US County Election Data

Andrew Voorhees, Ruchita Patel, Cotlon Stapper

Introduction

The 2016 election proved to be a nightmare for the predictive analytics community. After significant success in both the 2008 and 2012 elections, expectations were high for this election. But with almost nobody successfully predicting the outcome this time, that confidence now seems misplaced. For this project we explored economic and demographic data for each county and examined how those factors appeared to affect the county's voting pattern. We were a little daunted by the task of determining how teams of highly paid political analysts and statisticians could have been so wrong, so at the advice of Dr. Sun, we focused instead on what interesting patterns and trends we could find when clustering the data. Predicting elections is tricky business, but we hope that our report helps shed some light on how counties and states group together when voting and what factors influence the formation of those groups.

Data Collection

Data for the project came from a variety of sources, most of them taken from GitHub, where good samaritans had scraped election data from reputable sources. Voting data is recorded for each county, so we looked for data sources that had county level granularity. Our first source came from the United States Federal Census Bureau which released demographic data from their 2010-2012 census as well as their modeled estimates for through 2015. The data contained information on the ethnic and gender breakdown of a county, as well as how that information had changed across time.

In addition to our complete list of counties and demographic information, we needed voting results for both the 2012 and 2016 elections. These were found on GitHub and had originally been scraped from The Guardian and Townhall.com. These were easily joined to our existing data based on county code and year. We chose to consider the 2015 estimates as the demographic values for the 2016 election because that was our best approximation of the data. Unfortunately, Alaska did not release county level voting information for 2016. Because of its relatively small population and few electoral college votes, we felt that it was acceptable to remove Alaska prior to doing analysis. One small county in South Dakota and another in Hawaii were also thrown out due to problems with the data.

Finally we looked for additional features we could use for analyzing the counties. Google came to the rescue again, and we found county level economic information with very complete feature sets. The features we chose to include in our analysis were:

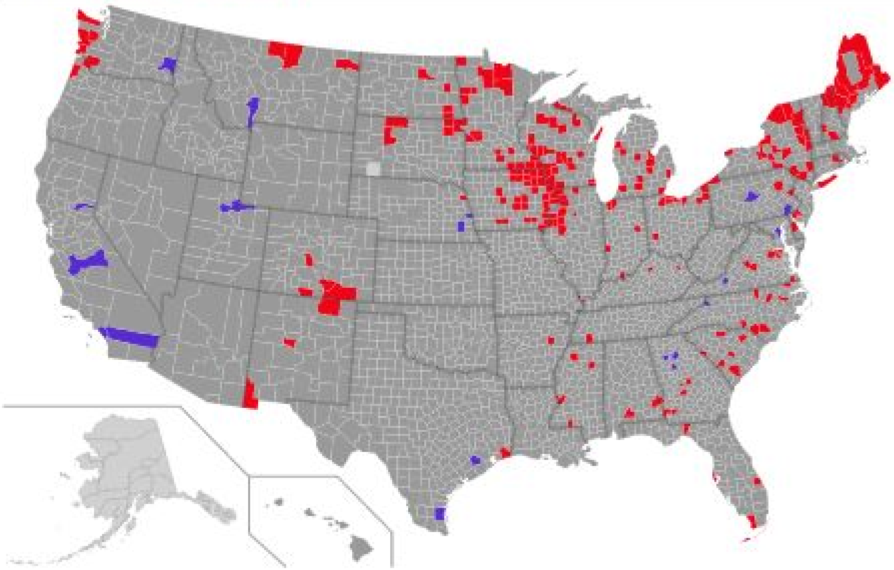
```
'black firms',
 'land area',
 '>= bachelor',
 'hawaiian firms',
```

'total firms',
'native american firms',
'pop density',
'non_english',
'same house > 1 yr',
'veterans',
'multiunit housing',
'asian firms',
'fips',
'home value',
'nonfarm empl % change',
'homeownership rate',
'hispanic firms',
'nonfarm homes',
'>= high school',
'retail sales',
'service sector sales',
'age > 65',
'merchant wholesaler sales',
'avg commute',
'nonemployer establishments',
'retail sales per capita',
'foreign_born',
'peeps per house',
'households',
'age < 18',
'building permits',
'nonfarm empl',
'manufacturing shipments',
'income per cap',
'below poverty',
'age < 5',
'housing units',
'women firms',
'household income'

We did not have each of these features by year, so we chose to use the same values for our 2012 and 2016 data points. With a complete data set, we moved on to the exploration and modeling.

