Covid- 19 Vaccine Development Through Antibody Activity Classification

Vaish Gajaraj (vg289), Meiqi Wu (mw849), Jialiang Sun (js3553), Yuwei Liu (yl3388)

1. Problem Outline

Medical researchers working on the covid-19 vaccine have a number of hurdles to overcome when addressing this new strain of the virus. The virus, which has decimated the lives of millions, has been fully sequenced by an international team of researchers. Now we as data scientists can discover the parts of the virus most likely to lend itself towards vaccination trials of these viruses using statistical learning methods.

2. Introduction to Cell Biology and Problem

The following discussion relies heavily on a background understanding of cell biology. For the ease of the reader, we have defined terms unfamiliar and provide a glossary below:

Antigen: A toxin, virus or other foreign substance that induces an immune response in the body, like antibody production. These are made up of a number of proteins[3].

B Cell: White blood cells that fight bacteria and viruses by making Y-shaped proteins called antibodies, which are specific to each pathogen and are able to lock onto the surface of an invading cell and mark it for destruction by other immune cells[16].

Antibody: Large, Y-shaped protein produced mainly by plasma cells that is used by the immune system to neutralize pathogens such as pathogenic bacteria and viruses[2].

Epitope: portion of a foreign protein, or antigen, that is capable of stimulating an immune response. An epitope is a section of the antigen that binds to a specific antigen receptor on the surface of a B cell. After binding to the B cell (like a puzzle piece), the body creates an immune response (i.e creates antibodies) to remove it[8].

Protein (Peptide) Sequence: Viruses are made up of proteins, and these proteins are made up of amino acids that fold into itself to create complex structures. The sequence of a protein refers to the unique ordering of amino acids that make up the protein. Each amino acid is unique and corresponds to one of twenty different encoded values, labeled with some capitalized roman letters. A smaller segment of a protein is a "peptide" [14].

Through analyzing the Covid 19's protein sequence, we seek to discover the parts of the virus that stimulate an immune response from the body (the parts with epitopes). Researchers can then separate these sections from the larger virus and administer them in the form of a vaccine. When delivered via a vaccine, these regions bind to our B cells and train our bodies to fight the foreign infection by producing antibodies.

A successful vaccine would produce large amounts of antigen-specific antibodies. In this case, those antibodies would prevent a full covid infection from ravaging our immune systems. From this data set, we will classify the sequences that are most likely epitopes (i.e. will produce antibodies in our bodies). With this antibody classification, virus researchers can further experimentally verify if they work in a vaccine.

To understand how we approached this problem, it is important to understand what is gained from a protein sequence. When studying the structure of cells or viruses, researchers gain a number of features correlated with antibody production. This includes a string of characters corresponding to the amino acids that make up the protein. Researchers also compile a list of tertiary characteristics of the protein that relate to unique interactions and properties that originate from the chemistry of the protein segment. These characteristics make up the bulk of features in our datasets. They include measures like hydrophobicity (how much the protein repels water) and aromaticity (how cyclic the chemical bonds are). The list of features available are further discussed later.

Finally, each row of the protein sequence in our training and validation sets (i.e. data researchers have studied intensely before) has a binary target value representing antibody

affinity. This is the key metric researchers use to identify the likelihood of a particular protein sequence binding to an epitope (which helps us fight off a virus through producing antibodies). Our model will learn the nuances of the protein features and try to predict this binary value of whether this sequence will bind to an epitope or not. In other words, we will use data to determine whether a certain segment of protein will induce human antibody responses. In biological literature, there are degrees to antibody affinity, with some protein sequences generating higher levels of activity than others. However, the authors of this dataset have confirmed that for the purposes of rapid experimentation, these distinctions were replaced with a binary system [13].

The authors of the data set provide a pre-sequenced and labeled B-Cells and SARS (a similar virus to Covid-19) proteins. Using datasets of protein sequences with known antibody activity levels, we can create models to predict the antibody affinity of Covid protein sequences. The sequences which provide a binary classification of 1 would be the most likely candidates for a covid-19 vaccine trial, as they have the ability to induce antibody activity with other proteins, most importantly our B-cells. One might conclude that this problem then simply comes down to creating an accurate and precise classification model for our Covid data. Therefore our goal is to develop a model with a reasonably high accuracy rate and F1 statistic such that we are not overfitting.

