

EARLY-STAGE DIABETES PREDICTION USING MACHINE LEARNING TECHNIQUE

A PROJECT REPORT

Submitted by

SONIYA.D [REGISTER NO:211420104258]

SOWMIYA. A.V [REGISTER NO:211420104259]

SWETHA.K [REGISTER NO:211420104280]

in partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING



PANIMALAR ENGINEERING COLLEGE, CHENNAI-600123

(An Autonomous Institution, Affiliated to Anna University, Chennai)

MARCH 2024

PANIMALAR ENGINEERING COLLEGE

(An Autonomous Institution, Affiliated to Anna University, Chennai)

BONAFIDE CERTIFICATE

Certified that this project report “**Early-Stage Diabetics Prediction Using Machine Learning**” is the bonafide work of “SONIYA.D[211420104258], SOWMIYA.A.V[211420104259],SWETHA.K[211420104280]” who carried, out the project work under my supervision.

Signature of the HOD with date

**DR L.JABASHEELA M.E.,
Ph.D.,
PROFESSOR AND HEAD,**

Department Of Computer Science
and Engineering
Panimalar Engineering College
Chennai 600-123

Signature of the Supervisor with date

**Dr. T. JACKULIN
M.E.,Ph.D.,
SUPERVISOR
ASSOCIATE PROFESSOR**

Department Of Computer Science
and Engineering
Panimalar Engineering College
Chennai 600-123

Certified that the above candidate(s) was examined in the End Semester Project Viva-Voce Examination held on

INTERNAL EXAMINER

EXTERNAL EXAMINER

DECLARATION OF THE STUDENT

We **SWETHA.K [211420104280], SONIYA.D [2114201054258], SOWMIYA. A.V [211420104259]** hereby declare that this project report titled **“EARLY-STAGE DIABETES PREDICTION USING MACHINE LEARNING TECHNIQUE”**, under the guidance of Dr. T. JACULIN.,ME.,Ph.D., is the original work done and we have not plagiarized or submitted to any other degree in any university by us.

SONIYA.D

SOWMIYA.A.V

SWETHA.K

ACKNOWLEDGMENT

Our profound gratitude is directed towards our esteemed Secretary and Correspondent, **Dr. P. CHINNADURAI, M.A., Ph.D.**, for his fervent encouragement. His inspirational support proved instrumental in galvanizing our efforts, ultimately contributing significantly to the successful completion of this project.

We want to express our deep gratitude to our Directors, **Tmt. C. VIJAYARAJESWARI, Dr. C. SAKTHI KUMAR, M.E., Ph.D., and Dr. SARANYASREE SAKTHI KUMAR, B.E., M.B.A., Ph.D.**, for graciously affordingus the essential resources and facilities for undertaking of this project.

Our gratitude is also extended to our Principal, **Dr. K. MANI, M.E., Ph.D.**, whose facilitation proved pivotal in the successful completion of this project.

We express our heartfelt thanks to **Dr. L. JABASHEELA, M.E., Ph.D.**, Head of the Department of Computer Science and Engineering, for granting the necessary facilities that contributed to the timely and successful completion of project.

We would like to express our sincere thanks to **Project Coordinator Mrs.VALARMATHI , M.E., Ph.D .**, and **Project Guide Dr.JACKULIN ,M.E.,Ph.D.**, and all the faculty members of the Department of CSE for their unwavering support for the successful completion of the project.

SONIYA.D [211420104258]

SOWMIYA. A.V [211420104259]

SWETHA.K[211420104280]

ABSTRACT

Early-stage diabetic prediction using machine learning is a significant research area that aims to improve the early detection and management of diabetes. Diabetes is a chronic metabolic disorder that affects a large population worldwide and can lead to severe health complications if left untreated. Machine learning algorithms have shown promise in analyzing diverse datasets and identifying patterns that may indicate the presence of early signs of diabetes. This paper presents an overview of the approach to developing a machine learning model for early-stage diabetic prediction. Early-stage diabetic prediction using machine learning has the potential to revolutionize healthcare by enabling timely intervention and personalized treatment for individuals at risk of developing diabetes. By identifying high-risk individuals early on, healthcare providers can implement preventive measures and lifestyle interventions to mitigate the progression of the disease and reduce associated complications. Future research in this field should focus on enhancing the accuracy and interpretability of predictive models, integrating additional data sources, and expanding the scope to cover various subtypes of diabetes. **Keywords** early-stage diabetic prediction, machine learning, feature selection, model training, hyper parameter tuning, validation, healthcare.

TABLE OF CONTENT

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	ii
	LIST OF FIGURES	iii
1.	INTRODUCTION	
	1.1 Overview	1
	1.2 Problem Definition	2
	1.3 Objective	3
2.	LITERATURE SURVEY	5
3.	SYSTEM ANALYSIS	
	3.1 Existing System	6
	3.2 Proposed System	7
	3.3 Feasibility Study	8
	3.4 Hardware Environment	9
	3.5 Software Environment	19
4.	SYSTEM DESIGN	
	4.1 ER Diagram	11
	4.2 Data Flow Diagram	12
	4.3 UML Diagram	13

CHAPTER NO	TITLE	PAGE NO
5.	SYSTEM ARCHITECTURE	
	5.1 System Architecture Diagram	16
	5.2 Module Design Specification	18
	5.3 Algorithms	22
6.	PERFORMANCE ANALYSIS	
	6.1 Performance Metrics and parameters	24
	6.2 Result and Discussion	27
7.	SYSTEM TESTING	
	7.1 Test Case Report	29
8.	CONCLUSION	
	8.1 Conclusion	33
	8.2 Future Enhancement	34
	APPENDICS	
	A.1 SDC GOALS	35
	A.2 CLIENT-SIDE CODING	36
	A.3 SCREEN SHOTS	49
	A.4 PATENT PROOF	52
	REFERENCES	54

LIST OF FIGURES

FIGNO	FIGURE DESCRIPTION	PAGE NO
4.1.1	ER DIAGRAM	25
4.2.1	DATA FLOW DIAGRAM	26
4.3.1	USE CASE DIAGRAMS	27
4.3.2	CLASS DIAGRAM	28
5.1.1	SYSTEM ARCHITECTURE	30

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

The paper discusses the significance of early-stage diabetic prediction using machine learning, highlighting its potential to improve the detection and management of diabetes. It emphasizes the role of machine learning algorithms in analyzing diverse datasets to identify patterns indicative of early signs of diabetes. By enabling timely intervention and personalized treatment, this approach could revolutionize healthcare by reducing complications associated with untreated diabetes. Future research should aim to enhance model accuracy and interpretability, integrate additional data sources, and cover various subtypes of diabetes. Keywords include early-stage diabetic prediction, machine learning, feature selection, model training, hyperparameter tuning, validation, and healthcare.

1.2 PROBLEM DEFINITION

- The problem addressed in this paper is the early detection and management of diabetes using machine learning techniques.
- Diabetes is a widespread chronic metabolic disorder that can lead to severe health complications if not treated promptly.
- Machine learning algorithms offer promise in analyzing diverse datasets to identify patterns indicative of early signs of diabetes.
- The paper presents an overview of the approach to developing a machine learning model for early-stage diabetic prediction, highlighting its potential to revolutionize healthcare by enabling timely intervention and personalized treatment.

- By identifying individuals at high risk of developing diabetes early on, healthcare providers can implement preventive measures and lifestyle interventions to mitigate disease progression and reduce associated complications.
- Future research in this area should aim to enhance the accuracy and interpretability of predictive models, integrate additional data sources, and extend the scope to cover various subtypes of diabetes.

1.3 OBJECTIVES

- To goal is to develop a machine learning model for Diabetics prediction, to potentially replace the updatable supervised machine learning classification models by predicting results in the form of best accuracy by comparing supervised algorithm.
- To project is that integration of diabetic's patient with computer-based prediction could reduce errors and improve prediction outcome. This suggestion is promising as data modeling and analysis tools, e.g.,
- To data mining, have the potential to generate a knowledge-rich environment which can help to significantly improve the quality of predicting Diabetics.

CHAPTER 2

LITERATURE SURVEY

LITERATURE SURVEY

1. Diabetes Prediction using Machine Learning Algorithms

Author: Aishwarya Mujumdar, Dr. Vaidehi

Year: 2019

Diabetes Mellitus is among critical diseases and lots of people are suffering from this disease. Age, obesity, lack of exercise, hereditary diabetes, living style, bad diet, high blood pressure, etc. can cause Diabetes Mellitus. People having diabetes have high risk of diseases like heart disease, kidney disease, stroke, eye problem, nerve damage, etc. Current practice in hospital is to collect required information for diabetes diagnosis through various tests and appropriate treatment is provided based on diagnosis. Big Data Analytics plays a significant role in healthcare industries. Healthcare industries have large volume databases.

2. Diabetes Prediction using Machine Learning Algorithms

Author: Aishwarya Mujumdar, Dr. Vaidehi

Year: 2019

Diabetes Mellitus is among critical diseases and lots of people are suffering from this disease. Age, obesity, lack of exercise, hereditary diabetes, living style, bad diet, high blood pressure, etc. can cause Diabetes Mellitus. People having diabetes have high risk of diseases like heart disease, kidney disease, stroke, eye problem, nerve damage, etc. Current practice in hospital is to collect required information for diabetes diagnosis through various tests and appropriate treatment is provided based on diagnosis. Big Data Analytics plays an significant role in healthcare industries. Healthcare industries have large volume databases. Using big data analytics one can study huge datasets and find hidden information, hidden patterns to discover knowledge from the data and predict outcomes accordingly. In existing method, the classification and prediction accuracy are not so high.

3. Diabetes Prediction Using Machine Learning

Author: KM Jyoti Rani

Year: 2020

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 causes 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.

4. : Application of Artificial Intelligence in Diabetes Education and Management

Author: Juan Li, Jin Huang

Year: 2020

Despite the rapid development of science and technology in healthcare, diabetes remains an incurable lifelong illness. Diabetes education aiming to improve the self-management skills is an essential way to help patients enhance their metabolic control and quality of life. Artificial intelligence (AI) technologies have made significant progress in transforming available genetic data and clinical information into valuable knowledge. The application of AI tech in disease education would be extremely beneficial considering their advantages in promoting individualization and full-course education intervention according to the unique pictures of different individuals. This paper reviews and discusses the most recent applications of AI techniques to various aspects of diabetes education.

