Machine Learning Business Report

V1.0

Prepared by: Lavanya Sreeram

Prepared for: GL Batch Nov 22

May 28, 2023

Document Information

Author:	Lavanya Sreeram
Document Title	Lavanya_Sreeram_28_05_2023
Issue Date:	28-05-2023
Version:	V 1.0
Subject:	Machine Learning Business Report by Lavanya Sreeram

Revision History

Version	**Date**	**Revised By**	**Reason**
1.0	28-05-2023	Lavanya Sreeram	First Issue

Contents

Figures:

Tables:

None

1 PROBLEM 1: ENSEMBLE TECHNIQUES

You are hired by one of the leading news channels CNBE who wants to analyze recent elections. This survey was conducted on 1525 voters with 9 variables. You have to build a model, to predict which party a voter will vote for on the basis of the given information, to create an exit poll that will help in predicting overall win and seats covered by a particular party.

DATA DESCRIPTION

Election_Data.xlsx data set is provided. Data Dictionary for Election_Data is as below:

1. vote: Party choice: Conservative or Labour

2. age: in years

3. economic.cond.national: Assessment of current national economic conditions, 1 to 5.

4. economic.cond.household: Assessment of current household economic conditions, 1 to 5.

5. Blair: Assessment of the Labour leader, 1 to 5.

6. Hague: Assessment of the Conservative leader, 1 to 5.

7. Europe: an 11-point scale that measures respondents' attitudes toward European integration. High scores represent 'Eurosceptic' sentiment.

8. political.knowledge: Knowledge of parties' positions on European integration, 0 to 3.

9. gender: female or male.

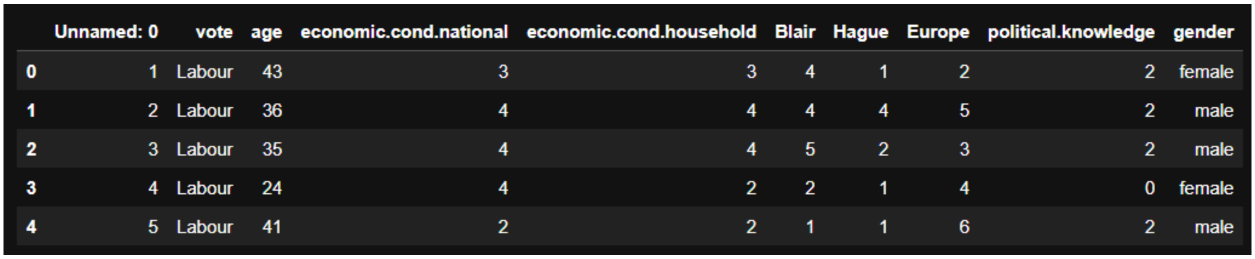
1.1 EXPLORATORY DATA ANALYSIS

Read the dataset. Describe the data briefly. Interpret the inferences for each. Initial steps like head() .info(), Data Types, etc . Null value check, Summary stats, Skewness must be discussed.

Answer:

1.1.1 Sample Data

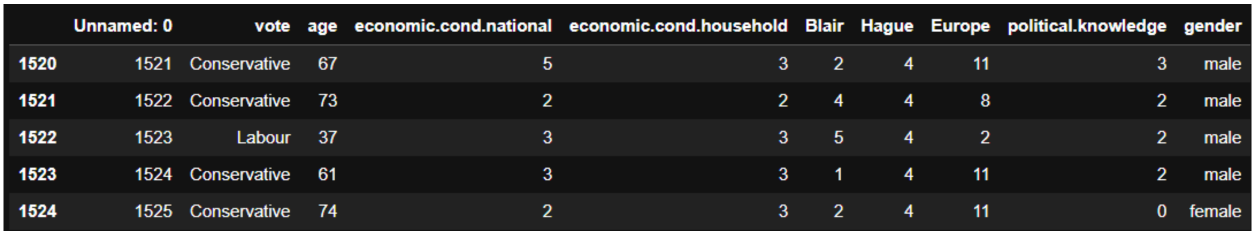
The sample Dataset has 1525 rows, 10 columns. 'vote is the Target Variable. The below snap sho ws the sample dataset first 5 observations.



	Unnamed: 0	vote	age	economic.cond.national	economic.cond.household	Blair	Hague	Europe	political.knowledge	gender
0	1	Labour	43	3	3	4	1	2	2	female
1	2	Labour	36	4	4	4	4	5	2	male
2	3	Labour	35	4	4	5	2	3	2	male
3	4	Labour	24	4	2	2	1	4	0	female
4	5	Labour	41	2	2	1	1	6	2	male

Fig1.1.1 Election Data first 5 observations – as given

The below snap shows the bottom 5 observations.



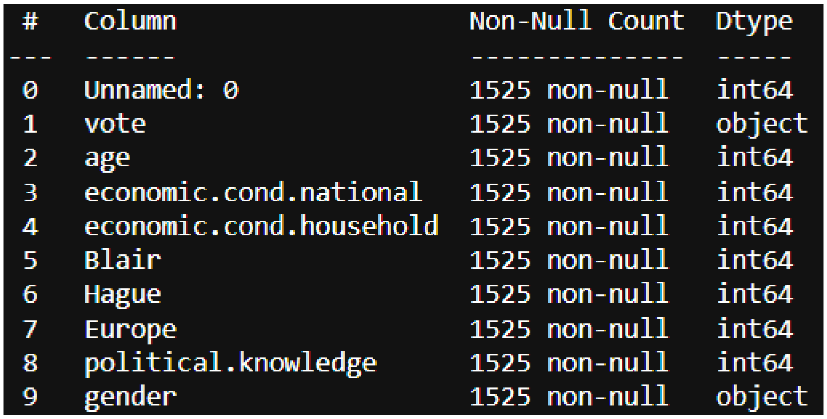
	Unnamed: 0	vote	age	economic.cond.national	economic.cond.household	Blair	Hague	Europe	political.knowledge	gender
1520	1521	Conservative	67	5	3	2	4	11	3	male
1521	1522	Conservative	73	2	2	4	4	8	2	male
1522	1523	Labour	37	3	3	5	4	2	2	male
1523	1524	Conservative	61	3	3	1	4	11	2	male
1524	1525	Conservative	74	2	3	2	4	11	0	female

Fig1.1.2 ElectionData bottom 5 observations – as given

On first look, it can be inferred that the feature 'Unnamed: 0' must be dropped. Additionally, it can be noted that the data is not scaled.

1.1.2 Info & Summary

The below snap shows the datatypes of all features of the dataset.



```
 #   Column                   Non-Null Count  Dtype 
---  ------                   --------------  ----- 
 0   Unnamed: 0               1525 non-null   int64 
 1   vote                     1525 non-null   object
 2   age                      1525 non-null   int64 
 3   economic.cond.national   1525 non-null   int64 
 4   economic.cond.household  1525 non-null   int64 
 5   Blair                    1525 non-null   int64 
 6   Hague                    1525 non-null   int64 
 7   Europe                   1525 non-null   int64 
 8   political.knowledge      1525 non-null   int64 
 9   gender                   1525 non-null   object
```

(Fig1.1.3): Election Data datatypes - initial

Except 'vote', 'gender' all the features in the data are numeric in nature ('int64'). 'vote is object type, and as per data dictionary the feature is either 'Labour' or 'Conservative'. 'vote' shall be encoded. 'gender' as per data dictionary is binary i.e. 'female' or 'male'. 'gender' observations can be encoded.

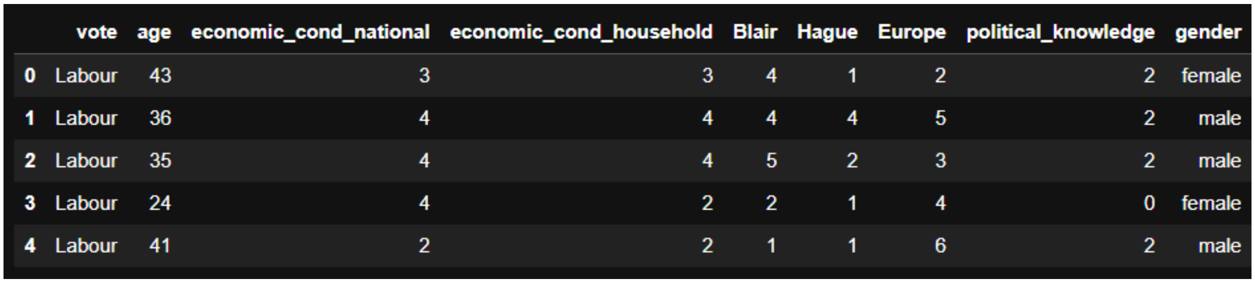
Drop ‘Unnamed:0’ feature. Rename the columns that contain punctuation ‘.’ With below new column names:

'economic.cond.national'-> 'economic_cond_national'

'economic.cond.household' -> 'economic_cond_household'

'political.knowledge' ->'political_knowledge'

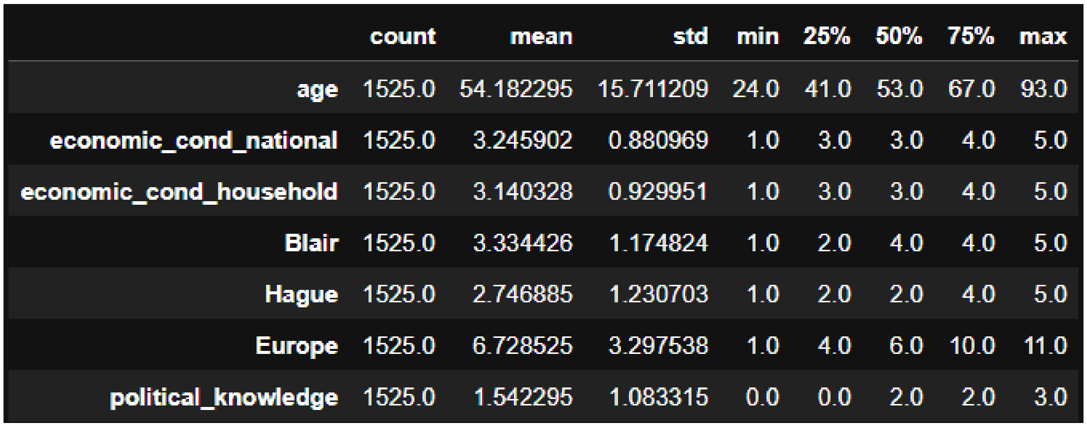
Now, the dataset top 5 observations are as shown below:



	vote	age	economic_cond_national	economic_cond_household	Blair	Hague	Europe	political_knowledge	gender
0	Labour	43	3	3	4	1	2	2	female
1	Labour	36	4	4	4	4	5	2	male
2	Labour	35	4	4	5	2	3	2	male
3	Labour	24	4	2	2	1	4	0	female
4	Labour	41	2	2	1	1	6	2	male

(Fig 1.1.4): Election Data after dropping ‘Unnamed:0’ feature & renaming columns

The statistical data descriptions is as shown in below snap.

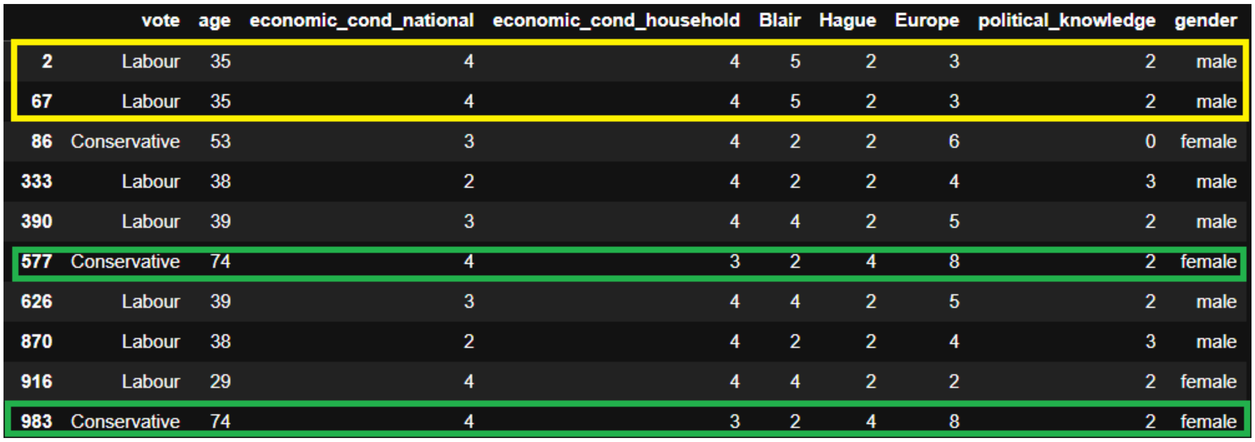


	count	mean	std	min	25%	50%	75%	max
age	1525.0	54.182295	15.711209	24.0	41.0	53.0	67.0	93.0
economic_cond_national	1525.0	3.245902	0.880969	1.0	3.0	3.0	4.0	5.0
economic_cond_household	1525.0	3.140328	0.929951	1.0	3.0	3.0	4.0	5.0
Blair	1525.0	3.334426	1.174824	1.0	2.0	4.0	4.0	5.0
Hague	1525.0	2.746885	1.230703	1.0	2.0	2.0	4.0	5.0
Europe	1525.0	6.728525	3.297538	1.0	4.0	6.0	10.0	11.0
political_knowledge	1525.0	1.542295	1.083315	0.0	0.0	2.0	2.0	3.0

(Fig1.1.5): Election Data statistical description -initial

1.1.3 Null & Duplicate Values

A null value check is performed. There are no null values in the dataset provided. Duplicated data is checked and there are 8 duplicate observations. The duplicates sample are shown below:



	vote	age	economic_cond_national	economic_cond_household	Blair	Hague	Europe	political_knowledge	gender
2	Labour	35	4	4	5	2	3	2	male
67	Labour	35	4	4	5	2	3	2	male
86	Conservative	53	3	4	2	2	6	0	female
333	Labour	38	2	4	2	2	4	3	male
390	Labour	39	3	4	4	2	5	2	male
577	Conservative	74	4	3	2	4	8	2	female
626	Labour	39	3	4	4	2	5	2	male
870	Labour	38	2	4	2	2	4	3	male
916	Labour	29	4	4	4	2	2	2	female
983	Conservative	74	4	3	2	4	8	2	female

Fig1.1.3.1 Election Data Duplicated values

Row 2,767 have exact same value for all features except that they belong to different person. Similar is the case with pairs of 577,983. These duplicates need to be dropped because they do not add any value to the study, be it associated with different people. These duplicates are dropped.

The new shape of the dataset is 1517 rows, 9 columns.

1.1.4 Univariate Analysis

Univariate analysis on the dataset.
The countplot for ‘vote’ is shown below.

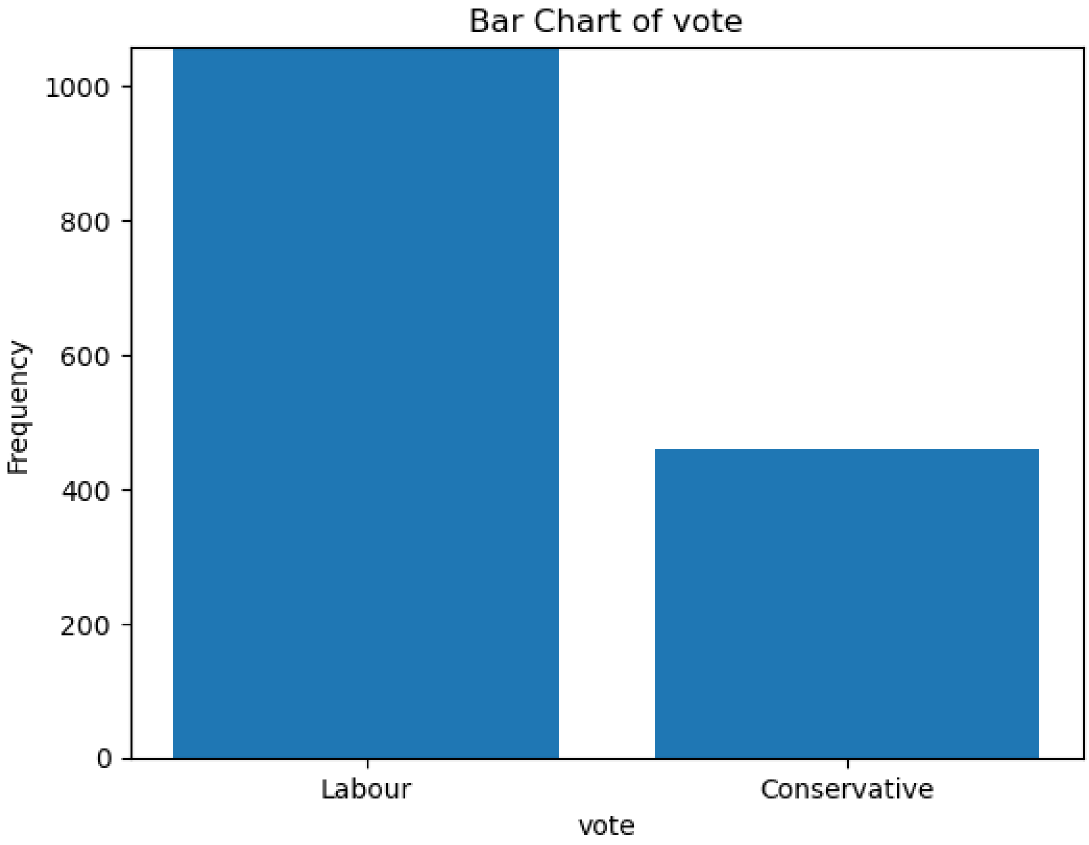


Fig1.1.4.1 Election Data ‘vote’ barchart

Above plot shows that the ‘vote’ i.e. the target variable has much higher counts for ‘Labour’ than ‘Conservative’. Hence, while splitting data stratify parameter shall be used.

Bar chart for ‘age’ feature is plotted and the plot shows that there is a very slight positive/right s kewness in age distribution. This means that the distribution of ages is slightly skewed towards t he higher values.

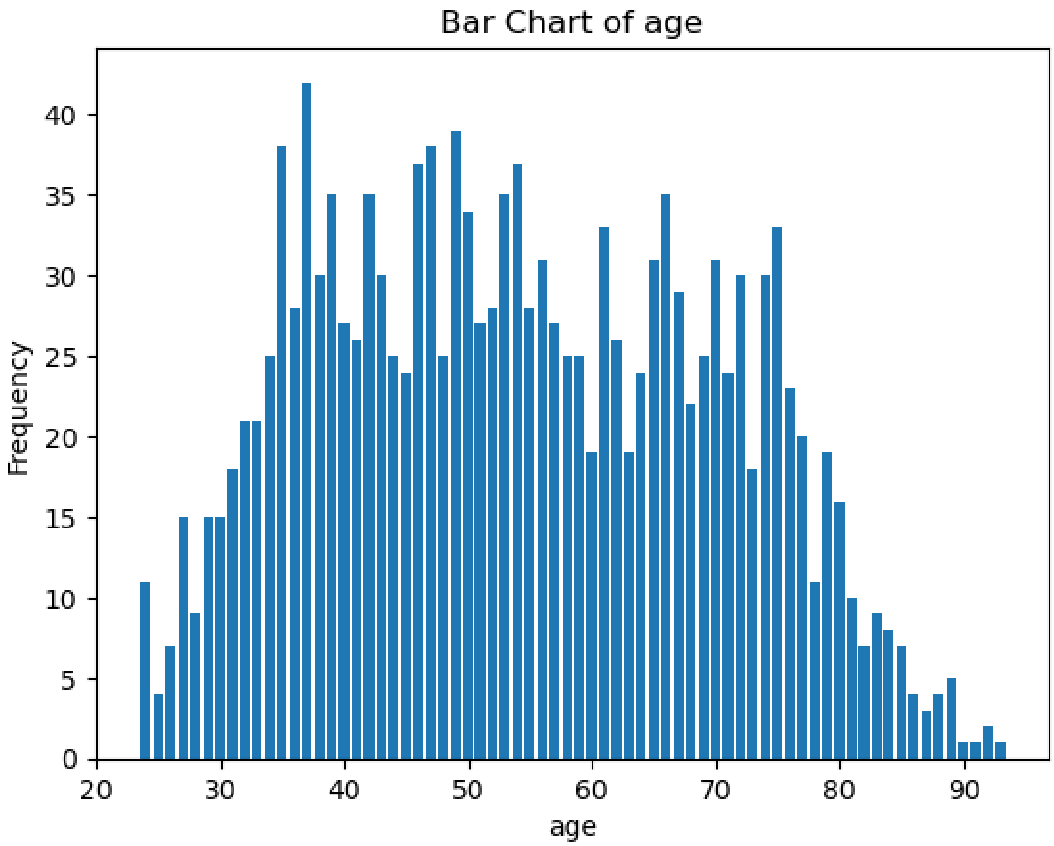
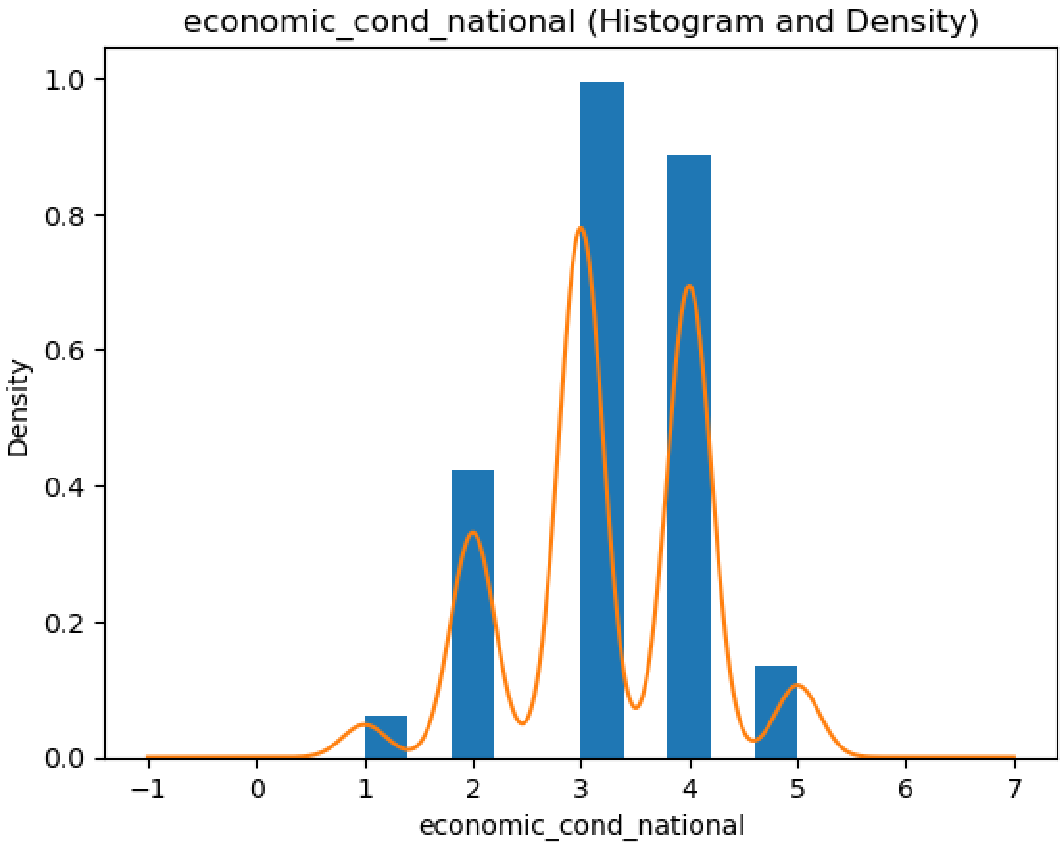
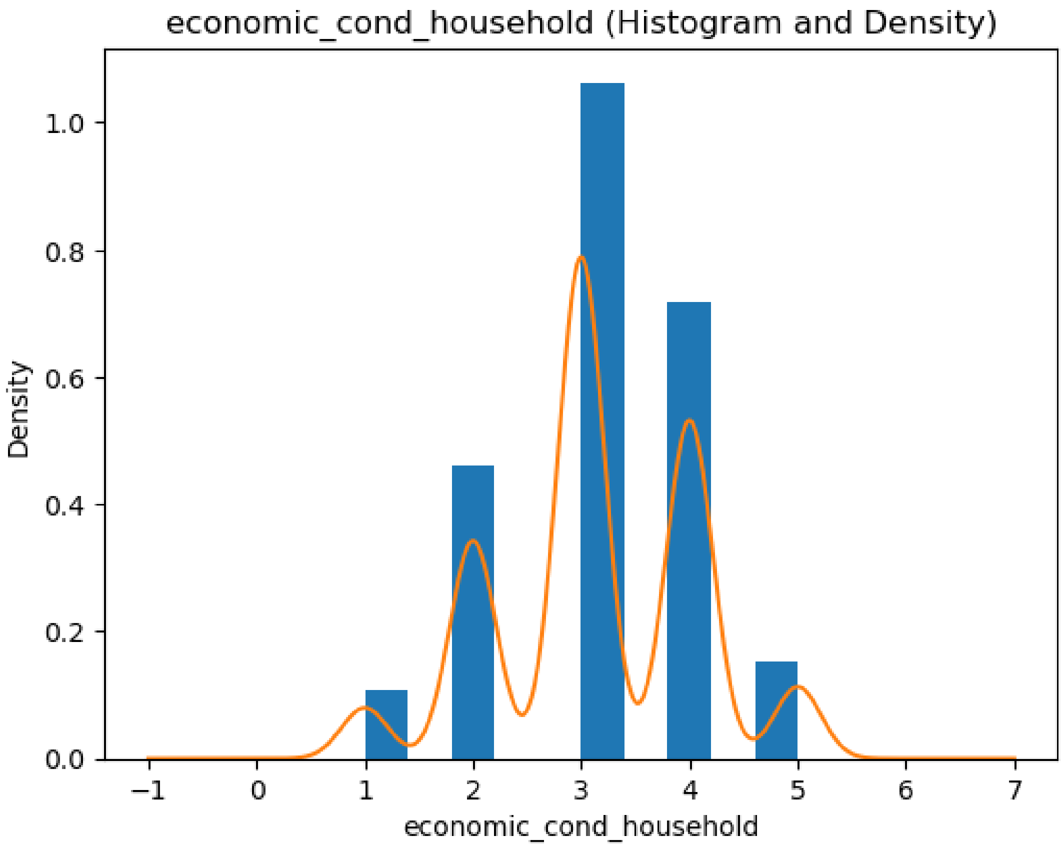


Fig1.1.4.2 Election Data ‘age count plot

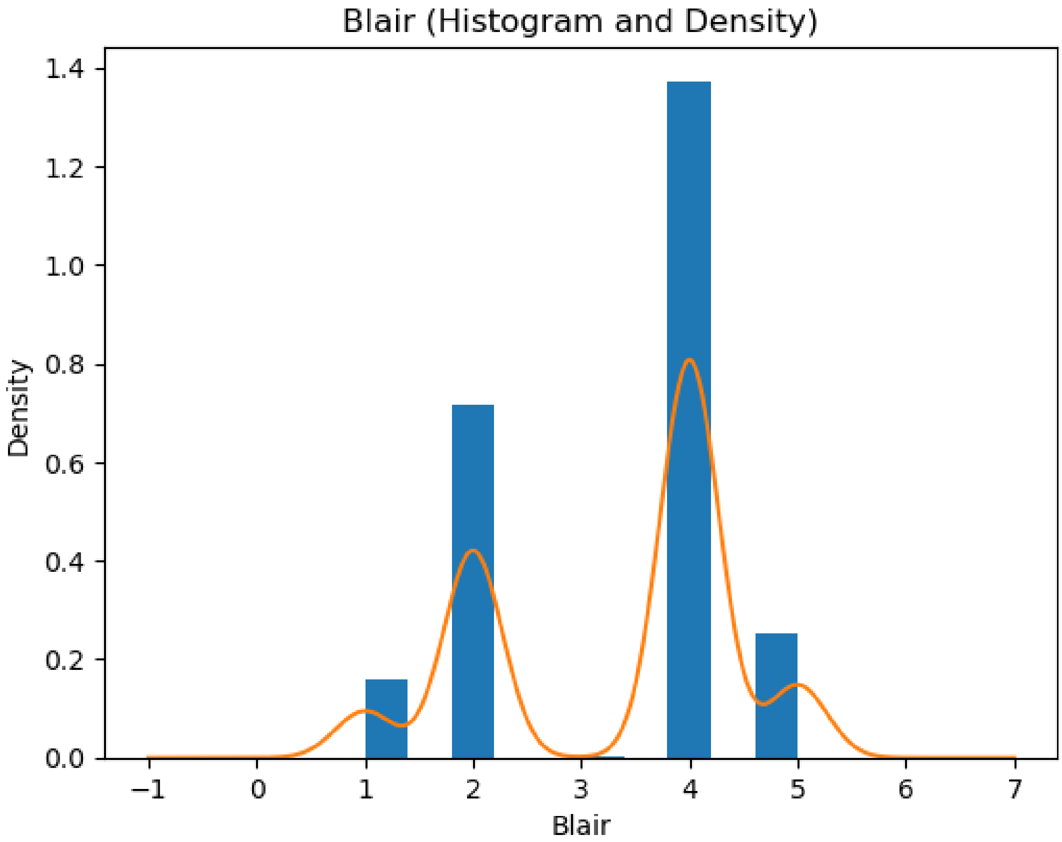
Histograms are plotted for the below features in the data.



