Priyadharshini K

Project Report - Machine Learning

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.

Dataset for Problem: Election_Data.xlsx
Data Ingestion: 11 marks

1.1 Read the dataset. Do the descriptive statistics and do the null value condition check. Write an inference on it. (4 Marks)
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.

Summary Statistics of the dataset

```
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)
```

Null Value Check

```
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
```

Inferences:

- The dataset has a total of 8 dependent variables and 1 independent variable. The dataset has both categorical, continuous and discrete data.
- Shape of the dataset: (1525,10)
- No duplicate values are present in the dataset
- Object Datatype variables – Vote, Gender
- Int Datatype variables Continuous variable – Unnamed: 0, age
- Int Datatype variables Discrete variable – economic.cond.national, economic.cond.household, Blair, Hague, Europe, political.knowledge
- Null values/missing values are not present in the dataset

	Unnamed: 0	age	economic.cond.national	economic.cond.household	Blair	Hague	Europe	political.knowledge
count	1525.000000	1525.000000	1525.000000	1525.000000	1525.000000	1525.000000	1525.000000	1525.000000
mean	763.000000	54.182295	3.245902	3.140328	3.334426	2.746885	6.728525	1.542295
std	440.373894	15.711209	0.880969	0.929951	1.174824	1.230703	3.297538	1.083315
min	1.000000	24.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.000000
25%	382.000000	41.000000	3.000000	3.000000	2.000000	2.000000	4.000000	0.000000
50%	763.000000	53.000000	3.000000	3.000000	4.000000	2.000000	6.000000	2.000000
75%	1144.000000	67.000000	4.000000	4.000000	4.000000	4.000000	10.000000	2.000000
max	1525.000000	93.000000	5.000000	5.000000	5.000000	5.000000	11.000000	3.000000

Summary statistics

- The mean age group of voters is 54 years, minimum and maximum age being 24 and 93 respectively.
- Assessment of current household economic condition is on par with the assessment of current national economic conditions.
- Labor leader Blair fares well in comparison to the Conservative Leader Hague.
- People have a mixed feeling towards European integration.
- Knowledge of parties' positions on European integration also has got a mean value of 1.5 indicates a thorough knowledge is not present with either of the parties.

Skewness:

Skewness of age : 0.14447848346551462
Skewness of economic.cond.national : -0.2402163142518291
Skewness of economic.cond.household : -0.14940490939119963
Skewness of Blair : -0.5348918666133158
Skewness of Hague : 0.15194998016716968
Skewness of Europe : -0.13581295528712456
Skewness of political.knowledge : -0.4264178682034399

Skewness assesses the extent to which a variable's distribution is symmetrical. Skewness of the variables present in the dataset is between -0.5 to 0.5, indicates that the data is fairly symmetrical.

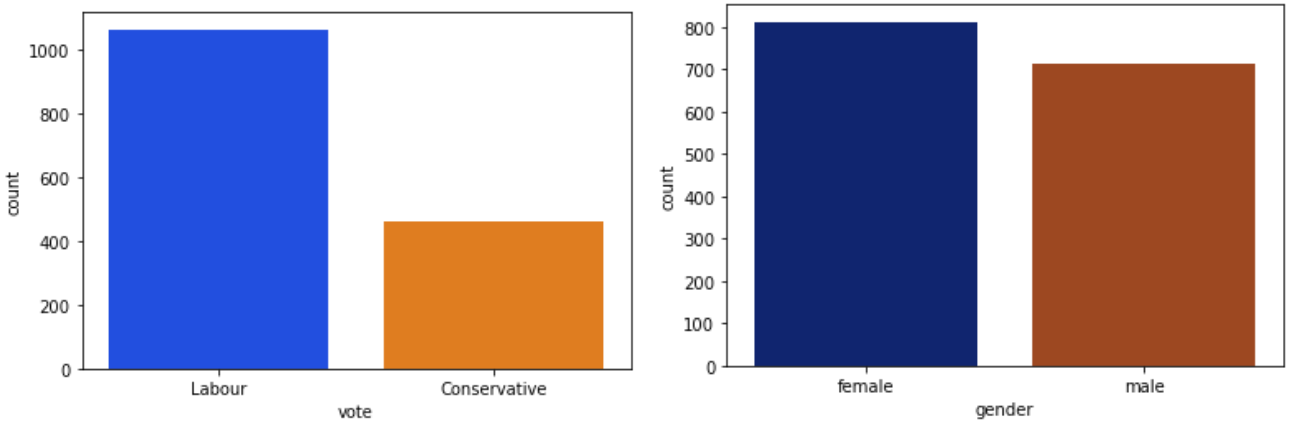
Kurtosis:

Kurtosis of age : -0.9477269632496834
Kurtosis of economic.cond.national : -0.2590870832450891
Kurtosis of economic.cond.household : -0.20955763368399438
Kurtosis of Blair : -1.0660228870170225
Kurtosis of Hague : -1.3911156666526612
Kurtosis of Europe : -1.237717874488492
Kurtosis of political.knowledge : -1.2165924068179326

Kurtosis is a measure of whether the distribution is too peaked i.e., a very narrow distribution with most of the responses in the center. The kurtosis of the continuous variables present in the dataset is less than zero, indicates that the distribution is light tailed.

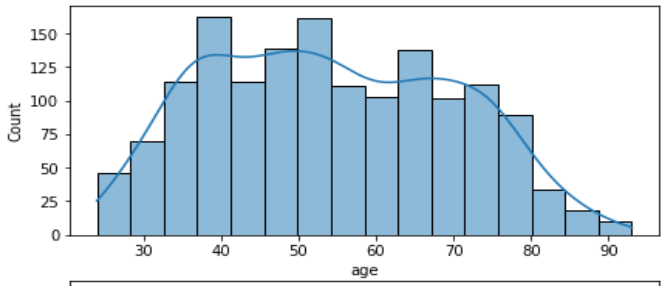
1.2 Perform Univariate and Bivariate Analysis. Do exploratory data analysis. Check for Outliers. Perform EDA (Check the null values, Data types, shape, Univariate, bivariate analysis). Also check for outliers (4 pts). Interpret the inferences for each (3 pts) Distribution plots(histogram) or similar plots for the continuous columns. Box plots, Correlation 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.

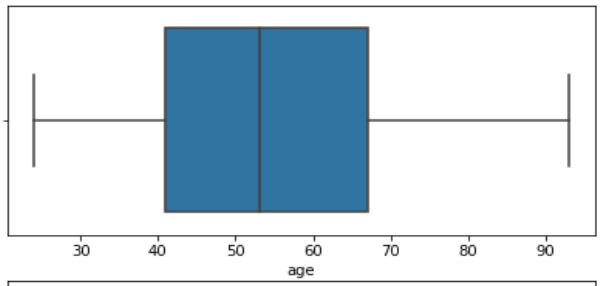
Univariate Analysis

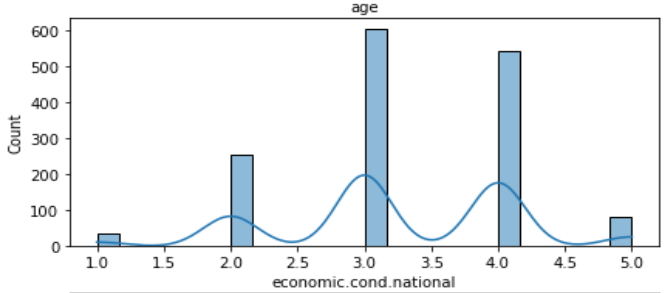


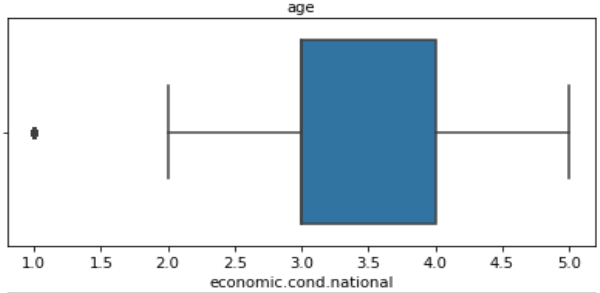
- Labour party has got 69.7 % vote and Conservative party has got 30 % votes, proving majority to Labor party

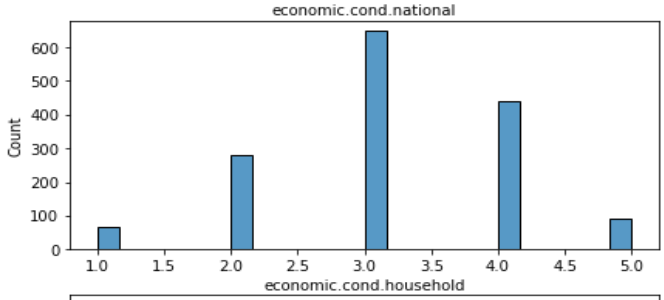
- Percentage of female voters (53.2%) are more than the percentage of male voters (46.7%).

