Diabetes Prediction using Machine Learning

A dissertation submitted in partial fulfillment of the requirement for the Degree of B.Sc (Engineering) in Information & Communication Technology

Submitted by

ID: 12009021

ID: 12009030

Session: 2019-2020

Department of

Information and Communication Technology

Faculty of Engineering

Comilla University

Cumilla-3506, Bangladesh

Submission Date: February 2025

ABSTRACT

Particularly in countries like Bangladesh, where delayed diagnosis and limited healthcare resources exacerbate its impact. Traditional diagnostic methods are often costly and time-consuming, leading to late-stage detection and increased health complications. This study explores the application of machine learning techniques for diabetes prediction, leveraging a dataset comprising key clinical parameters such as blood glucose levels, BMI, and HbA1c, as well as personal information parameters such as age, gender, smoking history, hypertension and heart disease. Two datasets were used to gain a better understanding of the patterns among Bangladeshi diabetic patients. One dataset consists of collected data, while the other is a combination of the collected data and a dataset from Kaggle. Various machine learning models, including Logistic Regression, Decision Trees, Random Forest, Support Vector Machines (SVM), and XGBoost, were evaluated for their predictive accuracy. Experimental results of the combined datasets indicate that ensemble models, particularly Random Forest and XGBoost, achieved the highest accuracy, exceeding 97% precision. The findings highlight the potential of AI-driven predictive analytics in enhancing diagnosis, optimizing resource allocation, and supporting data-driven decision-making in healthcare. Future advancements in this field may integrate only Bangladeshi diabetic patients' dataset from wearable devices and electronic medical records, paving the way for multi-disease prediction and improved patient outcomes.

ACKNOWLEDGEMENT

We would like to begin by expressing our deepest gratitude to our respected project supervisor, for his invaluable guidance, patience, and unwavering support throughout our research on "Diabetes Prediction Using Machine Learning." From the very beginning, his insightful suggestions and constructive feedback have played a crucial role in shaping our project, refining our methodologies, and improving our understanding of machine learning applications in healthcare. His encouragement and dedication to nurturing innovation have truly inspired us to think critically, work diligently, and strive for excellence.

We are also sincerely thankful to the Department of Information and Communication Technology, Comilla University**,** for providing us with a strong academic foundation, essential resources, and a stimulating learning environment. The knowledge and skills we have gained during our time at the university have been instrumental in executing this project successfully.

A special note of appreciation goes to Cumilla Diabetes Hospital and all the individuals who contributed real-world data to our research. Their willingness to share valuable medical information has allowed us to enhance our predictive model and ensure its relevance in real-life applications

Lastly, we would like to extend our appreciation to each other as research partners. As a team of two, we have worked side by side, sharing ideas, overcoming obstacles, and pushing each other towards success. This project has been a test of our determination, problem-solving abilities, and teamwork, and we are proud of what we have accomplished together. The journey of developing "Diabetes Prediction Using Machine Learning" has been both challenging and rewarding, and we believe that our work will contribute positively to the field of artificial intelligence in healthcare.

LIST OF FIGURES

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

The chronic metabolic disease known as diabetes mellitus (DM) is typified by high blood glucose levels brought on by the body's incapacity to make or efficiently use insulin. Type 1 diabetes, which is caused by the autoimmune destruction of insulin-producing pancreatic cells, Type 2 diabetes, which is mainly associated with insulin resistance and lifestyle factors, and gestational diabetes, which develops during pregnancy and raises the risk of Type 2 diabetes later in life, are the three main types of the disease. Serious side effects from uncontrolled diabetes might include kidney failure, neuropathy, cardiovascular disorders, and vision impairment.

1.1 Prevalence and Challenges of Diabetes in Bangladesh

Rapid urbanization, sedentary lifestyles, and dietary changes have all contributed to the rising prevalence of diabetes in Bangladesh, making it a serious public health concern. More than 13 million individuals in the nation have diabetes, according to recent epidemiological research, and this number is predicted to increase dramatically over the next several years. The disease's increasing impact is caused by a confluence of factors, including hereditary predisposition, ignorance, restricted access to medical care, and delayed diagnosis. The healthcare system is further burdened by the large number of people who go undetected or only seek treatment after suffering from serious problems.

Despite initiatives to manage diabetes, Bangladesh's healthcare system has a difficult time making prompt and precise diagnosis. Conventional diagnostic methods use clinical tests as oral, fasting blood glucose, and healthcare cost HbA1c assessments and oral glucose tolerance tests can be resource- and time-intensive. Delays in diagnosis frequently lead to late-stage detection, which raises the possibility of problems and medical expenses.

1.2 Application of Machine Learning in Diabetes Prediction

More effective and precise illness prediction models are now possible because to developments in data-driven technology. A branch of artificial intelligence called machine learning has demonstrated significant promise in the field of healthcare, especially in predictive analytics for e disease identification. Machine learning algorithms can improve diagnostic accuracy and facilitate prediction by identifying patterns and risk factors linked to diabetes through the analysis of vast amounts of patient data.

Numerous machine learning methods have been used extensively for diabetes prediction, such as Decision Trees, Random Forest, Support Vector Machines (SVM), and Logistic Regression. These models accurately categorize people as either diabetic or non-diabetic based on important clinical markers as blood pressure, glucose levels, BMI, and HbA1c level. Machine learning-based models provide improved predictive capabilities, decreased human error, and faster processing than traditional diagnostic techniques.

1.3 Objective of the Study

With an emphasis on their use in Bangladesh's healthcare system, this study attempts to assess how well machine learning algorithms predict diabetes. The study aims to enhance diagnosis, optimize resource allocation, and assist healthcare professionals in making data-driven decisions by utilizing real-life data and sophisticated computational approaches. The results of this study may help create screening instruments that are easier to use and more effective, which would lessen the burden of complications from diabetes in Bangladesh.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Diabetes prediction using machine learning has been widely explored in recent years, with researchers employing various classification algorithms to enhance accuracy and early diagnosis. Several studies have utilized datasets such as the PIMA Indian Diabetes Dataset (PIDD) and hospital-collected data to evaluate the performance of classifiers like Decision Tree, Naïve Bayes, Support Vector Machine (SVM), Artificial Neural Networks (ANN), and ensemble methods like Random Forest and XGBoost. Yasodha et al. [1] and Aljumah et al. [2] applied classification techniques using hospital datasets and the PIMA dataset, demonstrating the effectiveness of machine learning in diabetes detection. Studies by Vijiya Kumar et al. [3], Anjali C. et al. [4], and Sonar et al. [5] further compared multiple classifiers, highlighting Decision Tree and Naïve Bayes as high-performing models.

Advanced methodologies have also been introduced, such as genetic programming [7] and ensemble models [9], improving prediction reliability. Nai-arun et al. [8] explored bagging and boosting techniques, while Sisodia et al. [6] demonstrated the robustness of Naïve Bayes in diabetes classification. Additionally, KNN-based approaches were analyzed in [10], showing that Naïve Bayes outperformed KNN in predictive accuracy.

2.2 Related Works

To determine if a person has diabetes or not, Yasodha et al. [1] employ classification on a variety of dataset types. Data from a hospital warehouse, comprising 200 instances with nine attributes, is gathered to create the dataset for diabetes patients. These cases fall into two categories: urine and blood tests. The dataset is evaluated using a 10-fold cross-validation technique, which works well on tiny datasets, and WEKA is used for the implementation in order to classify the data. The classifiers Naïve Bayes, J48, REP Tree, and Random Tree are used to compare the results. According to the study's findings, J48 performs the best, with an accuracy of 60.2%.

By employing classification techniques, specifically Decision Tree and Naïve Bayes algorithms, to analyze patterns in the data, Aljumah et al. [2] hope to find solutions for diabetes detection. The study suggests a quicker and more effective way to detect the illness, which would enable early treatment. Using the PIMA dataset and a cross-validation methodology, the study discovered that the Naïve Bayes classifier achieved an accuracy of 79.5% with a 70:30 split, whereas the J48 algorithm achieved an accuracy rate of 74.8%.

K. Vijiya Kumar et al. [3] proposed a Random Forest algorithm for diabetes prediction, developing a system capable of early detection with high accuracy. The proposed framework delivers the best outcomes for diabetes prediction, proving that the system can forecast the disease effectively, efficiently, and instantaneously.

