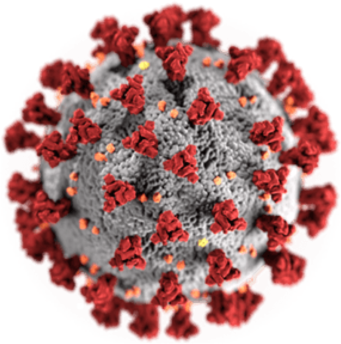


Project 3
Classification
COVID-19 Dataset

CS 5331 - Data Mining

Authors:

Liam Lowsley-Williams

Emily Fashenpour

Harrison Noble

I. Executive Summary

In early 2020, the United States was introduced to the novel COVID-19 virus which rapidly swept across the nation, infecting and killing thousands of US citizens. States and counties quickly began documenting case numbers and the death toll on a day to day basis to better track the effects of this virus. When infection and death numbers are cross referenced with census data about the certain characteristics of populations, interesting conclusions can be made regarding how the virus spreads in different parts of the country. Many health experts are concerned that a fourth wave, or surge in cases, in COVID-19 cases is occurring and that the US and the world will see a dramatic rise in cases, and therefore deaths [1]. Given this concern, we will examine COVID-19 case counts and other census data from counties all over the US to better determine which counties in Texas are the most at risk and would be most affected by this fourth surge in cases. Once the most at risk counties, and least at risk, are identified, the next best steps the counties could take will be recommended. These recommendations are intended to dampen a severe outbreak in the county. For example, if a county is highly at risk, then it may be recommended that the county enforce a mask mandate and close 75% of the non-essential business. This report is written for government healthcare entities in the state of Texas in hopes that they will consider our recommendations. We chose to analyze and classify counties as fatal or not fatal and found the best classification method was Random Forest. We also found that our model should not be the only tool or resource used to make final decisions on actions a county should take, but as a secondary guide and recommendation.

Table of Contents

II. Data Preparation

Data Features Used for Classification

For this project we modified our dataset from Project 2 to included and removed census data features and removed the timeseries Texas data. We also updated the number of confirmed cases in each county. Before, the data was from January 25, 2021. The confirmed case numbers were updated to reflect the number of confirmed cases as of April 24, 2021. From the “COVID-19 cases plus census” we extracted population information (total and per race), median income, income per capita, number of commuters by public transportation, number of housing units over $1 million, education levels, and number of people in poverty. We also used data for all counties in the US from the “COVID-19 cases plus census” dataset but we are limiting our focus to Texas, in terms of recommendation, on this report. **Because Texas has a vast population range at the county level, we decided to convert the features confirmed_cases and deaths to “per 1000 people” and divide most other features by the total_population to get a percent of the total population. We did this to better normalize the data and give a better understanding of the relative density of an attribute in a county. Without this, the data could skew to either the few very large counties with high populations, or the many very small counties with small populations**. Below is the description of the features for the combined dataset we will be using for analysis.

Table 1: Description of Features Used for Classification

Feature	DataType	Description
state	Nominal	Name of state in US
county_name	Nominal	Name of county in US
confirmed_cases	Ratio	Cumulative number of confirmed cases per 1000 people at 04/24/2021
deaths	Ratio	Cumulative number of deaths per 1000 people at 04/24/2021
fatality_rate	Ratio	Deaths divided by confirmed cases
commuters_by_public_transportation	Ratio	Percentage of population that take public transportation
poverty	Ratio	Percentage of population that live below the poverty line
white_pop	Ratio	Percentage of population identifying as White
black_pop	Ratio	Percentage of population identifying as African American
asian_pop	Ratio	Percentage of population identifying as Asian
hispanic_pop	Ratio	Percentage of population identifying as Hispanic
amerindian_pop	Ratio	Percentage of population identifying as American Indian

total_pop	Ratio	Total population of a given county
male_pop	Ratio	Percentage of population identifying as male
female_pop	Ratio	Percentage of population identifying as female
median_income	Ratio	Median income
income_per_capita	Ratio	Income per-capita
median_age	Ratio	Median age
masters_degree	Ratio	Percentage of population with a master's degree
bachelors_degree	Ratio	Percentage of population with a bachelor's degree
high_school_diploma	Ratio	Percentage of population with a highschool diploma
high_school_including_ged	Ratio	Percentage of population with a GED
in_undergrad_college	Ratio	Percentage of population in undergraduate college
in_school	Ratio	Percentage of population in any type of school
worked_at_home	Ratio	Percentage of population that works at home
walked_to_work	Ratio	Percentage of population that walks to work
gini_index	Ratio	Measure of wealth inequality within a county
million_dollar_housing_units	Ratio	Percentage of housing units costing over $1 million

Predictive Features & Class Variables

For our class variable, we decided to classify a county as either fatal or not fatal using the deaths per 1000 feature. If a county had a deaths per 1000 value greater than 1.6, then the county was determined to be fatal. Any value less than 1.6 was determined to be not fatal.

```
Fatal = {Deaths per 1000 > 1.6 = TRUE, else = FALSE}.
```

The class was chosen to be split at the 1.6 value because it gave a reasonably balanced class result (see Figure 1). Furthermore, after testing different cutoff values below and above 1.6, we found worse accuracy and kappa values, so we decided to stick with it.

We chose to look at the fatality of a county based on deaths per 1000 because we felt it is an important descriptor with relation to COVID-19. Determining whether a county was more or less dangerous based on census demographic information could help us understand what areas could be associated with higher risk when dealing with a future pandemic in the unfortunate event that one does occur again.

We decided to split our data based on several states to obtain a 80/20 split on the number of true and false values that we saw from our total dataset balance. As you can see below in Figure 1, our testing dataset contains a similar balance to our training dataset with the difference being it contained roughly 20% of our total dataset size. Below you can find information on our dataset balance and what features were most important when performing a Chi Squared test. We can see how all of our data features appear in the Chi Squared test, which is another reason we decided to pick the features outlined in Table 1.

Figure 1: Class Balance

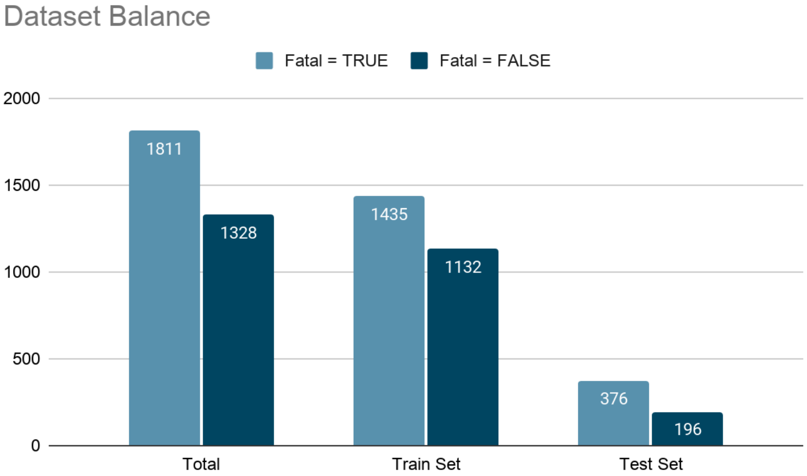
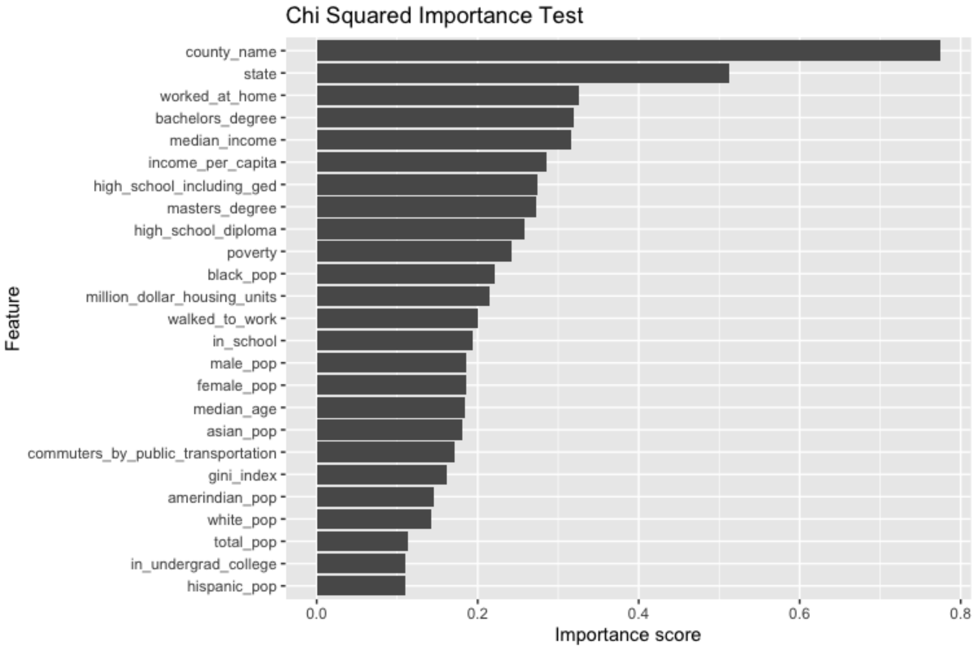


Table 2: Top 10 Chi Squared Feature Importance

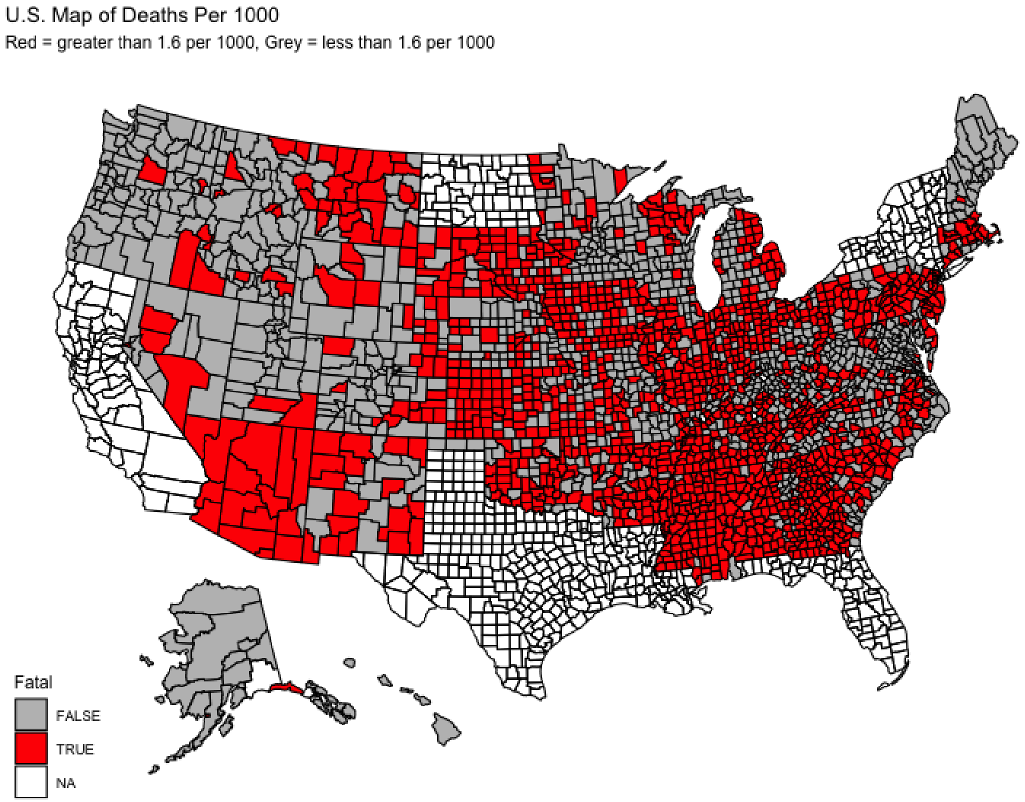
Feature Name	Value
county_name	0.774
state	0.513
worked_at_home	0.326
bachelors_degree	0.319
median_income	0.316
income_per_capita	0.285
high_school_including_ged	0.274
masters_degree	0.272
high_school_diploma	0.259
poverty	0.243

