Statistics 133 - Final Project - “Team Feelin’ the Bern”

Aniket Ketkar, Tanay Lathia, Eric Priest, Steve Wang

Abstract

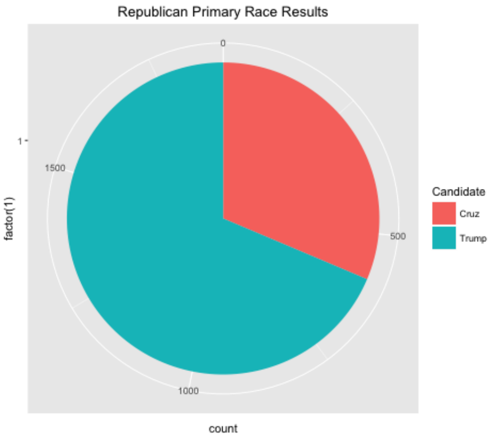
Background

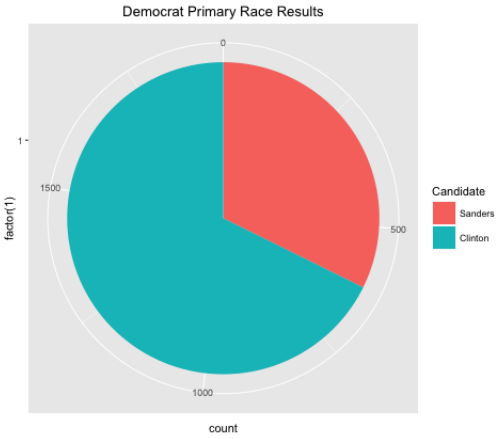
Questions for Analysis

Methods

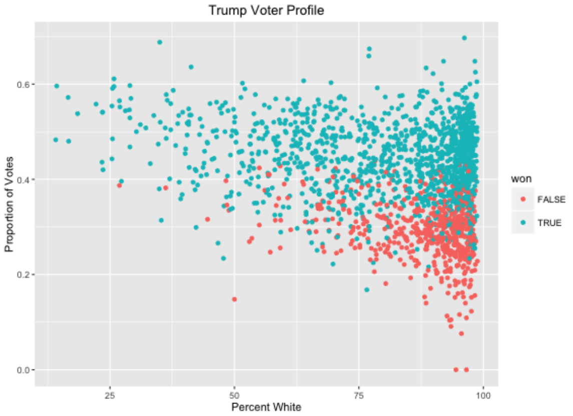
Exploratory Analysis

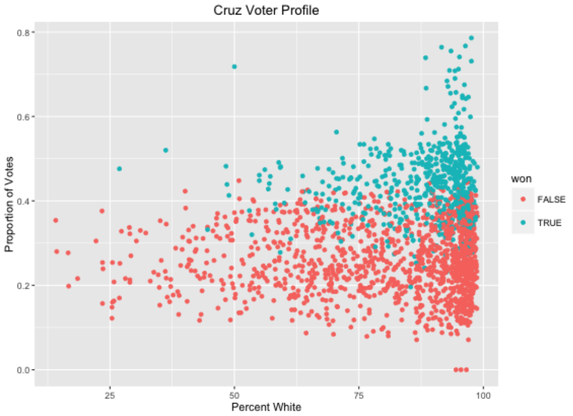
Before we jumped into performing machine learning to predict county results, we created a few visualizations to see any obvious trends in the data. The first is just a very simple pie-chart of how the race was stacking up at the time the data were collected.

