**MACHINE LEARNING**

**QUES: 1**

Tab 1:- Dataset Sample(head)--6

Tab 2:-Dataset Sample(tail)--6

Tab 3:- Dataset info--6

Tab 4:- Dataset description--7

Tab 5:- variable skewness--8

Tab 6:- Null values--8

Tab 7:-Dataset showing endoded value for categorical variable.--21

Tab 8:- Dataset info--22

Tab 9:- Model tuning train data for Logistic Regression--27

Tab 10:- Model tuning test data for Logistic Regression--27

Tab 11:- Model tuning train data for LDA--27

Tab 12:- Model tuning Test data for LDA--28

Tab 13:- Model tuning train data for Naive Bayes--28

Tab 14:- Model tuning test Data for NAive Bayes--28

Tab 15:-Model tuning train data for KNN--29

Tab 16:-Model tuning test data for KNN--29

Tab 17:-Model tuning train data for RandomForest Classifier--29

Tab 18:-Model tuning train data for RandomForest Classifier--29

Tab 19:-Accuracy data for all the models.--57

Fig 1:-Numeric Univariate Visulization--10

Fig 2:-Numeric Univariate Visulization--11

Fig 3:-Numeric Univariate Visulization--12

Fig 4:-Numeric Univariate Visulization--13

Fig 5:-Categoric Univariate Visulization--14

Fig 6:- Pairplot for Numeric Variables--15

Fig 7:- HeatMap --16

Fig 8:- Outliers--17

Fig 9:- Outliers--18

Fig 10:- Outliers --19

Fig 11:- Outliers--20

Fig 12:- Plot showing error vs K--26

Fig 13:- AUC\_ROC curve for train data of logistic Regression--31

Fig 14:- AUC\_ROC curve for test data of logistic Regression--32

Fig 15:- AUC\_ROC curve for train data of LDA--35

Fig 16:- AUC\_ROC curve for test data of LDA--36

Fig 17:- AUC\_ROC curve for train data of Naive BAyes--38

Fig 18:- AUC\_ROC curve for test data of Naive BAyes--39

Fig 19:- AUC\_ROC curve for train data of KNN--41

Fig 20:- AUC\_ROC curve for test data of KNN--42

Fig 21:- AUC\_ROC curve for train data of RandomForest classifier--44

Fig 22:- AUC\_ROC curve for test data of RandomForest classifier--45

Fig 23:-Confusion Matrix for train data(Bagging)--46

Fig 24:-AUC\_ROC curve for train data of Bagging--47

Fig 25:-Confusion Matrix for test data(Bagging)--46

Fig 26--AUC\_ROC curve for test data of Bagging--47

Fig 27:-Confusion Matrix for train data(adaboosting)--51

Fig 28:-AUC\_ROC curve for train data of adaboosting--52

Fig 29:-Confusion Matrix for test data(adaboosting)--53

Fig 30--AUC\_ROC curve for test data of adaboosting--54

Fig 31:-Confusion Matrix for train data(gradientBoosting)--55

Fig 32:-AUC\_ROC curve for train data of gradientBoosting--56

Fig 33:-Confusion Matrix for test data(gradientBoosting)--57

Fig 34--AUC\_ROC curve for test data of gradientBoosting--58

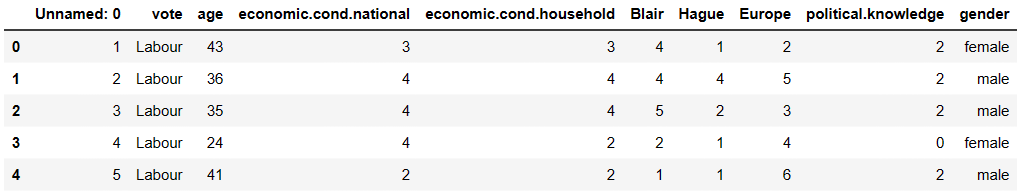
|  |  |
| --- | --- |
| **1.1) Read the dataset. Describe the data briefly. Interpret the inferences for each. Initial steps like head() .info(), Data Types, etc . Null value check, Summary stats, Skewness must be discussed.** | 4 |
| **1.2) Perform EDA (Check the null values, Data types, shape, Univariate, bivariate analysis). Also check for outliers (4 pts). Interpret the inferences for each (3 pts) Distribution plots(histogram) or similar plots for the continuous columns. Box plots. Appropriate plots for categorical variables. Inferences on each plot. Outliers proportion should be discussed, and inferences from above used plots should be there. There is no restriction on how the learner wishes to implement this but the code should be able to represent the correct output and inferences should be logical and correct.** | 7 |
| **1.3) Encode the data (having string values) for Modelling. Is Scaling necessary here or not?( 2 pts), Data Split: Split the data into train and test (70:30) (2 pts). The learner is expected to check and comment about the difference in scale of different features on the bases of appropriate measure for example std dev, variance, etc. Should justify whether there is a necessity for scaling. Object data should be converted into categorical/numerical data to fit in the models. (pd.categorical().codes(), pd.get\_dummies(drop\_first=True)) Data split, ratio defined for the split, train-test split should be discussed.** | 4 |
| **1.4) Apply Logistic Regression and LDA (Linear Discriminant Analysis) (2 pts). Interpret the inferences of both model s (2 pts). Successful implementation of each model. Logical reason should be shared if any custom changes are made to the parameters while building the model. Calculate Train and Test Accuracies for each model. Comment on the validness of models (over fitting or under fitting)** | 4 |
| **1.5) Apply KNN Model and Naïve Bayes Model (2pts). Interpret the inferences of each model (2 pts). Successful implementation of each model. Logical reason should be shared if any custom changes are made to the parameters while building the model. Calculate Train and Test Accuracies for each model. Comment on the validness of models (over fitting or under fitting)** | 4 |
| **1.6) Model Tuning (4 pts) , Bagging ( 1.5 pts) and Boosting (1.5 pts). Apply grid search on each model (include all models) and make models on best\_params. Compare and comment on performances of all. Comment on feature importance if applicable. Successful implementation of both algorithms along with inferences and comments on the model performances.** | 7 |
| **1.7 Performance Metrics: Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC\_AUC score for each model, classification report (4 pts) Final Model - Compare and comment on all models on the basis of the performance metrics in a structured tabular manner. Describe on which model is best/optimized, After comparison which model suits the best for the problem in hand on the basis of different measures. Comment on the final model.(3 pts)** | 7 |
| **1.8) Based on your analysis and working on the business problem, detail out appropriate insights and recommendations to help the management solve the business objective. There should be at least 3-4 Recommendations and insights in total. Recommendations should be easily understandable and business specific, students should not give any technical suggestions. Full marks should only be allotted if the recommendations are correct and business specific** |  |

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

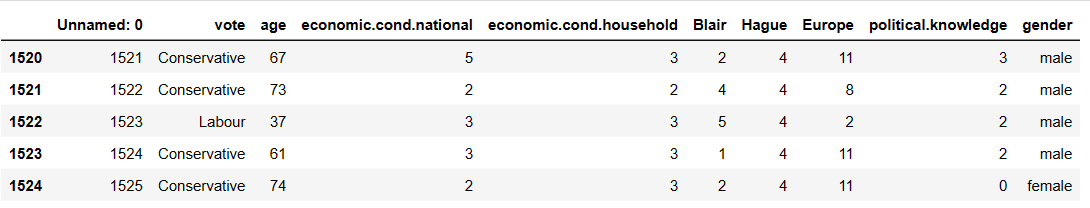
**Ans:-** We have loaded the all the required packages and loaded Election Data file using Pandas.

Dataset has 1525 rows and 10 columns.

We have viewed first and last few rows using head() and tail() functions respectively.

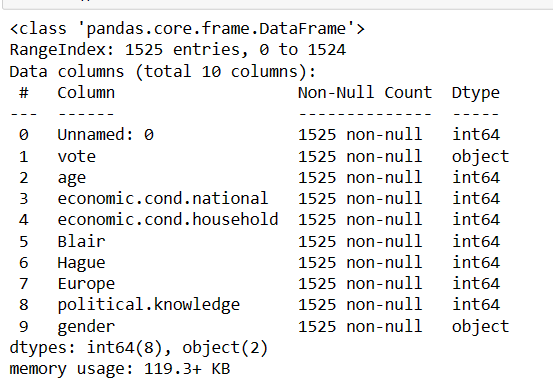
****

**Tab 1**

****

**Tab 2**

We can view dataset information using info()

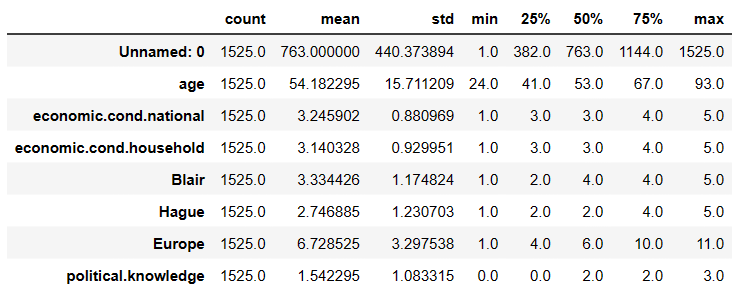
****

**Tab 3**

There are 8 variables with numeric datatype and 2 variable with object datatype.

There are no null values in the dataset.

Dataset can be described using describe() function.

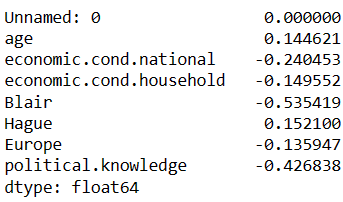


Tab 4

1. Mean,standard deviation, count etc can be seen using describe() function for all the numerical fields.
2. Minimum age for voter is 24 and maximum age is 93.Mean and 50% data is almost similar so there is almost no skewness in the data.
3. Assessment of current national economic condition and Assessment of current household economic condition ranges between 1 and 5.
4. Assessment score for Blair ranges between 1 and 5.On an Average,Blair has assessment score of 3.3.
5. Assessment score for Hague ranges between 1 and 5.On an Average,Hague has assessment score of 2.7.
6. Knowledge of parties’s Position on European integration ranges between 0 and 3.

There are no duplicate values in the Dataset.

Skewness of the data for each variable of the dataset is as below.

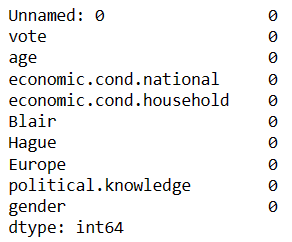


Tab 5

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

**Ans:-**

There are no null values in the dataset



Tab 6

Univariate Analysis

1. Numeric Values

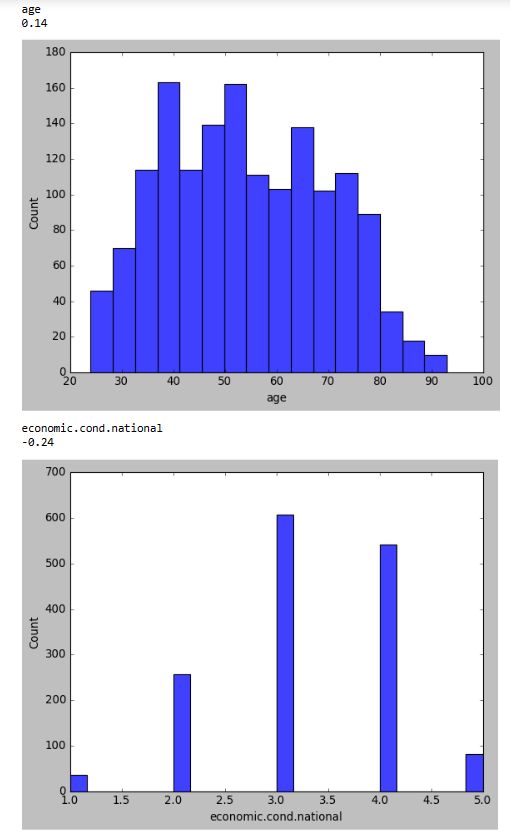


Fig 1

Observation:-

1. Both the variables have skewed distribution.
2. Age is slightly right skewed and economic.cond.national is slightly left skewed.
3. Cant say these two variable have outliers as the skewness is very little.

