[Year
]

EXIT POLL PREDICTION

M.ABINAYA

CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREDICTION OF ELECTION RESULTS

PROBLEM 1:

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

DATA INGESTION :

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

DESCRIPTIVE STATISTICS :

HEAD :

	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

TAIL :

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

DATA INFORMATION :

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1525 entries, 0 to 1524
Data columns (total 10 columns):
 #   Column                   Non-Null Count  Dtype 
---  ------                   --------------  ----- 
 0   Unnamed: 0               1525 non-null   int64 
 1   vote                     1525 non-null   object
 2   age                      1525 non-null   int64 
 3   economic.cond.national   1525 non-null   int64 
 4   economic.cond.household  1525 non-null   int64 
 5   Blair                    1525 non-null   int64 
 6   Hague                    1525 non-null   int64 
 7   Europe                   1525 non-null   int64 
 8   political.knowledge      1525 non-null   int64 
 9   gender                   1525 non-null   object
dtypes: int64(8), object(2)
memory usage: 119.3+ KB
```

DATA SUMMARY :

	count	mean	std	min	25%	50%	75%	max
Unnamed: 0	1525.0	763.000000	440.373894	1.0	382.0	763.0	1144.0	1525.0
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

NULL VALUES :

```
Unnamed: 0                 0
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
```

SKEWNESS :

```
Unnamed: 0                 0.000000
age                        0.144621
economic.cond.national    -0.240453
economic.cond.household   -0.149552
Blair                     -0.535419
Hague                      0.152100
Europe                    -0.135947
political.knowledge       -0.426838
dtype: float64
```

Check DUPLICATES :

```
dups=df.duplicated()
print("Total no of duplicate values = %d" % (dups.sum()))
df[dups]
```

```
Total no of duplicate values = 0
```

Unnamed: 0	vote	age	economic.cond.national	economic.cond.household	Blair	Hague	Europe	political.knowledge	gender

REMOVAL OF UNWANTED COLUMNS : "UNNAMED:0"

	vote	age	economic.cond.national	economic.cond.household	Blair	Hague	Europe	political.knowledge	gender
0	Labour	43	3	3	4	1	2	2	female
1	Labour	36	4	4	4	4	5	2	male
2	Labour	35	4	4	5	2	3	2	male
3	Labour	24	4	2	2	1	4	0	female
4	Labour	41	2	2	1	1	6	2	male
...	...	...	...	...	...	...	...	...	...
1520	Conservative	67	5	3	2	4	11	3	male
1521	Conservative	73	2	2	4	4	8	2	male
1522	Labour	37	3	3	5	4	2	2	male
1523	Conservative	61	3	3	1	4	11	2	male
1524	Conservative	74	2	3	2	4	11	0	female

1525 rows × 9 columns

- Presence of 1525 Rows and 10 Columns in the dataset.
- There are 8 Numerical and 2 Categorical variables.
- There are no NULL values in the dataset.
- There are no Duplicates in the dataset.
- There are no missing values .
- After removal of unwanted column – Unnamed:0 , now we have 1525 Rows and 9 Columns.
- 812 females & 713 Males taken part in survey
- 1063 votes polled in favour of Labour party and 462 votes in favour of Conservative party.

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

EXPLORATORY ANALYSIS :

NULL VALUES :

```
Unnamed: 0                 0
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
```

SHAPE :

```
df.shape

(1525, 9)
```

After removing unnamed: 0 column , we have 1525 Rows and 9 Columns in the dataset.

DATA TYPES :

```
Unnamed: 0                  int64
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
```

UNIQUE VALUES :

```
df.gender.value_counts()
```

```
female    812
male      713
Name: gender, dtype: int64
```

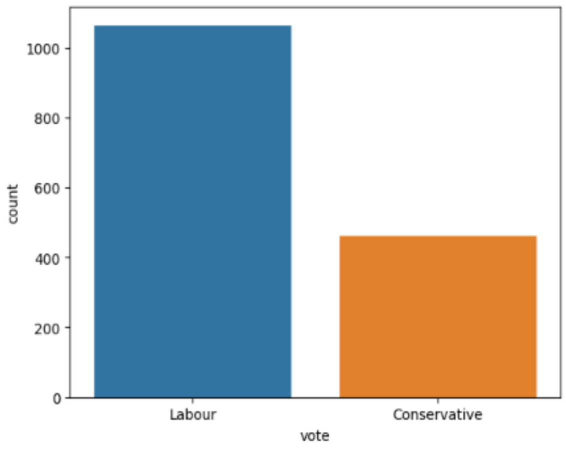
```
df.vote.value_counts()
```