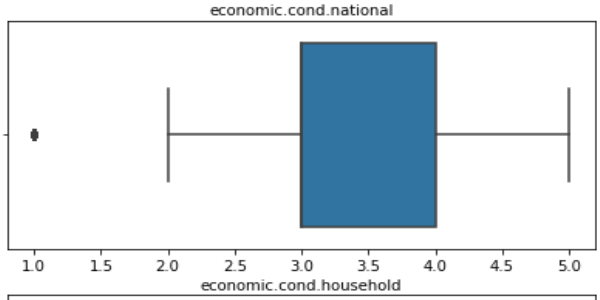

Count
age

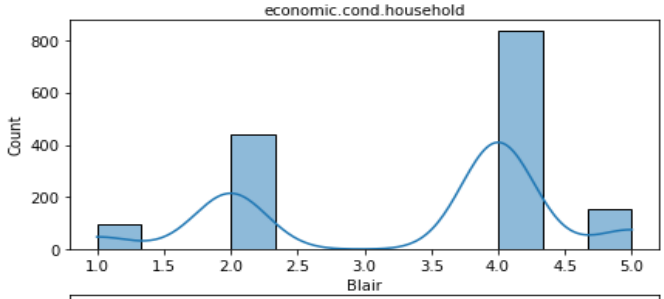

age

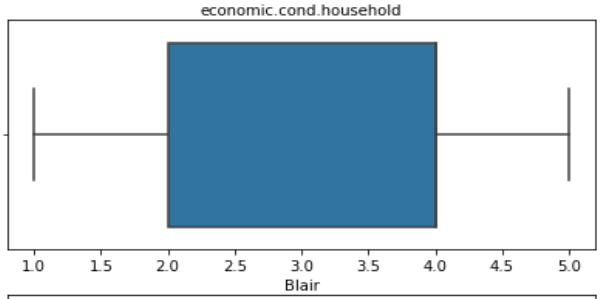

Count
economic.cond.national

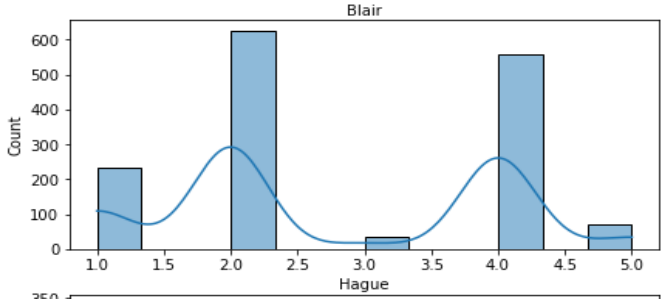

economic.cond.national

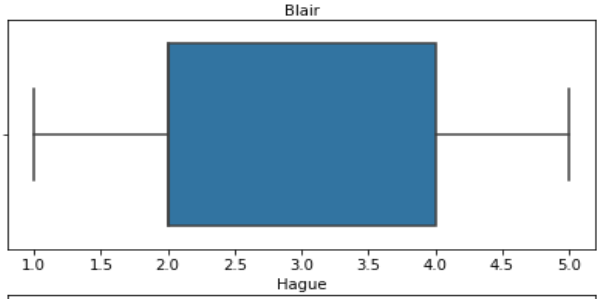

Count
economic.cond.household

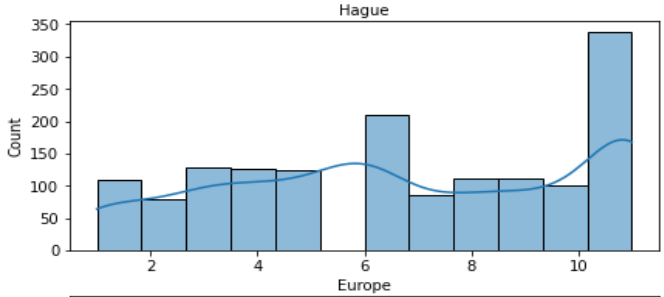

economic.cond.household

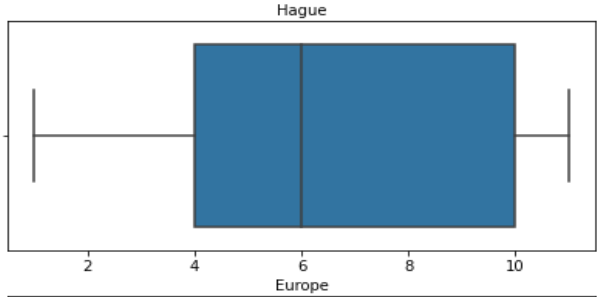

Count
Blair

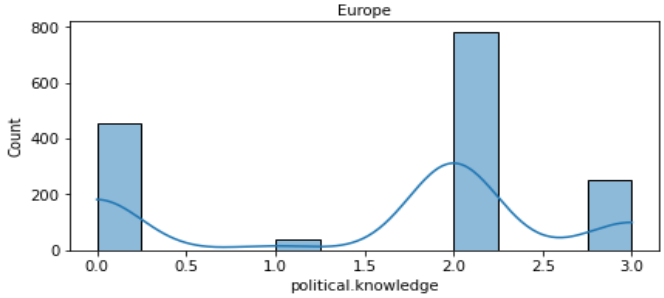

Blair

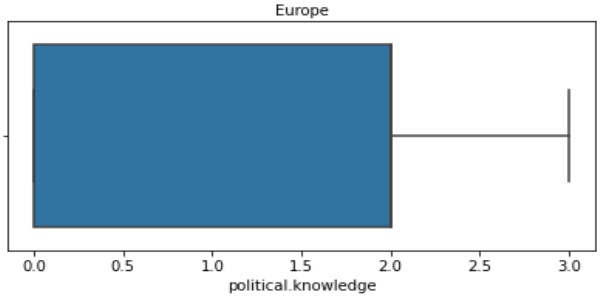

Count
Hague


Hague


Count
Europe


Europe

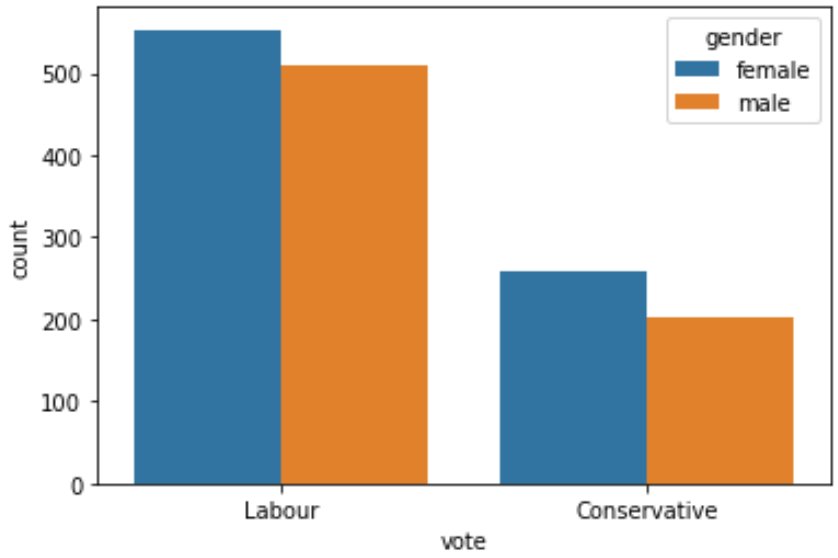

Count
political.knowledge

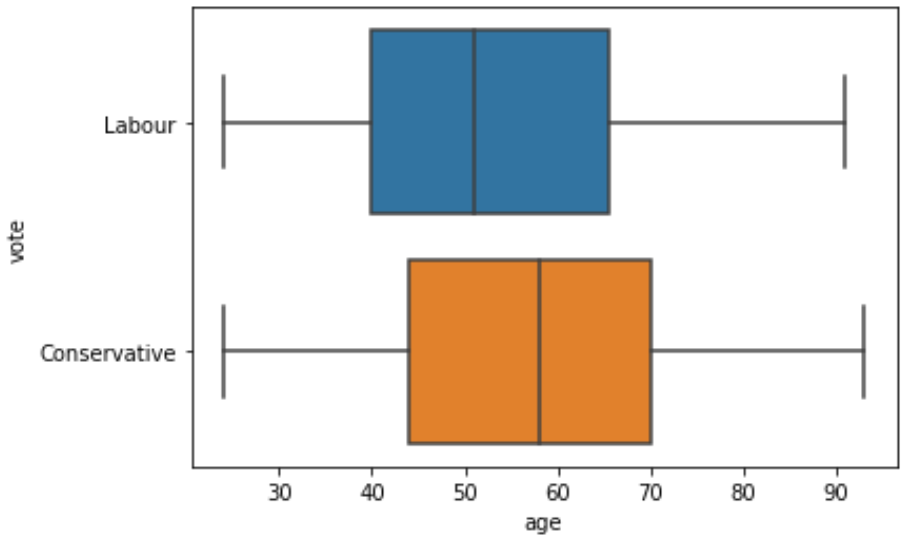

political.knowledge

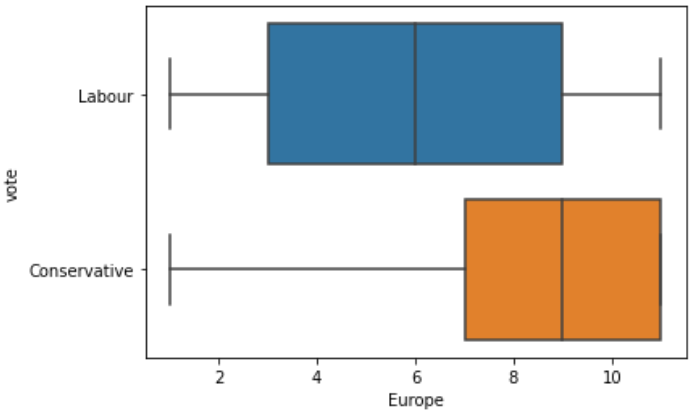
Inferences from Univariate Analysis:

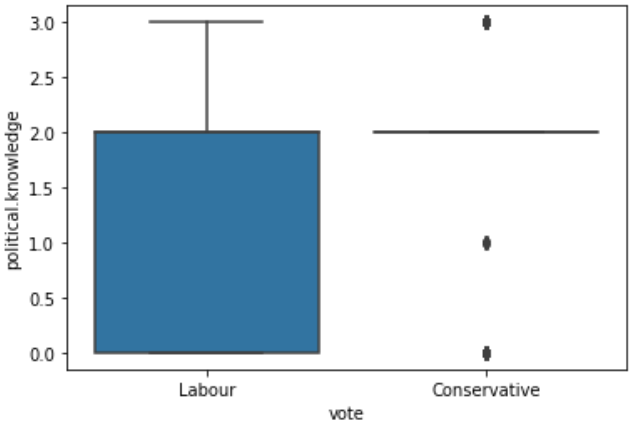
- Outliers are not present in the dataset except for assessment of current economic condition national, assessment of current economic condition household.
- Age variable is normally distributed.
- Labour party has got 69.7 % vote and Conservative party has got 30 % votes, proving majority to Labor party.
- Percentage of female voters (53%) are more than the percentage of male voters (47%).

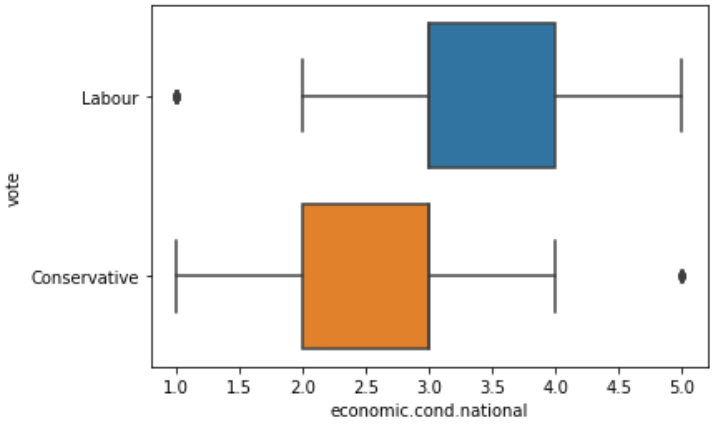
Bivariate Analysis

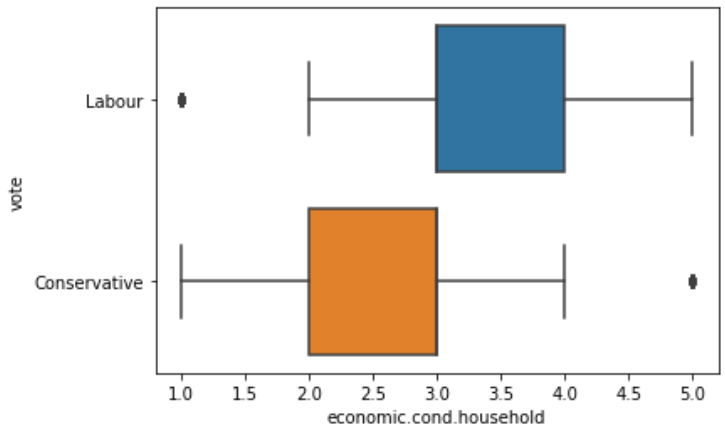





Labour
Conservative
vote
2
4
6
8
10
Europe


3.0
2.5
2.0
1.5
1.0
0.5
0.0
political.knowledge
Labour
Conservative
vote

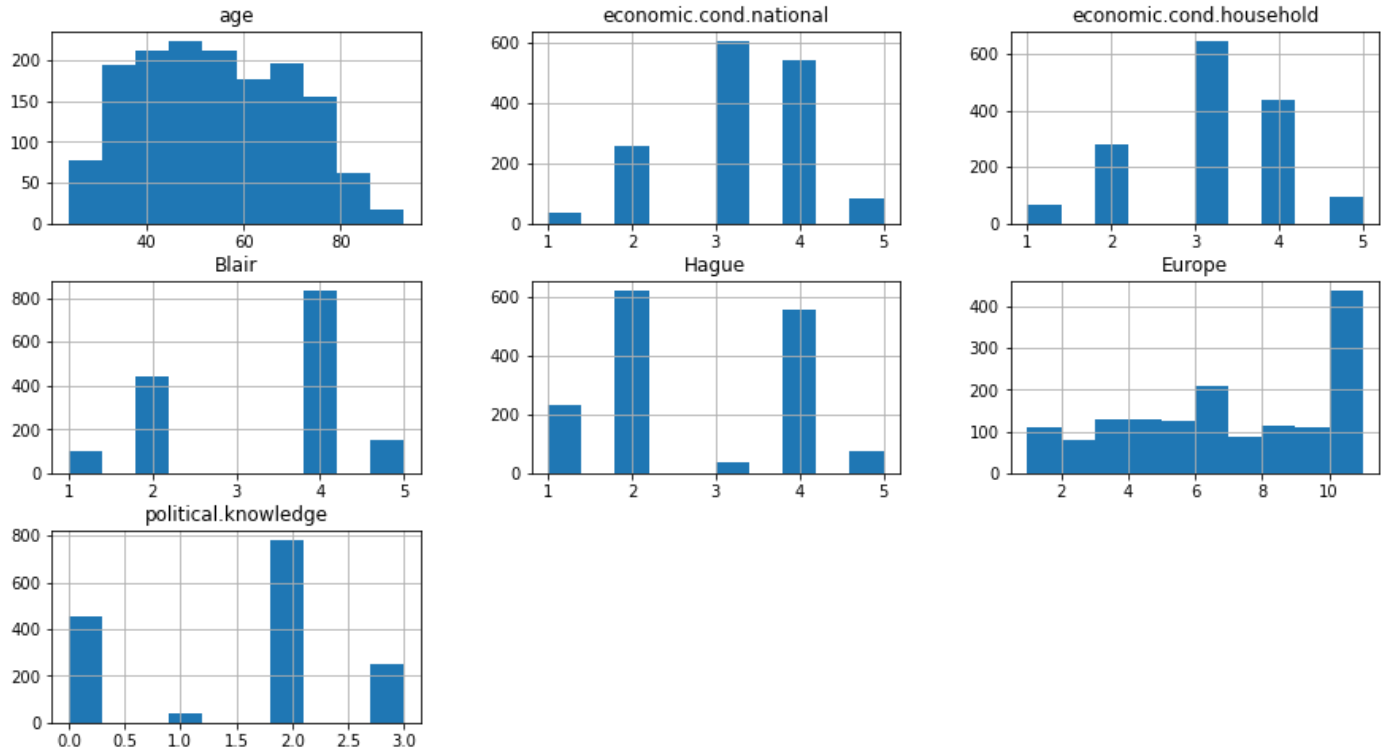

Labour
Conservative
vote
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
economic.cond.national