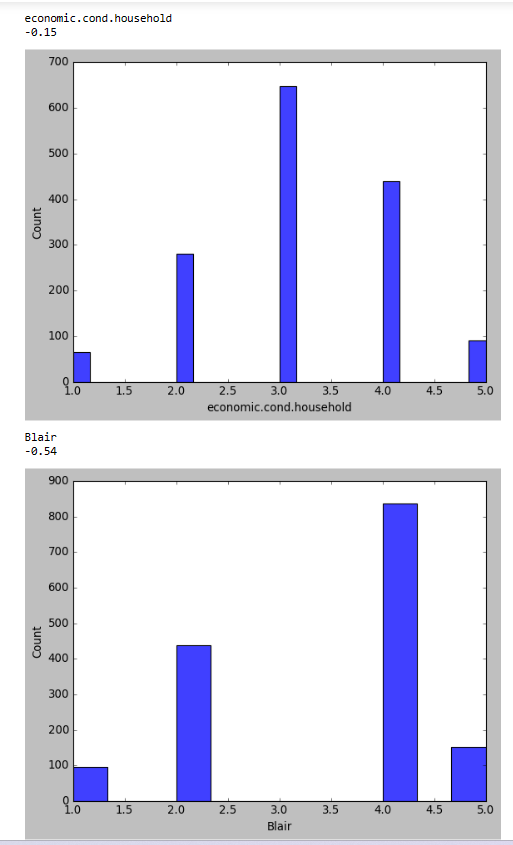


Fig 2

Observation:-

1. Both the variables have skewed distribution.
2. Both the given variables are left skewed.
3. Here we cant say if they have Outliers as the skewness is very little.

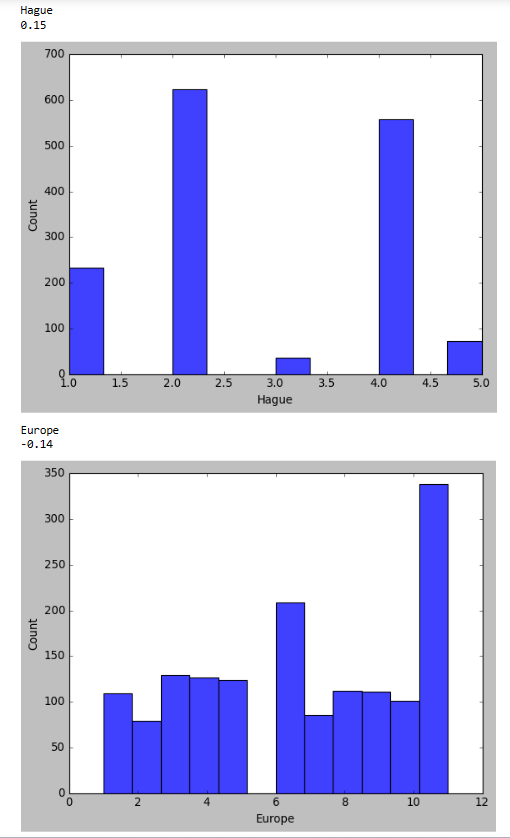


Fig 3

Observations:-

1. Both the variables given above have slight skewed distribution.
2. Hague is right skewed and Europe is left skewed with little or no Outliers.

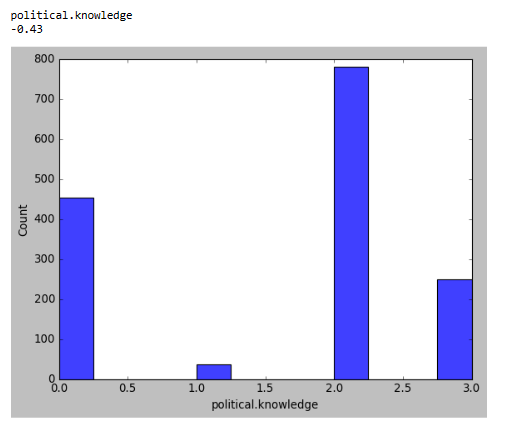


Fig 4

Observation:-

1. Political.knowledge has left skewed distribution.

Categorical Variables

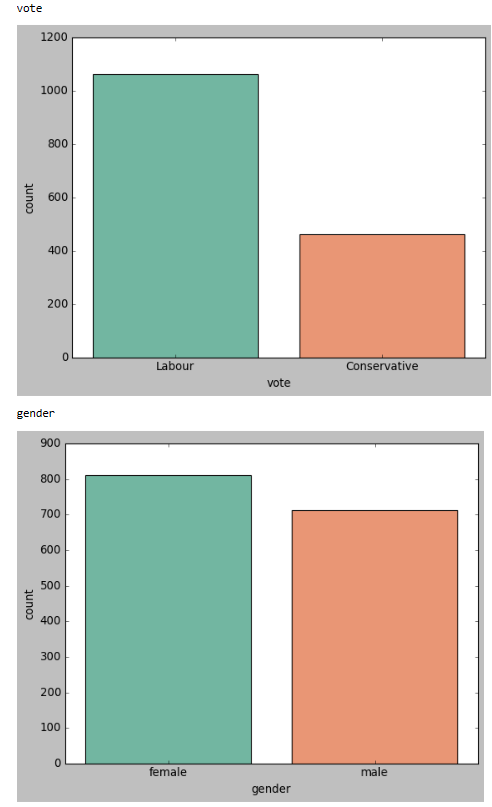


Fig 5

Observation:-

1. Most voted party is Labour than Conservative party which clearly means that Labour Party is likely to win the election.
2. There are more number of female than male.

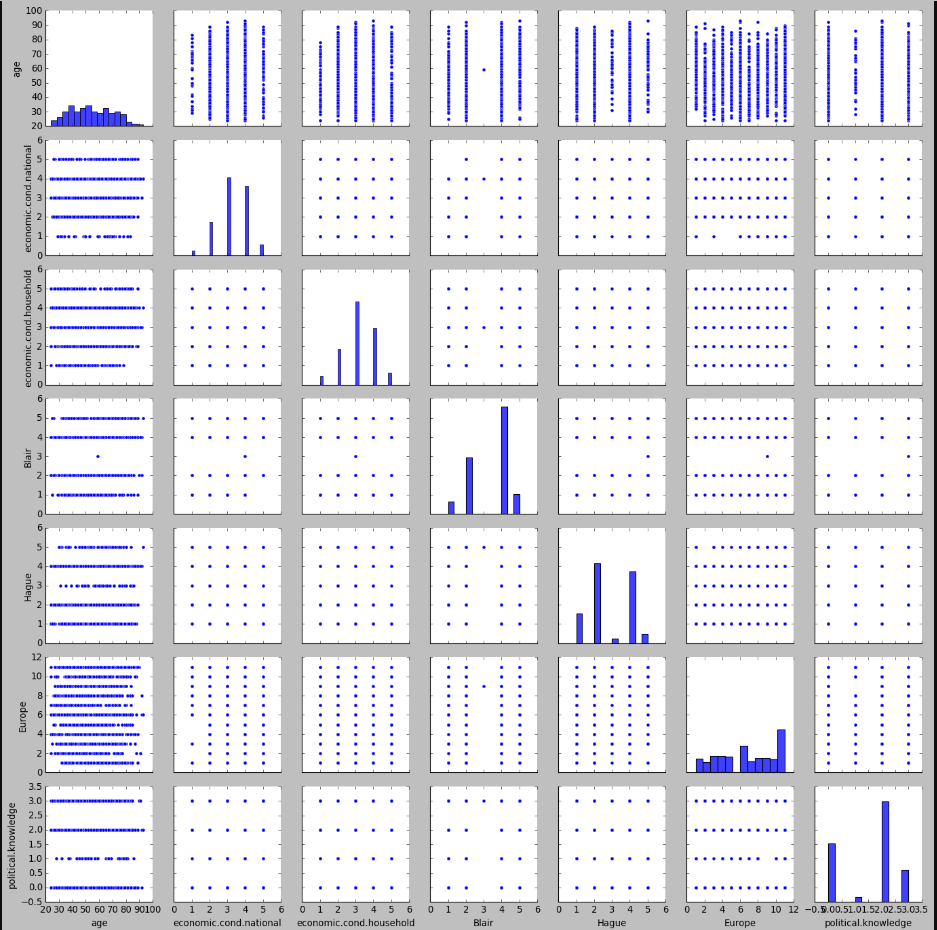


Fig 6

Pairplot shows the relationship between the variables in the form of scatterplot and the distribution of the variable in the form of histogram

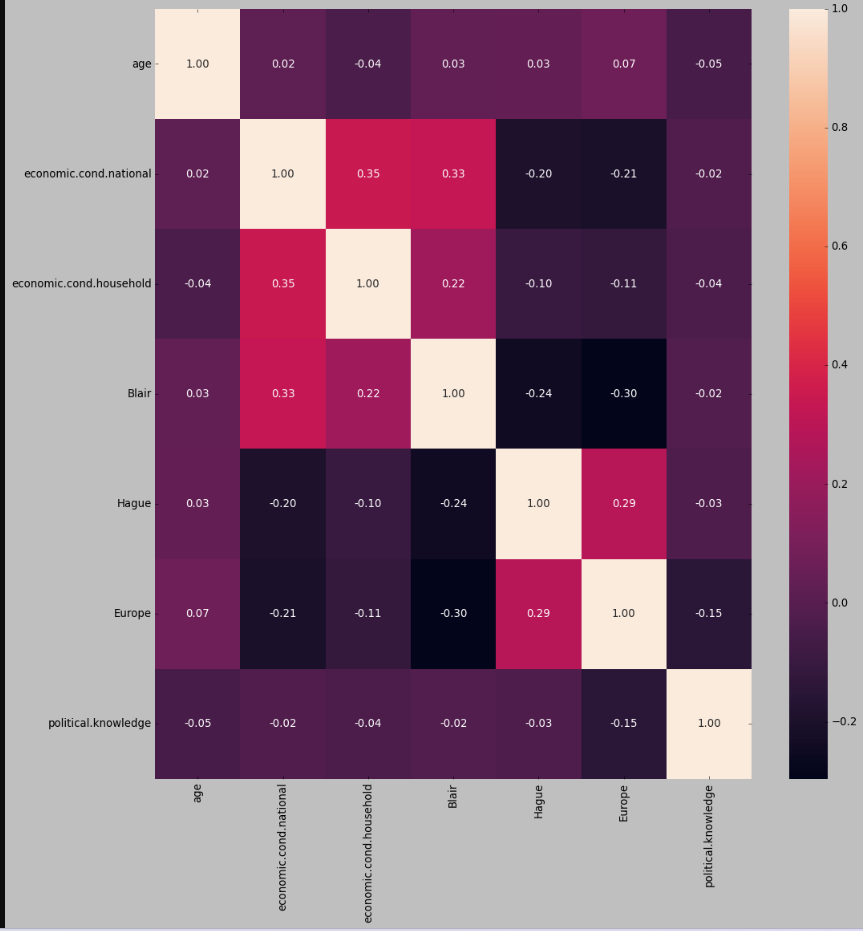


Fig 7

Observations:-

1. There is no strong Positive or Negative relation between the variables in Dataset.
2. Hague and Europe,Economic.cond.national and Economic.cond.household are mildly positively related to each other.
3. Hague has positive attitude towards Europe Integration unlike Blair.
4. Age does not decide Political knowledge of a person.

