Learning Latent Design Spaces in Protein Generative Models

Thesis by Pierce Hoenigman

In partial fulfillment of the requirements for the degree of
Bachelor of Science in Biological Sciences, Specialization in Quantitative Biology
With Research Honors

University of Chicago
Chicago, IL

Advised by Rama Ranganathan, MD, PhD

Thesis Committee:
Andrew Ferguson, PhD
Laurie Comstock, PhD

2024-2025 Research Honors Program
Directed by Allan Drummond, PhD

TABLE OF CONTENTS

List of Abbreviations

DAE	Disentangling AutoEncoder
DCA	Direct Coupling Analysis
DCI	Disentanglement, Completeness, and Informativeness
ELBO	Evidence Lower BOund
GF-Score	Grid Fitting Score
ICA	Independent Component Analysis
ICOD	Inverse Covariance Off-Diagonal
iid	independent and identically distributed
KLD	Kullback-Leibler Divergence
MMD	Maximum Mean Discrepancy
MSA	Multiple Sequence Alignment
PCA	Principal Component Analysis
PDB	Protein DataBank
SCA	Statistical Coupling Analysis
SVM	Support Vector Machine
VAE	Variational AutoEncoder
β-TC	β-Total Correlation

ACKNOWLEDGEMENTS

I would first and foremost like to thank Professor Rama Ranganathan for enabling this research and all the practical and intellectual growth it has brought me. Despite my scant knowledge of machine learning in February 2024, Rama entrusted me with a mostly independent project on the subject because of my desire to try computational research. Over the past year, this project has allowed me to learn through trial and error in months what would have taken me years of academic study to understand about machine learning and protein design. Even more importantly, Rama's physics-rooted perspective on biology has allowed me to see the subject I've been studying over the last four years in a completely different light; this quantitative perspective has reshaped my own ideas, course of academic study, and future career.

The support of others in the Ranganathan Lab to this project has been particularly crucial. The code for running the initial MMD VAE model was written by Nikša Praljak and Andrew Xu. Andrew has been greatly supportive over the past year, being responsive when I needed help designing a method or interpreting a result. I thank Nikša for keeping me updated on the state of the art in the field of protein generation, and for his depth of theoretical understanding; he has caught methodical errors in my procedure which I would likely never have seen. Dr. Federica Ferretti, who I have been working with to test the RBM models, has also been greatly helpful and kind in comparing and interpreting results. I also thank Tyler Kramlich for his mentorship during my initial months transitioning into the lab, and the rest of the lab for the enlightening conversations they have provided.

Next, I would like to thank a few Ranganathan Lab collaborators with whom I have discussed this project. Professor Madhav Mani of Northwestern University has been crucial in shaping the trajectory of my research into RBMs and in determining a sensible and theoretically sound experimental plan. Professor Olivier Rivoire of CNRS in France and Dr. Yaakov Kleeorin have also been helpful in making time for me to discuss their prior work to better understand different aspects of this research.

In addition, I would like to acknowledge my thesis mentors. Professor Andrew Ferguson is greatly knowledgeable about the field and has provided me useful suggestions on incorporating β-VAEs and other particular models which I would like to expand to. I thank Professor Laurie Comstock for her advice on structuring and presenting my research in an understandable way and for her general enthusiasm about the project. I thank Professor Allan Drummond for his guidance on the presentation of this work, and especially for his calming assurances about the thesis process.

Lastly, I thank my family and friends. My friends who have also been writing theses for their camaraderie when I just needed to know that someone else was as behind in their research as I was. In particular, I would like to thank Sonia Yuan for introducing me to the Ranganathan Lab. My family for their support and excitement for everything I do, and for their encouragement to embark in this new direction when I would have been much more comfortable continuing with wet lab research. This work is dedicated to anyone who has had to put up with me in times when this thesis has been my primary attention.

ABSTRACT

Protein design is a long-standing goal of synthetic biology and medicine. Despite recent advances in predicting protein structure from sequence data using deep learning methods, predicting and designing protein function from families of sequences remains an open question. Nonlinear, neural network-based models face two major challenges; first, they do not provide control over the particular properties of generated proteins, rather providing a natural-like distribution from which to sample. Additionally, protein generative models are overfit to phylogenetic noise and thus do not generate the full spectrum of sequence diversity desired. Here, I show that biasing variational autoencoder model learning with correlational information from multiple sequence alignments in toy protein datasets improves latent protein feature disentanglement, leading to a somewhat more interpretable latent space. I additionally demonstrate the difficulties in learning the protein space by showing how model interpretability decreases when toy sequences are sampled phylogenetically or with more correlational features. Next, I apply these findings to the S1A family of serine proteases, and find that improvements in learning behavior largely mirror those of toy models. Imposing greater correlational information on learning is found to alter generated sequence diversity, but maintains high-quality folding. These findings emphasize the importance of understanding the peculiar structure of protein data and establishing better formalisms for protein function in generative learning.

BACKGROUND

1.1 Protein Sequence and Function

Of the four major classes of biological molecules, proteins fulfill nearly all active, catalytic roles within the cell along with many structural purposes (1). Understanding protein properties is therefore essential both to attain a precise understanding of intracellular mechanics and to engineer proteins toward specific purposes in medicine, agriculture, or bioproduction. One manner of understanding of protein properties is to understand the protein sequence as encoding information about structure, function, and evolvability. Proteins generally fold independently of assistance in their native environment and modifications to their function often take the form of allosteric binding, the sites and signal transfer networks of which are sequence-encoded (2,3,4). Understandably, these properties are entangled; the amino acid sequence of chymotrypsin encodes the catalytic triad and determines binding pocket structure which leads to the particular ligand specificity and catalytic functionality of the enzyme (5). Likewise, a protein's binding affinity for two different substrates can determine its ability to evolve as a generalist or specialist depending on the evolutionary pressures applied (6).

Recently, the first major piece in understanding these protein properties has been achieved with AlphaFold structural prediction (7). Though not perfect for some cases such as disordered proteins and not yet applicable to protein complexes, the broad success of AlphaFold and its refinements over the last few years has shown that deep learning methods can be applied to solve difficult biological problems (8). With structure mostly solved, the next major piece in the protein puzzle is function. Unlike structural prediction, protein functional prediction has a wide range of potential outputs; rather than 3D atomic coordinates, function can be interpreted through a wide range of biological assays determining properties such as reaction rates, binding specificity, thermal stability, and more. Worse, while protein databases like Protein Data Bank (PDB) have standardized structural data storage, information on functional data is much less centralized (9). This difficulty means the problem of protein functional prediction and generation is most easily approached as an unsupervised problem, such that these properties must be learned from the sequences alone. Still, there are some supervised attempts at this problem which have achieved success in various measures, such as RFDiffusion and BioM3 (10,11). Nevertheless, as established, protein function is determined by structure and therefore by sequence, so functional information on proteins should theoretically be learnable through analyzing sequences alone.

One important factor contributing to the difficulty in learning sequence data is its structure. The assumptions behind many machine learning techniques require that their input data be independent and identically distributed (iid), meaning that each sample is mutually independent from every other sample and all come from the same distribution (12). This assumption is notably not satisfied in the case of protein data, which was generated through evolutionary process; the hierarchical, phylogenetic relationships between the sequences negate inter-sequence independence. Despite this issue, models assuming iid data are still applied to protein generation since a solution in model architecture has not been found. These models are still able to generate functional proteins, but the lack of filtering of phylogenetic correlations in the protein data from evolutionary selection-generated correlations has in part led to the interpretability and generated diversity complaints initially raised (10,11,13).

Neural network-based models of sufficient size and nonlinearity are known as ‘universal function approximators’ for their ability to fit the patterns in nearly any set of training data when given the appropriate architecture and training regime (14). This is an important motivation toward choosing neural network-based models, as the sequence to function mapping is extremely complex and nonlinear (15,16). Epistasis studies from deep mutational scanning data have shown that mutational effects are nonlinear up to the seventh order, i.e. that in some cases knowing the functional effects of mutating every combination of six residues in a group of seven would not enable the prediction of effects when mutating all seven residues (16). It is therefore important to choose a model as flexible to this nonlinearity as possible, a task which neural networks excel at.

Nevertheless, choosing the model architecture and training regimen is not a simple task, especially with the difficulty of finding proper metrics to evaluate protein function when using sequences alone. This difficulty again arises from the structure of the data. Due to its evolutionary generation, there are two contributing factors to residue correlations: functional correlation and phylogenetic correlation (13). Functional correlation represents the mutual conservation of amino acid positions which are important to a particular function of a protein or to its stability, and is generated through evolutionary selection. Phylogenetic correlations, meanwhile, show themselves in conservation of positions caused by many closely related members of the protein family having evolved from a common ancestor. As both forms of correlation show themselves in the positional correlation matrix of the protein family MSA, they are difficult to disentangle. Decoupling of these two forms of correlation is crucial for the generative portion of protein functional learning, as it is preferable for generative models not to be constrained by the existing course of evolution. The protein space is functionally infinite (on the order of 10^{130} possible sequences for a small 100-residue protein) and the evolutionary trajectory of existing protein families is but one random course out of likely many more that finds the solution to a certain problem (17). The ideal generative model would thus be able to uncover diverse solutions to the same functional problem rather than overfitting to existing evolutionary patterns, since the vastness of sequence space likely contains even more optimized solutions to natural and desired protein functions.

1.2 Protein Functional Decomposition

We will begin by inspecting some of the important models in the space of protein functional understanding, focusing on those which are used to predict protein functional properties but are non-generative. These include Direct Coupling Analysis (DCA), eigenspectrum cleaning, Inverse Covariance Off-Diagonal (ICOD), and Statistical Coupling Analysis (SCA).

The success of DCA is primarily in contact prediction, a test of identifying which residues in the protein are in close proximity to one another. This test is often used as a proxy for identifying correlations important to protein function, since contacting residues are able to physically influence one another and therefore send signals or actuate conformational change through the protein (18,19). The DCA method, like many, begins with a multiple sequence alignment (MSA) of proteins in a family of choice. The MSA is important as it allows the detection of positional differences between sequences. After downweighting sequences with high similarity (to prevent one large clade in the family from dominating the results), DCA calculates positional and pairwise amino acid frequencies for all positions and pairs of positions in the MSA. Using these measurements, DCA computes the direct information between each pair of amino acids, which is essentially the part of the mutual information between the two positions not caused by coupling between other sites of the protein (20). While DCA is successful in predicting residue contacts apart from evolutionary noise, this structure-based proxy for function is often not truly in alignment with functionally important positions. Functionally orthogonal groups of covarying positions known as sectors provide a more precise functional prediction task, however a more difficult one to measure since sectors have no ground truth measurement unlike contacts which can be gathered from structural data (21). The only way to truly validate sector identification is to perform excisions or swaps of sector positions, for example by converting the trypsin protease to chymotrypsin specificity without affecting other functions by swapping specific sequence segments between the two (22). It has been found that DCA is able to predict these larger sectors under a regularization regime inverse to that which allows prediction of local contacts (21).

