Assessment Report

on

“Diabetes Diagnosis”

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

Tanu Singh

By

Tanu Singh (202401100300262)

Under the supervision of

“Abhishek Shukla”

KIET Group of Institutions , Ghaziabad

Affiliated to

Dr. A.P.J. Abdul Kalam Technical University, Lucknow
(Formerly UPTU)

a) Introduction

Diabetes is a chronic medical condition that affects millions of people worldwide. Early detection and diagnosis of diabetes can significantly help in the management and prevention of complications associated with the disease.

This project aims to build a machine learning model that predicts the likelihood of diabetes based on several health features such as glucose levels, BMI, blood pressure, and age.

The dataset used in this project is the Pima Indians Diabetes Dataset, which contains health-related data for a group of women. The model will classify individuals as either diabetic (1) or non-diabetic (0) based on the input features.

b) Methodology

1. Data Collection

The dataset used in this project is the Pima Indians Diabetes Dataset, which includes the following features:

- **Pregnancies: Number of pregnancies a person has had.**
- **Glucose: Plasma glucose concentration after a 2-hour oral glucose tolerance test.**
- **BloodPressure: Diastolic blood pressure (mm Hg).**
- **SkinThickness: Triceps skinfold thickness (mm).**
- **Insulin: 2-hour serum insulin levels (mu U/ml).**
- **BMI: Body mass index (kg/m^2).**
- **DiabetesPedigreeFunction: A function that scores the likelihood of diabetes based on family history.**
- **Age: Age in years.**

- **Outcome: Whether the person has diabetes (1 = Yes, 0 = No).**

2. Data Preprocessing

- **Data Loading: The dataset is loaded from an Excel file (2. Diagnose Diabetes.xlsx).**
- **Data Splitting: The data is divided into features (X) and target (y). The target is the Outcome column, which indicates whether the person is diabetic or not.**
- **Scaling: The features are scaled using StandardScaler to ensure that all the features contribute equally to the model.**
- **Model Selection: The Random Forest Classifier algorithm is used for training the model, as it is an effective and powerful classifier for both small and large datasets.**

3. Model Training and Evaluation

- **Train-Test Split: The dataset is split into training (80%) and testing (20%) sets.**
- **Evaluation Metrics: The model's performance is evaluated using accuracy, confusion matrix, and a classification report (which includes precision, recall, and F1-score).**

C) CODE

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, accuracy_score,
confusion_matrix


