Python Assignment Report

Harshit Srivastava

April 14, 2024

1 Methodology

1.1 Data Preprocessing

The following preprocessing steps were performed on the data:

- Columns of `Total Assets` and `Liabilities` were converted to numeric format.
- Columns of `ID`, `Candidate Name`, `Constituency` were dropped as they were mostly unique to each candidate.
- Categorical columns like `Party` were converted to numeric using ordinal encoding. This was preferred over one-hot encoding because the number of unique values in each column was large, and one-hot encoding would have resulted in a large number of columns. The class to be predicted - `Education` - was also encoded using ordinal encoding.
- Training set was split into a training and validation set using a 85-15 split.

2 Experiment Details

The following models were trained on the data:

- Logistic Regression
- Decision Tree
- Random Forest
- K-Nearest Neighbors

Individually, the models did not perform well. Logistic regression predicted only one class. Decision Tree and Random Forest performed very well on the training set but did poorly on the validation set, which suggests that they were overfitting the training data. K-Nearest Neighbors performed better, which was still not satisfactory.

A new approach was tested where all the models were combined into an ensemble model. The ensemble model was created by taking the majority vote of the predictions of all the models. This ensemble model performed better than the individual models. Finally, this model was used to predict the class of the test data.

2.1 Data Insights

The following insights were obtained from the data:

We can see here that the training set has a high accuracy, but the validation set has a low accuracy. This suggests that the model is overfitting the training data.

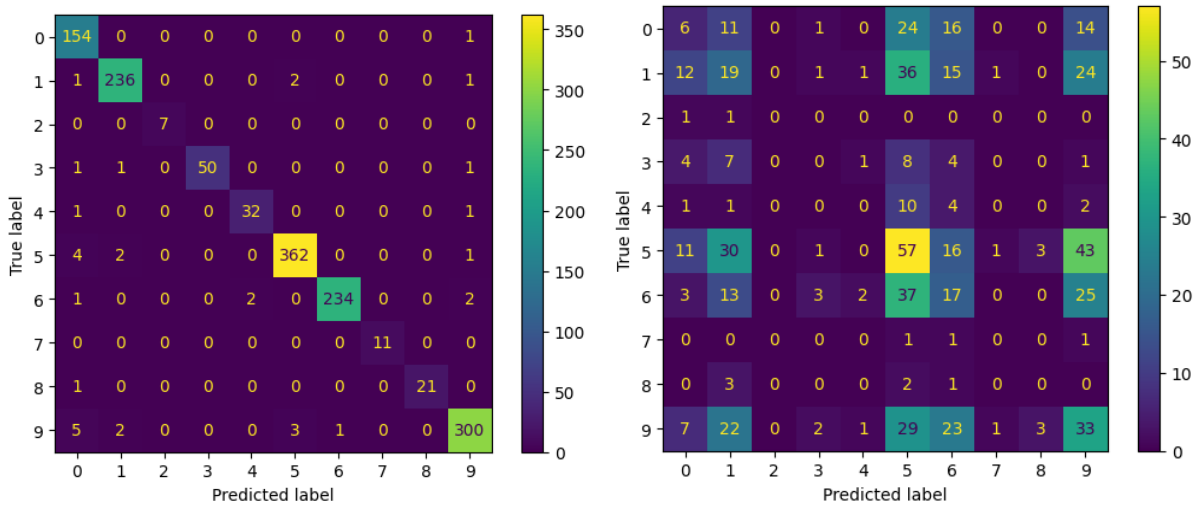

Figure 1: Confusion matrices of the training and validation sets for Random forest classifier(left and right respectively)

