A MAJOR PROJECT REPORT
ON DIABETES PREDICTION USING ML

Ms. Arti Roll No: 204200035	Mr. Anmol Bhardwaj Roll No:204200028
Mr. Bishal Roy Roll No:204200048	Mr. Hariram Roll No: 204200075
Mr. Devansh Lauhariya Roll No: 204200060	Mr. Abhishek Chandel Roll no: 204200005
Mr. Amit Prakash Roll no: 204200020	

Under the Guidance of Mr. Narendra Mohan (Assistant Professor)
(Department of Computer Engineering & Applications)

ऋते ज्ञानान्न मुक्तिः

GLA UNIVERSITY
MATHURA
Recognised by UGC Under Section 2(f)

Accredited with A+ Grade by NAAC

12-B Status from UGC

Institute of Engineering &

Technology GLA UNIVERSITY

281406, INDIA

DECLARATION

We declare that the work which is being presented in B.C.A, project "DIABETES PREDICTION”, is in partial fulfilment of the requirements for award of Bachelor of Computer Applications are submitted to the department of Computer Engineering and Applications of GLA University, Mathura, is an authentic record of our own work carried under the supervision Mr. Narendra Mohan, Assistant Professor.

The contents of the project report, in full or in parts, have not been submitted to any other institute or university for the award of any degree.

Ms. Arti
Roll No: 204200035

Mr. Anmol Bhardwaj
Roll No:204200028

Mr. Bishal Roy
Roll No:204200048

Mr. Hariram
Roll No: 204200075

Mr. Devansh Lauhariya
Roll No: 204200060

Mr. Abhishek Chandel
Roll No: 204200005

Mr. Amit Prakash
Roll No: 204200020

-: CERTIFICATE: -

This is to certify that the above statements made by the candidates are correct to the best of my/our knowledge and belief.

Project Supervisor
(Mr. Narendra Mohan)

Date:-

Project in Charge
(Mr. D.P. Yadav)

Program Coordinator
(Mr. Narendra Mohan)

Head Of Department
(Dr. Shashi Shekhar)

ACKNOWLEDGEMENT

We would like to thank GLA University for proposing the Major Project in our curriculum so that we can learn new concepts easily by doing hands-on. We would like to thank BCA department for encouraging us to learn more by doing practical. We would like to extend my regards to my mentor and my project guide **Mr. Narendra Mohan** who have helped us in getting this project ready.

They have always stood by us, whenever we needed them, and they have done a lot for us in order to get this project completed. Without their valuable help, this project would not have been completed. We would also like to thank the Internet for providing a large source of information that we needed for completing this project. Once, again We would like to thank all the people involved in this project from the bottom of my heart, and please forgive us if we have forgotten to mention any name. This project is completed with your help only.

We would also like to extend our immense gratitude to respected **Head of Department Prof. Dr. Shashi Shekhar** who allowed us to choose the topic for our dissertation. The experience was novel one and we would like to thank all the people , who have lent their valuable time for the completion of the report. Without their consideration it would have been difficult to

Complete the report

Ms. Arti	(Roll No: 204200035)	________________________
Mr. Anmol Bhardwaj	(Roll No:204200028)	________________________
Mr. Bishal Roy	(Roll No:204200048)	________________________
Mr. Hariram	(Roll No: 204200075)	________________________
Mr. Devansh Lauhariya	(Roll No: 204200060)	________________________
Mr. Abhishek Chandel	(Roll No: 204200005)	________________________
Mr. Amit Prakash	(Roll No: 204200020)	________________________

ABSTRACT

Diabetes is a chronic disease with the potential to cause a worldwide health care crisis. According to International Diabetes Federation 382 million people are living with diabetes across the whole world. By 2035, this will be doubled as 592 million. Diabetes is a disease caused due to the increase level of blood glucose. This high blood glucose produces the symptoms of frequent urination, increased thirst, and increased hunger. Diabetes is a one of the leading cause of blindness, kidney failure, amputations, heart failure and stroke. When we eat, our body turns food into sugars, or glucose. At that point, our pancreas is supposed to release insulin. Insulin serves as a key to open our cells, to allow the glucose to enter and allow us to use the glucose for energy. But with diabetes, this system does not work. Type 1 and type 2 diabetes are the most common forms of the disease, but there are also other kinds, such as gestational diabetes, which occurs during pregnancy, as well as other forms. Machine learning is an emerging scientific field in data science dealing with the ways in which machines learn from experience. The aim of this project is to develop a system which can perform early prediction of diabetes for a patient with a higher accuracy by combining the results of different machine learning techniques. The algorithms like K nearest neighbor, Logistic Regression, Random forest, Support vector machine and Decision tree are used. The accuracy of the model using each of the algorithms is calculated. Then the one with a good accuracy is taken as the model for predicting the diabetes.

Keywords:- XGB Classifier, Naïve Bayes, Support vector machine, Decision tree, Logistic Regression, Define grid search

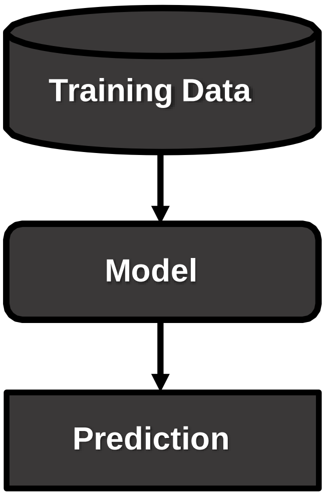
INDEX

List of Figure

List of Table: -

INTRODUCTION

Diabetes is noxious diseases in the world. Diabetes caused because of obesity or high blood glucose level, and so forth. It affects the hormone insulin, resulting in abnormal metabolism of crabs and improves level of sugar in the blood. Diabetes occurs when body does not make enough insulin. According to (WHO) World Health Organization about 422 million people suffering from diabetes particularly from low or idle income countries. And this could be increased to 490 billion up to the year of 2030. However prevalence of diabetes is found among various Countries like Canada, China, and India etc. Population of India is now more than 100 million so the actual number of diabetics in India is 40 million. Diabetes is major cause of death in the world. Early prediction of disease like diabetes can be controlled and save the human life. To accomplish this, this work explores prediction of diabetes by taking various attributes related to diabetes disease .



-: A Basic M.L Model: -

OBJECTIVE

The objectives of the system are-

- To find out an efficient machine learning method that is used in the diabetic prediction.
- Study the classification ,diagnosis and treatment of heart disease.
- To identify the ideal clinical symptoms of Diabetes.
- To evaluate the machine learning algorithms

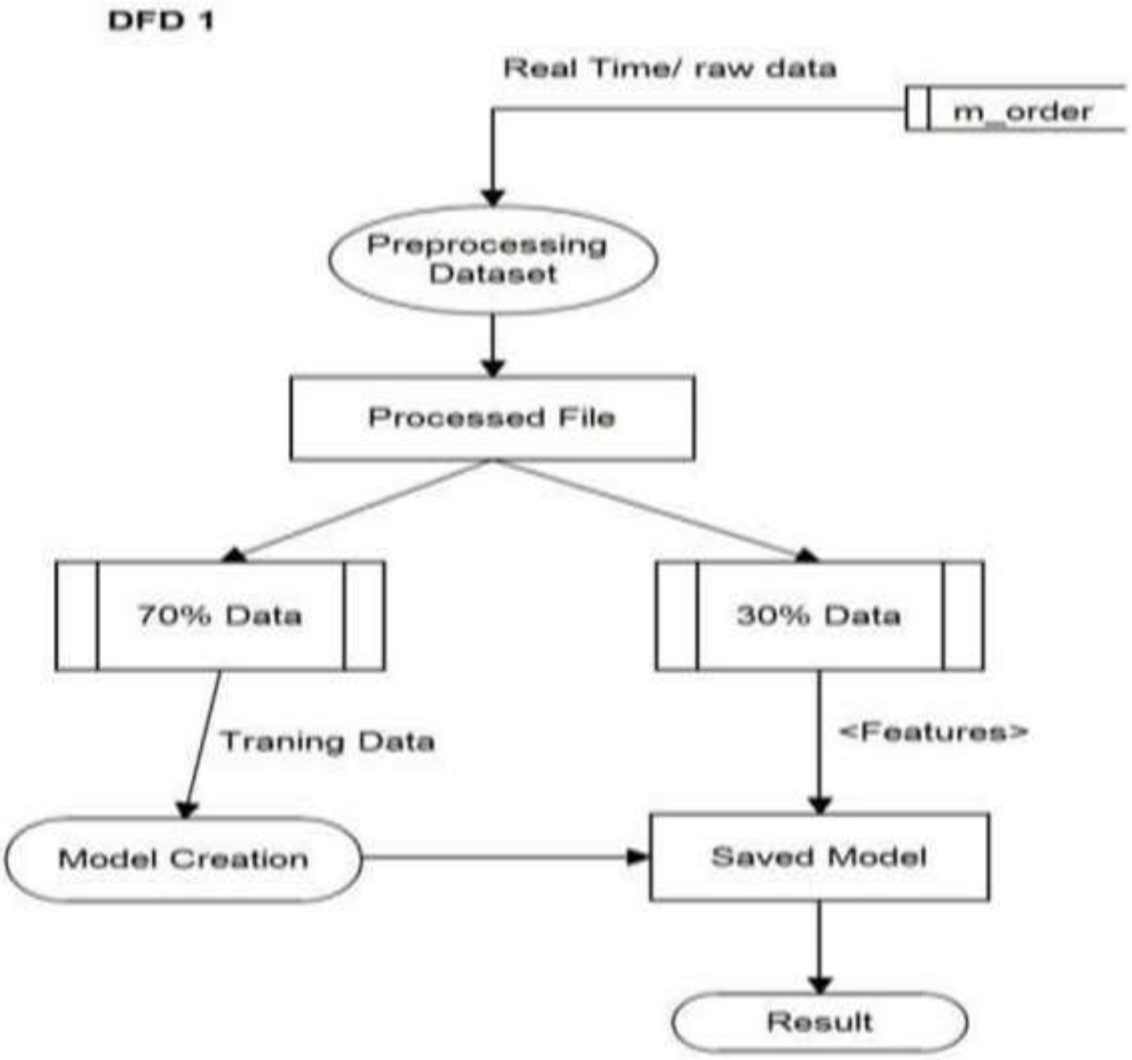
MOTIVATION

- The false-negative type in which a patient in reality is already a diabetic patient but test results tell that the person is not having diabetes.
- The false-positive type. In this type, patient in reality is not a diabetic patient but test reports say that he/she is a diabetic patient

ORGANIZATION OF PROJECT REPORT

PHASES	TIME DURATION
Software Requirement Specification	2 WEEK
System Design	3 WEEK
Coding	6 WEEK
Documentation	2 WEEK
Implementation	1 WEEK

DATA FLOW DIAGRAM



MACHINE LEARNING ALGORITHMS

The Models we are going to use for the prediction purpose are as follows:

1. Logistic Regression:

- It comes under the umbrella of Supervised machine learning.
- It is used for calculating value of categorical dependent variable(y) by using a given set of independent variables(x).
- It predicts the output of a categorical dependent variable. Therefore, outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.
- It is used for solving classification problems.
- In this, instead of fitting a regression line, we fit an "S" shaped sigmoid logistic function, which gives two max values (0 or 1).
- It is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.

