Problem 1: Exit Poll

Problem Statement:

You are hired by one of the leading news channel 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.

Questions:

Data Ingestion: (12 marks)

1. Read the dataset. Do the descriptive statistics and do null value condition check. Write an inference on it. (5 Marks)
2. Perform Univariate and Bivariate Analysis. Do exploratory data analysis. Check for Outliers. (7 Marks)

Data Preparation: (5 marks)

1. 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). (5 Marks)

Modelling: (26 marks)

1. Apply Logistic Regression and LDA (linear discriminant analysis). (5 marks)
2. Apply KNN Model and Naïve Bayes Model. Interpret the results. (7 marks)
3. Model Tuning, Bagging (Random Forest should be applied for Bagging) and Boosting. (7 marks)
4. 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. (7 marks)

Inference: (5 marks)

1. Based on these predictions, what are the insights? (5 marks)

Answer:

EDA

The number of variables in the data set: 1525
The number of samples in the data set: 10

The top 10 entries in the data set are:

	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
5	6	Labour	47	3	4	4	4	4	2	male
6	7	Labour	57	2	2	4	4	11	2	male
7	8	Labour	77	3	4	4	1	1	0	male
8	9	Labour	39	3	3	4	4	11	0	female
9	10	Labour	70	3	2	5	1	11	2	male

The columns in the data set are:

1. Unnamed: 0
2. vote
3. age
4. economic.cond.national
5. economic.cond.household
6. Blair
7. Hague
8. Europe
9. political.knowledge
10. gender

The info of the raw data is as shown below:

#	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

- vote and gender are object data types and categorical in nature (nominal scale)
- age is integer data type and continuous in nature (ratio scale)
- all other columns are integer data types and categorical in nature (by data definition it can be assumed to be ordinate scaled)
- There are no missing values as all the columns have 1525 non null values. The age variable does not have any special character in it. However all the variables must be checked if they have any undesirable/unrealistic values.

The five number summary of the data:

	count	unique	top	freq
vote	1525	2	Labour	1063
economic_cond_national	1525	5	3	607
economic_cond_household	1525	5	3	648
blair	1525	5	4	836
hague	1525	5	2	624
europe	1525	11	11	338
political_knowledge	1525	4	2	782
gender	1525	2	female	812

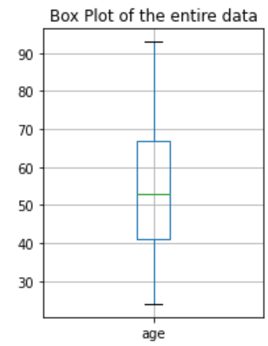
	count	mean	std	min	25%	50%	75%	max
age	1525	54.182295	15.711209	24	41	53	67	93

- The range of age is meaningful and the possibility of having undesirable/unrealistic values is ruled out
- The classes of categorical variables will be checked in univariate analysis
- age is almost normally distributed
- vote and gender can be converted into a binary variable as they have only 2 classes
- europe has 11 classes and the mode class has 338 entries
- All the other categorical columns have 4-5 classes and the class distribution is quite uneven as the model class by itself has almost 40% to 56% of the entire data entry

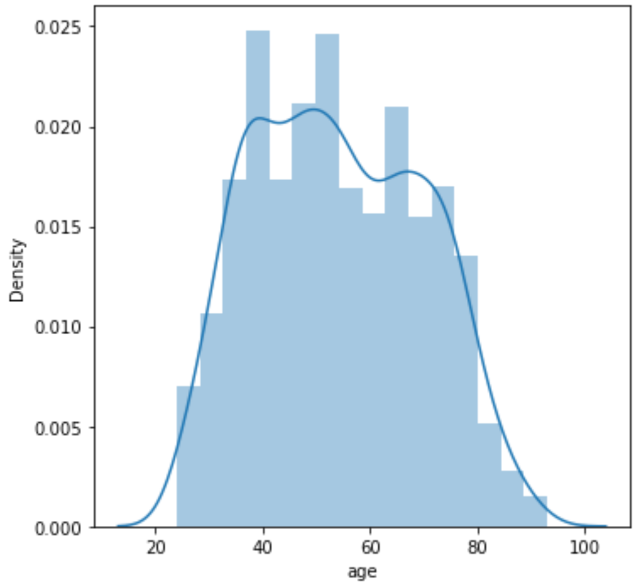
The target variable in the data set is vote.

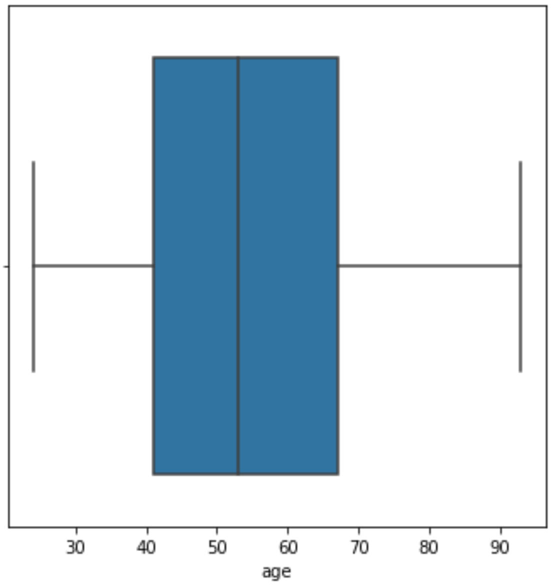
Univariate analysis

Box plot of the entire data set:



Variable: **age**





age is Positive or Right skewed.
The number of outliers in age is 0

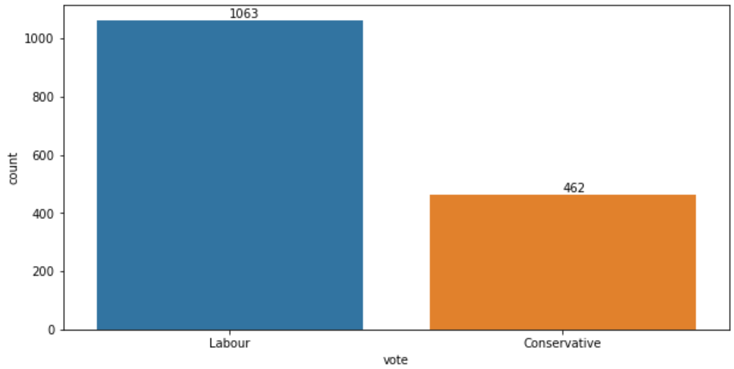
Variable: **vote**

The number of unique entries in the column vote : 2
The entry with the highest frequency in vote : Labour

Percentage share:

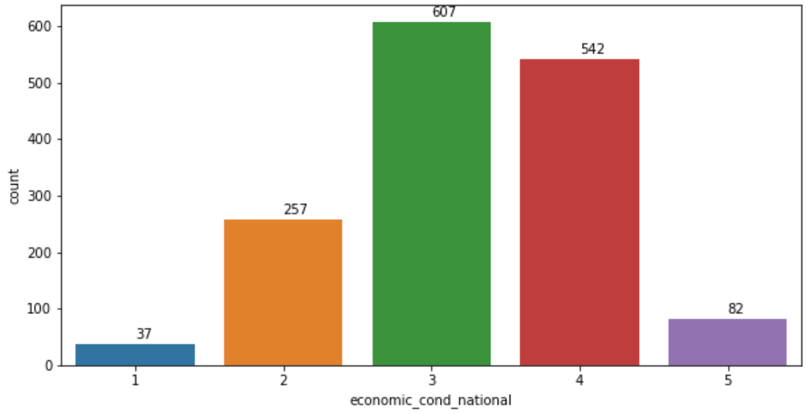
Class	% of Total
Labour	69.704918
Conservative	30.295082

Name: vote, dtype: float64



Variable: **economic_cond_national**

The number of unique entries in the column economic_cond_national : 5
The entry with the highest frequency in economic_cond_national : 3