One method which attempts to disentangle functional and phylogenetic signals in the covariance matrix was proposed by Qin and Colwell (13). Using random matrix theory, they showed that in a phylogenetically generated MSA without functional constraints, the eigenvalue distribution of its covariance matrix should follow a power law. In real protein families, this power law holds for all but the smallest eigenvalues, which they inferred was the result of functional covariance signals. By removing the top eigenmodes from the covariance matrix and leaving only those which were inferred to be functionally relevant, they were able to improve contact prediction by approximately sevenfold. Removing additional, small eigenmodes reduced the contact prediction precision thereafter (13). The method of eigenspectrum cleaning is important in showing a property of protein data and suggesting that the two types of correlation may be statistically disentangled. However its power is likely in combination with methods which identify specific functionally coevolving sectors in the protein family.

The ICOD method highlights a similar part of the eigenspectrum as Qin and Colwell’s cleaned eigenspectrum method, however without the need to infer family-specific cutoffs between large and small eigenvalue regimes. By inverting the MSA covariance matrix, the small, functionally-relevant eigenmodes become the most important. Additionally, the diagonal is set to zero, de-emphasizing conservation (23,24). ICOD has proven particularly useful in predicting sites of greatest mutational effect, outperforming SCA, but it has not yet been developed to compare sequences in sector space (23,24). This is important for the purpose of design, as we are not just interested in the groupings of important positions in the protein, but in differentiating function between multiple sequences such that protein design with precise functional specifications may take place.

Lastly, SCA is a method that more explicitly identifies functional sectors (25). After down-weighting closely related sequences, SCA adapts the MSA pairwise correlation tensor to measure the *significance* of conservations using the Kullback-Leibler entropy of each position. Compression of the resulting tensor results in a mapping between the spaces of positional and sequence correlations (26). Independent component analysis (ICA) is then utilized to identify the sectors and identify sequence positions in sector space (26). These adjustments for significance, rather than straight conservations and correlations, have allowed SCA to parse functional dimensions of protein families through phylogenetic noise. Although SCA performs worse than ICOD at contact prediction and predicting individual mutant effects due to its emphasis of the large eigenvalues representing conservation, it remains the only such method able to form a function space of a given protein family (13,23,24).

1.3 The S1A Family of Serine Proteases

The S1A protein family is a good model for testing protein learning and design due to its clear decomposition into functional groups and the large amount of available functional data on the family. The proteins of this family are largely proteases, often containing the catalytic triad of serine, histidine, and aspartic acid (5). Functional decomposition by SCA results in seven sectors. The largest mode effectively divides the family into haptoglobins and non-haptoglobins (27). Haptoglobins are non-catalytic proteins which bind to free hemoglobin in the bloodstream, and thus this first mode separates the non-catalytic members from the catalytic members of the family (27). The second mode separates trypsins and tryptases from chymotrypsins, with trypsin and tryptase having catalytic specificity for positively charged amino acids such as arginine and lysine while chymotrypsins have catalytic specificity for aromatic amino acids (27). Kallikreins are mostly on the side of trypsins and tryptases, but their division expresses the family's mixed specificity between trypsin-like and chymotrypsin-like specificity (27,28). Granzymes are even more all over the place, expressing the family's mixed trypsin/chymotrypsin and unique specificities (27,29). The third mode mostly expresses phylogeny, separating invertebrate from vertebrate proteases. However there is also a division between these two groups in the protease's catalytic lifetime, confounding whether this division is phylogenetic or functional (27). Regardless, the primary two modes do represent a functional, rather than phylogenetic, division. The final four modes are associated with protein stability more so than any particular function, emphasizing the finding that top functional modes give way to smaller, more numerous structural modes as seen in DCA regularization (21,27). Due to prior studies on this family, it has a large corpus of functional and mutational data and thus is a good target for understanding model learning.

1.4 Potts Models

Potts models are a class of generative model which enable turning the MSA covariance matrix of a protein family into a Hamiltonian energy model, which may then be used to sample sequences in order to generate new proteins. Field and coupling terms, representing singular positional conservation and pairwise covariance, respectively, make up the Hamiltonian and are inferred from the MSA. Using the Metropolis-Hastings sampling algorithm, a random sequence is presented with sequential mutations which it chooses to accept or reject based on probabilities dependent on whether they are lower energy states than the existing sequence (21,30). This is a simple means of sequence generation and has been shown to produce functional sequences, but there is little control over the particular generated sequence function and the diversity of generated sequences is poor compared to natural distributions, demonstrating overfitting (31). An additional concern is that different regularizations of the Potts couplings can highlight different correlation features in the covariance matrix while downplaying others. For example, large sectors of conserved positions important for function are highlighted by large regularization while small regularization terms highlight individual correlations important for protein stability (21).

1.5 Variational Autoencoders

The variational autoencoder is a deep learning model architecture which provides an opportunity to combine both the functional decomposition of coupling analysis methods with the generative capability of Potts models. VAEs come in many flavors but generally work by using a neural network 'encoder' to compress input data (a protein sequence from the MSA) into a low-dimensional 'latent' representation characterized by a multidimensional Gaussian with learned means and variances (32). A point sampled from this distribution can then be decoded using another neural network, the 'decoder,' into a novel protein sequence. Model training is controlled by a loss function that generally includes a cross-entropy term between the input and output sequences ('reconstruction loss'), ensuring that the model learns to generate outputs which appear to come from the same distribution as the protein family. Other loss terms such as Maximum Mean Discrepancy (MMD) between the input sequences and latent space may also ensure an optimal structure for the latent space, not overly sparse or compressed (33). However, as the VAE loss function derives from variational Bayesian techniques to marginalize over intractably large input and latent spaces, it also requires iid data (12,32).

While there are many model architectures with generative capabilities, VAEs stand out as one of the only configurations which produce an easily accessible compressed representation of the data. This is a useful property, as the distance between encoded points (proteins) in the latent space can convey information about the relationship between those proteins, perhaps encoding information about a protein's function. VAEs have been able to generate functional proteins with greater (though still limited) diversity of active sequences compared to Potts models (34). Unfortunately, the VAE latent space is typically entangled, meaning latent dimensions express a combination of functional features of interest rather than each feature of interest being expressed solely in its own latent dimension (35). In the case of proteins, this means that models have thus far been overfit to phylogenetic correlations while functional signals in the data are spread between dimensions. There are a multitude of adaptations to VAEs which have proposed improvements on the disentangling capabilities of the VAE (36,37,38,39). The most successful of these techniques, β-VAE and its successor β-TC VAE, re-weight a particular term of the loss function responsible for total correlation, a measure of variable dependence (36,37). This results in the model being penalized for not finding independent factors in the data distribution, leading to better disentanglement.

RESEARCH AIMS

Aim 1. A) Explore the dynamics of VAE model learning on toy protein datasets with varying complexity to determine the best model architectures for latent protein feature disentanglement, and **B)** examine how these findings are affected by phylogenetic sampling of toy data in order to get closer to real protein alignments.

Aim 2. A) Use the findings from Aim 1 to inform VAE design for models trained on the S1A family of serine proteases; assay latent protein feature disentanglement and **B)** the diversity and folding quality of sequences generated by the S1A-trained VAE models.

RESULTS

2.1 Added Constraints Improve Toy Model Disentanglement

To begin investigating the dynamics of VAE learning on toy datasets, a number of VAE models and toy sequence alignments were designed. First, an MMD VAE model as described by Zhao, Song, and Ermon was used (Fig. 1A), as this is the model which was used in the Ranganathan lab previously (33,34). Alongside this typical MMD VAE model, four loss functions were designed to alter learning patterns based on the MSA data. The first of the new loss functions incorporates conservation significance values from SCA, D_i^a. These are generated using the frequencies of each amino acid a in the MSA at position i, f_i^a, and taking the Kullback-Leibler divergence (KLD) with respect to the background distribution of amino acids across life, q^a (26). The partial derivative is then taken with respect to f_i^a to get the degree of conservation used in the model, $\frac{\partial D_i^a}{\partial f_i^a} = \phi_i^a$ (26).It was hypothesized that multiplying these positional weights with positional reconstruction loss values would penalize the model to focus on learning positions with important conservations while giving it more leeway on positions which are not expected to be conserved given the background.

In a similar vein, the pairwise correlation significance matrix calculated by SCA, $\tilde{C}_{ij}$, was used to alter the loss function in another new MMD VAE model. This matrix is calculated by taking the raw correlations between two amino acids (a, b) at two positions (i, j), C_{ij}^{ab}, and multiplying by the significance of conservation at each position, $\phi_i^a \phi_j^b$, then taking the Frobenius norm across the amino acids (26). The $C_{ij}^{\sim}$ matrix was then multiplied with the outer product of the reconstruction losses before taking the norm, such that the model would be penalized more harshly for missing more significant pairwise correlations in the MSA. In theory, the full $C_{ij}^{\sim ab}$ tensor could have been used to be more precise with this penalty, however the data in this tensor is quite sparse due to its large size compared to the data and would be inefficient to use in training.

The next MMD VAE model uses ICOD as described by Wang, Bitbol, and Wingreen (23). The covariance matrix of the MSA is taken, inverted, and the diagonal elements are set to zero. This is then applied to the reconstruction loss in a similar way to the $C_{ij}^{\sim}$ penalty described above. As ICOD builds upon a line of work that smaller covariances are actually more important for understanding protein constraints, I thought to test this hypothesis in the generative learning case by weighting the reconstruction loss with inverted positional covariances (13,23,24). This penalizes pairs with large correlation and down-weights individual conservation.

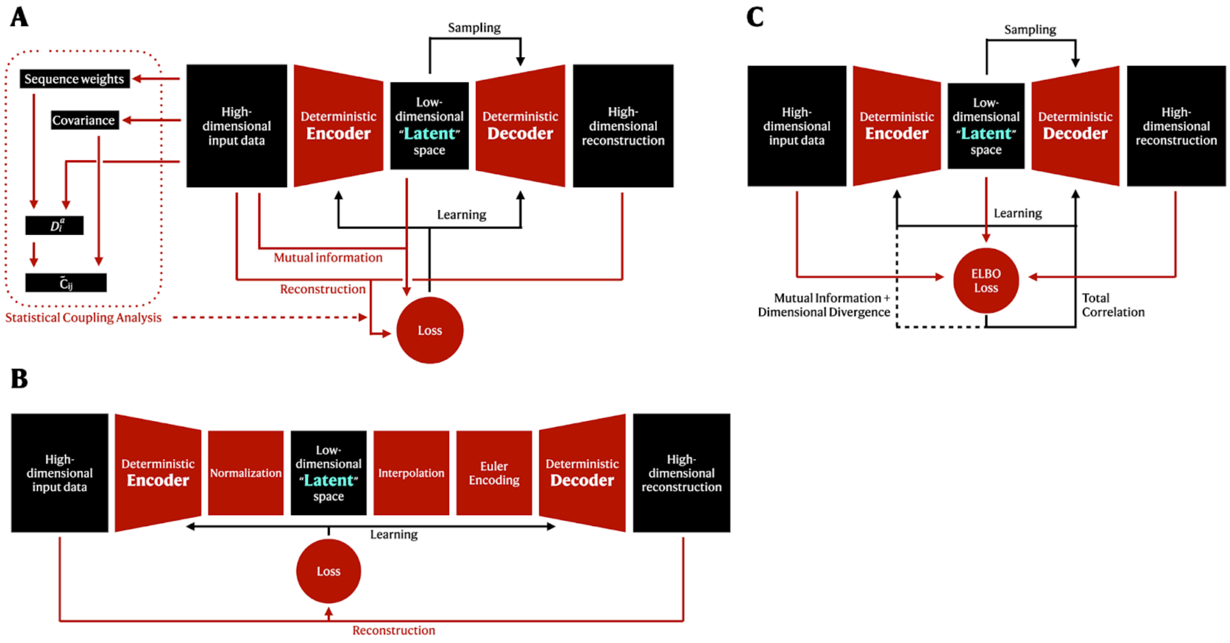

Figure 1. Comparison of VAE architectures and losses.