Anjali C. and Veena Vijayan V. [4] discussed about diabetes as a condition caused on by elevated blood sugar levels. Diabetes can be predicted and diagnosed using a variety of computerized information systems that use classifiers including Decision Tree, SVM, Naïve Bayes, and ANN algorithms. Model development is based on categorization approaches employing these algorithms, where Decision Tree models achieved 85% accuracy, Naïve Bayes 77%, and Support Vector Machine (SVM) 77.3%. The PIDD dataset was employed in the study, and the results show that these approaches are highly accurate.

Using classifiers like Decision Tree, SVM, Naïve Bayes, and ANN algorithms, Priyanka Sonar and Prof. K. JayaMalini et al. [5] developed a model based on categorization techniques. Decision Tree models achieved 85% precision, Naïve Bayes 77%, and Support Vector Machine (SVM) 77.3%. The PIDD dataset was used in the study, and the results show that these approaches are highly accurate.

To predict diabetes as accurately as possible, Sisodia et al. [6] used three distinct machine learning classifiers: Naïve Bayes (NB), SVM, and Decision Tree (DT). With an AUC of 0.819, their research showed that the Naïve Bayes classifier was the top-performing model. Furthermore, Sisodia et al. [18] discovered that the NB classifier has superior accuracy at 76.30% when compared to other machine learning techniques used on PIDD.

Pradhan and Bamnote [7] proposed a genetic programming approach for diabetes prediction, which outperformed other implemented techniques. Additionally, Nai-arun and Moungmai [8] used four machine learning methods—Decision Tree, ANN, Logistic Regression (LR), and Naïve Bayes—to classify diabetes risk, applying bagging and boosting techniques to increase robustness, and their experimental results showed that the Random Forest algorithm produced the best performance.

The authors of [9] discussed about how the proposed ensemble model from the PID dataset was used to predict diabetes, and how preprocessing is essential to a reliable and accurate prediction. With sensitivity, specificity, false omission rate, diagnostic odds ratio, and AUC of 0.789, 0.934, 0.092, 66.234, and 0.950, respectively, proposed ensembling classifier outperformed the others based on thorough testing. In AUC, this performs 2.00% better than the state-of-the-art findings. The suggested diabetes prediction paradigm outperforms the other approaches covered in the paper.

In order to predict diabetes based on a number of health characteristics in the dataset using supervised machine learning, in [10], the authors compared two k-Nearest Neighbor algorithms with the Naive Bayes algorithm. Their studies and the Confusion Matrix evaluation of the method show that the Naive Bayes algorithm works better than KNN. With an average of 76.07 percent accuracy, 73.37 percent precision, and 71.37 percent recall in Naive Bayes and an average of 73.33 percent accuracy, 70.25 percent precision, and 69.37 percent recall in KNN, the Naive Bayes algorithm performs better than KNN, according to the results of our experiments and an evaluation of the algorithm using Confusion Matrix.

CHAPTER 3

ALGORITHM, THEORY, AND TOOLS

3.1 Introduction

This chapter discusses the machine learning algorithms used in this study, the theoretical background behind them, and the tools employed for implementation. The primary objective is to provide an understanding of how these algorithms contribute to diabetes prediction.

3.2 Machine Learning Algorithms Used

3.2.1 Logistic Regression

A classification approach for supervised learning is logistic regression. Based on one or more predictors, it is used to calculate the likelihood of a binary response. They may be discrete or continuous. When we wish to categorize or separate certain data items into groups, we employ it. It categorizes the data in binary form, meaning that it only uses the numbers 0 and 1, which indicate whether a patient has diabetes or not. Finding the optimal fit, which describes the relationship between the target and predictor variables, is the primary goal of logistic regression. The linear regression model serves as the foundation for logistic regression. The sigmoid function is used by the logistic regression model to forecast the likelihood of both positive and negative classes.

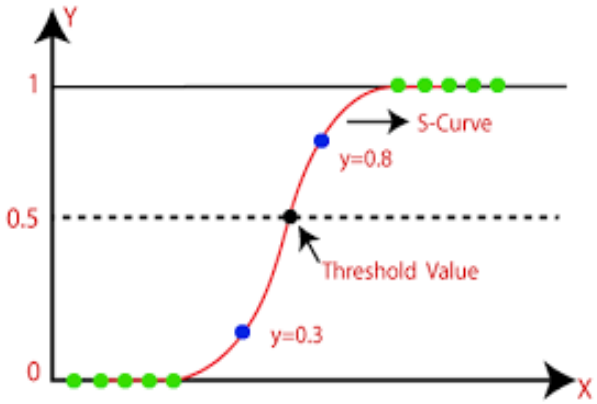

Figure3.1: Logistic Regression

3.2.2 Decision Tree

One fundamental classification technique is the decision tree. It is a method of supervised learning. When the response variable is categorical, a decision tree is employed. A decision tree is a model that shows the categorization process based on input features and has a tree-like structure. Types of input variables include text, graphs, discrete variables, continuous variables, and more.

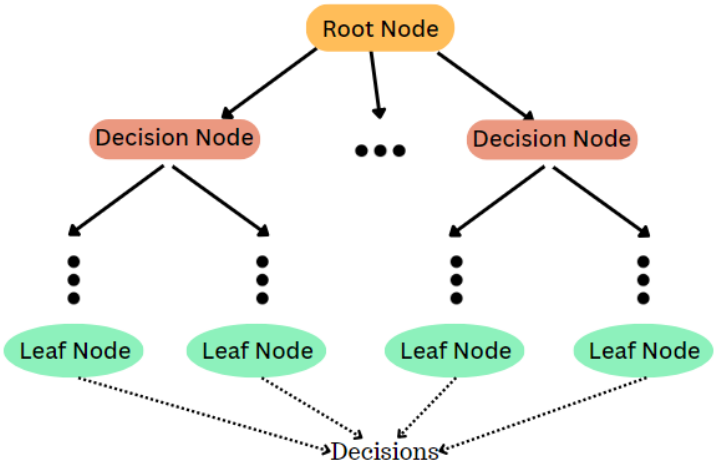

Figure 3.2: Decision Tree

3.2.3 Random Forest

It is type of ensemble learning method and also used for classification and regression tasks.

The accuracy it gives is grater then compared to other models. This method can easily handle large datasets. Random Forest is developed by Leo Bremen. It is popular ensemble Learning Method. Random Forest Improve Performance of Decision Tree by reducing variance. It operates by constructing a multitude of decision trees at training time and outputs the class that is the mode of the classes or classification or mean prediction (regression) of the individual trees.

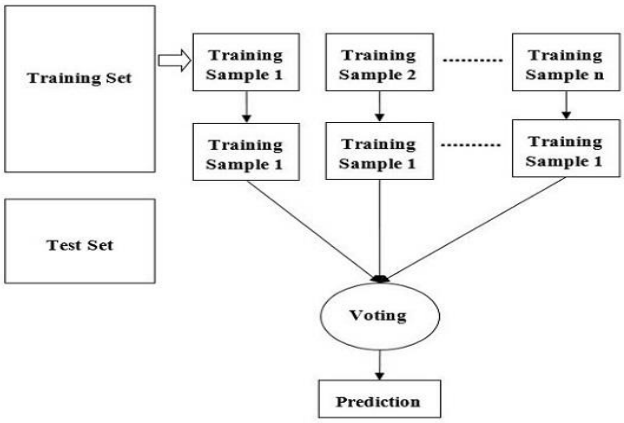

Figure 3.3: Random Forest

3.2.4 Support Vector Machine (SVM)

Support Vector Machine also known as SVM is a supervised machine learning algorithm. The most often used classification method is SVM. Two classes are separated by a hyperplane created by SVM. In high-dimensional space, it can produce a hyperplane or a collection of hyperplanes. Regression and classification are two more uses for this hyperplane. In addition to classifying things that are not supported by data, SVM distinguishes instances within particular classes. Hyperplanes are used to execute separation to the nearest training point for each class.

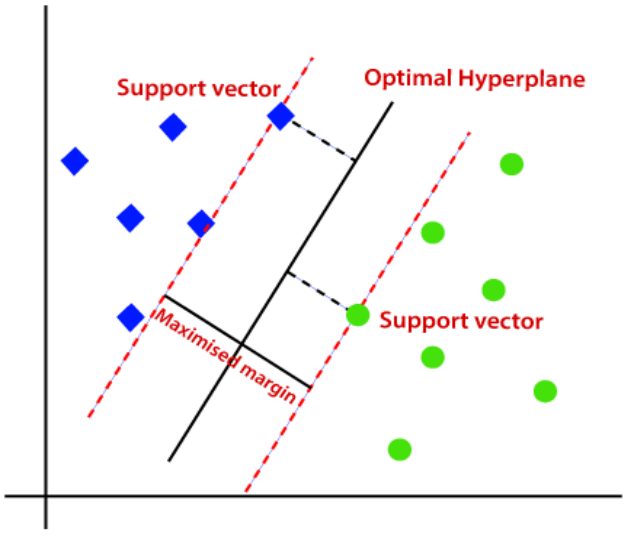

Figure 3.4: Support Vector Machine

3.2.5 XGBoost

XGBoost (Extreme Gradient Boosting) is an advanced ensemble learning algorithm that uses boosting to improve predictive accuracy. It is known for its speed and efficiency in handling large datasets.

