Diabetes Detection Using Machine Learning

Group Members:

AQDAS SULTAN,
ASANSOL ENGINEERING COLLEGE,
221080110375

SHIVAM PANDEY,
ASANSOL ENGINEERING COLLEGE,
221080110440

ANAS TANWEER,
ASANSOL ENGINEERING COLLEGE,
221080110366

ALTAMASH AHMED,
ASANSOL ENGINEERING COLLEGE,
221080110363

Table of Contents

Acknowledgement

I take this opportunity to express my profound gratitude and deep regards to my faculty Dr. Arnab Chakraborty for his exemplary guidance, monitoring, and constant encouragement throughout the course of this project. The blessing, help and guidance given by him time to time shall carry me a long way in the journey of life on which I am about to embark.

I am obliged to my project team members for the valuable information provided by them in their respective fields. I am grateful for their cooperation during the period of my assignment.

Group Members Name:

AQDAS SULTAN,
ASANSOL ENGINEERING COLLEGE,
221080110375

SHIVAM PANDEY,
ASANSOL ENGINEERING COLLEGE,
221080110440

ANAS TANWEER,
ASANSOL ENGINEERING COLLEGE,
221080110366

ALTAMASH AHMED,
ASANSOL ENGINEERING COLLEGE,
221080110363

Project Objective

Describing the Problem: Diabetes Detection

Problem Statement:

Diabetes mellitus is a chronic disease that affects millions of people worldwide. Early detection and timely intervention are crucial to prevent severe health complications. However, traditional diagnostic methods can be time-consuming and costly.

The Challenge:

To address this challenge, we aim to develop a machine learning model that can accurately predict the onset of diabetes based on various health indicators. This model will assist healthcare professionals in early diagnosis, allowing for timely treatment and improved patient outcomes.

Objective of the Diabetes Detection Project

The primary objective of this project is to develop a robust machine learning model capable of accurately predicting the onset of diabetes mellitus based on a set of relevant health indicators.

Problem Solving

To develop a machine learning model that accurately predicts the onset of diabetes based on relevant health indicators.

Proposed Solution:

- **Data Collection and Preprocessing:** Gather and clean healthcare data.
- **Model Selection and Training:** Train a logistic regression model on the prepared data.
- **Model Evaluation:** Assess the model's performance using various metrics.
- **Model Deployment:** Deploy the model as a web application or API for real-world use.

Project Scope

Project Scope: Diabetes Detection System

This project aims to develop a machine learning-based system capable of accurately predicting the onset of diabetes mellitus. The system will:

- **Data Acquisition and Preprocessing:** Collect and clean a dataset containing relevant health indicators, such as glucose levels, blood pressure, and BMI.
- **Model Development and Training:** Train a logistic regression model on the pre-processed data to learn the relationship between the input features and the target variable (diabetes diagnosis).
- **Model Evaluation:** Evaluate the model's performance using metrics like accuracy, precision, recall, and F1-score.
- **User Interface Development:** Create a user-friendly interface (e.g., web application, mobile app) to allow users to input their health information and receive a predicted diabetes risk assessment.
- **Deployment:** Deploy the system to a suitable platform (e.g., cloud-based server, local server) to make it accessible to users.

Limitations:

- The model's accuracy will depend on the quality and quantity of the training data.
- The system may not be able to account for all potential factors influencing diabetes risk.
- The system is intended for predictive purposes and should not be used as a substitute for professional medical advice.

Data Description

Source Of Data: Kaggle. The given dataset is a shortened version of the original dataset in Kaggle.

Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	Diabetes Pedigree Function	Age	Outcome
6	148	72	35	0	33.6	0.627	50	1
1	85	66	29	0	26.6	0.351	31	0
8	183	64	0	0	23.3	0.672	32	1
1	89	66	23	94	28.1	0.167	21	0
0	137	40	35	168	43.1	2.288	33	1
5	116	74	0	0	25.6	0.201	30	0
3	78	50	32	88	31	0.248	26	1
10	115	0	0	0	35.3	0.134	29	0
2	197	70	45	543	30.5	0.158	53	1
8	125	96	0	0	0	0.232	54	1
4	110	92	0	0	37.6	0.191	30	0
10	168	74	0	0	38	0.537	34	1
10	139	80	0	0	27.1	1.441	57	0
1	189	60	23	846	30.1	0.398	59	1
5	166	72	19	175	25.8	0.587	51	1
7	100	0	0	0	30	0.484	32	1
0	118	84	47	230	45.8	0.551	31	1
7	107	74	0	0	29.6	0.254	31	1
1	103	30	38	83	43.3	0.183	33	0
1	115	70	30	96	34.6	0.529	32	1
3	126	88	41	235	39.3	0.704	27	0
8	99	84	0	0	35.4	0.388	50	0
7	196	90	0	0	39.8	0.451	41	1
9	119	80	35	0	29	0.263	29	1
11	143	94	33	146	36.6	0.254	51	1
10	125	70	26	115	31.1	0.205	41	1
7	147	76	0	0	39.4	0.257	43	1
1	97	66	15	140	23.2	0.487	22	0
13	145	82	19	110	22.2	0.245	57	0
5	117	92	0	0	34.1	0.337	38	0

Model Building

Model Description

Logistic Regression

Logistic regression is a statistical method used for binary classification problems. It models the probability[1] of a binary response variable (in this case, diabetes diagnosis) based on one or more predictor variables.

Training Phase:

- The model is trained on the pre-processed dataset, learning the relationship between the input features (e.g., glucose level, blood pressure) and the target variable (diabetes diagnosis).
- The model's parameters (weights and bias) are adjusted iteratively using an optimization algorithm like gradient descent to minimize the loss function (e.g., cross-entropy loss).

Testing Phase:

- The trained model is evaluated on a separate testing dataset.
- The model predicts the probability of diabetes for each test instance.
- The predicted probabilities are converted into binary predictions (0 or 1) using a threshold.
- The model's performance is assessed using various metrics:
    - **Accuracy:** Proportion of correctly classified instances.
    - **Precision:** Proportion of positive predictions that are actually positive.
    - **Recall:** Proportion of actual positive instances that are correctly identified.
    - **AUC-ROC Curve:** Area Under the Receiver Operating Characteristic curve, which measures the model's ability to distinguish between positive and negative[2] classes.

Model Selection and Evaluation

- **Multiple Models:** While logistic regression is the primary model used, other models like decision trees, random forests, or support vector machines could be explored and compared.
- **Hyperparameter Tuning:** Experiment with different hyperparameters (e.g., regularization strength, learning rate) to optimize the model's performance.
- **Cross-Validation:** Use cross-validation techniques to assess the model's generalization performance and reduce the risk of overfitting.
- **Model Selection:** The final model is selected based on its performance on the validation set and its interpretability.

By carefully selecting and tuning the model, we aim to achieve a high level of accuracy and reliability in diabetes prediction.

Code

To further improve the diabetes detection model, we can consider the following strategies:

1. Data Quality and Quantity:

- **Data Cleaning:** Ensure data quality by addressing missing values, outliers, and inconsistencies.
- **Data Augmentation:** Generate synthetic data to increase the dataset size and improve model generalization.
- **Feature Engineering:** Create new features or transform existing ones to capture more relevant information.

2. Model Selection and Hyperparameter Tuning:

- **Ensemble Methods:** Combine multiple models (e.g., bagging, boosting) to improve predictive accuracy.
- **Deep Learning:** Explore deep learning techniques, such as neural networks, for complex pattern recognition.
- **Hyperparameter Optimization:** Use techniques like grid search or randomized search to find the optimal hyperparameters for the chosen model.

3. Model Evaluation and Validation:

- **Cross-Validation:** Employ cross-validation to assess the model's performance on different subsets of the data.
- **Confusion Matrix Analysis:** Analyze the confusion matrix to identify areas where the model is underperforming and focus on improving those areas.
- **Class Imbalance Handling:** If the dataset is imbalanced (more instances of one class than the other), use techniques like oversampling, undersampling, or class weighting to address the imbalance.

4. Model Interpretability:

- **Feature Importance Analysis:** Identify the most important features that contribute to the model's predictions.
- **Explainable AI Techniques:** Use techniques like SHAP or LIME to explain the model's decision-making process.

5. Continuous Improvement:

- **Regular Model Retraining:** Retrain the model periodically with new data to adapt to changing patterns and improve performance.
- **User Feedback:** Gather feedback from users to identify areas for improvement and incorporate it into the model's development.

