Machine Learning Project

NAME: SHOUNACK MANDAL

COURSE: PGP - DSBA Online Sep.

Date: 20/ March / 2022

Contents

LIST OF FIGURE

LIST OF TABLES

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 DICTIONARY

1. vote: Party choice: Conservative or Labour
2. age: in years
3. economic.cond.national: Assessment of current national economic conditions, 1 to 5.
4. economic.cond.household: Assessment of current household economic conditions, 1 to 5.
5. Blair: Assessment of the Labour leader, 1 to 5.
6. Hague: Assessment of the Conservative leader, 1 to 5.
7. Europe: an 11-point scale that measures respondents' attitudes toward European integration. High scores represent 'Eurosceptic' sentiment.
8. political.knowledge: Knowledge of parties' positions on European integration, 0 to 3.
9. gender: female or male.

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

DATASET

	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

Table 1 Dataset Election Data

Information on dataset

```
RangeIndex: 1525 entries, 0 to 1524
Data columns (total 9 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   vote                   1525 non-null   object
 1   age                    1525 non-null   int64
 2   economic.cond.national 1525 non-null   int64
 3   economic.cond.household 1525 non-null  int64
 4   Blair                  1525 non-null   int64
 5   Hague                  1525 non-null   int64
 6   Europe                 1525 non-null   int64
 7   political.knowledge    1525 non-null   int64
 8   gender                 1525 non-null   object
dtypes: int64(7), object(2)
```

Table 2 Information on Dataset

Summary of the dataset

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

Table 3 Dataset Summary

Duplicates

```
Total no of duplicate values =  8
```

	vote	age	economic.cond.national	economic.cond.household	Blair	Hague	Europe	political.knowledge	gender
67	Labour	35	4	4	5	2	3	2	male
626	Labour	39	3	4	4	2	5	2	male
870	Labour	38	2	4	2	2	4	3	male
983	Conservative	74	4	3	2	4	8	2	female
1154	Conservative	53	3	4	2	2	6	0	female
1236	Labour	36	3	3	2	2	6	2	female
1244	Labour	29	4	4	4	2	2	2	female
1438	Labour	40	4	3	4	2	2	2	male

Table 4 Duplicates and its values.

Uniques Values of the categorical variables

Vote:

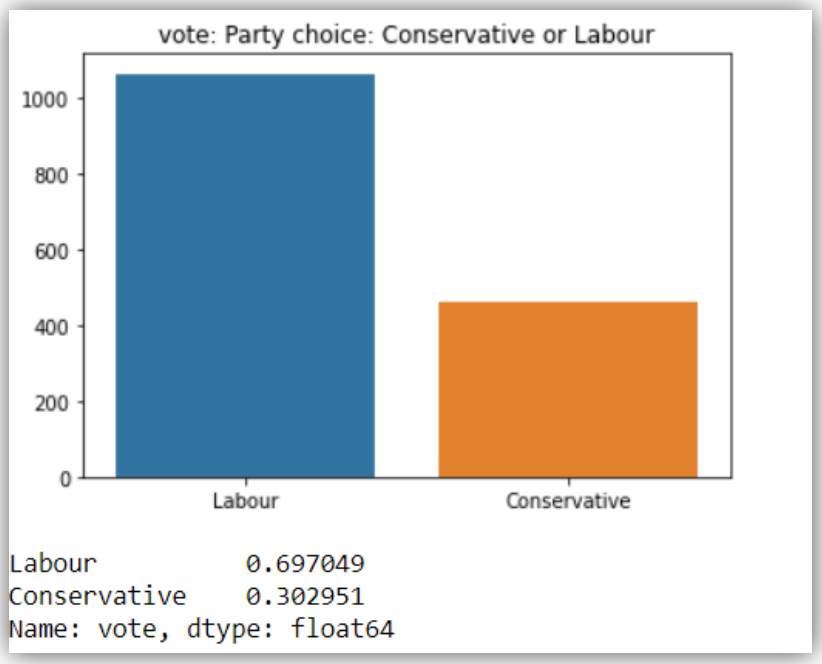


Figure 1 Categorical Variable : Vote

Gender:

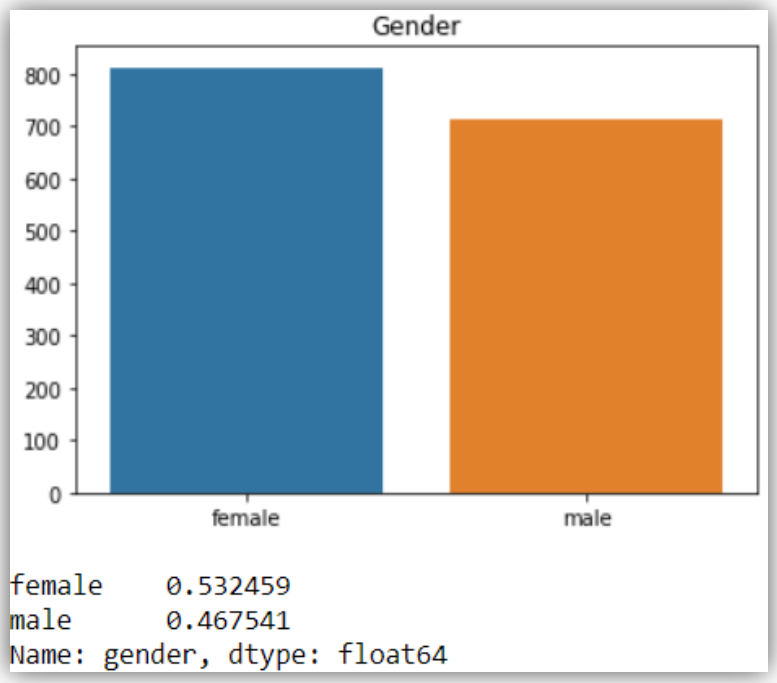


Figure 2 Categorical variable Gender

Inferences

- Two categorical variables: Vote and age variables with 2 unique values
- There is no null values in the dataset
- 69% vote by the labor party and 30 % by the side of conservative party.
- 53% of the voters are female and 46% are male.
- Age is continuous variable
- Minimum age for voters is 24
- Maximum age of the voters is 93
- Average age is 54
- maximum Assessment of current national economic conditions is 5
- Average Assessment of current household economic conditions is 3
- Minimum Assessment of the Labor leader is 1
- Hague: average Assessment of the Conservative leader is 2.7 whereas Blair: average Assessment of the Labour leader is 3.3

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

Histogram Distribution

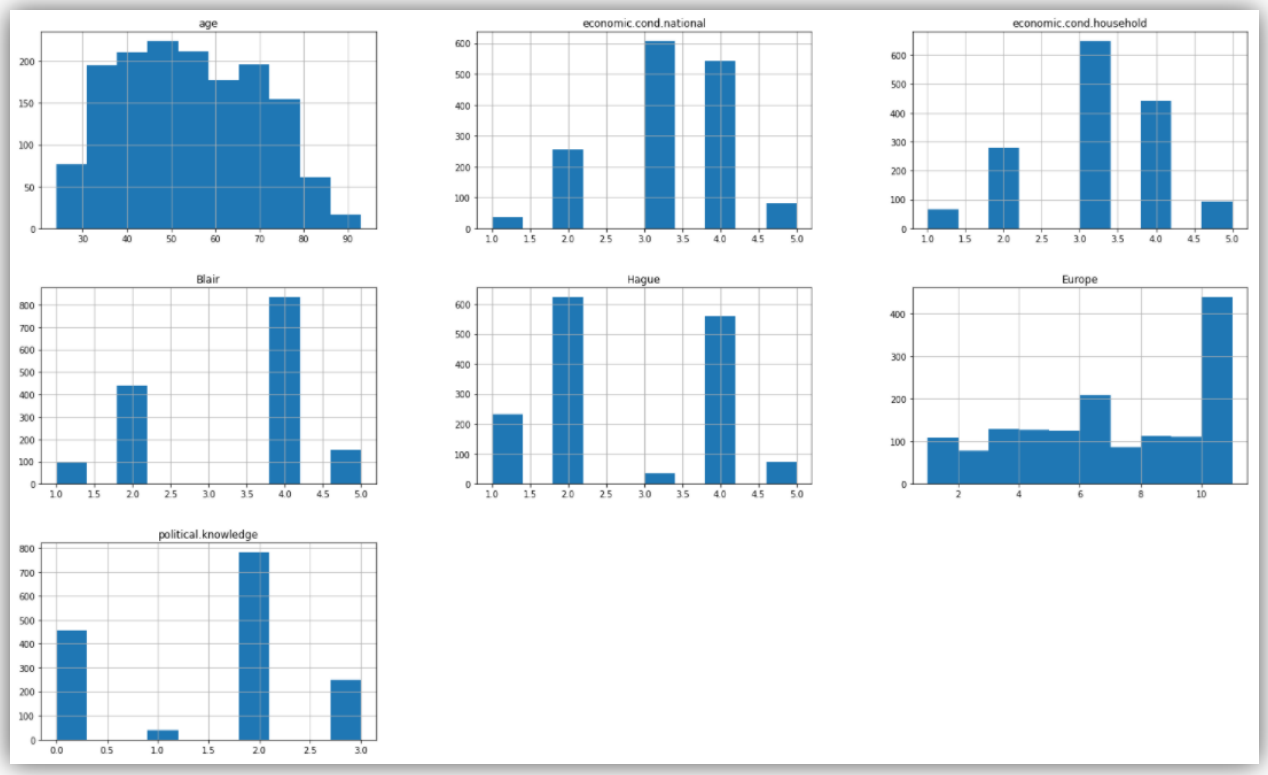


Figure 3 Histogram

Inferences

- Only the age variable is normally distributed and other variables has multimodalskewness seen.
- Only economic.cond.household have outliers

Univariate Analysis - Box plot Distribution

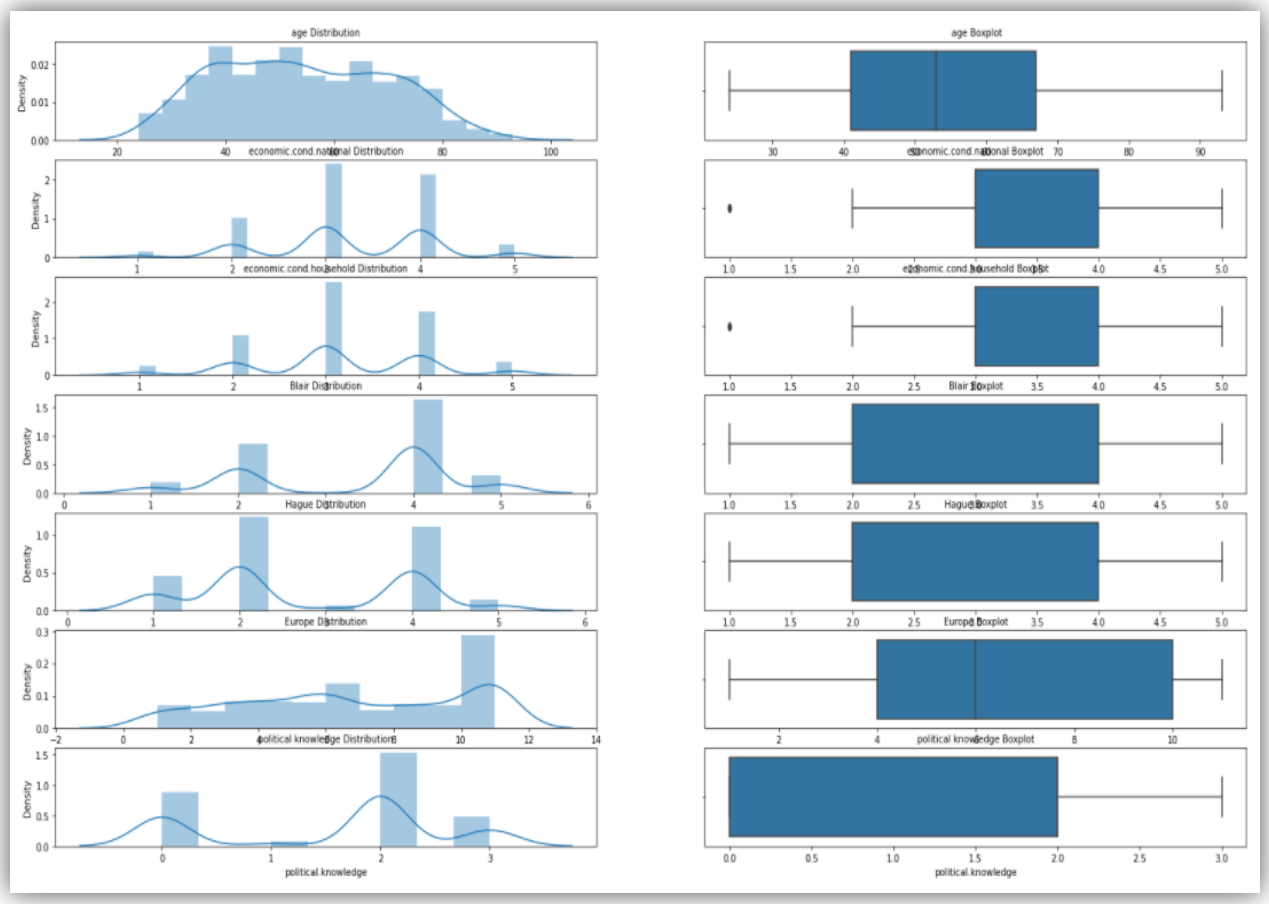


Figure 4 Histogram distribution and boxplots

Inferences

- only age is normally distributed others are multimodel skewness see
- only economic.cond.national and economic.cond.household have outliers.

