Case Study 1: Ensemble Methods

Alex Damisch

October 6, 2017

I. Introduction

The results of the United States' 2016 Presidential Election were shocking to much of the world, from pundits to journalists alike.[1] Nationally, it seems that there are widening divisions in the voting preferences of men and women, and those with and without college degrees.[2] Economic decline may have also played a role, especially in the Midwest.[3] Indeed, the widening gyre of voting preferences is particularly evident in the Midwest and South, compared to the traditionally more liberal coasts. Just two states east of Colorado and west of Virginia gave their electoral votes to Hillary Clinton: Minnesota (where I lived in November 2016) and Illinois (where I live now).

States award their electoral college votes in a "winner-takes-all" fashion (with the exception of Maine and Nebraska, who award their votes proportionally). However, some political scientists believe that looking at county-level data can help explain voting preferences that eventually sway states.[3] This is especially an interesting problem for states like Illinois, where one county can essentially decide an entire state's electoral college votes.

The goal of this case study is to classify counties based on whether the majority of their population voted for Donald Trump or Hillary Clinton, based on county-level data on per-capita commercial activity in the county and the demographics of the people living there. I made the commercial data per-capita (and some of the demographic data that was not already) because population would of course inflate those numbers. I also considered it somewhat trivial that counties with higher populations tend to be more liberal. Although this would not necessarily help predict the winner of the general election or even the popular vote, it should give insight to the types of people and the types of places tend to swing right or left.

2. Data Description

The data, called "2012 and 2016 Presidential Elections," was sourced from Kaggle.[4] The data set was compiled by Joel Wilson, a Data Scientist at matchbox.io.[5] The data was itself an agglomeration of two different data sets: A Git repository of election results sourced from *The Guardian* and Townhall[6], and an additional Kaggle data set of county-by-county demographic data, especially from the U.S. Census[7].

As I will detail in Part 3, the data set required *extensive* cleaning and pruning; there were many columns that I removed. As such, the following table of variable descriptions only encompasses those variables which I actually used in the data analysis.

Column Name	Description
Obama	% of 2012 ballots for Barack Obama
Romney	% of 2012 ballots for Mitt Romney
population2014	County population, 2014 estimate
population2010	County population, 2010 estimate
population_change	% change in population, 2010-2014
age5under	% persons under 5 years, 2014
age18under	% persons under 18 years, 2014
age65plus	% persons 65 years and over, 2014
female	% persons female, 2014
White	% persons White alone, 2014
Black	% persons Black or African-American alone, 2014

Column Name	Description
Native	% persons American Indian and Alaska Native alone, 2014
Asian	% persons Asian alone, 2014
PacificIslander	% persons Native Hawaiian and Other Pacific Islander alone, 2014
MultRaces	% persons Two or more races, 2014
Hispanic	% persons Hispanic or Latino, 2014
WhiteNotHispanic	% persons White alone, not Hispanic or Latino, 2014
SameHouse1Year	% persons living in same house 1 year and over, 2009-2013
ForeignBorn	% persons foreign-born, 2009-2013
NonEnglish	% persons $\geq$ 5 yrs, non-English language spoken at home, 2009-2013
Edu_highschool	% persons high school graduate or higher, $\geq$ 25 yrs, 2009-2013
Edu_bachelors	% persons Bachelor's degree or higher, $\geq$ 25 yrs, 2009-2013
Veterans	% persons Veterans, 2009-2013
Commute	Mean travel time to work (minutes), workers $\geq$ 16 yrs, 2009-2013
Homeownership	Homeownership rate, 2009-2013
MultiUnitHouses	% housing units in multi-unit structures, 2009-2013
MedianHomeValue	Median value of owner-occupied housing unites, 2009-2013
PersonsPerHousehold	Persons per household, 2009-2013
Income	Per capita money income, past 12 months (2013 dollars), 2009-2013
HouseholdIncome	Median household income, 2009-2013
Poverty	% persons below poverty level, 2009-2013
NonfarmEsts	Number of private nonfarm establishments per capita, 2013
NonfarmEmployed	% all persons with private nonfarm employment, 2013
NonfarmEmployedPctDel	% change in private nonfarm employment, 2012-2013
NonemployerEsts	Number of nonemployer establishments per capita, 2013
Firms	Number of firms per capita, 2007
BlackOwnedFirms	% of firms Black-owned, 2007
NativeOwnedFirms	% of firms American Indian- and Alaska Native-owned, 2007
AsianOwnedFirms	% of firms Asian-owned, 2007
PIOwnedFirms	% of firms Native Hawaiian- and other Pacicif Islander-owned, 2007
HispanicOwnedFirms	% of firms Hispanic-owned, 2007
WomenOwnedFirms	% of firms woman-owned, 2007
ManufacturerShipments	$ in manufacturer shipments per capita, 2007 (2010 population)
WholesalerSales	$ in wholesaler sales per capita, 2007 (2010 population)
Retail	$ in retail sales per capita, 2007
FoodServices	$ in accommodation and food services sales, 2007 (2010 population)
BuildingPermits	Number of building permits per capita, 2014
Density	Population per square mile, 2010
result2016	Winner of majority votes in county, 2016

There are a couple of potential problems here. First of all, not all of the data come from the same year, and none of the demographic data comes from 2016. Especially for the 2013 data, I think it's reasonable to expect that demographics don't usually shift quickly enough to have a drastic change in 3 years.

The biggest potential bias int he data is that most of the economic indicators come from 2007. 2007 is before the brunt of the Great Recession hit, so some of the numbers about firms and retail sales might actually be much lower than they were in 2007. However, our figures on homeownership rates were during some of the worst parts of the recession, in 2009 and 2010[8], so hopefully this will mitigate the effect of the overly optimistic retail figures.