```
Labour          1063
Conservative     462
Name: vote, dtype: int64
```

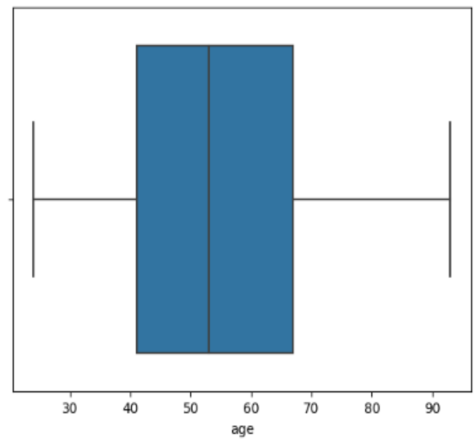
- 812 females & 713 Males taken part in survey
- 1063 votes polled in favour of Labour party and 462 votes in favour of Conservative party.

UNIVARIATE ANALYSIS :

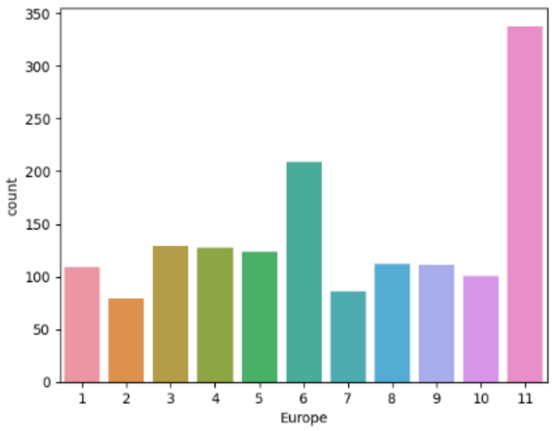
VOTE:



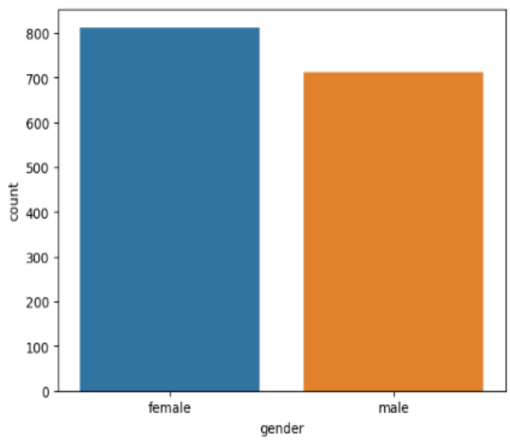
AGE :



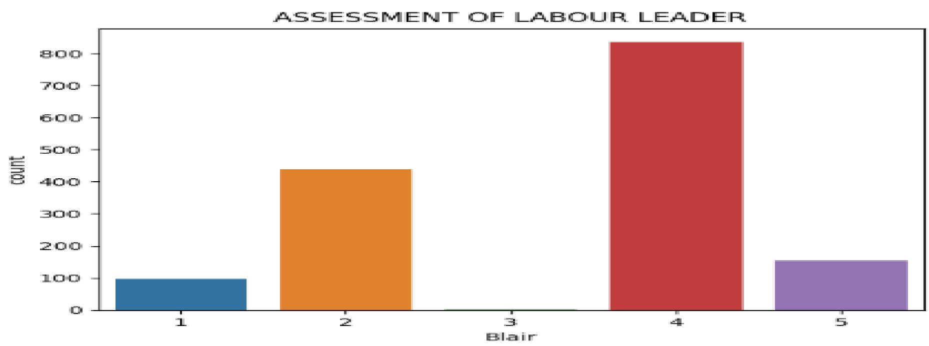
EUROPE :



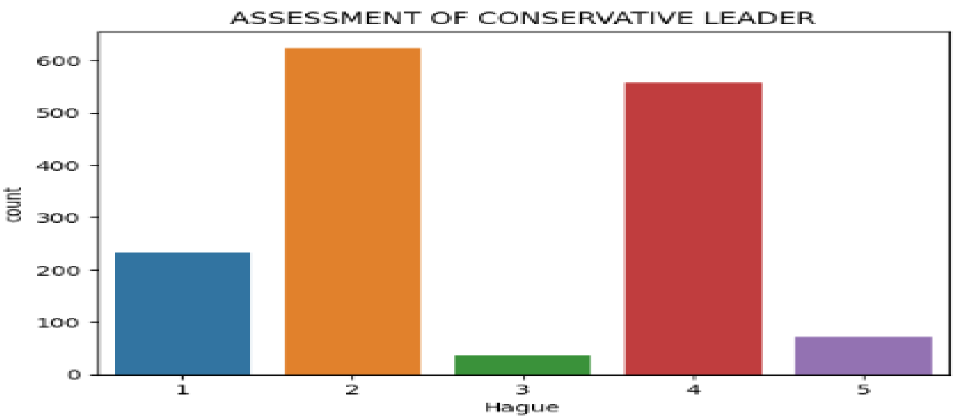
GENDER :



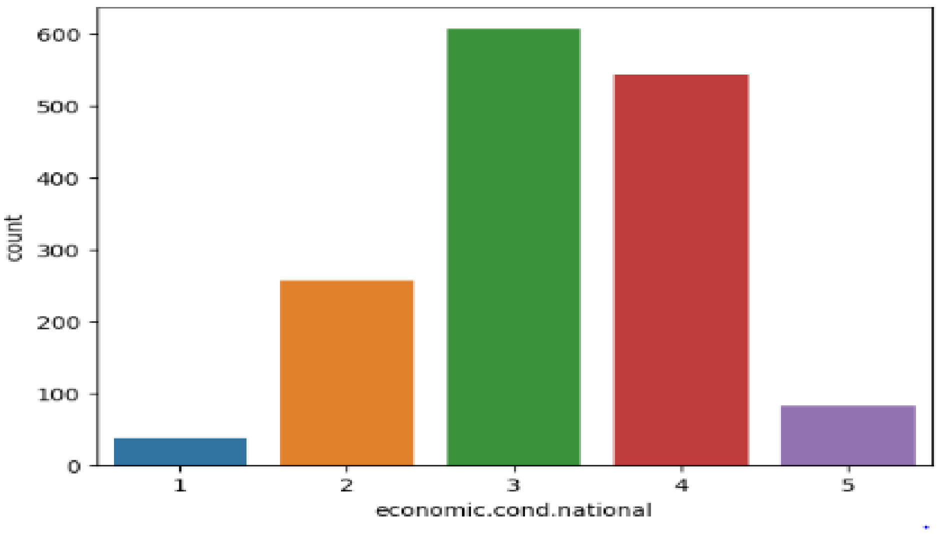
BLAIR :



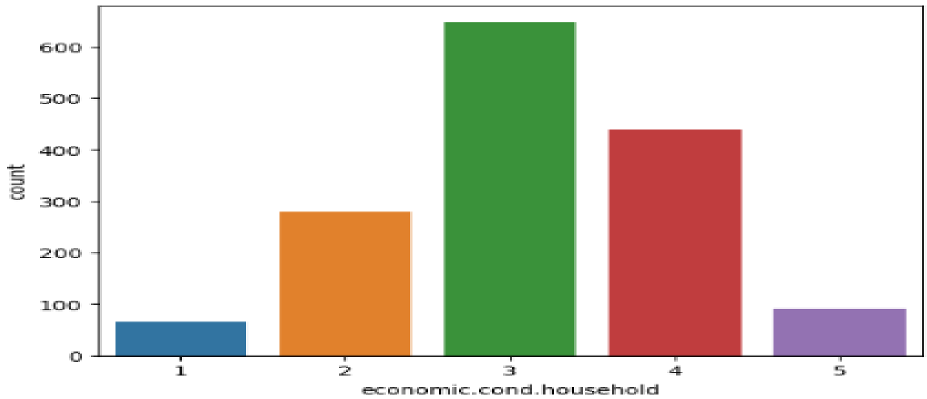
HAGUE :



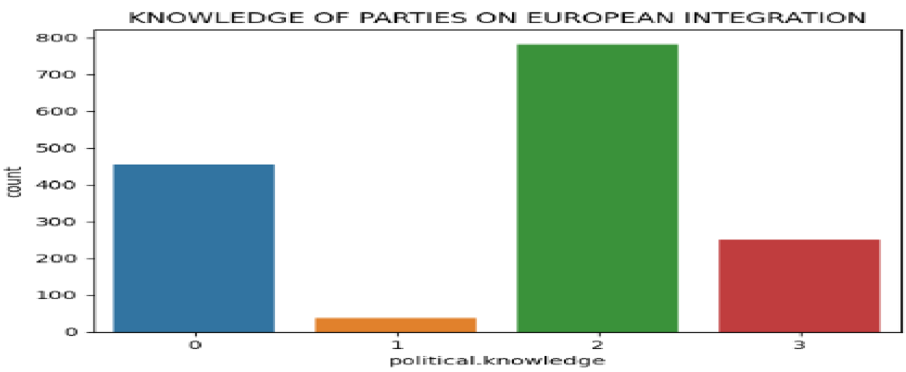
ECONOMIC.CONDN.NATIONAL :



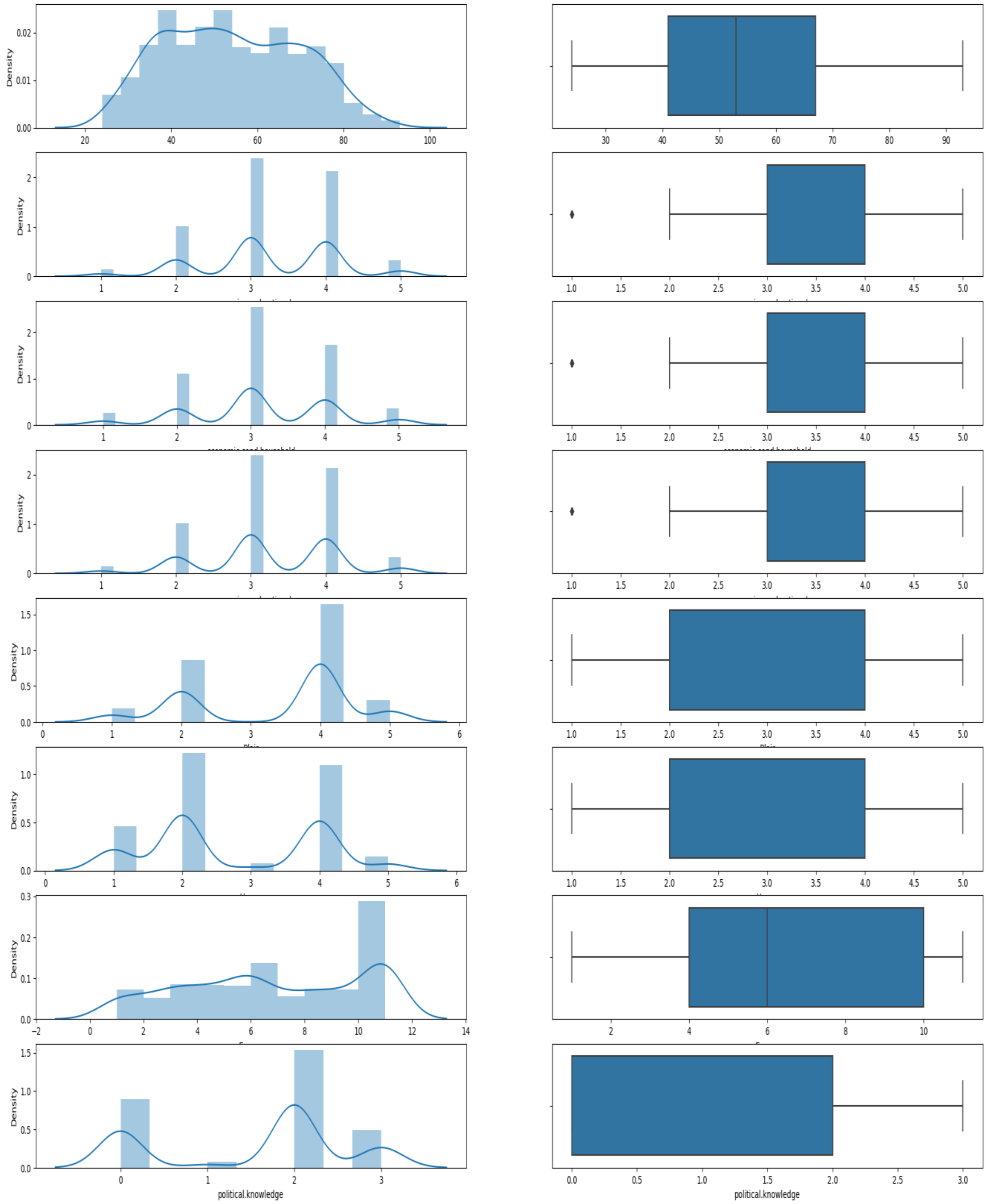
ECONOMIC.CONDN.HOUSEHOLD:



POLITICAL KNOWLEDGE :



Histplot and Boxplot of variables :

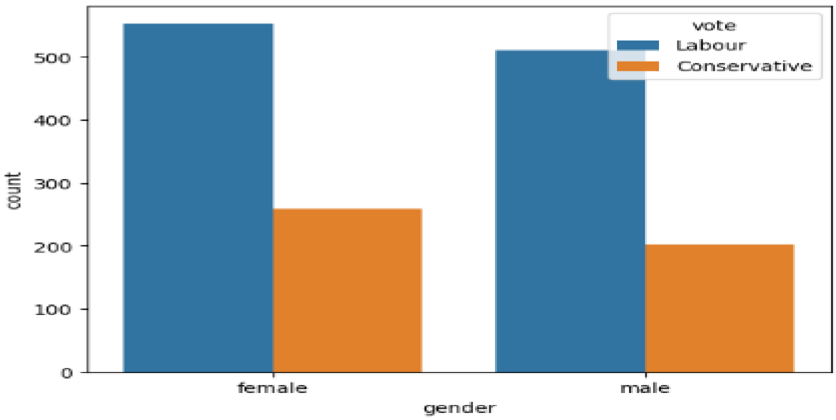


INFERENCES :

- Out of 1525 , 1063 peoples vote in favour of Labour party in UK Election and 460 vote in favour of Conservative party.
- People taken part in the survey are between 24 to 93 age group.
- Out of 1525,338 people strongly support Brexit (Euroseptic) i.e) 22% people.
- Here out of 1525 , 812 Females participated in the survey i.e) 53.2%
- Maximum number of people i.e) 624 provide 2 as highest rating to Conservative party and only 73 provide 5 as rating.
- The average score of economic.condn.national is 3.245.
- The average score of economic.condn.household is 3.137.
- In Blair , Rating 4 is higher than 2 whose value is 434.
- The average political knowledge among 1525 voters is 1.54.
- In Hague , 2 is slightly higher than the 2nd highest variable 4 whose value is 557. The average score of 'Hague' is 2.75.
- In Europe ,11 is moderately higher than the 2nd highest variable 6 whose value is 207.The average score of 'Europe' is 6.740

BIVARIATE ANALYSIS :

GENDER Vs VOTE :