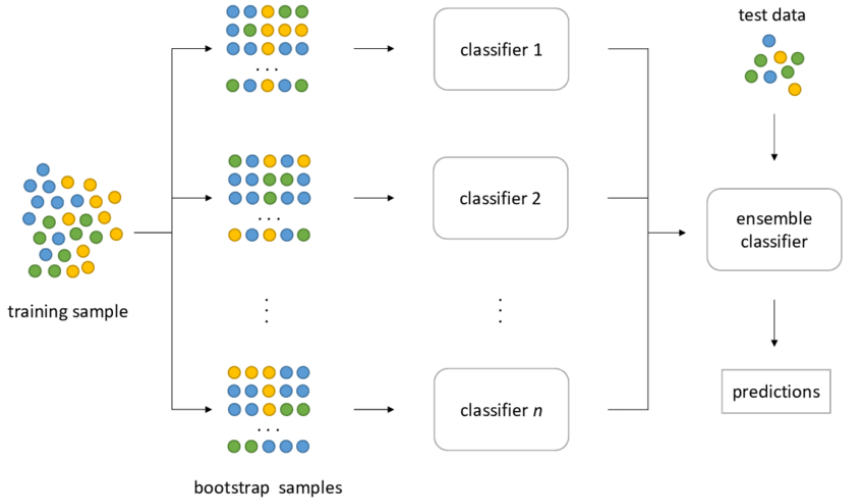

Figure 3.5: XGBoost

3.3 Performance Metrics

One of the crucial phases in creating a successful machine learning model is to evaluate the model's performance. Various measures are employed to evaluate the model's quality or performance; these metrics are referred to as evaluation metrics or performance metrics. We are able to better understand how successfully our model has worked with the provided data with to these performance metrics. By adjusting the hyper-parameters, we can enhance the model's performance in such a way. Each ML model aims to generalize well on unseen/new data, and performance metrics help determine how well the model generalizes on the new dataset.

There are various types of machine learning performance metrics, each providing an important angle on how a machine learning model is performing. In this study, we used the following machine learning performance metrics to evaluate our machine learning model.

- Classification Accuracy

- Precision
- Recall
- F1 score
- Confusion Matrix

3.3.1 Classification Accuracy

The simplest measure is classification accuracy. It is an essential indicator to evaluate a classification model's performance, giving a brief overview of the model's accuracy in making predictions. It is the proportion of instances in the data set that were accurately predicted to all instances. For balanced data sets, where each class is equally significant, accuracy is helpful.

$$\textbf{Accuracy} = \frac{\textit{Number of Correct Predictions}}{\textit{Total Number of Predictions}} \times 100\%$$

3.3.2 Precision

Precision focuses on the fraction of relevant instances among the retrieved instances. It's the ability of the classifier not to label a sample that is negative as positive. Precision is the ratio of true positives and total positives predicted:

$$\textbf{Precision} = \frac{\textit{True Positives}}{\textit{True Positives} + \textit{False Positives}}$$

3.3.3 Recall

In essence, a recall is the proportion of true positives to all ground truth positives. The classifier's recall measures its capacity to locate every important instance in a dataset. It is the classifier's capacity to identify every positive sample. When missing positive instances (false negatives) is more significant than having false positives, recall becomes crucial. It is the ratio of the true positive (TP) to the sum of true positive (TP) and false negative (FN).

$$\textbf{Recall} = \frac{\textit{True Positives}}{\textit{True Positives} + \textit{False Negatives}}$$

3.3.4 F1 score

F1 is a function of Precision and Recall. A statistic called the F1 Score is used to evaluate a binary classification model based on predictions made for the positive class. F1 score is a balanced metric that takes into account both false positives and false negatives. Precision and Recall are both represented by this type of single score. It is computed using precision and recall. It is calculated as the harmonic mean of precision and recall. F1 score is particularly helpful when working with data sets that are unbalanced. When there is an unequal distribution of classes or when the weights of false positives and false negatives are comparable, use the F1 score to strike a balance between precision and recall.

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

3.3.5Confusion Matrix

When true values are known, the classification model's performance on a set of test data is described using a confusion matrix, which is a tabular representation of the predicted outcomes of any binary classifier.

Although it can be used for classifiers with more than two classes, a typical confusion matrix for a binary classifier looks like the image below.

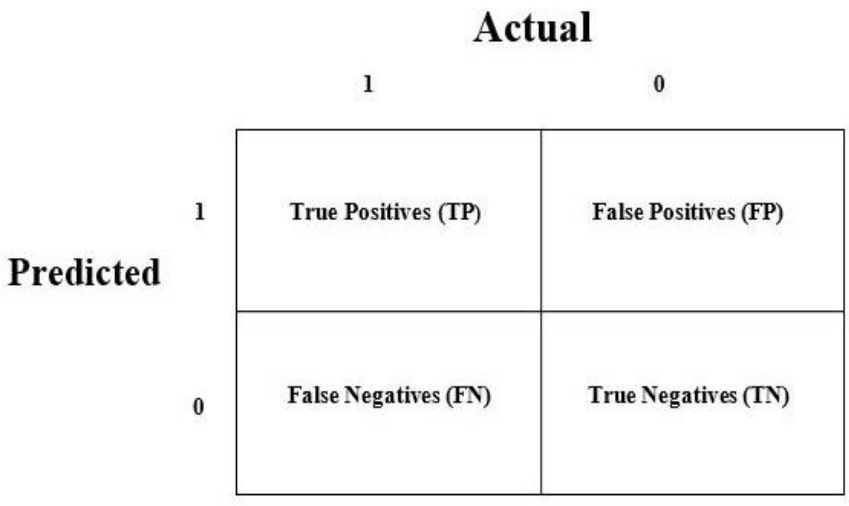

Figure 3.6: Confusion Matrix

3.4 Programming language and Environment

3.4.1 Python

Python has dynamic semantics and is a high-level, object-oriented, interpreted programming language. It is highly appealing for Rapid Application Development and for applications as a scripting or glue language for linking pre-existing components because of its high-level built-in data structures, dynamic typing, and dynamic binding. Python's easy-to-learn syntax prioritizes readability, which minimizes software maintenance costs. Python enables code reuse and software modularity by supporting modules and packages. For all major platforms, the Python interpreter and the large standard library are freely distributable and available in source or binary form. Python has a very quick edit-test-debug cycle and does not require a compilation phase. Python programs are straightforward to debug because a segmentation failure is never caused by a bug or incorrect input. Rather, the interpreter raises an exception when it finds a mistake. The interpreter prints a stack trace if the application fails to catch the exception. Setting breakpoints, evaluating arbitrary expressions, inspecting local and global variables, stepping through the code line by line, and more are all made possible by a source level debugger. The debugger's introspective nature is demonstrated by the fact that it was written in Python. However, adding a few print statements to the source code is frequently the fastest way to debug a program; this straightforward method works very well due to the short edit-test-debug cycle. [11]

3.4.2 Jupyter Notebook

Jupyter Notebook is a program for creating notebooks that falls under the Project Jupyter umbrella. Jupyter Notebook, which is based on the computational notebook format, provides quick and interactive ways to explore and visualize your data, prototype and explain your code, and share your ideas with others.

Notebooks offer a web-based application that can capture the entire computation process, including writing, documenting, and running code as well as sharing the outcomes, taking the console-based approach to interactive computing in a completely new direction.[12]

CHAPTER 4

METHODOLOGY

4.1 Introduction

The methodology for this diabetes prediction model involves a systematic approach to data handling, model training, and performance evaluation. The process integrates various techniques to ensure the data is properly prepared, analyzed, and utilized to build a robust prediction system. The architecture of the model is designed to efficiently process and predict diabetes outcomes based on the features in the dataset, ultimately aiming to provide accurate and reliable predictions. This section will provide an overview of the methodology and the components that drive the model's effectiveness.