5. Research on Diabetes Prediction Method Based on Machine Learning

Author: Jingyu Xue

Year: 2020

Diabetes mellitus (DM) is a metabolic disease characterized by high blood sugar. The main clinical types are type 1 diabetes and type 2 diabetes. Now, the proportion of young peoplesuffering from type 1 diabetes has increased significantly. Type 1 diabetes is chronic whenit occurs in childhood and adolescence and has a long incubation period. The early symptoms of the onset are not obvious, which may lead to failure to detect in time and delay treatment. Long- term high blood sugar can cause chronic damage and dysfunction of various tissues, especially eyes, kidneys, heart, blood vessels and nerves. Therefore, the early prediction of diabetes is particularly important. In this paper, we use supervised machine-learning algorithms like Support Vector Machine (SVM), Naive Bayes classifier and Light to train on the actual data of 520 diabetic patients and potential diabetic patients aged 16 to 90. Through comparative analysis of classification and recognition accuracy, the performance of support vector machine is the best.

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The difficulty for those with type 1 diabetes (T1D) is to provide exogenous insulin to keep blood glucose (BG) levels within a safe physiological range to prevent consequences that could be rather serious. Patients are helped by the artificial pancreas (AP) to overcome this difficulty by automating insulin injection. Even if insulin lowers blood glucose, the presence of another factor that raises blood glucose could enhance blood glucose regulation. Here, we build a model predictive control (MPC) algorithm that offers suggestions of carbohydrates (CHOs) as a second, glucose-increasing control input, in addition to insulin infusion. Because CHO consumption must be manually activated, much attention is taken to reduce any additional load that may be placed on patients. To achieve sparse CHO intake, a mixed logical-dynamical MPC formulation is used.

DEMERITS:

- They did not implement the deployment process.
- They implemented simple method. They did not Implement Machine Learning.
- They did not do data preprocessing and data cleaning process.
- They did not do data visualization by using Heat map, Pychart.
- Data analysis process Histogram, Plot and Graphs.
- They did not compare more than an algorithm to getting better accuracylevel.

3.2 PROPOSED SYSTEM

A machine learning-based system will be developed to predict the likelihood of individuals developing diabetes in the early stages. Comprehensive data on diabetes risk factors will be collected, including patient demographics, medical history, lifestyle factors, clinical measures, and blood sugar levels. The data will undergo preprocessing to handle missing values, outliers, and standardize formats, followed by feature selection to identify the most relevant features. An appropriate classification algorithm, such as logistic regression or decision trees, will be chosen and trained on the dataset using cross-validation techniques. Model hyperparameters will be optimized, and evaluation will be performed using metrics like accuracy, precision, recall, and F1-score, along with the utilization of a confusion matrix. Model interpretability will be enhanced using techniques like SHAP values or LIME. Deployment will be in a user-friendly interface accessible to healthcare professionals or individuals, with a focus on compliance with privacy regulations.

MERITS:

- We implemented the deployment process by using Frontend codes like Html, CSS, Bootstrap and Python Framework like Django or Flask.
- We implemented data preprocessing and data cleaning process by removing non-null values, missing values, duplicate values, unwanted data, and imbalanced data.
- We implemented data visualization by using Heat map, Pychart.
- We implemented Data analysis process by using Histogram, Plot and Graphs.
- We compared more than two algorithms to getting better accuracy level.
- We figure out performance and confusion metrics value properly.

3.3 FEASIBILITY STUDY

Data Wrangling

In this section of the report will load in the data, check for cleanliness, and then trim and clean given dataset for analysis. Make sure that the document steps carefully and justify for cleaning decisions.

Data Collection

The data set collected for predicting given data is split into Training set and Testset. Generally, 7:3 ratios are applied to split the Training set and Test set. The Data Model which was created using Random Forest, logistic, Decision tree algorithms and Support vector classifier (SVC) are applied on the Training set and based on the test result accuracy, Test set prediction is done.

Preprocessing

The data which was collected might contain missing values that may lead to inconsistency. To gain better results data need to be preprocessed to improve the efficiency of the algorithm. The outliers must be removed, and variable conversion need to be done.

Building the classification model

The prediction of diabetic, a high accuracy prediction model is effective because of the following reasons: It provides better results in classification problem.

- ▮ It is strong in preprocessing outliers, irrelevant variables, and a mix of continuous, categorical, and discrete variables.
- ▮ It produces out of bag estimate error which has proven to be unbiased in many tests and it is relatively easy to tune with .

3.4 SOFTWARE ENVIROMENT

- Operating System: Windows 10 or later
- Tool : Anaconda with Jupyter Notebook
- Language : python

3.5 HARDWARE ENVIRONMENT

- Processor : Intel i3
- Hard disk : minimum 80 GB
- RAM : minimum 2 GB

CHAPTER 4

SYSTEM DESIGN

4.1 Entity Relationship Diagram

That is a concise and accurate summary of what an ERD (Entity Relationship Diagram) is and its significance in the realm of information systems and database design. ERDs indeed play a crucial role in modeling the structure and relationships between various entities within a system, aiding in the understanding and development of databases. They serve as a blueprint for database design and are valuable tools throughout the software development lifecycle, from initial conceptualization to ongoing maintenance and optimization.

