 **PREDICTION OF DIABETES USING MACHINE LEARNING**

**A PROJECT REPORT**

***Submitted by***

**BALA SUBRA MANIYAN S R - 71051907006**

**SWEDHA B - 71051907026**

**THARUNKUMAR S - 71051907027**

**VIJAY S - 71051907029**

***A Project report submitted in partial fulfilment of the requirements for the award of the degree of***

**BACHELOR OF TECHNOLOGY**

**IN**

**INFORMATION TECHNOLOGY**

**COIMBATORE INSTITUTE OF ENGINEERING AND TECHNOLOGY**

**AUTONOMOUS : COIMBATORE 641 109**

**December 2022**

**AUTONOMOUS: COIMBATORE 641 109**

**BONAFIDE CERTIFICATE**

Certified that this project report **“PREDICTION OF DIABETES USING MACHINE LEARNING”** is the bonafide work of **“BALA SUBRA MANIYAN S R, SWEDHA B, VIJAY S, THARUN KUMAR S”** who carried out the project work under my supervision.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Dr.K.PUSHPALATHA M.E., Ph.D., Dr.K.PUSHPALATHA M.E., Ph.D.,**

**HEAD OF THE DEPARTMENT SUPERVISOR**

Department of Information Technology Department of Information Technology

Coimbatore Institute of Engineering and Coimbatore Institute of Engineering and

Technology, Technology,

Coimbatore – 641 109 Coimbatore – 641 109

Submitted for the university examination held on **01.12.2022** at Coimbatore Institute of Engineering and Technology, Coimbatore – 641109

**INTERNAL EXAMINER** **EXTERNAL EXAMINER**

**ACKNOWLEDGEMENT**

We, have great and immense pleasure in expressing the acknowledgment to the numerous contributors for the success of this project work. They all deserve credit and thanks to carry out this project successfully.

We would like to record our sincere indebtedness and gratitude to our beloved Director, **Dr.K.A.Chinnaraju M.Sc.,M.B.A.,Ph.D.,** for his noteworthy efforts to enhance our professional dexterity and co-curricular excellence.

We are grateful to our respected Principal**, Dr.N.Nagarajan, M.E., Ph.D**.,for providing us with necessary facilities to carry out our project work.

We express our sincere thanks to our Head of the Department and our Project guide coordinator **DR. K. PUSHPALATHA M.E., Ph.D.,** for her timely help and cooperation.

Finally, We extend our thanks to the management, faculty members, parents and our student friends for their support and encouragement and to all others, who extended their helping hands to us in the completion of our Project phase I.

**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 Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Support Vector Machine Comparison(SVM), XG Boost, K-Nearest Neighbour(KNN), Gaussian Naïve Bayes, Gradient Boosting Classifier and AdaBoost Classifier 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 :** Machine Learning, Diabetes, Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Support Vector Machine Comparison(SVM), XG Boost, K-Nearest Neighbour(KNN), Gaussian Naïve Bayes, Gradient Boosting Classifier and AdaBoost Classifier, Accuracy.

**TABLE OF CONTENTS**

**List of figures**

**List of tables**

**CHAPTERS**

**1.Introduction**

1.1 Overview

* + 1. Data Science
    2. Applications of Data Science

## 1.1.3 Statistics Concepts Needed for Data Science

## 2. Machine Learning and Data Modelling

2.1 Basic libraries used in Data Science

2.2 Diabetes Prediction Using Data Science

2.3 Classification Algorithms

2.4 Learners In Classification Algorithm

2.5 Types of ML Classification Algorithms

2.6 Evaluating Classification Model

2.7 Use Cases of Classification Algorithms

**3.Literature Survey**

3.1 Introduction

3.2 Related Works

3.3 Methodology

3.3.1 Data Set Description

3.3.2 Data Pre-Processing 3.3.3 Training and Testing Phase

**4.Proposed Model**

4.1 Dataset

4.2 Data Pre-processing

4.2.1 Categorical Variable Conversion

4.2.2 Feature Scaling

4.2.3 Principal Component Analysis (PCA)

4.3.5 Train-Test Split

4.3 Algorithms

###### 4.3.1 Random Forest

4.3.2 Support Vector Machine(SVM)

###### 4.3.3 Logistic Regression

4.3.4 XG Boost

###### 4.3.5 Decision Tree

4.3.6 K-Nearest Neighbour(KNN)

4.3.7 Gaussian Naïve Bayes

4.3.8 GradientBoostingClassifier

4.3.9 AdaBoost Classifier

**5. User Interface**

5.1 Web Application

5.1.1 Flask

5.1.2 JINJA Templates

5.2 Web Interface

5.2.1 HTML/CSS

5.2.2 Bootstrap

5.2.3 JavaScript

5.3 Database

5.3.1 MYSQL

**6.Experimental Analysis and Result**

6.1 System Configuration

6.1.1 Hardware Requirements

6.1.2 Software Requirements

6.2 Sample Code and Results

6.2.1 ML Modules

6.3 Web Interface Code and Result

6.3.1 Web Interface Results

6.4 Performance Matrix

**7.Conclusion**

**8.References**

**List of Figures:**

|  |  |
| --- | --- |
| **Fig** | **Name of the figure** |
| **3.3.1.1** | Predictions |
| **4.1** | PIMA Data Distribution |
| **4.3.1** | Random Forest First Stage |
| **4.3.2** | Support Vector Machine(SVM) |
| **4.3.5** | Decision Tree Principle |
| **5.2.2** | Bootstrap |

**List of Tables :**

|  |  |
| --- | --- |
| **Table No** | **Name of the table** |
| 1.1 | Confusion Matrix |
| 3.3.1 | Dataset Description |

**CHAPTER 1**

**INTRODUCTION**

* 1. **Overview**
     1. **Data Science**

Data science is the study of data. Like biological sciences is a study of biology, physical sciences, it’s the study of physical reactions. Data is real, data has real properties, and we need to study them if we’re going to work on them. Data Science involves data and some signs.

It is the process of using data to understand too many different things, to understand the world. Suppose when you have a model or proposed explanation of a problem, and you try to validate that proposed explanation or model with your data.

It is the skill of unfolding the insights and trends that are hiding (or abstract) behind data. It’s when you translate data into a story. So use storytelling to generate insight. And with these insights, you can make strategic choices for a company or an institution.

We can also define data science as a field that is about processes and systems to extract data of various forms and from various resources whether the data is unstructured or structured. The definition and the name came up in the 1980s and 1990s when some professors, IT Professionals, scientists were looking into the statistics curriculum, and they thought it would be better to call it data science and then later on data analytics derived.

Data Science is relevant today because we have millions of data available on single data or for single data. We didn’t use to worry about the lack of data. Now we have tons of data. In the past, we didn’t have defined algorithms, now we have algorithms. In the past, the software was not affordable by everyone because it was too expensive, so only industries with big-bucks can use it but now it is open source and freely available. In the past, we didn’t even think about storing a large amount of data, because the storage facilities are also very costly and now it is available for a fraction of a cost, we can have gazillions of data sets for a very low cost. Also, internet connectivity was not common and too costly. So, the tools to work with data, the variability of data, the ability to store, analyse data and last and most important Connectivity, it’s all cheap, it’s all available, it’s all ubiquitous, it’s here. There’s never been a better time to be a data scientist than now.

* + 1. **Applications Of Data Science**

Following are some of the applications that makes use of Data Science for it’s services :

* Internet Search Results (Google)
* Recommendation Engine (Spotify)
* Intelligent Digital Assistants (Google Assistant)
* Autonomous Driving Vehicle (Waymo)
* Spam Filter (Gmail)
* Abusive Content and Hate Speech Filter (Facebook)
* Robotics (Boston Dynamics)
* Automatic Piracy Detection (YouTube)

## 1.1.3 Statistics Concepts Needed for Data Science

Statistics make a central part of data science. Statistics is a broad field that offers many applications. Data scientists must know the statistics very well. This can be inferred from the fact that statistics help to interpret and organize data. The descriptive statistics and knowledge of probability are must-know data science concepts.

Below are the basic Statistics concepts that a Data Scientist should know:

### **1. Descriptive Statistics**

Descriptive statistics help to analyse the raw data to find the primary and necessary features from it. Descriptive statistics offers a way to visualize the data to present it in a readable and meaningful way. It is different from inferential statistics as it helps to visualize the data in a meaningful way in the form of plots. Inferential statistics, on the other hand, help in finding insights from data analysis.

### **2. Probability**

Probability is the mathematical branch that determines the likelihood of occurrence of any event in a random experiment. As an example, a toss of a coin predicts the probability of getting a red ball from a bag of coloured balls. Probability is a number whose value lies between 0 and 1. The higher the value, the event is more likely to happen.

There are different types of probability, depending upon the type of event. Independent events are the two or more occurrences of an event that are independent of each other. Conditional probability is the probability of occurrence of any event having a relationship with any other event.

### **3. Dimensionality Reduction**

Dimensionality reduction means reducing the dimensions of a data set so that it resolves many problems that do not exist in the lower dimension data. This is because there are many factors in the high dimensional data set and scientists need to create more samples for every combination of features.

This further increases the complexity of data analysis. Therefore, the dimensionality reduction concept resolves all these problems and offers many potential benefits such as lesser redundancy, fast computing, and fewer data to store.

### **4. Central Tendency**

The central tendency of a data set is a single value that describes the complete data by the identification of a central value.

There are different ways to measure the central tendency:

* **Mean:**It is the average value of the data set column.
* **Median:** It is the central value in the ordered data set.
* **Mode:**The value repeating most in the data set column**.**
* **Skewness:**It measures the symmetry of data distribution and determinesif there is a long tail on either or both sides of the normal distribution.
* **Kurtosis**: It defines whether the data has a normal distribution or has tails.

**5. Hypothesis Testing**

Hypothesis testing is to test the result of a survey. There are two types of hypothesis as part of hypothesis testing viz. Null hypothesis and Alternate Hypothesis. The null hypothesis is the general statement that has no relation to the surveyed phenomenon. The Alternate hypothesis is the contradictory statement of the Null hypothesis.

**6. Tests of significance**

Test of significance is a set of tests that helps to test the validity of the cited Hypothesis. Below are some of the tests that help in the acceptance or rejection of the Null Hypothesis.

* **P-value test:**It is the probability value that helps to prove that the null hypothesis is correct or not. If p-value > a, then the Null Hypothesis is correct. If p-value < a, then the Null Hypothesis is False, and we reject it. Here ‘a’ is some significant value which is almost equal to 0.5.
* **Z-Test:**Z-test is another way of testing the Null Hypothesis statement. It is used when the mean of two populations is different, and either their variances are known, or the size of the sample is large.
* **T-test:**A t-test is a statistical test that is performed when either the variance of the population is not known or when the size of the sample is small.

### **7. Sampling theory**

Sampling is the part of statistics that involves the data collection, data analysis, and data interpretation of the data which is collected from a random set of population. Under-sampling and oversampling techniques are followed in case we find the data is not good enough to get the interpretations. Under-sampling involves the removal of redundant data, and oversampling is the technique of imitating the naturally existing data sample.

### **8. Bayesian Statistics**

It is the statistical method that is based on the Bayes Theorem. Bayes theorem defines the probability of occurrence of an event depending upon the prior condition related to an event. Therefore, Bayesian Statistics determine the probability based on previous results.

Bayes Theorem also defines the conditional probability, which is the probability of occurrence of an event considering certain conditions to be true.

## 

## CHAPTER : 2

## MACHINE LEARNING AND DATA MODELLING

Machine learning is training the machine based on a specific data set with the help of a model. This trained model then makes future predictions. There are two types of machine learning modelling, i.e., supervised and unsupervised. The supervised learning works on structured data where we predict the target variable. The unsupervised machine learning works on unstructured data that has no target field.

Supervised machine learning has two techniques: classification and regression. The classification modeling technique is used when we want the machine to predict the category, while the regression technique determines the number. As an example, predicting the future sale of a car is a regression technique and predicting the occurrence of diabetes in a sample of the population is classification.

Below are some of the essential terms related to Machine learning that every Machine Learning Engineer and Data Scientist should know :

1. **Machine Learning:**

  Machine learning is the subset of artificial intelligence where the machine learns from the previous experience and uses that to make predictions for the future.

1. **Machine Learning Model:**

A Machine Learning model is built to train the machine using some mathematical representation which then makes predictions.

1. **Algorithm:**

The algorithm is the set of rules using which a Machine Learning Model gets created.

1. **Regression:**

Regression is the technique used to determine the relationship between independent and dependent variables. There are various regression techniques used for modeling in machine learning based on the data we have. Linear regression is the basic regression technique.

1. **Linear Regression:**

It is the most basic regression technique used in machine learning. It applies to the data where there is a linear relationship between the predictor and the target variable. Thus, we predict the target variable Y based on the input variable X, both of which are linearly related.

The below equation represents the linear regression:

Y=mX + c, where m and c are the coefficients.

There are many other regression techniques, such as Logistic regression, ridge regression, lasso regression, polynomial regression, etc.

1. **Classification:**

Classification is the type of machine learning modeling that predicts the output in the form of a predefined category. Whether a patient will have heart disease or not is an example of a classification technique.

1. **Training set:**

  The training set is part of the data set, which is used to train a machine learning model.

1. **Test set:**

It is part of the data set and has the same structure as the training set and tests the performance of the machine learning model.

1. **Feature:**

It is the predictor variable or an independent variable in the data set.

1. **Target:**

It is the dependent variable in the data set whose value is predicted by the machine learning model.

1. **Overfitting:**

Overfitting is the condition that leads to the overspecialization of the model. It occurs in the case of a complex data set.

1. **Regularization:**

This is the technique used to simplify the model and is a remedy to overfitting.

**2.1 Basic libraries used in Data Science**

Python is the most used language in data science, as it is the most versatile programming language and offers many applications. R is another language used by Data Scientists, but Python is more widely used. Python has a large number of libraries that make the life of a Data Scientist easy. Therefore, every data scientist should know these libraries.

Below are the most used libraries in Data Science:

1. **NumPy:**

  It is the basic library used for numerical computations. It is mainly used for data analysis.

1. **Pandas:**

It is the must-know library which is used for data cleaning, data storage, and time series.

1. **SciPy:**

  It is another python library which is used to solve differential equations and linear algebra.

1. **Matplotlib:**

  It is the data visualization library used to analyse correlation, determine outliers using scatter plot, and to visualize data distribution.

1. **TensorFlow:**

It is used for high-performance computations that reduce error by 50%. It is used for speech, image detection, time series, and video detection.

1. **Scikit-Learn:**

  It is used to implement supervised and unsupervised machine learning models.

1. **Keras:**

It runs easily on CPU and GPU, and supports the neural networks.

1. **Seaborn:**

It is another data visualization library used for multi-plot grids, histograms, scatterplots, bar charts, etc.

**2.2 Diabetes Prediction Using Data Science :**

Machine Learning is the emerging field in the world, various research are done every day on how to improve the accuracy of the models. New and more powerful models are developed to tackle the ever-increasing data in the world. With so much data available new models are made to improve the predictions. Usually in the past ML was mostly used in Forecasting, Sales, Finance etc or was mostly a research field. But now health care sector has seen a big jump in use of ML as a early diagnostic tool. With the help of ML and Hospitals data about patient we can create a very good model that can help us predicting the outcome. In our diabetes probelm also it can be used as the early detector tool to diagnose diabetes

**2.3 Classification Algorithms**

The Supervised Machine Learning algorithm can be broadly classified into Regression and Classification Algorithms. In Regression algorithms, we have predicted the output for continuous values, but to predict the categorical values, we need Classification algorithms.

The Classification algorithm is a Supervised Learning technique that is used to identify the category of new observations on the basis of training data. In Classification, a program learns from the given dataset or observations and then classifies new observation into a number of classes or groups. Such as, Yes or No, 0 or 1, Spam or Not Spam, cat or dog**,** etc. Classes can be called as targets/labels or categories.

Unlike regression, the output variable of Classification is a category, not a value, such as "Green or Blue", "fruit or animal", etc. Since the Classification algorithm is a Supervised learning technique, hence it takes labelled input data, which means it contains input with the corresponding output.

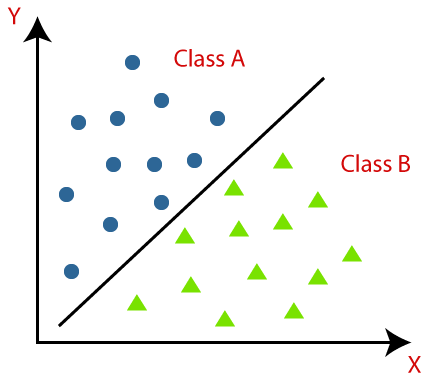
In classification algorithm, a discrete output function(y) is mapped to input variable(x).

y=f(x), where y = categorical output

The best example of an ML classification algorithm is Email Spam Detector.

The main goal of the Classification algorithm is to identify the category of a given dataset, and these algorithms are mainly used to predict the output for the categorical data.

Classification algorithms can be better understood using the below diagram. In the below diagram, there are two classes, class A and Class B. These classes have features that are similar to each other and dissimilar to other classes.



The algorithm which implements the classification on a dataset is known as a classifier. There are two types of Classifications:

**1.Binary Classifier:** If the classification problem has only two possible outcomes, then it is called as Binary Classifier.  
**Examples:** YES or NO, MALE or FEMALE, SPAM or NOT SPAM, CAT or DOG, etc.

**2.Multi-class Classifier:** If a classification problem has more than two outcomes, then it is called as Multi-class Classifier.  
**Example:** Classifications of types of crops, Classification of types of music.

**2.3 Learners In Classification Algorithm**

In the classification problems, there are two types of learners:

1. **Lazy Learners:**

Lazy Learner firstly stores the training dataset and wait until it receives the test dataset. In Lazy learner case, classification is done on the basis of the most related data stored in the training dataset. It takes less time in training but more time for predictions.  
**Example:** K-NN algorithm, Case-based reasoning

1. **Eager Learners:**

Eager Learners develop a classification model based on a training dataset before receiving a test dataset. Opposite to Lazy learners, Eager Learner takes more time in learning, and less time in prediction.

**Example:** Decision Trees, Naïve Bayes, ANN.

**2.4 Types of ML Classification Algorithms:**

Classification Algorithms can be further divided into the Mainly two category:

**1.Linear Models**

* Logistic Regression
* Support Vector Machines

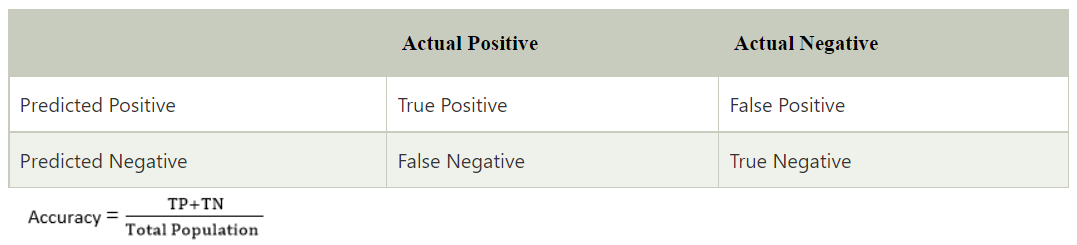
**2.Non-linear Models**

* K-Nearest Neighbours
* Kernel SVM
* Naïve Bayes
* Decision Tree Classification
* Random Forest Classification

## 2.5 Evaluating a Classification model:

**Confusion Matrix:**

* The confusion matrix provides us a matrix/table as output and describes the performance of the model.
* It is also known as the error matrix.
* The matrix consists of predictions result in a summarized form, which has a total number of correct predictions and incorrect predictions.

The matrix looks like as below table:

**TABLE 1.1** Confusion Matrix

**2.6 Use cases of Classification Algorithms**

Classification algorithms can be used in different places. Below are some popular use cases of Classification Algorithms:

* Email Spam Detection
* Speech Recognition
* Identifications of Cancer tumour cells.
* Biometric Identification, etc

**CHAPTER: 3**

**Literature Survey**

Sonu Kumari and Archana Singh proposed [3] an intelligent and effective methodology for the automated detection of Diabetes Mellitus using Neural Network. The paper [4] approached the aim of diagnoses by using ANNs and demonstrated the need for preprocessing and replacing missing values in the dataset being considered. Through the Modified training set, a better accuracy was achieved with lesser time required for training the set. Sajida[8] by using CPCSSN(Canadian primary care sentinel surveillance Network ) dataset and three machine learning methods to predict the diabetes Disses (DD) in early stage to safe human life at from early death .On this study Bagging ,Adaboost,and decision tree(J48) were used to predict the diabetes and the researcher was compare the result of those methods and concluded that Adaboost method was provide effective and better accuracy than the other methods in weka data mining tools. Sadri [20] used Naive Bayes, RBF Network and J48 datamining algorithms for diagnosing type II diabetes. They used WEKA tool. Finally they found Naive Bayes, having the accuracy rate of 76.96% than other algorithms. In this paper[27],Prediction of diabetes is done using ensemble voting classifiers for pima Indian diabetes dataset, in comparison with different classification algorithms, the highest accuracy of 80% and 81% is achieved for data set by using 10-fold cross validation and by spitting data into 30% testing and 70% training. J. Pradeep Kandhasamy, S. Balamurali [58] this research study compare the performance of algorithms those are used to predict diabetes using data mining techniques. Also authors classifiers J48 Decision Tree, KNearest Neighbors, and Random Forest, Support Vector Machines to classify patients with diabetes mellitus. Authors compared four prediction models for predicting diabetes mellitus using 8 important attributes under two different situations. One is before pre-processing the dataset. Here the studies conclude that the decision tree J48 classifier achieves higher accuracy of 73.82 % than other three classifiers. After pre-processing, the dataset given more accurate result when compared to the previous studies. In this case, both KNN (k=1) and Random Forest performance much better than the other three classifiers and they provide 100% accuracy. From this we can come to know that after removing the noisy data from our dataset it will provide good result for our problem.

This paper[5] shows how the Data mining classification algorithms say Naïve Bayes, Logistic Regression, C5.0, SVM and ANN are used to model actual Prediction of Diabetes Mellitus and a comparative analysis are made between them by making use of their Metric Measures say Accuracy, Precision, Sensitivity, Specificity and F1 Score. As a results of the research work, the C5.0 and Logistic Regression are equally good based on their Accuracy measures.Rahul and Minyechil Alehegn [7]studied various data mining techniques and its application . Application of machine learning algorithm were applied in different medical data sets. Single algorithm provided less accuracy than ensemble one.In most study decision tree provided high accuracy. In this study hybrid system Weka and java are the tools to predict diabetes dataset. This paper[18] is about various supervised classifier machine learning algorithms that were applied onto the training set which was obtained by eliminating attributes that did not have much context towards predicting diabetes. This was done using the chi-squared test and only that attributes which were ranked highest and was given more weightage and more likely to predict the onset of diabetes was considered. It was seen that on this training set the Neural Networks algorithm provided the most accurate results. The paper[44] analyses about the three types of diabetes and their causes. It also uses the prediction, classification technique. This provides the higher accuracy for the disease prediction. The research paper[45] explores about various Data mining algorithm approaches of data mining that have been utilized for diabetic disease prediction. In this paper Classification and Naive Bayes is one of the most used algorithms for the prediction of disease .