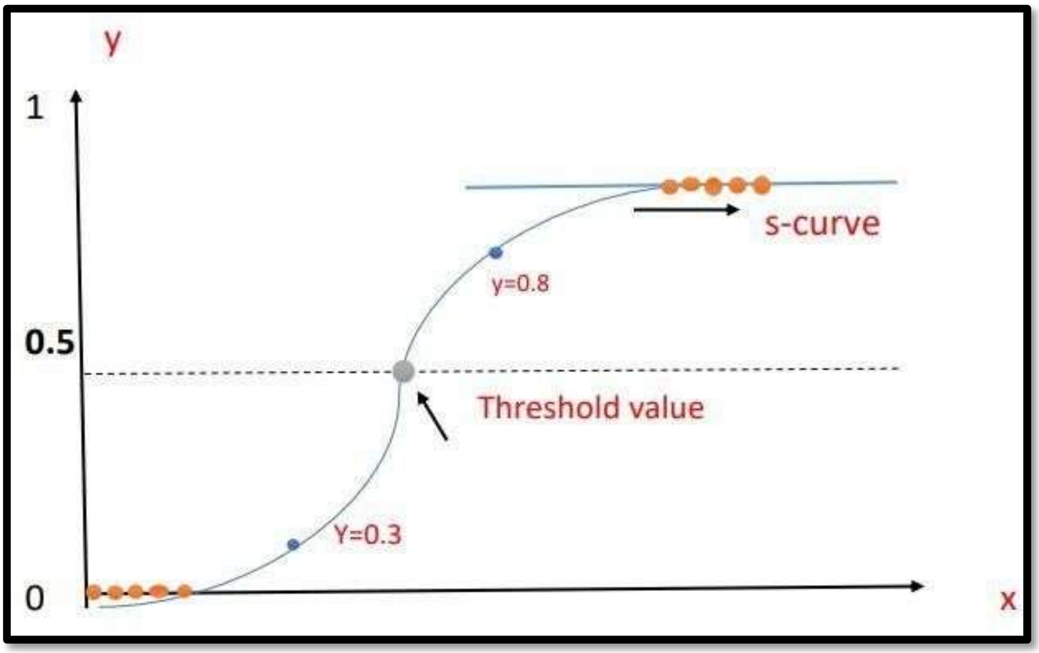


Figure 2.0 Graph of Logistic Regression

- It is a predictive model as regression; therefore, it is called logistic regression, but is used to classify samples; Therefore, it falls under the classification algorithm.

Logistic Function (Sigmoid Function):

- The sigmoid function is a mathematical function used to map the predicted values to probabilities.
- It maps any real value into another value within a range of 0 and 1.
- The value of the logistic regression must reside between 0 and 1, which can't go beyond this limit, so it forms a curve like the "S" form.
- The S-form curve is called the logistic function or Sigmoid function.
- In logistic regression, we use the concept of the threshold value, which defines the probability of either 0 or 1. Such as values above the threshold value tends to 1, and a value below the threshold values tends to 0.

Assumptions for Logistic Regression:

- The dependent variable need to be categorical.
- The independent variable should not have multi-collinearity with other independent variables.

Logistic Regression Equation:

$$log\left[\frac{y}{1-y}\right] = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \cdots + b_nx_n$$

Steps in Logistic Regression:

- Data Pre-processing.
- Fitting model to Training set.
- Predicting test result.
- Check Test accuracy of the result (Confusion matrix, f1 score, accuracy score, heat map, etc.)
- Visualizing the test set result (bar graph, line graph, scatterplot, pie chart, etc.).

2. K-Nearest Neighbour(KNN):

- It comes under the umbrella of Supervised Machine learning.
- K-NN algorithm assumes similarity b/w the new case or data and available cases and put the new case into the category that is most similar (more close to new data or case) to the available categories.

- It is a non-parametric algo, which means it does not make any assumption on underlying data.
- Also known as lazy learner ago because it does not learn from the training set immediately in place of that it stores dataset and at time of classification, it performs required action on dataset.

Why do we need a K-NN Algorithm?

- Suppose there are two categories, i.e., Category A and Category B, and we have a new data point x1, so this data point will lie in which of these categories. To solve this type of problem, we need a K-NN algorithm. With the help of K-NN, we can easily identify category or class of a particular dataset. Consider the below diagram:

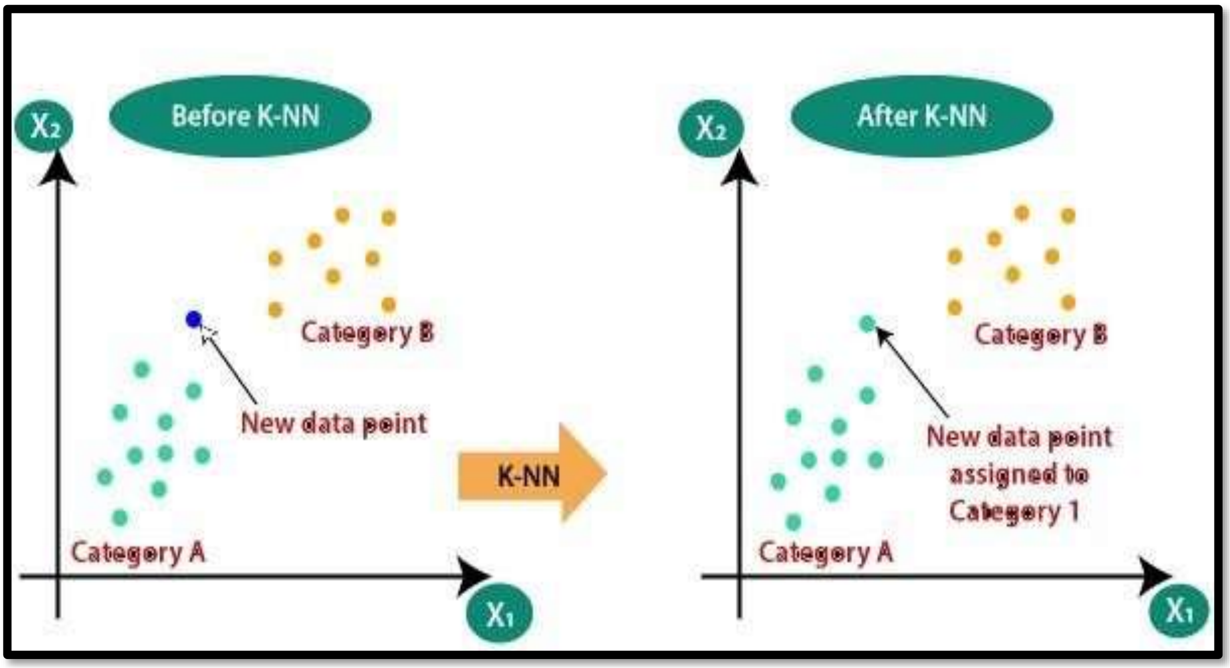


Figure 3.0 KNN Graphical Representation

How does K-NN work?

Its working can be explained on the basis of the below algorithm:

- Set value of K (total number of neighbors in the proximity).
- Calculate Euclidean distance of K number of neighbors.
- Take K nearest neighbours according to calculated Euclidean distance.
- Among these neighbours, count the number of t data points in each category.
- Assign new data point(s) to that category who is having the maxnumber of neighbours in comparison to other.
- Now or model is good to go.

Euclidean Equation: -

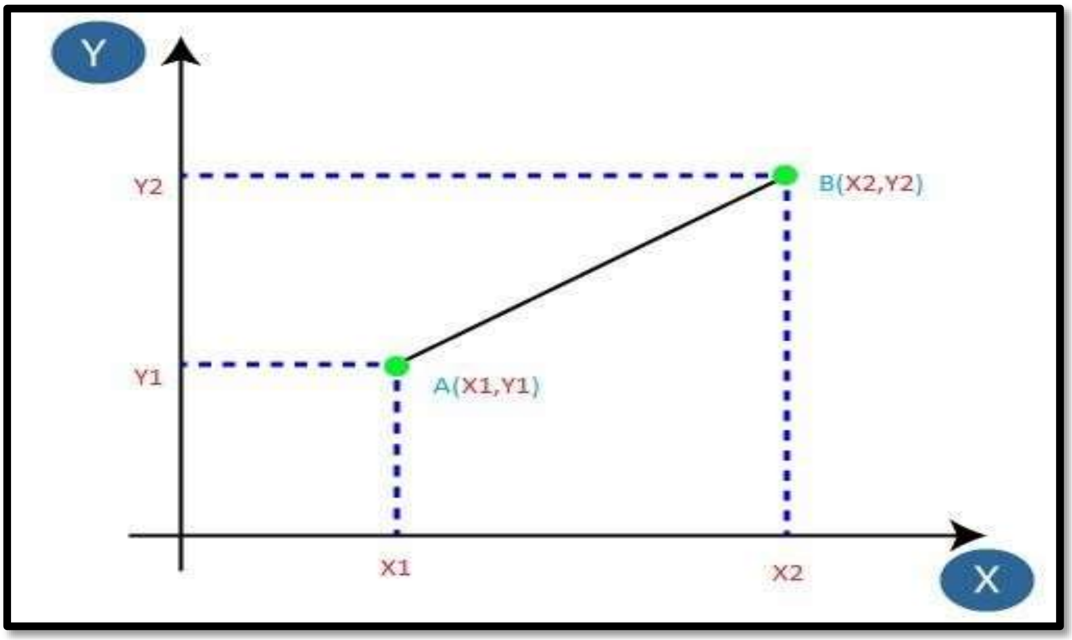


Figure 3.1 Representation of Euclidean Equation

Euclidian distance between A and B

= 1/2[(X2-X1) ^2 +(Y2-Y1) ^2]

How to select the value of K in the K-NN Algorithm?

- There is no specified way of determining the best value for 'K’, so we need to go for hit and trial. By default, we go for value of 'k' = 5(most preferred).
- A very low value for K such as K=1 or K=2, can lead to Overfitting and the effects of outliers in model.
- Quite large values for K can lead to under fitting.
- Medium values (neither too small nor too large) works fine with the model.

3. XGB Classifier:

- n gradient boosting decision trees, we combine a lot of weak learners for getting one strong learner.
- The weak learners are the individual decision trees. All the trees are connected in sequence and each tree tries to minimize error of the previous tree.
- The weak learners are fit in such a way that each new learner fits into the residuals of the previous step so as the model improves.

- Final model does the submission of the result of each step and thus a stronger learner is eventually achieved.

