INDIAN INSTITUTE OF TECHNOLOGY BOMBAY

STATISTICAL TECHNIQUES AND DATA MINING

VISHNU M
23M0025

PROJECT REPORT
Nov 2023

INTRODUCTION

This project revolves around the task of classification wherein various classification methods are explored and compared.

We obtain data from the National Institute of Diabetes and Digestive and Kidney Diseases. The objective of the dataset is to diagnostically predict whether or not a patient has diabetes, based on certain diagnostic measurements included in the dataset.

Motivation:

Building various classification models to accurately predict whether or not the patients in the dataset have diabetes or not.

Data Description:

The datasets consist of several medical predictor variables and one target variable, Outcome. Predictor variables includes the number of pregnancies the patient has had, their BMI, insulin level, age, and so on.

All patients here are females at least 21 years old of Pima Indian heritage.

Source of data:

This dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases.

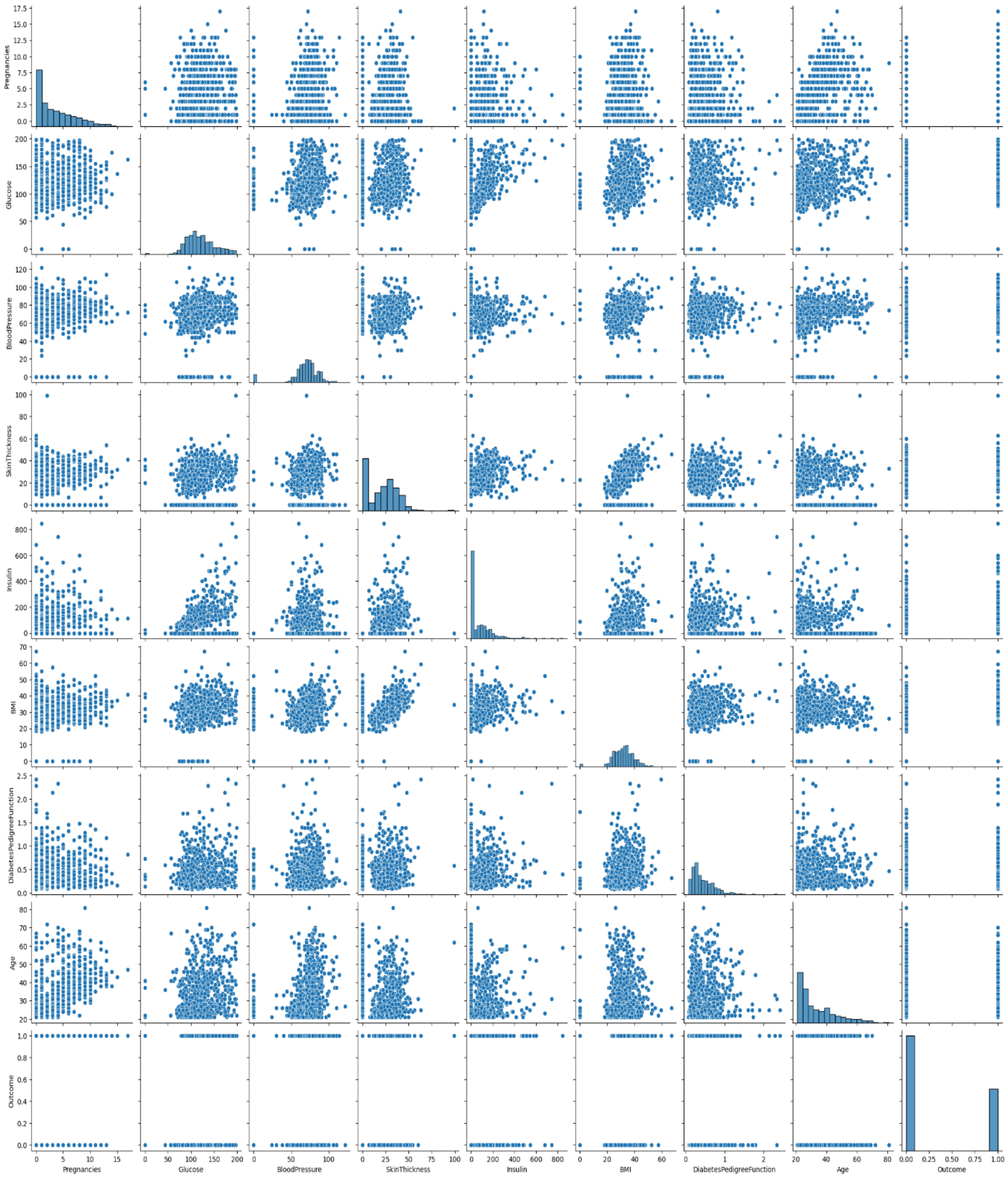
Tasks completed in the project:

- Exploratory Data Analysis
- Train-test splitting
- Developing various classification models on training data-
    - Logistic Regression
    - K- nearest neighbours
    - LDA
    - QDA
    - Naïve Bayes
    - Decision tree
    - Random Forest
- Validating various models and comparing their performances on the basis of-
    - Plotting ROC curve and computing area under the ROC curve (AUC)
    - Computing best threshold and corresponding sensitivity and specificity

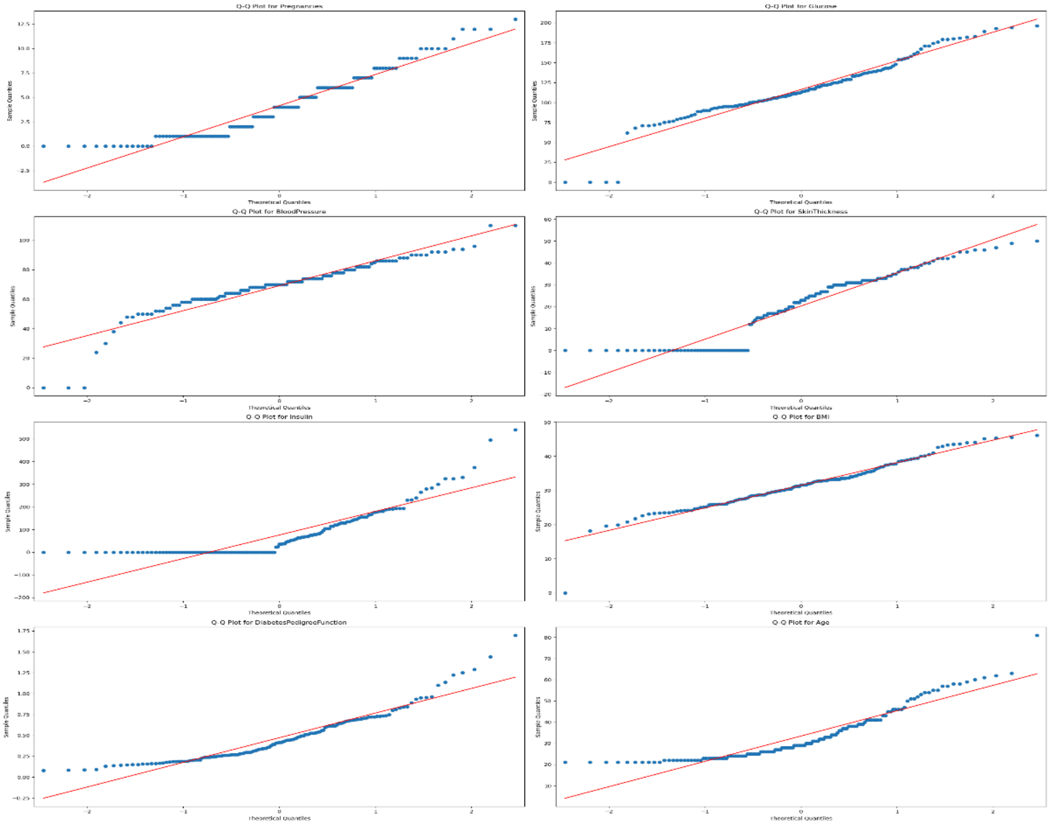
    - Calculating F1-score
- Analysis of the most appropriate classification model for this data set and justifying the probable underlying reason

