KHOA CÔNG NGHỆ THÔNG TIN

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN – ĐHQG TPHCM

TOÁN ỨNG DỤNG VÀ THỐNG KÊ CHO CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN 3: LINEAR REGRESSION

LỚP: 21CLC08

HỌ VÀ TÊN: NGÔ QUỐC QUÝ

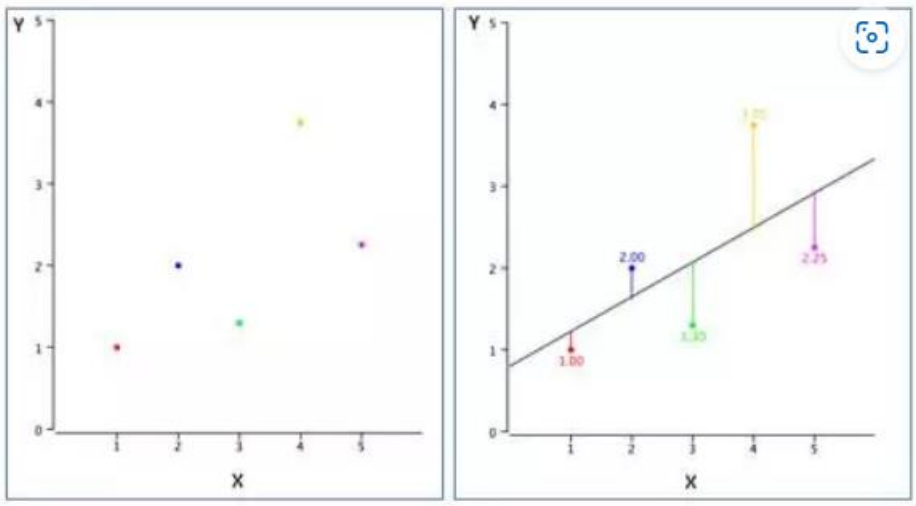
MSSV: 21127679

Mục Lục

I. Giới thiệu đồ án:

1. Linear Regression: [1]

- “Linear Regression” hay "Hồi quy tuyến tính" là một phương pháp thống kê để hồi quy dữ liệu với biến phụ thuộc có giá trị liên tục trong khi các biến độc lập có thể có một trong hai giá trị liên tục hoặc là giá trị phân loại. Nói cách khác "Hồi quy tuyến tính" là một phương pháp để dự đoán biến phụ thuộc (Y) dựa trên giá trị của biến độc lập (X). Nó có thể được sử dụng cho các trường hợp chúng ta muốn dự đoán một số lượng liên tục.
- Hồi quy tuyến tính bao gồm các khái niệm về:
    - Tương quan: Giải thích mối quan hệ giữa 2 biến
    - Phương sai: Đánh giá độ phân tán trong dữ liệu
    - Độ lệch chuẩn: Đánh giá độ phân tán trong dữ liệu – căn bậc 2 của phương sai
    - Phân phối chuẩn
    - Sai số
- Đường hồi quy tuyến tính: thể hiện đường gần đúng nhất để có thể giải thích được mối quan hệ giữa trục x và y.



- Nếu có một biến phụ thuộc là y và một biến độc lập là x, ta sẽ có công thức biểu diễn mối quan hệ của x và y như sau:

$$y = \theta_1 + \theta_2 . x$$

- Khi đào tạo mô hình, ta sẽ được cung cấp:
    - X: dữ liệu đào tạo đầu vào (đơn biến – một biến đầu vào).
    - y: nhãn dữ liệu.
- Hiệu suất của mô hình được đánh giá là có đủ để dự đoán được tương lai hoặc mối quan hệ mà bạn đã xây dựng giữa các biến độc lập và phụ thuộc là có đủ hay không.

