

MACHINE LEARNING

BUSINESS REPORT

RHEA.S.M

PGPDSBA Online Sep_B 2021

Table of Contents

| | |
|---|----|
| 1. Problem-1: Modelling..... | 4 |
| 1.1. Objective..... | 4 |
| 1.2. Descriptive and Exploratory Data Analysis | 4 |
| 1.2.1. Descriptive Data analysis: | 4 |
| 1.2.2. Univariate and Bivariate data Analysis: | 5 |
| 1.2.3. Correlation analysis: | 9 |
| 1.2.4. Outlier Analysis:..... | 10 |
| 1.3. Categorical Variables Treatment and Scaling | 11 |
| 1.3.1. Encoding of the Variables: | 11 |
| 1.3.2. Scaling of Variables: | 12 |
| 1.3.3. Data Split: | 12 |
| 1.4. Logistic Regression Analysis vs. LDA..... | 12 |
| 1.4.1. LR Models Performance and Inference: | 12 |
| 1.4.2. LDA Models Performance and Inference..... | 14 |
| 1.5. KNN Model and Naïve Bayes Model..... | 15 |
| 1.5.1. Naïve Bayes Models Performance and Inference:..... | 15 |
| 1.5.2. KNN Model Performance and Inference: | 16 |
| 1.6. Bagging and Boosting | 18 |
| 1.6.1. Bagging Performance and Inference: | 18 |
| 1.6.2. Boosting Performance and Inference:..... | 19 |
| 1.7. Models Performance and Inference | 21 |
| 1.8. Insights and Recommendations | 22 |
| 2. Problem-2: Text Mining and Analysis | 23 |
| 2.1. Objective..... | 23 |
| 2.2. Background..... | 23 |
| 2.3. Analysis Methodology..... | 23 |

List of Figures

| Figure No. | Name | Page No. |
|------------|--|----------|
| Fig 1 | Histograms with KDE and Box plots for continuous and ordinal variables | 6 |
| Fig 2 | Count plot for Categorical variables | 7 |
| Fig 3 | Box plots: Target Variable vs. continuous and ordinal variables. | 7 |
| Fig 4 | Swarm plots: Target Variable vs. continuous and ordinal variables. | 8 |
| Fig 5 | Pair Plot | 9 |
| Fig 6 | Heat map or Correlation plot for continuous/ordinal variables | 10 |
| Fig 7 | Outlier Analysis | 10 |

List of Tables

| Table No. | Name | Page No. |
|-----------|---|----------|
| Table 1 | Summary of Descriptive statistics information | 5 |
| Table 2 | Encoding of Categorical variables | 11 |
| Table 3 | Logistic Regression | 13 |
| Table 4 | Linear Discriminant Analysis | 14 |
| Table 5 | Naïve Bayes | 15 |
| Table 6 | KNN | 17 |
| Table 7 | Bagging | 18 |
| Table 8 | Boosting | 19 |
| Table 9 | Comparison between different models | 21 |
| Table 10 | Length of words and sentences | 23 |
| Table 11 | Length of words after removing stop words and punctuation | 23 |
| Table 12 | Three most common words | 23 |

1. Problem-1: Modelling

1.1. Objective

- The objective the problem is 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.
- We additionally also have to choose a Final Model, after comparing between all the models built and write inferences with regards to which model is best/optimized.

1.2. Descriptive and Exploratory Data Analysis

Background: 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:

| | |
|--------------------------|--|
| vote: | Party choice: Conservative or Labour. |
| age: | Age in years. |
| economic.cond.national: | Assessment of current national economic conditions, 1 to 5. |
| economic.cond.household: | Assessment of current household economic conditions, 1 to 5. |
| Blair: | Assessment of the Labour leader, 1 to 5. |
| Hague: | Assessment of the Conservative leader, 1 to 5. |
| Europe: | An 11-point scale that measures respondents' attitudes toward European integration. High scores represent 'Eurosceptic' sentiment. |
| political.knowledge: | Knowledge of parties' positions on European integration, 0 to 3. |
| gender: | Male or Female |

1.2.1. Descriptive Data analysis:

- Provided data set consists of total 9 variables in which 8 independent variables and one dependent variable.
 - a) Independent variables: age, economic.cond.national, economic.cond.household, Blair, Hague, Europe, political.knowledge.
 - b) Dependent variable: 'vote'.
- Data set contains total of 1525 entries among which 7 integer type variables and 2 object type variables.
- Duplicates were verified, 8 duplicate rows were present in the data set which were removed before further analysis was done on the data. There are no null values in the dataset.
- The following Table 1 consists the head(), tail(), info() and description both normal and statistical of the dataset at hand.
- In the Statistical description, it is valid that Political Knowledge has zero values since the rating scale to assess said knowledge starts from 0-3.

Table-1: Summary of Descriptive statistics information

Head and Tail of the 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 |

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

Info of dataset:

```
<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   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)
memory usage: 107.4+ KB
```

Describe function on dataset:

| | count | unique | top | freq | mean | std | min | 25% | 50% | 75% | max |
|-------------------------|--------|--------|--------|------|-----------|-----------|------|------|------|------|------|
| vote | 1525 | 2 | Labour | 1063 | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| age | 1525.0 | NaN | NaN | NaN | 54.182295 | 15.711209 | 24.0 | 41.0 | 53.0 | 67.0 | 93.0 |
| economic.cond.national | 1525.0 | NaN | NaN | NaN | 3.245902 | 0.880969 | 1.0 | 3.0 | 3.0 | 4.0 | 5.0 |
| economic.cond.household | 1525.0 | NaN | NaN | NaN | 3.140328 | 0.929951 | 1.0 | 3.0 | 3.0 | 4.0 | 5.0 |
| Blair | 1525.0 | NaN | NaN | NaN | 3.334426 | 1.174824 | 1.0 | 2.0 | 4.0 | 4.0 | 5.0 |
| Hague | 1525.0 | NaN | NaN | NaN | 2.746885 | 1.230703 | 1.0 | 2.0 | 2.0 | 4.0 | 5.0 |
| Europe | 1525.0 | NaN | NaN | NaN | 6.728525 | 3.297538 | 1.0 | 4.0 | 6.0 | 10.0 | 11.0 |
| political.knowledge | 1525.0 | NaN | NaN | NaN | 1.542295 | 1.083315 | 0.0 | 0.0 | 2.0 | 2.0 | 3.0 |
| gender | 1525 | 2 | female | 812 | NaN | NaN | NaN | NaN | NaN | NaN | NaN |

Null and Duplicates:

There are no null values.

Number of duplicated rows= 8

- Shape Before Duplicates Removal (1525, 9)
- Shape After Duplicates Removal: (1517, 9)

Statistical description of the dataset after removing duplicates:

| | age | economic.cond.national | economic.cond.household | Blair | Hague | Europe | political.knowledge |
|---------------|---------|------------------------|-------------------------|---------|---------|---------|---------------------|
| TOTAL RECORDS | 1517.00 | 1517.00 | 1517.00 | 1517.00 | 1517.00 | 1517.00 | 1517.00 |
| ZERO COUNT | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 454.00 |
| MIN | 24.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0.00 |
| MAX | 93.00 | 5.00 | 5.00 | 5.00 | 5.00 | 11.00 | 3.00 |
| RANGE | 69.00 | 4.00 | 4.00 | 4.00 | 4.00 | 10.00 | 3.00 |
| MEAN | 54.24 | 3.25 | 3.14 | 3.34 | 2.75 | 6.74 | 1.54 |
| MEDIAN | 24.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0.00 |
| MODE | 37.00 | 3.00 | 3.00 | 4.00 | 2.00 | 11.00 | 2.00 |
| VARIANCE | 246.54 | 0.78 | 0.87 | 1.38 | 1.52 | 10.88 | 1.18 |
| STD DEV | 15.70 | 0.88 | 0.93 | 1.17 | 1.23 | 3.30 | 1.08 |
| Q1 | 24.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0.00 |
| Q3 | 25.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0.00 |
| IQR | 26.00 | 1.00 | 1.00 | 2.00 | 2.00 | 6.00 | 2.00 |
| SKEWNESS | 0.14 | -0.24 | -0.14 | -0.54 | 0.15 | -0.14 | -0.42 |
| KURTOSIS | -0.94 | -0.26 | -0.21 | -1.06 | -1.39 | -1.24 | -1.22 |

1.2.2. Univariate and Bivariate data Analysis:

- Univariate analysis is the simplest form of analyzing data. Univariate data requires to analyze each variable separately. While, a Bivariate analysis will measure the correlations between two variables.
- Figure-1 shows individual distributions for all the continuous and ordinal variables present in the data set. It is observed that most variables are normally distributed except for 'age' and 'Europe' which is right skewed. The boxplots for each of the variables have also been plotted alongside the respective histograms.

- There seem to be outliers in 'economic.cond.national' and 'economic.cond.household' but since these are ordinal variables they should not be treated.