EXPLORATORY DATA ANALYSIS

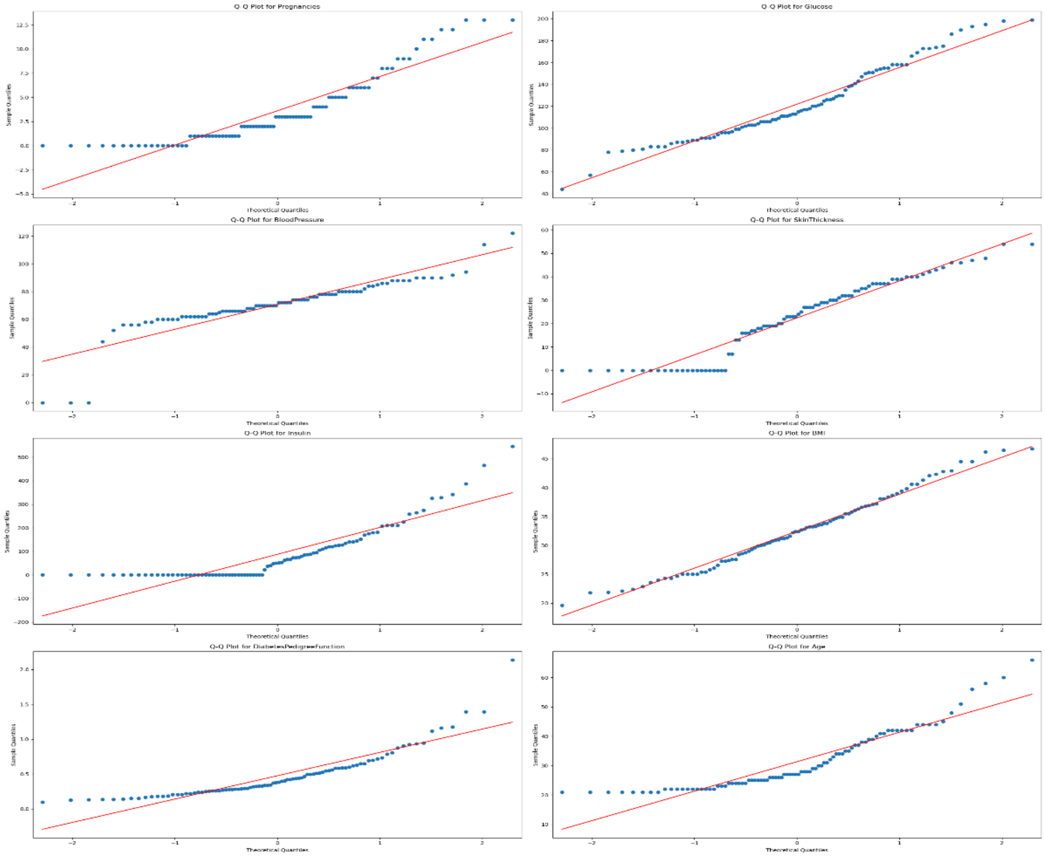
- **Scatterplot of all the variables**



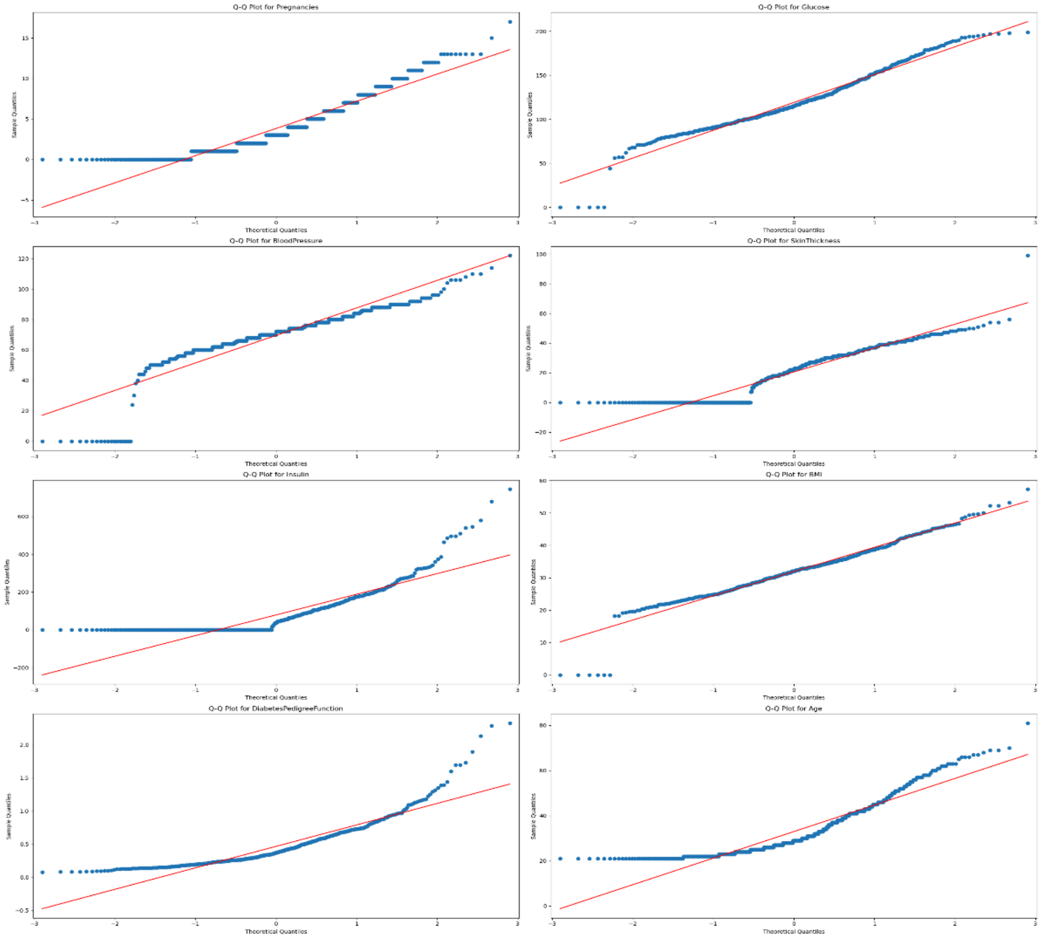
- **Test for normality**
  - **For class 0**


