US Election 2016 Analysis

Shivam Swarnkar

Abstract

This project analyzed US 2016 election county results with respect to county's demographic features like population, population change, Black population, Hispanic population, age, education, income, poverty and density. Logistic regression and subset regression were used to rank all the combinations of features in order of prediction accuracy. The best accuracy from all tested logistic regression models was 91.43%, while population was the most successful single feature with 86.97% accuracy result. After analyzing the logistic regression, same data samples were used to train a Neural Network with 1000 hidden layers which gave the prediction accuracy of 93.05% in 200 epochs. In addition to training with all features, NN was separately also trained with best and worst feature combinations ranked from subset regression which gave 93.57% and 82.5% accuracy respectively. From result analysis, it was concluded that Democrats had higher chances of winning where Hispanic population, Black population and Education rates were higher, while Republicans had higher chances of winning with high poverty, high density and low education features.

Objective

Objective of this project was to train and compare accuracy of Logistic regression and NN models using US Election 2016 data, and to find out if there are any feature patterns using subset regression.

Problem Statement

It's hard to predict human behavior. One of the ways to predict a behavior is by using election data. In all democratic nations, election plays an important role in almost everyone's life. Therefore, analyzing election results could help in understanding human behavior. It can help us in answering few questions like What generally affects people's voting behavior? Does demographic features play important roles in any election results? If yes then to what degree a demographic feature can affect election? Do these demographic patterns always work similarly or do they work in some specific time frame only? How does one country's people's behavior relate to other country's people's behaviors? In this project, I aim to answer few of these questions.

Background

Many people have been analyzing election data along with social media data and demographic data. In many countries, it has been noticed that politicians are starting to use AI bots to target specific population to affect population's thinking and win the elections. Many even follow immoral ways such as by spreading fake news to specific population segment. And with rising social media and advancing AI and ML fields, it's becoming impossible for humans to fight against it. Therefore, smarter bots to counter such immoral bots are needed. But before they can fight, they need to learn population pattern which can attract any immoral bot. In order to learn such patterns, we need to analyze social media data, election data, demographic data and potential bot data. Much work is being done by independent agencies and companies to achieve this goal which includes Facebook and Twitter. This project aims at analyzing election data and demographic data to find behavior patterns, so that such patterns can be used to train fake news buster bots.

This project does not build up from any of the previous work done in the area, instead it's an independent analysis of demographic features to analyze their relationship with election results.

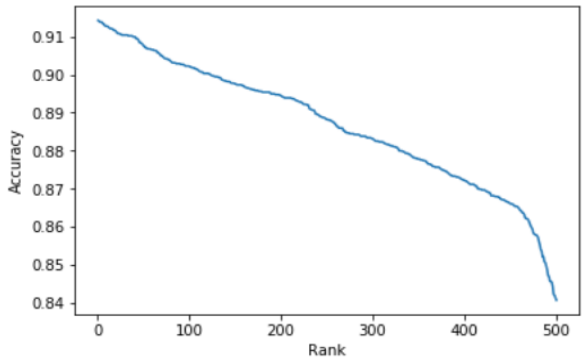
Approach

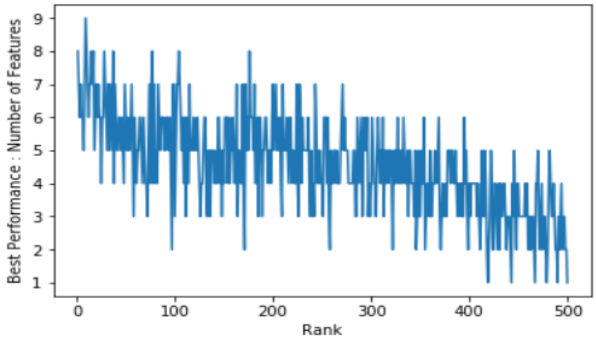
Project is divided into two main parts. First part applies sklearn linear_model logistic regression model to training data and plots several combinations of input features to analyze the result. Second part uses these results, and trains a Neural network with 1000 hidden layers with all features, best features and worst features combinations. Then best accuracy results from both parts are used to compare the accuracy. For all the calculations, the data was scaled.

