IME-672A
Data Mining & Knowledge Discovery

Corporate Rating

Pushpanjali Kumari *180569*

Acknowledgments-

I wish to express my sincere gratitude to my instructor **Dr. Faiz Hamid** for his valuable advice and guidance in completing this project. The way he presented each and every topic in the class made the topics very interesting and understandable, which helped a lot in making our project possible.

Problem Description -

What is a Corporate Credit Rating?

A corporate credit rating is an opinion of an independent agency regarding the likelihood that a corporation will fully meet its financial obligations(meet the terms of a contract) as they come due. A company's corporate credit rating indicates its relative ability to pay its creditors. It is important to keep in mind that corporate credit ratings are an opinion, not a fact.

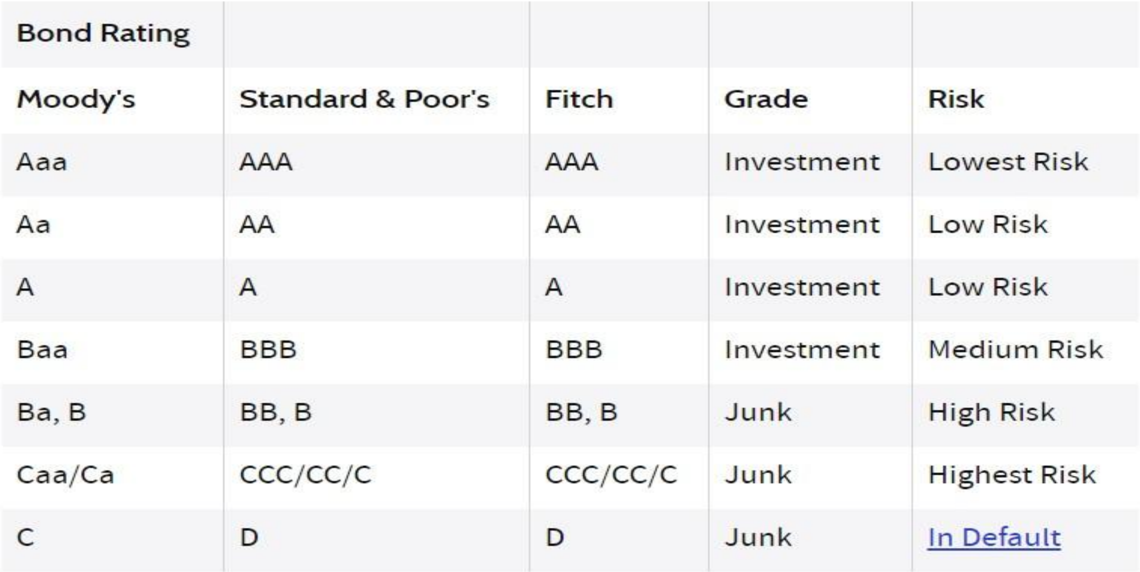
Key Takeaways

- Corporate credit ratings are the assessment of a company's ability to pay its debts according to an independent credit rating agency.
- The three biggest credit rating agencies are: Standard and Poor's (S&P), Moody's, and Fitch.
- Corporate credit rating trends, over time, may allow an investor to compare the credit-worthiness of competing corporations.

For example, Standard & Poor's uses "AAA" for the highest credit quality with the lowest credit risk, "AA" for the next best, followed by "A," then "BBB" for satisfactory credit

Description of the Data -

The ratings data set is an anonymized data set with corporate ratings where the ratings have been numerically encoded (1 = AAA, and so on). It has the following attributes:



Bond Rating				
Moody's	Standard & Poor's	Fitch	Grade	Risk
Aaa	AAA	AAA	Investment	Lowest Risk
Aa	AA	AA	Investment	Low Risk
A	A	A	Investment	Low Risk
Baa	BBB	BBB	Investment	Medium Risk
Ba, B	BB, B	BB, B	Junk	High Risk
Caa/Ca	CCC/CC/C	CCC/CC/C	Junk	Highest Risk
C	D	D	Junk	In Default

- **Spid: ID number**
  type - Nominal

- **Rating:**
  type - Ordinal
  range - 1 to 10

- **COMMEQTA: (Common equity to total assets)**
  Common equity is the amount that all common shareholders have invested in a company
  type - Ratio Numeric , continuous

- **LLPLOANS: (Loan loss provision to total loans) -**
  A loan loss provision is an income statement expense set aside to allow for uncollected loans and loan payments
  type - Ratio Numeric , continuous

- **COSTTOINCOME: (Operating costs to operating income)-**
  Operating costs are the ongoing expenses incurred from the normal day-to-day of running a business.Operating income reports the amount of profit realized from a business's ongoing operations.
  type - Ratio Numeric , continuous

- **ROE: (Return on equity)**

Return on equity is a measure for financial performance calculated by dividing net income by shareholder’s equity
type - Ratio Numeric , continuous

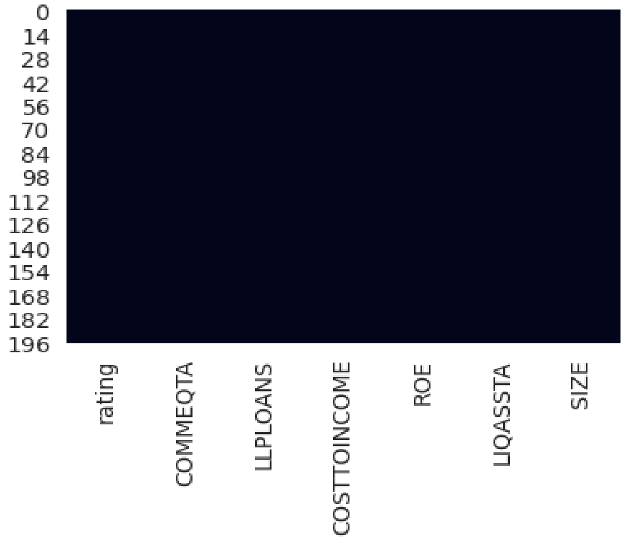
- **LIQASSTA: (Liquid assets to total assets)**
A liquid asset is something you own that can quickly and simply be converted into cash while retaining its market value.
type - Ratio Numeric , continuous

- **SIZE: (Natural logarithm of total assets)**
type - Ratio Numeric , continuous

FEATURES OF DATASET :

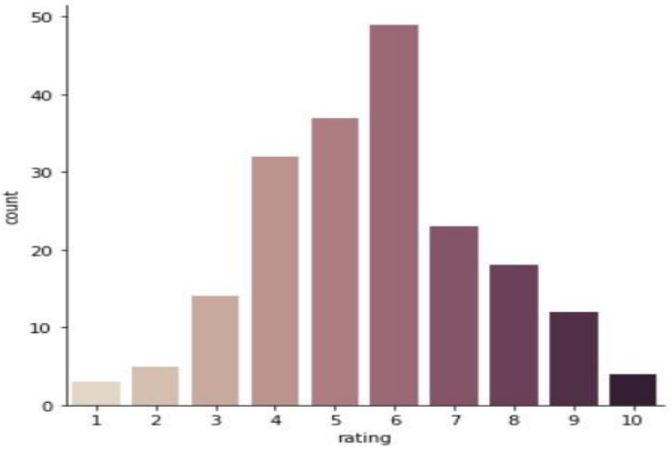
Data Preprocessing

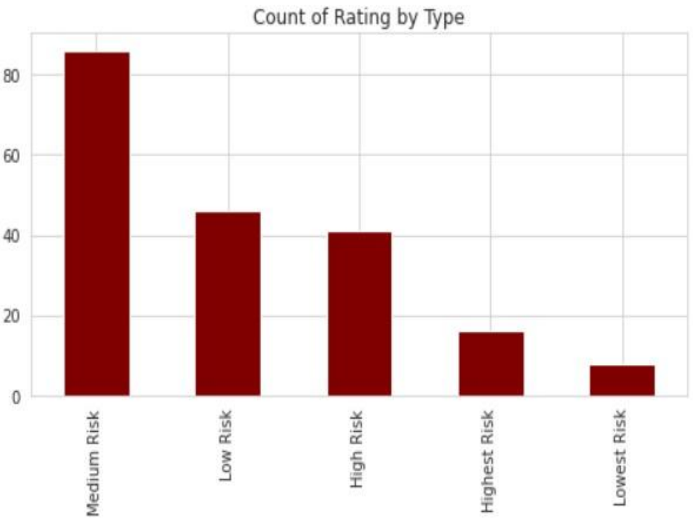
- There is neither any missing value nor any noisy data in our dataset.
- We have scaled ratings between 1-10 with categories as Lowest to highest risk.



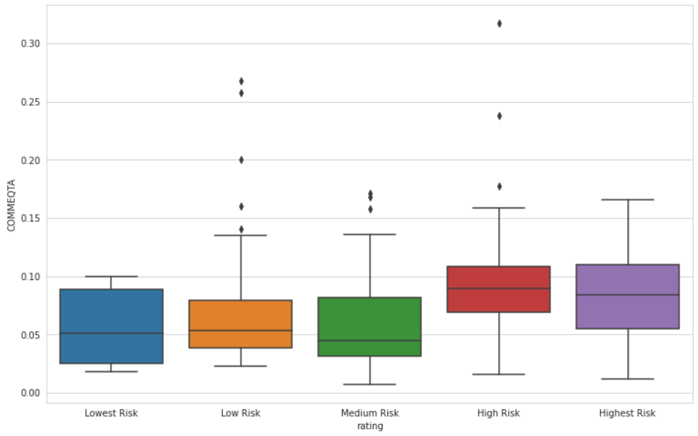
Data Visualisation -

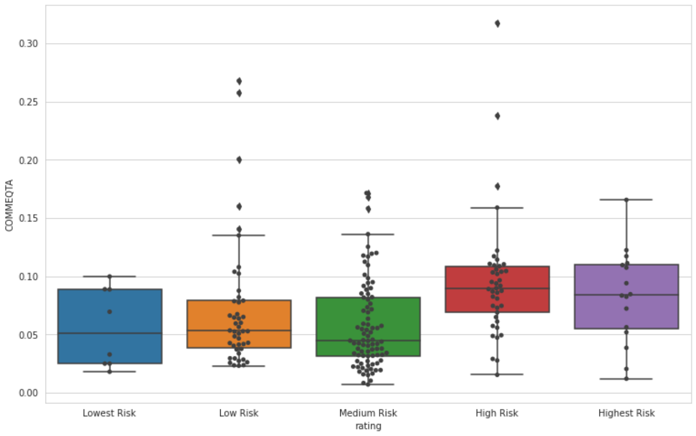
This plot shows medium risk rating count is more than others

