HEART FAILURE PREDICTION USING MACHINE LEARNING UML501

SNO.	Name	Roll Number
1	Samarth Babbar	102003180
2	Harsh Joshi	102003192
3	Gowrang ujjainia	102003203

Submitted to Dr. Harpreet Singh

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Department of Computer Science and Engineering

THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY, (A DEEMED TO BE UNIVERSITY), PATIALA, PUNJAB,INDIA

Dataset Link: https://archive.ics.uci.edu/ml/datasets/Heart+failure+clinical+records

About the dataset used: Cardiovascular diseases (CVDs) are the number 1 cause of death globally, taking an estimated 17.9 million lives each year, which accounts for 31% of all deaths worldwide.Heart failure is a common event caused by CVDs and this dataset contains 12 features that can be used to predict mortality by heart failure.

Most cardiovascular diseases can be prevented by addressing behavioral risk factors such as tobacco use, unhealthy diet and obesity, physical inactivity and harmful use of alcohol using population-wide strategies.

People with cardiovascular disease or who are at high cardiovascular risk (due to the presence of one or more risk factors such as hypertension, diabetes, hyperlipidaemia or other established disease) need early detection and management wherein a machine learning model can be of great help.

Description Table: It is a reference to Meanings, measurement units, and intervals of each feature of the dataset.

S.NO	FEATURES	DESCRIPTION AND DATA TYPES	RANGE
1	Age	Age of the patient(years)	[45-90]
2	Anaemia	Decrease of red blood cells or hemoglobin (Boolean)	[0,1]
3	High blood pressure	If the patient has hypertension (Boolean)	[0,1]
4	Creatinine phosphokinase	Level of the CPK enzyme in the blood (mcg/L)	[23-7861]
5	Diabetes	If the patient has diabetes (Boolean)	[0,1]
6	Ejection Fraction	Percentage of blood leaving the heart at each contraction	[14-80]
7	Platelets	Platelets in the blood (kiloplatelets/mL)	[25.01-850.00]
8	Sex	Woman or Man (Binary)	[0,1]
9	Serum creatinine	Level of serum creatinine in the blood (mg/dL)	[0.50-9.40]
10	Serum sodium	Level of serum sodium in the blood (mEq/L)	[114-148]
11	Smoking	If the patient smokes or not (Boolean)	[0,1]
12	Time	Follow up period (days)	[4,285]
13	Death event [target]	If the patient deceased during the follow up period (Boolean)	[0,1]

Fig 1.0 -> description table

Data PreProcessing

Analyzing output feature of the dataset in the form of count plot

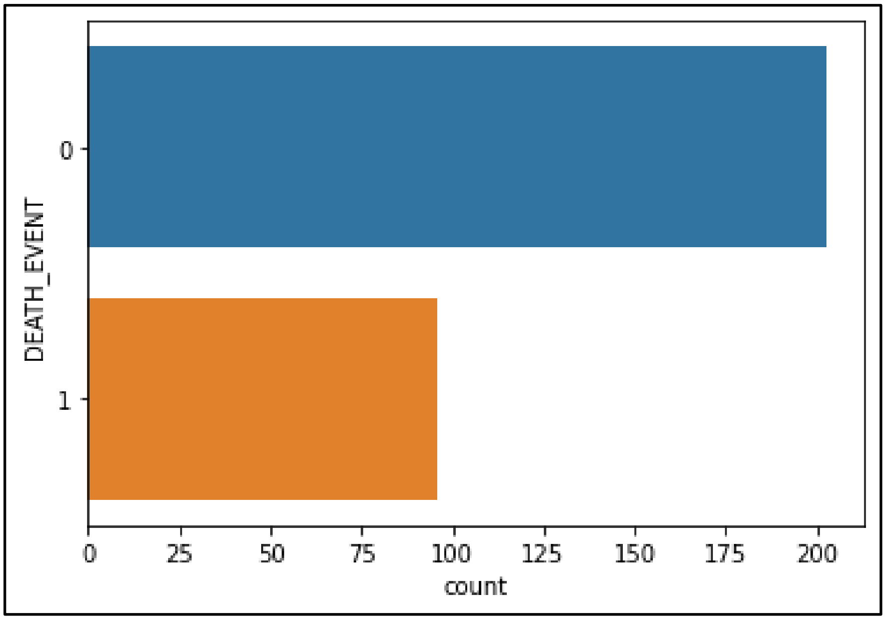


Fig 1.1

Count of **Null Values** for each feature in the dataset

age	0
anaemia	0
creatinine_phosphokinase	0
diabetes	0
ejection_fraction	0
high_blood_pressure	0
platelets	0
serum_creatinine	0
serum_sodium	0
sex	0
smoking	0
time	0
DEATH_EVENT	0

Fig 1.2

Dataset Tendency - For analyzing outliers in the dataset

Feature	Count	Mean	Std	Min	25%	50%	75%	Max
age	299	60.83389	11.89481	40	51	60	70	95
anaemia	299	0.431438	0.496107	0	0	0	1	1
creatinine_phosphokinase	299	581.8395	970.2879	23	116.5	250	582	7861
diabetes	299	0.41806	0.494067	0	0	0	1	1
ejection_fraction	299	38.08361	11.83484	14	30	38	45	80
high_blood_pressure	299	0.351171	0.478136	0	0	0	1	1
platelets	299	263358	97804.24	25100	212500	262000	303500	850000
serum_creatinine	299	1.39388	1.03451	0.5	0.9	1.1	1.4	9.4
serum_sodium	299	136.6254	4.412477	113	134	137	140	148
sex	299	0.648829	0.478136	0	0	1	1	1
smoking	299	0.32107	0.46767	0	0	0	1	1
time	299	130.2609	77.61421	4	73	115	203	285
DEATH EVENT	299	0.32107	0.46767	0	0	0	1	1

Fig 1.3

Data Plot in form of Histogram - It is used to illustrate the major features of the distribution of the data in a convenient form.

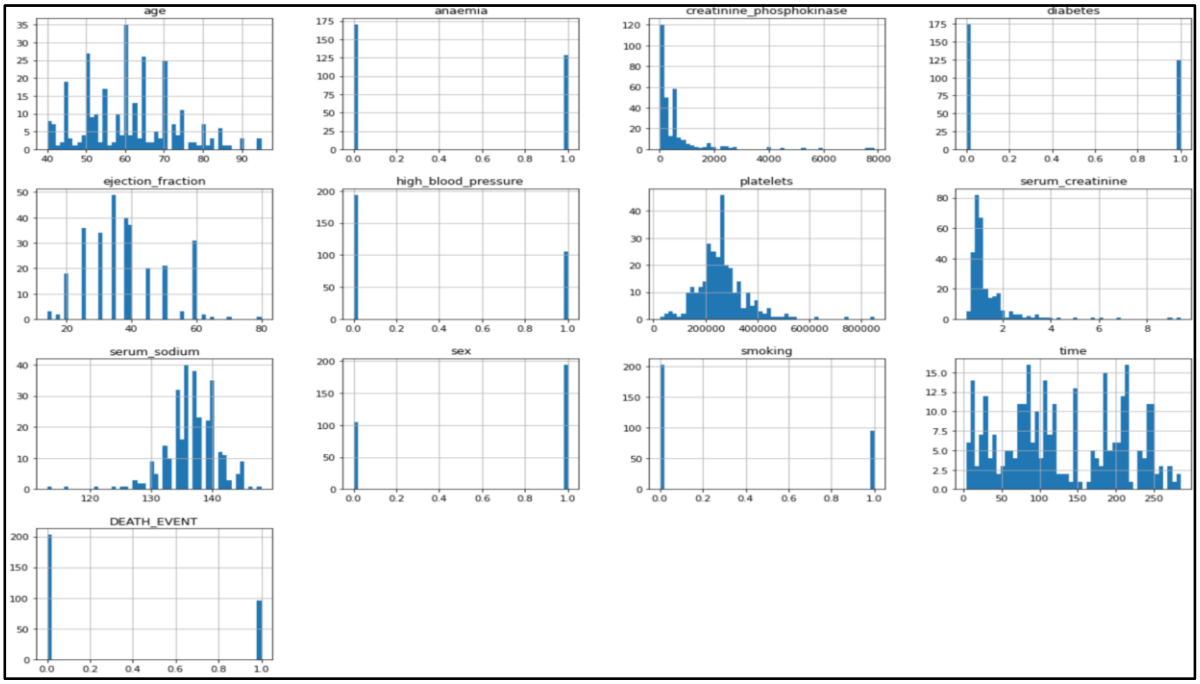


Fig 1.4

Density Plot - The Density Plot shows the smoothed distribution of the points along the numeric axis. The peaks of the density plot are at the locations where there is the highest concentration of points.

