Final Report

Bilgehan Çağıltay
Büşra Tayhan
Faraz Badali Naghadeh
Alize Sevgi Yalçınkaya

Exploring the Impact of Mutations on Protein Function

Abstract

Deep mutational scanning (DMS) is a technique used by researchers to assess the effects of mutations on protein function. This method achieves to identify connections between mutations and their effects on molecular function and evolutionary processes. In this project, we focused on pinpointing specific mutations and their locations on the GB1 protein that would affect its interaction with IgG-Fc proteins. By implementing the MuMi (mutation and minimization) method, we systematically generated all possible single mutations of GB1. This comprehensive approach allowed us to construct a network graph, which then facilitated the development of an XGBoost model, which delivers superior outcomes compared to prior investigations in the field.

Introduction

Proteins, crucial agents within cells, are composed of amino acids arranged in a chain-like structure, with each protein displaying a unique arrangement of these building blocks. [1], [2] The sequence of amino acids is determined by the genetic code encoded in genes. Mutations, or changes in genes, can introduce variations in the proteins they code for. This variation may involve the substitution of one type of amino acid for another, leading to a diverse array of protein structures. [3] While some mutations have no discernible impact, others can significantly affect the protein's functionality, potentially impeding its ability to perform its cellular role effectively.

In efforts to comprehensively understand the relationship between genetic variations and protein function, researchers employ a technique known as deep mutational scanning (DMS). [4], [5] DMS studies involve the systematic analysis of millions of mutated variants of a protein or regulatory sequence. By assaying these variants, scientists can create detailed maps that highlight how specific mutations influence the function of proteins or regulatory elements. [6] This approach aims to unravel the intricacies of molecular function and evolution by collecting extensive sequence–function pairs. The vastness of sequence space poses a challenge in obtaining a complete sequence–function map, but DMS provides valuable insights into the impact of mutations on proteins and their roles within the cell. [1]

This project centers on a detailed exploration of the alterations in binding efficiency concerning both regions and amino acids. The primary objective is to scrutinize these changes by meticulously comparing the compatibility of deep mutation screening experiments with the data derived from the application of the MuMi method to the GB1 protein structure.[7] By delving into this comparative analysis, the study seeks to unravel the nuanced impact that variations in genetic sequences may have on the binding efficiency of proteins. The intricate interplay between deep mutation screening and MuMi data offers a unique lens through which to understand how molecular changes influence the functional aspects of the GB1 protein structure. This investigation contributes valuable insights to the broader field of molecular biology, shedding light on the specific effects of genetic mutations on binding mechanisms at both the regional and amino acid levels.

Methods

Data extraction:

Deep mutation scan experimental data was obtained from the GB1 protein, which consists of 56 amino acids, in both its unbound (PDB Code: 1PGA) and IgG-Fc bound (PDB Code: 1FCC) forms, by utilizing MuMi (Mutation and Minimization) method. The choice of this protein was based on the availability of experimental data, serving as a reference to facilitate the identification of folding and binding energies through various models, additionally, the protein includes a relatively small number of amino acids.