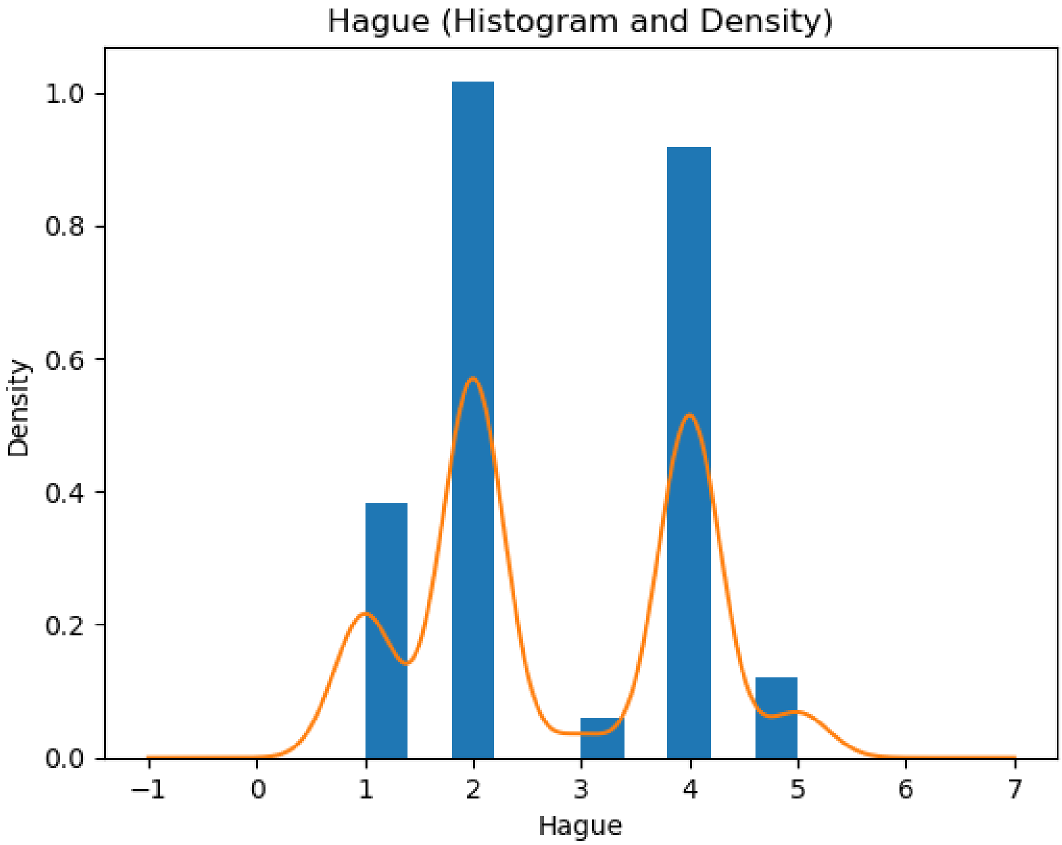
It is not evident with the density plot but with the histogram we can notice that the data is left skewed or has negative skew i.e the distribution of observations for the economic condition at the national level is slightly skewed towards lower values.



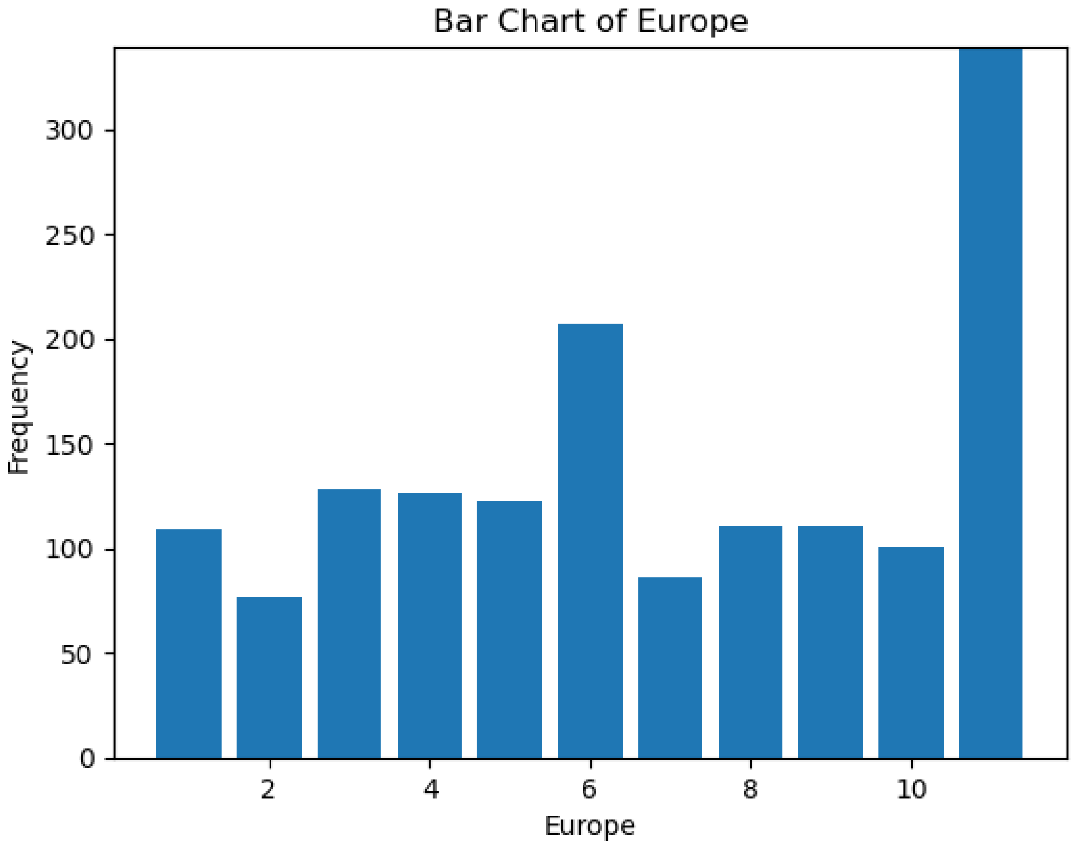
Once again with the ‘economic_cond_household’ feature there is a slight negative/let skewedness indicating that the distribution is slightly skewed toward the lower values.



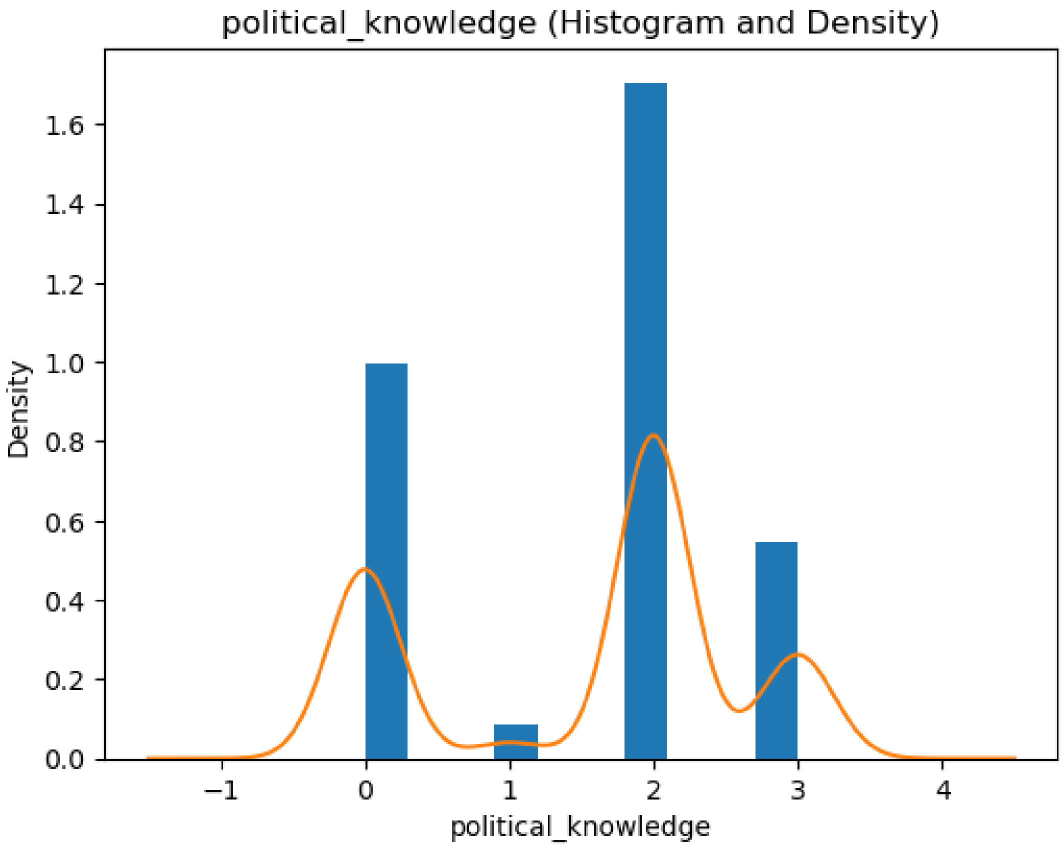
The distribution of observations for ‘Blair’ is skewed towards lower values meaning data is left skewed or has negative skew.



The value for ‘Hague’ is right or positively skewed. This means that the distribution of observations for Hague is slightly skewed towards higher values.



The distribution of observations for ‘Europe’ is slightly negatively or left skewed with distribution skewed toward lower values.



The distribution of observations for ‘political_knowledge’ is slightly negatively or left skewed with distribution skewed toward lower values.

The above histograms show that the data is slightly skewed. The magnitudes of the skewness values in these cases are relatively small, suggesting that the skewness is not extreme.

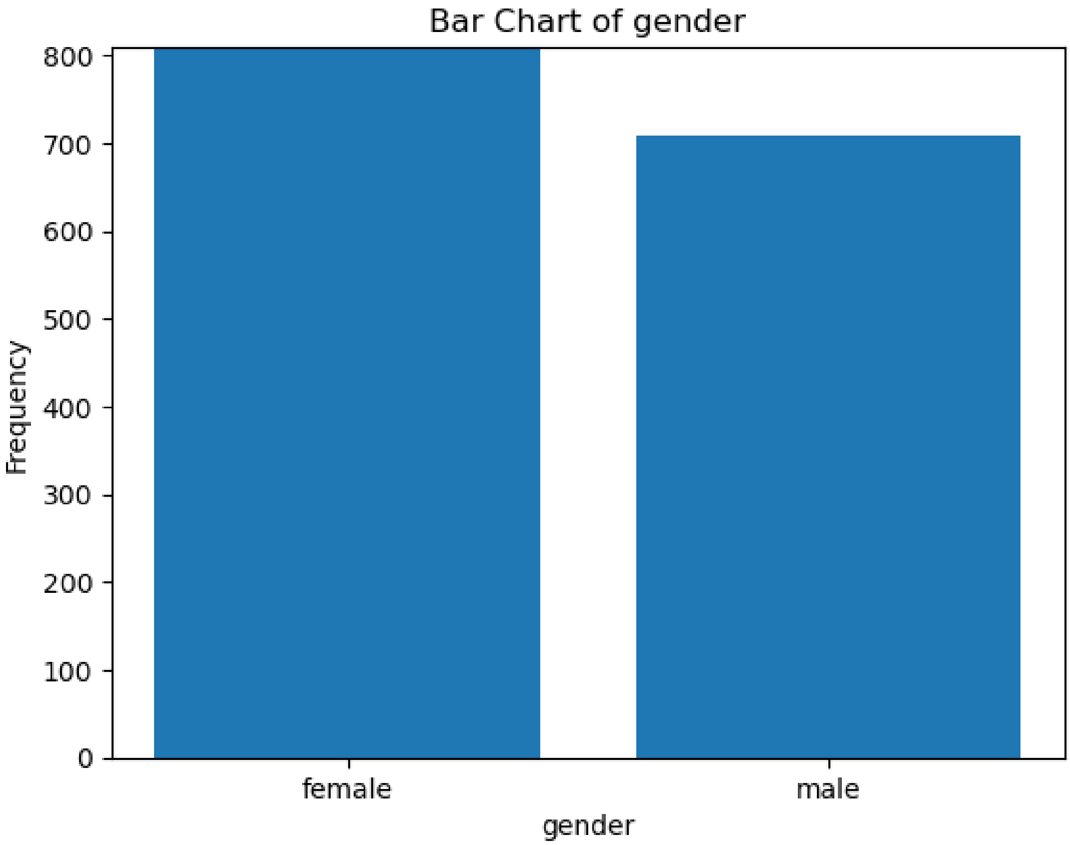


Fig1.1.4.3 Election Data univariate plots

From the above plot, it can be noted that the female observations are slightly more than male observations in the dataset.

1.1.5 Bivariate Analysis

Bivariate analysis is performed on the data. The Gender vs vote Bar plot explores the way in which different gender voted. Female gender voted slightly more than the male gender. Both gender predominantly voted ‘Labour’ over ‘Conservative’ in this dataset.

The plot is shown below.

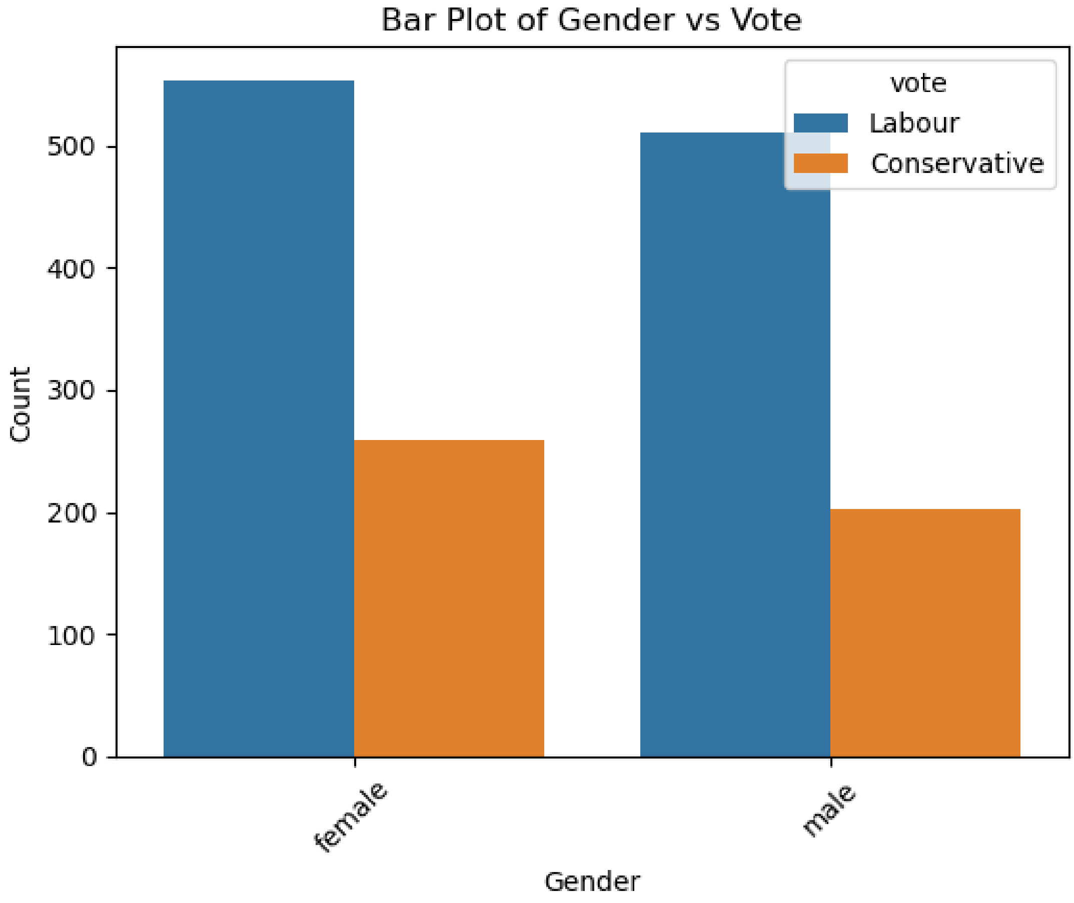
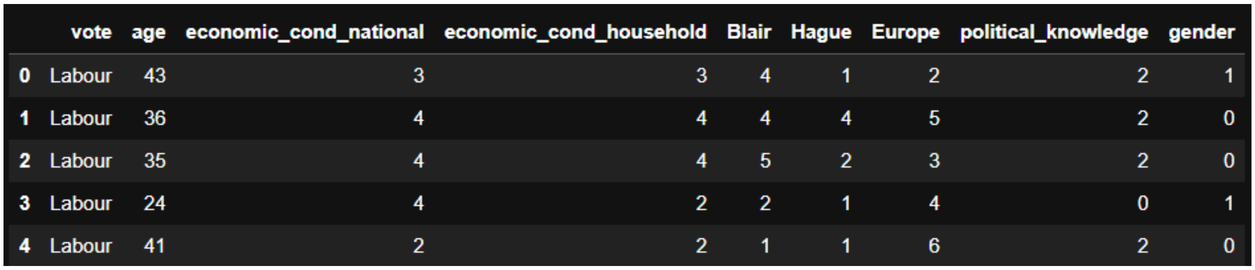


Fig 1.1.5.1 Election Data 'gender' vs 'vote' plots

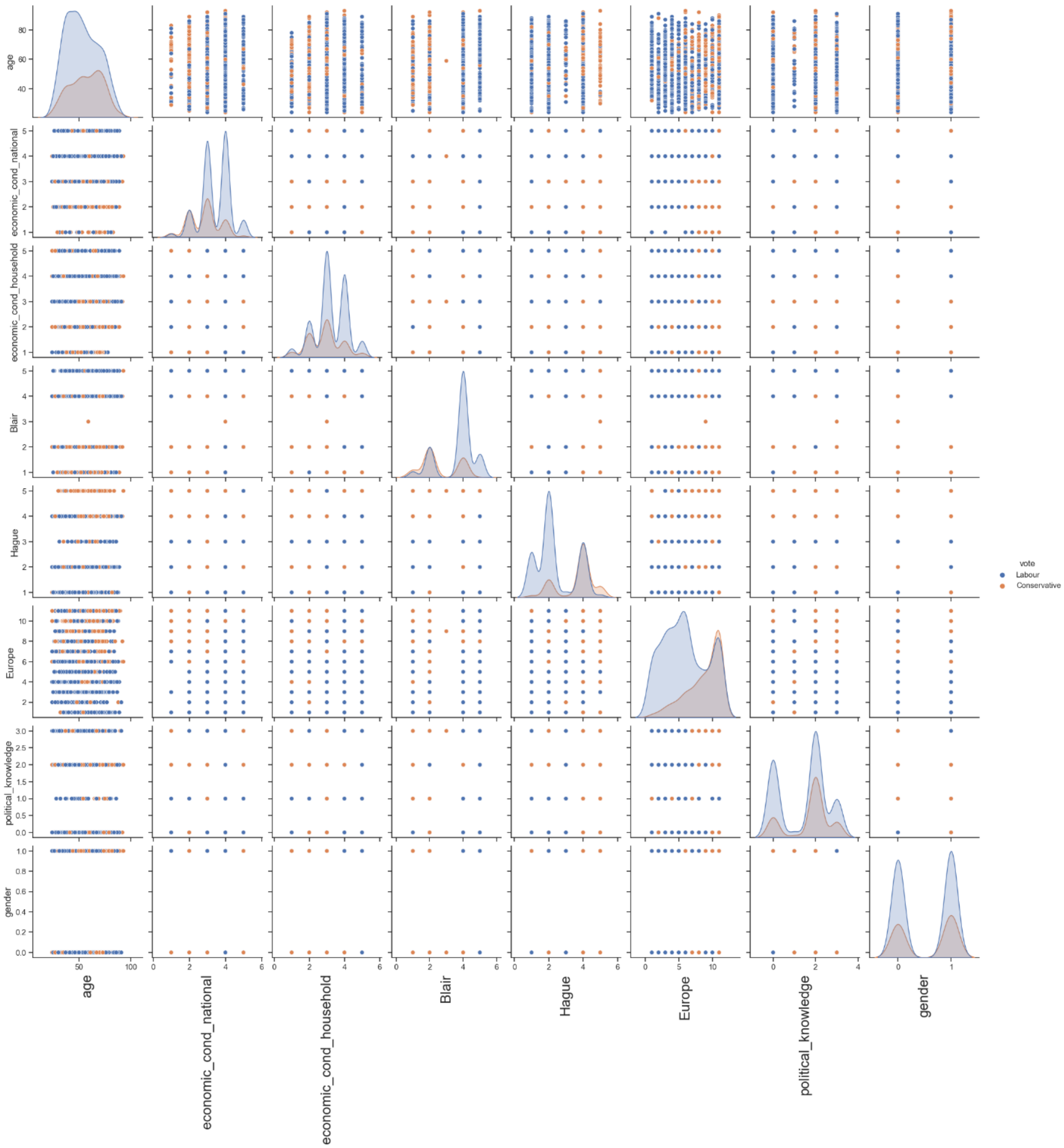
The 'gender' feature is categorical and can be encoded with binary values. The binary values considered for encoding are 'female' is 1, 'male' is 0. Encoding is performed. The feature is converted to int64. The sample dataset after encoding is shown below:



	vote	age	economic_cond_national	economic_cond_household	Blair	Hague	Europe	political_knowledge	gender
0	Labour	43	3	3	4	1	2	2	1
1	Labour	36	4	4	4	4	5	2	0
2	Labour	35	4	4	5	2	3	2	0
3	Labour	24	4	2	2	1	4	0	1
4	Labour	41	2	2	1	1	6	2	0

Fig 1.1.5.2 Election Data after 'gender' encoding

The pair plot of the dataset is shown below, using 'vote' value as hue:



(Fig 1.1.5.3): Election Data – Pairplot

The correlation heatmap between the independent variables is plotted. The heatmap is shown below.

