THE JOURNAL OF PHYSICAL CHEMISTRY B
A JOURNAL OF THE AMERICAN CHEMICAL SOCIETY

pubs.acs.org/JPCB

Article

The Role of Configurational Entropy in Miniprotein Stability

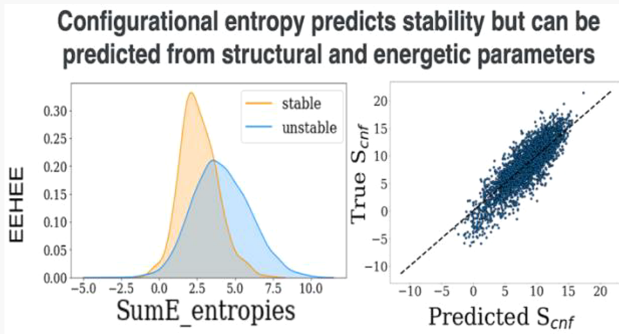
Jan D. Estrada Pabón, Hugh K. Haddox, Greg Van Aken, Ian M. Pendleton, Hamed Eramian, Jedediah M. Singer, and Joshua Schrier*

Cite This: https://doi.org/10.1021/acs.jpcb.0c09888

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: Predicting protein stability is a challenge due to the many competing thermodynamic effects. Through *de novo* protein design, one begins with a target structure and searches for a sequence that will fold into it. Previous work by Rocklin et al. introduced a data set of more than 16,000 miniproteins spanning four structural topologies with information on stability. These structures were characterized with a set of 46 structural descriptors, with no explicit inclusion of configurational entropy (S_{cnf}). Our work focused on creating a set of 17 descriptors intended to capture variations in S_{cnf} and its comparison to an extended set of 113 structural and energy model features that extend the Rocklin et al. feature set (R). The S_{cnf} descriptors statistically discriminate between stable and unstable distributions within topologies and best describe EEHEE topology stability (where E = β sheet and H = α helix). Between 50 and 80% of the variation in each S_{cnf} descriptor is described by linear combinations of R features. Despite containing useful information about minipeptide stability, providing S_{cnf} features as inputs to machine learning models does not improve overall performance when predicting protein stability, as the R features sufficiently capture the implicit variations.



■ INTRODUCTION

Protein folding is a grand challenge of biophysical chemistry. A protein's three-dimensional structure is determined by a balance of thermodynamic and kinetic effects, arising from noncovalent interactions among the residues and with the solvent, resulting in many possible local minima for a given amino acid sequence.[1,2] Predicting the ground state of an amino acid sequence requires the exploration of the many possible local minima. A naive search approach would grow exponentially with chain length, but natural proteins take advantage of funnel-shaped potential energy surfaces[3] and conserved nonlocal weak contacts[4] to accelerate the folding process. However, even with advances in computational methodologies and hardware, the direct simulation of protein folding remains a challenge.[5] Rather than proposing one or more sequences that may result in a desired structure, an alternative strategy of *de novo* protein design begins with a target structure and attempts to determine sequences that will produce it.[6] In addition to fundamental insight, *de novo* protein design has applications in the development of therapeutic protein−protein inhibitors[6] and novel biological nanomaterials.[7]

De novo miniproteins (of approximately 42 amino acids in length) provide a unique benchmark for protein design capability, as very few stable miniproteins are known in natural biological systems, yet their small size makes them tractable for computational analysis. Recently, Rocklin et al. described a high-throughput oligonucleotide library synthesis and protease-based stability assay that enabled the evaluation of 16,159 miniproteins, spanning 4 secondary-structure topologies, which expanded the number of known stable miniproteins by 2717 examples.[8] These miniproteins were designed without metal-ion binding sites or disulfide bonds, so that stability is an intrinsic property of the amino acid sequence, independent of external cofactors or cellular processing environment. Each protein is present in folded and unfolded forms with a ratio determined by the free energy of folding. The assay measures the stability of a protein based on the susceptibility of its ensemble to proteases; proteins with a higher fraction of its ensemble in the unfolded state are expected to be more susceptible to protease degradation. A valuable aspect of this data set is that it includes a complete record of both *stable* and *unstable* proteins. This inclusion of both "successes" and "failures" to produce stable proteins is important for the design of predictive scientific machine learning models.[9] Rocklin et al. used a logistic regression model to predict stability using 46 structural descriptors. Models trained and tested within a topology captured some contributions to stability (R^2 = 0.63−0.85 for predicted

Received: November 2, 2020
Revised: January 18, 2021

stability), but as depicted in Figure S1, this model primarily described stability differences between topologies, rather than intra-topology variations.

Protein folding involves a complex set of enthalpic and entropic contributions. The Rosetta force field[1] models many of the enthalpic ones, such as van der Waals interactions and inter-residue hydrogen bonding. It also indirectly models some of the entropic ones, such as the hydrophobic effect, through a term that quantifies the effects of desolvating protein atoms. However, it is unclear whether Rosetta captures other important entropic contributions. For instance, one of the major forces in protein folding is the loss in the configurational entropy (S_{cnf}) of the protein chain upon folding. Rosetta may partially capture this loss through reference energies—single constants for each of the 20 amino acids that are intended to help quantify unfolded-state free energies.[1] However, calculating the S_{cnf} of a protein is computationally challenging, as it depends on atom coordinate fluctuations and their subsequent correlations. One way around this challenge is by estimating S_{cnf} using a neural network trained to predict the S_{cnf} determined by molecular dynamics simulations.[10] The pretrained neural network can be used as a shortcut for estimating the configurational entropy of a protein, provided a structure (which is available to us as this is a *de novo* design problem) and a sequence (also available as the candidate sequence). Configurational entropy has been used as a consideration in designing thermostable mutations of adenylate kinase homologues.[11,12]

In this paper, we describe the creation of a set of 17 protein descriptors based on the Popcoen configurational entropy package by Goethe et al.[10] We then test for significant differences in the distributions of these entropy descriptors when comparing stable versus unstable proteins. We also discuss the impact of the newly generated descriptors on machine learning models trained to predict protein stability and evaluate how much of their variation arises independent of the extended Rocklin descriptor set, which includes a larger superset of structure and energy metrics described in Rocklin et al. Finally, a thorough comparison of the mutual information between the Rocklin-extended (R) and Popcoen entropy terms is provided in order to evaluate how much information is gained by the R set from the new descriptors. In addition to addressing the specific question of protein stability prediction, this work is also intended as a case study in how the role of novel physical contributions to energy models can be evaluated by using a combination of experimental data and machine learning methods.