Checking for Outliers

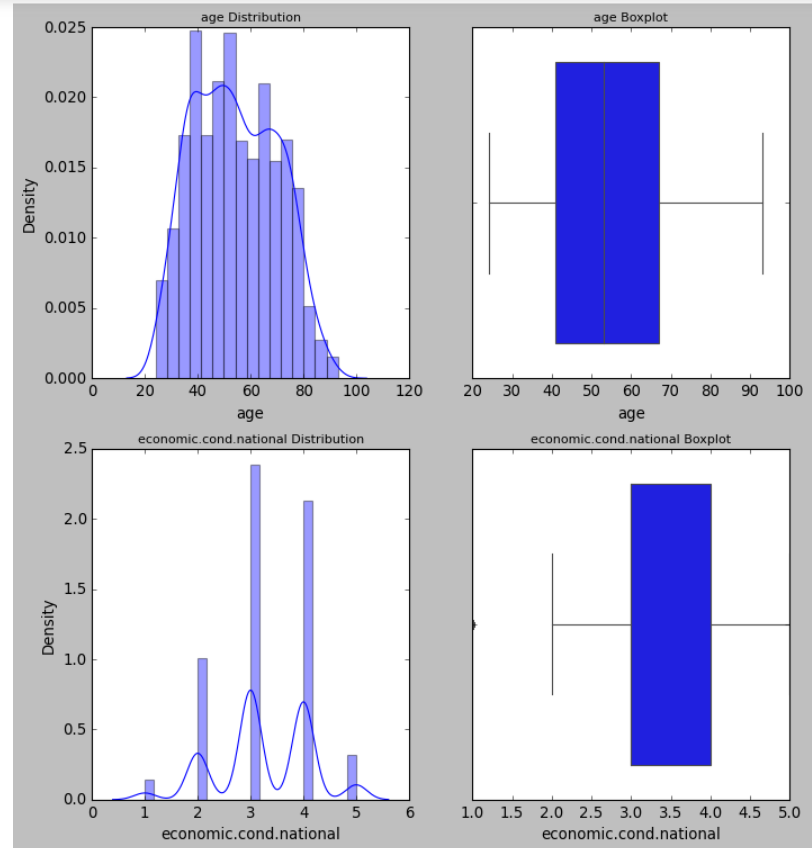


Fig 8

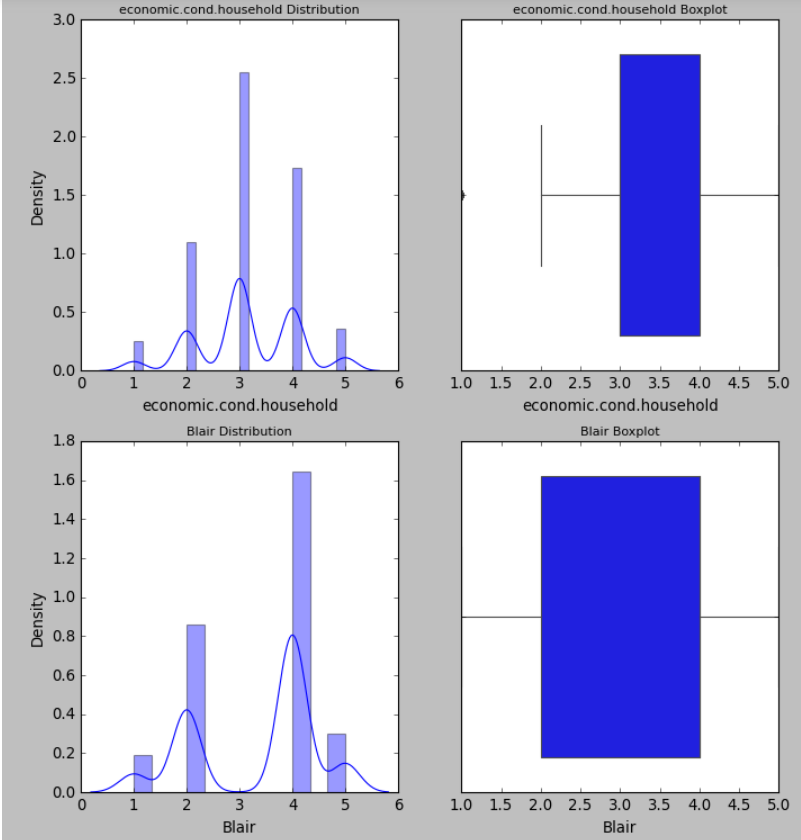


Fig 9

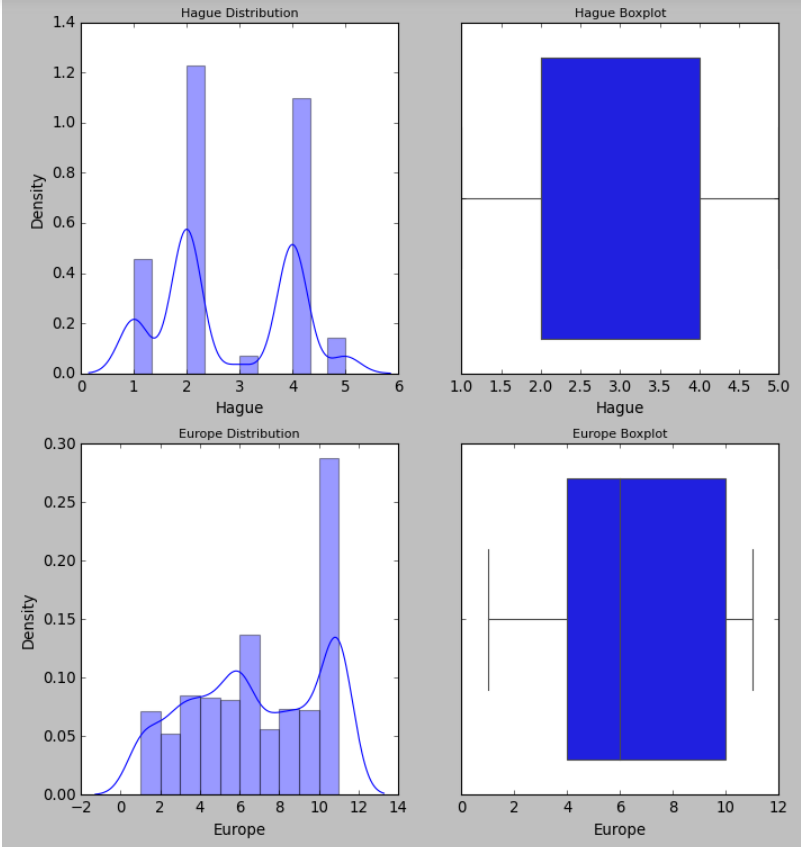


Fig 10

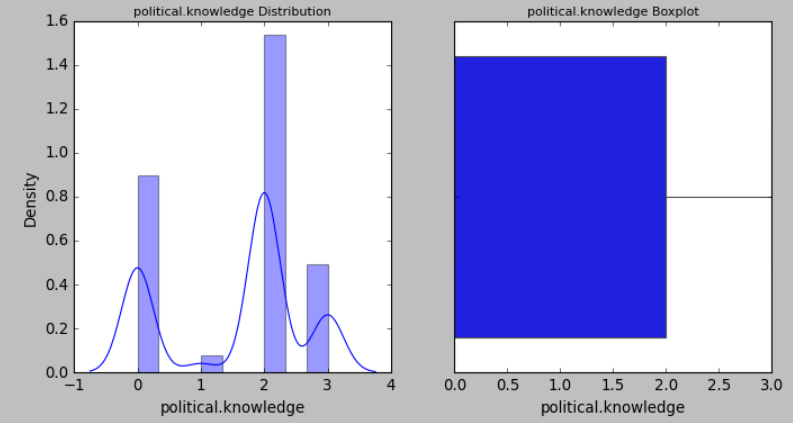


Fig 11

Observation:-

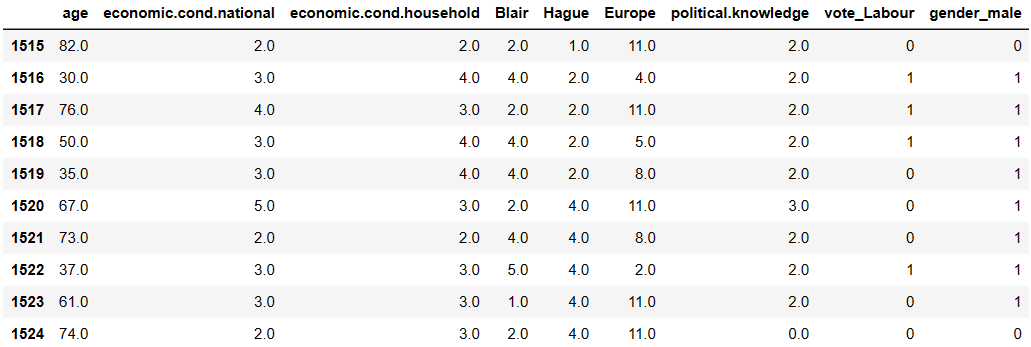
1. Outliers are not visible in the graph.
2. But because of the skewness and difference between mean and median we can say that there could be chance of having Outliers in all the above variable except age.
3. We will treat the Outliers using 1.5 \* IQR method.
4. We have treated all the Outliers if any.

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

**Ans:-**

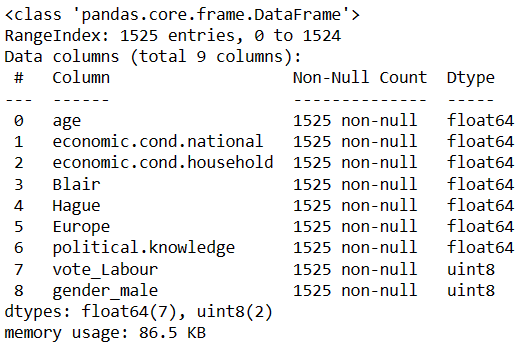
We have encoded two variables with object datatype and they are vote with two values Labour and Conservative and gender with male and female.

After encoding using pd.getdummies(drop\_first = True),data looks like shown below.



Tab 7

Now the datatype of vote and gender have changed to uint8.



Tab 8

There is no to scale the data as data for all the variables in the dataset is almost in the same range except for Age.We can perform scaling whenever there is a need to scale the data before building a model like LDA and KNN.

