ORIE5741 Final Report

Karina Fang, Kornelia Wang, Pengbo Hao

December 5, 2021

1 Introduction

In the year 2020, the world suffered from an unprecedented pandemic COVID-19. Not only has the COVID-19 caused a tremendous amount of death worldwide and significant losses in economies, but this contagious disease is also ever-changing, developing new strains and variants rapidly. Although many methods, including quarantine and testing, have been adopted to prevent the spread of the disease, it can be costly for society if the administration were to perform quarantine and testing on a large scale. Moreover, the number of testing kits is limited and may be distributed unevenly across the country. To circumvent these constraints, it is vital to answer the question: can we predict the possibility of testing positive given the characteristics of a certain patient. By building a model using the data available, we will be able to prioritize the individuals in need and allocate the testing kits more efficiently.

2 Data

2.1 Data Descirption

For this project, we use the Aggregated US survey data made available to the public by CMU. This dataset is collected by Facebook, and sampled users receive the invitation at the top of their News Feed and the surveys are conducted off the Facebook app. Every day, a new sample of Facebook users over 18 years old are invited to consent and participate in the survey and self-report their COVID-19-related symptoms. Using this survey data, we estimate the percentage of people who have a COVID-like illness, in a given location, on a given day. All numerical data points are adjusted to be the 7 day trailing average (today, and the 6 previous days).

The data originally has 162691 rows and 105 columns, containing 8 dimension columns and 50 signals. Among all the columns, the testing result columns, 35, 36, 37, 38 and 39, are our labels; the rest of the columns are our features.

2.2 Data Cleaning

We generate a data matrix, use column 35 as a target vector, and create a train/test split so that we can empirically test for overfitting. To make the 80 / 20 train test split, we shuffle the data and select the first 80% as the train data, with 20% held out for validation.

2.3 Feature Selection

For feature selection, first we consider numerical variables that might be useful for predicting the results: Estimated percentage of people reporting that they have experienced COVID-19-related symptoms. These symptoms include fever, cough, short of breath, difficulty breathing, tiredness or exhaustion, nasal congestion, a runny nose, muscle or joint aches, a sore throat, persistent pain or pressure in their chest, nausea or vomiting, diarrhea, loss of smell or taste, etc.

We also consider categorical variables: gender and age_bucket, as these variables are also informative. Since possible values of gender and age_bucket also contain “overall” (aggregation of all genders), we delete these rows to avoid duplication. Then we transform the data into one-hot vectors and concatenate those features with numerical data we had previously.

We only took the examples for which the result of the estimated percentage of people reporting that they have tested positive for COVID-19 is known into consideration. Since this survey item is only presented to respondents who report they are currently experiencing symptoms, most responses are blank.

After data cleaning, we have 45381 rows x 100 columns in our data matrix. No data is missing in the feature matrix.

3 Data Visulization

Data visualization is an important step to identify the important features that are useful in building the model and making predictions. We first visualized the correlation between different features to gain a better understanding of how different features might influence the probability of testing positive of a patient. Then, by plotting histograms of selected important features, we visualized the underlying distribution of selected features, which gave rise to the construction of the model.

3.1 Correlation Visualization

As we can observe in Figure 1.1, the dependent variable "smoothed_pct_tested_and_positive" is highly correlated to feature "smoothed_pct_self_fever" (7-day trailing average of the estimated percentage of people reporting that they have experienced a fever in the past 24 hours), "smoothed_pct_self_anosmia_ageusia"(7-day trailing average of the estimated percentage of people reporting that they have experienced the loss of smell or taste in the past 24 hours). Our dependent variable is also observed to be weakly correlated to feature "smoothed_pct_self_difficulty_breathing" (7-day trailing average of the estimated percentage of people reporting that they have experienced difficulty breathing in the past 24 hours), and "smoothed_pct_self_nausea_vomiting" (7-day trailing average of the estimated percentage of people reporting that they have experienced nausea or vomiting in the past 24 hours). All four features are aligned with the common belief of the symptoms of COVID-19. To better understand the data, we further plotted the histogram of each feature.