A) MMD-VAE architecture and components of SCA used in modifying the reconstruction loss in the described variations, **B)** DAE architecture, and **C)** β-TC VAE architecture, where mutual information and dimensional divergence ELBO components are used only in the vanilla VAE case. Trapezoidal boxes represent neural networks, while other red shapes describe different non-neural functions and black shapes show the location of accessible data in the models.

With these models in hand, the next task was to design a number of toy alignments to test them on, in order to determine the effect of increasingly complex correlations on model learning. Six patterns of correlation were chosen, five of which use a length of 6 and 4 possible amino acids: first, one with no correlations as a baseline control. Next, one with only positional conservation at the second and fourth positions. Third, two conserved pairs at the first/second positions and third/fifth positions. Fourth, three mutually conserved pairs forming a sector at the fourth through sixth positions. Fifth, a combined MSA with the positional and sector constraints, and the first/second position pair. Lastly, a larger alignment was tested with a length of 9 and 4 possible amino acids that includes three sectors of mutually conserved pairs, from first through third position, fourth through sixth position, and seventh through ninth position. This final pattern was designed to mimic the pattern of correlations in the S1A family, which has three such sectors of correlation (21,27).

These conservation and correlation constraints were input as fields and couplings in a Potts model developed by Kleeorin et al., 2023 to generate the toy MSAs (Fig. 2A) (21). The size of the toy protein family and number of amino acids were chosen as low values such that essentially complete sampling of the toy protein space could be achieved, as $4^6 < 10000$ (the number of sequences in each alignment). As such, the models would not face the same sparse coverage of the protein space as actual protein models. Sparse coverage was assayed with the larger S1A-like alignment, as $4^9 >> 10000$. The MSAs were then run through the pySCA package to obtain the D_i^a and $C_{ij}^{\sim}$ (Fig. 2A) matrices used in the SCA-based models (26).

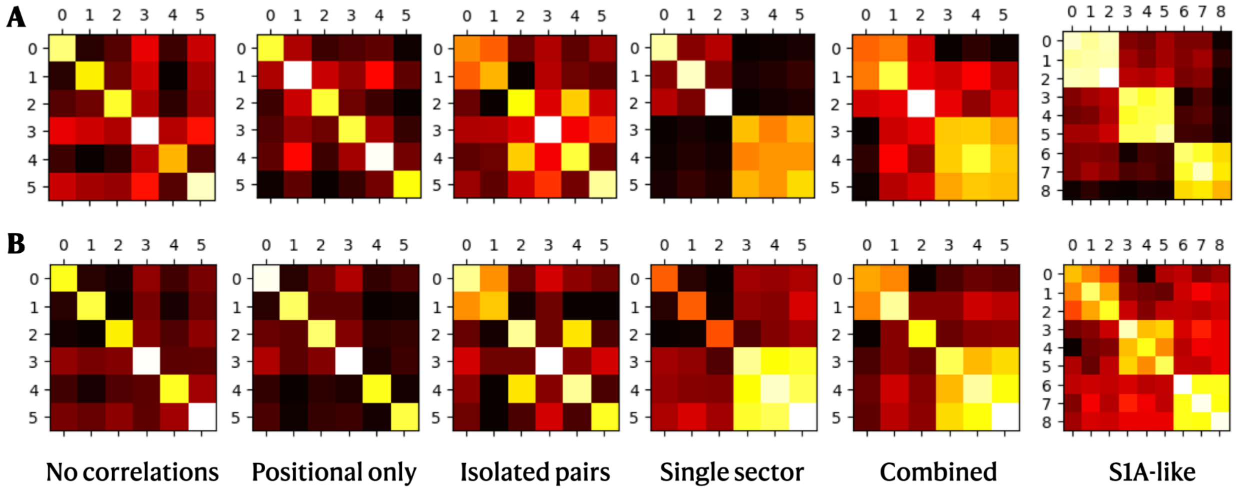

Figure 2. Comparison of toy alignment correlations.

A) Pairwise correlation matrices of Potts model-generated toy MSAs based on each of the labeled constraints, and **B)** pairwise correlation matrices of phylogenetic-Potts-sampled toy MSAs based on the labeled constraints. Matrices are colored on a log scale.

These encoded conservation and correlation constraints can be thought of as the functional factors to be disentangled, as there is an algorithmic pressure causing the conservation analogous to the evolutionary fitness function. To measure disentanglement, nearly all metrics require ground-truth *factors* (i.e., the protein functional dimensions) which are not known in real protein data. One of the most consistent metrics for disentanglement is Disentanglement, Completeness, and Informativeness (DCI) score, which breaks disentanglement down into three components (40). In this sense, disentanglement means that each *code* (position on a latent dimension) corresponds to only one factor (41). Completeness means that each factor only corresponds to one code, while informativeness means that all factors are represented by codes (i.e., all information in the function space is represented in the latent space) (41). In the toy model experiments, SCA was used to detect each covarying feature (e.g. a sector or a coupling) as a factor dimension of the MSA. These can be considered factors since they capture all input constraints on the data. As DCI needs at least three factors to be useful, only the combined and S1A-like alignments were tested with DCI (Fig. 3C).

One of the only metrics that claims the ability to capture disentanglement without using known ground truth codes was Grid-Fitting Score (GF-Score) as described by Cha and Thiyagalingam (35,39). This score measures the degree to which the distribution of training data is uniform over a multidimensional grid. Based on the work of Higgins et al., 2018, in proposing a definition of disentanglement based on symmetry groups, Cha and Thiyagalingam posit that more uniform distribution of the latent space indicates better disentanglement (35,39).

We see a number of interesting trends between model and toy alignment in terms of disentanglement scoring (Fig. 3A). The positional VAE training did not succeed for the non-phylogenetic alignments, likely due to very low positional weights resulting from saturated sampling of the small sequence space. The ICOD MMD model appears to do quite well in terms of GF-Score, but this advantage is diminished somewhat with more complex alignment patterns. The pairwise-constrained MMD model also seems to do better in general than the regular MMD model; it is quite consistent in its score and actually improves with the more complex S1A-like alignment. Surprisingly, however, the regular MMD model does quite well with the S1A-like alignment even though it does quite poorly with the rest of the alignments. It seemingly could be tripped up by positional conservation patterns in the alignment, as it did worst on alignments which included those. Turning to DCI score, all models appear to do quite poorly on disentanglement and completeness (Fig. 3C). The models are much more successful at informativeness, however. Interestingly, none of the models are able to capture the total features of the S1A-like alignment as well as in the combined conservation/pair/sector alignment, showing uniformly worse (but still good) informativeness across the board. On the whole from GF-Score, it appears that models with added constraints (in particular ICOD) perform better on disentanglement. Still, the regular MMD performance on S1A-like data called for the following experiments to observe if this trend holds when getting closer to a real alignment.

2.2 Added Constraint Advantages Persist in Phylogenetically-Sampled Toy Models

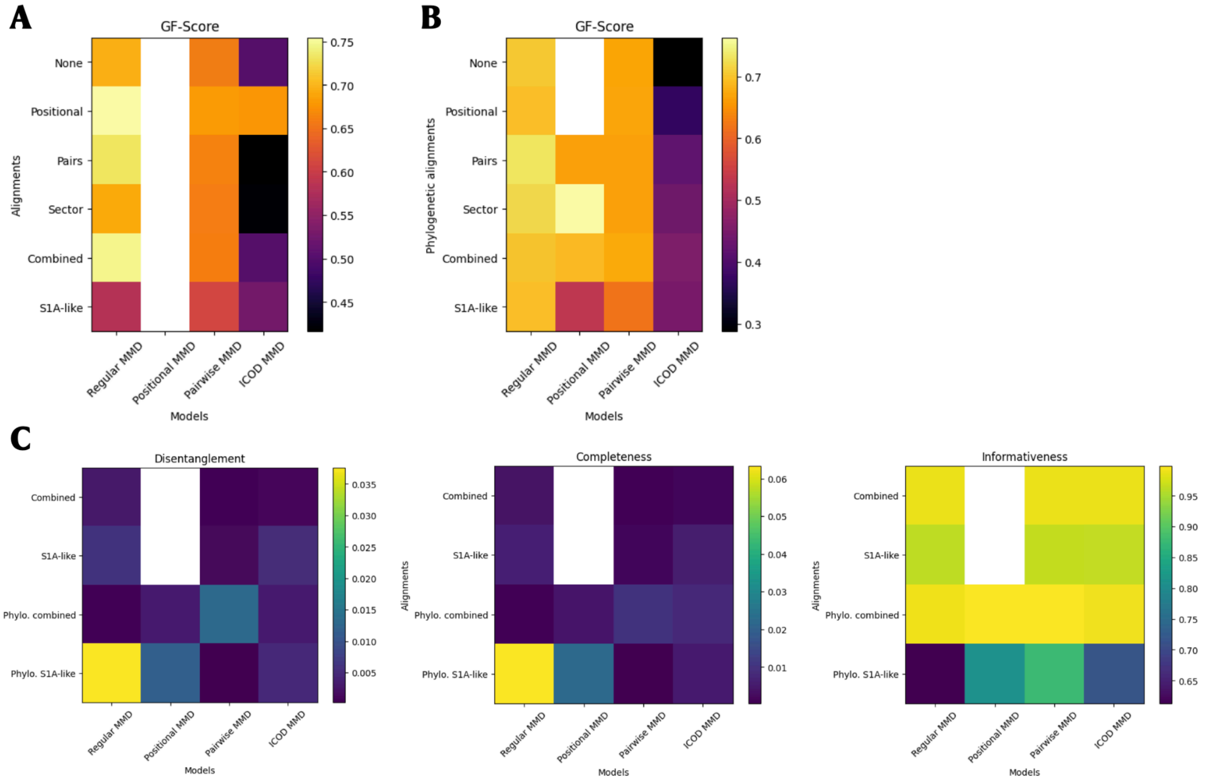

Figure 3. Grid-fitting score by model architecture, alignment, and sampling method.