Pradeep & Dr.Naveen [10] in this paper, the performance of machine learning techniques were compared and measured based on their accuracy. The accuracy of the technique is vary from before pre-processing and after pre-processing as they identified on this study. This indicates the in the prediction of diseases the pre-processing of data set has its own impact on on the performance and accuracy of the prediction. Song [6]describe and explain different classification Algorithms using different parameters such as Glucose, Blood Pressure, Skin Thickness, insulin, BMI, Diabetes Pedigree, and age. The researches were not included pregnancy parameter to predict diabetes disease (DD). In this research, the researchers were using only small sample data for prediction of Diabetes. The algorithms were used by this paper were five different algorithms GMM, ANN, SVM, EM, and Logistic regression. Finally. The researchers conclude that ANN (Artificial Neural Network) was providing High accuracy for prediction of Diabetes. P. Chen[19] in their work have performed statistical testing on medical measurement index results of both patients with diabetes and without diabetes. They have further used boosting algorithms to give excellent classification of diabetes model based on the given medical data. In this study [39]a medical bioinformatics analyses have been accomplished to predict the diabetes. The WEKA software was employed as mining tool for diagnosing diabetes. The Pima Indian diabetes database was acquired from UCI repository used for analysis. The dataset was studied and analyzed to build effective model that predict and diagnoses the diabetes disease. In this study we aim to apply the bootstrapping resampling technique to enhance the accuracy and then applying Naïve Bayes, Decision Trees and (KNN) and compare their performance.

Yunsheng[9] in his study was the new approach that used KNN algorithm by removing the outlier/OOB(out of bag) using DISKR(decrease the size of the training set for K-nearest neighbour .and also in this study the storage space was minimized. There for ,the space complexity is become less and efficient after removing a parameters or instances which have less effect or factor the researchers got better accuracy. V. Kumar and L. Velide,[21] used Data mining Approach for Prediction and Treatment Of diabetes Disease. The techniques they used as Naïve Bayes, JRip, J48 (4.5), DT, NN .They used WEKA tool for implementation. They got 68.5% of accuracy level for J48 algorithm. The research paper [46] elaborates about detailed review of existing data mining methods used for prediction of diabetes. It also gives about the types of diabetes disease Type1, type2, and type3.The aim of the diabetes is to predict the diabetes with the help of Data mining methods such as the K-Nearest Neighbor Algorithm, Bayesian Classifier, Naive Bayesian Classifier, Bayesian Network, all the methods are used for prediction of diabetes. This paper also mentions about the effects of diabetes on patients . In this paper[54,]the proposed methodology aims at providing an efficient hybrid classification framework for predicting and monitoring the Diabetes disease. The main aim of this research is to identify and construct models that would assist medical practitioners in an efficient way by the way benefiting the people to attain longer life in this world.

Monika and Pooja [11] have discussed about previous data processing techniques to retrieve information and current development in the research of medical sciences. Further we have elaborated terminologies and techniques of learning in data mining and machine learning. In this paper, D. Jeevanandhini , E. Gokul Raj ,V. Dinesh Kumar, N. Sasipriyaa [12] conducted performance analysis for type2 diabetes mellitus dataset to improve the accuracy by using clustering and classification algorithm .Here they compared the four prediction model using 8 important attributes .From this studies concludes that Support Vector Machine (SVM) classifier achieves higher accuracy of 77.82 % than other three classifiers. Dr. K. Thangadurai and N.Nandhin[13] used various data mining algorithm and are applied on Pima datasets. It is found that the genetic algorithm gives a better performance over five data mining algorithm. In this paper[49] we have used Matlab tool for analysis and performed comparison of selected classification algorithms. After the comparative analysis we examined that neural network algorithms is more accurate and has less error rate. Our interface also provides the user the choice of selecting suitable prediction algorithm. We conclude that ANN has more precision than other models.

This assessment paper[2] focuses on several predictive analysis procedures and approaches and it is utilizing premature estimate of a several cases of diabetes from patient record. The different approach analytics procedures are applied in health records field for foreseeing case of diabetes and to find out effective ways to treat them in better manner. In this paper [37],the AROC for the proposed GBM model is 84.7% with a sensitivity of 71.6% and the AROC for the proposed Logistic Regression model is 84.0% with a sensitivity of 73.4%. The GBM and Logistic Regression models perform better than the Random Forest and Decision Tree models. In this work[38], we have analyzed the early prediction of diabetes by taking into account various risk factors related to this disease using machine learning techniques. Extracting knowledge from real health care dataset can be useful to predict diabetic patients. To predict diabetes mellitus effectively, we have done our experiments using four popular machine learning algorithms, namely Support Vector Machine (SVM), Naive Bayes (NB), K-Nearest Neighbor (KNN) and C4.5 decision tree, on adult population data to predict diabetes mellitus.

**3.2 Related Works:**

Diabetes prediction is a classification technique with two mutually exclusive

possibleoutcomes, either the person is diabetic or not diabetic. After extensive research,

wecame to conclusion that although numerous classification techniques can be used forthe

purpose of prediction, the observed accuracy varied. On careful examination ofthe

performance of techniques used in prevalent works, logistic regression, KNN,Naive Bayes

[3], random forest, decision tree, and neural network [4], we found themat par when applied

to our dataset. KNN and logistic regression techniques were ableto achieve 80% accuracy.

The primary factor which influenced our algorithm selection was its adaptabilityand

compatibility with future applications. The inevitable shift of data storage towardDNA

makes neural networks the apparent choice. Neural networks use neurons totransmit data

across various layers, with each node working on a different weightedparameter to help

predict diabetes.

The point of this framework is to make an ML model, which can anticipatewith precision the

likelihood or the odds of a patient being diabetic. The ordinarydistinguishing process for the

location of diabetes is that the patient needs to visit asymptomatic focus. One of the key

issues of bio-informatics examination is to achieveprecise outcomes from the information.

Human mistakes or various laboratory testscan entangle the procedure of identification of the

disease. This model can foreseewhether the patient has diabetes or not, aiding specialists to

ensure that the patientin need of clinical consideration can get it on schedule and also help

anticipate theloss of human lives

Diabetes prediction is a classification technique with two mutually exclusive

possibleoutcomes, either the person is diabetic or not diabetic. After extensive research,

wecame to conclusion that although numerous classification techniques can be used forthe

purpose of prediction, the observed accuracy varied. On careful examination ofthe

performance of techniques used in prevalent works, logistic regression, KNN,Naive Bayes

[3], random forest, decision tree, and neural network [4], we found themat par when applied

to our dataset. KNN and logistic regression techniques were ableto achieve 80% accuracy.

The primary factor which influenced our algorithm selection was its adaptabilityand

compatibility with future applications. The inevitable shift of data storage towardDNA

makes neural networks the apparent choice. Neural networks use neurons totransmit data

across various layers, with each node working on a different weightedparameter to help

predict diabetes.

The point of this framework is to make an ML model, which can anticipatewith precision the

likelihood or the odds of a patient being diabetic. The ordinarydistinguishing process for the

location of diabetes is that the patient needs to visit asymptomatic focus. One of the key

issues of bio-informatics examination is to achieveprecise outcomes from the information.

Human mistakes or various laboratory testscan entangle the procedure of identification of the

disease. This model can foreseewhether the patient has diabetes or not, aiding specialists to

ensure that the patientin need of clinical consideration can get it on schedule and also help

anticipate theloss of human lives

Diabetes prediction is a classification technique with two mutually exclusive

possibleoutcomes, either the person is diabetic or not diabetic. After extensive research,

wecame to conclusion that although numerous classification techniques can be used forthe

purpose of prediction, the observed accuracy varied. On careful examination ofthe

performance of techniques used in prevalent works, logistic regression, KNN,Naive Bayes

[3], random forest, decision tree, and neural network [4], we found themat par when applied

to our dataset. KNN and logistic regression techniques were ableto achieve 80% accuracy.

The primary factor which influenced our algorithm selection was its adaptabilityand

compatibility with future applications. The inevitable shift of data storage towardDNA

makes neural networks the apparent choice. Neural networks use neurons totransmit data

across various layers, with each node working on a different weightedparameter to help

predict diabetes.

The point of this framework is to make an ML model, which can anticipatewith precision the

likelihood or the odds of a patient being diabetic. The ordinarydistinguishing process for the

location of diabetes is that the patient needs to visit asymptomatic focus. One of the key

issues of bio-informatics examination is to achieveprecise outcomes from the information.

Human mistakes or various laboratory testscan entangle the procedure of identification of the

disease. This model can foreseewhether the patient has diabetes or not, aiding specialists to

ensure that the patientin need of clinical consideration can get it on schedule and also help

anticipate theloss of human lives

Diabetes prediction is a classification technique with two mutually exclusive possible out comes, either the person is diabetic or not diabetic. After extensive research, we came to conclusion that although numerous classification techniques can be used for the purpose of prediction, the observed accuracy varied. On careful examination of the performance of techniques used in prevalent works, logistic regression, KNN, Naive Bayes[3], random forest, decision tree, and neural network [4], we found the mat par when applied to our dataset. KNN and logistic regression techniques were able to achieve 80% accuracy. The primary factor which influenced our algorithm selection was its adaptability and compatibility with future applications. The inevitable shift of data storage toward DNA makes neural networks the apparent choice. Neural networks use neurons to transmit data across various layers, with each node working on a different weighted parameter to help predict diabetes. The point of this framework is to make an ML model, which can anticipate with precision the likelihood or the odds of a patient being diabetic. The ordinary distinguishing process for the location of diabetes is that the patient needs to visit asymptomatic focus. One of the key issues of bio-informatics examination is to achieve precise outcomes from the information. Human mistakes or various laboratory tests can entangle the procedure of identification of the disease. This model can for see whether the patient has diabetes or not, aiding specialists to ensure that the patient in need of clinical consideration can get it on schedule and also help anticipate the loss of human lives.

**3.3 Methodology**

**3.3.1 Data Set Description:**

The dataset collected is originally from the PIMA Indians Diabetes Database is available on Kaggle. It consists of several medical analyst variables and one target variable. The objective of the dataset is to predict whether the patient has diabetes or not. The dataset consists of several independent variables and one dependent variable. The dataset consists of 25,000+ real time patients’ data and among 22 attributes is one of the attributes is target variable.

* The diabetes data set consists of 25,000 data points, with 22 features each.
* “Outcome” is the feature we are going to predict, 0 means No diabetic, 1 means pr
* e-diabetic and 2 means diabetics, there is no null values in dataset.

|  |  |  |
| --- | --- | --- |
| S.NO | Attribute Names | Description |
| 1 | Diabetes\_012 | 0= no diabetes  1= pre diabetes  2= diabetes |
| 2 | High BP | 0= no high BP  1=high BP |
| 3 | High Cholestrol | 0= no high cholesterol  1= high cholesterol |
| 4 | Cholestrol Check | 0= no cholesterol check in 5 years  1= yes cholesterol check in 5 years |
| 5 | BMI | Body Mass Index |
| 6 | Smoker | Have you smoked at least 100 cigarettes in your entire life?  [note: 5 packs = 100 cigarettes]  0=no  1=yes |
| 7 | Stroke | (Ever told) you have a Stroke  0=no  1=yes |
| 8 | Heart Diseases Attack | Coronary heart disease (CHD) or myocardial infarction (MI)  0=no  1=yes |
| 9 | Physical Activity | Physical activity in past 30 days-not including job  0=no  1= yes |
| 10 | Fruits | Consume Fruit 1 or more times per day  0= no  1=yes |
| 11 | Veggies | Consume vegetables 1 or more times per day  0=no  1=yes |
| 12 | Heavy Alcohol Consumption | Heavy drinkers (adult men having more than 14 drinks per week and adult women having more than 7 drinks per week)  0=no  1=yes |
| 13 | Any Healthcare | Have any kind of health care coverage, including health insurance, prepaid plans such as HMO, etc.  0=no  1=yes |
| 14 | No Doctor Cost | Was there a time in the past 12 months when you needed to see a doctor but could not because of cost.  0=no  1=yes |
| 15 | General Health | Would you say that in general your health is:  Scale 1-5  1=excellent  2= very good  3= good  4= fair  5= poor |
| 16 | Mental Health | Now thinking about your mental health, which includes stress, depression, and problems with emotions. |
| 17 | Physical Health | Now thinking about your physical health, which includes physical illness and injury, for how many days during the past 30. |
| 18 | Difference in Walk | Do you have serious difficulty walking or climbing stairs?  0=no  1=yes |
| 19 | Sex | 0=female  1=male |
| 20 | Age | 13-level age category (\_AGE5YR see codebook)  1=18-24  9=60-64  13=80 or older. |
| 21 | Education | Education level (EDUCA see codebook)  Scale1-6  1=Never attended school or only kindergarten  2= Grades 1 through 8 |
| 22 | Income | Income scale (INCOME 2 see codebook)  Scale 1-8  1= less than $10,000  5= less than $35,000  8= $75,000 or more |

**TABLE 3.3.1 :** Dataset Description

The diabetes data set consists of 2000 data points, with 9 features each.

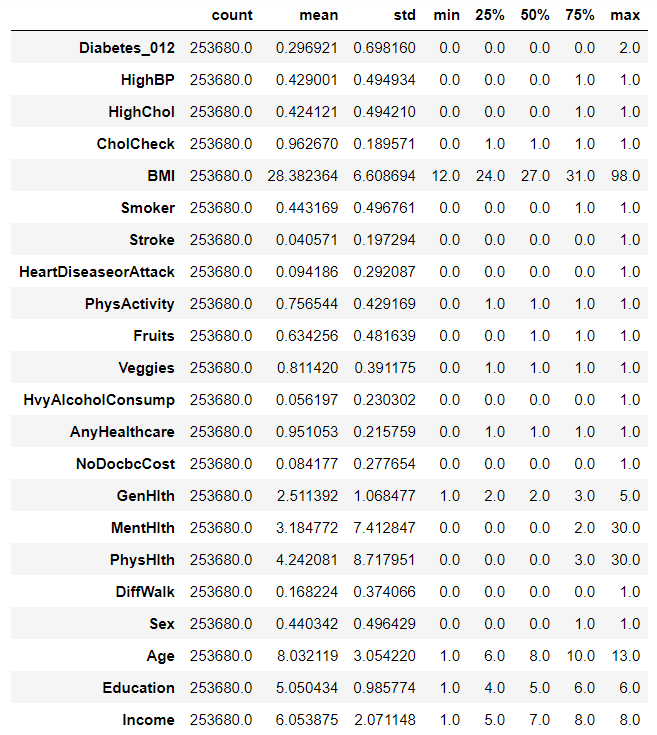
➔ “Outcome” is the feature we are going to predict, 0 means No diabetes, 1 means diabetes.

The diabetes data set consists of 2000 data points, with 9 features each.

➔ “Outcome” is the feature we are going to predict, 0 means No diabetes, 1 means diabetes.

The diabetes data set consists of 2000 data points, with 9 features each.

➔ “Outcome” is the feature we are going to predict, 0 means No diabetes, 1 means diabetes.



The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

consists of several independent variables and one dependent variable

The dataset collected is originally from the Pima Indians Diabetes Database is

available on Kaggle. It consists of several medical analyst variables and one target variable.

The objective of the dataset is to predict whether the patient has diabetes or not. The dataset

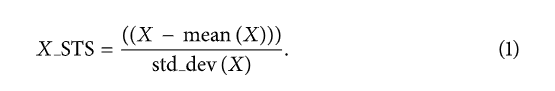
consists of several independent variables and one dependent variable

**Figure 3.3.1.1 :** Predictions

**3.3.2 Data Pre-processing :**

The term "data pre-processing" refers to the process of converting unstructured data to structured data, as well as resizing and removing undesirable data from a dataset. The dataset's missing traits are replaced by the mean value. The data is then randomly selected from the dataset to ensure that the data is circulated properly.

Data pre-processing is a crucial stage that transforms the data into a usable and efficient format, so that it can fit as an input to the machine learning algorithm. In our system, only one technique has been used for data pre-processing, which is data normalization. This latter is generally considered as the process of data structuring. It is also called StandardScaler normalization, where all the values of the attributes are within [−1, 1]. The StandardScaler formula is shown below in equation ([1](https://www.hindawi.com/journals/complexity/2021/6053824/#EEq1)), where *X* represents the input columns of the dataset to transform and *X*\_STS represents the transformed ones [39].



**3.3.3 Training and Testing Phase :**

This phase extracts the features from the dataset, and the testing phase will deliver new data to be examined to see how well our algorithm works and behaves when it comes to prediction. As previously stated, the dataset is divided into two pieces. To avoid fitting, cross-validation is performed. Each iteration of our approach uses a ten-fold approach to portion data, with nine-fold used for training and the remaining for testing.

**CHAPTER 4**

**PROPOSED MODEL**

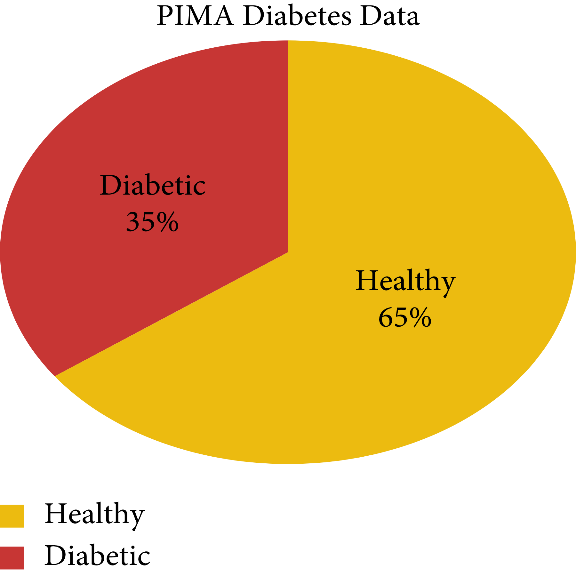
The proposed diabetes classification and prediction algorithm is evaluated on a publicly available PIMA Indian Diabetes dataset (<https://www.niddk.nih.gov/health-information/diabetes>). Besides, a comparative analysis is performed with state-of-the-art algorithms. The experimental results show the supremacy of the proposed algorithm as compared to state-of-the-art algorithms. The details of the dataset, performance measures, and comparative analysis performed are described in the following sections.

##### **4.1 Dataset :**

This study used the PIMA Indian Diabetes (PID) dataset taken from the National Institute of Diabetes and Kidney Diseases center [59]. The primary objective of using this dataset is to build an intelligent model that can predict whether a person has diabetes or not, using some measurements included in the dataset. There are eight medical predictor variables and one target variable in the dataset. Diabetes classification and prediction are a binary classification problem.

The dataset consists of 25,000+ records of different healthy and diabetic both male and female patients of age greater than twenty-one, as shown in Figure 4.1. The target variable outcome contains three values, 0, 1 and 2. The primary objective of using this dataset was to predict diabetes diagnostically. Whether a user has a chance of diabetes in the coming four years in both men and women belongs to PIMA Indian.

The dataset has a total of eight variables: glucose tolerance, no. of pregnancies, body mass index, blood pressure, age, insulin, and Diabetes Pedigree Function. All twenty two attributes are used for the training dataset in the classification model in this work.



**Figure 4.1 :** PIMA Data Distribution

**4.2 Data Pre-processing :**

**4.2.1 Categorical Variable Conversion :**

The dataset included both numerical and categorical features. Among which the ‘Diagnosis’ column had categorical feature, which says if the cancer is M = malignant or B = benign. The rest of the features are numerical. Most of the algorithms produce better result with numerical variable. In python, library “sklearn” requires features in numerical arrays and categorical variables cannot be fitted into a regression equation in their raw form. Hence, Label Encoder was used to transform non-numerical to numerical labels.

**4.2.2 Feature Scaling :**

The range of values of the attributes in the dataset varies widely and feature scaling is used to bring it to a standardized range. It is also known as data normalization. This is done because some algorithms will not function properly without it and data should be standardized before applying PCA as variables with higher and lower variance are going to be treated differently. In this paper, scikit-learn module sklearn pre-processing Standard Scaler is used to implement standardization in python. The standard score of a sample x is calculated as: z = x−u s

**4.2.3 Principal Component Analysis (PCA) :**

Principal Component Analysis is a method of dimension-reduction which reduces a large set of variables to a small set that still contains most of the information in the large set. PCA is basically a mathematical procedure that transforms a number of correlated variables into a smaller number of uncorrelated linear variables called principal components using uses an orthogonal transformation [1]. After standardizing the data, PCA was applied for Random Forest, SVM, Decision Tree, XG Boost, K-Nearest Neighbour, Logistic Regression and Gaussian Naïve Bayes classifiers.

**4.3.5 Train-Test Split :**

The data is normally split into two subsets: training data and testing data (and sometimes to three: train, validate and test). The training dataset is the actual dataset that is used to train the model (weights and biases in the case of Neural Network). The model sees and learns from this data. The test dataset is the sample of data used to provide an unbiased evaluation of a final model fit on the training dataset. While validation dataset is the sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyperparameters, but this is for frequent evaluation. The splitting ratio depends on the total number of samples and the actual training model. The train/test split was implemented using the train\_test\_split class of scikit-learn’s model\_selection package into a 70:30 ratio with 70% going to the training set and the rest to the test set which is considered to be ideal [8]. The Accuracy, Precision, Recall and F1 score for RF, SVM, KNN, LR, DT, XG and GNB was evaluated by using cross\_val\_score from sklearn. Cross validation package using 10 folds on the test dataset.

**4.3 Algorithms :**

The model deals with binary classification of labeled data and the algorithms were chosen based on that fact. But deep learning models like Artificial Neural Network and Convolutional Neural Network were also tested on this as there is no perfectly tailored single solution or one approach that fits all. The machine learning algorithms chosen for this problem are Random Forest (RF). Support Vector Machine (SVM), Logistic Regression (LR), XG Boost, K-Nearest Neighbour(KNN), Gaussian Naïve Bayes and Decision Tree. The results of the algorithms were compared to determine the best classifier for the problem.

###### **4.3.1 Random Forest :**

Random Forest is one of the most popular and powerful machine learning algorithm. It consists of many decision trees and outputs the class that is the mode of the class’s output by means of individual trees. After building decision trees on the sample sets, many trees are generated, thus creating a forest. It is good for classification problems like this one and other tasks like regression which functions as explained above by creating a multitude of trees at training and outputting the classes or mean predictions of each specific tree [9][3]. The numerous deep decision trees are trained on separate groups of the same dataset and averaged with the target of decreasing the variance [7].