Q-Q Plot for Pregnancies
Q-Q Plot for Glucose
Q-Q Plot for BloodPressure
Q-Q Plot for SkinThickness
Q-Q Plot for Insulin
Q-Q Plot for BMI
Q-Q Plot for DiabetesPedigreeFunction
Q-Q Plot for Age
Sample Quantiles
Theoretical Quantiles

- **For class 1**


Q-Q Plot for Pregnancies
Q-Q Plot for Glucose
Q-Q Plot for BloodPressure
Q-Q Plot for SkinThickness
Q-Q Plot for Insulin
Q-Q Plot for BMI
Q-Q Plot for DiabetesPedigreeFunction
Q-Q Plot for Age
Sample Quantiles
Theoretical Quantiles

 For combined class 0 and 1



Q-Q Plot for Pregnancies

Q-Q Plot for Glucose

Q-Q Plot for BloodPressure

Q-Q Plot for SkinThickness

Q-Q Plot for Insulin

Q-Q Plot for BMI

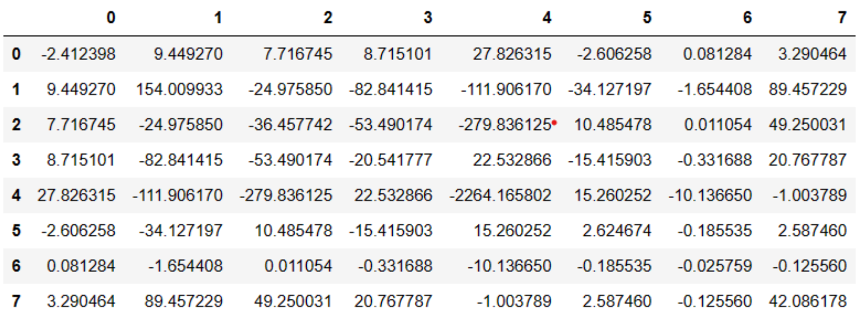
Q-Q Plot for DiabetesPedigreeFunction

Q-Q Plot for Age

Clearly from the above three sets of graphs, normality does not hold.

- **Covariance equality and independence condition**

  The below mentioned table gives the difference of the covariance matrices for class 0 and class 1.



	0	1	2	3	4	5	6	7
0	-2.412398	9.449270	7.716745	8.715101	27.826315	-2.606258	0.081284	3.290464
1	9.449270	154.009933	-24.975850	-82.841415	-111.906170	-34.127197	-1.654408	89.457229
2	7.716745	-24.975850	-36.457742	-53.490174	-279.836125	10.485478	0.011054	49.250031
3	8.715101	-82.841415	-53.490174	-20.541777	22.532866	-15.415903	-0.331688	20.767787
4	27.826315	-111.906170	-279.836125	22.532866	-2264.165802	15.260252	-10.136650	-1.003789
5	-2.606258	-34.127197	10.485478	-15.415903	15.260252	2.624674	-0.185535	2.587460
6	0.081284	-1.654408	0.011054	-0.331688	-10.136650	-0.185535	-0.025759	-0.125560
7	3.290464	89.457229	49.250031	20.767787	-1.003789	2.587460	-0.125560	42.086178

Observations:

1) For the condition of LDA to hold, that is the covariance matrices should be equal the above matrix should have all entries 0. Hence the condition of LDA does not hold

2) If features were independent between each class then off diagonal elements of this difference of the covariance matrices should be 0. But independence is not holding.

LOGISTIC REGRESSION

The logistic regression model:

$$\log log \frac{\pi}{1-\pi} = \beta_0 + \beta_1 X_1 + \dots + \beta_p X_p$$

Where π is the probability of success (probability that a breast mass is malignant), β_0 is the intercept and $\beta_1, \dots, \beta_p$ are the intercept of the corresponding feature.

An expression for π can be obtained from the above equation as:

$$\pi = \frac{e^{\beta_0+\beta_1 X_1+\dots+\beta_n X_n}}{1+e^{\beta_0+\beta_1 X_1+\dots+\beta_n X_n}}$$

Assumption for Logistic Regression satisfied?

To be able to apply logistic regression, multicollinearity should not be very high in the data. For this data average VIF (Variance Inflation Factor) came out to be 9.4 which is acceptable. So, logistic regression can be applied.

Steps of implementation:

1) Standardising the data
2) Fitting the logistic model on train data and including the intercept model
3) Predicting the probabilities of success
4) Find the best threshold value-
   Computing the difference between true positive rate and false positive rate. The corresponding threshold value for which this difference is maximum is the optimum value of threshold and then computing corresponding sensitivity and specificity
5) Plotting False positive rate on x-axis and true positive rate on y axis to plot the ROC curve and find the AUC

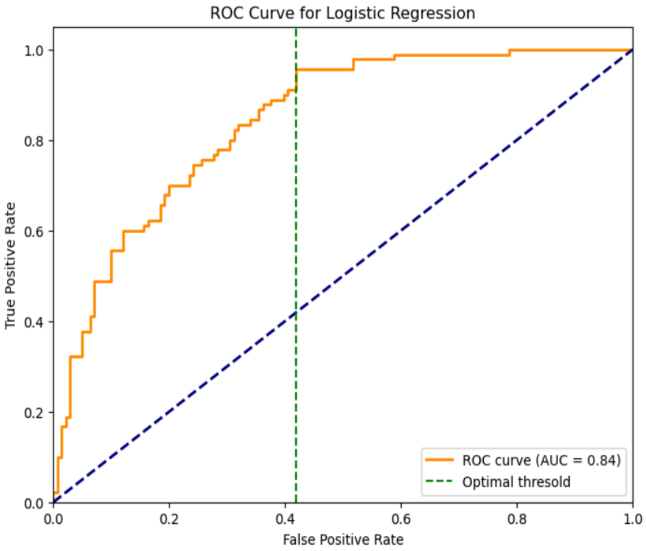
<u>RESULTS:</u>

1) AUC: 0.8424743892829
2) Best Threshold: 0.2053855868380612
   Corresponding TPR/Sensitivity: 0.9555555555555556
   Corresponding Specificity: 0.581560283687943
3) ROC curve:

   An ROC is a graph showing the performance of a classification model at all classification thresholds. It plots two parameters: True Positive Rate (TPR/Sensitivity) and False Positive Rate (FPR/1-Specificty).