Figure 2: Chi Squared Feature Importance Graph



In Figure 3 below, you can see the states that were used in our training dataset. The states that are excluded from the training dataset for the testing dataset were chosen to be Texas, California, North Dakota, Florida, Vermont, Louisiana, and New York. We chose to leave these states because we felt they were a good mix of large and small states with varying outbreak sizes that would be best for testing out models. We also left Texas in the testing dataset because we want to focus our analysis and later recommendations on it, so we wanted to see how well each of the models predicted the fatality of each county in Texas.

Figure 3: Map of Training Dataset


U.S. Map of Deaths Per 1000
Red = greater than 1.6 per 1000, Grey = less than 1.6 per 1000
Fatal
FALSE
TRUE
NA

III. Modeling

For our classification models, we decided to create four models total. Each model uses different classification techniques but all use the same class variable. Each model also uses 10 fold cross validation. In our case, the class variable will be deaths per 1000 people. We then compare how each model performs using a variety of different metrics.

Classification Model 1 - RPART

For our first model, we decided to use the RPART method for classification. Below outlines our training and testing for our RPART model and displays our results.

Training

During training we saw a best complexity parameter of 0.005300353 when using a tune length of 10. With this complexity parameter we obtained a best accuracy of 66.70% and an associated kappa value of 31.19%. Below in Figure 4.1 we can see a graph of the different complexity parameters used and their associated accuracies. We can also see in Figure 4.2 the variable importance that each feature used during training.

Figure 4.1: Accuracy vs. Changing Hyper-parameter

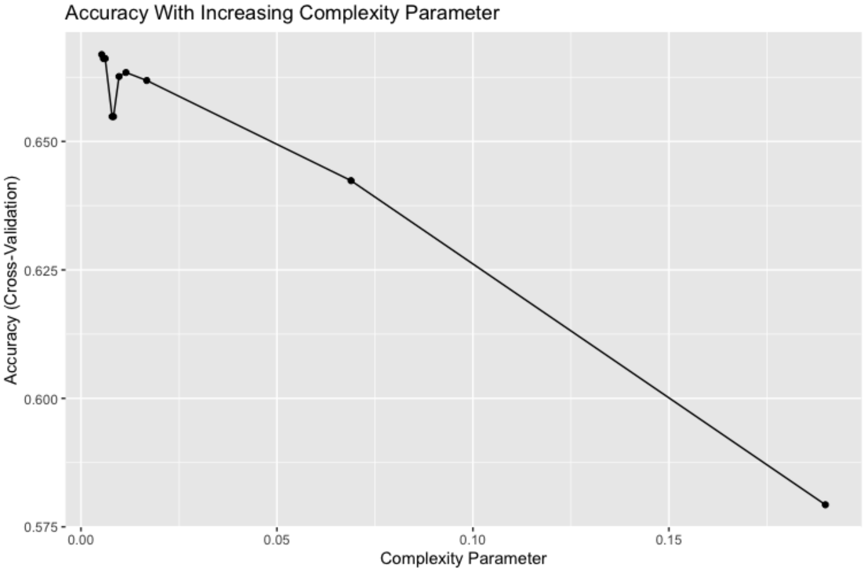
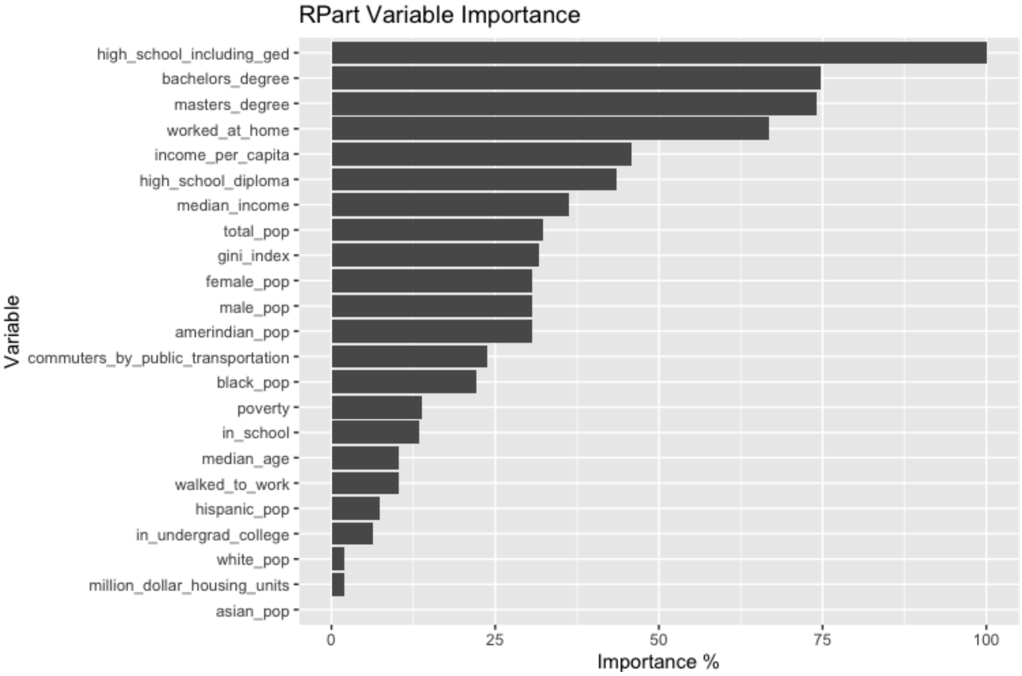


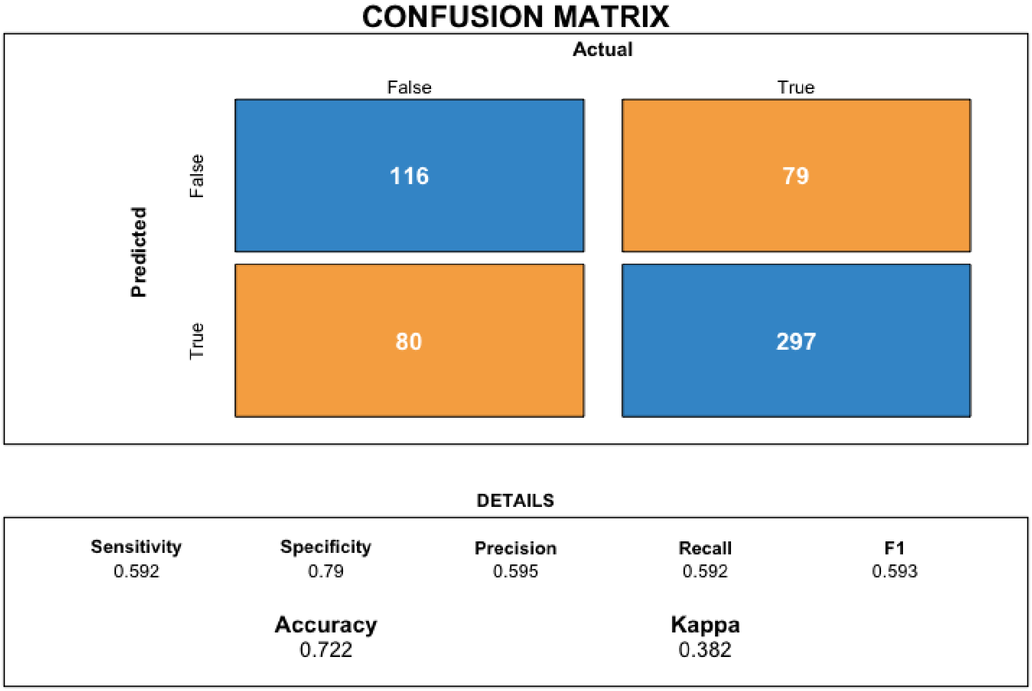
Figure 4.2: Variable Importance for RPart Classification Method



Testing

During testing we observed an accuracy of 72.2% and an associated kappa value of 38.2%. We felt pretty good with these results however they were not the most ideal. That being said, we felt somewhat confident in using this model to predict a county as being fatal or not. Below you can see a confusion matrix of our results in Figure 4.3. What is not shown here however is our no information rate and P-Value. Our no information rate was 65.73%. It is good that our accuracy was above that percentage. In addition, our P-Value was 0.0005464 which was decently low and further boosted some confidence in our results. Additionally, in Figures 4.4 and 4.5 we can see a comparison of performance from the actual classifications and our predicted classifications. For the most part the majority of these areas were covered correctly, however, clearly the model could be better with classification. In order for the model to be useful to use, we would rather it predict a county as fatal and be wrong than have it predict a county as not fatal when it actually is fatal. This means we would rather be over-cautious. Our precision, which is a ratio of the number of counties we predicted to be not fatal when they were actually not fatal to the total number we predicted fatal, was not the most ideal at 59.2%. This lower than ideal precision would mean more counties would be classified as not fatal when they are fatal. However, after examining all the other classification models, RPart was one of the better performers and could be a potentially useful model for our classification needs.

