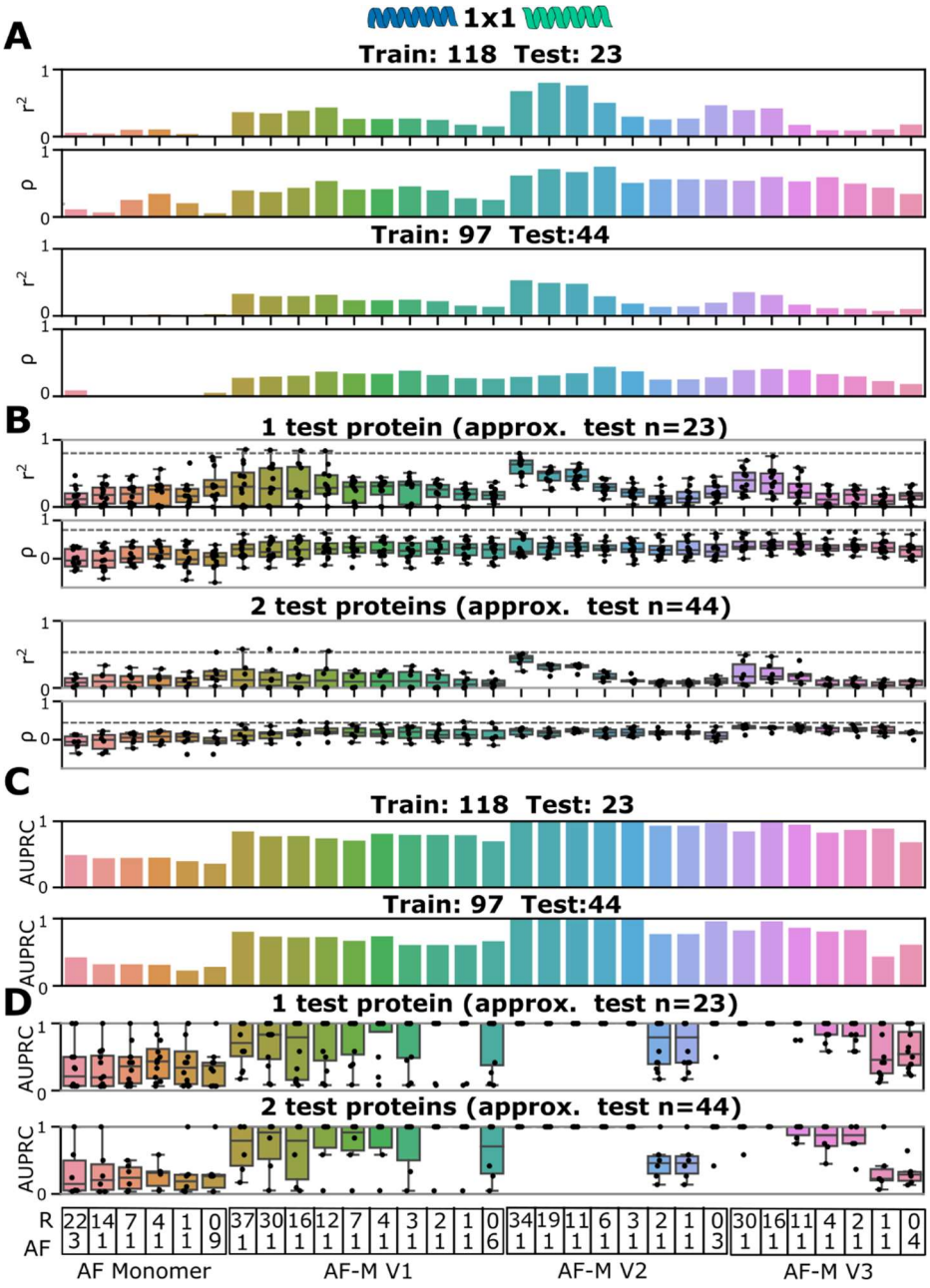


Figure 2.11. Results for the two train/test split for the NICP models. The x-axis labels shown the number of Rosetta derived (R) and AF-metric (AF) features used as model inputs for the models trained. A. Regression results, for all feature sets. B. Regression results for the held-out protein models, showing the distribution of the results for all 12 possible single protein train/test splits (top) and all six possible held out dimer train/test splits (bottom). The dashed lines correspond to the highest performing held-out interaction models on similar sized test sets C. AUPRC results for the held-out interaction classification models. D.

Classification performance for the held-out protein models for all 12 possible single protein train/test splits (top) and all six possible held out dimer train/test splits (bottom).

The mALb interactions, as noted above, lack clearly defined on- and off-target interactions. Instead, the protein set consists of either truncations (T1, T2) of the ends of the coils to investigate protein length requirements for coil-loop-coil binding, and mutations to remove hydrogen bond networks at the interface of the original mALb dimer design by substituting non-hydrogen bond compatible amino acids at key positions. Therefore, this is a more challenging protein set to attempt to predict values for both due to the structure of the dimers, and less dramatic sequence changes leading to large binding strength changes. The regression models trained to predict L-FC values had low performance for smaller test sets, but had better performance (particularly for Spearman ρ) for larger test sets (Figure 2.11A). Unlike in the NICP models, the held-out protein mALb predictors had equal or better performance in general for the single held-out protein tasks. However, for the larger held out protein sets, the held-out protein models began to perform generally worse (Figure 2.11B).

This counter-intuitive increase in performance with larger test sets, and hence smaller training sets, in the held-out interaction train/test split may be in part due to the distribution of LFC values. Due to the p_{adj}-ranking based splitting method, for smaller datasets the test set has generally higher LFC values than the train set, with the distributions becoming more similar as the test set size was increased (Figure 2.11C). This difference in distributions may also be contributing to the performance of the single-held out protein models, as the train/test distributions generated by holding out a single protein were generally similar (Figure 2.9D). At any rate, the inclusion of additional features did not result in dramatic improvements in model performance as in NICP held-out interaction regression. Overall, this suggests that though AF metrics and energetic terms have promise to classify interacting from non-interacting PPIs, the strengths of new protein interactions even for the simple alpha-coil based dimer families used here can be difficult to rank with this approach.

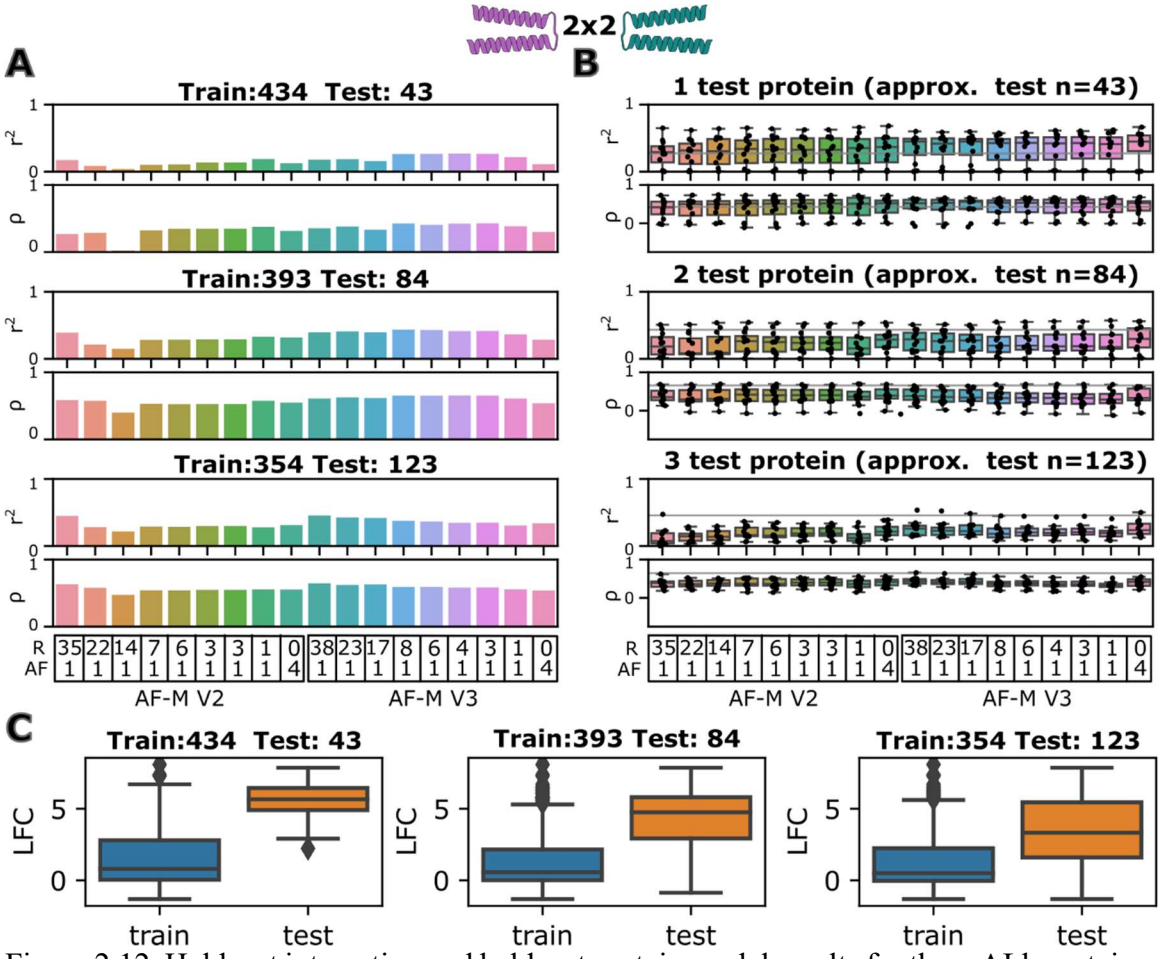


Figure 2.12. Held-out interaction and held-out protein model results for the mALb proteins. The features used as inputs in each model are shown in the boxes below the plots, where R denotes Rosetta simulation derived features and AF denotes AF-metrics. A. The results of the held-out interaction train/test split for test sets of increasing size. B. The results for the held-out protein train/test split, using 12 uniform randomly selected sets of proteins of size 1, 2, and 3 for the test set. The dashed lines on the graphs correspond to the highest performing models in corresponding approximate test-set size matches for the held-out interaction train/test splits. D. The distribution of the train/test splits for the held-out interaction task.

Chapter 3: Exploring Feature Representations for Septin-12 Protein-Protein Interaction Variant Effect Predictions

Here, a dataset of 12K septin-12 mutants was designed, and their effects on the septin-12 septin-1 interaction measured using a DMS version of MP3-seq. This interaction was then modeled using varying sizes of training sets, and their performance compared in predicting single, double, and triple mutant sequences. Finally, these models were assessed for predicting mutations at positions unseen in their training. Chronologically, this work was completed before writing and publishing the main MP3-seq paper. This was done so that the main method version could be cited in the DMS version. During the writing, modeling, and publishing work outlined in Chapter 2, this chapter's modeling section became slightly outdated – the past modeling work will be shown in this chapter and plans for updating the modeling section will be outlined in Chapter 5.

For this project, Alberto Carignano and Alex Baryshev modified the MP3-seq assay method for many vs. one mutant PPI measurement, were involved in dataset design, and conducted data collection. Additionally, Quoc Tran and Cirstyn Michel ran experiments for septin interaction data collection. My contributions consisted of designing the library of mutant sequences to measure, merging replicates, and modeling the interaction data.

3.1. Background and motivation

Understanding how missense mutations affect protein-protein interactions (PPIs) is essential for interpreting their impact on human health. Some mutations weaken interactions, while others may overly stabilize them, potentially preventing proteins from engaging with alternative partners. This challenge becomes more complex when both proteins in an interaction are mutated, or when multiple missense mutations occur in a single protein, as their cumulative effects can be non-additive due to epistasis (see Chapter 1.1.2). Predicting the effects of such complex variation is

therefore critical for connecting genomic variants to phenotypes. These predictions are also important in synthetic biology, where tuning interaction strength is key to designing effective protein binders.

The prediction of mutation effects on PPIs is part of the broader field of protein fitness prediction, where "fitness" refers to traits such as stability or binding affinity. In recent years, this field has shifted toward using protein representations derived from language models trained on large sequence databases, or unsupervised models trained on individual protein families to build supervised predictors from small datasets (86,153,154). For larger datasets, deep learning models using one-hot encodings, structural inputs, and natural language representations have been applied to fitness prediction(155). Additionally, general mutation effect predictors are often trained on datasets like SKEMPI (a large collection of $\Delta\Delta G$ datasets (156)) or the IMEX datsetes (a curated dataset of mutant effects on PPIs (157)) However, questions remain about their ability to generalize to mutations at unobserved positions (158,159). With advances in protein language models (pLMs), evolutionary representation models, and accurate structural predictors, there is growing potential to identify which types or combinations of protein representations are most effective for predicting the effects of missense mutations on PPIs. Indeed, many current approaches combine features across domains to make predictions (113,160–162).

3.1.1. Protein variant effect predictors

Many general variant effect predictors have been developed to predict if a variant has a significant effect on protein function. Older general predictors used some combination of sequence conservation, biophysical amino acid properties, and structural features to produce said scores (163,164). These general effect predictors are designed to predict single mutant effects for any protein and substitution pair, like SIFT (165) and PolyPhen (166). Such generic models have begun to be surpassed by unsupervised models that attempt to learn values based on a single family's evolutionary conservation. Evolutionary conservation-based models use MSAs to attempt to learn representations of a families conservation, through

approaches like training deep generative models on the MSA sequences (167), or simple probabilistic graphical models (168). One flaw with these MSA-trained coevolutionary models is that their performance depends on possessing a high-quality MSA for the target protein, which may not always be available.

To predict specific variant effects, such as variant effects on individual PPIs, supervised models, including CNNs, RNNs, and random forests, are trained on labeled mutation data. In low-data regimes, averaging embeddings from pLMs has proven useful (86), with follow-up work showing that combining one-hot encodings and MSA-derived likelihoods can outperform simpler approaches on multiple DMS benchmarks (154). More complex models, combine data representations from pLM and evolutionary sources, and have reached good performances. There are many DMS databases to aid with developing these types of predictors, and there have been multiple reviews on common approaches to train predictors for targeted variant effect prediction problems and benchmarking effect predictors (163,164,169,170).

3.1.2. The septin protein family

Septins are a conserved family of GTP-binding, membrane-associated proteins present in most eukaryotes. They play critical roles in cell structure, division, and organization. Mutations or deletions in septin genes have been linked to health issues ranging from infertility to neuromuscular disorders (171,172). The 13 human septins are grouped into four groups based on sequence homology and domain structure: SEPT2 (septin-1, -2, -4, -5), SEPT3 (septin-3, -9, -12), SEPT6 (septin-6, -8, -10, -11, -14), and SEPT7 (septin-7) (173). Septins form heteromeric complexes—typically hexamers or octamers—which further polymerize into filaments. These assemblies are stabilized via two GTPase domain interfaces, G and NC, that mediate septin–septin PPIs (174,175). Canonical human septin complexes include SEPT7–SEPT6–SEPT2–SEPT2–SEPT6–SEPT7 (hexamer) or SEPT2–SEPT6–SEPT7–SEPT9–SEPT9–SEPT7–SEPT6–SEPT2 (octamer), although alternative binding arrangements have been proposed (176,177).

While SEPT2, SEPT6, and SEPT7 family members contain coiled-coil domains, SEPT3 group proteins do not (177,178). These proteins—expressed primarily in the brain (septin-3), testis (septin-12), and broadly throughout the body (septin-9)—participate only in the octameric complex. Crystal structure studies of SEPT3 family proteins reveal conserved secondary structures at G and NC interfaces, which are critical for filament assembly (177). Functional impacts of missense mutations near these interfaces were assessed using Y2H assays; in septin-12, eight of nine tested mutations disrupted binding, highlighting the importance of interface integrity for septin function (179).

3.2. DMS dataset design

To assess thousands of mutant interactions in parallel, we integrated a designed library into the MP3-Seq (140)plasmid, replacing a segment of the wild-type protein. Septin-12 was chosen as a test case for validating the DMS MP3-Seq assay, due to its health relevance and known mutation-sensitive interfaces previously characterized by Y2H (179). A 61-amino acid region (V162–V222)—the containing multiple mutations from (179)—was targeted for mutation. Rather than using random mutagenesis, we constructed a focused library comprising all single amino acid substitutions, a subset of triple mutants for model benchmarking, and the remaining slots filled with double mutants, totaling 12,000 sequences. Mutations were enriched at known interface residues and filtered to avoid highly destabilizing changes, as Y2H cannot distinguish between interaction loss due to misfolding versus disruption of binding.

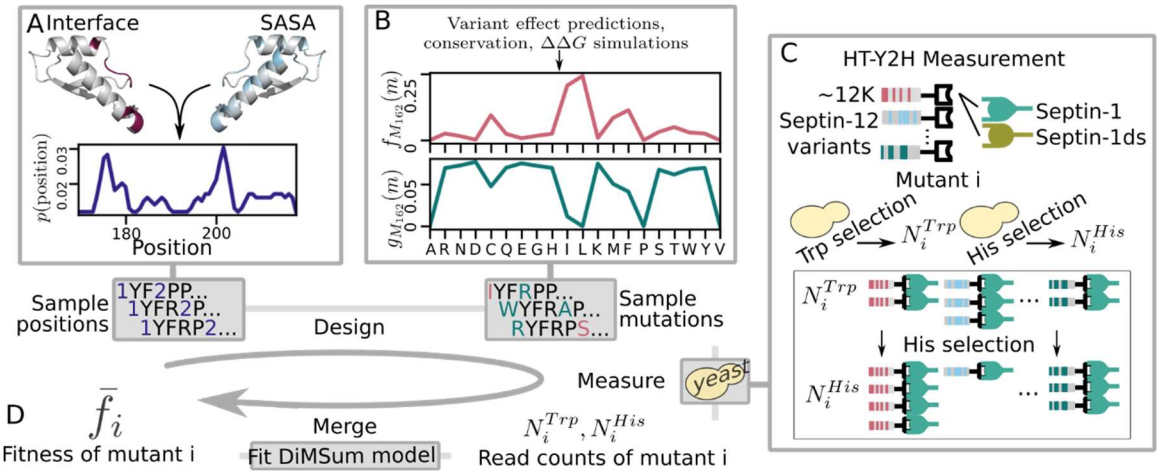


Figure 3.1. Dataset design and collection. A. Mutant dataset design by sampling from positions 162-222 in septin-12 using smoothed interface and SASA values as sampling weights B. Mutation substitutions for each sampled position are selected from either a tolerated (pink) or intolerated (green) distribution. Shown distributions are for V162. C. Mutants are measured using a HT-Y2H assay. D. Finally, TRP (called HIS+ in MP3-seq) and HIS- read counts are merged with the DiMSum (180) package to get a fitness value for each successfully recovered mutant per PPI

To guide mutant library design toward both functional interface coverage and assay-compatibility, heuristic sampling distributions were developed. Substitution sampling weights were constructed by integrating predictions from multiple variant effect predictors, a coevolutionary model, ΔΔG simulations, and a position weight matrix (PWM) derived from the septin-12 MSA. For each position, two substitution distributions were created—one favoring "tolerated" mutations and one favoring "intolerated" mutations—excluding the most damaging substitutions per site. A separate positional sampling distribution across residues 162–222 was generated using surface accessibility and predicted interface contacts from homology models (Figure 3.1A–B). To ensure even mutational coverage, four double mutants were allocated per pairwise position combination. Remaining double and triple mutants were sampled using the positional and tolerated/intolerated substitution distributions. For 5% of double mutants, previously excluded substitutions were reintroduced to test extreme cases.

The final designed library was assayed using the HT-Y2H method (Chapter 2) against wild-type binding partners septin-1 and its splice isoform septin-1DS. Unlike the normal MP3-seq plasmid assemblies which contain both interacting proteins, this approach used full-length septin-1 or septin-1DS was included the designed library of mutant sequences which recombined with wild-type septin-12 fragments during assembly (Figure 3.1C). Two biological replicates were performed for each interaction pair. Barcode fitness values from HIS+ and HIS- reporters were quantified using DiMSUM (180) (Figure 3.1D). Of the 12,000 designed mutants, we recovered 1158/1159 single, 9604/10241 double, and 535/600 triple mutants for the septin-12–septin-1 interaction.

3.3. Assay results

Fitness measurements for septin-12 mutants showed minimal differences when interacting with either septin-1 (S1) or its splice isoform septin-1DS (S1DS). This result is expected, as the only difference between S1 and S1DS is an additional exon in S1DS that extends a predicted disordered region (Figure 3.2A). The correlation (r^2) between merged replicate fitness values for S1 and S1DS interactions was 0.85. Replicate fitness correlations before merging were 0.64 for S1 and 0.85 for S1DS. Both interaction assays revealed a clear bimodal distribution of variant fitness, separating near-wild-type from low-fitness mutants (Figures 3.2B–D). Single mutant effects across positions were highly similar between S1 and S1DS (Figure 3.2B, top). Certain interface positions tolerated substitutions well, while others were highly intolerant, consistent with structural models where core interface residues are more constrained than peripheral ones (181). Notably, P174 and H170—both located within the conserved alpha-2 helix of SEPT9-family septins—were surprisingly substitution-tolerant, suggesting potential flexibility even within structurally critical regions (177). The interaction interface can be seen colored by per-position fitness, and a close-up of the mutation region, can be seen in Figure 3.3E-F.