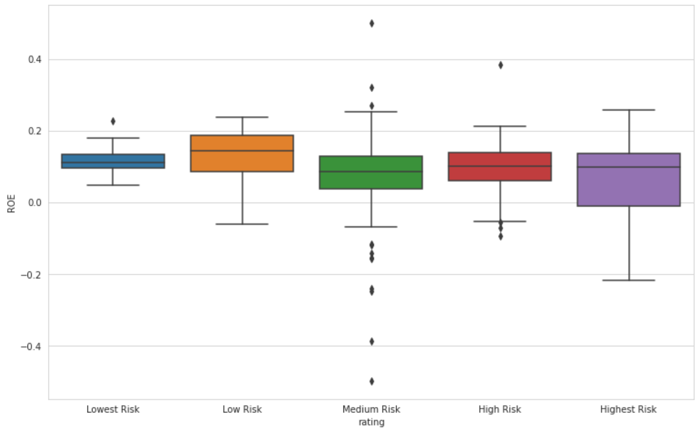


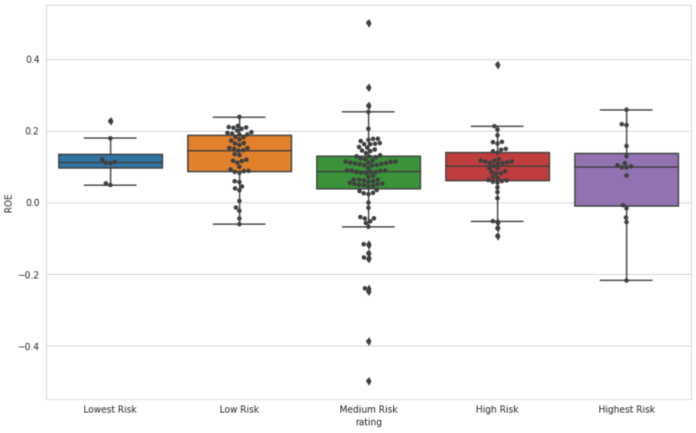


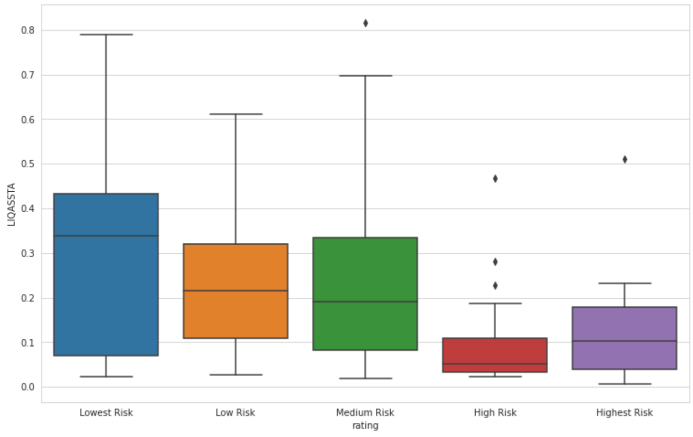
BOX & VIOLIN PLOTS -

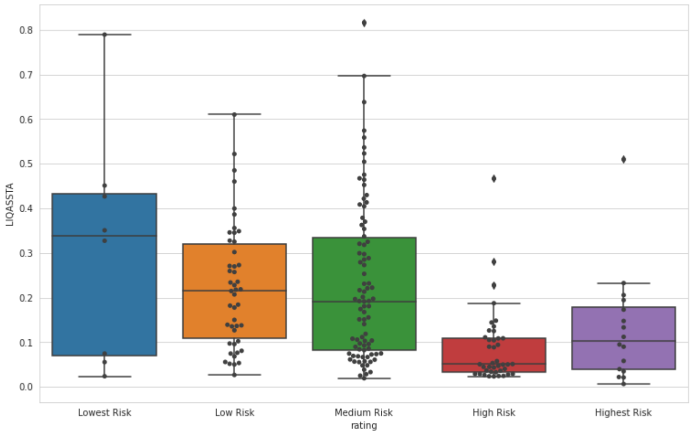


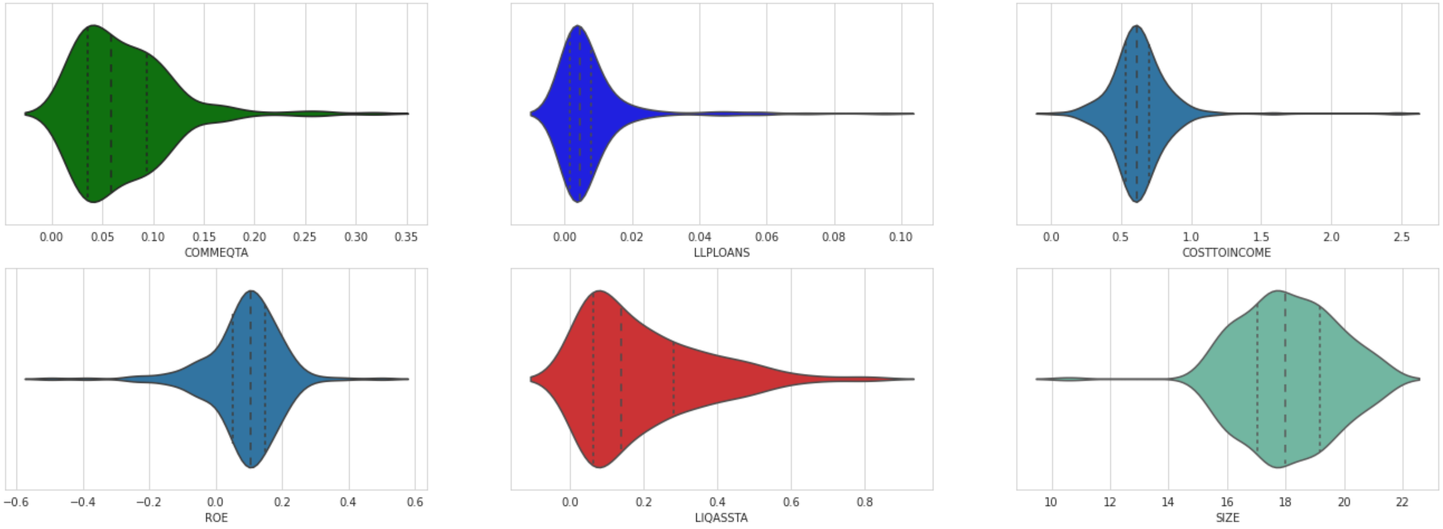




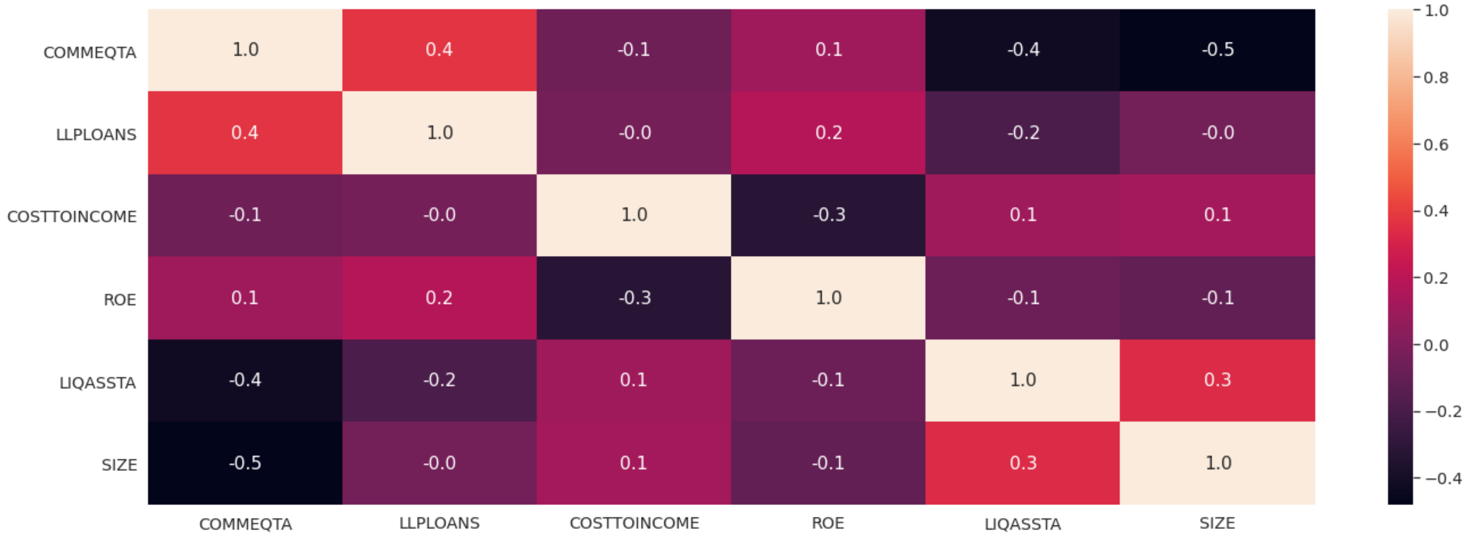






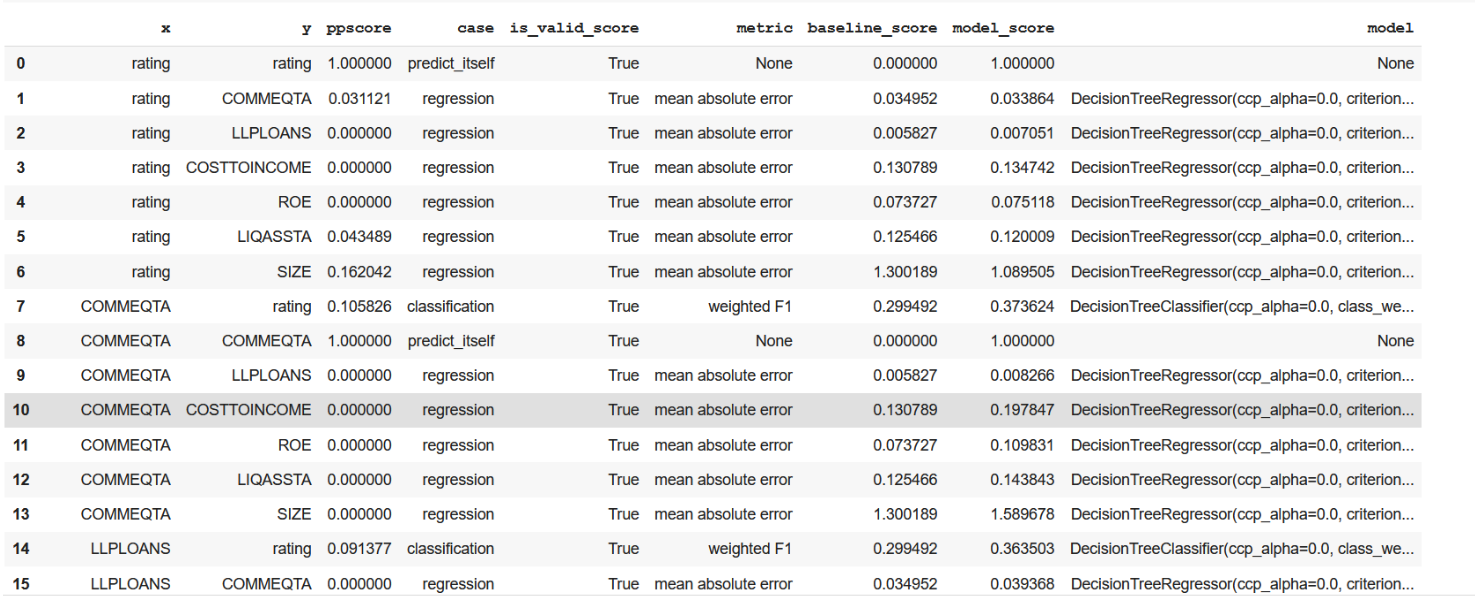


CORRELATION MATRIX-

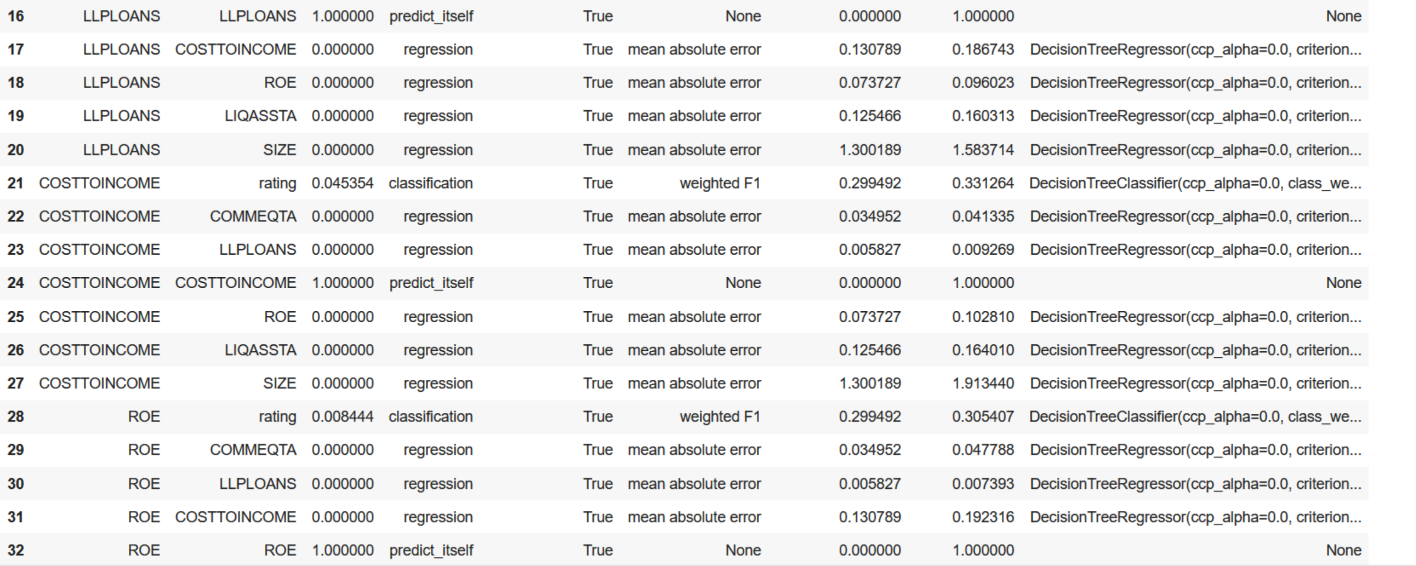


	COMMEQTA	LLPLOANS	COSTTOINCOME	ROE	LIQASSTA	SIZE
COMMEQTA	1.0	0.4	-0.1	0.1	-0.4	-0.5
LLPLOANS	0.4	1.0	-0.0	0.2	-0.2	-0.0
COSTTOINCOME	-0.1	-0.0	1.0	-0.3	0.1	0.1
ROE	0.1	0.2	-0.3	1.0	-0.1	-0.1
LIQASSTA	-0.4	-0.2	0.1	-0.1	1.0	0.3
SIZE	-0.5	-0.0	0.1	-0.1	0.3	1.0

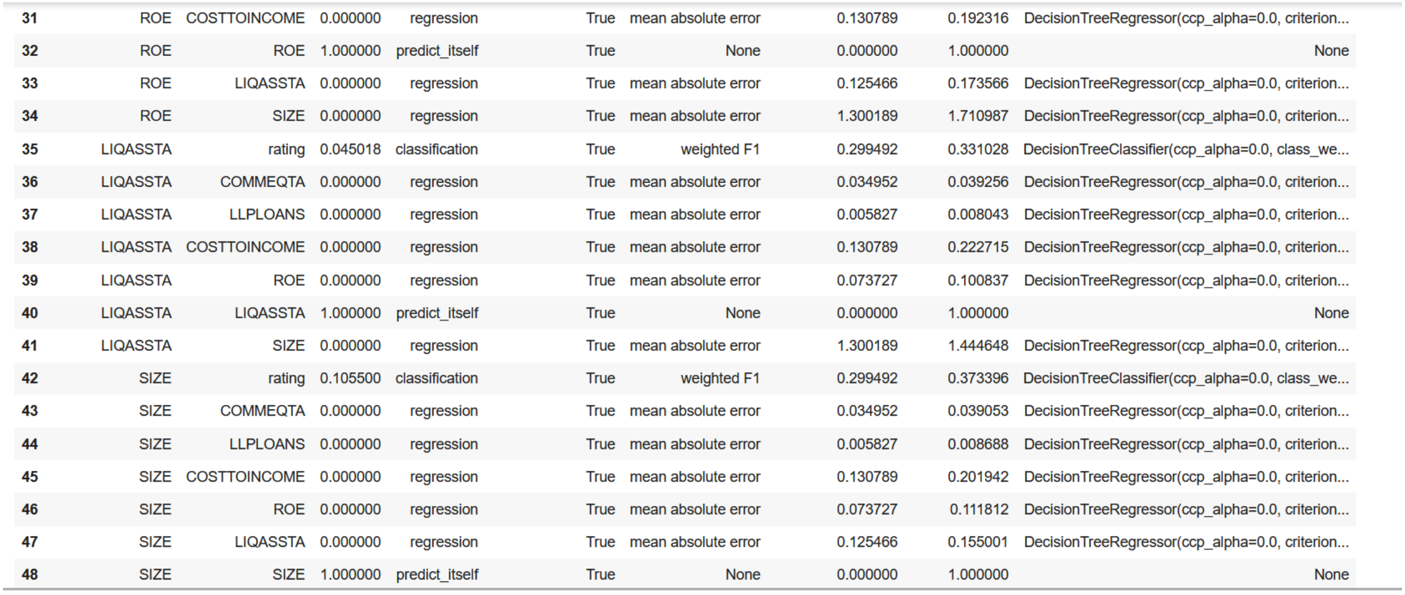
PPS CHART -



	x	y	ppscore	case	is_valid_score	metric	baseline_score	model_score	model
0	rating	rating	1.000000	predict_itself	True	None	0.000000	1.000000	None
1	rating	COMMEQTA	0.031121	regression	True	mean absolute error	0.034952	0.033864	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
2	rating	LLPLOANS	0.000000	regression	True	mean absolute error	0.005827	0.007051	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
3	rating	COSTTOINCOME	0.000000	regression	True	mean absolute error	0.130789	0.134742	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
4	rating	ROE	0.000000	regression	True	mean absolute error	0.073727	0.075118	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