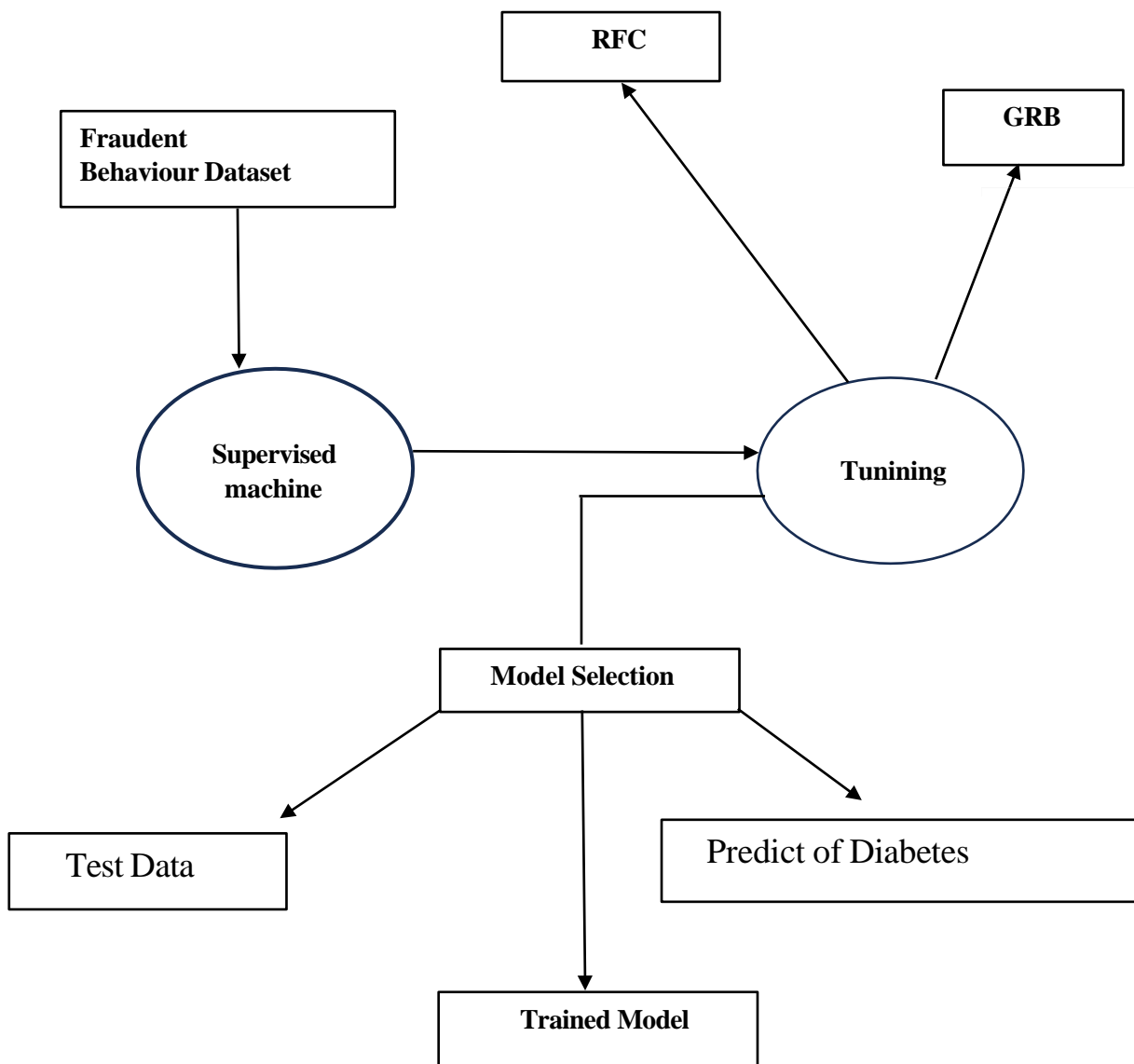


Fig 4.1.1. ER DIAGRAM

4.2 DATA FLOW DIAGRAMS

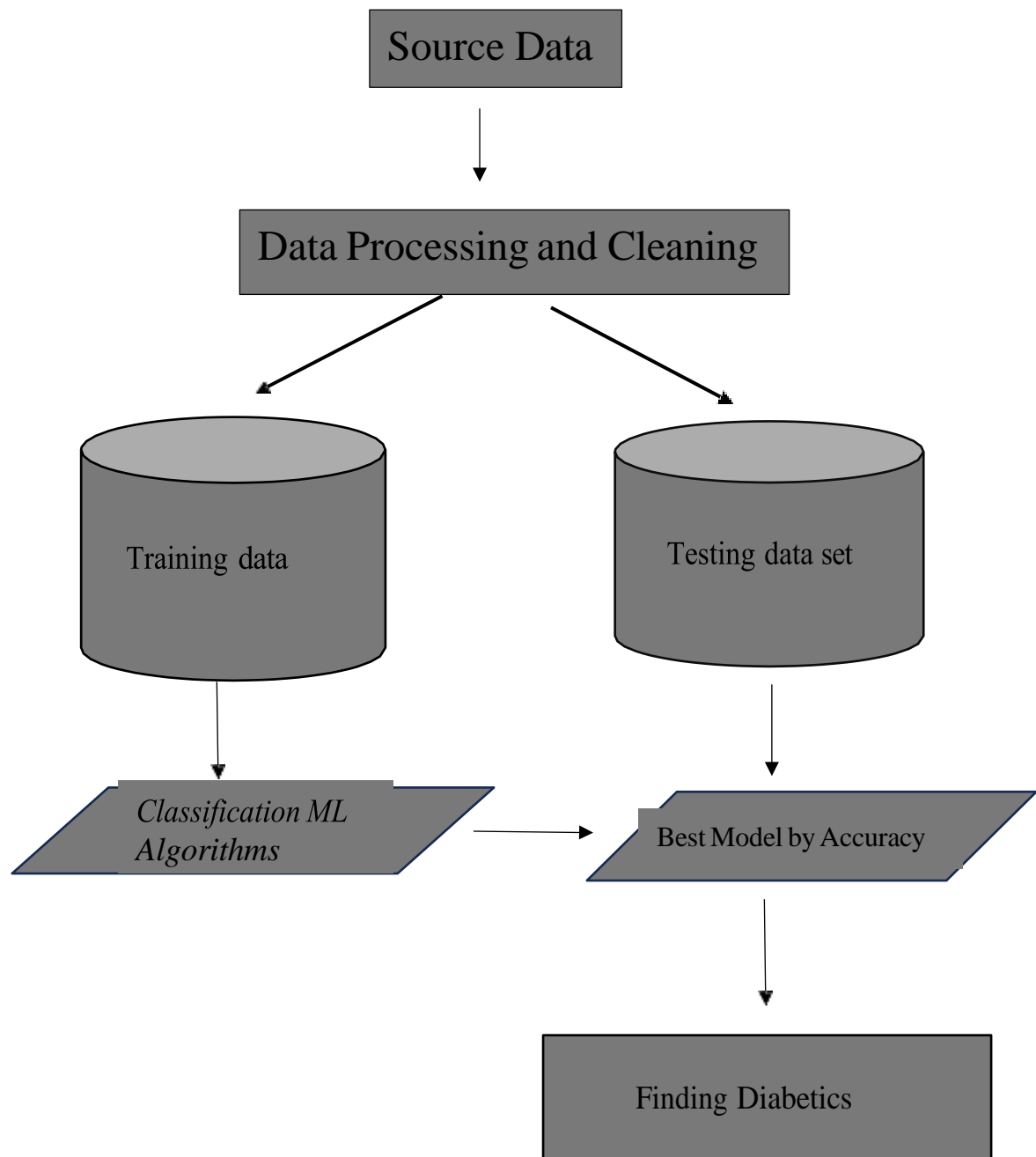


Fig 4.2.1. DATA FLOW DIAGRAMS

4.3 UML DIAGRAMS

4.3.1 USE CASE DIAGRAM

Use case diagrams are considered for high level requirement analysis of a system. So, when the requirements of a system are analyzed, the functionalities are captured in use cases. So, it can say that use cases are nothing, but the system functionalities written in an organized manner.

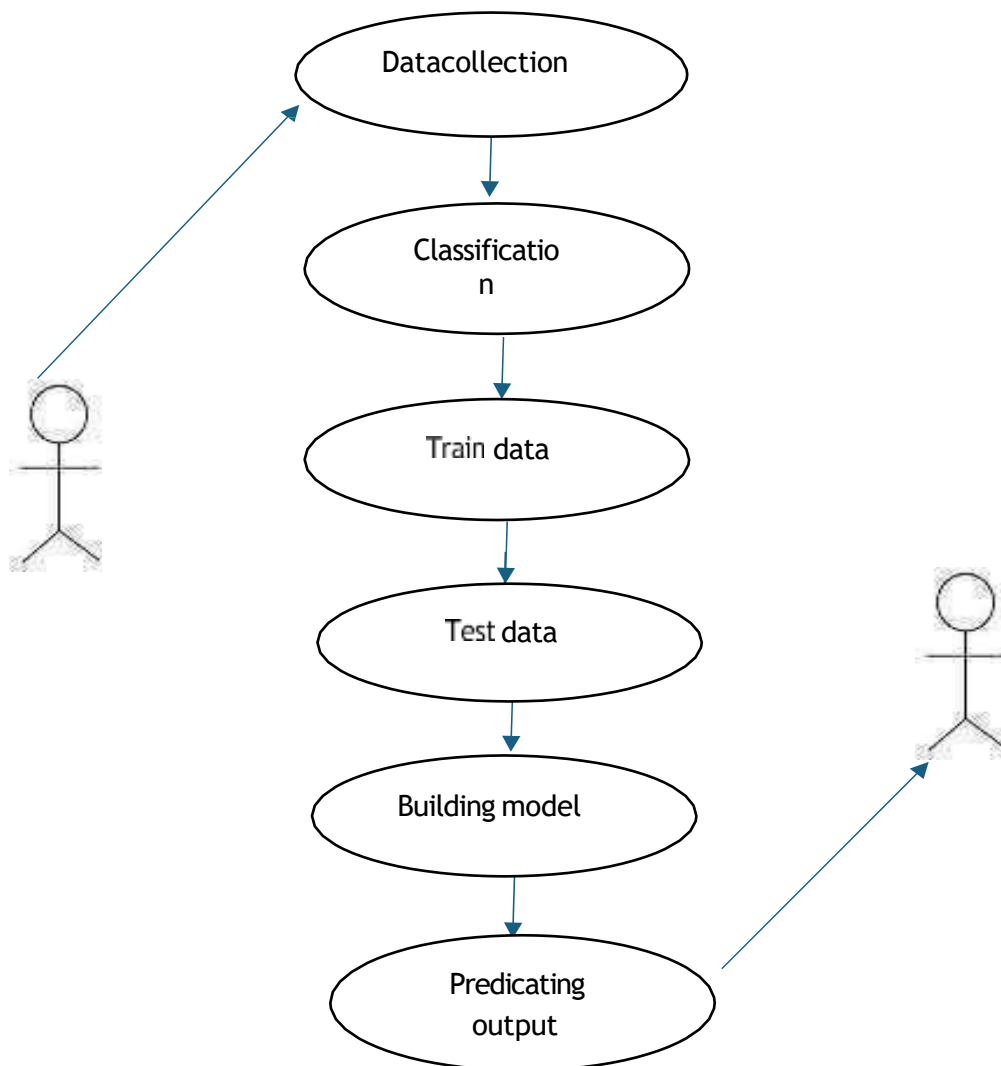


Fig 4.3.1.1 USECASE DIAGRAM

4.3.2 CLASS DIAGRAM

A class diagram is a visual representation of the static aspects of a system, encompassing various application elements. A set of class diagrams illustrates the entire system, each diagram's name reflecting its specific aspect. It is crucial to pre-identify elements and their relationships, along with clearly outlining each class's responsibilities (attributes and methods). To avoid complexity, limit properties to essential ones. Notes can clarify specific aspects, ensuring comprehensibility to developers. Before finalizing, sketch the diagram on plain, paper, iterating as needed for accuracy.