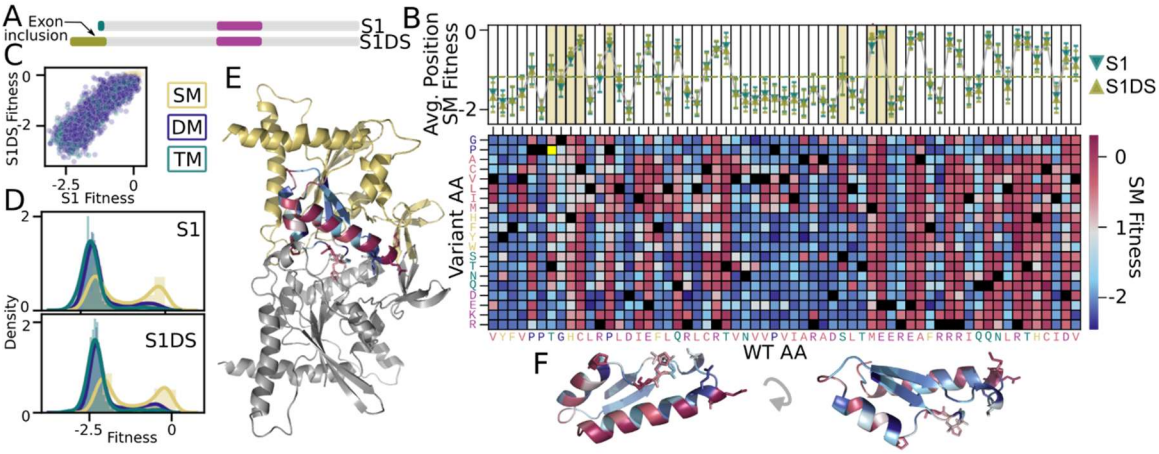


Figure 3.2. PPI dataset visualization A S1 and S1DS structure B. S1 and S1DS fitness distributions by mutant type C. S1 and S1DS fitness agreement, $r^2 = 0.85$ D. Single mutant fitness values for all possible AA substitutions at positions 162-222 in septin-12. Black boxed are WT positions, yellow is unrecovered mutant. Top plot is the average of the position fitness values for both S1 and S1DS, where shaded boxes are interface positions and the dashed line is the average of all single mutant fitness values E. septin-12 (yellow) and S1 (teal) homology model, mutation region is colored blue/red according to per-position fitness values F. Close ups of mutated region, colored by average single mutant fitness values

3.4. Correlation with generic and evolutionary variant effect predictors

Fitness measurements were correlated with variant effect predictors to provide performance baselines for modeling (Fig 3.3). EVE (182) evolutionary indices gave the highest correlation for single mutants in (Spearman ρ =0.76) (Fig 3.3B). These values are not outside the typical correlation range seen for single mutant fitness correlations with other DMS datasets (182). Since most variant effect predictors are not designed to evaluate multi-mutant sequences, their performance on double and triple mutants was approximated by summing the predictor scores of the corresponding single mutants. An exception is EVE, which uses a variational autoencoder (VAE) that can be adapted to evaluate full-sequence likelihoods, though the original EVE study did not report performance on multi-mutant prediction(182). As an additional baseline for multi-mutant modeling, we also included a naïve additive predictor that sums experimentally measured fitness values of the

constituent single mutations—representing an optimistic lower bound that supervised models must exceed to be useful in predicting septin-12 variant effects (Figure 3.3).

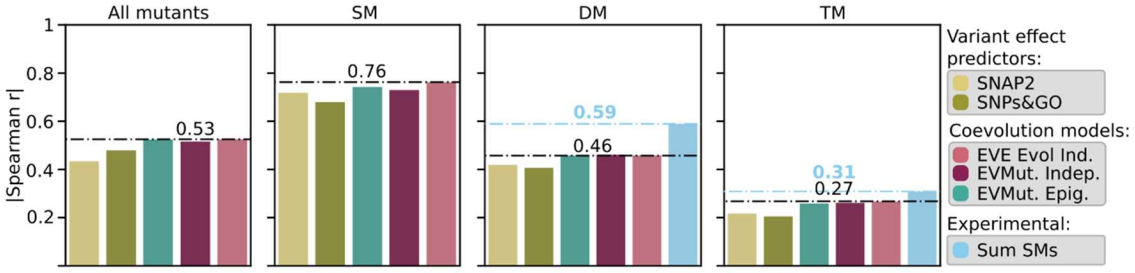


Figure 3.3. Baselines for fitness prediction Baselines for mutant fitness correlation with outputs from variant effect predictors SNAP2 (183) and SNPs&Go (184), coevolution models EVE evolutionary index (182) and EvMutation (168), and summing single mutant fitness. The highest Spearman r of the prediction methods is shown with the black line, while the sum of the single mutants is shown in blue.

3.5. Supervised fitness prediction

Five different protein sequence representations commonly used in fitness prediction literature were evaluated for modeling the septin-12–S1 protein–protein interaction (PPI) (Figure 3.4A). The first, and most widely used, representation was simple one-hot (1H) encoding of the 61-residue mutated region, resulting in a 61×20 binary matrix (or a 1220-dimensional vector when flattened). One-hot encoded sequences were also input to pretrained protein language models (pLMs), where the final hidden layer was extracted either as a latent matrix (LMA) representation or averaged along the sequence length to produce a latent mean embedding (LME). For LMA inputs, we used two pLMs: the MuPIPR model from (185), designed for general mutation effect prediction on PPI stability, and the Bepler model from the BioEmbeddings framework (153), yielding latent dimensions of 128 and 121, respectively. For LME inputs, we used the updated ESM-1b model (1), taking the average of the final hidden layer across all positions in the full mutant sequence to obtain a 1024-dimensional vector. To incorporate an evolutionary representation, a multiple

sequence alignment (MSA) for septin-12 was constructed following the methods of (167), and an EVE model was trained as described in (182). Latent space vectors (LVs) were extracted for all mutant sequences, and their log-likelihoods under the EVE model were computed. Following (154), these EVE log-likelihood scores were concatenated with the one-hot (1H) encoded sequences to form the final P1H input representation used in model training. The different inputs can be seen in Figure 3.4A.

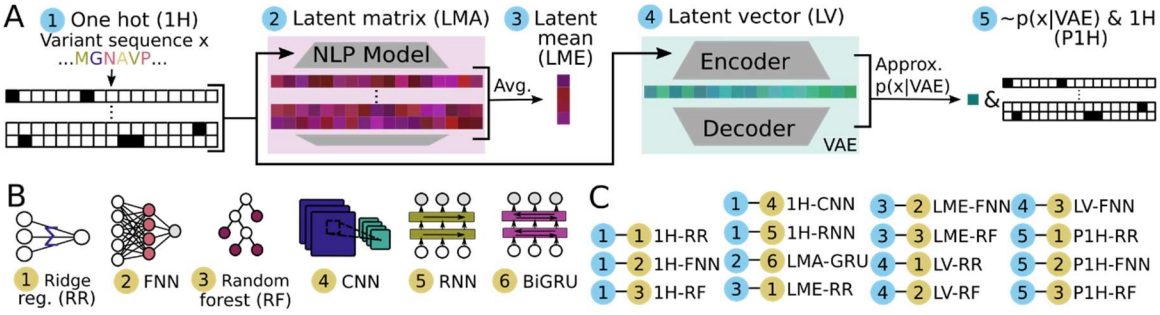


Figure 3.4. Fitness predictors attempted and results A. All input representations evaluated for protein sequences B. All ML model types attempted for fitness prediction C.. Combinations of inputs and model types attempted

For fitness prediction modeling, we tested several architectures: L2-regularized ridge regression (RR), multi-layer feed-forward neural networks (FNN), random forests (RF), convolutional neural networks (CNN), simple recurrent neural networks (RNN), and bidirectional gated recurrent unit networks (BiGRU), depending on the input representation (Figure 3.4B). Vector-based inputs—namely, 1H, LME, LV, and P1H representations—were used to train RR, FNN, and RF approaches (Figure 3.4C). For matrix-form inputs, we trained CNN and RNN models on the 1H representation and BiGRUs on the LMA representation. While CNNs have previously been applied to LMA-type embeddings, we chose BiGRUs due to the sequential and contextual nature of the LMA representation, consistent with prior work such as (186).

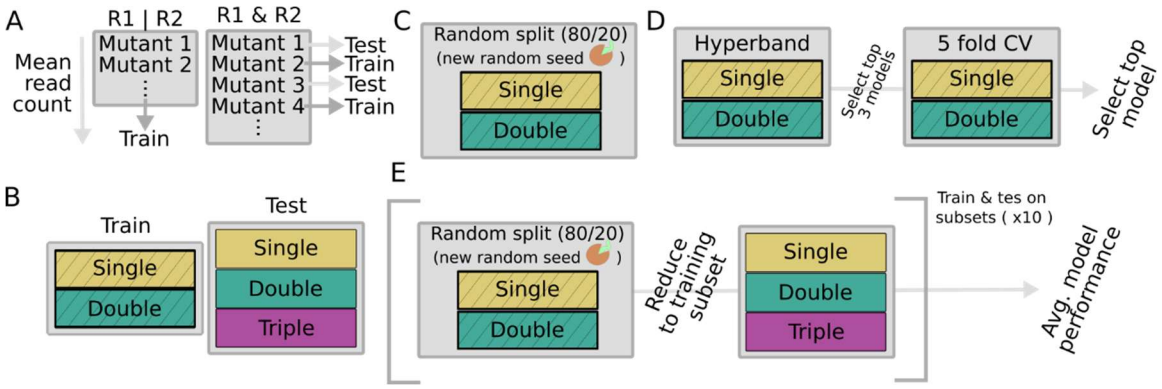


Figure 3.5. Quality aware train/test splitting, and employed modeling strategies. A) Dataset train-test split procedure. Mutants that were in a single replicate were used for training, and mutants that were in both replicates were ranked by average starting HIS- read count, and every other mutant was assigned to the test set until the goal number of double and single test mutants was reached. The remaining mutants were assigned to the train set B) The train set was composed of single and double mutants, while the test set included all triple mutants and high read count single and double mutants C) For RR and RF, a single stratified k-fold split and random search was used for hyperparameter optimization D) For FNN, CNN, RNN, and BiGRU models, Hyperband tuning was used with a single random split, and the top 3 models from Hyperband were evaluated with 5-fold CV on the train data from the same random split to select a top model for the input/model combination E) To evaluate models, 10 random splits were made of the training data, and models were trained on subsets of the data. Performance was averaged for the 10 seeds

There are multiple strategies for train–test splitting in protein fitness prediction tasks(153). In our study, all triple mutants were held out as a test set by design to evaluate model performance on higher-order mutation prediction. For the remaining single and double mutants, we opted against random partitioning and instead designed a split based on data quality. This strategy leverages the inherent variability of biological assays by reserving the highest-confidence data for testing, providing a more meaningful assessment of model generalization. Similar quality-based partitioning has been applied successfully in sequencing studies (75). Because some mutants appeared in only one replicate, and these single-replicate observations generally corresponded to lower-fitness variants, we adopted a quality-aware strategy train/test split strategy. All mutants present in only one replicate were used for training. Among mutants present in both replicates, we ranked them by HIS- read

count and selected every other high-confidence variant for inclusion in the test set (Figure 3.5A).

To efficiently explore hyperparameters across the fifteen input–model combinations, three tuning strategies were employed. For RR models, multiple regularization strengths were tested, and the optimal value was selected based on performance on validation data specific to each RR variant. For RF models a randomized grid search to identify suitable hyperparameters. RR and RF models used a single 80/20 train–validation split (Figure 3.5C). For all deep learning models, FNN,CNN,RNN, and BiGRU, Hyperband tuning (187) was used with an 80/20 train–validation split to perform large-scale hyperparameter exploration. Hyperband is a resource-efficient tuning algorithm that combines random search with early stopping, allowing many configurations to be tested with varying training durations. Following the Hyperband search, the top three model configurations were retrained using 5-fold cross-validation, and the best-performing model was selected for final evaluation on the held-out test sets (Figure 3.5D-E).

For model evaluation, fitness prediction was assessed for single, double, and triple mutants across increasing training set sizes using ten independent 80/20 train–validation splits, with performance averaged per input–model combination. The top-performing model for each input type at training set sizes of $N = 1{,}723$ and $N = 8{,}609$ mutants is shown in Fig. 3.6, alongside the best-performing unsupervised baselines from Figure 3.3. With small training sets, models incorporating evolutionary likelihoods or language model-derived representations outperform others in both Spearman ρ and mean squared error (MSE) for single and double mutant fitness prediction. Notably, the LMA-BiGRU model demonstrates superior performance in ranking triple mutant fitness, achieving a Spearman ρ of 0.38 ± 0.02 with the smaller training set, which improves to 0.45 ± 0.03 with the larger set. For single and double mutant prediction, the best models were P1H-FNN and LMA-BiGRU (Figure 3.6).

While CNN and RNN models show performance gains with increased training data, the vector representation models (e.g., RR, FNN, RF) exhibit only marginal improvements, suggesting early performance saturation. Indeed, input–model combinations resembling low-N strategies (e.g., P1H-RR) show rapid gains with small datasets but quickly plateau with additional data. Furthermore, in the low-data regime, these models perform only slightly better than simply using unsupervised EVE evolutionary index scores. Taken together, these results indicate that leveraging latent matrix representations from language models, particularly the LMA-BiGRU, are the most effective for generalizing to higher-order mutant fitness prediction, including extrapolation to triple mutants.

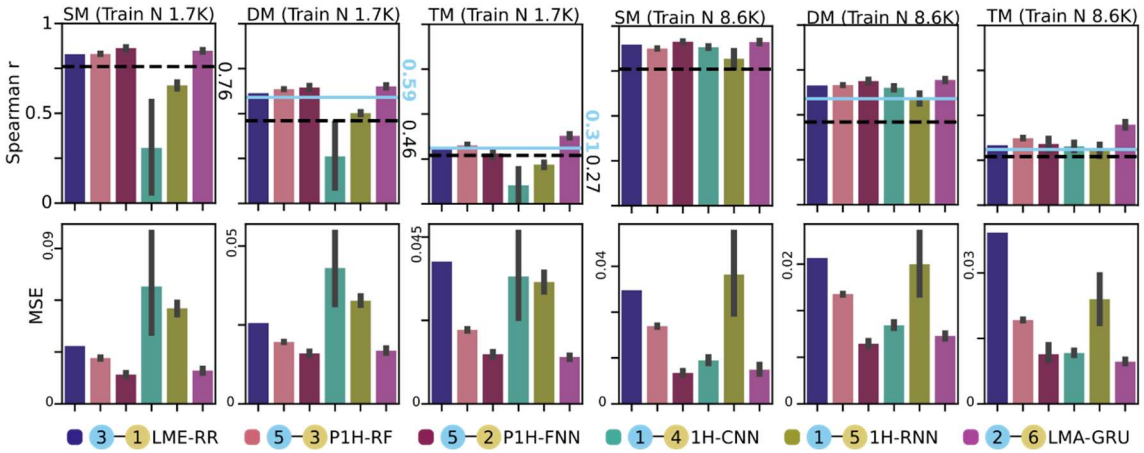


Figure 3.6. Model performance for two of the training set sizes. (Top) Spearman ρ of predicted vs. true fitness for the best performing models from each sub-type, trained on single and double mutants (Bottom) MSE of predicted and true fitness values for the best performing models of each subtype.

3.6. Comparisons with allele frequencies, low-throughput experiments, and literature values

A desirable property of fitness predictors is the ability to reliably rank the effects of higher-order mutations and generalize to variants that differ substantially from the training set. In particular, predictors capable of generalizing to previously unseen regions could prove valuable for interpreting the functional consequences of human

genetic variants. Missense mutations in septin-12 cataloged in the Genome Aggregation Database (gnomAD), a large international compendium of exome and genome sequencing data, are shown in Figure 3.7A (188). While most are classified as variants of uncertain significance, two are known to be pathogenic and linked to male infertility. The T89M mutation is known to impair GTPase activity and is believed to disrupt septin filament formation by interfering with protein–protein interactions (PPIs) (172). Similarly, D197N disrupts both GTP binding and the septin-12–septin-1 interaction (172). D197N was one of eight septin-12 mutations, spanning a range of allele frequencies, whose interaction effects were previously measured using Y2H assays (179). Six of these eight mutants fall within the 61-amino acid region we targeted in our study (Figure 3.7B–C). Among these, five were found to disrupt septin-12 interactions with septin-1, -5, and -7, while one had no detectable effect. Notably, V169E and D197N were shown to disrupt PPIs without affecting protein stability (172,179). Fitness values from our MP3-seq assay for these six clinically relevant mutants are shown in Figure 3.7B. To verify the reproducibility of our high-throughput measurements, a low-replicate, plate-based Y2H assay was performed in parallel (Figure 3.7C), confirming the disruptive nature of these variants and demonstrating consistency between throughput scales.

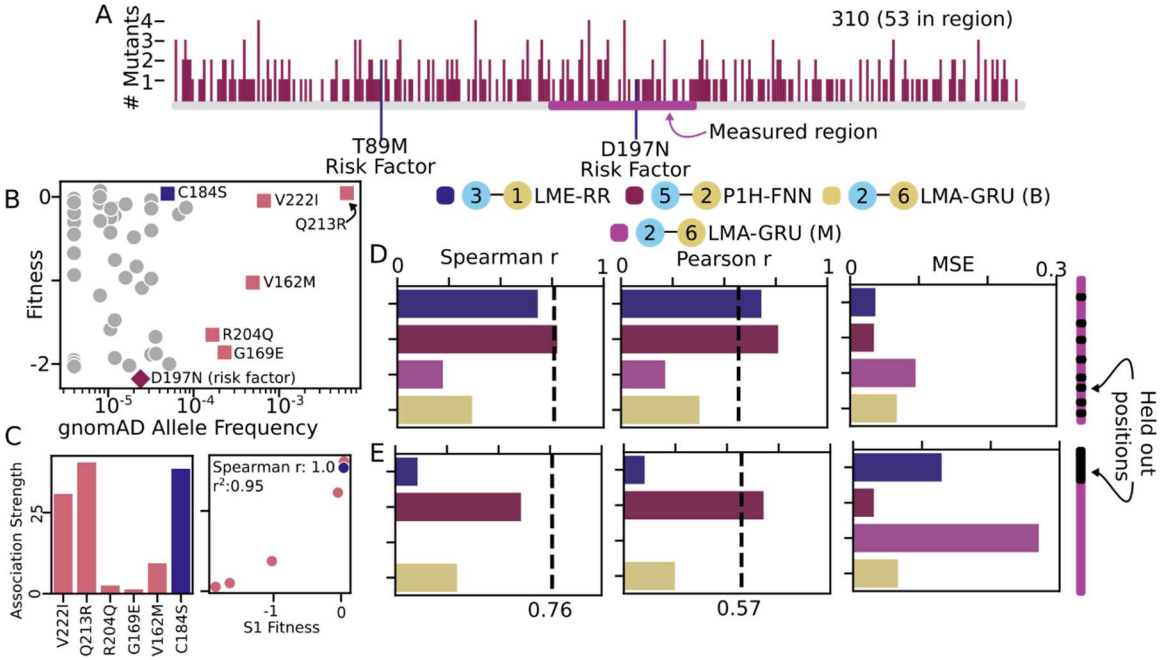


Figure 3.7. Human septin-12 variants and positional extrapolation performance A) All gnomAD missense variants for septin-12, by location and number B) gnomAD allele frequency for human variants in the measured region vs. S1 fitness, red is a known disease associated variant, pink squares are Fragoza disruptive variants, blue square is Fragoza nondisruptive variant C) Low throughput results (left) of the Fragoza septin 12 mutants, correlation of low throughput and HT-Y2H S1 fitness D) Performance of top models on with test sets comprised of held out positions, where 8 positions were randomly selected from 162-222 (top) or from the end of the measured region (bottom) to be held out for testing. The randomly selected positions were 166, 173, 192, 198 204, 213, 214, and 217.

3.7. Predictions in held-out regions of mutations

The LMA and LME-based predictors trained above have the potential to predict the effects of mutations across the entire septin-12 sequence, as these models are trained using representations of the full protein. However, before applying these predictors to unmeasured regions, it is important to evaluate their ability to generalize to unseen positions rather than just unseen mutations. To this end, we modified the train/test paradigm used in earlier evaluations. Two new test sets were constructed, each comprising eight held-out positions: in one, eight random positions within the measured region were selected; in the other, the first eight positions in the assayed window (residues 162–168) were held out. Any sequence containing a substitution at one of the selected positions was assigned to the test set, while the remaining sequences were used for training.