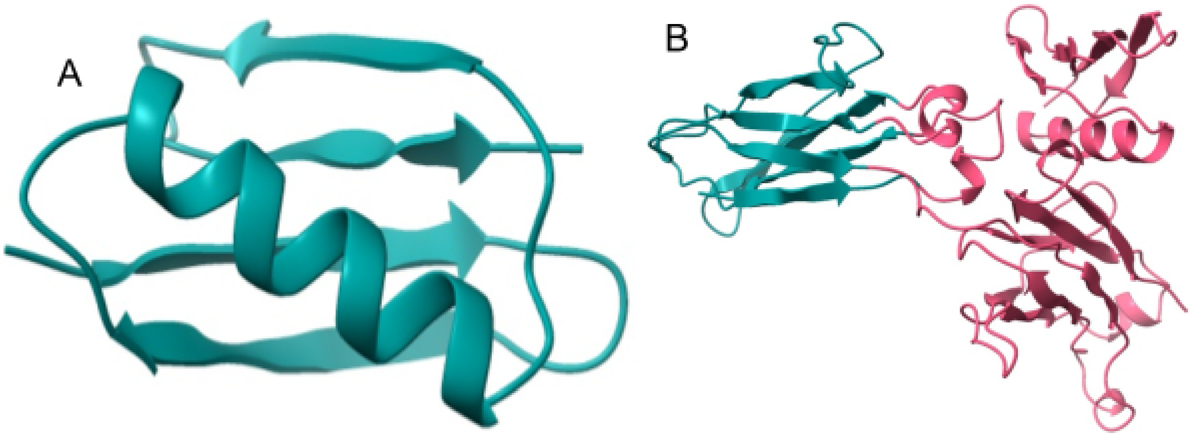


Figure 1. *Unbound GB1 structure* **(A)**, *and IgG-Fc bound form* **(B)**

In the MuMi method, all possible single mutations on GB1 were transformed using the VMD software (56 x 19 = 1064 structures). Subsequently, they were placed in an isotonic saline solution (0.15 molar) and subjected to 10,000 steps of optimization under the Charmm36 force field using the NAMD software. The final structures were then saved in a PDB format after the optimization process. In this stage, to understand the relationship between conformational changes and binding energy, at first, hydrogen bonds between GB1 and IgG-Fc proteins are calculated from PDB structures. On the other hand, protein conformational changes are driven not only by a protein's internal atom–atom interactions, Solvent accessible surface area (SASA) of proteins also gives us clues to understand protein folding and stability. Based on these insights, by utilizing these PDB structures, solvent-accessible surface area

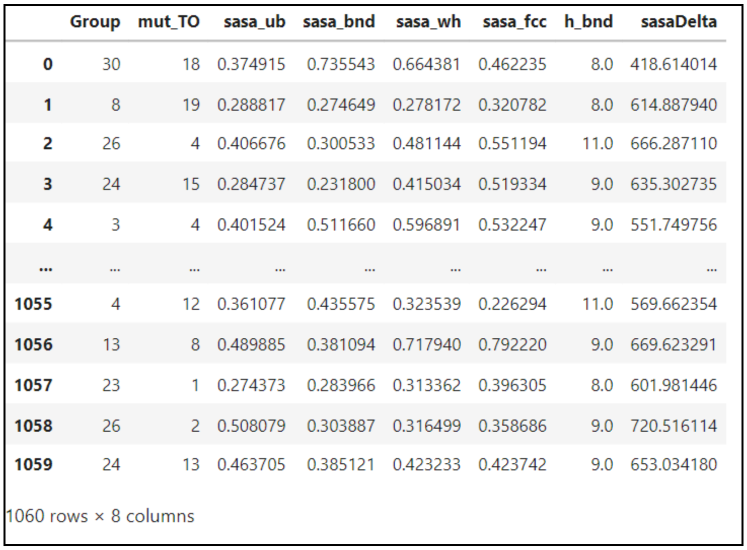
(SASA) in both bound and unbound states is also calculated. Binding energies ($\Delta G_{binding}$) are also obtained from experimental data(56x19). Based on these calculations and the research process, a dataset was created.

Feature Extraction and Data Preprocessing:

A major challenge regarding this project was to choose an appropriate approach to split data for training and testing since every row of the data represents an amino acid in a sequence and slicing rows would change the protein sequence which can change the three-dimensional structure of the protein and impact the learnability of the protein. Additionally, the sliced part of the protein for the train data might not contain useful information regarding binding energy and SASA, values which would make the training data unusable. The test data portion may also contain a significant portion of the crucial information.

Another approach for splitting data into train and test is to split the data based on the columns, which would mean splitting based on each mutation residue. However, this method would result in a low amount of data points.

A solution to this issue is to rethink how we can interpret the dataset. As detailed above, each data row represents a 56 amino acid chain-long protein and each column shows different mutations, because there are 20 amino acids in nature and each residue has a unique amino acid type; 19 columns are included in the dataset. Therefore, it is essential to interpret the dataset as a 2D representation of the different aspects of 3D data. By doing this, instead of thinking of the number of rows or columns as data points to be split across, we can think of each cell as a datapoint, which would allow us to increase our datapoint count and do a better train-test split. In doing so, we created a dataframe where every row is a specific mutation at a specific residue. This also helped reduce our dimensionality and enabled us to create a network graph to try and capture the relationships between particular mutations.