Multivariate Analysis: Pair Plot

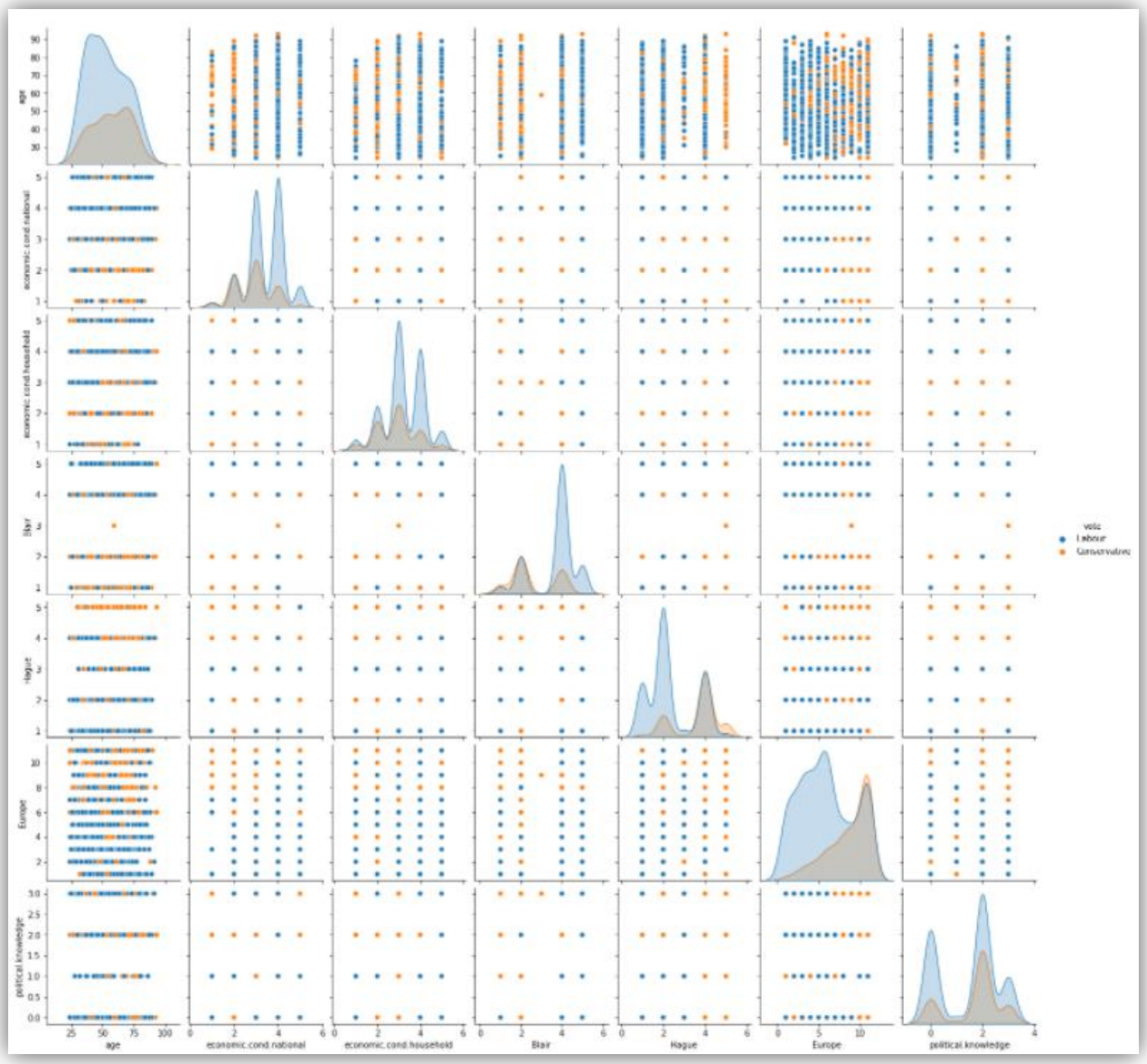


Figure 5 Pair Plot

Inferences

- There is no linear relationship between variables
- Some of the attributes look like they may have an exponential distribution
- Conservative party: Knowledge of parties' positions on European integration is unknown

Multivariate Analysis: Correlation Heat Map Plot

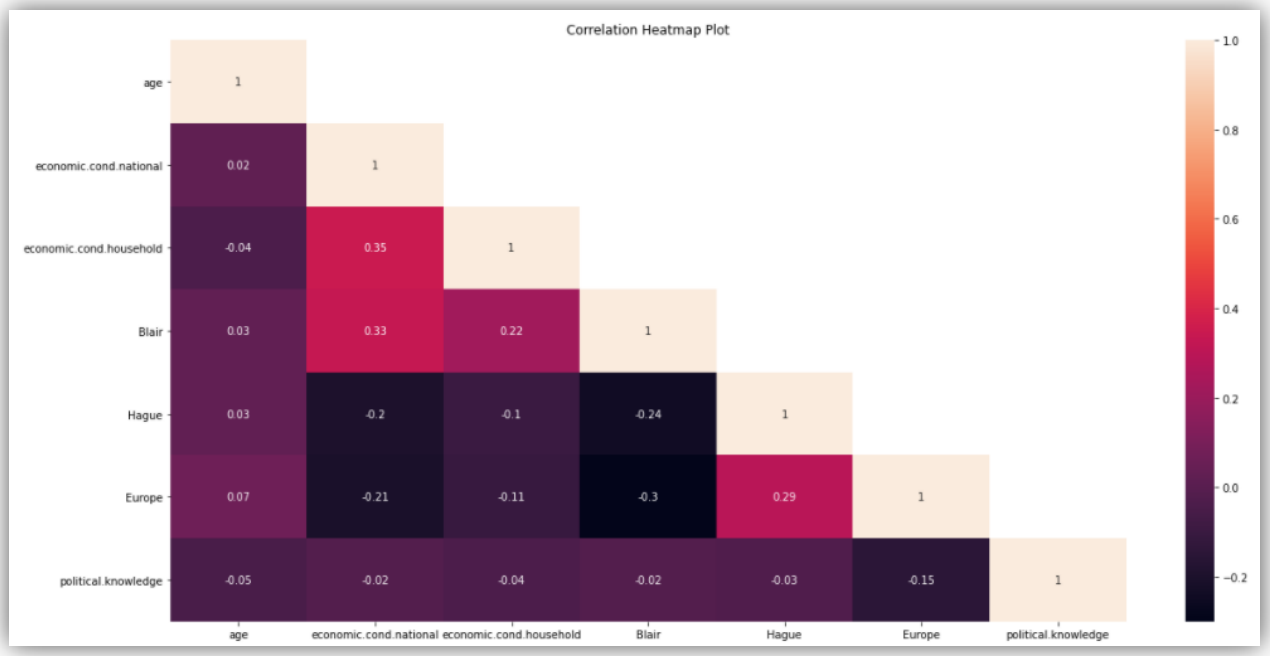


Figure 6 Correlation heat map

Inferences

There is very less correlation between the variables

➢ The highest positive correlation is seen between “economic_cond_national” and “economic_cond_household” (35%) with nearly similar results seen from “Blair” and “economic_cond_national” (35%)

➢ The highest negative correlation is seen between “Blair” and “Europe” (29%) with nearly similar results seen from “Blair” and “Hague” (24%)

➢ so, there is less or no chance of multi_collinearity

Bivariate Analysis: Strip plot between “Hague” and “Age”

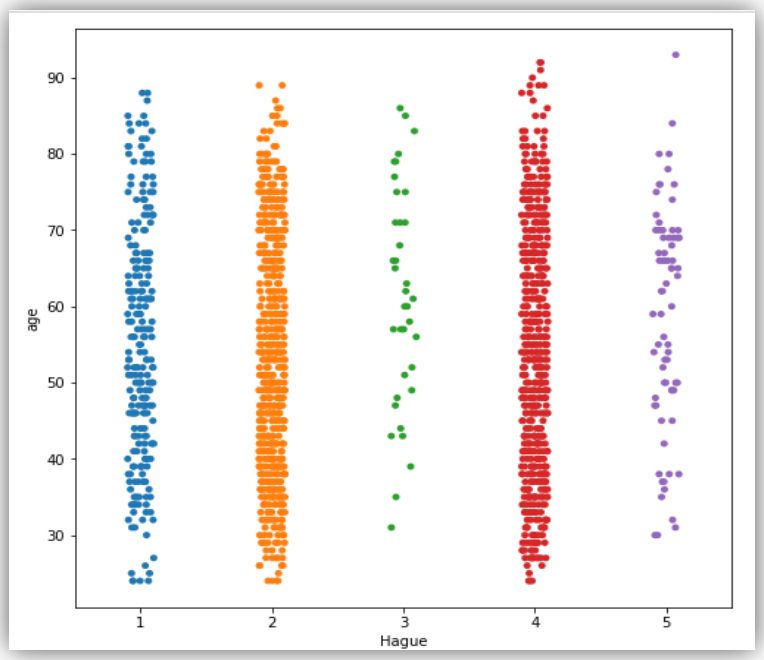


Figure 7 Stir plot "Hague" and "Age"

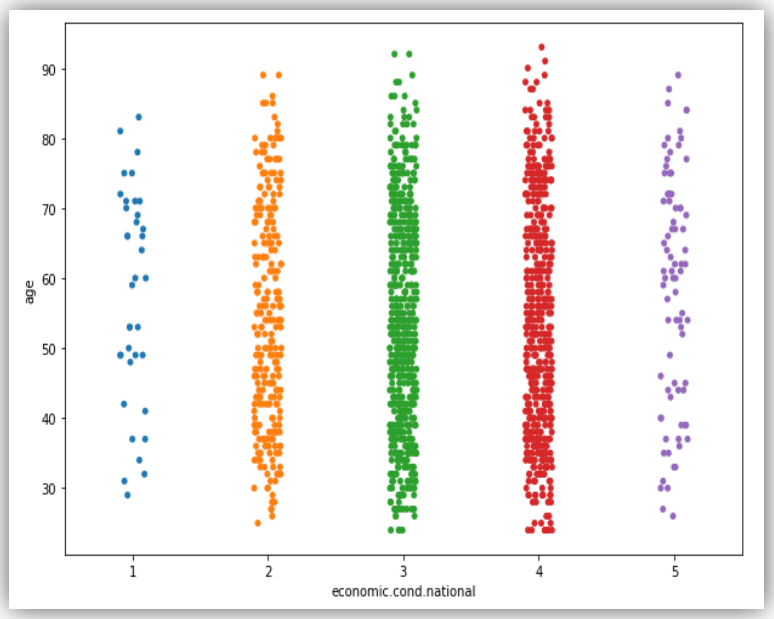


Figure 8 Stir plot "economic.cond.national" and "Age"

Above plot is a strip plot with jitter as True that really shows the distribution points on the assessment of the Conservative leader “Hague” on voters of various age. more voters are distributed in 2 and 4 group