```
vote          gender
Conservative  female    259
              male      203
Labour        female    553
              male      510
Name: gender, dtype: int64
```

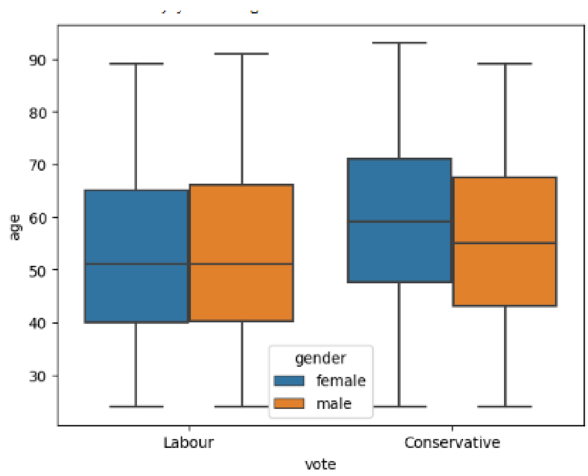
- From above we can say that Labour party has more vote than Conservative party.
- Female votes are more than Male votes.
- Female participation is slightly higher than Male.

VOTE Vs ECONOMIC.CONDN.NATIONAL :

```
vote          economic.cond.national
Conservative  3                          200
              2                          140
              4                           92
              1                           21
              5                            9
Labour        4                          450
              3                          407
              2                          117
              5                           73
              1                           16
Name: economic.cond.national, dtype: int64
```

- Labour party has higher votes.
- 82 people give a score of 5.Among them,73 voted for Labour party
- 542 people gave a score of 4.Among them 450 voted for Labour party.
- 607 people gave a score of 3.Among them,407 people voted for Labour party.
- 257 people gave a score of 2.Among them,117 people voted for Labour party
- 17 people gave a score of 1. Among them , 16 voted for Labour party.

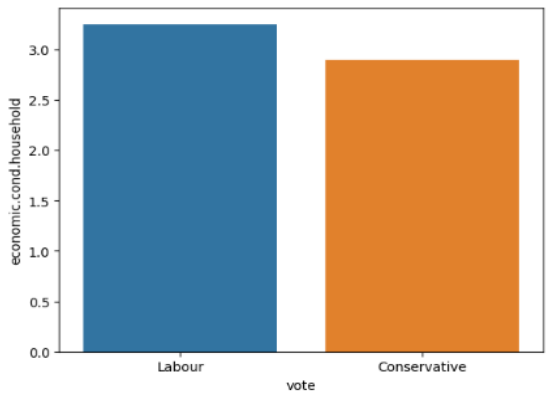
Vote Vs Age :




```
vote          gender
Conservative  female    259
              male      203
Labour        female    553
              male      510
Name: gender, dtype: int64
```

- In every age group, labour party got more votes than conservative party.
- In both the genders, labour party got more votes than conservative party.

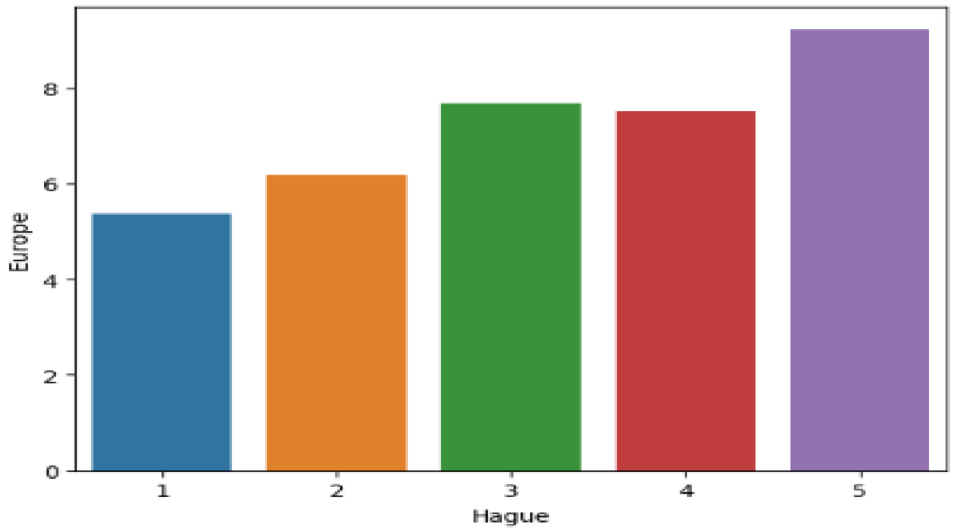
Vote vs Economic.cond.household :