Figure 4.3: RPart Confusion Matrix



CONFUSION MATRIX

		Actual	
		False	True
Predicted	False	116	79
	True	80	297

DETAILS

Sensitivity	Specificity	Precision	Recall	F1
0.592	0.79	0.595	0.592	0.593

Accuracy	Kappa
0.722	0.382

Figure 4.4: Map of Predicted Classifications

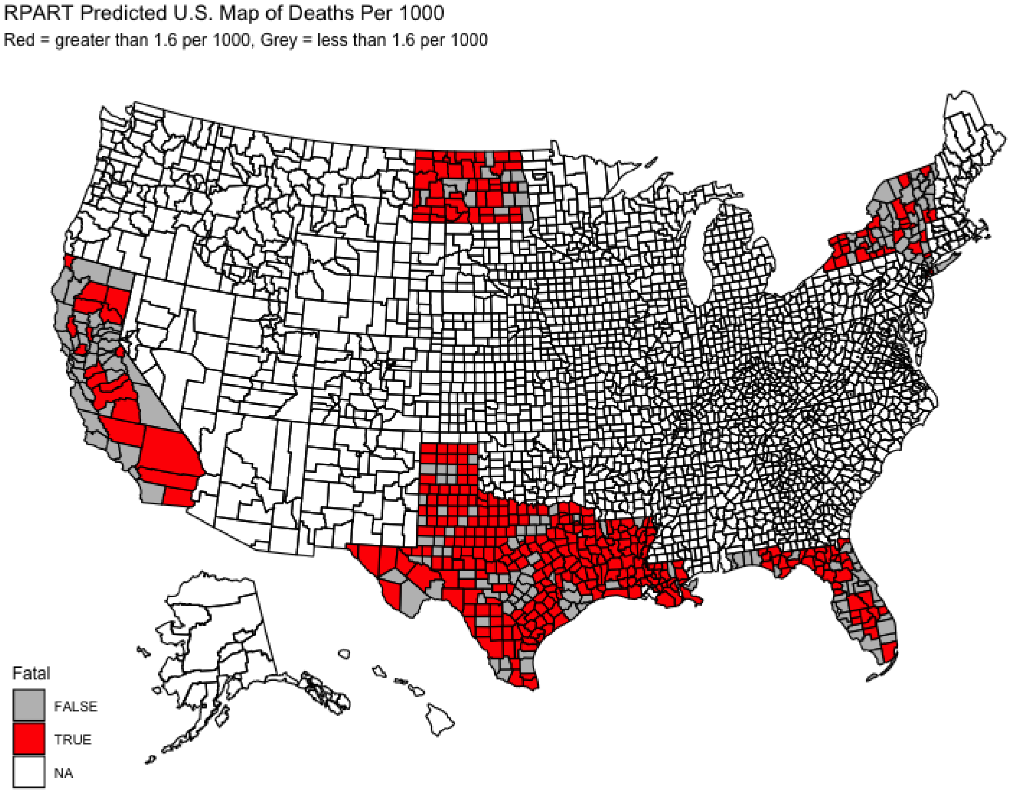
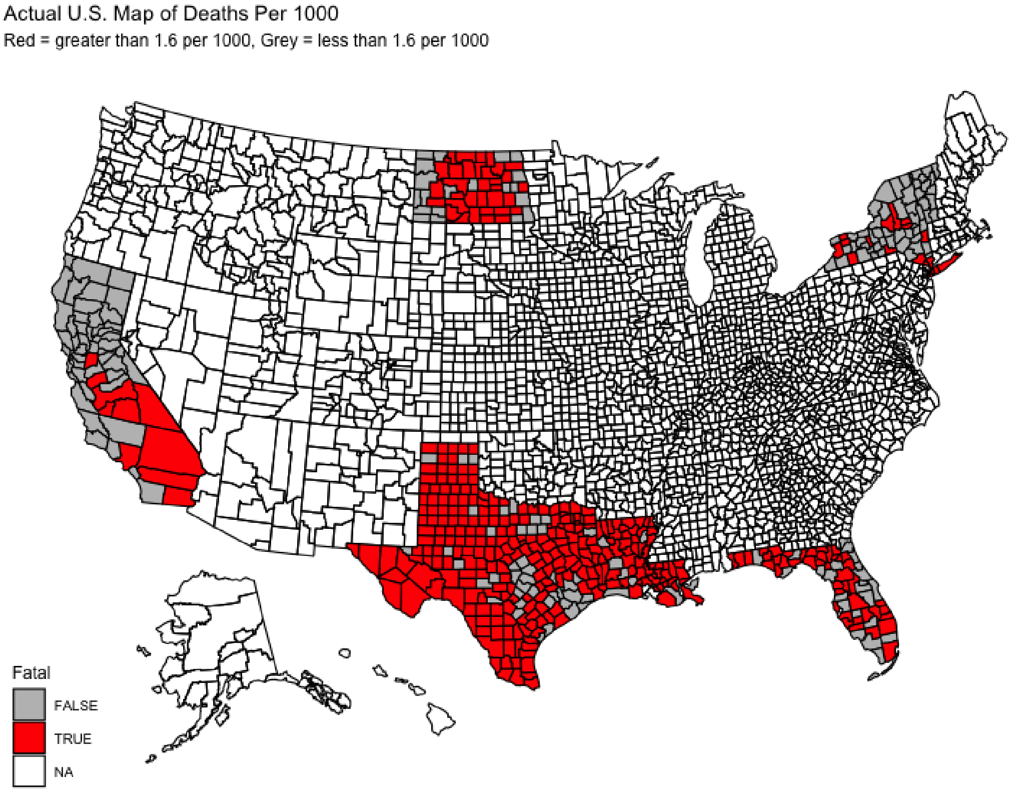

RPART Predicted U.S. Map of Deaths Per 1000
Red = greater than 1.6 per 1000, Grey = less than 1.6 per 1000
Fatal
FALSE
TRUE
NA

Figure 4.5: Map of Actual Classifications


Actual U.S. Map of Deaths Per 1000
Red = greater than 1.6 per 1000, Grey = less than 1.6 per 1000
Fatal
FALSE
TRUE
NA

Classification Model 2 - Random Forest

For our second model, we decided to use the Random Forest (RF) method for classification. Below outlines our training and testing for our Random Forest model and displays our results.

Training

During training we saw a best MTRY parameter of 4 when using a tune length of 10. With this MTRY parameter we obtained a best accuracy of 72.30% and an associated kappa value of 42.96%. Below in Figure 5.1 we can see a graph of the different complexity parameters used and their associated accuracies. We can also see in Figure 5.2 the variable importance that each feature used during training.

Figure 5.1: Accuracy vs. Changing Hyper-Parameter

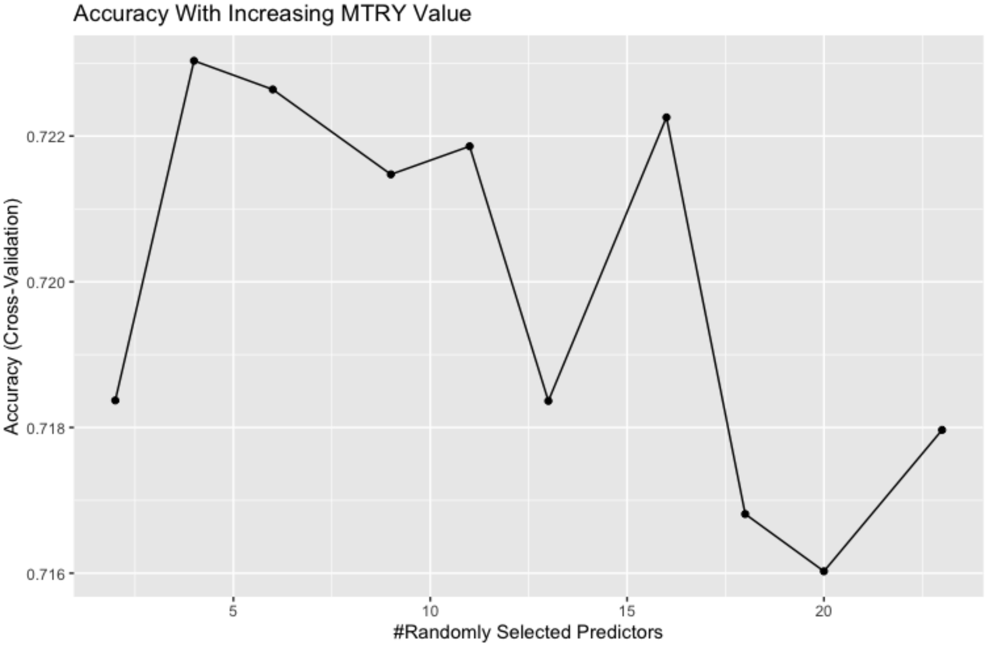
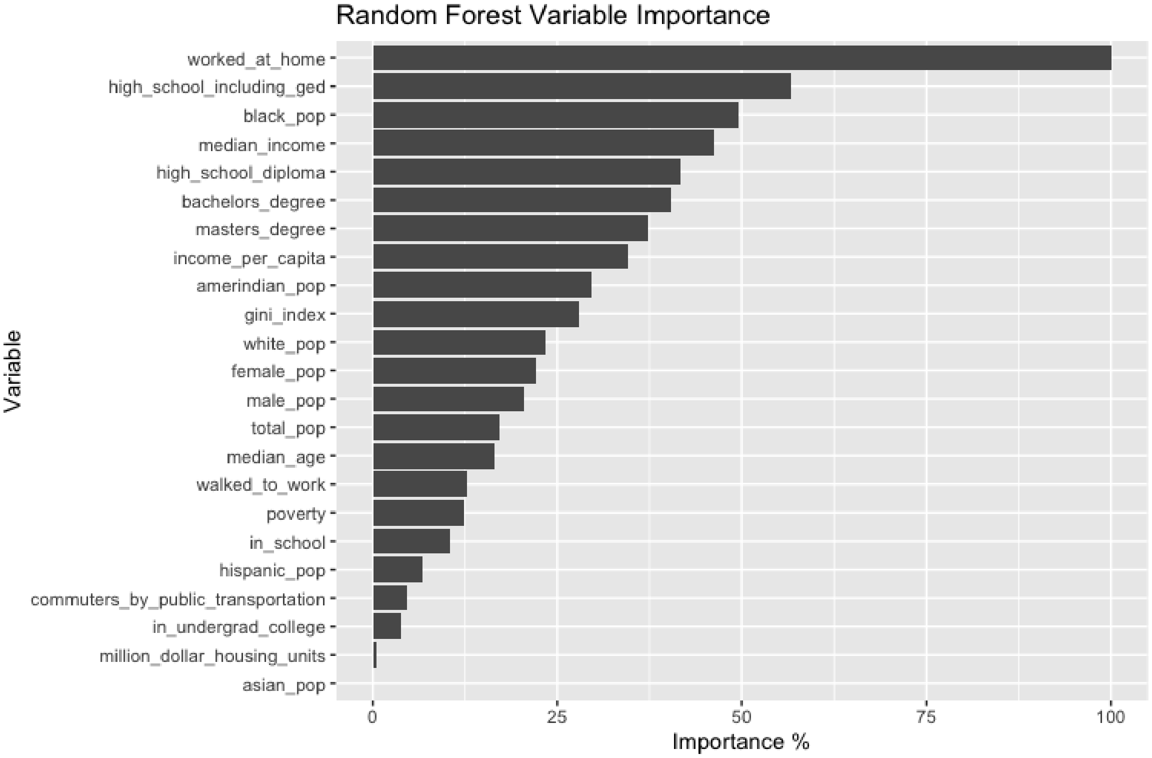


Figure 5.2: Variable Importance for Random Forest Classifier



Testing

During testing we observed an accuracy of 74.1% and an associated kappa value of 43.4%. These results were better than our RPART classifier and we felt good with them even though they were not the most ideal. Therefore, we still felt confident in using this model to predict a county as being fatal or not. Below you can see a confusion matrix of our results in Figure 5.3. What is not shown here however is our no information rate and P-Value. Our no information rate was 65.73%. It is good that our accuracy was above that percentage. In addition, our P-Value was 0.000009483 which was really low and further boosted some confidence in our results. Additionally, in Figures 5.4 and 5.5 we can see a comparison of performance from the actual classifications and our predicted classifications. For the most part the majority of these areas were covered correctly and were better than RPART. This model was by far our best performing model of all the ones we constructed. The precision value is the highest at 61.5%, along with the accuracy and Kappa values. The false positive rate was 1 higher than the lowest we saw through all of our models, however, the benefits we saw from the higher accuracies greatly outweigh this difference. Therefore, we deemed this classifier as the best one we could use in practice of all our classifiers.