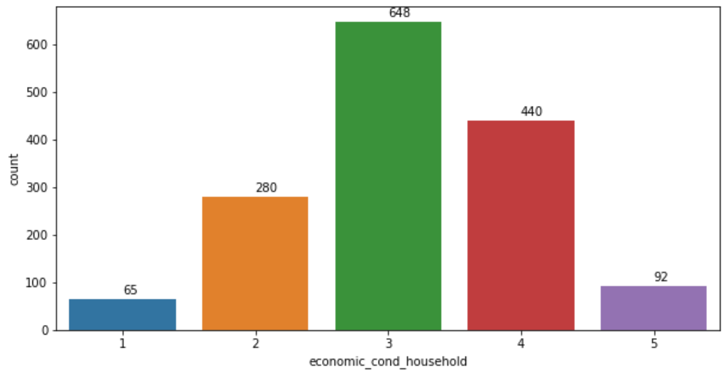
Percentage share:

Class	% of Total
3	39.803279
4	35.540984
2	16.852459
5	5.377049
1	2.42623

Name: economic_cond_national, dtype: float64

Variable: **economic_cond_household**

The number of unique entries in the column economic_cond_household : 5
The entry with the highest frequency in economic_cond_household : 3



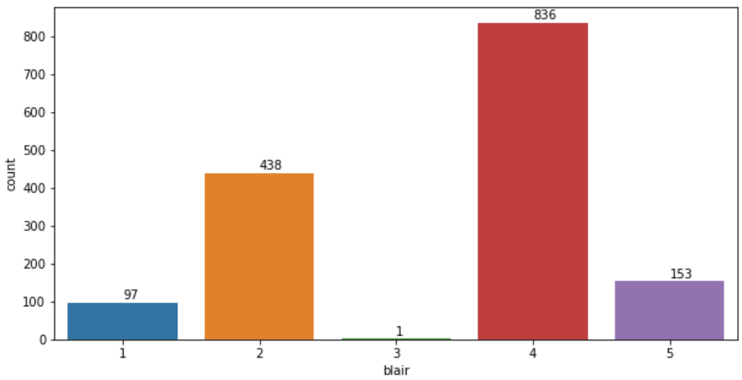
Percentage share:

Class	% of Total
3	42.491803
4	28.852459
2	18.360656
5	6.032787
1	4.262295

Name: economic_cond_household, dtype: float64

Variable: **blair**

The number of unique entries in the column blair : 5
The entry with the highest frequency in blair : 4



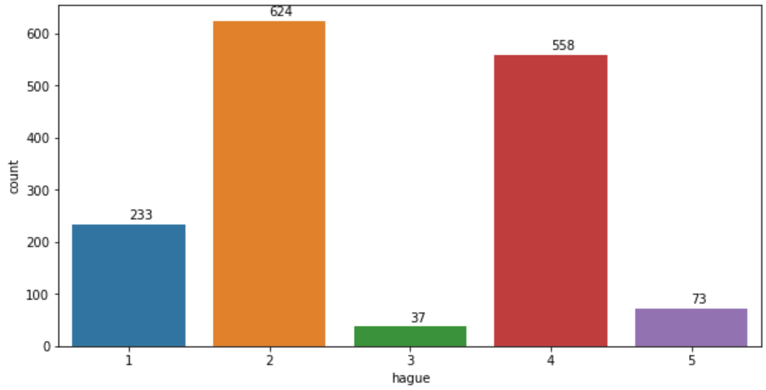
Percentage share:

Class	% of Total
4	54.819672
2	28.721311
5	10.032787
1	6.360656
3	0.065574

Name: blair, dtype: float64

Variable: **hague**

The number of unique entries in the column hague : 5
The entry with the highest frequency in hague : 2



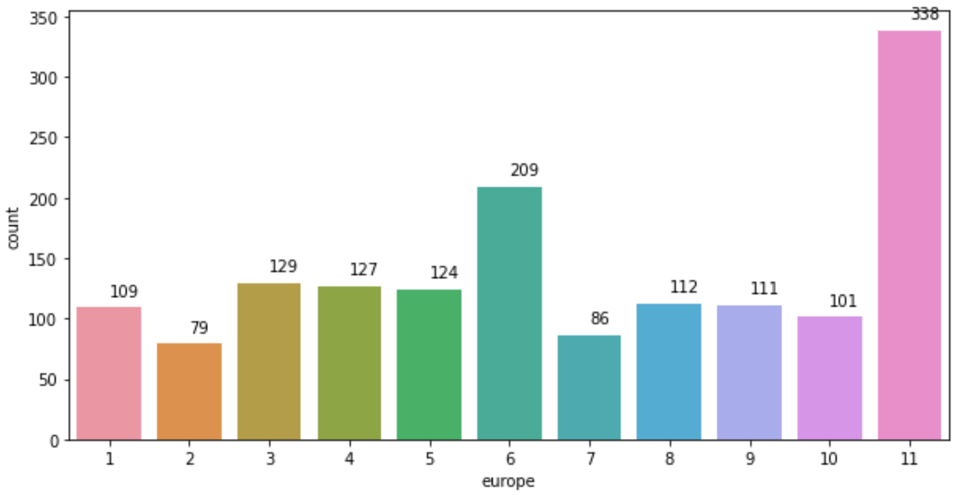
Percentage share:

Class	% of Total
2	40.918033
4	36.590164
1	15.278689
5	4.786885
3	2.42623

Name: hague, dtype: float64

Variable: **europe**

The number of unique entries in the column europe : 11
The entry with the highest frequency in europe : 11



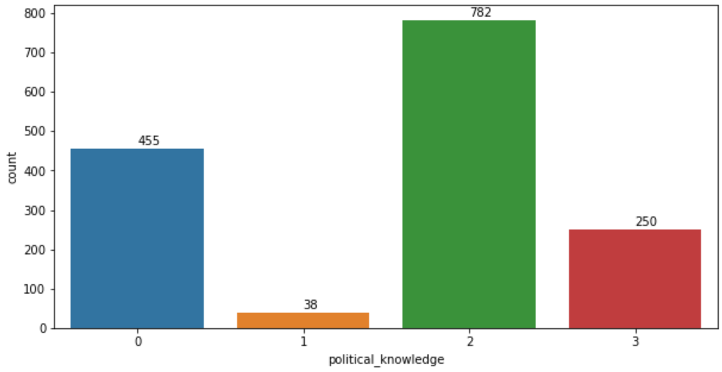
Percentage share:

Class	% of Total
11	22.163934
6	13.704918
3	8.459016
4	8.327869
5	8.131148
8	7.344262
9	7.278689
1	7.147541
10	6.622951
7	5.639344
2	5.180328

Name: europe, dtype: float64

Variable: **political_knowledge**

The number of unique entries in the column political_knowledge : 4
The entry with the highest frequency in political_knowledge : 2



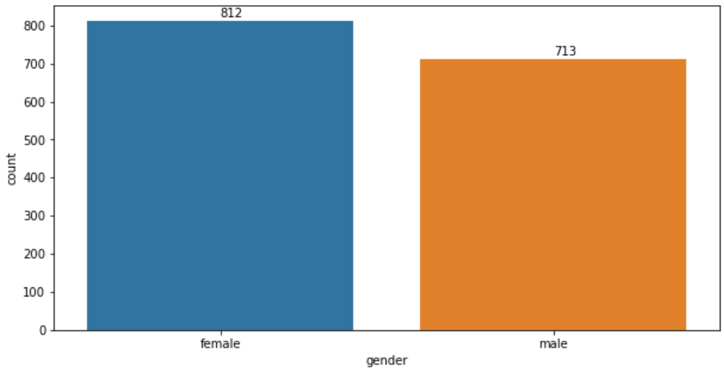
Percentage share:

Class	% of Total
2	51.278689
0	29.836066
3	16.393443
1	2.491803

Name: political_knowledge, dtype: float64

Variable: **gender**

The number of unique entries in the column gender : 2
The entry with the highest frequency in gender : female



Percentage share:

Class	% of Total
female	53.245902
male	46.754098

Name: gender, dtype: float64

- age variable has no outliers and is almost normally distributed (very slightly right skewed)
- vote (target variable is not imbalanced) not required to balance data
- Except in vote and europe variables, all the other variables have their modal class at 40%-55% of the total number of entries
- economic_cond_national and economic_cond_household have very similar class distributions
- there is equal sampling of male and female voters
- blair and hague is the assessment scores of labour and conservative leader. The score for 3 is the least this very clearly shows that almost the entire crowd has already made up their mind as to whom they want to vote
- the blair and hague score also shows that for blair modal class is 4 while for hague it is 2. Predominantly, the population has scored higher for blair than hague. Close to 65% of the population has marked more than 3 in blair while the same for hague is around 40%. This solidifies the above inference .
- In blair the class 2 is half of the number of class 4. However in hague the difference is very less. This shows that the population has clearly distinguished the labour to give.