3. Data Cleaning

As previously mentioned, data cleaning was an arduous task for this data set. Here is a list of columns that I removed and why:

- County FIPS codes (numerous duplicates)–not needed for classification
- Number/proportion of votes for each party, total number of votes in 2016, raw and per-point difference in number of votes in 2016–Essentially the same as predicting who won in 2016
- County name and state abbreviation–I concatenated these and used them as row names instead
- Number of votes for each party, total number of votes in 2016, raw and per-point difference in number of votes in 2016–I retained just the proportion of votes for Obama and Romney in 2012 as predictors, and the rest of these are redundant to that.
- An additional estimate for 2010 population–Redundant
- Number of households, number of housing units–I converted all of the economic and demograhpic data to per-person or percentage numbers (as I will discuss later). I wanted those variables to be proportional to population size. I would have divided out the population for housing units, but there was already a column for "persons per household," so I deleted it.
- Retail sales–Similar to the number of households/housing units logic, there was already a column for "retail sales per capita," so I deleted the column of the raw amount of retail sales.
- Land area in square miles–I used density instead, which I think is a more interesting metric.
- Joel Wilson's fitted predictions and deviations for 2012 and 2016 candidates winning that county–Obviously my goal was to come up with my own predictions!

Before I removed the columns of raw votes, I created a column called `result2016` for whoever won the county. Because of third-party candidates, the sum of Trump and Clinton's votes did not equal 1 for any county; however, there was no county where neither Trump nor Clinton captured over 50% of the votes.

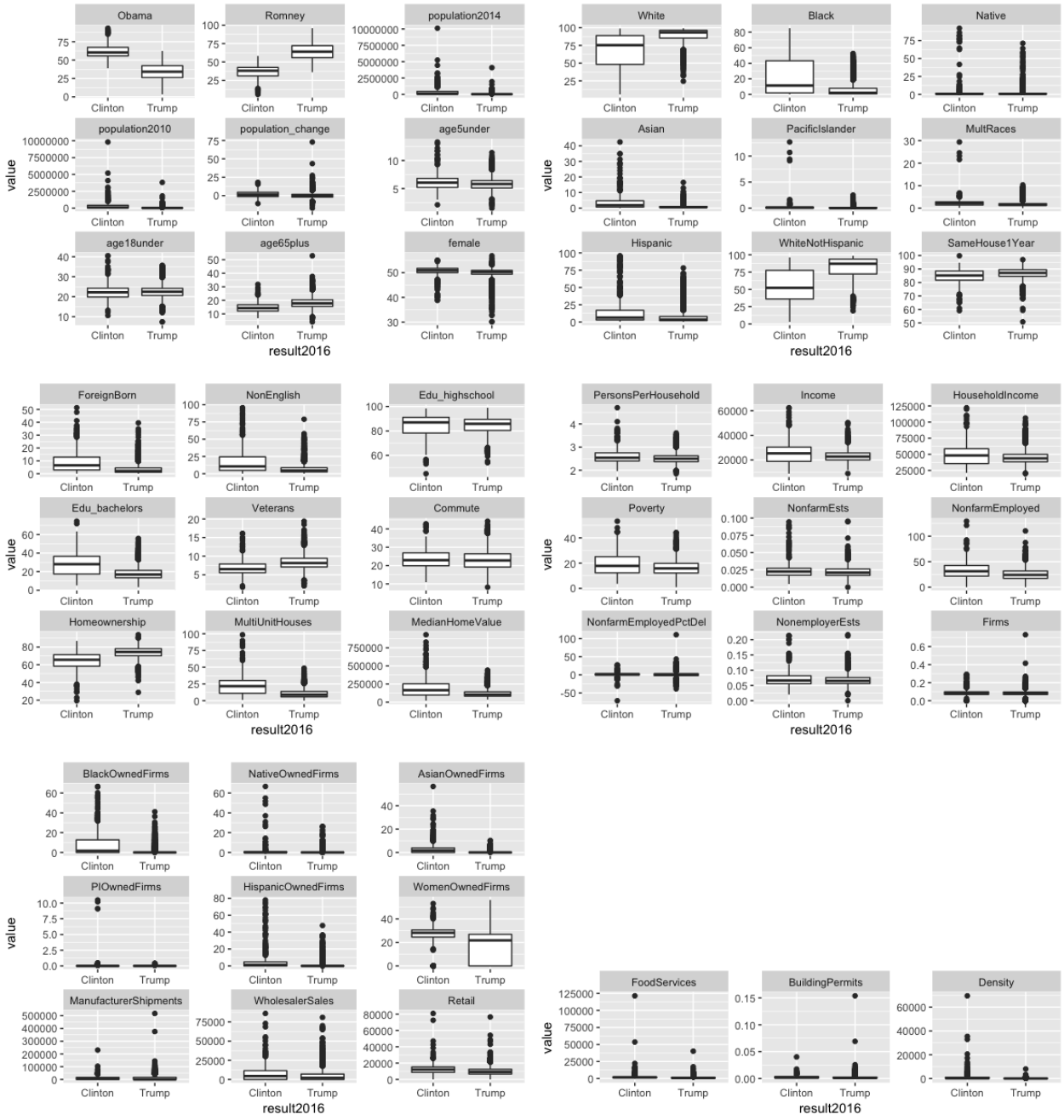
In the raw data set, some of the per-capita/proportional data were in percentage points out of 100; some were proportions out of 1. I changed it to the former. I also made the number of veterans and number of people working in nonfarm employment sectors into percentages of the total population. To my knowledge, the latter is not a widely-used measure of unemployment, since we are not accounting for people who are not seeking work, minors, the elderly, etc. However, for comparison it should hopefully provide some idea about how employment levels vary from county to county. Because the number of veterans was a 2009-2013 estimate, I divided by the 2010 population. Because the nonfarm private employment was a 2013 estimate, I divied by the 2014 population.