3. Project Importance

2020 has proven to be one of the most challenging years in modern history. As the virus spread across the world, hundreds of thousands of people have perished and millions of lives have been upended. As of mid-December, the world approaches nearly 70 million confirmed cases with 1.57 million deaths, reported by the CDC [5].

Since the first breakout of the notorious pandemic, there has been a worldwide shut down of companies, factories and schools, as well as a substantial negative impact on the global economy. The shortage of medical resources and test kits once led to a huge increase in daily confirmed cases and massive panic among citizens.

Currently, many countries, including the U.S., are still fighting hard to control COVID-19. Humanity's only defense at the moment remains largely in social distancing and sanitization efforts, as only recently a successful vaccine has been released. Developers of the vaccine have used countless methods to increase their experimentation capacity, including a number of computational and statistical methods similar to ours. The effective distribution of a vaccine and continued monitoring of the virus will be the key to ending the COVID-19 pandemic.

4. Data Description and Visualization

The files "input_bcell.csv" and "input_sars.csv" contain 14387 and 520 segments of B-Cells and SARS protein sequences respectively. These two data sets contain target labels for antibody affinities. To build a comprehensive model of protein antibody activity, we combine these two datasets into a larger set, which we can split into training, validation and testing sets.

The file input_covid.csv contains only features but not labels, and our goal is to predict the antibody affinity label for data points in input_covid.csv. Below are the features with short descriptions:

parent_protein_id: ID of parent protein.
protein_seq: Amino acid sequence of parent protein.
start_position: Start position of the peptide in the parent protein.
end_position: End position of the peptide in the parent protein.
peptide_seq: Sequence of peptide segmented by above two parameters.
chou_fasman: the probability that a given sequence of amino acids will fold back onto itself and create a more complex structure [7].
emini: A feature of the peptide describing relative surface area accessible by a solvent like water [4].
kolaskar_tongaokar: A feature of the peptide derived from the chemical properties of the amino acid measuring the capacity for the peptide to bind to an antigen[4].

parker: A feature of the peptide describing hydrophobicity (how much it repels water)[9].
isoelectric_point: A feature of the entire protein, referring to the pH at which the molecule carries no net electrical charge[9].
aromaticity: A feature of the protein, measuring how cyclical the molecular structure appears[1].
hydrophobicity: A feature of the protein, measuring its repulsion to water[11].
stability: A feature of the protein measuring the likelihood a protein will denature (unfolded or breakdown) or retain its shape[15].

Of the 13 features, we drop three: parent_protein_id, protein_seq and peptide_seq. Exploiting these textual representations of peptide sequences require techniques beyond the scope of our abilities.

Analyzing this data, we find no missing values. We also dropped duplicate records and got a final combined dataset of 14896 records. Our dataset has 761 unique parent proteins and 14841 unique peptides, which means that there are a number of combinations of protein and peptides with different features. In the set, we found 4032 records of peptides with antibody activity (= 1) against 10864 non active records (= 0). The percentage of antibody activity in our dataset is therefore 27.07%.

5. Exploratory Data Analysis

Of our 10 features, we explored the differences between features that have an antibody valence and those without an antibody valence. We consider both features pertaining to the smaller peptide segment and those from the full parent protein. We visualized the differences between each feature's distribution and compared the mean of each feature from the target and non-target groups. One of these findings is presented in Figure 1. This figure demonstrates that the isoelectric pH for antibody active peptides is lower than those without antibody activity. To simplify readability, more of these findings can be found in the "EDA.ipynb" file in our GitHub. The important point is that this analysis demonstrates the difference between the peptides with antibody activity and those without. Our statistical models will hopefully be able to take advantage of these differences.