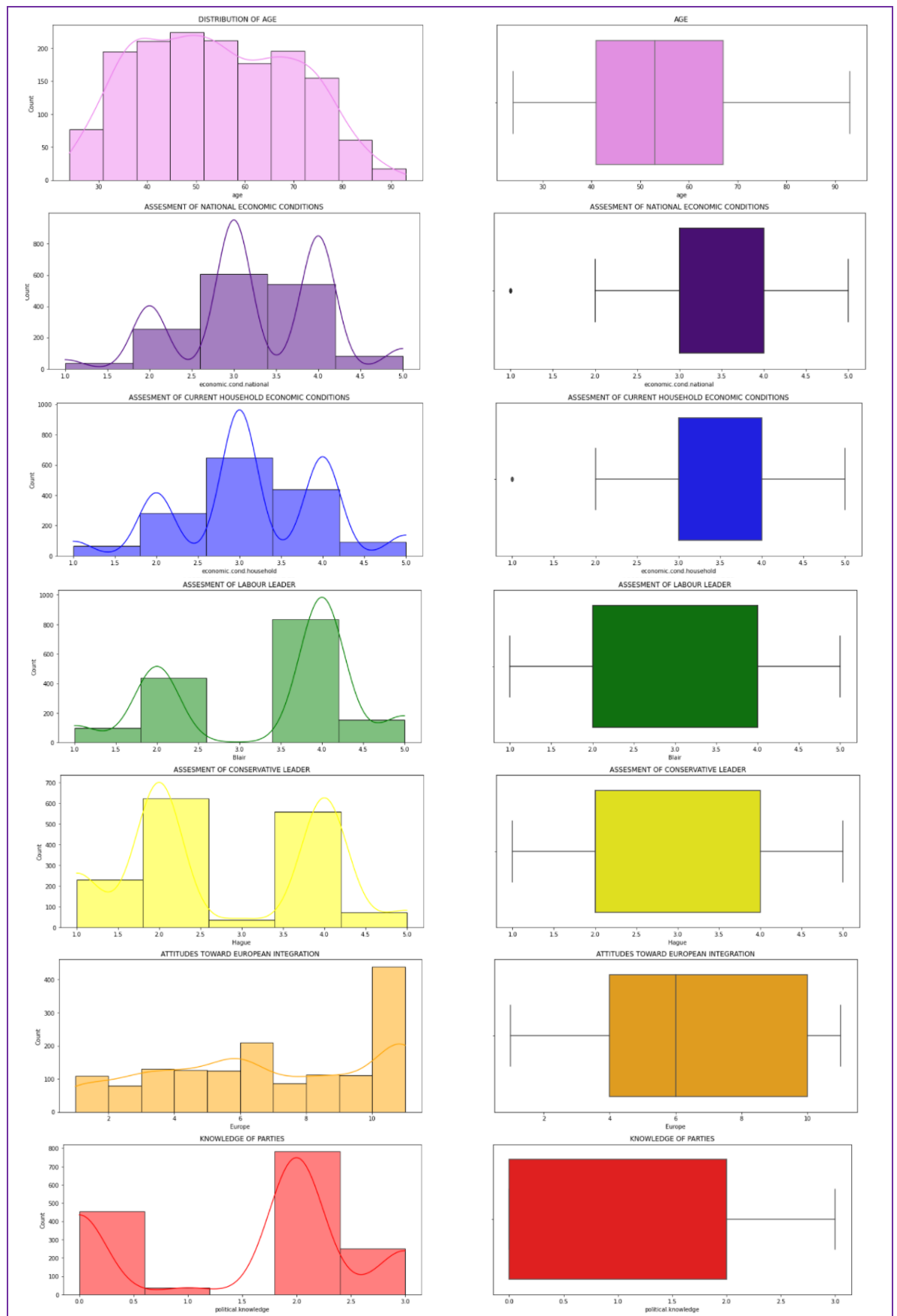


Figure-1 Histograms with KDE and Box plots for continuous and ordinal variables

- Figure-2 shows count plot of categorical variables of vote and gender. From the figure it is observed that there are more female voters than males in the dataset provided. And the maximum voters prefer to have a labour leader rather than a conservative one.

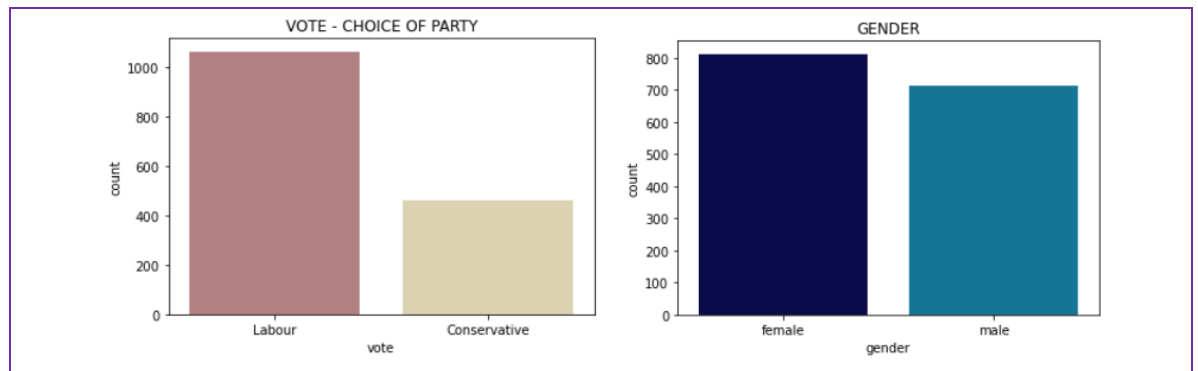


Figure-2 Count plot for Categorical variables

- Figure-3 shows us the comparison between the target variable and continuous and ordinal variables using box plots. The average age of voters preferring a Labour Leader are about 50 years, while voters preferring a conservative leader are 60 years.

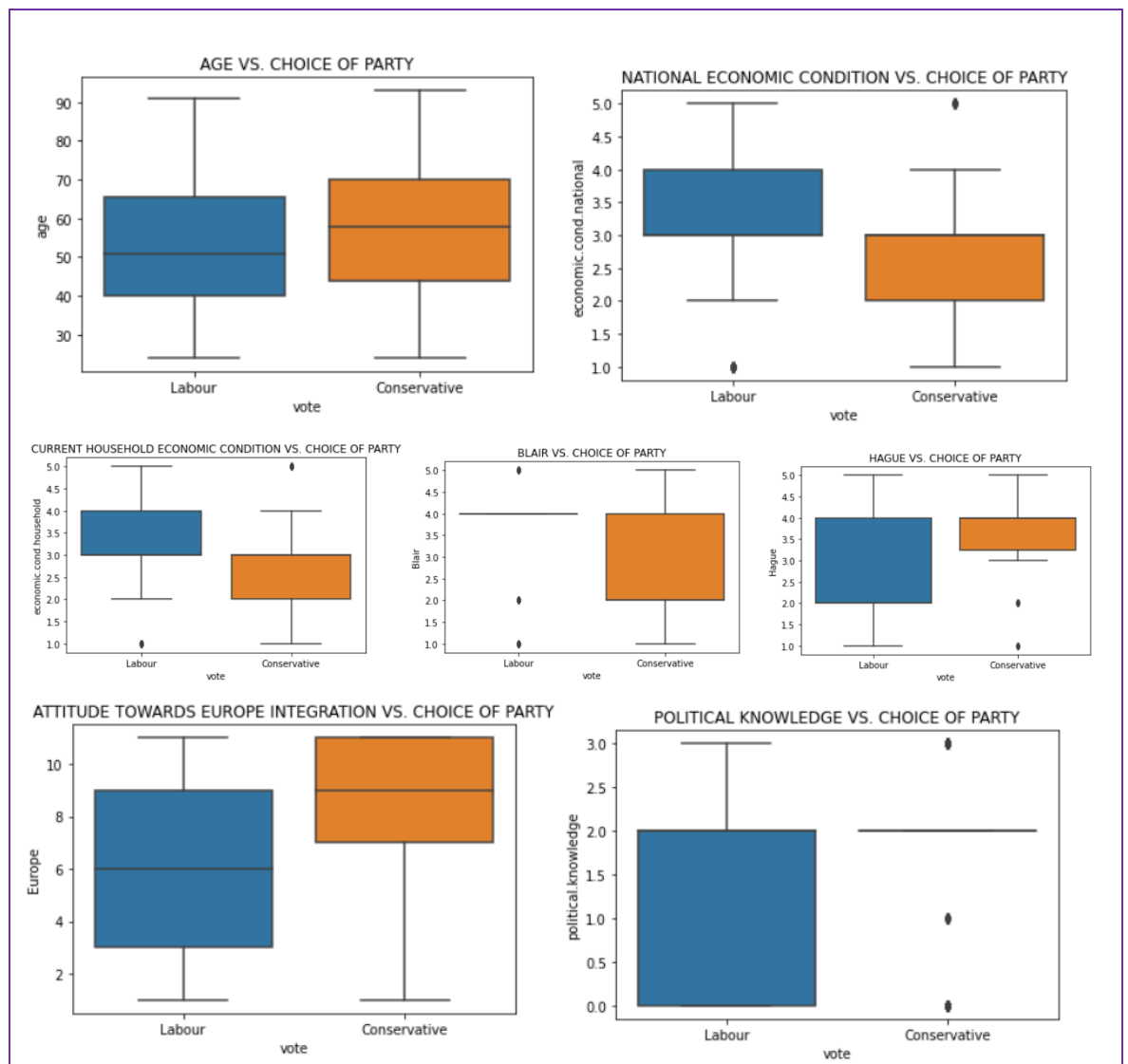


Figure-3 Box plots: Target Variable vs. continuous and ordinal variables.