- Một số cách đánh giá:
    - Mean Squared Error (MSE)

$$MSE = \frac{1}{n}\sum_{n}^{i=1}(y_i - \hat{y}_i)^2$$

    - Root Mean Squared Error (RMSE)

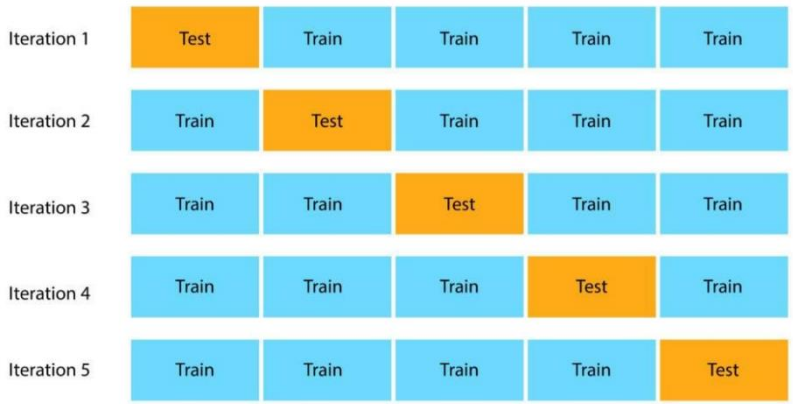
$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{n}\sum_{n}^{i=1}(y_i - \hat{y}_i)^2}$$

    - Mean Absolute Error (MAE)

$$MAE = \frac{1}{n}\sum_{n}^{i=1}|y_i - \hat{y}_i|$$

2. K Fold Cross Validation: [2]

- Cross validation là một kỹ thuật lấy mẫu để đánh giá mô hình học máy trong trường hợp dữ liệu không được lớn.
- Tham số quan trọng trong kỹ thuật này là k, tham số này đại diện cho số nhóm được chia ra. Khi giá trị của k được lựa chọn, ta sẽ lấy trực tiếp giá trị đó trực trong tên của phương pháp đánh giá. (ví dụ: 5-fold cross validation).
- Kỹ thuật bao gồm các bước như sau:
    - Xáo trộn dataset.
    - Chia dataset thành k nhóm
    - Với mỗi nhóm:
        - Sử dụng nhóm hiện tại để đánh giá hiệu quả mô hình
        - Các nhóm còn lại được sử dụng để huấn luyện mô hình (k – 1)
        - Huấn luyện mô hình
        - Đánh giá và sau đó hủy mô hình.
    - Tổng hợp hiệu quả của mô hình dựa vào các số liệu đánh giá.



- Trong hình trên (trường hợp k = 5) , mỗi một iteration, ta sẽ tính được 1 score, và sau cả quá trình, ta sẽ được 5 score sau mỗi lần huấn luyện, và ta sẽ lấy trung bình của 5 score trên.

$$overall\ score = \frac{score1\ +\ score2\ +\ score\,3\ +\ score4\ +\ score5}{5}$$

- Các chiến thuật cấu hình k phù hợp và phổ biến:
    - Đại diện: Giá trị của k được chọn để mỗi tập train/test đủ lớn, có thể đại diện về mặt thống kê cho dataset chứa nó.
    - k = 10: Giá trị của k được chọn để mỗi tập train/test đủ lớn, có thể đại diện về mặt thống kê cho dataset chứa nó.
    - k = n: Giá trị của k được gán cố định bằng n , với n là kích thước của dataset, như vậy mỗi mẫu sẽ được sử dụng để đánh giá mô hình một lần. Cách tiếp cận này còn có tên leave-one-out cross-validation.

3. Mục tiêu đồ án:

- Ứng dụng linear regression và k fold cross validation để tìm ra các yếu tố quyết định để mức lương của các kỹ sư ngay sau khi tốt nghiệp. Các yếu tố như điểm số ở các cấp/trường đại học, kỹ nằng của ứng viên, sự liên kết giữa các trường đại học và các khu công nghiệp/ công ty công nghệ, bằng cấp của sinh viên và điều kiện thị trường cho ngành công nghiệp cụ thể sẽ ảnh hưởng đến điều này.
- Bộ dữ liệu bao gồm: 34 thuộc tính.
- Sau bước tiền xử lí, bộ dữ liệu chỉ còn 24 thuộc tính với 1 giá trị mục tiêu là Salary (y) và 23 đặc trưng (X) để tìm ra giá trị mục tiêu: `Gender`, `10percentage`, `12percentage`, `CollegeTier`, `Degree`, `collegeGPA`, `CollegeCityTier`, `English`, `Logical`, `Quant`, `Domain`, `ComputerProgramming`, `ElectronicsAndSemicon`, `ComputerScience`, `MechanicalEngg`, `ElectricalEngg`, `TelecomEngg`, `CivilEngg`, `conscientiousness`, `agreeableness`, `extraversion`, `nueroticism`, `openess_to_experience`.