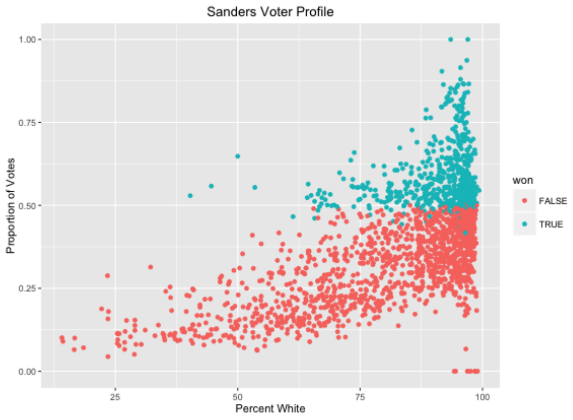


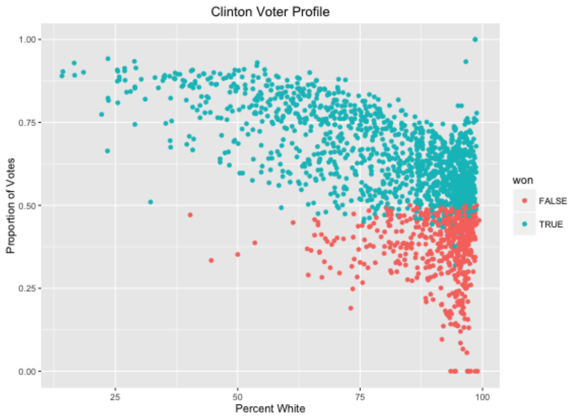


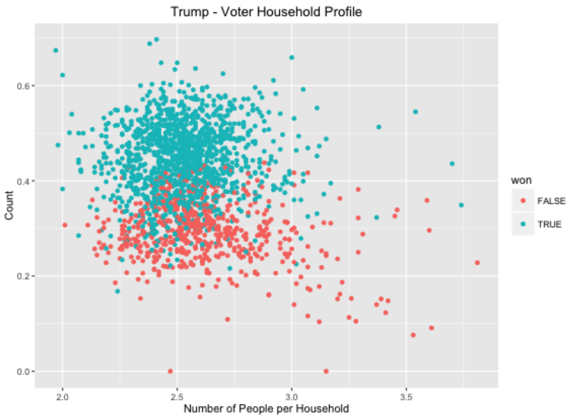
These graphs show the percentage of counties won by each candidate in their respective party. These plots are a great spot to begin our exploratory analysis because firstly they provide a motivation for the analysis: Sanders and Cruz are both on the backfoot in their party races. They need to determine which voter groups they are not hitting and refocus their efforts in order to stand a chance at claiming the nominations. Furthermore, it is important to consider these charts because the skewed nature of the current state of the primaries will bias any results we generate - especially when using K Nearest Neighbors, an unsupervised learning technique. The nature of the this tool, which we will describe in greater detail in the next section, lends itself to following the bias of the data that has already been collected; so, it is important to be fully aware of the biases before extrapolating results for predictions.

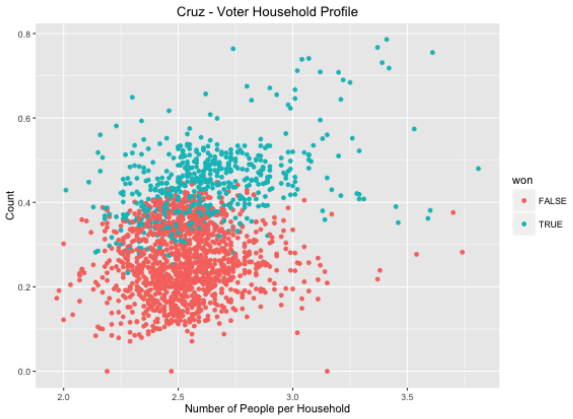


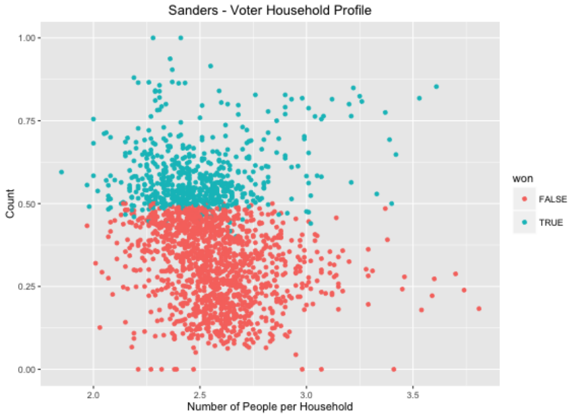


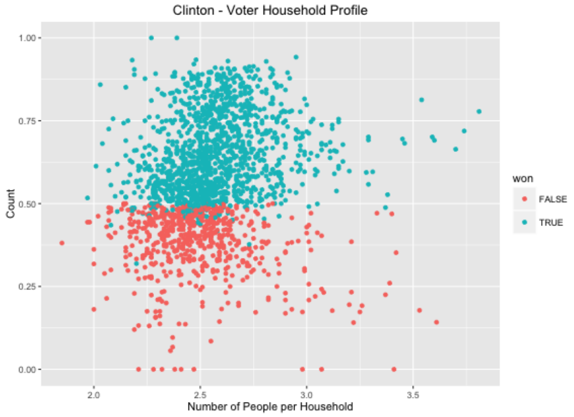

Sanders Voter Profile
Proportion of Votes
1.00
0.75
0.50
0.25
0.00
25
50
75
100
Percent White
won
FALSE
TRUE