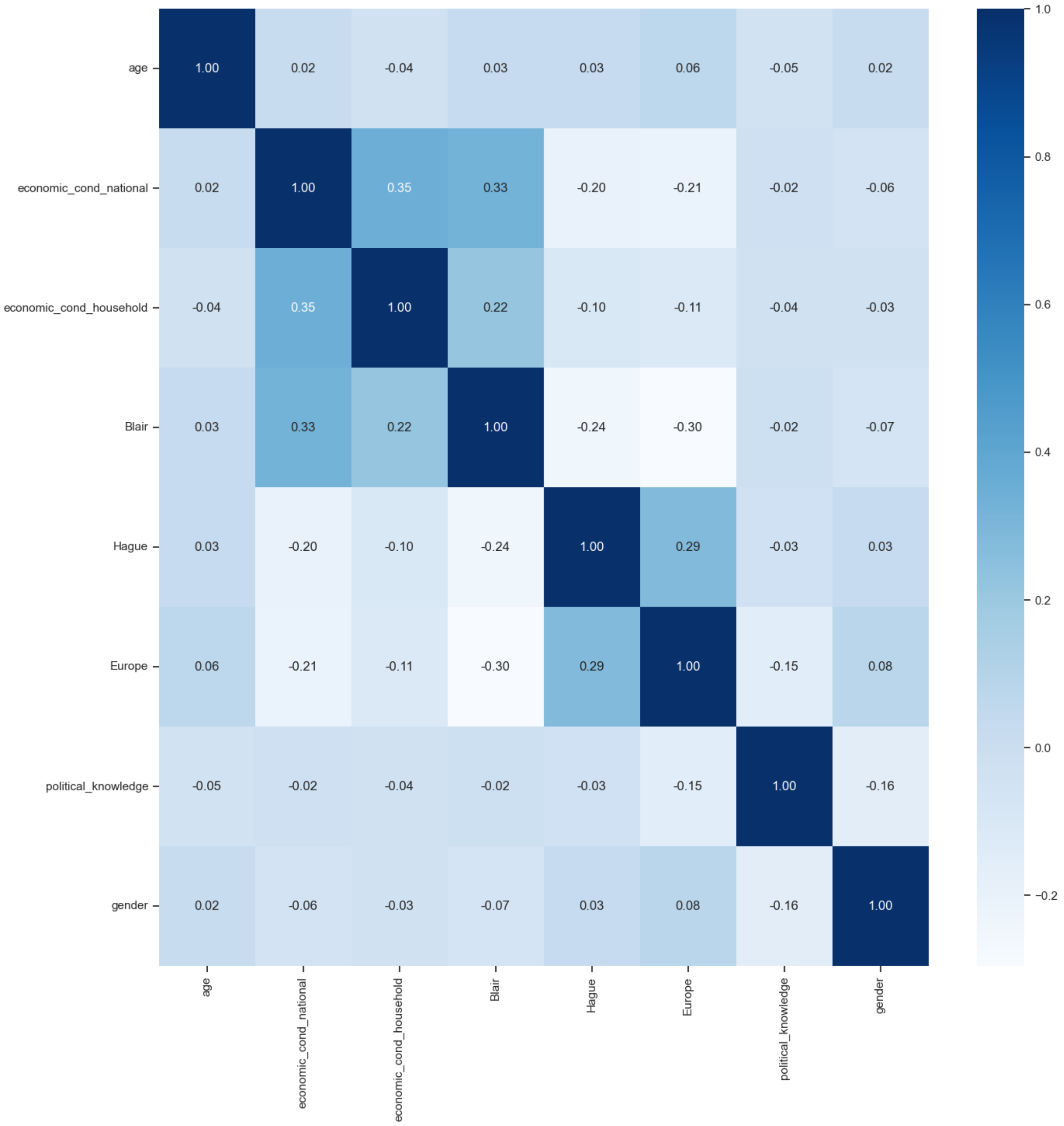


Fig 1.1.5.4 Heatmap – of independent variables - before outliers

1. Age has a very weak positive correlation (0.0186) with economic_cond_national, indicating a slight relationship between age and the perception of national economic conditions. Aged individuals are more slightly likely to vie conditions as positive/doing good.
2. Economic_cond_national has a moderate positive correlation (0.348) with economic_cond_household, suggesting that individuals who perceive national economic

conditions positively also tend to perceive their household economic conditions positively.

3. Blair has a moderate positive correlation (0.326) with economic_cond_national, indicating that individuals who support Blair's party (Labour) are more likely to perceive positive national economic conditions.
4. Hague has a weak negative correlation (-0.201) with economic_cond_national, suggesting that individuals who support Hague's party (Conservative) are less likely to perceive positive national economic conditions.
5. Europe has a weak negative correlation (-0.209) with economic_cond_national, implying that individuals who support European integration are less likely to perceive positive national economic conditions.
6. Political_knowledge has a weak negative correlation (-0.157) with gender, suggesting that individuals with higher political knowledge are slightly less likely to be male.
7. Political_knowledge has a weak positive correlation (0.076) with Europe, indicating that individuals with higher political knowledge are slightly more likely to support Europe sentiment.

! We should remember co-relation does not mean causation. For more insights a detailed analysis of data should be done.

There are several moderate co-relations between independent variables and this should be treated. Here are few suggetsions:

1. 'Hague' & 'Blair' features carry opposing information, and these features have more weight on target variable as these features are capturing information on the Party leader. 'Blair' & 'hague' need to be analysed and a new feature can be created – drop Blair & Hague.
2. economic_cond_national & economic_cond_household also are co-related, individuals carry similar sentiment about both.

1.2 DATA PREPARATION

Perform EDA (Check the null values, Data types, shape, Univariate, bivariate analysis). Also check for outliers (4 pts). Interpret the inferences for each (3 pts) Distribution plots(histogram) or similar plots for the continuous columns. Box plots. Appropriate plots for categorical variables. Inferences on each plot. Outliers proportion should be discussed, and inferences from above used plots should be there. There is no restriction on how the learner wishes to implement this but the code should be able to represent the correct output and inferences should be logical and correct.

Answer:

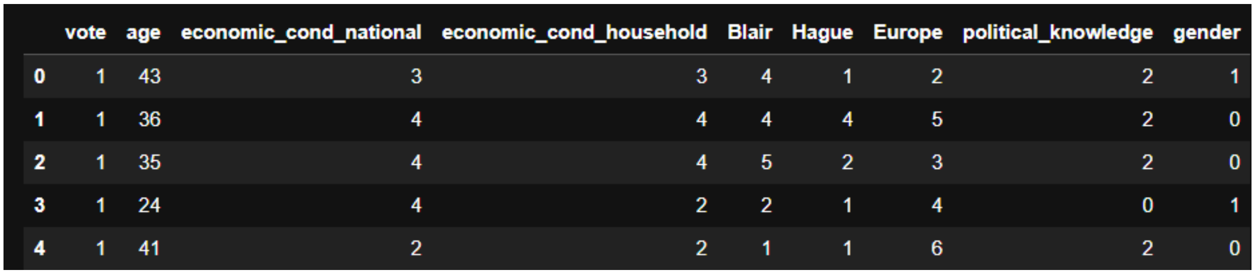
1.2.1 Null Values

There are no values in the data. Duplicate observations have been treated.

1.2.2 Encode Data

The ‘gender’ feature and ‘vote’ are categorical features have categorical values and can be converted to binary values. ‘Gender’ is converted to binary in above section. The ‘vote’ feature is the target variable with Labour – 1057 & Conservative – 460 counts. The data is dominatingly ‘Labour’. This shall be considered in train_test_split. The binary values considered for encoding ‘vote’ are ‘Labour’ is 1 and ‘Conservative’ is 0.

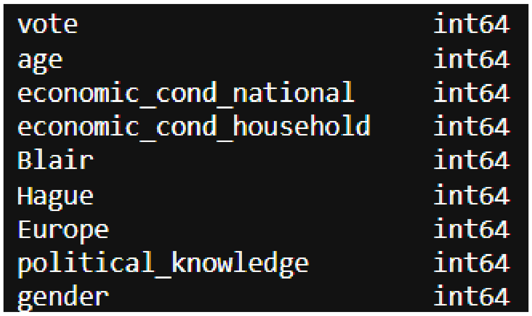
After encoding, the sample dataset looks like below.



	vote	age	economic_cond_national	economic_cond_household	Blair	Hague	Europe	political_knowledge	gender
0	1	43	3	3	4	1	2	2	1
1	1	36	4	4	4	4	5	2	0
2	1	35	4	4	5	2	3	2	0
3	1	24	4	2	2	1	4	0	1
4	1	41	2	2	1	1	6	2	0

Fig1.2.2.1 Election Data after encoding

The dataset types are converted to numeric as all columns are numeric. Below figure shows the datatypes of the feature after the encoding:



```
vote                       int64
age                        int64
economic_cond_national     int64
economic_cond_household    int64
Blair                      int64
Hague                      int64
Europe                     int64
political_knowledge        int64
gender                     int64
```

Fig1.2.2.2 Election Datatypes after converting datatype

1.2.3 Outliers

The boxplot is plotted to visualize the outliers in all numerical features.

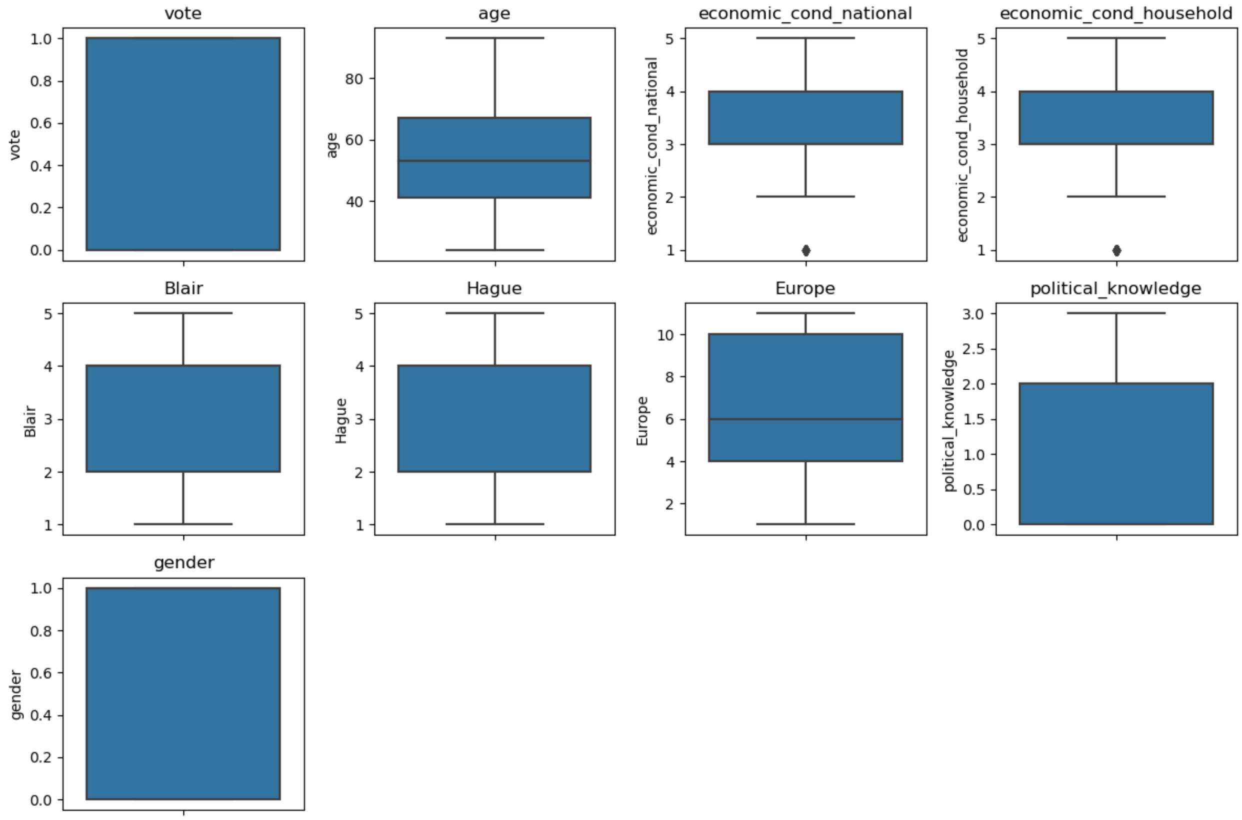


Fig1.2.3.1 Election Data Boxplot showing outliers

There are outliers in ‘economic_cond_household’ & ‘economic_cond_national’ as shown in above plot.

The black dots in the boxplots show that there are outliers in these columns. Except 'institutions' remaining features have outliers. Majority of the features are uniform with very slight skewness in the data. This can be seen in the boxplots as slightly larger tails with small magnitude (Y value)indicating slight skew. ‘age’ has right skew. ‘Europe’ has left skew. Skewness is discussed in pairplot & same observations can be inferred here.

Treat outliers:

All the outliers are treated by adjusting them to the lower and upper bound values calculated by the IQR value.

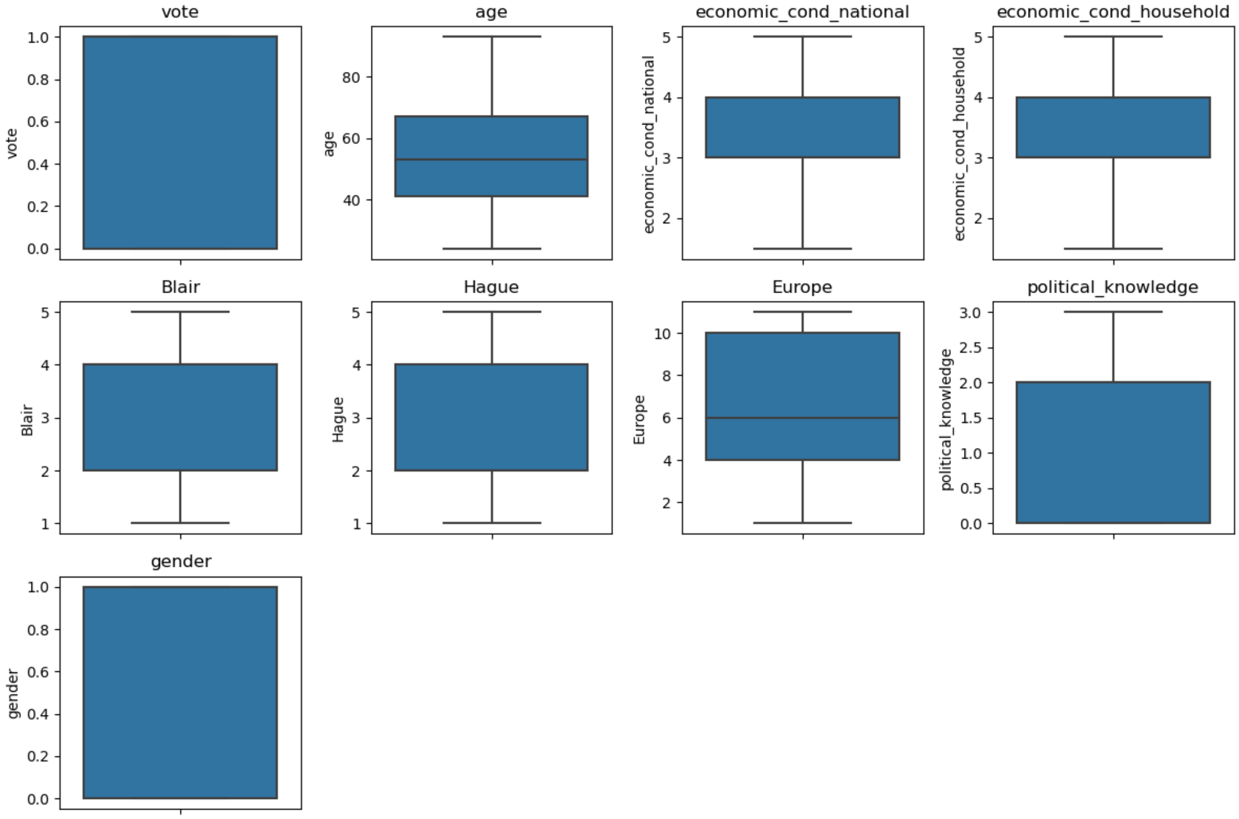
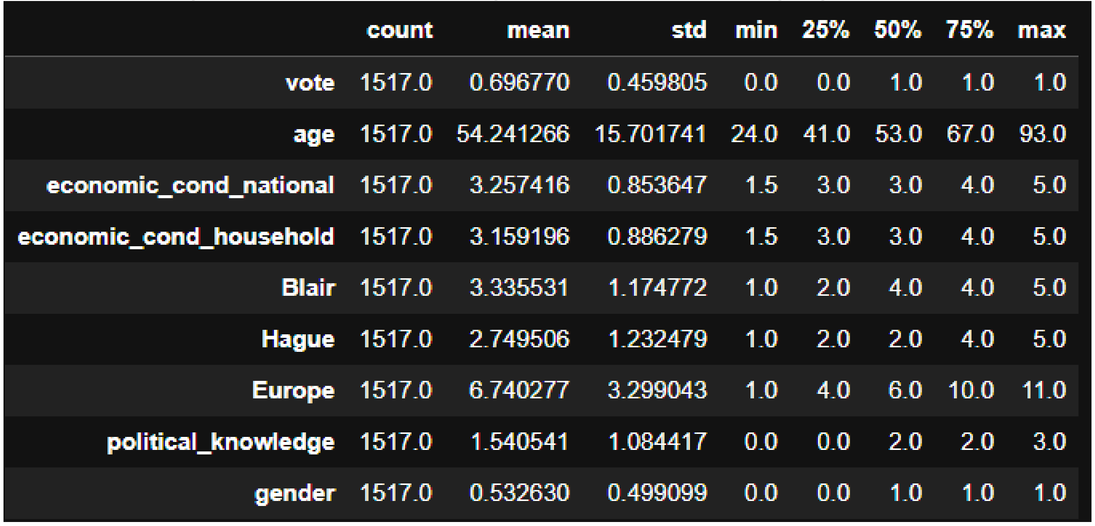


Fig1.2.3.2 Election Data Boxplot after treating outliers

The above boxplot is after treating the outliers showing no outliers. The data now does not have outliers and the data seems distributed with a slight negative skew in data for these two features.

1.2.4 Scaling

The below snap shows the data description after the data is prepared.



	count	mean	std	min	25%	50%	75%	max
vote	1517.0	0.696770	0.459805	0.0	0.0	1.0	1.0	1.0
age	1517.0	54.241266	15.701741	24.0	41.0	53.0	67.0	93.0
economic_cond_national	1517.0	3.257416	0.853647	1.5	3.0	3.0	4.0	5.0
economic_cond_household	1517.0	3.159196	0.886279	1.5	3.0	3.0	4.0	5.0
Blair	1517.0	3.335531	1.174772	1.0	2.0	4.0	4.0	5.0
Hague	1517.0	2.749506	1.232479	1.0	2.0	2.0	4.0	5.0
Europe	1517.0	6.740277	3.299043	1.0	4.0	6.0	10.0	11.0
political_knowledge	1517.0	1.540541	1.084417	0.0	0.0	2.0	2.0	3.0
gender	1517.0	0.532630	0.499099	0.0	0.0	1.0	1.0	1.0

Fig1.2.4.1 Election Data after scaling

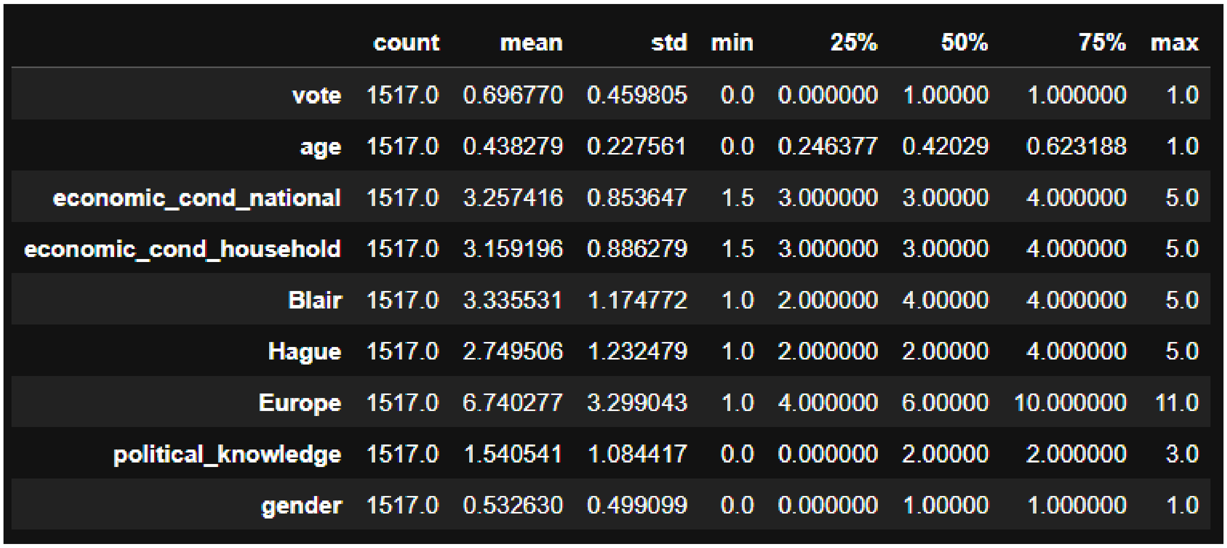
Scaling - Magnitude Differences:

Generally decision trees are not sensitive to scales as splitting process is based on comparing feature values against thresholds, rather than the actual magnitude of the value. Scaling can be case to case. Scaling can help to ensure that all features contribute equally to the model.

In our case, from above data description we can see that the variable 'age' feature has a relatively wide range with a standard deviation of approximately 15.7. Scaling can be beneficial in this case. ‘age’ feature contains continuous values & the values are higher in magnitude compared to other features which are categorical ratings.

Hence MinMax scaling is performed on ‘age’, while preserving all other features as is.

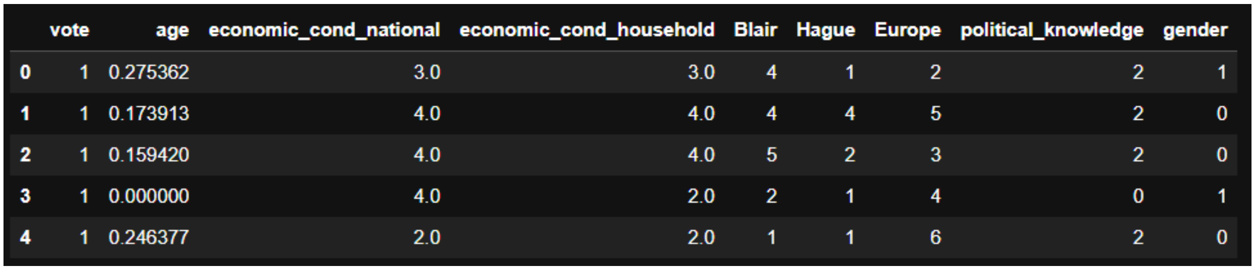
The data description after all data is prepared is now shown below and it can be seen that the scales are very similar for all features.



	count	mean	std	min	25%	50%	75%	max
vote	1517.0	0.696770	0.459805	0.0	0.000000	1.00000	1.000000	1.0
age	1517.0	0.438279	0.227561	0.0	0.246377	0.42029	0.623188	1.0
economic_cond_national	1517.0	3.257416	0.853647	1.5	3.000000	3.00000	4.000000	5.0
economic_cond_household	1517.0	3.159196	0.886279	1.5	3.000000	3.00000	4.000000	5.0
Blair	1517.0	3.335531	1.174772	1.0	2.000000	4.00000	4.000000	5.0
Hague	1517.0	2.749506	1.232479	1.0	2.000000	2.00000	4.000000	5.0
Europe	1517.0	6.740277	3.299043	1.0	4.000000	6.00000	10.000000	11.0
political_knowledge	1517.0	1.540541	1.084417	0.0	0.000000	2.00000	2.000000	3.0
gender	1517.0	0.532630	0.499099	0.0	0.000000	1.00000	1.000000	1.0

Fig1.2.4.2 Election Data description after scaling

Below figure shows the dataset sample after scaling.



	vote	age	economic_cond_national	economic_cond_household	Blair	Hague	Europe	political_knowledge	gender
0	1	0.275362	3.0	3.0	4	1	2	2	1
1	1	0.173913	4.0	4.0	4	4	5	2	0
2	1	0.159420	4.0	4.0	5	2	3	2	0
3	1	0.000000	4.0	2.0	2	1	4	0	1
4	1	0.246377	2.0	2.0	1	1	6	2	0

Fig1.2.4.3 Election Data description after scaling

1.2.5 Co-relation

The heatmap showing co-relation between all features is plotted.

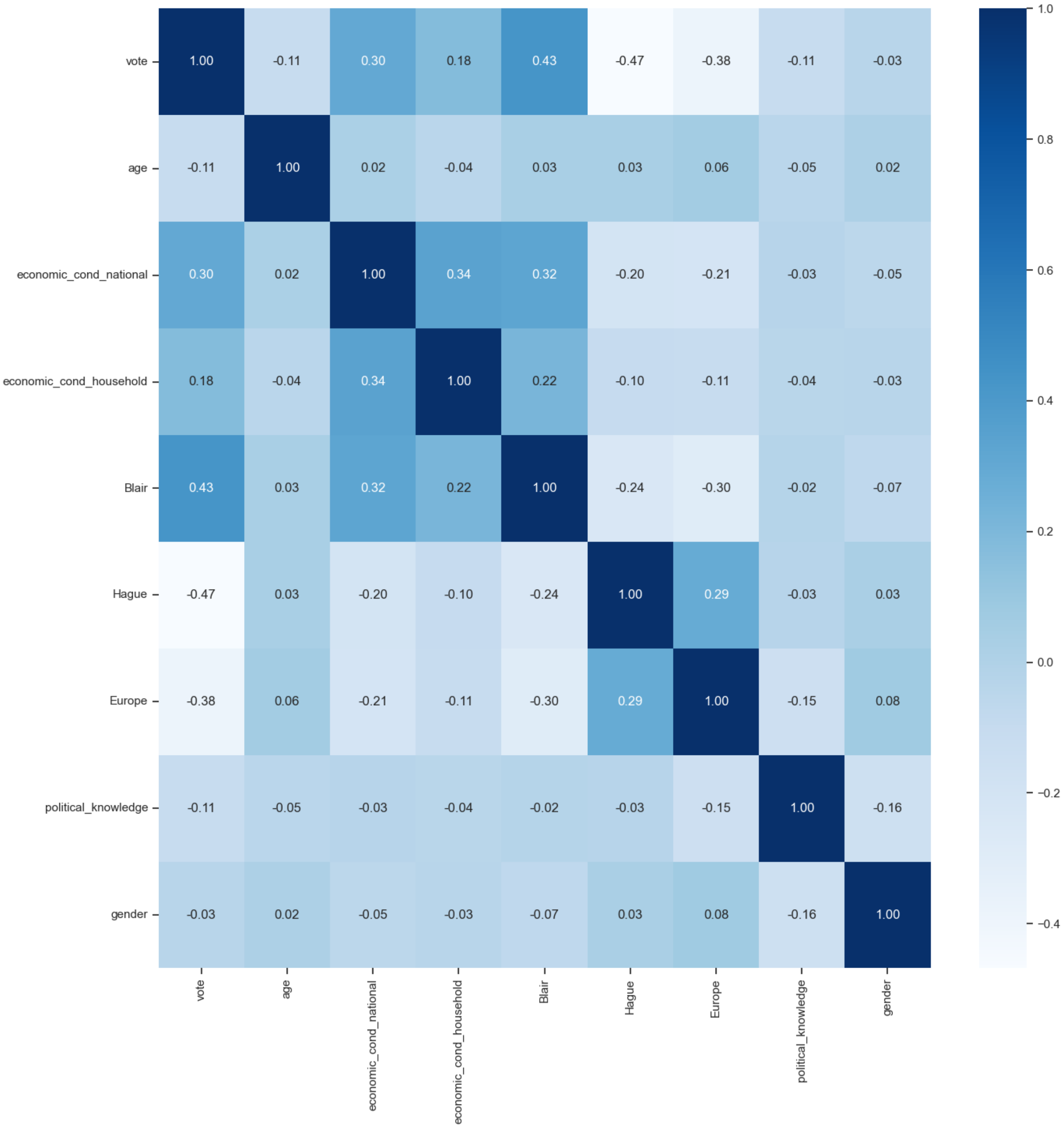


Fig1.2.5.1 Heatmap – of independent variables - after treating outliers

Based on the above correlation matrix plot:

Co-relation between independent variables is discussed in Bivariate analysis using heatmap.