■ COMPUTATIONAL METHODS

Minipeptide sequences, computed structures (in PDB format), Rocklin-extended (R) descriptors, and experimental stabilities reported in Rocklin et al.[8] are publicly available at https://github.com/jandestrada/Scnf_Publication; a detailed manifest is provided in the Supporting Information. The R descriptors are partially derived from the 3.10 version of the Rosetta force field which includes the Rosetta full-atom and Talaris2013 energy functions,[8] as well as structural and sequence metrics. The 113 features are described in the section on "Definition of scoring metrics" (pp 58−62) of the Supporting Information of ref 8. A 46-feature subset was selected for constructing the logistic regression models used in ref 8; for that reason, we refer to the full 113-feature set used here as "extended". The data set comprises 16,159 tested protein designs of four different secondary-structure groupings (denoted "topologies"). Secondary structure was obtained from DSSP[13] information, which represents residues in alpha helices as "H", in β sheets as "E", and in loops as "L". A topology is constructed from a repetition of these secondary structures, with loops implicit between letters. For instance, EEHEE represents a topology of two β sheets, followed by an α helix, followed by two β sheets, each connected by loops.

The stability score was calculated using the approach in Singer et al.,[22] a refinement of the Rocklin et al. approach. In brief, fluorescently tagged proteins are expressed on the surface of yeast cells. They are challenged with increasing concentrations of protease, sorted by fluorescence, and sequenced. This yields an empirical resistance to degradation by protease, expressed as EC_{50}: how much protease is necessary to degrade half of the proteins in a given time. This is a function both of a protein's likelihood of being folded (i.e., its stability) and its inherent sequence-specific resistance to the protease when unfolded. The latter is predicted by an "unfolded-state model", a computational model trained to predict EC_{50} as a function of local amino acid motifs for proteins in an unfolded state. The difference between the empirical EC_{50} and the unfolded-state model's prediction is the stability score. This value is on a $\log_{10}$ scale and can be used by machine learning models as a label for regression or, by thresholding it, as a label for classification. A stability score of 0 indicates a protein whose structure confers no additional resistance to proteolysis beyond that predicted by its amino acid sequence. A stability score of 1 indicates that the protein's resistance to the action of the protease is 10 times greater than that predicted by the susceptibility of its amino acid sequence. By convention, when assigning a binary class label, a stability score greater than 1 is "stable". Between 19.6 and 45.5% of tested sequences in each topology are stable (Figure S2).

Configurational entropies (S_{cnf}) were computed using Popcoen version 1.[10] Popcoen uses an artificial neural network trained on molecular dynamic simulations of proteins to carry out fast (∼0.1 s per protein) estimates of how much the protein fluctuates across all backbone and side-chain torsion angles.[10] The Popcoen entropy estimation is calculated by decomposing total S_{cnf} into a sum of per-residue contributions, S_i, where i denotes the residue index. Goethe et al. frame the entropy calculation in terms of bond-angle torsion coordinates, given that most coordinates will have negligible fluctuations due to the covalent structure of the protein. Each S_i is composed of two terms: a marginal entropy of a given torsion angle and a mutual information between two torsion angles. This sum is applied across the entire set of torsion angles in the protein. However, since a brute force computation across the entire set is unfeasible due to its high dimension, an approach similar to the second-order maximum information spanning tree (MIST) approximation of entropy is used. The S_i values are calculated from molecular dynamic (MD) simulations of proteins in explicit solvent (TIP3P water), where they gather the distribution of torsion angle fluctuations and integrate in order to obtain marginal entropies and mutual information between torsion angles. Therefore, higher values of S_{cnf} suggest the protein has higher flexibility in its torsion angles, since it exhibits higher amino acid backbone and side-chain fluctuations. Here, the input to Popcoen is a *de novo* structure and sequence information (contained in a PDB file) and the outputs are predicted configurational entropy values for each residue and for the entire protein. Using the list of per-residue