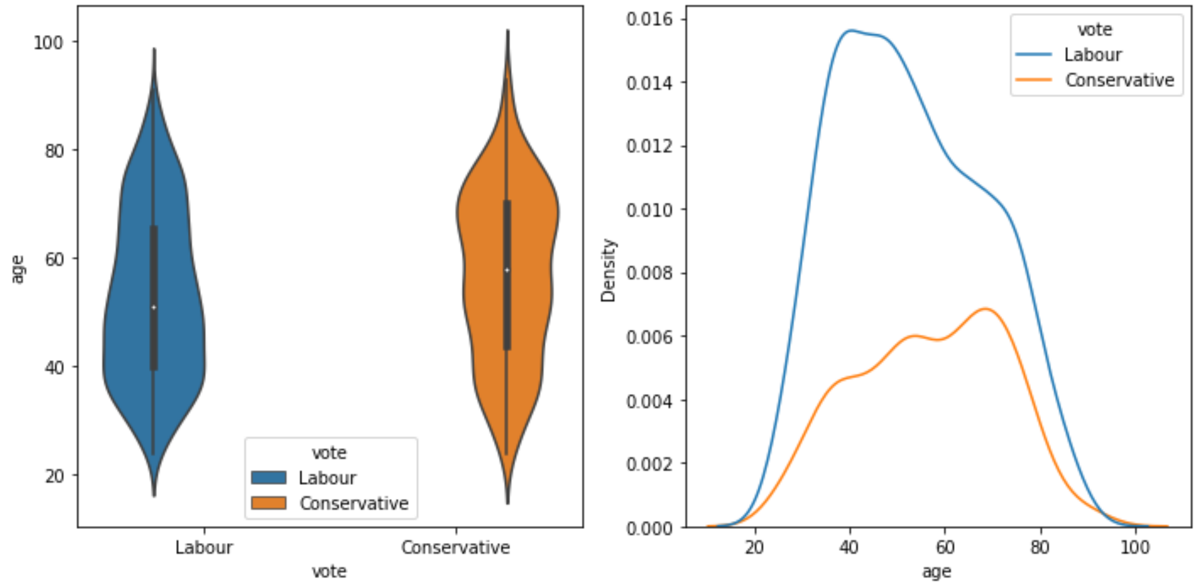
Ascore of 4 however in hague they are confused and so half the difference is very less.

- the modal class in europe variable is 11 highly euroskeptic the next highest frequency class is the middle class (6th class) which mostly depicts neutral feeling towards EU. The remaining distribution of classes are almost the same and hence may prove to be a weak predictor