II. Thư viện và hàm sử dụng:

1. Function và class trong các thư viện: [4] [5] [6]

- import pandas as pd: (library)
    - Đọc file csv, tạo dataframe và series.
- import numpy as np: (library)
    - Tạo ra ma trận và xử lí các phép toán (viết hàm cross_validation).
- from sklearn.model_selection import cross_val_score (function)
    - Sử dụng để tìm ra score cho câu d.

    - Đầu vào:
        - estimator: Truyền vào một linear regression model
        - X: Tập dữ liệu cần thực hiện k fold cross validate
        - y: tập mục tiêu
        - scoring: kiểu đánh giá
        - cv: tham số k (5, 10 …)
    - Đầu ra:
        - score: Sau khi thực hiệc k fold cross validate, giá trị score được trả ra là 1 mảng gồm k phần tử là k score của mỗi iteration.
- from sklearn.model_selection import LinearRegression (class)
    - Sử dụng để tìm ra score cho câu d.
    - Đầu vào: Không có
    - Đầu ra: Tạo ra một mô hình linear regression.
- import matplotlib.pyplot as plt (library)
    - Hỗ trợ tạo ra đồ thị biểu diễn độ tương quan giữa các đặc trưng.
- import seaborn as sns (library)
    - Hỗ trợ tạo ra đồ thị biểu diễn độ tương quan giữa các đặc trưng.
- sample (function) (from pandas):
    - Đầu vào: frac = 1
    - Đầu ra: một tập dataset đã được xáo trộn.

2. Các function và class tự viết:

- OLSLinearRegression (class)
    - fit (hàm con):
        - Đầu vào: X (ma trận đặc trưng), y (vector kết quả)
        - Đầu ra: Chính đối tượng hiện tại sau khi thực hiện
        - Mô tả:
            - Tính ma trận nghịch đảo của ma trận X transposed nhân với chính nó (X.T @ X). Đây là bước chính để tính toán các tham số mô hình tối ưu dựa trên phương pháp OLS.
            - Nhân ma trận nghịch đảo trên với ma trận X.T để tính toán vector tham số self.w.
            - Lưu vector tham số này vào biến self.w để sử dụng cho việc dự đoán và truy xuất sau này.
            - Trả về chính đối tượng hiện tại (self) sau khi đã huấn luyện.
    - get_params (hàm con):
        - Đầu vào: Không có
        - Đầu ra: Trả về vector tham số mô hình ‘self.w’
        - Mô tả: Kiểm tra và truy xuất các tham số sau khi mô hình đã được huấn luyện.