RF is one of the most common uses of classifier integration. As shown in Figure 4.3.1, RF is made up of numerous separate Decision Tree classifiers that vote on test samples according to a set of criteria [53, 54]. The steps are as follows:

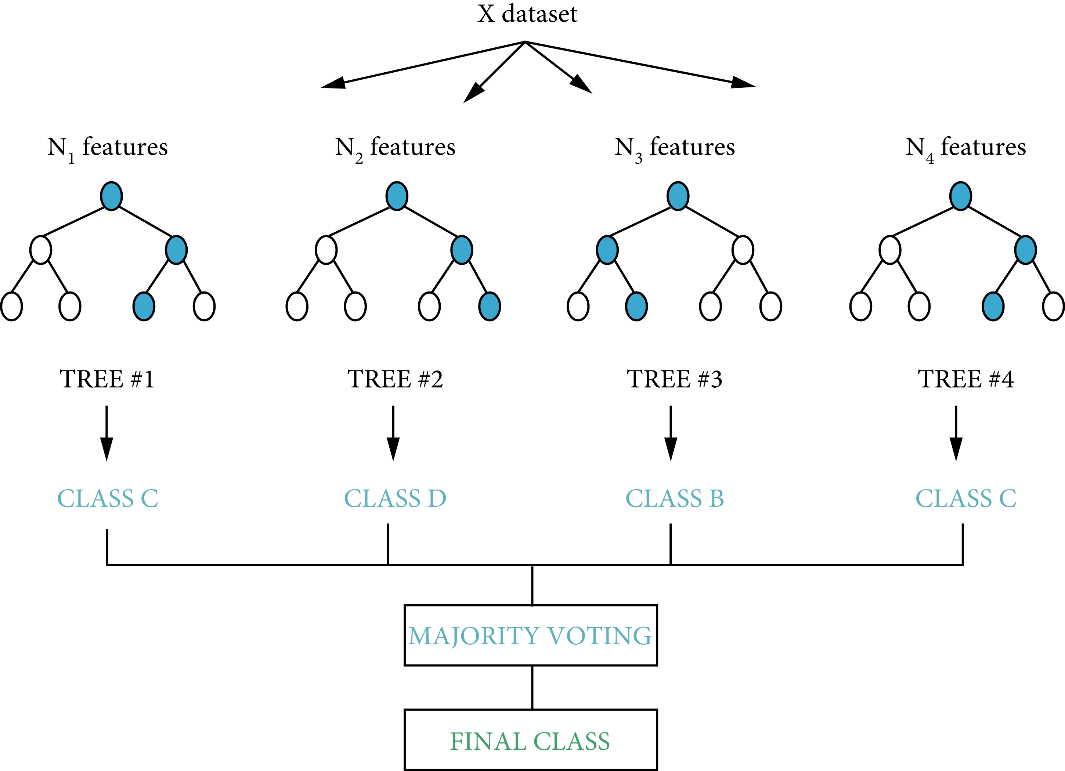
(i)Extracting some samples from the training set as a training subset using the bootstrap method, which is a self-help sampling approach.

(ii)A number of features are randomly picked from the feature set for the training subset as the basis for splitting each node of the Decision Tree.

(iii)Repeat steps (i)-(ii) to generate a large number of training subsets and Decision Trees, which are then combined to build a Random Forest.

(iv)The test set's samples are fed into the Random Forest, where each Decision Tree makes a choice based on the data. After receiving the findings, the results are voted on using a voting technique to determine the sample categorization results.

(v)Repeat step (iv) until all of the test sets have been classified [55].

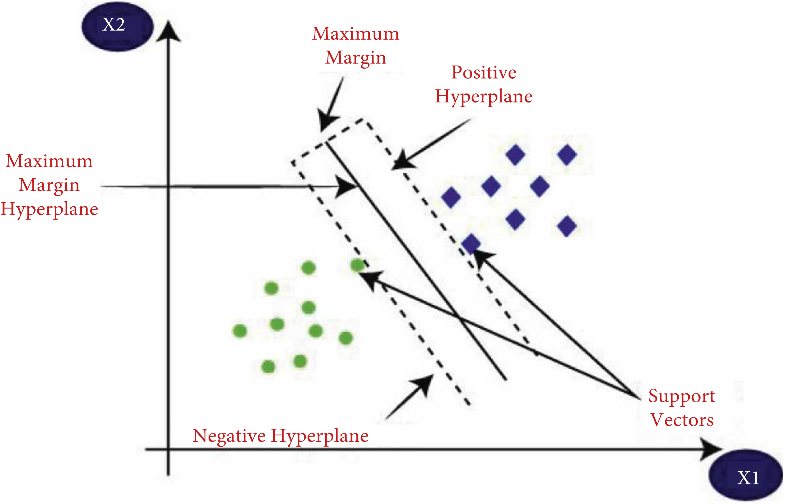


**Figure 4.3.1 :** Random Forest First Stage

###### **4.3.2 Support Vector Machine(SVM) :**

Support Vector Machine (SVM) is a supervised machine learning technique that is broadly used in pattern recognition and classification problems, especially when the dataset has exactly two classes. SVMs are used to find the ideal hyperplane which separates the classes. An input pattern, called feature vector, is taken by the classifier and determines to which class it belongs to. It classifies linearly separable data, but in general, the feature vectors might not be linearly separable. The kernel trick is used to fix this [11]. SVM uses kernel methods to map input data to higher dimensional data and provides a fast training algorithm. It is used for pattern classification and regression. The performance of an SVM classifier matters on the choice of the kernel function. Different kernel functions are used for different classification tasks. In this project, SVC class from scikit-learn was used to implement SVM. However, SVM can be memory-intensive and complicated to interpret and tune.

SVM is a non-probabilistic classifier with a separating hyperplane as its formal definition. The technique creates an ideal hyperplane with the greatest distance from the support vectors based on the available training data (supervised learning). This hyperplane is a line that divides a plane into two classes in two-dimensional space. The epsilon *ε*, regularization, and kernel parameters are the SVM classifier's tuning parameters [6, 44]. The principle of SVM is shown below in Figure 4.3.2.



**Figure 4.3.2 :** Support Vector Machine(SVM)

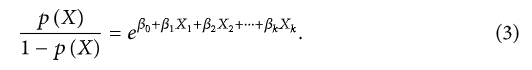
###### **4.3.3 Logistic Regression :**

Logistic Regression (LR) is a subset of generalized linear models which deals with the analysis of binary data, which seeks out the best-fitting model for describing the connection between dependent and independent predictors [40, 41]. When it comes to predicting sickness or health status, the LR model is most commonly used [42, 43]. Based on the risk factors given, the LR model can calculate the likelihood of an individual acquiring diabetes disease [43].

If a person suffers from diabetes disease, the value of target is 1;otherwise, target is 0. We determined that the probability of an individual developing diabetes disease is *P* (*X*). The LR model's formula is defined as follows:



After exponentiating both sides, we obtain



The probability of an individual developing diabetes disease can be written as

###### **4.3.4 XGBoost :**

XGBoost is an open-source software library that implements optimized distributed gradient boosting machine learning algorithms under the Gradient Boosting framework.

XGBoost, which stands for Extreme Gradient Boosting, is a scalable, distributed gradient boosted decision tree (GBDT) machine learning library. It provides parallel tree boosting and is the leading machine learning library for regression, classification, and ranking problems.

It’s vital to an understanding of XGBoost to first grasp the machine learning concepts and algorithms that XGBoost builds upon: supervised machine learning, decision trees, ensemble learning, and gradient boosting.

Supervised machine learning uses algorithms to train a model to find patterns in a dataset with labels and features and then uses the trained model to predict the labels on a new dataset’s features.

## XGBoost Benefits and Attributes :

The list of benefits and attributes of XGBoost is extensive, and includes the following:

* A large and growing list of data scientists globally that are actively contributing to XGBoost open source development
* Usage on a wide range of applications, including solving problems in regression, classification, ranking, and user-defined prediction challenges
* A library that’s highly portable and currently runs on OS X, Windows, and Linux platforms
* Cloud integration that supports AWS, Azure, Yarn Clusters and other ecosystems
* Active production use in multiple organizations across various vertical market areas
* A library that was built from the ground up to be efficient, flexible, and portable

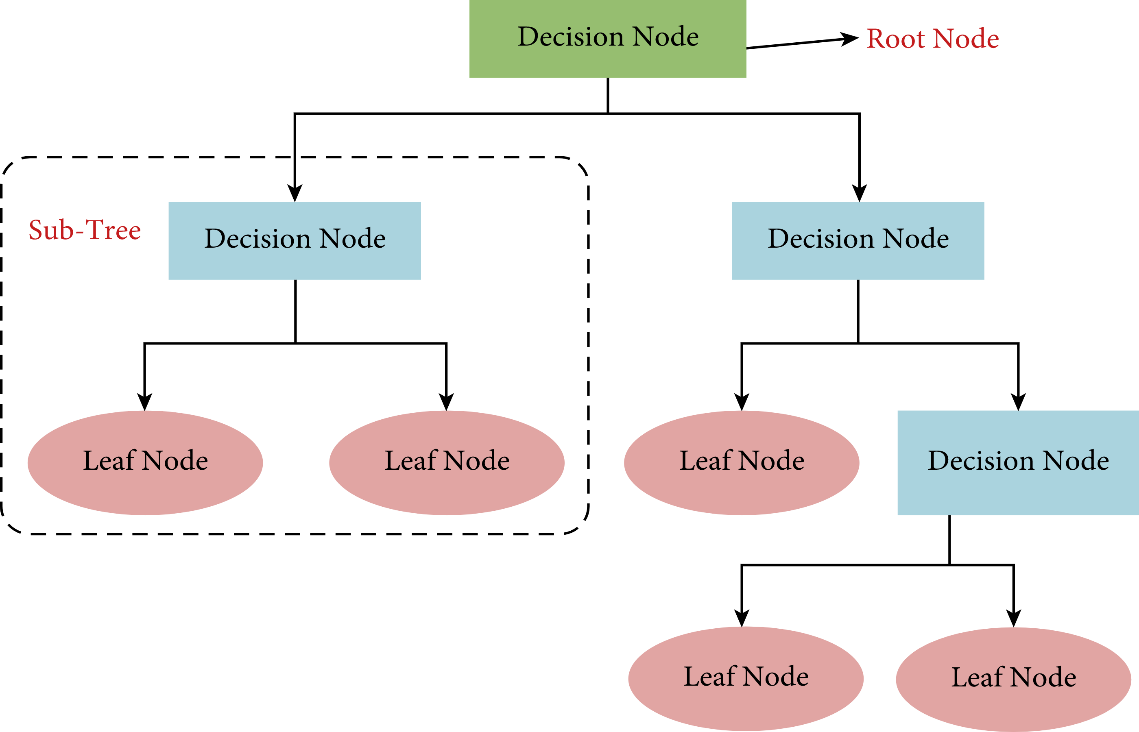
###### **4.3.5 Decision Tree :**

Decision Tree is a Supervised learning techniquethat 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. In a Decision tree, there are two nodes, which are the Decision Node andLeaf Node. 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. The decisions or the test are performed on the basis of features of the given dataset. Itis agraphical representation for getting all the possible solutions to a problem/decision based on given conditions. It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure. In order to build a tree, we use the CART algorithm, which stands for Classification and Regression Tree algorithm. A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.

There are various algorithms in Machine learning, so choosing the best algorithm for the given dataset and problem is the main point to remember while creating a machine learning model. Below are the two reasons for using the Decision tree:

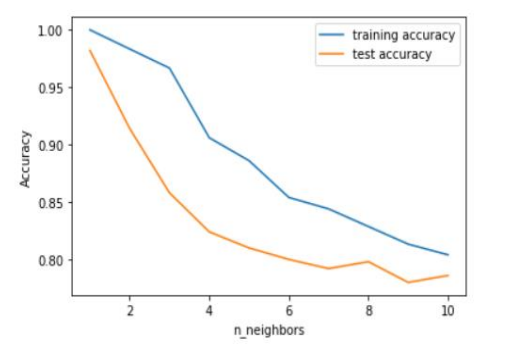
* Decision Trees usually mimic human thinking ability while making a decision, so it is easy to understand.
* The logic behind the decision tree can be easily understood because it shows a tree-like structure

DT is a nonparametric supervised learning algorithm for regression and classification tasks. DT (Figure 4.3.5) can be seen as a construction model that includes root node, division, and leaf node. Each internal node represents a test on an attribute, each division represents the outcome of test, and each leaf node grips the class label. The opening node in the tree is the root node. First, an attribute is selected and sited at the root node. Then, a division is made for each possible value. This splits dataset into subgroups, one for every value of the attribute. The tree process is recursively repeated for each division using only those cases that reach the branch. When all cases on a node have the same classification, the tree progress can be stopped. Usually, entropy or classification error is used to define the best tree division [51, 52].



**Figure 4.3.5 :** Decision Tree Principle

**4.3.6 K-Nearest Neighbor(KNN) :**

****The k-NN algorithm is arguably the simplest machine learning algorithm. Building the model consists only of storing the training data set. To make a prediction for a new data point, the algorithm finds the closest data points in the training data set, its “nearest neighbors.” First, let’s investigate whether we can confirm the connection between model complexity and accuracy:

The above plot shows the training and test set accuracy on the y-axis against the setting of n\_neighbors on the x-axis. Considering if we choose one single nearest neighbor, the prediction on the training set is perfect. But when more neighbors are considered, the training accuracy drops, indicating that using the single nearest neighbor leads to a model that is too complex. The best performance is somewhere around 9 neighbors.

**4.3.7 Naive Bayes :**

The Naive Bayes Algorithm is another Machine Learning algorithm for classification problems. Naive Bayes is an efficient classification algorithm in data mining that can handle missing values during classification [8] [9]. Naive Bayes Algorithm is pretty fast Machine Learning and efficient model, basically, this model is used for text classification few known examples are spam filtration, sentimental analysis and classifying new articles. The named of Naive is called for it's some sort of features distinct of an event of another feature. And Bayes refers to the statistician and philosopher Thomas Bayes theorem [10]. The NB theorem can be expressed mathematically as follows:

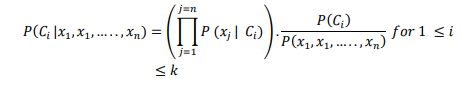
****

* P (A | B): Probability of occurrence of event A given the event B is true.
* P (A) and P (B): Probabilities of the occurrence of event A and B respectively.
* P (B | A ): Probability of the occurrence of event B given the event A is true.

Bayes Theorem for Naive Bayes Algorithm state the following relationship

****

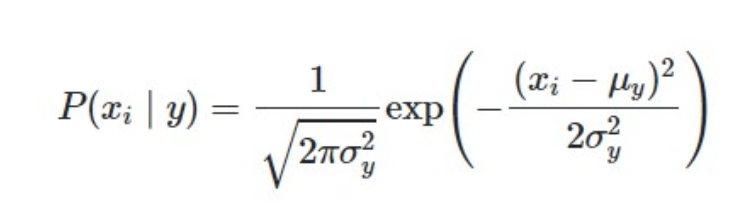
This relationship can be simplified as follows

****

**Gaussian Naïve Bayes :**

Naive Bayes Classifiers are based on the Bayes Theorem. One assumption taken is the strong independence assumptions between the features. These classifiers assume that the value of a particular feature is independent of the value of any other feature. In a supervised learning situation, Naive Bayes Classifiers are trained very efficiently. Naive Bayed classifiers need a small training data to estimate the parameters needed for classification. Naive Bayes Classifiers have simple design and implementation and they can applied to many real life situations.

When working with continuous data, an assumption often taken is that the continuous values associated with each class are distributed according to a normal (or Gaussian) distribution. The likelihood of the features is assumed to be



Sometimes assume variance

* is independent of Y (i.e., σi),
* or independent of Xi (i.e., σk)
* or both (i.e., σ)

Gaussian Naive Bayes supports continuous valued features and models each as conforming to a Gaussian (normal) distribution.

An approach to create a simple model is to assume that the data is described by a Gaussian distribution with no co-variance (independent dimensions) between dimensions. This model can be fit by simply finding the mean and standard deviation of the points within each label, which is all what is needed to define such a distribution.

**4.3.8 AdaBoost Classifier :**

AdaBoost (Adaptive Boosting) is a Boosting machine learning meta-algorithm which theoretically can be used to significantly reduce the error of any learning algorithm that consistently generates classifiers whose performance is a little better than random guessing. It is a nominal class classifier using the Adaboost ensemble method which means only nominal class problems can be solved. It is Often dramatically improves performance, but sometimes over fits. On the other hand, Bagging is an ensemble method that creates separate samples of the training dataset and creates a classifier for each sample. It reduces the variance .

Each instance in the training dataset is weighted. The initial weight is set to:

weight(xi) = 1/n

Where xi is the i’th training instance and n is the number of training instances

**4.3.9 Gradient Boosting Classifier :**

Gradient boosting method creates step-wise process and increments the algorithm on basis of loss function [10]. The errors are detected and rectified to improve the accuracy. Generally, boosting checks models which decrease the loss function obtained from trained samples. From these calculations the errors are measured and analysed for optimal prediction of results. Loss function calculates the range of detected rate which compares with desired target. Onward stepwise process is most popular method for updating different with various attributes. The accuracy is optimized by reducing loss function and adding base learners at all stages.

**Steps for gradient Boosting method :**

1. Consider a sample of target values as P

2. Estimate the error in target values

3. Update and adjust the weights to reduce error M

4. P[x]=p[x]+alpha M[x]

5. Model Learners are analysed and calculated by loss function F

6. Repeat steps till desired & target result P

**CHAPTER : 5**

**USER INTERFACE**

The user interface (UI) is the point of human-computer interaction and communication in a device. This can include display screens, keyboards, a mouse and the appearance of a desktop. It is also the way through which a user interacts with an application or a website.

The growing dependence of many businesses on web applications and mobile applications has led many companies to place increased priority on UI in an effort to improve the user's overall experience.

### **Types of user interfaces**

The various types of user interfaces include:

* graphical user interface (GUI)
* command line interface (CLI)
* menu-driven user interface
* touch user interface
* voice user interface (VUI)
* form-based user interface
* Natural Language user interface

### **Examples of user interfaces**

Some examples of user interfaces include:

* computer mouse
* remote control
* virtual reality
* ATMs
* speedometer
* the old iPod click wheel

Websites such as Airbnb, Dropbox and Virgin America display strong user interface design. Sites like these have created pleasant, easily operable, User Centered Design  (UCD) that focus on the user and their needs.

### **UI and UX**

The UI is often talked about in conjunction with user experience (UX), which may include the aesthetic appearance of the device, response time and the content that is presented to the user within the context of the user interface. Both terms fall under the concept of human-computer interaction (HCI), which is the field of study focusing on the creation of computer technology and the interaction between humans and all forms of IT design. Specifically, HCI studies areas such as UCD, UI design and UX design.

An increasing focus on creating an optimized user experience has led some to carve out careers as UI and UX experts. Certain languages, such as HTML and CSS, have been geared toward making it easier to create a strong user interface and experience.

### **Graphical UIs**

Elements of a GUI include such things as windows, pull-down menus, buttons, scroll bars and icons. With the increasing use of multimedia as part of the GUI, sound, voice, motion video and virtual reality are increasingly becoming the GUI for many applications.

### **Mobile UIs**

The emerging popularity of mobile applications has also affected UI, leading to something called mobile UI. Mobile UI is specifically concerned with creating usable, interactive interfaces on the smaller screens of smartphones and tablets and improving special features, like touch controls.

**5.1 Web Application**

A web-application is an application program that is usually stored on a remote server, and users can access it through the use of **Software** known as **web-browser.**

It is a type of computer program that usually runs with the help of a web browser and also uses many web technologies to perform various tasks on the internet.

A web application can be developed for several uses, which can be used by anyone like it can be used as an individual or as a whole organization for several reasons.

In general, a web application can contain online shops (or we can also say them e-commerce shops), webmail's, calculators, social media platforms, etc. There is also some kind of web application that usually requires a special kind of web browser to access them. We cannot access those kinds of web applications by using regular web- browsers. However, most of the web applications available on the internet can be accessed using a **standard web browser.**

If we talk about the web application in general, a web application usually uses a combination of the server-side scripts such as PHP, ASP [for handling the information/ data storage and retrieval of the data. Some of them also use the client-side scripts such as](https://www.javatpoint.com/php-tutorial)[JavaScript, HTML](https://www.javatpoint.com/javascript-tutorial)[to represent the data/information in front of the users, and some of the web applications are also using both](https://www.javatpoint.com/html-tutorial)**[server-side](https://www.javatpoint.com/html-tutorial)**[and](https://www.javatpoint.com/html-tutorial)**[client-side](https://www.javatpoint.com/html-tutorial)**[at the same time.](https://www.javatpoint.com/html-tutorial)

[It allows the users to communicate with the organization or companies by using the online form, online forums, shopping carts, content management system, and much more.Apart from that web applications also allow its users to create documents, share them, or share the data/ information. By using the web application, users can collaborate on same projects by event when they are not available on the same geographical location.](https://www.javatpoint.com/html-tutorial)

**[5.1.1 FLASK](https://www.javatpoint.com/php-tutorial)**

[Flask is a web framework that provides libraries to build lightweight web applications in python. It is developed by](https://www.javatpoint.com/php-tutorial)**[Armin Ronacher](https://www.javatpoint.com/php-tutorial)**[who leads an international group of python enthusiasts (POCCO). It is based on WSGI toolkit and jinja2 template engine. Flask is considered as a micro framework.](https://www.javatpoint.com/php-tutorial)

## Features of Flask

Some features which make Flask an ideal framework for web application development are:

1. Flask provides a development server and a debugger.
2. It uses Jinja2 templates.
3. It is compliant with WSGI 1.0.
4. It provides integrated support for unit testing.
5. Many extensions are available for Flask, which can be used to enhance its functionalities.

**5.1.2 JINJA TEMPLATE**

Jinja is a fast, expressive, extensible templating engine. Special placeholders in the template allow writing code similar to Python syntax. Then the template is passed data to render the final document.

It includes:

* Template inheritance and inclusion.
* Define and import macros within templates.
* HTML templates can use autoescaping to prevent XSS from untrusted user input.
* A sandboxed environment can safely render untrusted templates.
* Async support for generating templates that automatically handle sync and async functions without extra syntax.
* I18N support with Babel.
* Templates are compiled to optimized Python code just-in-time and cached, or can be compiled ahead-of-time.
* Exceptions point to the correct line in templates to make debugging easier.
* Extensible filters, tests, functions, and even syntax.

Jinja’s philosophy is that while application logic belongs in Python if possible, it shouldn’t make the template designer’s job difficult by restricting functionality too much.

**5.2 Web Interface**

The simplest type of web management is your web user interface or interface web. This is due to the point-and-select capabilities enabling you to jumpstart your firewall management. To define web interface, it is a straightforward interface with configuration options located on your browser’s left-hand side. Your menu is either based on the DHTML or dynamic hypertext markup language default or is Java-based. There is no difference in functionality but there are slight variations to the look and feel.

To define a web interface, you must understand it is configured by default to only work over your HTTP or hypertext transfer protocol. You can also configure your web interface for working over your HTTPS or hypertext transfer protocol secure. This offers you the mechanism required for securing your web management traffic. You can use the majority of popular web browsers with your website interface including:

* Firefox
* Internet Explorer
* Chrome

The question of what is a web interface can be answered by a mechanism enabling you to interact with the software or content you are running through your web browser on a remote server. Your web server downloads your web page content enabling you to use your browser to interact with the content. Your browser performs the function of a client. you can store your content on a remote server due to the distributed nature of your web browser in addition to receiving convenient content access.