Now, based on the correlation matrix with 'vote' feature, below are comments about individual variables on 'vote':

1. Co-relation of 'vote' can be seen with Blair, age, economic_cond_national, economic_cond_household.
2. Vote (target variable) has moderate positive correlation with Blair (0.43), suggesting as Blair assessment value increases, the likelihood of voting for the Labour party also increases. This can be understood as Blair is leader of Labour party.
1. Similarly, Hague (Assessment of the Conservative leader) has a moderate negative correlation with Vote (-0.47), suggesting that a higher assessment of the Conservative leader is associated with a lower likelihood of voting for the Labour party & higher .
2. Economic condition variables (economic_cond_national and economic_cond_household) show moderate and slightly positive correlations with Vote respectively, indicating that as the perception of economic conditions improves, the likelihood of voting for either party increases.
3. Older individuals may be slightly less likely to vote for either party.There is high co-relation between the independent variables. This needs to be addressed. It's important to note that correlation alone does not imply causation. Further analysis and domain knowledge are needed to determine the underlying relationships and causality between variables.

! These insights provide a glimpse into the relationships between different variables in the dataset. However, it's important to note that correlation does not imply causation, and further analysis and domain knowledge are necessary to draw meaningful conclusions.

1.3 TRAIN_TEST_SPLIT

Encode the data (having string values) for Modelling. Is Scaling necessary here or not?(2 pts), Data Split: Split the data into train and test (70:30) (2 pts). The learner is expected to check and comment about the difference in scale of different features on the bases of appropriate measure for example std dev, variance, etc. Should justify whether there is a necessity for scaling. Object data should be converted into categorical/numerical data to fit in the models. (pd.categorical().codes(), pd.get_dummies(drop_first=True)) Data split, ratio defined for the split, train-test split should be discussed.

Answer:

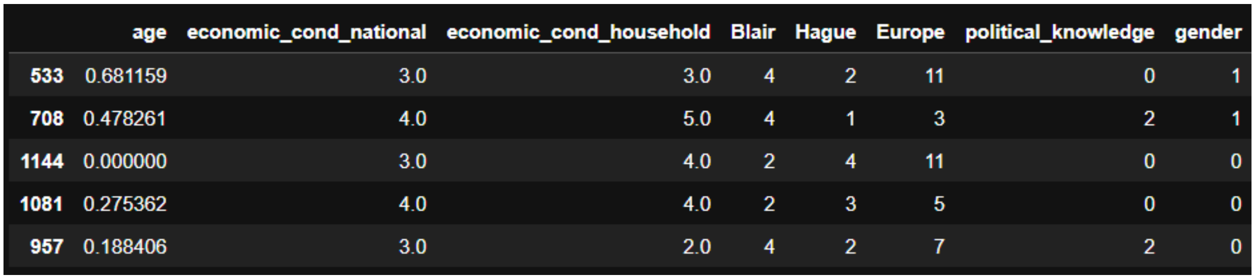
Encoding and Scaling have been discussed in above section.

1.3.1 Split Data

The ‘vote feature is the target variable, y. Rest of the data is independent variables or X. Using sklearn train_test_split, the dataset is split.

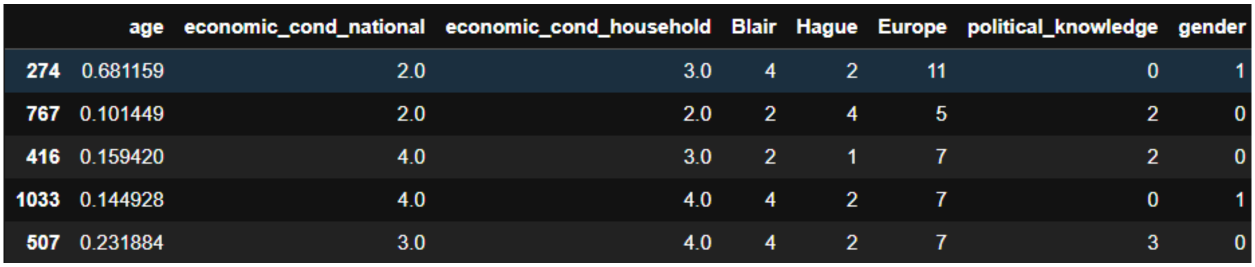
1. Standard 70 30 split is performed for training and testing data. This split percentage has ample 70% training data and 30 % testing data to check the model's performance on unseen data.
2. Random_state 1 is used to replicate the results.
3. stratify parameter is used on ‘vote’ variable as ‘vote’ feature has imbalance in terms of observations counts. Stratify will ensure a balance in the data and will not cause biasness while Training or Testing the mode.

The training and test dataset sample is shown below:



	age	economic_cond_national	economic_cond_household	Blair	Hague	Europe	political_knowledge	gender
533	0.681159	3.0	3.0	4	2	11	0	1
708	0.478261	4.0	5.0	4	1	3	2	1
1144	0.000000	3.0	4.0	2	4	11	0	0
1081	0.275362	4.0	4.0	2	3	5	0	0
957	0.188406	3.0	2.0	4	2	7	2	0

Fig1.3.1.1 Election Data- Train data sample



	age	economic_cond_national	economic_cond_household	Blair	Hague	Europe	political_knowledge	gender
274	0.681159	2.0	3.0	4	2	11	0	1
767	0.101449	2.0	2.0	2	4	5	2	0
416	0.159420	4.0	3.0	2	1	7	2	0
1033	0.144928	4.0	4.0	4	2	7	0	1
507	0.231884	3.0	4.0	4	2	7	3	0

Fig1.3.1.2 Election Data- Test data sample

Note that the dataset has some degree of correlation & multi collinearity as shown in heatmaps in above section.

1.4 LOGISTIC REGRESSION & LDA

Apply Logistic Regression and LDA (Linear Discriminant Analysis) (2 pts). Interpret the inferences of both model s (2 pts). Successful implementation of each model. Logical reason should be shared if any custom changes are made to the parameters while building the model. Calculate Train and Test Accuracies for each model. Comment on the validness of models (over fitting or under fitting)

Answer:

Tuning hyperparameters is crucial for optimizing model performance and finding the right balance between underfitting and overfitting.

1.4.1 Hyper-parameters

Logistic Regression and LDA (Linear Discriminant Analysis) Parameters:

1. The max_iter parameter sets an upper limit on the number of iterations allowed, ensuring that the algorithm doesn't run indefinitely. In first model, max_iter for Logistic Model is set as 1000.
2. The LDA n_components hyperparameter determines the number of components or dimensions to keep in the transformed space. In first model, n_components is set to 1- since we have 2 classes. LDA will project the data onto a one-dimensional subspace. The resulting transformed feature space will have one dimension.

1.4.2 Model & Performance

The Logistic Regression model & LDA model performance metrics are shown below:

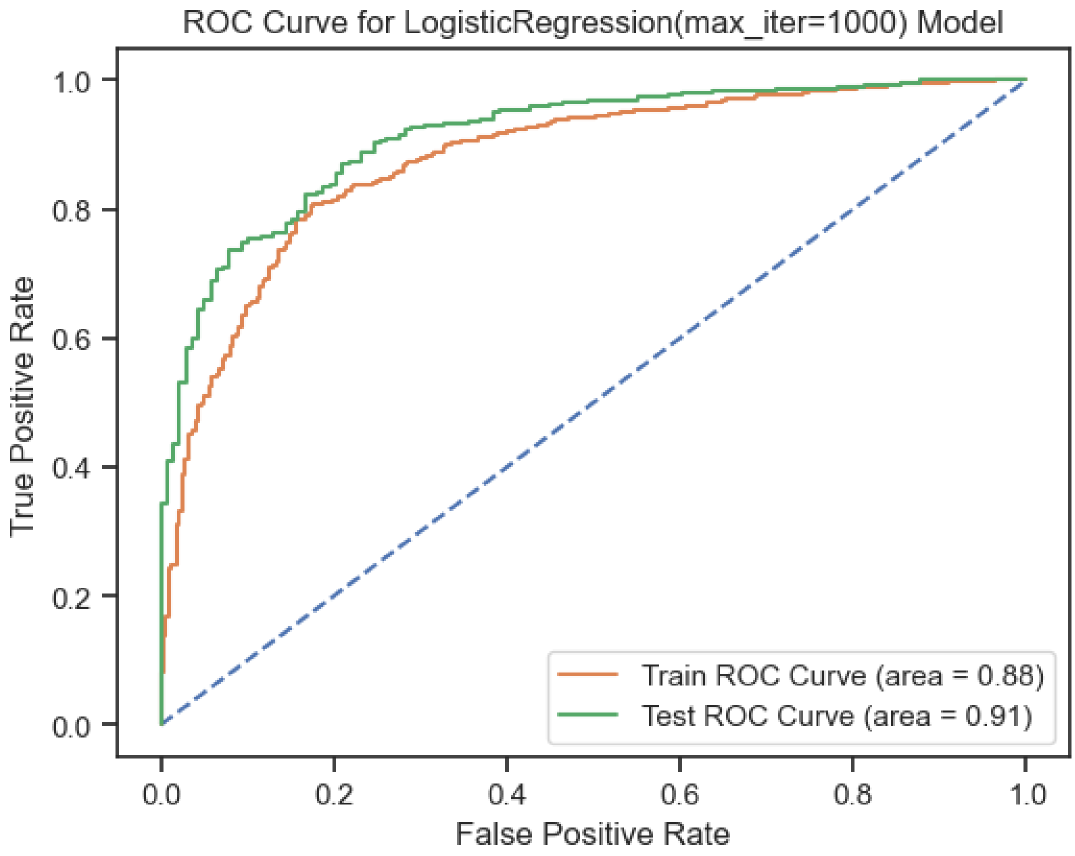


Fig1.4.2.1 Model 1- Linear Regression ROC Curve

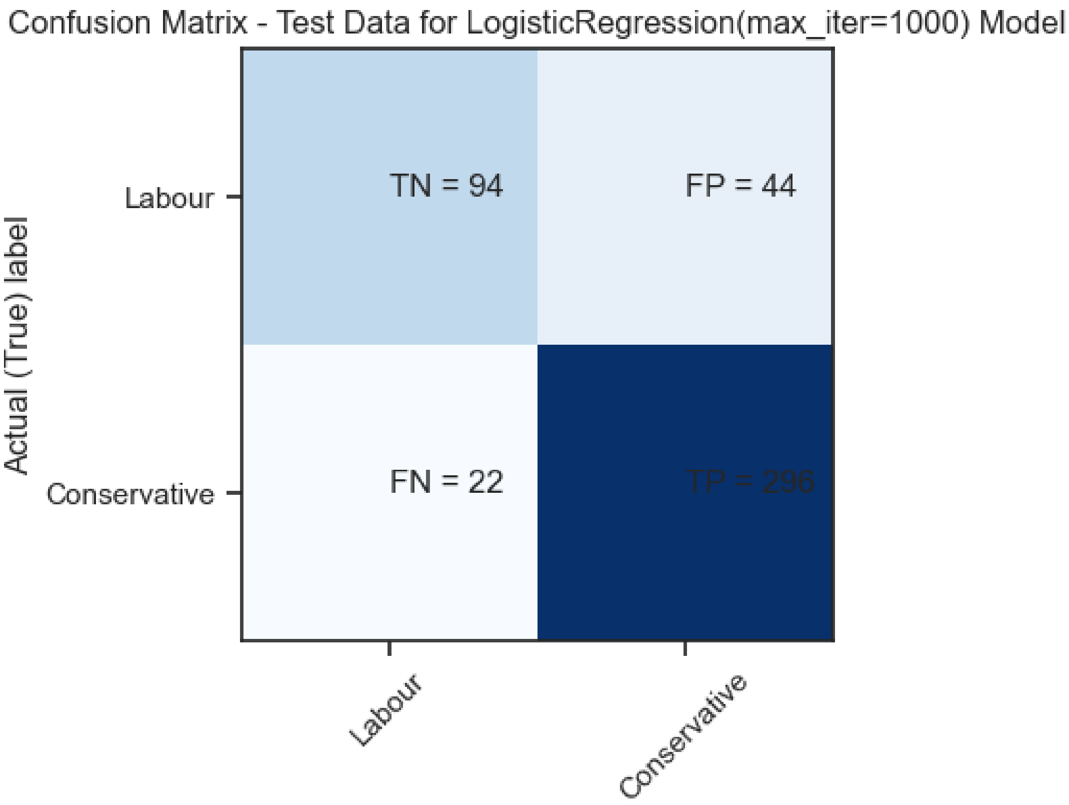
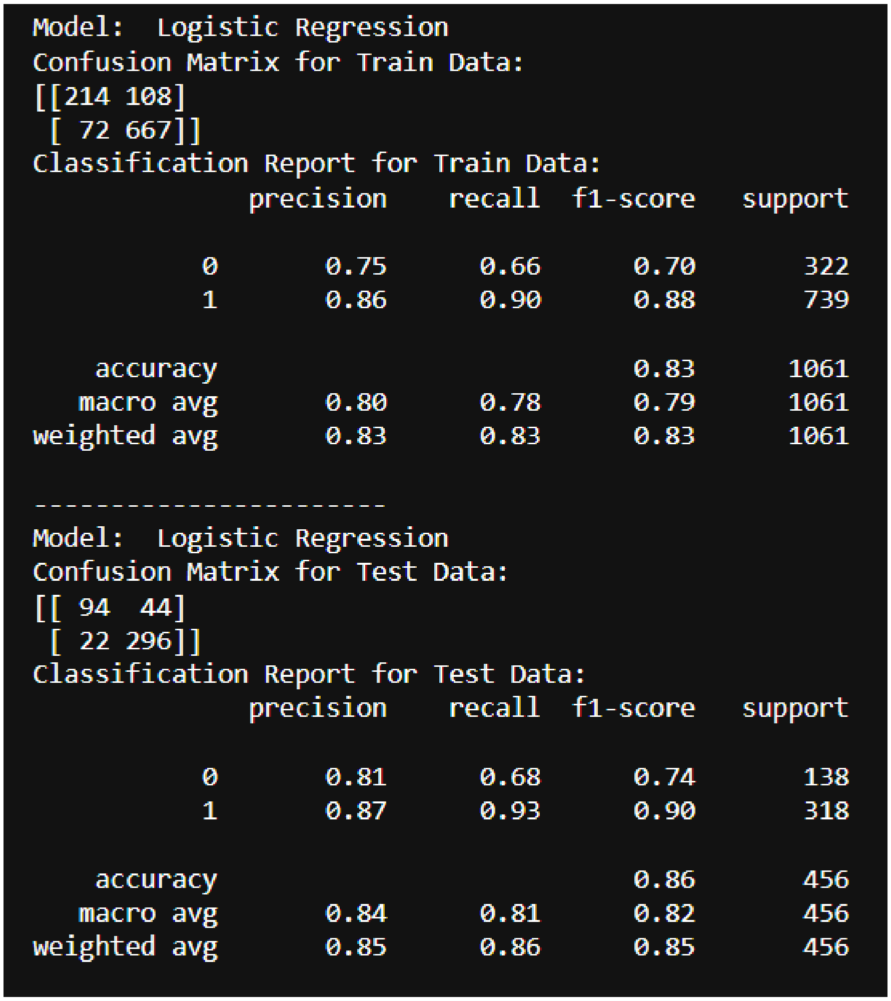


Fig1.4.2.2 Model 1- Linear Regression Confusion Matrix Plot



```
Model:  Logistic Regression
Confusion Matrix for Train Data:
[[214 108]
 [ 72 667]]
Classification Report for Train Data:
              precision    recall  f1-score   support

           0       0.75      0.66      0.70       322
           1       0.86      0.90      0.88       739

    accuracy                           0.83      1061
   macro avg       0.80      0.78      0.79      1061
weighted avg       0.83      0.83      0.83      1061

-----------------------
Model:  Logistic Regression
Confusion Matrix for Test Data:
[[ 94  44]
 [ 22 296]]
Classification Report for Test Data:
              precision    recall  f1-score   support

           0       0.81      0.68      0.74       138
           1       0.87      0.93      0.90       318

    accuracy                           0.86       456
   macro avg       0.84      0.81      0.82       456
weighted avg       0.85      0.86      0.85       456
```

Fig1.4.2.3 Model 1- Linear Regression Model Summary

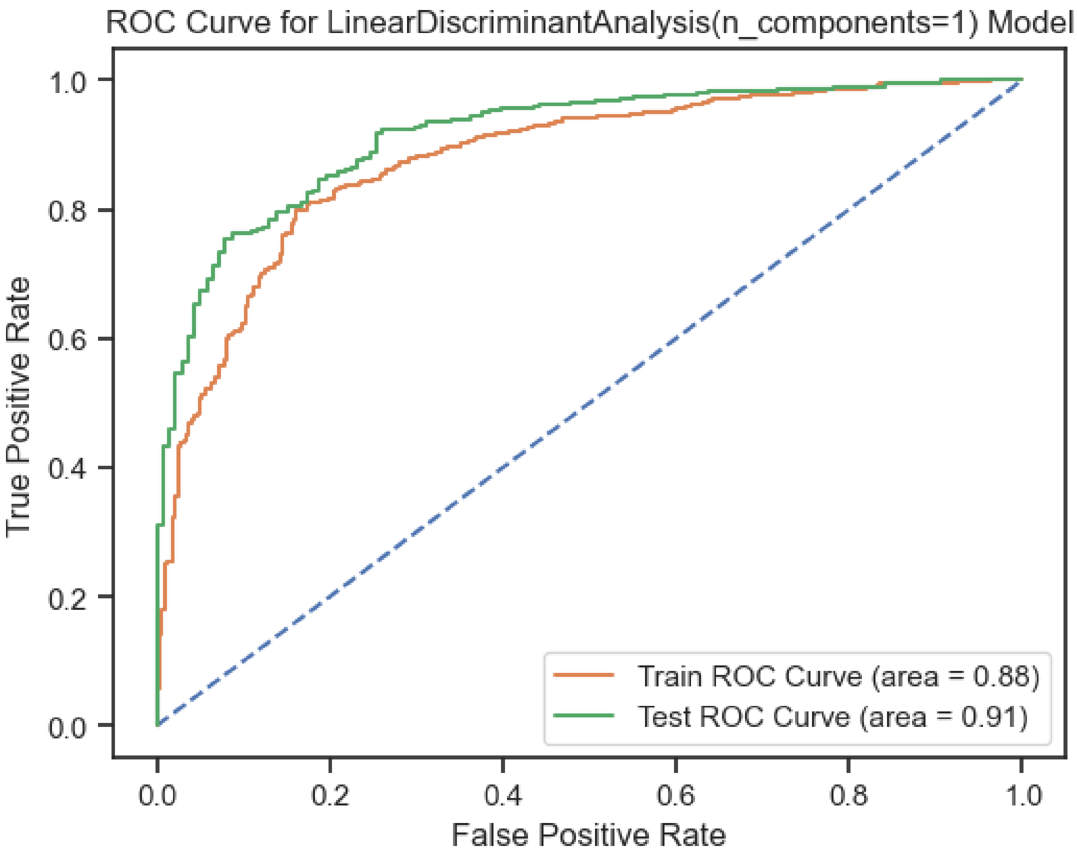


Fig1.4.2.4 Model 1- LDA ROC Curve

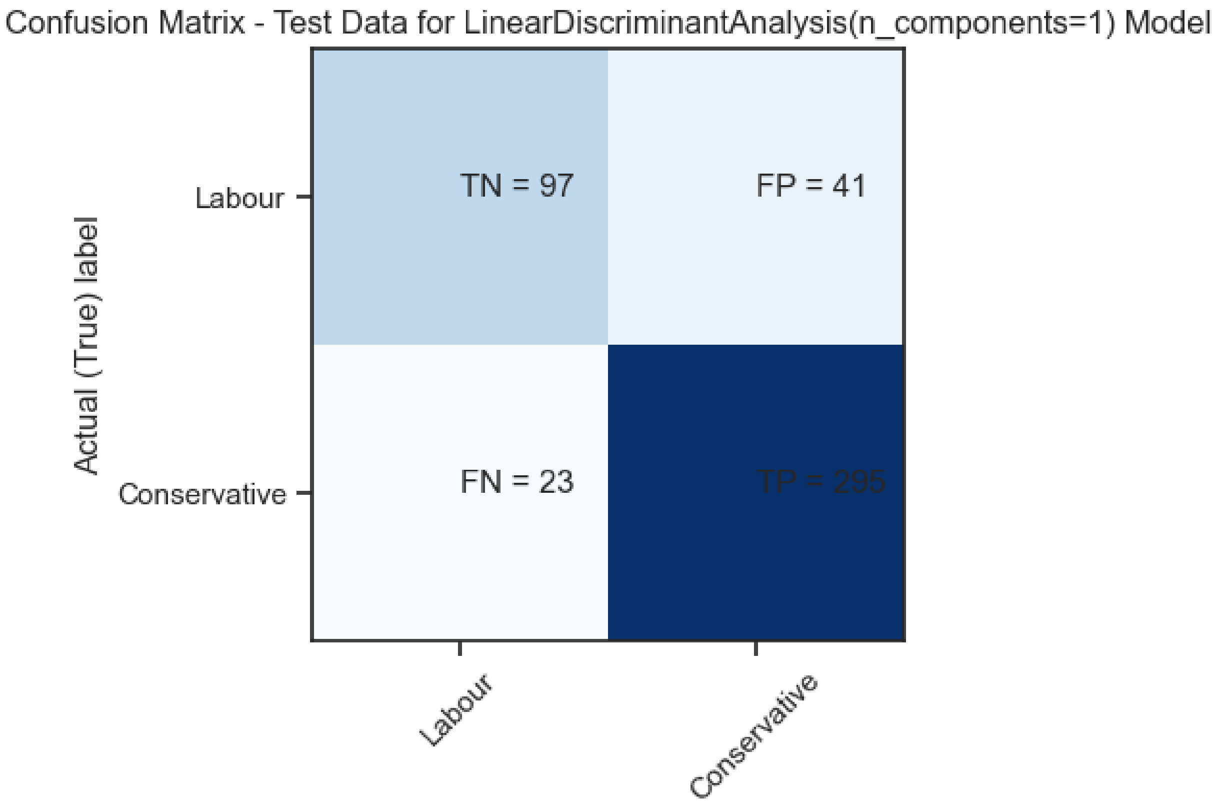
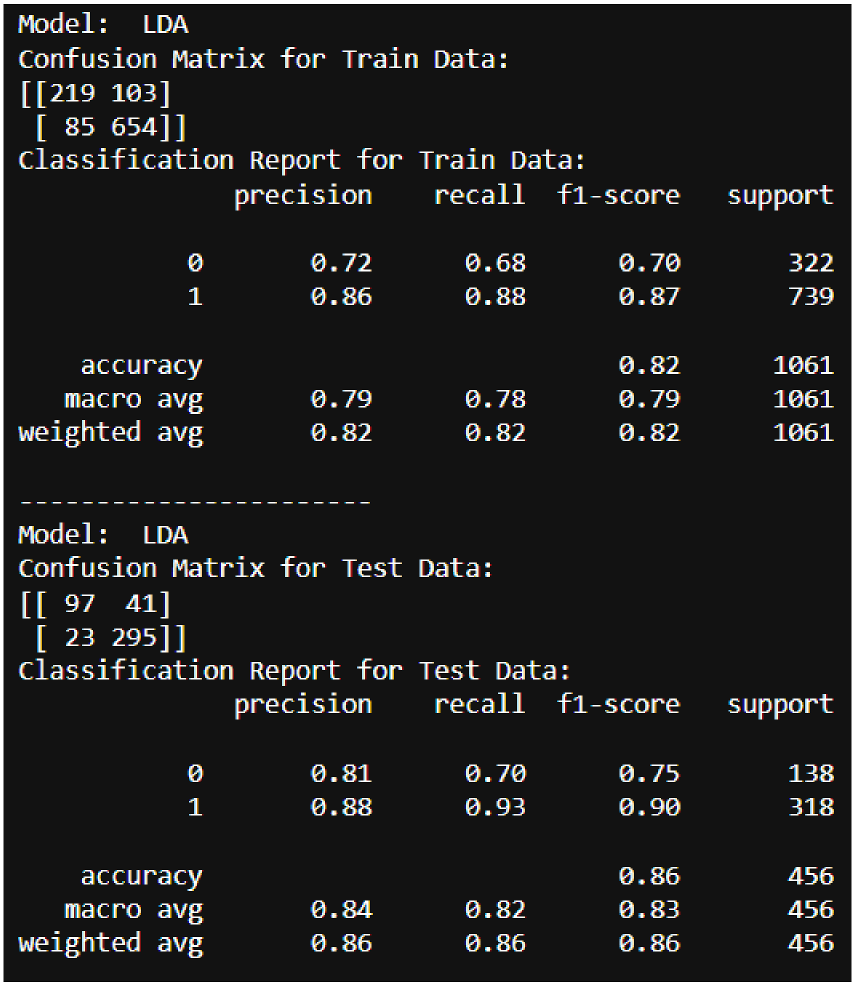


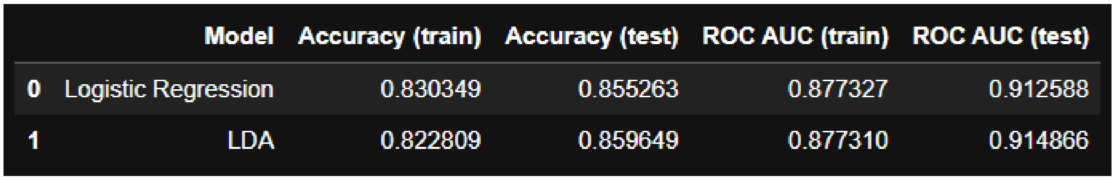
Fig1.4.2.5 Model 1- LDA Confusion Matrix Plot



```
Model:  LDA
Confusion Matrix for Train Data:
[[219 103]
 [ 85 654]]
Classification Report for Train Data:
              precision    recall  f1-score   support

           0       0.72      0.68      0.70       322
           1       0.86      0.88      0.87       739

    accuracy                           0.82      1061
   macro avg       0.79      0.78      0.79      1061
weighted avg       0.82      0.82      0.82      1061

-----------------------
Model:  LDA
Confusion Matrix for Test Data:
[[ 97  41]
 [ 23 295]]
Classification Report for Test Data:
              precision    recall  f1-score   support

           0       0.81      0.70      0.75       138
           1       0.88      0.93      0.90       318

    accuracy                           0.86       456
   macro avg       0.84      0.82      0.83       456
weighted avg       0.86      0.86      0.86       456
```

Fig1.4.2.6 Model 1- LDA Model Summary

1.4.3 Inference



	Model	Accuracy (train)	Accuracy (test)	ROC AUC (train)	ROC AUC (test)
0	Logistic Regression	0.830349	0.855263	0.877327	0.912588
1	LDA	0.822809	0.859649	0.877310	0.914866

Fig1.4.3.1 Model 1- Logistic & LDA quick performance summary

Inference: Based on these two model results & classification reports, both models have similar performance in terms of accuracy, precision, and F1-score. However, the Logistic Regression model achieves slightly higher accuracy and F1-scores on both train and test data compared to LDA.

