2020 Election Project

March 09, 2022

Project Goal

We are primarily working towards building state/county-level red/blue map plots that are commonly shown on media coverage or google search.

Additionally, we will combine the United States county-level census data with the election data. Our target would then be building and selecting classification models to predict the election winner.

Data

In the dataset **election.raw**, we have the data containing county-level election results.

In the dataset **census**, we have the 2017 United States county-level census data.

Election Data

1 - ***Report the dimension of election.raw. Are there missing values in the data set? Compute the total number of distinct values in state in election.raw to verify that the data contains all states and a federal district.***

Let's get an overview of the **election.raw** data set (see code in .rmd file).

There are 31167 rows, 5 columns and 0 missing values in the dataset. Additionally, we know that there are 51 distinct values in **state** in **election.raw**, which consists of all the states and a federal district.

Census Data

2 - ***Report the dimension of census. Are there missing values in the data set? Compute the total number of distinct values in county in census. Compare the values of total number of distinct county in census with that in election.raw. Comment on your findings.***

Now let's get an overview of the **census** data set.

There are 3220 rows, 37 columns and 1 missing value in the dataset. Additionally, we see that there are 1955 distinct values in **County** in **census**.

Comparing the number of distinct counties in **census** to the number of distinct states in **election.raw**, we expectedly see that there are many more distinct counties than states throughout the US.

Data wrangling

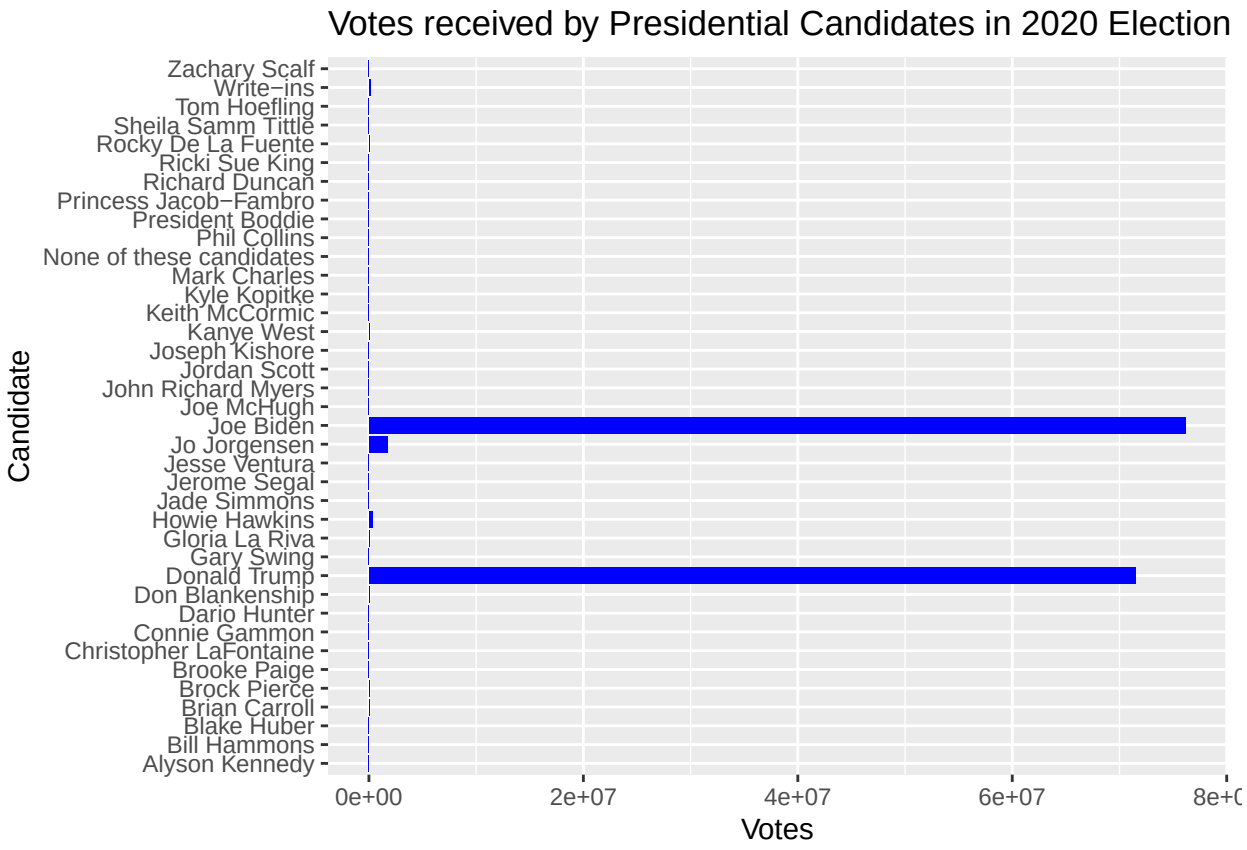
3 - ***Construct aggregated data sets from election.raw data: i.e.,***

In **election.state**, we aggregated data to see the amount of votes each candidate got in each state. In **election.total**, we aggregated data to see the amount of votes each candidate got nationwide.

4 - ***How many named presidential candidates were there in the 2020 election? Draw a bar chart of all votes received by each candidate. You can split this into multiple plots or may prefer to plot the results on a log scale. Either way, the results should be clear and legible! (For fun: spot Kanye West among the presidential candidates!)***

Analyzing the election.raw dataset, we know that there are 38 candidates in the 2020 election, including write-ins (write-ins counts as only 1).

Now let's draw a bar chart of all votes received by each candidate.



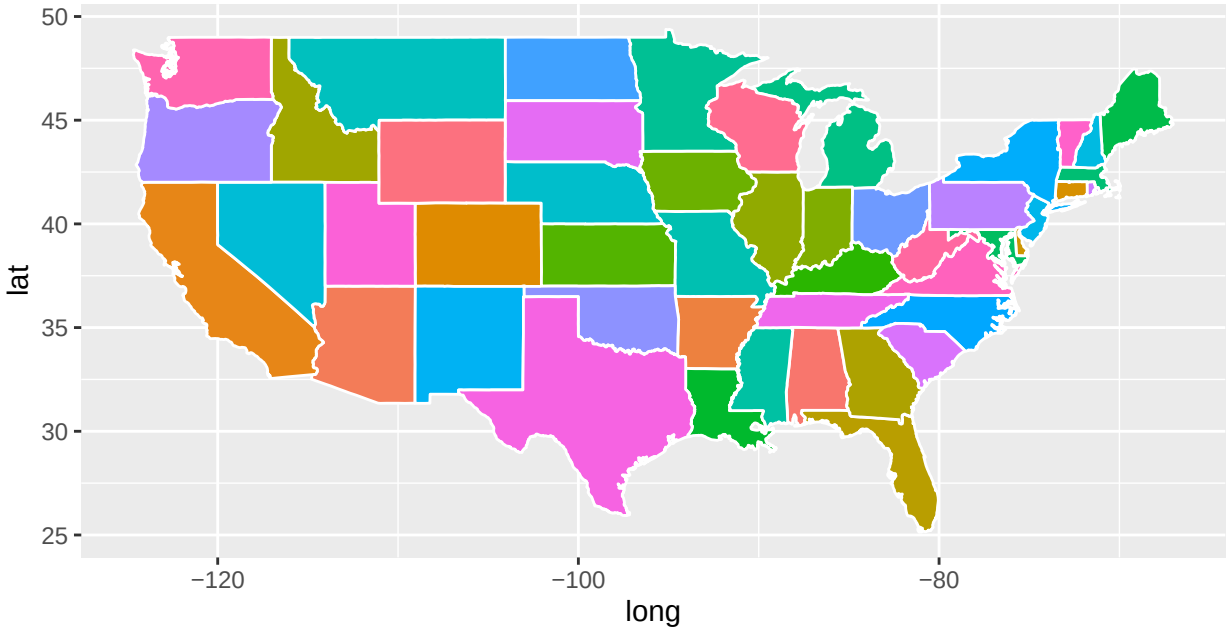
5 - ***Create data sets county.winner and state.winner by taking the candidate with the highest proportion of votes in both county level and state level.***

Now we create the data sets **county.winner** and **state.winner** by taking the candidate with the highest proporation of votes in both county level and state level.

Visualization

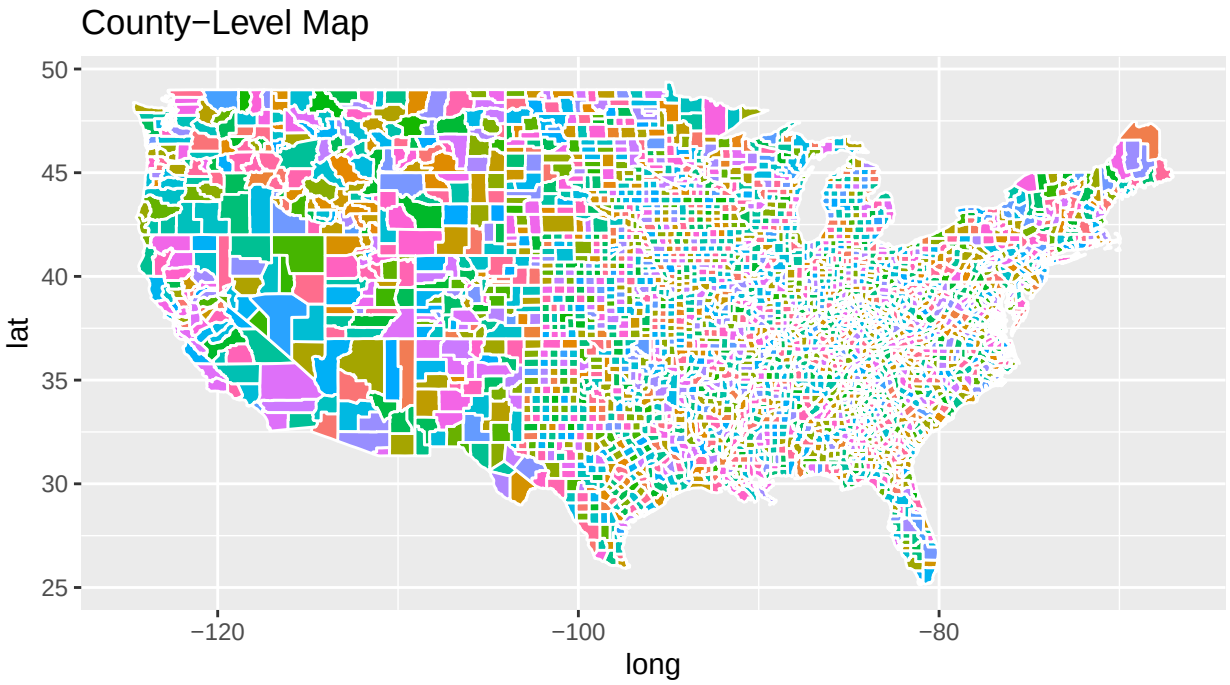
Visualization is important for gaining insight and intuition. We use ggplot2 to draw maps.