```
vote          economic.cond.household
Conservative  3                          198
              2                          126
              4                           87
              1                           28
              5                           23
Labour        3                          450
              4                          353
              2                          154
              5                           69
              1                           37
Name: economic.cond.household, dtype: int64
```

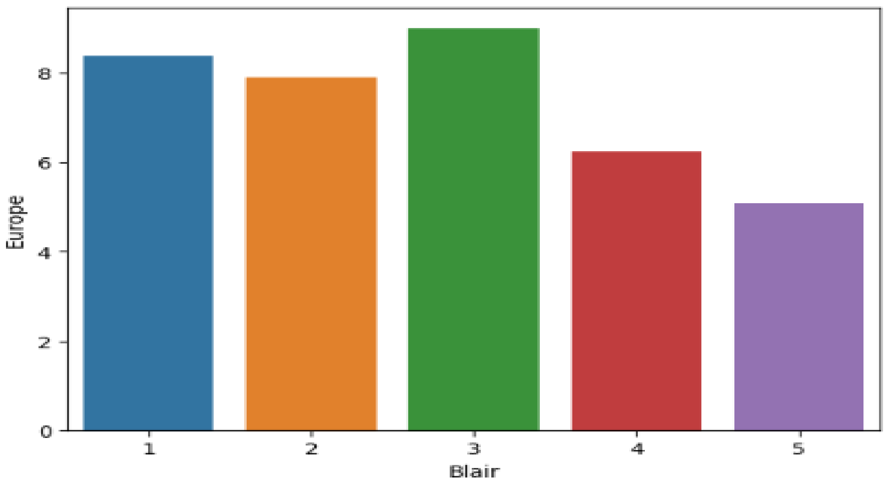
- On the whole, Labour party has got more votes than Conservative party.
- Out of 92 people who gave a score of 5, 69 people voted for labour party.
- Out of 65 people who gave a score of 1, 37 voted for the labour party and 28 voted for the conservative party.

HAGUE Vs EUROPE :



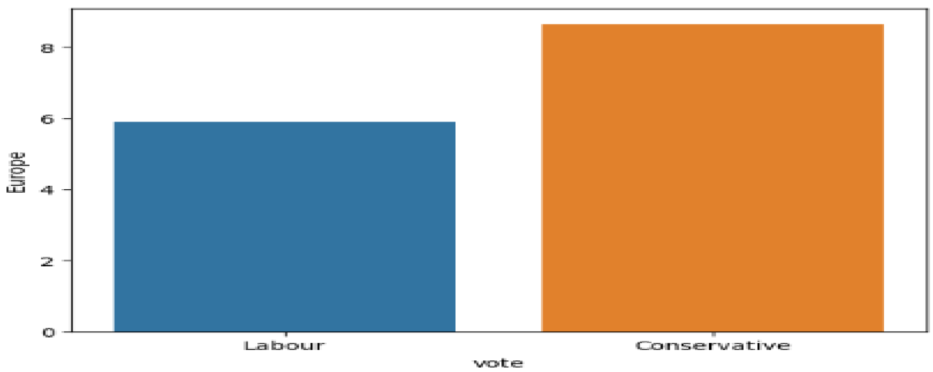
Those who strongly support (HAGUE = 5) Conservative party provide a maximum of 10 points for Brexit. so we conclude that conservative party supporters favours brexit more than labour party supporters.

Europe vs Hague :



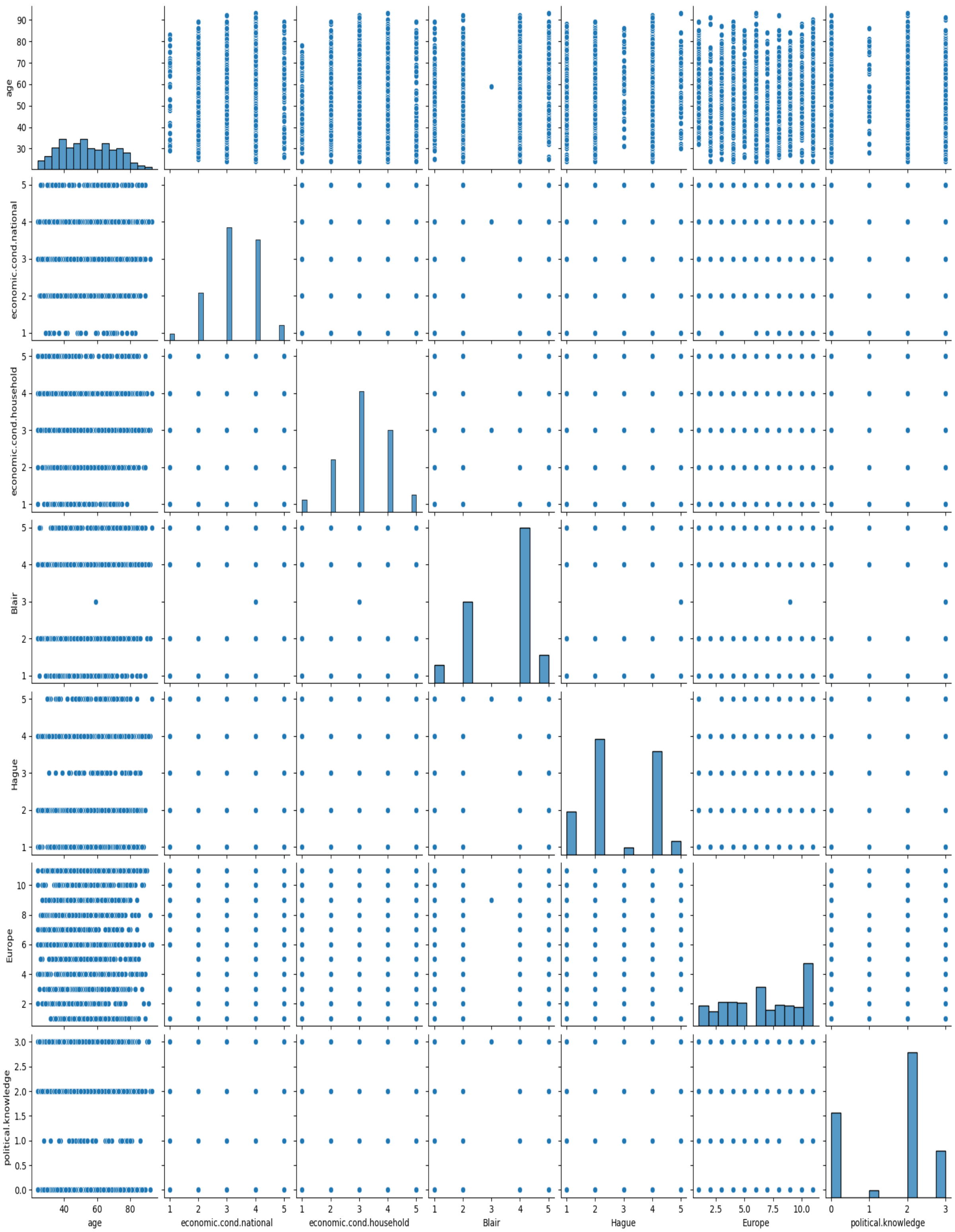
Those who strongly support (blair = 5) labour party provide only a max of 5 point for brexit.

Vote vs Europe :



Out of 338 people who gave a score of 11, 166 people voted for the labour party and 172 people have voted for the conservative party.

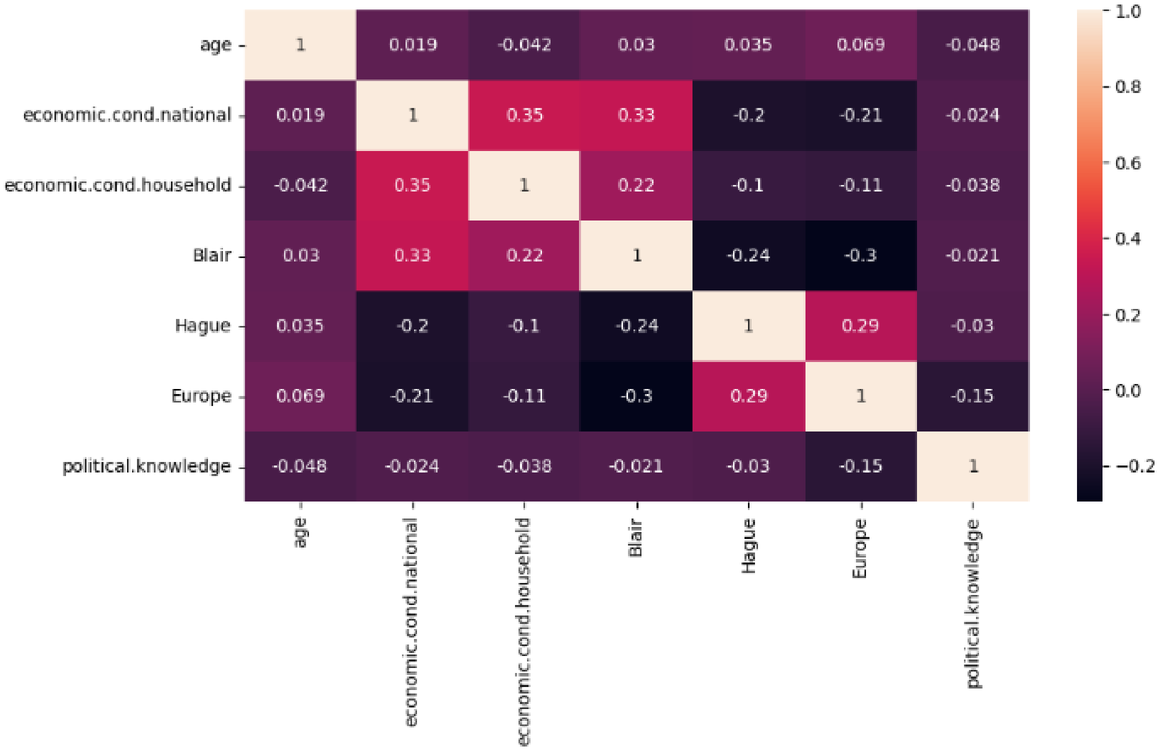
PAIRPOLOT :



age
economic.cond.national
economic.cond.household
Blair
Hague
Europe
political.knowledge

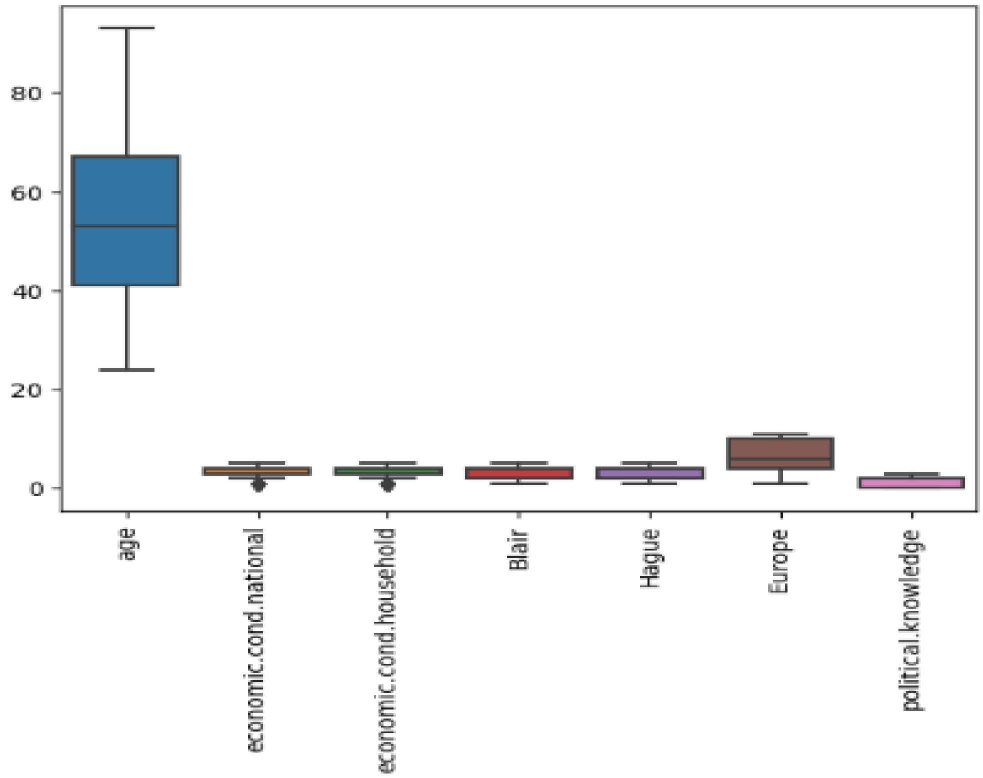
- Blair, Europe and political.knowledge' variables are slightly left skewed.
- All other variables seem to be normally distributed.
- Also we can see that, there is mostly no correlation between the variables.

HEATMAP :



- 'economic.cond.national' with 'economic.cond.household' have moderate positive correlation.
- 'Blair' with 'economic.cond.national' and 'economic.cond.household' have moderate positive correlation.
- 'Europe' with 'Hague' have moderate positive correlation.
- 'Hague' with 'economic.cond.national' and 'Blair' have moderate negative correlation.
- 'Europe' with 'economic.cond.national' and 'Blair' have moderate negative correlation.

OUTLIERS :



Presence of Outliers only in economic.cond.national & economic.cond.household. It wont affect further proceedings so no need to treat it.

	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

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

DATA AFTER ENCODING :

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

Here we have encoded the vote column and gender column.Here in vote_Labour =1 means the voter votes in favour of Labour party and gender_male=1 means the person is Male.

DATA INFO:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1525 entries, 0 to 1524
Data columns (total 9 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   age                      1525 non-null   int64
 1   economic.cond.national   1525 non-null   int64
 2   economic.cond.household  1525 non-null   int64
 3   Blair                    1525 non-null   int64
 4   Hague                    1525 non-null   int64
 5   Europe                   1525 non-null   int64
 6   political.knowledge      1525 non-null   int64
 7   vote_Labour              1525 non-null   uint8
 8   gender_male              1525 non-null   uint8
dtypes: int64(7), uint8(2)
memory usage: 86.5 KB
```