Figure 5.3: RF Confusion Matrix

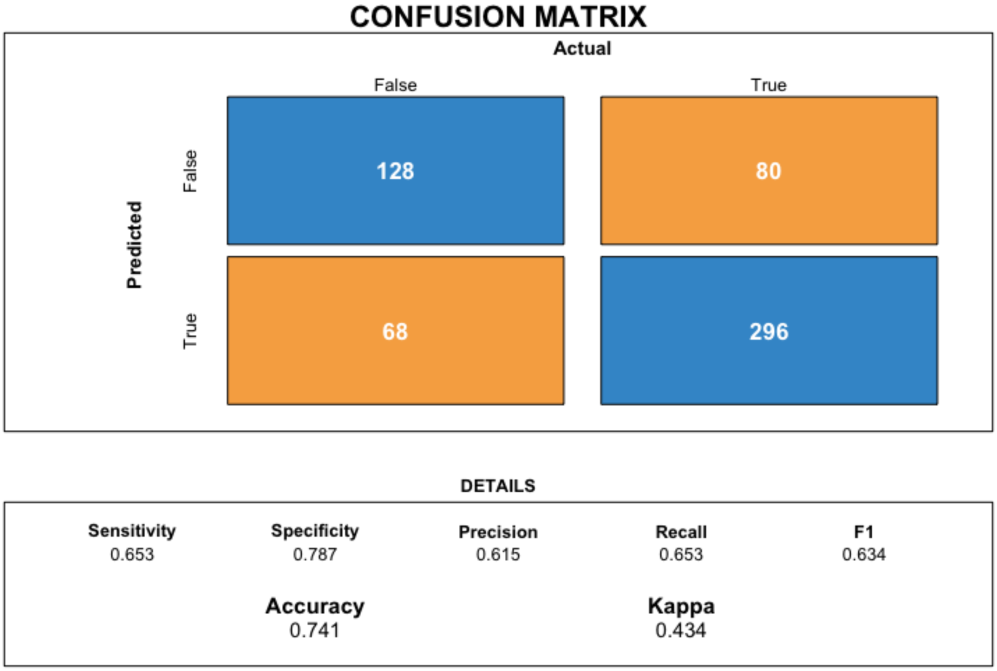


Figure 5.4: Map of Predicted Classifications

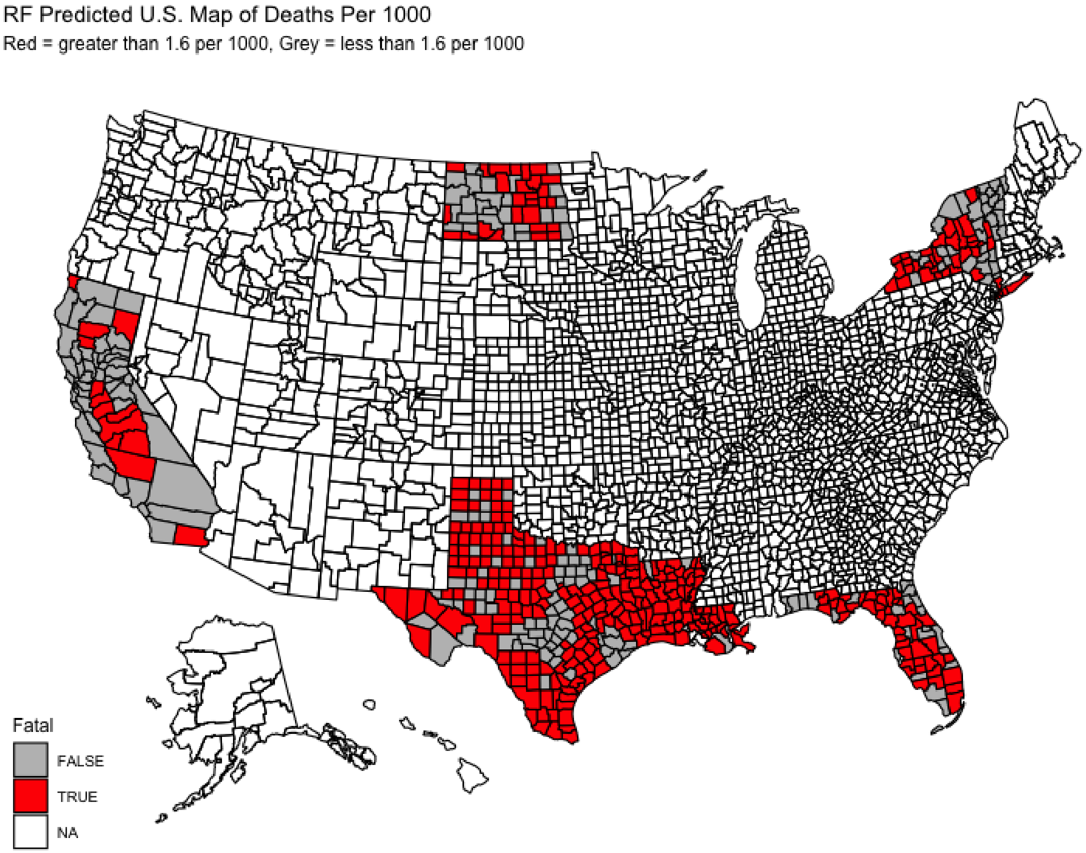
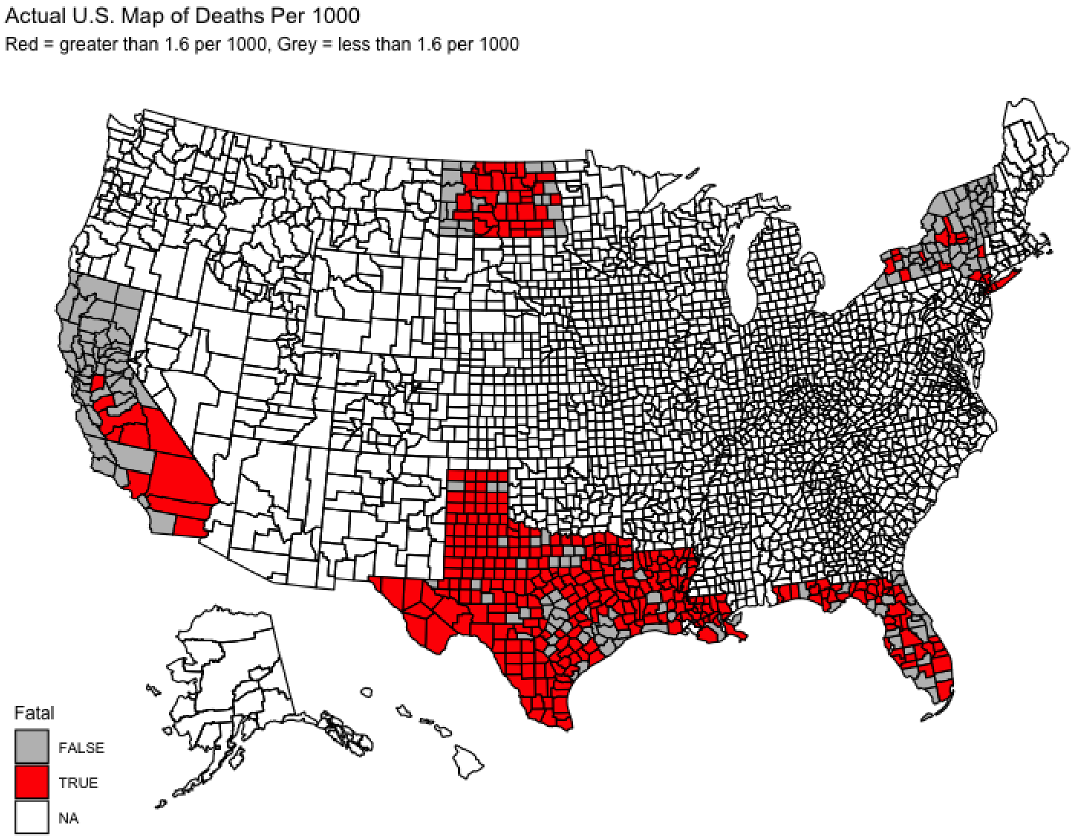

RF Predicted U.S. Map of Deaths Per 1000
Red = greater than 1.6 per 1000, Grey = less than 1.6 per 1000
Fatal
FALSE
TRUE
NA

Figure 5.5: Map of Actual Classifications


Actual U.S. Map of Deaths Per 1000
Red = greater than 1.6 per 1000, Grey = less than 1.6 per 1000
Fatal
FALSE
TRUE
NA

Classification Model 3 - Naive Bayes

For our third model, we decided to use the Naive Bayes (NB) method for classification. Below outlines our training and testing for our Naive Bayes model and displays our results.

Training

During training we saw that using a kernel resulted in slightly higher accuracies and a higher kappa value. Using the kernel, we obtained a best accuracy of 67.74% and an associated Kappa value of 34.66%. When not using a kernel we saw an accuracy of 67.24% and a Kappa of 30.91%. In addition, throughout the training we used an “adjust” value of 1 and an “FL” value of 0. Below in Figure 6.1 we can see a graph of the difference in accuracy when not using a kernel (Gaussian) vs using a kernel (nonparametric). We can also see in Figure 6.2 the variable importance that each feature used during training.