- Figure-4 shows us the comparison between the target variable and continuous and ordinal variables using swarm plots. We can see that there are a lot of voters from various age groups leaning to a Labour leader. The assessment of the national economic condition according to the voters seem to be better with a Labour leader. There is not much difference in the Assessment of the current household conditions.

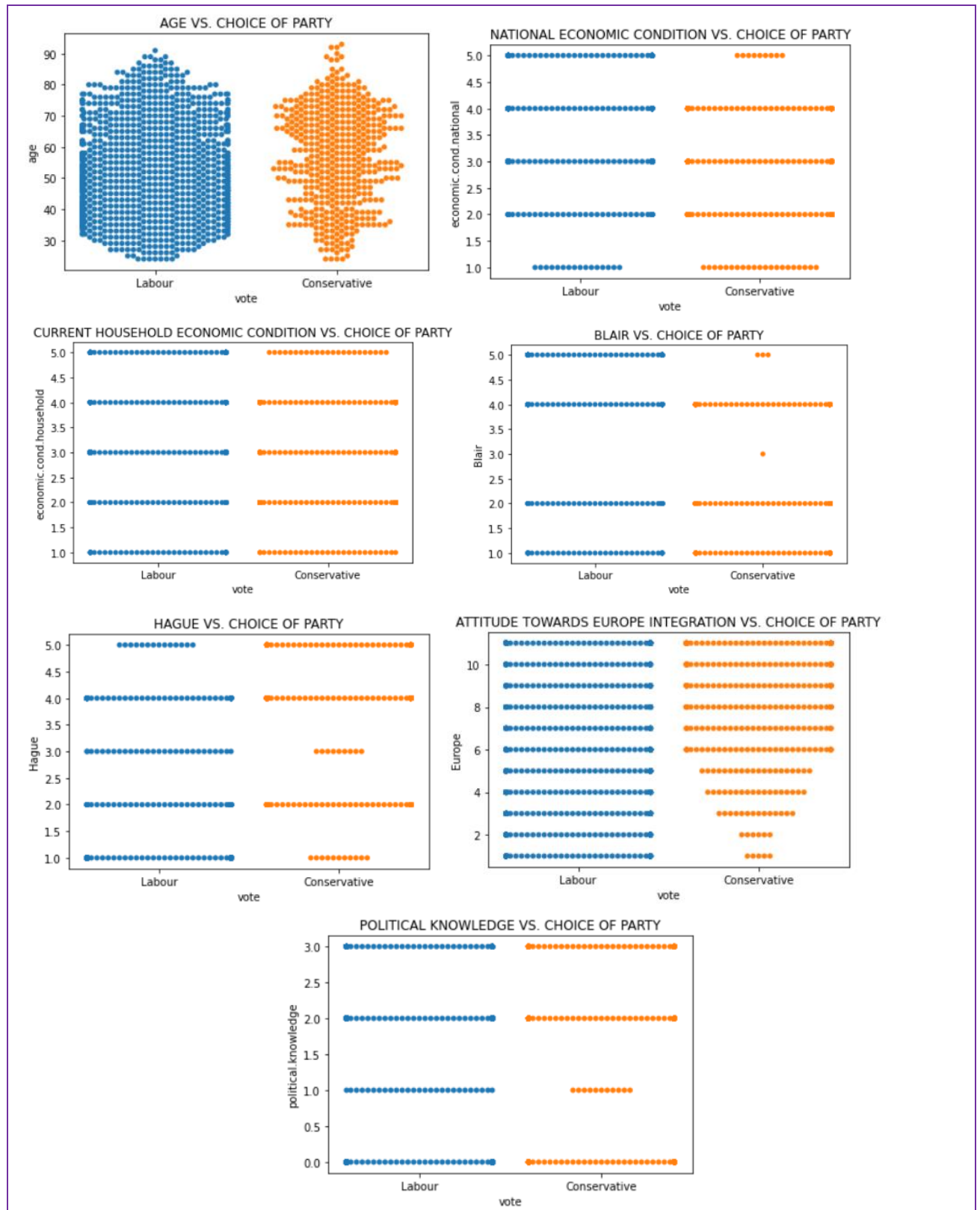


Figure-4 Swarm plots: Target Variable vs. continuous and ordinal variables.

- A pairplot plot a pairwise relationships in a dataset. Figure-5 represents the pair plot of continuous/ordinal with target variable = vote set as a hue to help determine and interpret relationships with distribution plots.

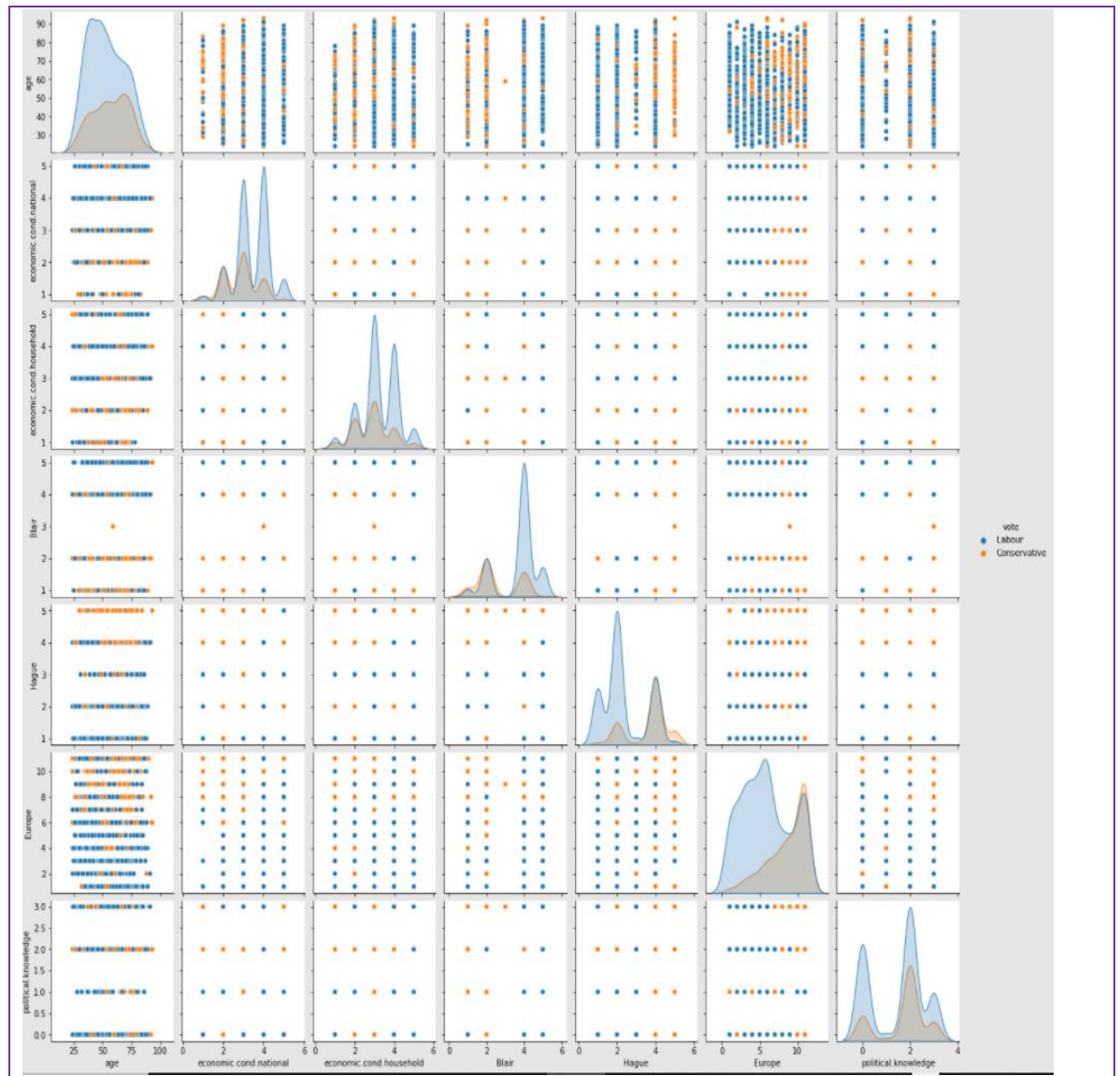


Figure-5 Pair Plot

1.2.3. Correlation analysis:

- Figure-6 demonstrates the heat map or correlation plot of variables. A heat map is a two-dimensional representation of data in which values are represented by colors. A simple heat map provides an immediate visual summary of information.
- As per figure it is observed that there is a weak correlation between the variables while mostly positive there are some negatively weak too. Most of the values in the below figure are lesser than 0.4, hence it can be said that there is not much of an actual relationship between the variables. That is the value of one variable does not have any effect on another variable. Hence concluding that they are all fairly independent in nature.