https://doi.org/10.1021/acs.jpcb.0c09888
J. Phys. Chem. B XXXX, XXX, XXX−XXX

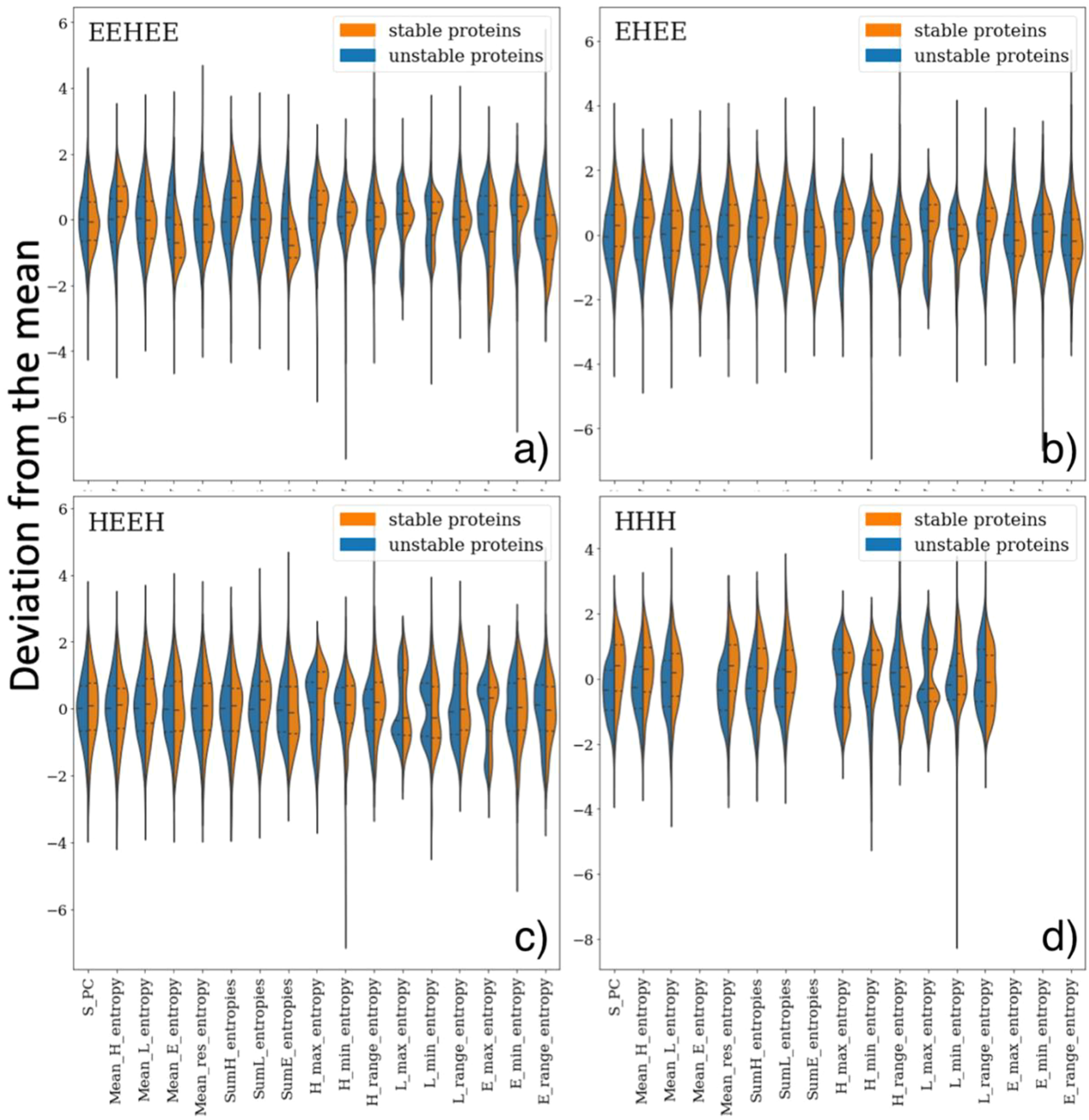


Figure 1. Standardized configurational entropy descriptor distributions within each topology. Blue regions represent stable proteins as defined by stability score; orange regions represent unstable proteins. Distribution differences between stable and unstable categories in descriptors such as *Mean_H_entropy*, *Mean_E_entropy*, and *SumH_entropy* demonstrate the entropy descriptors' potential for classification.

values and DSSP information, we created descriptors containing average, total, maximum, minimum, and range of S_{cnf} values per secondary-structure type. The nomenclature of the descriptors follows a set pattern: *Mean_X_entropy*, *X_min_entropy*, *X_max_entropy*, *X_range_entropy*, and *SumX_entropies* (X = H, E, or L). In addition to these descriptors, the descriptor set includes *S_PC*, the sum of all per-residue entropies, i.e., the total S_{cnf} estimate. These computations were performed using an (Actor-BAsed COmputing) system containerization of the Popcoen software.[14,15] Popcoen's per-residue definition of S_{cnf} allows us to calculate S_{cnf} for any given secondary structure, as long as DSSP information is available. Our calculated *SumX_entropies* features should be interpreted as a sum of partial configurational entropies across residues in a given secondary structure, where a higher *SumX_entropies* value means Popcoen predicted the protein exhibits high backbone and side-chain fluctuations. Positive correlations between stability and S_{cnf} features would suggest that proteins exhibiting high backbone and side-chain fluctuations are more enriched in stable proteins than those with lower fluctuations.

Resulting distributions were characterized using the Kolmogorov−Smirnov (KS) test from SciPy version 1.3.0. A variety of machine learning (ML) models were trained on these features using SciKit-Learn[16] version 0.20.2—specifically, Random Forest Classifier, Scalar Vector Machine, Gradient Boosted Classifier, Keras Neural Network, K-Nearest Neighbors, Naive Bayes Classifier, Gaussian Mixture Model, and Decision Tree Classifier. For each model, our general strategy consisted of starting with a 5-fold cross validation per model. In general, k-fold cross validation is a method for estimating prediction quality, by dividing the data into k subsets ("folds"), training the model on all but one $(k - 1)$ of the subsets, and testing the model on the remaining subset; this is repeated k times, using a different subset for testing each time. We report the uncertainties associated with these. After cross validation, models were audited using Shapley additive explanations

https://doi.org/10.1021/acs.jpcb.0c09888
J. Phys. Chem. B XXXX, XXX, XXX−XXX

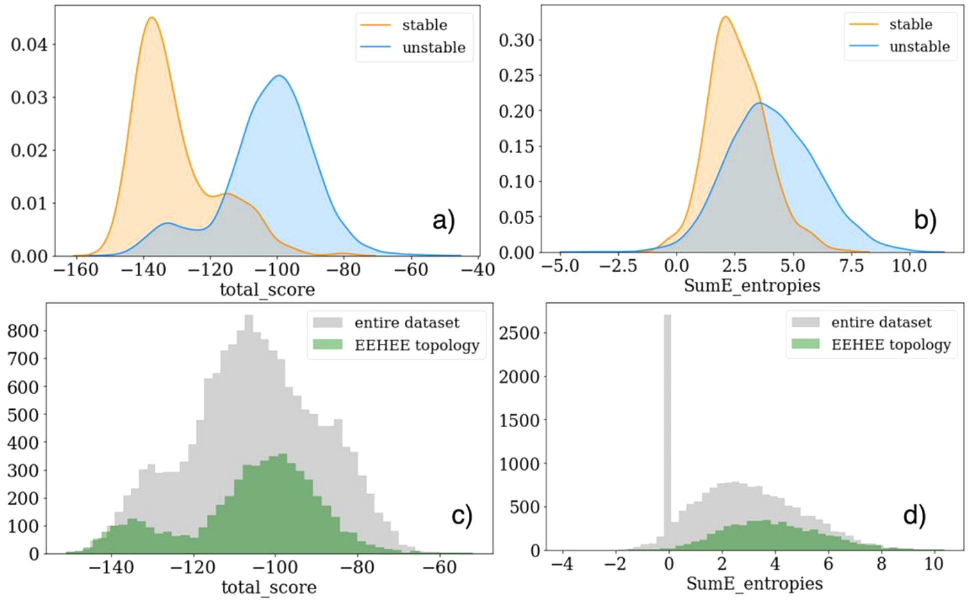


Figure 2. (a) Kernel density estimate plot of stable and unstable EEHEE protein distributions for Rocklin-extended features representing a score calculated using Rosetta and (b) generated feature SumE_entropies representing the total S_{cnf} of β sheet residues. Stable EEHEE proteins favor β sheet rigidity. Two-sample Kolmogorov−Smirnov test value for total_score: 0.71 (0*). (c) Distribution of total_score values and (d) SumE_entropies values across the entire data set and EEHEE topology. The large spike on 0 corresponds to topologies with β sheets.

(SHAP),[17] a game-theoretic method for assessing the direct feature influence on model predictions. Finally, a SciKit-Learn linear regression was used to measure whether the variation from the created features could be explained from existing R features. An *Explain Like I'm 5* (eli5, version 0.9.0)[18] permutation analysis was performed to identify features most important for subsequent linear regression model construction. Mutual information feature selection (Figure S3) and KBest down selection from SciKit-Learn were used to identify which R features captured the most information from the Popcoen S_{cnf}. Pandas version 1.0.1 was used to interact with the data.[19] The complete data sets and an interactive Python 3.70 Jupyter notebook performing the analyses described in this paper are available at https://github.com/jandestrada/Scnf_Publication.