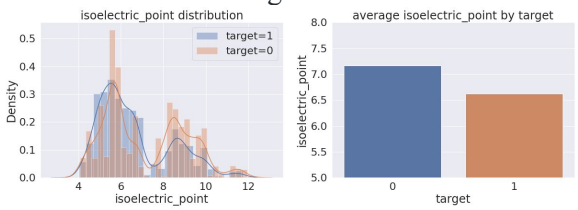

Figure 1. Isoelectric point Distribution

In addition, we checked the statistical correlation between the features to educate our usage of linear models. To do this we created a correlation matrix:

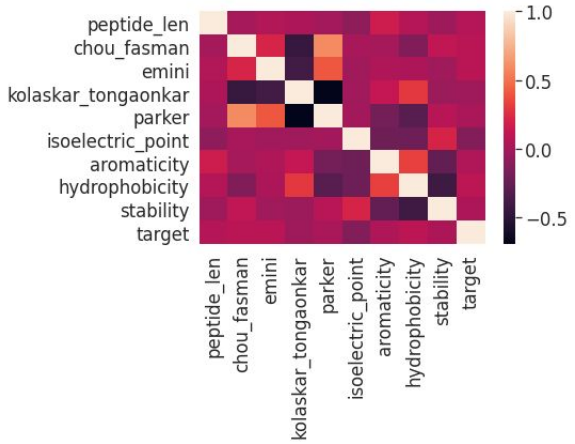

Figure 2. Correlation Matrix of Features

This matrix shows us that for the most part, our independent features are not highly positively linearly correlated with each other. This informs our decision to pick a logistic regression model first to model our data.

6. Model Selection

6.1 Cross Validation and Training

Before we train our data, we randomly select 20% for our testing set. Then, we use a k-fold cross validation schema to train our model. In k-fold cross-validation, the original data is randomly shuffled and split into k equal sized subdivisions (folds). One of these subdivisions is kept as the validation set for testing each k-fold. The other subdivisions are used as training data. The cross-validation then repeats a total of k times, each time swapping for a new validation set. The k results are then averaged into a single result. This method ensures that all of the data is used once for training and validation, such that

we ensure we pick the best model among the data[10]. Furthermore, this procedure reduces bias and can help in detecting whether we are overfitting or underfitting by providing a list of validation dataset losses and accuracies.
Our training scheme also calculates a number of useful metrics like accuracy, precision, recall and F1 score. The formula for these values is given below:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}$$

Where FP is a false positive (incorrectly classified as 1), TP is a correctly classified row, FN is a false negative and TN is a correctly predicted negative reading (classified as zero). These scores allow us to evaluate how good our models are at correctly discovering peptides with antibody activity.

6.2 Logistic Regression

The first model applied to our data was the logistic regression model. We took advantage of the binary logistic regression model, given an output space of either {0,1}. In addition, we applied L2 regularization, L1 regularization, and no regularization to avoid the risk of overfitting. The mathematical formula for these methods follows; the first describing general logistic regression, the second with L2 regularization and the third with L1 regularization:

$$\text{minimize} \sum_{i=1}^{n} \ell\left(x_i, y_i; w\right) + r(w)$$

$$\text{minimize} \sum_{i=1}^{n} \left(y_i - w^T x_i\right)^2 + \lambda \sum_{i=1}^{n} w_i^2$$

$$\text{minimize} \sum_{i=1}^{n} \left(y_i - w^T x_i\right)^2 + \lambda \sum_{i=1}^{n} |w_i|$$

From [Table 1] L2 regularization shows the model performance. For each variant, we collected accuracies, precisions, recall, and f1 scores for the L2, L1, and no regularization as provided by our cross validation [Table 1].

Logistic Regression	Accuracy	Precision	Recall	F1
w/ L2	0.731101	0.577338	0.064951	0.116734
w/ L1	0.730909	0.574128	0.064601	0.116102
w/ no reg.	0.728610	0.534691	0.065826	0.117179

Table 1. Testing Dataset Against Logistic Regularization Methods

From this analysis, we conclude that L2 regularization provides the greatest benefit and minimizes our overfitting the most. However, the low recall and low F1 scores tell us our model struggles with classifying a multitude of peptide sequences. In other words, we are missing on positively classifying a number of peptides that would work in our vaccine experiments but not identifying them. This suggests our model is actually underfitting or not properly modeling the data. We can further observe the shortcomings of the logistic regression model by viewing its feature importances [Figure 3].

The model weights the start and end positions of the peptide sequence the greatest and does not consider the protein markers greatly, which from a biological standpoint does not make much sense. Therefore, we turn to other models.

