

A PROJECT REPORT ON
DIABETES PREDICTION USING MACHINE LEARNING
TECHNIQUES



SUBMITTED IN PARTIAL FULFILLMENT FOR THE DEGREE
OF

MASTER OF COMPUTER SCIENCE

BY

Ms. KIRAN SAHU

(ROLL NO.-230708)

UNDER THE GUIDANCE OF

Dr. ANAMIKA SHUKLA SHARMA

HEAD

(DEPARTMENT OF COMPUTER SCIENCE)

GOVT. E. RAGHAVENDRA RAO P.G. SCIENCE COLLEGE BILASPUR
(CHHATTISGARH INDIA)

SESSION 2024-25

DEPARTMENT OF COMPUTER SCIENCE

GOVT. E. RAGHVENDRA RAO P.G. SCIENCE COLLEGE BILASPUR(C.G.)

SESSION 2024-25



A PROJECT REPORT ON

**DIABETES PREDICTION USING MACHINE LEARNING
TECHNIQUES**

BY

KIRAN SAHU

SUBMITTED IN PARTIAL FULFILLMENT FOR THE DEGREE

OF

MASTER OF COMPUTER SCIENCE

UNDER THE GUIDANCE OF

Dr. ANAMIKA SHUKLA SHARMA

HEAD

(DEPT. OF COMPUTER SCIENCE)

SUBMITTED TO

Dr. ANAMIKA SHUKLA SHARMA

HEAD

(DEPT. OF COMPUTER SCIENCE)

CERTIFICATE

This is to certify that the project entitled “ **DIABETES PREDICTION USING MACHINE LEARNING TECHNIQUES** ” is carried out by **Kiran Sahu (230719)** under my guidance and supervision for award of the degree Master of Computer Science, Department of Computer Science, Government E. Raghavendra Rao PG Science College, Bilaspur (CG).

To the best of my knowledge and belief the project: -

1. Embodies the work of the candidate herself.
2. Has duly been completed in the specified time.
3. Fulfils the requirements of the ordinance relating to M.Sc. degree of the college, and
4. Is up to the standard both in respect of content and language for being referred to the examiners.

Project Guide:

Dr. ANAMIKA SHUKLA SHARMA

Head (Department of Computer Science)

Head of the department

Dr. ANAMIKA SHUKLA SHARMA

DECLARATION

I declare that the project work entitled “ **DIABETES PREDICTION USING MACHINE LEARNING TECHNIQUES** ” has been carried by me under the supervision of internal guide **Dr. Anamika Shukla Sharma**, at the Department of Computer Science, Government E. Raghavendra Rao PG Science College Bilaspur (CG).

I further declare that this is an original work carried by me and to the best of my knowledge this project report does not contain any part of any work, that has been submitted for the award of any degree either in this University or in any other University / Deemed university without proper citation.

Kiran Sahu

M.sc IV Sem (Computer Science)

(230708)

**(Govt. E. Raghavendra Rao P.G. Science College
Bilaspur (C.G.))**

ACKNOWLEDGEMENT

This acknowledgement transcends the reality of formality when I would like to express deep gratitude and respect to all those people behind the scenes who guided, inspired and helped me for the completion of my project work. I consider myself fortunate enough to get such an outstanding project. This project would add as an asset to my academic profile, and I take this opportunity to express my sincere thanks to **Dr. Anamika Shukla Sharma**, Head (Department of Computer Science), Government E. Raghvendra Rao Post Graduate Science College, Bilaspur (C.G), for providing me a golden opportunity to work on this project and for her able guidance, valuable suggestion and discussion whenever required. I also express special thanks to **Dr. Kajal Kiran Gulhare**, Head (Department of Information Science), Government E. Raghvendra Rao Post Graduate Science College, Bilaspur (C.G), for her constant motivation and support. Finally, I express my indebtedness to **Mr. Harish Dewangan**, Assistant Professor (Department of Computer Science), Government E. Raghvendra Rao Post Graduate Science College, Bilaspur (C.G), and **Mr. Manishankar**, Assistant Professor (Department of Computer Science), Government E. Raghvendra Rao Post Graduate Science College, Bilaspur (C.G) for their valuable suggestions and co-operation without which it could not have been completed in time.

Kiran Sahu
M.sc IV Sem
(Computer Science)

ABSTRACT

Diabetes mellitus is a chronic metabolic disorder that has become a major global health concern, affecting millions of people across the world. Early detection and intervention are crucial in preventing severe complications associated with the disease. This thesis explores the application of machine learning techniques for the effective prediction of diabetes using patient health data. The study utilizes the Pima Indian Diabetes Dataset, which includes features such as glucose levels, insulin, BMI, and age, among others. Various supervised learning models, including decision trees, support vector machines, random forests, and gradient boosting classifiers, were implemented and evaluated to determine their accuracy, precision, and overall performance in predicting diabetes. Data preprocessing techniques such as feature scaling and imputation were employed to enhance model efficiency. Visual analytics were also used to identify key patterns and relationships within the dataset. The results demonstrate that machine learning can significantly aid in the early diagnosis of diabetes, offering potential benefits for both patients and healthcare providers. The thesis concludes with recommendations for model improvement and future research directions involving real-time data integration and personalized prediction systems.

Keywords: Diabetes Mellitus, Machine Learning, Prediction Models, Pima Indian Dataset, Data Preprocessing, Healthcare Analytics, Supervised Learning, Early Diagnosis

CONTENT

CHAPTER 1: INTRODUCTION [1-11]

1.1 Project Overview

1.1.1 Brief introduction to Diabetes Prediction

1.1.2 Importance and Applications

1.1.3 Objectives of the project

1.2 Problem Statement

1.2.1 The need for Diabetes Prediction

1.2.2 Challenges in Diabetes Prediction

1.3 Scope of the Project

1.3.1 What the project covers

1.3.2 What the project does not cover

1.4 Literature review

CHAPTER 2: METHODOLOGY [12-17]

2.1 Data Collection

2.1.1 Source of data

2.1.2 Data characteristics and description

2.2 Data Preprocessing

2.2.1 Handling missing values

2.2.2 Data cleaning

2.3 Model Building

2.4.1 Algorithms used (e.g., Logistic Regression, SVM)

2.4.2 Justification for the choice of algorithms

2.4.3 Front End Module Diagrams

2.4.4 Back End Module Diagrams

2.4.5 DATA FLOW DIAGRAM

CHAPTER 3: IMPLEMENTATION

[18-25]

3.1 Development Environment

3.1.1 Tools and software used (e.g., Python, Jupyter Notebook, Visual Studio Code)

3.1.2 System requirements

3.2 Visualizations and Charts

3.2.1 Histogram for each numerical feature

3.2.2 Box Plot by outcome

3.2.3 Age Distribution by Diabetes

3.2.4 Violin plot Visualization

3.2.5 Pairplot Visualization

3.3 Key function, Modules and User Interface

3.3.1 Key functions and modules used

3.3.2 User Interface

CHAPTER 4: EVALUATION

[26-30]

4.1 Model Evaluation Metrics

4.1.1 Metrics used (e.g., accuracy, precision, Confusion metrics)

4.1.2 Why these metrics are important

4.2 Results

4.2.1 Performance of the model on the training and test data

CHAPTER 5: DEPLOYMENT

[31-33]

5.1 Streamlit App

5.1.1 Introduction to Streamlit

5.1.2 Steps to create the Streamlit app

5.1.3 Complete Streamlit App Code

Chapter 6: Conclusion and Future Work

[34-36]

6.1 Conclusion

6.2 Future Work

6.3 References

CHAPTER-1

INTRODUCTION

1.1 Project Overview

1.1.1 Brief Introduction to Diabetes Prediction

Diabetes mellitus is a chronic and potentially life-threatening condition that arises when the body either does not produce enough insulin or is unable to effectively use the insulin it produces. This leads to elevated glucose levels in the blood, which, if left untreated, can cause serious health complications such as cardiovascular disease, kidney failure, vision impairment, and nerve damage. According to the World Health Organization (WHO), the prevalence of diabetes has been rising at an alarming rate globally, with millions of new cases diagnosed every year.

Traditional diagnostic methods for diabetes often rely on periodic medical testing, such as fasting blood sugar, oral glucose tolerance tests, and HbA1c levels. While effective, these methods may not always allow for timely intervention, especially in high-risk populations where regular medical check-ups are inaccessible or ignored.

With the advent of machine learning and data-driven approaches, predictive analytics has emerged as a powerful tool in the healthcare domain. Diabetes prediction models aim to identify individuals at high risk of developing the disease by analyzing historical health records and lifestyle-related data. These models can offer early warnings, enabling timely lifestyle changes or medical intervention, thus potentially preventing the onset of diabetes or minimizing its complications.

1.1.2 Importance and Applications

The importance of diabetes prediction lies in its potential to transform the landscape of healthcare from reactive treatment to proactive and preventive care. With the growing prevalence of diabetes worldwide—especially type 2 diabetes driven by poor lifestyle habits and genetic factors—early detection becomes vital. Machine learning-based prediction systems allow for timely identification of individuals at high risk, enabling early interventions that can delay or even prevent the onset of the disease. These systems are also scalable, cost-effective, and can reach underserved populations who may not have regular access to healthcare facilities.