RESULTS AND DISCUSSION

Configurational Entropy Feature Distributions of Stable and Unstable Miniproteins Aggregated across All Topologies Have Modest Differences. Figure S4 compares the distributions of the 17 configurational entropy features introduced in our work for stable (orange) and unstable (blue) proteins when including all topologies. To quantify the differences between the distributions, we applied the two-sample Kolmogorov−Smirnov (KS) test on stable and unstable distributions to rank the S_{cnf} features' potential for classification. The two-sample KS test is a non-parametric test that measures the distance between two empirical distributions, outputting a decimal value between 0 (no distance) and 1 (most distance). The null hypothesis is that both distributions are drawn from the same sample, and thus have no distance between them. This test is an appropriate choice because it makes fewer assumptions than its parametric counterparts. The KS test statistic also has a corresponding p-value, based on the sample sizes. Results are shown in Figure S5 and Table S1. The loop and α helix features, *L_min_entropy*, *SumH_entropies*, *Mean_L_entropy*, and *L_range_entropy*, showed the highest KS values, with 0.22, 0.14, 0.13, and 0.10, respectively. The top entropy feature, *L_min_entropy*, had a higher KS value than 58 of 113 Rocklin-extended (R) features. For reference, the highest KS value for the R features was *score_per_res* (0.65, p-value <0.05). From this, we conclude that, although no individual S_{cnf} descriptor predicts stability and the collection of S_{cnf} descriptors alone may not be capable of predicting protein stability, they are at least as descriptive as the majority of the previously used R features. A notebook carrying out KS analysis can be found in the supporting GitHub repository.[20]

Configurational Entropy Feature Distributions for Stable and Unstable Miniproteins Are More Distinct When Considering Individual Topologies. Visualizing S_{cnf} feature distributions using all topologies did not greatly separate stable from unstable proteins, which motivates visualizing distributions within a specific topology (Figure 1). Three out of four topologies show differences in feature distributions, of which EEHEE showed the strongest distribution separation. Table S1 shows that SumE_entropies in EEHEE topology have the largest discrepancy between stable and unstable proteins (KS value: 0.391, p value <10^{-10}). Unlike the case of considering all proteins (see Table S2), the distributions for stable and unstable EEHEE designs are less symmetrical, indicating a better ability to distinguish stability using these features. This is quantitatively described by higher KS values between stable and unstable distributions. Miniproteins in the EEHEE topology tend to be stable with a low β sheet S_{cnf} value (fewer internal degrees of freedom) and a high α helix S_{cnf} value (more internal degrees of freedom). This opposing trend between strands and helices highlights that S_{cnf} values are not always correlated with stability in the same direction, even within the same topology. This result could suggest that rigid ordering of one secondary-structural element could be compensated by increased flexibility in another element. However, since protein stability arises from differences between folded and unfolded states and since S_{cnf} only quantifies the degree of disorder in the folded state, S_{cnf} is not fully able to address this hypothesis. Figure 2 shows distributions for the entropy feature with the highest KS value (SumE_entropies) and the R feature with the highest KS value (total_score). The distributions have been separated by stable and unstable miniproteins. Figure 2 suggests stable EEHEE proteins favor a low β sheet configurational entropy, highlighting the entropy features potential for classification.

Entropy Features Describe Stability for Three out of Four Topologies. To compare the models' true-positive and false-positive rate, we quantified their performance through the receiver operating characteristic area under curve (ROC-AUC, hereafter abbreviated "AUC") score. First, for every topology, we trained a 5-fold cross validated model using S_{cnf} features, where the average is shown on the diagonal in Figure 3a. Three out of four models showed predictive power with AUCs greater than 0.50. Next, we evaluated whether a model trained on one topology would have predictive power on another topology. This result is shown in the off-diagonal in Figure 3a, where some topologies succeeded but the signal was not generally strong. Thus, training and testing within a topology generally exhibits higher AUC scores than across topologies, although the high performance in some off-diagonal results suggests some predictive generalizability. Testing on the EEHEE topology leads to the best performance, followed by EHEE and HHH; learning on the HEEH topology was not generally successful. It is worth noting that the HEEH topology contained the smallest amount of stable miniproteins, making it inherently difficult for any model to achieve high performance (Figure S2). The best result off-diagonal belongs to the HHH−EEHEE train−test pair. Parts b and c of Figure 3 show how the total α helix entropy distribution difference that is present in HHH is more pronounced in EEHEE. Specifically, the KS value for *EEHEE SumH_entropy* (second column) is greater than that for *HHH SumH_entropies* (first column). This trend is seen for *SumH_entropies* across all topologies (Figure S6). This cross-topology consistency could lead to improved performance. However, cross-topology inconsistency could lead to consistent misprediction, as in the case of the EHEE−HEEH train−test pair's minimum loop residue S_{cnf}. Notice the peak in stable proteins for EHEE directly overlaps with a peak of unstable proteins in HEEH, possibly leading to consistent misprediction and an AUC score below 0.500.

S_{cnf} Metrics Are Correlated with Amino Acid Composition. Having found that S_{cnf} features correlate with protein stability, we next sought to understand the physical basis of this correlation. S_{cnf} measures the number of accessible microstates as a protein fluctuates about all backbone and side-chain torsion angles. Since some side chains are inherently more flexible than others and since side chains differ in how much they constrain backbone flexibility, we hypothesized that S_{cnf} measurements would correlate with the underlying amino acid composition of the protein. To test this hypothesis, we created a new set of descriptors that quantify the amino acid composition within different secondary-structure elements. These descriptors have the pattern *frac_X_Y*, where X = amino acid and Y = secondary structure, and the values have the pattern (*number of X in amino acids of topology Y*)/(*number of amino acids of topology Y*). For example, frac_A_H for a sequence *AYPFA* with secondary structure *HHLHH* would be 2/4, or 0.5.

We then used multiple linear regression to search for a correlation between the above features and S_{cnf} values for specific secondary-structure elements for a given topology.