5	rating	LIQASSTA	0.043489	regression	True	mean absolute error	0.125466	0.120009	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
6	rating	SIZE	0.162042	regression	True	mean absolute error	1.300189	1.089505	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
7	COMMEQTA	rating	0.105826	classification	True	weighted F1	0.299492	0.373624	DecisionTreeClassifier(ccp_alpha=0.0, class_we...
8	COMMEQTA	COMMEQTA	1.000000	predict_itself	True	None	0.000000	1.000000	None
9	COMMEQTA	LLPLOANS	0.000000	regression	True	mean absolute error	0.005827	0.008266	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
10	COMMEQTA	COSTTOINCOME	0.000000	regression	True	mean absolute error	0.130789	0.197847	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
11	COMMEQTA	ROE	0.000000	regression	True	mean absolute error	0.073727	0.109831	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
12	COMMEQTA	LIQASSTA	0.000000	regression	True	mean absolute error	0.125466	0.143843	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
13	COMMEQTA	SIZE	0.000000	regression	True	mean absolute error	1.300189	1.589678	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
14	LLPLOANS	rating	0.091377	classification	True	weighted F1	0.299492	0.363503	DecisionTreeClassifier(ccp_alpha=0.0, class_we...
15	LLPLOANS	COMMEQTA	0.000000	regression	True	mean absolute error	0.034952	0.039368	DecisionTreeRegressor(ccp_alpha=0.0, criterion...