Standard deviation is a measure of the amount of variation or [dispersion](https://en.wikipedia.org/wiki/Statistical_dispersion) of a set of values. A low standard [deviation](https://en.wikipedia.org/wiki/Deviation_(statistics)) indicates that the values tend to be close to the [mean](https://en.wikipedia.org/wiki/Mean).

Standard deviation for all the variables in our dataset is very low which makes value very close to mean making it almost Normal distribution.

We have split the data into 70:30 ratio before building each model.

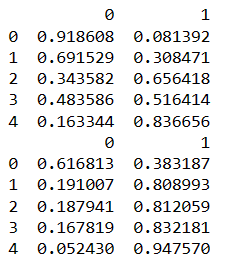
**1.4) Apply Logistic Regression and LDA (Linear Discriminant Analysis) (2 pts). Interpret the inferences of both models (2 pts). Successful implementation of each model. Logical reason should be shared if any custom changes are made to the parameters while building the model. Calculate Train and Test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

**Ans:-**

We have copied predictor variable into X dataframe and target variable into y dataframe.

Then we have split the data into train and test into 70:30 ratio.

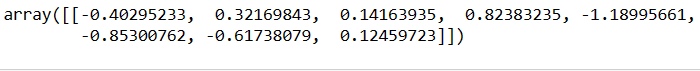
Logistic Regression

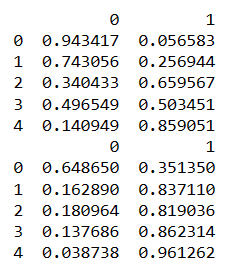
1. We have applied Logistic Regression to the training data and testing data
2. Then we have predicted test and training data
3. We have calculated predicted class and their probability
4. 
5. Accuracy for train model is 83.9
6. Accuracy for train model is 82.0
7. So Model is neither Overfit nor Underfit.
8. This cannot be considered accurate as our data is imbalanced and Accuracy is almost similar on train and test data.

LDA

1. We have fitted training data into LDA model.
2. We have calculated intercept value and coefficient for linear discriminant function.





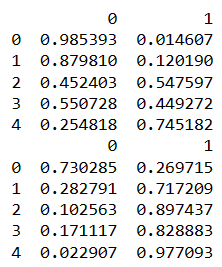
1. Training and Test Data Probability Prediction
2. 
3. Accuracy for Training data is 83.9 and for Test data is 81.8 which is almost similar.
4. So Model is neither Overfit nor Underfit.

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

**Ans:-**

Naïve Bayes Model

1. We have applied Naïve Bayes Model to the training data and testing data
2. Then we have predicted test and training data
3. We have calculated predicted class and their probability.



1. Accuracy for training set is 83.2 and for Test data is 82.3
2. So Model is neither Overfit nor Underfit.

KNN Model

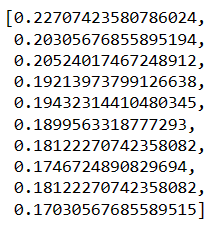
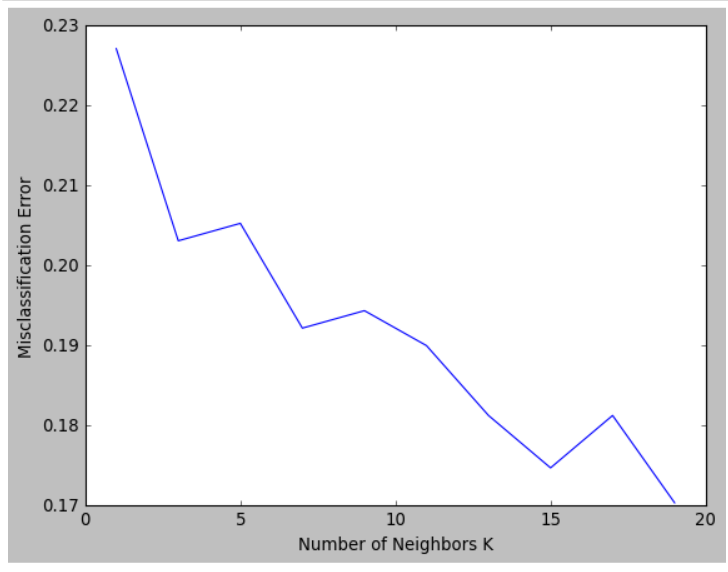
1. We have applied KNN Model to the training data and testing data
2. Before applying KNN we have scaled the predictor variables using zscore technique.
3. Then we have predicted test and training data
4. Accuracy for training data is 86.6 and for test data, Accuracy is 79.4.
5. Model is neither underfit nor overfit.
6. Accuracy for train data using 7 KNN neighbours is 85.9 and for test data is 80.7 which is still perfect data.
7. We have performed accuracy metrics for values from 1,3,5....19.
8. Below is the list that holds accuracy scores for values from 1,3,5,…19.
9. 
10. Plotted misclassification error vs k (with k value on X-axis) using matplotlib.
11. 

Fig 12

1. For K = 18 it is giving the best test accuracy lets check train.
2. Accuracy for KNN for K=18 for train is 83.7 and for test is 82.3.

**1.6)Model Tuning (4 pts) , Bagging ( 1.5 pts) and Boosting (1.5 pts). Apply grid search on each model (include all models) and make models on best\_params. Compare and comment on performances of all. Comment on feature importance if applicable. Successful implementation of both algorithms along with inferences and comments on the model performances.**

**Ans:-**

**Model Tuning**

Logistic Regression:-

1. We have perfomed Model Tuning using GridSearchCv on Logistic Regression.
2. Below are the results
3. Train Data

|  |  |  |
| --- | --- | --- |
|  | Before Grid Search | After Grid Search |
| Accuracy | 83.9 | 83.9 |
| Precision | 0.77(0 class) | 0.77(0 class) |
| Recall | 0.69(0 class) | 0.69(0 class) |
| Precision | 0.87(1 Class) | 0.87(1 Class) |
| Recall | 0.91(1 Class) | 0.91(1 Class) |

Tab 9

1. Test Data

|  |  |  |
| --- | --- | --- |
|  | Before Grid Search | After Grid Search |
| Accuracy | 82.0 | 82.0 |
| Precision | 0.70(0 class) | 0.70(0 class) |
| Recall | 0.65(0 class) | 0.65(0 class) |
| Precision | 0.87(1 Class) | 0.87(1 Class) |
| Recall | 0.89(1 Class) | 0.89(1 Class) |

Tab 10

1. There is no improvement in the performance for Logistic Regression after applying GridsearchCv.

Linear Discriminant Analysis

1. We have perfomed Model Tuning using GridSearchCv on LDA..
2. Below are the results
3. Train Data

|  |  |  |
| --- | --- | --- |
|  | Before Grid Search | After Grid Search |
| Accuracy | 83.9 | 83.9 |
| Precision | 0.76(0 class) | 0.76(0 class) |
| Recall | 0.71(0 class) | 0.71(0 class) |
| Precision | 0.87(1 Class) | 0.87(1 Class) |
| Recall | 0.90(1 Class) | 0.90(1 Class) |

Tab 11

1. Test Data

|  |  |  |
| --- | --- | --- |
|  | Before Grid Search | After Grid Search |
| Accuracy | 81.8 | 81.8 |
| Precision | 0.69(0 class) | 0.69(0 class) |
| Recall | 0.66(0 class) | 0.66(0 class) |
| Precision | 0.87(1 Class) | 0.87(1 Class) |
| Recall | 0.88(1 Class) | 0.88(1 Class) |

Tab 12

1. There is no improvement in the performance for LDA after applying GridsearchCv.

Naïve Bayes Theorem

1. We have perfomed Model Tuning using GridSearchCv on NaiveBayes Theorem.

Below are the results

1. Train Data

|  |  |  |
| --- | --- | --- |
|  | Before Grid Search | After Grid Search |
| Accuracy | 83.2 | 83.2 |
| Precision | 0.73(0 class) | 0.73(0 class) |
| Recall | 0.72(0 class) | 0.72(0 class) |
| Precision | 0.88(1 Class) | 0.88(1 Class) |
| Recall | 0.88(1 Class) | 0.88(1 Class) |

Tab 13

1. Test Data

|  |  |  |
| --- | --- | --- |
|  | Before Grid Search | After Grid Search |
| Accuracy | 82.3 | 82.3 |
| Precision | 0.68(0 class) | 0.68(0 class) |
| Recall | 0.72(0 class) | 0.72(0 class) |
| Precision | 0.89(1 Class) | 0.89(1 Class) |
| Recall | 0.86(1 Class) | 0.86(1 Class) |

Tab 14

1. There is no improvement in the performance for NaiveBayes Theorem after applying GridsearchCv.

KNN

1. We have perfomed Model Tuning using GridSearchCv on KNN.

Below are the results

1. Train Data

|  |  |  |
| --- | --- | --- |
|  | Before Grid Search | After Grid Search |
| Accuracy | 86.6 | 86.6 |
| Precision | 0.81(0 class) | 0.81(0 class) |
| Recall | 0.74(0 class) | 0.74(0 class) |
| Precision | 0.89(1 Class) | 0.89(1 Class) |
| Recall | 0.92(1 Class) | 0.92(1 Class) |

Tab 15

1. Test Data

|  |  |  |
| --- | --- | --- |
|  | Before Grid Search | After Grid Search |
| Accuracy | 79.4 | 79.4 |
| Precision | 0.65(0 class) | 0.65(0 class) |
| Recall | 0.60(0 class) | 0.60(0 class) |
| Precision | 0.85(1 Class) | 0.85(1 Class) |
| Recall | 0.87(1 Class) | 0.87(1 Class) |

Tab 16

1. There is no improvement in the performance for KNN after applying GridsearchCv

RandomForest Classifier

1. We have RandomForest Classifier to training and test data.
2. We have predicted Training and test data.
3. Accuracy for training data is 0.99 and for test data is 0.82.
4. We have perfomed Model Tuning using GridSearchCv on RandomForest Classifier.

Below are the results

1. Train Data

|  |  |  |
| --- | --- | --- |
|  | Before Grid Search | After Grid Search |
| Accuracy | 0.99 | 0.99 |
| Precision | 1.0(0 class) | 1.0(0 class) |
| Recall | 1.0(0 class) | 1.0(0 class) |
| Precision | 1.0(1 Class) | 1.0(1 Class) |
| Recall | 1.0(1 Class) | 1.0(1 Class) |

Tab 17

1. Test Data

|  |  |  |
| --- | --- | --- |
|  | Before Grid Search | After Grid Search |
| Accuracy | 82.0 | 82.0 |
| Precision | 0.68(0 class) | 0.68(0 class) |
| Recall | 0.68(0 class) | 0.68(0 class) |
| Precision | 0.68(1 Class) | 0.68(1 Class) |
| Recall | 0.68(1 Class) | 0.68(1 Class) |

Tab 18

1. There is no improvement in the performance for RandomForest Classifier after applying GridsearchCv.

We have applying Ensemble techniques on RandomForest Classifier

Bagging:-

1. Accuracy for Bagging on train data is 0.99 and for test data is 0.80
2. There is no improvement in Accuracy on train and test data after applying bagging on randomforest classifier.

AdaBoosting

1. Accuracy for AdaBoosting on train data is 0.83 and for test data is 0.82
2. There is bit improvement in performance on train and test data after applying Adaboosting on randomforest classifier. Infact there is improvement in confusion matrix. Instead of allotting all the data to single class there is proper distribution of data.

