Chapter 1

Introduction

1.1 Background

Diabetes is a chronic disease that occurs either when the pancreas does not produce enough insulin or when the body cannot effectively use the insulin it produces. Insulin is a hormone that regulates blood glucose [1].Carbohydrate foods provide our body with its main energy source everybody, even those people with diabetes, needs carbohydrate.

According to (WHO) World Health Organization about 830 million people suffering from diabetes particularly from low or idle income countries. In 2022, 14% of adults aged 18 years and older were living with diabetes, an increase from 7% in 1990. In 2021, diabetes was the direct cause of1.6 million deaths and 47% of all deaths due to diabetes occurred before the age of 70 years. Since 2000, mortality rates from diabetes have been increasing [1].

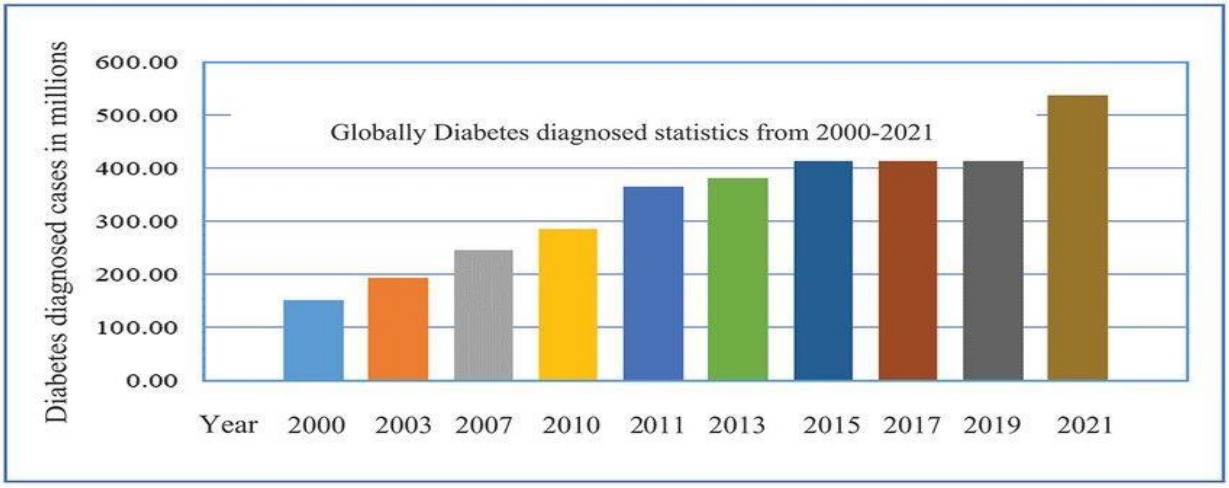


Figure 1.1: Diabetes Impact Statistics [29]

If we have type 1 diabetes, we body makes little or no insulin. we immune system attacks and destroys the cells in we pancreas that make insulin. Type 1 diabetes is usually diagnosed in children and young adults, although it can appear at any age. Type 2 diabetes is the most common form of diabetes. Type 2 diabetes has historically been diagnosed primarily in adults. But adolescents and young adults are developing Type 2 diabetes at an alarming rate because of family history and higher rates of obesity and physical inactivity. Gestational diabetes: This type develops in some people during pregnancy. Gestational diabetes usually goes away after pregnancy. However, if we have gestational diabetes, we're at a higher risk of developing Type 2 diabetes later in life[28]. Some of the symptoms of type 1 diabetes and type 2 diabetes are: Feeling more thirsty than usual, Urinating often, Losing weight, without trying, Feeling tired and weak, Having blurry vision.

Early detection of such disorders allows for disease control and the preservation of human life. This study focuses on the diabetes prediction using machine learning approaches in order to attain this goal.

Several diabetes prediction algorithms [7][11][13][17] have been proposed and published in recent years. They conducted thorough tests on outlier rejection and filling missing variables in order to improve the ML model's performance. They shows, the adoption of Machine Learning classification-based techniques in illness diagnosis and treatment can reduce medicalerrors and human costs dramatically. According to the findings of the study, When compared to other methods for data classification, machine learning-based classification techniques offer a promising performance in prediction accuracy.

This research focuses on developing a diabetes prediction model utilizing machine learning algorithms and data mining techniques. The National Institute of Diabetes and Digestive and Kidney Diseases' Pima Indians Diabetes Dataset, which contains information on female diabetic patients, was used in this study. In our proposed approach, different ML classifiers were used. Grid search technique is applied to improve accuracy. Various approaches of machine learning models are carried out in order to find the best classifier, which employs the best performing preprocessing from prior experiments.

1.2 Problem Statement

Diabetes is a chronic and complex disease, which is considered a major threat to public health worldwide. According to the International Diabetes Federation (IDF) report, the number of diabetic patients is increasing rapidly every year. If diabetes is not detected at an early stage, it can cause various complications including heart disease, kidney problems, nervous system damage and blindness. Diagnosing diabetes in the conventional medical system requires time-consuming and expensive tests, which in many cases makes it difficult to detect the disease at an early stage.

On the other hand, it is possible to predict the risk of diabetes using machine learning algorithms by analyzing the patient's physical parameters (such as age, weight, BMI, blood pressure, glucose levels, insulin, etc.) and lifestyle information. This can play an important role in determining the risk at an early stage and providing timely treatment in the healthcare sector.

Therefore, developing an effective and accurate machine learning-based diabetes prediction model is an important challenge of the present time. The objective of this research is to develop a prediction model using various machine learning algorithms that will be able to quickly and accurately identify the risk of diabetes.

1.3 Motivation

The main motivation for this research is to find a way to detect silent and rapidly growing diseases like diabetes at an early stage. Diabetes currently affects billions of people worldwide and complications and medical costs are increasing due to not being detected at an early stage (World Health Organization, 2023). Machine learning is able to determine the probability of the disease in advance by analyzing various risk factors such as age, glucose levels, BMI, lifestyle and family history together.

Despite the limitations of conventional screening methods, machine learning helps doctors quickly and accurately identify high-risk individuals. The availability of large datasets makes it possible to create accurate models that can play an important role in detecting diseases at an early stage, reducing complications and medical costs, and increasing public awareness. As a result, this research can make a significant contribution to the prevention and control of diabetes.

1.4 Research Objectives

The main objective of this project is to provide early and accurate prediction of diabetes using machine learning models. This will help patients to be informed about their risk in a timely manner and help them take preventive measures.
Specific Objectives:

- To develop a robust machine learning model using PIMA Indian Diabetes Dataset that can accurately predict the presence or risk of diabetes.
- To identify the best performing model by comparing different machine learning algorithms.
- To improve the accuracy of the model.
- To identify the key risk factors associated with diabetes that have the greatest impact on the prognosis of the disease.

1.5 Thesis Organization

The organization of our thesis is given below:

Chapter 1(Introduction): This opening chapter provides a comprehensive introduction to the thesis. The relevance of diabetes prediction using machine learning is emphasized. The chapter delineates the research problem and application areas and outlines the significant contributions and motivations driving the study. Moreover, it articulates the objectives of the thesis, setting the stage for the subsequent chapters.

Chapter 2 (Related Work): This chapter delves into an extensive review of related works in sentiment analysis. Existing research papers, methodologies, and techniques relevant to diabetes prediction using machine learning are critically examined. The literature review serves as the foundation for the proposed method and provides a contextual framework for the study.

Chapter 3 (Proposed Methodology): It provides a detailed overview of the techniques employed in developing the diabetes prediction system. This chapter offers a comprehensive insight into the research methodology, from data collection and preprocessing to feature extraction and the selection of machine learning.

Chapter 4 (Results and Analysis): Presents the experimental results and analysis of the proposed methodology. It includes performance evaluation metrics such as accuracy, precision, recall, FI-score, confusion matrix, ROC curve. The chapter discusses the effectiveness of the proposed models and compares their results.

Chapter 5 (Conclusion and future works): Summarizes the key findings of the research, discusses its contributions, and provides recommendations for future work in diabetes prediction using machine learning.

Chapter 2

Related Works

2.1 Introduction

Diabetes prediction using machine learning has emerged as an important research area in recent years. Diabetes is a chronic disease that can lead to serious complications if not detected early. Using medical data, clinical records, and lifestyle information, researchers have applied various ML algorithms to determine the risk of diabetes. These studies have attempted to improve the accuracy of prediction using feature selection, classification models. As a result, models based on traditional diagnosis methods can play an effective role in early detection and preventive measures. This section reviews the existing literature on diabetes risk prediction, highlighting the methods, datasets, and models used in previous studies, which serve as the basis for our study. Here, we are show some related work.

2.2 Comparison of Machine Learning Models for Predicting Type 2 Diabetes Risk Using the Pima Indians Diabetes Dataset.

This study evaluated the predictive performance of four machine learning models using the PIMA Indian data set. Among them, the accuracy of 85%, sensitivity of 79%, and AUC-ROC of 91%, values of XG-Boost are superior to other models. Analyzing the importance of attributes, the three most relevant attributes were glucose level, BMI and age, well established from clinical knowledge. Hyperparameter Tuning grid search and 5-fold cross-validation were used to optimize the performance of each model. [2]

2.3 Diabetes Prediction Using Machine Learning.

The goal of this project was to create a system that could combine the results of different machine learning models to make early predictions with higher accuracy. The k-Nearest Neighbors, logistic regression, Decision Tree, Random Forest, SVM algorithm is used in this study. In this experiment, three different values of regularization parameter were used in the logistic relation model. The parameters are c=1, c=0.01, c=100 and the hyperplane was determined in the SVM model. Four kernels are used to deferant the hyperplane, the kernels are LINEAR, POLY, RVF and SIGMOID. [3]

2.4 Diabetes Prediction using Machine Learning Techniques.

This paper focuses on the early prediction of diabetes using various machine learning techniques to achieve higher accuracy. It highlights that untreated diabetes can lead to severe health problems like heart and kidney issues, blood pressure, and eye damage. Early prediction can help control the disease and save lives. The aim is to analyze the performance and accuracy of these methods and identify the most important features for prediction. The results indicate that Random Forest achieved higher accuracy compared to the other machine learning techniques used. The project achieved an overall classification accuracy of 77%. [4]

2.5 Diabetes Disease Prediction Using Machine Learning Algorithms.

Arwatki Chen Lyngdoh et al. conducted research on predicting diabetes disease using 5 supervised ML Algo : KNN, Naive Bayes, Decision Tree Classifier, Random Forest, and SVM. by including current risk variables and performing cross-validation, they achieved consistent accuracy with the KNN classifier achieving a hig accuracy of 76%. The main objective of the study was to identify the best outcomes for accurately predicting diabetes disease, considering accuracy and computing time. [5]

2.6 Diabetes prediction using supervised machine learning.

This paper investigates the use of supervised machine learning algorithms, specifically K-Nearest Neighbor (KNN) and Naive Bayes, to predict diabetes based on various health attributes. The study aims to provide a quick and efficient method for diabetes detection to prevent severe conditions, as delays in diagnosis often lead to advanced stages of the disease.K-fold cross-validation (specifically 10-fold cross-validation) was used to validate the results.The study concluded that the Naive Bayes algorithm generally outperformed KNN in predicting diabetes.The average accuracy for Naive Bayes was 76.07%, with a precision of 73.37% and recall of 71.37%. [6]

2.7 Prediction of Diabetes using Classification Algorithms.

This paper focuses on designing a model to predict the likelihood of diabetes in patients with maximum accuracy using machine learning classification algorithms. The study utilizes three algorithms: Decision Tree, Support Vector Machine (SVM), and Naive Bayes. The results indicate that the Naive Bayes algorithm outperforms the others, achieving the highest accuracy of 76.30%. The findings were further validated using Receiver Operating Characteristic (ROC) curves.The authors suggest that this designed system could be extended in the future to diagnose other diseases and improve diabetes analysis with additional machine learning algorithms. [7]

2.8 Early Prediction of Diabetes Mellitus Using Machine Learning.

This paper focuses on the early prediction of Diabetes Mellitus using machine learning classification algorithms. Four machine learning algorithms were employed: Linear Discriminant Analysis (LDA), K-nearest neighbor (KNN), Support Vector Machine (SVM), and Random Forest (RF). The performance of these algorithms was evaluated using statistical measures such as sensitivity (recall), precision, specificity, F-score, and accuracy. K-fold cross-validation with K=10 was used to check the effectiveness and measure the performance of the model. The experimental results show that the Random Forest (RF) algorithm achieved the highest accuracy of 87.66%, outperforming the other classification algorithms for early-stage diabetes prediction. [8]

2.9 Techniques of Machine Learning for the Purpose of Predicting Diabetes Risk in PIMA Indians

This paper explores the use of machine learning techniques to predict diabetes risk, specifically focusing on the PIMA Indian population, which has a high prevalence of the disease. Important features for predicting diabetes risk were chosen using correlation matrices and recursive feature elimination (RFE). The study also utilized other models such as KNN, AdaBoost, Naive Bayes,

and XGBoost. Among the models tested, the random forest model demonstrated the best performance across several metrics, including F1 score, AUC score, accuracy, precision, and recall. Specifically, the random forest model achieved 78.12% accuracy, 75.68% precision, 55.13% recall, 63.87% F1 score, and 0.83% AUC. [9]

2.10 Diabetes Prediction Using Machine Learning Algorithms and Ontology.

It presents a comparative study and review of popular machine learning (ML) techniques and ontology-based ML classification for predicting diabetes. The study considers several classification algorithms: Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Artificial Neural Network (ANN), Naive Bayes, Logistic Regression, and Decision Tree. In 10-fold cross-validation mode, Ontology, SVM, and Logistic Regression achieved the highest accuracy values of 77.5%, 77.3%, and 77.2% respectively. In split test mode (66% split), Logistic Regression, Ontology, and SVM achieved 80.1%, 79.7%, and 79.3% accuracy respectively. [10]

2.11 Analisis Perbandingan Ensemble Machine Learning Dengan Teknik Somote Untuk Prediksi Diabetes.

This paper explores the prediction of diabetes disease using an analysis of five supervised machine learning algorithms: K-Nearest Neighbors (KNN), Naïve Bayes, Decision Tree Classifier, Random Forest, and Support Vector Machine. The experiment split the dataset into an 80:20 ratio for training and testing, respectively. After 10-fold cross-validation, the KNN classifier achieved the highest accuracy of 76%. Other classifiers also achieved accuracies above 70%. [11]

2.12 Diabetes Prediction Using Machine Learning Techniques.

This research paper explores using machine learning to predict diabetes. The study uses the PIMA Indian Diabetes Dataset and compares four machine learning models: Logistic Regression, Decision Tree, Random Forest, and Support Vector Machine (SVM). The paper concludes that the. Random Forest model had the most robust performance across all evaluation metrics and is a good candidate for use in predictive healthcare. Random Forest achieved the highest ROC-AUC score of 0.83, indicating strong performance in distinguishing between diabetic and non-diabetic patients. It had an accuracy of 75% and showed a good balance in its predictions for both classes. Future work could involve hyperparameter optimization, feature engineering, and cross-dataset validation to improve the models' performance and generalizability. [15]

2.13 Early Diabetes Detection using Novel Hybird Machine Learning Algorithm.

In their paper propose a new hybrid machine learning technique for the early detection of diabetes. The study highlights that early diagnosis and healthcare monitoring are significant global health issues. The authors note that machine learning can be a cost-effective and fast alternative to traditional methods for detecting diabetes. The proposed hybrid model combines the Decision Tree (DT) and Gradient Boosting Classifiers (GBC) algorithms. The Pima Indian Diabetes Dataset (PIDD) was used to train and test the model using the K-fold cross-validation approach.

The proposed hybrid algorithm achieved a forecast accuracy of 90.5% and a K-fold cross-validation accuracy of 92%.In comparison, the Decision Tree and Gradient Boost Classifiers alone achieved accuracy rates of 78% and 80%, respectively. [16]

2.14 SHAP-Based Explainable Framework for Disease Prediction and Comorbidity Risk Assessment in Healthcare.

The research paper proposes an explainable machine learning framework based on SHAP (SHapley Additive exPlanations) for predicting diabetes and assessing comorbidity risk. Initial performance is benchmarked using four baseline models: Logistic Regression, Support Vector Machines, Random Forest, and XGBoost SHAP analysis identified glucose, BMI, and age as the most critical features for diabetes prediction. Before feature selection, XGBoost achieved the highest performance among the baseline models with an accuracy of 0.84 and an AUC of 0.87. The XGBoost model combined with GA (GA-XGBoost) achieved the best performance with an accuracy of 0.86 and an AUC of 0.89. [17]

2.15 Comparative Analysis of Random Forest and Support Vector Machine for Classifying Pima Indians Diabetes Dataset.

This study compares the effectiveness of the Random Forest (RF) and Support Vector Machine (SVM) machine learning algorithms for classifying the Pima Indians Diabetes Dataset. The study uses 10-fold cross-validation to evaluate both models on key metrics: accuracy, precision, recall, and F1-score. The results highlighted Random Forest as the more stable and reliable model, achieving an average accuracy of 76.3% and consistently strong results across all folds. In contrast, while the SVM with a polynomial kernel delivered slightly better precision (74.57%), it fell short in terms of overall accuracy, recall, and F1-score when compared to Random Forest. [18]

2.16 Literature Review Summary

The table 1 below presents a brief overview of the important research papers on the use of machine learning in diabetes prediction. This table provides a comparative analysis of the dataset used in each study, the results of which machine learning models and/or algorithms were applied, and which models or algorithms performed better.

