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)
- **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

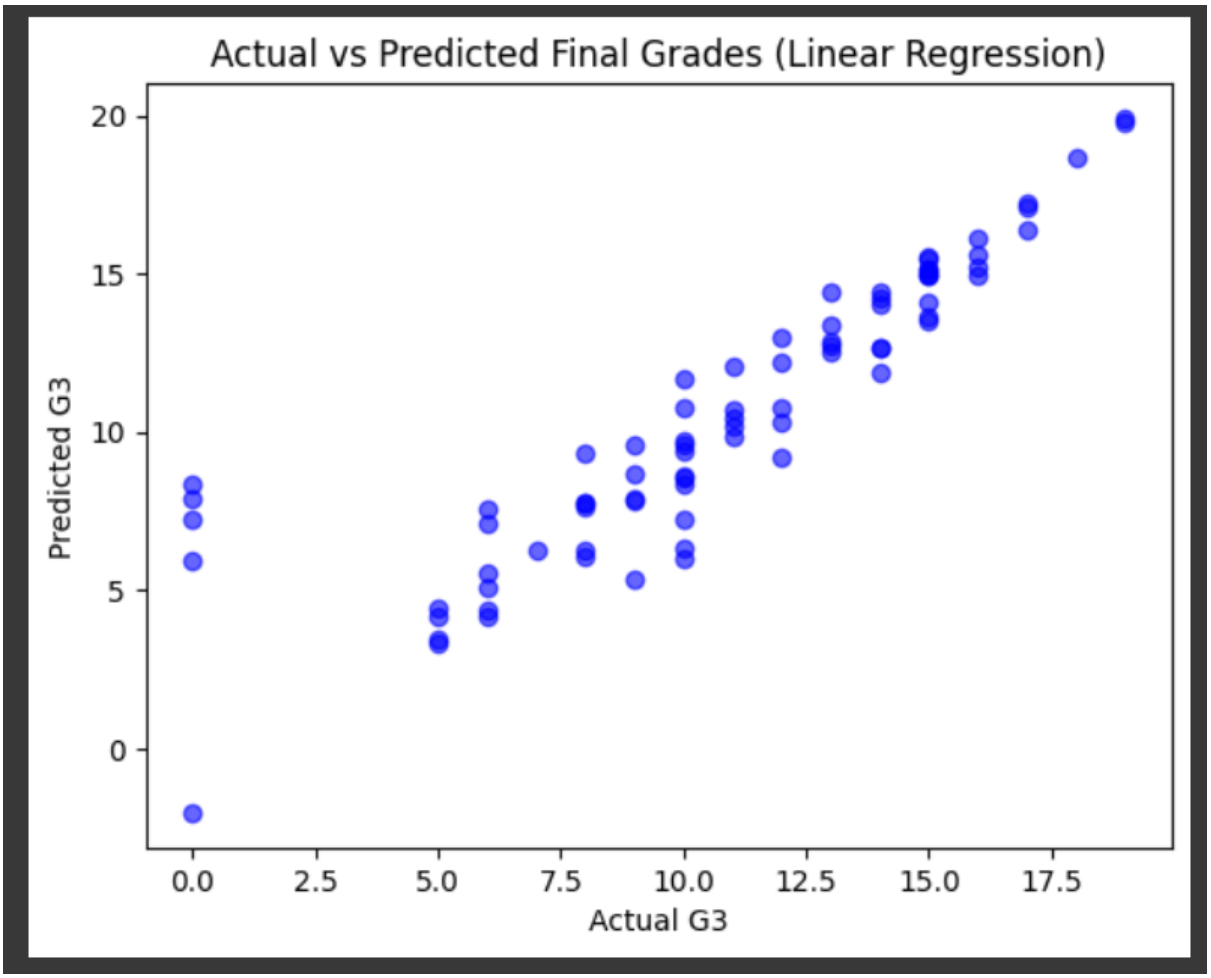
1. **Data Loading**: Imported dataset using pandas.
2. **Feature Selection**: Selected only numeric features relevant to final grade prediction.
3. **Train-Test Split**: Split data into 80% training and 20% testing.
4. **Model Training**: Applied Linear Regression from scikit-learn.
5. **Model Evaluation**: Measured performance using R^2 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))
```

```
Coefficients: [-0.07123057 -0.45581289  0.0392449   0.14446336  0.97961532]
Intercept: -1.6213124035190898
Mean Squared Error: 4.466503212015601
R² Score: 0.7821754247320557
```

```
# 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
- **R^2 Score:** 0.7822