GradientBoost

1. Accuracy for GradientBoost on train data is 0.87 and for test data is 0.83
2. There is bit improvement in performance on train and test data after applying GradientBoost on randomforest classifier. Infact there is improvement in confusion matrix. Instead of allotting all the data to single class there is proper distribution of data.

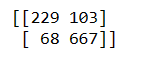
**1.7 Performance Metrics: Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC\_AUC score for each model, classification report (4 pts) Final Model - Compare and comment on all models on the basis of the performance metrics in a structured tabular manner. Describe on which model is best/optimized, After comparison which model suits the best for the problem in hand on the basis of different measures. Comment on the final model.(3 pts)**

**Ans:-**

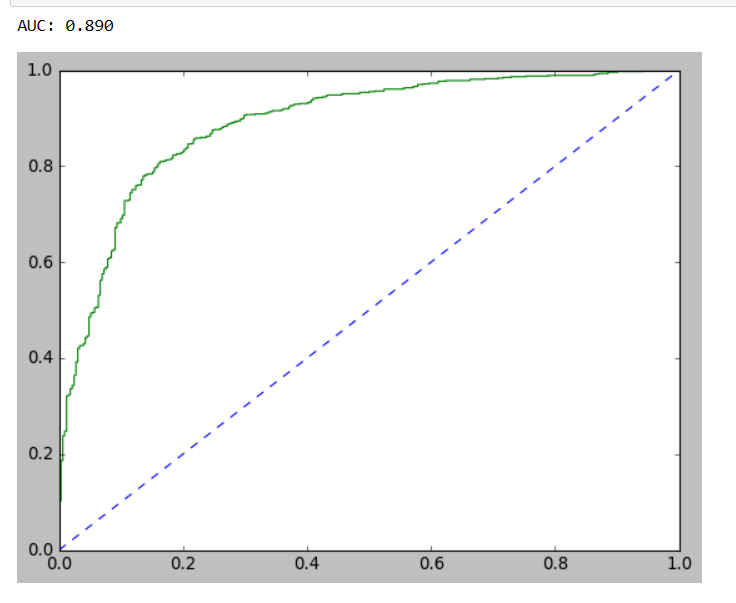
1. **Logistic Regression**

**Train data**

**Confusion matrix**

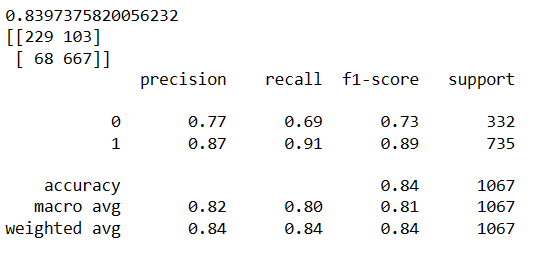
****

**ROC curve**

****

**Fig 13**

**Classification Report**

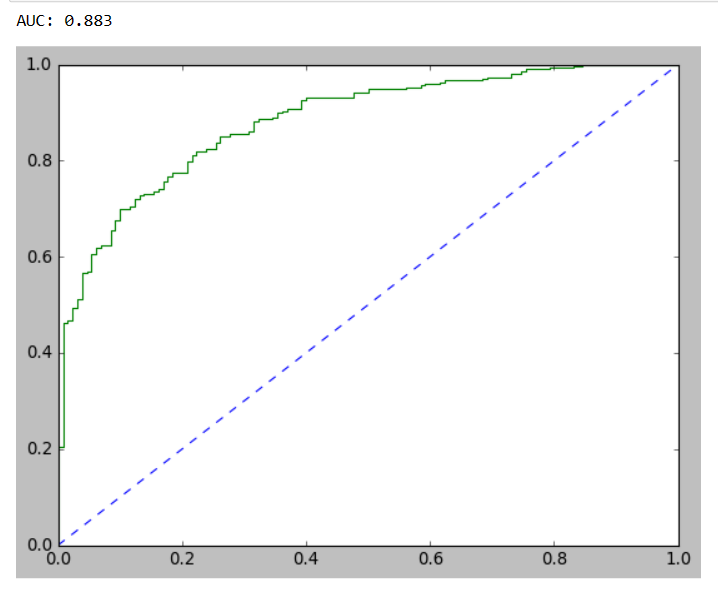
****

**Test Data**

**Confustion Matrix**

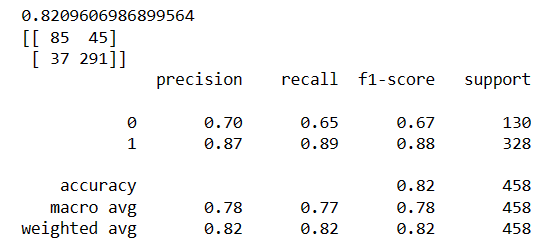
****

**ROC curve**

****

**Fig 14**

**Classification Report**

****

**Inferences**

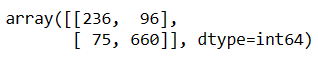
1. For predicting Conservative Party Voter (Conservative 0 ):
2. Precision (77%) – 77% of people predicted to not vote for Labour party.
3. Recall (69%) – Out of all the People who are actually not voting to Labour Party, 69% of people have been predicted correctly.
4. For predicting Labour Party Voter (Label 1):
5. Precision (87%) – 87% of people predicted to vote for Labour Party out of all People predicted to Vote for Labour Party.
6. Recall (91%) – Out of all the people actually who voted for Labour Party, 91% have been predicted correctly.

### Overall accuracy of the model – 83 % of total predictions are correct

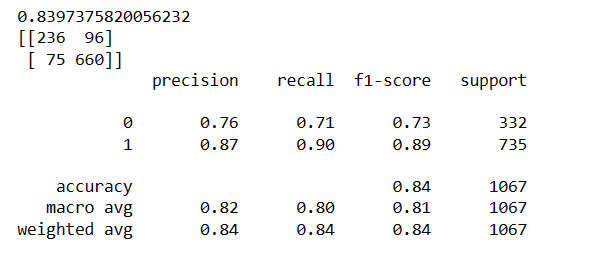
1. Accuracy, AUC, Precision and Recall for test data is almost inline with training data. This proves no overfitting or underfitting has happened, and overall the model is a good model for classification.
2. LDA

Train Data

Confusion Matrix



Classification Report



ROC curve

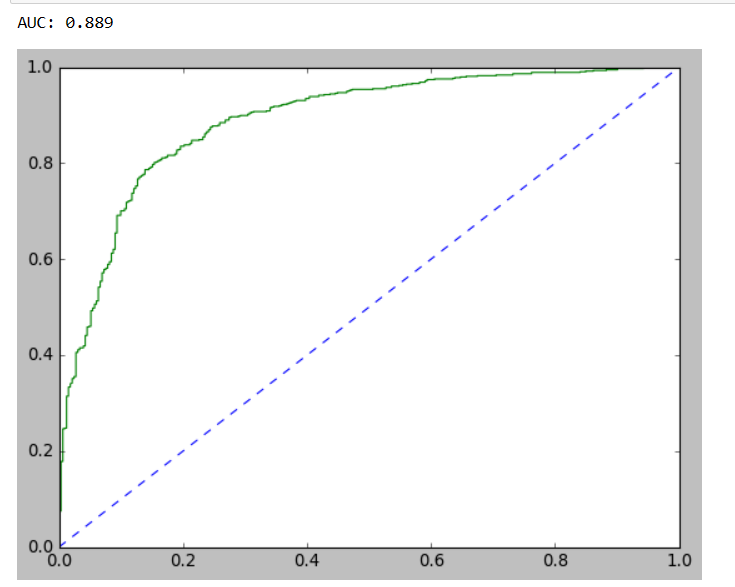
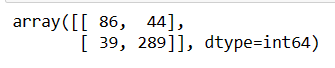


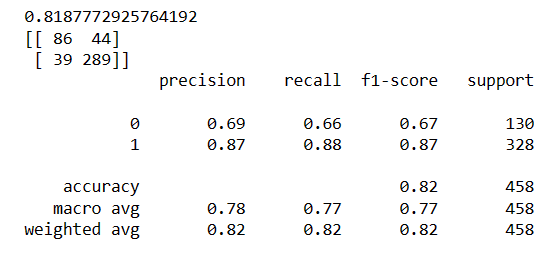
Fig 15

Test Data

Confusion Matrix



Classification Report



ROC curve

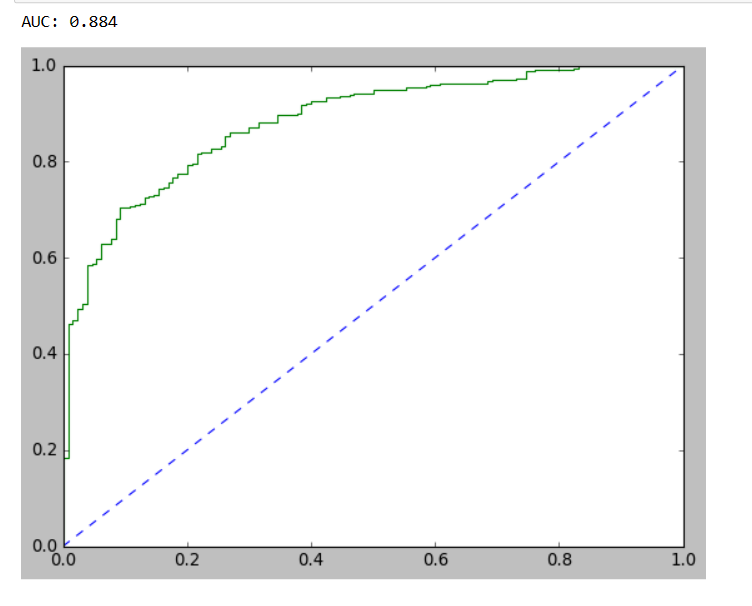


Fig 16

Observation:-

1. For predicting Conservative Party Voter (Conservative 0 ):
2. Precision (76%) – 76% of people predicted to not vote for Labour party.
3. Recall (71%) – Out of all the People who are actually not voting to Labour Party, 69% of people have been predicted correctly.
4. For predicting Labour Party Voter (Label 1):
5. Precision (87%) – 87% of people predicted to vote for Labour Party out of all People predicted to Vote for Labour Party.
6. Recall (90%) – Out of all the people actually who voted for Labour Party, 91% have been predicted correctly.

### Overall accuracy of the model – 84 % of total predictions are correct

1. Accuracy, AUC, Precision and Recall for test data is almost inline with training data. This proves no overfitting or underfitting has happened, and overall the model is a good model for classification.