**All the above inferences are made assuming that the higher the number better it is and vice versa.

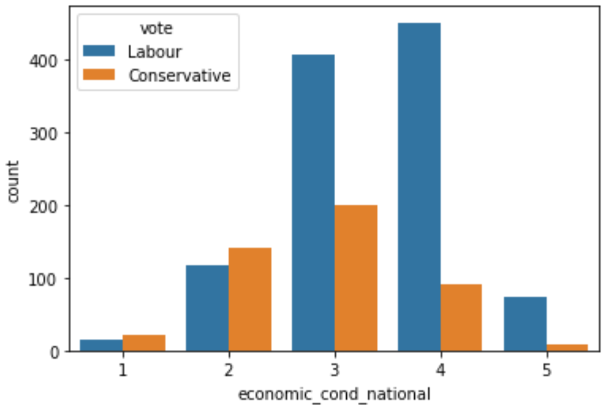
Bivariate analysis

age vs Vote



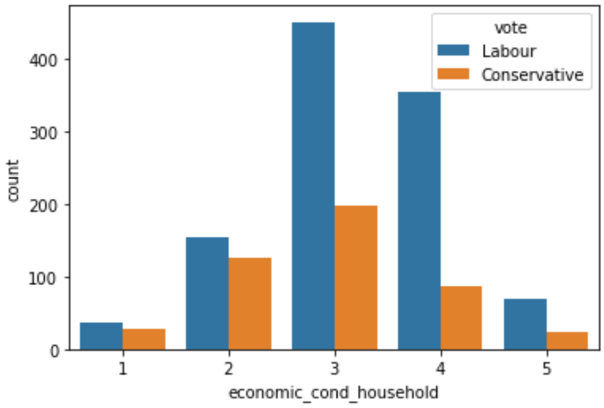
---

economic_cond_national vs vote



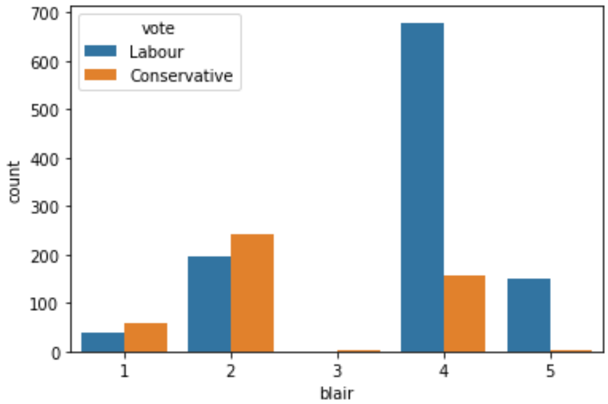
---

economic_cond_household vs vote



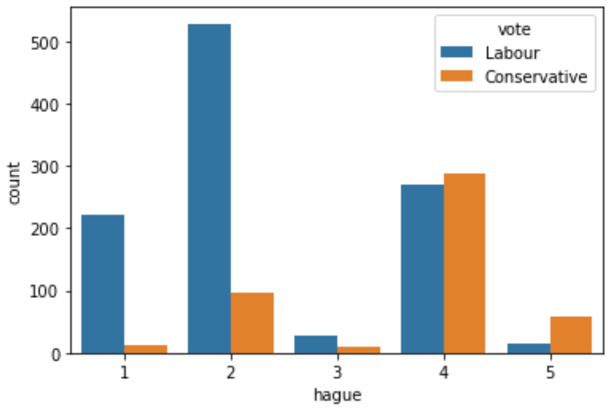
---

blair vs vote



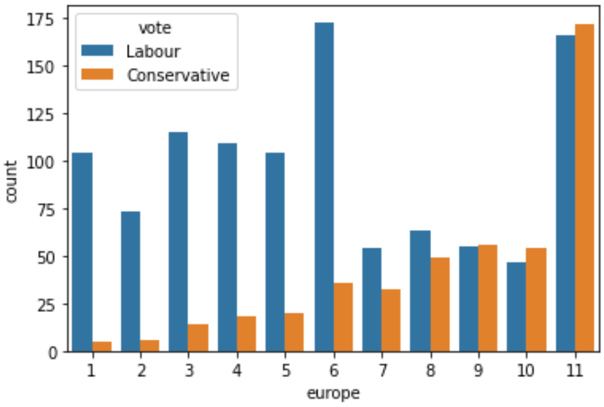
---

hague vs vote



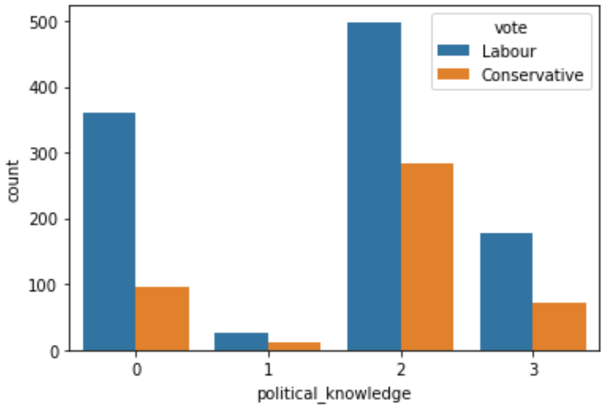
---

europe vs vote



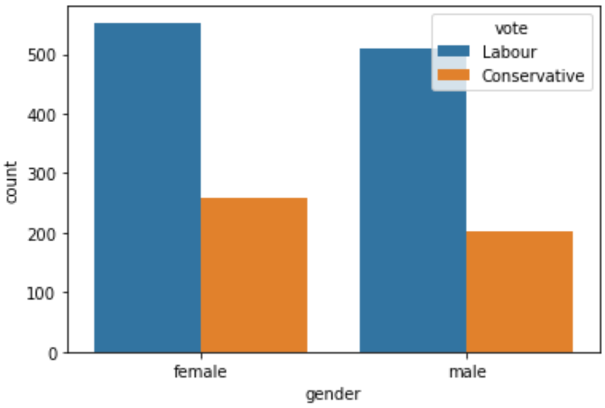
---

political_knowledge vs vote



---

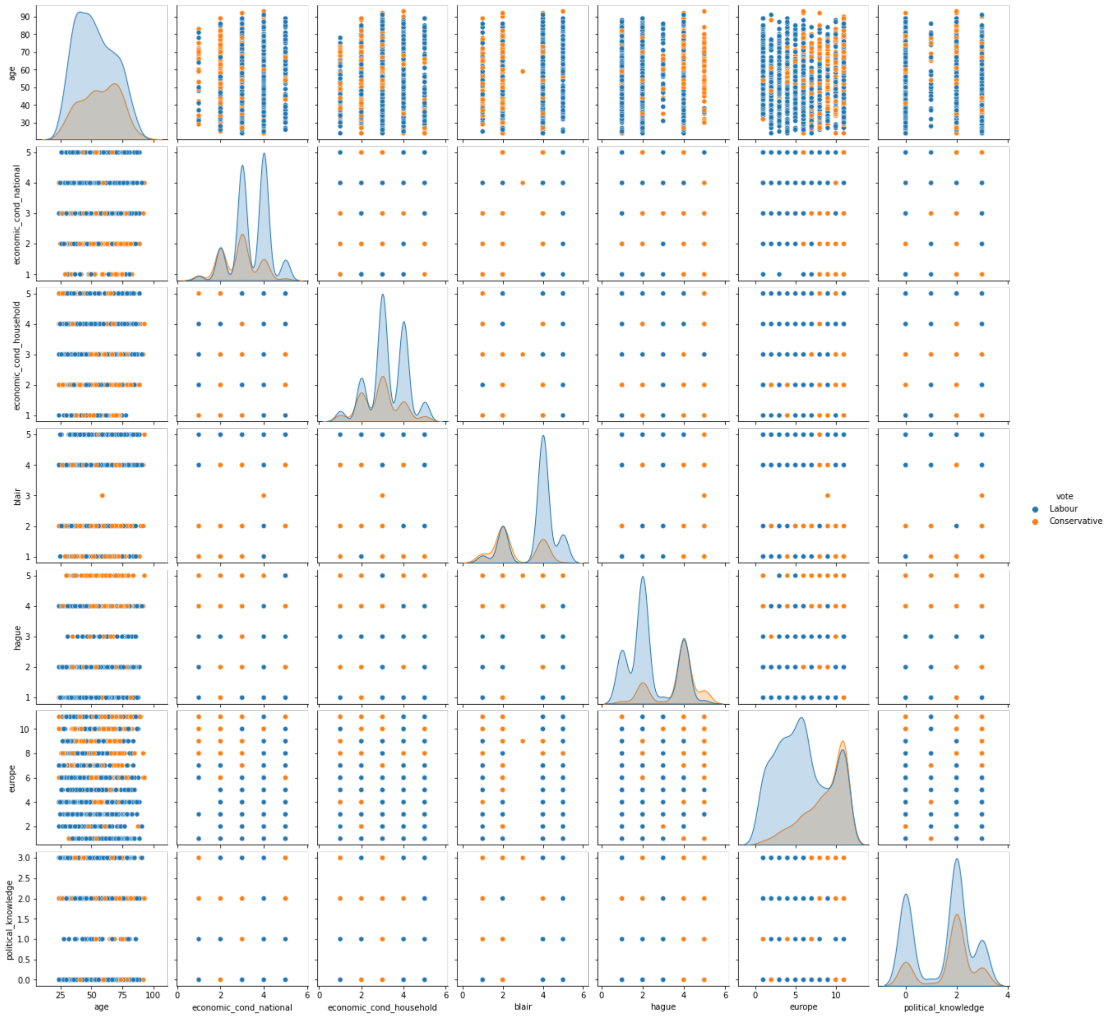
gender vs vote



- Lower age groups are more likely to vote for labour party and older age groups incline towards conservative party but since the graphs are almost overlapping this may be a poor predictor
- For lower scores in economic_cond_national and blair, voters are likely to vote for conservative party and for voters with higher scores are more likely to vote for labour party
- In economic_cond_household and political_knowledge, for all the scores, the voters are likely to vote for labour party. Hence, this variable may prove to be a poor predictor
- For lower scored in hague and europe, voters are likely to vote for labour party and for voters with higher scores are more likely to vote for conservative party

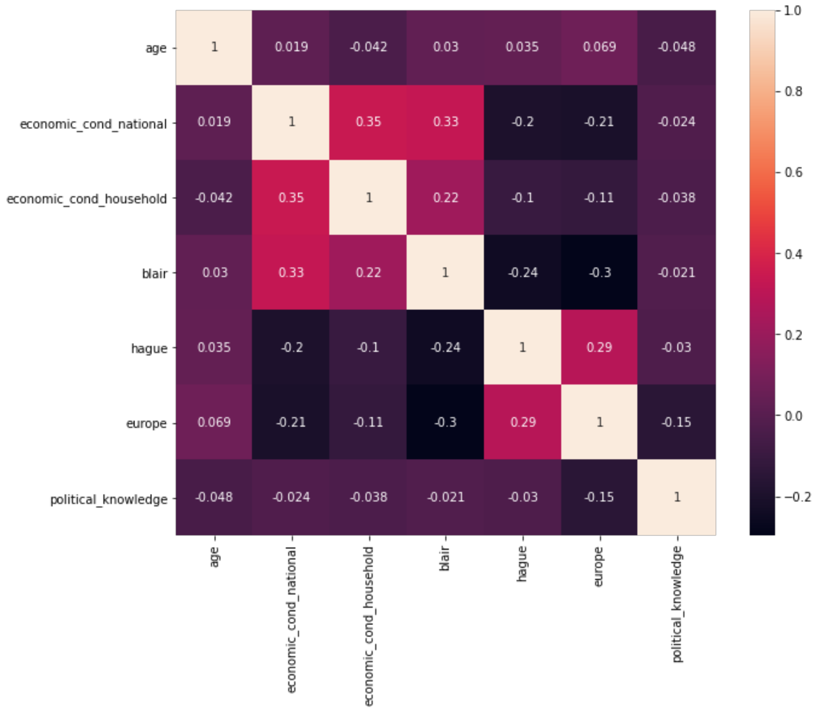
Multivariate analysis

Pair plot:



- From the pair plot, it can be observed that the labour class has a distinct region whereas the conservative class almost always over laps with the labour region. This may cause a little difficulty in predicting the conservative class.

Correlation heatmap:



- There is no multi-collinearity amongst the independent variables.

blair score	party	vote count
1	Conservative	59
	Labour	38
2	Conservative	242
	Labour	196
3	Conservative	1
4	Labour	679
	Conservative	157
5	Labour	150
	Conservative	3

hague score	party	vote count
1	Labour	222
	Conservative	11
2	Labour	528
	Conservative	96
3	Labour	28
	Conservative	9
4	Conservative	287
	Labour	271
5	Conservative	59
	Labour	14

Form the above table we can observe that in the blair tab for the score of 4 there are 836 voters out of which 679 voted for labour party. However out of 558 voters who marked a score of 4 in hague only 287 voted for the conservative class. This further strengthens the inference that either hague assessment isn't very effective in assessing if high score of hague is actually that the voter will vote for the conservative party.

Data Processing

No outliers or missing values are present in the data.

We will remove the repeated rows as these are redundant and they also add weightage to the same data point. The data is now reduced to 1517 rows × 9 columns.

Since the variables age and gender are nominal, we will proceed with dummy variable creation because label encoding is not a desired encoding technique for nominal data. The number of columns and rows remain unchanged as vote and gender are binary class variables. Assuming that the order in which they are rated have a meaning, the other variables except age are left as it is.