By implementing these strategies, we can enhance the accuracy, reliability, and interpretability of the diabetes detection model, ultimately leading to better healthcare outcomes.

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, roc_auc_score, roc_curve, accuracy_score
import matplotlib.pyplot as plt

# Load the dataset
data = pd.read_csv('DIABETICS DETECTION_DATA SET.csv')

# Replace zeros with NaN for specific columns
columns_with_zeros = ['Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI']
data[columns_with_zeros] = data[columns_with_zeros].replace(0, pd.NA)

# Fill missing values with the median of each column
data.fillna(data.median(), inplace=True)

# Define features and target variable
X = data.drop('Outcome', axis=1)
y = data['Outcome']

# Split the data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42, stratify=y)

# Initialize and train the Logistic Regression model
model = LogisticRegression(max_iter=1000, random_state=42)
model.fit(X_train, y_train)

# Make predictions
y_pred = model.predict(X_test)
y_pred_proba = model.predict_proba(X_test)[:, 1]
```

```
# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
roc_auc = roc_auc_score(y_test, y_pred_proba)
print(f"Accuracy: {accuracy:.2f}")
print(f"ROC-AUC Score: {roc_auc:.2f}")
print("\nClassification Report:")
print(classification_report(y_test, y_pred))

# Plot ROC curve
fpr, tpr, thresholds = roc_curve(y_test, y_pred_proba)
plt.figure(figsize=(8, 6))
plt.plot(fpr, tpr, label=f"ROC Curve (AUC = {roc_auc:.2f})", color='blue')
plt.plot([0, 1], [0, 1], 'k--', label="Random Classifier")
plt.xlabel("False Positive Rate")
plt.ylabel("True Positive Rate")
plt.title("ROC Curve")
plt.legend()
plt.show()

from google.colab import drive
drive.mount('/content/drive')

import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Load the dataset
data = pd.read_csv('DIABETICS DETECTION_DATA SET.csv')

# Replace zeros with NaN for specific columns
columns_with_zeros = ['Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI']
```

```
data[columns_with_zeros] = data[columns_with_zeros].replace(0, pd.NA)

# Fill missing values with the median for simplicity
data.fillna(data.median(), inplace=True)

# Set a style for Seaborn
sns.set(style="whitegrid")

# 1. Distribution plot for numerical features
plt.figure(figsize=(12, 8))
for i, column in enumerate(data.columns[:-1], 1):
    plt.subplot(3, 3, i)
    sns.histplot(data[column], kde=True, bins=30, color='blue')
    plt.title(f"Distribution of {column}")
plt.tight_layout()
plt.show()

# 2. Correlation heatmap
plt.figure(figsize=(10, 8))
correlation = data.corr()
sns.heatmap(correlation, annot=True, fmt=".2f", cmap="coolwarm", square=True)
plt.title("Correlation Heatmap")
plt.show()

# 3. Pair plot for feature relationships
sns.pairplot(data, hue='Outcome', diag_kind='kde', palette='Set2')
plt.suptitle("Pair Plot of Features", y=1.02)
plt.show()

# 4. Boxplot for feature distributions by Outcome
plt.figure(figsize=(12, 8))
for i, column in enumerate(data.columns[:-1], 1):
```

```
    plt.subplot(3, 3, i)
    sns.boxplot(x='Outcome', y=column, data=data, palette='Set1')
    plt.title(f"{column} by Outcome")
plt.tight_layout()
plt.show()

