# Separating selection from mutation in antibody language models

Frederick A. Matsen IV[1,2,3,4,*], Will Dumm[1], Kevin Sung[1], Mackenzie M. Johnson[1], David Rich[1], Tyler Starr[5], Yun S. Song[6], Julia Fukuyama[7], and Hugh K. Haddox[1]

[1]Computational Biology Program, Fred Hutchinson Cancer Center, Seattle, WA 98109
[2]Department of Genome Sciences, University of Washington, Seattle, WA 98195
[3]Department of Statistics, University of Washington, Seattle, WA 98195
[4]Howard Hughes Medical Institute, Seattle, WA 98109
[5]Department of Biochemistry, University of Utah, Salt Lake City, UT 84112
[6]Computer Science Division and Department of Statistics, University of California, Berkeley, CA 94720
[7]Department of Statistics, Indiana University, Bloomington, IN 47405
[*]Corresponding author: matsen@fredhutch.org

October 17, 2025

**keywords:** antibody language model, somatic hypermutation, affinity maturation, mutation-selection model, antibody engineering, functional prediction

## Abstract

Antibodies are encoded by nucleotide sequences that are generated by V(D)J recombination and evolve according to mutation and selection processes. Existing antibody language models, however, focus exclusively on antibodies as strings of amino acids and are fitted using standard language modeling objectives such as masked or autoregressive prediction. In this paper, we first show that fitting models using this objective implicitly incorporates nucleotide-level mutation processes as part of the protein language model, which degrades performance when predicting effects of mutations on functional properties of antibodies. To address this limitation, we devise a new framework: a Deep Amino acid Selection Model (DASM) that learns the selection effects of amino-acid mutations while explicitly factoring out the nucleotide-level mutation process. By fitting selection as a separate term from the mutation process, the DASM exclusively quantifies functional effects: effects that change some aspect of the function of the antibody. This factorization leads to substantially improved performance on standard functional benchmarks. Moreover, our model is an order of magnitude smaller and multiple orders of magnitude faster to evaluate than existing approaches, as well as being readily interpretable.

# Introduction

Antibodies are remarkable molecules that can bind essentially any target with high affinity and specificity. They are generated naturally through V(D)J recombination and refined through affinity maturation in germinal centers. Antibodies are also important drugs, and improving binding and other important properties is thus an area of active research. A major goal in the field is to predict the effect of changing one amino acid for another at a given site of an antibody, both for antibody engineering and for understanding naturally occurring antibodies.

Antibody "foundation" language models are trained on large datasets of naturally occurring antibody sequences to assist with this and related problems. These models are trained using the masked objective, in which a site is masked from an antibody amino-acid sequence, and the model is trained to predict the masked amino acid given the remaining sequence. Recent models of this type have hundreds of millions or billions of parameters, and are trained on around a billion sequences [1].

While the masked objective has been very useful for modeling human language [2], this approach may not be ideal for learning functional effects of mutations to antibody sequences. To understand why, it is useful to consider: what could a language model learn in order to succeed under the masked modeling objective? First, it could memorize the germline genes [3,4] and learn about the probabilities of V(D)J recombination. Second, it could learn the codon table, as according to this table some amino-acid mutations are much more likely than others. Third, it could learn rates of somatic hypermutation, because codons containing mutable nucleotide sites are more likely to deviate from germline than those in sites that are less mutable [5]. Finally, after all of these other effects, the model could learn about the impact of amino acid mutations on antibody function, which is the desired signal [6].

In this paper, we first demonstrate that masked language models learn all these factors shaping antibody sequences, despite most being irrelevant for functional prediction. Indeed, we show that conflating mutation and selection processes degrades performance on functional prediction tasks. We then develop a model that explicitly accounts for phylogenetic relationships, the codon table, and somatic hypermutation patterns, allowing it to focus exclusively on functional effects.

Our approach separates mutation and selection processes by encoding functional effects in a Deep Amino acid Selection Model (DASM) while explicitly modeling mutation using a separate fixed model [7]. The DASM, trained on substantially less data, outperforms AbLang2 [3] and general protein language models including ESM2 [8, 9] and ProGen2-small [10]. Unlike existing models, DASMs process complete sequences in a single pass and directly output selection factors for all possible mutations. DASMs are thus orders of magnitude faster than existing models, enabling intensive use on a laptop without a GPU. We provide a paired (heavy and light chain) model with open weights and associated code as part of our `netam` package `https://github.com/matsengrp/netam`.

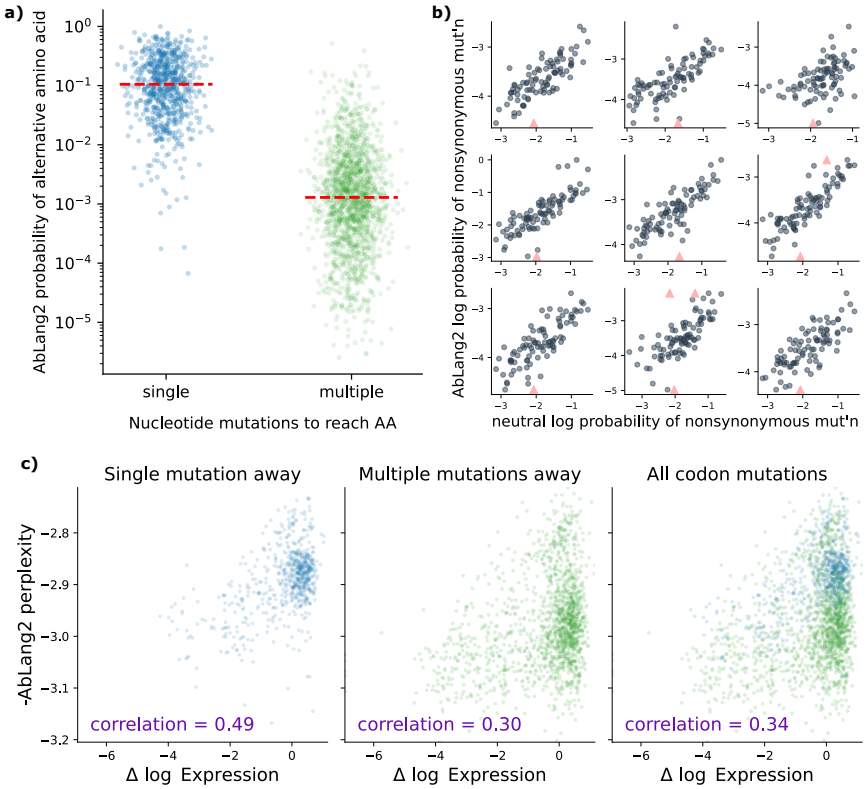


Figure 1: Nucleotide-level mutation processes distort protein language model predictions. **a**: AbLang2 assigns almost 100× lower probabilities to amino acids requiring multiple nucleotide mutations compared to single-mutation variants. Each point represents a possible amino acid substitution at a single site in the amino acid sequence. **b**: AbLang2 probabilities correlate with neutral somatic hypermutation probabilities across the V-encoded portion of nine naive sequences, demonstrating how the model is strongly impacted by mutation bias. Each point represents a site in the sequence. Triangles are outliers that have been brought into the y range. **c**: AbLang2 functional prediction accuracy drops substantially for amino acids that are multiple (2 or 3) nucleotide mutations away from the wildtype codon. Data from [11].

# Results

## Antibody language models are biased by nucleotide-level mutation processes

We first sought to understand the effect of nucleotide-level mutation processes on antibody language model prediction. To do so, we used AbLang2 [3] as a

case study. It has been implemented with more consideration of biology than other models, as it distinguishes between germline-encoded and non-germline-encoded sites in its training. To start, we examined a single naive BCR sequence obtained from a recent study that performed deep sequencing of human antibody repertoires [12]. We iterated over each site in this antibody's protein sequence, masked the site's amino acid, and used AbLang2 to compute the likelihood of each of the 19 alternative amino acids at that site, then normalized to get a probability. Next, using the antibody's nucleotide sequence, we split the alternative amino acids at a given site into two groups: those that can be encoded by a codon a single nucleotide change away from the original codon, and those that require multiple mutations.

We found a striking difference in the AbLang2 amino-acid probabilities (Figure 1a). The median probability for amino acids requiring multiple mutations to a codon was almost two orders of magnitude lower compared to those that only required one. This bias is consistent with the hypothesis that the AbLang2 predictions are influenced by mutation processes unrelated to selection.

The bias can be explained by the masked training procedure. It has already been established that language models trained with the masked objective memorize germline sequences [3, 4]. During somatic hypermutation (SHM), single-nucleotide-per-codon mutations to the germline sequence are much more likely than multi-nucleotide codon mutations. As a result, when predicting relative probabilities of alternative amino acids, language models might be expected to assign much higher probabilities to alternative amino acids that only require a single-nucleotide codon mutation (Figure 1a).

Given this observation, we also hypothesized that AbLang2 probabilities are influenced by differences in the rate of SHM between sites. SHM is a purpose-built enzymatic process that introduces mutations into BCR-coding DNA [13], which occurs independently of the process of natural selection in the germinal center. The rate of SHM varies by an order of magnitude or more between sites, and these biases are well characterized using probabilistic models [7,14]. To test our hypothesis, we examined nine arbitrarily selected sequences from [12]. For each sequence, we used a recent model of SHM [7] to compute per-nucleotide-site rates, and then used the method of [15] to convert these rates into per-codon-site probabilities of nonsynonymous mutations.

We found a clear correlation between these neutral SHM probabilities and probabilities of mutations estimated using AbLang2 (Figure 1b). The Wald test for a nonzero slope reported a p-value below machine precision in each case.

We next sought to understand if these biases distorted predictions on a functional prediction task. To do so, we used the largest dataset [11] of the FLAb [6] collection of benchmarks. Following the protocol in FLAb, we investigated the correlation between a mutation's effect on an experimentally measured phenotype, here expression, and the mutation's plausibility (quantified as pseudo-perplexity) according to a model. Perplexity (as defined in the Methods) is the standard way of evaluating the plausibility of a sequence according to a model. Pseudo-perplexity is a variant of perplexity (Methods) which is the standard means of evaluating perplexity for masked language models [6]. In both defi-

nitions, lower perplexity corresponds to a more plausible sequence, and we use negative perplexity here so that correlations between model predictions (negative perplexity) and observable (expression) should be positive. We split these predictions by number of nucleotide mutations per codon mutation, as before.

We found a significant drop (from 0.49 to 0.30) in predictive performance when going from amino acids that required one mutation to those that required multiple mutations (Figure 1c). Furthermore, when we considered all amino acids together, the correlation remained at 0.34. When we colored these data points by their probability of mutation under an SHM process (Figure S1), we can clearly see that the data points are spread out by the AbLang2 model according to their SHM rate, hindering functional prediction.

In summary, we found that nucleotide-level effects hamper the ability of AbLang2 to predict functional effects of mutations. We then developed an alternative modeling framework that directly factors out nucleotide-level effects.

## Fitting a deep amino acid selection model (DASM)

We implemented a model to learn amino-acid preferences of antibodies without being influenced by germline genes, phylogeny, or SHM biases (Figure 2). To do so, we extended our previous work estimating a single selection value for every site [15] to estimating a value for every alternative amino acid at every site. This extension is analogous to going from a dN/dS type estimate [16, 17] to a sitewise mutation-selection model [18]. Our previous model only operated on antibody heavy chains. This new model operates on heavy chains, light chains, or heavy-light pairs.

As in this previous work, we performed clonal-family clustering, phylogenetic inference, and ancestral sequence reconstruction to generate collections of nucleotide "parent-child pairs" or PCPs (Figure 2a). This resulted in around 2 million PCPs that were used for training (Table S1). Instead of predicting the likelihood of nonsynonymous mutations, we predicted the likelihood of codon mutations using a deep-learning analog of a mutation-selection model. The neutral-probability model was inferred separately using out-of-frame data [7]. We also added a "multihit correction" (see Methods) to this neutral model which accounts for the spatial clustering of SHM [19], resulting in an elevated probability of multiple mutations in a given codon.