Figure 6.1: Accuracy vs. Kernel Method

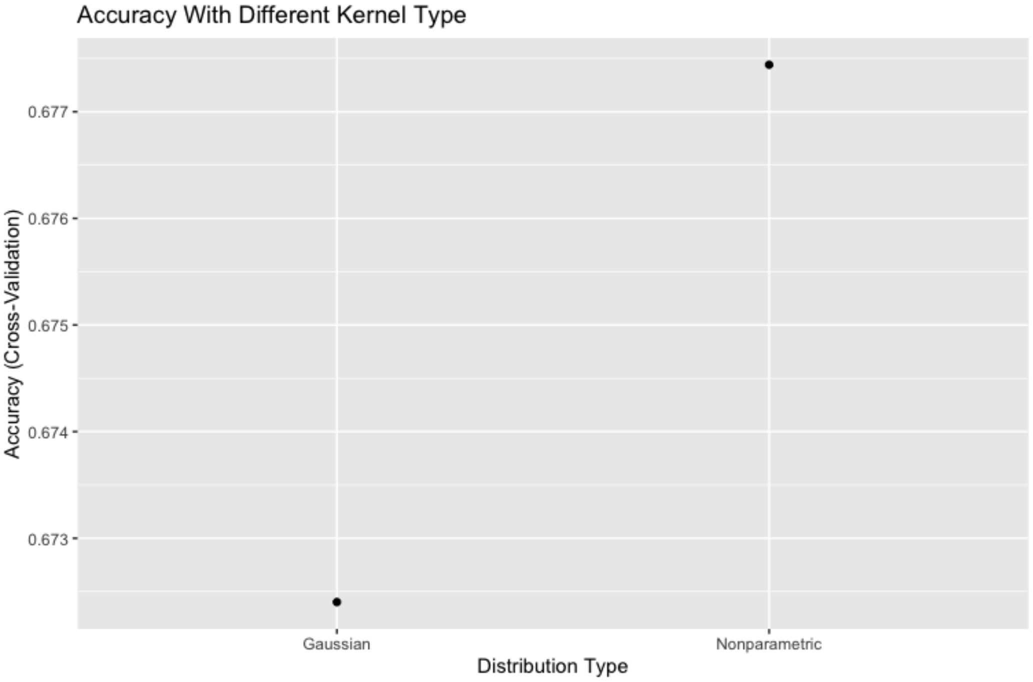
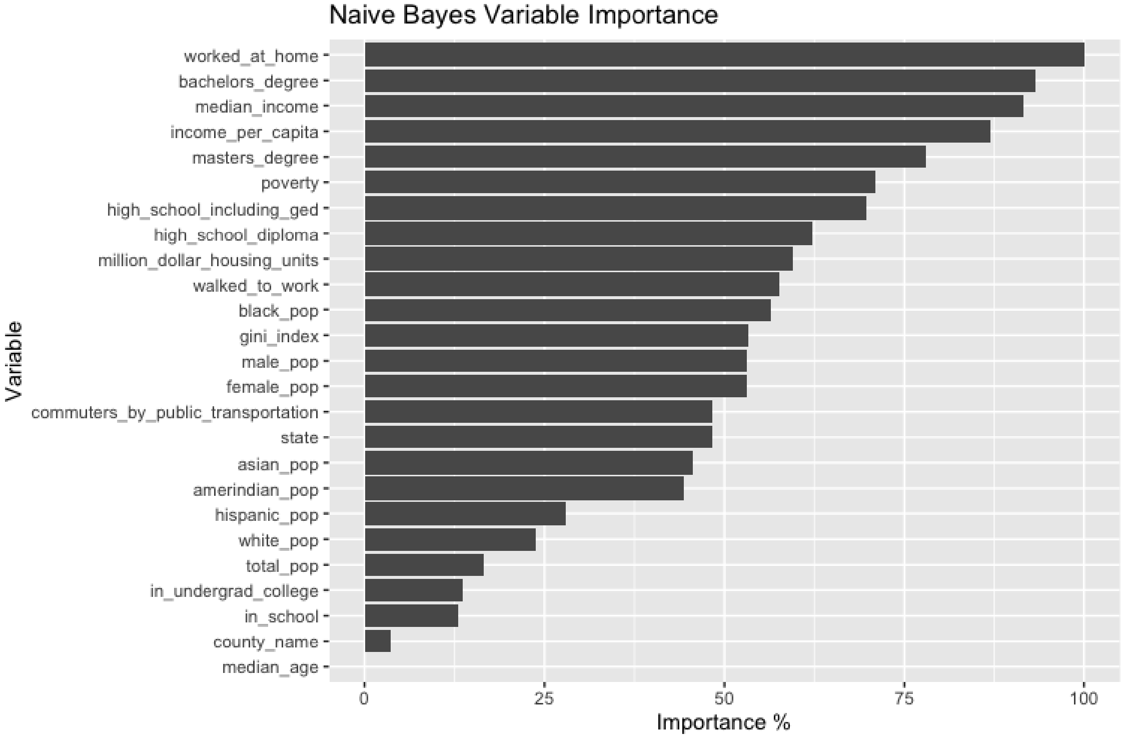


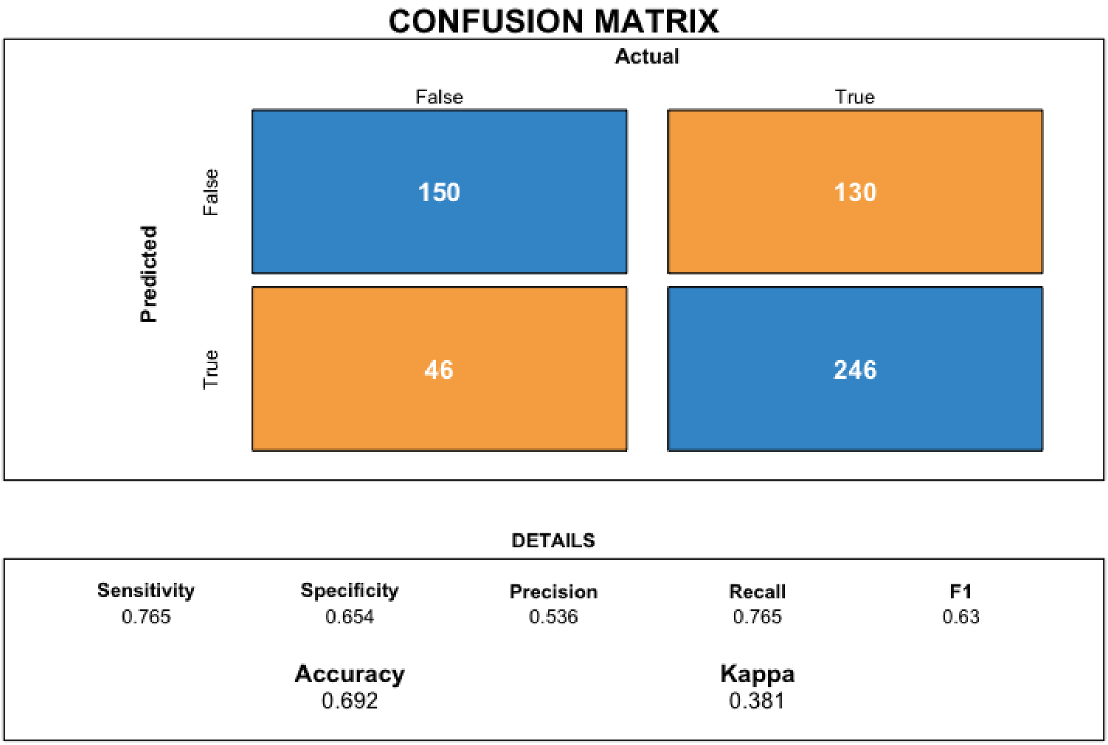
Figure 6.2: Variable Importance for Naive Bayes Classifier



Testing

During testing we observed an accuracy of 69.2% and an associated kappa value of 38.1%. These results were worse than both of our previous classifiers by quite a bit. Thus, we do not feel confident in using this model to predict a county as being fatal or not. Below you can see a confusion matrix of our results in Figure 6.3. What is not shown here however is our no information rate and P-Value. Our no information rate was 65.73% and it is good that our accuracy was at least above that percentage. In addition, our P-Value was 0.04202 which was fairly high and further retracted some confidence in our results. Additionally, in Figures 6.4 and 6.5 we can see a comparison of performance from the actual classifications and our predicted classifications. The predictions were not great and our false positive rate was way too high. Our precision, which is a ratio of the number of counties we predicted to be not fatal that were actually not fatal to the total number we predicted fatal, was pretty terrible being 53.6%. This was worrying to us as this model would deem an area as not fatal when it was actually fatal with a fairly high probability. We did not like this as we would prefer to err on the side of caution when determining whether a county was fatal or not. Thus this model was not very useful to us and we would likely not use it for our purposes.