Key applications of diabetes prediction models include:

1. **Preventive Healthcare and Lifestyle Modification:** Predictive tools can empower individuals with knowledge about their risk level, encouraging healthier behaviors such as dietary improvements, regular exercise, weight management, and routine blood sugar monitoring. These tools often serve as the first line of defense in combating diabetes, especially among at-risk populations.
2. **Clinical Decision Support Systems (CDSS):** When integrated into hospital or clinic information systems, these models assist healthcare professionals in diagnosing prediabetes or diabetes with greater speed and accuracy. They can complement traditional diagnostic tests by adding an intelligent layer that considers multiple risk factors simultaneously, helping in personalized treatment planning.
3. **Mobile Health (mHealth) and Telemedicine:** With the surge in smartphone usage and digital health applications, diabetes prediction models can be embedded in mobile apps that perform risk assessments in real time. These apps often include interactive dashboards, reminders for glucose monitoring, medication adherence tools, and even AI-driven virtual health coaches, making healthcare more accessible and patient-centric.
4. **Population Health Management and Public Policy:** At a macro level, health agencies and policymakers can use aggregated prediction model outcomes to identify geographical regions or demographic groups with higher diabetes risk. This information is critical for launching community-based awareness campaigns, planning screening drives, and ensuring the optimal distribution of medical resources such as test kits, clinics, and diabetic education programs.

5. **Insurance and Risk Assessment:** Health insurance providers can incorporate prediction models to assess policyholders' risk of chronic conditions like diabetes. This aids in designing more personalized and cost-effective insurance plans, while also promoting early preventive strategies that benefit both the insurer and the insured.
6. **Wearable and IoT Integration:** When combined with data from wearable health devices (e.g., fitness trackers, continuous glucose monitors, smartwatches), prediction models can provide continuous monitoring and alert systems. This fusion of AI and IoT can result in highly adaptive and responsive healthcare systems that support long-term diabetes management and early alerts for sudden glucose fluctuations.
7. **Healthcare Research and Innovation:** Academic and clinical researchers can use prediction models to uncover hidden correlations and trends within patient data. This can lead to new hypotheses about the causes and progression of diabetes, ultimately guiding the development of better treatment options and public health strategies.

Diabetes prediction models not only contribute to improving individual patient outcomes but also enhance the efficiency and effectiveness of the broader healthcare ecosystem. They represent a crucial step toward a smarter, data-driven approach to chronic disease management.

1.1.3 Objectives of the Project

The primary aim of this project is to design and evaluate machine learning models for the prediction of diabetes using historical patient data. The specific objectives are as follows:

1. **To explore and preprocess the diabetes dataset** by handling missing values, outliers, and scaling features to prepare the data for modeling.
2. **To implement and compare multiple machine learning algorithms**, including decision trees, random forests, support vector machines (SVM) for classifying individuals as diabetic or non-diabetic.
3. **To evaluate model performance** using standard classification metrics such as accuracy, precision, recall, F1-score, and ROC-AUC to determine the most effective algorithm.
4. **To visualize the data and model results** using appropriate charts and graphs for better interpretability and communication of findings.
5. **To identify key features** that contribute most to the prediction of diabetes, thereby offering clinical insights into potential risk factors.
6. **To provide recommendations** for improving model performance and suggest future directions for research, including integration with real-time health monitoring systems.
7. **Deploy the Model as a Web Application:** Develop a user-friendly web application using Streamlit to demonstrate the functionality of the diabetes prediction system. The application should allow users to input medical details and receive instant classification results.

This project thus contributes to the growing body of research in health informatics and demonstrates the real-world applicability of machine learning in predictive diagnostic

1.2 Problem Statement

Diabetes mellitus is a pervasive non-communicable disease (NCD) that presents a serious threat to global health. According to the International Diabetes Federation (IDF), the number of adults living with diabetes is projected to rise from 537 million in 2021 to 643 million by 2030 and 783 million by 2045. Despite being largely preventable and manageable through early interventions, many individuals remain undiagnosed or are diagnosed too late, when complications have already begun to manifest. This late diagnosis leads to increased morbidity, mortality, and healthcare expenditures.

The traditional approaches to diagnosing diabetes—such as fasting plasma glucose tests, oral glucose tolerance tests, and HbA1c measurements—although effective, are reactive in nature. These diagnostic tests are typically conducted only after symptoms appear or during periodic health checkups. This results in a significant gap in early detection, particularly in low-resource settings or among individuals who do not undergo regular screening.

In this context, predictive modeling using machine learning offers a powerful alternative by analyzing patterns in existing data to identify individuals who are at risk of developing diabetes, even before any physical symptoms arise. However, building accurate, robust, and generalizable prediction models presents its own set of challenges, ranging from data quality issues to algorithm selection and interpretability. This project aims to address these issues by developing and evaluating data-driven models that can predict diabetes efficiently and reliably.

1.2.1 The Need for Diabetes Prediction

The need for diabetes prediction arises from both medical and socioeconomic perspectives. From a medical standpoint, early identification of individuals at high risk enables timely lifestyle interventions and clinical treatments, which can prevent or delay the onset of the disease. This is especially critical in the case of **Type 2 diabetes**, which often develops gradually and can remain asymptomatic for years. Prediction models can play a vital role in identifying those individuals who are in the **prediabetic** stage and are thus more amenable to non-pharmacological interventions like exercise, diet, and behavioral changes.

From a socioeconomic viewpoint, diabetes places a massive burden on healthcare systems and economies. The cost of managing complications such as kidney failure, heart disease, stroke, and lower-limb amputations is significantly higher than the cost of preventive care. By using predictive analytics to screen at-risk populations and prioritize care, healthcare providers can optimize resource allocation and reduce overall costs.

Additionally, in underserved or rural regions where access to laboratory diagnostics is limited, digital tools powered by prediction models can serve as cost-effective, portable, and scalable solutions. These tools can function on mobile devices, making them particularly useful in community health programs or telemedicine environments.

In summary, diabetes prediction is not just a technical advancement but a public health necessity that can transform healthcare from reactive to preventive, minimize suffering, and ensure more equitable access to early diagnosis and care.

1.2.2 Challenges in Diabetes Prediction

While the benefits of diabetes prediction using machine learning are substantial, there are several technical, ethical, and operational challenges that must be addressed to ensure the development of effective models.

1. **Data Quality and Availability:** One of the biggest challenges in predictive modeling is obtaining high-quality, comprehensive, and well-labeled datasets. Medical data often suffers from missing values, inconsistencies, and noise. In the case of diabetes datasets, important lifestyle variables such as diet, stress, and physical activity are often missing or underrepresented. Furthermore, privacy concerns can restrict access to large-scale, diverse datasets needed for model training.

2. **Class Imbalance:** In many publicly available datasets, the number of individuals with diabetes is significantly lower than those without the disease. This class imbalance can bias models toward predicting the majority class, leading to high overall accuracy but poor performance in detecting actual diabetic cases (low recall). Techniques such as oversampling, undersampling, or synthetic data generation must be carefully applied to address this issue.

3. **Feature Selection and Relevance:** Selecting the right features that truly influence the onset of diabetes is crucial. Irrelevant or redundant features can mislead the learning algorithm, resulting in decreased model performance. In real-world datasets, features may also be correlated or exhibit non-linear relationships that are hard to capture with simple models.
4. **Model Interpretability:** Healthcare professionals often require explanations for why a model predicted a certain outcome. Black-box models such as deep neural networks or ensemble methods like random forests may offer high accuracy but lack interpretability, which can hinder their adoption in clinical settings. There is a growing demand for explainable AI (XAI) in medical applications to ensure trust and transparency.
5. **Generalization and Bias:** A model trained on data from one population may not generalize well to another due to demographic, genetic, or lifestyle differences. This can introduce bias and lead to inaccurate predictions, particularly in underrepresented populations. Building fair and generalizable models requires diverse training data and continuous model validation.
6. **Integration into Clinical Workflows:** Even the most accurate models may fail to deliver impact if they are not integrated effectively into clinical workflows. Factors such as user interface design, real-time performance, and compatibility with existing Electronic Health Record (EHR) systems can influence the utility of prediction tools in practice.
7. **Ethical and Legal Considerations:** Using patient data for prediction raises ethical questions around data privacy, informed consent, and algorithmic accountability. Compliance with regulations such as GDPR (General Data Protection Regulation) or HIPAA (Health Insurance Portability and Accountability Act) is essential when designing healthcare models.

1.3 Scope of the Project

1.3.1 What the Project Covers

1. Data Collection and Preprocessing:

- o **Data Source:** The project utilizes the spam.csv dataset, which contains labeled SMS messages categorized as either spam or ham (non-spam).
- o **Data Cleaning:** This includes handling missing values, removing irrelevant characters, and standardizing the format to ensure consistency.

2. Feature Extraction:

- o **Feature Selection:** Identifying and selecting the most relevant features that contribute to distinguishing diabetes patients.

3. Model Building:

- o **Algorithm Selection:** The project involves experimenting with different machine learning algorithms, such as Logistic Regression, Random Forest and Support Vector Machines (SVM), to find the most effective model for diabetes prediction.
- o **Training the Model:** The selected algorithm is trained on the pre-processed dataset to learn the patterns associated with diabetes patient data.

4. Model Evaluation:

- o **Evaluation Metrics:** The performance of the model is assessed using metrics such as accuracy, precision, recall, and F1-score to ensure its reliability and effectiveness.
- o **Validation:** Cross-validation techniques are applied to validate the model and prevent overfitting.

5. **Deployment:**

- o **Streamlit Web Application:** A user-friendly web application is developed using Streamlit, allowing users to input patient details and receive instant classification results (predicting diabetes).
- o **Loading Pre-trained Model:** The trained model saved as model.pkl files, which are loaded in the Streamlit app for real-time predictions.