The selection component of the model estimates per-site per-amino-acid selection factors given an input amino-acid sequence via a transformer-encoder (Figure 2b). We trained DASMs of several sizes ($\sim$1M, $\sim$4M, $\sim$7M) using joint optimization of branch length $t$ and parameters of the DASM. We found that the $\sim$4M parameter model performed the best according to our objective function. This DASM has 8 attention heads, 32 dimensions per head, a feedforward dimension of 1024, 5 transformer layers, and a dropout probability of 0.1; we will use this version for the rest of the paper.

As an initial comparison between DASM and AbLang2, we used DASM to predict selection factors for the unmutated antibody sequence used in the Koenig benchmark [11]; we found that the selection factors were predictive of functional

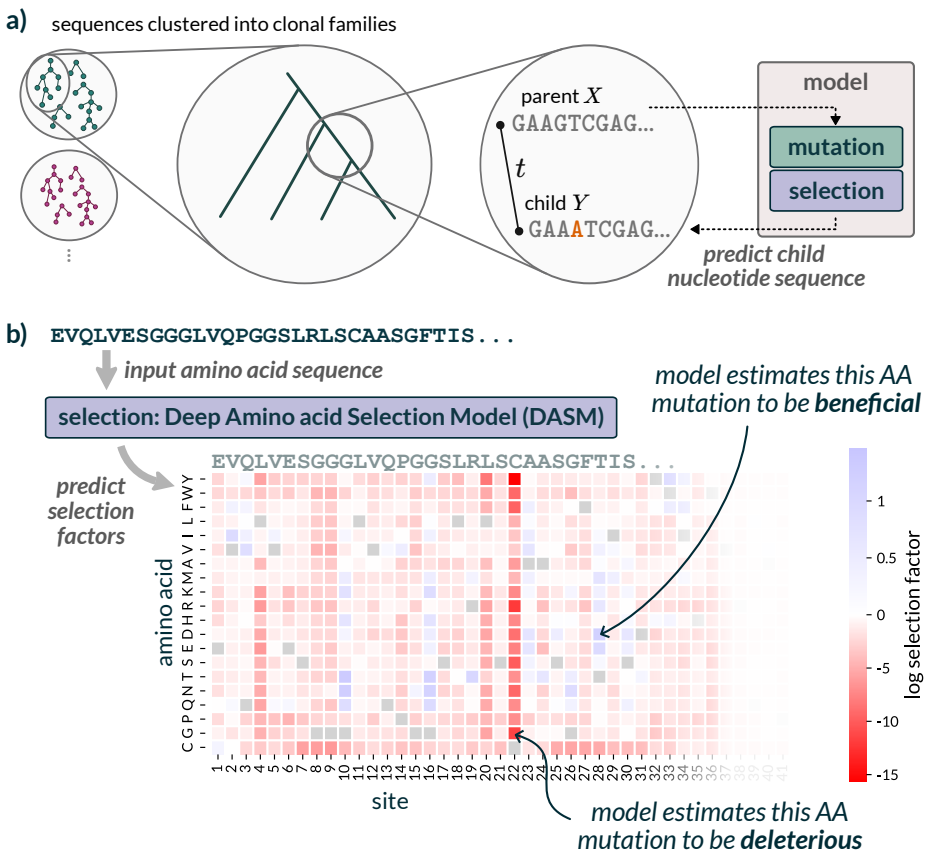


Figure 2: Our model separates mutation from selection to predict functional effects without nucleotide-level biases. **a**: Our model combines a fixed mutation component (trained on non-functional data) with a learned selection component (DASM transformer). Training uses inferred parent-child sequence pairs from reconstructed B cell phylogenies to predict natural affinity maturation after a jointly-inferred time $t$. **b**: The DASM directly predicts selection factors for all amino acid substitutions at every position in a single forward pass. Positive factors indicate beneficial changes, and negative factors indicate deleterious changes.

measurements irrespective of the number of mutations per codon (Figures 3a and S2). We were surprised by the strength of the correlation for amino-acid mutations that require multi-nucleotide codon mutations, given that training signal should be weaker for those rarer mutations. We also found that the selection factors showed very similar distributions for amino-acid mutations requiring single- vs. multi-nucleotide codon mutations (Figure S3). Furthermore, the DASM selection factors captured the alternating pattern of selective con-

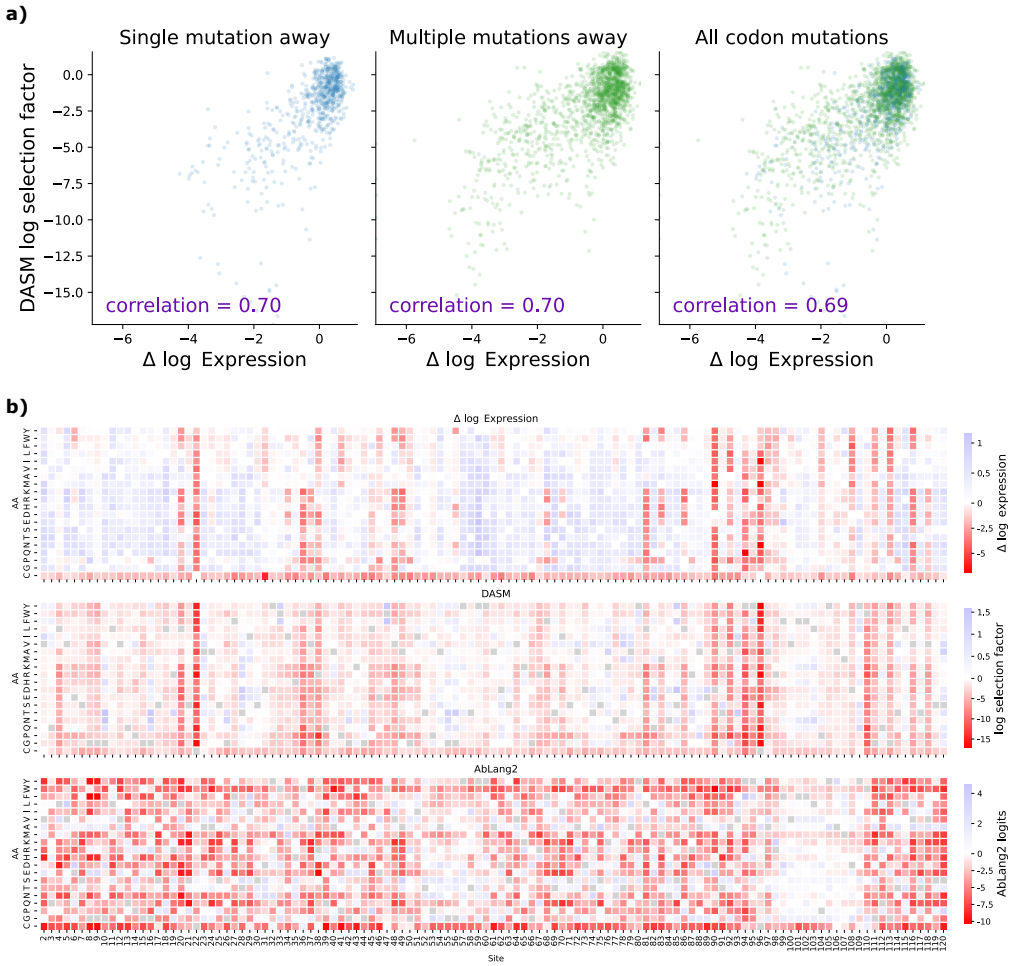


Figure 3: Comparing model predictions with experimentally measured effects of mutations on antibody expression from [11]. **a**: The DASM maintains high predictive accuracy on functional effects of mutations regardless of codon accessibility. The correlation is equally high for amino-acid mutations that only require a single-nucleotide mutation (left plot) vs. amino-acid mutations that require multi-nucleotide mutations (center plot), demonstrating successful separation of mutation bias from functional effects. Compare Figure 1c. **b**: DASM predictions mimic patterns in the expression data. For additional heatmap comparisons see Figure S4.

straint on beta sheets evident in the expression data (red columns in Figures 3b and S4).

## Thrifty+DASM accurately predicts affinity maturation

Next, we evaluated the ability of the DASM to predict the course of natural affinity maturation. To do so, we used PCPs from the `Rodriguez` data (Table S1), which were not used for training the DASM, nor for training AbLang2. For each PCP, we quantified each model's ability to predict both the location of nonsynonymous mutations in the child sequence and the identity of the mutant amino acid. For the DASM, we computed a mutation's probability by multiplying the mutation's selection factor, predicted by the DASM, with its neutral mutation probability, predicted using the same fixed neutral mutation model used to train the DASM [7].

We found that the DASM-based approach was better than AbLang2 at predicting the location of nonsynonymous mutations observed in the PCPs (Figure S5ab). In fact, the overlap between observed and expected is substantially better than what we were able to obtain in a highly controlled mouse experiment [20] with a customized mutation model and a deep mutational scan that quantified how mutations affect binding to the relevant antigen [21].

We also found that the DASM-based approach was better than AbLang2 at predicting the identities of mutant amino acids. To compare the models at this task, we used a notion of "conditional perplexity". Specifically, for each PCP where the parent and child sequences differ on the amino-acid level, we calculated the perplexity of the residues in the child that are different from the parent, conditioning probabilities on there being a substitution (see cond-ppl definition in Methods). We do this to give AbLang2 the best chance of succeeding: although it might have difficulty predicting the occurrence of mutations (Figure S5/conseb) it might still be able to predict the identity of mutations.

Indeed, the DASM had lower conditional perplexity than AbLang2 on 1,000 sequences from the `Rodriguez` [12] dataset (Figure S5c). The median value for DASM was 4.88, compared to 7.39 for AbLang2. In addition to having a lower median, the DASM also has fewer very large values.

## The DASM outperforms masked and autoregressive models at predicting functional effects of mutation

We next performed a more comprehensive characterization of the ability of models to predict mutational effects. We expanded our suite of comparator models to include ESM2 [8, 9], a general protein language model trained with a masked objective, and ProGen2 [10], an autoregressive language model. We used the small general variant of ProGen2, which the FLAb benchmarking paper found to be the best overall model [6], and the 650M variant of ESM2. To predict the plausibility of mutant sequences under these models, we followed current practice [6] and calculated each sequence's pseudo-perplexity for AbLang2 [3] and ESM2 [22], each sequence's perplexity for ProGen2, and then compared the model outputs to experimentally measured mutational effects using Pearson correlation. To predict mutational effects using the DASM, we simply fed the DASM with the unmutated sequence from the DMS experiment and used the

output selection factors as predicted effects. We cannot use perplexity because selection factors alone do not give likelihoods.

To start, we evaluated these models on the two largest datasets from the FLAb collection of benchmarks [6]. The first data set, from Koenig et al. [11], is a deep mutational scanning experiment performed on the Fab of the anti-VEGF antibody G6.31. The authors generated single-site saturated mutagenesis libraries incorporating all possible amino-acid mutations to both the variable heavy and variable light domains. Using phage display, they then performed two independent selection steps: one in which they selected for Fab variants that were stably expressed, and another in which they selected for Fab variants that could bind VEGF. They quantified mutational effects on these properties by computing enrichment ratios of each mutation in selected versus unselected pools, as quantified using deep sequencing.

| | Binding | | Expression | |
|---|---|---|---|---|
| Model | Heavy | Light | Heavy | Light |
| AbLang2 | −0.032 | 0.049 | 0.344 | 0.084 |
| DASM | **0.335** | **0.316** | **0.688** | **0.674** |
| ESM2 | −0.001 | 0.308 | 0.418 | 0.524 |
| ProGen2 | 0.156 | 0.276 | 0.559 | 0.568 |

Table 1: Correlation of models with predicting effects of single mutations on antibody expression and antigen binding, as measured in [11].

We found that the DASM outperformed the other models on both binding and expression (Table 1, Figure S4).

We next turned to the second-biggest dataset of the FLAb benchmark, from Shanehsazzadeh et al. [23]. In this study, the authors took an anti-HER2 antibody, trastuzumab, and used deep generative models to redesign the heavy chain complementarity determining regions (HCDRs). They generated libraries of HCDR variants, including both HCDR3-only designs and HCDR123 designs. Unlike the first dataset described above, many variants had multiple mutations relative to the unmutated antibody sequence. The experiments expressed antibody variants in E. coli, and sorted by FACS based on binding signal. We restricted our attention to sequences with the two most common heavy chain amino acid lengths (119 and 120) as the other lengths did not have enough data to make a full comparison.