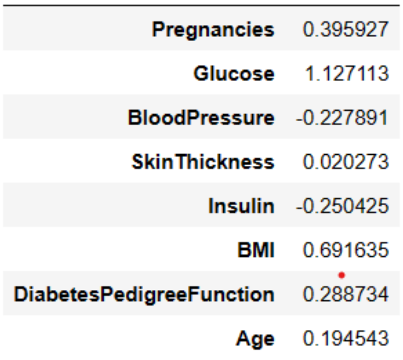
   The formulas are given below:

$$TPR = \frac{TP}{TP+FN} \qquad FPR = \frac{FP}{FP+TN}$$



4) F1 score: 0.7319148936170213

5) Regression coefficients:

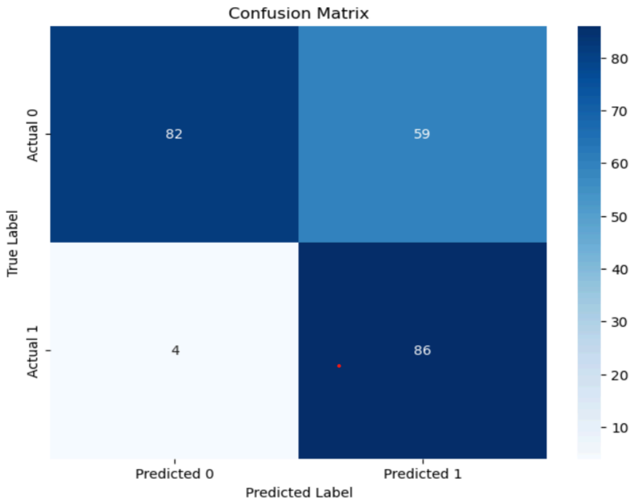


Pregnancies	0.395927
Glucose	1.127113
BloodPressure	-0.227891
SkinThickness	0.020273
Insulin	-0.250425
BMI	0.691635
DiabetesPedigreeFunction	0.288734
Age	0.194543

6) Confusion matrix

A Confusion Matrix is a matrix that summarizes the performance of a model on a set of test data. We fit the model obtained using the train dataset and apply it on the test dataset and check whether it aligns with the response.

It displays 4 values: True Positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN). The confusion matrix obtained from the logistic model is shown:



K- Nearest Neighbours

What is KNN?

K-Nearest Neighbours examines the labels of a chosen number of data points surrounding a target data point, in order to make a prediction about the class that the data point falls into.

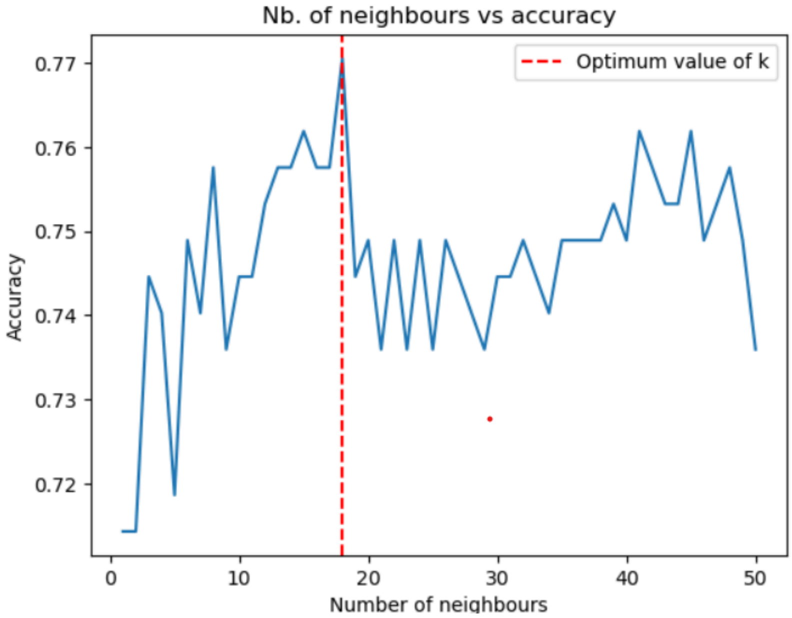
KNN doesn't make any assumptions about the data, meaning it can be used for a wide variety of problems.

Steps of implementation:

1) Standardising the data
2) Calculating the accuracy over various values of k (number of neighbours) in a range. The plot of k on x-axis vs accuracy on y-axis is made and the optimum value of k is selected, that is, the value of k for which accuracy is highest.
3) We fit the KNN model to the train data by taking k=18 and considering weights as inverse of distance.
4) The predicted probabilities P(Y=1), for this model is found for the test data.
5) Finding the optimum threshold:
   Computing the difference between true positive rate and false positive rate. The corresponding threshold value for which this difference is maximum is the optimum value of threshold and then computing corresponding sensitivity and specificity
6) Plotting False positive rate on x-axis and true positive rate on y axis to plot the ROC curve and find the AUC

RESULTS:

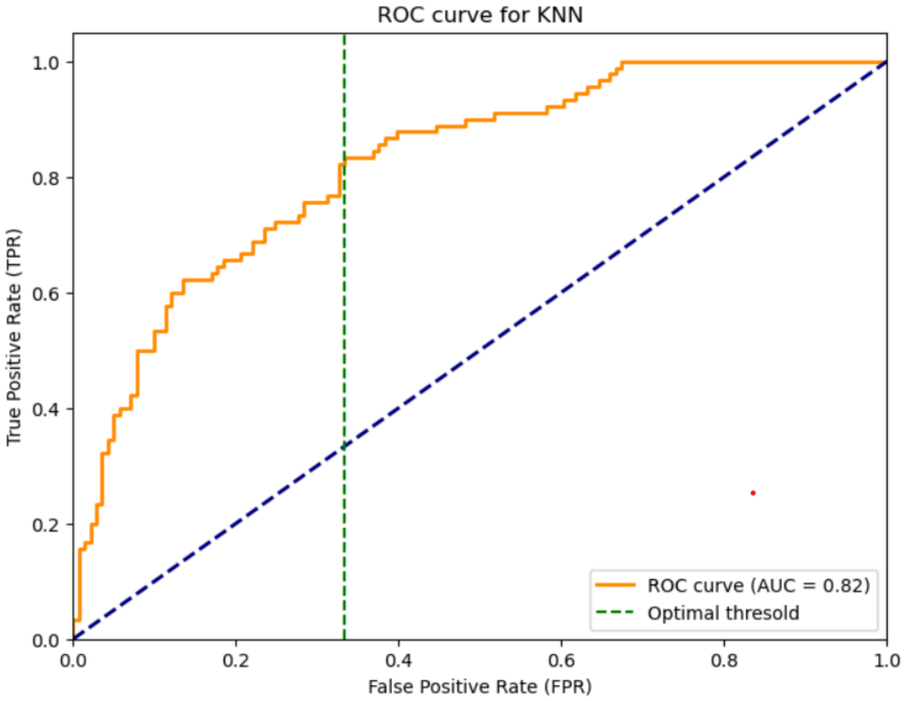
1) Accuracy curve



2) AUC=0.82

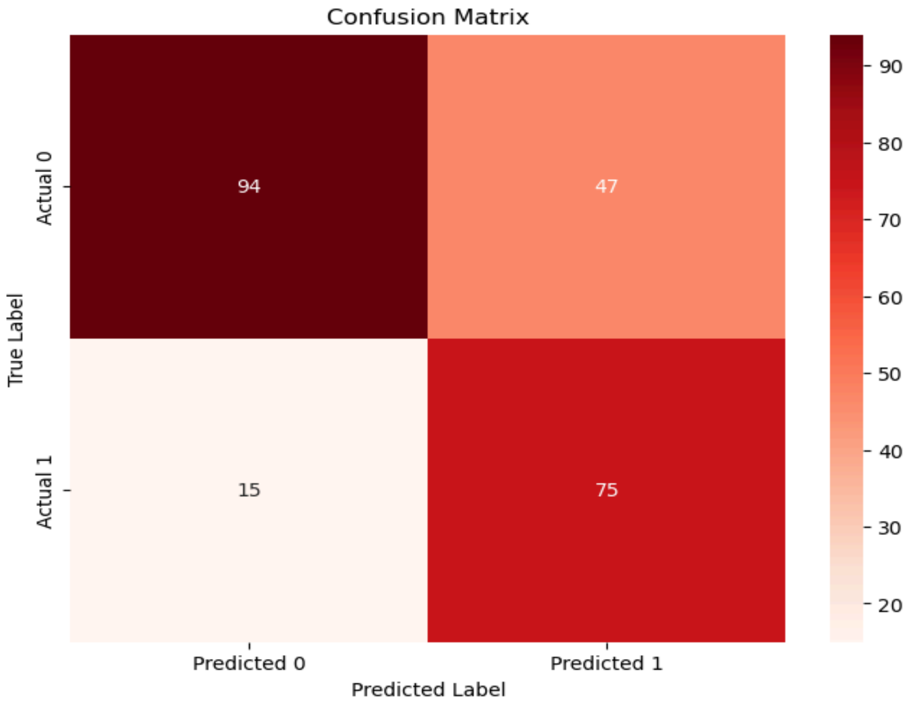
3) Best Threshold: 0.3006326569577335
Corresponding TPR/Sensitivity: 0.8333333333333334
Corresponding Specificity: 0.6666666666666667

