PREDICTING NRM's SCORE AT PRESIDENTIAL LEVEL AND PARLIAMENTARY LEVEL

MCS 7227: Data Analytics and Visualisation

Sulaiman Kagumire
2100702296
2021/HD05/2296U

Introduction

Since 2001, the NRM party has been the only constant factor in all elections. This report describes the feature engineering steps taken in obtaining two (2) datasets capable of predicting the percentage of NRM votes against total cast votes at the presidential level and NRM MPs percentage at parliamentary levels, against combined opposition, respectively. I also describe how I trained classical machine learning models to make the above predictions. The best model is deployed on streamlit at https://ksulaiman1-mcs-7227-deploy-nez3d4.streamlitapp.com/.

Data collection and cleaning

2001-2021 election data for presidents and MPs were obtained from the https://www.ec.or.ug/ website in pdf format.
The datasets included presidential results for each year and parliamentary results. Using online pdf converters, this data was converted to CSV and Excel file formats for proper analysis and reading as DataFrames with pandas library.

Unfortunately, 2001 parliamentary data is not available on the EC website. 2006 parliamentary data is available however, the candidate's political parties were not included in the tables. Both these election years were not included in the analysis for NRM parliamentary scores.

Also, 2001 and 2006 presidential data were not included in predicting NRM presidential scores. This was because there are many new districts in 2011, 2016, and 2021, that did not exist in 2001 and 2006. This created a big number of missing values for those districts in those two years. The final decision was to eliminate 2001 and 2006 data to prevent overfitting.

The converted CSV and excel data files were intensively cleaned using NumPy and Pandas python libraries. Data cleaning included:

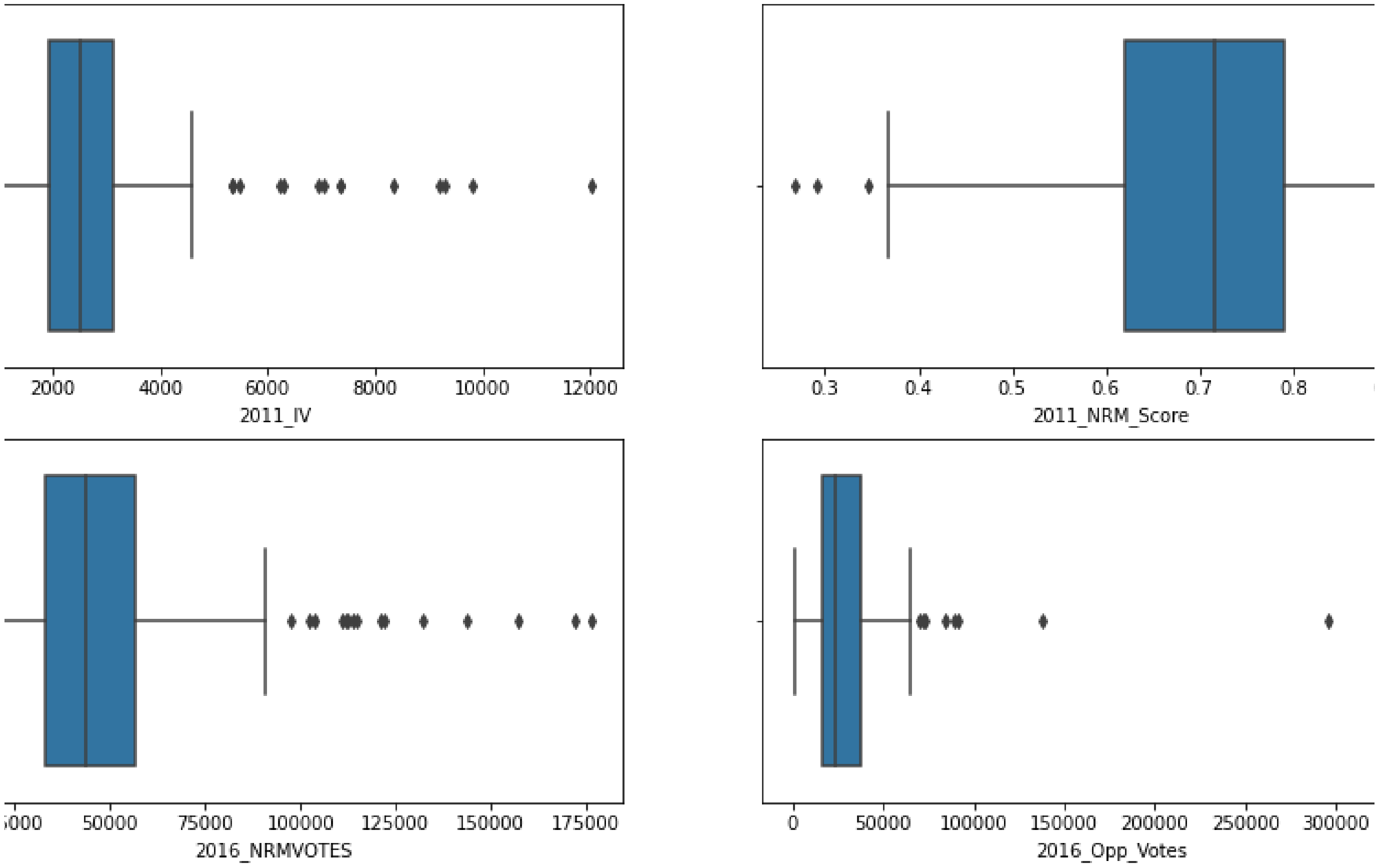
- removing empty lines
- removing duplicates
- Dropping and Imputing missing data