I scaled the numbers of nonfarm establishments, nonemployer establishments, and total number of firms, manufactures shipments, merchant wholesaler sales, accommodation and food services sales, and building permits to per-person numbers. When applicable, I first multiplied them by $1,000 (some commerce metrics were originally in $1,00s of dollars.) Again, I wanted the levels of commerce to be relative because I wanted them to be proportional for each county–obviously counties with higher populations would have larger raw sales. Because the commerce figures were cited for 2007, I divided by the 2010 population estimate.

4. Data Analysis

In this data set, there were 488 counties that voted for Clinton, and 2,624 counties that voted for Trump, for a total of 3112 counties. (For the rest of the paper, I will use "voted for" a certain candidate to indicate that the majority of a county's votes were for that candidate, and that the county belongs to that class.) Clinton won the popular vote, with 62,298,328 votes to Trump's 60,995,541 votes.

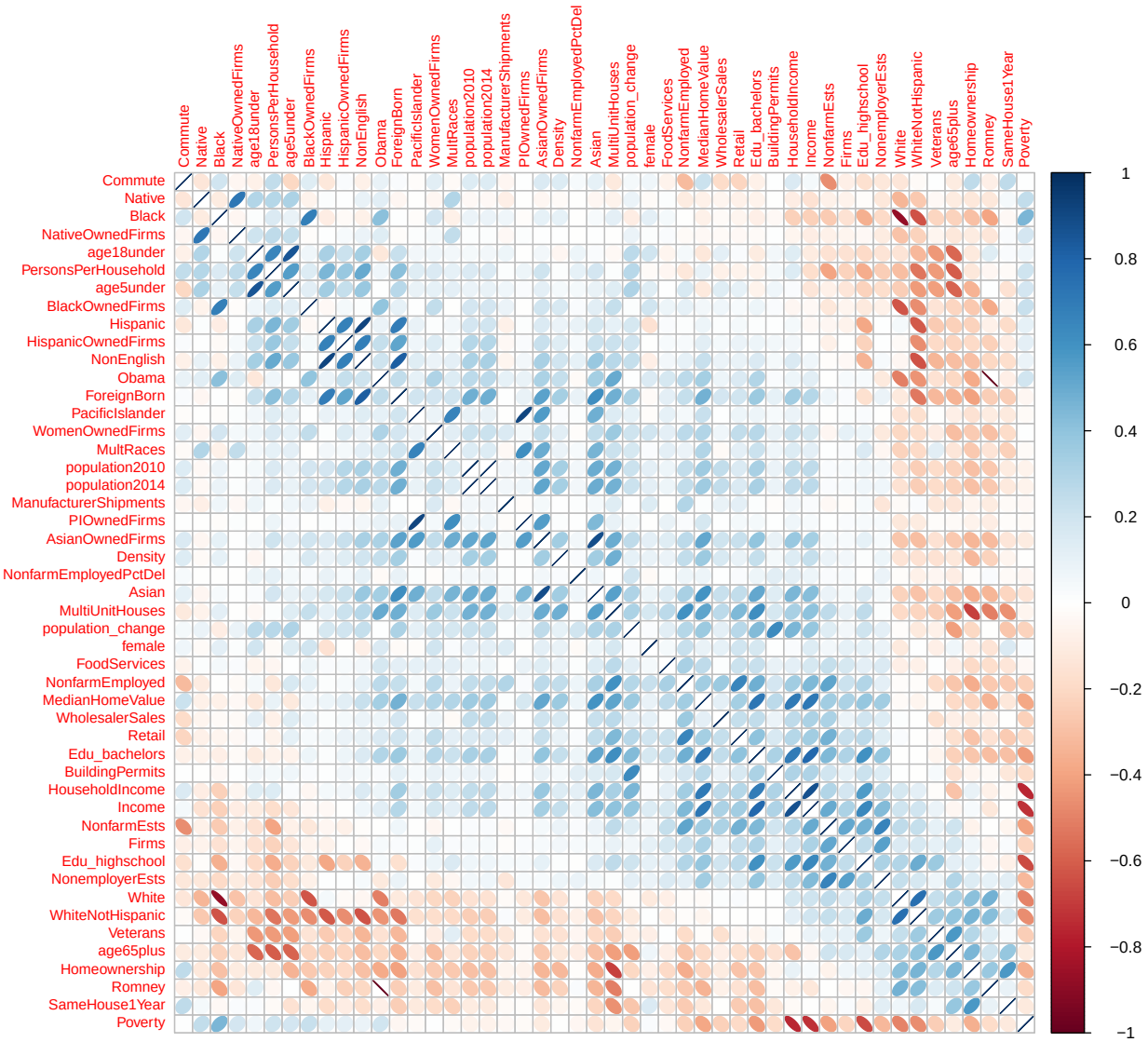
Because all of my variables were numeric other than the class, I used box plots to compare the distributions of the numeric data between counties that voted for Clinton and those that voted for Trump.