	Group	mut_TO	sasa_ub	sasa_bnd	sasa_wh	sasa_fcc	h_bnd	sasaDelta
0	30	18	0.374915	0.735543	0.664381	0.462235	8.0	418.614014
1	8	19	0.288817	0.274649	0.278172	0.320782	8.0	614.887940
2	26	4	0.406676	0.300533	0.481144	0.551194	11.0	666.287110
3	24	15	0.284737	0.231800	0.415034	0.519334	9.0	635.302735
4	3	4	0.401524	0.511660	0.596891	0.532247	9.0	551.749756
...	...	...	...	...	...	...	...	...
1055	4	12	0.361077	0.435575	0.323539	0.226294	11.0	569.662354
1056	13	8	0.489885	0.381094	0.717940	0.792220	9.0	669.623291
1057	23	1	0.274373	0.283966	0.313362	0.396305	8.0	601.981446
1058	26	2	0.508079	0.303887	0.316499	0.358686	9.0	720.516114
1059	24	13	0.463705	0.385121	0.423233	0.423742	9.0	653.034180

1060 rows × 8 columns

Table 1: *Standardized feature data frame that is used in network graph creation*

In our data, the original amino acid in each residue was NaN, so we removed those rows from our dataframe, as the NaN values were common between each dataframe, creating a few unusable rows. In the experimentation part of this project, the binding energy was unable to be observed for the first residue for any mutation so we filled that with the mean of each mutation. This project focused on predicting which position and transition may trigger the binding activity, to achieve this aim, using each possible position and transition became very important, for this reason, instead of deleting the entire row, filling the binding energy of the first position with the mean binding energy values was believed to be more productive. The resulting feature dataframe from this achieved a shape of 1060x7, where there were 7 features and 1060 data points. Aside from the number of hydrogen bonds between the bounded proteins and the label, everything was standardized accordingly.

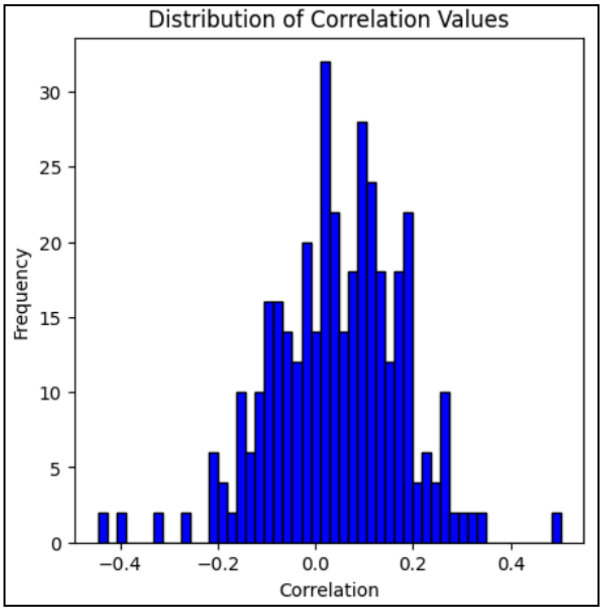
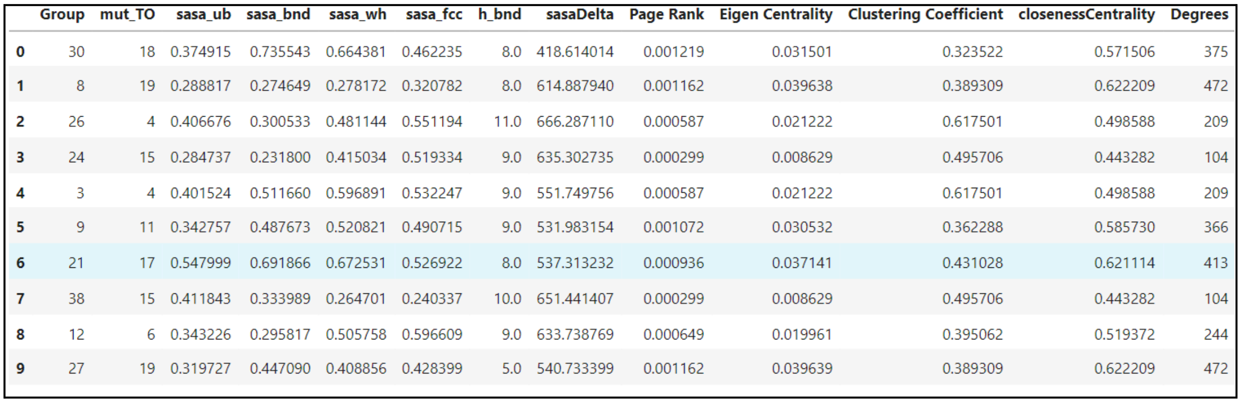


