|  |  |
| --- | --- |
|  | Project – Machine Learning |
|  |  |
|  | Varun Kumar  PGP-DSBA  3/19/22 |

Contents

[Problem 1: 5](#_Toc98585346)

[You are hired by one of the leading news channels CNBE who wants to analyse recent elections. This survey was conducted on 1525 voters with 9 variables. You have to build a model, to predict which party a voter will vote for on the basis of the given information, to create an exit poll that will help in predicting overall win and seats covered by a particular party. 5](#_Toc98585347)

[1.1 Read the dataset. Do the descriptive statistics and do the null value condition check. Write an inference on it. 5](#_Toc98585348)

[Data Info: 6](#_Toc98585349)

[Descriptive Statistics for the dataset: 7](#_Toc98585350)

[Skewness: 8](#_Toc98585351)

[1.2 Perform Univariate and Bivariate Analysis. Do exploratory data analysis. Check for Outliers. 8](#_Toc98585352)

[Univariate Analysis: 8](#_Toc98585353)

[Bivariate Analysis: 11](#_Toc98585354)

[Multivariate Analysis: 14](#_Toc98585355)

[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). 16](#_Toc98585356)

[Encoding the dataset: 16](#_Toc98585357)

[Scaling: 17](#_Toc98585358)

[Splitting the data into train and test: 17](#_Toc98585359)

[1.4 Apply Logistic Regression and LDA (linear discriminant analysis). 18](#_Toc98585360)

[Logistic Regression: 18](#_Toc98585361)

[Feature Importance: 19](#_Toc98585362)

[Model Evaluation: 20](#_Toc98585363)

[LDA (linear discriminant analysis): 21](#_Toc98585364)

[Model Evaluation: 22](#_Toc98585365)

[1.5 Apply KNN Model and Naïve Bayes Model. Interpret the results. 23](#_Toc98585366)

[KNN Model: 23](#_Toc98585367)

[Model Evaluation: 23](#_Toc98585368)

[Naive Bayes Model: 25](#_Toc98585369)

[Model Evaluation: 25](#_Toc98585370)

[1.6 Model Tuning, Bagging (Random Forest should be applied for Bagging), and boosting. 26](#_Toc98585371)

[Hyperparameter tuning using GridsearchCV: 26](#_Toc98585372)

[Logistic Regression with GridsearchCV: 26](#_Toc98585373)

[LDA with GridsearchCV: 28](#_Toc98585374)

[KNN Model with GridsearchCV: 28](#_Toc98585375)

[Bagging with Random Forest: 29](#_Toc98585376)

[Boosting: 30](#_Toc98585377)

[AdaBoosting (Adaptive Boosting) 30](#_Toc98585378)

[Gradient Boosting 30](#_Toc98585379)

[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. 31](#_Toc98585380)

[Logistic Regression with GridsearchCV: 31](#_Toc98585381)

[LDA with GridsearchCV: 32](#_Toc98585382)

[KNN Model with GridsearchCV: 34](#_Toc98585383)

[Bagging with Random Forest: 35](#_Toc98585384)

[AdaBoost Classifier: 37](#_Toc98585385)

[Gradient Boost Classifier: 38](#_Toc98585386)

[Comparison between Models: 40](#_Toc98585387)

[1.8 Based on these predictions, what are the insights? 43](#_Toc98585388)

[Problem 2: 44](#_Toc98585389)

[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: 44](#_Toc98585390)

[2.1 Find the number of characters, words, and sentences for the mentioned documents. 44](#_Toc98585391)

[Characters in Roosevelt speech 44](#_Toc98585392)

[Characters in Kennedy speech 44](#_Toc98585393)

[Characters in Nixon speech: 45](#_Toc98585394)

[Words in Roosevelt speech 45](#_Toc98585395)

[Words in Kennedy speech 45](#_Toc98585396)

[Words in Nixon speech 45](#_Toc98585397)

[Sentences in Roosevelt speech 45](#_Toc98585398)

[Sentences in Kennedy speech 45](#_Toc98585399)

[Sentences in Nixon speech 45](#_Toc98585400)

[2.2 Remove all the stopwords from all three speeches. 46](#_Toc98585401)

[Stop Words: 46](#_Toc98585402)

[Word count before removing stopwords 46](#_Toc98585403)

[word count after removing stopwords: 46](#_Toc98585404)

[Sample sentence after removing Stop words: 46](#_Toc98585405)

[2.3 Which word occurs the most number of times in his inaugural address for each president? Mention the top three words. 47](#_Toc98585406)

[Top five words in Roosevelt's speech (after removing the stopwords) 47](#_Toc98585407)

[Top five words in Kennedy's speech (after removing the stopwords) 47](#_Toc98585408)

[Top five words in Nixon's speech (after removing the stopwords) 48](#_Toc98585409)

[2.4 Plot the word cloud of each of the speeches of the variable. 48](#_Toc98585410)

[Roosevelt speech: 48](#_Toc98585411)

[Kennedy speech 49](#_Toc98585412)

[Nixon speech 49](#_Toc98585413)

[Figure 1 Data Set Info 6](#_Toc98585414)

[Figure 2 Data Description 7](#_Toc98585415)

[Figure 3 Economic Condition 8](#_Toc98585416)

[Figure 4 Assessments of Leaders 9](#_Toc98585417)

[Figure 5 Age, Sentiments, Knowledge 9](#_Toc98585418)

[Figure 6 Vote & Gender 10](#_Toc98585419)

[Figure 7 Proportion of votes & gender 10](#_Toc98585420)

[Figure 8 Vote vs Gender 11](#_Toc98585421)

[Figure 9 National Economy vs Vote 11](#_Toc98585422)

[Figure 10 Household Economy vs Vote 12](#_Toc98585423)

[Figure 11 Labour voters vs Economy 12](#_Toc98585424)

[Figure 12 Conservative Voters vs Economy 13](#_Toc98585425)

[Figure 13 Age vs Assessment of Candidates 13](#_Toc98585426)

[Figure 14 PairPlot 14](#_Toc98585427)

[Figure 15 Heatmap 14](#_Toc98585428)

[Figure 16 Boxplot 15](#_Toc98585429)

[Figure 17 Training Dataset Comparison 40](#_Toc98585430)

[Figure 18 Heatmap of Training Dataset Comparison 40](#_Toc98585431)

[Figure 19 ROC-AUC For Training Dataset 41](#_Toc98585432)

[Figure 20 Comparison of Models - Testing Dataset 41](#_Toc98585433)

[Figure 21 Heatmap Model Performance - Test Dataset 42](#_Toc98585434)

[Figure 22 ROC-AUC - Test Dataset 42](#_Toc98585435)

[Figure 23 Cross-Validation 43](#_Toc98585436)

[Figure 24 Heatmap of Cross-Validation 43](#_Toc98585437)

[Figure 25 Sentence Count 45](#_Toc98585438)

[Table 1 Data Set 5](#_Toc98585439)

[Table 2 Data set After Removing 'Unnamed' Column 6](#_Toc98585440)

[Table 3 Skewness 8](#_Toc98585441)

[Table 4 Feature Importance 19](#_Toc98585442)

[Table 5 Model Eval (Logit) 20](#_Toc98585443)

[Table 6 Model Eval (LDA) 22](#_Toc98585444)

[Table 7 Model Eval KNN 24](#_Toc98585445)

[Table 8 Model Eval (Naive Bayes) 25](#_Toc98585446)

[Table 9 Feature Importance (Logit with Gridsearch) 27](#_Toc98585447)

[Table 10 Model Eval (Logit Regression with GridsearchCV) 31](#_Toc98585448)

[Table 11 Model Eval (LDA with GridsearchCV) 32](#_Toc98585449)

[Table 12 Model Eval KNN with GridsearchCV 34](#_Toc98585450)

[Table 13 Model Eval Bagging with RF 36](#_Toc98585451)

[Table 14 Model Eval AdaBoost 37](#_Toc98585452)

[Table 15 Model Eval Gradient Boost 39](#_Toc98585453)

[Table 16 Character and Word Count 45](#_Toc98585454)

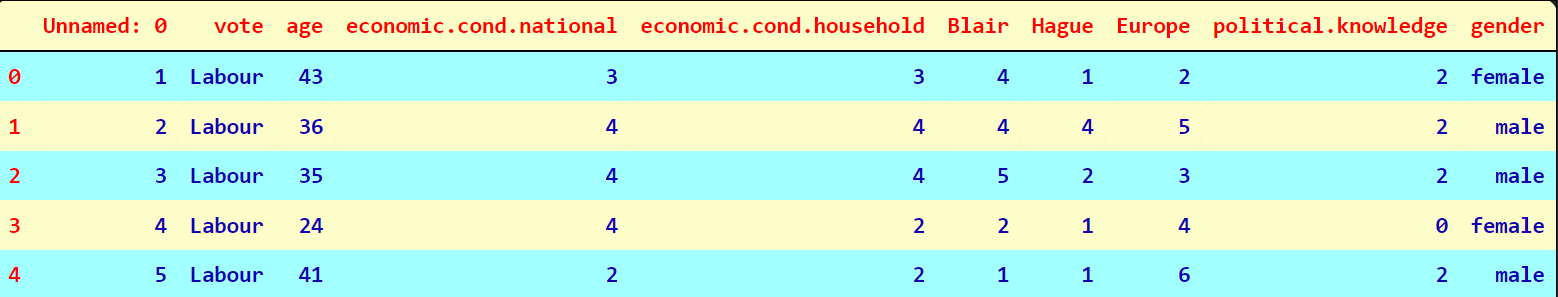
# Problem 1:

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

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

The csv file was imported and converted into a data frame and the first few records are being displayed.