A) GF-Score for each VAE model and Potts-sampled MSA, **B)** GF-Score for each VAE model and phylogenetically-sampled MSA, and **C)** Disentanglement, completeness, and informativeness scores for combined constraints and S1A-like alignments under both Potts and phylogenetic-Potts sampling. DCI score is in range [0,1] with 1 being best, while lower GF-score is better

After gathering the results for the above models and alignments, the next step was to push the toy models even closer to the real protein family by changing the alignment sampling method. While the Potts model sampling provides a distribution of toy protein sequences that are convincingly sampled from a number of conservation and correlation constraints, each toy protein in the Potts-generated MSAs is generated individually. Real proteins do not come about by such a process, but by parallel evolution from ancestor sequences. Thus, a phylogenetic simulation with a Potts model fitness function was designed to generate a convincing distribution of sampled proteins generated from a common precursor protein with evolutionary trajectories splitting at regular generational intervals (Fig. 2B).

For the phylogenetically-sampled MSAs, we see that most, but not all, of the trends observed from the non-phylogenetically-sampled MSAs persist (Fig. 3B). The ICOD model again wins out in terms of GF-Score, maintaining the trend of decreasing in performance with increasing alignment complexity. When moving to real protein data with complex alignments, it will be seen whether this advantage persists or crosses over with the other methods. The pairwise MMD model is again quite consistent in scoring better than the regular MMD model, and still improves when moving to the more complex S1A-like alignment. Interestingly, this time the regular MMD model performs better on average than in the non-phylogenetic alignments, however the phylogenetic sampling seems to have nullified its advantage on the S1A-like alignment. This time, it is the positional MMD model that comes out second to ICOD in that alignment. Yet the results of this model are somewhat inconsistent, performing poorly on the single sector alignment, despite performing well on the 3-sector S1A-like alignment.

Moving to DCI scores again, it is again apparent that all models perform quite well on informativeness with the combined constraints alignment, and much worse on the S1A-like alignment (Fig. 3C). This time there is more separation between model performance on this alignment, however, with the positional MMD and particularly the pairwise MMD model performing much better than the rest. The only model worth mentioning in disentanglement and completeness is regular MMD on the S1A-like alignment. Despite apparently outperforming the rest, it is worth noting that the ideal score for these metrics is 1.00, meaning none of the models make any significant achievement to sector disentanglement despite the simple dataset.

2.3 Pairwise Correlations Improve VAE Disentanglement on S1A Serine Proteases

From what was learned studying the VAE models on toy alignments, it was then determined to test such models on real protein data with the S1A MSA. The S1A family has been well-studied and characterized, particularly with regards to its statistical correlations and functional features, and thus evaluation of latent spaces and generative capabilities is more feasible (19,26,27). The MSA used for training contains 1444 sequences of length 223 from across the domains of life.

One additional MMD VAE model included in the S1A studies which was not in the toy model experiments used reconstruction loss modified by sequence weight was tested. Sequence weights are a way of adjusting an MSA so that correlation is not unduly influenced by large clades of very similar sequences (26). Thus, sequences are down-weighted proportional to their similarity to other similar sequences. It was therefore thought that multiplying reconstruction losses by corresponding sequence weights would mean that many near sequences would not pull the model toward learning their features significantly more than a different clade of fewer sequences. This was done for S1A models and not for toy models, as the oversampling regime of short sequences in the toy model case would have severely downweighted all of the sequences, since there were many identical repeats. In the case of the phylogenetic toy models, the bifurcation of evolutionary trajectories at regular intervals means that clade size is invariant at each time point, and thus they do not need sequence reweighting.

Lastly, I compared the altered MMD models against a few unaltered models as described in the literature. The first of these is the Disentangling Autoencoder (DAE), implemented as described by Cha and Thiyagalingam (Fig. 1C) (39). This one was tested since the code-agnostic

Grid-Fitting Score (GF-Score) disentanglement metric was taken from the same paper. It uses only reconstruction loss, but works by normalizing encodings to ensure each feature is evenly distributed in the latent space. Before decoding, it also includes an interpolation layer which smooths the latent space, and an Euler encoding to orthogonalize the latent dimensions (39). The β-TC VAE architecture was used as described in Chen et al. (Fig. 1B) (37). This model decomposes the Evidence Lower Bound (ELBO) loss, the traditional loss used in VAE models, into three components: mutual information, total correlation, and dimensional KLD. It upweights the loss penalty on total correlation to encourage the model to orthogonalize its latent dimensions while downweighting or removing penalties on the other components. These encourage the model to share information between the input and latent space and to differentiate the dimensions, respectively (37). A vanilla VAE model like the first VAE described by Kingma and Welling was also tested without separating the ELBO loss into these components (32).

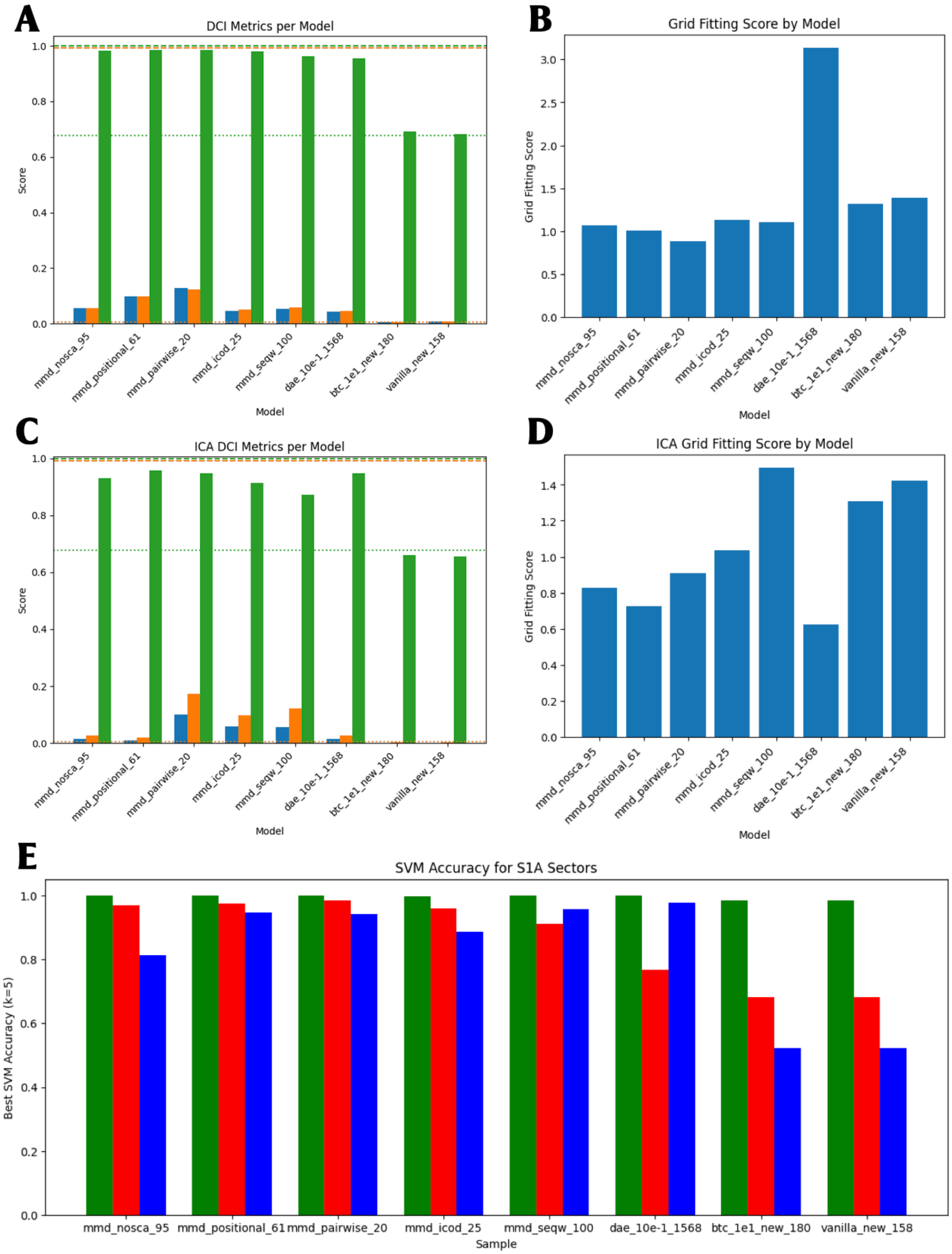

Figure 4. Latent space disentanglement metrics for S1A VAE models.

A) DCI metrics for each of the eight models described. Blue: disentanglement, orange: completeness, green: informativeness. 0 is best and 1 is worst for all scores. The positive (dashed) and negative (dotted) controls are means over all the models, with each having its factors compared against themselves or against random data, respectively. **B)** GF-Score for the same models, using a grid size of 10. A lower GF-Score is better. **C)** DCI metrics for latent spaces projected to 3 dimensions with ICA, and **D)** corresponding GF-Scores. **E)** Accuracy of best linear SVM hyperplane for separating functional categories in each model latent space. Green: catalytic/non-catalytic, red: trypsin/chymotrypsin, blue: vertebrate/invertebrate. For all graphs, models are arranged classic MMD, position-weighted MMD, pairwise-weighted MMD, ICOD-weighted MMD, sequence-weighted MMD, DAE, β-TC VAE, and vanilla VAE, with final numbers indicating optimal training epochs.

A DCI comparison used the sector values for each sequence (the U_{ICA} matrix in SCA) as ground truth factors, while latent encodings were taken as codes. After training the models on S1A data with latent dimension 10, it was found that most models perform very well on informativeness (i.e., capturing all variation in the SCA-derived functional dimensions), however perform quite poorly in disentanglement and completeness (Fig. 4A). This indicates that, overall, the functional properties are encoded in the latent space but are often spread across multiple latent dimensions, and as such cannot be accessed directly. The β-TC VAE and vanilla VAE perform quite poorly on all metrics, even informativeness. The two models which diverge from the general trend are the positional and pairwise SCA-loss MMD VAE models, which manage to achieve significantly higher, but still poor, disentanglement and completeness. Encoding pairwise correlation significance values especially improves these metrics. The better informativeness performance of these two models on the toy datasets seems to have foreshadowed their better overall performance with natural data.

The GF-Score indicates a similar story (Fig. 4B). The best score is achieved by the pairwise conservation MMD VAE, with the positional conservation MMD VAE behind it. This is followed by the three other MMD VAEs and then the β-TC and vanilla VAEs. Lastly, the DAE shows an incredibly high (poor) GF-Score, which was surprising given that the DAE was specifically designed to minimize GF-Score (39). This high of a score indicates that the data is largely clustered in one part of the latent space, with a few outliers stretching the overall span of the space. The performance of the positionally-weighted and pairwise-weighted MMD VAEs seems to have surpassed ICOD when increasing to natural data complexity, as was predicted from the toy models.

To ensure that the disentanglement was actually low and not just a result of a latent space misaligned with the underlying functional pattern of sequences, Independent Component Analysis (ICA) was performed to align the latent spaces along the top three axes of variance. The same DCI and GF-Score analysis was then performed (Fig. 3C-D), however no significantly improved model was found. This time, the pairwise SCA, ICOD, and sequence-weighted MMD VAEs were the only ones with any success in disentanglement and completeness, with pairwise SCA again beating out the rest. Informativeness was mostly preserved despite the fewer dimensions. However this time, DAE performed best in terms of GF-Score with the sequence-weighted MMD VAE performing worst and others in between. The success of the positionally-weighted MMD VAE on the toy data is also mimicked here, performing second to the DAE. These analyses seem to show that realigning the modes of maximum variance in the latent spaces does not bring out improved functional separation, except in the case of the DAE.