Exploring County Election Results

When we first started looking at the data, our main focus was to ask which party would a county vote for given the features mentioned above. In doing this, we wanted to take a look at the results of the elections of 2012 and 2016 to help us gather information about counties that made a significant impact from 2012 to 2016. It was originally hard to tell in a map where all the counties were filled based on voting density (percentage of votes that went to each county with their respective color), since there were not that many counties that had changed. The figure below shows the 238 counties that changed from the 2012 to 2016 election, the color represents which party the county changed to in 2016. (Red for republican and blue for democratic)

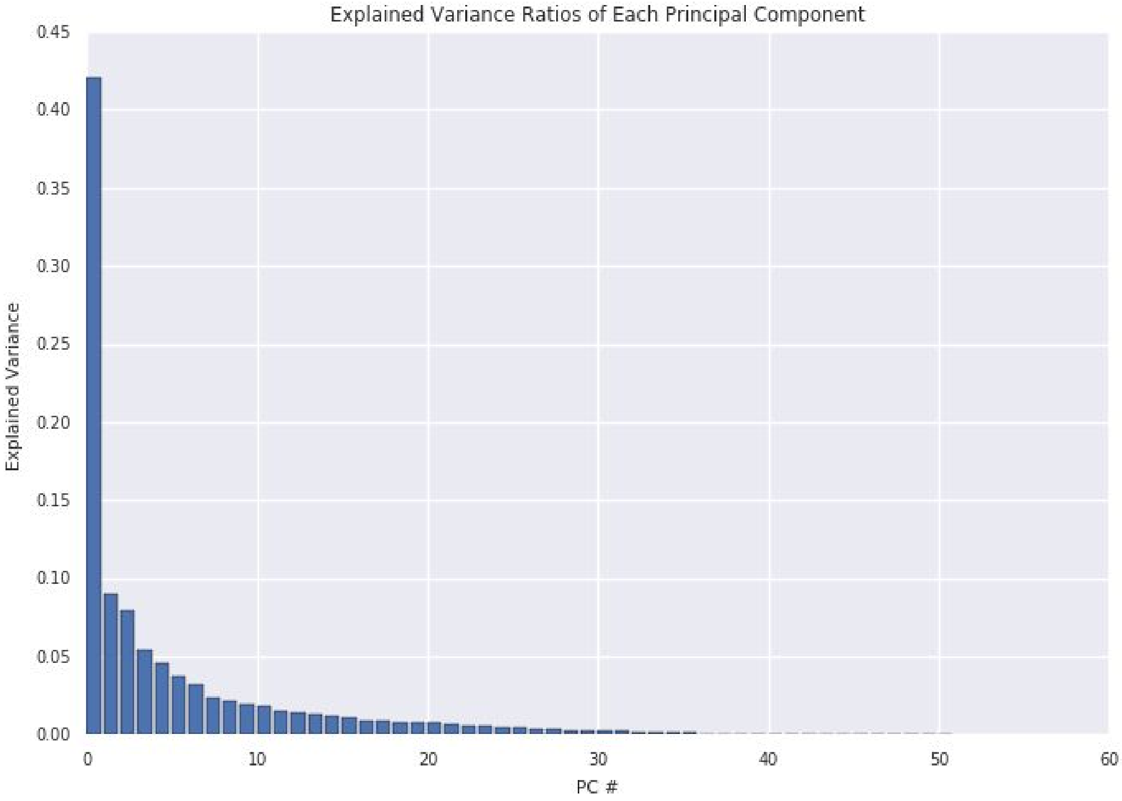


After looking at this map, there was no real way of saying these counties specifically impacted the election results. Although these are the counties that had just changed, there were more counties that had gotten more dense with the party. For example a county in Texas voted 73% republican in 2012, but in 2016 we saw that the same county voted 89% republican. In this case we decided to take a look at clustering analysis and further explore what in this election was different than that of 2012.