Name	Year	Article Title	Dataset	Model Used	Performance
Mitushi Soni. et al.[4]	2020	Diabetes Prediction using Machine Learning Techniques	Pima Indians Diabetes Dataset	1.K-Nearest Neighbor 2.Logistic Regression 3.Decision Tree 4.Support Vector Machine 5.Gradient Boosting 6. Random Forest	1. 69% 2. 75% 3. 74% 4. 73% 5. 77% 6. 75%

Name	Year	Article Title	Dataset	Model Used	Performance
KM JYOTI RANI[3]	2020	Diabetes Prediction Using Machine Learning	Not mentioned	1. K-Nearest Neighbors (KNN) 2.Logistic Regression 3.Decision Tree 4.Random Forest 5.SVM	1. 78% 2. 78% 3. 99% 4. 97% 5. 77%
Zhengyi Zhang[2]	2025	Comparison of Machine Learning Models for Predicting Type 2 Diabetes Risk Using the Pima Indians Diabetes Dataset	Pima Indians Diabetes Dataset	1.Logistic Regression 2.Random Forest 3.Support Vector Machine (SVM) 4. XGBoost	1. 69% 2. 75% 3. 74% 4. 73% 5. 77% 6. 75%
Arwatki Chen Lyngdoh. et al[5]	2020	Diabetes Disease Prediction Using Machine Learning Algorithms	Pima Indians Diabetes Dataset	1.K-Nearest Neighbor 2. Naive Bayes 3.Decision Tree 4.Random Forest 5.Support Vector Machine	1. 76% 2. 75% 3. 74% 4. 71% 5. 71%
Muhammad Exell Febrian. et al[6]	2022	Diabetes prediction using supervised machine learning	Pima Indians Diabetes Dataset	1.K-Nearest Neighbor 2. Naive Bayes	1. 77% 2. 78%
Gaurav Tripathi. et al[8]	2020	Early Prediction of Diabetes Mellitus Using Machine Learning	Pima Indians Diabetes Dataset	1.Linear Discriminant Analysis 2.K- Nearest Neighbour 3.Support Vector Machine 4.Random Forest	1. 76% 2. 79% 3. 80% 4. 87%
Bhukya Madhu. et al[9]	2023	Techniques of Machine Learning for the Purpose of Predicting Diabetes Risk in PIMA Indians	Pima Indians Diabetes Dataset	1. K- Nearest Neighbour 2. Random Forest Classifier	1. 86% 2. 88%
Deepti Sisodia. et al[7]	2022	Prediction of Diabetes using Classification Algorithms	Pima Indians Diabetes Dataset	1. Naive Bayes 2. SVM 3.Decision Tree	1. 76% 2. 65% 3. 73%

Name	Year	Article Title	Dataset	Model Used	Performance
Bhukya Madhu. et al[9]	2023	Techniques of Machine Learning for the Purpose of Predicting Diabetes Risk in PIMA Indians	Pima Indians Diabetes Dataset	1. K- Nearest Neighbour 2. Random Forest Classifier 3.Logistic Regression 4.Decision Tree Classifier 5.Ada Boost Classifier 6.Navies Bayes Classifier 7.XG Boost Model	1.86% 2.88% 3. 82% 4. 85% 5. 90% 6. 80% 7. 91%
Hakim El Massari. et al[10]	2022	Diabetes Prediction Using Machine Learning Algorithms and Ontology	Pima Indians Diabetes Dataset	1.Support Vector Machine 2.K- Nearest Neighbour 3. ANN 4.Logistic Regression 5.Navies Bayes 6.Decision Tree 7. Ontology	1. 81% 2. 80% 3. 83% 4. 82% 5. 82% 6. 80% 7. 81%
Nur Tri Ramadhanti Adiningrum. et al[11]	2025	Analisis Perbandingan Ensemble Machine Learning Dengan Teknik Somote Untuk Prediksi Diabetes.	Pima Indians Diabetes Dataset	1.Voting Classifier 2.Decision Tree 3.Random Forest 4.Navies Bayes 5.Logistic Regression 6. KNN 7. AdaBoost 8. SVM	1. 81% 2. 80% 3. 81% 4. 76% 5. 79% 6. 79% 7. 79% 8. 77%
Hejia Zhou. et al[12]	2025	A robust and generalized framework in diabetes classification across heterogeneous environments	Pima Indians Diabetes Dataset	1.Random forest 2.Gradient boosting 3.Multilayer perception 4. XGBoost 5. Ensemble. 6.Deep Learning	1. 77% 2. 77% 3. 65% 4. 79% 5. 77% 6 .75%
TN Srinivas Rao. et al[13]	2024	An Optimized Hybrid Ensemble Machine Learning Model for Accurate Diabetes Prediction and Early Diagnosis	Pima Indians Diabetes Dataset	1.Decision Tree (DT). 2.Support Vector Machine 3.Gradient Boosting (GBM) 4.Proposed Ensemble.	1. 83.72 2. 86.20 3. 89.00 4. 92.50
Mohammad Raquibul	2025	Machine Learning Based Prediction and Insights of Diabetes Disease :	Pima Indians Diabetes Dataset	1.Logistic regression 2.k-Nearest Neighbour	1.81% 2. 84%

Name	Year	Article Title	Dataset	Model Used	Performance
Hossain. et al[14]		Pima Indian and Frankfurt Dataset.		3.Navies Bayes 4.Decision Tree 5.Random Forest 6.XGBoost	3. 85% 4. 89% 5. 88% 6. 92%
Ms. Minal Mangesh Kusam. et al[15]	2025	Diabetes Prediction Using Machine Learning Techniques	Pima Indians Diabetes Dataset	1.Logistic regression. 2.DecisionTree. 3.Random Forest	1. 75% 2. 72% 3. 75%

Table 2.1: Summary of existing works on diabetes prediction using machine learning.

Chapter 3

Methodology

3.1 Introduction

This chapter presents the methodology adopted to predict diabetes using machine learning models. The overall research process involves data collection, preprocessing, feature selection, model development, and performance evaluation. The dataset used in this study is the Pima Indians Diabetes dataset, which contains diagnostic measurements of female patients aged 21 years or older. The following sections describe the dataset, data preprocessing steps, applied models, and evaluation techniques in detail.

3.2 System Architecture

The main objective of this study is to develop and evaluate a machine learning (ML) model to accurately predict an individual's risk of developing diabetes based on data. To achieve this objective, we followed a systematic approach, which includes dataset collection, data preprocessing, feature selestion, Data partitiong, Model building, and performance evaluation. A clear and concise picture of this entire process is presented in Figure3.1, which highlights the main steps of our project.

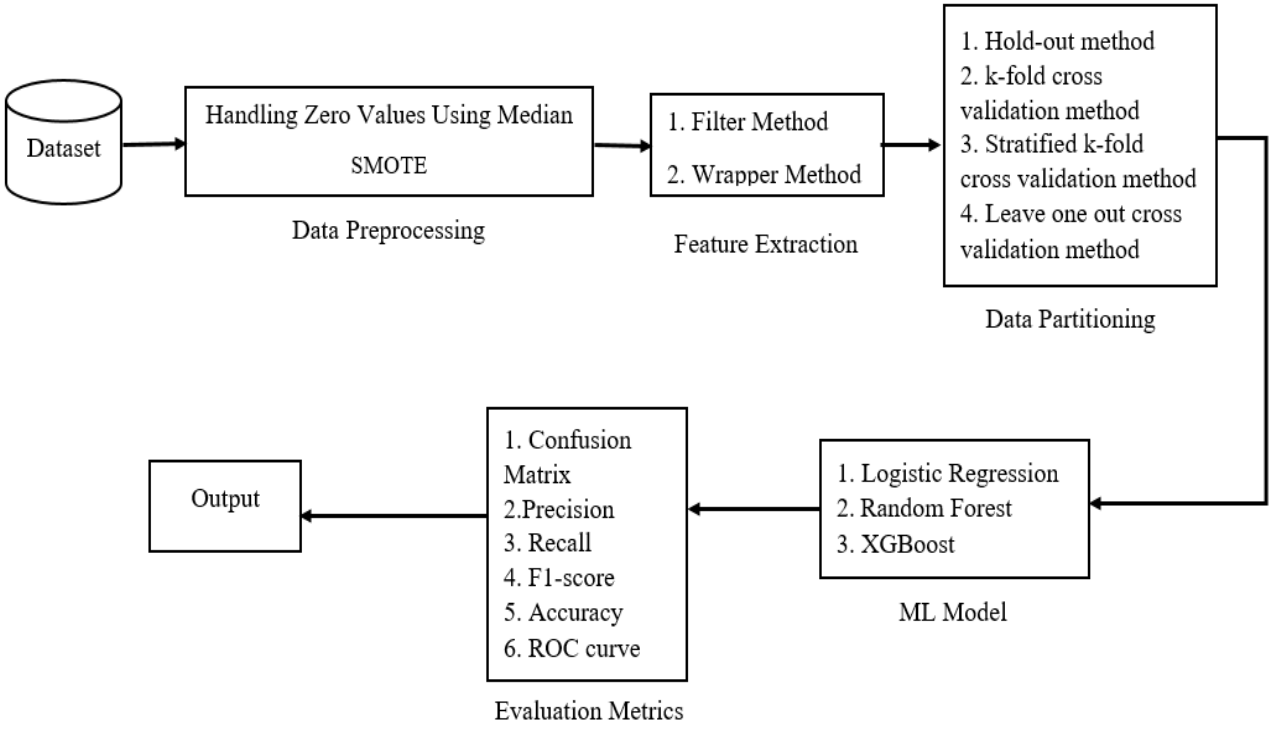


Figure 3.1: Operational Process of Diabetes Prediction

3.3 Data Collection

In this work, Pima Indian Diabetes Dataset has been used. This data set is collected from UCI repository. This particular dataset has been widely used in machine learning experiments. The Pima Indians diabetes database dataset consists of 768 data which is divided into 9 attributes and 2 classes with a total of class '1'for diabetic (268) and class '0' for non-diabetic patients (500).

3.4 Dataset Description

No	Feature Name	Feature Description	Feature Data Type
1	Pregnancies	Number of times pregnant.	Numerical
2	Glucose	Plasma glucose concentration a 2 hours in an oral glucose tolerance test.	Numerical
3	Blood Pressure	Diastolic blood pressure (mm Hg).	Numerical
4	Skin Thickness	Triceps skin fold thickness (mm).	Numerical
5	Insulin	2-Hour serum insulin (mu U/ml).	Numerical
6	BMI	Body mass index (weight in kg/(height in m)^2).	Numerical
7	Diabetes Pedigree Function	Diabetes pedigree function.	Numerical
8	Age	Patient's Age.	Numerical
9	Outcome	Target variable (1 if diabetic, else 0).	Numerical

Table 3.1: Dataset Feature Description.

3.5 Descriptive Statistics of the Dataset

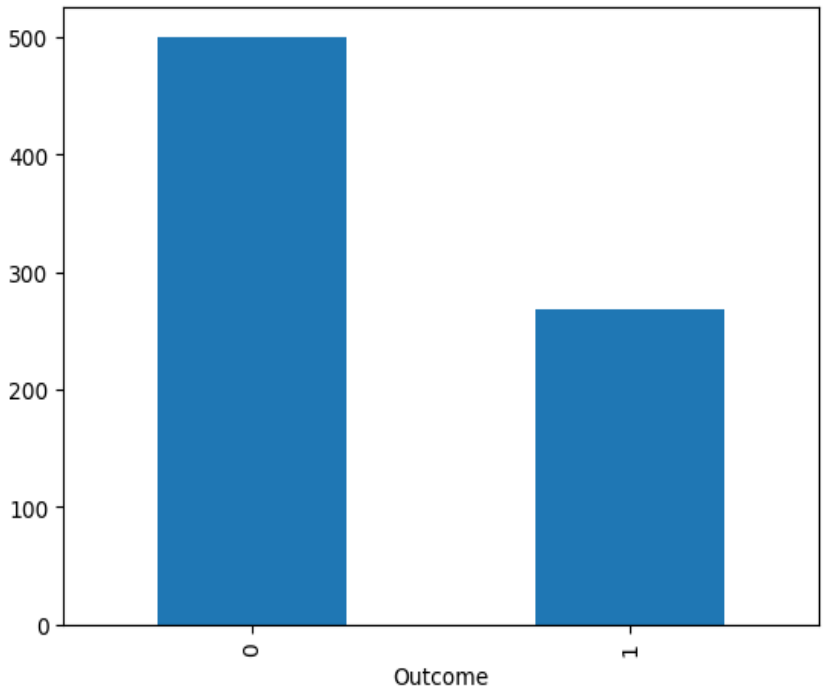
Among the participants, approximately 65% were non-diabetic, while 35% were diabetic, reflecting a moderate class imbalance.The demographic feature "Age" ranges from 21 to 81 years, with a mean of 33.2 years. The majority of individuals fall within the 21–41 age group, suggesting that the dataset is predominantly composed of younger to middle-aged patients. The number of pregnancies varies from 0 to 17, with an average of 3.85, indicating that many women in the dataset had relatively few pregnancies.In terms of clinical variables, the average glucose concentration is 120.9 mg/dl, with values ranging from 0 to 199. Notably, the presence of zero values in glucose and other variables (such as blood pressure, skin thickness, insulin, and BMI) represents missing or unrecorded measurements, which require preprocessing. Blood pressure values range between 0 and 122 mmHg, with a mean of 69.1 and a median of 72. Similarly, skin thickness measurements range from 0 to 99 mm, with a mean of 20.5 mm.The insulin variable is highly skewed, with values ranging from 0 to 846 and a mean of 79.8, reflecting a wide variation among patients. BMI (Body Mass Index) ranges from 0 to 67.1, with an average of 32, indicating that a large proportion of patients are overweight or obese which is consistent with common diabetes risk factors.

	Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	DPF	Age	Outcome
Count	768.00	768.00	768.00	768.00	768.00	768.00	768.00	768.00	768.00
mean	3.84	120.89	69.10	20.53	79.79	31.99	0.47	33.24	0.34
std	3.36	31.97	19.35	15.95	115.24	7.88	0.33	11.76	0.47
min	0.00	0.00	0.00	0.00	0.00	0.00	0.07	21.00	0.00
25%	1.00	99.00	62.00	0.00	0.00	27.30	0.24	24.00	0.00
50%	3.00	117.00	72.00	23.00	30.50	32.00	0.37	29.00	0.00
75%	6.00	140.25	80.00	32.00	127.25	36.600	0.62	41.00	1.00
Max	17.00	199.00	122.00	99.00	846.00	67.10	2.42	81.00	1.00

Table 3.2: Summary Stats for Diabetes Dataset

3.6 Data Visualization

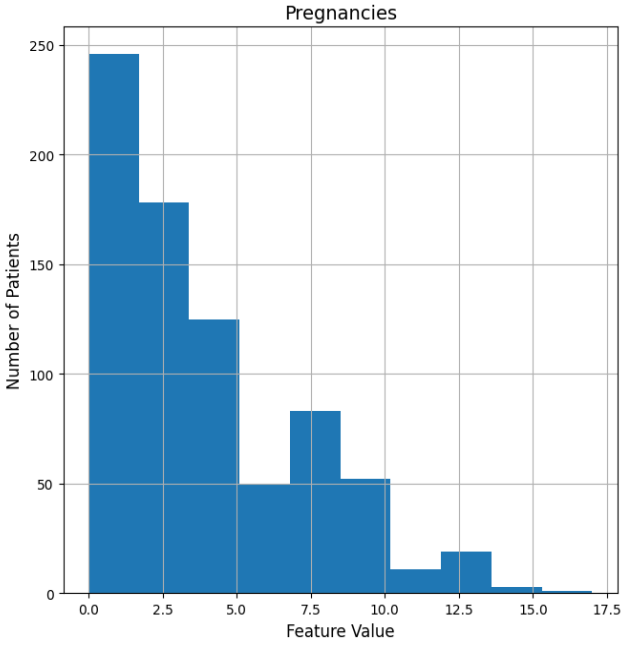
Data visualization is important to have a better idea about the data. It is visual context of data information. It helps to understand the pattern of data. For achieving goal it is necessary to interpret big and complex dataset, data visualization will a great role in this area. As the purpose of data analysis to gain insights, data visualization is important because also has a great impact on human body. Outcome can be plotted to get a better view. As outcome column has two type values 0 and 1. In PIDD, There is 500 healthy (when outcome=0) and 268 (when the outcome=1) diabetic cases.