6. **Documentation:**

- o Comprehensive documentation is created, detailing each step of the project, including data collection, preprocessing, feature extraction, model building, evaluation, and deployment.

1.3.2 What the Project Does Not Cover

1. **Real-time Data Collection:** The project does not involve the real-time collection of patients' detailed data from mobile devices or live streams. It relies solely on the diabetes.csv dataset for training and evaluation purposes.
2. **Integration with Mobile Applications:** The scope of the project does not extend to integrating the spam detection system into mobile applications or hospital institute. The focus is on developing a standalone web application.
3. **Real-time System Scalability:** The focus is on developing a proof-of-concept system and does not address scalability issues for handling large-scale real-time diabetes prediction in a production environment.
4. **Automated Model Updates:** The project does not cover the implementation of automated systems for continuously updating and retraining the model with new data to adapt to evolving spam tactics.
5. **User Authentication and Security:** The web application developed using Streamlit does not include features for user authentication, data encryption, or other security measures typically required for a production-grade application.

1.4 Literature Review

1. "Diabetes Disease Diagnosis Using Artificial Neural Networks" – Patil et al. (2010): This study applied Artificial Neural Networks (ANN) on the Pima Indian Diabetes Dataset (PIDD) to build a predictive model. The model demonstrated high performance in classifying diabetic vs. non-diabetic patients. The researchers emphasized the ANN's ability to model complex non-linear relationships. The paper concludes that neural networks are highly suitable for early diagnosis tasks, although it lacks interpretability and requires substantial data preprocessing.

2. "Early Detection of Diabetes Using Machine Learning Tools" – Sisodia & Sisodia (2018): The authors implemented and compared four classifiers—Decision Tree, Naive Bayes, Support Vector Machine, and K-Nearest Neighbors—on the PIDD. The study found that the Decision Tree classifier gave the best accuracy (~76%). This paper highlights the importance of algorithm selection and provides a baseline for traditional machine learning techniques in diabetes prediction.

3. "Machine Learning Techniques for Diabetes Prediction Using the Pima Indian Dataset" – Jayanthi et al. (2017): This research explores multiple ML algorithms (Logistic Regression, Random Forest, SVM, and ANN) to predict diabetes. Random Forest achieved the highest accuracy among all. The study demonstrates that ensemble methods often perform better due to their ability to reduce variance and overfitting. It also discusses the role of hyperparameter tuning in improving model efficiency.

4. "Diabetes Prediction Using Machine Learning Algorithms: A Comparative Study" – Kumar & Vohra (2020): This comparative study evaluated Logistic Regression, Decision Trees, Random Forest, and Gradient Boosting. Gradient Boosting outperformed other models with an accuracy above 80%. The study also explored the impact of feature selection and normalization. It suggests that advanced ensemble techniques can provide more reliable predictions when fine-tuned correctly.

5. "A Review on Diabetes Prediction Using Machine Learning Techniques" – Kavakiotis et al. (2017): This paper is a comprehensive survey of machine learning applications in diabetes care, including prediction, diagnosis, and monitoring. It categorizes the techniques into supervised and unsupervised learning and provides critical insights into their strengths and limitations. It also highlights the increasing

role of deep learning and the importance of data preprocessing in healthcare datasets.

6. "Deep Learning for Diabetes Prediction Using Big Data" – Zhang et al. (2019):

Zhang et al. utilized deep learning techniques on large-scale Electronic Health Records (EHR) to predict Type 2 diabetes. The paper employs deep neural networks (DNNs) and outperforms traditional ML models in accuracy and precision. This study demonstrates how big data combined with deep learning can lead to better predictive performance, though at the cost of increased complexity and reduced explainability.

7. "Prediction of Diabetes Using Random Forest Algorithm" – Deo (2015):

This paper focuses exclusively on the Random Forest algorithm applied to the Pima dataset. The Random Forest classifier achieved an accuracy of around 77%. It highlights the importance of ensemble learning and variable importance in understanding the key features affecting diabetes.

8. "A Novel Approach for Diabetes Prediction Using Data Mining Techniques" –

Nayak et al. (2013): Nayak et al. proposed a hybrid model using Decision Trees and Genetic Algorithms. The hybrid approach outperformed individual models and also helped in identifying the most critical features. This paper is notable for introducing optimization techniques in the prediction process, adding a layer of intelligence to standard ML workflows.

9. "Smart Health Monitoring System for Predicting Diabetes Using Machine

Learning Algorithms" – Reddy et al. (2021): This study integrates ML models into a health monitoring system. The system employs algorithms like Random Forest and SVM and is designed for mobile health applications. The paper demonstrates how ML can be integrated into IoT environments for proactive healthcare delivery.

10. "Explainable AI for Medical Diagnosis: A Case Study on Diabetes

Prediction" – Liu et al. (2022): This paper focuses on the interpretability of models. Using SHAP with XGBoost, it not only predicts diabetes risk but also explains feature importance for each prediction. The work underscores the rising importance of Explainable AI (XAI) in gaining trust from healthcare professionals.

CHAPTER-2

METHODOLOGY

2.1 Data Collection

2.1.1 Source of Data The dataset for this project was sourced from Kaggle. The dataset, titled 'diabetes.csv', is a collection of patient records labelled as 1 or 0 for diabetic or not.

Dataset Link : <https://www.kaggle.com/datasets/mathchi/diabetes-data-set>

2.1.2 Data Characteristics and Description

- **Number of Instances:** The dataset comprises 768 patient records.
- **Features:** The primary features are the record of patients:
 1. *Pregnancies*: Number of times a patient has been pregnant.
 2. *Glucose*: Plasma glucose concentration of 2 hours in an oral glucose tolerance test.
 3. *BloodPressure*: Value of Diastolic blood pressure (mm Hg).
 4. *SkinThickness*: Triceps skin fold thickness (mm).
 5. *Insulin*: 2-Hour serum insulin (mu U/ml).
 6. *BMI*: Body mass index (weight in kg/(height in m)²)
 7. *DiabetesPedigreeFunction*: Diabetes pedigree function.
 8. *Age*: Age of a patient in years
 9. *Outcome*: Class variable (0 or 1). Whether a patient is diabetic or not.
- **Labels:** Each message is labeled as either 1 or 0, indicating if a patient is diabetic or not.

2.2 Data Preprocessing

2.2.1 Handling Missing Values

- The dataset was inspected for any missing values. Rows with missing values were removed to maintain data quality and integrity.

2.2.2 Data Cleaning

- **Duplicate Removal:** Any duplicate messages were identified and removed to avoid redundant data affecting the model's performance.
- **Lowercasing:** All text data was converted to lowercase to ensure uniformity.
- **Dealing with outliers:** Finding and removing outliers for better performance of machine learning models.

2.3 Model Building

2.3.1 Algorithm used: In developing the SMS spam classifier, various machine learning algorithms were implemented and evaluated to determine the most effective model. The following table summarizes the classifiers used, their parameters, and their performance metrics:

2.3.2 Table of used Algorithm:

Model	Accuracy	Recall	Precision	F1 Score	Description
Logistic Regression	87.01%	87.01%	86.97%	86.72%	A simple yet powerful linear model suitable for binary classification problems.
Decision Tree	75.32%	75.32%	76.85%	75.77%	A tree-based model that is easy to interpret but prone to overfitting.
Random Forest	80.52%	80.52%	80.36%	80.42%	An ensemble of decision trees that improves accuracy and reduces overfitting.
Support Vector Machine	83.12%	83.12%	82.85%	82.84%	A robust model for high-dimensional spaces; performs well with margin separation.
K-Nearest Neighbour	77.92%	77.92%	77.46%	77.56%	A distance-based algorithm that is simple but sensitive to data scaling.

2.3.2 Justification for the Choice of Algorithms

The selection of algorithms for the Diabetes prediction classifier was guided by the need to balance performance, interpretability, and computational efficiency. Each algorithm was chosen based on its unique strengths and suitability for the characteristics of the diabetes prediction classification problem.

Logistic Regression

Logistic Regression is a popular and effective model for binary classification problems like diabetes prediction, where the outcome is either diabetic or non-diabetic. Its strength lies in its simplicity, interpretability, and solid statistical foundation. The model assumes a linear relationship between the input features and the log-odds of the outcome, which works well when the data is well-behaved. In healthcare, explainability is critical, and Logistic Regression offers coefficients that can help interpret the influence of each predictor, such as glucose level or BMI, on the outcome. This makes it suitable for clinical settings where transparency is valued as much as predictive power.

Decision Tree

Decision Trees are intuitive and mimic human decision-making by splitting data based on feature thresholds. They are especially useful in medical domains because their logic can be easily visualized and understood by healthcare professionals. For diabetes prediction, a decision tree can clearly show which combinations of risk factors lead to a positive diagnosis. However, their tendency to overfit on training data limits their generalization, which can explain the relatively lower accuracy in some cases. Despite this, their interpretability and ease of implementation make them a valuable tool for initial modeling and feature understanding.

Random Forest

Random Forest improves upon Decision Trees by building multiple trees and averaging their predictions, which significantly reduces the risk of overfitting and enhances model robustness. For diabetes prediction, it helps capture complex interactions between features such as insulin levels, BMI, and age. Moreover, Random

Forest provides internal estimates of feature importance, allowing researchers to identify which factors contribute most to predicting diabetes. Its balance between performance and interpretability makes it a strong candidate for both research and clinical decision support systems.