    - predict (hàm con):
        - Đầu vào: X (ma trận đặc trưng)
        - Mô tả: Phương thức này được sử dụng để dự đoán kết quả dự đoán dựa trên mô hình đã huấn luyện. Nó nhận đối số X (ma trận đặc trưng của dữ liệu đầu vào) và thực hiện dự đoán bằng cách tính tổng trọng số thích ứng với từng đặc trưng trong X. Cụ thể, nó nhân mỗi giá trị trong X với tương ứng với trọng số từ self.w, sau đó tính tổng theo hàng (axis=1) để có kết quả dự đoán.
- mae (function):
    - Đầu vào: y (dãy giá trị thực tế), y_hat (dãy giá trị dự đoán).
    - Đầu ra: Một số thực, đại diện cho giá trị trung bình của sự sai lệch tuyệt đối giữa y và y_hat.
    - Mô tả: hàm tính sự sai lệch tuyệt đối giữa mỗi cặp giá trị tương ứng trong y và y_hat, sau đó tính giá trị trung bình của các sai lệch này bằng cách sử dụng hàm np.mean(). Giá trị trả về cuối cùng là giá trị trung bình này.
- cross_validation (function):
    - Đầu vào: X (ma trận đặc trưng), y (vector mục tiêu), k (số lượng fold sẽ chia).
    - Đầu ra: Giá trị trung bình của mae sau k iterations.
    - Mô tả:
        - Tính kích thước của tập test bằng cách chia độ dài tập X cho k.
        - Sử dụng một vòng lặp để chia dữ liệu thành k phần, mỗi phần sẽ được sử dụng lần lượt làm tập kiểm tra. Các phần còn lại sẽ được sử dụng làm tập huấn luyện.
        - Đối với mỗi lần kiểm tra chéo, tạo các DataFrame Pandas từ các mảng NumPy để sử dụng với mô hình OLSLinearRegression. Sau đó, huấn luyện mô hình trên tập huấn luyện và sử dụng mô hình đã huấn luyện để dự đoán kết quả trên tập kiểm tra.
        - Tính toán sai số tuyệt đối trung bình (MAE - Mean Absolute Error) giữa dự đoán và kết quả thực tế trên mỗi lần kiểm tra chéo.
        - Cuối cùng, trả về giá trị trung bình của các MAE trên tất cả các lần kiểm tra chéo. Điều này là một thước đo cho độ lỗi trung bình của mô hình trên dữ liệu.

III. Kết quả và giải thích:

Câu 1a

- Sử dụng toàn bộ 11 đặc trưng đầu tiên để huấn luyện.
- Kết quả:
  - Giá trị cho công thức hồi quy cho mô hình

```
0    -22756.512821
1       804.503156
2      1294.654565
3    -91781.897531
4     23182.388679
5      1437.548672
6     -8570.661985
7       147.858299
8       152.888476
9       117.221846
10    34552.286221
dtype: float64
```

  - Mae:

```
MAE:  104863.7775403339
```

  - Công thức hồi quy:

$$Salary = -22756.513 * Gender + 804.503 * 10percentage + 1294.654 * 12percentage + (-91781.898) * ColloegeTier + 23182.389 * Degree + 1437.549 * collegeGPA + (-8570.662) * CollegeCityTier + 147.858 * Englist + 152.888 * Logical + 117.223 * Quant + 34552.286 * Domain$$

Câu 1b:

- Xây dựng trên duy nhất 1 đặc trưng với các đặc trưng bao gồm 'conscientiousness', 'agreeableness', 'extraversion', 'nueroticism', 'openess_to_experience' .
- Kết quả:
  - Mae nhỏ nhất:

```
MAE:  ['nueroticism', 298887.0886000371]
```

  - Bảng đặc trưng:

	Features	MAE
0	conscientiousness	306289.197791
1	agreeableness	301253.719611
2	extraversion	306851.585531
3	nueroticism	298887.088600
4	openess_to_experience	303104.430081

- Dữ liệu cho công thức hồi quy:

```
0   -56546.303753
dtype: float64
```

- Mae tốt nhất được xây dựng từ model vừa huấn luyện:

```
Best mae in best_personality_feature_model:  291019.693226953
```

- Công thức hồi quy:

$$Salary = -56546.304 * Neuroticism$$

- Giải thích:
  - Lựa chọn tập dữ liệu bao gồm các đặc trưng theo đề bài.
  - Thực hiện vòng lặp và shuffle tập dữ liệu, lấy tập X từ tập dữ liệu được lấy ban đầu, và y là cột salary, thực hiện k-fold bằng cách gọi hàm cross_validtion đã được mô tả ở trên.
  - Tìm giá trị mae nhỏ nhất trong từng đặc trưng.
  - Huấn luyện lại dựa vào đặc trưng tốt nhất tương tự câu 1a.

