“COVID-19 Mask Analysis”

By

Monroe May
Thomas Lindbloom
Michael Gruver

April 23rd, 2024

“On my honor as an Aggie, I have neither given nor received unauthorized aid on this academic work.” This is entirely my own work or—if authorized by my instructor—the work of the team I was assigned to-- except as documented in a bibliography or acknowledged below.

Monroe May, Thomas Lindbloom, Michael Gruver

Submitted to fulfill the requirement for Project 2
ISEN 413, Spring 2024
Department of Industrial & Systems Engineering
Texas A&M University

Executive Summary

In early 2020 COVID-19 quickly swept the world and changed people's lives forever. Among these changes to everyday life was the introduction of three healthy habits for individuals to practice to minimize the spread of the airborne disease. The most polarizing of the three habits was wearing a mask in public. Many people had their own opinions on this topic and it caused ripples of discussion on whether to comply.

Texas A&M University surveyed 177 individuals in the community, asking a range of questions that culminated in the question “How often do you wear a mask in public settings?”. The survey and the individual responses have been analyzed using three different methods to predict how often people wear masks in public given their respective answers. The three methods utilized are the Tree-Based, Random Forest, and Ordinal Regression.

These models were performed and compared against each other. It was found that the Random Forest provided the best fit to the data. This model used 46 of the survey questions to predict the answer of the individual's culminating question. It achieved 75% accuracy when comparing the predicted data to the actual testing data to verify its results.

Overall, it can be concluded that in order to predict an individual's likelihood of wearing a mask in public, given their answers to the survey questions, a random forest should be utilized for this classification problem.

Contributions:
Thomas Lindbloom - 100/100
Michael Gruver - 100/100
Monroe May - 100/100

Section 1: Analysis of the Methods

Section 1.1: Analysis of Tree-Based Method

The tree-based method is a versatile tool used to predict categorical data. It excels at predicting data that can be categorized, such as in this project, where the data is used to predict the likelihood of an individual wearing a mask. One of the key strengths of the tree-based method is its interpretability. The trees it produces are easy to read and understand. Factors such as the lengths of the branches and the representation of nodes can further enhance one's understanding of the model. In this model, predictions fall into three categories: rarely, most of the time, and always. The model's performance can be evaluated using metrics such as mean squared error and accuracy, which can be measured against test data.

To perform a Tree-based analysis, the initial step involves data preparation and splitting it into training and test sets. The dataset was meticulously prepared by excluding entries with missing values, ensuring that the model was trained solely on complete and meaningful survey responses, thus avoiding unnecessary predictions. Subsequently, the first Tree model was constructed, as depicted in *Figure 1*. This model identified question Q66-2 as the root node, which inquires about an individual's agreement to plan on wearing a mask in public. This question serves as the primary factor for categorizing the predicted response variable. Notably, the root node posses the longest branches, indicating the highest uncertainty level. This is expected, as it represents the initial question used for prediction. The model then identified a set of survey questions critical to making predictions. For categorizing predictions as "rarely," the model pinpointed Q99-8 and Q11-4. Alternatively, for "Most of the Time" and "Always," the model highlighted Q99-8, Q45-2, Q11-16, Q28-1, Q11-10, Q11-11, and Q11-15 as pivotal nodes. Based on the responses to these questions, the model would predict a response. These questions played a crucial role in prediction accuracy. The initial model yielded an error of 59 and an accuracy of 0%. An accuracy of 0% suggests that this model is not a good fit. However, it's also plausible that the model predicted the data very closely but not precisely, implying that while the predictions may not have been very accurate, they could have been very close.