A subset of the best-performing models from each category was retrained on the new splits, and their performance on single mutant predictions at held-out positions was evaluated (Figure 3.7D–E). All models exhibited a substantial drop in predictive performance, even for randomly selected held-out positions. Performance was further degraded when the held-out positions were contiguous, suggesting difficulty extrapolating to unobserved sequence segments. This limitation in generalization to new positions has been observed in other deep mutational scanning modeling studies (158,189). These results suggest that applying predictors trained only on the measured 61-amino acid window—or trained on low-quality or incomplete datasets—

to make predictions across the full length of septin 12, including unmeasured positions, would likely yield unreliable results. Also worth noting is that the LMA-BiGRU model trained with the Bepler (153) representations has similar performance on the held-out position single mutants for randomly selected positions and the contiguous held-out positions at the end of the selection (Fig 5E). This may be because the Bepler representation has a signal spread out in the matrix when examining the difference between mutant and wild-type representations, while the MuPIPR LMA (1) has differences to wild-type localized around the mutated window.

Chapter 4: Interpreting neural networks for biological sequences by learning stochastic masks

Understanding how and why neural networks arrive at their predictions is challenging, largely due to their complex structures and nonlinearities. Model interpretation is the subfield focused on understanding what has been learned by a trained model: it spans from understanding the model's global behavior to the specifics when making individual predictions. One popular method of interpreting more complex models, such as neural networks, is feature attribution, where scores are assigned to input features to represent their final 'contribution' to a model's end prediction. Global behavior patterns can often be drawn from examining feature attributes on a dataset scale. In sequence-to-function prediction, model interpretability and feature attribution techniques are key to connecting predictions to established biology, learning new regulatory rules, and validating sequence designs.

In published work where I am the second author, I worked closely with Johannes Linder to develop and test a feature attribution method developed specifically for biological sequence inputs. This work involved creating complex interpretation tasks involving recovering interdependent feature sets. My main contributions to the paper were the development of an upstream open reading frame attribution task, a permutation-test $\Delta\Delta G$ and HBNet recovery task for dimer prediction, and recovering Rosetta energy metrics for structural predictions to compare against structure attribution scores.

4.1. Feature attribution background for one-hot encoded sequence predictors

Methods for interpreting genomic regulatory predictors have been reviewed elsewhere (190–192), but some relevant background approaches are summarized here. A basic model interpretation approach for CNNs is visualizing what filters have

learned. The weights of first-layer filters can be visualized as sequence motifs similar to position weight matrices (PWMs), which sometimes correspond to known cis-regulatory elements (70). This approach has been widely used to generate motif-like representations from MPRA-trained CNNs (73,76,193,194). Extensions to deeper layers have revealed combinations of motifs (74), and architectural modifications can encourage shallow filters to be more interpretable (195–197).

However, understanding regulatory logic requires more than just motif identification. Nonlinear interactions between sequence features are common in biology, making it necessary to consider feature combinations and their context rather than treating features independently or observing filter weights for individual motifs. Neural network interpretation techniques—such as gradient-based methods, in silico mutagenesis, and SHAP—that assign importance scores to input features can be used to identify relevant elements across a sequence (198–201). These methods are particularly helpful when comparing wild-type and variant sequence predictions, revealing which features contribute to a change in predicted function (202).

Nevertheless, many attribution methods rely on local perturbations and may fail to capture nonlinear dependencies. For example, 5′ untranslated regions (UTRs) can contain multiple upstream start codons (uAUGs)(203–205) before the primary start. Which AUG is chosen as the start of translation depends on how much the surrounding context represses its usage. Two nearby uAUGs may "hide" each other from methods using local gradient approximation, as each AUG can repress translation initiation at the primary start independently. Saturation mutagenesis, which systematically exchanges one nucleotide in the sequence and approximates its importance by change in prediction, also struggles with such nonlinearities. With two uAUGs, local methods and saturation mutagenesis could incorrectly conclude that neither uAUG would be repressive, as knocking down only one would not change the prediction (206,207) To address this limitation, model interpretation methods focusing on feature relationships across entire datasets have been developed. One such method is TF-MoDISco, which identifies recurring patterns in attribution scores

to reveal context-aware motif usage (208). These derived motifs can then be analyzed for cooperative or distance-dependent effects (199,209). For instance, De Almeida et al. (210) used TF-MoDISco with a CNN trained on fly enhancer activity to identify developmental and housekeeping enhancer motifs and experimentally validated their positional dependencies using an MPRA.

A common feature attribution method relevant to the work in (141) is mask-based methods, sometimes called removal-based methods. These involve creating an input 'mask', which is used to remove subsets of input features and summarize how each removed feature affects the model predictions (192,211). Mask-based attribution methods prior to Scramblers can be seen in Table 4.1, along with a breakdown of the characteristics that differentiate their approaches. The most important characteristics of a masking method are the form masked values take, the optimization algorithm for creating masks, and whether the original model is being interrogated or if a surrogate model must be trained. Mask-based methods originated in computer vision, with early methods masking inputs by fading or blurring features. While suited for visual inputs, fading and blurring nucleotide and protein sequences is inappropriate given their limited alphabets. Similarly, some masking methods Pre-Scrambler replaced masked values with zeros, which is typically not in-distribution for sequence predictors that expect discrete one-hot encoded patterns as inputs. This out of distribution effect often necessitates training a new predictor model capable of handling zeros to try and reconstruct the original predictions (201,212). Finally, masked positions can be replaced with samples from the marginal distribution of that input, typically calculated from part of the training dataset. Alternatively, it can be replaced with counterfactual samples generated in another manner, such as a generative network as in (213). Many of the methods in Table 4.1 use per-example methods, such as per-example sampling. This is less efficient than parametric mask generation methods, which can be trained once and used for many attributions afterward.

Method	Optimization	Mask type	Task	Predictor
Scramblers (141)	Parametric model	Samples	C,B	Existing
Fong et al. (Torchray) (214)	Per-example SGD	Faded, Blurred	C	Existing
Dabkowski (Saliency) (215)	Parametric model	Faded, Blurred	C	Existing
Yoon (INVASE) (216)	Parametric model	Zero	C,N	New
Chen et al (L2X) (212)	Parametric model	Zero	C,N	New
Carter et al (SIS) (201)	Per-example recursion	Mean, samples	C,N,B	Existing
Zintgraf (217)	Per-example sampling	Samples	C	Existing
Chang (213)	Per-example SGD	Counterfactual	C	Existing

Table 4.1. Overview of relevant masking-based feature attribution method. Optimization refers to how masks are constructed (made with a parametric model, per-example approach). Mask refers to how masked features are replaces, or what they are replaced with. Task refers to what task(s) the method was demonstrated on (C = computer vision, N = natural language processing, B = biology). Predictor indicates if the method interpreted an existing predictor, or if a new predictor was trained as part of the attribution method.

4.2. Scrambler networks and the inclusion and occlusion objectives

Scrambler neural networks are the mask-based feature attribution described in (141). As the majority of my contributions to this paper were in developing meaningful metrics for evaluating the function of predictors based on the known biology of their prediction subject, I will only give an overview of scramblers here. In brief, given a differentiable pre-trained predictor $\mathcal{P}$ which takes in a one-hot encoded sequence input $x \in \{0,1\}^{(N \times M)}$ the goal of the network, $\mathcal{S}$, is to learn to output a set of real-values importance scores $\mathcal{S}(x) \in (0, \infty]^N$ such that when these scores can be used to produce a PSSM-like probability distribution $\widehat{x_{\mathcal{S}}}$ given in Equation 4.1 where $\tilde{b} \in$

$\mathbb{R}^{N\times M}$ is background distribution, σ is the SoftMax equation $\sigma(l)_{ij} = \frac{e^{l_{ij}}}{\sum_{k=1}^{M} e^{l_{ik}}}$, and $\dot{\mathcal{S}}(x) \in (0,\infty]^{N\times M}$ are the importance scores $\mathcal{S}(x)$ broadcast at position i to all channels j.

$$\widehat{x_{\mathcal{S}}} = \sigma\left(\log \tilde{b} + x \times \dot{\mathcal{S}}(x)\right) \quad (4.1)$$

In practice, $\mathcal{S}$ was a residual network with dilated convolutions, and $\tilde{b}$ was taken to be the mean input pattern across the training set of $\mathcal{P}$. When drawing samples from $\widehat{x_{\mathcal{S}}}$, if $\mathcal{S}(\mathrm{x})_{\mathrm{i}}$is close to 0, $\widehat{\mathrm{x}_{\mathcal{S},\mathrm{i}}}$ becomes $\tilde{\mathrm{b}}_{\mathrm{i}}$ (the background distribution at i), and when $\mathcal{S}(\boldsymbol{x})_i$ is close to ∞, $\widehat{\boldsymbol{x}_{\mathcal{S},i}}$ becomes x_{i} (the original input at i).

To train $\mathcal{S}$ for one of the formulations of scramblers, inclusion, K discrete samples would then be drawn from $\widehat{\boldsymbol{x}_{\mathcal{S}}}$. These samples would be given to the original predictor, to get $\mathcal{P}\left(x_{\mathcal{S}}^{(k)}\right)$. By minimizing the Kullback–Leibler (KL)-divergence between this and the original prediction, $\mathcal{P}(x)$, we can force $\mathcal{S}$ to learn to reconstruct the original prediction for the drawn sample. To limit $\mathcal{S}$ to learn masks which include a small set of features capable of preserving the original prediction, a conservation penalty term is necessary. This penalty term is given by the KL-divergence between the $\widehat{x_{\mathcal{S}}}$ and the background distribution $\tilde{b}$, which is fit towards a target conservation value t_{bits} and weighted with a parameter λ. The full inclusion objective is given in Equation 4.2. During training, gradients are backpropagated to $\mathcal{S}$ using either Softmax Straight-Through estimation (218) or the Gumbel distribution (219).

$$\min_{\mathcal{S}} \left(\frac{1}{k}\Sigma_{k=1}^{K}\mathrm{KL}[\mathcal{P}(x_{\mathcal{S}}^{k}) \parallel \mathcal{P}(\mathrm{x})]\right) + \lambda \cdot \left(\mathrm{t}_{\text{bits}} - \frac{1}{\mathrm{N}} \cdot \mathrm{KL}[\tilde{\mathrm{b}} \parallel \hat{\mathrm{x}}_{\mathcal{S}}]\right)^2 \quad (4.2)$$

In another formulation of scrambler termed occlusion, the goal is to find the smallest set of features in $\boldsymbol{x}$ to randomize (i.e., maximize the conservation of $\widehat{\boldsymbol{x}_{\mathcal{S}}}$) such that the KL-divergence of the sampled predictions and the input prediction is maximized. Equation 4.3 shows how $\widehat{\boldsymbol{x}_{\mathcal{S}}}$ is calculated for the occlusion objective, and Equation 4.4 gives the full training objective. It is important to note that the sets of features

that the inclusion and occlusion scramblers find are not necessarily the same - as they were formulated to pick up different logical relations between features. For features with an 'OR' type relationship, where the presence of either is sufficient to produce a prediction, occlusion would identify all features, while inclusion may only find some of the OR-relationship features. Alternatively, with AND-relationship features, inclusion would identify all the features, while occlusion could potentially find some of the features. The difference between these two scrambler formulations can be seen in Figure 4.1.

$$\widehat{x_S} = \sigma\left(\log \tilde{b} + x/\dot{S}(x)\right) \quad (4.3)$$

$$\min_S \; -\left(\frac{1}{k}\Sigma_{k=1}^{K}\mathrm{KL}[\mathcal{P}(x_S^k) \parallel \mathcal{P}(\mathrm{x})]\right) + \lambda \cdot \left(\mathrm{t_{bits}} - \frac{1}{\mathrm{N}} \cdot \mathrm{KL}[\tilde{\mathrm{b}} \parallel \hat{\mathrm{x}}_S]\right)^2 \quad (4.4)$$

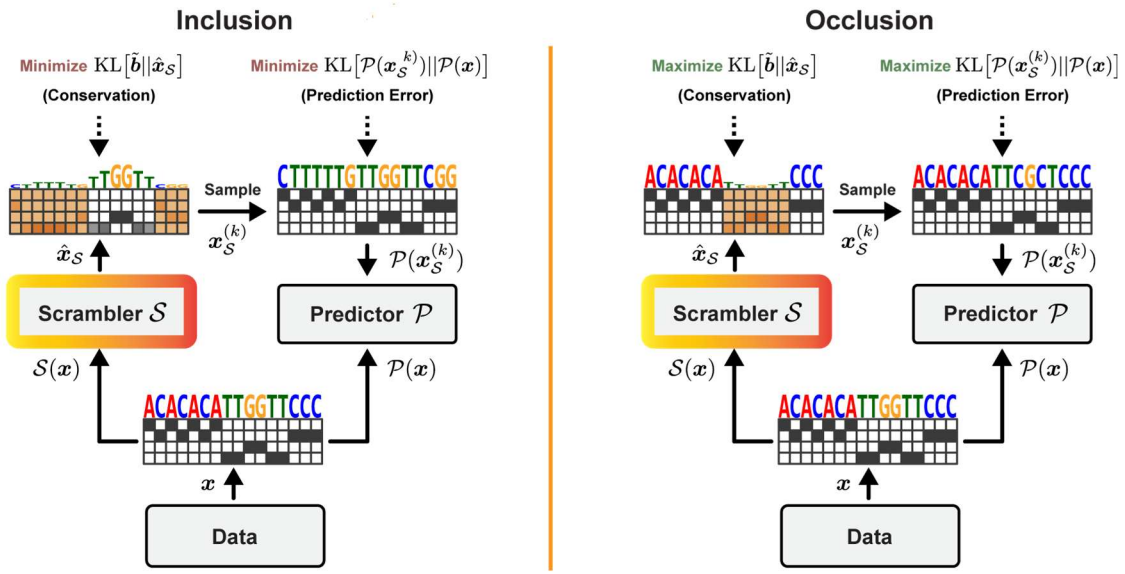


Figure 4.1. Overview of the two scrambler formulations. Masked area is shown in orange on the output of the scramblers. Left is the inclusion scrambler, which learns to minimize the conservation to find the smallest set of features to keep such that the prediction error is minimized. Right is the occlusion scrambler, which learns to maximize the conservation of the original sequence and find the smallest set of original features to mask such that the prediction error is maximized.

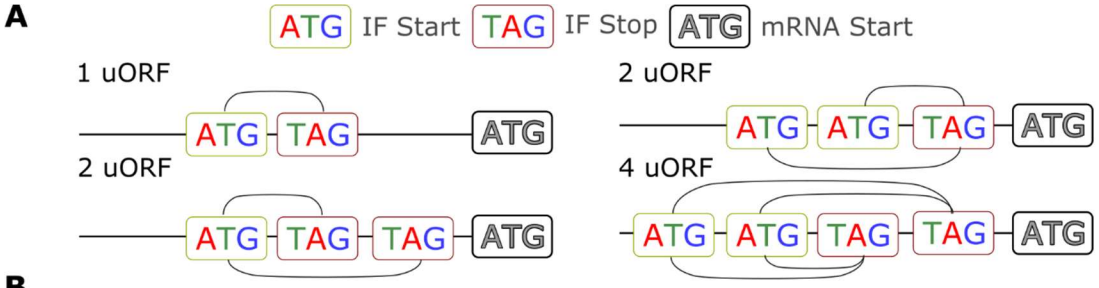
4.3. A benchmark for 5' UTR translation efficiency rules

Complex regulatory logic in the 5′ UTR controls mRNA translation efficiency. For example, an in-frame (IF) start and stop codon create an IF upstream open reading frame (uORF), which typically represses translation, while neither an IF upstream start nor upstream stop codon individually represses translation (NAND logic). Sequences with multiple IF starts and stops can further complicate translational logic by creating NAND-OR hybrid functions with overlapping IF uORFs. As such, 5'UTRs present an excellent benchmarking opportunity for mask-based attribution methods.

The model selected for interpretation was the CNN Optimus 5-Prime (75), which was trained on 5' UTR MPRA data to predict mean ribosomal load (MRL). MRL can act as a proxy for translation efficiency, as transcripts with a high MRL can be translated by the multiple ribosomes, while those with lower MRL have less ribosomes engaging in translation. Inclusion scramblers were trained for this task on 10,000 UTRs. Both a high conservation penalty ($\lambda = 10$) and a low conservation penalty ($\lambda = 1$) were trained to investigate the effects of encouraging scramblers to select a smaller feature set.

To investigate the ability of different attribution methods to recover uORF logic, synthetic sequences were generated to have increasing numbers of uORFs. Sequences with one uORF (1 upstream start, 1 upstream stop), two uORFS (1 upstream start, 2 upstream stops and 2 upstream starts, 1 upstream stop), and four uORFS were generated (2 upstream starts, 2 upstream stops). To generate these sequences, start sequences from the original Optimus 5-Prime dataset were selected (egfp_unmod_1, (75)). Only sequences which contained no upstream starts or stops, with MRL that fell between the 5^{th} and 10^{th} percentile were selected. This set of sequences, n = 537, then had the necessary number of IF positions sampled uniformly from all possible IF positions, with the rest of the sequence remaining fixed. For each of the four datasets, n = 512 sequences were created by inserting ATG for the necessary

upstream stars, and TAG for the necessary upstream stops (see Figure 4.2A to see the structure of generated IF uORF sequences).



B

Mean Squared Error	1 Start 1 Stop 12nt	1 Start 1 Stop 9nt	1 Start 1 Stop 6nt	1 Start 2 Stop 12nt	1 Start 2 Stop 9nt	1 Start 2 Stop 6nt	2 Start 1 Stop 12nt	2 Start 1 Stop 9nt	2 Start 1 Stop 6nt	2 Start 2 Stop 12nt	2 Start 2 Stop 9nt	2 Start 2 Stop 6nt
SCRAMBLER (Lo. P.)	0.30	0.35	0.55	0.36	0.51	0.99	0.29	0.46	1.52	0.43	1.00	2.14
SCRAMBLER (Hi. P.)	0.34	0.37	0.58	0.39	0.50	0.75	0.48	0.53	0.99	0.54	0.70	1.41
PERTURBATION	0.33	0.39	0.54	0.66	0.84	1.23	0.36	0.55	1.12	0.87	1.39	2.18
GRADIENT	0.47	0.57	0.91	0.53	0.67	1.09	0.62	0.99	1.75	0.72	1.20	2.15
GUIDED BACKPROP	1.37	1.41	1.49	1.59	1.65	1.80	2.60	2.68	2.71	2.85	2.93	3.11
DEEPLIFT	0.39	0.44	0.69	0.49	0.62	1.04	0.28	0.42	1.26	0.41	0.85	2.13
DEEPSHAP	0.58	0.64	0.88	0.77	0.87	1.18	0.37	0.56	1.30	0.62	1.13	2.06
INT. GRAD.	0.39	0.45	0.71	0.46	0.58	0.98	0.35	0.52	1.26	0.39	0.78	2.00
TORCHRAY (BLUR)	0.62	0.73	0.77	0.70	0.88	0.94	0.77	1.09	1.32	1.10	1.44	1.89
TORCHRAY (FADE)	0.72	0.84	0.90	0.89	1.07	1.13	1.01	1.21	1.40	1.29	1.58	1.86
SALIENCY MODEL	0.60	0.83	1.26	0.66	1.02	1.51	0.86	1.52	2.35	0.92	1.69	2.62
SIS (MEAN)	0.41	0.50	1.23	0.42	0.64	1.48	0.36	1.06	2.47	0.55	1.60	2.98
ZERO SCRAMBLER	1.69	1.76	1.81	1.96	2.01	2.08	2.82	2.84	2.86	3.23	3.27	3.29
L2X	1.54	1.58	1.65	1.75	1.81	1.90	2.48	2.51	2.60	2.83	2.87	2.96
INVASE	1.35	1.41	1.51	1.53	1.70	1.84	2.09	2.26	2.45	1.83	2.21	2.65

C

Mean Squared Error	1 Start 1 Stop 6nt	1 Start 2 Stop 9nt	2 Start 1 Stop 9nt	2 Start 2 Stop 12nt
SCRAMBLER (Lo. P.)	0.55	0.51	0.46	0.43
L2X	1.65	1.81	2.51	2.83
INVASE	1.51	1.70	2.26	1.83
L2X (FULL DATA)	1.65	1.78	2.40	2.66
INVASE (FULL DATA)	1.81	2.05	2.80	3.16

Figure 4.2. Synthetic overlapping uORF sequence design and evaluation. A. IF positions for low MRL sequences from the original dataset were selected with no existing upstream starts or stops. To create each set, 2, 3, 3, or 4 IF positions were sampled without replacement from all possible IF positions and then filled with ATG or TAG in the order shown in the figure to create the necessary overlapping reading frames. B. The MSE values of model predictions when keeping only the top N nucleotides fixed when making predictions C. MSE values of model predictions for L2X and INVASE trained on the full dataset to attempt to improve performance

Both scramblers and all other benchmarked methods were used to score the synthetic datasets. These benchmarked methods consisted of an *in silico* mutagenesis method, Perturb, which exchanged the categorical value of one letter at a time and estimates the absolute value in model predictions as the importance score. Also tested were a variety of gradient-based attribution methods, Gradient Saliency (203), Guided Backprop (220), Integrated Gradients (221), DeepLIFT (200) (using Rescale from DeepExplain (222)), SHAP DeepExplainer(198), the preservation/perturbation methods of Fong et al. (TorchRay) (223), Dabkowski et al. (Saliency model)(215) and Carter et al. (sufficient input subsets or SIS)(201) and the feature selection methods L2X(212) and INVASE(216). To determine if the inserted IF starts and stops were identified by the methods, we ranked the nucleotides by their importance scores. Then, for each method, we kept only the top N scores (where N = 6, 9, or 12) fixed while all other positions were replaced with random samples from a background distribution. The MSE of the predictions for these sequences versus the original synthetic set was then calculated, and can be seen in Figure 4.2B. For the simplest case with one uORF, nearly all methods performed well. As more IF elements were added, scramblers continued to perform well compared to most other methods. In particular, the high penalty scrambler outperformed all methods in the most difficult MSE task, where only N = 6 positions were retained for the multi uORF datasets. Additionally, as the L2X and INVASE methods performance depends on training an interpreter model, a benchmark where L2X and INVASE had access to the full 260K UTR dataset for training the interpreter was conducted. However, this did not improve L2X performance drastically and resulted in higher MSE values for INVASE (Figure 4.2C).