We assessed the ability of models to predict these binding measurements. For the language models, we computed the perplexity of each variant, as before. For the DASM, we calculated a consensus sequence for each sequence length, and then used the DASM to compute selection factors for that sequence. Then, we computed a score for each variant by summing the log selection factors for each amino-acid mutation in the variant relative to the consensus sequence. As above, the resulting correlation is substantially higher for the DASM compared to the other models (Table 2).

| Seq. length | AbLang2 | DASM | ESM2 | ProGen2 |
|---|---|---|---|---|
| 119 | 0.263 | **0.458** | 0.248 | 0.074 |
| 120 | 0.166 | **0.518** | 0.337 | 0.052 |

Table 2: Correlation of models with binding measurements on the data of [23], which typically involves multi-mutant variants. See Figure S6 for scatterplots.

For our third validation, we sought to see how our model could predict affinity in the context of large-scale yeast display experiments using MAGMA-seq [24]. We were especially interested in this assay because the measurement is affinity-only rather than a conflation of affinity and expression as in the previous data sets. Because none of these models are trained in an epitope-specific way this will be a challenge for the models. Specifically, MAGMA-seq determines quantitative monovalent binding affinities through deep sequencing of barcoded Fab libraries sorted at multiple antigen concentrations in a manner analogous to Tite-Seq [25]. These data are not present in the FLAb benchmarks. We used the largest datasets in the original MAGMA-seq paper [24] which test variants of influenza antibodies. We also used MAGMA-seq data from [26] which combinatorially applied mutations from development trajectories of mature human antibodies against SARS-CoV-2 spike protein.

| Source | Antibody | AbLang2 | DASM | ESM2 | ProGen2 |
|---|---|---|---|---|---|
| Petersen [24] | 222-1C06 | 0.041 | **0.248** | −0.039 | −0.030 |
| | 319-345 | 0.079 | **0.279** | 0.199 | 0.222 |
| Kirby [26] | 002-S21F2 | −0.246 | **−0.165** | −0.283 | −0.353 |
| | Ab_2-15 | −0.325 | **0.094** | −0.265 | −0.069 |
| | C118 | −0.624 | **0.276** | −0.293 | −0.159 |

Table 3: Correlations between model predictions and binding affinity. The "Petersen" data is from an experiment probing the rules of recognition for influenza neutralizing antibodies [24] and "Kirby" data is from combinatorial libraries applying combinations of mutations on the path from naive to mature SARS-CoV-2 antibodies [26]. See Figure S7 for corresponding scatter plots.

We found that the DASM performed substantially better than other models for this task (Table 3, Figure S7). Nearly all correlation coefficients were small, pointing to the difficulty in predicting effects of mutations on antigen-specific binding affinities. Nevertheless, the DASM was the only model to have a positive correlation for most datasets. The UCA_002-S21F2 lineage proved challenging for all models, with all models having a negative correlation. This may not be surprising given that 002-S21F2 uses a rare VH5-51/VK1-33 gene combination found in only 3 out of 5252 SARS-CoV-2 antibodies in the CoV-AbDab database [27].

## DASM models are orders of magnitude faster than competing models, and much smaller

We next compared the computational requirements for doing these evaluations. We imagined two settings: one where a scientist wishes to evaluate a small collection of sequences on their laptop without a GPU, and the other where they wish to evaluate a larger collection of sequences on a GPU server. Our timing script ran the DASM model, as well as AbLang2 and ESM2 using stepwise masking to obtain amino acid likelihoods. The local machine was a MacBook Pro with a M2 Max chip and 96 GB of memory. The GPU server had an AMD EPYC 75F3 32-core processor with 995 GB of memory and an NVIDIA A100 80GB GPU.

| Model | CPU (s/seq) | GPU (s/seq) |
|---|---|---|
| DASM | 0.0097 | 0.0053 |
| AbLang2 | 11.0529 | 0.5196 |
| ESM2 | 112.5989 | 7.6090 |

Table 4: Computational efficiency comparison on sequences from the MAGMA-seq experiments. 10 sequences were run on CPU, and 100 on the GPU server.

The DASM achieved dramatic computational efficiency gains compared to masked language models (Table 4). On a CPU, the DASM evaluates sequences over 1,000 times faster than AbLang2 and more than 10,000 times faster than ESM2. On a GPU, the DASM is 100 times faster than AbLang2 and over 1,000 times faster than ESM2. This speed advantage comes from the DASM providing predictions for all variants in a single pass, eliminating the need for iterative masking procedures required by masked language models. (We note that it is possible to make predictions from masked language models without masking, although this is not considered best practice because of the influence of the unmasked residue.)

This evaluation was actually generous to the masked language models, as it assumes that we are evaluating likelihoods of masked sites and combining them to evaluate multi-mutant variants. That is in contrast to what was actually done in the performance evaluation, which is direct evaluation of the sequence perplexity for each sequence individually (which should be more accurate). ProGen2 was not included in this comparison because it is categorically slower for this use case: autoregressive models for a panel of variants require running each variant individually through the model.

The DASM benchmarked here is also significantly smaller than alternate models. It has 4M parameters, compared to 45M for AbLang2, 650M for ESM2, and 151M for ProGen2. We suspect that this efficiency is enabled by DASM learning selection effects only, rather than selection and mutation-level effects as for existing models.

In summary, we imagine end-users will find it convenient to download a small (23 MB) weights file and run the DASM on their laptop, as compared to

existing models requiring more powerful hardware.

# Discussion

We have motivated and developed a new direction for protein language modeling. To motivate this approach, we showed how masked language modeling implicitly learned the codon table and somatic hypermutation rates, properties that are orthogonal to antibody function. This incorporation degrades model performance for predicting effects of mutations on antibody expression and binding.

To address this limitation, we introduced a new framework: a Deep Amino Acid Selection Model (DASM), which models selection effects separately from mutation effects. By modeling these components independently, the DASM more accurately predicts the functional consequences of amino-acid mutations while having dramatically reduced computational requirements. The success of our approach highlights the importance of incorporating biological realities into machine-learning models for protein engineering. This aligns with recent calls for deeper dialogue between machine learning and evolutionary biology to address phylogenetic biases in biological foundation models [28].

Thrifty+DASM can be viewed as a neural-network extension of the models rooted in the work of Halpern and Bruno [18]. There, the probability of a codon mutation is expressed as the product of that mutation under a neutral nucleotide process, times a term representing the selection for or against the corresponding amino-acid substitution at that site. Although the original model expressed selection coefficients in a parameter-sparse way using equilibrium frequencies, further model elaborations allowed for per-site-per-amino acid selection factors, as described here, either in a fixed-effects [29, 30] or random-effects [31] framework. Other related work includes developing evolutionary models using deep mutational scanning (DMS) experiments [32,33], and models of sequence-structure compatibility [34]. Thrifty+DASM also extends our previous work that estimates a single value of natural selection at every site [15]; such a model is not comparable to an antibody foundation model because it does not estimate per-amino-acid probabilities.

By separating out selection from mutation and the effect of evolutionary contingency, DASMs provide direct interpretability: the selection factors indicate which amino-acid mutations are beneficial or deleterious at each position. This interpretability could prove valuable for antibody engineering, where understanding which specific mutations would drive improved expression or binding is crucial for rational design. More broadly, this interpretability could help delineate the relationship between antibody sequence, structure, and function, which we plan to explore in future manuscripts. For users wishing to explore the relationship between DASM selection factors and structure for human antibodies in the SAbDAb database [35], we have made interactive 3D visualizations of DASM selection factors using `dms-viz` [36] available at `https://matsen.group/dasm-viz/v1/`.

We will also explore fine-tuning of these models. While DASM provides better-than-state-of-the-art zero-shot performance on binding benchmarks, recent work for binding focuses on using protein embeddings as inputs to subsequent classifiers [37], or fine-tuning for binding prediction [1, 38]. Because the DASM framework separates mutation from functional properties, a fine-tuned DASM may improve on these previous efforts.

We are beginning the process of extending DASMs to other settings. Although antibodies form an interesting first application of DASMs, and have many properties that make DASMs a useful tool, DASMs are not restricted to the antibody case. Viral sequences also exist in abundance and may be amenable to the DASM approach. Rather than many clonal families for antibodies, we will fit a DASM with many separate alignments for related but distinct evolutionary histories.

Taking this idea to its logical extent, we would like to train a DASM model analogous to ESM [22] for all proteins. However, this will be a major computational undertaking. Even setting training such a model aside, doing phylogeny and ancestral sequence reconstruction on all protein alignments will require careful planning. If we followed the steps used here for phylogeny and ancestral sequence reconstruction on the 2.6 million sequence alignments used for the MSA Transformer [39] it would take around 3,000 CPU-years.

DASMs represent a new paradigm for deep models of protein function that leverage evolutionary information directly. Even in this initial implementation, DASMs outperform existing foundation models while requiring orders of magnitude less computation, parameters, and training data. This efficiency suggests that evolutionary structure provides a powerful approach for understanding protein function.

# Methods

## Data

BCR sequence data was processed with `partis` [40] to cluster into clonal families and infer germlines. Inferred insertions or deletions were reversed, so that all sequences align to the naive sequence without gaps. We selected clonal families with at least two productive sequences; a sequence is considered productive if the canonical cysteine and tryptophan codons that flank the CDR3 are in the same frame as the start of the V segment (although they can be mutated), and there are no stop codons. We excluded sequences with stop codons. Following the training of other LLMs (e.g. [3]) we also excluded sequences with mutated conserved "signature" cysteines, in contrast to our previous work [15].

As in our previous work, tree inference and ancestral sequence reconstruction were performed with the K80 substitution model using the naive sequence as outgroup, allowing mutation rate heterogeneity across sites using a 4-category FreeRate model using IQ-Tree [41]. However, for paired data we used the *edge-linked-proportional* partition model in IQ-Tree that allowed the heavy and light

chains to evolve at overall different rates [42].

Once this was done, we had a set of parent-child pairs (PCPs) that correspond to the pairs of parent and child sequences on the edges of the phylogenetic tree (Figure 2a). We used these PCPs to train the model.

We denote pairs of parent and child sequences as $(X, Y)$, where $X$ is the parent sequence and $Y$ is the child sequence. We use $\bar{X}$ to denote the amino acid sequence corresponding to $X$.

# Model

## Formulation and loss

Assume we are given a parent codon sequence $X$ and a child sequence $Y$. We will use $\bar{c}$ to denote the amino acid sequence of codon $c$, and use $\bar{Z}$ to denote the amino acid sequence of codon sequence $Z$. We will use $j$ to denote codon sites. As before [15], we model the neutral probability of a mutation to codon $c$ as the product of per-site mutation probabilities, resulting in the probability $p_{j,c}(t, X)$ of a mutation to codon $c$ at site $j$ after time $t$.

The selection term $f_{j,\bar{c}}(\bar{X})$ is a "selection factor" that quantifies the natural selection happening at site $j$ if it was to be changed to $\bar{c}$. If this value is greater than 1, it predicts that the mutation would be beneficial in the course of affinity maturation, while if it is less than 1, it predicts that the mutation would be deleterious. We parameterize $f$ using a shared amino-acid embedding followed by a transformer-encoder [43] neural network followed by a simple linear layer.

We will now define the likelihood $\ell_{j,c}(t, X)$ of codon $c$ at site $j$ given time $t$ and parent sequence $X$. When $c$ is a non-wildtype codon it is

$$\ell_{j,c}(t, X) := \begin{cases} p_{j,c}(t, X) f_{j,\bar{c}}(\bar{X}) & \text{if } c \text{ codes for an amino acid} \\ 0 & \text{if } c \text{ codes for stop.} \end{cases} \tag{1}$$

where $f_{j,a}(\bar{X}) := 1$ when $a$ is the amino acid for the wildtype codon. We then take the likelihood of the wildtype codon to be 1 minus the sum of these non-WT codons.

The overall likelihood for a parent-child pair is then

$$\prod_j \ell_{j,y_j}(t, X)$$

where $y_j$ is the $j$th codon of $Y$. We optimize this likelihood jointly over the branch lengths $t$ and parameters of the selection model $f$. Because $\ell_{j,c}(t, X) = p_{j,c}(t, X)$ when $c$ codes for the wildtype amino acid, this approach effectively fixes the selection factors for neutral substitutions to be 1 for the purposes of branch length optimization. This is useful because it gives us a "gauge" that eliminates a potential unidentifiability in the model.