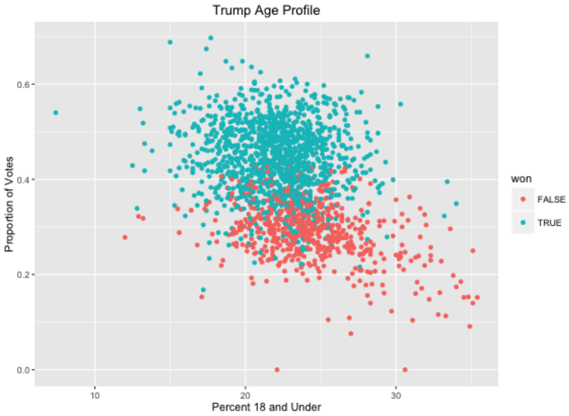

Clinton Voter Profile
Proportion of Votes
1.00
0.75
0.50
0.25
0.00
25
50
75
100
Percent White
won
FALSE
TRUE

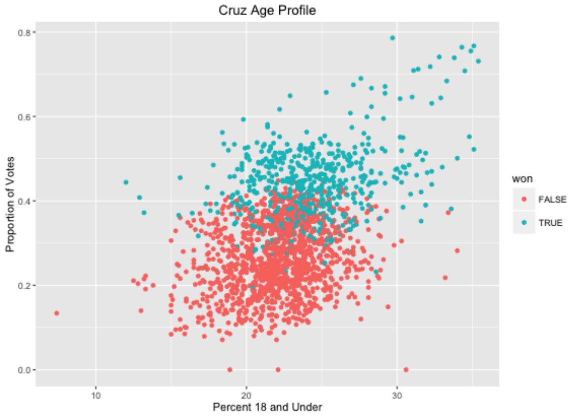

Trump - Voter Household Profile
Count
0.6
0.4
0.2
0.0
2.0
2.5
3.0
3.5
Number of People per Household
won
FALSE
TRUE

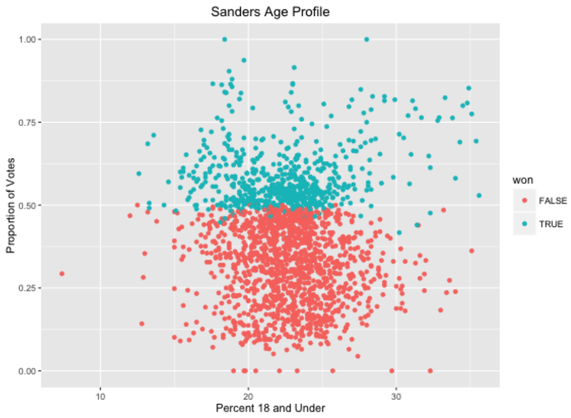

Cruz - Voter Household Profile
Count
0.8
0.6
0.4
0.2
0.0
2.0
2.5
3.0
3.5
Number of People per Household
won
FALSE
TRUE

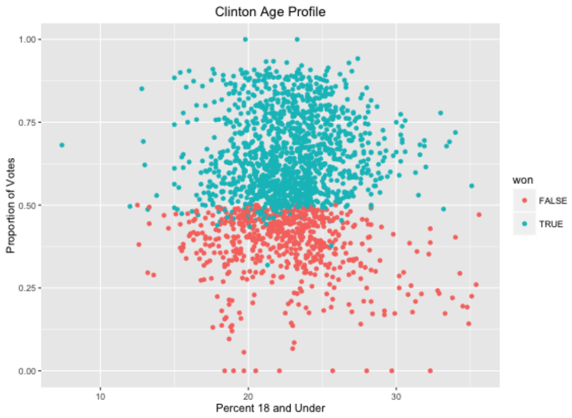

Sanders - Voter Household Profile
Count
1.00
0.75
0.50
0.25
0.00
2.0
2.5
3.0
3.5
Number of People per Household
won
FALSE
TRUE