SCALING :

The data contains features varying in magnitudes, units and range between the 'age' column and other columns. We need to bring all features to the same level of magnitudes. This can be acheived by scaling.

Here we use Min Max sacling method , Data after scaling :

	count	mean	std	min	25%	50%	75%	max
economic.cond.household	1525.0	0.535082	0.232488	0.0	0.50	0.500000	0.750000	1.0
economic.cond.national	1525.0	0.561475	0.220242	0.0	0.50	0.500000	0.750000	1.0
Blair	1525.0	0.583607	0.293706	0.0	0.25	0.750000	0.750000	1.0
Hague	1525.0	0.436721	0.307676	0.0	0.25	0.250000	0.750000	1.0
Europe	1525.0	0.572852	0.329754	0.0	0.30	0.500000	0.900000	1.0
political.knowledge	1525.0	0.514098	0.361105	0.0	0.00	0.666667	0.666667	1.0

SPLIT THE DATA :

```
x=election.drop('vote_Labour',axis=1)
y=election.pop('vote_Labour')
```

```
x_train,x_test , y_train, y_test = train_test_split(x,y,test_size = .30 ,random_state = 1)
```

Here split the data into 70:30 ratio with random_state =1 .

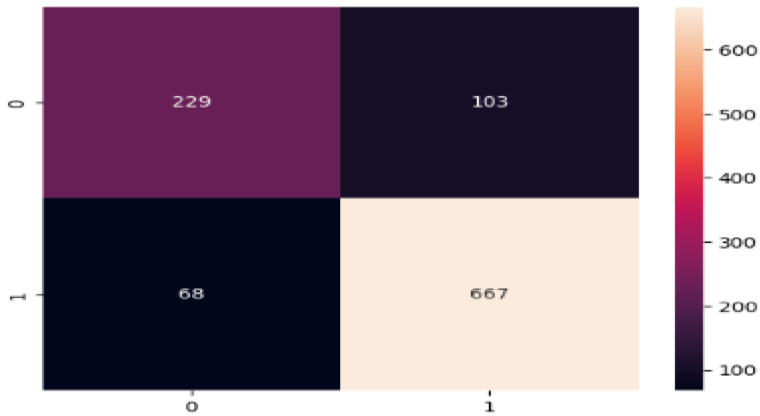
1.4) Apply Logistic Regression and LDA (Linear Discriminant Analysis) (2 pts). Interpret the inferences of both models (2 pts). Successful implementation of each model. Logical reason should be shared if any custom changes are made to the parameters while building the model. Calculate Train and Test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

LOGISTIC REGRESSION :