Especially early in training it can happen that $p_{j,c}(t, X) f_{j,\bar{c}}(\bar{X})$ is greater than one, or that the sum of such terms is greater than one. In these cases, the sum is clamped to be a little less than 1.

### Perplexity

Perplexity is a common metric for evaluating autoregressive language models such as ProGen2 [10]: it is the geometric mean of the inverse probabilities of the model generating each observed token given its context. These inverse probabilities can be interpreted as the effective number of tokens that the model considers plausible at each position. This is equivalent to the exponential of the average negative log probability of each token given its preceding context:

$$\text{ppl}(x) = \exp\left(-\frac{1}{n}\sum_{i=1}^{n}\log p(x_i \mid x_{<i})\right),$$

where $x = (x_1, x_2, \ldots, x_n)$ is a sequence of $n$ tokens, and $x_{<i}$ represents all tokens preceding position $i$.

For encoder-only models like ESM2 and AbLang2, which mask tokens rather than using autoregressive prediction, we follow the authors of these models and instead calculate a "pseudo-perplexity" using the probability of each token given all other tokens in the sequence:

$$\text{pseudo-ppl}(x) = \exp\left(-\frac{1}{n}\sum_{i=1}^{n}\log p\big(x_i \mid x_{\setminus i}\big)\right),$$

where $x_{\setminus i}$ denotes the sequence with token $i$ masked out.

We also use a notion of "conditional" perplexity: for each parent-child pair $(x, y)$ of sequences that differ on the amino acid level, calculate the perplexity of the residues that are different from the parent, conditioning probabilities on there being an amino acid substitution. That is, if we let $S_{x,y}$ be the sites that differ between $x$ and $y$, then

$$\text{cond-ppl}(x, y) = \exp\left(-\frac{1}{|S_{x,y}|}\sum_{i \in S_{x,y}}\log p(\bar{y}_i \mid x;\ \bar{y}_i \neq \bar{x}_i)\right)$$

where bar represents amino acid sequence as before.

## Multihit correction

In the SHM process, having a mutation at a site increases the probability of mutations at nearby sites [19]. Although a general solution to this problem can easily lead down the path of intractability, we wanted to incorporate this phenomenon into our work.

Inspired by the work of [44], we added several simple "multihit" rate multipliers to our neutral model. These multipliers account for the varying probabilities of mutations based on the number of nucleotide changes required.

We found that, on neutrally-evolving out-of-frame data, such a correction substantially improved model fit (Figure S9). Before correction, the model systematically underestimates multi-hit mutations. After training the multihit

model with correction factors for 1, 2, and 3 mutations per codon, the observed-expected agreement improved substantially for these classes.

This multihit correction was trained on the same separate out of frame data as used in training our neutral model [7] and was incorporated into the neutral mutation probabilities for DASM training.

## Heavy and light rates

Light chains mutate at a lower rate than heavy chains during affinity maturation. To quantify this relative rate, we used IQ-Tree's rate partition feature during phylogenetic inference on paired heavy-light chain data (see Data section). This allowed us to estimate separate substitution rates for heavy and light chains within each clonal family. We found the median relative rate of light to heavy chains across all clonal families to be 0.63, which we used as our fixed light chain rate adjustment parameter. This estimate is broadly concordant with previous observations that light chains evolve at roughly half the rate of heavy chains [45].

This relative rate is incorporated into the likelihood calculation by scaling the neutral mutation rates for light chain sequences by the light chain rate adjustment factor before combining them with selection factors. Specifically, when evaluating paired heavy-light chain sequences, the per-site neutral mutation rates $p_{j,c}(t, X)$ for light chain sites are multiplied by this adjustment factor, while heavy chain rates remain unscaled.

## Model implementation, training, and evaluation

This model was implemented in PyTorch 2.5 [46]. Models were trained using the RMSprop optimizer, with 4 cycles, each consisting of branch length optimization then neural network optimization.

We used the following software: Altair [47,48], BioPython [49], Matplotlib [50], pandas [51], pytest [52], Seaborn [53], and Snakemake [54].

### LLM model score computation

ESM2 scores were calculated using the FLAb paper methodology, separately evaluating heavy and light chains then averaging the pseudo-perplexities, while AbLang2 directly evaluated paired heavy-light sequences. ProGen2 scores were computed following the same FLAb methodology of separately evaluating heavy and light chains then averaging perplexities.

### Koenig evaluation

We obtained the Koenig et al. deep mutational scanning data from the FLAb benchmark repository [6] (commit 67738ee, April 17, 2024). For DASM evaluation, we used the wild-type G6.31 heavy and light chain sequences as input to predict selection factors for each possible amino acid substitution at each site.

The model's log selection factors were compared against the experimental log enrichment ratios via Pearson correlation. We evaluated performance separately for heavy and light chains across both binding and expression datasets.

### Shanehsazzadeh evaluation

We obtained the Shanehsazzadeh et al. binding affinity data from the FLAb benchmark repository [6] (commit 67738ee, April 17, 2024). We subset to the "zero-shot" component of the dataset. For DASM evaluation, we partitioned the data by heavy chain amino acid length (119 and 120 residues) due to the diverse nature of the designed sequences. For each length group, we computed a site-by-site consensus sequence to serve as the reference sequence for model predictions. The aggregate log selection factor for each variant was calculated as the sum of log selection factors for each position that differed from the consensus sequence. Model performance was evaluated through Pearson correlation analysis between the aggregate log selection factors and experimental binding affinities.

### MAGMA-seq evaluation

We combined data from two complementary studies using the MAGMA-seq methodology: Petersen et al. [24] providing CDR-targeted mutagenesis data around mature influenza antibodies and Kirby et al. [26] providing UCA to mature antibody evolution trajectories for SARS-CoV-2 antibodies. All sequence variants were assigned to their corresponding antibody systems using reference matching with sequence similarity thresholds. To ensure data quality, experimental replicates were aggregated using geometric mean in $\log_{10}$ space, and high-variance measurements (coefficient of variation $> 0.5$) were filtered out as in the original papers. After deduplication and quality filtering, the unified dataset contained 1,128 sequences across 6 antibody systems: 4 Kirby UCAs and 2 Petersen mature antibodies. The 1_20 antibody system was dropped from the Kirby data as it only had 7 assigned sequences. Reference sequences were used as inputs for DASM score calculations. Model performance was evaluated using Pearson correlation with binding affinity ($-\log_{10} K_D$).

## Data availability

Models and inference code can be found at `https://github.com/matsengrp/netam`, including a simple means of accessing the pretrained model demonstrated in the `notebooks/dasm_demo.ipynb` notebook. Our reproducible experiments are available at `https://github.com/matsengrp/dasm-experiments`. Relevant preprocessed data has been uploaded to Zenodo at `https://doi.org/10.5281/zenodo.17322891`.

## Reproducing figures

The following figures were made in notebooks, which can be found in the `notebooks/dasm_paper` directory of the experiments repository.

- Figure 1 is made in `nt_process_in_llms.ipynb`.
- Table 1 and Figures 3, S1, S2, S3, and S4 are made in `koenig.ipynb`.
- Figure S5 is made by the Snakemake pipeline and in `perplexity.ipynb`.
- Table 2 and Figure S6 are made in `shanehsazzadeh.ipynb`.
- Table S1 is made in `data_summaries.ipynb`.

Figure S9 is made in `multihit_model_exploration.ipynb` in the `https://github.com/matsengrp/thrifty-experiments-1` repository.

Our repository file `data/whitehead/MAGMA_PIPELINE_STRUCTURE.md` describes how Figure S7 and Table 3 are made.

Table 4 is made by `timing_direct_gpu.py` and `make_timing_table.py`.

# Competing interests

The authors have no competing interests to disclose.

# Acknowledgements

We are grateful to the authors of [55] and the lab of Corey Watson for sharing preprocessed data. We thank Antoine Koehl for helpful discussions, as well as Michael Chungyon and Timothy Whitehead for help with data processing. This work supported by NIH grants R01-AI146028 (PI Matsen), R56-HG013117 and R01-HG013117 (PI Song), DP2-AI177890 (PI Starr), and Searle Scholars Program (PI Starr).

Scientific Computing Infrastructure at Fred Hutch funded by ORIP grant S10OD028685. Frederick Matsen is an investigator of the Howard Hughes Medical Institute.

ChatGPT and Claude AI models were used to draft code and text for this manuscript.

# References

[1] Justin Barton, Aretas Gaspariunas, David A Yadin, Jorge Dias, Francesca L Nice, Danielle H Minns, Olivia Snudden, Chelsea Povall, Sara Valle Tomas, Harry Dobson, James H R Farmery, Jinwoo Leem, and Jacob D Galson. A generative foundation model for antibody sequence understanding. *bioRxiv*, page 2024.05.22.594943, May 2024. URL: `https:`

//www.biorxiv.org/content/10.1101/2024.05.22.594943v1, doi:10.1101/2024.05.22.594943.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv [cs.CL]*, October 2018. URL: http://arxiv.org/abs/1810.04805, arXiv:1810.04805.

[3] Tobias H Olsen, Iain H Moal, and Charlotte M Deane. Addressing the antibody germline bias and its effect on language models for improved antibody design. *Bioinformatics*, page btae618, October 2024. URL: https://academic.oup.com/bioinformatics/advance-article/doi/10.1093/bioinformatics/btae618/7845256, doi:10.1093/bioinformatics/btae618.

[4] Karenna Ng and Bryan Briney. Focused learning by antibody language models using preferential masking of non-templated regions. *Patterns (N. Y.)*, 6(6):101239, June 2025. URL: http://dx.doi.org/10.1016/j.patter.2025.101239, doi:10.1016/j.patter.2025.101239.

[5] Zizhang Sheng, Chaim A Schramm, Rui Kong, NISC Comparative Sequencing Program, James C Mullikin, John R Mascola, Peter D Kwong, and Lawrence Shapiro. Gene-Specific substitution profiles describe the types and frequencies of amino acid changes during antibody somatic hypermutation. *Front. Immunol.*, 8:537, May 2017. URL: http://dx.doi.org/10.3389/fimmu.2017.00537, doi:10.3389/fimmu.2017.00537.

[6] Michael Chungyoun, Jeffrey Ruffolo, and Jeffrey Gray. FLAb: Benchmarking deep learning methods for antibody fitness prediction. *bioRxiv*, page 2024.01.13.575504, January 2024. URL: https://www.biorxiv.org/content/10.1101/2024.01.13.575504v1.abstract, doi:10.1101/2024.01.13.575504.

[7] Kevin Sung, Mackenzie M Johnson, Will Dumm, Noah Simon, Hugh Haddox, Julia Fukuyama, and Frederick A Matsen IV. Thrifty wide-context models of B cell receptor somatic hypermutation. *eLife*, March 2025. URL: https://elifesciences.org/reviewed-preprints/105471, doi:10.7554/elife.105471.1.

[8] Roshan Rao, Joshua Meier, Tom Sercu, Sergey Ovchinnikov, and Alexander Rives. Transformer protein language models are unsupervised structure learners. In *International Conference on Learning Representations*, October 2020. URL: https://openreview.net/pdf?id=fylclEqgvgd.

[9] Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C Lawrence Zitnick, Jerry Ma, and Rob Fergus. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl. Acad. Sci. U. S. A.*,

118(15):e2016239118, April 2021. URL: https://www.pnas.org/doi/abs/10.1073/pnas.2016239118, doi:10.1073/pnas.2016239118.

[10] Erik Nijkamp, Jeffrey Ruffolo, Eli N Weinstein, Nikhil Naik, and Ali Madani. ProGen2: Exploring the boundaries of protein language models. *arXiv [cs.LG]*, June 2022. URL: http://arxiv.org/abs/2206.13517, arXiv:2206.13517.

[11] Patrick Koenig, Chingwei V Lee, Benjamin T Walters, Vasantharajan Janakiraman, Jeremy Stinson, Thomas W Patapoff, and Germaine Fuh. Mutational landscape of antibody variable domains reveals a switch modulating the interdomain conformational dynamics and antigen binding. *Proc. Natl. Acad. Sci. U. S. A.*, January 2017. URL: http://dx.doi.org/10.1073/pnas.1613231114, doi:10.1073/pnas.1613231114.