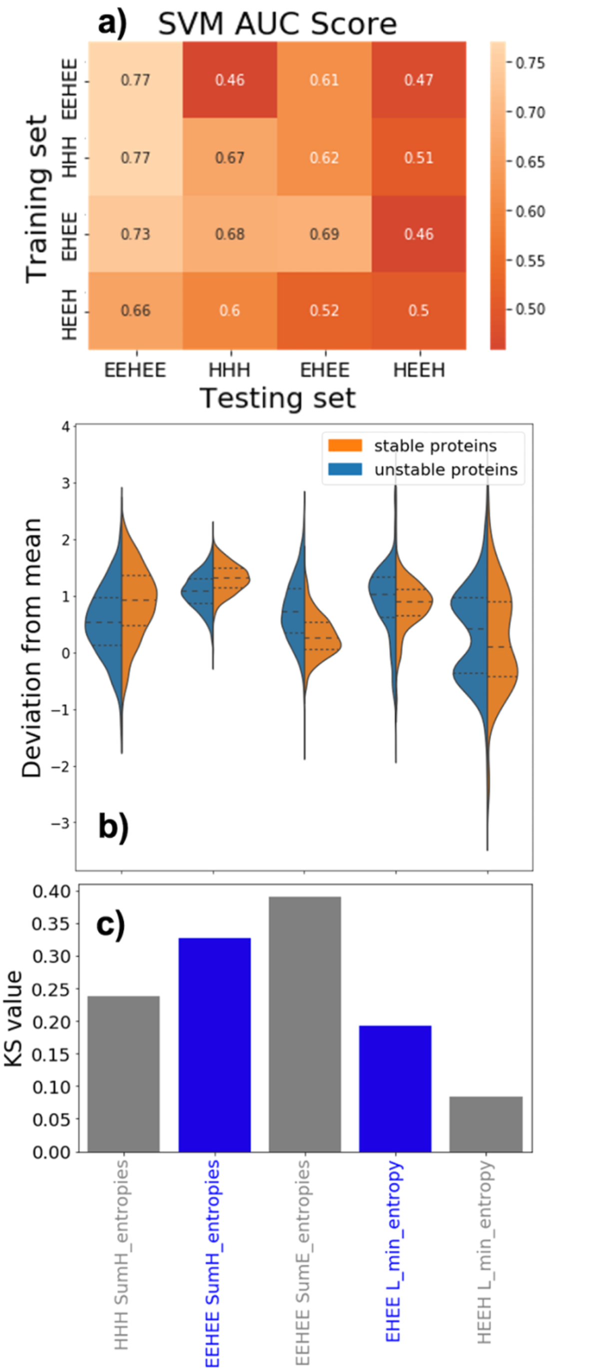


Figure 3. (a) Heatmap showing AUC score results for SVM trained on the topology on the *y*-axis and tested on the topology on the *x*-axis using only entropy features. Results on the diagonal correspond to a 5-fold cross validation within a topology. Scores below 0.50 could suggest trends present in the training set lead to consistent mispredictions in the testing set. (b) Kernel density distributions for entropy features within specific topologies detailed in panel c's *x*-axis. The *y*-axis shows the displacement from the mean for each feature, where the mean was calculated collectively from all topologies, which is why features are not centered around zero. (c) Two-sample Kolmogorov−Smirnov values for distribution of stable−unstable miniproteins.

Figures S7, S8, S9, and S10 show the results of this analysis for all four topologies. In each case, the regressions show an intermediate correlation between amino acid composition and S_{cnf}, within each given topology. The amino acids that are most predictive of entropy are not always the same between secondary structures and topologies. This may partially stem from different secondary structures having different underlying frequencies of amino acids, due to Rosetta's bias during design or other restrictions imposed in the design protocol. However, comparing the effects of amino acids in the same secondary structure largely eliminates this influence. Next, we sought to identify amino acids that were correlated with both stability (Figure S11) and S_{cnf}. We identified multiple amino acids that fit this profile. For instance, stable EEHEE designs tend to have low S_{cnf} in strands and high S_{cnf} in helices. We observe that strand S_{cnf} and stability are both negatively correlated with the frequency of E and K residues in strands, whereas helical S_{cnf} and stability are both positively correlated with the frequency of M and F in helices (Figure S7). When considered jointly, designs that are most depleted in E and K residues in strands and most enriched for M and F in helices tend to be qualitatively much more stable than designs showing the opposite trend (Figure S12). The EHEE topology also tends to be more stable when β sheet S_{cnf} is lower and α helix S_{cnf} is higher. In this case, the frequency of W in strands is negatively correlated with strand S_{cnf} and positively correlated with stability, while the frequency of M in helices is positively correlated with both helical S_{cnf} and stability (Figure S8). Finally, designs from the HHH topology tend to be more stable with high values of total S_{cnf}. We find that S_{cnf} and stability are both correlated with the frequency of V and I in helices and both negatively correlated with the frequency of P and N in loops (Figure S9). This analysis did not provide useful information for the HEEH topology given there were no correlations between HEEH proteins' S_{cnf} and stability (Figures S10 and S13). As with EEHEE, grouping EHEE and HHH designs based on these patterns in amino acid composition yields groups of designs with substantially different stability distributions (Figures S14 and S15). Together, these results suggest that the predictive power of S_{cnf} features arises partly from a protein's amino acid composition.

Existing Rocklin-Extended (R) Features Partially Describe Configurational Entropy Variations. We assessed the correlation between R features and total S_{cnf} in order to evaluate whether they implicitly capture variations from S_{cnf} features. We used a linear regression to evaluate this correlation, which showed an R^2 score of 0.811. However, due to the covariant nature of the data (see Figure S16), we needed to take a different approach to understand what features were responsible for explaining the variance. Two strategies were used to determine the most important R features. Scikit-learn's KBest feature downselection method was used to determine which features account for most of the variance in Popcoen's S_{cnf}, where "k" features are chosen which maximize the F1 score of the linear regression on an 80−20 training−testing data split. Using $k = 40$ leads to $R^2 = 0.646$. Using $k = 10$ leads to $R^2 = 0.465$. Finally, if $k = 1$, the selected feature is *lk_ball_iso*, the Lazaridis−Karplus solvation free energy, which leads to $R^2 = 0.414$ (Figure 4); the negative slope is consistent with a larger S_{cnf} reducing the total free energy. Proteins with higher S_{cnf} have a slightly higher hydrophobicity, but these two properties are essentially uncorrelated ($R^2 = 0.032$). An increase in our S_{cnf} features corresponds to an increase in internal degrees of freedom for a given miniprotein. Feature sets for these k values can be found in Table S3 and in the supporting GitHub repository under **Model_pred_Spc**.[21] We

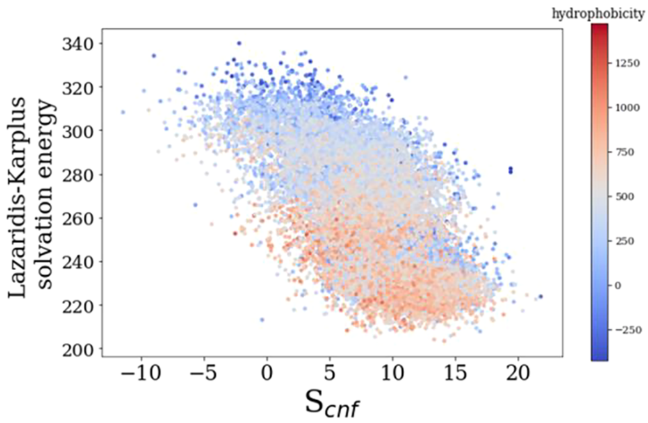


Figure 4. Lazaridis−Karplus solvation energy as a function of configurational entropy. Each point indicates a protein in this data set, with total hydrophobicity color coded.