To directly evaluate the models on their ability to recover the inserted starts and stops, we calculated the ability of the methods for ranking the inserted nucleotides as among the most important features. The strictest version of this was referred to as ‘maximum solution’, where all stop and start codons had to be amongst the highest scored positions to count as successful identification for any given method. The inclusion scrambler performed well on this task across all benchmark sets – only

outperformed by DeepLIFT on the 2 start, 1 stop dataset. The fraction of successfully ranked sequences can be seen in FIGX. Another version of the recall task considered the minimum solution, where we counted a method successfully identifying at least three start and three stop nucleotides as a success. In that case, the fraction of correctly ranked sequences was higher for other methods, excluding L2X and INVASE which still largely failed the ranking tests (Figure 4.3A).

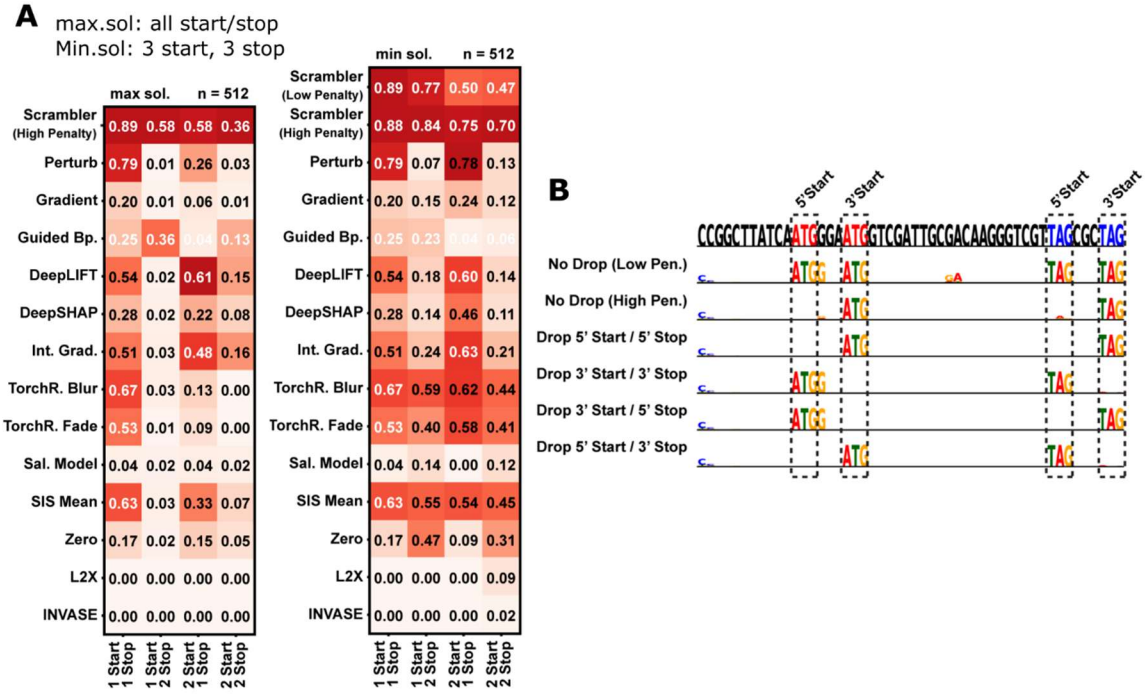


Figure 4.3. Nucleotide ranking tests and an example of targeted attribution explorations. A. The results of the maximum solution (max sol) and minimum solution (min sol) feature ranking tests. Numbers shown are the ratio of sequences out of the total set (n = 512) correctly ranked by the attribution methods. B. An example sequence attribution of the low and high conservation penalty scramblers for one foteh 2 Start, 2 Stop designed sequences. The use of dropout scramblers allows the recovery of all four of the reading frames in the sequence.

One of the advantages of Scramblers over the other compared methods, was that layers could be introduced to force certain features to be ignored or included in attribution. One such layer is the dropout layer, which introduces random dropout patterns to the training of the Scrambler. Post-training, this layer can be used to

force the Scrambler to try and recover different feature sets. Using the dropout layer, different uORFs can be individually extracted from the synthetic sequences. By adding an importance score dropout layer to the low penalty inclusion scrambler and training it with random dropout patterns, we could dynamically explore IF uORF sets. For example, for the synthetic four uORF sequence in Figure 4.3B, the non-dropout low penalty scrambler finds all IF starts and stops, and the high penalty scrambler only finds one uORF. By using dropout patterns to exclude either of the IF starts and stops, the dropout scrambler can be used to dynamically find the alternative uORFs in the synthetic sequence.

4.4. Recovering the binding determinants of designed heterodimers

Interpreting the important features for protein-protein interactions can be difficult, given that protein sequence distributions have narrow manifolds for stably folded sequences. Therefore, any mask-based feature attribution method must ensure that masked sequences stay in distribution for predictions to remain accurate. For this attribution task, we used a set of rationally designed coiled-coil heterodimers as the training set, which were created to have networks for hydrogen bonds (HBNets) at the interface of the heterodimer to control binding specificity (149,224) (see Figure 4.4A for an example dimer). A RNN was trained to classify two protomers as being designed to interact or not, using a set of 180k designed pairs as the positive training data and randomly paired protomers as the negative set. This model reached a high level of accuracy, AUC = 0.96, on n=26,459 test datapoints (Figure 4.4B).

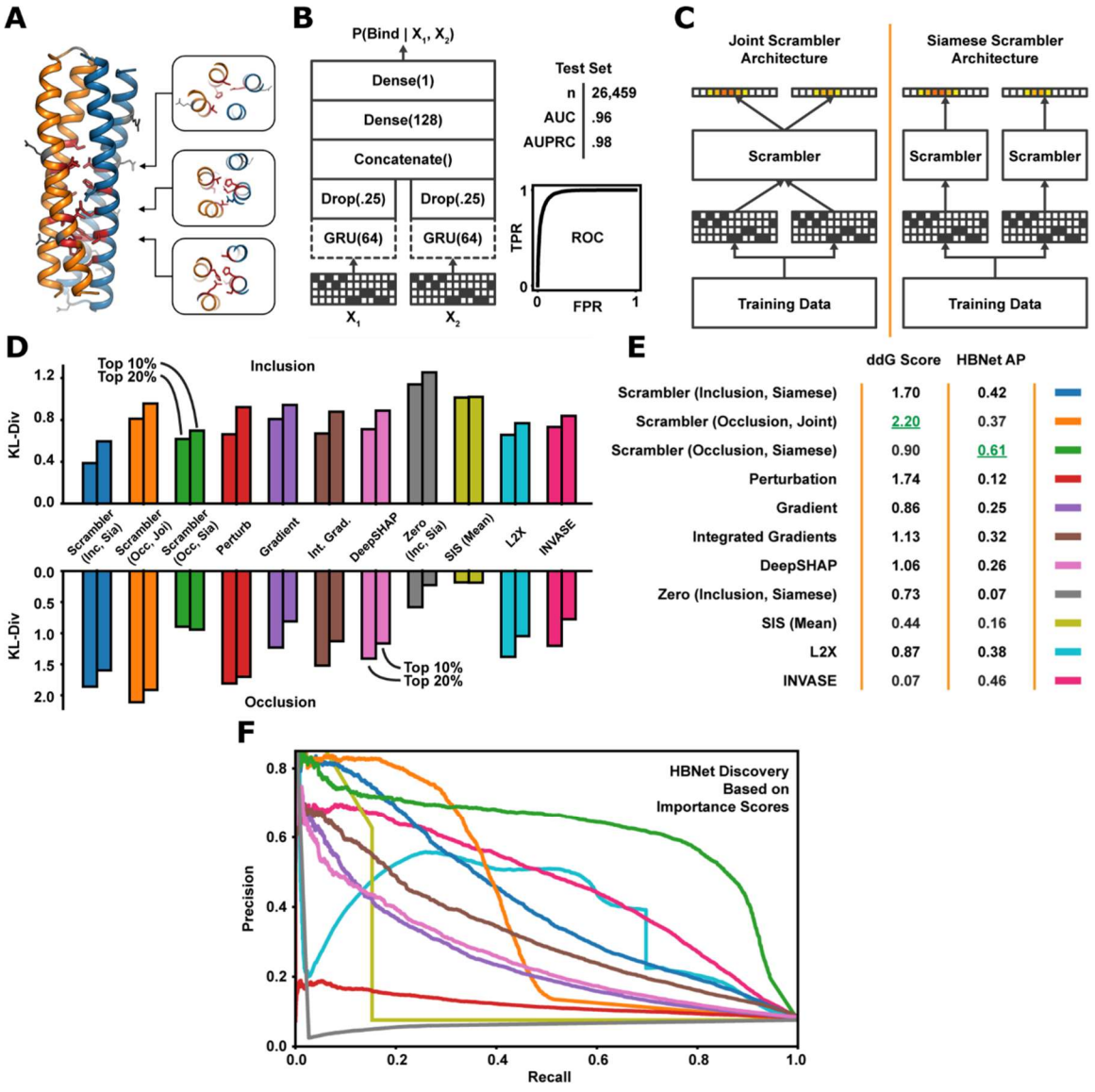


Figure 4.4. Scramblers for understanding designed coiled-coil interactions. A. The structure of the designed coiled-coil heterodimers, showing the hydrogen bond networks buried at the interface. B. The structure and performance of the dimer classifier RNN, and its performance on a test set. C. The two options for scrambler architecture, given the paired nature of the inputs. D. The KL-divergence of sequences where the top X% most important positions identified by a given feature attribution were either kept fixed with the rest of the sequence being replaced from a background (inclusion) or had the X% positions replaced from a background while the rest of the protein sequences were kept fixed (occlusion). E. The performance of the different methods in the $\Delta\Delta G$ and HB recovery challenges. F. The precision recall curve for E, used to calculate the average precision values.

Another interesting challenge for this feature attribution task is that the binder sequences have varying lengths. This is because the heterodimers generally follow a conserved heptad repeat structure to ensure coiledness, but this heptad pattern can have varying offsets. While this does not vastly change the behavior of attribution methods with per-input methods, it does introduce additional complications for those training attribution models, such as scramblers. When training scramblers for the classifier RNN, we used a separate background distribution for each sequence length in the dataset. An additional structural consideration for the scramblers for this RNN was that each prediction involved two input sequences, which meant there were two scrambler architectures to consider: one which would take in two input sequences at once (a joint architecture) and one which would take in one sequence at a time (a Siamese architecture) (Figure 4.4C).

The best performing methods can be identified in several ways. The first is how important the features selected are for preserving/destroying the original predictions. Using a test set of 478 designed heterodimers, the KL-divergence of the original predictions and sequences randomized to test attribution method performance was calculated. These randomized sequences were generated by replacing all but the top X% most important amino acid residues in the test set sequences with random samples (inclusion) or, conversely, replacing the top X% amino acids with random samples and keeping the remaining sequence fixed (occlusion). Multiple scrambler versions were trained, and it was found that a Siamese inclusion scrambler ($t_{\text{bits}} = 0.25$) and joint occlusion scrambler ($t_{\text{bits}} = 2.4$) had the best (lowest and highest, respectively) median KL divergence for these tasks (Figure 4.4D).

However, similar to the 5' UTR attribution task described above, this dataset was selected for study because pre-defined, performance-relevant features were known for the task: the HBNets. The HBNets were a key part of the design process outlined in (149), where dimers were designed to have a minimum of four residues involved in HBNets such that the net contacted all four helices and all heavy-atom donors and acceptors in the network were satisfied. However, in later steps in the design process,

the HBNets could become disrupted, which made their identification from the final designs difficult. Therefore, more relaxed criteria than the original design specifications were used to recover as many potential HBNet residues as possible from the test set structures (HBNetStapleInterface protocol with min_network_size = 3, min_helices_contacted_by_network = 3, hb_threshold = –0.3, and find_only_native_networks = true and the ref2015 score function). Overall, the joint occlusion scrambler excelled at identifying HBNet positions compared to all other methods tested. One reason may be that the Siamese occlusion architecture, which saw only one input sequence at a time, was constrained to learn global, partner-independent features corresponding to potential HBNet positions. Examples of these attributions can be seen in Figure 4.5 and the positions identified as important by other methods. Meanwhile, the joint inclusion scrambler appeared to identify a subset of HBNet positions and hydrophobic residues at the interface necessary for binding specifically to the cognate partner.

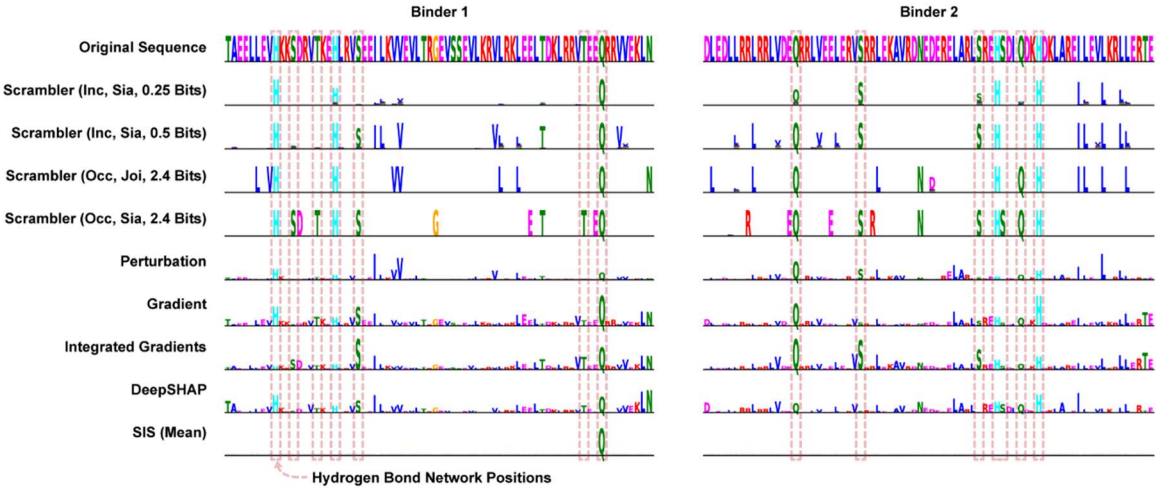


Figure 4.5. An example of the attributions for one of the designed pairs. The boxed positions are recovered HBNet positions.

To investigate how much important positions contributed to binding specifically to the cognate partner, another metric was calculated for each method. As all test set heterodimers included the predicted structure, it was possible to conduct energetic simulations to quantify how much each position contributed to the overall

interaction using *in silico* alanine scanning. Computational alanine scanning was carried out for all positions in a heterodimer pair using PyRosetta (4). Each position was mutated to alanine and repacked with the neighboring residues, which were prevented from designing and repacking. The InterfaceAnalyzerMover (set_pack_input=False, set_pack_separated=True) was used to calculate the mean mutation $\Delta\Delta G$ in REU at each position. Then, for each attribution method, the eight most important residues per dimer were selected, and the difference in the mean $\Delta\Delta G$ between the top residues and the mean $\Delta\Delta G$ between all other residues were computed. Higher $\Delta\Delta Gs$ between the top residues and all other residues meant these positions were more important to the stability of the complex. In addition, permutation tests for n = 10000 eight-residue samples were conducted for the test set to verify that the top $\Delta\Delta G$ - other $\Delta\Delta G$ was significant for the different methods.

The joint occlusion scrambler identified residues that destabilized the complex the most (mean difference $\Delta\Delta G = 2.20$; Figure 4.4E), whereas the Siamese model discovered substantially more HBNet positions (AUPRC = 0.61, Figure 4.4F). These results support that the Siamese architecture learned discriminative features for interacting and non-interacting pairs (HBNet residues). In contrast, the join architecture led to identifying positions that were key for interacting pairs. Also of note is that some methods, like Integrated Gradients and DeepSHAP, return signs for their scores, with positive values indicating that the features contribute more to the prediction. Therefore, the same benchmarks were run, but only the positive-valued scores for these methods were considered. These positive-only scores led to the methods having larger perturbations of model predictions than previously, but the scramblers still had better $\Delta\Delta G$ scores and HBNet discovery rates (Figure 4.4E). This suggests that using a parametric masking model results in more generalizable features, potentially by ignoring spurious RNN signals.

Chapter 5: Ongoing projects

In this chapter, I outline recent work and work in progress. In particular, I talk about a project I am involved in working with a large scRNA-seq dataset, where I work to condense human immune system literature databases. In addition, I outline septin modeling update ideas.

5.1. The human immune cell dictionary – a massive scRNA experiment

PPIs exist in the context of a system – these interactions drive programs of cellular behavior. While being able to predict protein interactions or what perturbations on those interactions may do, it is only a fraction of the whole of their role in the cell. To understand the downstream effects of interactions, other types of experiments are often necessary. An example of such interactions are cytokine-receptor interactions in the immune system.

5.1.1. Background

Cytokines are signaling proteins secreted by cells, often by cells of the immune system, which have effects on other cells . They are a critical component of how immune cells communicate with each other, with immune cells often secreting cytokines that interact with receptor complexes on the surface of other immune cells, activating signaling cascades within the cell. These cascades can then result in changes in cell behavior – such as by upregulating or downregulating the expression of certain genes or resulting in cell movement.

Recently, Parse Biosciences used a new single-scaled up, automated workflow called GigaLab to collect scRNA-seq data for human immune cells exposed to 90 different types of cytokines (225). Single cell RNA-seq (scRNA-seq) is a technique in which the expression profiles of single cells from a sample can be collected via high-

throughput sequencing (226). Peripheral blood mononuclear cell (PBMC) samples from 12 donors were used, resulting in a huge dataset of over 10 million cells. The sheer scale and number of treatments conducted in the experiment resulted in a massive dataset which is intrinsically difficult to analyze and extract new relationships from due to its scale. This project is also being worked on currently by Lukas Oesinghaus in the Seelig Lab, members of the Theiss Lab, and employees of Parse Biosciences.

My contribution to the project was to comb available human immune databases, determine which were helpful when evaluating immune cell treatment responses, and combine them into a reference format which would for easy comparison with the reference values. To accomplish this, it was necessary to determine which immune cell types could potentially be present in PBMC samples. It then had to be determined which of these cell types were present in the Parse data. Finally, these cell types needed to be connected to known immune-cell specific and cytokine-specific information and compared qualitatively and quantitatively with the Parse values.

5.1.2. Relevant databases for constructing an immune-cytokine reference

Human immune system cells are defined mainly by their descent paths, programs of active genes, and their location of residence. In adults, most immune cells originate from stem cells in the bone marrow and must travel out of it to mature fully. In some cases, this involves travelling to highly specialized tissues and cell populations known as secondary lymphoid organs to mature into functional immune cells. Immune cell migration to and from different tissues typically occurs through two main paths: the lymphatic system, and the circulatory system. Determining which immune cells can be present in PBMC samples, and potentially detected by the scRNA experiments, depends on identifying which immune cell populations present in the blood.

Additionally, the technique through which the PBMC samples were collected contributes heavily to what cell types are present in the samples. Different methods

exist for PBMC prep, but their end goal is to separate whole blood into layers containing different cell types, typically with some kind of gradient centrifugation (227). This results in the components of blood being separated into several layers: plasma, multi-nucleated/multi-lobed nuclei cells, and mononuclear cells. PBMCs consist of the extracted mononuclear cell layer. Small amounts of non-mononuclear cells can end up in the PBMC sample as contaminants, typically consisting of platelets, red blood cells, or granulocytes (228).

With these facts in mind, a starting point to identify which cell types are capable of traveling in the blood was identified. The Cell Ontology (CL) (229) is a ontology resource with the goal of assigning every cell canonical, natural cell type a unique identifier, a brief description, and to store relationships between cells and other ontologies. There are several subsets of the Cell Ontology based on expert selected subsets – one of these is referred to as *blood_and_immune* which was selected to be the starting set of cells for annotation. Also of importance were other databases identified as key for creating a literature reference to compare the Parse data against.