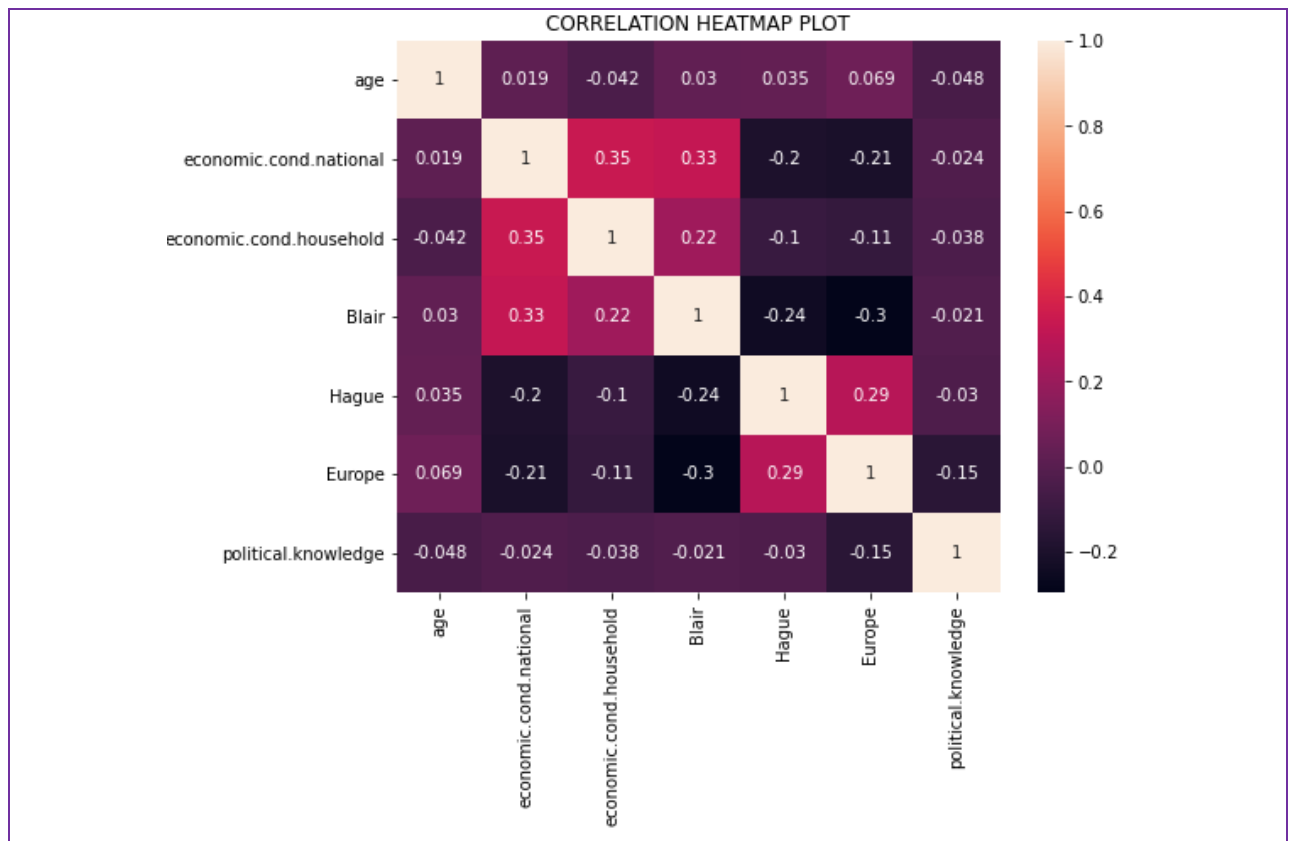


Figure-6 Heat map or Correlation plot for continuous/ordinal variables

1.2.4. Outlier Analysis:

- Data set contains outliers by plotting the box plot for all continuous/ordinal variables. Figure-7 represents outlier analysis before treatment.
- There are no outliers in age which is the only continuous variable, while there are outliers shown in 2 of the ordinal variables. But since they are ordinal in nature, a decision of not treating the outliers were taken.
- Though there are lesser number of voters that have chosen a lower rating on the assessments than the others, everyone's opinion matters and hence though during analysis it is an outlier. Hence proving they mustn't be treated.

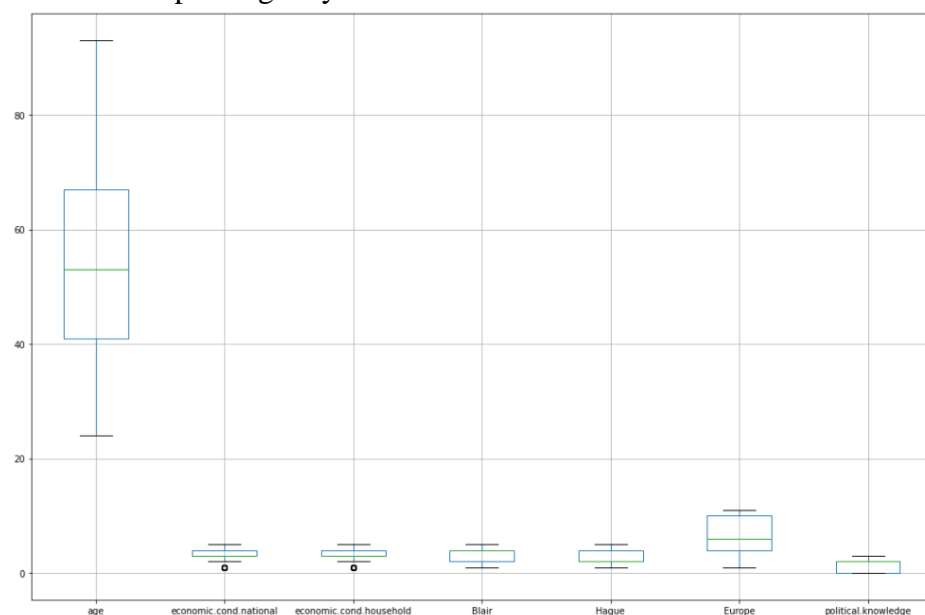


Figure-7 Outlier Analysis

1.3. Categorical Variables Treatment and Scaling

1.3.1. Encoding of the Variables:

Table-2: Encoding of Categorical variables

No. of Unique Categorical Values:

```
VOTE : 2
Conservative    462
Labour          1063
Name: vote, dtype: int64

GENDER : 2
male        713
female      812
Name: gender, dtype: int64
```

Output of encoded data:

- **feature: vote**
['Labour', 'Conservative']
Categories (2, object): ['Conservative', 'Labour']
[1 0]
- **feature: gender**
['female', 'male']
Categories (2, object): ['female', 'male']
[0 1]

Head of Encoded dataset:

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

Data Types after Encoding:

```
vote                int8
age                 int64
economic.cond.national  int64
economic.cond.household int64
Blair               int64
Hague              int64
Europe             int64
political.knowledge int64
gender             int8
dtype: object
```

Summary:

- ✓ Unique sub-categories were counted in each categorical variable of vote and gender and presented in the Table.
- ✓ As per the table, both gender and vote have only 2 sub-categories.
- ✓ It is necessary to convert these sub-categorical variables in to integer values to proceed with further analysis. Hence the data is now encoded, and unique codes for each variable can be seen.
- ✓ After encoding each variable we can now proceed with the modelling part.

1.3.2. Scaling of Variables:

- Feature scaling is a method used to normalize the range of independent variables or features of data. In data processing, it is also known as data normalization and is generally performed during the data preprocessing step.
- We need to perform Feature Scaling when we are dealing with Gradient Descent Based algorithms (Linear and Logistic Regression) and Distance-based algorithms (KNN, K-means) as these are very sensitive to the range of the data points.
- Scaling is not mandatory for LDA and Naive Bayes. But if we decide to scale the data it doesn't matter. Since these modelling techniques are not affected by feature scaling.
- Decision trees and Tree-based ensemble methods (RF, XGB) are invariant to feature scaling, but still, it might be a good idea to rescale/standardize your data.
- Hence all the models have been built after scaling/Standardizing the data using the zscore technique.

1.3.3. Data Split:

- Data split was performed with 70:30 ratio of Train and Test data using defined random state.
- Total of 8 independent variable were present in the X data frame whereas Y contains the dependent variable of vote/choice of party.

