Trường Đại học Khoa học Tự nhiên ĐHQG-HCM

Khoa Công nghệ Thông tin

BÁO CÁO
Đồ án 3: LINEAR REGRESSION

Môn học

Toán Ứng dụng và thống kê cho CNTT (MTH00057)

Giảng viên bộ môn Vũ Quốc Hoàng
Nguyễn Văn Quang Huy
Lê Thanh Tùng
Phan Thị Phương Uyên

Họ và tên Lê Thị Hoài Thư
MSSV 21127176
Lớp 21CLC02

Ngày 24 tháng 8 năm 2023

1 Tổng quan

Nội dung đồ án

Mục tiêu của đồ án là tìm hiểu các yếu tố quyết định mức lương và việc làm của các kỹ sư ngay sau khi tốt nghiệp. Các yếu tố như điểm số ở các cấp/trường đại học, kỹ năng của ứng viên, sự liên kết giữa trường đại học và các khu công nghiệp/công ty công nghệ, bằng cấp của sinh viên và điều kiện thị trường cho các ngành công nghiệp cụ thể sẽ ảnh hưởng đến điều này.

Bộ dữ liệu được sử dụng trong đồ án này thu thập tại Ấn Độ, nơi có hơn 6000 cơ sở đào tạo kỹ thuật công nghệ với khoảng 2,9 triệu sinh viên đang học tập. Mỗi năm, trung bình có 1,5 triệu sinh viên tốt nghiệp chuyên ngành Công nghệ/Kỹ thuật, tuy nhiên do thiếu kỹ năng cần thiết, ít hơn 20% trong số họ có việc làm phù hợp với chuyên môn của mình. Bộ dữ liệu này không chỉ giúp xây dựng công cụ dự đoán mức lương mà còn cung cấp thông tin về các yếu tố ảnh hưởng đến mức lương và chức danh công việc trên thị trường lao động.

Tuy nhiên, dữ liệu sử dụng cho đồ án đã được thực hiện các bước tiền xử lý sau:

- Loại bỏ các cột có giá trị là chuỗi.
- Loại bỏ các cột liên quan đến định danh và năm.

Sau quá trình đó, bộ dữ liệu mới có:

- 2998 dòng dữ liệu.
- 24 cột dữ liệu gồm:
  - 1 giá trị mục tiêu ($\mathbf{y}$): Salary (tính bằng Indian Rupee)
  - 23 đặc trưng giải thích ($\mathbf{X}$) (giúp tìm giá trị mục tiêu)

Thực hiện phân tích ảnh hưởng của một hoặc các đặc trưng đến mức lương của các kỹ sư dựa trên dữ liệu đã cho.

Nội dung báo cáo

Trình bày kết quả, đánh giá và nhận xét các mô hình đã xây dựng.

- Liệt kê các thư viện và lý do sử dụng chúng.
- Liệt kê các hàm đã sử dụng và mô tả các hàm đó.
- Báo cáo và nhận xét kết quả từ toàn bộ các mô hình xây dựng được.
- Yêu cầu 1b, 1c và 1d: Giải thích hoặc nêu giả thuyết cho mô hình đạt kết quả tốt nhất ở mỗi yêu cầu.
- Yêu cầu 1d: Trình bày toàn bộ quá trình và lý do trích chọn/thiết kế các đặc trưng cho $\mathbf{m}$ mô hình mà sinh viên xây dựng.
- Tài liệu tham khảo.