Other important databases are cell marker databases – specifically those which connect CL IDs to marker genes. Marker genes are genes which are expressed in certain cell types, which allow cell types to be differentiated from one another in sc-RNA studies by comparing their marker gene profiles. These genes are typically identified by clustering cells, and determining which genes are differentially expressed in which clusters (230). Typically, these are then manually verified by human experts or matched with common marker gene sets using specialized tools (231). Several works have compiled databases of marker genes connected specifically to CL IDs: including the human reference atlas (HRA) (232) and CellMarker2.0 (233). CellMarker2.0 is a manually curated collection of marker genes from over 24k published datasets for human and mouse cells, while the HRA is a multi-national consortium, working to map the locations of all the cell types in the human body at multiple organizational levels (i.e., tissues, organs, etc.). The HRA resources include

a set of tables which provide standardized names for major anatomical structures, cell types, and biomarkers/marker genes known as ASCT+B tables.

The other set of relevant databases are those storing connections between cells and cytokines. Despite this being an important subject area, resources for this topic are significantly lacking. Of relevance here are two systematic resources attempting to create overviews of expected general responses and effects in immune cells when cytokines bind to their receptors, with very different approaches to meeting their goal. In one, ImmunoGlobe, immunologists combed a standard immunology textbook and created manual annotations for relationships between immune cell types and cytokines for humans and mice (234). The other systematic main database is ImmunExpresso, which was created through sentiment analysis on PubMed abstracts prior to July, 2017 (235). This effort used vocabularies of cell types, cytokines, and verbs to identify subjects and actor relationships in the abstracts and determine the number of papers supporting said relationships. For example, a cytokine actor with a cell subject and a 'positive' verb could mean a sentence where a cytokine promoted proliferation of a cell. For the purpose of clarity, relationships in these databases where a cytokine interacting with a receptor and leading to a change in cell activity will be denoted as $cytokine \rightarrow cell$, while relationships like excretion where a cell produces a cytokine will be denoted as $cell \rightarrow cytokine$.

5.1.3. Constructing a reference human immune cell network

To construct a unified immune cell reference using the data resources described above, the *immune/blood* subset of the Cell Ontology was used as a basis. The union of all CL IDS in the subset, along with all CL IDs in the ACST+B tables, CellMarker2.0, ImmunExpresso, and ImmunoGlobe were taken. The overlaps of these sets revealed that the blood/immune CL subset was missing relevant cell types, so under the assumption that other immune cell types were present in the CL graph but not captured in the union of CL IDs in identified resources, the nearest neighbors of the superset of CL IDS were added to the final set of potentially relevant CLI DS. This resulted in a directed CL graph with 578 nodes, 750 *is_a* edges (relationships

where a cell is a subtype of another cell, i.e., a CD4 T cell is a T cell), and 213 *develops_from* edges (relationships where a cell develops from another cell type).

Iterative steps were then taken to remove or reduce non-PBMC cells and merge cell nodes together for ease of analysis. For example, all non-human cells were removed, and non-PBMC cells were removed such as endothelial and epithelial cells. The next graph reduction step focused on identifying stages where tissue migration occurred during development through literature review. Information from other large-scale PBMC based scRNA-seq studies (236,237) was used to identify cell types which may have been present in their PBMC samples. As groupings were established, it became apparent that many *is_a* and *develops_from* relationships in the network were missing or needed modification. These edges were inserted as needed. The most relevant stages in adult immune cell development are outlined below per cell type:

- B cell: Common lymphoid progenitor (CLP) differentiate into transitional-B cells in the bone marrow, which then leave via the blood and head to secondary lymphoid organs for development to naïve B cells. Both naïve B cells and memory B cells can circulate in blood. However, it is important to note that when B cells become active and turn into antibody secreting plasmablasts and plasma cells, plasma cells are primarily bone-marrow or secondary lymphoid tissue resident. (238–241).
- T cell: CLP cells exit the bone marrow and arrive in the thymus via the bloodstream and become pro-T cells. These cells become thymocytes, which are immature T cells which must go through stages of selection to become mature T cells. The various T cell subsets exit the thymus. These can be naïve T cells, or those which do not require stimulation to become active such as natural killer T (NKT) cells. Some T cell sets reside mostly in tissues, and are not present in PBMCs, except perhaps when they travel to their tissues of residence from the thymus. (242–244)
- Monocyte/macrophages: Classical monocytes (CD14 high) monocytes are thought to leave the bone marrow and can transition into intermediate and non-classical (CD16 high) monocytes. Monocytes can become macrophages or dendritic cells after migration into tissues (245,246).

- ILC/NK cells: Innate lymphoid cells (ILCs) are thought to leave the bone marrow immature as innate lymphoid precursors and mature in tissues. Natural Killer (NK) cells are of the same general lineage, and are though to leave as partially mature NK cells/natural killer progenitor cells and mature outside of the bone marrow (247–249)
- Dendritic cells: Dendritic cells (DCs) have a more complicated developmental pathway – for example, monocytes can turn into dendritic cells. It is thought that conventional DCs (cDCs) leave the bone marrow in a more immature form, while plasmacytoid DCs (pDCs) leave in a more mature form (250,251).

The final cell differentiation network possessed 105 cell type and grouped cell type nodes, 85 is_a relationships, and 46 *develops_from* relationships. The individual CL IDs associated with the nodes were attached to their marker gene information, and the marker genes per a cell type could be gathered for different levels of analysis by selecting a node and all its descendants.

5.1.4. Incorporating marker gene information

Post-network completion, 118 cell types had marker gene information, and 839 of the marker genes appeared in the highly variable genes in the Parse data. Dotplots showing the mean expression of a gene and the fraction of cells in a group were generated to double-check and support the assignments given by the Theiss lab. For example, they were used to verify that no macrophages were present in the data, verify that small populations in the datasets matched granulocyte contamination, and identify additional markers which showed strong differences for the cell populations selected (see Figure 5.1 for additional marker genes suggested for the Parse clusters to support their identification).

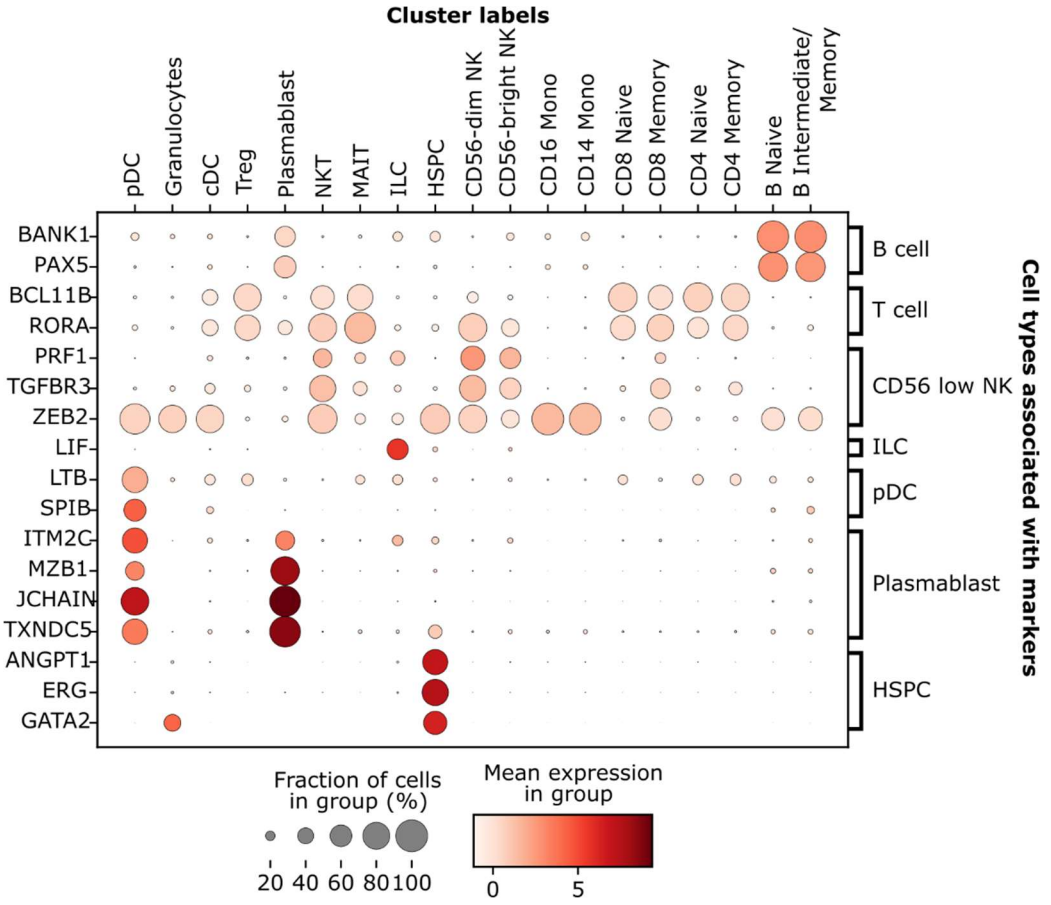


Figure 5.1. Dotplot of marker genes to aid with cluster identification. Found via database merging with CL network, and manual examination of strongly expressed markers . Example made with four participants PBS control data.

5.1.5. Comparing immune cell-type specific cytokine responses to literature values

To compare the cytokine effect database values to the Parse responses, further compression of relevant cell types was needed, and the cytokines information had to be standardized. The cell types used for analysis elsewhere in the currently ongoing Parse data paper included B cells, CD4+ T cells, CD8+ T cells, regulatory T cells (Treg), NKT cells, NK Low and NK Hi cells, ILC cells, pDCs, cDCs, CD14 Monocytes, CD16 monocytes, and HSPC cells. These clusters were used for

psuedobulking counts, and for calculating the differential expression between control conditions and the cytokine treatment conditions. Nodes in the network were selected to be grouped together for effect and secretion analysis (Figure 5.2). Since the databases did not typically discriminate between NK subtypes or monocyte subtypes in annotations, these were grouped together. For comparing sc-RNA data to the databases, the largest population group was used (NK low for NK cells, and CD14 monocytes for the monocytes, respectively). Post standardization of cytokines, the cytokine-effect databases could be merged, yielding the number of unique $cytokine \rightarrow cell$ and $cell \rightarrow cytokine$ relationships shown in FIGX. An immediate issue with the available data was that there is a lack of agreement in the databases for effect types in the non-secretion data, and multiple recorded labels per $cytokine \rightarrow cell$ effect. ImmunExpresso had a high degree of $cytokine \rightarrow cell$ relationships with both negative, positive, and neutral labels. However, the number of negative and positive supporting papers was correlated ($r^2 = 0.64$, $n = 381$) indicating that this was at least partially due to the research popularity of a $cytokine \rightarrow cell$ relation. To simplify analysis, the supporting paper count was taken to be the sum of all papers for the $cytokine \rightarrow cell$ relation. Similarly, there were overlaps in the labels assigned by the ImmunoGlobe curators per relationship. Relationships with multiple labels were grouped together depending on if they did or did not contain a 'Inhibit' label. (Figure 5.2).

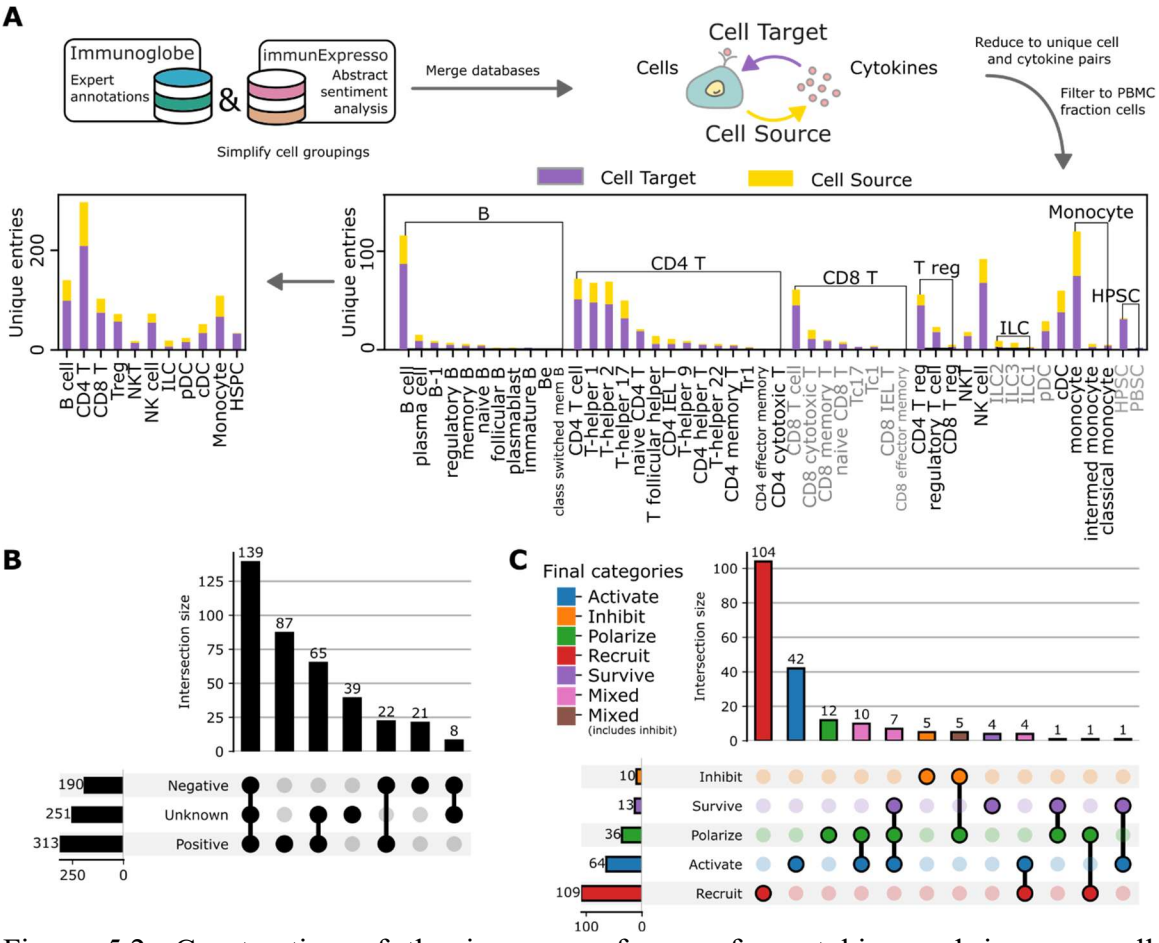


Figure 5.2. Construction of the immune reference for cytokine and immune cell relationships. A. The two databases Immunoglobe (234) and ImmunExpresso (235) were merged into a single resource of *cytokine* → *cell* (purple) and *cell* → *cytokine* (yellow) relationships. These were the standardized to only include relationships for cells corresponding to the clusters in the Parse dataset. B. Upset plot of the ImmunExpresso categories. Note that most relationships have all annotation types present. C. Upset plot of the expert-assigned annotations in ImmunoGlobe. The final categories are the groupings which are used in heatmap visualization.

To compare the Parse cytokine dictionary responses to this data, some kind of overall metric would be needed. In a recent scRNA-seq paper investigating the effects of cytokine treatment at scale, mice were treated with different cytokines, and their draining lymph nodes collected for scRNA-seq measurement (237). Since this is a very similar study to the Parse experiment, and a similar scale, much of the project analysis for the Parse human dictionary mirrors this paper. However, the Mouse

dictionary did not conduct any large-scale literature comparisons to direct exploration of trends to differentiate new and unknown treatment effects. A global magnitude of cytokine response was calculated per cell type treatment using the Euclidean distance between the centroid vectors of cytokine-treated and PBS-treated clusters per cell type. These values were then winsorized so that all values over the 95th percentile were replaced with the 95th , and min-max scaled between 0 and 100. A similar metric was developed by Lukas Oesinghaus, winsorizing and min-max scaling cell cluster distances from the PBS control cluster to provide an overall effect score of the cytokine on a given cell type. He also incorporated a second metric given by $R_{\text{strength}} = \sum_{i=1}^{N} |\log_2 \text{FC}_i| \cdot -\log_{10}(p_{\text{adj},i})$, which scaled the LFC of each gene by its significance. The final metric used was the average of the winsorized and min-max normalized $R_{strength}$ and Euclidean distance metrics, where winsorization and min-max scaling was done per cell type across the tested cytokines. Operating under the assumption that the strongest general responses seen in the human and mouse scRNA-seq experiments would be more likely to have been noted at some point in the literature, the relationship between paper numbers and response magnitudes was investigated. An immediate issue in doing so was extreme outliers of the paper values – therefore, the absolute value of supporting paper counts was similarly winsorized and min-max scaled per cell type. Heatmaps of the shared cytokine and cell type relationships for the human and mouse experiments can be seen in Figure 5.3, along with the immunoGlobe and ImmunExpresso combined values for said relationships.

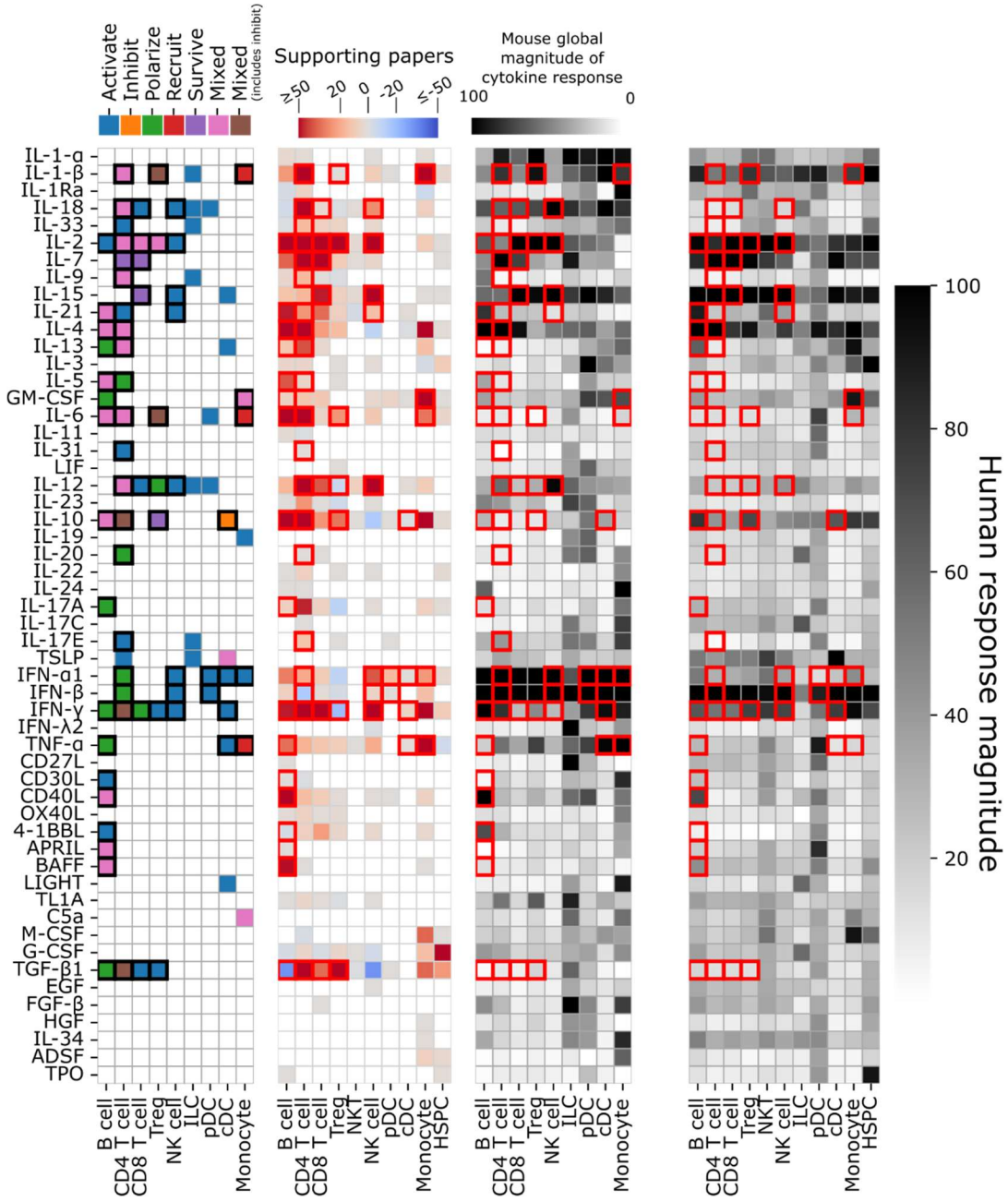


Figure 5.3. Visualization of *cytokine* → *cell* relationships. From left to right, the ImmunoGlobe (234), ImmunExpresso (235), mouse response magnitude (237), and human response magnitude for cell types and cytokines shared between the human and mouse dictionaries. Boxed cells are those present in both databases.

