MACHINE LEARNING

- Election Prediction Analysis
- Speeches of the Presidents Analysis

Election Prediction Analysis

Table of Contents

Table List:

Figure List

Problem Statement 1:

You are hired by one of the leading news channels CNBE who wants to analyse 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 Dictionary for Election Analyse:

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 Read the dataset. Do the descriptive statistics and do the null value condition check. Write an inference on it.

The objectives of EDA can be summarized as follow:

- Maximize insight into the data/understand the data structure.
- EDA is an approach to analyse date using non-visual and visual techniques.
- EDA involves through analyse of data to understand the current business situation.
- EDA objective is to extract "'Gold' from the "'Data mine'' based on domain understanding.

As a first step, importing all the necessary libraries, we think that will be requiring to perform the EDA.

Loading the data set – Loading the ' Election_Data.xlsx ' file using pandas. For this we will be using read excel file.

Following is the output from Jupyter.

Head of the dataset: After reading the CSV file, the head command gives the below output.

	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

Table 1: Sample Data

From the above head table, we infer that,

Dataset contains of 10 variables such as 'Unnamed: 0', 'vote', 'age', 'economic.cond.national', 'economic.cond.household', 'Blair', 'Hague', 'Europe', 'political.knowledge', 'gender'. Variable 'Unnamed: 0' is not useful for our analysis and it will be dropped in future.

There are 9 Independent variables and 1 dependent variable as 'Vote'.

info() is used to check the Information about the data and the datatypes of each respective attributes: Output from Info command is –

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1525 entries, 0 to 1524
Data columns (total 9 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   vote                    1525 non-null   object
 1   age                     1525 non-null   int64
 2   economic.cond.national  1525 non-null   int64
 3   economic.cond.household 1525 non-null   int64
 4   Blair                   1525 non-null   int64
 5   Hague                   1525 non-null   int64
 6   Europe                  1525 non-null   int64
 7   political.knowledge     1525 non-null   int64
 8   gender                  1525 non-null   object
dtypes: int64(7), object(2)
memory usage: 107.4+ KB
```

From Info () function we infer that, the dataset has 1524 instances with 9 attributes. 7 integer type, and 2 object type (Strings in the column).

Here, apart from variable 'age' all other integer types are considered as category since there values are not in continuous number they are in range, in order to perform univariant & bivariant analysis in future we will be converting the integer type to object type for the variables 'economic.cond.national', 'economic.cond.household', 'Blair', 'Hague', 'Europe', 'political.knowledge'.

Duplication check: Output from duplicated with sum command is –

```
The dataset has 8 duplications.
```

Though there is no customer ID or any unique identifier to determine whether it is true duplication or not, here we are dropping the duplicate value since it is less in number compared to size of the data and this will further avoid bias in analysis.

Duplication check after dropping: Output from duplicated with sum command is –

```
The dataset has 0 duplication.
```

Null value check: Output from isnull with sum command is –

```
vote                    0
age                     0
economic.cond.national  0
economic.cond.household 0
Blair                   0
Hague                   0
Europe                  0
political.knowledge     0
gender                  0
dtype: int64
```

Dataset doesn't contain any null values.

Describe for numerical and categorical columns

Describe method will help us see how data is spread for the numerical values, also we can see the minimum value, mean values, different percentile values and maximum values.

Output from Describe with Transpose command for numeric features is –

	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

Table 2: Descriptive Statistics (Numeric Columns)

Output from Describe with Transpose command for categorical features is –

	count	unique	top	freq
vote	1525	2	Labour	1063
gender	1525	2	female	812

Table 3: Descriptive Statistics (Categorical Columns)

Observation: -

- From the above Descriptive stats table, we can infer that, there is no bad /anomalies present in the data.
- The mean 'age' of the surveyed voters is around 54yrs, the minimum voters age is 24yrs to the maximum age of 93yrs.
- In terms of ‘gender’ variable we infer that, people choose to vote ‘Labour’ Party than Conservative Party. In terms of voting, compared to male gender, female gender polling is slightly high.
- Most of the participants/voters around (75%) gives a score of 3-4 for the current national economic conditions.
- Most of the participants/voters around (75%) gives a score of 3-4 for the current household economic conditions.
- As per voter’s survey, we infer that, around 40% gives a rating of 4 or below for Labour Party Leader 'Blair' & Conservative Party Leader ‘Hague’.
- In Europe variable we infer that max value as 11, our data dictionary indicates an 11-point scale that measures respondents' attitudes toward European integration. High scores represent ‘Eurosceptic’ sentiment.
- Around 67% of the surveyed voters are familiar with viewpoints of the 'Labour' and 'Conservative' parties on European integration and rest of the surveyed voters have no notion or just had a rudimentary understanding of these parties.

Checking the Skewness and Kurtosis:

	Skewness
Hague	0.15
age	0.14
Europe	-0.14
economic.cond.household	-0.15
economic.cond.national	-0.24
political.knowledge	-0.43
Blair	-0.54

	kurtosis
economic.cond.household	-0.21
economic.cond.national	-0.26
age	-0.95
Blair	-1.07
political.knowledge	-1.22
Europe	-1.24
Hague	-1.39

Table 4: Skewness & Kurtosis

As we aware of that except variable 'age' other variables are in range scale and should be considered as categorical. Variable 'age' almost has normal distribution and doesn't have high positive & negative kurtosis.

1.2 Perform Univariate and Bivariate Analysis. Do exploratory data analysis. Check for Outliers.

Null value check: Output from isnull with sum command is –

```
vote                        0
age                         0
economic.cond.national      0
economic.cond.household     0
Blair                       0
Hague                       0
Europe                      0
political.knowledge         0
gender                      0
dtype: int64
```

Dataset doesn't contain any null values.

Data Type check:

```
vote                       object
age                         int64
economic.cond.national      int64
economic.cond.household     int64
Blair                       int64
Hague                       int64
Europe                      int64
political.knowledge         int64
gender                     object
dtype: object
```

As we discussed earlier, apart from variable 'age' all other integer types are considered as category since there values are not a continuous number they are in range scale (Ordinal Values), in order to perform univariant & bivariant analysis we convert the integer type variables of 'economic.cond.national', 'economic.cond.household', 'Blair', 'Hague', 'Europe', 'political.knowledge' to object type.

Data Type check after conversion: -

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1525 entries, 0 to 1524
Data columns (total 9 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   vote                     1525 non-null   object
 1   age                      1525 non-null   int64
 2   economic.cond.national   1525 non-null   object
 3   economic.cond.household  1525 non-null   object
 4   Blair                    1525 non-null   object
 5   Hague                    1525 non-null   object
 6   Europe                   1525 non-null   object
 7   political.knowledge      1525 non-null   object
 8   gender                   1525 non-null   object
dtypes: int64(1), object(8)
memory usage: 107.4+ KB
```

Shape of the dataset: Output from shape command is –

```
The dataset has 1525 rows and 10 columns
```

Univariant Analysis: - Histogram & Boxplot (Numeric Column)

The objective of univariant analysis is to derive the data, define, analyse and summarize the pattern present in it. In a dataset, it explores each variable separately such as Numerical variable and Categorical variable. Some of the patterns that can be easily identified with univariant analysis are Central Tendency (mean, mode and median), Dispersion (range, variance), Quartiles (interquartile range), and Standard deviation. Univariant analysis can be described and visualize with the help of most used plots of Histogram/Distplot and Barplot.

Column - Age

```
Skewness of age: 0.14
Kurtosis of age: -0.94
Outliers of age: 0.0
```

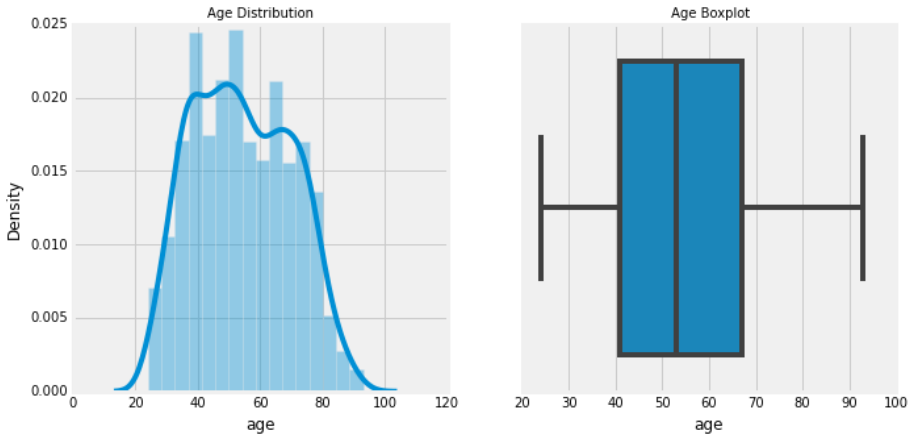


Fig 1: Histplot & Box Plot – Univariant Analysis (Age)

Observation: -

- From the above graph we infer that, that Mean & Median 'age' of the surveyed voters is around 54yrs, there minimum age is around 24yrs and maximum age is around 93yrs.
- The distribution of 'age' is slightly right skewed with skewness value of 0.14.

- The distribution is almost normally distributed.
- The histplot shows most of data are distributed from 25 to 80.
- The box plot of the 'age' variable shows no outliers.

Univariant Analysis: - Pie Chart (Categorical Columns)

A pie chart is a circle that is divided into areas, or slices. Each slice represents the count or percentage of the observations of a level for the variable.

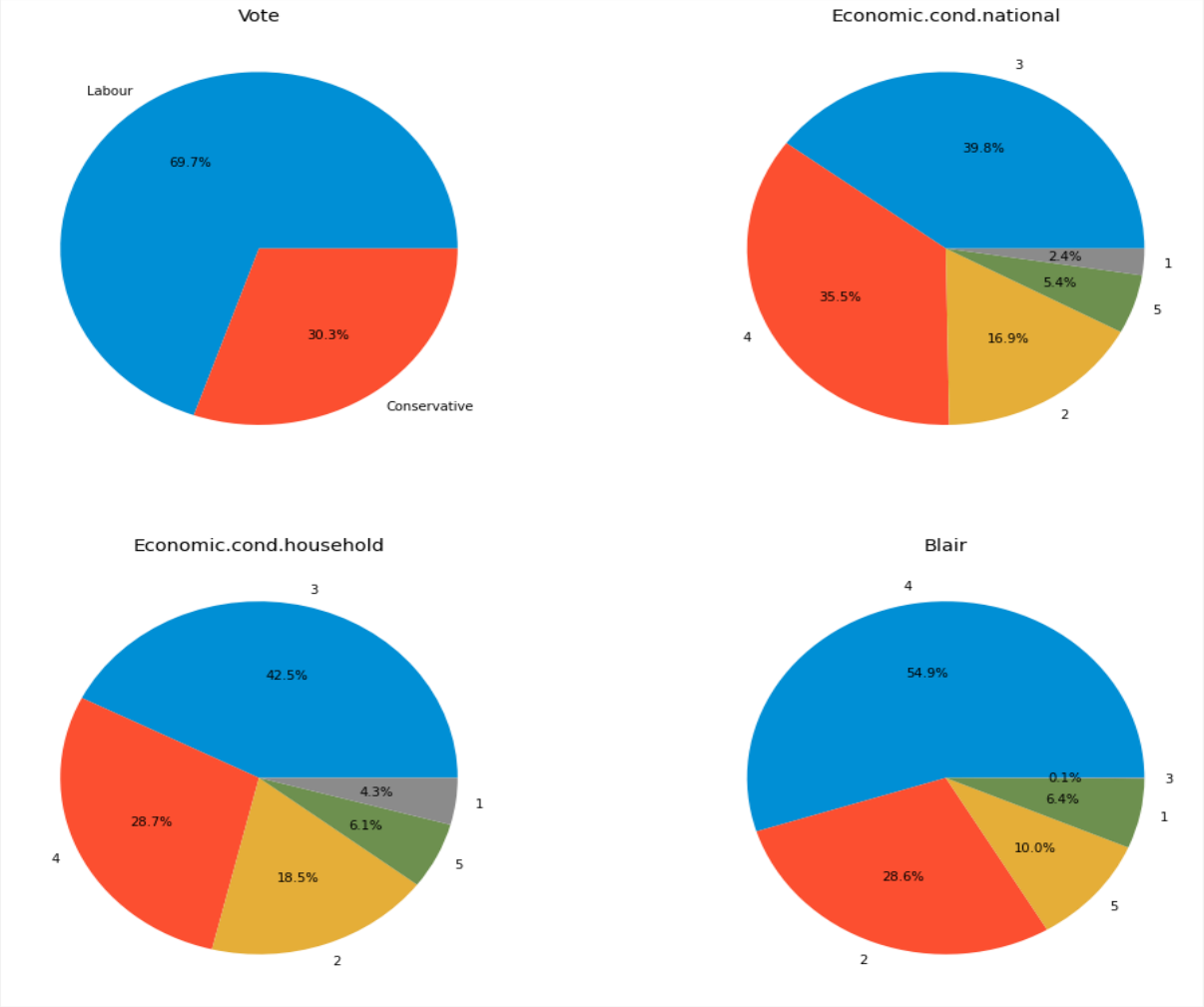


Fig 2: Pie Chart – Univariant Analysis (Categorical columns)

Observation: -

- Vote - The Target variable 'Vote' shows that around 69.7% of the people choose to vote Labour party than Conservative party.
- economic.cond.national (Assessment of current national economic conditions, 1 to 5) - For economic.cond.national , around 75.3% of people opted the rating between 3-4 and only

5.4% of voters opted the high rating of 5 and 19.3% of people opted low rating scale ranges between 1-4.

- economic.cond.household (Assessment of current household economic conditions, 1 to 5) - For economic.cond.household, around 71.2% people opted the rating score as 3-4, 18.5% voters gives the rate score of 2, around 4.3% people opted the rating score of 1 and very few, ie around 6.1% voters opted the high rating score of 5.
- Blair: Assessment of the Labour leader, 1 to 5 – Most of the people/voters around 54.9% gives the rating of 4, 28.6% voters give the rate score of 2, around 10.0% voters give the rate score of 5 and very few opted the rating score of 1 & 3 ie 6.5%.

Univariant Analysis: - Count Plot (Categorical Columns)

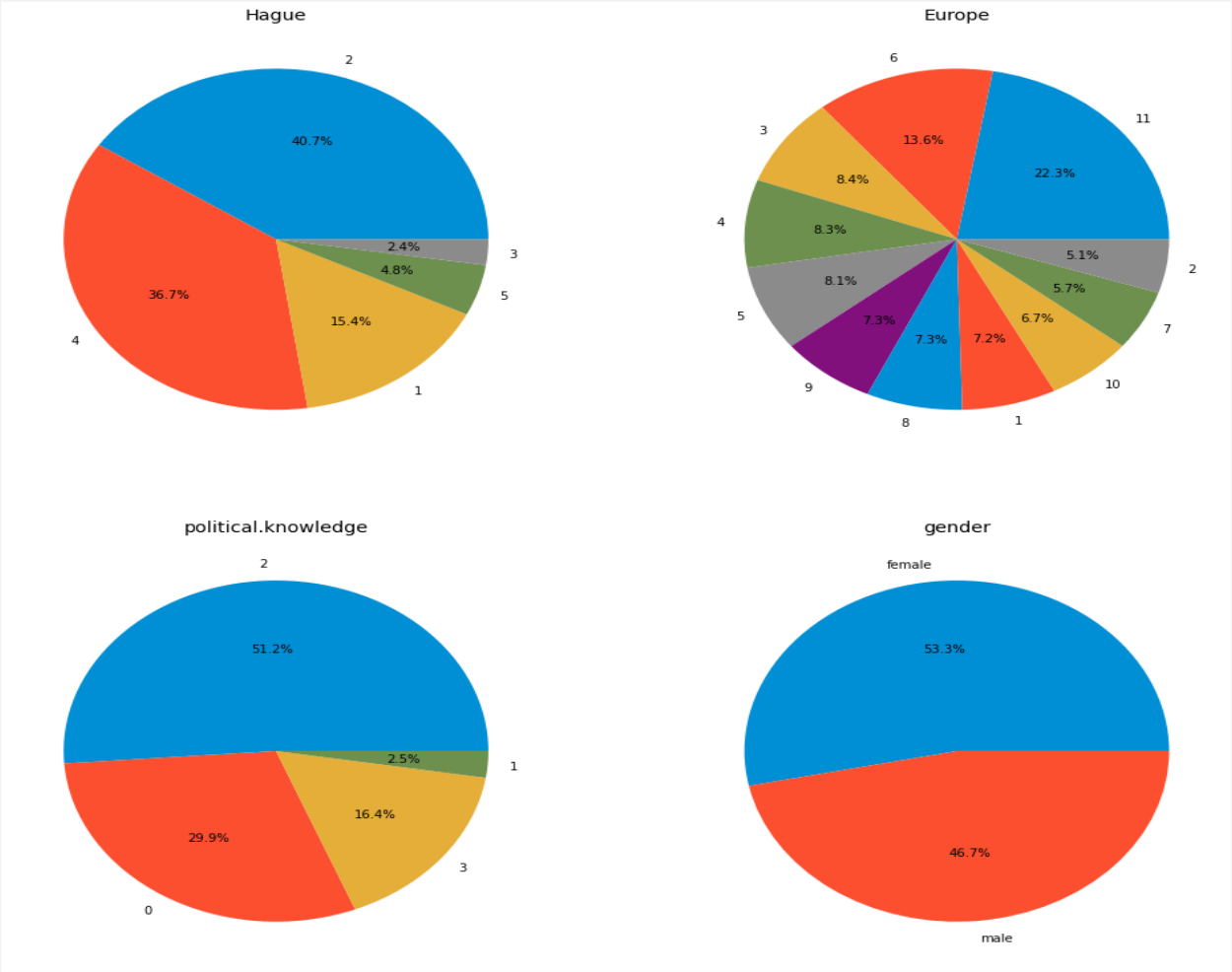


Fig 3: Pie Chart – Univariant Analysis (Categorical columns)

Observation: -

Hague: Assessment of the Conservative leader, 1 to 5 - For conservative party leader 'Hague' around 40.7% of the voters gave a rating of 2, 36.7% of voters gave a rating of 4 and very few voters round 4.8% gave a rating of 5 and 17.8% voters opted the rating scale of 1 & 3.

Europe: an 11-point scale that measures respondents' attitudes toward European integration. High scores represent 'Eurosceptic' sentiment - Though 22.3% of surveyed voters are strongly Eurosceptic and opted the rating scale of 11, the rest of the surveyed voters appear to be evenly spread between low and high Eurosceptic emotions, also we infer that, voters who opted the average rating of 6 is slightly high. For this We can assume that the party that does not favour the European Integration might get more votes from the people.

Political.knowledge: Knowledge of parties' positions on European integration, 0 to 3 - 31.4% of the voters have least political knowledge with the level of 0-1 regarding the party's position on European integration. 16.4% of voters have high political knowledge regarding the party's position on European integration. 51.2% of voters have fair with the level of 2 political knowledge regarding the party's position on European integration.

Gender: female or male - Around 53.3% of the voters are female and 46.7% of the voters are males, there's no significant difference found in gender in terms of polling a vote.

Bivariant Analysis: - For Bi-variant analysis of Target (Vote) Vs continuous variables, we shall use Strip plot.

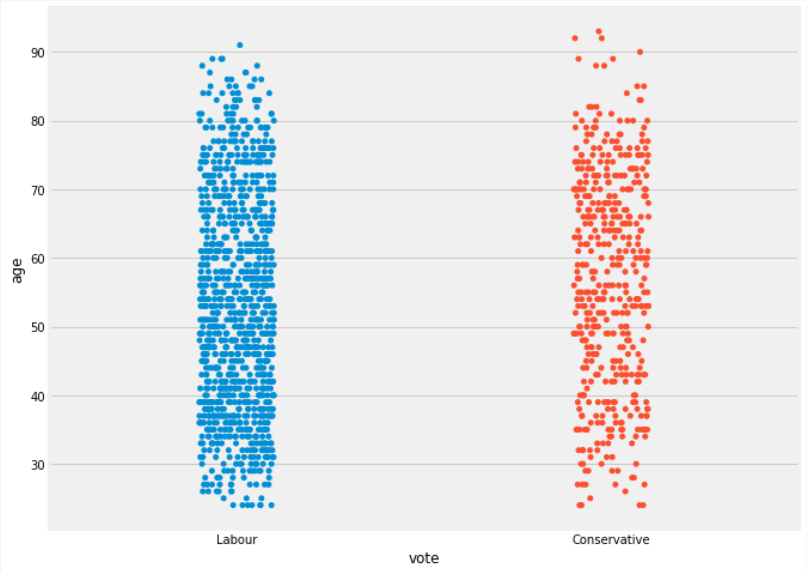


Fig 4: Strip Plot - Bivariant Analysis (Age vs Vote)

Observation: -

From Strip plot we infer that 'Labour' party in terms of age has slightly high density than 'Conservative' party.

Bivariant Analysis: - For Bi-variant analysis of Target (Vote) Vs Categorical variables, we shall use box plot.

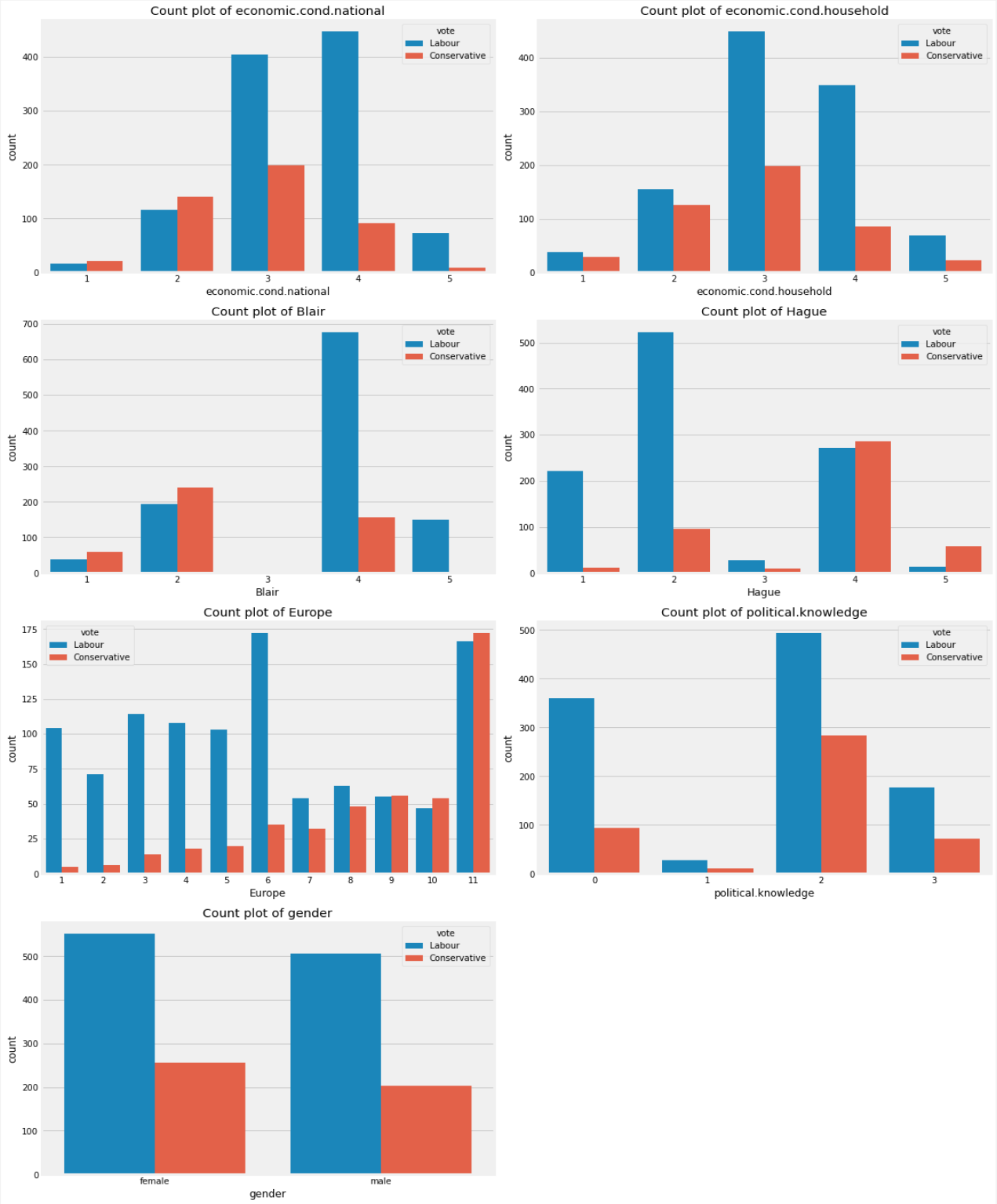


Fig 5: Count Plot - Bivariant Analysis (Categorical columns)

Observation: -

Economic.cond.national vs Vote –

- As per voter survey, most of the people opted the rating score of 3-4 for 'Labour' party than 'Conservative' party.
- There's no significant difference found in both the parties in terms less rating scale range 1-2 & high rating scale of 5 is slightly high for 'Labor Party' than 'Conservative Party'.

Economic.cond.household vs Vote –

- Many voters assessed the rating scale of 3 - 4 of economic conditions and household conditions favouring for 'Labour' party than 'Conservative' party.
- There's no significant difference found in both the parties in terms less rating scale range 1-2 & high rating scale of 5 is slightly high for 'Labor Party' than 'Conservative Party'.

Blair/Hague vs Vote –

- For Labor party leader 'Blair' most of the surveyed voters gave a rating of 4 and for 'Conservative' party leader 'Hague' most of the surveyed voters gave a rating of 2, but some of the voters gave the rating scale of 4.
- One plausible explanation we can assume that, voters who polled for Blair and opted higher rating might have low rated the 'Hague' leader. The rating scale of Blair is around 500, which is more or less equal to the rating scale of 2 for 'Conservative' party leader Hague.

Europe vs Vote –

- We can observe that as respondents with strong negative attitude toward European integration are more likely to vote Conservative party and respondents who are likely to vote Labour party seems to have even distribution.
- As scores towards 'Eurosceptic' sentiment get increases people are likely to vote conservative party.