Table 1 Data Set



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.

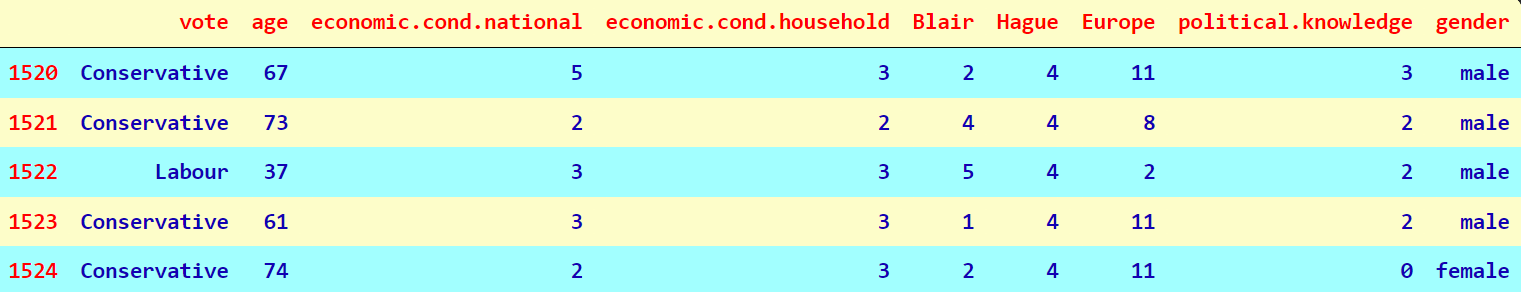
7. Europe: an 11-point scale that measures respondents' attitudes toward European integration. High scores represent ‘Eurosceptic’ sentiment.

8. political.knowledge: Knowledge of parties' positions on European integration, 0 to 3.

9. gender: female or male.

We can see that Unnamed variable is of no use at this moment, let us drop and read the dataset.

Table 2 Data set After Removing 'Unnamed' Column



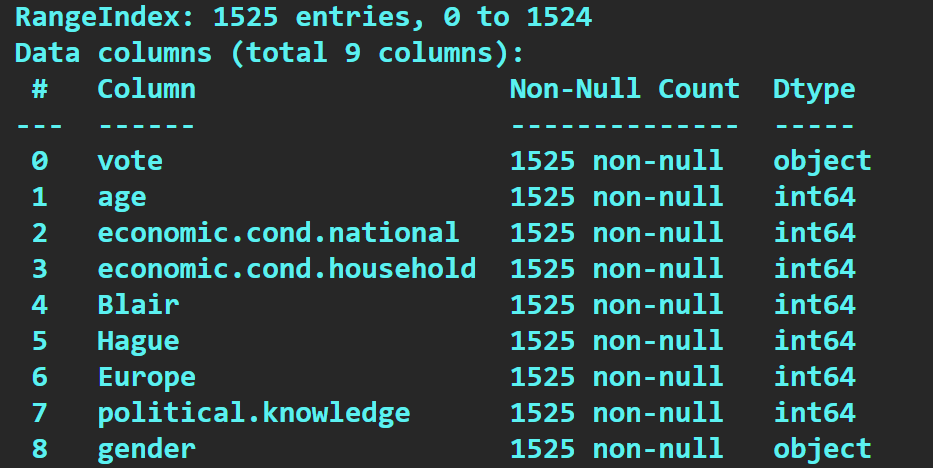
Let us understand the shape of the Dataset.

Number of rows: 1525

Number of columns: 9

### Data Info:

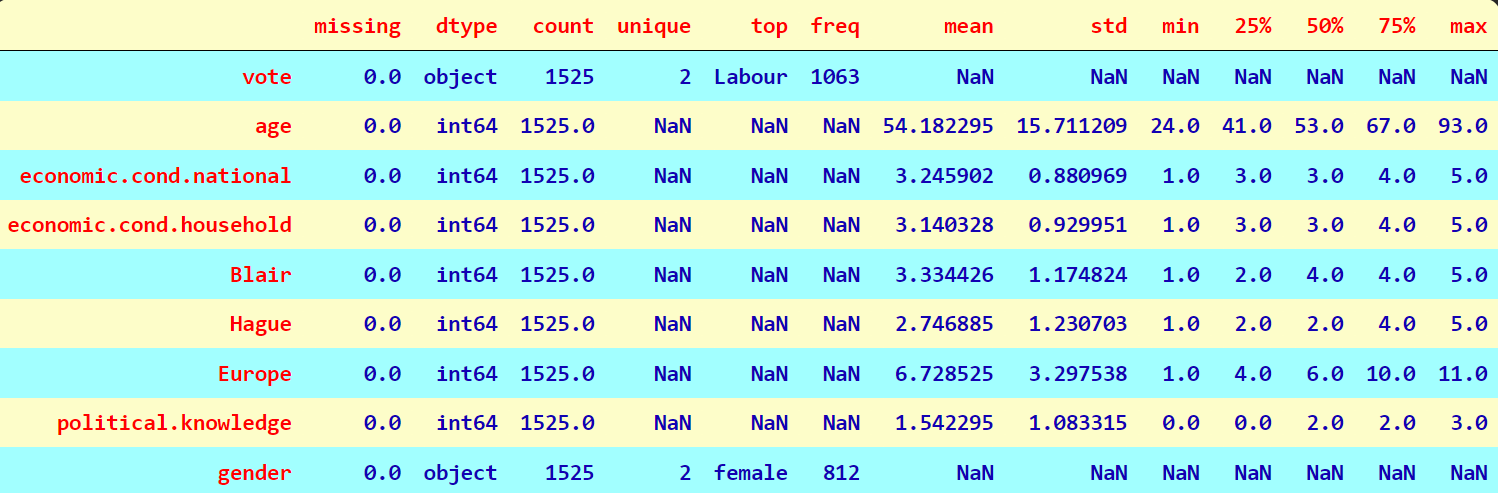
Figure 1 Data Set Info



All the variables except vote and gender are int64 datatypes. But when looking at the values in the dataset for the other variables, they all look like categorical columns except age.

### Descriptive Statistics for the dataset:

Figure 2 Data Description



From the above snippet it is evident that the dataset does not have null values.

From the above snippet we can come to a conclusion that the dataset has only one integer column which is ’age’ The mean and median for the only integer column ‘age’ is almost same indicating the column is normally distributed. ‘vote’ has two unique values Labour and Conservative, which is also a dependent variable. There are 462 individuals who voted for Conservative and 1063 for Labour. ‘gender’ has two unique values male and female. The male and female voters are briefly divided across “Labour” and “Conservative” parties. People prefer Labour party more over the Conservative parity Rest all the columns has object variables with ‘Europe’ being highest having 11 unique values. Since the ‘vote’ variable is the target, we therefore have ‘vote’ as the dependent and rest 8 variables as the independent or predictor variables.

The dataset has few duplicates and removing them is the best choice as duplicates does not add any value. There are 8 duplicate records, we drop those records. After removing duplicates, shape of the dataset is changed.

Number of rows: 1517

Number of columns: 9

### Skewness:

Table 3 Skewness

|  |  |
| --- | --- |
| Feature | Skewness |
| Age | 0.14 |
| Economic.cond.national | -0.24 |
| Economic.cond.household | -0.14 |
| Blair | -0.54 |
| Hague | 0.15 |
| Europe | -0.14 |
| Political.knowledge | -0.42 |

We can see that almost every observation is good to go ahead with the modelling, Except Blair, which has little bit higher side of skewness.

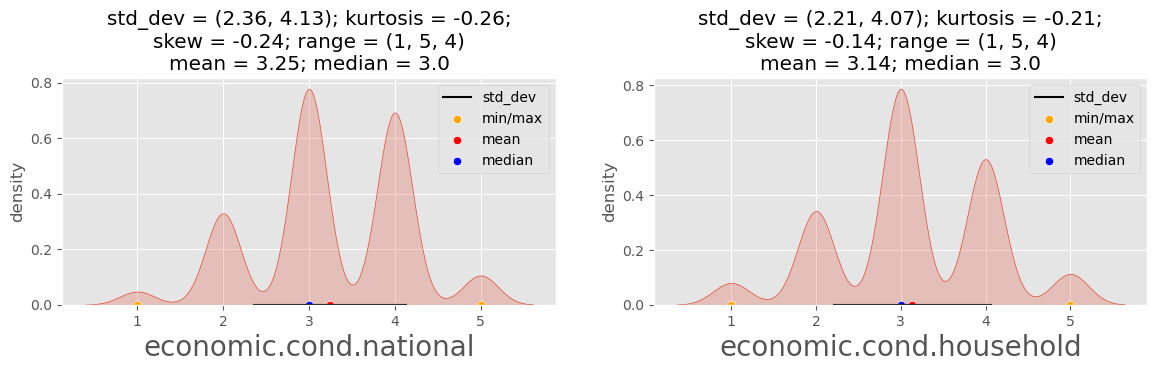
➢ Skewness is a measure of the asymmetry of the probability distribution of a real-valued random variable about its mean.