Clinton - Voter Household Profile
Count
1.00
0.75
0.50
0.25
0.00
2.0
2.5
3.0
3.5
Number of People per Household
won
FALSE
TRUE


Trump Age Profile
Proportion of Votes
0.6
0.4
0.2
0.0
10
20
30
Percent 18 and Under
won
FALSE
TRUE


Cruz Age Profile
Proportion of Votes
0.8
0.6
0.4
0.2
0.0
10
20
30
Percent 18 and Under
won
FALSE
TRUE





Despite the shortcomings of KNN, our full exploratory analysis suggested that it made the most sense for us to use in our attempt at classification. Above are a selection of the plots we generated while reviewing the dataset. In every plot, each data point represents a county. The y-axis is always the proportion of votes that the plot's candidate received. The point of interest is always the x-axis which is the feature we were examining. In order from top to bottom, the plots examine how well each candidate does with respect to: proportion of white voters, average number of people per household, and proportion of young voters. Furthermore the plots are organized such that Cruz and Trump are next to each other and Clinton and Sanders are also next to each other in each row. This is so that inter-party comparisons (which are the only valid ones we can make based on this data) are easier. Also, the proportions of voters for Cruz plus Trump add up to 1 (approximately) and the same goes for Sanders and Clinton. This is because only these four candidates got any real traction in the election and this makes our assumption that there are only four candidates in the race.

The most compelling plot is the first one - how the proportion of votes received varies with proportion with white voters. We can see that for both Trump and Clinton if there are more than 50% non-white voters it is very, very likely that they win the county. The county is more of a tossup for counties with more white voters, but seeing this first plot made us hopeful that we could generate a linear classifier. Using just one feature, we can predict very accuracy counties with less than 50% white voters. However, as we explored more of the features in the data set, we found that no other demographic indicator was nearly as powerful as amount of whites. For example, with number of people per household Cruz seems to have more appeal than Trump for families however this doesn't seem to impact the Democratic race. There is also a similar trend that favors Cruz for younger voters - however it is not nearly as significant as the white voters feature.

Because of the limited data separation that we could get from adding more features, we determined that a linear classifier would not do well. The features we had access did not provide enough resolution between certain counties. Because of these limitations, we decided that KNN would be an appropriate model that would be the easiest for us to tune to get accurate results.

Machine Learning - K Nearest Neighbors - Equations

K Nearest neighbors is a nonparametric machine learning technique to find similar points in a dataset. It makes the assumption that similar points will have similar results, so point's label is predicted as the average label of its neighbors. Check out `knn.R` for the team's implementation of KNN.

KNN does not make any assumptions about the structure of the dataset, underlying trends, or high level ideas. It just finds the nearest neighbors. For large datasets, like the one in this paper, KNN must run through all of the training points before it can make a prediction. Performance is guaranteed to be linear ($O(nf)$) with the size of the training data n and number of features f. The more features, the farther points will be in space due to the curse of dimensionality, so we had to cut down on the featureset to make predictions better.

Let a single data point be $\mathbf{x_i} = (f_1, f_2, f_3...)$, in entire dataset $\mathbf{X}$. Then for training data point $\mathbf{x_j}$ for $i \neq j$, define the L2 distance metric for a single training example as $t_j = \sqrt{\Sigma(\mathbf{x_i} - \mathbf{x_j})^2}$. The training point $\mathbf{x_j}$ with minimum t_j is the closest example to the desired prediction point.

The algorithm is:

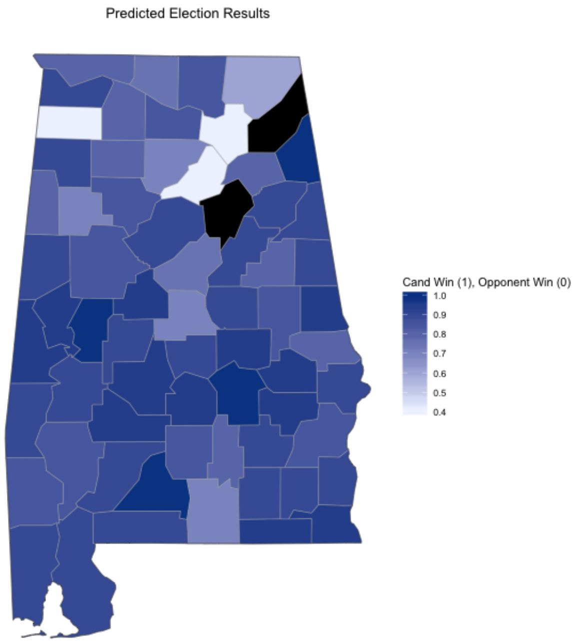
1. For all $i \neq j$, compute t_j
2. Find k smallest t_j
3. Return average of these k values.