Political.knowledge vs Vote –

- We can see that, people with knowledge level 2 for both the parties in terms of position on European integration is almost equally distributed.
- People who voted for 'Labour' party has knowledge level 0 about position on European integration than 'Conservation' party.
- Voters has some fair knowledge level 1 & 3 for both the parties in terms of position on European integration.

Gender vs Vote –

- Both the male and female gender polled the vote for 'Labour' party than 'Conservative' party.

Multivariant Analysis: - Pair Plot & Heat Map

Pair Plot: - A pair plot gives us correlation graphs between all numerical variables in the dataset. Thus, from the graphs we can identify the relationships between all numerical variables.

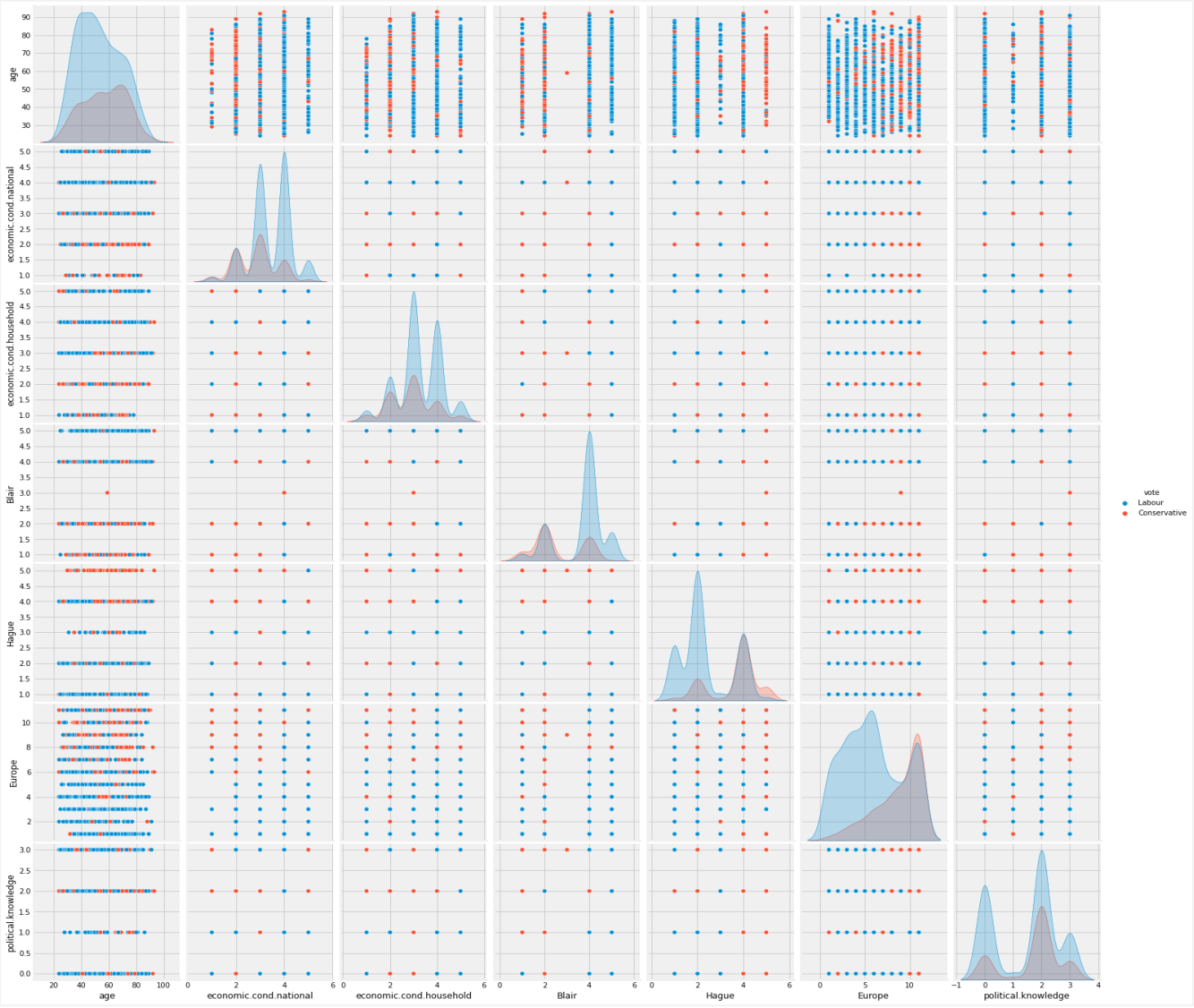


Fig 6: Pair Plot - Multivariant Analysis

Observation: -

- There is no defined pattern in the above graphs depicting the relation between the variables because we are aware of that, expect variable 'Age' all other variables are in ordinal type.
- The above displot diagonals & off diagonals scatter plot, we infer that for all the variables, the distribution & scatter points got overlapped the classes of target variable and variable 'Europe' target classes are slightly well sperated.

Heatmap: -

A heatmap gives us the correlation between numerical variables. If the correlation value is tending to 1, the variables are highly positively correlated whereas if the correlation value is close to 0, the variables are not correlated. Also, if the value is negative, the correlation is negative. That means, higher the value of one variable, the lower is the value of another variable and vice-versa.

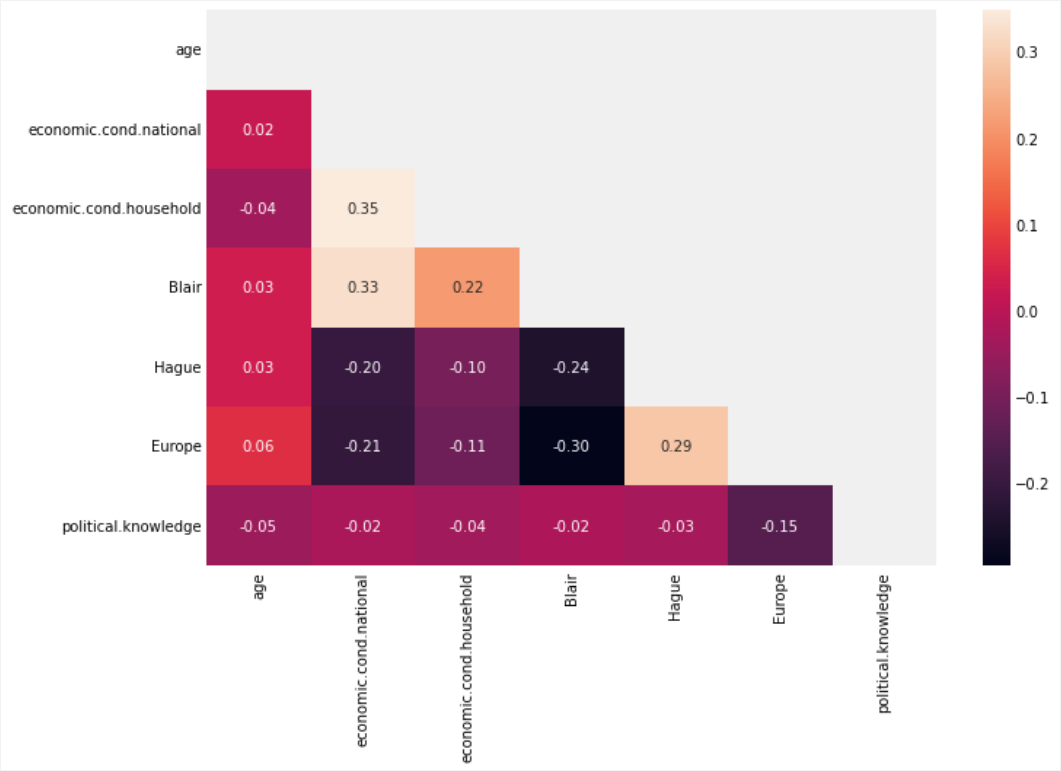


Fig 7: Heatmap - Multivariant Analysis

Observation: -

- Negative Correlation is an indication that mentioned variables move in the opposite direction, so this indicates people who are voting for Blair are not voting for Hague.
- Ratings of household economic condition and national economic condition have some significant positive correlation of 0.35.
- Respondents giving high rating to conservative party have certain extent of correlation with Europe with positive correlation of 0.29. That means voters who are against Europe integration with high score of Eurosceptic' sentiment is more likely to vote Hague.
- Participants giving high rating to national economic condition are supporters of labour party with positive correlation of 0.33.
- There is negative correlation between age and & political knowledge.

There is strong relationship found between the variables, hence there is no problem of multicollinearity among variables in the dataset.

Checking for outliers: -

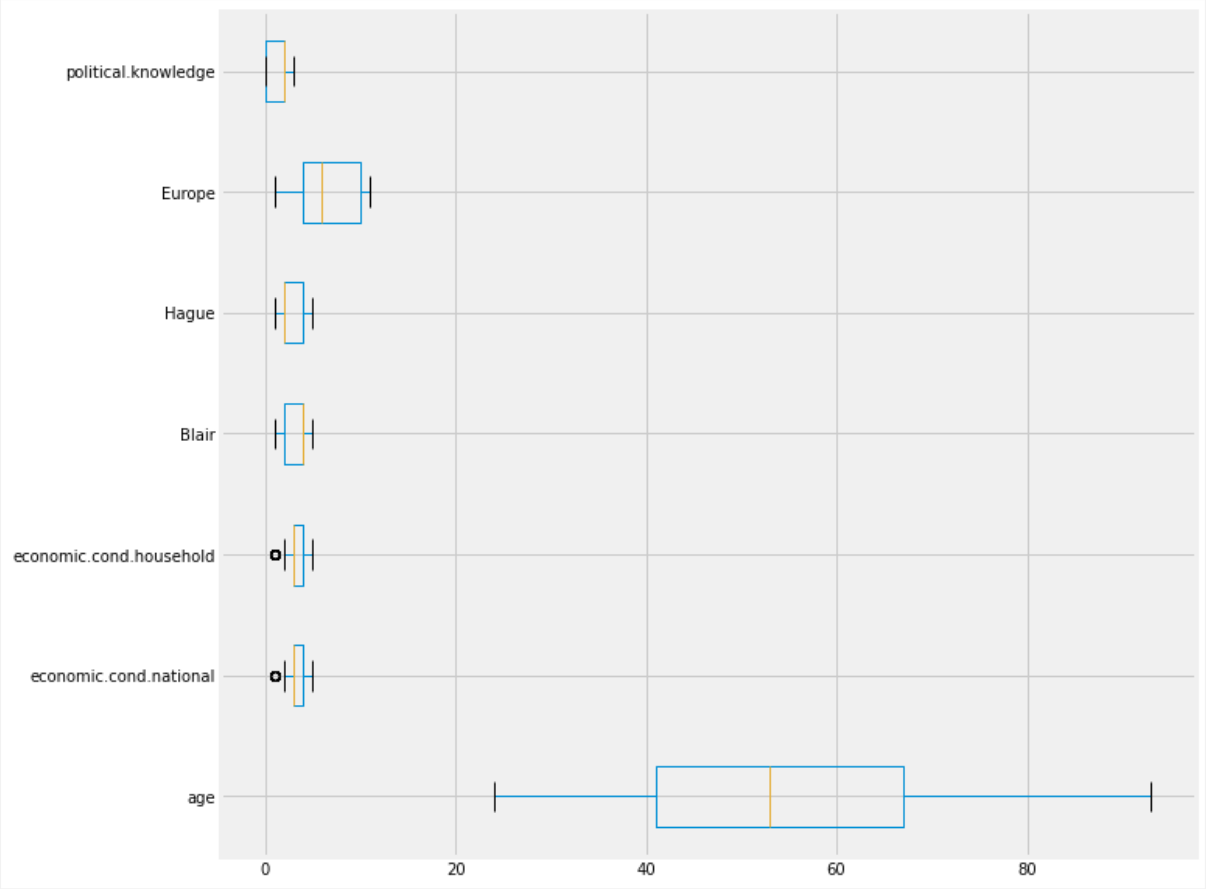


Fig 8: Boxplot (Outlier Check)

Outlier Value: -

	Outliers
economic.cond.household	0.65
economic.cond.national	0.37
Blair	0.00
Hague	0.00
Europe	0.00
political.knowledge	0.00

Table 5: Outlier Value

Observation: -

Outliers are present only in variable economic_condition_household and national, but here we are not treating the outliers since 'Economic Condition Household' and 'Economic condition National' are of ordinal type, i.e., they follow certain order or degree of magnitude and outlier treatment is applicable only for continuous variables, not for ordinal/ categorical variables.

1.3 Encode the data (having string values) for Modelling. Is Scaling necessary here or not? Data Split: Split the data into train and test (70:30)

1. Feature Engineering

Feature engineering is a technique used to encode categorical features into numerical values so that machine learning algorithm can understand. Most popular categorical converting technique is One hot encoding or Label encoding. Here, we use label encoding for categorical values to converted into simple numerical values without losing an information. During Label encoding all categorical features are labelled in a numeric value by alphabetical order.

Below is the output retrieved from Jupyter: -

```
feature: vote
['Labour', 'Conservative']
Categories (2, object): ['Conservative', 'Labour']
[1 0]


feature: gender
['female', 'male']
Categories (2, object): ['female', 'male']
[0 1]
```

2. Checking the head of the dataset after label encoding: -

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

Table 6: Encoded table (categorical columns)

3. Checking the Info of the dataset after label encoding: -

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1517 entries, 0 to 1524
Data columns (total 9 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   vote                    1517 non-null   int8
 1   age                     1517 non-null   int64
 2   economic.cond.national  1517 non-null   int64
 3   economic.cond.household 1517 non-null   int64
 4   Blair                   1517 non-null   int64
 5   Hague                   1517 non-null   int64
 6   Europe                  1517 non-null   int64
 7   political.knowledge     1517 non-null   int64
 8   gender                  1517 non-null   int8
dtypes: int64(7), int8(2)
memory usage: 130.1 KB
```

By checking the head and info of the dataset after label encoding all object types got converted to number.

Is Scaling necessary?

For this dataset we are building logistic Regression, LDA, KNN, Naïve bayes, Bagging and Boosting models.

- Logistic Regression and Linear Discriminant Analysis (LDA) finds its coefficients using the variation between the classes, so scaling is not required.
- Navie bayes algorithm is based on probability not on distance, so it doesn't require feature scaling.
- On the other hand, KNN requires scaling of data, because KNN uses the Euclidean distance between two data points to find nearest neighbours. Distance & Gradient descent algorithms are sensitive to magnitudes. The features with high magnitudes will weigh more than features with low magnitudes. Role of Scaling is mostly important these algorithms.
- Random Forest is a tree-based model and hence feature scaling not required. This algorithm requires partitioning, even if we apply Normalization, the result will be the same.

In a given dataset most of the variables take ordinal values ranging from 0 to 11. However, variable age takes a continuous values ranging from 24 to 93. We thus notice that variable with very high/ low values cannot be compared for analysis and leads to high variance during distance-based model building such as KNN & Gradient Boosting models.

Here we are using scaled data only for KNN & Gradient Boosting. All other models use unscaled dataset while building a model.

4. Scaling the dataset only for KNN and Gradient Boosting Models:

Scaling is only for the continuous variables and ordinal variables. If we are applying Min-Max scaling even binary values won't change, so we are scaling the data using Min-Max technique.

Min-max = (X–Xmin)/(Xmax–Xmin)

Below is the output from Jupyter using MinMaxScaler library in sklearn.preprocessing package:

	vote	age	economic.cond.national	economic.cond.household	Blair	Hague	Europe	political.knowledge	gender
0	1.0	0.275362	0.50	0.50	0.75	0.00	0.1	0.666667	0.0
1	1.0	0.173913	0.75	0.75	0.75	0.75	0.4	0.666667	1.0
2	1.0	0.159420	0.75	0.75	1.00	0.25	0.2	0.666667	1.0
3	1.0	0.000000	0.75	0.25	0.25	0.00	0.3	0.000000	0.0
4	1.0	0.246377	0.25	0.25	0.00	0.00	0.5	0.666667	1.0

Table 7: Scaled Data (Min-Max)

5. Checking the proportion of observations using Target variable 'Vote'.

	Vote (Target Variable)
Conservative Party - Class 0	0.303
Labour Party - Class 1	0.697

By observing an above output retrieved from Jupyter, we can say that there is no drastic variance found in terms of class imbalance, and we have reasonable proportions in both the classes, the dataset is now ready for train and test split and model building.

6. Extracting the target column into separate vectors for training set and test set.

- Here we store the inpendent features in variable X (‘age', 'economic.cond.national', 'economic.cond.household', 'Blair', 'Hague', 'Europe', 'political.knowledge', 'gender') and dependent feature/Target feature in Y variable(‘vote’).
- Train data will hold an independent variable whereas test data will hold a dependent variable of the dataset.

7. Splitting data into training and test set.

- Inorder to perform this step, from the package sklearn.model_selection we imported train_test_split.
- Now we split the data into 70 -30 ratio, where the train data hold 70% of the data and test data holds 30% of the data. The random state mentioned here is 1.

Below output is retrieved from Jupter using shape command -

```
X_train (1061, 8)
X_test (456, 8)
train_labels (1061,)
test_labels (456,)
```

- Train dataset has 1061 records i.e., 70% of the total dataset.
- Test dataset contains 456 records i.e., 30% of the total dataset.

Now our data set is ready for building a model to predict which party a voter will vote for on the basis of the given information and to create an exit poll that will help in predicting overall win and seats covered by a particular party.

In order to do our analysis, we are expected to build model using Logistic Regression, LDA, KNN Model and Naïve Bayes Model. As mentioned, scaling might be necessary only for KNN & Gradient Boosting model and not be necessary for other three models.

1.4 Apply Logistic Regression and LDA (linear discriminant analysis)

Logistic Regression: -

- Linear Discriminant Analysis is a dimensionality reduction technique that is commonly used for supervised classification problems. It is used for modelling differences in groups i.e., separating two or more classes. It is used to project the features in higher dimension space into a lower dimension space.
- LDA works when the measurements made on independent variables for each observation are continuous quantities. When dealing with categorical independent variables, the equivalent technique is discriminant correspondence analysis.
- When we fit our training data into Linear Discriminant model. By default, LDA takes a custom cut-off probability as 0.5. At first, we'll create our LDA model with cut-off probability as 0.5 and validate the performance, then we'll check the performs with multiple cut-off probabilities and decide which one performs best.

1. Building a Logistic Regression Model

Import the necessary library of LogisticRegression from sklearn.linear_model.

In this step we fit the train data and labels in Logistic Regression model.

2. **Default Estimator**

Below output is obtained from Jupyter that results the default estimator for building Logistic Regression model with random_state=1.

LogisticRegression(tol=0.0001, random_state=1, solver='lbfgs', max_iter=100, verbose=0)

There are various input parameters applicable: -

- 'max_iter', 'penalty',solver', 'tol' which will helps us to find best grid for prediction of the better model
- max_iter is an integer (100 by default) that defines the maximum number of iterations by the solver during model fitting.
- solver is a string ('liblinear' by default) that decides what solver to use for fitting the model. Other options are 'newton-cg', 'lbfgs', 'sag', and 'saga'.
- penalty is a string ('l2' by default) that decides whether there is regularization and which approach to use. Other options are 'l1', 'elasticnet', and 'none'.

The reason we chose the default parameter is because, here we first validate how the model perform with default parameter prior to Hyperparameter Tuning.

3. **Prediction of training and testing dataset using predict command in train and test data.**
4. **Getting the Predicted Classes and Probs.**

Below is the output retrieved in Jupyter using proba command on train & test data: -

Train Data

	0	1
0	0.929638	0.070362
1	0.095496	0.904504
2	0.294741	0.705259
3	0.110759	0.889241
4	0.015879	0.984121

Test Data

	0	1
0	0.429655	0.570345
1	0.144201	0.855799
2	0.005882	0.994118
3	0.846562	0.153438
4	0.058577	0.941423

5. **Model Performance – Accuracy Score**

Logistic Regression Model					
Sl.No	Index	Train Data		Test Data	
1	TN	199		110	
2	TP	688		266	
3	FN	66		37	
4	FP	108		43	
5	Accuracy	0.84		0.82	
6	AUC Score	0.89		0.87	
		Conservative	Labour	Conservative	Labour
7	Precision	0.75	0.86	0.75	0.86
8	Recall	0.65	0.91	0.72	0.88
9	F1 Score	0.70	0.89	0.74	0.87

Table 8: Model performance Metrices - Logistic Regression

Observation: -

- Cases voters polled, there are 37 instances where model predicted as not opted.
- Cases voters not polled, but model predicted them to be polled are 43.
- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Overall, the metrics are good fit.
- Further the model improved using feature engineering, hyper parameter tuning (including combination of various parameters).

6. Checking the Coefficients

```
The coefficient for age is -0.012429719238147176
The coefficient for economic.cond.national is 0.6660208499315987
The coefficient for economic.cond.household is 0.09273129807765693
The coefficient for Blair is 0.6199222763887081
The coefficient for Hague is -0.7994359051594999
The coefficient for Europe is -0.20263748326607367
The coefficient for political.knowledge is -0.30364698893202025
The coefficient for gender is 0.20684228575263883
```

The sign of a regression coefficient tells you whether there is a positive or negative correlation between each independent variable the dependent variable. A positive coefficient indicates that as the value of the independent variable increases, the mean of the dependent variable also tends to increases. A negative coefficient suggests that as the independent variable increases, the dependent variable tends to decrease.

- The features economic condition national and ‘Labour’ leader ‘Blair’ contributes largely in the dataset.
- economic.cond.national have more positive coeffiecient . A positive coefficient indicates that as the value of the independent variable increases, the mean of the dependent variable also tends to increase the vote.

LDA Model (linear discriminant analysis)

- Linear Discriminant Analysis is a dimensionality reduction technique that is commonly used for supervised classification problems. It is used for modelling differences in groups i.e., separating two or more classes. It is used to project the features in higher dimension space into a lower dimension space.
- LDA works when the measurements made on independent variables for each observation are continuous quantities. When dealing with categorical independent variables, the equivalent technique is discriminant correspondence analysis.
- When we fit our training data into Linear Discriminant model. By default, LDA takes a custom cut-off probability as 0.5. At first, we'll create our LDA model with cut-off probability as 0.5 and validate the performance, then we'll check the performs with multiple cut-off probabilities and decide which one performs best.

1. **Building a Linear Discriminant Analysis Model**

Import the necessary library of LinearDiscriminantAnalysis from sklearn.discriminant_analysis -

In this step we fit the train data and labels in Linear Discriminant Analysis model.

2. **Default Estimator**

Below output is obtained from Jupyter that results the default estimator for building Linear Discriminant Analysis with random_state=1.

LinearDiscriminantAnalysis (tol=0.0001, random_state=1, solver='svd')

There are various input parameters applicable: -

- 'max_iter', 'penalty',solver', 'tol ' which will helps us to find best grid for prediction of the better model
- max_iter is an integer (100 by default) that defines the maximum number of iteration s by the solver during model fitting.
- solver is a string ('liblinear' by default) that decides what solver to use for fitting the model. Other options are 'newton-cg', 'lbfgs', 'sag', and 'saga'. ➢ here 'solver':['svd', 'lsqr', 'eigen'] are used with others parameters has default
- 'svd': Singular value decomposition (default). Does not compute the covariance matri x, therefore this solver is recommended for data with many features.
- 'lsqr': Least squares solution. Can be combined with shrinkage or custom covariance estimator.
- 'eigen': Eigenvalue decomposition. Can be combined with shrinkage or custom covariance estimator.

The reason we chose the default parameter is because, here we first validate how the model perform with default parameter prior to Hyperparameter Tuning.

3. **Prediction of training and testing dataset using predict command in train and test data.**

4. **Getting the Predicted Classes and Probs.**

Below is the output retrieved in Jupyter using proba command on train & test data: -

Train Data

	0	1
0	0.949216	0.050784
1	0.078241	0.921759
2	0.307389	0.692611
3	0.078963	0.921037
4	0.012161	0.987839

Test Data

	0	1
0	0.462093	0.537907
1	0.133955	0.866045
2	0.006414	0.993586
3	0.861210	0.138790
4	0.056545	0.943455

5. Model Performance Metrics

Linear Discriminative Analysis Model					
Sl.No	Index	Train Data		Test Data	
1	TN	200		111	
2	TP	685		269	
3	FN	69		34	
4	FP	107		42	
5	Accuracy	0.83		0.83	
6	AUC Score	0.89		0.89	
		Conservative	Labour	Conservative	Labour
7	Precision	0.74	0.86	0.77	0.86
8	Recall	0.65	0.91	0.73	0.89
9	F1 Score	0.69	0.89	0.74	0.88

Table 9: Model performance Metrices – LDA

Observation: -

- There's no drastic difference found between Logistic Regression and LDA model.
- Cases voters polled, there are 34 instances where model predicted as not opted.
- Cases voters not polled, but model predicted them to be polled are 42.
- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Overall, the metrics are good fit.
- Further the model improved using feature engineering, hyper parameter tuning (including combination of various parameters).

1.5 Apply KNN Model and Naïve Bayes Model. Interpret the results.

K-Nearest Neighbors Model

- KNN algorithm also known as K-Nearest Neighbors Algorithm is used to solve the both problems of classification as well as regression. This working principle of algorithm is mainly based on feature similarity in both classification and regression problem. KNN classifier is different from other probabilistic classifiers where the model comprises a learning step of computing probabilities from a training sample and use them for future prediction of a test sample. In probability-based model once the model is trained the training sample could be thrown away and classification is done using the computed probabilities.

- In KNN there is no learning step involved instead the dataset is stored in memory and is used to classify the test query on the fly. KNN is also known as Lazy learner as it does not create a model using training set in advance. It is one of the simplest methods of classification. In KNN, the term `k' is a parameter which refers to the number of nearest neighbours. The classification procedure for a query point q works in two steps as:
  - Find the K neighbours in the dataset which are closet to q based on the similarity measure.
  - Use these K neighbours to determine the class of q using majority voting.

- ❖ Distance Measure - KNN classifier needs to compute similarities or distances of test query from each sample point in training dataset. Several methods are used to compute the distances and the choice completely depends of the types of features in the dataset. The popular distance measurements are Euclidean Distance.

1. **Building K-Nearest Neighbors Model**

Import the necessary library of KNeighborsClassifier from sklearn.neighbors -

In this step we fit the train data and labels in KNeighborsClassifier.

2. **Default Estimator**

Below output is obtained from Jupyter that results the default estimator for building KNNmodel.

KNeighborsClassifier (n_neighbors=5, weights='uniform', algorithm='auto', algorithm='auto', leaf_size=30, p=2, metric='minkowski', metric_params=None, n_jobs=None)

The reason we chose the default parameter is because, here we first validate how the model perform with default parameter prior to Hyperparameter Tuning.

3. Prediction of training and testing dataset using predict command in train and test data.
4. Getting the Predicted Classes and Probs.

Below is the output retrieved in Jupyter using proba command on train & test data: -

Train Data

	0	1
0	0.8	0.2
1	0.0	1.0
2	0.4	0.6
3	0.2	0.8
4	0.0	1.0

Test Data

	0	1
0	0.6	0.4
1	0.4	0.6
2	0.2	0.8
3	0.4	0.6
4	0.0	1.0

5. **Model Performance Metrics**

K-Nearest Neighbors Model					
Sl.No	Index	Train Data		Test Data	
1	TN	222		107	
2	TP	687		267	
3	FN	67		36	
4	FP	85		46	
5	Accuracy	0.86		0.82	
6	AUC Score	0.93		0.87	
		Conservative	Labour	Conservative	Labour
7	Precision	0.77	0.89	0.75	0.85
8	Recall	0.72	0.91	0.70	0.88
9	F1 Score	0.74	0.90	0.72	0.87

Table 10: Model performance Metrices – KNN

Observation: -

- There's no drastic difference found between Logistic Regression, LDA model & KNN Model.
- Cases voters polled, there are 36 instances where model predicted as not polled.
- Cases voters not polled, but model predicted them to be polled are 46.
- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Overall, the metrics are good fit.
- Further the model can be improved using feature engineering, hyper parameter tuning (including combination of various parameters).

Naive Bayes Model

- Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e., every pair of features being classified is independent of each other.
- The algorithm while calculating likelihoods of numerical features it assumes the feature to be normally distributed and then we calculate probability using mean and variance of that feature only and also it assumes that all the predictors are independent to each other. Scale doesn't matter. Performing a feature scaling in this algorithm may not have much effect.
- In a supervised learning situation, Naive Bayes Classifiers are trained very efficiently. Naive Bayed classifiers need a small training data to estimate the parameters needed for classification. Naive Bayes Classifiers have simple design and implementation and they can apply to many real-life situations. Gaussian Naive Bayes is a variant of Naive Bayes that follows Gaussian normal distribution and supports continuous data.

1. **Building Naive Bayes Model**

Import the necessary library of GaussianNB from sklearn.naive_bayes

In this step we fit the train data and labels in GaussianNB.

2. **Default Estimator**

Below output is obtained from Jupyter that results the default estimator for building KNNmodel.

GaussianNB (priors=None, var_smoothing=1e-09)

The reason we chose the default parameter is because, here we first validate how the model perform with default parameter prior to Hyperparameter Tuning.

3. **Prediction of training and testing dataset using predict command in train and test data.**

4. **Getting the Predicted Classes and Probs.**

Below is the output retrieved in Jupyter using proba command on train & test data: -

Train Data

	0	1
0	0.984678	0.015322
1	0.065437	0.934563
2	0.271735	0.728265
3	0.080026	0.919974
4	0.007648	0.992352

Test Data

	0	1
0	0.536792	0.463208
1	0.120285	0.879715
2	0.000332	0.999668
3	0.945240	0.054760
4	0.039267	0.960733

5. Model Performance Metrics

Naive Bayes Model					
Sl.No	Index	Train Data		Test Data	
1	TN	211		112	
2	TP	675		263	
3	FN	79		40	
4	FP	96		41	
5	Accuracy	0.84		0.82	
6	AUC Score	0.89		0.88	
		Conservative	Labour	Conservative	Labour
7	Precision	0.73	0.88	0.74	0.87
8	Recall	0.69	0.90	0.73	0.87
9	F1 Score	0.71	0.89	0.73	0.87

Table 11: Model performance Metrices – Naïve Bayes

Observation: -

- There's no drastic difference found between Logistic Regression, LDA model, KNN Model & Naïve Bayes Model, but LDA & KNN Model performs slightly better than Naïve Bayes Model.
- Cases voters polled, there are 40 instances where model predicted as not polled.
- Cases voters not polled, but model predicted them to be polled are 41.
- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Overall, the metrics are good fit.
- Further the model can be improved using feature engineering, hyper parameter tuning (including combination of various parameters or technique of SMOTE).

1.6 Model Tuning, Bagging (Random Forest should be applied for Bagging), and boosting.

Model Tuning 1 - Logistic Regression Model

Initially, we fitted the train data and labels in Logistic Regression model and validated the performance. Now based on the above model performance we tunning them using Grid search. Here we will choose the best parameters to re-built the model and calculate the performance with the help of Classification report of accuracy, recall, precision and F1 score in both train and test data.

Grid Search: Grid search divides the hyperparameter domain into distinct grids. Then, using cross-validation, it attempts every possible combination of values in this grid, computing some performance measures. The ideal combination of values for the hyperparameters is the point on the grid that maximizes the average value in cross-validation. Grid search is a comprehensive technique that considers all possible combinations in order to locate the best point in the domain.

Import the necessary library of LogisticRegression from sklearn.linear_model package and import GridSearchCV from the package sklearn.model_selection.

1. **Hyperparameter Tuning**