Consider the following map that contains information to to draw white polygons outlining states.



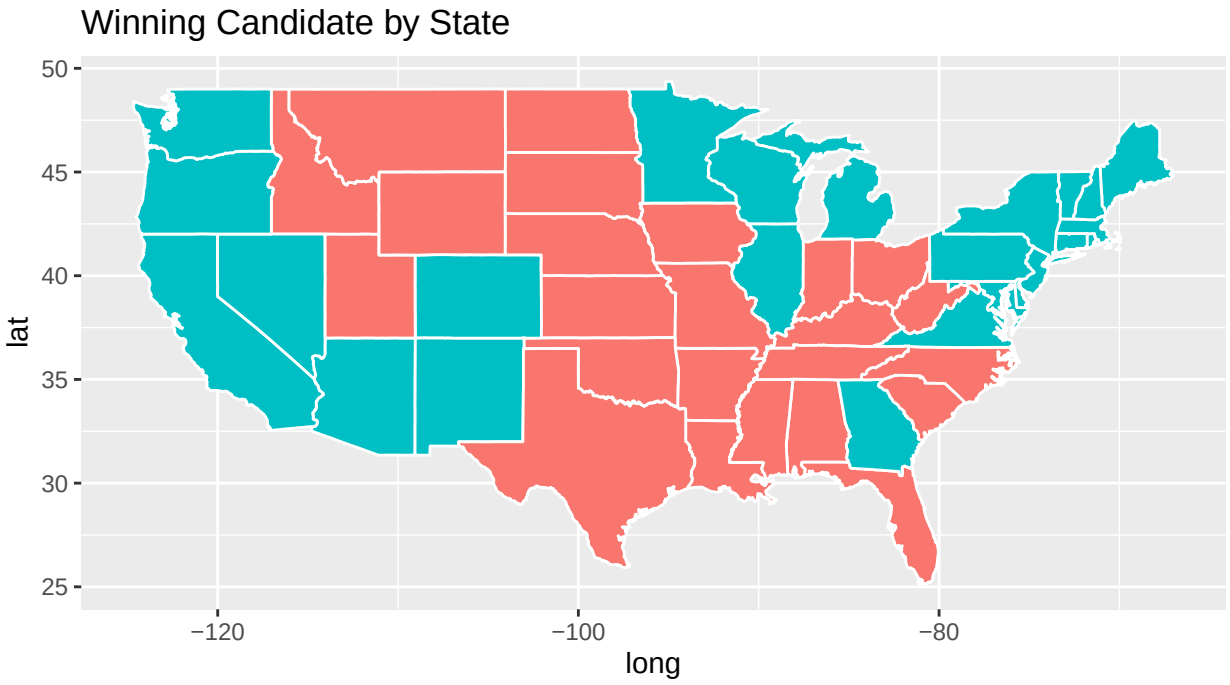
6 - ***Use similar code to above to draw county-level map by creating counties = map_data("county"). Color by county.***

We use similar code to draw the county-level map below.



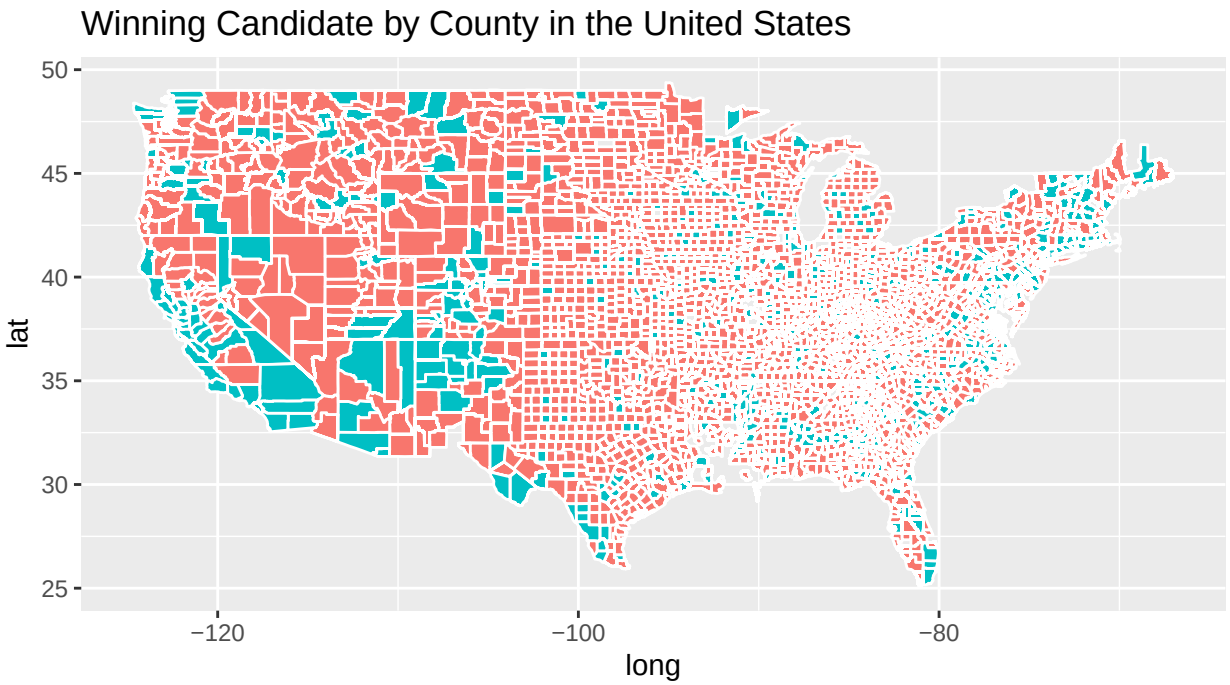
7 - *Now color the map by the winning candidate for each state*

Here we look to color the map by the winning candidate for each state. This map should be consistent with the New York Times Map and other electoral college winner maps. Blue indicates a state won for Biden, while red indicates a state won for Trump.



8 - *Color the map of the state of California by the winning candidate for each county.*

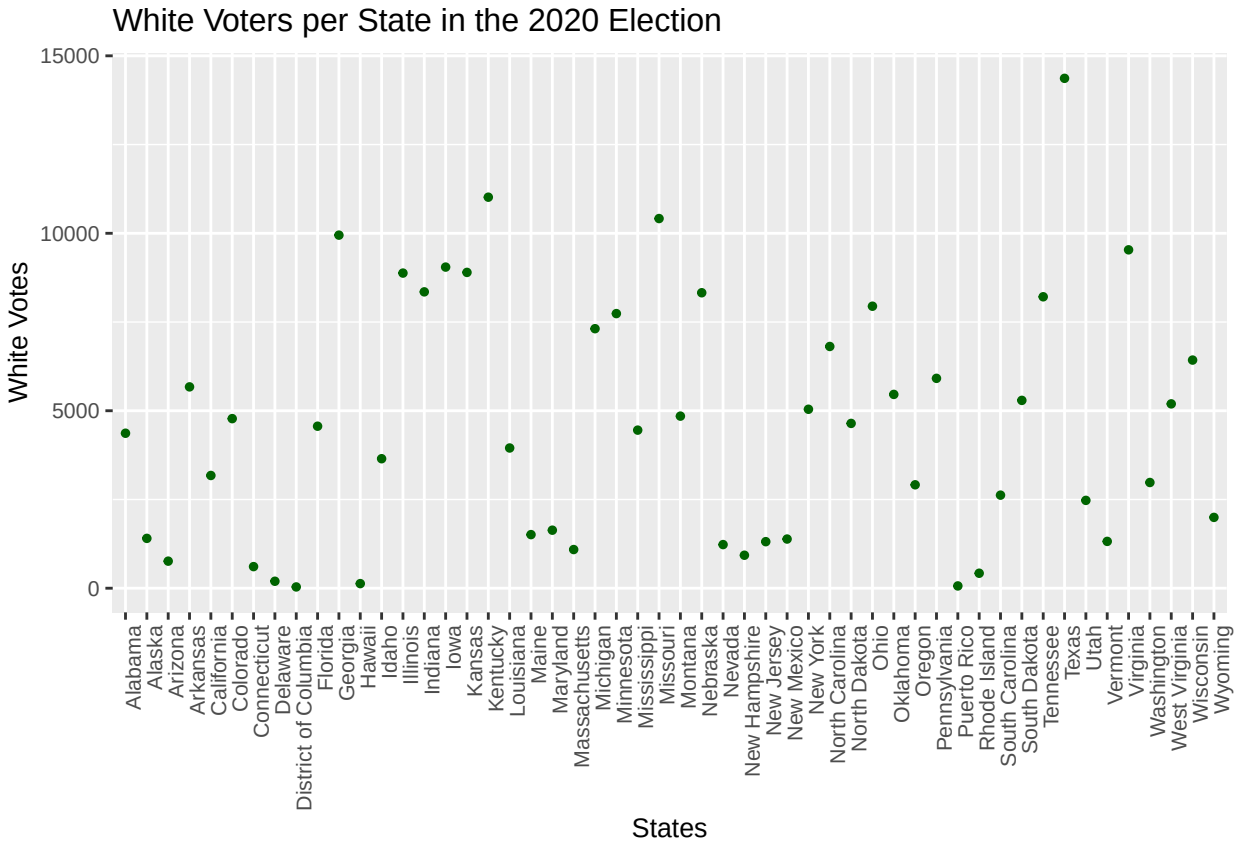
Here we look to color the map of the state of California by the winning candidate for each county. Note that some counties have not finished counting the votes and therefore do no have a winner.