Mục lục

2 Những thư viện đã sử dụng

Các thư viện được sử dụng trong đồ án bao gồm `numpy, pandas, sklearn, matplotlib`. Cụ thể:

- Thư viện `pandas`: dùng để làm việc dễ dàng và trực quan với dữ liệu có cấu trúc dạng bảng.
- Thư viện `numpy`: dùng để thực hiện tính toán hiệu quả với ma trận và mảng.
- Thư viện `sklearn`:
  + Module `sklearn.metrics`: dùng `mean_absolute_error` để tính sai số (độ lỗi) tuyệt đối trung bình.
  + Module `sklearn.utils`: dùng `shuffle` để xáo trộn bộ dữ liệu huấn luyện khi thực hiện `k-fold cross validation`.
  + Module `sklearn.feature_selection`: dùng `VarianceThreshold` để chọn mô hình tốt.
- Thư viện `matplotlib`:
  + Module `matplotlib.pyplot`: vẽ biểu đồ.
- Thư viện `scipy`:
  + Modul `scipy.stats`: dùng `pearsonr` để tính giá trị tương quan của biến `salary` và các đặc trưng còn lại.

Cài đặt `np.set_printoptions(precision=3)`[1] nhằm hỗ trợ làm tròn đến 3 chữ số thập phân khi in tham số.

3 Những hàm đã sử dụng

3.1 Hàm thư viện

• shuffle

Parameters:

`arrays`: Kiểu cấu trúc dữ liệu có index như arrays, lists, dataframes.

Returns:

`shuffled_arrays`: Mảng copy đã được trộn, bản gốc không bị ảnh hưởng.

Hàm `shuffle`[2] hỗ trợ trộn bộ dữ liệu. Khi không thay đổi các tham số mặc định khác (không trình bày ở trên) thì hàm sẽ thực hiện xáo theo hàng và giữ nguyên dữ liệu thuộc cột như thứ tự cột ban đầu.

• mean_absolute_error

Parameters:

`y_true`: Bộ dữ liệu mục tiêu.

`y_pred`: Bộ dữ liệu được dự đoán từ mô hình hồi quy tuyến tính.

Returns:

`loss`: float, giá trị **MAE**.

Hàm `mean_absolute_error`[3] hỗ trợ tính sai số (độ lỗi) tuyệt đối trung bình. Công thức tính **MAE** như sau, với **n** là số lượng mẫu quan sát:

$$\textbf{MAE} = \frac{1}{n}\sum_{n}^{i=1} |\texttt{y\_true}_i - \texttt{y\_pred}_i|$$

3.2 Lớp `OLSLinearRegression`

Lớp tìm kiếm mô hình hồi quy tuyến tính sử dụng phương pháp bình phương tối thiểu (OLS Linear Regression) được tham khảo từ tài liệu thực hành **Lab04** của môn học.

Để lấy được mô hình, ta cần fit vào tập dữ liệu huấn luyện. Hàm `get_params()` của object trả về tham số của mô hình sau khi đã fit dữ liệu. Hàm `predict()` dùng để dự đoán dữ liệu đích từ mô hình hồi quy tuyến tính, ứng với bộ dữ liệu đầu vào.

• fit(`self, X, y`)

Input: Bộ dữ liệu đầu vào `[X: indexable data structure]`
Dữ liệu thực tế cần dự đoán (`salary`) `[y: indexable data structure]`
Output: Mô hình hồi quy đã fit `[self: <class 'OLSLinearRegression'>]`

Ý tưởng: Công thức tìm tham số w của mô hình được khai triển như sau:

$$w = (\mathtt{X}^\mathtt{T}\mathtt{X})^{-1}\mathtt{X}^\mathtt{T}\mathtt{y}$$

Mô tả: Lần lượt cài đặt theo công thức, với `X.T` sẽ trả về ma trận chuyển vị của ma trận X. Hàm `np.linalg.inv()` dùng để lấy ma trận nghịch đảo và phép nhân ma trận được thực hiện thông qua toán tử @. Thực hiện gán `self.w` là kết quả của tham số tính trên rồi trả về mô hình (`self`) đã gán tham số.

• get_params(`self`)

Input: Không có dữ liệu đầu vào.
Output: Danh sách tham số của mô hình hồi quy tuyến tính `[res: np.ndarray]`

Mô tả: Trả về `self.w`.