Reporting all of the summary statistics seems a little silly, so instead I will discuss some variables where there is a marked difference in "Clinton counties" versus "Trump counties." The first two boxplots are basically self-evident–By and large, most counties vote fairly consistently for one party year-to-year: Clinton's counties had much higher proportions of votes for Obama in 2012, and the same for Trump's counties for Romney in 2012. In the first row, it looks like Trump's counties were slightly older on average (`age65plus`). The counties that Clinton won were strikingly less White and more Black than those that Trump won–note how that in the graph for `Black`, the median for Clinton's counties is above the 4th quarter for Trump's. The counties that Clinton won seemed more diverse across the board: Clinton won counties with higher percentages of of `PacificIslander` and `MultRaces`. What shocks me the most in the demograhic data, actually, is that Trump won several counties with upwards of 50% of people identifying as `Hispanic`–most of them in extreme southern Texas.

In the second row, we see that Clinton counties also, on average, have higher population of people who are `ForeignBorn` and `NonEnglish` speakers at home, but more likely to have a Bachelor's degree. Clinton counties also have lower rates of `Homeownership` and more `MultiUnitHouses` per capita, validating my assumption in the Introduction that cities were more liberal. However, the `MedianHomeValue` was also higher on average in Clinton counties, so those counties had fewer homeowners but with more valuable homes.

Many of the economic indicators look the same for Clinton and Trump counties, in my opinion with two exceptions. One is `BlackOwnedFirms`, which is not terribly surprising considering that Clinton's counties had higher percentages of Black people in the first place. The other exception is `WomenOwned Firms`. Now, it is worth noting that some counties had a bizarre dearth of women–the highest percentage of women in any county was 56.8%, but the lowest percentage of women in any county was 30.1% (!). I did a little research, and some of those counties with very few women had military facilities, some had large prisons, some had large plants or heavy industry that presumably employ nearly all men. In any case, if you don't have any women in your county, it's hard for them to own businesses.



Even after careful, eye-straining study, the correlations among the explanatory variables are not particularly useful here. We find out that greater numbers of a certain minority mean that they tend to own more businesses there. Counties that have a higher `HouseholdIncome` have a higher `MedianHomeValue`, and lower rates of `Poverty`. If we were trying to model this based on logistic regression or anything linear, we would be in huge trouble because of multicollinearity; however, using classification methods should produce the desired results. How a county voted in 2012 and its racial makeup will definitely be important predictors for how it voted in 2016.

5. Experimental Results

Based on the structure of my data, I chose Naive Bayes and decision treees for the individual classificaiton techniques. These were the best methods to choose because the data was already labelled as being "Trump" or "Clinton" counties, and I wanted to usesupervised classificationmethods. I went through several steps of parameter tuning, which I will describe in the experimental analysis method. When coming up with trees, I used 10-fold cross-validation. My final tree split based on information gain with a minimum node size of 10 cases, and used post-pruning with a complexity parameter of 0.01. For my ensemble classification method, I used random forests. I will discuss the parameter tuning in the experimental analysis section.

When comparing the 3 methods, I used an F-test. For TP=True Positive (the number of Clinton counties correctly characterized as such) and FP = False Positive (the number of Trump counties incorrectly characterized as Clinton counties),

$$F = 2 * \frac{TP * (TP/(TP+FP))}{TP + (TP/(TP+FP))}$$

Note that $(TP/(TP+FP))$ is the "precision." The naive Bayes classifier had an F-score of 1.58, the decision tree had an F-score of 1.67, and random forests had an F-score of 1.92. So, overall I preferred the random forest method for its classification accuracy. I especially had confidence in the random forest method because it seems extremely well-suited to a binary classification problem where we have so many different

6. Experimental Analysis

Naive Bayes

The first method that I used was naive Bayes. Because naive Bayes doesn't really involve any parameter tuning **like the other two methods**, I used 10-fold cross-validation with a testing and validation set. The training set had 2801 rows and the test set had 311 rows. Here was the final confusion matrix for the naive Bayes classifier:

Here is a selection of the mean and standard deviation for the naive Bayes classifier. I only included the variables that were on a 0-100% scale, because the weights vary a lot more for the other variables:

```
## $age5under
##          age5under
## Y             [,1]     [,2]
##   Clinton 6.125977 1.459275
##   Trump   5.842984 1.117930
##
## $age18under
##          age18under
## Y             [,1]     [,2]
##   Clinton 22.34598 4.047961
##   Trump   22.58445 3.180578
##
```

```
## $age65plus
##          age65plus
## Y             [,1]     [,2]
##   Clinton 14.78000 3.708334
##   Trump   18.13567 4.237411
##
## $female
##          female
## Y             [,1]     [,2]
##   Clinton 50.55126 2.293603
##   Trump   49.85101 2.172207
##
## $White
##          White
## Y             [,1]     [,2]
##   Clinton 68.65195 23.51928
##   Trump   88.53352 11.45335
##
## $Black
##          Black
## Y              [,1]     [,2]
##   Clinton 22.047126 23.91756
##   Trump    6.898267 10.36992
##
## $Native
##          Native
## Y            [,1]      [,2]
##   Clinton 3.09931 11.798402
##   Trump   1.80503  5.167499
##
## $Asian
##          Asian
## Y              [,1]     [,2]
##   Clinton 3.7103448 5.147673
##   Trump   0.9090871 1.053939
##
## $PacificIslander
##          PacificIslander
## Y               [,1]      [,2]
##   Clinton 0.18781609 0.7726092
##   Trump   0.08064243 0.1482718
##
## $MultRaces
##          MultRaces
## Y             [,1]     [,2]
##   Clinton 2.302759 1.916769
##   Trump   1.764877 1.069301
##
## $Hispanic
##          Hispanic
## Y              [,1]     [,2]
##   Clinton 15.767816 21.43343
##   Trump    7.829079 11.21530
##
```

```
## $WhiteNotHispanic
##          WhiteNotHispanic
## Y             [,1]     [,2]
##   Clinton 54.73770 24.78042
##   Trump   81.56412 15.28900
##
## $SameHouse1Year
##          SameHouse1Year
## Y             [,1]     [,2]
##   Clinton 84.64437 5.408926
##   Trump   86.74793 4.082587
##
## $ForeignBorn
##          ForeignBorn
## Y             [,1]     [,2]
##   Clinton 9.407126 8.877171
##   Trump   3.577853 4.038278
##
## $NonEnglish
##          NonEnglish
## Y              [,1]      [,2]
##   Clinton 17.928276 18.780018
##   Trump    7.548774  8.704098
##
## $Edu_highschool
##          Edu_highschool
## Y             [,1]     [,2]
##   Clinton 84.21908 8.931025
##   Trump   84.53779 6.535318
##
## $Edu_bachelors
##          Edu_bachelors
## Y             [,1]      [,2]
##   Clinton 27.91103 12.863828
##   Trump   18.25735  6.867614
##
## $Veterans
##          Veterans
## Y             [,1]     [,2]
##   Clinton 6.872245 2.272792
##   Trump   8.360784 2.043572
```