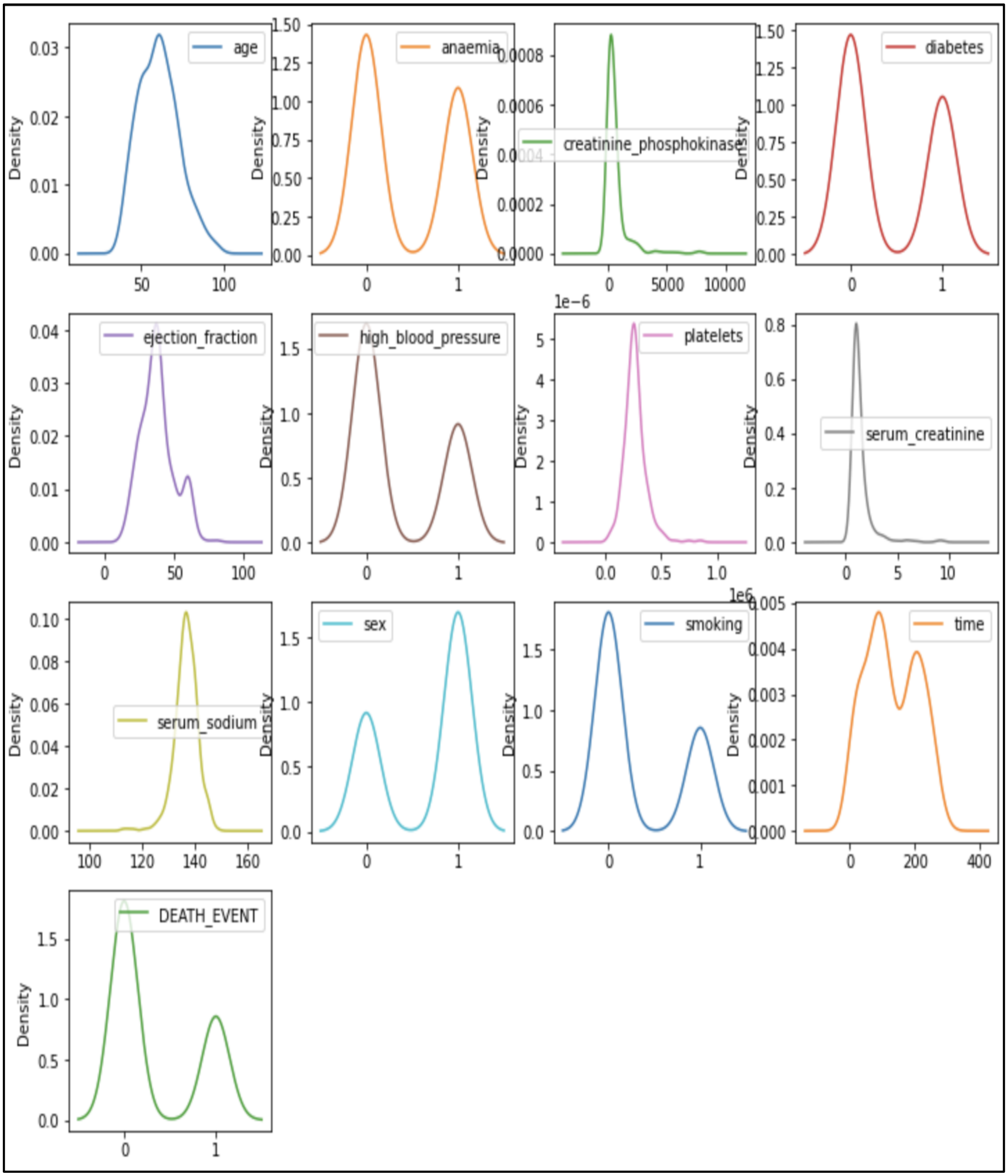
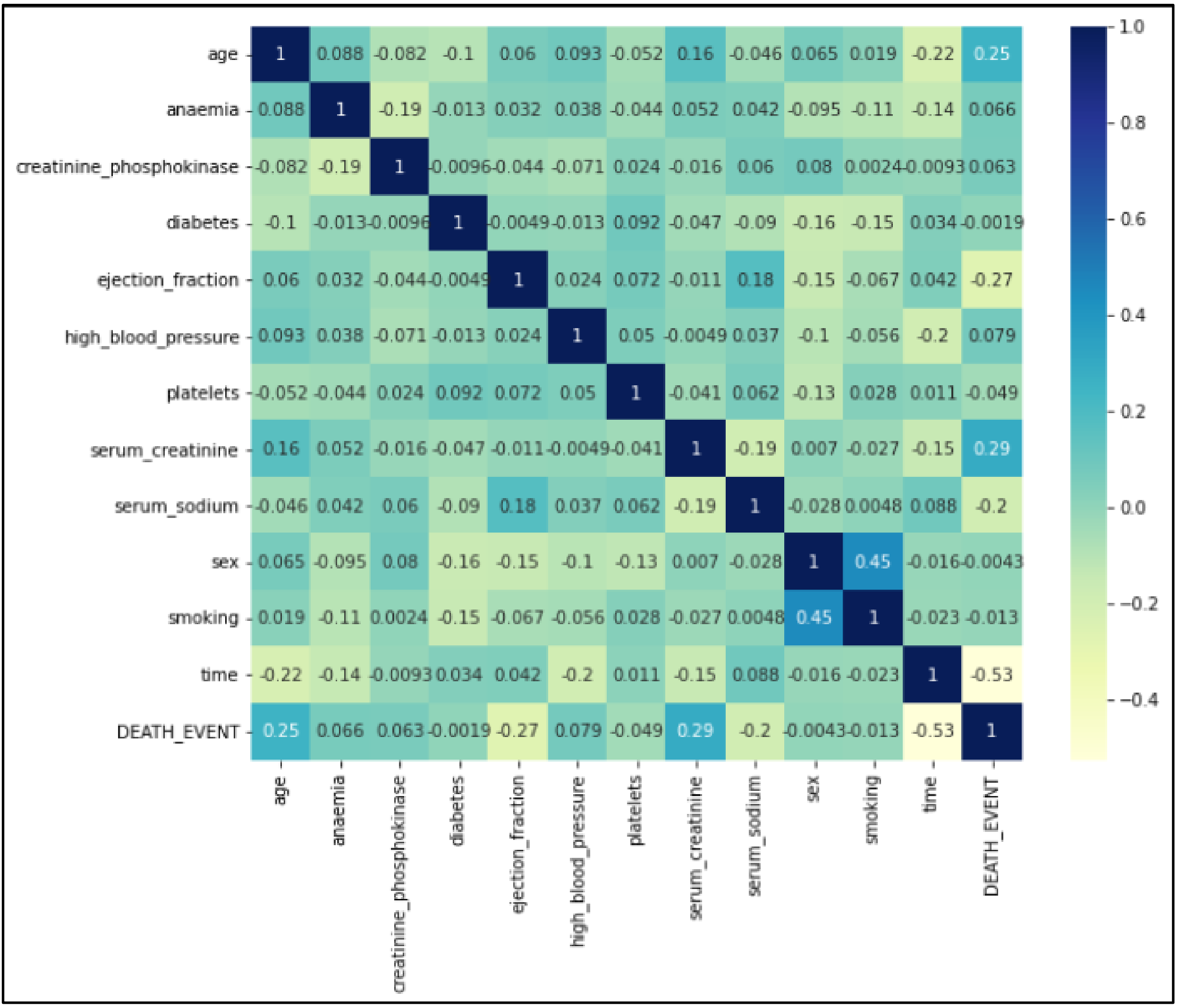

age
Density
0.03
0.02
0.01
0.00
50
100
anaemia
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0
1
creatinine_phosphokinase
0.0008
0.0006
0.0004
0.0002
0.0000
0
5000
10000
diabetes
ejection_fraction
0.04
0.03
0.02
0.01
0.00
0
50
100
high_blood_pressure
1.5
1.0
0.5
0.0
platelets
1e−6
5
4
3
2
1
0
0.0
0.5
1.0
1e6
serum_creatinine
0.8
0.6
0.4
0.2
0.0
0
5
10
serum_sodium
0.10
0.08
0.06
0.04
0.02
0.00
100
120
140
160
sex
smoking
time
0.005
0.004
0.003
0.002
0.001
0.000
0
200
400
DEATH_EVENT

Fig 1.5

DATA REDUCTION

Heat Map- A heat map represents these coefficients to visualize the strength of correlation among variables. It helps find features that are best for Machine Learning model building.
The heat map transforms the correlation matrix into color coding



Here we are only considering the data which have absolute correlation greater than 0.05 to remove redundant features from the dataset as that can cause overfitting. Moreover, Model performed better without left parameters.

Feature	Correlation
age	0.253729
anaemia	0.06627
creatinine_phosphokinase	0.062728
ejection_fraction	-0.2686
high_blood_pressure	0.079351
serum_creatinine	0.294278
serum_sodium	-0.1952
time	-0.52696

Data Splitting - Data is divided into two-parts, one part is used to evaluate or test the data and the other to train the model. After Considering various split ratio the model was giving best performance with a ratio as 75 % of training data and 25 % of testing data from total dataset after removing redundancies.

Total Dataset Values and Features :- (299,8)
Training Dataset Values and Features :- (224,8)
Testing Dataset Values and Features :- (75,8)

ML AlGORITHMS APPLIED ON THE DATASET

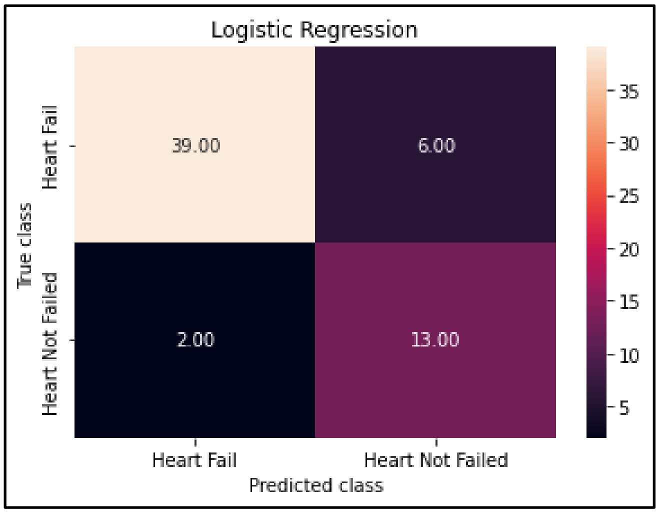
1.LOGISTIC REGRESSION - Logistic regression is a statistical analysis method to predict a binary outcome, such as yes or no, based on prior observations of a data set. A logistic regression model predicts a dependent data variable by analyzing the relationship between one or more existing independent variables.

Without Regularization - In this approach we don’t penalize the cost function by adding a factor. It does not work properly if there is any overfitting in the model. After following this approach the results are mentioned below.

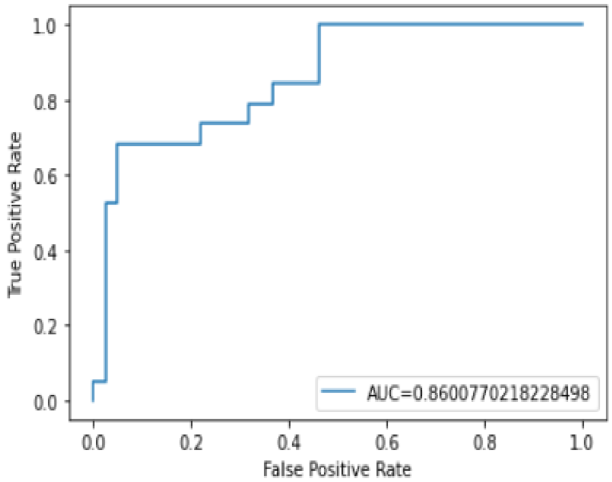
Death Event	Precision	Recall	F1-score
0	0.77	0.95	0.91
1	0.67	0.42	0.52

The Model gave Accuracy of **86.6%** on Test data.