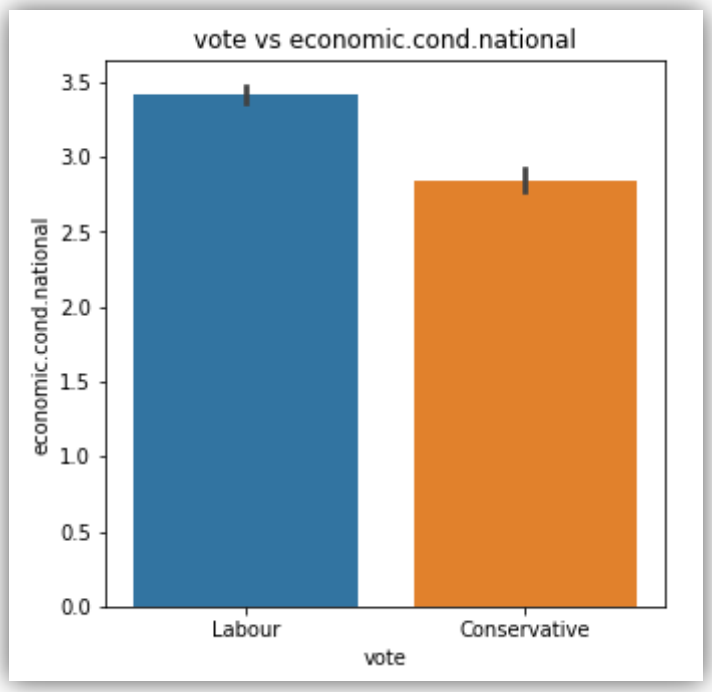


Figure 9 Barplot vote vs economic.cond.national

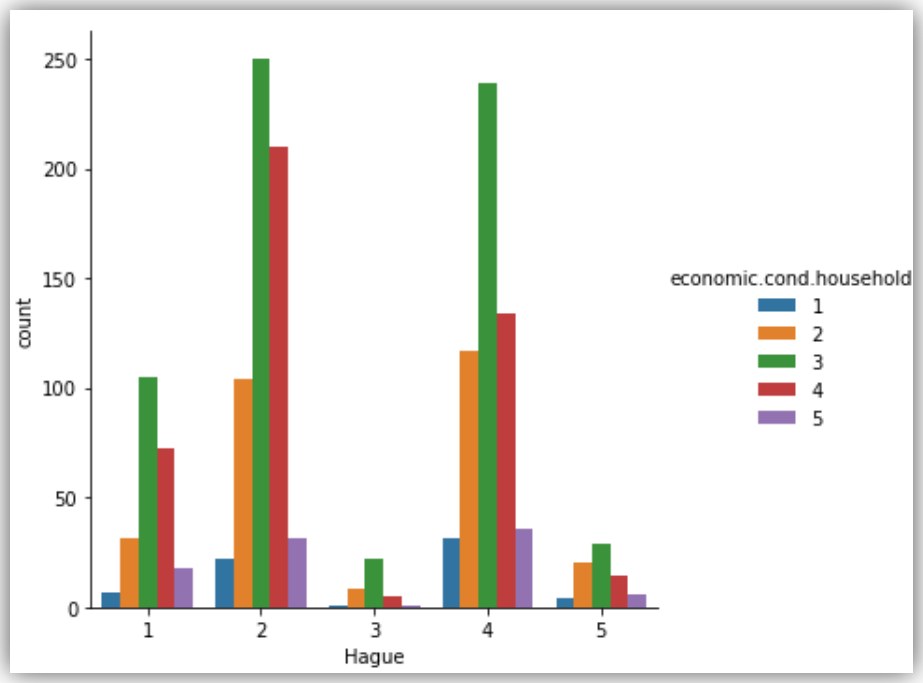


Figure 10 Catplot Analysis - Hague (count) on economic.cond.household

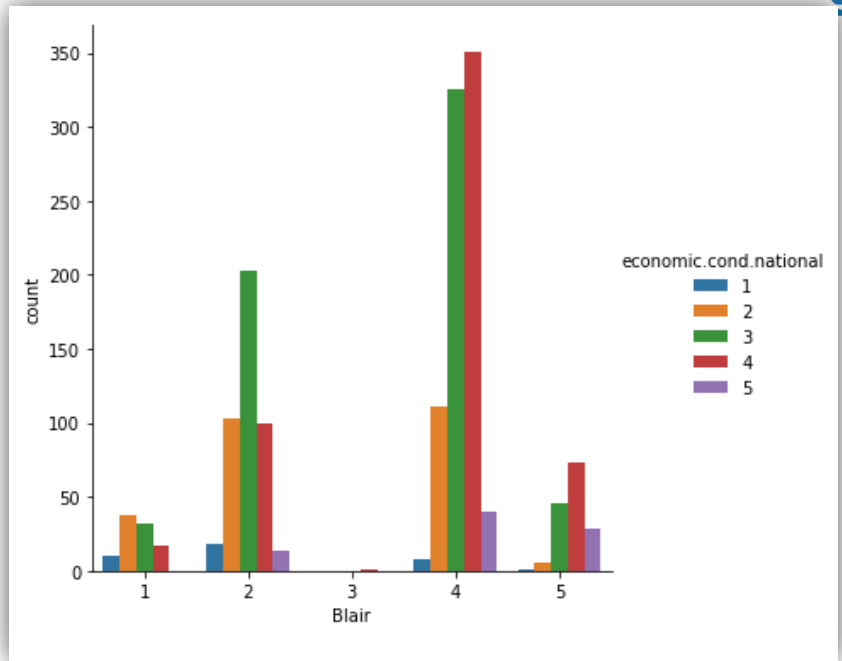


Figure 11Catplot Analysis - Blair(count) on economic.cond.national

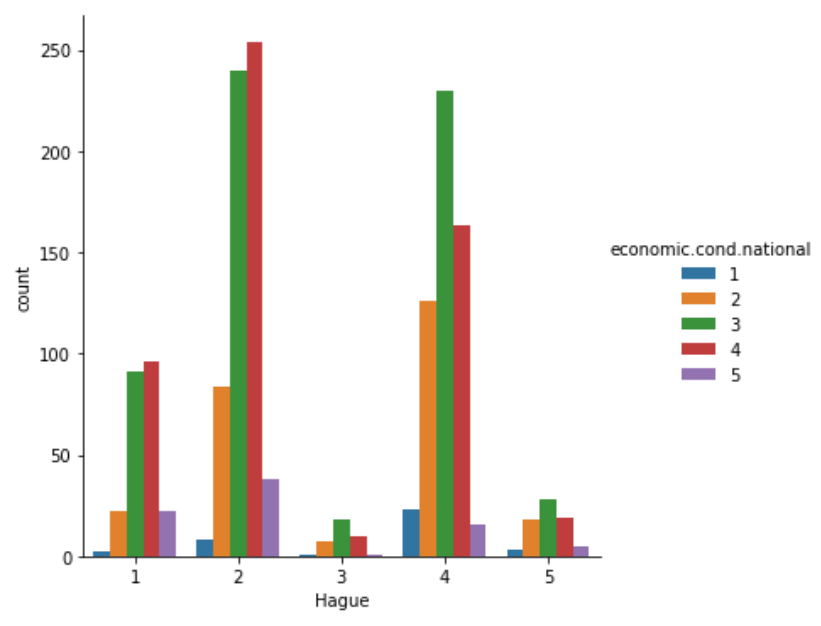


Figure 12 Catplot Analysis - Hague(count) on economic.cond.national

Assessment of current national economic conditions with Blair shows no 3 cluster have very less distribution where as no 4 cluster have more distribution