➢ Age have positive skewness whereas other variables have negative skewness ➢ Blair has more skewness

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

### Univariate Analysis:

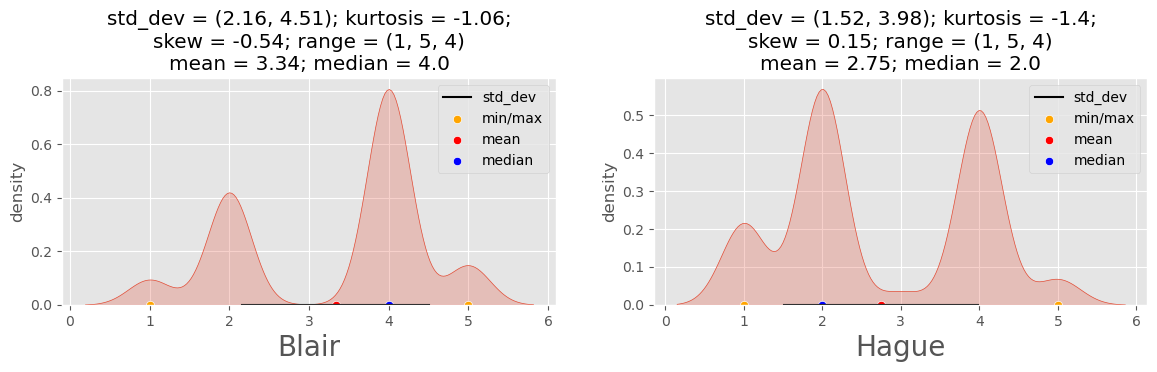
Figure 3 Economic Condition



#### Economic Condition:

As we can see that National & Household economy, both have the highest frequency under the category “3”, with the mean 3.25 and 3.14 respectively.

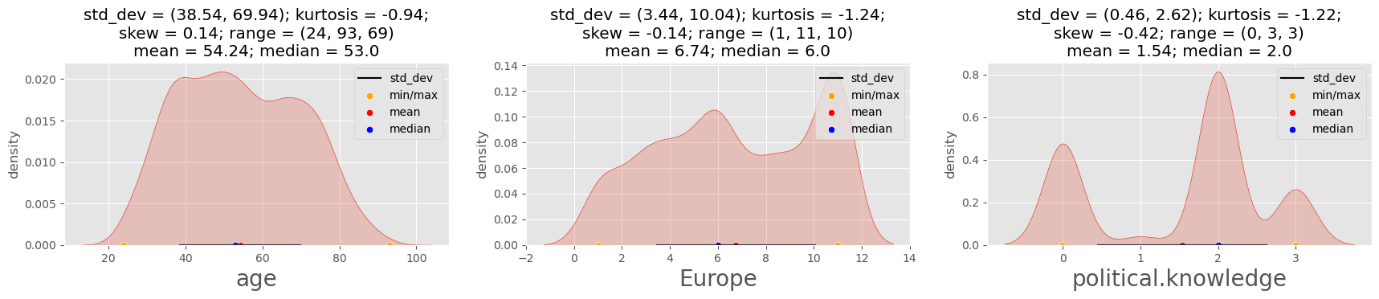
Figure 4 Assessments of Leaders



#### Assessment:

As we can see that leader of Labour Party Blair have maximum density of 4 assessment, on the other hand leader of Conservative Party Hague the maximum density of 2 assessment.

Figure 5 Age, Sentiments, Knowledge



#### Age:

Age of people varies from 24 to 93 with the range of 69. Age of maximum people fall in between 40 to 60 with mean of 54 (approx.).

#### Europe:

We can observe that people high ‘Eurosceptic’ sentiment. Density of ‘Eurosceptic’ sentiment between 10 and 11 is very high.

#### Political Knowledge:

Overall, Political Knowledge of the people is average with maximum density 2.

Figure 6 Vote & Gender

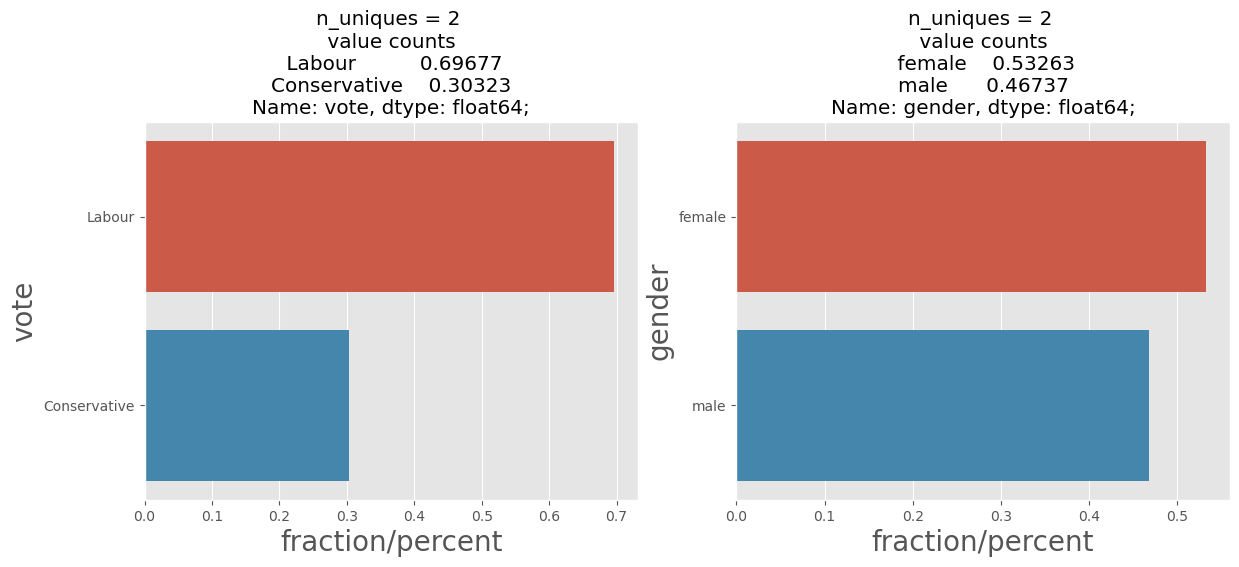
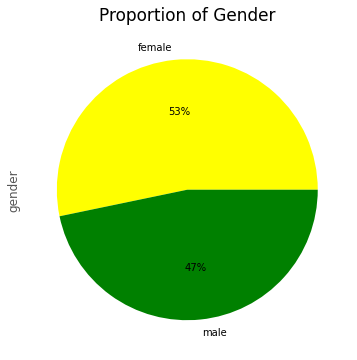
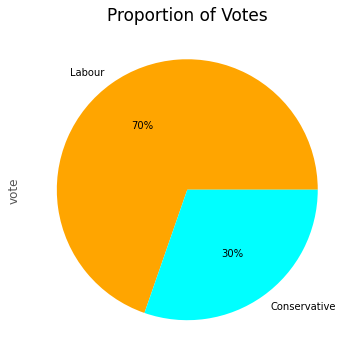


Figure 7 Proportion of votes & gender



#### Vote:

Labour Party leader Blair got high number of votes (70% approx.) as compared to the Conservative Party leader Hague.

#### Gender:

Number of female voters is higher by 6% approx. as compared to the male voters.

#### Inference

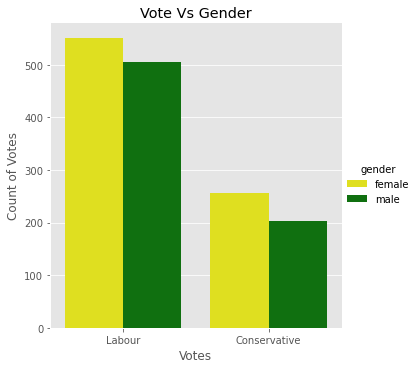
➢ Only age variable is normally distributed and other variables has multimodal skewness seen.