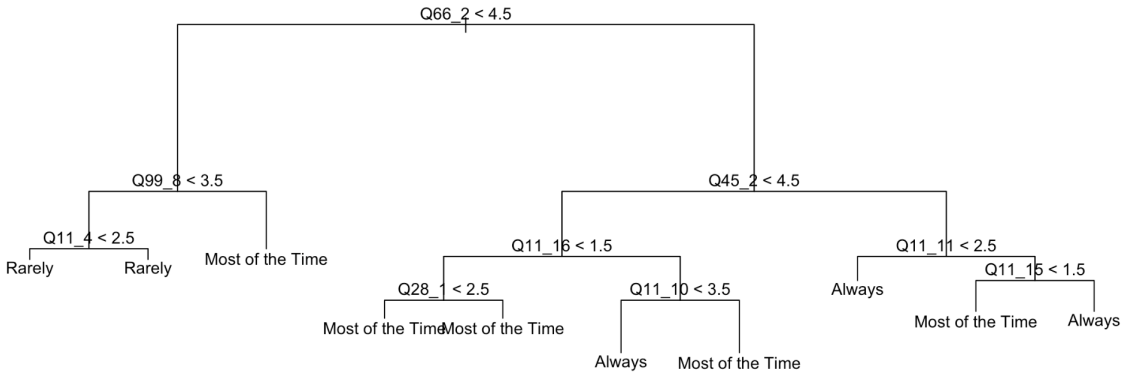

Figure 1: Unpruned Tree

The next step in the Tree-based Method was to prune the tree. This was accomplished by evaluating each pruned level from two to 200 and comparing the mean squared error for each. The pruned tree with the lowest mean squared error was considered the best fit. It was determined that pruning the tree to the best five questions resulted in the best fit. This pruned tree is illustrated in *Figure 2*. Similar to the initial model, this pruned tree identified Q66-2 as the root node. Subsequently, Q99-8 was identified to indicate "Rarely". For "Most of the Time" and "Always," the pivotal nodes were Q99-8, Q45-2, Q11-16, and Q11-10. This figure is more concise and easier to interpret. Additionally, it achieved a lower error of 44. However, similar to the initial model, this pruned model also had a 0% accuracy, indicating once again a poor fit and prediction rate.

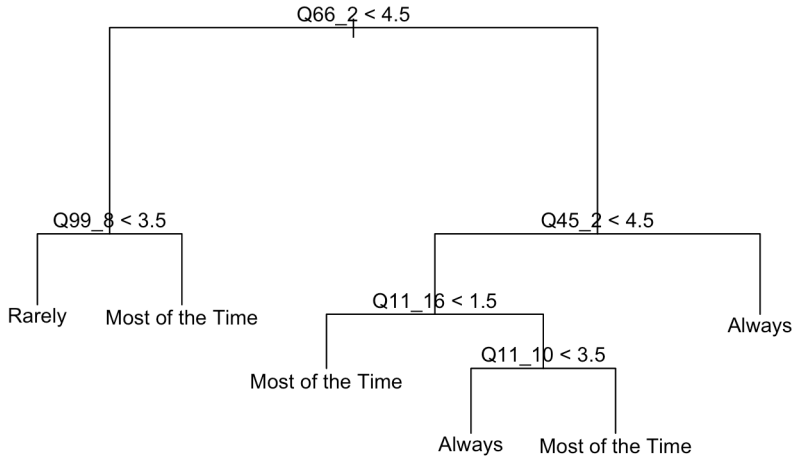

Figure 2: Pruned Tree

Overall, the Tree-based method appears to be a poor fit for this model and most likely will not be used as the final model for this project.

Section 1.2: Analysis of Random Forest Method

Random forest is able to reduce bias in the model by comparing many decision trees and improve a model to ensure overfitting is minimized. This will help with improving accuracy in test data sets while maintaining a lower bias than a normal tree would see. Random forest also is able to handle both categorical and numerical data very easily with little need for data pre-processing and can streamline the model building process, this is why we have chosen to evaluate a model with this method.

The data is split into test and training data sets. In this model for random forest, the training data set used 80 percent of the dataset to formulate a model. The initial random forest model is fitted utilizing the number of predictors that is optimal to predict the response variable is $\sqrt{p}$ in this model. The initial random forest model utilizes $\sqrt{151}$ predictors in the various combinations of the number of different random trees that are generated. For the first use of cross validation to find the optimal values of trees(ntree) to generate. After finding that value we then will use it to find the optimal number of predictors(mtry).