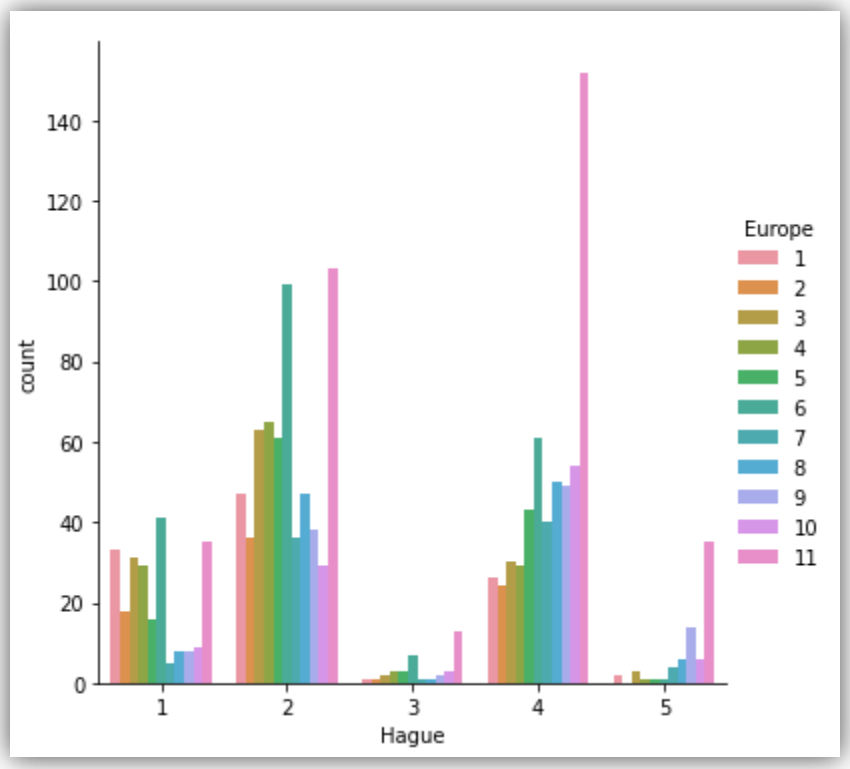


Figure 13Catplot Analysis - Hague(count) on Europe

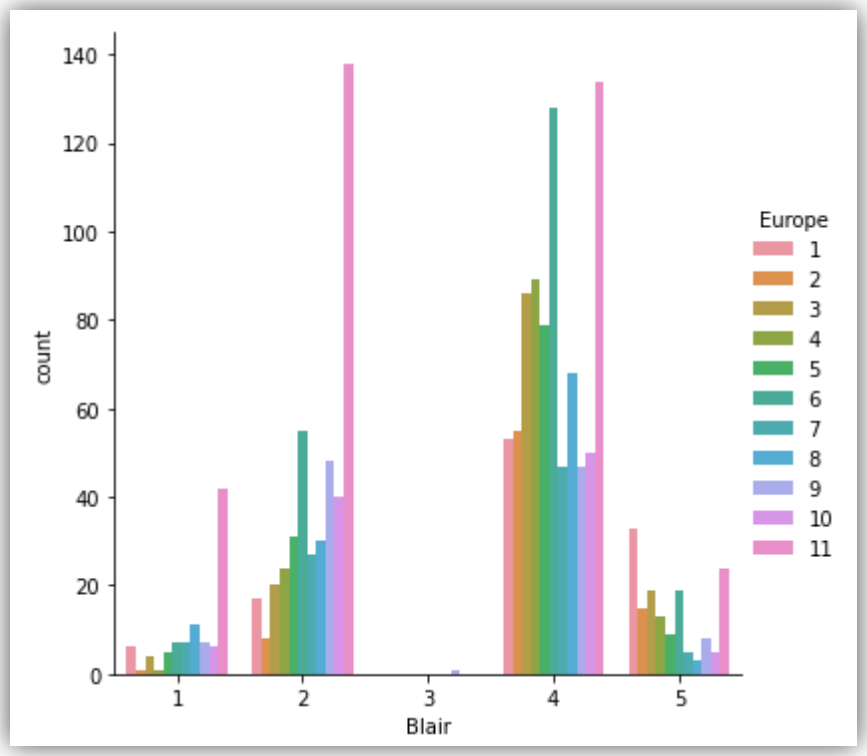


Figure 14 Catplot Analysis - Blair(count) on Europe

Boxplot of all the variables

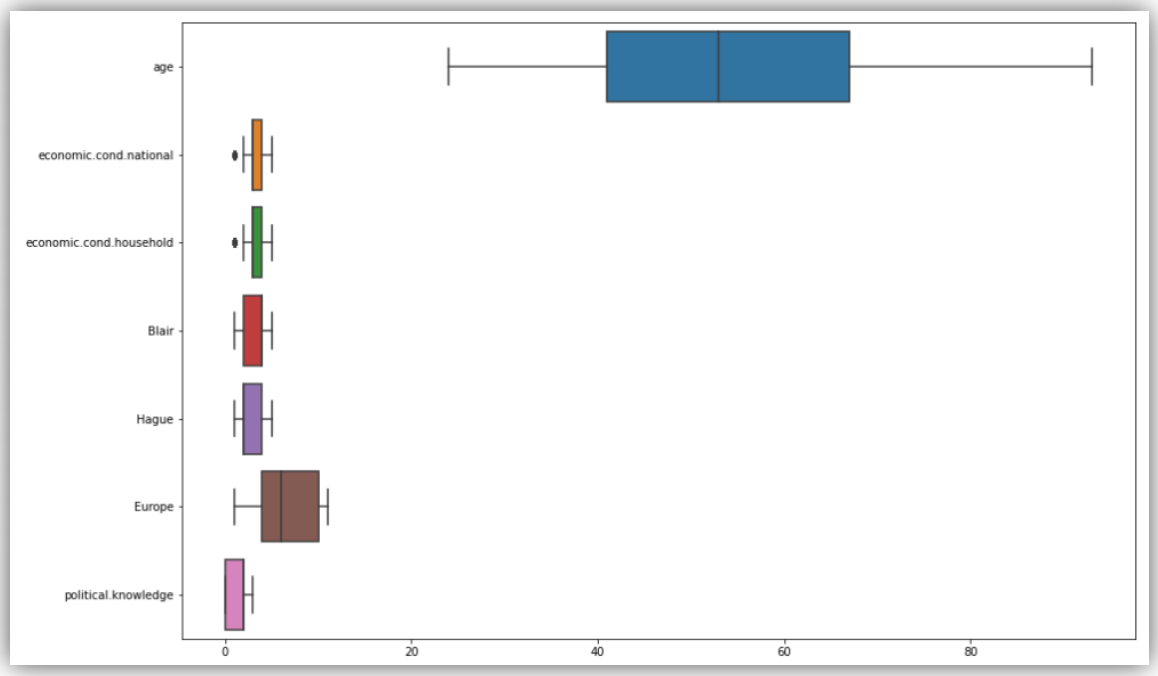


Figure 15 Boxplot without outlier treatment

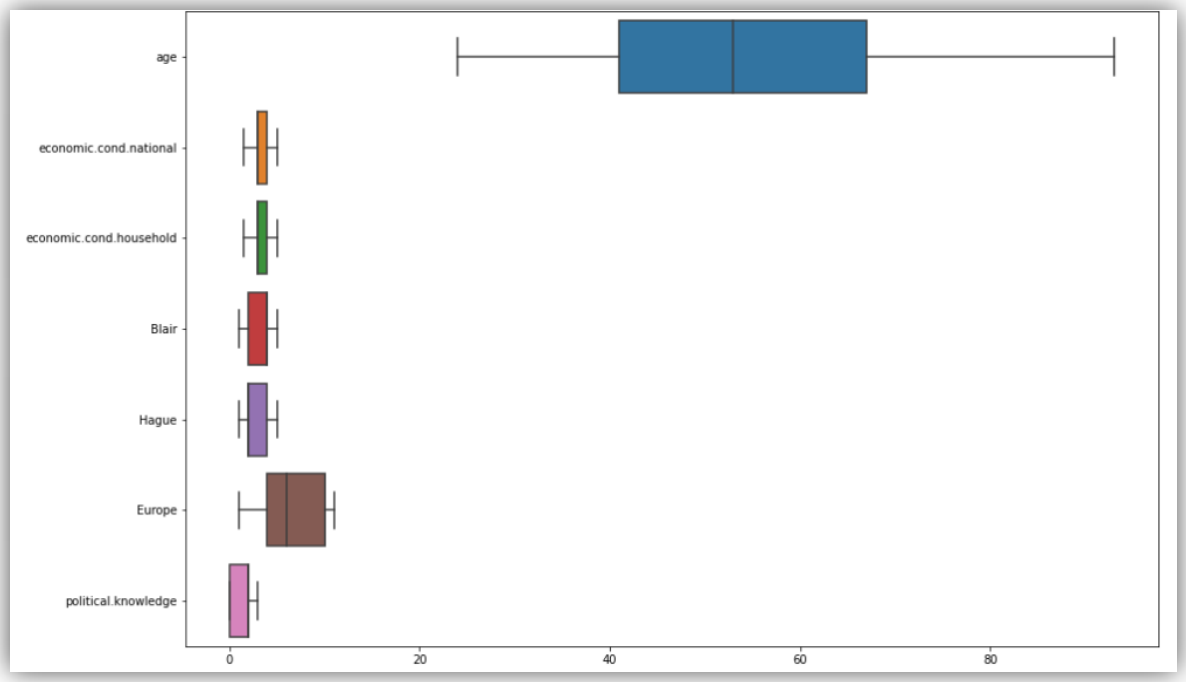


Figure 16 Boxplot with outliers treatment done

Inferences

- All outliers are removed for further analysis
- There is no linear relationship between variables
- Some of the attributes look like they may have an exponential distribution
- Conservative party: Knowledge of parties' positions on European integration is unknown
- can be easily observed that relatively younger people have voted for “Labour” party in comparison to that of older people who voted for “Conservative” party.
- There is an evenly distributed number of people when it comes to thier knowledge about their party's
  position on European integration.
- Majority of European people have voted for “Labour” party
- There exists an outlier for economic.cond.household and economic.cond.national

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)

Converting Object variables to categorical variables

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


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

Table 5 Conversion of object to categorical values

```
1    0.697049
0    0.302951
Name: vote, dtype: float64
```

Table 6 Check for the value type after conversion

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1525 entries, 0 to 1524
Data columns (total 9 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   vote                    1525 non-null   int8
 1   age                     1525 non-null   int64
 2   economic.cond.national  1525 non-null   int64
 3   economic.cond.household 1525 non-null   int64
 4   Blair                   1525 non-null   int64
 5   Hague                   1525 non-null   int64
 6   Europe                  1525 non-null   int64
 7   political.knowledge     1525 non-null   int64
 8   gender                  1525 non-null   int8
dtypes: int64(7), int8(2)
memory usage: 86.5 KB
```

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

Table 7 Converting int8 to int64 variables

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

Table 8the variables are converted into int64 datatype for model prediction

Inferences

- Codes are an array of integers which are the positions of the actual values in the categories array.
- Here vote and gender are categorical variables are now converted into integers using codes
- All the variables in the data frame are integers

Scaling the data

- Differences in the scales across input variables may increase the difficulty of the problem being modelled.
- This means that you are transforming your data so that it fits within a specific scale, like 0-100 or 0-1.
- Usually, the distance-based methods (E.g.: KNN) would require scaling as it is sensitive to extreme difference and can cause a bias.
- Tree-based method uses split method (E.g.: Decision Trees) would not require scaling in general as its unnecessary
- In this dataset, age is only continuous variable and rest of the variables have 1 to 5. Age variable is only scaled because it is continuous variable
- The method of scaling performed only on the 'age' variable is the Z-score scaling.
- Z-score scaling is the most common form of scaling that takes from the formula (x – mean) /Standard deviation.
- All the model prediction will be done on the scaled data.

Train and Test split

Classification of the scalled data into X(Independent _ Trained Data) and Y (Dependent _ Test Data)

```
The training set for the independent variables: (1067, 8)
The training set for the dependent variable: (1067, 1)
The test set for the independent variables: (458, 8)
The test set for the dependent variable: (458, 1)
```

Table 9 Train- Test Split

Inferences

- spilting the dataset into train and test set to build Logistic regression and LDA model (70:30)
- X_train :70% of data randomly chosen from the 8 columns. These are training independent variables
- X_test :30% of data randomly chosen from the 8 columns. These are test independent variables
- y_train :70% of data randomly chosen from the "vote" column. These are training dependent variables
- y_test :30% of data randomly chosen from the "vote" columns. These are test dependent variables

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

Logistic Regression Model

Logistic regression is a fundamental classification technique.It belongs to the group of linear classifiers and is somewhat similar to polynomial and linear regression.It is the go-to method for binary classification problems (problems with two class values). SKLearn model is been used.

Applying GridSearchCV for Logistic Regression

	0	1
0	0.614904	0.385096
1	0.187476	0.812524
2	0.190163	0.809837
3	0.164022	0.835978
4	0.053384	0.946616

	0	1
0	0.932007	0.067993
1	0.688387	0.311613
2	0.333625	0.666375
3	0.476424	0.523576
4	0.159520	0.840480

Table 10 The probabilities on the test and training set

Fit the model to the training set

We now fit our model to the GridSearchCV for Logistic Regression model by training the model with our independent variable and dependent variables.

Inference

Using GridsearchCV, we input various parameters like 'max_iter', 'penalty',solver', 'tol' which will helps us to find best grid for prediction of the better model max_iter is an integer (100 by default) that defines the maximum number of iterations by the solver during model fitting.
Solver is a string ('liblinear' by default) that decides what solver to use for fitting the model. Other options are 'newton-cg', 'lbfgs', 'sag', and 'saga'.

- Accuracy score of training data:83.9%
- Accuracy score of test data:82.3%

Interception of the model

```
The intercept for the model is : [3.41372291]
```

Table 11 The Intercept of the Final Model

The Coefficients for each of the independent attributes

```
The coefficient for age is -0.02061149603608451
The coefficient for economic.cond.national is 0.33511260131867593
The coefficient for economic.cond.household is 0.1586947887622719
The coefficient for Blair is 0.5714163761002934
The coefficient for Hague is -0.832124018843385
The coefficient for Europe is -0.23715909065832222
The coefficient for political.knowledge is -0.4775201244813325
The coefficient for gender is 0.2881911839338333
```

Table 12 The coefficient of the features

Plot for all feature importance in graph

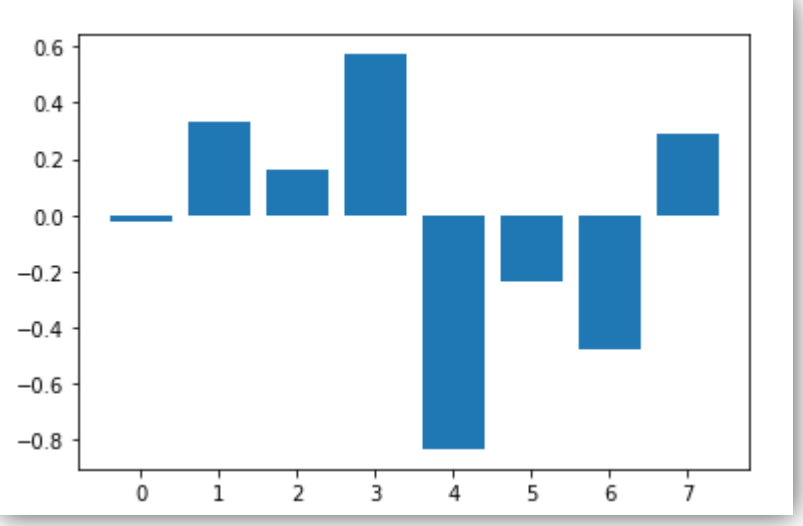


Figure 17 Features Importance shown in graph

Inference

- The coefficients for each of the independent attributes

- The sign of a regression coefficient tells you whether there is a positive or negative correlation between each independent variable the dependent variable. A positive coefficient indicates that as the value of the independent variable increases, the mean of the dependent variable also tends to increase. A negative coefficient suggests that as the independent variable increases, the dependent variable tends to decrease.
- economic.cond.national have more positive coeffiecient . A positive coefficient indicates that as the value of the independent variable increases, the mean of the dependent variable also tends to increase the vote

Linear Discriminant Analysis

- Linear Discriminant Analysis (LDA) is a dimensionality reduction technique which is commonly used for the supervised classification problems.
- It is used for modeling differences in groups i.e., separating two or more classes. It is used to project the features in higher dimension space into a lower dimension space.
- Library used in LDA is sklearn

Applying GridSearchCV for LDA

```
GridSearchCV(cv=7, estimator=LogisticRegression(), n_jobs=-1,
             param_grid={'max_iter': [1000, 100], 'penalty': ['l2'],
                         'solver': ['saga'], 'tol': [1e-05, 0.001]},
             scoring='accuracy')
```

Figure 18 GRIDSEARCHCV

Prediction on the training and test set

	0	1
0	0.620313	0.379687
1	0.203781	0.796219
2	0.181892	0.818108
3	0.174803	0.825197
4	0.050110	0.949890

	0	1
0	0.937942	0.062058
1	0.683460	0.316540
2	0.346059	0.653941
3	0.469925	0.530075
4	0.170748	0.829252

Table 13 The probabilities for the training and test data

Inferences

- Using GridsearchCV, we input various parameters like 'max_iter', 'penalty',solver', 'tol' which will helps us to find best grid for prediction of the better model
- max_iter is an integer (100 by default) that defines the maximum number of iterations by the solver during model fitting.
- solver is a string ('liblinear' by default) that decides what solver to use for fitting the model. Other options are 'newton-cg', 'lbfgs', 'sag', and 'saga'.
- here ‘solver':['svd', 'lsqr', 'eigen'] are used with others parameters has default
- ‘svd’: Singular value decomposition (default). Does not compute the covariance matrix, therefore this solver is recommended for data with many features.
- ‘lsqr’: Least squares solution. Can be combined with shrinkage or custom covariance estimator.
- ‘eigen’: Eigenvalue decomposition. Can be combined with shrinkage or custom covariance estimator.
- bestgrid:{'solver': 'svd'}
- Training Data Class Prediction with a cut-off value of 0.5
- Test Data Class Prediction with a cut-off value of 0.5
- Accuracy score of training data:83.69%
- Accuracy score of test data:81.87%

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

KNN MODEL

KNN is a non-parametric and lazy learning algorithm. Non-parametric means there is no assumption for underlying data distribution.In KNN, K is the number of nearest neighbors. The number of neighbors is the core deciding factor.

KNN CLASSIFIER MODEL

First, import the KNeighborsClassifier module and create KNN classifier object by passing argument number of neighbors in KNeighborsClassifier() function.

- Then, fit your model on the train set using fit() and perform prediction on the test set using predict().
- Let us build KNN classifier model for k=15.

MCE

```
[0.2379912663755459,
 0.20960698689956336,
 0.19650655021834063,
 0.19432314410480345,
 0.18558951965065507,
 0.18122270742358082,
 0.1746724890829694,
 0.17248908296943233,
 0.1834061135371179,
 0.1834061135371179]
```

Table 14 Misclassification error table

PLOTING MISCLASSIFICATION ERROR VS k (WITH k VALUE ON THE X-AXIS)

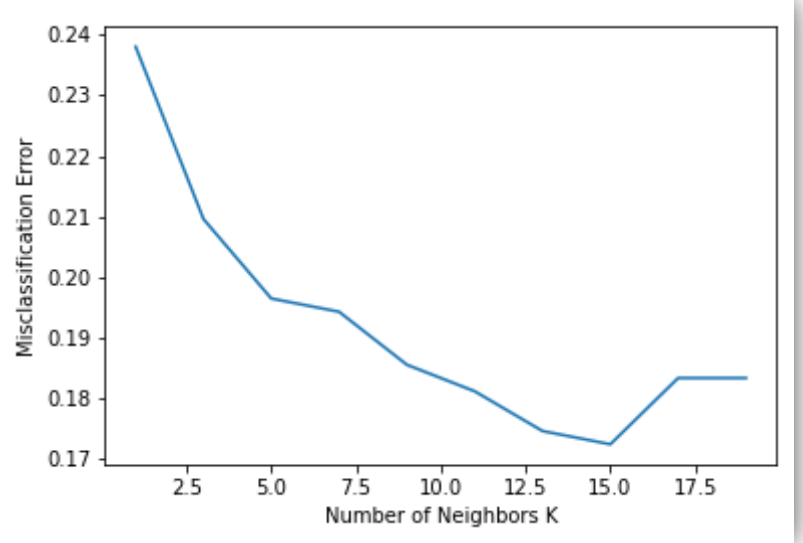


Figure 19 plot misclassification error vs k

PERFORMANCE MATRIX ON THE TRAINING AND TEST DATA AND THEIR ACCURACY SCORES

```
0.8434864104967198
[[237  95]
 [ 72 663]]
              precision    recall  f1-score   support

           0       0.77      0.71      0.74       332
           1       0.87      0.90      0.89       735