# STEP 1: Load the Excel file
file_path = '/content/2. Diagnose Diabetes.xlsx'
```

df = pd.read_excel(file_path)

 # STEP 2: Show basic info

print("✅ Dataset Loaded. Shape:", df.shape)

print(df.head())

 # STEP 3: Split features and target

target_column = 'Outcome'

if target_column not in df.columns:

raise ValueError(f"Column '{target_column}' not found in the dataset.

Please update the target_column variable.")

X = df.drop(target_column, axis=1)

y = df[target_column]

 # STEP 4: Train/Test Split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,

random_state=42)

 # STEP 5: Scale features

scaler = StandardScaler()

X_train_scaled = scaler.fit_transform(X_train)

X_test_scaled = scaler.transform(X_test)

 # STEP 6: Train the model

model = RandomForestClassifier(random_state=42)

model.fit(X_train_scaled, y_train)

 # STEP 7: Evaluate the model

y_pred = model.predict(X_test_scaled)

print("\n📊 Model Evaluation:")

print("Accuracy:", accuracy_score(y_test, y_pred))

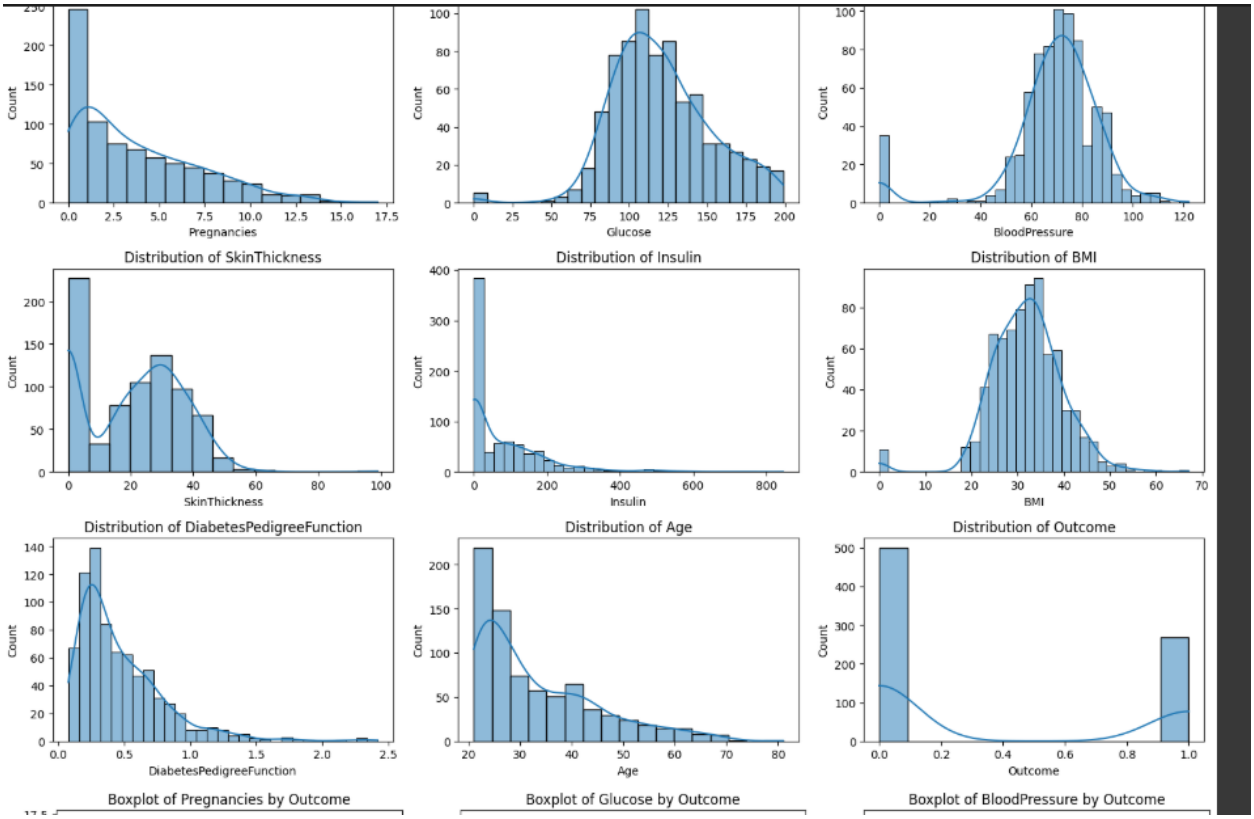
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))

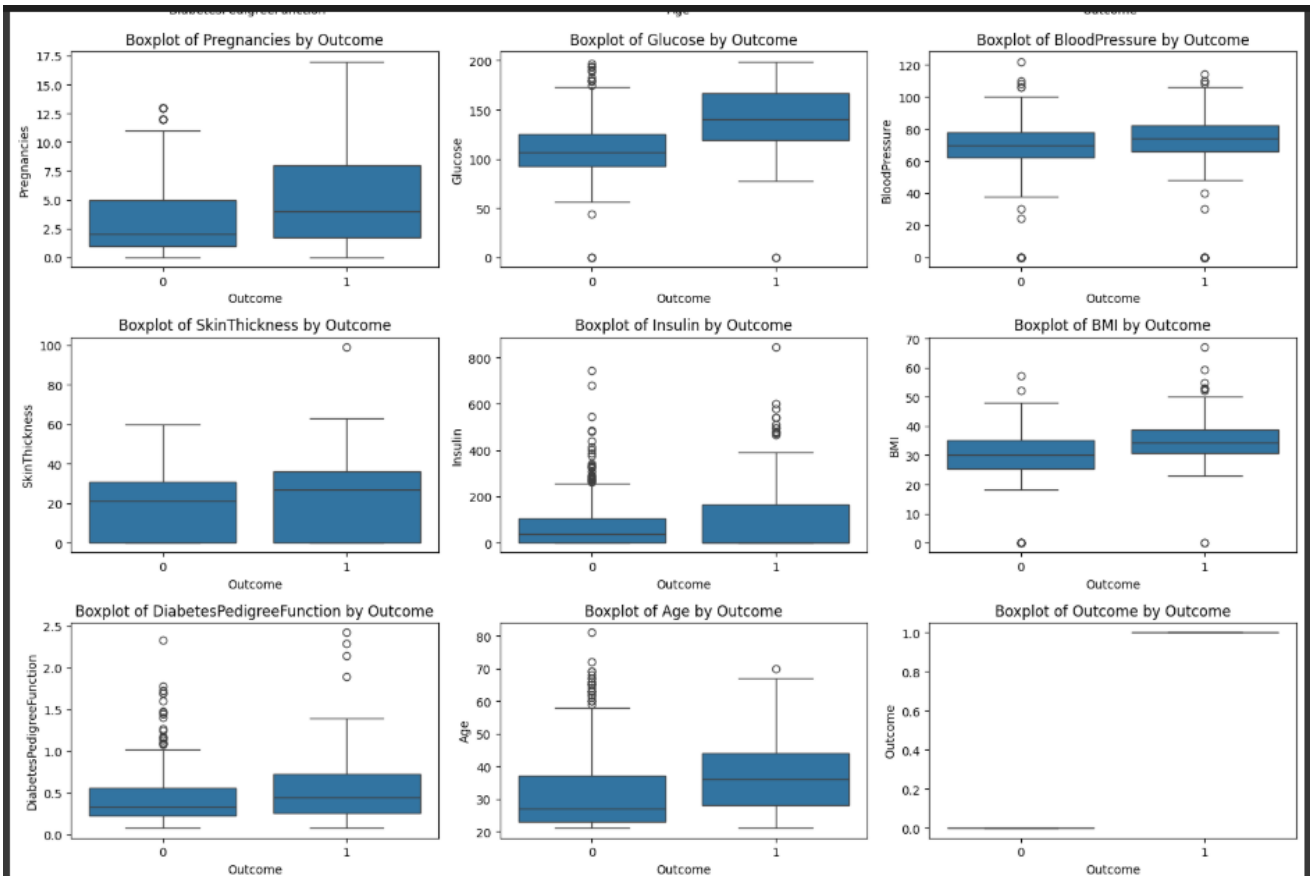
print("Classification Report:\n", classification_report(y_test, y_pred))

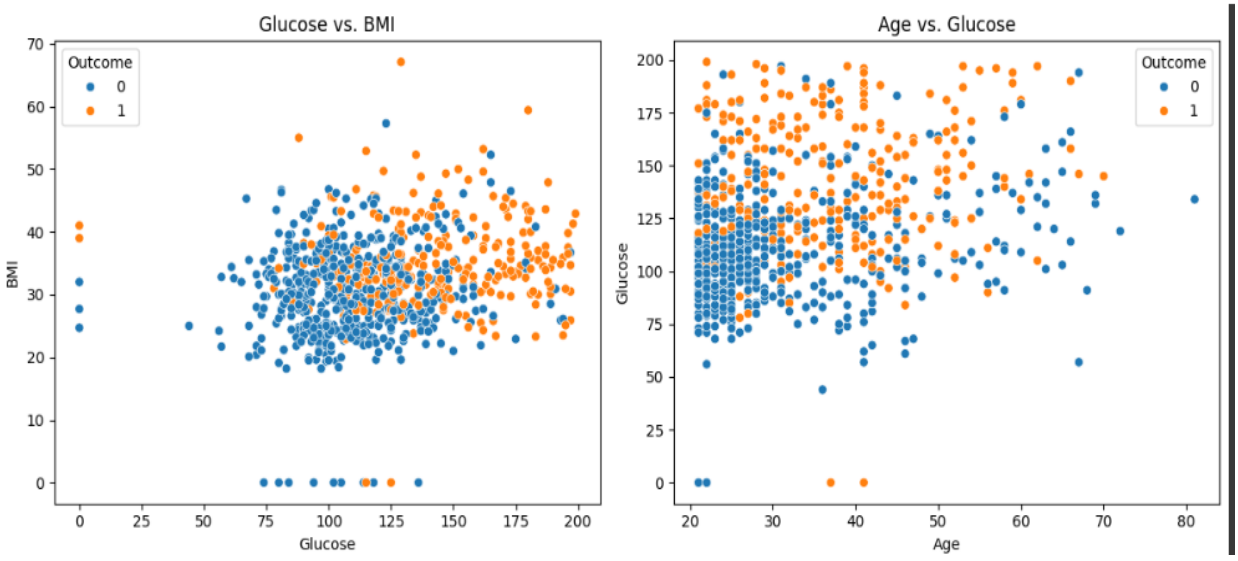
D) OUTPUT

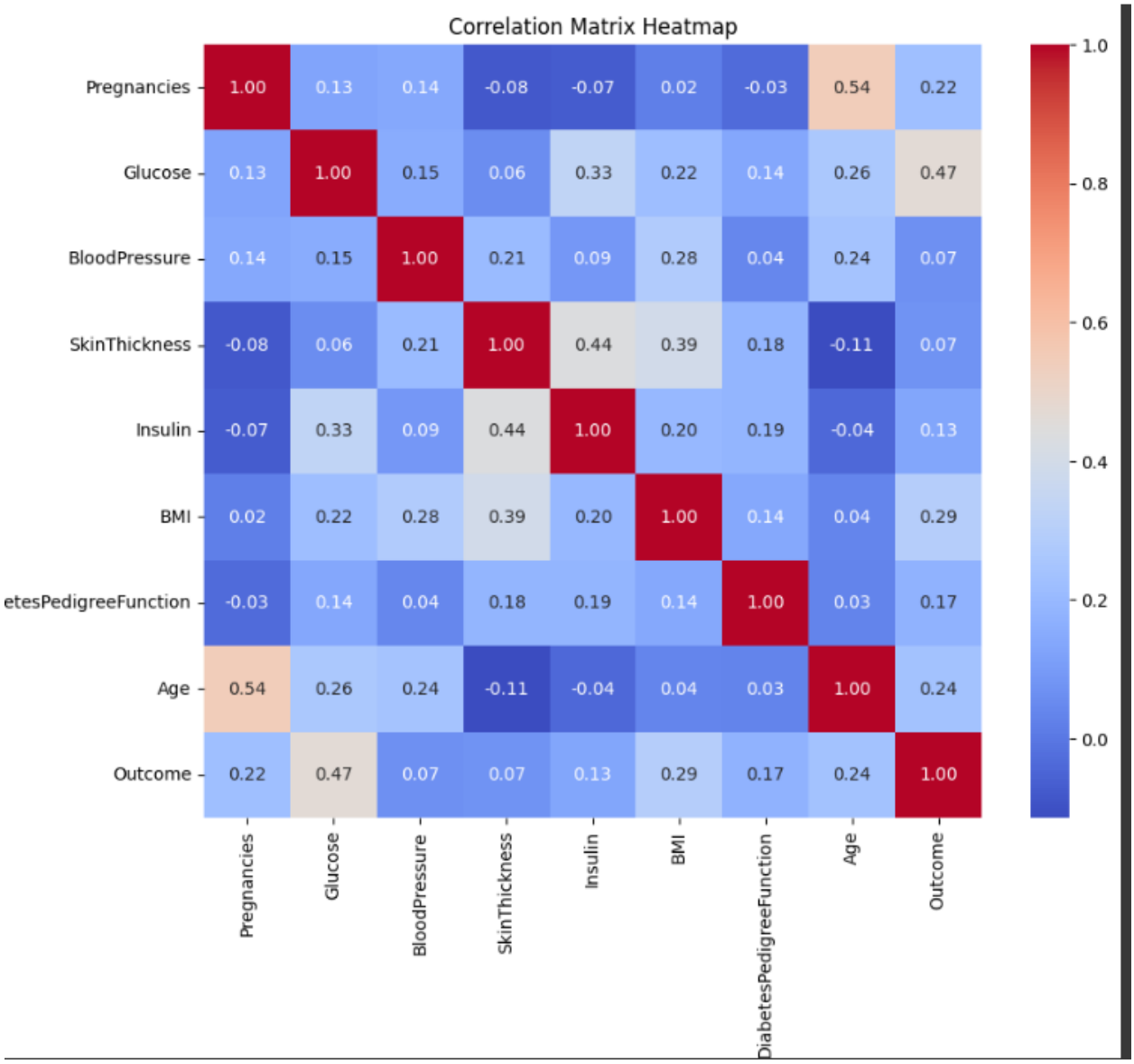
```
Summary Statistics (including zeros):
          Glucose  BloodPressure  SkinThickness     Insulin         BMI  \
count  768.000000     768.000000     768.000000  768.000000  768.000000
mean   120.894531      69.105469      20.536458   79.799479   31.992578
std     31.972618      19.355807      15.952218  115.244002    7.884160
min      0.000000       0.000000       0.000000    0.000000    0.000000
25%     99.000000      62.000000       0.000000    0.000000   27.300000
50%    117.000000      72.000000      23.000000   30.500000   32.000000
75%    140.250000      80.000000      32.000000  127.250000   36.600000
max    199.000000     122.000000      99.000000  846.000000   67.100000