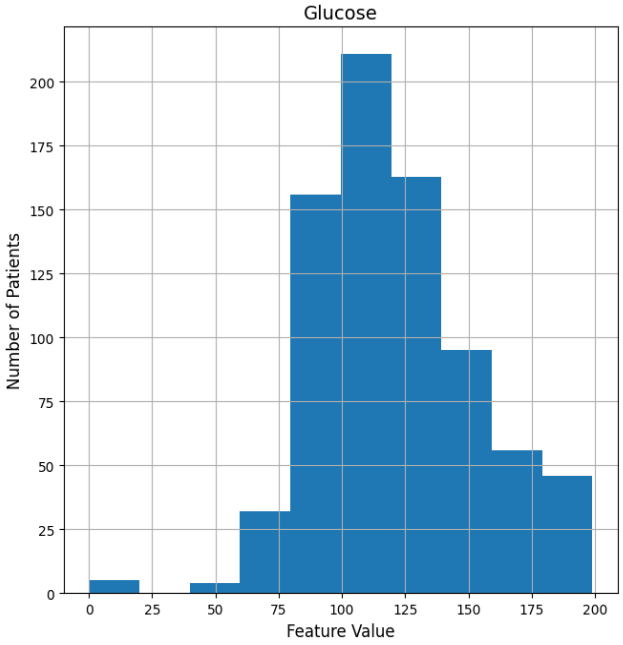


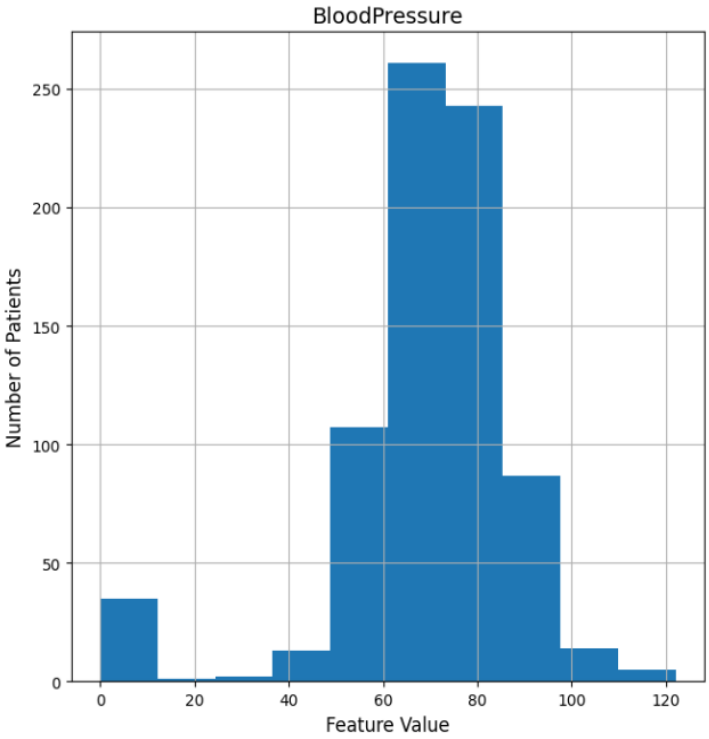
.Figure 3.2: Outcome Cases

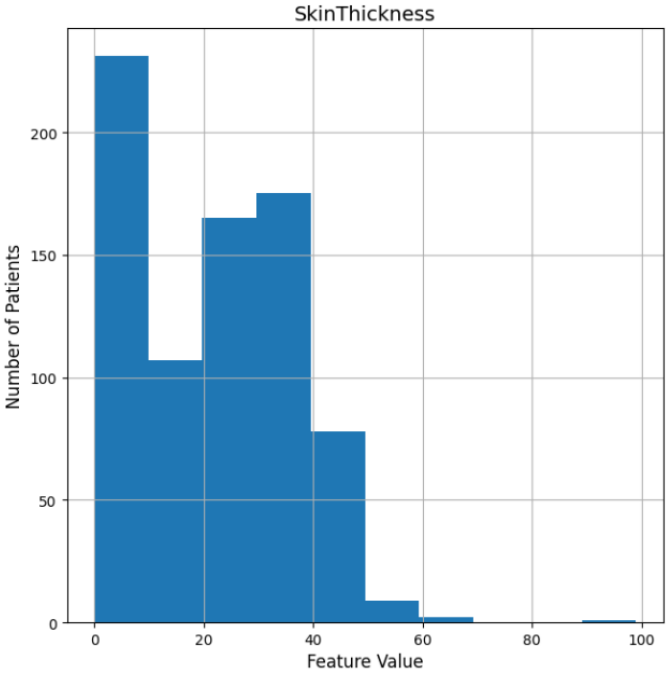
3.6.1 Histogram

A Histogram is a graph that shows the distribution of data across different ranges (bins). It uses adjacent bars to represent the frequency of values within each range









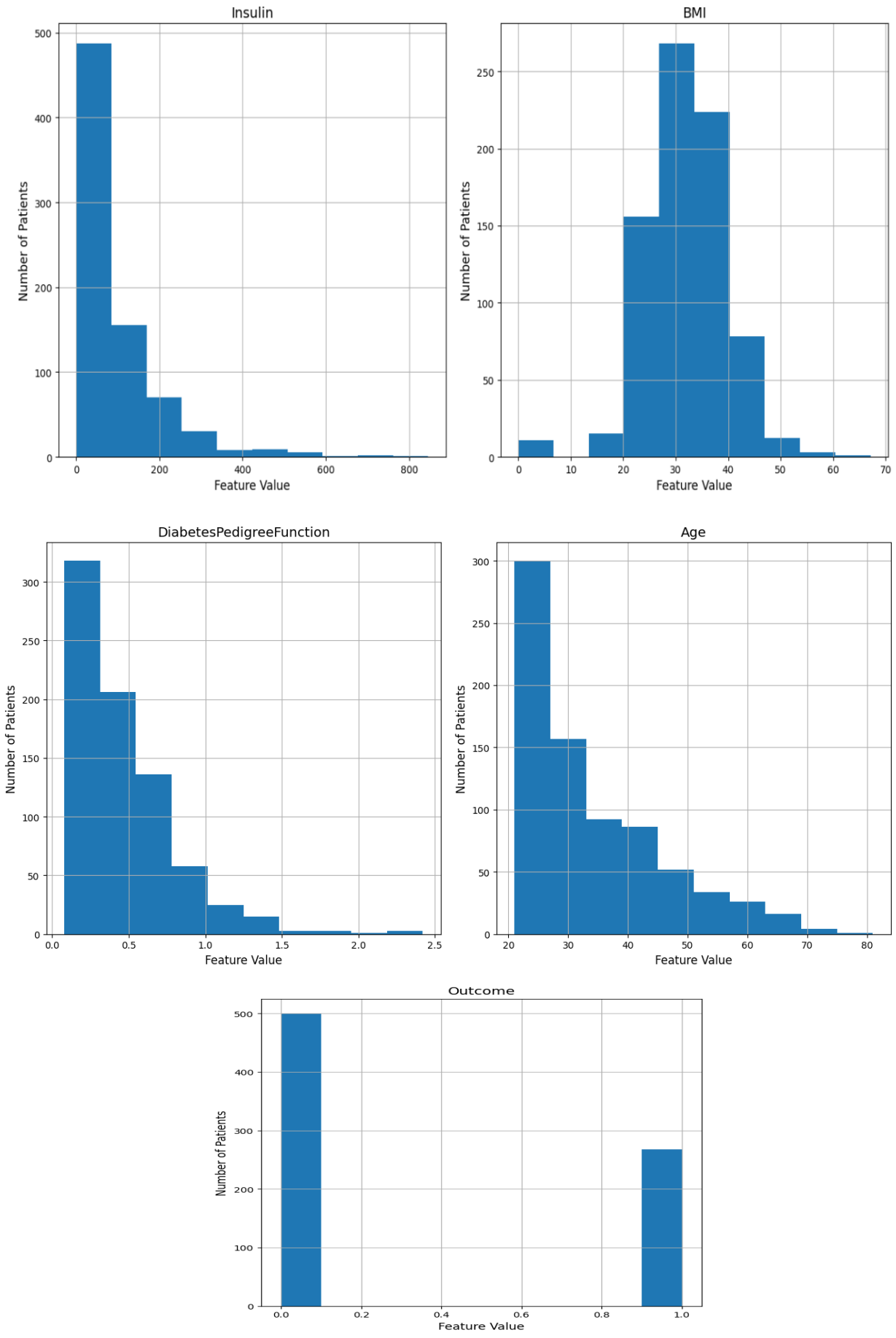


Figure 3.3: Histogram Relation Between Single Attribute

- **Pregnancy history:**

  Important in determining the patient's diabetes risk. Those who have a history of pregnancy and diabetes before are more likely to develop diabetes in the future. As can be seen from the histogram, most patients have a pregnancy count between 0 and 5, and a few records have values of 10 or more. The distribution is Right-Skewed, because most of the values are downward sloping. Although the feature has some outliers, it is important in determining the patient's health risk

- **Glucose:**

  Glucose level in the blood, when higher than normal, is known as hyperglycemia, which indicates diabetes. In our dataset, Glucose values generally range between 70 and 150, but a few records have a value of 0, which is not realistic and can be considered as missing values. The overall distribution appears roughly normal. Therefore, although Glucose is a highly important feature for predicting diabetes, these missing or incorrect values need to be addressed during preprocessing.

- **Blood Pressure:**

  In our dataset, most of the patients' blood pressure is between 60 and 80. However, some values of 0 are found in the histogram, which is not possible in reality and these should be treated as missing values. Although the overall distribution is fairly normal, these zero values are creating some problems. Therefore, they need to be fixed before building a predictive model.

- **Skin Thickness:**

  Skin is the largest organ in our body, which contains collagen and is related to insulin production. The value of the SkinThickness feature in the dataset is usually found between 0 and 30. However, in many cases, 0 values are found, which are not real values but should be considered as missing values. As a result, the distribution is completely skewed. Therefore, missing value imputation must be done before using this feature, otherwise the accuracy of the model may be affected.

- **Insulin**:

  Looking at the histogram of our dataset, it can be seen that most of the values are between 0 and 100. However, many records have values of 0, which are not real values but should be considered missing values. Again, some values are much higher, such as 200, 400 and even 800, which are clear outliers. As a result, the distribution has become very right-skewed. So the Insulin feature has both missing values on one hand and extreme outliers on the other hand, which need to be cleaned up before building the model.

- **BMI** :

  Generally, the higher the BMI, the higher the risk of diabetes. In our dataset, most of the BMI values are between 20 and 40. The histogram is roughly normal distribution, but some records have values of 0, which is not possible in reality and should be treated as missing values. So, although BMI is a useful feature, these missing values must be fixed before building a predictive model.

- **Diabetes Pedigree Function:**

  In our dataset, the values are usually between 0 and 1. However, in some cases, values up to

2 are found, which can be considered outliers. Looking at the histogram, it can be seen that the distribution is right-skewed. So even if most of the values are in the normal range, these outliers can affect the model.

- **Age:**

  According to the histogram of our dataset, most patients are between 20 and 50 years old, although there are some records with ages of 70 or older, which can be considered outliers. Verdict: Age is an important and reliable feature, even though the distribution is Right-Skewed.

3.6.2 Correlation Matrix

The correlation matrix shows Figure 3.4 the degree of relationship among the different features of the dataset. From the analysis, it is observed that Glucose has a moderate positive correlation (0.49) with Outcome, which indicates that higher glucose levels are strongly associated with the presence of diabetes. Similarly, BMI (0.31), Age (0.24), and Pregnancies (0.22) also show a positive correlation with Outcome, suggesting that individuals with higher body mass index, older age, and greater number of pregnancies are more likely to develop diabetes. Among the feature-to-feature relationships, a strong positive correlation is found between Pregnancies and Age (0.54), which is expected since the number of pregnancies generally increases with age. SkinThickness and BMI (0.54) also show a strong relationship, reflecting that people with higher BMI tend to have greater skin thickness. In addition, Glucose and Insulin (0.42) demonstrate a notable positive correlation, indicating their biological link in regulating blood sugar. On the other hand, PedigreeFunction and BloodPressure Overall, the correlation matrix highlights that Glucose, BMI, Age, and Pregnancies are the most influential variables in relation to the diabetes outcome.

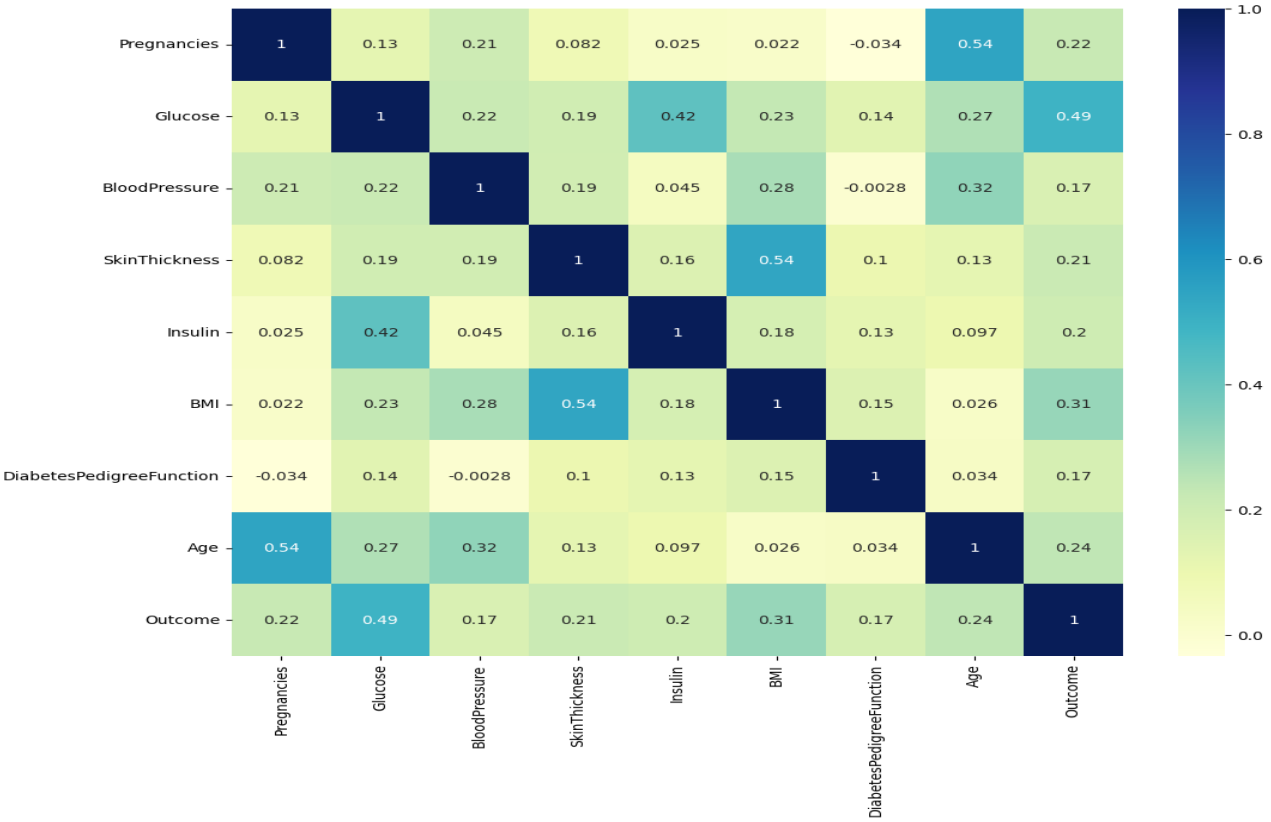
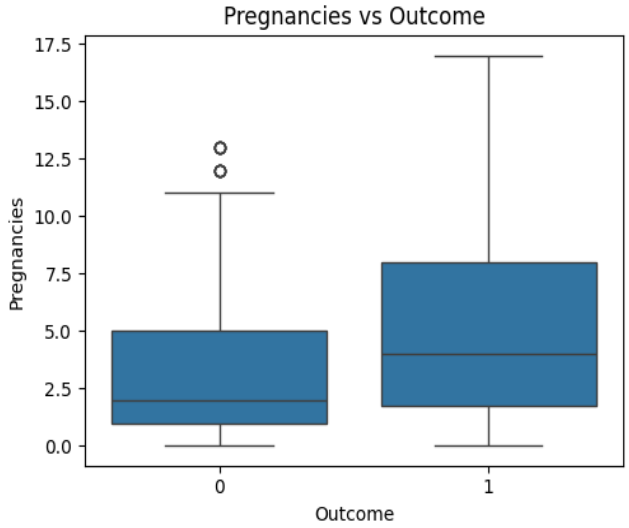
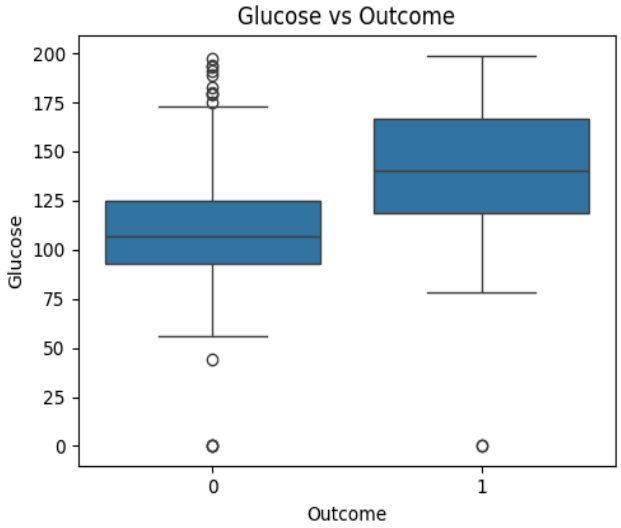