• predict(`self, X`)

Input: Bộ dữ liệu đầu vào `[X: indexable data structure]`
Output: Bộ dữ liệu được dự đoán từ mô hình hồi quy tuyến tính `[y_predict]`

Ý tưởng: Để tính toán được bộ dữ liệu dự đoán, ta triển khai theo công thức sau:

$$\texttt{y\_predict}_i = w_1 \times \texttt{features\_1}_i + w_2 \times \texttt{features\_2}_i + \ldots$$

Mô tả: Để đảm bảo danh sách tham số của mô hình là mảng 1 chiều, ta gọi hàm `ravel()`[4]. Thực hiện nhân mảng tham số với bộ dữ liệu đầu vào ta sẽ được mảng mới gồm lần lượt $w_1 \times$`feature_1`$_1$, ... Sau cùng gọi hàm `np.sum` với `axis=1` để tính tổng theo chiều ngang của mảng, kết quả thu được chính là `y_predict`.

3.3 Hàm tự tổ chức

• train_lr_model(features)

Input: Danh sách tên các đặc trưng cần huấn luyện `[features: list]`
Output: Mô hình hồi quy đã huấn luyện `[lr: <class 'OLSLinearRegression'>]`

Ý tưởng: Hàm trả về mô hình đã được huấn luyện trên bộ dữ liệu `(X_train, y_train)` dựa trên các đặc trưng được truyền vào.

Mô tả: Ép kiểu các cột dữ liệu của danh sách đặc trưng về `np.ndarray` để dễ tính toán. Sau cùng, chương trình trả về lớp mô hình hồi quy tuyến tính bằng cách gọi hàm `fit()` của lớp `OLSLinearRegression` và truyền bộ dữ liệu cần huấn luyện vào.

• divide_size(size, k_fold=5)

Input: Kích thước mẫu cần chia `[size: int]`
Output: Mảng các kích thước đã được chia `[res: np.ndarray, dtype=int]`

Ý tưởng: Hàm chia kích thước mẫu thành các kích thước con sao cho tổng thể sự chênh lệch giữa mỗi cặp là nhỏ nhất.

Mô tả: Khởi tạo mảng kết quả gồm toàn giá trị 1 có kích thước bằng `k_fold` và nhân toàn bộ ma trận với phần nguyên của phép chia `size` cho `k_fold`. Sau cùng ta cộng 1 cho m (m = phần dư của phép chia `size` cho `k_fold`) phần tử cuối cùng của mảng kết quả rồi `astype(int)` khi trả về.

• rank_features(features, k_fold=5)

Input: Danh sách tên các đặc trưng cần phân tích `[features: list]`
Output: Bảng kết quả trung bình giá trị **MAE** các bộ mẫu theo từng đặc trưng `[res: np.ndarray, dtype=float]`

Ý tưởng: Hàm sử dụng K-Fold Cross Validation (mặc định k=5) để tìm ra mô hình hồi quy tuyến tính theo đặc trưng tốt nhất. Với mỗi fold thực hiện huấn luyện và đo độ lỗi **MAE** trên chính fold đó. Sau đó lấy trung bình độ lỗi của từng đặc trưng trên tổng số các fold. Đặc trưng nào có trung bình giá trị MAE nhỏ nhất thì sự tương quan/liên quan của đặc trưng đó đến lương là lớn nhất.

Mô tả: Đầu tiên thực hiện `shuffle`[2] bộ dữ liệu huấn luyện sau đó tách riêng các đặc trưng (`X_train`) và lương (`y_train`) tương ứng.

