Machine Learning is defined as an application of artificial intelligence where available information is used through algorithms to process or assist the processing of statistical data. While Machine Learning involves concepts of automation

PROJECT ML

Towards AI

Sudheendra K

PROJECT ML

Contents

PROJECT ML

PROJECT ML

PROJECT ML

Problem 1:

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

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

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

Table 1. 1 Head of the dataset

Starting on with loading the data we could see that the first five rows of the data

• The data set had 1525 rows and 9 columns. After dropping the duplicate values, there are 1517 rows and 9 columns.

• It has 7 numerical data types and 2 categorical data types.

PROJECT ML

- *There is no null value in any column.*

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1525 entries, 1 to 1525
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: 119.1+ KB
```

Table 1. 2 Information of the dataset

- *The data set has 1525 rows and 9 columns.*
- *It has 7 numerical data types and 2 categorical data types.*

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

Table 1. 3 Description of the dataset

As the categorical type was already coded the description of the data set will be not coded in further

vote	0
age	0
economic.cond.national	0
economic.cond.household	0
Blair	0
Hague	0
Europe	0
political.knowledge	0
gender	0
dtype: int64	

Table 1. 4 Null Details

- *There is no null value in any column.*

1.2 Perform Univariate and Bivariate Analysis. Do exploratory data analysis. Check for Outliers. (7 Marks)

vote	0.858449
age	0.144621
economic.cond.national	-0.240453
economic.cond.household	-0.149552
Blair	-0.535419
Hague	0.152100
Europe	-0.135947
political.knowledge	-0.426838
gender	0.130239
dtype: float64	

Table 1. 5 Skewness of the variables

The rule of thumb of skewness is:

- *If the skewness is between -0.5 and 0.5, the data are fairly*
- *symmetrical.*
- *If the skewness is between -1 and – 0.5 or between 0.5 and 1, the data are moderately skewed.*
- *If the skewness is less than -1 or greater than 1, the data are*
- *highly skewed.*

Therefore, from the above table of data the insights could be drawn as following:

- *Here, we can see that there isn't much skewness in the data.*
- *All the values seem to be between -0.5 and 0.5.*
- *The value of 'Blair' is a little bit higher than -0.5.*
- *The data overall, is fairly symmetrical.*

Univariate Analysis

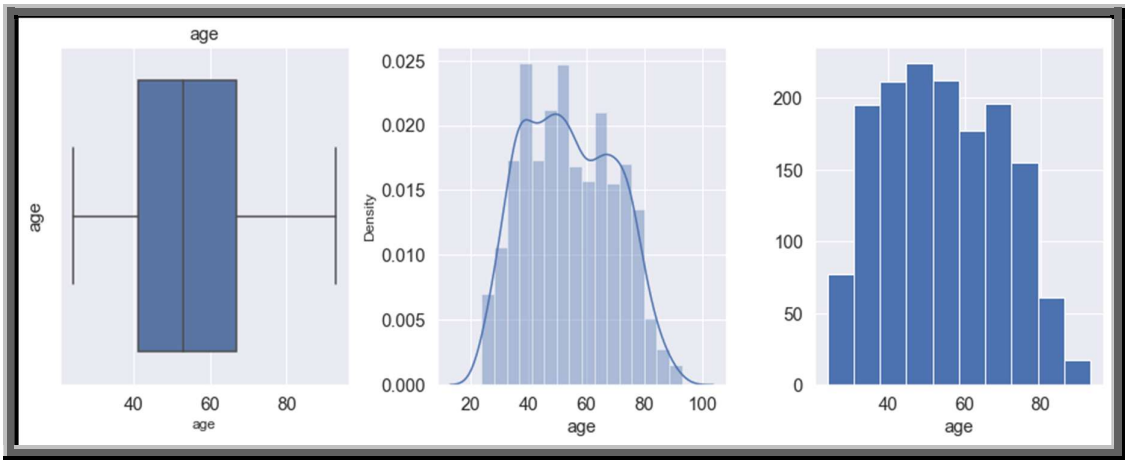


Figure 1. 1 Variable Age Plot

- *The data is normally distributed.*
- *Maximum number of people are aged between 40 and 70.*
- *Outliers are not present.*

- *The minimum value is 24 and the maximum value is 93.*
- *The mean value is 54.18*

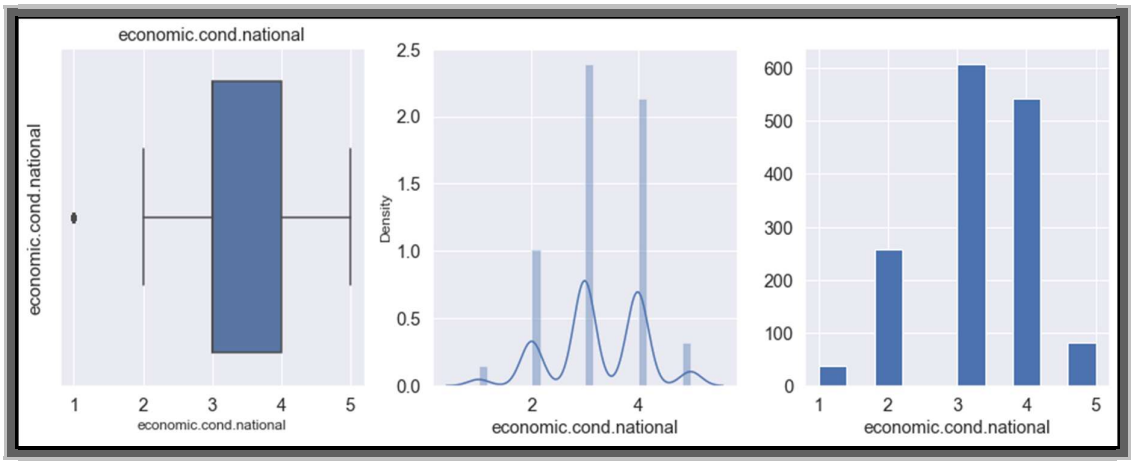


Figure 1. 2 Variable Economic.cond.national plot

- *The top 2 variables are 3 and 4.*
- *1 has the least value which is 37.*
- *3 has the highest value which is 607.*
- *3 is slightly higher than the 2nd highest variable 4 whose value is 542.*
- *The average score of 'economic.cond.national' is 3.24*

PROJECT ML

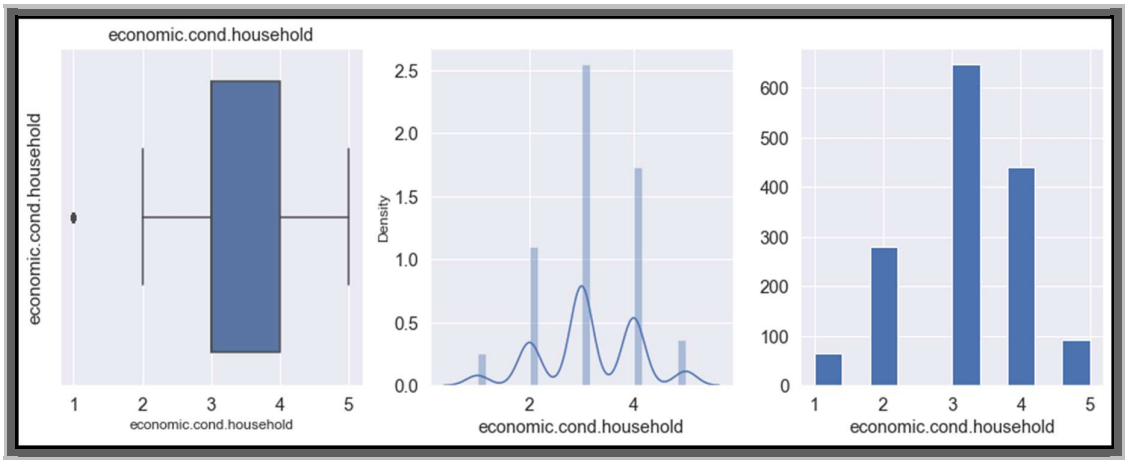


Figure 1. 3 Variable Econamic.cond.household Plot

- *The top 2 variables are 3 and 4.*
- *1 has the least value which is 65.*
- *3 has the highest value which is 648.*
- *3 is moderately higher than the 2nd highest variable 4 whose value is 440.*
- *The average score of 'economic.cond.household' is 3.14*

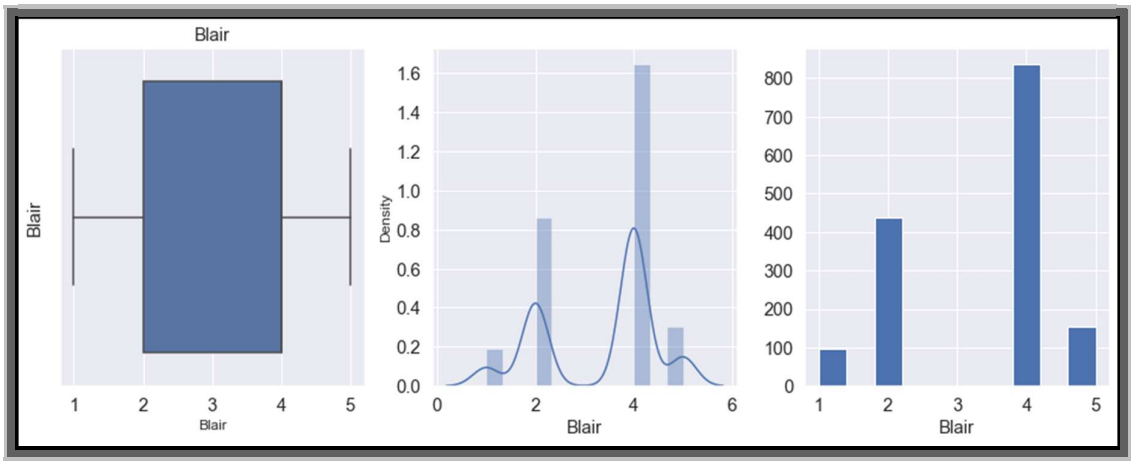


Figure 1. 4 Variable Blair Plot

PROJECT ML

- *The top 2 variables are 2 and 4.*
- *3 has the least value which is 1.*
- *4 has the highest value which is 836.*
- *4 is much higher than the 2nd highest variable 2 whose value is 438.*

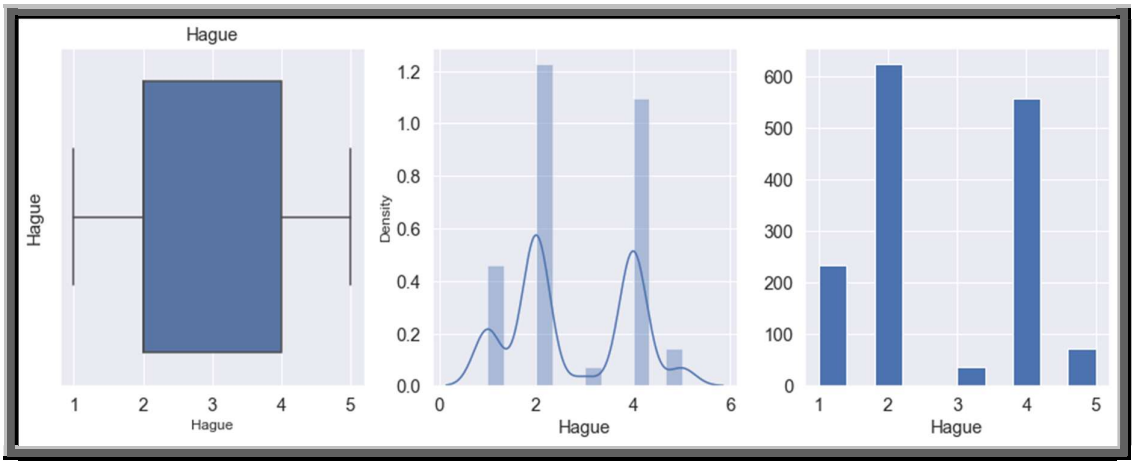


Figure 1. 5 Variable Hague Plot

- *The top 2 variables are 2 and 4.*
- *3 has the least value which is 37.*
- *2 has the highest value which is 624.*
- *2 is slightly higher than the 2nd highest variable 4 whose value is 558.*
- *The average score of 'Blair' is 2.75*

PROJECT ML

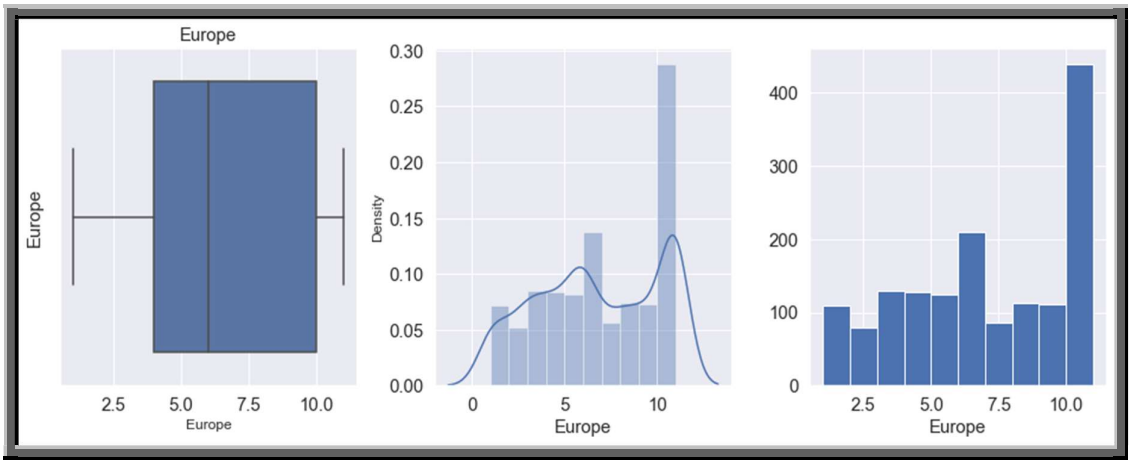


Figure 1. 6 Variable Europe Plot

- *The top 2 variables are 11 and 6.*
- *2 has the least value which is 79.*
- *11 has the highest value which is 338.*
- *11 is moderately higher than the 2nd highest variable 6 whose value is 209.*
- *The average score of 'Europe' is 6.73*

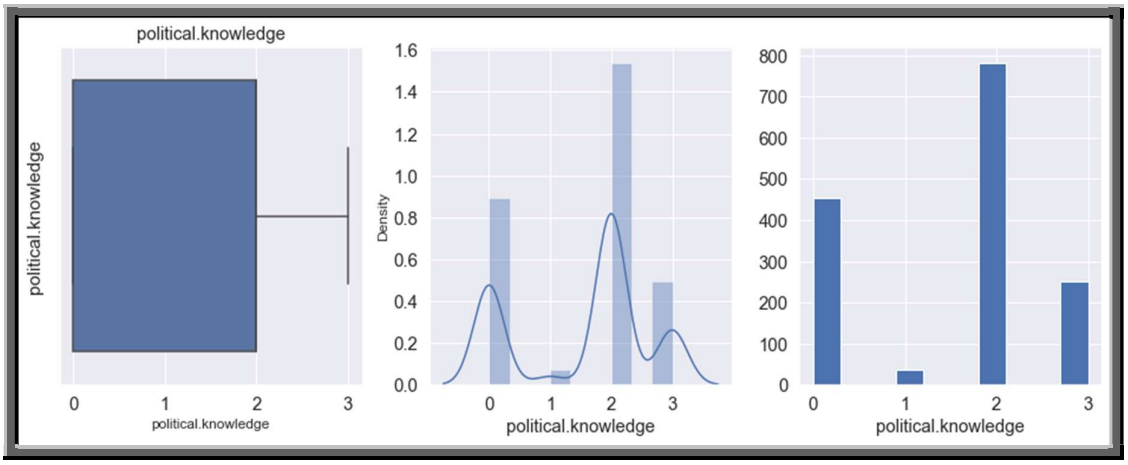
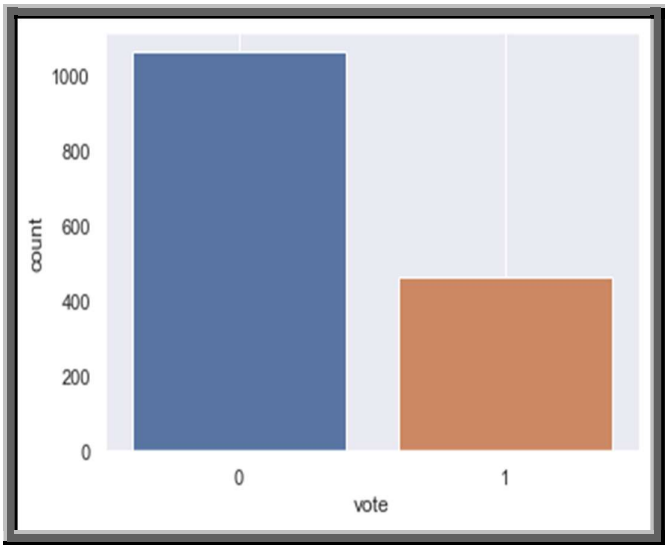


Figure 1. 7 Variable Political.knowledge Plot

- *The top 2 variables are 2 and 0.*

- *1 has the least value which is 38.*
- *2 has the highest value which is 782.*
- *2 is much higher than the 2nd highest variable 0 whose value is 455.*
- *We can see that, 455 out of 1525 people do not have any knowledge of parties' positions on European integration which is 29.93% of the total population.*
- *The average score of ' Political.knowledge ' is 1.54*

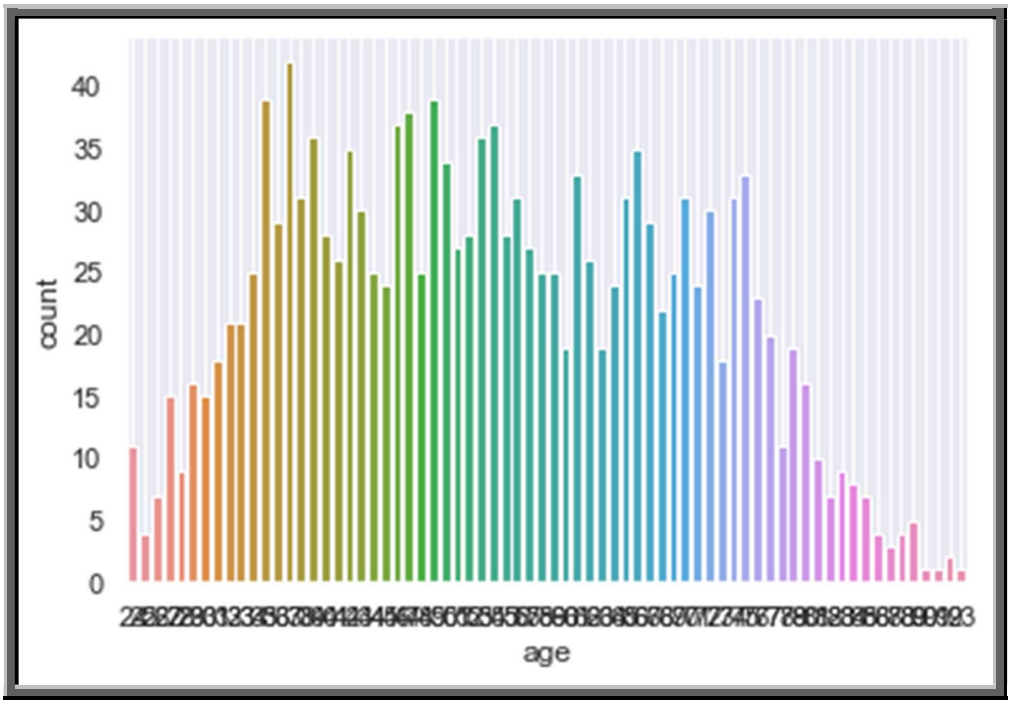