```
param_grid = {
'solver':['newton-cg','liblinear','lbfgs'],
'max_iter':[10000,15000],
'penalty':['none','l1','l2'],
'verbose':[True],
'n_jobs':[2],
'tol':[0.001,0.00000001],
}
```

- Penalized logistic regression imposes a penalty to the logistic model for having too many variables. This results in shrinking the coefficients of the less contribute variables toward zero. This is also known as regularization. In our grid search, we take 'L2' and 'none' as our arguments and check which is preferred by grid search.
- Solver is a process that runs for the optimization of the weights in the model. The solver uses a Coordinate Descent (CD) algorithm that solves optimization problems by successively performing approximate minimization along coordinate directions or coordinate hyperplanes. Different solvers take a different approach to get the best fit model. In our case, we have taken 'sag', 'lbfgs', 'liblinear' and 'newton-cg' as our arguments. We will check which is preferred by grid search.
- Tol is the tolerance of optimization. When the training loss is not improved by at least the given tol on consecutive iterations, convergence is considered to be reached and the training stops. We will be checking for tolerance of 0.0001 and 0.00001.
- The logistic regression uses an iterative maximum likelihood algorithm to fit the data. There are no set criteria for maximum iterations. The solver will run the model till it reaches convergence or till the max iterations, you have provided. In this case, we have given 10000 and 15000 as inputs. We will see which fits better.
- Here we take cross-validation as 3 and scoring as F1 for our grid search.

2. Best Estimator

Below output is obtained from Jupyter that results the best estimator for building our decision tree and that is obtained using a grid search cv function.

> LogisticRegression(max_iter=10000, n_jobs=2, penalty='l1', random_state=1, solver='liblinear', tol=1e-08, verbose=True)

3. Checking the Coefficients

```
The coefficient for age is -0.012429719238147176
The coefficient for economic.cond.national is 0.6660208499315987
The coefficient for economic.cond.household is 0.09273129807765693
The coefficient for Blair is 0.6199222763887081
The coefficient for Hague is -0.7994359051594999
The coefficient for Europe is -0.20263748326607367
The coefficient for political.knowledge is -0.30364698893202025
The coefficient for gender is 0.20684228575263883
```

The features economic condition national and 'Labour' leader 'Blair' contributes largely in the dataset.

4. Model performance for tuned Logistic Regression Model: -

a. Confusion Matrix

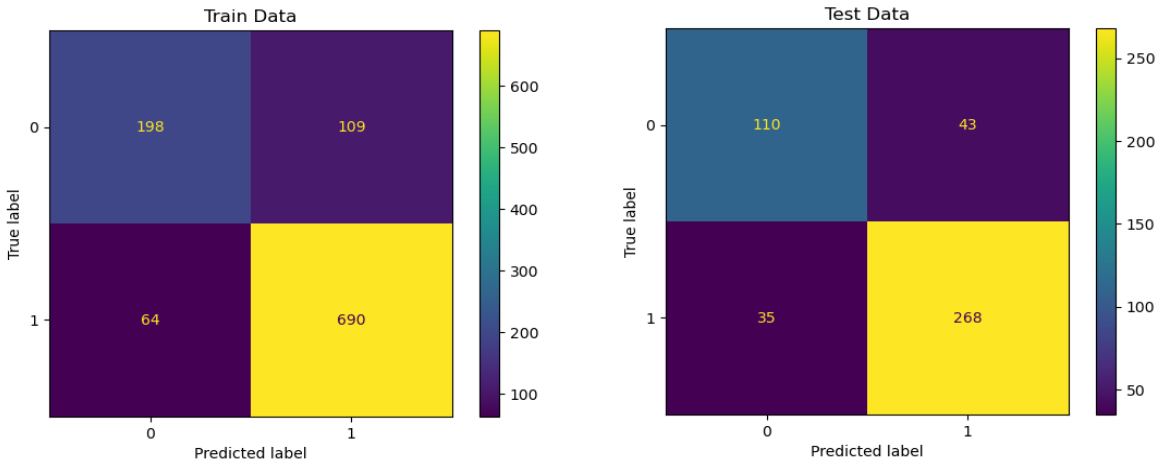


Fig 9: Confusion Matrix - Tunned Logistic Regression

b. Classification Report

Train Report

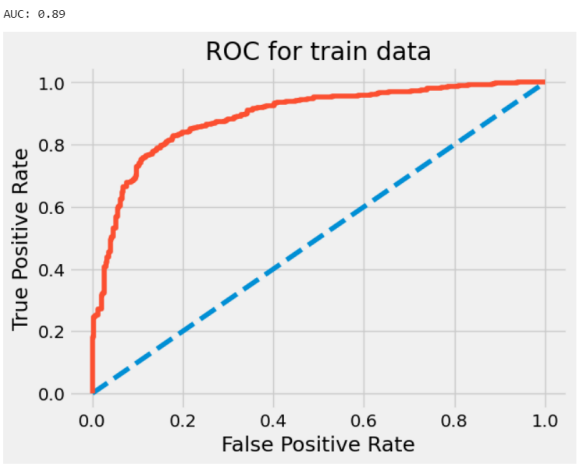
	precision	recall	f1-score	support
0	0.76	0.64	0.70	307
1	0.86	0.92	0.89	754
accuracy			0.84	1061
macro avg	0.81	0.78	0.79	1061
weighted avg	0.83	0.84	0.83	1061

Test Report

	precision	recall	f1-score	support
0	0.76	0.72	0.74	153
1	0.86	0.88	0.87	303
accuracy			0.83	456
macro avg	0.81	0.80	0.81	456
weighted avg	0.83	0.83	0.83	456

Table 12: Classification Report - Tunned Logistic Regression

c. ROC Curve and AUC score



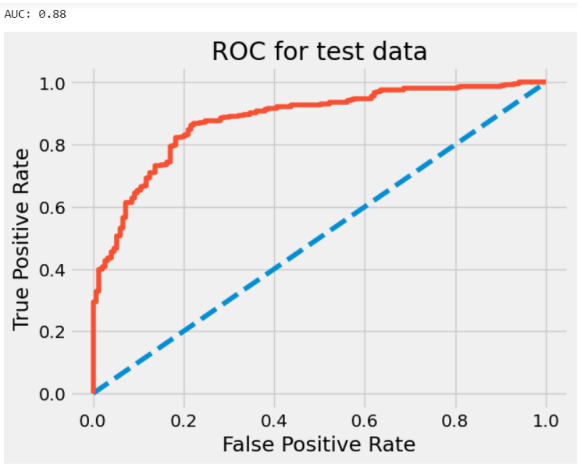


Fig 10: ROC Curve - Tunned Logistic Regression

d. Train & Test report comparison table between Default & Tunned LR Model

Logistic Regression Model					
Sl.No	Index	Train Data		Test Data	
1	TN	199		110	
2	TP	688		266	
3	FN	66		37	
4	FP	108		43	
5	Accuracy	0.84		0.82	
6	AUC Score	0.89		0.87	
		Conservative	Labour	Conservative	Labour
7	Precision	0.75	0.86	0.75	0.86
8	Recall	0.65	0.91	0.72	0.88
9	F1 Score	0.7	0.89	0.73	0.87

Tunned – Logistic Regression Model					
Sl.No	Index	Train Data		Test Data	
1	TN	198		110	
2	TP	690		268	
3	FN	64		35	
4	FP	109		43	
5	Accuracy	0.84		0.83	
6	AUC Score	0.89		0.88	
		Conservative	Labour	Conservative	Labour
7	Precision	0.76	0.86	0.76	0.86
8	Recall	0.64	0.92	0.72	0.88
9	F1 Score	0.7	0.89	0.74	0.87

Table 13: Model performance Metrices - Default Logistic Regression & Tunned Logistic Regression

Observation: -

- There's no drastic difference found even after applying grid search with hyper parameters is almost similar to normal Logistic regression model. There is slight increase in performance in the conservative class, Accuracy & AUC score.
- Cases voters polled, there are 35 instances where model predicted as not polled.
- Cases voters not polled, but model predicted them to be polled are 43.
- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Overall, the metrics are good fit.

Model Tuning 2 - Linear Discriminant Analysis Model

Initially, we fitted the train data and labels in Linear Discriminant Analysis Model and validated the performance. Now based on the above model performance we tunning them using Grid search. Here we will choose the best parameters to re-built the model and calculate the performance with the help of Classification report of accuracy, recall, precision and F1 score in both train and test data.

Grid Search: Grid search divides the hyperparameter domain into distinct grids. Then, using cross-validation, it attempts every possible combination of values in this grid, computing some performance measures. The ideal combination of values for the hyperparameters is the point on the grid that maximizes the average value in cross-validation. Grid search is a comprehensive technique that considers all possible combinations in order to locate the best point in the domain.

Model performance for tuned Logistic Regression Model: -

1. **Importing Packages**

Import the necessary library of LinearDiscriminantAnalysis Model:from sklearn.linear_model package and import GridSearchCV from the package sklearn.model_selection.

2. **Using custom probability cut-off technique for tunning LDA model**

Initial LDA model was build based on default custom cut-off probability (i.e., 0.5). Here we'll test our model with several cut-off probabilities and choose the one that gives best result. To do so, we first start from probability 0.1 and work our way up to 0.9 with an interval level of 1. By checking each of the probabilities recall, F1 score & Precision values we can determine which cut-off probability gives the best recall, F1 score & Precision score and rebuilt the model with best cut-off probability.

	Recall	precision	Accuracy	F1 Score
0.1	0.99	0.74	0.75	0.85
0.2	0.97	0.77	0.77	0.86
0.3	0.96	0.80	0.80	0.87
0.4	0.94	0.83	0.83	0.89
0.5	0.92	0.86	0.84	0.89
0.6	0.87	0.89	0.83	0.88
0.7	0.83	0.92	0.82	0.87
0.8	0.73	0.95	0.78	0.82
0.9	0.56	0.96	0.67	0.70

Table 14: Probability cut-off range table (0.1 - 0.9)

From the above table we can see that the 'Recall' is decreasing from 0.1 to 0.9 and F1 score is increasing till 0.5 and gets drop after that, Precision is increasing from 0.4 to 0.9. The cut off probability 0.4 provides the optimal balance of recall, precision and F1 score. As per the result we'll rebuild the model with probability cut off as 0.4 and check the performances of metrics.

1. Model performance for tuned Linear Discriminative Analysis Model: -

a. Confusion Matrix

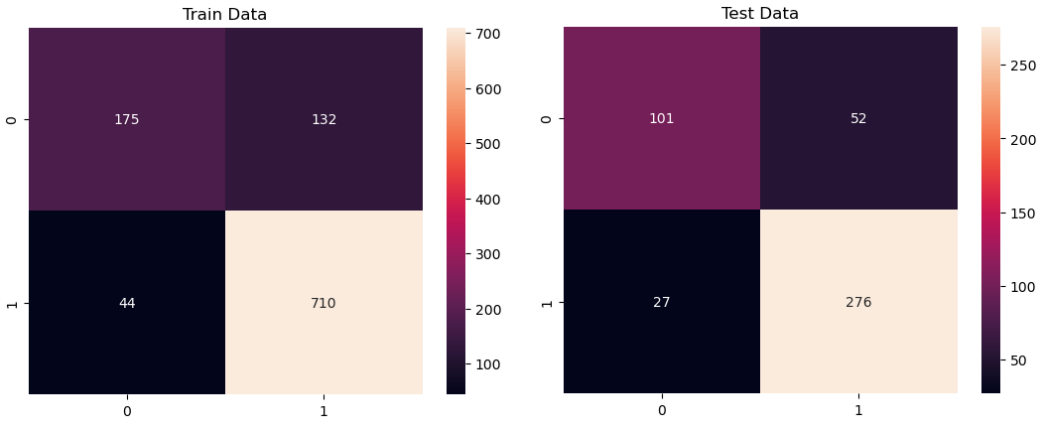


Fig 11: Confusion Matrix - Tunned LDA

b. Classification Report

Training Data report :

	precision	recall	f1-score	support
0	0.80	0.57	0.67	307
1	0.84	0.94	0.89	754
accuracy			0.83	1061
macro avg	0.82	0.76	0.78	1061
weighted avg	0.83	0.83	0.82	1061

Testing Data report:

	precision	recall	f1-score	support
0	0.79	0.66	0.72	153
1	0.84	0.91	0.87	303
accuracy			0.83	456
macro avg	0.82	0.79	0.80	456
weighted avg	0.82	0.83	0.82	456

Table 15: Classification Report - Tunned Linear Discriminative Analysis

c. ROC Curve and AUC score

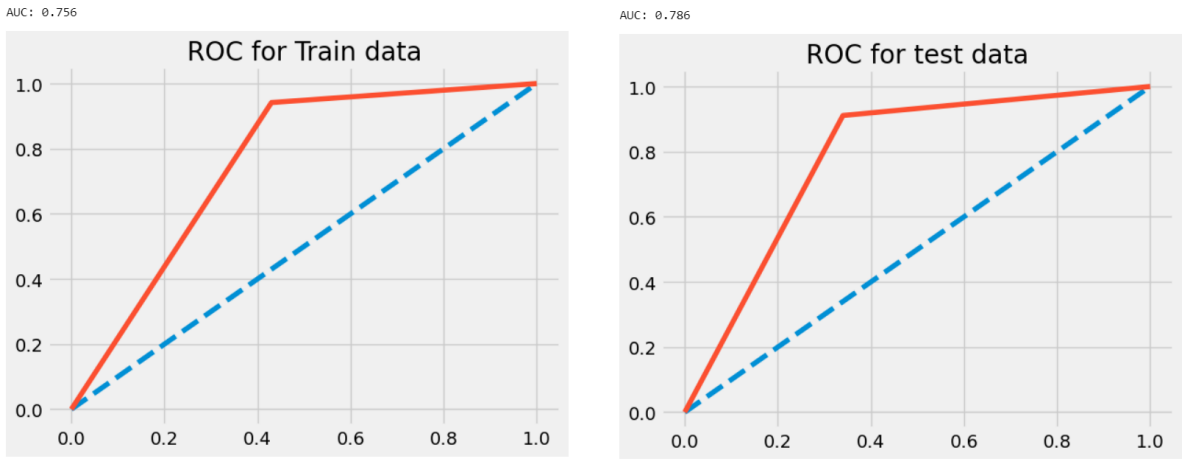


Fig 12: ROC Curve - Tunned LDA

d. Train and Test report comparison table between Default & Tuned Model

Linear Discriminative Analysis Model (Cut_off 0.5)					
Sl.No	Index	Train Data		Test Data	
1	TN	200		111	
2	TP	685		269	
3	FN	69		34	
4	FP	107		42	
5	Accuracy	0.83		0.83	
6	AUC Score	0.89		0.89	
		Conservative	Labour	Conservative	Labour
7	Precision	0.74	0.86	0.77	0.86
8	Recall	0.65	0.91	0.73	0.89
9	F1 Score	0.69	0.89	0.74	0.88

Tunned – Linear Discriminate Analysis Model (Cut_off 0.4)					
Sl.No	Index	Train Data		Test Data	
1	TN	175		101	
2	TP	710		276	
3	FN	44		27	
4	FP	132		52	
5	Accuracy	0.83		0.83	
6	AUC Score	0.76		0.79	
		Conservative	Labour	Conservative	Labour
7	Precision	0.80	0.84	0.79	0.84
8	Recall	0.57	0.94	0.66	0.91
9	F1 Score	0.67	0.89	0.72	0.87

Table 16: Model performance Metrices - Default LDA & Tunned LDA

Observation: -

- There's no drastic difference found even after applying grid search with hyper parameters is almost similar to normal LDA model. There is slight increase in performance in the conservative class in terms of precision.
- There is slight increase in performance in the Labour class in terms of Recall.
- In tunned model cases voters polled are 27 instances where model predicted as not polled. We observer that FN decreased from 34 to 27.
- In tunned model Cases voters not polled, but model predicted them to be polled are 52, We observer that FP increased from 42 to 52.
- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Overall, the metrics are good fit.

Model Tuning 3 - K-Nearest Neighbors Model

Initially, we fitted the train data and labels in K-Nearest Neighbors Model and validated the performance with default value K=5. Now based on the above model performance we are tuning them using Grid search. Here we will choose the best parameters to re-built the model and calculate the performance with the help of Classification report of accuracy, recall, precision and F1 score in both train and test data.

1. Importing Packages

Import the necessary library of KNeighborsClassifier Model from sklearn.neighbors package.

Running a loop for K= 1 to 19 odd numbers and find MSE

Run the KNN with no of neighbours to be 1,3,5..19 and Find the optimal number of neighbours from K=1,3,5,7....19* using the Mis classification error

Hint: Misclassification error (MCE) = 1 - Test accuracy score. Calculated MCE for each model with neighbours = 1,3,5...19 and find the model with lowest MCE.

MCE = [1 - x for x in ac_scores]