Classification report : Train data

	precision	recall	f1-score	support
0	0.77	0.69	0.73	332
1	0.87	0.91	0.89	735
accuracy			0.84	1067
macro avg	0.82	0.80	0.81	1067
weighted avg	0.84	0.84	0.84	1067

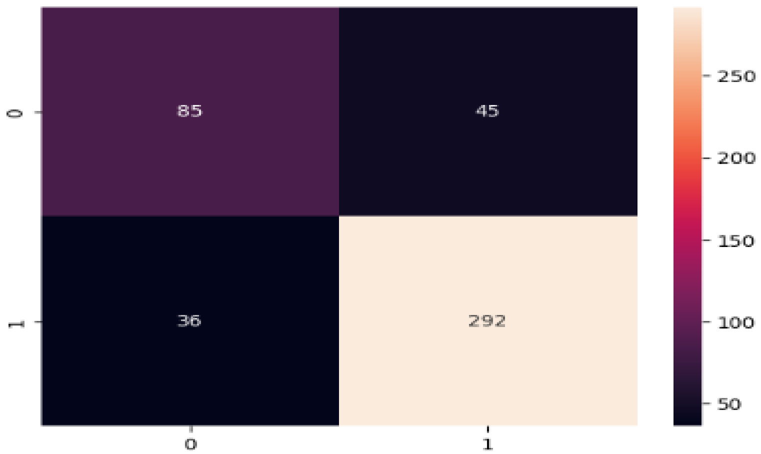
Confusion Matrix : Train data



Classification Report : Test data

	precision	recall	f1-score	support
0	0.70	0.65	0.68	130
1	0.87	0.89	0.88	328
accuracy			0.82	458
macro avg	0.78	0.77	0.78	458
weighted avg	0.82	0.82	0.82	458

Confusion matrix : Test data



OBSERVATION :

- ACCURACY :
  - Train Data : 84 %
  - Test Data : 82 %
- The error in the test data is slightly higher than the train data, which is absolutely fine because the error margin is low and the error in both train and test data is not too high. Thus, the model is not over-fitted or under-fitted.

LINEAR DISCRIMINANT ANALYSIS :

Train Data :

Classification Report :

	precision	recall	f1-score	support
0	0.70	0.76	0.73	308
1	0.90	0.87	0.88	759
accuracy			0.84	1067
macro avg	0.80	0.81	0.81	1067
weighted avg	0.84	0.84	0.84	1067

Confusion Matrix : Train Data

[[233 75]
[99 660]]

Test Data :

Classification Report :

	precision	recall	f1-score	support
0	0.66	0.69	0.67	125
1	0.88	0.87	0.87	333
accuracy			0.82	458
macro avg	0.77	0.78	0.77	458
weighted avg	0.82	0.82	0.82	458

Confusion Matrix :

[[86 39]
[44 289]]

OBSERVATION REPORT :

- ACCURACY :
  - Train Data : 84%
  - Test Data : 82%
- The model is not over-fitted or under-fitted. The error in the test data is slightly higher than the train data, which is absolutely fine because the error margin is low and the error in both train and test data is not too high. Thus, the model is not over-fitted or under-fitted.

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

<u>KNN MODEL:</u>

Train Data :

Classification Report :

	precision	recall	f1-score	support
0	0.79	0.73	0.76	332
1	0.88	0.91	0.90	735
accuracy			0.86	1067
macro avg	0.84	0.82	0.83	1067
weighted avg	0.86	0.86	0.86	1067

Confusion Matrix :

```
[[244  88]
 [ 63 672]]
```

Test Data :

Classification Report :

	precision	recall	f1-score	support
0	0.61	0.62	0.62	130
1	0.85	0.84	0.85	328
accuracy			0.78	458
macro avg	0.73	0.73	0.73	458
weighted avg	0.78	0.78	0.78	458

Confusion Matrix :

```
[[ 81  49]
 [ 51 277]]
```

Observation report :

- Accuracy :
  - Train Data : 86 %
  - Test Data : 78 %
- Here we take K value as 5.
- As we can see, the train data has a 86% accuracy and test data has 78% accuracy. The difference is more than 5%. So, we can infer that the KNN model is over-fitted.

NAÏVE BAYES MODEL :

Train Data :

Classification Report :

	precision	recall	f1-score	support
0	0.74	0.72	0.73	332
1	0.88	0.88	0.88	735
accuracy			0.83	1067
macro avg	0.81	0.80	0.80	1067
weighted avg	0.83	0.83	0.83	1067

Confusion Matrix :

```
[[240  92]
 [ 86 649]]
```

Test Data :

Classification Report :

	precision	recall	f1-score	support
0	0.68	0.72	0.70	130
1	0.89	0.87	0.88	328
accuracy			0.83	458
macro avg	0.78	0.79	0.79	458
weighted avg	0.83	0.83	0.83	458

Confusion Matrix :

```
[[ 94  36]
 [ 44 284]]
```

Obseravation Report :

- ACCURACY :
  - Train Data : 83.31%
  - Test Data : 82.53 %
- The model is not over-fitted or under-fitted. The error in the test data is slightly higher than the train data, which is absolutely fine because the error margin is low and the error in both train and test data is not too high. Thus, the model is not over-fitted or under-fitted.

BAGGING MODEL :

DECISION TREE :

Train Data :

Classification Report :

	precision	recall	f1-score	support
0	1.00	1.00	1.00	332
1	1.00	1.00	1.00	735
accuracy			1.00	1067
macro avg	1.00	1.00	1.00	1067
weighted avg	1.00	1.00	1.00	1067

Confusion Matrix :

```
[[331   1]
 [  0 735]]
```

Test Data :

Classification Report :

	precision	recall	f1-score	support
0	0.64	0.64	0.64	130
1	0.86	0.86	0.86	328
accuracy			0.80	458
macro avg	0.75	0.75	0.75	458
weighted avg	0.80	0.80	0.80	458

Confusion Matrix :

```
[[ 83  47]
 [ 46 282]]
```

OBSERVATION :

- ACCURACY :
    - Train Data : 100%
    - Test Data : 80%
- Here in Train Dateset, the model is over-fitted.In Train dataset,the accuracy is 100% and test data accuracy is 80%. The difference is more than 10%. So, we can infer that the Decision tree model is over-fitted.
- After using Bagging model,we still getting the model as over fitted.

RANDOM FOREST :

```
GridSearchCV
estimator: RandomForestClassifier
RandomForestClassifier
```

```
RandomForestClassifier
RandomForestClassifier(max_depth=10, min_samples_leaf=25, min_samples_split=30,
                       random_state=0)
```

Train Data :

Classification Report :

	precision	recall	f1-score	support
0	0.81	0.63	0.71	332
1	0.85	0.93	0.89	735
accuracy			0.84	1067
macro avg	0.83	0.78	0.80	1067
weighted avg	0.84	0.84	0.83	1067

Confusion Matrix:

```
[[208 124]
 [ 48 687]]
```

Test Data :

Classification Report :

	precision	recall	f1-score	support
0	0.75	0.63	0.68	130
1	0.86	0.91	0.89	328
accuracy			0.83	458
macro avg	0.80	0.77	0.79	458
weighted avg	0.83	0.83	0.83	458

Confusion Matrix :

```
[[ 82  48]
 [ 28 300]]
```

OBSERVATION :

- ACCURACY :
  - Train Data : 84%
  - Test Data : 83%
- The model is not over-fitted or under-fitted. The error in the test data is slightly higher than the train data, which is absolutely fine because the error margin is low and the error in both train and test data is not too high. Thus, the model is not over-fitted or under-fitted.

BOOSTING :

ADABOOSTING :

```
GridSearchCV
estimator: AdaBoostClassifier
AdaBoostClassifier
```

```
AdaBoostClassifier
AdaBoostClassifier(algorithm='SAMME', learning_rate=0.1, n_estimators=1000)
```

- ACCURACY :
    - Train Data : 84%
    - Test Data : 83%
- The model is not over-fitted. The values are good. Therefore, the model is a good model.

