# **Practical – 6**

**Aim:** **Write a program to implement K-Nearest Neighbors.**

* Code:

import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

import seaborn as sns

from sklearn.model\_selection import train\_test\_split

from sklearn.neighbors import KNeighborsRegressor

from sklearn.metrics import mean\_squared\_error, r2\_score

data\_set = pd.read\_csv('/content/drive/MyDrive/temp/practical\_4\_2.csv')

print(data\_set)

X = data\_set[['Match\_Duration', 'Loot\_Collected', 'Enemies\_Defeated']]

y = data\_set['Player\_Score']

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

knn\_model = KNeighborsRegressor(n\_neighbors=3)

knn\_model.fit(X\_train, y\_train)

y\_pred = knn\_model.predict(X\_test)

print("Mean Squared Error:", mean\_squared\_error(y\_test, y\_pred))

print("R² Score:", r2\_score(y\_test, y\_pred))

plt.figure(figsize=(10, 6))

sns.scatterplot(x=y\_train, y=knn\_model.predict(X\_train), color='green', label='Training Data')

plt.xlabel('Actual Player Score')

plt.ylabel('Predicted Player Score')

plt.title('KNN: Training Data - Actual vs Predicted')

plt.legend()

plt.show()

plt.figure(figsize=(10, 6))

sns.scatterplot(x=y\_test, y=y\_pred, color='purple', label='Test Data')

plt.xlabel('Actual Player Score')

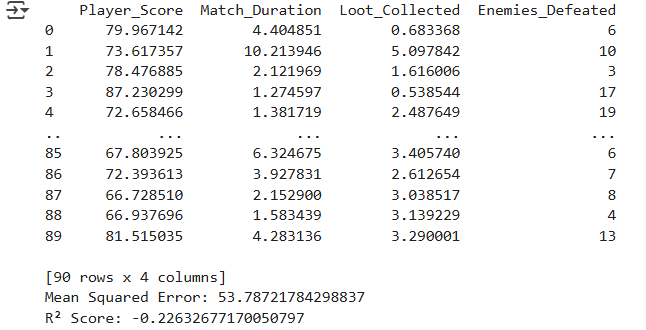
plt.ylabel('Predicted Player Score')

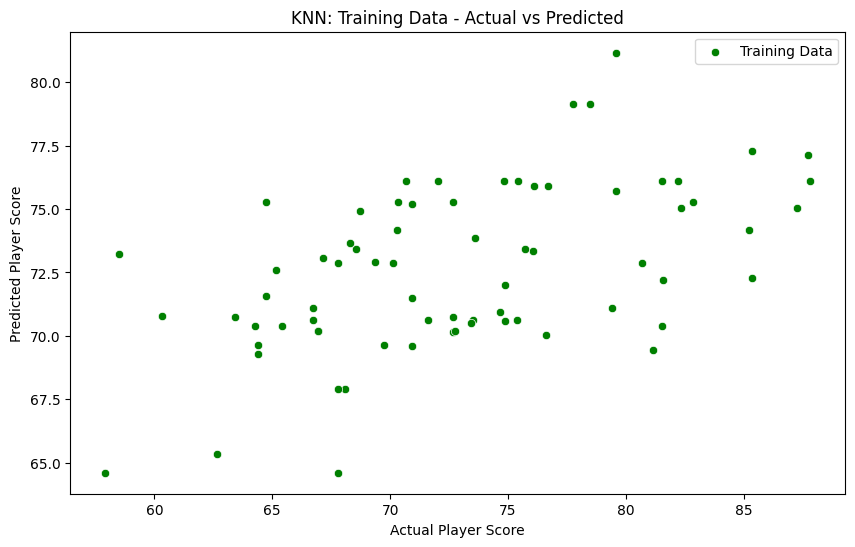
plt.title('KNN: Test Data - Actual vs Predicted')

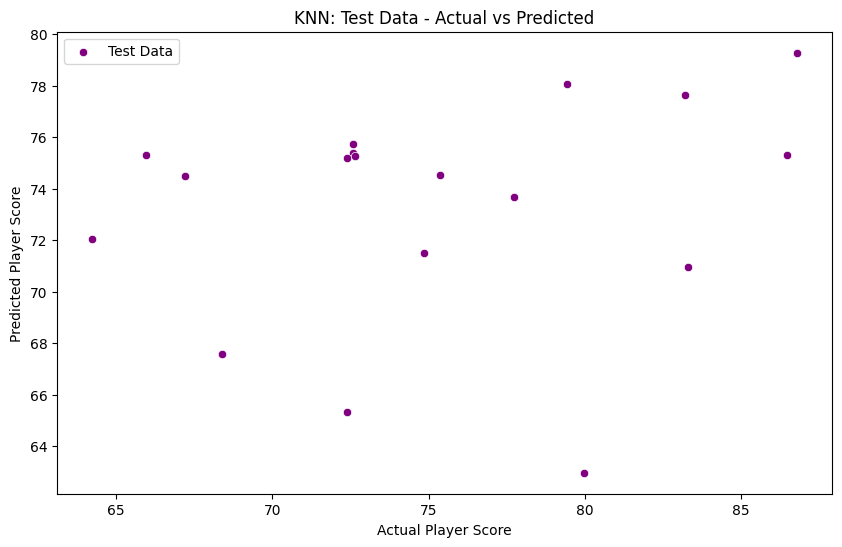
plt.legend()

plt.show()

* Output







Faculty Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_