```
0    1063
1     462
Name: vote, dtype: int64
```

Figure 1. 8 Variable vote count plot

PROJECT ML

- *0 has higher number of votes. It has more than double the votes of conservative party.*
- *0 has 1063 votes.*
- *1 has 462 votes.*




```
37    42
49    39
35    39
47    38
54    37
      ..
87     3
92     2
90     1
93     1
91     1
Name: age, Length: 70, dtype: int64
```

Figure 1. 9 Variable Age count plot

- *The data is normally distributed.*
- *Maximum number of people are aged between 40 and 70.*

PROJECT ML

- *Outliers are not present.*
- *The minimum value is 24 and the maximum value is 93.*
- *The mean value is 54.18*

Bivariate analysis

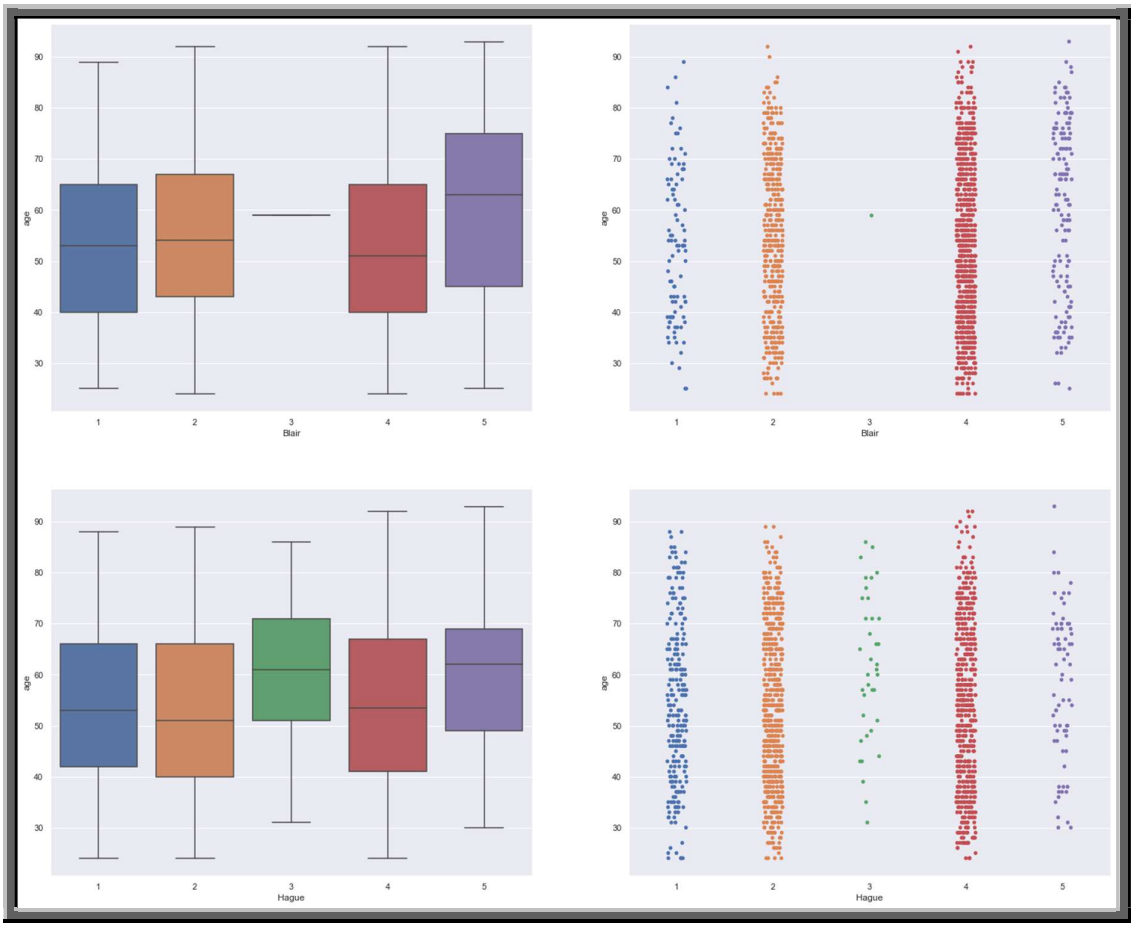


Figure 1. 10 Blair and Hague with comparison of age

- *We can clearly see that, the '0' has got more votes than the '1' party.*
- *In every age group, the '0' has got more votes than the '1' party.*

PROJECT ML

- *Female votes are considerably higher than the male votes in both parties.*
- *In both genders, the '0' has got more votes than the '1' party.*

PROJECT ML

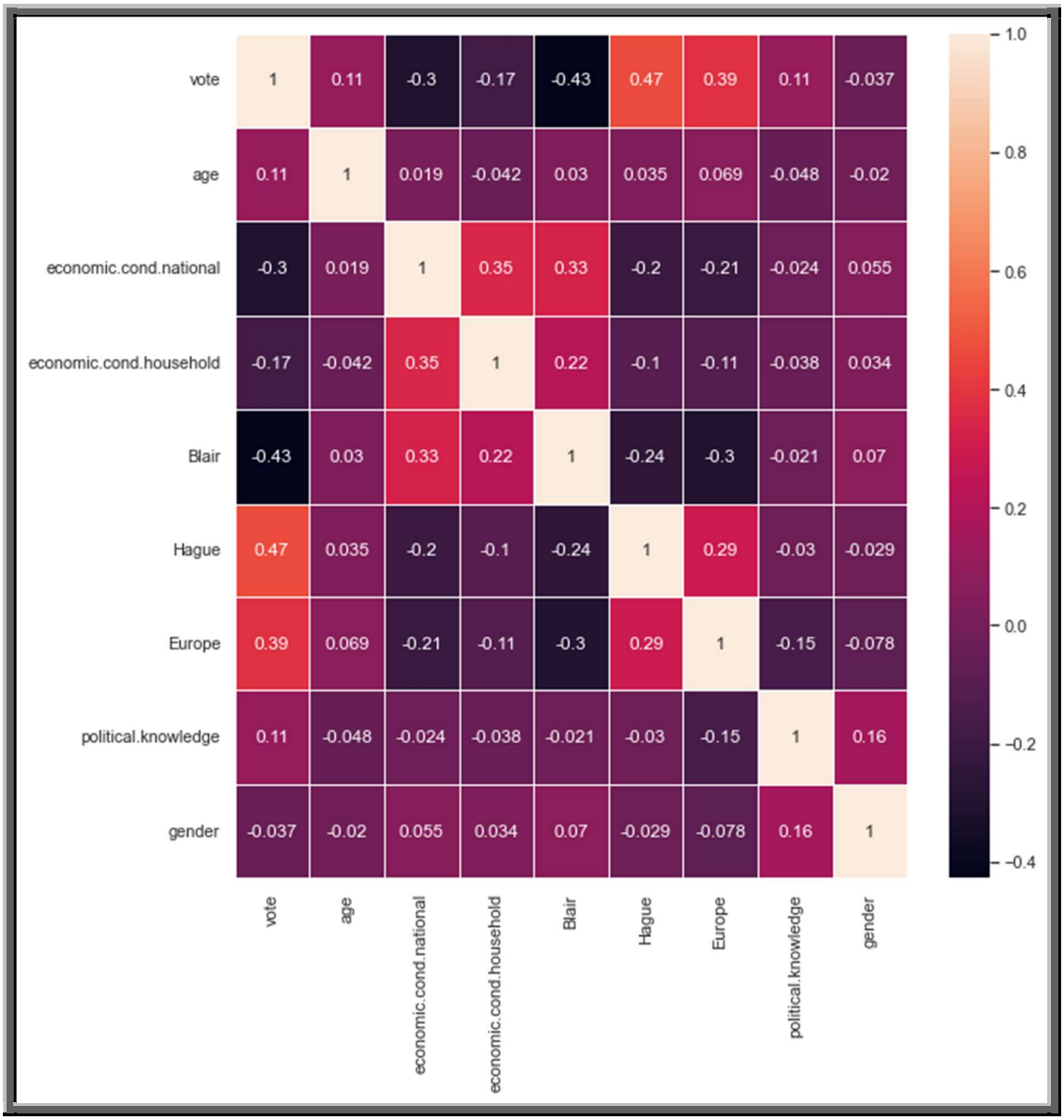


Figure 1. 11 Heat Map

- *The correlation heat map helps us to visualize the correlation between two variables.*
- *We can see that, mostly there is no correlation in the dataset through this matrix. There are some variables that are moderately positively correlated and some that are slightly negatively correlated.*

PROJECT ML

- *‘economic.cond.national’ with ‘economic.cond.household’ have moderate positive correlation.*
- *‘Blair' with 'economic.cond.national' and ‘economic.cond.household’ have moderate positive correlation.*
- *‘Europe’ with ‘Hague’ have moderate positive correlation.*
- *'Hague' with 'economic.cond.national' and 'Blair' have*
- *moderate negative correlation.*
- *'Europe' with 'economic.cond.national' and 'Blair' have moderate negative correlation.*

PROJECT ML

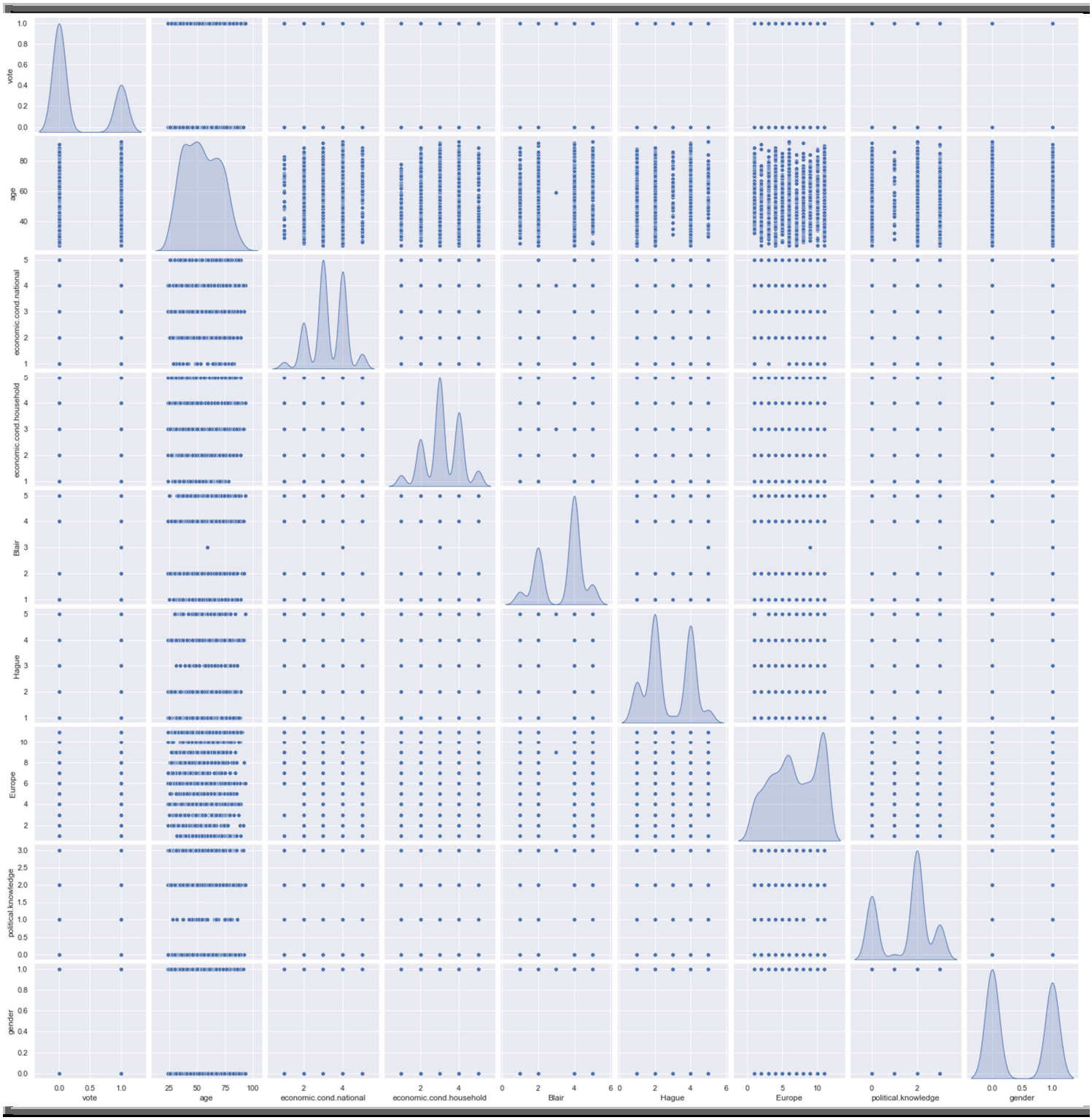


Figure 1. 12 Pair plot

PROJECT ML

- *Pair plot is a combination of histograms and scatter plots.*
- *From the histogram, we can see that, the 'Blair','Europe' and 'political.knowledge' variables are slightly left skewed.*
- *All other variables seem to be normally distributed.*
- *From the scatter plots, we can see that, there is mostly no correlation between the variables.*
- *We can use the correlation matrix to view them more clearly.*
- *Correlation matrix is a table which shows the correlation coefficient between variables. Correlation values range from -1 to +1. For values closer to zero, it means that, there is no linear trend between two variables. Values close to 1 means that the correlation is positive.*

1.3 Encode the data (having string values) for Modelling. Is Scaling necessary here or not? Data Split: Split the data into train and test (70:30). (4 Marks)

- *The dataset contains features highly varying in magnitudes, units and range between the 'age' column and other columns.*