There is a tremendous amount of data currently in existence due to these kinds of interfaces. The most frequently used web applications include:

* Webmail
* Instant messaging
* Online shopping
* Online document sharing
* Social media

**5.2.1 HTML/CSS**

**HTML**

HTML is an acronym which stands for **Hyper Text Markup Language** which is used for creating web pages and web applications. Let's see what is meant by Hypertext Markup Language, and Web page.

**Hyper Text:** HyperText simply means "Text within Text." A text has a link within it, is a hypertext. Whenever you click on a link which brings you to a new webpage, you have clicked on a hypertext. HyperText is a way to link two or more web pages (HTML documents) with each other.

**Markup language:** A markup language is a computer language that is used to apply layout and formatting conventions to a text document. Markup language makes text more interactive and dynamic. It can turn text into images, tables, links, etc.

**Web Page:** A web page is a document which is commonly written in HTML and translated by a web browser. A web page can be identified by entering an URL. A Web page can be of the static or dynamic type. **With the help of HTML only, we can create static web pages.**

Hence, HTML is a markup language which is used for creating attractive web pages with the help of styling, and which looks in a nice format on a web browser. An HTML document is made of many HTML tags and each HTML tag contains different content.

## Features of HTML

1) It is a very **easy and simple language**. It can be easily understood and modified.

2) It is very easy to make an **effective presentation** with HTML because it has a lot of formatting tags.

3) It is a **markup language**, so it provides a flexible way to design web pages along with the text.

4) It facilitates programmers to add a **link** on the web pages (by html anchor tag), so it enhances the interest of browsing of the user.

5) It is **platform-independent** because it can be displayed on any platform like Windows, Linux, and Macintosh, etc.

6) It facilitates the programmer to add **Graphics, Videos, and Sound** to the web pages which makes it more attractive and interactive.

7) HTML is a case-insensitive language, which means we can use tags either in lower-case or upper-case.

**CSS**

CSS stands for Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. It can also be used with any kind of XML documents including plain XML, SVG and XUL.

CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications and user interfaces for many mobile applications.

These are the three major benefits of CSS:

## 1) Solves a big problem

Before CSS, tags like font, color, background style, element alignments, border and size had to be repeated on every web page. This was a very long process. For example: If you are developing a large website where fonts and color information are added on every single page, it will be become a long and expensive process. CSS was created to solve this problem. It was a W3C recommendation.

## 2) Saves a lot of time

CSS style definitions are saved in external CSS files so it is possible to change the entire website by changing just one file.

## 3) Provide more attributes

CSS provides more detailed attributes than plain HTML to define the look and feel of the website.

**5.2.2 Bootstrap**

* Bootstrap is the most popular HTML, CSS and JavaScript framework for developing a responsive and mobile friendly website.
* It is absolutely free to download and use.
* It is a front-end framework used for easier and faster web development.
* It includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many others.
* It can also use JavaScript plug-ins.
* It facilitates you to create responsive designs.

Following are the main advantage of Bootstrap:

* It is very easy to use. Anybody having basic knowledge of HTML and CSS can use Bootstrap.
* It facilitates users to develop a responsive website.
* It is compatible on most of browsers like Chrome, Firefox, Internet Explorer, Safari and Opera etc.



**Figure 5.2.2 :** Bootstrap

### **Bootstrap package contains**

**Scaffolding:** Bootstrap provides a basic structure with Grid System, link styles, and background.

**CSS:** Bootstrap comes with the feature of global CSS settings, fundamental HTML elements style and an advanced grid system.

**Components:** Bootstrap contains a lot of reusable components built to provide iconography, dropdowns, navigation, alerts, pop-overs, and much more.

**JavaScript Plugins:** Bootstrap also contains a lot of custom jQuery plugins. You can easily include them all, or one by one.

**Customize:** Bootstrap components are customizable and you can customize Bootstrap's components, LESS variables, and jQuery plugins to get your own style.

**5.2.3 JavaScript**

JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

JavaScript was first known as LiveScript**,** but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name LiveScript. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers.

The [ECMA-262 Specification](http://www.ecma-international.org/publications/index.html) defined a standard version of the core JavaScript language.

* JavaScript is a lightweight, interpreted programming language.
* Designed for creating network-centric applications.
* Complementary to and integrated with Java.
* Complementary to and integrated with HTML.
* Open and cross-platform

## Client-Side JavaScript

Client-side JavaScript is the most common form of the language. The script should be included in or referenced by an HTML document for the code to be interpreted by the browser.

It means that a web page need not be a static HTML, but can include programs that interact with the user, control the browser, and dynamically create HTML content.

The JavaScript client-side mechanism provides many advantages over traditional CGI server-side scripts. For example, you might use JavaScript to check if the user has entered a valid e-mail address in a form field.

The JavaScript code is executed when the user submits the form, and only if all the entries are valid, they would be submitted to the Web Server.

JavaScript can be used to trap user-initiated events such as button clicks, link navigation, and other actions that the user initiates explicitly or implicitly.

## Advantages of JavaScript

The merits of using JavaScript are :

* **Less server interaction :** You can validate user input before sending the page off to the server. This saves server traffic, which means less load on your server.
* **Immediate feedback to the visitors :** They don't have to wait for a page reload to see if they have forgotten to enter something.
* **Increased interactivity :** You can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard.
* **Richer interfaces :** You can use JavaScript to include such items as drag-and-drop components and sliders to give a Rich Interface to your site visitors.

## Limitations of JavaScript

We cannot treat JavaScript as a full-fledged programming language. It lacks the following important features −

* Client-side JavaScript does not allow the reading or writing of files. This has been kept for security reason.
* JavaScript cannot be used for networking applications because there is no such support available.
* JavaScript doesn't have any multi-threading or multiprocessor capabilities.

Once again, JavaScript is a lightweight, interpreted programming language that allows you to build interactivity into otherwise static HTML pages.

## JavaScript Development Tools

One of major strengths of JavaScript is that it does not require expensive development tools. You can start with a simple text editor such as Notepad. Since it is an interpreted language inside the context of a web browser, you don't even need to buy a compiler.

To make our life simpler, various vendors have come up with very nice JavaScript editing tools. Some of them are listed here −

* **Microsoft FrontPage :** Microsoft has developed a popular HTML editor called FrontPage. FrontPage also provides web developers with a number of JavaScript tools to assist in the creation of interactive websites.
* **Macromedia Dreamweaver MX** **:** Macromedia Dreamweaver MX is a very popular HTML and JavaScript editor in the professional web development crowd. It provides several handy prebuilt JavaScript components, integrates well with databases, and conforms to new standards such as XHTML and XML.
* **Macromedia HomeSite 5 :** HomeSite 5 is a well-liked HTML and JavaScript editor from Macromedia that can be used to manage personal websites effectively.

**5.3 Database**

## Database

  Database is a separate application that stores a collection of data. Each database has one or more distinct APIs for creating, accessing, managing, searching and replicating the data it holds.

Other kinds of data stores can also be used, such as files on the file system or large hash tables in memory but data fetching and writing would not be so fast and easy with those type of systems.

Nowadays, we use relational database management systems (RDBMS) to store and manage huge volume of data. This is called relational database because all the data is stored into different tables and relations are established using primary keys or other keys known as Foreign Keys.

A Relational DataBase Management System (RDBMS) is a software that −

* Enables you to implement a database with tables, columns and indexes.
* Guarantees the Referential Integrity between rows of various tables.
* Updates the indexes automatically.
* Interprets an SQL query and combines information from various tables.

## RDBMS Terminology

Before we proceed to explain the MySQL database system, let us revise a few definitions related to the database.

* **Database :** A database is a collection of tables, with related data.
* **Table :** A table is a matrix with data. A table in a database looks like a simple spreadsheet.
* **Column :** One column (data element) contains data of one and the same kind, for example the column postcode.
* **Row :** A row (= tuple, entry or record) is a group of related data, for example the data of one subscription.
* **Redundancy :** Storing data twice, redundantly to make the system faster.
* **Primary Key :** A primary key is unique. A key value can not occur twice in one table. With a key, you can only find one row.
* **Foreign Key :** A foreign key is the linking pin between two tables.
* **Compound Key :** A compound key (composite key) is a key that consists of multiple columns, because one column is not sufficiently unique.
* **Index :** An index in a database resembles an index at the back of a book.
* **Referential Integrity :** Referential Integrity makes sure that a foreign key value always points to an existing row.

**5.3.1 MySQL**

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons

* MySQL is released under an open-source license. So you have nothing to pay to use it.
* MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
* MySQL uses a standard form of the well-known SQL data language.
* MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.
* MySQL works very quickly and works well even with large data sets.
* MySQL is very friendly to PHP, the most appreciated language for web development.
* MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).
* MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

**CHAPTER : 5**

**EXPERIMENTAL ANALYSIS AND RESULT**

**5.1 System configuration**

This project can run on commodity hardware. We ran entire project on an Intel I5 processor with 8 GB Ram, 2 GB Nvidia Graphic Processor, It also has 2 cores which runs at 1.7 GHz, 2.1 GHz respectively. First part of the is training phase which takes 10-15 mins of time and the second part is testing part which only takes few seconds to make predictions and calculate accuracy.

**5.1.1 Hardware Requirements**

* RAM: 4 GB
* Storage: 500 GB
* CPU: 2 GHz or faster
* Architecture: 32-bit or 64-bit

**5.1.2 Software requirements**

* Python 3.5 in Google Colab is used for data pre-processing, model training and prediction.
* Operating System: windows 7 and above or Linux based OS or MAC OS.
* Anaconda Navigator.
* Jupyter Notebook.
* Spyder.
* Vs Code.
* Py charm.
* Python Packages.

**5.2 Sample Code and Results**

**5.2.1 ML Modules :**

**CODE :**

## #Import required libraries

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

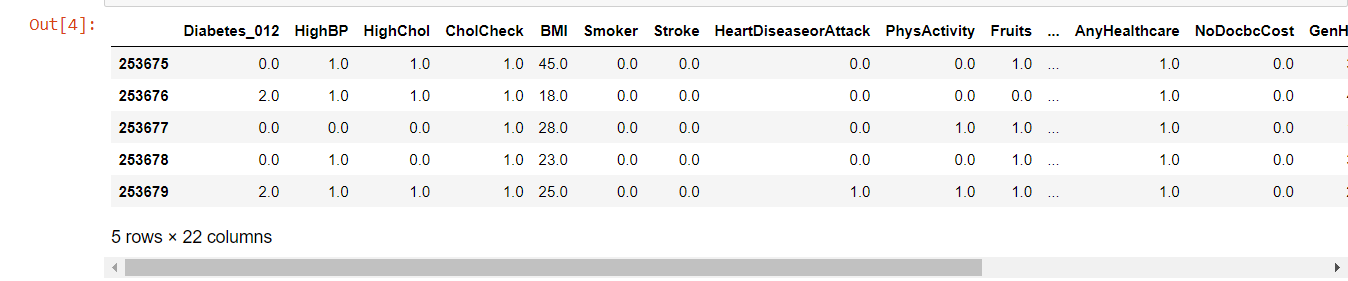
## #Reading the dataset

## df = pd.read\_csv('data')

## #Analyse the data

## df.head()

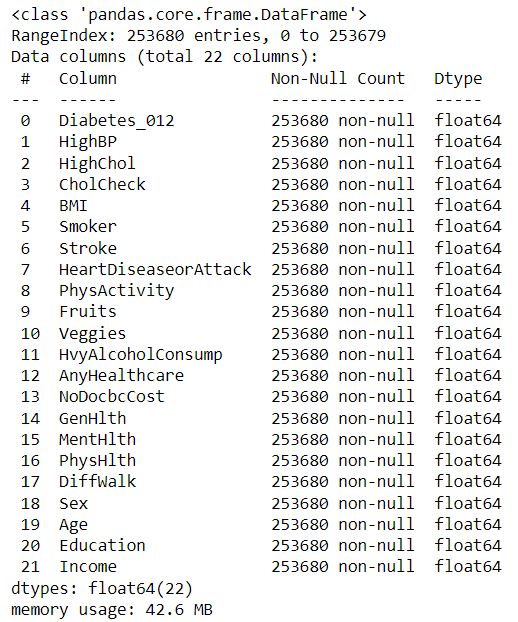
df.tail()

****

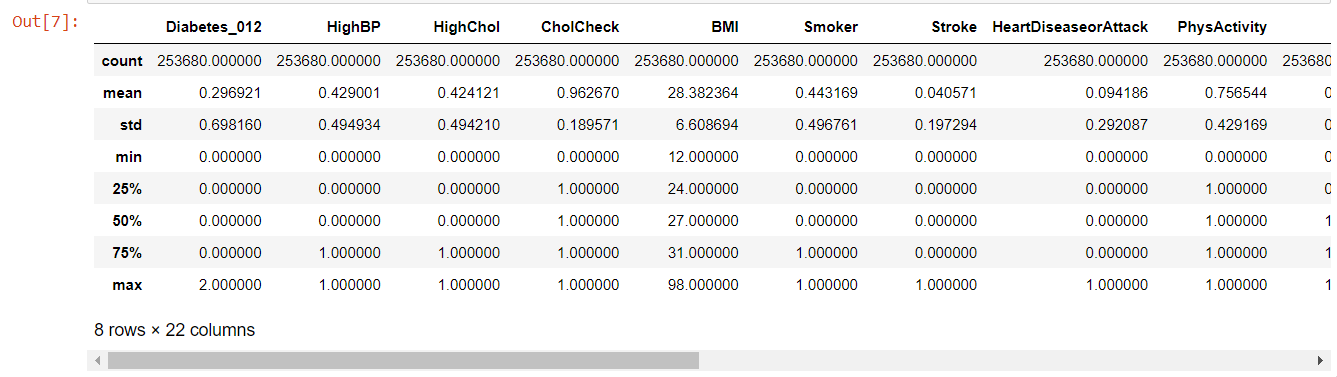
df.shape



df.info()



df.describe()



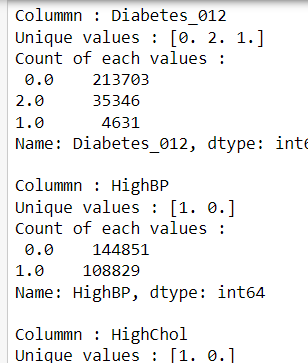
for i in df.columns:

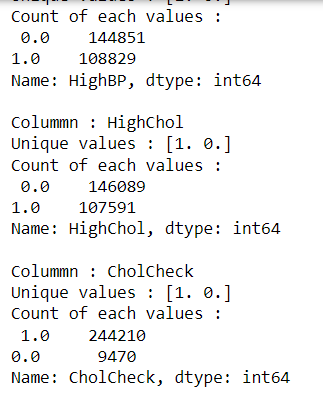
print("Colummn :", i)

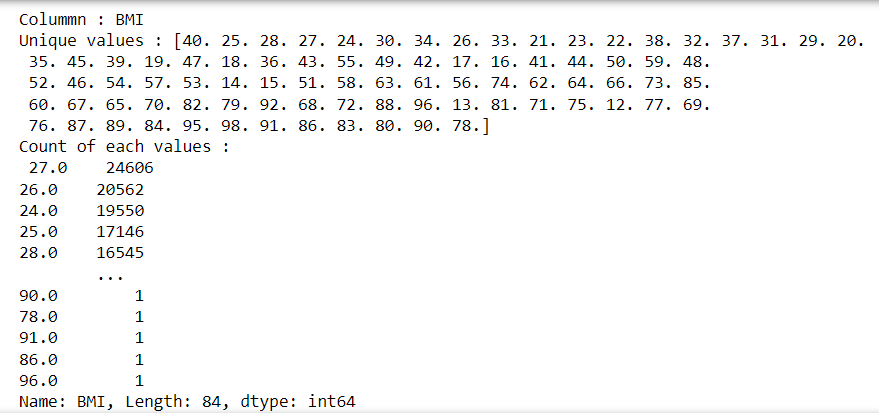
print("Unique values :", df[i].unique())

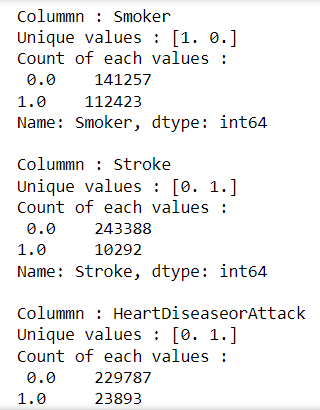
print("Count of each values :\n", df[i].value\_counts())

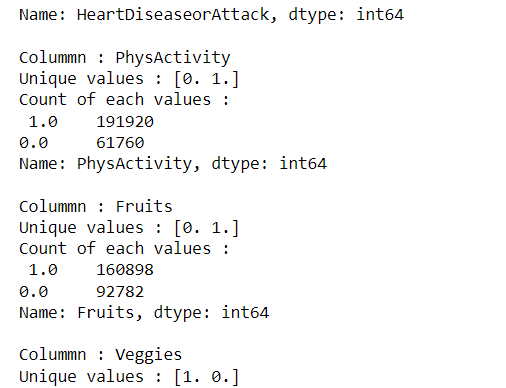
print()

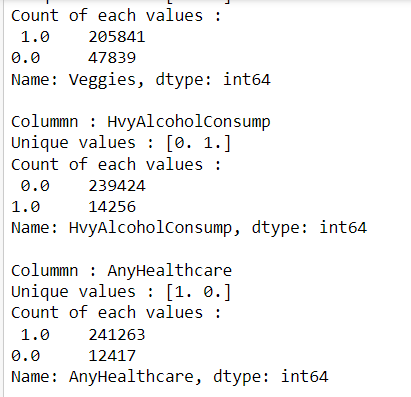


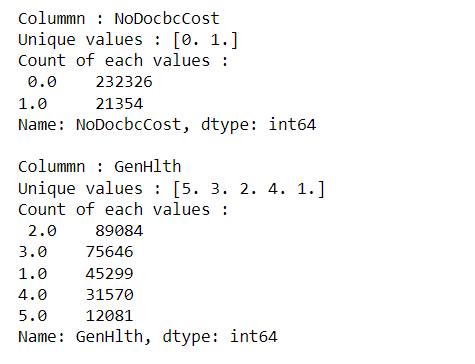


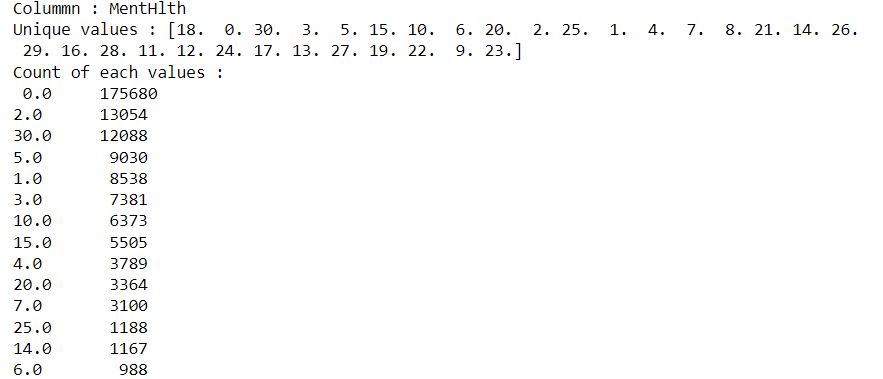


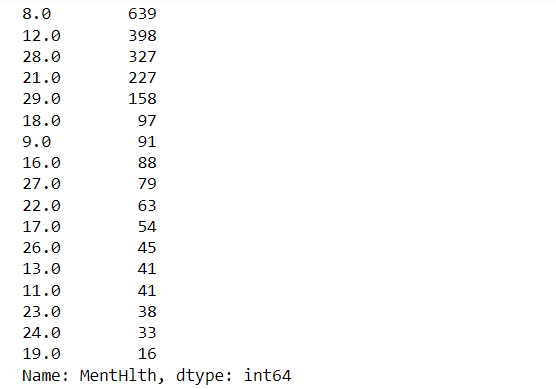


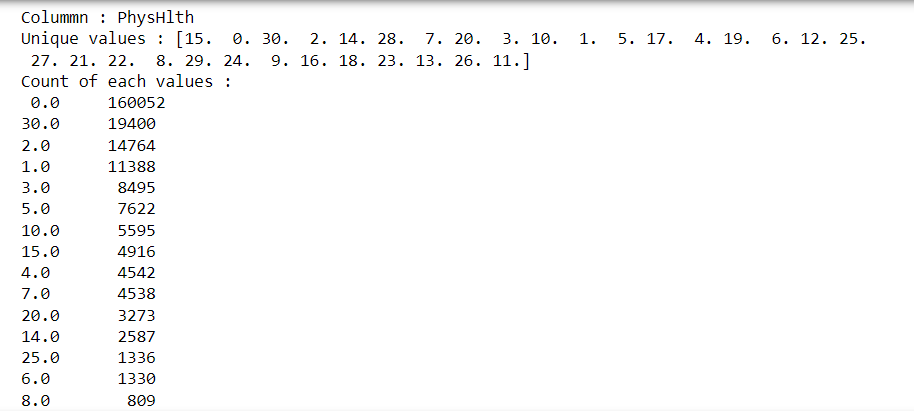


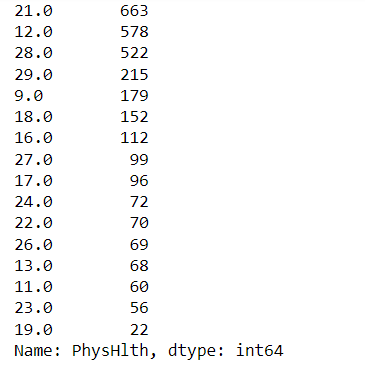


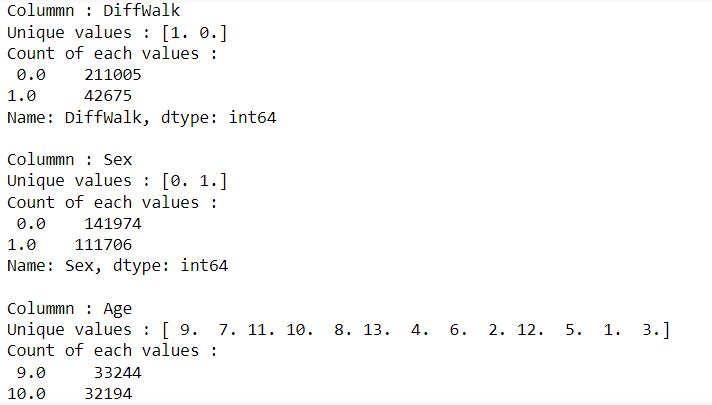


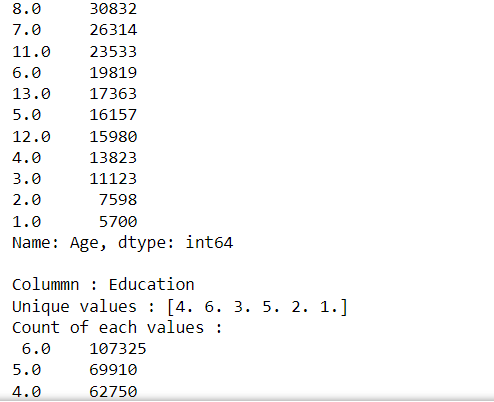
****

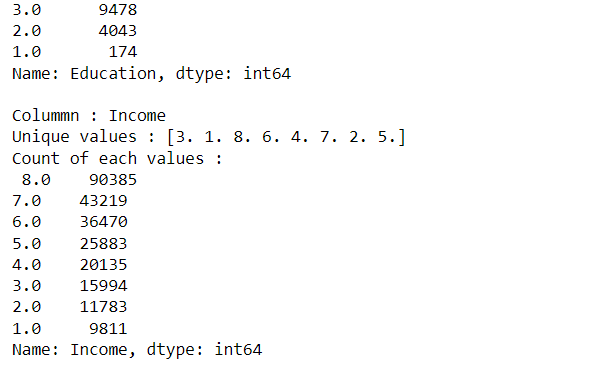
****

****

****

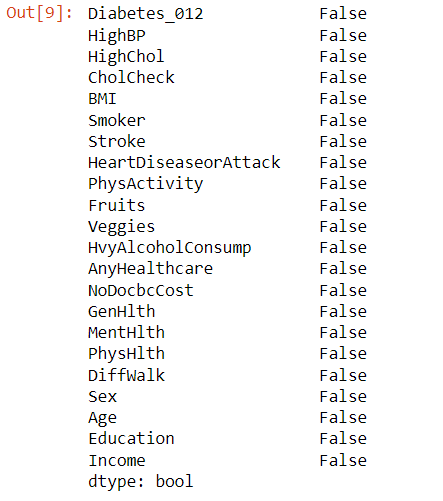




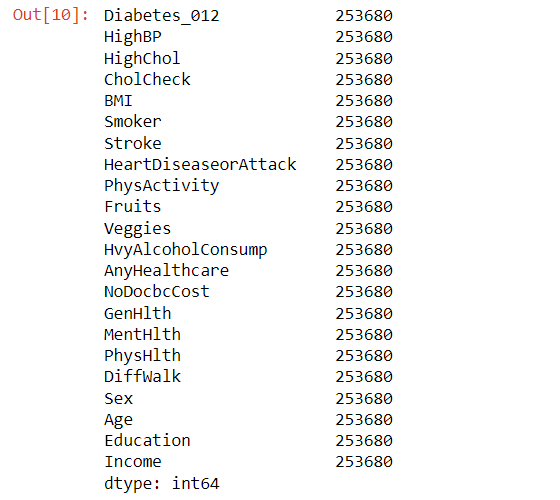


#Handling the missing values

df.isnull().any()