### Bivariate Analysis:

#### Vote vs Gender:

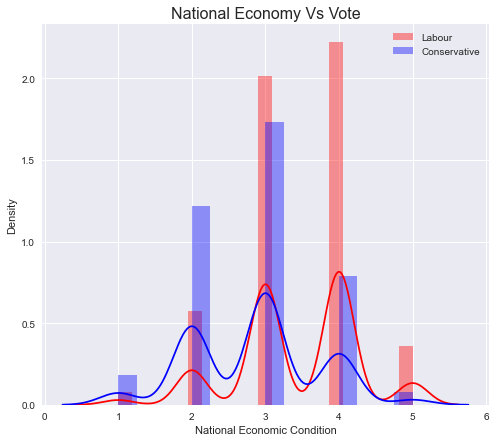
As we can see from the below figure, that, both males and females both voted for Labour Party more.

Figure 8 Vote vs Gender



#### National Economy vs Vote:

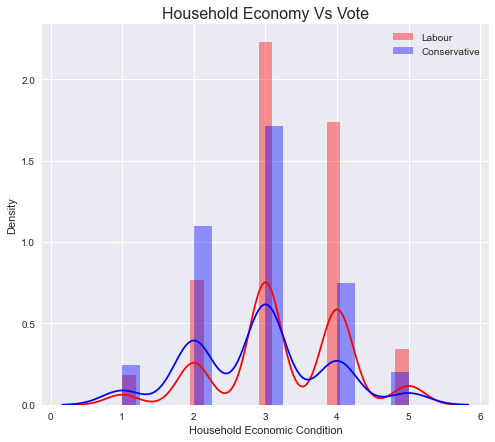
Figure 9 National Economy vs Vote



We can observe that voters with National Economy above 2 is in the favour of Labour Party.

#### Household economy vs Votes:

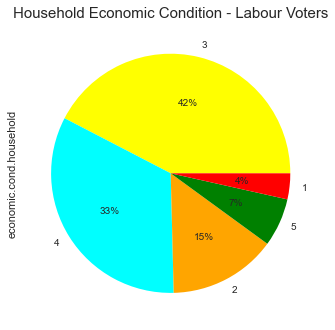
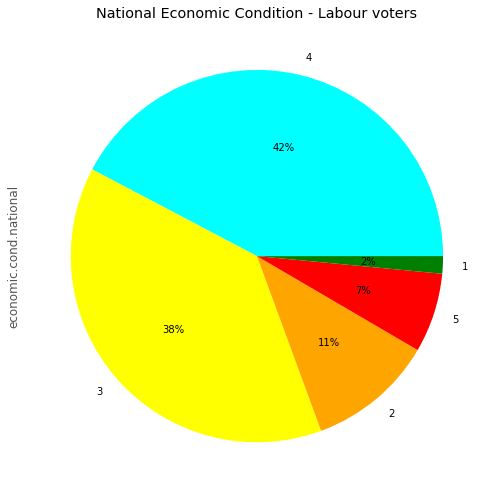
Figure 10 Household Economy vs Vote



We can observe that voters with Household Economy above 2 is in the favour of Labour Party.

#### Labour voters vs Economy:

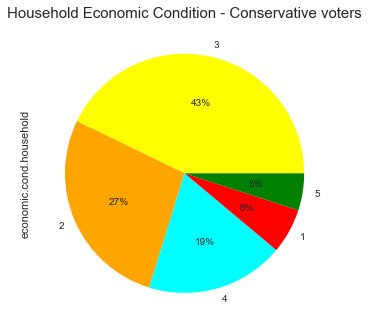
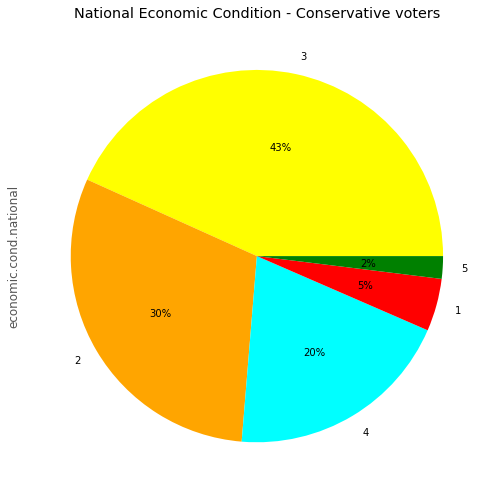
Figure 11 Labour voters vs Economy



National and household economy condition of Labour voters are under category “4”and “3” is maximum.

#### Conservative voters vs Economy:

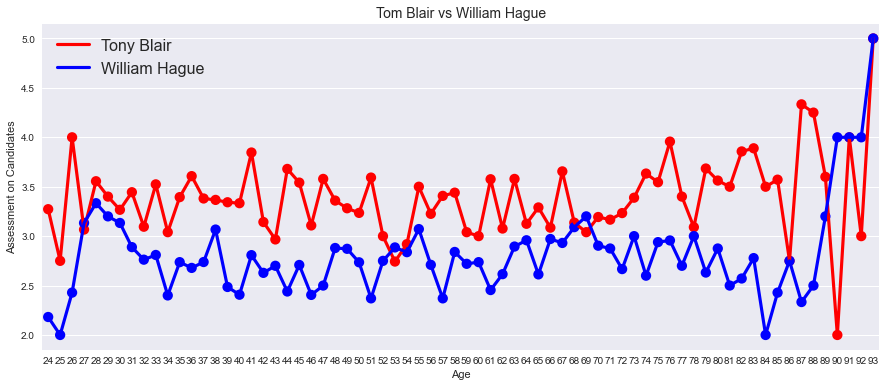
Figure 12 Conservative Voters vs Economy



National and household economy condition of Conservative voters are under category “3” is maximum.

#### Age vs Assessment of Candidates:

Figure 13 Age vs Assessment of Candidates



Popularity of Blair from Labour party is high in every age category as compared to the popularity of Hague from Conservative Party.

### Multivariate Analysis:

Figure 14 PairPlot

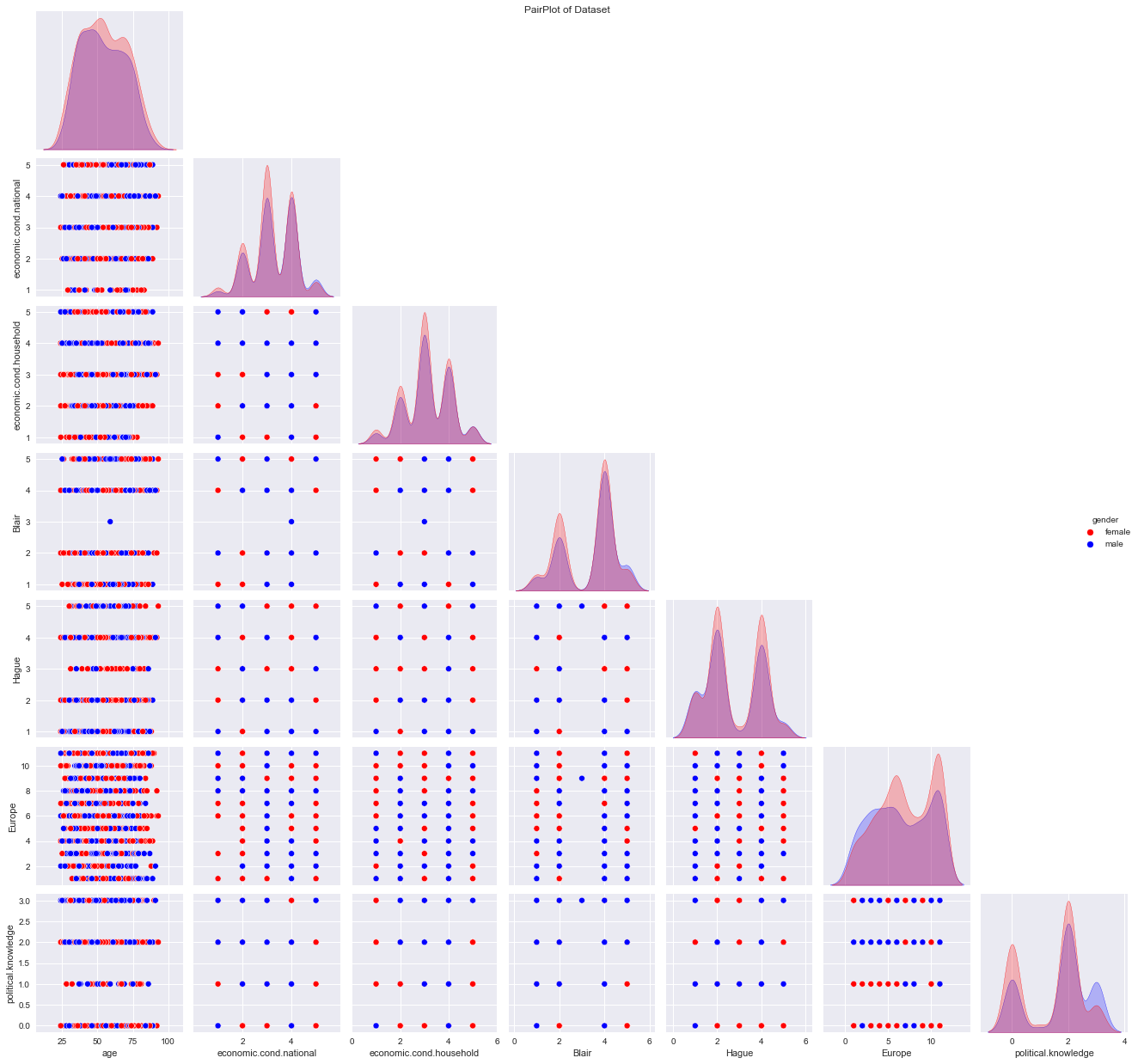


Figure 15 Heatmap

 Inference:

➢ There is no linear relationship between variables

➢ Some of the attributes look like they may have an exponential distribution

➢There is very less correlation between the variables

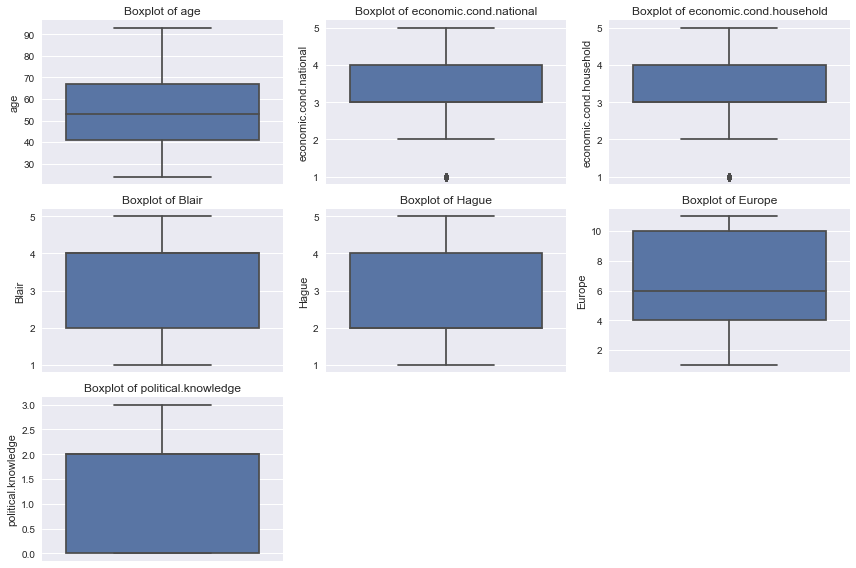
➢ The highest positive correlation is seen between “economic\_cond\_national” and “economic\_cond\_ household” (35%) with nearly similar results seen from “Blair” and “economic\_cond\_national” (0.35)

➢ The highest negative correlation is seen between “Blair” and “Europe” (-0.30) with nearly similar results seen from “Blair” and “Hague” (-0.24)

➢ so, there is less or no chance of multi collinearity

#### Outlier:

Figure 16 Boxplot



There is one outlier present in both, National & Household economic condition. We are not going to remove these outliers as it can be true in the real-world scenario and this will affect our model as well.

Observation:

• 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 their 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).

### Encoding the dataset:

The variables ‘vote’ and ‘gender’ have string values. Converting them into numeric values for modelling. We have encoded the Categorical values (gender & vote) using label encoding.

['female', 'male']

[0,1]

['Labour', 'Conservative']

[1,0]

We can see that vote has two codes 0 represent conservative and 1 represent labour votes. Gender variable is also assigned with respective codes 0 for female and 1 for male.

#### Inference

➢ 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:

➢ Most of the times, your dataset will contain features highly varying in magnitudes, units, and range. But since, most of the machine learning algorithms use Euclidean distance between two data points in their computations, this is a problem.

➢ 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 it’s 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)

### Splitting the data into train and test:

Let us split the dataset into 70% training and 30% test size. Here vote is the target variables and rest are the independent variables.

Dimensions of the training and test data

Number of rows and columns of the training set for the independent variables: (1061, 8)

Number of rows and columns of the training set for the dependent variable: (1061, 1)

Number of rows and columns of the test set for the independent variables: (456, 8)

Number of rows and columns of the test set for the dependent variable: (456, 1)

Total number of Observations are 1517.

#### Inference

splitting 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:

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).

Two libraries of Logistic regression

1. sklearn

2. statsmodel

Here for the model sklearn library is used

#### Parameters:

We used default parameters, which are following:

'C': 1.0,

'class\_weight': None,

'dual': False,

'fit\_intercept': True,

'intercept\_scaling': 1,

'l1\_ratio': None,

'max\_iter': 10000,

'multi\_class': 'auto',

'n\_jobs': 1,

'penalty': 'none',

'random\_state': None,

'solver': 'newton-cg',

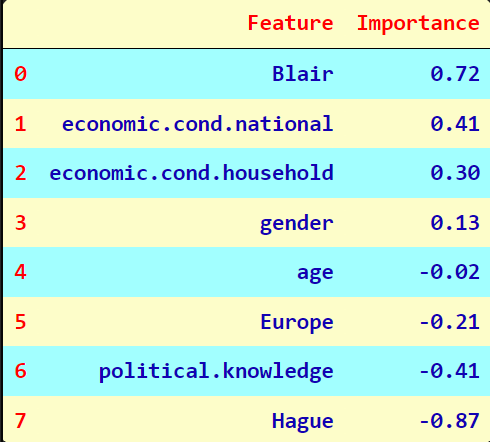
'tol': 0.0001,

'verbose': True,

'warm\_start': False

### Feature Importance:

Table 4 Feature Importance



#### 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 coefficient . 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

➢As we can observe that Blair himself is a very important factor, whereas his opponent Hauge has negative impact. Thus, Blair got very high proportion of votes as well.

### Model Evaluation:

Table 5 Model Eval (Logit)

|  |  |
| --- | --- |
| Training Dataset | Test Dataset |
|  |  |
|  |  |
|  |  |

As we can see from the above table that accuracy for training data is 0.83 and for test data is 0.81. And AUC score for training is 0.90 & for test is 0.87.

Here, F1 & accuracy is the best evaluation matrix, because both the dependent factors is equally important.

The model is not overfitting or underfitting. Training and Testing results shows that the model is excellent with the similar values for both the training & testing.

### LDA (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 modelling differences in groups i.e., separating two or more classes. It is used to project the features in higher dimension space into a lower dimension space.

➢ library used in LDA is sklearn

#### Parameters:

We used default parameters which are following:

'covariance\_estimator': None,

'n\_components': None,

'priors': None,

'shrinkage': None,

'solver': 'svd',

'store\_covariance': False,

'tol': 0.0001}

### Model Evaluation:

Table 6 Model Eval (LDA)

|  |  |
| --- | --- |
| Training Dataset | Test Dataset |
|  |  |
|  |  |
|  |  |

As we can see from the above table that accuracy for training data is 0.83 and for test data is 0.82. And AUC score for training is 0.90 & for test is 0.87.

Here, F1 & accuracy is the best evaluation matrix, because both the dependent factors is equally important.

The model is not overfitting or underfitting. Training and Testing results shows that the model is excellent with the similar values for both the training & testing.

Both logit and LDA perform similar to each other, LDA model is marginally better then logistic.

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

### KNN Model:

First, we scale the dataset using sklearn’s inbuilt function ‘StandardScaler’, because KNN is distance-based model therefore all the values need to be on the same scale. After scaling, mean of the dataset is nearly 0 & standard deviation is 1.

We are going to apply KNN using sklearn library. We fit the training data and predict for the test data.

#### Parameters:

We are using default parameters which are following:

'algorithm': 'auto',

'leaf\_size': 30,

'metric': 'minkowski',

'metric\_params': None,

'n\_jobs': None,

'n\_neighbors': 5,

'p': 2,

'weights': 'uniform'

### Model Evaluation:

Table 7 Model Eval KNN

|  |  |
| --- | --- |
| Training Dataset | Test Dataset |
|  |  |
|  |  |
|  |  |

### 

Training and Testing results shows that the model is excellent with good precision, recall & f1 values. This KNN model have good accuracy and AUC values. Accuracy for training is 0.86 & for test is 0.83, whereas, AUC score for training is 0.93 & for test is 0.84.

The model is not overfitting or underfitting. Training and Testing results shows that the model is excellent with the similar values for both the training & testing.

### Naive Bayes Model:

Naive Bayes is called naive because it assumes that each input variable is independent. This is a strong assumption and unrealistic for real data; however, the technique is very effective on a large range of complex problems.

We are going to apply Naïve Bayes using in built function ‘GaussianNB’ from sklearn library. We fit the training data and predict for the test data.

### Model Evaluation:

Table 8 Model Eval (Naive Bayes)

|  |  |
| --- | --- |
| Training Dataset | Test Dataset |
|  |  |
|  |  |
|  |  |

Training and Testing results shows that the model neither overfitting nor underfitting. The Naive Bayes model also performs well with nearly moderate accuracy and f1 values. Accuracy for training is 0.84 & for test is 0.80, whereas, AUC score for training is 0.89 & for test is 0.86.