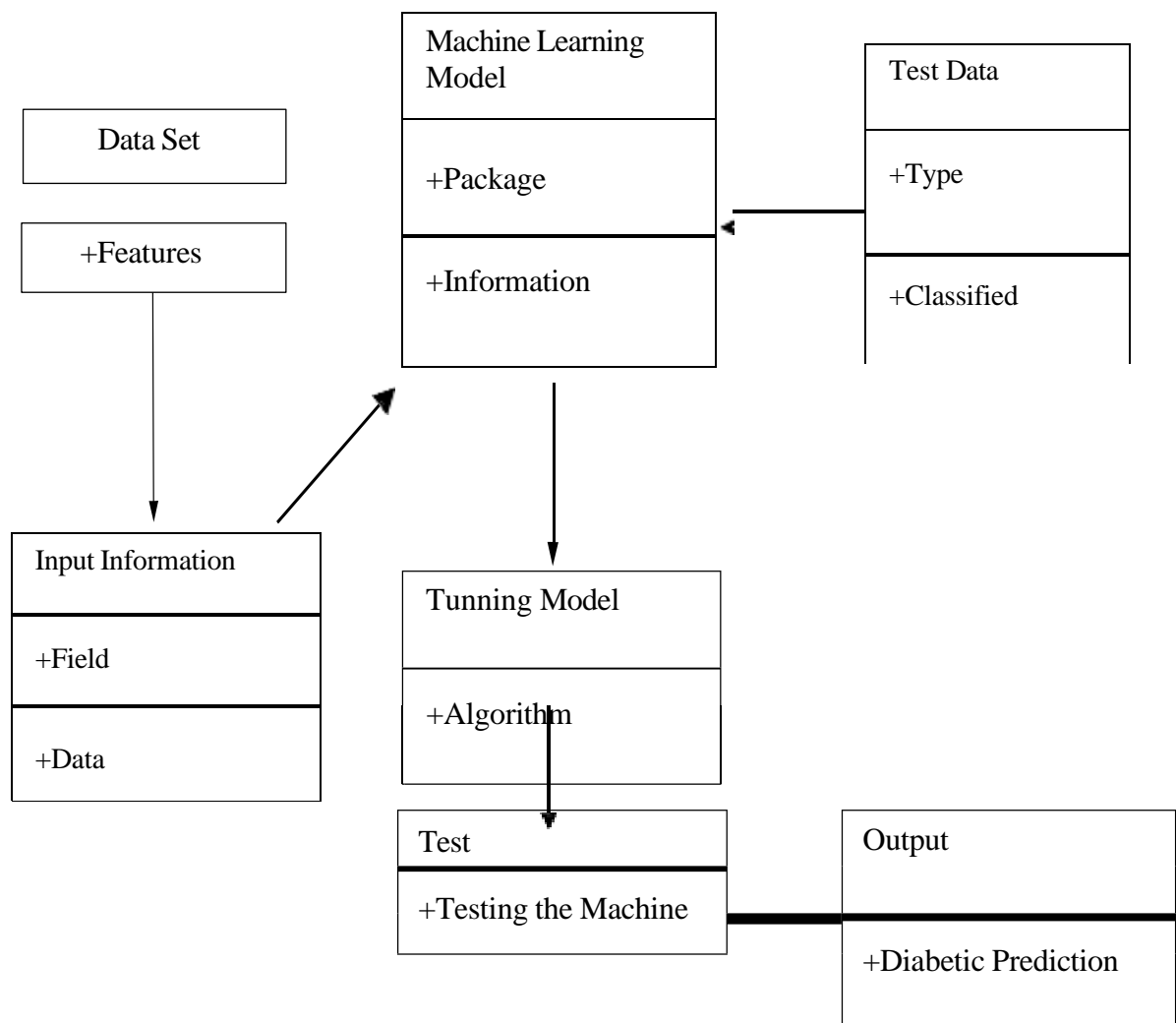


Fig .4.3.2.1 CLASS DIAGRAM

CHAPTER 5

SYSTEM ARCHITECTURE

5.1 SYSTEM ARCHITECTURE DIAGRAM

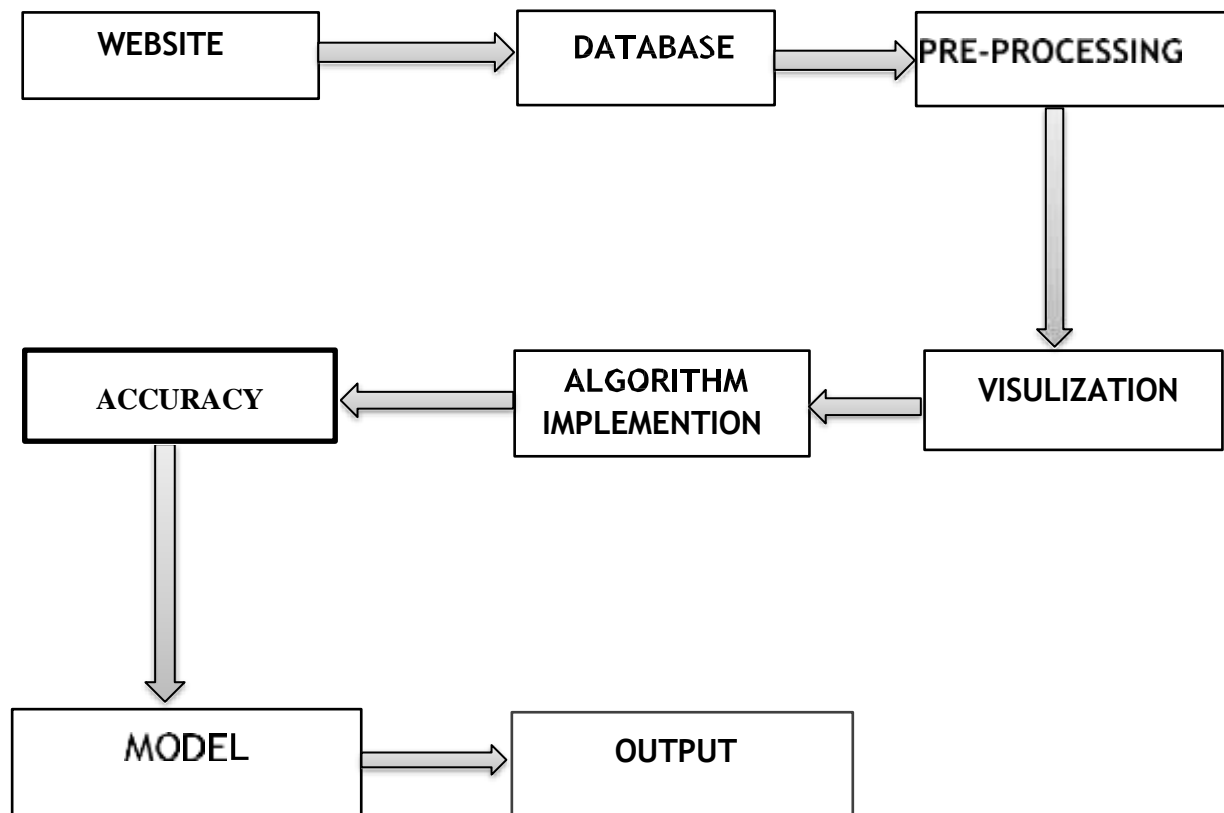


FIG 5.1.1 ARCHITECTURE DIAGRAM

In the figure of designing a system for managing diabetes, a robust architecture is crucial to ensure effectiveness and reliability. The system architecture would typically encompass several key components: pre-processing, visualization, algorithm implementation, accuracy assessment, model development, and input/output functionalities.

Firstly, the system would incorporate pre-processing mechanisms to handle raw data efficiently. This involves cleaning, filtering, and transforming data collected from various sources such as glucose monitors, insulin pumps, or wearable devices. Pre-processing ensures data consistency and prepares it for further analysis. Visualization tools play a vital role in providing meaningful insights into diabetic data. Interactive dashboards, charts, and graphs allow users to interpret trends, patterns, and anomalies in their glucose levels, insulin intake, dietary habits, and physical activities. This visual representation enhances understanding and decision-making for both patients and healthcare professionals.

Accuracy assessment is essential to evaluate the performance of the implemented algorithms. Metrics like sensitivity, specificity, and precision measure the model's ability to correctly identify diabetic conditions and predict future outcomes. Regular validation and testing ensure the reliability and effectiveness of the system. Model development involves creating predictive models tailored to individual patients' needs and characteristics. These models consider factors like insulin sensitivity, carbohydrate intake, physical activity levels, and stress levels to provide personalized recommendations for managing diabetes effectively.

Input and output functionalities facilitate seamless interaction between users and the system. Users can input relevant information such as blood glucose readings, meals, medications, and exercise routines. The system then generates actionable insights, alerts, reminders, and recommendations based on the input data and model predictions.

.

5.2 MODULE DESIGN SPECIFICATION

5.2.1 Data Pre-processing

Validation techniques in machine learning are used to get the error rate of the Machine Learning (ML) model, which can be considered as close to the true error rate of the dataset. If the data volume is large enough to be representative of the population, you may not need the validation techniques. However, in real-world scenarios, to work with samples of data that may not be a true representative of the population of given dataset. To finding the missing value, duplicate value, and description of data type whether it is float variable or integer. The sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyper parameters.

The evaluation becomes more biased as skill on the validation dataset is incorporated into the model configuration. The validation set is used to evaluate a given model, but this is for frequent evaluation. It as machine learning engineers use this data to fine-tune the model hyper parameters. Data collection, data analysis, and the process of addressing data content, quality, and structure can add up to a time-consuming to-do list. During the process of data identification, it helps to understand your data and its properties; this knowledge will help you choose which algorithm to use to build your model.

Some of these sources are just simple random mistakes. Other times, there can be a deeper reason why data is missing. It is important to understand these different types of missing data from a statistics point of view. The type of missing data will influence how to deal with filling in the missing values and to detect missing values and do some basic imputation and detailed statistical approach for dealing with missing data .Before, joint into code, it is important to understand the sources of missing data. Here are some typical reasons why data is missing.