4) ROC curve



5) F1-score: 0.6503067484662578

6) Confusion Matrix



LDA AND QDA

LDA:

What are assumptions of LDA?

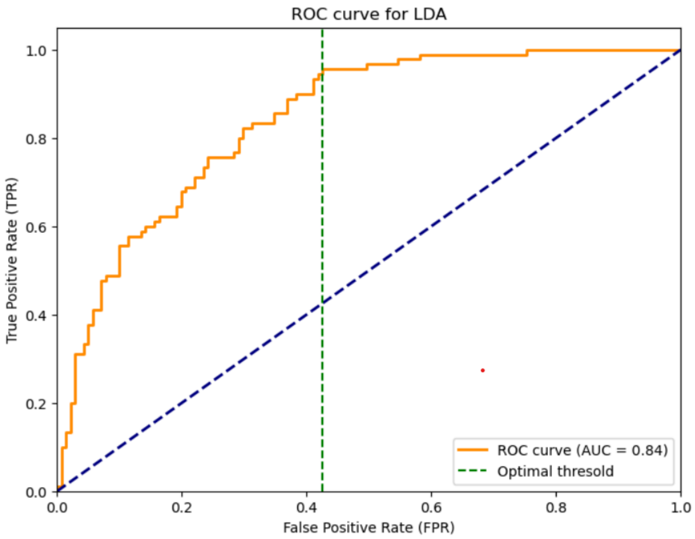
LDA makes a lot of assumptions, such as the 1) sample measurements are independent from each other, 2) distributions are normal, and 3) co-variance of the measurements are identical across different classes.

Steps of implementation:

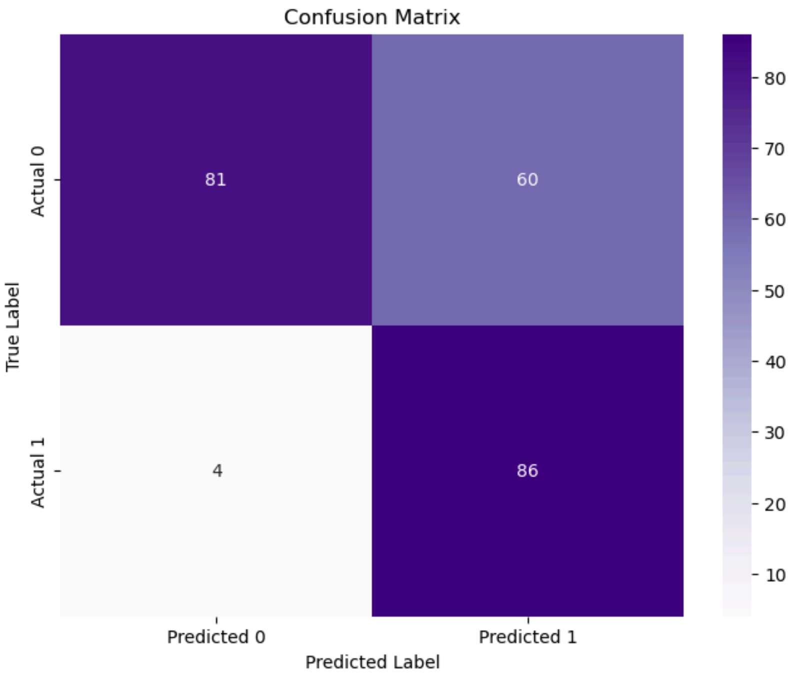
1) Standardising the data
2) Fitting the linear discriminant analysis model on train data and including the intercept model
3) Predicting the probabilities of success
4) Find the best threshold value-
   Computing the difference between true positive rate and false positive rate. The corresponding threshold value for which this difference is maximum is the optimum value of threshold and then computing corresponding sensitivity and specificity
5) Plotting False positive rate on x-axis and true positive rate on y axis to plot the ROC curve and find the AUC

RESULTS

1) AUC: 0.8426319936958234
2) Best Threshold: 0.19660891986862056
   Corresponding TPR/Sensitivity: 0.9555555555555556
   Corresponding Specificity: 0.574468085106383
3) ROC curve



4) F1 score: 0.728813559322034
5) Confusion matrix



QDA:

What is QDA?

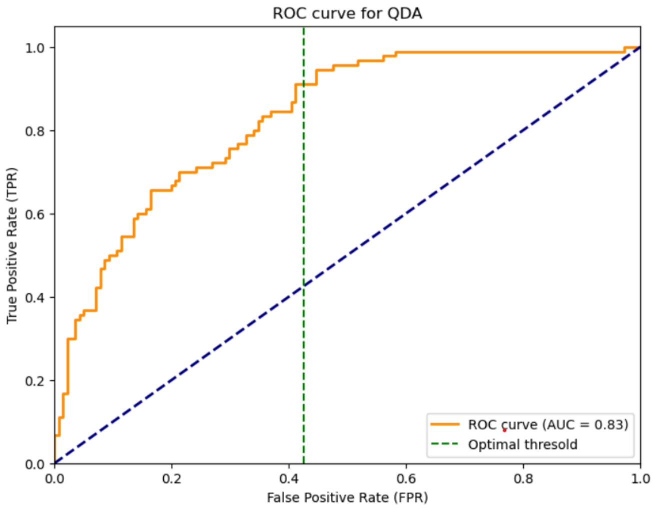
Like LDA, the QDA classifier assumes that the observations from each class of *Y* are drawn from a Gaussian distribution. However, unlike LDA, QDA assumes that each class has its own covariance matrix.

Steps of implementation:

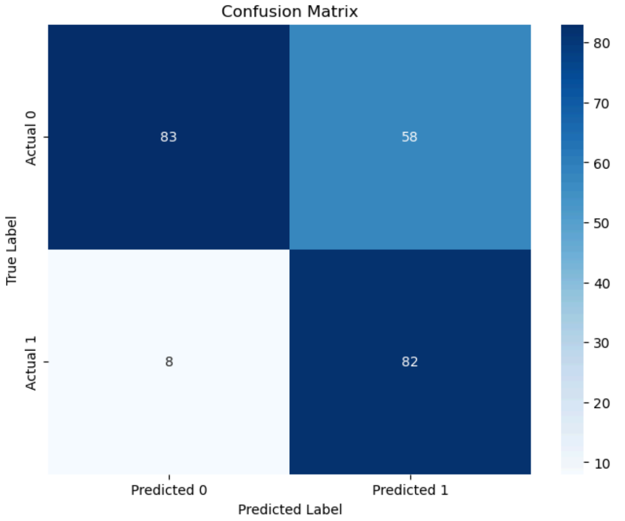
On similar lines as that of linear discriminant analysis.