CONFUSION MATRIX



ROC CURVE



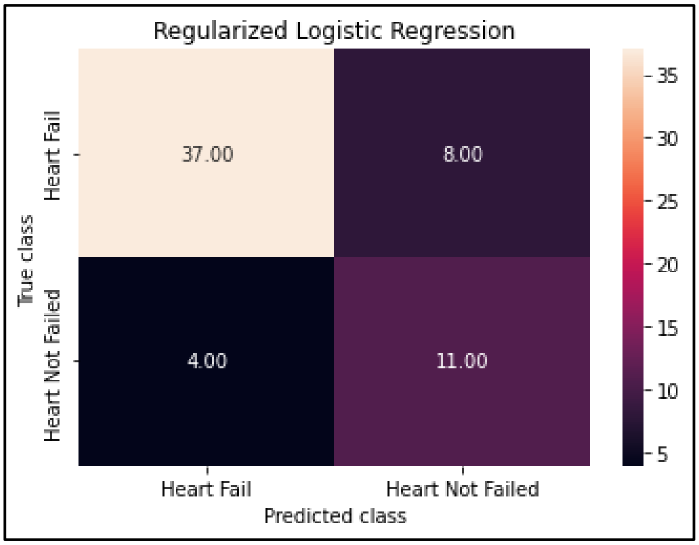
CrossValidation - Cross-validation is a technique in which we train our model using the subset of the data-set and then evaluate using the complementary subset of the data-set. With Cross Validation the mean accuracy for logistic regression without regularization is **81.03%** with **11 folds**.

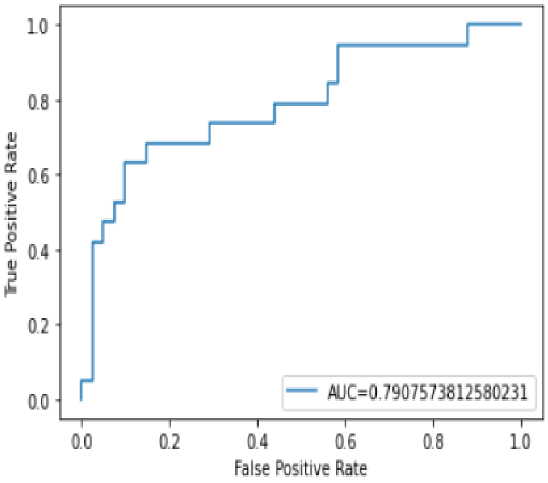
After Regularization - Regularization is used to remove Overfitting in the model. It adds an additional penalizing parameter to the cost function.

Death Event	Precision	Recall	F1-score
0	0.82	0.90	0.86
1	0.67	0.42	0.52

The Model gave Accuracy of **80%** on Test data. With L2 Regularization and max iterations as **10000.**

CONFUSION MATRIX





With Cross Validation mean accuracy for logistic regression with regularization is **82.55%** with **12 folds**.

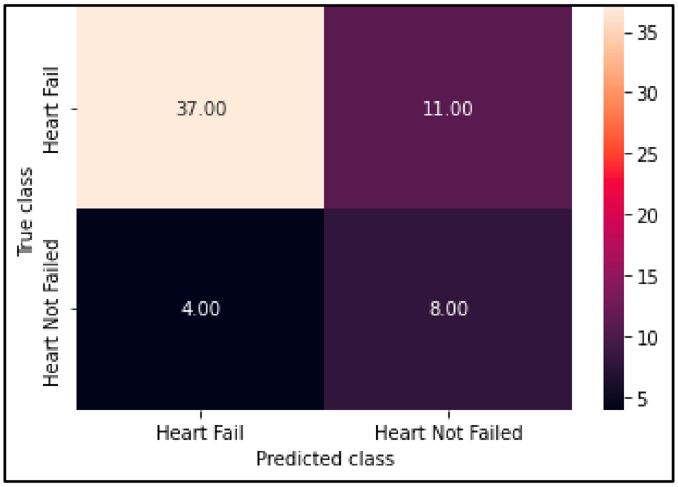
2.K-NEAREST NEIGHBOR(KNN) - KNN is a simple, supervised machine learning (ML) algorithm that can be used for classification or regression tasks - and is also frequently used in missing value imputation. K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data.

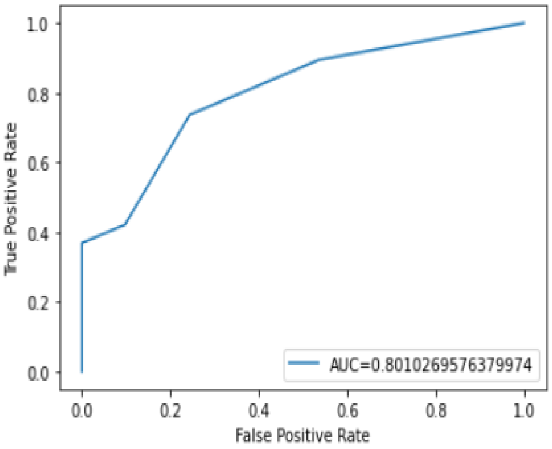
Model Evaluation Parameters -

Death Event	Precision	Recall	F1-score
0	0.77	0.90	0.83
1	0.67	0.42	0.52

The model gave accuracy of **75%** with nearest neighbors as **5**.

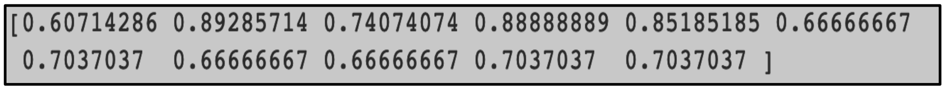
CONFUSION MATRIX





With Cross Validation the mean accuracy for KNN is **73.57%** with **11 folds.**

Accuracy Table For All Folds



```
[0.60714286 0.89285714 0.74074074 0.88888889 0.85185185 0.66666667
 0.7037037  0.66666667 0.66666667 0.7037037  0.7037037 ]
```

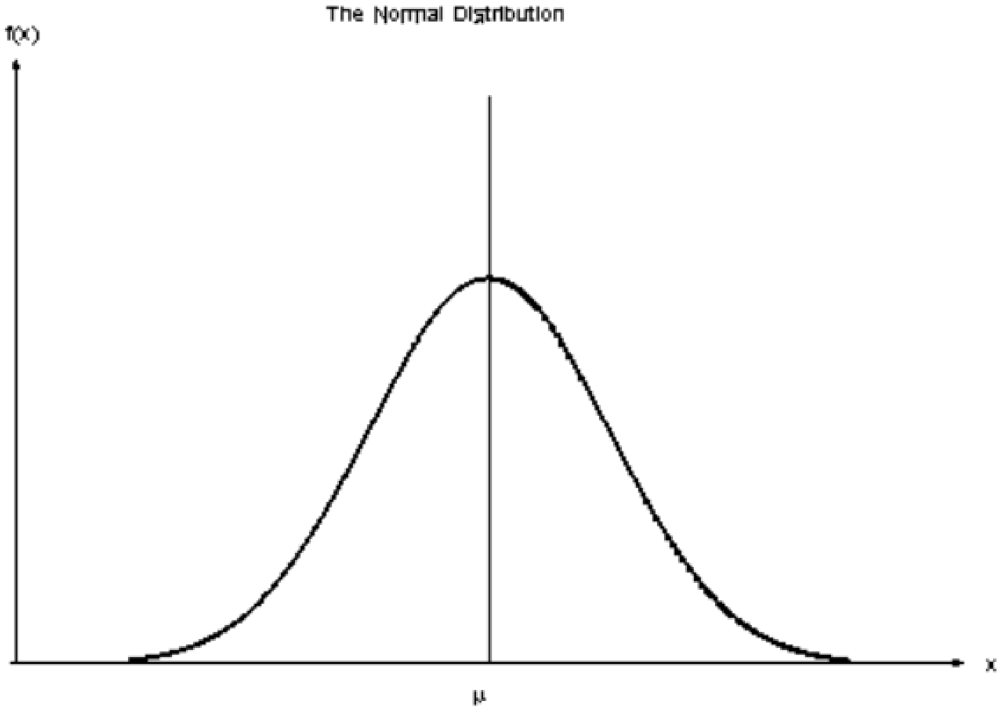
3.NAIVE BAYES CLASSIFIERS - Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

Bayes Theorem:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Using Bayes theorem, we can find the probability of **A** happening, given that **B** has occurred. Here, **B** is the evidence and **A** is the hypothesis. The assumption made here is that the predictors/features are independent. That is presence of one particular feature does not affect the other. Hence it is called naive.

GAUSSIAN NAIVE BAYES: When the predictors take up a continuous value and are not discrete, we assume that these values are sampled from a gaussian distribution.

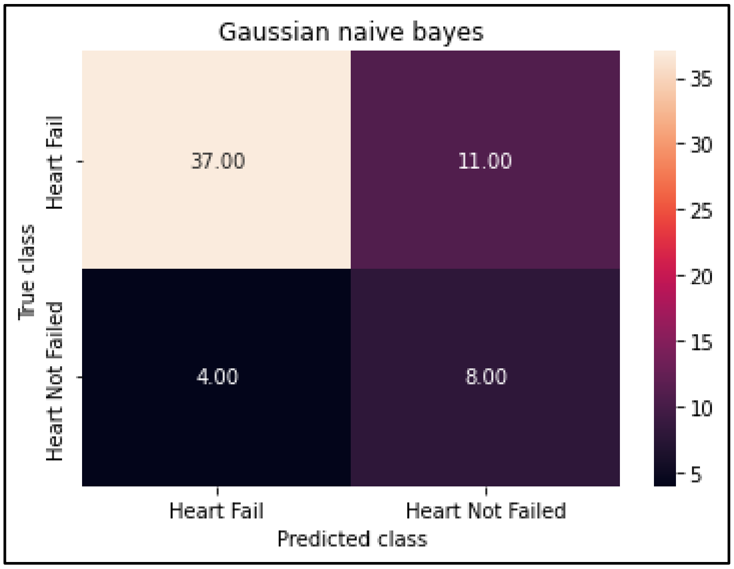


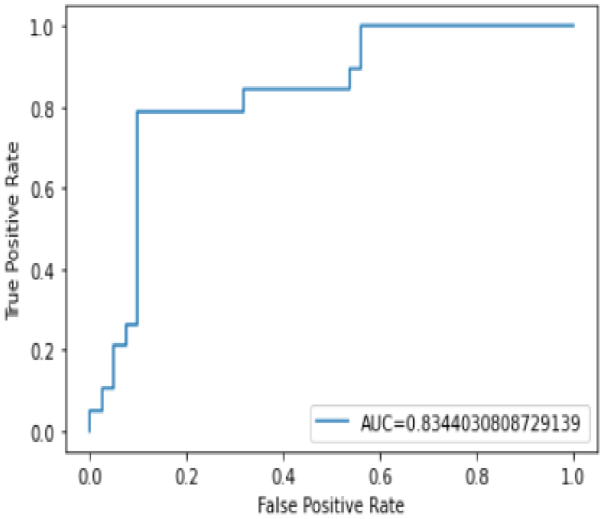
Gaussian Distribution(Normal Distribution)