Figure 2: *Distribution of correlation values, used in determining network edges.*

To create a network graph to capture these relationships, the similarity of the change of each mutation at each residue caused to the overall structure of the molecule was used; these were then used to create a correlation matrix. This was achieved by creating a ΔSASA feature, which is the difference between unstandardized SASA unbound and SASA bound. SASA is a very important metric for understanding conformational changes in the protein. Therefore, using the difference between SASA bound and unbound forms might give us good insight to make predictions about protein interactions. The Networkx library of Python was used to create this network graph with edges created from ΔSASA and nodes with the attributes of their entire row. The nodes were identified by their residue number (Group) and what that amino acid was mutated to (mut_TO).

Using this network graph, 5 more features were extracted; page rank, degree, eigen centrality, clustering coefficient, and closeness centrality. These were picked due to their relation to a node's properties. Other network attributes were not selected as they either did not relate to a node but an overall characteristic of the entire network, or because the usefulness of the property did not relate to the use case of this project.

Page rank describes the importance of a node by taking into account the importance of its neighbours. The degree of a node describes the number of connections a node has. The eigen centrality of a node measures a node's importance by considering the importance of its neighbours while taking into account the centrality of a node. The clustering coefficient measures the degree to which nodes in a graph tend to cluster together. Finally, closeness centrality describes the proximity of a node to other nodes in the graph.



	Group	mut_TO	sasa_ub	sasa_bnd	sasa_wh	sasa_fcc	h_bnd	sasaDelta	Page Rank	Eigen Centrality	Clustering Coefficient	closenessCentrality	Degrees
0	30	18	0.374915	0.735543	0.664381	0.462235	8.0	418.614014	0.001219	0.031501	0.323522	0.571506	375
1	8	19	0.288817	0.274649	0.278172	0.320782	8.0	614.887940	0.001162	0.039638	0.389309	0.622209	472
2	26	4	0.406676	0.300533	0.481144	0.551194	11.0	666.287110	0.000587	0.021222	0.617501	0.498588	209
3	24	15	0.284737	0.231800	0.415034	0.519334	9.0	635.302735	0.000299	0.008629	0.495706	0.443282	104
4	3	4	0.401524	0.511660	0.596891	0.532247	9.0	551.749756	0.000587	0.021222	0.617501	0.498588	209
5	9	11	0.342757	0.487673	0.520821	0.490715	9.0	531.983154	0.001072	0.030532	0.362288	0.585730	366
6	21	17	0.547999	0.691866	0.672531	0.526922	8.0	537.313232	0.000936	0.037141	0.431028	0.621114	413
7	38	15	0.411843	0.333989	0.264701	0.240337	10.0	651.441407	0.000299	0.008629	0.495706	0.443282	104
8	12	6	0.343226	0.295817	0.505758	0.596609	9.0	633.738769	0.000649	0.019961	0.395062	0.519372	244
9	27	19	0.319727	0.447090	0.408856	0.428399	5.0	540.733399	0.001162	0.039639	0.389309	0.622209	472

Table 2: *Standardized feature dataframe without one-hot-encoding.*

To get a better idea of which mutations affected binding energy the most, mut_TO was one-hot encoded, which brought the size of the dataframe to 1060*32 (19 mutations, 13 features).

As a result of this preliminary work, a large feature set was acquired. This new feature set contained the normalized SASA values, which were normalized between zero and one. The unbounded protein's SASA (sasa unbound), the bounded protein's SASA, (sasa bound), the SASA of both the mutated protein as well as the bounding protein's SASA (sasa whole), and the SASA of the bounding protein (sasa fcc) were added. The values of intermolecular hydrogen bond counts (H - bond), ΔSASA, and the relevant network properties were also added to the feature list. A sample of this full feature set can be seen in Table 3.