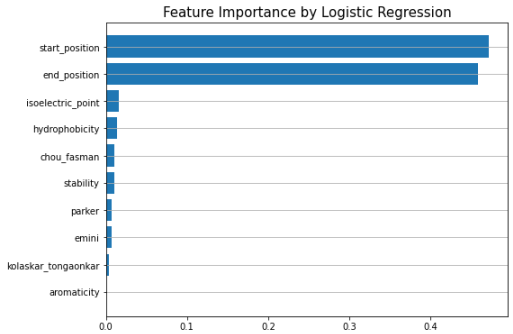

Figure 3. Logistic Feature Importances

6.3 Random Forest

The next model applied is a random forest model, which relies on a number of relatively uncorrelated decision trees working together in order to produce a classification output. This method, described in section 5, works by first drawing a bootstrap sample from the training dataset. The tree is then developed by recursively selecting m variables at random from a set of p variables. After selecting the best 'split point' for the decision tree among the m variables, you can create two nodes in the tree. This continues until you output an "ensemble" of trees at a max predetermined height for each tree. Below is a image of how these models work:

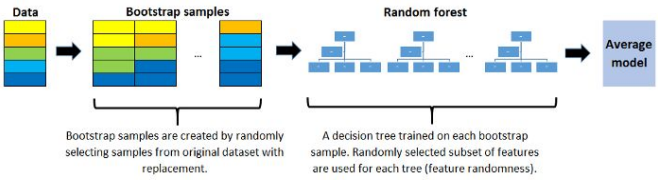

Figure 4. Pictorial Representation of Random Forest Model with max depth of 2[17]

Finding this predetermined height is the key to developing a random forest model that best represents the data. To accomplish this, we applied a grid search method to which searches a list of predefined heights and measures each height's performance using the negative log loss function, as discussed in the regularization lecture. The negative log loss provides a score for which the grid search can evaluate performance on the data set. This hyperparameter optimization method revealed after testing the depths 16,17,18, and 19, that a tree depth of 18 provides the greatest model accuracy. The results of the model on our testing data can be found in [Table 2].

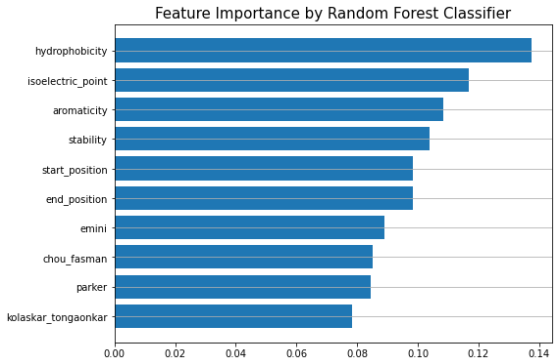

Figure 5. Random Forest Feature Importance

6.4 Gradient Boosted Trees

Gradient Boosting Model is an ensemble method where shallow trees are built sequentially, each trying to correct the errors (residuals) of its predecessor. To calculate these errors, the model we implemented uses a negative log likelihood function for the binary classification case:

$$\ell(y) = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot \log\left(p\left(y_i\right)\right) + \left(1 - y_i\right) \cdot \log\left(1 - p\left(y_i\right)\right)$$

In this formula, the $p(y)$ is the probability of a predicted non-zero label for all N rows of data, and each y_i represents the true label. This loss function provides a measure of uncertainty which each tree iteration tries to improve on.

This works in a similar recursive fashion to the random forest model, but with the added goal of reducing errors on each recursive step (this is what's known as boosting) [12]. Below is a clear pictorial explanation of the method:

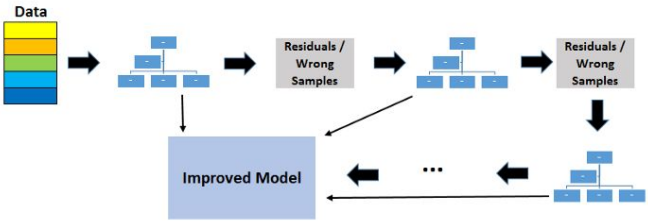

Figure 6. Pictorial Representation of Gradient Boosted Trees with tree depth of 2[17]

This tree-based model often provides higher accuracy to the Random Forest model and trains faster, so we applied it under the possibility it could improve results over the Random forest. We applied a similar discrete grid search