16	LLPLOANS	LLPLOANS	1.000000	predict_itself	True	None	0.000000	1.000000	None
17	LLPLOANS	COSTTOINCOME	0.000000	regression	True	mean absolute error	0.130789	0.186743	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
18	LLPLOANS	ROE	0.000000	regression	True	mean absolute error	0.073727	0.096023	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
19	LLPLOANS	LIQASSTA	0.000000	regression	True	mean absolute error	0.125466	0.160313	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
20	LLPLOANS	SIZE	0.000000	regression	True	mean absolute error	1.300189	1.583714	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
21	COSTTOINCOME	rating	0.045354	classification	True	weighted F1	0.299492	0.331264	DecisionTreeClassifier(ccp_alpha=0.0, class_we...
22	COSTTOINCOME	COMMEQTA	0.000000	regression	True	mean absolute error	0.034952	0.041335	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
23	COSTTOINCOME	LLPLOANS	0.000000	regression	True	mean absolute error	0.005827	0.009269	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
24	COSTTOINCOME	COSTTOINCOME	1.000000	predict_itself	True	None	0.000000	1.000000	None
25	COSTTOINCOME	ROE	0.000000	regression	True	mean absolute error	0.073727	0.102810	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
26	COSTTOINCOME	LIQASSTA	0.000000	regression	True	mean absolute error	0.125466	0.164010	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
27	COSTTOINCOME	SIZE	0.000000	regression	True	mean absolute error	1.300189	1.913440	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
28	ROE	rating	0.008444	classification	True	weighted F1	0.299492	0.305407	DecisionTreeClassifier(ccp_alpha=0.0, class_we...
29	ROE	COMMEQTA	0.000000	regression	True	mean absolute error	0.034952	0.047788	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
30	ROE	LLPLOANS	0.000000	regression	True	mean absolute error	0.005827	0.007393	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
31	ROE	COSTTOINCOME	0.000000	regression	True	mean absolute error	0.130789	0.192316	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
32	ROE	ROE	1.000000	predict_itself	True	None	0.000000	1.000000	None

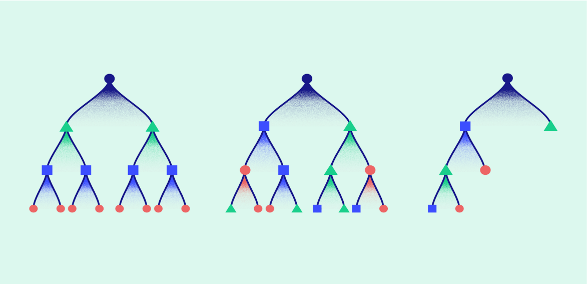


31	ROE	COSTTOINCOME	0.000000	regression	True	mean absolute error	0.130789	0.192316	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
32	ROE	ROE	1.000000	predict_itself	True	None	0.000000	1.000000	None
33	ROE	LIQASSTA	0.000000	regression	True	mean absolute error	0.125466	0.173566	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
34	ROE	SIZE	0.000000	regression	True	mean absolute error	1.300189	1.710987	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
35	LIQASSTA	rating	0.045018	classification	True	weighted F1	0.299492	0.331028	DecisionTreeClassifier(ccp_alpha=0.0, class_we...
36	LIQASSTA	COMMEQTA	0.000000	regression	True	mean absolute error	0.034952	0.039256	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
37	LIQASSTA	LLPLOANS	0.000000	regression	True	mean absolute error	0.005827	0.008043	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
38	LIQASSTA	COSTTOINCOME	0.000000	regression	True	mean absolute error	0.130789	0.222715	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
39	LIQASSTA	ROE	0.000000	regression	True	mean absolute error	0.073727	0.100837	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
40	LIQASSTA	LIQASSTA	1.000000	predict_itself	True	None	0.000000	1.000000	None
41	LIQASSTA	SIZE	0.000000	regression	True	mean absolute error	1.300189	1.444648	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
42	SIZE	rating	0.105500	classification	True	weighted F1	0.299492	0.373396	DecisionTreeClassifier(ccp_alpha=0.0, class_we...
43	SIZE	COMMEQTA	0.000000	regression	True	mean absolute error	0.034952	0.039053	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
44	SIZE	LLPLOANS	0.000000	regression	True	mean absolute error	0.005827	0.008688	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
45	SIZE	COSTTOINCOME	0.000000	regression	True	mean absolute error	0.130789	0.201942	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
46	SIZE	ROE	0.000000	regression	True	mean absolute error	0.073727	0.111812	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
47	SIZE	LIQASSTA	0.000000	regression	True	mean absolute error	0.125466	0.155001	DecisionTreeRegressor(ccp_alpha=0.0, criterion...
48	SIZE	SIZE	1.000000	predict_itself	True	None	0.000000	1.000000	None

Model Building -

After visualizing and preprocessing the data, we split the data into training and testing data using stratified sampling, splitting it in a 4:1 ratio. We used the training dataset for training 11 models: XGBoost, Gradient Boosted Tree Regression model, Random forest, Support vector machine, Multilayer perceptron, Gaussian Naive bayes, latent dirichlet allocation, Qualitative Data Analysis, K-Nearest Neighbor, Linear regression, Decision tree classifier, Decision tree regressor.

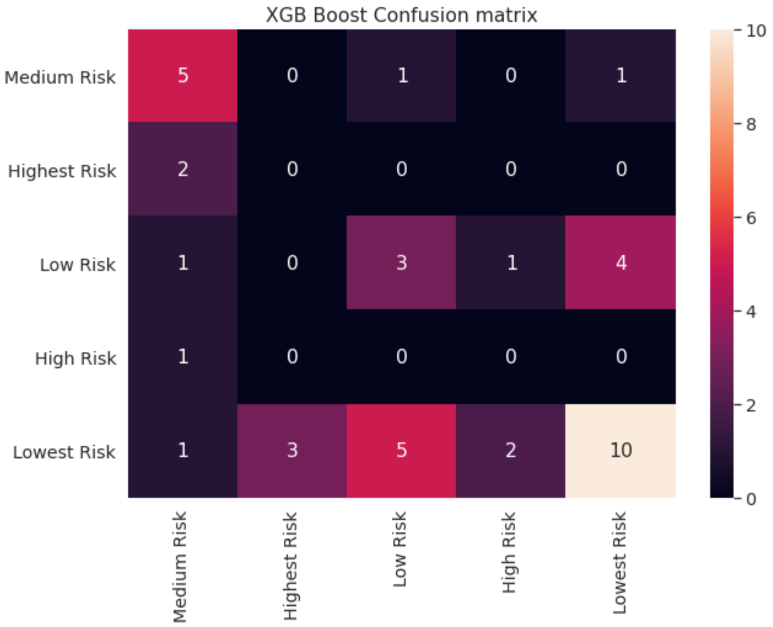
1. XGBoost -



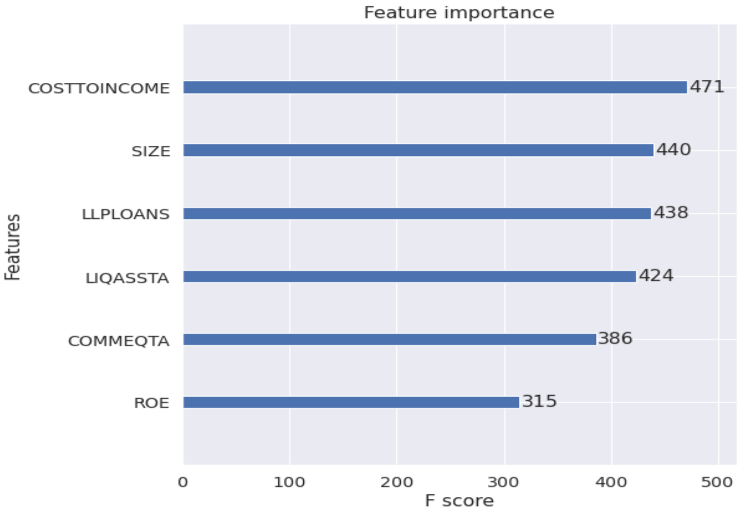
XGB is a popular supervised learning algorithm. It is used in supervised learning in ML. It is an additive and sequential model which converts weak learners into stronger ones by adding weights to them.

XGB	0.45

Accuracy:	



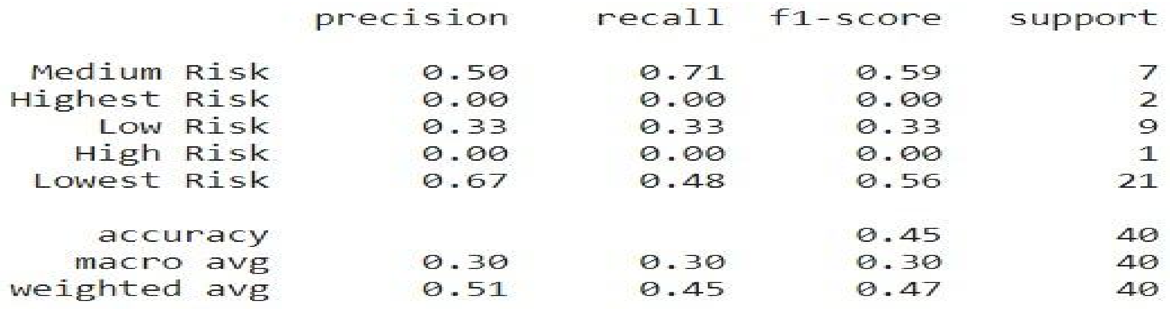
	TP	FP	TN	FN
Medium risk	5	2	28	5
Highest risk	0	2	35	3
Low risk	3	6	25	6
High risk	0	1	36	3
Lowest risk	10	11	14	5



- Operating costs to operating income has the highest value of f-score which shows it’s importance in XBG model. Cost to income is the most important variable towards rating prediction.

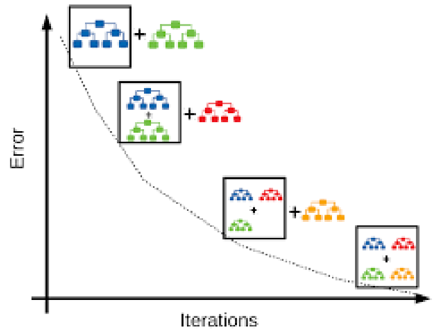
- Size of the company is the 2nd most important variable followed by LLPLOANS, COMMEQTA AND ROE.