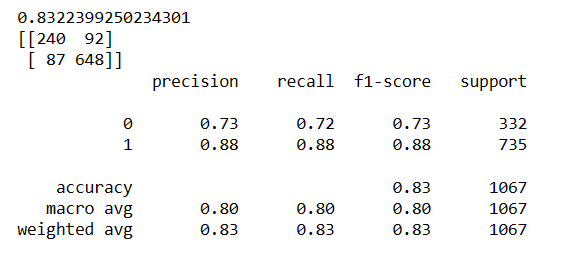
**Naïve Bayes**

**Train Data**

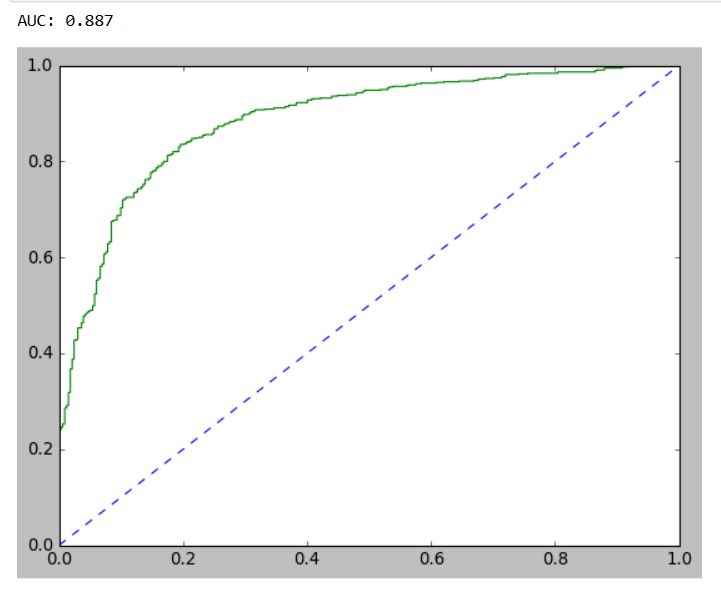
**Confusion Matrix**

****

**Classification Report**

****

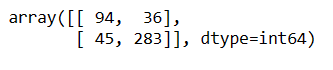
**ROC Curve**

****

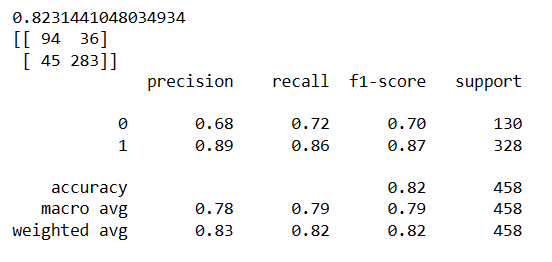
**Fig 17**

**Test Data**

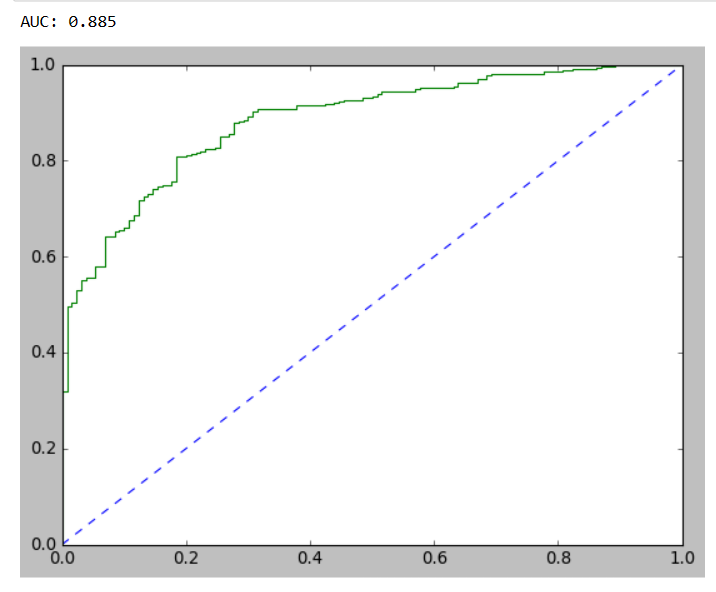
**Confusion Matrix**

****

**Classification Report**

****

**Roc Curve**

****

**Fig 18**

**Observation**

1. For predicting Conservative Party Voter (Conservative 0 ):
2. Precision (73%) – 73% of people predicted to not vote for Labour party.
3. Recall (72%) – Out of all the People who are actually not voting to Labour Party, 69% of people have been predicted correctly.
4. For predicting Labour Party Voter (Label 1):
5. Precision (88%) – 88% of people predicted to vote for Labour Party out of all People predicted to Vote for Labour Party.
6. Recall (88%) – Out of all the people actually who voted for Labour Party, 91% have been predicted correctly.

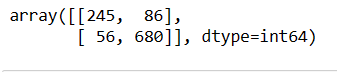
### Overall accuracy of the model – 83 % of total predictions are correct

1. Accuracy, AUC, Precision and Recall for test data is almost inline with training data. This proves no overfitting or underfitting has happened, and overall the model is a good model for classification.