Tiếp đến thực hiện khởi tạo một mảng toàn 0 để lưu giữ giá trị MAE kích thước (`len(features)` × `k_fold`) với số dòng là số đặc trưng, số cột là số lượng fold sẽ chia. Ngoài ra cần khởi tạo mảng lưu các kích thước dữ liệu đã được chia gọn nhất bằng hàm `divide_size()`. Cài đặt 2 vòng lặp lồng nhau:

- Vòng lặp ngoài để lấy thứ tự và dữ liệu bộ dữ liệu con (`X_sample, y_sample`)
- Vòng lặp trong để lấy từng đặc trưng cần phân tích. Thực hiện fit sau đó lưu hết các giá trị MAE của các đặc trưng trên cùng bộ dữ liệu sau khi vòng lặp này kết thúc.

Kết quả trả về thông qua hàm `np.mean()` sẽ cho ta mảng một chiều (như một vector hàng) các giá trị trung bình MAE. Vì vậy, cần `reshape()` nó thành một vector cột để thư viện `pandas` thuận tiện in bảng kết quả.

• rank_models(features, k_fold=5)

Input: Danh sách chứa danh sách tên các đặc trưng của mô hình cần phân tích.
`[features: list]`

Output: Bảng kết quả trung bình giá trị **MAE** các bộ mẫu theo từng mô hình.
`[res: np.ndarray, dtype=float]`

Ý tưởng: Hàm sử dụng K-Fold Cross Validation (mặc định k=5) để tìm ra mô hình hồi quy tuyến tính tốt nhất trong các mô hình. Với mỗi fold thực hiện huấn luyện và đo độ lỗi **MAE** trên chính fold đó. Sau đó lấy trung bình độ lỗi của từng mô hình trên tổng số các fold. Mô hình nào có trung bình giá trị MAE nhỏ nhất thì mô hình đó là mô hình tốt nhất.

Mô tả: Đầu tiên thực hiện `shuffle`[2] bộ dữ liệu huấn luyện sau đó tách riêng các đặc trưng (`X_train`) và lương (`y_train`) tương ứng.

Tiếp đến thực hiện khởi tạo một mảng toàn 0 để lưu giữ giá trị MAE kích thước (`len(features)` $\times$ `k_fold`) với số dòng là số mô hình, số cột là số lượng fold sẽ chia. Ngoài ra cần khởi tạo mảng lưu các kích thước dữ liệu đã được chia gọn nhất bằng hàm `divide_size()`. Cài đặt 2 vòng lặp lồng nhau:

- Vòng lặp ngoài để lấy thứ tự và dữ liệu bộ dữ liệu con (`X_sample, y_sample`)
- Vòng lặp trong để lấy từng mô hình cần phân tích. Thực hiện fit sau đó lưu hết các giá trị MAE của các mô hình trên cùng bộ dữ liệu sau khi vòng lặp này kết thúc.

Kết quả trả về thông qua hàm `np.mean()` sẽ cho ta mảng một chiều (như một vector hàng) các giá trị trung bình MAE. Vì vậy, cần `reshape()` nó thành một vector cột để thư viện `pandas` thuận tiện in bảng kết quả.

4 Kết quả & nhận xét

4.1 Kết quả

4.1.1 Phân tích dựa trên 11 đặc trưng đầu tiên đề bài cung cấp

Bộ 11 đặc trưng đầu tiên bao gồm: `Gender`, `10percentage`, `12percentage`, `CollegeTier`, `Degree`, `collegeGPA`, `CollegeCityTier`, `English`, `Logical`, `Quant`, `Domain`.

Huấn luyện một lần duy nhất 11 đặc trưng trên trên bộ dữ liệu `train` và kiểm tra trên bộ dữ liệu `test` ta được kết quả sau:

MAE: `104863.77754033315`
params: `[-22756.513, 804.503, 1294.655, -91781.898, 23182.389, 1437.549, -8570.662, 147.858, 152.888, 117.222, 34552.286]`

Công thức hồi quy:

$$\textbf{Salary} = \texttt{-22756.513} \times Gender \texttt{ + 804.503} \times 10percentage \texttt{ + 1294.655} \times 12percentage$$
$$\texttt{- 91781.898} \times CollegeTier \texttt{ + 23182.389} \times Degree \texttt{ + 1437.549} \times collegeGPA$$
$$\texttt{- 8570.662} \times CollegeCityTier \texttt{ + 147.858} \times English \texttt{ + 152.888} \times Logical$$
$$\texttt{+ 117.222} \times Quant \texttt{ + 34552.286} \times Domain$$