Of the data we see above that was normalized to a percentage basis, it looked like `WhiteNotHispanic`, `Edu_highschool`, and `Homeownership` all had relatively high means for both candidates, so we know they were influential to the naive Bayes classifier.

This is how the naive Bayes classifier performed on the test set:

```
##
## pred      Clinton Trump
##   Clinton      38     9
##   Trump        15   249
```

Here are some of the relevant summary statistics:

```
##       Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
```

```
##   9.228296e-01   7.142201e-01   8.873513e-01   9.499316e-01   8.295820e-01
## AccuracyPValue  McnemarPValue
##   1.332887e-06   3.074342e-01
```

```
##          Sensitivity          Specificity       Pos Pred Value
##            0.7169811            0.9651163            0.8085106
##       Neg Pred Value            Precision               Recall
##            0.9431818            0.8085106            0.7169811
##                   F1           Prevalence       Detection Rate
##            0.7600000            0.1704180            0.1221865
## Detection Prevalence    Balanced Accuracy
##            0.1511254            0.8410487
```

Overall, the naive Bayes classifier had an estimated accuracy of 92.2%, so it did pretty well. With 95% confidence, we can say that its true accuracy lies between 88.74% and 94.99%.

This classifier used Clinton as the "positive" class. This classifier was weakest in its sensitivity, or its ability to correctly identify Clinton's counties as such. It classified well over a quarter of Clinton's counties as Trump counties. On the other hand, the model did much better at specificity–under 4% of Trump's counties were incorrectly predicted to be Clinton's counties. The results of the Bayesian classifier definitely surprised me, especially in the "direction." Most people were overly optimistic about Clinton's chances for the election, and the Bayesian classifier was overly pessimistic.

Decision Trees

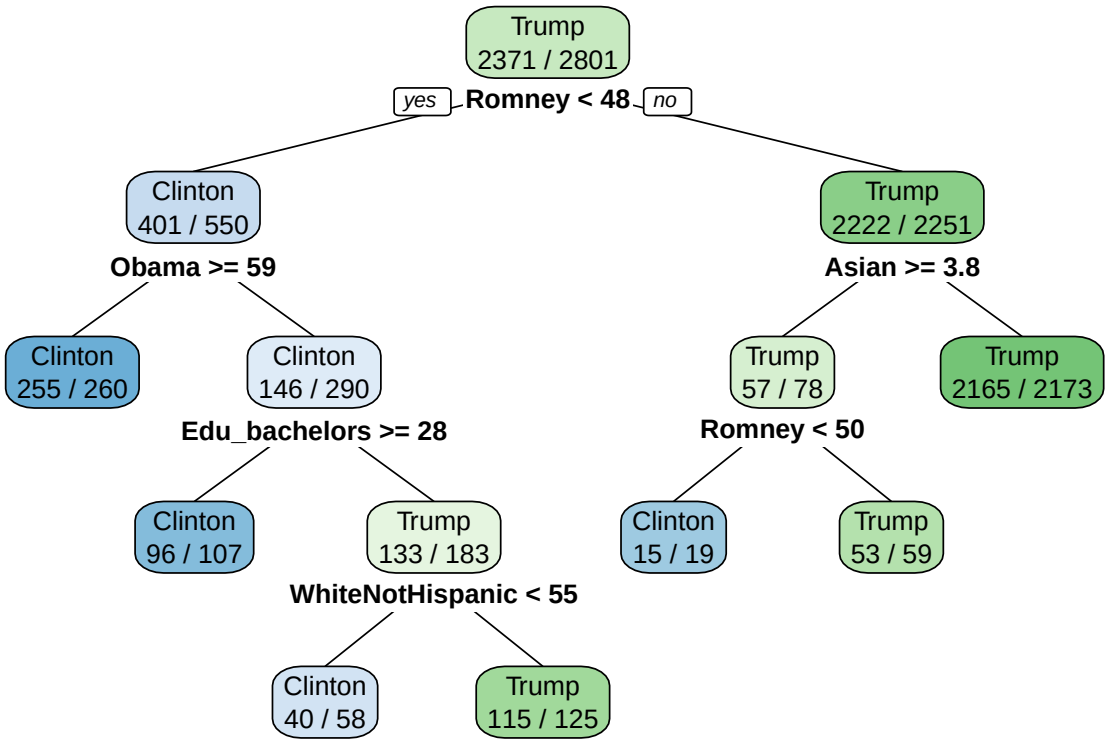
The next method that I tried was decision trees. I used the `rpart` package to build my trees. I split the data into 80% for training, 10% for testing, and 10% for evaluation.