RESULTS

1) AUC: 0.83
2) Best Threshold: 0.1757112643115787
   Corresponding TPR/Sensitivity: 0.9111111111111111
   Corresponding Specificity: 0.5886524822695036
3) ROC curve



4) F1 score: 0.713043478260869
5) Confusion matrix



NAÏVE BAYES

Assumptions

It assumes that predictors in a Naïve Bayes model are conditionally independent, or unrelated to any of the other feature in the model.

As conditional density of the feature is not gaussian so we estimate the joint density by KDE method.

Steps of implementation:

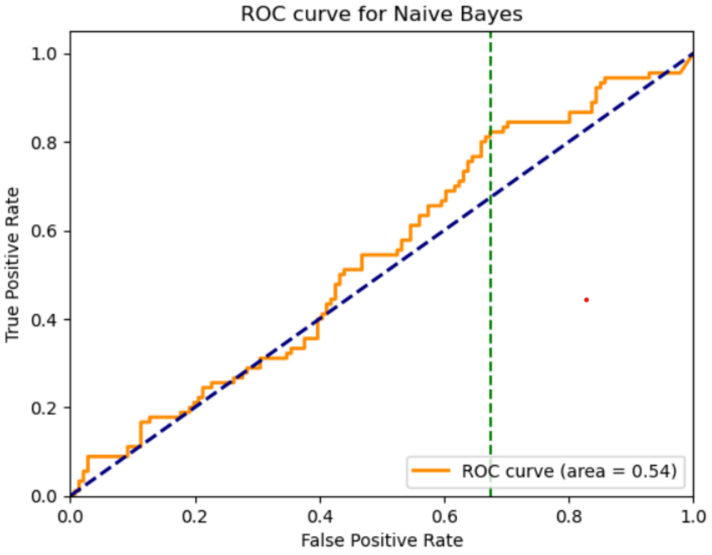
On similar lines as that of LDA and QDA.

RESULTS

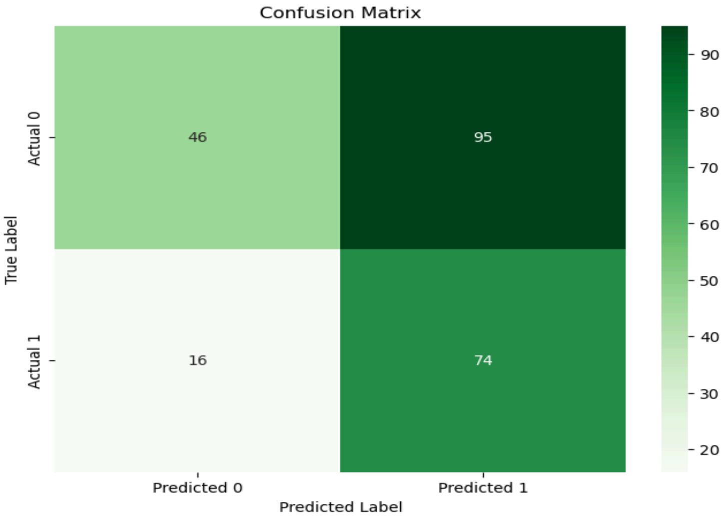
1) AUC : 0.54
2) Best Threshold: 2.4505008152457274e-06

Corresponding TPR/Sensitivity: 0.8222222222222222
Corresponding Specificity: 0.3262411347517731

3) ROC curve



4) Confusion matrix



DECISION TREE

Decision Tree 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.**

Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.

It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions. A decision tree simply asks a question, and based on the answer (Yes/No), it further splits the tree into subtrees.

The following steps are involved in the decision tree algorithm:

1. We start with the root node, which contains the entire dataset.
2. We find the best attribute in the dataset using Attribute Selection Measure.
3. The root node is divided into subsets that contains possible values for the best attributes.
4. The decision tree node is generated which contains the best attribute.
5. New decision trees are recursively made using the subsets of the dataset. This is continued till a stage where the nodes cannot be further classified.

Attribute Selection Measures:

- Information Gain: measures changes in entropy after the segmentation of a dataset based on an attribute. It calculates how much information a feature provides about the class.

$$= Entropy(S) - [(Weighted\ Avg) \times Entropy(each\ feature)$$

  Where $Entropy(S) =- P(M) \log log\ (2)\ P(M) - P(B) \log log\ (2)\ P(B)$
- Gini's Index: An attribute with low Gini's index is preferred over a high one. It is given by

  $1 - \sum_J P_J^2.$

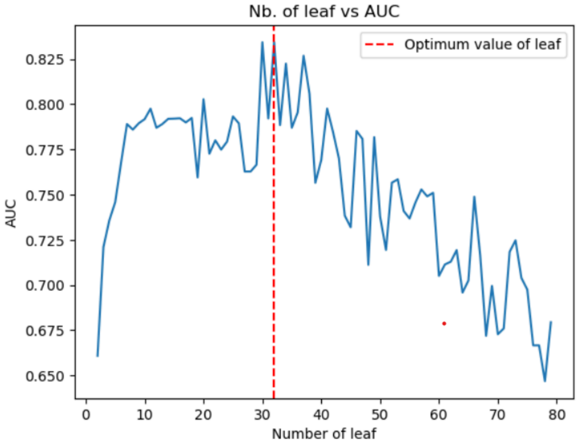
In order to select the optimal number of nodes, we plot the difference of TPR and FPR against the number of nodes. We select that node as optimal which has the maximum difference

Steps of implementation:

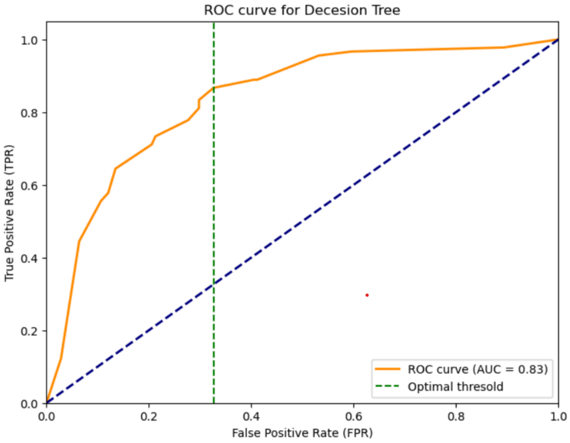
1) Standardising the data
2) Finding the optimum number of leaves by maximising the AUC value
3) Fitting the decision tree classifier model on train data by taking optimum number of leaves as 32
4) Finding predicted probabilities on the test data
5) Finding the optimum threshold:
   Computing the difference between true positive rate and false positive rate. The corresponding threshold value for which this difference is maximum is the optimum value of threshold and then computing corresponding sensitivity and specificity
6) Plotting False positive rate on x-axis and true positive rate on y axis to plot the ROC curve and find the AUC
7) Printing the final decision tree