[12] Oscar L Rodriguez, Yana Safonova, Catherine A Silver, Kaitlyn Shields, William S Gibson, Justin T Kos, David Tieri, Hanzhong Ke, Katherine J L Jackson, Scott D Boyd, Melissa L Smith, Wayne A Marasco, and Corey T Watson. Genetic variation in the immunoglobulin heavy chain locus shapes the human antibody repertoire. *Nature Comm.*, 14(1):4419, July 2023. URL: http://dx.doi.org/10.1038/s41467-023-40070-x, doi:10.1038/s41467-023-40070-x.

[13] Bas Pilzecker and Heinz Jacobs. Mutating for good: DNA damage responses during somatic hypermutation. *Front. Immunol.*, 10:438, March 2019. URL: http://dx.doi.org/10.3389/fimmu.2019.00438, doi:10.3389/fimmu.2019.00438.

[14] Gur Yaari, Jason A Vander Heiden, Mohamed Uduman, Daniel Gadala-Maria, Namita Gupta, Joel N H Stern, Kevin C O'Connor, David A Hafler, Uri Laserson, Francois Vigneault, and Steven H Kleinstein. Models of somatic hypermutation targeting and substitution based on synonymous mutations from high-throughput immunoglobulin sequencing data. *Front. Immunol.*, 4:358, November 2013. URL: http://dx.doi.org/10.3389/fimmu.2013.00358, doi:10.3389/fimmu.2013.00358.

[15] Frederick A Matsen IV, Kevin Sung, Mackenzie M Johnson, Will Dumm, David Rich, Tyler Starr, Yun S Song, Philip Bradley, Julia Fukuyama, and Hugh K Haddox. A sitewise model of natural selection on individual antibodies via a transformer-encoder. *Mol. Biol. Evol.*, page msaf186, August 2025. URL: https://dx.doi.org/10.1093/molbev/msaf186, doi:10.1093/molbev/msaf186.

[16] N Goldman and Z Yang. A codon-based model of nucleotide substitution for protein-coding DNA sequences. *Mol. Biol. Evol.*, 11(5):725–736, September 1994. URL: http://www.ncbi.nlm.nih.gov/pubmed/7968486.

[17] S V Muse and B S Gaut. A likelihood approach for comparing synonymous and nonsynonymous nucleotide substitution rates, with application to the

chloroplast genome. *Mol. Biol. Evol.*, 11(5):715–724, September 1994. URL: http://www.ncbi.nlm.nih.gov/pubmed/7968485.

[18] A L Halpern and W J Bruno. Evolutionary distances for protein-coding sequences: modeling site-specific residue frequencies. *Mol. Biol. Evol.*, 15(7):910–917, July 1998. URL: http://www.ncbi.nlm.nih.gov/pubmed/9656490.

[19] Natanael Spisak, Aleksandra M Walczak, and Thierry Mora. Learning the heterogeneous hypermutation landscape of immunoglobulins from high-throughput repertoire data. *Nucleic Acids Res.*, 48(19):10702–10712, November 2020. URL: http://dx.doi.org/10.1093/nar/gkaa825, doi:10.1093/nar/gkaa825.

[20] William S DeWitt, Ashni A Vora, Tatsuya Araki, Jared G Galloway, Tanwee Alkutkar, Juliana Bortolatto, Tiago B R Castro, Will Dumm, Chris Jennings-Shaffer, Tongqiu Jia, Luka Mesin, Gabriel Ozorowski, Juhee Pae, Duncan K Ralph, Jesse D Bloom, Armita Nourmohammad, Yun S Song, Andrew B Ward, Tyler N Starr, Frederick A Matsen, IV, and Gabriel D Victora. Replaying germinal center evolution on a quantified affinity landscape. *bioRxiv*, page 2025.06.02.656870, June 2025. URL: https://www.biorxiv.org/content/10.1101/2025.06.02.656870v1.abstract, doi:10.1101/2025.06.02.656870.

[21] Mackenzie M Johnson, Kevin Sung, Hugh K Haddox, Ashni A Vora, Tatsuya Araki, Gabriel D Victora, Yun S Song, Julia Fukuyama, and Frederick A Matsen, IV. Nucleotide context models outperform protein language models for predicting antibody affinity maturation. *bioRxiv*, page 2025.06.16.659977, June 2025. URL: https://www.biorxiv.org/content/10.1101/2025.06.16.659977v1.abstract, doi:10.1101/2025.06.16.659977.

[22] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan Dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, March 2023. URL: http://dx.doi.org/10.1126/science.ade2574, doi:10.1126/science.ade2574.

[23] Amir Shanehsazzadeh, Sharrol Bachas, Matt McPartlon, George Kasun, John M Sutton, Andrea K Steiger, Richard Shuai, Christa Kohnert, Goran Rakocevic, Jahir M Gutierrez, Chelsea Chung, Breanna K Luton, Nicolas Diaz, Simon Levine, Julian Alverio, Bailey Knight, Macey Radach, Alex Morehead, Katherine Bateman, David A Spencer, Zachary McDargh, Jovan Cejovic, Gaelin Kopec-Belliveau, Robel Haile, Edriss Yassine, Cailen McCloskey, Monica Natividad, Dalton Chapman, Joshua

Bennett, Jubair Hossain, Abigail B Ventura, Gustavo M Canales, Muttappa Gowda, Kerianne A Jackson, Jennifer T Stanton, Marcin Ura, Luka Stojanovic, Engin Yapici, Katherine Moran, Rodante Caguiat, Amber Brown, Shaheed Abdulhaqq, Zheyuan Guo, Lillian R Klug, Miles Gander, and Joshua Meier. Unlocking de novo antibody design with generative artificial intelligence. *bioRxiv*, page 2023.01.08.523187, January 2023. URL: https://www.biorxiv.org/content/10.1101/2023.01.08.523187v4, doi:10.1101/2023.01.08.523187.

[24] Brian M Petersen, Monica B Kirby, Karson M Chrispens, Olivia M Irvin, Isabell K Strawn, Cyrus M Haas, Alexis M Walker, Zachary T Baumer, Sophia A Ulmer, Edgardo Ayala, Emily R Rhodes, Jenna J Guthmiller, Paul J Steiner, and Timothy A Whitehead. An integrated technology for quantitative wide mutational scanning of human antibody fab libraries. *Nat. Commun.*, 15(1):3974, May 2024. URL: https://www.nature.com/articles/s41467-024-48072-z, doi:10.1038/s41467-024-48072-z.

[25] Rhys M Adams, Thierry Mora, Aleksandra M Walczak, and Justin B Kinney. Measuring the sequence-affinity landscape of antibodies with massively parallel titration curves. *Elife*, 5, December 2016. URL: http://dx.doi.org/10.7554/eLife.23156, doi:10.7554/eLife.23156.

[26] Monica B Kirby, Brian M Petersen, Jonathan G Faris, Siobhan P Kells, Kayla G Sprenger, and Timothy A Whitehead. Retrospective SARS-CoV-2 human antibody development trajectories are largely sparse and permissive. *Proc. Natl. Acad. Sci. U. S. A.*, 122(4):e2412787122, January 2025. URL: https://www.pnas.org/doi/abs/10.1073/pnas.2412787122, doi:10.1073/pnas.2412787122.

[27] Sanjeev Kumar, Anamika Patel, Lilin Lai, Chennareddy Chakravarthy, Rajesh Valanparambil, Elluri Seetharami Reddy, Kamalvishnu Gottimukkala, Meredith E Davis-Gardner, Venkata Viswanadh Edara, Susanne Linderman, Kaustuv Nayak, Kritika Dixit, Pragati Sharma, Prashant Bajpai, Vanshika Singh, Filipp Frank, Narayanaiah Cheedarla, Hans P Verkerke, Andrew S Neish, John D Roback, Grace Mantus, Pawan Kumar Goel, Manju Rahi, Carl W Davis, Jens Wrammert, Sucheta Godbole, Amy R Henry, Daniel C Douek, Mehul S Suthar, Rafi Ahmed, Eric Ortlund, Amit Sharma, Kaja Murali-Krishna, and Anmol Chandele. Structural insights for neutralization of omicron variants BA.1, BA.2, BA.4, and BA.5 by a broadly neutralizing SARS-CoV-2 antibody. *Sci. Adv.*, 8(40):eadd2032, October 2022. URL: http://dx.doi.org/10.1126/sciadv.add2032, doi:10.1126/sciadv.add2032.

[28] Ryan York and Prachee Avasthi. Phylogenies and biological foundation models. *Arcadia Science*, jun 17 2025. URL: https://research.arcadiascience.com/pub/idea-phylogenies-bfms/release/1, doi:10.57844/arcadia-znum-bm22.

[29] Asif U. Tamuri, Mario dos Reis, and Richard A. Goldstein. Estimating the distribution of selection coefficients from phylogenetic data using sitewise mutation-selection models. *Genetics*, 190(3):1101–1115, March 2012. URL: http://dx.doi.org/10.1534/genetics.111.136432, doi:10.1534/genetics.111.136432.

[30] Asif U Tamuri, Nick Goldman, and Mario dos Reis. A penalized-likelihood method to estimate the distribution of selection coefficients from phylogenetic data. *Genetics*, 197(1):257–271, May 2014. URL: http://dx.doi.org/10.1534/genetics.114.162263, doi:10.1534/genetics.114.162263.

[31] Nicolas Rodrigue and Nicolas Lartillot. Site-heterogeneous mutation-selection models within the PhyloBayes-MPI package. *Bioinformatics*, 30(7):1020–1021, April 2014. URL: http://dx.doi.org/10.1093/bioinformatics/btt729, doi:10.1093/bioinformatics/btt729.

[32] Jesse D Bloom. Identification of positive selection in genes is greatly improved by using experimentally informed site-specific models. *Biol. Direct*, 12(1):1, January 2017. URL: http://dx.doi.org/10.1186/s13062-016-0172-z, doi:10.1186/s13062-016-0172-z.

[33] Sarah K Hilton, Michael B Doud, and Jesse D Bloom. phydms: software for phylogenetic analyses informed by deep mutational scanning. *PeerJ*, 5:e3657, July 2017. URL: http://dx.doi.org/10.7717/peerj.3657, doi:10.7717/peerj.3657.

[34] Douglas M Robinson, David T Jones, Hirohisa Kishino, Nick Goldman, and Jeffrey L Thorne. Protein evolution with dependence among codons due to tertiary structure. *Mol. Biol. Evol.*, 20(10):1692–1704, July 2003. URL: http://dx.doi.org/10.1093/molbev/msg184, doi:10.1093/molbev/msg184.

[35] James Dunbar, Konrad Krawczyk, Jinwoo Leem, Terry Baker, Angelika Fuchs, Guy Georges, Jiye Shi, and Charlotte M Deane. SAbDab: the structural antibody database. *Nucleic Acids Res.*, 42(Database issue):D1140–6, January 2014. URL: http://dx.doi.org/10.1093/nar/gkt1043, doi:10.1093/nar/gkt1043.

[36] William W Hannon and Jesse D Bloom. dms-viz: Structure-informed visualizations for deep mutational scanning and other mutation-based datasets. *J. Open Source Softw.*, 9(99):6129, July 2024. URL: https://joss.theoj.org/papers/10.21105/joss.06129, doi:10.21105/joss.06129.

[37] Meng Wang, Jonathan Patsenker, Henry Li, Yuval Kluger, and Steven H Kleinstein. Language model-based B cell receptor sequence embeddings can effectively encode receptor specificity. *Nucleic Acids Res.*, 52(2):548–557, January 2024. URL: http://dx.doi.org/10.1093/nar/gkad1128, doi:10.1093/nar/gkad1128.

[38] Meng Wang, Jonathan Patsenker, Henry Li, Yuval Kluger, and Steven H Kleinstein. Supervised fine-tuning of pre-trained antibody language models improves antigen specificity prediction. *PLoS Comput. Biol.*, 21(3):e1012153, March 2025. URL: http://dx.doi.org/10.1371/journal.pcbi.1012153, doi:10.1371/journal.pcbi.1012153.