4.2 Diabetes Prediction Model

The architecture diagram for the diabetes prediction model consists of five key modules. These modules ensure a structured approach to developing an accurate predictive system. They include:

i. Dataset Collection – Gathering relevant data for model training.
ii. Data Pre-processing – Cleaning, transforming, and preparing the data.
iii. Classifier – Implementing machine learning algorithms for prediction.
iv. Implementation – Deploying the trained model for real-world use.
v. Evaluation – Assessing the model’s performance using various metrics.

Each module plays a crucial role in enhancing the accuracy and reliability of diabetes prediction.

The following diagram illustrates the architecture of the diabetes prediction model, highlighting the five key modules and their interactions in the prediction process.

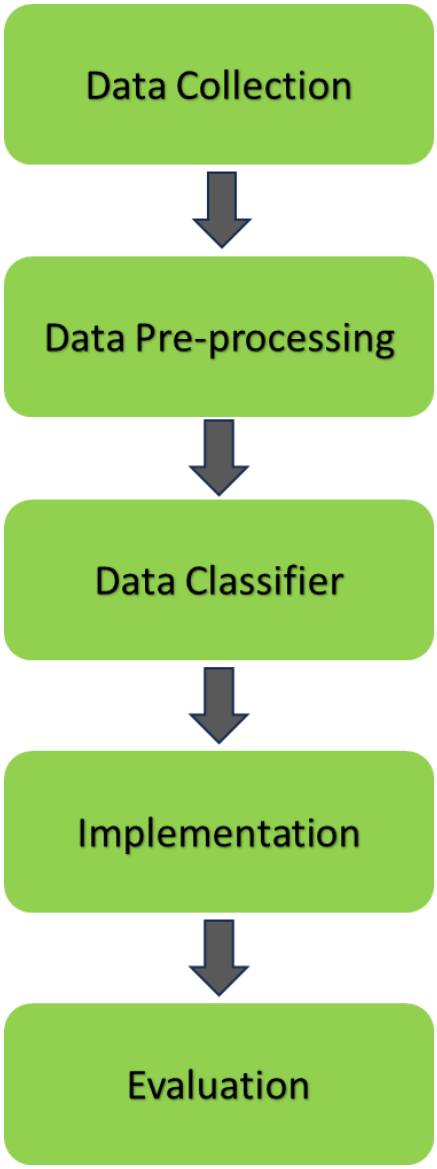

Figure 4.1: Diagram for Diabetes Prediction Model

i. Dataset Collection

The first step in the prediction model involves gathering relevant data to train and evaluate machine learning models. In our study, we used two sources of data:

1. **Publicly Available Kaggle Dataset**

- Contains 100,000 patient records with multiple health-related attributes.
- Features include Age, Gender, Body Mass Index (BMI), Hypertension, Heart Disease, Smoking History, HbA1c Level, and Blood Glucose Level.
- The dataset was selected due to its comprehensive nature and diverse patient information.

2. **Real-World Data from Cumilla Diabetes Hospital and Google form**
    - A supplementary dataset of 238 patients was collected from a local hospital in Cumilla and dataset of 104 general people using google form. We merged these two datasets. This dataset was included to enhance diversity and credibility by incorporating real-world cases.
    - By merging this with the Kaggle dataset, we aimed to create a more generalized machine-learning model.
    - After collecting the datasets, we concatenated them to create a comprehensive dataset for improved model performance.

ii. Data Pre-processing

Before training the machine learning models, it was necessary to clean and transform the data to ensure consistency, accuracy, and quality. The following pre-processing techniques were applied:

1. **Handling Missing Values**
    - Missing values were identified and addressed using appropriate techniques such as mean/median imputation or removal (if necessary).
    - This step prevents biased predictions and ensures the model learns from complete data.

2. **Removing Duplicates**
    - Duplicate records were checked and removed to avoid redundancy and prevent the model from being biased toward repeated data points.

3. **Outlier Detection and Handling**
    - Statistical methods such as the Interquartile Range (IQR) method and visualization techniques (boxplots) were used to detect and handle outliers.
    - Handling outliers ensures the model is not influenced by extreme values that can skew predictions.

4. **Feature Encoding (One-Hot Encoding for Categorical Variables)**
    - Certain categorical variables (e.g., Gender, Smoking History) needed to be converted into numerical format.
    - One-Hot Encoding was applied to transform categorical data into a format suitable for machine learning algorithms.

5. **Feature Scaling (Normalization and Standardization)**
    - Since different features have varying ranges (e.g., Age vs. Blood Glucose Level), Min-Max Scaling or Standardization was applied.
    - This step helps models like Logistic Regression and SVM perform better by ensuring features are on a similar scale.
    - By performing these pre-processing steps, we ensured a clean, structured, and well-prepared dataset for training the machine-learning models.

iii. Classifier (Model Training)

Once the dataset was cleaned and preprocessed, the next step was model selection and training. We used five different machine learning algorithms to determine which model performed best for diabetes prediction:

- Logistic Regression
- Decision Tree
- Random Forest
- Support Vector Machine (SVM)
- XGBoost

iv. Implementation

After selecting the machine learning models, we proceeded with the training, validation, and testing process:

1. **Splitting the Dataset**
   - The dataset was split into training (80%) and testing (20%) sets.
   - This helped evaluate the generalization capability of each model.

2. **Applying Hyperparameter Tuning (for Random Forest Model in Notebook 3)**
   - Grid Search was used to find the optimal values for key parameters:
     - n_estimators (Number of trees in the forest)
     - max_depth (Depth of each tree)
     - min_samples_split (Minimum samples needed to split a node)
     - min_samples_leaf (Minimum samples at a leaf node)
   - Fine-tuning these parameters improved accuracy and reduced overfitting.

3. **Training the Models**
   - Each model was trained using the preprocessed dataset.
   - Cross-validation was used to prevent overfitting and ensure consistency across different subsets of data.

4. **Handling Class Imbalance**

   Class imbalance is a common issue in medical datasets, where the number of diabetic cases is significantly lower than non-diabetic cases. To ensure fair learning and avoid model bias, a combination of Synthetic Minority Over-sampling Technique (SMOTE) and Random Under-Sampling (RUS) was applied:

   - **SMOTE (Synthetic Minority Over-sampling Technique)**

- SMOTE was used to generate synthetic samples for the diabetic class, increasing its size to **10% of the majority class**.
- It works by selecting a minority instance, finding its k-nearest neighbors, and generating new synthetic samples through interpolation.
- This prevents overfitting caused by simple duplication of existing samples.

- **Random Under-Sampling (RUS)**
  - After oversampling the minority class, RUS was applied to reduce the non-diabetic cases to **twice the number of diabetic cases**.
  - This ensured a more balanced dataset while retaining important information from the majority class.
- **SMOTE + RUS Hybrid Approach**
  - The combination of SMOTE and RUS maintained a balanced dataset while avoiding excessive overfitting or information loss.
  - This improved model performance by allowing it to learn patterns from both classes more effectively.

v. Evaluation

To assess the effectiveness of the models, we used multiple evaluation metrics:

1. Accuracy
2. Precision
3. Recall (Sensitivity)
4. F1-Score

CHAPTER 5

RESULT AND DISCUSSION

5.1 Introduction

This chapter presents the results of the diabetes prediction models, analyzing their performance based on multiple evaluation metrics. A comparison of different machine learning algorithms is provided, highlighting the impact of dataset size, preprocessing techniques, and hyperparameter tuning.

5.2 Visualization feature relations

To better understand the dataset and its characteristics, we applied various visualizations before training the machine learning model. These figures provide insights into feature distributions, relationships between variables, and their impact on diabetes. The following graphs illustrate key patterns and trends observed in the dataset, helping to inform feature selection and preprocessing decisions.

5.2.1 Feature distribution

Understanding the distribution of key features is essential in identifying patterns and potential biases in the dataset. The following visualizations depict the distribution of age, gender, BMI, and other categorical variables such as hypertension, heart disease, and smoking history. These insights help in assessing feature variability and their relevance to diabetes prediction.