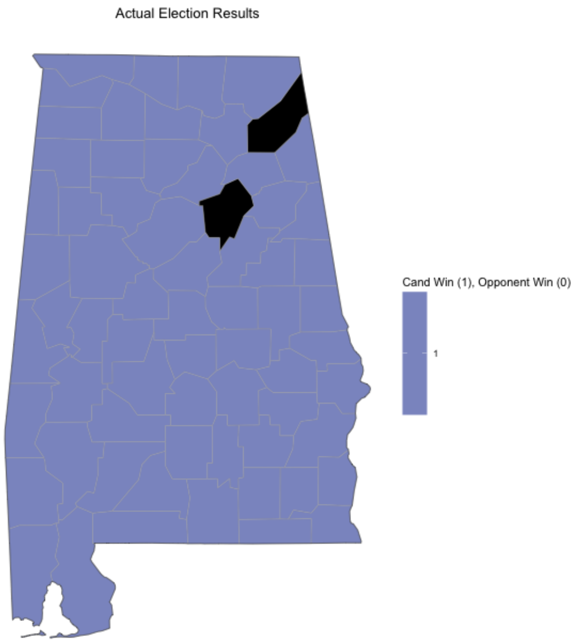
As you can see, this involves the entire training set. For state level predictions, we removed the state we are predicting on, leaving approximately 1600 counties to predict on. Each state prediction takes approximately 4 minutes (approximately 20 individual predictions).

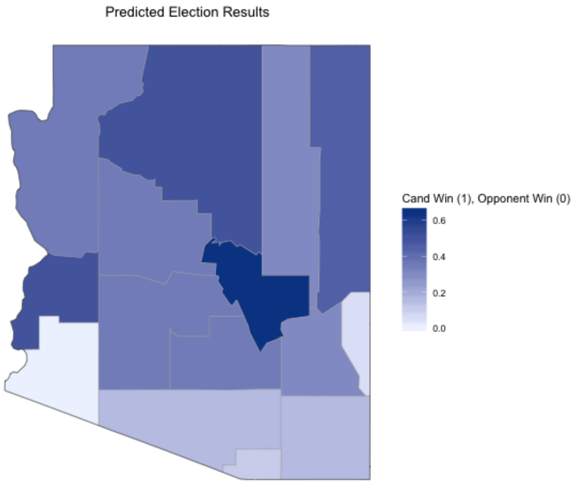
Despite the downsides, KNN performed much better than chance. Assuming two people races, a uniform assumption has expected win rate of 0.5, but every state had at least 70% success, with some over 90

