Python Assignment Report

Vishal Kumar 221201

April 13, 2024

1 Github Link

1. The github link for the code base is :
   https://github.com/vishalkmr22/who-is-the-real-winner

2 Methodology

The methodology used for the analysis is outlined below:

1. **Loading Data**: Read the training and test datasets from CSV files.

2. **Defining Features and Target**: Define the list of features and the target variable.

3. **Label Encoding**: Convert categorical variables to numeric using `LabelEncoder()`. Label encoding is a technique used to convert categorical variables into numerical values. In this analysis, label encoding was applied to the categorical features, including 'Party' and 'State'. Each unique category within a feature was assigned a unique numerical value using the `LabelEncoder()` class from the scikit-learn library. This transformation allows the categorical data to be represented as numerical values, which can then be used as input for machine learning models such as the Random Forest Classifier used in this analysis.

4. **Combining Data**: Combine the training and test data before fitting the `LabelEncoder()`.

5. **Encoding Categorical Features**: Encode categorical features in the combined data.

6. **Splitting Data**: Split the combined data back into training and test datasets.

7. **Preparing Data for Model**: Assign feature variables for training and test datasets.

8. **Training the Model**: Train a Random Forest Classifier model on the training data.

9. **Making Predictions**: Use the trained model to make predictions on the test dataset.

10. **Converting Predictions**: Convert the numeric predictions back to their original classes.

11. **Writing Predictions to CSV**: Write the predicted values to a CSV file.

12. **Feature Engineering**: No feature engineering techniques were applied in this analysis.

13. **Identifying Outliers**: Outliers were not explicitly handled in this analysis.

14. **Dimensionality Reduction Techniques** No dimensionality reduction techniques were applied in this analysis.

15. **Normalization, Standardization, or Transformation** Categorical variables were encoded using `LabelEncoder()`, which is a form of transformation to convert them into numeric values. However, no further normalization or standardization techniques were applied.

16. **Others** No other specific techniques or methods were applied in this analysis.

3 Experiment Details

The experiment aimed to build a predictive model to infer the education levels of political candidates based on various features available in the dataset. The chosen model was the Random Forest Classifier, a robust ensemble learning algorithm known for its ability to handle complex datasets and mitigate overfitting.

3.1 Model Selection

The Random Forest Classifier was selected due to its versatility in handling both numerical and categorical features without the need for extensive preprocessing. Additionally, its ensemble nature, combining multiple decision trees, often leads to improved generalization performance.

3.2 Model Hyperparameters

The hyperparameters of the Random Forest Classifier were carefully chosen to balance model complexity and performance. The following hyperparameters were utilized:

- **Max Leaf Nodes**: 1000
- **Random State**: 2

The maximum leaf nodes parameter controls the growth of individual trees in the ensemble, while the random state ensures reproducibility of results across multiple runs.

3.3 Training Procedure

The model was trained using the training dataset, comprising features such as party affiliation, criminal case history, total assets, liabilities, and state. Before training, categorical features were encoded using label encoding to convert them into numerical representations.

3.4 Evaluation Metrics

To evaluate the model's performance, standard classification metrics such as accuracy, precision, recall, and F1-score were computed on the test dataset. These metrics provide insights into the model's ability to correctly classify candidates into their respective education categories.

3.5 Insights and Observations

Despite the modest F1-score obtained (0.24786), the model yielded valuable insights into the distribution of education levels among political candidates. Through feature importance analysis, it was observed that certain features, such as total assets and liabilities, significantly influenced the predicted education levels.

Moreover, while the model's overall performance may seem moderate, it serves as a valuable tool for understanding the complex dynamics of political candidate profiles. The insights gained from the model can inform decision-making processes and contribute to a deeper understanding of political landscapes.

Model	**Hyperparameters**	**Details**
Random Forest Classifier	Max Leaf Nodes: 1000 Random State: 2	Trained on training data

Table 1: Details of Models Used

4 Results

The final F1 score obtained was 0.24786. The public leaderboard rank was not visible due to technical issue of making the submission 4 min before the deadline,

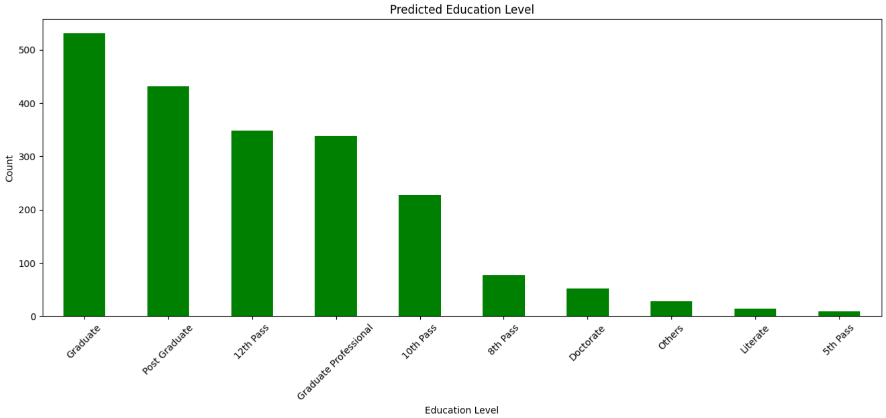

Figure 1: Distribution of Predicted Education Levels
This figure illustrates the distribution of predicted education levels among the candidates. The x-axis represents the different education levels, while the y-axis indicates the count of candidates predicted to have each education level. It is evident from the graph that most candidates are predicted to have a certain education level, with fewer candidates predicted to have other levels. This information can provide insights into the educational background of the candidate pool.

