Worksheet 1.1

Student Name: Ayush Yadav **UID: 25IMC13004**
Branch: MCA (AI & ML) **Section/Group: MAM-1 (A)**
Semester: II **Date of Performance: 10/01/26**
Subject Name: Machine Learning Lab **Subject Code: 25CAP-672**

1. Aim/Overview of the practical:

Experiment-1:

1. Implementation of Python Basic Libraries such as
   a) Math
   b) Numpy
   c) Matplotlib
   d) Seaborn
   e) Scipy

2. Creation and loading different Datasets in Python using Jupyter Notebook

3. Write a programs to implement Linear regression on Jupyter notebook
   a) Download different datasets from Kaggle or UCI ML repository
   b) Import all the necessary modules for the datasets.
   c) Find out the best fit line along with the MSE, RMSE

2. Coding:

PART-1

```
import math
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

#MATH
print("Square root: ", math.sqrt(25))
print("Factorial: ", math.factorial(5))
print("Power: ", math.pow(2,3))
print("Ceiling: ", math.ceil(4.3))
print("Floor: ", math.floor(4.7))
print("Pie: ", math.pi)
print("Euler's Number: ", math.e)

#NUMPY
arr = np.array([1,2,3,4,5])

print("Array: ", arr)
print("Mean: ", np.mean(arr))
print("Sum: ", np.sum(arr))
print("Max: ", np.max(arr))

matrix = np.array([[2,4], [6,8]])
print("Matrix:\n", matrix)
print("Transpose:\n", matrix.T)

#MATPLOT LIBRARY
x = [1,2,3,4,5]
y= [10,20,30,40,50]
plt.plot(x,y)
plt.xlabel("X Axis")
plt.ylabel("Y Axis")
```

```
plt.title("Line Graph")
plt.show()

#SEABORN
data = [10,20,30,40,50]
sns.histplot(data)
plt.title("Seaborn Histogram")
plt.show()

#SCIPY LIBRARY
from scipy import stats
data = [10,20,30,40,50]
print("Mean: ", stats.tmean(data))
print("Median:  ", stats.scoreatpercentile(data,50))
print("Mode: ", stats.mode(data))
```

OUTPUT:-

```
[Running] python -u "e:\CLASS_CODING\ML\tempCodeRunnerFile.py"
Square root:  5.0
Factorial:  120
Power:  8.0
Ceiling:  5
Floor:  4
Pie:  3.141592653589793
Euler's Number:  2.718281828459045
Array:  [1 2 3 4 5]
Mean:  3.0
Sum:  15
Max:  5
Matrix:
 [[2 4]
 [6 8]]
Transpose:
 [[2 6]
 [4 8]]
```

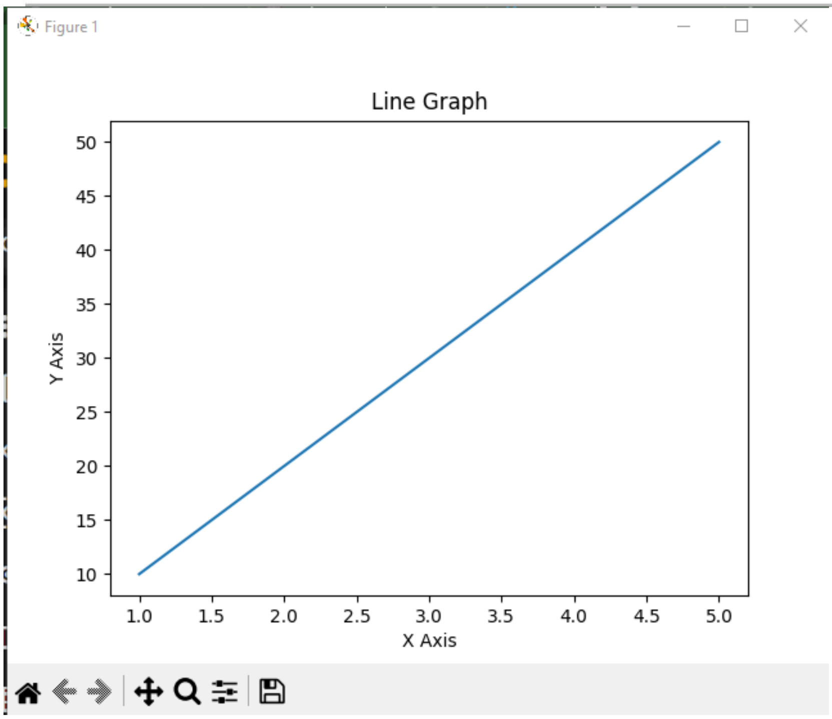


Figure 1
Line Graph
Y Axis
X Axis
50
45
40
35
30
25
20
15
10
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0