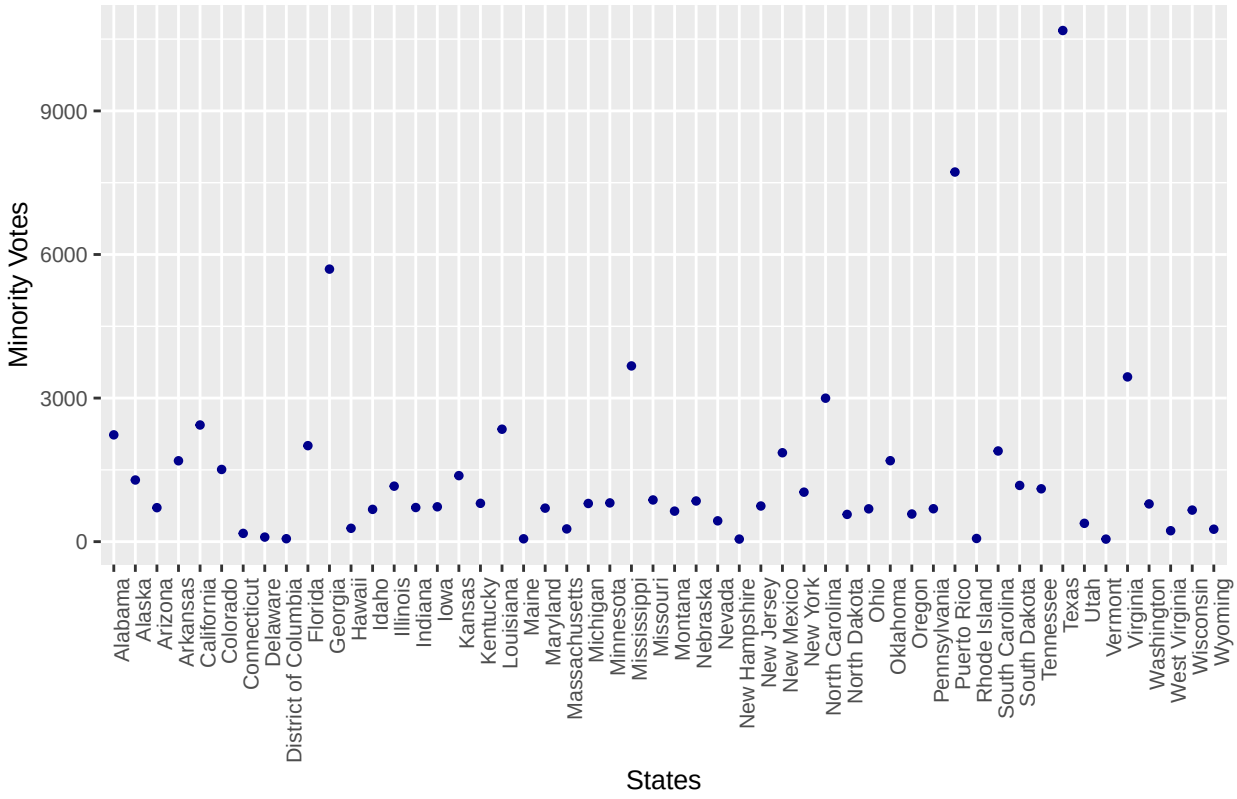


9 - ***Create a visualization of your choice using census data. Many exit polls noted that demographics played a big role in the election. Use this Washington Post article and this R graph gallery for ideas and inspiration.***

Exit polls told us that the demographics played a major role in the outcome of the 2020 election. With this in mind, we take a look at some of these racial demographics, specifically, the amount of white people that voted in each state versus the amount of minorities that voted.

Using **census** we combine *Hispanic*, *Black*, *Native*, *Asian*, and *Pacific* attributes to form *Minority*


White Voters per State in the 2020 Election
White Votes
15000
10000
5000
0
Alabama
Alaska
Arizona
Arkansas
California
Colorado
Connecticut
Delaware
District of Columbia
Florida
Georgia
Hawaii
Idaho
Illinois
Indiana
Iowa
Kansas
Kentucky
Louisiana
Maine
Maryland
Massachusetts
Michigan
Minnesota
Mississippi
Missouri
Montana
Nebraska
Nevada
New Hampshire
New Jersey
New Mexico
New York
North Carolina
North Dakota
Ohio
Oklahoma
Oregon
Pennsylvania
Puerto Rico
Rhode Island
South Carolina
South Dakota
Tennessee
Texas
Utah
Vermont
Virginia
Washington
West Virginia
Wisconsin
Wyoming
States


Minority Voters per State in the 2020 Election
Minority Votes
9000
6000
3000
0
Alabama
Alaska
Arizona
Arkansas
California
Colorado
Connecticut
Delaware
District of Columbia
Florida
Georgia
Hawaii
Idaho
Illinois
Indiana
Iowa
Kansas
Kentucky
Louisiana
Maine
Maryland
Massachusetts
Michigan
Minnesota
Mississippi
Missouri
Montana
Nebraska
Nevada
New Hampshire
New Jersey
New Mexico
New York
North Carolina
North Dakota
Ohio
Oklahoma
Oregon
Pennsylvania
Puerto Rico
Rhode Island
South Carolina
South Dakota
Tennessee
Texas
Utah
Vermont
Virginia
Washington
West Virginia
Wisconsin
Wyoming
States

From these plots, we see that white voters showed up to the polls in much higher amounts than minorities in 2020. This is expected, as whites are the majority racial demographic in the United States.

10 - ***The census data contains county-level census information. In this problem we clean and aggregate the information as follows. Many columns are perfectly colineared, in which case one column should be deleted.***

Here we attempt to clean and aggregate this information. After cleaning this information, we print the first several rows the new edited data set.

```
## # A tibble: 6 x 26
##   CountyId State County TotalPop   Men White VotingAgeCitizen Income Poverty
##      <dbl> <chr> <chr>     <dbl> <dbl> <dbl>            <dbl>  <dbl>   <dbl>
## 1     1001 Alab~ Autau~    55036  48.9  75.4             74.5  55317    13.7
## 2     1003 Alab~ Baldw~   203360  48.9  83.1             76.4  52562    11.8
## 3     1005 Alab~ Barbo~    26201  53.3  45.7             77.4  33368    27.2
## 4     1007 Alab~ Bibb ~    22580  54.3  74.6             78.2  43404    15.2
## 5     1009 Alab~ Bloun~    57667  49.4  87.4             73.7  47412    15.6
## 6     1011 Alab~ Bullo~    10478  53.6  21.6             78.4  29655    28.5
## # ... with 17 more variables: ChildPoverty <dbl>, Professional <dbl>,
## #   Service <dbl>, Office <dbl>, Production <dbl>, Drive <dbl>, Carpool <dbl>,
## #   Transit <dbl>, OtherTransp <dbl>, WorkAtHome <dbl>, MeanCommute <dbl>,
## #   Employed <dbl>, PrivateWork <dbl>, SelfEmployed <dbl>, FamilyWork <dbl>,
## #   Unemployment <dbl>, Minority <dbl>
```

Dimensionality Reduction

11 - ***Run PCA for the cleaned county level census data (with State and County excluded). ave the first two principle components PC1 and PC2 into a two-column data frame, call it pc.county. Discuss whether you chose to center and scale the features before running PCA and the reasons for your choice. What are the three features with the largest absolute values of the first principal component? Which features have opposite signs and what does that mean about the correlation between these features?***

Below we run PCA for the cleaned county level census data.

Here we decided to center and scale the features before running PCA. If we don't center and scale the features before running PCA, most of the principal components that we observed would be driven by a weighted variable that has the largest mean and variance. This would not be productive, as it makes incredibly difficult to scale the other variables evenly.

```
##                   pc.rotation...1.
## TotalPop                0.02836514
## Men                     0.02140376
## White                   0.29354420
## VotingAgeCitizen        0.01905852
## Income                  0.32103183
## Poverty                 0.38174893
## ChildPoverty            0.38107064
## Professional            0.22995055
## Service                 0.20230483
## Office                  0.06700086
```

```
## Production             0.08138976
## Drive                  0.10194990
## Carpool                0.06131734
## Transit                0.03823974
## OtherTransp            0.02383669
## WorkAtHome             0.20418190
## MeanCommute            0.07813004
## Employed               0.35109781
## PrivateWork            0.06782284
## SelfEmployed           0.12362408
## FamilyWork             0.06576187
## Unemployment           0.33698496
## Minority               0.29709397
```

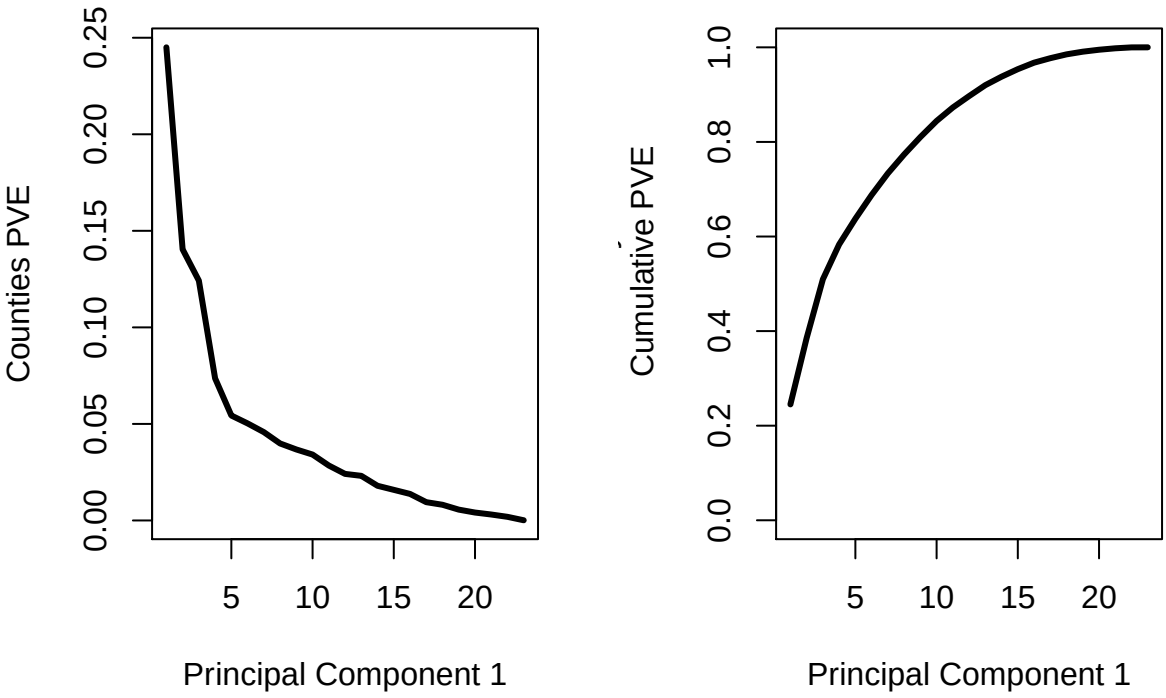
As we see above, the three features with the largest absolute values of the first principal component are *Poverty*, *ChildPoverty*, and *Employed*

```
##                   pc.rotation...1.
## TotalPop                0.02836514
## Men                     0.02140376
## White                   0.29354420
## VotingAgeCitizen        0.01905852
## Income                  0.32103183
## Poverty                -0.38174893
## ChildPoverty           -0.38107064
## Professional            0.22995055
## Service                -0.20230483
## Office                 -0.06700086
## Production             -0.08138976
## Drive                  -0.10194990
## Carpool                -0.06131734
## Transit                 0.03823974
## OtherTransp            -0.02383669
## WorkAtHome              0.20418190
## MeanCommute            -0.07813004
## Employed                0.35109781
## PrivateWork             0.06782284
## SelfEmployed            0.12362408
## FamilyWork              0.06576187
## Unemployment           -0.33698496
## Minority               -0.29709397
```

For **pc.county**, *Poverty*, *ChildPoverty*, *Service*, *Office*, *Production*, *Drive*, *Carpool*, *OtherTransp*, *MeanCommute*, *Unemployment*, and *Minority* all have opposite signs. Since these features have opposite signs, this means that they are negatively correlated. Since all these variables are negatively correlated, they have an inverse relationship with the factor PCA.