Model Evaluation Parameters -

Death Event	Precision	Recall	F1-score
0	0.77	0.90	0.83
1	0.67	0.42	0.52

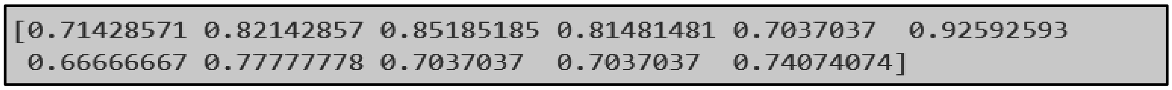
CONFUSION MATRIX





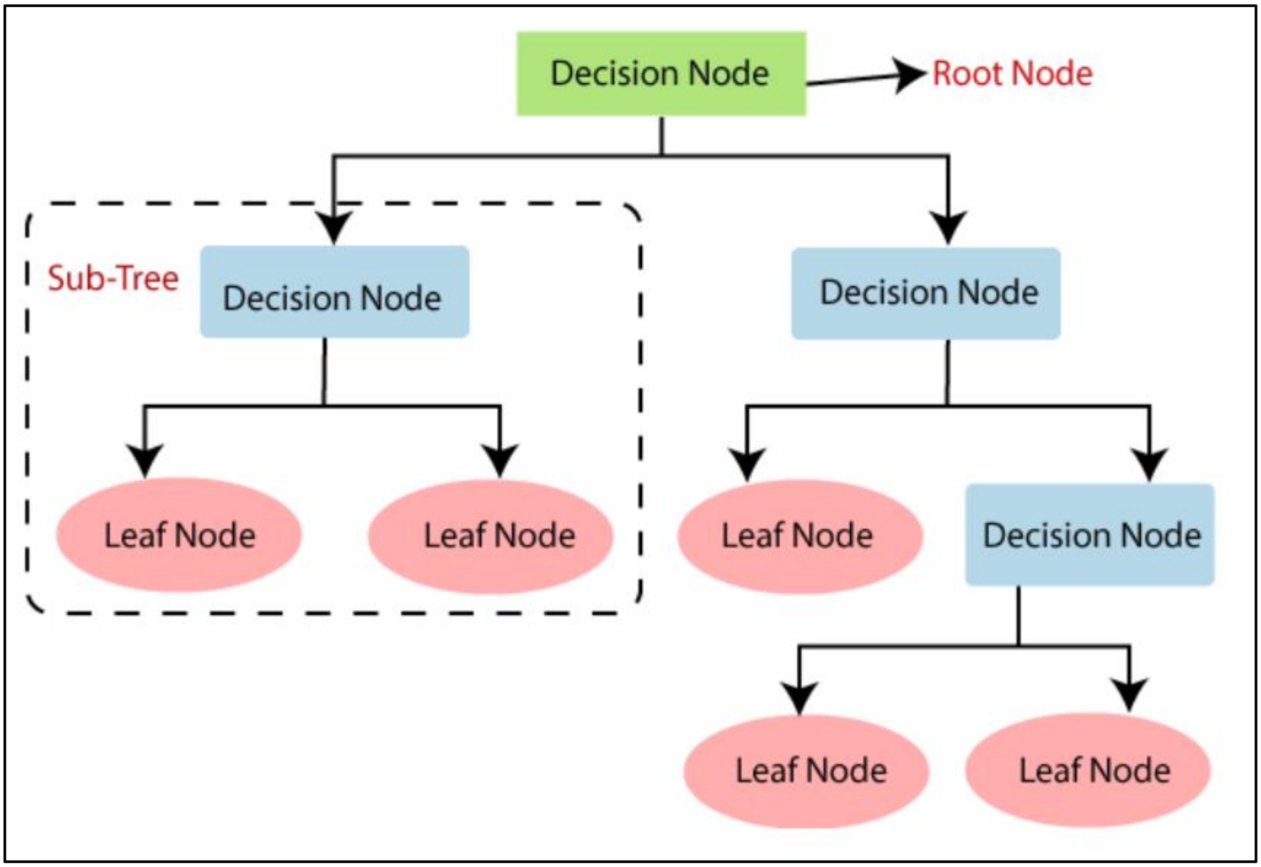
Gaussian naive bayes gave very similar results to KNN.
This model gave an accuracy score of **75%** on test data and **77.8%** on training data
For k cross validation it gave an mean accuracy score of **76.58%** for **11 folds.**

Accuracy for 11 folds is given below



```
[0.71428571 0.82142857 0.85185185 0.81481481 0.7037037  0.92592593
 0.66666667 0.77777778 0.7037037  0.7037037  0.74074074]
```

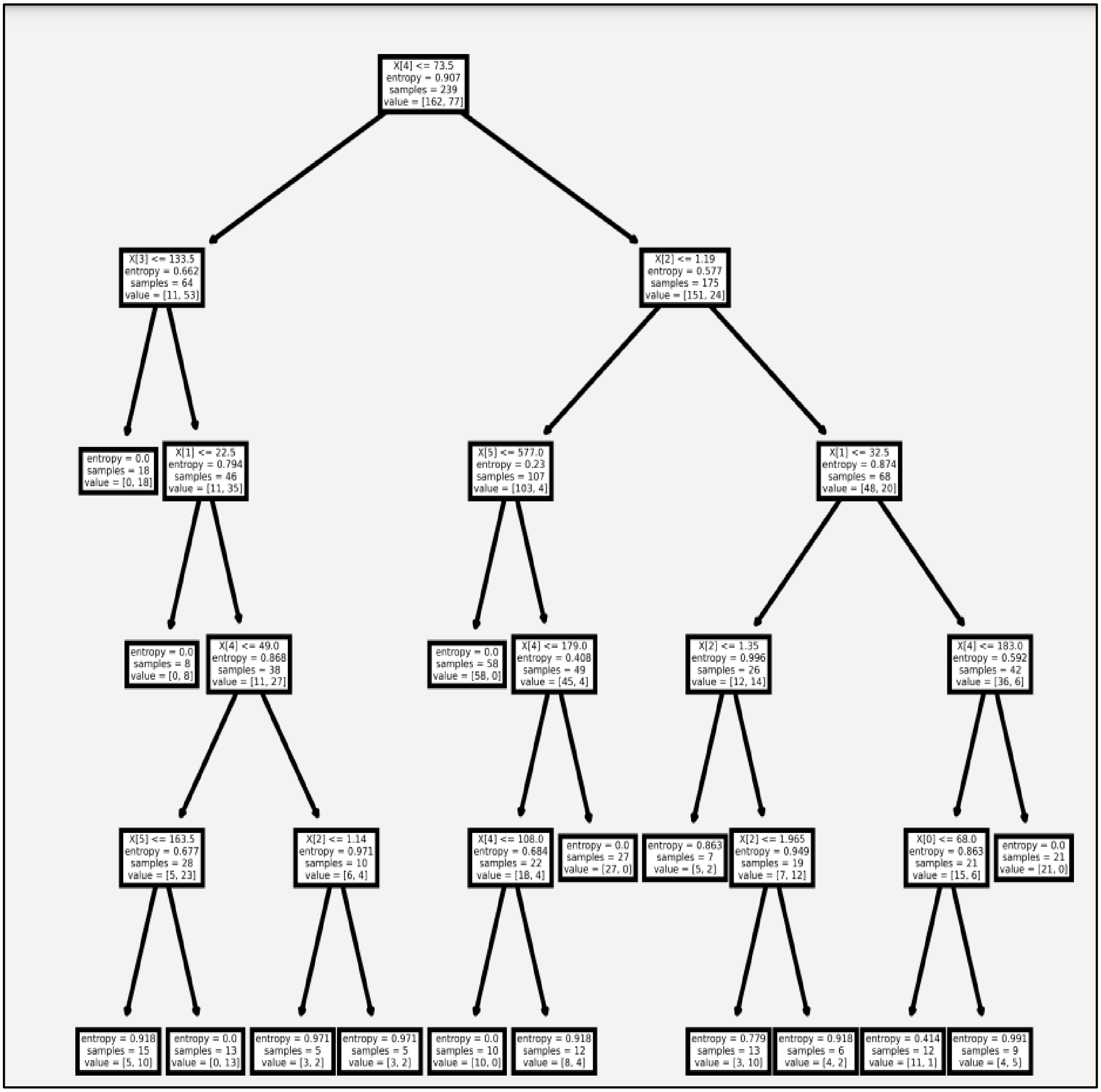
4. DECISION TREE CLASSIFIER - Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.



Growing a tree involves deciding on **which features to choose** and **what conditions to use** for splitting, along with knowing when to stop. As a tree generally grows arbitrarily, **you will need to trim it down** for it to look beautiful. Let's start with a common technique used for splitting.

The performance of a tree can be further increased by ***pruning***. *It involves* ***removing the branches that make use of features having low importance***. This way, we reduce the complexity of the tree, and thus increase its predictive power by reducing overfitting.