Labour
Conservative
vote
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
economic.cond.household

Inferences from Bivariate Analysis:

- The votes of female voters are more of each of the parties, Labor and Conservative
- People in the age group of 40 years to 65 years have voted for Labor party. People in the age group of 44 years to 70 years have voted for Conservative party.
- People with ‘Eurosceptic’ sentiment has voted for Conservative party.
  Majority of the people who don't oppose closer connections between Britain and the European Union have voted for Labor party.
  The strength of the people who possess Eurosceptic sentiment is comparatively lesser than the other group.
- People who possesses political knowledge of parties' positions on European integration have voted for Labor party.
  Conservative party vote has a mix of voters with minimum to complete political knowledge of party’s position on European Integration.
- People whose current national economic conditions (both national and household) are greater have preferred to vote for Labor party.
  People who are in the lower band of economic conditions (both national and household) have preferred to vote for conservative party.

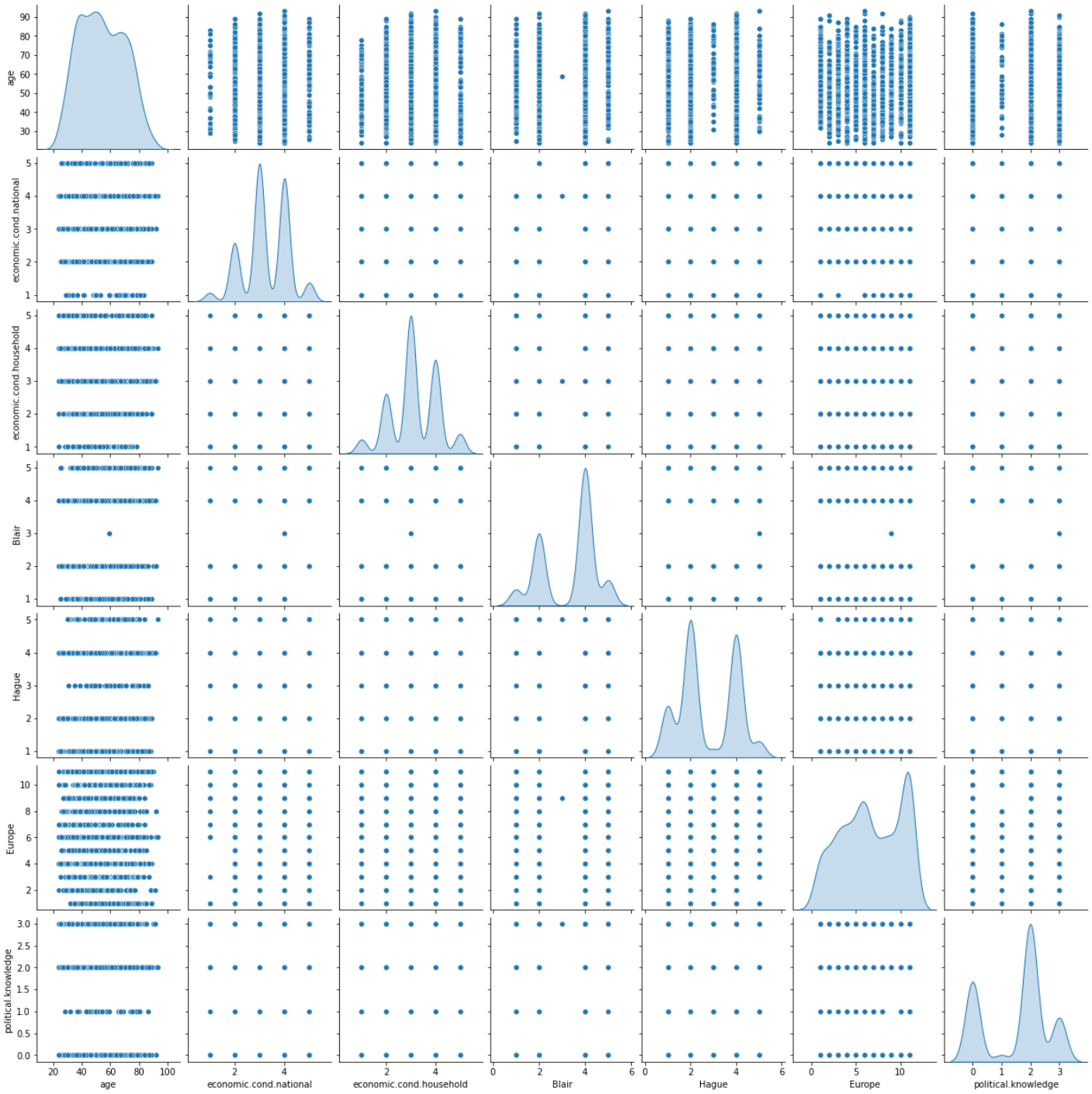
Distribution of the Variables:

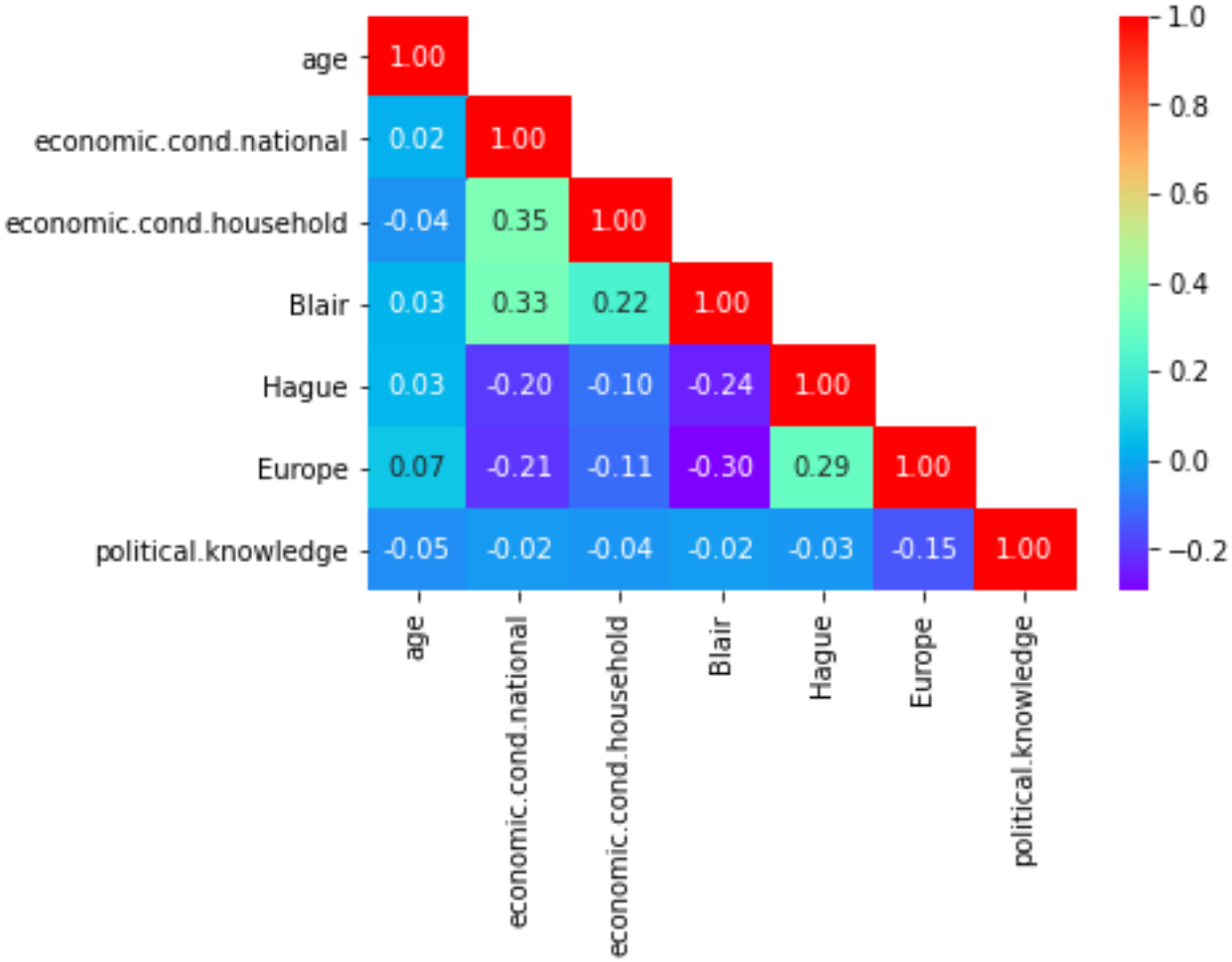


Age is normally distributed. Europe though the distribution is continuous, it is skewed towards right.

Other variables, 'economic.cond.national', 'economic.cond.household','Blair', 'Hague', 'political.knowledge' are discrete

Pair Plot:


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

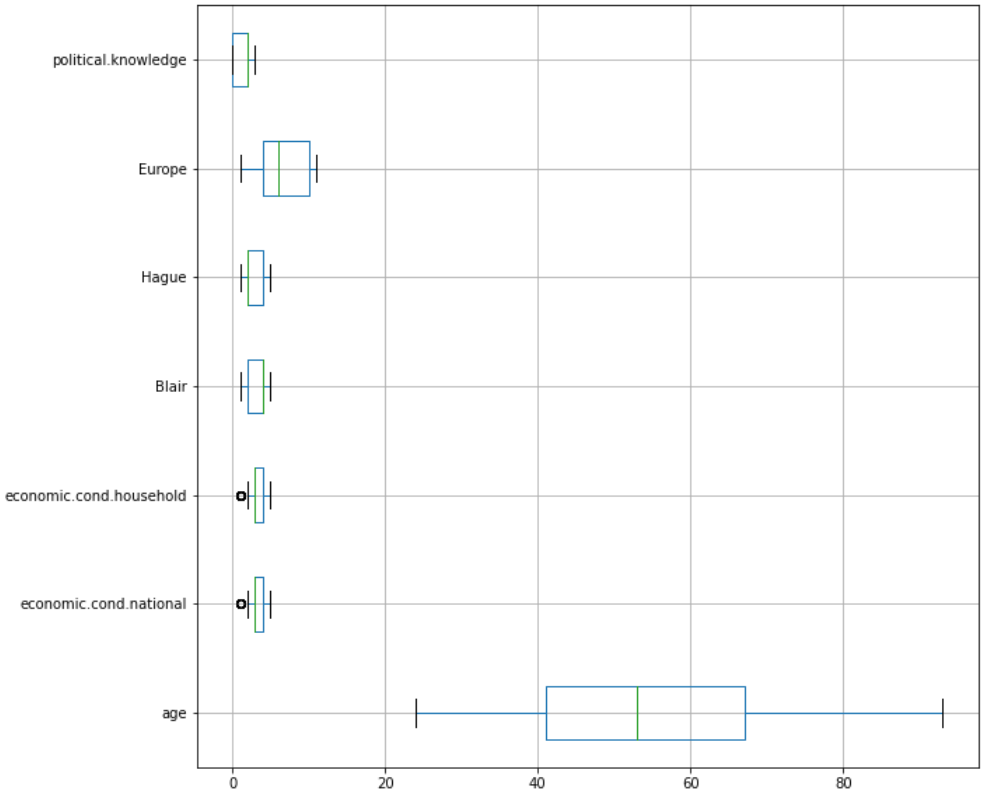


Inferences from Bivariate Analysis:

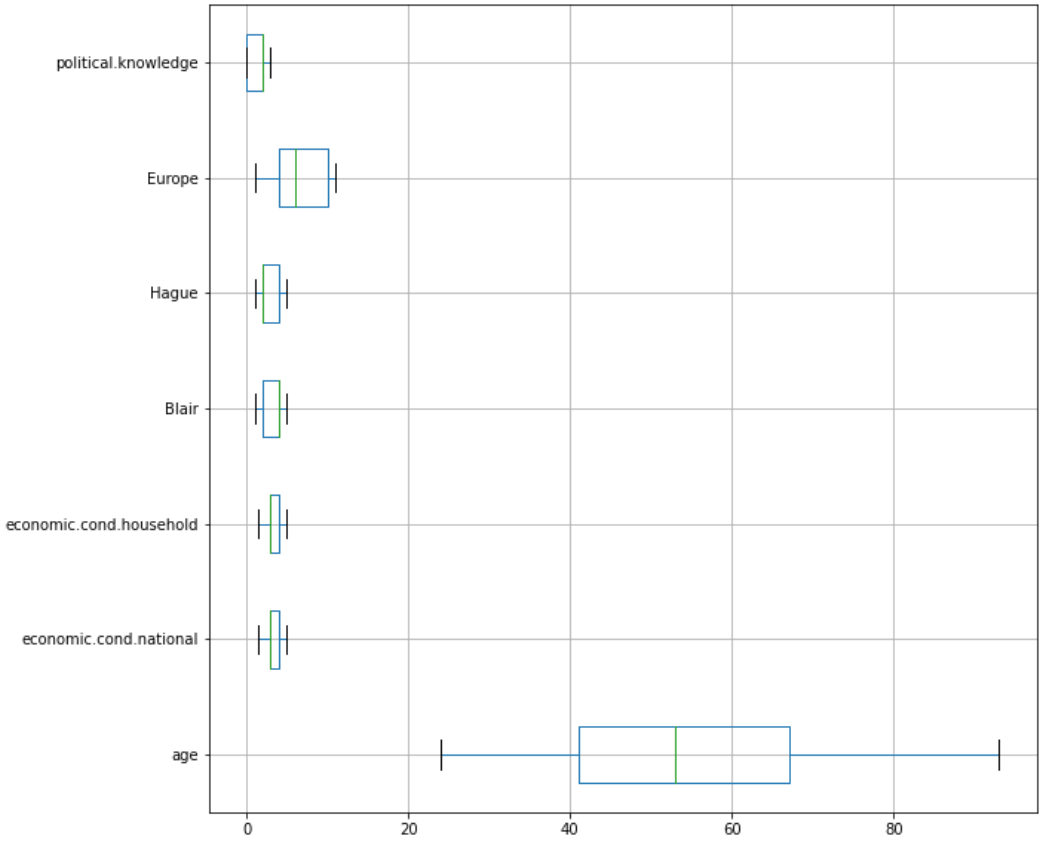
No co-relation exists between any of the variables in the dataset

People are inclined towards Labor party than the Conservative party

Outlier Check



After Removing Outliers:



Proportion of Outliers:

National economic condition - 1st Quartile(Q1) is: 3.0
National economic condition - 3rd Quartile(Q3) is: 4.0
IQR of national economic condition is 1.0
Lower outliers in Current national economic condition: 1.5
Upper outliers in Current national economic condition: 5.5
No of outliers in Current national economic condition upper : 0
No of outliers in Current national economic condition lower : 0
% of Outlier in Current national economic condition upper: 0 %
% of Outlier in Current national economic condition lower: 0 %

Household economic conditions - 1st Quartile(Q1) is: 3.0
Household economic conditions - 3rd Quartile(Q3) is: 4.0
IQR of economic.cond.household is 1.0
Lower outliers in Household economic conditions: 1.5
Upper outliers in Household economic conditions: 5.5
No of outliers in Household economic conditions upper: 0
No of outliers in Household economic conditions lower: 0
% of Outlier in Household economic conditions upper: 0 %
% of Outlier in Household economic conditions lower: 0 %

There are nearly no outliers in most of the numerical columns, only outlier is in Assessment of current national economic conditions and Assessment of current household economic conditions variable.

Data Preparation: 4 marks
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). (4 Marks)

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.

Label Encoding is performed for the independent categorical variable. Drop First is used to ensure that multiple columns created based on the levels of categorical variable are not included else it will result in multicollinearity.

```
df =pd.get_dummies(df, columns=cat1,drop_first=True)
df.head()
```

	vote	age	economic.cond.national	economic.cond.household	Blair	Hague	Europe	political.knowledge	gender_male
0	Labour	43	3.0	3.0	4	1	2	2	0
1	Labour	36	4.0	4.0	4	4	5	2	1
2	Labour	35	4.0	4.0	5	2	3	2	1
3	Labour	24	4.0	2.0	2	1	4	0	0
4	Labour	41	2.0	2.0	1	1	6	2	1

SCALING :

	Unnamed: 0	age	economic.cond.national	economic.cond.household	Blair	Hague	Europe	political.knowledge
count	1525.000000	1525.000000	1525.000000	1525.000000	1525.000000	1525.000000	1525.000000	1525.000000
mean	763.000000	54.182295	3.245902	3.140328	3.334426	2.746885	6.728525	1.542295
std	440.373894	15.711209	0.880969	0.929951	1.174824	1.230703	3.297538	1.083315
min	1.000000	24.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.000000
25%	382.000000	41.000000	3.000000	3.000000	2.000000	2.000000	4.000000	0.000000
50%	763.000000	53.000000	3.000000	3.000000	4.000000	2.000000	6.000000	2.000000
75%	1144.000000	67.000000	4.000000	4.000000	4.000000	4.000000	10.000000	2.000000
max	1525.000000	93.000000	5.000000	5.000000	5.000000	5.000000	11.000000	3.000000

Variance of Age is 246.84207478163592
Variance of economic.cond.national is 0.7275026892130287
Variance of economic.cond.household is 0.7837309065874962
Variance of Blair is 1.3802116948496193
Variance of Hague is 1.5146310399724625
Variance of Europe is 10.87375930467708
Variance of political knowledge is 1.1735708446280282

The standard deviation of age is 15.7 whereas the economic cond. National and economic. Co nd. Household are 0.8 and 0.9 and the std. deviation of Blair, Hague & political knowledge are i n the range of 1.08 whereas std. deviation of Europe is 3.29. This indicates that all the continuo us variables are in different scale.

Variance of Age, economic.cond.national, economic.cond.household, Blair, Hague, Europe, Po litical Knolwedge are on different ranges (Age in 246.84, Europe in 10.87 and others in 1's), i.e., in different scales.
Performing features scaling does not have much impact on Logistic Regression, Random Fore st, Decision Tree and Gaussian Naive Bayes. Hence Scaling is not required.
Scaling is needed only for MLP Classifier and KNN models.

Train-Test Split

```
# Split X and y into training and test set in 70:30 ratio
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30 , random_state=1)
print('Train Dataset - Independent Variables\n',X_train.head())
print('Test Dataset - Dependent Variable\n',X_test.head())
print('Train Dataset - Independent Variables\n',y_train.head())
print('Test Dataset - Dependent Variable\n',y_test.head())
```

Number of rows and columns of the training set for the independent variables: (1067, 8)
Number of rows and columns of the training set for the dependent variable: (1067,)
Number of rows and columns of the test set for the independent variables: (458, 8)
Number of rows and columns of the test set for the dependent variable: (458,)

```
Train Dataset - Independent Variables
       age  economic.cond.national  economic.cond.household  Blair  Hague
1453   62                      3.0                      3.0      2      2
275    49                      3.0                      3.0      2      2
1130   74                      4.0                      4.0      4      4
1153   57                      2.0                      3.0      4      2
1172   24                      4.0                      5.0      4      4

      Europe  political.knowledge  gender_male
1453      11                    2            0
275        8                    0            0
1130       7                    0            1
1153       6                    2            0
1172       6                    0            1

Train Dataset - Dependent Variables
 1453          Labour
275     Conservative
1130          Labour
1153          Labour
1172    Conservative
Name: vote, dtype: object
```

```
Test Dataset - Independent Variable
       age  economic.cond.national  economic.cond.household  Blair  Hague
91     49                      1.5                      1.5      2      4
1194   34                      3.0                      3.0      2      4
201    51                      2.0                      2.0      4      4
613    30                      2.0                      3.0      4      4
283    42                      3.0                      3.0      2      2

      Europe  political.knowledge  gender_male
91         8                    3            0
1194       9                    2            1
201        4                    2            1
613        7                    3            1
283        9                    0            1

Test Dataset - Dependent Variable
 91      Conservative
1194          Labour
201           Labour
613     Conservative
283           Labour
Name: vote, dtype: object
```

Modeling:
1.4 Apply Logistic Regression and LDA (linear discriminant analysis).

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 behind the selection of different values for the parameters involved in each model. Calculate Train and Test Accuracies for each model. Comment on the validness of models (over fitting or under fitting)

Logistic Regression Model:

Formulate a logistic regression model on the train data.

```
LogReg_model = LogisticRegression(max_iter=1000)
LogReg_model.fit(X_train, y_train)
```

```
LogisticRegression(max_iter=1000)
```

Predicting on Training and Test dataset

```
ytrain_predict = LogReg_model.predict(X_train)
ytest_predict = LogReg_model.predict(X_test)
```

Logistic Regression Model Evaluation

Inferences from Logistic Regression Model:

Accuracy of the Logistic Regression in Training dataset 0.8397375820056232

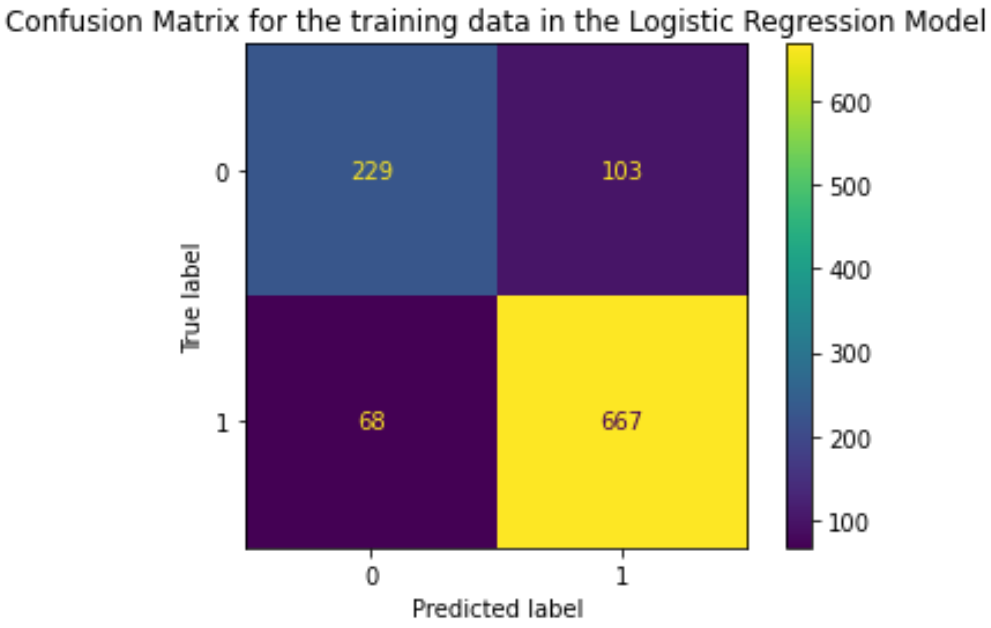
Accuracy of the Logistic Regression in Test dataset 0.8209606986899564

The accuracy of the training dataset and the test data set are almost similar and hence there is no over fitting or underfitting.

When the Logistic Regression is subjected to different parameters, solvers - 'sag', 'lbfgs', 'newton' , tol':[0.0001,0.00001], 'penalty':['l2','none'] – Grid search CV resulted in the optimized best parameters as follows, 'penalty': 'l2', 'solver': 'lbfgs', 'tol': 0.0001

However, the accuracy score remains the same with optimized parameters as well.