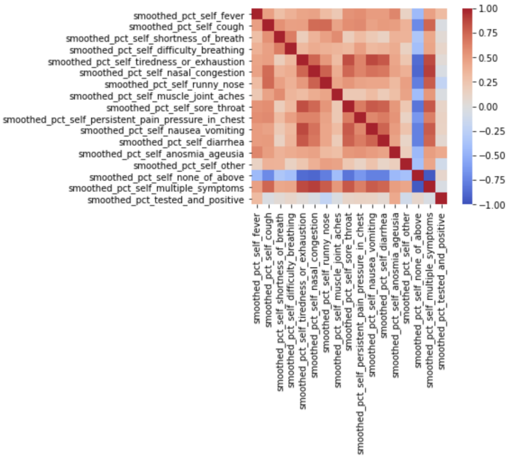


Figure 1.1

Figure 1: correlation visualization

3.2 Histogram Visualization

Before plotting the histogram, we first provide the summary data for the four features of interest in Table 1.1.

Then, we plotted the histogram for each feature of interest and observed the underlying distribution of the data. As we can observe in Figure 1.2, while there are a significant amount of people reporting no sign of fever in the past 24 hours, the remaining percentage of people reporting a fever is approximately log normal distributed with mean around 0.8%. Similarly, from Figure 1.3, there are a moderately large amount of people reporting no experience of loss of smell or taste in the past 24 hours, the rest of the data suggesting an approximate log normal distribution with mean around 1.2%. However, for the features that are weakly correlated to the dependent variable, as suggested in Figure 1.4 and Figure 1.5, the distribution of the data remains unclear. In order to have a better understanding of the data, we conducted preliminary analyses on the data utilizing linear regression.

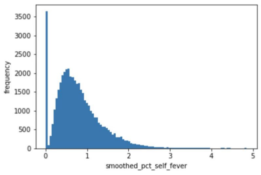


Figure 1.2

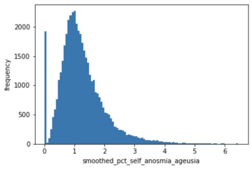
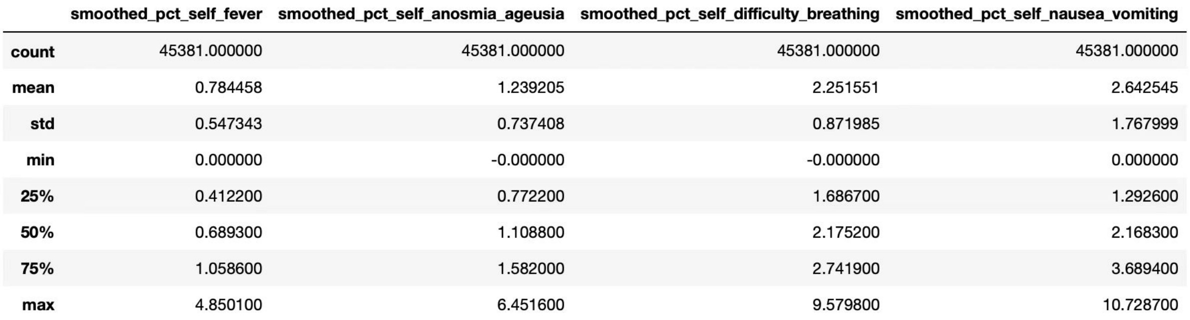


Figure 1.3



	smoothed_pct_self_fever	smoothed_pct_self_anosmia_ageusia	smoothed_pct_self_difficulty_breathing	smoothed_pct_self_nausea_vomiting
count	45381.000000	45381.000000	45381.000000	45381.000000
mean	0.784458	1.239205	2.251551	2.642545
std	0.547343	0.737408	0.871985	1.767999
min	0.000000	-0.000000	-0.000000	0.000000
25%	0.412200	0.772200	1.686700	1.292600
50%	0.689300	1.108800	2.175200	2.168300
75%	1.058600	1.582000	2.741900	3.689400
max	4.850100	6.451600	9.579800	10.728700