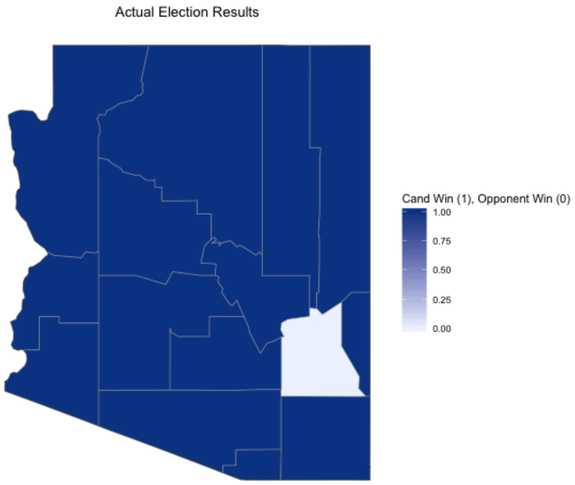
Trump v Cruz Plots

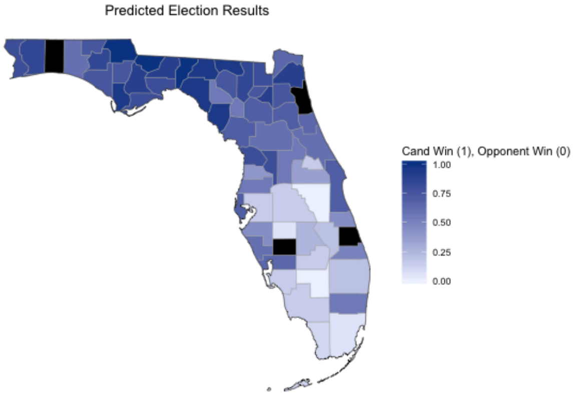
Trump's data worked mostly well with KNN. Below are some examples of states. The predictions are on the left, the actual results are on the right. Darker blue areas correspond to his predicted wins.

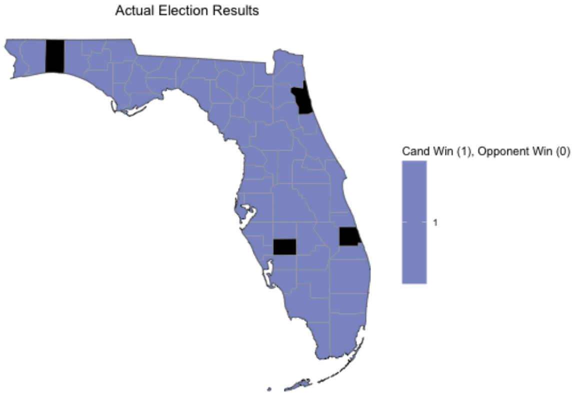

Predicted Election Results
Cand Win (1), Opponent Win (0)
1.0
0.9
0.8
0.7
0.6
0.5
0.4