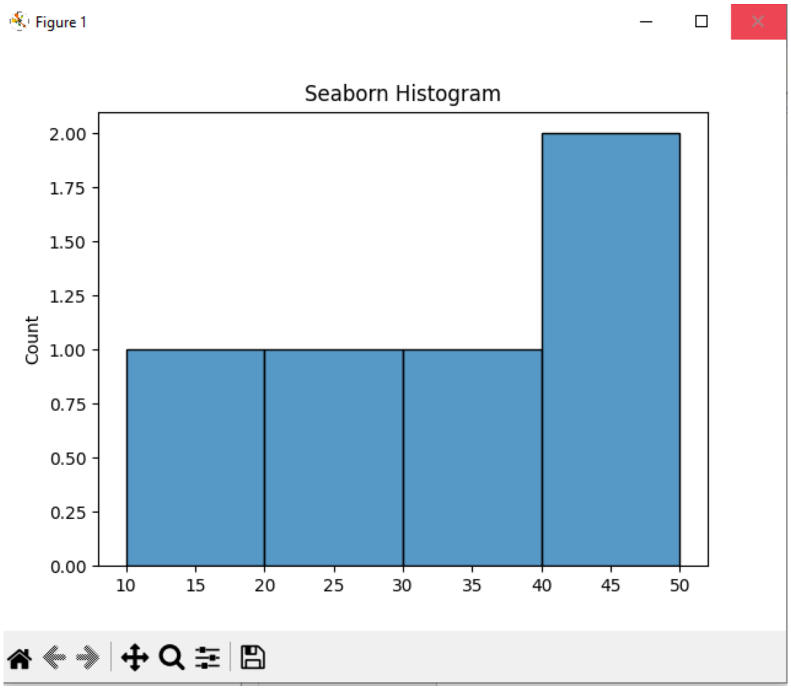


Figure 1
Seaborn Histogram
Count
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
10
15
20
25
30
35
40
45
50