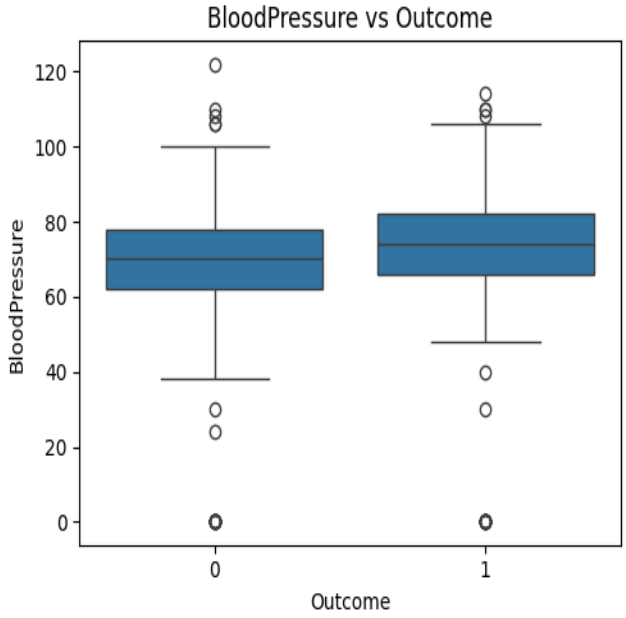
Figure 3.4: Correlation Matrix

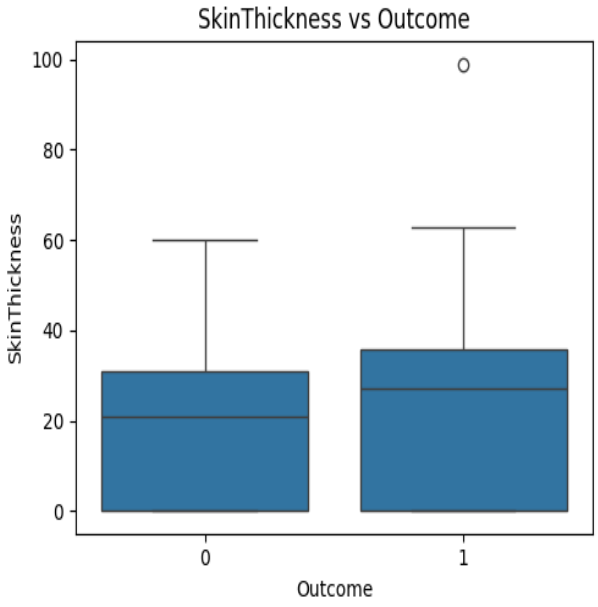
3.6.3 Boxplots

To detect potential outliers and understand the distribution of the dataset features, boxplots were generated. The boxplots provide information about the median, interquartile range (IQR), and extreme values of each attribute. This visualization helps to identify unusual data points, which may indicate either abnormal patient conditions or possible measurement errors. Moreover, by analyzing the spread of the data, boxplots allow us to observe which features have a wider variability and which are more consistent. Detecting these variations is important because highly skewed or outlier-dominated features may affect the performance of machine learning models. Therefore, boxplots not only help in data cleaning but also provide insights for feature selection and preprocessing steps.









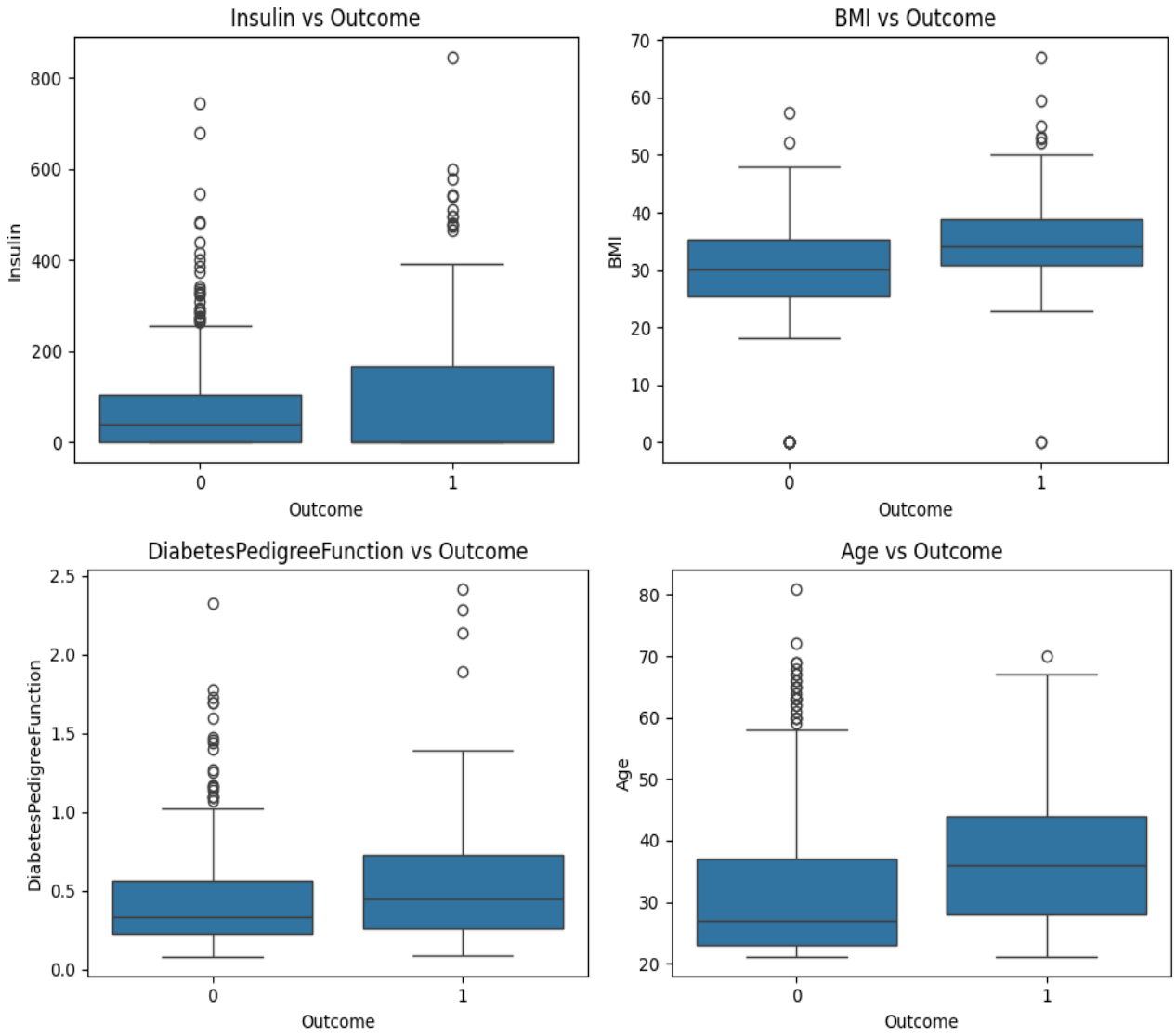


Figure 3.5 Boxplots of Different Features Against Outcome

- **Pregnancies vs Outcome:**

  The boxplot show a strange relationship between the number of pregnancies and diabetes. According to the graphs, the increased number of pregnancies highlights an increased risk of diabetes.

- **Glucose vs Outcome:**

  Glucose level plays a significant role in determining whether the patient has diabetes. Patients with a median glucose level of less than 120 are more likely to be nondiabetic. Patients with a median glucose level greater than 140 are more likely to be diabetic. Therefore, high glucose levels are a good indicator of diabetes.

- **Blood Pressure vs Outcome:**

  The boxplot provide a clear understanding of the relationship between blood pressure and diabetes. The boxplot shows that the median blood pressure for diabetic patients is slightly higher than non diabetic patients. The violin plot shows that the distribution of blood

pressure for diabetic patients is slightly higher than for nondiabetic patients. However, there has not been enough evidence to conclude that blood pressure is a good predictor of diabetes.

- **Skin Thickness vs Outcome:**

  Here, both the boxplot and violinplot reveal the effect of diabetes on skin thickness. As observed in the boxplot, the median skin thickness is higher for diabetic patients than nondiabetic patients. Nondiabetic patients have a median skin thickness of nearly 20, compared to almost 30 in diabetic patients. The violin plot shows the distribution of patients' skin thickness among the patients, where the nondiabetic ones have a greater distribution near 20, people with diabetes have a smaller distribution near 20, and increased distribution near 30. Therefore, skin thickness can be an indicator of diabetes.

- **Insulin vs Outcome:**

  Here, the boxplot show the distribution of insulin levels in patients. In nondiabetic patients, the insulin level is near 100, whereas in diabetic patients, the insulin level is near 200. In the violin plot, we can see that the distribution of insulin levels in nondiabetic patients is more spread out near 100, whereas, in diabetic patients, the distribution is contracted and shows a little spread in higher insulin levels. This indicates that the insulin level is a good indicator of diabetes**.**

- **BMI vs Outcome:**

  Nondiabetic patients have a normal BMI within the range of 25–35, whereas diabetic patients have a BMI greater than 35. Plot reveals the BMI distribution, where the nondiabetic patients have an increased spread from 25 to 35, with narrows after 35. However, in diabetic patients, there is an increased spread at 35 and an increased spread at 45–50 compared to nondiabetic patients. Therefore, BMI is a good predictor of diabetes, and obese people are more likely to be diabetic.

- **Diabetes Pedigree Function vs Outcome:**

  The Diabetes Pedigree Function (DPF) calculates diabetes likelihood depending on the subject's age and diabetic family history. The boxplot shows that patients with lower DPF are much less likely to have diabetes. The patients with higher DPF are much more likely to have diabetes. In the violin plot, the majority of the nondiabetic patients have a DPF of 0.25–0.35, whereas the diabetic patients have an increased DPF, which is shown by their distribution in the violin plot where there is an increased spread in the DPF from 0.5 -1.5. Therefore, the DPF is a good indicator of diabetes.

- **Age vs Outcome**:

  Nondiabetic patients are generally younger, with the age distribution concentrated between 20–35 years. Their spread narrows after 35, with fewer outliers in higher ages. On the other hand, diabetic patients tend to be older, with a wider distribution ranging from 30–50 years, and several outliers extending beyond 60 years. The median age of diabetic patients is clearly higher than that of nondiabetic patients. Therefore, age is an important predictor of diabetes, and older individuals are more likely to develop diabetes compared to younger individuals.

3.7 Data Preprocessing

Data processing is an important step in making raw data suitable and accurate for analysis. Usually, a dataset cannot be used directly because it may contain various types of problems, such as missing values, outliers, duplicate observations or inconsistent data. If these problems are not resolved, the analysis results may be incorrect or the model may not learn correctly. Therefore, the data processing step involves data cleaning, transformation and standardization.
In PIDD, there are no missing or zero values directly but there are many zero values which are unreasonable, the features mentioned glucose, blood pressure, skinfold thickness, insulin, human BMI have zero values which cannot be zero. So we are treating those values as null values. To replace null values we are using Median Method. So that our results are good.

Features Name	Missing values
Pregnancies	0
Glucose	0
BloodPressure	0
SkinThickness	0
Insulin	0
BMI	0
DiabetesPedigreeFunction	0
Age	0
Outcome	0

Table3.3: Missing Value Count

Features Name	Zero value
Pregnancies	111
Glucose	5
BloodPressure	35
SkinThickness	227
Insulin	374
BMI	11
DiabetesPedigreeFunction	0
Age	0
Outcome	500

Table3.4: Zero Value Count

3.7.1 Median Method

Median imputation is a statistical technique for handling missing, zero value by replacing the missing, zero values in a dataset with the median of the observed (non-missing) values in that column. This method is particularly useful for skewed data distributions and datasets with outliers, as the median is less affected by extreme values than the mean (average).

We used the Median Imputation Method to resolve the zero values present in the dataset. In this method, where a value was zero for each attribute, the median value of the corresponding attribute was inserted. The median is a robust central tendency measure, which is not easily affected by outliers or extreme values. Our dataset had numerical attributes, where median imputation was the most appropriate method.
Working Process of Median [19]:

1. All non-missing values of the attribute are arranged in ascending order.
2. The number of values is determined as n.
   - If n is odd, Median $= \frac{n+1}{2}$
   - If n is even, Median $= \frac{1}{2}\{\frac{n}{2} + (\frac{n}{2}+1)\}$
3. The median value is placed where the value is missing.

3.7.2 SMOTE

SMOTE is a popular over-sampling method that addresses the problem of class imbalance in datasets. When the number of examples in one class is too small compared to the other class, machine learning models tend to be biased towards the majority class and fail to learn the minority class properly. This can lead to poor predictions for the minority class, even if overall accuracy seems high.SMOTE solves this problem by generating new synthetic examples of the minority class rather than simply duplicating existing ones. It selects a minority class sample and identifies its nearest neighbors in the feature space. Then, it creates new synthetic points along the line segments joining the sample and its neighbors.

3.8 Feature Extraction

Feature extraction is an important step in which the necessary and influential features are selected from the raw data which increases the performance of the model. In this process, we basically extract the information from each attribute or column of the dataset that has the most impact on the diagnosis. In this study, we are using two feature selection methods, which are described below.

3.8.1 Filter Method

The Filter Method is a simple and effective approach to feature selection for diabetes prediction. This method uses statistical measures to determine the importance of each feature, which is independent of any machine learning model. Basically, the relationship between each feature and the target variable (Diabetes Outcome) is analyzed to determine which features are important. [20]
The Filter Method is usually completed in three steps:

1. The statistical relationship between each feature and the Diabetes Outcome is measured.
2. The features are ranked based on the relationship metrics.
3. In the last step, the selected features are used for model training, excluding less important or irrelevant features.

The main advantage of the Filter Method is that it is fast and easy to interpret. In addition, since it is model-independent, it can be used before any machine learning algorithm. The limitation is that it only considers the importance of individual features, so information about feature interactions is not visible.After applying the filter method, the relationship between each feature and the target variable was calculated. Based on the ranking results, the five most relevant features were identified. The features were: Pregnancies, Glucose, BMI, DiabetesPedigreeFunction and Age.

3.8.2 Wrapper Method

Wrapper methods are a powerful method for feature selection. They test how a model performs on different feature subsets. Instead of just looking at statistical relationships like filter methods, wrapper techniques actually try different feature subsets to see which one works best for the

model. This results in much more accurate feature selection, because the entire process is directly related to the performance of the classifier [21].
In the context of diabetes prediction using machine learning, wrapper methods have been applied to the Pima Indians diabetes dataset to identify the most relevant features that improve prediction accuracy.In the wrapper approach, features are selected by directly evaluating the performance of the model. In this case, a subset of different features is selected and the model is trained and tested on them. The performance of the model is measured for each subset and the features that contribute the most to the model's accuracy are identified as important features. The analysis results show that the five most relevant features are: Glucose, BloodPressure, BMI, DiabetesPedigreeFunction, Age .