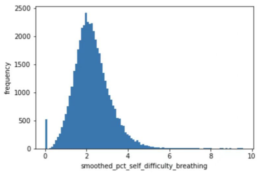


Figure 1.4

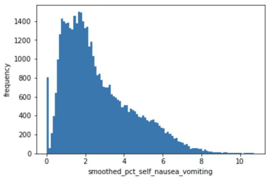


Figure 1.5

4 Model Selection

4.1 Quadratic Loss

We first construct a linear model with only quadratic loss without any regularizations, and we use this model as a basic model for comparisons with other potential models we fit later. The model uses least squares methods applied using linear regression theory, which is based on the quadratic loss function

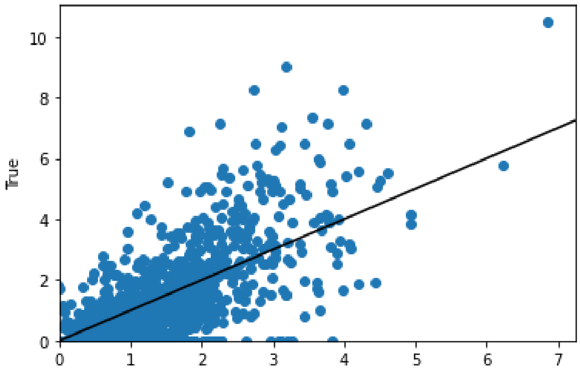


Figure 1.6 Predicted

After fitting the linear model, we also computed the MSE for both training and testing data. We then run the OLS function and bypass any SVD convergence errors by refitting the model. Figure 1.6 shows our linear prediction, with a training MSE of 1.4434 and a testing MSE of 1.4465. The training MSE and testing MSE are similar with each other and they are low based on our original dataset.

We then used five-fold cross-validation to retrain our data. Cross-validation is a statistical method used to estimate the skill of machine learning models. It is commonly used in applied machine learning to compare and select a model for a given predictive modeling problem because it is easy to implement and results in skill estimates that generally have a lower bias than other methods. We then shuffled our data and split it into training-test-validation sets and rerun the process of linear regression. This method allows us to lower the bias in the dataset. After applying the cross-validation, we get a score of 0.372 and the average mean squared error of 1.4452.

To evaluate the result of the simple linear regression model, we need to add regularization into our model. Therefore, we tried two regularizations, Quadratic Loss + l1 model and Quadratic Loss + l2 model to compare with the current regression model.

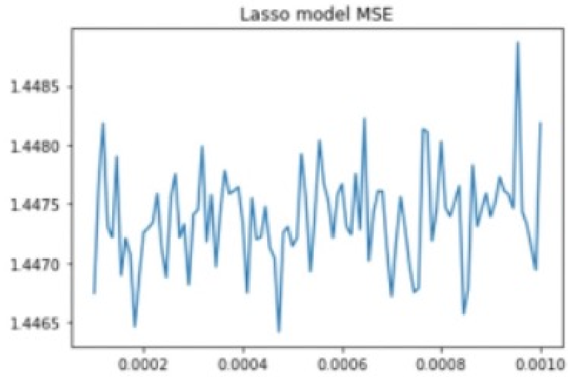
4.1.1 Quadratic Loss + l1 model

In this model, we add a regularizer in the loss function to conduct a Lasso regression model on our datasets. In this way, the model is guaranteed to produce a unique solution and prevent overfitting. What is more, with the Lasso model, we penalize large w less than quadratic regularization. The equations are as follows:

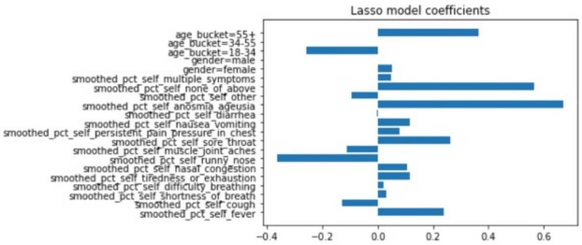
$$minimize \sum_{i=1}^{n}(y_i - w^T x_i)^2 + \lambda \sum_{i=1}^{n} |w_i|$$

We also conduct 5-folds cross-validation in the train datasets in order to use all observations for both training and validating. The MSE for each fold is averaged to represent the average error of the model type.