12 - *Determine the number of minimum number of PCs needed to capture 90% of the variance for the analysis. Plot proportion of variance explained (PVE) and cumulative PVE.*

Here we determine the number of minimum number of PCs needed to capture 90% of the variance for the analysis. We also plot proportion of variance explained (PVE) and cumulative PVE.



Above we see the plots for proportion of variance explained (PVE) and cumulative PVE. As we see above, the minimum number of PCs needed to capture 90% of the variance for the analysis is 13.

Clustering

13 - ***With census.clean (with State and County excluded), perform hierarchical clustering with complete linkage. Cut the tree to partition the observations into 10 clusters. Re-run the hierarchical clustering algorithm using the first 2 principal components from pc.county as inputs instead of the original features. Compare the results and comment on your observations. For both approaches investigate the cluster that contains Santa Barbara County. Which approach seemed to put Santa Barbara County in a more appropriate clusters? Comment on what you observe and discuss possible explanations for these observations.***

Now with **census.clean**, we perform hierarchical clustering with complete linkage. First, we cut the tree to partition the observations into 10 clusters.

```
## first.clust
##    1    2    3    4    5    6    7    8    9   10
## 2924  191    6    5   31    1   17    6   34    4
```

Now we re-run the hierarchical clustering algorithm using the first 2 principal components from **pc.county** as inputs instead of the original features.

```
## second.clust
##    1    2    3    4    5    6    7    8    9   10
## 2064  404  252   76  209   20  163    1   17   13
```

Now let's investigate the cluster that contains *Santa Barbara County*.

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

```
## [1] 7
```

In the first cluster before using PCA, clustering decreases from 2924 to 31 in the first 5 clusters. It then decreases to 1, increases to 17, decrease to 6, increases to 34, and decreases to 4.

When we recluster with PCA in the second cluster, clustering decreases from 2064 to 209 in the first 5 clusters. Compared to the first cluster, the second cluster similarly follows the same pattern of increasing and decreasing after the 5th cluster.

From these trends, I belive Santa Barbara is more appropriately place in Clust 2. Clust 2 is the complete linkage cluster, which is more suitable as a complete link since it is less susceptible to noise and outliers. We also know that a smaller distance from the mean illustrates a more appropriate cluster and contains less variance than variables further away from each other.

Classification

Here our goal is to see if we can use census information in a county to predict the winner in that county.

```
## # A tibble: 1,805 x 24
##    candidate TotalPop   Men White VotingAgeCitizen Income Poverty ChildPoverty
##    <fct>        <dbl> <dbl> <dbl>            <dbl>  <dbl>   <dbl>        <dbl>
##  1 Joe Biden   555036  48.4  58.5             72.7  68336    11.9         15.9
##  2 Donald T~   215551  48.5  74.9             76.8  57901    12           20.8
##  3 Joe Biden   672391  47.4  36               74.8  77649    17.4         25.5
##  4 Joe Biden   259865  48.4  62               77.2  45478    23.3         22
##  5 Donald T~    27537  52.4  81.9             75.1  59506    17.2         22
##  6 Donald T~   180117  49.7  77.3             76.1  50283    15.4         22.1
##  7 Donald T~   568183  48.9  75.3             78.3  51536    13.4         19.4
##  8 Joe Biden  1890416  48.7  38.2             66.0  54895    14           19.1
##  9 Donald T~   173236  48.7  84.4             83.4  46511    12           19.6
## 10 Donald T~   141373  48.4  88.6             83.2  40574    17.4         30.6
## # ... with 1,795 more rows, and 16 more variables: Professional <dbl>,
## #   Service <dbl>, Office <dbl>, Production <dbl>, Drive <dbl>, Carpool <dbl>,
## #   Transit <dbl>, OtherTransp <dbl>, WorkAtHome <dbl>, MeanCommute <dbl>,
## #   Employed <dbl>, PrivateWork <dbl>, SelfEmployed <dbl>, FamilyWork <dbl>,
## #   Unemployment <dbl>, Minority <dbl>
```

14 - *Understand the code above. Why do we need to exclude the predictor party from election.cl?*

We need to exclude the predictor *party* from **election.cl** because with the given set of candidates, *party* is one of the predictors that can be perfectly predicted from one or more of the other independent variables. It would be redundant and counterproductive to include this as a part of **election.cl**.

Now we partition the data into 80% training and 20% testing

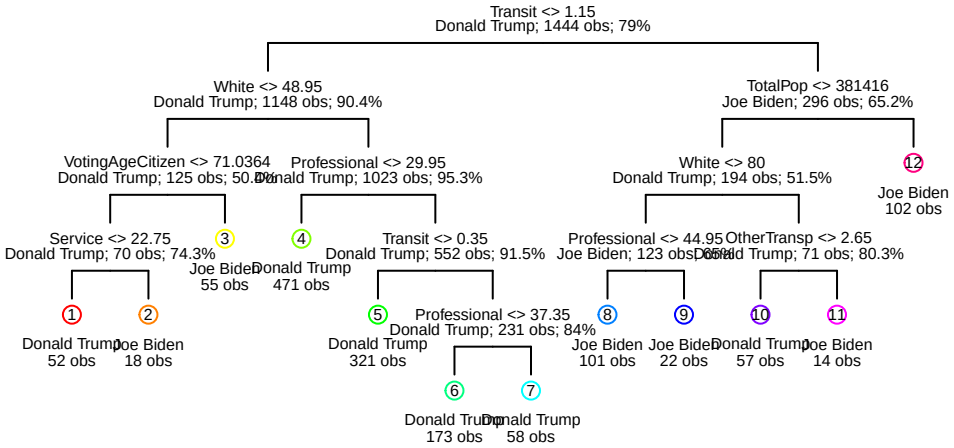
We use the following code to define 10 cross-validation folds:

Below is the error rate function. And the object records is used to record the classification performance of each method in the subsequent problems.

15 - ***Decision tree: train a decision tree by cv.tree(). Prune tree to minimize misclassification error. Be sure to use the folds from above for cross-validation. Visualize the trees before and after pruning. Save training and test errors to records object. Interpret and discuss the results of the decision tree analysis. Use this plot to tell a story about voting behavior.***

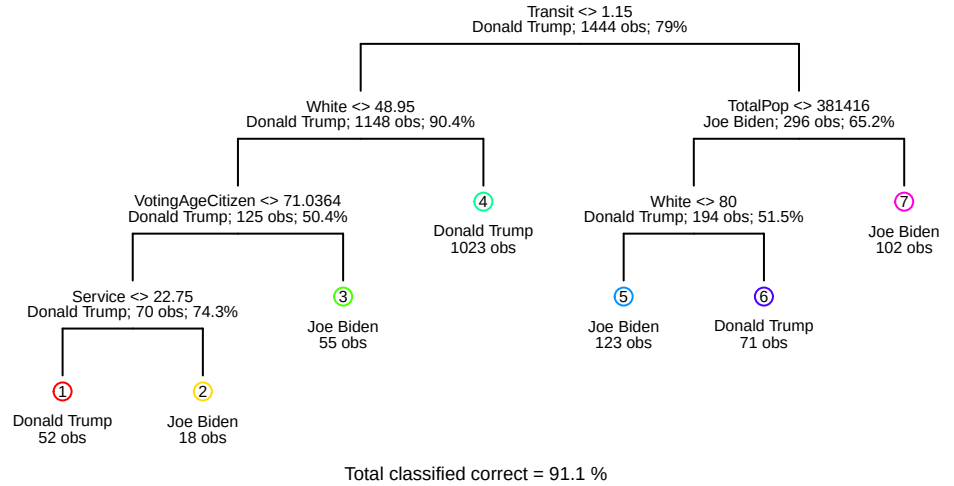
Here we train a decision tree by cv.tree. We prune the tree to minimize misclassification error. Be sure to use the folds from above for cross-validation. Visualize the trees before and after pruning. Save training and test errors to records object. Interpret and discuss the results of the decision tree analysis. Use this plot to tell a story about voting behavior.

Unpruned Decision Tree



Now that we have the initial decision tree, we now work to prune the tree.

Pruned Decision Tree



Now we save these training and test errors to **records** object.

	train.error	test.error
tree	0.0893352	0.1301939
logistic	NA	NA
lasso	NA	NA

As we see above, our initial unpruned decision tree had a 91.3% classification success rate, a solid statistic considering the dataset is large.

Continuing on to our pruned decision tree, we notice that it received a total of 91.1% classification success, 0.2% less than our unpruned tree. The lower classification success could be a product of pruning the tree, as pruning tends to decrease the number of variables that the decision tree utilizes. As we compare the pruned decision tree vs the unpruned tree, we see the pruned tree has fewer variables. Having fewer variables could play a part in having a less accurate model.

However, a 0.2% difference between the two trees is not a significant difference. In this scenario, the pruned decision tree provides a much more clear visualization of our data by allowing us to more clearly identify the factors that can affect a voter's decision making.