In a final attempt to probe disentanglement, a Support Vector Machine (SVM) was optimized with a penalty parameter sweep to find the best linear hyperplane separating each functional category in the model latent spaces (Fig. 3E). Results were validated with 5-fold cross-validation. More promising than the ICA analysis, this showed that for most of the models, the functional categories could be segregated with high fidelity using a linear hyperplane in the latent space. This was particularly true for the positional and pairwise-weighted MMD VAEs, with β-TC and vanilla VAEs performing poorly again. This linear separation is particularly important, as it means the latent space is a linear transformation away from functional disentanglement. This could greatly reduce the work required to find a proper agnostic function for disentanglement, as nonlinear methods may not need to be considered.

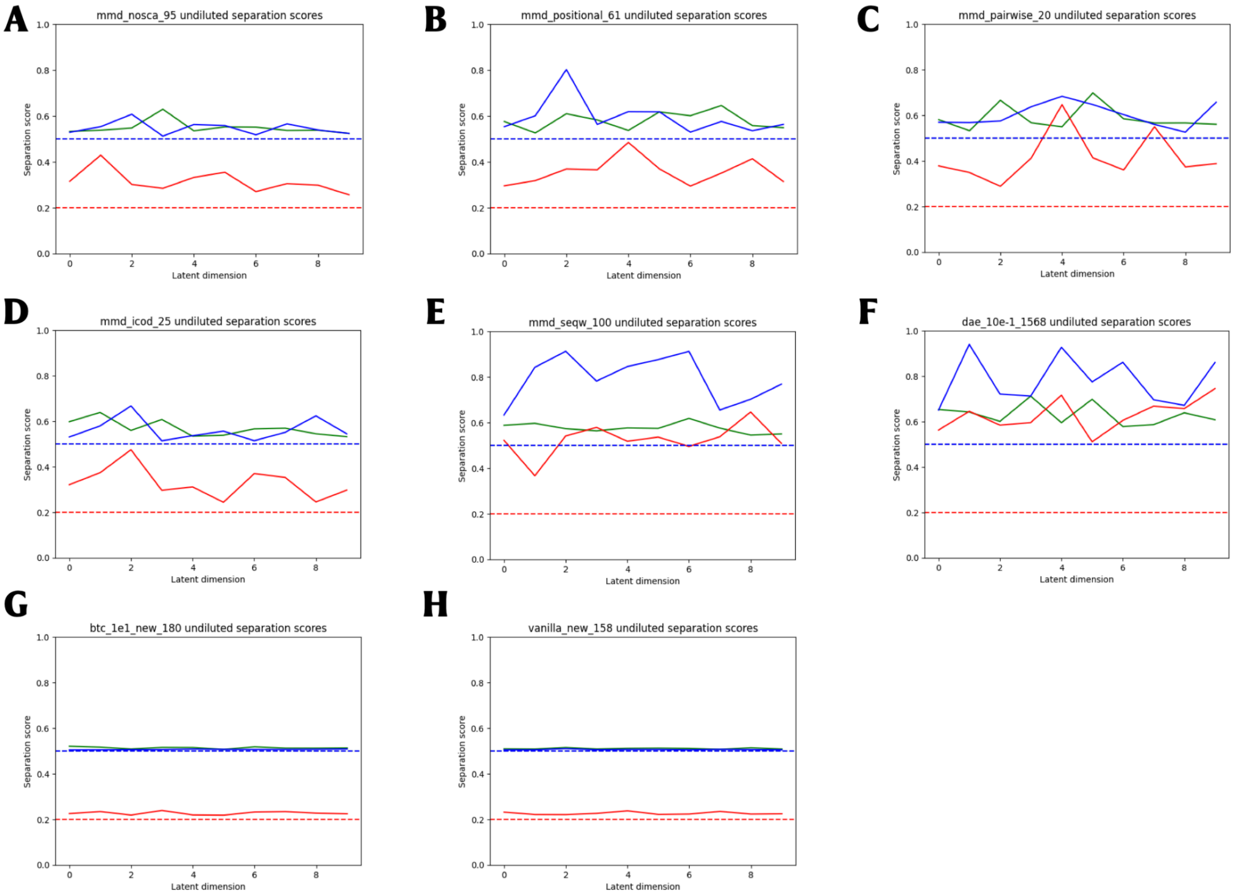

Figure 5. Dimension-wise disentanglement in S1A VAE models.

A-F) Separation scores for each model by each of the 10 latent dimensions, using the functional and phylogenetic annotations of catalytic activity or no catalytic activity (green), ligand specificity class (red; trypsin, tryptase, kallikrein, granzyme, or chymotrypsin), and vertebrate or invertebrate (blue). Expected scores (dashed; red and blue are both at 0.5) are calculated based on a uniform distribution of the sub-categories. Again, models are arranged **A)**

classic MMD, **B)** position-weighted MMD, **C)** pairwise-weighted MMD, **D)** ICOD-weighted MMD, **E)** sequence-weighted MMD, **F)** DAE, **G)** β-TC VAE, and **E)** vanilla VAE, with final numbers indicating optimal training epochs.

To more granularly understand the state of the latent space, it is desirable to observe the distribution of known functional annotations along each dimension. This led to the need for a metric which could eliminate the need to plot a distribution for every latent dimension in each model for each annotation. Thus, *separation score* was formulated as a simple, interpretable metric that works for multiple categories: for each of the chosen *n* bins dividing a latent dimension and *m* possible values the annotation can take:

$$Separation\ score\ = \sum_{i}^{m}\sum_{j}^{n} P(i|j)P(j|i)$$

This score falls between 0 and 1, where 1 is the best (most segregated) and 0 is the worst. Intuitively, this is the sum over buckets and potential annotation values of the probability of finding annotation *i* in bucket *j* multiplied by the probability of being in bucket *j* given category *i*. In comparing the dimensional separation scores for each annotation (Fig. 5), a similar progression from vanilla MMD VAE to positional then pairwise MMD VAE as in Figure 4 can be seen here as well; the dimensions generally improve in their separation of the respective annotations (Fig. 5A-C). Note that it is not necessary for every dimension to separate each annotation (factor) as well as possible; that would be impossible and actually undesirable, since orthogonal dimensions are the goal. A positive example of this can be seen in the pairwise MMD VAE model's separation scores, where peaks of separation for specificity categories (like trypsin versus chymotrypsin) fall opposite to peaks of separation for catalytic versus non-catalytic. This also appears to be somewhat the case in the DAE model, which shows impressively high separation overall (Fig. 5F). The sequence-weighted VAE also shows impressively high latent separation per dimension (Fig. 5E), though it is more evenly distributed. Perhaps this can explain why, despite these high scores, it does not perform better than the other models in DCI scores. Why DAE does not perform better on DCI metrics given this separation (or vice versa) is also a conflict which needs to be investigated further. The β-TC VAE and vanilla VAE perform especially poorly on separating the latent space; as they consistently underperformed the other models, they were dropped when going into inference and generation.

2.4 Increasing Loss Constraints Reduce Variability in Generated Serine Proteases

Now that latent disentanglement has been compared between the VAE models, it was essential to ensure that increasing disentanglement did not come at the cost of other goals of a protein generative model such as generated sequence quality or diversity. To sample sequences, the S1A MSA was passed through each of the trained models to infer the latent embeddings of each sequence. Their distribution in the latent space was then fit by a Bayesian-Gaussian mixture model, which was then sampled and decoded to produce 100 new sequences per model. Twenty of these were selected at random, and put through protein BLAST to identify top natural hit sequences for each generated protein (42). The data from each generated sequence's *top* BLAST hit was then extracted.

In particular, the local and global identity of the alignment between the natural and generated sequences was of interest. Local identity compares the percent identity of domains conserved between the two proteins, while global identity compares the percent identity over the entire alignment (Fig. 6A,C). These can also be weighted by the BLOSUM matrix to quantify whether the divergence is simply swapping to functionally analogous amino acids or not (Fig. 6B,D).

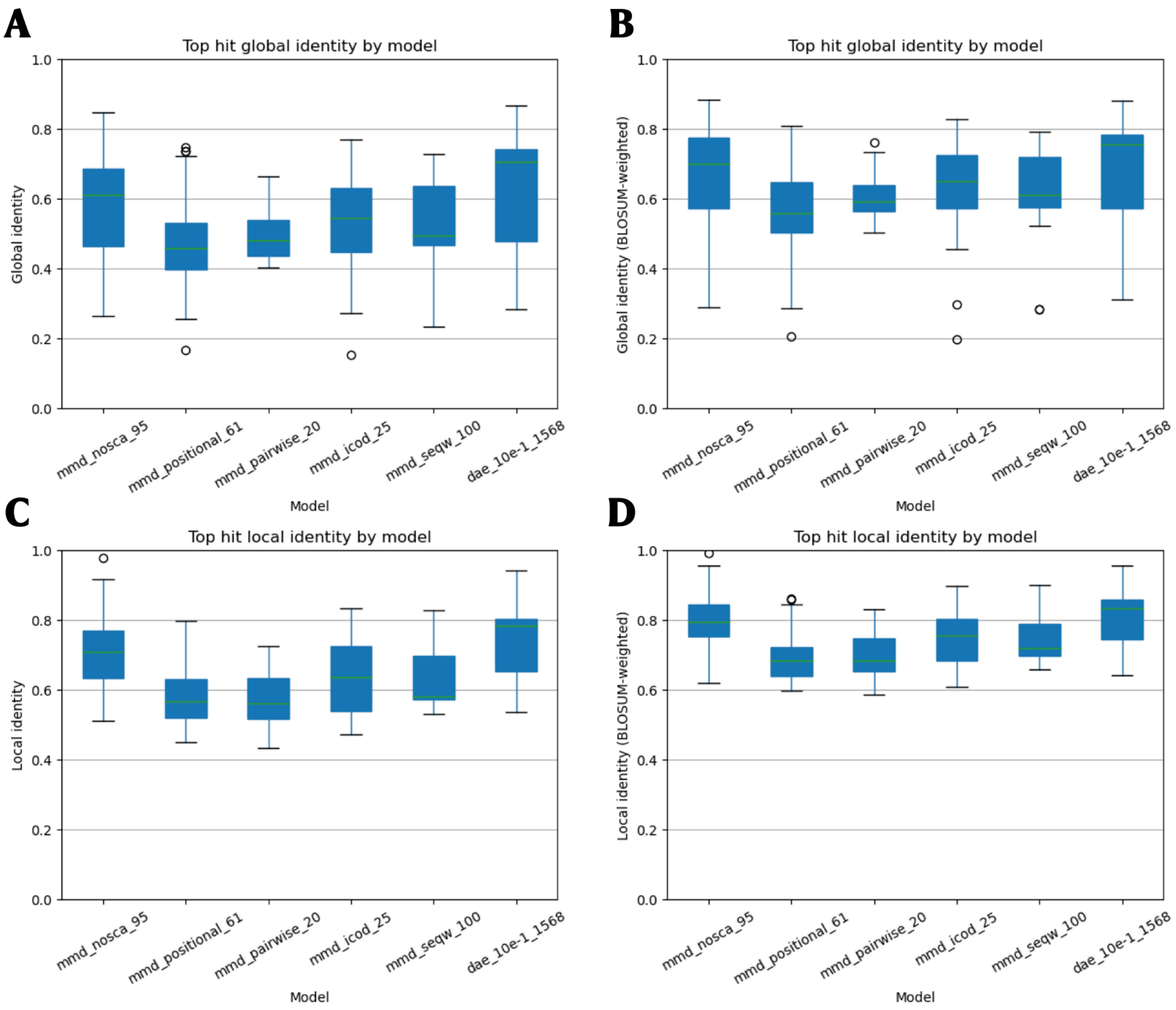

Figure 6. Identity of VAE-generated sequences to nearest BLAST hits.