PCA

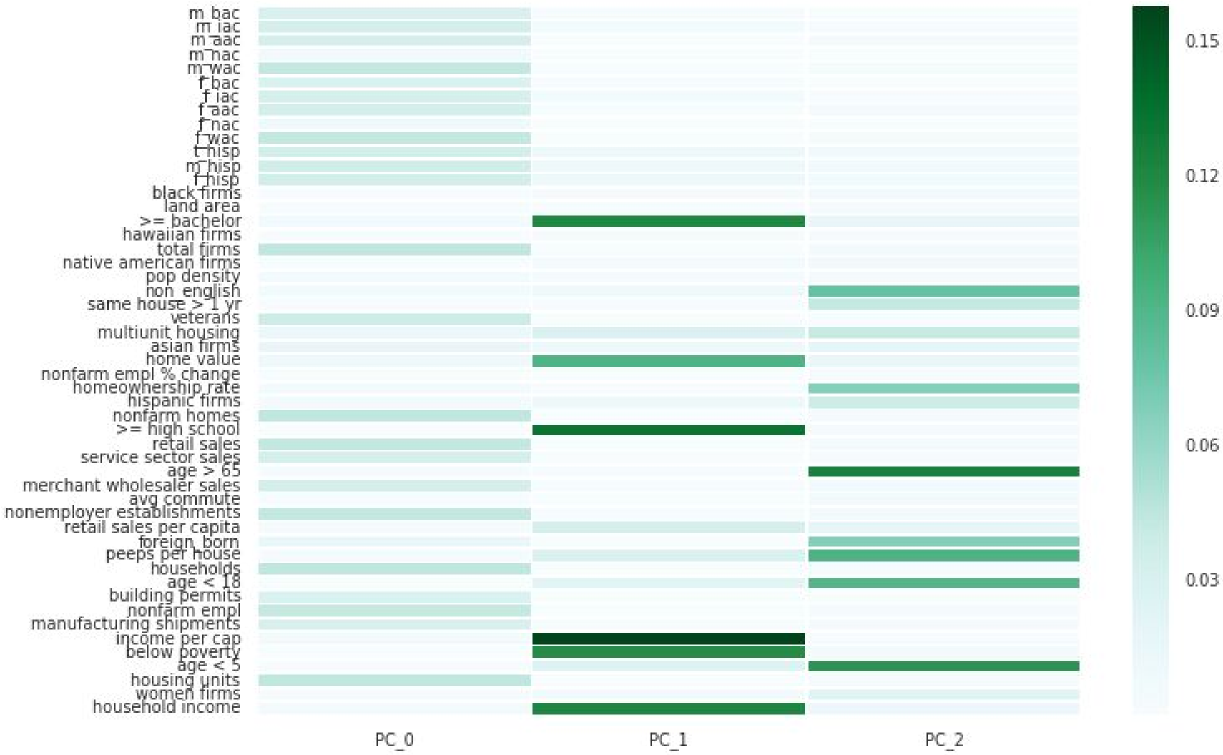
Of particular interest to us was how we might be able to cluster counties and what the voting patterns and general characteristics of those clusters might be like. However, prior to running any clustering algorithms we had to deal with the problem of many of our variables being highly correlated. Most clustering algorithms depend on some sort of distance measure between data points and having two correlated features like say '# of white males' and '# of white females' might weight that aspect when building the cluster more than we would like. We needed a way to deal with redundancy in our data.

We chose to handle this problem via Principal Component Analysis (PCA). The general idea is to reduce your features into a smaller set of 'principal components' (PC's) each of which is composed of linear combinations of your original variables and is orthogonal to every other PC. This solved the problem or correlated data and provided the added benefit of reducing the dimensionality of our data, making it easier to visualize. Prior to running PCA we chose to standardize our data so that each feature had a mean of 0 and a standard deviation of 1. The scree plot below shows the explained variance ratios we obtained for each PC after running PCA with our data.



We decided to retain 3 PCs mostly because the amount of variance explained by the higher PCs appeared insignificant, but also because three dimensions was conveniently easy to visualize.

After deciding on which PCs to keep, we examined which features were most important to each PC, hoping to find intuitive explanations for what each PC represented.