Results and Evaluation

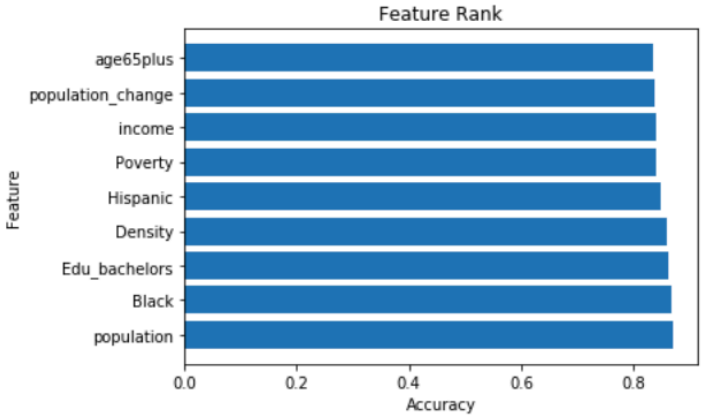
The best logistic regression model produced accuracy of 91.43% where best Neural Network model produced accuracy of 93.57%. Therefore, NN performed better than logistic regression to solve this problem.

Following left figure is showing the accuracy drop from best to worst combinations and right figure is showing number of features used in model v/s accuracy rank. From right figure, we can see that as we decrease number of features used in model, we get lower rank. However, there is a fluctuation because some features are less relevant than others. And some features may also decrease the accuracy.



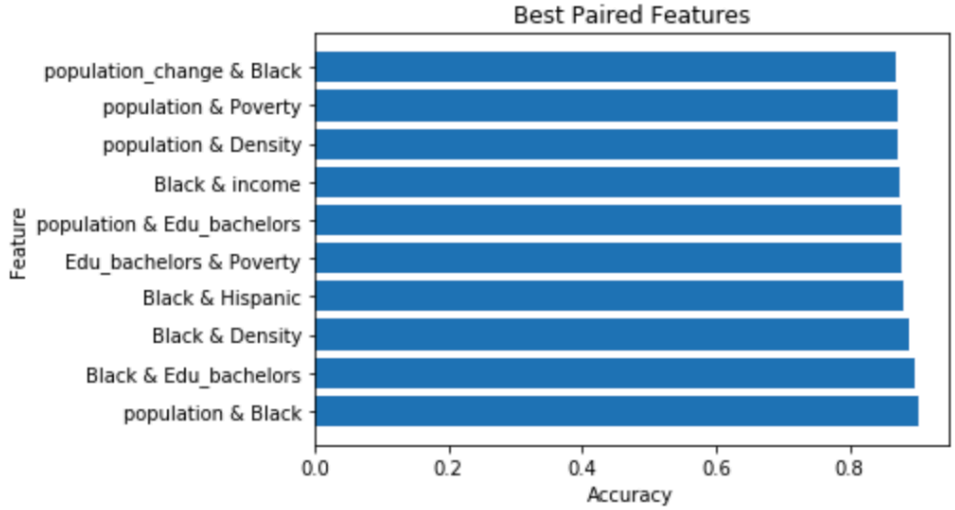


From subset regression, following feature rank was determined where age65plus was worst and population was the best.

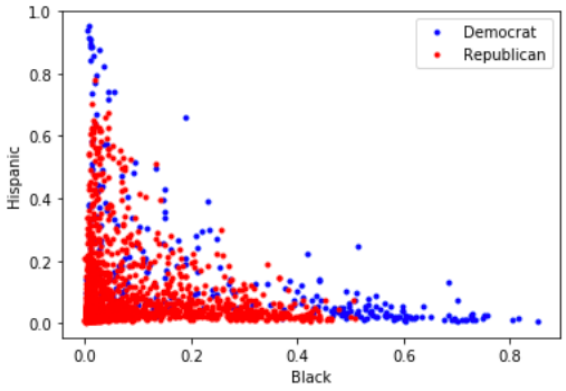


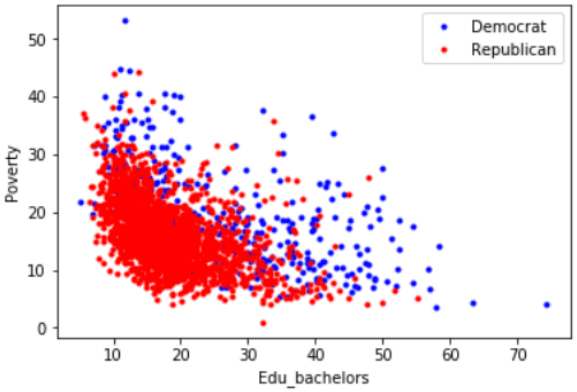
The best feature combination included population, population change, Black population, Hispanic population, education', income, poverty, density but not age65plus feature. It gave 91.43% accuracy in logistic regression and 93.57% accuracy in Neural Network. This result indicated that age65plus is not a good feature to use in prediction, and single feature ranking also indicated the same thing. age65plus was ranked the worst feature for prediction.