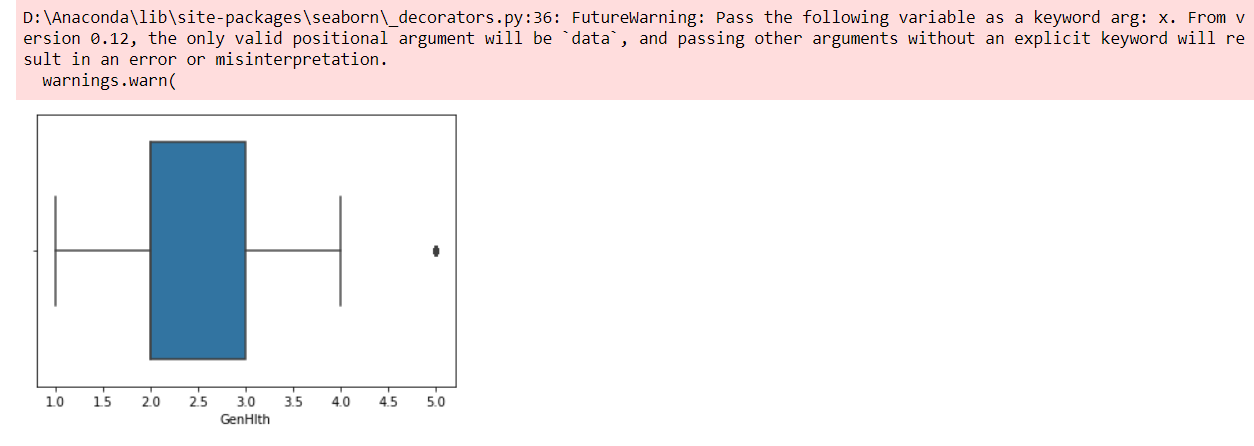
df.isnull().count()



## # Find and Replace the Outliers

sns.boxplot(df['GenHlth'])

plt.show()

****

q1 = df['GenHlth'].quantile(0.25)

q3 = df['GenHlth'].quantile(0.75)

IQR = q3 - q1

upper\_limit = q3 + 1.5\*IQR

lower\_limit = q1 - 1.5\*IQR

print("Upper Limit :", upper\_limit,"\nLower Limit :", lower\_limit)

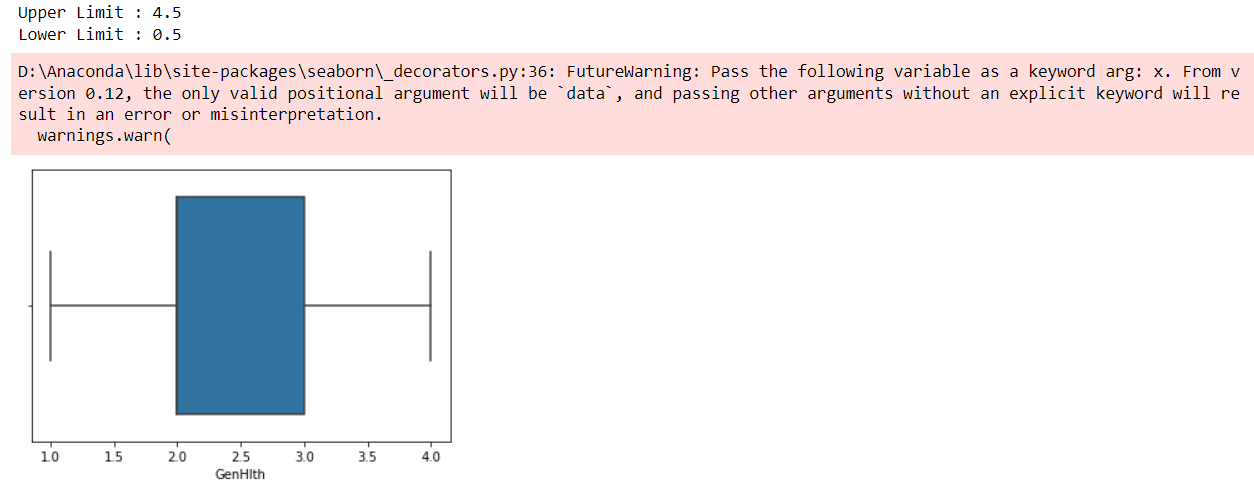
df['GenHlth'] = np.where(df['GenHlth']>upper\_limit, df['GenHlth'].median(), df['GenHlth'])

sns.boxplot(df['GenHlth'])

plt.show()

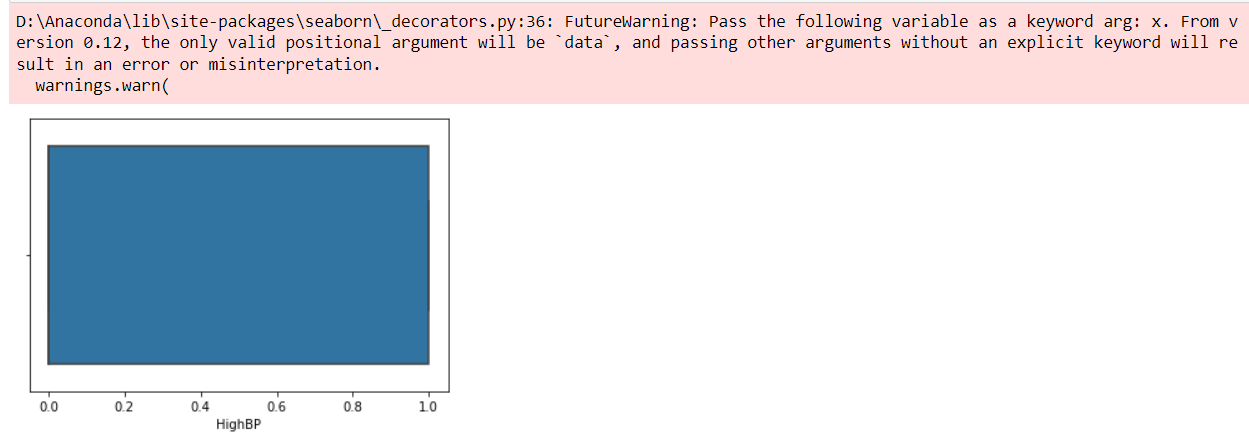
sns.boxplot(df['HighBP'])

plt.show()



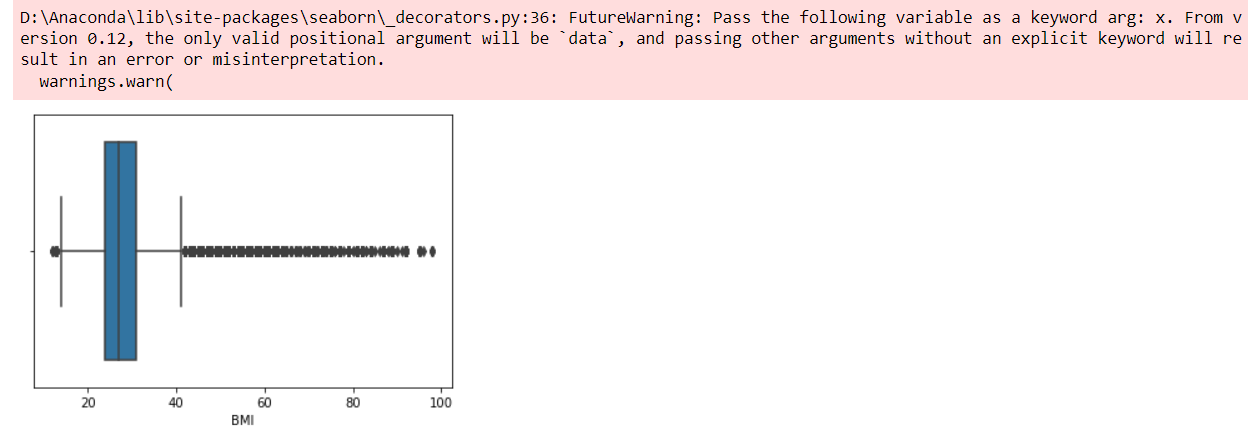
sns.boxplot(df['HighBP'])

plt.show()



sns.boxplot(df['BMI'])

plt.show()



q1 = df['BMI'].quantile(0.25)

q3 = df['BMI'].quantile(0.75)

IQR = q3 - q1

upper\_limit = q3 + 1.5\*IQR

lower\_limit = q1 - 1.5\*IQR

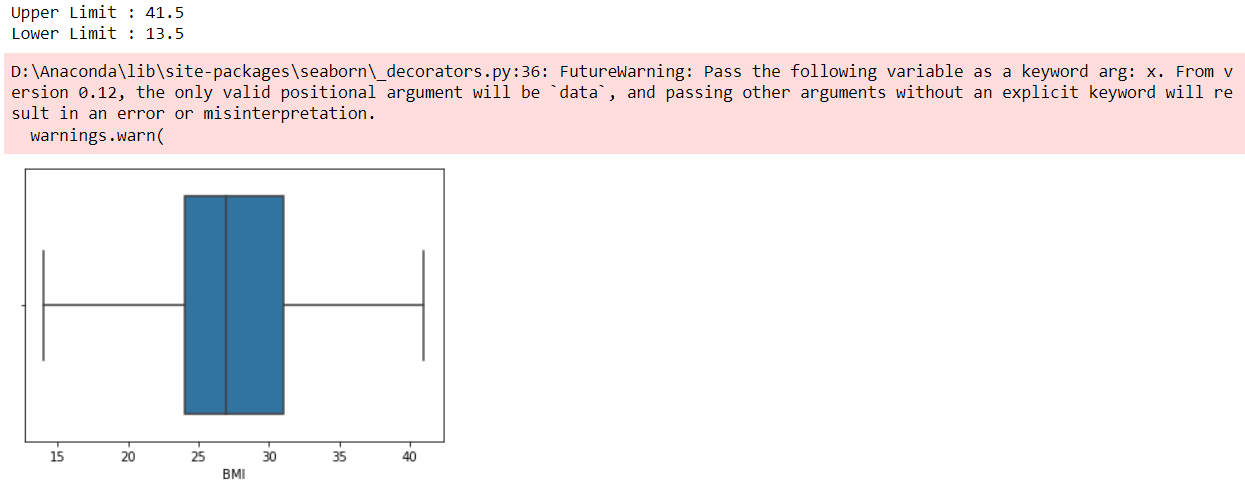
print("Upper Limit :", upper\_limit,"\nLower Limit :", lower\_limit)

df['BMI'] = np.where(df['BMI']<lower\_limit, df['BMI'].median(), df['BMI'])

df['BMI'] = np.where(df['BMI']>upper\_limit, df['BMI'].median(), df['BMI'])

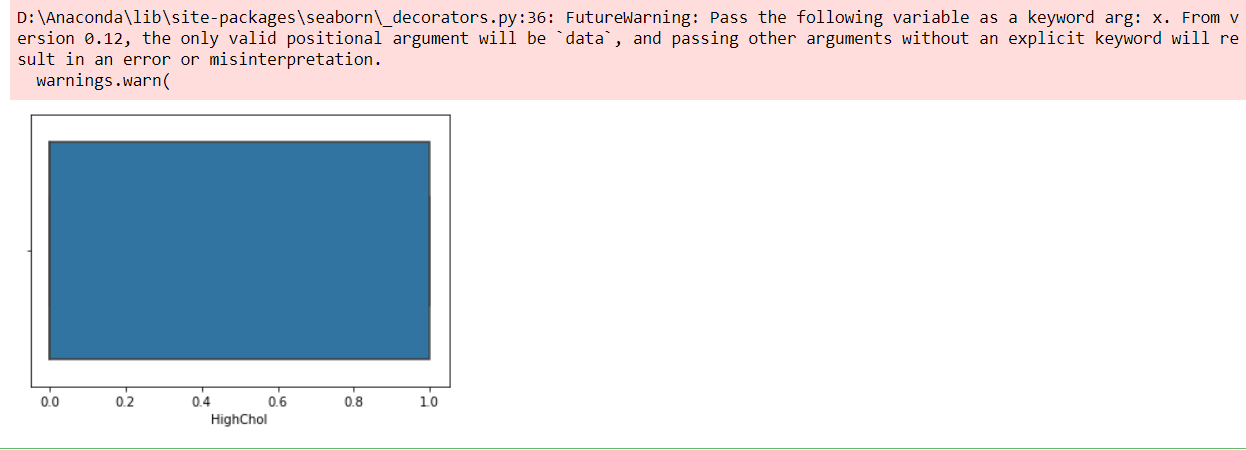
sns.boxplot(df['BMI'])

plt.show()

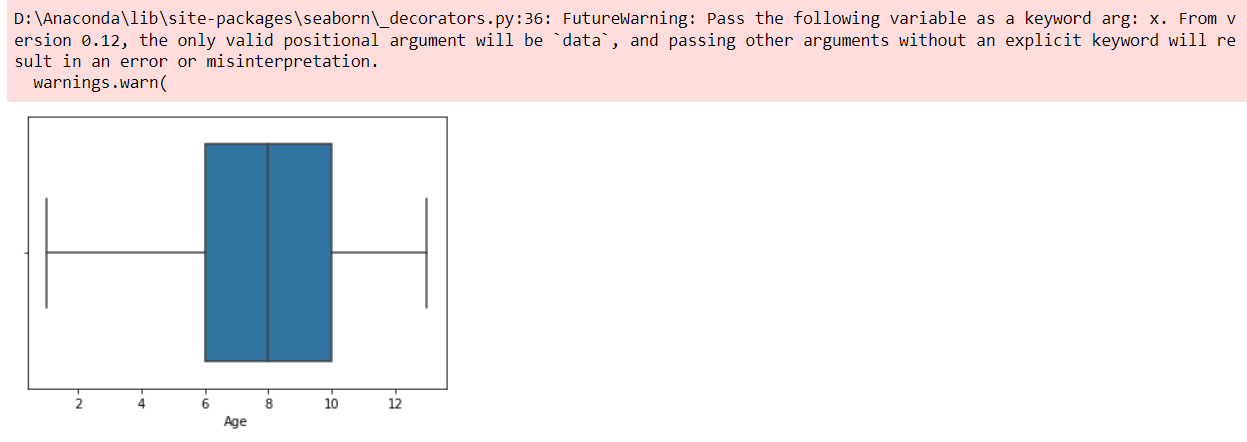


sns.boxplot(df['HighChol'])

plt.show()

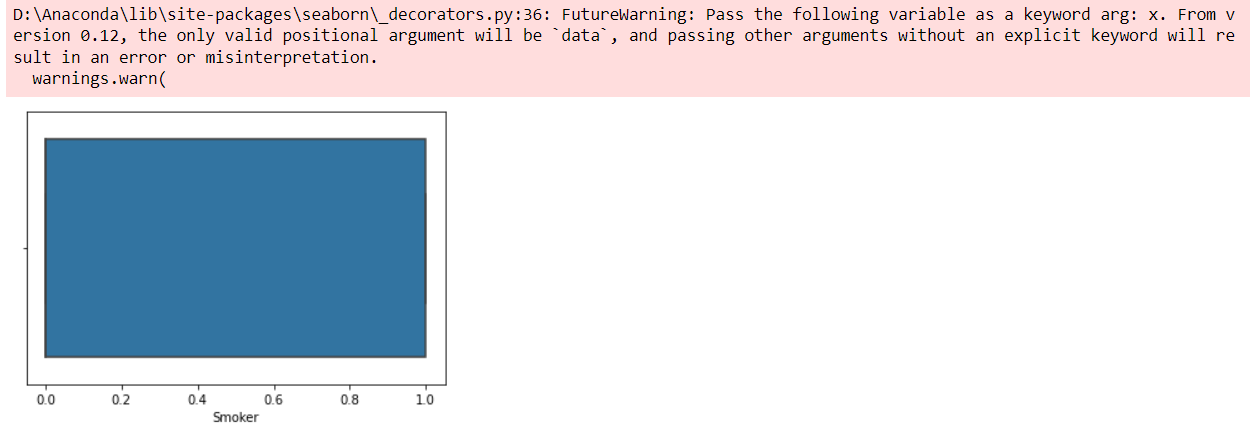


sns.boxplot(df['Age'])

plt.show()

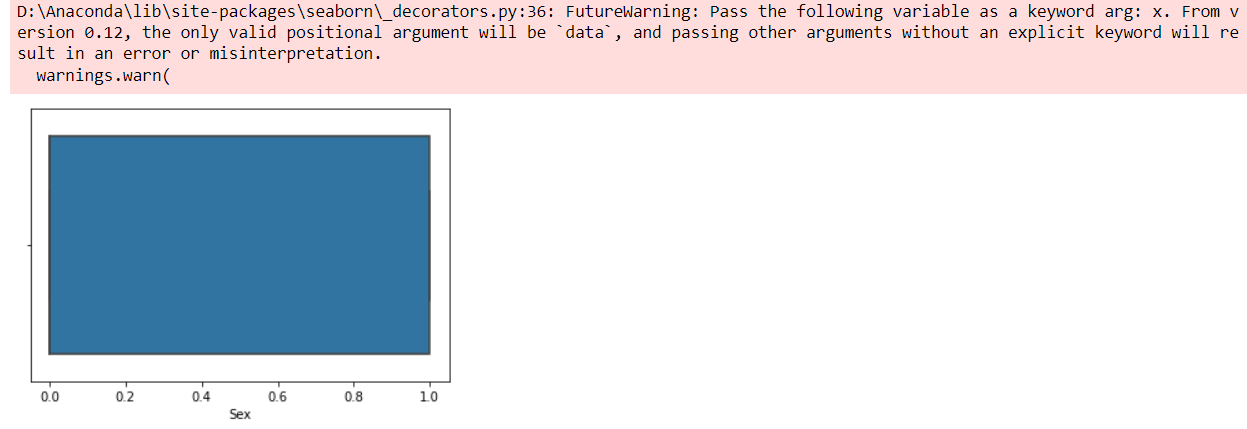
sns.boxplot(df['Smoker'])

plt.show()



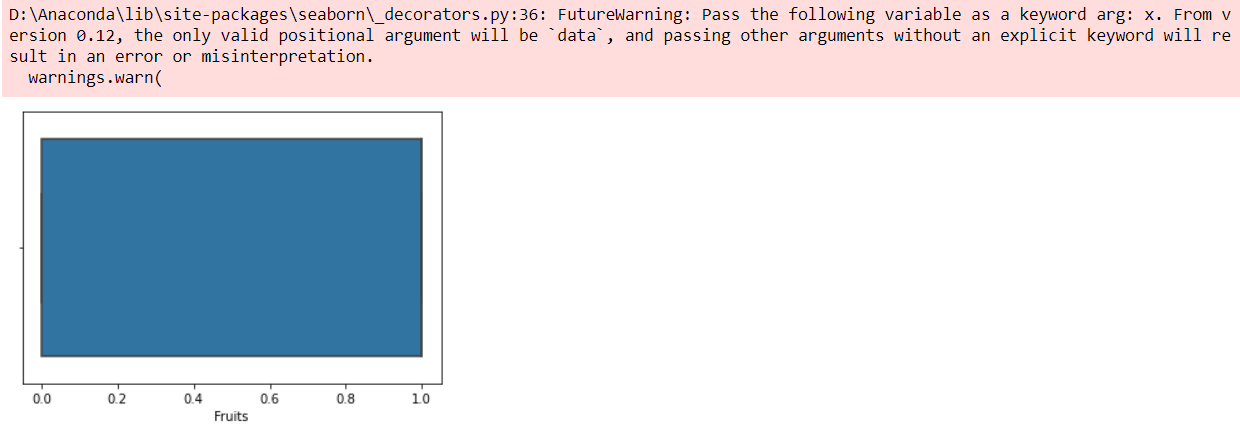
sns.boxplot(df['Sex'])

plt.show()