Group	Mutate to	Sasa unbound	Sasa bound	Sasa whole	Sasa fcc	H - bond	Sasa Delta	Page Rank	Eigen Centrality	Cluster Coeff.	Closeness Centrality	Degrees
18	TYR	0.191932	0.3314	0.3378	0.3818	10.0	531.33	0.001	0.03972	0.38907	0.62257	473
19	ARG	0.507663	0.5535	0.5356	0.4327	9.0	588.19	0.001	0.04669	0.34103	0.68455	571
28	ARG	0.494113	0.2322	0.2478	0.3268	7.0	750.70	0.001	0.04676	0.34109	0.68499	572

Table 3: *Sample values of the feature set*

Model:

Due to expert advisory, the LASSO model was used first. Since the project aimed to determine what changes and aspects in a residue affected the binding energy the most, it was logical to use a LASSO model as it did feature extraction while training. However, since the selected features were not independent of each other and the *label* contained many and very important outliers, the LASSO model could not perform well. Most importantly, looking at the data through correlation matrices, heatmaps, and graphs, it was observed that the features and the label did not have a linear relationship. This same issue was encountered with ElasticNet as well. Even though LASSO is robust towards outliers due to the L1 penalty term on the coefficients, it wasn't able to handle outliers in the label. The second biggest problem was the multicollinearity in the data: LASSO can handle some dependency between the features but this particular data has more than it could without sacrificing interpretability and therefore it caused the model's coefficients to vary dramatically and LASSO's variable selection suffered. [8] This wasn't able to be offset by using ElasticNet because of the nonlinearity and so non-linear models were used instead.

Based on the experiments and inferences, it was found that using regression methods on this data is difficult to get accurate results. This is due to the features in our dataset being highly dependent on each other, the existence of extreme outlier data points in certain features, and minimal robustness in linear regression algorithms. Due to these reasons, we believe that achieving positive results while using any linear regression model will be difficult. Although these drawbacks implied that linear regression and similar models would have poor performance, these models were tried to compare against each other. For this reason, nonlinear models are chosen for predictions, such as XGBoost and SVR (Support Vector Regression with a nonlinear kernel). Both XGBoost and SVR are suitable for scenarios where the dataset isn't very large, these approaches are also applicable for complex data models that can automatically assess and select the most relevant features, thereby enhancing the model's accuracy and interpretability. SVR did not return promising predictions of the data even with hyperparameter-tuning, but XGBoost, which was thought to hold more potential due to its utilization of L1 regularization like LASSO regression, showed much more promising results after hyperparameters-tuning.

Results and Discussion

To test the selected features' predictive capabilities, we first trained a LASSO regression model on them as a baseline. Due to the shortcomings and incompatibility of LASSO regression with the characteristics of our features, the linear regression model had a mean squared error of 3.963 and an R^2 value of -0.0013. Even with hyperparameter-tuning, the model could only pick very few features. This was also observed with ElasticNet, which had a mean squared error of 3.964 and an R^2 value of -0.0016, which meant that both models were performing worse than baseline. A similar characteristic was observed with the SVR model's results, which had a mean squared error of 3.523 and an R^2 of 0.025.

XGBoost, through hyperparameter-tuning, was able to achieve a lower mean squared error of 1.035 and a higher R^2 value of 0.74319. This means that XGBoost was able to capture the correlation between features well and learn the properties of the mutated protein. To achieve these scores, hypertuning was implemented through a grid search for max_depth, learning_rate, n_estimators, colesample_bytree and alpha with the best values of 5, 0.1, 200, 0.7 and 1.5 respectively.

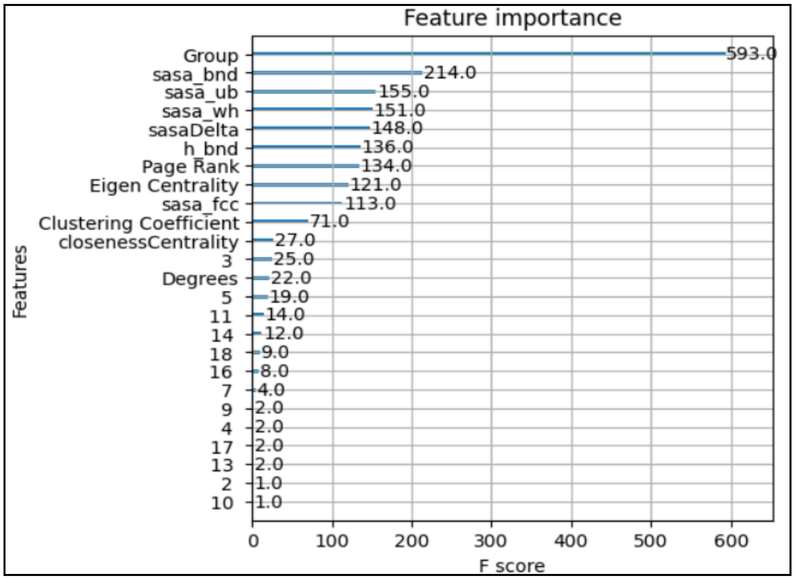


Figure 3: *Feature importance in affecting F-score. It can be seen that 3, 5 and 11; which are THR, LEU and LYS respectively, are the most prominent mutations.*

