Understanding voter migration between US Presidential Elections in 2016 and 2020

18 February 2021

ST309 Group Project Report

Table of Contents

Introduction

Having won the 2016 presidential election in a self-proclaimed "landslide victory" (Seipel, 2016) of 306 to 232 over Hillary Clinton, Trump himself has now lost his 2020 re-election bid by the same electoral college margin. The reasons for his defeat are being speculated and studied, and the re-election held in the context of the COVID-19 pandemic added yet another dimension of political consideration for voters. The problem tackled in this research is hence to understand the shift in voter movement towards Trump from 2016 to 2020, with a special focus on the influence of the COVID-19 pandemic.

Our report is structured as follows. We began with a literature review on news coverage of the 2020 election to identify factors which may be relevant in explaining changes in Trump's vote share. Next, we created our own dataset, which contains election results and demographic variables on county-level, cleaned and merged from multiple sources. Subsequently, we applied both classification methods and multiple linear regression to determine the best model in explaining the direction and size of changes in Trump's voter share. Based on the performance with training and testing data, we selected the 7-variable linear model constructed from cross-validation to be our best model. Afterwards, we interpreted the significance of each of these seven variables with applied understandings from US electoral politics. Finally, we conclude with how our findings contribute to the existing literature on electoral outcomes in the US while noting ways in which our research can be improved for future comparative analyses.

Problem Description and Literature Review

Electoral behaviour is an important subject as voting is a democratic exercise of one's political opinions; its outcomes have historically pointed to underlying social or economic challenges faced by certain parts of the population. Therefore, this study of electoral outcomes in the United States will not only corroborate or refute existing assertions made in popular post-election coverage but also contribute to a wider understanding of American society and its progress. In particular, the goal of our analysis is to understand large-scale voter movements between the 2016 and 2020 presidential elections.

Existing reporting from political scientists and newspapers have proposed many potential explanations for the voter movement in the 2020 election, particularly in the areas of ethnicity, education, gender, and income/wealth. All big newspapers have framed Biden's victory as being largely dependent on stronger turnout from Black Americans: The New York Times on its first page (Eligon and Burch, 2020, p.1), The Washington Post (Crumpton, 2020), and The Guardian (Albright, 2020). Meanwhile, they have also noticed an opposite shift among Hispanic voters who have voted for Trump in much greater numbers in 2020 than they had in 2016 leading him to a victory in the key state of Florida (Fessenden et al., 2020). The white population has reportedly been split according to level of education (college graduates have turned out more strongly against Trump), their gender (women voted more Democratic), and socioeconomic status (Trump further increased his share among poorer, rural

areas) (Bryant, 2020). Another point of contention has been whether the number of COVID-19 cases and deaths had a significant impact on the election outcome. While NPR argues that higher COVID-19 death rates are associated with more support for Trump on the county-level (McMinn and Stein, 2020), others find no statistically significant relationship between COVID-19 death rates and Trump's vote share in 2016 and 2020 on the state-level (Masket, 2021).

In this literature, compelling theories to explain changes in Trump's relative vote have been proposed, yet these analyses are often inadequate as they focus on a limited set of factors and only provide empirical observations in support of their claims. For example, multiple news outlets have noted that more cases in a county have led to higher support for Trump, yet they did not provide any reasonable explanation and often failed to weigh for other variables, such as that Trump supporters are on average older and more susceptible to a severe form of the disease (e.g., McMinn and Stein, 2020).

Therefore, in our project, we plan to address the inadequacy of the existing literature by using data analysis to compare the relative importance of a wide range of factors in explaining voter movement between the 2016 and 2020 election. Although we cannot directly infer causality from correlations in our observational analysis, systematic analysis of data will give us empirical evidence to assess explanations in the literature. By conducting our own independent analysis, we hope to uncover new possible correlations and invite different perspectives on the data, which can contribute to a deeper understanding of voting behaviour and patterns and add explanatory value to this historic election.

Data Source

To explain voter migration or change in candidate Donald J. Trump's relative vote share in the US presidential election in 2016 and 2020, we needed data in two areas: election results and factors which could explain these results. We also decided to collect these cross-sectional data at county-level, in order to have a sufficient number of observations for our data analysis.

For the 2016 election, we used verified county-level results available on MIT Election lab (2018a). For the 2020 election, it was more difficult to obtain official results because the election happened recently and there was a delay in the results' certification due to the COVID-19 pandemic and legal challenges. Instead, we used a dataset from GitHub where the authors scraped county-level vote share for each candidate from results published on The Guardian, townhall.com, Fox News, Politico, and the New York Times (McGovern, 2020). We verified the accuracy of these scraped data with a few randomised checks.

Next, for factors which might explain voter migration, we also have multiple data sources. For COVID-19 factors, we decided to use the cumulative number of COVID-19 cases and deaths in each county published by The New York Times (2020a) and we selected 3 November 2020, the election day, as the cut-off point. We think cases and deaths are good proxies because the higher the number of cases

and deaths, the more severe the impact COVID-19 will have on the county. This could then influence voters' choice of presidential candidate because Trump and Biden proposed very different COVID-19 policy responses. For demographic characteristics, we relied on two data sources. The first one is Atlas of Rural and Small Town America published by the Economic Research Service (ERS) of the US Department of Agriculture (2020). The second one is another dataset from MIT Election lab (2018b), originally designed for analysing the context of the 2018 US midterm election. Both sources contain county-level observations on a variety of socioeconomic variables, such as race and ethnicity, migration, education, household size, income and unemployment in our model. Although some demographic data were taken from the last US population census in 2010, we believe the race and ethnicity composition of US counties has remained relatively stable over the past decade. Therefore, these datasets are still relevant and appropriate for our analysis.

Data Preparation

Our goal in data preparation was to create a matrix such that each row represented a county and each column represented an independent or dependent variable. All datasets used for our analysis contained a standardised county identification number, called Federal Information Processing Standards (FIPS) code, that served as the basis to merge the columns of all datasets into one. In the Rural Atlas datasets from ERS, rows containing state summaries had to be deleted while new variables that could be of analytical use were added, e.g., the change of counties' unemployment rate while Trump was in office. Missing COVID-19 data for six counties in New York, Nevada, and Texas were added from official state agencies' reports (USA Facts, 2020). Subsequently, we created an additional independent variable that contains the per capita COVID-19 cases and deaths until Election Day 2020 (`cases_per_100000` and `deaths_per_100000`) rather than absolute numbers.

For our dependent variables, we could not use the election data directly because the 2016 and 2020 presidential election data reported solely the absolute number of votes and we aimed to compare the election results in percentage points of vote share. Therefore, we created two new variables: first, a continuous variable called `voter_movement_to_GOP` which records the change of candidate Donald Trump's relative vote share from 2016 to 2020 for our regression analysis. Second, a Boolean variable `GOP_increase` reflecting whether Trump gained votes for our two-way classification models. For example, if Trump received 60 per cent of the votes in 2016 and 55 per cent of the votes in 2020, the first dependent variable would equal `-0.05` and the second show `FALSE`.

We faced little problems from missing values in our datasets. Only two observations contained missing values and were subsequently deleted. However, the state of Alaska is not divided into counties and was differently divided in so-called 'organised and unorganised boroughs' in the election and the census (HuffPost, 2016) and could therefore not be used for our analysis. Lastly, Puerto Rico, which

belongs to the United States but is not allowed to vote in federal elections (Ponsa-Kraus, 2020, p.19) consequently lacks the election data necessary for our analysis and was removed.

Overall, our dataset contains data on 3,111 out of 3,143 US counties and county-equivalents. 185 columns of our dataset were reduced to 69 after removing duplicates, outdated census data, and variables containing absolute values rather than relative values. Absolute values are futile as counties differ dramatically in size (the smallest county is Loving County in Texas with 169 inhabitants, the biggest is Los Angeles County in California with a population greater than 10 million). All other independent variables that could in any way affect voting behaviour remained. We ended up with a 3111 x 69 matrix which we split into two datasets: 1,000 for training and 2,111 for testing purposes. The size of the training data `n = 1,000` was determined so that the training data set is large enough to allow for well-founded inferences but small enough that the performance on the training data returns meaningful results (James et al., 2013, p.166).

Data Description

After cleaning our data sets, we conducted a preliminary descriptive analysis to understand `voter_movement_to_GOP`, the dependent variable which we seek to explain. First, we calculated the five key summary statistics for this variable and plotted a histogram to examine its distribution. The greatest shift *towards* Trump (i.e., the largest increase in Trump's county-level vote share in 2020 compared to 2016) was 28.1 percentage points in Starr, Texas. The greatest shift *away* from Trump was a 14.8-point decrease of his vote share in Jackson, Missouri. The median change in vote share was 1.5 percentage points, with an interquartile range of 2.7 percentage points. Looking at these summary statistics suggested that there are potential outliers in our dependent variable. Furthermore, based on the histogram (*Figure 1*), we could also see that a few counties experienced exceptionally large shifts away from or towards Trump. However, overall, the variable was relatively symmetrically distributed and surprisingly, although Trump lost the election, more counties have experienced a voter movement towards Trump than away from him in the 2020 election. This might reflect the fact that Trump gained support from more rural counties that tend to have a lower number of inhabitants (*Figure 2*).

Next, we used Tableau to create visual representations of `voter_movement_to_GOP` and two key variables of interest related to COVID-19 (`cases_per_100,000` and `deaths_per_100,000`). The first map below illustrates the voter movement towards Trump—blue represents a decrease in vote share, or shift away from Trump, while the gradation to red represents a positive shift towards Trump in the 2020 election (*Figure 2*). Our second map shows the COVID-19 deaths (*Figure 3*) and cases (*Figure 4*) per 100,000 by county with red indicating counties which were impacted more by COVID-19. Since red regions in *Figure 2* roughly correlated with red regions on the *Figure 3 and 4*, we suspected there might be a positive correlation between these variables. Indeed, if we do not control

for any other factors, we can find a very weak positive correlation coefficient of `0.11` between `voter_movement_to_GOP` and `cases_per_100,000` and correlation coefficient of `0.045` between `deaths_per_100,000` and `voter_movement_to_GOP` (*Table 1*). However, to properly understand the presence and the significance of COVID-19 and other factors (drawn from our literature review) in predicting the shift of votes towards Trump, we proceeded to conduct further analysis.

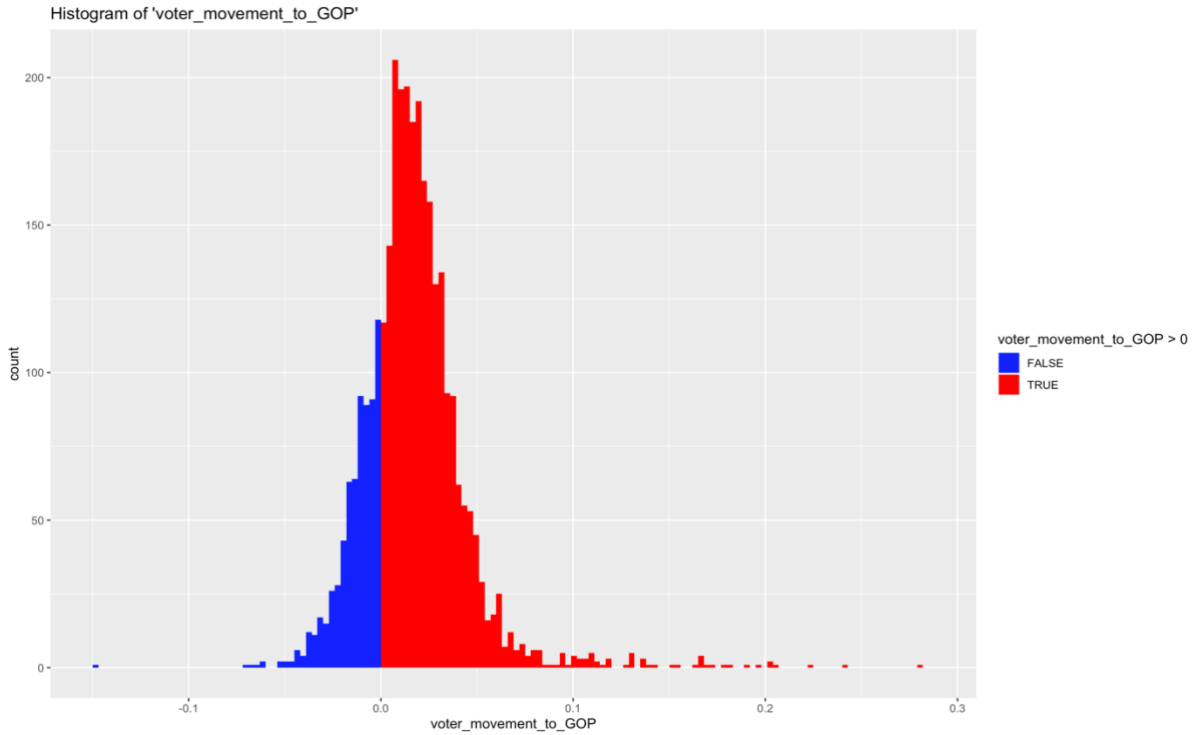


Figure 1: Distribution of the target variable `voter_movement_to_GOP`

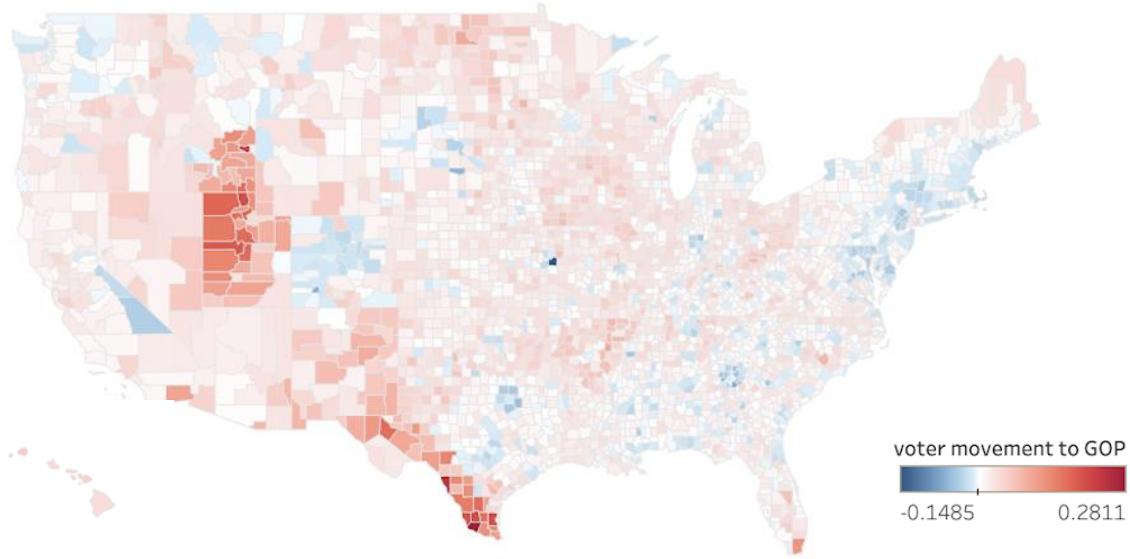


Figure 2: Voter movement towards Trump at the County Level

Correlation Matrix

	cases_per_100000	deaths_per_100000	voter_movement_to_GOP
cases_per_100000	1	0.510	0.110
deaths_per_100000	0.510	1	0.045
voter_movement_to_GOP	0.110	0.045	1

Table 1: Correlation between voter movement and COVID-19 cases and deaths

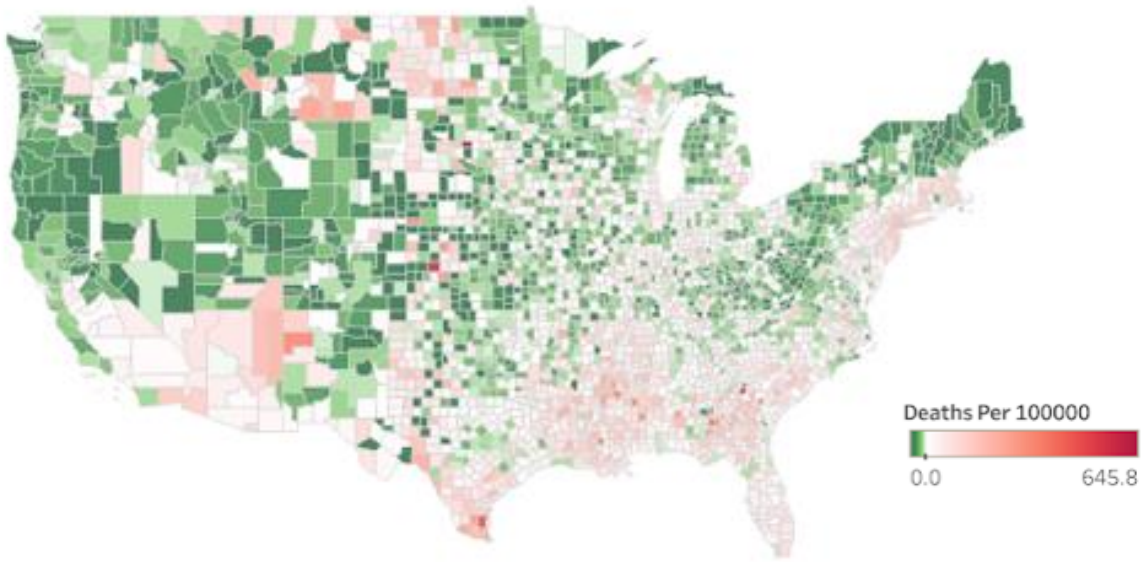


Figure 3: COVID-19 deaths per 100,000 inhabitants

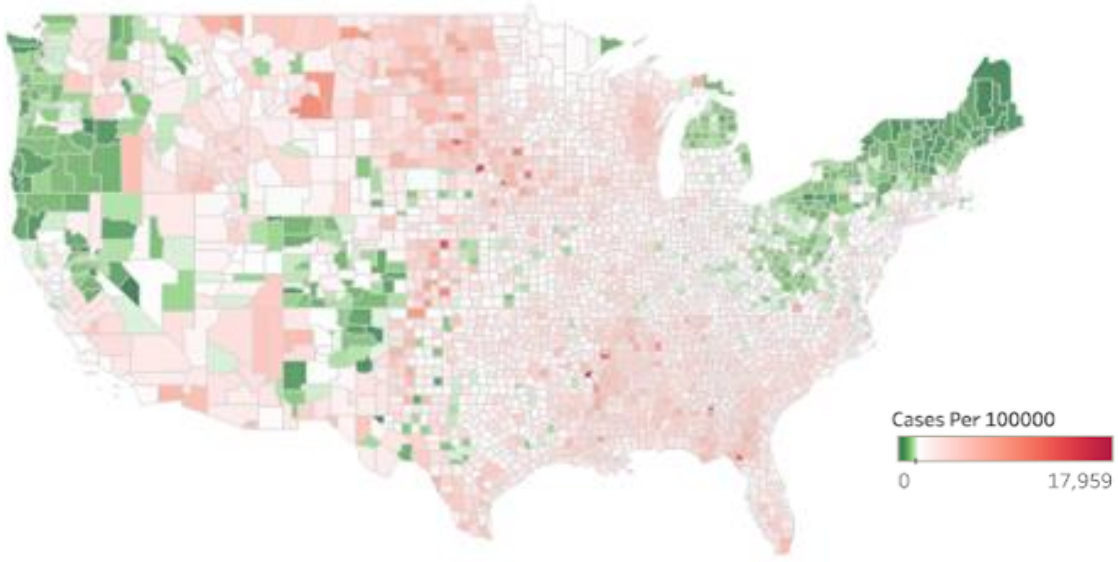


Figure 4: COVID-19 cases per 100,000 inhabitants

Methodology

We conducted our data analysis in two stages. In stage one, we used classifiers to identify which factors are relevant in explaining the *direction* of voter migration. In stage two, we used linear regression to analyse which factors can explain the *direction and size* of changes in Trump's county-level vote share.