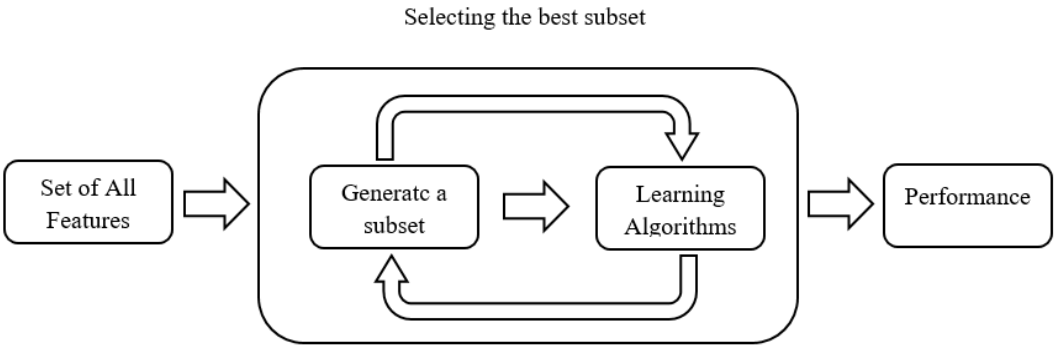


Figure 3.6: Process of a Wrapper Method [22]

3.9 Data Partitioning

Partitioning the dataset correctly is an important step in building a machine learning model. If the data is not partitioned correctly, the model can suffer from overfitting or underfitting problems. This study uses various data partitioning techniques to accurately evaluate the performance of the model.They are described below.

3.9.1 Hold-out Method

In our work, we used the Hold-out method to divide the dataset. This is actually the simplest and most common method. Here, the data is divided into two parts—a training set and a testing set. we used 80% of the data to train the model and checked how the model was performing with the remaining 20% of the data. The main purpose of this method is to train the model with the training part and see its performance on unseen data with the testing part.

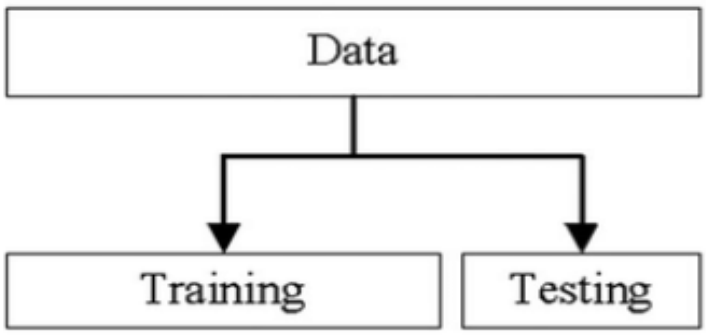


Figure 3.7: Hold-Out Method [23]

3.9.2 Stratified K-Fold Validation Method

We also used Stratified K-Fold Cross Validation in our work. It works like a regular K-Fold, but here the class distribution (ratio of Outcome = 0 and 1) in each fold is kept the same as in the previous dataset. Since the number of diabetic and non-diabetic patients in my diabetes dataset is not equal, using stratification was very necessary. This prevents the model from being biased and trains in a balanced way. I used k=10 here, so that the distribution of outcomes remains equal each time the model is trained and tested. This approach has made the model performance more reliable.

Figure 3.8: Stratified K-Fold Cross Validation Method [24]

3.9.3 K-Fold Cross Validation Method

In addition to the hold-out method, I also used K-Fold Cross Validation. In this method, the entire dataset is divided into k equal parts. Each time, one part is considered the testing set and the remaining k-1 parts are considered the training set. In this way, the model is trained and tested a total of k times. Finally, the performance of the model is determined by taking the average results of all of them. In this method, each data is used once for testing, so the model can be evaluated more reliably. For our diabetes dataset, we used k=5. This means that the dataset is divided into 5 parts and each time a different part is used as the test set. The advantage of this is that due to the split, the result that depends on the hold-out once, here it is verified repeatedly. Therefore, the performance of the model is found to be relatively more stable

Figure 3.9: K-Fold Cross Validation Method [23]

3.9.4 LOOCV (Leave-One-Out Cross Validation) Method

We also used Leave-One-Out Cross Validation (LOOCV) in our work. In this method, only one data point from the dataset is taken as the testing set each time and all the remaining data are used as the training set. In this way, each data point in the dataset is used for testing once.If the dataset has n data points, then in LOOCV the model has to be trained and tested a total of n times. As a result, each data point is used to validate the model, which helps to accurately measure the performance of the model.In the case of our diabetes dataset, this method gave very reliable model results. However, it took more time, because a separate model had to be trained for each data point. LOOCV works very well for small datasets.

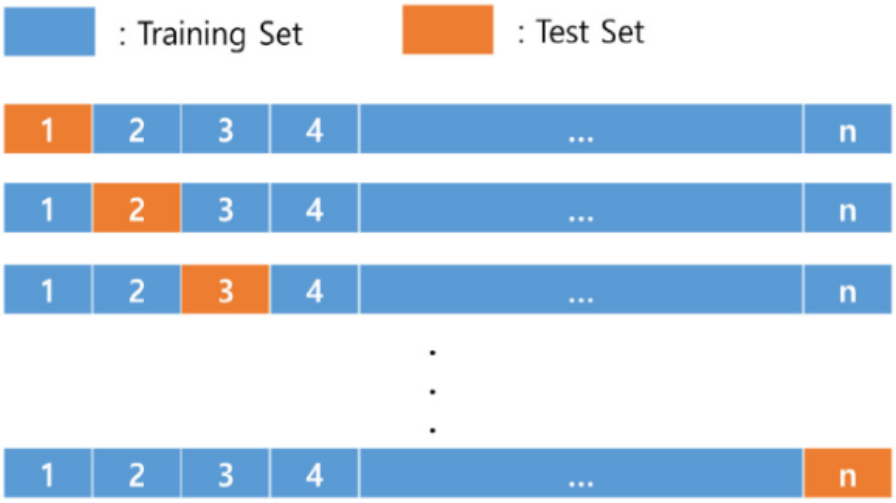


Figure 3.10: LOOCV Method [25]

3.10 Model Selection

In this study, we explored the effectiveness of various machine learning models for predicting diabetes. The selection of models was made to conduct a comparative analysis on the same dataset to ensure reliable and fair evaluation. By implementing and testing multiple algorithms under the same conditions, we aim to identify the most suitable model for accurate prediction and better generalization on unseen data.
We employed the following machine learning models for classification: Logistic Regression, Random Forest, XGBoost.

3.10.1 Logistic Regression(LR)

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables. Logistic regression predicts the output of a categorical dependent variable. Therefore, the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1. Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets. Logistic Regression can be used to classify the observations

using different types of data and can easily determine the most effective variables used for the classification [26].The below image is showing the logistic function:

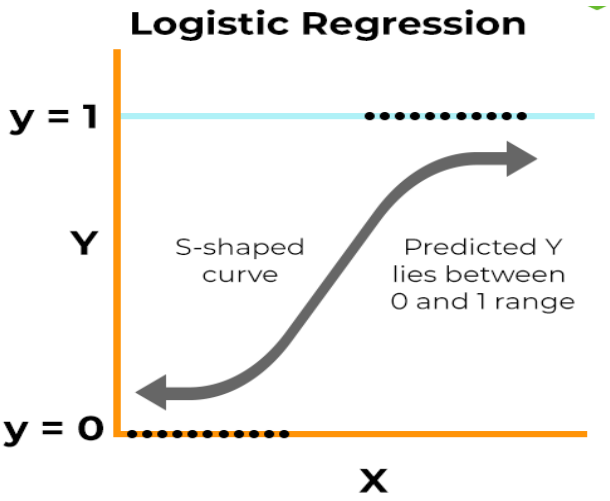


Figure 3.11 Logistic Regression

3.10.2 Random Forest (RF)

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 below image is showing the Random Forest

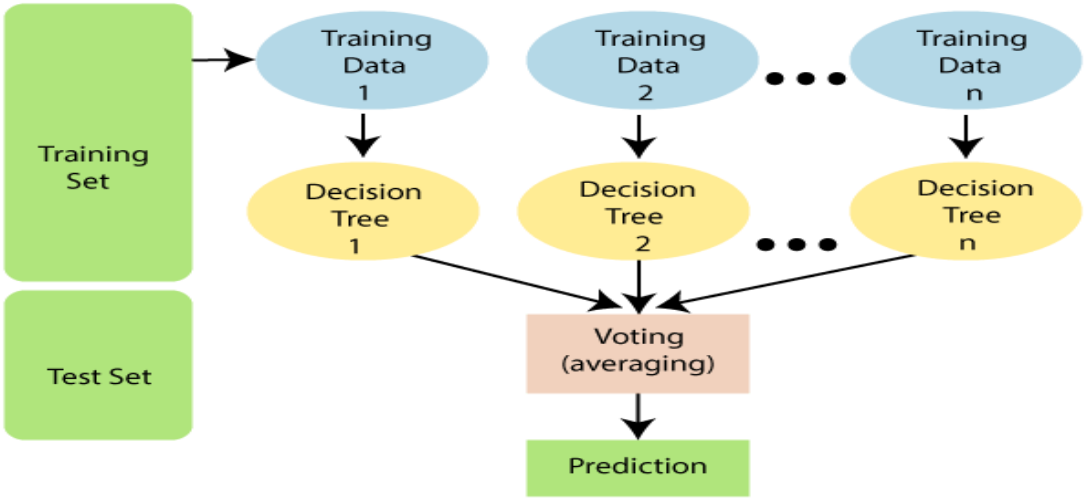


Figure 3.12: Random Forest

3.10.3 XGBoost

XGBoost (Extreme Gradient Boosting) is a highly efficient and scalable implementation of the Gradient Boosting framework. It is an ensemble learning technique that builds a strong predictive model by combining the outputs of multiple weak learners, typically decision trees.
The algorithm works by constructing decision trees sequentially, where each new tree attempts to correct the errors made by the previous ones. Unlike traditional boosting methods, XGBoost introduces several advanced features such as regularization (L1 and L2), parallel processing, tree pruning, and handling of missing values, which make it both faster and more accurate
XGBoost is widely used in machine learning competitions and real-world applications because of its ability to achieve high predictive performance while controlling overfitting. It supports both classification and regression tasks and is particularly effective for structured/tabular datasets. The below image is showing the Random Forest

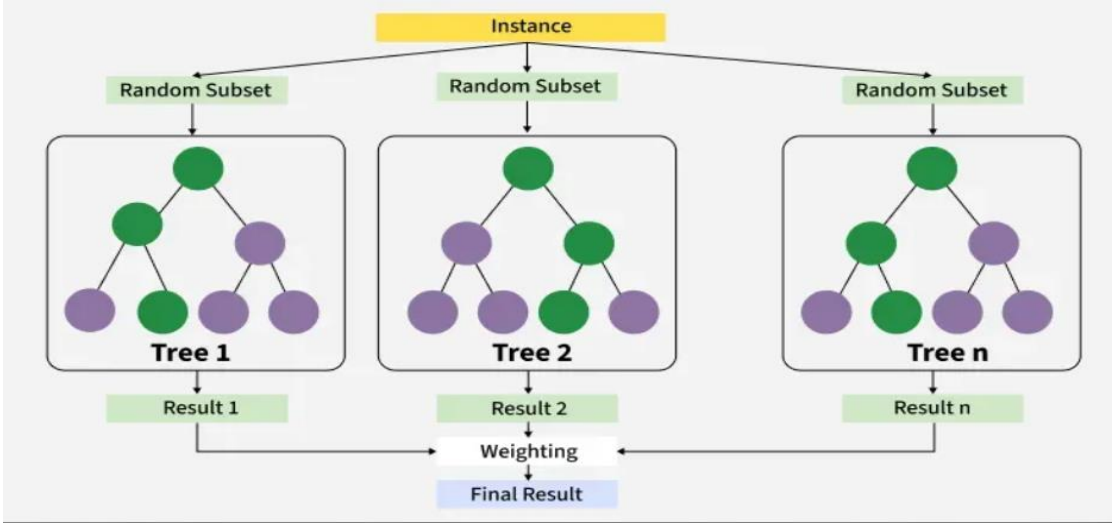


Figure 3.13: XGBoost [27]

3.11 Hyperparameter Tuning

Hyperparameter tuning basically means adjusting some important settings of a machine learning model, which are not directly learned from the data itself. These parameters are fixed in advance and control the overall structure or behavior of the learning process. Unlike model parameters hyperparameters need to be set before the training process starts. The choice of proper hyperparameter values can significantly influence the performance of a model, as they determine how well the model generalizes to unseen data.Hyperparameter tuning is the process of finding the most appropriate values of these hyperparameters so that the model achieves the best possible performance. Different strategies can be used for this purpose, such as Grid Search**,** Random Search, or more advanced methods like Bayesian Optimization**.** These approaches test different combinations of hyperparameters and evaluate the model performance using techniques like cross-validation.

3.11.1 Grid Search

In our work, Grid Search has been used for hyperparameter tuning. This is a popular method where a grid is created by determining the possible hyperparameter values of the model in advance. In fact, Grid Search systematically tests each hyperparameter combination to see in

which settings the model performs best. The results of each combination are measured by metrics such as accuracy, precision, recall or F1-score, and the one that gives the best result is chosen as the final hyperparameter.Although Grid Search can be computationally expensive, it provides a very reliable way to explore the parameter space thoroughly. By applying this method, we ensure that the chosen hyperparameters are not based on guesswork but on an objective and systematic evaluation process. As a result, the final model becomes more robust, consistent, and capable of achieving higher generalization performance on unseen data.Its main advantage is that all possibilities are tested, so no important settings are left out. However, it is very time-consuming and computationally expensive if the search space is large [28]

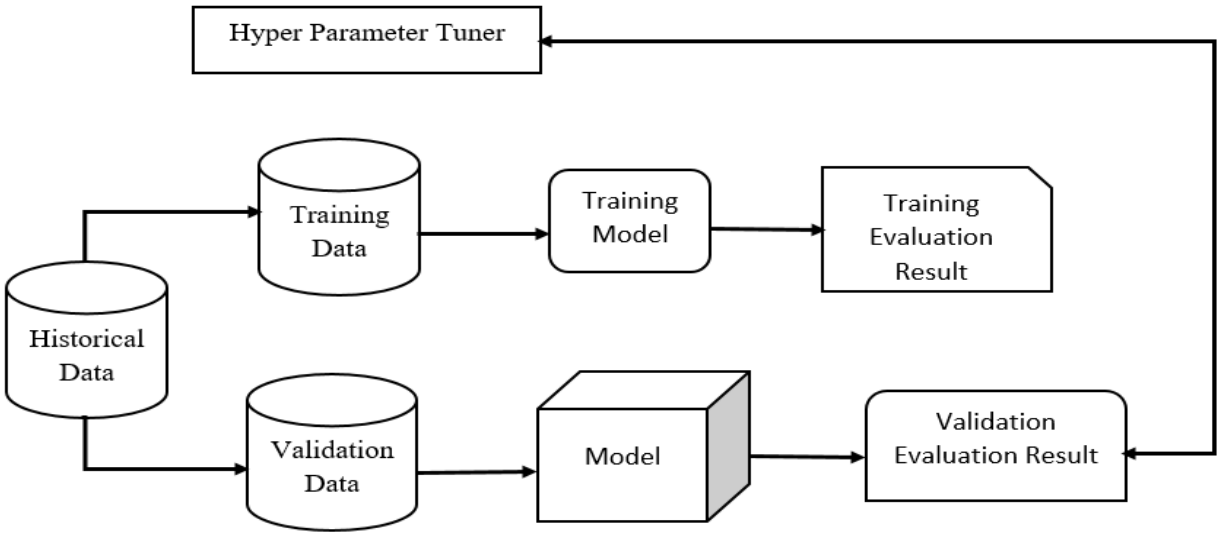


Figure 3.14: Grid-Search Architecture [29]

3.12 The hypermat parameters used in various machine learning algorithms are given in the table below

ML Model	Hyperparameter Name	Meaning
Logistic Regression (LG)	K	What are the most important features that will be used?
	C	Regularization parameter controlling the inverse of the regularizanon strength
	penalty	Type of regularization penalty (eg. "1" or "12")
	n_features_to_select	Number of top features to select using RFE
XGBoost	n_estimators	Number of decision trees
	max_depth	Maximum depth of the tree
	learning_rate	Step size shrinkage
	n_features_to_select	Number of top features selected by RFE

ML Model	Hyperparameter Name	Meaning
Random Forast (RF)	n_estimators	Number of decision trees in the forest.
	max_depth	Maximum depth of each decision tree.
	min_samples_split	Minimum number of samples required to split an internal node of a decision tree.
	min_samples_leaf	At least how many samples will there be in the leaf node?
	bootstrap	Will sampling be done with bootstrap or with the entire dataset?

Table 3.5: Hyperparameters of Different ML Model

3.13 Evaluation Metrics

This study evaluated model performance using various metrics, including precision (P), recall (R), weighted average F1-score (F), and accuracy (A), along with the confusion matrix, ROC curve. These metrics provide a comprehensive view of how well the model performs across both majority and minority classes. By analyzing multiple evaluation measures instead of relying only on accuracy, the study ensures a more reliable and balanced assessment of model effectiveness.

- **Precision (P)**

  Precision is the ability of a classification model to identify only relevant or correct data points. It indicates the accuracy of the model's positive predictions. Mathematically, precision is defined as:

$$Precision = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}}$$

  Here, True Positives (TP) are the data points that are actually positive and the model correctly predicted them as positive. On the other hand, False Positives (FP) are the data points that are actually negative but the model incorrectly predicted them as positive.