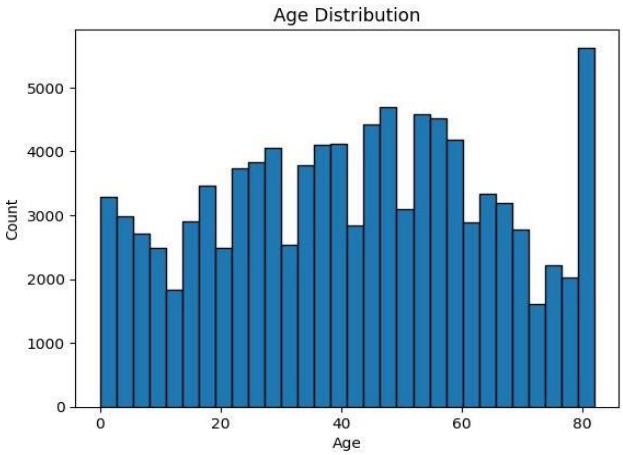

Figure 5.1: Bar Chat of Age Distribution

Age Distribution in the Total Dataset

Figure 4.1 shows the age distribution bar graph that visually represents the frequency of different age groups in the dataset, helping to identify patterns in age demographics and diabetes prevalence. The x-axis displays age ranges, while the y-axis shows the number of individuals in each category. The highest number of participants is observed approximately at age 80, with a count of 5,625, while the lowest count is approximately at age 72, with 1,615 participants. The height of each bar indicates the distribution of individuals across different ages, allowing for easy comparison

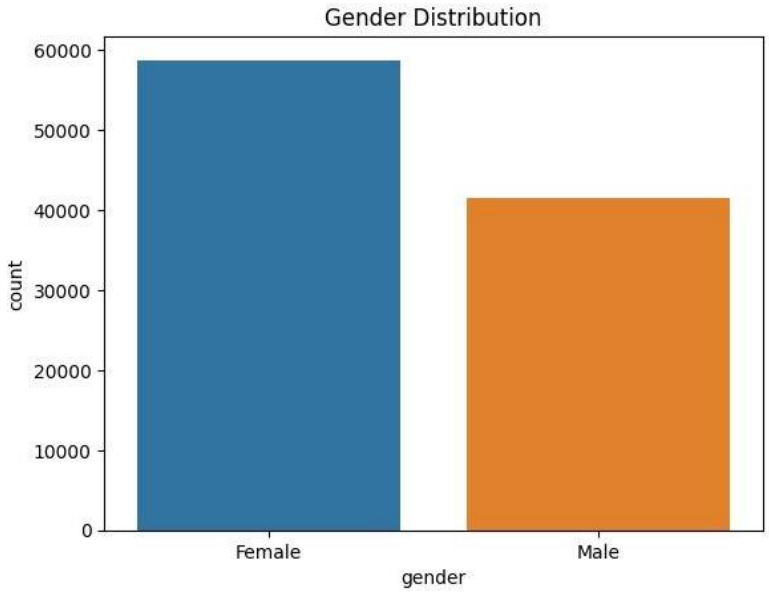

Figure 5.2: Bar Chart of Gender Distribution

Gender Distribution in the Total Dataset

Figure 4.2 shows the gender distribution bar graph that illustrates the proportion of male and female individuals in the dataset. The x-axis represents the gender categories, while the y-axis indicates the number of individuals in each group. The dataset consists of 58,744 female participants and 41,579 male participants, with the female count being noticeably higher. This visual representation allows for an easy comparison of gender distribution, providing insights into the overall demographic composition of the dataset.

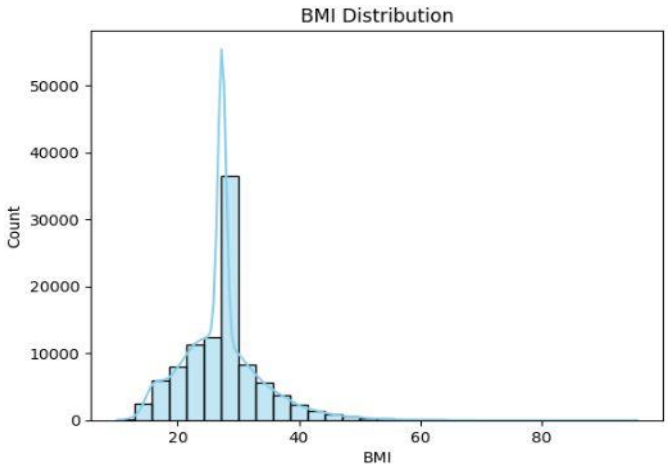

Figure 5.3: Bar chart of BMI Distribution

BMI Distribution in the Total Dataset

Figure 4.3 shows the BMI distribution graph that represents the spread of Body Mass Index (BMI) values in the dataset. The x-axis displays different BMI ranges, while the y-axis shows the number of individuals within each range. This visualization helps identify trends in body weight categories, such as underweight, normal weight, overweight, and obesity. A peak in BMI range 22-25 indicates the most common body compositions in the dataset.

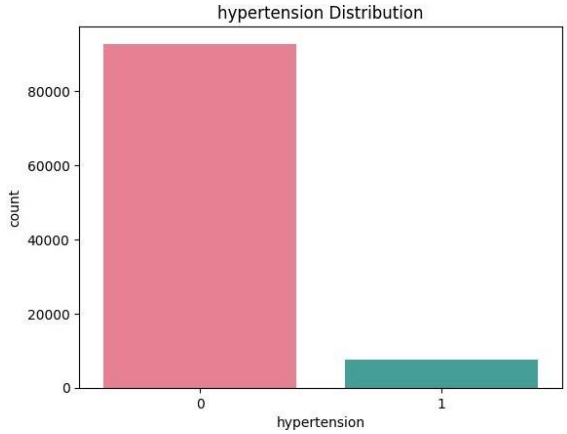

Figure 5.4: Bar Chart of Hypertension Distribution

Hypertension vs. Non-Hypertension Distribution in the Total Dataset

The hypertension distribution bar graph compares the number of individuals with and without hypertension. The x-axis represents two categories: Hypertension (1) and No Hypertension (0), while the y-axis shows the count of individuals in each group. Since the bar for the "No Hypertension" category is taller, it indicates that most individuals in the dataset do not have hypertension.

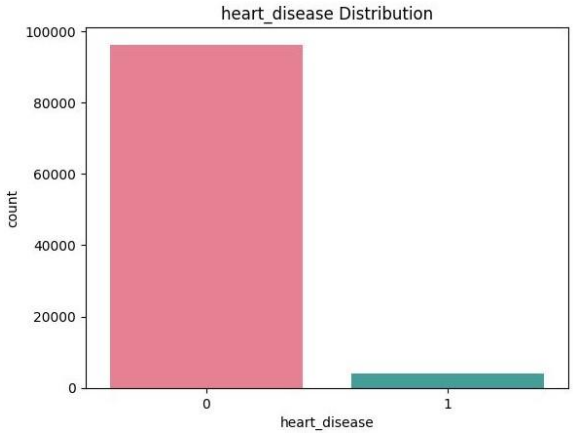

Figure 5.5: Bar Chart of Heart Disease Distribution

Heart Disease vs. Non-Heart Disease Distribution in the Total Dataset

The heart disease distribution graph displays the number of individuals with and without heart disease. The x-axis consists of two categories: Heart Disease (1) and No Heart Disease (0), and

the y-axis represents the number of individuals in each category. Here, a significantly taller bar for "No Heart Disease" indicates that most individuals do not have heart disease. This graph provides a quick overview of the dataset's heart disease distribution, making it easier to analyze its prevalence.