[39] Roshan Rao, Jason Liu, Robert Verkuil, Joshua Meier, John F Canny, Pieter Abbeel, Tom Sercu, and Alexander Rives. MSA transformer. *bioRxiv*, page 2021.02.12.430858, July 2021. URL: https://www.biorxiv.org/content/10.1101/2021.02.12.430858v2, doi:10.1101/2021.02.12.430858.

[40] Duncan K Ralph and Frederick A Matsen, 4th. Inference of B cell clonal families using heavy/light chain pairing information. *PLoS Comput. Biol.*, 18(11):e1010723, November 2022.

[41] Bui Quang Minh, Heiko A Schmidt, Olga Chernomor, Dominik Schrempf, Michael D Woodhams, Arndt von Haeseler, and Robert Lanfear. Iq-tree 2: New models and efficient methods for phylogenetic inference in the genomic era. *Mol. Biol. Evol.*, 37(5):1530–1534, February 2020. doi:10.1093/molbev/msaa015.

[42] Olga Chernomor, Arndt von Haeseler, and Bui Quang Minh. Terrace aware data structure for phylogenomic inference from supermatrices. *Syst. Biol.*, 65(6):997–1008, November 2016. URL: https://academic.oup.com/sysbio/article/65/6/997/2281634, doi:10.1093/sysbio/syw037.

[43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł Ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I Guyon, U V Luxburg, S Bengio, H Wallach, R Fergus, S Vishwanathan, and R Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 6000–6010. Curran Associates, Inc., 2017. URL: http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

[44] Alexander G Lucaci, Sadie R Wisotsky, Stephen D Shank, Steven Weaver, and Sergei L Kosakovsky Pond. Extra base hits: Widespread empirical support for instantaneous multiple-nucleotide changes. *PLoS One*, 16(3):e0248337, March 2021. URL: http://dx.doi.org/10.1371/journal.pone.0248337, doi:10.1371/journal.pone.0248337.

[45] Cole G Jensen, Jacob A Sumner, Steven H Kleinstein, and Kenneth B Hoehn. Inferring B cell phylogenies from paired H and L chain BCR sequences with dowser. *J. Immunol.*, 212(10):1579–1588, 15 May 2024. URL: http://dx.doi.org/10.4049/jimmunol.2300851, doi:10.4049/jimmunol.2300851.

[46] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein,

Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. *arXiv*, December 2019. URL: https://papers.nips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf, arXiv:1912.01703.

[47] Arvind Satyanarayan, Dominik Moritz, Kanit Wongsuphasawat, and Jeffrey Heer. Vega-lite: A grammar of interactive graphics. *IEEE transactions on visualization and computer graphics*, 23(1):341–350, 2017.

[48] Jacob VanderPlas, Brian Granger, Jeffrey Heer, Dominik Moritz, Kanit Wongsuphasawat, Arvind Satyanarayan, Eitan Lees, Ilia Timofeev, Ben Welsh, and Scott Sievert. Altair: Interactive statistical visualizations for python. *Journal of Open Source Software*, 3(32):1057, 2018. doi:10.21105/joss.01057.

[49] Peter J. A. Cock, Tiago Antao, Jeffrey T. Chang, Brad A. Chapman, Cymon J. Cox, Andrew Dalke, Iddo Friedberg, Thomas Hamelryck, Frank Kauff, Bartek Wilczynski, and Michiel J. L. de Hoon. Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics*, 25(11):1422–1423, 03 2009. arXiv:https://academic.oup.com/bioinformatics/article-pdf/25/11/1422/48989335/bioinformatics_25_11_1422.pdf, doi:10.1093/bioinformatics/btp163.

[50] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. doi:10.1109/MCSE.2007.55.

[51] Wes McKinney. Data Structures for Statistical Computing in Python. In Stéfan van der Walt and Jarrod Millman, editors, *Proceedings of the 9th Python in Science Conference*, pages 56 – 61, 2010. doi:10.25080/Majora-92bf1922-00a.

[52] Holger Krekel, Bruno Oliveira, Ronny Pfannschmidt, Floris Bruynooghe, Brianna Laugher, and Florian Bruhin. pytest 8.1.1, 2004. URL: https://github.com/pytest-dev/pytest.

[53] Michael L. Waskom. seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60):3021, 2021. doi:10.21105/joss.03021.

[54] Felix Mölder, Kim Philipp Jablonski, Brice Letcher, Michael B Hall, Christopher H Tomkins-Tinch, Vanessa Sochat, Jan Forster, Soohyun Lee, Sven O Twardziok, Alexander Kanitz, Andreas Wilm, Manuel Holtgrewe, Sven Rahmann, Sven Nahnsen, and Johannes Köster. Sustainable data analysis with snakemake. *F1000Res.*, 10(33):33, April 2021. URL: https://f1000research.com/articles/10-33/pdf, doi:10.12688/f1000research.29032.2.

[55] Catherine Tang, Artem Krantsevich, and Thomas MacCarthy. Deep learning model of somatic hypermutation reveals importance of sequence context beyond hotspot targeting. *iScience*, 25(1):103668, January 2022. URL: http://dx.doi.org/10.1016/j.isci.2021.103668, doi:10.1016/j.isci.2021.103668.

[56] David B Jaffe, Payam Shahi, Bruce A Adams, Ashley M Chrisman, Peter M Finnegan, Nandhini Raman, Ariel E Royall, Funien Tsai, Thomas Vollbrecht, Daniel S Reyes, N Lance Hepler, and Wyatt J McDonnell. Functional antibodies exhibit light chain coherence. *Nature*, 611(7935):352–357, November 2022. URL: http://dx.doi.org/10.1038/s41586-022-05371-z, doi:10.1038/s41586-022-05371-z.

[57] Stefano Vergani, Ilya Korsunsky, Andrea Nicola Mazzarello, Gerardo Ferrer, Nicholas Chiorazzi, and Davide Bagnara. Novel method for high-throughput full-length IGHV-D-J sequencing of the immune repertoire from bulk b-cells with single-cell resolution. *Front. Immunol.*, 8:1157, September 2017. URL: http://dx.doi.org/10.3389/fimmu.2017.01157, doi:10.3389/fimmu.2017.01157.

[58] Eric Engelbrecht, Oscar L Rodriguez, William Lees, Zach Vanwinkle, Kaitlyn Shields, Steven Schultze, William S Gibson, David R Smith, Uddalok Jana, Swati Saha, Ayelet Peres, Gur Yaari, Melissa L Smith, and Corey T Watson. Germline polymorphism in the immunoglobulin kappa and lambda loci explain variation in the expressed light chain antibody repertoire. *bioRxivorg*, page 2025.05.28.656470, June 2025. URL: https://www.biorxiv.org/content/10.1101/2025.05.28.656470v1.abstract, doi:10.1101/2025.05.28.656470.

# Supplementary Materials

| purpose | name | samples | clonal families | PCPs | median mutations |
|---|---|---|---|---|---|
| train | `JaffePairedCC` | 4 | 50,776 | 209,599 | 7 |
| train | `TangCC` | 21 | 45,267 | 651,899 | 2 |
| train | `VanwinkleheavyTrainCC` | 149 | 21,269 | 124,985 | 4 |
| train | `VanwinklelightTrainCC1m` | 330 | 2,658 | 1,000,000 | 2 |
| test | `RodriguezCC` | 51 | 3,592 | 38,050 | 5 |

Table S1: Data used in this paper. CC means that PCPs with mutated cysteines are excluded from the data. `JaffePairedCC` is paired data from [56] sequenced using 10X. `TangCC` data is heavy chain data from [55, 57] and was sequenced using the methods of [57]. `VanwinkleheavyTrainCC1m` is a subset of the heavy chain data from [58] sequenced using Takara 5'RACE BCR kit. `VanwinklelightTrainCC1m` is a 1M subset of the light chain data from [58] sequenced using Takara 5'RACE BCR kit. `RodriguezCC` data is the $5'$ RACE heavy chain data from [12] and is used only for testing. The "samples" column is the number of individual samples in the dataset; in these datasets, each sample is from a distinct individual. "Clonal families" is the number of clonal families in the dataset. "PCPs" is the number of parent-child pairs in the dataset. "Median mutations" is the median number of mutations per PCP in the dataset.

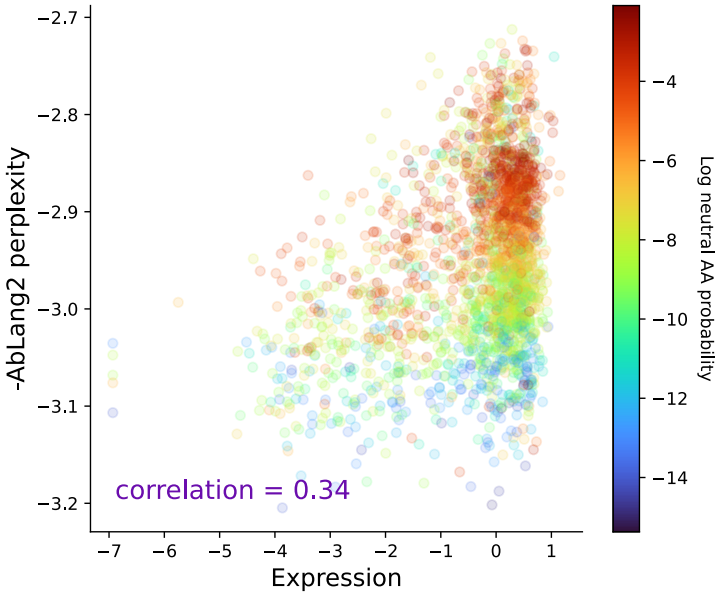


Figure S1: Conflating mutation and selection hinders functional prediction. This plot is analogous to the rightmost panel of Figure 1c, but colored by mutability according to the Thrifty [7] model.

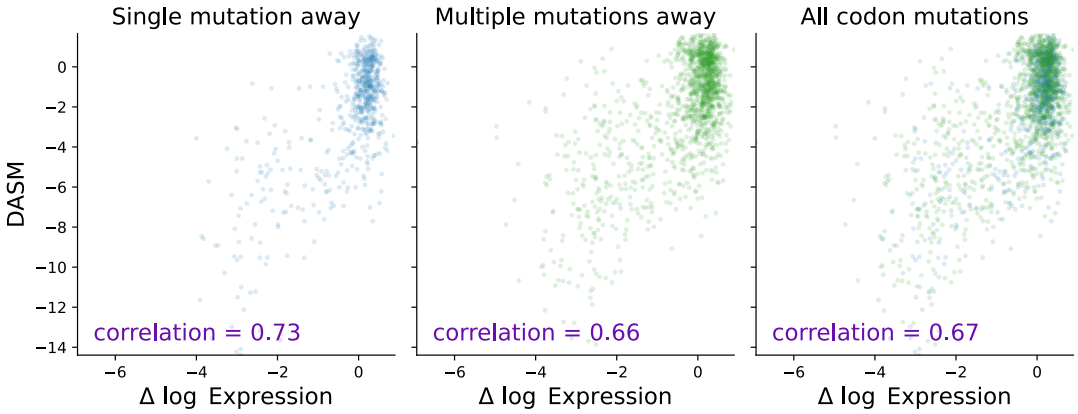


Figure S2: DASM removes codon bias in light chain functional predictions. Equivalent plot as Figure 3 but for light chain.

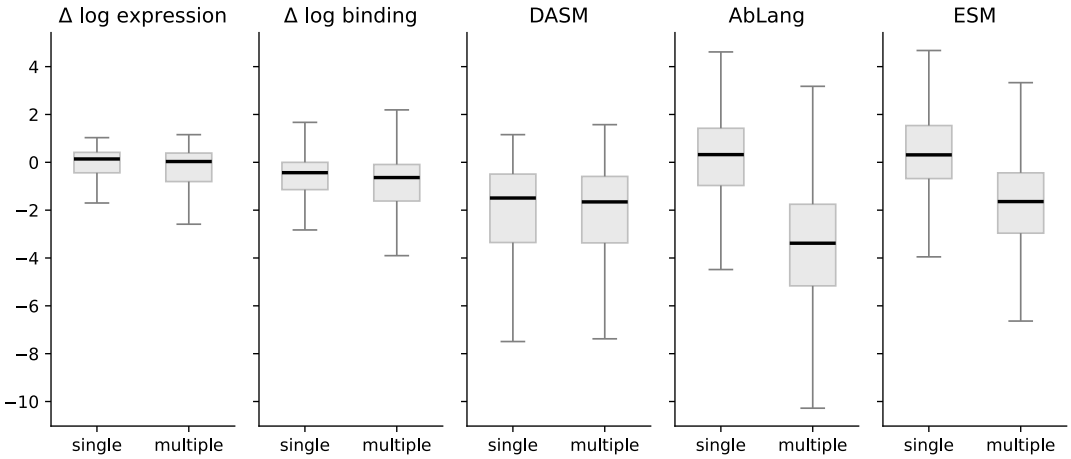


Figure S3: DASM selection factors are similar between codon neighbor amino acids and non-neighbors (compare Figure 1a), in contrast to other models. Comparison done on the heavy chain of the Koenig [11] wildtype sequence.