- **Recall (R)**

  Recall is a performance metric that measures how many of the true positive cases the model correctly identified as positive. In other words, it indicates the model's Sensitivity or True Positive Rate.

  Mathematically, recall is defined as:

$$Recall = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FP)}}$$

- **F1-Score (F)**

  F1-Score is a performance metric used to evaluate the accuracy of a classification model by combining Precision and Recall into a single value. It is the harmonic mean of Precision and Recall, meaning it balances both metrics and gives a better understanding of how well the model is performing, especially when the dataset is imbalanced.

$$F1 - Score = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **Accuracy (A)**

  Accuracy is a widely used performance metric that measures how many of the total predictions the model makes correctly. It refers to what percentage of all data points the model was able to correctly classify.
  Mathematically, accuracy is defined as:

$$Accuracy = \frac{\text{True Positives (TP)} + \text{True Negatives (TN)}}{\text{Total Number of Predictions}}$$

- **Confusion Matrix**

  A confusion matrix is a table that summarizes the performance of a classification model. It shows how well the model's predictions match the actual values in a dataset. This helps in understanding the types of errors the model makes and provides more detailed insights than just overall accuracy.

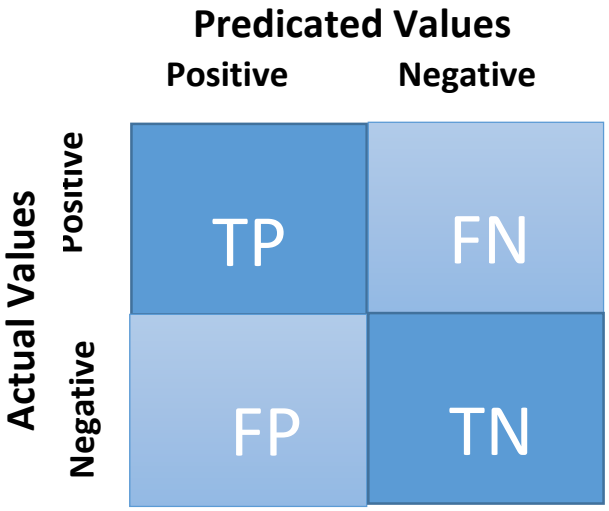


Figure 3.15: Confusion Matrix

Where,

- True Positive (TP) = Observation is positive and is predicted to be positive.
- False Negative (FN) = Observation is positive but is predicted negative.
- True Negative (TN) = Observation is negative and is predicted to be negative.
- False Positive (FP) =Observation is negative but is predicted positive.

- **ROC Curve/AUC**

  The Receiver Operating Characteristic (ROC) curve is a graphical tool used to evaluate the performance of a classification model. It represents the model's ability to distinguish between positive and negative classes by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR) across different threshold values. A model with strong discriminative ability will have a curve that bends sharply toward the top-left corner, indicating high sensitivity and low false alarm rate.

  The Area Under the Curve (AUC) provides a single numerical measure of overall model performance. An AUC value closer to 1 suggests that the model is highly effective at correctly classifying both positive and negative cases, whereas a value closer to 0.5 indicates performance no better than random guessing. This makes ROC and AUC essential tools for comparing and interpreting classification models.

Chapter 4

Results

4.1 Results Analysis

In this study, we are using three machine learning models for the prediction of diabetic patients. The models are Logistic Regression, Random Forest, XGBoost. For feature selection with the model, we used filter method, wrapper method and hold-out, k-fold cross validation, stratified k-fold cross validation method, leave-one-out cross validation method for data partition. For the analysis of the results, we are generating Classification Reports, ROC Curves, Cofusion Matrices. The combination performance of each model is described below.

4.2 Logistic Regression

The Table 4.1 below shows the performance of a combination of the Logistic Regression model with the Filter Method, Wrapper Method for feature selection and the Hold-out Method, k-fold cross validation, Stratified k-fold cross validation method, leave-one-out cross validation for data partitioning.

Model Name	**Feature Selection Method**	**Data Partitioning Method**	**Accuracy %**	**Precision(P) Recall(R) F1-Score(F)**	**ROC Curve Accuracy**
Logistic Regression	Filter Method	Hold-out Method	77%	P:(0)81%(1)70% R:(0)85%(1)64% F:(0)83%(1)55%	84%
		k-fold cross validation	77%	P:(0)81%(1)70% R:(0)85%(1)64% F:(0)83%(1)67%	82%
		Stratified k-fold cross validation method	71%	P:(0)75%(1)60% R:(0)82%(1)50% F:(0)78%(1)54%	81%
		leave-one-out cross validation	69%	P:(075%(1)57% R:(0)80%(1)50% F:(0)77%(1)53%	81%
	Wrapper Method	Hold-out Method	75%	P:(0)81%(1)70% R:(0)85%(1)64% F:(0)83%(1)67%	81%
		k-fold cross validation	77%	P:(0)85%(1)65% R:(0)79%(1)73% F:(0)82%(1)69%	85%
		Stratified k-fold cross validation method	78%	P:(0)81%(1)71% R:(0)87%(1)61% F:(0)84%(1)66%	84%
		leave-one-out cross validation	76%	P:(0)81%(1)70% R:(0)85%(1)64% F:(0)83%(1)67%	84%

Table 4.1: Logistic Regression Performance of Different Feature Selection and Data Partitioning Combinations

In the case of logistic regression models, different accuracy was obtained using different feature selection and data partitioning techniques. Using the Filter Method, the accuracy was 77% in both Hold-out and K-fold Cross Validation, 71% in Stratified K-fold and 69% in Leave-One-Out. On the other hand, using the Wrapper Method, the accuracy was 75% in Hold-out, 77% in K-fold Cross Validation, 78% in Stratified K-fold and 76% in Leave-One-Out. Overall, it can be seen that the Wrapper Method with Stratified K-fold Cross Validation provides the best accuracy 78% and Filter Method with Hold-Out provides the best accuracy 77%. The Confusion Matrix, Classification Report and Roc Curve of the combination of these two are described below.

- **Confusion Matrix of Logistic Regression Model using Filter Method with Hold-Out Method**

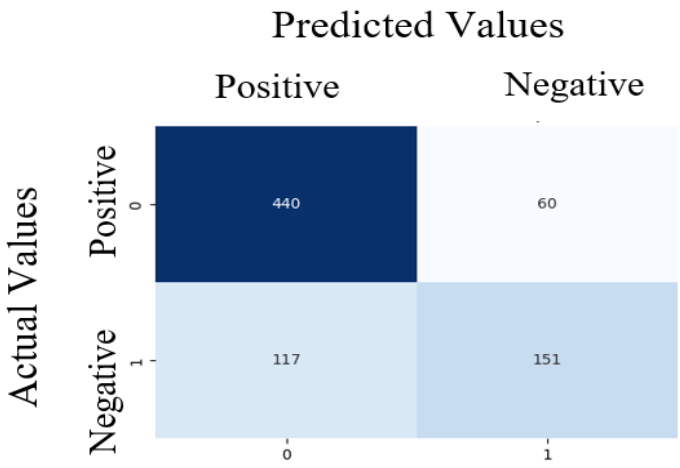


Figure 4.1: Confusion Matrix (Logistic Regression uses Filter Methods with Hold-Out methods)

It can be seen that the model correctly classified a total of 440 samples as class 0 and 151 samples as class 1 in this combination of filter methods with hold-out methods. However, the model incorrectly classified 60 samples of class 0 as class 1 and 117 samples of class 1 as class 0.

- **Confusion Matrix of Logistic Regression Model using Wrapper Method with Stratified K-fold Cross Validation**

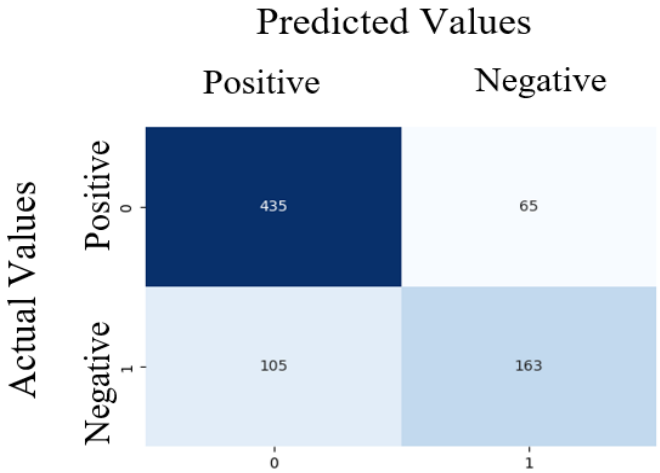


Figure 4.2: Confusion Matrix (Logistic Regression uses Wrapper Method with Stratified K-fold Cross Validation)

It can be seen that the model correctly classified a total of 435 samples as class 0 and 163 samples as class 1 in this combination of filter methods with hold-out methods. However, the model incorrectly classified 65 samples of class 0 as class 1 and 105 samples of class 1 as class 0.

- **Classification Reports of Logistic Regression Model using Filter Method with Hold-Out Method**

The classification report of this model shows that the precision, recall and f1-score for class 0 are around 0.79, 0.88 and 0.83**,** meaning that the model is able to capture class 0 quite well. The values for class 1 are a bit lower, around 0.72, 0.56 and 0.63, but still acceptable. The accuracy of the model is approximately 77%**,** so it can be said that the model is performing reasonably well for both classes and is giving a balanced outcome overall.

	Precision	**Recall**	**F1-score**	**Support**
0	0.79	0.88	0.83	500
1	0.72	0.56	0.63	268
Accuracy			0.77	768
Macro Avg	0.75	0.72	0.73	768
Weighted Avg	0.76	0.77	0.76	768

Table 4.2: Classification Reports(Logistic Regression uses Filter Methods with Hold-Out methods)

- **Classification Reports of Logistic Regression Model using Wrapper Method with Stratified K-fold Cross Validation**

The classification report of this model shows that the precision, recall and f1-score for class 0 are around 0.81, 0.87 and 0.84 with a support of 500 samples, meaning that the model is performing well in detecting class 0. On the other hand, the values for class 1 are slightly lower 0.71 precision, 0.61 recall and 0.66 f1-score with a support of 268 samples, which indicates that the model faces some difficulty in identifying class 1 correctly. The overall accuracy of the model is approximately 78%, so it can be said that the model is giving a reasonably balanced performance across both classes.

	Precision	**Recall**	**F1-score**	**Support**
0	0.81	0.87	0.84	500
1	0.71	0.61	0.66	268
Accuracy			0.78	768
Macro Avg	0.75	0.74	0.75	768
Weighted Avg	0.77	0.77	0.77	768

Table 4.3: Classification Reports(Logistic Regression uses Wrapper Method with Stratified K-fold Cross Validation)

- **ROC Curves of Logistic Regression Model using Filter Method with Hold-Out Method**

In Figure 4.3 The blue line in the plot represents the actual performance of the model. It shows a strong ability to classify between the positive and negative classes. The AUC value for this ROC curve is 0.84, which is considered a high value. This indicates that the model was able to correctly distinguish between the positive and negative classes in approximately 84% of cases.

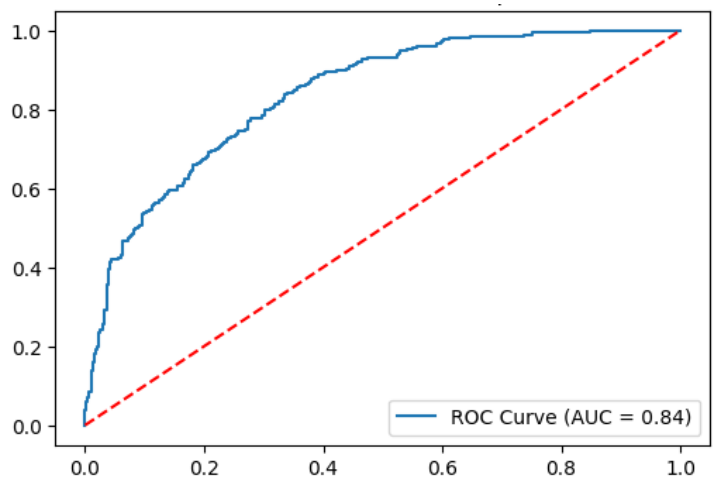
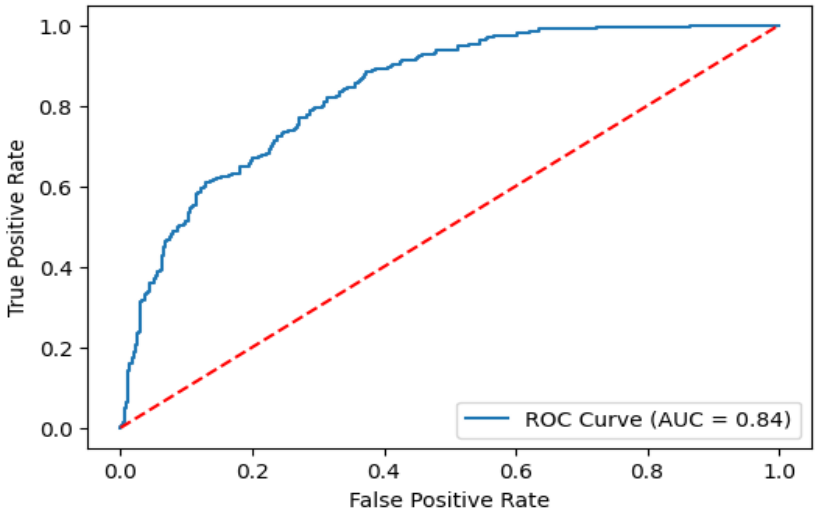


Figure 4.3: ROC Curves (Logistic Regression uses Filter Methods with Hold-Out methods)

- **ROC Curves of Logistic Regression Model using Wrapper Method with Stratified K-fold Cross Validation**

Based on the Figure 4.4 the blue line represents the actual performance of the model, which demonstrates strong classification ability. The AUC (Area Under the Curve) value for this ROC curve is 0.84, which is considered a very good score. This means the model was able to correctly distinguish between positive and negative classes in approximately 84% of cases



.Figure 4.4: ROC Curves (Logistic Regression uses Wrapper Method with Stratified K-fold Cross Validation)

4.3 XGBoost

The Table 4.4 below shows the performance of a combination of the XGBoost model with the Filter Method, Wrapper Method for feature selection and the Hold-out Method, k-fold cross validation, Stratified k-fold cross validation method, leave-one-out cross validation for data partitioning.

Model Name	Feature Selection Method	Data Partitioning Method	Accuracy %	Precision(P) Recall(R) F1-Score(F)	ROC Curve Accuracy
XGBoost	Filter Method	Hold-out Method	75%	P:(0)79%(1)67% R:(0)84%(1)59% F:(0)82%(1)63%	83%
		k-fold cross validation	77%	P:(0)80%(1)71% R:(0)87%(1)60% F:(0)83%(1)65%	83%
		Stratified k-fold cross validation method	73%	P:(0)75%(1)65% R:(0)85%(1)48% F:(0)80%(1)55%	81%
		leave-one-out cross validation	87%	P:(0)89%(1)81% R:(0)90%(1)80% F:(0)90%(1)81%	94%
	Wrapper Method	Hold-out Method	77%	P:(0)80%(1)70% R:(0)86%(1)59% F:(0)83%(1)64%	82%
		k-fold cross validation	73%	P:(0)76%(1)64% R:(0)85%(1)50% F:(0)80%(1)56%	82%
		Stratified k-fold cross validation method	78%	P:(0)81%(1)71% R:(0)87%(1)61% F:(0)84%(1)66%	83%
		Filter Method	77%	P:(0)81%(1)69% R:(0)85%(1)62% F:(0)83%(1)62%	83%

Table 4.4: XGBoost Performance of Different Feature Selection and Data Partitioning Combinations

In the case of logistic regression models, different accuracy was obtained using different feature selection and data partitioning techniques. Using the Filter Method, the accuracy was 77% in Hold-out, 77% K-fold Cross Validation, 73% in Stratified K-fold cross validation and 87% in Leave-One-Out. On the other hand, using the Wrapper Method, the accuracy was 77% in Hold-out, 73% in K-fold Cross Validation, 78% in Stratified K-fold cross validation and 77% in Leave-One-Out. Overall, it can be seen that the Filter Method with Leave-One-Out provides the best accuracy 87% and Wrapper Method with Stratified k-fold cross validation method provides the best accuracy 78%. The Confusion Matrix, Classification Report and ROC Curve of the combination of these two are described below.