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

Table 15Performance Matrix on train data set

```
0.8275109170305677
[[ 92  38]
 [ 41 287]]
              precision    recall  f1-score   support

           0       0.69      0.71      0.70       130
           1       0.88      0.88      0.88       328

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

Table 16 Performance Matrix on test data set

GAUSSIAN NAÏVE BAYES

Naive Bayes is a classification technique based on the Bayes theorem. It is a simple but powerful algorithm for predictive modelling under supervised learning algorithms.

Gaussian Naive Bayes – This is a variant of Naive Bayes which supports continuous values and has an assumption that each class is normally distributed. All we would have to do is estimate the mean and standard deviation of the continuous variable

```
0.8331771321462043
[[240  92]
 [ 86 649]]
              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
```

Table 17 Accuraccy, Confussion and Classification Matrix of on Train data

```
0.8253275109170306
[[ 94  36]
 [ 44 284]]
              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
```

Table 18 Accuraccy, Confussion and Classification Matrix of on Test data

Accuracy score of training data:83.3%
Accuracy score of test data:82.5%

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

Model Tuning

Tuning is the process of maximizing a model's performance without overfitting or creating too high of a variance. In machine learning, this is accomplished by selecting appropriate "hyperparameters."
Hyperparameters can be thought of as the "dials" or "knobs" of a machine learning model. Choosing an appropriate set of hyperparameters is crucial for model accuracy, but can be computationally challenging. Hyperparameters differ from other model parameters in that they are not learned by the model automatically through training methods. Instead, these parameters must be set manually. Many methods exist for selecting appropriate hyperparameters.

Grid Search

Grid Search, also known as parameter sweeping, is one of the most basic and traditional methods of hyperparametric optimization. This method involves manually defining a subset of the hyperparametric space and exhausting all combinations of the specified hyperparameter subsets. Each combination's performance is then evaluated, typically using cross-validation, and the best performing hyperparametric combination is chosen.

Bagging with randomforest

A Bagging classifier is an ensemble meta-estimator that fits base classifiers each on random subsets of the original dataset and then aggregate their individual predictions (either by voting or by averaging) to form a final prediction. Such a meta-estimator can typically be used to reduce the variance of a black-box estimator (e.g., RandomForest), by introducing randomization into its construction procedure and then making an ensemble out of it.

We now fit randomforest classifier model to the bagging model by training the model with our independent variable and dependent variables.At this point, you have the classification model defined.