A) Global identity distribution of top BLAST hits per model, **B)** BLOSUM-weighted. **C)** Local identity distribution of top BLAST hits per model, **D)** BLOSUM-weighted.

Interestingly, despite the added constraints of the positional and pairwise MMD VAE models (as well as the sequence-weighted MMD VAE) to the loss, the global and local identities of top BLAST hits actually decrease in comparison to the less-constrained vanilla MMD VAE. The ICOD MMD VAE also shows this, but to not as great an extent. Despite this diversification on average, however, increasing constraints appear to shrink the *range* of top hit identities. The pairwise MMD VAE might have a median top hit global identity approximately 10% lower than the vanilla MMD VAE, but its overall range is also less than half. The DAE model does not significantly change the distribution compared to the regular MMD model. BLOSUM-weighting does not appear to significantly alter these observed trends.

Next, the best top hit, median top hit, and lowest top hit (by BLAST bit score) from each model were taken along with their corresponding generated sequences to compare the range of generative quality with regard to structure (Fig. 7). The generated sequence structures were predicted using AlphaFold (7). The hit structures were taken from PDB or the AlphaFold database where available (all of which were structures generated by AlphaFold), but many of them were not available and had to be generated manually using AlphaFold. Some of the proteins were not well characterized, so the structures may contain terminal regions that do not naturally make it into the protein. Regardless, the generated and natural structures were aligned using PDB's Pairwise Structure Alignment Tool, with TM-Align set as the particular algorithm (43).

Regardless of model and percent identity, the structures of the generated and natural sequences are highly conserved. In particular, the secondary structure of α-helices and β-sheets match almost exactly, though this starts to break down in rare cases with the 'Furthest top hit' column. Where there is a bit more variation is the loops between secondary structures. These are naturally less constrained by hydrogen bonding compared to secondary structures and so it is unsurprising that in highly altered sequences they show more propensity to shift in comparison to natural sequences. This is especially apparent in longer loops, such as those in the Group 3 allergen-like, ICOD MMD VAE-generated sequence. It is more difficult to tell whether these shifts, or even if very minor shifts in secondary structures, are crucial to protein function. Yet structural comparison remains perhaps the most accurate *in silico* generated sequence quality metric available at the moment, and all tested models appear to perform quite well.

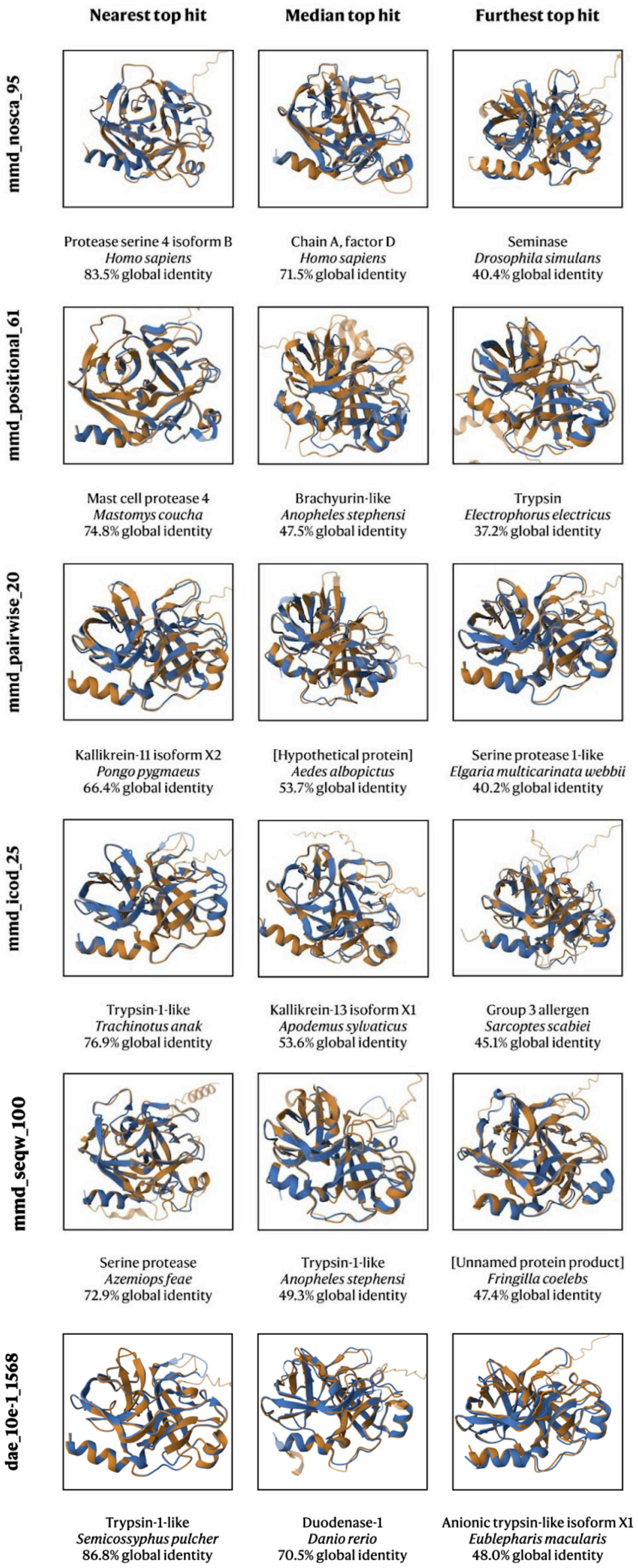

Figure 7. Structural overlay of generated sequences to nearest BLAST hits.

Blue represents the generated protein while orange represents the natural BLAST hit protein. Below each box is the protein name, organism, and percent unweighted global identity of the hit protein. Proteins are generally aligned with the family's terminal α-helix in the bottom left for better comparison between sequences.

DISCUSSION & FUTURE DIRECTIONS

In this study, it was found that adding constraints such as positional, pairwise, and ICOD weights improves disentanglement in VAE models trained on toy alignments, though with the models facing greater difficulty from increasing numbers of alignment constraints and phylogenetic sampling. These trends were largely replicated in real S1A data, with the pairwise MMD VAE model particularly outperforming the rest despite low disentanglement and completeness scores across the board. Although the high informativeness achieved by most models led to the observation that the latent space is functionally separable with high accuracy (especially by positional and pairwise constraint models), applying ICA to the latent space did not realign latent dimensions toward a more advantageous representation of the function space. Upon moving to the generative regime, adding loss constraints to VAE models appears to restrict the range of generative diversity, but improves diversity of generated sequences on average despite penalizing more for not conforming to natural correlation patterns. This potentially indicates the success of the SCA-derived weight terms in capturing more of the non-phylogenetic correlation than other models. Lastly, all models generated sequences of natural quality based on AlphaFold predictions and structure alignments, regardless of sequence identity.

There are a few main lines of questioning that arise from this study. The first is in further understanding disentanglement and model performance. As is apparent from the results, different disentanglement metrics often have different assessments of a latent space. It is a well-known issue that disentanglement metrics are often unreliable and conflicting, even with known ground-truth factors (40). The toy models used here attempted to get as close to real protein data as possible while still preserving known ground truth factors, however even these factors had to be statistically-derived. This gets at another problem that there is know formalism for protein function to measure as ground truth factors; SCA sectors provide one of the more compelling and agnostically derivable descriptions, however still do not provide a complete representation of the protein's functionality. Thus, further studies into both disentanglement metrics and defining protein function are likely needed to fully understand model performance, even for toy systems. These two problems likely require general advances in representation learning and protein physics, respectively.

Two potentially more immediate directions to approach continuation of this project are in altering different handles of model learning and learning secondary models. This study only inspected the loss function as a handle on model learning. Although it is perhaps the most obvious and easy feature to tune VAE learning, there are many other architectural adaptations that may be considered. Many successful modern models utilize VAEs as only a portion of their architectures, and a model solving the functional disentanglement problem will likely be achieved by extensive iterative experimentation with larger, more convoluted architectures as has been the case with AlphaFold and others unless more principles-based design is innovated (7). The loss function is still certainly important, and would potentially be able to solve the functional disentanglement problem if the pairwise correlation matrix could be further enhanced to filter out correlation resulting from phylogeny rather than function.

Secondary models are therefore perhaps the most enticing option for pursuing functional disentanglement. This does not necessarily mean another neural network-based model, but rather one which can reversibly rearrange the latent space into a disentangled space such that the functional characterization of a sequence may be viewed and changed before being projected back into the latent and then generated sequence space. If this study has shown one thing, it is that a functionally separated latent space already exists within the VAE latent space (as proven by the high-accuracy linear SVM fitting), just one which is not aligned with the latent dimensions. If there were a way to agnostically identify these functional modes without having to resort to annotation, then this problem would be essentially solved. Although ICA did not work in this case, there are other avenues by which to approach this question that are worth pursuing.

METHODS

Toy Data Generation

Toy model data sets were generated using the UndersamplingCoevolution Matlab repository with the field and coupling patterns as described in section 2.1 (21). Additional parameters are as shown in Table 1.

Parameter	**Value**	**Parameter**	**Value**	**Phylogenetic parameter**	**Value**
N	10000	L	6 (9 for S1A-like)	b	13
delta_t	200000	J0	2	m	5
q	4	h0	2	initial_evolution_gens	500

Table 1. Toy dataset generation parameters.

For the phylogenetically-sampled toy datasets, the method described in Qin and Colwell was used (13). This involves generating a random sequence, evolving it according to the Potts Hamiltonian constraints using the same Metropolis-Hastings process as in the non-phylogenetic case for a certain number of generations (initial_evolution_gens, Table 1). However, instead of repeating this process independently for *N* sequences, the sequence is duplicated and the evolution run independently on both sequences for *m* proposed mutations. This is repeated for *b* diverging generations, in this case producing 8192 sequences to best match up with the 10000 of the non-phylogenetic alignment.