Looking more closely at the pruned tree, we see that individuals the worked service jobs were much more likely to vote for Trump than Biden. Additionally, individuals that commuted on public transportation, who also were white, were much more likely to vote for Trump than Biden. Overall, individuals that utilized transit tended to vote for Trump over Biden.

16 - ***Run a logistic regression to predict the winning candidate in each county. Save training and test errors to records variable. What are the significant variables? Are they consistent with what you saw in decision tree analysis? Interpret the meaning of a couple of the significant coefficients in terms of a unit change in the variables.***

Now we run a logistic regression to predict the winning candidate in each county.

```
##
## Call:
## glm(formula = factor(candidate) ~ ., family = "binomial", data = election.tr)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.8427  -0.2544  -0.0993  -0.0222   3.3552
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -2.027e+01  1.189e+01  -1.705  0.08822 .
## TotalPop          1.891e-06  7.630e-07   2.479  0.01319 *
## Men              -3.220e-03  6.133e-02  -0.053  0.95812
## White            -2.038e-01  8.562e-02  -2.380  0.01730 *
## VotingAgeCitizen  1.752e-01  3.130e-02   5.598 2.17e-08 ***
## Income           -1.335e-05  2.002e-05  -0.667  0.50492
## Poverty           1.683e-02  5.159e-02   0.326  0.74421
## ChildPoverty     -4.406e-04  3.200e-02  -0.014  0.98901
## Professional      3.130e-01  5.176e-02   6.047 1.48e-09 ***
## Service           3.265e-01  6.276e-02   5.202 1.97e-07 ***
## Office            1.587e-01  6.313e-02   2.514  0.01194 *
## Production        2.291e-01  5.368e-02   4.267 1.98e-05 ***
## Drive            -2.175e-01  5.685e-02  -3.827  0.00013 ***
## Carpool          -2.247e-01  7.541e-02  -2.980  0.00288 **
## Transit           5.601e-02  1.151e-01   0.487  0.62660
```

```
## OtherTransp       1.305e-01  1.167e-01   1.118  0.26370
## WorkAtHome       -1.312e-01  8.802e-02  -1.491  0.13595
## MeanCommute       3.580e-02  3.200e-02   1.119  0.26330
## Employed          2.442e-01  4.228e-02   5.776 7.65e-09 ***
## PrivateWork       7.147e-02  2.819e-02   2.535  0.01124 *
## SelfEmployed      1.955e-02  5.889e-02   0.332  0.73986
## FamilyWork       -3.708e-01  3.352e-01  -1.106  0.26854
## Unemployment      1.475e-01  5.167e-02   2.855  0.00431 **
## Minority         -5.898e-02  8.261e-02  -0.714  0.47525
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1483.67  on 1443  degrees of freedom
## Residual deviance:  523.43  on 1420  degrees of freedom
## AIC: 571.43
##
## Number of Fisher Scoring iterations: 7
```

```
##               true
## predicted      Donald Trump Joe Biden
##   Donald Trump         1109        65
##   Joe Biden              32       238
```

```
##               true
## predicted      Donald Trump Joe Biden
##   Donald Trump          279        22
##   Joe Biden               8        52
```

	train.error	test.error
tree	0.0893352	0.1301939
logistic	0.0671745	0.0831025
lasso	NA	NA

The significant variables here are TotalPop, White, VotingAgeCitizen, Professional, Service, Office, Production, Drive, Carpool, Employed, PrivateWork, and Unemployment. Several of these significant variables were also prevalent in our previous portion regarding the decision trees.

Looking specifically at *Unemployment*, we understand that this is a significant variable as unemployment plays a big factor in determining what socio-economic class an individual belongs to, which also has voting party implications.

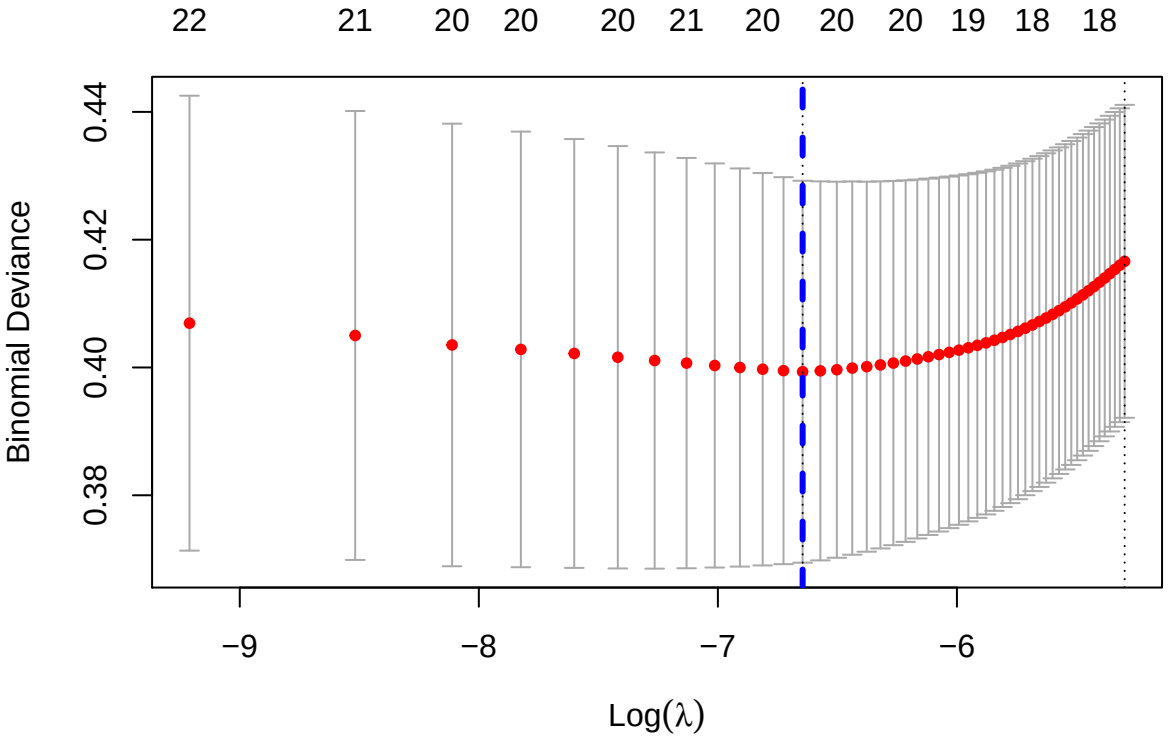
Service is also another significant factor, as individuals that work service jobs are more likely to vote Republican. Often times, service jobs are viewed as "Blue-Collar" work, a characteristic often connected to right-leaning individuals in the middle of the United States. A 1-unit change in *Service* has a relatively significant shift in voting behavior.

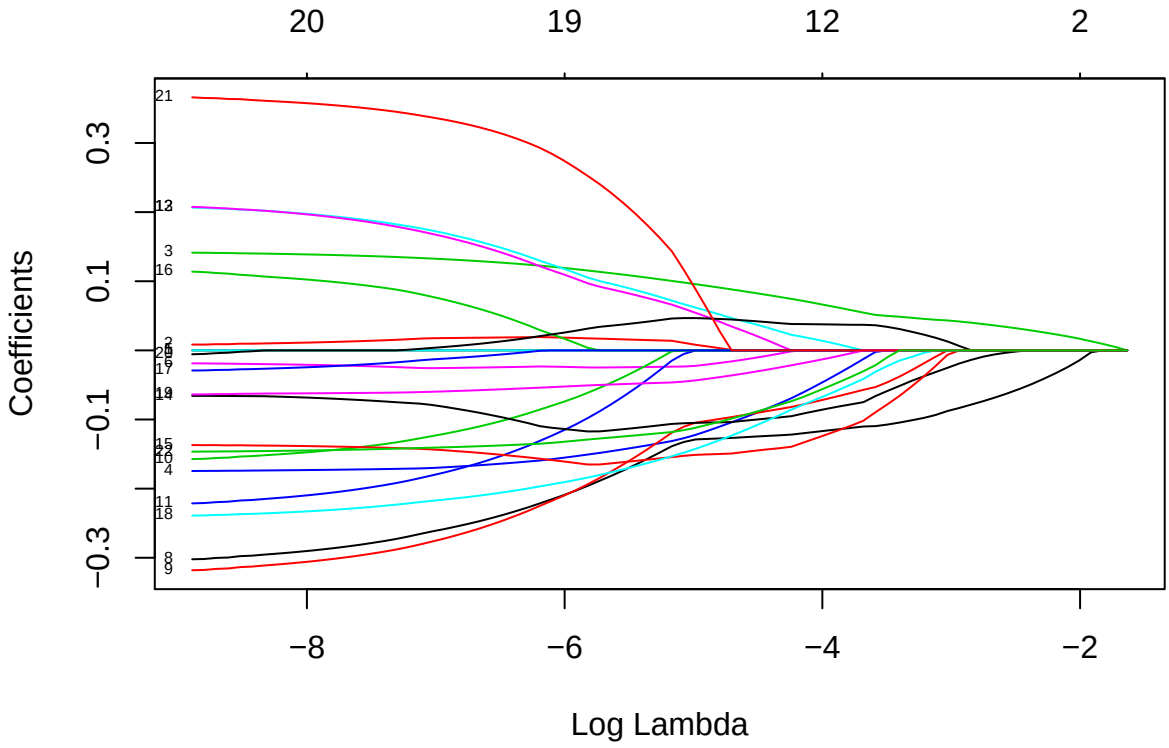
Carpool is another significant factor. People that carpool are likely more enivronmentally conscious, a characteristic associated with more left-leaning individuals. Conservatives typically have a lesser belief in climate change, so it is fair to say more environmentally conscious individuals likely lean left.

When looking at the results of logistic regression that attempt to predict the winning candidate in each county, we see the model predicted the outcomes fairly well. Out of all the counties predicted for Donald Trump to win, Trump was actually won 279 of these counties and Biden was won 22 of these counties. Out of all the counties predicted for Joe Biden to win, Biden actually won 52 of these and Trump won 8. Note

that these values above are from our regression on the test set, which is only a smaller portion of the whole dataset.