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

### Hyperparameter tuning using GridsearchCV:

We pass a list of parameters to the GridsearchCV function, and this will select best parameters automatically.

### Logistic Regression with GridsearchCV:

List of parameters:

class\_weight= {0:2,1:1}

'penalty' ['l2','none','elasticnet'],

'C': log of (-3,3,20),

'solver' ['newton-cg', 'lbfgs', 'sag', 'saga', 'liblinear']

***class\_weight****− dict or ‘balanced’ optional, default = none*

It represents the weights associated with classes. If we use the default option, it means all the classes are supposed to have weight one. On the other hand, if you choose class\_weight: balanced, it will use the values of y to automatically adjust weights.

***penalty****− ‘L1’, ‘L2’, ‘elasticnet’ or none, optional, default = ‘L2’*

This parameter is used to specify the norm L1 (Lasso) or L2 (Ridge) used in penalization (regularization).

***C****− optional, default=1.0*

It represents the inverse of regularization strength, which must always be a positive float.

***solver****− [‘newton-cg’, ‘lbfgs’, ‘liblinear’, ‘saag’, ‘saga’], optional, default = ‘liblinear’*

This parameter represents which algorithm to use in the optimization problem. Followings are the properties of options under this parameter −

* **liblinear** − It is a good choice for small datasets. It also handles L1 penalty. For multiclass problems, it is limited to one-versus-rest schemes.
* **newton-cg** − It handles only L2 penalty.
* **lbfgs** − For multiclass problems, it handles multinomial loss. It also handles only L2 penalty.
* **saga** − It is a good choice for large datasets. For multiclass problems, it also handles multinomial loss. Along with L1 penalty, it also supports ‘elasticnet’ penalty.
* **sag** − It is also used for large datasets. For multiclass problems, it also handles multinomial loss.

After passing above mentioned parameters to GridsearchCV we get the following best parameters for model building.

C=0.69,

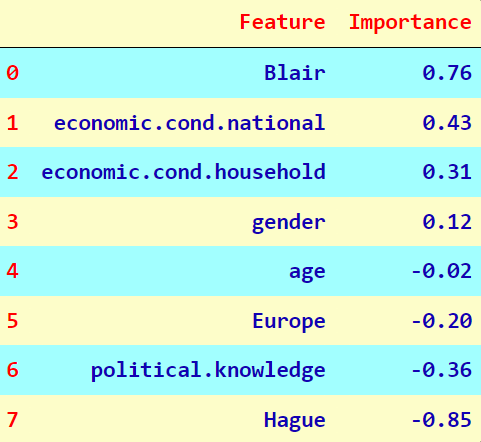
class\_weight= {0: 2, 1: 1},

solver='liblinear',

penalty: 'l2’

#### Feature Importance:

Table 9 Feature Importance (Logit with Gridsearch)



Feature important is more or less same as we get in the basic model.

### LDA with GridsearchCV:

List of parameters:

solver: ['svd', 'lsqr', 'eigen'],

tol: [0.00001,0.0001,0.001,0.01]

**solver: *[‘svd’, ‘lsqr’, ‘eigen’], default=’svd’***

**Solver to use, possible values:**

* ‘svd’: Singular value decomposition (default). Does not compute the covariance matrix, therefore this solver is recommended for data with a large number of 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.

**Tol*: default=1.0e-4***

Absolute threshold for a singular value of X to be considered significant, used to estimate the rank of X. Dimensions whose singular values are non-significant are discarded. Only used if solver is ‘svd’.

After passing above mentioned parameters to GridsearchCV we get the following best parameters for model building.

'solver': 'svd',

'tol': 1e-05

### KNN Model with GridsearchCV:

List of Parameters:

n\_neighbors: range (5,20),

weights: ['uniform', 'distance'],

metric: ['minkowski', 'euclidean', 'canberra', 'manhattan']

**n\_neighbors*, default=5***

Number of neighbours to use by default for kneighbours queries.

**Weights *[‘uniform’, ‘distance’], default=’uniform’***

Weight function used in prediction. Possible values:

* ‘uniform’: uniform weights. All points in each neighbourhood are weighted equally.
* ‘distance’: weight points by the inverse of their distance. in this case, closer neighbours of a query point will have a greater influence than neighbours which are further away.

**metric*, default=’minkowski’***

* The distance metric to use for the tree. The default metric is minkowski, and with p=2 is equivalent to the standard Euclidean metric.

After passing above mentioned parameters to GridsearchCV we get the following best parameters for model building.

metric: 'canberra',

n\_neighbors: 5,

weights: 'uniform'

### Bagging with Random Forest:

Bagging (Bootstrap Aggregation)

• Reduced chances of over fitting by training each model only with a randomly chosen subset of the training data. Training can be done in parallel.

• Essentially trains a large number of “strong” learners in parallel (each model is an over fit for that subset of the data)

• Combines (averaging or voting) these learners together to "smooth out" predictions.

For Bagging, lets us import RandomForestClassifier & BaggingClassifier from sklearn library. First, create an object for random forest with required parameters followed by an object of bagging. Apply all the parameters and fit the model. Parameters are following:

class\_weight= [0: 4,1: 1.5],

min\_samples\_leaf=2,

min\_samples\_split=4,

n\_estimators=50,

random\_state=1234

### Boosting:

• Trains a large number of "weak" learners in sequence. A weak learner is a simple model that is only slightly better than random (e.g. One depth decision tree).

• Miss-classified data weights are increased for training the next model. So, training has to be done in sequence.

• Boosting then combines all the weak learners into a single strong learner.

### AdaBoosting (Adaptive Boosting)

• In AdaBoost, the successive learners are created with a focus on the ill fitted data of the previous learner

• Each successive learner focuses more and more on the harder to fit data i.e., their residuals in the previous tree

For AdaBoosting, we use sklearn library and import AdaBoostClassifier. Then we create an object with few parameters and fit the training data.

### Gradient Boosting

• Each learner is fit on a modified version of original data (original data is replaced with the x values and residuals from previous learner

• By fitting new models to the residuals, the overall learner gradually improves in areas where residuals are initially high

For Gradient Boosting, we use sklearn library and import GradientBoostingClassifier. Then we create an object with parameters and fit the training data.

## 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 with GridsearchCV:

#### Model Evaluation:

Table 10 Model Eval (Logit Regression with GridsearchCV)

|  |  |
| --- | --- |
| Training Dataset | Testing Dataset |
|  |  |
|  |  |
|  |  |
|  | |

As we can see from the above table that accuracy for training data is 0.83 and for test data is 0.81. And AUC score for training is 0.90 & for test is 0.86. These values are nearly similar to the base model which we create previously.

Here, F1 & accuracy is the best evaluation matrix, because both the dependent factors is equally important.

The model is not overfitting or underfitting. Training and Testing results shows that the model is excellent with the similar values for both the training & testing.

### LDA with GridsearchCV:

#### Model Evaluation:

Table 11 Model Eval (LDA with GridsearchCV)

|  |  |
| --- | --- |
| Training Dataset | Testing Dataset |
|  |  |
|  |  |
|  |  |
|  | |

As we can see from the above table that accuracy for training data is 0.83 and for test data is 0.82. And AUC score for training is 0.90 & for test is 0.87.

Here, F1 & accuracy is the best evaluation matrix, because both the dependent factors is equally important.

The model is not overfitting or underfitting. Training and Testing results shows that the model is excellent with the similar values for both the training & testing.

Both logit and LDA perform similar to each other, LDA model is marginally better then logistic.

### KNN Model with GridsearchCV:

#### Model Evaluation:

Table 12 Model Eval KNN with GridsearchCV

|  |  |
| --- | --- |
| Training Dataset | Test Dataset |
|  |  |
|  |  |
|  |  |
|  | |

Training and Testing results shows that the model is excellent with good precision, recall & f1 values. This KNN model have good accuracy and AUC values. Accuracy for training is 0.87 & for test is 0.81, whereas, AUC score for training is 0.94 & for test is 0.84.

The model is moderate fit. Training and Testing results shows that the model is moderate fit with the moderate difference between values for both the training & testing.

### Bagging with Random Forest:

#### Model Evaluation:

Table 13 Model Eval Bagging with RF

|  |  |
| --- | --- |
| Training Dataset | Test Dataset |
|  |  |
|  |  |
|  |  |
|  | |

Training and Testing results shows that the model is just okay with difference between the values of precision, recall & f1 values. Accuracy for training is 0.91 & for test is 0.82, whereas, AUC score for training is 0.97 & for test is 0.87.

The model is at edge of overfitting. Training and Testing results shows that the model is moderate fit with the moderate difference between values for both the training & testing.

### AdaBoost Classifier:

#### Model Evaluation:

Table 14 Model Eval AdaBoost

|  |  |
| --- | --- |
| Training Dataset | Test Dataset |
|  |  |
|  |  |
|  |  |
|  | |

Training and Testing results shows that the model is just okay with difference between the values of precision, recall & f1 values. Accuracy for training is 0.85 & for test is 0.80, whereas, AUC score for training is 0.92 & for test is 0.84.