These are the outcomes for 8 different models on the testing sets:

Model	Min. Split	Split	Post-Prune? Cp?	Accuracy CI	Sensitivity	Specificity
1	20	GINI	N/A	(0.9178, 0.9703)	0.8103	0.9802
2	20	GINI	Yes, 0.01	(0.9268, 0.9895)	0.83877	1.0000
3	20	Info	N/A	(0.9648, 0.9998)	0.9677	1.0000
4	20	Info	Yes, 0.01	(0.9648, 0.9998)	0.9677	1.0000
5	10	GINI	N/A	(0.9448, 0.996)	0.9032	1.0000
6	10	GINI	Yes, 0.01	(0.9448, 0.996)	0.9032	1.0000
7	10	Info	N/A	(0.9648, 0.9998)	0.9677	1.0000
8	10	Info	Yes, 0.01	(0.9648, 0.9998)	0.9677	1.0000

Based on these numbers, and because information gain and post-pruning seemed to generally give better results, I decided to use model 8 on the validation data.

As was fairly predictable, the results of the 2012 election, race, and education all had major parts to play in how a county voted.



Here is how model 8 performed on the validation data:

```
## Confusion Matrix and Statistics
##
##
## pred      Clinton Trump
##   Clinton      20     3
##   Trump         7   125
##
##                Accuracy : 0.9355
##                  95% CI : (0.8846, 0.9686)
##     No Information Rate : 0.8258
##     P-Value [Acc > NIR] : 5.417e-05
##
##                   Kappa : 0.7618
##  Mcnemar's Test P-Value : 0.3428
##
##             Sensitivity : 0.7407
##             Specificity : 0.9766
##          Pos Pred Value : 0.8696
##          Neg Pred Value : 0.9470
##              Prevalence : 0.1742
##          Detection Rate : 0.1290
##    Detection Prevalence : 0.1484
##       Balanced Accuracy : 0.8587
##
##        'Positive' Class : Clinton
```

```
##
```

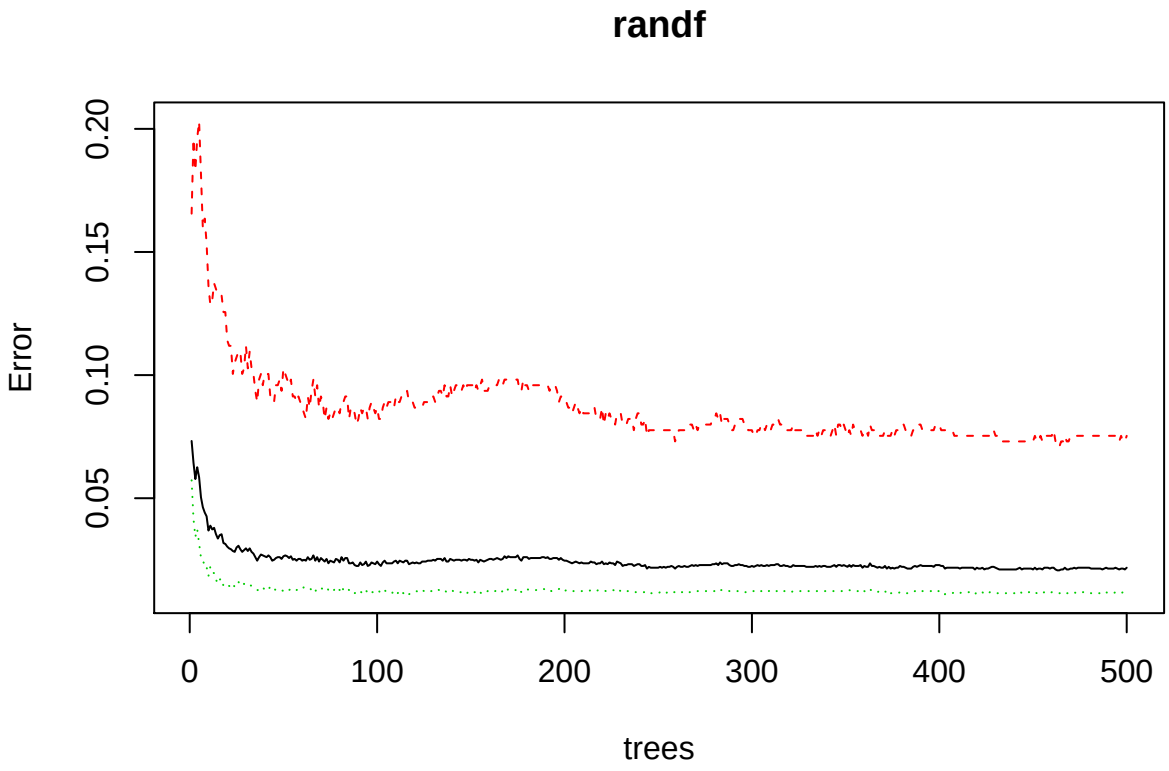
Overall, this model underperformed a little compared to its performance on the test data, especially with respect to sensitivity. The sensitivity was not quite as poor as what we saw with naive Bayes, but the test and evaluation sets just shook out a bit differently than the model expected. But we can still say with 95% confidence that its true accuracy is between 88.46% and 96.86%.

Random Forest

Finally, I used random forests for my ensemble classifiers. I simply used a 10 to 1 testing and training set and looked at the class predictions for the training set.