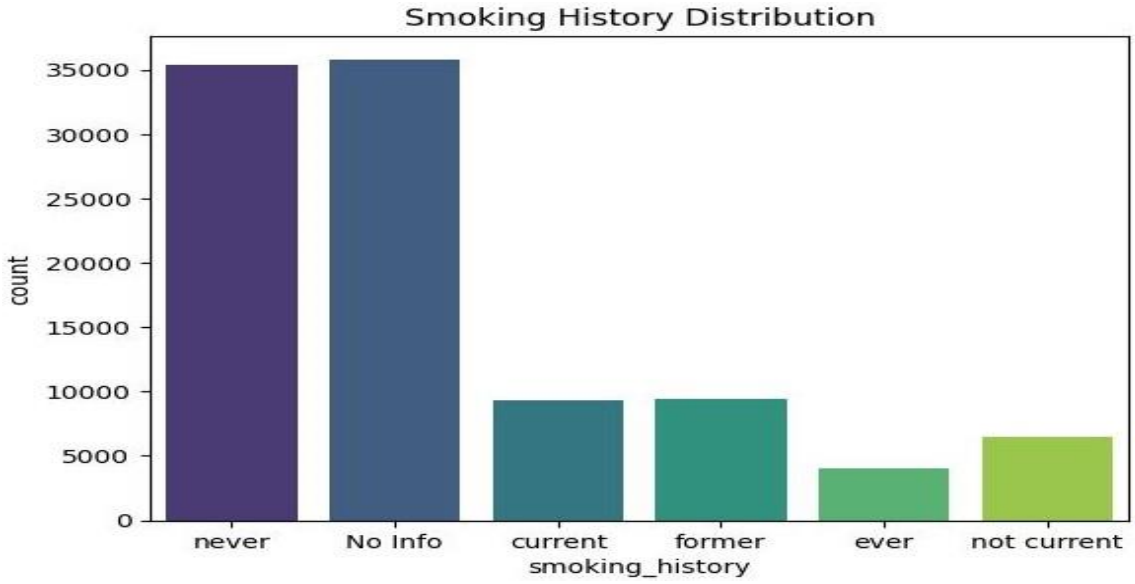

Figure 5.6: Bar Chart of Smoking History Distribution

Smoking History Distribution in the Total Dataset

The smoking history distribution graph illustrates the frequency of individuals based on their smoking habits. The x-axis represents different smoking categories, such as Never, Former Smoker, Current Smoker, Occasional Smoker, and others, while the y-axis indicates the number of individuals in each category. The dataset shows that the 'Never Smoked' and 'No Info' categories have the highest counts, significantly surpassing all others. The 'Current Smoker' and 'Former Smoker' categories have almost the same count, while the 'Not-Current Smoker' category is slightly lower than these two but still higher than the 'Ever Smoked' category, which has the lowest count. This visualization provides a clear understanding of smoking behavior trends in the dataset, highlighting the dominance of non-smokers and individuals with missing smoking history data.

5.2.2 Feature-wise Analysis of Diabetes Trends

To analyze how different features relate to diabetes, we visualized their distributions based on diabetes status. These graphs illustrate the impact of factors such as BMI, age, gender, and blood glucose level on diabetes prevalence. Understanding these relationships helps in feature selection and model interpretation.

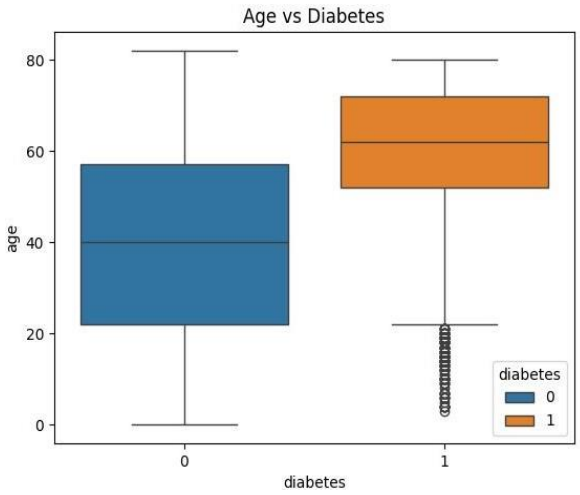

Figure 5.7: Box Plot of Age vs Diabetes Status

Age vs Diabetes Analysis:

A boxplot was used to analyze the relationship between age and diabetes classification. The visualization shows that diabetic individuals generally belong to an older age group compared to non-diabetic individuals. The median age for diabetic patients is a little noticeable, indicating that age is a significant factor in diabetes prevalence. The distribution of ages also shows a wider range for diabetic individuals, suggesting that diabetes affects a broad age spectrum. Additionally, extreme age values (outliers) are present, which may represent unique cases.

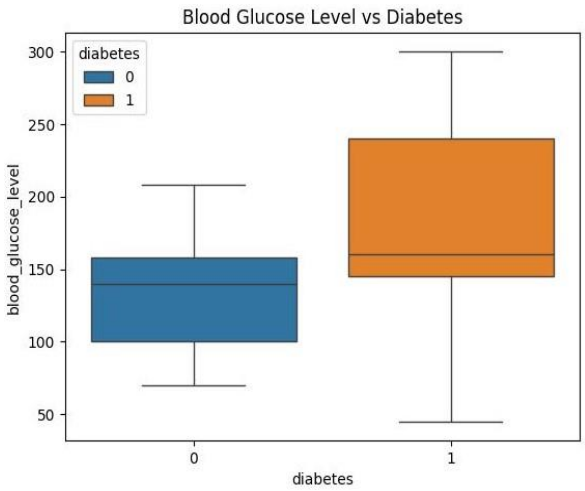

Figure 5.8: Box Plot of Blood Glucose vs Diabetes

Blood glucose vs Diabetes Analysis

The boxplot illustrates the distribution of blood glucose levels for diabetic and non-diabetic individuals. Non-diabetic patients typically have blood glucose levels ranging from approximately 100 to 170, while diabetic patients exhibit higher levels, ranging from around 150 to 230. This clear distinction in glucose levels between the two groups highlights its significance as a key feature for diabetes prediction. The visualization effectively captures the spread and central tendency of the data, aiding in better understanding and model development.

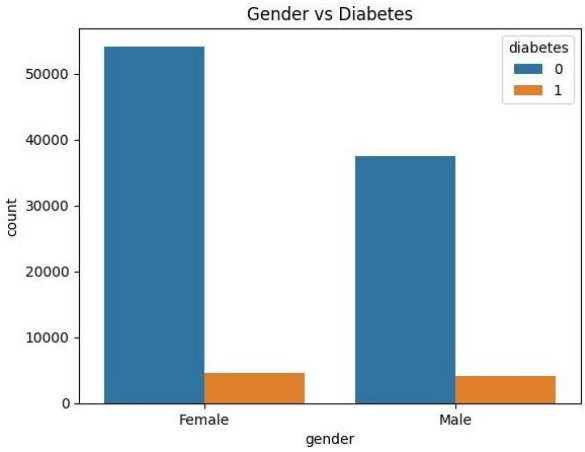

Figure 5.9: Bar Chart of Gender vs Diabetes

Gender vs Diabetes Analysis:

A count plot was generated to explore the relationship between gender and diabetes prevalence. The analysis reveals that diabetes is present in both genders, with 4,618 diabetic cases among females and 4,161 among males. Additionally, the dataset includes 54,126 non-diabetic females and 37,418 non-diabetic males, indicating a higher overall female participation. While the number of diabetic cases appears slightly higher in females, this may be influenced by the dataset distribution.

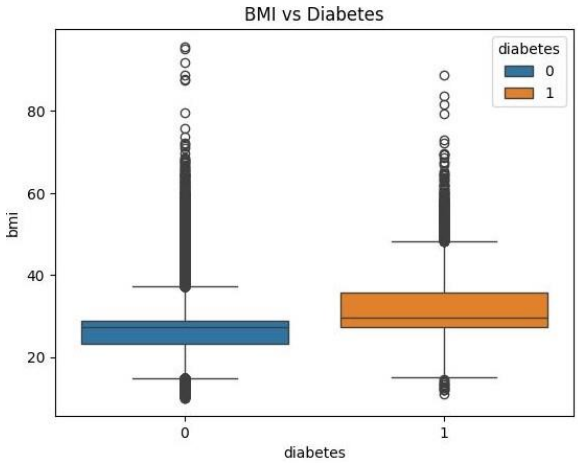

Figure 5.10: Box Plot of BMI vs Diabetes

BMI vs Diabetes Analysis:

A boxplot was used to analyze the relationship between Body Mass Index (BMI) and diabetes classification. The visualization indicates that individuals with diabetes tend to have a higher BMI compared to non-diabetic individuals. The median BMI for diabetic individuals appears higher, suggesting a potential correlation between BMI and diabetes. Additionally, the spread of BMI values (interquartile range) is wider for diabetic patients, with some outliers indicating extreme BMI values. This analysis highlights the importance of BMI as a predictive factor in diabetes classification.

5.2.3 Multivariate Feature Relationships

To visualize the relationships among numerical features and their correlation with diabetes, a pair plot was generated. This plot helps in identifying potential patterns, clustering tendencies, and dependencies between variables, providing insights into how different features contribute to diabetes classification.

Figure 5.11: Pair Plot of Feature Relationships with Diabetes

5.3 Model Performance Comparison

To evaluate model effectiveness, we trained and tested multiple machine learning algorithms on different datasets. The models were assessed using accuracy, precision, recall, and F1-score.