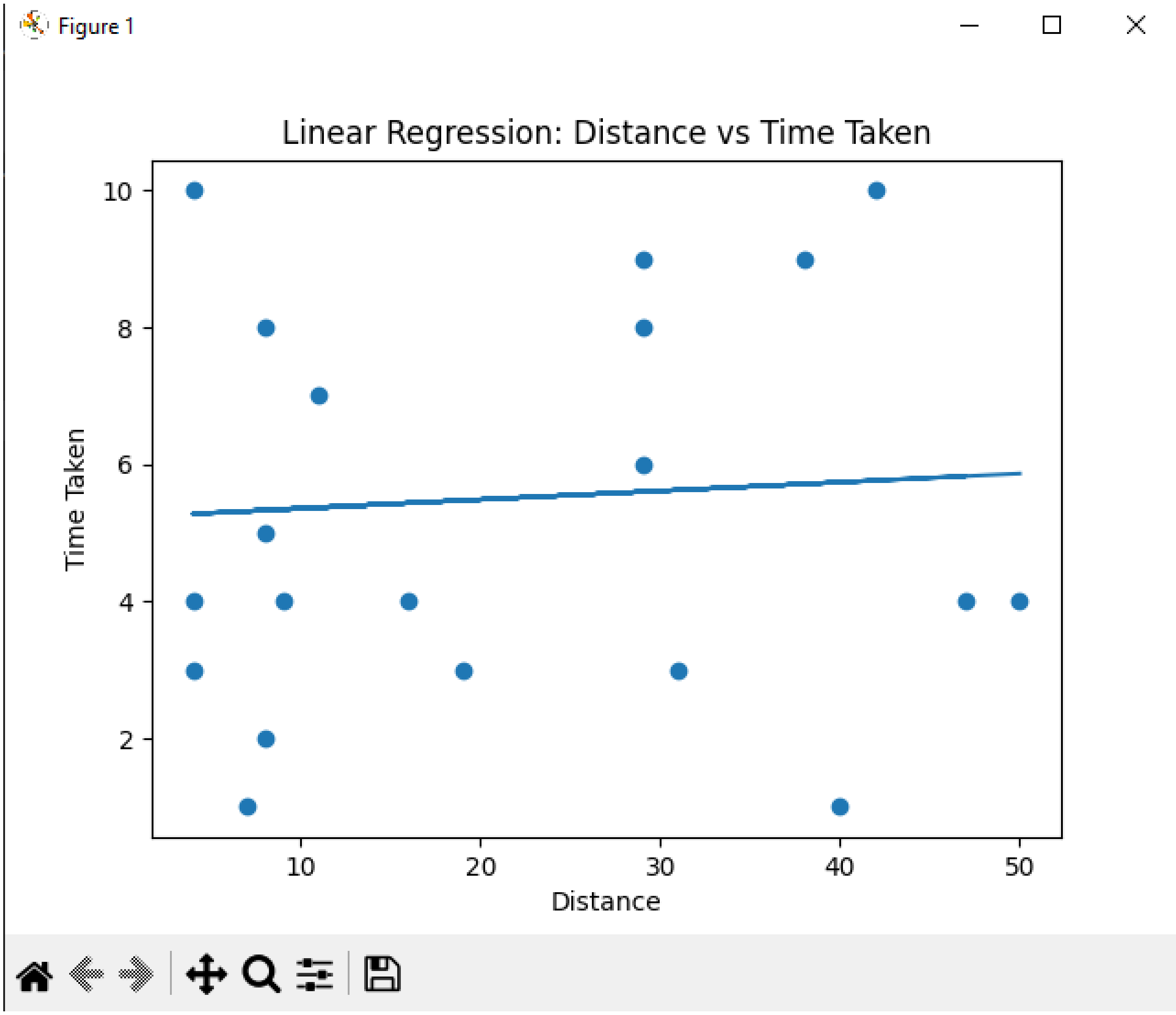
PART-2

```
import os
import numpy as np
import pandas as pd
import matplotlib as plt

data = {
    'Name': ['Ayush', 'Yadav', 'Rao', 'Sahab'],
    'Age' : [20,24,23,21],
    'City': ['Delhi', 'Haryana', 'Chandigarh', 'Punjab']
}
df_show = pd.DataFrame(data)
print(df_show)

df_csv = pd.read_csv('E:\\Users\\Dell\\Downloads\\archive\\Sensitivity_Soil_Nutrient_Pools.csv')
print(df_csv)

df_excel = pd.read_excel('D:\\AYUSH\\RANDOM SHIT\\student_data.xlsx', engine='openpyxl')
print(df_excel)
```

```
[Running] python -u "e:\CLASS_CODING\ML\W1.2.py"
    Name  Age        City
0  Ayush   20       Delhi
1  Yadav   24     Haryana
2    Rao   23  Chandigarh
3  Sahab   21      Punjab
            SampleID site  block  ...  year      type       faith_pd
0      GMDR-FK-2018-4   FK      1  ...  2018  bacteria  3.912.908.739
1     GMDR-TB-2018-45   TB      3  ...  2018  bacteria  3.298.556.468
2      GMDR-TB-2018-8   TB      1  ...  2018  bacteria  3.332.257.877
3     GMDR-TB-2018-31   TB      2  ...  2018  bacteria  3.806.342.641
4     GMDR-TB-2018-23   TB      2  ...  2018  bacteria  4.135.953.774
...               ...  ...    ...  ...   ...       ...            ...
1001  GMDR-FK-2022-23   FK      2  ...  2022     fungi  2.229.139.198
1002  GMDR-FK-2022-20   FK      2  ...  2022     fungi  1.572.091.467
1003  GMDR-TB-2022-49   TB      3  ...  2022     fungi  3.400.527.221
1004  GMDR-FK-2022-52   FK      3  ...  2022     fungi  1.679.936.128
1005  GMDR-TB-2022-50   TB      3  ...  2022     fungi  3.077.840.854

[1006 rows x 11 columns]
         UID         Name  ...              Subjects                Marks
0  25MCI10003      Lakshit  ...  ADBMS;AI;DAA;Python  80.0;85.0;82.0;90.0
1  25MCI10004        Aryan  ...  ADBMS;AI;DAA;Python  80.0;85.0;75.0;88.0
2  25MCI10002         Amit  ...  ADBMS;AI;DAA;Python  82.0;89.0;80.0;90.0
3  25MCI10001  Ayush Yadav  ...  ADBMS;AI;DAA;Python  94.0;96.0;92.0;98.0

[4 rows x 6 columns]
```

PART-3

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
from sklearn.linear_model import LinearRegression

df = pd.read_excel('E:\\Users\\Dell\\Documents\\w1.xlsx')
X = df[['Distance']]
y = df['Time_Taken']
X_train, X_test, y_train, y_test = train_test_split(
   X, y, test_size=0.2, random_state=42
)
model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
plt.scatter(X_test, y_test)
plt.plot(X_test, y_pred)
plt.xlabel("Distance")
plt.ylabel("Time Taken")
plt.title("Linear Regression: Distance vs Time Taken")
plt.show()
mse = mean_squared_error(y_test, y_pred)
rmse = np.sqrt(mse)
print("Mean Squared Error (MSE):", mse)
print("Root Mean Squared Error (RMSE):", rmse)
```




```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS

[Running] python -u "e:\CLASS_CODING\ML\21.3.py"
Mean Squared Error (MSE): 8.056271499025042
Root Mean Squared Error (RMSE): 2.8383571831298897

[Done] exited with code=0 in 35.761 seconds
```

3. **Learning outcomes (What I have learnt):**

- Understand and use basic Python libraries such as **Math, NumPy, Matplotlib, and Seaborn** for numerical computation and visualization.
- Load, create, and explore datasets using **Pandas** in Jupyter Notebook.
- Understand the concept of **Linear Regression** and its role in predicting continuous values.
- Implement a **Linear Regression model** using real-world datasets.
- Visualize the **best fit line** to analyze the relationship between independent and dependent variables.
- Evaluate model performance using **Mean Squared Error (MSE)** and **Root Mean Squared Error (RMSE)**.
- Interpret regression results and understand how prediction errors reflect model accuracy.