The chart above shows the squared coefficient of each original feature for each PC. Darker colors indicate higher values. To our dismay, the the first PC appeared to be a combination of many features, making it difficult to interpret. However, the second PC showed a strong relationship to income and education variables indicating the a combination of those two was important for separating our data.

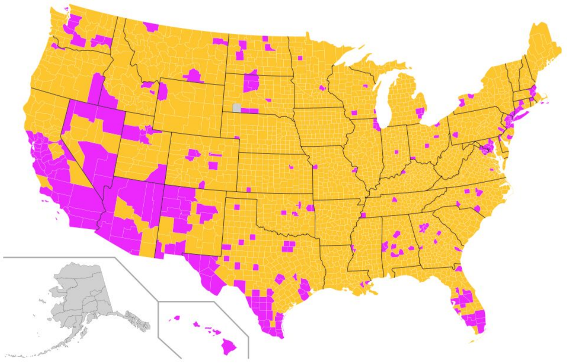
We projected our data onto our first three PCs and with newly uncorrelated data in hand, we moved onto clustering.

Clustering Algorithms

We mainly looked at two clustering algorithms for this dataset: KMeans and DBSCAN. We thought DBSCAN might create good clusters for our data because it works well with varied cluster sizes, and KMeans is a great general-purpose clustering algorithm.

DBSCAN

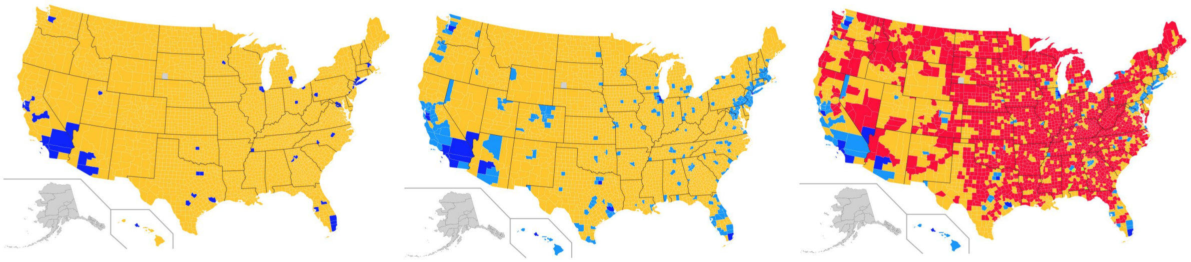
DBSCAN proved to be tough to use. Due to its sensitivity to its maximum distance parameter and the lack of volume for our data, the algorithm tended to either consume a large portion of our data into a cluster 0 , or it left the rest of the counties out as noise points (-1) because the counties were not close enough.



DBSCAN clusters. Orange is the large cluster 0, pink is the noisy data (-1).

KMeans

KMeans performed much better. After a few iterations we found that k=4 produced the most reasonable and understandable clusters from our data, and quickly noticed that the clusters looked very similar to what we know about the voting distributions in America today - one of the clusters contained Los Angeles, Seattle, and New York counties for example, which are known to be among the most liberal.

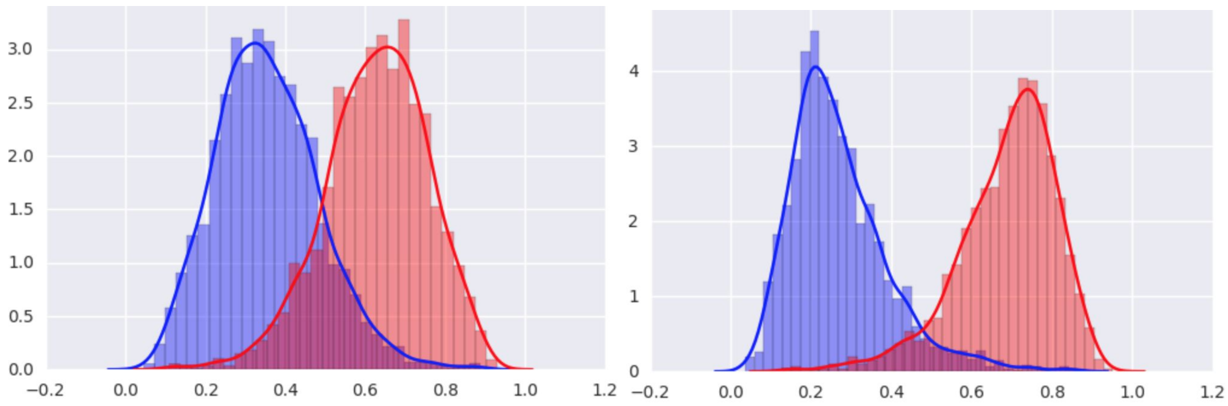


Two, three and four clusters using the KMeans clustering algorithm.