RESULTS

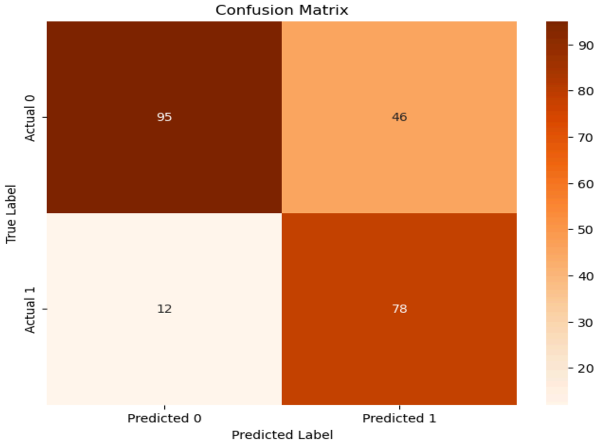
1) AUC vs number of leaves plot



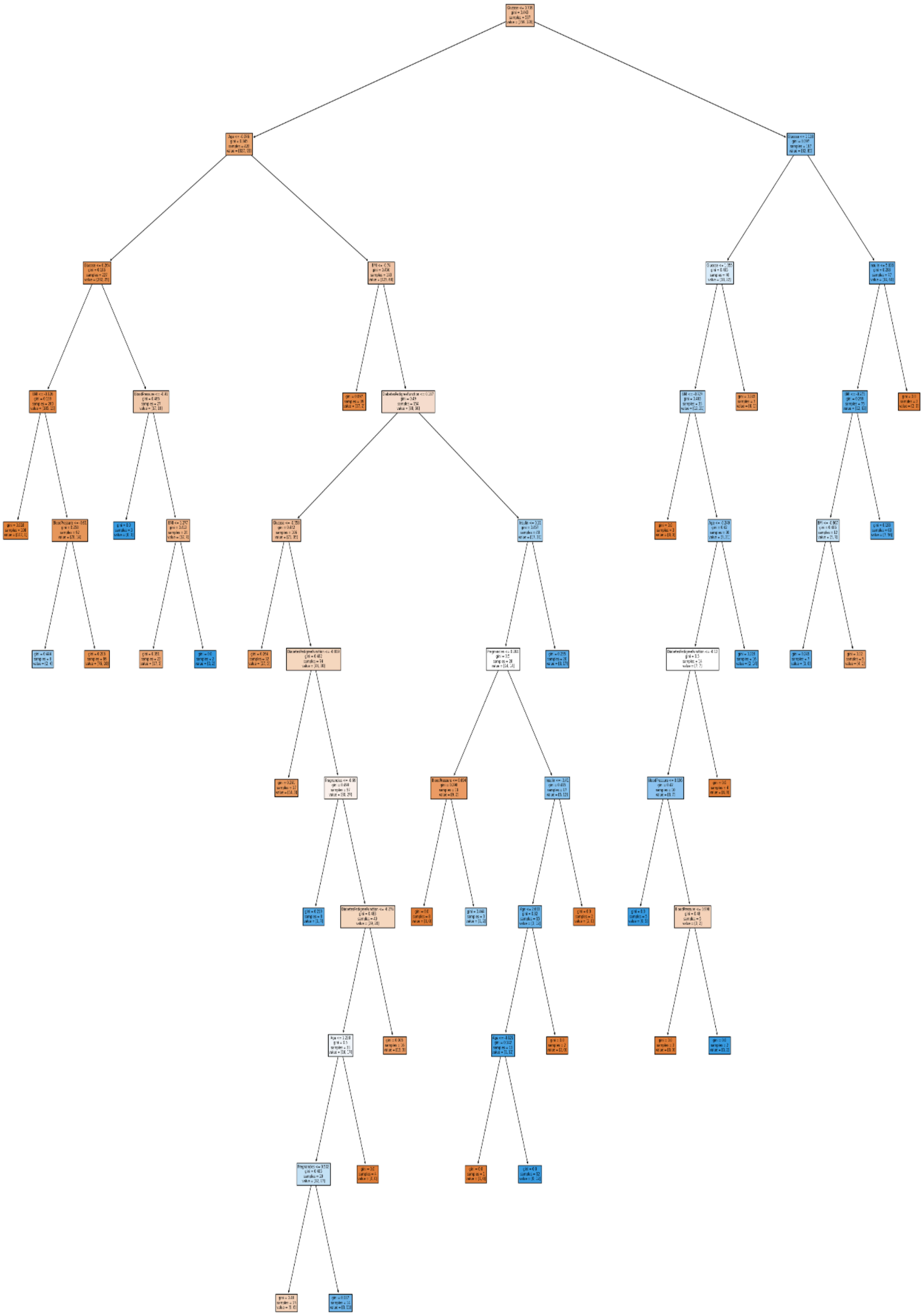
2) AUC: 0.83
3) Best Threshold: 0.17647058823529413
   Corresponding TPR/Sensitivity: 0.8666666666666667
   Corresponding Specificity: 0.6737588652482269
4) ROC curve



5) Confusion Matrix



6) Decision tree diagram



Random Forest Classifier

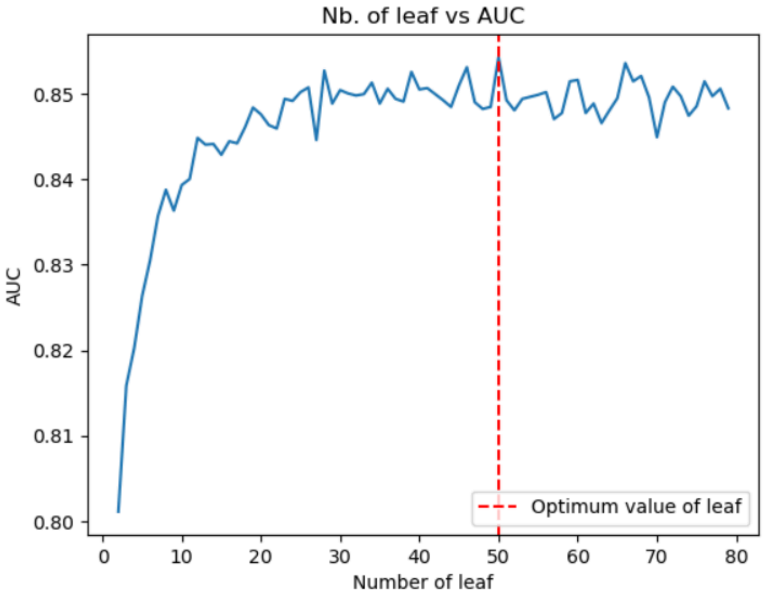
Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset.