Model Type	Accuracy	Precision	Recall	F1
Random Forest	0.849717	0.782149	0.624825	0.694678
Gradient Boosting	0.836878	0.759870	0.590511	0.664550
Neural Network	0.7854	0.6102	0.6005	0.6049

Table 2. Performance for Three Additional Models

hyperparameter optimization, this time finding an optimal tree depth of 6. The results of this analysis can be found in [Table 2]. Although both models have similar accuracies, in the context of this problem false positives are more costly. The selection of useful peptides is of great importance for vaccine development, so we want to pick the model that outputs a smaller false positive rate. Random Forest is better (a rate of 0.218 vs. 0.239 for Gradient Boosting Model). In terms of false-negative rates, Random Forest (0.131) also performs better than Gradient Boosting Model (0.142). This can also be seen by simply looking at the F1 score of the random forest model, which is higher than that of the gradient booster model. Therefore, we believe Random Forest provides a better model.
When taking a look into feature importances for the gradient boosting model [Figure 4], it's highly similar to the results for the random forest model, which is to be expected given they are both tree-based models.

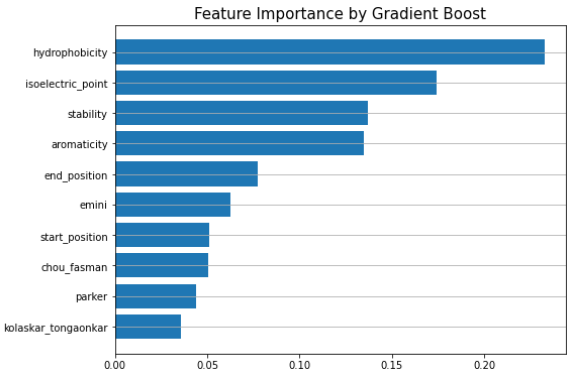

Figure 5. Gradient Boosting Feature Importance

6.5 Dense Neural Networks

We also experimented with modeling out data using a "Dense Neural Network". This variant of neural networks was developed by former Cornell faculty Gao Huang, et. al. Neural Networks are a deep learning framework that mimics human neuron behavior to map a set of labeled inputs to a set of labeled outputs. They descend from the original perceptron schema, however utilize a number of techniques to increase performance including improving each iteration via a log likelihood loss [6]. Our neural network consists of a number of interconnected models that are organized in layers, which "feed forward" learned models of the input space. First it introduces the protein data as an input layer, then applies non linear transformations to represent this data in a number of intermediary (hidden) layers, and returns labels as an output. The dense neural network also corrects every layer ahead of it, improving the model function based on errors calculated through a negative log likelihood function like the gradient boosting method[9]. This process is more clearly illustrated in the diagram below:

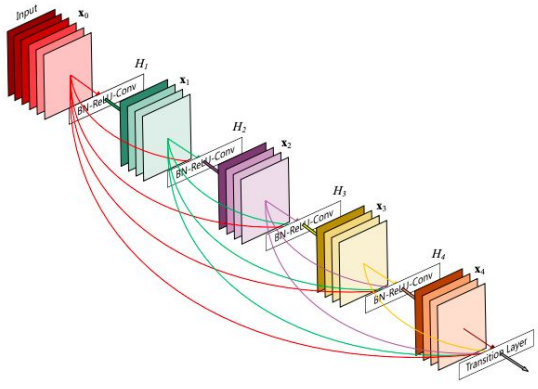

Figure 6. Dense Neural Network Framework[9]

Neural networks showed promise in that our data might contain hidden relationships between features that cannot be captured by only tree-based models. The model performed worse than tree-based models [Table 2], which may be due to the low number of features in the data. Neural networks perform better when presented with a massive number of features and data points. In addition, neural networks are difficult to interpret since hidden layers often don't convey useful information. Therefore, we decided to choose Random Forest as our main model.

