

Write python program to implement Simple Linear Regression from scratch using two different optimization methods—Ordinary Least Squares (Closed-form) and Gradient Descent (Iterative)—and evaluate their performance using five key regression metrics.

### Part 1: Data Generation

Generate a synthetic dataset with a linear relationship and some added noise:

- x values: 100 random points between 0 and 50.
- y values:  $y = 3x + 7 + \text{noise}$  (where noise is a random Gaussian distribution).

### Part 2: Implementation (No Libraries)

You are **prohibited** from using scikit-learn for the model or metrics. You may use NumPy for array operations and Matplotlib for plotting.

- **Function 1: `fit_ols(x, y)`** – Returns m and c using the closed-form equations.
- **Function 2: `fit_gd(x, y, learning_rate, epochs)`** – Returns m and c after n iterations.
- **Function 3: `evaluate_metrics(y_true, y_pred, n_features)`** – Returns a dictionary of the five metrics.

**Part 3: The Metrics Suite** Calculate the following for both models: **MAE, MSE, RMSE,  $R^2$ , Adjusted  $R^2$  score.**

### Part 4: Expected Output Table

Metric	OLS (Analytical)	Gradient Descent (Iterative)
MAE		
MSE		
RMSE		
$R^2$		
Adj $R^2$		