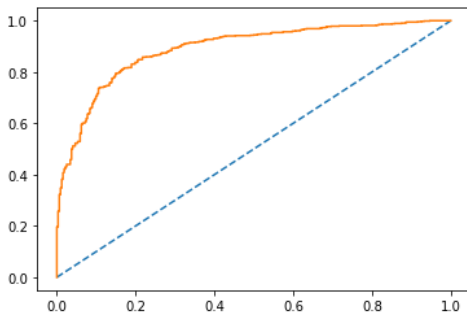
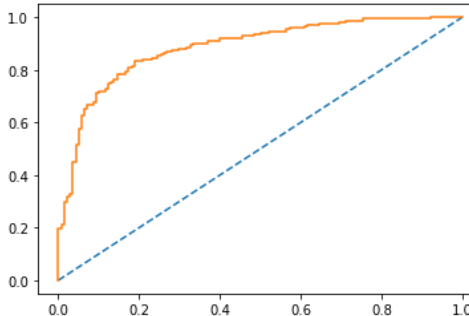
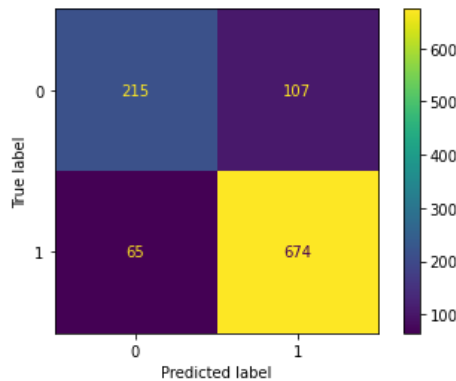
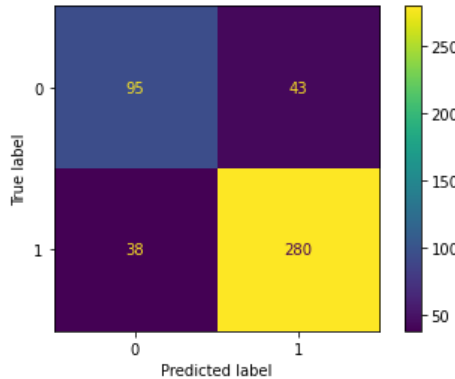
1.4. Logistic Regression Analysis vs. LDA

- LDA works when all the independent/predictor variables are continuous (not categorical) and follow a Normal distribution. Whereas in Logistic Regression this is not the case and categorical variables can be used as independent variables while making predictions.
- Logistic Regression fit applied for X Train and Y Train data set using sklearn Logistic Regression model function.
- Also, Linear Discriminant Analysis performed for X Train and Y Train data.

1.4.1. LR Models Performance and Inference:

- As per Logistic Regression analysis the following summary metrics are presented.
- The model is tuned using a param grid and the best parameter is chosen from the said grid and applied to the model.
- Logistic regression does not really have any critical hyper parameters to tune. Sometimes, you can see useful differences in performance or convergence with different solvers (solver). The 'newton-cg', 'sag', and 'lbfgs' solvers support only L2 regularization with primal formulation, or no regularization. The 'liblinear' solver supports both L1 and L2 regularization, with a dual formulation only for the L2 penalty. For multiclass problems, only 'newton-cg', 'sag', 'saga' and 'lbfgs' handle multinomial loss. Chosen solver is 'sag'.
- Penalty: A regression model that uses L1 regularization technique is called Lasso Regression and model which uses L2 is called Ridge Regression. The key difference between these two is the penalty term. Ridge regression adds "squared magnitude" of coefficient as penalty term to the loss function. Chosen penalty 'l2'
- Tolerance: Is the stopping criteria. This tells scikit to stop searching for a minimum (or maximum) once some tolerance is achieved, i.e. once you're close enough.
- Maximum number of iterations taken for the solvers to converge.

Table-3 Logistic Regression

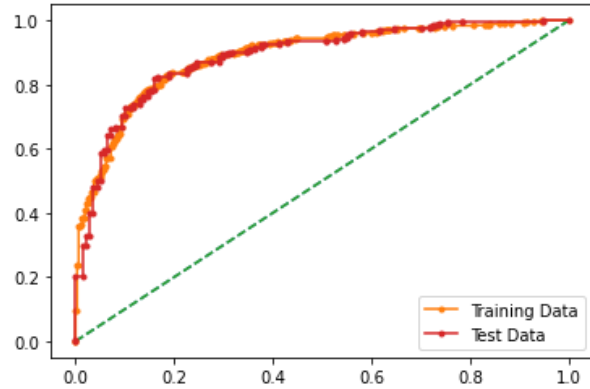
| Train Data Set | Test Data Set | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--|---|-----------|----------|----------|---------|---|------|------|------|-----|---|------|------|------|-----|----------|--|--|------|------|-----------|------|------|------|------|--------------|------|------|------|------|---|--|-----------|--------|----------|---------|---|------|------|------|-----|---|------|------|------|-----|----------|--|--|------|-----|-----------|------|------|------|-----|--------------|------|------|------|-----|
| AUC: 0.890 | AUC: 0.885 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|  |  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Model Score: 0.8378887841658812 | Model Score: 0.8223684210526315 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Confusion Matrix: | Confusion Matrix: | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|  |  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Classification Report: | Classification Report: | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <table><tr><td></td><td>precision</td><td>recall</td><td>f1-score</td><td>support</td></tr><tr><td>0</td><td>0.77</td><td>0.67</td><td>0.71</td><td>322</td></tr><tr><td>1</td><td>0.86</td><td>0.91</td><td>0.89</td><td>739</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.84</td><td>1061</td></tr><tr><td>macro avg</td><td>0.82</td><td>0.79</td><td>0.80</td><td>1061</td></tr><tr><td>weighted avg</td><td>0.83</td><td>0.84</td><td>0.83</td><td>1061</td></tr></table> | | precision | recall | f1-score | support | 0 | 0.77 | 0.67 | 0.71 | 322 | 1 | 0.86 | 0.91 | 0.89 | 739 | accuracy | | | 0.84 | 1061 | macro avg | 0.82 | 0.79 | 0.80 | 1061 | weighted avg | 0.83 | 0.84 | 0.83 | 1061 | <table><tr><td></td><td>precision</td><td>recall</td><td>f1-score</td><td>support</td></tr><tr><td>0</td><td>0.71</td><td>0.69</td><td>0.70</td><td>138</td></tr><tr><td>1</td><td>0.87</td><td>0.88</td><td>0.87</td><td>318</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.82</td><td>456</td></tr><tr><td>macro avg</td><td>0.79</td><td>0.78</td><td>0.79</td><td>456</td></tr><tr><td>weighted avg</td><td>0.82</td><td>0.82</td><td>0.82</td><td>456</td></tr></table> | | precision | recall | f1-score | support | 0 | 0.71 | 0.69 | 0.70 | 138 | 1 | 0.87 | 0.88 | 0.87 | 318 | accuracy | | | 0.82 | 456 | macro avg | 0.79 | 0.78 | 0.79 | 456 | weighted avg | 0.82 | 0.82 | 0.82 | 456 |
| | precision | recall | f1-score | support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0.77 | 0.67 | 0.71 | 322 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0.86 | 0.91 | 0.89 | 739 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| accuracy | | | 0.84 | 1061 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| macro avg | 0.82 | 0.79 | 0.80 | 1061 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| weighted avg | 0.83 | 0.84 | 0.83 | 1061 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | precision | recall | f1-score | support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0.71 | 0.69 | 0.70 | 138 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0.87 | 0.88 | 0.87 | 318 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| accuracy | | | 0.82 | 456 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| macro avg | 0.79 | 0.78 | 0.79 | 456 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| weighted avg | 0.82 | 0.82 | 0.82 | 456 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

- In general, a model fits the data well if the differences between the observed values and the model's predicted values are small and unbiased. The train and test model scores are not too far apart from one another hence this is a good fit model.
- Accuracy might not be best possible metric based on which we can take a decision, so we need to be aware of other metrics such as precision, recall, f1-score, which becomes more relevant to choose the best model.
- Precision is the fraction of true positive examples among the examples that the model classified as positive. In other words, the number of true positives divided by the number of false positives plus true positives.
- Recall, also known as sensitivity, is the fraction of examples classified as positive, among the total number of positive examples. In other words, the number of true positives divided by the number of true positives plus false negatives.
- When both the recall and precision values are important we look at the F1-score, as it is the harmonic mean of precision and recall. It combines precision and recall into a single number. It is a measure of the models accuracy on the dataset. These scores are closer to 1 hence stating that they have a good accuracy.

1.4.2: LDA Models Performance and Inference

- As per Linear Discriminant Analysis the following summary metrics are presented.
- The model is tuned using a param grid and the best parameter is chosen from the said grid and applied to the model.
- Solvers: chosen 'lsqr' by the grid
 - a) 'svd': Singular value decomposition (default). Does not compute the covariance matrix, therefore this solver is recommended for data with a large number of features.
 - b) 'lsqr': Least squares solution. Can be combined with shrinkage or custom covariance estimator.
 - c) 'eigen': Eigenvalue decomposition. Can be combined with shrinkage or custom covariance estimator.
- Shrinkage: This should be left to None if covariance_estimator is used. Note that shrinkage works only with 'lsqr' and 'eigen' solvers. Set to 'auto'.
- Tol: Absolute threshold for a singular value of X to be considered significant, used to estimate the rank of X.