Classification report

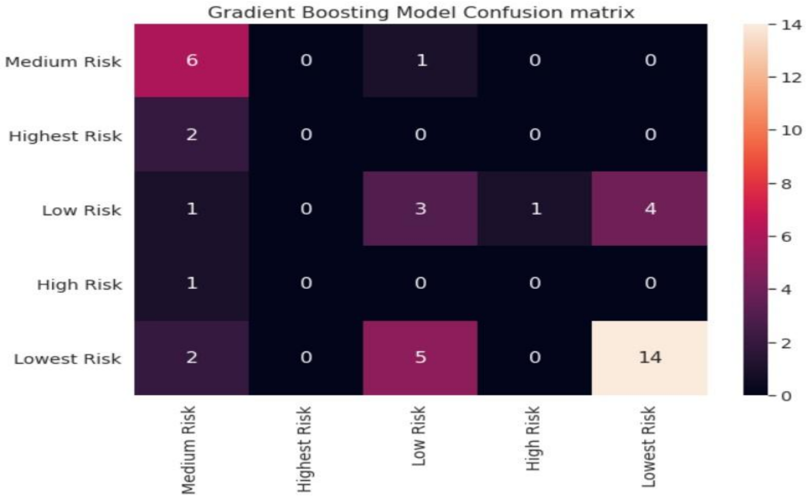


```
              precision    recall  f1-score   support

 Medium Risk       0.50      0.71      0.59         7
Highest Risk       0.00      0.00      0.00         2
    Low Risk       0.33      0.33      0.33         9
   High Risk       0.00      0.00      0.00         1
 Lowest Risk       0.67      0.48      0.56        21

    accuracy                           0.45        40
   macro avg       0.30      0.30      0.30        40
weighted avg       0.51      0.45      0.47        40
```

2. GBT -



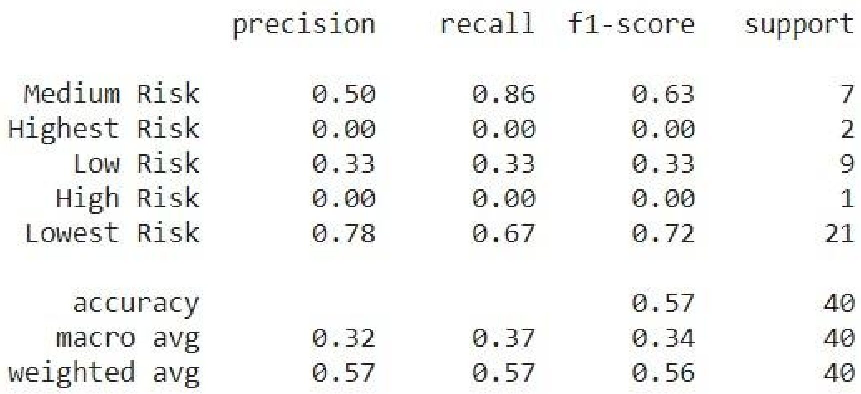
Gradient boosted trees is a learning algorithm for regression. It can be used for classification problems. Since corporate credit rating too is a type of classification problem thus we used GBT to predict the ratings.


Gradient Boosting Model Confusion matrix

	Medium Risk	Highest Risk	Low Risk	High Risk	Lowest Risk
Medium Risk	6	0	1	0	0
Highest Risk	2	0	0	0	0
Low Risk	1	0	3	1	4
High Risk	1	0	0	0	0
Lowest Risk	2	0	5	0	14

GBT Accuracy:	0.575

Classification report



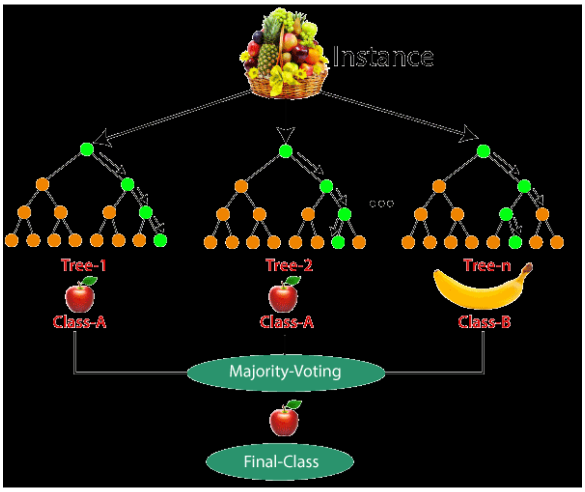
```
               precision    recall  f1-score   support

  Medium Risk       0.50      0.86      0.63         7
 Highest Risk       0.00      0.00      0.00         2
     Low Risk       0.33      0.33      0.33         9
    High Risk       0.00      0.00      0.00         1
  Lowest Risk       0.78      0.67      0.72        21

     accuracy                           0.57        40
    macro avg       0.32      0.37      0.34        40
 weighted avg       0.57      0.57      0.56        40
```

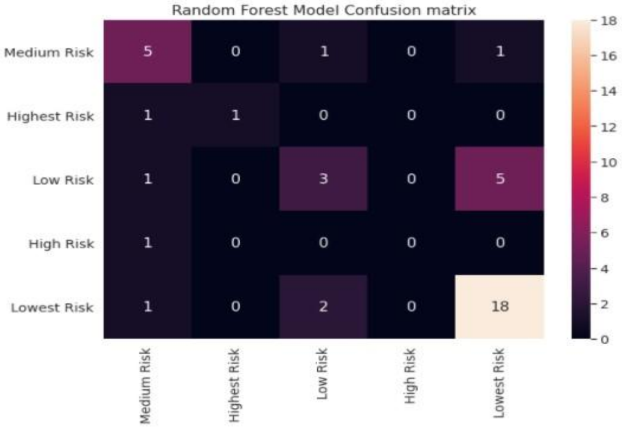
3 . Random-Forest (RF) -

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operate by constructing a multitude of decision trees at training time.RF builds multiple decision trees and merges them together to get a more accurate and stable prediction

Since we had to predict the rating like many other algorithms we used this to predict the ratings.

We got maximum accuracy for the RF model.

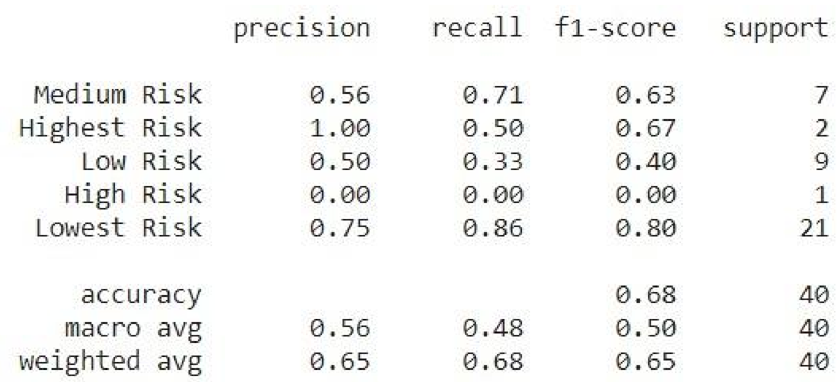




	Medium Risk	Highest Risk	Low Risk	High Risk	Lowest Risk
Medium Risk	5	0	1	0	1
Highest Risk	1	1	0	0	0
Low Risk	1	0	3	0	5
High Risk	1	0	0	0	0
Lowest Risk	1	0	2	0	18

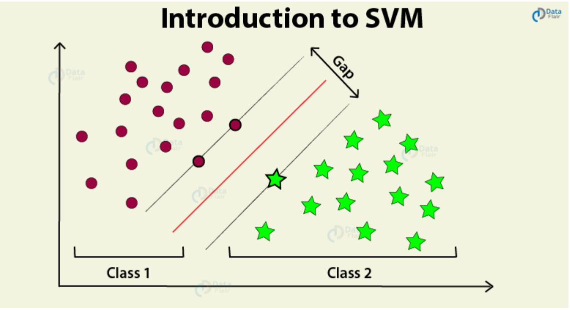
RF Accuracy:	**0.675**

Classification report

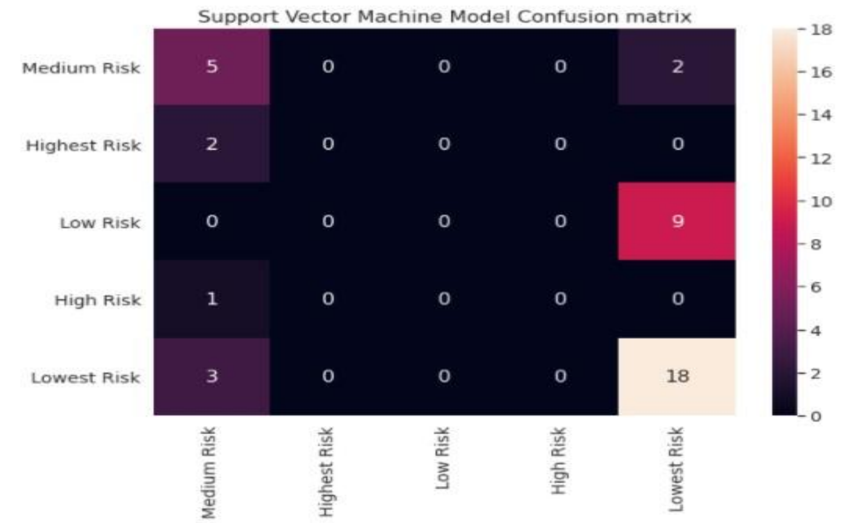


```
               precision    recall  f1-score   support

  Medium Risk       0.56      0.71      0.63         7
 Highest Risk       1.00      0.50      0.67         2
     Low Risk       0.50      0.33      0.40         9
    High Risk       0.00      0.00      0.00         1
  Lowest Risk       0.75      0.86      0.80        21

     accuracy                           0.68        40
    macro avg       0.56      0.48      0.50        40
 weighted avg       0.65      0.68      0.65        40
```

4. Support vector machine(SVM) -

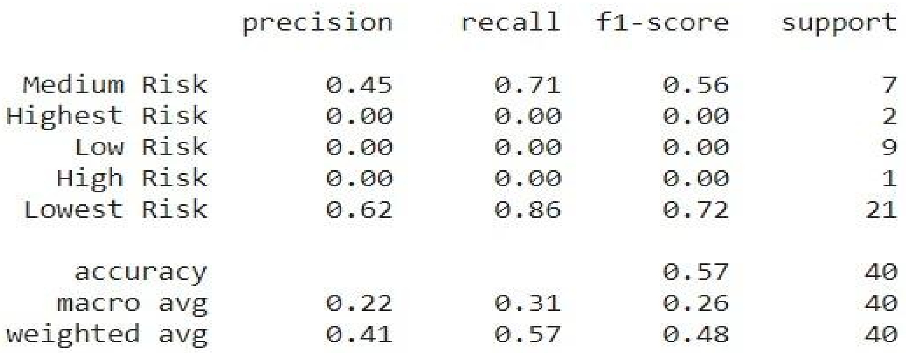


SVM algorithms too can be used for classification problems. SVM works by finding a hyperplane in N-dimensional space that distinctly classifies the data points.



