Green University of Bangladesh
Department of Computer Science and Engineering (CSE)
Faculty of Sciences and Engineering
Semester: (Spring, Year:2025), B.Sc. in CSE (Day)

Report Title: Diabetes Prediction using Linear Regression.

CourseTitle: Machine Learning Lab
Course Code: CSE-412 Section:221-D13

Student Details

Name	ID
Abu Bakkar Siddik	221902265

Course Teacher's Name : Md. Sabbir Hosen Mamun
Submission Date: 10.07.2025

[For Teachers use only: Don't Write Anything inside this box]

Marks: ……	**Signature:.....**
Comments:.....	**Date:.....**

TITLE OF THE LAB REPORT

Diabetes Prediction using Linear Regression

2. OBJECTIVES

The objective of learning linear regression is to develop an understanding of a fundamental statistical and machine learning technique used for predictive modeling and understanding the relationships between variables. Linear regression is a simple yet powerful method used in various fields, including statistics, economics, finance, and machine learning. Here are the primary objectives of learning linear regression:

• Understanding the Basics: Learn the fundamental concepts of linear regression, including the terminology (dependent and independent variables, coefficients, intercept, etc.) and the mathematical representation of linear regression models.

• Model Building: Learn how to build a linear regression model by selecting appropriate independent variables (features) and estimating coefficients that best fit the data.

• Interpretation: Develop the ability to interpret the coefficients of a linear regression model. Understand how changes in the independent variables affect the dependent variable.

IMPLEMENTATION

Code:

```
import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

%matplotlib inline
```

```
!git clone
https://github.com/ShawonTech/Machine-Learning-Lab/tree/main



dataset = pd.read_csv('/content/drive/MyDrive/diabetes.csv')

dataset.head()


dataset.plot(x='Glucose', y='BloodPressure', style='o')

plt.title('Glucose vs BloodPressure')

plt.xlabel('Glucose')

plt.ylabel('BloodPressure')

plt.show()


cols_with_zero_invalid = ['Glucose', 'BloodPressure',
'SkinThickness', 'Insulin', 'BMI']

dataset[cols_with_zero_invalid] =
dataset[cols_with_zero_invalid].replace(0, np.nan)

dataset[cols_with_zero_invalid] =
dataset[cols_with_zero_invalid].fillna(dataset[cols_with_zero_invalid
].mean())

dataset.loc[0, 'Glucose'] = dataset['Glucose'].max()


dataset.loc[0, 'Glucose'] = dataset['Glucose'].max()



min_age = dataset['Age'].min()

min_glucose = dataset['Glucose'].min()

dataset.loc[dataset['Age'] == min_age, 'Glucose'] = min_glucose
```

```
X = dataset.drop('Outcome', axis=1).values

y = dataset['Outcome'].values


from sklearn.linear_model import LinearRegression

regressor = LinearRegression ()

regressor.fit(X_train , y_train)


y_pred_continuous = regressor.predict(X_test)

y_pred = np.round(y_pred_continuous).astype(int)

from sklearn import metrics

print('Mean Squared Error:', metrics.mean_squared_error(y_test ,
y_pred))


from sklearn.metrics import accuracy_score, confusion_matrix,
precision_score, recall_score, f1_score


accuracy = accuracy_score(y_test, y_pred)

conf_matrix = confusion_matrix(y_test, y_pred)

precision = precision_score(y_test, y_pred)

recall = recall_score(y_test, y_pred)

f1 = f1_score(y_test, y_pred)


print(f'Accuracy: {accuracy:.4f}')

print('Confusion Matrix:')

print(conf_matrix)

print(f'Precision: {precision:.4f}')

print(f'Recall: {recall:.4f}')

print(f'F1 Score: {f1:.4f}')
```

OUTPUT

labreport.ipynb

File Edit View Insert Runtime Tools Help

Share Gemini

Commands + Code + Text Run all

RAM Disk

Files

drive

sample_data

Disk 70.03 GB available

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

```
[3] dataset = pd.read_csv('/content/drive/MyDrive/diabetes.csv')
    dataset.head()
```

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

Next steps: Generate code with dataset | View recommended plots | New interactive sheet

```
[4] dataset.plot(x='Glucose', y='BloodPressure', style='o')
    plt.title('Glucose vs BloodPressure')
    plt.xlabel('Glucose')
    plt.ylabel('BloodPressure')
    plt.show()
```