Table-4 Linear Discriminant Analysis

| Train Data Set | | | | Test Data Set | | | | | | | | | | | | | | | | | | | | | |
|---|-----------|--------|----------|--|--------------|-----------|--------|----------|---------|---|---|-----|-----|---|----|-----|--|---|---|---|----|----|---|----|-----|
| AUC: 0.890 | | | | AUC: 0.888 | | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | | | | | | |
| Model Score: 0.8331762488218661 | | | | Model Score: 0.831140350877193 | | | | | | | | | | | | | | | | | | | | | |
| Confusion Matrix: | | | | | | | | | | | | | | | | | | | | | | | | | |
| <div><div><p>Training Data</p><table><tr><td></td><td>0</td><td>1</td></tr><tr><td>0</td><td>219</td><td>103</td></tr><tr><td>1</td><td>74</td><td>665</td></tr></table></div><div><p>Test Data</p><table><tr><td></td><td>0</td><td>1</td></tr><tr><td>0</td><td>98</td><td>40</td></tr><tr><td>1</td><td>37</td><td>281</td></tr></table></div></div> | | | | | | | | | 0 | 1 | 0 | 219 | 103 | 1 | 74 | 665 | | 0 | 1 | 0 | 98 | 40 | 1 | 37 | 281 |
| | 0 | 1 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 219 | 103 | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 74 | 665 | | | | | | | | | | | | | | | | | | | | | | | |
| | 0 | 1 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 98 | 40 | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 37 | 281 | | | | | | | | | | | | | | | | | | | | | | | |
| Classification Report: Classification Report of the training data: | | | | Classification Report: Classification Report of the test data: | | | | | | | | | | | | | | | | | | | | | |
| | precision | recall | f1-score | support | | precision | recall | f1-score | support | | | | | | | | | | | | | | | | |
| 0 | 0.75 | 0.68 | 0.71 | 322 | 0 | 0.73 | 0.71 | 0.72 | 138 | | | | | | | | | | | | | | | | |
| 1 | 0.87 | 0.90 | 0.88 | 739 | 1 | 0.88 | 0.88 | 0.88 | 318 | | | | | | | | | | | | | | | | |
| accuracy | | | 0.83 | 1061 | accuracy | | | 0.83 | 456 | | | | | | | | | | | | | | | | |
| macro avg | 0.81 | 0.79 | 0.80 | 1061 | macro avg | 0.80 | 0.80 | 0.80 | 456 | | | | | | | | | | | | | | | | |
| weighted avg | 0.83 | 0.83 | 0.83 | 1061 | weighted avg | 0.83 | 0.83 | 0.83 | 456 | | | | | | | | | | | | | | | | |

- The train and test model scores are not too far apart from one another hence this is a good fit model.
- Accuracy might not be best possible metric based on which we can take a decision, so we need to be aware of other metrics such as precision, recall, f1-score, which becomes more relevant to choose the best model.
- When both the recall and precision values are important we look at the F1-score, as it is the harmonic mean of precision and recall. It combines precision and recall into a single number. It is a measure of the models accuracy on the dataset. These scores are closer to 1 hence stating that they have a good accuracy.

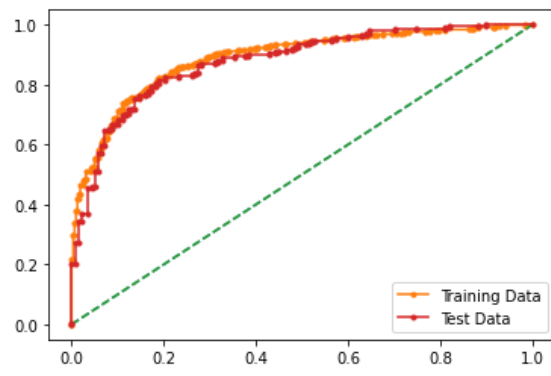
1.5. KNN Model and Naïve Bayes Model

- A general difference between KNN and other models is the large real time computation needed by KNN compared to others. KNN vs. Naive Bayes: Naive Bayes is much faster than KNN due to KNN's real-time execution.
- Gaussian Naïve Bayes fit is applied for X Train and Y Train data set using sklearn GaussianNB model function.
- KNN Analysis is also performed for X Train and Y Train data.

1.5.1. Naïve Bayes Models Performance and Inference:

- As per Naïve Bayes Analysis the following summary metrics are presented.
- Hyper-parameter tuning is not a valid method to improve Naive Bayes classifier accuracy.
- Like all machine learning algorithms, we can boost the Naive Bayes classifier by applying some simple techniques to the dataset, like data pre-processing and feature selection.
- Hence we went on to modelling with the default parameters set.

Table-5 Naïve Bayes

| Train Data Set | | Test Data Set | | | | | | | | | | | | | | | | | | | |
|---|-----|---------------------------------|--|---|-----|----|---|----|-----|--|---|---|---|-----|----|---|----|-----|--|---|---|
| AUC: 0.888 | | AUC: 0.877 | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | | |
| Model Score: 0.8303487276154571 | | Model Score: 0.8201754385964912 | | | | | | | | | | | | | | | | | | | |
| Confusion Matrix: | | | | | | | | | | | | | | | | | | | | | |
| <div><div><div>Training Data</div><table><tr><td>0</td><td>228</td><td>94</td></tr><tr><td>1</td><td>86</td><td>653</td></tr><tr><td></td><td>0</td><td>1</td></tr></table></div><div><div>Test Data</div><table><tr><td>0</td><td>100</td><td>38</td></tr><tr><td>1</td><td>44</td><td>274</td></tr><tr><td></td><td>0</td><td>1</td></tr></table></div></div> | | | | 0 | 228 | 94 | 1 | 86 | 653 | | 0 | 1 | 0 | 100 | 38 | 1 | 44 | 274 | | 0 | 1 |
| 0 | 228 | 94 | | | | | | | | | | | | | | | | | | | |
| 1 | 86 | 653 | | | | | | | | | | | | | | | | | | | |
| | 0 | 1 | | | | | | | | | | | | | | | | | | | |
| 0 | 100 | 38 | | | | | | | | | | | | | | | | | | | |
| 1 | 44 | 274 | | | | | | | | | | | | | | | | | | | |
| | 0 | 1 | | | | | | | | | | | | | | | | | | | |

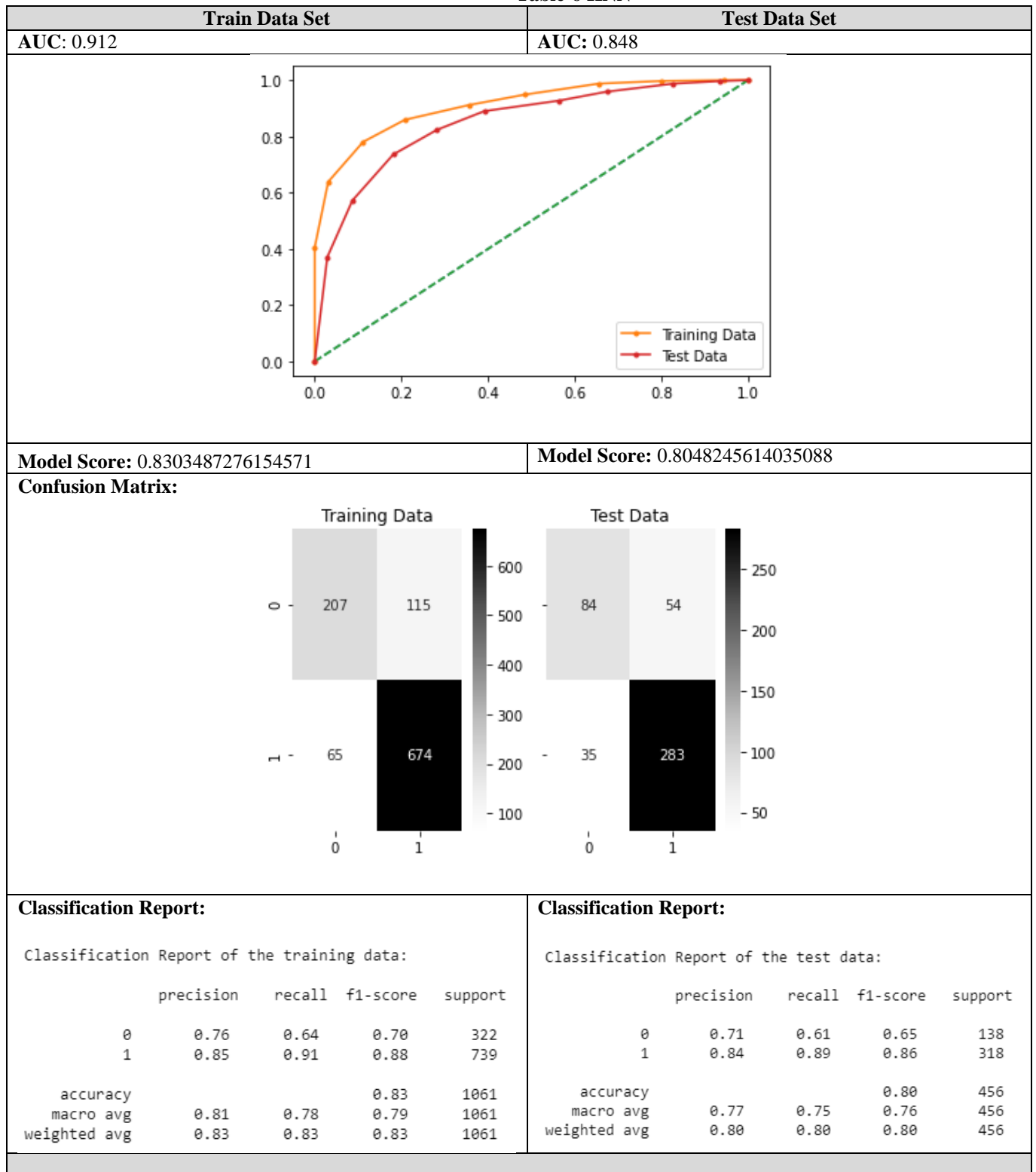
| Classification Report: | | | | | Classification Report: | | | | |
|---|-----------|--------|----------|---------|---|-----------|--------|----------|---------|
| Classification Report of the training data: | | | | | Classification Report of the test data: | | | | |
| | precision | recall | f1-score | support | | precision | recall | f1-score | support |
| 0 | 0.73 | 0.71 | 0.72 | 322 | 0 | 0.69 | 0.72 | 0.71 | 138 |
| 1 | 0.87 | 0.88 | 0.88 | 739 | 1 | 0.88 | 0.86 | 0.87 | 318 |
| accuracy | | | 0.83 | 1061 | accuracy | | | 0.82 | 456 |
| macro avg | 0.80 | 0.80 | 0.80 | 1061 | macro avg | 0.79 | 0.79 | 0.79 | 456 |
| weighted avg | 0.83 | 0.83 | 0.83 | 1061 | weighted avg | 0.82 | 0.82 | 0.82 | 456 |