Support Vector Machine (SVM)

Support Vector Machines are effective in high-dimensional spaces and are particularly useful when classes are not linearly separable. In diabetes prediction, where feature boundaries can be subtle and overlapping, SVMs use kernel functions to map features into a higher-dimensional space for better separation. This allows the model to form an optimal hyperplane that maximizes the margin between diabetic and non-diabetic classes. The result is a strong generalization performance, especially when data is properly scaled. Though SVMs are less interpretable, their high accuracy makes them well-suited for precision-oriented diagnostic tools.

K-Nearest Neighbour (KNN)

KNN is a non-parametric, instance-based learning algorithm that classifies new data points based on the majority class of their nearest neighbors. It performs well in scenarios where the data distribution is non-linear and no prior assumptions are made about the underlying structure. In the context of diabetes prediction, KNN can be effective if the dataset is normalized and clean. However, its performance can degrade with high-dimensional data and large datasets due to increased computational cost. Despite these limitations, KNN is simple to implement and can serve as a useful benchmark or secondary model.

2.3.3 Front End Module Diagrams:

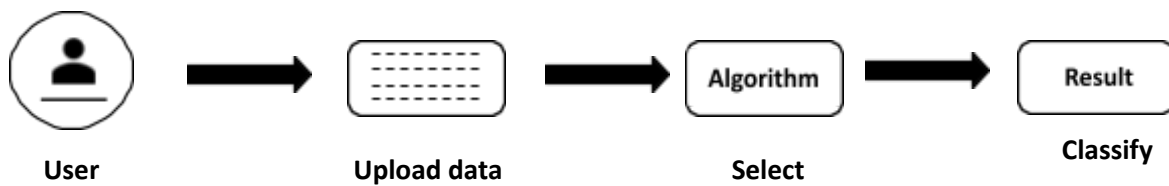


Fig.1 Front End Module

2.3.4 Back End Module Diagrams:

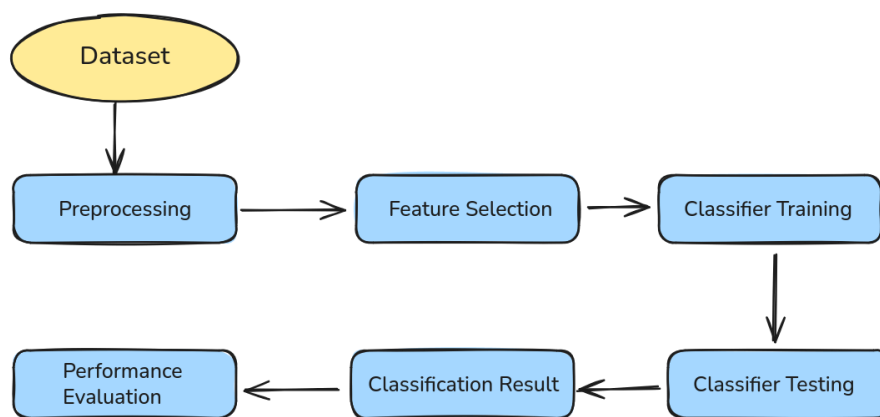
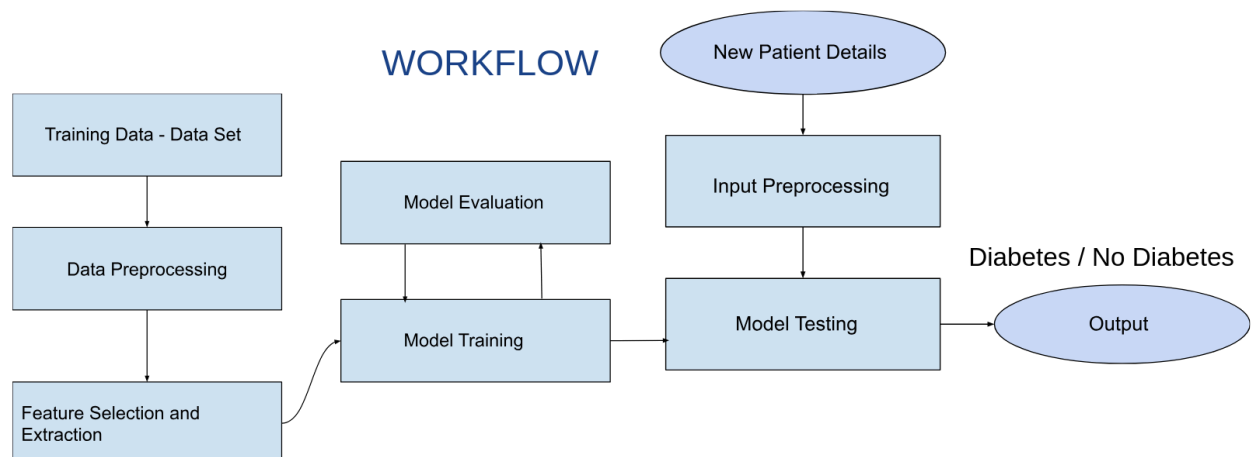


Fig.2 Back End Module

2.3.5 DATA FLOW DIAGRAM:

- Data flow diagrams are used to graphically represent the flow of data in a business information system. DFD describes the processes that are involved in a system to transfer data from the input to the file storage and report generation. Data flow diagrams can be divided into logical and physical. The logical data flow diagram describes the flow of data through a system to perform certain functionality of a business. The physical data flow diagram describes the implementation of the logical data flow



CHAPTER-3

IMPLEMENTATION

3.1 Development Environment

3.1.1 Tools and Software Used

For the development of the Diabetes prediction classifier, the following tools and software were utilized:

- **Python:** The primary programming language used for developing the SMS spam classifier. Python's extensive libraries and frameworks made it an ideal choice for implementing machine learning models and data processing.
- **Jupyter Notebook:** This interactive development environment was used for experimenting with the model and performing data analysis. Jupyter Notebook allowed for an iterative approach to code development and testing.
- **Visual Studio Code (VS Code):** This code editor was employed for writing and organizing the project's code. VS Code's extensions and features enhanced productivity and code management.
- **VirtualBox:** A virtualization software used to create a controlled environment for development and testing. It ensured that the development environment remained consistent across different machines.
- **Streamlit:** A framework used for creating the web application interface for the SMS spam classifier. Streamlit facilitated the deployment of the model, allowing users to interact with it through a web browser.
- **Scikit-learn:** This machine learning library was used to build and evaluate the classifier model. It provided the necessary tools for data preprocessing, model training, and validation.
- **Pandas:** A data manipulation and analysis library in Python. Pandas was used for handling and processing the dataset.
- **Numpy:** A fundamental package for scientific computing in Python, used for numerical operations and data manipulation.

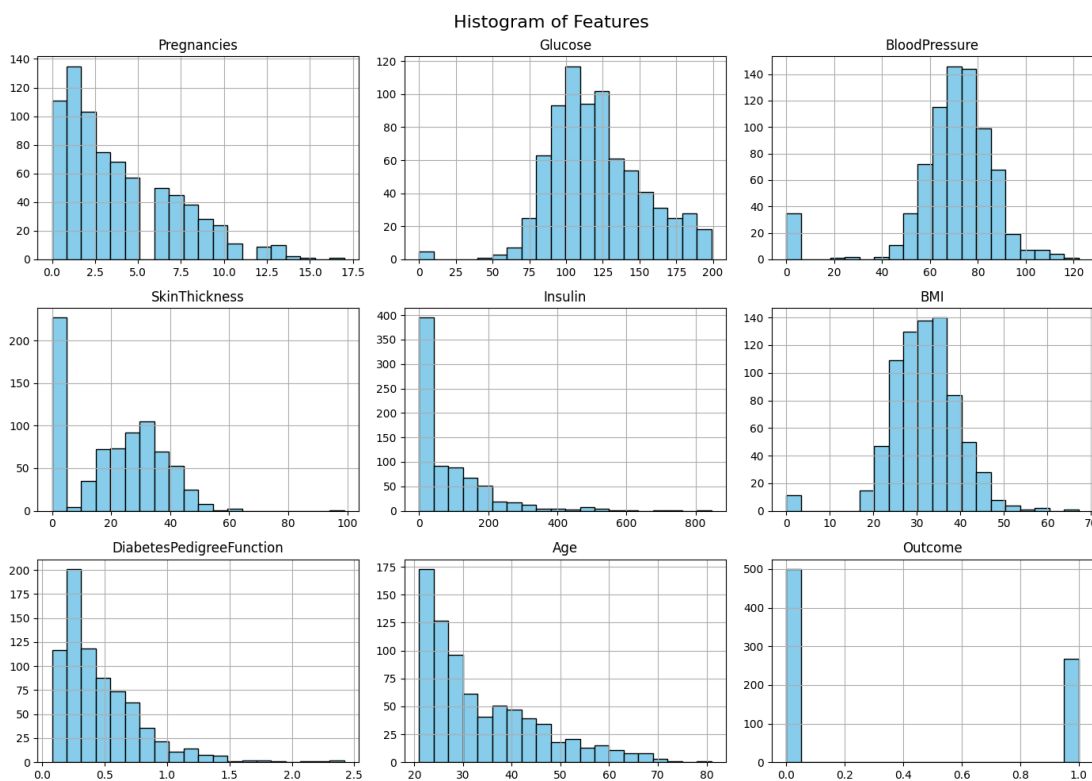
3.1.2 System Requirements

To ensure the smooth functioning of the development environment, the following system requirements were necessary:

- **Operating System:** Windows 10 or higher
- **Processor:** Intel Core i5 or equivalent
- **Memory:** 8 GB RAM or more
- **Storage:** 50 GB of free disk space
- **Python Version:** Python 3.12.3
- **IDE/Editor:** Jupyter Notebook and Visual Studio Code
- **Virtualization Software:** VirtualBox

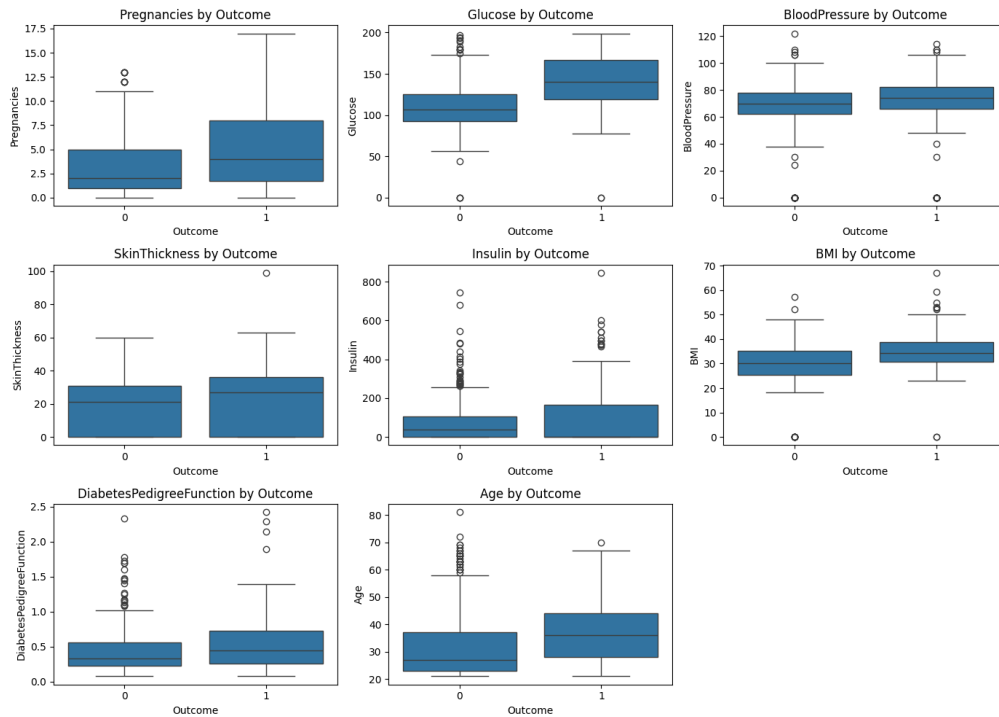
3.2 Visualizations and Charts

Visualizations play a crucial role in understanding the data and the model's performance.



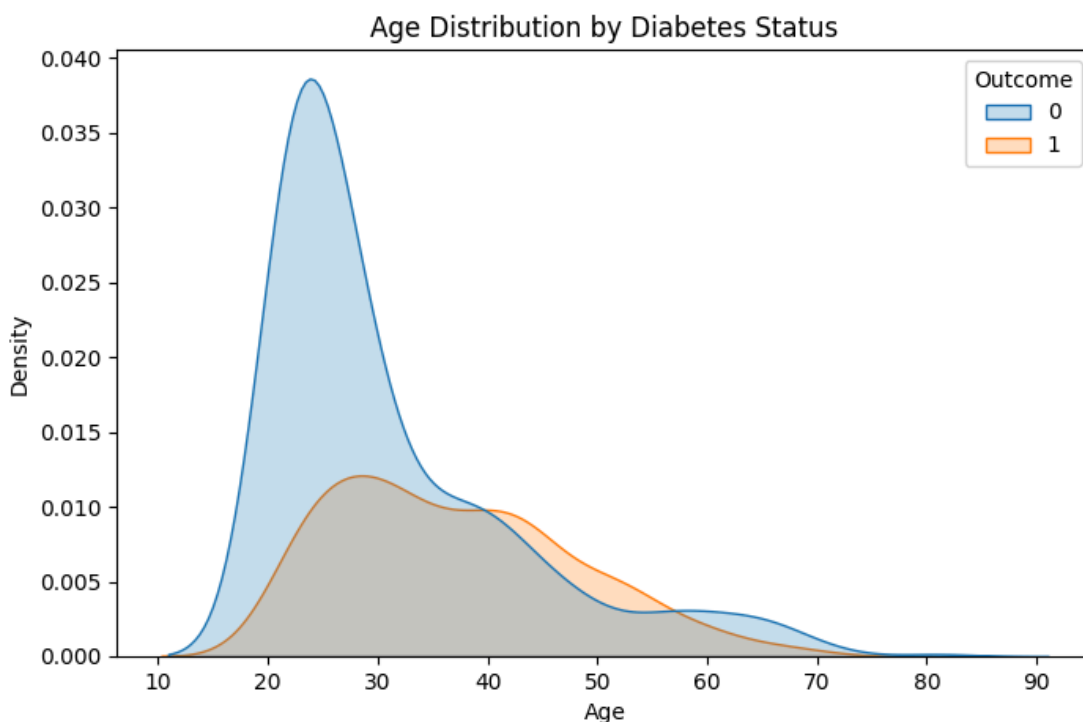
3.2.1 Histogram for each numerical feature

The above diagram provides a visual representation of the distribution of each numerical feature across the dataset. This visualization helps in better understanding of patients across different features.



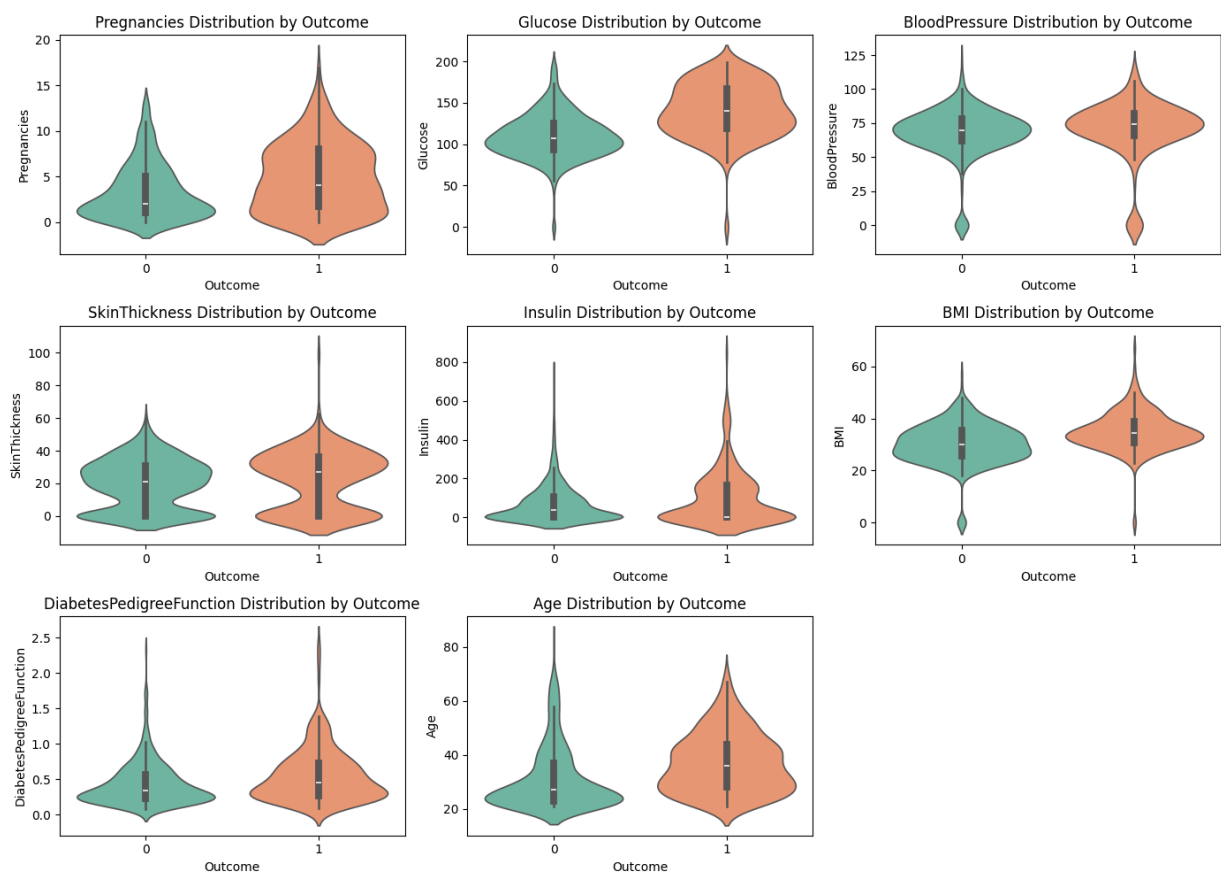
3.2.2 Box Plot by outcome

The above diagram provides a visual representation of the box plot of features across the dataset. This visualization helps in understanding the InterQuartile Regions for all the numerical features and concludes the presence of outliers in the dataset. This visualisation can be helpful in data cleaning.



3.2.3 Age Distribution by Diabetes

The above diagram provides a visual representation of the age distribution by diabetes in the dataset. This visualization helps in understanding the relation between age and diabetes. From the above diagram we can conclude the most of the diabetic patients lie between the age of year 25 to 35.