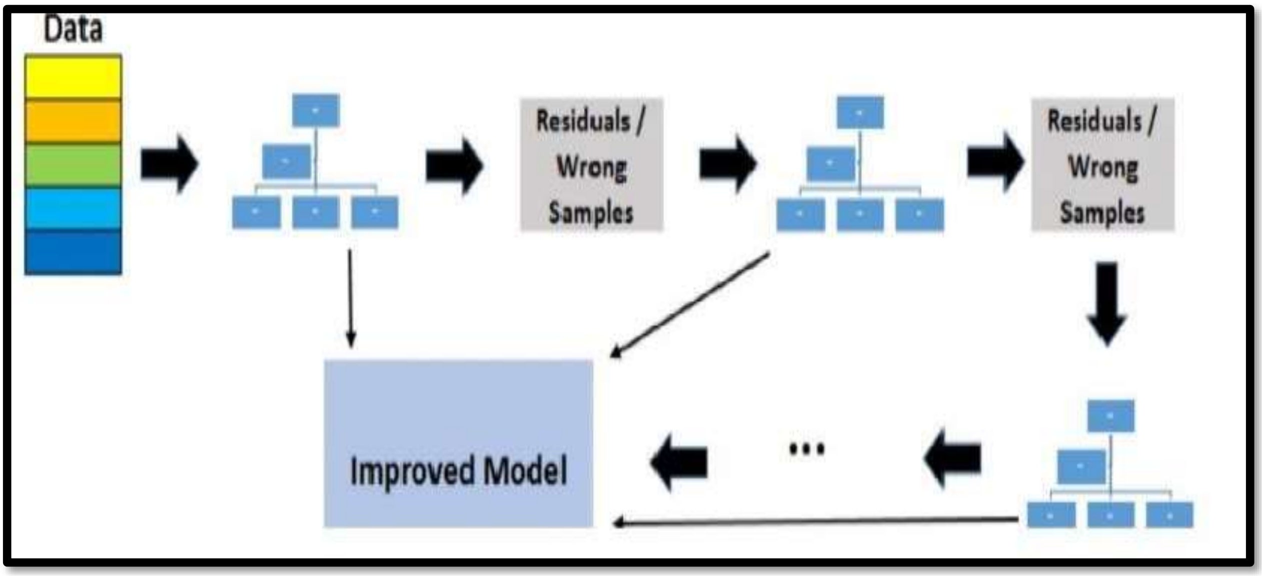


Figure 4.0 Diagram of Gradient Boost

- **This type of boosting has three main components: -**

A. **Loss Function –** The role of the loss function is to estimate how good the model is at making predictions with the given data. For example, if we're trying to predict the weight of a person depending on some input variables (a regression problem), then the loss function would be something that helps us find the difference between the predicted weights and the observed weights. On the other hand, if we're trying to categorize if a person will like a certain movie based on their personality, we'll require a loss function that helps us understand how accurate our model is at classifying people who did or didn't like certain movies.

B. **Weak Learner –** A weak learner is one that classifies our data but with high error rate. These are typically decision trees (also known as decision stumps).

C. **Additive Model –** This is the iterative and sequential approach of adding trees (weak learners) one step at a time. After each iteration, we need to be closer to our final model. In other words, each iteration should reduce the value of our loss function.

<u>Steps to work with gradient Boosting: -</u>

- Calculate average/mean of the target variable for first model.
- Calculate residuals for each sample.

residual = Actual value - Predicted value

- Construct a decision tree. We build a tree with the goal of predicting the Residuals.
- Predict the target label using all the trees within the ensemble.

= **Average price + learning rate*Residual predicted by decision tree**

- Compute the new residuals

Residual = Actual value - Predicted value

- Repeat steps 3 to 5 until the number of iterations matches the number specified by the hyper parameter (numbers of estimators).

- Once trained, use all of the trees in the ensemble to make a final prediction as to value of the target variable.

4. Naïve Baye's Classifier:

- **Baye's Theorem-**

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

Where,

- **P(A):** Class Prior Probability
- **P(B|A):** Likelihood
- **P(A|B):** Posterior Probability
- **P(A):** Predictor Prior Probability

Naïve Baye's Classifier: It is a set of algo. Based on baye's theorem. It is kind of algo. Which uses the bayes theorem.

Then what is Baye's theorem:

- In mathematics it is a probability theory.
- It is most important part of probability.
- This theory named after Thomas bayes.
- Its describe the probability of any events based on previous knowledge of that events.

Steps:

- As the first step toward prediction using naïve Bayes, you will have to estimate frequency of each and every attribute.

- Calculate possibilities of each attribute.

- Normalizing or return the standard condition of the values.

$$P(YES) = \frac{\text{Probability of YES}}{\text{Probability of YES + Probability of NO}}$$

$$P(NO) = \frac{\text{Probability of NO}}{\text{Probability of YES + Probability of NO}}$$

Why Naïve Baye’s?

- It is very fast and efficient to use on discrete as well as continuous data.
- It helps us to compute the conditional probability of event based on previous probabilities of two or more events.
- It requires the less amount of training data from which we trained our ML model.
- It is naïve Bayes it assumes that the occurrence of certain thing or features are independent to each other like identified a fruit by their particular shape, taste, colour and weight.

Apply Naïve Baye’s Classifier on text data in NLP (Natural Language Processing)

- NLP helps computer to communicate with human with their own language.
- In NLP we usually perform pre-processing steps:

a) STOP WORD
b) STEMMING
c) BAG OF WORDS
d) TF-IDF

After apply these steps we got vectors of specific sentences and we also have output features. Like:

P (YES|GIVEN SENTENCE)
GIVEN SENTENCE may be x1, x2, x3, x4, x5, xn

Vector: -

- It is used to represent numerical characteristics of data.
- It is the probability of X when Y is already happened.

Disadvantage:

- Its biggest disadvantage is that it implicitly assumes that all the attributes are unrelated to each other or this is not seen or happens in real life.

5. Decision Tree:

- It supports tools that uses tree like model.
- It is one way to display algorithm that only contains conditional.
- Control statement decision tree are commonly use in operation research.
- Specially in decision analysis to reach a goal but also a popular tool in machine learning.
- It is a decision support tool that uses a tree-like model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility.
- It is one way to display an algorithm that only contains conditional control statements.
- These are commonly used in operations research, specifically in decision analysis, to help to identify a strategy most likely to reach a goal, but are also a popular tool in machine learning.

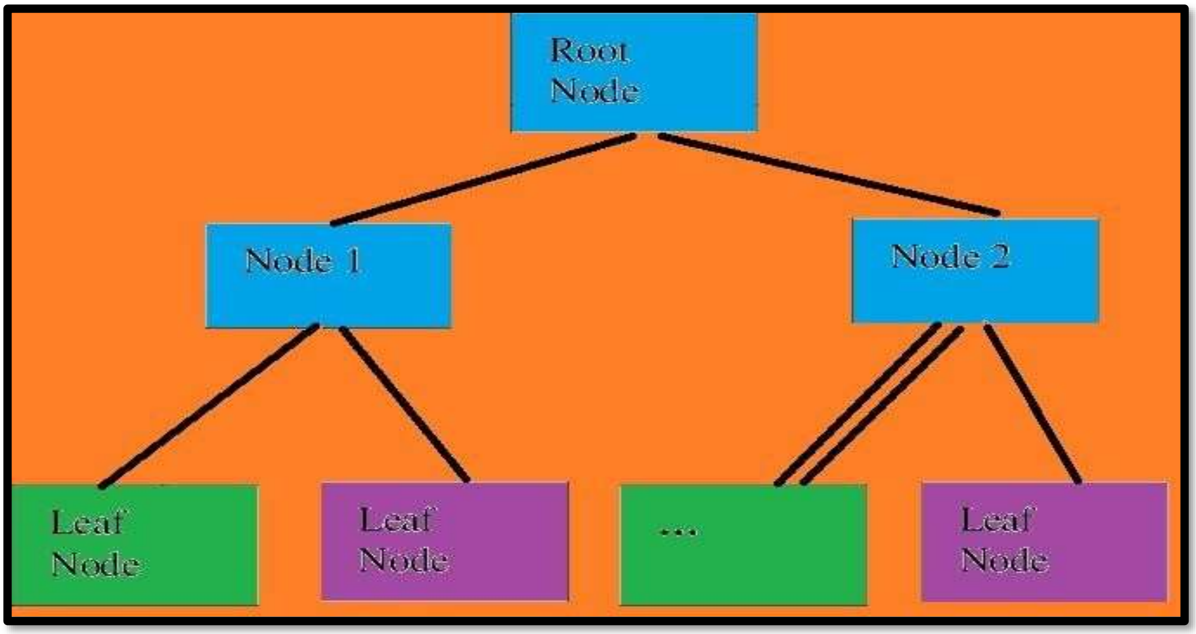


Figure 5.0 Diagram of Decision Tree

Advantage:

- Decision tree are simple to understand and people to understand decision tree model after a long explanation.
- Can be combined with other decision techniques.

- Use a white box model. if a given result is provided by model
- Help to determine worst, best, and expected values from different scenarios.

6. Random Forest:

- It is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems.
- One of the most important features of random forest model is that it can handle the data set containing continuous variables.
- It is an advance form of machine learning algorithm that is widely used in classification and regression problem.
- It performs better results for classification problem. It builds decision tree on different samples and take majority vote for classification and average.

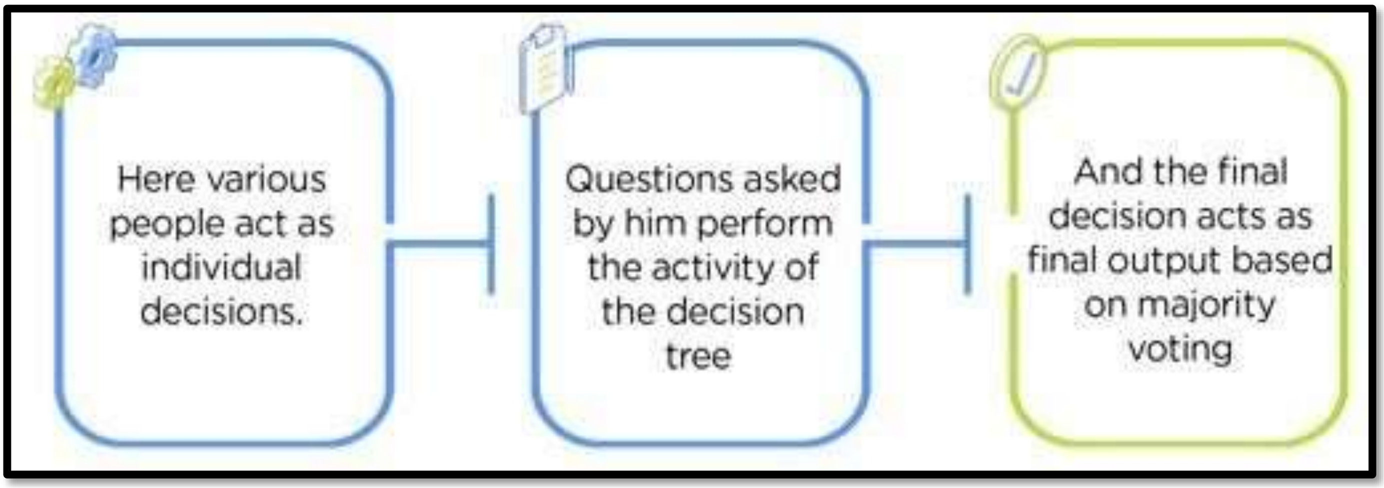

Here various people act as individual decisions.
Questions asked by him perform the activity of the decision tree
And the final decision acts as final output based on majority voting

Figure 6.0 Working Process

Working of Random Forest Algorithm: -

- **Bagging–** Bagging creates a different training subset from sample training data with replacement & the final output is based on majority voting. For example, Random Forest.
- **Boosting–** It combines weak learners into strong learners by creating sequential models such that the final model has highest accuracy. For example, ADA BOOST, XG BOOST,etc.