Stage One: Classification

Here we used our binary dependent variable `GOP_increase` in two-class classifications. Classifiers are desirable because they do not assume a linear relationship, which reduces collinearity concerns and allows our model to be more flexible. Moreover, tree-based models are computationally efficient and can easily search through our large number of variables.

We began with a simple classification tree which has 25 terminal nodes. Next, we used Boosting and Random Forest to reduce the variance of the tree. Finally, we applied non-parametric K-Nearest Neighbours (KNN) methods with 3-NN and 5-NN classifiers to account for the possibility of non-linearity. Afterwards, we compared all classifiers according to two criteria: misclassification rate and AUC values in ROC graphs (see *Table 2* and *Figure 5*). We defined a true positive as the case where both the model prediction and test data tell us there was an increase in Trump's vote share. As shown in *Table 2*, Random Forest performs the best with the lowest misclassification rate and the highest AUC value. This is because Random Forests force each split to consider only a subset of predictors, which decorrelates the trees and makes predictions less variable and more reliable (James et al., 2013, pp.319–320). Furthermore, the five most important variables in our Random Forest model were `Ed5CollegePlusPct`, `PopDensity`, `ForeginBornCaribPct` and `PctEmAgriculture` and `PctEmpServices` (see *Figure 6*). This suggests that education level, population density, ethnicity and sector of employment are useful in explaining the *direction* of voter migration.

However, classifiers have limitations. They treat all instances of increase in Trump's vote share as the same, whether it was by 0.0001 per cent or by 20 per cent. They also do not tell us how much change in each variable can lead to a 1 percentage point change in Trump's vote share between 2016 and 2020. Nevertheless, this is still a good starting point. For example, the poor performance of KNN methods gives us more confidence in a linear relationship between our predictors and the change in Trump's county-level vote share. Therefore, we will apply linear regression in the next stage, in order to fully understand the *direction and size* of voter migration.

Classifier	Misclassification rate on testing data	Area Under the Curve (AUC) Value
Simple tree	18%	0.689
Bagging	14%	0.885
Random Forest	13%	0.888
3-Nearest Neighbour	19%	0.736
5-Nearest Neighbour	18%	0.755

Table 2: Five classifiers and their performance on our testing data

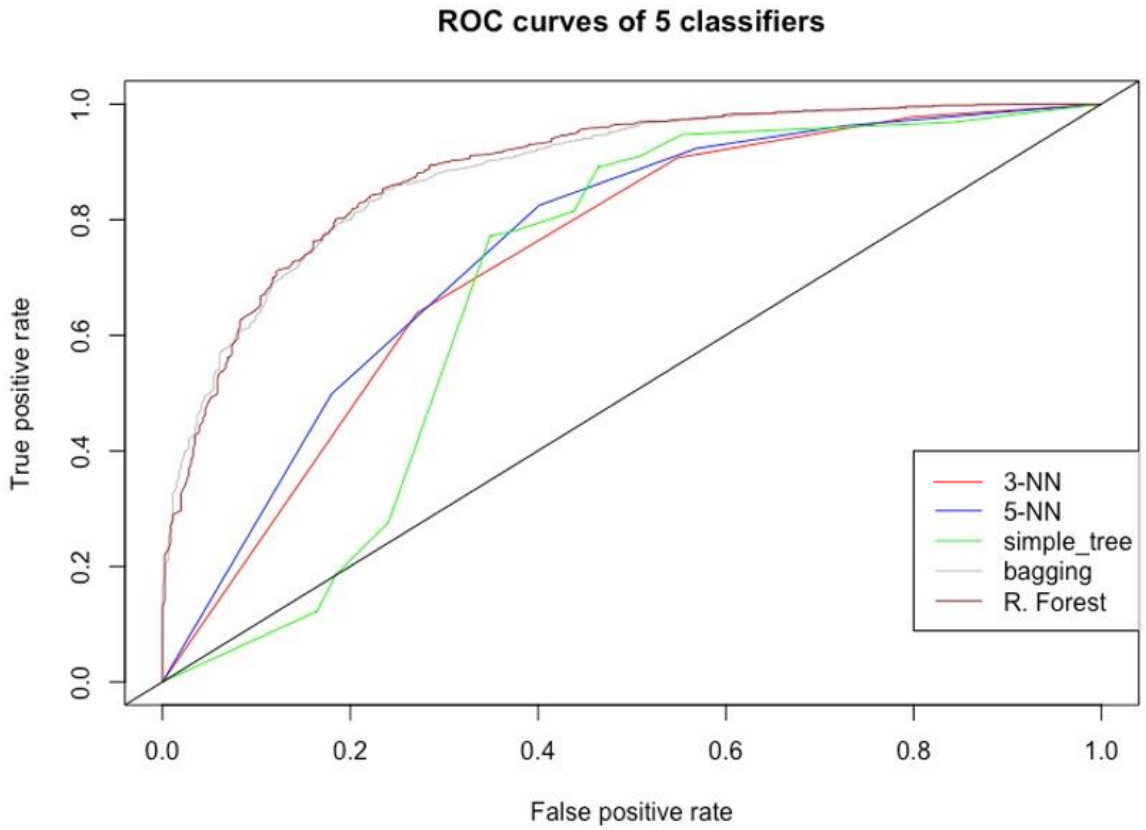


Figure 5: Five classifiers and their performance on our testing data

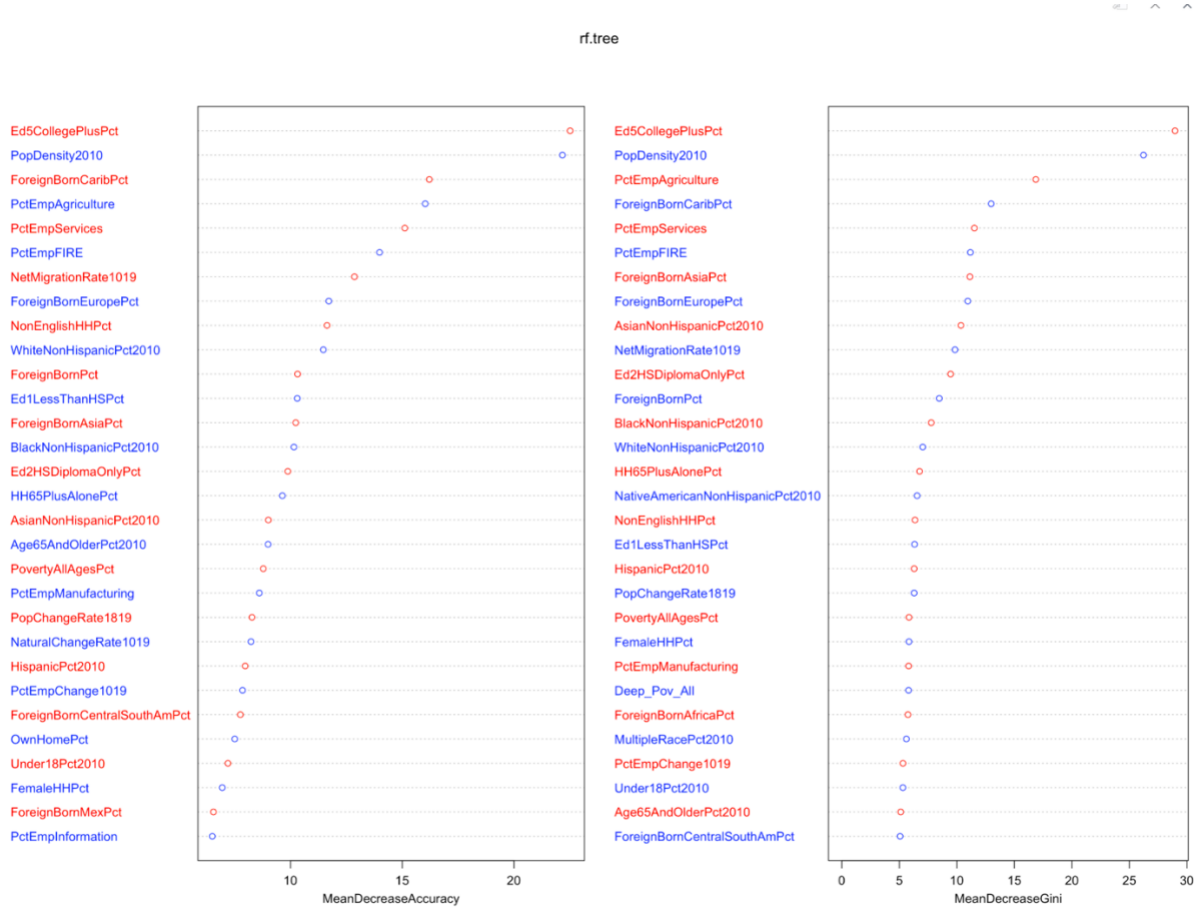


Figure 6: The importance of variables in the random forest model

Stage Two: Linear Models

Here we used our continuous dependent variable `voter_movement_to_GOP` and conducted multiple linear regression analyses. We removed one category from education and employment variables respectively (`PctEmpInformation` and `Ed3SomeCollegePct`) to avoid perfect collinearity. This left us with 47 potential predictors. Theoretically, best subset selection is preferred to stepwise, forward, and backward, because it considers all possible combinations and provides a globally optimal solution. However, in practice, to calculate the best subset is computationally intensive and infeasible for over 40 predictors (James et al., 2013, p.207). Additionally, a linear model with a large number of variables is difficult to interpret and prone to overfitting. Therefore, we made a compromise where we applied best subset selection, but only chose to look for best models with a maximum of 10 predictors (`nvmax=10`).

While the best subset approach returns linear models with the lowest training error rate (smallest RSS or R-squared), the training error is not a good estimation of the test error rate. We want to select the model with the lowest estimated test error because this model is likely to explain more variation in the change of Trump's county-level vote share in the test data. First, we indirectly estimated test error by making a mathematical adjustment (BIC, Cp, Adjusted R-squared) to the training error to "account for the bias due to overfitting" (James et al., 2013, p.210). All three criteria recommended the same 10-variable model (`lm_math` or model 1). Next, we used the 10th-fold cross-validation (cv) approach and built a 7-variable model (`lm_cv` or model 2) with the lowest cv errors. The advantage of cv is by splitting the data into 10 folds and selecting one of them as a validation dataset, it estimates the test error directly. We summarised the two linear models below (see *Table 3*).

Evaluation of Models

At first glance, the performance of both models on training data is similar. Both F-statistics strongly reject the null hypothesis that all coefficients in the regression are zero. Each individual coefficient is also statistically significant. Hence the relationships which we observe in these models most likely exist in reality and are not just due to sampling variation. Furthermore, the direction of the effect (i.e., plus or minus signs of each coefficient) is consistent across both models. Finally, model 1 has a slightly higher R-squared than model 2 potentially because it contains more independent variables. While model 2 explains 34.5 per cent of the variation in voter movement, model 1 explains around 36.4 per cent.

Next, we checked for underfitting by examining residual plots of both models. In the `Residual vs. Fitted` and `Scale-Location` plots (*Figure 7* and *8*), both models generally have patternless residuals. The second normal Q-Q plot is more problematic because for both models, we observe some heavy tails which exceed the normal range of `[-2,2]` for standardised residuals. In the `Residual vs. leverage` plots, we also find some observations which have large residuals and leverage. This suggests that our model potentially suffers from underfitting which could explain the low R-squared on training data.

Linear Models

	Dependent variable:	
	voter_movement_to_GOP	
	(1)	(2)
PctEmpManufacturing	0.0004*** (0.0002, 0.001)	
PctEmpServices	0.0005*** (0.0002, 0.001)	
NetMigrationRate1019	-0.0004*** (-0.001, -0.0002)	-0.0004*** (-0.001, -0.0002)
NaturalChangeRate1019	0.002*** (0.001, 0.003)	
Age65AndOlderPct2010	0.001*** (0.001, 0.002)	
HispanicPct2010	0.0005*** (0.0003, 0.001)	0.001*** (0.0003, 0.001)
NonEnglishHHPct	0.004*** (0.003, 0.004)	0.004*** (0.003, 0.005)
Ed5CollegePlusPct	-0.001*** (-0.002, -0.001)	-0.001*** (-0.001, -0.001)
FemaleHHPct	-0.002*** (-0.002, -0.001)	-0.002*** (-0.002, -0.001)
ForeignBornCentralSouthAmPct	-0.003*** (-0.003, -0.002)	-0.003*** (-0.003, -0.002)
cases_per_100000		0.00000*** (0.00000, 0.00000)
Constant	0.013 (-0.007, 0.033)	0.053*** (0.047, 0.060)
Observations	1,000	1,000
R2	0.364	0.345
Adjusted R2	0.357	0.340
Residual Std. Error	0.023 (df = 989)	0.023 (df = 992)
F Statistic	56.532*** (df = 10; 989)	74.518*** (df = 7; 992)
Note:		*p<0.1; **p<0.05; ***p<0.01

Table 3: Estimates of the coefficients of the model 1 (`lm_math`) (left) and the model 2 (right) (`lm_cv`)

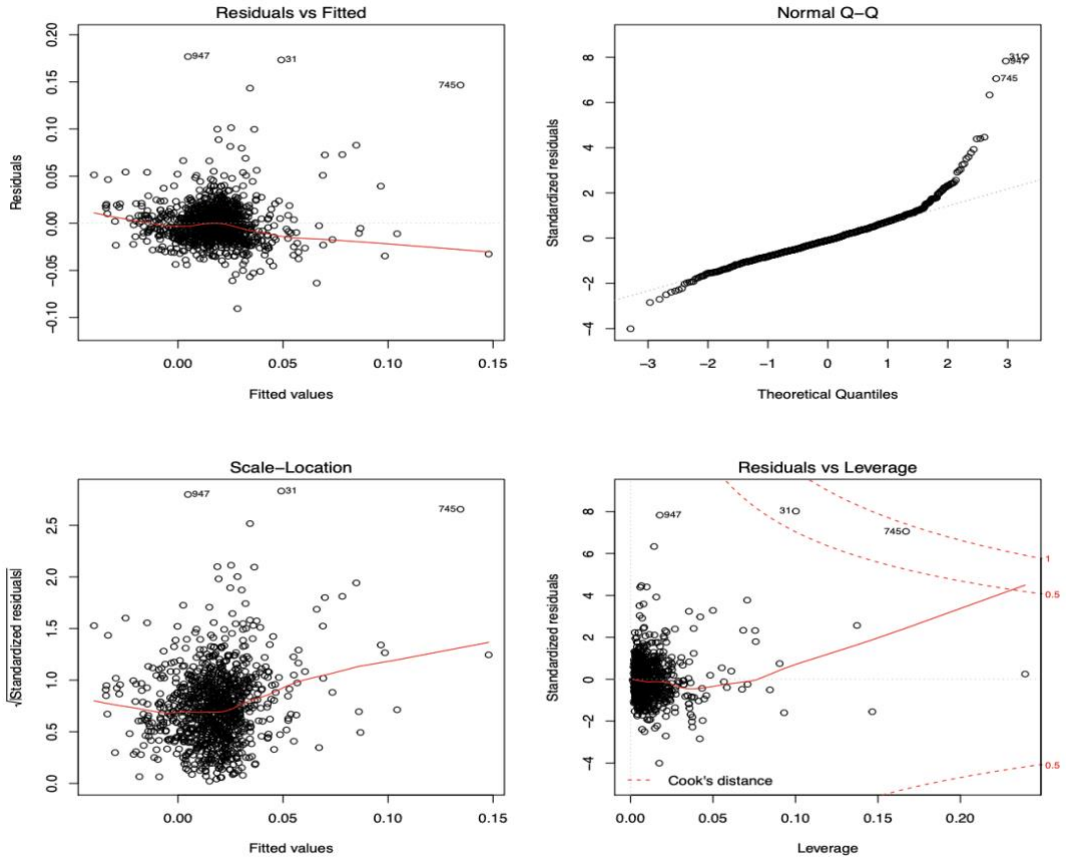


Figure 7: Diagnostic plots of model 1 (`lm_math`)

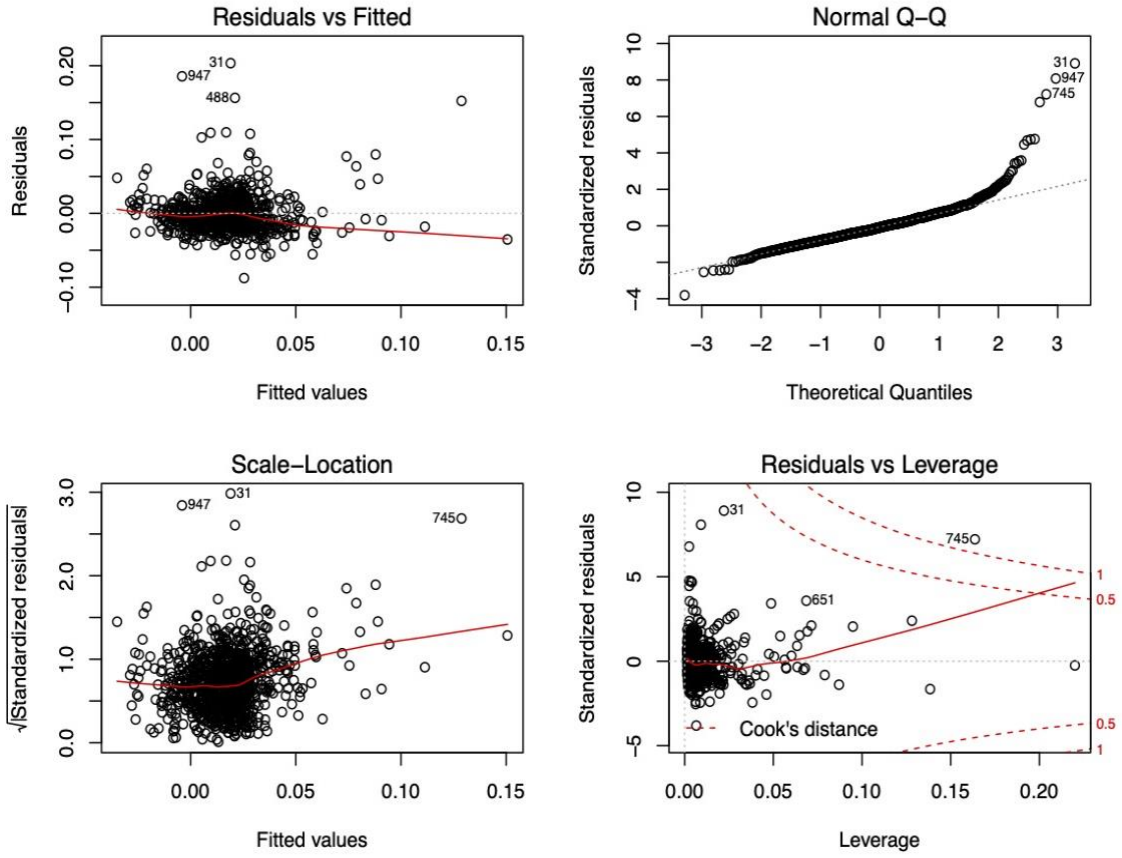


Figure 8: Diagnostic plots of model 2 (`lm_cv`)

```
Interpreting the CV Model
=================================================================
                                    Dependent variable:
                                -------------------------------
                                   100 * voter_movement_to_GOP
-----------------------------------------------------------------
NetMigrationRate1019             -26.117*** (-26.137, -26.097)
HispanicPct2010                   19.903*** (19.883, 19.922)
NonEnglishHHPct                     2.504*** (2.413, 2.595)
Ed5CollegePlusPct                 -8.385*** (-8.402, -8.369)
FemaleHHPct                       -5.717*** (-5.753, -5.681)
ForeignBornCentralSouthAmPct      -3.747*** (-3.814, -3.681)
cases_per_100000              7,640.491*** (7,640.491, 7,640.491)
Constant                            0.188 (-0.433, 0.808)
-----------------------------------------------------------------
Observations                                  1,000
R2                                            0.345
Adjusted R2                                   0.340
Residual Std. Error                     2.309 (df = 992)
F Statistic                          74.518*** (df = 7; 992)
=================================================================
Note:                                *p<0.1; **p<0.05; ***p<0.01
```

Table 4: Model 2 Adjusted for Interpretation. The inverse of the β estimates showing the change of the respective predictor that is associated with a 1 percentage point increase in Trump support.