found correlations between amino acid composition and secondary-structure configurational entropy in order to identify what the entropy features were capturing. Lasso regression was used in order to maximize the regression performance and minimize the number of features used (Figure 5b). Using an α value of 0.1, an R^2 of 0.732 was observed, yielding a set of 34 features. Using this approach, we find a smaller set of features and a regression with a higher R^2 score when compared to the K-Best approach. This improvement is observed because LASSO regression applies weights to feature coefficients in order to penalize high magnitude features, which can happen when features are highly covariant. K-Best selects its features using one-way ANOVA between the feature and label; therefore, it neglects covariance between features. The eli5 permutation analysis was performed (Figure 5c), confirming the observed set of features from the Lasso regression. The most important feature is *lk_ball_iso*. Although the data set has a large abundance of stable α helix containing structures (Figure S2), there is no apparent effect on the feature importance values (see supporting notebook[21]). When training the LASSO regression to predict each S_{cnf} feature, the two best R^2 were for S_PC and SumE_entropies; the worst R^2 were for all loop entropy features (L_range_entropy, SumL_entropies, L_min_entropy, L_max_entropy, and Mean_L_entropy) (see Figure S17).

Figure S18 summarizes the results of this feature selection process as a Venn diagram. K-Best and Lasso regression agree in their assignment of 11 features as important to configurational entropy, corresponding to solvation energy (lk_ball_iso, net_sol_per_res), attractive interactions (fa_atr), packing quality (holes, degree), and hydrogen bonding (hbond_sr_bb_per_sheet, hbond_sr_bb).

Machine Learning Models Incorporating Configurational Entropy Features Do Not Predict Protein Stability Better than Those Trained on Rocklin-Extended Features (R) Alone. Eight classifiers—support vector machine (SVM), random forest (RF), neural network (NN), gradient boosted (GBC) tree, K-nearest neighbors (K-NN), naive bayes (NB), decision tree (DT), and Gaussian mixture model (GMM)—were trained and tested on a 5-fold cross validation split which included all protein topologies, where training sets were sampled to have equal amounts of stable and unstable miniproteins, due to overall data set class imbalance. The results are shown in Figure 6. The RF, SVM, NBC, and K-

https://doi.org/10.1021/acs.jpcb.0c09888
J. Phys. Chem. B XXXX, XXX, XXX−XXX

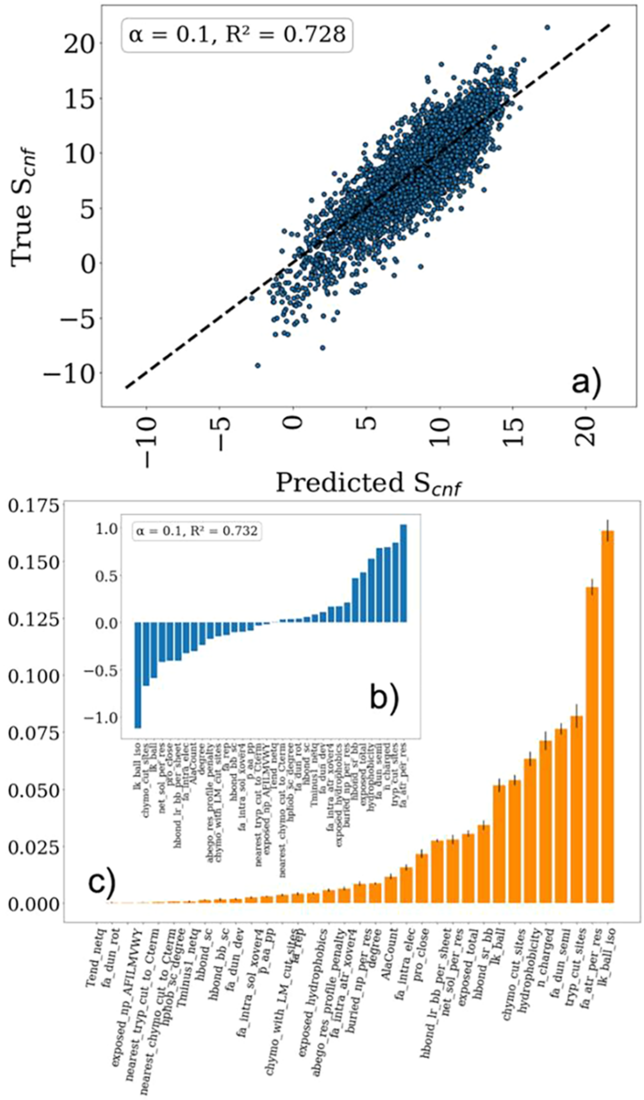


Figure 5. Performance of a LASSO linear regression model using Rocklin-extended (R) descriptors to predict the protein's total configurational entropy (S_{cnf}) as computed by Popcoen. (a) Predicted value of S_{cnf} on the *x*-axis and the Popcoen value of S_{cnf} on the *y*-axis for the test set. The small spread about the bisectrix between predicted and true values shows the R descriptors accurately capture most of the variation in S_{cnf}. (b) LASSO linear regression coefficients. (c) eli5 permutation feature importances. Descriptors were standardized prior to fitting in order to comparatively visualize their contributions.

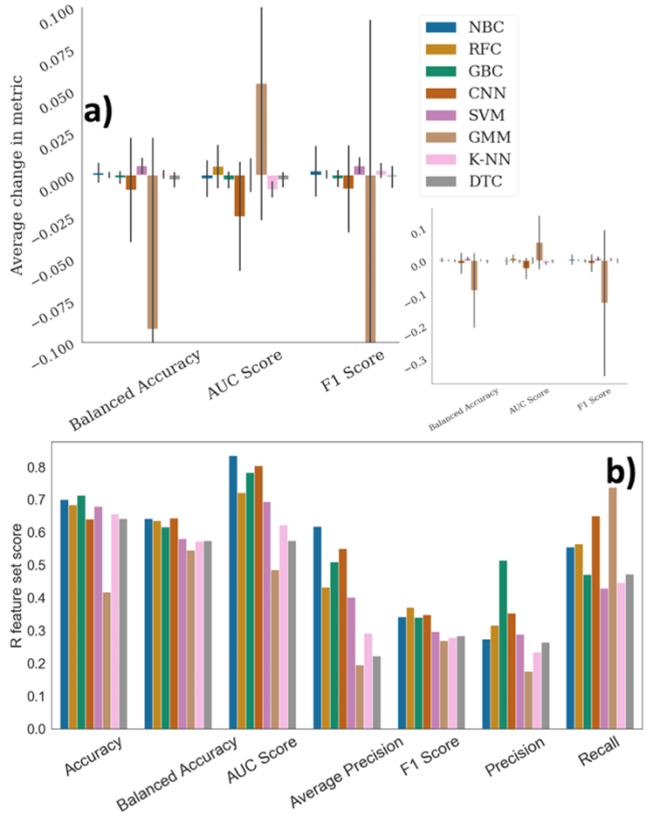


Figure 6. (a) Difference between the average performance for models including Rocklin-extended (R) descriptors and configurational entropy (S_{cnf}) descriptors versus only R descriptors. Models shown include support vector machines (SVMs), random forest classifier (RFC), convolutional neural network (CNN), gradient boosted classifier (GBC), K-nearest neighbors (K-NN), naive Bayes classifier (NBC), decision tree classifier (DTC), and Gaussian mixture model (GMM). Error bars show differences across 5-fold cross validation. The inset shows the total magnitude of standard deviation across cross validation. (b) Baseline values for ML models trained using the R feature set.