```
[0.24342105263157898,
 0.20175438596491224,
 0.17982456140350878,
 0.1600877192982456,
 0.1578947368421053,
 0.16666666666666663,
 0.16228070175438591,
 0.1578947368421053,
 0.16228070175438591,
 0.16447368421052633]
```

2. Misclassification error vs k Plot

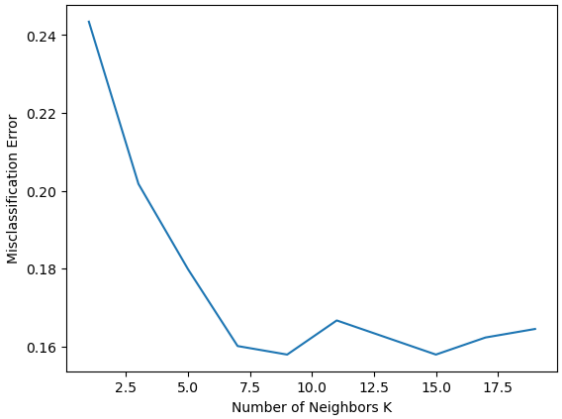


Fig 13: Misclassification error vs k Plot

From the above graph and MSE we see that for K = 15 it is giving the best test accuracy and no drastic increase found from K=15.

3. Rebuilding K-Nearest Neighbors Model for K=15

Fitting the KNN model with n_neighbors (k=15) and predicting it on scaled train and test dataset.

4. Model performance for tuned KNN Model: -

a. Confusion Matrix

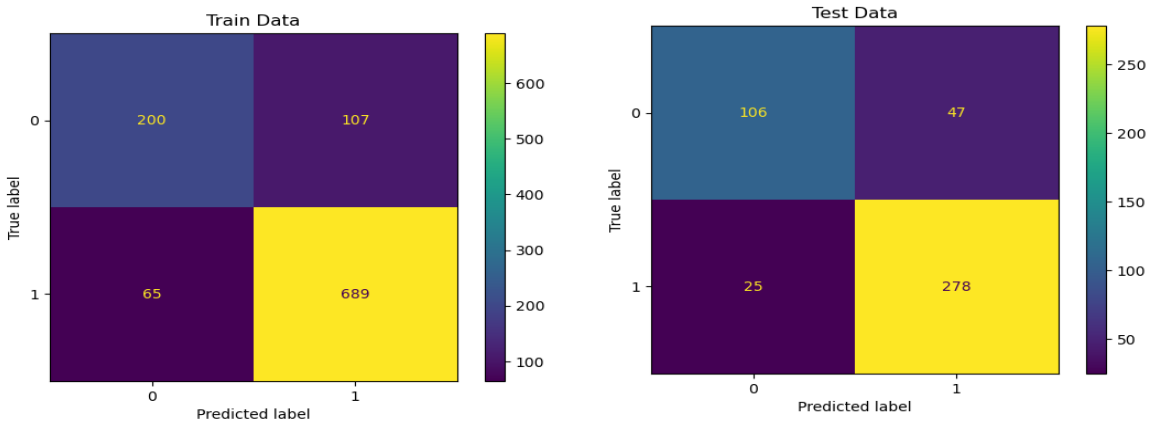


Fig 14: Confusion Matrix - Tunned K-Nearest Neighbors

b. Classification Report

Training Data report:

	precision	recall	f1-score	support
0	0.75	0.65	0.70	307
1	0.87	0.91	0.89	754
accuracy			0.84	1061
macro avg	0.81	0.78	0.79	1061
weighted avg	0.83	0.84	0.83	1061

Testing Data report:

	precision	recall	f1-score	support
0	0.81	0.69	0.75	153
1	0.86	0.92	0.89	303
accuracy			0.84	456
macro avg	0.83	0.81	0.82	456
weighted avg	0.84	0.84	0.84	456

Table 17: Classification Report - Tunned K-Nearest Neighbors

c. ROC Curve and AUC score

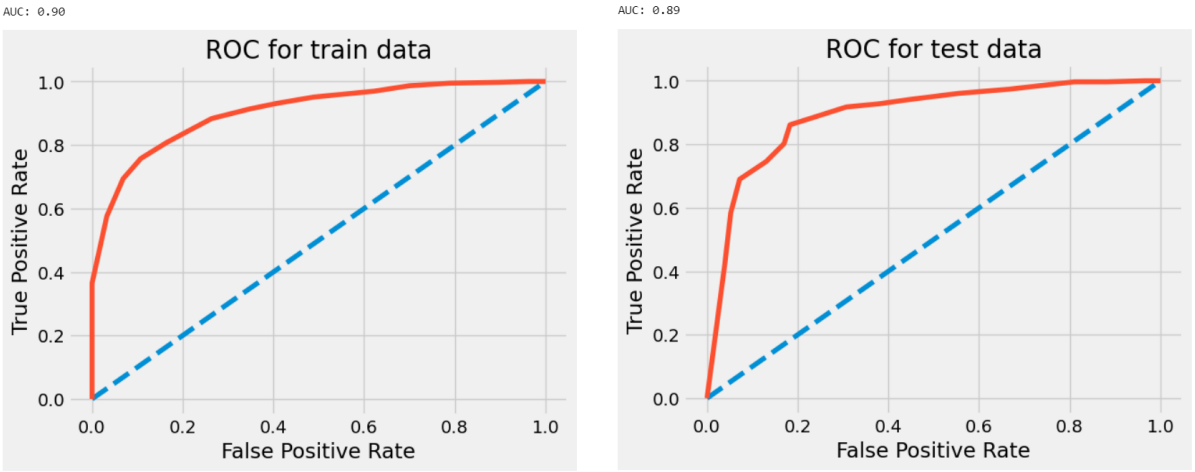


Fig 15: ROC Curve - Tunned KNN

d. Train and Test report comparison table between Default & Tuned Model

K-Nearest Neighbors Model (K=5)					
Sl.No	Index	Train Data		Test Data	
1	TN	222		107	
2	TP	687		267	
3	FN	67		36	
4	FP	85		46	
5	Accuracy	0.86		0.82	
6	AUC Score	0.93		0.87	
		Conservative	Labour	Conservative	Labour
7	Precision	0.77	0.89	0.75	0.85
8	Recall	0.72	0.91	0.70	0.88
9	F1 Score	0.74	0.90	0.72	0.87

Tunned K-Nearest Neighbors Model (K=15)					
Sl.No	Index	Train Data		Test Data	
1	TN	200		106	
2	TP	689		278	
3	FN	65		25	
4	FP	107		47	
5	Accuracy	0.84		0.84	
6	AUC Score	0.90		0.89	
		Conservative	Labour	Conservative	Labour
7	Precision	0.75	0.87	0.81	0.86
8	Recall	0.65	0.91	0.69	0.92
9	F1 Score	0.74	0.89	0.75	0.89

Table 18: Model performance Metrices - Default KNN & Tunned KNN

Observation: -

- There's no drastic difference found even after applying grid search with hyper parameters is almost similar to normal LDA model. There is slight increase in performance in terms of Accuracy, AUC Score, also there is slight increase in conservative class in terms of precision & F1 score and, also slight increase found in Labour class in terms of Precision, Recall & F1 score.

- In tunned model cases voters polled are 25 instances where model predicted as not polled. We observer that, in tunned model FN value decreased from 36 to 25.
- In tunned model Cases voters not polled, but model predicted them to be polled are 47, We observer that, in tunned model FN value slightly increase from 46 to 47.
- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Overall, the metrics are good fit.

Model Tuning 4 - Naïve Bayes model with SMOTE technique

The term SMOTE denotes (Synthetic Minority Oversampling Technique). It aims to balance the class distribution by randomly increasing minority class by drawing new samples along lines between existing minority data points. Here we will rebuild Naïve Bayes Model using SMOTE technique and check the performances.

1. **Importing Packages**

Import the necessary library of SMOTE from imblearn.over_sampling package & GaussianNB from sklearn.naive_bayes.

2. **Rebuilding Naïve bayes model with smote technique**

Fitting the SMOTE in train dataset to balances the minority class by replicating the samples. Then the balanced data is fitted into Gaussian naïve bayes model and predicting the train and test dataset.

3. **Model performance for tuned Naïve bayes model with smote technique**

a. Confusion Matrix

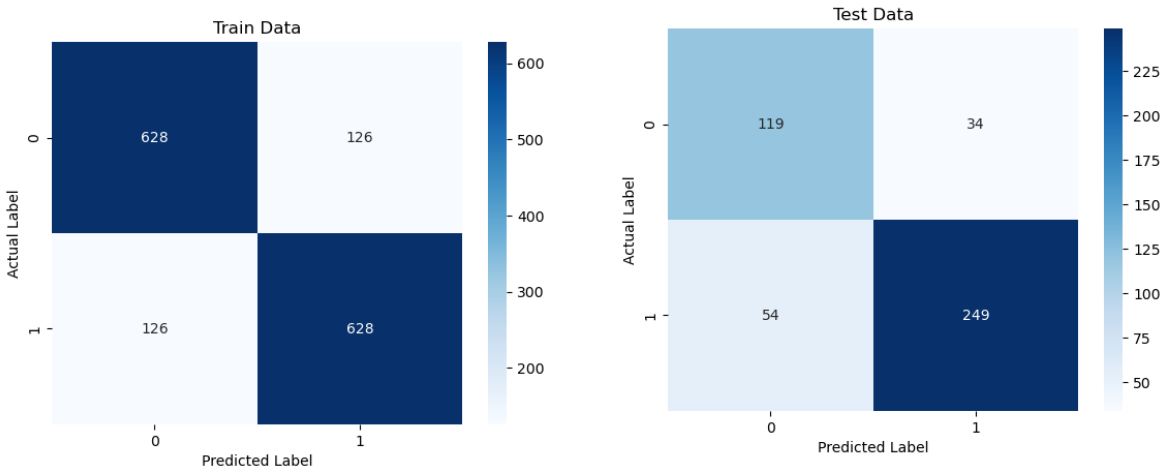


Fig 16: Confusion Matrix - Tunned Naïve Bayes

b. Classification Report

Training Data report:

	precision	recall	f1-score	support
0	0.83	0.83	0.83	754
1	0.83	0.83	0.83	754
accuracy			0.83	1508
macro avg	0.83	0.83	0.83	1508
weighted avg	0.83	0.83	0.83	1508

Testing Data report:

	precision	recall	f1-score	support
0	0.69	0.78	0.73	153
1	0.88	0.82	0.85	303
accuracy			0.81	456
macro avg	0.78	0.80	0.79	456
weighted avg	0.82	0.81	0.81	456

Table 19: Classification Report - Tunned Naïve Bayes

c. ROC Curve and AUC score

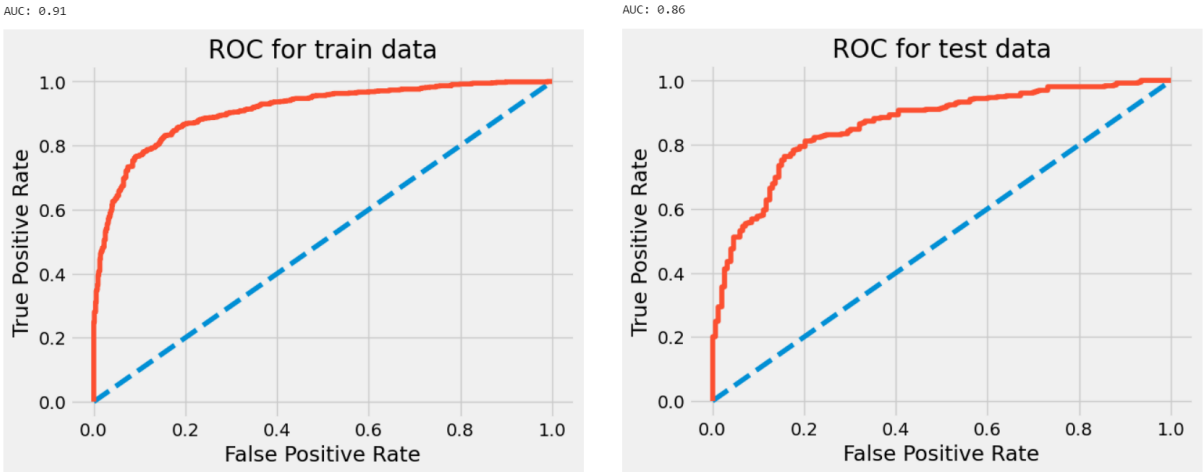


Fig 17: ROC Curve - Tunned Naïve Bayes

d. Train and Test report comparison table between Default & Tuned Model

Naive Bayes Model					
Sl.No	Index	Train Data		Test Data	
1	TN	211		112	
2	TP	675		263	
3	FN	79		40	
4	FP	96		41	
5	Accuracy	0.84		0.82	
6	AUC Score	0.84		0.82	
		Conservative	Labour	Conservative	Labour
7	Precision	0.73	0.88	0.74	0.87
8	Recall	0.69	0.90	0.73	0.87
9	F1 Score	0.71	0.89	0.73	0.87

Tunned Naive Bayes Model using SMOTE					
Sl.No	Index	Train Data		Test Data	
1	TN	628		119	
2	TP	628		249	
3	FN	126		54	
4	FP	126		34	
5	Accuracy	0.83		0.81	
6	AUC Score	0.91		0.86	
		Conservative	Labour	Conservative	Labour
7	Precision	0.83	0.83	0.69	0.88
8	Recall	0.83	0.83	0.78	0.82
9	F1 Score	0.83	0.83	0.73	0.85

Table 20: Model performance Metrices - Default Naïve Bayes & Tunned Naïve Bayes

Observation: -

- There's no drastic difference found even performing SMOTE technique and almost similar to normal Naïve Bayes Model. There is slight increase in performance in terms of AUC Score, also there is slight increase in conservative class in terms of Recall, also slight increase found in Labour class in terms of Precision & F1 score.

- In tunned model cases voters polled are 54 instances where model predicted as not polled. We observer that, in tunned model FN value increased from 40 to 54.
- In tunned model Cases voters not polled, but model predicted them to be polled are 34, We observer that, in tunned model FN value slightly decrease from 41 to 34.
- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Overall, the metrics are good fit.

Bagging (Random Forest should be applied for Bagging)

- Bagging is also known as Bootstrap aggregating; it is an ensemble learning technique that helps to improve the performance and accuracy of the machine learning algorithms. It is used to deal with bias-variance trade-offs and reduce the variance of prediction model.
- Random forests provide an improvement over bagged trees by way of a small tweak that makes the correlation between trees smaller. When building these decision trees, each time a split is considered, we don't take all the predictors but only a random subset of them. This forces the trees to use different predictors to split at different times. Even with the same training samples, if we grow two trees, we will get two different trees, because it will pick different variables each time.
- One good thing about random forest and bagging is that you can't overfit by putting in more trees. The benefit of adding more trees is that it brings a variance down more, and at some point, the variance will stop decreasing. Adding more trees won't help but will never hurt you. By looking at OOB errors (Out-of-bag (OOB) error is a straightforward way to estimate test error of the model.) you can just decide when you've done enough.

1. **Importing Packages**

Import the necessary libraries of RandomForestClassifier & BaggingClassifier from sklearn.ensemble package.

2. **Hyperparameter Tunning**

Here we fitting the train data in Random Forest model using RandomForestClassifier from sklearn.ensemble.

max_depth=4, max_features=5, min_samples_leaf=25, min_samples_split=50, n_estimators=101

As per the industries standards we are taking various hyper parameters to build our decision tree, they are as follows:-

- Max_depth = (This can be of any range) - To prune our decision tree to the best height.
- min_samples_leaf' = 2-3 % of the dataset observation.
- min_samples_split = 3 times of min_sample_ leaf.

3. Feature Importance

Variable importance is used to check the importance of variables and how the Decision tree is split into branches.

Below is the output retrieved from Jupyter using feature_importances_ command:-

	Imp
Hague	0.353292
Blair	0.268343
Europe	0.221729
political.knowledge	0.057267
economic.cond.national	0.056988
age	0.028539
economic.cond.household	0.011371
gender	0.002471

4. Building Random Forest Model using Bagging

Bagging classifier is fitted to training data with base estimator as Random Forest and predicting the train and test dataset.

5. Model performance of Bagging (Random Forest)

a. Confusion Matrix

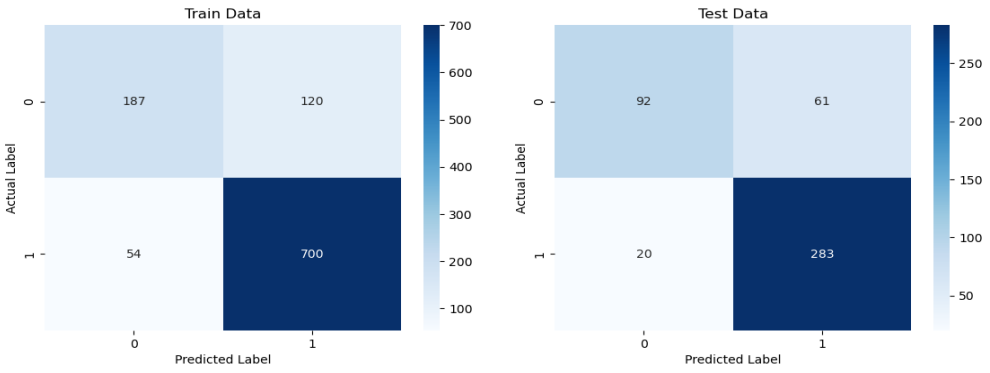


Fig 18: Confusion Matrix - Bagging

b. Classification Report

Training Data report:

	precision	recall	f1-score	support
0	0.78	0.61	0.68	307
1	0.85	0.93	0.89	754
accuracy			0.84	1061
macro avg	0.81	0.77	0.79	1061
weighted avg	0.83	0.84	0.83	1061

Testing Data report:

	precision	recall	f1-score	support
0	0.82	0.60	0.69	153
1	0.82	0.93	0.87	303
accuracy			0.82	456
macro avg	0.82	0.77	0.78	456
weighted avg	0.82	0.82	0.81	456

Table 21: Classification Report - Bagging

c. ROC Curve and AUC score

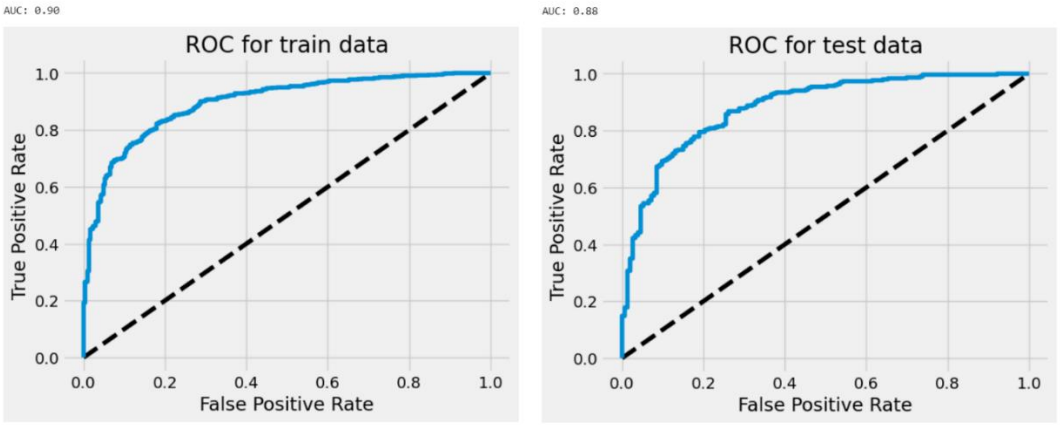


Fig 19: ROC Curve - Bagging

d. Train and Test report comparison table for Bagging (Random Forest)

Bagging (Random Forest) Model					
Sl.No	Index	Train Data		Test Data	
1	TN	187		92	
2	TP	700		283	
3	FN	54		20	
4	FP	120		61	
5	Accuracy	0.84		0.82	
6	AUC Score	0.90		0.88	
		Conservative	Labour	Conservative	Labour
7	Precision	0.78	0.85	0.82	0.82
8	Recall	0.61	0.93	0.60	0.93
9	F1 Score	0.68	0.89	0.69	0.87

Table 22: Model performance Metrices - Bagging

Observation: -

- From the analysis we can say that, model does a better job of correctly classifying the Labour Party voters then Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Bagging with Random Forest has performed exceptionally well on the train data for both the 0classes 0 and 1 in terms of Recall, Precision, F1 score.
- There is no drastic difference found between testing & training dataset this indicates the model is neither overfit or underfit.

Boosting Model:

The term 'Boosting' refers to a family of algorithms which converts weak learner to strong learners Boosting is used to create a collection of predictors. In this technique, learners are learned sequentially with early learners fitting simple models to the data and then analysing data for errors. Consecutive trees (random sample) are fit and at every step, the goal is to improve the accuracy from the prior tree.

For choosing the right distribution, here are the following steps: -

Step 1: The base learner takes all the distributions and assign equal weight or attention to each observation.

Step 2: If there is any prediction error caused by first base learning algorithm, then we pay higher attention to observations having prediction error. Then, we apply the next base learning algorithm.

Step 3: Iterate Step 2 till the limit of base learning algorithm is reached or higher accuracy is achieved.

Finally, it combines the outputs from weak learner and create a strong learner which eventually improves the prediction power of the model. Boosting highly focuses on mis-classified class and higher errors.

AdaBoost (Adaptive Boosting)

AdaBoost or Adaptive Boosting is the first Boosting ensemble model. The method automatically adjusts its parameters to the data based on the actual performance in the current iteration. Meaning, both the weights for re-weighting the data and the weights for the final aggregation are re-computed iteratively.

In practice, this boosting technique is used with simple classification trees or stumps as base-learners, which resulted in improved performance compared to the classification by one tree or another single base-learner.

1. **Importing Packages**

Import the necessary library of AdaBoostClassifier from sklearn.ensemble package.

2. **Building AdaBoosting**
   Here we fitting the train data in AdaBoostClassifier model using AdaBoostClassifier from sklearn.ensemble with n_estimators = 100 and random state =1 and predicting the training and testing dataset. The n_estimators parameter is used to control the number of weak learners, learning_rate parameter controls the contribution of all the vulnerable learners in the final output, base_estimator parameter helps to specify different machine learning algorithms.

3. **Model performance of AdaBoosting**

a. Confusion Matrix

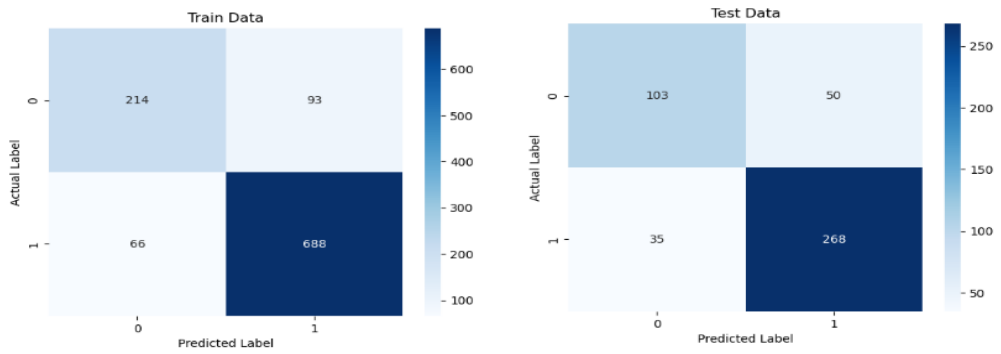


Fig 20: Confusion Matrix – Ada Bossting

b. Classification Report

Training Data report:

	precision	recall	f1-score	support
0	0.76	0.70	0.73	307
1	0.88	0.91	0.90	754
accuracy			0.85	1061
macro avg	0.82	0.80	0.81	1061
weighted avg	0.85	0.85	0.85	1061

Testing Data report:

	precision	recall	f1-score	support
0	0.75	0.67	0.71	153
1	0.84	0.88	0.86	303
accuracy			0.81	456
macro avg	0.79	0.78	0.79	456
weighted avg	0.81	0.81	0.81	456

Table 23: Classification Report - Ada Boosting

c. ROC Curve and AUC score

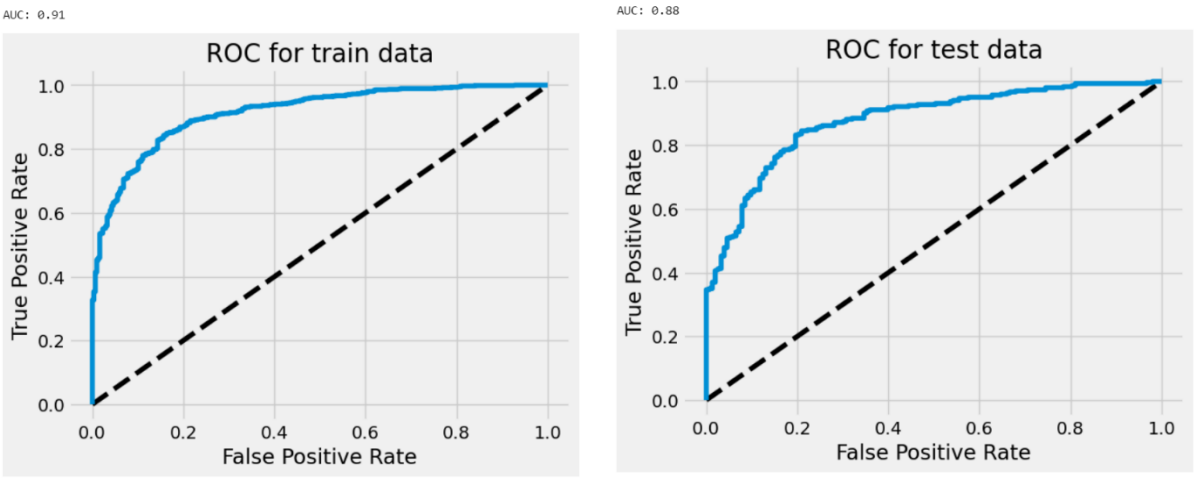


Fig 21: ROC Curve - Ada Bossting

d. Train and Test report comparison table for AdaBossting

AdaBossting Model					
Sl.No	Index	Train Data		Test Data	
1	TN	214		103	
2	TP	688		268	
3	FN	66		35	
4	FP	93		50	
5	Accuracy	0.85		0.81	
6	AUC Score	0.91		0.88	
		Conservative	Labour	Conservative	Labour
7	Precision	0.76	0.88	0.75	0.84
8	Recall	0.70	0.91	0.67	0.88
9	F1 Score	0.73	0.90	0.71	0.86

Table 24: Model performance Metrices - Ada Boosting

Observation: -

- In cases voters polled are 35 instances where model predicted as not polled.
- In Cases voters not polled, but model predicted them to be polled are 50, We observer that,

- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Overall, the metrics are good fit.

Gradient Boosting Model

In gradient boosting algorithm, we train multiple models sequentially, then for each new model, the model gradually minimizes the loss function using the Gradient Descent method. The Gradient Boosting Tree algorithm takes decision trees as a weak leaner, because the nodes in a decision tree consider different branch of features for selecting the best split, which means all the trees are not the same. Hence, they will capture different outputs from the data.

1. **Importing Packages**

Import the necessary library of GradientBoostingClassifier from sklearn.ensemble package.

2. **Building Gradient Boosting**

Here we fitting the train data in Gradient Boosting model using GradientBoostingClassifier from sklearn.ensemble with random state =1 and predicting the training and testing dataset.

3. **Model performance of Gradient Boosting**

a. Confusion Matrix

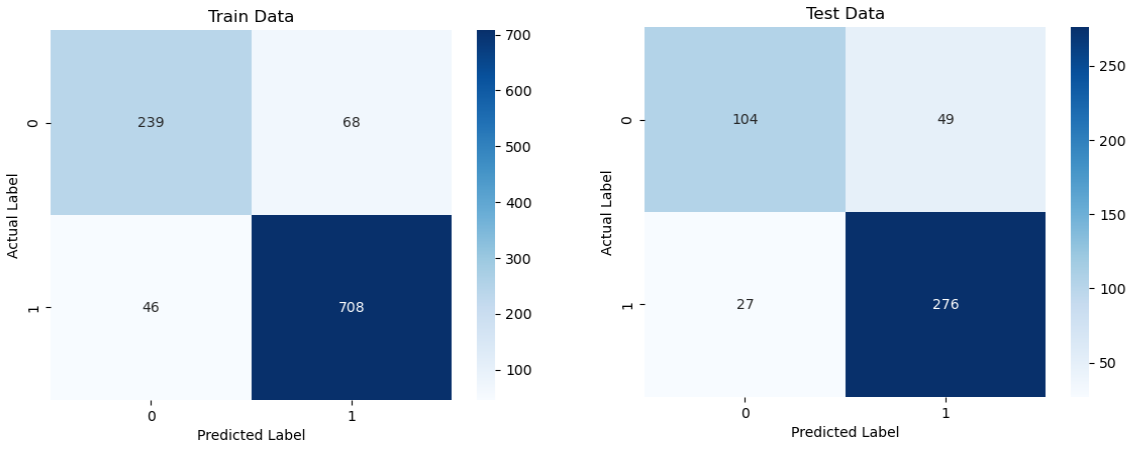


Fig 22: Confusion Matrix – Gradient Boosting

b. Classification Report

Training Data report:

	precision	recall	f1-score	support
0	0.84	0.78	0.81	307
1	0.91	0.94	0.93	754
accuracy			0.89	1061
macro avg	0.88	0.86	0.87	1061
weighted avg	0.89	0.89	0.89	1061

Testing Data report:

	precision	recall	f1-score	support
0	0.79	0.68	0.73	153
1	0.85	0.91	0.88	303
accuracy			0.83	456
macro avg	0.82	0.80	0.81	456
weighted avg	0.83	0.83	0.83	456

Table 25: Classification Report - Gradient Boosting

c. ROC Curve and AUC score

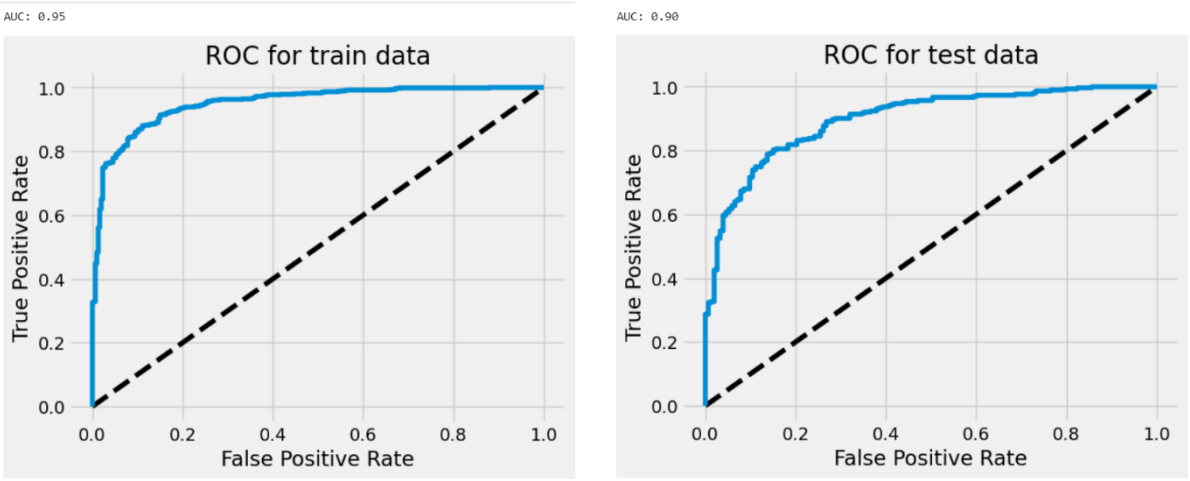


Fig 23: ROC Curve - Gradient Boosting

d. Train and Test report comparison table for AdaBossting

Gradient Bossting Model					
Sl.No	Index	Train Data		Test Data	
1	TN	239		104	
2	TP	708		276	
3	FN	46		27	
4	FP	68		49	
5	Accuracy	0.89		0.83	
6	AUC Score	0.95		0.90	
		Conservative	Labour	Conservative	Labour
7	Precision	0.84	0.91	0.79	0.85
8	Recall	0.78	0.94	0.68	0.91
9	F1 Score	0.81	0.93	0.73	0.88

Table 26: Model performance Metrices - Gradient Boosting

Observation: -

- In cases voters polled are 27 instances where model predicted as not polled.
- In Cases voters not polled, but model predicted them to be polled are 49, We observer that,
- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Overall, the metrics are good fit.

1.7 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. Final Model: Compare the models and write inference which model is best/optimized.

Model performance:

This helps us to understand how good our model got trained. Model performance can be done only after the prediction of training and testing dataset . Here we validate if the model is underfitting or overfitting by checking certain parameters. Following methods are used to evaluate the model performance:

- **Confusion Matrix**

A confusion matrix is a table that is used to define the performance of a classification algorithm. A confusion matrix visualizes and summarizes the performance of a classification algorithm.

		Predicted	
		Negative	Positive
Actual	Negative	TN	FP
	Positive	FN	TP

-TN,TP - Correct Prediction (True Negative, True Positive)
- FP,FN - Incorrect prediction (False Positive, False Negative)

- **Classification Report**

1. Accuracy : Accuracy are used to identify how accurately/Cleanly , the model classifies the data point. Lesser the false predictions, more the accuracy.
Accuracy = (TP + TN) / (TP + TN + FP + FN).

2. Precision: Among the points identified as positive by the model, but how many points are actual positive.
If type(I) error is low pression will he high. Type(I) error and precision are inversely proportional to each other.
Precision = TP/(TP + FP).

3. Recall (Sensitivity): How many of the actual true data points are identified as True data points by the model. False Negative are those points should have been identified as True. Higher the sensitivity lowers the false negative(Type(II) error). Type(II) error and sensitivity are inversely proportional to each other

Recall = TP/(TP + FN)

4. Score: F1 Score computes an hormonic mean between Precision and Recall. It tells us both Type(I) and Type(II) error in a particular model is higher or lower on an average. If the F1 is good, that indicated model contains less false positives and less false negatives. F1 score is considered to be perfect when it tends to be 1 and model is a total failure when it tends to be 0.

F1 score = 2 x [(Precision x Recall) / (Precision + Recall)

- **ROC Curve:** ROC curve is a graphic representation of classifier performance. This curve plots two parameters: True Positive Rate. False Positive Rate. Higher the cure stronger the model, flatter the ROC curve weakest the model.

- **AUC Score:** AUC score gives the value of area under the ROC curve. The higher the AUC score, the better the performance of the model at distinguishing between the positive and negative.

Model 1 – Logistic Regression

a) Confusion Matrix

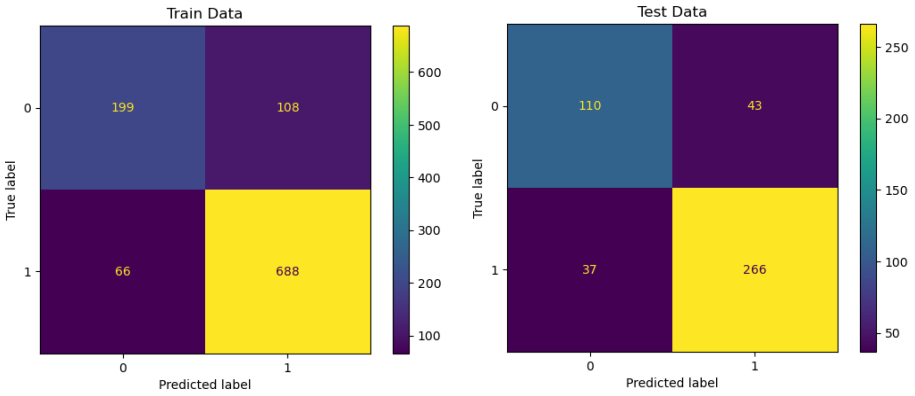


Fig 24: Confusion Matrix - Logistic Regression

b) Classification Report

Train Report

	precision	recall	f1-score	support
0	0.75	0.65	0.70	307
1	0.86	0.91	0.89	754
accuracy			0.84	1061
macro avg	0.81	0.78	0.79	1061
weighted avg	0.83	0.84	0.83	1061

Test Report

	precision	recall	f1-score	support
0	0.75	0.72	0.73	153
1	0.86	0.88	0.87	303
accuracy			0.82	456
macro avg	0.80	0.80	0.80	456
weighted avg	0.82	0.82	0.82	456

Table 27: Classification Report - Logistic Regression

c) ROC Curve and AUC score

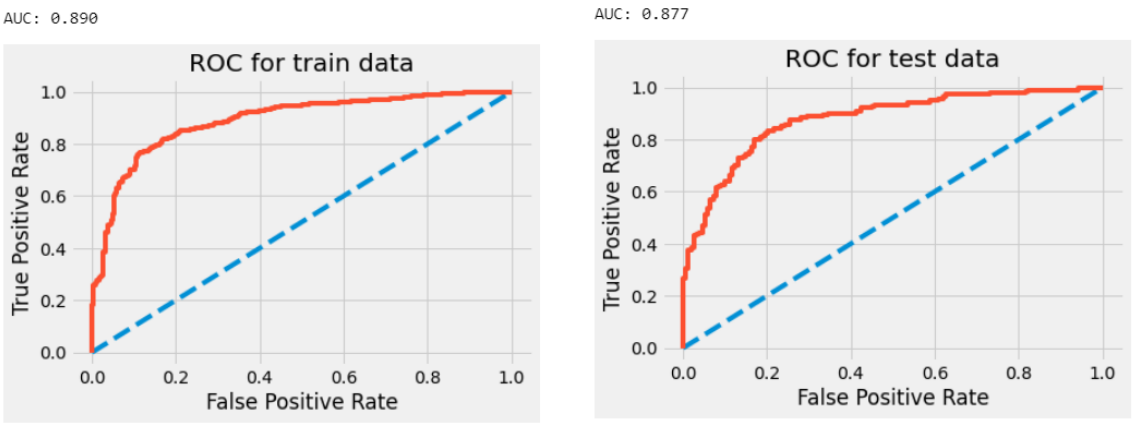


Fig 25: ROC Curve - Logistic Regression

d) Train and Test report comparison table

Logistic Regression Model					
Sl.No	Index	Train Data		Test Data	
1	TN	199		110	
2	TP	688		266	
3	FN	66		37	
4	FP	108		43	
5	Accuracy	0.84		0.82	
6	AUC Score	0.89		0.87	
		Conservative	Labour	Conservative	Labour
7	Precision	0.75	0.86	0.75	0.86
8	Recall	0.65	0.91	0.72	0.88
9	F1 Score	0.70	0.89	0.73	0.87

Table 8: Model performance Metrices - Logistic Regression

Model 2 – Tuned Logistic Regression

a. Confusion Matrix

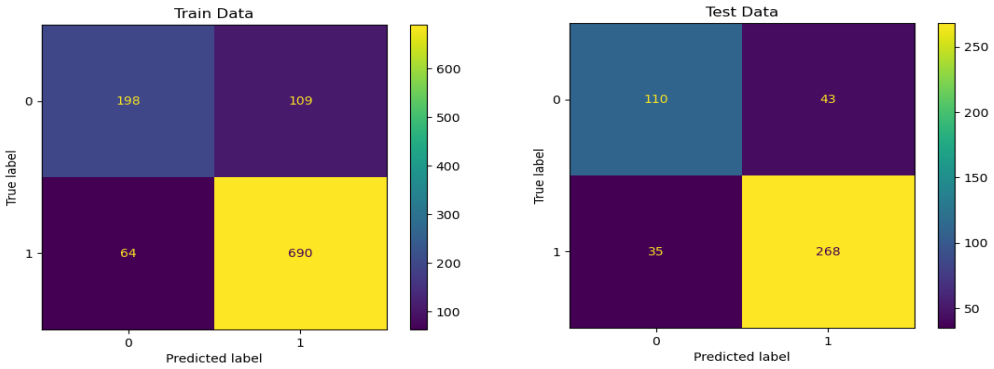


Fig 9: Confusion Matrix - Tunned Logistic Regression

b. Classification Report

Train Report

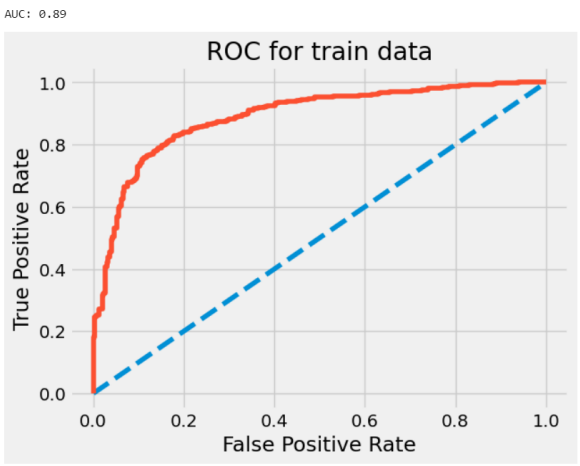
	precision	recall	f1-score	support
0	0.76	0.64	0.70	307
1	0.86	0.92	0.89	754
accuracy			0.84	1061
macro avg	0.81	0.78	0.79	1061
weighted avg	0.83	0.84	0.83	1061

Test Report

	precision	recall	f1-score	support
0	0.76	0.72	0.74	153
1	0.86	0.88	0.87	303
accuracy			0.83	456
macro avg	0.81	0.80	0.81	456
weighted avg	0.83	0.83	0.83	456

Table 12: Classification Report - Tunned Logistic Regression

c. ROC Curve and AUC score



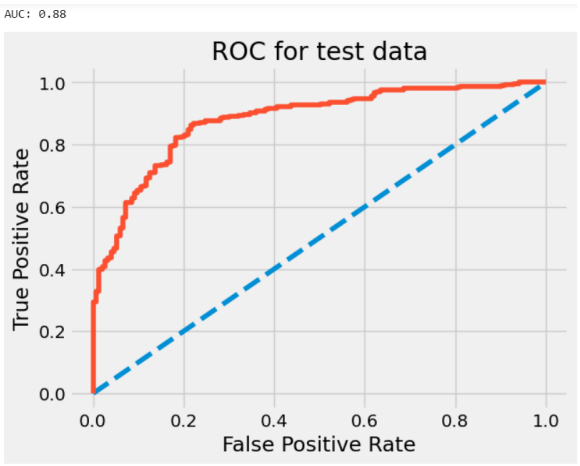


Fig 10: ROC Curve - Tunned Logistic Regression

d. Train & Test report comparison table between Default & Tunned LR Model

Logistic Regression Model					
Sl.No	Index	Train Data		Test Data	
1	TN	199		110	
2	TP	688		266	
3	FN	66		37	
4	FP	108		43	
5	Accuracy	0.84		0.82	
6	AUC Score	0.89		0.87	
		Conservative	Labour	Conservative	Labour
7	Precision	0.75	0.86	0.75	0.86
8	Recall	0.65	0.91	0.72	0.88
9	F1 Score	0.7	0.89	0.73	0.87

Tunned – Logistic Regression Model					
Sl.No	Index	Train Data		Test Data	
1	TN	198		110	
2	TP	690		268	
3	FN	64		35	
4	FP	109		43	
5	Accuracy	0.84		0.83	
6	AUC Score	0.89		0.88	
		Conservative	Labour	Conservative	Labour
7	Precision	0.76	0.86	0.76	0.86
8	Recall	0.64	0.92	0.72	0.88
9	F1 Score	0.7	0.89	0.74	0.87

Table 13: Model performance Metrices - Default Logistic Regression & Tunned Logistic Regression

Observation: -

- There's no drastic difference found even after applying grid search with hyper parameters is almost similar to normal Logistic regression model. There is slight increase in performance in the conservative class, Accuracy & AUC score.
- Cases voters polled, there are 35 instances where model predicted as not polled.
- Cases voters not polled, but model predicted them to be polled are 43.

- ❖ In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- ❖ From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- ❖ Overall, the metrics are good fit.

Model 3 - Linear Discriminant Analysis Model

a. Confusion Matrix

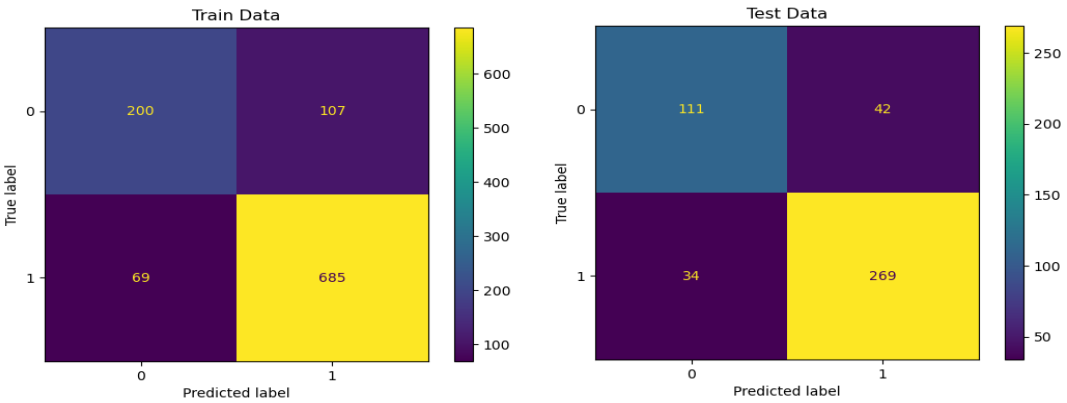


Fig 26: Confusion Matrix -LDA

b. Classification Report

Train Report

	precision	recall	f1-score	support
0	0.74	0.65	0.69	307
1	0.86	0.91	0.89	754
accuracy			0.83	1061
macro avg	0.80	0.78	0.79	1061
weighted avg	0.83	0.83	0.83	1061

Test Report

	precision	recall	f1-score	support
0	0.77	0.73	0.74	153
1	0.86	0.89	0.88	303
accuracy			0.83	456
macro avg	0.82	0.81	0.81	456
weighted avg	0.83	0.83	0.83	456

Table 29: Classification Report - LDA

c. ROC Curve and AUC score

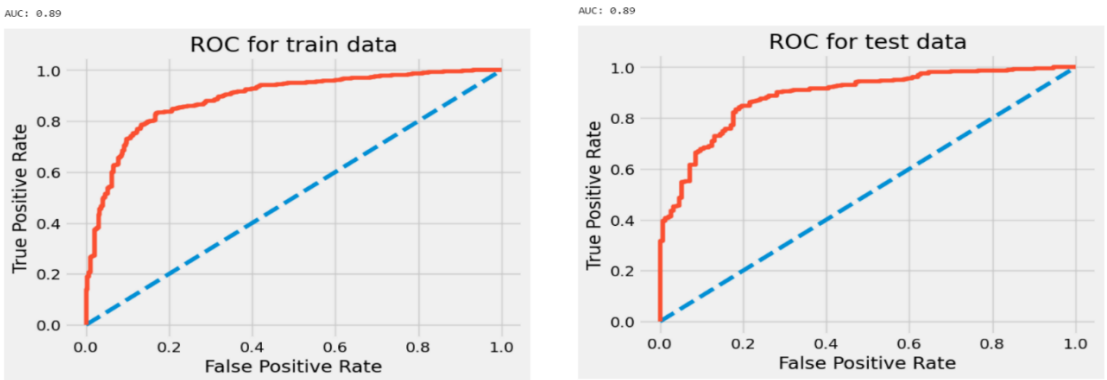


Fig 27: ROC Curve - LDA

d. Train and Test report comparison table

Linear Discriminative Analysis Model					
Sl.No	Index	Train Data		Test Data	
1	TN	200		111	
2	TP	685		269	
3	FN	69		34	
4	FP	107		42	
5	Accuracy	0.83		0.83	
6	AUC Score	0.89		0.89	
		Conservative	Labour	Conservative	Labour
7	Precision	0.74	0.86	0.77	0.86
8	Recall	0.65	0.91	0.73	0.89
9	F1 Score	0.69	0.89	0.74	0.88

Table 30: Model performance Metrices - LDA