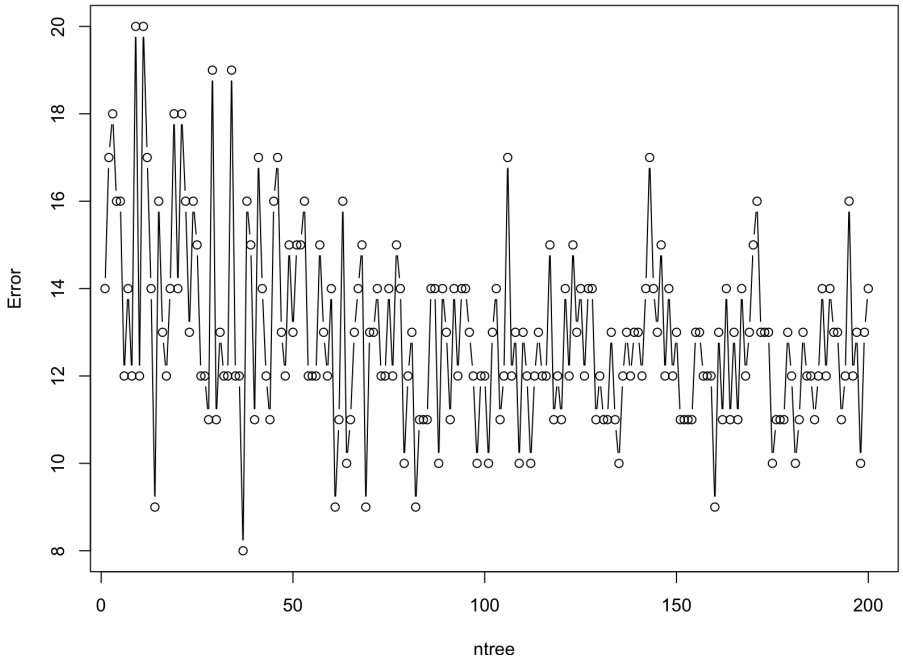

Figure 3: Error vs. ntree

Shown above in ***Figure 3*** is the program iterating through different values of ntree up to 200, the most optimal value for number of trees to generate and compare was found to be 35 which generated an error of 8. Next we will replicate this process for finding the optimal value to use for mtry.