Nevertheless, both models still performed well on test data. We evaluated the performance of our models by calculating actual test error rate or Means Squared Error (MSE) using our test data. We compared both models to the benchmark of predicting test data with the mean value of training data. Both of our models performed better than the benchmark. Model 1 has a slightly lower test MSE, but this difference is negligible. Overall, we are still satisfied with our models given that they beat the benchmark performance of goodness of fit in predicting test data. In particular, we prefer the model 2 (`lm_cv`) because it achieves similar performance in both training and test data with only seven variables. We will now focus our interpretation of results on model 2.

Interpretation of Models

As shown in *Table* 4 above, an increase in Trump's share of votes is correlated with the percentages of residents of Hispanic descent (`HispanicPct2010`), non-English speaking households (`NonEnglishHHPct`), and per capita COVID-19 cases (`cases_per_100000`). A decrease in Trump support is correlated to the percentages of residents with a college degree (`Ed5CollegePlusPct`), residents born in Central and South America (`ForeignBornCentralSouthAmPct`), the ratio of female-headed households (`FemaleHHPct`), and the net migration rate between 2010 and 2019 (`NetMigrationRate1019`). Our random forest classifier ranked the importance of all of the above variables in the upper third of all predictors—except for COVID-19 cases which we will discuss in detail below. In the following, we will analyse the observational relationships which we have found between the predictors and change in Trump's relative vote share and compare our findings with our expectations set out in our literature review.

Minority demographics

In post-election analyses, it has been widely reported that Trump increased his stance among immigrant communities. The New York Times reported that in all precincts with 65 per cent or more residents of Hispanic or Asian descent, voters shifted to Trump, though Biden still won all of these counties overall (Cai and Fessenden, 2020). Most Hispanic communities have traditionally been leaning democratic and turned out strongly for Hillary Clinton in 2016, often alienated by Trump's discriminatory rhetoric against Central and South American immigrants and his hard stance on deportation. However, studies examining exit polls have noticed a shift in priorities among Hispanic voters away from immigration policy and towards the economy, health care, and education (Galbraith and Callister, 2020). Particularly through his economic policy and stand on freedom of religion, Trump was able to appeal to these voters, which is reflected in our model that shows a roughly 1-point increase for Trump for every 20 percentage points more Hispanics in a given county (see *Table 4* and *Figure 9*). Overall, two-thirds of Hispanics remained with the Democratic party, but Trump made a notable gain of 10 per cent according to exit polls (The New York Times, 2016, 2020b). Other immigrant communities shifted to Trump as well, though reportedly not as strongly as Hispanics. In our model, this is reflected in the positive correlation of the percentage of people not speaking English at home with increased Trump support.

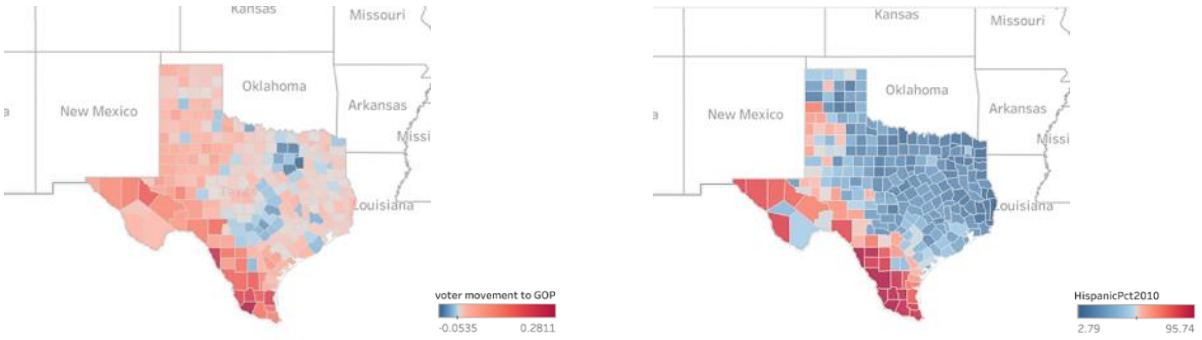


Figure 9: Voter movement in the state of Texas (left) and the percentage of the county population of Hispanic descent (right).

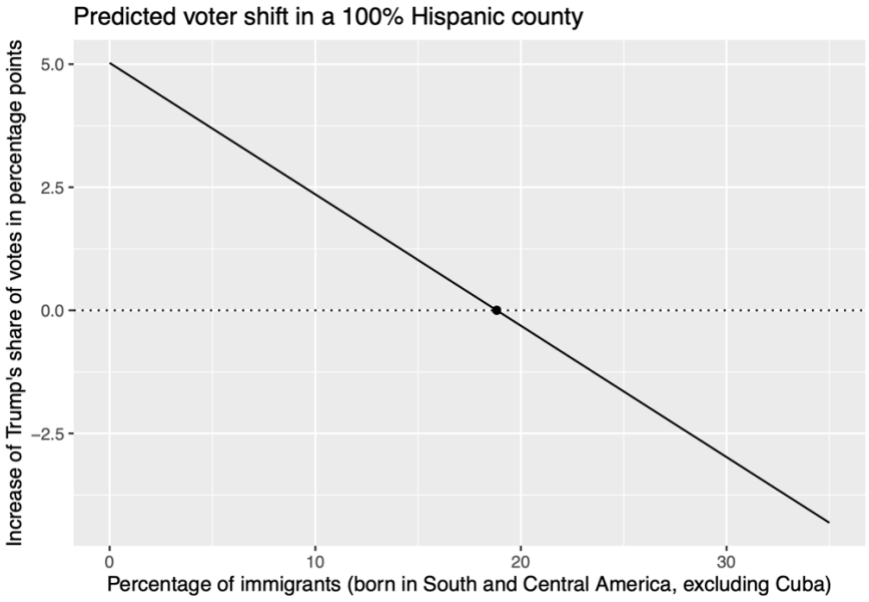


Figure 10: Change in Trump support in a hypothetical 100 per cent Hispanic county

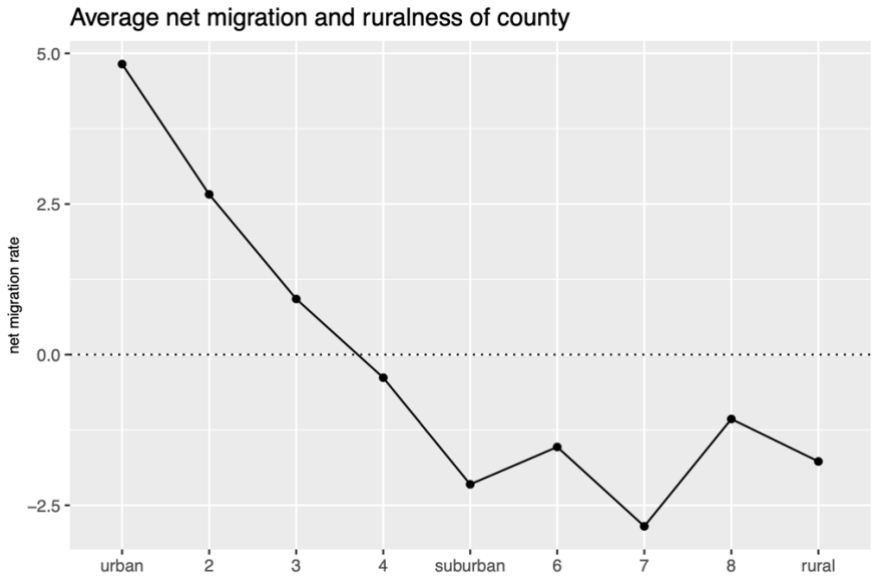


Figure 11: Net Migration in relation to the ruralness of a county

However, our model also points to a more complex relationship that has been little-noticed in other analyses: Hispanic communities voted differently along the lines of first-generation immigrants and American-born descendants. While overall, the percentage of residents with Hispanic descent were positively correlated with Trump support, communities with a higher percentage of residents born in Central or South America were negatively correlated with Trump support. This seems paradoxical but could be explained by the differing priorities of first-generation (i.e., foreign-born) Hispanics and their American-born descendants. For the former, Trump's tough immigration and deportation policies were presumably a major threat, while the latter, who do not have to worry about deportation, have other priorities according to the exit poll discussed above. In conclusion, our model inferred a 1-point decrease in Trump support for every additional 4 percentage points of Central and South American immigrants in a county (see *Table 4*). Taken together with the *increase* in support from Hispanics overall, this creates a dynamic shown in *Figure 10*. Our model predicted an increase in Trump's vote share among Hispanic counties with less than 18 per cent first-generation Central South American (non-Cuban) immigrants and a decrease among those counties with more than 18 per cent Central South American (non-Cuban) immigrants.

Female vote

Polls have reported a decrease in support for Trump among women (Vavreck, 2020, p.28). While we lack the data on individuals' poll interviews to make such a claim, our model finds that Trump lost relative votes where the percentage of female-led households are high. This can be a mere coincidence or due to correlations with the omitted variable of ruralness of a county, but it can also be that a county with more female-headed households has more female than male residents overall. We do not have data on the percentage of women in each county, but it is reportedly as high as 57 per cent in Pulaski County, Georgia, and as low as 28 per cent in Crowley County, Colorado (Thomas, 2012).

Net migration

Economic prosperity has for long been a determining factor for political leanings. The so-called 'Rust Belt' epitomises the rapid decline of manufacturing factories and an accompanying pauperisation of unemployed blue-collar workers. The loss of work opportunities has gone hand in hand with a rural exodus as many people have moved to urban and suburban areas (see *Figure 11* and Johnson and Lichter, 2019). This movement is captured in our net migration rate variable (`NetMigrationRate1019`) that indicates the relative increase of a county's population through immigration and emigration. The negative correlation with Trump support suggests that counties in disadvantaged areas that have seen a lot of people moving away have increasingly supported Trump (see *Table 4*). This is unsurprising given Trump's frequent attempts to direct economic policy at this key constituency in key swing states, going as far as to provoke a trade war with China. While this hasn't helped blue-collar workers in the Rust Belt, they nevertheless identify more strongly with Trump as someone who shares their strongly conservative world views.

Education

`Ed5CollegePlusPct` was ranked as the most important variable in our Random Forest Model and in our linear regression model, the percentage of residents with a college degree was also found to be significantly correlated with a decrease in Trump support (*Table 4*). Trump lost 1 percentage point in 2020 compared to 2016 for every additional 8 percentage points of people with a college degree in a county. This is consistent with exit polls and other political analyses that have noted a significant divide in partisan support according to education level. While in the 20th century, college-educated voters tended to vote Republican, this has shifted in the last four elections which have seen an ever-growing education gap. According to our analysis, this gap increased this year as higher educated counties turned out even stronger against the Republican party than they had in 2016. Many explanations for the education gap have been offered, most notably, political surveys have found a link in economic opportunities and educational attainment that affected party affiliation (Harris, 2020).

COVID-19 cases

Our model finds that an increase of 7,640 COVID-19 cases per 100,000 inhabitants correlates with a 1 percentage point increase in Trump support (*Table 4*). As we have discussed in our literature review, many journalists have scrambled to explain why COVID cases might have led to an increase in Trump support, yet they fail to consider the reverse. Indeed, increased Trump support may well have led to more cases in a given county as Republican voters have followed his message of downplaying the virus and were more likely to disregard social distancing guidelines according to a study published in *Nature* (Gollwitzer et al., 2020). This could be an instance of retrocausality, or reverse causality, where the relationship between variables works oppositely as expected. Overall, the variation in interpretations of this correlation between different publications symbolises a broader issue in statistical analysis, namely that causations cannot be shown through observational linear regression analysis. We merely obtain observational relationships that ought to be underpinned with qualitative investigation in order to be of analytical value.

Conclusion

The findings from our model confirmed many of the trends that other analyses of voter behaviour have noticed, despite their ability to work with precinct-level or even exit poll data that is more detailed than our county-level data. Nevertheless, our analysis provided additional complexities to the well-known shift in Hispanic votes where we noticed the divide between first-generation Hispanic immigrants and their American or Cuban-born descendants. Our results also found a plausible explanation for the impact of COVID-19 on the election, or rather, the effect of the election on COVID-19.

However, our analysis also has its limitations. The larger turnout of African American voters for Biden which was widely reported in the weeks following the election is absent from our analysis. This stems from the fact that our analysis treats counties equally, irrespective of their size. The stark

increase in Democratic votes from the African American community which proved integral to Biden's election victory was limited to a handful of big, urban counties in key swing states. In fact, they can be narrowed down to Milwaukee, Waukesha and Madison in Wisconsin, Atlanta in Georgia, and Phoenix in Arizona which each account for only one of our 3,111 observations (Kolko and Monkovic, 2020). Our model is also unable to capture the increase in turnout at county-level from 2016 to 2020. Therefore, while we were able to highlight important trends in voting patterns, we cannot definitively deduce the effect of our predictors on the election outcome. This is owed, on one hand, to the different sizes of our counties, yet on the other hand, even if they had equal size, the electoral college voting system does not allow for inferences on the entire election as large trends in deeply "red" or "blue" states may not change the final result while small shifts as seen in the Cuban community in Florida 2020 can alter the electoral college votes significantly. Not allowing a direct inference on the election outcome forms a limitation of our model that can only be fixed by performing the same analysis on all 50 states separately, for which we lacked space and time.

Another area of improvement is the use of more detailed data, e.g., on precinct-level, which has yet to be made available for the 2020 election data. Additionally, by using a multiple regression model, we have assumed linear relationships between the independent and dependent variables; an assumption that is unlikely to hold in real-life. Our linear model could also be optimised by introducing interaction variables to explore heterogeneous effects, such as the combined effect of ethnicity and COVID-19 cases. Alternatively, we could have tried shrinkage methods such as Lasso regression to significantly reduce the variance of coefficients and improve the fit of our model.

Yet overall, we have achieved our goal of understanding the large voter movements between 2016 and 2020 by using seven different models. The binary classification models revealed which factors are important for an increase in Trump's vote share regardless of the size of the increase, while multiple linear regression models gave us a richer understanding of the substantive and statistical significance of each individual factor in explaining the increase in Trump's vote share. Our final cv-model is quite successful as it explained 34.5 per cent of the variation in training data and outperformed the benchmark MSE predictions in test data. Acknowledging that our seven-variable model will not account for all the complexities of the US political landscape, we believe that its performance in relation to its simplicity still makes it a valuable tool of analysis for the 2020 US presidential election outcomes.

Bibliography

Albright, C. (2020) Black voters drove Joe Biden's victory – and have offered this country a reboot. *The Guardian*. 10 November. [online]. Available from: https://www.theguardian.com/commentisfree/2020/nov/10/black-voters-drove-joe-biden-victory-reboot-2020 (Accessed 25 January 2021).

Bryant, M. (2020) US voter demographics: election 2020 ended up looking a lot like 2016. *The Guardian*. 5 November. [online]. Available from: https://www.theguardian.com/us-news/2020/nov/05/us-election-demographics-race-gender-age-biden-trump (Accessed 20 November 2020).

Cai, W. & Fessenden, F. (2020) Immigrant Neighborhoods Shifted Red as the Country Chose Blue. *The New York Times*. 20 December. [online]. Available from: https://www.nytimes.com/interactive/2020/12/20/us/politics/election-hispanics-asians-voting.html?pageType=LegacyCollection&collectionName=Polls+and+Voters&label=Polls+and+Voters&module=hub_Band&region=inline&template=storyline_band_recirc.

Crumpton, T. (2020) Black women saved the Democrats. Don't make us do it again. *The Washington Post*. 7 November. [online]. Available from: https://www.washingtonpost.com/outlook/2020/11/07/black-women-joe-biden-vote/ (Accessed 25 January 2021).

Eligon, J. & Burch, A. D. S. (2020) Black Voters Helped Deliver Biden a Presidential Victory. Now What? *The New York Times*. 11 November, p.1.

Fessenden, F. et al. (2020) Even in Defeat, Trump Found New Voters Across the U.S. *The New York Times*. 16 November. [online]. Available from: https://www.nytimes.com/interactive/2020/11/16/us/politics/election-turnout.html?referrer=masthead (Accessed 20 November 2020).

Galbraith, Q. & Callister, A. (2020) Why Would Hispanics Vote for Trump? Explaining the Controversy of the 2016 Election. *Hispanic Journal of Behavioral Sciences*. 42 (1), 77–94.

Gollwitzer, A. et al. (2020) Partisan differences in physical distancing are linked to health outcomes during the COVID-19 pandemic. *Nature Human Behaviour*. 4 (11), 1186–1197.

Harris, A. (2020) America Is Divided by Education. *The Atlantic*. 7 November. [online]. Available from: https://www.theatlantic.com/education/archive/2018/11/education-gap-explains-american-politics/575113/ (Accessed 25 January 2021).

HuffPost (2016) 13 Percent of Alaskans Live in No-Man's Land. *The Huffington Post*. [online]. Available from: https://www.huffpost.com/entry/13-percent-of-alaskans-li_b_9151768?guccounter=1&guce_referrer=aHR0cHM6Ly93d3cuZ29vZ2xlLmNvbS8&guce_referrer_sig=AQAAACA6VUnUgnZualSiE26PsDtfrBVpRXKNKXLZXrMYubHYKFeND8CHkMtOL7efwJkQxYgkkZQkwaIw08o3_qA71XIx7mkaWSfDM1wAvlGxAdsMqVr2nJDcR2EF-1b4Z9UheySpGXKcP-jXe4SlvPjhq301KG4fjZQJUfqw8rb3uV7D (Accessed 25 January 2021).

James, G. et al. (eds.) (2013) *An introduction to statistical learning: with applications in R*. Springer texts in statistics 103. New York: Springer.