Model 4 - Tuned Linear Discriminant Analysis Model

a. Confusion Matrix

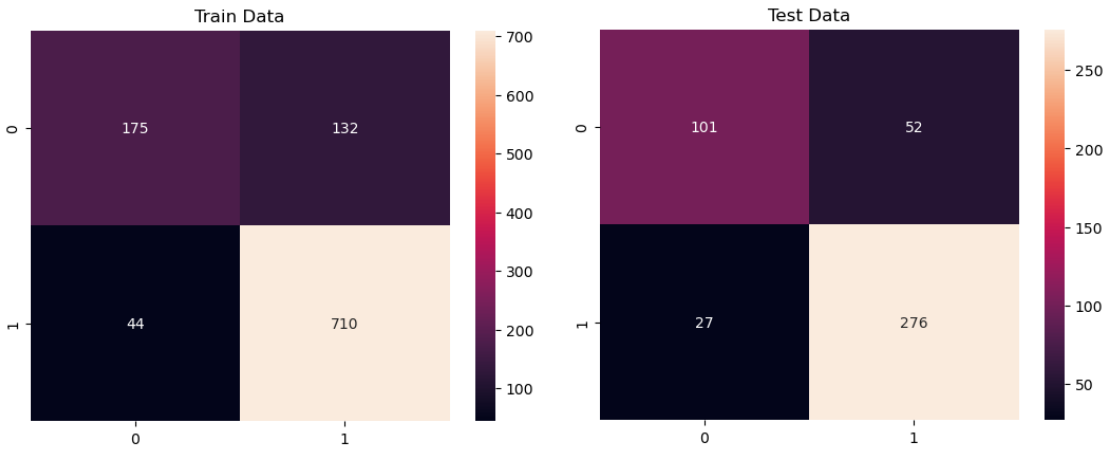


Fig 11: Confusion Matrix - Tunned LDA

b. Classification Report

Training Data report :

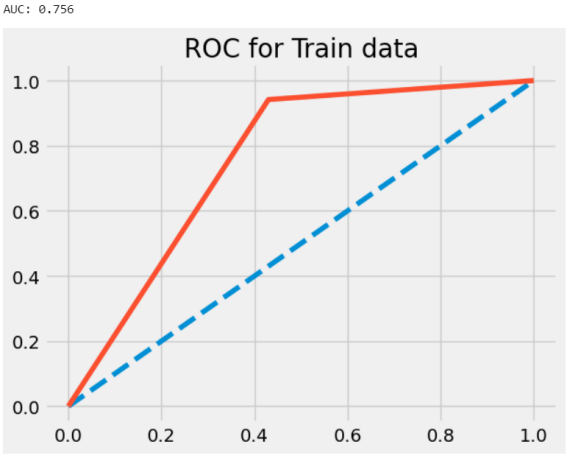
	precision	recall	f1-score	support
0	0.80	0.57	0.67	307
1	0.84	0.94	0.89	754
accuracy			0.83	1061
macro avg	0.82	0.76	0.78	1061
weighted avg	0.83	0.83	0.82	1061

Testing Data report:

	precision	recall	f1-score	support
0	0.79	0.66	0.72	153
1	0.84	0.91	0.87	303
accuracy			0.83	456
macro avg	0.82	0.79	0.80	456
weighted avg	0.82	0.83	0.82	456

Table 15: Classification Report - Tunned Linear Discriminative Analysis

c. ROC Curve and AUC score



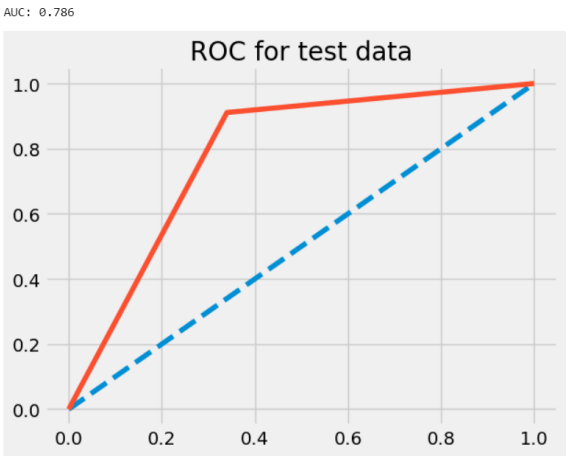


Fig 12: ROC Curve - Tunned LDA

d. Train and Test report comparison table between Default & Tuned Model

Linear Discriminative Analysis Model (Cut_off 0.5)					
Sl.No	Index	Train Data		Test Data	
1	TN	200		111	
2	TP	685		269	
3	FN	69		34	
4	FP	107		42	
5	Accuracy	0.83		0.83	
6	AUC Score	0.89		0.89	
		Conservative	Labour	Conservative	Labour
7	Precision	0.74	0.86	0.77	0.86
8	Recall	0.65	0.91	0.73	0.89
9	F1 Score	0.69	0.89	0.74	0.88

Tunned – Linear Discriminate Analysis Model (Cut_off 0.4)					
Sl.No	Index	Train Data		Test Data	
1	TN	175		101	
2	TP	710		276	
3	FN	44		27	
4	FP	132		52	
5	Accuracy	0.83		0.83	
6	AUC Score	0.76		0.79	
		Conservative	Labour	Conservative	Labour
7	Precision	0.80	0.84	0.79	0.84
8	Recall	0.57	0.94	0.66	0.91
9	F1 Score	0.67	0.89	0.72	0.87

Table 16: Model performance Metrices - Default LDA & Tunned LDA

Observation: -

- There's no drastic difference found even after applying grid search with hyper parameters is almost similar to normal LDA model. There is slight increase in performance in the conservative class in terms of precision.
- There is slight increase in performance in the Labour class in terms of Recall.
- In tunned model cases voters polled are 27 instances where model predicted as not polled. We observer that FN decreased from 34 to 27.
- In tunned model Cases voters not polled, but model predicted them to be polled are 52, We observer that FP increased from 42 to 52.
- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.

- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Overall, the metrics are good fit.

Model 5 - K-Nearest Neighbors Model

a. Confusion Matrix

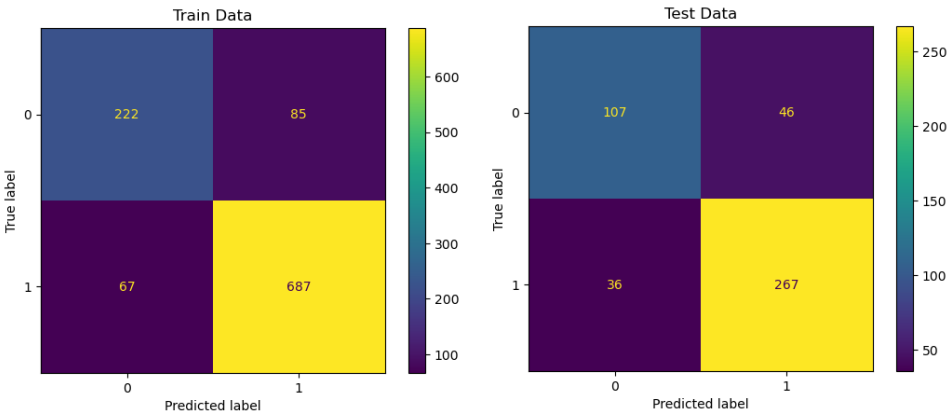


Fig 28: Confusion Matrix -KNN

b. Classification Report

Train Report

	precision	recall	f1-score	support
0	0.77	0.72	0.74	307
1	0.89	0.91	0.90	754
accuracy			0.86	1061
macro avg	0.83	0.82	0.82	1061
weighted avg	0.85	0.86	0.86	1061

Test Report

	precision	recall	f1-score	support
0	0.75	0.70	0.72	153
1	0.85	0.88	0.87	303
accuracy			0.82	456
macro avg	0.80	0.79	0.79	456
weighted avg	0.82	0.82	0.82	456

Table 31: Classification Report - KNN

c. ROC Curve and AUC score

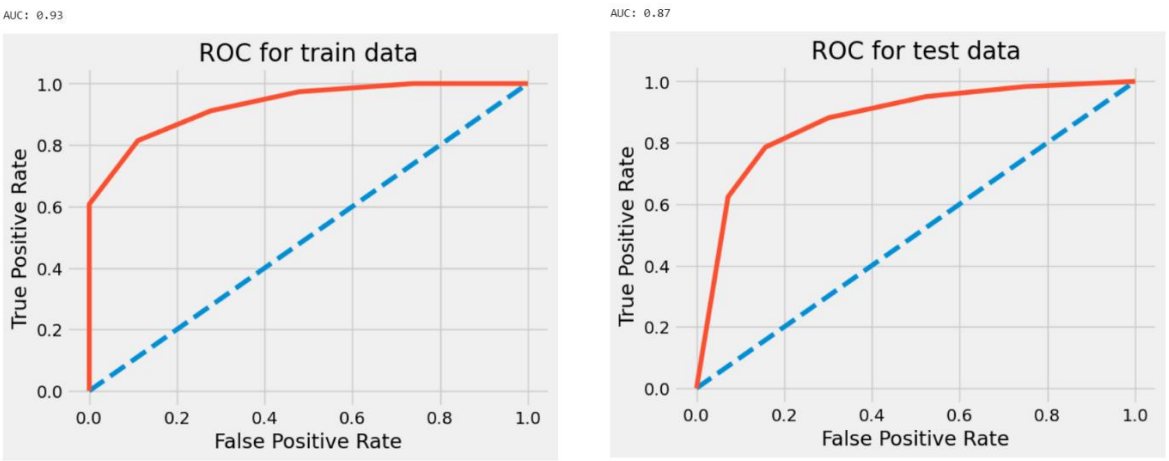


Fig 29: ROC Curve - KNN

d. Train and Test report comparison table

K-Nearest Neighbors Model					
Sl.No	Index	Train Data		Test Data	
1	TN	222		107	
2	TP	687		267	
3	FN	67		36	
4	FP	85		46	
5	Accuracy	0.86		0.82	
6	AUC Score	0.93		0.87	
		Conservative	Labour	Conservative	Labour
7	Precision	0.77	0.89	0.75	0.85
8	Recall	0.72	0.91	0.70	0.88
9	F1 Score	0.74	0.90	0.72	0.87

Table 32: Model performance Metrices - KNN

Model 6 – Tuned K-Nearest Neighbors Model

a. Confusion Matrix

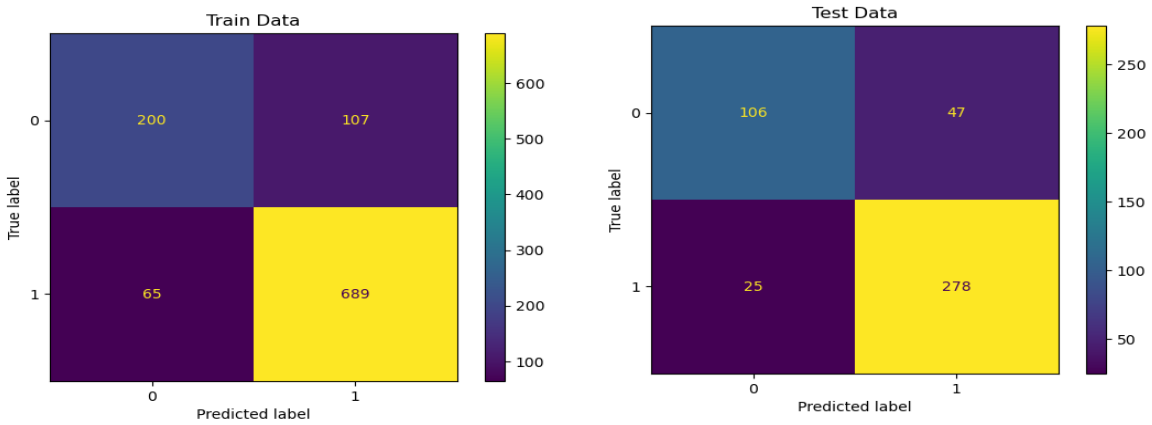


Fig 14: Confusion Matrix - Tunned K-Nearest Neighbors

b. Classification Report

Training Data report:

	precision	recall	f1-score	support
0	0.75	0.65	0.70	307
1	0.87	0.91	0.89	754
accuracy			0.84	1061
macro avg	0.81	0.78	0.79	1061
weighted avg	0.83	0.84	0.83	1061

Testing Data report:

	precision	recall	f1-score	support
0	0.81	0.69	0.75	153
1	0.86	0.92	0.89	303
accuracy			0.84	456
macro avg	0.83	0.81	0.82	456
weighted avg	0.84	0.84	0.84	456

Table 17: Classification Report - Tunned K-Nearest Neighbors

c. ROC Curve and AUC score

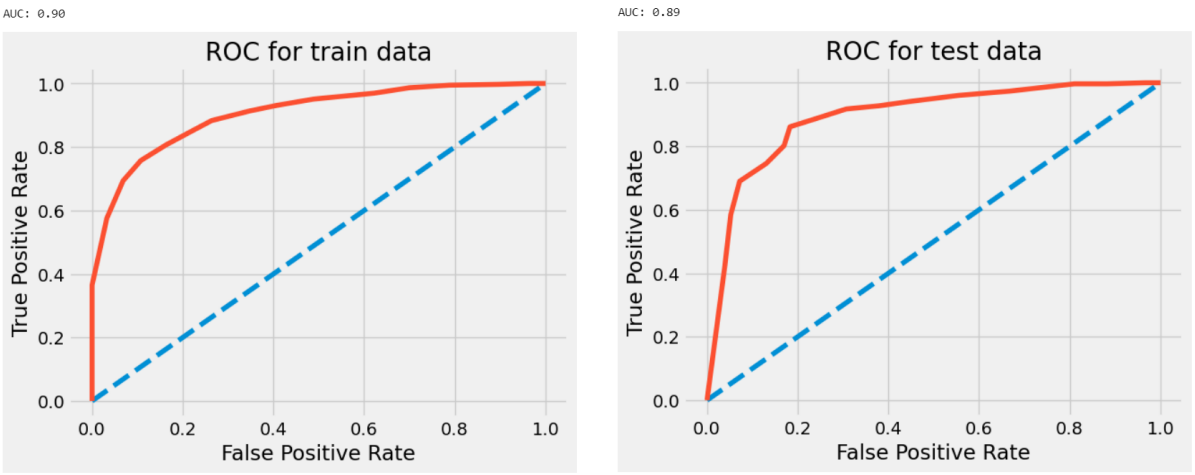


Fig 15: ROC Curve - Tunned KNN

d. Train and Test report comparison table between Default & Tuned Model

K-Nearest Neighbors Model (K=5)					
Sl.No	Index	Train Data		Test Data	
1	TN	222		107	
2	TP	687		267	
3	FN	67		36	
4	FP	85		46	
5	Accuracy	0.86		0.82	
6	AUC Score	0.93		0.87	
		Conservative	Labour	Conservative	Labour
7	Precision	0.77	0.89	0.75	0.85
8	Recall	0.72	0.91	0.70	0.88
9	F1 Score	0.74	0.90	0.72	0.87

Tunned K-Nearest Neighbors Model (K=15)					
Sl.No	Index	Train Data		Test Data	
1	TN	200		106	
2	TP	689		278	
3	FN	65		25	
4	FP	107		47	
5	Accuracy	0.84		0.84	
6	AUC Score	0.90		0.89	
		Conservative	Labour	Conservative	Labour
7	Precision	0.75	0.87	0.81	0.86
8	Recall	0.65	0.91	0.69	0.92
9	F1 Score	0.74	0.89	0.75	0.89

Table 18: Model performance Metrices - Default KNN & Tunned KNN

Observation: -

- There's no drastic difference found even after applying grid search with hyper parameters is almost similar to normal KNN model. There is slight increase in performance in terms of Accuracy, AUC Score, also there is slight increase in conservative class in terms of precision & F1 score and, also slight increase found in Labour class in terms of Precision, Recall & F1 score.
- In tunned model cases voters polled are 25 instances where model predicted as not polled. We observer that, in tunned model FN value decreased from 36 to 25.
- In tunned model Cases voters not polled, but model predicted them to be polled are 47, We observer that, in tunned model FN value slightly increase from 46 to 47.
- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.

❖ Overall, the metrics are good fit.

Model 7 – Naïve Bayes Model

a. Confusion Matrix

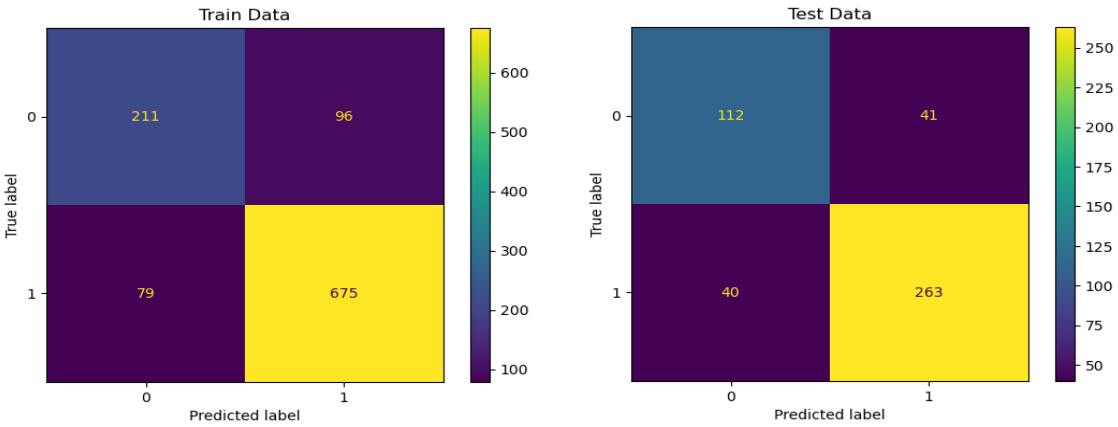


Fig 30: Confusion Matrix -Naïve Bayes

b. Classification Report

Train Report

	precision	recall	f1-score	support
0	0.73	0.69	0.71	307
1	0.88	0.90	0.89	754
accuracy			0.84	1061
macro avg	0.80	0.79	0.80	1061
weighted avg	0.83	0.84	0.83	1061

Test Report

	precision	recall	f1-score	support
0	0.74	0.73	0.73	153
1	0.87	0.87	0.87	303
accuracy			0.82	456
macro avg	0.80	0.80	0.80	456
weighted avg	0.82	0.82	0.82	456

Table 33: Classification Report – Naïve Bayes

c. ROC Curve and AUC score

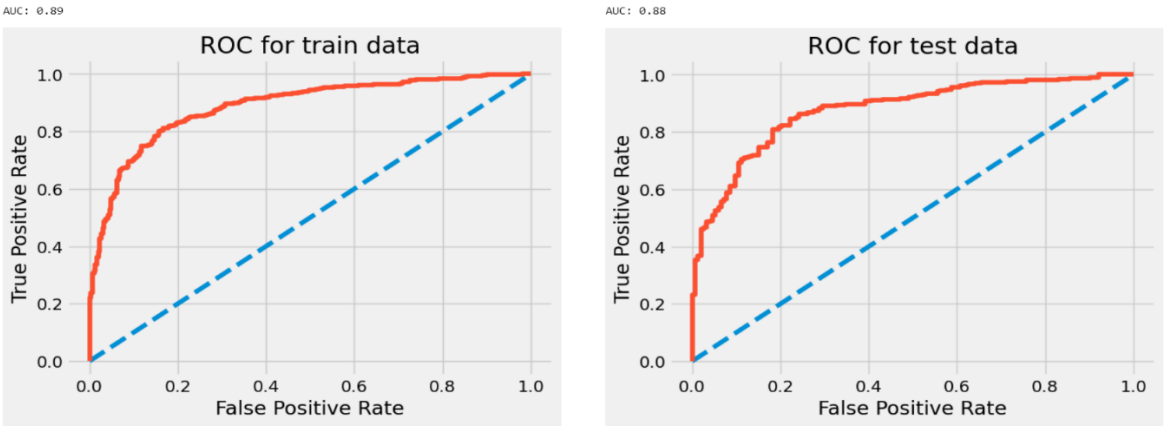


Fig 31: ROC Curve - Naïve Bayes

d. Train and Test report comparison table

Naive Bayes Model					
SI.No	Index	Train Data		Test Data	
1	TN	211		112	
2	TP	675		263	
3	FN	79		40	
4	FP	96		41	
5	Accuracy	0.84		0.82	
6	AUC Score	0.89		0.88	
		Conservative	Labour	Conservative	Labour
7	Precision	0.73	0.88	0.74	0.87
8	Recall	0.69	0.90	0.73	0.87
9	F1 Score	0.71	0.89	0.73	0.87

Table 34: Model performance Metrices - Naïve Bayes

Model 8 – Tuned Naïve Bayes Model using SMOTE

a. Confusion Matrix

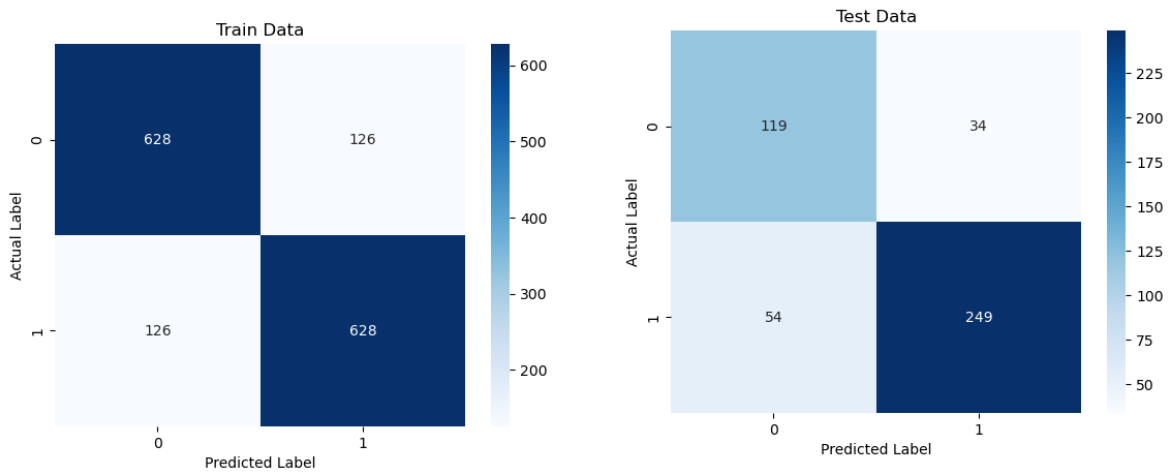


Fig 16: Confusion Matrix - Tunned Naïve Bayes

b. Classification Report

Training Data report:

	precision	recall	f1-score	support
0	0.83	0.83	0.83	754
1	0.83	0.83	0.83	754
accuracy			0.83	1508
macro avg	0.83	0.83	0.83	1508
weighted avg	0.83	0.83	0.83	1508

Testing Data report:

	precision	recall	f1-score	support
0	0.69	0.78	0.73	153
1	0.88	0.82	0.85	303
accuracy			0.81	456
macro avg	0.78	0.80	0.79	456
weighted avg	0.82	0.81	0.81	456

Table 19: Classification Report - Tunned Naïve Bayes

c. ROC Curve and AUC score

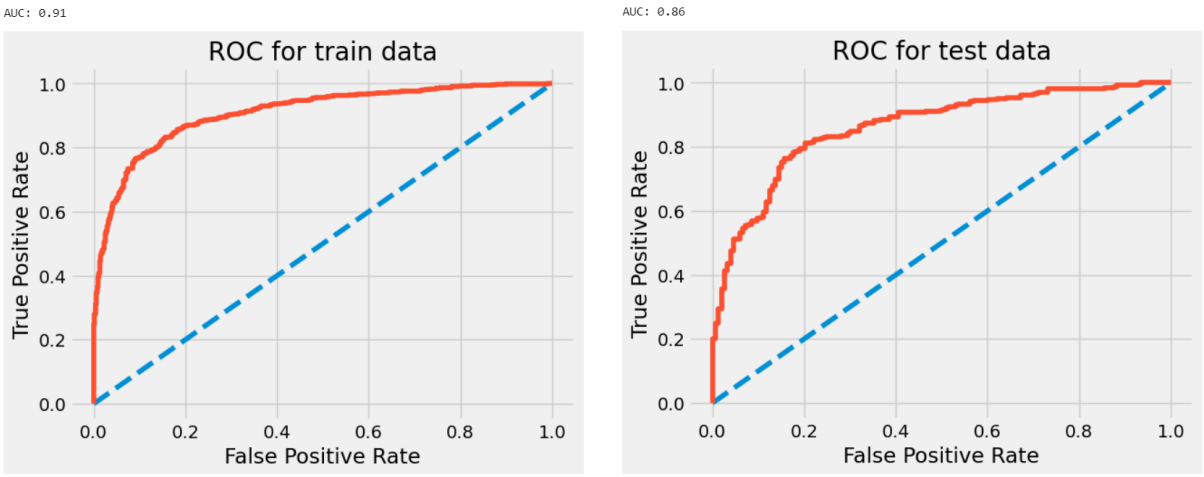


Fig 17: ROC Curve - Tunned Naïve Bayes

d. Train and Test report comparison table between Default & Tuned Model

Naive Bayes Model					
Sl.No	Index	Train Data		Test Data	
1	TN	211		112	
2	TP	675		263	
3	FN	79		40	
4	FP	96		41	
5	Accuracy	0.84		0.82	
6	AUC Score	0.84		0.82	
		Conservative	Labour	Conservative	Labour
7	Precision	0.73	0.88	0.74	0.87
8	Recall	0.69	0.90	0.73	0.87
9	F1 Score	0.71	0.89	0.73	0.87

Tunned Naive Bayes Model using SMOTE					
Sl.No	Index	Train Data		Test Data	
1	TN	628		119	
2	TP	628		249	
3	FN	126		54	
4	FP	126		34	
5	Accuracy	0.83		0.81	
6	AUC Score	0.91		0.86	
		Conservative	Labour	Conservative	Labour
7	Precision	0.83	0.83	0.69	0.88
8	Recall	0.83	0.83	0.78	0.82
9	F1 Score	0.83	0.83	0.73	0.85

Table 20: Model performance Metrices - Default Naïve Bayes & Tunned Naïve Bayes

Observation: -

- There's no drastic difference found even performing SMOTE technique and almost similar to normal Naïve Bayes Model. There is slight increase in performance in terms of AUC Score, also there is slight increase in conservative class in terms of Recall, also slight increase found in Labour class in terms of Precision & F1 score.
- In tunned model cases voters polled are 54 instances where model predicted as not polled. We observer that, in tunned model FN value increased from 40 to 54.
- In tunned model Cases voters not polled, but model predicted them to be polled are 34, We observer that, in tunned model FN value slightly decrease from 41 to 34.
- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Overall, the metrics are good fit.

Model 9 – Bagging (Random Forest)

a. Confusion Matrix

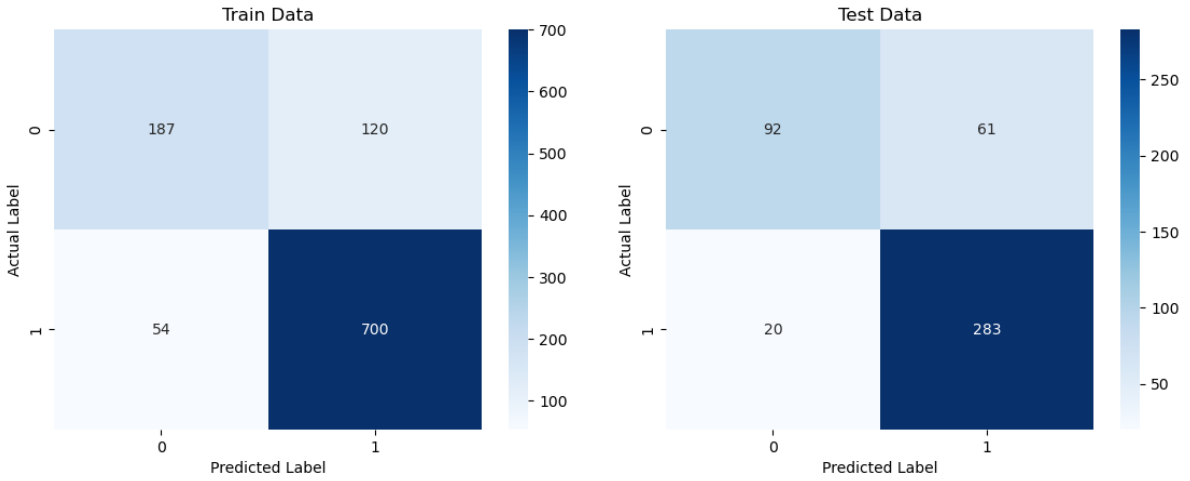


Fig 18: Confusion Matrix - Bagging

b. Classification Report

Training Data report:

	precision	recall	f1-score	support
0	0.78	0.61	0.68	307
1	0.85	0.93	0.89	754
accuracy			0.84	1061
macro avg	0.81	0.77	0.79	1061
weighted avg	0.83	0.84	0.83	1061

Testing Data report:

	precision	recall	f1-score	support
0	0.82	0.60	0.69	153
1	0.82	0.93	0.87	303
accuracy			0.82	456
macro avg	0.82	0.77	0.78	456
weighted avg	0.82	0.82	0.81	456

Table 21: Classification Report - Bagging

c. ROC Curve and AUC score

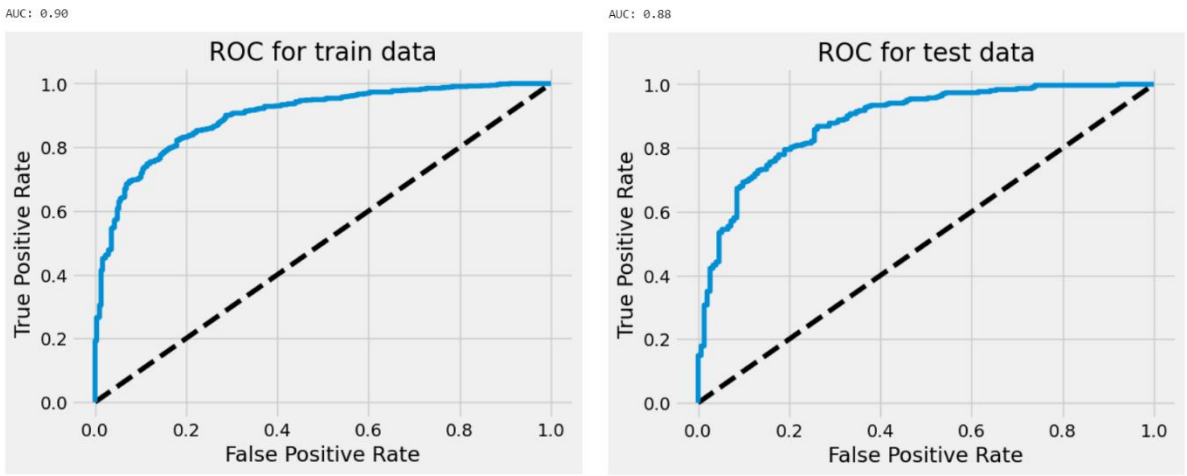


Fig 19: ROC Curve - Bagging

d. Train and Test report comparison table for Bagging (Random Forest)

Bagging (Random Forest) Model					
Sl.No	Index	Train Data		Test Data	
1	TN	187		92	
2	TP	700		283	
3	FN	54		20	
4	FP	120		61	
5	Accuracy	0.84		0.82	
6	AUC Score	0.90		0.88	
		Conservative	Labour	Conservative	Labour
7	Precision	0.78	0.85	0.82	0.82
8	Recall	0.61	0.93	0.60	0.93
9	F1 Score	0.68	0.89	0.69	0.87

Table 22: Model performance Metrices - Bagging

Observation: -

- From the analysis we can say that, model does a better job of correctly classifying the Labour Party voters then Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Bagging with Random Forest has performed exceptionally well on the train data for both the classes 0 and 1 in terms of Recall, Precision, F1 score.
- There is no drastic difference found between testing & training dataset this indicates the model is neither overfit or underfit.

Model 10 – Ada Boosting

a. Confusion Matrix

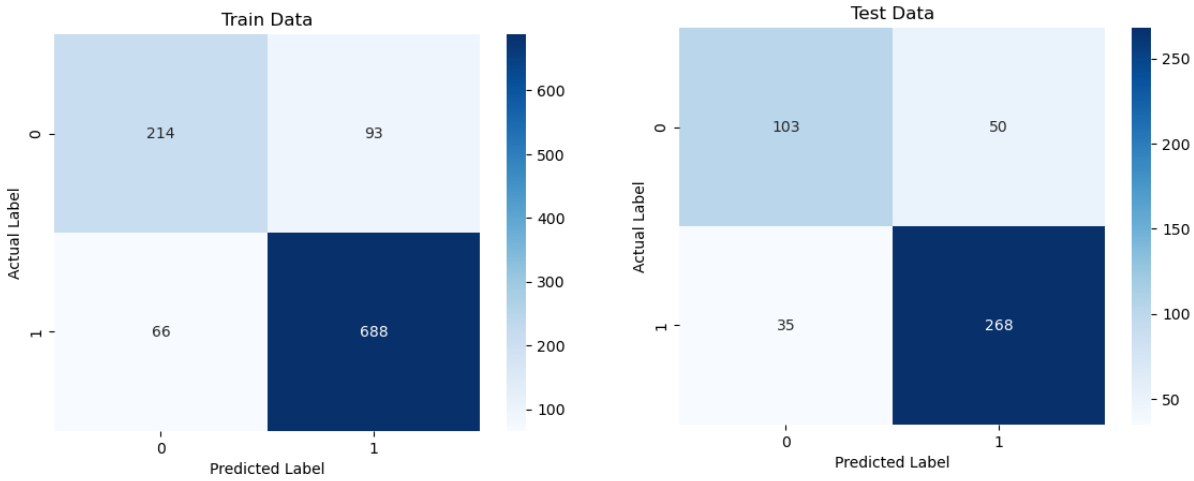


Fig 20: Confusion Matrix – Ada Boosting

b. Classification Report

Training Data report:

	precision	recall	f1-score	support
0	0.76	0.70	0.73	307
1	0.88	0.91	0.90	754
accuracy			0.85	1061
macro avg	0.82	0.80	0.81	1061
weighted avg	0.85	0.85	0.85	1061

Testing Data report:

	precision	recall	f1-score	support
0	0.75	0.67	0.71	153
1	0.84	0.88	0.86	303
accuracy			0.81	456
macro avg	0.79	0.78	0.79	456
weighted avg	0.81	0.81	0.81	456

Table 23: Classification Report - Ada Boosting

c. ROC Curve and AUC score

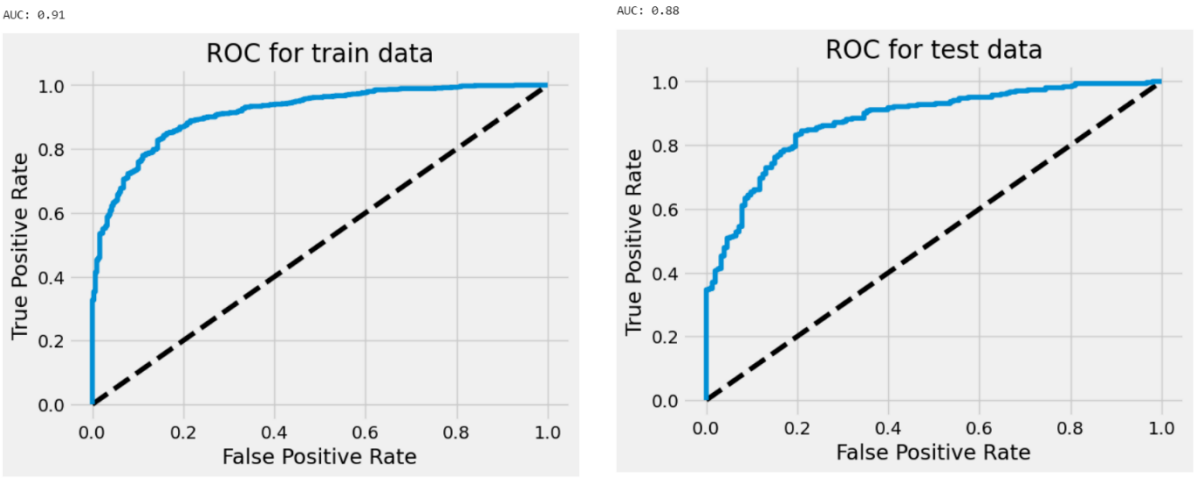


Fig 21: ROC Curve - Ada Boosting

d. Train and Test report comparison table for AdaBossting

AdaBossting Model					
Sl.No	Index	Train Data		Test Data	
1	TN	214		103	
2	TP	688		268	
3	FN	66		35	
4	FP	93		50	
5	Accuracy	0.85		0.81	
6	AUC Score	0.91		0.88	
		Conservative	Labour	Conservative	Labour
7	Precision	0.76	0.88	0.75	0.84
8	Recall	0.70	0.91	0.67	0.88
9	F1 Score	0.73	0.90	0.71	0.86

Table 24: Model performance Metrices - Ada Boosting

Observation: -

- In cases voters polled are 35 instances where model predicted as not polled.
- In Cases voters not polled, but model predicted them to be polled are 50, We observer that,
- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Overall, the metrics are good fit.

Model 11 - Gradient Boosting Model

a. Confusion Matrix

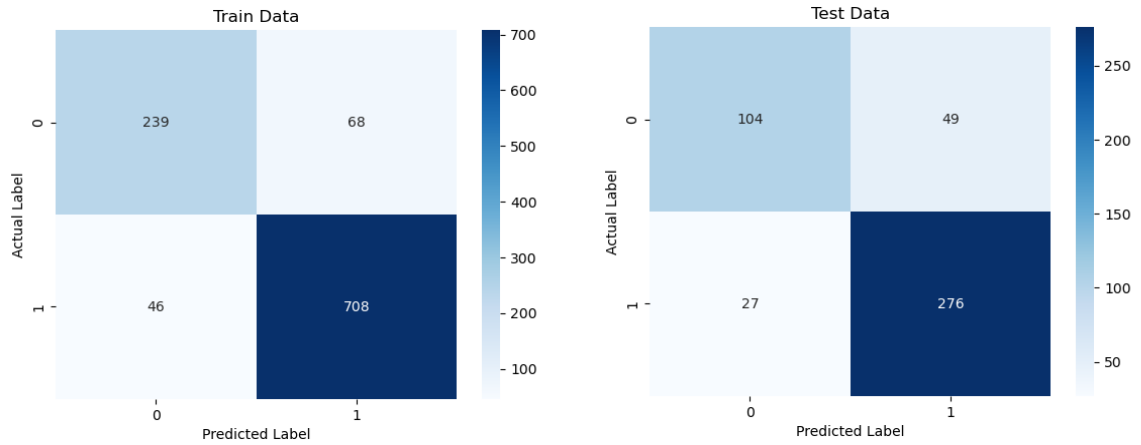


Fig 22: Confusion Matrix – Gradient Boosting

b. Classification Report

Training Data report:

	precision	recall	f1-score	support
0	0.84	0.78	0.81	307
1	0.91	0.94	0.93	754
accuracy			0.89	1061
macro avg	0.88	0.86	0.87	1061
weighted avg	0.89	0.89	0.89	1061

Testing Data report:

	precision	recall	f1-score	support
0	0.79	0.68	0.73	153
1	0.85	0.91	0.88	303
accuracy			0.83	456
macro avg	0.82	0.80	0.81	456
weighted avg	0.83	0.83	0.83	456

Table 25: Classification Report - Gradient Boosting

c. ROC Curve and AUC score

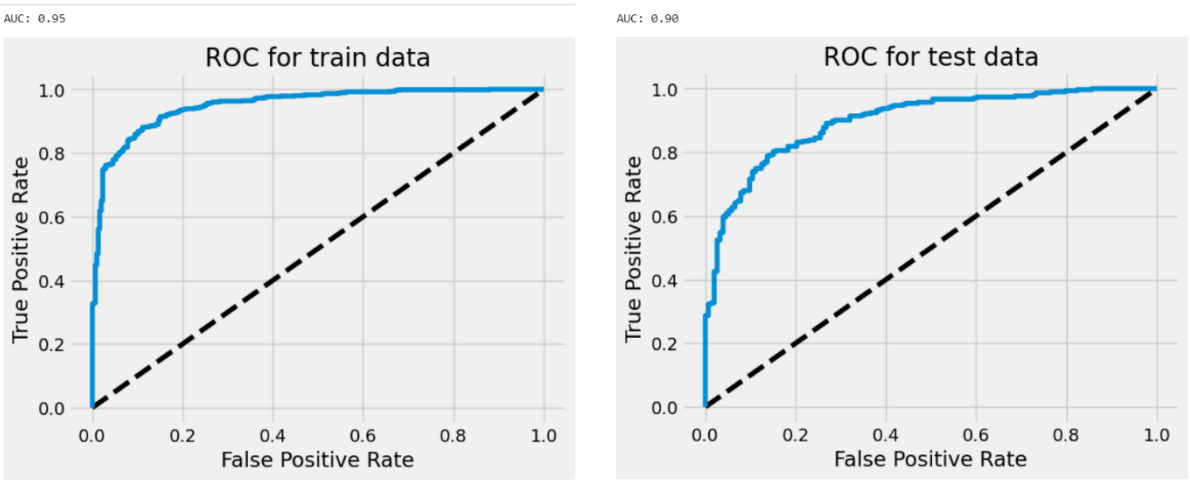


Fig 23: ROC Curve - Gradient Boosting

d. Train and Test report comparison table for AdaBossting

Gradient Bossting Model					
Sl.No	Index	Train Data		Test Data	
1	TN	239		104	
2	TP	708		276	
3	FN	46		27	
4	FP	68		49	
5	Accuracy	0.89		0.83	
6	AUC Score	0.95		0.90	
		Conservative	Labour	Conservative	Labour
7	Precision	0.84	0.91	0.79	0.85
8	Recall	0.78	0.94	0.68	0.91
9	F1 Score	0.81	0.93	0.73	0.88

Table 26: Model performance Metrices - Gradient Boosting

Observation: -

- In cases voters polled are 27 instances where model predicted as not polled.
- In Cases voters not polled, but model predicted them to be polled are 49, We observer that,
- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.
- Overall, the metrics are good fit.

Comparison of the performance metrics for the 11 models

Conservative Party (Class 0)

	Recall	Precision	F1 Score	Accuracy	AUC
LR Train	0.75	0.65	0.70	0.84	0.89
LR Test	0.75	0.72	0.74	0.83	0.88
LR_Tuned_Train	0.76	0.64	0.70	0.84	0.89
LR_Tuned_Test	0.76	0.72	0.74	0.83	0.88
LDA Train	0.74	0.65	0.69	0.83	0.89
LDA Test	0.74	0.65	0.69	0.83	0.89
LDA_Tuned_Train	0.80	0.57	0.67	0.83	0.76
LDA_Tuned_Test	0.80	0.57	0.67	0.83	0.79
KNN Train	0.77	0.72	0.74	0.86	0.93
KNN_Test	0.75	0.70	0.72	0.82	0.87
KNN_Tuned_Train	0.75	0.65	0.70	0.84	0.90
KNN_Tunned_Test	0.81	0.69	0.75	0.84	0.89
NB Train	0.73	0.69	0.71	0.84	0.89
NB Test	0.74	0.73	0.73	0.82	0.88
NB_Smote_Train	0.83	0.83	0.83	0.83	0.91
NB_Smote_Test	0.69	0.78	0.73	0.81	0.86
BAGGING Train	0.78	0.61	0.68	0.84	0.90
BAGGING Test	0.82	0.60	0.69	0.82	0.88
ADA Train	0.76	0.70	0.73	0.85	0.91
ADA Test	0.75	0.67	0.71	0.81	0.88
Gradient Train	0.84	0.78	0.81	0.89	0.95
Gradient Test	0.79	0.68	0.73	0.83	0.90

Labour Party (Class 1)

	Recall	Precision	F1 Score	Accuracy	AUC
LR Train	0.86	0.91	0.89	0.84	0.89
LR Test	0.86	0.88	0.87	0.83	0.88
LR_Tuned_Train	0.86	0.92	0.89	0.84	0.89
LR_Tuned_Test	0.86	0.88	0.87	0.83	0.88
LDA Train	0.86	0.91	0.89	0.83	0.89
LDA Test	0.86	0.91	0.89	0.83	0.89
LDA_Tuned_Train	0.84	0.94	0.89	0.83	0.76
LDA_Tuned_Test	0.84	0.94	0.89	0.83	0.79
KNN Train	0.89	0.91	0.90	0.86	0.93
KNN_Test	0.85	0.88	0.87	0.82	0.87
KNN_Tuned_Train	0.87	0.91	0.89	0.84	0.90
KNN_Tunned_Test	0.86	0.92	0.89	0.84	0.89
NB Train	0.88	0.90	0.89	0.84	0.89
NB Test	0.87	0.87	0.87	0.82	0.88
NB_Smote_Train	0.83	0.83	0.83	0.83	0.91
NB_Smote_Test	0.88	0.82	0.85	0.81	0.86
BAGGING Train	0.85	0.93	0.89	0.84	0.90
BAGGING Test	0.82	0.93	0.87	0.82	0.88
ADA Train	0.88	0.91	0.90	0.85	0.91
ADA Test	0.84	0.88	0.86	0.81	0.88
Gradient Train	0.91	0.94	0.93	0.89	0.95
Gradient Test	0.85	0.91	0.88	0.83	0.90

Table 35: Performance comparison metrics for the 11 models

Inferences on Comparison of Model performance:

- Here we built 11 models namely Logistic Regression, Logistic Tunned Regression, Linear Discriminative Analysis, Linear Tunned Discriminative Analysis, KNN Model, KNN Tunned model, Naïve Bayes, Naïve Bayes using SMOTE, Bagging Model, AdaBoosting & Gradient Model. We have checked the Performance metrices.
- Problem statement, states that “Need 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.
- As a result, we can conclude that, Precision/ Recall will be more crucial i.e., both type I and type II error are equally important. Hence 'f1 score' is considered to predict the problem and considered as most important performance metric here. (Note: here we are considering 'f1 score ' because it calculates the non-weighted average of minority and majority class).
- All the 11 model gives best result and performed reasonably stable enough to be used for making any future predictions. The train and test values aren't that so far for all the 11 modules, thus there seems to be no concern in overfitting or underfitting.
- Comparing the F1 Score, Recall, Precision, Accuracy & ACC among 11 models, KNN Tunned Model gives stable result both in class 0 (Conservative Party) & class 1 (Labour Party)

- From the above interpretation we conclude that tunned K - Nearest Neighbour gives stable values in terms of Accuracy, AUC score, Precision, Recall and F1 score. From this we conclude tunned K - Nearest Neighbour seems to be optimised model.

Final Model: Tunned K-Nearest Neighbor Model

Conservative Party (Class 0)

	KNN_Tuned_Train	KNN_Tunned_Test
Recall	0.75	0.81
Precision	0.65	0.69
F1 Score	0.70	0.75
Accuracy	0.84	0.84
AUC	0.90	0.89

Labour Party (Class 1)

	KNN_Tuned_test	KNN_Tunned_Test
Recall	0.86	0.86
Precision	0.92	0.92
F1 Score	0.89	0.89
Accuracy	0.84	0.84
AUC	0.89	0.89

Inference: -

Class 0 (Conservative Party)

- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- In test data the recall value is 0.81, Precision value is 0.69, F1 score is 0.75, the Accuracy score is 0.84 & AUC score is 0.89.

Class 1 (Labour Party)

- In both train and test data an accuracy, precision and recall not much difference found and they are nearly identical to each other. This shows that, model is neither overfitting nor underfitting.
- In test data the recall value is 0.86, Precision value is 0.92, F1 score is 0.89, the Accuracy score is 0.84 & AUC score is 0.89.

From the analysis it can be said that the model does a better job in correctly classifying the Labour Party voters than the Conservative voters. The high precisions, recall and f1 scores indicate that overall, the model seems to classify the respondents well.

Here we conclude that, Tunned KNN Model is best optimized model to create an exit poll for the news channel CNBE that will aid in predicting overall win and seats covered by a particular political party: "Conservative" or "Labour."

1.8 Based on these predictions, what are the insights?

Business Insights:

An election exit poll is a poll of voters taken immediately after they have exited the polling stations. In order to anticipate the election outcome, voters are asked who they voted for? Exit polls are also used to gather demographic information on voters and to learn the reason for the vote towards one of the parties. Because real votes are cast anonymously and polling is the only way to gather this data. Politicians mostly rely on public opinion data to determine their positions on numerous political matters. Political parties build election rallies and campaigns on this data, which highlights certain significant social, economic, and cultural feelings among the general people.

- Sample size is considered as an important aspect for the research. Larger sample size can provide more accurate mean values and identifying outliers that could skew the data in a smaller sample and provide a smaller margin of error that helps predict accurately overall win and seats covered by a particular party.
- The attributes 'Hague' and 'Blair' important features in predicting the dependent Variable.
- Almost all of the developed models giving better classification on Labour Party voters than Conservative Party voters.
- The parties should keep a close eye on public's perception of European integration and educate the public about their perspective. This appears to have a significant effect in their decision to vote for one of the two parties.
- As per people survey, the Labour Party leader seem to have positive ratings.

Recommendations:

1. **To Predict which party appears to have a better chance of winning:**

If your using exit poll to effectively predict the outcome of a current election, a model with a better accuracy score should be considered. This will aid in properly predicting results 90% of the time, representing the genuine circumstance in real elections.

2. **To Build a new election campaign:**

The key objective of the business challenge is to create a new marketing campaign for a political party, it is essential to understand what are the sentiments of 80% of the population. In this situation, a model that accurately predicts on either class 0 or class 1 will be able to effectively help detect the masses' sentiments. As a result, any model with a better F1 score in either the Labour or Conservative parties can be considered.

3. **To Find out whether fraudulent activities going on:**

Election booths are frequently manipulated to obtain a specific result. To detect fraudulent behaviour within an area, exit polls might be conducted across several regions or seats. If the exit poll results differ from the results of the real elections, it is possible to pinpoint the source of the fraud, resulting in re-elections. In this situation, a model with more than 90% scores on either class 1 or class 0 on Recall or Precision can be considered. Precision is a measure of the relevancy of the results, whereas recall is a measure of the number of actually relevant results returned.

4. **To determine how certain political campaign can influence the success or failure rate:**

A successful campaign is one that causes the general public's preference to shift from one political party to another. In this situation, an exit poll will be done to ascertain the public's general feelings. The campaigning activity will then be carefully carried out in order to influence the voters' mindsets and ideas who voted for the other party. The next exit poll will be performed to see if people who previously voted for the opposing party have now switched their votes to the concerned party. This will aid in determining the campaign's effectiveness. The model with the greatest score for either class 0 or class 1 may be used to predict whether or not a voter would vote for the particular party.

Speeches of the Presidents Analysis

Table of Contents

Table List:

Figure List:

Problem 2:

In this particular project, we are going to work on the inaugural corpora from the nltk in Python. 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

2.1 Find the number of characters, words, and sentences for the mentioned documents

After importing all the necessary libraries, we download the inaugural data from NLTK corpus. There are multiple speeches given by various leaders and this can be seen by calling. fileids( ) function.

For our analysis, we are going to focus on the following three speeches:
1. 1941-Roosevelt.txt
2. 1961-Kennedy.txt
3. 1973-Nixon.txt

We can use the. raw () function to depict the raw file before processing

Snapshots of all the raw files are given below:

1. 1941 - Roosevelt's Speech:


```
'On each national day of inauguration since 1789, the people have renewed their sense of dedication to the United States.\n\nIn
Washington\'s day the task of the people was to create and weld together a nation.\n\nIn Lincoln\'s day the task of the people
was to preserve that Nation from disruption from within.\n\nIn this day the task of the people is to save that Nation and its i
nstitutions from disruption from without.\n\nTo us there has come a time, in the midst of swift happenings, to pause for a mome
nt and take stock -- to recall what our place in history has been, and to rediscover what we are and what we may be. If we do n
ot, we risk the real peril of inaction.\n\nLives of nations are determined not by the count of years, but by the lifetime of th
e human spirit. The life of a man is three-score years and ten: a little more, a little less. The life of a nation is the fulln
ess of the measure of its will to live.\n\nThere are men who doubt this. There are men who believe that democracy, as a form of
Government and a frame of life, is limited or measured by a kind of mystical and artificial fate that, for some unexplained rea
son, tyranny and slavery have become the surging wave of the future -- and that freedom is an ebbing tide.\n\nBut we Americans
know that this is not true.\n\nEight years ago, when the life of this Republic seemed frozen by a fatalistic terror, we proved
that this is not true. We were in the midst of shock -- but we acted. We acted quickly, boldly, decisively.\n\nThese later year
s have been living years -- fruitful years for the people of this democracy. For they have brought to us greater security and,
I hope, a better understanding that life\'s ideals are to be measured in other than material things.\n\nMost vital to our prese
nt and our future is this experience of a democracy which successfully survived crisis at home; put away many evil things; buil
t new structures on enduring lines; and, through it all, maintained the fact of its democracy.\n\nFor action has been taken wit
hin the three-way framework of the Constitution of the United States. The coordinate branches of the Government continue freely
to function. The Bill of Rights remains inviolate. The freedom of elections is wholly maintained. Prophets of the downfall of A
merican democracy have seen their dire predictions come to naught.\n\nDemocracy is not dying.\n\nWe know it because we have see
n it revive--and grow.\n\nWe know it cannot die -- because it is built on the unhampered initiative of individual men and women
joined together in a common enterprise -- an enterprise undertaken and carried through by the free expression of a free majorit
y.\n\nWe know it because democracy alone, of all forms of government, enlists the full force of men\'s enlightened will.\n\nWe
know it because democracy alone has constructed an unlimited civilization capable of infinite progress in the improvement of hu
man life.\n\nWe know it because, if we look below the surface, we sense it still spreading on every continent -- for it is the
most humane, the most advanced, and in the end the most unconquerable of all forms of human society.\n\nA nation, like a perso
n, has a body--a body that must be fed and clothed and housed, invigorated and rested, in a manner that measures up to the obje
ctives of our time.\n\nA nation, like a person, has a mind -- a mind that must be kept informed and alert, that must know itsel
f, that understands the hopes and the needs of its neighbors -- all the other nations that live within the narrowing circle of
the world.\n\nAnd a nation, like a person, has something deeper, something more permanent, something larger than the sum of all
its parts. It is that something which matters most to its future -- which calls forth the most sacred guarding of its presen
t.\n\nIt is a thing for which we find it difficult -- even impossible -- to hit upon a single, simple word.\n\nAnd yet we all u
nderstand what it is -- the spirit -- the faith of America. It is the product of centuries. It was born in the multitudes of th
ose who came from many lands -- some of high degree, but mostly plain people, who sought here, early and late, to find freedom
more freely.\n\nThe democratic aspiration is no mere recent phase in human history. It is human history. It permeated the ancie
nt life of early peoples. It blazed anew in the middle ages. It was written in Magna Charta.\n\nIn the Americas its impact has
been irresistible. America has been the New World in all tongues, to all peoples, not because this continent was a new-found la
nd, but because all those who came here believed they could create upon this continent a new life -- a life that should be new
in freedom.\n\nIts vitality was written into our own Mayflower Compact, into the Declaration of Independence, into the Constitu
tion of the United States, into the Gettysburg Address.\n\nThose who first came here to carry out the longings of their spirit,
and the millions who followed, and the stock that sprang from them -- all have moved forward constantly and consistently toward
an ideal which in itself has gained stature and clarity with each generation.\n\nThe hopes of the Republic cannot forever toler
ate either undeserved poverty or self-serving wealth.\n\nWe know that we still have far to go; that we must more greatly build
the security and the opportunity and the knowledge of every citizen, in the measure justified by the resources and the capacity
of the land.\n\nBut it is not enough to achieve these purposes alone. It is not enough to clothe and feed the body of this Nati
on, and instruct and inform its mind. For there is also the spirit. And of the three, the greatest is the spirit.\n\nWithout th
e body and the mind, as all men know, the Nation could not live.\n\nBut if the spirit of America were killed, even though the N
ation\'s body and mind, constricted in an alien world, lived on, the America we know would have perished.\n\nThat spirit -- tha
t faith -- speaks to us in our daily lives in ways often unnoticed, because they seem so obvious. It speaks to us here in the C
apital of the Nation. It speaks to us through the processes of governing in the sovereignties of 48 States. It speaks to us in
our counties, in our cities, in our towns, and in our villages. It speaks to us from the other nations of the hemisphere, and f
rom those across the seas -- the enslaved, as well as the free. Sometimes we fail to hear or heed these voices of freedom becau
se to us the privilege of our freedom is such an old, old story.\n\nThe destiny of America was proclaimed in words of prophecy
spoken by our first President in his first inaugural in 1789 -- words almost directed, it would seem, to this year of 1941: "Th
e preservation of the sacred fire of liberty and the destiny of the republican model of government are justly considered deepl
y, finally, staked on the experiment intrusted to the hands of the American people."\n\nIf we lose that sacred fire--if we let
it be smothered with doubt and fear -- then we shall reject the destiny which Washington strove so valiantly and so triumphantl
y to establish. The preservation of the spirit and faith of the Nation does, and will, furnish the highest justification for ev
ery sacrifice that we may make in the cause of national defense.\n\nIn the face of great perils never before encountered, our s
trong purpose is to protect and to perpetuate the integrity of democracy.\n\nFor this we muster the spirit of America, and the
faith of America.\n\nWe do not retreat. We are not content to stand still. As Americans, we go forward, in the service of our c
ountry, by the will of God.\n'
```


Fig 1: Snapshot of 1941- Roosevelt's speech

2. 1961 – Kennedy's Speech:


```
'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 a
s a beginning -- signifying renewal, as well as change. For I have sworn I before you and Almighty God the same solemn oath our
forebears l prescribed nearly a century and three quarters ago.\n\nThe world is very different now. For man holds in his mortal
hands the power to abolish all forms of human poverty and all forms of human life. And yet the same revolutionary beliefs for w
hich our forebears fought are still at issue around the globe -- the belief that the rights of man come not from the generosity
of the state, but from the hand of God.\n\nWe dare not forget today that we are the heirs of that first revolution. Let the wor
d go forth from this time and place, to friend and foe alike, that the torch has been passed to a new generation of Americans -
- born in this century, tempered by war, disciplined by a hard and bitter peace, proud of our ancient heritage -- and unwilling
to witness or permit the slow undoing of those human rights to which this Nation has always been committed, and to which we are
committed today at home and around the world.\n\nLet every nation know, whether it wishes us well or ill, that we shall pay any
price, bear any burden, meet any hardship, support any friend, oppose any foe, in order to assure the survival and the success
of liberty.\n\nThis much we pledge -- and more.\n\nTo those old allies whose cultural and spiritual origins we share, we pledge
the loyalty of faithful friends. United, there is little we cannot do in a host of cooperative ventures. Divided, there is litt
le we can do -- for we dare not meet a powerful challenge at odds and split asunder.\n\nTo those new States whom we welcome to
the ranks of the free, we pledge our word that one form of colonial control shall not have passed away merely to be replaced by
a far more iron tyranny. We shall not always expect to find them supporting our view. But we shall always hope to find them str
ongly supporting their own freedom -- and to remember that, in the past, those who foolishly sought power by riding the back of
the tiger ended up inside.\n\nTo those peoples in the huts and villages across the globe struggling to break the bonds of mass
misery, we pledge our best efforts to help them help themselves, for whatever period is required -- not because the Communists
may be doing it, not because we seek their votes, but because it is right. If a free society cannot help the many who are poor,
it cannot save the few who are rich.\n\nTo our sister republics south of our border, we offer a special pledge -- to convert ou
r good words into good deeds -- in a new alliance for progress -- to assist free men and free governments in casting off the ch
ains of poverty. But this peaceful revolution of hope cannot become the prey of hostile powers. Let all our neighbors know that
we shall join with them to oppose aggression or subversion anywhere in the Americas. And let every other power know that this H
emisphere intends to remain the master of its own house.\n\nTo that world assembly of sovereign states, the United Nations, our
last best hope in an age where the instruments of war have far outpaced the instruments of peace, we renew our pledge of suppor
t--to prevent it from becoming merely a forum for invective -- to strengthen its shield of the new and the weak -- and to enlar
ge the area in which its writ may run.\n\nFinally, to those nations who would make themselves our adversary, we offer not a ple
dge but a request: that both sides begin anew the quest for peace, before the dark powers of destruction unleashed by science e
ngulf all humanity in planned or accidental self-destruction.\n\nWe dare not tempt them with weakness. For only when our arms a
re sufficient beyond doubt can we be certain beyond doubt that they will never be employed.\n\nBut neither can two great and po
werful groups of nations take comfort from our present course -- both sides overburdened by the cost of modern weapons, both ri
ghtly alarmed by the steady spread of the deadly atom, yet both racing to alter that uncertain balance of terror that stays the
hand of mankind\'s final war.\n\nSo let us begin anew -- remembering on both sides that civility is not a sign of weakness, and
sincerity is always subject to proof. Let us never negotiate out of fear. But let us never fear to negotiate.\n\nLet both sides
explore what problems unite us instead of belaboring those problems which divide us.\n\nLet both sides, for the first time, for
mulate serious and precise proposals for the inspection and control of arms -- and bring the absolute power to destroy other na
tions under the absolute control of all nations.\n\nLet both sides seek to invoke the wonders of science instead of its terror
s. Together let us explore the stars, conquer the deserts, eradicate disease, tap the ocean depths, and encourage the arts and
commerce.\n\nLet both sides unite to heed in all corners of the earth the command of Isaiah -- to "undo the heavy burdens ... a
nd to let the oppressed go free."\n\nAnd if a beachhead of cooperation may push back the jungle of suspicion, let both sides jo
in in creating a new endeavor, not a new balance of power, but a new world of law, where the strong are just and the weak secur
e and the peace preserved.\n\nAll this will not be finished in the first 100 days. Nor will it be finished in the first 1,000 d
ays, nor in the life of this Administration, nor even perhaps in our lifetime on this planet. But let us begin.\n\nIn your hand
s, my fellow citizens, more than in mine, will rest the final success or failure of our course. Since this country was founded,
each generation of Americans has been summoned to give testimony to its national loyalty. The graves of young Americans who ans
wered the call to service surround the globe.\n\nNow the trumpet summons us again -- not as a call to bear arms, though arms we
need; not as a call to battle, though embattled we are -- but a call to bear the burden of a long twilight struggle, year in an
d year out, "rejoicing in hope, patient in tribulation" -- a struggle against the common enemies of man: tyranny, poverty, dise
ase, and war itself.\n\nCan we forge against these enemies a grand and global alliance, North and South, East and West, that ca
n assure a more fruitful life for all mankind? Will you join in that historic effort?\n\nIn the long history of the world, only
a few generations have been granted the role of defending freedom in its hour of maximum danger. I do not shrink from this resp
onsibility -- I welcome it. I do not believe that any of us would exchange places with any other people or any other generatio
n. The energy, the faith, the devotion which we bring to this endeavor will light our country and all who serve it -- and the g
low from that fire can truly light the world.\n\nAnd so, my fellow Americans: ask not what your country can do for you -- ask w
hat you can do for your country.\n\nMy fellow citizens of the world: ask not what America will do for you, but what together we
can do for the freedom of man.\n\nFinally, whether you are citizens of America or citizens of the world, ask of us the same hig
h standards of strength and sacrifice which we ask of you. With a good conscience our only sure reward, with history the final
judge of our deeds, let us go forth to lead the land we love, asking His blessing and His help, but knowing that here on earth
God\'s work must truly be our own.\n'
```


Fig 2: Snapshot of 1961-Kennedy's speech

3. 1973 – Nixon's Speech

'Mr. Vice President, Mr. Speaker, Mr. Chief Justice, Senator Cook, Mrs. Eisenhower, and my fellow citizens of this great and go
od country we share together:\n\nWhen we met here four years ago, America was bleak in spirit, depressed by the prospect of see
mingly endless war abroad and of destructive conflict at home.\n\nAs we meet here today, we stand on the threshold of a new era
of peace in the world.\n\nThe central question before us is: How shall we use that peace? Let us resolve that this era we are a
bout to enter will not be what other postwar periods have so often been: a time of retreat and isolation that leads to stagnati
on at home and invites new danger abroad.\n\nLet us resolve that this will be what it can become: a time of great responsibilit
ies greatly borne, in which we renew the spirit and the promise of America as we enter our third century as a nation.\n\nThis p
ast year saw far-reaching results from our new policies for peace. By continuing to revitalize our traditional friendships, and
by our missions to Peking and to Moscow, we were able to establish the base for a new and more durable pattern of relationships
among the nations of the world. Because of America\'s bold initiatives, 1972 will be long remembered as the year of the greates
t progress since the end of World War II toward a lasting peace in the world.\n\nThe peace we seek in the world is not the flim
sy peace which is merely an interlude between wars, but a peace which can endure for generations to come.\n\nIt is important th
at we understand both the necessity and the limitations of America\'s role in maintaining that peace.\n\nUnless we in America w
ork to preserve the peace, there will be no peace.\n\nUnless we in America work to preserve freedom, there will be no freedo
m.\n\nBut let us clearly understand the new nature of America\'s role, as a result of the new policies we have adopted over the
se past four years.\n\nWe shall respect our treaty commitments.\n\nWe shall support vigorously the principle that no country ha
s the right to impose its will or rule on another by force.\n\nWe shall continue, in this era of negotiation, to work for the l
imitation of nuclear arms, and to reduce the danger of confrontation between the great powers.\n\nWe shall do our share in defe
nding peace and freedom in the world. But we shall expect others to do their share.\n\nThe time has passed when America will ma
ke every other nation\'s conflict our own, or make every other nation\'s future our responsibility, or presume to tell the peop
le of other nations how to manage their own affairs.\n\nJust as we respect the right of each nation to determine its own futur
e, we also recognize the responsibility of each nation to secure its own future.\n\nJust as America\'s role is indispensable in
preserving the world\'s peace, so is each nation\'s role indispensable in preserving its own peace.\n\nTogether with the rest o
f the world, let us resolve to move forward from the beginnings we have made. Let us continue to bring down the walls of hostil
ity which have divided the world for too long, and to build in their place bridges of understanding -- so that despite profound
differences between systems of government, the people of the world can be friends.\n\nLet us build a structure of peace in the
world in which the weak are as safe as the strong -- in which each respects the right of the other to live by a different syste
m -- in which those who would influence others will do so by the strength of their ideas, and not by the force of their arms.\n
\nLet us accept that high responsibility not as a burden, but gladly -- gladly because the chance to build such a peace is the
noblest endeavor in which a nation can engage; gladly, also, because only if we act greatly in meeting our responsibilities abr
oad will we remain a great Nation, and only if we remain a great Nation will we act greatly in meeting our challenges at hom
e.\n\nWe have the chance today to do more than ever before in our history to make life better in America -- to ensure better ed
ucation, better health, better housing, better transportation, a cleaner environment -- to restore respect for law, to make our
communities more livable -- and to insure the God-given right of every American to full and equal opportunity.\n\nBecause the r
ange of our needs is so great -- because the reach of our opportunities is so great -- let us be bold in our determination to m
eet those needs in new ways.\n\nJust as building a structure of peace abroad has required turning away from old policies that f
ailed, so building a new era of progress at home requires turning away from old policies that have failed.\n\nAbroad, the shift
from old policies to new has not been a retreat from our responsibilities, but a better way to peace.\n\nAnd at home, the shift
from old policies to new will not be a retreat from our responsibilities, but a better way to progress.\n\nAbroad and at home,
the key to those new responsibilities lies in the placing and the division of responsibility. We have lived too long with the c
onsequences of attempting to gather all power and responsibility in Washington.\n\nAbroad and at home, the time has come to tur
n away from the condescending policies of paternalism -- of "Washington knows best."\n\nA person can be expected to act respons
ibly only if he has responsibility. This is human nature. So let us encourage individuals at home and nations abroad to do more
for themselves, to decide more for themselves. Let us locate responsibility in more places. Let us measure what we will do for
others by what they will do for themselves.\n\nThat is why today I offer no promise of a purely governmental solution for every
problem. We have lived too long with that false promise. In trusting too much in government, we have asked of it more than it c
an deliver. This leads only to inflated expectations, to reduced individual effort, and to a disappointment and frustration tha
t erode confidence both in what government can do and in what people can do.\n\nGovernment must learn to take less from people
so that people an do more for themselves.\n\nLet us remember that America was built not by government, but by people -- not by
welfare, but by work -- not by shirking responsibility, but by seeking responsibility.\n\nIn our own lives, let each of us ask
-- not just what will government do for me, but what can I do for myself?\n\nIn the challenges we face together, let each of us
ask -- not just how can government help, but how can I help?\n\nYour National Government has a great and vital role to play. An
d I pledge to you that where this Government should act, we will act boldly and we will lead boldly. But just as important is t
he role that each and every one of us must play, as an individual and as a member of his own community.\n\nFrom this day forwar
d, let each of us make a solemn commitment in his own heart: to bear his responsibility, to do his part, to live his ideals --
so that together, we can see the dawn of a new age of progress for America, and together, as we celebrate our 200th anniversary
as a nation, we can do so proud in the fulfillment of our promise to ourselves and to the world.\n\nAs America\'s longest and m
ost difficult war comes to an end, let us again learn to debate our differences with civility and decency. And let each of us r
each out for that one precious quality government cannot provide -- a new level of respect for the rights and feelings of one a
nother, a new level of respect for the individual human dignity which is the cherished birthright of every American.\n\nAbove a
ll else, the time has come for us to renew our faith in ourselves and in America.\n\nIn recent years, that faith has been chall
enged.\n\nOur children have been taught to be ashamed of their country, ashamed of their parents, ashamed of America\'s record
at home and of its role in the world.\n\nAt every turn, we have been beset by those who find everything wrong with America and
little that is right. But I am confident that this will not be the judgment of history on these remarkable times in which we ar
e privileged to live.\n\nAmerica\'s record in this century has been unparalleled in the world\'s history for its responsibilit
y, for its generosity, for its creativity and for its progress.\n\nLet us be proud that our system has produced and provided mo
re freedom and more abundance, more widely shared, than any other system in the history of the world.\n\nLet us be proud that i
n each of the four wars in which we have been engaged in this century, including the one we are now bringing to an end, we have
fought not for our selfish advantage, but to help others resist aggression.\n\nLet us be proud that by our bold, new initiative
s, and by our steadfastness for peace with honor, we have made a break-through toward creating in the world what the world has
not known before -- a structure of peace that can last, not merely for our time, but for generations to come.\n\nWe are embarki
ng here today on an era that presents challenges great as those any nation, or any generation, has ever faced.\n\nWe shall answ
er to God, to history, and to our conscience for the way in which we use these years.\n\nAs I stand in this place, so hallowed
by history, I think of others who have stood here before me. I think of the dreams they had for America, and I think of how eac
h recognized that he needed help far beyond himself in order to make those dreams come true.\n\nToday, I ask your prayers that
in the years ahead I may have God\'s help in making decisions that are right for America, and I pray for your help so that toge
ther we may be worthy of our challenge.\n\nLet us pledge together to make these next four years the best four years in America
\'s history, so that on its 200th birthday America will be as young and as vital as when it began, and as bright a beacon of ho
pe for all the world.\n\nLet us go forward from here confident in hope, strong in our faith in one another, sustained by our fa
ith in God who created us, and striving always to serve His purpose.\n'

Fig 3: Snapshot of 1973 – Nixon's Speech

1. **Checking the Number of Characters in each speech by using len (inaugural.raw()) function:**