3.2.4 Violin plot Visualization

The above diagram provides a visual representation of the distribution of diabetic and non-diabetic patients. This visualization provides more detail than box plot. A violin plot depicts distributions of numeric data for one or more groups using density curves. The width of each curve corresponds with the approximate frequency of data points in each region. Densities are frequently accompanied by an overlaid chart type, such as box plot, to provide additional information.

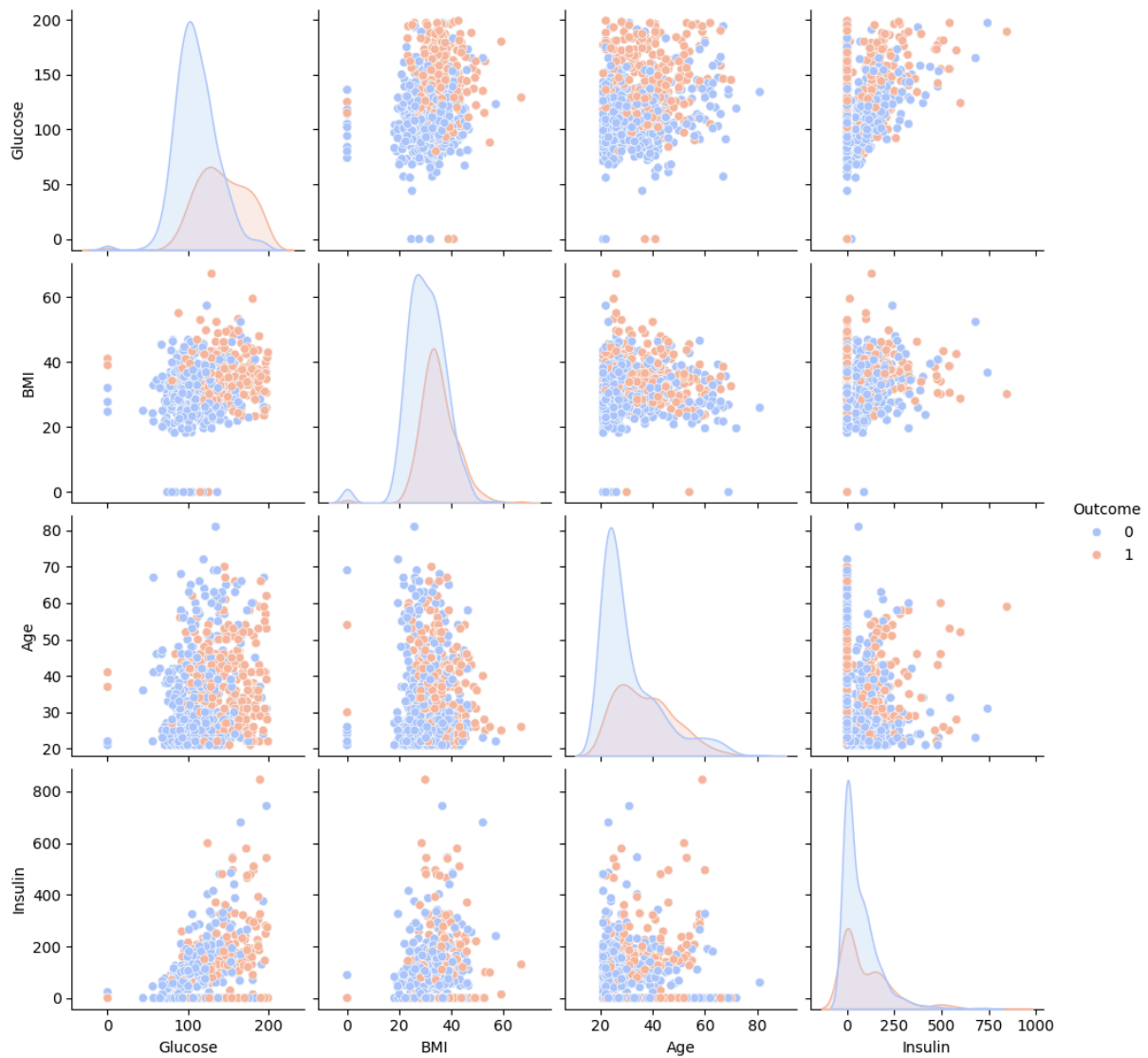


Fig.7 Pairplot Visualization

3.2.5 Pairplot Visualization

The following code snippet and corresponding diagram provide a comprehensive visual representation of the relationships between multiple features in the dataset. The pairplot displays scatter plots for each pair of features, along with histograms for each feature, colored by the target variable. This visualization helps in understanding how different features interact with each other and their individual distributions.

3.3 Key function, Modules and User Interface

3.3.1 Key Functions and Modules

Used Pandas

- **Purpose:** Pandas is a powerful data manipulation library in Python that provides data structures like DataFrame to efficiently handle and analyze large datasets.
- **Key Functions:**
 - `read_csv()`: Loads the dataset from a CSV file.
 - `drop()`: Removes unnecessary columns.
 - `isnull()`, `fillna()`: Handles missing values.

NumPy

- **Purpose:** NumPy is a fundamental package for scientific computing in Python, offering support for large, multi-dimensional arrays and matrices.
- **Key Functions:**
 - `array()`: Creates an array.
 - `reshape()`: Reshapes data without changing its data.
 - `mean()`, `std()`: Computes mean and standard deviation.

Scikit-learn

- **Purpose:** It is used for feature extraction, model selection, training, and evaluation in our SMS spam detection project.
- **Key Functions:**
 - `train_test_split()`: Splits the dataset into training and testing sets.
 - `LogisticRegression()`: Logistic Regression classifier.
 - `accuracy_score()`, `precision_score()`, `confusion_matrix()`: Evaluates model performance

Matplotlib

- **Purpose:** It is used for creating visualizations like bar charts and word clouds to analyze the dataset and model performance.
- **Key Functions:**
 - o `pyplot.figure()`, `pyplot.show()`: Creates and displays plots.
 - o `pyplot.hist()`: Plots histograms for data distribution.

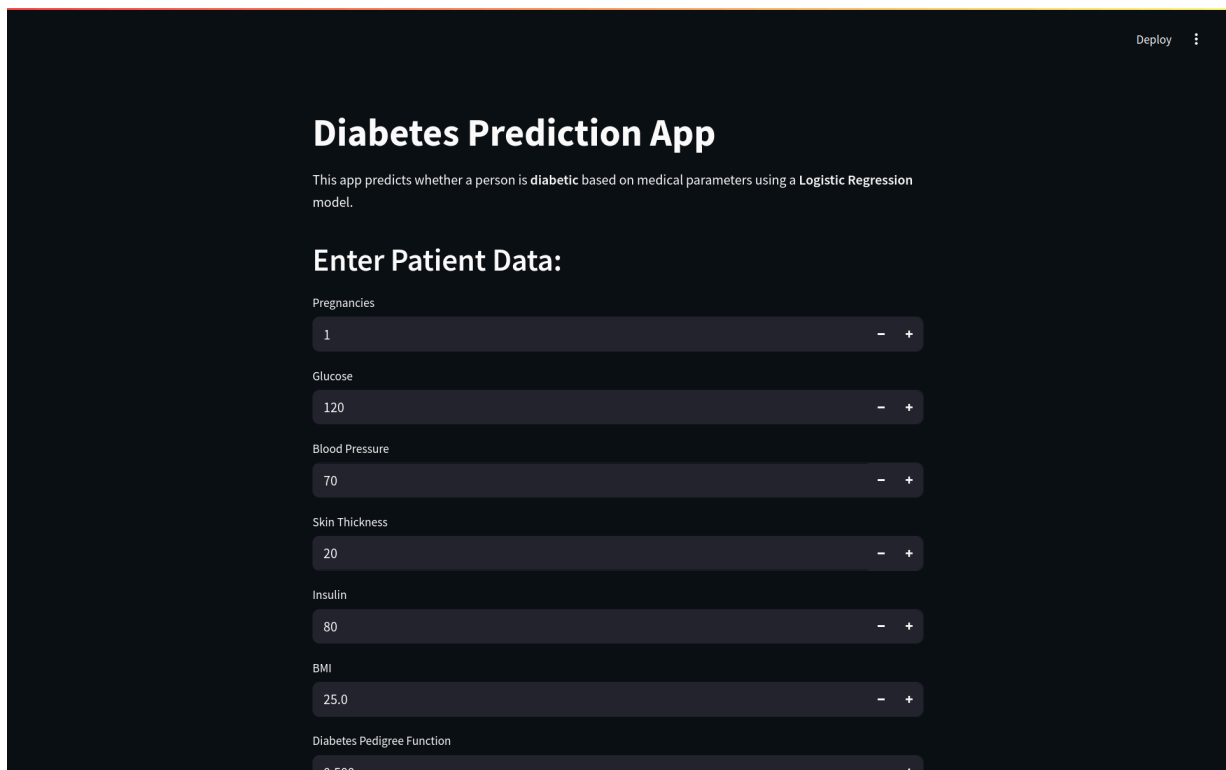
Seaborn

- **Purpose:** Seaborn is a Python visualization library based on Matplotlib that provides a high-level interface for drawing attractive and informative statistical graphics.
- **Key Functions:**
 - o `sns.heatmap()`: Plots heatmaps.
 - o `sns.pairplot()`: Plots pairwise relationships in a dataset.

Streamlit

- **Purpose:** Streamlit is an open-source Python library that makes it easy to create and share beautiful, custom web apps for machine learning and data science.
- **Key Functions:**
 - o `st.title()`, `st.write()`: Displays text in the web app.
 - o `st.file_uploader()`: Allows file upload in the web app.
 - o `st.button()`: Adds a button to trigger actions.
 - o `st.write()`, `st.success()`, `st.error()`: Displays messages based on predictions.