SVM Accuracy:	**0.575**

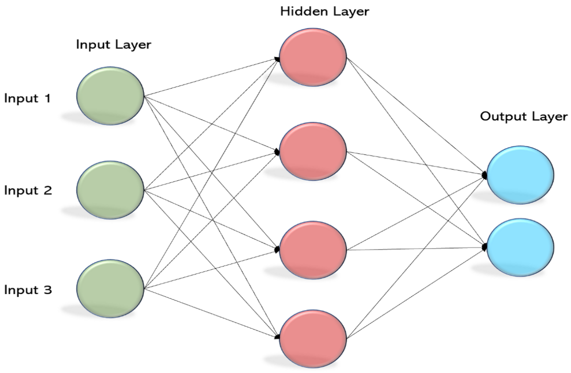
Classification report



```
               precision    recall  f1-score   support

  Medium Risk       0.45      0.71      0.56         7
 Highest Risk       0.00      0.00      0.00         2
     Low Risk       0.00      0.00      0.00         9
    High Risk       0.00      0.00      0.00         1
  Lowest Risk       0.62      0.86      0.72        21

     accuracy                           0.57        40
    macro avg       0.22      0.31      0.26        40
 weighted avg       0.41      0.57      0.48        40
```

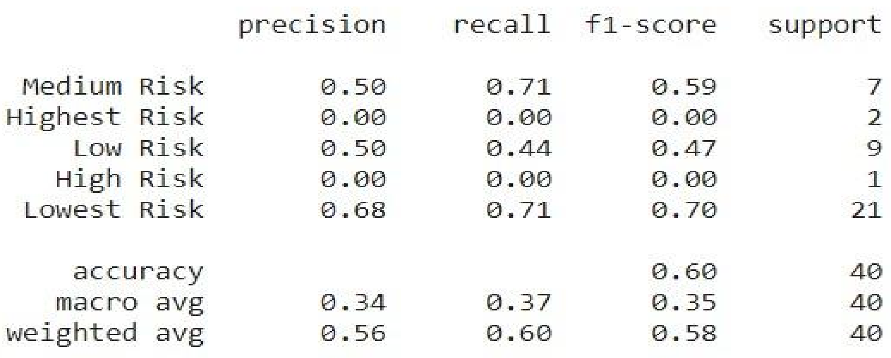
5. Multi-layer Perceptron Classifier (Neural Network) -

The multilayer perceptron (MLP) is a feedforward artificial neural network model that maps input data sets to a set of appropriate outputs. An MLP consists of multiple layers and each layer is fully connected to the following one. The nodes of the layers are neurons with nonlinear activation functions, except for the nodes of the input layer. Between the input and the output layer, there may be one or more nonlinear hidden layers.



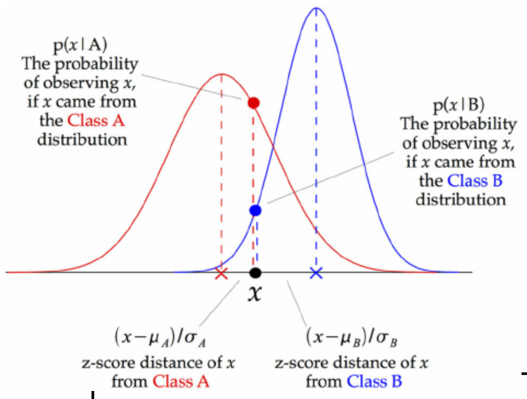
Activation function used - f(x) = tanh(x)

MLP Accuracy:	**0.6**



```
               precision    recall  f1-score   support

 Medium Risk        0.50      0.71      0.59         7
Highest Risk        0.00      0.00      0.00         2
    Low Risk        0.50      0.44      0.47         9
   High Risk        0.00      0.00      0.00         1
 Lowest Risk        0.68      0.71      0.70        21

    accuracy                            0.60        40
   macro avg        0.34      0.37      0.35        40
weighted avg        0.56      0.60      0.58        40
```

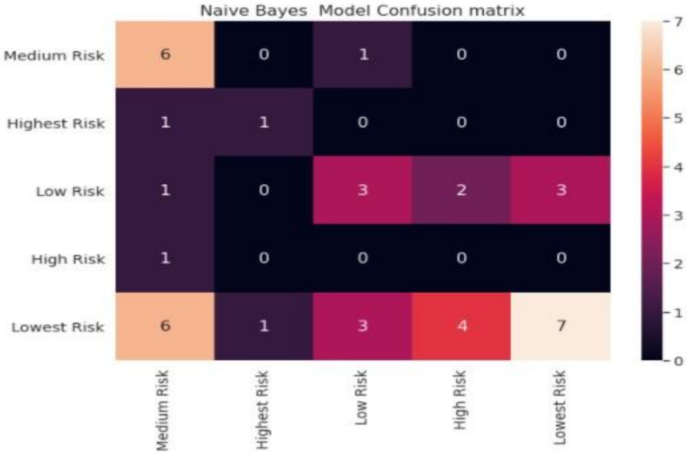
6. Gaussian Naive Bayes Classifier (GNB) -


p(x | A)
The probability of observing x, if x came from the Class A distribution
p(x | B)
The probability of observing x, if x came from the Class B distribution
x
$(x-\mu_A)/\sigma_A$
z-score distance of x from Class A
$(x-\mu_B)/\sigma_B$
z-score distance of x from Class B

Gaussian Naive Bayes is a variant of Naive Bayes that follows Gaussian normal distribution and supports continuous data. GNB is a basic classification algorithm and we used it just to check how it performs on the given dataset.

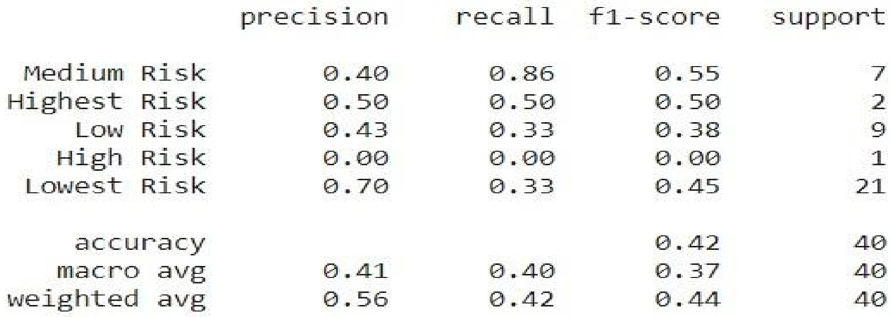
GNB	**0.425**

Accuracy:	



	Medium Risk	Highest Risk	Low Risk	High Risk	Lowest Risk
Medium Risk	6	0	1	0	0
Highest Risk	1	1	0	0	0
Low Risk	1	0	3	2	3
High Risk	1	0	0	0	0
Lowest Risk	6	1	3	4	7

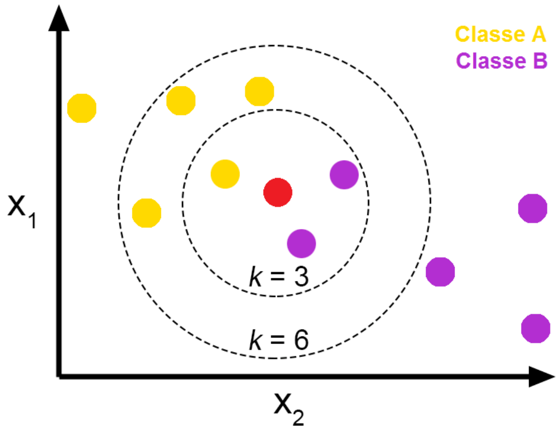
Classification report

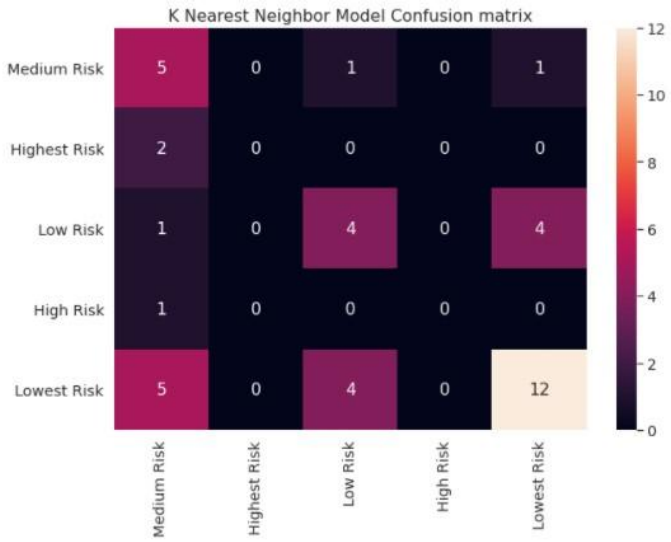


```
              precision    recall  f1-score   support

 Medium Risk       0.40      0.86      0.55         7
Highest Risk       0.50      0.50      0.50         2
    Low Risk       0.43      0.33      0.38         9
   High Risk       0.00      0.00      0.00         1
 Lowest Risk       0.70      0.33      0.45        21

    accuracy                           0.42        40
   macro avg       0.41      0.40      0.37        40
weighted avg       0.56      0.42      0.44        40
```

7. K-Nearest Neighbours (KNN) -

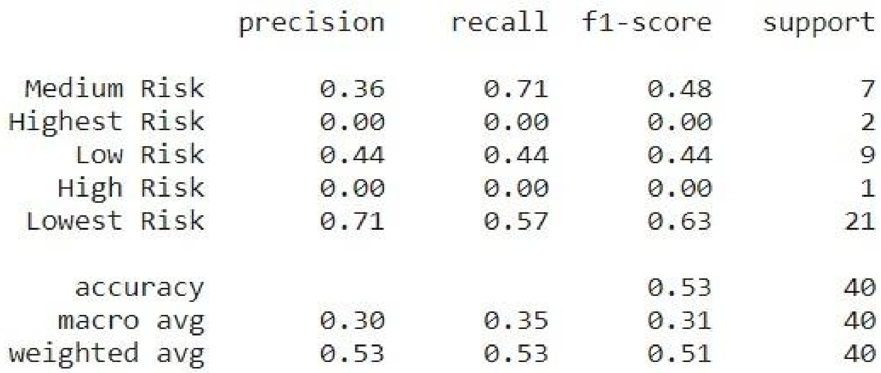
The k-nearest neighbors (KNN) algorithm is a simple, supervised machine learning algorithm that can be used to solve both classification and regression problems. It's easy to implement and understand but has the major drawback of becoming significantly slower as the size of that data in use grows. We have specified to use 7 neighbors as default.