By adding l1 regularizer, the model gives us a unique solution. The following graph shows the model's mean squared error using 5-fold cross-validation given different lambda values.



As the graph indicates, the minimum Lasso model MSE is around 1.4463, and the value of the best lambda is around 0.0005. Using the optimal parameter to fit the test sets, the coefficients are computed as:



As expected, the Lasso model produces a more sparse solution. The coefficient plot shows that the model eliminates some of the original variables.

For the numerical variables, anosmia and ageusia contribute most positively to the positive results, and runny nose contributes most negatively to the results. For the categorical variables, coefficients of age buckets are also interesting to discuss. For 55+ people, there is a positive coefficient. For people in 18-34, the coefficient is negative.

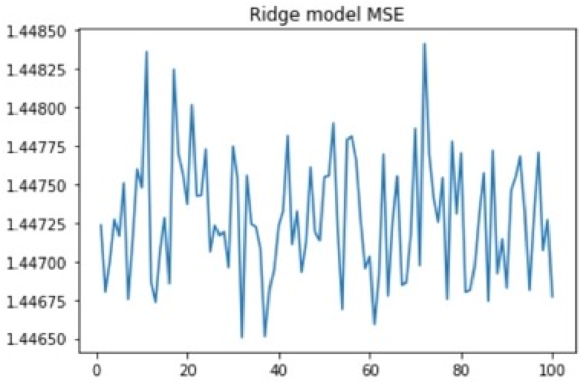
4.1.2 Quadratic Loss + l2 model

In this model, instead of l1 we pick l2 norm as the regularizer. The model is also known as the Ridge model and we are solving the following equation:

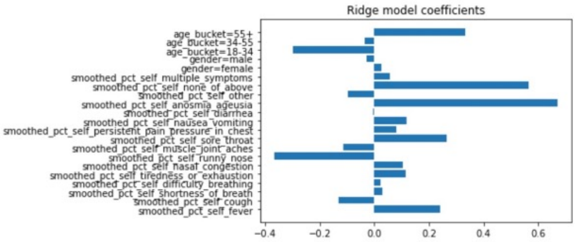
$$minimize \sum_{i=1}^{n}(y_i - w^T x_i)^2 + \lambda \sum_{i=1}^{n} w_i^2$$

Compared to Lasso, Rider does not produce spares solutions, but put rather equal weight on each entry of w. This model is also validated through 5-folds cross-validation method.

The following graph shows the model's mean squared error using 5-fold cross-validation given different lambda values.



After applying the cross-validation, we found the optimal lambda is around 32 and it gives a MSE around 1.4465, which is a bit bigger than the Lasso model. The coefficients fitting on test sets are computed as:



Compared with the Lasso model, in the Ridge model none of the variables are eliminated, age bucket in 34-55 and gender=male now have nonzero coefficients, which correspond to our discussion in sparsity. Similar to Lasso, Ridge also indicates that anosmia and ageusia symptoms have the largest positive coefficient, and runny nose symptom has negative coefficient.

4.1.3 Conclusion From Quadratic Model

For both models, anosmia and ageusia contribute most positively to the positive results, and runny nose contributes most negatively to the results. In terms of the categorical variables, for 55+ people, there is a positive coefficient, but for people in 18-34, the coefficient is negative. Therefore, we could know that the loss of the senses of smell (anosmia) and taste (ageusia) is the most useful method to predict whether a person has COVID or not, and symptoms like runny nose may not be a good indicator for predicting COVID positive possibilities. What is more, younger people have less possibility to get COVID while older people have higher risk. In terms of accuracy, it indicates how confident we are using these models to make predictions. For Quadratic Regression models with regularizations, the MSE of the Lasso and the Ridge are all less than 1.5, which shows our models have a relatively small test error. We are confident in using these models for future predictions and computations.