Confusion Matrix - Logistic Regression Model



Classification Report - Logistic Regression Model

Classification Report of the training data in Logistic Regression Model

	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

Classification Report of the test data in Logistic Regression Model

	precision	recall	f1-score	support
0	0.70	0.65	0.67	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

AUC and ROC - Logistic Regression Model

AUC score for Training Data in the Logistic Regression Model: 0.890

AUC score of Test data in the Logistic Regression Model: 0.883

LDA

Build LDA Model

```
clf = LinearDiscriminantAnalysis()
LDA_model=clf.fit(X_train,y_train)
```

Prediction on Training and Test dataset

```
# Training Data Class Prediction with a cut-off value of 0.5
pred_class_train = LDA_model.predict(X_train)

# Test Data Class Prediction with a cut-off value of 0.5
pred_class_test = LDA_model.predict(X_test)
```

Confusion Matrix of Training Data and Test Data in LDA model

Classification Report of Training Data and Test Data in LDA model

Classification Report of the training data:

	precision	recall	f1-score	support
0	0.76	0.71	0.73	332
1	0.87	0.90	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

Classification Report of the test data:

	precision	recall	f1-score	support
0	0.69	0.66	0.67	130
1	0.87	0.88	0.87	328
accuracy			0.82	458
macro avg	0.78	0.77	0.77	458
weighted avg	0.82	0.82	0.82	458

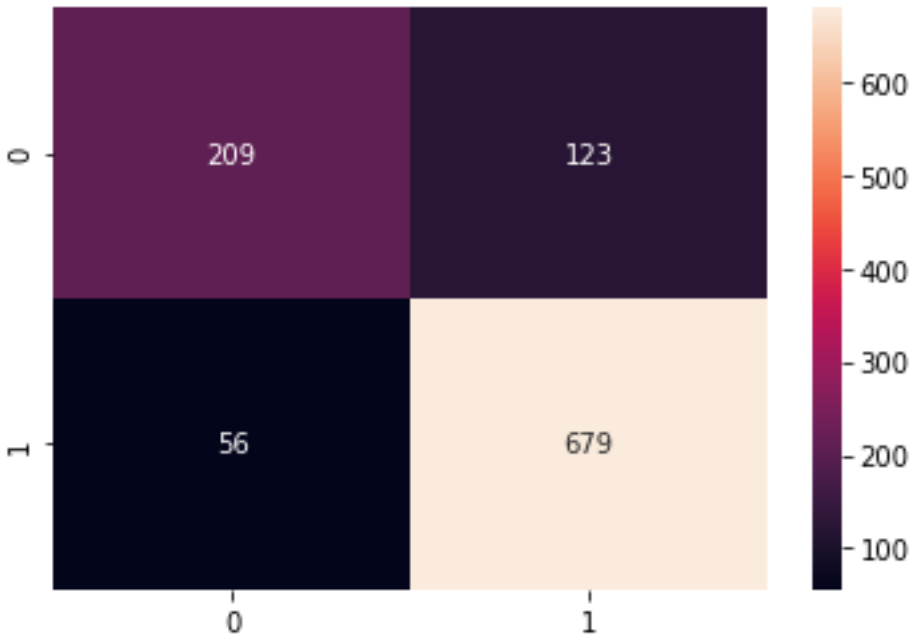
AUC and ROC for the training data and test data in LDA model

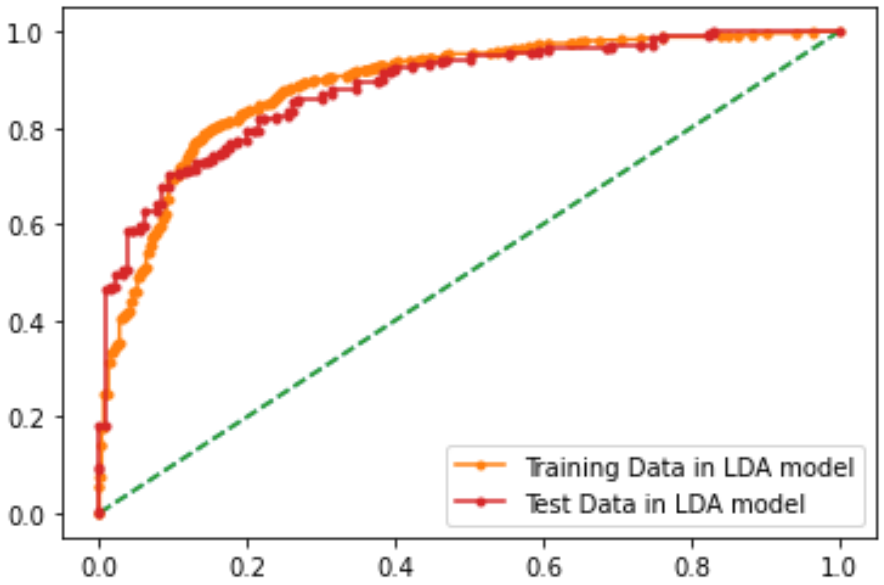
AUC for the Training Data in the LDA model: 0.889

AUC for the Test Data in the LDA model: 0.884

Change the cut-off values for maximum accuracy

```
for j in np.arange(0.1,1,0.1):
    custom_prob = j #defining the cut-off value of our choice
    custom_cutoff_data=[]#defining an empty list
    for i in range(0,len(y_train)):#defining a loop for the length of the test data
        if np.array(pred_prob_train[:,1])[i] > custom_prob:#issuing a condition for our probability values to be
            #greater than the custom cutoff value
            a=1#if the probability values are greater than the custom cutoff then the value should be 1
        else:
            a=0#if the probability values are less than the custom cutoff then the value should be 0
        custom_cutoff_data.append(a)#adding either 1 or 0 based on the condition to the end of the list defined by us
    print(round(j,3),'\n')
    print('Accuracy Score',round(metrics.accuracy_score(y_train,custom_cutoff_data),4))
    print('F1 Score',round(metrics.f1_score(y_train,custom_cutoff_data),4),'\n')
    plt.figure(figsize=(6,4))
    print('Confusion Matrix')
    sns.heatmap(metrics.confusion_matrix(y_train,custom_cutoff_data),annot=True,fmt='.4g'),'\n\n'
    plt.show();
```





Inferences from LDA model:

LDA model is affected by a class imbalance problem. Since we only have 1525 observations, if re-build the same LDA model with more number of data points, an even better model could be built.

Accuracy of the training dataset is 84% and that of the test dataset is 82%. The model is tuned for different values of cut off values.

The optimum accuracy score and F1 score for the LDA model is achieved for Cut off value of 0 .4

Accuracy Score in LDA model: 0.8322
F1 Score in LDA model: 0.8835

1.5 Apply KNN Model and Naïve Bayes Model. Interpret the results. (4 marks)

Apply KNN Model and Naïve Bayes Model (2pts). Interpret the inferences of each model (2 pts). Successful implementation of each model. Logical reason behind the selection of different values for the parameters involved in each model. Calculate Train and Test Accuracies for each model. Comment on the validness of models (over fitting or under fitting)

Gaussian Naive Bayes

```
from sklearn.naive_bayes import GaussianNB
from sklearn import metrics
```

```
NB_model = GaussianNB()
NB_model.fit(X_train, y_train)
```

Performance Matrix on train data set

```
Accuracy score of the Gaussian NB model - Train Dataset
 0.8322399250234301
Confusion Matrix of the Gaussian NB model - Train Dataset
 [[240  92]
 [ 87 648]]
Classification Report of the Gaussian NB model - Train Dataset
              precision    recall  f1-score   support

           0       0.73      0.72      0.73       332
           1       0.88      0.88      0.88       735

    accuracy                           0.83      1067
   macro avg       0.80      0.80      0.80      1067
weighted avg       0.83      0.83      0.83      1067
```

Performance Matrix on test data set

```
Accuracy score of the Gaussian NB model - Test Dataset
 0.8231441048034934
Confusion Matrix of the Gaussian NB model - Test Dataset
 [[ 94  36]
 [ 45 283]]
Classification Report of the Gaussian NB model - Test Dataset
              precision    recall  f1-score   support

           0       0.68      0.72      0.70       130
           1       0.89      0.86      0.87       328

    accuracy                           0.82       458
   macro avg       0.78      0.79      0.79       458
weighted avg       0.83      0.82      0.82       458
```

Inferences from Gaussian NB Model:

Gaussian NB algorithm had training set accuracy of 83 % and testing set accuracy of 82%. The model seems to be performing well.

KNN Model

A good KNN performance usually requires preprocessing of data to make all variables similarly scaled and centered

Apply z-score on continues columns to see the performance for KNN

X[["age","economic.cond.national","economic.cond.household","Blair","Hague","Europe","political.knowledge"]]=X[["age","economic.cond.national","economic.cond.household","Blair","Hague","Europe","political.knowledge"]].apply(zscore)

Building a KNN model

```
from sklearn.neighbors import KNeighborsClassifier
KNN_model=KNeighborsClassifier()
KNN_model.fit(X_train,y_train)
```

```
KNeighborsClassifier()
```

```
## Performance Matrix on train data set
y_train_predict = KNN_model.predict(X_train)
model_score = KNN_model.score(X_train, y_train)
print('Accuracy score of the KNN model - Train Dataset\n',model_score)
print('Confusion Matrix of the KNN model - Train Dataset\n',metrics.confusion_matrix(y_train, y_train_predict))
print('Classification Report of the KNN model - Train Dataset\n',metrics.classification_report(y_train, y_train_predict))
```

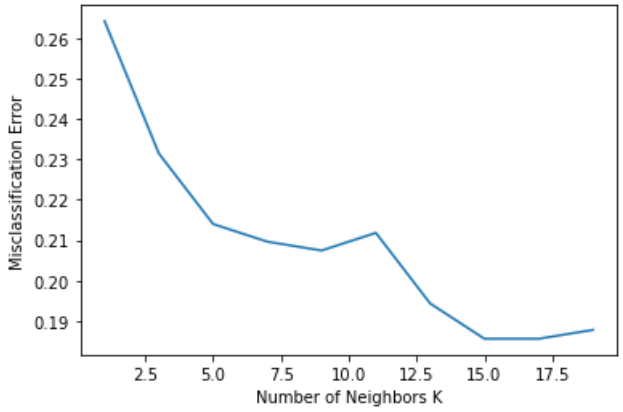
```
Accuracy score of the KNN model (n_neighbors=7)- Train Dataset
 0.8500468603561387
Confusion Matrix of the KNN model (n_neighbors=7) - Train Dataset
 [[233  99]
 [ 61 674]]
Classification Report of the KNN model (n_neighbors=7)- Train Dataset
              precision    recall  f1-score   support

           0       0.79      0.70      0.74       332
           1       0.87      0.92      0.89       735

    accuracy                           0.85      1067
   macro avg       0.83      0.81      0.82      1067
weighted avg       0.85      0.85      0.85      1067

Accuracy score of the KNN model (n_neighbors=7)- Test Dataset
 0.7903930131004366
Confusion Matrix of the KNN model (n_neighbors=7) - Test Dataset
 [[ 78  52]
 [ 44 284]]
Classification Report of the KNN model (n_neighbors=7)- Test Dataset
              precision    recall  f1-score   support

           0       0.64      0.60      0.62       130
           1       0.85      0.87      0.86       328

    accuracy                           0.79       458
   macro avg       0.74      0.73      0.74       458
weighted avg       0.79      0.79      0.79       458
```

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

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

For K = 15 it is giving the best test accuracy lets check train and test for K=17 with other evaluation metrics

With K = 15, accuracy score for the training dataset is 82.47 and the test dataset is 81.44



1.6 Model Tuning, Bagging (Random Forest should be applied for Bagging), and Boosting. (7 marks)

1.6) Model Tuning (4 pts) , Bagging (1.5 pts) and Boosting (1.5 pts). Apply grid search on each model (include all models) and make models on best_params. Define a logic behind choosing particular values for different hyper-parameters for grid search. Compare and comment on performances of all. Comment on feature importance if applicable. Successful implementation of both algorithms along with inferences and comments on the model performances.

Model Tuning:

Grid Search and Cross Validation are the popularly used model tuning methods.

Grid Search evaluates all the combinations from a list of desired hyper-parameters and reports which combination has the best accuracy. For every model there are many hyper-parameters, so a good way to define the best set of hyper-parameters is by trying different combinations and comparing the results.