# 5. Count plot for Outcome
plt.figure(figsize=(6, 4))
sns.countplot(x='Outcome', data=data, palette='Set1')
plt.title("Count of Diabetes Outcomes")
plt.xlabel("Outcome (0: No Diabetes, 1: Diabetes)")
plt.ylabel("Count")
plt.show()
```

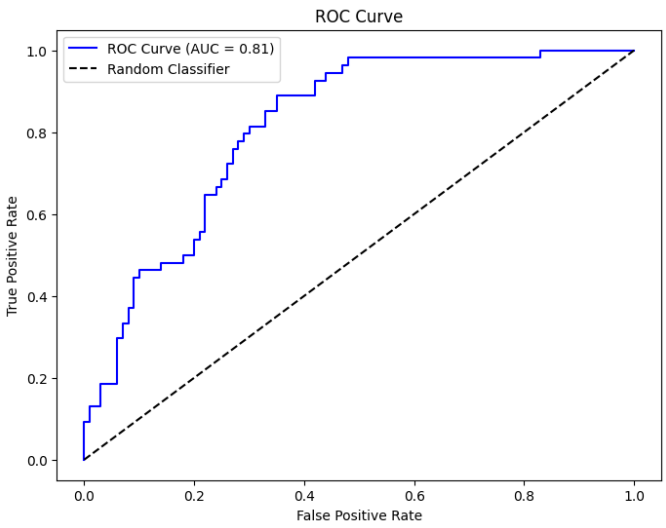
OUITPUT:

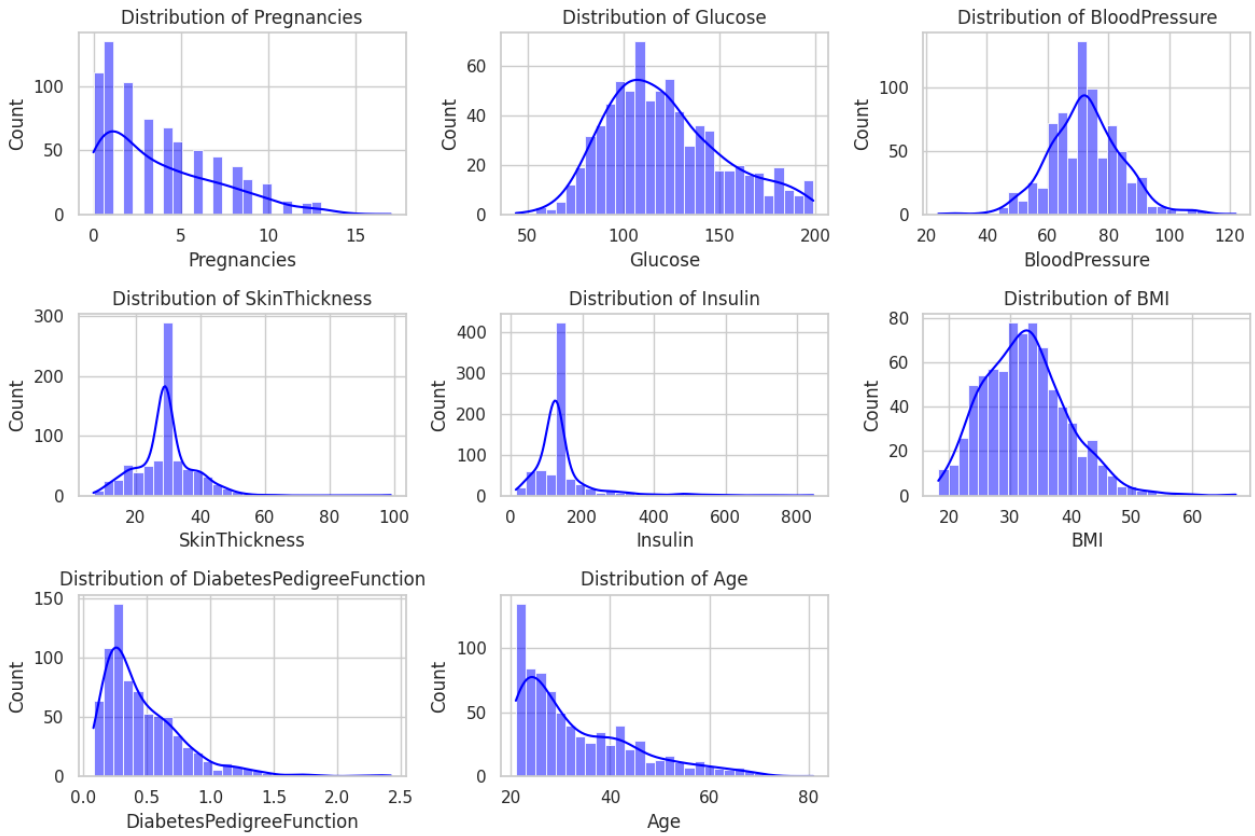
Accuracy: 0.70

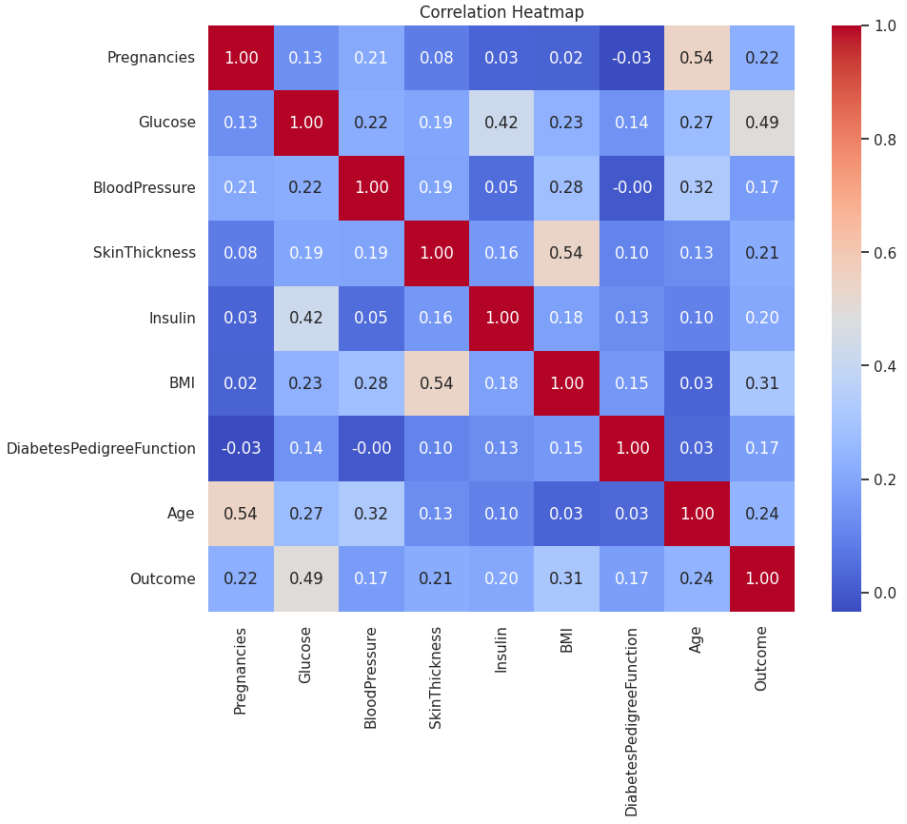
ROC-AUC Score: 0.81

Classification Report:

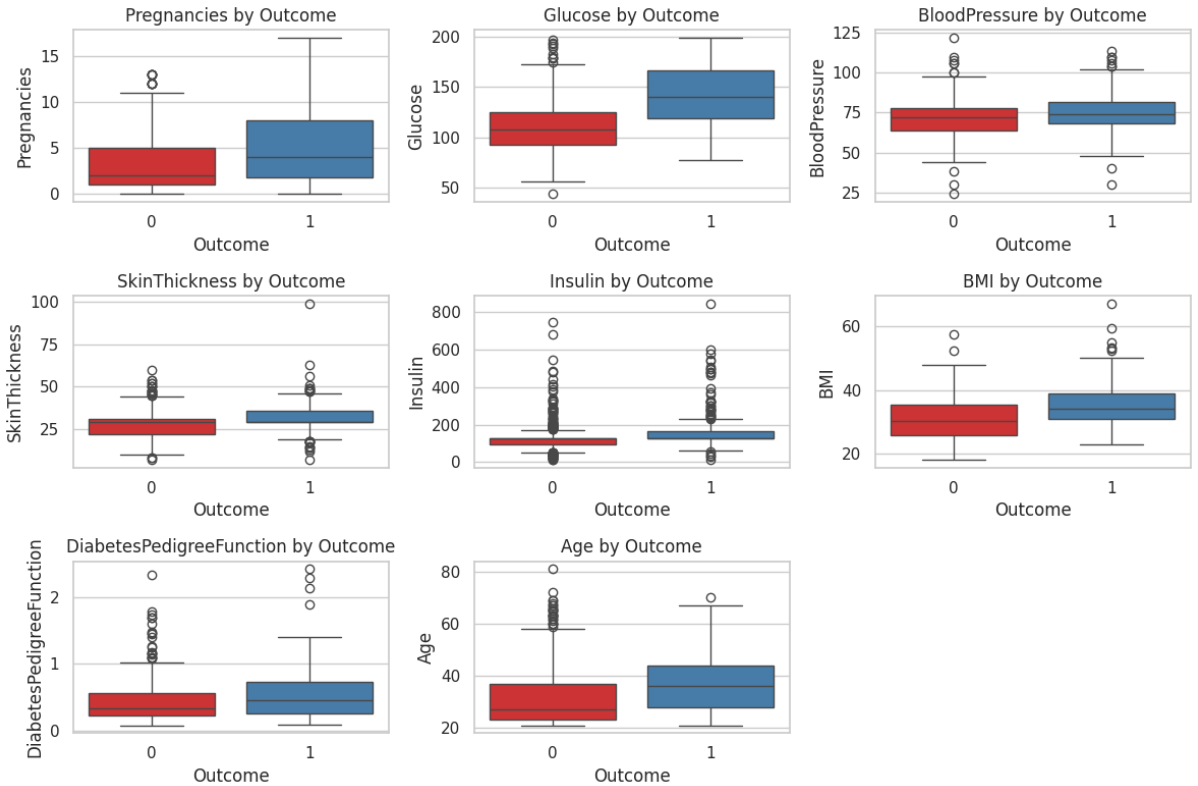
	precision	recall	f1-score	support
0	0.75	0.81	0.78	100
1	0.59	0.50	0.54	54
accuracy			0.70	154
macro avg	0.67	0.66	0.66	154
weighted avg	0.69	0.70	0.70	154

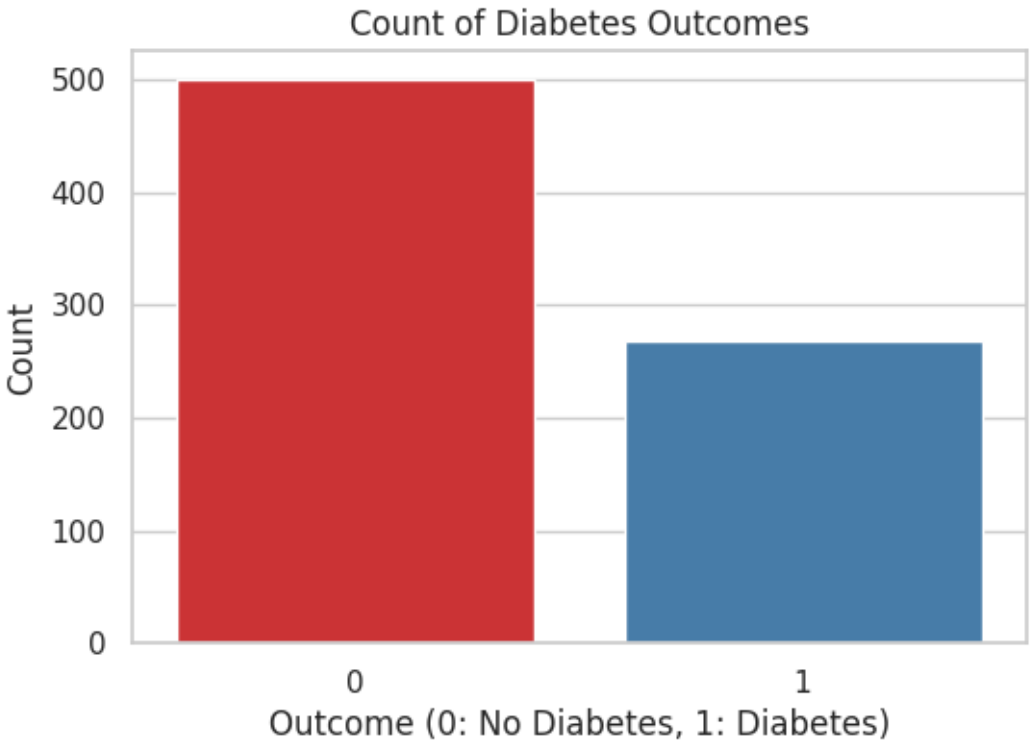



Distribution of Pregnancies
Count
Pregnancies
Distribution of Glucose
Count
Glucose
Distribution of BloodPressure
Count
BloodPressure
Distribution of SkinThickness
Count
SkinThickness
Distribution of Insulin
Count
Insulin
Distribution of BMI
Count
BMI
Distribution of DiabetesPedigreeFunction
Count
DiabetesPedigreeFunction
Distribution of Age
Count
Age


Correlation Heatmap
Pregnancies 1.00 0.13 0.21 0.08 0.03 0.02 -0.03 0.54 0.22
Glucose 0.13 1.00 0.22 0.19 0.42 0.23 0.14 0.27 0.49
BloodPressure 0.21 0.22 1.00 0.19 0.05 0.28 -0.00 0.32 0.17