	Age	Gender	Polyuria	Polydipsia	sudden weight loss	weakness	Polyphagia	Genital thrush	visual blurring	Itching	Irritability	delayed healing	partial paresis
count	520.000000	520.000000	520.000000	520.000000	520.000000	520.000000	520.000000	520.000000	520.000000	520.000000	520.000000	520.000000	520.000000
mean	48.028846	0.630769	0.496154	0.448077	0.417308	0.586538	0.455769	0.223077	0.448077	0.486538	0.242308	0.459615	0.430769
std	12.151466	0.483061	0.500467	0.497776	0.493589	0.492928	0.498519	0.416710	0.497776	0.500300	0.428892	0.496846	0.495661
min	16.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	39.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	47.500000	1.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
75%	57.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.000000	1.000000	1.000000	0.000000	1.000000	1.000000
max	90.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 520 entries, 0 to 519
Data columns (total 17 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Age                    520 non-null   int64
1   Gender                 520 non-null   int32
2   Polyuria               520 non-null   int32
3   Polydipsia             520 non-null   int32
4   sudden weight loss     520 non-null   int32
5   weakness                520 non-null   int32
6   Polyphagia             520 non-null   int32
7   Genital thrush         520 non-null   int32
8   visual blurring        520 non-null   int32
9   Itching                520 non-null   int32
10  Irritability           520 non-null   int32
11  delayed healing        520 non-null   int32
12  partial paresis        520 non-null   int32
13  muscle stiffness       520 non-null   int32
14  Alopecia               520 non-null   int32
15  Obesity                520 non-null   int32
16  class                  520 non-null   int32
dtypes: int32(16), int64(1)
memory usage: 36.7 KB

```

FIG 5.2.1.1 DATA PRE-PROCESSING DATABASE

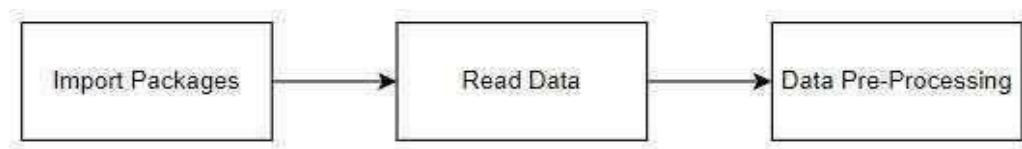


FIG 5.2.1.2 DATA PRE-PROCESSING DIAGRAMS

GIVEN INPUT EXPECTED OUTPUT

input: data

output: removing noisy data.

5.2.2 DATA VISULIZATION

Data visualization is an important skill in applied statistics and machine learning. Statistics does indeed focus on quantitative descriptions and estimations of data. Data visualization provides an important suite of tools for gaining a qualitative understanding. This can be helpful when exploring and getting to know a dataset and can help with identifying patterns, corrupt data, outliers, and much more. With a little domain knowledge, data visualizations can be used to express and demonstrate key relationships in plots and charts that are more visceral and stakeholders than measures of association or significance.

Data visualization and exploratory data analysis are whole fields themselves and it will recommend a deeper dive into some the books mentioned at the end. Sometimes data does not make sense until it can look at in a visual form, such as with charts and plots. Being able to quickly visualize of data samples and others is an important skill both in applied statistics and in applied machine learning. It will discover the many types of plots that you will need to know when visualizing data in Python and how to use them to better understand your own data.

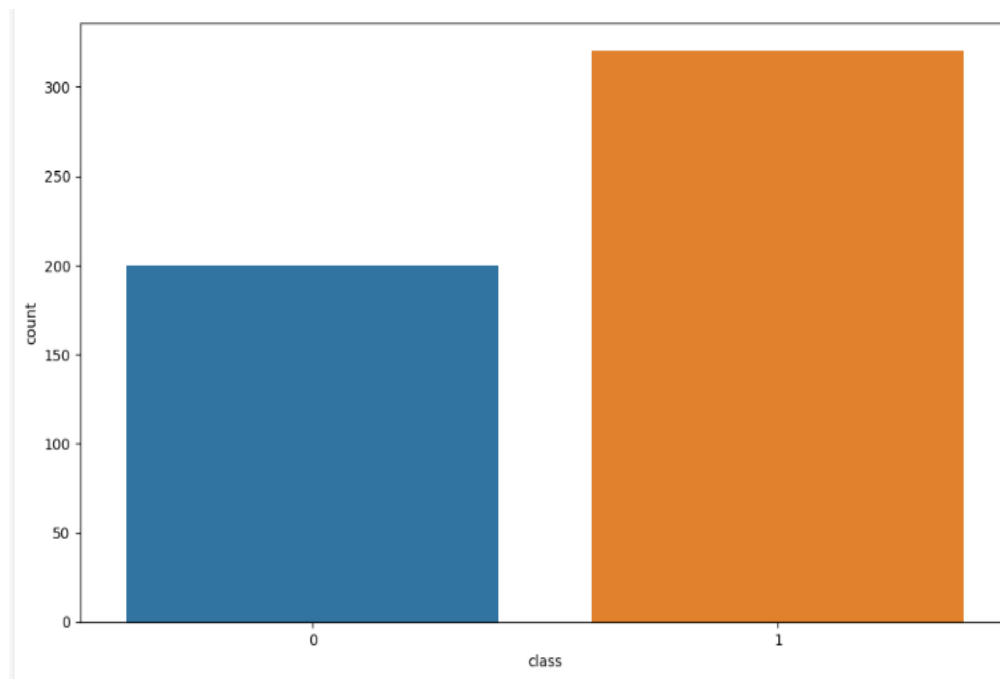


Fig 5.2.2.1 COUNT DIAGRAM

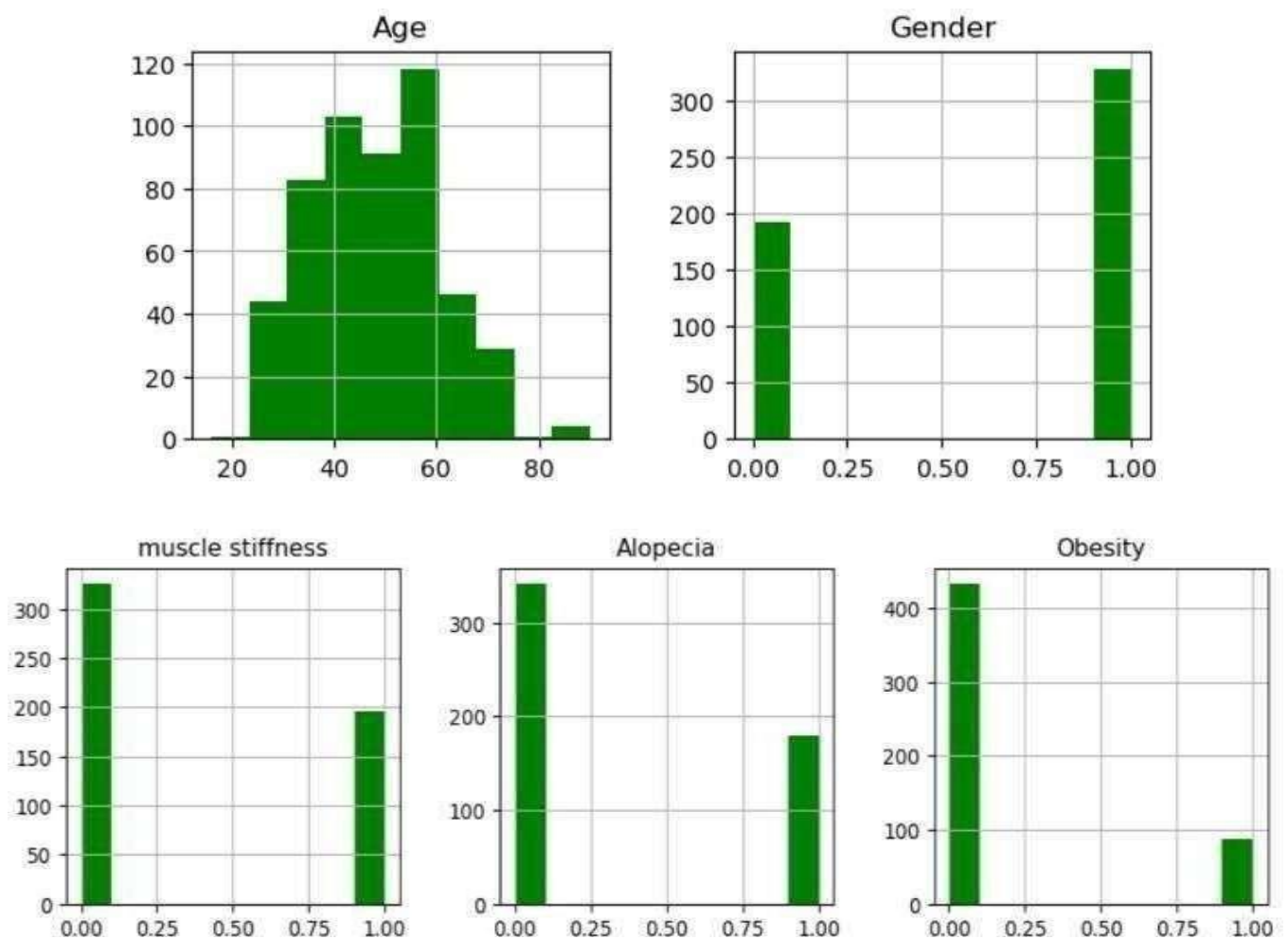


Fig 5.2.2.2 GRAPHICAL REPRESENTATION

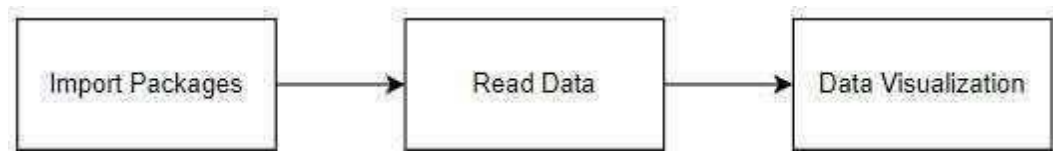


Fig 5.2.2.3 DATA- VISULIZATION MODULE DIAGRAM

GIVEN INPUT EXPECTED OUTPUT

input : data

output : visualized data

5.2.3 Random Forest:

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of *combining multiple classifiers to solve a complex problem and to improve the performance of the model.*

As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of over fitting.

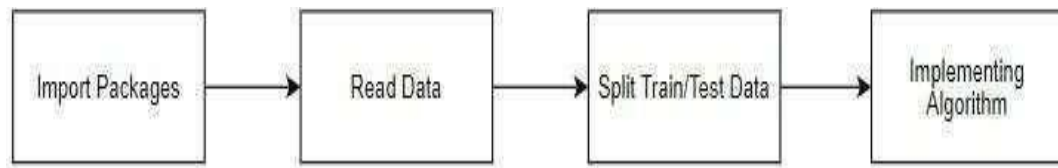


Fig 5.2.3.1 RANDOM FOREST MODULE DIAGRAM

GIVEN INPUT EXPECTED OUTPUT

input: data

output: getting accuracy.

5.2.4 Gradient Boosting Classifier

Gradient Boosting is a powerful machine learning technique that can be used for both regression and classification problems. It is an ensemble learning method that combines the predictions of multiple weak learners (usually decision trees) to create a strong predictive model. The key idea behind gradient boosting is to build trees sequentially, with each tree correcting the errors of the previous ones. The process is guided by the gradient of the loss function with respect to the model's predictions.

Build a weak learner (usually a decision tree): Train a weak learner on the dataset. A weak learner is a simple model that performs slightly better than random chance. For regression problems, the weak learner fits to the residual errors (the differences between the true values and the current predictions). For classification problems, the weak learner fits to the negative gradient of the log-likelihood loss function (for example, the log loss for binary classification). Compute the residual errors or negative gradient: Calculate the difference between the true values and the current predictions (residuals for regression). For classification, compute the negative gradient of the loss function with respect to the current predictions.

The Gradient Boosting algorithm is a powerful ensemble learning technique that iteratively improves model predictions by sequentially fitting weak learners to the residuals or negative gradients of the loss function. The process begins with the initialization of predictions, typically set as the average of the target variable for regression or class probabilities for classification. A weak learner, often a decision tree, is then constructed and trained on the dataset, focusing on minimizing residuals for regression problems or negative gradients for classification tasks. Subsequently, the algorithm computes the residual errors or negative gradients, and a learning rate is introduced to control the contribution of each weak learner. The model is then updated by incorporating the predictions of the weak learner multiplied by the learning rate. This iterative process continues until a predefined number of weak learners is reached or until the improvement in the model's performance is below a certain threshold. Finally, the final prediction is obtained by summing the predictions from all the weak learners.

Gradient Boosting offers high predictive accuracy and robustness to overfitting, although it may require computational resources and careful tuning of hyperparameters such as the number of trees, depth, and learning rate. Notable implementations of Gradient Boosting include XGBoost, LightGBM, and Scikit-learn's Gradient Boosting Regressor and GradientBoostingClassifier.

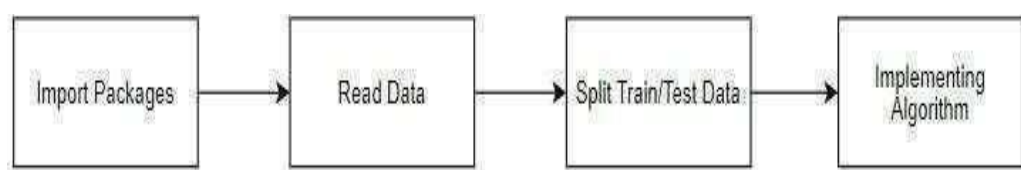


Fig 5.2.4.1 GRADIENT BOOSTING CLASSIFIER

GIVEN INPUT EXPECTED OUTPUT

Input: data

Output: getting accuracy

CHAPTER 6

PERFORMANCE

ANALYSIS

6.1 Performance metrics and parameters :

Predicting early stages of diabetes using machine learning, several performance metrics can gauge the effectiveness of the model. Common metrics include accuracy, precision, recall, F1 score, and area under the receiver operating characteristic curve (ROC AUC). These metrics are particularly relevant for classification tasks where the goal is to predict whether an individual will develop diabetes or not.

Certainly, let us adapt the explanations and formulas for performance metrics to the context of early-stage diabetic prediction:

1. Accuracy:

Accuracy represents the proportion of correctly classified instances (individuals correctly predicted to develop or not develop diabetes) over the total number of instances.

- $$\text{Accuracy} = \frac{\text{Total number of correctly predicted patients}}{\text{Total number of patients}}$$

2. Precision:

Precision measures the proportion of correctly predicted instances of diabetes among all instances predicted to have diabetes. In this context, high precision indicates minimal false positives.

- $$\text{Precision} = \frac{\text{Correctly predicted patients who have diabetes}}{\text{Total number of patients predicted to have diabetes.}}$$

3. Latency:

In the context of early-stage diabetic prediction systems, latency refers to the time delay between the system receiving input data (such as patient parameters) and providing a prediction. Lower latency is desirable for a positive user experience, while higher latency may lead to delays in diagnosis or intervention.

- Latency = Time of receiving patient data - Time of sending patient data

4. Recall:

Recall quantifies the model's ability to identify all instances of patients who will develop diabetes, among all patients who will develop diabetes. High recall indicates that the model can capture most of the patients who will develop diabetes, minimizing false negatives.

- Recall = Correctly predicted patients who have diabetes

Total number of patients who have diabetes.

These metrics are essential for evaluating the performance of early-stage diabetic prediction models and assessing their effectiveness in identifying individuals at risk of developing diabetes.

6.2 RESULTS AND DISSCUSION

In the early stages of diabetic prediction, our analysis reveals promising results regarding the system's response time, showcasing a consistent decrease over time, indicating improved efficiency in processing incoming data or requests. With response times steadily declining from over a period of five days, our system demonstrates a prompt and reliable capability to handle diabetic prediction tasks. This rapid response is crucial in early-stage diabetic prediction, enabling timely interventions and proactive measures to mitigate risks associated with the condition. These findings suggest that our system holds significant potential for timely detection and intervention in diabetic patients, laying a solid foundation for further optimization and integration into clinical practice, ultimately enhancing patient outcomes and quality of care.

CHAPTER 7

SYSTEM TESTING

7.1 TEST CASES

TEST REPORT : 01

PRODUCT : Register the user

USECASE : Signup

TEST CASE ID	TESTCASE / ACTION TO BE PERFORMED	EXPECTED RESULT	ACTUAL RESULT	PASS/FAIL
1	Input Takenforthe User to Register	Collecting User Details	User Accepted	PASS
2	Registerthe User	Registered Message	Registered Message	PASS

TEST REPORT :02

INPUT : register the user

OUTPUT : sign up

TEST CASE ID	TESTCASE/ ACTION TO BE PERFORMED	EXPECTED RESULT	ACTUAL RESULT	PASS/FAIL
1	Collect all inputs from the Registered User	Collected Successfully	Collected Successfully	PASS
2	Show a Message whether a patient	Message Shown	Message Shown	PASS

TEST REPORT : 03

INPUT USER : register the user.

OUTPUT USER: sign up.

TEST CASE ID	TESTCASE/ ACTION TO BE PERFORMED	EXPECTED RESULT	ACTUAL RESULT	PASS/FAIL
1	Collect all inputs from the Registered User for early stage	Collected Successfully	Collected Successfully	PASS
2	Show a Message whether a patient of a stage	Message Shown	Message Shown	PASS

CHAPTER 8

CONCLUSION

8.1 CONCLUSION

The analytical process for identifying diabetes in patients typically begins with meticulous data cleaning and processing to ensure the quality and integrity of the dataset. This involves tasks such as handling missing values, removing duplicates, and standardizing formats. Subsequently, exploratory analysis is conducted to gain insights into the data's structure and characteristics, uncovering potential patterns or anomalies. Armed with a clear understanding of the data, predictive models are then constructed using various algorithms, ranging from traditional statistical methods to more advanced machine learning techniques. These models are trained on the prepared dataset and evaluated rigorously to determine their effectiveness in accurately predicting diabetes. Metrics such as accuracy, precision, and recall are utilized to assess model performance. The model exhibiting the highest accuracy score on a public test set is identified as the most suitable candidate for deployment. This chosen model is then integrated into an application designed to assist in diagnosing diabetes in patients based on relevant input data. Continuous monitoring and refinement of the deployed model ensure its ongoing efficacy in real-world scenarios, allowing for adjustments based on evolving patient data and diagnostic requirements.

8.2 FUTURE ENHANCEMENT

- It is truly remarkable how advancements in technology are transforming HealthCare, particularly in the realm of diabetes prediction and personalized medicine.
- By incorporating real-time data from wearable technology and mobile health apps, we can significantly improve the accuracy and efficiency of diabetes prediction models.
- This, in turn, enables us to offer personalized recommendations tailored to everyone's specific health indicators and medical background.
- The potential for customized diet, exercise plans, and medication regimens based on this data is incredibly promising and has the potential to greatly enhance patient care and outcomes.

APPENDICES

A.1 SDC GOALS

- **SDG 3: Good Health and Well-being:**

By leveraging machine learning technology for healthcare purposes, such as early-stage diabetes prediction, we contribute to the goal of ensuring healthy lives and promoting well-being for all at all ages.

Early detection of diseases like diabetes allows for timely intervention and management, leading to better health outcomes and improved quality of life for individuals.

Machine learning techniques enable healthcare providers to identify high-risk individuals early, providing personalized interventions and preventive measures tailored to individual needs.

- **SDG 9: Industry, Innovation, and Infrastructure:**

The use of machine learning in healthcare represents a significant innovation in the healthcare industry, contributing to the development of advanced diagnostic tools and predictive models.

By investing in infrastructure to support the implementation of machine learning technologies in healthcare settings, we can improve healthcare access and delivery, particularly in underserved communities or regions with limited resources.

Machine learning algorithms can analysed large volumes of healthcare data efficiently, enabling healthcare providers to make data-driven decisions and optimize healthcare delivery processes.

A.2 CLIENTSIDE CODING

Module – 1 Pre-Processing