The model is moderate fit. Training and Testing results shows that the model is moderate fit with the moderate difference between values for both the training & testing.

### Gradient Boost Classifier:

#### Model Evaluation:

Table 15 Model Eval Gradient Boost

|  |  |
| --- | --- |
| Training Dataset | Test Dataset |
|  |  |
|  |  |
|  |  |
|  | |

Training and Testing results shows that the model is poor with a good difference between the values of precision, recall & f1 values. Accuracy for training is 1.0 & for test is 0.79, whereas, AUC score for training is 1.0 & for test is 0.85.

The model is overfit. Training and Testing results shows that the model is overfit with a good difference between values for both the training & testing.

### Comparison between Models:

#### Training Dataset:

Figure 17 Training Dataset Comparison



Figure 18 Heatmap of Training Dataset Comparison

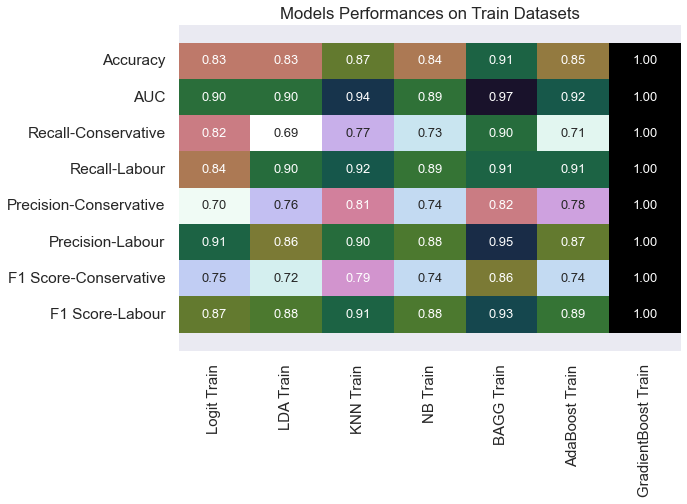
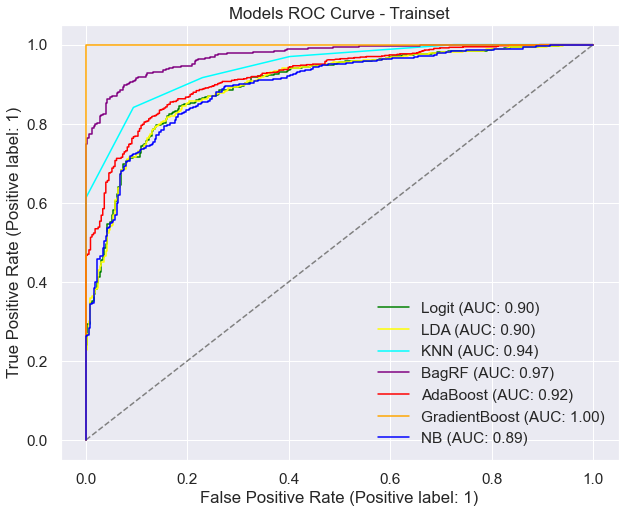


Figure 19 ROC-AUC For Training Dataset



#### Testing Dataset:

Figure 20 Comparison of Models - Testing Dataset

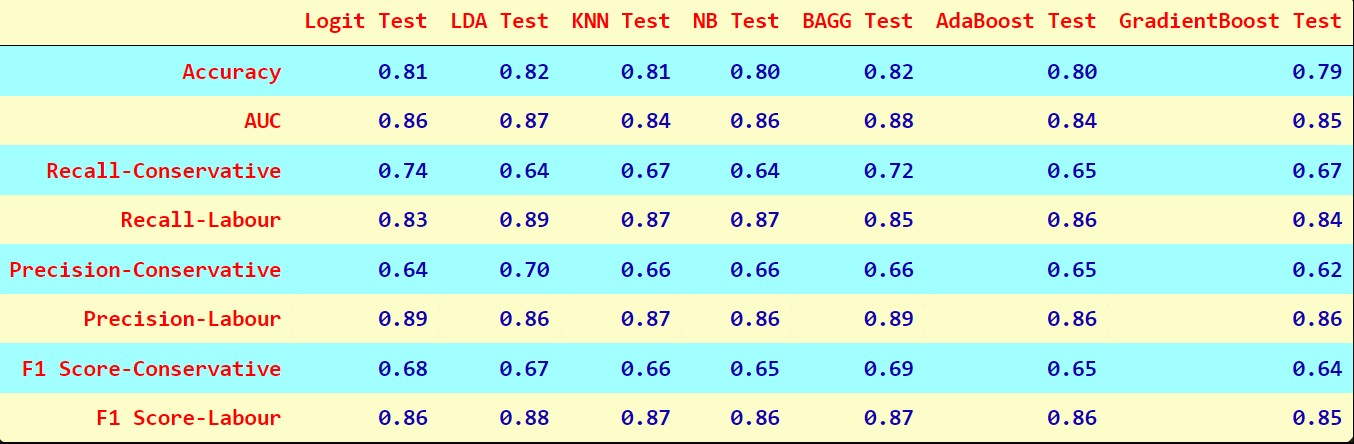


Figure 21 Heatmap Model Performance - Test Dataset

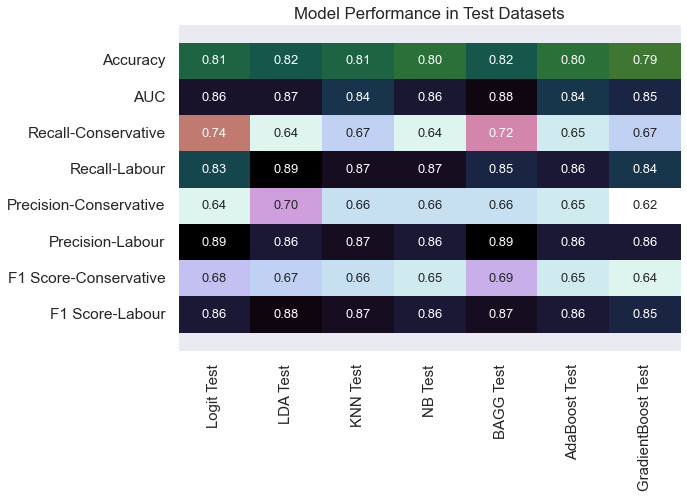
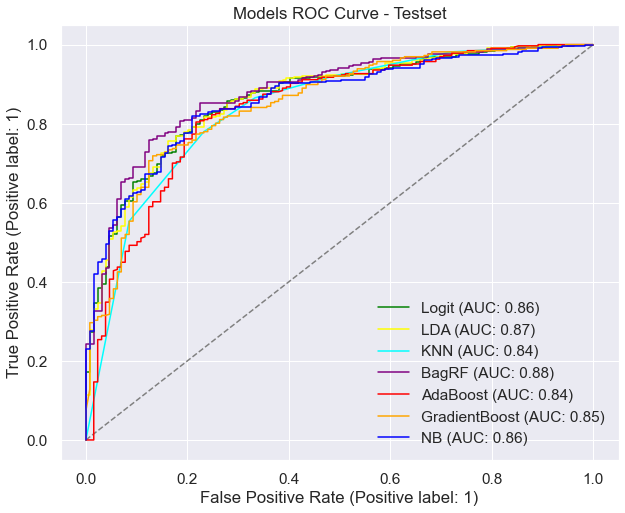


Figure 22 ROC-AUC - Test Dataset



#### Cross-Validation:

Figure 23 Cross-Validation

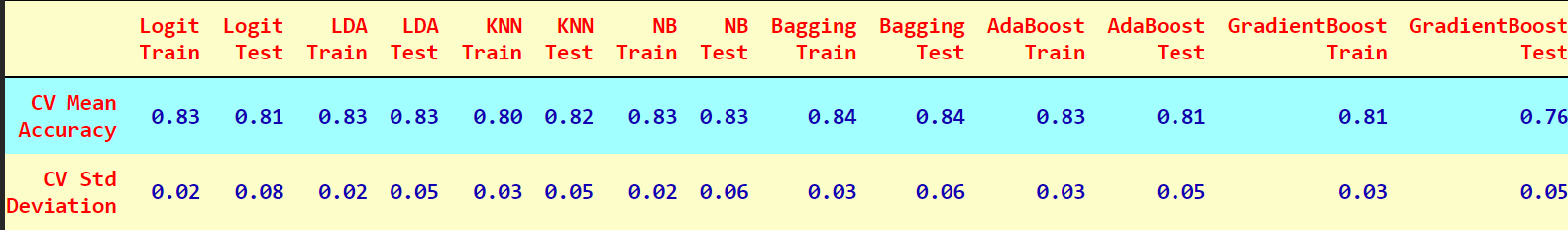


Figure 24 Heatmap of Cross-Validation



We built 7 models for this particular problem. Those are following:

Overfit Models: Gradient Boosting Classifier is an overfit model. When we run cross-validation it improves little bit on unseen data and can attain moderate level.