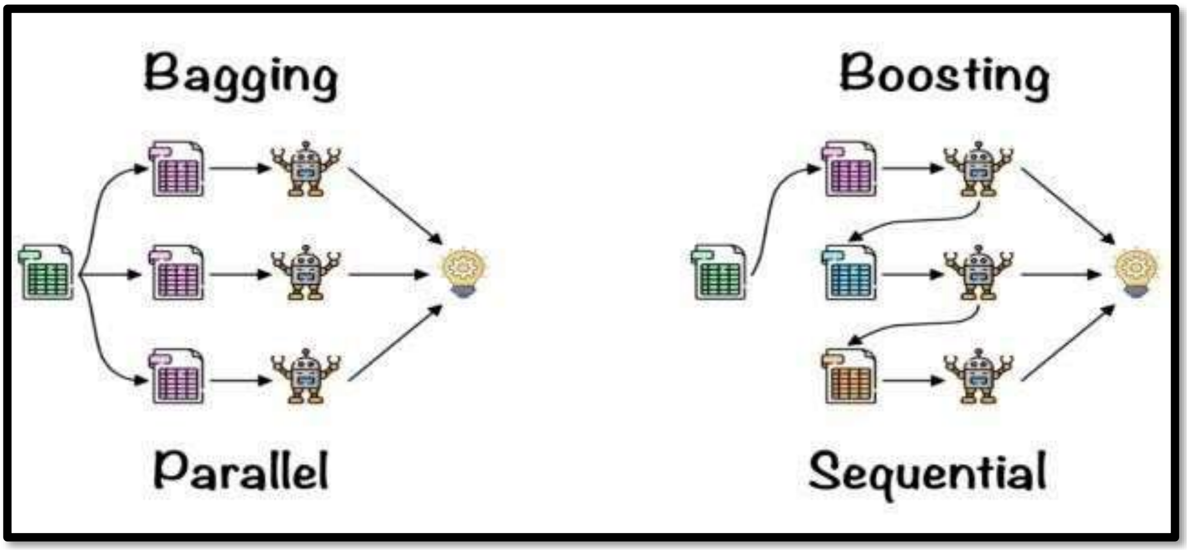


Figure 6.1 Boosting and Bagging

Important Features of Random Forest: -

- **Diversity-** Not all attributes/variables/features are considered while making an individual tree, each tree is different.
- **Immune to the curse of dimensionality-** Since each tree does not consider all the features, the feature space is reduced.
- **Parallelization-** Each tree is created independently out of different data and attributes. This means that we can make full use of the CPU to build random forests.
- **Train-Test split-** In a random forest we don't have to segregate the data for train and test as there will always be 30% of the data which is not seen by the decision tree.
- **Stability-** Stability arises because the result is based on majority voting/ averaging.

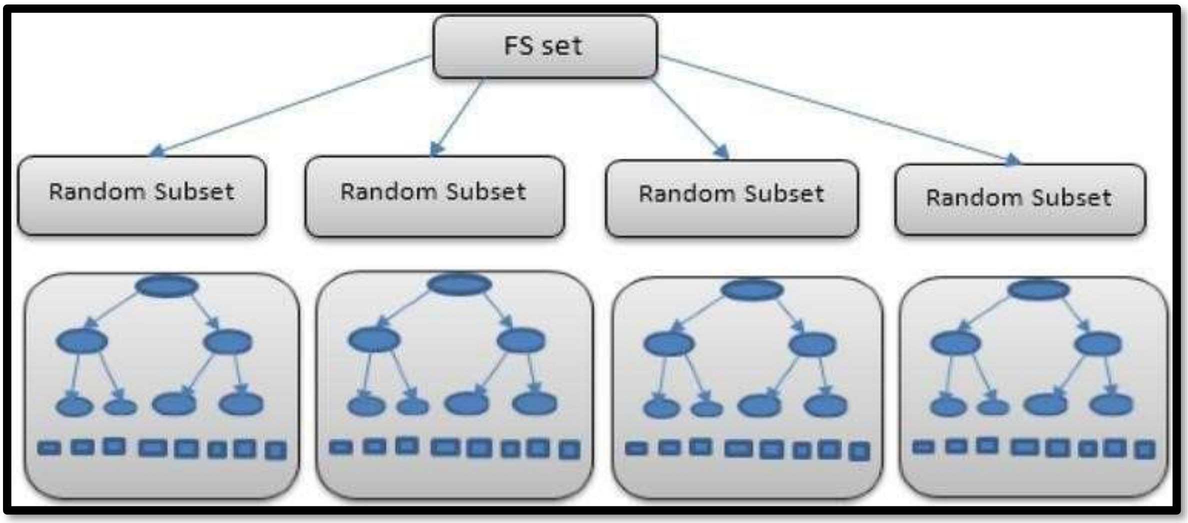


Figure 6.2 Random Forest Diagram III

PROBLEM STATEMENT

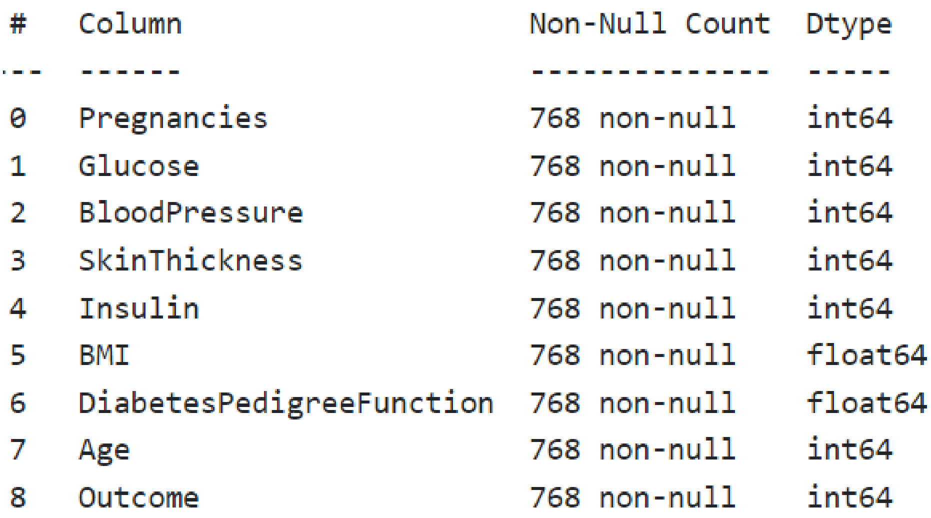
Diabetes is a most common disease caused by a group of metabolic disorders. It is also known as Diabetic mellitus. It affects the organs of the human body. It can be controlled by predicting this disease earlier. If diabetics patient is untreated for a long time, it may lead to increase blood sugar. Now a days, Healthcare industries generating large volume of data. Machine Learning algorithms and statistics are used to predict the disease with the help of current and past data. Machine learning techniques helps the doctors to predict early stage for diabetics. Diabetics patient medical record and different types of algorithms are added in dataset for experimental analysis. we use logistic regression, random forest, decision tree classifier and gradient boosting to predict whether a patient has diabetes based on diagnostic measurements. Performance and accuracy of the applied algorithms is discussed and compared. Doctors rely on common knowledge for treatment. When common knowledge is lacking, studies are summarized after some number of cases have been studied. But this process takes time, whereas if machine learning is used, the patterns can be identified earlier. For using machine learning, a huge amount of data is required. There is very limited amount of data available depending on the disease. Also, the number of samples having no diseases is very high compared to number of samples actually having the disease.

REQUIREMENTS

- **MS-Office**
- **Jupyter Notebook**
- **Python**
- **Numpy**
- **Pandas**
- **Sklearn**
- **Pyttsx3**
- **PySimpleGUI**
- **Dataset**
- **Machine learning Algos.**

DATA ANALYSIS

Diabetes dataset has 9 attributes in total. All the person in records are females and the number of pregnancies they have had has been recorded as the first attribute of the dataset. Second is the value of Plasma glucose concentration a 2 hours in an oral glucose tolerance test and then is the Diastolic blood pressure (mm Hg), fourth in line is the Triceps skin fold thickness (mm), then is the 2-Hour serum insulin (mu U/ml), sixth is Body mass index (weight in kg/ (height in m) ^2) and then seventh is the Diabetes pedigree function and the second last value is the that of the Age (years). The ninth column is that of the Class variable (0 or 1), 0 for no diabetes and 1 for the presence.



```
 #   Column                    Non-Null Count  Dtype  
---  ------                    --------------  -----  
 0   Pregnancies               768 non-null    int64  
 1   Glucose                   768 non-null    int64  
 2   BloodPressure             768 non-null    int64  
 3   SkinThickness             768 non-null    int64  
 4   Insulin                   768 non-null    int64  
 5   BMI                       768 non-null    float64
 6   DiabetesPedigreeFunction  768 non-null    float64
 7   Age                       768 non-null    int64  
 8   Outcome                   768 non-null    int64  
```

Table 1.0 Table of Data Analysis

METHEDOLOGY

- Problem Solving Methods are concerned with efficient realization of functionality. This is an important characteristics of Problem Solving Methods and should be deal with it explicitly.
- Problem Solving Methods achieve this efficiency by making assumptions about resources provided by their context (such as domain knowledge) and by assumptions about the precise definition of the task. It is important to make these assumptions explicit as it give the reason about Problem Solving Methods.
- The process of constructing Problem Solving Methods is assumption-based. During this process assumptions are added that facilitate efficient open rationalization of the desired functionality

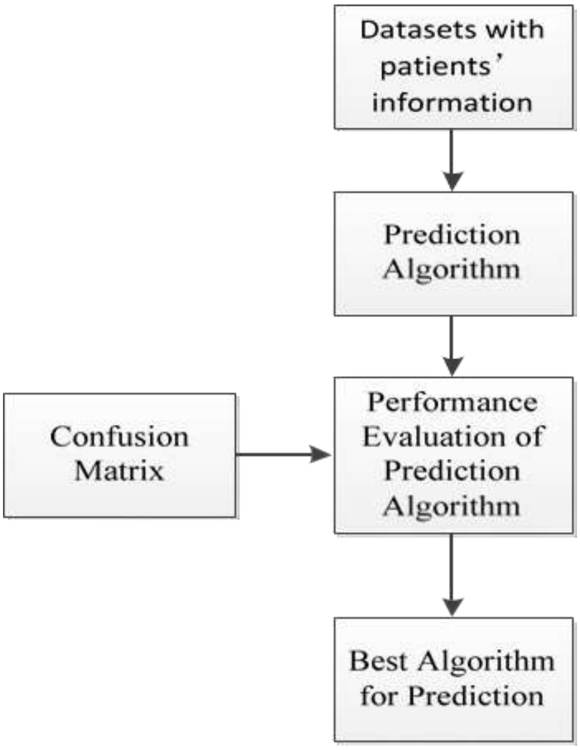

Datasets with patients' information
Prediction Algorithm
Confusion Matrix
Performance Evaluation of Prediction Algorithm
Best Algorithm for Prediction

Figure 7.0 Process diagram

MERITS OF PROPOSED SYSTEM

- Public health is a fundamental concern for protecting and preventing the community from health hazard diseases Governments are spending a considerable amount of their gross domestic product (GDP) for the welfare of the public, and initiatives such as vaccination have prolonged the life expectancy of people . However, for the last many years, there has been a considerable emergence of chronic and genetic diseases affecting public health. Diabetes mellitus is one of the extremely life-threatening diseases because it contributes to other lethal diseases, i.e., heart, kidney, and nerve damage .
- Diabetes is a metabolic disorder that impairs an individual's body to process blood glucose, known as blood sugar. This disease is characterized by hyperglycemia resulting from defects in insulin secretion, insulin action, or both . An absolute deficiency of insulin secretion causes type 1 diabetes (T1D). Diabetes drastically spreads due to the patient's inability to use the produced insulin. It is called type 2 diabetes (T2D) . Both types are increasing rapidly, but the ratio of increase in T2D is higher than T1D. 90 to 95% of cases of diabetes are of T2D.