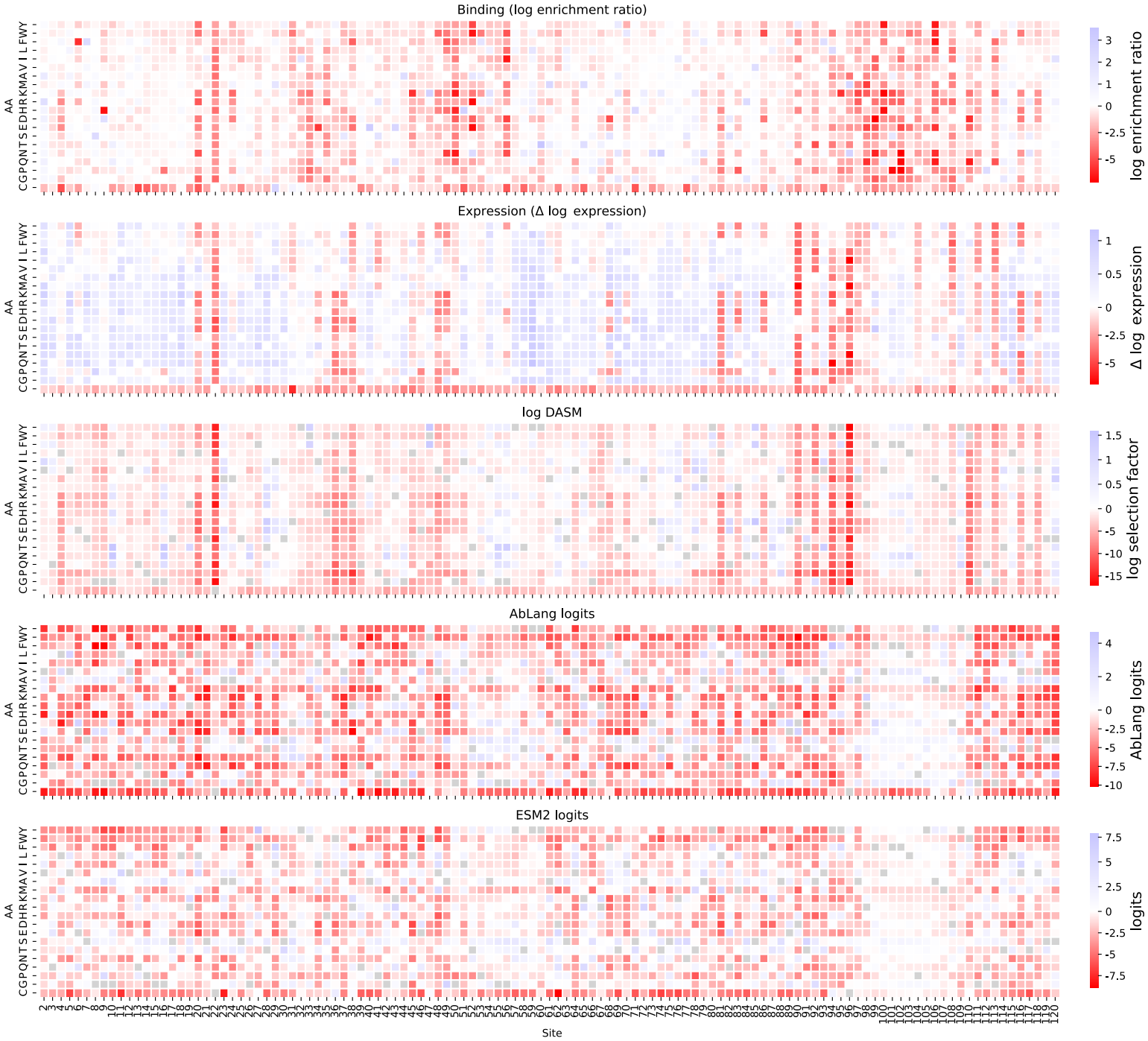


Figure S4: Heatmap showing the heavy-chain data of [11] along with model predictions.

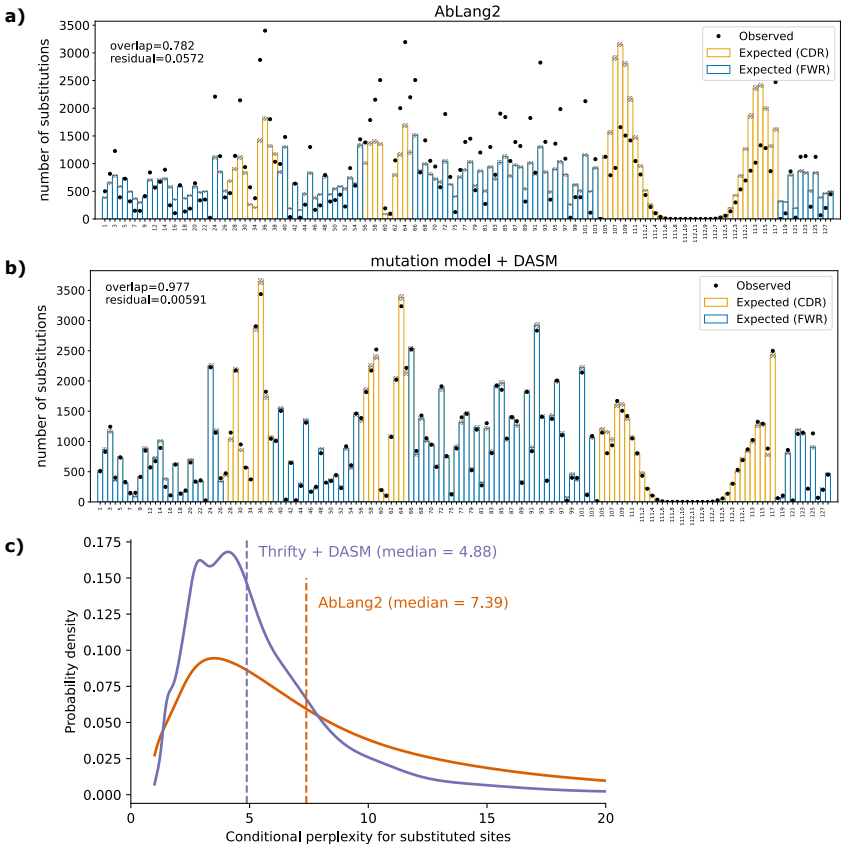


Figure S5: The DASM with the Thrifty neutral model accurately assesses probabilities of natural affinity maturation paths. **a**, **b**: The DASM is better than AbLang2 at predicting the location of nonsynonymous mutations observed in PCPs withheld from model training. **c**: The DASM achieves lower conditional perplexity (median 4.88 vs 7.39) when predicting amino acid identity at mutated sites, with fewer extreme outliers. The conditional perplexity is the perplexity of the child amino acid, conditioned on there being a mutation at that site. Note that due to the inherent stochasticity of affinity maturation, there is a lower limit to this conditional perplexity that is substantially greater than 1.

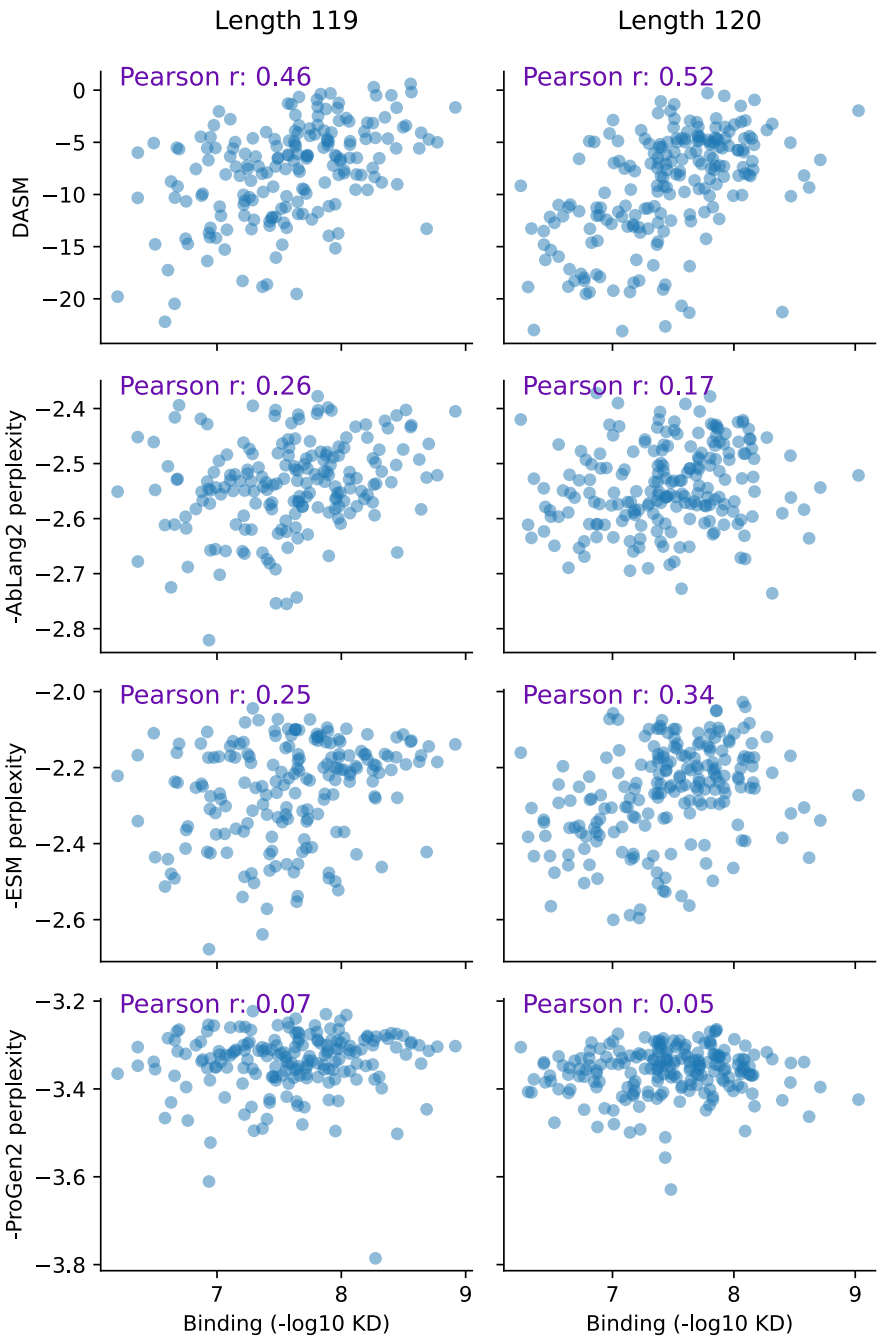


Figure S6: Scatterplots for the data of [23] zero-shot data set, partitioned by sequence length.