Moderate Models: Adaboost Classifier, Bagging with random forest, KNN and Naïve Baye’s model are moderate fit. These models can improve but with a good amount of variation as shown during cross-validation.

Good Models: Logit regression and LDA are good models, cross validation also shows that.

But we will choose LDA over logit regression because Logit regression shows more variation then LDA during cross validation. Hence, considering all factors and cross validation we will go with LDA.

## 1.8 Based on these predictions, what are the insights?

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.

1. Comparing all the performance measure, LDA model is performing best. Although there are some more models such as Logit Regression, KNN which is performing almost same as that LDA. But LDA model is very consistent when train and test results are compared with each other. Along with other parameters such as recall and precision values, AUC score, ROC-AUC curve, those results are pretty good in this model.
2. Labour party is performing better then conservative from huge margin, people like Labour leader Blair way more then the Conservative party leader Hague. Conservative party need to improve the image of their leader.
3. Female voters turn out greater than male voters, female voters can be the key to the elections. Parties need to focus on their issues as well.
4. Those who have better national economic conditions are preferring to vote for Labour party.
5. Those who have very high political knowledge have voted for Conservative party. But the number of those persons is low, therefore it doesn’t make any impact on the result of election.
6. Looking at the assessment for both the leaders, Labour leaders are performing well as he has got better ratings in assessment.

# Problem 2:

# In this particular project, we are going to work on the inaugural corpora from the nltk in Python. We will be looking at the following speeches of the Presidents of the United States of America:

1. President Franklin D. Roosevelt in 1941
2. President John F. Kennedy in 1961
3. President Richard Nixon in 1973

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

### Characters in Roosevelt speech

The number of characters in Roosevelt speech are: 7571

### Characters in Kennedy speech

The number of characters in Kennedy speech are: 7618

### Characters in Nixon speech:

The number of characters in Nixon speech are: 9991

Table 16 Character and Word Count

|  |  |
| --- | --- |
| Character Count | Word Count |
|  | |

### Words in Roosevelt speech

The number of Words in Roosevelt speech are: 1323

### Words in Kennedy speech

The number of Words in Kennedy speech are: 1364

### Words in Nixon speech

The number of Words in Nixon speech are: 1769

### Sentences in Roosevelt speech

The number of sentences in Roosevelt speech are: 68

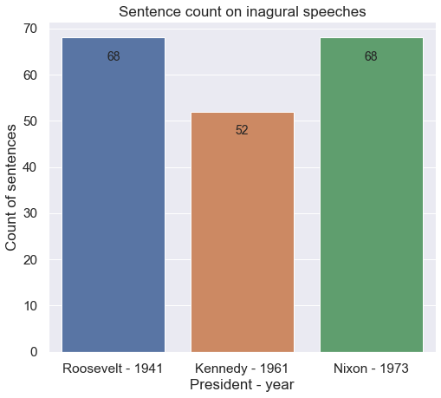
### Sentences in Kennedy speech

The number of sentences in Kennedy speech are: 52

### Sentences in Nixon speech

The number of sentences in Nixon speech are: 68

Figure 25 Sentence Count



## 2.2 Remove all the stopwords from all three speeches.

In natural language processing, useless words (data), are referred to as stop words.

### 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.

### Word count before removing stopwords

|  |  |
| --- | --- |
| 1941- Roosevelt | 1323 |
| 1961- Kennedy | 1364 |
| 1973- Nixon | 1769 |

### word count after removing stopwords:

|  |  |
| --- | --- |
| 1941- Roosevelt | 617 |
| 1961- Kennedy | 658 |
| 1973- Nixon | 775 |

### Sample sentence after removing Stop words:

#### President Roosevelt-

‘national day inauguration since 1789 people renewed sense dedication united states washingtons day task people create weld together nation lincolns day task people preserve nation disruption within day task people save nation institutions disruption without come time midst swift happenings pause moment take stock recall place history’

#### President Kennedy-

‘vice president johnson speaker chief justice president eisenhower vice president nixon president truman reverend clergy fellow citizens observe today victory party celebration freedom symbolizing end well beginning signifying renewal well change sworn almighty god solemn oath forebears l prescribed nearly century three quarters ago’

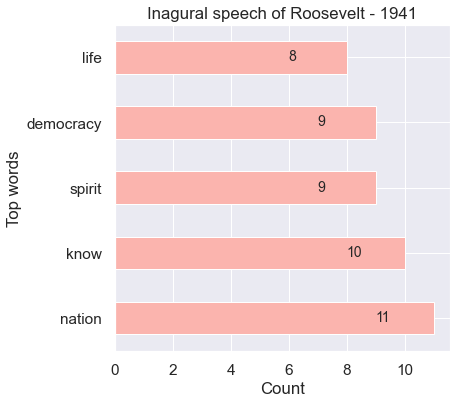
#### President Nixon:

‘vice president speaker chief justice senator cook mrs eisenhower fellow citizens great good country share together met four years ago america bleak spirit depressed prospect seemingly endless war abroad destructive conflict home meet today stand threshold new era peace world central question use peace resolve era enter postwar periods’

## 2.3 Which word occurs the most number of times in his inaugural address for each president? Mention the top three words.

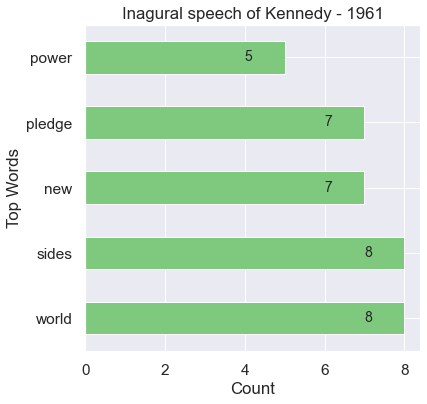
### Top five words in Roosevelt's speech (after removing the stopwords)

|  |
| --- |
| nation 11 |
| know 10 |
| spirit 9 |
| democracy 9 |
| life 8 |



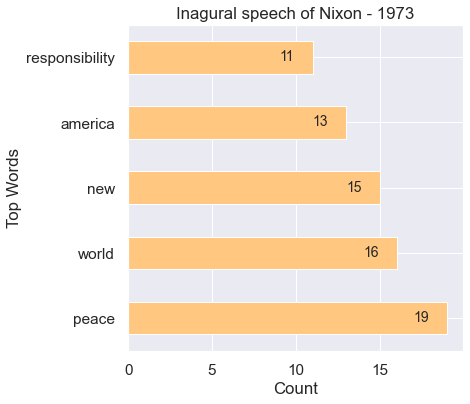
### Top five words in Kennedy's speech (after removing the stopwords)

|  |
| --- |
| world 8 |
| sides 8 |
| new 7 |
| pledge 7 |
| power 5 |



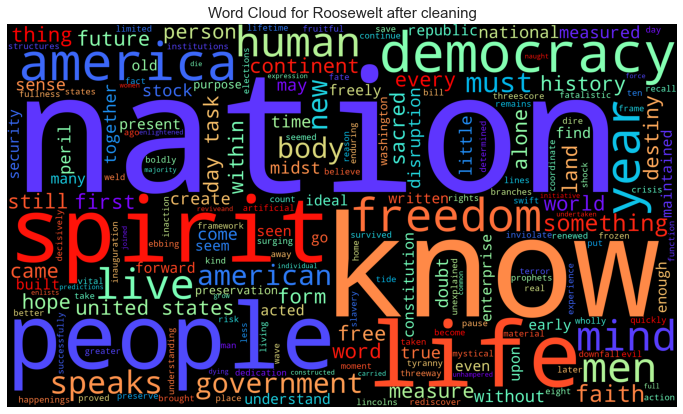
### Top five words in Nixon's speech (after removing the stopwords)

|  |
| --- |
| peace 19 |
| world 16 |
| new 15 |
| america 13 |
| responsibility 11 |

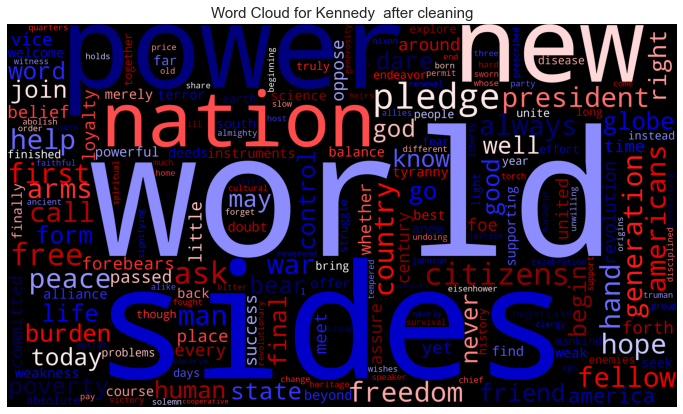


## 2.4 Plot the word cloud of each of the speeches of the variable.

### Roosevelt speech:



### Kennedy speech



### Nixon speech