5.3.1 Results from Notebook 1 (BD Diabetes Dataset - 341 Entries)

The BD Diabetes dataset consists of 341 entries, which include various patient attributes related to diabetes risk factors. This dataset was used to train and evaluate different machine learning models. The results obtained from the models are as follows:

Model Comparison Results:

	Model	Accuracy	CV Mean	CV Std
0	Logistic Regression	0.826087	0.790438	0.038088
1	Decision Tree	0.811594	0.815892	0.053105
2	Random Forest	0.797101	0.819663	0.051840
3	SVM	0.797101	0.815960	0.027390
4	XGBoost	0.811594	0.834411	0.045601

Figure 5.12: Model Comparison Results for Diabetes Prediction from Notebook 1

The Logistic Regression model achieved the highest accuracy of 82.61%, followed closely by Decision Tree and XGBoost at 81.16%. Cross-validation results indicate that the models had consistent performance, with XGBoost showing the highest CV Mean (83.44%), suggesting stable generalization.

Although the dataset was expanded to 341 entries, it still remains relatively small, which may impact model performance and generalizability. The variations in **CV Std values** indicate slight fluctuations in model performance across different validation folds. The results highlight the effectiveness of these models in identifying diabetes patterns within the dataset while emphasizing the need for larger and more diverse data to improve prediction reliability. The confusion matrix of the test data is presented as following;

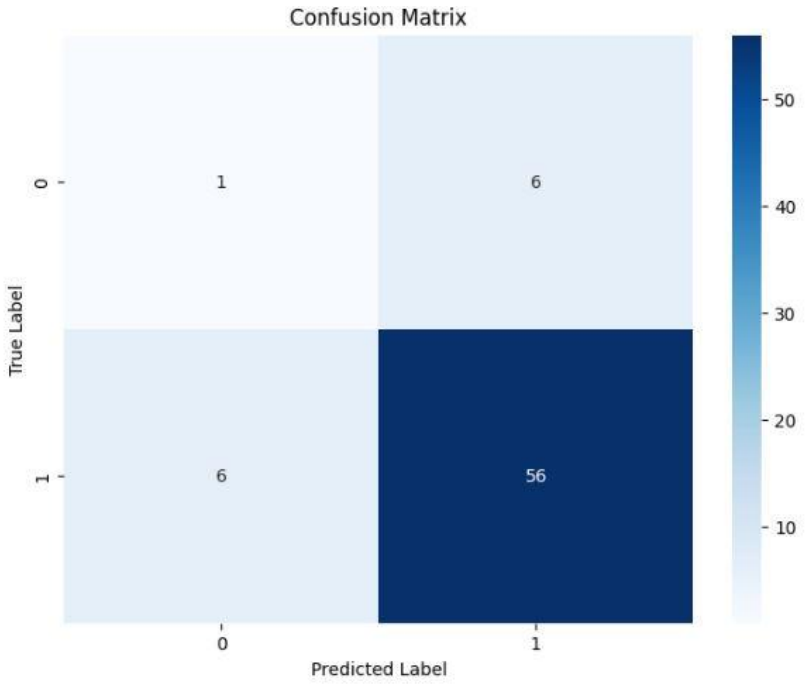

Figure 5.13: Confusion Matrix from notebook 1

The confusion matrix provides insights into the model's classification performance. The matrix shows that the model correctly predicted **1** true negative case (0,0) and **56** true positive cases (1,1), indicating successful identification of most diabetic patients. However, there are **6** false negatives (1,0), where diabetic patients were misclassified as non-diabetic, and **6** false positives (0,1), where non-diabetic individuals were incorrectly classified as diabetic. These misclassifications highlight areas for potential model improvement, such as adjusting decision thresholds or exploring feature engineering techniques.

5.3.2 Results from Notebook 2 (Kaggle dataset + BD Diabetes Dataset – 100,341 Entries)

In Notebook 3, we applied the same code as used in Notebook 1 and Notebook 2 to train machine learning models on a significantly larger dataset. This dataset was created by combining the 100,000-entry Kaggle dataset with 238 real-world patient records from Cumilla Diabetes Hospital, increasing the total number of records to 100,238. The results obtained from different models are as follows:

Model Comparison Results:

	Model	Accuracy	CV Mean	CV Std
0	Logistic Regression	0.959241	0.960123	0.000505
1	Decision Tree	0.951517	0.950967	0.001063
2	Random Forest	0.969455	0.969940	0.000471
3	SVM	0.962230	0.962664	0.000919
4	XGBoost	0.970851	0.970712	0.000670

Figure 5.14: Model Comparison Results for Diabetes Prediction from Notebook 2

Compared to Notebook 2 (BD Diabetes Dataset - 341 Entries), the performance of all models significantly improved after using a much larger dataset. For example:

- Logistic Regression accuracy increased from 82.61% to 95.92%, showing that a larger dataset helps improve model generalization.
- Decision Tree improved from 81.16% to 95.15%, indicating better learning from diverse data.
- SVM and XGBoost showed superior performance, achieving 96.22% and 97.09% accuracy, respectively.

The confusion matrix of the test data is presented as following;

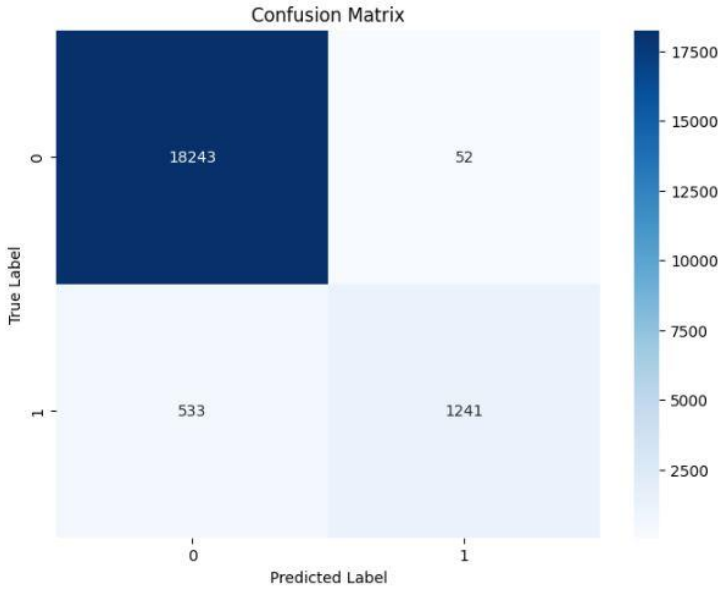

Figure5.15: Confusion Matrix from notebook 2

The confusion matrix demonstrates strong model performance with 18,243 true negatives (0,0) and 1,241 true positives (1,1), indicating accurate classification for most cases. However, there are 533 false negatives (1,0), where diabetic patients were misclassified as non-diabetic, and 52 false positives (0,1), where non-diabetic individuals were incorrectly predicted as diabetic. While the false positive rate is low, the number of false negatives suggests a need for further optimization, such as adjusting class weights or using more advanced techniques to improve sensitivity in detecting diabetic cases.

5.4 Comparison of Model Performance Across Datasets

In this study, two different datasets were used to train machine learning models for diabetes prediction. In Notebook 1, the dataset consisted of data collected from a diabetes hospital and additional samples gathered via Google Forms. The highest accuracy achieved was 82.6% (Logistic Regression), while other models, such as Decision Tree, Random Forest, SVM, and XGBoost, exhibited moderate performance, with accuracies ranging from 79.7% to 81.6%. The cross-validation (CV) scores also showed some variability, as indicated by relatively higher standard deviations.

In Notebook 2, an enhanced dataset was used by combining the previous dataset with a larger Kaggle dataset. This significantly improved model performance, with accuracies exceeding 95% across all models. The Random Forest (96.9%) and XGBoost (97.1%) models performed exceptionally well, indicating strong generalization. The reduction in CV standard deviation suggests increased dataset stability, likely due to a larger, more diverse sample size.

The primary reason for the improvement in accuracy is the increase in dataset size and diversity. Larger datasets help models learn better feature representations, reducing the risk of overfitting and improving generalization. In contrast, smaller datasets can introduce noise and bias, limiting the models' ability to accurately predict unseen data. Additionally, tree-based models like Random Forest and XGBoost performed the best due to their ability to handle complex relationships and interactions within the dataset. Logistic Regression performed slightly lower, as it assumes a linear

relationship between features and the target variable, which may not fully capture the complexity of diabetes prediction.

5.4.1 Comparison visualization of data from Kaggle dataset and collected dataset

To evaluate the impact of adding collected data, we compared the distributions and feature relationships between the original Kaggle dataset and the updated dataset, which includes additional collected data. These visualizations highlight any changes in data distribution and ensure consistency in feature representation for improved model performance.