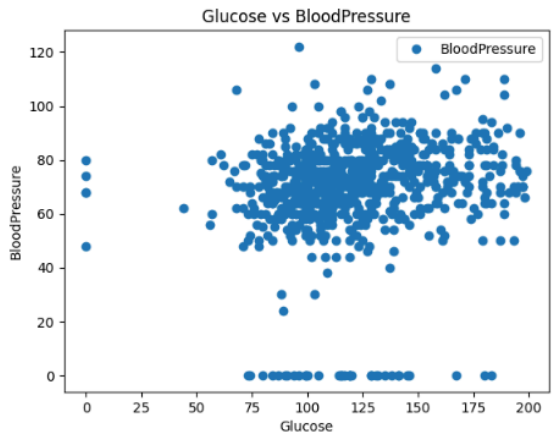
Variables Terminal 8:18 AM Python 3

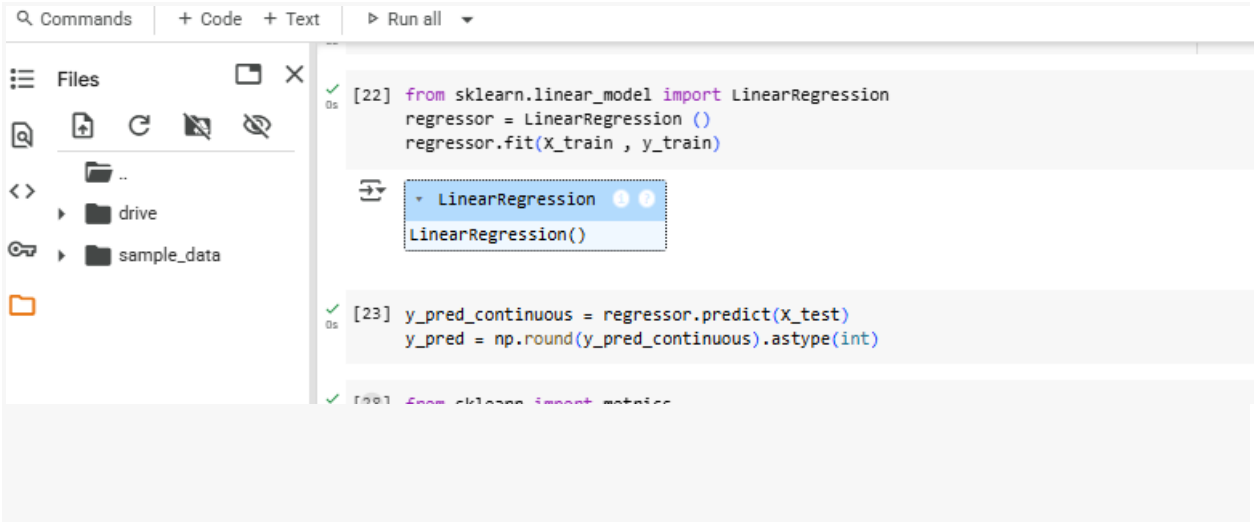
Files

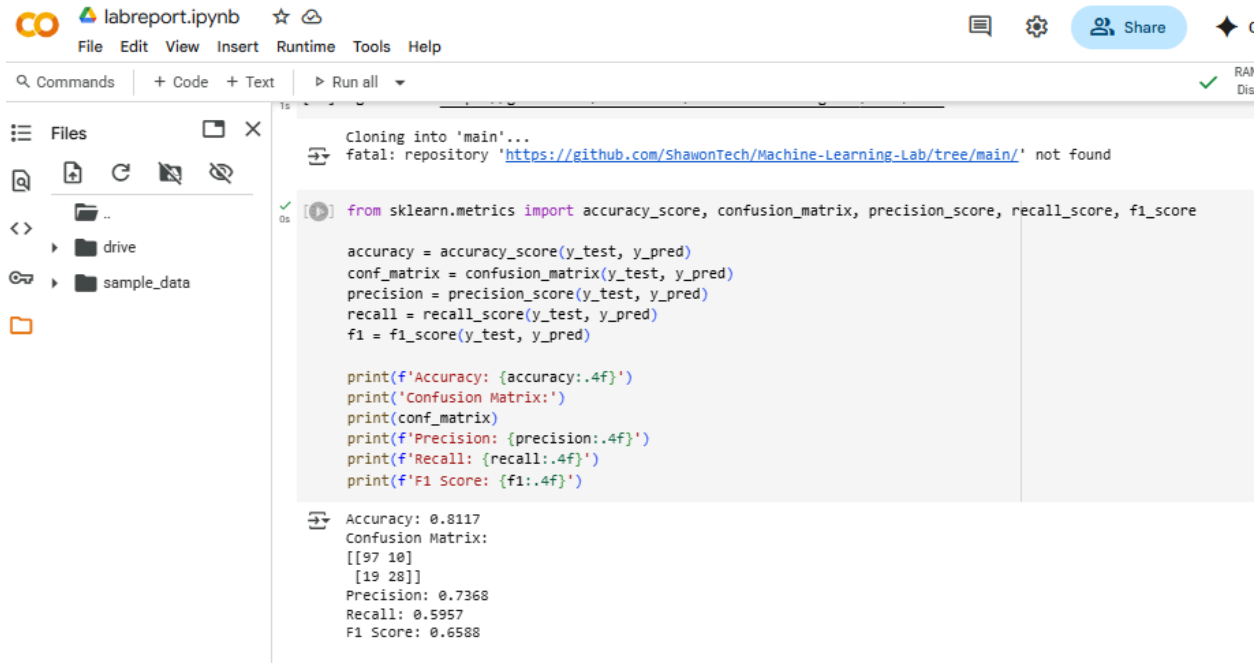
drive

sample_data

```
plt.ylabel('BloodPressure')
plt.show()
```







Analysis and Discussion

Linear Regression can be used for binary prediction in a very basic way, but it is not ideal for classification problems.

For a real-world diabetes prediction model, it's better to use Logistic Regression or other classification algorithms that are built for this purpose

.
Despite limitations, the experiment provides valuable insights into model behavior and importance of preprocessing.

GitHub Link:
https://github.com/ShawonTech/Machine-Learning-Lab/tree/main