17 - ***One way to control overfitting in logistic regression is through regularization. Use the cv.glmnet function from the glmnet library to run a 10-fold cross validation and select the best regularization parameter for the logistic regression with LASSO penalty. Set*** $\lambda = seq(1, 50) \cdot 1e-4$ ***in cv.glmnet() function to set pre-defined candidate values for the tuning parameter*** λ***. What is the optimal value of*** λ ***in cross validation? What are the non-zero coefficients in the LASSO regression for the optimal value of*** λ***? How do they compare to the unpenalized logistic regression? Comment on the comparison. Save training and test errors to the records variable.***






```
##                                1
## (Intercept)        2.337139e+01
## TotalPop          -1.957794e-06
## Men                1.835089e-02
## White              1.288499e-01
## VotingAgeCitizen  -1.659224e-01
## Income             0.000000e+00
## Poverty           -2.480415e-02
## ChildPoverty       0.000000e+00
## Professional      -2.463360e-01
## Service           -2.567317e-01
## Office            -1.093702e-01
## Production        -1.619329e-01
## Drive              1.565779e-01
## Carpool            1.504251e-01
## Transit           -9.081273e-02
## OtherTransp       -1.484584e-01
## WorkAtHome         5.919274e-02
## MeanCommute       -8.291153e-03
## Employed          -2.097245e-01
## PrivateWork       -5.744986e-02
## SelfEmployed       9.800829e-03
## FamilyWork         3.219141e-01
## Unemployment      -1.385507e-01
## Minority           0.000000e+00
```

	train.error	test.error
tree	0.0893352	0.1301939
logistic	0.0671745	0.0831025
lasso	0.0713296	0.0858726

The optimal λ value in cross validation is 0.0013.

The non-zero coefficients in the LASSO regression for the optimal value of λ were all of the variables excluding *Income*, *ChildPoverty*, and *Minority*. This is expected, as many of the variables have influence in the outcome.

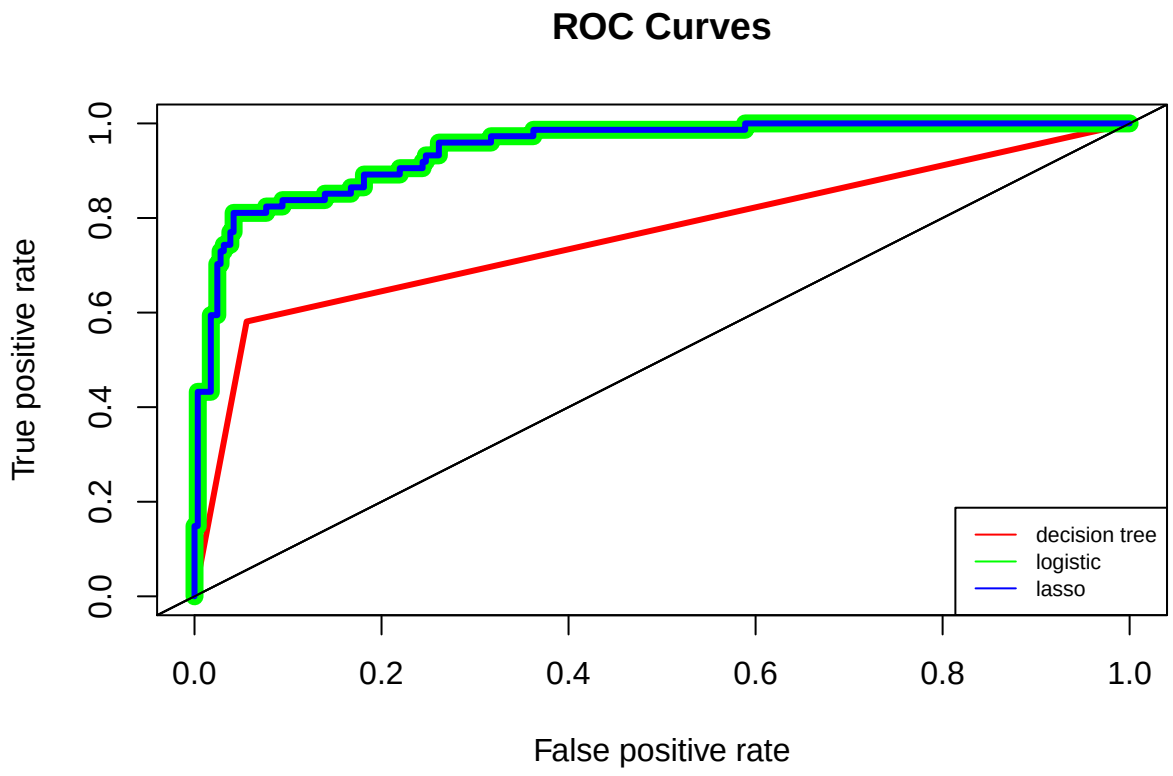
The LASSO regression is used for data sets with not enough data, which has high variance estimates. This is in contrast to logistic regression, which is better for big data. We use the LASSO regression to utilize the shrinkage method and reduce the variance. However, because our data set is large and many of our variables influence our outcome, they don't have a coefficient of zero.

The largest non-zero coefficients that from the LASSO regression are *Transit*, *SelfEmployed*, *MeanCommute*, and *PrivateWork*. When comparing this with the results of the unpenalized logistic regression, we see that *Transit* and *PrivateWork* are also two of the highest coefficient estimates. Additionally, we realize that the LASSO regression has less variables work with when compared to the logistic regression, as some of the variables in the LASSO regression equal 0.

Lastly, the LASSO and logistic regression fits look very similar as the errors are so very close to each other. The logistic regression model displays to us that there is already enough data to estimate the coefficients to a high accuracy. We can state that the LASSO regression does not necessarily offer any additional significant information with the smaller data set.

18 - *Compute ROC curves for the decision tree, logistic regression and LASSO logistic regression using predictions on the test data. Display them on the same plot. Based on your classification results, discuss the pros and cons of the various methods. Are the different classifiers more appropriate for answering different kinds of questions about the election?*

Here we compute ROC curves for the decision tree, logistic regression, and LASSO regression using predictions on the test data.



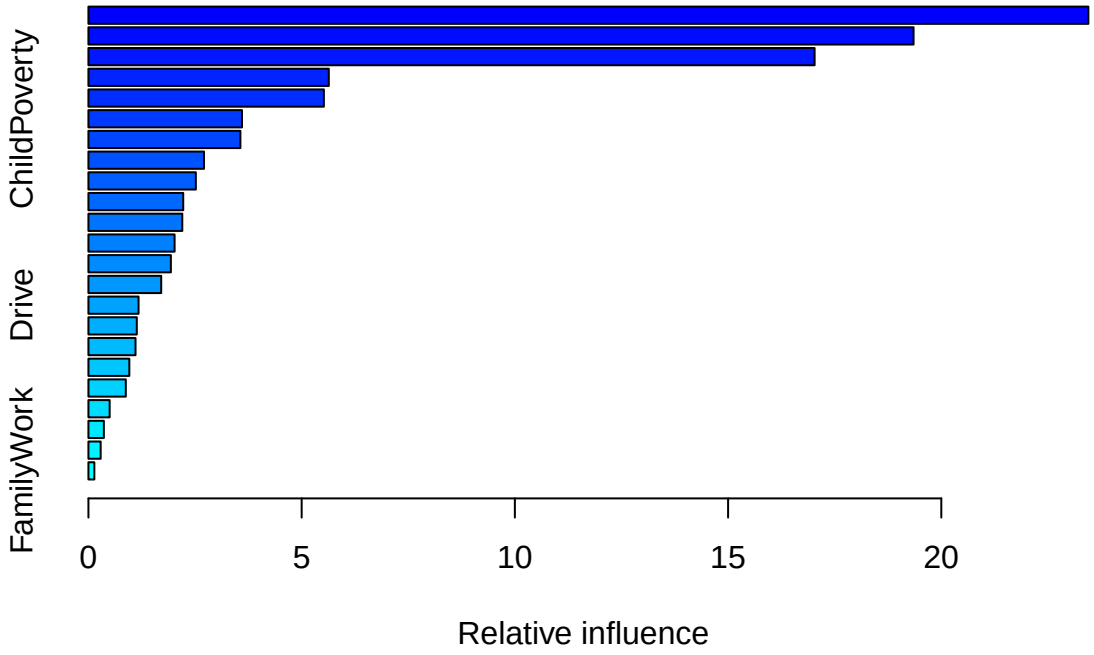
Above, we see the ROC curves on the same plot. Note that the AUC for the decision tree, logistic regression, and LASSO regression are 0.762666, 0.9444863, and 0.9444863, respectively. From these values and the ROC Curves, we can conclude that the logistic regression and LASSO regression give the highest true positive rates. Analyzing the AUC values, we also can conclude that the logistic and LASSO models are the best predictive models.

When compared to the other two models, the decision tree fails to analyze voter behavior as competently. The LASSO and logistic models gives us more insight on voter behavior.

19 - *Explore additional classification methods. Consider applying additional two classification methods from KNN, LDA, QDA, SVM, random forest, boosting, neural networks etc. (You may research and use methods beyond those covered in this course). How do these compare to the tree method, logistic regression, and the lasso logistic regression?*

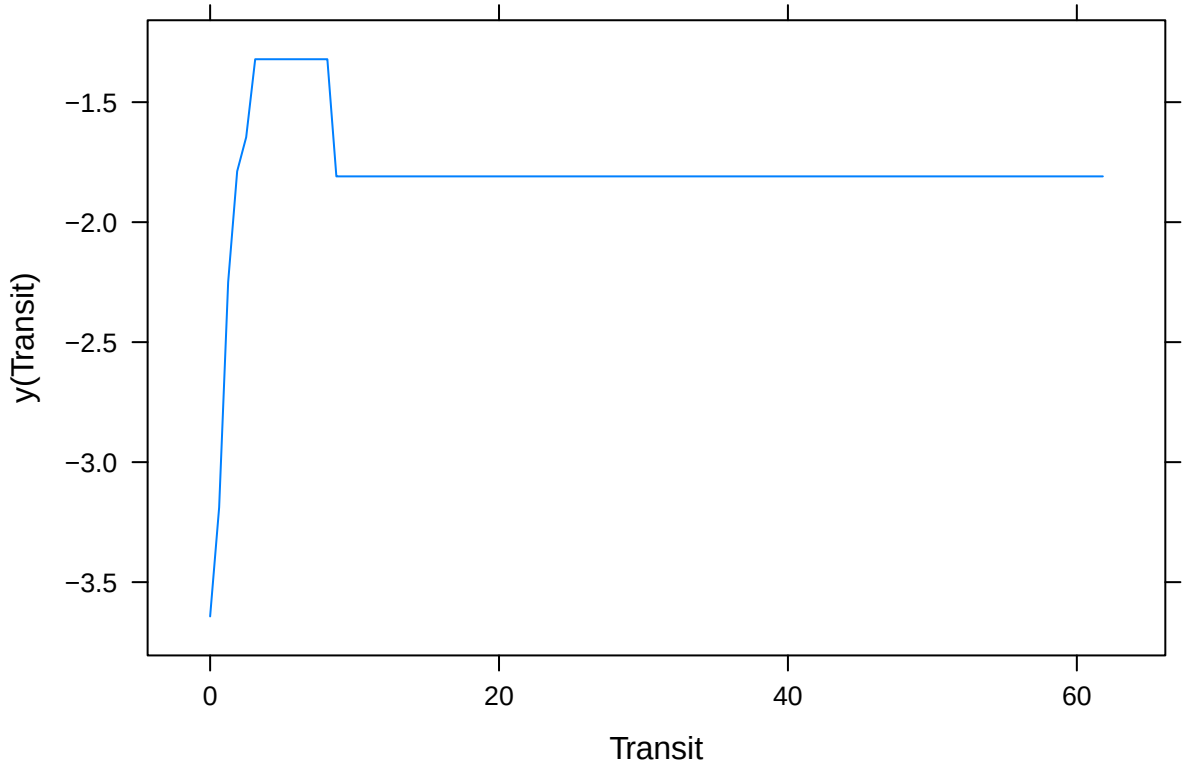
Here we will apply two additional classification methods, specifically random forests and boosting.