- *But since, most of the machine learning algorithms use Euclidian distance between two data points in their computations, this is a problem.*
- *If left alone, these algorithms only take in the magnitude of features neglecting the units.*
- *The results would vary greatly between different units, 1 km and 1000 metres.*
- *The features with high magnitudes will weigh in a lot more in the distance calculations than features with low magnitudes.*
- *To supress this effect, we need to bring all features to the same level of magnitudes. This can be achieved by scaling.*

PROJECT ML

- *In this case, we have a lot of encoded, ordinal, categorical and continuous variables. So, we use the minmaxscaler technique to scale the data.*

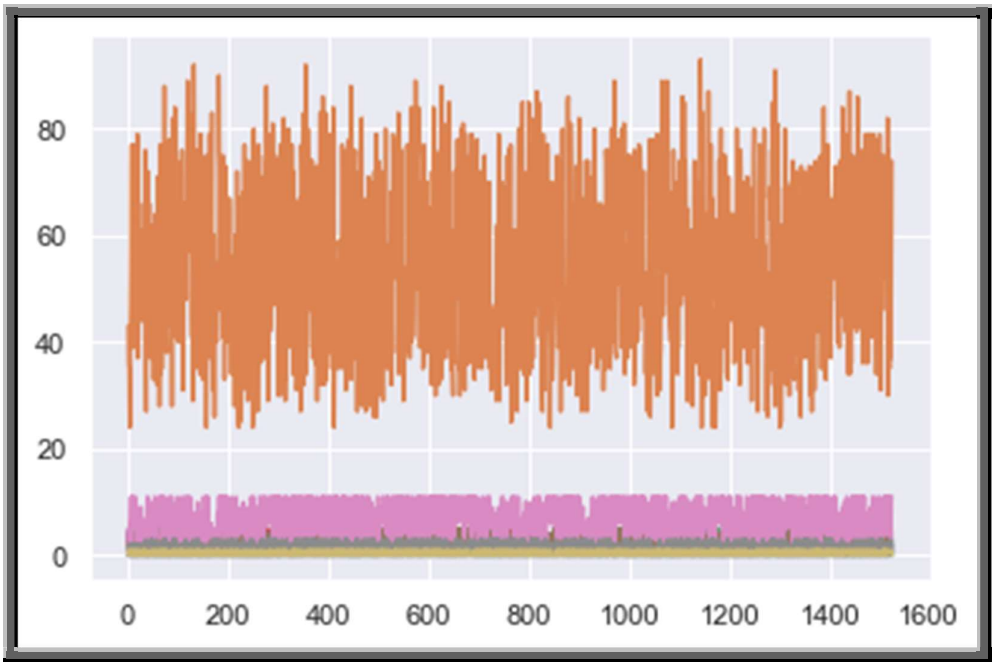


Figure 1. 13 Before Scaling

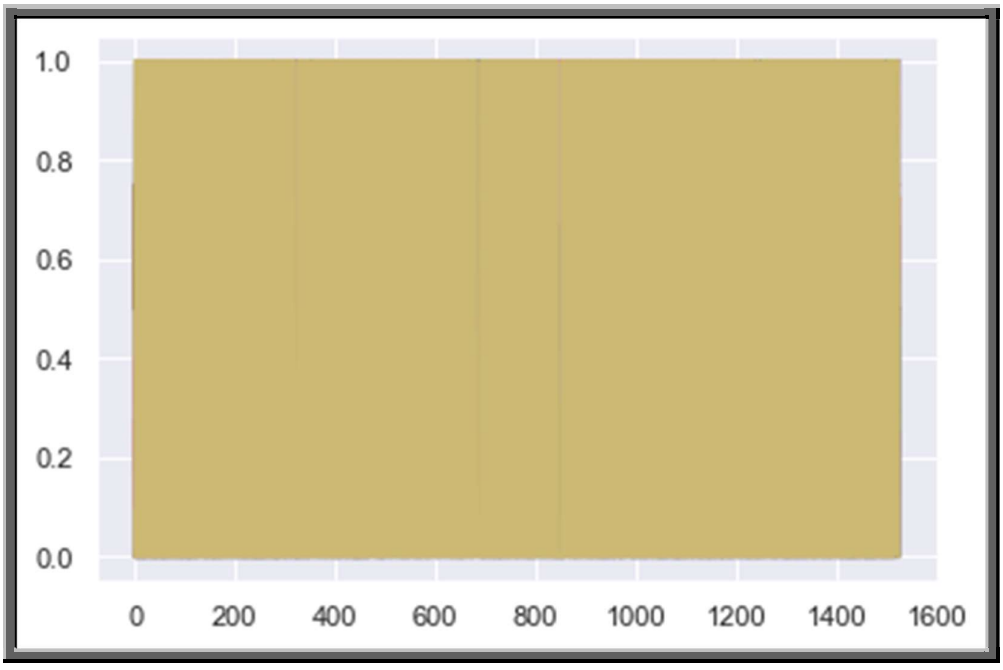


Figure 1. 14 After Scaling

PROJECT ML

Train-test-split:

- *Our model will use all the variables and 'vote_Labour' is the target variable. The train-test split is a technique for evaluating the performance of a machine learning algorithm. The procedure involves taking a dataset and dividing it into two subsets.*
- *Train Dataset: Used to fit the machine learning model.*
- *Test Dataset: Used to evaluate the fit machine learning model.*

The data is divided into 2 subsets, training and testing set. Earlier, we have extracted the target variable 'vote_Labour' in a separate vector for subsets. Random state chosen as 1.

- *Training Set: 70 percent of data.*
- *Testing Set: 30 percent of the data.*

PROJECT ML

1.4 Apply Logistic Regression(LR) and LDA (linear discriminant analysis). (4 marks)

Logistic Regression

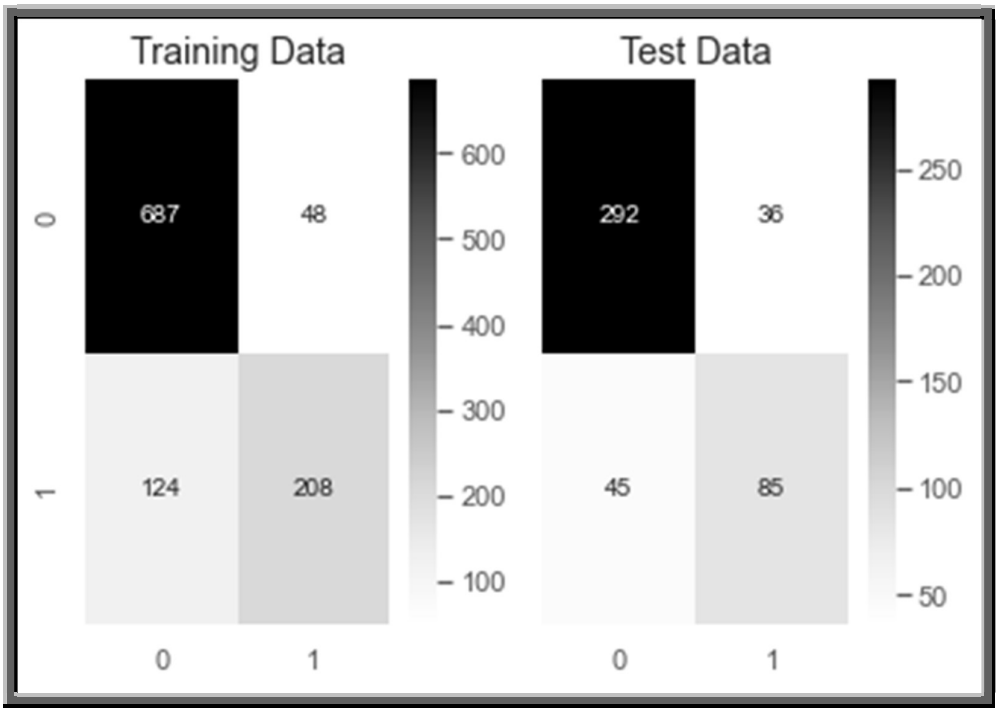


Figure 1. 15 LR Confusion Matrix

	precision	recall	f1-score	support
0	0.87	0.91	0.89	735
1	0.77	0.69	0.73	332
accuracy			0.84	1067
macro avg	0.82	0.80	0.81	1067
weighted avg	0.84	0.84	0.84	1067

Table 1. 6 LR Classification Report Training Data

Observation Train data:

- *Accuracy: 84%*
- *Precision: 77%*
- *Recall: 69%*

- *F1-Score: 73%*

	precision	recall	f1-score	support
0	0.87	0.89	0.88	328
1	0.70	0.65	0.68	130
accuracy			0.82	458
macro avg	0.78	0.77	0.78	458
weighted avg	0.82	0.82	0.82	458

Table 1. 7 LR Classification Report Testing Data

Observation Test data:

- *Accuracy: 82%*
- *Precision: 70%*
- *Recall: 65%*
- *F1-Score: 68%*

PROJECT ML

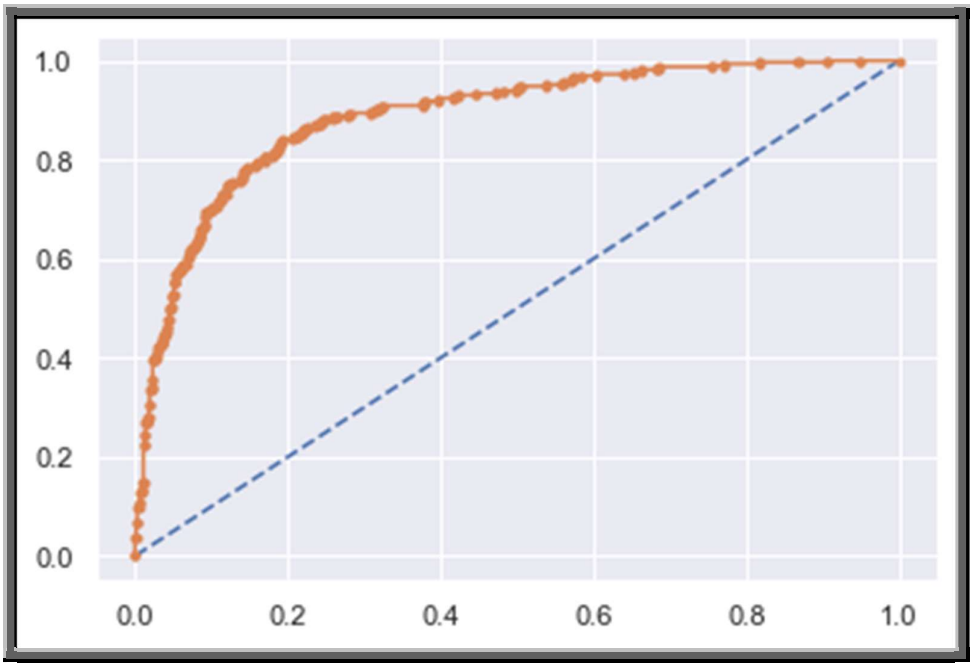


Figure 1. 16 LR AUC and ROC Training Data

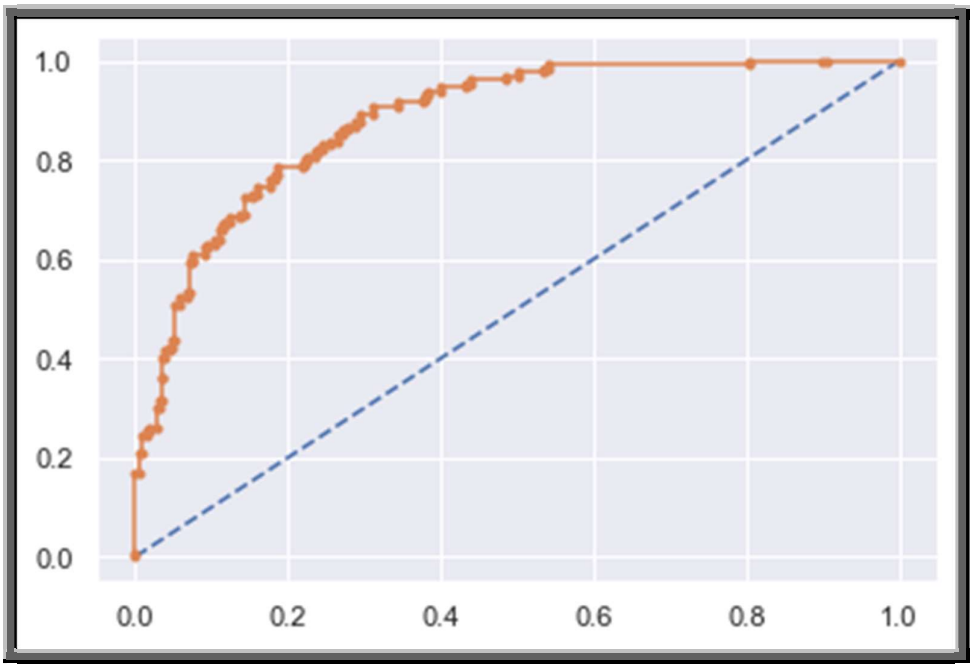


Figure 1. 17 LR AUC and ROC Testing Data

Evaluation of the model:

- *The model is neither over-fitted nor under-fitted.*

- *The error in the test data is slightly higher than the train data, which is absolutely fine because the error margin is low and the error in both train and test data is not too high. Thus, the model is not over-fitted or under-fitted.*