ARCHITECTURAL DIAGRAM FOR PROPOSED METHOD

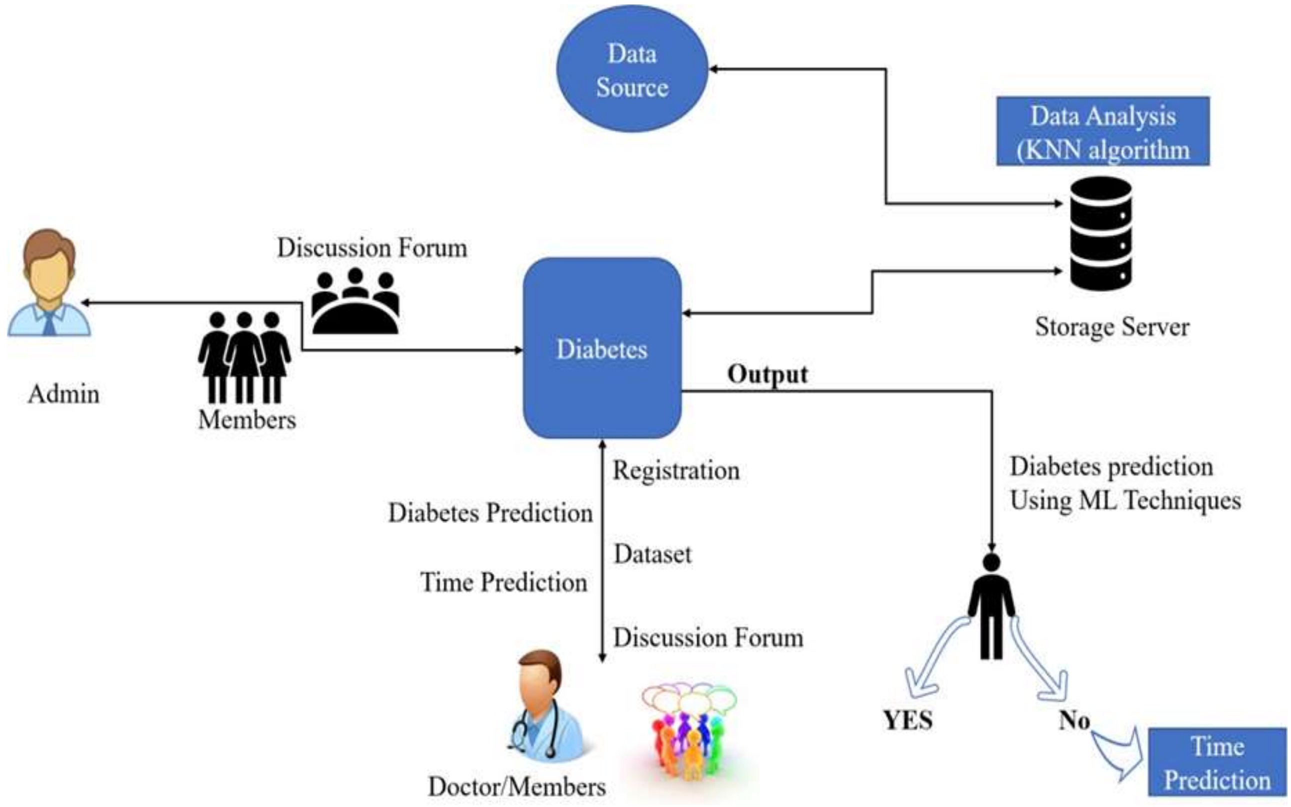
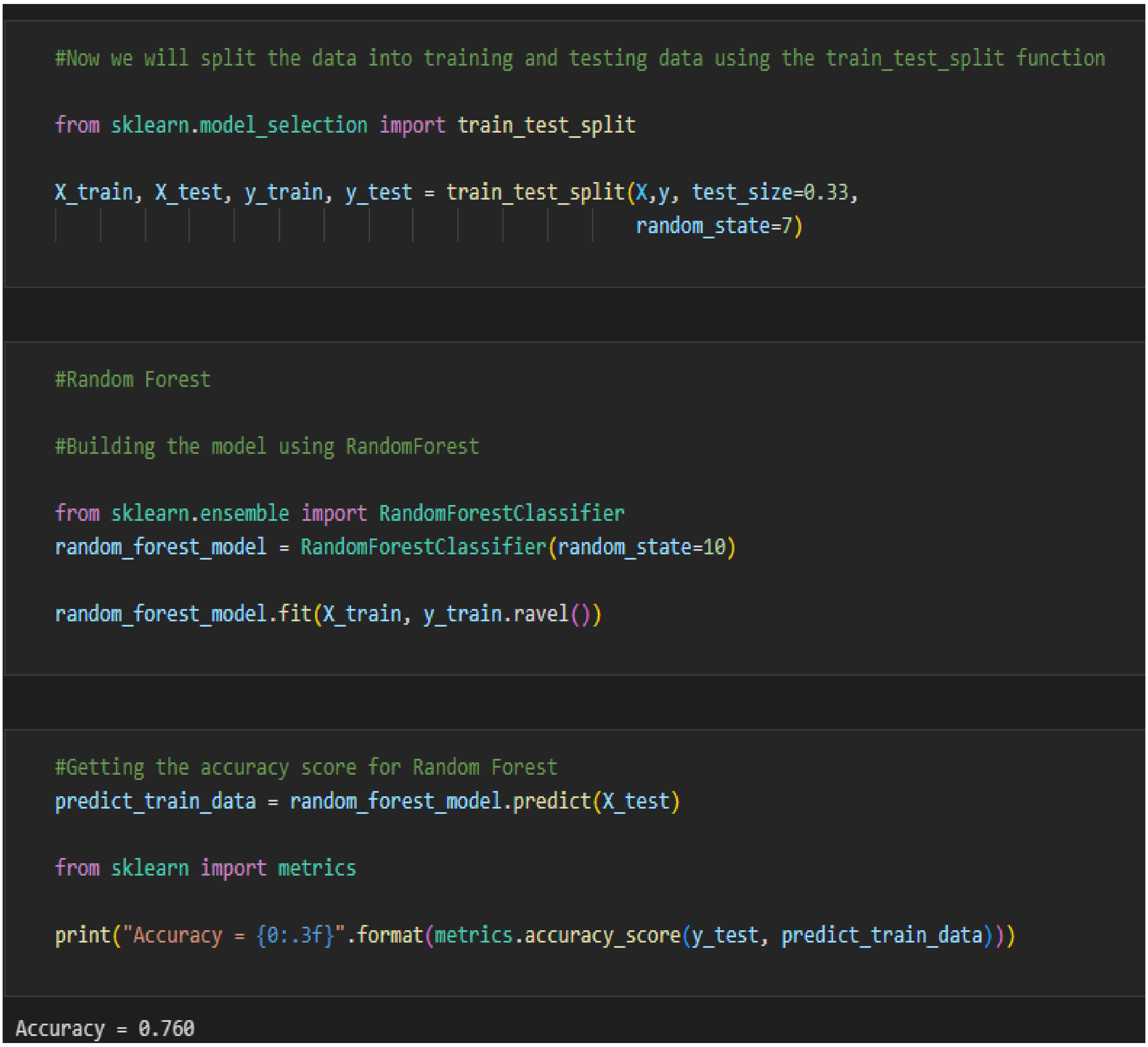


Figure 8.0 Architectural Diagram

User Interface and Implementation

Random Forest Classification:-



```
#Now we will split the data into training and testing data using the train_test_split function

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X,y, test_size=0.33,
                                                     random_state=7)
```

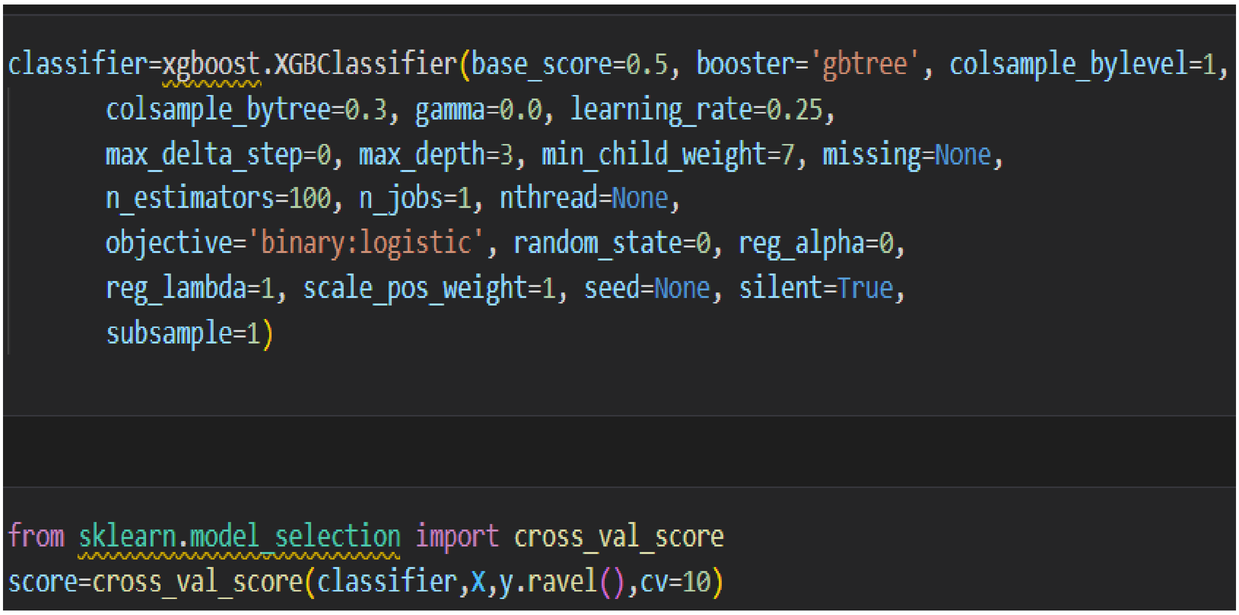
```
#Random Forest

#Building the model using RandomForest

from sklearn.ensemble import RandomForestClassifier
random_forest_model = RandomForestClassifier(random_state=10)

random_forest_model.fit(X_train, y_train.ravel())
```

```
#Getting the accuracy score for Random Forest
predict_train_data = random_forest_model.predict(X_test)

from sklearn import metrics

print("Accuracy = {0:.3f}".format(metrics.accuracy_score(y_test, predict_train_data)))
```

```
Accuracy = 0.760
```