Johnson, K. M. & Lichter, D. T. (2019) Rural Depopulation: Growth and Decline Processes over the Past Century: Rural Depopulation. *Rural Sociology*. [Online] 84 (1), 3–27.

Kolko, J. & Monkovic, T. (2020) The Places That Had the Biggest Swings Toward and Against Trump. *The New York Times*. 7 December. [online]. Available from: https://www.nytimes.com/2020/12/07/upshot/trump-election-vote-shift.html (Accessed 31 January 2021).

Masket, S. (2021) How Much Did COVID-19 Affect The 2020 Election? *FiveThirtyEight*. 27 January. [online]. Available from: https://fivethirtyeight.com/features/how-much-did-covid-19-affect-the-2020-election/ (Accessed 31 January 2021).

McGovern, T. (2020) *United States General Election Presidential Results by County from 2008 to 2020* [online]. Available from: https://github.com/tonmcg/US_County_Level_Election_Results_08-20#united-states-general-election-presidential-results-by-county-from-2008-to-2020 (Accessed 1 December 2020).

McMinn, S. & Stein, R. (2020) Many Places Hard Hit By COVID-19 Leaned More Toward Trump In 2020 Than 2016. *NPR*. 6 November. [online]. Available from: https://www.npr.org/sections/health-shots/2020/11/06/930897912/many-places-hard-hit-by-covid-19-leaned-more-toward-trump-in-2020-than-2016?t=1605713108201 (Accessed 20 November 2020).

MIT Election Data And Science Lab (2018a) *County Presidential Election Returns 2000-2016*. [online]. Available from: https://dataverse.harvard.edu/citation?persistentId=doi:10.7910/DVN/VOQCHQ (Accessed 14 February 2021).

MIT Election Data And Science Lab (2018b) *Election Context 2018* [online]. Available from: https://github.com/MEDSL/2018-elections-unoffical/blob/master/election-context-2018.md (Accessed 25 November 2020).

Ponsa-Kraus, C. D. (2020) Make Puerto Rico a State Now. *The New York Times*. 4 November, p.19.

Seipel, A. (2016) FACT CHECK: Trump Falsely Claims A 'Massive Landslide Victory'. *NPR*. 11 December. [online]. Available from: https://www.npr.org/2016/12/11/505182622/fact-check-trump-claims-a-massive-landslide-victory-but-history-differs?t=1612784117344^ (Accessed 8 February 2021).

The New York Times (2020a) *COVID-19 Data* [online]. Available from: https://github.com/nytimes/covid-19-data (Accessed 20 November 2020).

The New York Times (2020b) National Exit Polls: How Different Groups Voted. *The New York Times*. 3 November. [online]. Available from: https://www.nytimes.com/interactive/2020/11/03/us/elections/exit-polls-president.html (Accessed 25 January 2021).

The New York Times (2016) Election 2016: Exit Polls. *The New York Times*. 8 November. [online]. Available from: https://www.nytimes.com/interactive/2016/11/08/us/politics/election-exit-polls.html (Accessed 25 January 2021).

Thomas, G. S. (2012) Women hold population edge in two-thirds of all U.S. counties. *The Business Journals*. 24 September. [online]. Available from: https://www.bizjournals.com/bizjournals/on-numbers/scott-thomas/2012/09/women-hold-population-edge-in.html (Accessed 8 February 2021).

U.S. Department of Agriculture, Economic Research Service (2020) *Atlas of Rural and Small-Town America* [online]. Available from: https://www.ers.usda.gov/data-products/atlas-of-rural-and-small-town-america/download-the-data/ (Accessed 25 November 2020).

USA Facts (2020) *New York Coronavirus Cases and Deaths* [online]. Available from: https://usafacts.org/visualizations/coronavirus-covid-19-spread-map/state/new-york (Accessed 20 December 2020).

Vavreck, L. (2020) It's Not Just Suburban Women. A Lot of Groups Have Turned Against Trump. *The New York Times*. 3 November, p.28.

ST309 Group Project Methodology (Data Cleaning)

Merging election datasets

Datasource for 2020 US presidential election results ("GitHub 2020 Main Dataset.csv") is: https://github.com/tonmcg/US_County_Level_Election_Results_08-20

Datasource for 2016 US presidential election results ("MIT Election Lab 2000-2016 Dataset.csv") is: https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/VOQCHQ

Setting Working Directory and loading packages and dataset

```
setwd("~/OneDrive/Uni/SOAS University of London/Modules/year 3/Elementary Data Analytics/Group Project/
library(readr)
results2020 = read_csv("GitHub 2020 Main Dataset.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   state_name = col_character(),
##   county_fips = col_character(),
##   county_name = col_character(),
##   votes_gop = col_double(),
##   votes_dem = col_double(),
##   total_votes = col_double(),
##   diff = col_double(),
##   per_gop = col_double(),
##   per_dem = col_double(),
##   per_point_diff = col_double()
## )
```

```
results2016 = read_csv("MIT Election Lab 2000-2016 Dataset.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   year = col_double(),
##   state = col_character(),
##   state_po = col_character(),
##   county = col_character(),
##   FIPS = col_double(),
##   office = col_character(),
##   candidate = col_character(),
##   party = col_character(),
##   candidatevotes = col_double(),
##   totalvotes = col_double(),
##   version = col_double()
## )
```

Cleaning 2020 data

```
results2020$county_fips = as.integer(results2020$county_fips)
```

Cleaning 2016 data

```
per_candidate = results2016[,9]/results2016[,10]
results2016 = cbind(results2016,per_candidate)
results2016$party <- as.factor(results2016$party)
results2016 = na.omit(results2016)
results2016 = results2016[results2016[,1]==2016,]
results2016.GOP = results2016[results2016[,8]=="republican",]
results2016.GOP = results2016.GOP[order(results2016.GOP[,2],results2016.GOP[,5]),]
results2016.GOP = results2016.GOP[-1609,] #Kansas City missing in 2020 data
#Oglala Lakota County is wrongly called 46113 instead of 46102 in 2016 data
results2016.GOP[2428,5] = 46102
results2016.GOP = results2016.GOP[order(results2016.GOP[,2],results2016.GOP[,5]),]
results2016.Dem = results2016[results2016[,8]=="democrat",]
results2016.Dem = results2016.Dem[order(results2016.Dem[,2],results2016.Dem[,5]),]
results2016.Dem = results2016.Dem[-1609,]
results2016.Dem[2428,5] = 46102
results2016.Dem = results2016.Dem[order(results2016.Dem[,2],results2016.Dem[,5]),]
per_diff_2016 = results2016.GOP[,12]-results2016.Dem[,12]
```

Create dependent variable

```
voter_movement = results2020$per_gop-results2016.GOP[,12] #change in support for GOP
```

Create one big matrix with all election data

```
resultsall = results2020[,c(1,2,3,8,9,10)]
resultsall = cbind(resultsall,results2016.GOP[,12],results2016.Dem[,12],
                   per_diff_2016,voter_movement)
colnames(resultsall) = c("State","FIPS","County","per_GOP_2020","per_Dem_2020",
                         "per_diff_2020","per_GOP2016","per_Dem_2016",
                         "per_diff_2016","voter_movement_to_GOP")
resultsall = resultsall[-c(68:107),] # removing Alaska
```

Export as csv

```
write.csv(resultsall,"election_data.csv")
```

Merging predictors

We have multiple datasources for our predictors.

To capture the influence of covid-19 at county-level, we used cumulative counts of covid cases and deaths in each county published by The New York Times on Github: https://github.com/nytimes/covid-19-data

Set new working directory and import covid variables:

```r
NYTCases <-read.csv("NYT Nov3rd Covid.csv")
dim(NYTCases)
```

```
## [1] 3243   10
```

```r
#We selected the election date (11.03.2020) as the cut-off point
NYTNovThird <- subset(NYTCases,date=="2020-11-03")
dim(NYTNovThird)
```

```
## [1] 3243   10
```

```r
#We save this as a new dataset
write.csv(NYTNovThird,"NYT Nov3rd Covid.csv")

covid = read_csv("NYT Nov3rd Covid.csv") #We import this data into our working directory
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   X1 = col_double(),
##   X.3 = col_double(),
##   X.2 = col_double(),
##   X.1 = col_double(),
##   X = col_double(),
##   date = col_date(format = ""),
##   county = col_character(),
##   state = col_character(),
##   fips = col_double(),
##   cases = col_double(),
##   deaths = col_double()
## )
```

Next we imported datasets containing other relevant demographic variables.

("Jobs.csv","Income.csv","People.csv") are demographic variables taken from Atlas of Rural and Small Town America provided by the Economic Research Service (ERS) of the US Department of Agriculture. The datasource is: https://www.ers.usda.gov/data-products/atlas-of-rural-and-small-town-america/download-the-data/

("MIT Election Lab 2016 Pres with Demographic Variables.csv") comes from MIT Election Lab 2018 Election Analysis Dataset, which is originally designed as a complementary dataset for analyzing the 2018 US General Election. However, we believe that these variables will also be relevant for our focus on the 2016 and 2020 US Presidential Elections. The datasource is: https://github.com/MEDSL/2018-elections-unoffical/blob/master/election-context-2018.md

Now we import all these new data into our working directory.

```r
setwd("~/OneDrive/Uni/SOAS University of London/Modules/year 3/Elementary Data Analytics/Group Project/
#ERS Rural Atlas Data
covid = NYTNovThird
income_data = read_csv("Income.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   FIPS = col_character(),
##   State = col_character(),
##   County = col_character(),
##   MedHHInc = col_double(),
##   PerCapitaInc = col_double(),
##   PovertyUnder18Pct = col_double(),
##   PovertyAllAgesPct = col_double(),
##   Deep_Pov_All = col_double(),
##   Deep_Pov_Children = col_double(),
##   PovertyUnder18Num = col_double(),
##   PovertyAllAgesNum = col_double()
## )
```

```
job_data = read_csv("Jobs.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double(),
##   FIPS = col_character(),
##   State = col_character(),
##   County = col_character()
## )
## i Use `spec()` for the full column specifications.
```

```
people = read_csv("People.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double(),
##   FIPS = col_character(),
##   State = col_character(),
##   County = col_character()
## )
## i Use `spec()` for the full column specifications.
```

```
#MIT Election Lab Analysis Dataset
rural = read_csv("MIT Election Lab 2016 Pres with Demographic Variables.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double(),
##   state = col_character(),
##   county = col_character()
## )
## i Use `spec()` for the full column specifications.
```

Then we clean and export income data: remove summary data for each state and remove states Alaska (because voting districts are not the counties) and Puerto Rico (because they are not allowed to vote).

```
income_data = income_data[-c(1,2),]
income_data = income_data[-c(68:102),]
income_data = income_data[-83,]
income_data = income_data[-158,]
income_data = income_data[-216,]
income_data = income_data[-280,]
income_data = income_data[-288,]
income_data = income_data[-c(291, 293, 361, 521, 527, 572, 675, 768, 868, 974,
                             1095, 1160, 1177, 1202, 1217, 1301, 1389, 1472,
                             1588, 1645, 1739, 1757, 1768, 1790, 1824, 1887,
                             1988, 2042, 2131, 2209, 2246, 2314, 2320, 2367, 2434,
                             2530, 2785, 2815, 2830, 2965, 3005, 3061, 3134, 3158),]
income_data = income_data[-520,]
income_data = income_data[-2887,]
income_data = income_data[-c(3113:3190),]
write.csv(income_data,"income_final.csv")
```

Clean and export job data

```
job_data = job_data[-c(1, 2, 70, 104, 120, 196, 255, 320, 329, 333, 335, 403, 563,
                       569, 614, 717, 810, 910, 1016, 1137, 1202, 1219, 1244, 1259,
                       1343, 1431, 1514, 1630, 1687, 1781, 1799, 1810, 1832, 1866,
                       1929, 2030, 2084, 2173, 2251, 2288, 2356, 2362, 2409, 2476,
                       2572, 2827, 2857, 2872, 3007, 3047, 3103, 3176, 3200),]
job_data = job_data[!(job_data$State=="AK"),]
job_data = job_data[!(job_data$State=="PR"),]
job_data = job_data[!(job_data$FIPS==15005),]
job_data = job_data[!(job_data$FIPS==51515),]
write.csv(job_data,"Jobs_final.csv")
```

Clean and export covid data: Add seven rows that are missing Sources:https://usafacts.org/visualizations/coronavirus-covid-19-spread-map/state/new-york

```
covid = na.omit(covid)
covid = covid[!(covid$state=="Alaska"),]
covid = covid[order(covid$fips),]
covid = covid[-c(3106:3110),]
missing1 = c(700000, "03/11/2020", "Esmerelda", "Nevada", 32009, 0, 0)
missing2 = c(700000, "03/11/2020", "Bronx", "New York", 36005, 56165, 4997)
missing3 = c(700000, "03/11/2020", "Kings", "New York", 36047, 76663, 7409)
missing4 = c(700000, "03/11/2020", "New York", "New York", 36061, 36980, 3198)
missing5 = c(700000, "03/11/2020", "Queens", "New York", 36081, 78013, 7297)
missing6 = c(700000, "03/11/2020", "Richmond", "New York", 36085, 17777, 1097)
missing7 = c(700000, "03/11/2020", "Loving", "Texas", 48301, 0, 0)
covid = rbind(covid,missing1,missing2,missing3,missing4,missing5,missing6,missing7)
covid$fips = as.integer(covid$fips)
covid = covid[order(covid$fips),]
write.csv(covid,"covid_final.csv")
```

Clean and export people data

```
people = people[-c(1, 2, 70, 100, 116, 192, 251, 316, 325, 329, 331, 399, 559, 565,
                   610, 713, 806, 906, 1012, 1133, 1198, 1215, 1240, 1255, 1339,
                   1427, 1510, 1626, 1683, 1777, 1795, 1806, 1828, 1862, 1925,
                   2026, 2080, 2169, 2247, 2284, 2352, 2358, 2405, 2472, 2568,
                   2823, 2853, 2868, 3002, 3042, 3098, 3171),]
people = people[!(people$State=="AK"),]
people = people[!(people$State=="PR"),]
people = people[!(people$FIPS==15005),]
people = people[!(people$FIPS==51515),]
people = people[order(people$FIPS),]
write.csv(people,"people_final.csv")
```

Clean and export rural-urban code

```
rural = rural[-c(1799,2888),]
rural = rural[,39]
write.csv(rural,"rural_code_final.csv")
```

Merging election data with predictors

Delete FIPS and State columns

```
income_data = income_data[,-c(1,2)]
job_data = job_data[,-c(1,2)]
covid = covid[,-c(4,5)]
people = people[,-c(1,2)]
```

Create new variables for proportional covid cases and deaths

```
alldata = cbind(resultsall,income_data,job_data,covid,people,rural)
alldata$cases = as.numeric(alldata$cases)
```

```
## Warning: NAs introduced by coercion
```

```
cases_per_100000 = alldata$cases*100000/alldata$TotalPopEst2019
alldata$deaths = as.numeric(alldata$deaths)
```

```
## Warning: NAs introduced by coercion
```

```
deaths_per_100000 = alldata$deaths*100000/alldata$TotalPopEst2019
alldata = cbind(alldata,cases_per_100000,deaths_per_100000)
```

Export data

```
write.csv(alldata,"alldata_final.csv")
```

Add variable with employment change over Trump presidency

```
PctEmpChange1619 = (alldata[,21]-alldata[,24])/alldata[,24]
alldata = cbind(alldata,PctEmpChange1619)
```

Deleting not needed predictors

```
usefuldata_1 = alldata[,-c(3, 13, 14, 17:20, 22:23, 24, 25:28, 31:32, 43:90, 93,
                           95, 100:103, 113:124, 139, 141:146, 148:153, 156,
                           158:175, 178:181)]
```

Delete NAs

```
usefuldata = na.omit(usefuldata_1)
nrow(usefuldata_1)-nrow(usefuldata)
```

```
## [1] 8
```

Only one observation is lost by removing NAs.

Split in training and testing data and export

```
set.seed(1)
trainrows = sample(1:nrow(usefuldata),1000)
train = usefuldata[trainrows,]
test = usefuldata[-trainrows,]
write.csv(train,"train.csv")
write.csv(test,"test.csv")
write.csv(usefuldata,"train_and_test.csv")
```

ST309 Group Project Methodology (Data Analysis)

Preparation

```
> rm(list=ls())
> setwd("~/Documents/R/ST309/ST309 Project/R Code and Output")
> #please change this to your working directory with the data
> train<-read.csv("train.csv")
> dim(train)
```

```
[1] 1000   68
```

```
> sum(is.na(train))
```

```
[1] 0
```

```
> test<-read.csv("test.csv")
> sum(is.na(test))
```

```
[1] 0
```

```
> dim(test)
```

```
[1] 2111   68
```

```
> train_and_test <- read.csv("train_and_test.csv")
> sum(is.na(train_and_test))
```

```
[1] 0
```

```
> dim(train_and_test)
```

```
[1] 3111   68
```

```
> library(stargazer)
> library(tidyverse)
> library(broom)
> library(knitr)
> library(ggplot2)
```

There are no more missing values after we have cleaned our dataset. We see that we have 3111 observations in total, with 2111 observations in the test dataset and 1000 observations in the training dataset.