NN models on average exhibited improved precision when both R and configurational entropy (S_{cnf}) features were included; these models also exhibited reduced recall. However, the balanced accuracy (i.e., the average accuracy of predicting stable and unstable proteins) did not show the same change. Inclusion of S_{cnf} features slightly reduces these models' likelihood for false positives on average. We trained and tested the models within each topology, since we showed the features have the potential to be predictive based on our KS value analysis. ML models incorporating S_{cnf} features and R features perform on average the same as ML models incorporating only R features. Performance for R models can be accessed at *Scnf_ML_Performance*.

Machine Learning Models Using a Subset of the Rocklin-Extended Feature Set Are Able to Predict Protein Stability. A set of 55 R descriptors whose Kolmogorov−Smirnov (KS) value was greater than the highest configurational entropy (S_{cnf}) descriptor's KS value was selected. Machine learning models were trained using this reduced feature set (R_sub) and the reduced R set with S_{cnf} features (RS_sub). The difference between using RS_sub and all R features is on average 0.0056 across all models across all metrics, suggesting the remaining 58 R features are nonessential for our predictive models. The difference between RS_sub and R_sub is on average 0.0071 across all models, suggesting entropy features do not significantly improve these models (Table S4). Including S_{cnf} features in addition to the reduced feature set does not improve the models. This result could be explained using the fact that 35/55 features in the reduced R set were shown above to implicitly capture S_{cnf} through Lasso regression and K-Best feature downselection (Table S5). This implies that the contribution of configurational entropy to protein stability prediction can be described by the subset of retained R features.

Models Containing Only Entropy Features Evaluated on a Leave-One-Out Topology (LOO) Train−Test Split

https://doi.org/10.1021/acs.jpcb.0c09888
J. Phys. Chem. B XXXX, XXX, XXX−XXX

Exhibited Inferior Scores When Compared to Models Containing All Rocklin-Extended Features. We surveyed all of the ML models, training on all topologies except one. This left-out topology served as the test set in order to measure the model's potential to generalize to very different types of interactions. Inclusion of S_{cnf} features with R features does not greatly change the performance compared to only R features, with all models achieving an AUC around 0.5−0.6. These results suggest entropy features do *not* make models more generalizable to unseen topologies (Figures S19−S24). An exception is the case of the DTC; models including R and S_{cnf} features slightly outperformed models when including both data sets (see Figure 7).

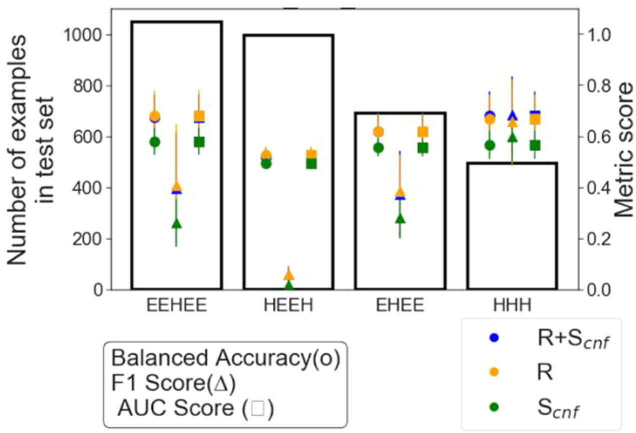


Figure 7. Decision tree classifier (DTC) performance on the leave-one-out train−test split where the model is trained on all topologies except one, which is used as the testing set. The figure shows models trained using only the set of 17 entropy features (S_{cnf}) often perform near the same level as models trained using all 113 Rocklin-extended features (R).

■ CONCLUSIONS

Predicting the stability of small *de novo* proteins can advance the development of new targeted therapeutics[6] and bionanomaterials.[7] Machine learning models trained on large, high-throughput experimental data sets can assist in these predictions but depend upon the quality of the input features used to describe the proteins. Machine learning interpretability can also be aided by using physically motivated input descriptors. Previous work focused on a subset of structural descriptors and energy features extracted from the Rosetta force field; we have extended this to include configurational entropy (S_{cnf}) protein descriptors. S_{cnf} features on their own adeptly discriminate between stable and unstable miniprotein designs, shown by Kolmogorov−Smirnov values for the highest S_{cnf} value, which was greater than 58 features based on Rosetta energy function properties and geometric features (R). S_{cnf} features discriminate stable and unstable designs for three out of four topologies. S_{cnf} features captured interactions in the EEHEE topology best, followed by EHEE and HHH. S_{cnf} features captured interactions in HEEH the least, possibly due to the small number of stable miniproteins. Although configurational entropy is not explicitly included in the R features, 81% of the configurational entropy variation is captured by a subset of 12 R features (Figure 5), dominated by protein packing, solvation energy, and attractive−repulsive interactions. However, machine learning model precision was not improved by addition of these entropy features. These results suggest that, although the R features do not explicitly incorporate configurational entropy, machine learning models built on a sufficiently large and diverse feature set can use a subset of them to describe the measure of configurational entropy used in this study, which estimates the amount that a protein would fluctuate in an MD simulation. Although many of the features correlated to S_{cnf} are Rosetta energy terms, some are not (e.g., number of charged or hydrophobic residues and the distribution of backbone torsion angles in designs relative to native proteins), suggesting there may be interactions missing from Rosetta to completely capture S_{cnf}.

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jpcb.0c09888.