When considering the shared database annotations, there is a decent correlation for both human and mouse magnitudes with the paper values (ρ=0.42 for Parse data, ρ

= 0.33 for mouse data, Figure 5.3A. Note that pDC, cDC, and HSPC cells were removed from correlation due to low number of literature annotations, and HSPC cells not being annotated in the mouse dataset). As can be seen, there are multiple *cytokine* $\rightarrow$ *cell* relationships with high magnitude for mice and human data (annotated on the plot). Some of these, such as the strong effects of IL-4 on B cells, and IL-2 on NK and Treg cells, correspond to high numbers of supporting papers per cell type, verifying that these scRNA-seq studies match literature trends for well-studied cytokines. Perhaps more interesting are the effects where there was a strong response in both datasets but low supporting paper counts - IFN-β on NK cells and CD4 T cells, and IL-1-β on Tregs- as these correspond to less studied interactions and potentially new knowledge about immune responses uncovered by the scale of the Parse experiment and mouse experiments to guide further gene expression analysis.

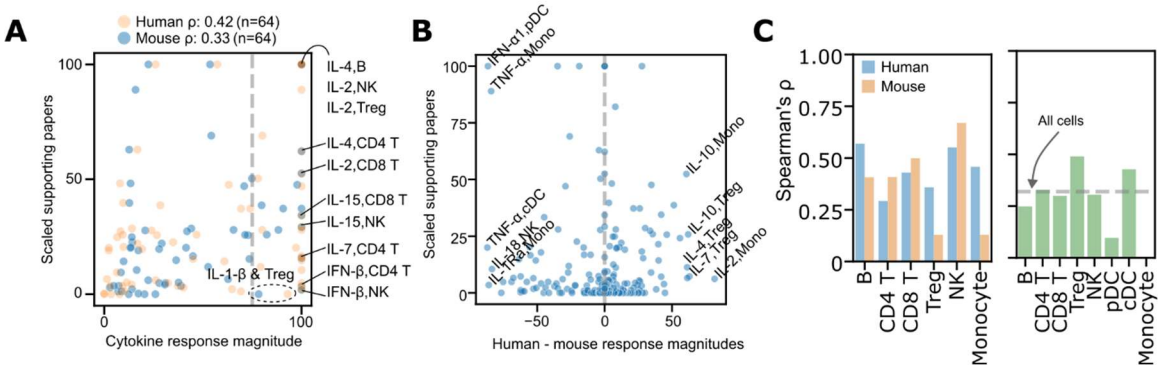


Figure 5.4. Correlation of cytokine response magnitudes with supporting paper counts. A. Min-max scaled supporting paper counts versus magnitudes for relationships in both databases (note that pDC, cDC, and HSPCs were removed for low numbers of annotations). B. Diffference between human and mouse magnitude values, versus the scaled supporting papers for all cell types and all ImmunExpresso annotations. C. Spearman's ρ for human and mouse magnitudes versus the scaled paper count values (left), Spearman's ρ for human and mouse response magnitudes for each cell type. Dashed line is overall correlation for all human and mouse values.

In addition, pairs with large differences in the mouse and human magnitudes are interesting treatments for closer study (Figure 5.3B). These differences may come

in part from the different timescales, concentrations, and the origins of the samples used in the scRNA experiments. However, it is worth nothing that there is a decent correlation between the magnitudes for the two experiments either way (ρ=0.32, n =729). Finally, we can compare the overall trends per cell-type. Some cell types behave closer to what is expected by the literature values for the human magnitudes, and some for the mouse. Additionally, we can see that overall correlations for mice and human response magnitudes disagree most in the monocyte and pDC cell types. For monocytes, this may be in part due having different roles in a lymph node context than in circulation (252).

5.2. Outline of septin-12 update plans

In the gap of time since the completion of the initial septin-12 modeling work and the current day, the variant effect prediction field has continued to advance. In particular, methods combining inputs have grown in popularity beyond the simple P1H input used in Chapter 3. As such, works surveying the prediction aross datsets sizes and testing simpler representations are less needed. Instead, the models will be replaced with those focused on structural predictor inputs, and pLM inptus.

Given the lack of agreement about the utility of predicted mutant complexes in predicting PPI changes (139), predicting complexes for a set of representative wild-type like and strongly detrimental mutations for analysis would be an interesting contribution to the literature. The modeling work in Chapter 2 using simple AF metrics and energetic values offers one such avenue to train predictors, as well as the models which use structural inputs for $\Delta\Delta G$ prediction and decoy ranking (133–137).

The majority of the basic model types investigated previously will be removed – and instead, varieties of the LME and LMA inputs with current pLMs will be explored. This choice is motivated by the performance of the BiGRU model trained with the Bepler (1) representations has similar performance on the held-out position

single mutants for randomly selected positions and the contiguous held-out positions at the end of the selection, and in part due to peculiarities in the representation space of pLM embeddings for the DMS set. The Bepler representation had signal spread out in the LMA representation when examining the difference between mutant and wild-type representations, while the MuPIPR LMA (1) has differences to wild-type localized around the mutated window (Figure 5.4A-B). While comparisons of pLM performance exist, not much work has been done evaluating the embedding spaces themselves when perturbed by mutations in the sequence. It would be interesting to use a variety of LMA and LME representations and compare the performance of models trained on said representations with similarities in the perturbation of their embedding space between mutant and wild-type protein sequences.

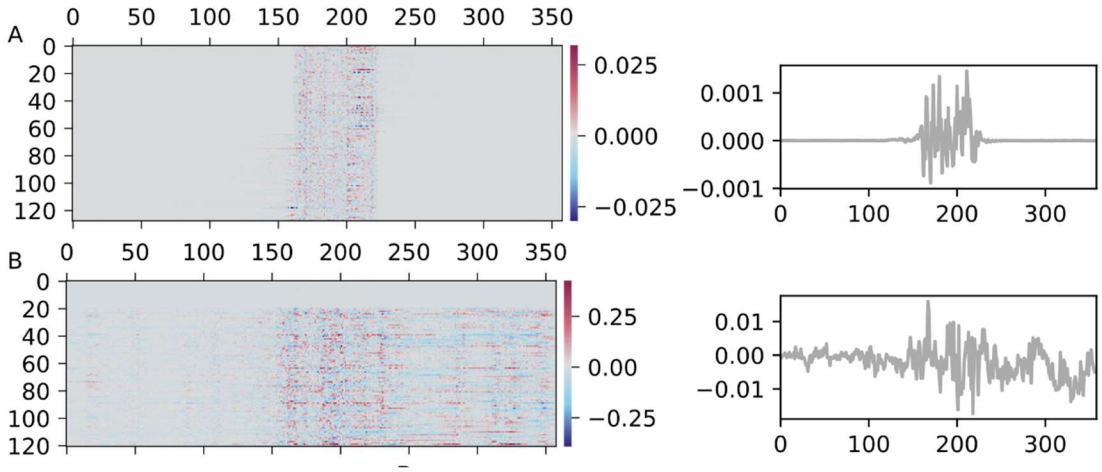


Figure 5.5. Differences in wild type amd mutant septin-12 embeddings across the DMS window. A. MuPIPR (1) mutant embedding differences from WT and their averages across septin-12 single mutants B) Bepler (153) mutant embedding differences from WT and their averages across septin-12 single mutants.

However, there is one general effect prediction technique which would be interesting to replace some of the previous baseline predictors with. Recent work claiming that epistasis is severely overestimated, and that DMS datasets should be modeled without a set wild type sequence. This approach argues that a reference-free

approach should be taken with DMS datasets, and demonstrates good performance at predicting epistatic interactions with this new viewpoint for over 20 DMS datasets (18). Reanalyzing the septin-12 dataset with this as a comparison to the other methods would provide a valuable benchmark, particularly for the held-out position tasks.

References

1. Konstantinidou M, Arkin MR. Molecular glues for protein-protein interactions: Progressing toward a new dream. Cell Chem Biol. 2024 Jun 20;31(6):1064–88.

2. Arkin MR, Tang Y, Wells JA. Small-Molecule Inhibitors of Protein-Protein Interactions: Progressing toward the Reality. Chemistry & Biology. 2014 Sep 18;21(9):1102–14.

3. Moretti R, Bender BJ, Allison B, Meiler J. Rosetta and the Design of Ligand Binding Sites. Methods Mol Biol. 2016;1414:47–62.

4. Chaudhury S, Lyskov S, Gray JJ. PyRosetta: a script-based interface for implementing molecular modeling algorithms using Rosetta. Bioinformatics. 2010 Mar 1;26(5):689–91.

5. Bennett NR, Coventry B, Goreshnik I, Huang B, Allen A, Vafeados D, et al. Improving de novo protein binder design with deep learning. Nat Commun. 2023 May 6;14(1):2625.

6. Vedula S, Bronstein A, Marx A. The end of protein structure prediction: Improving prediction accuracy in chimeric proteins by windowed multiple sequence alignment [Internet]. bioRxiv; 2024 [cited 2025 May 14]. p. 2024.10.06.616858. Available from: https://www.biorxiv.org/content/10.1101/2024.10.06.616858v1

7. Nooren IMA, Thornton JM. Diversity of protein-protein interactions. EMBO J. 2003 Jul 15;22(14):3486–92.

8. Perkins JR, Diboun I, Dessailly BH, Lees JG, Orengo C. Transient protein-protein interactions: structural, functional, and network properties. Structure. 2010 Oct 13;18(10):1233–43.

9. Meyer K, Selbach M. Peptide-based interaction proteomics. Mol Cell Proteomics. 2020 Apr 28;

10. Alborzi SZ, Nacer AA, Najjar H, Ritchie DW, Devignes MD. PPIDomainMiner: Inferring domain-domain interactions from multiple sources of protein-protein interactions. PLOS Computational Biology. 2021 Aug 9;17(8):e1008844.

11. Yellaboina S, Tasneem A, Zaykin DV, Raghavachari B, Jothi R. DOMINE: a comprehensive collection of known and predicted domain-domain interactions. Nucleic Acids Research. 2011 Jan 1;39(suppl_1):D730–5.

12. Mosca R, Céol A, Stein A, Olivella R, Aloy P. 3did: a catalog of domain-based interactions of known three-dimensional structure. Nucleic Acids Research. 2014 Jan 1;42(D1):D374–9.

13. Luo Q, Pagel P, Vilne B, Frishman D. DIMA 3.0: Domain Interaction Map. Nucleic Acids Res. 2011 Jan;39(Database issue):D724–9.

14. Fuxreiter M. Context-dependent, fuzzy protein interactions: Towards sequence-based insights. Curr Opin Struct Biol. 2024 Aug;87:102834.

15. Tompa P, Fuxreiter M. Fuzzy complexes: polymorphism and structural disorder in protein–protein interactions. Trends in Biochemical Sciences. 2008 Jan 1;33(1):2–8.

16. Otwinowski J. Biophysical Inference of Epistasis and the Effects of Mutations on Protein Stability and Function. Mol Biol Evol. 2018 Oct 1;35(10):2345–54.

17. Starr TN, Thornton JW. Epistasis in protein evolution. Protein Sci. 2016 Jul;25(7):1204–18.

18. Park Y, Metzger BPH, Thornton JW. The simplicity of protein sequence-function relationships. Nat Commun. 2024 Sep 11;15(1):7953.

19. Starita LM, Ahituv N, Dunham MJ, Kitzman JO, Roth FP, Seelig G, et al. Variant Interpretation: Functional Assays to the Rescue. Am J Hum Genet. 2017 Sep 7;101(3):315–25.

20. La Fleur A, Shi Y, Seelig G. Decoding biology with massively parallel reporter assays and machine learning. Genes Dev. 2024;38(17–20):843–65.

21. Oliphant AR, Struhl K. An efficient method for generating proteins with altered enzymatic properties: application to beta-lactamase. Proc Natl Acad Sci U S A. 1989 Dec;86(23):9094–8.

22. Tuerk C, Gold L. Systematic evolution of ligands by exponential enrichment: RNA ligands to bacteriophage T4 DNA polymerase. Science. 1990 Aug 3;249(4968):505–10.

23. Ellington AD, Szostak JW. In vitro selection of RNA molecules that bind specific ligands. Nature. 1990 Aug 30;346(6287):818–22.

24. Wang Z, Rolish ME, Yeo G, Tung V, Mawson M, Burge CB. Systematic identification and analysis of exonic splicing silencers. Cell. 2004 Dec 17;119(6):831–45.

25. Chen, I-Tsuen, Chasin, Lawrence A. Direct Selection for Mutations Affecting Specific Splice Sites in a Hamster Dihydrofolate Reductase Minigene. Molecular and Cellular Biology. 1993;13:289–300.

26. Junqueira D, Cilenti L, Musumeci L, Sedivy JM, Zervos AS. Random mutagenesis of PDZOmi domain and selection of mutants that specifically bind the Myc proto-oncogene and induce apoptosis. Oncogene. 2003 May;22(18):2772–81.

27. Patwardhan RP, Lee C, Litvin O, Young DL, Pe'er D, Shendure J. High-resolution analysis of DNA regulatory elements by synthetic saturation mutagenesis. Nat Biotechnol. 2009 Dec;27(12):1173–5.

28. Inoue F, Ahituv N. Decoding enhancers using massively parallel reporter assays. Genomics. 2015 Sep;106(3):159–64.

29. Patwardhan RP, Hiatt JB, Witten DM, Kim MJ, Smith RP, May D, et al. Massively parallel functional dissection of mammalian enhancers in vivo. Nat Biotechnol. 2012 Feb 26;30(3):265–70.

30. Melnikov A, Murugan A, Zhang X, Tesileanu T, Wang L, Rogov P, et al. Systematic dissection and optimization of inducible enhancers in human cells using a massively parallel reporter assay. Nat Biotechnol. 2012 Feb 26;30(3):271–7.

31. Ke S, Shang S, Kalachikov SM, Morozova I, Yu L, Russo JJ, et al. Quantitative evaluation of all hexamers as exonic splicing elements. Genome Res. 2011 Aug;21(8):1360–74.

32. Fowler DM, Araya CL, Fleishman SJ, Kellogg EH, Stephany JJ, Baker D, et al. High-resolution mapping of protein sequence-function relationships. Nat Methods. 2010 Sep;7(9):741–6.

33. Ernst A, Gfeller D, Kan Z, Seshagiri S, M. Kim P, D. Bader G, et al. Coevolution of PDZ domain–ligand interactions analyzed by high-throughput phage display and deep sequencing. Molecular BioSystems. 2010;6(10):1782–90.

34. Experimental illumination of a fitness landscape | PNAS [Internet]. [cited 2025 May 25]. Available from: https://www.pnas.org/doi/abs/10.1073/pnas.1016024108

35. Vanella R, Kovacevic G, Doffini V, Fernández de Santaella J, Nash MA. High-throughput screening, next generation sequencing and machine learning: advanced methods in enzyme engineering. Chem Commun (Camb). 2022 Feb 17;58(15):2455–67.

36. Weile J, Roth FP. Multiplexed assays of variant effects contribute to a growing genotype-phenotype atlas. Hum Genet. 2018 Sep;137(9):665–78.

37. Araya CL, Fowler DM. Deep mutational scanning: assessing protein function on a massive scale. Trends Biotechnol. 2011 Sep;29(9):435–42.

38. Emili AQ, Cagney G. Large-scale functional analysis using peptide or protein arrays. Nat Biotechnol. 2000 Apr;18(4):393–7.

39. Uetz P, Giot L, Cagney G, Mansfield TA, Judson RS, Knight JR, et al. A comprehensive analysis of protein–protein interactions in Saccharomyces cerevisiae. Nature. 2000 Feb;403(6770):623–7.

40. Liu X, Abad L, Chatterjee L, Cristea IM, Varjosalo M. Mapping protein–protein interactions by mass spectrometry. Mass Spectrometry Reviews [Internet]. [cited 2025 May 26];n/a(n/a). Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/mas.21887

41. Smits AH, Vermeulen M. Characterizing Protein–Protein Interactions Using Mass Spectrometry: Challenges and Opportunities. Trends in Biotechnology. 2016 Oct;34(10):825–34.

42. Rouet R, Jackson KJL, Langley DB, Christ D. Next-Generation Sequencing of Antibody Display Repertoires. Front Immunol. 2018 Feb 2;9:118.

43. Younger D, Berger S, Baker D, Klavins E. High-throughput characterization of protein-protein interactions by reprogramming yeast mating. Proc Natl Acad Sci USA. 2017 14;114(46):12166–71.

44. Brückner A, Polge C, Lentze N, Auerbach D, Schlattner U. Yeast Two-Hybrid, a Powerful Tool for Systems Biology. International Journal of Molecular Sciences. 2009 Jun;10(6):2763–88.

45. Luck K, Kim DK, Lambourne L, Spirohn K, Begg BE, Bian W, et al. A reference map of the human binary protein interactome. Nature. 2020 Apr;580(7803):402–8.

46. Diss G, Lehner B. The genetic landscape of a physical interaction. Barkai N, editor. eLife. 2018 Apr 11;7:e32472.

47. Erffelinck ML, Ribeiro B, Perassolo M, Pauwels L, Pollier J, Storme V, et al. A user-friendly platform for yeast two-hybrid library screening using next generation sequencing. Moses AM, editor. PLoS ONE. 2018 Dec 21;13(12):e0201270.

48. Jin F, Avramova L, Huang J, Hazbun T. A yeast two-hybrid smart-pool-array system for protein-interaction mapping. Nat Methods. 2007 May;4(5):405–7.

49. Rajagopala SV, Uetz P. Analysis of Protein–Protein Interactions Using Array-Based Yeast Two-Hybrid Screens. In: Stagljar I, editor. Yeast Functional Genomics and Proteomics [Internet]. Totowa, NJ: Humana Press; 2009 [cited 2020 Jun 29]. p. 223–45. (Methods in Molecular Biology; vol. 548). Available from: http://link.springer.com/10.1007/978-1-59745-540-4_13

50. Trigg SA, Garza RM, MacWilliams A, Nery JR, Bartlett A, Castanon R, et al. CrY2H-seq: a massively multiplexed assay for deep-coverage interactome mapping. Nat Methods. 2017 Aug;14(8):819–25.

51. Weimann M, Grossmann A, Woodsmith J, Özkan Z, Birth P, Meierhofer D, et al. A Y2H-seq approach defines the human protein methyltransferase interactome. Nat Methods. 2013 Apr;10(4):339–42.

52. Yachie N, Petsalaki E, Mellor JC, Weile J, Jacob Y, Verby M, et al. Pooled-matrix protein interaction screens using Barcode Fusion Genetics. Mol Syst Biol. 2016 Apr 22;12(4):863.

53. Yang F, Lei Y, Zhou M, Yao Q, Han Y, Wu X, et al. Development and application of a recombination-based library versus library high- throughput

yeast two-hybrid (RLL-Y2H) screening system. Nucleic Acids Res. 2018 16;46(3):e17.

54. Yang JS, Garriga-Canut M, Link N, Carolis C, Broadbent K, Beltran-Sastre V, et al. rec-YnH enables simultaneous many-by-many detection of direct protein–protein and protein–RNA interactions. Nat Commun. 2018 Dec;9(1):3747.

55. Yu H, Tardivo L, Tam S, Weiner E, Gebreab F, Fan C, et al. Next-generation sequencing to generate interactome datasets. Nat Methods. 2011 Jun;8(6):478–80.

56. Banerjee S, Velásquez-Zapata V, Fuerst G, Elmore JM, Wise RP. NGPINT: a next-generation protein–protein interaction software. Briefings in Bioinformatics. 2021 Jul 20;22(4):bbaa351.

57. Velásquez-Zapata V, Elmore JM, Banerjee S, Dorman KS, Wise RP. Next-generation yeast-two-hybrid analysis with Y2H-SCORES identifies novel interactors of the MLA immune receptor. Przytycka TM, editor. PLoS Comput Biol. 2021 Apr 2;17(4):e1008890.

58. Boldridge WC, Ljubetič A, Kim H, Lubock N, Szilágyi D, Lee J, et al. A multiplexed bacterial two-hybrid for rapid characterization of protein-protein interactions and iterative protein design. Nat Commun. 2023 Aug 2;14(1):4636.

59. Quartley E, Alexandrov A, Mikucki M, Buckner FS, Hol WG, DeTitta GT, et al. Heterologous Expression of L. major proteins in S. cerevisiae: a test of solubility, purity, and gene recoding. J Struct Funct Genomics. 2009 Sep;10(3):233–47.

60. Sharon E, Kalma Y, Sharp A, Raveh-Sadka T, Levo M, Zeevi D, et al. Inferring gene regulatory logic from high-throughput measurements of thousands of systematically designed promoters. Nat Biotechnol. 2012 May 20;30(6):521–30.

61. Mogno I, Kwasnieski JC, Cohen BA. Massively parallel synthetic promoter assays reveal the in vivo effects of binding site variants. Genome Res. 2013 Nov;23(11):1908–15.

62. Noderer WL, Flockhart RJ, Bhaduri A, Diaz de Arce AJ, Zhang J, Khavari PA, et al. Quantitative analysis of mammalian translation initiation sites by FACS-seq. Mol Syst Biol. 2014 Aug 28;10(8):748.