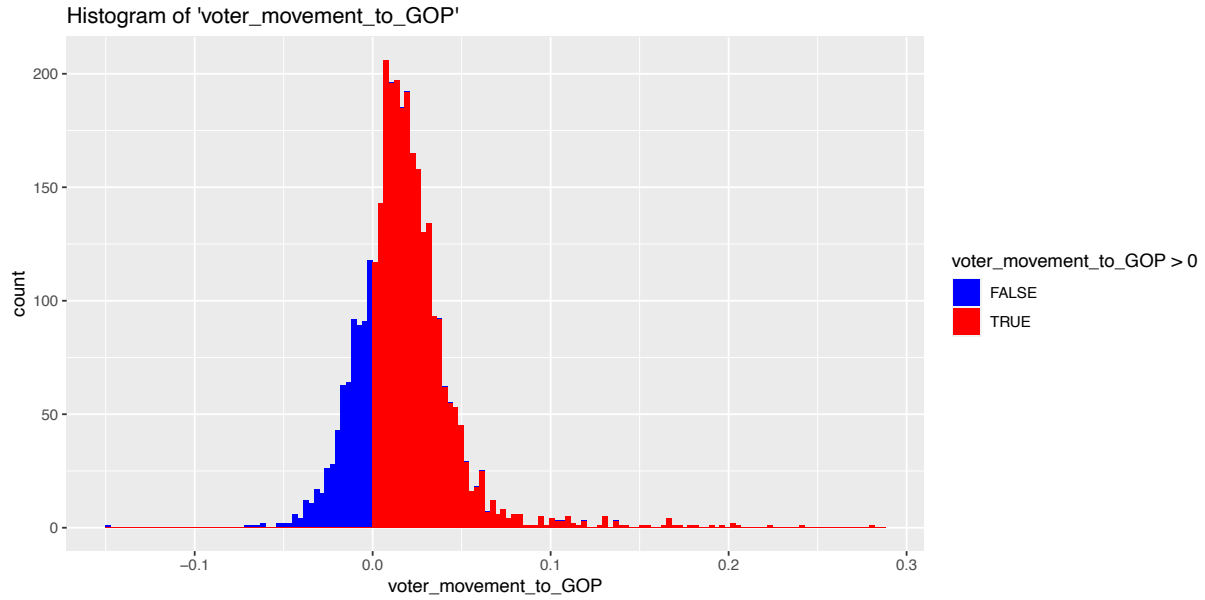
Preliminary Analysis

For the data description section of our report, we focused specifically to understand "voter_movement_to_GOP", our main dependent variable of interest. We first found the five-point summary for this variable and then produced a histogram.

```
> #5 Number Summary Statistics for the variable
> summary(train_and_test$voter_movement_to_GOP)
```

```
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
-0.148478  0.002144  0.015024  0.017100  0.028747  0.281147
```

```
> #Histogram
> ggplot(train_and_test, aes(voter_movement_to_GOP, fill=voter_movement_to_GOP>0))+
+   scale_fill_manual(values = c("blue", "red"))+
+   geom_histogram(binwidth = 0.01, breaks = seq(-0.15, 0.29, by = 0.003))+
+   ggtitle("Histogram of 'voter_movement_to_GOP'")
```



""

Finally, based on maps which we produced in Tableau, we suspected that there might be a positive correlation between covid variables ("cases_per_100000", "deaths_per_100000"and "voter_movement_to_GOP"). Therefore in our preliminary analysis, we produced a simple correlation matrix of these variables as summarized below.

```
> correlation.matrix <- cor(train_and_test[,c("cases_per_100000",
+                                             "deaths_per_100000","voter_movement_to_GOP")])
> stargazer(correlation.matrix, type="text",title="Correlation Matrix")
```

```
Correlation Matrix
================================================================================
                      cases_per_100000 deaths_per_100000 voter_movement_to_GOP
--------------------------------------------------------------------------------
cases_per_100000             1               0.510               0.114
deaths_per_100000          0.510               1                 0.045
voter_movement_to_GOP      0.114             0.045                 1
--------------------------------------------------------------------------------
```

Part One 2-Class Classification

First, we create a binary variable "GOP_increase" in both training and test data.

```
> GOP_increase = test$voter_movement_to_GOP>0
> test = cbind(test,GOP_increase)
> test$GOP_increase = as.factor(test$GOP_increase)
> GOP_increase = train$voter_movement_to_GOP>0
> train = cbind(train,GOP_increase)
> train$GOP_increase = as.factor(train$GOP_increase)
> dim(train)
```

```
[1] 1000   69
```

```
> dim(test)
```

```
[1] 2111   69
```

Next, we remove redundant variables from the training dataset. This includes variables which we have used to construct our dependent variable "voter_movement_to_GOP" and demographic variables we included for reference but based on our subject knowledge are not suitable for explaining increase in share of votes for parties at county level.This leaves us with 50 variables in total ("GOP_increase", "voter_movement_to_GOP" and 48 potential predictors)

```
> train1 <- subset(train, select=-c(State,FIPS,per_GOP_2020,per_Dem_2020,per_diff_2020,per_GOP2016,
+                             per_Dem_2016,per_diff_2016,County,MedHHInc,cases,deaths,TotalPopEst2019,
+                             TotalPopEst2015,TotalPopEst2016,AvgHHSize,TotalHH,X,ForeignBornNum))
> dim(train1)
```

```
[1] 1000   50
```

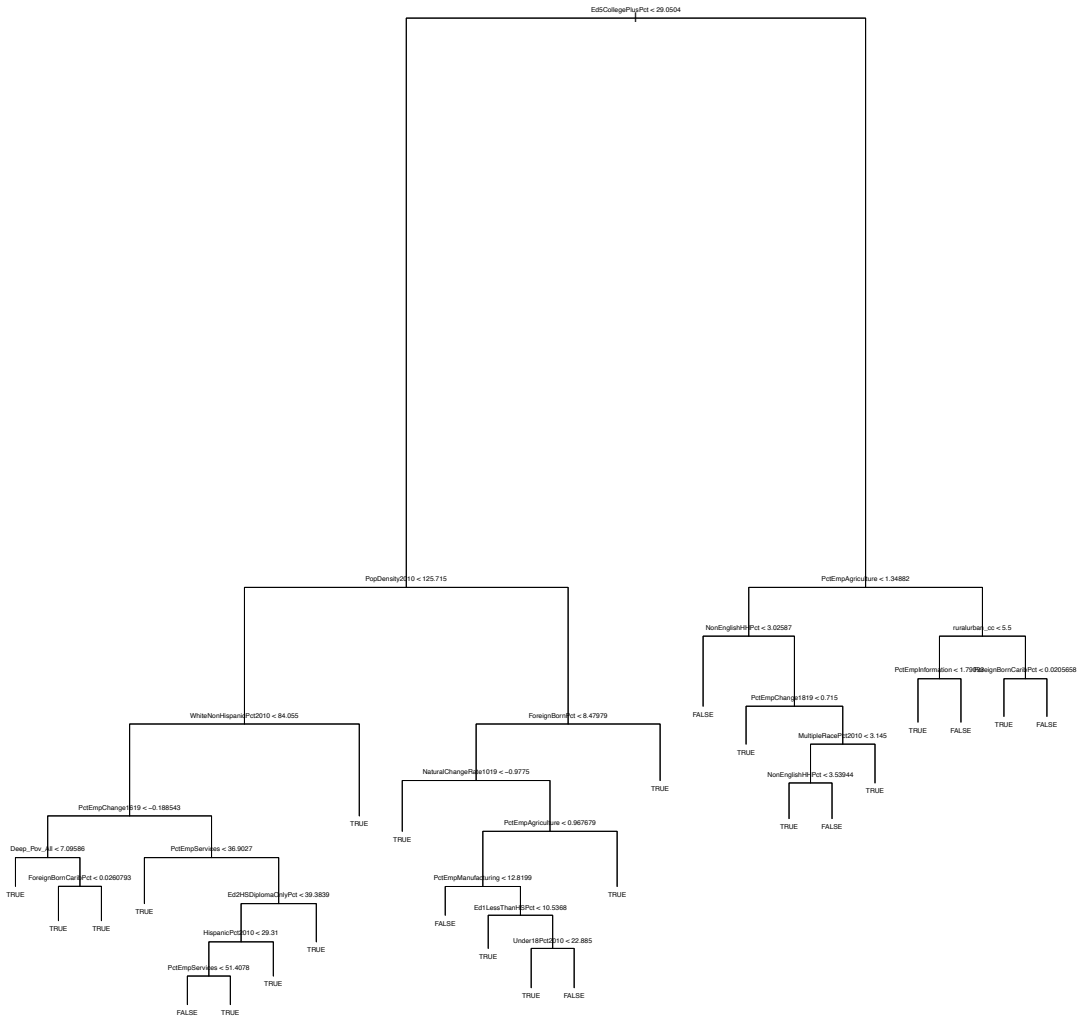
Simple Classification Tree

We start by constructing a simple classification tree.

```
> library(tree)
> simple_tree=tree(GOP_increase~.-voter_movement_to_GOP, data=train1)
> summary(simple_tree)
```

```
Classification tree:
tree(formula = GOP_increase ~ . - voter_movement_to_GOP, data = train1)
Variables actually used in tree construction:
 [1] "Ed5CollegePlusPct"       "PopDensity2010"
 [3] "WhiteNonHispanicPct2010" "PctEmpChange1619"
 [5] "Deep_Pov_All"            "ForeignBornCaribPct"
 [7] "PctEmpServices"          "Ed2HSDiplomaOnlyPct"
 [9] "HispanicPct2010"         "ForeignBornPct"
[11] "NaturalChangeRate1019"   "PctEmpAgriculture"
[13] "PctEmpManufacturing"     "Ed1LessThanHSPct"
[15] "Under18Pct2010"          "NonEnglishHHPct"
[17] "PctEmpChange1819"        "MultipleRacePct2010"
[19] "ruralurban_cc"           "PctEmpInformation"
Number of terminal nodes:  25
Residual mean deviance:  0.4532 = 441.9 / 975
Misclassification error rate: 0.082 = 82 / 1000
```

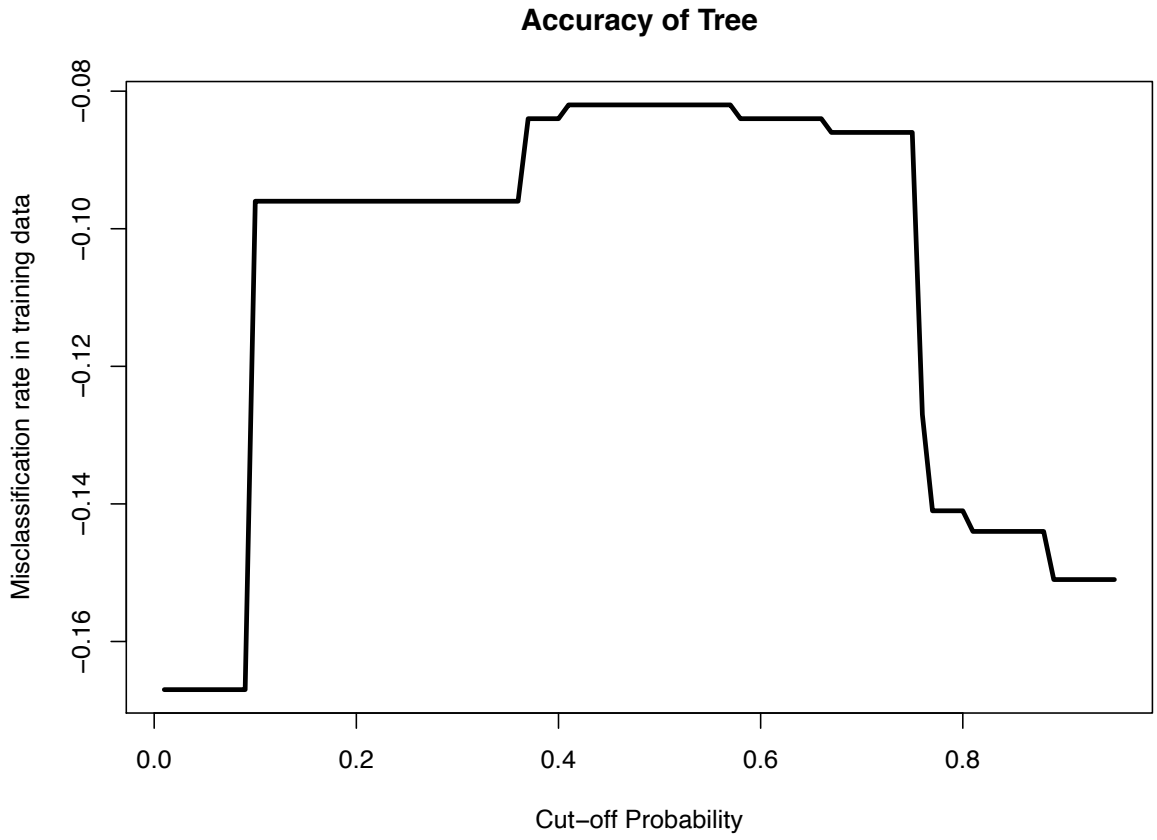
```
> plot(simple_tree)
> text(simple_tree, pretty=0.5, cex=0.5)
```



Now we conduct the cost-benefit analysis on training data to find the apppriate cut-off probability. We assume symmetrical costs for false positives and false negatives.

```
> predTrain.tree = predict(simple_tree, train1, type="vector")[,2]
> #False positives and false negatives are treated equally
> CB = matrix(c(0,-1,-1,0),nrow=2,byrow=T)
> a = seq(0.01,0.95,0.01)
> expected.profit = vector(length = length(a))
> for(i in 1:length(a)) {
+   pred = ifelse((predTrain.tree>=a[i]),"GOP_increase", "GOP_decrease")
+   confusion = table(pred, train1$GOP_increase, deparse.level = 2)
+   expected.profit[i] = sum(CB*confusion)/sum(confusion)
+ }
> plot(a,expected.profit, type="l",lwd=3, xlab="Cut-off Probability",
```

```
+      ylab="Misclassification rate in training data", main="Accuracy of Tree")
```




```
> a[expected.profit==max(expected.profit)]
```

```
 [1] 0.41 0.42 0.43 0.44 0.45 0.46 0.47 0.48 0.49 0.50 0.51 0.52 0.53 0.54 0.55
[16] 0.56 0.57
```

A natural choice is hence $\hat{a} = 0.5$ as our cut-off probability.

```
> a.hat = 0.5
```

This allows us to calculate misclassification rate of our simple tree on test data.

```
> predTest.tree = predict(simple_tree, test, type="vector")[,2]
> pred = ifelse((predTest.tree>=a.hat),"GOP_increase","GOP_decrease")
> confusion = table(pred, test$GOP_increase, deparse.level = 2)
> confusion
```

```
              test$GOP_increase
pred           FALSE TRUE
  GOP_decrease   236  163
  GOP_increase   225 1487
```

```
> (confusion[1,2]+confusion[2,1])/sum(confusion)
```

```
[1] 0.1837991
```

We define a true positive as the case where both the simple tree and the test data tell us there is an increase in Trump's vote share ("GOP_increase" is TRUE). The misclassification error rate for our simple tree is 18%

Bagging

Next, we tried to improve our simple tree with bagging and random forest.

```
> library(randomForest)
> set.seed(1)
> bag.tree = randomForest(GOP_increase~.-voter_movement_to_GOP,data=train1,mtry=48,importance=T)
> #We have 50 variables in total
> #so removing GOP_increase and voter_movement_to_GOP gives us mtry=48
> bag.tree
```

```
Call:
 randomForest(formula = GOP_increase ~ . - voter_movement_to_GOP,      data = train1, mtry = 48, importa
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 48

        OOB estimate of  error rate: 14%
Confusion matrix:
      FALSE TRUE class.error
FALSE   133   97  0.42173913
TRUE     43  727  0.05584416
```

The confusion matrix can be calculated as followed:

```
> pred = predict(bag.tree, newdata=test)
> confusion2 = table(pred,test$GOP_increase, deparse.level = 2)
> confusion2
```

```
       test$GOP_increase
pred    FALSE TRUE
  FALSE   255   99
  TRUE    206 1551
```

```
> (confusion2[1,2]+confusion2[2,1])/sum(confusion2)
```

```
[1] 0.1444813
```

Bagging has a misclassifcation rate of 14%.

Random Forest

```
> set.seed(1)
> rf.tree = randomForest(GOP_increase~.-voter_movement_to_GOP,data=train1,mtry=7,importance=T)
> #Random forest uses m=sqrt(p), which is apprxoimately 7 in this case
> rf.tree
```

```
Call:
 randomForest(formula = GOP_increase ~ . - voter_movement_to_GOP,      data = train1, mtry = 7, importa
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 7

        OOB estimate of  error rate: 14.1%
Confusion matrix:
      FALSE TRUE class.error
FALSE   130  100  0.43478261
TRUE     41  729  0.05324675
```

```
> pred = predict(rf.tree, newdata=test)
> confusion2 = table(pred,test$GOP_increase, deparse.level = 2)
> confusion2
```

```
       test$GOP_increase
pred    FALSE TRUE
  FALSE   255   72
  TRUE    206 1578
```

```
> (confusion2[1,2]+confusion2[2,1])/sum(confusion2)
```

```
[1] 0.1316911
```

We have a misclassification rate of 13% after using Random Forest.

KNN

We also used 3-KNN and 5-KNN Classifiers to account for potential nonlinear relationships between our predictors and Trump's vote share on the county-level.

```
> library(dplyr)
> Xtrain = select(train,-c(State,FIPS,per_GOP_2020,per_Dem_2020,per_diff_2020,per_GOP2016,
+                          per_Dem_2016,per_diff_2016,per_diff_2016,County,MedHHInc,cases,
+                          deaths,TotalPopEst2019,AvgHHSize,TotalHH,
+                          voter_movement_to_GOP,GOP_increase))
> Xtest = select(test,-c(State,FIPS,per_GOP_2020,per_Dem_2020,per_diff_2020,
+                        per_GOP2016,per_Dem_2016,per_diff_2016,per_diff_2016,
+                        County,MedHHInc,cases,deaths,TotalPopEst2019,AvgHHSize,
+                        TotalHH,voter_movement_to_GOP,GOP_increase))
> dim(Xtrain)
```

```
[1] 1000   52
```

```
> dim(Xtest)
```

```
[1] 2111   52
```

```
> D=-cor(t(Xtest), t(Xtrain))+1
> inDex=matrix(nrow=nrow(D), ncol=5)
> for (i in 1:nrow(D)) inDex[i,]=sort.int(D[i,], index.return = T)$ix[1:5]
> predKNN=matrix(nrow=nrow(D), ncol=2)
> Y = train$GOP_increase
> Y = as.logical(Y)
> for(i in 1:nrow(D)) predKNN[i,]=c(mean(Y[inDex[i,1:3]]), mean(Y[inDex[i,]]))
```

```
> #3-NN
> pred = ifelse((predKNN[,1]>=0.5),TRUE,FALSE)
> confusion = table(pred,test$GOP_increase, deparse.level = 2)
> confusion
```

```
       test$GOP_increase
pred    FALSE TRUE
  FALSE   208  153
  TRUE    253 1497
```

```
> (confusion[1,2]+confusion[2,1])/sum(confusion)
```

```
[1] 0.1923259
```

```
> #5-NN
> pred = ifelse((predKNN[,2]>=0.5),TRUE,FALSE)
> confusion = table(pred,test$GOP_increase, deparse.level = 2)
> confusion
```

```
       test$GOP_increase
pred    FALSE TRUE
  FALSE   199  127
  TRUE    262 1523
```

```
> (confusion[1,2]+confusion[2,1])/sum(confusion)
```

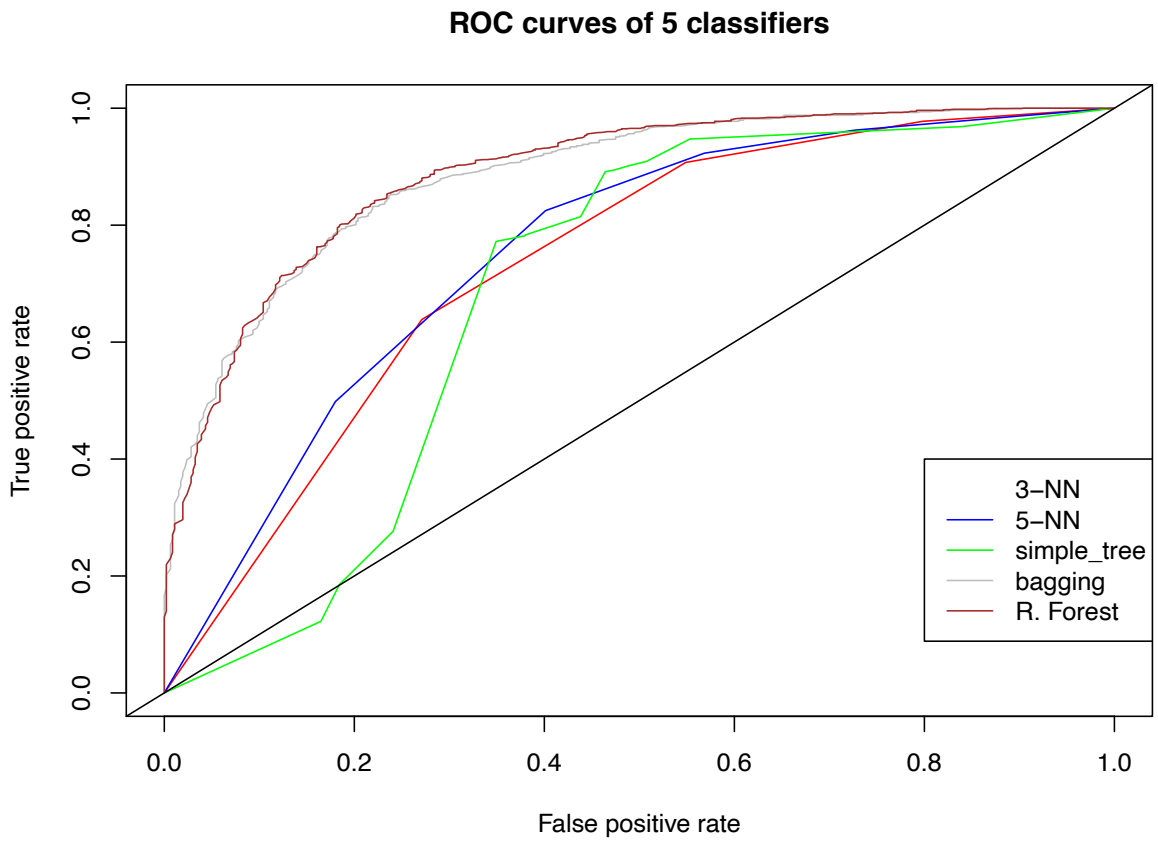
```
[1] 0.1842729
```

The misclassification rates are 19% and 18% for 3-NN and 5-NN, respectively.