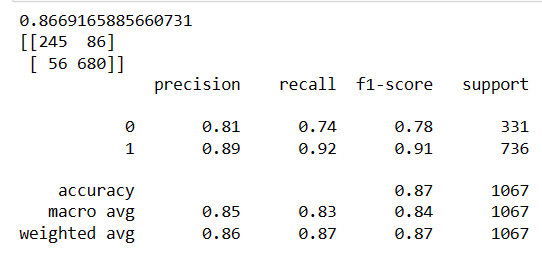
**KNN**

**Train Data**

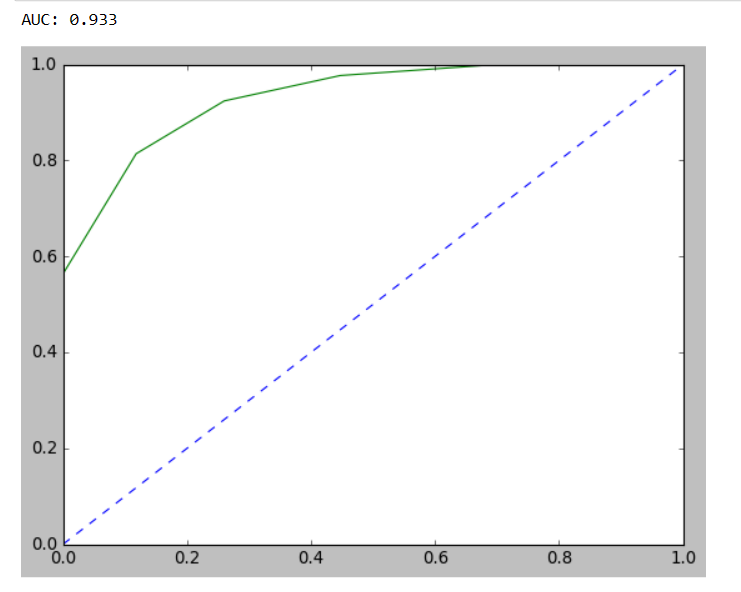
**Confusion Matrix**

****

**Classification Report**

****

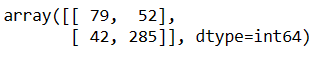
**ROC curve**

****

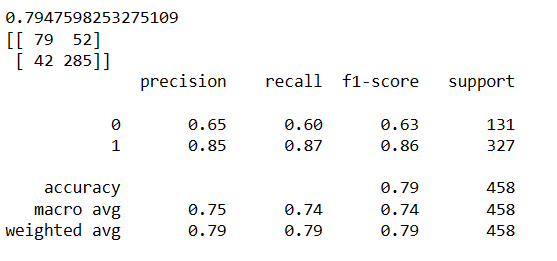
**Fig 19**

**Test data**

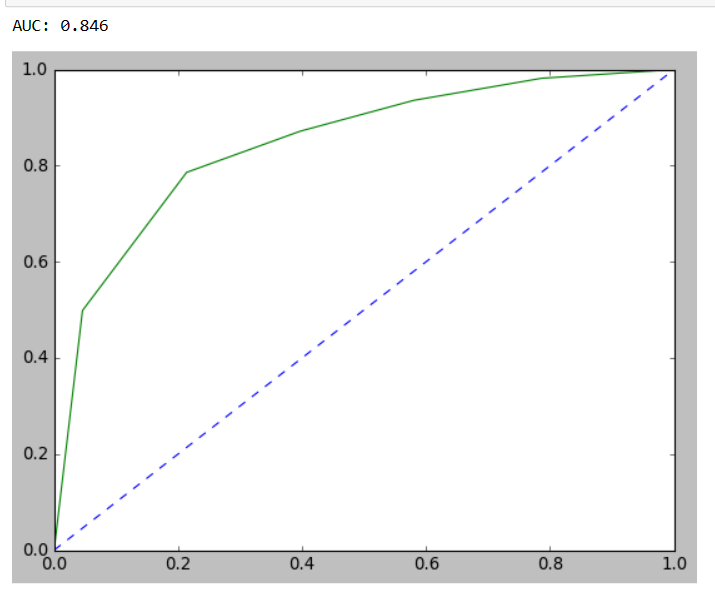
**Confusion Matrix**

****

**Classification Report**

****

**ROC Curve**

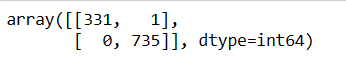
****

**Fig 20**

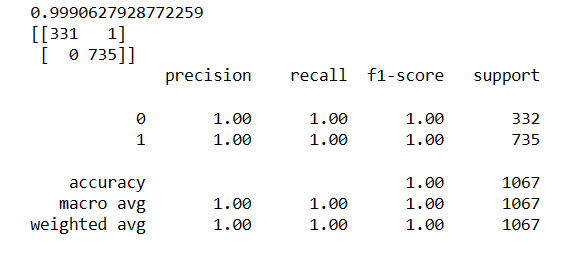
**RANDOM FOREST CLASSIFFIER**

**TRAIN DATA**

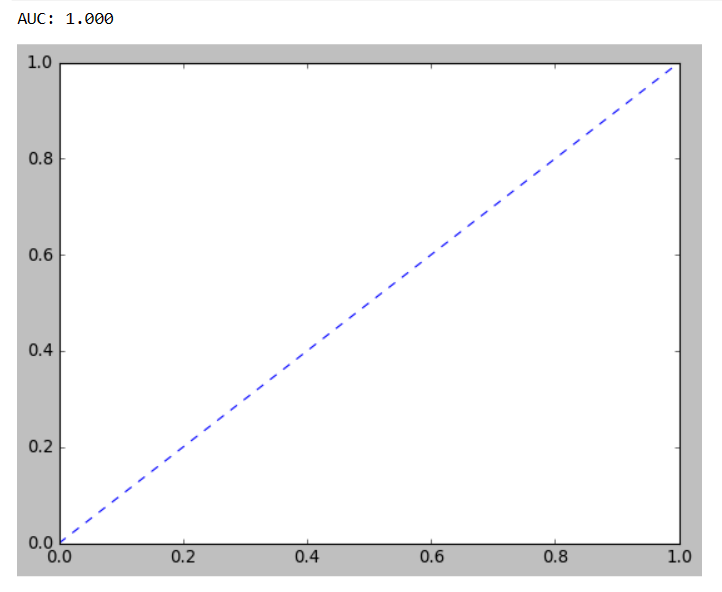
**Confusion Matrix**

****

**Classification Report**

****

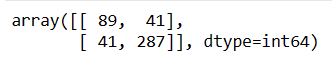
**ROC curve**

****

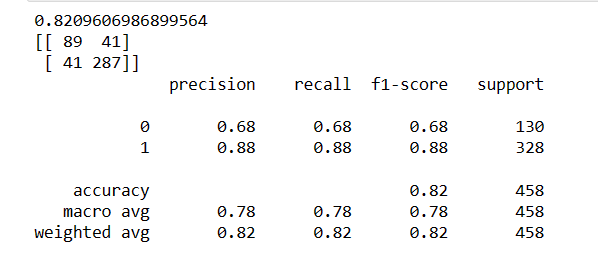
**Fig 21**

**TEST DATA**

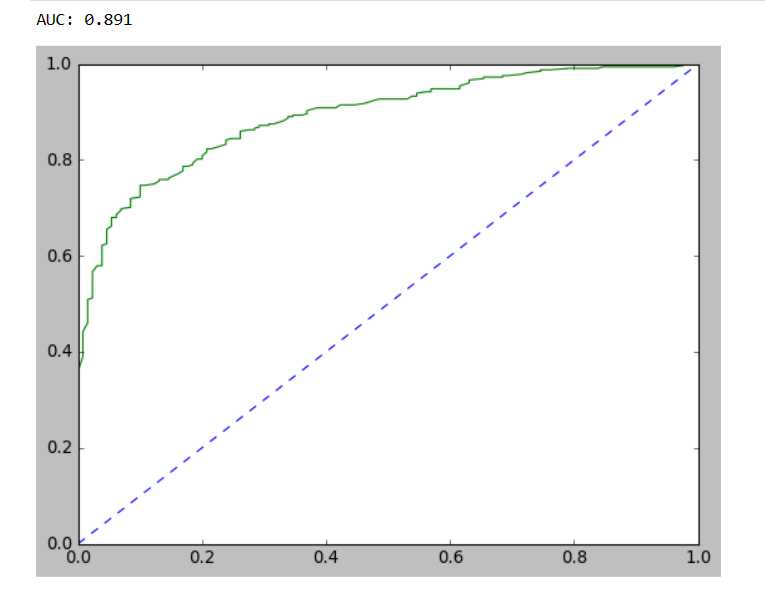
**Confusion Matrix**

****

**Classification Report**

****

**ROC Curve**

****

**Fig 22**

In this case, we can see that Classification of data is one sided. Accuracy is 0.99 which is perfect. But this is not possible in real life scenario. So we have used Ensemble techniques to improve the performance of the data.

1. Bagging

Train Data

Confusion Matrix

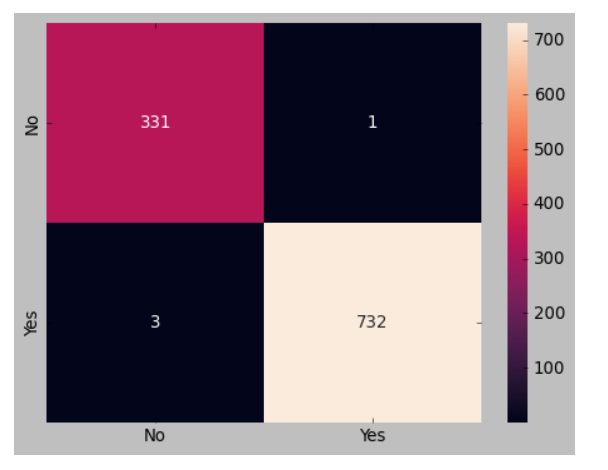
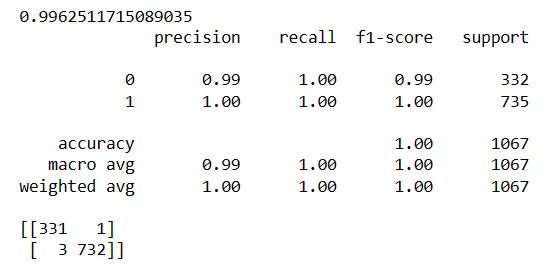