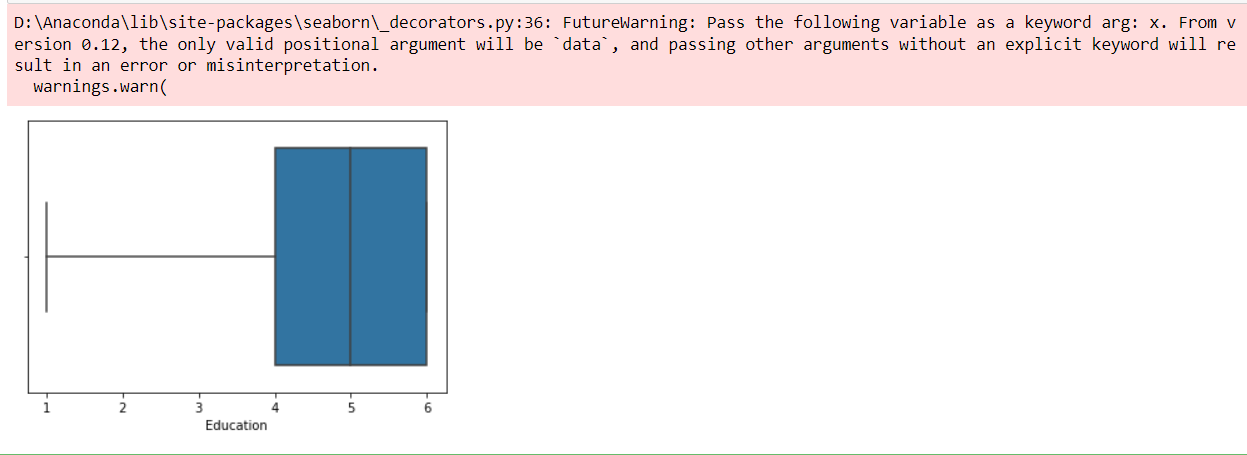
sns.boxplot(df['Fruits'])

plt.show()



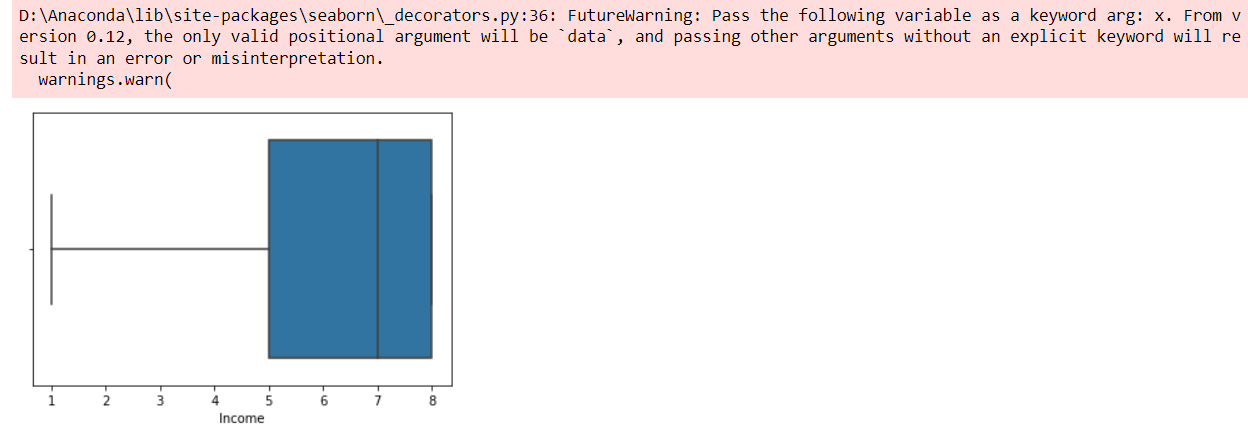
sns.boxplot(df['Education'])

plt.show()



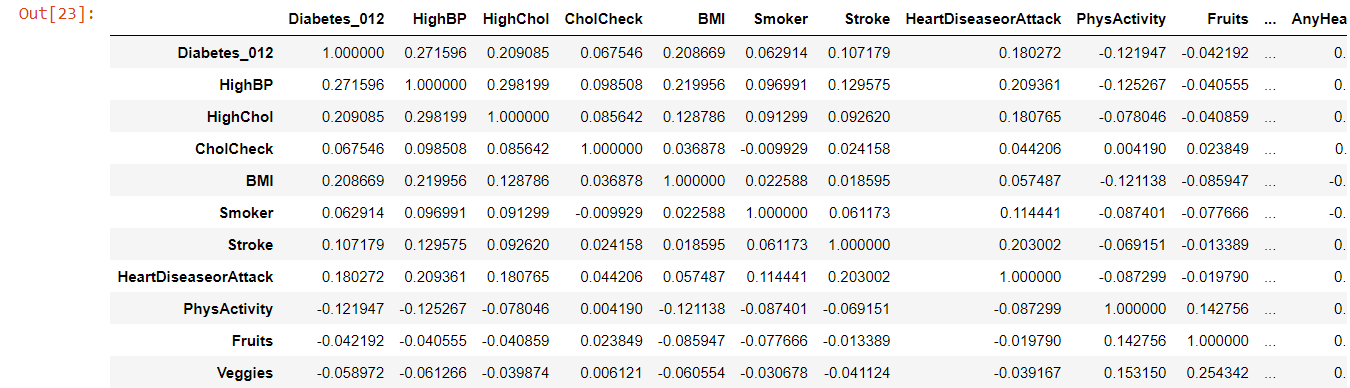
sns.boxplot(df['Income'])

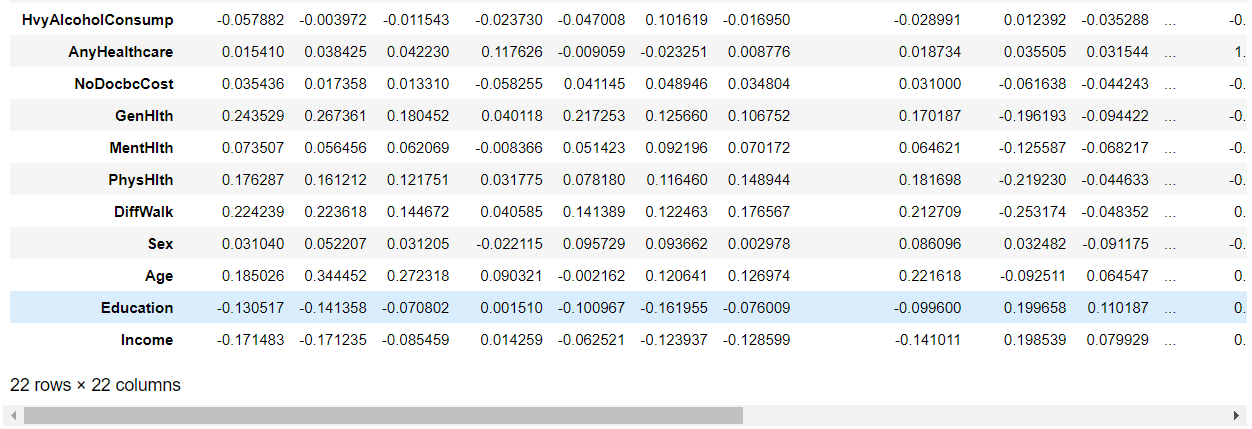
plt.show()



#Find the correlation

df.corr()

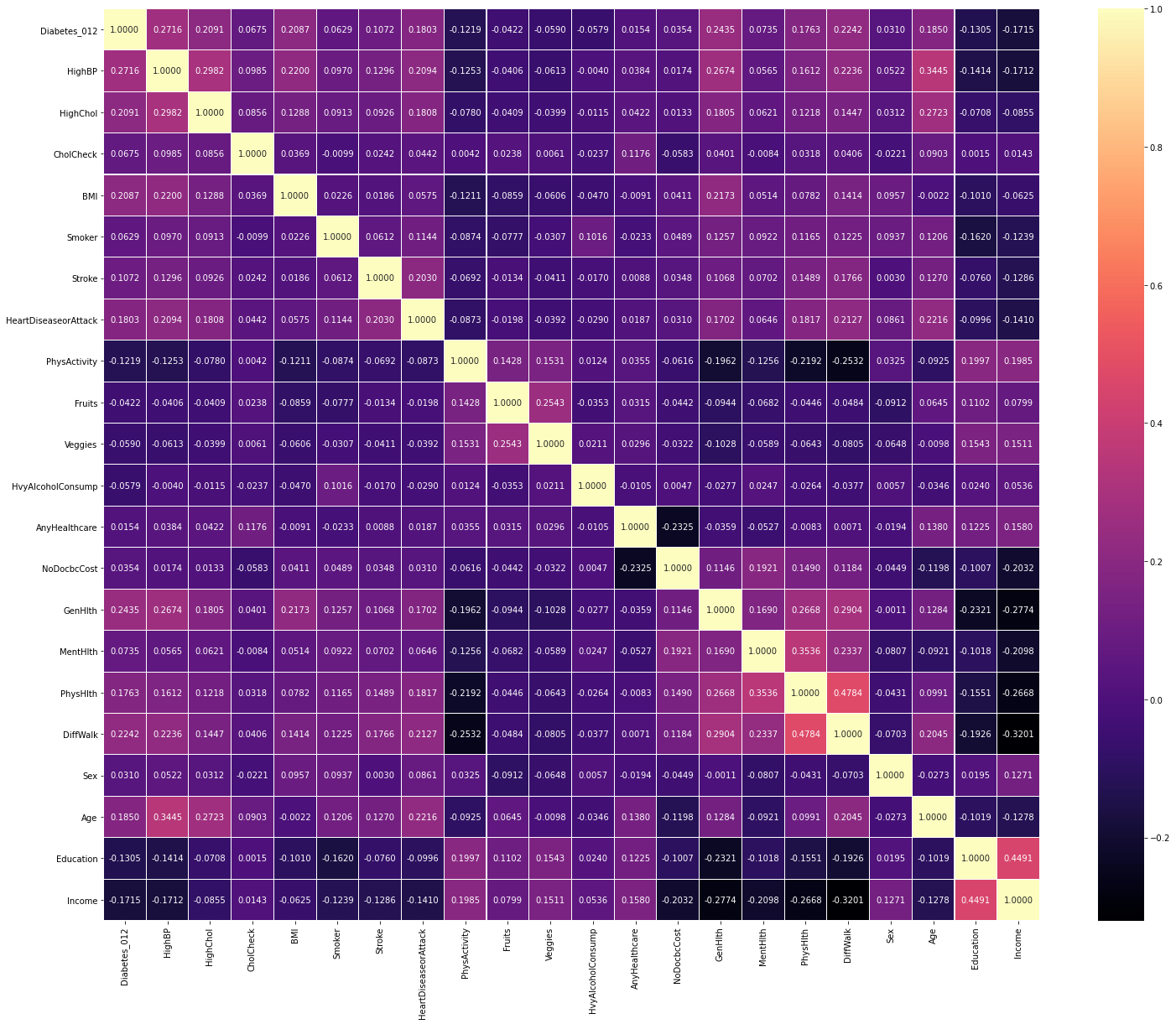




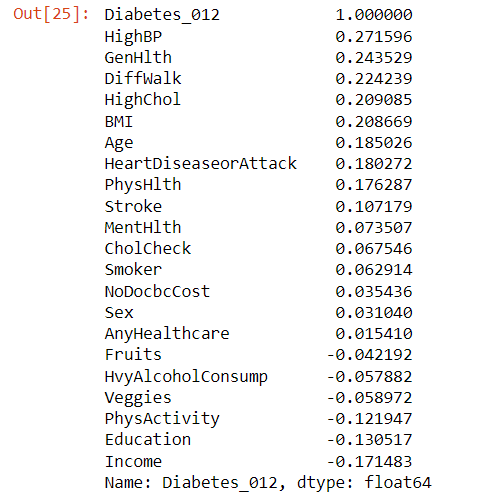
plt.figure(figsize=(25, 20))

sns.heatmap(df.corr(), annot=True, linewidths=0.08, fmt= '.4f',cmap="magma")

plt.show()



df.corr()['Diabetes\_012'].sort\_values(ascending=False)



## #Splitting dependent and independent columns

#independent variable

X = df.drop(['Diabetes\_012'], axis=1)

#dependent variable

y = df['Diabetes\_012']

#shape of independent and dependent variable

print("Shape of independent variable :", X.shape)

print("Shape of dependent variable :", y.shape)

## 

# Scale the independent variable

from sklearn.preprocessing import StandardScaler

ss = StandardScaler()

X = pd.DataFrame(ss.fit\_transform(X), columns=X.columns)

X.head()

#Find the balance and unbalanced data

dv = pd.value\_counts(df['Diabetes\_012'], sort=True)

dv.plot(kind='bar', rot=0)

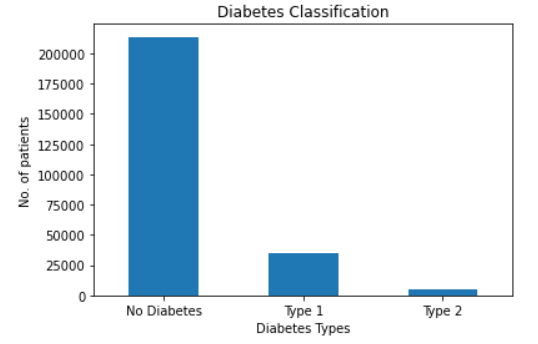
plt.title('Diabetes Classification')

plt.xticks(range(3), ["No Diabetes", "Type 1", "Type 2"])

plt.xlabel('Diabetes Types')

plt.ylabel('No. of patients')

plt.show()



#under sampling is used to balance the unbalanced data

from imblearn.under\_sampling import NearMiss

nm = NearMiss()

X\_res, y\_res = nm.fit\_resample(X, y)

from collections import Counter

print("Data before balancing :", Counter(y))

print("Data after balancing :", Counter(y\_res))



#Splitting the data into train and test

from sklearn.model\_selection import train\_test\_split

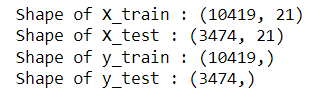
X\_train, X\_test, y\_train, y\_test = train\_test\_split(X\_res, y\_res, test\_size = 0.25, random\_state = 0)

print("Shape of X\_train :", X\_train.shape)

print("Shape of X\_test :", X\_test.shape)

print("Shape of y\_train :", y\_train.shape)

print("Shape of y\_test :", y\_test.shape)



#Training and Testing Model

from sklearn.linear\_model import LogisticRegression

from sklearn.tree import DecisionTreeClassifier

from sklearn.ensemble import RandomForestClassifier, GradientBoostingClassifier, AdaBoostClassifier

from sklearn.neighbors import KNeighborsClassifier

from sklearn.naive\_bayes import GaussianNB

from sklearn.svm import SVC

from xgboost import XGBClassifier

from sklearn.metrics import accuracy\_score

def test(models, X\_res, y\_res, iterations = 100):

results = {}

for i in models:

r2\_train = []

r2\_test = []

for j in range(iterations):

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X\_res,

y\_res,

test\_size= 0.25)

r2\_test.append(accuracy\_score(y\_test,

models[i].fit(X\_train,

y\_train).predict(X\_test)))

r2\_train.append(accuracy\_score(y\_train,

models[i].fit(X\_train,

y\_train).predict(X\_train)))

results[i] = [np.mean(r2\_train), np.mean(r2\_test)]

return pd.DataFrame(results, index=['Training Results', 'Testing Results'])

models = {'Logistic': LogisticRegression(max\_iter=200),

'Descision Tree': DecisionTreeClassifier(),

'Random Forest': RandomForestClassifier(),

'Gradient Boost': GradientBoostingClassifier(),

'Ada Boost': AdaBoostClassifier(),

'Neighbors': KNeighborsClassifier(),

'Naive Bayes': GaussianNB(),

'SVM': SVC(),

'Xgboost': XGBClassifier()}

test(models, X\_res, y\_res)

#Hyperparameter Tuning

def evaluate(models):

results = {}

for i, j in models.items():

results[i] = [j.best\_params\_, j.best\_estimator\_, j.best\_score\_]

return pd.DataFrame(results, index=['Best Parameter', 'Best Estimator', 'Best Score'])

#Using RandomizedSearchCV

from sklearn.model\_selection import RandomizedSearchCV

rf\_params ={'max\_depth':[3,5,10,None],

'n\_estimators':[100,200,300,400,500],

'max\_features':['sqrt', 'log2', None],

'criterion':['gini', 'entropy', 'log\_loss'],

'bootstrap':[True, False],

'min\_samples\_leaf':[1,2,4],

'min\_samples\_split':[2,5,10]}

dt\_params ={'max\_depth':[3,5,10,None],

'splitter':['best', 'random'],

'max\_features':['sqrt', 'log2', None],

'criterion':['gini', 'entropy', 'log\_loss'],

'min\_samples\_leaf':[1,2,4],

'min\_samples\_split':[2,5,10]}

gb\_params ={'max\_depth':[3,5,10,None],

'n\_estimators':[100,200,300,400,500],

'max\_features':['sqrt', 'log2', None],

'criterion':['friedman\_mse', 'squared\_error'],

'loss':['log\_loss', 'exponential'],

'min\_samples\_leaf':[1,2,4],

'min\_samples\_split':[2,5,10]}

ab\_params ={'algorithm':['SAMME', 'SAMME.R'],

'n\_estimators':[100,200,300,400,500]}

lg\_params ={'penalty':['l1', 'l2', 'elasticnet', 'none'],

'solver':['newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga'],

'multi\_class':['auto', 'ovr', 'multinomial'],

'C':np.logspace(-5, 8, 15)}

kn\_params ={'weights':['uniform', 'distance'],

'algorithm':['auto', 'ball\_tree', 'kd\_tree', 'brute'],

'n\_neighbors':list(range(1, 31))}

gn\_params ={'var\_smoothing':np.logspace(0, -9, num=100)}

xb\_params ={'learning\_rate':[0.05, 0.10, 0.15, 0.20, 0.25, 0.30],

#'max\_depth':[3,5,10,None],

#'min\_child\_weight':[1,3,5,7],

#'gamma':[0.0,0.1,0.2,0.3,0.4],

#'colsample\_bytree':[0.3,0.4,0.5,0.7],

'n\_estimators':[100,200,300,400,500],

'reg\_alpha':[0.5,0.2,1],

'reg\_lambda':[2,3,5],

'booster':['gbtree', 'gblinear']}

sv\_params ={'C':np.logspace(-5, 8, 15),

'kernel':['linear', 'poly', 'rbf', 'sigmoid', 'precomputed'],

'gamma':['scale', 'auto'],

}

rfrs = RandomizedSearchCV(estimator = RandomForestClassifier(), param\_distributions = rf\_params, cv = 3, n\_iter = 5)

dtrs = RandomizedSearchCV(estimator = DecisionTreeClassifier(), param\_distributions = dt\_params, cv = 3, n\_iter = 5)

gbrs = RandomizedSearchCV(estimator = GradientBoostingClassifier(), param\_distributions = gb\_params, cv = 3, n\_iter = 5)

abrs = RandomizedSearchCV(estimator = AdaBoostClassifier(), param\_distributions = ab\_params, cv = 3, n\_iter = 5)

lgrs = RandomizedSearchCV(estimator = LogisticRegression(), param\_distributions = lg\_params, cv = 3, n\_iter = 5)

knrs = RandomizedSearchCV(estimator = KNeighborsClassifier(), param\_distributions = kn\_params, cv = 3, n\_iter = 5)

gnrs = RandomizedSearchCV(estimator = GaussianNB(), param\_distributions = gn\_params, cv = 3, n\_iter = 5)

xbrs = RandomizedSearchCV(estimator = XGBClassifier(), param\_distributions = xb\_params, cv = 3, n\_iter = 5)

svrs = RandomizedSearchCV(estimator = SVC(), param\_distributions = sv\_params, cv = 3, n\_iter = 5)

models2 = {'Random Forest': rfrs.fit(X\_res, y\_res).best\_estimator\_,

'Decision Tree': dtrs.fit(X\_res, y\_res).best\_estimator\_,

'Gradient Boost': gbrs.fit(X\_res, y\_res).best\_estimator\_,

'Ada Boost': abrs.fit(X\_res, y\_res).best\_estimator\_,

'Logistic Regression': lgrs.fit(X\_res, y\_res).best\_estimator\_,

'Neighbors': knrs.fit(X\_res, y\_res).best\_estimator\_,

'Naive Bayes': gnrs.fit(X\_res, y\_res).best\_estimator\_,

'XGBoost': xbrs.fit(X\_res, y\_res).best\_estimator\_,

'SVM': svrs.fit(X\_res, y\_res).best\_estimator\_

}

test(models2, X\_res, y\_res)

cv = {

'Logistic': lgrs,

'Descision Tree': dtrs,

'Random Forest': rfrs,

'Gradient Boost': gbrs,

'Ada Boost': abrs,

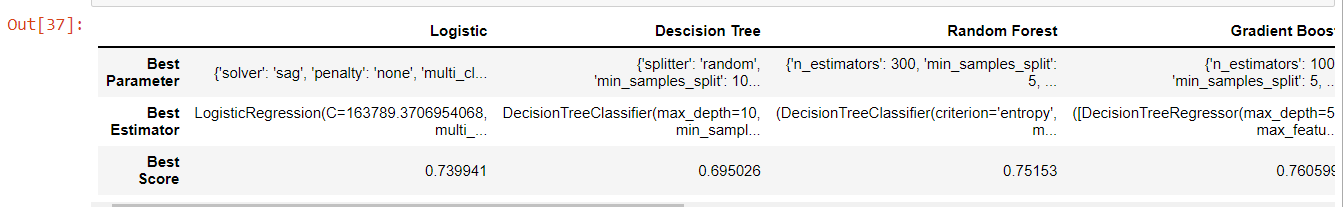
'Neighbors': knrs,

'Naive Bayes': gnrs,

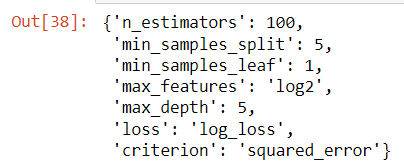
'SVM': svrs,

'Xgboost': xbrs

}

evaluate(cv)

gbrs.best\_params\_



## #Build, Train and Test the Best Model

# was selected because it performs well by compared with others, those results are seen above cells.

gb\_cv = GradientBoostingClassifier(n\_estimators = 100,

min\_samples\_split = 5,

min\_samples\_leaf = 1,

max\_features = 'log2',

max\_depth = 5,

loss = 'log\_loss',

criterion = 'squared\_error')

gb\_cv.fit(X\_train, y\_train)

y\_pred\_train = gb\_cv.predict(X\_train)

y\_pred = gb\_cv.predict(X\_test)

## #Model Evaluation

## print(pd.DataFrame({'Actual': y\_test,'Predicted':y\_pred}))

## 

print("Training accuracy :", accuracy\_score(y\_train, y\_pred\_train))

print("Testing accuracy :", accuracy\_score(y\_test, y\_pred))

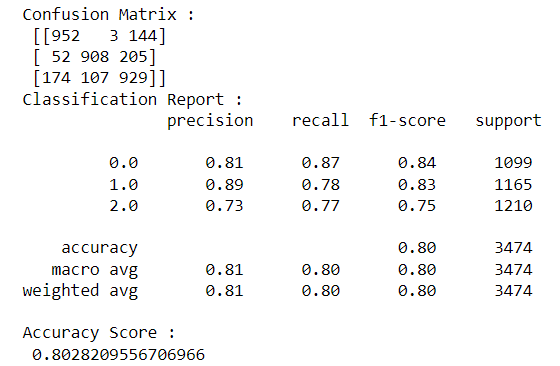


from sklearn.metrics import accuracy\_score, classification\_report, confusion\_matrix

print("Confusion Matrix : \n", confusion\_matrix(y\_test,y\_pred))

print("Classification Report : \n", classification\_report(y\_test,y\_pred))

print("Accuracy Score : \n", accuracy\_score(y\_test, y\_pred))



#Save the model

import pickle

with open('uaep\_model.pkl', 'wb') as files:

pickle.dump(ridge\_cv, files)

**6.3** **Web Interface Code and Result**

#Contact.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=">

    <link rel="stylesheet" href="static/css/contact.css">

    <title>Contact Us </title>

    <script src="https://kit.fontawesome.com/c32adfdcda.js" crossorigin="anonymous"></script>

</head>

<body>

    <section>

        <div class="section-header">

            <div class="container">

                <h2>Contact Us</h2>

                <p>Have questions ?