Summary of Classification Models and Evaluation

Finally, we use ROC curve to compare all classifiers.

```
> #Simple_Tree
> predTest.simple = predict(simple_tree, newdata=test, type="vector")[,2]
> #Bagging
> predTest.bag = predict(bag.tree, newdata=test, type="prob")[,2]
> #Random Forest
> predTest.rf = predict(rf.tree, newdata=test, type="prob")[,2]
> #KNN is simply predKNN
>
> library(ROCR)
> pred5=prediction(data.frame(predKNN, predTest.simple, predTest.bag, predTest.rf),
+                  data.frame(test$GOP_increase, test$GOP_increase,
+                             test$GOP_increase,test$GOP_increase,test$GOP_increase))
> roc=performance(pred5, measure ="tpr", x.measure ="fpr")
> plot(roc, col=as.list(c("red","blue","green", "grey","brown")),
+      main="ROC curves of 5 classifiers")
> legend(0.8, 0.4, c("3-NN","5-NN","simple_tree", "bagging", "R. Forest"),
+        col=c("red","blue","green", "grey","brown"), lty=c(0,1,1,1,1))
> abline(0,1)
```



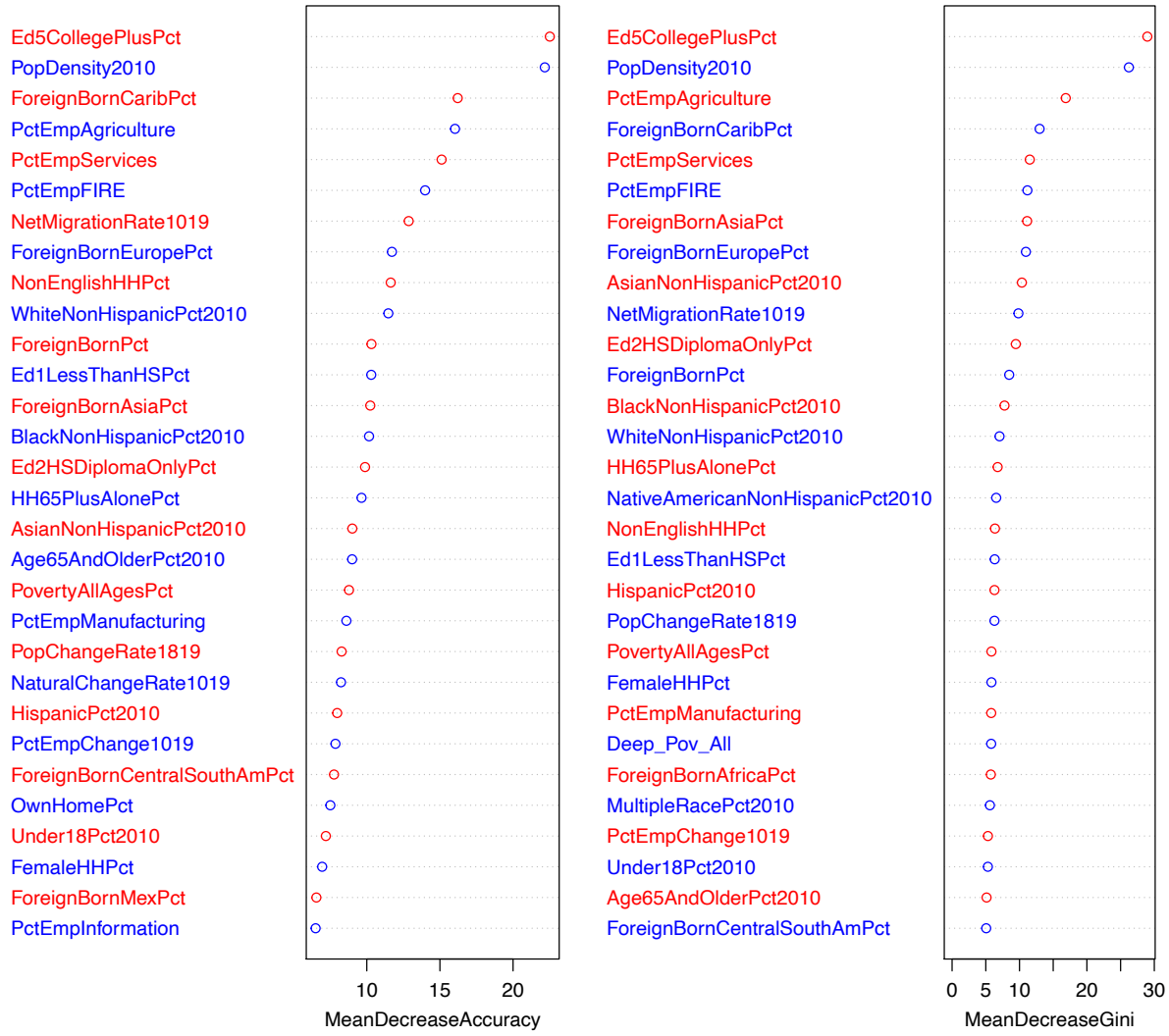
We will also calculate the AUC (Area under the Curve) values for thse classifers

```
> performance(pred5, measure ="auc")@y.values

[[1]]
[1] 0.7358095

[[2]]
[1] 0.7547026

[[3]]
[1] 0.6891553

[[4]]
[1] 0.8848321

[[5]]
[1] 0.8878177
> which.max(performance(pred5, measure ="auc")@y.values)

[1] 5
```

The maximum AUC value is 0.888 and comes from the Random Forest model. Since Random Forest also has the lowest misclassfication rate among all the models we tried, this is our best classification model for predicting whether a county experienced an increase or decrease in vote share for Trump between 2016 and 2020.

From the Random Forest, we see that the following variables are important indicators of whether or not there has been an increase in the share of votes for the Republicans between 2020 and 2016.

```
> varImpPlot(rf.tree, col=c("blue","red"), cex=1)
```


rf.tree
Ed5CollegePlusPct
PopDensity2010
ForeignBornCaribPct
PctEmpAgriculture
PctEmpServices
PctEmpFIRE
NetMigrationRate1019
ForeignBornEuropePct
NonEnglishHHPct
WhiteNonHispanicPct2010
ForeignBornPct
Ed1LessThanHSPct
ForeignBornAsiaPct
BlackNonHispanicPct2010
Ed2HSDiplomaOnlyPct
HH65PlusAlonePct
AsianNonHispanicPct2010
Age65AndOlderPct2010
PovertyAllAgesPct
PctEmpManufacturing
PopChangeRate1819
NaturalChangeRate1019
HispanicPct2010
PctEmpChange1019
ForeignBornCentralSouthAmPct
OwnHomePct
Under18Pct2010
FemaleHHPct
ForeignBornMexPct
PctEmpInformation
10
15
20
MeanDecreaseAccuracy
Ed5CollegePlusPct
PopDensity2010
PctEmpAgriculture
ForeignBornCaribPct
PctEmpServices
PctEmpFIRE
ForeignBornAsiaPct
ForeignBornEuropePct
AsianNonHispanicPct2010
NetMigrationRate1019
Ed2HSDiplomaOnlyPct
ForeignBornPct
BlackNonHispanicPct2010
WhiteNonHispanicPct2010
HH65PlusAlonePct
NativeAmericanNonHispanicPct2010
NonEnglishHHPct
Ed1LessThanHSPct
HispanicPct2010
PopChangeRate1819
PovertyAllAgesPct
FemaleHHPct
PctEmpManufacturing
Deep_Pov_All
ForeignBornAfricaPct
MultipleRacePct2010
PctEmpChange1019
Under18Pct2010
Age65AndOlderPct2010
ForeignBornCentralSouthAmPct
0
5
10
20
30
MeanDecreaseGini

Part Two Linear Regression Models

First, we remove redundant variables from the training dataset. This includes variables which we have used to construct our dependent variable "voter_movement_to_GOP" and demographic variables we included for reference but based on our subject knowledge are not suitable for explaining increase in share of votes for parties at county level.

Because this is a linear model, we will also exclude the binary y-variable outcome "GOP_increase" and one category from Employment and Education variables "PctEmpInformation" and"Ed3SomeCollegePct" in order to avoid the problem of perfect collinearity.This leaves us with 47 potential predictors.

```
> train2 <- subset(train,select=-c(State,FIPS,per_GOP_2020,per_Dem_2020,per_diff_2020,
+                 per_GOP2016,per_Dem_2016,per_diff_2016,County,MedHHInc,cases,deaths,
+                 TotalPopEst2019,TotalPopEst2015,TotalPopEst2016,AvgHHSize,TotalHH,X,
+                 ForeignBornNum,PctEmpInformation,Ed3SomeCollegePct,GOP_increase))
> dim(train2)
```

```
[1] 1000   47
```

We decide to use best subset selection criteria to select our variables for the linear model. Best subset calculation is computationally intensive and setting the nvmax=46 will be infeasible for the computer to handle. At the same time, a model with 46 variable is also unlikely to be a good model given the difficulty in interpretation. After discussion within the team, we eventually agreed on a reasonable value being nvmax=10.

Caution: Even with nvmax=10, the following code takes around 2-3 minutes to run.

```
> library(leaps)
> subset.train=regsubsets(voter_movement_to_GOP~., data=train2, nvmax=10)
> summary(subset.train)
```

```
Subset selection object
Call: regsubsets.formula(voter_movement_to_GOP ~ ., data = train2,
    nvmax = 10)
46 Variables  (and intercept)
                                      Forced in Forced out
PovertyAllAgesPct                         FALSE      FALSE
Deep_Pov_All                              FALSE      FALSE
UnempRate2019                             FALSE      FALSE
PctEmpChange1019                          FALSE      FALSE
PctEmpChange1819                          FALSE      FALSE
PctEmpAgriculture                         FALSE      FALSE
PctEmpMining                              FALSE      FALSE
PctEmpConstruction                        FALSE      FALSE
PctEmpManufacturing                       FALSE      FALSE
PctEmpTrade                               FALSE      FALSE
PctEmpTrans                               FALSE      FALSE
PctEmpFIRE                                FALSE      FALSE
PctEmpServices                            FALSE      FALSE
PctEmpGovt                                FALSE      FALSE
PopChangeRate1819                         FALSE      FALSE
NetMigrationRate1019                      FALSE      FALSE
NaturalChangeRate1019                     FALSE      FALSE
```

```
Net_International_Migration_Rate_2010_2019     FALSE      FALSE
PopDensity2010                                 FALSE      FALSE
Under18Pct2010                                 FALSE      FALSE
Age65AndOlderPct2010                           FALSE      FALSE
WhiteNonHispanicPct2010                        FALSE      FALSE
BlackNonHispanicPct2010                        FALSE      FALSE
AsianNonHispanicPct2010                        FALSE      FALSE
NativeAmericanNonHispanicPct2010               FALSE      FALSE
HispanicPct2010                                FALSE      FALSE
MultipleRacePct2010                            FALSE      FALSE
ForeignBornPct                                 FALSE      FALSE
ForeignBornEuropePct                           FALSE      FALSE
ForeignBornMexPct                              FALSE      FALSE
NonEnglishHHPct                                FALSE      FALSE
Ed1LessThanHSPct                               FALSE      FALSE
Ed2HSDiplomaOnlyPct                            FALSE      FALSE
Ed4AssocDegreePct                              FALSE      FALSE
Ed5CollegePlusPct                              FALSE      FALSE
FemaleHHPct                                    FALSE      FALSE
HH65PlusAlonePct                               FALSE      FALSE
OwnHomePct                                     FALSE      FALSE
ForeignBornAfricaPct                           FALSE      FALSE
ForeignBornAsiaPct                             FALSE      FALSE
ForeignBornCentralSouthAmPct                   FALSE      FALSE
ForeignBornCaribPct                            FALSE      FALSE
ruralurban_cc                                  FALSE      FALSE
cases_per_100000                               FALSE      FALSE
deaths_per_100000                              FALSE      FALSE
PctEmpChange1619                               FALSE      FALSE
1 subsets of each size up to 10
Selection Algorithm: exhaustive
          PovertyAllAgesPct Deep_Pov_All UnempRate2019 PctEmpChange1019
1  ( 1 )  " "               " "          " "           " "
2  ( 1 )  " "               " "          " "           " "
3  ( 1 )  " "               " "          " "           " "
4  ( 1 )  " "               " "          " "           " "
5  ( 1 )  " "               " "          " "           " "
6  ( 1 )  " "               " "          " "           " "
7  ( 1 )  " "               " "          " "           " "
8  ( 1 )  " "               " "          " "           " "
9  ( 1 )  "*"               " "          " "           " "
10  ( 1 ) " "               " "          " "           " "
          PctEmpChange1819 PctEmpAgriculture PctEmpMining PctEmpConstruction
1  ( 1 )  " "              " "               " "          " "
2  ( 1 )  " "              " "               " "          " "
3  ( 1 )  " "              " "               " "          " "
4  ( 1 )  " "              " "               " "          " "
5  ( 1 )  " "              " "               " "          " "
6  ( 1 )  " "              " "               " "          " "
7  ( 1 )  " "              " "               " "          " "
8  ( 1 )  " "              " "               " "          " "
9  ( 1 )  " "              " "               " "          " "
10  ( 1 ) " "              " "               " "          " "
          PctEmpManufacturing PctEmpTrade PctEmpTrans PctEmpFIRE PctEmpServices
```

```
1  ( 1 )  " "                 " "          " "          " "         " "
2  ( 1 )  " "                 " "          " "          " "         " "
3  ( 1 )  " "                 " "          " "          " "         " "
4  ( 1 )  " "                 " "          " "          " "         " "
5  ( 1 )  " "                 " "          " "          " "         " "
6  ( 1 )  " "                 " "          " "          " "         " "
7  ( 1 )  " "                 " "          " "          " "         " "
8  ( 1 )  " "                 " "          " "          "*"         " "
9  ( 1 )  "*"                 " "          " "          " "         " "
10  ( 1 ) "*"                 " "          " "          " "         "*"
          PctEmpGovt PopChangeRate1819 NetMigrationRate1019
1  ( 1 )  " "         " "               " "
2  ( 1 )  " "         " "               " "
3  ( 1 )  " "         " "               " "
4  ( 1 )  " "         " "               " "
5  ( 1 )  " "         " "               " "
6  ( 1 )  " "         " "               "*"
7  ( 1 )  " "         " "               "*"
8  ( 1 )  " "         " "               " "
9  ( 1 )  " "         " "               " "
10  ( 1 ) " "         " "               "*"
          NaturalChangeRate1019 Net_International_Migration_Rate_2010_2019
1  ( 1 )  " "                   " "
2  ( 1 )  " "                   " "
3  ( 1 )  " "                   " "
4  ( 1 )  " "                   " "
5  ( 1 )  " "                   " "
6  ( 1 )  " "                   " "
7  ( 1 )  "*"                   " "
8  ( 1 )  " "                   " "
9  ( 1 )  " "                   " "
10  ( 1 ) "*"                   " "
          PopDensity2010 Under18Pct2010 Age65AndOlderPct2010
1  ( 1 )  " "            " "            " "
2  ( 1 )  " "            " "            " "
3  ( 1 )  " "            " "            " "
4  ( 1 )  " "            " "            " "
5  ( 1 )  " "            " "            " "
6  ( 1 )  " "            " "            " "
7  ( 1 )  " "            " "            " "
8  ( 1 )  " "            "*"            " "
9  ( 1 )  " "            "*"            " "
10  ( 1 ) " "            " "            "*"
          WhiteNonHispanicPct2010 BlackNonHispanicPct2010
1  ( 1 )  " "                     " "
2  ( 1 )  " "                     " "
3  ( 1 )  " "                     " "
4  ( 1 )  " "                     " "
5  ( 1 )  " "                     " "
6  ( 1 )  " "                     " "
7  ( 1 )  " "                     " "
8  ( 1 )  " "                     " "
9  ( 1 )  " "                     " "
10  ( 1 ) " "                     " "
```

```
          AsianNonHispanicPct2010 NativeAmericanNonHispanicPct2010
1  ( 1 )  " "                     " "
2  ( 1 )  " "                     " "
3  ( 1 )  " "                     " "
4  ( 1 )  " "                     " "
5  ( 1 )  " "                     " "
6  ( 1 )  " "                     " "
7  ( 1 )  " "                     " "
8  ( 1 )  " "                     " "
9  ( 1 )  " "                     " "
10  ( 1 ) " "                     " "
          HispanicPct2010 MultipleRacePct2010 ForeignBornPct
1  ( 1 )  " "             " "                 " "
2  ( 1 )  " "             " "                 " "
3  ( 1 )  " "             " "                 " "
4  ( 1 )  " "             " "                 " "
5  ( 1 )  "*"             " "                 " "
6  ( 1 )  "*"             " "                 " "
7  ( 1 )  "*"             " "                 " "
8  ( 1 )  "*"             " "                 " "
9  ( 1 )  "*"             " "                 " "
10  ( 1 ) "*"             " "                 " "
          ForeignBornEuropePct ForeignBornMexPct NonEnglishHHPct
1  ( 1 )  " "                  " "               " "
2  ( 1 )  " "                  " "               "*"
3  ( 1 )  " "                  " "               "*"
4  ( 1 )  " "                  " "               "*"
5  ( 1 )  " "                  " "               "*"
6  ( 1 )  " "                  " "               "*"
7  ( 1 )  " "                  " "               "*"
8  ( 1 )  " "                  " "               "*"
9  ( 1 )  " "                  " "               "*"
10  ( 1 ) " "                  " "               "*"
          Ed1LessThanHSPct Ed2HSDiplomaOnlyPct Ed4AssocDegreePct
1  ( 1 )  " "              " "                 " "
2  ( 1 )  " "              " "                 " "
3  ( 1 )  " "              " "                 " "
4  ( 1 )  " "              " "                 " "
5  ( 1 )  " "              " "                 " "
6  ( 1 )  " "              " "                 " "
7  ( 1 )  " "              " "                 " "
8  ( 1 )  " "              " "                 " "
9  ( 1 )  "*"              " "                 " "
10  ( 1 ) " "              " "                 " "
          Ed5CollegePlusPct FemaleHHPct HH65PlusAlonePct OwnHomePct
1  ( 1 )  "*"               " "         " "              " "
2  ( 1 )  "*"               " "         " "              " "
3  ( 1 )  "*"               "*"         " "              " "
4  ( 1 )  "*"               "*"         " "              " "
5  ( 1 )  "*"               "*"         " "              " "
6  ( 1 )  "*"               "*"         " "              " "
7  ( 1 )  "*"               "*"         " "              " "
8  ( 1 )  "*"               "*"         " "              " "
9  ( 1 )  "*"               "*"         " "              " "
```

```
10  ( 1 ) "*"                "*"          " "                " "
          ForeignBornAfricaPct ForeignBornAsiaPct ForeignBornCentralSouthAmPct
1   ( 1 )  " "                  " "                " "
2   ( 1 )  " "                  " "                " "
3   ( 1 )  " "                  " "                " "
4   ( 1 )  " "                  " "                "*"
5   ( 1 )  " "                  " "                "*"
6   ( 1 )  " "                  " "                "*"
7   ( 1 )  " "                  " "                "*"
8   ( 1 )  " "                  " "                "*"
9   ( 1 )  " "                  " "                "*"
10  ( 1 ) " "                   " "                "*"
          ForeignBornCaribPct ruralurban_cc cases_per_100000 deaths_per_100000
1   ( 1 )  " "                  " "           " "                " "
2   ( 1 )  " "                  " "           " "                " "
3   ( 1 )  " "                  " "           " "                " "
4   ( 1 )  " "                  " "           " "                " "
5   ( 1 )  " "                  " "           " "                " "
6   ( 1 )  " "                  " "           " "                " "
7   ( 1 )  " "                  " "           " "                " "
8   ( 1 )  " "                  " "           "*"                " "
9   ( 1 )  " "                  " "           " "                " "
10  ( 1 ) " "                   " "           " "                " "
          PctEmpChange1619
1   ( 1 )  " "
2   ( 1 )  " "
3   ( 1 )  " "
4   ( 1 )  " "
5   ( 1 )  " "
6   ( 1 )  " "
7   ( 1 )  " "
8   ( 1 )  " "
9   ( 1 )  " "
10  ( 1 ) " "
```