KNN Accuracy:	0.525

Classification report



```
              precision    recall  f1-score   support

 Medium Risk       0.36      0.71      0.48         7
Highest Risk       0.00      0.00      0.00         2
    Low Risk       0.44      0.44      0.44         9
   High Risk       0.00      0.00      0.00         1
 Lowest Risk       0.71      0.57      0.63        21

    accuracy                           0.53        40
   macro avg       0.30      0.35      0.31        40
weighted avg       0.53      0.53      0.51        40
```

8. Logistic Regression Model (LR) -

Logistic regression is a process of modeling the probability of a discrete outcome given an input variable. Logistic regression is a useful analysis method for classification problems, where you are trying to determine if a new sample fits best into a category.
We used Multinomial Logistic Regression as it can model scenarios where there are more than two possible discrete outcomes. In the algorithm, we used a newton-cg solver which uses newton's method to compute the second derivatives.

LR Accuracy:	0.6

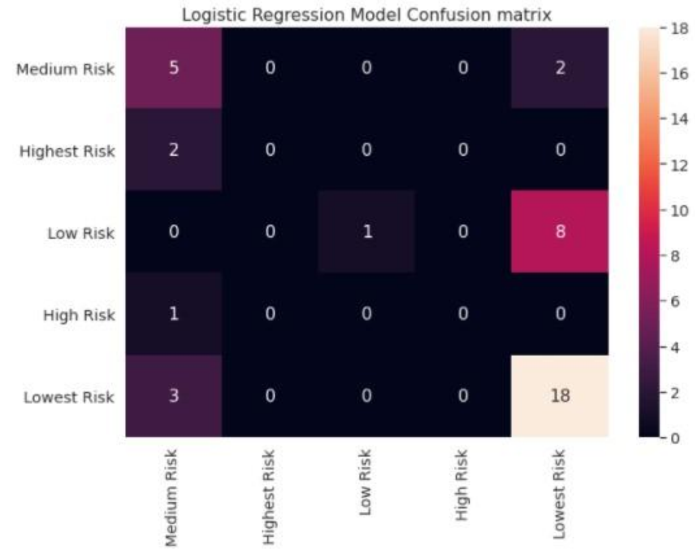
Limitations of Logistic Regression

Logistic regression is a simple and powerful linear classification algorithm. It also has limitations that suggest the need for alternate linear classification algorithms.

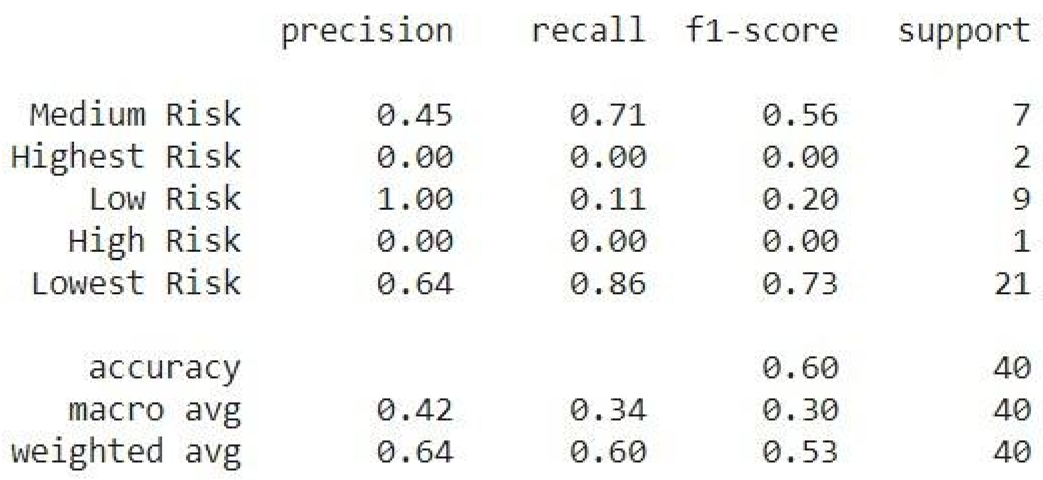
- Two-Class Problems. Logistic regression is intended for two-class or binary classification problems. It can be extended for multi-class classification but is rarely used for this purpose.

- Unstable With Well Separated Classes. Logistic regression can become unstable when the classes are well separated.
- Unstable With Few Examples. Logistic regression can become unstable when there are few examples from which to estimate the parameters.

Linear Discriminant Analysis does address each of these points and is the go-to linear method for multi-class classification problems.

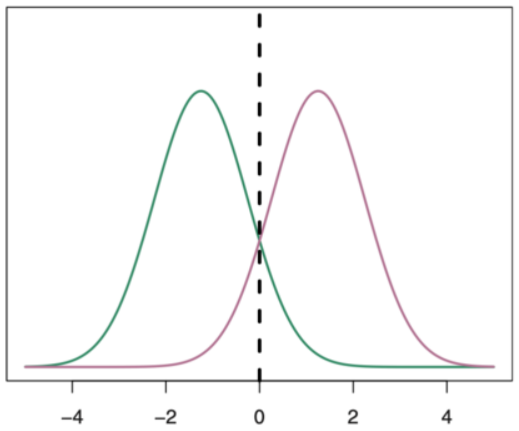


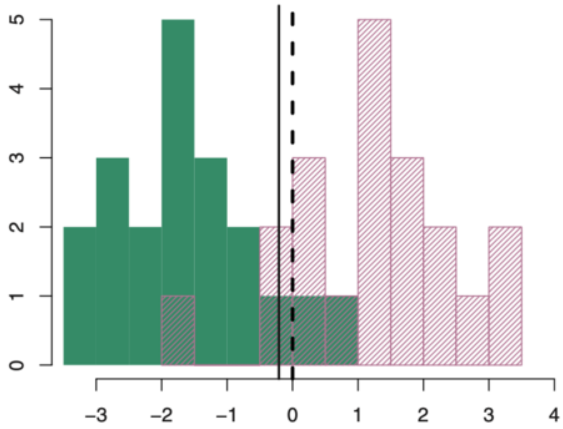
Classification report



```
              precision    recall  f1-score   support

 Medium Risk       0.45      0.71      0.56         7
Highest Risk       0.00      0.00      0.00         2
    Low Risk       1.00      0.11      0.20         9
   High Risk       0.00      0.00      0.00         1
 Lowest Risk       0.64      0.86      0.73        21

    accuracy                           0.60        40
   macro avg       0.42      0.34      0.30        40
weighted avg       0.64      0.60      0.53        40
```

9.**Linear Discriminant Analysis** :



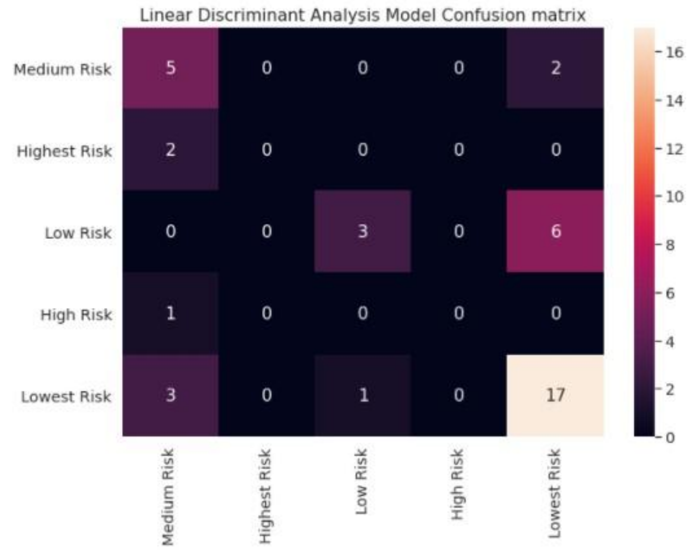


The representation of LDA is straightforward. It consists of statistical properties of your data, calculated for each class. For a single input variable (x) this is the mean and the variance of the variable for each class. For multiple variables, this is the same properties calculated over the multivariate Gaussian, namely the means and the covariance matrix.

LDA makes some simplifying assumptions about data:

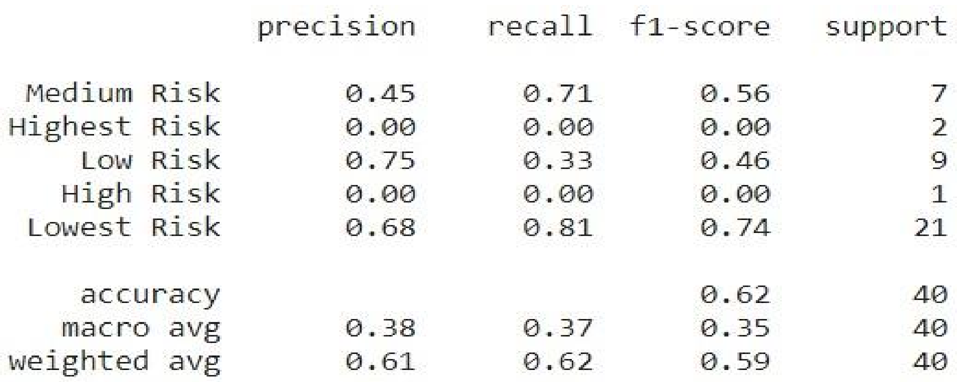
1. That your data is Gaussian, that each variable is shaped like a bell curve when plotted.
2. That each attribute has the same variance, that values of each variable vary around the mean by the same amount on average.

Since These Assumptions were really close to what we had Inferred from data analysis we Decided to Implement the model and it ended up being one of the best performers out of the bunch.