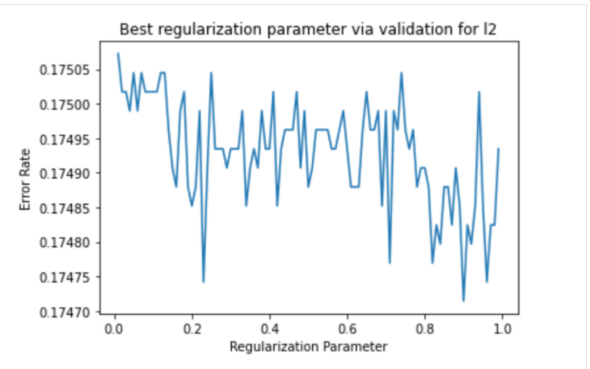
4.2 Logistic Regression

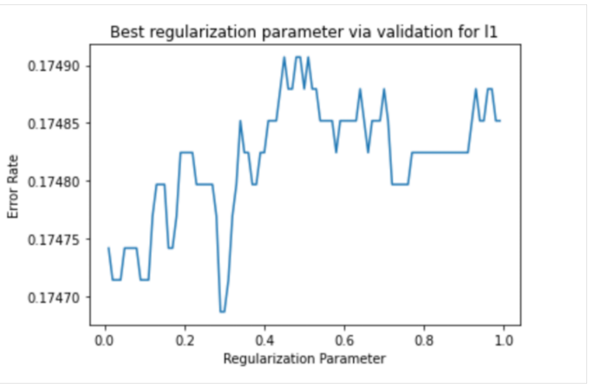
Besides building models that predict the probability of testing positive of a patient with given conditions, we also attempt to answer the question "whether a patient with given conditions will test positive." To construct a solution to this classification problem, we adopt the Logistic Regression model with l2 and l1 regularizer. The equations we are trying to minimize are as follows:

$$l2 : minimize \sum_{i=1}^{n} \log(1 + \exp{(-w^T x)}) + \lambda \sum_{i=1}^{n} {w_i}^2$$

$$l1 : minimize \sum_{i=1}^{n} \log(1 + \exp{(-w^T x)}) + \lambda \sum_{i=1}^{n} |w_i|$$

Previously, the dependent variable of interest was the percentage of people tested and positive (smoothed_pct_tested_and_positive). However, to use logistic regression, we need to change the real-valued variable into categorical variables. Our project uses the following rule to realize the changes: if the variable smoothed_pct_tested_and_positive is originally 0, then we assign 0, meaning the test result is negative; if the variable smoothed_pct_tested_and_positive is greater than 0, then we assign 1, meaning the test result is positive. Then, we perform the logistic regression with l2 and l1 regularizer accordingly.





Firstly, we randomly separated the original training set into two subsets: a new training set that contains 80% of the data in the original training set and a validation set that contains 20% of the data in the original training set. Using the validation set and finding the minimal error rate, we selected the best regularization parameter for the logistic regression with l2 and l1 regularizer respectively. As shown in the figures and the table, the best regularization parameter for l2 is 0.9 and the best regularization parameter for l1 is 0.29. We approximate the average error rate and calculate the average score of the logistic regression model with l2 and l1 regularizer, using the 5-fold cross-validation. As summarized in the table, the average error rate for l1 is smaller than that of l2 and the score for l1 is higher than that of l2. Therefore, we can conclude that the logistic regression with l1 performs slightly better than the logistic regression with l2.

Moreover, we further calculated the Type I error and Type II error in our model. Type I error stands for the false positive rate in the test set and Type II error stands for the false negative rate in the test set. As shown in the table, the model with l2 has higher Type I error and lower Type II error, whereas the model with l1 has lower Type I error and higher Type II error.