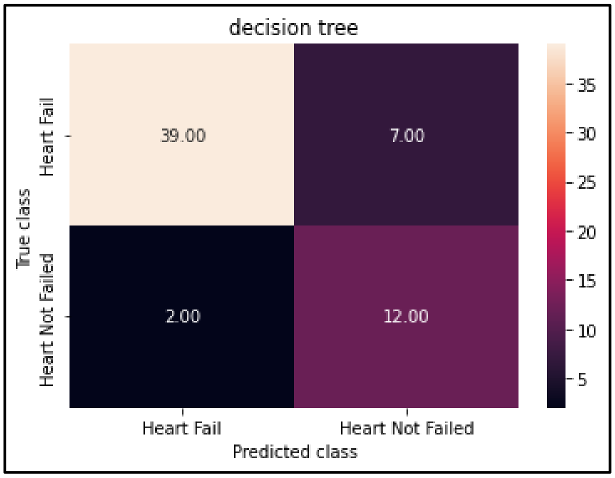
For the given dataset following decision tree was generated:

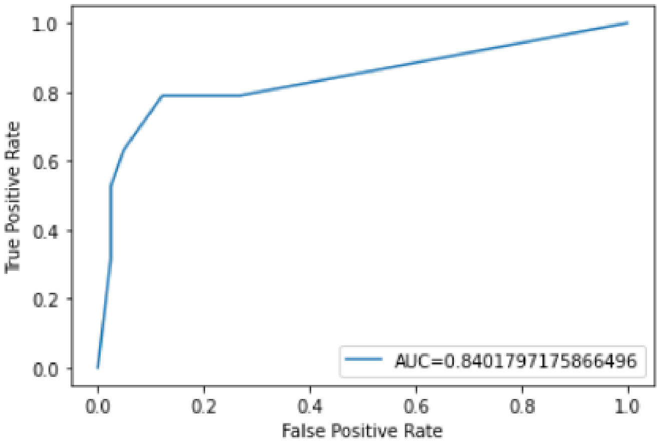


Model Evaluation Parameters -

Death Event	Precision	Recall	F1-score
0	0.85	0.95	0.90
1	0.86	0.63	0.73

CONFUSION MATRIX



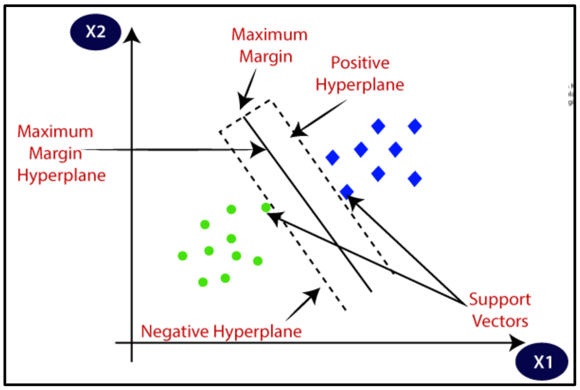


This is indeed the best performing model for the given dataset with an accuracy score of **85%** when the decision tree was trimmed to **max_depth** equal to **5**. Trimming data drastically increased accuracy from **75% to 85%.** For cross validation, the model gave a mean accuracy of **82.86%** for **61 folds**.

Support Vector Machine Algorithm

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane. SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called support vectors, and hence the algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane:



5. Linear SVM -

Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and the classifier is used as Linear SVM classifier.

The working of the SVM algorithm can be understood by using an example. Suppose we have a dataset that has two tags (green and blue), and the dataset has two features x1 and x2. We want a classifier that can classify the pair(x1, x2) of coordinates in either green or blue. Consider Fig 1. image:

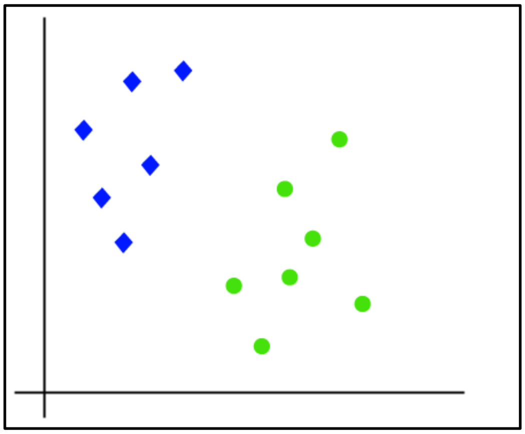


Fig 1.

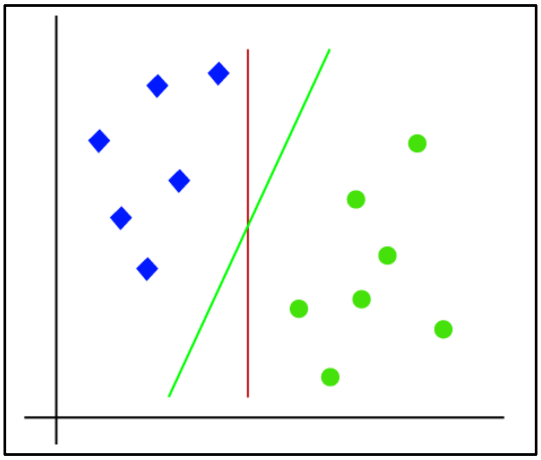
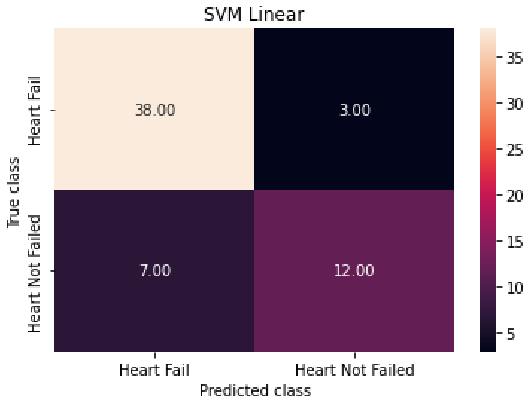


Fig2.

So as it is 2-d space so by just using a straight line, we can easily separate these two classes. But there can be multiple lines that can separate these classes. Consider the image Fig 2.

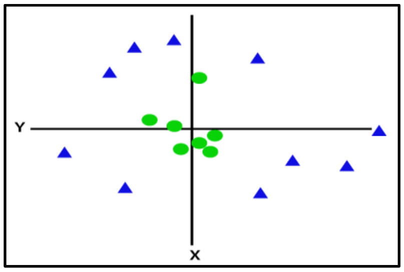
Hence, the SVM algorithm helps to find the best line or decision boundary; this best boundary the region is called a hyperplane. SVM algorithm finds the closest point of the lines from both the classes. These points are called support vectors. The distance between the vectors and the hyperplane is called the margin. And the goal of SVM is to maximize this margin.
The hyperplane with maximum margin is called the optimal hyperplane.

CONFUSION MATRIX



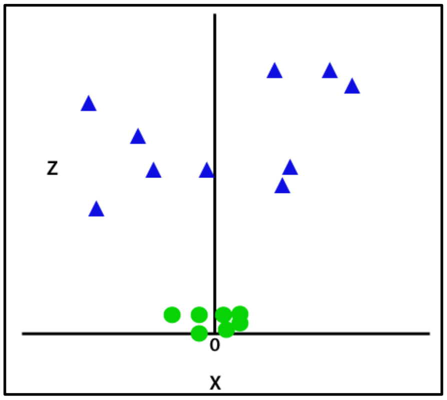
6. Non-Linear SVM (RBF SVM) -

If data is linearly arranged, then we can separate it by using a straight line, but for non-linear data, we cannot draw a single straight line. Consider the below image:

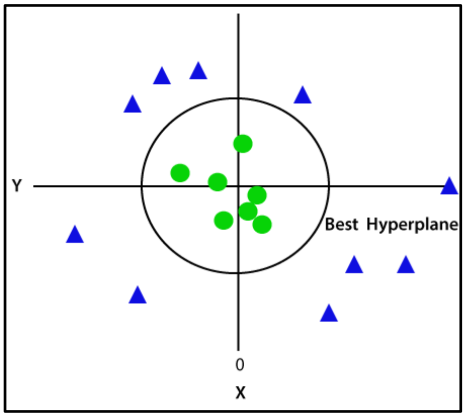


So to separate these data points, we need to add one more dimension. For linear data, We have used two dimensions x and y, so for non-linear data, we will add a third dimension z. It can be calculated as: $z=x^2+y^2$

By adding the third dimension, the sample space will become as below image:

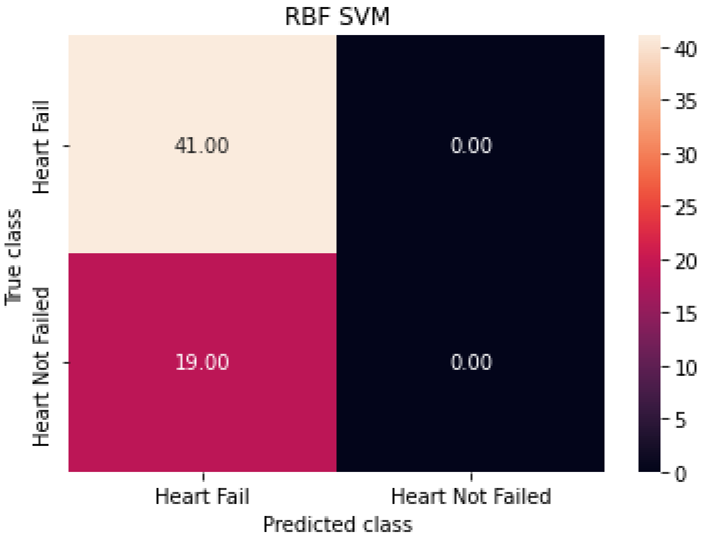


Sample Space



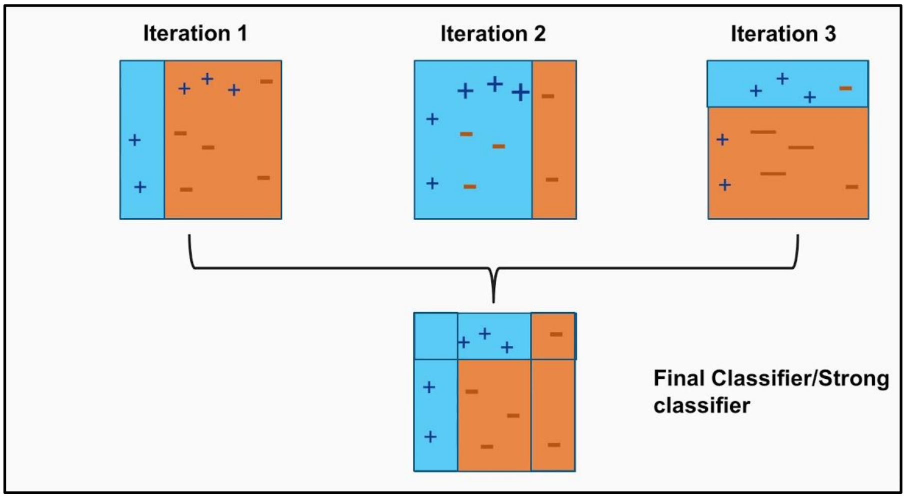
Output Hyperplane using SVM

CONFUSION MATRIX

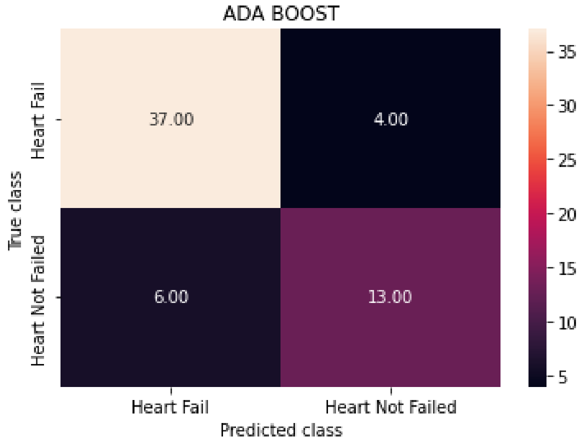


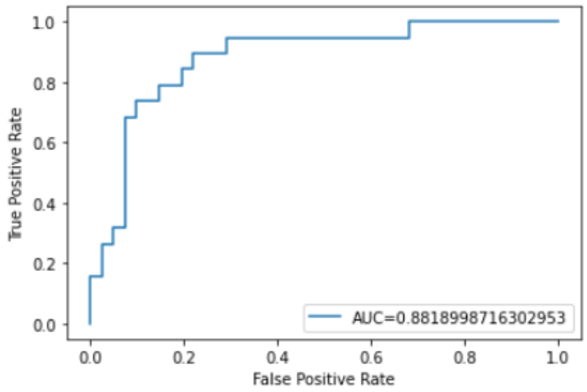
7. AdaBoost-

AdaBoost is short for Adaptive Boosting and is a very popular boosting technique that combines multiple “weak classifiers” into a single “strong classifier”. Boosting is an ensemble modeling technique that attempts to build a strong classifier from the number of weak classifiers. It is done by building a model by using weak models in series. Firstly, a model is built from the training data. Then the second model is built which tries to correct the errors present in the first model. This procedure is continued and models are added until either the complete training data set is predicted correctly or the maximum number of models are added.



