Project Proposal

Problem

This experiment will examine the relationship between demographic composition and candidate preference in U.S. presidential elections, using county-by-county data for both of these variables. The selected demographic data will address such factors as age distribution, income distribution, and racial/ethnic composition, among others. A primary goal of this experiment is to determine which demographic trends most heavily influence a voting district's disposition toward a particular candidate. An ideal solution will be able to divide the U.S. into voting blocs based solely on demographic similarities between counties. This experiment will be concerned only with individual, recent presidential elections – primarily the 2016 presidential election, and possibly others if time permits.

A model of this nature could prove useful to historians, political pundits, or anyone else interested in analyzing the outcomes of past presidential elections. It should be noted, however, that this experiment only seeks to establish a rough correlation between demographic trends and presidential candidate preference on a local basis. Aside from exit polls, no comprehensive data is available that indicates how individuals of specific demographic groups voted in past elections. Additionally, this experiment will take into account the demographic information of each county at large, remaining blind to such factors as number of registered voters and voter turnout. As such, this experiment does not aim to provide a reliable assessment of candidate preference for specific demographic groups, but rather to examine the local demographic conditions that predispose a county to vote for a particular candidate.

Data

We have already obtained our county-by-county election return data from the Harvard Dataverse website. This data encompasses the past four presidential elections, from 2000 to 2016. The sole purpose of this data set is to provide labels for the county demographic data. Thus, the only relevant features are the votes for each candidate, which are divided between the Republican nominee, the Democratic nominee, and an "Other" category.

Our model will ultimately seek to analyze counties in terms of lean toward a particular candidate. The pre-processing phase of the experiment will therefore seek to label counties in terms of a slight, moderate, or significant lean toward a particular candidate. Accordingly, the number of votes for the prevailing candidate will need to be calculated as a percentage of total votes for each county, and thresholds for slight, moderate, and significant lean will need to be determined. It might also be necessary to include a label for any toss-up counties, as defined by a minuscule lean.

The features of our data will come from the U.S. Census Bureau website, which provides comprehensive demographic data for most, if not all, U.S. counties. The Census Bureau provides APIs for various data sets, but we were unable to find an API or raw data source for the data we need. The necessary data is only available on "QuickFacts" pages for each county.

Fortunately, each QuickFacts URL adheres to a consistent format based on the county name. Furthermore, the structure of each page's HTML lends itself easily to web scraping. Thus, the QuickFacts page for each county can be obtained using the county name in the election returns data, and the HTML can be scraped for the necessary demographic data.

Once the labels have been obtained from the election returns data and the features have been obtained from the QuickFacts pages, we will have a complete data set suitable for training and testing a model. Prior to training and testing, though, we will likely perform some degree of feature reduction to eliminate any demographic information that does not have a notable effect on voting outcome, e.g. percentage of the population between the ages of 0 and 16. Principal component analysis might also provide an interesting preliminary assessment of feature significance.

Algorithmic & Experimental Approach

Treating this experiment as a classification problem would be a fairly trivial endeavor. As an initial approach, we might train a classifier using k-nearest neighbor or even a decision tree. Since our data will likely contain many features, and the boundaries between classes might not be very distinct, a decision tree has the potential to outperform k-nearest neighbor. The best approach would be to test both of these algorithms and stick with the one that performs better. We could then verify that county demographic trends do, indeed, correlate with election returns by assessing the classifier's accuracy.

The more interesting analysis, however, will come from an unsupervised learning approach, likely utilizing the k-means algorithm. Clustering will allow us to group counties into "voting blocs" based solely on demographic similarities. Determining a reasonable number of clusters will require some trial and error, depending primarily on how many demographic features we decide to include in our data. Once an optimal number of clusters has been determined, we can use our model to create the voting blocs, which will facilitate various forms of analysis. Using the labels of the bloc members, we will be able to compare the voting preferences of different demographic spheres within the U.S. We could also test the predictive value of these blocs by comparing our classifier results to our own predictions based on our cluster analysis. Finally, we could compare our voting bloc trends for the 2016 election to the demographic information in the 2016 exit polls.

Development Plan

1. Data Procurement & Pre-Processing
    - Divide election return data into four separate sets, one for each election
    - Obtain labels from election return data
    - Obtain features from Census Bureau QuickFacts pages
    - Package all data into a data set format suitable for a scikit-learn model
2. Preliminary Analysis and Implementation
    - Experiment with feature reduction to eliminate any irrelevant features in data
    - Use principal component analysis to get a preliminary look at most important features

- Train a classifier and test its accuracy to verify existence of a correlation between experiment variables
- Obtain clusters of related counties using a k-means model

3. Experimental Analysis of Results
    - Analyze voting trends among county clusters and related demographic trends
    - Compare trends in clusters to classifier results for same counties
    - Compare relation between demographic trends and voting results to similar information in corresponding exit polls
    - If accuracy is insufficient or results do not seem realistic, revise model and analyze new results

Brendan will take responsibility for most of the work for the first checkpoint, as he has extensive experience with web scraping in Python.

The second two checkpoints will be more of a group effort, as they rely less on specialized programming knowledge and more on machine learning concepts emphasized in the course. For the second checkpoint, Brendan and Zack can focus more on the implementation phase of the model itself, while Winston and Ben can focus more on feature preprocessing and principal component analysis. If necessary, one pair can also work more on the classifier, while the other works more on the clustering, but neither of these tasks falls far outside what each of us has already accomplished in the homework assignments.

The third checkpoint will require collective group effort.