Figure 6.3: NB Confusion Matrix



CONFUSION MATRIX

		Actual	
		False	True
Predicted	False	150	130
	True	46	246

DETAILS

Sensitivity	Specificity	Precision	Recall	F1
0.765	0.654	0.536	0.765	0.63

Accuracy	Kappa
0.692	0.381

Figure 6.4: Map of Predicted Classifications

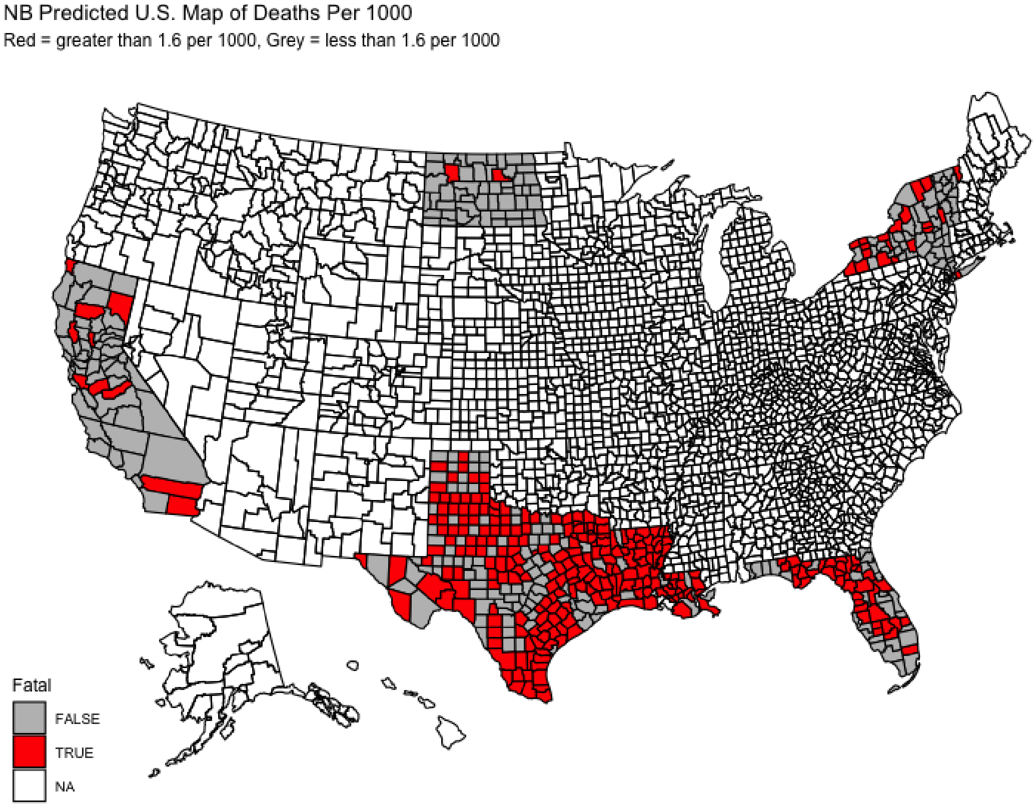
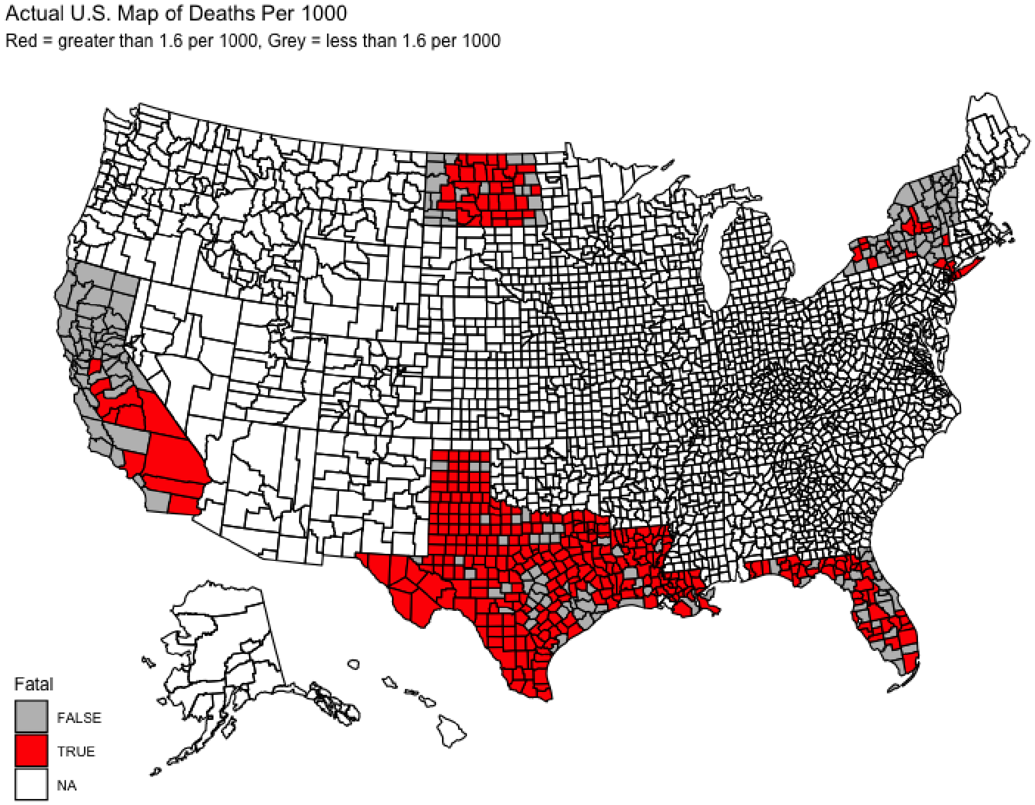

NB Predicted U.S. Map of Deaths Per 1000
Red = greater than 1.6 per 1000, Grey = less than 1.6 per 1000
Fatal
FALSE
TRUE
NA

Figure 6.5: Map of Actual Classifications


Actual U.S. Map of Deaths Per 1000
Red = greater than 1.6 per 1000, Grey = less than 1.6 per 1000
Fatal
FALSE
TRUE
NA

Classification Model 4 - K Nearest Neighbors

For our fourth model, we decided to use the K-Nearest Neighbors (KNN) method for classification. Below outlines our training and testing for our K-Nearest Neighbors model and displays our results.

Training

During training we saw that the best number of neighbors to use was 11 when using a tune length of 10. With this number of neighbors we obtained a best accuracy of 70.04% and an associated kappa value of 38.08%. Below in Figure 7.1 we can see a graph of the number of neighbors used and their associated accuracies. We can also see in Figure 7.2 the variable importance that each feature used during training.

Figure 7.1: Accuracy vs. Number of Neighbors

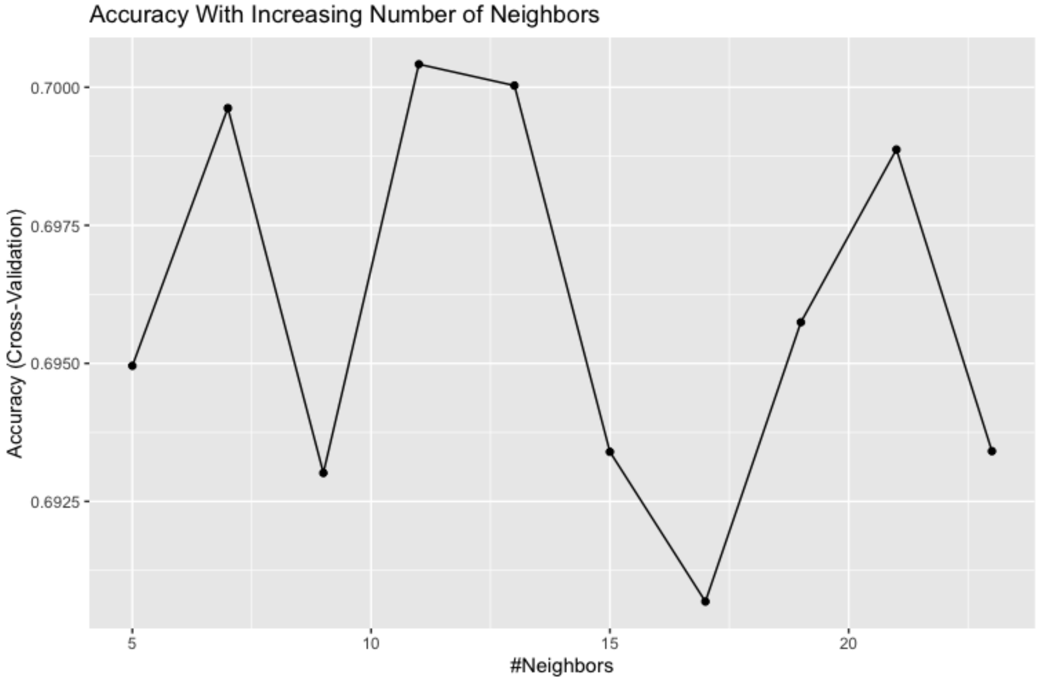
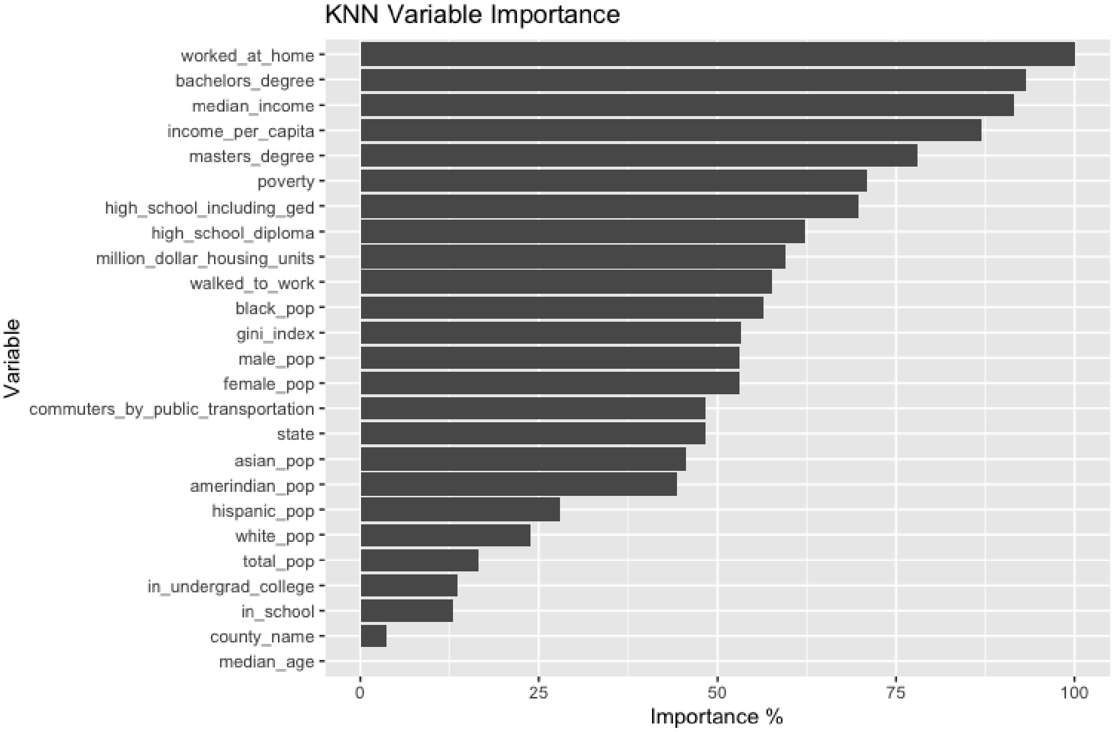


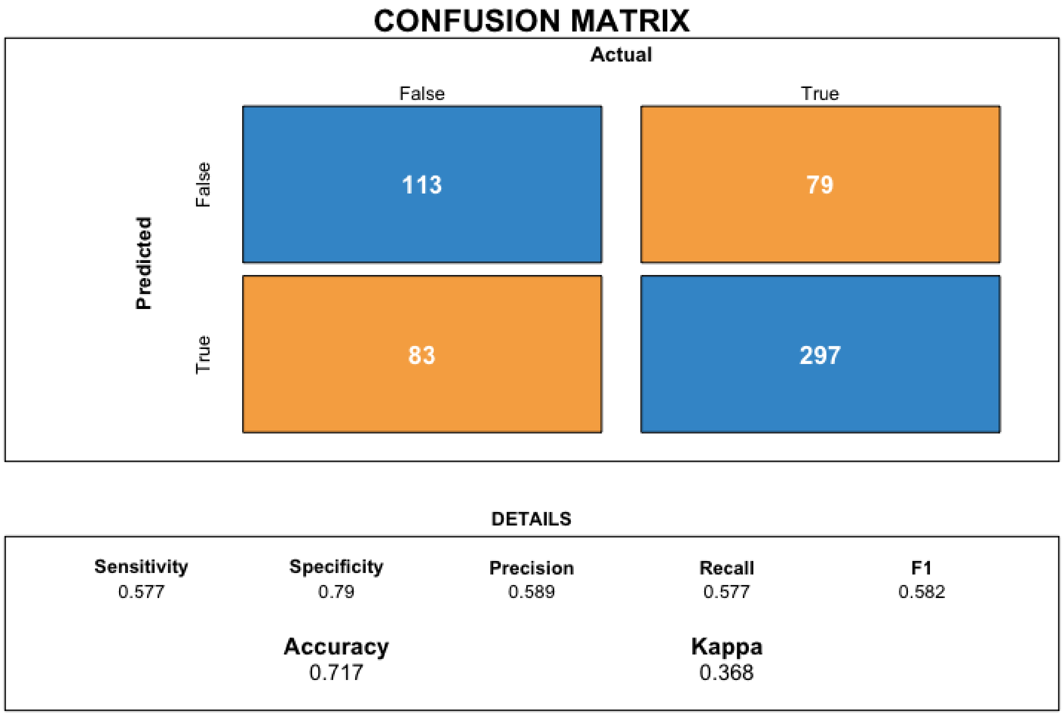
Figure 7.2: Variable Importance for KNN Classifier