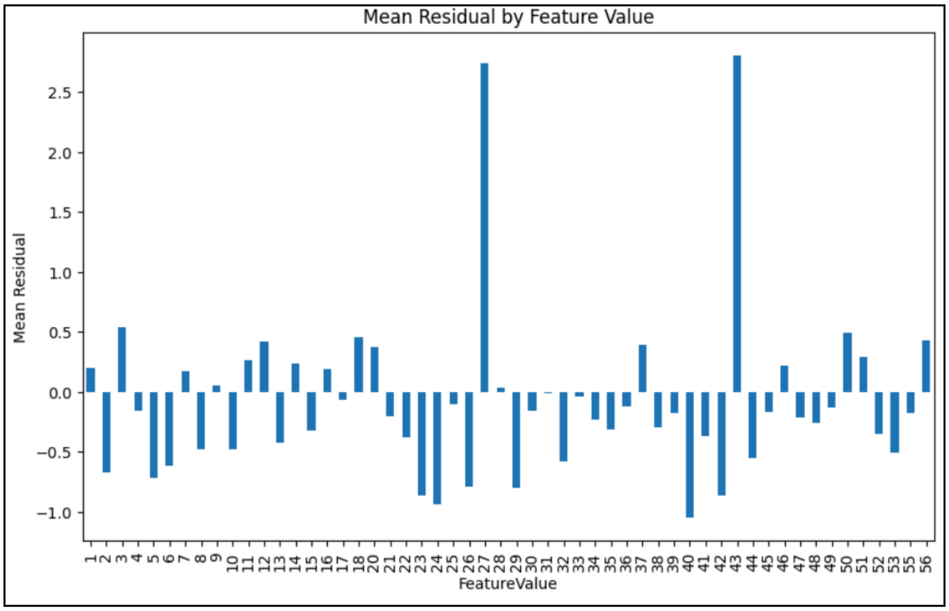


Figure 4: *This graph clarifies the challenges to predicting the binding energies of residue numbers 27 and 43; these points are the outliers. Even though 31 also shares a similar pattern in the binding energies data, the model is able to predict that outlier.*

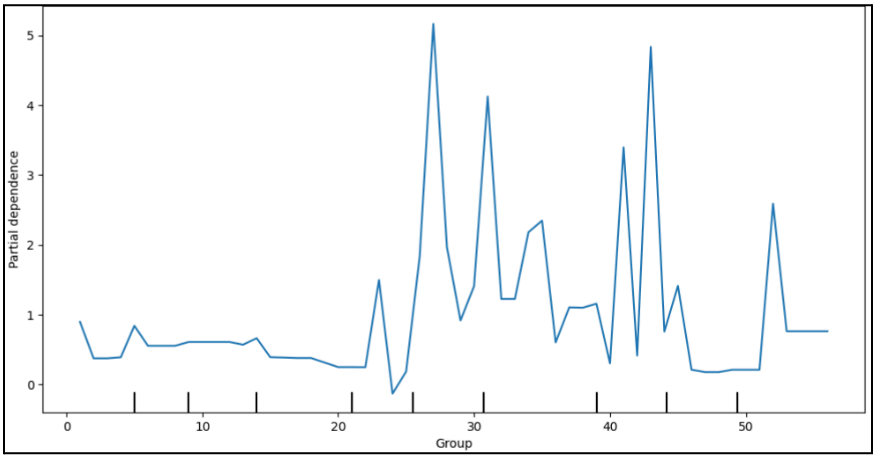


Figure 5: *This graph shows which residues are most influential in prediction. This is in line with the previous graph. The peaks indicate unfavorable interactions in this dataset. Generally, large negative ΔG values suggest a high affinity, meaning the molecules will bind tightly. Conversely, a smaller negative ΔG or a positive ΔG indicates lower affinity, suggesting weaker or non-spontaneous binding.*

Conclusions / Future Works

The LASSO model struggled due to non-independent features, significant outliers, and a lack of linear relationships between features and labels, issues that also affected the ElasticNet model. Multicollinearity further undermined LASSO's performance, as it could not handle the high dependency among features without losing interpretability. Consequently, linear regression methods, in general, were deemed inadequate due to these challenges and the presence of extreme outliers. As a result, non-linear models like XGBoost and SVR were used in this project. While SVR did not yield satisfactory results even after hyperparameter tuning, XGBoost showed more promise, benefiting from L1 regularization similar to LASSO.