- The train and test model scores are not too far apart from one another hence this is a good fit model.
- Accuracy might not be best possible metric based on which we can take a decision, so we need to be aware of other metrics such as precision, recall, f1-score, which becomes more relevant to choose the best model.
- When both the recall and precision values are important we look at the F1-score, as it is the harmonic mean of precision and recall. It combines precision and recall into a single number. It is a measure of the models accuracy on the dataset. These scores are closer to 1 hence stating that they have a good accuracy.

1.5.2. KNN Model Performance and Inference:

- As per KNN Model the following summary metrics are presented.
- The model is tuned using a param grid and the best parameter is chosen from the said grid and applied to the model.
- n_neighbors, default=5: Number of neighbours to use by default for k neighbours queries. The n_neighbors chosen from the grid is 9.
- Weights, default='uniform': Weight function used in prediction.
 - a) 'uniform' : uniform weights. All points in each neighbourhood are weighted equally.
 - b) '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.
 - c) [callable] : a user-defined function which accepts an array of distances, and returns an array of the same shape containing the weights.
- Leaf_size, default=30: This can affect the speed of the construction and query, as well as the memory required to store the tree. The optimal value depends on the nature of the problem.
- 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. We have used 'manhattan' distance as it was the best parameter chosen from the grid, it is a distance metric between two points in a N dimensional vector space.

Table-6 KNN



- The train and test model scores are not too far apart from one another hence this is a good fit model. Not an ideal good fit, it is slightly underfit.
- Accuracy might not be best possible metric based on which we can take a decision, so we need to be aware of other metrics such as precision, recall, f1-score, which becomes more relevant to choose the best model.
- When both the recall and precision values are important we look at the F1-score, as it is the harmonic mean of precision and recall. It combines precision and recall into a single number. It is a measure of the models accuracy on the dataset. These scores are closer to 1 hence stating that they have a good accuracy.

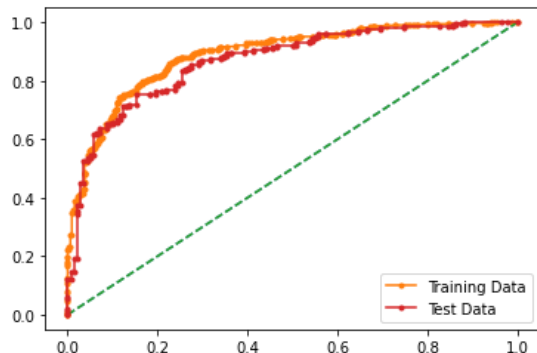
1.6. Bagging and Boosting

- Boosting is a method of merging different types of predictions. Bagging decreases variance, not bias, and solves over-fitting issues in a model. Boosting decreases bias, not variance. In Bagging, each model receives an equal weight.
- Boosting fit is applied for X Train and Y Train data set using sklearn AdaBoostClassifier model function.
- Bagging Analysis is also performed for X Train and Y Train data.

1.6.1. Bagging Performance and Inference:

- As per Bagging Model (Random Forest) the following summary metrics are presented.
- The Random Forest is tuned using a param grid and the best parameter is chosen from the said grid and applied to the Bagging Model.
- `n_estimators`, default=100: The number of trees in the forest.
- `Max_depth`, default=None: The maximum depth of the tree. If None, then nodes are expanded until all leaves are pure or until all leaves contain less than `min_samples_split` samples.
- `Min_samples_split` or float, default=2: The minimum number of samples required to split an internal node.
- `Min_samples_leaf` or float, default=1: The minimum number of samples required to be at a leaf node. A split point at any depth will only be considered if it leaves at least `min_samples_leaf` training samples in each of the left and right branches. This may have the effect of smoothing the model, especially in regression.
- `max_features`, default="auto": The number of features to consider when looking for the best split.

Table-7 Bagging

| Train Data Set | | Test Data Set | | | | | | | | | | | | | | | | | | | |
|---|-----|---------------------------------|--|---|-----|-----|---|----|-----|--|---|---|---|----|----|---|----|-----|--|---|---|
| AUC: 0.891 | | AUC: 0.869 | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | | |
| Model Score: 0.7898209236569275 | | Model Score: 0.7982456140350878 | | | | | | | | | | | | | | | | | | | |
| Confusion Matrix: | | | | | | | | | | | | | | | | | | | | | |
| <div><div><div>Training Data</div><table><tr><td>0</td><td>129</td><td>193</td></tr><tr><td>1</td><td>30</td><td>709</td></tr><tr><td></td><td>0</td><td>1</td></tr></table></div><div><div>Test Data</div><table><tr><td>0</td><td>59</td><td>79</td></tr><tr><td>1</td><td>13</td><td>305</td></tr><tr><td></td><td>0</td><td>1</td></tr></table></div></div> | | | | 0 | 129 | 193 | 1 | 30 | 709 | | 0 | 1 | 0 | 59 | 79 | 1 | 13 | 305 | | 0 | 1 |
| 0 | 129 | 193 | | | | | | | | | | | | | | | | | | | |
| 1 | 30 | 709 | | | | | | | | | | | | | | | | | | | |
| | 0 | 1 | | | | | | | | | | | | | | | | | | | |
| 0 | 59 | 79 | | | | | | | | | | | | | | | | | | | |
| 1 | 13 | 305 | | | | | | | | | | | | | | | | | | | |
| | 0 | 1 | | | | | | | | | | | | | | | | | | | |

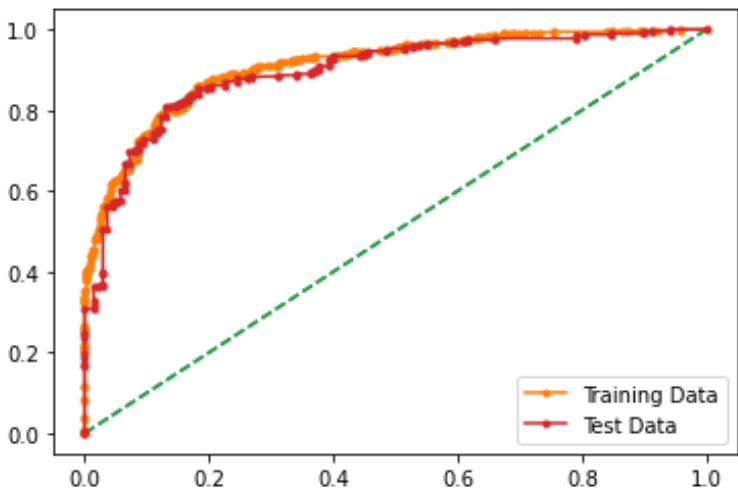
| Classification Report: | | | | | Classification Report: | | | | |
|------------------------|-----------|--------|----------|---------|------------------------|-----------|--------|----------|---------|
| | precision | recall | f1-score | support | | precision | recall | f1-score | support |
| 0 | 0.81 | 0.40 | 0.54 | 322 | 0 | 0.82 | 0.43 | 0.56 | 138 |
| 1 | 0.79 | 0.96 | 0.86 | 739 | 1 | 0.79 | 0.96 | 0.87 | 318 |
| accuracy | | | 0.79 | 1061 | accuracy | | | 0.80 | 456 |
| macro avg | 0.80 | 0.68 | 0.70 | 1061 | macro avg | 0.81 | 0.69 | 0.72 | 456 |
| weighted avg | 0.79 | 0.79 | 0.76 | 1061 | weighted avg | 0.80 | 0.80 | 0.78 | 456 |

- The train and test model scores are not too far apart from one another hence this is a good fit model.
- Accuracy might not be best possible metric based on which we can take a decision, so we need to be aware of other metrics such as precision, recall, f1-score, which becomes more relevant to choose the best model.
- When both the recall and precision values are important we look at the F1-score, as it is the harmonic mean of precision and recall. It combines precision and recall into a single number. It is a measure of the models accuracy on the dataset. These scores are closer to 1 hence stating that they have a good accuracy.