Linear Discriminant Analysis

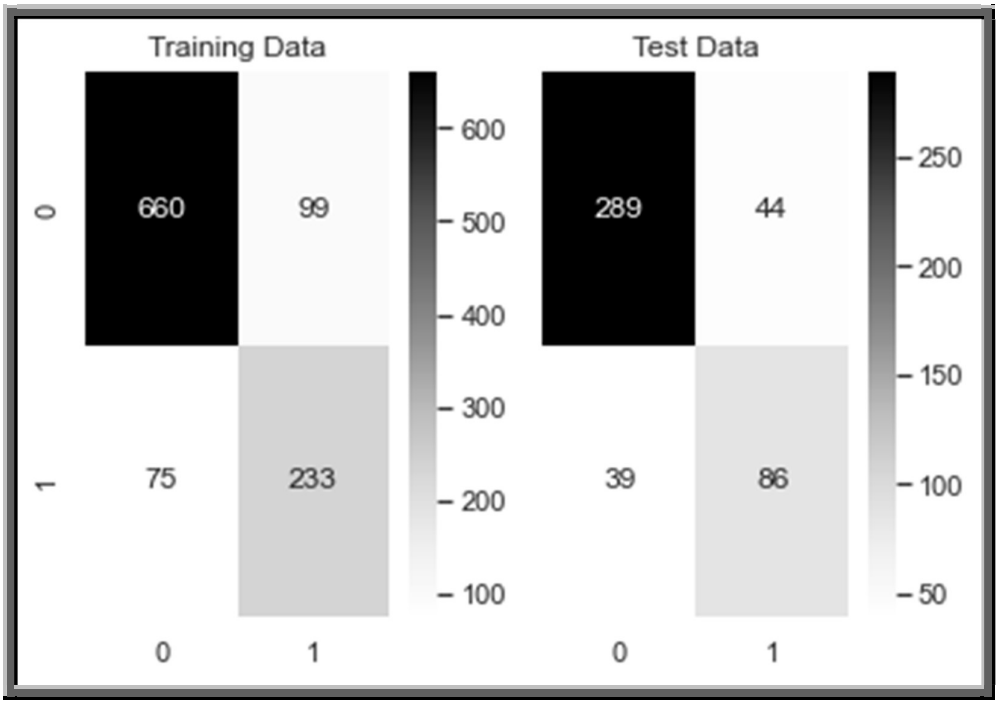


Figure 1. 18 Confusion matrix LDR

0	0.90	0.87	0.88	759
1	0.70	0.76	0.73	308
accuracy			0.84	1067
macro avg	0.80	0.81	0.81	1067
weighted avg	0.84	0.84	0.84	1067

Table 1. 8 LDR Classification report Training data

Observation Train data:

- *Accuracy: 84%*

Precision: 70%

- *Recall: 76%*
- *F1-Score:73%*

0	0.88	0.87	0.87	333
1	0.66	0.69	0.67	125
accuracy			0.82	458
macro avg	0.77	0.78	0.77	458
weighted avg	0.82	0.82	0.82	458

Table 1. 9 LDR Classification Report Testing data

Observation Test data:

- *Accuracy: 82%*
- *Precision: 66%*
- *Recall: 69%*
- *F1-Score: 67%*

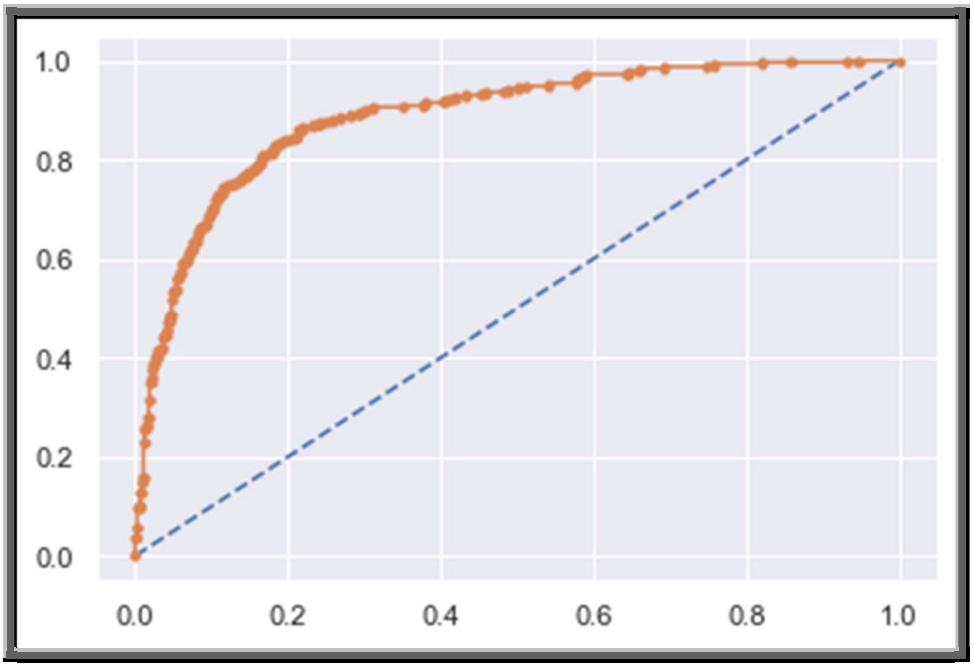


Figure 1. 19 LDR AUC and ROC for training data

Evaluation of the model:

- *The error in the test data is slightly higher than the train data, which is absolutely fine because the error margin is low and the error in both train and test data is not too high. Thus, the model is not over-fitted or under-fitted.*

PROJECT ML

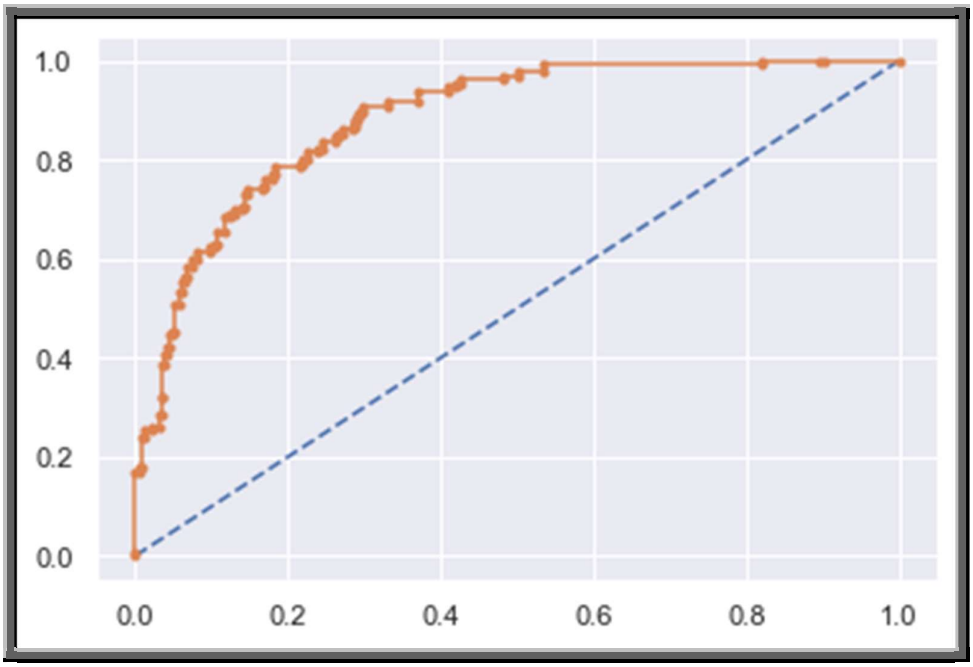


Figure 1. 20 LDR AUC and ROC for testing data

PROJECT ML

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

KNN

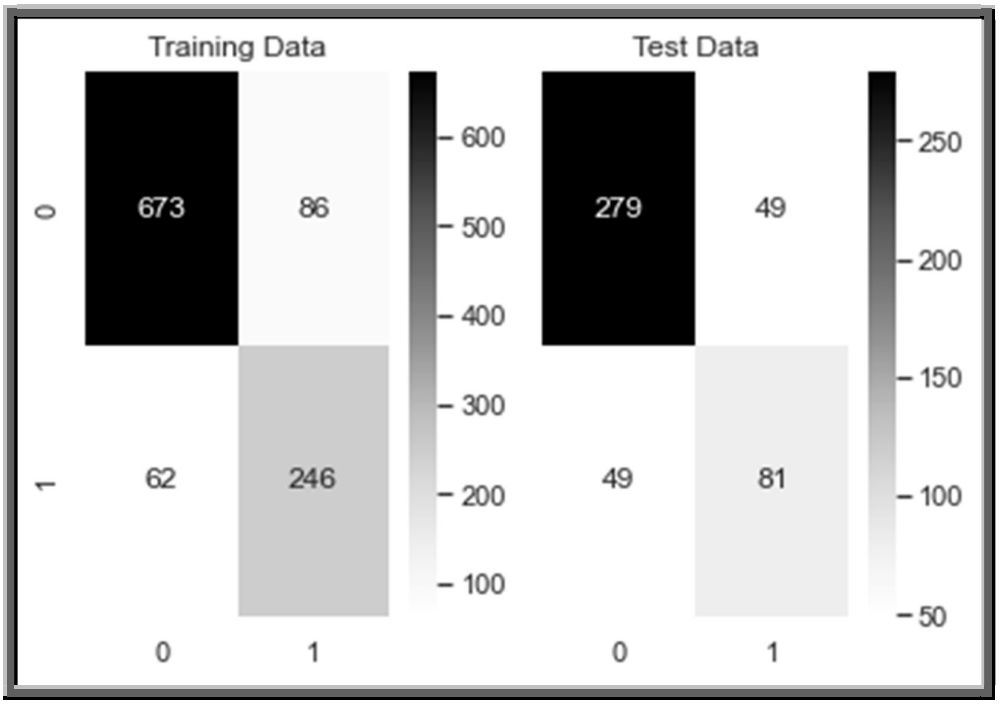


Figure 1. 21 KNN Confusion Matrix

	precision	recall	f1-score	support
0	0.92	0.89	0.90	759
1	0.74	0.80	0.77	308
accuracy			0.86	1067
macro avg	0.83	0.84	0.83	1067
weighted avg	0.87	0.86	0.86	1067

Table 1. 10 KNN Classification Report Training Data

Observation Train data:

- *Accuracy: 86%*
- *Precision: 74%*

- *Recall: 80%*
- *F1-Score:77%*

0	0.85	0.85	0.85	328
1	0.62	0.62	0.62	130
accuracy			0.79	458
macro avg	0.74	0.74	0.74	458
weighted avg	0.79	0.79	0.79	458

Table 1. 11 KNN Classification Report Testing Data

Observation Test data:

- *Accuracy: 79%*
- *Precision: 62%*
- *Recall: 62%*
- *F1-Score:62%*

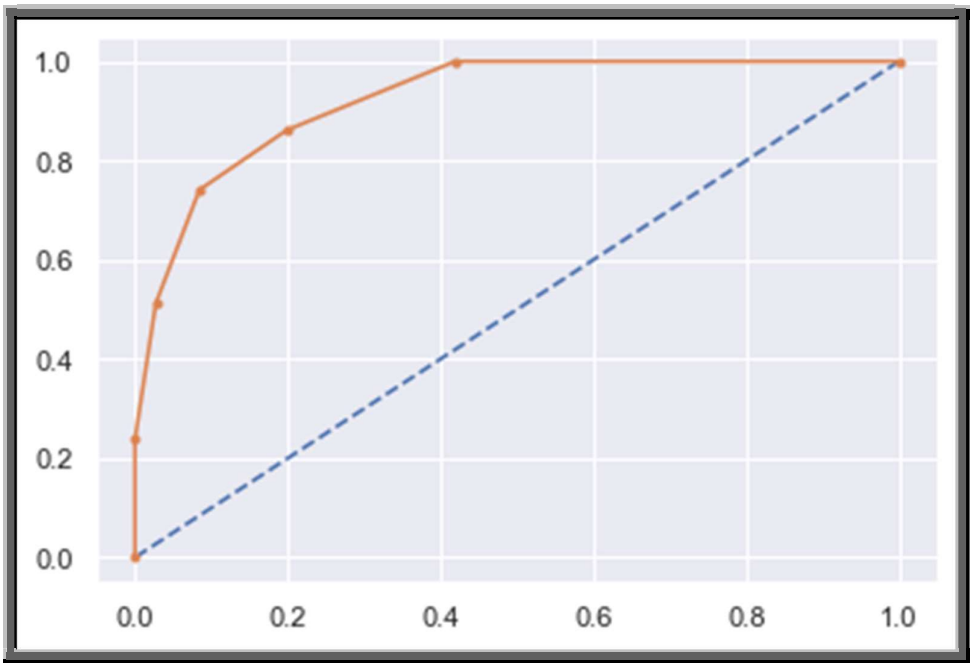


Figure 1. 22 KNN AUC and ROC for Training Data

PROJECT ML

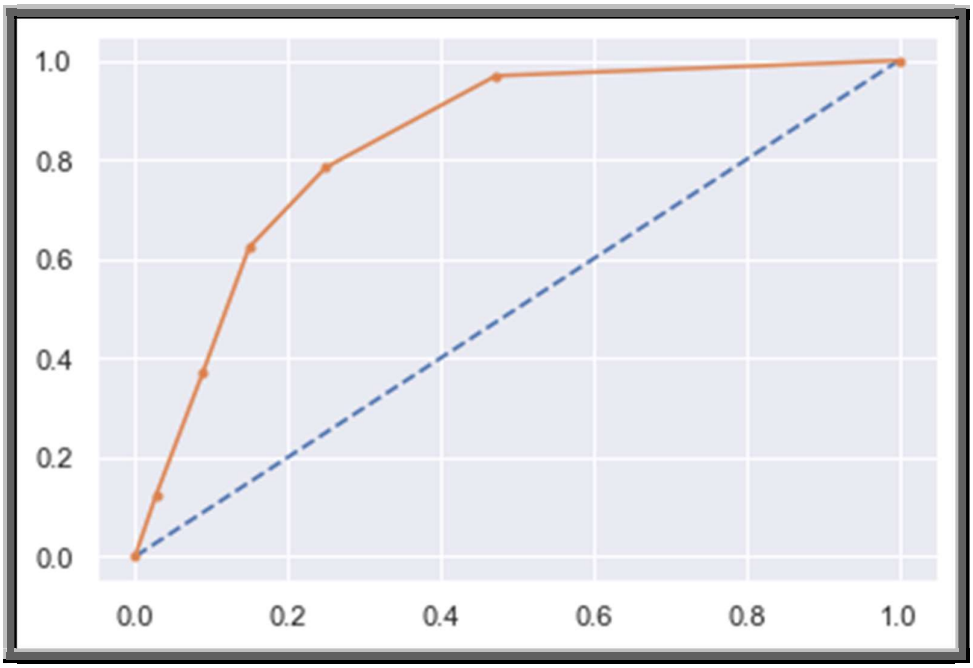


Figure 1. 23 KNN AUC and ROC for Testing Data

PROJECT ML

Naïve Bayes

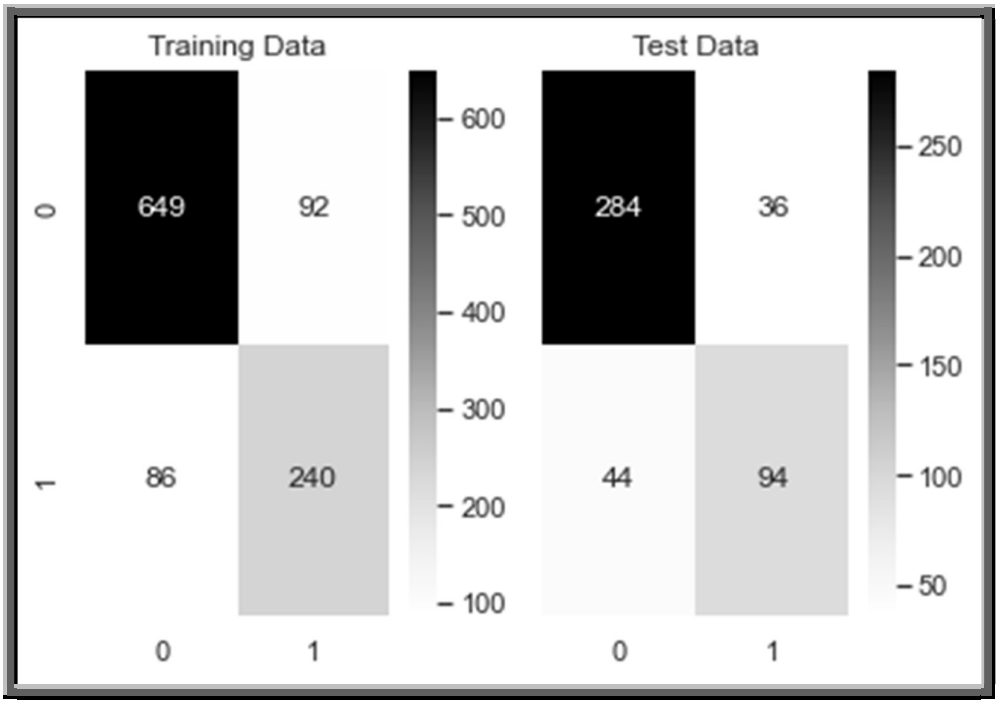


Figure 1. 24 NB Confusion Matrix

	precision	recall	f1-score	support
0	0.88	0.88	0.88	741
1	0.72	0.74	0.73	326
accuracy			0.83	1067
macro avg	0.80	0.81	0.80	1067
weighted avg	0.83	0.83	0.83	1067

Table 1. 12 NB Classification Report Training Data

Observation Train data:

- *Accuracy: 83%*
- *Precision: 72%*
- *Recall: 74%*
- *F1-Score:73%*

PROJECT ML

	precision	recall	f1-score	support
0	0.87	0.89	0.88	320
1	0.72	0.68	0.70	138
accuracy			0.83	458
macro avg	0.79	0.78	0.79	458
weighted avg	0.82	0.83	0.82	458

Table 1. 13 NB Classification Report Testing Data

Observation Test data:

- *Accuracy: 83%*
- *Precision: 72%*
- *Recall: 68%*
- *F1-Score:70%*

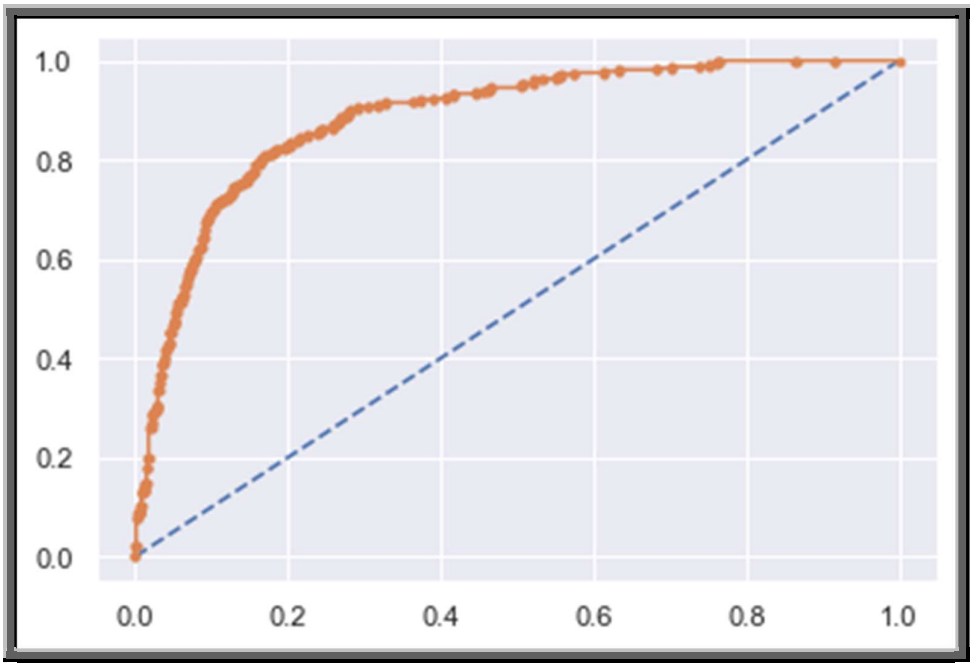


Figure 1. 25 NB AUC and ROC for Training Data

PROJECT ML

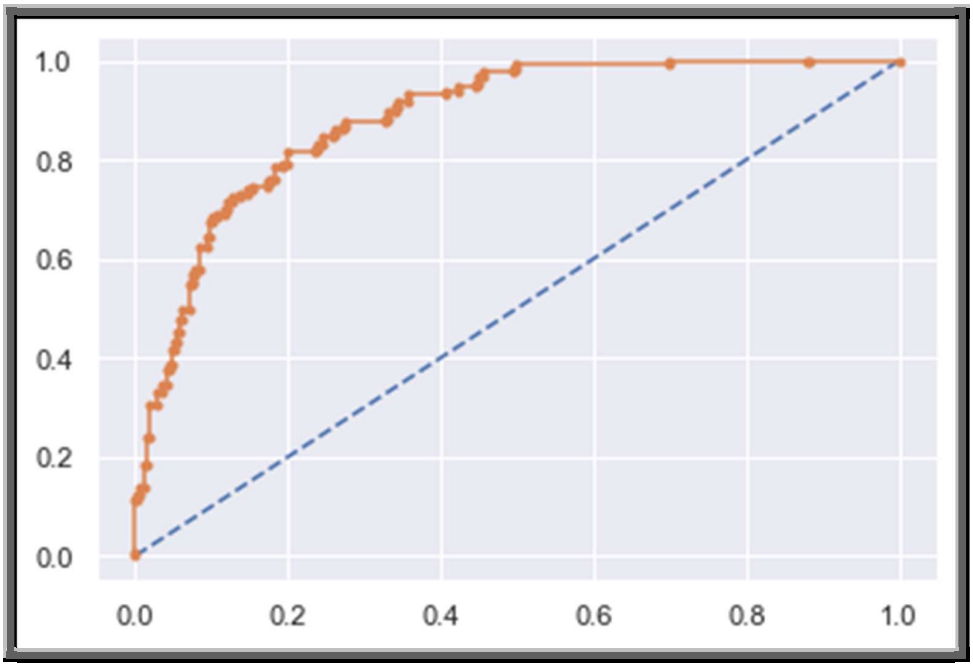