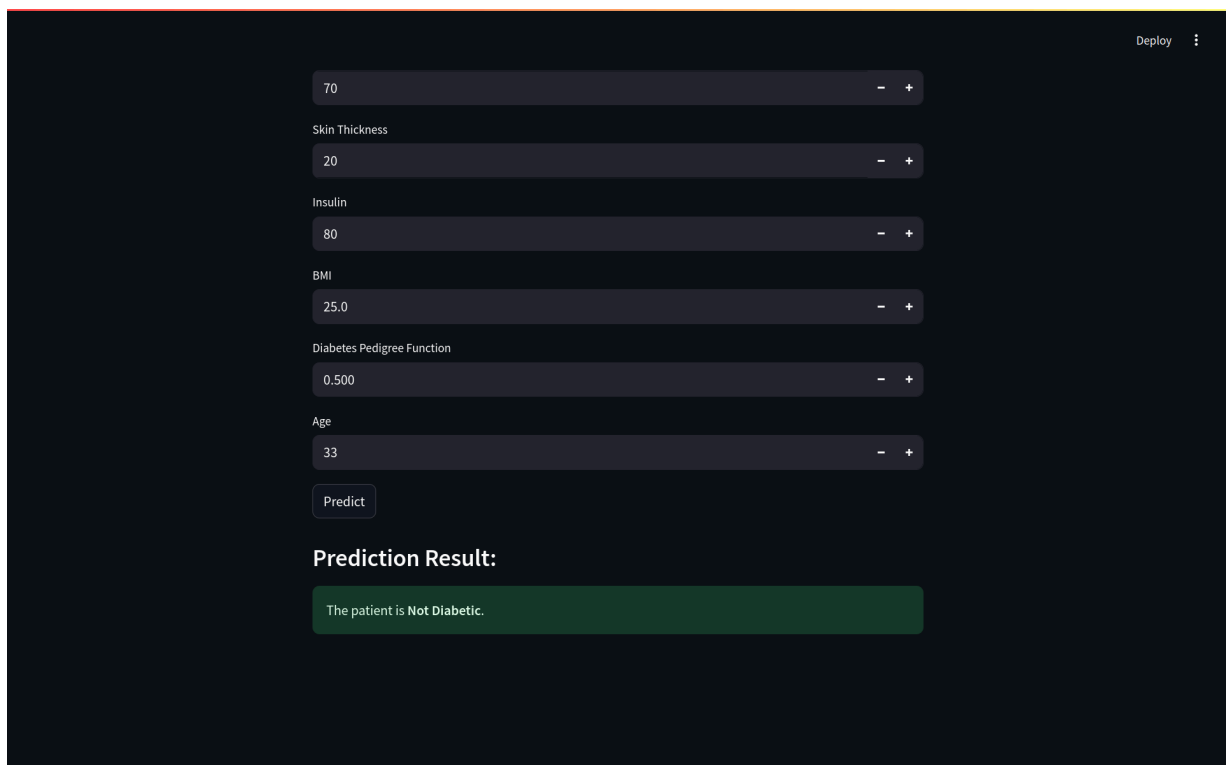
3.3.2 User interface



The screenshot shows the 'Diabetes Prediction App' interface. At the top right is a 'Deploy' button. Below the title, a description states: 'This app predicts whether a person is diabetic based on medical parameters using a Logistic Regression model.' The section 'Enter Patient Data:' contains several input fields with numerical values and minus/plus buttons for adjustment:

Parameter	Value
Pregnancies	1
Glucose	120
Blood Pressure	70
Skin Thickness	20
Insulin	80
BMI	25.0
Diabetes Pedigree Function	0.500

Fig.12 User Interface - Input



The screenshot shows the output of the app. The input fields from the previous figure are still visible. Below them is a 'Predict' button. The 'Prediction Result:' section displays a green box with the text: 'The patient is Not Diabetic.'

Fig.13 User Interface - Output

CHAPTER-4

EVALUATION

4.1 Model Evaluation Metrics for SMS Spam Detection

4.1.1 Metrics Used

1. Accuracy

- o **Definition:** The ratio of correctly predicted instances to the total instances. It is calculated as:

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Instances}}$$

- o **Usage:** Measures the overall correctness of the model. In the SMS spam detection model, accuracy was used to provide a general sense of how well the model distinguishes between spam and ham messages.

2. Precision

Definition: The ratio of correctly predicted positive observations to the total predicted positives. It is calculated as:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

Usage: Indicates how many of the messages classified as spam were actually spam. High precision is crucial to minimize the number of legitimate messages incorrectly labeled as spam.

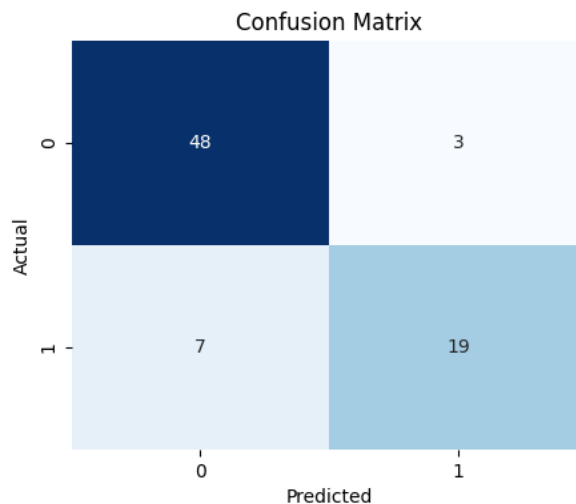
3. Confusion Matrix

Definition: The confusion matrix is a fundamental tool for evaluating the performance of a classification model. It provides a detailed breakdown of the model's predictions and actual outcomes, allowing us to understand the types of errors made.

Structure of the Confusion Matrix

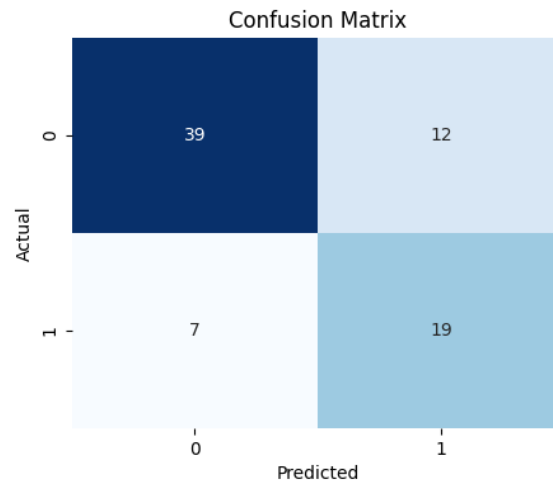
A confusion matrix is a 2x2 table for a binary classification problem. The four quadrants of the matrix represent:

- **True Positives (TP):** Correctly predicted diabetic patient.
- **True Negatives (TN):** Correctly predicted non-diabetic patient.
- **False Positives (FP):** non-diabetic patient incorrectly predicted as diabetic (Type I error).
- **False Negatives (FN):** Diabetic patient incorrectly predicted as non-diabetic (Type II error).



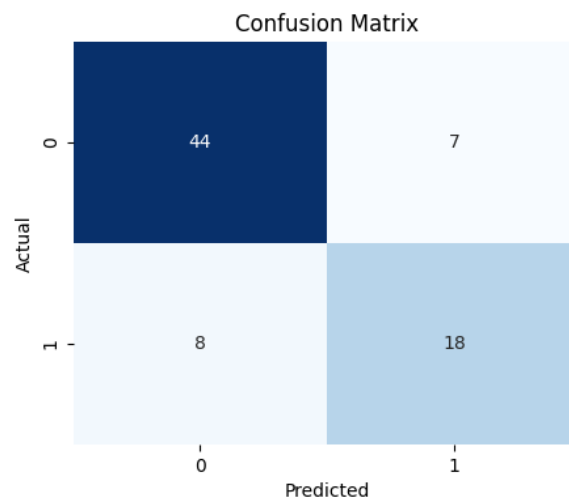
Here's the Logistic Regression confusion matrix visualization in test data:

- True Negatives (Non-Diabetic correctly classified): 48
- False Positives (Non-Diabetic incorrectly classified as Diabetic): 3
- False Negatives (Diabetic incorrectly classified as Non-Diabetic): 7
- True Positives (Diabetic correctly classified): 19



Here's the Decision Tree confusion matrix visualization in test data:

- True Negatives (Non-Diabetic correctly classified): 39
- False Positives (Non-Diabetic incorrectly classified as Diabetic): 12
- False Negatives (Diabetic incorrectly classified as Non-Diabetic): 7
- True Positives (Diabetic correctly classified): 19



Here's the Bernoulli Naïve Bayes confusion matrix visualization:

- True Negatives (Non-Diabetic correctly classified): 44
- False Positives (Non-Diabetic incorrectly classified as Diabetic): 7
- False Negatives (Diabetic incorrectly classified as Non-Diabetic): 8
- True Positives (Diabetic correctly classified): 18

4.1.2 Why These Metrics Are Important

1. **Accuracy: Importance:** While accuracy gives a quick snapshot of the model's performance, it might not be sufficient alone, especially with imbalanced datasets like diabetes prediction where the number of non-diabetic patients often outnumbers diabetic patients. High accuracy might be misleading if the model is biased towards predicting the majority class.
2. **Precision: Importance:** Precision is crucial in scenarios where false positives need to be minimized. In diabetes disease prediction, high precision ensures that legitimate patients are not incorrectly filtered out, which is essential for maintaining trust in the model.
3. **Confusion Matrix: Importance:** This tool provides a more granular view of the model's performance, showing not just how often it is correct, but also the types of errors it makes. This can guide further tuning and improvement of the model.