SkinThickness 0.08 0.19 0.19 1.00 0.16 0.54 0.10 0.13 0.21
Insulin 0.03 0.42 0.05 0.16 1.00 0.18 0.13 0.10 0.20
BMI 0.02 0.23 0.28 0.54 0.18 1.00 0.15 0.03 0.31
DiabetesPedigreeFunction -0.03 0.14 -0.00 0.10 0.13 0.15 1.00 0.03 0.17
Age 0.54 0.27 0.32 0.13 0.10 0.03 0.03 1.00 0.24
Outcome 0.22 0.49 0.17 0.21 0.20 0.31 0.17 0.24 1.00
Pregnancies
Glucose
BloodPressure
SkinThickness
Insulin
BMI
DiabetesPedigreeFunction
Age
Outcome
1.0
0.8
0.6
0.4
0.2
0.0


Pair Plot of Features
Pregnancies
Glucose
BloodPressure
SkinThickness
Insulin
BMI
DiabetesPedigreeFunction
Age
Outcome
0
1


Pregnancies by Outcome
Glucose by Outcome
BloodPressure by Outcome
SkinThickness by Outcome
Insulin by Outcome
BMI by Outcome
DiabetesPedigreeFunction by Outcome
Age by Outcome
Outcome


Count of Diabetes Outcomes
Count
500
400
300
200
100
0
0
1
Outcome (0: No Diabetes, 1: Diabetes)

Future Scope of Improvements

While the current model provides a solid foundation for diabetes prediction, there are several avenues for future improvement:

1. Incorporating Additional Health Indicators:

- **Genetic Factors:** Explore the role of genetic markers in diabetes susceptibility.
- **Lifestyle Factors:** Consider factors like diet, exercise habits, and stress levels.