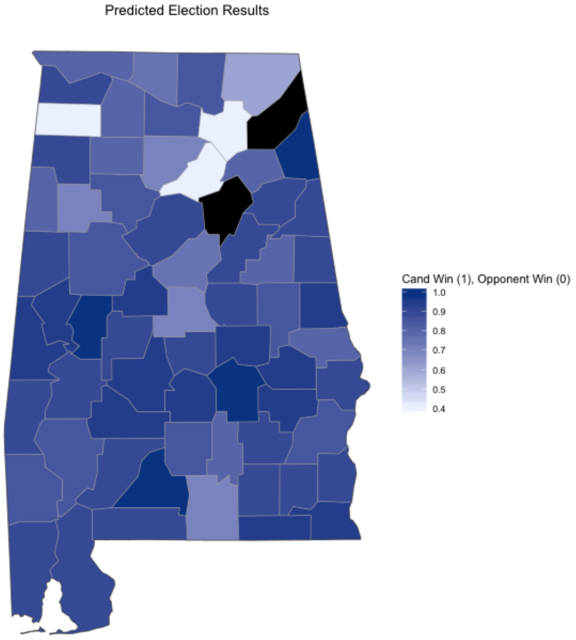

Actual Election Results
Cand Win (1), Opponent Win (0)
1

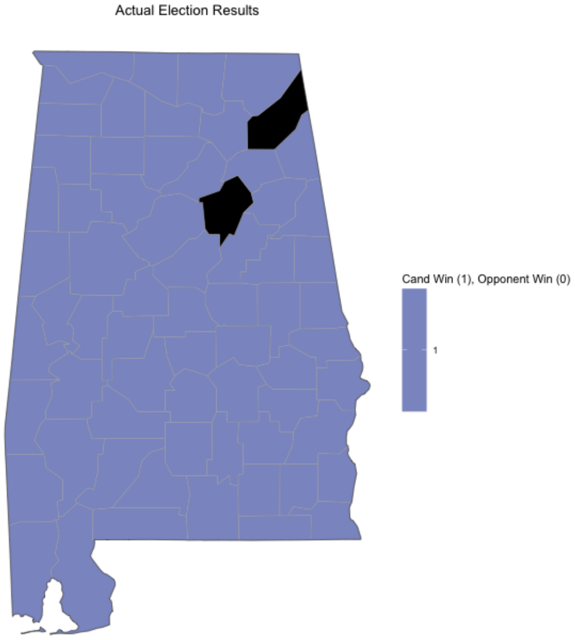

Predicted Election Results
Cand Win (1), Opponent Win (0)
0.6
0.4
0.2
0.0

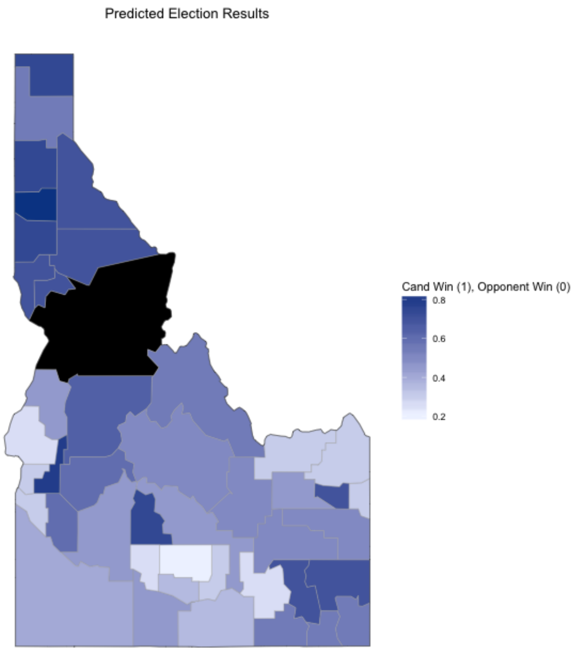

Actual Election Results
Cand Win (1), Opponent Win (0)
1.00
0.75
0.50
0.25
0.00

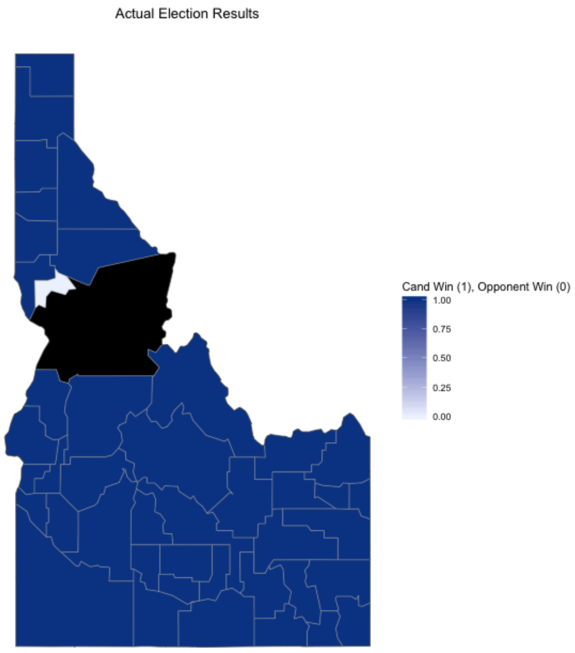

Predicted Election Results
Cand Win (1), Opponent Win (0)
1.00
0.75
0.50
0.25
0.00


Actual Election Results
Cand Win (1), Opponent Win (0)
1


Predicted Election Results
Cand Win (1), Opponent Win (0)
1.0
0.9
0.8
0.7
0.6
0.5
0.4