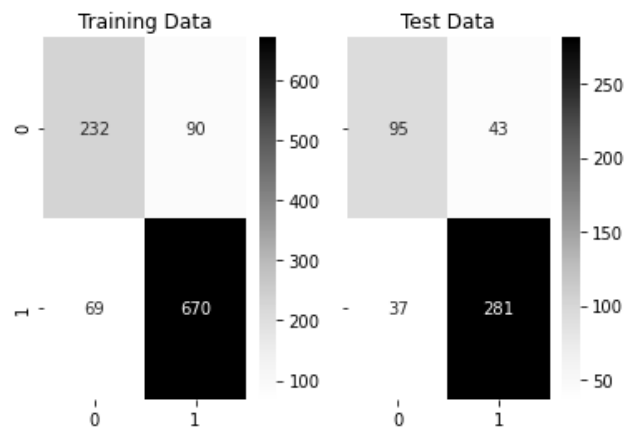
1.6.2. Boosting Performance and Inference:

- As per Boosting Model the following summary metrics are presented. There have been two types of Boosting performed in the Jupyter Notebook, the better of both have been selected and showcased here.
- We have chosen to go with the Adaptive Boosting model, since the values seem closer and the model represents a good fit in comparison with the Gradient Boosting Model.
- Base_estimator, default=None: The base estimator from which the boosted ensemble is built. If None, then the base estimator is DecisionTreeClassifier initialized with max_depth=1.
- n_estimators, default=50: The maximum number of estimators at which boosting is terminated. In case of perfect fit, the learning procedure is stopped early.
- Learning_rate, default=1.0: Weight applied to each classifier at each boosting iteration. A higher learning rate increases the contribution of each classifier. There is a trade-off between the learning_rate and n_estimators parameters.

Table-8 Boosting

| Train Data Set | Test Data Set |
|--|---------------------------------|
| AUC: 0.911 | AUC: 0.897 |
|  | |
| Model Score: 0.8501413760603205 | Model Score: 0.8245614035087719 |

Confusion Matrix:



Classification Report:

| | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 0 | 0.77 | 0.72 | 0.74 | 322 |
| 1 | 0.88 | 0.91 | 0.89 | 739 |
| accuracy | | | 0.85 | 1061 |
| macro avg | 0.83 | 0.81 | 0.82 | 1061 |
| weighted avg | 0.85 | 0.85 | 0.85 | 1061 |

Classification Report:

| | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 0 | 0.72 | 0.69 | 0.70 | 138 |
| 1 | 0.87 | 0.88 | 0.88 | 318 |
| accuracy | | | 0.82 | 456 |
| macro avg | 0.79 | 0.79 | 0.79 | 456 |
| weighted avg | 0.82 | 0.82 | 0.82 | 456 |

- The train and test model scores are not too far apart from one another hence this is a good fit model.
- Accuracy might not be best possible metric based on which we can take a decision, so we need to be aware of other metrics such as precision, recall, f1-score, which becomes more relevant to choose the best model.
- When both the recall and precision values are important we look at the F1-score, as it is the harmonic mean of precision and recall. It combines precision and recall into a single number. It is a measure of the models accuracy on the dataset. These scores are closer to 1 hence stating that they have a good accuracy.

1.7. Models Performance and Inference

- All the models were compared w.r.t AUC and model scores/Accuracy values for both Train and Test data (refer below table).
- The LDA model seems to be the best fit of the lot, the train and the test values seem closest for this model type. In general, a model fits the data well if the differences between the observed values and the model's predicted values are small and unbiased. The train and test model scores are not too far apart from one another hence this is a good fit model.
- Accuracy might not be best possible metric based on which we can take a decision, so we need to be aware of other metrics such as precision, recall, f1-score, which becomes more relevant to choose the best model.
- LDA performs better than the other model on the basis of Accuracy, AUC and Recall.
- Recall, also known as sensitivity, is the fraction of examples classified as positive, among the total number of positive examples. In other words, the number of true positives divided by the number of true positives plus false negatives.
- Precision is the fraction of true positive examples among the examples that the model classified as positive. In other words, the number of true positives divided by the number of false positives plus true positives.
- When both the recall and precision values are important we look at the F1-score, as it is the harmonic mean of precision and recall. It combines precision and recall into a single number. It is a measure of the models accuracy on the dataset. These scores are closer to 1 hence stating that they have a good accuracy.
- The Precision and F1 Score seem to be similar for all, hence that cannot be used as a distinguishing factor.
- LDA performs better than the other model on the basis of Accuracy, AUC and Recall.
- The Precision and F1 Score seem to be similar for all, hence that cannot be used as a distinguishing factor.
- Hence considering all the models, LDA is the better model on prediction of overall win and seats covered by a particular party.

Table 9- Comparison between different models

| | Logistic Train | Logistic Test | LDA Train | LDA Test | GNB Train | GNB Test | KNN Train | KNN Test | Bagging Train | Bagging Test | Boosting Train | Boosting Test |
|-----------|-------------------|------------------|--------------|-------------|--------------|-------------|--------------|-------------|------------------|-----------------|-------------------|------------------|
| Accuracy | 0.84 | 0.82 | 0.83 | 0.83 | 0.83 | 0.82 | 0.83 | 0.80 | 0.79 | 0.80 | 0.85 | 0.82 |
| AUC | 0.890 | 0.885 | 0.890 | 0.888 | 0.888 | 0.877 | 0.912 | 0.848 | 0.891 | 0.869 | 0.911 | 0.897 |
| Recall | 0.91 | 0.88 | 0.90 | 0.88 | 0.88 | 0.86 | 0.91 | 0.89 | 0.96 | 0.96 | 0.91 | 0.88 |
| Precision | 0.86 | 0.87 | 0.87 | 0.88 | 0.87 | 0.88 | 0.85 | 0.84 | 0.79 | 0.79 | 0.88 | 0.87 |
| F1 Score | 0.89 | 0.87 | 0.88 | 0.88 | 0.88 | 0.87 | 0.88 | 0.86 | 0.86 | 0.87 | 0.89 | 0.88 |

1.8. Insights and Recommendations

- Data set contains total of 1525 entries among which 7 integer type variables and 2 object type variables. Out of which 8 are independent variables and one dependent variable. Duplicates were verified, 8 duplicate rows were present in the data set which were removed. There are no null values in the dataset.
- The Target Variable is Vote and through visual representation it can clearly be seen that the voters have preferred to choose a Labour leader. There have also been more female voters than male.
- It is observed that there is a weak correlation between the variables while mostly positive there are some negatively weak too. Most of the values in the below figure are lesser than 0.4, hence it can be said that there is not much of an actual relationship between the variables. That is the value of one variable does not have any effect on another variable. Hence concluding that they are all fairly independent in nature.
- There are no outliers in age which is the only continuous variable, while there are outliers shown in 2 of the ordinal variables. But since they are ordinal in nature, a decision of not treating the outliers were taken. Though there are lesser number of voters that have chosen a lower rating on the assessments than the others, everyone's opinion matters though during analysis it is an outlier.
- The LDA model seems to be the best fit of the lot, the train and the test values seem closest for this model type. In general, a model fits the data well if the differences between the observed values and the model's predicted values are small and unbiased. The train and test model scores are not too far apart from one another hence this is a good fit model.
- LDA performs better than the other model on the basis of Accuracy, AUC and Recall. The Precision and F1 Score seem to be similar for all, hence that cannot be used as a distinguishing factor.

2. Problem-2: Text Mining and Analysis

2.1. Objective

- The objective the problem is to extract machine-readable facts from the 3 speeches of the Presidents of the United States of America.
- The purpose of Text Analysis is to create structured data out of free text content. The process can be thought of as slicing and dicing heaps of unstructured, heterogeneous documents into easy-to-manage and interpret data pieces.

2.2. Background

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

President Franklin D. Roosevelt in 1941

President John F. Kennedy in 1961

President Richard Nixon in 1973

2.3. Analysis Methodology

- We firstly imported the necessary libraries for Text Mining and Analysis. The Natural Language Toolkit, or more commonly NLTK, is a suite of libraries and programs for symbolic and statistical natural language processing for English written in the Python programming language.
- We download the keyword given in the question ie: Inaugural and extract the 3 speeches using the `raw()` function.
- We then went to check on the unique words and sentences that can be found in all the 3 speeches.
- The following is a table with the length of characters, words and sentences in each speech:

Table 10: Length of words and sentences

| | Franklin D. Roosevelt | John F. Kennedy | Richard Nixon |
|------------------|-----------------------|-----------------|---------------|
| Characters (raw) | 7571 | 7618 | 9991 |
| Words | 1536 | 1546 | 2028 |
| Sentences | 68 | 52 | 69 |

- We then removed all the stop words from each of the speeches. We then proceeded to stem and remove punctuations from the speeches. There were a total of 211 stop words which was removed from all the speeches.

Table 11: Length of words after removing stop words and punctuation

| | Franklin D. Roosevelt | John F. Kennedy | Richard Nixon |
|------------------|-----------------------|-----------------|---------------|
| Characters (raw) | 4590 | 4771 | 5950 |
| Words (split) | 627 | 693 | 833 |

- Moving on to finding out the most common words(top 3):

Table 12: Three most common words

| Franklin D. Roosevelt | John F. Kennedy | Richard Nixon |
|--|--|--|
| [('know', 10), ('spirit', 9), ('us', 8)] | [('us', 12), ('world', 8), ('Let', 8)] | [('us', 26), ('peace', 19), ('new', 15)] |

- ### A. Franklin D. Roosevelt

[illegible][illegible]