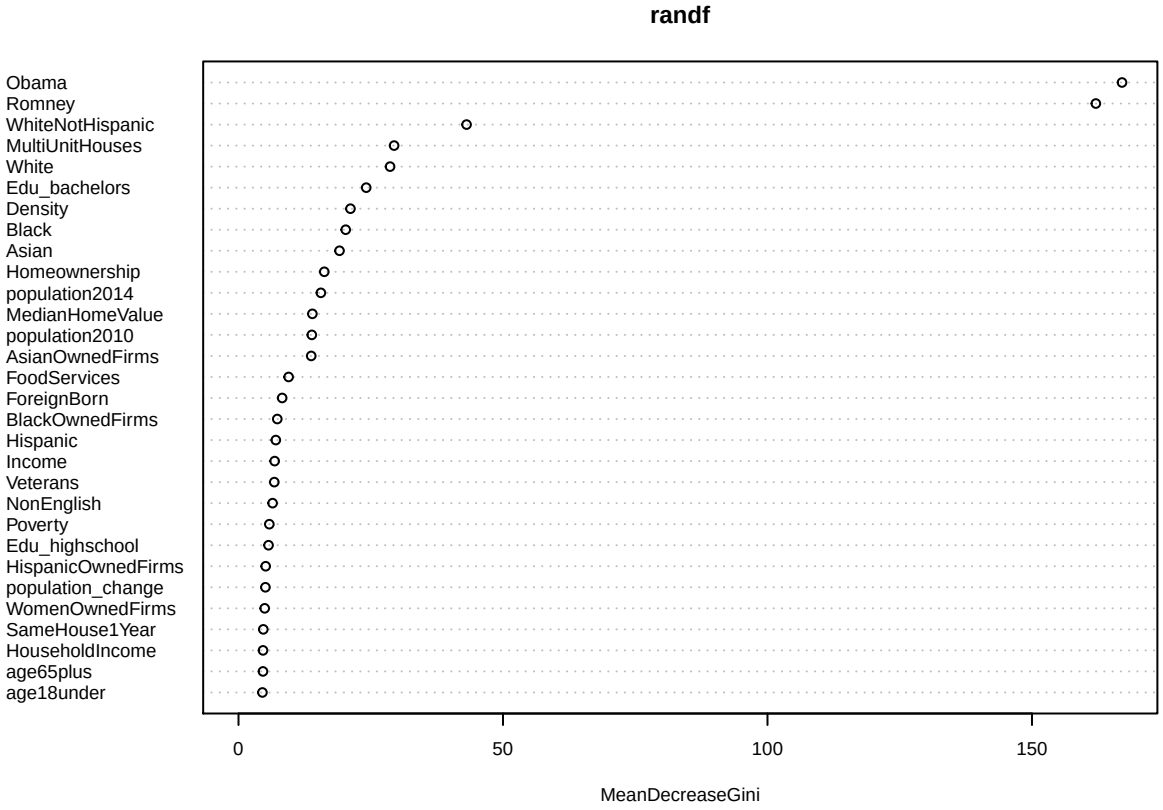
```
## Confusion Matrix and Statistics
##
##
## pred      Clinton Trump
##   Clinton      46     1
##   Trump         4   260
##
##                Accuracy : 0.9839
##                  95% CI : (0.9629, 0.9948)
##     No Information Rate : 0.8392
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.9389
##  Mcnemar's Test P-Value : 0.3711
##
##             Sensitivity : 0.9200
##             Specificity : 0.9962
##          Pos Pred Value : 0.9787
##          Neg Pred Value : 0.9848
##              Prevalence : 0.1608
##          Detection Rate : 0.1479
##    Detection Prevalence : 0.1511
##       Balanced Accuracy : 0.9581
##
##        'Positive' Class : Clinton
##
```

Random forest was by far the best method yet; even basically using the defaults for `randomForest` achieved a sensitivity of 0.9200 and a specificity of .9962, which seemed very balanced compared to the other methods I had tried so far. I am 95% confident that the true accuracy of this model was between 96.29% and 99.48%.



The error rate for random forests declined sharply after the first 20 or 30 trees.

Because random forests can seem kind of opaque, the variable importance plot can help us understand what is going on "underneath":



As we can see, the winner of the 2012 election plays a huge part in the random forest, just like it did with our more standard decision tree. I was really interested to see how `MultiUnitHouses` and `Density` also contributed to the random forest.

6. Conclusion

Overall, I didn't get any radical new insight from the demographic data (other than being amazed at the power of random forests)–most of the results were fairly common-sense, which I suppose demonstrates that the models didn't go haywire anywhere. Given more time, I would have liked to tried tweaking the complexity parameter for the decision trees a little more. I would also have liked to search for more current economic data for the different counties, although it is possible that the effects of the recession are still mostly captured in the data above.

What I actually found most surprising was how much my classifiers tended to often overestime *Trump's* chances, and not Clinton's. This really goes against the prevailing media narrative in the lead-up to the 2016 election.

Appendix

A. Citations

[1] Arrieta-Kenna, R. (2016), "The Worst Political Predictions of 2016," *Politico Magazine* [online]. Available at http://www.politico.com/magazine/story/2016/12/the-worst-political-predictions-of-2016-214555.

[2] Tyson, A. and Maniam, S. (2016), "Behind Trump's victory: Divisions by race, gender, education," *Pew Research Center* [online]. Available at http://www.pewresearch.org/fact-tank/2016/11/09/behind-trumps-victory-divisions-by-race-gender-education/.

[3] Collingwood, L. (2016), "The county-by-county data on Trump voters shows why he won," *The Washington Post* [online]. Available at https://www.washingtonpost.com/news/monkey-cage/wp/2016/11/19/the-country-by-county-data-on-trump-voters-shows-why-he-won/.