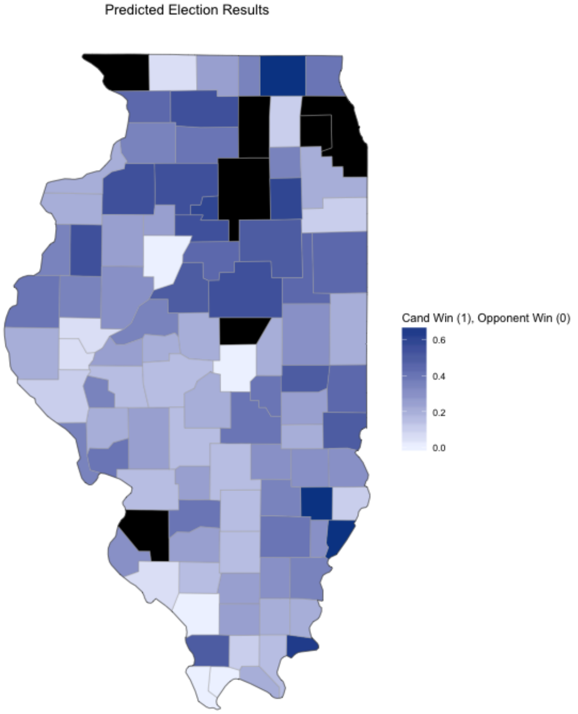

Actual Election Results
Cand Win (1), Opponent Win (0)
1

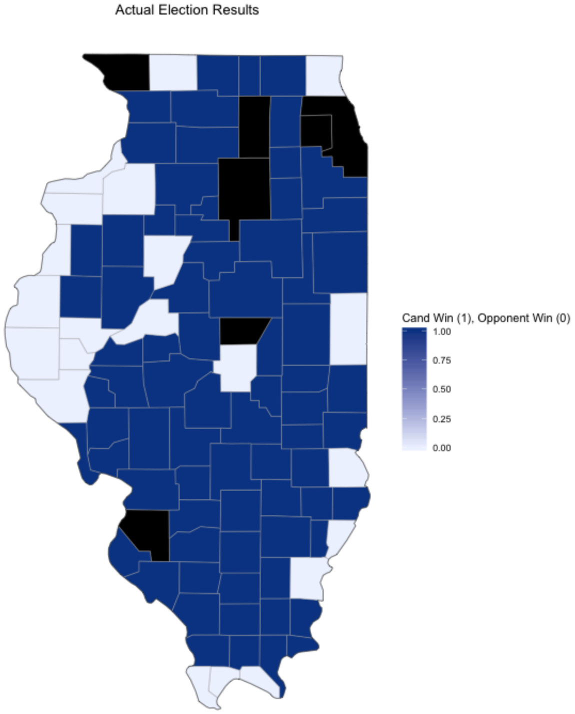


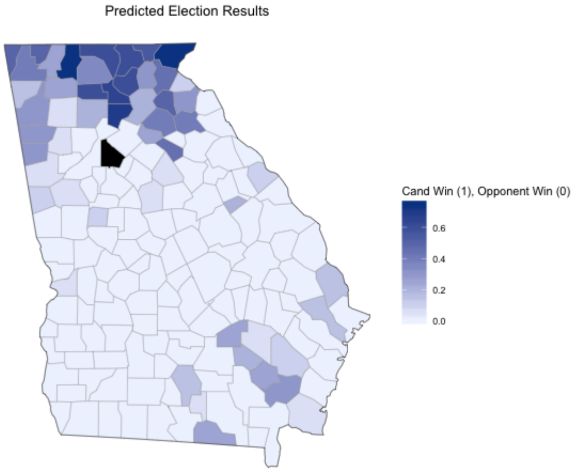


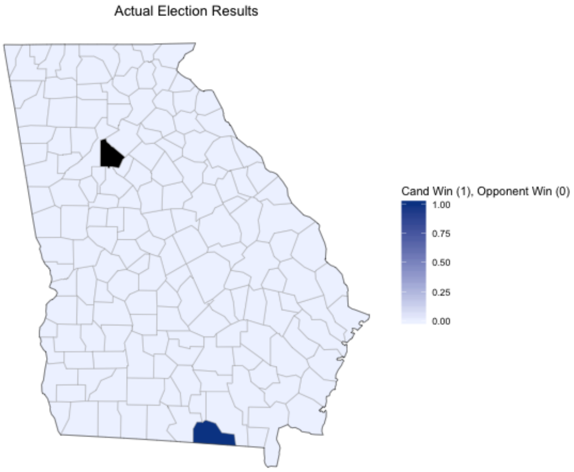
Sanders v Clinton Plots

The model somewhat works for Democratic candidates. Here are the predictions for Sanders:









KNN Analysis

The model works on a macro scale for each state, correctly predicting the majority of counties. It does well for homogeneous states, or states with clear demographic boundaries. A homogenous example is Sander's overwhelming loss in Georgia, since the counties are small and similar to each other and other Southern Clinton wins. Illinois has high variance between Chicago and other counties, and Northern states split more evenly between Sanders and Clinton, so the neighbors were split as well, leading to poorer performance.

The model also worked somewhat well for Republican candidates, in particular in Idaho. Idaho's central regions voted for Trump and the periphery voted for Cruz, and the KNN model accurately captures this pattern. This is a big success for the model.

Overall, the model works well when the training data is relatively homogenous. If a candidate has clearly won similar states, the model correctly predicts that they will win future similar states. In addition, if there are clear boundaries within a state, the model will pick up on that fact.

Final Analysis - Answering the Questions

Code Appendix