                    Shoot us an Email.<br>

                    We're here to help and answer any question you might have.We look forwward to hearing from you <br>

                    Whether you have a question about feature, trials, pricing, need a demo or anything else , our team

                    is ready to answer all your questions</p>

            </div>

        </div>

        <div class="container">

            <div class="row">

                <div class="contact-info">

                    <div class="contact-info-item">

                        <div class="contact-info-icon">

                            <i class="fas fa-home"></i>

                        </div>

                        <div class="contact-info-content">

                            <h4>Address</h4>

                            <p>06/36 Narasipuram,<br /> Coimbatore, Tamil Nadu,<br>India <br />641109</p>

                        </div>

                    </div>

                    <div class="contact-info-item">

                        <div class="contact-info-icon">

                            <i class="fas fa-phone"></i>

                        </div>

                        <div class="contact-info-content">

                            <h4>Phone</h4>

                            <p>0422-2970703</p>

                        </div>

                    </div>

                    <div class="contact-info-item">

                        <div class="contact-info-icon">

                            <i class="fas fa-envelope"></i>

                        </div>

                        <div class="contact-info-content">

                            <h4>Email</h4>

                            <p>info@cietcbe.edu.in</p>

                        </div>

                    </div>

                </div>

                <div class="contact-form">

                    <form action="" id="contact-form">

                        <h2>Send Message</h2>

                        <div class="input-box">

                            <input type="text" required="true" name="">

                            <span>Full Name</span>

                        </div>

                        <div class="input-box">

                            <input type="email" required="true" name="">

                            <span>Email</span>

                        </div>

                        <div class="input-box">

                            <textarea required="true" name=""></textarea>

                            <span>Type your Message...</span>

                        </div>

                        <div class="input-box">

                            <input type="submit" value="Send" name="">

                        </div>

                    </form>

                </div>

            </div>

        </div>

    </section>

</body>

</html>

#Form.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="static/css/form.css">

     <!-- font awesome cdn link  -->

     <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css">

     <!-- custom css file link  -->

     <link rel="stylesheet" href="static/css/home.css">

    <title>Document</title>

</head>

<body>

    <header class="header1">

        <a href="#" class="logo"> <i class="fas fa-heartbeat"></i> medcare. </a>

        <nav class="navbar">

            <a href="{{url\_for('home')}}">home</a>

            <a href="{{url\_for('contact')}}">Contact</a>

        </nav>

        <div id="menu-btn" class="fas fa-bars"></div>

    </header>

    <section>

        <div class="container">

            <div>

                <h3>You can't predict the result without filling all the input or without a valid information</h3>

            </div>

          <form action="{{url\_for('predict')}}" method="post">

            <div class="step step-1 active">

              <div class="form-group">

                <label for="firstName">Enter your Name :</label>

                <input type="text" id="firstName" name="name" required>

              </div>

              <div class="form-group">

                <label for="age">Enter your Age : </label>

                            <select name="age" id="age">

                                <option value="1">1 to 24</option>

                                <option value="2">25 to 29</option>

                                <option value="3">30 to 34</option>

                                <option value="4">35 to 39</option>

                                <option value="5">40 to 44</option>

                                <option value="6">45 to 49</option>

                                <option value="7">50 to 54</option>

                                <option value="8">55 to 59</option>

                                <option value="9">60 to 64</option>

                                <option value="10">65 to 69</option>

                                <option value="11">70 to 74</option>

                                <option value="12">75 to 79</option>

                                <option value="13">80 or older</option>

                            </select required>

              </div>

              <div class="form-group">

                <label for="gender">Enter your Gender</label>

                <select name="gender" id="gender" required>

                    <option value="1">Male</option>

                    <option value="0">Female</option>

                </select required>

              </div>

              <div class="form-group">

                 <label for="bmi">What is your  Body Mass Intex(BMI):<br><b>Formula : weight (kg) / [height (m)]2 </b> </label>

                            <input type="text" id="bmi" name="bmi" required>

              </div>

              <div class="form-group">

                <label for="mhealth">In past 30 days,how many days did you had a mental health

                    problem<br>[<b>Note:</b>It includes stress, depression, and problems with emotions ]

                </label>

                <input type="number" min="0" max="30" required name="mhealth">

              </div>

              <div class="form-group">

                <label for="phealth">In past 30 days,how many days did you had a physical health

                    problem<br>[<b>Note:</b>It includes physical illness and injury]<br>

                </label>

                <input type="number" min="0" max="30" required name="phealth">

              </div>

              <div class="form-group">

                <label for="bp">Do you have a high BP ( Blood Pressure ) ?</label>

                <select name="bp" id="bp" required>

                    <option value="1">Yes</option>

                    <option value="0">No</option>

                </select>

              </div>

              <button type="button" class="next-btn">Next</button>

            </div>

            <!-- second panel -->

            <div class="step step-2">

              <div class="form-group">

                <label for="hc">Do you have high Cholesterol ?</label>

                <select name="hc" id="hc" required>

                    <option value="1">Yes</option>

                    <option value="0">No</option>

                </select>

              </div>

              <div class="form-group">

                <label for="cc">Have you checked your Cholesterol in last 5 years ?</label>

                            <select name="cc" id="cc" required>

                                <option value="1">Yes</option>

                                <option value="0">No</option>

                            </select>

              </div>

              <div class="form-group">

                <label for="smoker">Have you smoked at least 100 cigarettes in your entire life ?<br>

                    [<b>Note:</b>

                    5

                    packs =

                    100 cigarettes]</label>

                <select name="smoker" id="smoker" required>

                    <option value="1">Yes</option>

                    <option value="0">No</option>

                </select>

              </div>

              <div class="form-group">

                <label for="stroke">Have you ever had an Stroke ?</label>

                            <select name="stroke" id="stroke" required>

                                <option value="1">Yes</option>

                                <option value="0">No</option>

                            </select>

              </div>

              <div class="form-group">

                <label for="hearta">Do you have any Heart Disease <b>or</b> Ever had an HeartAttack ? <br>

                    [Coronary

                    heart

                    disease

                    (CHD) <b>or</b> Myocardial infarction (MI)] </label>

                <select name="hearta" id="hearta" required>

                    <option value="1">Yes</option>

                    <option value="0">No</option>

                </select>

              </div>

              <div class="form-group">

                <label for="phact">Did you do any physical activity in 30 days ?<br>[<b>Note:</b> Do not

                    include

                    your

                    job ]

                    :</label>

                <select name="phact" id="phact" required>

                    <option value="1">Yes</option>

                    <option value="0">No</option>

                </select>

              </div>

              <div class="form-group">

                <label for="fruit">Do you consume any fruit one or more times per day ? </label>

                            <select name="fruit" id="fruit" required>

                                <option value="1">Yes</option>

                                <option value="0">No</option>

                            </select>

              </div>

              <button type="button" class="previous-btn">Prev</button>

              <button type="button" class="next-btn">Next</button>

            </div>

            <!-- third panel  -->

            <div class="step step-3">

              <div class="form-group">

                <label for="veggies">Do you consume Vegetables one or more times per day ? </label>

                            <select name="veggies" id="veggies" required>

                                <option value="1">Yes</option>

                                <option value="0">No</option>

                            </select>

              </div>

              <div class="form-group">

                <label for="drinker">Are you an Heavy Drinker ?<br>

                    [Note: <em>adult men</em> having more than 14 drinks per week <b>and</b> <em>adult

                        women</em>

                    having more than 7 drinks per week]</label>

                <select name="drinker" id="drinker" required>

                    <option value="1">Yes</option>

                    <option value="0">No</option>

                </select>

              </div>

              <div class="form-group">

                <label for="hcare">Do you have any kind of health care coverage ?<br> [<b>Note:</b>

                    including

                    health

                    insurance, prepaid plans such as HMO, etc..]</label>

                <select name="hcare" id="hcare" required>

                    <option value="1">Yes</option>

                    <option value="0">No</option>

                </select>

              </div>

              <div class="form-group">

                <label for="cost">Was there a time in the past 12 months when you needed to see a doctor but

                    could

                    not because of cost ?</label>

                <select name="cost" id="cost" required>

                    <option value="1">Yes</option>

                    <option value="0">No</option>

                </select>

              </div>

              <div class="form-group">

                <label for="walking">Do you have serious difficulty walking or climbing stairs ?</label>

                            <select name="walking" id="walking" required>

                                <option value="1">Yes</option>

                                <option value="0">No</option>

                            </select>

              </div>

              <div class="form-group">

                <label for="education">What is your education level in Education level scale (EDUCA see codebook) scale 1-6 :</label>

                <select name="education" id="education">

                    <option value="1">Never attended school or only kindergarten </option>

                    <option value="2">Grades 1 through 8 (Elementary)</option>

                    <option value="3">Grades 9 through 11 (Some high school)</option>

                    <option value="4">Grade 12 or GED (High school graduate)</option>

                    <option value="5">College 1 year to 3 years (Some college or technical school)</option>

                    <option value="6">College 4 years or more (College graduate)</option>

                </select>

              </div>

              <div class="form-group">

                <label for="income">What will be your income in Income scale (INCOME2 see codebook) <br><b>scale 1-8 :</b></label>

                            <select name="income" id="income">

                                <option value="1">Less than $10,000 </option>

                                <option value="2">$10,000 to less than $15,000</option>

                                <option value="3">$15,000 to less than $20,000</option>

                                <option value="4">$20,000 to less than $25,000</option>

                                <option value="5">$25,000 to less than $35,000</option>

                                <option value="6">$35,000 to less than $50,000</option>

                                <option value="7">$50,000 to less than $75,000</option>

                                <option value="8">$75,000 or more</option>

                            </select>

              </div>

              <div class="form-group">

                <label for="ghealth">What would be your health in general scale of <b>1-5</b><br>

                </label>

                <select name="ghealth" id="ghealth">

                    <option value="1">Excellent</option>

                    <option value="2">Very Good</option>

                    <option value="3">Good</option>

                    <option value="4">fair</option>

                    <option value="5">poor</option>

                </select>

              </div>

              <button type="button" class="previous-btn">Prev</button>

              <input type="submit" class="submit-btn" value="Predict">

              <!-- <button type="submit" class="submit-btn">Submit</button> -->

            </div>

          </form>

        </div>

      </section>

      <script type="text/javascript" src="static/js/form.js"></script>

</body>

</html>

#Home.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>complete responsive hospital website design </title>

    <!-- font awesome cdn link  -->

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css">

    <!-- custom css file link  -->

    <link rel="stylesheet" href="static/css/home.css">

</head>

<body>

<!-- header section starts  -->

<header class="header">

    <a href="#" class="logo"> <i class="fas fa-heartbeat"></i> medcare. </a>

    <nav class="navbar">

        <a href="{{url\_for('home')}}">home</a>

        <a href="{{url\_for('contact')}}">Contact</a>

    </nav>

    <div id="menu-btn" class="fas fa-bars"></div>

</header>

<!-- header section ends -->

<!-- home section starts  -->

<section class="home" id="home">

    <div class="image">

        <img src="static/img/home-img.svg" alt="">

    </div>

    <div class="content">

        <h3>stay safe, stay healthy</h3>

        <p>“Time and health are two precious assets that we don’t recognize and appreciate until they have been depleted.” </p>

        <a href="#" class="btn"> Know More <span class="fas fa-chevron-right"></span> </a>

        <a href="{{url\_for('form')}}" class="btn"> Predict My Health <span class="fas fa-chevron-right"></span> </a>

    </div>

</section>

<!-- home section ends -->

<!-- custom js file link  -->

<script src="static/js/script.js"></script>

</body>

</html>

#Index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="https://pro.fontawesome.com/releases/v5.10.0/css/all.css" integrity="sha384-AYmEC3Yw5cVb3ZcuHtOA93w35dYTsvhLPVnYs9eStHfGJvOvKxVfELGroGkvsg+p" crossorigin="anonymous" />

    <link rel="stylesheet" href="static/css/index.css">

    <title>signin-signup</title>

</head>

<body>

    <div class="container">

        <div class="signin-signup">

            <form action="{{url\_for('login')}}" class="sign-in-form" method="post">

                <h2 class="title">Sign in</h2>

                <div class="input-field">

                    <i class="fas fa-user"></i>

                    <input type="text" placeholder="E-mail" name="email">

                </div>

                <div class="input-field">

                    <i class="fas fa-lock"></i>

                    <input type="password" placeholder="Password" name="password">

                </div>

                <input type="submit" value="Login" class="btn">

                <p class="social-text">Or Sign in with social platform</p>

                <div class="social-media">

                    <a href="#" class="social-icon">

                        <i class="fab fa-facebook"></i>

                    </a>

                    <a href="" class="social-icon">

                        <i class="fab fa-twitter"></i>

                    </a>

                    <a href="" class="social-icon">

                        <i class="fab fa-google"></i>

                    </a>

                    <a href="" class="social-icon">

                        <i class="fab fa-linkedin-in"></i>

                    </a>

                </div>

                <p class="account-text">Don't have an account? <a href="#" id="sign-up-btn2">Sign up</a></p>

            </form>

            <form action="{{url\_for('register')}}" class="sign-up-form" method="post">

                <h2 class="title">Sign up</h2>

                <div class="input-field">

                    <i class="fas fa-user"></i>

                    <input type="text" placeholder="Username" name="name">

                </div>

                <div class="input-field">

                    <i class="fas fa-envelope"></i>

                    <input type="text" placeholder="email" name="email">

                </div>

                <div class="input-field">

                    <i class="fas fa-lock"></i>

                    <input type="password" placeholder="password" name="password">

                </div>

                <input type="submit" value="Sign up" class="btn">

                <p class="social-text">Or Sign in with social platform</p>

                <div class="social-media">

                    <a href="#" class="social-icon">

                        <i class="fab fa-facebook"></i>

                    </a>

                    <a href="" class="social-icon">

                        <i class="fab fa-twitter"></i>

                    </a>

                    <a href="" class="social-icon">

                        <i class="fab fa-google"></i>

                    </a>

                    <a href="" class="social-icon">

                        <i class="fab fa-linkedin-in"></i>

                    </a>

                </div>

                <p class="account-text">Already have an account? <a href="#" id="sign-in-btn2">Sign in</a></p>

            </form>

        </div>

        <div class="panels-container">

            <div class="panel left-panel">

                <div class="content">

                    <h3>Member of Brand?</h3>

                    <p>“Who you're connected to is very important because that connection can lead to you being innovative or becoming complacent.” </p>

                    <button class="btn" id="sign-in-btn">Sign in</button>

                </div>

                <img src="static/img/signin1.svg" alt="" class="image">

            </div>

            <div class="panel right-panel">

                <div class="content">

                    <h3>New to Brand?</h3>

                    <p>You may say I'm a dreamer, but I'm not the only one. I hope someday you'll join us. And the world will live as one.”</p>

                    <button class="btn" id="sign-up-btn">Sign up</button>

                </div>

                <img src="static/img/signup1.svg" alt="" class="image">

            </div>

        </div>

    </div>

    <script src="static/js/app.js"></script>

</body>

</html>

#Application Code

from flask import Flask, flash, redirect, request, render\_template, session, url\_for

import sqlite3

import pickle

import numpy as np

# creating database

con=sqlite3.connect("database.db")

print("Opened database successfully")

con.execute("create table if not exists user(pid integer primary key, name text, email text, password text,status BOOLEAN)")

print("User table created successfully")

con.execute("create table if not exists details(pid integer primary key, HighBP REAL, HighChol REAL, CholCheck REAL, BMI REAL, Smoker REAL, Stroke REAL, HeartDiseaseorAttack REAL, PhysActivity REAL, Fruits REAL, Veggies REAL, HvyAlcoholConsump REAL, AnyHealthcare REAL, NoDocbcCost REAL, GenHlth REAL, MentHlth REAL, PhysHlth REAL, DiffWalk REAL, Sex REAL, Age REAL, Education REAL, Income REAL)")

print("Details table created successfully")

con.close()

app = Flask(\_\_name\_\_)

app.secret\_key="#@diabeticpredictionflaskapp@#"

# routeing the application

@app.route("/",methods=['POST','GET'])

def index():

    return render\_template("index.html")

@app.route("/loginpage",methods=['POST','GET'])

def loginpage():

    return render\_template("index.html")

@app.route("/home")

def home():

    return render\_template('home.html')

@app.route("/contact")

def contact():

    return render\_template('contact.html')

@app.route("/form")

def form():

    return render\_template('form.html')

@app.route("/wrong")

def wrong():

    return render\_template('wrong.html')

# login and register

@app.route('/register',methods=['GET','POST'])

def register():

    if request.method == 'POST':

        try:

            name=request.form['name']

            email=request.form['email']

            password=request.form['password']

            con=sqlite3.connect("database.db")

            cur=con.cursor()

            cur.execute("INSERT INTO user(name,email,password) VALUES (?,?,?)",(name,email,password))

            con.commit()

            flash("Registered successfully","success")

            print("Registered")

        except:

            con.rollback()

            flash("Problem in Registration, Please try again","danger")

        finally:

            return redirect(url\_for('login'))

            con.close()

    else:

        return render\_template("wrong.html")

@app.route('/login',methods=['POST','GET'])

def login():

    if request.method =='POST':

        email = request.form['email']

        password = request.form['password']

        con=sqlite3.connect("database.db")

        con.row\_factory=sqlite3.Row

        cur=con.cursor()

        cur.execute("SELECT \* FROM user where email=? and password=?",(email,password))

        data=cur.fetchone()

        if data:

            session["email"]=data["email"]

            print("sent to home")

            return redirect(url\_for("home"))

        else:

            flash("Username or Password is incorrect","danger")

            print("not sent to home")

    return redirect(url\_for('loginpage'))

@app.route("/predict",methods = ['POST','GET'])

def predict():

    name = request.form.get("name")

    print(name)

    age = request.form.get("age")

    gender = request.form.get("gender")

    bmi = request.form.get("bmi")

    mhealth = request.form.get("mhealth")

    phealth = request.form.get("phealth")

    bp = request.form.get("bp")

    hc = request.form.get("hc")

    cc = request.form.get("cc")

    smoker = request.form.get("smoker")

    stroke = request.form.get("stroke")

    hearta = request.form.get("hearta")

    phact = request.form.get("phact")

    fruit = request.form.get("fruit")

    veggies = request.form.get("veggies")

    drinker = request.form.get("drinker")

    hcare = request.form.get("hcare")

    cost = request.form.get("cost")

    walking = request.form.get("walking")

    education = request.form.get("education")

    income = request.form.get("income")

    ghealth = request.form.get("ghealth")

    model = pickle.load(open("second\_model.pkl","wb"))

    feature = np.array([[bp,hc,cc,bmi,smoker,stroke,hearta,phact,fruit,veggies,drinker,hcare,cost,ghealth,mhealth,phealth,walking,gender,age,education,income,model]])

    print(name)

    print(feature)

    prediction = model.predict(feature)

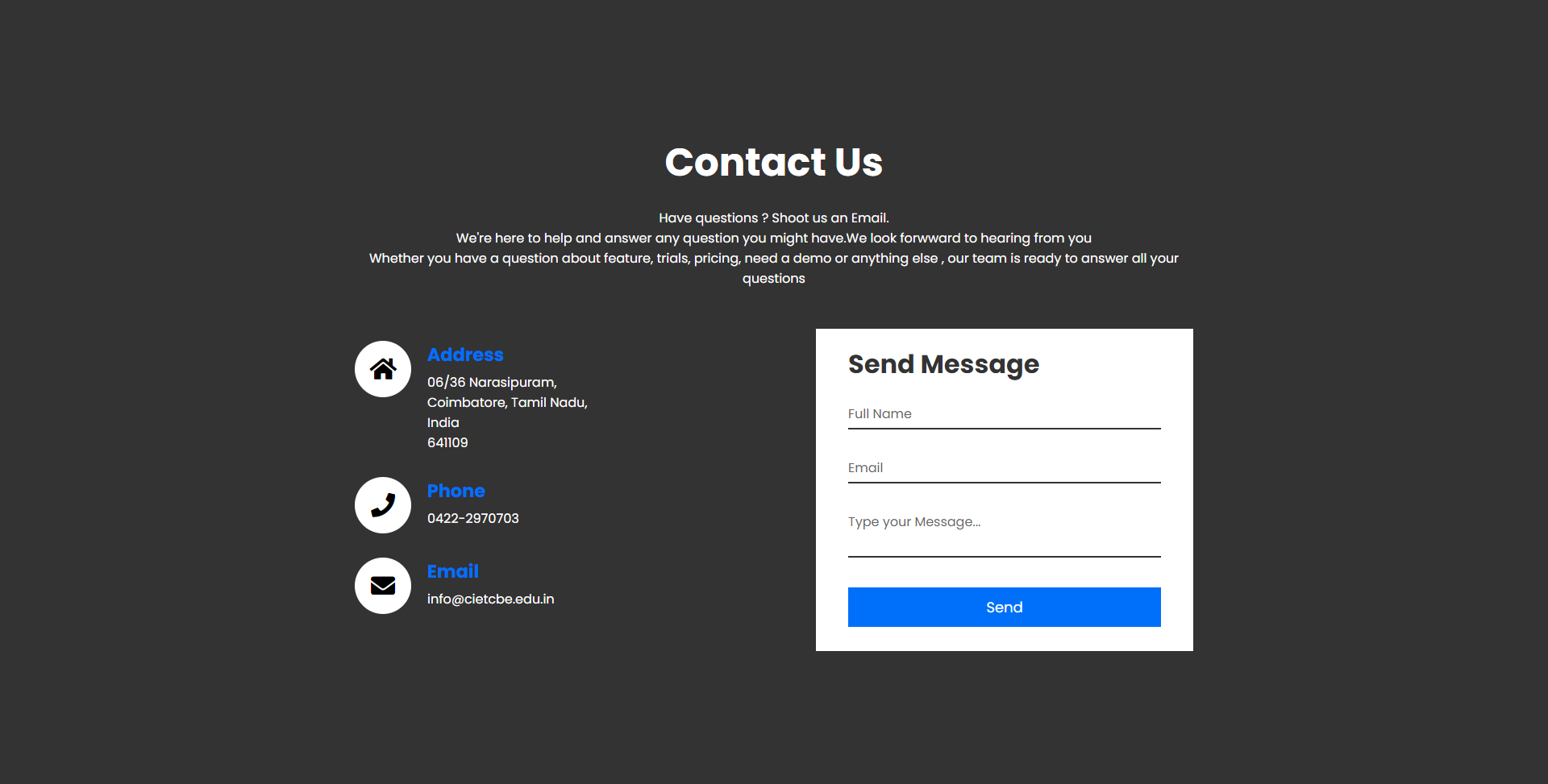
    print(prediction)

    return render\_template("home.html")