Steps of implementation:

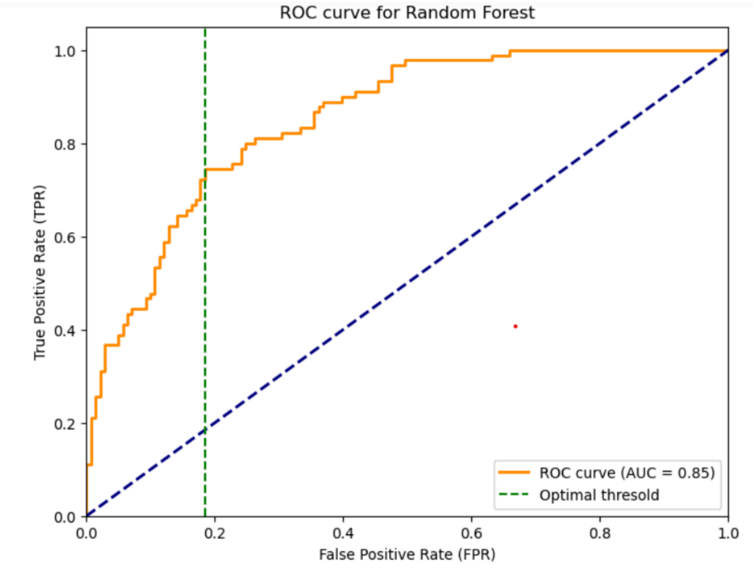
1) Standardising the data and finding the optimum number of leaves by maximising the AUC value
2) Fitting the Random Forest classifier model on train data by taking optimum number of leaves as 50 as shown in result1
3) Finding predicted probabilities on the test data
4) Finding the optimum threshold:
   Computing the difference between true positive rate and false positive rate. The corresponding threshold value for which this difference is maximum is the optimum value of threshold and then computing corresponding sensitivity and specificity
5) Plotting False positive rate on x-axis and true positive rate on y axis to plot the ROC curve and find the AUC

RESULTS

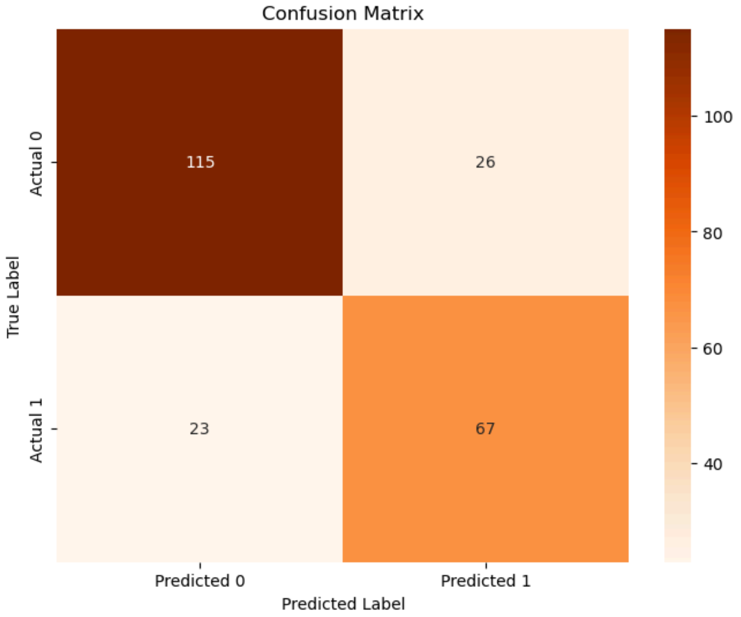
1) AUC vs number of leaves plot



2) AUC: 0.85
3) Best Threshold: 0.43253785128637035
   Corresponding TPR/Sensitivity: 0.7444444444444445
   Corresponding Specificity: 0.8156028368794326
4) ROC curve



5) Confusion Matrix



CONCLUSION

Summary

Model	AUC	Sensitivity	Specificity
Logistic	0.842	0.96	0.58
KNN	0.82	0.83	0.67
LDA	0.83	0.96	0.57
QDA	0.83	0.91	0.58
Naïve Bayes	0.54	0.82	0.33
Decision tree	0.83	0.87	0.68
Random Forest	0.85	0.74	0.81

Observations:

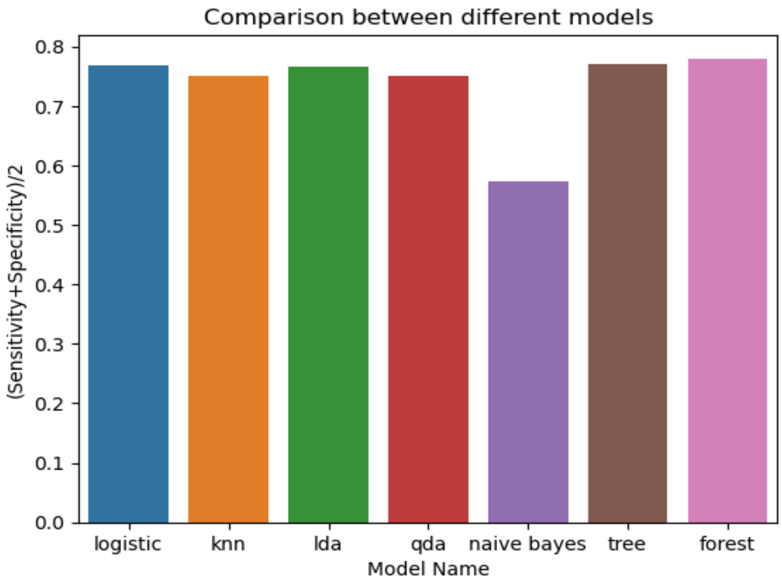
1) As seen from EDA, the independence condition does not hold, therefore Naïve Bayes performs poorly.
2) Only the Random Forest classifier could capture give good specificity.
3) The conditions of LDA and QDA do not hold as seen from EDA, hence it does not perform well and its specificity is quite low.

Sensitivity and specificity

In our problem we would want to have a high value of both sensitivity and specificity. This is because if sensitivity is low then the patients who actually have diabetes will not be detected while if specificity is low then the patients who actually do not have diabetes will also be classified as diabetic patients and then they will be put under unnecessary medication. So, we compute a measure to select the best model-

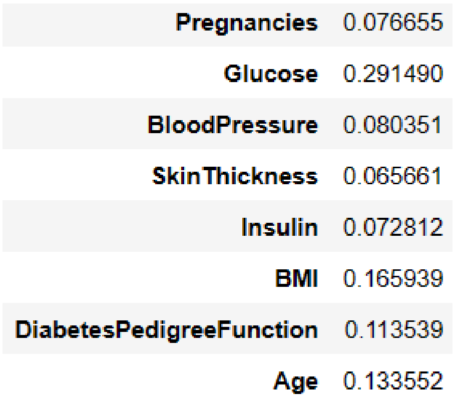
(Sensitivity+ Specificity)/2 at the best threshold

- The below mentioned graph shows the bar plot of the above measure for the 7 models.



Observation: It can be seen that with reference to our measure, Naïve Bayes performs most poorly while Random Forest performs the best with a small margin.

- We next proceed towards interpreting the feature importance of the explanatory variables using the best i.e. Random Forest model.



Pregnancies	0.076655
Glucose	0.291490
BloodPressure	0.080351
SkinThickness	0.065661
Insulin	0.072812
BMI	0.165939
DiabetesPedigreeFunction	0.113539
Age	0.133552

It can be seen that feature importance of glucose is the highest that is, the presence of glucose in blood highly impacts the result that if a person is diabetic or not.
Number of pregnancies is shown to not affect much if a person is diabetic or not.

Hence Random Forest Classifier is the most appropriate classification model among the seven.