Mathematical Adjustments (BIC, Cp and Adjusted R-squared)

Although best subset selection gives us the best model in terms of RSS (or training error), this is not a good estimate of the test error. So now we use two alternative approaches to estimate the test error. Our first approach is to use mathematical adjustments (BIC, Cp and Adjusted R-squared) to adjust the RSS and estimate test error indirectly.

```
> which.min(summary(subset.train)$bic)
```

```
[1] 10
```

```
> #We prefer BIC than AIC because the BIC will give us a simpler model
> #BIC criterion suggests 10 variables
> which.min(summary(subset.train)$cp)
```

```
[1] 10
```

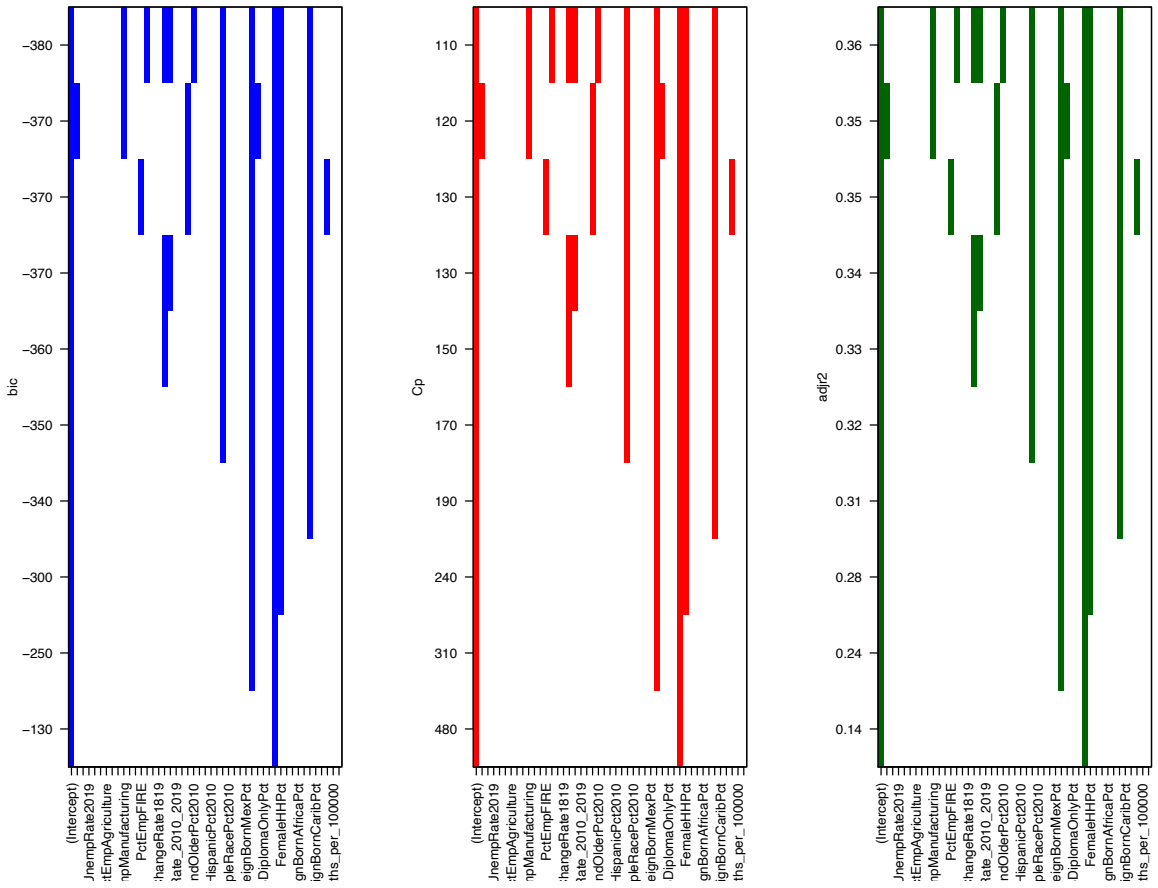
```
> #Cp criterion also recommends the model with 10 variables
> which.max(summary(subset.train)$adjr2)
```

```
[1] 10
```

```
> #Adjusted-R-squared criterion also recommends the model with 10 variables
```

We can also plot the ranking of all models according to BIC, Cp, Adjr2 criteria. We see that all three criteria yield similar result.

```
> par(mfrow=c(1,3))
> #BIC plot
> plot(subset.train,scale="bic", col="blue")
> #Cp plot
> plot(subset.train,scale="Cp", col="red")
> #Adjr2 plot
> plot(subset.train,scale="adjr2", col="dark green")
```



So the 10 coefficients in the best linear model according bic, cp and adjusted R-squared is

```
> coef(subset.train,10)
```

```
                 (Intercept)          PctEmpManufacturing
               0.0128508637                0.0004316640
              PctEmpServices         NetMigrationRate1019
               0.0004737816               -0.0003866061
       NaturalChangeRate1019         Age65AndOlderPct2010
               0.0022010788                0.0013539243
             HispanicPct2010              NonEnglishHHPct
               0.0004896623                0.0035212589
           Ed5CollegePlusPct                FemaleHHPct
              -0.0014442842               -0.0016332384
ForeignBornCentralSouthAmPct
              -0.0025902534
```

We will store this model for comparison later.

```
> lm_math=lm(voter_movement_to_GOP~PctEmpManufacturing+PctEmpServices+NetMigrationRate1019
+            +NaturalChangeRate1019+Age65AndOlderPct2010+HispanicPct2010+NonEnglishHHPct
+            +Ed5CollegePlusPct+FemaleHHPct+ForeignBornCentralSouthAmPct, data=train2)
> tidy(lm_math)
```

```
# A tibble: 11 x 5
   term                          estimate std.error statistic  p.value
   <chr>                            <dbl>     <dbl>     <dbl>    <dbl>
 1 (Intercept)                    0.0129    0.0101        1.27 2.03e- 1
 2 PctEmpManufacturing            0.000432  0.000118      3.67 2.60e- 4
 3 PctEmpServices                 0.000474  0.000149      3.18 1.51e- 3
 4 NetMigrationRate1019          -0.000387  0.000100     -3.86 1.22e- 4
 5 NaturalChangeRate1019          0.00220   0.000383      5.75 1.19e- 8
 6 Age65AndOlderPct2010           0.00135   0.000333      4.07 5.08e- 5
 7 HispanicPct2010                0.000490  0.000101      4.86 1.38e- 6
 8 NonEnglishHHPct                0.00352   0.000469      7.50 1.40e-13
 9 Ed5CollegePlusPct             -0.00144   0.000119    -12.2  7.19e-32
10 FemaleHHPct                   -0.00163   0.000216     -7.57 8.78e-14
11 ForeignBornCentralSouthAmPct -0.00259   0.000344     -7.53 1.17e-13
```

Cross Validation

The second approach we used is to perform a 10th-fold cross-validation to directly estimate the test error. 10-th fold cross-validation means instead of doing a binary split (validation and training data) once, we divide the data into 10 folds, take out one fold as the validation dataset each time and use the rest as training data. With each of the 10 splits, we calculate the RSS and the sum of all RSS divided by 10 is the cross-validation error.

First, we create the empty matrix for storing cv errors.

```
> folds=rep(1:10, length=nrow(train2))
> table(folds)
```

```
folds
  1   2   3   4   5   6   7   8   9  10
100 100 100 100 100 100 100 100 100 100
```

```
> length(folds)
```

```
[1] 1000
```

```
> set.seed(1) #essential to maintain consistency
> folds=sample(folds, replace=F);folds #add randomness to this group number
```

```
   [1]  6  9  9 10  9  1  9 10  8  7  7  7  7  4 10  4 10  5  1  1  5  7  5  9
  [25]  8  5  7  2  7  2  1 10  1  2  1  6  4  4  2  1  4  4  2  6  6  9  3  2
  [49]  1 10  4  7  5  8  8  8  9  5 10 10 10  2  6  2  5  2  2  8  2  3  1  9
  [73]  4  5  7  2  8  6  5  3  5  5  8  7  6  5  1 10  4  1  9  1  2  3 10  4
  [97]  5  7  1  7  4  5  9  7  5  1 10  8  8  6  9  6 10  6  3  6  1  7 10 10
 [121]  8  1  2  3  2  5  2  8  1  6  1  5  6  3  4  8  7  9  6  5  3  1 10 10
 [145]  9  6  8  3  3  3  9  1  6  3  7  2  5  9  2  9  1  5  2 10  8  7  8  7
 [169]  1  9  5  1  8  7  8  4  5  7  6 10  1  5  2  9  2  4  7  8 10  4  9  3
 [193]  8  3  9  6  3  9  7  6  1  7  4  4  4  8  9 10  1  9  9  3  1 10  6  8
 [217]  8  4  8  1  1  7  2  9  4  7  8  4  7  5  1  8  5  3  5 10  9  3  1  7
 [241]  4 10 10  2  6  9  2  7  4  8  2  8  5  3  2  5  4  4  2  3  9  3  5  2
 [265]  4  2  1  4  5  1  4 10  7  6  5  1  9  8  5  8  2  2  4  4 10  8  3  6
 [289] 10  4  6  1  3  3  7  2  8  7  8  1  3  9  8  9  4  4  8  4  1  4  7  5
 [313]  3  4  6  3  2  3  1  8  9  7 10  2  4  5  3  2  3  3  3  7  4  5  2 10
 [337]  4  3  4  7  2  5  1  3  3  7  9  2  7  6  5  1  2  8  5  7 10  8  3  9
 [361]  1  4  6 10 10 10  8  8  1 10  3  8  8  7  7  9  8  8  1  1 10  2  9  8
 [385]  3  2 10  6  9  1  4  4  9  5  9 10 10  7  1  8  9  3  1 10  8  1  9  5
 [409]  1  4  3 10  2  6  4  4  5 10  2 10  2  1  5  4  5 10  3  4  9  7  9  3
 [433]  8  6  9  2  2  4  6  7  7  4  7  8  3  9 10 10  5  3 10  1  6  9  3  9
 [457]  6  5  9  9 10  3  4  6  2  7  2  6  8  5  4  6  5  2  1  6  2 10  2  2
 [481]  9  6  8  6  9  3  5  2 10  5  3  2  1  8  7  5  3  6  8  9  2  7  3  8
 [505]  6  3  7 10  8  2  4  1  8  8  5  7  9  2 10  6  5  3  2  4  1  7  4  3
 [529] 10  3  4  8  2 10 10  1  8  3  1  2  9  6  5  4  6  4  1  6  8  5  6  1
 [553]  5  5 10  9  8  9  6  8  4  7  3  7  6  5  5  3 10  6  9  5  2  1  1  8
 [577]  5  1  2  7  5  2  2  6  1  1  5  6 10  4  7  1 10  1  3  7 10  8  8 10
 [601]  9  1  6  4  6  6  8  4  3  3  3  6  6  4  6  1  2  2  7  2  7  2  2  6
 [625]  5  5  8  7  3  9  5  6  7  9  5  3  3  5  7  2  9  7  9 10 10  3  9 10
 [649]  5  2  6  7  8  1  9  4  7  5  6  1  2  6  7  2  9  3  9  3  6  7 10  7
 [673]  6  1  2 10  6 10  1  8  3  4  7 10  9  9  3  8 10  8  4  2  4  8  3  1
 [697] 10  7  2  5  3  5 10  9  6 10  7 10  7  2  9 10  1  9  3  4  5 10  7  6
 [721]  9  7  2  3  6  9  6 10  3  1  3  4  1  8 10  9  6  1  5  4  8  9  3  2
```

```
 [745]  9  6  2  3  5  1  4  6  8  6  7  4  9  4  4  5  2  5  1  1  8  6  4  1
 [769]  9  9  3  6  9  1  5  3  6  3  4 10  9  6  7  1  2 10  4  2  5  6  1  4
 [793]  3  2  8  1  7  4  4  6  3  3  9 10  6  3  2  2  5  6  5  8  4 10  7  2
 [817]  9  9  6  7  1  5  6  4  1  3  8  7  2  9  8  3  3  1  3  8  4  6  7  5
 [841]  8  1  4  4  6  1 10  7  3  6  3  8  6  1  7  1  1  5  5  7  2 10  3  1
 [865]  8  7  4  7  8  8  5  8  3  4  7  2  4  1  6  7  7  2  8  9  2  6  9  9
 [889]  7  6  8  7  8  5  8  2  3  9  4  2  4  5  9  7  5  5  5 10  9  5  7  4
 [913]  2  1  7  6  4  1 10  4  9 10  6  9  4  8 10  4 10  6  7  3  8  3  4  6
 [937]  1 10  3  1  9  6 10  8  2  7  8  4  5  5  9  8  2  8  2 10  7 10  8  5
 [961]  3  7  3 10  4  9 10  5 10  2  1  9  8  5 10  5 10  9 10  4  7  5  6 10
 [985]  8  6  6  7  9  7  9  6  4  6 10  3  2  3  5  5
```

Next, we use the CV function "predict.regsubsets.r" from Moodle.

```
> predict.regsubsets=function(object ,newdata ,id){
+ form=as.formula(object$call[[2]])
+ mat=model.matrix(form ,newdata )
+ coefi=coef(object, id=id)
+ xvars=names (coefi)
+ mat[,xvars]%*% coefi
+ }
```

Then we apply this function to conduct cross-validation exercise. Caution: because there is a loop within a loop, the following code takes around 10 minutes to run.

```
> cv.errors=matrix(nrow=10, ncol=10)
> for(j in 1:10) {
+     best.fit=regsubsets(voter_movement_to_GOP~., data=train2[folds!=j,], nvmax=10)
+     for(i in 1:10) {
+     pred=predict(best.fit, train2[folds==j,], id=i)
+     cv.errors[j,i]=mean((train2$voter_movement_to_GOP[folds==j]-pred)^2)
+     }
+ }
```

Now we find which model has the minimum cv error.

```
> cvErrors=apply(cv.errors, 2, mean)
> cvErrors
```

```
 [1] 0.0006975907 0.0006417457 0.0006391212 0.0005972266 0.0005884539
 [6] 0.0005889724 0.0005840213 0.0006021377 0.0006102013 0.0006074081
```

```
> which.min(cvErrors)
```

```
[1] 7
```

Since the cv criterion says that the model with the minimum cv error is the best, we should select the model with 7 variables.

```
> coef(best.fit, 7)
```

```
                 (Intercept)         NetMigrationRate1019
                5.270476e-02                -4.142601e-04
             HispanicPct2010              NonEnglishHHPct
                5.696721e-04                 3.886283e-03
           Ed5CollegePlusPct                  FemaleHHPct
               -1.140466e-03                -1.907336e-03
ForeignBornCentralSouthAmPct             cases_per_100000
               -2.691266e-03                 1.635787e-06
```

This allows us to build a linear model with these 7 cv variables.

```
> lm_cv <- lm (voter_movement_to_GOP~NetMigrationRate1019+HispanicPct2010+NonEnglishHHPct+
+               Ed5CollegePlusPct+FemaleHHPct+ForeignBornCentralSouthAmPct+cases_per_100000,
+              data=train2)
> tidy(lm_cv)
```

```
# A tibble: 8 x 5
  term                           estimate   std.error statistic  p.value
  <chr>                             <dbl>       <dbl>     <dbl>    <dbl>
1 (Intercept)                    0.0533     0.00316         16.9  3.17e-56
2 NetMigrationRate1019          -0.000383   0.000102        -3.77 1.75e- 4
3 HispanicPct2010                0.000502   0.0000995        5.05 5.24e- 7
4 NonEnglishHHPct                0.00399    0.000466         8.57 4.01e-17
5 Ed5CollegePlusPct             -0.00119    0.0000843      -14.1  1.70e-41
6 FemaleHHPct                   -0.00175    0.000184        -9.50 1.49e-20
7 ForeignBornCentralSouthAmPct -0.00267    0.000338        -7.89 7.72e-15
8 cases_per_100000               0.00000131 0.000000394      3.32 9.20e- 4
```

Summary of Linear Models and Evaluation

We can summarize the two linear models in a table below.

```
> stargazer(lm_math,lm_cv,type="text",algin=T,
+           title="Comparing Linear Models", ci=T,ci.level=0.95, single.row=T)
```