Therefore, based on the provided information, the Logistic Regression model may be considered a slightly better model for this classification task.

1.5 KNN & NAÏVE-BAYES

Apply KNN Model and Naïve Bayes Model (2pts). Interpret the inferences of each model (2 pts). Successful implementation of each model. Logical reason should be shared if any custom changes are made to the parameters while building the model. Calculate Train and Test Accuracies for each model. Comment on the validness of models (over fitting or under fitting).

Answer:

1.5.1 Hyperparameters

1. KNN-'n_neighbors' specifies number of nearest neighbors to consider when making predictions for a new data point. The n_neighbors can also be set with domain understanding. Example- if the patterns in the data are complex, use a lower value We are not setting any hyper-parameters now in this section.
2. We can play with different models of 'n_ neighbors' values as 3, 5, 7, or 10. Then compare performance.
3. Alternately, the cross-validation technique also determines optimal KNN n_neighbors. This approach shall be considered in the following section
4. For Naïve Bayes as well, we are not setting hyper parameters in this model.

This approach is to see how the model performs and tune it accordingly in following sections.

1.5.2 Model & Performance

Below are the performance metrics from the KNN model & Naïve Bayess

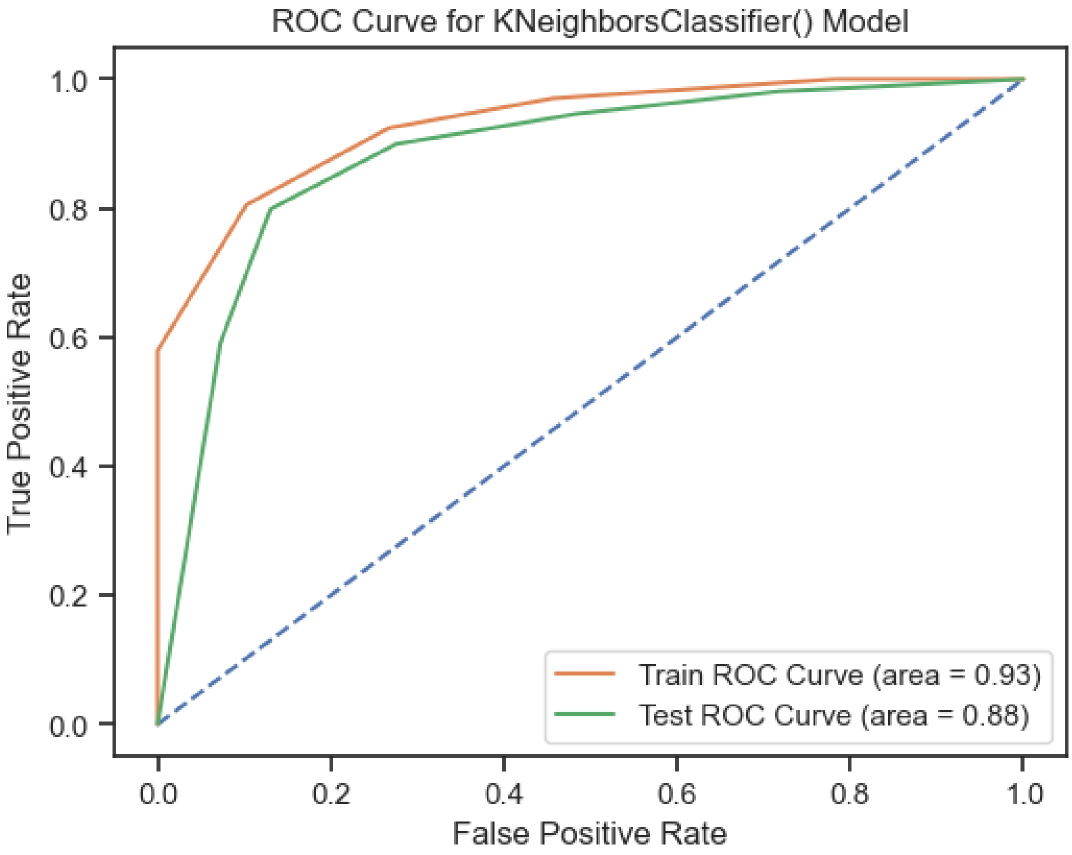


Fig1.5.2.1 KNN Model 1- ROC Curve

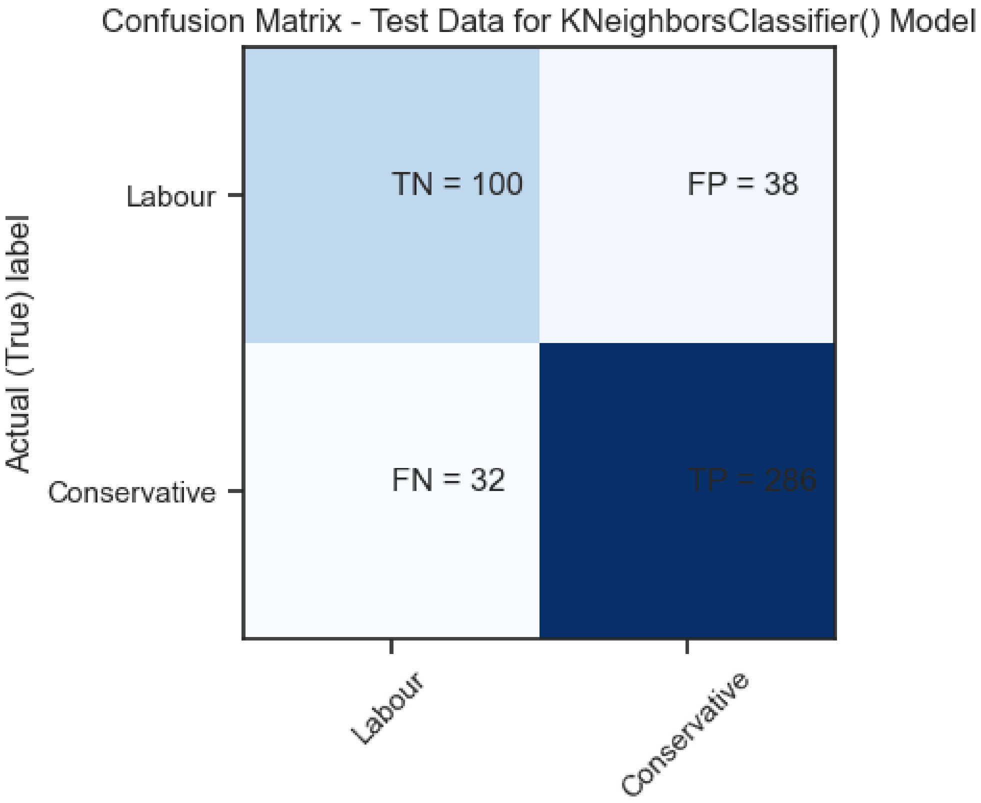
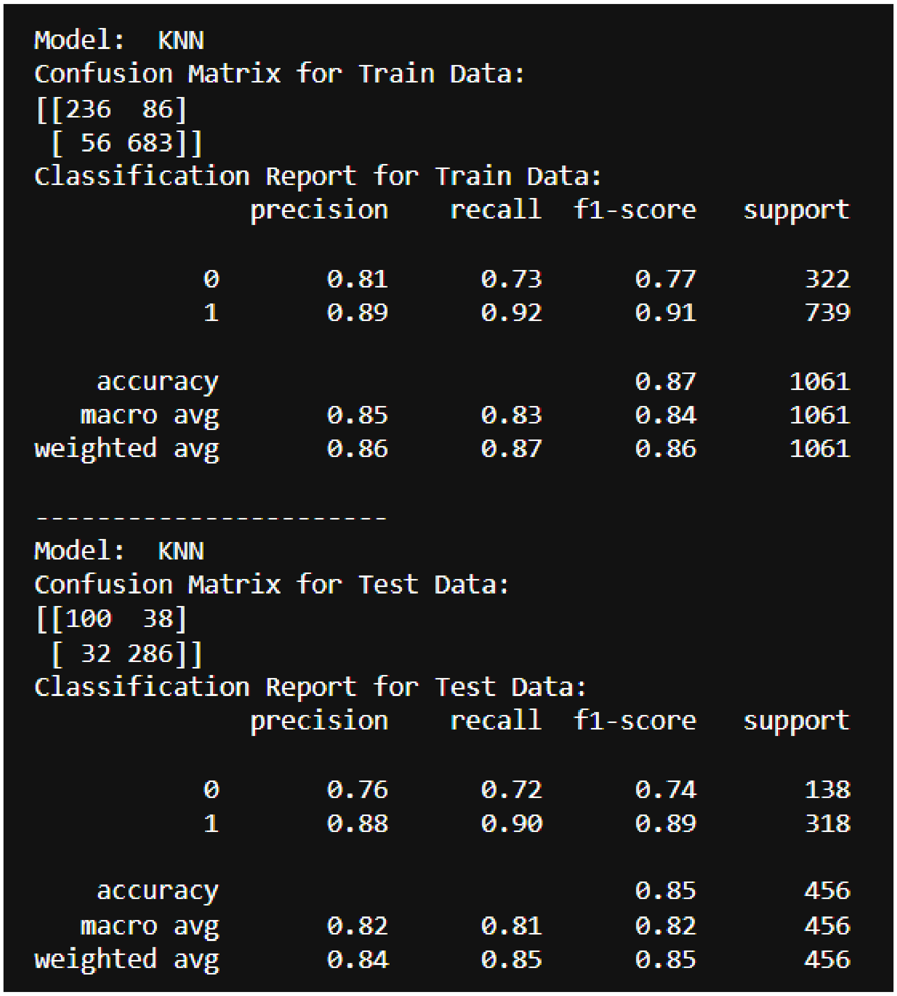


Fig1.5.2.2 KNN Model 1- Confusion Matrix



```
Model:  KNN
Confusion Matrix for Train Data:
[[236  86]
 [ 56 683]]
Classification Report for Train Data:
              precision    recall  f1-score   support

           0       0.81      0.73      0.77       322
           1       0.89      0.92      0.91       739

    accuracy                           0.87      1061
   macro avg       0.85      0.83      0.84      1061
weighted avg       0.86      0.87      0.86      1061

-----------------------
Model:  KNN
Confusion Matrix for Test Data:
[[100  38]
 [ 32 286]]
Classification Report for Test Data:
              precision    recall  f1-score   support

           0       0.76      0.72      0.74       138
           1       0.88      0.90      0.89       318

    accuracy                           0.85       456
   macro avg       0.82      0.81      0.82       456
weighted avg       0.84      0.85      0.85       456
```

Fig1.5.2.3 KNN Model 1- Summary

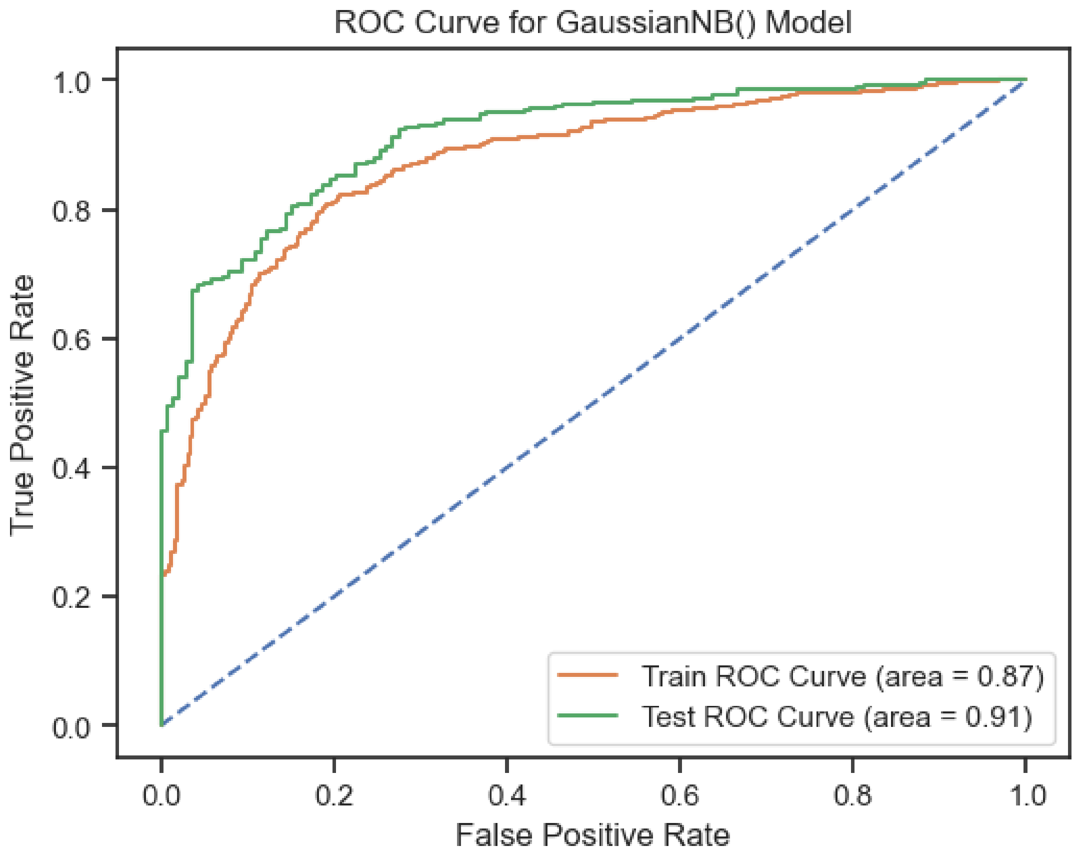


Fig1.5.2.4 Naïve Bayes Model 1- ROC Curve

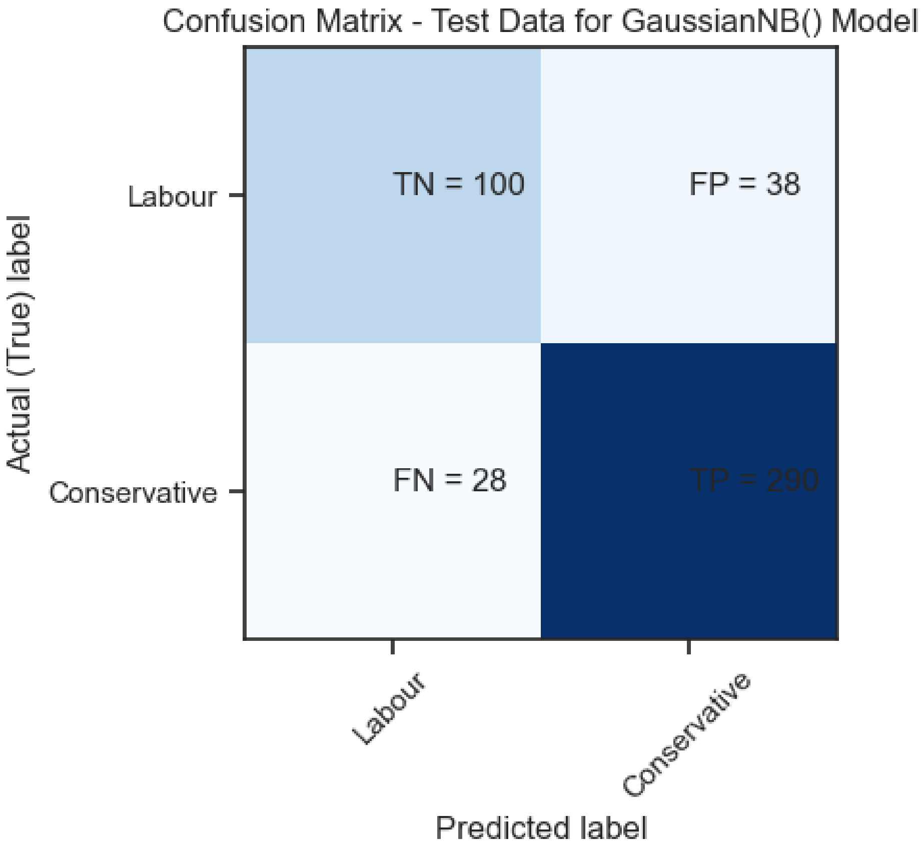
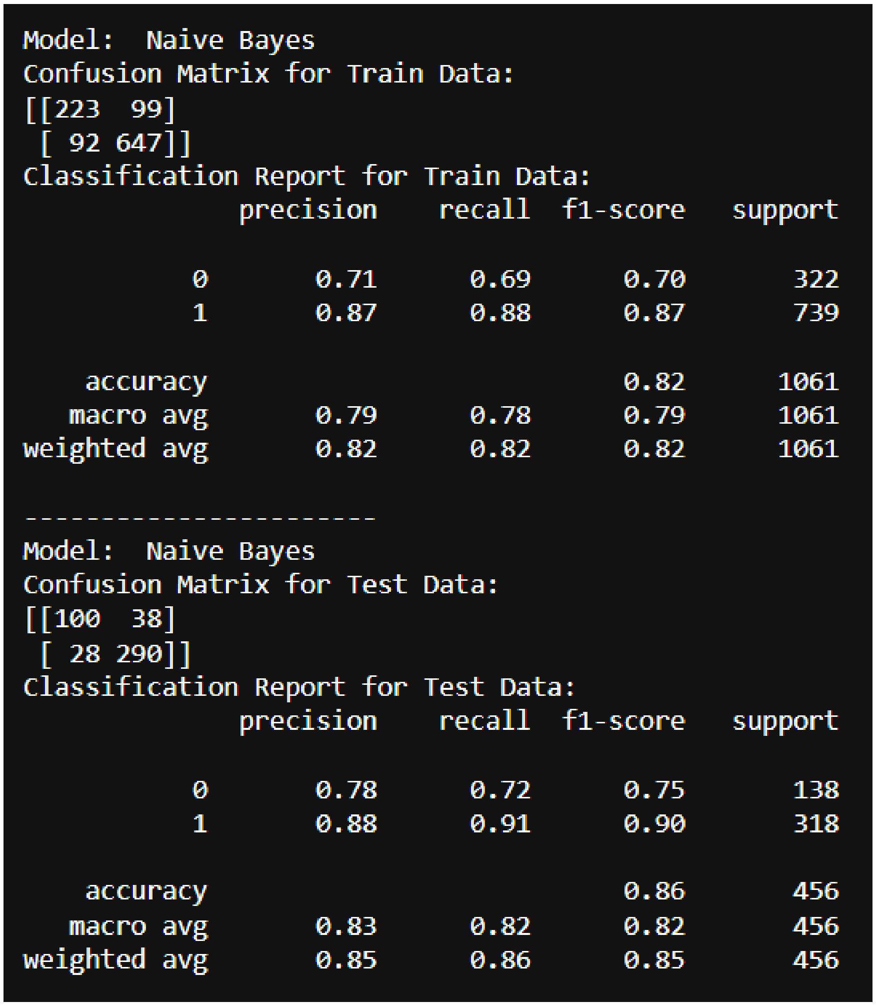


Fig1.5.2.5 Naïve Bayes Model 1- Confusion Matrix

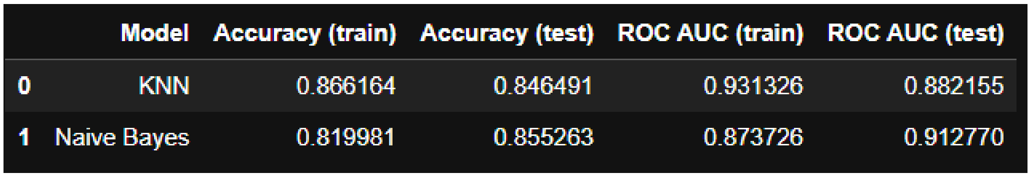


```
Model:  Naive Bayes
Confusion Matrix for Train Data:
[[223  99]
 [ 92 647]]
Classification Report for Train Data:
              precision    recall  f1-score   support

           0       0.71      0.69      0.70       322
           1       0.87      0.88      0.87       739

    accuracy                           0.82      1061
   macro avg       0.79      0.78      0.79      1061
weighted avg       0.82      0.82      0.82      1061

-----------------------
Model:  Naive Bayes
Confusion Matrix for Test Data:
[[100  38]
 [ 28 290]]
Classification Report for Test Data:
              precision    recall  f1-score   support

           0       0.78      0.72      0.75       138
           1       0.88      0.91      0.90       318

    accuracy                           0.86       456
   macro avg       0.83      0.82      0.82       456
weighted avg       0.85      0.86      0.85       456
```

Fig1.5.2.6 Naïve Bayes Model 1- Summary

1.5.3 Inference

A quick summary is shown below:



	Model	Accuracy (train)	Accuracy (test)	ROC AUC (train)	ROC AUC (test)
0	KNN	0.866164	0.846491	0.931326	0.882155
1	Naive Bayes	0.819981	0.855263	0.873726	0.912770

Fig1.5.3.1 KNN & Naïve Bayes Model Quick Summary

Inferences:

1. Both models show good 86.6% accuracy on training dataset.
2. There is potential overfitting with KNN as the test accuracy decreased to 84.6%.
3. KNN model performed better than Naïve Bayes in terms of accuracy, precision, recall, an d F1-score on both the training and test data. However, the difference in performance is not very high.

4. Comparing all the summary of the two models i.e. KNN & Naïve Bayes, it can said that it will be beneficial to consider other models or fine tuning these models. Example- Model t uning will be used to find the best KNN neighbors.

1.6 MODEL TUNING

Model Tuning (4 pts) , Bagging (1.5 pts) and Boosting (1.5 pts). Apply grid search on each model (include all models) and make models on best_params. Compare and comment on performances o f all. Comment on feature importance if applicable. Successful implementation of both algorithms along with inferences and comments on the model performances

Answer:

1.6.1 Logistic Model Tuned

The ROC Curve for the best logistic model is shown below.