4.1 Results

The Diabetes Disease Prediction model was evaluated using several key performance metrics. The Logistic regression model provided the best results with outstanding precision and accuracy.

Model Used

The **Logistic Regression** model was selected for the final evaluation due to its superior performance.

4.1.1 Performance Metrics

Accuracy: The accuracy of the model is the proportion of correctly classified messages (both diabetic and non-diabetic) out of the total records. It is calculated as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Using this model, the accuracy achieved is:

Accuracy=0.87

This indicates that 87% of the records were correctly classified by the model.

Precision: Precision measures the proportion of correctly identified positive results (diabetic) out of all positive results predicted by the classifier. It is calculated as:

$$\text{Precision} = \frac{TP}{TP + FP}$$

For the Logistic Regression model, the precision score is:

Precision=0.86

This means that every message predicted as diabetic patient by the model was diabetic, showing an exceptional precision score of 86%.

CHAPTER-5

DEPLOYMENT

5.1 Streamlit App

5.1.1 Introduction to Streamlit

Streamlit is an open-source Python library that makes it easy to create and share custom web apps for machine learning and data science. It turns data scripts into shareable web apps in minutes. With Streamlit, you can build and deploy powerful data applications quickly and easily.

Key Features:

- Interactive widgets for user input.
- Real-time updates and seamless integration with Python scripts.
- Built-in support for data visualization libraries like Matplotlib, Seaborn, and Plotly.

5.1.2 Steps to Create the Streamlit App

1. Install Streamlit:

First, ensure that Streamlit is installed in your Python environment. You can install it using pip:

```
pip install streamlit
```

2. Create a New Python Script:

Create a new Python script (e.g., `diabetes_app.py`) where you will write the code for your Streamlit app.

3. Import Necessary Libraries:

Import Streamlit and other necessary libraries at the beginning of your script:

```
``import streamlit as st
import numpy as np
import pandas as pd
import pickle
import joblib
from sklearn.preprocessing import StandardScaler``
```

5.1.3 Complete Streamlit App Code:

```
import streamlit as st
import numpy as np
import pandas as pd
import pickle
import joblib
from sklearn.preprocessing import StandardScaler

# Load the trained model
model = joblib.load("log_reg_diabetes_model.pkl")

# Title and description
st.title("Diabetes Prediction App")
st.write("""
This app predicts whether a person is **diabetic** based on medical parameters
using a **Logistic Regression** model.
""")

# Input form
st.header("Enter Patient Data:")

pregnancies = st.number_input("Pregnancies", min_value=0, max_value=20,
value=1)
glucose = st.number_input("Glucose", min_value=0, max_value=300, value=120)
blood_pressure = st.number_input("Blood Pressure", min_value=0,
max_value=200, value=70)
skin_thickness = st.number_input("Skin Thickness", min_value=0, max_value=100,
value=20)
insulin = st.number_input("Insulin", min_value=0, max_value=1000, value=80)
bmi = st.number_input("BMI", min_value=0.0, max_value=70.0, value=25.0,
format="%.1f")
dpf = st.number_input("Diabetes Pedigree Function", min_value=0.0,
max_value=2.5, value=0.5, format="%.3f")
age = st.number_input("Age", min_value=1, max_value=120, value=33)
```

```
# Prediction
if st.button("Predict"):
    input_data = np.array([[pregnancies, glucose, blood_pressure, skin_thickness,
                             insulin, bmi, dpf, age]])

    scaler = StandardScaler()
    x = scaler.fit_transform(input_data)

    prediction = model.predict(x)
    result = "Diabetic" if prediction[0] == 1 else "Not Diabetic"

    st.subheader("Prediction Result:")
    st.success(f"The patient is **{result}**.")
```

CHAPTER-6
CONCLUSION
AND
FUTURE WORK

6.1 **Conclusion-** In this project, we developed a robust Diabetes Disease Prediction system using a Logistic Regression classifier. The model was trained and tested on a dataset of diabetes patients, with a significant focus on achieving high accuracy and precision. The key achievements and findings of this project are summarized below:

- **High Precision and Accuracy:** The Logistic Regression model achieved a precision score of 86% and an accuracy score of 87% on the test data. This indicates that the model is highly effective in distinguishing diabetic patients from non-diabetic patients, with minimal false positives.
- **Efficient Use of Confusion Matrix:** The confusion matrix provided insights into the true positives, true negatives, false positives, and false negatives. This helped in understanding the model's performance in a detailed manner and highlighted its effectiveness in correctly classifying messages.
- **User-Friendly Streamlit App:** A Streamlit-based web application was developed to make the model accessible to end-users. The app allows users to input a patient's details and receive immediate diabetic classification results, enhancing user interaction and usability.
- **Deployment and Challenges:** The app was successfully deployed, with careful handling of challenges such as dependency management, file loading issues, and network connectivity. These steps ensured a smooth and reliable user experience.

Overall, the project demonstrates the potential of machine learning techniques, particularly Logistic Regression classifiers, in developing effective diabetes disease prediction systems. The High performance metrics indicate that the model can be reliably used in real-world applications to detect diabetes patients and improve communication efficiency.

6.2 Future Work

While the current implementation of the diabetes disease prediction system is effective, there are several areas for potential improvement and future exploration:

1. **Integration with Other Classifiers:** Exploring and integrating other machine learning classifiers, such as Boosting and Bagging, could further improve the model's accuracy and robustness. Combining multiple classifiers in an ensemble approach might yield better performance and reduce the likelihood of misclassifications.
2. **Handling Imbalanced Data:** Although the current model performs well, further techniques to handle imbalanced datasets, such as SMOTE (Synthetic Minority Over-sampling Technique) or cost-sensitive learning, could be implemented to ensure even better performance on datasets with a high imbalance ratio.
3. **Feature Engineering:** Additional feature engineering techniques could be explored to improve the model's ability to differentiate between spam and non-spam messages. This includes using word embeddings, bi-grams, and tri-grams, or incorporating meta-features such as message length and sender information.
4. **Real-time Disease Detection:** Implementing the system in a real-time environment, such as integrating with messaging platforms or email services, to provide instant disease detection and filtering. Ensuring the system is scalable and can handle large volumes of messages with minimal latency.
5. **User Feedback Mechanism:** Introducing a feedback mechanism in the Streamlit app to allow users to report misclassified messages. This feedback could be used to retrain and fine-tune the model, improving its accuracy over time.
6. **Enhanced Data Privacy and Security:** Ensuring that the system adheres to data privacy regulations and implements robust security measures to protect user data. Exploring techniques for anonymizing data to ensure user privacy while maintaining the effectiveness of the model.

6.3 REFERENCES

- ❑ Kavakiotis, I., Tsave, O., Salifoglou, A., Maglaveras, N., Vlahavas, I., & Chouvarda, I. (2017). *Machine learning and data mining methods in diabetes research*. **Computational and Structural Biotechnology Journal**, 15, 104–116. <https://doi.org/10.1016/j.csbj.2016.12.005>
- ❑ Smith, J. W., Everhart, J. E., Dickson, W. C., Knowler, W. C., & Johannes, R. S. (1988). *Using the ADAP learning algorithm to forecast the onset of diabetes mellitus*. **Proceedings of the Annual Symposium on Computer Application in Medical Care**, 261–265.
- ❑ Sisodia, D., & Sisodia, D. S. (2018). *Prediction of diabetes using classification algorithms*. **Procedia Computer Science**, 132, 1578–1585. <https://doi.org/10.1016/j.procs.2018.05.221>
- ❑ Jayanthi, A., Ramesh, D., & Sivakumar, S. (2017). *Machine learning algorithms for diabetes diagnosis: A comparative study*. **International Journal of Engineering and Technology**, 7(2.7), 223–226.
- ❑ Kumar, A., & Vohra, R. (2020). *Comparative analysis of machine learning models for diabetes prediction*. **International Journal of Engineering Research & Technology**, 9(4), 179–183.
- ❑ Polat, K., & Güneş, S. (2007). *An expert system approach based on principal component analysis and adaptive neuro-fuzzy inference system to diagnosis of diabetes disease*. **Digital Signal Processing**, 17(4), 702–710. <https://doi.org/10.1016/j.dsp.2006.09.005>
- ❑ Zhang, X., Zhao, L., Qi, L., & Liu, X. (2019). *Deep learning-based analysis of medical data for diabetes prediction*. **Multimedia Tools and Applications**, 78(18), 25047–25063. <https://doi.org/10.1007/s11042-018-7151-0>
- ❑ Deo, R. C. (2015). *Machine learning in medicine*. **Circulation**, 132(20), 1920–1930. <https://doi.org/10.1161/CIRCULATIONAHA.115.001593>
- ❑ Nayak, S. R., Dash, R., Majhi, B., & Mishra, D. (2013). *A hybrid approach for classification using machine learning techniques*. **International Journal of Computer Applications**, 73(6), 30–34.
- ❑ Liu, Y., Chen, P. H. C., & Krause, J. (2022). *How to read articles that use machine learning: Users' guides to the medical literature*. **JAMA**, 327(22), 2211–2217. <https://doi.org/10.1001/jama.2022.6627>