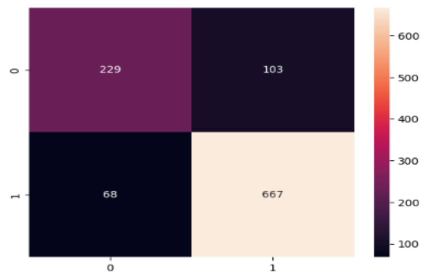
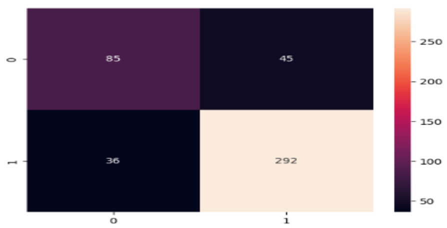
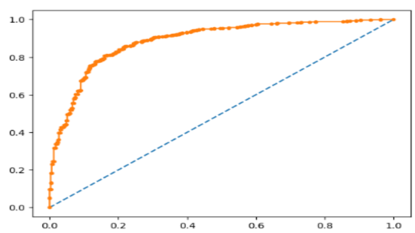
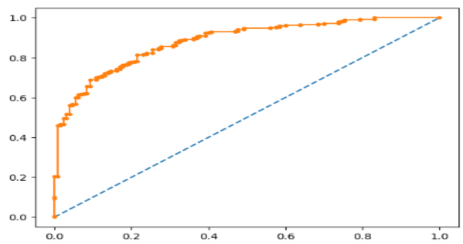
GRADIENT BOOSTING :

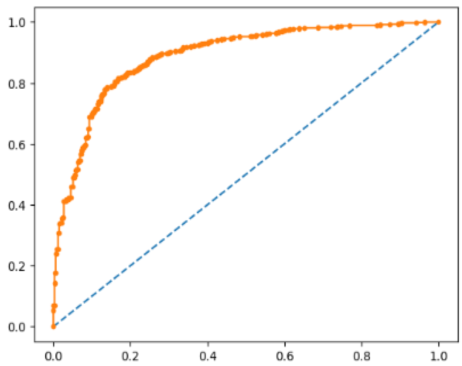
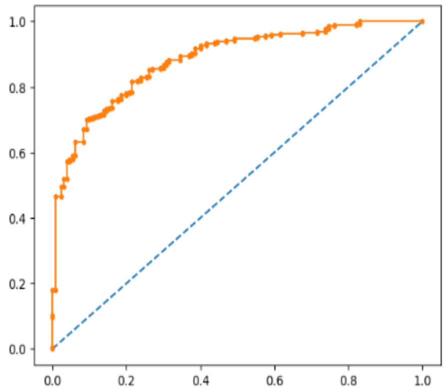
```
GradientBoostingClassifier
GradientBoostingClassifier(random_state=1)
```

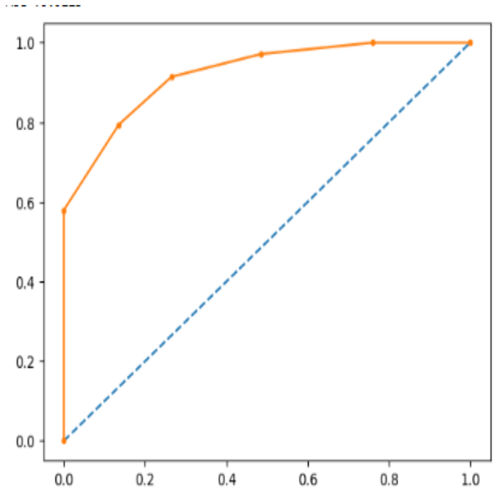
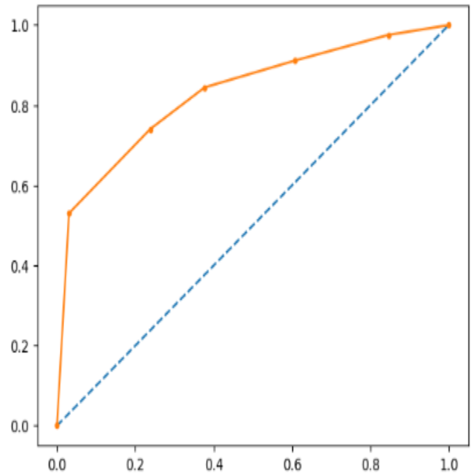
```
DecisionTreeClassifier
DecisionTreeClassifier(max_depth=5, min_samples_leaf=15, min_samples_split=70,
                       random_state=0)
```

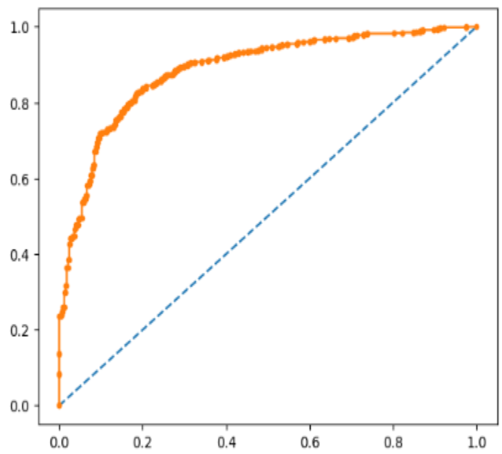
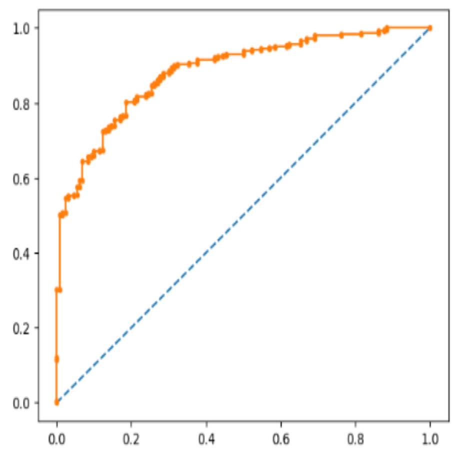
- ACCURACY :
    - Train Data : 89%
    - Test Data : 81%
- The model is not over-fitted. The values are better than AdaBoosting model. The model is a good model.On the whole,this is a good model.

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, classification report (4 pts) Final Model - Compare and comment on all models on the basis of the performance metrics in a structured tabular manner. Describe on which model is best/optimized, After comparison which model suits the best for the problem in hand on the basis of different measures. Comment on the final model.(3 pts)

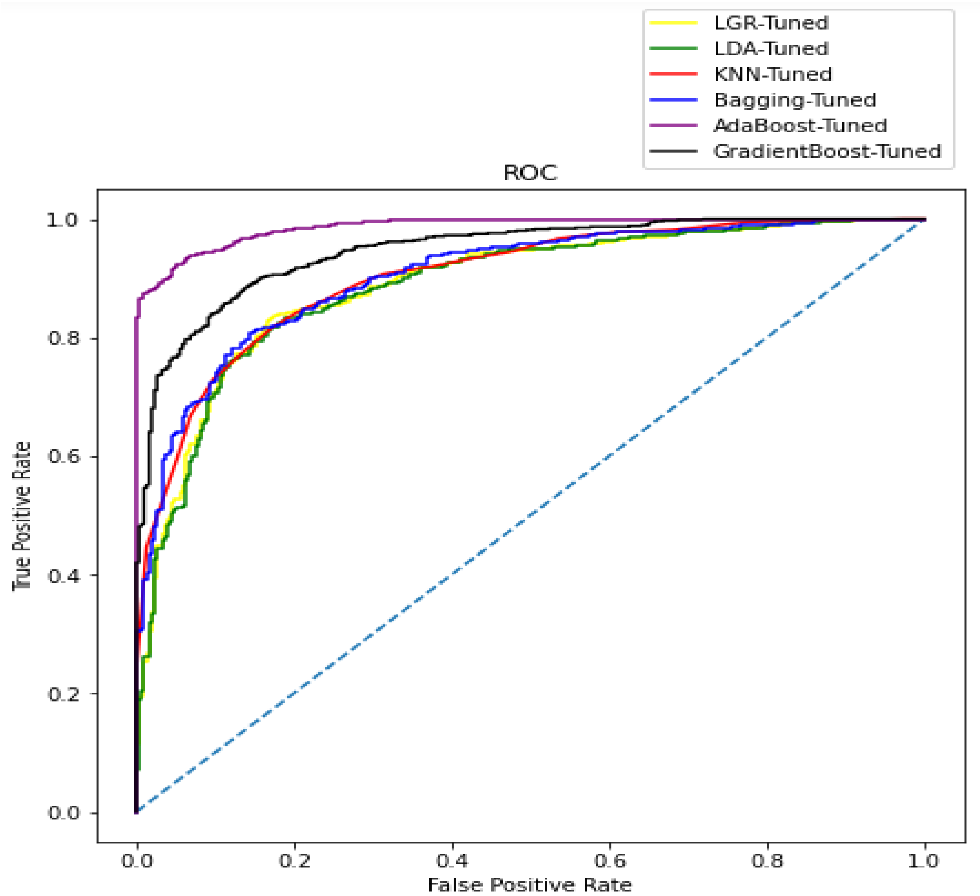
	Performance Metrics	Train Data	Test Data
LOGISTIC REGRESSION	Accuracy	84%	82%
	Confusion Matrix		
	ROC CURVE		
	AUC SCORE	88.94 %	88.25 %
	Classification Report	precision recall f1-score support 0 0.77 0.69 0.73 332 1 0.87 0.91 0.89 735 accuracy 0.84 1067 macro avg 0.82 0.80 0.81 1067 weighted avg 0.84 0.84 0.84 1067	precision recall f1-score support 0 0.70 0.65 0.68 130 1 0.87 0.89 0.88 328 accuracy 0.82 458 macro avg 0.78 0.77 0.78 458 weighted avg 0.82 0.82 0.82 458
	F1 SCORE	89%	88%

