Viết chương trình ***python lưu vào file google colab và nộp lại link chia sẻ*** với kết quả thực hiện các việc sau thực hiện các thao tác sau

Bài 1.

1. Import thư viện numpy với tên rút gọn là 'np'.
2. Import thư viện matplotlib.pyplot với tên rút gọn là 'plt'.
3. Import 'read_csv' function từ pandas.
4. Upload file '50_Startups.csv' lên google colab
5. Đọc vào DataFrame từ file '50_Startups.csv'. Đọc ma trận **X** và **y** như mục 6 từ DataFrame
6. Tính giá trị trung bình, độ lệch chuẩn, giá trị lớn nhất giá trị nhỏ nhất, của y
7. Dùng hàm 'plt.scatter' để vẽ đồ thị sự phụ thuộc của **y** và từng feature (từng cột **X**)
8. Vẽ histogram của y và các fetures trong X, dùng hàm plt.titlte('mean = %5.1f, var = %5.1f' % (mean, var)), để hiển thị title của các histogram là giá trị mean và variance của các biến số.
9. Tính hệ số tương quan (correlation coeficient) của y và các features. Và cho nhận xét.
10. Thông thường khi làm tiền xử lý dữ liệu thì các dữ liệu thường được chuẩn hoá về khoảng [0, 1] (minmax scaling), hoặc được chuẩn hoá sao cho mean = 0 và variance = 1 (standarlization). Việc này sẽ làm cho các thuật toán tối ưu dễ hội tụ hơn.
    a. Minmax scaling: x_norm = (x – x.min())/(x.max() – x.min()
    b. Standardization: x_norm = (x-x.mean())/x.std()
    c. Viết chương trình python để chuyển X thành X1 với giá trị min của các cột là 0 và giá trị max là 1, và tương tự chuyển y thành y1.
    d. Viết chương trình python để standardize (theo các cột) X thành X2. Tính mean và std theo các cột của, và tương tự chuyển y thành y2.

Bài 2. Giả sử 'profit' của các startups phụ thuộc tuyến tính vào 'R&D spend', 'Administration', và 'Marketting spend' như sau: $\hat{y} = a_1x_1 + a_2x_2 + a_3x_3 + b$, trong đó $\hat{y}$ là giá trị dự đoán 'profit' của các startups. Hàm loss function (L) phụ thuộc vào a_1, a_2, a_3, và b:

$$L(a,b) = \sum_i \frac{1}{2m}(\hat{y}_i - y_i)^2, \text{ với } \hat{y} = a_1x_1 + a_2x_2 + a_3x_3 + b.$$

1. Viết hàm python để tính hàm riêng của L theo a_1, a_2, a_3, và b:Sử dụng thuật toán gradient descent tìm a_1, a_2, a_3, và b, với dữ liệu là X1 và y1
2. Vẽ đồ thị sự phụ thuộc của hàm loss function vào số bước thực hiện gradient descent
3. Sử dụng giá trị a_1, a_2, a_3, và b tính giá trị dự đoán cho các startup: y_hat
4. Dùng hàm plt.scatter để vẽ đồ thị y_hat theo y.
5. Đánh giá sai số của phép dự đoán bằng RMSE, MAE, và R2