63. Shalem O, Sharon E, Lubliner S, Regev I, Lotan-Pompan M, Yakhini Z, et al. Systematic dissection of the sequence determinants of gene 3' end mediated expression control. PLoS Genet. 2015 Apr;11(4):e1005147.

64. Rosenberg AB, Patwardhan RP, Shendure J, Seelig G. Learning the sequence determinants of alternative splicing from millions of random sequences. Cell. 2015 Oct 22;163(3):698–711.

65. Shen SQ, Myers CA, Hughes AEO, Byrne LC, Flannery JG, Corbo JC. Massively parallel cis-regulatory analysis in the mammalian central nervous system. Genome Res. 2016 Feb;26(2):238–55.

66. Soemedi R, Cygan KJ, Rhine CL, Wang J, Bulacan C, Yang J, et al. Pathogenic variants that alter protein code often disrupt splicing. Nat Genet. 2017 Jun;49(6):848–55.

67. Leppek K, Byeon GW, Kladwang W, Wayment-Steele HK, Kerr CH, Xu AF, et al. Combinatorial optimization of mRNA structure, stability, and translation for RNA-based therapeutics. Nat Commun. 2022 Mar 22;13(1):1536.

68. Chiang HL, Chen YT, Su JY, Lin HN, Yu CHA, Hung YJ, et al. Mechanism and modeling of human disease-associated near-exon intronic variants that perturb RNA splicing. Nat Struct Mol Biol. 2022 Nov;29(11):1043–55.

69. Mikl M, Eletto D, Nijim M, Lee M, Lafzi A, Mhamedi F, et al. A massively parallel reporter assay reveals focused and broadly encoded RNA localization signals in neurons. Nucleic Acids Res. 2022 Oct 14;50(18):10643–64.

70. Alipanahi B, Delong A, Weirauch MT, Frey BJ. Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. Nat Biotechnol. 2015 Aug;33(8):831–8.

71. Zhou J, Troyanskaya OG. Predicting effects of noncoding variants with deep learning-based sequence model. Nat Methods. 2015 Oct;12(10):931–4.

72. Kleftogiannis D, Kalnis P, Bajic VB. DEEP: a general computational framework for predicting enhancers. Nucleic Acids Res. 2015 Jan;43(1):e6.

73. Cuperus JT, Groves B, Kuchina A, Rosenberg AB, Jojic N, Fields S, et al. Deep learning of the regulatory grammar of yeast 5' untranslated regions from 500,000 random sequences. Genome Res. 2017 Dec;27(12):2015–24.

74. Bogard N, Linder J, Rosenberg AB, Seelig G. A Deep Neural Network for Predicting and Engineering Alternative Polyadenylation. Cell. 2019 Jun 27;178(1):91-106.e23.

75. Sample PJ, Wang B, Reid DW, Presnyak V, McFadyen IJ, Morris DR, et al. Human 5' UTR design and variant effect prediction from a massively parallel translation assay. Nat Biotechnol. 2019 Jul;37(7):803–9.

76. Vainberg Slutskin I, Weinberger A, Segal E. Sequence determinants of polyadenylation-mediated regulation. Genome Res. 2019 Oct;29(10):1635–47.

77. Cheng J, Çelik MH, Kundaje A, Gagneur J. MTSplice predicts effects of genetic variants on tissue-specific splicing. Genome Biol. 2021 Mar 31;22(1):94.

78. Eraslan G, Avsec Ž, Gagneur J, Theis FJ. Deep learning: new computational modelling techniques for genomics. Nat Rev Genet. 2019 Jul;20(7):389–403.

79. Freschlin CR, Fahlberg SA, Romero PA. Machine learning to navigate fitness landscapes for protein engineering. Curr Opin Biotechnol. 2022 Jun;75:102713.

80. Chen L, Zhang Z, Li Z, Li R, Huo R, Chen L, et al. Learning protein fitness landscapes with deep mutational scanning data from multiple sources. Cell Systems. 2023 Aug 16;14(8):706-721.e5.

81. Quang D, Xie X. DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. Nucleic Acids Res. 2016 Jun 20;44(11):e107.

82. Angermueller C, Lee HJ, Reik W, Stegle O. DeepCpG: accurate prediction of single-cell DNA methylation states using deep learning. Genome Biol. 2017 Apr 11;18(1):67.

83. Zhao L, Wang J, Hu Y, Cheng L. Conjoint Feature Representation of GO and Protein Sequence for PPI Prediction Based on an Inception RNN Attention Network. Molecular Therapy Nucleic Acids. 2020 Dec 4;22:198–208.

84. Rives A, Meier J, Sercu T, Goyal S, Lin Z, Liu J, et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. Proceedings of the National Academy of Sciences. 2021 Apr;118(15):e2016239118.

85. Lin Z, Akin H, Rao R, Hie B, Zhu Z, Lu W, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. Science. 2023 Mar 17;379(6637):1123–30.

86. Biswas S, Khimulya G, Alley EC, Esvelt KM, Church GM. Low-N protein engineering with data-efficient deep learning. Nature Methods. 2021 Apr;18(4):389–96.

87. Karollus A, Avsec Ž, Gagneur J. Predicting mean ribosome load for 5'UTR of any length using deep learning. PLoS Comput Biol. 2021 May;17(5):e1008982.

88. Consens ME, Dufault C, Wainberg M, Forster D, Karimzadeh M, Goodarzi H, et al. To Transformers and Beyond: Large Language Models for the Genome [Internet]. arXiv; 2023 [cited 2025 May 25]. Available from: http://arxiv.org/abs/2311.07621

89. Tang T, Zhang X, Liu Y, Peng H, Zheng B, Yin Y, et al. Machine learning on protein–protein interaction prediction: models, challenges and trends. Briefings in Bioinformatics. 2023 Mar 1;24(2):bbad076.

90. Li S, Wu S, Wang L, Li F, Jiang H, Bai F. Recent advances in predicting protein–protein interactions with the aid of artificial intelligence algorithms. Current Opinion in Structural Biology. 2022 Apr 1;73:102344.

91. Guo Z, Yamaguchi R. Machine learning methods for protein-protein binding affinity prediction in protein design. Front Bioinform. 2022 Dec 16;2:1065703.

92. Zhang J, Durham J, Qian Cong. Revolutionizing protein–protein interaction prediction with deep learning. Current Opinion in Structural Biology. 2024 Apr 1;85:102775.

93. Ma S, Kosorok MR. Detection of gene pathways with predictive power for breast cancer prognosis. BMC Bioinformatics. 2010 Jan 1;11:1.

94. Hart GT, Ramani AK, Marcotte EM. How complete are current yeast and human protein-interaction networks? Genome Biol. 2006 Dec 1;7(11):120.

95. Smialowski P, Pagel P, Wong P, Brauner B, Dunger I, Fobo G, et al. The Negatome database: a reference set of non-interacting protein pairs. Nucleic Acids Res. 2010 Jan;38(Database issue):D540–4.

96. Blohm P, Frishman G, Smialowski P, Goebels F, Wachinger B, Ruepp A, et al. Negatome 2.0: a database of non-interacting proteins derived by literature mining, manual annotation and protein structure analysis. Nucleic Acids Res. 2014 Jan;42(Database issue):D396-400.

97. Hermjakob H, Montecchi-Palazzi L, Lewington C, Mudali S, Kerrien S, Orchard S, et al. IntAct: an open source molecular interaction database. Nucleic Acids Res. 2004 Jan 1;32(Database issue):D452–5.

98. Yang X, Coulombe-Huntington J, Kang S, Sheynkman GM, Hao T, Richardson A, et al. Widespread Expansion of Protein Interaction Capabilities by Alternative Splicing. Cell. 2016 Feb 11;164(4):805–17.

99. Ben-Hur A, Noble WS. Choosing negative examples for the prediction of protein-protein interactions. BMC Bioinformatics. 2006 Mar 20;7 Suppl 1(Suppl 1):S2.

100. Trabuco LG, Betts MJ, Russell RB. Negative protein–protein interaction datasets derived from large-scale two-hybrid experiments. Methods. 2012 Dec 1;58(4):343–8.

101. Neumann D, Roy S, Minhas FUAA, Ben-Hur A. On the choice of negative examples for prediction of host-pathogen protein interactions. Front Bioinform. 2022;2:1083292.

102. Subedi S, Shukla H, Uversky VN, Tripathi T. Chapter 12 - Physical principles and molecular interactions underlying protein phase separation. In: Tripathi T, Uversky VN, editors. The Three Functional States of Proteins [Internet]. Academic Press; 2025 [cited 2025 May 26]. p. 197–212. Available from: https://www.sciencedirect.com/science/article/pii/B9780443218095000089

103. Eid FE, ElHefnawi M, Heath LS. DeNovo: virus-host sequence-based protein–protein interaction prediction. Bioinformatics. 2016 Apr 15;32(8):1144–50.

104. Dey L, Chakraborty S, Mukhopadhyay A. Machine learning techniques for sequence-based prediction of viral-host interactions between SARS-CoV-2 and human proteins. Biomed J. 2020 Oct;43(5):438–50.

105. Dunham B, Ganapathiraju MK. Benchmark Evaluation of Protein–Protein Interaction Prediction Algorithms. Molecules. 2021 Dec;27(1):41.

106. Bernett J, Blumenthal DB, List M. Cracking the black box of deep sequence-based protein–protein interaction prediction. Briefings in Bioinformatics. 2024 Mar 1;25(2):bbae076.

107. Park Y, Marcotte EM. A flaw in the typical evaluation scheme for pair-input computational predictions. Nat Methods. 2012 Dec;9(12):1134–6.

108. Lv G, Hu Z, Bi Y, Zhang S. Learning Unknown from Correlations: Graph Neural Network for Inter-novel-protein Interaction Prediction [Internet]. arXiv; 2021 [cited 2025 May 16]. Available from: http://arxiv.org/abs/2105.06709

109. Deng Y, Xu X, Qiu Y, Xia J, Zhang W, Liu S. A multimodal deep learning framework for predicting drug-drug interaction events. Bioinformatics. 2020 Aug 1;36(15):4316–22.

110. Ghosh S, Mitra P. MaTPIP: A deep-learning architecture with eXplainable AI for sequence-driven, feature mixed protein-protein interaction prediction. Comput Methods Programs Biomed. 2024 Feb;244:107955.

111. Szymborski J, Emad A. RAPPPID: towards generalizable protein interaction prediction with AWD-LSTM twin networks. Bioinformatics. 2022 Aug 10;38(16):3958–67.

112. Szymborski J, Emad A. INTREPPPID—an orthologue-informed quintuplet network for cross-species prediction of protein–protein interaction. Briefings in Bioinformatics. 2024 Sep 1;25(5):bbae405.

113. Cheng P, Mao C, Tang J, Yang S, Cheng Y, Wang W, et al. Zero-shot prediction of mutation effects with multimodal deep representation learning guides protein engineering. Cell Res. 2024 Sep;34(9):630–47.

114. Gao Z, Jiang C, Zhang J, Jiang X, Li L, Zhao P, et al. Hierarchical graph learning for protein–protein interaction. Nat Commun. 2023 Feb 25;14(1):1093.

115. Albu AI, Bocicor MI, Czibula G. *MM-StackEns*: A new deep multimodal stacked generalization approach for protein–protein interaction prediction. Computers in Biology and Medicine. 2023 Feb 1;153:106526.

116. Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, et al. Highly accurate protein structure prediction with AlphaFold. Nature. 2021 Aug;596(7873):583–9.

117. Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, et al. Applying and improving AlphaFold at CASP14. Proteins. 2021 Dec;89(12):1711–21.

118. Evans R, O'Neill M, Pritzel A, Antropova N, Senior A, Green T, et al. Protein complex prediction with AlphaFold-Multimer [Internet]. bioRxiv; 2022 [cited 2022 Jul 23]. Available from: https://www.biorxiv.org/content/10.1101/2021.10.04.463034v2

119. Yang J, Anishchenko I, Park H, Peng Z, Ovchinnikov S, Baker D. Improved protein structure prediction using predicted interresidue orientations. Proceedings of the National Academy of Sciences. 2020 Jan 21;117(3):1496–503.

120. Krishna R, Wang J, Ahern W, Sturmfels P, Venkatesh P, Kalvet I, et al. Generalized biomolecular modeling and design with RoseTTAFold All-Atom. Science. 2024 Mar 7;384(6693):eadl2528.

121. Abramson J, Adler J, Dunger J, Evans R, Green T, Pritzel A, et al. Accurate structure prediction of biomolecular interactions with AlphaFold 3. Nature. 2024 Jun;630(8016):493–500.

122. Humphreys IR, Pei J, Baek M, Krishnakumar A, Anishchenko I, Ovchinnikov S, et al. Computed structures of core eukaryotic protein complexes. Science. 2021 Nov;374(6573):eabm4805.

123. Burke DF, Bryant P, Barrio-Hernandez I, Memon D, Pozzati G, Shenoy A, et al. Towards a structurally resolved human protein interaction network. Nat Struct Mol Biol. 2023 Feb;30(2):216–25.

124. Yu D, Chojnowski G, Rosenthal M, Kosinski J. AlphaPulldown-a python package for protein-protein interaction screens using AlphaFold-Multimer. Bioinformatics. 2023 Jan 1;39(1):btac749.

125. Guzmán-Vega FJ, Arold ST. AlphaCRV: a pipeline for identifying accurate binder topologies in mass-modeling with AlphaFold. Bioinform Adv. 2024;4(1):vbae131.

126. Zhou F, Guo S, Peng X, Zhang S, Men C, Duan X, et al. Benchmarking AlphaFold3-like Methods for Protein-Peptide Complex Prediction [Internet].

bioRxiv; 2025 [cited 2025 May 14]. p. 2025.03.09.642277. Available from: https://www.biorxiv.org/content/10.1101/2025.03.09.642277v3

127. Manshour N, Ren JZ, Esmaili F, Bergstrom E, Xu D. Comprehensive Evaluation of AlphaFold-Multimer, AlphaFold3 and ColabFold, and Scoring Functions in Predicting Protein-Peptide Complex Structures [Internet]. bioRxiv; 2024 [cited 2025 May 14]. p. 2024.11.11.622992. Available from: https://www.biorxiv.org/content/10.1101/2024.11.11.622992v1

128. Schmid EW, Walter JC. Predictomes, a classifier-curated database of AlphaFold-modeled protein-protein interactions. Mol Cell. 2025 Mar 20;85(6):1216-1232.e5.

129. Mischley V, Maier J, Chen J, Karanicolas J. PPIscreenML: Structure-based screening for protein-protein interactions using AlphaFold. bioRxiv. 2024 Apr 30;2024.03.16.585347.

130. Bryant P, Pozzati G, Elofsson A. Improved prediction of protein-protein interactions using AlphaFold2. Nature Communications. 2022 Mar;13(1):1265.

131. Hu W, Ohue M. SpatialPPI: Three-dimensional space protein-protein interaction prediction with AlphaFold Multimer. Comput Struct Biotechnol J. 2024 Dec;23:1214–25.

132. Strom JM, Luck K. Bias in, bias out – AlphaFold-Multimer and the structural complexity of protein interfaces. Current Opinion in Structural Biology. 2025 Apr 1;91:103002.

133. Johansson-Åkhe I, Mirabello C, Wallner B. InterPepRank: Assessment of Docked Peptide Conformations by a Deep Graph Network. Front Bioinform [Internet]. 2021 Oct 25 [cited 2025 May 26];1. Available from: https://www.frontiersin.org/journals/bioinformatics/articles/10.3389/fbinf.2021.763102/full

134. Réau M, Renaud N, Xue LC, Bonvin AMJJ. DeepRank-GNN: a graph neural network framework to learn patterns in protein–protein interfaces. Bioinformatics. 2023 Jan 1;39(1):btac759.

135. Renaud N, Geng C, Georgievska S, Ambrosetti F, Ridder L, Marzella DF, et al. DeepRank: a deep learning framework for data mining 3D protein-protein interfaces. Nat Commun. 2021 Dec 3;12(1):7068.

136. Wang X, Flannery ST, Kihara D. Protein Docking Model Evaluation by Graph Neural Networks. Front Mol Biosci [Internet]. 2021 May 25 [cited 2025 May 26];8. Available from: https://www.frontiersin.org/journals/molecular-biosciences/articles/10.3389/fmolb.2021.647915/full

137. Wang X, Terashi G, Christoffer CW, Zhu M, Kihara D. Protein docking model evaluation by 3D deep convolutional neural networks. Bioinformatics. 2020 Apr 1;36(7):2113–8.

138. Outeiral C, Nissley DA, Deane CM. Current structure predictors are not learning the physics of protein folding. Bioinformatics. 2022 Apr;38(7):1881–7.

139. Pak MA, Markhieva KA, Novikova MS, Petrov DS, Vorobyev IS, Maksimova ES, et al. Using AlphaFold to predict the impact of single mutations on protein stability and function [Internet]. bioRxiv; 2021 [cited 2022 Jul 23]. Available from: https://www.biorxiv.org/content/10.1101/2021.09.19.460937v1

140. Baryshev A, La Fleur A, Groves B, Michel C, Baker D, Ljubetič A, et al. Massively parallel measurement of protein–protein interactions by sequencing using MP3-seq. Nat Chem Biol. 2024 Nov;20(11):1514–23.

141. Linder J, La Fleur A, Chen Z, Ljubeti A, Baker D, Kannan S, et al. Interpreting Neural Networks for Biological Sequences by Learning Stochastic Masks. Nat Mach Intell. 2022 Jan;4(1):41–54.

142. Love MI, Huber W, Anders S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biol. 2014 Dec;15(12):550.

143. DasGupta A. Chapter 18 The Trimmed Mean. In: Asymptotic Theory of Statistics and Probability. Springer Science+Business Media, LLC; 2008. p. 271–8.

144. Lebar T, Lainšček D, Merljak E, Aupič J, Jerala R. A tunable orthogonal coiled-coil interaction toolbox for engineering mammalian cells. Nature Chemical Biology. 2020 May;16(5):513–9.

145. Thomas F, Boyle AL, Burton AJ, Woolfson DN. A Set of *de Novo* Designed Parallel Heterodimeric Coiled Coils with Quantified Dissociation Constants in the Micromolar to Sub-nanomolar Regime. J Am Chem Soc. 2013 Apr 3;135(13):5161–6.

146. Berger S, Procko E, Margineantu D, Lee EF, Shen BW, Zelter A, et al. Computationally designed high specificity inhibitors delineate the roles of BCL2 family proteins in cancer. Elife. 2016 02;5.

147. Rogers JM, Wong CT, Clarke J. Coupled folding and binding of the disordered protein PUMA does not require particular residual structure. J Am Chem Soc. 2014 Apr 9;136(14):5197–200.

148. Plaper T, Aupič J, Dekleva P, Lapenta F, Keber MM, Jerala R, et al. Coiled-coil heterodimers with increased stability for cellular regulation and sensing SARS-CoV-2 spike protein-mediated cell fusion. Sci Rep. 2021 Apr 28;11(1):9136.

149. Chen Z, Boyken SE, Jia M, Busch F, Flores-Solis D, Bick MJ, et al. Programmable design of orthogonal protein heterodimers. Nature. 2019 Jan;565(7737):106–11.

150. Ljubetič A, Gradišar H, Jerala R. Advances in design of protein folds and assemblies. Current Opinion in Chemical Biology. 2017 Oct;40:65–71.

151. Potapov V, Kaplan JB, Keating AE. Data-Driven Prediction and Design of bZIP Coiled-Coil Interactions. PLOS Computational Biology. 2015 Feb 19;11(2):e1004046.

152. Mason JM, Schmitz MA, Müller KM, Arndt KM. Semirational design of Jun-Fos coiled coils with increased affinity: Universal implications for leucine zipper prediction and design. Proc Natl Acad Sci U S A. 2006 Jun 13;103(24):8989–94.

153. Dallago C, Schütze K, Heinzinger M, Olenyi T, Littmann M, Lu AX, et al. Learned Embeddings from Deep Learning to Visualize and Predict Protein Sets. Current Protocols. 2021;1(5):e113.

154. Hsu C, Nisonoff H, Fannjiang C, Listgarten J. Learning protein fitness models from evolutionary and assay-labeled data. Nature Biotechnology. 2022 Jul;40(7):1114–22.

155. Sarfati H, Naftaly S, Papo N, Keasar C. Predicting mutant outcome by combining deep mutational scanning and machine learning. Proteins. 2022 Jan;90(1):45–57.

156. Jankauskaitė J, Jiménez-García B, Dapkūnas J, Fernández-Recio J, Moal IH. SKEMPI 2.0: an updated benchmark of changes in protein–protein binding energy, kinetics and thermodynamics upon mutation. Bioinformatics. 2019 Feb 1;35(3):462–9.

157. del-Toro N, Duesbury M, Koch M, Perfetto L, Shrivastava A, Ochoa D, et al. Capturing variation impact on molecular interactions in the IMEx Consortium mutations data set. Nat Commun. 2019 Jan 2;10(1):10.

