**Assignment-1 Report**

**Course:** CS502 - Advanced Pattern Recognition  
**Submitted by:** Vinay Sadupati (2201ai44)  
**Submission Date:** 19 Sept 2025

**Title**

Predicting Student Final Grades using Linear Regression**Introduction**

In this assignment, we apply a Linear Regression model to the Student Performance Dataset (UCI Machine Learning Repository). The dataset contains demographic, social, and academic information of students, including their first period grade (G1), second period grade (G2), and final grade (G3).

The goal is to predict the final grade (G3) of students using selected features such as study time, absences, failures, and past grades.**Dataset Description**

* **Source:** Student Performance Dataset ([UCIstudentPerformance](https://www.kaggle.com/datasets/robertgarcia/uclstudentperformance))
* **Data Files Used:** student-mat.csv
* **Target Variable:**
  + G3 → Final Grade (0–20)
* **Features Used:**
  + studytime → Weekly study time (1–4)
  + failures → Number of past class failures (0–3)
  + absences → Number of school absences
  + G1 → First period grade (0–20)
  + G2 → Second period grade (0–20)

**Methodology**

1. **Data Loading**: Imported dataset using pandas.
2. **Feature Selection**: Selected only numeric features relevant to final grade prediction.
3. **Train-Test Split**: Split data into 80% training and 20% testing.
4. **Model Training**: Applied Linear Regression from scikit-learn.
5. **Model Evaluation**: Measured performance using R² score and Mean Squared Error (MSE).
6. **Visualization**: Compared actual vs predicted grades using scatter plot.

**Code Implementation**

import pandas as pd

import matplotlib.pyplot as plt

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

from sklearn.linear\_model import LinearRegression

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

# Load dataset

data = pd.read\_csv("student-mat.csv", sep=";")

# Features and target

X = data[["studytime", "failures", "absences", "G1", "G2"]]

y = data["G3"]

# Train-test split

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

# Linear Regression model

model = LinearRegression()

model.fit(X\_train, y\_train)

# Predictions

y\_pred = model.predict(X\_test)

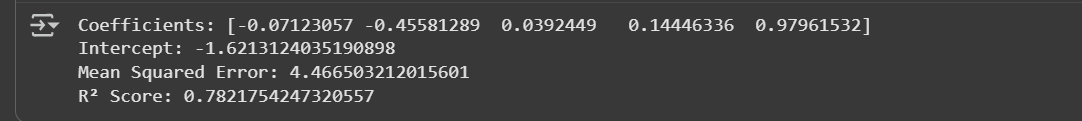
# Evaluation metrics

print("Coefficients:", model.coef\_)

print("Intercept:", model.intercept\_)

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

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



# Visualization

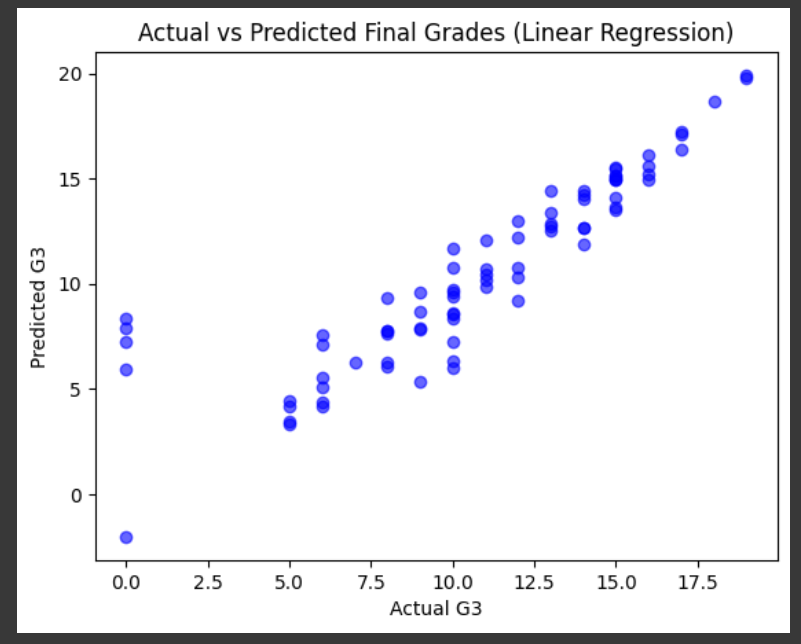
plt.scatter(y\_test, y\_pred, color="blue", alpha=0.6)

plt.xlabel("Actual G3")

plt.ylabel("Predicted G3")

plt.title("Actual vs Predicted Final Grades")

plt.show()



**Results**

* **Coefficients:**
  + Study Time → -0.0712
  + Failures → -0.4558
  + Absences → +0.0392
  + G1 → +0.1445
  + G2 → +0.9796
* **Intercept:** -1.6213
* **Mean Squared Error (MSE):** 4.4665
* **R² Score:** 0.7822