User Interface and Implementation

XGB Boost Classifier:-

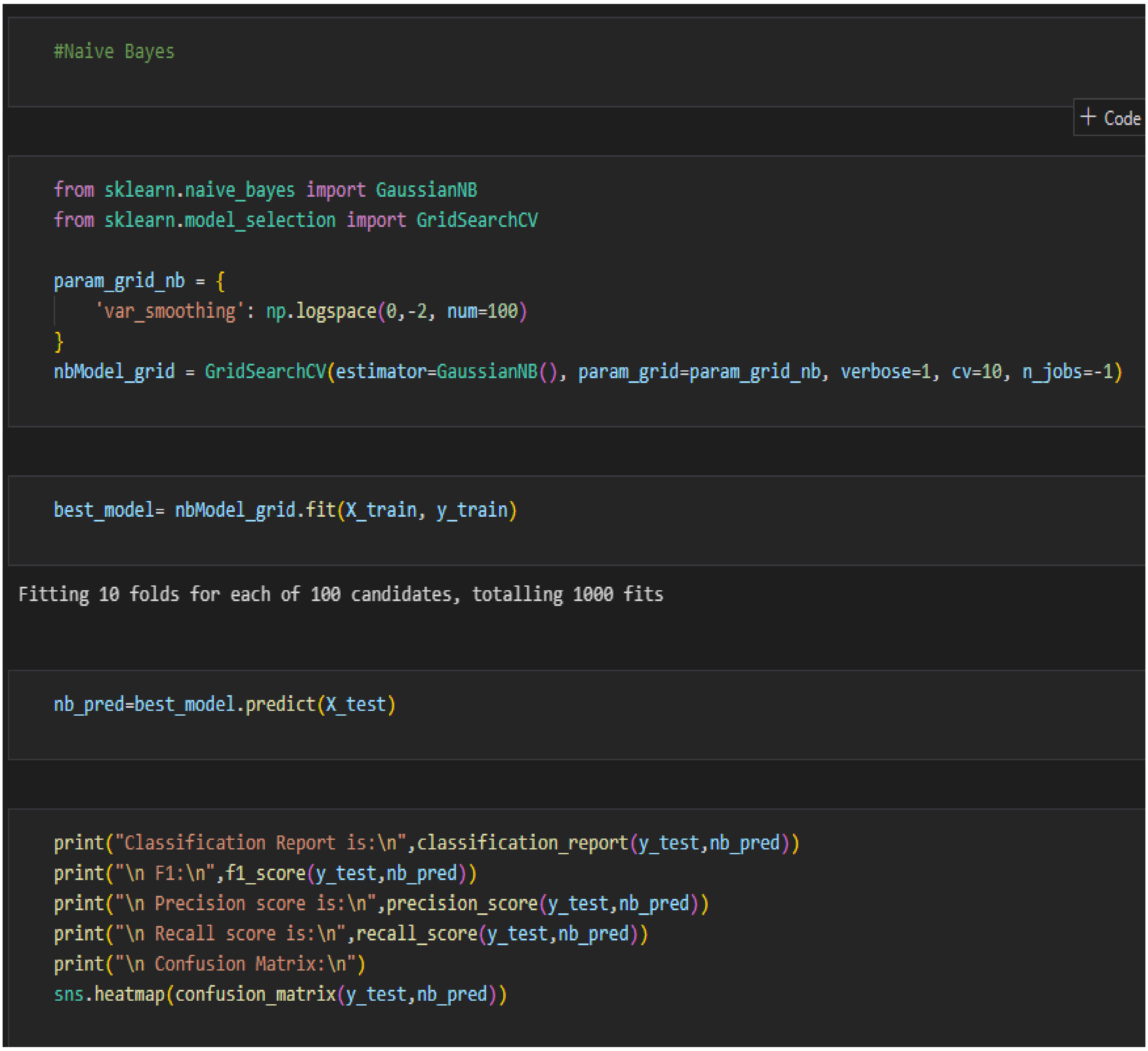


```
classifier=xgboost.XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
       colsample_bytree=0.3, gamma=0.0, learning_rate=0.25,
       max_delta_step=0, max_depth=3, min_child_weight=7, missing=None,
       n_estimators=100, n_jobs=1, nthread=None,
       objective='binary:logistic', random_state=0, reg_alpha=0,
       reg_lambda=1, scale_pos_weight=1, seed=None, silent=True,
       subsample=1)
```

```
from sklearn.model_selection import cross_val_score
score=cross_val_score(classifier,X,y.ravel(),cv=10)
```

User Interface and Implementation

Naïve Bayes:-



```
#Naive Bayes
```

```
from sklearn.naive_bayes import GaussianNB
from sklearn.model_selection import GridSearchCV

param_grid_nb = {
    'var_smoothing': np.logspace(0,-2, num=100)
}
nbModel_grid = GridSearchCV(estimator=GaussianNB(), param_grid=param_grid_nb, verbose=1, cv=10, n_jobs=-1)
```

```
best_model= nbModel_grid.fit(X_train, y_train)
```

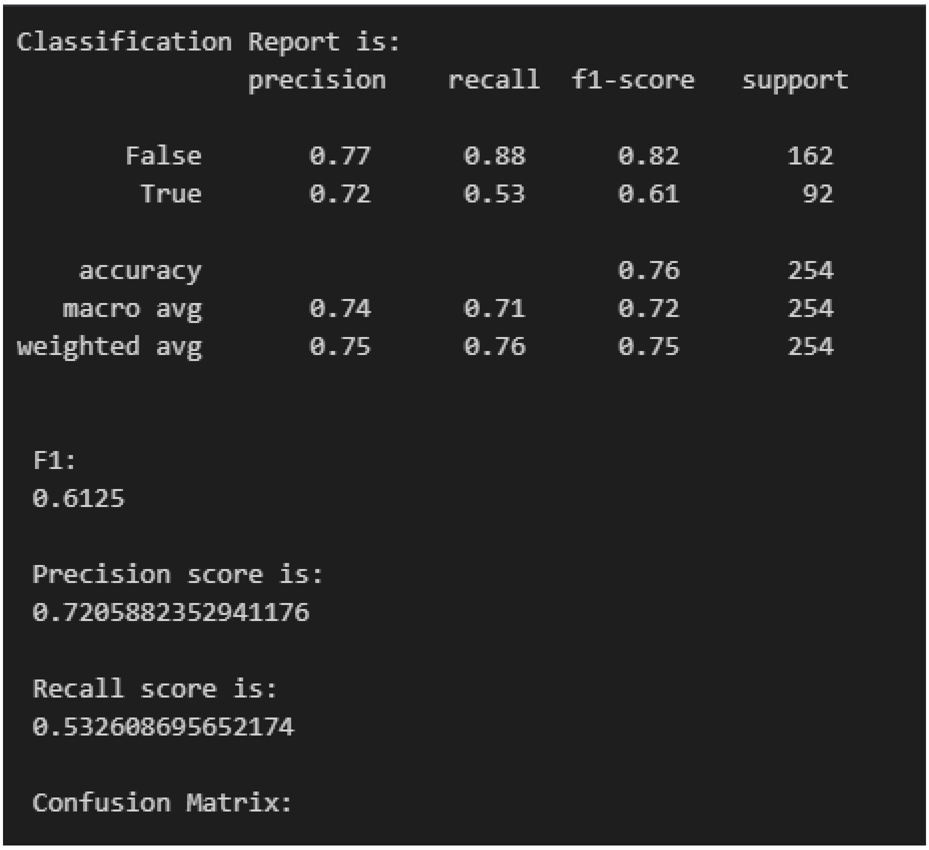
```
Fitting 10 folds for each of 100 candidates, totalling 1000 fits
```

```
nb_pred=best_model.predict(X_test)
```

```
print("Classification Report is:\n",classification_report(y_test,nb_pred))
print("\n F1:\n",f1_score(y_test,nb_pred))
print("\n Precision score is:\n",precision_score(y_test,nb_pred))
print("\n Recall score is:\n",recall_score(y_test,nb_pred))
print("\n Confusion Matrix:\n")
sns.heatmap(confusion_matrix(y_test,nb_pred))
```

User Interface and Implementation

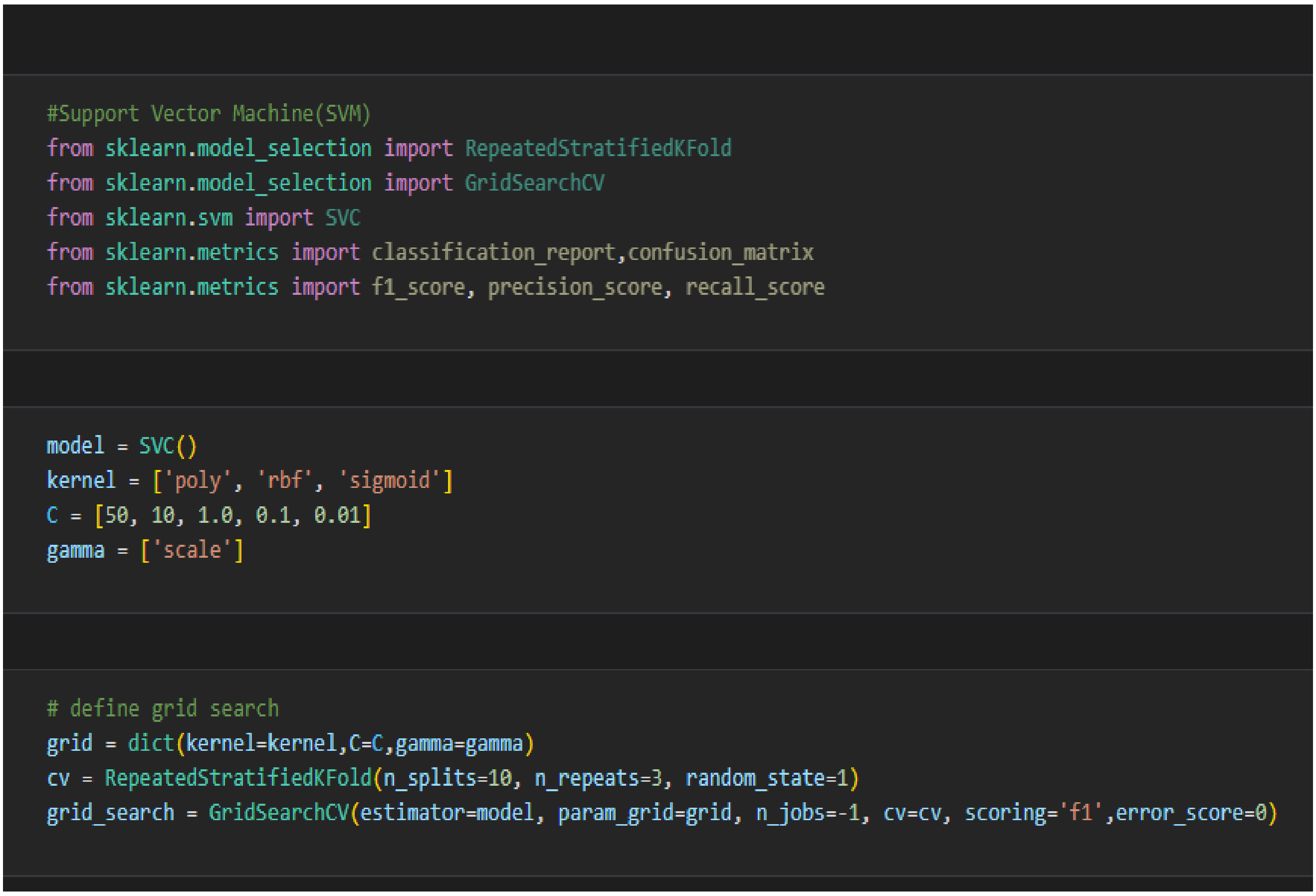
Output:-



```
Classification Report is:
               precision    recall  f1-score   support

       False       0.77      0.88      0.82       162
        True       0.72      0.53      0.61        92

    accuracy                           0.76       254
   macro avg       0.74      0.71      0.72       254
weighted avg       0.75      0.76      0.75       254


 F1:
 0.6125

 Precision score is:
 0.7205882352941176

 Recall score is:
 0.532608695652174

 Confusion Matrix:
```