Testing

During testing we observed an accuracy of 71.7% and an associated Kappa value of 36.8%. These results were better than our Naive Bayes classifier, but still not as good as our RF or RPART classifiers, performing only slightly worse than our RPART classifier. Thus, we do not feel as confident in using this model to predict a county as being fatal or not as we already have better alternatives. Below you can see a confusion matrix of our results in Figure 7.3. What is not shown here however is our no information rate and P-Value. Our no information rate was 65.73% and it is good that our accuracy was at least above that percentage. In addition, our P-Value was 0.001386 which was not super high but we have seen better in our other classifiers. Additionally, in Figures 7.4 and 7.5 we can see a comparison of performance from the actual classifications and our predicted classifications. The predictions were not super ideal but they were still way better than our Naive Bayes classifier and our false positive rate was much lower which made us more confident in using this model for predictions. Our precision, which is a ratio of the number of counties we predicted to be not fatal that were actually not fatal to the total number we predicted fatal, was not the greatest at 58.9% but it was still better than Naive Bayes. Regardless this was still slightly worrying to us as this model would deem an area as not fatal when it was actually fatal with a fairly decent probability. We did not like this as we would prefer to err on the side of caution when determining whether a county was fatal or not. However, we will discuss the overall performance of these models later on in this report.

Figure 7.3: KNN Confusion Matrix



CONFUSION MATRIX

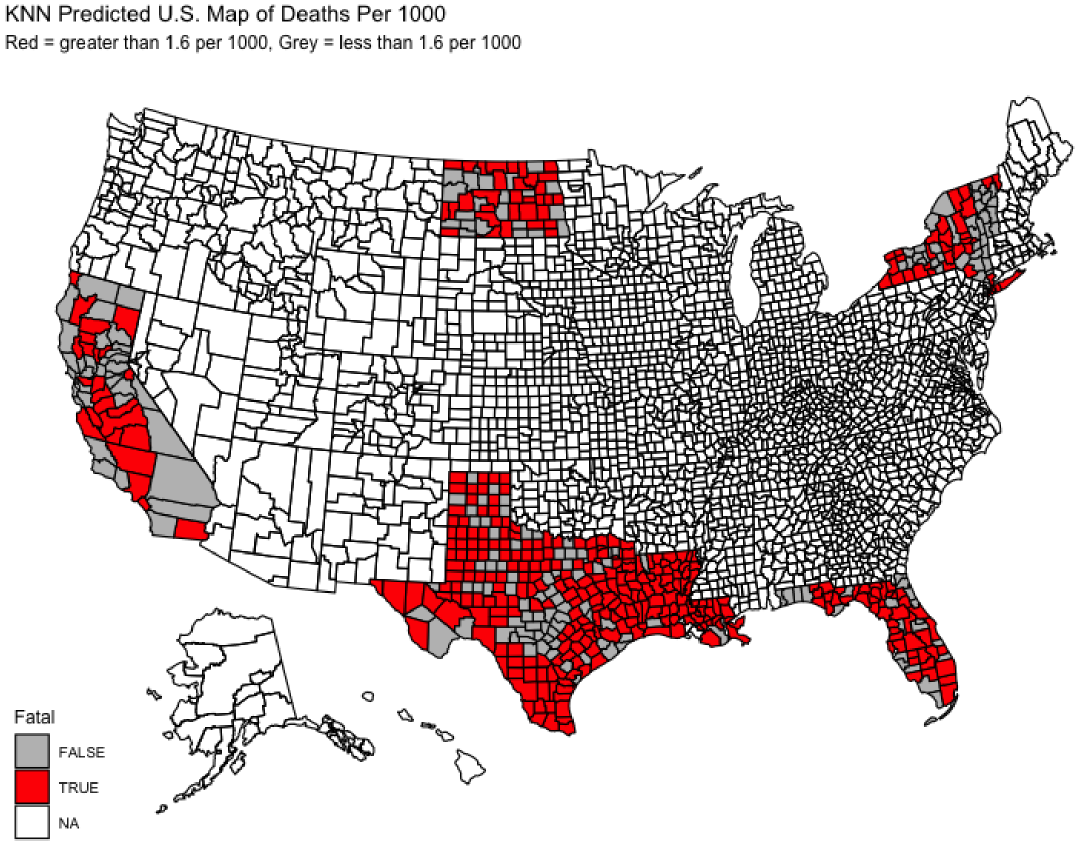
		Actual	
		False	True
Predicted	False	113	79
	True	83	297

DETAILS

Sensitivity	Specificity	Precision	Recall	F1
0.577	0.79	0.589	0.577	0.582

Accuracy	Kappa
0.717	0.368

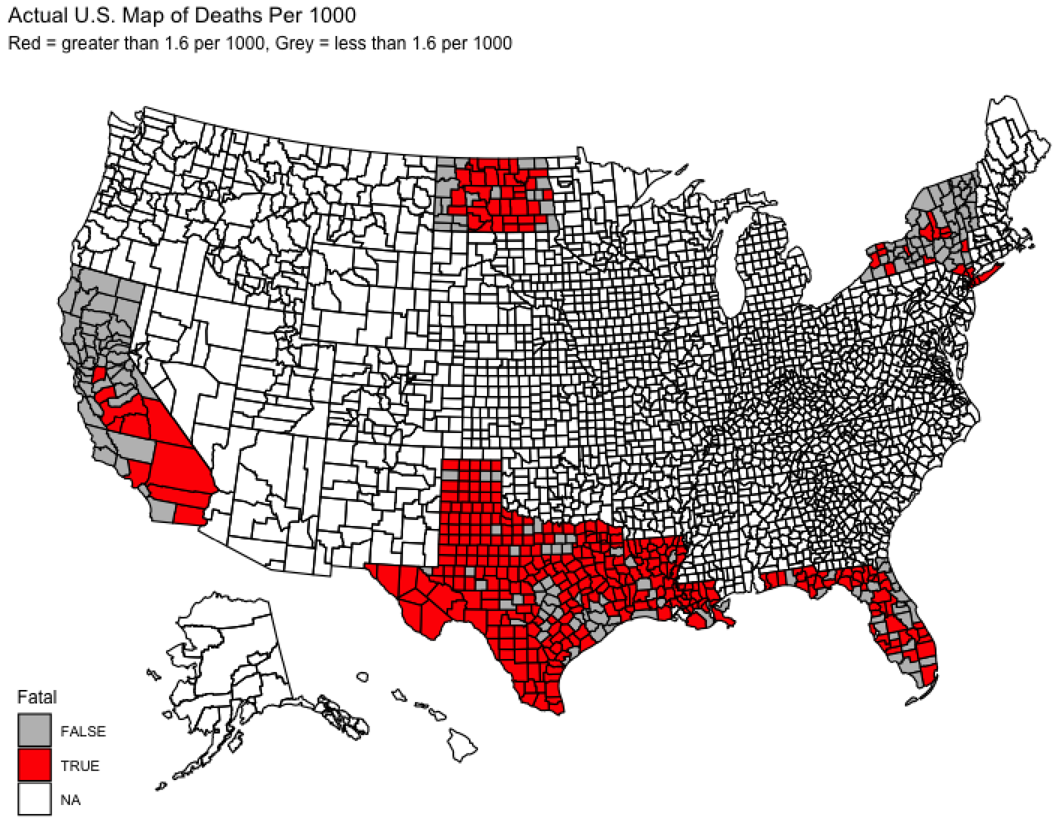
Figure 7.4: Map of Predicted Classifications



KNN Predicted U.S. Map of Deaths Per 1000

Red = greater than 1.6 per 1000, Grey = less than 1.6 per 1000

Figure 7.5: Map of Actual Classifications


Actual U.S. Map of Deaths Per 1000
Red = greater than 1.6 per 1000, Grey = less than 1.6 per 1000
Fatal
FALSE
TRUE
NA

Model Performance

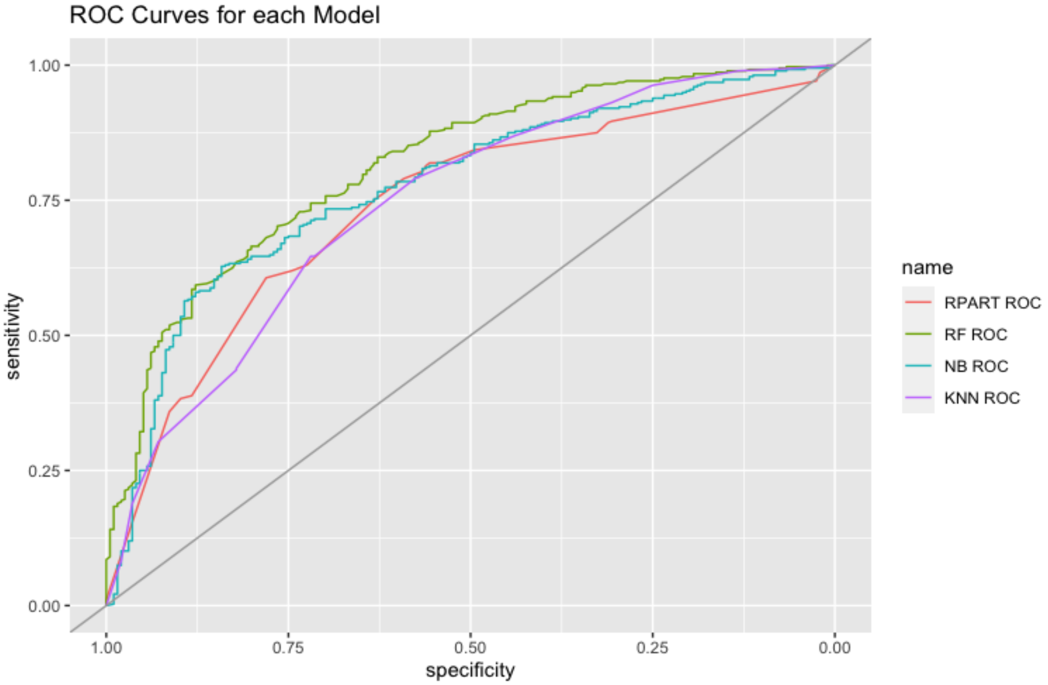
Now that we have trained and tested all four models, we can now evaluate each model's performance to determine if there are any statistical differences or advantages with any of the models.