`clf = GridSearchCV()`

In order to avoid manually setting different percentages for training and testing sets, use of the cross_validate function will divide the training set into k folds and then try the different combinations where each of the combinations will use a different fold as the test set and the remaining k-1 folds as the train set. k is the desired number of folds. Usually k=5 works well.

`sklearn.model_selection.cross_validate()`

Applying GridSearchCV for Logistic Regression

Best Parameters : {'penalty': 'l2', 'solver': 'lbfgs', 'tol': 0.0001} LogisticRegression (max_iter=10000, n_jobs=2)

```
grid={'penalty':['l2','none'],
      'solver':['sag','lbfgs','newton'],
      'tol':[0.0001,0.00001]}
```

```
LogReg_Opt_model = LogisticRegression(max_iter=10000,n_jobs=2)
```

```
grid_search = GridSearchCV(estimator = LogReg_Opt_model, param_grid = grid, cv = 3,n_jobs=-1,scoring='f1')
```

```
grid_search.fit(X_train, y_train)
```

```
GridSearchCV(cv=3, estimator=LogisticRegression(max_iter=10000, n_jobs=2),
             n_jobs=-1,
             param_grid={'penalty': ['l2', 'none'],
                         'solver': ['sag', 'lbfgs', 'newton'],
                         'tol': [0.0001, 1e-05]},
             scoring='f1')
```

```
print(grid_search.best_params_,'\n')
print(grid_search.best_estimator_)
```

```
{'penalty': 'l2', 'solver': 'lbfgs', 'tol': 0.0001}

LogisticRegression(max_iter=10000, n_jobs=2)
```

```
best_model = grid_search.best_estimator_
```

Grid Search on Random Forest

```
param_grid = {
    'max_depth': [5,10],
    'max_features': [4, 8],
    'min_samples_leaf': [3, 15,30],
    'min_samples_split': [30,50,75,100],
    'n_estimators': [300,400,500, 600]
}

rfr = RandomForestRegressor(random_state=1)

grid_search = GridSearchCV(estimator = rfr, param_grid = param_grid, cv = 3)
```

```
grid_search.fit(X_train,y_train)
```

```
GridSearchCV(cv=3, estimator=RandomForestRegressor(random_state=1),
             param_grid={'max_depth': [5, 10], 'max_features': [4, 8],
                         'min_samples_leaf': [3, 15, 30],
                         'min_samples_split': [30, 50, 75, 100],
                         'n_estimators': [300, 400, 500, 600]})
```

```
print(grid_search.best_params_)
print(grid_search.best_estimator_)
```

```
{'max_depth': 5, 'max_features': 4, 'min_samples_leaf': 3, 'min_samples_split': 30, 'n_estimators': 500}
RandomForestRegressor(max_depth=5, max_features=4, min_samples_leaf=3,
                      min_samples_split=30, n_estimators=500, random_state=1)
```

Cross Validation on Naive Bayes Model

```
from sklearn.model_selection import cross_val_score
scores = cross_val_score(NB_model, X_train, y_train, cv=10)
scores
```

```
array([0.81308411, 0.82242991, 0.82242991, 0.85046729, 0.81308411,
       0.81308411, 0.8317757 , 0.88679245, 0.82075472, 0.82075472])
```

```
scores = cross_val_score(NB_model, X_test, y_test, cv=10)
scores
```

```
array([0.82608696, 0.86956522, 0.82608696, 0.80434783, 0.76086957,
       0.80434783, 0.84782609, 0.91304348, 0.88888889, 0.82222222])
```

Cross Validation on KNN Model

```
from sklearn.model_selection import cross_val_score
scores = cross_val_score(KNN_SM_model, X_train_res, y_train_res, cv=10)
scores
```

```
array([0.78231293, 0.79591837, 0.82993197, 0.82312925, 0.82312925,
       0.82993197, 0.85714286, 0.87755102, 0.87755102, 0.87755102])
```

```
scores = cross_val_score(KNN_SM_model, X_test, y_test, cv=10)
scores
```

```
array([0.84782609, 0.82608696, 0.82608696, 0.7826087 , 0.80434783,
       0.7826087 , 0.93478261, 0.80434783, 0.88888889, 0.8       ])
```

After 10 fold cross validation, scores both on train and test data set respectively for all 10 folds are not almost same. The score varies from 78% to 93%
Hence our model tuning is required.

Bagging:

Building Random Forest Model

```
from sklearn.ensemble import RandomForestClassifier

RF_model=RandomForestClassifier(n_estimators=100,random_state=1)
RF_model.fit(X_train, y_train)
```

```
RandomForestClassifier(random_state=1)
```

```
Accuracy score of the Training Dataset - Random Forest Classifier

 0.9990627928772259
Confusion Matrix of the Training Dataset -  Random Forest Classifier

 [[331   1]
 [  0 735]]
Classification Report of the Training Dataset - Random Forest Classifier

              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
```

```
Accuracy score of the Test Dataset - Random Forest Classifier

 0.8209606986899564
Confusion Matrix of the Test Dataset -  Random Forest Classifier

 [[ 90  40]
 [ 42 286]]
Classification Report of the Test Dataset - Random Forest Classifier

              precision    recall  f1-score   support

           0       0.68      0.69      0.69       130
           1       0.88      0.87      0.87       328

    accuracy                           0.82       458
   macro avg       0.78      0.78      0.78       458
weighted avg       0.82      0.82      0.82       458
```

Random forest output proves to be a clear case of Overfit

Bagging

```
from sklearn.ensemble import BaggingClassifier
from sklearn.ensemble import RandomForestClassifier

RF_model=RandomForestClassifier(n_estimators=100,random_state=1)

Bagging_model=BaggingClassifier(base_estimator=RF_model,n_estimators=100,random_state=1)
Bagging_model.fit(X_train, y_train)
```

```
BaggingClassifier(base_estimator=RandomForestClassifier(random_state=1),
                  n_estimators=100, random_state=1)
```

```
Accuracy score of the Training Dataset - Bagging Random Forest Classifier

 0.9653233364573571
Confusion Matrix of the Training Dataset - Bagging Random Forest Classifier

 [[304  28]
 [  9 726]]
Classification Report of the Training Dataset - Bagging Random Forest Classifier

              precision    recall  f1-score   support

           0       0.97      0.92      0.94       332
           1       0.96      0.99      0.98       735

    accuracy                           0.97      1067
   macro avg       0.97      0.95      0.96      1067
weighted avg       0.97      0.97      0.97      1067
```

```
Accuracy score of the Test Dataset - Bagging Random Forest Classifier

 0.8362445414847162
Confusion Matrix of the Test Dataset - Bagging Random Forest Classifier

 [[ 92  38]
 [ 37 291]]
Classification Report of the Test Dataset - Bagging Random Forest Classifier

              precision    recall  f1-score   support

           0       0.71      0.71      0.71       130
           1       0.88      0.89      0.89       328

    accuracy                           0.84       458
   macro avg       0.80      0.80      0.80       458
weighted avg       0.84      0.84      0.84       458
```

Inferences:

Bagging is a case of Overfitting as the accuracy in the Train dataset is 97% whereas the Test dataset has an accuracy of only 84%. Model seems to be valid only if the difference in the accuracy between training and test dataset is in the range of 10%.

Boosting:

Ada boosting

```
from sklearn.ensemble import AdaBoostClassifier

ADB_model = AdaBoostClassifier(n_estimators=100,random_state=1)
ADB_model.fit(X_train,y_train)
```

```
AdaBoostClassifier(n_estimators=100, random_state=1)
```

```
Accuracy score of the Train Dataset - Ada Boosting

 0.8472352389878163
Confusion Matrix of the Train Dataset - Ada Boosting

 [[238  94]
 [ 69 666]]
Classification Report of the Train Dataset - Ada Boosting

              precision    recall  f1-score   support

           0       0.78      0.72      0.74       332
           1       0.88      0.91      0.89       735

    accuracy                           0.85      1067
   macro avg       0.83      0.81      0.82      1067
weighted avg       0.84      0.85      0.85      1067
```

```
Accuracy score of the Test Dataset - Ada Boosting

 0.8187772925764192
Confusion Matrix of the Test Dataset - Ada Boosting

 [[ 90  40]
 [ 43 285]]
Classification Report of the Test Dataset - Ada Boosting

              precision    recall  f1-score   support

           0       0.68      0.69      0.68       130
           1       0.88      0.87      0.87       328

    accuracy                           0.82       458
   macro avg       0.78      0.78      0.78       458
weighted avg       0.82      0.82      0.82       458
```

Inferences:

- Ada boosting works well as the Accuracy score of the Train Data set is 0.85% and Test Data set is 0.82%
- F1-score for Class 1 is 0.91 in the train data set and 0.87 in the test data set.
- Recall % is also good with 0.91 % for Train data and 0.87% for the test data

Gradient Boosting

```
from sklearn.ensemble import GradientBoostingClassifier

gbcl = GradientBoostingClassifier(random_state=1)
gbcl = gbcl.fit(X_train, y_train)
```

```
Accuracy score of the Train Dataset - Gradient Boosting

 0.8865979381443299
Confusion Matrix of the Train Dataset - Gradient Boosting

 [[262  70]
 [ 51 684]]
Classification Report of the Train Dataset - Gradient Boosting

              precision    recall  f1-score   support

           0       0.84      0.79      0.81       332
           1       0.91      0.93      0.92       735

    accuracy                           0.89      1067
   macro avg       0.87      0.86      0.87      1067
weighted avg       0.89      0.89      0.89      1067
```

```
Accuracy score of the Test Dataset - Gradient Boosting

 0.8318777292576419
Confusion Matrix of the Test Dataset - Gradient Boosting

 [[ 96  34]
 [ 43 285]]
Classification Report of the Test Dataset - Gradient Boosting

              precision    recall  f1-score   support

           0       0.69      0.74      0.71       130
           1       0.89      0.87      0.88       328

    accuracy                           0.83       458
   macro avg       0.79      0.80      0.80       458
weighted avg       0.84      0.83      0.83       458
```

Inferences:

- Gradient boosting works well as the Accuracy score of the Train Data set is 0.89% and Test Data set is 0.83%
- F1-score for Class 1 is 0.92 in the train data set and 0.88 in the test data set.
- Recall % is also good with 0.93 % for Train data and 0.87%

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. (7 marks)
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)

The models taken into consideration are as follows.

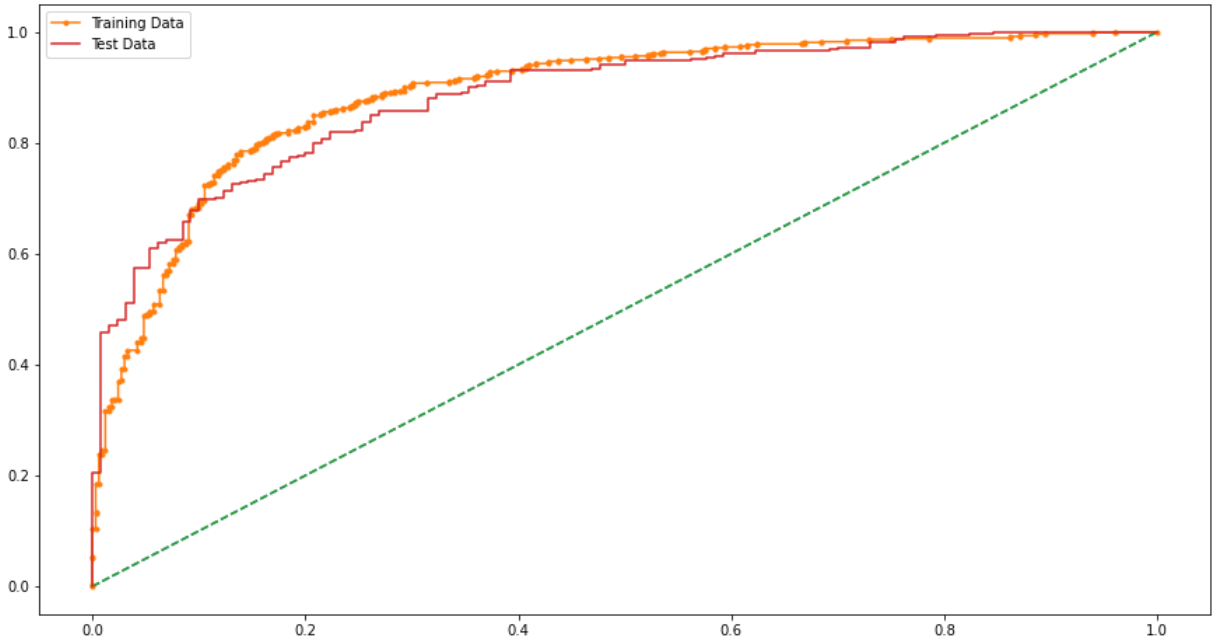
- Logistic Regression
- LDA
- Gaussian NB
- KNN
- Decision Tree
- Random Forest