List of Jupyter notebook and description of content, additional mutual information analysis, breakdown of topology-specific relationships, and additional linear regression results (PDF)

■ AUTHOR INFORMATION

Corresponding Author

Joshua Schrier − *Department of Chemistry, Haverford College, Haverford, Pennsylvania 19041, United States; Department of Chemistry, Fordham University, The Bronx, New York 10458, United States;* orcid.org/0000-0002-2071-1657; Email: jschrier@fordham.edu

Authors

Jan D. Estrada Pabón − *Department of Chemistry, Haverford College, Haverford, Pennsylvania 19041, United States*
Hugh K. Haddox − *Institute for Protein Design, University of Washington, Seattle, Washington 98195, United States*
Greg Van Aken − *Department of Chemistry, Haverford College, Haverford, Pennsylvania 19041, United States*
Ian M. Pendleton − *Department of Chemistry, Haverford College, Haverford, Pennsylvania 19041, United States*
Hamed Eramian − *Netrias LLC, Arlington, Virginia 22201, United States*
Jedediah M. Singer − *Two Six Technologies, Arlington, Virginia 22203, United States*

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jpcb.0c09888

Notes

The authors declare no competing financial interest.
Additional data sets and an interactive Python 3.70 Jupyter notebook performing the analyses described in this paper are available at https://github.com/jandestrada/Scnf_Publication.

■ ACKNOWLEDGMENTS

We thank Sorelle Friedler and Gabriel Rocklin for helpful conversations and acknowledge the Texas Advanced Computing Center (TACC) at The University of Texas at Austin for providing HPC resources that have contributed to the research results reported within this paper. This study is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR001118C0036. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA. J.S. acknowledges the Henry Dreyfus Teacher-Scholar Award (TH-14-010).

■ REFERENCES

(1) Alford, R. F.; Leaver-Fay, A.; Jeliazkov, J. R.; O'Meara, M. J.; DiMaio, F. P.; Park, H.; Shapovalov, M. V.; Renfrew, P. D.; Mulligan, V. K.; Kappel, K.; Labonte, J. W.; Pacella, M. S.; Bonneau, R.; Bradley, P.; Dunbrack, R. L.; Das, R.; Baker, D.; Kuhlman, B.; Kortemme, T.; Gray, J. J. The Rosetta All-Atom Energy Function for Macromolecular Modeling and Design. *J. Chem. Theory Comput.* **2017**, *13* (6), 3031−3048.

(2) Dill, K. A.; Ozkan, S. B.; Shell, M. S.; Weikl, T. R. The Protein Folding Problem. *Annu. Rev. Biophys.* **2008**, *37*, 289−316.

(3) Ferreiro, D. U.; Komives, E. A.; Wolynes, P. G. Frustration, Function and Folding. *Curr. Opin. Struct. Biol.* **2018**, *48*, 68−73.

(4) Bergasa-Caceres, F.; Haas, E.; Rabitz, H. A. Nature's Shortcut to Protein Folding. *J. Phys. Chem. B* **2019**, *123* (21), 4463−4476.

(5) Perez, A.; Morrone, J. A.; Simmerling, C.; Dill, K. A. Advances in Free-Energy-Based Simulations of Protein Folding and Ligand Binding. *Curr. Opin. Struct. Biol.* **2016**, *36*, 25−31.

(6) Huang, P.-S.; Boyken, S. E.; Baker, D. The Coming of Age of de Novo Protein Design. *Nature* **2016**, *537* (7620), 320−327.

(7) Gao, X.; Matsui, H. Peptide-Based Nanotubes and Their Applications in Bionanotechnology. *Adv. Mater.* **2005**, *17* (17), 2037−2050.

(8) Rocklin, G. J.; Chidyausiku, T. M.; Goreshnik, I.; Ford, A.; Houliston, S.; Lemak, A.; Carter, L.; Ravichandran, R.; Mulligan, V. K.; Chevalier, A.; Arrowsmith, C. H.; Baker, D. Global Analysis of Protein Folding Using Massively Parallel Design, Synthesis, and Testing. *Science* **2017**, *357* (6347), 168−175.

(9) Raccuglia, P.; Elbert, K. C.; Adler, P. D. F.; Falk, C.; Wenny, M. B.; Mollo, A.; Zeller, M.; Friedler, S. A.; Schrier, J.; Norquist, A. J. Machine-Learning-Assisted Materials Discovery Using Failed Experiments. *Nature* **2016**, *533* (7601), 73−76.

(10) Goethe, M.; Gleixner, J.; Fita, I.; Rubi, J. M. Prediction of Protein Configurational Entropy (Popcoen). *J. Chem. Theory Comput.* **2018**, *14* (3), 1811−1819.

(11) Atkinson, J. T.; Jones, A. M.; Nanda, V.; Silberg, J. J. Protein Tolerance to Random Circular Permutation Correlates with Thermostability and Local Energetics of Residue-Residue Contacts. *Protein Eng., Des. Sel.* **2019**, *32* (11), 489−501.

(12) Atkinson, J. T.; Jones, A. M.; Nanda, V.; Silberg, J. J. Family Permutation Profiling Identifies a Dynamic Protein Domain as Functionally Tolerant to Increased Conformational Entropy. 2019, bioRxiv 840603. biorxiv.org e-Print archive.

(13) Kabsch, W.; Sander, C. Dictionary of Protein Secondary Structure: Pattern Recognition of Hydrogen-Bonded and Geometrical Features. *Biopolymers* **1983**, *22* (12), 2577−2637.

(14) Van Aken, G. Implementing an Actor-Based Computing System for High-Throughput Featurization of Protein Structures for Machine Learning; Haverford College, 2019.

(15) Van Aken, G. gavanaken/Abaco-Popcoen. https://github.com/gavanaken/Abaco-Popcoen (accessed September 15, 2020).

(16) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; Duchesnay, E. Scikit-Learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12* (Oct), 2825−2830.

(17) Lundberg, S. M.; Lee, S.-I. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems 30*; Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: 2017; pp 4765−4774.

(18) Permutation Importance — ELI5 0.9.0 documentation. https://eli5.readthedocs.io/en/latest/blackbox/permutation_importance.html (accessed February 10, 2020).

(19) McKinney, W. Data Structures for Statistical Computing in Python; Austin, TX, 2010; pp 56−61. DOI: 10.25080/Majora-92bf1922-00a.

(20) jandestrada. Scnf_Feature_Statistics notebook. https://github.com/jandestrada/Scnf_Publication/blob/master/Scnf_Feature_Statistics.ipynb (accessed March 18, 2021).

(21) Model_pred_Spc notebook. https://github.com/jandestrada/Scnf_Publication/blob/master/Model_pred_Spc.ipynb (accessed March 18, 2021).

(22) Singer, J. M.; Novotney, S.; Strickland, D.; Haddox, H. K.; Leiby, N.; Rocklin, G. J.; Chow, C. M.; Roy, A.; Bera, A. K.; Motta, F. C.; Cao, L.; Strauch, E.-M.; Chidyausiku, T. M.; Ford, A.; Ho, E.; Mackenzie, C. O.; Eramian, H.; DiMaio, F.; Grigoryan, G.; Vaughn, M.; Stewart, E. H.; Baker, D.; Klavins, E. Large-scale design and refinement of stable proteins using sequence-only models. *bioRxiv* 2021.03.12.435185; DOI: 10.1101/2021.03.12.435185.