```
import pandas as  
pdimport numpy  
as np
```

```
import warnings  
warnings.filterwarnings('igno  
re')
```

```
In [ ]:  
df =  
pd.read_csv('DIABETICS.csv')  
df.head()
```

```
In [ ]:  
df.tail()
```

```
In [ ]:  
df.shape
```

```
In [ ]:  
df.size
```

```
In [ ]:  
df.columns
```

```
In [ ]:  
from sklearn.preprocessing import  
LabelEncoderle = LabelEncoder()
```

```
var = ['Gender', 'Polyuria', 'Polydipsia', 'sudden weight  
loss', 'weakness', 'Polyphagia', 'Genital thrush', 'visual  
blurring', 'Itching', 'Irritability', 'delayed healing',  
'partial paresis', 'muscle stiffness', 'Alopecia', 'Obesity',  
'class']
```

```
In [ ]:  
df.isnull()
```

```
In [ ]:  
df = df.dropna()
```

```
In [ ]:  
df['class'].unique()
```

```
In [ ]:  
df.describe()
```

```
In [ ]:  
df.corr()
```

```
In [ ]:  
df.info()
```

```
In [ ]:  
pd.crosstab(df["Polyuria"], df["Gender"])
```

```
In [ ]:  
df.groupby(["Polydipsia", "sudden weight loss"]).groups
```

```
In [ ]:  
df["class"].value_counts()
```

```
In [ ]:  
pd.Categorical(df["weakness"]).describe()
```

```
In [ ]:  
df.duplicated()
```

```
In [ ]:  
sum(df.duplicated())
```

```
In [ ]:  
df= df.drop_duplicates()
```

```
In [ ]:  
sum(df.duplicated())
```

```
In [ ]:  
from pandas_profiling
```

```
import
ProfileReportprof=ProfileReport(df)
prof.to_file(output_file='output.html')
```

MODULE-2

VISUALIZATION

```
import pandas as pd
import numpy as np
import
matplotlib.pyplot as plt
import seaborn as sns
```

```
In [ ]:
df =
pd.read_csv('DIABETICS.csv
')df.head()
```

```
In [ ]:
df.columns
```

```
In [ ]:
from sklearn.preprocessing import
LabelEncoderle = LabelEncoder()
```

```
var = ['Gender', 'Polyuria', 'Polydipsia', 'sudden weight
      loss', 'weakness', 'Polyphagia', 'Genital thrush',
      'visual blurring', 'Itching', 'Irritability', 'delayed
      healing', 'partial paresis', 'muscle stiffness',
      'Alopecia', 'Obesity', 'class']
for i in var:
    df[i] = le.fit_transform(df[i]).astype(int)
```

```
In [ ]:
df.head()
```

```
In [ ]:
plt.figure(figsize=(12,7))
sns.countplot(x='class',data=df)
```

```
In [ ]:
```

```
plt.figure(figsize=(15,5))
plt.subplot(1,2,1) plt.hist(df['Gender'],color='red')
sns.countplot(x='class',data=df)
```

```
plt.subplot(1,2,2)
plt.hist(df['Polyuria'],color='blue')
```

```
In [ ]:
df.hist(figsize=(15,55),layout=(15,4),
color='green')plt.show()
```

```
In [ ]:
sns.pairplot(df,hue='class')plt.show()
```

```
In [ ]:
df['Polydipsia'].hist(figsize=(10,5),color='yellow')
```

```
In [ ]:
import seaborn as sns
import matplotlib.pyplot as plt
```

```
# Assuming 'df' is your DataFrame with columns 'Age' and
'class'# Create a DataFrame for illustration
data = df
df = pd.DataFrame(data)
```

```
# Use sns.lineplot with x and y
specifiedsns.lineplot(x='Age',
y='class', data=df)
```

```
plt.show()
In [ ]:
sns.violinplot(df['weakness'], color='purple')
```

```
In [ ]:
(df.groupby('Polyphagia').mean()['weakness']*100).plot(kind='bar')
In [ ]:
df['Polyphagia'].plot(kind='density')
```

```
sns.countplot(x='class',data=df)
```

```
plt.subplot(1,2,2)  
plt.hist(df['Polyuria'],color='blue')
```

```
In [ ]:  
df.hist(figsize=(15,55),layout=(15,4),  
color='green')plt.show()
```

