Assessment Report

on

“Diagnose Diabetes”

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

CSE-AI

By

Vaishnavi Mishra

(202401100300272)

(52, CSE-AI D)

Under the supervision of

Sir Abhishek Shukla

KIET Group of Institutions, Ghaziabad

April, 2025

Diagnose Diabetes Report

Introduction

Diabetes represents a growing health concern on a global scale. Timely diagnosis is essential for effective intervention and treatment. This project seeks to develop a machine learning model that predicts the likelihood of an individual having diabetes based on their medical records.

We will employ the PIMA Indian Diabetes Dataset, which encompasses various medical attributes, including glucose levels, blood pressure, body mass index (BMI), and a corresponding label indicating the presence of diabetes.

Objective

The goal is to classify individuals as either diabetic or non-diabetic based on their medical data.

Methodology

- Data Loading: The diabetes dataset, stored in a CSV file format, was imported utilizing the pandas library.
- Data Splitting: The dataset was divided into input features (denoted as X) and target labels (denoted as y).
- Train-Test Split: Eighty percent of the dataset was allocated for training purposes, while twenty percent was reserved for testing.
- Model Training: A Random Forest Classifier was developed and trained on the prepared data.
- Prediction: The model was employed to predict outcomes for the test dataset.
- Evaluation: The performance of the model was assessed using a Confusion Matrix along with Accuracy,Precision, and Recall metrics.
- Visualization: Seaborn was utilized to create a heatmap representing the Confusion Matrix for enhanced clarity and interpretation.

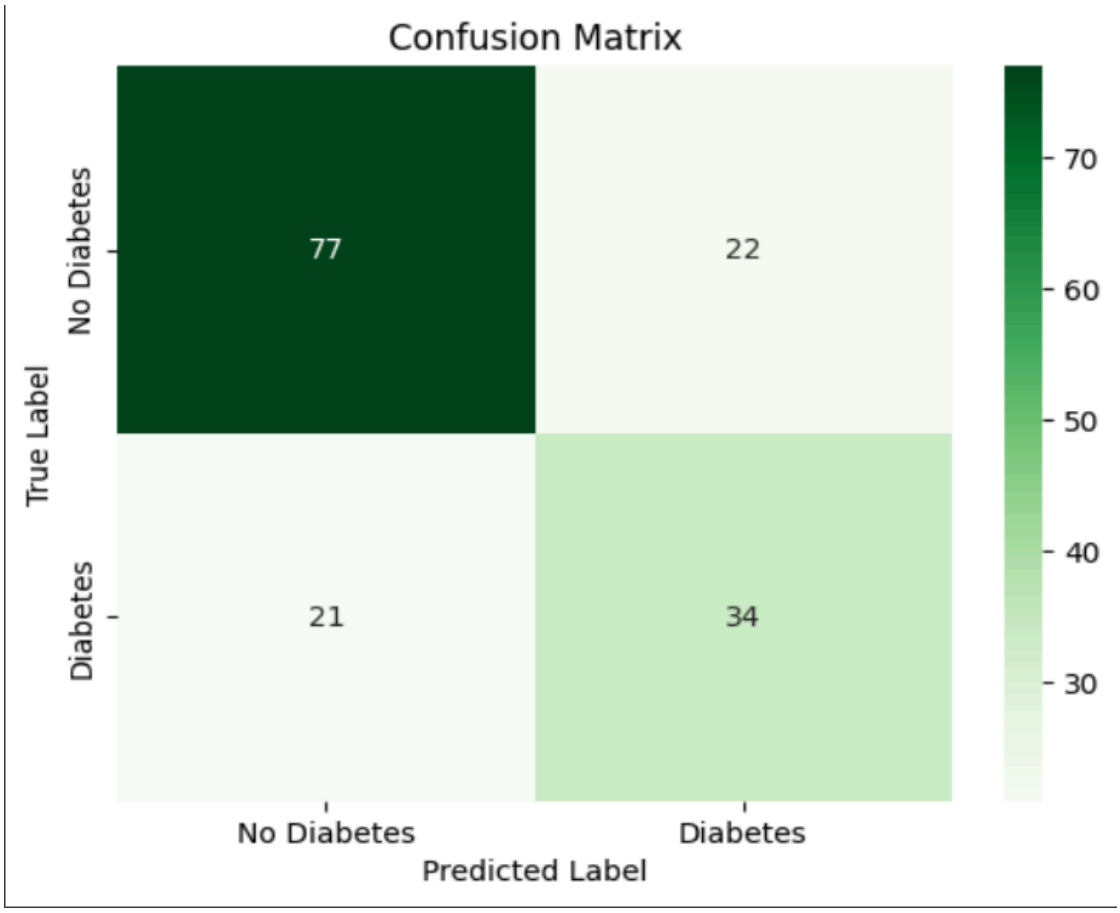
Code

The diagnosis was conducted using Python with the Pandas, Matplotlib, and Seaborn libraries.

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import confusion_matrix, accuracy_score, precision_score, recall_score

# Load the dataset containing medical records
df = pd.read_csv("2. Diagnose Diabetes.csv")

# Separate features and target variable
X = df.drop("Outcome", axis=1)  # Features
y = df["Outcome"]               # Target (0 = No Diabetes, 1 = Diabetes)

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

# Initialize and train a Random Forest Classifier
clf = RandomForestClassifier()
clf.fit(X_train, y_train)

# Predict outcomes on the test set
y_pred = clf.predict(X_test)

# Create and display the confusion matrix as a heatmap
cm = confusion_matrix(y_test, y_pred)
sns.heatmap(cm, annot=True, fmt='d', cmap='Greens',
            xticklabels=["No Diabetes", "Diabetes"],
            yticklabels=["No Diabetes", "Diabetes"])
plt.xlabel("Predicted Label")
plt.ylabel("True Label")
plt.title("Confusion Matrix")
plt.show()

# Calculate and print evaluation metrics
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred)
recall = recall_score(y_test, y_pred)

print(f"Accuracy: {accuracy:.2f}")
print(f"Precision: {precision:.2f}")
print(f"Recall: {recall:.2f}")
```

Output / Results

- Expected output:
  Accuracy: 0.72
  Precision: 0.61
  Recall: 0.62
- The confusion matrix heatmap visualizes the model's performance by comparing predicted values to actual values.



References/Credits

- Dataset: Diabetes Classification: Pima Indians Diabetes Dataset
  Source: UCI Machine Learning Repository
  (Available at: https://archive.ics.uci.edu/ml/datasets/Pima+Indians+Diabetes)
- Libraries Used:
  - Pandas: https://pandas.pydata.org
  - Seaborn: https://seaborn.pydata.org
  - Scikit-learn: https://scikit-learn.org
- IDE Used: Google Colab