Figure 1. 26 NB AUC and ROC for Testing Data

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

AdaBoost

PROJECT ML

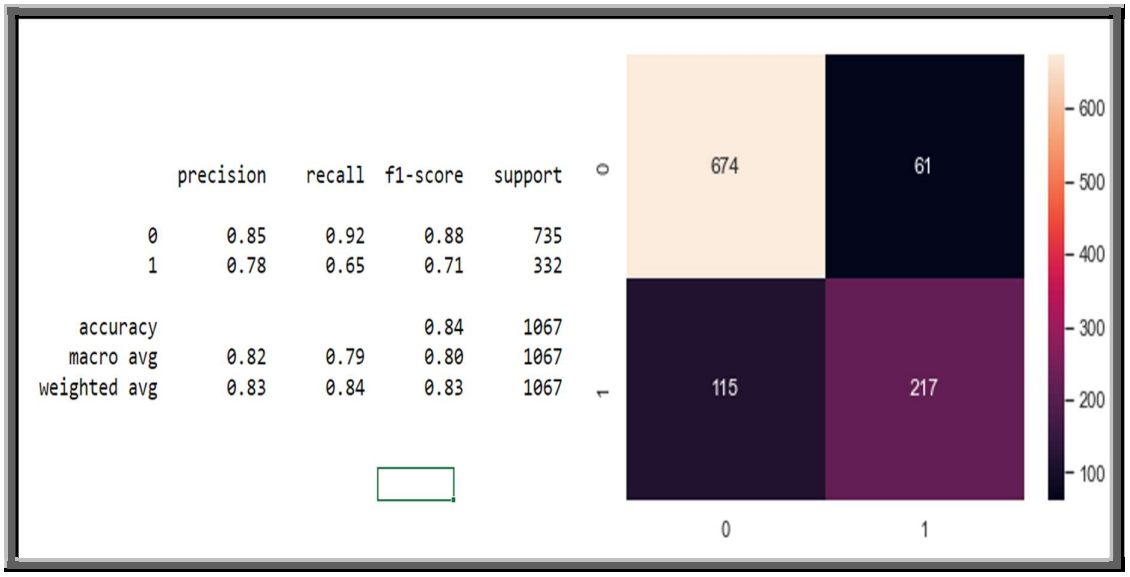


Figure 1. 27 ADB Classification Report and Confusion Matrix Training Data

Observation Train data:

- *Accuracy: 84%*
- *Precision: 78%*
- *Recall: 65%*
- *F1-Score:71%*

PROJECT ML

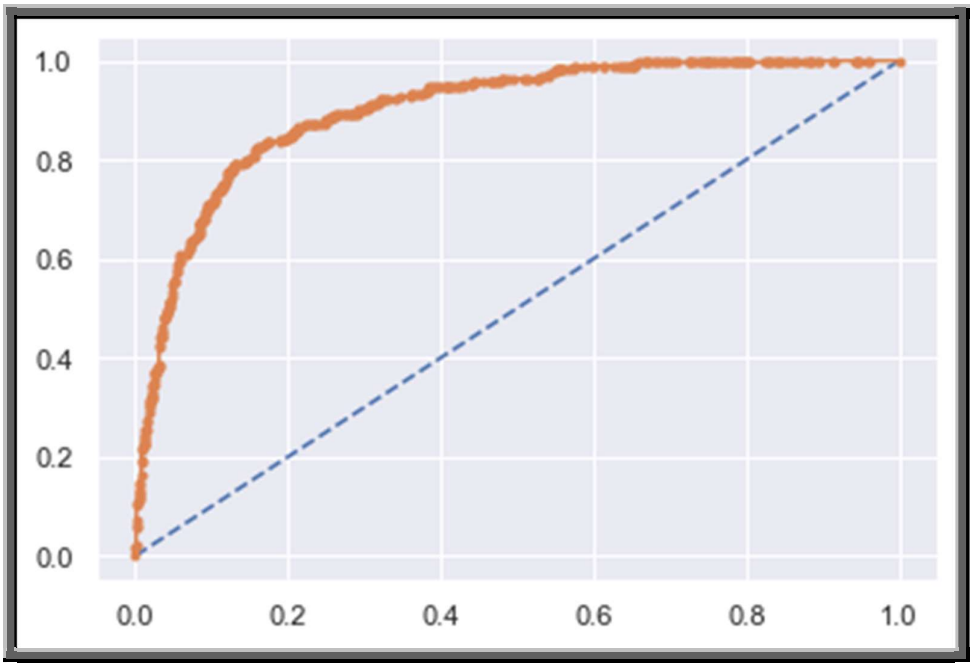


Figure 1. 28 ADB AUC and ROC for Training Data

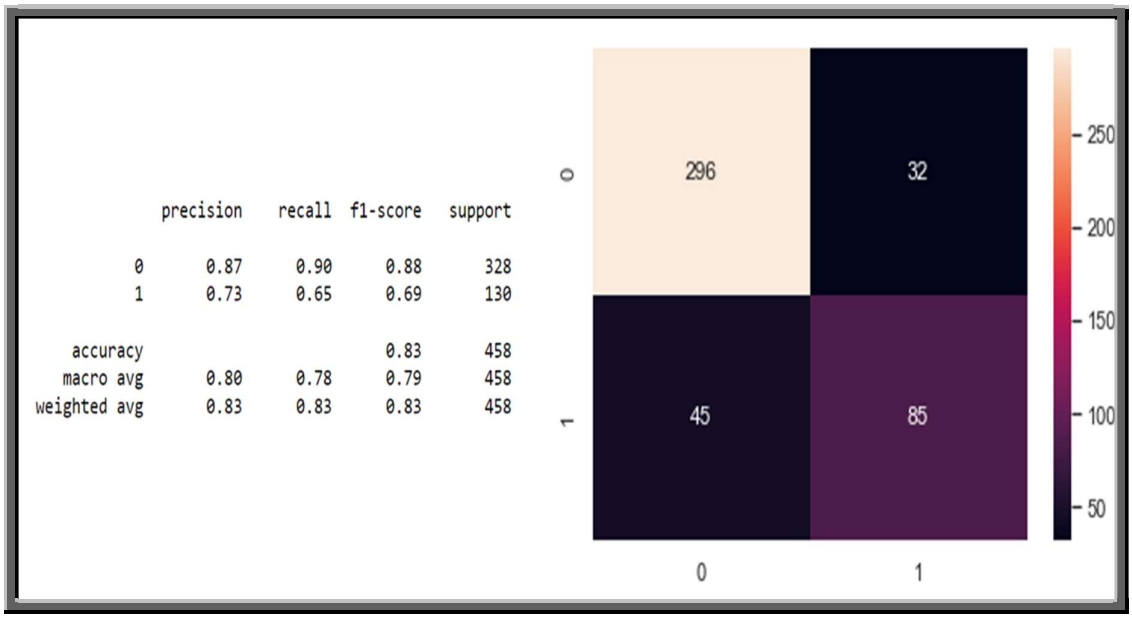


Figure 1. 29 ADB Classification Report and Confusion Matrix Testing Data

Observation Test data:

- *Accuracy: 83%*
- *Precision: 73%*
- *Recall: 65%*

- *F1-Score:69%*

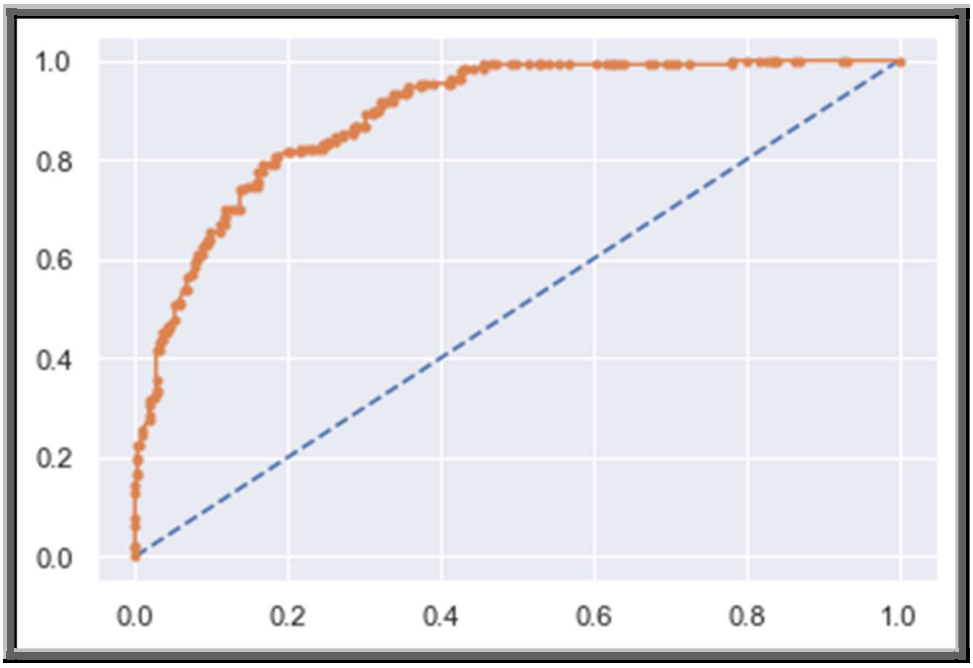
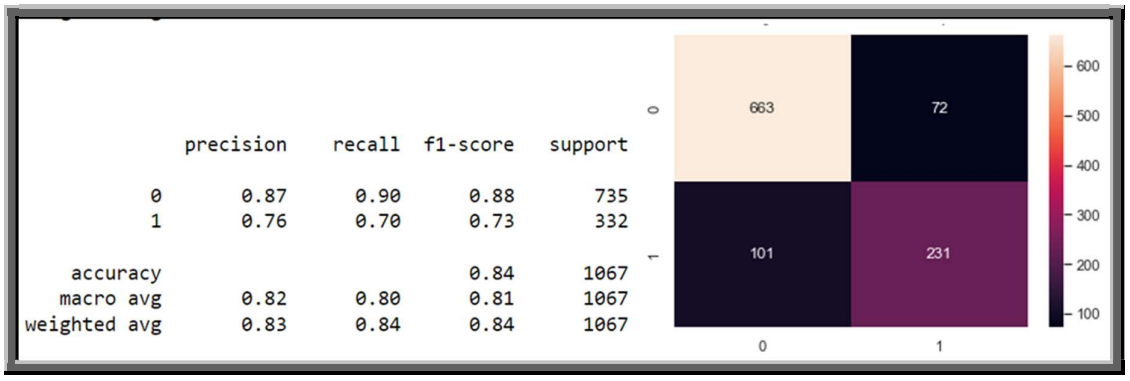


Figure 1. 30 ADB AUC and ROC for Testing Data

Decision Tree



	precision	recall	f1-score	support
0	0.87	0.90	0.88	735
1	0.76	0.70	0.73	332
accuracy			0.84	1067
macro avg	0.82	0.80	0.81	1067
weighted avg	0.83	0.84	0.84	1067

Figure 1. 31 Decision Forest Classification report and Confusion matrix Training Data

Observation Train data:

- *Accuracy: 84%*
- *Precision: 76%*

- *Recall: 70%*
- *F1-Score:73%*

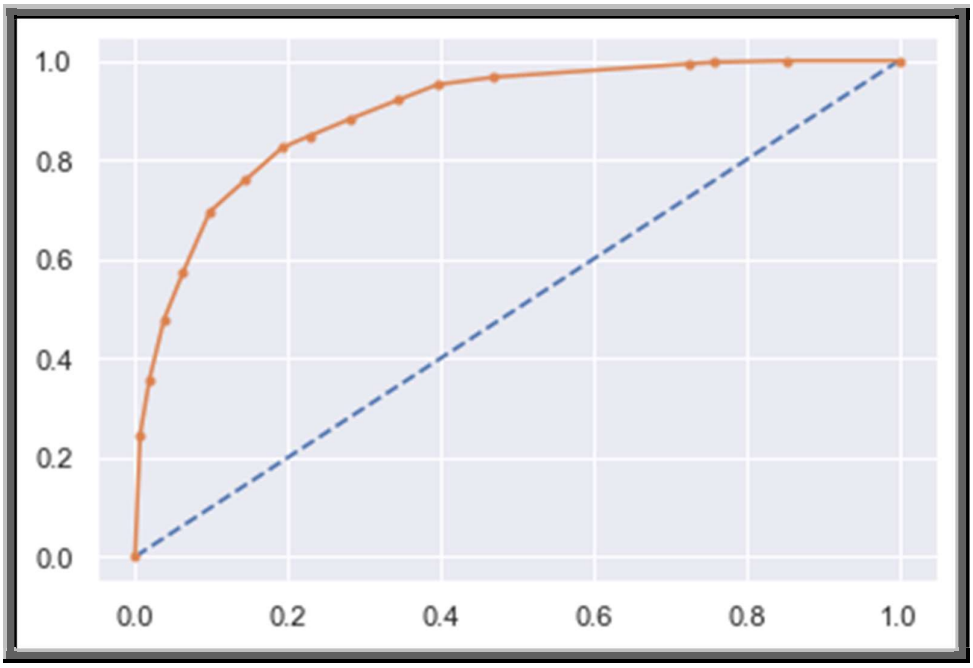
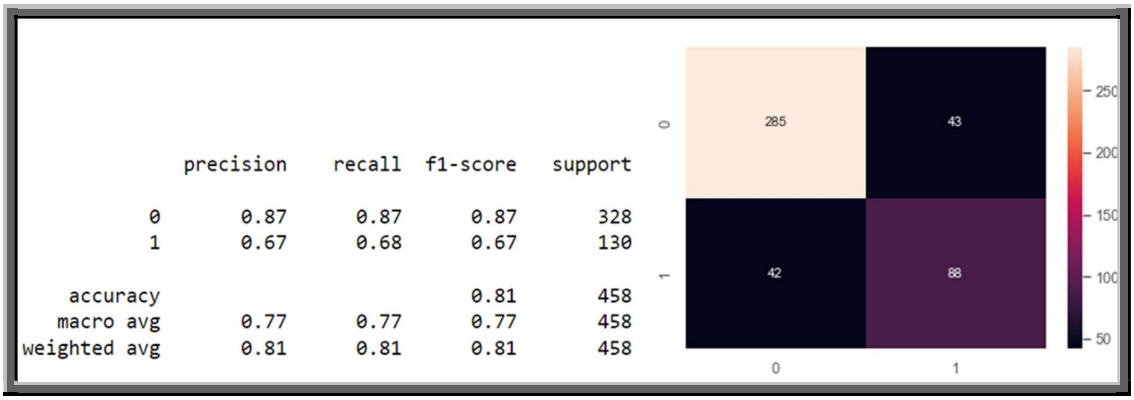


Figure 1. 32 Decision Tress AUC and ROC curve Training Data



	precision	recall	f1-score	support
0	0.87	0.87	0.87	328
1	0.67	0.68	0.67	130
accuracy			0.81	458
macro avg	0.77	0.77	0.77	458
weighted avg	0.81	0.81	0.81	458

Figure 1. 33 Decision Forest Classification report and Confusion matrix Testing Data

Observation Test data:

- *Accuracy: 81%*
- *Precision: 67%*
- *Recall: 68%*

- *F1-Score:67%*

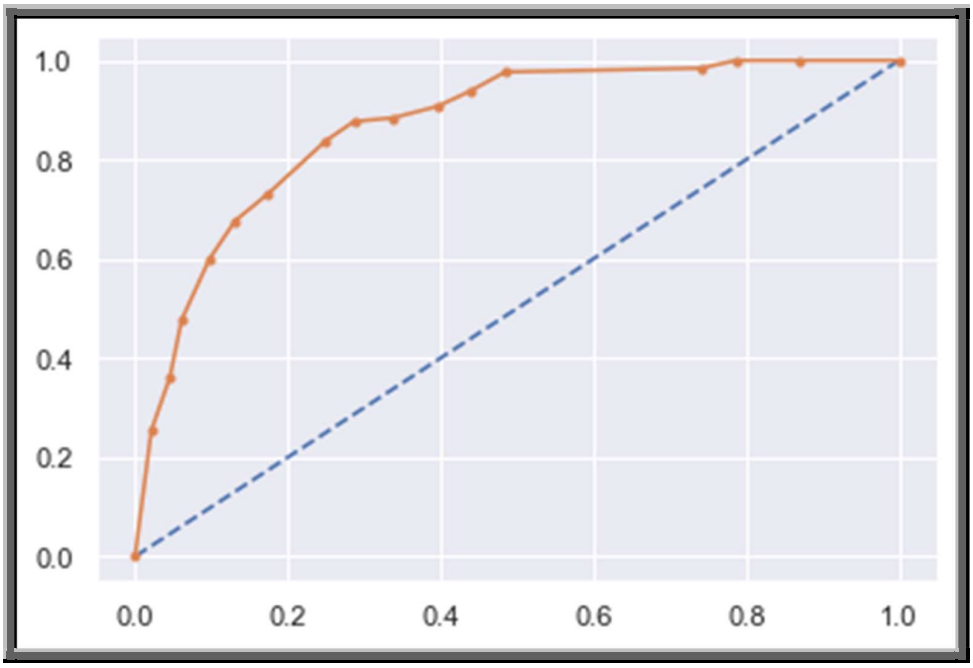
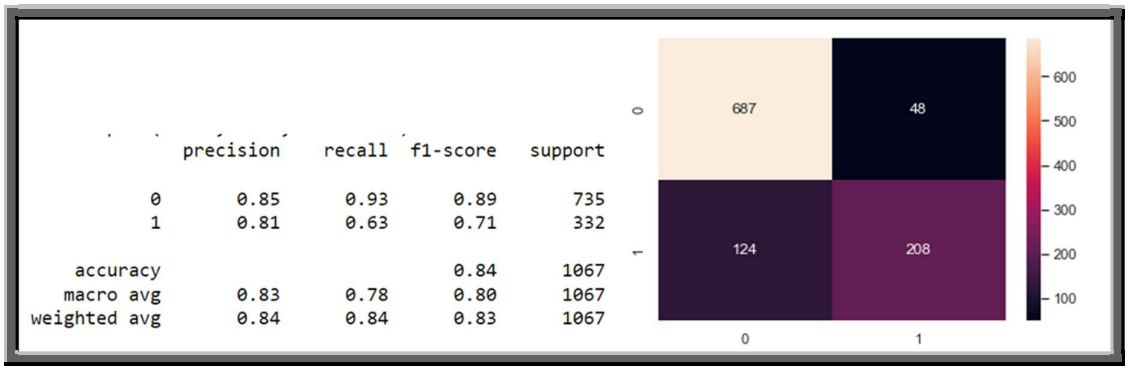


Figure 1. 34 Decision Tress AUC and ROC curve Testing Data

Random Foresting



	precision	recall	f1-score	support
0	0.85	0.93	0.89	735
1	0.81	0.63	0.71	332
accuracy			0.84	1067
macro avg	0.83	0.78	0.80	1067
weighted avg	0.84	0.84	0.83	1067

Figure 1. 35 Random Forest Classification report and Confusion matrix Training Data

Observation Train data:

- *Accuracy: 84%*
- *Precision: 81%*
- *Recall: 63%*
- *F1-Score: 71%*

PROJECT ML

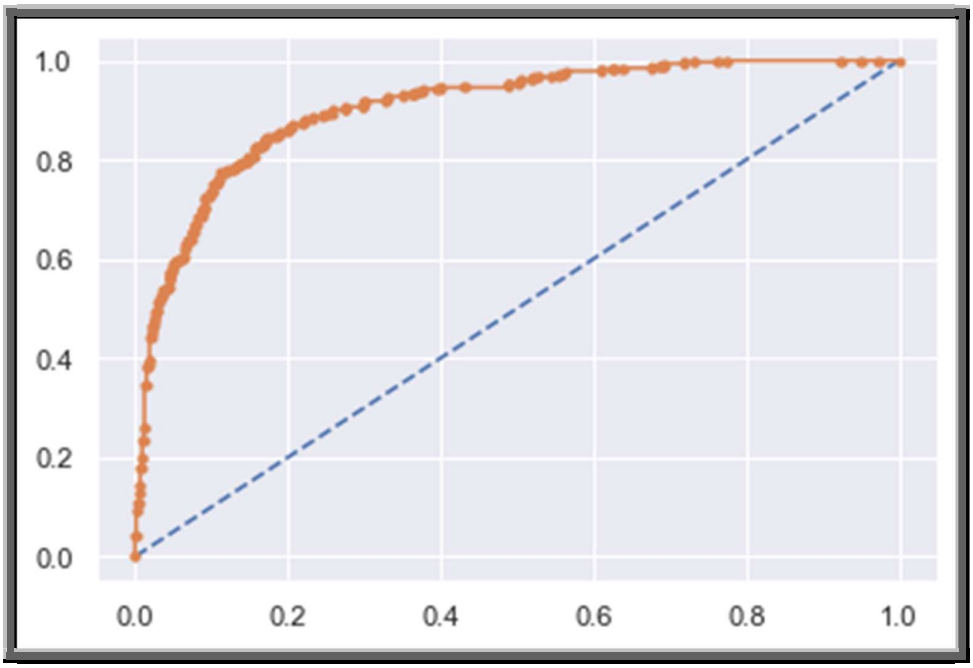


Figure 1. 36 Random Forest AUC and ROC curve Training Data

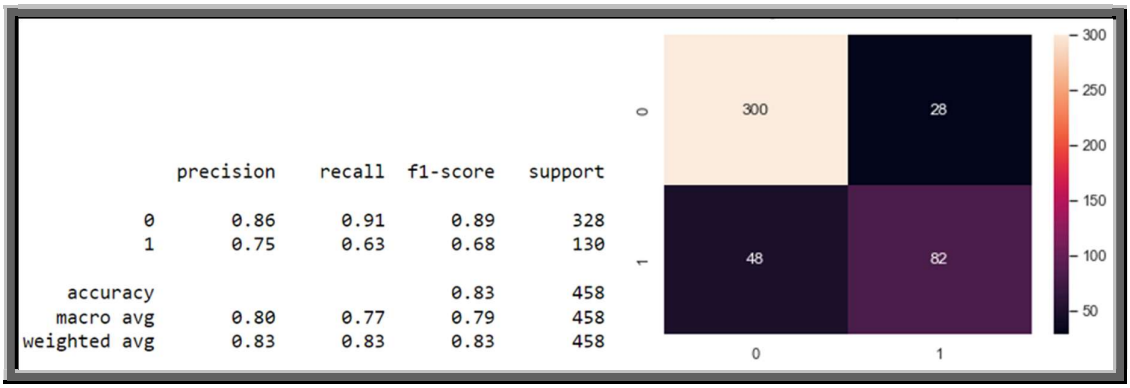


Figure 1. 37 Random Forest Classification report and Confusion matrix Testing Data

Observation Test data:

PROJECT ML

- *Accuracy: 83%*
- *Precision: 75%*
- *Recall: 63%*
- *F1-Score:68%*

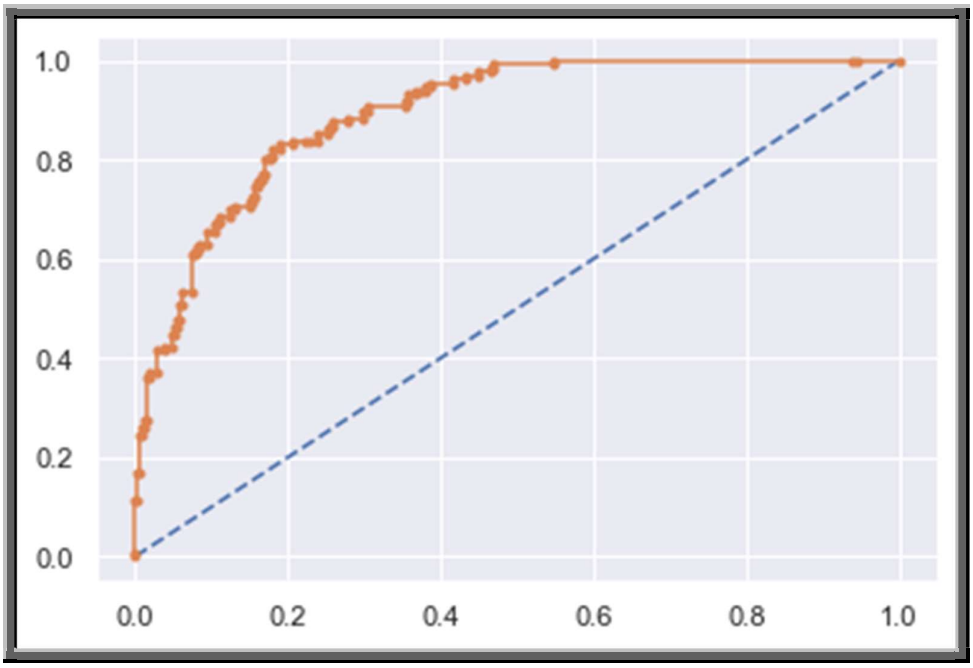


Figure 1. 38 Random Forest AUC and ROC curve Testing Data

1.7 Performance Metrics: Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC_AUC score for each model. Final Model: Compare the models and write inference which model is best/optimized. (7 marks)

- *In all the models, tuned ones are better than the regular models. So, we compare only the tuned models and describe which model is the best/optimized.*