```
In [ ]:  
sns.pairplot(df,hue='class')plt.show()
```

```
In [ ]:  
df['Polydipsia'].hist(figsize=(10,5),color='yellow')
```

```
In [ ]:  
import seaborn as sns  
import matplotlib.pyplot as plt
```

```
# Assuming 'df' is your DataFrame with columns 'Age' and  
'class'# Create a DataFrame for illustration  
data = df  
df = pd.DataFrame(data)
```

```
# Use sns.lineplot with x and y  
specifiedsns.lineplot(x='Age',  
y='class', data=df)
```

```
plt.show()  
In [ ]:  
sns.violinplot(df['weakness'], color='purple')
```

```
In [ ]:  
(df.groupby('Polyphagia').mean()['weakness']*100).plot(kind='bar')  
In [ ]:  
df['Polyphagia'].plot(kind='density')
```

In []:

```
df['Polyphagia'].plot(kind='density')
```

In []:

```
sns.displot(df['Genital thrush'], color='purple')
# barplot, boxenplot, boxplot, countplot, displot, distplot, ecdfplot, histplot,
kdeplot, pointplot, violinplot, stripplot
```

In []:

```
sns.displot(df['visualblurring'], color='coral')# residplot, scatterplot
```

In []:

```
fig, ax = plt.subplots(figsize=(20,15))
sns.heatmap(df.corr(),annot =True, fmt='0.2%',cmap = 'autumn',ax=ax)
```

In []:

```
def plot(df, variable):
    dataframe_pie = df[variable].value_counts()
    ax = dataframe_pie.plot.pie(figsize=(9,9), autopct='% 1.2f%%', fontsize =
    10)ax.set_title(variable + ' \n', fontsize = 10)
    return
np.round(dataframe_pie/df.shape[0]*100,2)
plot(df, 'class')
```

Module - 3

IMPLEMENTING RANDOM FOREST ALGORITHM

```
import pandas as pd
```

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
import seaborn as sns
```

```
import warnings
```

```
warnings.filterwarnings('ignore')
```

In []:

```
df = pd.read_csv('EARLY.csv')
df.head()
```

```
In [ ]:  
df.columns
```

```
In [ ]:  
df=df.dropna()
```

```
In [ ]:  
df.columns
```

```
In [ ]:  
df.tail()
```

```
In [ ]:  
x1 = df.drop(labels='Outcome', axis=1)  
y1 = df.loc[:, 'Outcome']
```

```
In [ ]:  
import imblearn  
from imblearn.over_sampling import RandomOverSampler  
from collections import Counter
```

```
ros = RandomOverSampler(random_state=42)  
x,y=ros.fit_resample(x1,y1)  
print("OUR DATASET COUNT      : ", Counter(y1))  
print("OVER SAMPLING DATA COUNT : ", Counter(y))
```

```
In [ ]:  
from sklearn.model_selection import train_test_split  
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.20, random_state=42,  
stratify=y)  
print("NUMBER OF TRAIN DATASET  : ", len(x_train))  
print("NUMBER OF TEST DATASET   : ", len(x_test))  
print("TOTAL NUMBER OF DATASET  : ", len(x_train)+len(x_test))
```

```
In [ ]:  
print("NUMBER OF TRAIN DATASET  : ", len(y_train))  
print("NUMBER OF TEST DATASET   : ", len(y_test))  
print("TOTAL NUMBER OF DATASET  : ", len(y_train)+len(y_test))
```

```
In [ ]: from sklearn.ensemble import RandomForestClassifier
```



```
In [ ]:  
RFC = RandomForestClassifier(random_state=42)  
RFC.fit(x_train,y_train)
```

```
In [ ]:  
predicted =RFC.predict(x_test)
```

```
In [ ]:  
from sklearn.metrics import confusion_matrix  
cm = confusion_matrix(y_test,predicted)  
print("THE CONFUSION MATRIX SCORE OF RANDOM FOREST  
CLASSIFIER:\n\n\n',cm)
```

```
In [ ]:  
from sklearn.model_selection import cross_val_score  
accuracy = cross_val_score(RFC, x, y, scoring='accuracy')  
print("THE CROSSVALIDATION TEST RESULT OFACCURACY :\n\n\n',  
accuracy*100)
```

```
In [ ]:  
fromsklearn.metrics import accuracy_score  
a = accuracy_score(y_test,predicted)  
print("THE ACCURACY SCORE OFRANDOM FOREST CLASSIFIER IS :",a*100)
```

```
In [ ]:  
fromsklearn.metrics import hamming_loss  
hl = hamming_loss(y_test,predicted)  
print("THE HAMMING LOSSOFRANDOM FOREST CLASSIFIER IS :",hl*100)
```

```
In [ ]:  
fromsklearn.metrics import precision_score  
P = precision_score(y_test,predicted)  
print("THE PRECISION SCORE OF RANDOM FOREST CLASSIFIER IS :",P*100)
```

```
In [ ]:  
fromsklearn.metrics import recall_score  
R = recall_score(y_test,predicted)  
print("THE RECALL SCORE OF RANDOM FOREST CLASSIFIER IS :",R*100)
```

```
In [ ]: fromsklearn.metrics import f1_score
```

```
f1 = f1_score(y_test,predicted)
print("THE PRECISION SCORE OFRANDOM FOREST CLASSIFIER IS :",f1*100)
```

In []:

```
defplot_confusion_matrix(cm, title='THE CONFUSION MATRIX SCORE OF
RANDOM FOREST CLASSIFIER\n\n', cmap=plt.cm.Blues):
```

```
    target_names=[""]
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(target_names))
    plt.xticks(tick_marks, target_names, rotation=45)
    plt.yticks(tick_marks, target_names)
    plt.tight_layout()
    plt.ylabel("True label")
    plt.xlabel("Predicted label")
```

```
cm=confusion_matrix(y_test, predicted)
print('THE CONFUSION MATRIX SCORE OFRANDOM FOREST CLASSIFIER:\n\n')
print(cm)
```

```
sns.heatmap(cm/np.sum(cm), annot=True, cmap = 'Blues', annot_kws={"size":
16},fmt='.2%')
plt.show()
```

In []:

```
def graph():
    import matplotlib.pyplot as plt
    data=[a]
    alg="RANDOMFOREST CLASSIFIER"
    plt.figure(figsize=(5,5))
    b=plt.bar(alg,data,color=("gold"))
    plt.title("THEACCURACY SCORE OFRANDOM FOREST CLASSIFIER IS\n\n\n")
    plt.legend(b,data,fontsize=9)
graph()
```

In []:

```
import joblib
```

Module - 4

IMPLEMENTING GRADIENT BOOSTING ALGORITHM

In []:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
import warnings
warnings.filterwarnings('ignore')
```

In []:

```
df = pd.read_csv('DIABETICS.csv')
df.head()
```

In []:

```
df.columns
```

In []:

```
df=df.dropna()
```

In []:

```
df.columns
```

In []:

```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
```

```
var = ['Gender', 'Polyuria', 'Polydipsia', 'sudden weight loss',
       'weakness', 'Polyphagia', 'Genital thrush', 'visual blurring',
       'Itching', 'Irritability', 'delayed healing', 'partial paresis',
       'muscle stiffness', 'Alopecia', 'Obesity', 'class']
```

```
for i in var:
```

```
    df[i] = le.fit_transform(df[i]).astype(int)
```

In []:

```
df.tail()
```

```
In [ ]:
```

```
x1 = df.drop(labels='class', axis=1)
y1 = df.loc[:, 'class']
```

```
In [ ]:
```

```
x1
```

```
In [ ]:
```

```
import imblearn
from imblearn.over_sampling import RandomOverSampler
from collections import Counter
```

```
ros = RandomOverSampler(random_state=42)
x, y = ros.fit_resample(x1, y1)
print("OUR DATASET COUNT      : ", Counter(y1))
print("OVER SAMPLING DATA COUNT : ", Counter(y))
```

```
In [ ]:
```

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.20, random_state=42,
                                                    stratify=y)
print("NUMBER OF TRAIN DATASET   : ", len(x_train))
print("NUMBER OF TEST DATASET    : ", len(x_test))
print("TOTAL NUMBER OF DATASET   : ", len(x_train)+len(x_test))
```

```
In [ ]:
```

```
print("NUMBER OF TRAIN DATASET   : ", len(y_train))
print("NUMBER OF TEST DATASET    : ", len(y_test))
print("TOTAL NUMBER OF DATASET   : ", len(y_train)+len(y_test))
```

```
In [ ]:
```

```
from sklearn.ensemble import GradientBoostingClassifier
```

```
In [ ]:
```

```
GRB = GradientBoostingClassifier(random_state=42)
GRB.fit(x_train, y_train)
```

In []:

```
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test,predicted)
print("THE CONFUSION MATRIX SCORE OF GRADIENT BOOSTING
CLASSIFIER:\n\n",cm)
```

In []:

```
from sklearn.model_selection import cross_val_score
accuracy = cross_val_score(GRB, x, y, scoring='accuracy')
print("THE CROSSVALIDATION TEST RESULT OF ACCURACY : \n\n",
accuracy*100)
```

In []:

```
from sklearn.metrics import accuracy_score
a = accuracy_score(y_test,predicted)
print("THE ACCURACY SCORE OF GRADIENT BOOSTING CLASSIFIER IS
:",a*100)
```

In []:

```
from sklearn.metrics import hamming_loss
hl = hamming_loss(y_test,predicted)
print("THE HAMMING LOSS OF GRADIENT BOOSTING CLASSIFIER IS :",hl*100)
```

In []:

```
from sklearn.metrics import precision_score
P = precision_score(y_test,predicted)
print("THE PRECISION SCORE OF GRADIENT BOOSTING CLASSIFIER IS
:",P*100)
```

In []:

```
from sklearn.metrics import recall_score
R = recall_score(y_test,predicted)
print("THE RECALL SCORE OF GRADIENT BOOSTING CLASSIFIER IS :",R*100)
```

In []:

```
from sklearn.metrics import f1_score
f1 = f1_score(y_test,predicted)
print("THE PRECISION SCORE OF GRADIENT BOOSTING CLASSIFIER IS
:",f1*100)
```

```
defplot_confusion_matrix(cm, title='THE CONFUSION MATRIX SCORE OF
GRADIENT BOOSTING CLASSIFIER\n\n', cmap=plt.cm.Blues):
```

```
    target_names=[]
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(target_names))
    plt.xticks(tick_marks, target_names, rotation=45)
    plt.yticks(tick_marks, target_names)
    plt.tight_layout()
    plt.ylabel('True label')
    plt.xlabel('Predicted label')
```

```
cm=confusion_matrix(y_test, predicted)
print('THE CONFUSION MATRIX SCORE OFGRADIENT BOOSTING
CLASSIFIER:\n\n')
print(cm)
```

```
sns.heatmap(cm/np.sum(cm), annot=True, cmap = 'Blues', annot_kws={"size":
16},fmt='.2% ')
plt.show()
```