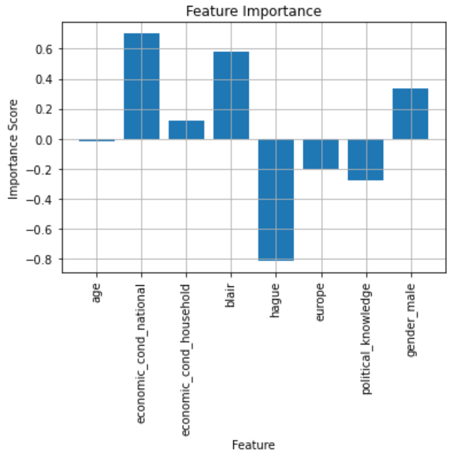
For many models we can proceed with the raw data as is it. However for some models that use distance as a measure, we will have to make the variables unit independent. We will proceed with standard scalar to scale the data for all such models that use distance as a measure.

We will split the data into test and train sets having the ratio of 70:30 respectively. We will train the data with the train set and then test it with the testing set.

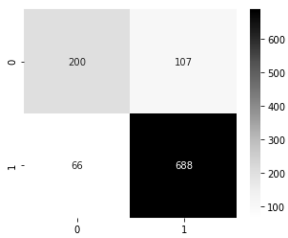
Model Building:

We will train and test various different types of classification models. They are listed below:

1. Logistic Regression : Non Scaled data
    a. No Hyperparameters : This model is a simple logistic regression model without any hyperparameters.

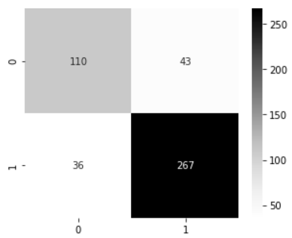



```
Logistic Regression Model (Train Data):
Confusion Matrix:
```



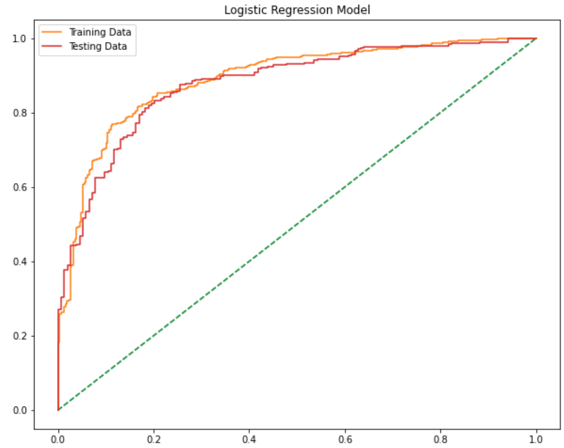

```
The classification report:
              precision    recall  f1-score   support

           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
```

```
Logistic Regression Model (Test Data):
Confusion Matrix:
```

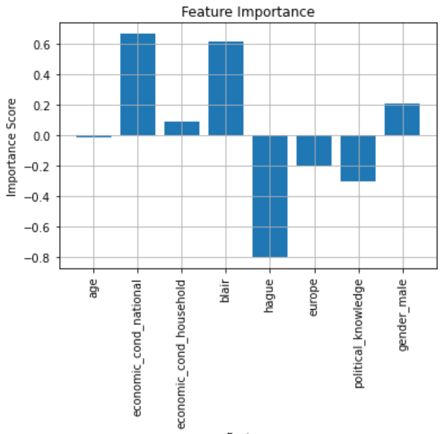



```
The classification report:
              precision    recall  f1-score   support

           0       0.75      0.72      0.74       153
           1       0.86      0.88      0.87       303

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

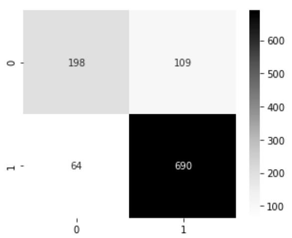
AUC (Train): 0.890
AUC (Test): 0.880



b. With Model Tuning : This model is a logistic regression model designed with hyperparameters. The best hyperparameters were obtained by doing a grid search.
Best parameters: {'max_iter': 50, 'penalty': 'l2', 'solver': 'liblinear', 'tol': 0.0001}

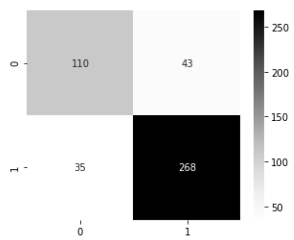


Logistic Regression Model [Post tunning] (Train Data):
Confusion Matrix:



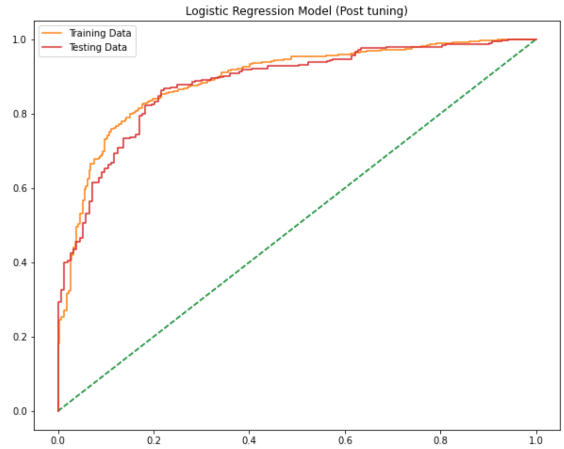

```
The classification report:
              precision    recall  f1-score   support

           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
```

Logistic Regression Model [Post tunning] (Test Data):
Confusion Matrix:



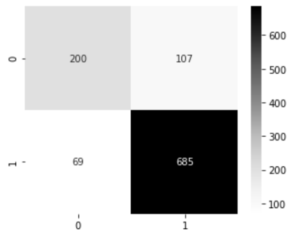

```
The classification report:
              precision    recall  f1-score   support