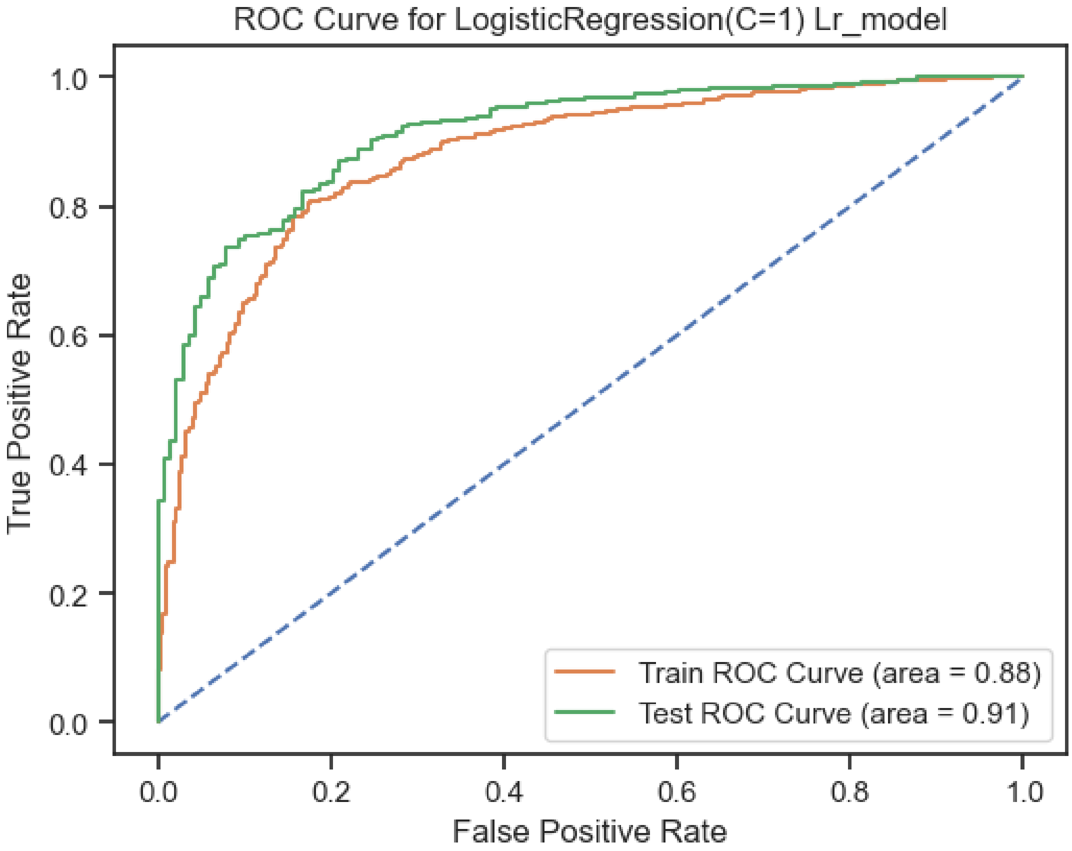
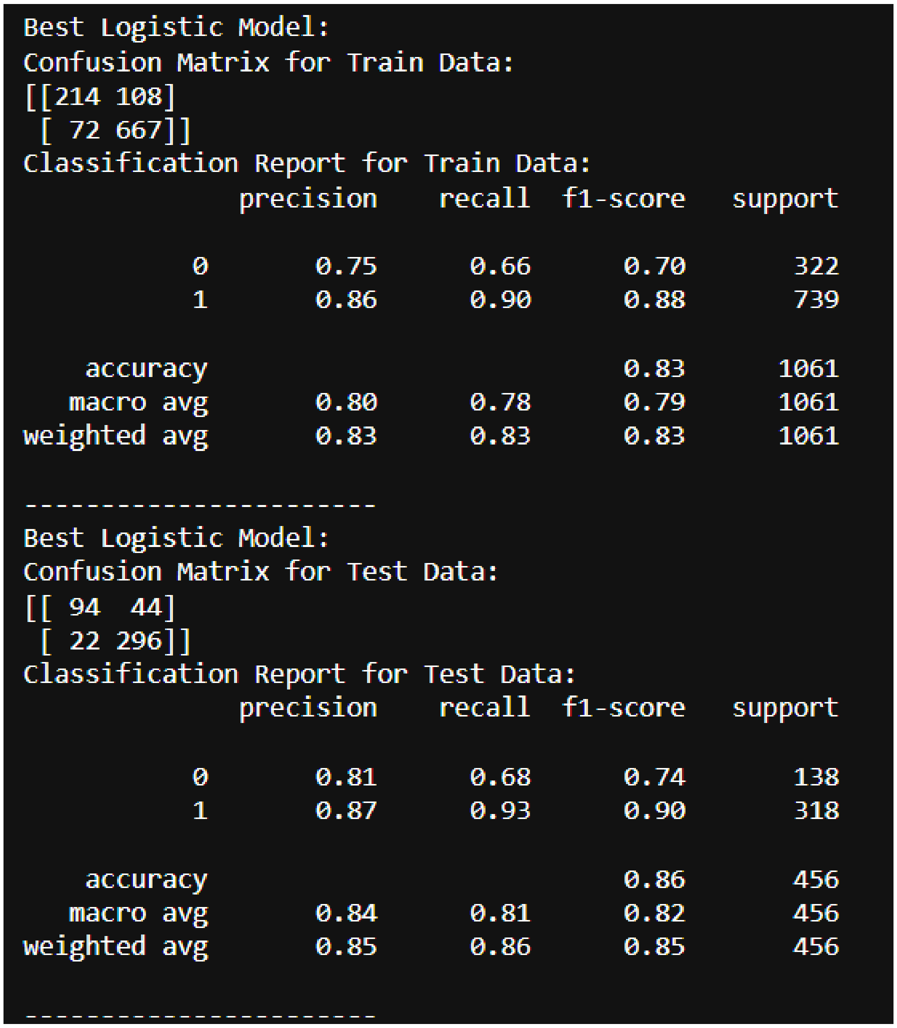


Fig 1.6.1.1 Logistic Model 2 – ROC Curve



```
Best Logistic Model:
Confusion Matrix for Train Data:
[[214 108]
 [ 72 667]]
Classification Report for Train Data:
              precision    recall  f1-score   support

           0       0.75      0.66      0.70       322
           1       0.86      0.90      0.88       739

    accuracy                           0.83      1061
   macro avg       0.80      0.78      0.79      1061
weighted avg       0.83      0.83      0.83      1061

-----------------------
Best Logistic Model:
Confusion Matrix for Test Data:
[[ 94  44]
 [ 22 296]]
Classification Report for Test Data:
              precision    recall  f1-score   support

           0       0.81      0.68      0.74       138
           1       0.87      0.93      0.90       318

    accuracy                           0.86       456
   macro avg       0.84      0.81      0.82       456
weighted avg       0.85      0.86      0.85       456

-----------------------
```

Fig 1.6.1.2 Logistic Model 2 - Summary

1.6.2 KNN Tuned

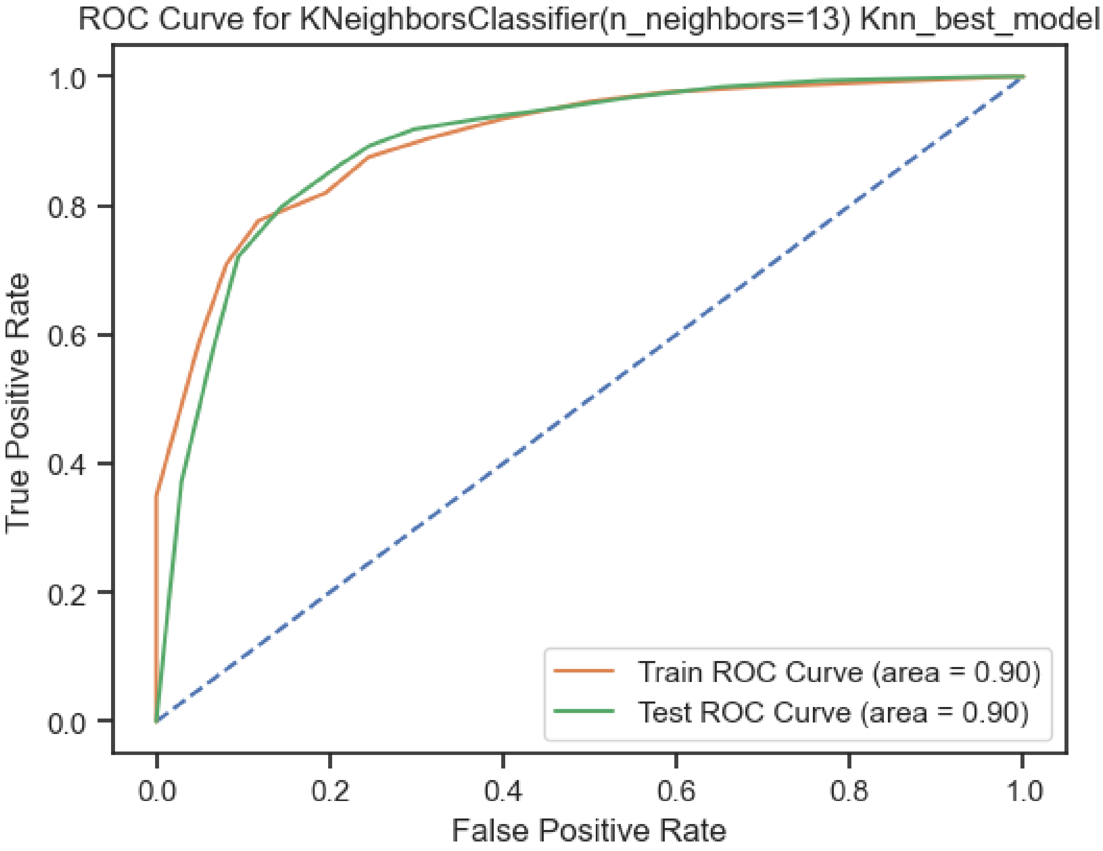
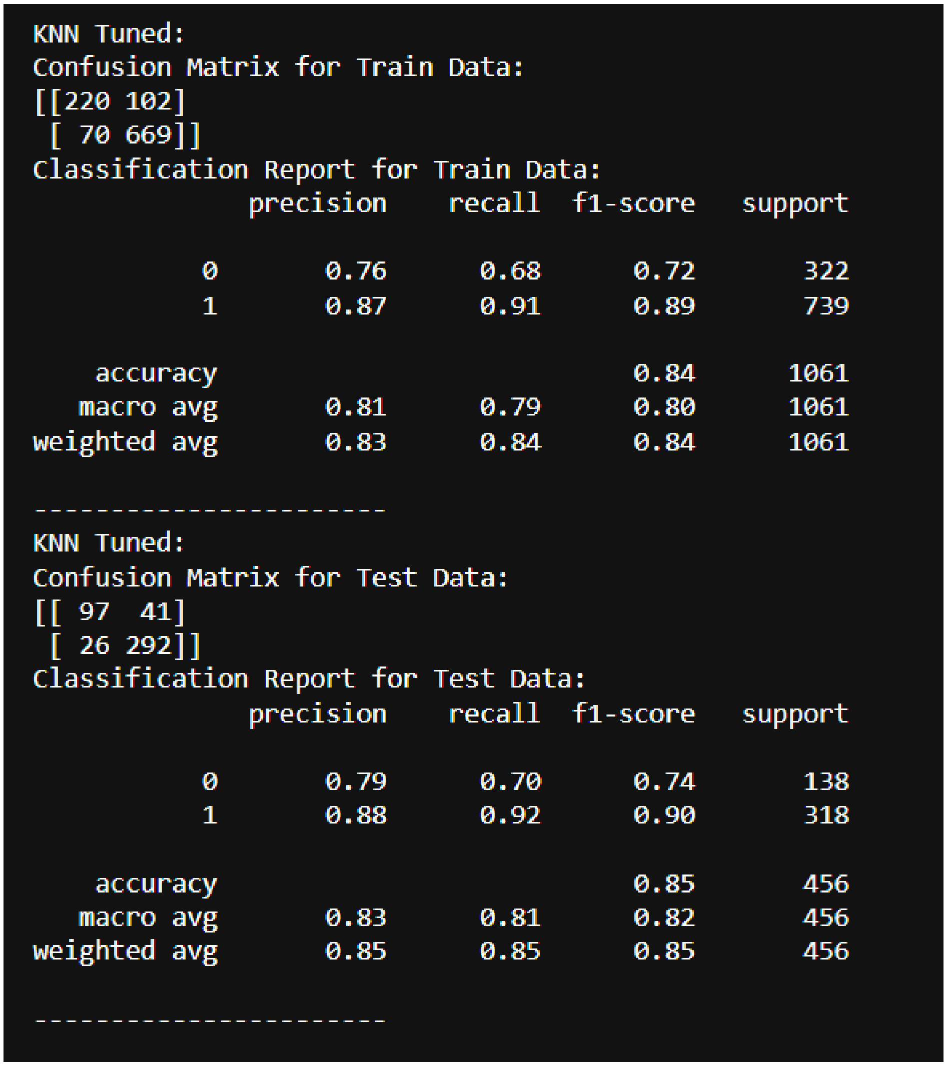


Fig1.6.2.1 KNN Tuned – ROC Curve



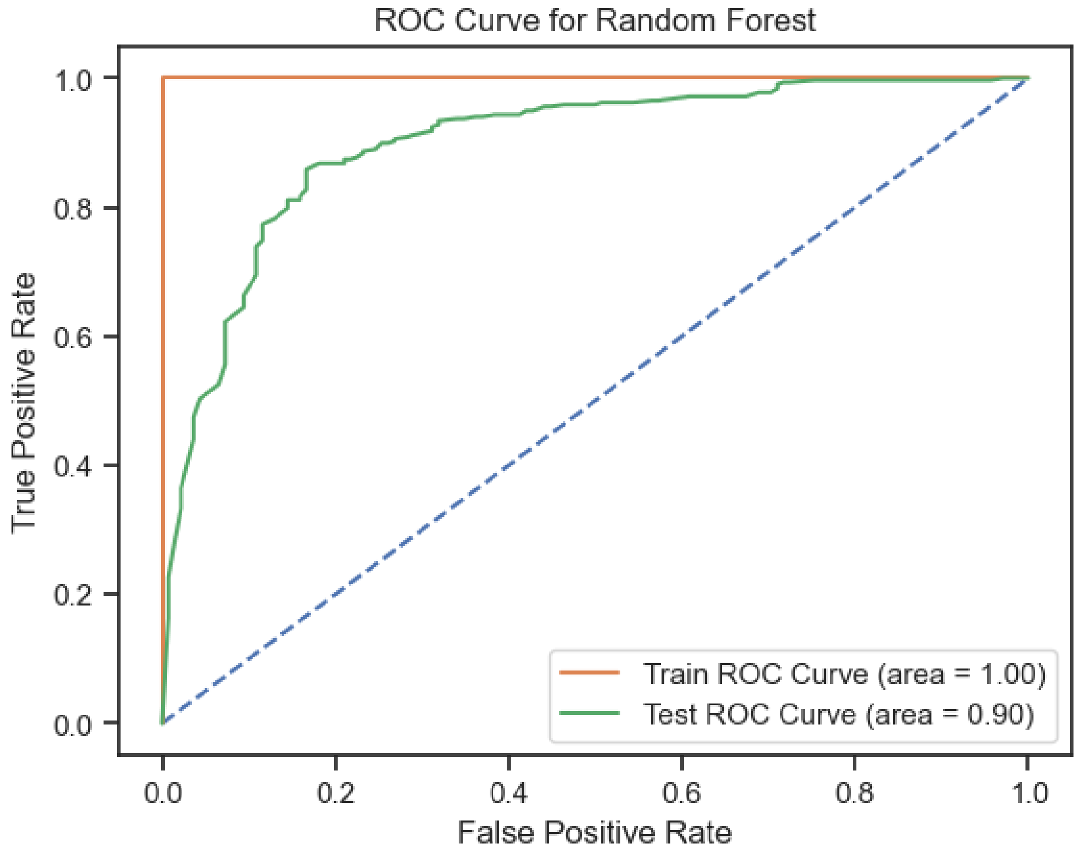
```
KNN Tuned:
Confusion Matrix for Train Data:
[[220 102]
 [ 70 669]]
Classification Report for Train Data:
              precision    recall  f1-score   support

           0       0.76      0.68      0.72       322
           1       0.87      0.91      0.89       739

    accuracy                           0.84      1061
   macro avg       0.81      0.79      0.80      1061
weighted avg       0.83      0.84      0.84      1061

-----------------------
KNN Tuned:
Confusion Matrix for Test Data:
[[ 97  41]
 [ 26 292]]
Classification Report for Test Data:
              precision    recall  f1-score   support

           0       0.79      0.70      0.74       138
           1       0.88      0.92      0.90       318

    accuracy                           0.85       456
   macro avg       0.83      0.81      0.82       456
weighted avg       0.85      0.85      0.85       456

-----------------------
```

Fig1.6.2.2. KNN Tuned - Summary

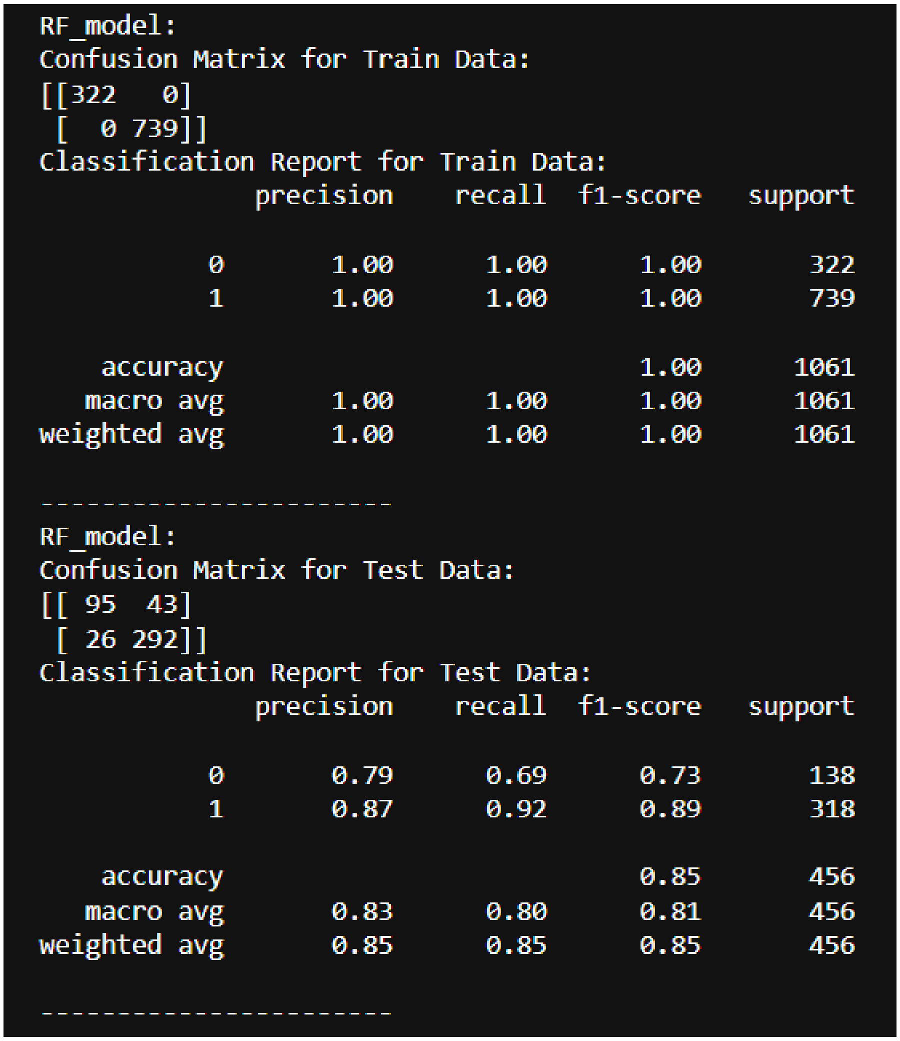
1. Best 'n_neighbors' are given as 13 by the model.

1.6.3 Bagging – Random Forest & Random Forest Tuned

Without any hyper parameters below is the ROC_Curve plot.



Below is the Model Summary:



```
RF_model:
Confusion Matrix for Train Data:
[[322   0]
 [  0 739]]
Classification Report for Train Data:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       322
           1       1.00      1.00      1.00       739

    accuracy                           1.00      1061
   macro avg       1.00      1.00      1.00      1061
weighted avg       1.00      1.00      1.00      1061

-----------------------
RF_model:
Confusion Matrix for Test Data:
[[ 95  43]
 [ 26 292]]
Classification Report for Test Data:
              precision    recall  f1-score   support

           0       0.79      0.69      0.73       138
           1       0.87      0.92      0.89       318

    accuracy                           0.85       456
   macro avg       0.83      0.80      0.81       456
weighted avg       0.85      0.85      0.85       456

-----------------------
```

<u>Hyperparameters:</u>

1. Base estimator is set as RandomForestClassifier() i.e. The Random Forest classifier is used which is an ensemble method that combines multiple decision trees to make predictions.
2. N_estimators determines the number of base estimators (Random Forests) to be included in the ensemble. The grid search is performed with values [50, 100, 200]. Increasing the number of estimators can improve the model's performance, but it also increases computational complexity. The optimal value is typically determined through cross-validation.
3. max_features: This parameter determines the maximum number of features to consider for each base estimator. The grid search is performed with values [None, 'sqrt', 'log2', 0.5]. Here's the significance of each value:
   - None: It considers all features for each base estimator.
   - 'sqrt': It considers the square root of the total number of features.
   - 'log2': It considers the logarithm base 2 of the total number of features.
   - 0.5: It considers half of the total number of features.

The purpose of limiting the number of features is to reduce the correlation among base estimators and improve the diversity of the ensemble, leading to better performance. The choice of the best value depends on the specific dataset and problem.

4. cv: This parameter specifies the number of folds for cross-validation during the grid search. In this case, cv=5 indicates 5-fold cross-validation, where the training data is divided into 5 subsets, and the model is trained and evaluated on different combinations of these subsets.

The performance metrics are shown below.

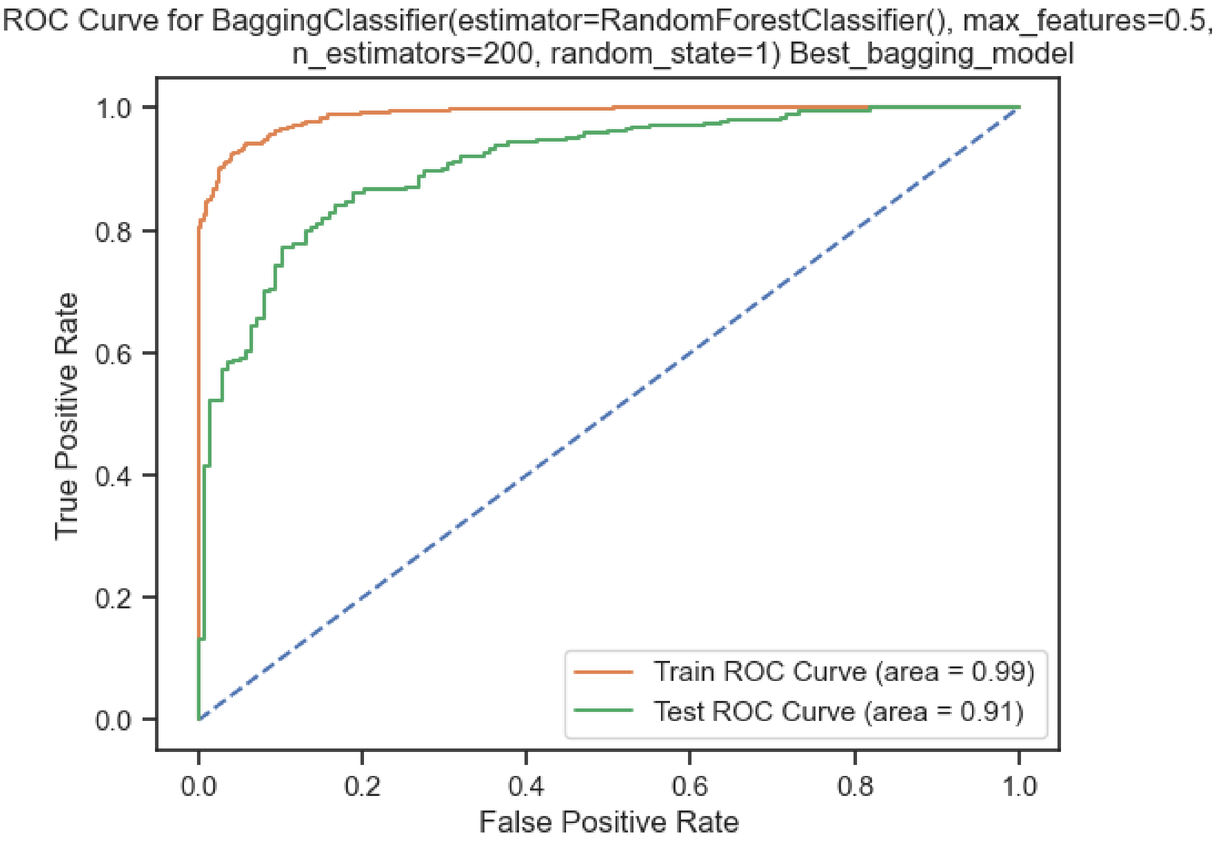
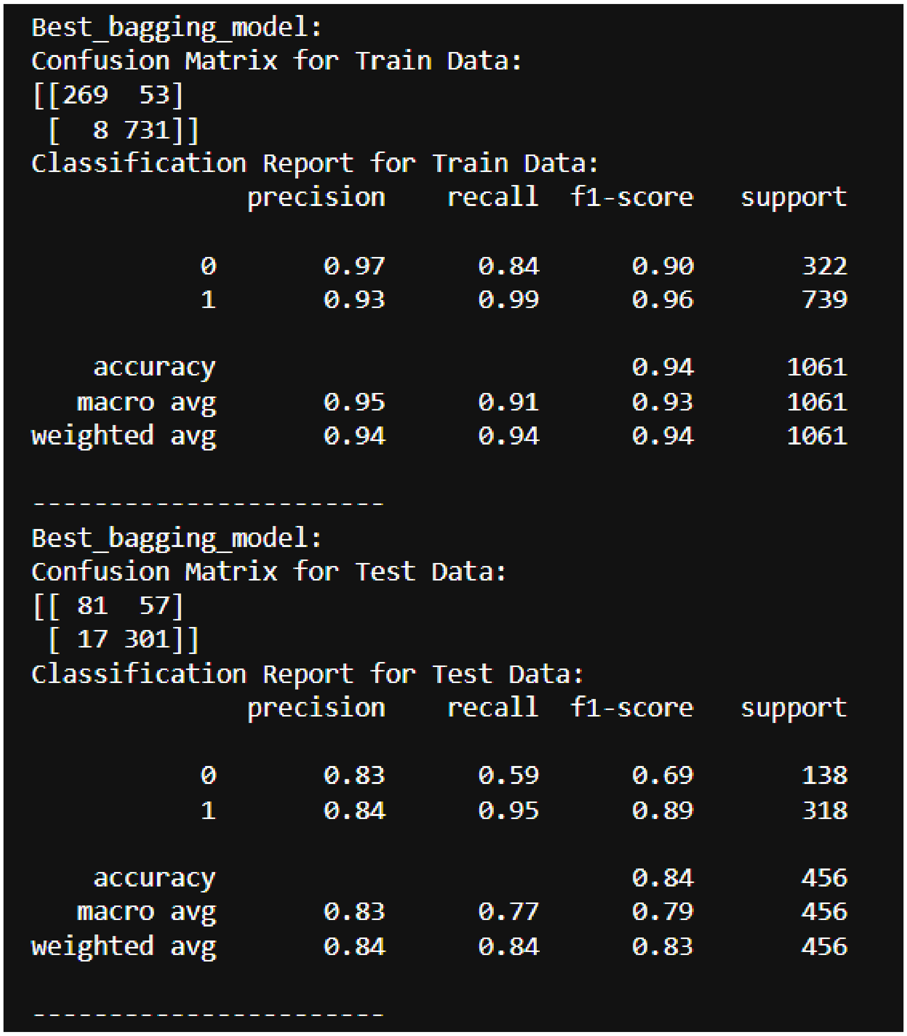