Linear Discriminant Analysis Model Confusion matrix

	Medium Risk	Highest Risk	Low Risk	High Risk	Lowest Risk
Medium Risk	5	0	0	0	2
Highest Risk	2	0	0	0	0
Low Risk	0	0	3	0	6
High Risk	1	0	0	0	0
Lowest Risk	3	0	1	0	17

Classification report



```
               precision    recall  f1-score   support

  Medium Risk       0.45      0.71      0.56         7
 Highest Risk       0.00      0.00      0.00         2
     Low Risk       0.75      0.33      0.46         9
    High Risk       0.00      0.00      0.00         1
  Lowest Risk       0.68      0.81      0.74        21

     accuracy                           0.62        40
    macro avg       0.38      0.37      0.35        40
 weighted avg       0.61      0.62      0.59        40
```

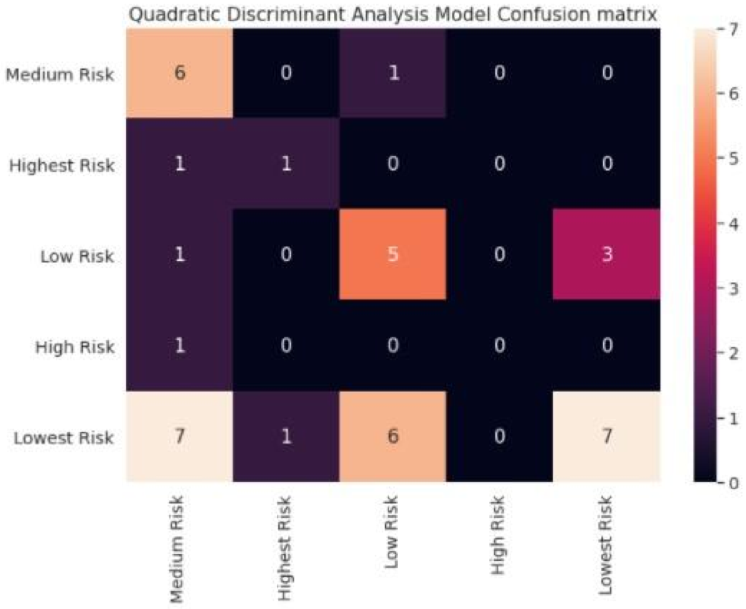
10.**Quadratic Discriminant Analysis** :

Quadratic Discriminant Analysis (QDA) is similar to LDA based on the fact that there is an assumption of the observations being drawn from a normal distribution. The difference is that QDA assumes that each class has its own covariance matrix, while LDA does not.

The QDA makes these assumptions about the data :

- Observation of each class is drawn from a normal distribution (same as LDA).
- QDA assumes that each class has its own covariance matrix (Different from LDA).

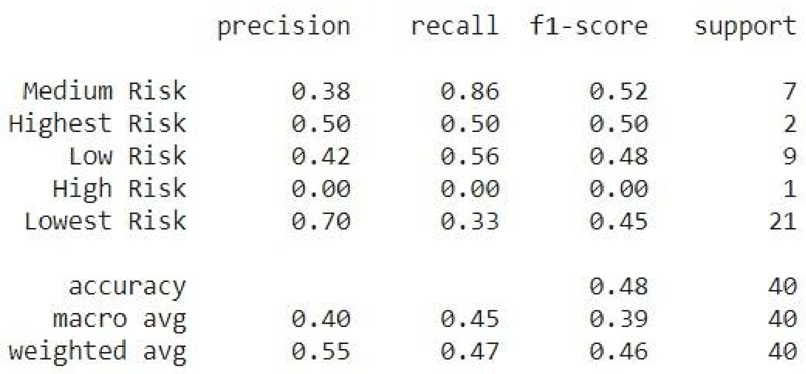
In conclusion, LDA is less flexible than QDA because we have to estimate fewer parameters. This can be good when we have only a few observations in our training data which was the case with us.



Quadratic Discriminant Analysis Model Confusion matrix

	Medium Risk	Highest Risk	Low Risk	High Risk	Lowest Risk
Medium Risk	6	0	1	0	0
Highest Risk	1	1	0	0	0
Low Risk	1	0	5	0	3
High Risk	1	0	0	0	0
Lowest Risk	7	1	6	0	7

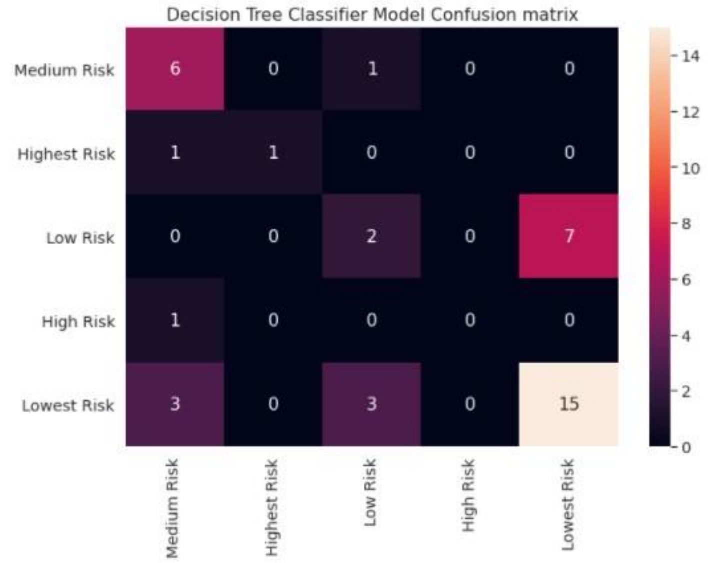
Classification report



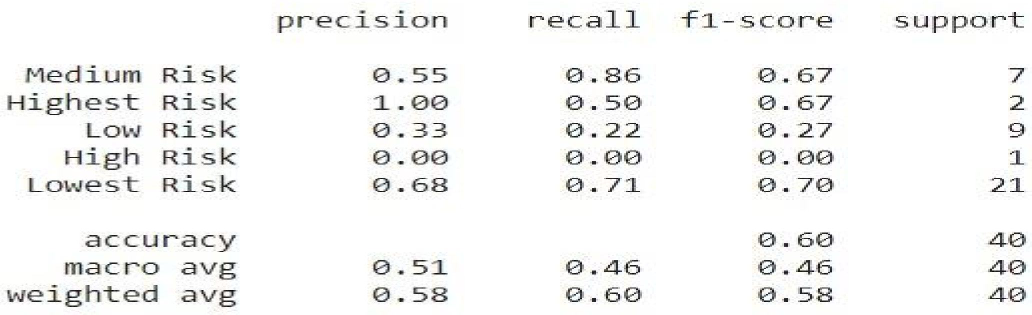
```
              precision    recall  f1-score   support

 Medium Risk       0.38      0.86      0.52         7
Highest Risk       0.50      0.50      0.50         2
    Low Risk       0.42      0.56      0.48         9
   High Risk       0.00      0.00      0.00         1
 Lowest Risk       0.70      0.33      0.45        21

    accuracy                           0.48        40
   macro avg       0.40      0.45      0.39        40
weighted avg       0.55      0.47      0.46        40
```

11.<u>Decision Tree Classifier</u> :

A decision tree is a flowchart-like tree structure where an internal node represents a feature(or attribute), the branch represents a decision rule, and each leaf node represents the outcome. The topmost node in a decision tree is known as the root node. It learns to partition on the basis of the attribute value. It partitions the tree in a recursive manner call recursive partitioning. This flowchart-like structure helps you in decision-making. It's visualization like a flowchart diagram that easily mimics human-level thinking. That is why decision trees are easy to understand and interpret.

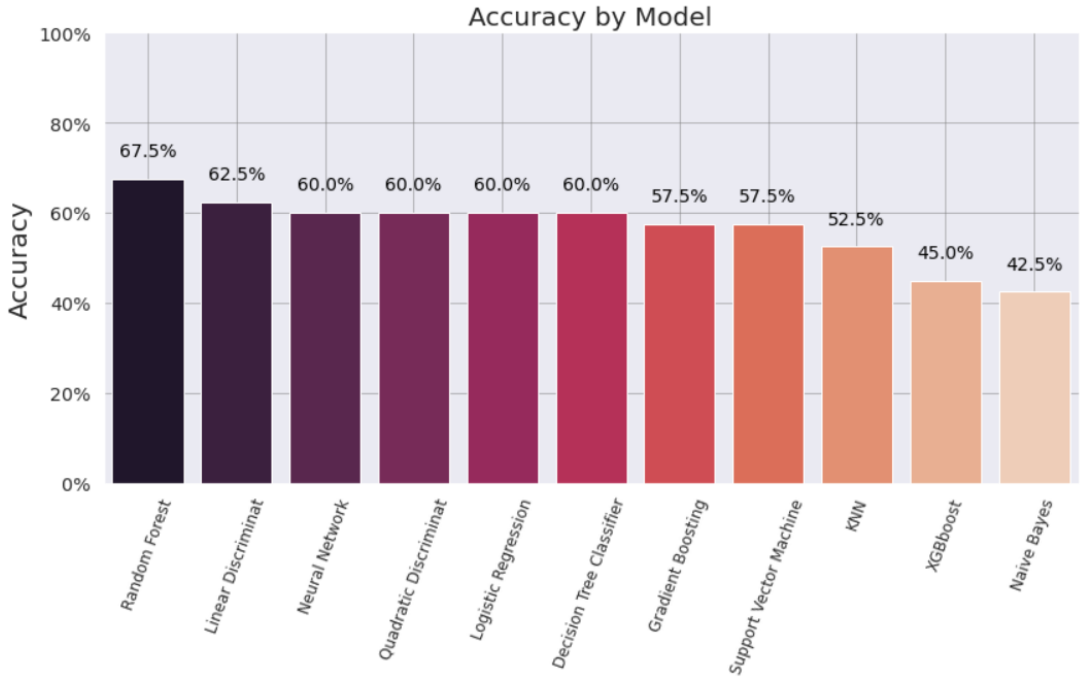


Classification report



```
              precision    recall  f1-score   support

 Medium Risk       0.55      0.86      0.67         7
Highest Risk       1.00      0.50      0.67         2
    Low Risk       0.33      0.22      0.27         9
   High Risk       0.00      0.00      0.00         1
 Lowest Risk       0.68      0.71      0.70        21

    accuracy                           0.60        40
   macro avg       0.51      0.46      0.46        40
weighted avg       0.58      0.60      0.58        40
```

Conclusion -



- **XGB Accuracy: 0.45**
- **GBT Accuracy: 0.575**
- **RF Accuracy: 0.675**
- **SVM Accuracy: 0.575**
- **MLP Accuracy: 0.6**
- **GNB Accuracy: 0.425**
- **KNN Accuracy: 0.525**
- **LR Accuracy: 0.6**
- **LDA Accuracy: 0.625**
- **QDA Accuracy: 0.475**
- **Decision Tree classifier: 0.6**

Since Random-Forest/Decision-Trees Does not inherently make any and/or require any Correlation between attributes as the decision tree is a distribution-free or non-parametric method. Instead, it deploys a greedy strategy that instead looks for the best dividing metric possible at that given step, it also is relatively less data-hungry than some of the other models that we used.

Due to the above-mentioned reasons, the Random Forests ended up performing better than every other model. Followed closely by, Decision Trees, LDA, Neural nets, and QDA.