[4] Wilson, J. (2016), "2012 and 2016 Presidential Elections," *Kaggle* [online]. Available at https://www.kaggle.com/joelwilson/2012-2016-presidential-elections.

[5] Wilson, J. (2017), "Joel Wilson," *Kaggle* [online]. Available at https://www.kaggle.com/joelwilson}.

[6] McGovern, T. (2016), "County-Level Presidential General Election Results for 2012 - 2016," *GitHub* [online]. Available at https://github.com/tonmcg/County_Level_Election_Results_12-16.

[7] Hammer, B. (2016), "2016 US Election," *Kaggle* [online]. Available at https://www.kaggle.com/benhamner/2016-us-election/data.

[8] Bukszpan, D. (2012), "Industries Hit Hardest by the Recession," *CNBC* [online]. Available at https://www.cnbc.com/2012/06/01/Industries-Hit-Hardest-by-the-Recession.html.

B. Code

Data Cleaning

```
library(readr)
votes <- read.csv("~/Documents/School/2017-2018/CSC 529/votes.csv")
rownames(votes) <- paste(votes$county_name, votes$state_abbr)
any(votes$Clinton <0.5 && votes$Trump < .5) # FALSE
# In every county, either Clinton or Trump won the majority of the votes
votes$result2016 <- ifelse(votes$Clinton > votes$Trump, "Clinton", "Trump")
votes$result2016 <- factor(votes$result2016)
votes <- votes[,c(19:20, 26:28, 30:48,50:52,54:70,72:74,76,83)]
# This keeps the columns we want and deletes the oens we don't
votes <- data.frame(votes)
# Again, the column names were just atrocious when I first got this data set
colnames(votes)[6:7]<- c("age5under", "age18under")
colnames(votes)[9] <- "female"
colnames(votes)[12:15] <- c("Native","Asian", "PacificIslander","MultRaces")
colnames(votes)[17:19] <- c("WhiteNotHispanic","SameHouse1Year","ForeignBorn")
colnames(votes)[22:28] <- c("Edu_bachelors","Veterans","Commute","Homeownership","MultiUnitHouses",
                            "MedianHomeValue","PersonsPerHousehold")
colnames(votes)[30] <- "HouseholdIncome"
colnames(votes)[32:47] <- c("NonfarmEsts","NonfarmEmployed","NonfarmEmployedPctDel",
                    "NonemployerEsts","Firms","BlackOwnedFirms","NativeOwnedFirms",
                    "AsianOwnedFirms","PIOwnedFirms","HispanicOwnedFirms",
                    "WomenOwnedFirms",
                    "ManufacturerShipments","WholesalerSales","Retail",
                    "FoodServices","BuildingPermits")
votes[,c(1:2, 10:11, 16)] <- votes[,c(1:2, 10:11, 16)]*100
votes$Veterans <- (votes$Veterans/votes$population2010)*100
votes[,c(32,35:36,47)] <- votes[,c(32,35:36,47)]/votes$population2014
votes[,c(33)] <- (votes[,c(33)]/votes$population2014)*(100)
votes[,c(43,44,46)] <- (votes[,c(43,44,46)]*1000)/votes$population2010
```

Data Analysis

```
library(reshape)
library(ggforce)
molten <- votes
molten$id <- rownames(votes)
molten <- melt(molten, id.vars=c("id","result2016"), measure.vars=1:48)
n_pages <- ceiling(
  length(levels(molten$variable)) / 9
)

# Draw each page
for (i in seq_len(n_pages)) {
    print(ggplot(molten) + geom_boxplot(aes(x=result2016,y=value))+
        facet_wrap_paginate(~variable, ncol = 3, nrow = 3, scales="free",page = i))
}
# I saved each image separately and then added them in Markdown
```

Experimental Results

```
f <- function(tp, fp){2*(tp*(tp/(tp+fp)))/(tp+(tp/(tp+fp)))}
f(38,9) # Naive Bayes
f(20,3) # Decision Trees
f(46,1) # Random Forest
```

Experimental Analysis

```
set.seed(9999) #set.seed was how I was able to get consistent results
# So I didn't have to change the numbers every time I re-knit!
library(caret)
library(rpart)
library(rpart.plot)
n <- round(nrow(votes)/10)
index <- sample(1:nrow(votes), n)
train <- votes[-index,]
test <- votes[index,]
eval <- test[1:155,]
test <- test[156:311,]
library(tree)
# Model 1
model1 <- rpart(result2016~., data=train)
rpart.plot(model1, type=2, extra=2, fallen.leaves=FALSE)
pred <- predict(model1, test, type = "class")
t <- table(pred,test$result2016)
confusionMatrix(t)
# Follow the same rpart.plot...confusionMatrix(t) for the rest of the models
# Model 2
model2 <- rpart(result2016~., data=train)
model2 <- prune(model2, cp=0.01)
# Model 3
model3 <- rpart(result2016~., data=train, parms=list(split="information"))
# Model 4
model4 <- rpart(result2016~., data=train, parms=list(split="information"))
model4 <- prune(model4, cp=0.01)
```

```r
# Model 5
model5 <- rpart(result2016~., data=train, control=rpart.control(minsplit=10))
# Model 6
model6 <- rpart(result2016~., data=train,control=rpart.control(minsplit=10))
model6 <- prune(model6, cp=0.01)
# Model 7
model7 <- rpart(result2016~., data=train,
parms=list(split="information"),control=rpart.control(minsplit=10))
# Model 8
model8 <- rpart(result2016~., data=train,
parms=list(split="information"),control=rpart.control(minsplit=10))
model8 <- prune(model8, cp=0.01)
```