4.2.1 Conclusion From Logistic Regression

From the logistic regression, we conclude that the model with l1 regularizer has a lower average error and higher score, and it also has a lower false positive rate and higher false negative rate. The accuracy of our model is approximately 83% on average. Regarding choosing the appropriate model to use in the real world, we suggest taking into consideration both the tradeoff between Type I error and Type II error and the average accuracy. For institutions that value the accuracy of the model the most, we suggest using the logistic regression with l1 regularization. For institutions that prefer a more conservative approach in assessing whether a patient will test positive for Covid-19, we suggest using logistic regression with l2 regularization, since this model predicts more positive test results for people without Covid-19. Therefore, the logistic regression with l2 regularization takes more precautions.

Penalty	L1	L2
Best Regularization	0.9	0.29
Avg. Error Rate	0.17471	0.17469
Score	0.82529	0.82531
False Positive Rate	0.16263	0.16257
False Negative Rate	0.01315	0.01318

5 Discussion

5.1 Fairness

As for the fairness for models we built, among all features used, most features are symptoms observed, and we only used gender and age as part of features. Unlike when predicting whether a person should be approved for parole or mortgage, there are no intentional discriminatory acts associated with the result generated in our model. What is more, if we insist on using specific notions like unawareness to evaluate our model on whether it is fair for different genders or age buckets, it may result in poor accuracy since these are the features that correlate with healthy conditions and would have an impact on COVID results. Therefore, there is very limited fairness concern in our models.

5.2 Weapon of Math Destruction

Besides fairness, it is also important for us to consider whether our project might produce a Weapon of Math Destruction. Firstly, the outcome of our models is measurable. Secondly, our models do not create self-fulfilling or defeating feedback loops. Lastly, our predictions do not have negative consequences when used properly. However, concerns arise when models are used inappropriately. If the institutions take extreme or unethical acts against people predicted to test positive or have a high probability of testing positive, the misused models will lead to harmful results. Nonetheless, in general, the misuse of our model is rare. Therefore, when used with caution, it is unlikely for our model to become a Weapon of Math Destruction.

5.3 Would you be willing to use them in production to change how your company or enterprise makes decisions?

We believe the results of our models could facilitate the testing process in terms of prioritizing the individuals in need and allocating the testing kits more efficiently. By applying our models, medical institutions could get a clear understanding of whether a patient may get COVID or not simply by looking at their symptoms. The symptom data collected from the patients could also provide a benchmark for companies or enterprises to approximate the demands in production.

6 Conclusions

6.1 Conclusion

In this project, we used the computational method we learned in class to build models which enable us to predict the probability of testing positive for COVID-19 given the person's current symptoms. We first build a quadratic model without regression to serve as a base when comparing with other models. Then we build another two regression models with l1 and l2 loss and we achieved an MSE of 1.44. We then build logistic models to help determine whether a patient with given conditions will test positive for COVID-19.

From our models, we conclude that loss of the senses of smell (anosmia) and taste (ageusia) is the most useful method to predict whether a person has COVID or not, and symptoms like runny nose may not be a good indicator for predicting COVID positive possibilities. In addition, age is also an important feature when deciding whether a person has COVID. From our results, people older than 55 are more likely to be tested positive for covid when the symptoms discussed above.

After trying different models, we are confident with our results, since we achieved a relatively low MSE for our quadratic regression models and approximately 82.5% accuracy for our logistic models.

6.2 Future Steps

There are still processes that could be done in the future. First, we could improve our models by taking more features into consideration. We are planning to explore more factors which might contribute to the consequence of testing positive for covid-19 such as the number of people tested positive in the neighbourhood and their symptoms. In addition, we could add categorical variables to our response variable such as the level of symptoms, so that we could use Decision Trees and Neural Networks to help better analyze our data.

Also, we could compare our result with the result of other researches and papers to discuss different methods we used and the advantages and disadvantages for different models. We could also apply those methods into our current result to build a more effective model and better predict the probability of testing positive for COVID-19.