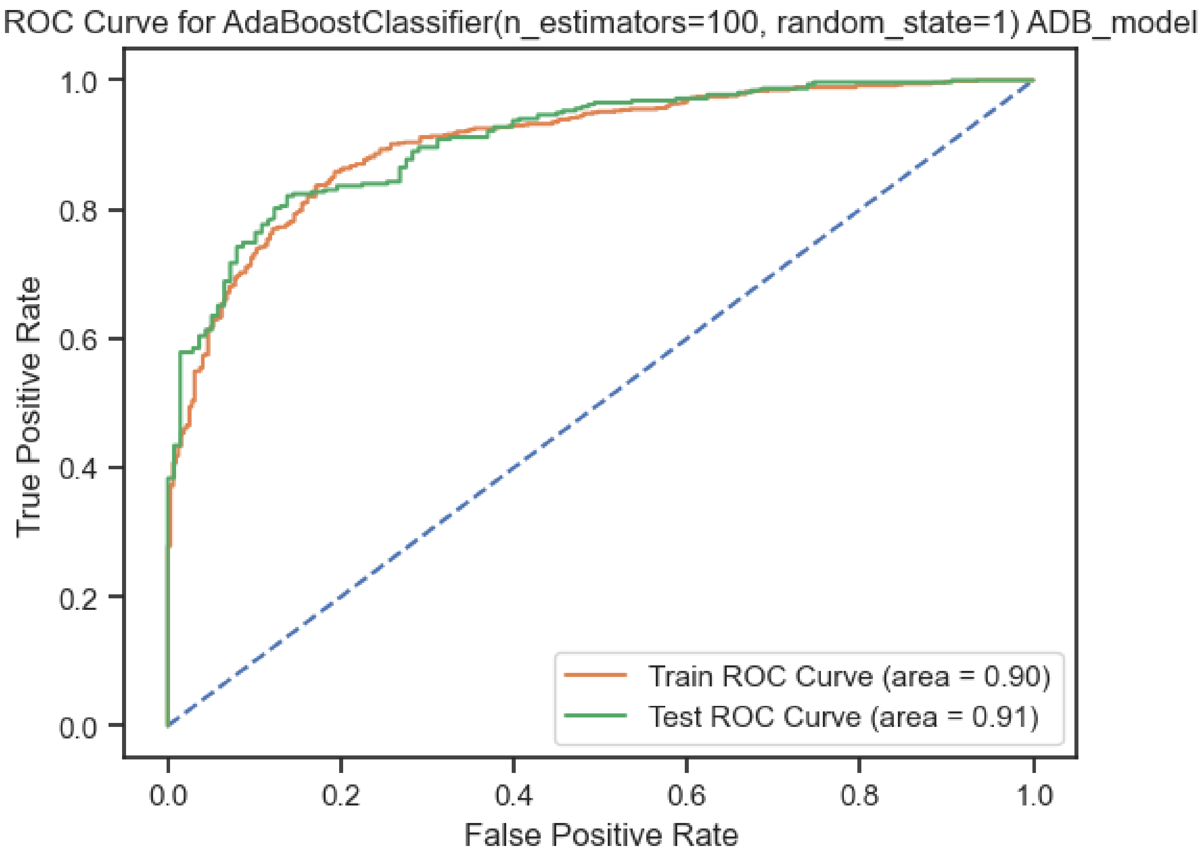
Fig1.6.3.3 Random Forest Tuned – ROC Curve

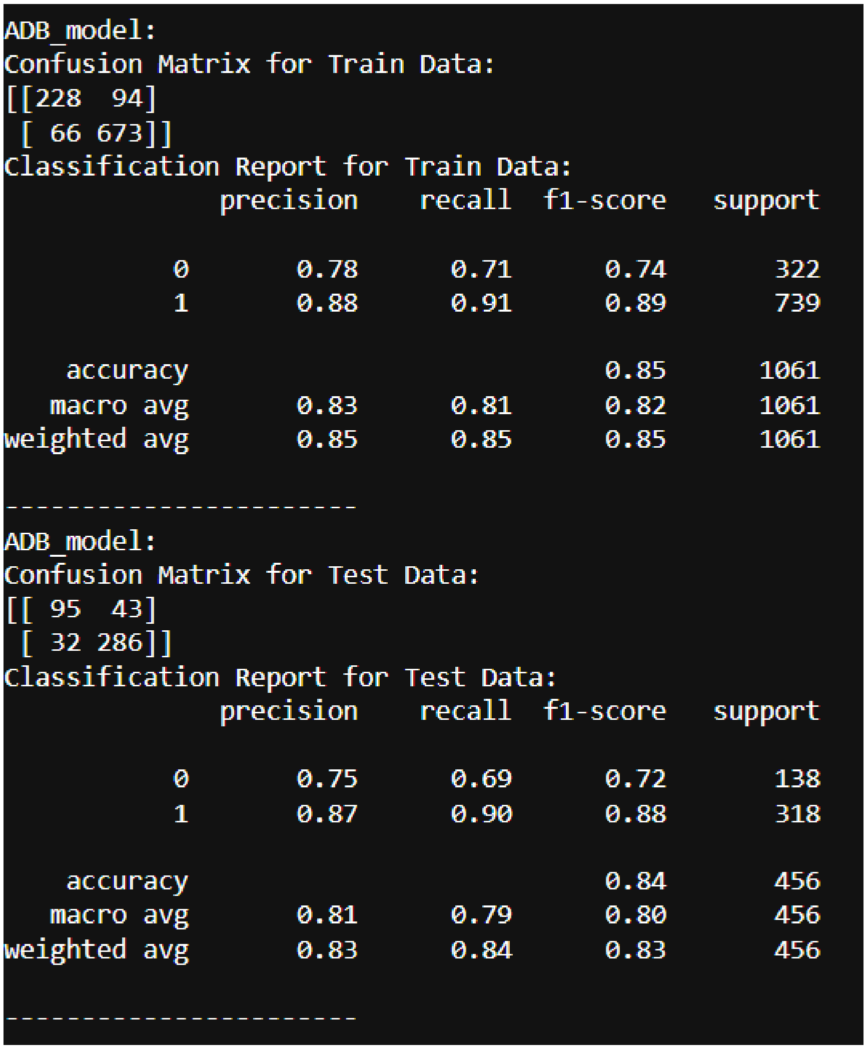


```
Best_bagging_model:
Confusion Matrix for Train Data:
[[269  53]
 [  8 731]]
Classification Report for Train Data:
              precision    recall  f1-score   support

           0       0.97      0.84      0.90       322
           1       0.93      0.99      0.96       739

    accuracy                           0.94      1061
   macro avg       0.95      0.91      0.93      1061
weighted avg       0.94      0.94      0.94      1061

-----------------------
Best_bagging_model:
Confusion Matrix for Test Data:
[[ 81  57]
 [ 17 301]]
Classification Report for Test Data:
              precision    recall  f1-score   support

           0       0.83      0.59      0.69       138
           1       0.84      0.95      0.89       318

    accuracy                           0.84       456
   macro avg       0.83      0.77      0.79       456
weighted avg       0.84      0.84      0.83       456

-----------------------
```

Fig1.6.3.4 Random Forest Tuned – Summary

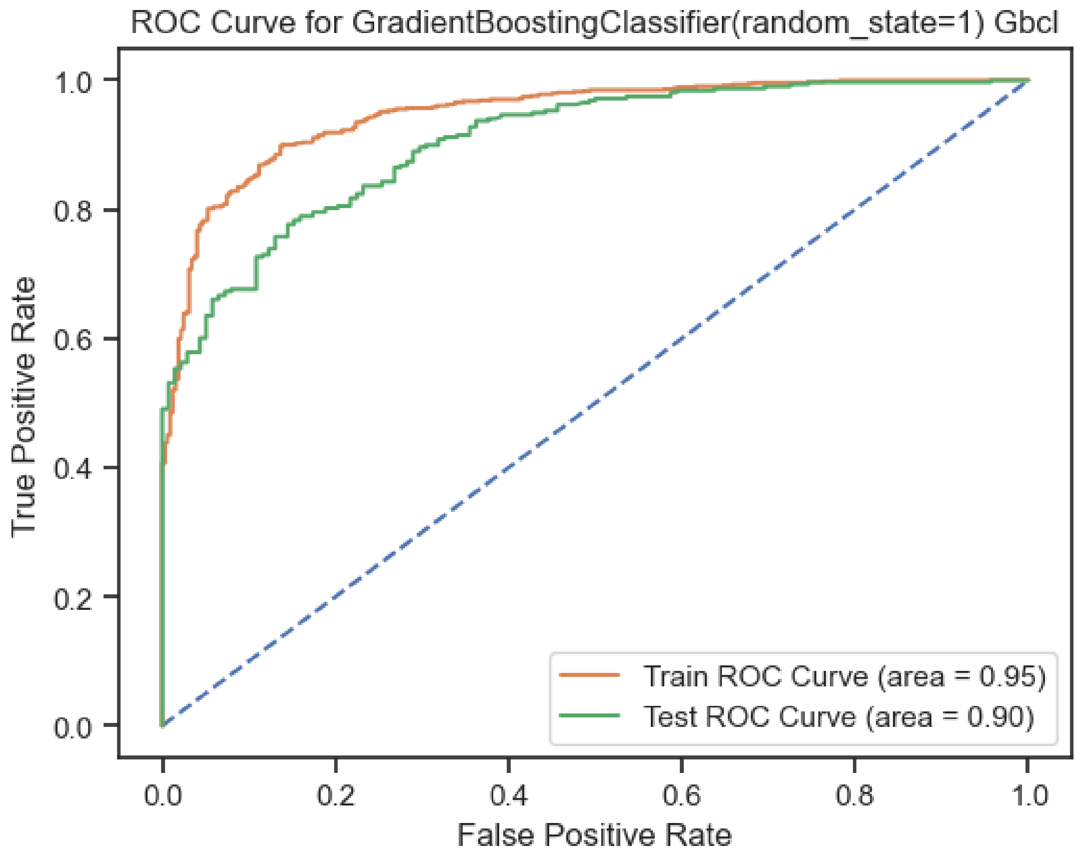
1.6.4 Boosting – Adaboost & Adaboost Tuned

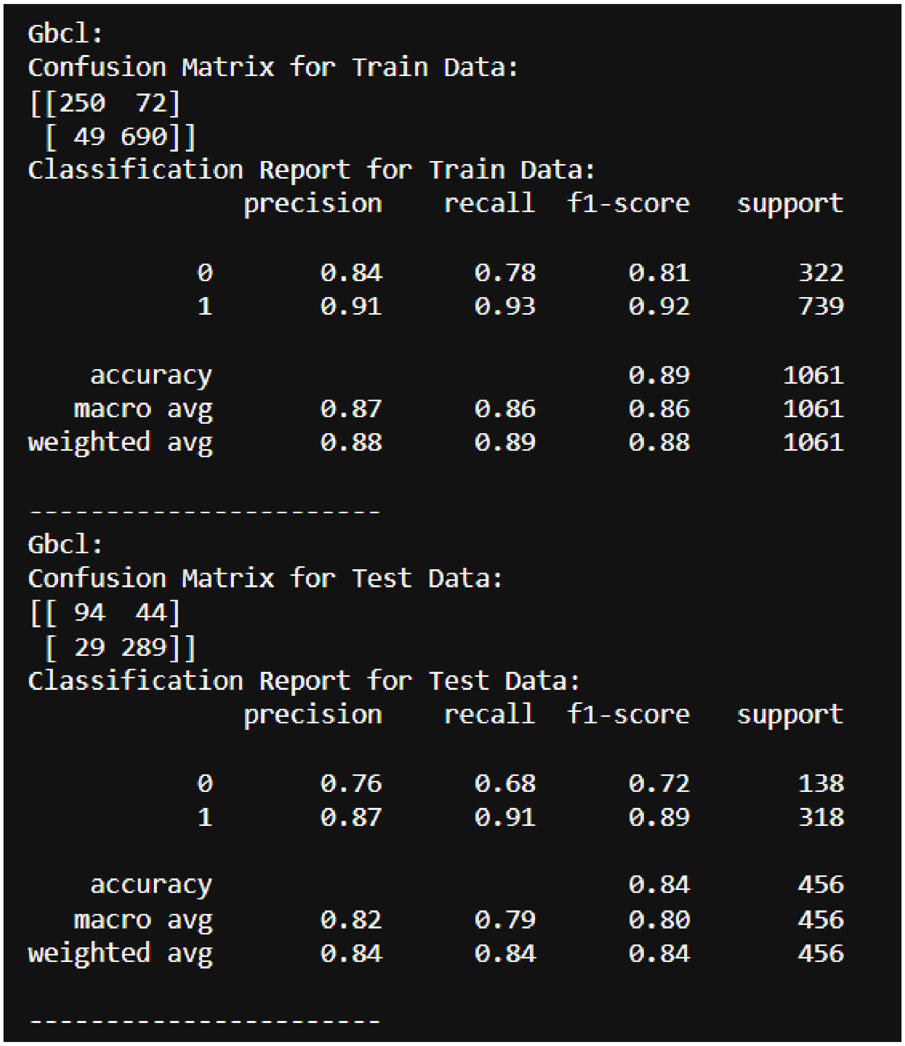

ROC Curve for AdaBoostClassifier(n_estimators=100, random_state=1) ADB_model
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
Train ROC Curve (area = 0.90)
Test ROC Curve (area = 0.91)



```
ADB_model:
Confusion Matrix for Train Data:
[[228  94]
 [ 66 673]]
Classification Report for Train Data:
              precision    recall  f1-score   support

           0       0.78      0.71      0.74       322
           1       0.88      0.91      0.89       739

    accuracy                           0.85      1061
   macro avg       0.83      0.81      0.82      1061
weighted avg       0.85      0.85      0.85      1061

-----------------------
ADB_model:
Confusion Matrix for Test Data:
[[ 95  43]
 [ 32 286]]
Classification Report for Test Data:
              precision    recall  f1-score   support

           0       0.75      0.69      0.72       138
           1       0.87      0.90      0.88       318

    accuracy                           0.84       456
   macro avg       0.81      0.79      0.80       456
weighted avg       0.83      0.84      0.83       456

-----------------------
```

1.6.5 Gradient Boost


ROC Curve for GradientBoostingClassifier(random_state=1) Gbcl
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
Train ROC Curve (area = 0.95)
Test ROC Curve (area = 0.90)

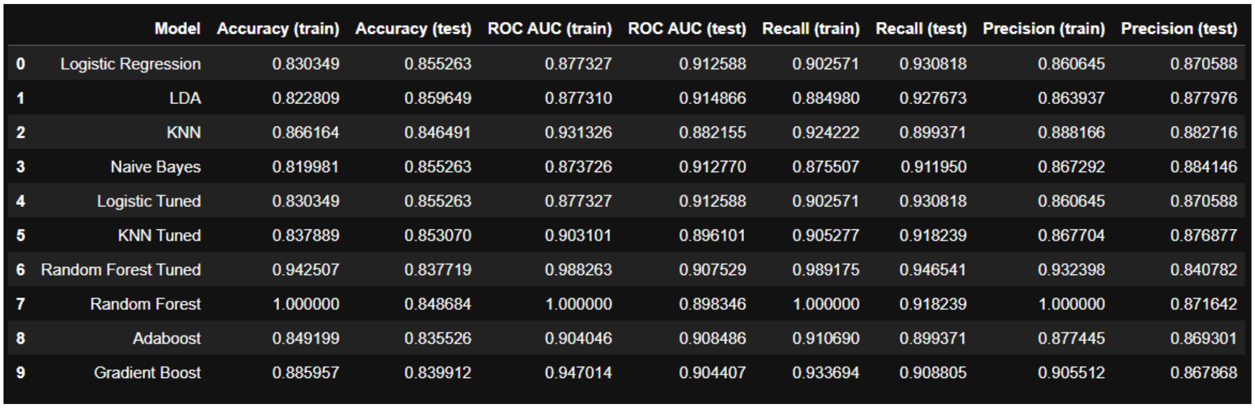


```
Gbcl:
Confusion Matrix for Train Data:
[[250  72]
 [ 49 690]]
Classification Report for Train Data:
              precision    recall  f1-score   support

           0       0.84      0.78      0.81       322
           1       0.91      0.93      0.92       739

    accuracy                           0.89      1061
   macro avg       0.87      0.86      0.86      1061
weighted avg       0.88      0.89      0.88      1061

-----------------------
Gbcl:
Confusion Matrix for Test Data:
[[ 94  44]
 [ 29 289]]
Classification Report for Test Data:
              precision    recall  f1-score   support

           0       0.76      0.68      0.72       138
           1       0.87      0.91      0.89       318

    accuracy                           0.84       456
   macro avg       0.82      0.79      0.80       456
weighted avg       0.84      0.84      0.84       456

-----------------------
```

1.7 PERFORMANCE METRICS

Performance Metrics: Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC_AUC score for each model, classification report (4 pts) Final Model - Compare and comment on all models on the basis of the performance metrics in a structured tabular manner. Describe on which model is best/optimized, After comparison on which model suits the best for the problem in hand on the basis of different measures. Comment on the final model.(3 pts)

Answer:



	Model	Accuracy (train)	Accuracy (test)	ROC AUC (train)	ROC AUC (test)	Recall (train)	Recall (test)	Precision (train)	Precision (test)
0	Logistic Regression	0.830349	0.855263	0.877327	0.912588	0.902571	0.930818	0.860645	0.870588
1	LDA	0.822809	0.859649	0.877310	0.914866	0.884980	0.927673	0.863937	0.877976
2	KNN	0.866164	0.846491	0.931326	0.882155	0.924222	0.899371	0.888166	0.882716
3	Naive Bayes	0.819981	0.855263	0.873726	0.912770	0.875507	0.911950	0.867292	0.884146
4	Logistic Tuned	0.830349	0.855263	0.877327	0.912588	0.902571	0.930818	0.860645	0.870588
5	KNN Tuned	0.837889	0.853070	0.903101	0.896101	0.905277	0.918239	0.867704	0.876877
6	Random Forest Tuned	0.942507	0.837719	0.988263	0.907529	0.989175	0.946541	0.932398	0.840782
7	Random Forest	1.000000	0.848684	1.000000	0.898346	1.000000	0.918239	1.000000	0.871642
8	Adaboost	0.849199	0.835526	0.904046	0.908486	0.910690	0.899371	0.877445	0.869301
9	Gradient Boost	0.885957	0.839912	0.947014	0.904407	0.933694	0.908805	0.905512	0.867868

Fig 1.7 Performance Metrics all models

1. Based on above metrics and performance plots from the models, the best model in terms of accuracy on the test set is the "LDA" (Linear Discriminant Analysis) model with an accuracy of 0.859649. However, accuracy is not the only metric for choosing a model.

2. The "Random Forest Tuned" model achieves the highest ROC AUC score on the training set (0.988263), indicating excellent predictive capability. However, its performance on the test set is slightly lower (0.907529), suggesting a potential overfitting issue.

3. Looking at the recall and precision scores, the "Random Forest Tuned" model shows high values for both metrics on both the training and test sets. This indicates that the model has a good ability to identify positive cases (recall) and has a low false-positive rate (precision). Model can correctly classify positive cases while maintaining a low rate of false positives.

4. Considering the overall analysis, the "Random Forest Tuned" model seems to perform well in terms of various evaluation metrics.

5. However, it is recommended to further evaluate and validate the model's performance using other techniques such as cross-validation and check its performance on unseen data.

1.8 BUSINESS REPORT

Based on your analysis and working on the business problem, detail out appropriate insights and recommendations to help the management solve the business objective. There should be at least 3-4 Recommendations and insights in total. Recommendations should be easily understandable a nd business specific, students should not give any technical suggestions. Full marks should only be allotted if the recommendations are correct and business specific.

Answer:

1. Recommended to use the "Random Forest Tuned" model as it is the top-performing model, considering its high accuracy, ROC AUC, recall, and precision scores. Understand the overfitting issue.

2 PROBLEM 2: TEXT ANALYSIS

In this problem, we are going to work on the ‘inaugural’ corpus from ‘nltk’. From the ‘inaugural’ corpus, we will be looking at the following speeches of the Presidents of the United States of America:

1. President Franklin D. Roosevelt in 1941
2. President John F. Kennedy in 1961
3. President Richard Nixon in 1973

DATA DESCRIPTION

Dataset for Problem 2 is Speeches. Data Description is given below:

1. ‘NLTK’ has 'inaugural' corpus that contains the inaugural speeches given by U.S.A Presidents.
2. In the project we are working with *three* different inaugural speeches from the nltk ‘inaugural’ corpus : the speech by Franklin D. Roosevelt in 1941, the speech by John F. Kennedy in 1961, and the speech by Richard Nixon in 1973 which will be the corpus for our project.
3. The corpus contains the raw text of the specified speeches.

2.1 EXPLORATORY DATA ANALYSIS

Find the number of characters, words and sentences for the mentioned documents. (Hint: use .words(), .raw(), .sent() for extracting counts).

Answer:

2.1.1 Sample Data

Using nltk’s ‘sent_tokenize’, the sample 3 sentences of each president’s speeches are extracted. The sample sentences are as below:

Sample sentences of Roosevelt's Speech:

1. 'On each national day of inauguration since 1789, the people have renewed their sense of dedication to the United States.'
2. ‘In Washington's day the task of the people was to create and weld together a nation.’
3. ‘In Lincoln's day the task of the people was to preserve that Nation from disruption from within.’

Sample sentences of Kennedy's Speech:

1. ‘Vice President Johnson, Mr. Speaker, Mr. Chief Justice, President Eisenhower, Vice President Nixon, President Truman, reverend clergy, fellow citizens, we observe today not a victory of party, but a celebration of freedom -- symbolizing an end, as well as a beginning -- signifying renewal, as well as change.’

2. 'For I have sworn I before you and Almighty God the same solemn oath our forebears I prescribed nearly a century and three quarters ago.'
3. 'The world is very different now.'

Sample sentences of Nixon's Speech:

1. ‘Mr. Vice President, Mr. Speaker, Mr. Chief Justice, Senator Cook, Mrs. Eisenhower, and my fellow citizens of this great and good country we share together:\n\nWhen we met here four years ago, America was bleak in spirit, depressed by the prospect of seemingly endless war abroad and of destructive conflict at home.'
2. 'As we meet here today, we stand on the threshold of a new era of peace in the world.'
3. 'The central question before us is: How shall we use that peace?'

Inference: Roosevelt starts with the historical tradition. Kennedy’s starts with celebration and a change. Nixon speech is formal and is about reality of nation.