Performance metrics	Accuracy Score		Recall - Class 1		F1-score		Confusion Matrix		Area under Curve	
Machine Learning Models	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
Logistic Regression	0.84	0.82	0.91	0.89	0.89	0.88	[[229 103] [68 667]]	[[85 45] [37 291]]	0.89	0.883
LDA (cut off - 0.4)	0.82	0.83	0.9	0.92	0.89	0.87	[[209 123] [56 679]]	[[86 44] [39 289]]	0.889	0.884
Gaussian NB	0.83	0.82	0.88	0.86	0.88	0.87	[[240 92] [87 648]]	[[94 36] [45 283]]		
KNN (K=15)	0.82	0.81	0.91	0.88	0.88	0.87	[[212 120] [67 668]]	[[86 44] [41 287]]		
Random Forest	1.00	0.82	1	0.87	1	0.87	[[331 1] [0 735]]	[[90 40] [42 286]]		
Bagging	0.97	0.84	0.99	0.89	0.98	0.89	[[304 28] [9 726]]	[[92 38] [37 291]]		
Ada Boosting	0.85	0.82	0.91	0.87	0.89	0.87	[[238 94] [69 666]]	[[90 40] [43 285]]		
Gradient Boosting	0.89	0.83	0.93	0.87	0.92	0.88	[[262 70] [51 684]]	[[96 34] [43 285]]		

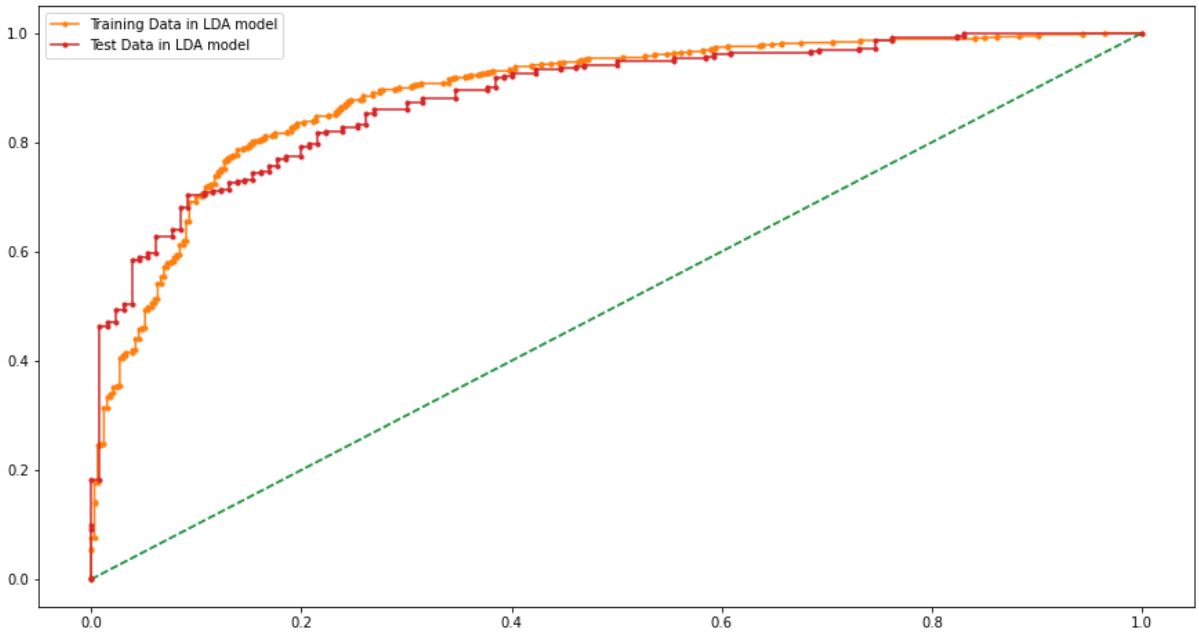
Inferences:

- Logistic Regression, LDA, Gaussian NB, KNN, Boosting seems to work well for the model in our case with respect to Train and Test accuracy.
- Random forest and Bagging seem to be a clear case of overfitting.
- Compared to all the other models, Logistic Regression gives the best fit accuracy between training and testing data.
- The model performs well in terms of F1 score and recall as well. Hence the best preferred model for this dataset is Logistic Regression.

ROC Curve for the Logistic Regression Model



ROC curve for LDA model



Inference: 5 marks

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

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.

Business Insights from Logistic Regression model:

```
# Fit the Logistic Regression model
model = LogisticRegression(solver='newton-cg',max_iter=10000,penalty='none',verbose=True,n_jobs=2)
Class=model.fit(X_train, y_train)
```

```
[Parallel(n_jobs=2)]: Using backend LokyBackend with 2 concurrent workers.
[Parallel(n_jobs=2)]: Done   1 out of   1 | elapsed:    0.9s finished
```

```
print(Class.coef_, Class.intercept_)
```

```
[[-0.02065241  0.37302829  0.16997518  0.57009179 -0.83832316 -0.23804024
  -0.4788061   0.30145485]] [3.28222157]
```

age --> -0.02065241

economic.cond.national --> 0.37302829

economic.cond.household --> 0.16997518

Blair --> 0.57009179

Hague --> -0.83832316

Europe --> -0.23804024

political.knowledge --> -0.4788061

gender_male --> 0.30145485

The important factors that determines which party gets the vote is as follows:

• Assessment of the Labor leader Blair has got co-efficient of 0.57

• Next feature of importance goes to the Assessment of current national economic conditions with co-efficient of 0.37

• The next feature that contributes is Gender followed by the assessment of current household economic condition.

• The feature Europe, Hague, political knowledge, age has very minimum importance in prediction of vote.

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:

President Franklin D. Roosevelt in 1941

President John F. Kennedy in 1961

President Richard Nixon in 1973

(Hint: use .words(), .raw(), .sent() for extracting counts)

2.1 Find the number of characters, words, and sentences for the mentioned documents. – 3 Marks

Number of Characters in President Roosevelt speech 7571
Number of Words in President Roosevelt speech 1360
Number of sentences in Roosevelt speech 68

Number of Characters in President John F. Kennedy speech 7618
Number of Words in President John F. Kennedy speech 1390
Number of sentences in President John F. Kennedy speech 53

Number of Characters in President Richard Nixon in 1973 speech 9991
Number of Words in President Richard Nixon in 1973 speech 1819
Number of sentences in President Richard Nixon in 1973 speech 69

2.2 Remove all the stop words from all three speeches. – 3 Marks

Roosevelt Speech – After stop words are removed:

'On national day inauguration since 1789, people renewed sense dedication United States. In Washington\'s day task people create weld together nation. In Lincoln \'s day task people preserve Nation disruption within. In day task people save N ation institutions disruption without. To us come time, midst swift happenings, pause moment take stock -- recall place history been, rediscover may be. If not, risk real peril inaction. Lives nations determined count years, lifetime human s pirit. The life man three-score years ten: little more, little less. The life na tion fullness measure live. There men doubt this. There men believe democracy, f orm Government frame life, limited measured kind mystical artificial fate that, unexplained reason, tyranny slavery become surging wave future -- freedom ebbing tide. But Americans know true. Eight years ago, life Republic seemed frozen fata listic terror, proved true. We midst shock -- acted. We acted quickly, boldly, d ecisively. These later years living years -- fruitful years people democracy. Fo r brought us greater security and, I hope, better understanding life\'s ideals m easured material things. Most vital present future experience democracy successf ully survived crisis home; put away many evil things; built new structures endur ing lines; and, all, maintained fact democracy. For action taken within three-wa y framework Constitution United States. The coordinate branches Government conti nue freely function. The Bill Rights remains inviolate. The freedom elections wh olly maintained. Prophets downfall American democracy seen dire predictions come naught. Democracy dying. We know seen revive--and grow. We know cannot die -- bu ilt unhampered initiative individual men women joined together common enterprise

-- enterprise undertaken carried free expression free majority. We know democrac y alone, forms government, enlists full force men\'s enlightened will. We know d emocracy alone constructed unlimited civilization capable infinite progress impr ovement human life. We know because, look surface, sense still spreading every c ontinent -- humane, advanced, end unconquerable forms human society. A nation, l ike person, body--a body must fed clothed housed, invigorated rested, manner mea sures objectives time. A nation, like person, mind -- mind must kept informed al ert, must know itself, understands hopes needs neighbors -- nations live within narrowing circle world. And nation, like person, something deeper, something per manent, something larger sum parts. It something matters future -- calls forth s acred guarding present. It thing find difficult -- even impossible -- hit upon s ingle, simple word. And yet understand -- spirit -- faith America. It product ce nturies. It born multitudes came many lands -- high degree, mostly plain people, sought here, early late, find freedom freely. The democratic aspiration mere rec ent phase human history. It human history. It permeated ancient life early peopl es. It blazed anew middle ages. It written Magna Charta. In Americas impact irre sistible. America New World tongues, peoples, continent new-found land, came bel ieved could create upon continent new life -- life new freedom. Its vitality wri tten Mayflower Compact, Declaration Independence, Constitution United States, Ge ttysburg Address. Those first came carry longings spirit, millions followed, sto ck sprang -- moved forward constantly consistently toward ideal gained stature c larity generation. The hopes Republic cannot forever tolerate either undeserved poverty self-serving wealth. We know still far go; must greatly build security o pportunity knowledge every citizen, measure justified resources capacity land. B ut enough achieve purposes alone. It enough clothe feed body Nation, instruct in form mind. For also spirit. And three, greatest spirit. Without body mind, men k now, Nation could live. But spirit America killed, even though Nation\'s body mi nd, constricted alien world, lived on, America know would perished. That spirit -- faith -- speaks us daily lives ways often unnoticed, seem obvious. It speaks us Capital Nation. It speaks us processes governing sovereignties 48 States. It speaks us counties, cities, towns, villages. It speaks us nations hemisphere, ac ross seas -- enslaved, well free. Sometimes fail hear heed voices freedom us pri vilege freedom old, old story. The destiny America proclaimed words prophecy spo ken first President first inaugural 1789 -- words almost directed, would seem, y ear 1941: "The preservation sacred fire liberty destiny republican model governm ent justly considered deeply, finally, staked experiment intrusted hands America n people." If lose sacred fire--if let smothered doubt fear -- shall reject dest iny Washington strove valiantly triumphantly establish. The preservation spirit faith Nation does, will, furnish highest justification every sacrifice may make cause national defense. In face great perils never encountered, strong purpose p rotect perpetuate integrity democracy. For muster spirit America, faith America. We retreat. We content stand still.