158. Gelman S, Fahlberg SA, Heinzelman P, Romero PA, Gitter A. Neural networks to learn protein sequence–function relationships from deep mutational scanning data. Proceedings of the National Academy of Sciences. 2021 Nov;118(48):e2104878118.

159. Sandhu M, Mater AC, Matthews DS, Spence MA, Lenskiy AA, Jackson C. Investigating the determinants of performance in machine learning for protein fitness prediction [Internet]. bioRxiv; 2025 [cited 2025 May 26]. p. 2020.09.30.319780. Available from: https://www.biorxiv.org/content/10.1101/2020.09.30.319780v4

160. Tuncbag N, Keskin O, Nussinov R, Gursoy A. Prediction of Protein Interactions by Structural Matching: Prediction of PPI Networks and the Effects of Mutations on PPIs that Combines Sequence and Structural Information. Methods Mol Biol. 2017;1558:255–70.

161. Liu Z, Qian W, Cai W, Song W, Wang W, Maharjan DT, et al. Inferring the Effects of Protein Variants on Protein–Protein Interactions with Interpretable Transformer Representations. Research. 2023 Sep 11;6:0219.

162. Zhou Y, Myung Y, Rodrigues CHM, Ascher DB. DDMut-PPI: predicting effects of mutations on protein–protein interactions using graph-based deep learning. Nucleic Acids Research. 2024 Jul 5;52(W1):W207–14.

163. Livesey BJ, Marsh JA. Using deep mutational scanning to benchmark variant effect predictors and identify disease mutations. Molecular Systems Biology. 2020 Jul;16(7):e9380.

164. Livesey BJ, Marsh JA. Interpreting protein variant effects with computational predictors and deep mutational scanning. Disease Models & Mechanisms. 2022 Jun;15(6):dmm049510.

165. Ng PC, Henikoff S. SIFT: predicting amino acid changes that affect protein function. Nucleic Acids Research. 2003 Jul;31(13):3812–4.

166. Adzhubei IA, Schmidt S, Peshkin L, Ramensky VE, Gerasimova A, Bork P, et al. A method and server for predicting damaging missense mutations. Nat Methods. 2010 Apr;7(4):248–9.

167. Riesselman AJ, Ingraham JB, Marks DS. Deep generative models of genetic variation capture the effects of mutations. Nature Methods. 2018 Oct;15(10):816–22.

168. Hopf TA, Ingraham JB, Poelwijk FJ, Schärfe CPI, Springer M, Sander C, et al. Mutation effects predicted from sequence co-variation. Nature biotechnology. 2017 Feb;35(2):128–35.

169. Livesey BJ, Marsh JA. Updated benchmarking of variant effect predictors using deep mutational scanning. Mol Syst Biol. 2023 Aug 8;19(8):e11474.

170. Livesey BJ, Marsh JA. Variant effect predictor correlation with functional assays is reflective of clinical classification performance. Genome Biol. 2025 Apr 22;26(1):104.

171. Neubauer K, Zieger B. The Mammalian Septin Interactome. Frontiers in Cell and Developmental Biology [Internet]. 2017 [cited 2022 Jul 23];5. Available from: https://www.frontiersin.org/articles/10.3389/fcell.2017.00003

172. Kuo YC, Lin YH, Chen HI, Wang YY, Chiou YW, Lin HH, et al. SEPT12 mutations cause male infertility with defective sperm annulus. Human Mutation. 2012 Apr;33(4):710–9.

173. Kinoshita M. The septins. Genome Biology. 2003 Oct;4(11):236.

174. Valadares NF, d' Muniz Pereira H, Ulian Araujo AP, Garratt RC. Septin structure and filament assembly. Biophysical Reviews. 2017 Oct;9(5):481–500.

175. Sirajuddin M, Farkasovsky M, Hauer F, Kühlmann D, Macara IG, Weyand M, et al. Structural insight into filament formation by mammalian septins. Nature. 2007 Sep;449(7160):311–5.

176. Mendonça DC, Macedo JN, Guimarães SL, Barroso da Silva FL, Cassago A, Garratt RC, et al. A revised order of subunits in mammalian septin complexes. Cytoskeleton. 2019;76(9–10):457–66.

177. Castro DKS do V, da Silva SM de O, Pereira HD, Macedo JNA, Leonardo DA, Valadares NF, et al. A complete compendium of crystal structures for the human SEPT3 subgroup reveals functional plasticity at a specific septin interface. IUCrJ. 2020 Mar;7(Pt 3):462–79.

178. Versele M, Thorner J. Some assembly required: yeast septins provide the instruction manual. Trends in cell biology. 2005 Aug;15(8):414–24.

179. Fragoza R, Das J, Wierbowski SD, Liang J, Tran TN, Liang S, et al. Extensive disruption of protein interactions by genetic variants across the allele frequency spectrum in human populations. Nature Communications. 2019 Sep;10(1):4141.

180. Faure AJ, Schmiedel JM, Baeza-Centurion P, Lehner B. DiMSum: an error model and pipeline for analyzing deep mutational scanning data and diagnosing common experimental pathologies. Genome Biology. 2020 Aug;21(1):207.

181. David A, Sternberg MJE. The Contribution of Missense Mutations in Core and Rim Residues of Protein–Protein Interfaces to Human Disease. Journal of Molecular Biology. 2015 Aug;427(17):2886–98.

182. Frazer J, Notin P, Dias M, Gomez A, Min JK, Brock K, et al. Disease variant prediction with deep generative models of evolutionary data. Nature. 2021 Nov;599(7883):91–5.

183. Hecht M, Bromberg Y, Rost B. Better prediction of functional effects for sequence variants. BMC genomics. 2015;16 Suppl 8:S1.

184. Capriotti E, Calabrese R, Fariselli P, Martelli PL, Altman RB, Casadio R. WS-SNPs&GO: a web server for predicting the deleterious effect of human protein variants using functional annotation. BMC Genomics. 2013 May 28;14(3):S6.

185. Zhou G, Chen M, Ju CJT, Wang Z, Jiang JY, Wang W. Mutation effect estimation on protein-protein interactions using deep contextualized

representation learning. NAR genomics and bioinformatics. 2020 Jun;2(2):lqaa015.

186. Luo Y, Jiang G, Yu T, Liu Y, Vo L, Ding H, et al. ECNet is an evolutionary context-integrated deep learning framework for protein engineering. Nature Communications. 2021 Sep;12(1):5743.

187. Li L, Jamieson K, DeSalvo G, Rostamizadeh A, Talwalkar A. Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization. Journal of Machine Learning Research. 2018;18(185):1–52.

188. Karczewski KJ, Francioli LC, Tiao G, Cummings BB, Alföldi J, Wang Q, et al. The mutational constraint spectrum quantified from variation in 141,456 humans. Nature. 2020 May;581(7809):434–43.

189. Mater AC, Sandhu M, Jackson C. The NK Landscape as a Versatile Benchmark for Machine Learning Driven Protein Engineering [Internet]. bioRxiv; 2020 [cited 2022 Jul 23]. Available from: https://www.biorxiv.org/content/10.1101/2020.09.30.319780v3

190. Novakovsky G, Dexter N, Libbrecht MW, Wasserman WW, Mostafavi S. Obtaining genetics insights from deep learning via explainable artificial intelligence. Nat Rev Genet. 2023 Feb;24(2):125–37.

191. Azodi CB, Tang J, Shiu SH. Opening the Black Box: Interpretable Machine Learning for Geneticists. Trends Genet. 2020 Jun;36(6):442–55.

192. Karim MR, Islam T, Shajalal M, Beyan O, Lange C, Cochez M, et al. Explainable AI for Bioinformatics: Methods, Tools and Applications. Briefings in Bioinformatics. 2023 Sep 1;24(5):bbad236.

193. Klie A, Laub D, Talwar JV, Stites H, Jores T, Solvason JJ, et al. Predictive analyses of regulatory sequences with EUGENe. Nat Comput Sci. 2023 Nov;3(11):946–56.

194. Reimão-Pinto MM, Castillo-Hair SM, Seelig G, Schier AF. The regulatory landscape of 5′ UTRs in translational control during zebrafish embryogenesis [Internet]. bioRxiv; 2023 [cited 2024 Jul 24]. p. 2023.11.23.568470. Available from: https://www.biorxiv.org/content/10.1101/2023.11.23.568470v1

195. Novakovsky G, Fornes O, Saraswat M, Mostafavi S, Wasserman WW. ExplaiNN: interpretable and transparent neural networks for genomics. Genome Biology. 2023 Jun 27;24(1):154.

196. Koo PK, Eddy SR. Representation learning of genomic sequence motifs with convolutional neural networks. PLOS Computational Biology. 2019 Dec 19;15(12):e1007560.

197. Koo PK, Ploenzke M. Improving representations of genomic sequence motifs in convolutional networks with exponential activations. Nat Mach Intell. 2021 Mar;3(3):258–66.

198. Lundberg S, Lee SI. A Unified Approach to Interpreting Model Predictions. 2017 [cited 2024 Mar 26]; Available from: https://arxiv.org/abs/1705.07874

199. Avsec Ž, Agarwal V, Visentin D, Ledsam JR, Grabska-Barwinska A, Taylor KR, et al. Effective gene expression prediction from sequence by integrating long-range interactions. Nat Methods. 2021 Oct;18(10):1196–203.

200. Shrikumar A, Greenside P, Kundaje A. Learning Important Features Through Propagating Activation Differences [Internet]. arXiv; 2019 [cited 2024 Jul 24]. Available from: http://arxiv.org/abs/1704.02685

201. Carter B, Bileschi M, Smith J, Sanderson T, Bryant D, Belanger D, et al. Critiquing Protein Family Classification Models Using Sufficient Input Subsets. Journal of Computational Biology. 2020 Aug;27(8):1219–31.

202. Minnoye L, Taskiran II, Mauduit D, Fazio M, Aerschot LV, Hulselmans G, et al. Cross-species analysis of enhancer logic using deep learning. Genome Res. 2020 Dec 1;30(12):1815–34.

203. Simonyan K, Vedaldi A, Zisserman A. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps [Internet]. arXiv; 2014 [cited 2025 May 26]. Available from: http://arxiv.org/abs/1312.6034

204. Whiffin N, Karczewski KJ, Zhang X, Chothani S, Smith MJ, Evans DG, et al. Characterising the loss-of-function impact of 5' untranslated region variants in 15,708 individuals. Nat Commun. 2020 May 27;11(1):2523.

205. Calvo SE, Pagliarini DJ, Mootha VK. Upstream open reading frames cause widespread reduction of protein expression and are polymorphic among humans. Proceedings of the National Academy of Sciences. 2009 May 5;106(18):7507–12.

206. Greenside P, Shimko T, Fordyce P, Kundaje A. Discovering epistatic feature interactions from neural network models of regulatory DNA sequences. Bioinformatics. 2018 Sep 1;34(17):i629–37.

207. Seitz EE, McCandlish DM, Kinney JB, Koo PK. Interpreting cis-regulatory mechanisms from genomic deep neural networks using surrogate models. Nat Mach Intell. 2024 Jun;6(6):701–13.

208. Shrikumar A, Tian K, Avsec Ž, Shcherbina A, Banerjee A, Sharmin M, et al. Technical Note on Transcription Factor Motif Discovery from Importance Scores (TF-MoDISco) version 0.5.6.5 [Internet]. arXiv; 2020 [cited 2025 May 26]. Available from: http://arxiv.org/abs/1811.00416

209. Agarwal V, Kelley DR. The genetic and biochemical determinants of mRNA degradation rates in mammals. Genome Biol. 2022 Nov 23;23(1):245.

210. de Almeida BP, Schaub C, Pagani M, Secchia S, Furlong EEM, Stark A. Targeted design of synthetic enhancers for selected tissues in the Drosophila embryo. Nature. 2023 Dec 12;

211. Lin C, Covert I, Lee SI. On the Robustness of Removal-Based Feature Attributions [Internet]. arXiv; 2023 [cited 2025 May 26]. Available from: http://arxiv.org/abs/2306.07462

212. Chen J, Song L, Wainwright MJ, Jordan MI. Learning to Explain: An Information-Theoretic Perspective on Model Interpretation [Internet]. arXiv; 2018 [cited 2025 May 13]. Available from: http://arxiv.org/abs/1802.07814

213. Chang CH, Creager E, Goldenberg A, Duvenaud D. Explaining Image Classifiers by Counterfactual Generation [Internet]. arXiv; 2019 [cited 2025 May 26]. Available from: http://arxiv.org/abs/1807.08024

214. Understanding Deep Networks via Extremal Perturbations and Smooth Masks | IEEE Conference Publication | IEEE Xplore [Internet]. [cited 2025 May 26]. Available from: https://ieeexplore.ieee.org/document/9010039

215. Dabkowski P, Gal Y. arXiv.org. 2017 [cited 2025 May 26]. Real Time Image Saliency for Black Box Classifiers. Available from: https://arxiv.org/abs/1705.07857v1

216. Yoon J, Jordon J, Schaar M van der. INVASE: Instance-wise Variable Selection using Neural Networks. In 2018 [cited 2025 May 26]. Available from: https://openreview.net/forum?id=BJg_roAcK7

217. Zintgraf LM, Cohen TS, Adel T, Welling M. Visualizing Deep Neural Network Decisions: Prediction Difference Analysis [Internet]. arXiv; 2017 [cited 2025 May 26]. Available from: http://arxiv.org/abs/1702.04595

218. Chung J, Ahn S, Bengio Y. Hierarchical Multiscale Recurrent Neural Networks [Internet]. arXiv; 2017 [cited 2025 May 26]. Available from: http://arxiv.org/abs/1609.01704

219. Jang E, Gu S, Poole B. Categorical Reparameterization with Gumbel-Softmax [Internet]. arXiv; 2017 [cited 2025 May 26]. Available from: http://arxiv.org/abs/1611.01144

220. Springenberg JT, Dosovitskiy A, Brox T, Riedmiller M. arXiv.org. 2014 [cited 2025 May 26]. Striving for Simplicity: The All Convolutional Net. Available from: https://arxiv.org/abs/1412.6806v3

221. Sundararajan M, Taly A, Yan Q. Axiomatic Attribution for Deep Networks [Internet]. arXiv; 2017 [cited 2025 May 26]. Available from: http://arxiv.org/abs/1703.01365

222. Ancona M, Ceolini E, Öztireli C, Gross M. Towards better understanding of gradient-based attribution methods for Deep Neural Networks [Internet]. arXiv; 2018 [cited 2025 May 26]. Available from: http://arxiv.org/abs/1711.06104

223. Fong R, Patrick M, Vedaldi A. Understanding Deep Networks via Extremal Perturbations and Smooth Masks. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV) [Internet]. 2019 [cited 2025 May 26]. p. 2950–8. Available from: https://ieeexplore.ieee.org/document/9010039

224. Maguire JB, Boyken SE, Baker D, Kuhlman B. Rapid Sampling of Hydrogen Bond Networks for Computational Protein Design. J Chem Theory Comput. 2018 May 8;14(5):2751–60.

225. Parse GigaLab: 10M+ Cells in a Single Run - Redefining scRNA-seq [Internet]. Parse Biosciences. [cited 2025 May 26]. Available from: https://www.parsebiosciences.com/gigalab/

226. Jovic D, Liang X, Zeng H, Lin L, Xu F, Luo Y. Single-cell RNA sequencing technologies and applications: A brief overview. Clin Transl Med. 2022 Mar 29;12(3):e694.

227. Böyum A. Isolation of mononuclear cells and granulocytes from human blood. Isolation of monuclear cells by one centrifugation, and of granulocytes by combining centrifugation and sedimentation at 1 g. Scand J Clin Lab Invest Suppl. 1968;97:77–89.

228. Schenz J, Obermaier M, Uhle S, Weigand MA, Uhle F. Low-Density Granulocyte Contamination From Peripheral Blood Mononuclear Cells of Patients With Sepsis and How to Remove It - A Technical Report. Front Immunol. 2021;12:684119.

229. Diehl AD, Meehan TF, Bradford YM, Brush MH, Dahdul WM, Dougall DS, et al. The Cell Ontology 2016: enhanced content, modularization, and ontology interoperability. Journal of Biomedical Semantics. 2016 Jul 4;7(1):44.

230. Chen Z, Wang C, Huang S, Shi Y, Xi R. Directly selecting cell-type marker genes for single-cell clustering analyses. Cell Rep Methods. 2024 Jul 15;4(7):100810.

231. Zhang Y, Mao S, Mukherjee S, Kannan S, Seelig G. UNCURL-App: Interactive database-driven analysis of scRNA-Seq data [Internet]. bioRxiv; 2023 [cited 2025 May 26]. p. 2020.04.15.043737. Available from: https://www.biorxiv.org/content/10.1101/2020.04.15.043737v2

232. Börner K, Teichmann SA, Quardokus EM, Gee JC, Browne K, Osumi-Sutherland D, et al. Anatomical structures, cell types and biomarkers of the Human Reference Atlas. Nat Cell Biol. 2021 Nov;23(11):1117–28.

233. Hu C, Li T, Xu Y, Zhang X, Li F, Bai J, et al. CellMarker 2.0: an updated database of manually curated cell markers in human/mouse and web tools based on scRNA-seq data. Nucleic Acids Research. 2023 Jan 6;51(D1):D870–6.

234. Atallah MB, Tandon V, Hiam KJ, Boyce H, Hori M, Atallah W, et al. ImmunoGlobe: enabling systems immunology with a manually curated intercellular immune interaction network. BMC Bioinformatics. 2020 Aug 10;21(1):346.

235. Kveler K, Starosvetsky E, Ziv-Kenet A, Kalugny Y, Gorelik Y, Shalev-Malul G, et al. Immune-centric network of cytokines and cells in disease context identified by computational mining of PubMed. Nat Biotechnol. 2018 Jul;36(7):651–9.

236. Gong Q, Sharma M, Kuan EL, Glass MC, Chander A, Singh M, et al. Longitudinal Multi-omic Immune Profiling Reveals Age-Related Immune Cell Dynamics in Healthy Adults [Internet]. bioRxiv; 2024 [cited 2025 May 26]. p. 2024.09.10.612119. Available from: https://www.biorxiv.org/content/10.1101/2024.09.10.612119v1

237. Cui A, Huang T, Li S, Ma A, Pérez JL, Sander C, et al. Dictionary of immune responses to cytokines at single-cell resolution. Nature. 2024 Jan;625(7994):377–84.

238. Bagwell CB, Hill BL, Wood BL, Wallace PK, Alrazzak M, Kelliher AS, et al. Human B-cell and progenitor stages as determined by probability state modeling of multidimensional cytometry data. Cytometry Part B: Clinical Cytometry. 2015;88(4):214–26.

239. Wang Y, Liu J, Burrows PD, Wang JY. B Cell Development and Maturation. In: Wang JY, editor. B Cells in Immunity and Tolerance [Internet]. Singapore: Springer; 2020 [cited 2025 May 26]. p. 1–22. Available from: https://doi.org/10.1007/978-981-15-3532-1_1

240. Melchers F. Checkpoints that control B cell development. J Clin Invest. 2015 Jun 1;125(6):2203–10.

241. Morgan D, Tergaonkar V. Unraveling B cell trajectories at single cell resolution. Trends in Immunology. 2022 Mar 1;43(3):210–29.

242. Zlotoff DA, Bhandoola A. Hematopoietic progenitor migration to the adult thymus. Ann N Y Acad Sci. 2011 Jan;1217:122–38.

243. Adu-Berchie K, Obuseh FO, Mooney DJ. T Cell Development and Function. Rejuvenation Res. 2023 Aug;26(4):126–38.

244. Singh J, Zúñiga-Pflücker JC. Producing proT cells to promote immunotherapies. Int Immunol. 2018 Nov 14;30(12):541–50.

245. Dash SP, Gupta S, Sarangi PP. Monocytes and macrophages: Origin, homing, differentiation, and functionality during inflammation. Heliyon. 2024 Apr 30;10(8):e29686.

246. Teh YC, Chooi MY, Chong SZ. Behind the monocyte's mystique: uncovering their developmental trajectories and fates. Discovery Immunology. 2023 Jan 1;2(1):kyad008.

247. Cherrier DE, Serafini N, Di Santo JP. Innate Lymphoid Cell Development: A T Cell Perspective. Immunity. 2018 Jun 19;48(6):1091–103.

248. Zook EC, Kee BL. Development of innate lymphoid cells. Nat Immunol. 2016 Jul;17(7):775–82.

249. Eberl G, Colonna M, Di Santo JP, McKenzie ANJ. Innate lymphoid cells. Innate lymphoid cells: a new paradigm in immunology. Science. 2015 May 22;348(6237):aaa6566.

250. Puhr S, Lee J, Zvezdova E, Zhou YJ, Liu K. Dendritic cell development-History, advances, and open questions. Semin Immunol. 2015 Dec;27(6):388–96.

251. Collin M, Bigley V. Human dendritic cell subsets: an update. Immunology. 2018 May;154(1):3–20.

252. Jakubzick CV, Randolph GJ, Henson PM. Monocyte differentiation and antigen-presenting functions. Nat Rev Immunol. 2017 Jun;17(6):349–62.