CONFUSION MATRIX

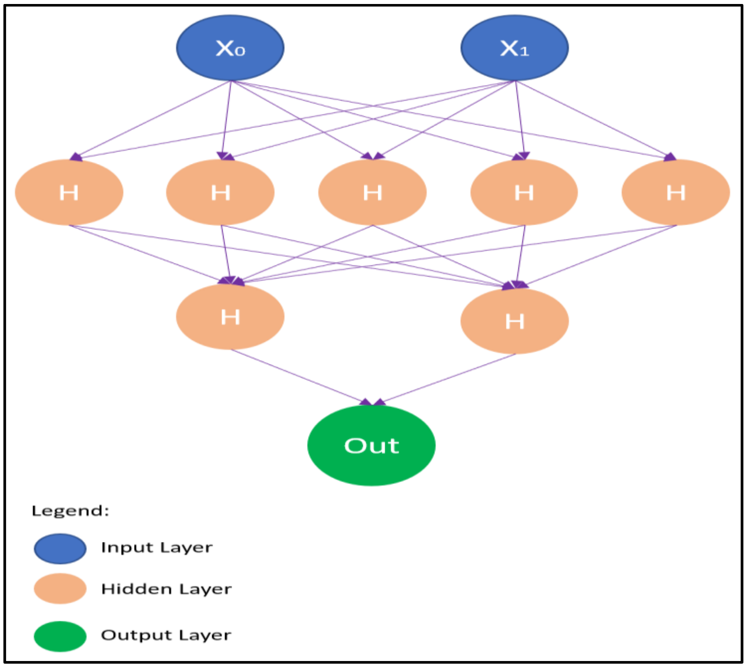




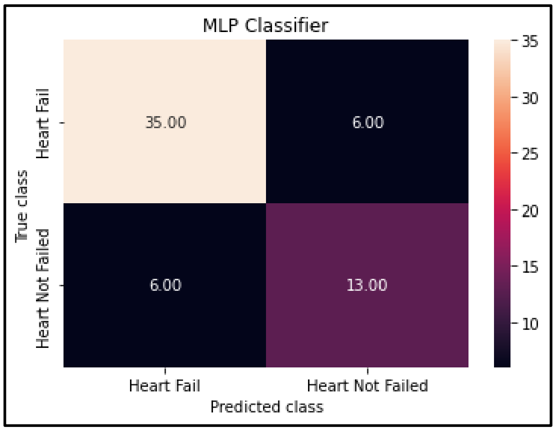
8. MLP Classifier

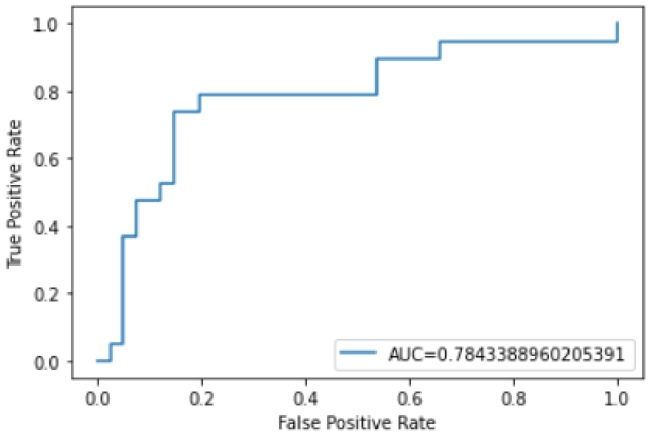
MLPClassifier stands for Multi-layer Perceptron classifier which in the name itself connects to a Neural Network. Unlike other classification algorithms such as Support Vectors or Naive Bayes Classifier, MLPClassifier relies on an underlying Neural Network to perform the task of classification.

An MLP consists of at least three layers of nodes: an input layer, a hidden layer and an output layer. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. MLP utilizes a supervised learning technique called backpropagation for training.Its multiple layers and non-linear activation distinguish MLP from a linear perceptron. It can distinguish data that is not linearly separable.



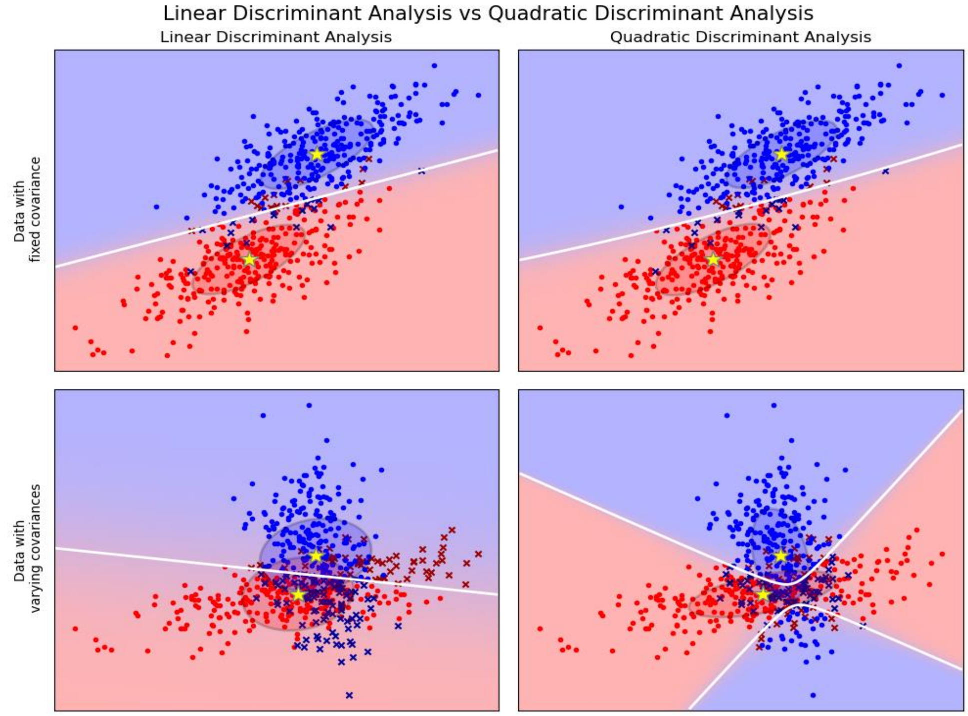
CONFUSION MATRIX



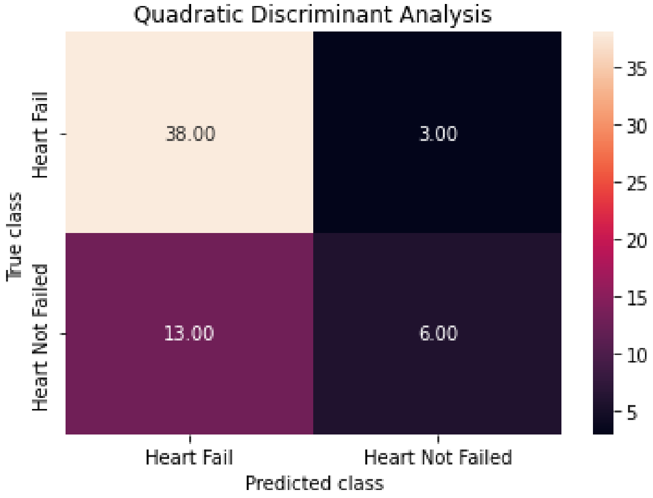


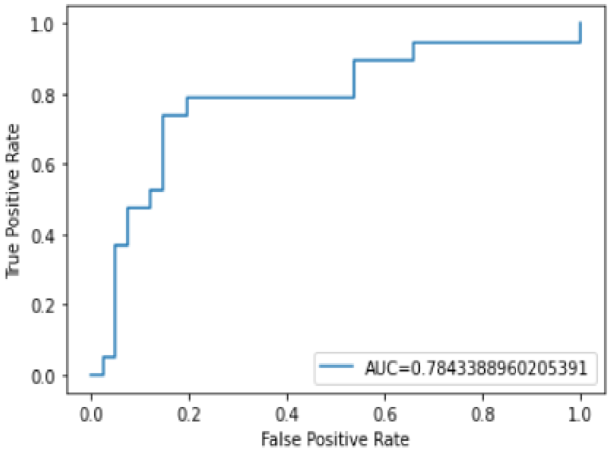
9. Quadratic Discriminant Analysis

Quadratic Discriminant Analysis (QDA) is a generative model. It assumes that each class follow a Gaussian distribution. The class-specific prior is simply the proportion of data points that belong to the class, the class-specific mean vector is the average of the input variables that belong to the class, the class-specific covariance matrix is just the covariance of the vectors that belong to the class.