User Interface and Implementation

Support Vector Machine:-



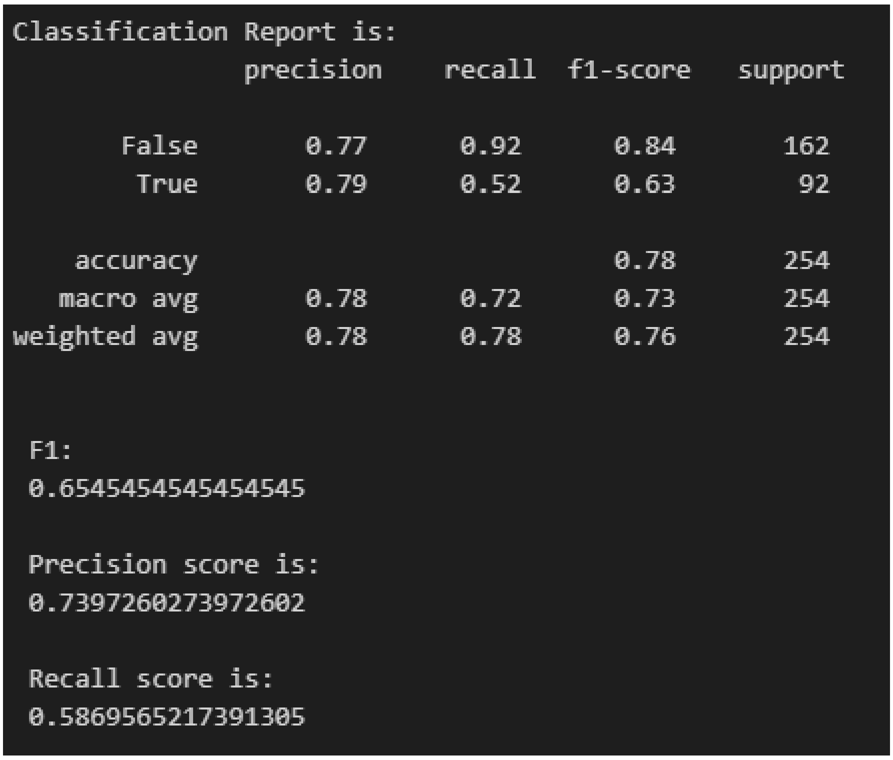
```
#Support Vector Machine(SVM)
from sklearn.model_selection import RepeatedStratifiedKFold
from sklearn.model_selection import GridSearchCV
from sklearn.svm import SVC
from sklearn.metrics import classification_report,confusion_matrix
from sklearn.metrics import f1_score, precision_score, recall_score
```

```
model = SVC()
kernel = ['poly', 'rbf', 'sigmoid']
C = [50, 10, 1.0, 0.1, 0.01]
gamma = ['scale']
```

```
# define grid search
grid = dict(kernel=kernel,C=C,gamma=gamma)
cv = RepeatedStratifiedKFold(n_splits=10, n_repeats=3, random_state=1)
grid_search = GridSearchCV(estimator=model, param_grid=grid, n_jobs=-1, cv=cv, scoring='f1',error_score=0)
```

User Interface and Implementation

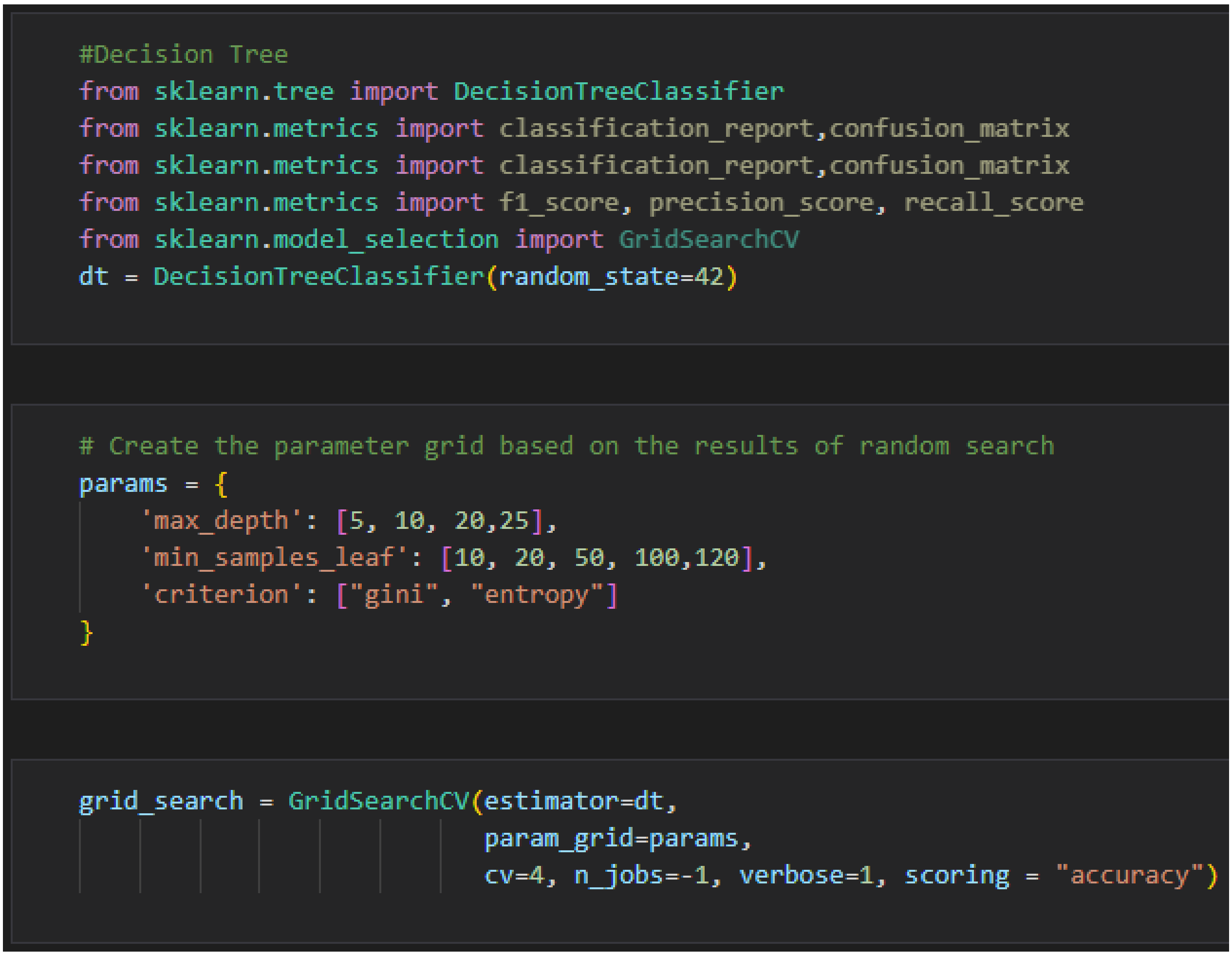
Output:-



```
Classification Report is:
              precision    recall  f1-score   support

       False       0.77      0.92      0.84       162
        True       0.79      0.52      0.63        92

    accuracy                           0.78       254
   macro avg       0.78      0.72      0.73       254
weighted avg       0.78      0.78      0.76       254


 F1:
 0.6545454545454545

 Precision score is:
 0.7397260273972602

 Recall score is:
 0.5869565217391305
```

User Interface and Implementation

Decision Tree:-

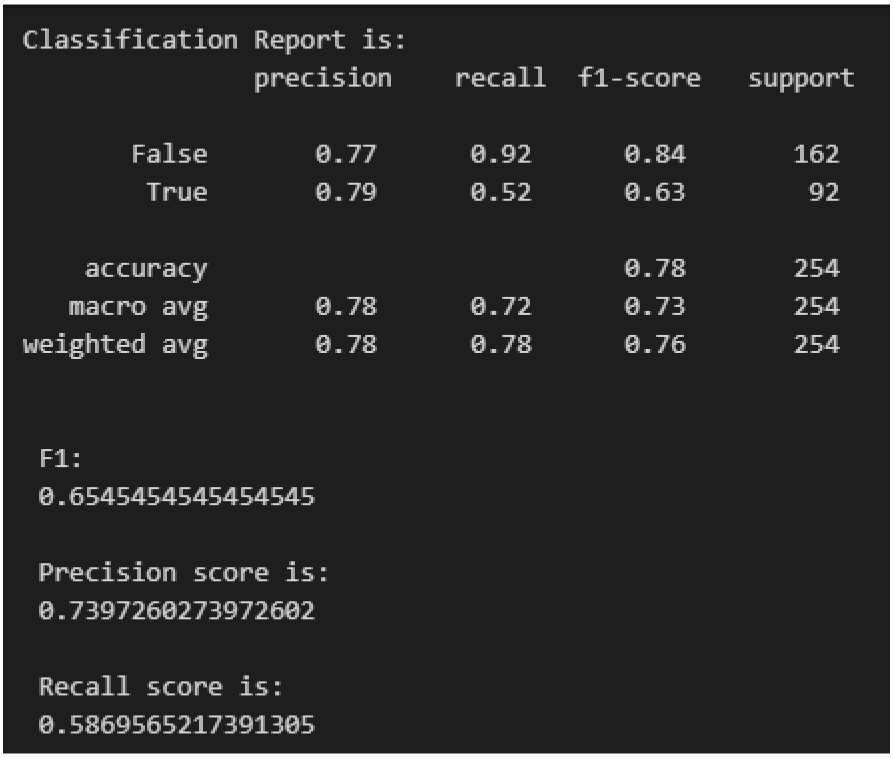


```
#Decision Tree
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import classification_report,confusion_matrix
from sklearn.metrics import classification_report,confusion_matrix
from sklearn.metrics import f1_score, precision_score, recall_score
from sklearn.model_selection import GridSearchCV
dt = DecisionTreeClassifier(random_state=42)
```

```
# Create the parameter grid based on the results of random search
params = {
    'max_depth': [5, 10, 20,25],
    'min_samples_leaf': [10, 20, 50, 100,120],
    'criterion': ["gini", "entropy"]
}
```

```
grid_search = GridSearchCV(estimator=dt,
                           param_grid=params,
                           cv=4, n_jobs=-1, verbose=1, scoring = "accuracy")
```