- **Confusion Matrix of XGBoost Model using Filter Method with Leave-One-Out Cross Validation**

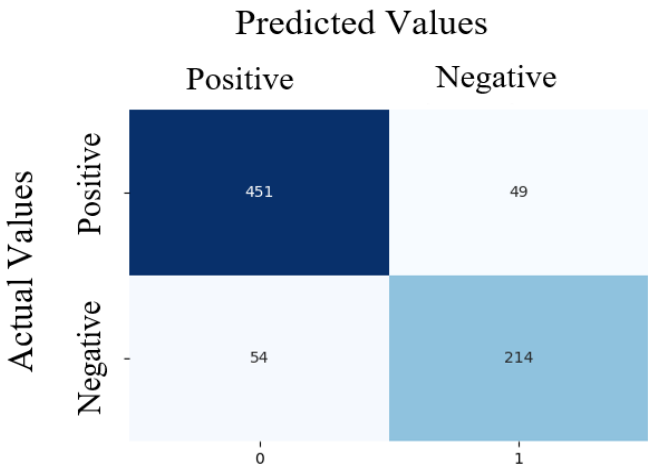


Figure 4.5: Confusion Matrix (XGBoost Model using Filter Method with Leave-One-Out Cross Validation)

It can be seen that the model correctly identified a total of 451 samples as class 0 and 214 samples as class 1. However, the model incorrectly identified 49 class 0s as class 1 and 54 class 1s as class 0.

- **Confusion Matrix of XGBoost Model using Wrapper Method with Stratified k-fold cross validation method**

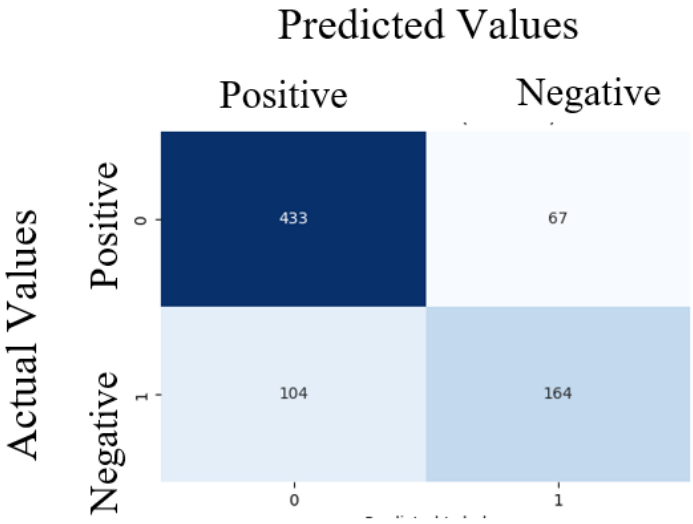


Figure 4.6: Confusion Matrix (XGBoost Model using Wrapper Method with Stratified k-fold cross validation method)

It can be seen that the model correctly identified a total of 433 samples as class 0 and 164 samples as class 1. However, the model incorrectly identified 67 class 0s as class 1 and 104 class 1s as class 0.

- **Classification Reports of XGBoost Model using Filter Method with Leave-One- Out Cross Validation**

The classification report of this model shows that the precision, recall and f1-score for class 0 are around 0.90, meaning that the model is able to capture class 0 well. The values for class 1 are a bit lower, around 0.81, but not bad. The accuracy of the model is approximately 87%, so it can be said that the model is doing well in both classes and is giving balanced performance.

	Precision	**Recall**	**F1-score**	**Support**
0	0.89	0.90	0.90	500
1	0.81	0.80	0.81	268
Accuracy			0.87	768
Macro Avg	0.85	0.85	0.85	768
Weighted Avg	0.87	0.87	0.87	768

Table 4.5: Classification Report(XGBoost Model using Filter Method with Leave-One- Out Cross Validation)

- **Classification Reports of XGBoost Model using Wrapper Method with Stratified k-fold cross validation method**

The classification report of this model shows that the precision, recall, and f1-score for class 0 are 0.81, 0.87, and 0.84 respectively, with a support of 500 samples. This means the model is performing well in detecting class 0.On the other hand, the values for class 1 are slightly lower. The precision is 0.71, recall is 0.61, and the f1-score is 0.66, with a support of 268 samples. This indicates that the model faces some difficulty in correctly identifying class 1.The overall accuracy of the model is approximately 78%. So, it can be said that the model is providing a reasonably balanced performance across both classes.

	Precision	**Recall**	**F1-score**	**Support**
0	0.81	0.87	0.84	500
1	0.71	0.61	0.66	268
Accuracy			0.78	768
Macro Avg	0.76	0.74	0.75	768
Weighted Avg	0.77	0.78	0.77	768

Table 4.6: Classification Report (XGBoost Model using Wrapper Method with Stratified k-fold cross validation method)

- **ROC Curves of XGBoost Model using Filter Method with Leave-One- Out Cross Validation**

In the Figure 4.7 below, As you can see, the blue line is well above the red line, which proves that the model performed well. The AUC value of this ROC Curve is 0.94, which is

a very high value. This means that the model was able to correctly distinguish between positive and negative classes in 94% of the cases

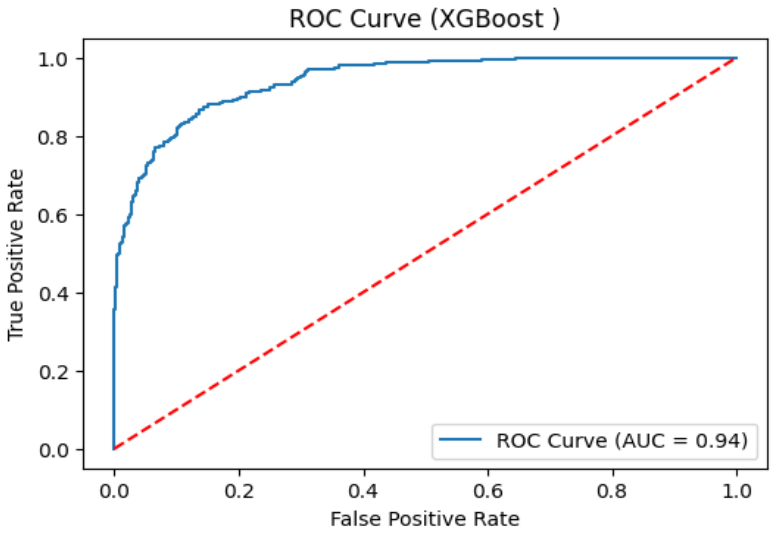


Figure 4.7: ROC Curves (XGBoost Model using Filter Method with Leave-One- Out Cross Validation)

- **ROC Curves of XGBoost Model using Filter Method with Leave-One- Out Cross Validation.**

In the Figure 4.8 below ,The blue line in the plot represents the model's actual performance, showing a strong ability to classify between the positive and negative classes. The AUC value for this ROC curve is 0.83, which is considered a very good score. This indicates that the model was able to correctly distinguish between positive and negative classes in approximately 83% of cases.

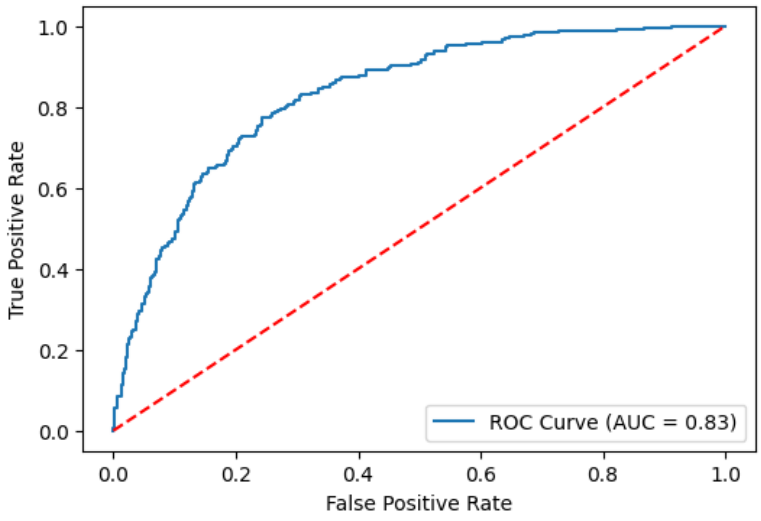


Figure 4.8: ROC Curves (XGBoost Model using Wrapper Method with Stratified k-fold cross validation method)

4.4 Random Forest

The Table 4.6 below shows the performance of a combination of the Random Forest model with the Filter Method, Wrapper Method for feature selection and the Hold-out Method, k-fold cross validation, Stratified k-fold cross validation method, leave-one-out cross validation for data partitioning.

Random Forest	Filter Method	Hold-out Method	76%	P:(0)78%(1)70% R:(0)87%(1)56% F:(0)83%(1)62%	83%
		k-fold cross validation	77%	P:(0)81%(1)69% R:(0)85%(1)62% F:(0)83%(1)65%	83%
		Stratified k-fold cross validation method	76%	P:(0)80%(1)68% R:(0)85%(1)62% F:(0)83%(1)63%	82%
		leave-one-out cross validation	77%	P:(0)79%(1)70% R:(0)87%(1)52% F:(0)82%(1)64%	83%
	Wrapper Method	Hold-out Method	74%	P:(0)77%(1)68% R:(0)87%(1)52% F:(0)82%(1)59%	83%
		k-fold cross validation	76%	P:(0)79%(1)67% R:(0)84%(1)59% F:(0)82%(1)63%	82%
		Stratified k-fold cross validation method	78%	P:(0)81%(1)70% R:(0)86%(1)62% F:(0)83%(1)66%	83%
		leave-one-out cross validation	76%	P:(0)80%(1)67% R:(0)84%(1)62% F:(0)82%(1)64%	82%

Table 4.7: Random Forest Performance of Different Feature Selection and Data Partitioning Combinations

In the case of Random Forest, different accuracy was obtained using different feature selection and data partitioning techniques. Using the Filter Method, the accuracy was 76% in both Hold-out and Stratified K-fold cross validation, 77% in both K-fold cross validation and Leave-One-Out cross validation. On the other hand, using the Wrapper Method, the accuracy was 74% in Hold-out, 76% in both K-fold Cross Validation and Leave-One-Out cross validation, 78% in Stratified K-fold cross validation, it can be seen that the Wrapper Method with Stratified K-fold Cross Validation provides the best accuracy 78% and Filter Method with k-fold cross validation provides the best accuracy 77%. The Confusion Matrix, Classification Report and ROC Curve of the combination of these two are described below.

- **Confusion Matrix of Random Forest Model using Filter Method with k-fold cross validation**

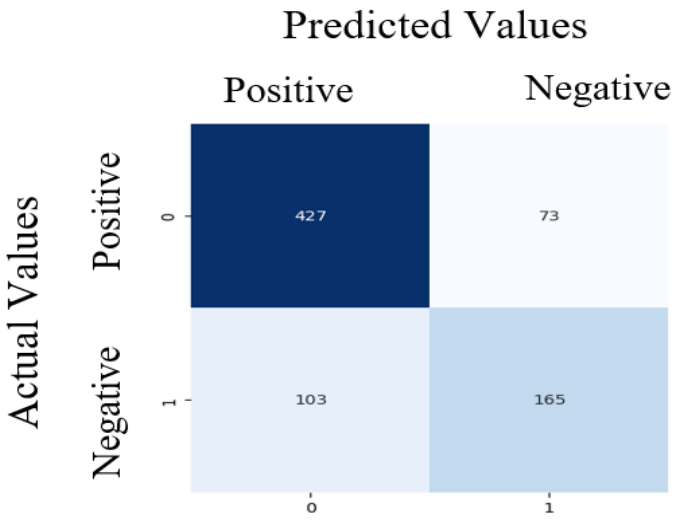


Figure 4.9: Confusion Matrix (Random Forest Model using Filter Method with k-fold cross validation method)

It can be seen that the model correctly identified a total of 427 samples as class 0 and 165 samples as class 1. However, the model incorrectly identified 73 class 0s as class 1 and 103 class 1s as class 0.

- **Confusion Matrix of Random Forest Model using Wrapper Method with Stratified K-fold Cross Validation**

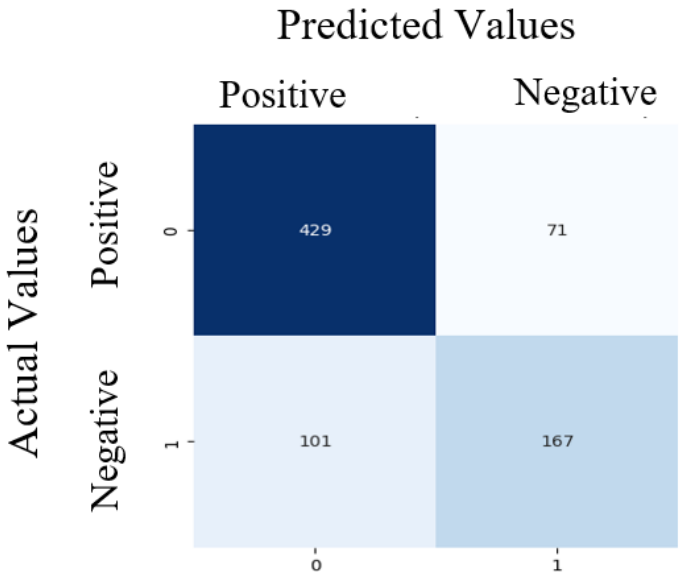


Figure 4.10: Confusion Matrix (Random Forest Model using Wrapper Method with Stratified K-fold Cross Validation)

It can be seen that the model correctly identified a total of 429 samples as class 0 and 167 samples as class 1. However, the model incorrectly identified 71 class 0s as class 1 and 101 class 1s as class 0.

- **Classification Reports of Random Forest Model using Filter Method with k-fold cross validation**

The model performs well in classifying class 0. This is shown by its strong precision of 0.81 and recall of 0.85, which results in a high f1-score of 0.83. There are 500 samples in this class.For class 1, the performance is not as strong. The precision is 0.69, and the recall is 0.62, leading to a lower f1-score of 0.65. The model struggles more with identifying this class, which has 268 samples.Overall, the model's accuracy is 0.77 (77%). While this is a decent score, the report highlights an imbalance in performance, as the model is more effective at identifying class 0 than class 1.

	Precision	**Recall**	**F1-score**	**Support**
0	0.81	0.85	0.83	500
1	0.69	0.62	0.65	268
Accuracy			0.77	768
Macro Avg	0.75	0.73	0.74	768
Weighted Avg	0.77	0.77	0.77	768

Table 4.8: Classification Report (Random Forest Model using Filter Method with k-fold cross validation method)

- **Classification Reports of Random Forest Model using Wrapper Method with Stratified K-fold Cross Validation**

The model shows strong performance in classifying class 0. This is evident from the high precision of 0.81, recall of 0.86, and f1-score of 0.83 for this class, which has a support of 500 samples.Conversely, the model's performance is not as good for class 1. The precision is lower at 0.70, and the recall is even lower at 0.62, leading to an f1-score of 0.66. The support for this class is 268 samples.Overall, the model's accuracy is 0.78 (or 78%). While this is a decent score, the report clearly shows an imbalance in performance, as the model is more effective at identifying and classifying class 0 than it is with class 1.

	Precision	**Recall**	**F1-score**	**Support**
0	0.81	0.86	0.83	500
1	0.70	0.62	0.66	268
Accuracy			0.78	768
Macro Avg	0.76	0.74	0.75	768
Weighted Avg	0.77	0.78	0.77	768

Table 4.9: Classification Report (Random Forest Model using Wrapper Method with Stratified K-fold Cross Validation)

- **ROC Curves of Random Forest Model using Filter Method with k-fold cross validation**

In the Figure 4.11 below,The blue line in the plot represents the model's actual performance, which shows a strong ability to classify between positive and negative cases. The AUC for this ROC curve is 0.83. This is a very good score, indicating that the model was able to correctly distinguish between the two classes in approximately 83% of cases.

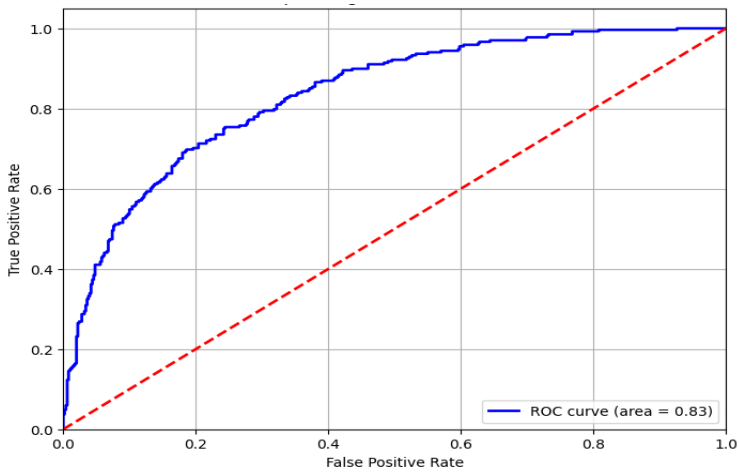


Figure 4.11: ROC Curves (Random Forest Model using Filter Method with Filter Method with k-fold cross validation)

- **ROC Curves of Random Forest Model using Wrapper Method with Stratified K-fold Cross Validation**

In the Figure 4.12 below,The blue line in the plot represents the model's actual performance, showing a strong ability to classify between the positive and negative classes. The AUC value for this ROC curve is 0.83, which is considered a very good score. This indicates that the model was able to correctly distinguish between positive and negative classes in approximately 83% of cases.