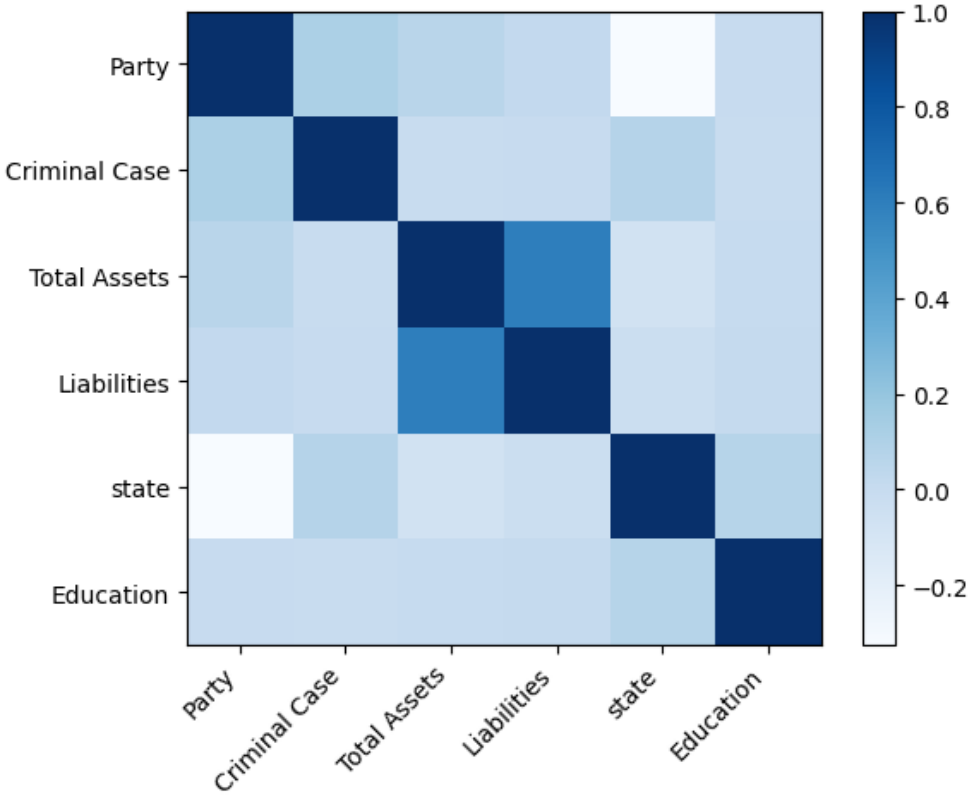

Figure 3: Correlation matrix of the data

The correlation matrix shows that there is no correlation between the `Education` column and the other columns. This suggests that the models are not able to learn the relationship between the features and the target variable. This is the reason why the models are not able to predict the class of the test data.