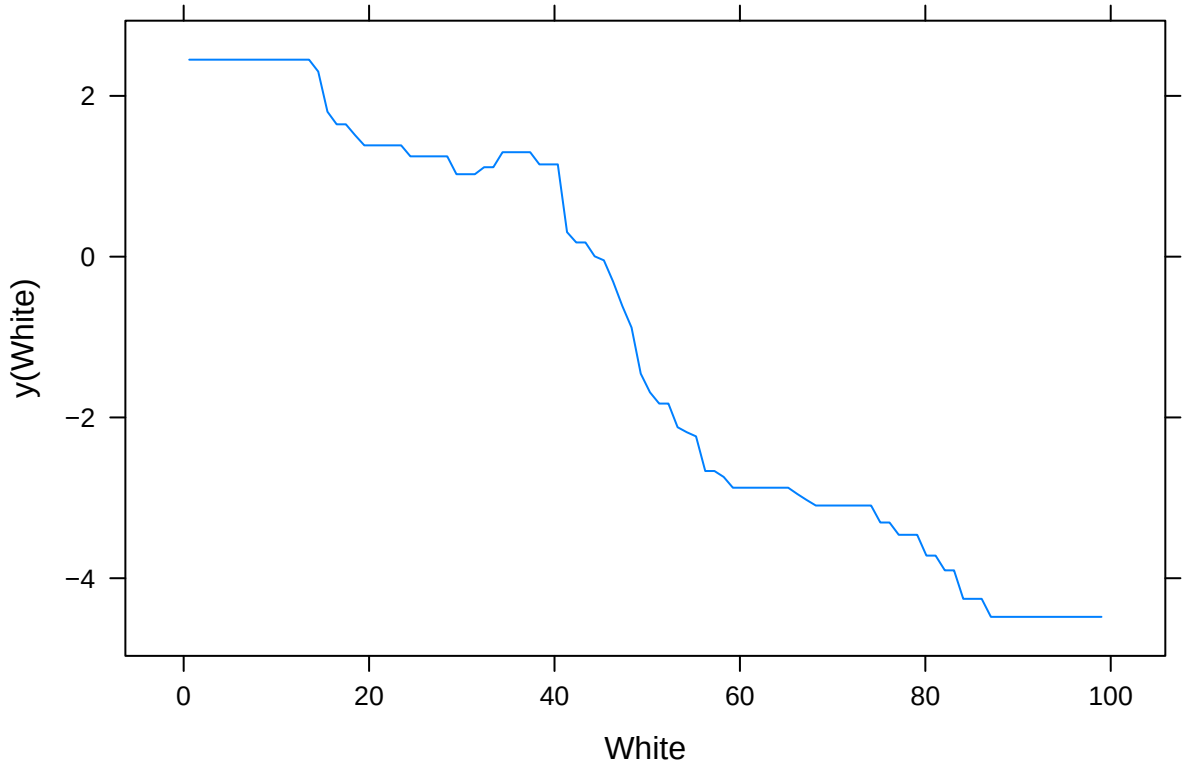
First, lets take a look at **boosting**




```
##                                 var    rel.inf
## Transit                     Transit 23.4486346
## White                         White 19.3497196
## TotalPop                   TotalPop 17.0288377
## Employed                   Employed  5.6368906
## Professional           Professional  5.5210106
## ChildPoverty           ChildPoverty  3.6012230
## VotingAgeCitizen VotingAgeCitizen  3.5637694
## SelfEmployed           SelfEmployed  2.7101961
## OtherTransp             OtherTransp  2.5191407
## Service                     Service  2.2232324
## Production               Production  2.2009432
## Income                       Income  2.0206283
## Men                             Men  1.9333435
```

```
## Unemployment         Unemployment  1.7063023
## Drive                       Drive  1.1757157
## PrivateWork          PrivateWork  1.1341721
## Minority                Minority  1.1041621
## Poverty                  Poverty  0.9591054
## WorkAtHome            WorkAtHome  0.8766881
## Office                    Office  0.4978401
## Carpool                  Carpool  0.3623228
## MeanCommute          MeanCommute  0.2871889
## FamilyWork            FamilyWork  0.1389329
```





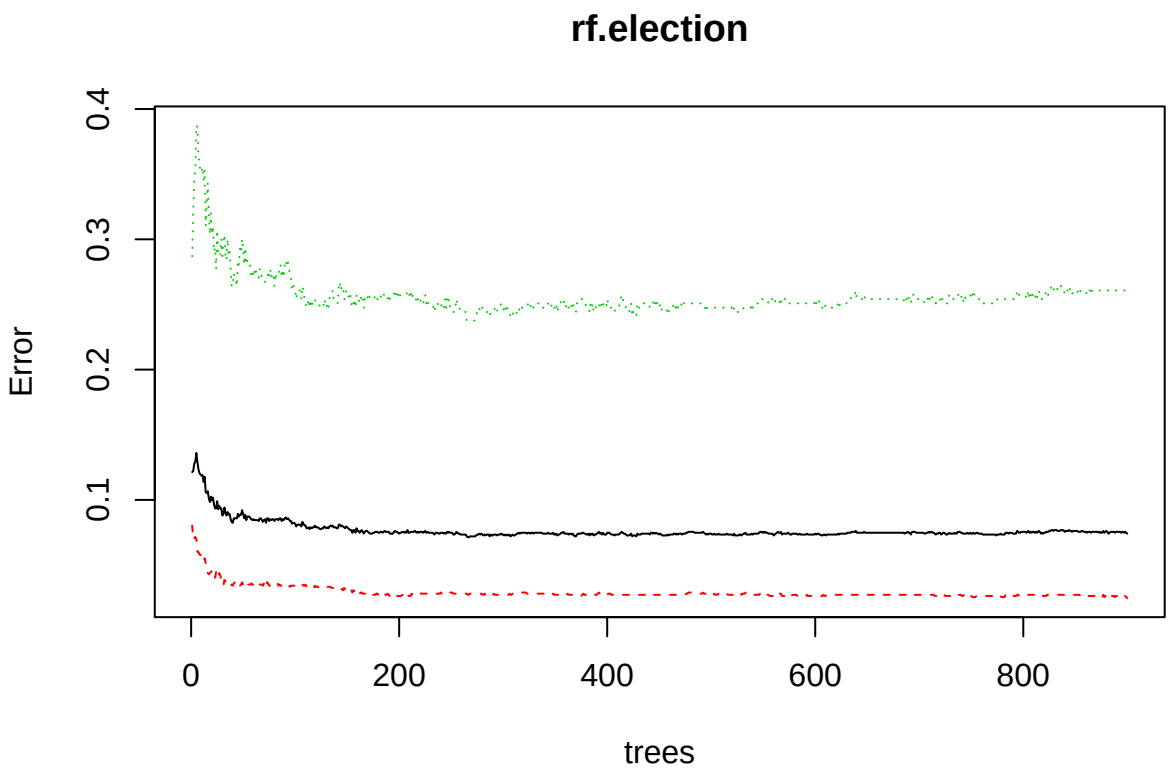
	test.error
boosting	0.9972299

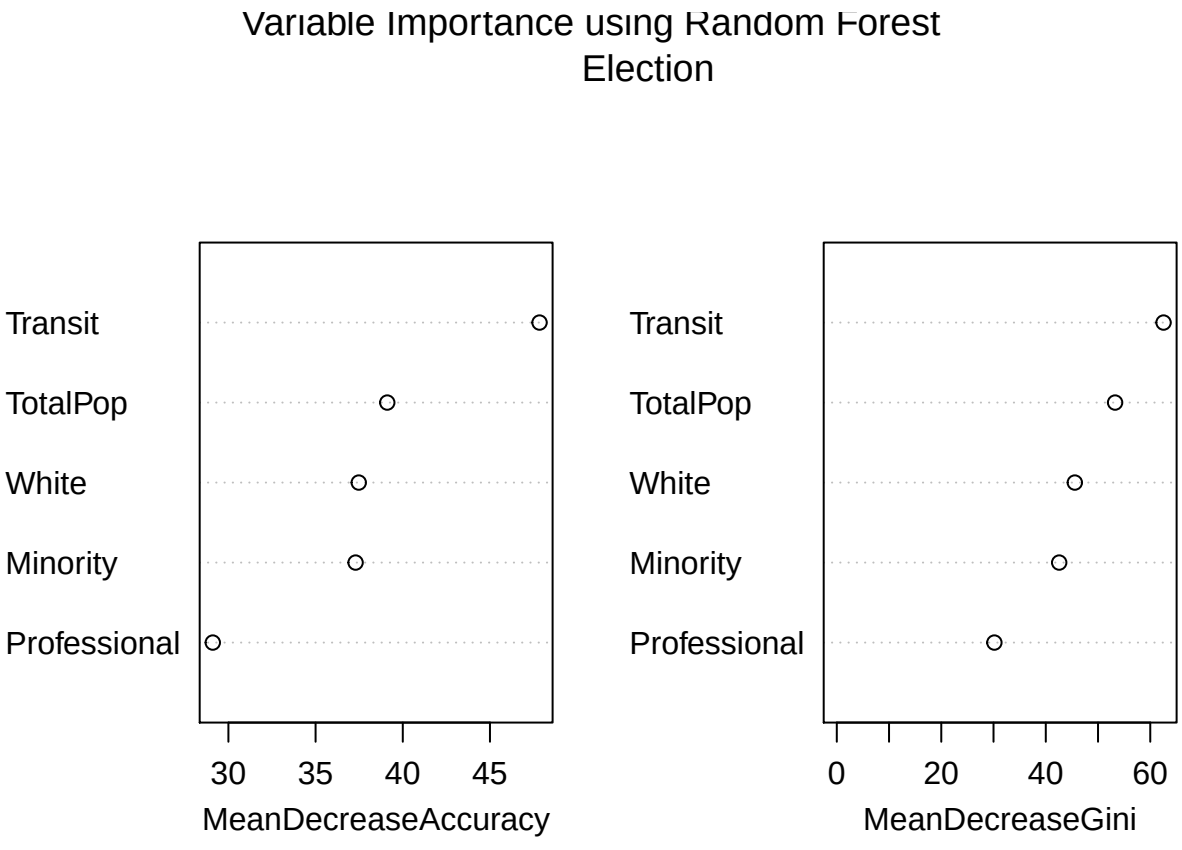
As we see in our work above, the boosting model shows an error of 0.9972299, a very high error. Although intially surprising to see a signifcantly high error, this is likely a result of boosting being an algorithm is more fit for smaller data sets.