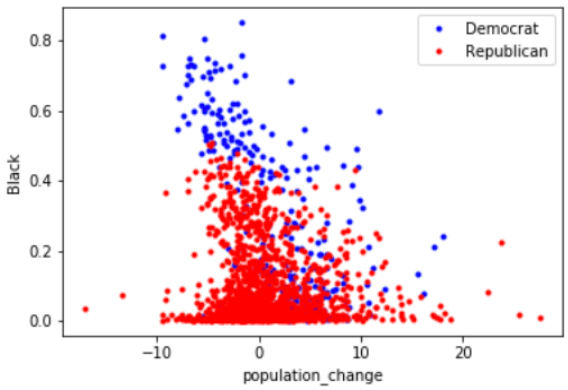
Top pair features were used to find patterns in data.

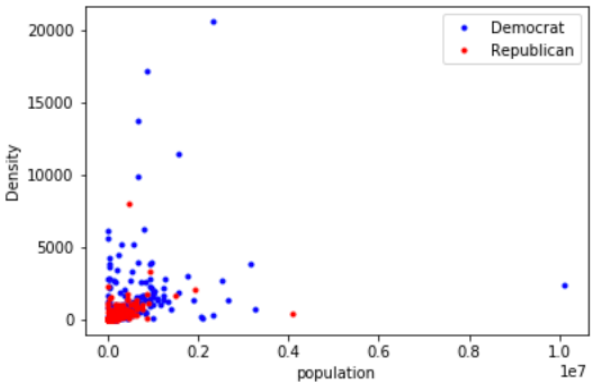


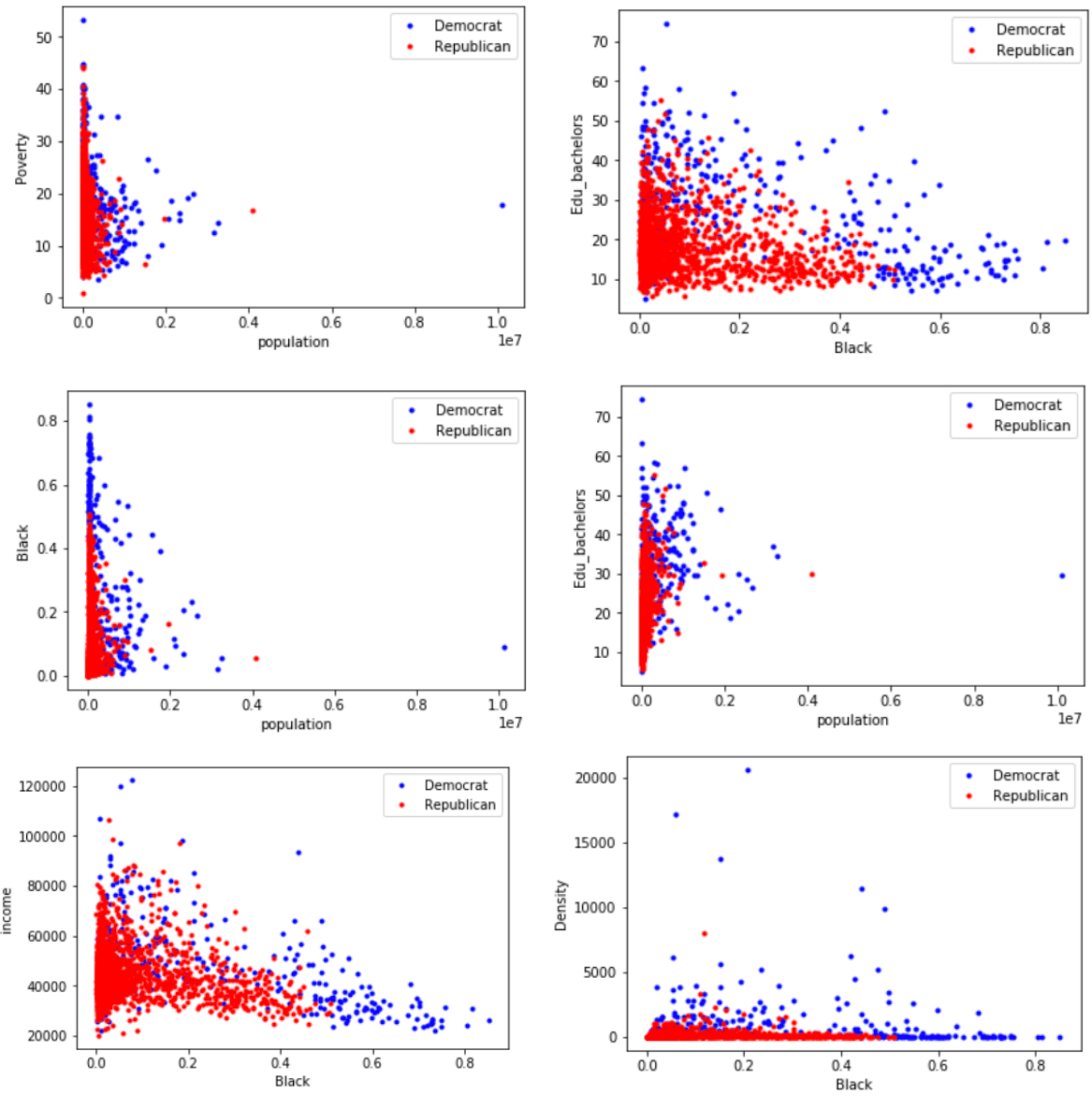
Top 10 features were selected and plotted to find patterns.