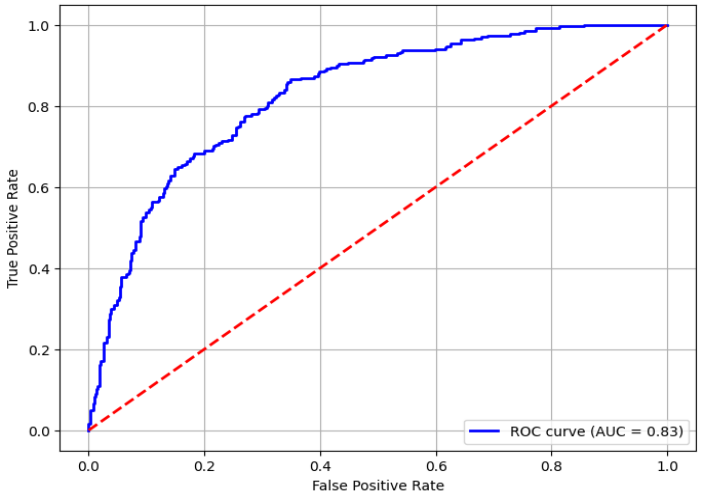


Figure 4.12: ROC Curves (Wrapper Method Model using Filter Method with Wrapper Method with Stratified K-fold Cross Validation)

4.5 Handling Class Imbalances

We are handling the classifier imbalance approach which other researchers have not done and after handling the new approach my previous result which was 87% significance is dropping, dropping to 84%.

- **Confusion Matrix**

  It can be seen that the model correctly identified a total of 411 samples as class 0 and 235 samples as class 1. However, the model incorrectly identified 89 class 0 samples as class 1 and 33 class 1 samples as class 0.

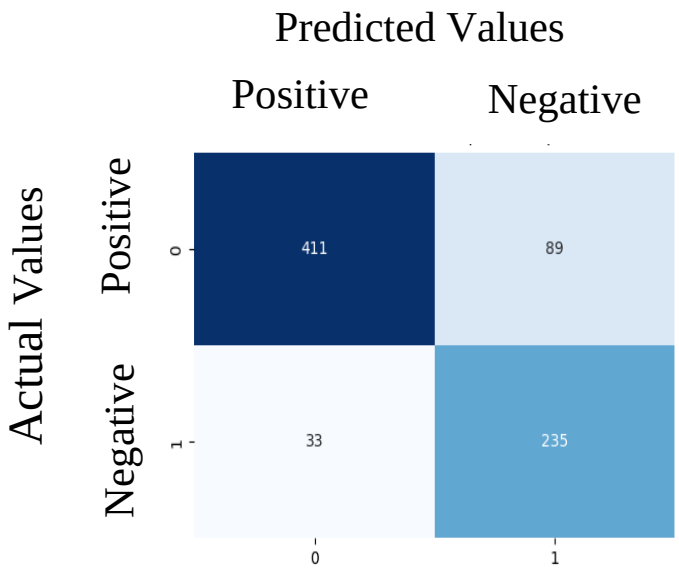


Figure 4.13: Confusion Matrix (XGBoost-SMOTE)

- **ROC Curves**

  In Figure 4.4, the blue line indicates the actual performance of the model, which achieved strong classification ability. The AUC value of this ROC curve is 0.93, which is considered very high. This indicates that the model was able to correctly distinguish between positive and negative classes in about 93% of the cases.

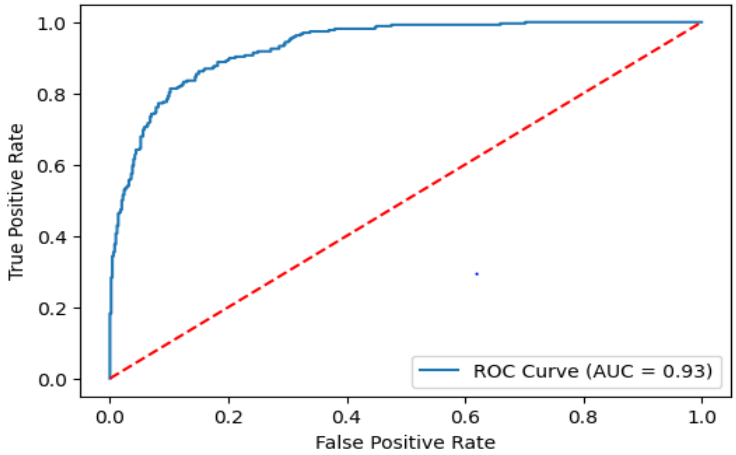


Figure 4.14: ROC Curves (XGBoost-SMOTE)

In this study, the SMOTE method was applied to solve the existing class imbalance problem in the dataset. While many studies have neglected this issue, this study has newly included it. We have done class imbalance in our study. The results show that the performance of the model has slightly decreased after imbalance handling where the accuracy was 87% earlier, it has dropped to 84%. When building a model for diabetes prediction, class imbalance is often observed in the dataset, where one class has more samples and the other has relatively less. As a result, the model is more inclined towards the larger class and ignores the smaller class, which affects the overall accuracy of the prediction.

4.6 Comparison with Previous Work

A review of previous studies has shown that various datasets and models have been used for diabetes prediction. Most of the studies have used Pima Indian Diabetes Dataset and applied models such as Logistic Regression, Random Forest, SVM, XGBoost etc. However, in many cases, class imbalance, feature selection and hyperparameter tuning have not been used properly. Our study has overcome these limitations. Data has been prepared using Missing value handling. In addition, the performance of the model has been improved through feature selection and hyperparameter tuning. The table below presents a comparison of our proposed model and previous studies.

Name	**Year**	**Used Model**	**Best Proposed Model**	**Performance**
Samala Kavya Sri.et al[30]	2025	1.Logistic Regression 2. Naive Bayes 3. K-Nearest Neighbors 4.Gradient Boosting Classifier	1.Gradient Boosting Classifier	98%
Emre Sedar Saygili.et al [31]	2025	1. Random Forest 2. XGBoost 3. LightGBM 4. logistic regression	1. Random Forest	94%
Nur Tri Ramadhanti Adiningrum. et al[11]	2025	1.Voting Classifier 2.Decision Tree 3.Random Forest 4.Navies Bayes 5.Logistic Regression	1.Voting Classifier	81%
Zhengyi Zhang[2]	2025	1.Logistic Regression 2.Random Forest 3.Support Vector Machine (SVM) 4. XGBoost	1. XGBoost	85%
Our Proposed Model	2025	1. Logistic regression. 2. Random Forest. 3. XGBoost	1.XGBoost	**87% 84%(with balanced class)**

Table 4.10: Comparative Analysis with Other Related Work the Using Same Dataset

Chapter 5

Conclusion and Future Works

5.1 Conclusion

This study uses and compares various machine learning algorithms for diabetes prediction. The Median method has been applied to solve problems such as missing values in the dataset. In addition, the performance of the models has been improved through hyperparameter tuning. The predictions have been made using Logistic Regression, Random Forest, XGBoost models. The experimental results showed that Performed best using Filter Method and Leave-One-Out Cross Validation method with XGboost Model and achieved the highest accuracy 87%, and the ROC Curve accuracy is 93%. In this combination, we have balanced the class through SMOTE, which previously dropped the result down to 84%. Especially, improved results have been obtained in metrics such as Recall and ROC-AUC, which are very important in diabetes detection. Overall, it can be said that machine learning based prediction systems can play an effective role in diabetes detection.

5.2 Future Works

Although this study showed promising results in machine learning-based diabetes prediction, there are still some areas that can be improved in the future. the dataset currently used was relatively limited. Using a larger and more diverse dataset in the future will further enhance the generalization ability of the model. only conventional machine learning algorithms were used here in the future, improved performance can be obtained by using models based on Deep Learning (such as Neural Networks, CNN, RNN, etc.).

References

[1] *Diabetes* (no date) *World Health Organization*. Available at: https://www.who.int/news-room/fact-sheets/detail/diabetes (Accessed: 24 July 2025).

[2] Zhang, Zhengyi. "Comparison of Machine Learning Models for Predicting Type 2 Diabetes Risk Using the Pima Indians Diabetes Dataset." Journal of Innovations in Medical Research 4, no. 1 (2025): 65-71.

[3] Rani, K. J. "Diabetes prediction using machine learning." International Journal of Scientific Research in Computer Science, Engineering and Information Technology 6, no. 4 (2020): 294-305.

[4] Soni, Mitushi, and Sunita Varma. "Diabetes prediction using machine learning techniques." International Journal of Engineering Research & Technology (IJERT) 9, no. 9 (2020): 921-925.

[5] Arwatki Chen Lyngdoh, Nurul Amin Choudhury, Soumen Moulik, "Diabetes Disease Prediction Using Machine Learning Algorithms", IEEE EMBS Conference on Biomedical Engineering and Sciences (IECBES) 2020, DOI: 10.1109/IECBES48179.2021.9398759.

[6] Febrian, Muhammad Exell, Fransiskus Xaverius Ferdinan, Gustian Paul Sendani, Kristien Margi Suryanigrum, and Rezki Yunanda. "Diabetes prediction using supervised machine learning." *Procedia Computer Science* 216 (2023): 21-30.

[7] Sisodia, Deepti, and Dilip Singh Sisodia. "Prediction of diabetes using classification algorithms." *Procedia computer science* 132 (2018): 1578-1585.

[8] G. Tripathi and R. Kumar,"Early Prediction of Diabetes Mellitus Using Machine Learning", 2020 8th International Conference on Reliability Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), 2020.

[9] Madhu, Bhukya, Veerender Aerranagula, Riyaz Mahomad, V. Ravindernaik, K. Madhavi, and Gopal Krishna. "Techniques of Machine Learning for the Purpose of Predicting Diabetes Risk in PIMA Indians." In E3S Web of Conferences, vol. 430, p. 01151. EDP Sciences, 2023.

[10] El Massari, Hakim, Zineb Sabouri, Sajida Mhammedi, and Noreddine Gherabi. "Diabetes prediction using machine learning algorithms and ontology." Journal of ICT Standardization 10, no. 2 (2022): 319-337.

[11] Adiningrum, Nur Tri Ramadhanti, and Nisa Hanum Harani. "ANALISIS PERBANDINGAN ENSEMBLE MACHINE LEARNING DENGAN TEKNIK SMOTE UNTUK PREDIKSI DIABETES." *JEIS: Jurnal Elektro dan Informatika Swadharma* 5, no. 1 (2025): 121-130.

[12] Zhou, Hejia, Saifur Rahman, Maia Angelova, Clinton R. Bruce, and Chandan Karmakar. "A robust and generalized framework in diabetes classification across heterogeneous environments." Computers in Biology and Medicine 186 (2025): 109720.

[13] Rao, TN Srinivas, Shaik Azad, D. Yashwanth, A. Sai Tilak, G. Karthik Reddy, and Y. Bhaskar Reddy. "An Optimized Hybrid Ensemble Machine Learning Model for Accurate Diabetes Prediction and Early Diagnosis." Macaw International Journal of Advanced Research in Computer Science and Engineering 10, no. 1s (2024): 16-23.

[14] Hossain, Mohammad Raquibul, Md Jamal Hossain, Md Mijanoor, and Mohammad Manjur Alam Rahman. "Machine Learning Based Prediction and Insights of Diabetes Disease: Pima Indian and Frankfurt Datasets." Journal of Mechanics of Continua and Mathematical Sciences 20, no. 1 (2025).

[15] Kusam, Minal Mangesh, and Vaishnavi Bovane. "Diabetes Prediction Using Machine Learning Techniques." Available at SSRN 5371065 (2025).

[16] Kalaivani, B., A. Chandrasekhar, P. Sangeetha, M. Renukadevi, and M. Prakash. "Early Diabetes Detection using Novel Hybird Machine Learning Algorithm." In The 2025 International Conference on Advanced Research in Electronics and Communication Systems (ICARECS-2025), pp. 423-433. Atlantis Press, 2025.

[17] Deshmukh, Shriya. "SHAP-Based Explainable Framework for Disease Prediction and Comorbidity Risk Assessment in Healthcare." Available at SSRN 5368585 (2025).

[18] Priyatma, Johanes Eka, and Mikael Raditya Agung Sasmita. "Comparative Analysis of Random Forest and Support Vector Machine for Classifying Pima Indians Diabetes Dataset."

[19] GeeksforGeeks, "Median in Statistics," *GeeksforGeeks*, Dec. 11, 2018. https://www.geeksforgeeks.org/maths/median/

[20] Duch, Włodzisław. "Filter methods." In Feature extraction: foundations and applications, pp. 89-117. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006.

[21] Le, Tuan Minh, Thanh Minh Vo, Tan Nhat Pham, and Son Vu Truong Dao. "A novel wrapper–based feature selection for early diabetes prediction enhanced with a metaheuristic." IEEE access 9 (2020): 7869-7884.

[22] Kumar, Lalit, and Kusum Kumari Bharti. "A novel hybrid BPSO–SCA approach for feature selection." *Natural Computing* 20, no. 1 (2021): 39-61.

[23] Aladwani, Faisal, and Adel Elsharkawy. "Improved prediction of heavy oil viscosity at various conditions utilizing various supervised machine learning regression." *Petroleum Science and Technology* 41, no. 4 (2023): 406-424.

[24] Duan, Xiong. "Automatic identification of conodont species using fine-grained convolutional neural networks." Frontiers in Earth Science 10 (2023): 1046327.

[25] Cha, Gi-Wook, Hyeun Jun Moon, Young-Min Kim, Won-Hwa Hong, Jung-Ha Hwang, Won-Jun Park, and Young-Chan Kim. "Development of a prediction model for demolition waste generation using a random forest algorithm based on small datasets." International Journal of Environmental Research and Public Health 17, no. 19 (2020): 6997.

[26] Sperandei, Sandro. "Understanding logistic regression analysis." *Biochemia medica* 24, no. 1 (2014): 12-18.

[27]GeeksforGeeks. 2021. "XGBoost." GeeksforGeeks. September 18, 2021. https://www.geeksforgeeks.org/machine-learning/xgboost/.

[28] Anggreani, Desi. "Grid Search Hyperparameter Analysis in Optimizing The Decision Tree Method for Diabetes Prediction." Indonesian Journal of Data and Science 5, no. 3 (2024): 190-197.

[29] Nagaraj, P., V. Muneeswaran, K. Muthamil Sudar, A. Naga Vardhan Reddy, G. Deshik, and C. Charan Kumar Reddy. "Ensemble machine learning (grid search & random forest) based enhanced medical expert recommendation system for diabetes mellitus prediction." In 2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC), pp. 757-765. IEEE, 2022.

[30] Sri, Samala Kavya, Penta Ashok Kumar, Valluri Divya, Chokkakula Raja Rajeswari, and Reesu Sravani. "A Predictive Model for Diabetes Diagnosis Using Machine Learning." In *2025 6th International Conference on Data Intelligence and Cognitive Informatics (ICDICI)*, pp. 1-4. IEEE, 2025.

[31] Saygili, Emre Sedar, Adnan Batman, and Ersen Karakilic. "Predicting stimulated C-peptide in type 1 diabetes using machine learning: a web-based tool from the T1D exchange registry." Diabetes Research and Clinical Practice (2025): 112453.

[32] Chellappan, Dinesh, and Harikumar Rajaguru. "Generalizability of machine learning models for diabetes detection a study with nordic islet transplant and PIMA datasets." *Scientific Reports* 15, no. 1 (2025): 4479.

[33] Ramana, K. Seshadri, N. Asra Shaheen, S. Safa Chowdary, Shaik Afroz Jaha, Syeda Uzma Tasneem, and Talari Renuka. "Supervised Machine Learning Approach for Diabetes Detection and the Impact." Advances in Artificial Intelligence and Machine Learning: Proceedings of ERCICAM 2024, Volume 2 1335 (2025): 81.

[34] Paul, Tonmoy Kumar, Md Abdul Based, Md Zubair Rahman, and Sad Shariar Helali. "A Comparative Analysis of Machine Learning Models in Advanced Diabetes Diagnosis." In 2025 6th International Conference for Emerging Technology (INCET), pp. 1-6. IEEE, 2025.

[35] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). *SMOTE: Synthetic Minority Over-sampling Technique.* Journal of Artificial Intelligence Research, 16, 321–357.