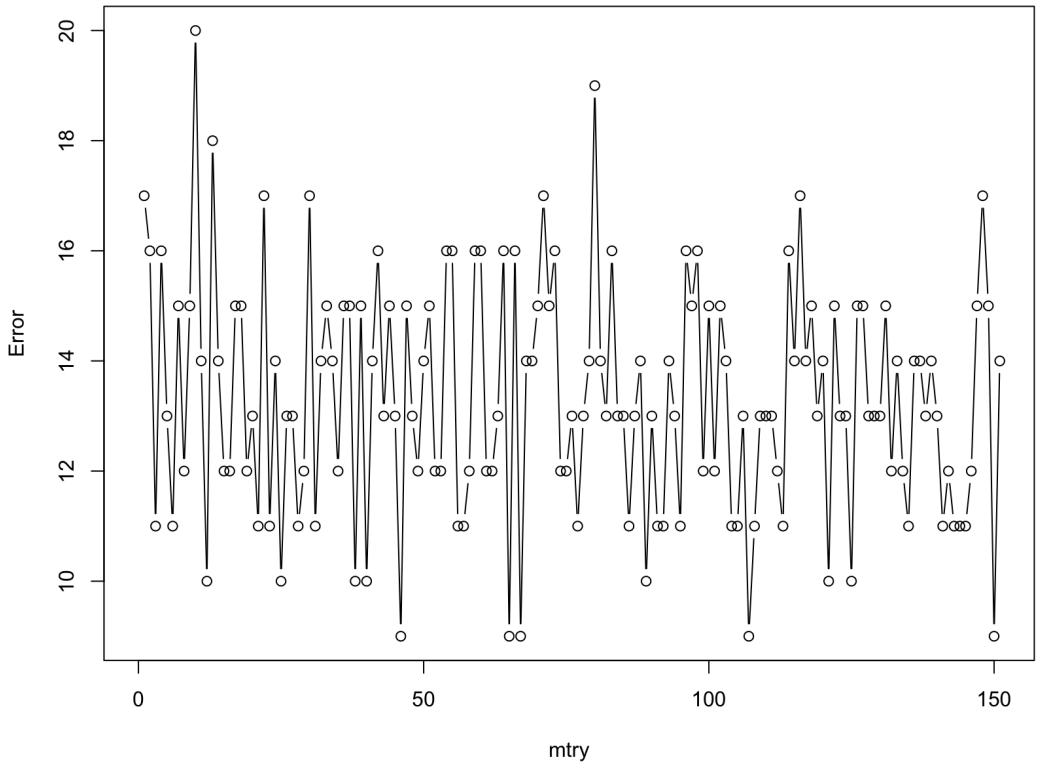

Figure 4: Error vs. mtry

Looking at ***Figure 4*** above, we see that the optimal value of mtry to reduce the total error of the model is found to be the value of 46. When setting the mtry to 46 and using the optimal value for ntree as 35, the final model then can be fitted and evaluated for the overall error after optimizing these values for the least amount of error. With the fitted final model, the absolute error is found to be 9. This model is considerably accurate in predicting the values. Looking at the confusion matrix in the ***Table 1*** below we can see how the data has classified the test data set and see where it has misclassified certain data points in the wrong category. Looking at the data we see that the model is able to predict new classifications based off of the predictors.

	Always	Most of the Time	Rarely
Always	9	2	1
Most of the Time	5	9	0
Rarely	1	2	6

Table 1: Confusion Matrix

Section 1.3: Analysis of Ordinal Regression Method

The Ordinal classification model for classification is a good option due to the fact that there are more than two categories and the categories can be ranked or put in an order. When trying to decide how likely someone is to wear a mask the categories of likelihood are ranked from rarely being at the bottom and always ranking at the top. Ordinal regression allows for a more nuanced and ordered understanding of classification (hence the name ordinal). The resulting coefficients can be interpreted similarly to logistic regression, making it easier to understand how the predictors impact the ordinal outcome.

A weakness of using ordinal regression is that it assumes that the probability ratios for changing between ordered categories is the same given that predictor. This simplifies the model and can make interpretation easier but can lead to bias estimates.

When fitting the ordinal regression model the response variable's different categories had to be assigned an order so that the model could distinguish the ordered nature of the response variable it was trying to predict. Initially an ordinal regression model was fit using all of the predictors resulting in 25 misclassified predictions with an absolute error of 44 on the when testing against the 30 data point set aside from the training data to use as test data. This was not good performance and was likely due to an overfit high variance model due to the high number of predictors. A best subset selection attempt to reduce the number of predictors was used and failed due to it being too computationally heavy and there being high multicollinearity between the predictors. To deal with this issue the dimensionality reduction technique principal component analysis was used to reduce the number of predictors and only retain the ones that account for the maximum variance.

