# Python Assignment Report

Debraj Karmakar Roll no: 220329

10th April, 2024

## 1 Methodology

### 1.1 Data Processing Steps

The fields given in the train dataset were ID, Candidate, Constituency, Party, Criminal Case, Total Assets, Liabilities, state and Education.
Out of these fields, the fields ID, Candidate, Constituency, Party, Criminal Case, Total Assets, Liabilities and state were to be used as input to the model and the field Education was to be used as the output.

Most of the fields were categorical in nature. For efficient input to the model, the significant fields were converted to one-hot encoded vectors and other non-significant fields were dropped after taking some specific features from them.

The following steps were taken to process the data:

1. ID field: The field ID was dropped as it was insignificant to the model.
2. Candidate field: The candidate field contained the names of the candidates, which were unique for each candidate. However, some particular features were extracted from this field and then it was dropped. Details about the features are given in the next section.
3. Constituency field: The Constituency field contained the names of the constituencies, which were also very much unique. So, some particular features were extracted from this field and then it was dropped. Details about the features are given in the next section.
4. Party field: The Party field contained the names of the parties to which the candidates belonged. Since number of distinct parties was small, this field was converted to a one-hot encoded vector.
5. Criminal Case field: The Criminal Case field contained numerical value. So, it was kept as it is.
6. Total Assets field: The Total Assets field contained asset values followed by a string $'Crore+'$ or $'Lac+'$. The asset values were converted to its value in crore and the string was dropped.
7. Liabilities field: The Liabilities field also contained liability values followed by a string $'Crore+'$ or $'Lac+'$. The liability values were converted to its value in crore and the string was dropped.
8. State field: The State field contained the names of the states to which the constituencies belonged. Since number of distinct states was small, this field was converted to a one-hot encoded vector.
9. Education field: The Education field contained the education qualification of the candidates. Since the number of distinct education qualifications was small, this field was converted to a one-hot encoded vector.

### 1.2 Feature Engineering

The following features were extracted from the Candidate and Constituency fields:

1. Candidate field: Some people's name have a prefix like $'Dr.'$, $'Adv.'$ and $'Prof.'$, which gave information about their academic qualification. So, the presence of these prefixes were checked and the such prefixes were stored as one-hot encoded vectors.

2. Constituency field: The constituency name also contained the caste or race of some people, which might give information about their education. So, the presence of such caste was checked and the caste were stored as one-hot encoded vectors.

### 1.3 Increasing Representation of Minority Classes

We have multiplied the minority classes by a suitable factor to increase their representation in the dataset. This was done to prevent the model from being biased towards the majority classes.

### 1.4 Dimensionality Reduction Techniques

Most of the attributes of the training dataset was converted into one-hot encoded vectors. Since the number additional attributes added by one-hot encoding is the number of distinct values of the original attribute, the dimensionality of the dataset increased significantly.
In order to reduce the dimensionality of the dataset, attributes that had the same value for most of the samples were dropped. This was done because such attributes do not provide any information to the model.

The following changes were done to reduce the dimensionality of the dataset:

1. In the training dataset, out of 2059 entries, 1595 entries did not have any information regarding their cast. Hence, after extracting the caste from the constituency field, the field with no caste information was dropped.

2. Out of 2059 entries of the training data, 1986 entries did not have any information regarding the presence of $'Dr.'$, $'Adv.'$ or $'Prof.'$ in their name. Hence, after extracting the presence of these prefixes from the candidate field, the field with no such prefixes was dropped.

3. The ID field was not related to the output and was dropped.

### 1.5 Normalization, Standardization or Transformation used

We have tried using MinMaxScaler to scale the numerical attributes to the range $[0, 1]$. This was done in order to prevent the model from being biased towards the attributes with higher values. However, the model performed better without scaling the numerical attributes. Hence, we have not used any scaling technique.

## 2 Experiment Details

### 2.1 Model Used

The following models were used to predict the education qualification of the candidates:

1. Linear Regression: On using Linear Regression, the model gave an f1 score of about 0.20, which suggested that education level was not linearly dependent on the input features.

2. State Vector Machine: This model gave an f1 score of about 0.18, which was not better than the Linear Regression model.

3. K Nearest Neighbors: This model gave an f1 score of about 0.20, which was also not better.

4. XGBoost: This model gave an f1 score of about 0.15, which was also not better.

5. Random Forest: This model gave an f1 score of about 0.23, which was better than the Linear Regression model.

From the above results, it was observed that **Random Forest** model performed better than the other models. Hence, we have used Random Forest model for the final prediction.

## 2.2 Tuning Parameters

The parameters for the random forest model were tuned using **GridSearchCV**. The best tuned parameters for this model were:

- n_estimators: 275
- max_depth: 1000
- min_samples_split: 10
- min_samples_leaf: 1
- cv: 10
- verbose: 2

The link for the code of Random Forest Implementation is: here

## 2.3 Data Insights

The following insights were observed from the data:

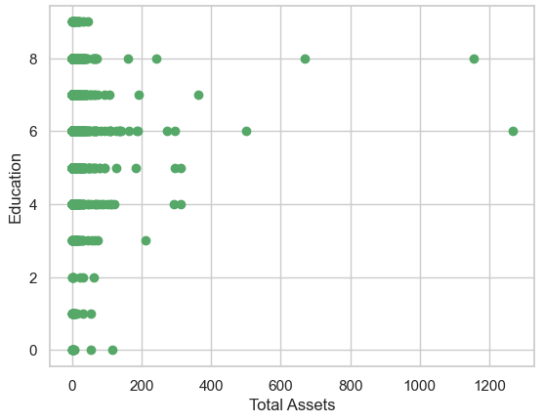


Figure 1: Education vs Total Assets plot

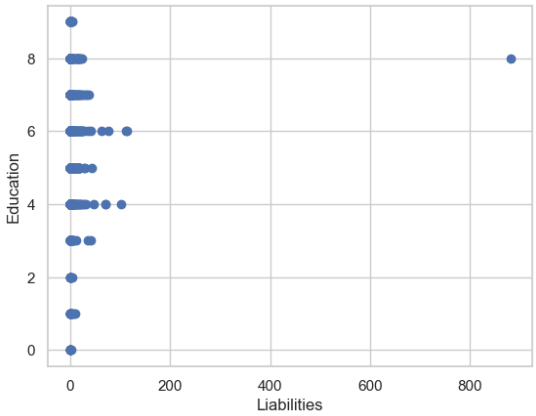


Figure 2: Education vs Liabilities plot

The plots in Figure 1 and Figure 2 show that people with higher education qualification have higher assets. Also, we should note that people with higher assets tend to have higher liabilities. This suggests that education qualification is dependent on the total assets and liabilities of the candidates.

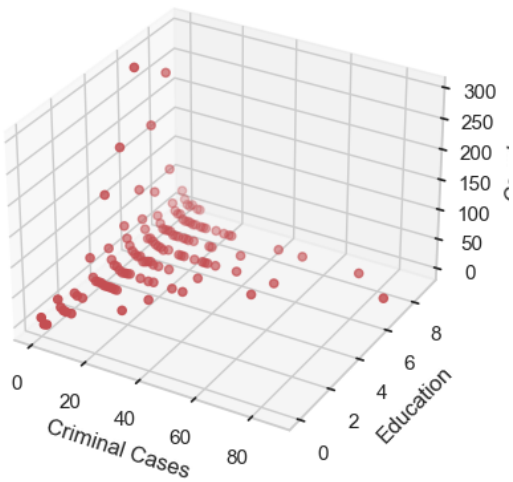


Figure 3: Count of people vs Education, Criminal Case

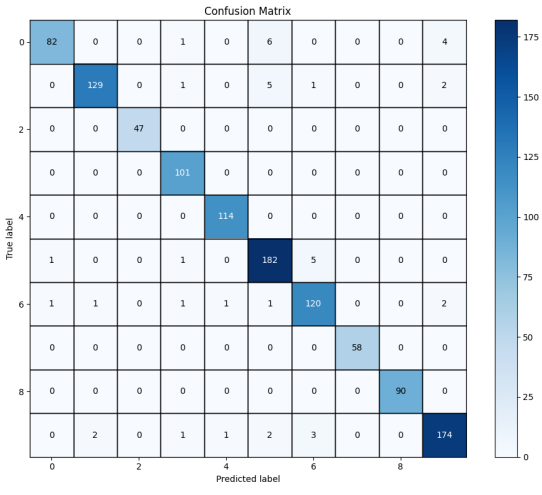


Figure 4: Confusion Matrix for 20% of the training data

The plot in Figure 3 shows that although most people have lower number of criminal cases, people with high number of criminal cases tend to have higher educational qualification.
The confusion matrix in Figure 4 shows that the model is able to predict the education qualification of the candidates with a good accuracy for 20% of the training data.

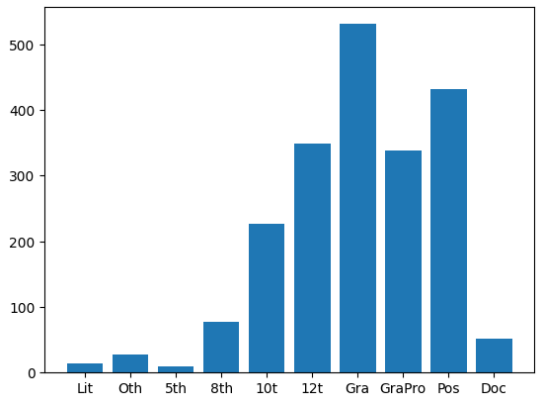


Figure 5: Distribution of people before accounting for minority classes

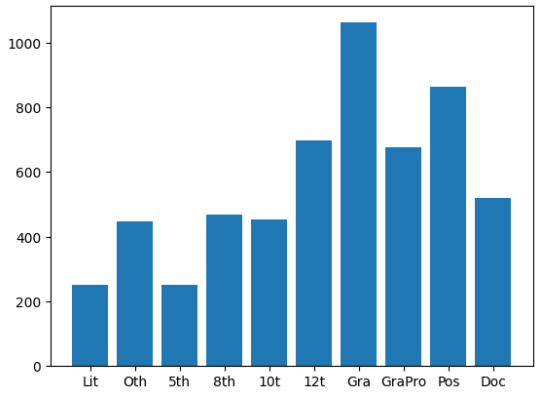


Figure 6: Distribution of people after accounting for minority classes

## 3 Results

The F1 score of the model on the test data is -

1. Public Score: F1 score = 0.31784 and Rank 6
2. Private Score: F1 score = 0.24190 and Rank 75

## 4 References

- Random Forests blog
- Feature Selection Techniques
- Hyperparameter Tuning
- One Hot Encoding
- Other than these references, many other websites and videos on youtube were refered to understand the concepts and implement the code.