2.1.2 Data Statistical Description

Text analysis is performed on the raw data. Below is the summary of number of characters, numb er of words, number of sentences from each president’s speech (Data cleanup is not performed y et):

1941-Roosevelt.txt:
No. of Characters: 7571
No. of Words: 1536
No. of Sentences: 68

1961-Kennedy.txt:
No. of Characters: 7618
No. of Words: 1546
No. of Sentences: 52

1973-Nixon.txt:
No. of Characters: 9991
No. of Words: 2028
No. of Sentences: 69

2.1.3 Univariate Analysis

Nltk’s ‘word_tokenize’ is used to tokenize the words from raw speech. Using nltk’s standard ‘stopwords’, the stopwords from the speeches are removed. Only words are considered and then lowercase is implemented on all words from the 3 speeches. (Data cleanup is done without stemming or lemmatization to get first-idea of how data is.) Now, a barchat is plotted on top 5 frequent words from each speech. The visualization is shown below:

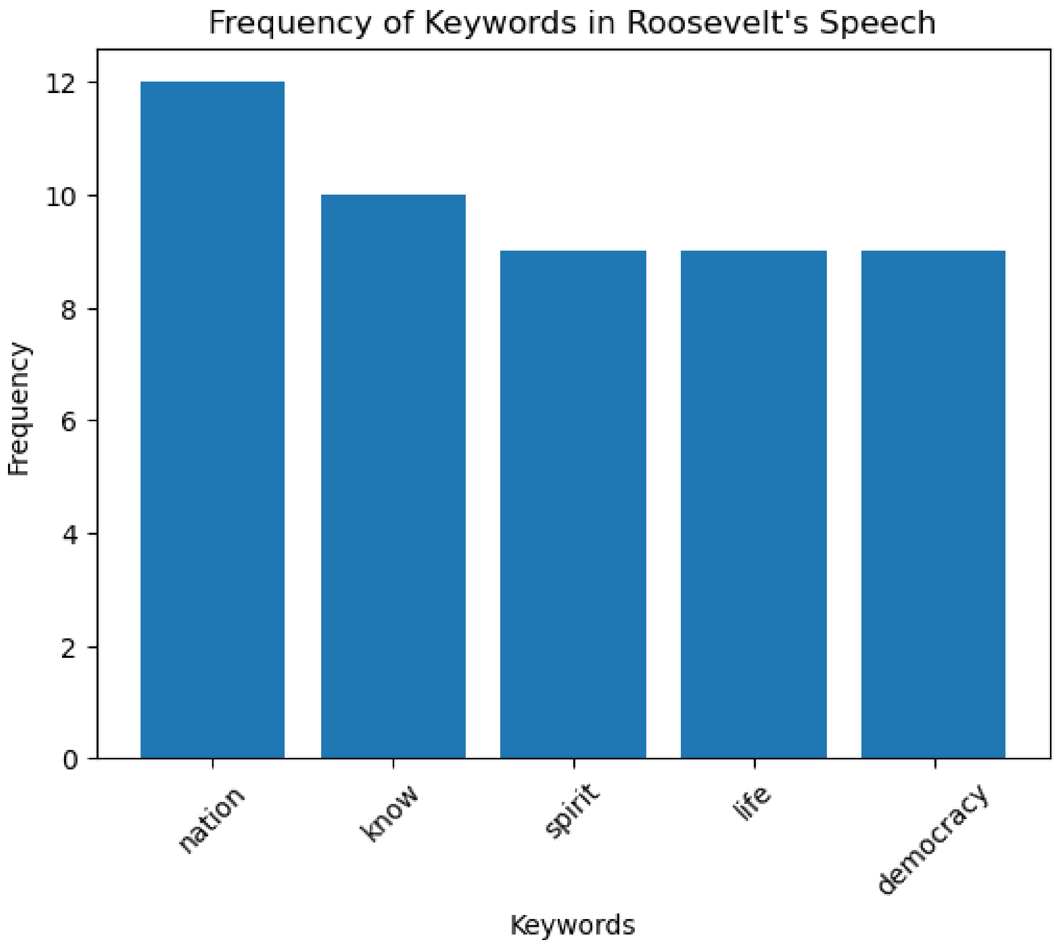


Fig2.1.3.1 Frequency of Keywords in Roosevelt's Speech before lemmatization

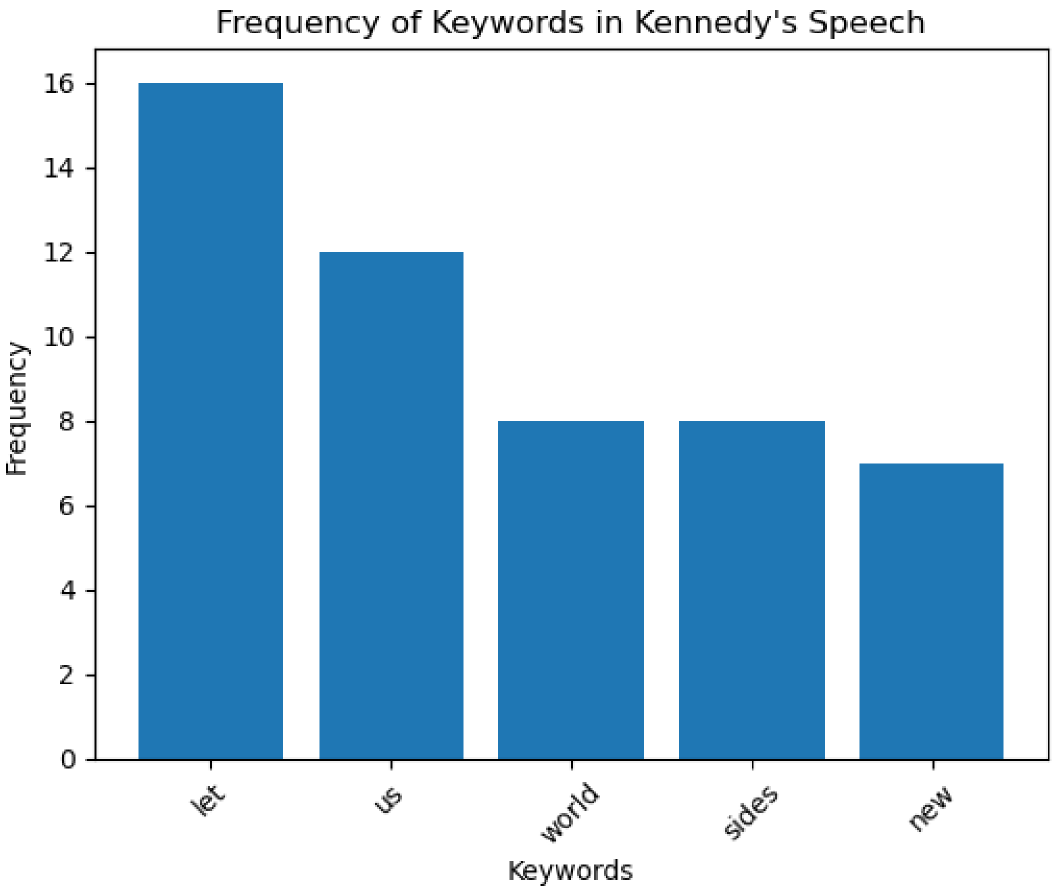


Fig2.1.3.2 Frequency of Keywords in Kennedy's Speech before lemmatization

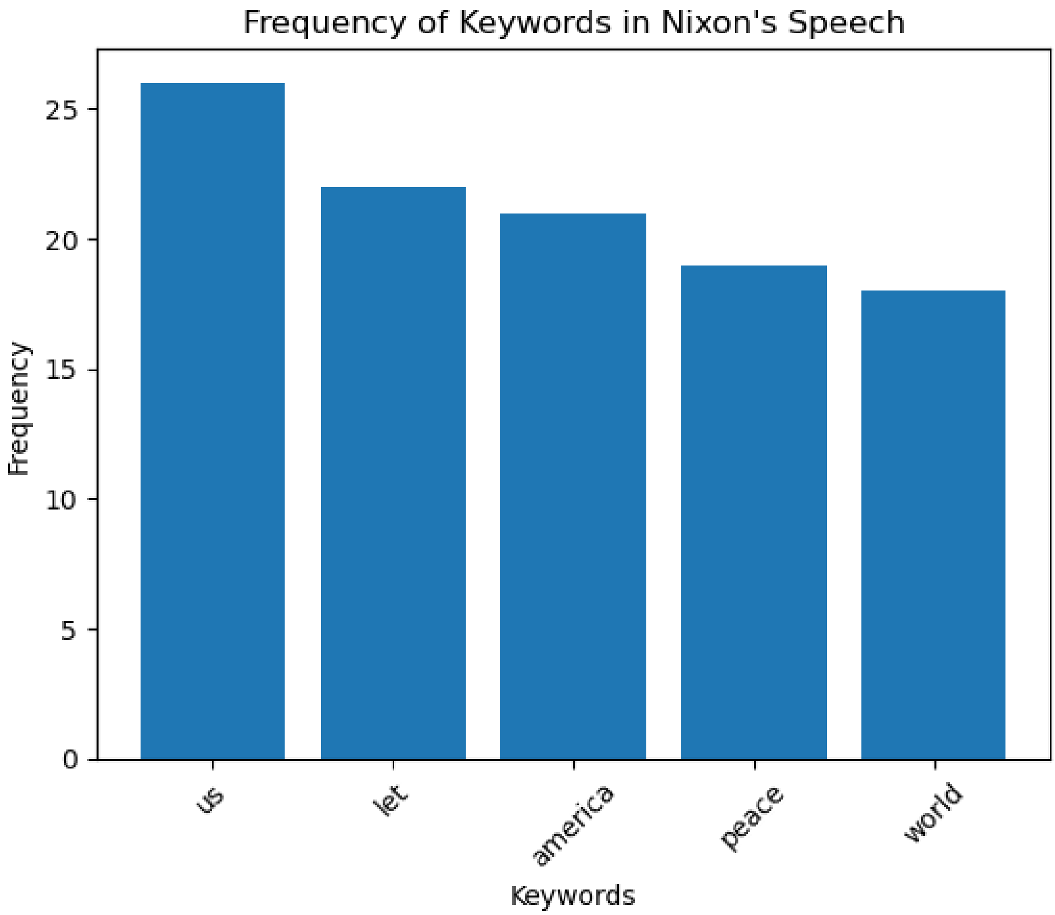


Fig2.1.3.3 Frequency of Keywords in Nixon's Speech before lemmatization

2.2 PREPARE DATA FOR WORDCLOUD

Remove all the stopwords from the three speeches. Show the word count before and after the removal of stopwords. Show a sample sentence after the removal of stopwords.

Answer:

2.2.1 Stopwords

'Stopwords' are downloaded from the nltk package and the standard 'english' language stopwords are used to clean the data. Punctuation is cleaned from the data. Numbers are used rarely in these 3 speeches; however all numbers are also removed from the data.

The below summary shows the count of words before and after removing stopwords/punctuation/numbers from the text:

Roosevelt words count before and after removing stopwords are : 1536 & 627 resepctively.
Kennedy words count before and after removing stopwords are : 1546 & 692 resepctively.

Nixon words count before and after removing stopwords are : 2028 & 832 resepctively.

The sample sentences from each president's speech after removing the stopwords and punctuation & numbers are provided below:

Sample sentence after removing stopwords (1941-Roosevelt.txt):

national day inauguration since people renewed sense dedication united states washington day task people create weld together nation lincoln day

Sample sentence after removing stopwords (1961-Kennedy.txt):

vice president johnson mr speaker mr chief justice president eisenhower vice president nixon president truman reverend clergy fellow citizens observe

Sample sentence after removing stopwords (1973-Nixon.txt):

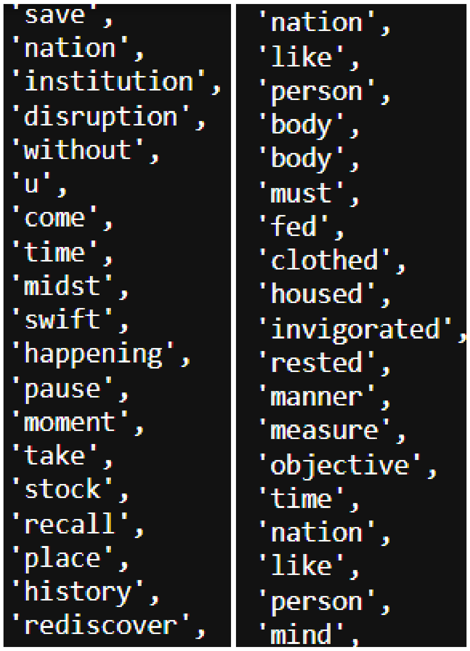
mr vice president mr speaker mr chief justice senator cook mrs eisenhower fellow citizens great good country share together met

Note:

Consider extending words like 'mr', 'United States', 'America' to the Stopwords list. 'USA', 'US', 'United States', 'America' are all variations of the same country which also is similar to 'nation', 'country'. These should be handled and then extending these to stopwords list will be based on the purpose of the text analysis. If the speeches are analyzed for domestic or for global impact. For now, these are not implemented.

2.2.2 Lemmatization

Standard 'WordNetLemmatizer' from the nltk.stem is used. Below is a sample of the words after applying standard lemmatization.

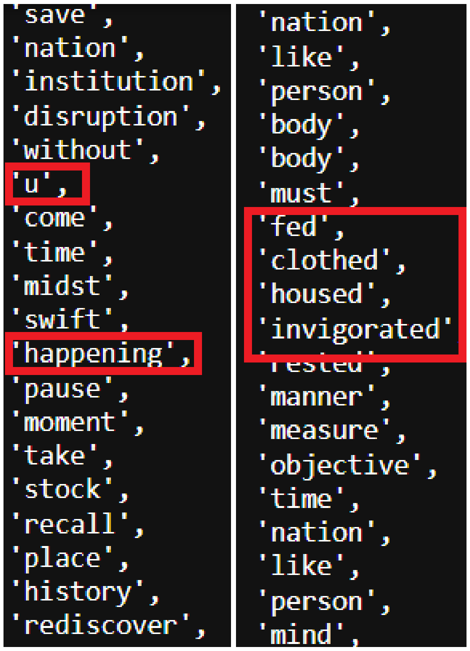


```
'save',
'nation',
'institution',
'disruption',
'without',
'u',
'come',
'time',
'midst',
'swift',
'happening',
'pause',
'moment',
'take',
'stock',
'recall',
'place',
'history',
'rediscover',
```

```
'nation',
'like',
'person',
'body',
'body',
'must',
'fed',
'clothed',
'housed',
'invigorated',
'rested',
'manner',
'measure',
'objective',
'time',
'nation',
'like',
'person',
'mind',
```

Fig2.2.2.1 Standard Lemmatizer

Verbs are not reduced to the base form. Words like 'us' are reduced to 'u' as they were considered as plural. These are shown in figure below.



```
'save',
'nation',
'institution',
'disruption',
'without',
'u',
'come',
'time',
'midst',
'swift',
'happening',
'pause',
'moment',
'take',
'stock',
'recall',
'place',
'history',
'rediscover',
```

```
'nation',
'like',
'person',
'body',
'body',
'must',
'fed',
'clothed',
'housed',
'invigorated'
'rested',
'manner',
'measure',
'objective',
'time',
'nation',
'like',
'person',
'mind',
```

Fig2.2.2.2 Standard Lemmatizer output showing areas of improvement

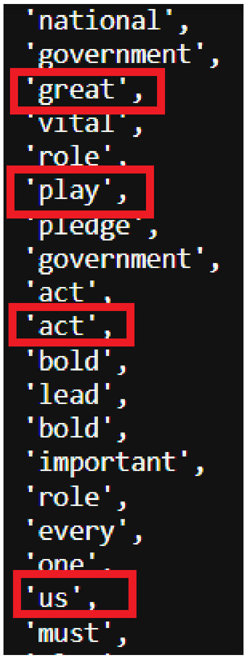
Observation from standard lemmatization:

1. 'Us' does not have to be lemmatized as 'u' and a rule can be added to skip lemmatizing 'us', thus preserving the word.

2. Roman Numbers are used like ‘ii’ from the ‘world war 2’. In this specific example, the frequency of such Roman Numbers is very small. Such roman numbers can be ignored in this case.
3. The verbs can be further lemmatized to base form using 'pos_tag'.

To address above issues, convert the POS tags to WordNet-compatible POS tags. This helps in obtaining the base or dictionary form of the words, which can improve the quality of the lemmatized output.

A sample lemmatized word output from Nixon’s speech looks like this now:



```
'national',
'government',
'great',
'vital',
'role',
'play',
'pledge',
'government',
'act',
'act',
'bold',
'lead',
'bold',
'important',
'role',
'every',
'one',
'us',
'must',
```

Fig2.2.2.3 POS_tag Lemmatizer

‘us’ is preserved. Many verbs are now in base form. There is however improvement needed to make sure more words like ‘us’ are preserved from original speech. There is also work needed to convert adjectives, adverbs & verbs to base/dictionary form.

<u>Sample sentence after lemmatization Roosevelt's speech:</u>

national day inauguration since people renew sense dedication unite state washington day task people create weld together nation lincoln day

<u>Sample sentence after lemmatization in Kennedy's speech:</u>

vice president johnson mr speaker mr chief justice president eisenhower vice president nixon president truman reverend clergy fellow citizen observe

<u>Sample sentence after lemmatization in Nixon's speech:</u>

mr vice president mr speaker mr chief justice senator cook mr eisenhower fellow citizens great good country share together meet

2.3 FREQUENT WORDS

Which word occurs the most number of times in his inaugural address for each president? Mention the top three words. (after removing the stopwords) 3

Answer:

Top 3 frequent words and the frequency with which the words are repeated is shown in the summary below:

Top 3 words from 1941-Roosevelt.txt:

nation : 15
life : 10
know : 10

Top 3 words from 1961-Kennedy.txt:

let : 16
us : 12
world : 8

Top 3 words from 1973-Nixon.txt:

us : 26
let : 22
america : 21

2.4 WORD CLOUD

Plot the word cloud of each of the three speeches. 3

Answer:

Below word cloud visually shows words used for each speech

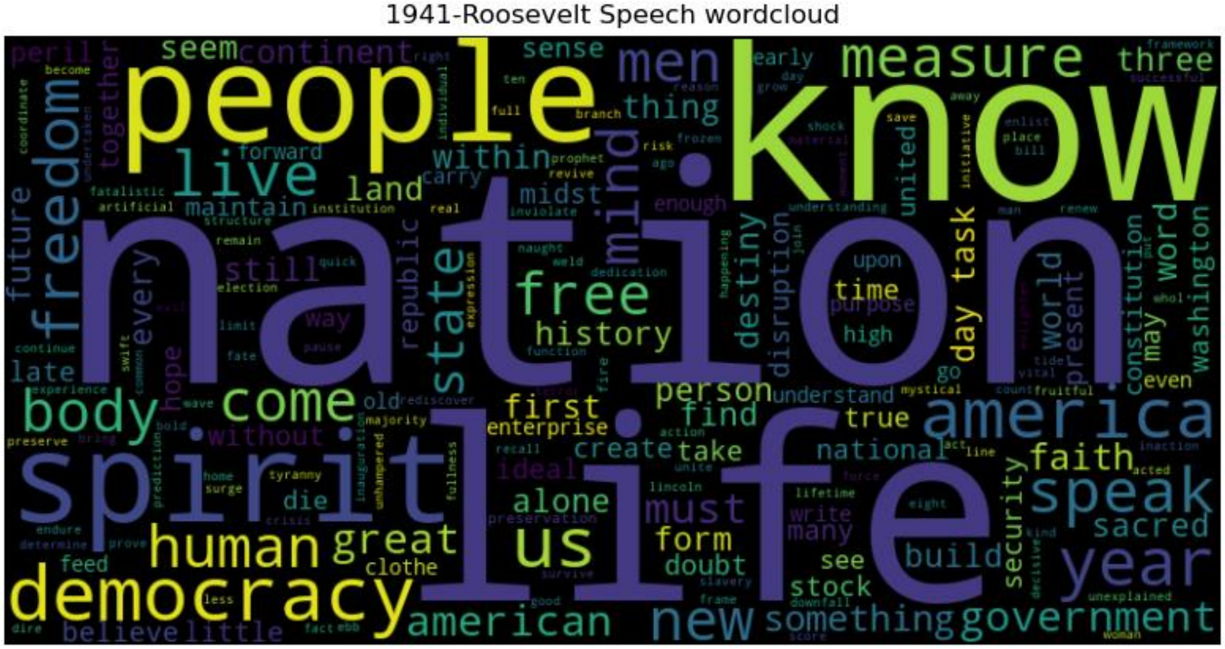


Fig2.4.1 Wordcloud- Roosevelt's speech

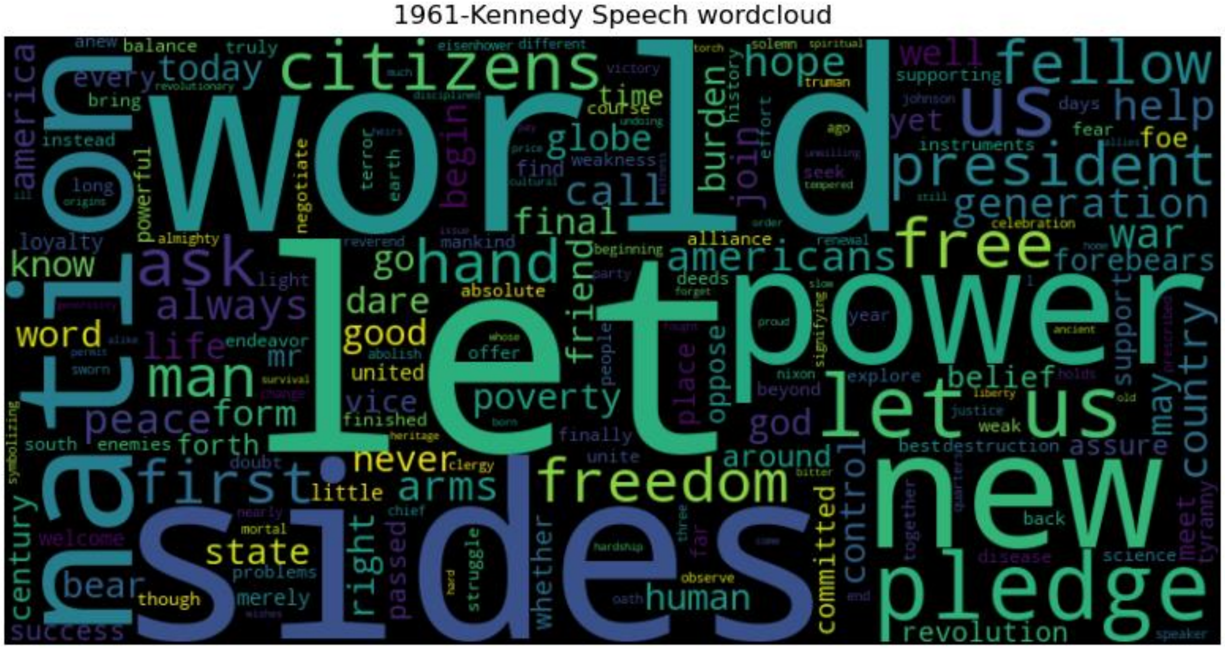


Fig2.4.2 Wordcloud- Kennedy's speech

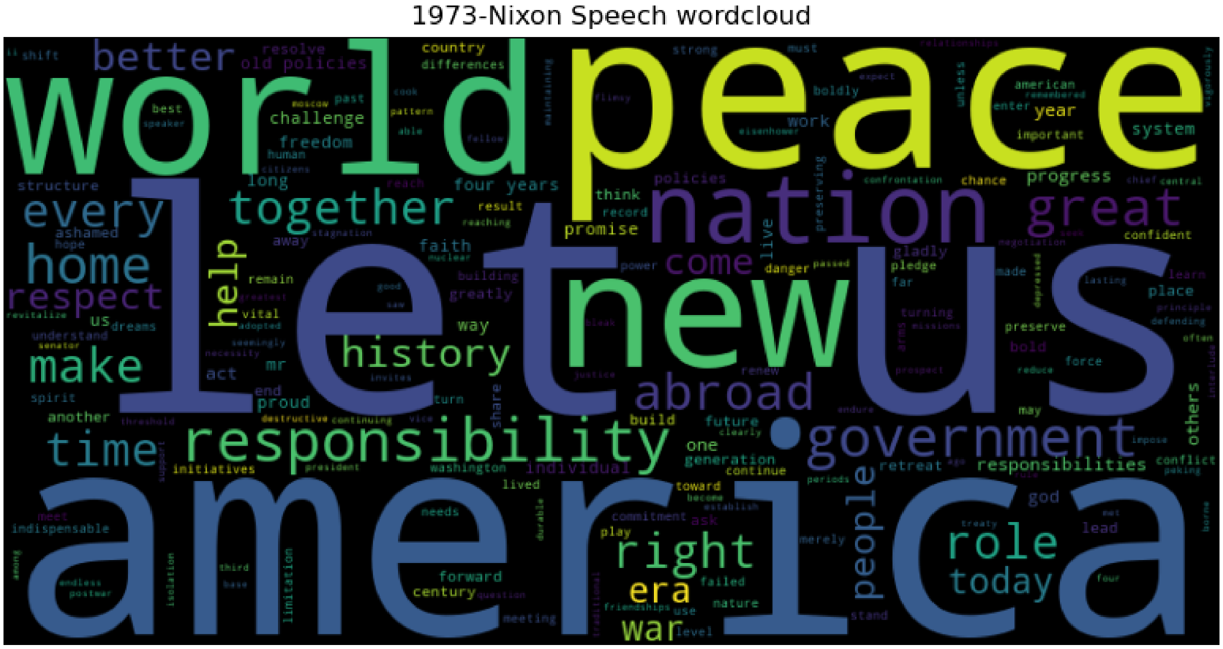
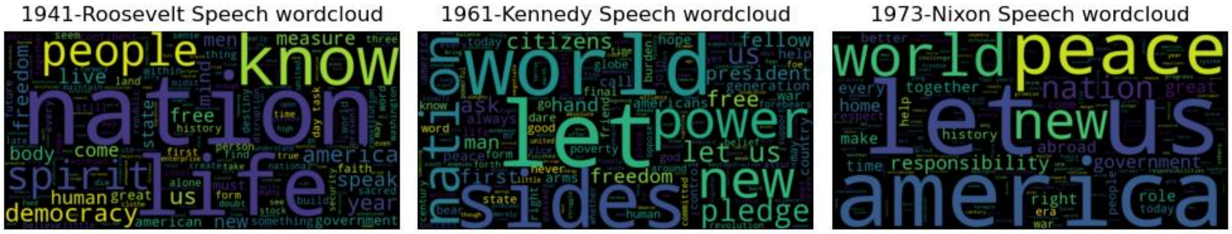

1973-Nixon Speech wordcloud

Fig2.4.3 Wordcloud- Nixon's speech

Quick look at the wordclouds of 3 speeches is below:


1941-Roosevelt Speech wordcloud

1961-Kennedy Speech wordcloud

1973-Nixon Speech wordcloud

Fig2.4.4 Wordcloud Quick look - All speeches

Based on the word cloud, below can be inferred:

Roosevelt's speech:

- Key words: nation, life, know, people, spirit, democracy, us, year, america, live.
- Roosevelt's speech talks about the spirit of the nation and the speech inspired the audience to 'know', emphasises on 'nation', 'spirit', 'life', 'people'.

Kennedy's speech:

- Key words: let, us, world, side, power, new, nation, pledge, ask, citizen.
- Insights: The speech seems bold with words like 'dare', 'power'. It calls for collective action for the nation to come together. The speech has an aspirational tone, calling 'citizens' to actively participate. Speech seems to set the stage or talk about the 'new world' 'power'.

Nixon's speech:

- Key words: us, let, america, peace, world, responsibility, new, nation, great, make.
- Highlights ‘world’, ‘peace’ and talks about the desire to make positive impact on world. Especially america’s role in it with words like ‘responsibility’.

The speeches are inline with America’s role, global actions, global policies ever since.

!!! Major note & caution - only so much can be inferred about tone & mood of the speeches from a wordcloud. Deeper analysis with access to history is needed to arrive at a more nuanced and correct inference.

3 REFLECTION REPORT

Please reflect on all that you learnt and fill this reflection report. You have to copy the link and paste it on the URL bar of your respective browser. https://docs.google.com/forms/d/e/1FAIpQLScKuVyrmTTM7Pboh0IB4YIBUbJp2NrDZcsY4SCRn3ZUkwmLGg/viewform

<Completed>