```
a. Number of characters in Roosevelt file: 7571
b. Number of characters in Kennedy file: 7618
c. Number of characters in Nixon file: 9991
```

2. **Checking the Number of Words in each speech by using list (inaugural.words()) in len function on each speech:**

```
a. Number of words in Roosevelt file: 1536
b. Number of words in Kennedy file: 1546
c. Number of words in Nixon file: 2028
```

3. **Check the Number of Sentences in each speech by using list (inaugural.sents()) in len function on each speech:**

```
a. Number of sentences in Roosevelt file: 68
b. Number of sentences in Kennedy file: 52
c. Number of sentences in Nixon file: 69
```

2.2) 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.

StopWords

The process of converting data to something a computer can understand is referred to as pre-processing. One of the major forms of pre-processing is to filter out useless data. In natural language processing, useless words (data), are referred to as stop words.

A stop word is a commonly used word (such as “the”, “a”, “an”, “in”) that a search engine has been programmed to ignore, both when indexing entries for searching and when retrieving them as the result of a search query.

We would not want these words to take up space in our database, or taking up valuable processing time. For this, we can remove them easily, by storing a list of words that you consider to stop words. NLTK(Natural Language Toolkit) in python has a list of stopwords stored in 16 different languages. You can find them in the nltk_data directory using the command (R_number_of_words).

Following are the list of stopwords in NLTK directory:

```
['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've", "you'
ll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his', 'himself', 'she',
"she's", 'her', 'hers', 'herself', 'it', "it's", 'its', 'itself', 'they', 'them', 'their', 'th
eirs', 'themselves', 'what', 'which', 'who', 'whom', 'this', 'that', "that'll", 'these', 'thos
e', 'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'have', 'has', 'had', 'having', '
do', 'does', 'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or', 'because', 'as', 'unt
il', 'while', 'of', 'at', 'by', 'for', 'with', 'about', 'against', 'between', 'into', 'through
', 'during', 'before', 'after', 'above', 'below', 'to', 'from', 'up', 'down', 'in', 'out', 'on
', 'off', 'over', 'under', 'again', 'further', 'then', 'once', 'here', 'there', 'when', 'where
', 'why', 'how', 'all', 'any', 'both', 'each', 'few', 'more', 'most', 'other', 'some', 'such',
'no', 'nor', 'not', 'only', 'own', 'same', 'so', 'than', 'too', 'very', 's', 't', 'can', 'will
', 'just', 'don', "don't", 'should', "should've", 'now', 'd', 'll', 'm', 'o', 're', 've', 'y',
'ain', 'aren', "aren't", 'couldn', "couldn't", 'didn', "didn't", 'doesn', "doesn't", 'hadn', "
hadn't", 'hasn', "hasn't", 'haven', "haven't", 'isn', "isn't", 'ma', 'mightn', "mightn't", 'mu
stn', "mustn't", 'needn', "needn't", 'shan', "shan't", 'shouldn', "shouldn't", 'wasn', "wasn't
", 'weren', "weren't", 'won', "won't", 'wouldn', "wouldn't"]+ Punctuation
```

Note: You can even modify the list by adding words of your choice in the english .txt. file in the stopwords directory.

EDA Steps for Speech 1-('1941-Roosevelt.txt')

The word count before removal of stopwords for Roosevelt's Speech is 1536 words.

1. Remove punctuation and special character like (--)

	Speech1
0	On each national day of inauguration since 178...
1	In Washington's day the task of the people was...
2	In Lincoln's day the task of the people was to...
3	In this day the task of the people is to save ...
4	To us there has come a time, in the midst of s...
5	If we do not, we risk the real peril of inaction.
6	Lives of nations are determined not by the cou...
7	The life of a man is three-score years and ten...
8	The life of a nation is the fullness of the me...
9	There are men who doubt this.

	Speech1
0	on each national day of inauguration since 178...
1	in washingtons day the task of the people was ...
2	in lincolns day the task of the people was to ...
3	in this day the task of the people is to save ...
4	to us there has come a time in the midst of sw...
5	if we do not we risk the real peril of inaction
6	lives of nations are determined not by the cou...
7	the life of a man is threescore years and ten ...
8	the life of a nation is the fullness of the me...
9	there are men who doubt this

Table 1: Remove punctuation Speech1

2. Convert the Speech into Lower Case Conversion and check the Data frame of Speech 1

	Speech1
0	On each national day of inauguration since 178...
1	In Washingtons day the task of the people was ...
2	In Lincolns day the task of the people was to ...
3	In this day the task of the people is to save ...
4	To us there has come a time in the midst of sw...

	Speech1
0	on each national day of inauguration since 178...
1	in washingtons day the task of the people was ...
2	in lincolns day the task of the people was to ...
3	in this day the task of the people is to save ...
4	to us there has come a time in the midst of sw...

Table 2: Lower Case Conversion Speech1

3. Checking of Stopwords in Each Sentence of Speech 1

Number of the Stopwords Present in Roosevelt file is 711

4. Stopwords Count Comparison Before and After

	Speech1	stopwords	Speech_1_after_remove_stopwords	after_remove_stopwords
0	on each national day of inauguration since 178...	9	national day inauguration since 1789 people re...	0
1	in washingtons day the task of the people was ...	8	washingtons day task people create weld togeth...	0
2	in lincolns day the task of the people was to ...	9	lincolns day task people preserve nation disru...	0
3	in this day the task of the people is to save ...	12	day task people save nation institutions disru...	0
4	to us there has come a time in the midst of sw...	26	us come time midst swift happenings pause mome...	0
5	if we do not we risk the real peril of inaction	7	risk real peril inaction	0
6	lives of nations are determined not by the cou...	11	lives nations determined count years lifetime ...	0
7	the life of a man is threescore years and ten ...	8	life man threescore years ten little little less	0
8	the life of a nation is the fullness of the me...	11	life nation fullness measure live	0
9	there are men who doubt this	4	men doubt	0

Table 3: Stop word comparison Speech1

5. Sample Sentence Before Removal of Stopwords (Ignore '')

```
'on each national day of inauguration since 1789 the people have renewed their sense of dedication to the united states'
```

6. Sample Sentence After Removal of Stopwords (Ignore '')

```
'national day inauguration since 1789 people renewed sense dedication united states'
```

7. Word count before and after the removal of stopwords.

The word count before removal of stopwords for Roosevelt's Speech is 1536 words.
The word count after removal of stopwords for Roosevelts' Speech is 627 words.

EDA Steps for Speech 2 - ('1961 – Kennedy's.txt')

The word count before removal of stopwords for Kennedy's Speech is 1546 words.

1. Remove punctuation and special character like (--)

	Speech2
0	Vice President Johnson, Mr. Speaker, Mr. Chief...
1	For I have sworn I before you and Almighty God...
2	The world is very different now.
3	For man holds in his mortal hands the power to...
4	And yet the same revolutionary beliefs for whi...
5	We dare not forget today that we are the heirs...
6	Let the word go forth from this time and place...
7	Let every nation know, whether it wishes us we...
8	This much we pledge -- and more.
9	To those old allies whose cultural and spiritu...

	Speech2
0	Vice President Johnson Mr Speaker Mr Chief Jus...
1	For I have sworn I before you and Almighty God...
2	The world is very different now
3	For man holds in his mortal hands the power to...
4	And yet the same revolutionary beliefs for whi...
5	We dare not forget today that we are the heirs...
6	Let the word go forth from this time and place...
7	Let every nation know whether it wishes us wel...
8	This much we pledge and more
9	To those old allies whose cultural and spiritu...

Table 4: Remove punctuation Speech2

2. Convert the Speech into Lower Case Conversion and check the Data frame of Speech 2

	Speech2
0	Vice President Johnson, Mr. Speaker, Mr. Chief...
1	For I have sworn I before you and Almighty God...
2	The world is very different now.
3	For man holds in his mortal hands the power to...
4	And yet the same revolutionary beliefs for whi...

	Speech2
0	vice president johnson mr speaker mr chief jus...
1	for i have sworn i before you and almighty god...
2	the world is very different now
3	for man holds in his mortal hands the power to...
4	and yet the same revolutionary beliefs for whi...

Table 5: Lower Case Conversion Speech2

3. Checking of Stopwords in Each Sentence of Speech 2

Number of the Stopwords Present in Roosevelt file is 672

4. Sample Stopwords Count Comparison Before and After

	Speech2	stopwords	Speech_2_after_remove_stopwords	after_remove_stopwords
0	vice president johnson mr speaker mr chief jus...	13	vice president johnson mr speaker mr chief jus...	0
1	for i have sworn i before you and almighty god...	12	sworn almighty god solemn oath forebears I pre...	0
2	the world is very different now	4	world different	0
3	for man holds in his mortal hands the power to...	10	man holds mortal hands power abolish forms hum...	0
4	and yet the same revolutionary beliefs for whi...	22	yet revolutionary beliefs forebears fought sti...	0

Table 6: Stop word comparison Speech2

5. Sample Sentence Before Removal of Stopwords (Ignore ‘’)

```
'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'
```

6. Sample Sentence After Removal of Stopwords (Ignore ‘’)

```
'vice president johnson mr speaker mr chief justice president eisenhower vice president nixon president truman reverend clergy
fellow citizens observe today victory party celebration freedom symbolizing end well beginning signifying renewal well change'
```

7. Word count before and after the removal of stopwords.

   a. The word count before removal of stopwords for Kennedy’s Speech is 1546 words.
   b. The word count after removal of stopwords for Roosevelts’ Speech is 627 words.

EDA Steps for Speech 2 - ('1973 – Nixon’s.txt')

The word count before removal of stopwords for Roosevelt’s Speech is 2028 words.

1. Remove punctuation and special character like (--)

	Speech3
0	Mr. Vice President, Mr. Speaker, Mr. Chief Jus...
1	As we meet here today, we stand on the thresho...
2	The central question before us is: How shall w...
3	Let us resolve that this era we are about to e...
4	Let us resolve that this will be what it can b...

	Speech3
0	Mr Vice President Mr Speaker Mr Chief Justice ...
1	As we meet here today we stand on the threshol...
2	The central question before us is How shall we...
3	Let us resolve that this era we are about to e...
4	Let us resolve that this will be what it can b...

Table 7: Remove punctuation Speech3

2. Convert the Speech into Lower Case Conversion and check the Data frame of Speech 3

	Speech3
0	Mr. Vice President, Mr. Speaker, Mr. Chief Jus...
1	As we meet here today, we stand on the thresho...
2	The central question before us is: How shall w...
3	Let us resolve that this era we are about to e...
4	Let us resolve that this will be what it can b...

	Speech3
0	mr vice president mr speaker mr chief justice ...
1	as we meet here today we stand on the threshol...
2	the central question before us is how shall we...
3	let us resolve that this era we are about to e...
4	let us resolve that this will be what it can b...

Table 8: Lower Case Conversion Speech3

3. Checking of Stopwords in Each Sentence of Speech 2

Number of the Stopwords Present in Roosevelt file is 969

4. Sample Stopwords Count Comparison Before and After

	Speech3	stopwords	Speech_3_after_remove_stopwords
0	mr vice president mr speaker mr chief justice ...	17	mr vice president mr speaker mr chief justice ...
1	as we meet here today we stand on the threshol...	11	meet today stand threshold new era peace world
2	the central question before us is how shall we...	6	central question us shall use peace
3	let us resolve that this era we are about to e...	21	let us resolve era enter postwar periods often...
4	let us resolve that this will be what it can b...	21	let us resolve become time great responsibilit...

Table 9: Stop word comparison Speech3

5. Sample Sentence Before Removal of Stopwords (Ignore ‘’)

```
'mr vice president mr speaker mr chief justice senator cook mrs eisenhower and my fellow citizens of this great and good countr
y we share together when 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'
```

6. Sample Sentence After Removal of Stopwords (Ignore ‘’)

```
'mr vice president mr speaker mr chief justice senator cook mrs eisenhower fellow citizens great good country share together me
t four years ago america bleak spirit depressed prospect seemingly endless war abroad destructive conflict home'
```

7. Word count before and after the removal of stopwords.

   a. The word count before removal of stopwords for Kennedy’s Speech is 2028 words.
   b. The word count after removal of stopwords for Roosevelts’ Speech is 833 words.

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

Speech 1 - Top Three Words. (After Removing the Stopwords) in ('1941-Roosevelt.txt') with frequency count ----[('nation', 11), ('know', 10), ('spirit', 9)].

Speech 2 - Top Three Words. (After Removing the Stopwords) in ('1961-Kennedy.txt') with frequency count ----[('let', 16), ('us', 12), ('world', 8)].

Speech 3 - Top Three Words. (After Removing the Stopwords) in ('1973-Nixon.txt') with frequency count ----[('us', 26), ('let', 22), ('peace', 19)].

Speech_Name	1st Most Occured Word	2nd Most Occured word	3rd Most Occured Word
1941-Roosevelt.txt	nation	know	spirit
1961-Kennedy.txt	let	us	world
1973-Nixon.txt	us	let	peace

Table 10: Top three words comparison

Note - As we know that we can include "us" and "let" and "know" in stopwords, but in FAQ these words are mentioned as not to remove. So, we didn’t remove these words from the list. In future these words can be removed based on context and sentiments of the business problem.

2.4) Plot the word cloud of each of the three speeches. (after removing the stopwords)

Word Cloud for Roosevelt's Speech (after cleaning)

Fig4: Word Cloud for Roosevelt's Speech (after cleaning)

Word Cloud for Kennedy's Speech (after cleaning)

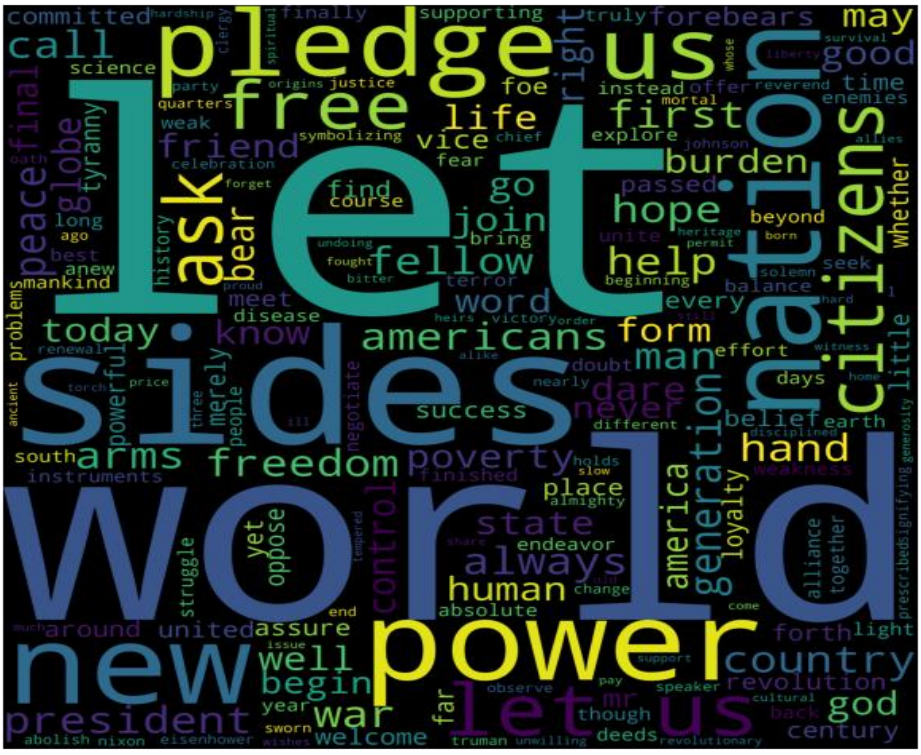


Fig 5: Word Cloud for Kennedy's Speech (after cleaning)

Word Cloud for Nixon's Speech (after cleaning)

Fig 6: Word Cloud for Nixon’s Speech (after cleaning)