We first look at the receiver operating characteristic (ROC) curve and the area under the ROC curve (AUC). The ROC curve plots the true positive rate (sensitivity) and the false positive rate (1 - specificity) at all classification thresholds. The AUC is the measurement of the space underneath the plotted ROC curve. Table 3 shows the AUC value for each of our models and Figure 8 shows the plotted ROC curve. To make sense of the data in the table and figure shown below, an ideal classification (predictions were all true positives and true negatives) model would have an AUC value of 1, meaning the sensitivity and specificity values are both 100%, and the false positive rate is 0%. To put it easily, the closer the curve is to the top left corner of the graph, the better the classification model. Additionally, the diagonal gray line shown in Figure 8 represents a random classifier, anywhere under that line is considered worse than a random classifier. From the data below, we can see that our best performing model is our Random Forest (RF) classifier with an AUC of 0.8143. Our second best is Naive Bayes with an AUC of 0.7757. Both our RPART and K-Nearest Neighbors models have similar AUCs at 0.7371 and 0.7422 respectively. From the data in Table 3 and Figure 8, we can start to assume our RF classification model is our best model, however, we run further tests on each model below to determine if these differences are statistically significant enough to deem it our best model.

Table 3: Area Under the ROC Curve

Model	Area Under the Curve
RPART	0.7371
RF	0.8143
NB	0.7757
KNN	0.7422

Figure 8: Model ROC Curves



The below figures give a detailed look into the evaluation and performance statistics between our four models. Figure 9 outlines the accuracy and kappa statistics for each model across all cross validation folds in a box plot format. In this Figure, the further right the box plot appears, the better the model performs. We can see that our Random Forest (RF) classifier performs the best in both accuracy and kappa, however, the kappa box plot has a very wide distribution between the max and min values. Our K-Nearest Neighbors (KNN) classifier has a much more consistent range in the evaluation statistics across all folds, which could make it a more reliable classifier. Both our Naive Bayes and RPART classifiers have very similar box plots in terms of location along the number line and box plot shape, however the NB classifier looks like it has a slight advantage over RPART. Figure 10 outlines the performance comparison between models based on their evaluation criteria (accuracy and kappa). In this figure, the box plots represent the performance differences between the models. If 0 falls in the range of the box plot distribution, there is not a statistically significant difference between models based on the evaluation criteria. For example, looking at the difference between RF and KNN (second to last row in Figure 10), we can see that the box plots cross over 0 in both accuracy and kappa, meaning there is no statistical difference in the way these two models perform. From Figure 10, we can see that RF is our only model that performs statistically better than other models. In the first row of Figure 10, the performance difference between RPART and RF is shown, because RF's performance is being subtracted from RPART's performance, the negative box plot range signifies RF performs better than RPART. In row 4, we can see that the positive boxplot range signifies RF performs better than NB because the NB performance is being subtracted from the RF performance.

Figure 9: Model Evaluation Statistics

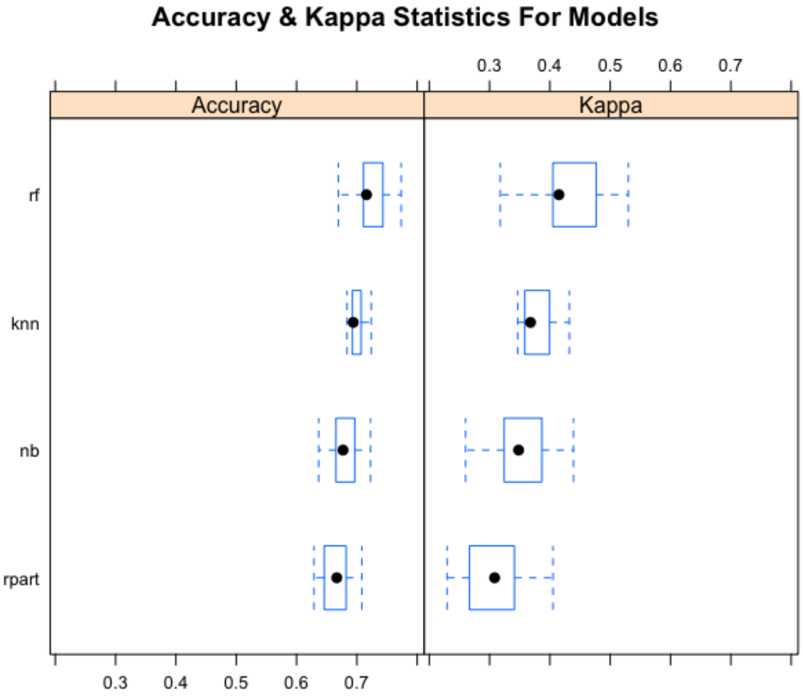
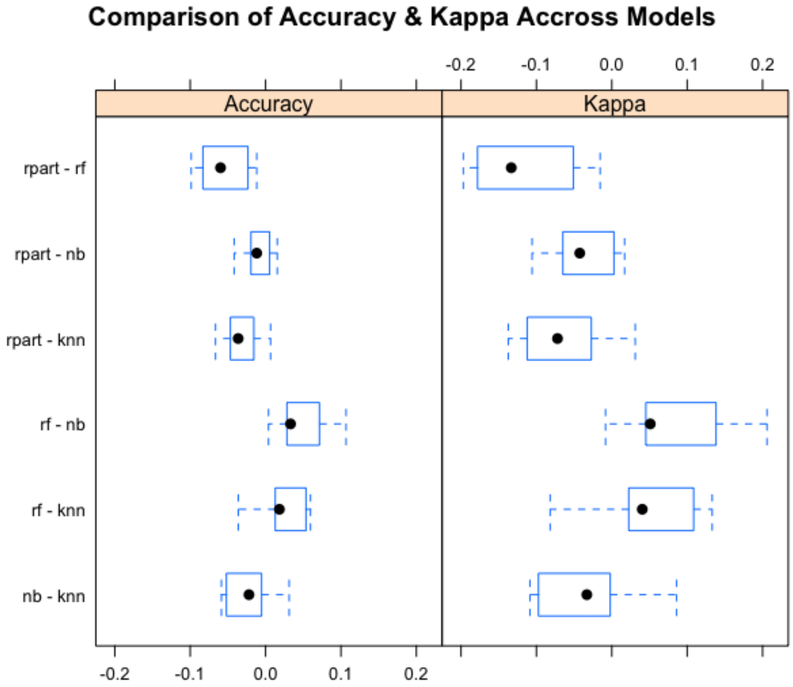


Figure 10: Model Performance Comparison



IV. Evaluation

Our stakeholders, government healthcare entities in the state of Texas, would use our model as an extra tool to help make decisions regarding a county's risk of fatality from COVID-19. It would be a useful tool to see which counties are likely the most at risk, and least at risk, for a potential fourth wave of COVID-19 and would help guide their decisions on where resources should be sent. Government entities could also use it as a guide on what restrictions they put forward. For example, if a county is classified as fatal, this means more resources, such as doctors and medicine, should be sent to that county to help fight infections and hopefully lower the number of deaths. Local governments could also issue a mask mandate and close or lower the occupation limit in businesses. However, our model should not be the only tool used. Stakeholders should use several other resources to make the final decisions on resources and restrictions.

The model's value could be assessed by looking at the predictions made by the model and how they compare to the actual response and result of whether a county became fatal over time. This would essentially mean that the model could be used to predict whether a county would be fatal, then based on this prediction, actions would be taken. Then over time, given the actions taken, one could evaluate the performance of the model in determining its effectiveness.

V. Deployment

When using the best classifier that we were able to obtain, that being Random Forest, in practice, we would likely use it as a preliminary screening classifier. This is because our model was not amazing or super accurate by any means but it did perform decently. Therefore, we would not want our model to be the sole provider of information when making decisions regarding the degree of fatality within a county. We would certainly want to pull in additional information from other sources to make that decision. Ultimately we would want to make it very clear that our model should not be used as the sole depictor of whether a county is fatal or not based on census data and should be used as a secondary reference to more reliable information.

To ensure that our model would stay up to date with predicting whether a given county is deemed fatal or not, we would ensure to update the data every 2-3 weeks, retraining the model each time. This way we wouldn't be updating the model too frequently and large shifts in terms whether a county would cross the fatal classification line, in either direction, could be captured.

VI. Conclusion

In conclusion, when determining whether a county was fatal or not based on governmental census data, we found that using the Random Forest classification method was the best performing model by far. Using this model we were able to get the highest accuracy and Kappa value on our test data of 74.1% and 43.4% respectively. That being said, the classifier was not perfect by any means. It was the best model we could generate, however, the false positive rate and false negative rates were still worrying. We were primarily focused on the false positive rate here as we did not want to deem a county as not fatal when it was actually fatal. With the Random Forest model we still had 80 predictions that fell into this category when we performed testing. Thus, we deemed our model useful but moreso as a secondary form of providing analysis into determining whether a given county was fatal or not.

Our worst classifier by far during training and testing was our Naive Bayes classifier that resulted in an accuracy and Kappa of 69.2% and 38.1% respectively. These values were far worse than our best model and even more worrying was the false positive rate that we observed. In our testing, we found that the Naive Bayes classifier was classifying 130 counties as being not fatal when they were actually fatal. This greatly worried us as our true positive rate was 150 which is very close to that false positive number. However, during model performance analysis, we found that both Naive Bayes and RPART performed similarly, therefore we can conclude that these models were our worst performers. We determined that we would not use these models whatsoever in predicting whether a county was fatal or not for our stakeholders as we had better performing models to use.

All in all, none of our classifiers were amazing but they were fairly decent. Upon reflection, there may be some additional tuning or feature selection that we could perform to get a better classifier or maybe a better method exists for our classification task. Regardless, in the end our best classifier was still able to perform fairly decently, and we would recommend it be used complimentary to other information when making a prediction on whether a county is fatal or not with regards to COVID-19.

VII. References

[1] COVID-19 Fourth Wave Information.
https://www.washingtonpost.com/health/2021/04/04/covid-fourth-wave/

[2] COVID-19 cases plus census dataset.
https://smu.instructure.com/files/4270322/download?download_frd=1

[3] COVID-19 cases TX dataset.
https://smu.instructure.com/files/4270321/download?download_frd=1