PROJECT ML

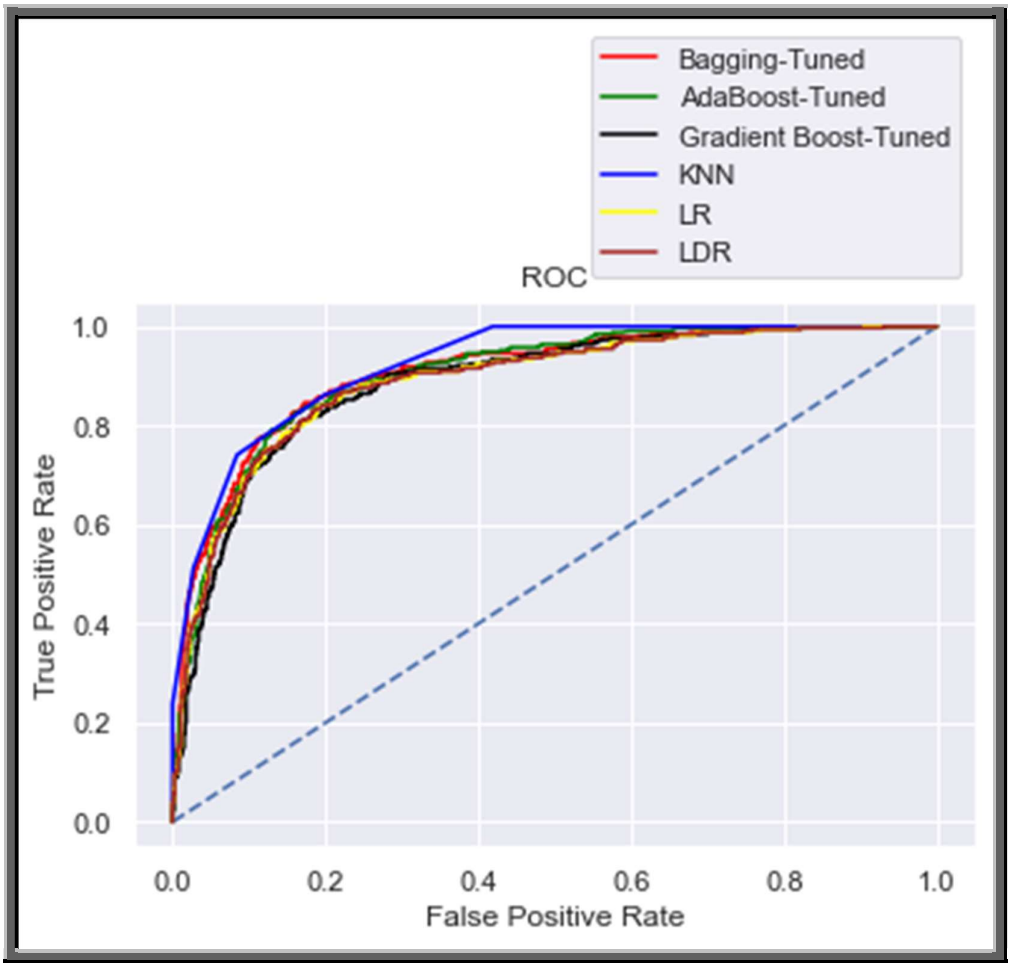


Figure 1. 39 Model Evaluation Training Data

- *All the tuned models have high values and every model is good. But as we can see, the most consistent tuned model in both train and test data is the Gradient Boost model.*
- *The gradient boost model performs the best accuracy score in train and 87.28% accuracy score in test. Also, it has the best AUC score in both train and test data which is the highest of all the models.*
- *It also has the best precision score and recall of which is also the highest of all the models. So, we conclude that Gradient Boost tuned model is the optimized model.*

PROJECT ML

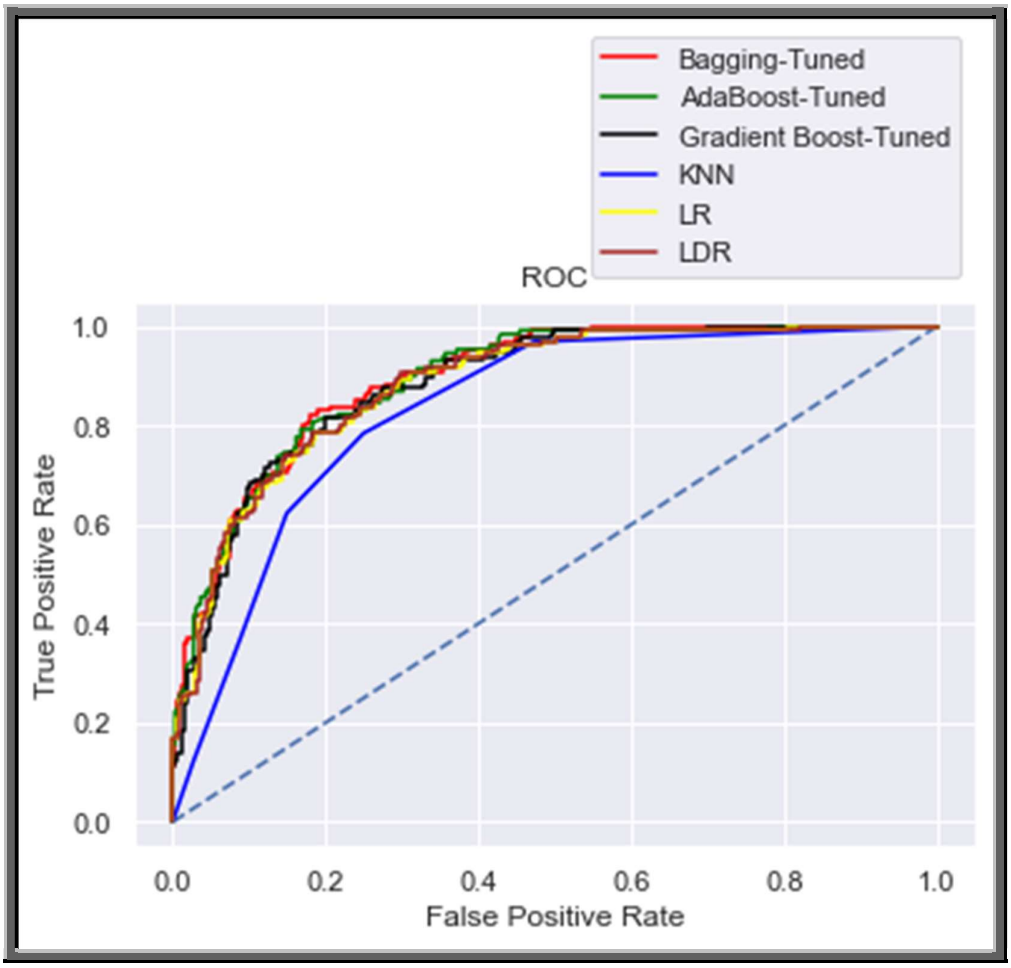


Figure 1. 40 Model Evaluation Testing data

Score	LR		LDA		KNN		Naïve Bayes	
	Train	Test	Train	Test	Train	Test	Train	Test
Accuracy	0.84	0.82	0.84	0.82	0.86	0.79	0.83	0.83
AUC	0.889	0.882	0.889	0.884	0.924	0.832	0.886	0.885
Recall	0.69	0.65	0.76	0.69	0.8	0.62	0.74	0.68
Precision	0.77	0.7	0.7	0.66	0.74	0.62	0.72	0.72
F1 Score	0.73	0.68	0.73	0.67	0.77	0.62	0.73	0.7

Table 1. 14 Comparison of report among the models'

PROJECT ML

Score	AdaBoosting		Decision Tree		Random Forest	
	Train	Test	Train	Test	Train	Test
Accuracy	0.84	0.83	0.84	0.81	0.84	0.83
AUC	0.902	0.893	0.896	0.87	0.906	0.895
Recall	0.65	0.65	0.7	0.68	0.63	0.63
Precision	0.78	0.73	0.76	0.67	0.81	0.75
F1 Score	0.71	0.69	0.73	0.67	0.71	0.68

Table 1. 15 Tuned Models' Evaluation

- *As we can see from the above tabular comparison, there is not much difference between the performance regular LR model and tuned LR model.*
- *The values are high overall and there is no over-fitting or under-fitting. Therefore, both models are equally good models.*
- *The gradient boost classifier after tuning, has improved the model significantly.*
- *The difference between the train and test accuracies has also been reduced.*
- *Overall, the tuned Gradient Boost classifier is a better model.*
- *As we can see, there is not much difference between the performance of regular LDA model and tuned LDA model.*
- *The values are high overall and there is no over-fitting or under-fitting.*
- *Hence, both models are equally good models.*
- *There is no under-fitting or over-fitting in any of the tuned models.*

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

PROJECT ML

Major Insights

- *"0" has more than double the votes of "1".*
- *Most number of people have given a score of 3 and 4 for the national economic condition and the average score is 3.14*
- *Blair has higher number of votes than Hague and the scores are much better for Blair than for Hague.*
- *The average score of Blair is 3.33 and the average score of Hague is 2.74. So, here we can see that, Blair has a better score.*
- *On a scale of 0 to 3, about 30% of the total population has zero knowledge about politics/parties.*