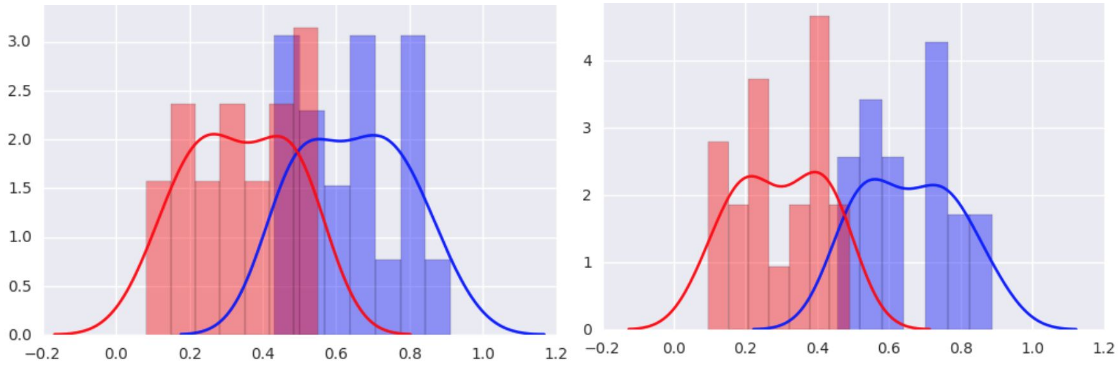
Voting Distribution of the KMeans Clusters

Cluster (2016)	Size	Dem	Rep	**Cluster (2012)**	Size	Dem	Rep
0	1998	63%	35%	**0**	2004	65% **(+2%)**	30% **(-5%)**
1	970	59%	40%	**1**	965	58% **(+1%)**	37% **(-3%)**
2	127	46%	53%	**2**	123	42% **(-4%)**	53% **(0%)**
3	16	39%	60%	**3**	19	29% **(-10%)**	66% **(+6%)**

We grouped each cluster into a label as “Democratic”, “Slightly Democratic”, “Slightly Republican”, and “Republican. The Democratic and Republican cluster voting distributions are shown below.



Voting distribution for the ‘Republican’ cluster in 2012 vs. 2016



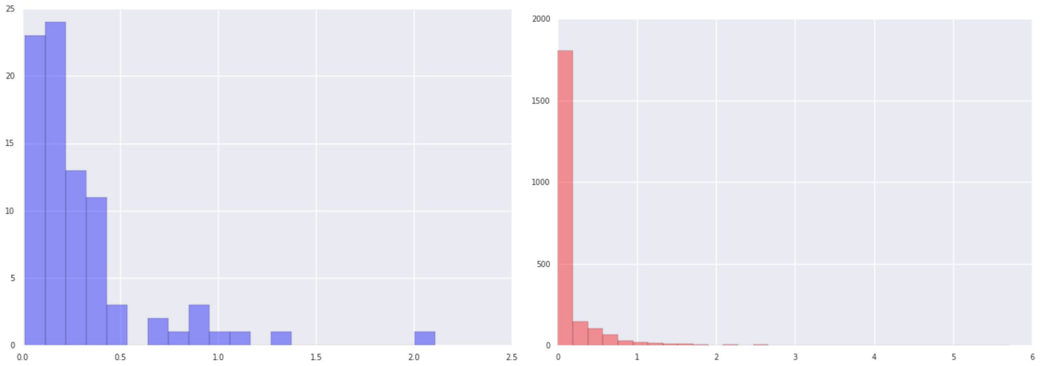
Voting distribution for the ‘Democratic’ cluster in 2012 vs. 2016

What is interesting to note is that it was easier to classify the voting distributions in 2016 than it was in 2012 - that is, there was a clear shift from the counties from the middle to each side. This means that the 2016 Democratic cluster had more counties which voted less republican and more democrat than it had in 2012.

