ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN

CHƯƠNG TRÌNH CHẤT LƯỢNG CAO

ĐỒ ÁN 03 – LINEAR REGRESSION

MÔN: TOÁN ỨNG DỤNG VÀ THỐNG KÊ CHO CÔNG NGHỆ THÔNG TIN

HỌ VÀ TÊN: LÊ ĐỨC ĐẠT

MSSV: 20127674

LỚP: 20CLC08

TP.HCM, 01/08/2022

1. Danh sách các công việc đã hoàn thành:

STT	Yêu cầu	Tiến độ hoàn thành	Lí do(nếu có)
1	Sử dụng toàn bộ 10 đặc trưng	100%	
2	Xây dựng mô hình 1 đặc trưng, tìm mô hình có kết quả tốt nhất	100%	
3	Tự xây dựng mô hình, tìm mô hình cho kết quả tốt nhất	0%	Chịu, em làm ra error, định copy luôn nhưng thấy con số 0đ nên thôi, bỏ luôn ạ.

2. Danh sách các thư viện đã dùng:
   - Numpy: để chuyển dữ liệu sang dạng array (mảng), đồng thời tính toán các phép toán.
   - Pandas: chủ yếu làm dataframe và đọc dữ liệu (mẫu của GV).
3. Ý tưởng làm, mô tả các hàm:

   *** Ngoài việc đọc dữ liệu (của GV), thì em có bổ sung:
   - 1 biến articles lấy danh sách các article của dữ liệu trong file ‘train.csv’

     ```
     articles = list(train.columns.values)
     ```
   - 1 biến data để chuyển dữ liệu trong file ‘train.csv’ sang kiểu array bằng thư viện numpy ở trên để dễ thao tác.

     ```
     data = np.array(train)
     ```

     Source: https://www.geeksforgeeks.org/change-data-type-of-given-numpy-array/.

a. Sử dụng toàn bộ 10 đặc trưng đề bài cung cấp:

- Ý tưởng: Ta tĩm giá trị của x bằng cách sử dụng công thức ma trận x = A^(t)b.

  Trong đó: A: Ma trận được lấy từ dữ liệu sau khi bỏ cột Life expectancy

  b: vector của cột Life expectancy.

- Các hàm sử dụng:

  + def exponential: dùng để tính x ^ cho mô hình hoogi quy tuyến tính dựa vào dữ liệu ‘train.csv’

  + def linearRegression: Kết quả của pp Hồi quy tuyến tính áp dụng trên dữ liệu đã có sẵn, chỉ in ra các giá trị của vector xPow được tính từ hàm xPowData.

- Kết quả:

	x	Các tính chất	Giá trị tương ứn
0	x1	Adult Mortality	0.01510
1	x2	BMI	0.09022
2	x3	Polio	0.04292
3	x4	Diphtheria	0.13928
4	x5	HIV/AIDS	-0.56733
5	x6	GDP	-0.00010
6	x7	Thinness age 10-19	0.74071
7	x8	Thinness age 5-9	0.19093
8	x9	Income composition of resources	24.50597
9	x10	Schooling	2.39351

b. Xây dựng 1 mô hình đặc trưng, tìm mô hình có kết quả tốt nhất:

- Ý tưởng: Dùng phương pháp Cross Validation của GV để tính sai số trung bình cho 10 dữ liệu, mỗi dữ liệu có sẵn có 1 và chỉ 1 tính chất, và dữ liệu tốt nhất khi và chỉ khi có sai sốt trung bình tốt nhất. Ta sẽ chia các dữ liệu thành 4 phần: 1, 2, 3 và 4, và sẽ tính sai số 4 lần. Ta sẽ lấy trung

bình để lấy sai số trung bình của dữ liệu tương ứng với dữ liệu mà ta xét. Cuối cùng, ta xếp hạng để chọn ra dữ liệu tốt nhất để lập mô hình hồi quy tuyến tính.

- Các hàm sử dụng:

  + def surplusX: ta tính sai số 1 cặp dữ liệu đồng thời trên train và test.

  + def averageSurplusX: Dùng phương pháp Cross Validation để tính sai số trung bình của 1 dữ liệu. Nó sẽ trả về sai số trung bình và danh sách 4 số 1, 2, 3 và 4.

  + def surplusAverageAttribute: Tính sai số trung bình của các dữ liệu (1 dữ liệu : 1 đặc trưng). Nó sẽ quét qua 10 dữ liệu bang cách dùng for, mỗi dữ liệu chỉ có 1 tính chất, ta gọi hàm def averageSurplusX lại để tính sai số trung bình của dữ liệu đang được quét ở trên.

  + def rankTable: Xếp hạng các sai số, sau đó in dưới dạng bảng dataframe.

  + def bestDataTable: từ rankTable ở trên, ta chọn 1 sai số bé nhất -> tốt nhất, và cũng đưa về dạng bảng.

- Kết quả:

  + Rank Table:

	Các tính chất	Sai số 1	Sai số 2	Sai số 3	Sai số 4	Sai số trung bình	Rank
0	Income composition of resources	8.621934	9.004420	7.791497	10.651009	9.017215	1
1	Schooling	7.950007	10.220004	8.550248	11.468748	9.547251	2
2	Diphtheria	9.548679	9.617267	10.217174	9.397534	9.695164	3
3	Polio	9.122239	10.850506	11.885671	10.779121	10.659384	4
4	BMI	20.657908	21.106088	19.294803	23.196867	21.063916	5
5	Adult Mortality	40.288426	43.730484	42.586144	35.976985	40.645510	6
6	Thinness age 5-9	38.970765	49.016939	55.141616	38.485292	45.403653	7
7	Thinness age 10-19	40.045328	49.742231	54.297764	39.698210	45.945883	8
8	GDP	58.103451	59.996213	57.579455	60.270391	58.987377	9
9	HIV/AIDS	66.053463	67.394397	69.816191	61.046109	66.077540	10

+ Best Rank Table:

	x	Các tính chất	Giá trị tương ứng
0	x1	Income composition of resources	104.70548

4. Tư liệu tham khảo:
   - Github anh Kiều Công Hậu: https://github.com/kieuconghau/linear-regression.
   - Github anh Đoàn Đình Toàn: https://github.com/t3bol90/ST-MA-Lab04.
   - ML cơ bản: https://machinelearningcoban.com/2016/12/28/linearregression/.
   - https://stackoverflow.com/questions/21926020/how-to-calculate-rmse-using-ipython-numpy.
   - https://www.delftstack.com/howto/python/rmse-python/.