S1A Alignment Data

The S1A MSA used to train the models is 1444 sequences of length 223. This includes gap characters, making a one-hot encoding tensor of dimension (1444, 223, 21). The sequences included in the alignment come from all kingdoms of life. The alignment of 1470 sequences was produced by Halabi et al. in 2009, but was slightly narrowed down based on available functional annotations (27).

pySCA

SCA was performed on the toy and S1A alignments using the pySCA package (26), which was modified to handle differing numbers of amino acids. In particular, the seqw (sequence weights), Csca ($\tilde{C}_{ij}$), Wia (ϕ_i^a), and Upica (sequences in sector coordinate space) output vectors and matrices were utilized.

Model Design

There were three VAE model architectures which were employed; these are MMD VAE, β-TC VAE, and DAE. The official code for the MMD VAE as described in the paper by Zhao, Song, and Ermon inspired the version I started work on, which was written by Nikša Praljak using the PyTorch library for Python instead of TensorFlow which was used in the original code (33). This baseline model without alteration (other than to learn on protein data) was termed 'mmd_nosca' as the other MMD models would receive mostly SCA-based losses. Further models were generated by altering this original model with different loss functions. The first of these, 'mmd_positional,' multiplied the cross-entropy reconstruction loss positionally with the flattened Wia matrix from pySCA. 'mmd_pairwise' was achieved by taking the outer product of the reconstruction loss vector, taking the Hadamard product with the Csca matrix from pySCA, and then taking the norm across positions to yield the loss vector. ‘mmd_icod’ used the same process but rather than the Csca matrix used the inverse covariance matrix of the alignment with diagonal set to 0 as described by Wang et al. (23). ‘mmd_seqw’ simply multiplied the overall reconstruction loss *per sequence* (across the batch) by the sequence weights from pySCA.

The DAE model was implemented as described by Cha and Thiyagalingam, using the Orthogonality-Enforced Latent Space in Autoencoders GitHub repository (39). The β-TC VAE was implemented as described in Chen et al., using the beta-tcvae GitHub repository (37). This code was modified to handle protein data rather than image data (without convolution), and with a softmax layer at the end of the decoder to turn the amino acid dimension into a probability distribution. The Vanilla VAE model used the β-TC architecture with a β value of 1 and without the --tcvae flag. All encoder and decoder models were given four layers with a hidden dimension 1.5x the input data dimension for more precise comparison.

Training & Hyperparameter Tuning

All S1A models were trained first at 400 epochs and a train-test split such that the optimal epoch for validation loss could be located, after which they were re-trained at that optimal epoch (Table 2). The DAE model did not hit a validation loss minimum in the first 400 epochs, so it was trained for longer. Toy models were more numerous so were trained at 100 epochs each, though this was validated to be near the minimum validation loss epoch for all tested models. S1A models were trained with ten latent dimensions, while toy models were trained with three latent dimensions. Learning rate was 1e-4, batch size 128, and dropout 0.1 for all models. β and α hyperparameters for β-TC VAE, Vanilla VAE, and DAE were optimized alongside epoch for S1A models (Table 3).

Model	**Best S1A Epoch**	**Hyperparameter**	**Range**	**Optimum**
Regular MMD	95	N/A		
MMD positional	61			
MMD pairwise	20			
MMD ICOD	25			
MMD sequence-weighted	100			
DAE	1568	α	1e-3 → 1e-1	0.1
β-TC	180	β	1e1 → 1e3	10
Vanilla	158	β		1

Table 2. Model hyperparameters.

Inference & Latent Sampling

Inference on the toy and S1A models was performed by running alignment data through the trained model and extracting the latent encodings. For the DAE model, latent encodings were taken after the normalization step. Latent sampling was performed using a Bayesian Gaussian mixture model with parameters in Table 3, then decoded using the trained model decoders (for DAE, sampled embeddings were first put through interpolation and Euler encoding).

Parameter	**Value**	**Parameter**	**Value**
n_samples	100	weight_concentration_prior_type	dirichlet_distribution
n_components	*3 times the number of latent dimensions*	weight_concentration_prior	1e-5
covariance_type	tied	tol	0.001
init_params	k-means++	max_iter	200

Table 3. Parameters for model latent sampling.

Disentanglement Metrics

DCI score was implemented using the disentanglement_lib repository by Locatello et al. (44). The pySCA Upica matrix was used as the factor matrix, while inferred latent embeddings were used as the codes. Continuous factors and a random forest model were used. Positive controls used Upica as both factors and codes, while negative controls used Upica as the factors and random data as the controls. GF-Score was implemented as described by Cha and Thiyagalingam, using a grid size of 10 for all models (39). Separation score was calculated separately for each functional categorization and latent dimension, with twenty bins per dimension. Each sub-categorization (e.g. 'chymotrypsin' under 'specificity' functional categorization) has a bin score of the probability of finding that sub-categorization in the bin multiplied by the probability of being in that bin given the sub-categorization, summed over the bins. The median value for sub-categorizations was taken as the dimensional result for that category. ICA was performed using FastICA from sklearn with three components (mirroring the three functional sectors). SVM was performed using SVC from sklearn with C hyperparameter optimization from 10^{-3} to 10^{2} and a linear kernel.

BLAST Identification of Natural Neighbors

Twenty sequences from latent sampling were entered into NCBI pBLAST (42). The bit score, local and global identity, and BLOSUM-weighted local and global identity were collected for each of the twenty top hits. For each model, the highest, median, and lowest hit score of the twenty top hits were collected, along with the generated sequence and corresponding natural sequence.

Comparison of Predicted and Natural Structures

The generated sequences were run through AlphaFold colabfold on default settings to generate structural files (7). The corresponding natural sequences as fetched from NCBI RefSeq were also run through AlphaFold colabfold as most structures were not available, with the exception of protease serine 4 isoform B [Homo sapiens], seminase [Drosophila simulans], and serine protease [Azemiops feae] from the AlphaFold Protein Structure Database, and Chain A, FACTOR D [Homo sapiens] from Protein DataBank (PDB) (45,46,47). Alignment was performed using the PDB Pairwise Structure Alignment tool with the TM-Align algorithm (43,46).

REFERENCES

(1) Morris, R.; Black, K. A.; Stollar, E. J. Uncovering Protein Function: From Classification to Complexes. Essays in Biochemistry 2022, 66 (3), 255–285. https://doi.org/10.1042/EBC20200108.

(2) Feldman, D. E.; Frydman, J. Protein Folding in Vivo: The Importance of Molecular Chaperones. Current Opinion in Structural Biology 2000, 10 (1), 26–33. https://doi.org/10.1016/S0959-440X(99)00044-5.

(3) Rebeaud, M. E.; Mallik, S.; Goloubinoff, P.; Tawfik, D. S. On the Evolution of Chaperones and Cochaperones and the Expansion of Proteomes across the Tree of Life. Proc. Natl. Acad. Sci. U.S.A. 2021, 118 (21), e2020885118. https://doi.org/10.1073/pnas.2020885118.

(4) Rouviere, E.; Ranganathan, R.; Rivoire, O. Emergence of Single- versus Multi-State Allostery. PRX Life 2023, 1 (2), 023004. https://doi.org/10.1103/PRXLife.1.023004.

(5) Blow, D. M. Structure and Mechanism of Chymotrypsin. Acc. Chem. Res. 1976, 9 (4), 145–152. https://doi.org/10.1021/ar50100a004.

(6) Sachdeva, V.; Husain, K.; Sheng, J.; Wang, S.; Murugan, A. Tuning Environmental Timescales to Evolve and Maintain Generalists. Proc. Natl. Acad. Sci. U.S.A. 2020, 117 (23), 12693–12699. https://doi.org/10.1073/pnas.1914586117.

(7) Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Žídek, A.; Potapenko, A.; Bridgland, A.; Meyer, C.; Kohl, S. A. A.; Ballard, A. J.; Cowie, A.; Romera-Paredes, B.; Nikolov, S.; Jain, R.; Adler, J.; Back, T.; Petersen, S.; Reiman, D.; Clancy, E.; Zielinski, M.; Steinegger, M.;

Pacholska, M.; Berghammer, T.; Bodenstein, S.; Silver, D.; Vinyals, O.; Senior, A. W.; Kavukcuoglu, K.; Kohli, P.; Hassabis, D. Highly Accurate Protein Structure Prediction with AlphaFold. Nature 2021, 596 (7873), 583–589. https://doi.org/10.1038/s41586-021-03819-2.

(8) Brotzakis, Z. F.; Zhang, S.; Murtada, M. H.; Vendruscolo, M. AlphaFold Prediction of Structural Ensembles of Disordered Proteins. Nat Commun 2025, 16 (1), 1632. https://doi.org/10.1038/s41467-025-56572-9.

(9) De Crécy-lagard, V.; Amorin De Hegedus, R.; Arighi, C.; Babor, J.; Bateman, A.; Blaby, I.; Blaby-Haas, C.; Bridge, A. J.; Burley, S. K.; Cleveland, S.; Colwell, L. J.; Conesa, A.; Dallago, C.; Danchin, A.; De Waard, A.; Deutschbauer, A.; Dias, R.; Ding, Y.; Fang, G.; Friedberg, I.; Gerlt, J.; Goldford, J.; Gorelik, M.; Gyori, B. M.; Henry, C.; Hutinet, G.; Jaroch, M.; Karp, P. D.; Kondratova, L.; Lu, Z.; Marchler-Bauer, A.; Martin, M.-J.; McWhite, C.; Moghe, G. D.; Monaghan, P.; Morgat, A.; Mungall, C. J.; Natale, D. A.; Nelson, W. C.; O'Donoghue, S.; Orengo, C.; O'Toole, K. H.; Radivojac, P.; Reed, C.; Roberts, R. J.; Rodionov, D.; Rodionova, I. A.; Rudolf, J. D.; Saleh, L.; Sheynkman, G.; Thibaud-Nissen, F.; Thomas, P. D.; Uetz, P.; Vallenet, D.; Carter, E. W.; Weigele, P. R.; Wood, V.; Wood-Charlson, E. M.; Xu, J. A Roadmap for the Functional Annotation of Protein Families: A Community Perspective. Database 2022, 2022, baac062. https://doi.org/10.1093/database/baac062.

(10) Watson, J. L.; Juergens, D.; Bennett, N. R.; Trippe, B. L.; Yim, J.; Eisenach, H. E.; Ahern, W.; Borst, A. J.; Ragotte, R. J.; Milles, L. F.; Wicky, B. I. M.; Hanikel, N.; Pellock, S. J.; Courbet, A.; Sheffler, W.; Wang, J.; Venkatesh, P.; Sappington, I.; Torres, S. V.; Lauko, A.; De Bortoli, V.; Mathieu, E.; Ovchinnikov, S.; Barzilay, R.; Jaakkola, T.

S.; DiMaio, F.; Baek, M.; Baker, D. De Novo Design of Protein Structure and Function with RFdiffusion. Nature 2023, 620 (7976), 1089–1100. https://doi.org/10.1038/s41586-023-06415-8.

(11) Praljak, N.; Yeh, H.; Moore, M.; Socolich, M.; Ranganathan, R.; Ferguson, A. L. Natural Language Prompts Guide the Design of Novel Functional Protein Sequences. Synthetic Biology November 11, 2024. https://doi.org/10.1101/2024.11.11.622734.