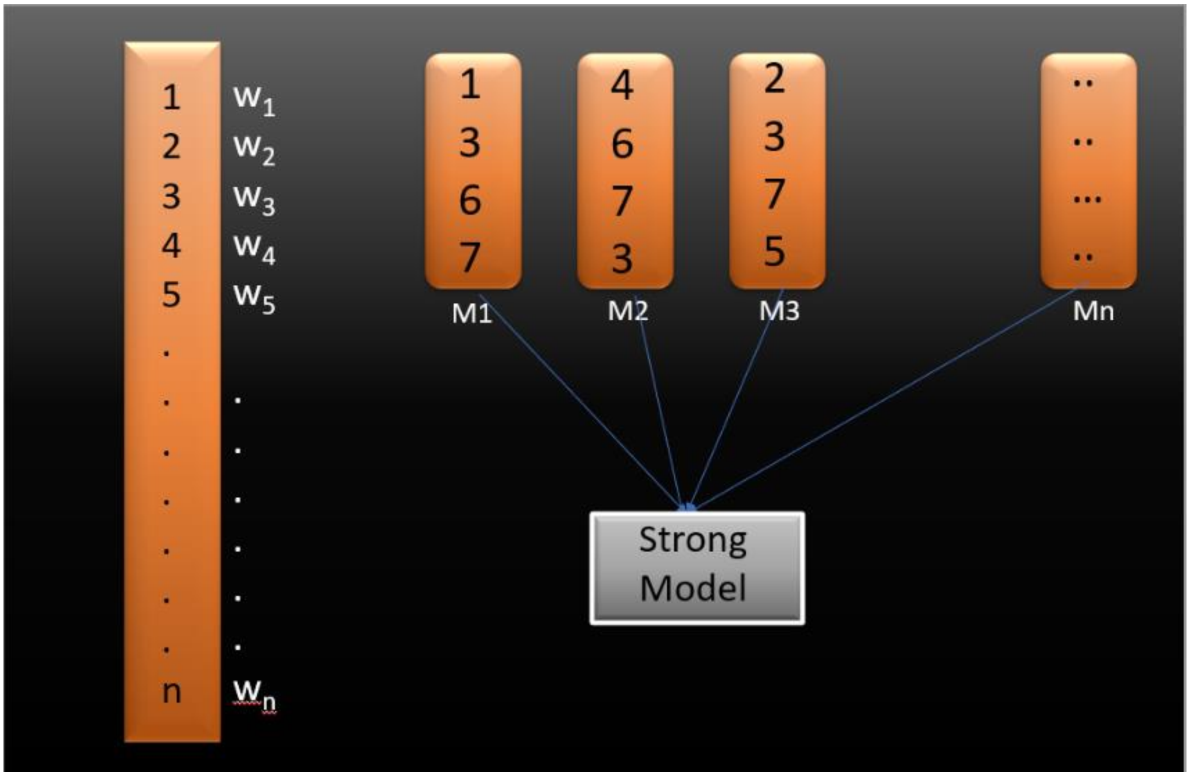
CONFUSION MATRIX



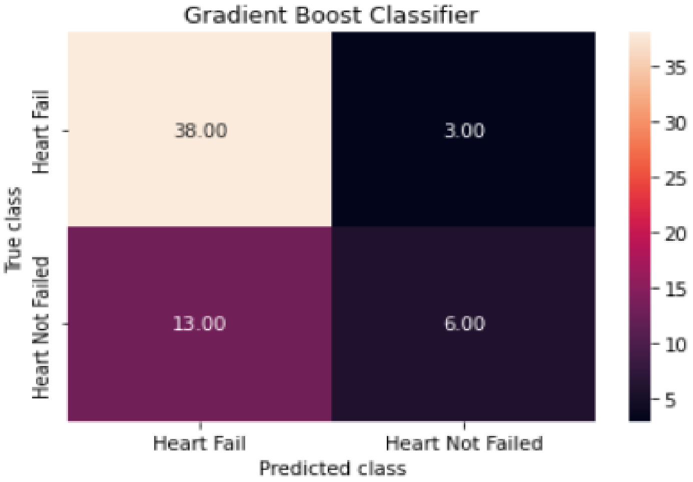


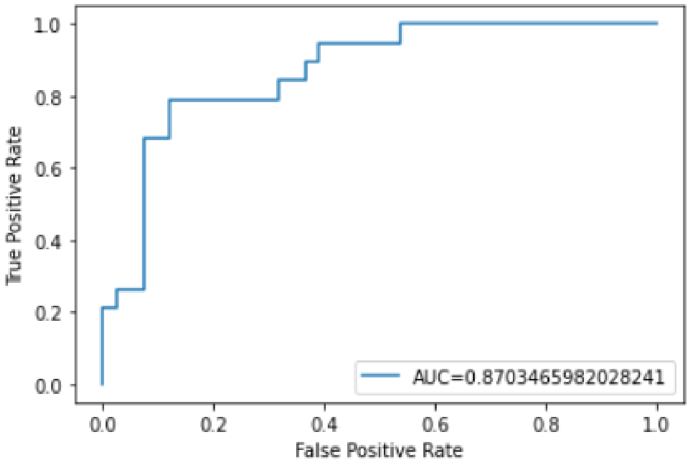
10.Gradient boosting Classifier

Gradient boosting is a method standing out for its prediction speed and accuracy, particularly with large and complex datasets. From Kaggle competitions to machine learning solutions for business, this algorithm has produced the best results. We already know that errors play a major role in any machine learning algorithm. There are mainly two types of error, bias error and variance error. Gradient boost algorithm *helps us minimize bias error of the model*



CONFUSION MATRIX



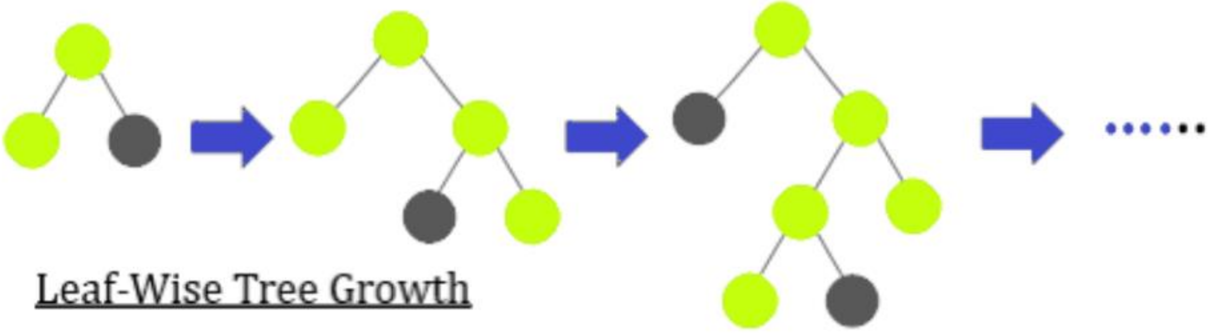


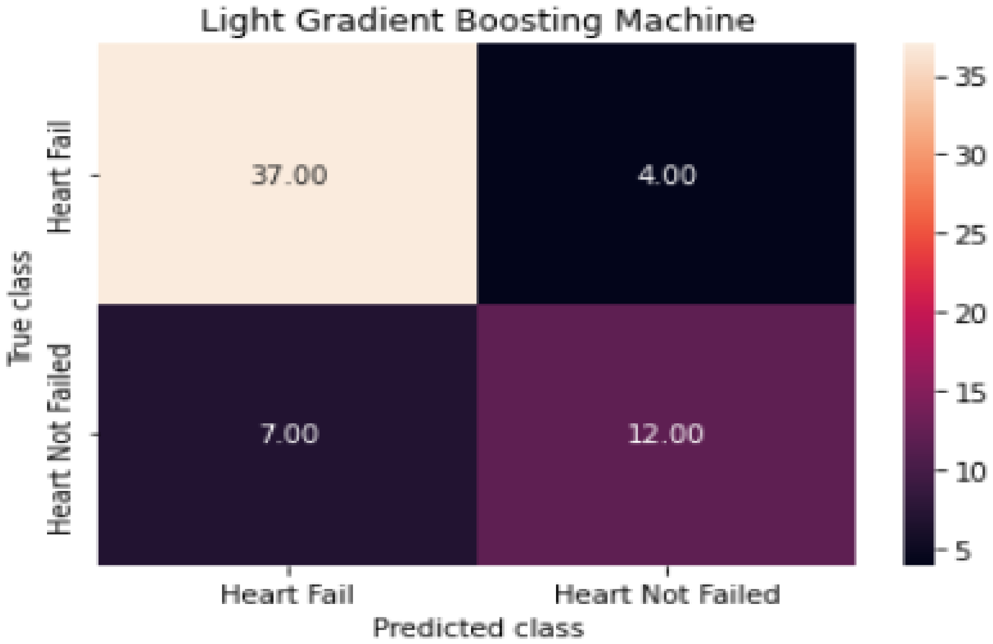
11.Light gradient boosting machine

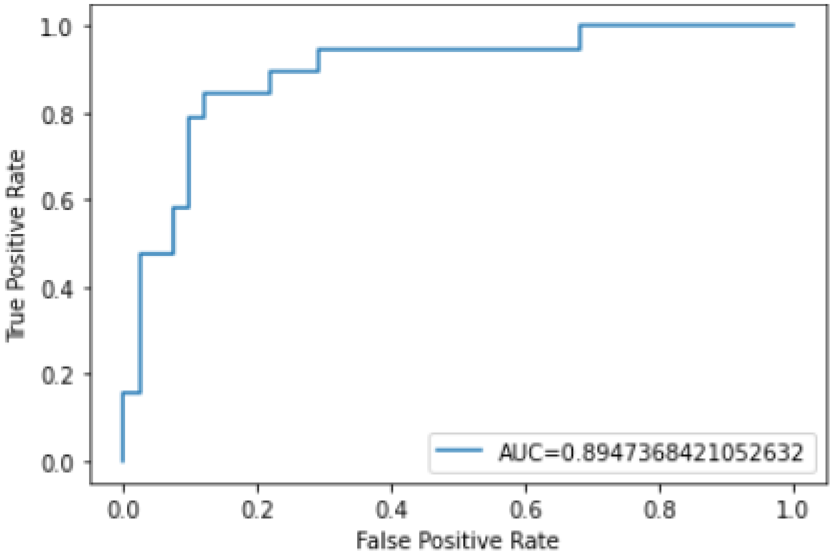
LightGBM is a gradient boosting framework based on decision trees to increases the efficiency of the model and reduces memory usage.