The principal component analysis reduced the predictures used from 151 to 46 and the now model with just those predictors performed much better than the original. When testing against the released test data it resulted with 15 misclassifications out of the 59 data points provided and an absolute error of 21 taking in account how far off a misclassification was from the true classification. Below is a figure showing the 46 predictors captured by the principal component analysis and their respective coefficient for the model.

```
Coefficients:
          Estimate Std. Error z value        Pr(>|z|)
Q69female -1.71653    0.70507  -2.435         0.01491 *
PC1        0.56740    0.08769   6.471 0.0000000000976 ***
PC2        0.31674    0.09743   3.251         0.00115 **
PC3        0.04363    0.13229   0.330         0.74157
PC4       -0.20242    0.12360  -1.638         0.10149
PC5        0.27168    0.12644   2.149         0.03166 *
PC6       -0.23621    0.15681  -1.506         0.13197
PC7       -0.44116    0.17676  -2.496         0.01257 *
PC8       -0.04510    0.15459  -0.292         0.77050
PC9        0.50442    0.19625   2.570         0.01016 *
PC10       0.07589    0.14943   0.508         0.61153
PC11       0.53887    0.18405   2.928         0.00341 **
PC12      -0.01622    0.17094  -0.095         0.92441
PC13       0.17172    0.21109   0.814         0.41592
PC14      -0.57955    0.25453  -2.277         0.02279 *
PC15      -0.05250    0.17903  -0.293         0.76933
PC16      -0.52736    0.23284  -2.265         0.02352 *
PC17      -0.30702    0.21018  -1.461         0.14408
PC18      -0.15414    0.19276  -0.800         0.42393
PC19      -0.11693    0.23363  -0.501         0.61672
PC20      -0.02752    0.24014  -0.115         0.90876
PC21      -0.11217    0.27462  -0.408         0.68293
PC22      -0.57216    0.25872  -2.211         0.02700 *
PC23      -0.67578    0.25390  -2.662         0.00778 **
PC24       0.03784    0.22853   0.166         0.86850
PC25      -0.08666    0.25905  -0.335         0.73799
PC26       0.35481    0.23914   1.484         0.13790
PC27      -0.12137    0.27039  -0.449         0.65352
PC28       0.79679    0.31609   2.521         0.01171 *
PC29      -0.67931    0.32134  -2.114         0.03452 *
PC30       0.09444    0.26314   0.359         0.71967
PC31      -0.01547    0.29615  -0.052         0.95834
PC32       0.27521    0.27068   1.017         0.30929
PC33       0.24754    0.29916   0.827         0.40798
PC34       0.38535    0.28407   1.357         0.17493
PC35       0.03955    0.29358   0.135         0.89283
PC36       0.22619    0.31109   0.727         0.46717
PC37      -0.19065    0.29040  -0.656         0.51150
PC38      -0.55366    0.29186  -1.897         0.05783 .
PC39       0.52575    0.31061   1.693         0.09052 .
PC40       0.77468    0.34147   2.269         0.02329 *
PC41      -0.88275    0.33654  -2.623         0.00871 **
PC42      -0.49002    0.36403  -1.346         0.17827
PC43      -0.99175    0.34267  -2.894         0.00380 **
PC44       1.03161    0.42654   2.419         0.01558 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 1: Final Ordinal Model Coefficient

Section 2: Comparison & Results

After looking at the three models we have developed so far, we have selected the random forest for our final model due to it scoring the lowest absolute error as seen in ***Table 1*** below. We have ruled out using our tree-based model since it was not accurate enough for predicting new values. We can see that the best models that we found both utilized 46 predictors.

Model	Number of predictors	# Misclassified	Absolute Error
Tree-based	5	All	415
Random Forest	46	12	13
Ordinal	46	15	21

Table 1: Comparison of the Models

The initial test error was computed for these models by withholding 20% of the data to be used for testing purposes, while using the other 80% to train our models with. There were a total of 35 points available to test our models with. The random forest model had only misclassified 12 points while Ordinal had 15 points.

Section 3: Evaluation of Best Model

Looking into the evaluation of the random forest model against the newly provided testing data, the absolute error calculated using this method was 20. But after looking further we have determined that this model can now utilize all data points from our training data set, so we can fit an improved model to try and reduce this error some more. The testing data set contained 60 points. In the table below we can see where the model had missed-categorized the most data in the boundary between “Most of the Time” and “Always”. This model had an absolute error of 20, but we will explore other methods in the next section to try and minimize the error in the model.

	Always	Most of the Time	Rarely
Always	15	9	0
Most of the Time	2	13	2
Rarely	2	4	13

Table 2: Confusion Matrix (Testing data)

Section 4: Improving the Best Model

After initially using the random forest models with the given test dataset, our ordinal model scored a higher score, but we soon realized that now we no longer need to withhold some data from the training set that was provided in phase 1 of the project. Now we can fit both models to the whole dataset from the original training dataset file. The random forest model using all of the training data to build a model is able to reach an error of 17, whereas before it was reaching an absolute error score of 20. So we will use this model as our final model for the project.we can see the differences between table 2 and table 3 show that more of those points between the “Most of the Time” and “Always” has decreased as well. In the end our final model had ended with a final absolute error of 17.

	Always	Most of the Time	Rarely
Always	14	3	0
Most of the Time	3	17	1
Rarely	2	6	13

Table 3: Confusion Matrix (Improved Testing data model)