From analyzing above plots, following patterns were concluded

- Higher Hispanic and Higher Black population voted in favor of Democrat (Hispanic vs Black, Black vs population_change, Black vs population, density vs Black)
- Poverty and Education had mixed results, but medium poverty and lower education voted in favor of Republican (Poverty vs Education, Education vs Black, Education vs Population)
- Lower income voted in favor of Republicans, where higher education voted for democrats.
- Less dense county's voted in favor of Republicans

In other words, more education, larger black and Hispanic population with high dense poverty places voted more in favor of Democrats where less education, less dense but high/medium

poverty, less population and less income places voted in favor of Republicans. Black and Hispanic patter was expected as Democrat's image supports people of colors and immigrants, however pattern in education, poverty and density were not expected. Another interesting learning was that before analysis, it was expected that age will be an important feature, however, age was ranked as the worst.

Conclusion & Future Goals

Neural network performed better logistic regression which was expected as the data is not linear, and neural network clearly provided higher order of complexity than Logistic regression. Visual graphs helped in determining some useful strong patterns, some of which were also expected. Since some useful patterns were found, and all models produced good results, we can conclude that project was successful as it achieved its objectives.

In future, I aim to use these patterns and results to train an actual bot as stated in background. I'd also like to test these models with previous election results, and see if they behave similarly. I also aim to compare these results and patterns with international election data to see if there are any similarity between them. Many countries have more than 2 party system (ex. India has more than 10 political parties), which can give some interesting insights about human behaviors.

How to Use

Before running any files from the project, please make sure you have installed and downloaded all the required following dependencies and data files.

- Numpy
- Pandas
- Sklearn
- Matplotlib
- Tensorflow
- Keras
- Data Files
    - votes-train.csv **[used to train the models]**
    - votes-test.csv **[used to test the models]**

To reproduce the results, you should first run the Logistic Regression.ipynb file, which will give you step by step results in form of printed strings and graphs. This file is well commented and should be easy to use. However, if you want, you can also you LogisticRegression.py file to reproduce results as well. In the last sections of Logistic Regression.ipnb file, you can find several graphs which can help in visual analysis of features.

To reproduce the results from Neural Network, you can run NeuralNet(Tensor-flow).ipynb file. Remember that this file hardwires the results from Logistic regression for best and worst features, therefore any changes in main data files will not update worst and best features in NN. Use of ipynb file is recommended for learning/understanding the project.

The file BackpropogationEntrop.m (matlab file) is an attempt at making N0-N1-N2 type of Neural Network from scratch. This file is independent from project; therefore, it can be used to create and train any N0-N1-N2 type of Neural Network. This implementation uses cross-entropy as cost function, sigmoid as transfer function and quadratic cost function to calculate error, which is used as convergence condition.

You can download ProjectReport.pdf and ProjectPresentation.pdf for more information.

Note:

This project was built as a final project for the class Intro to Machine Learning taught by Professor Sundeep Rangan, at NYU Tandon School of Engineering.

All project artifacts (documents and code) were produced by Shivam Swarnkar. No previous project artifacts such as source codes were used in this project.

References:

- **NA**