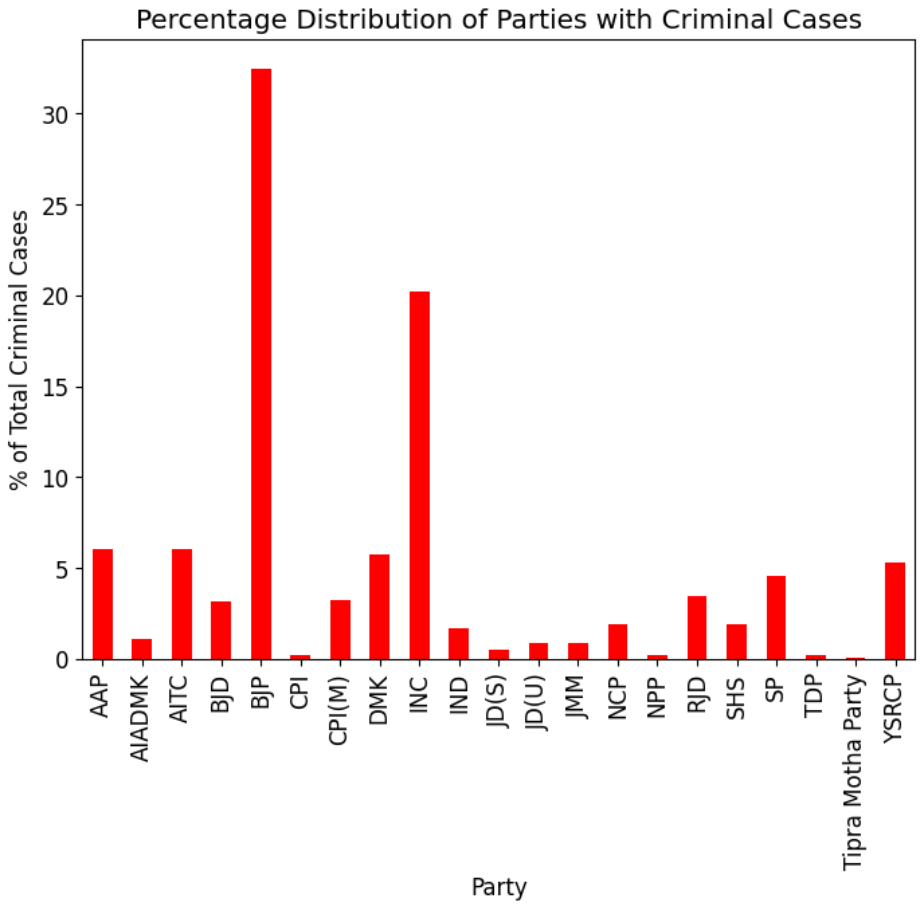

Figure 3: Percentage of distribution of parties with candidates having the most criminal records

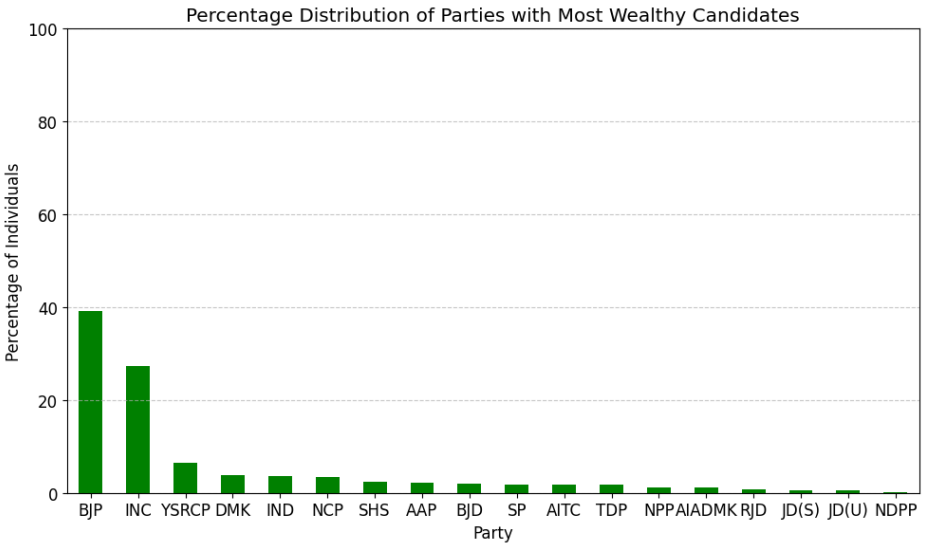

Figure 4: Percentage of distribution of parties with candidates having the most wealth

3 Results

Final F1 score on the test data is as follows:

- Public Score: 0.22542, Rank 158
- Private Score: 0.21073, Rank 93

Note that these ranks include the many deleted submissions.

4 Final Submission

All the codes are present in the following GitHub repository.
`https://github.com/harshit-784/CS253-Python-Assignment`

The documentation for libraries used here can be found at the following links:

- Matplotlib: `https://matplotlib.org/stable/contents.html`
- Pandas: `https://pandas.pydata.org/docs/`
- Numpy: `https://numpy.org/doc/stable/`
- Scikit-learn: `https://scikit-learn.org/stable/user_guide.html`
- Seaborn: `https://seaborn.pydata.org/tutorial.html`

I referred to this blog for the data preprocessing techniques: `https://www.geeksforgeeks.org/data-preprocessing-machi`

This blog helped me understand the different ML models for classification models: `https://www.datacamp.com/blog/classification-machine-learning`