6.6 Application of Tree-Based models to Covid data

After conducting this model comparison analysis, we apply the random forest model to output target labels for the covid data set. The results of this are found below in Figure 6:

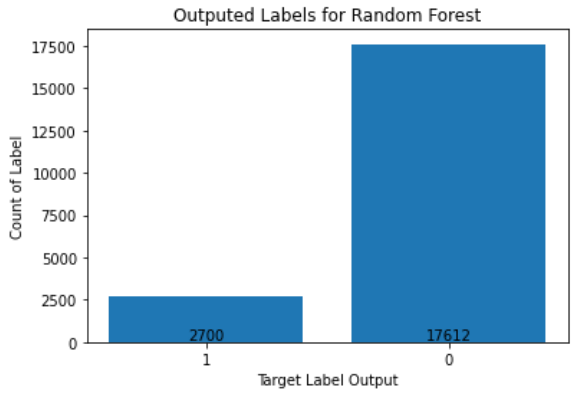

Figure 7. Outputted Labels for Random Forest

From this graph, we see that from a total sample space of 20312 labels, we predict 2700 peptide sequences that could contribute to a potential Covid-19 vaccination trial. These predicted labels should be carefully considered against the approximately 84% accuracy rate of the random forest model.

Because the gradient boosting method provides such a similar accuracy, we also look into this method's predictions. Below we see this graph, with 9907 antibody producing peptides predicted against the 20312 samples:

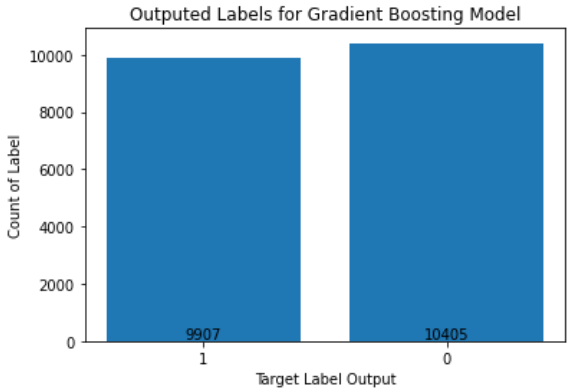

Figure 8. Predicted Labels from Gradient Boost

There is a massive difference here between the results of the random forest model and the gradient boosting model despite the relatively small differences between these models. For researchers working on vaccine development, the random forest is a much better model despite providing fewer peptides with positive antibody activity, as it is important to only try peptides that are the most likely to predict true positives (the random forest).

7. Discussion

Considering that our data collection process and research result will not cause any impact on the COVID-19 virus itself or any of its behavior, our models do not directly produce a Weapon of Math Destruction. Our outcomes are easily measurable by biologists and biochemists. It's possible our model is not accurate enough and harms the vaccine development effort, but either way it is critical that biologists verify our findings. Researchers who use these types of models to speed up their experimentation process understand the limitations of statistical models. They must apply a healthy amount of skepticism when a model makes the claim that a certain peptide could be useful in a vaccine trial. Certainly, they should not use these findings in any human trials before extensive verification.

In our research problem, there are no such obvious human-applicable grouping criteria for the Covid 19 virus. Therefore we are confident there is a slim chance our models lead to prejudice or discrimination. Hence, we did not assign any protected attributes in our models.

Future explorations of the antibody valence problem should include more than 10 features to better capture the complexity of antibody valence. For example, in our dataset we did not include the "protein_seq" and "peptide_seq" features due to the difficulty of transforming these textual representations of peptides into useful information. A future model might improve on our performance through correctly adding these features.

Another limitation of our model lies in the black box nature of the random forest model. Our model does not provide biologists a simple explanation of how it calculates antibody valence, and it very well may be doing something biologically nonsensical. Random forest's black box is not sufficient to determine this, nor ascertain a theory for the biological phenomena.

8. Conclusion

In this project, we build our COVID-19 virus antibody valence prediction model based on protein analysis of B cell proteins and the SARS virus proteins. These data sets allowed us to develop models that represented the antibody valence of peptide segments. We then applied these models to covid data to find the segments of the covid virus protein that create antibodies in humans during a vaccine trial.

After trying four different models, we found the performance of the random forest the best. With an 84.9% accuracy, our model identified a total of 2700 potential protein segments in the Covid virus with which to test a vaccine with.

In the random forest model, hydrophobicity, isoelectric_point, and aromaticity present the highest feature importances, namely that the three protein features are the most important ones to determine the antibody valence in Bcell, SARS and COVID-19 among the 10 features.

There remains much room for improvement of our modeling. The random forest is subject to limitations like limited features and poor interpretability. Adding more features like the protein_seq and peptide_seq, as well as exploring other more interpretable models is a promising direction for development.