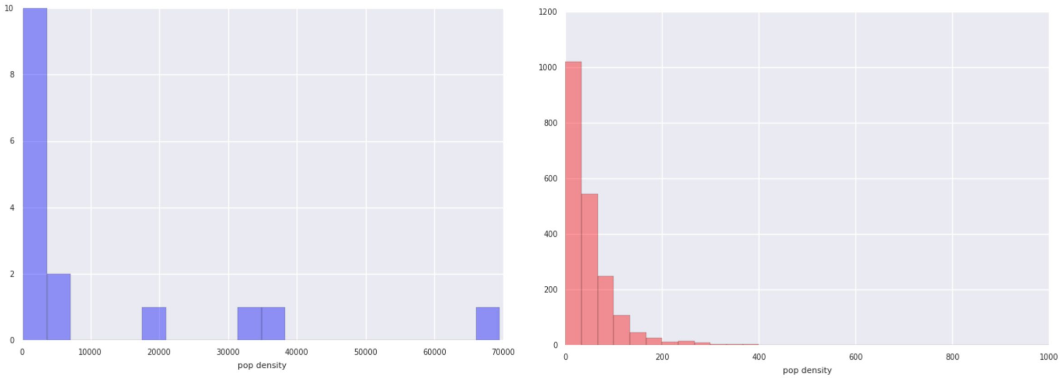
We saw that the biggest changes in our data between 2012 and 2016 was demographic information. The increase of Black, Indian, and Hispanic Americans in certain democratic counties influenced the clusters. For example, the “Democratic” cluster increased by three large counties: Alameda, Riverside, and San Bernardino county.

Properties of the Clusters

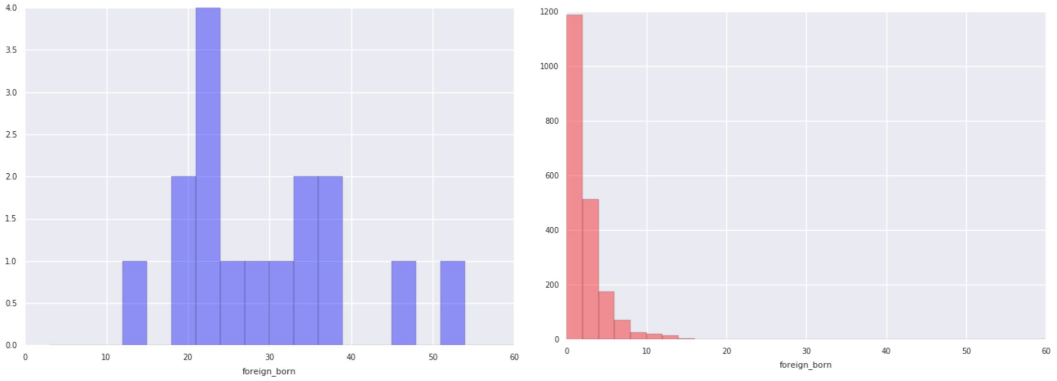
We wanted to examine some of main characteristics and differences between the “Democratic” and “Republican” clusters that we got from KMeans.



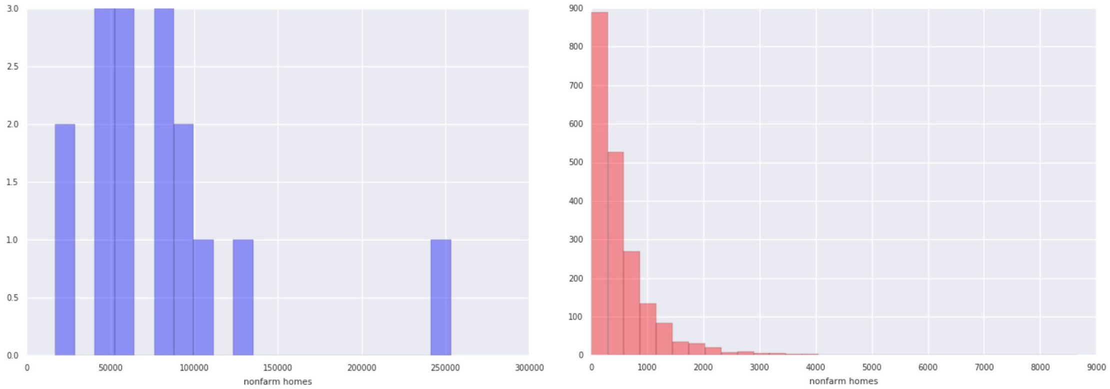
% of Male African Americans for counties in the Democratic and Republican clusters