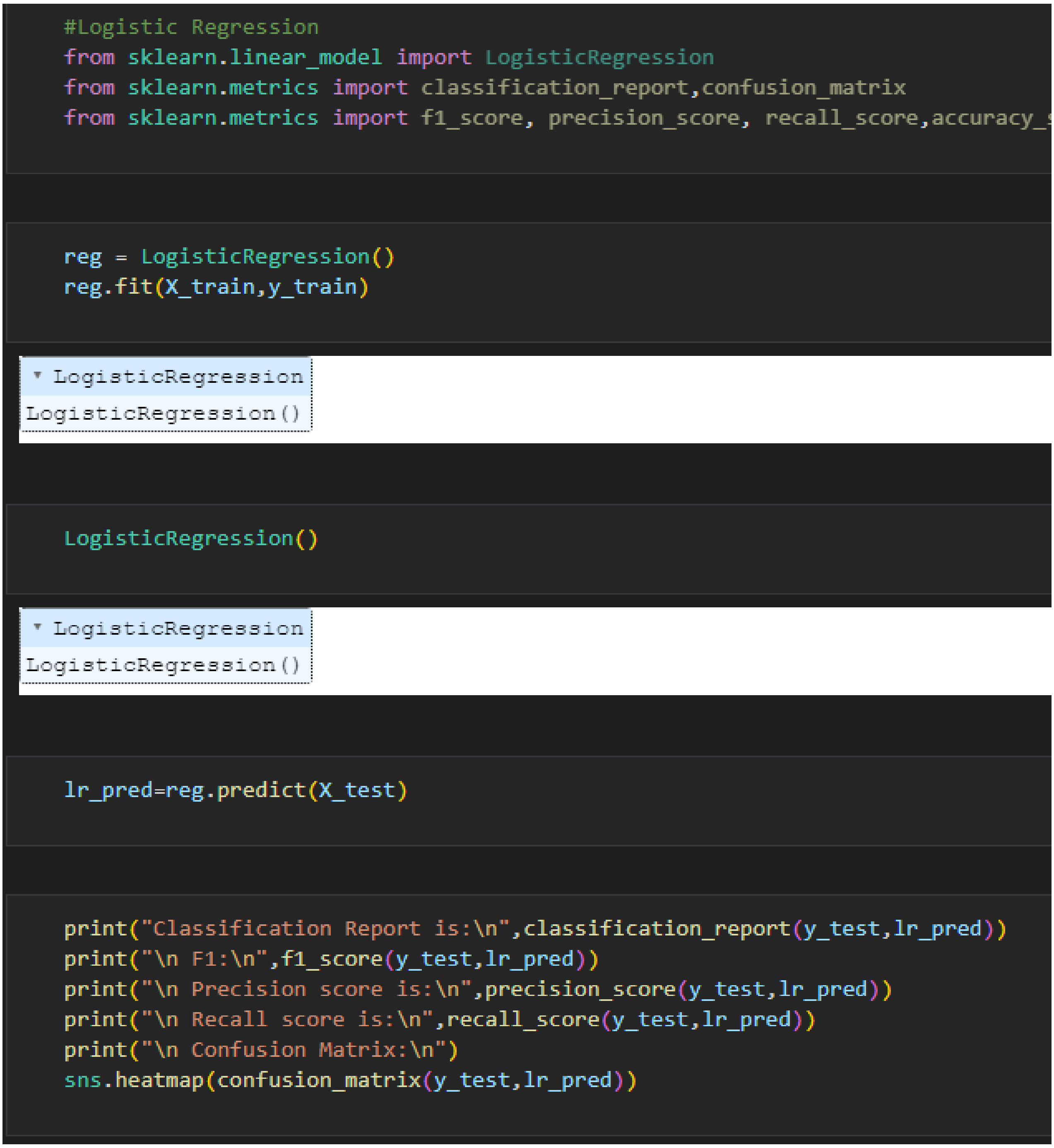
Output:-



```
Classification Report is:
              precision    recall  f1-score   support

       False       0.77      0.92      0.84       162
        True       0.79      0.52      0.63        92

    accuracy                           0.78       254
   macro avg       0.78      0.72      0.73       254
weighted avg       0.78      0.78      0.76       254


 F1:
 0.6545454545454545

 Precision score is:
 0.7397260273972602

 Recall score is:
 0.5869565217391305
```

User Interface And Implementation

Logistic Regression:-



```
#Logistic Regression
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report,confusion_matrix
from sklearn.metrics import f1_score, precision_score, recall_score,accuracy_
```

```
reg = LogisticRegression()
reg.fit(X_train,y_train)
```

```
▾ LogisticRegression
LogisticRegression()
```

```
LogisticRegression()
```

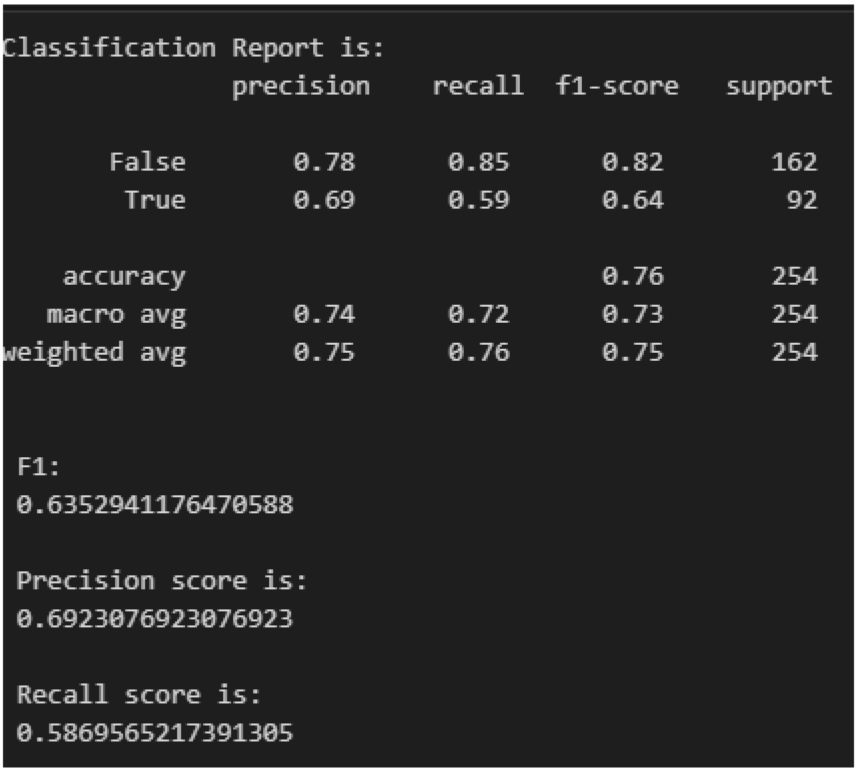
```
▾ LogisticRegression
LogisticRegression()
```

```
lr_pred=reg.predict(X_test)
```

```
print("Classification Report is:\n",classification_report(y_test,lr_pred))
print("\n F1:\n",f1_score(y_test,lr_pred))
print("\n Precision score is:\n",precision_score(y_test,lr_pred))
print("\n Recall score is:\n",recall_score(y_test,lr_pred))
print("\n Confusion Matrix:\n")
sns.heatmap(confusion_matrix(y_test,lr_pred))
```

User Interface And Implementation

Output:-



```
Classification Report is:
               precision    recall  f1-score   support

       False       0.78      0.85      0.82       162
        True       0.69      0.59      0.64        92

    accuracy                           0.76       254
   macro avg       0.74      0.72      0.73       254
weighted avg       0.75      0.76      0.75       254


 F1:
 0.6352941176470588

 Precision score is:
 0.6923076923076923

 Recall score is:
 0.5869565217391305
```

RESULT

Model	Accuracy
Random forest classifier	0.760
XGB Classifier	0.739
Naïve Bayes	0.76
Support vector machine	0.78
Decision tree	0.78
Logistic Regression	0.76
Define grid search	0.739

Table 2.0 Result Table

CONCLUSION

Although there is no clear research showing that there is an exact relationship between diabetes and age ,there is a clear trend of younger diabetes now. Early detection of diabetes plays a vit a role in treatment ,and the emergence of machine learning has revolutionized the study of diabetes risk prediction. With the continuous advancement of data mining methods, we have studied various methods of diagnosing diabetes. We found that SVM has the highest accuracy through the confusion matrix evaluation test. However, this kind of research needs to be updated regularly with more instance data sets. Finally, we. can see that data mining algorithms through research, machine learning techniques and various other technologies have made outstanding contributions in the medical field and disease diagnosis. It is hoped that it can help clinicians make better judgments on disease status.

REFERENCES

R. Kavitha, M. Kavitha and R. Srinivasan, "Crop Recommendation in Precision Agriculture using Supervised Learning Algorithms," 2022 3rd International Conference for Emerging Technology (INCET), Belgaum, India, 2022, pp. 1-4.
doi: 10.1109/INCET54531.2022.9824155
keywords: {Productivity;Support vector machines;Supervised learning;Soil;Data collection;Agriculture;Seeds (agriculture);agriculture;crop recommendation;voting;K-Nearest Neighbor},
URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9824155&isnumber=9823966

Z. Doshi, S. Nadkarni, R. Agrawal and N. Shah, "AgroConsultant: Intelligent Crop Recommendation System Using Machine Learning Algorithms," 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 2018, pp. 1-6.
doi: 10.1109/ICCUBEA.2018.8697349
keywords: {Agriculture;Soil;Machine learning algorithms;Meteorology;Prediction algorithms;Training;Classification algorithms;crop prediction;machine learning;crop recommendation system;smart farming;multi-label classification},
URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8697349&isnumber=8697217

D. Modi, A. V. Sutagundar, V. Yalavigi and A. Aravatagimath, "Crop Recommendation Using Machine Learning Algorithm," 2021 5th International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, 2021, pp. 1-5.
doi: 10.1109/ISCON52037.2021.9702392
keywords: {Productivity;Support vector machines;Machine learning algorithms;Crops;Soil;Prediction algorithms;Agriculture;Agriculture;Soil parameters;SVM algorithm;classification;Confusion Matrix},
URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9702392&isnumber=9702296

G. Chauhan and A. Chaudhary, "Crop Recommendation System using Machine Learning Algorithms," 2021 10th International Conference on System Modeling & Advancement in Research Trends (SMART), MORADABAD, India, 2021, pp. 109-112.
doi: 10.1109/SMART52563.2021.9676210
keywords: {Machine learning algorithms;Costs;Crops;Humidity;Soil;Market research;Agriculture;Machine Learning Techniques;Recommendation System},
URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9676210&isnumber=9675300

D. H. Patel, H. B. Mirani, A. R. Vasant and N. M. Chaudhari, "Advancement in Agronomy Using Machine Learning Approach," 2022 IEEE 7th International conference for Convergence in Technology (I2CT), Mumbai, India, 2022, pp. 1-5.
doi: 10.1109/I2CT54291.2022.9824171
keywords: {Productivity;Temperature sensors;Irrigation;Databases;Soil moisture;Crops;Humidity;Agronomy;Machine Learning;Smart Irrigation;Crop recommendation},
URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9824171&isnumber=9823960

D. N. V. S. L. S. Indira, M. Sobhana, A. H. L. Swaroop and V. Phani Kumar, "KRISHI RAKSHAN - A Machine Learning based New Recommendation System to the Farmer," 2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2022, pp. 1798-1804.
doi: 10.1109/ICICCS53718.2022.9788221keywords: {Radio frequency;Productivity;Biological system modeling;Crops;Soil;Predictive models;Prediction algorithms;Machine Learning;Crop Productivity;Crop prediction;Fertilizer Recommendation and Suggestion;Disease Detection},
URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9788221&isnumber=9787985

Meeting Photos

GPS Map Camera
Google
null
22/03/23 04:21 PM GMT +05:30