Câu 1c

- Xây dựng trên duy nhất 1 đặc trưng với các đặc trưng bao gồm 'English', 'Logical', 'Quant'.
- Kết quả:
  - Mae nhỏ nhất:

```
Best MAE:  ['Quant', 116613.7924310602]
```

  - Bảng đặc trưng:

	Features	MAE
0	English	120550.735783
1	Logical	119198.663764
2	Quant	116613.792431

  - Dữ liệu cho công thức hồi quy:

```
0    585.895381
dtype: float64
```

  - Mae tốt nhất xây dựng từ model vừa huấn luyện:

```
Best mae in best_personality_feature_model:  106819.5776198967
```

- Công thức hồi quy:

$$Salary = 585.895 * Quant$$

- Giải thích: thay đổi tập dữ liệu và thực hiện tương tự câu 1b

Câu 1d

- Tự xây dựng mô hình và tìm mô hình cho kết quả tốt nhất.
- Kết quả:
  - Danh sách các model:

	Mô hình	MAE
0	model1	115355.777575
1	model2	117818.650588
2	model3	117828.404083
3	model4	114626.689800

  - Dữ liệu xây dựng công thức hồi quy từ mô hình tốt nhất:

```
0       206.501649
1    -82570.246621
2      1513.503236
3       209.366131
4      1992.911864
dtype: float64
```

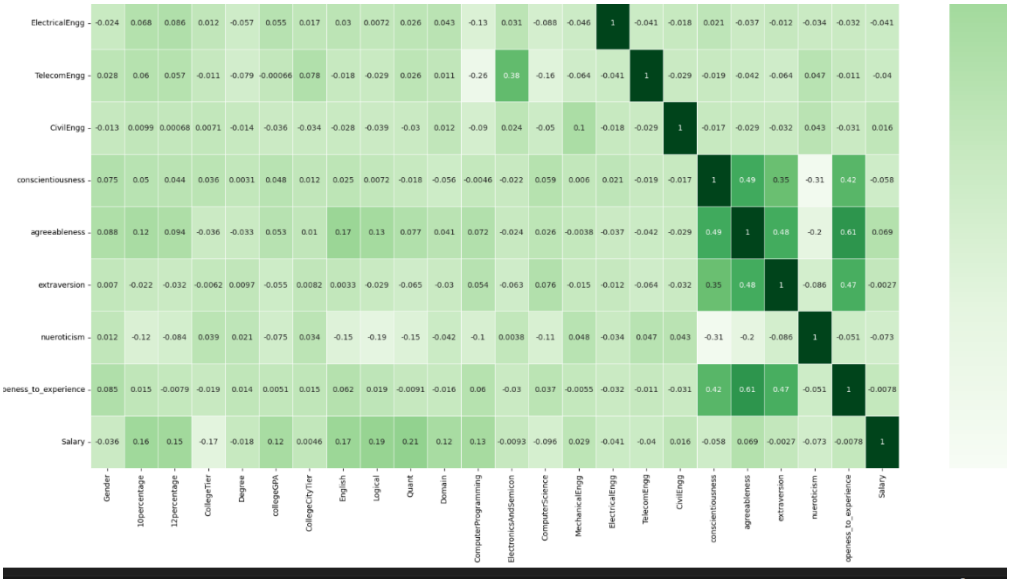
  - Mae xây dựng từ mô hình tốt nhất:ư

```
MAE:  104184.56578915539
```

- Công thức hồi quy:

$$Salary = 206.502 * Quant + (-82570) * CollegeTier + 1513.503 * 12percentage + 209.366 * English + 1992.912 * collegeGPA$$

- Xây dựng biểu đồ tương quan tìm ra đặc trưng có ảnh hưởng tích cực đến Salary. **[3]**