```
BaggingClassifier(base_estimator=RandomForestClassifier(max_depth=4,
                                                        max_features=5,
                                                        min_samples_leaf=25,
                                                        min_samples_split=50,
                                                        n_estimators=101),
                  n_estimators=101, random_state=1)
```

Table 19 Bagging with randomforest

The Probability on the training and test set

	0	1
0	0.471364	0.528636
1	0.313726	0.686274
2	0.345166	0.654834
3	0.128110	0.871890
4	0.160473	0.839527

	0	1
0	0.780111	0.219889
1	0.785020	0.214980
2	0.207929	0.792071
3	0.347034	0.652966
4	0.314477	0.685523

Table 20 Probability for training and test dataset

Accuracy score of training data:83.59%
Accuracy score of test data:81.65%

Boosting

Boosting is an ensemble strategy that is consecutively builds on weak learners in order to generate one final strong learner. A weak learner is a model that may not be exactly accurate or may not take many predictors into account. By building a weak model, making conclusions about the various feature importance's and parameters, and then using those conclusions to build a new, stronger model, Boosting can effectively convert weak learners into a strong learner.

GridSearchCV ADA boosting

AdaBoost uses decision stumps as weak learners. A Decision Stump is a Decision Tree model that only splits off at one level, ergo the final prediction is based off only one feature. When AdaBoost makes its first Decision Stump, all observations are weighted evenly.

The Probability on the training and test set

	0	1
0	0.507082	0.492918
1	0.493435	0.506565
2	0.489977	0.510023
3	0.489038	0.510962
4	0.496081	0.503919

	0	1
0	0.515603	0.484397
1	0.509824	0.490176
2	0.497409	0.502591
3	0.499906	0.500094
4	0.490884	0.509116

Table 21 Probability for training and test dataset

Accuracy score of training data:84.72%
Accuracy score of test data:81.87%

Gradient Boosting

GB builds an additive model in a forward stage-wise fashion; it allows for the optimization of arbitrary differentiable loss functions. In each stage n_classes_ regression trees are fit on the negative gradient of the binomial or multinomial deviance loss function. Binary classification is a special case where only a single regression tree is induced.

The Probability on the training and test set

	0	1
0	0.666601	0.333399
1	0.294602	0.705398
2	0.172143	0.827857
3	0.182579	0.817421
4	0.207914	0.792086

	0	1
0	0.865302	0.134698
1	0.861115	0.138885
2	0.204126	0.795874
3	0.207006	0.792994
4	0.276934	0.723066

Table 22 The Probability on the training and test set

- Accuracy score of training data:86.59%
- Accuracy score of test data:82.96%

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.

Logistic Regression Model

Logistic regression is a fundamental classification technique.It belongs to the group of linear classifiers and is somewhat similar to polynomial and linear regression.It is the go-to method for binary classification problems (problems with two class values). SKLearn model is been used.

Confusion matrix on the training and test data

Figure 20 Confusion matrix on the training and test data

Inferences

Training data:
True Negative : 196 False Positive : 105
False Negative : 68 True Positive : 666

Test data:
True Negative : 113 False Positive : 46
False Negative : 35 True Positive : 291

Classification Report of training and test data

	precision	recall	f1-score	support
0	0.77	0.68	0.72	332
1	0.86	0.91	0.88	735
accuracy			0.84	1067
macro avg	0.82	0.79	0.80	1067
weighted avg	0.83	0.84	0.83	1067

	precision	recall	f1-score	support
0	0.69	0.65	0.67	130
1	0.86	0.89	0.88	328
accuracy			0.82	458
macro avg	0.78	0.77	0.77	458
weighted avg	0.82	0.82	0.82	458

Table 23 Classification Report of training and test data

AUC and ROC for the training and test data

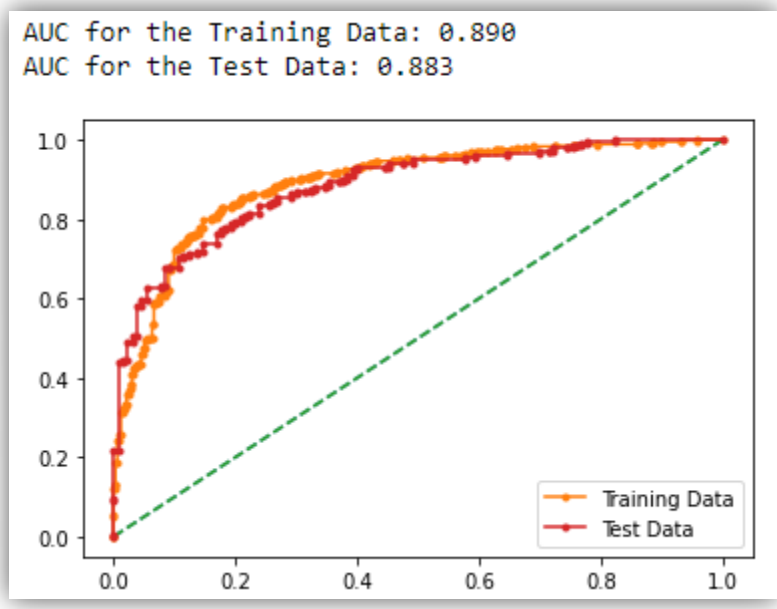


Figure 21 AUC and ROC for the training and test data

Inferences

Train Data:

➢ AUC: 89%
➢ Accuracy: 84%
➢ precision : 86%
➢ recall : 91%
➢ f1 :88%

Test Data:

➢ AUC: 88.3%
➢ Accuracy: 82%
➢ precision: 86%
➢ recall : 89%
➢ f1 : 88%

Training and Test set results are almost similar, this proves no overfitting or underfitting.

LDA

LDA_train_precision	0.87
LDA_train_recall	0.9
LDA_train_f1	0.88

LDA_test_precision	0.87
LDA_test_recall	0.88
LDA_test_f1	0.87

Table 24 Inferences on precision, recall and f1 for training and test data

CONFUSION MATRIX FOR TRAIN DATA

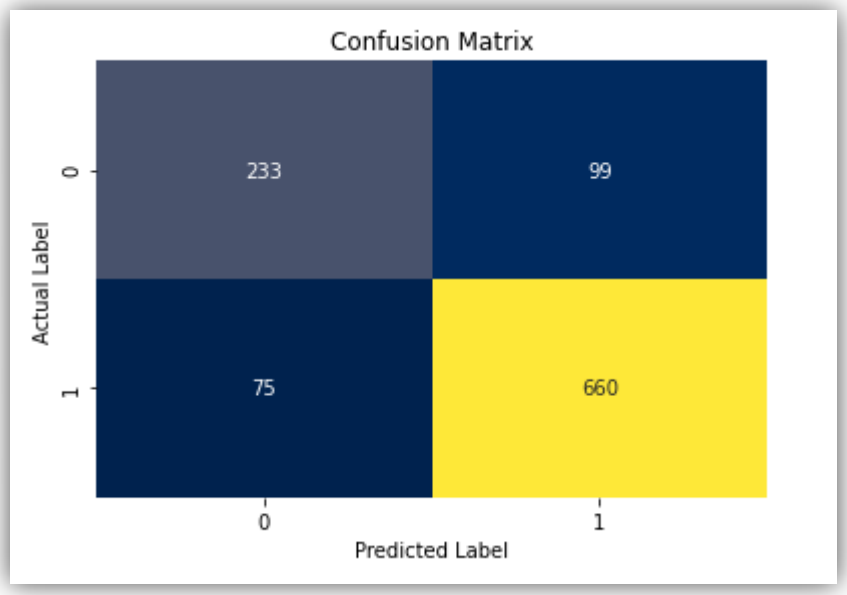


Figure 22 Confusion matrix for train data

CONFUSION MATRIX FOR TEST DATA

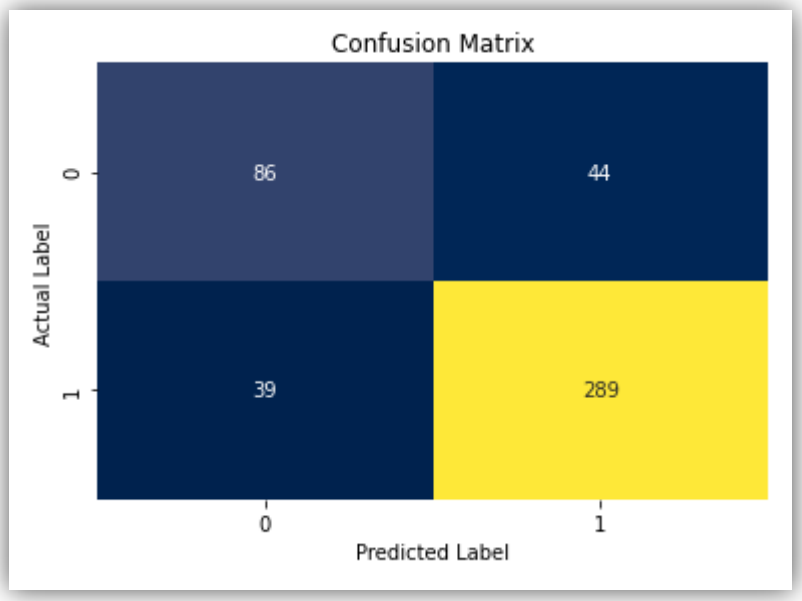


Figure 23 Confusion matrix for test data

Plotting confusion matrix for the different models for the Training and Test Data

Figure 24 confusion matrix for the different models for the Training and Test Data

CLASSIFICATION REPORT FOR LDA

	precision	recall	f1-score	support
0	0.76	0.70	0.73	332
1	0.87	0.90	0.88	735
accuracy			0.84	1067
macro avg	0.81	0.80	0.81	1067
weighted avg	0.83	0.84	0.84	1067

	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

Table 25 Clasification report for LDA

AUC and ROC for the training and test data

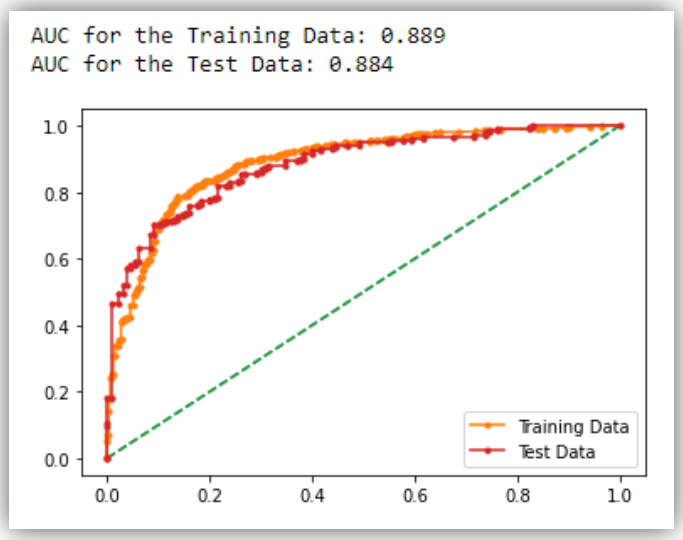


Figure 25 AUC and ROC graph for training and test data

KNN MODEL

knn_train_precision	0.87
knn_train_recall	0.9
knn_train_f1	0.89

knn_test_precision	0.88
knn_test_recall	0.88
knn_test_f1	0.88

Table 26 Inferences on precision, recall and f1 for training and test data

CONFUSION MATRIX

Confusion matrix on the training and test data

Figure 26 Confusion matrix on the training and test data

Inference:

Training data:
- ➢ True Negative : 237 False Positive : 95
- ➢ False Negative : 72 True Positive : 663

Test data:
- ➢ True Negative : 92 False Positive : 38
- ➢ False Negative : 41 True Positive : 287

Classification Report of training and test data

	precision	recall	f1-score	support
0	0.77	0.71	0.74	332
1	0.87	0.90	0.89	735
accuracy			0.84	1067
macro avg	0.82	0.81	0.81	1067
weighted avg	0.84	0.84	0.84	1067

	precision	recall	f1-score	support
0	0.69	0.71	0.70	130
1	0.88	0.88	0.88	328
accuracy			0.83	458
macro avg	0.79	0.79	0.79	458
weighted avg	0.83	0.83	0.83	458

Table 27 Classification Report of training and test data

AUC and ROC for the training and test data

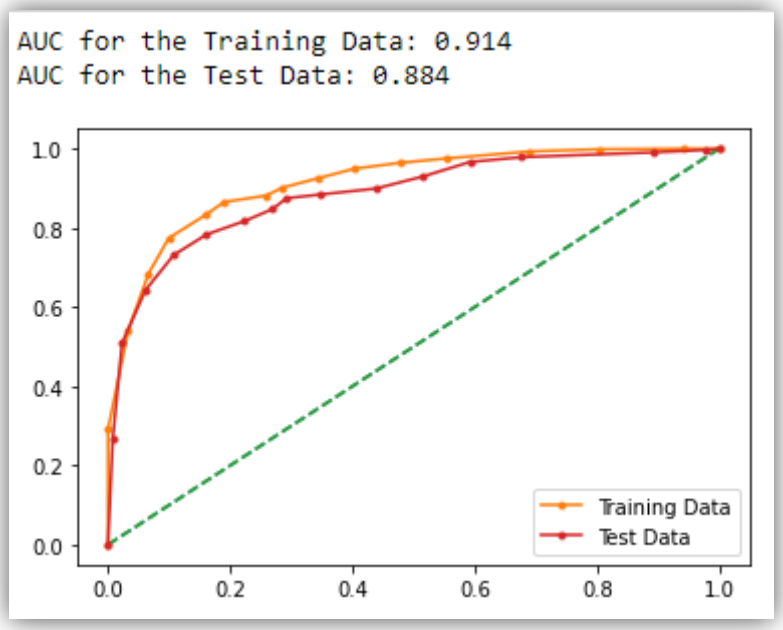


Figure 27 AUC and ROC for the training and test data

Inference

Train Data:

➢ AUC: 91%
➢ Accuracy: 84%
➢ precision : 87%
➢ recall : 90%
➢ f1 :89%

Test Data:

➢ AUC: 88.4%
➢ Accuracy: 83%
➢ precision :88%
➢ recall : 88%
➢ f1 : 88%
➢ Training and Test set results are almost similar,This proves no overfitting or underfitting

Naive Bayes model

nb_train_precision	0.88
nb_train_recall	0.88
nb_train_f1	0.88

nb_test_precision	0.89
nb_test_recall	0.87
nb_test_f1	0.88

Table 28 Inferences on precision, recall and f1 for training and test data

Confusion matrix on the training and test data

Figure 28 Confusion matrix on the training and test data

Inference

Training data:

➢ True Negative : 240 False Positive : 92

➢ False Negative : 86 True Positive : 649

Test data:

➢ True Negative : 94 False Positive : 36

➢ False Negative : 44 True Positive : 284

Classification Report of training and test data

	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

Table 29 Classification Report of training and test data

AUC and ROC for the training and test data

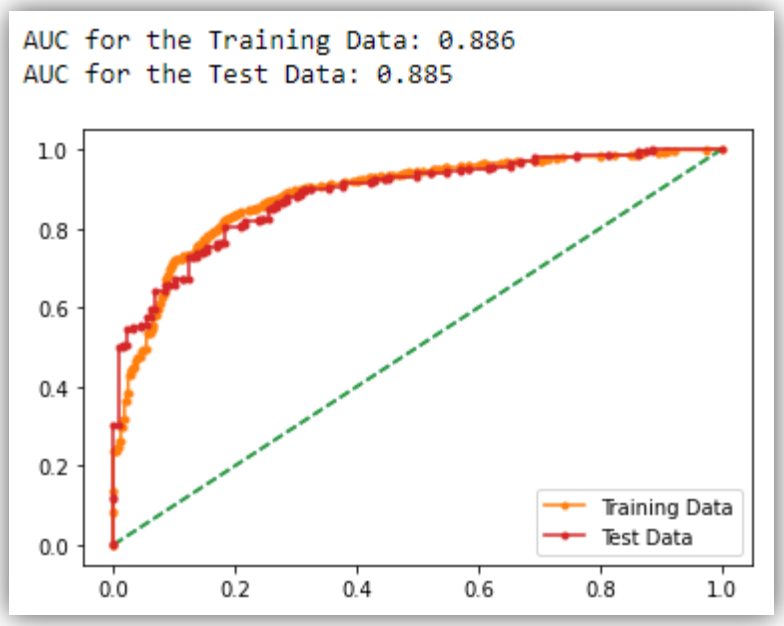


Figure 29 AUC and ROC for the training and test data

Inference

Train Data:

➢ AUC: 88.6%
➢ Accuracy: 083%
➢ precision : 88%
➢ recall : 88%
➢ f1 :88%

Test Data:

➢ AUC: 88.6%
➢ Accuracy: 83%
➢ precision :89%
➢ recall : 87%
➢ f1 : 88%

Bagging with Randomforest

bag_train_precision	0.85
bag_train_recall	0.92
bag_train_f1	0.89

bag_test_precision	0.86
bag_test_recall	0.89
bag_test_f1	0.87

Table 30 Inferences on precision, recall and f1 for training and test data

Confusion matrix on the training and test data

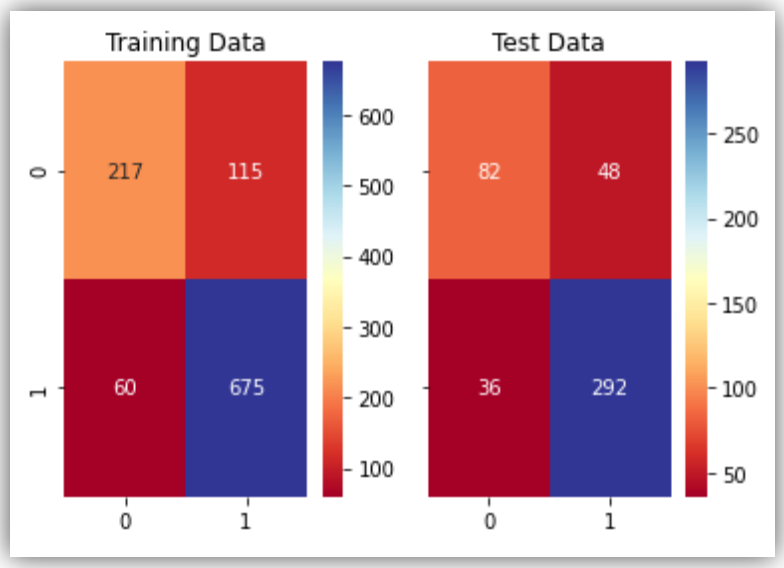


Figure 30 Confusion matrix on the training and test data

Classification Report of training and test data

	precision	recall	f1-score	support
0	0.78	0.65	0.71	332
1	0.85	0.92	0.89	735
accuracy			0.84	1067
macro avg	0.82	0.79	0.80	1067
weighted avg	0.83	0.84	0.83	1067

	precision	recall	f1-score	support
0	0.69	0.63	0.66	130
1	0.86	0.89	0.87	328
accuracy			0.82	458
macro avg	0.78	0.76	0.77	458
weighted avg	0.81	0.82	0.81	458

Table 31 Classification Report of training and test data

AUC and ROC for the training and test data

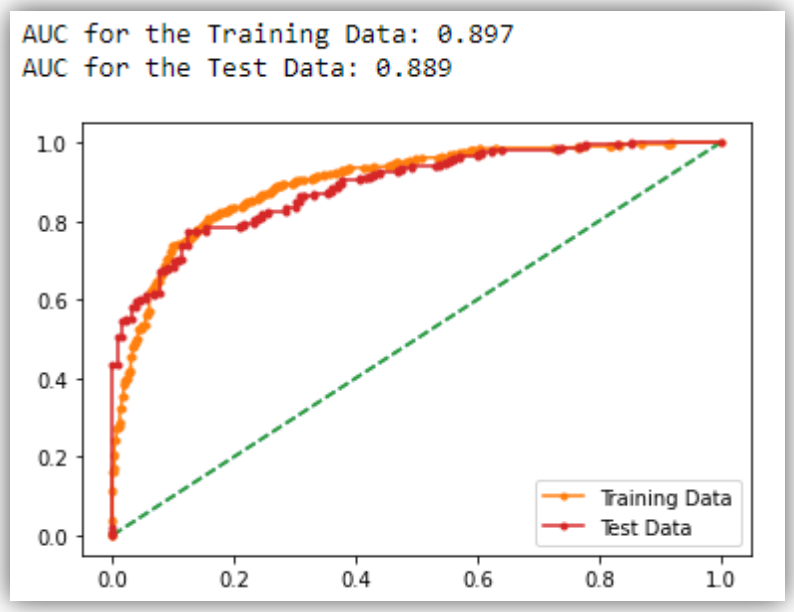


Figure 31 AUC and ROC for the training and test data

AdaBoostClassifier

ADa_train_precision	0.87
ADa_train_recall	0.91
ADa_train_f1	0.89

ADa_test_precision	0.87
ADa_test_recall	0.88
ADatest_f1	0.87

Table 32 Inferences on precision, recall and f1 for training and test data

Confusion matrix on the training and test data

Figure 32 Confusion matrix on the training and test data

Classification Report of training and test data

	precision	recall	f1-score	support
0	0.77	0.70	0.74	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	0.69	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

Table 33 Classification Report of training and test data

AUC and ROC for the training and test data

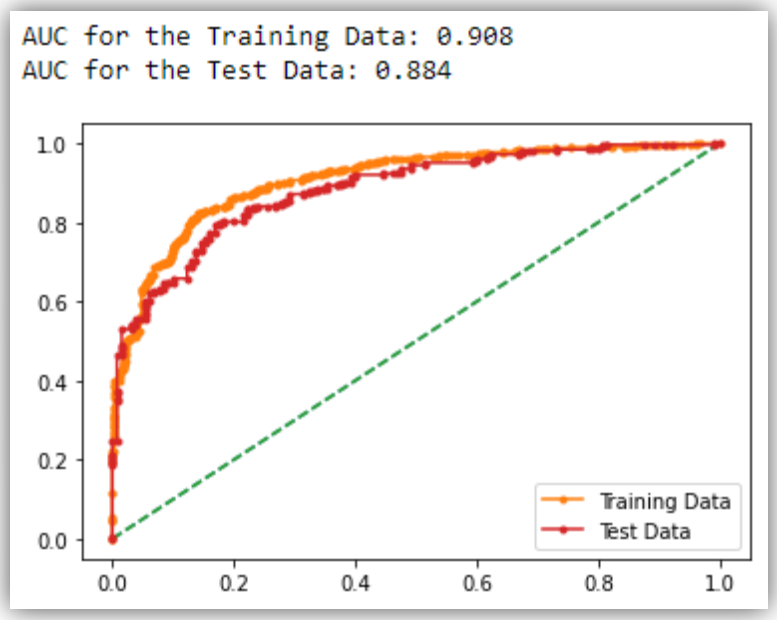


Figure 33 AUC and ROC for the training and test data

Gradient Boosting