           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
```

AUC (Train): 0.890
AUC (Test): 0.880



2. Linear Discriminant Analysis : Non Scaled data
    a. Simple LDA Model : This model is a simple LDA model that uses the maximum likelihood as 0.5.

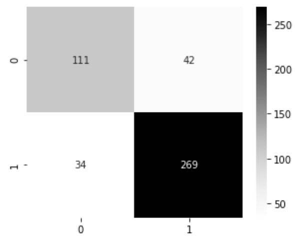
LDA Model (Train Data):
Confusion Matrix:



The classification 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

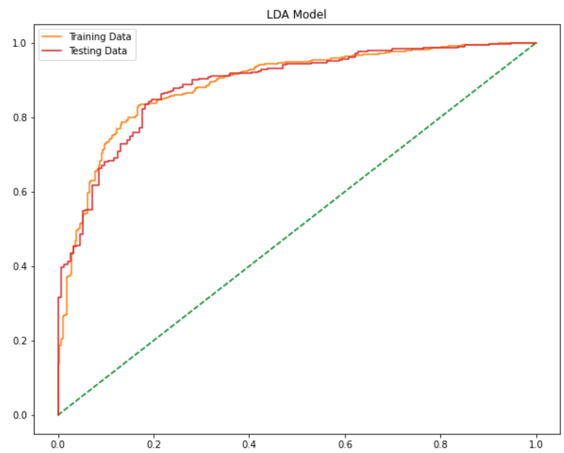
LDA Model (Test Data):
Confusion Matrix:



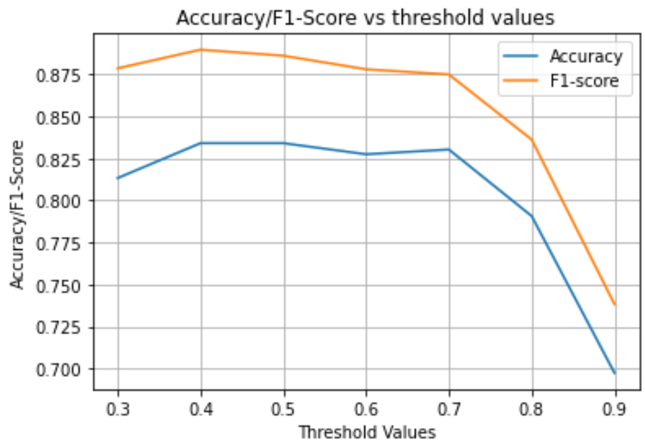
The classification report:

	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

AUC (Train): 0.890
AUC (Test): 0.890

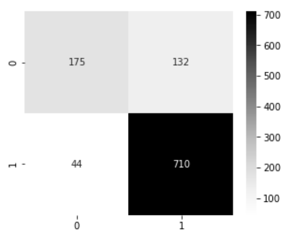


    b. LDA with custom maximum likelihood : This is an LDA model that uses a custom threshold probability.
    When a Accuracy/F1-score is plotted for various values of the max likelihood/threshold then the below graph is obtained:



It can be observed that the highest values of accuracy and F1-score are at 0.4

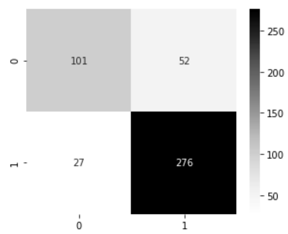
```
LDA Model Custom Cut-off=0.4 (Train Data):
When the cut-off probability is 0.4
Confusion Matrix:
```



The classification 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

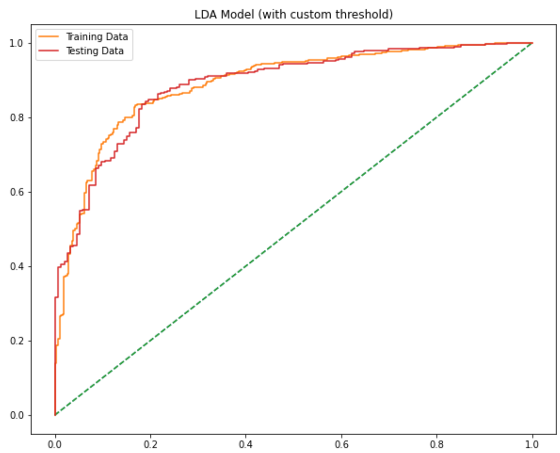
```
LDA Model Custom Cut-off=0.4 (Test Data):
When the cut-off probability is 0.4
Confusion Matrix:
```



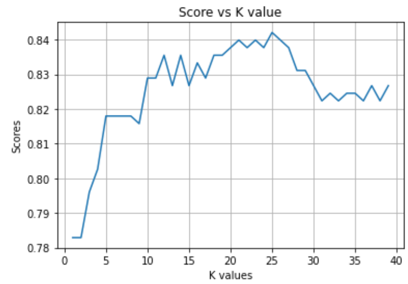
The classification report:

	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

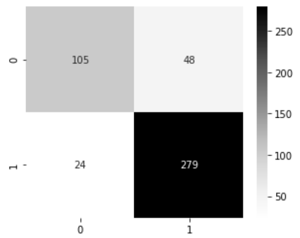
AUC (Train): 0.890
AUC (Test): 0.890



3. K Nearest Neighbours : Scaled Data
    a. KNN Model : We will first find the number of appropriate neighbours required to get maximum accuracy for the model. Then we will design a model with the previously obtained number of neighbours.
    Ideally as a rule of thumb for a binary classification the value of k taken should be around square root of the number of rows (=39).
    On evaluating the scores for various values of k the below graph is obtained

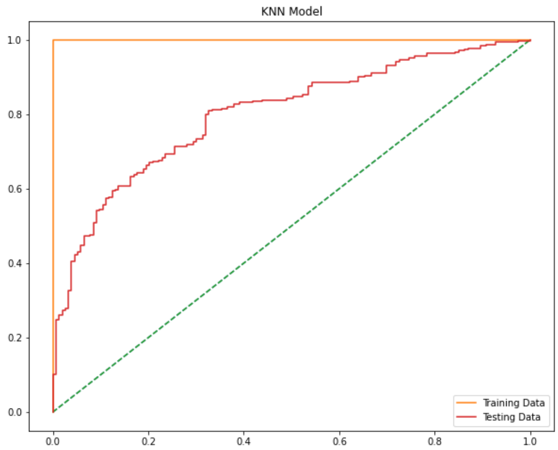



```
KNN Model (Test Data):
Confusion Matrix:
```



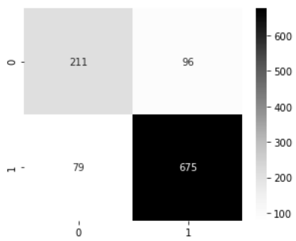

```
The classification report:
              precision    recall  f1-score   support

           0       0.81      0.69      0.74       153
           1       0.85      0.92      0.89       303

    accuracy                           0.84       456
   macro avg       0.83      0.80      0.82       456
weighted avg       0.84      0.84      0.84       456
```

AUC (Train): 1.000
AUC (Test): 0.800



4. Naive Bayes : Non Scaled Data
    a. Simple Naive Bayes Model : This is a simple naive bayes model.

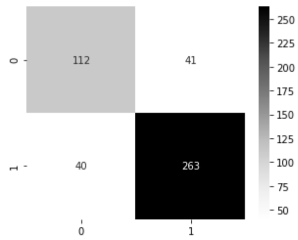
Naive Bayes Model (Train Data):
Confusion Matrix:



The classification 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

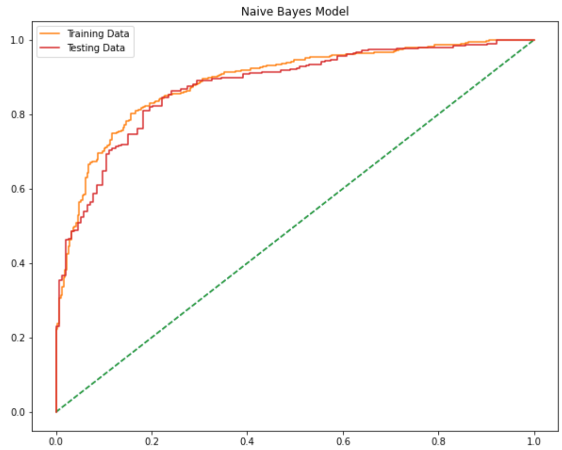
Naive Bayes Model (Test Data):
Confusion Matrix:



The classification report:

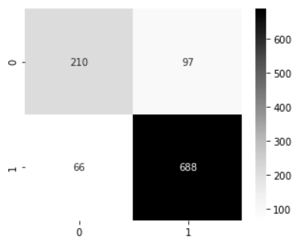
	precision	recall	f1-score	support
0	0.74	0.73	0.73	153
1	0.87	0.87	0.87	303
accuracy			0.82	456
macro avg	0.80	0.80	0.80	456
weighted avg	0.82	0.82	0.82	456

AUC (Train): 0.890
AUC (Test): 0.880



5. Boosting : Non Scaled Data
   a. Adaptive Boosting : In this model technique we will proceed with the default base classifier (simple CART model).

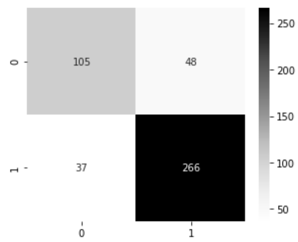
ADA Boost Model (Train Data):
Confusion Matrix:



The classification report:

	precision	recall	f1-score	support
0	0.76	0.68	0.72	307
1	0.88	0.91	0.89	754
accuracy			0.85	1061
macro avg	0.82	0.80	0.81	1061
weighted avg	0.84	0.85	0.84	1061

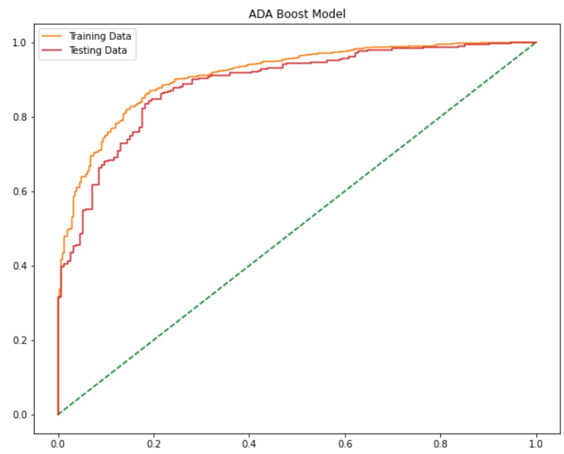
ADA Boost Model (Test Data):
Confusion Matrix:



The classification report:

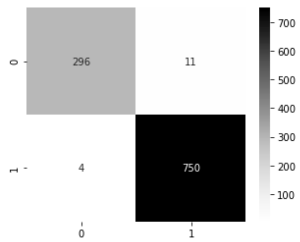
	precision	recall	f1-score	support
0	0.74	0.69	0.71	153
1	0.85	0.88	0.86	303
accuracy			0.81	456
macro avg	0.79	0.78	0.79	456
weighted avg	0.81	0.81	0.81	456

AUC (Train): 0.920
AUC (Test): 0.860



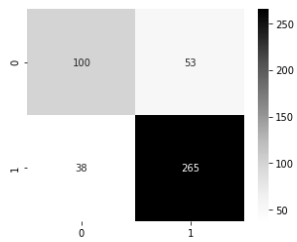
b. Gradient Boosting : In this model technique we will proceed with the default base classifier (simple CART model).