Một phần biể đồ (do biểu đồ quá lớn) – cụ thể trong file ipynb

- Các model xây dựng:
    - Model 1: Sử dụng kết hợp 5 đặc trưng Logical, English, Quant, Domain, ComputerProgramming.
        - Lí do sử dụng: Đây là điểm số của các phần quan trọng của AMCAT, sau khi quan sát biểu đồ tương quan, các đặc trưng này có ảnh hưởng khá tốt đến khả năng về mức lương của các kỹ sư sau khi tốt nghiệp.
    - Model 2: Sử dụng căn bậc 2 của đặc trưng kết hợp 10percentage, 12percentage, CollegeTier, Degree, collegeGPA.
        - Lí do sử dụng: Đây là các điểm số trung bình ở các cấp, nhận thấy sau khi quan sát biểu đồ tương quan, việc điểm số ở các cấp bậc thấp như 10, 12 có thể ít ảnh hưởng hơn so với bằng cấp ở bậc đại học. Sử dụng căn bậc 2 của các điểm số này để tỉ lệ này tăng cao hơn nhằm đánh giá thử việc khi có điểm trung bình cao ở các cấp bậc có ảnh hưởng nhiều đến mức lương sau khi tốt nghiệp đại học hay không.
    - Model 3: Sử dụng một features mới được tạo từ việc cộng 3 đặc trưng 10percentage, 12percentage, collegeGPA.
        - Lí do sử dụng: Nhận thấy ở model2, việc sử dụng điểm số trung bình ở các cấp bậc khi căn bậc 2 có vẻ không hiệu quả để dự đoán mức lương, việc tạo thêm một mô hình chỉ bao gồm điểm trung bình ở các cấp để kiểm tra xem những đặc trực trên có thật sự ảnh hưởng nhiều đến mức lương của kĩ sau khi tốt nghiệp hay không.
    - Model 4: Sử dụng kết hợp 5 đặc trưng Quant, CollegeTier, 12percentage, English, collegeGPA.
        - Lí do sử dụng: Gần giống với model1 và model2, do điểm số trung bình ở các cấp bậc học thấp không ảnh hưởng quá nhiều, tuy nhiên, ở model1 nhận thấy bài kiểm tra định lượng ở AMCAT, và các điểm trung bình ở bậc cao hơn như lớp 12, đại học, địa phương và trình độ tiếng anh có ảnh hưởng rất nhiều đến mức lương của kĩ sư khi tốt nghiệp, việc sử dụng 5 đặc trưng là gom những yếu tố có vẻ như là tốt nhất ảnh hưởng đến mức lương của các kĩ sư sau tốt nghiệp.
- Giải thích cách làm:
    - Model 1:
        - Sử dụng hàm có sẵn LinearRegression từ thư viện sklearn để tạo ra một mô hình.
        - Copy dữ liệu từ train và xáo trộn.
        - Lấy các dữ liệu bao gồm 5 đặc trưng, và 1 tập mục đích Salary, từ dữ liệu vừa xáo trộn sau đó thực hiện bằng cách gọi hàm cross_val_score từ thư viện sklearn.

        - Do đối số scoring truyền vào hàm cross_val_score là neg_mean_absolute_error nên tất cả giá trị trả ra đều là âm, do đố cần nhân với trừ 1 cho tất cả giá trị trong tập score để lấy được dữ liệu đúng.
        - Truyền dữ liệu sau khi chia trung bình bằng np.mean() và thêm vào một danh sách các mae được xây dựng từ model mới.
    - Model 2:
        - Tương tự với model1
        - Có thêm bước căn bậc 2 tất cả các dữ liệu trong dataset để có một tập dữ liệu mới.
    - Model 3:
        - Tương tự những model trên, tuy nhiên có thêm bước tạo ra một cột mới tên 'sum_col' để chứa tổng các điểm trung bình.
    - Model 4:
        - Giống model 1.

IV. Tài liệu tham khảo:

[1] (Nguyen Duong, 30/5/2027)**Nguyen Duong,** *Linear Regression - Hồi quy tuyến tính trong Machine Learning*, ngày truy cập: 22/8/2023

https://viblo.asia/p/linear-regression-hoi-quy-tuyen-tinh-trong-machine-learning-4P856akRlY3

[2] *Trí tuệ nhân tạo***,** ngày truy cập: 22/8/2023

https://trituenhantao.io/kien-thuc/gioi-thieu-ve-k-fold-cross-validation/

[3] NgocTien0110, *Applied-Mathematics-and-Statistics*, ngày truy cập: 22/8/2023

https://github.com/NgocTien0110/Applied-Mathematics-and-Statistics/blob/main/project%203/20127641.docx

[4] sklearn.org

sklearn.model_selection.cross_val_score — scikit-learn 1.3.0 documentation

[5] numpy.org

https://numpy.org/

[6] pandas.pydata.org

https://pandas.pydata.org/

[7] sklearn.org

sklearn.linear_model.LinearRegression — scikit-learn 1.3.0 documentation