```
gbcl_train_precision  0.88
gbcl_train_recall  0.93
gbcl_train_f1  0.9
```

```
gbcl_test_precision  0.9
gbcl_test_recall  0.86
gbcl_test_f1  0.88
```

Table 34 Inferences on precision, recall and f1 for training and test data

Confusion matrix on the training and test data

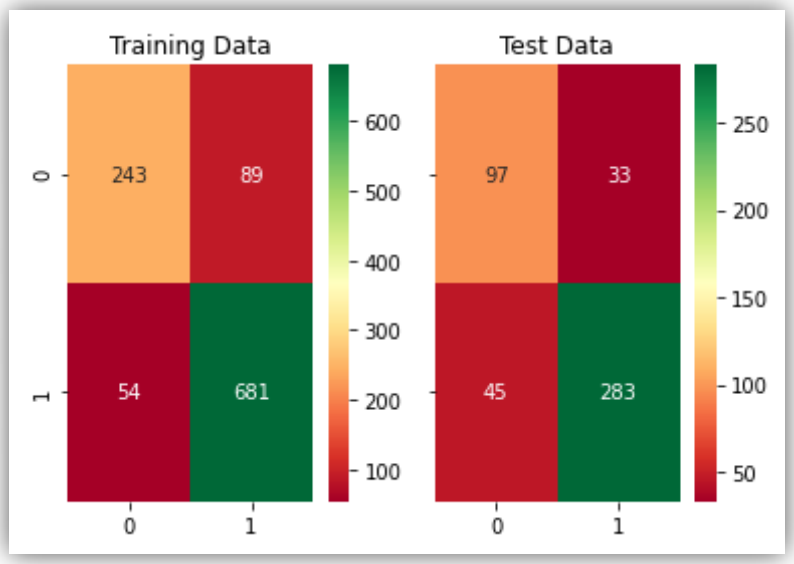


Figure 34 Confusion matrix on the training and test data

Classification Report of training and test data

	precision	recall	f1-score	support
0	0.82	0.73	0.77	332
1	0.88	0.93	0.90	735
accuracy			0.87	1067
macro avg	0.85	0.83	0.84	1067
weighted avg	0.86	0.87	0.86	1067

	precision	recall	f1-score	support
0	0.68	0.75	0.71	130
1	0.90	0.86	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

Table 35 Classification Report of training and test data

AUC and ROC for the training and test data

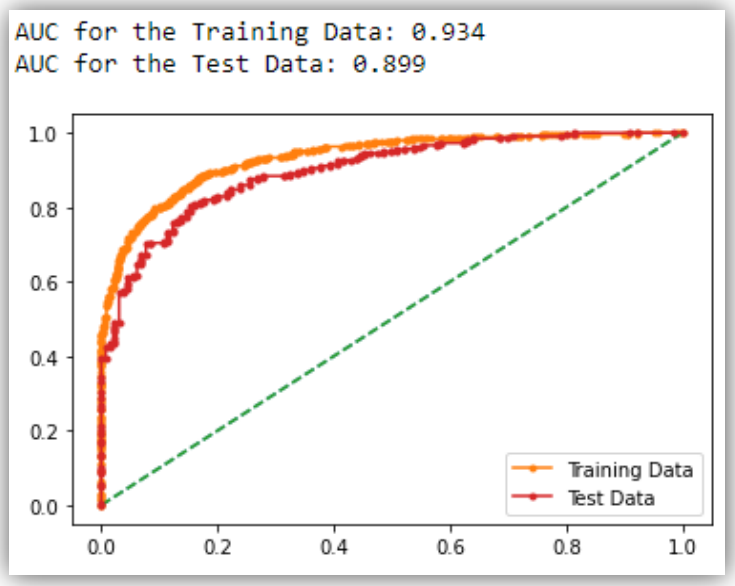


Figure 35 AUC and ROC for the training and test data

Final Model: Comparing all the models

	LR Train	LR Test	LDA Train	LDA Test	KNN Train	KNN Test	NB Train	NB Test	BAGGING Train	BAGGING Test	ADA Train	ADA Test	Gradient Train	Gradient Test
Accuracy	0.84	0.82	0.84	0.82	0.84	0.83	0.83	0.83	0.84	0.82	0.84	0.82	0.87	0.87
AUC	0.89	0.88	0.89	0.88	0.91	0.88	0.89	0.88	0.90	0.89	0.91	0.88	0.93	0.90
Recall	0.91	0.89	0.90	0.88	0.90	0.88	0.88	0.87	0.92	0.89	0.91	0.88	0.93	0.86
Precision	0.86	0.86	0.87	0.87	0.87	0.88	0.88	0.89	0.85	0.86	0.87	0.87	0.88	0.90
F1 Score	0.88	0.88	0.88	0.87	0.89	0.88	0.88	0.88	0.89	0.87	0.89	0.87	0.90	0.88

Table 36 Final Model: Comparing all the models

Inferences

- Almost all the models performed well with accuracy between 82% to 84%. Gradient boosting improved the accuracy to 87% so it is better model for to predict which party a voter will vote.
- Comparing all the model ,Gradient boosting model is best model for this dataset with accuracy of 87% in both training and test set
- AUC of Train and test in Gradient boosting model is 93% and 90% respectively
- f1 score of Train and test in Gradient boosting model is 91% and 88% respectively.
- Precision of Train and test in Gradient boosting model is 88% and 90% respectively.
- Recall of Train and test in Gradient boosting model is 93% and 86% respectively.
- Accuracy , AUC, Precision ,Recall for test data are almost in line with training data in Gradient boosting model. This indicates no over fitting or under fitting in the model.
- Gradient boosting improved the accuracy to 87% so it is better model for to predict which party a voter will vote.

Overall the Optimized Model

For the sample voter details provided, the following conclusions are made by the models:

Model	Prediction
Logistic Regression	conservative party
Linear Discriminant Analaysis	conservative party
K-Nearest Neighbour	conservative party
Naive Bayes	labour party
Bagging(with Random Forest)	conservative party
Adaptive Boosting	conservative party
Gradient Boosting	conservative party

Table 37 Predicted conclusions for the models

Almost all the models performed well with accuracy between 82% to 84% with scaled data. But Gradient boosting is best and optimized model with accuracy of 87% and also best AUC,Precision,f1 score, Recall all in support of predicting the conservative party.

1.8 Based on these predictions, what are the insights?

Conclusion

Almost all the models performed well with accuracy between 82% to 84%. Gradient boosting improved the accuracy to 87% so it is better model for to predict which party a voter will vote. Comparing all the model ,Gradient boosting model is best model for this dataset with accuracy of 87% in both training and test set. AUC of Train and test in Gradient boosting model is 93% and 90% respectively. f1 score of Train and test in Gradient boosting model is 91% and 88% respectively. Precision of Train and test in Gradient boosting model is 88% and 90% respectively.Recall of Train and test in Gradient boosting model is 93% and 86% respectively.Accuracy , AUC, Precision ,Recall for test data are almost in line with training data in Gradient boosting model. This indicates no over fitting or under fitting in the model.Gradient boosting improved the accuracy to 87% so it is better model for to predict which party a voter will vote.

The main business objective of this project is to build a model to predict which party a voter will vote for based on the given information, to create an exit poll that will help in predicting overall win and seats covered by a particular party.

Various model was built on scaled dataset in that it is found that in comparison Accuracy , AUC, Precision ,Recall for test data are almost in line for the Gradient Boosting model gave best/optimized accuracy with 87% to predict which party a voter will vote based on given information and clearly an exit poll can be built that can help in overall win and seats covered by a particular party.

Only 13% of the people were made false predictions that the votes to be in favour were actually predicted against the labour party.

Model	Prediction
Logistic Regression	conservative party
Linear Discriminant Analaysis	conservative party
K-Nearest Neighbour	conservative party
Naive Bayes	labour party
Bagging(with Random Forest)	conservative party
Adaptive Boosting	conservative party
Gradient Boosting	conservative party

Table 38 Predicted conclusions for the models

Almost all the models performed well with accuracy between 82% to 84% with scaled data. But Gradient boosting is best and optimized model with accuracy of 87% and also best AUC,Precision,f1 score, Recall all in support of predicting the conservative party.

Problem2

Text analysis on 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)

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:

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

Loaded the required packages and extract three speeches using the given code nltk : Inaugural:-

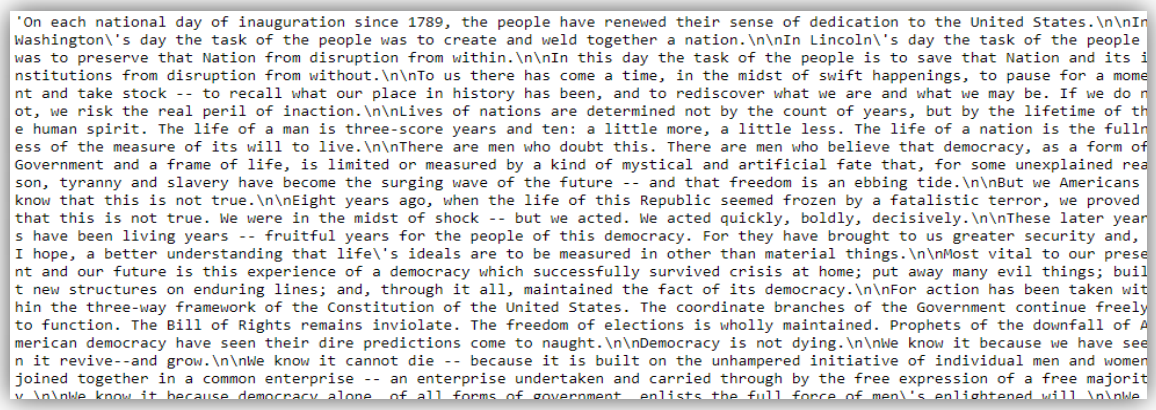
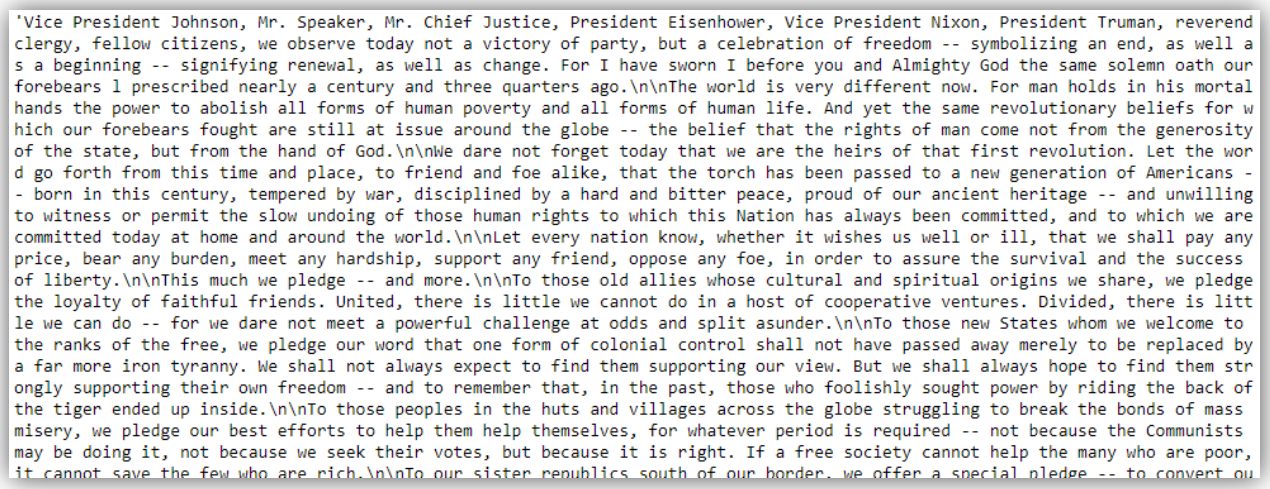


Figure 36 Sample of the speech given by President Franklin D. Roosevelt in 1941

The total number of characters, words and sentences for the mentioned speech given by President Franklin D. Roosevelt in 1941:-

	1941-Roosevelt.txt
Character	7571
Words	1536
Sentences	68

Table 39 Total number of characters, words and sentences of speech given by Franklin D. Roosevelt in 1941

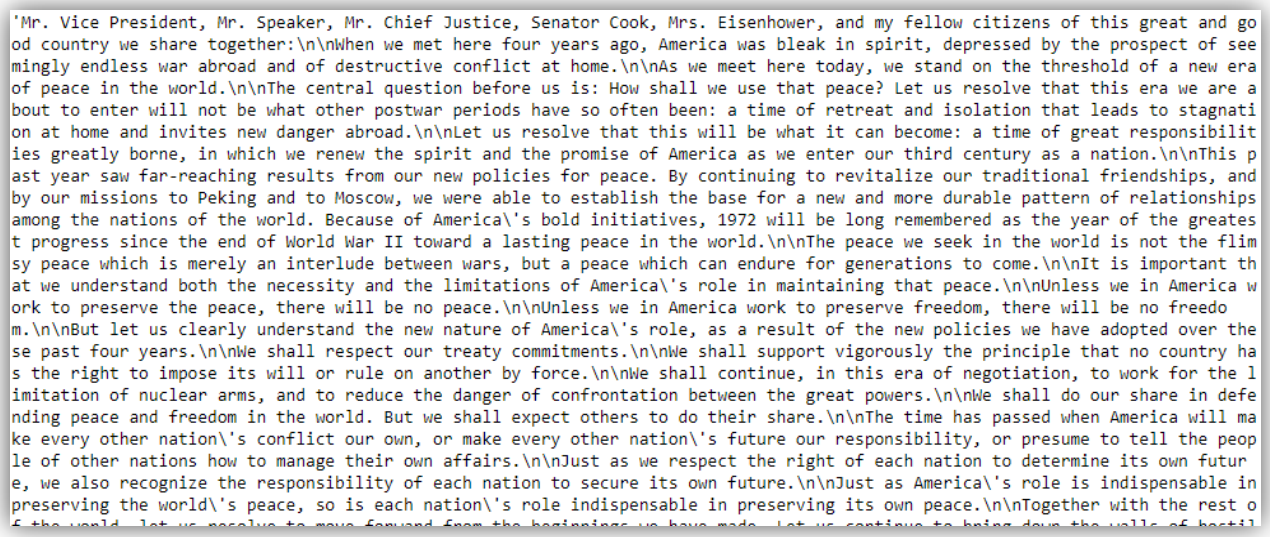


'Vice President Johnson, Mr. Speaker, Mr. Chief Justice, President Eisenhower, Vice President Nixon, President Truman, reverend clergy, fellow citizens, we observe today not a victory of party, but a celebration of freedom -- symbolizing an end, as well a s a beginning -- signifying renewal, as well as change. For I have sworn I before you and Almighty God the same solemn oath our forebears l prescribed nearly a century and three quarters ago.\n\nThe world is very different now. For man holds in his mortal hands the power to abolish all forms of human poverty and all forms of human life. And yet the same revolutionary beliefs for w hich our forebears fought are still at issue around the globe -- the belief that the rights of man come not from the generosity of the state, but from the hand of God.\n\nWe dare not forget today that we are the heirs of that first revolution. Let the wor d go forth from this time and place, to friend and foe alike, that the torch has been passed to a new generation of Americans - - born in this century, tempered by war, disciplined by a hard and bitter peace, proud of our ancient heritage -- and unwilling to witness or permit the slow undoing of those human rights to which this Nation has always been committed, and to which we are committed today at home and around the world.\n\nLet every nation know, whether it wishes us well or ill, that we shall pay any price, bear any burden, meet any hardship, support any friend, oppose any foe, in order to assure the survival and the success of liberty.\n\nThis much we pledge -- and more.\n\nTo those old allies whose cultural and spiritual origins we share, we pledge the loyalty of faithful friends. United, there is little we cannot do in a host of cooperative ventures. Divided, there is litt le we can do -- for we dare not meet a powerful challenge at odds and split asunder.\n\nTo those new States whom we welcome to the ranks of the free, we pledge our word that one form of colonial control shall not have passed away merely to be replaced by a far more iron tyranny. We shall not always expect to find them supporting our view. But we shall always hope to find them str ongly supporting their own freedom -- and to remember that, in the past, those who foolishly sought power by riding the back of the tiger ended up inside.\n\nTo those peoples in the huts and villages across the globe struggling to break the bonds of mass misery, we pledge our best efforts to help them help themselves, for whatever period is required -- not because the Communists may be doing it, not because we seek their votes, but because it is right. If a free society cannot help the many who are poor, it cannot save the few who are rich.\n\nTo our sister republics south of our border, we offer a special pledge -- to convert ou

Figure 37 Sample of the speech given by President John F. Kennedy in 1961

	1961-Kennedy.txt
Character	7618
Words	1546
Sentences	52

Table 40 Total number of characters, words and sentences of the speech by John F. Kennedy in 1961



'Mr. Vice President, Mr. Speaker, Mr. Chief Justice, Senator Cook, Mrs. Eisenhower, and my fellow citizens of this great and good country we share together:\n\nWhen we met here four years ago, America was bleak in spirit, depressed by the prospect of seemingly endless war abroad and of destructive conflict at home.\n\nAs we meet here today, we stand on the threshold of a new era of peace in the world.\n\nThe central question before us is: How shall we use that peace? Let us resolve that this era we are about to enter will not be what other postwar periods have so often been: a time of retreat and isolation that leads to stagnation at home and invites new danger abroad.\n\nLet us resolve that this will be what it can become: a time of great responsibilities greatly borne, in which we renew the spirit and the promise of America as we enter our third century as a nation.\n\nThis past year saw far-reaching results from our new policies for peace. By continuing to revitalize our traditional friendships, and by our missions to Peking and to Moscow, we were able to establish the base for a new and more durable pattern of relationships among the nations of the world. Because of America\'s bold initiatives, 1972 will be long remembered as the year of the greatest progress since the end of World War II toward a lasting peace in the world.\n\nThe peace we seek in the world is not the flimsy peace which is merely an interlude between wars, but a peace which can endure for generations to come.\n\nIt is important that we understand both the necessity and the limitations of America\'s role in maintaining that peace.\n\nUnless we in America work to preserve the peace, there will be no peace.\n\nUnless we in America work to preserve freedom, there will be no freedom.\n\nBut let us clearly understand the new nature of America\'s role, as a result of the new policies we have adopted over these past four years.\n\nWe shall respect our treaty commitments.\n\nWe shall support vigorously the principle that no country has the right to impose its will or rule on another by force.\n\nWe shall continue, in this era of negotiation, to work for the limitation of nuclear arms, and to reduce the danger of confrontation between the great powers.\n\nWe shall do our share in defending peace and freedom in the world. But we shall expect others to do their share.\n\nThe time has passed when America will make every other nation\'s conflict our own, or make every other nation\'s future our responsibility, or presume to tell the people of other nations how to manage their own affairs.\n\nJust as we respect the right of each nation to determine its own future, we also recognize the responsibility of each nation to secure its own future.\n\nJust as America\'s role is indispensable in preserving the world\'s peace, so is each nation\'s role indispensable in preserving its own peace.\n\nTogether with the rest of the world, let us resolve to move forward from the beginnings we have made. Let us continue to bring down the walls of hostil

Figure 38 Sample of the speech given by President Richard Nixon in 1973

	1973-Nixon.txt
Character	9991
Words	2028
Sentences	69

Table 41 Total number of characters, words and sentences of the speech by President Richard Nixon in 1973

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

Stop Words

A stop word is a commonly used word (such as “the”, “a”, “an”, “in”) that a search engine has been programmed to ignore, both when indexing entries for searching and when retrieving them as the result of a search query. We would not want these words to take up space in our database, or taking up valuable processing time. For this, we can remove them easily, by storing a list of words that you consider to stop words.

Most common words in Roosevelt speech after removing stop words

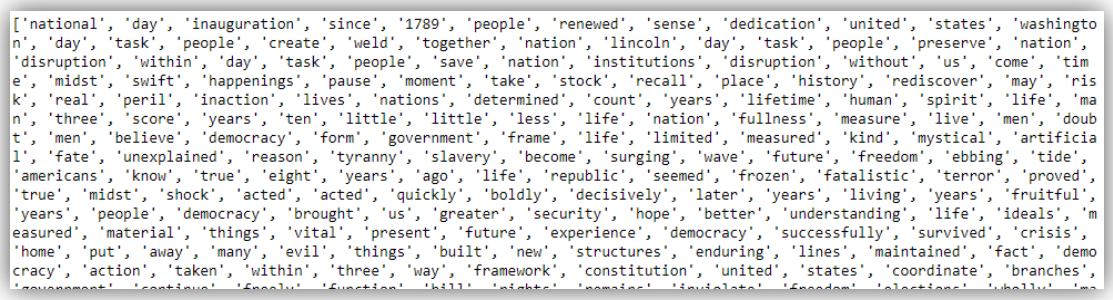


Figure 39 Sample of the most common words in Roosevelt speech after removing stopwords

Most common words in Kennedy speech after removing stop words

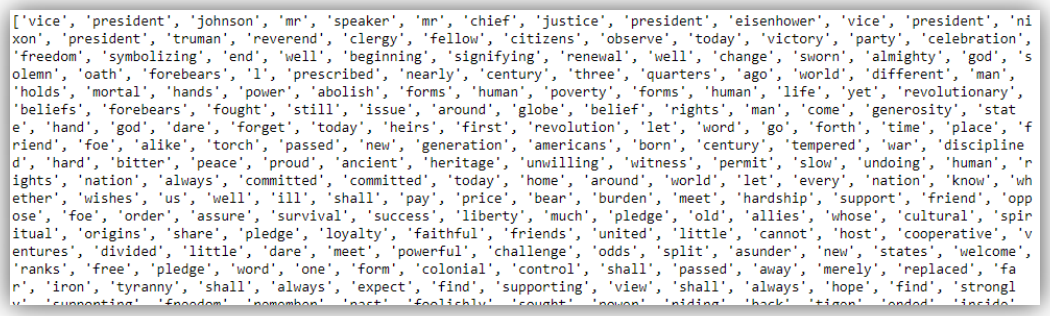
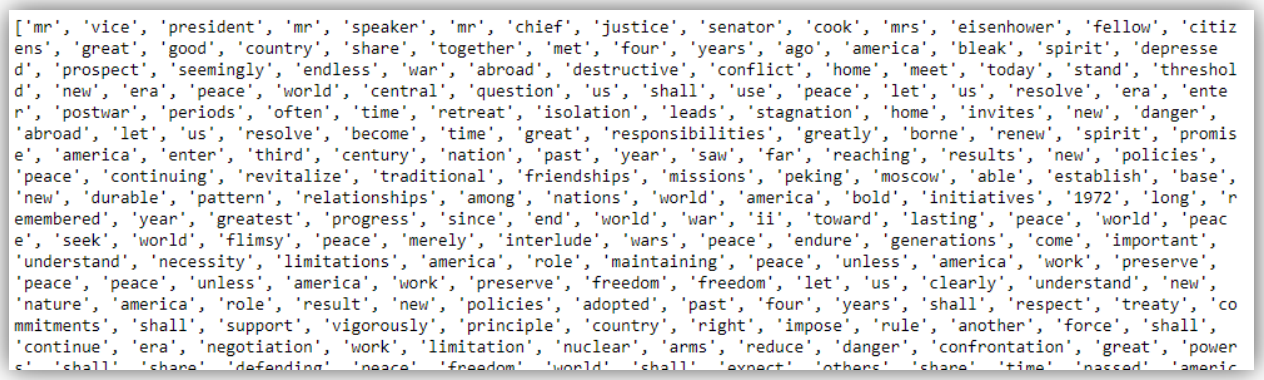


Figure 40 Sample of the Most common words in Kennedy speech after removing stopwords

Most common words in Nixon speech after removing stop words

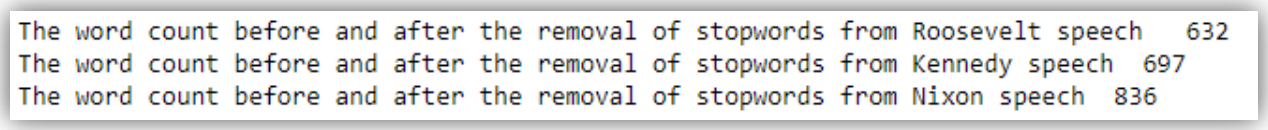