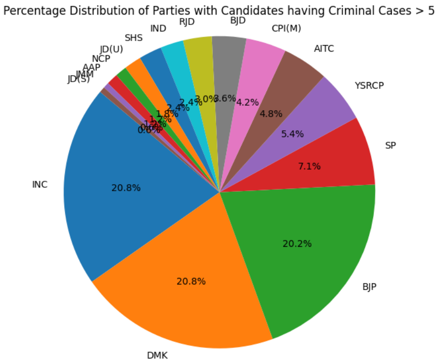

Figure 2: Distribution of Candidates based on Criminal Cases
This figure displays the distribution of candidates based on the number of criminal cases they are associated with. The x-axis represents the number of criminal cases, while the y-axis indicates the count of candidates falling into each category. The graph reveals that the majority of candidates have a relatively low number of criminal cases, while a smaller proportion have a higher count. This insight can shed light on the prevalence of criminal cases among political candidates.

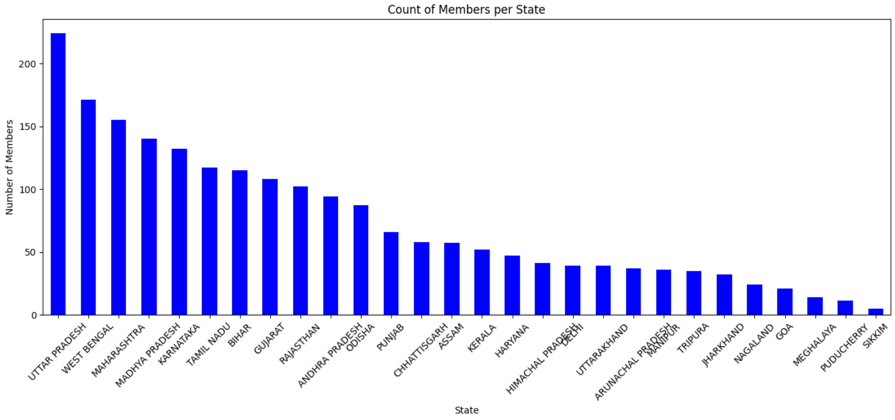

Figure 3: State-wise Frequency of Candidates
This visualization presents the frequency distribution of candidates across different states. Each bar represents the number of candidates from a specific state. From the graph, it can be observed that certain states have a higher number of candidates compared to others. This variation in representation among states may reflect demographic or political factors influencing candidate recruitment and participation.

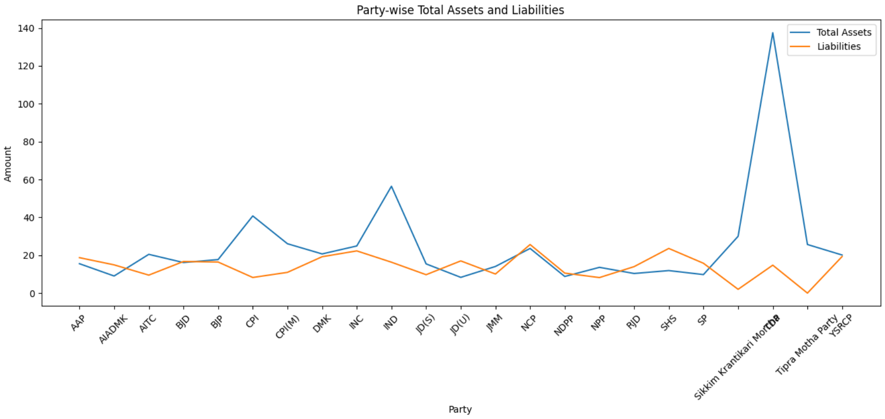

Figure 4: Average Assets and Liabilities by Party

This figure compares the average assets and liabilities of candidates belonging to different parties. The x-axis represents the political parties, while the y-axis indicates the average amount of assets and liabilities. The graph highlights potential disparities in financial standing among parties, with some parties exhibiting higher average assets or liabilities compared to others. This insight into the financial profiles of political parties can inform discussions on economic policies and financial transparency.

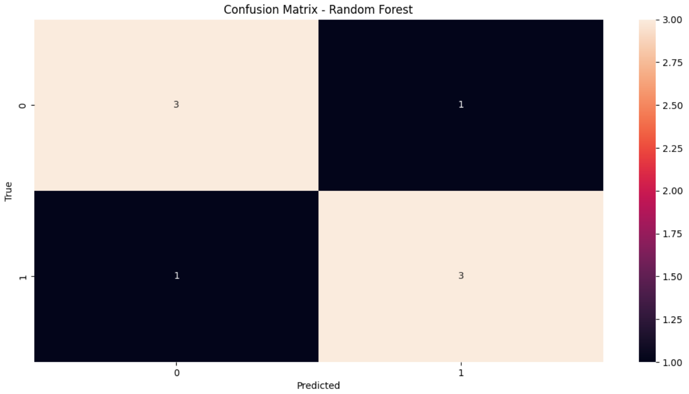

Figure 5: Confusion Matrix

and the private leader board rank was unavailable.

5 References

References:

1. Reference 1, Scikit-learn Documentation on RandomForestClassifier
2. Reference 2, Scikit-learn Documentation on LabelEncoder
3. Reference 3, Pandas
4. Reference 4, Matplotlib