Fig 23

Classification Report



ROC curve

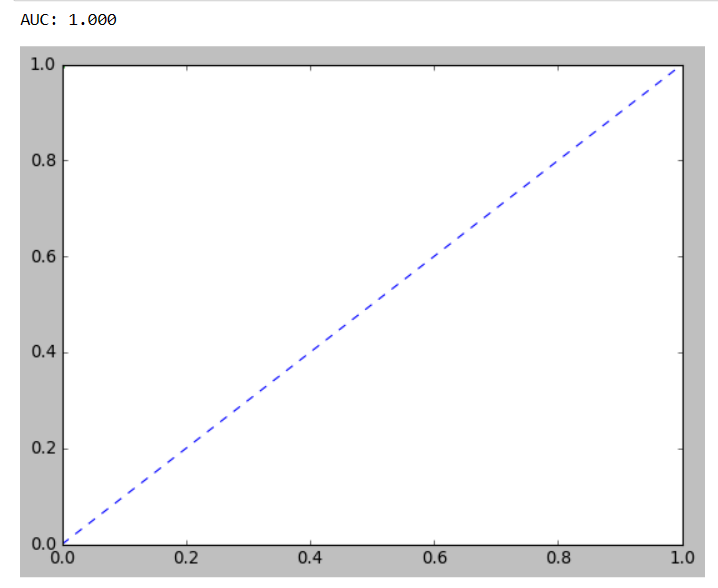


Fig 24

Test data

Confusion Matrix and Classification Report

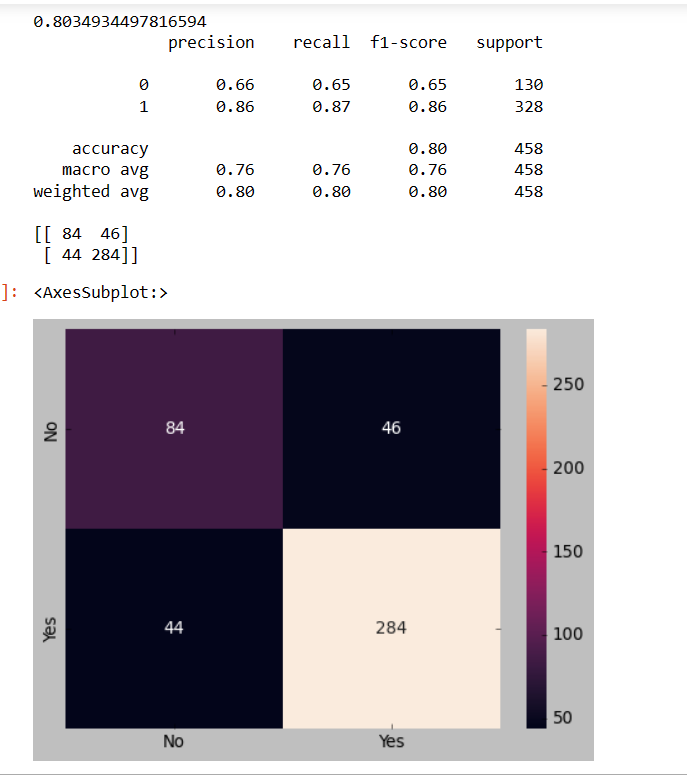


Fig 25

ROC curve

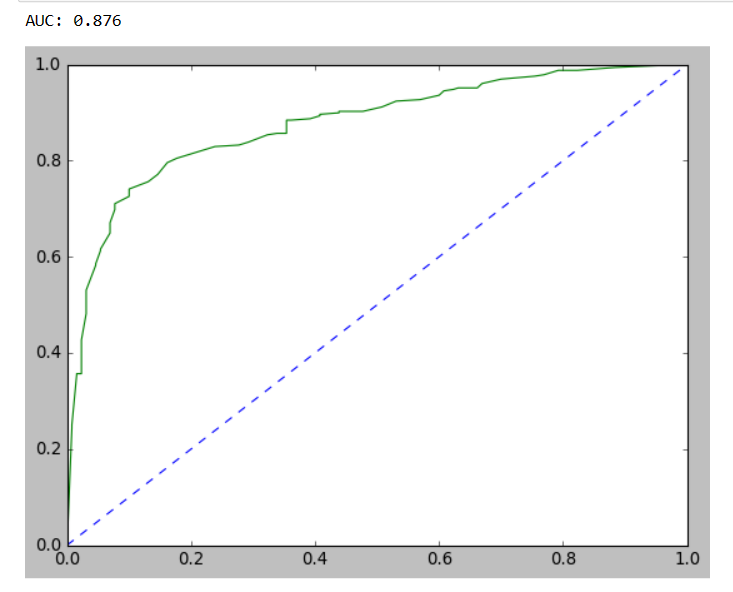


Fig 26

BOOSTING

AdaBoosting

Train

Confusion matrix and classification Report

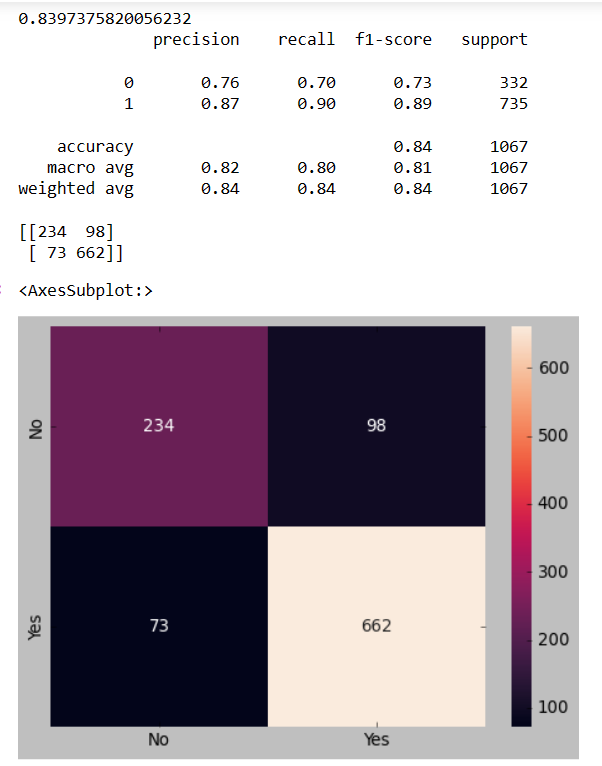


Fig 27

ROC CURVE

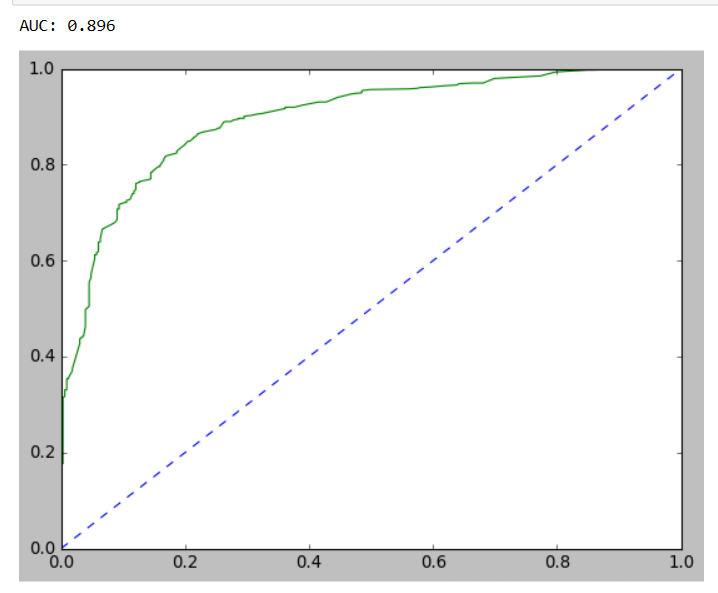


Fig 28

Test data

Confusion matrix and Claasification report

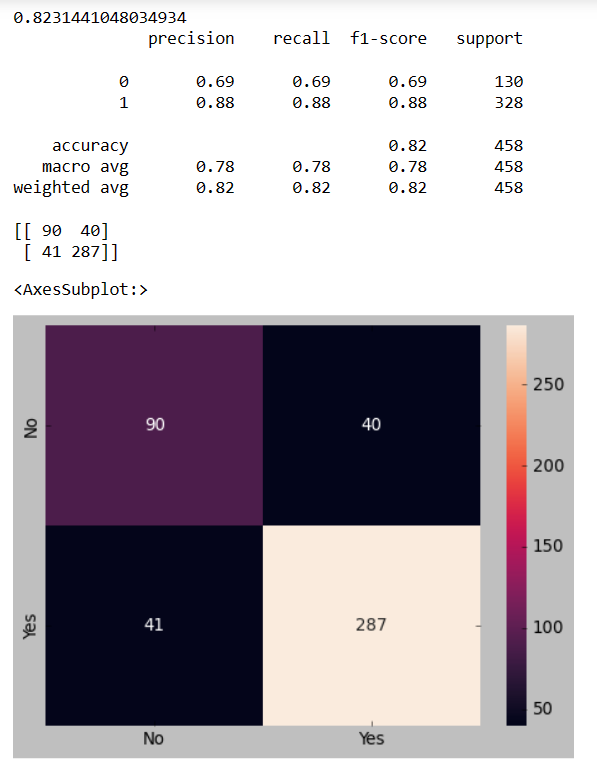


Fig 29

ROC curve

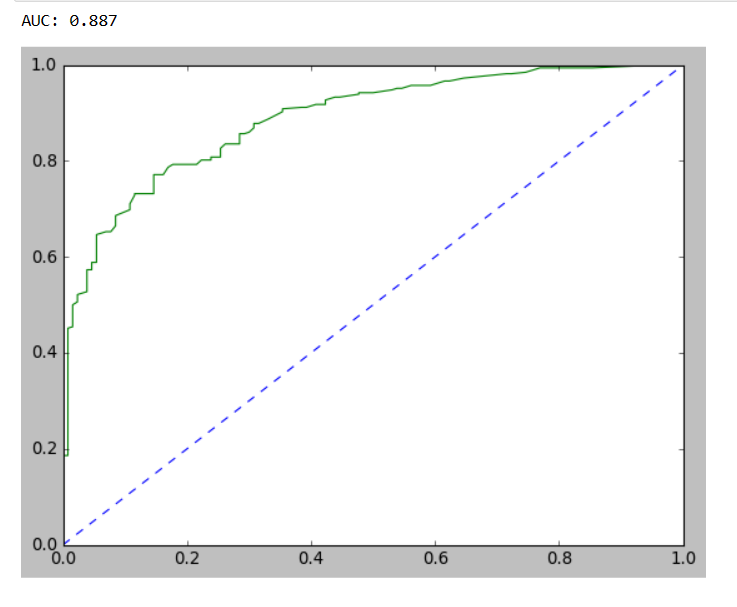


Fig 30

Gradient Boosting

Train Data

Confusion Matrix and Classification Report

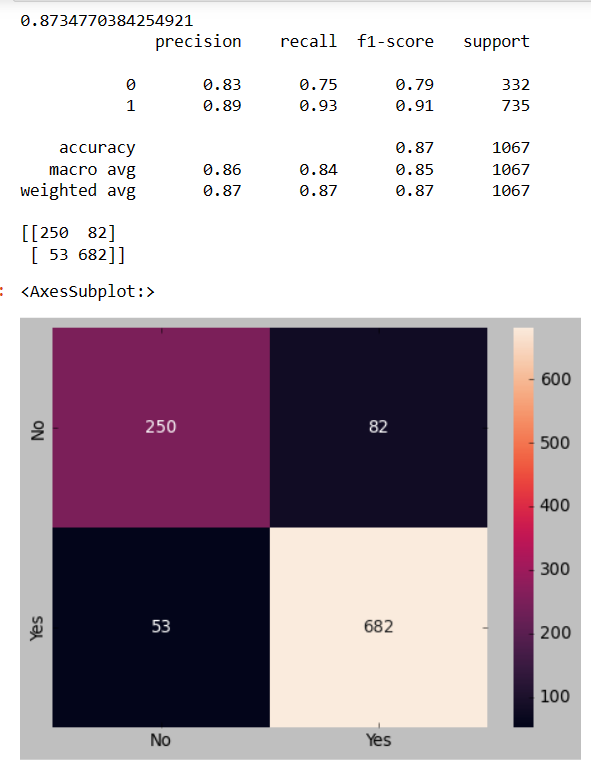


Fig 31

Roc curve

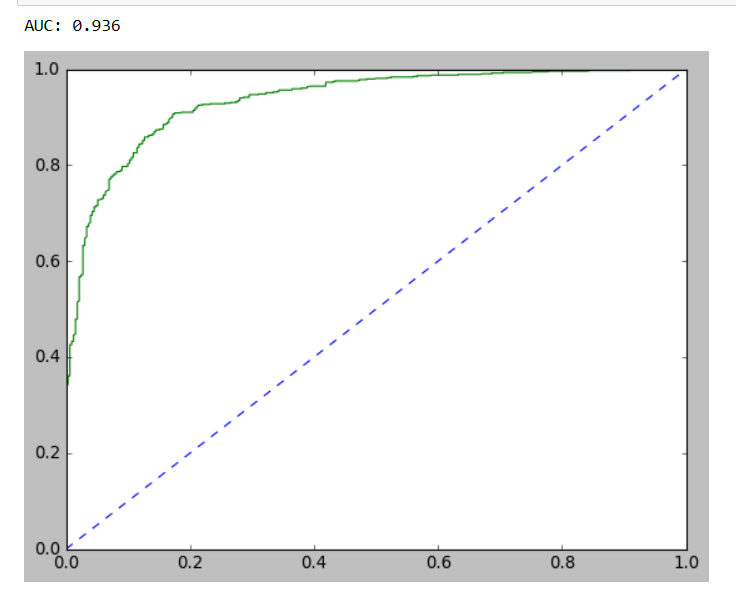


Fig 32

Test data

Confusion matrix and classification report

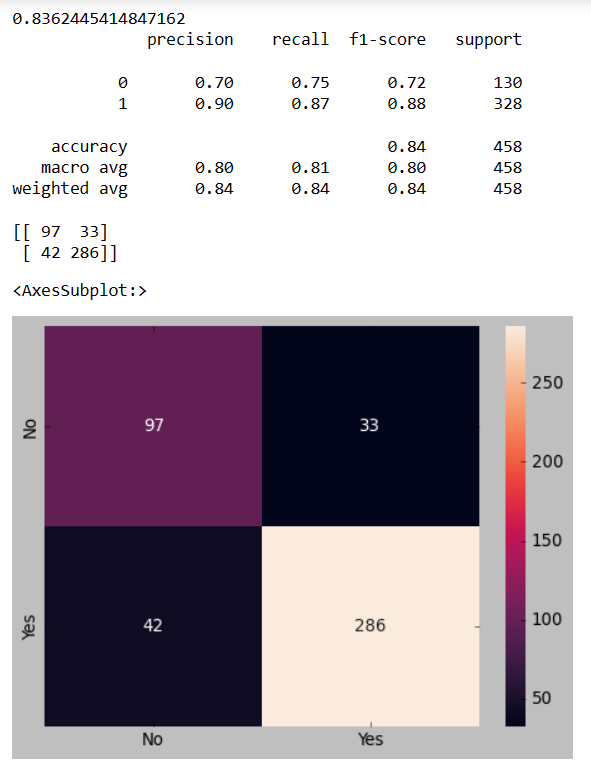


Fig 33

Roc curve

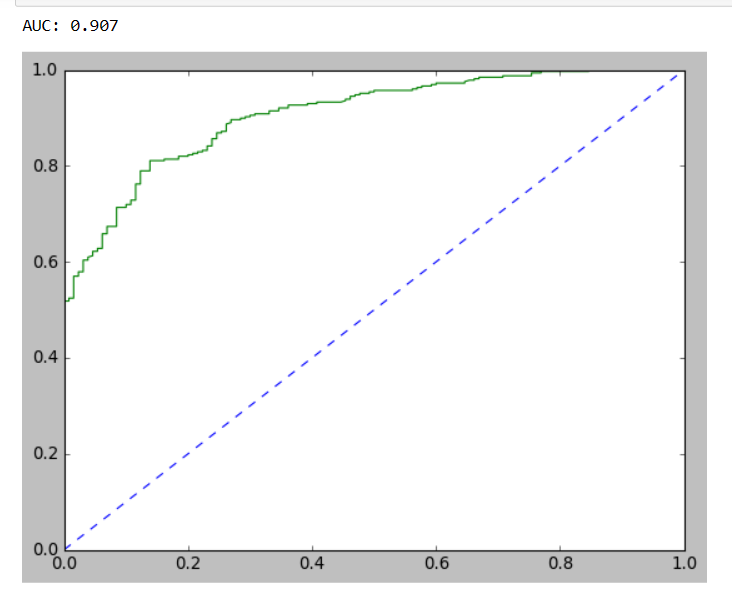


Fig 34

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Logistic | LDA | NB | KNN |
| Accuracy | 84 | 84 | 83.2 | 86.6 |
| Roc\_auc\_score | 0.89 | 0.89 | 0.88 | 0.93 |

Tab 19

We can see that KNN model has highest Accuracy which 87% . so we can say that KNN model has performed the best among all the models.

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

**Ans:-**

1. We have imported required libraries and loaded dataset
2. There are 1525 rows and 10 columns
3. There are 8 integer data type variable and 2 object datatype.
4. There were no null values in the data in interger columns
5. There were no duplicate values in the dataset.
6. Outliers were present which was treated.
7. Data for each variable(numeric and catergoric) was visualized using various visualization techniques.
8. All the data in catergorical variable was encoded into numeric values.
9. 1st we have used Logistic regression model.
10. Confusion matrix,classification report and AUC and ROC curve was plotted for both test and training data.
11. Model tuning was done on logistic regression to improve performance but ended up getting the same result for Accuracy as before.
12. 2nd we have used LDA model.
13. Confusion matrix,classification report and AUC and ROC curve was plotted for both test and training data.
14. Model tuning was done on LDA to improve performance but ended up getting the same result for Accuracy as before.
15. 3rd we have used NaiveBayes model.
16. Confusion matrix,classification report and AUC and ROC curve was plotted for both test and training data.
17. Model tuning was done on NaiveBayes to improve performance but ended up getting the same result for Accuracy as before.
18. 3rd we have used KNN model.
19. Confusion matrix,classification report and AUC and ROC curve was plotted for both test and training data.
20. Model tuning was done on KNN to improve performance but ended up getting the same result for Accuracy as before.
21. Then at last we have used Random Forest Classifier.
22. Confusion matrix,classification report and AUC and ROC curve was plotted for both test and training data.
23. We have used 2 Ensemble techniques to improve performance of RandomForest Classifier
24. 1st was Bagging which did not improve its performance as much.
25. 2nd was Boosting. Adaboosting and GradientBoosting.Boosting helped in improving performance of Random Forest Classifier Model.
26. Accuracy of KNN was Highest among all.
27. Most of the people are predicted to vote for Labour Party than Conservative Party.
28. Hague has positive attitude towards Europe Integration unlike Blair.
29. There was skewness in the data.
30. Most of the variables were showing less value of standard deviation and median for most of the value was close to mean showing almost normal distribution.
31. There are more females than males. So voter population comprises more of females.