```
['mr', 'vice', 'president', 'mr', 'speaker', 'mr', 'chief', 'justice', 'senator', 'cook', 'mrs', 'eisenhower', 'fellow', 'citiz
ens', 'great', 'good', 'country', 'share', 'together', 'met', 'four', 'years', 'ago', 'america', 'bleak', 'spirit', 'depresse
d', 'prospect', 'seemingly', 'endless', 'war', 'abroad', 'destructive', 'conflict', 'home', 'meet', 'today', 'stand', 'threshol
d', 'new', 'era', 'peace', 'world', 'central', 'question', 'us', 'shall', 'use', 'peace', 'let', 'us', 'resolve', 'era', 'ente
r', 'postwar', 'periods', 'often', 'time', 'retreat', 'isolation', 'leads', 'stagnation', 'home', 'invites', 'new', 'danger',
'abroad', 'let', 'us', 'resolve', 'become', 'time', 'great', 'responsibilities', 'greatly', 'borne', 'renew', 'spirit', 'promis
e', 'america', 'enter', 'third', 'century', 'nation', 'past', 'year', 'saw', 'far', 'reaching', 'results', 'new', 'policies',
'peace', 'continuing', 'revitalize', 'traditional', 'friendships', 'missions', 'peking', 'moscow', 'able', 'establish', 'base',
'new', 'durable', 'pattern', 'relationships', 'among', 'nations', 'world', 'america', 'bold', 'initiatives', '1972', 'long', 'r
emembered', 'year', 'greatest', 'progress', 'since', 'end', 'world', 'war', 'ii', 'toward', 'lasting', 'peace', 'world', 'peac
e', 'seek', 'world', 'flimsy', 'peace', 'merely', 'interlude', 'wars', 'peace', 'endure', 'generations', 'come', 'important',
'understand', 'necessity', 'limitations', 'america', 'role', 'maintaining', 'peace', 'unless', 'america', 'work', 'preserve',
'peace', 'peace', 'unless', 'america', 'work', 'preserve', 'freedom', 'freedom', 'let', 'us', 'clearly', 'understand', 'new',
'nature', 'america', 'role', 'result', 'new', 'policies', 'adopted', 'past', 'four', 'years', 'shall', 'respect', 'treaty', 'co
mmitments', 'shall', 'support', 'vigorously', 'principle', 'country', 'right', 'impose', 'rule', 'another', 'force', 'shall',
'continue', 'era', 'negotiation', 'work', 'limitation', 'nuclear', 'arms', 'reduce', 'danger', 'confrontation', 'great', 'power
s', 'shall', 'share', 'defending', 'peace', 'freedom', 'world', 'shall', 'expect', 'others', 'share', 'time', 'passed', 'americ
```

Figure 41 Sample of the most common words in Nixon speech after removing stopwords

Data cleaning process done on all the three speeches by converting all the characters to lower case and then remove stopwords and special characters /punctuation's and assign it to new variable in the form lists.

The word count before and after the removal of stopwords:-



```
The word count before and after the removal of stopwords from Roosevelt speech   632
The word count before and after the removal of stopwords from Kennedy speech  697
The word count before and after the removal of stopwords from Nixon speech  836
```

Table 42 The word count after the removal of stopwords from the speeches

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)

Most Frequent words in 1941-Roosevelt Speech

	Count
nation	12
know	10
spirit	9
democracy	9
life	9

Table 43 Most Frequent words in 1941-Roosevelt Speech

Most Frequent words in 1961-Kennedy Speech

	Count
let	16
us	12
sides	8
world	8
new	7

Table 44 Most Frequent words in 1961-Kennedy Speech

Most Frequent words in 1973-Nixon Speech

	Count
us	26
let	22
america	21
peace	19
world	18

Table 45 Most Frequent words in 1973-Nixon Speech

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

Figure 42 Word Cloud for 1941-Roosevelt Speech (after cleaning)

Figure 43 Word Cloud for 1961-Kennedy Speech (after cleaning)

Figure 44Word Cloud for 1973-Nixon Speech (after cleaning)

Thank you…