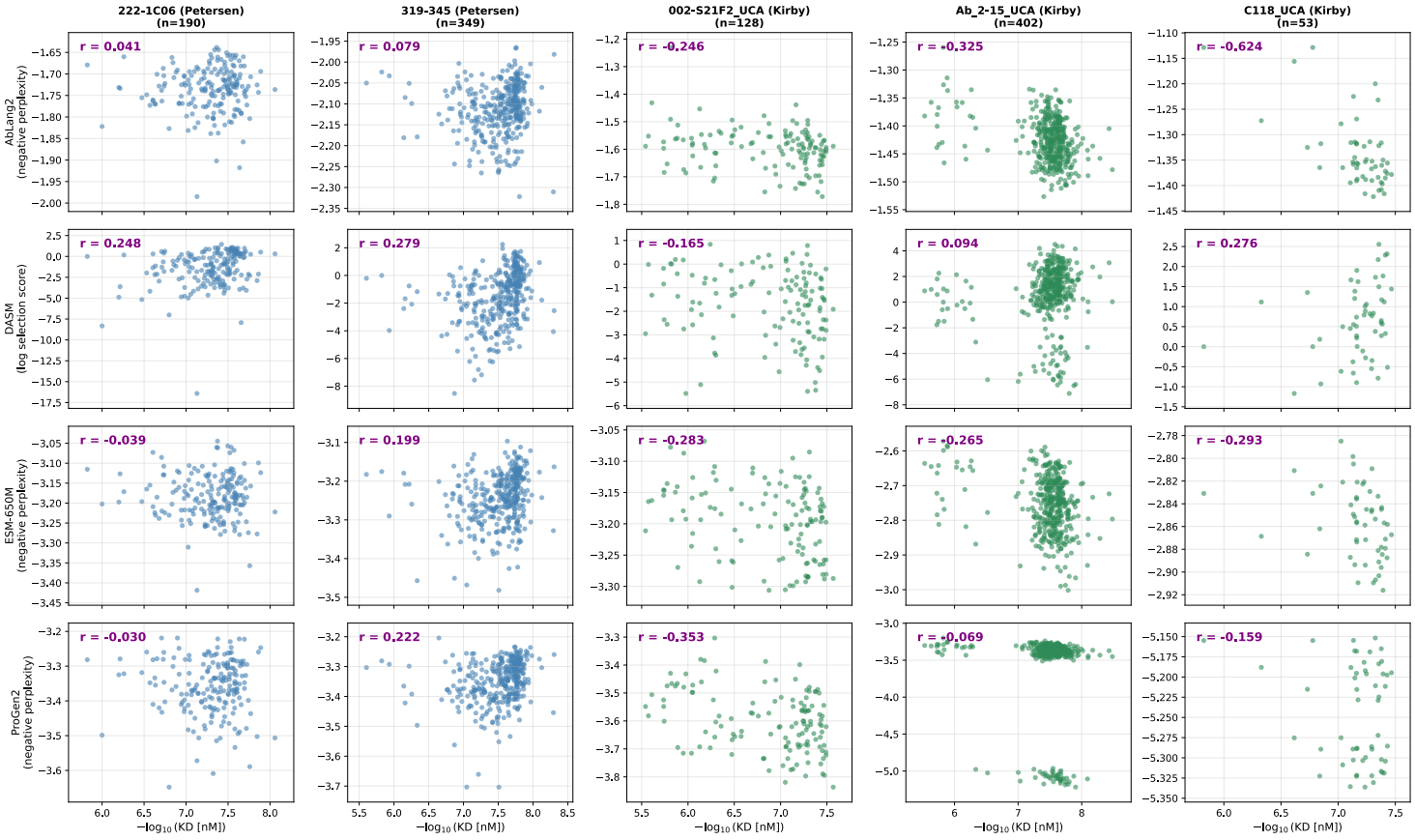


Figure S7: Model predictions versus experimentally measured binding affinity for five antibodies using the MAGMA-seq protocol. "Petersen" data are from an experiment probing the rules of recognition for influenza neutralizing antibodies [24] and "Kirby" data are from combinatorial libraries applying combinations of mutations on the path from naive to mature [26]. Each row shows a different model (DASM, ESM2, AbLang2, ProGen2) and each column shows a different antibody lineage.

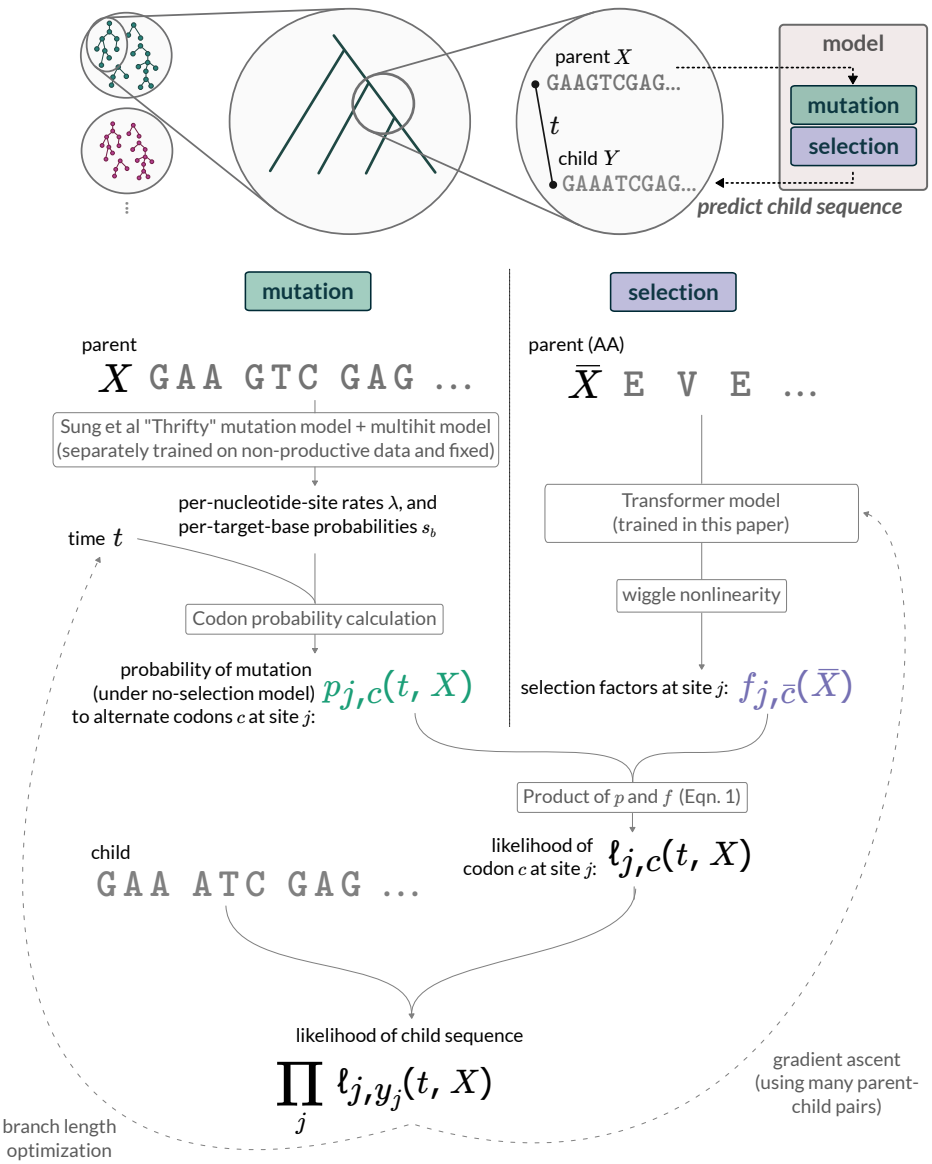


Figure S8: The model is trained to predict the probability of a child sequence given the parent sequence. It is divided into mutation and selection components. Mutation: given a parent sequence $X$, probability of mutation to alternate codons after time $t$ is calculated using a model of SHM [7] and a "multihit model" (see Methods) and then aggregated into codons as in [15] to obtain $p_{j,c}(t, X)$. Selection: given an amino acid translation $\bar{X}$ of the parent, the transformer-encoder gives per-site selection factors $f_{j,\bar{c}}(\bar{X})$. These are then multiplied (1) and summed to give the probability of the observed child sequence at every site. This gives a likelihood for a parent-child pair. The algorithm maximizes the likelihood across branch lengths $t$ for each parent-child pair as well as across the parameters of the transformer model for all parent-child pairs in the dataset (dashed lines).

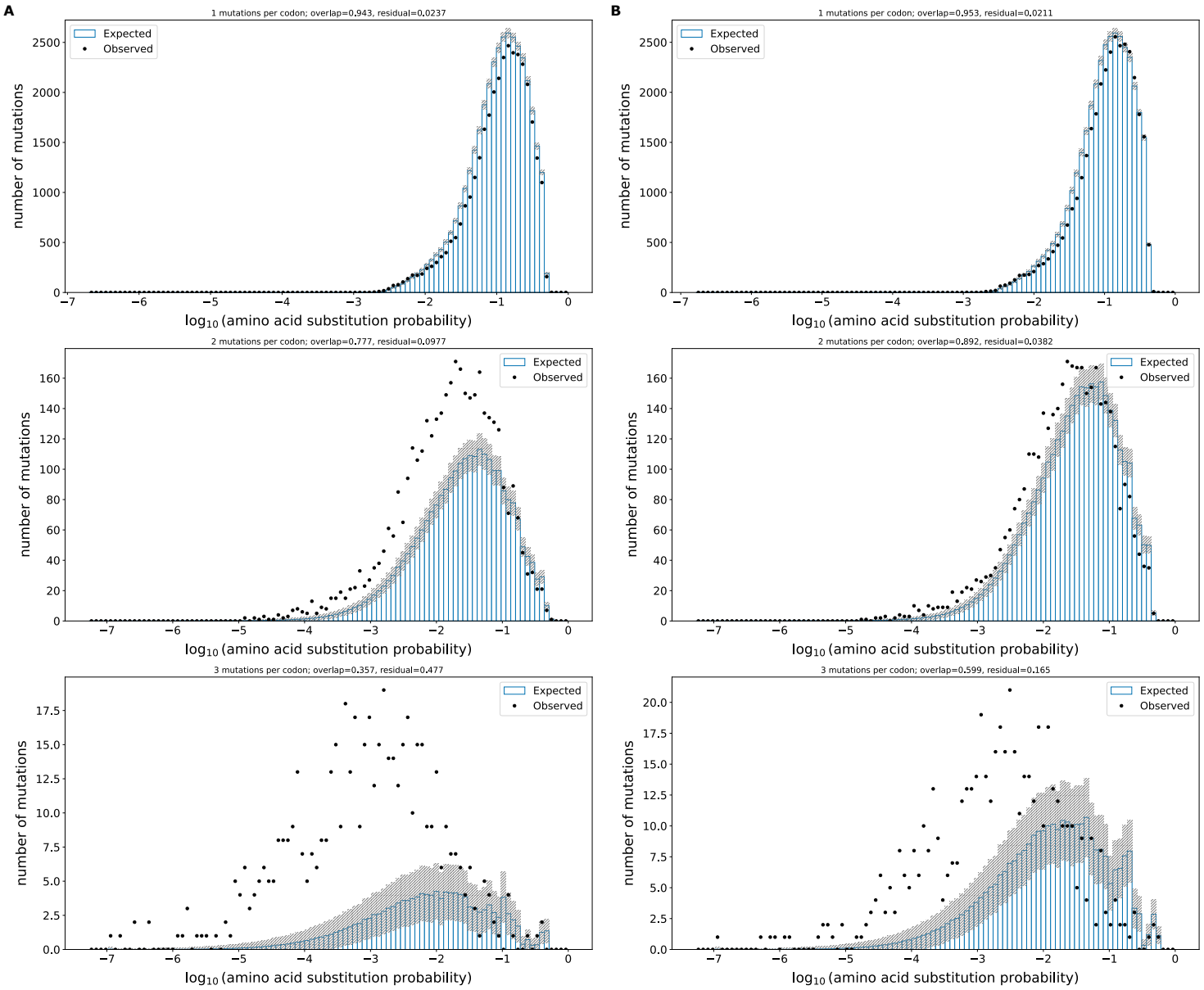


Figure S9: Models that allow for multiple hits per codon (right column) show better model fit than without (left column) on the out-of-frame data from [55] as prepared in [7]. These observed-versus-expected plots compare the observed number of mutations to the computationally predicted number across bins of mutation probability. For each bin, the expected number of mutations is calculated as the sum of mutation probabilities for all sites in that bin, while the observed count represents the actual mutations found in those sites. The overlap metric quantifies the area of overlap between observed and expected distributions divided by their average area, while the residual metric measures the normalized root-mean-square difference between observed and expected counts. These plots are faceted by the number of mutations per codon.