(12) Kingma, D. P.; Welling, M. An Introduction to Variational Autoencoders. 2019. https://doi.org/10.48550/ARXIV.1906.02691.

(13) Qin, C.; Colwell, L. J. Power Law Tails in Phylogenetic Systems. Proc. Natl. Acad. Sci. U.S.A. 2018, 115 (4), 690–695. https://doi.org/10.1073/pnas.1711913115.

(14) Lu, Y.; Lu, J. A Universal Approximation Theorem of Deep Neural Networks for Expressing Probability Distributions. arXiv 2020. https://doi.org/10.48550/ARXIV.2004.08867.

(15) Poelwijk, F. J.; Krishna, V.; Ranganathan, R. The Context-Dependence of Mutations: A Linkage of Formalisms. PLoS Comput Biol 2016, 12 (6), e1004771. https://doi.org/10.1371/journal.pcbi.1004771.

(16) Poelwijk, F. J.; Socolich, M.; Ranganathan, R. Learning the Pattern of Epistasis Linking Genotype and Phenotype in a Protein. Nat Commun 2019, 10 (1), 4213. https://doi.org/10.1038/s41467-019-12130-8.

(17) Dryden, D. T. F.; Thomson, A. R.; White, J. H. How Much of Protein Sequence Space Has Been Explored by Life on Earth? J. R. Soc. Interface. 2008, 5 (25), 953–956. https://doi.org/10.1098/rsif.2008.0085.

(18) Morcos, F.; Pagnani, A.; Lunt, B.; Bertolino, A.; Marks, D. S.; Sander, C.; Zecchina, R.; Onuchic, J. N.; Hwa, T.; Weigt, M. Direct-Coupling Analysis of Residue Coevolution Captures Native Contacts across Many Protein Families. Proc. Natl. Acad. Sci. U.S.A. 2011, 108 (49). https://doi.org/10.1073/pnas.1111471108.

(19) Süel, G. M.; Lockless, S. W.; Wall, M. A.; Ranganathan, R. Evolutionarily Conserved Networks of Residues Mediate Allosteric Communication in Proteins. Nat Struct Biol 2003, 10 (1), 59–69. https://doi.org/10.1038/nsb881.

(20) Weigt, M.; White, R. A.; Szurmant, H.; Hoch, J. A.; Hwa, T. Identification of Direct Residue Contacts in Protein–Protein Interaction by Message Passing. Proc. Natl. Acad. Sci. U.S.A. 2009, 106 (1), 67–72. https://doi.org/10.1073/pnas.0805923106.

(21) Kleeorin, Y.; Russ, W. P.; Rivoire, O.; Ranganathan, R. Undersampling and the Inference of Coevolution in Proteins. Cell Systems 2023, 14 (3), 210-219.e7. https://doi.org/10.1016/j.cels.2022.12.013.

(22) Hedstrom, L.; Szilagyi, L.; Rutter, W. J. Converting Trypsin to Chymotrypsin: The Role of Surface Loops. Science 1992, 255 (5049), 1249–1253. https://doi.org/10.1126/science.1546324.

(23) Wang, S.-W.; Bitbol, A.-F.; Wingreen, N. S. Revealing Evolutionary Constraints on Proteins through Sequence Analysis. PLoS Comput Biol 2019, 15 (4), e1007010. https://doi.org/10.1371/journal.pcbi.1007010.

(24) Dietler, N.; Abbara, A.; Choudhury, S.; Bitbol, A.-F. Impact of Phylogeny on the Inference of Functional Sectors from Protein Sequence Data. PLoS Comput Biol 2024, 20 (9), e1012091. https://doi.org/10.1371/journal.pcbi.1012091.

(25) Lockless, S. W.; Ranganathan, R. Evolutionarily Conserved Pathways of Energetic Connectivity in Protein Families. Science 1999, 286 (5438), 295–299. https://doi.org/10.1126/science.286.5438.295.

(26) Rivoire, O.; Reynolds, K. A.; Ranganathan, R. Evolution-Based Functional Decomposition of Proteins. PLoS Comput Biol 2016, 12 (6), e1004817. https://doi.org/10.1371/journal.pcbi.1004817.

(27) Halabi, N.; Rivoire, O.; Leibler, S.; Ranganathan, R. Protein Sectors: Evolutionary Units of Three-Dimensional Structure. *Cell* 2009, 138 (4), 774–786. https://doi.org/10.1016/j.cell.2009.07.038.

(28) Kalinska, M.; Meyer-Hoffert, U.; Kantyka, T.; Potempa, J. Kallikreins – The Melting Pot of Activity and Function. Biochimie 2016, 122, 270–282. https://doi.org/10.1016/j.biochi.2015.09.023.

(29) Trapani, J. A. Granzymes: A Family of Lymphocyte Granule Serine Proteases. Genome Biol 2001, 2 (12), reviews3014.1. https://doi.org/10.1186/gb-2001-2-12-reviews3014.

(30) Levy, R. M.; Haldane, A.; Flynn, W. F. Potts Hamiltonian Models of Protein Co-Variation, Free Energy Landscapes, and Evolutionary Fitness. Current Opinion in Structural Biology 2017, 43, 55–62. https://doi.org/10.1016/j.sbi.2016.11.004.

(31) Russ, W. P.; Figliuzzi, M.; Stocker, C.; Barrat-Charlaix, P.; Socolich, M.; Kast, P.; Hilvert, D.; Monasson, R.; Cocco, S.; Weigt, M.; Ranganathan, R. An Evolution-Based Model for Designing Chorismate Mutase Enzymes. Science 2020, 369 (6502), 440–445. https://doi.org/10.1126/science.aba3304.

(32) Kingma, D. P.; Welling, M. Auto-Encoding Variational Bayes. arXiv 2013. https://doi.org/10.48550/ARXIV.1312.6114.

(33) Zhao, S.; Song, J.; Ermon, S. InfoVAE: Information Maximizing Variational Autoencoders. arXiv 2017. https://doi.org/10.48550/ARXIV.1706.02262.

(34) Praljak, N.; Lian, X.; Ranganathan, R.; Ferguson, A. L. ProtWave-VAE: Integrating Autoregressive Sampling with Latent-Based Inference for Data-Driven Protein Design. ACS Synth. Biol. 2023, 12 (12), 3544–3561. https://doi.org/10.1021/acssynbio.3c00261.

(35) Higgins, I.; Amos, D.; Pfau, D.; Racaniere, S.; Matthey, L.; Rezende, D.; Lerchner, A. Towards a Definition of Disentangled Representations. arXiv 2018. https://doi.org/10.48550/ARXIV.1812.02230.

(36) Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; Lerchner, A. Beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In International Conference on Learning Representations; 2017.

(37) Chen, R. T. Q.; Li, X.; Grosse, R.; Duvenaud, D. Isolating Sources of Disentanglement in Variational Autoencoders. arXiv 2018. https://doi.org/10.48550/ARXIV.1802.04942.

(38) Cha, J.; Thiyagalingam, J. Disentangling Autoencoders (DAE). arXiv 2022. https://doi.org/10.48550/ARXIV.2202.09926.

(39) Cha, J.; Thiyagalingam, J. Orthogonality-Enforced Latent Space in Autoencoders: An Approach to Learning Disentangled Representations. In Proceedings of the 40th International Conference on Machine Learning; Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., Scarlett, J., Eds.; Proceedings of Machine Learning Research; PMLR, 2023; Vol. 202, pp 3913–3948.

(40) Carbonneau, M.-A.; Zaidi, J.; Boilard, J.; Gagnon, G. Measuring Disentanglement: A Review of Metrics. arXiv 2020. https://doi.org/10.48550/ARXIV.2012.09276.

(41) Eastwood, C.; Williams, C. K. I. A Framework for the Quantitative Evaluation of Disentangled Representations. In International Conference on Learning Representations; 2018.

(42) Altschul, S. F.; Gish, W.; Miller, W.; Myers, E. W.; Lipman, D. J. Basic Local Alignment Search Tool. Journal of Molecular Biology 1990, 215 (3), 403–410. https://doi.org/10.1016/S0022-2836(05)80360-2.

(43) Zhang, Y. TM-Align: A Protein Structure Alignment Algorithm Based on the TM-Score. Nucleic Acids Research 2005, 33 (7), 2302–2309. https://doi.org/10.1093/nar/gki524.

(44) Locatello, F.; Bauer, S.; Lucic, M.; Rätsch, G.; Gelly, S.; Schölkopf, B.; Bachem, O. Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations. 2018. https://doi.org/10.48550/ARXIV.1811.12359.

(45) O'Leary, N. A.; Wright, M. W.; Brister, J. R.; Ciufo, S.; Haddad, D.; McVeigh, R.; Rajput, B.; Robbertse, B.; Smith-White, B.; Ako-Adjei, D.; Astashyn, A.; Badretdin, A.; Bao, Y.; Blinkova, O.; Brover, V.; Chetvernin, V.; Choi, J.; Cox, E.; Ermolaeva, O.; Farrell, C. M.; Goldfarb, T.; Gupta, T.; Haft, D.; Hatcher, E.; Hlavina, W.; Joardar, V. S.; Kodali, V. K.; Li, W.; Maglott, D.; Masterson, P.; McGarvey, K. M.; Murphy, M. R.; O'Neill, K.; Pujar, S.; Rangwala, S. H.; Rausch, D.; Riddick, L. D.; Schoch, C.; Shkeda, A.; Storz, S. S.; Sun, H.; Thibaud-Nissen, F.; Tolstoy, I.; Tully, R. E.; Vatsan, A. R.; Wallin, C.; Webb, D.; Wu, W.; Landrum, M. J.; Kimchi, A.; Tatusova, T.; DiCuccio, M.; Kitts, P.; Murphy, T. D.; Pruitt, K. D. Reference Sequence (RefSeq) Database at NCBI:

Current Status, Taxonomic Expansion, and Functional Annotation. Nucleic Acids Res 2016, 44 (D1), D733–D745. https://doi.org/10.1093/nar/gkv1189.

(46) Berman, H.; Henrick, K.; Nakamura, H. Announcing the Worldwide Protein Data Bank. Nat Struct Mol Biol 2003, 10 (12), 980–980. https://doi.org/10.1038/nsb1203-980.

(47) Tunyasuvunakool, K.; Adler, J.; Wu, Z.; Green, T.; Zielinski, M.; Žídek, A.; Bridgland, A.; Cowie, A.; Meyer, C.; Laydon, A.; Velankar, S.; Kleywegt, G. J.; Bateman, A.; Evans, R.; Pritzel, A.; Figurnov, M.; Ronneberger, O.; Bates, R.; Kohl, S. A. A.; Potapenko, A.; Ballard, A. J.; Romera-Paredes, B.; Nikolov, S.; Jain, R.; Clancy, E.; Reiman, D.; Petersen, S.; Senior, A. W.; Kavukcuoglu, K.; Birney, E.; Kohli, P.; Jumper, J.; Hassabis, D. Highly Accurate Protein Structure Prediction for the Human Proteome. Nature 2021, 596 (7873), 590–596. https://doi.org/10.1038/s41586-021-03828-1.