	Perfomance Metrics	Train Data	Test Data
	Accuracy	84%	82%
LDA	Confusion Matrix	[[233 75] [99 660]]	[[86 39] [44 289]]
	ROC Curve		
	AUC Score	88.9%	88.38 %
	Classification Report	precision recall f1-score support 0 0.70 0.76 0.73 308 1 0.90 0.87 0.88 759 accuracy 0.84 1067 macro avg 0.80 0.81 0.81 1067 weighted avg 0.84 0.84 0.84 1067	precision recall f1-score support 0 0.66 0.69 0.67 125 1 0.88 0.87 0.87 333 accuracy 0.82 458 macro avg 0.77 0.78 0.77 458 weighted avg 0.82 0.82 0.82 458
	F1 SCORE	88%	87%

	Performnace Metrics	Train Data	Test Data
	Accuracy	86 %	78%
	Confusion Matrix	[[244 88] [63 672]]	[[81 49] [51 277]]
KNN	ROC Curve		
	AUC Score	92.2%	82.9 %
	Classification Report	precision recall f1-score support 0 0.79 0.73 0.76 332 1 0.88 0.91 0.90 735 accuracy 0.86 1067 macro avg 0.84 0.82 0.83 1067 weighted avg 0.86 0.86 0.86 1067	precision recall f1-score support 0 0.61 0.62 0.62 130 1 0.85 0.84 0.85 328 accuracy 0.78 458 macro avg 0.73 0.73 0.73 458 weighted avg 0.78 0.78 0.78 458
	F1 SCORE	90%	85%

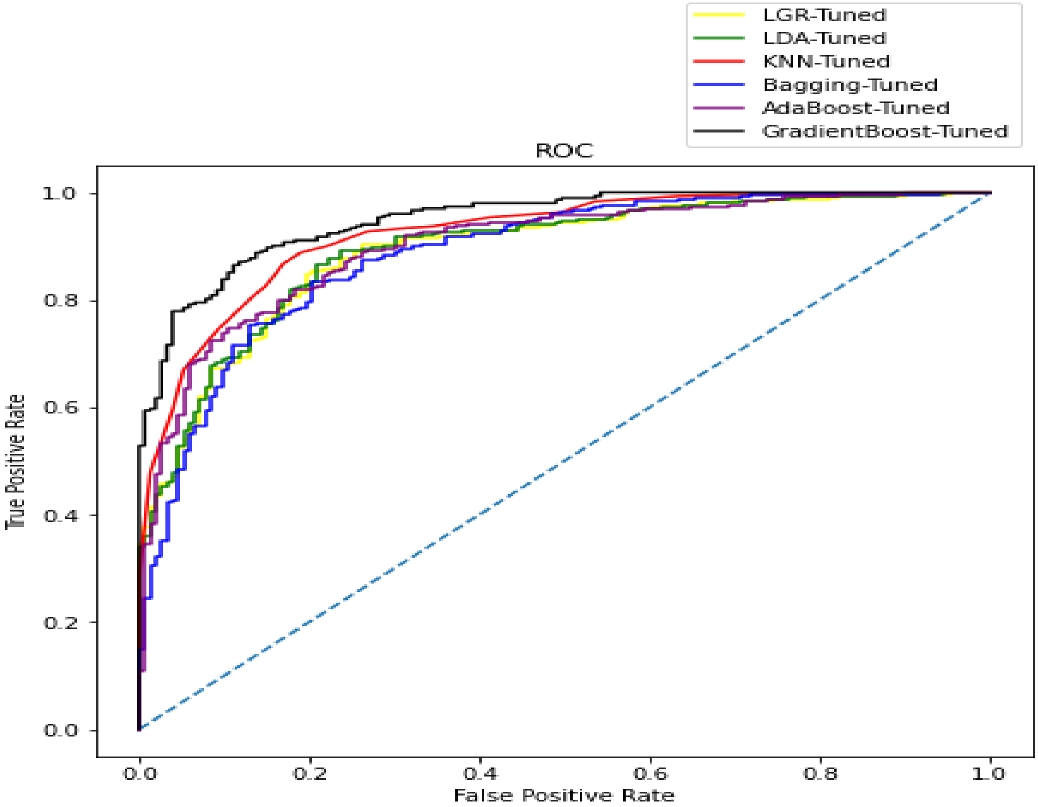
	Performance Metrics	Train Data	Test Data
	Accuracy	83.3%	82.5%
	Confusion Matrix	[[240 92] [86 649]]	[[94 36] [44 284]]
NAÏVE BAYES	ROC CURVE		
	AUC score	88.65 %	88.45 %
	Classification Report	precision recall f1-score support 0 0.74 0.72 0.73 332 1 0.88 0.88 0.88 735 accuracy 0.83 1067 macro avg 0.81 0.80 0.80 1067 weighted avg 0.83 0.83 0.83 1067	precision recall f1-score support 0 0.68 0.72 0.70 130 1 0.89 0.87 0.88 328 accuracy 0.83 458 macro avg 0.78 0.79 0.79 458 weighted avg 0.83 0.83 0.83 458
	F1 score	88%	88%

ROC CURVE FOR TRAINED DATA OF ALL MODELS :



The tuning of the Gradient Boost model has improved the model further. The values are high. The better is better than the regular model.

ROC MODEL FOR ALL TEST DATA :



In all the models, tuned ones are better than the regular models. So, we compare only the tuned models and describe which model is the best/optimized.

Conclusion :

- There is no under-fitting or over-fitting in any of the tuned models.

- All the tuned models have high values and every model is good.

  But as we can see, the most consistent tuned model in both train and test data is the Gradient Boost model.

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

- Labour party has more than double the votes of conservative party.
- Most number of people have given a score of 3 and 4 for the national economic condition and the average score is 3.24
- Blair has higher number of votes than Hague and the scores are much better for Blair than for Hague.
- On a scale of 0 to 3, about 30% of the total population has zero knowledge about politics.
- People who gave a low score of 1 to a certain party, still decided to vote for the same party instead of voting for the other party. This can be because of lack of political knowledge among the people.
- People who have higher Eurosceptic sentiment, has voted for the conservative party and lower the Eurosceptic sentiment, higher the votes for Labour party.
- All models performed well on training data set as well as test dataset. The tuned models have performed better than the regular models.
- There is no over-fitting in any model except Random Forest and Bagging regular models
- Gradient Boosting model tuned is the best/optimized model

RECOMENDATION :

- Gathering more data will also help in training the models and thus improving the predictive powers
- Using Gradient Boosting model without scaling for predicting the outcome as it has the best optimized performance
- We can also create a function in which all the models predict the outcome in sequence. This will helps in better understanding and the probability of what the outcome will be.
- We can conclude that Labour party has more votes in the election from the given dataset because they got support due to Brexit and improvement in economic conditions of Nation and Household.

THANK YOU