It uses two novel techniques: Gradient-based One Side Sampling and Exclusive Feature Bundling (EFB) which fulfills the limitations of histogram-based algorithm that is primarily used in all GBDT (Gradient Boosting Decision Tree) frameworks. The two techniques of GOSS and EFB described below form the characteristics of LightGBM Algorithm. They comprise together to make the model work efficiently and provide it a cutting edge over other GBDT frameworks.

$$\tilde{V}_j(d) = \frac{1}{n}\left(\frac{\left(\sum_{x_i\in A_l} g_i + \frac{1-a}{b}\sum_{x_i\in B_l} g_i\right)^2}{n_l^j(d)} + \frac{\left(\sum_{x_i\in A_r} g_i + \frac{1-a}{b}\sum_{x_i\in B_r} g_i\right)^2}{n_r^j(d)}\right)$$

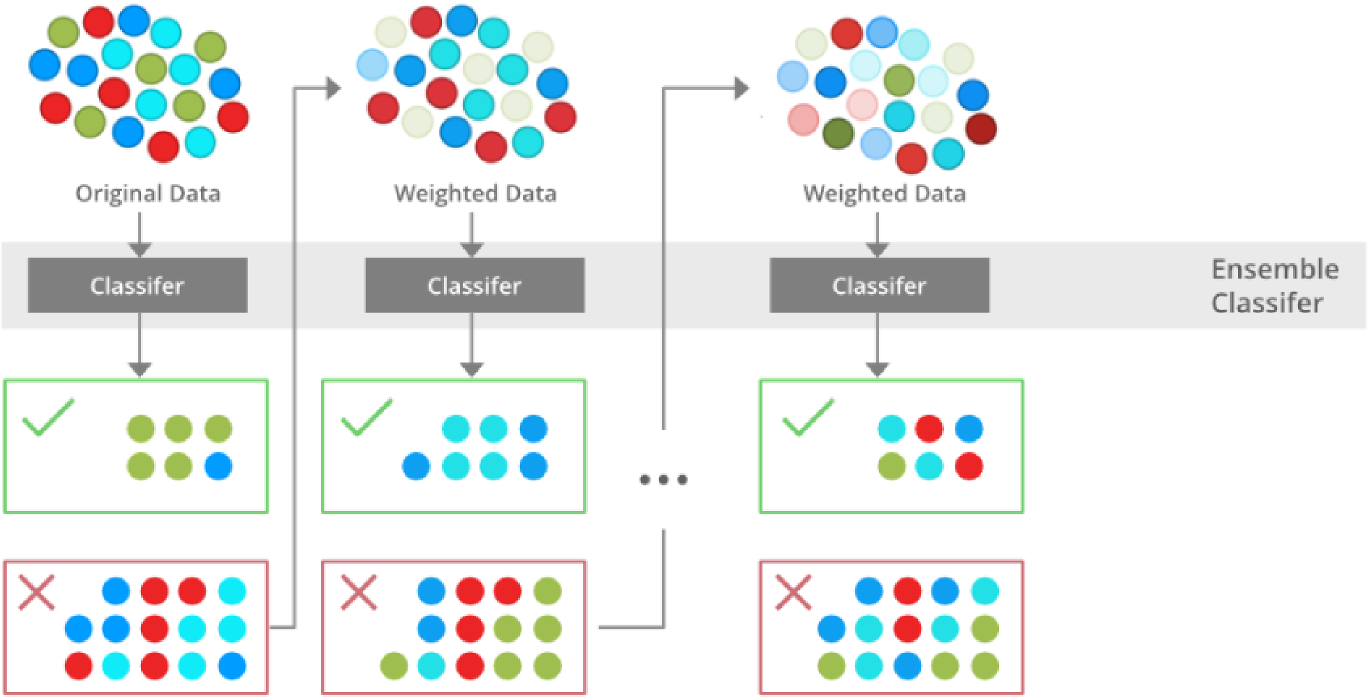



Light Gradient Boosting Machine
True class
Heart Fail
Heart Not Failed
37.00
4.00
7.00
12.00
Heart Fail
Heart Not Failed
Predicted class
35
30
25
20
15
10
5


True Positive Rate
1.0
0.8
0.6
0.4
0.2
0.0
AUC=0.8947368421052632
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate

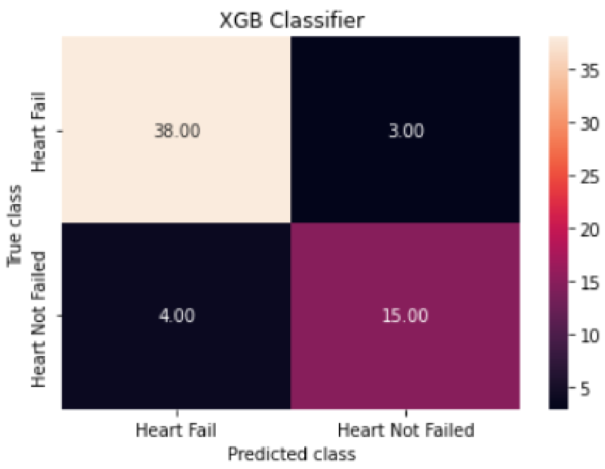
12.XGBOOST

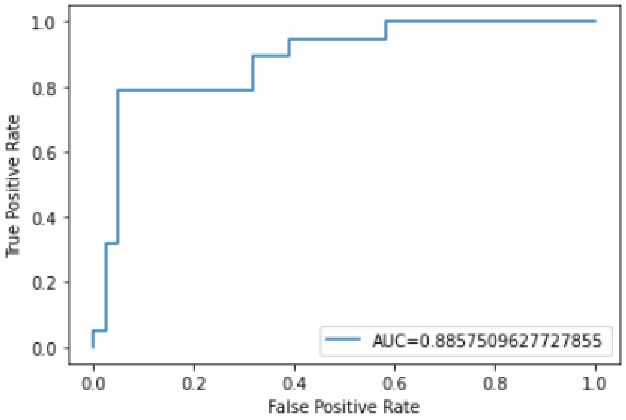
XgBoost stands for Extreme Gradient Boosting, which was proposed by the researchers at the University of Washington. It is a library written in C++ which optimizes the training for Gradient Boosting.



XGBoost is an implementation of Gradient Boosted decision trees. XGBoost models majorly dominate in many Kaggle Competitions.

In this algorithm, decision trees are created in sequential form. Weights play an important role in XGBoost. Weights are assigned to all the independent variables which are then fed into the decision tree which predicts results. The weight of variables predicted wrong by the tree is increased and these variables are then fed to the second decision tree. These individual classifiers/predictors then ensemble to give a strong and more precise model. It can work on regression, classification, ranking, and user-defined prediction problems.

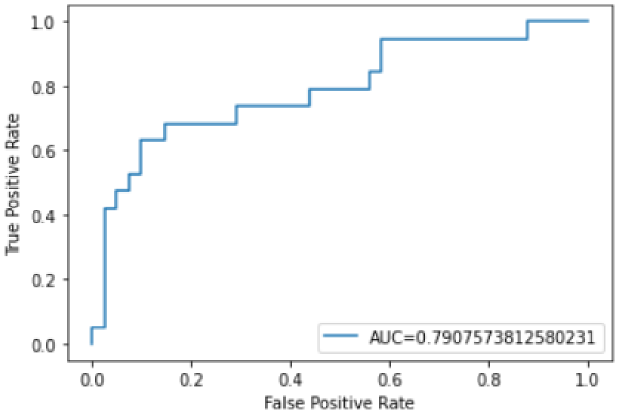




13.Extra tree classifier

Extremely Randomized Trees Classifier(Extra Trees Classifier) is a type of ensemble learning technique which aggregates the results of multiple de-correlated decision trees collected in a “forest” to output it’s classification result. In concept, it is very similar to a Random Forest Classifier and only differs from it in the manner of construction of the decision trees in the forest.

Each Decision Tree in the Extra Trees Forest is constructed from the original training sample. Then, at each test node, Each tree is provided with a random sample of k features from the feature-set from which each decision tree must select the best feature to split the data based on some mathematical criteria (typically the Gini Index). This random sample of features leads to the creation of multiple de-correlated decision trees.



Comparison between Algorithm based on Model Evaluation Parameter

		Accuracy	Precision	Recall	F1 score	Sensitivity	Specificity	AUC (ROC)	Error Rate	FPR
1	Logistic Regression (without Reg.)	86.6%	72%	68.5%	71.5%	95.1%	68.4%	86.1%	13.4%	86.6%
2	Logistic Regression (with Reg.)	80%	74.5%	66%	69%	90.2%	42.1%	79.1%	20%	73.3%
3	KNN	75%	72%	66%	67.5%	90.2%	42.1%	80.1%	25%	66.7%
4	Naïve Bayes	75%	72%	66%	67.5%	90.2%	42.1%	83.44%	25%	66.7%
5	Decision Tree	85%	85.5%	79%	81%	79%	NAN	84.1%	15%	85.7%
6	Decision Tree (Gini index)	83.3%	84%	77%	79%	95.1%	57.8%	72.7%	16.7%	42.1%
7	Linear SVM	83.3%	84%	93%	88%	86.3%	80%	NAN	16.7%	63.1%
8	RBF SVM	68.3%	68%	50%	81%	68.3%	NAN	NAN	31.7%	0%
9	AdaBoost	83.3%	86%	90%	88%	86%	76.4%	88.1%	16.7%	68.5%
10	MLP Classifier	80%	85%	85%	85%	85.3%	68.1%	78.43%	20%	68.5%
11	Quadratic Discriminant Analysis	73.3%	75%	93%	83%	74.5%	66.6%	78.43%	26.7%	31.5%

12	Gradient Boosting Classifier	73.3%	72%	73%	70%	73.4%	66.7%	87.1%	26.7%	33.3%
13	Light Gradient boosting machine	81.67%	81%	82%	81%	84.1%	75%	89.47%	18.23%	25%
14	XGBOOST	88.4%	88%	88%	88%	90.47%	83.34%	88.6%	11.6%	16.67%
15	Extra tree classifier	83.34%	83%	83%	83%	92.6%	80%	92.74%	17.7%	20%

Conclusion

After Analyzing the model using all the 15 Algorithm we can fairly conclude that the most accurate and precise predictions were made out with XGBOOST with the accuracy being 88.4%.Logistic regression came forward as second best algorithm with 86.6% accuracy.Naive bayes had the lowest runtime with significance
Difference from other classification algorithms.

Citations

(1) Ghost. Directed by Jerry Zucker. Paramount Pictures, 1990.

(2) "The top 10 causes of death ", World Health Organization. https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death.

(3) "Cardiovascular Diseases", World Health Organization. https://www.who.int/health-topics/cardiovascular-diseases.

(4) "CARMAT Receives the CE Marking for Its Total Artificial Heart", Bloomberg https://www.bloomberg.com/press-releases/2020-12-23/carmat-receives-the-ce-marking-for-its-total-artificial-heart