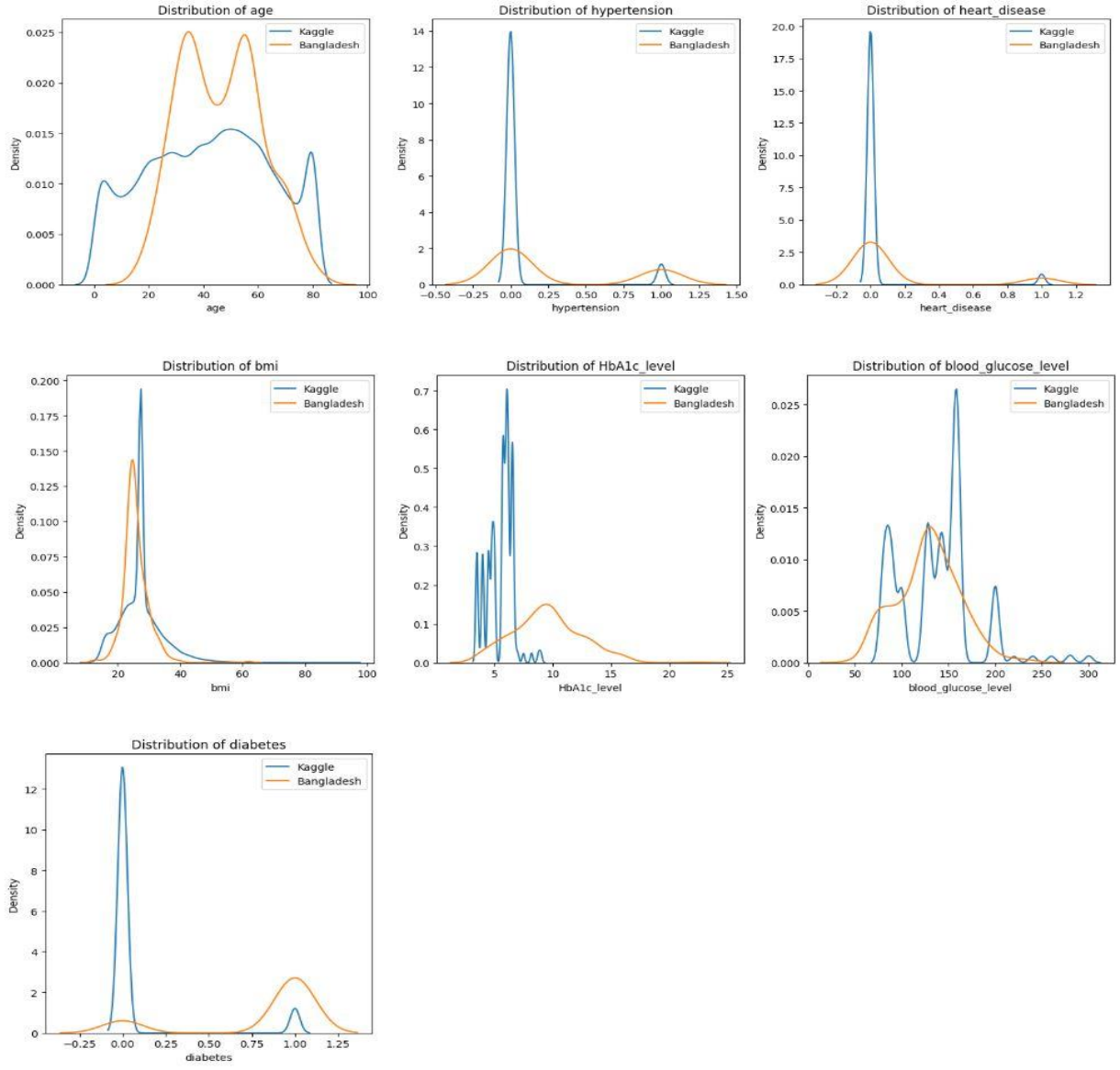

Figure 5.16: Comparison of Kaggle dataset and collected dataset

5.4.2 Analysis of Data Comparison

The visualizations provide insights into the differences and similarities between the Kaggle dataset and the collected dataset (hospital and Google Form data). Several key observations can be made:

Feature Distribution: The distributions of key features such as age, BMI, blood glucose levels, and HbA1c levels vary between the two datasets. The Kaggle dataset appears to have a more balanced distribution across different ranges, whereas the collected dataset may have skewed distributions due to a limited sample size or specific demographic biases.

Correlation Between Features: The relationship between variables such as hypertension, heart disease, and diabetes status may exhibit differences between the two datasets. The Kaggle dataset, being larger, might show clearer trends compared to the collected dataset.

Effect on Model Performance: The differences in feature distributions and class balances could explain the variation in model performance observed in previous sections. The combined dataset (Kaggle + collected) led to higher model accuracy due to increased diversity and improved generalization.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

In this study, we used a sizable dataset gathered from both public sources and actual patient records to create a machine learning-based diabetes prediction model. To assess how well several machine learning algorithms predicted diabetes, we used Logistic Regression, Decision Tree, Random Forest, Support Vector Machine (SVM), and XGBoost. Our results demonstrated that when trained on a bigger dataset, ensemble models like Random Forest and XGBoost obtained the maximum accuracy, with over 97% precision.

The experiment shows how predictive models may help with early diagnosis and risk assessment, highlighting the value of big data and machine learning in healthcare. The findings highlight how crucial high-quality, varied data is to enhancing model accuracy and dependability. Furthermore, our examination of feature relationships—such as age, blood glucose levels, and BMI—confirms their important significance in predicting the risk of diabetes.

6.2 Future scope

In order to increase forecast accuracy in the future, we want to include real-time data collecting from wearable technology and electronic medical records. We can advance to predictive analytics that can not only evaluate diabetes risk but also foresee the start of associated problems like cardiovascular disease by integrating real-time monitoring. We also intend to investigate multi-disease prediction algorithms that can conduct a comprehensive analysis of patient data, offering insights into additional possible health hazards associated with diabetes.

This study lays the groundwork for future investigations into AI-powered healthcare solutions, and more accurate and useful forecasts may be made as machine learning and data availability continue to progress. Using real-life big data exclusively from Bangladesh will improve the accuracy and clarity of predictions, making them more reliable for future research on region-specific prediction.

References:

[1] Aljumah, A.A., Ahamad, M.G. and Siddiqui, M.K., 2013. Application of data mining: Diabetes health care in young and old patients. Journal of King Saud University-Computer and Information Sciences, 25(2), pp.127-136.

[2]. Arora, R., 2012. Comparative analysis of classification algorithms on different datasets using WEKA. International Journal of Computer Applications, 54(13).

[3] Harikrishnan, J.P.M. and Vijayakumar, K., 2022. Performance Analysis of Neural Network Based Classifiers for the Prediction of Diabetes. MOLECULAR SCIENCES AND APPLICATIONS, 2, pp.24-28.

[4] Vijayan, V.V. and Anjali, C., 2015, December. Prediction and diagnosis of diabetes mellitus—A machine learning approach. In 2015 IEEE Recent Advances in Intelligent Computational Systems (RAICS) (pp. 122-127). IEEE.

[5] Sonar, P. and JayaMalini, K., 2019, March. Diabetes prediction using different machine learning approaches. In 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC) (pp. 367-371). IEEE.

[6] Sisodia, D. and Sisodia, D.S., 2018. Prediction of diabetes using classification algorithms. Procedia computer science, 132, pp.1578-1585.

[7] Pradhan, M. and Bamnote, G.R., 2015. Design of classifier for detection of diabetes mellitus using genetic programming. In Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014: Volume 1 (pp. 763-770). Springer International Publishing.

[8] Nai-Arun, N. and Moungmai, R., 2015. Comparison of classifiers for the risk of diabetes prediction. Procedia Computer Science, 69, pp.132-142.

[9] Hasan, M.K., Alam, M.A., Das, D., Hossain, E. and Hasan, M., 2020. Diabetes prediction using ensembling of different machine learning classifiers. IEEE Access, 8, pp.76516-76531.

[10] Febrian, M.E., Ferdinan, F.X., Sendani, G.P., Suryanigrum, K.M. and Yunanda, R., 2023. Diabetes prediction using supervised machine learning. Procedia Computer Science, 216, pp.21-30.

[11] Python Software Foundation, n.d. What is Python? Executive Summary. Python.org. Available at: https://www.python.org/doc/essays/blurb/ [Accessed 13 Feb. 2025].

[12] Project Jupyter, n.d. The Jupyter Notebook. Jupyter Notebook Documentation. Available at: https://jupyter-notebook.readthedocs.io/en/stable/notebook.html [Accessed 13 Feb. 2025].