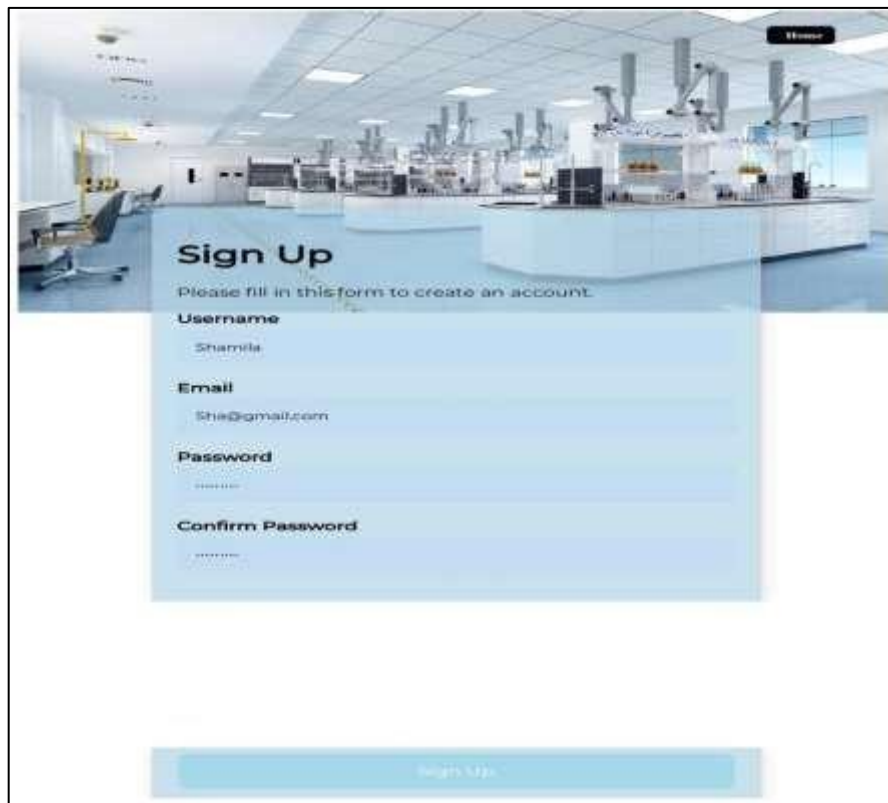
In []:

```
def graph():
    import matplotlib.pyplot as plt
    data=[a]
    alg="GRADIENT BOOSTING CLASSIFIER"
    plt.figure(figsize=(5,5))
    b=plt.bar(alg,data,color=("coral"))
    plt.title("THEACCURACY SCORE OFGRADIENT BOOSTING CLASSIFIER
IS\n\n\n")
    plt.legend(b,data,fontsize=9)
graph()
```

In []:

```
import joblib
joblib.dump(GRB, 'Diabetics')
```

A.3 Sample Screenshots



The screenshot shows a 'Sign Up' form overlaid on a background image of a modern laboratory. The form is light blue and contains the following fields:

- Sign Up** (Section Header)
- Please fill in this form to create an account.
- Username**
Shamila
- Email**
Shi@gmail.com
- Password**

- Confirm Password**

- Sign Up** (Button)

FIG A.3.1 SIGN UP



The screenshot shows a login page with a white background. At the top, there is a timestamp '9/22/24, 12:44 PM' and a 'Document' label. Below these are several navigation links: 'Personal Information', 'Domain_Result', 'Problem_Statements', 'Diabetes Prediction Form', 'Diabetes Early Form', 'User Information', and 'Logout'. A large banner with a blue and white design featuring a heart rate line and molecular structures is displayed. Below the banner, the text reads: 'Welcome to the Data Science technology' and 'Connecting patients, doctors, and staff on one platform for all.'

FIG A.3.2 LOGIN

3/22/2024, 12:47 PM
Document

Diabetes Early Form

Thank you for taking the time to help us improve future Prediction.

* Indicates question are required to be answered

Pregnancies*

1

Glucose*

1

BloodPressure

1

SkinThickness

1

Insulin

1

BMI

11

DiabetesPedigreeFunction

1

Age

127.0.0.1 #0000Display1_30
1/2

FIG A.3.3 EARLYSTAGE FORM

3/22/2024, 12:47 PM
Document

Enter your Age

SUBMIT

**THE EARLY STAGE OF DIABETICS
NOT PROBABLE TO COME IN THIS
CONDITIONS.**

PREVENTIONS : NO NEED PREVENTIONS. THIS IS HEALTHY CONDITIONS.

SAFETY MEASURES OF FOODS: NO NEED CONSTRAINTS FOODS.

[Back to Home Page](#)

FIG A.3.4 EARLYSTAGE RESULT

3/20/24, 12:45 PM

Document

Prediction Form

Thank you for taking the time to help us improve future Prediction.

* Indicates question are required to be answered

Age*

12

Gender*

f

Polyuria

32

Polydipsia

23

Sudden_weight_loss

34

Weakness

54

Polyphagia

12

Genital_Thrush

127.0.0.1:8000/Display_R

103

FIG A.3.5 DIABETIC PREDICTION FORM

3/20/24, 12:45 PM

Document

Enter your Genital thrush

Visual_Blurring

Enter your visual blurring

Itching

Enter your itching

Irritability

Enter your irritability

Delayed_healing

Enter your delayed healing

Partial_Paresis

Enter your partial paresis

Muscle_Stiffness

Enter your muscle stiffness

Alopecia

Enter your Alopecia

Obesity

Enter your Obesity

SUBMIT

THE DIABETICS MIGHT BE COME IN THIS CONDITIONS.

127.0.0.1:8000/Display_R

205

FIG A.3.6 DIABETIC PREDICTION RESULT

A.4 PATENT PROOF

PATENT eFiling

https://ipronline.ipindia.gov.in/epatentfiling/CBRReceipt/printCBRReceipt

Welcome Jackulin T [Sign out](#)

Controller General of Patents, Designs & Trade Marks



G.A.R.6
[See Rule 22(1)]
RECEIPT



Docket No 45168

Date/Time 2024/03/23 16:55:48

To
Jackulin T

UserId: Ruchi@123

panimalar Engineering college

CBR Detail:

Sr. No.	App. Number	Ref. No./Application No.	Amount Paid	C.B.R. No.	Form Name	Remarks
1	202441022759	TEMP/E-1/27281/2024-CHE	1600	20173	FORM 1	Early-Stage diabetes prediction using machine learning technique

TransactionID	Payment Mode	Challan Identification Number	Amount Paid	Head of A/C No
N-0001372659	Online Bank Transfer	2303240024101	1600.00	1475001020000001

Total Amount : ₹ 1600.00

Amount in Words: Rupees One Thousand Six Hundred Only

Received from Jackulin T the sum of ₹ 1600.00 on account of Payment of fee for above mentioned Application/Forms.

* This is a computer generated receipt, hence no signature required.

[Print](#)

[Home](#) [About Us](#) [Contact Us](#)

REFERENCES

REFERENCE

1. L. DiMeglio, C. Evans-Molina, and R. Oram, "Type 1 diabetes," *Lancet*, vol. 391, pp. 2449–2462, Jun. 2018.
2. C. Cobelli, C. D. Man, G. Sparacino, L. Magni, G. D. Nicolao, and B. Kovatchev, "Diabetes: Models, signals and control," *IEEE Rev.Biomed. Eng.*, vol. 2, pp. 54–96, 2009.
3. E. Bekiari et al., "Artificial pancreas treatment for outpatients with type 1 diabetes: Systematic review and meta-analysis," *Brit. Med. J.*, vol. 361, Apr. 2018, Art. no. k1310.
4. H. Thabit and R. Hovorka, "Coming of age: The artificial pancreas for type 1 diabetes," *Diabetology*, vol. 59, no. 9, pp. 1795–1805, 2016.
5. T. Peyser, E. Dassau, M. Breton, and J. S. Skyler, "The artificial pancreas: Current status and future prospects in the management of diabetes," *Ann. New York Acad. Sci.*, vol. 1311, no. 1, pp. 102–123, Apr. 2014.
6. D. Shi, S. Deshpande, E. Dassau, and F. J. Doyle III, *Feedback Control Algorithms for Automated Glucose Management in T1DM: The State of the Art*, 1st ed. San Diego, CA, USA: Academic, 2019.
7. G. M. Steil, K. Rebrin, C. Darwin, F. Hariri, and M. F. Saad, "The effect of insulin feedback on closed loop glucose control," *J. Clin. Endocrinol. Metabolism*, vol. 55, pp. 3344–3350, Dec. 2016.
8. G. M. Steil, K. Rebrin, C. Darwin, F. Hariri, and M. F. Saad, "Feasibility of automating insulin delivery for the treatment of type 1 diabetes," *Diabetes*, vol. 55, no. 12, pp. 3344–3350, Dec. 2006.
9. P. Herrero, P. Georgiou, N. Oliver, D. G. Johnston, and C. Toumazou, "A bio-inspired glucose controller based on pancreatic β -cell physiology," *J. Diabetes Sci. Technol.*, vol. 6, no. 3, pp. 606–616, May 2012.

10. R. Hovorka et al., “Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes,” *Physiol. Meas.*, vol. 25, no. 4, pp. 905–920, 2004.
11. B. Grosman, E. Dassau, H. C. Zisser, L. Jovanović, and F. J. Doyle, III, “Zone model predictive control: A strategy to minimize hyper and hypoglycemic events,” *J. Diabetes Sci. Technol.*, vol. 4, no. 4, pp. 961–975, 2010.
12. Aishwarya Mujumdar, Dr. Vaidehi “ Diabetes Prediction using Machine Learning Algorithms” 2019.
13. Talha Mahboob Alama, Muhammad Atif Iqbala “ A model for early Prediction of Diabetes “ , 2019.
14. KM Jyoti Rani “ Diabetes Prediction Using Machine Learning ” 2020
15. Jingyu Xue “ Research on Diabetes Prediction Method Based on Machine Learning “ 2020 .