After experimentation, XGBoost proved to be the most useful model for the data. The nonlinear relationship between the features and label was represented and overfitting was avoided. THR, LEU and LYS proved to be the most influential mutations along with the 27th and 43rd residues. This work is able to make a strong argument for the theory that residues are the determining factor in the prediction of binding energy.

In its current state, the study finds solid correlations but without any information regarding the coordinates of the molecules themselves. Educated conjectures imply intramolecular and intermolecular interactions to be influential in predicting the binding energy. The main assumption of this project was proven correct still; the most influential property of this molecule is the residues. This also supports the argument that the locations and interactions of certain atoms could be very important to further prediction.

Considering these inferences, further studies will focus on making predictions based on internal interactions in the protein, such as hydrogen bond interactions, electrostatic interactions, and disulfide bridges. The backbone of the protein could be key in comprehending how the protein's conformation varies across different mutations.To accomplish this, Root Mean Squared Deviation (RMSD) for these datasets may be determined by contrasting them with the native (wild type) structure, and the discrepancies in RMSD could serve as the edges between these nodes. Furthermore, creating an additional network graph focusing on the distances between Cβ atoms could aid in understanding the connections between binding and other internal interactions. The addition of the extracted features from these network graphs could be very beneficial to the prediction of the binding energies.

Appendix

Büşra Tayhan was mainly responsible for the biological explanations and expertise sections. She also contributed to data cleanup and relevant coding where necessary.

Bilgehan Çağıltay was mainly responsible for data cleanup, feature extraction, data normalization, and network analysis.

Alize Sevgi Yalçınkaya and Faraz Badali Naghadeh were mainly responsible for machine learning algorithms' applications. They also handled any feature incompatibilities that arose from this process.

While these listed tasks were the main works of each team member, all team members worked equally hard on all these tasks whenever it was necessary.

Naghadeh, F.B., Çağıltay, B., Tayhan, B., Yalçınkaya, A.S., 2023. CS512. Retrieved from https://github.com/return0ftheFaraz/CS512

References

[1] J. M. Schmiedel and B. Lehner, "Determining protein structures using deep mutagenesis," *Nature Genetics 2019 51:7*, vol. 51, no. 7, pp. 1177–1186, Jun. 2019, doi: 10.1038/s41588-019-0431-x.

[2] S. Ovchinnikov, H. Kamisetty, and D. Baker, "Robust and accurate prediction of residue-residue interactions across protein interfaces using evolutionary information," *Elife*, vol. 2014, no. 3, May 2014, doi: 10.7554/ELIFE.02030.

[3] G. Diss and B. Lehner, "The genetic landscape of a physical interaction," *Elife*, vol. 7, Apr. 2018, doi: 10.7554/ELIFE.32472.

[4] J. Otwinowski, "Biophysical Inference of Epistasis and the Effects of Mutations on Protein Stability and Function," *Mol Biol Evol*, vol. 35, no. 10, pp. 2345–2354, Oct. 2018, doi: 10.1093/MOLBEV/MSY141.

[5] D. M. Fowler and S. Fields, "Deep mutational scanning: a new style of protein science," *Nature Methods 2014 11:8*, vol. 11, no. 8, pp. 801–807, Jul. 2014, doi: 10.1038/nmeth.3027.

[6] C. A. Olson, N. C. Wu, and R. Sun, "A comprehensive biophysical description of pairwise epistasis throughout an entire protein domain," *Curr Biol*, vol. 24, no. 22, pp. 2643–2651, 2014, doi: 10.1016/J.CUB.2014.09.072.

[7] G. Ozbaykal, A. Rana Atilgan, and C. Atilgan, “In silico mutational studies of Hsp70 disclose sites with distinct functional attributes,” *Proteins*, vol. 83, no. 11, pp. 2077–2090, Nov. 2015, doi: 10.1002/PROT.24925.

[8] Hastie, T.J., Tibshirani, R.J., and Friedman, J.H. (2001). The Elements of Statistical Learning: Data Mining, Inference and Prediction. Springer, New York.