Utilizing the summary function, we realize that Transit and White are two variables with high relative influence. This is consistent with some of the other information we have seen previously.

In conclusion, the boosting method is not great at fitting our large data set compared to the logistic regression, LASSO regression and other methods we have explored. The boosting method is likely the least insightful method we have used, as its plots and graphs don't provide us with enough information on how to interpret the importance of these variables to our data and voter behavior.

Next, lets take a look at **random forests**


rf.election
Error
0.1
0.2
0.3
0.4
0
200
400
600
800
trees


Variable Importance using Random Forest
Election
Transit
TotalPop
White
Minority
Professional
30
35
40
45
MeanDecreaseAccuracy
Transit
TotalPop
White
Minority
Professional
0
20
40
60
MeanDecreaseGini

	train.error	test.error
tree	0.0893352	0.1301939
logistic	0.0671745	0.0831025
lasso	0.0713296	0.0858726

	test.error
boosting	0.9972299
random forest	0.0914127

Next, we analyzed our dataset using a random forest model by creating more trees (900 as opposed to 700 in the boosting model). We get an error of 0.0914127, marginally larger than the decision tree, logistic and lasso regression. The random forest analyzes the data decently, especially for the data set being quite large.

Analzying the model, the Variance Importance plot displays that the variables Transit, TotalPop, White, Minority, and Professional play the biggest roles in decreasing the Gini impurity, one of the main goals of this method. This method is consistent with some of the other methods that we have seen, such as the decision tree, which also shows that Transit, White, TotalPop and Professional are key variables.

When analyzing these factors in the context of the 2020 election, we understand that these variables all play significant influence in voter behavior. A voter's racial demographic as well as social-economic status played a role in their choice of presidential candidate.

In conclusion, the random forest tree is an informative method that helps identify strong predictors in the data set despite it being prone to over-fitting similar to the logistical regression model.

From this question, we clearly observe that boosting was the worse method compared to random forest, as the error was nearly 1. The random forest error was more along the lines of the decision tree, logistic, and lasso regression models.

20 - *Tackle at least one more interesting question. Creative and thoughtful analysis will be rewarded!*

In this question, we conduct an exploratory analysis of the "swing counties", counties that models predict Biden and Trump both are equally likely to win. What makes these counties so hard to predict?

There were several main swing counties in this election. Three key swing counties in 2020 were.

1) Maricopa county, Arizona
2) Waukesha county, Wisconsin
3) Lackawanna county, Pennsylvania

Maricopa County

```
## # A tibble: 3 x 5
##   state   county   candidate    party   votes
##   <chr>   <chr>    <fct>        <fct>   <dbl>
## 1 Arizona Maricopa Joe Biden    DEM   1027269
## 2 Arizona Maricopa Donald Trump REP    980494
## 3 Arizona Maricopa Jo Jorgensen LIB     31069
```

```
## # A tibble: 4 x 3
## # Groups:   state [1]
##   state   candidate      votes
##   <chr>   <fct>          <dbl>
## 1 Arizona Donald Trump 1626679
```

```
## 2 Arizona Jo Jorgensen   49984
## 3 Arizona Joe Biden    1643664
## 4 Arizona Write-ins       2208
```

We see above that Biden won Maricopa county by a slim margin, just under 5000 votes. This win in Maricopa county was the main reason Biden won in Arizona, as the margin between Biden and Trump in the entire state was under 3000 votes. The winner of Maricopa nearly always wins Arizona, as the majority of the population in Arizona resides within Maricopa.

Maricopa was a county that was red in 2016 when Trump beat Clinton, and historically has been red for the last 70 years. This county is typically narrowly won, and is a difficult county to effectively predict. Maricopa is typically a county with large communities of white and hispanic voters. Biden won this county largely due to the large support of a majority 56.3% of white voters in Maricopa. Trump, however, kept Maricopa close due to the increase in support amongst hispanic voters, which make up 30.6% of the Maricopa population. Democrats, overall, failed to resonate as much with several Minority demographics, and in the key county of Maricopa, is a huge reason why this county will remain difficult to predict.

Lackawanna County

```
## # A tibble: 4 x 5
##   state        county     candidate    party votes
##   <chr>        <chr>      <fct>        <fct> <dbl>
## 1 Pennsylvania Lackawanna Joe Biden    DEM   61124
## 2 Pennsylvania Lackawanna Donald Trump REP   51501
## 3 Pennsylvania Lackawanna Jo Jorgensen LIB    1070
## 4 Pennsylvania Lackawanna Write-ins    WRI     280
```

```
## # A tibble: 4 x 3
## # Groups:   state [1]
##   state        candidate      votes
##   <chr>        <fct>          <dbl>
## 1 Pennsylvania Donald Trump 3315998
## 2 Pennsylvania Jo Jorgensen   77517
## 3 Pennsylvania Joe Biden    3361668
## 4 Pennsylvania Write-ins       9956
```

Looking specifically at the numbers for this county, we see that Biden won Lackawanna county by around 10000 votes. Compared to the 45000 difference on the state level in Pennsylvania with nearly 7 million votes cast, a difference of 10000 in a single county played a huge role in Biden's victory.

A big reason why Biden won this region was that Lackawanna was home to Scranton, Joe Biden's home town. Lackawanna is made-up of a majority of working-class democrats, a demographic Biden did well with in the polls. Although Trump surprisingly won Pennsylvania in 2016, Biden did enough amongst the 86.6% white individuals out of entire population in Lackawanna. This was a county that was difficult to predict mainly due to Trump's surprising amount of support in Pennsylvania in 2016, a drastic shift from Obama's landslide wins in Lackawanna in 2012 and 2008.

Waukesha County

```
## # A tibble: 6 x 5
##   state     county   candidate       party  votes
##   <chr>     <chr>    <fct>           <fct>  <dbl>
## 1 Wisconsin Waukesha Donald Trump    REP   159633
```

```
## 2 Wisconsin Waukesha Joe Biden       DEM   103867
## 3 Wisconsin Waukesha Jo Jorgensen    LIB     3023
## 4 Wisconsin Waukesha Write-ins       WRI      798
## 5 Wisconsin Waukesha Brian Carroll   ASP      331
## 6 Wisconsin Waukesha Don Blankenship CST      305
```

```
## # A tibble: 6 x 3
## # Groups:   state [1]
##   state     candidate         votes
##   <chr>     <fct>             <dbl>
## 1 Wisconsin Brian Carroll      5417
## 2 Wisconsin Don Blankenship    5206
## 3 Wisconsin Donald Trump    1610030
## 4 Wisconsin Jo Jorgensen      38271
## 5 Wisconsin Joe Biden       1630569
## 6 Wisconsin Write-ins          7980
```

The initial statistical analysis of Waukesha county displays that Trump won by a significant margin in Waukesha, while Biden won Wisconsin by a marginal amount (under 3000 votes).

While Trump's convincing win in Waukesha initially seems insignificant to Biden, Biden massively closed the gap in Waukesha county compared to previous election results of Republican incumbents, especially 2016 against Clinton. Biden's pattern of doing better amongst white voters, as well as performing well with working-class democrats was a key reason why he closed the margin in Waukesha. Waukesha is made up of 89.2% white voters, 13% voters working service jobs, and 45.6% voters working professional jobs.

21 - ***(Open ended) Interpret and discuss any overall insights gained in this analysis and possible explanations. Use any tools at your disposal to make your case: visualize errors on the map, discuss what does/doesn't seems reasonable based on your understanding of these methods, propose possible directions (collecting additional data, domain knowledge, etc).***

An interesting result involved visualizing the county winners on the map of United States. We understand that Biden won the electoral college and popular vote by a decent margin, but when we visualize the county winners, we see that Trump won the majority of the counties. Most of Biden's voters were consolidated on the east and west coast. I believe a more effective visualization of the county winners on the United States map would be to alter the county size on the map be based on their population, relative to the size of the rest of counties. This would be a more effective visualization as opposed to seeing just a mostly red map, which doesn't paint the entire picture.

Initially, I was interested to see that Texas and Florida were two of the 50 states with a high level of minority voter turnout. Trump performed better among hispanic communities than Biden, a key reason why Trump won Texas and the massive swing state of Florida, where several pundits predict Biden would perform better.

One possible direction this project can be taken in is performing more exploratory data analysis on data from the 2016 election. This would be especially interesting to see how Trump did in counties/states that he won in 2016 versus Biden's performance in 2020. In 2016, Trump's election to office was seen as a surprise by millions, and I believe it would interesting to specifically analyze his performance from 2016 to 2020.

The 2020 election also saw a drastic increase in voter turnout, likely due to the influx of mail-in ballots. A comparison with the 2016 election would display which states and counties saw the biggest proportional increase in 2020, and whether this played a major difference in the outcome of the election. Several political pundits predicted mail-in ballots to be primarily democratic, and it would be interesting to analyze this further when compared to 2016.