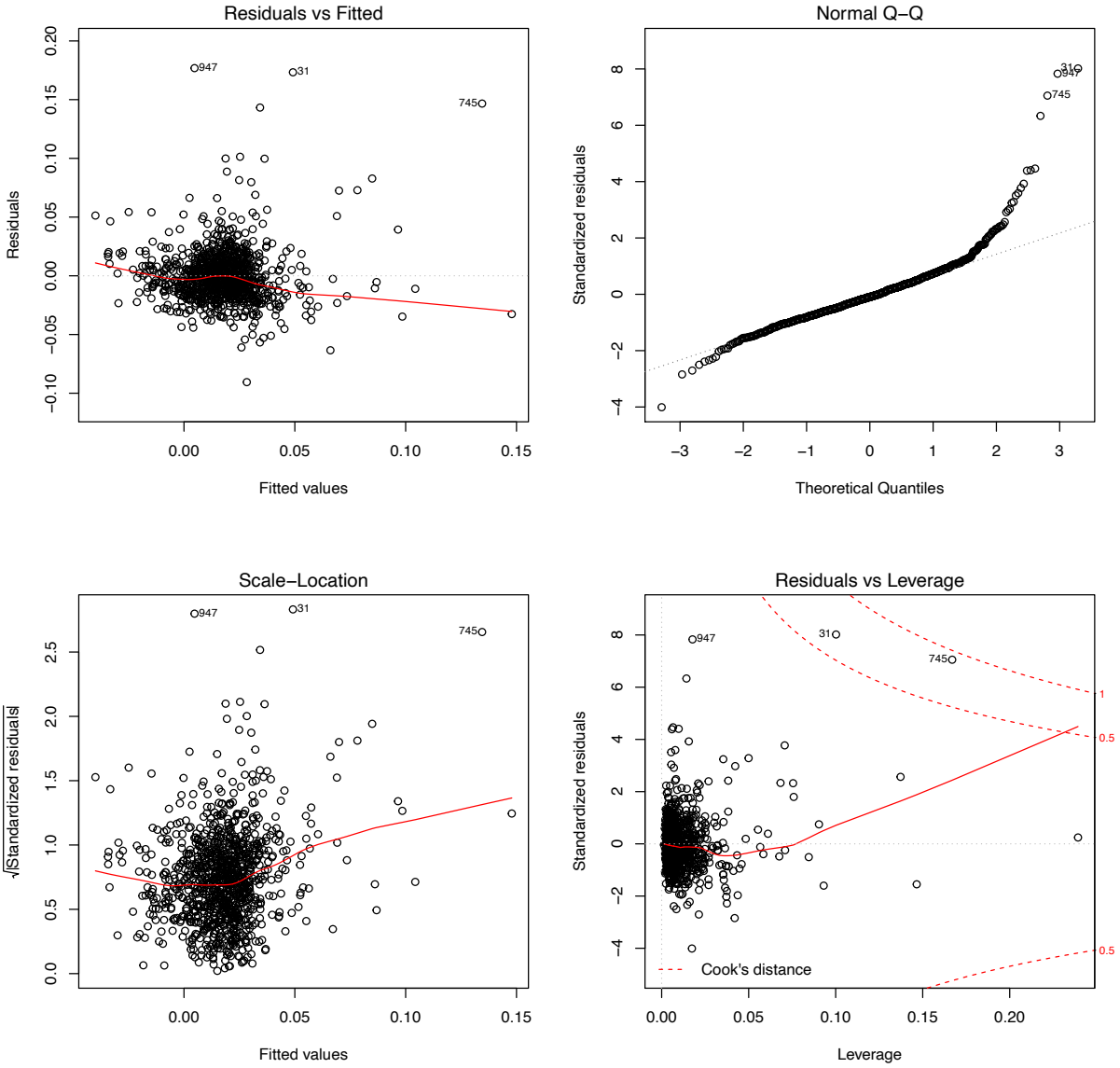
```
Comparing Linear Models
================================================================================================
                                                Dependent variable:
                                    ------------------------------------------------------------
                                                voter_movement_to_GOP
                                              (1)                              (2)
------------------------------------------------------------------------------------------------
PctEmpManufacturing                0.0004*** (0.0002, 0.001)
PctEmpServices                     0.0005*** (0.0002, 0.001)
NetMigrationRate1019              -0.0004*** (-0.001, -0.0002) -0.0004*** (-0.001, -0.0002)
NaturalChangeRate1019               0.002*** (0.001, 0.003)
Age65AndOlderPct2010                0.001*** (0.001, 0.002)
HispanicPct2010                   0.0005*** (0.0003, 0.001)     0.001*** (0.0003, 0.001)
NonEnglishHHPct                     0.004*** (0.003, 0.004)       0.004*** (0.003, 0.005)
Ed5CollegePlusPct                 -0.001*** (-0.002, -0.001)   -0.001*** (-0.001, -0.001)
FemaleHHPct                       -0.002*** (-0.002, -0.001)   -0.002*** (-0.002, -0.001)
ForeignBornCentralSouthAmPct      -0.003*** (-0.003, -0.002)   -0.003*** (-0.003, -0.002)
cases_per_100000                                               0.00000*** (0.00000, 0.00000)
Constant                            0.013 (-0.007, 0.033)        0.053*** (0.047, 0.060)
------------------------------------------------------------------------------------------------
Observations                                1,000                            1,000
R2                                          0.364                            0.345
Adjusted R2                                 0.357                            0.340
Residual Std. Error                  0.023 (df = 989)                 0.023 (df = 992)
F Statistic                        56.532*** (df = 10; 989)         74.518*** (df = 7; 992)
================================================================================================
Note:                                                              *p<0.1; **p<0.05; ***p<0.01

Comparing Linear Models
====
TRUE
----
```

From the table above, we observe that all coefficients are statistically significant and both models can explain around 35-36% variation in the dependent variable.

Next, we examined the residual plots of these models

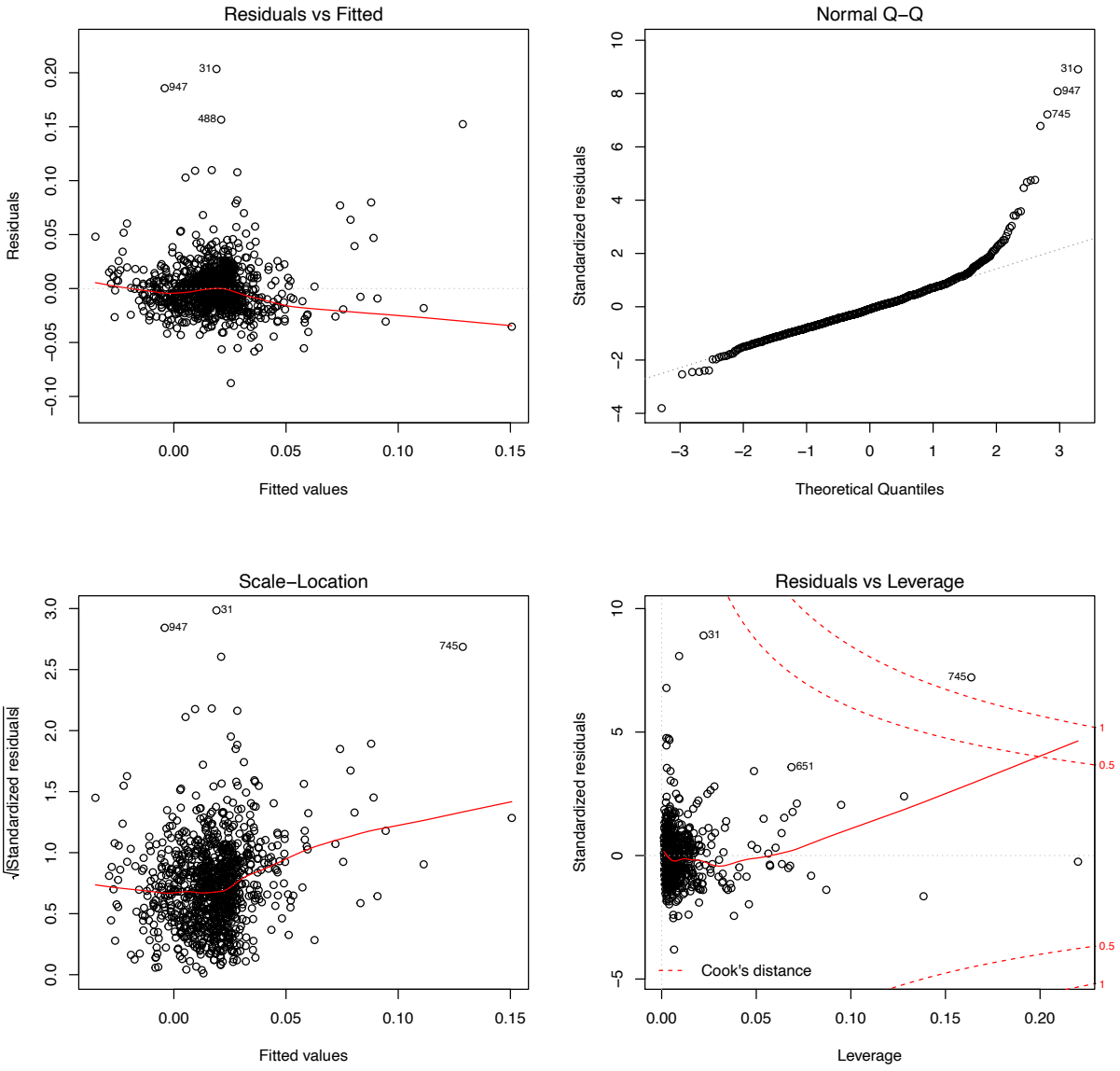
Residual Plot for BIC,Cp and Ajr2 Model

```
> par(mfrow=c(2,2))
> plot(lm_math)
```



Residual Plot for CV Model

```
> par(mfrow=c(2,2))
> plot(lm_cv)
```



For both models, we observe there are some bad leverage points which indicates the potential problem of underfitting in our model.

Next, we checked for problems of multicollinearity. We find no problems of multicollinearity because the VIF values are relatively small.

```
> library(olsrr)
> ols_vif_tol(lm_math)
```

```
                     Variables Tolerance      VIF
1           PctEmpManufacturing 0.7029515 1.422573
2                PctEmpServices 0.4353441 2.297033
3           NetMigrationRate1019 0.8441133 1.184675
4          NaturalChangeRate1019 0.2769387 3.610907
5           Age65AndOlderPct2010 0.2813838 3.553865
6                HispanicPct2010 0.2723093 3.672295
7                NonEnglishHHPct 0.2293396 4.360347
8              Ed5CollegePlusPct 0.3984312 2.509844
9                    FemaleHHPct 0.5832651 1.714486
10 ForeignBornCentralSouthAmPct 0.2940867 3.400358
```

```
> ols_vif_tol(lm_cv)
```

```
                     Variables Tolerance      VIF
1          NetMigrationRate1019 0.8424018 1.187082
2               HispanicPct2010 0.2870839 3.483303
3               NonEnglishHHPct 0.2387726 4.188086
4             Ed5CollegePlusPct 0.8096786 1.235058
5                   FemaleHHPct 0.8236633 1.214088
6 ForeignBornCentralSouthAmPct 0.3129686 3.195209
7              cases_per_100000 0.8561854 1.167971
```

Finally, we evaluated the performance of the two linear models on test data by calculating their actual test Mean Squared Error (MSE).

```
> y.test=test$voter_movement_to_GOP
>
> #Model selected based on Mathematical Adjustment (BIC, Cp and Adjr2)
> math.pred=predict(lm_math, test)
> MSE_math = mean((math.pred-y.test)^2)
>
> #CV Model
> cv.pred= predict(lm_cv, test)
> MSE_cv = mean((cv.pred-y.test)^2)
>
> #We can compare the performance of our models to the benchmark
> #of using the average training y-value to predict the y-value in testing data
> ytrain.avg=mean(train2$voter_movement_to_GOP)
> MSE_avg=mean((ytrain.avg-y.test)^2)
>
> lmCompare <- matrix(c(MSE_avg, MSE_math, MSE_cv),ncol=1,byrow=T)
> rownames(lmCompare) <- c("Average y-value in training", "BIC,Cp and Adjr2 Model",
+                          "CV Model")
> colnames(lmCompare) <- c("Test Mean Squared Error (MSE)")
> lmCompare <- as.table(lmCompare); lmCompare
```

```
                          Test Mean Squared Error (MSE)
Average y-value in training                  0.0007399153
BIC,Cp and Adjr2 Model                       0.0004972727
CV Model                                     0.0005004732
```

We can see that clearly both the model selected BIC, Cp and Adjr2 and CV model outperform the benchmark model of using average y-value in training data.Given the similar performance of both models on training and test data, overall we prefer CV model because of its simplicity with fewer number of predictors.

Interpreting the CV Model

In order to make the interpretation easier, we applied two transformations to the CV model.

First, many independent variables (e.g. HispanicPct2010) are percentages expressed in percentage points (range from 0 to 100). However, the dependent variable "voter_movement_to_GOP" is percentages expressed in decimals (its absolute value ranges from 0 to 1). Hence we multiplied the dependent variable by 100 to also express it in percentage points. This scaled up all beta coefficient estimates by 100.

Next, the new beta coefficient estimates tells us that 1 unit change in each predictor is responsible for BETA percentage points increase in Trump's vote share. This implies that for 1 percentage point increase in Trump's vote share, we need 1/BETA unit change in each predictor. Since this interpretation makes it easier for us to analyse how each predictor explains changes in Trump's vote share, we calculated reciprocals of all beta coefficients.

The transformed regression table is summarized below.

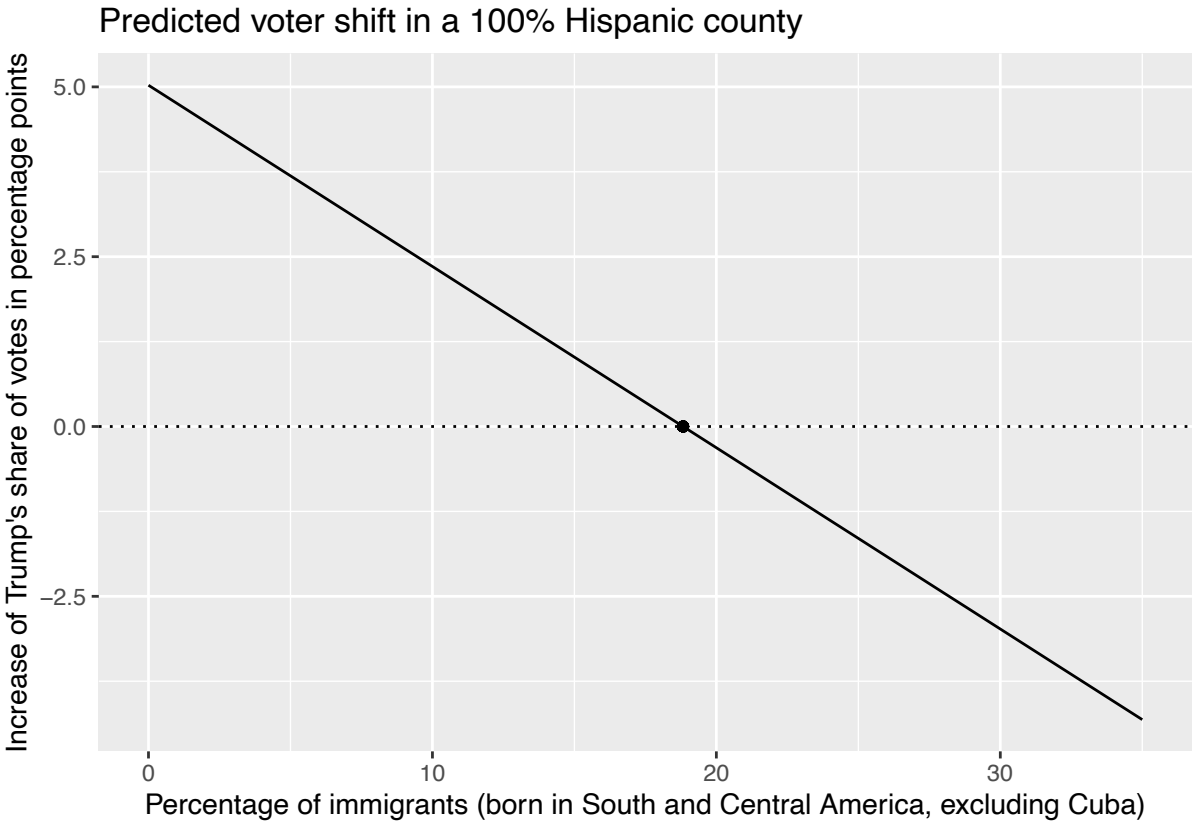
```
> lm_cv_transformed <- lm (100*voter_movement_to_GOP~NetMigrationRate1019
+                          +HispanicPct2010+NonEnglishHHPct+Ed5CollegePlusPct+FemaleHHPct
+                          +ForeignBornCentralSouthAmPct+cases_per_100000, data=train2)
> coef_cv_transformed <- 1/coef(lm_cv_transformed)
> stargazer(lm_cv_transformed,coef=list(coef_cv_transformed),ci=T,ci.level=0.95,
+           type="text",algin=T,title="Interpreting the CV Model",single.row=T)
```

```
Interpreting the CV Model
================================================================
                                    Dependent variable:
                                -------------------------------
                                   100 * voter_movement_to_GOP
----------------------------------------------------------------
NetMigrationRate1019            -26.117*** (-26.137, -26.097)
HispanicPct2010                  19.903*** (19.883, 19.922)
NonEnglishHHPct                    2.504*** (2.413, 2.595)
Ed5CollegePlusPct                -8.385*** (-8.402, -8.369)
FemaleHHPct                      -5.717*** (-5.753, -5.681)
ForeignBornCentralSouthAmPct     -3.747*** (-3.814, -3.681)
cases_per_100000             7,640.491*** (7,640.491, 7,640.491)
Constant                           0.188 (-0.433, 0.808)
----------------------------------------------------------------
Observations                              1,000
R2                                        0.345
Adjusted R2                               0.340
Residual Std. Error                 2.309 (df = 992)
F Statistic                       74.518*** (df = 7; 992)
================================================================
Note:                                *p<0.1; **p<0.05; ***p<0.01

Interpreting the CV Model
====
TRUE
----
```

Finally, we produced two more graphs to assist our interpretations.

The first graph is produced for analyzing the variable ForeignBornCentralSounthAmPct:

```
> cvtable = tidy(lm_cv)
> Hiscoeff = as.numeric(cvtable[3,2])
> Forcoeff = as.numeric(cvtable[7,2])
> intersec = (-Hiscoeff*100)/Forcoeff
> x=seq(0,35,1)
> y=(x*Forcoeff+Hiscoeff*100)*100
> data.frame(cbind(x,y)) %>%
+   ggplot(aes(x=x,y=y)) +
+   geom_line() +
+   ggtitle("Predicted voter shift in a 100% Hispanic county") +
+   xlab("Percentage of immigrants (born in South and Central America, excluding Cuba)") +
+   ylab("Increase of Trump's share of votes in percentage points") +
+   geom_hline(yintercept=0,linetype="dotted") +
+   geom_point(aes(x=intersec,y=0))
```



The second graph is produced for analyzing the variable NetMigrationRate1019

```
> migration = vector(length = 9)
> ruralness = seq(1,9,1)
> for (i in 1:9) {
+   a = train_and_test %>%
+     filter(ruralurban_cc==i)
+   migration[i] = mean(a$NetMigrationRate1019)
+ }
> data = data.frame(migration,ruralness)
> ggplot(data, aes(x=ruralness, y=migration)) +
+   geom_line(stat="identity") +
+   geom_point(aes(x=ruralness, y=migration)) +
+   ggtitle("Average net migration and ruralness of county") +
+   xlab("") +
+   ylab("net migration rate") +
+   theme(axis.title.y = element_text(size = 8)) +
+   geom_hline(yintercept=0,linetype="dotted") +
+   scale_x_discrete(limits=c("urban","2","3","4","suburban","6","7","8","rural"))
```