- **Environmental Factors:** Account for environmental exposures that may influence diabetes risk.

2. Advanced Machine Learning Techniques:

- **Deep Learning:** Utilize deep neural networks for more complex pattern recognition and feature extraction.
- **Ensemble Methods:** Combine multiple models to improve overall performance and robustness.

3. Real-time Monitoring and Personalized Recommendations:

- **Mobile App:** Develop a mobile app that allows users to track their health metrics and receive personalized recommendations.
- **Continuous Learning:** Update the model with real-time data to adapt to changing patterns and individual needs.

4. Ethical Considerations and Bias Mitigation:

- **Fairness and Bias:** Ensure that the model is fair and unbiased, avoiding discriminatory outcomes.
- **Privacy and Security:** Protect user data and comply with relevant data privacy regulations.

5. Explainable AI:

- **Model Interpretability:** Develop techniques to explain the model's decision-making process, increasing user trust and understanding.

By pursuing these future directions, we can create a more comprehensive and effective diabetes detection system that empowers individuals to take control of their health and prevent the onset of diabetes.

Certificate

This is to certify that Mr. AQDAS SULTAN of Asansol Engineering College, registration number: 221080110375, has successfully completed a project on Diabetes Detection Using using Machine Learning under the guidance of Mr Dr. Arnab Chakraborty.

……..

Dr. Arnab Chakraborty

Certificate

This is to certify that Mr. SHIVAM PANDEY of Asansol Engineering College, registration number: 221080110440, has successfully completed a project on Diabetes Detection Using using Machine Learning under the guidance of Mr Dr. Arnab Chakraborty.

……..

Dr. Arnab Chakraborty

Certificate

This is to certify that Mr. ALTAMASH AHMED of Asansol Engineering College, registration number: 221080110363, has successfully completed a project on Diabetes Detection Using using Machine Learning under the guidance of Mr Dr. Arnab Chakraborty.

……..

Dr. Arnab Chakraborty

Certificate

This is to certify that Mr. ANAS TANWEER of Asansol Engineering College, registration number: 221080110366, has successfully completed a project on Diabetes Detection Using using Machine Learning under the guidance of Mr Dr. Arnab Chakraborty.

……..

Dr. Arnab Chakraborty