       DiabetesPedigreeFunction         Age
count                768.000000  768.000000
mean                   0.471876   33.240885
std                    0.331329   11.760232
min                    0.078000   21.000000
25%                    0.243750   24.000000
50%                    0.372500   29.000000
75%                    0.626250   41.000000
max                    2.420000   81.000000

Summary Statistics (excluding zeros):
          Glucose  BloodPressure  SkinThickness     Insulin         BMI  \
count  763.000000     733.000000     541.000000  394.000000  757.000000
mean   121.686763      72.405184      29.153420  155.548223   32.457464
std     30.535641      12.382158      10.476982  118.775855    6.924988
min     44.000000      24.000000       7.000000   14.000000   18.200000
25%     99.000000      64.000000      22.000000   76.250000   27.500000
50%    117.000000      72.000000      29.000000  125.000000   32.300000
75%    141.000000      80.000000      36.000000  190.000000   36.600000
max    199.000000     122.000000      99.000000  846.000000   67.100000

       DiabetesPedigreeFunction         Age
count                768.000000  768.000000
mean                   0.471876   33.240885
std                    0.331329   11.760232
min                    0.078000   21.000000
```

```
50%     117.000000      72.000000      29.000000   125.000000    32.300000
75%     141.000000      80.000000      36.000000   190.000000    36.600000
max     199.000000     122.000000      99.000000   846.000000    67.100000

       DiabetesPedigreeFunction         Age
count                768.000000  768.000000
mean                   0.471876   33.240885
std                    0.331329   11.760232
min                    0.078000   21.000000
25%                    0.243750   24.000000
50%                    0.372500   29.000000
75%                    0.626250   41.000000
max                    2.420000   81.000000

Outcome Counts:
 Outcome
0    500
1    268
Name: count, dtype: int64

Outcome Proportions:
 Outcome
0    0.651042
1    0.348958
Name: proportion, dtype: float64

Correlation with Outcome:
 Glucose                    0.466581
BloodPressure              0.065068
SkinThickness              0.074752
Insulin                    0.130548
BMI                        0.292695
DiabetesPedigreeFunction   0.173844
Age                        0.238356
Name: Outcome, dtype: float64
```




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


Glucose vs. BMI
Outcome
0
1
BMI
Glucose
Age vs. Glucose
Outcome
0
1
Glucose
Age



References and Credits

- **Dataset Source**: Pima Indians Diabetes Dataset - UCI Machine Learning Repository
- **Tools Used**:
    - Pandas for data manipulation
    - Scikit-learn for model building and evaluation
    - RandomForestClassifier for classification
    - StandardScaler for normalization