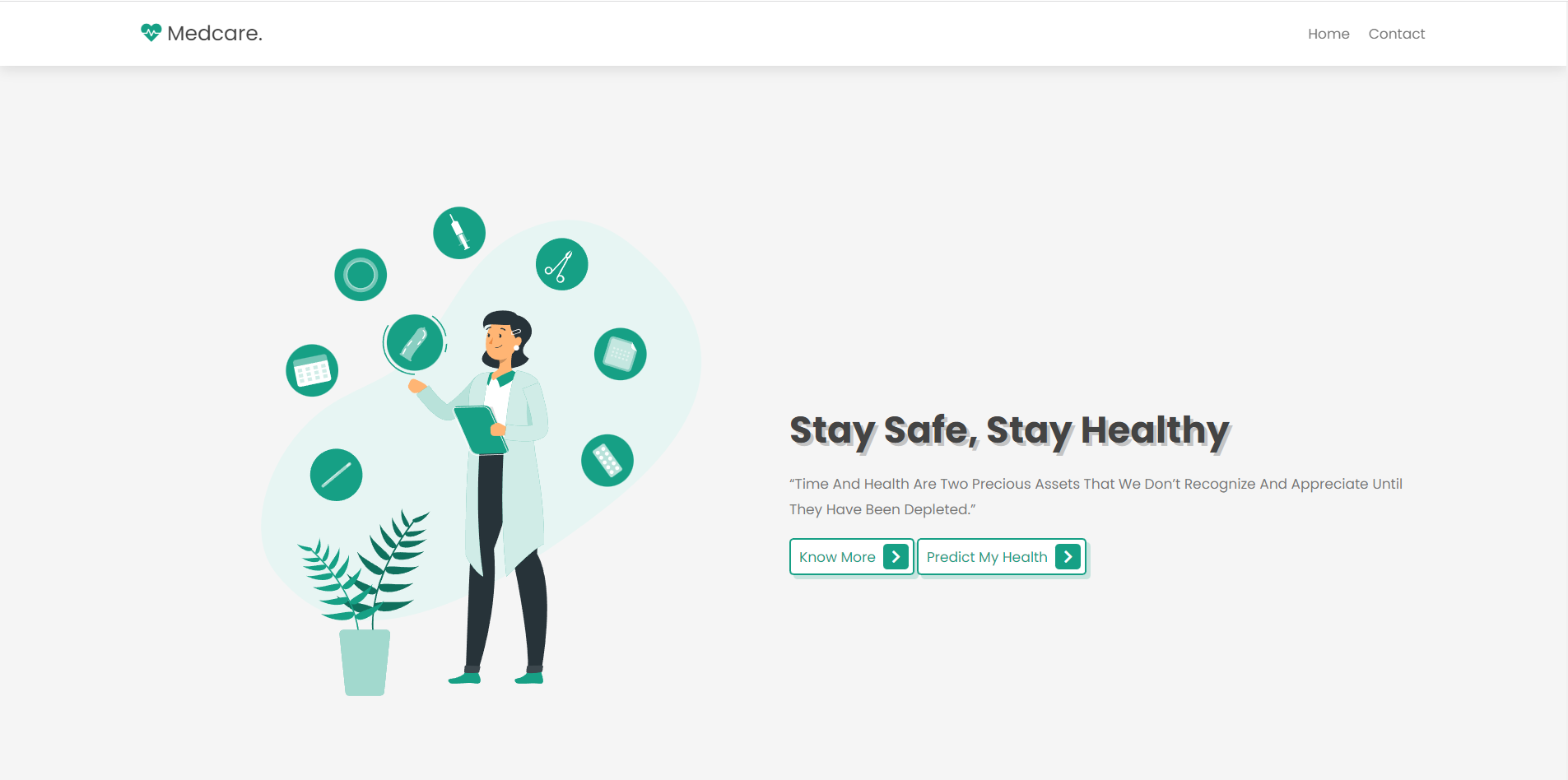
if \_\_name\_\_ == '\_\_main\_\_':

    app.run(Debug = True)

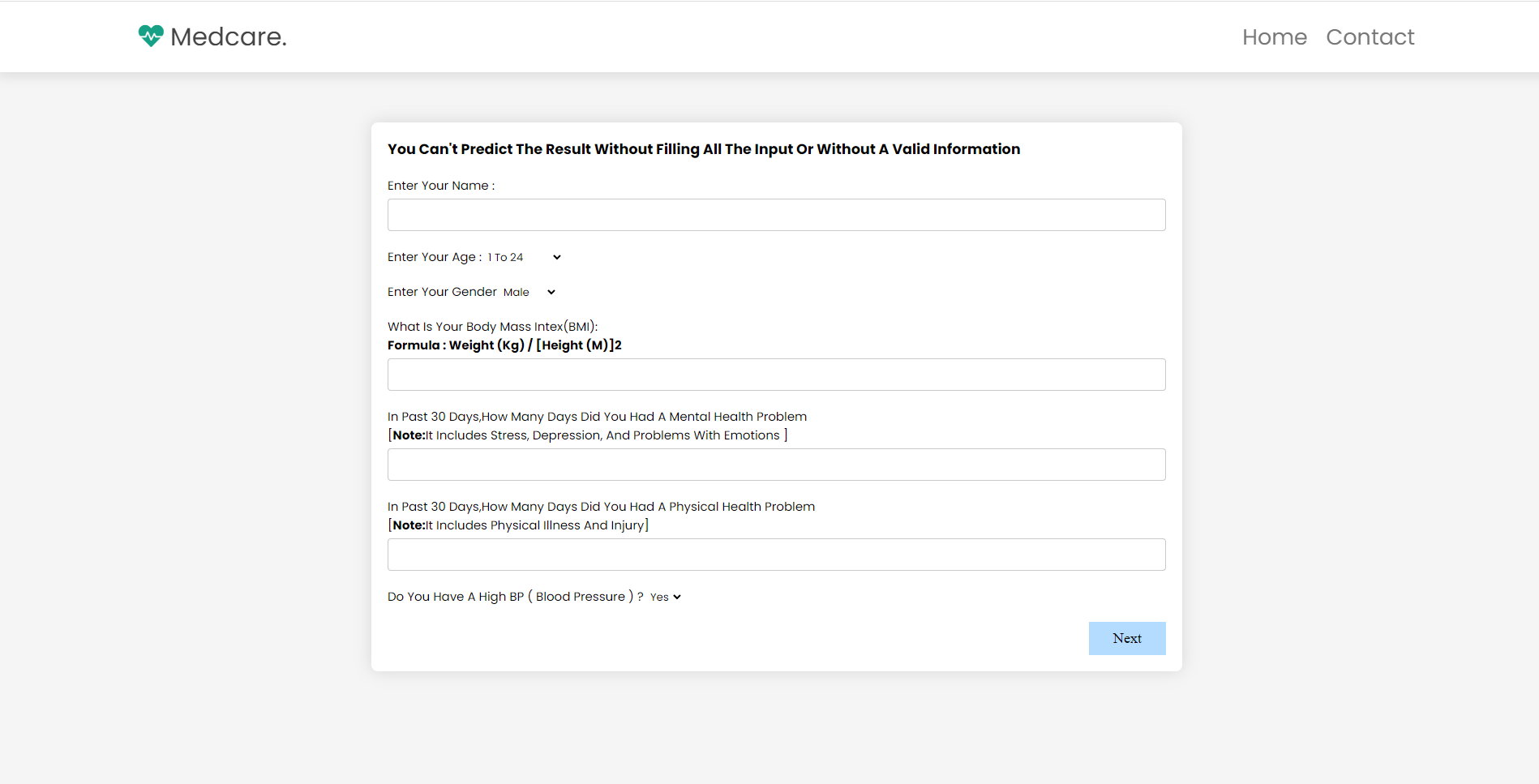
**6.3.1 Web Interface Results :**

#Contact Page  


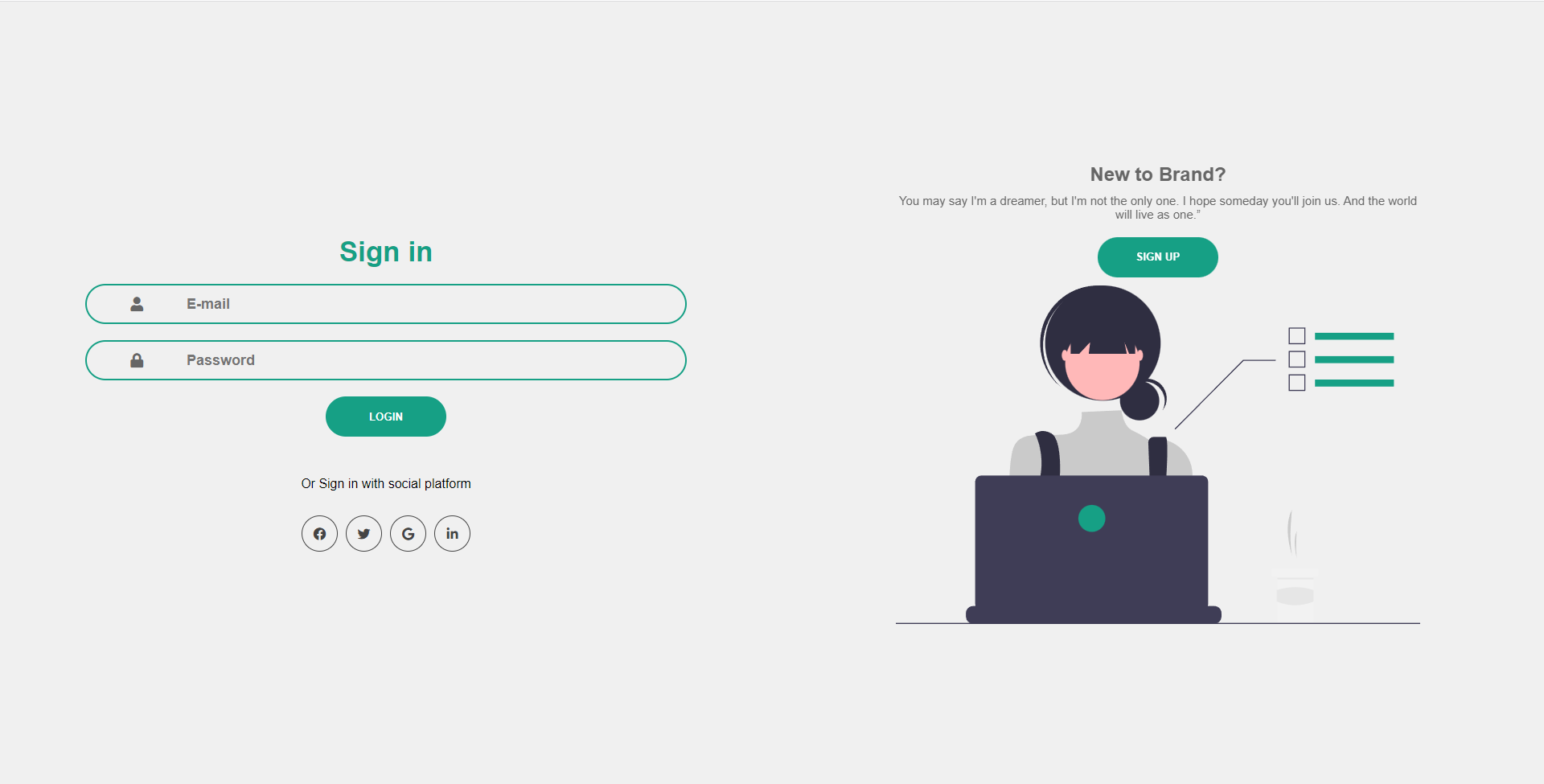
#Home Page



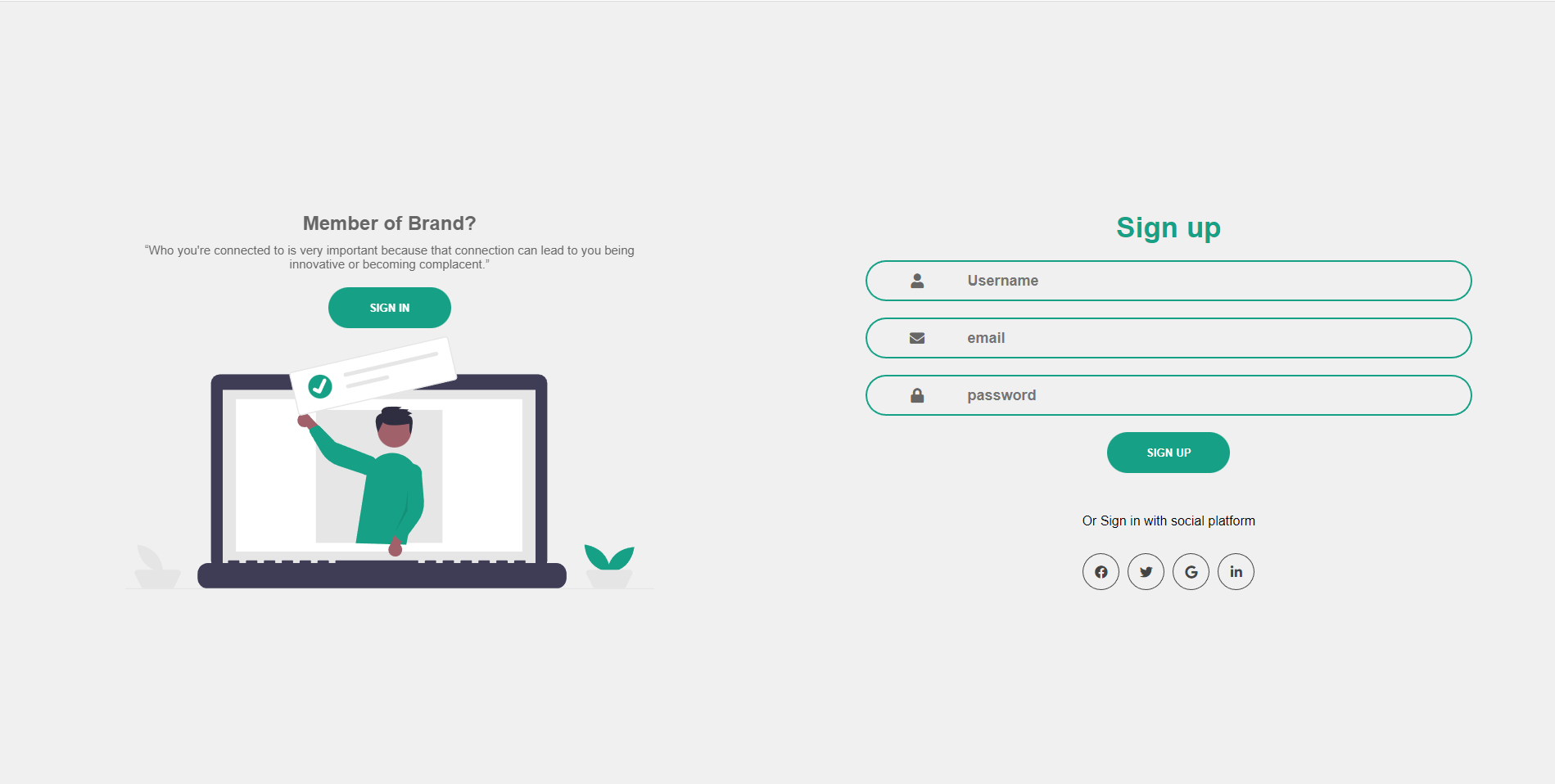
#Prediction Page



#signin Page



#Signup Page



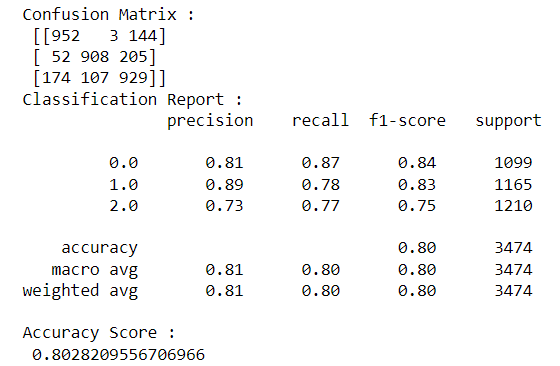
**6.4 Performance Matrix**

from sklearn.metrics import accuracy\_score, classification\_report, confusion\_matrix

print("Confusion Matrix : \n", confusion\_matrix(y\_test,y\_pred))

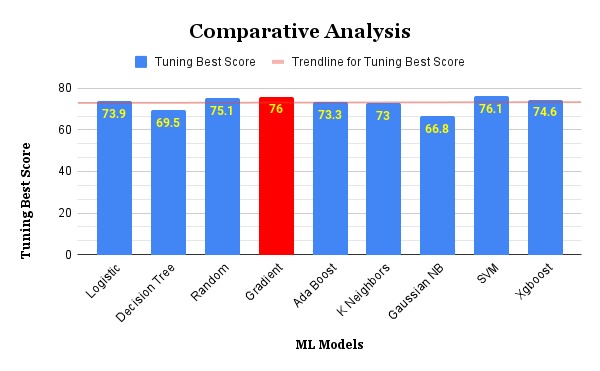
print("Classification Report : \n", classification\_report(y\_test,y\_pred))

print("Accuracy Score : \n", accuracy\_score(y\_test, y\_pred))



**Comparative Analysis**

Compare the accuracy of the classification algorithms of Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Support Vector Machine Comparison(SVM), XG Boost, K-Nearest Neighbour(KNN), Gaussian Naïve Bayes, Gradient Boosting Classifier and AdaBoost Classifier.



**7.Conclusion** **and Future work**

We proposed a web based application for the successful prediction of Diabetes Diseases. From different machine learning algorithms Gradient Boost with Standard Scaling Method and Hyperparameter Tuning provide us highest accuracy. As we have proposed and developed an approach for diabetes disease prediction using machine learning algorithm, it has significant potential in the field of medical science for the detection of various medical data accurately.

**8.References**

1. Gauri D. Kalyankar, Shivananda R. Poojara and Nagaraj V. Dharwadkar,” Predictive Analysis of Diabetic Patient Data Using Machine Learning and Hadoop”, International Conference On I-SMAC, 978-1-5090-3243-3, 2017.
2. Ayush Anand and Divya Shakti,” Prediction of Diabetes Based on Personal Lifestyle Indicators”, 1st International Conference on Next Generation Computing Technologies, 978-1-4673-6809-4, September 2015.
3. B. Nithya and Dr. V. Ilango,” Predictive Analytics in Health Care Using Machine Learning Tools and Techniques”, International Conference on Intelligent Computing and Control Systems, 978-1-5386-2745-7, 2017.
4. Dr Saravana kumar N M, Eswari T, Sampath P and Lavanya S,” Predictive Methodology for Diabetic Data Analysis in Big Data”, 2nd International Symposium on Big Data and Cloud Computing, 2015.
5. Aiswarya Iyer, S. Jeyalatha and Ronak Sumbaly**“**Diagnosis of Diabetes Using Classification Mining Techniques”, International Journal of Data Mining & Knowledge Management Process (IJDKP), 5 (1) (January 2015)
6. P. Suresh Kumar and S. Pranavi “Performance Analysis of Machine Learning Algorithms on Diabetes Dataset using Big Data Analytics”, International Conference on Infocom Technologies and Unmanned Systems, 978-1-5386-0514-1, Dec. 18-20, 2017.
7. Mani Butwall, Shraddha Kumar**“**A Data Mining Approach for the Diagnosis of Diabetes Mellitus using Random Forest Classifier”, International Journal of Computer Applications, 120 (Number 8, 2015)
8. K. Rajesh, V. Sangeetha**“Application of Data Mining Methods and Techniques for Diabetes Diagnosis”**International Journal of Engineering and Innovative Technology (IJEIT), 2 (3) (September 2012)
9. Humar Kahramanli, Novruz Allahverdi**“Design of a Hybrid System for the Diabetes and Heart Disease” E**xpert Systems with Applications: An International Journal, 35 (1-2) (July, 2008)
10. B.M. Patil, R.C. Joshi and Durga Toshniwal, “Association Rule for Classification of Type-2 Diabetic Patients”, ICMLC ‘10 Proceedings of the 2010 Second International Conference on Machine Learning and Computing, February 09 - 11, 2010.
11. Dost Muhammad Khan1, Nawaz Mohamudally2**“An Integration of K-means and Decision Tree (ID3) towards a more Efficient Data Mining Algorithm ”**Journal Of Computing, 3 (12) (December 2011)
12. Statista Number of smartphone users worldwide from 2016 to 2021. Available from <https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/>. Accessed 1 April 2020
13. Hood M, Wilson R, Corsica J, Bradley L, Chirinos D, Vivo A. What do we know about mobile applications for diabetes self-management? A review of reviews. *J Behav Med* 2016;39:981–994 [[PubMed](https://pubmed.ncbi.nlm.nih.gov/27412774)] [[Google Scholar](https://scholar.google.com/scholar_lookup?journal=J+Behav+Med&title=What+do+we+know+about+mobile+applications+for+diabetes+self-management?+A+review+of+reviews&author=M+Hood&author=R+Wilson&author=J+Corsica&author=L+Bradley&author=D+Chirinos&volume=39&publication_year=2016&pages=981-994&pmid=27412774&)]
14. Business of Apps App download and usage statistics 2019. Available from <https://www.businessofapps.com/data/app-statistics/>. Accessed 29 March 2020
15. Annie A. Report: the state of mobile 2020. Available from <https://www.appannie.com/en/go/state-of-mobile-2020/>. Accessed 29 March 2020
16. Statista Number of mHealth apps available in the Apple App Store from 1st quarter 2015 to 3rd quarter 2019. Available from <https://www.statista.com/statistics/779910/health-apps-available-ios-worldwide/>. Accessed 30 March 2020
17. Statista Number of mHealth apps available at Google Play from 1st quarter 2015 to 3rd quarter 2019. Available from <https://www.statista.com/statistics/779919/health-apps-available-google-play-worldwide/>. Accessed 30 March 2020
18. Istepanian RSH, Laxminarayan S, Pattichis CS, Eds. *M-Health: Emerging Mobile Health Systems*. Boston, MA, Springer, 2006 [[Google Scholar](https://scholar.google.com/scholar_lookup?title=M-Health:+Emerging+Mobile+Health+Systems&publication_year=2006&)]