```
Gradient Boost Model (Train Data):
Confusion Matrix:
```



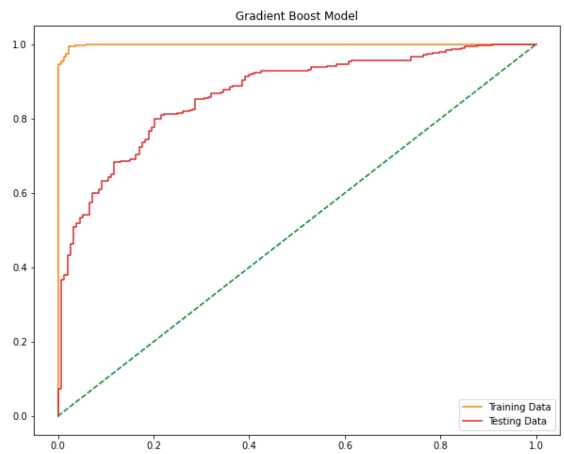

```
The classification report:
              precision    recall  f1-score   support

           0       0.99      0.96      0.98       307
           1       0.99      0.99      0.99       754

    accuracy                           0.99      1061
   macro avg       0.99      0.98      0.98      1061
weighted avg       0.99      0.99      0.99      1061
```

```
Gradient Boost Model (Test Data):
Confusion Matrix:
```



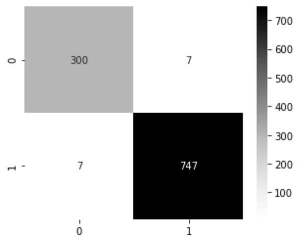

```
The classification report:
              precision    recall  f1-score   support

           0       0.72      0.65      0.69       153
           1       0.83      0.87      0.85       303

    accuracy                           0.80       456
   macro avg       0.78      0.76      0.77       456
weighted avg       0.80      0.80      0.80       456
```

AUC (Train): 1.000
AUC (Test): 0.870



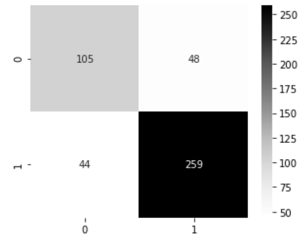
6. Bagging : Non Scaled Data
   a. Basic Bagging : In this model technique we will proceed with the default base classifier (complex CART model).

Bagging Model (Train Data):
Confusion Matrix:



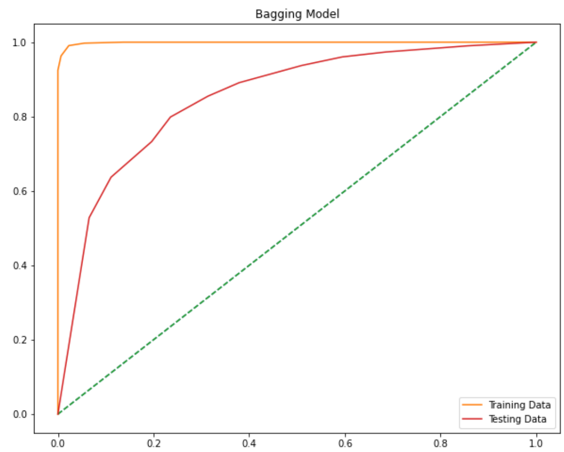

```
The classification report:
              precision    recall  f1-score   support

           0       0.98      0.98      0.98       307
           1       0.99      0.99      0.99       754

    accuracy                           0.99      1061
   macro avg       0.98      0.98      0.98      1061
weighted avg       0.99      0.99      0.99      1061
```

Bagging Model (Test Data):
Confusion Matrix:



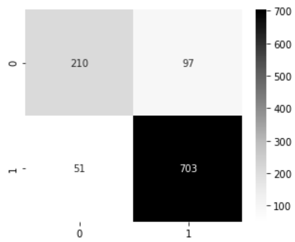

```
The classification report:
              precision    recall  f1-score   support

           0       0.70      0.69      0.70       153
           1       0.84      0.85      0.85       303

    accuracy                           0.80       456
   macro avg       0.77      0.77      0.77       456
weighted avg       0.80      0.80      0.80       456
```

AUC (Train): 1.000
AUC (Test): 0.850



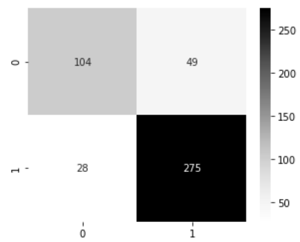
b. Random Forest : Since random forest uses bagging technique we will proceed with random forest to implement a bagging technique on the data. To obtain the best parameters for random forest we will perform a grid search.
Best hyperparameters from the grid search:
{'max_features': 4, 'min_samples_leaf': 2, 'min_samples_split': 50, 'n_estimators': 200}

Random Forest Model (Train Data):
Confusion Matrix:



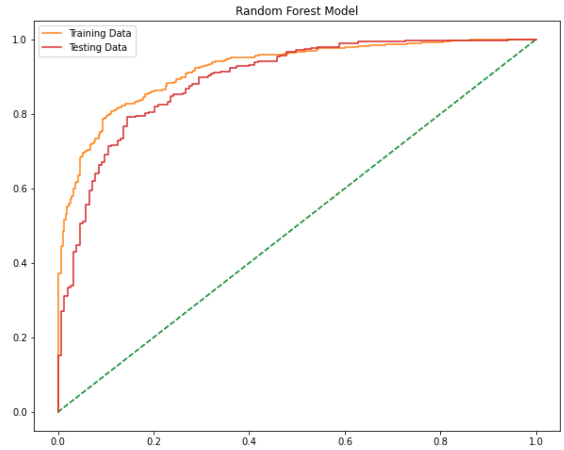

```
The classification report:
              precision    recall  f1-score   support

           0       0.80      0.68      0.74       307
           1       0.88      0.93      0.90       754

    accuracy                           0.86      1061
   macro avg       0.84      0.81      0.82      1061
weighted avg       0.86      0.86      0.86      1061
```

Random Forest Model (Test Data):
Confusion Matrix:




```
The classification 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      0.79      0.80       456
weighted avg       0.83      0.83      0.83       456
```

AUC (Train): 0.920
AUC (Test): 0.890



The all the model parameters are consolidated in one table to make it easy for comparison:

	Logistic Regression (Train)	Logistic Regression_ Tune (Train)	LDA (Train)	LDA [Thresh>0.4] (Train)	Naive Bayes (Train)	ADA Boost (Train)	Gradient Boost (Train)	Bagging (Train)	Random Forest_Tune (Train)
Accuracy	84	84	83	83	83	83	99	99	86
AUC	89	89	89	89	89	91	100	100	92
Recall	91	92	91	94	91	91	99	99	93
Precision	87	86	86	84	86	86	99	99	88
F1 Score	89	89	89	89	89	89	99	99	90

	Logistic Regression (Test)	Logistic Regression_Tune (Test)	LDA (Test)	LDA [Thresh>0.4] (Test)	KNN (Test)	Naive Bayes (Test)	ADA Boost (Test)	Gradient Boost (Test)	Bagging (Test)	Random Forest_Tune (Test)
Accuracy	83	83	83	83	84	83	81	80	80	83
AUC	88	88	89	89	80	88	88	87	85	89
Recall	88	88	89	91	92	89	88	87	85	91
Precision	86	86	86	84	85	86	85	83	84	85
F1 Score	87	87	88	87	89	88	86	85	85	88

Upon carefully comparing the scores of all the models the following is the conclusion:

- KNN, gradient boosting and simple bagging models are over fitted and hence cannot be used.
- All the other models are performing as expected for the majority class. However, for the minority class they are performing fairly okay. A few models that have performed well for both classes are with a good accuracy score are:
    - Random forest with model tunning
    - Linear Discriminant Analysis (with and without custom threshold)

We need a model that performs well for both the classes. Since these have a very good performance in the majority class and a fairly satisfactory performance in the minority class we will try to address the class imbalance and then see if the performance improves.

We will perform SMOTE and obtain class balance and then analyse the models.

On training the models after over sampling:

	Logistic Regression (Train)	Logistic Regression_Tune (Train)	LDA (Train)	LDA [Thresh>0.4] (Train)	Naive Bayes (Train)	ADA Boost (Train)	Gradient Boost (Train)	Bagging (Train)	Random Forest_Tune (Train)
Accuracy	84	84	84	83	84	84	98	99	86
AUC	91	91	91	91	91	94	100	100	95
Recall	83	83	83	87	83	83	98	98	85
Precision	84	84	85	80	84	85	99	99	88
F1 Score	84	84	84	83	84	84	98	99	86

	Logistic Regression (Test)	Logistic Regression_Tune (Test)	LDA (Test)	LDA [Thresh>0.4] (Test)	KNN (Test)	Naive Bayes (Test)	ADA Boost (Test)	Gradient Boost (Test)	Bagging (Test)	Random Forest_Tune (Test)
Accuracy	80	81	81	82	66	80	80	81	80	79
AUC	87	88	87	87	87	86	86	88	85	88
Recall	80	81	80	84	100	81	84	84	83	81
Precision	89	90	90	88	66	88	86	87	86	87
F1 Score	84	85	85	86	80	84	85	85	84	84

On carefully analysing the results after treating class imbalance, the following can be observed:

- Gradient boosting, bagging and random forest are all over fitted models.
- Logistic regression (with tunning) and LDA model perform well for both the classes, with respect to the accuracy and recall score.
- On looking deeper into the performance of logistic regression (with tunning) and the LDA model based on the F1-Scores we can say that the logistic regression model performs better.

Summary:

We will hence proceed with the logistic regression model for production.

Technical Details of the model:

- Logistic Regression model is the model that performs the best.
- The train data was balanced using SMOTE
- Only Gender and vote was encoded using dummy variables all others were retained as it is assuming that the order of rating had a meaning.
- The hyper parameters for logistic regression model are:
  {'max_iter': 25, 'penalty': 'l2', 'solver': 'sag', 'tol': 1e-05}
- hague (negatively) and blair (positively) are major contributors to the prediction followed by economic_cond_houseold (positively) and gender (positively)
- Even post class imbalance is addressed the labour class has lower accuracy. This can be further supported by the inference drawn from the multivariate analysis, which proves that the voters who rated labour party are very clear in voting for labour where as the voters who rated conservative party high still had a large chunk of voters voting for labour party. This shows that the labour party is not very clear in its objectives or is a very volatile party and so people are sceptic.

Problem 2: Text Mining

Problem Statement:

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

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:

- President Franklin D. Roosevelt in 1941
- President John F. Kennedy in 1961
- President Richard Nixon in 1973

Questions:

1. Find the number of characters, words and sentences for the mentioned documents. (3 Marks)
   (Hint: use .words(), .raw(), .sent() for extracting counts)
2. Remove all the stopwords from all the three speeches. (3 Marks)
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) – (3 Marks)
4. Plot the word cloud of each of the speeches of the variable. (after removing the stopwords) – (3 Marks) [refer to the End-to-End Case Study done in the Mentored Learning Session]

Answer:

The inaugural corpus is a corpus that is integrated within the nltk package. It contains a total of 58 speeches.

Exploratory Analysis:

Some basic exploratory analysis on the speeches are:

1. Word count in each speech:

president name	speeches	word_count
Franklin D. Roosevelt	On each national day of inauguration since 178...	1323
John F. Kennedy	Vice President Johnson, Mr. Speaker, Mr. Chief...	1364
Richard Nixon	Mr. Vice President, Mr. Speaker, Mr. Chief Jus...	1769

2. Character count in each speech (includes spaces as well):

president name	speeches	char_count
Franklin D. Roosevelt	On each national day of inauguration since 178...	7571
John F. Kennedy	Vice President Johnson, Mr. Speaker, Mr. Chief...	7618
Richard Nixon	Mr. Vice President, Mr. Speaker, Mr. Chief Jus...	9991

3. Average word length for each speech:

president name	speeches	avg_word
Franklin D. Roosevelt	On each national day of inauguration since 178...	4.539706
John F. Kennedy	Vice President Johnson, Mr. Speaker, Mr. Chief...	4.461871
Richard Nixon	Mr. Vice President, Mr. Speaker, Mr. Chief Jus...	4.465091

4. The number of default English stop words in each speech:

president name	speeches	stopwords
Franklin D. Roosevelt	On each national day of inauguration since 178...	632
John F. Kennedy	Vice President Johnson, Mr. Speaker, Mr. Chief...	618
Richard Nixon	Mr. Vice President, Mr. Speaker, Mr. Chief Jus...	899

   Close to half of the words in each speech are stop words.

5. The number of sentences in each speech:
   The speeches are transcribed with a new line character after each statement. So we will consider statements to be split by a new line.

president name	speeches	sent_count
Franklin D. Roosevelt	On each national day of inauguration since 178...	38
John F. Kennedy	Vice President Johnson, Mr. Speaker, Mr. Chief...	27
Richard Nixon	Mr. Vice President, Mr. Speaker, Mr. Chief Jus...	51

6. The number of numeric data in each speech:

president name	speeches	numerics
Franklin D. Roosevelt	On each national day of inauguration since 178...	2
John F. Kennedy	Vice President Johnson, Mr. Speaker, Mr. Chief...	1
Richard Nixon	Mr. Vice President, Mr. Speaker, Mr. Chief Jus...	1

Pre-Processing:

The data will have to be cleaned up for the following:

1. Special characters: All the special characters will be removed out as they do not convey any sort of information.
2. Lower case conversion: All the text data is converted to lower case because that way we ca avoid treating the same word as different words. Ex: 'America' and 'america' will be treated as different words.
3. Removing stop words: We will remove the stop words as they do not have any informative/predictive power.
4. Stemming of words: The words in the corpus are stemmed. This converts all the words to the root word. Ex: 'inaugural', 'inauguration' etc will all be converted to the root word (i.e. 'inaugur').
5. Custom removal of useless words: We will remove words that we feel has no or very less meaning/predictive power. For this we will first check for the frequencies of the top 50 words and manually pick up words that we feel will not help in analysis and then remove them out of the corpus manually.

Top 10 words used by each president in their speeches:

Top 10 words used by President Franklin D. Roosevelt in his speech are:

Word	Frequency
spirit	9
democracy	9
life	8
people	7
freedom	6
years	6
mind	5
speaks	5
human	5
men	4

Top 10 words used by President John F. Kennedy in his speech are:

Word	Frequency
world	8
sides	8
pledge	7
new	7
free	5
power	5
citizens	5
cannot	4
americans	4
arms	4

Top 10 words used by President Richard Nixon in his speech are:

Word	Frequency
peace	19
world	16
new	15
responsibility	11
government	10
great	9
home	9
abroad	8
better	7
history	7

Word Cloud for each president:

Word cloud for President Franklin D. Roosevelt

Word cloud for President John F. Kennedy

Word cloud for President Richard Nixon

Summary:

- More than 50% of the words used in each speech is made up of stop words
- President Franklin D. Roosevelt
  - The top 3 words used were: democracy, spirit and life
  - From the word cloud we can observe that he emphasized on people, democracy, life, freedom, spirit, mind, faith.
  - **His speech is mainly focused on the spiritual development and liberation of people.**
- President John F. Kennedy
  - The top 3 words used were: world, sides and new
  - From the word cloud we can observe that he emphasized on sides of political parties, world, power, new, free, citizens, poverty, god, war, peace. There are a verity of words all having the same weightage.
  - His speech is focused on the variety of topics ranging from peace, internal policies, worldly affairs, god, war, power, science, arms.
  - **He probably has new and holistic idea for the future of America.**
- President Richard Nixon
  - The top three words used were: peace, world, new
  - From the world cloud we can observe that he emphasized on world, peace, new, responsibility, home, together, abroad, respect, rights, history.
  - **His speech is focused mainly on the increasing international relations, unity amongst people, focusing on home as a simple unit of the country, government and new policies/ideas. It also looks like he has a lot to learn from the past mistakes as he refers to history many times.**