- *People who gave a low score of 1 to a certain party, still decided to vote for the same party instead of voting for the other party. This can be because of lack of political knowledge among the people.*
- *People who have higher Eurosceptic sentiment, has voted for the conservative party and lower the Eurosceptic sentiment, higher the votes for Labour party.*
- *Out of 454 people who gave a score of 0 for political knowledge, most people have voted for the "0" and lesses people have voted for the "1".*
- *All models performed well on training data set as well as test data set. The tuned models have performed better than the regular models.*
- *There is no over-fitting in any model except Random Forest and Bagging regular models.*
- *Gradient Boosting model tuned is the best/optimized model.*

Business recommendations:

- *Hyper-parameters tuning is an import aspect of model building. There are limitations to this as to process these combinations, huge amount of processing power is required. But if tuning can*

be done with many sets of parameters, we might get even better results.

- *Gathering more data will also help in training the models and thus improving the predictive powers.*
- *We can also create a function in which all the models predict the outcome in sequence. This will help in better understanding and the probability of what the outcome will be.*
- *Using Gradient Boosting model without scaling for predicting the outcome as it has the best optimized*

Problem 2:

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

President Franklin D. Roosevelt in 1941

President John F. Kennedy in 1961

President Richard Nixon in 1973

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

The No. of character, words and sentences present in President Franklin Roosevelt's speech are 7571, 1536 and 68 Respectively.

The No. of character, words and sentences present in President John Kennedy's speech are 7618, 1546 and 52 Respectively.

The No. of character, words and sentences present in President Richard Nixon's speech are 9991, 2028 and 69 Respectively.

PROJECT ML

2.2 Remove all the stopwords from all three speeches. – 3 Marks

Before, removing the stop-words, we have changed all the letters to lowercase and we have removed special characters.

Word count before and after the removal of stop-words:

Before the removal of stop-words,

- *President Franklin D. Roosevelt's speech have 1536 words.*
- *President John F. Kennedy's speech have 1546 words.*
- *President Richard Nixon's speech have 2028 words.*

After the removal of stop-words,

- *President Franklin D. Roosevelt's speech have 608 words.*
- *President John F. Kennedy's speech have 651 words.*
- *President Richard Nixon's speech have 773 words.*

2.3 Which word occurs the most number of times in his inaugural address for each president? Mention the top three words. (After removing the stopwords) – 3 Marks

Top 3 words in Roosevelt's speech:

```
[('nation', 12), ('spirit', 9), ('life', 9), ('democracy', 9)]
```

- *Nation - 12 times*

- *Spirit - 9 times*
- *Life - 9 times*

Top 3 words in Kennedy's speech:

```
[('america', 21), ('peace', 19), ('world', 18), ('nation', 11)]
```

- *America - 12 times*
- *Peace - 9 times*
- *World - 9 times*

Top 3 words in Nixons's speech:

```
[('america', 21), ('peace', 19), ('world', 18), ('nation', 11)]
```

- *America - 21 times*
- *Peace - 19 times*
- *World - 18 times*

2.4 Plot the word cloud of each of the speeches of the variable. (After removing the stopwords) – 3 Marks [refer to the End-to-End Case Study done in the Mentored Learning Session]

PROJECT ML

Word cloud of Roosevelt's speech:

PROJECT ML

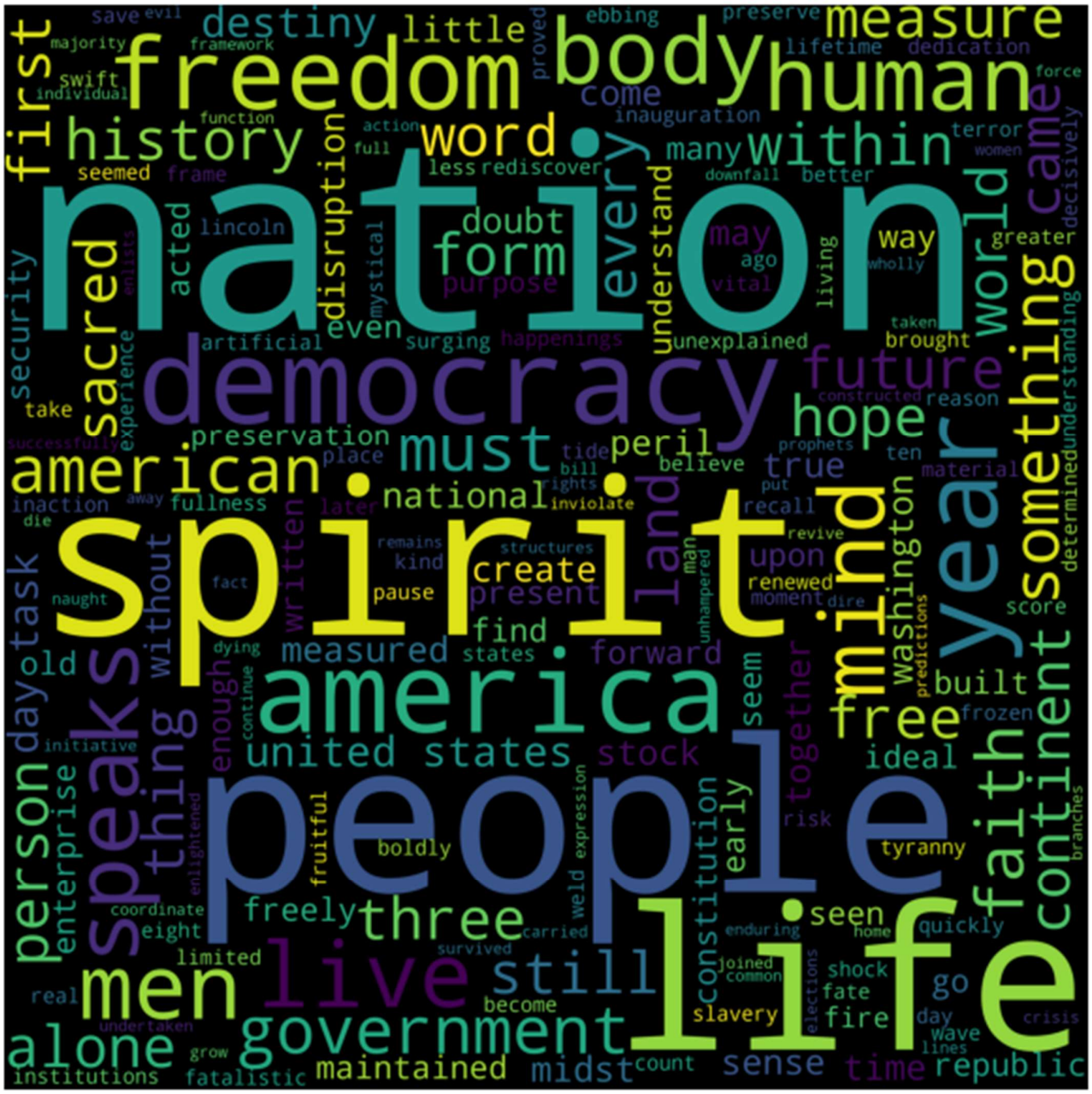


Figure 2. 1 Word cloud of Roosevelt's speech

PROJECT ML

Word cloud of Kennedy's speech:

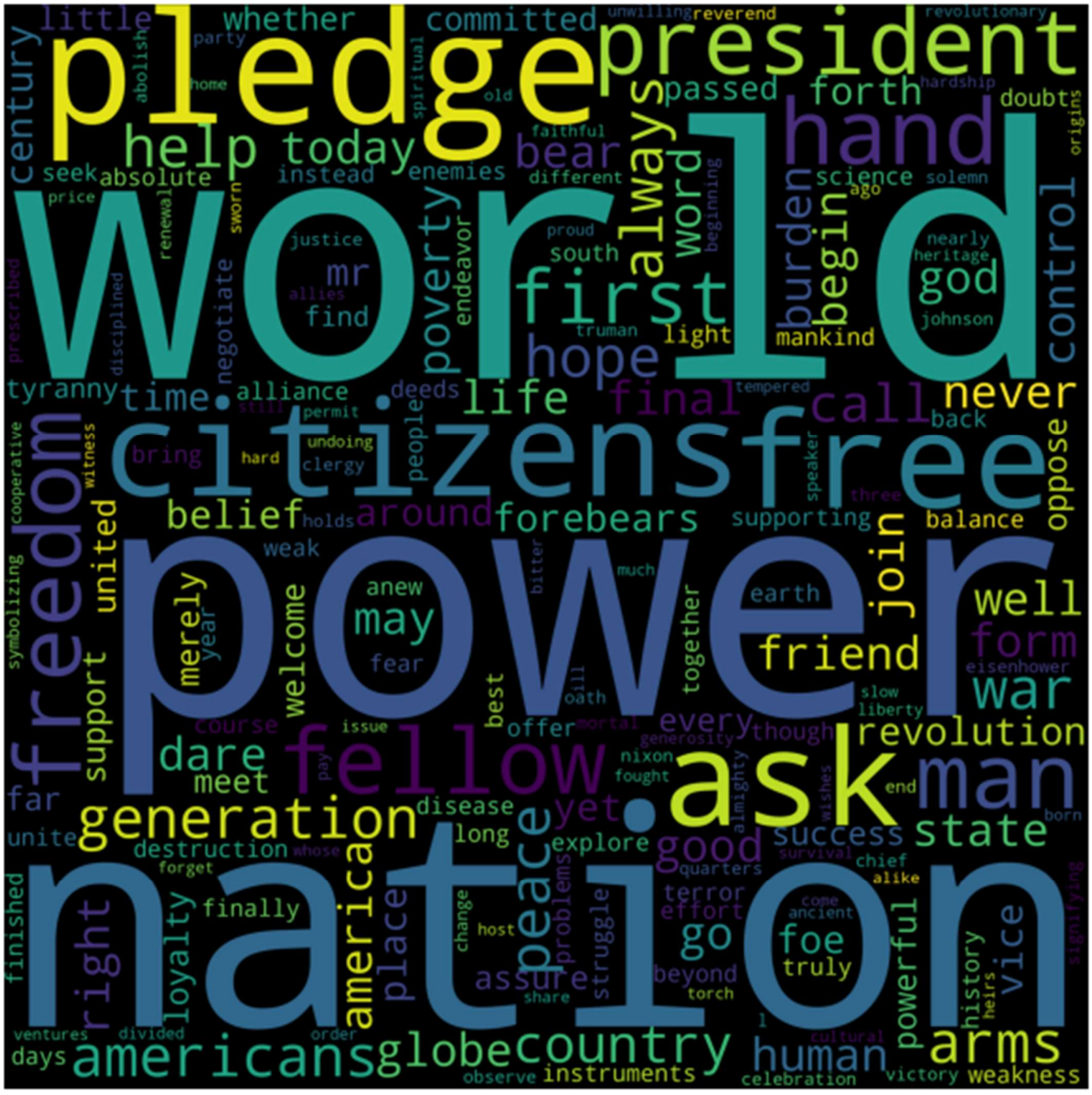


Figure 2. 2 Word cloud of Kennedy's speech

PROJECT ML

Word cloud of Nixon's speech:

Figure 2. 3 Word cloud of Nixon's speech