We would like to acknowledge the efforts of Dr. Noumi, et.al. for making the data set used in this analysis publicly available via Kaggle. Their insights provided us with a strong introduction with which to tackle this problem.

We would further like to acknowledge the breadth of experience gained from the course *Big Messy Data* (ORIE 4741). In our project, we used multiple techniques learned from the class. First, we applied cross validation in modeling to ensure the best use of each data point, and to tackle variance bias tradeoff. We also applied logistic regression utilizing a number of regularization methods from class to minimize overfitting. Also, our use of random forest was inspired by section 5's lecture. as our baseline and main models. For our tree based models, random forest and gradient boosting, we used a grid search hyperparameter optimization method to solve for optimal tree height parameters.

9. References

[1] Anjana, R., Vaishnavi, M. K., Sherlin, D., Kumar, S. P., Naveen, K., Kanth, P. S., & Sekar, K. (2012). Aromatic-aromatic interactions in structures of proteins and protein-DNA complexes: A study based on orientation and distance. *Bioinformation*, *8*(24), 1220–1224. https://doi.org/10.6026/97320630081220

[2] *Antibody*. (n.d.). Genome.Gov. Retrieved December 13, 2020, from https://www.genome.gov/genetics-glossary/Antibody

[3] *Antigen: Medlineplus medical encyclopedia*. (n.d.). Retrieved December 13, 2020, from https://medlineplus.gov/ency/article/002224.htm

[4] *B cell help*. (n.d.). Retrieved December 13, 2020, from http://tools.iedb.org/bcell/help/

[5] CDC. (2020, March 28). *Covid-19 cases, deaths, and trends in the us | cdc covid data tracker*. Centers for Disease Control and Prevention. https://covid.cdc.gov/covid-data-tracker

[6] Georgevici, A. I., & Terblanche, M. (2019). Neural networks and deep learning: A brief introduction.

Intensive Care Medicine, 45(5), 712–714. https://doi.org/10.1007/s00134-019-05537-w

[7] Chou, P. Y., & Fasman, G. D. (1974). Prediction of protein conformation. *Biochemistry*, *13*(2), 222–245. https://doi.org/10.1021/bi00699a002

[8] *Epitope—An overview | sciencedirect topics*. (n.d.). Retrieved December 13, 2020, from https://www.sciencedirect.com/topics/neuroscience/epitope

[9] Huang, G., Liu, Z., van der Maaten, L., & Weinberger, K. Q. (2018). Densely connected convolutional networks. ArXiv:1608.06993 [Cs]. http://arxiv.org/abs/1608.06993

[10] Hastie, T., Tibshirani, R., & Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed). Springer.

[11]Law, K.-Y. (2014). Definitions for hydrophilicity, hydrophobicity, and superhydrophobicity: Getting the basics right. *The Journal of Physical Chemistry Letters*, *5*(4), 686–688. https://doi.org/10.1021/jz402762h

[12] Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. *Frontiers in Neurorobotics*, *7*. https://doi.org/10.3389/fnbot.2013.00021

[13] Noumi, T., Inoue, S., Fujita, H., Sadamitsu, K., Sakaguchi, M., Tenma, A., & Nakagami, H. (2020). Epitope prediction of antigen protein using attention-based lstm network. *BioRxiv*, 2020.07.27.224121. https://doi.org/10.1101/2020.07.27.224121

[14] *Peptide*. (n.d.). Genome.Gov. Retrieved December 13, 2020, from https://www.genome.gov/genetics-glossary/Peptide

[15] *Protein stability—An overview | sciencedirect topics*. (n.d.). Retrieved December 13, 2020, from https://www.sciencedirect.com/topics/biochemistry-genetics-and-molecular-biology/protein-stability#:~:text=This%20chapter%20describes%20protein%20stability,between%20two%20large%20opposing%20forces.

[16] *What's the difference between b-cells and t-cells?* (2018, December 24). Cancer Treatment Centers of America. https://www.cancercenter.com/community/blog/2017/05/whats-the-difference-b-cells-and-t-cells

[17] Yıldırım, S. (2020, February 17). *Gradient boosted decision trees-explained*. Medium. https://towardsdatascience.com/gradient-boosted-decision-trees-explained-9259bd8205af