Population densities for counties in the Democratic and Republican clusters



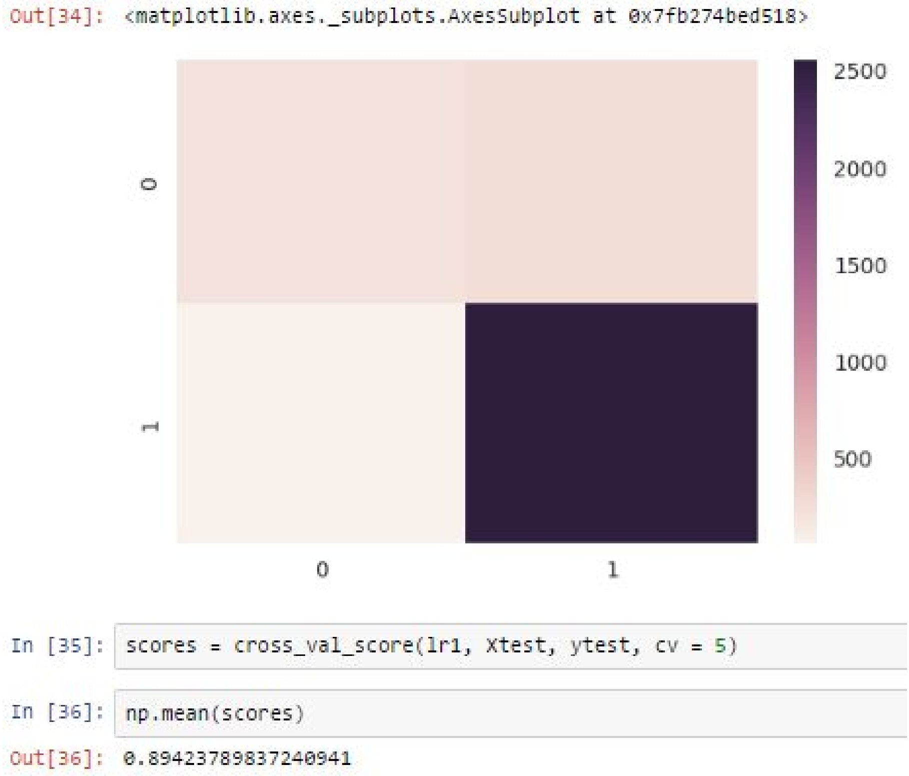
% of foreign born Americans for counties in the Democratic and Republican clusters



Number of Non-Farm homes for counties in the Democratic and Republican clusters

Logistic Regression and Closing Remarks

We built a model using logistic regression to help go back to our original question and help us predict what party a county would belong to. Using logistic regression we ran two different models, where one would test on 2012 data and test on 2016 data, and the other would run a 70-30 train test split. In these regression models, we found the accuracy of predicting a correct county to be roughly 88% accurate. We took a look at what the model was actually predicting, it is shown in the figure below.



```
Out[34]: <matplotlib.axes._subplots.AxesSubplot at 0x7fb274bed518>
```

```
In [35]: scores = cross_val_score(lr1, Xtest, ytest, cv = 5)

In [36]: np.mean(scores)

Out[36]: 0.89423789837240941
```

The confusion matrix shows us that almost all of the predictions are made as predicting the county will vote majority republican. This gives us a higher accuracy because most counties in the US are republican counties, therefore our regression model, was not very helpful in helping us predict based on demographic data which way the county will end up voting. Overall, in looking at this data, although there is no certain correct answer, we found that creating a logistic regression model was not very helpful in helping us predict a county outcome or in fact tell us who will win the election. What we did find interesting was the way demographic data greatly impacted the results in the 2016 election.