These steps were applied to both presidential and parliamentary data files.

Exploratory Data Analysis

Statistical data analysis was performed for each cleaned data file in each election year. For example, Fig. 1 below shows outlier points in four features of the combined data.

Machine learning algorithms are sensitive to the range and distribution of attribute values. Data outliers can spoil and mislead the training process resulting in longer training times, less accurate models and ultimately poorer results. Therefore, these outliers were dealt with by replacing them with the mean value of a column.



Feature Engineering

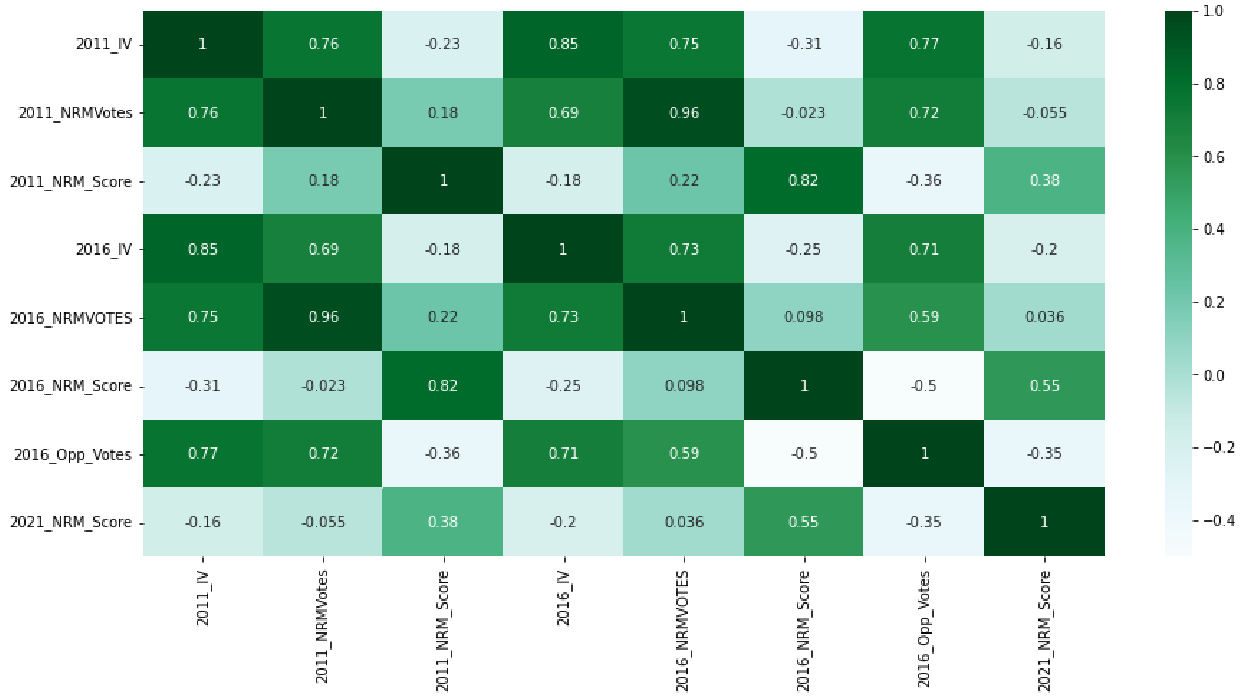
1. Presidential Data

After cleaning and removing outliers in presidential data for each election year, different features were engineered to form new features that could be used for training and inference. For example:

- All opposition candidates' votes in each district were added to form a total opposition vote.
- The target feature (NRM SCORE) was created by dividing NRM votes over the total valid votes from each district for each election year.
- All columns with opposition candidates were dropped to mitigate the curse of dimensionality.
- Based on feature correlation coefficients, one of the features with high linear correlation was dropped to mitigate overfitting and the curse of dimensionality.

Feature Correlation

Correlation is a term that refers to the strength of a relationship between two variables where a strong, or high, correlation means that two or more variables have a strong relationship with each other while a weak or low correlation means that the variables are hardly related. Some features or columns with high correlation coefficients between them were dropped.



For example, features named “2011_NRMVotes”, “2016_IV”, and “2016_NRM_Score” were dropped as their coefficients of 0.96, 0.85, and 0.82, respectively, are high with other features.

The final dataset, for predicting the 2021 NRM presidential score per district, included four features: 2011_IV'(2011 total Invalid Votes per district), '2011_NRM_Score'(2011 NRM percentage per district), '2016_NRMVOTES'(2016 NRM votes per district), and '2016_Opp_Votes'(2016 total opposition votes per district). The target feature was '2021_NRM_Score' (2021 NRM percentage per district).

Model Training and Evaluation

Before training a model, this final dataset was normalized using the StandardScaler preprocessing estimator. The idea behind StandardScaler is that it transforms data such that its distribution will have a mean value of 0 and a standard deviation of 1.
The standardised data was split into training and testing data for training and evaluation, with a ratio of 80:20.
Various Machine Learning algorithms were trained and tested. These include XGB Regressor, GradientBoosting Regressor, Light GBM Regressor, RandomForest Regressor, and DecisionTree

Regressor. However, the Gradient Boosting Regressor algorithm gave the best accuracy score and lower mean squared error of **66.9% and 0.03,** respectively.

Deploying the best model

GradientBoosting Regressor was pickled and saved as a pickle file. It was deployed as a streamlit web application:
Link to deployment - https://ksulaiman1-mcs-7227-deploy-nez3d4.streamlitapp.com/
Link to GitHub repository with code and results: https://github.com/ksulaiman1/MCS-7227

2. Parliamentary data

As stated above, only 2011, 2016 and 2021 data was used for predicting the percentage of NRM parliamentary MPs per district.

The above similar steps were deployed on the parliamentary data, for Women and elected MPs.
Five important features were considered when training the prediction model:

- □ 2011_NRM_total_Votes'(2011 total NRM Votes per district)
- □ '2011_total_Opp_Votes'(2011 total opposition votes per district),
- □ '2016_NRM_total_Mps'(2016 NRM total MPs per district),
- □ '2016_total_Opp_Mps'(2016 total MPs per district),
- □ '2016_total_Opp_Votes' (2016 total Opposition votes per district)

Model Training and Testing

As stated above, this data too was standardised in order to keep all feature data in the same range.
The standardised data was split into training and testing data for training and evaluation, with a ratio of 80:20.
Various Machine Learning algorithms were trained and tested. These include Extra Tree Regressor, GradientBoosting Regressor, RandomForest Regressor, and DecisionTree Regressor. However, the Extra Tree Classifier Regressor algorithm gave the best accuracy score and lower mean squared error of **59.8% and 0.093,** respectively.
The Extra Tree Regressor model was pickled and saved on GitHub:
https://github.com/ksulaiman1/MCS-7227

Discussion

As seen from the above results, the models do not perform well on both parliamentary and presidential data. This is because the data was too small. All models were underfitting the data.

Obviously more data, and even using Multilayer perceptrons might help increase the model accuracy and reduce mean squared error.