Kennedy Speech – After stop words are removed:

'Vice President Johnson, Mr. Speaker, Mr. Chief Justice, President Eisenhower, V ice President Nixon, President Truman, reverend clergy, fellow citizens, observe today victory party, celebration freedom -- symbolizing end, well beginning -- s ignifying renewal, well change. For I sworn I Almighty God solemn oath forebears l prescribed nearly century three quarters ago. The world different now. For man holds mortal hands power abolish forms human poverty forms human life. And yet r evolutionary beliefs forebears fought still issue around globe -- belief rights man come generosity state, hand God. We dare forget today heirs first revolution . Let word go forth time place, friend foe alike, torch passed new generation Am ericans -- born century, tempered war, disciplined hard bitter peace, proud anci ent heritage -- unwilling witness permit slow undoing human rights Nation always committed, committed today home around world. Let every nation know, whether wis hes us well ill, shall pay price, bear burden, meet hardship, support friend, op

pose foe, order assure survival success liberty. This much pledge -- more. To old allies whose cultural spiritual origins share, pledge loyalty faithful friends . United, little cannot host cooperative ventures. Divided, little -- dare meet powerful challenge odds split asunder. To new States welcome ranks free, pledge word one form colonial control shall passed away merely replaced far iron tyranny. We shall always expect find supporting view. But shall always hope find strongly supporting freedom -- remember that, past, foolishly sought power riding back tiger ended inside. To peoples huts villages across globe struggling break bonds mass misery, pledge best efforts help help themselves, whatever period required -- Communists may it, seek votes, right. If free society cannot help many poor, cannot save rich. To sister republics south border, offer special pledge -- convert good words good deeds -- new alliance progress -- assist free men free governments casting chains poverty. But peaceful revolution hope cannot become prey hostile powers. Let neighbors know shall join oppose aggression subversion anywhere Americas. And let every power know Hemisphere intends remain master house. To world assembly sovereign states, United Nations, last best hope age instruments war far outpaced instruments peace, renew pledge support--to prevent becoming merely forum invective -- strengthen shield new weak -- enlarge area writ may run. Finally, nations would make adversary, offer pledge request: sides begin anew quest peace, dark powers destruction unleashed science engulf humanity planned accidental self-destruction. We dare tempt weakness. For arms sufficient beyond doubt certain beyond doubt never employed. But neither two great powerful groups nations take comfort present course -- sides overburdened cost modern weapons, rightly alarmed steady spread deadly atom, yet racing alter uncertain balance terror stays hand mankind\'s final war. So let us begin anew -- remembering sides civility sign weakness, sincerity always subject proof. Let us never negotiate fear. But let us never fear negotiate. Let sides explore problems unite us instead belaboring problems divide us. Let sides, first time, formulate serious precise proposals inspection control arms -- bring absolute power destroy nations absolute control nations. Let sides seek invoke wonders science instead terrors. Together let us explore stars, conquer deserts, eradicate disease, tap ocean depths, encourage arts commerce. Let sides unite heed corners earth command Isaiah -- "undo heavy burdens ... let oppressed go free." And beachhead cooperation may push back jungle suspicion, let sides join creating new endeavor, new balance power, new world law, strong weak secure peace preserved. All finished first 100 days. Nor finished first 1,000 days, life Administration, even perhaps lifetime planet . But let us begin. In hands, fellow citizens, mine, rest final success failure course. Since country founded, generation Americans summoned give testimony national loyalty. The graves young Americans answered call service surround globe. Now trumpet summons us -- call bear arms, though arms need; call battle, though embattled -- call bear burden long twilight struggle, year year out, "rejoicing hope, patient tribulation" -- struggle common enemies man: tyranny, poverty, disease, war itself. Can forge enemies grand global alliance, North South, East West , assure fruitful life mankind? Will join historic effort? In long history world , generations granted role defending freedom hour maximum danger. I shrink responsibility -- I welcome it. I believe us would exchange places people generation. The energy, faith, devotion bring endeavor light country serve -- glow fire truly light world. And so, fellow Americans: ask country -- ask country. My fellow citizens world: ask America you, together freedom man. Finally, whether citizens America citizens world, ask us high standards strength sacrifice ask you. With good conscience sure reward, history final judge deeds, let us go forth lead land love, asking His blessing His help, knowing earth God\'s work must truly own.'

Nixon Speech – After stop words are removed:

'Mr. Vice President, Mr. Speaker, Mr. Chief Justice, Senator Cook, Mrs. Eisenhower, fellow citizens great good country share together: When met four years ago,

America bleak spirit, depressed prospect seemingly endless war abroad destructiv e conflict home. As meet today, stand threshold new era peace world. The central question us is: How shall use peace? Let us resolve era enter postwar periods of ten been: time retreat isolation leads stagnation home invites new danger abroad . Let us resolve become: time great responsibilities greatly borne, renew spirit promise America enter third century nation. This past year saw far-reaching resu lts new policies peace. By continuing revitalize traditional friendships, missio ns Peking Moscow, able establish base new durable pattern relationships among na tions world. Because America\'s bold initiatives, 1972 long remembered year grea test progress since end World War II toward lasting peace world. The peace seek world flimsy peace merely interlude wars, peace endure generations come. It impo rtant understand necessity limitations America\'s role maintaining peace. Unless America work preserve peace, peace. Unless America work preserve freedom, freedo m. But let us clearly understand new nature America\'s role, result new policies adopted past four years. We shall respect treaty commitments. We shall support v igorously principle country right impose rule another force. We shall continue, era negotiation, work limitation nuclear arms, reduce danger confrontation great powers. We shall share defending peace freedom world. But shall expect others sh are. The time passed America make every nation\'s conflict own, make every natio n\'s future responsibility, presume tell people nations manage affairs. Just res pect right nation determine future, also recognize responsibility nation secure future. Just America\'s role indispensable preserving world\'s peace, nation\'s role indispensable preserving peace. Together rest world, let us resolve move fo rward beginnings made. Let us continue bring walls hostility divided world long, build place bridges understanding -- despite profound differences systems govern ment, people world friends. Let us build structure peace world weak safe strong -- respects right live different system -- would influence others strength ideas , force arms. Let us accept high responsibility burden, gladly -- gladly chance build peace noblest endeavor nation engage; gladly, also, act greatly meeting re sponsibilities abroad remain great Nation, remain great Nation act greatly meeti ng challenges home. We chance today ever history make life better America -- ens ure better education, better health, better housing, better transportation, clea ner environment -- restore respect law, make communities livable -- insure God-g iven right every American full equal opportunity. Because range needs great -- r each opportunities great -- let us bold determination meet needs new ways. Just building structure peace abroad required turning away old policies failed, build ing new era progress home requires turning away old policies failed. Abroad, shi ft old policies new retreat responsibilities, better way peace. And home, shift old policies new retreat responsibilities, better way progress. Abroad home, key new responsibilities lies placing division responsibility. We lived long consequ ences attempting gather power responsibility Washington. Abroad home, time come turn away condescending policies paternalism -- "Washington knows best." A perso n expected act responsibly responsibility. This human nature. So let us encourag e individuals home nations abroad themselves, decide themselves. Let us locate r esponsibility places. Let us measure others themselves. That today I offer promi se purely governmental solution every problem. We lived long false promise. In t rusting much government, asked deliver. This leads inflated expectations, reduce d individual effort, disappointment frustration erode confidence government peop le do. Government must learn take less people people themselves. Let us remember America built government, people -- welfare, work -- shirking responsibility, se eking responsibility. In lives, let us ask -- government me, I myself? In challe nges face together, let us ask -- government help, I help? Your National Governm ent great vital role play. And I pledge Government act, act boldly lead boldly. But important role every one us must play, individual member community. From day forward, let us make solemn commitment heart: bear responsibility, part, live id eals -- together, see dawn new age progress America, together, celebrate 200th a nniversary nation, proud fulfillment promise world. As America\'s longest diffic

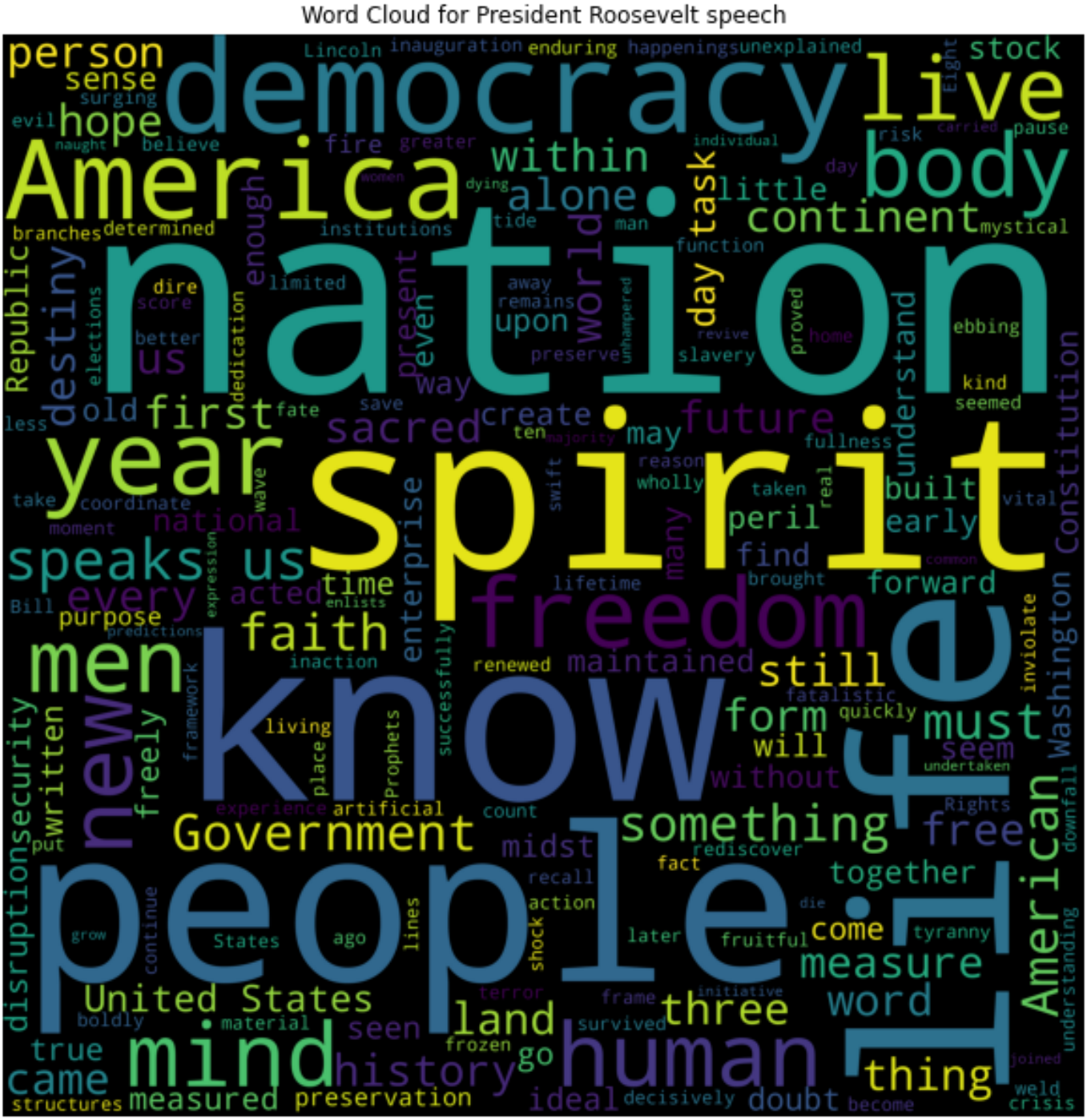
ult war comes end, let us learn debate differences civility decency. And let us reach one precious quality government cannot provide -- new level respect rights feelings one another, new level respect individual human dignity cherished birth right every American. Above else, time come us renew faith America. In recent ye ars, faith challenged. Our children taught ashamed country, ashamed parents, ash amed America\'s record home role world. At every turn, beset find everything wro ng America little right. But I confident judgment history remarkable times privi leged live. America\'s record century unparalleled world\'s history responsibili ty, generosity, creativity progress. Let us proud system produced provided freed om abundance, widely shared, system history world. Let us proud four wars engage d century, including one bringing end, fought selfish advantage, help others res ist aggression. Let us proud bold, new initiatives, steadfastness peace honor, m ade break-through toward creating world world known -- structure peace last, mer ely time, generations come. We embarking today era presents challenges great nat ion, generation, ever faced. We shall answer God, history, conscience way use ye ars. As I stand place, hallowed history, I think others stood me. I think dreams America, I think recognized needed help far beyond order make dreams come true. Today, I ask prayers years ahead I may God\'s help making decisions right Americ a, I pray help together may worthy challenge. Let us pledge together make next f our years best four years America\'s history, 200th birthday America young vital began, bright beacon hope world. Let us go forward confident hope, strong faith one another, sustained faith God created us, striving always serve His purpose.'

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) – 3 Marks

```
Top 3 words that occur the most number of times in Roosevelt inaugural
address
 --     22
It      13
We      10
The      9
Top 3 words that occur the most number of times in Kennedy inaugural a
ddress
 --     24
us      11
Let      8
let      8
Top 3 words that occur the most number of times in Nixon inaugural add
ress
 us     25
--      17
new     15
Let     13
```

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

Word Cloud for President Roosevelt Speech



Word Cloud for President Kennedy Speech

Word Cloud for President Kennedy Speech