4.1.2 Phân tích ảnh hưởng của đặc trưng tính cách đến mức lương của các kỹ sư dựa trên điểm các bài kiểm tra của AMCAT

Bộ đặc trưng tính cách bao gồm: `conscientiousness`, `agreeableness`, `extraversion`, `nueroticism`, `openess_to_experience`.

Sử dụng K-fold Cross Validation với k=5 để tìm ra đặc trưng tốt nhất trong bộ các đặc trưng. Lần lượt thử nghiệm từng đặc trưng trên các bộ và tính trung bình thì thu được bảng trung bình sai số tuyệt đối trung bình trên các bộ mẫu cho các mô hình như sau:

STT	Mô hình với 1 đặc trưng	MAE
1	`conscientiousness`	`305710.169433`
2	`agreeableness`	`300025.897224`
3	`extraversion`	`306243.429954`
4	`nueroticism`	`298976.356108`
5	`openess_to_experience`	`302827.764004`

Từ bảng kết quả trên, ta thấy được giá trị trung bình MAE của đặc trưng `nueroticism` là nhỏ nhất. $\Longrightarrow$ Đây là đặc trưng ảnh hưởng/liên quan nhiều nhất đến mức lương của các kỹ sư trong số 5 đặc trưng trên.

Chọn đặc trưng `nueroticism` để huấn luyện trên bộ dữ liệu `train` và kiểm tra trên bộ dữ liệu `test` ta được kết quả sau:

MAE: `291019.693226953`
params: `[-56546.304]`

Công thức hồi quy:

$$\textbf{Salary} = \texttt{-56546.304} \times nueroticism$$

4.1.3 Phân tích ảnh hưởng của đặc trưng ngoại ngữ, lô-gic, định lượng đến mức lương của các kỹ sư dựa trên điểm các bài kiểm tra của AMCAT

Bộ đặc trưng ngoại ngữ, lô-gic, định lượng bao gồm: `English, Logical, Quant`.

Sử dụng K-fold Cross Validation với k=5 để tìm ra đặc trưng tốt nhất trong bộ các đặc trưng. Lần lượt thử nghiệm từng đặc trưng trên các bộ và tính trung bình thì thu được bảng trung bình sai số tuyệt đối trung bình trên các bộ mẫu cho các mô hình như sau:

STT	Mô hình với 1 đặc trưng	MAE
1	`English`	`121651.036399`
2	`Logical`	`120331.163039`
3	`Quant`	`117881.028835`

Từ bảng kết quả trên, ta thấy được giá trị trung bình MAE của đặc trưng định lượng (`Quant`) là nhỏ nhất. $\Longrightarrow$ Đây là đặc trưng ảnh hưởng/liên quan nhiều nhất đến mức lương của các kỹ sư trong số 3 đặc trưng trên.

Chọn đặc trưng `nueroticism` để huấn luyện trên bộ dữ liệu `train` và kiểm tra trên bộ dữ liệu `test` ta được kết quả sau:

MAE: `106819.5776198967`
params: `[585.895]`

Công thức hồi quy:

$$\textbf{Salary} = 585.895 \times Quant$$

4.1.4 Tự xây dựng mô hình, tìm mô hình cho kết quả tốt nhất

Xây dựng 3 mô hình bao gồm:

- Mô hình 1: Sử dụng 16 đặc trưng (`10percentage, 12percentage, collegeGPA, English, Logical, Quant, ComputerProgramming, ElectronicsAndSemicon, ComputerScience, MechanicalEngg, ElectricalEngg, TelecomEngg, CivilEngg, conscientiousness, nueroticism, openess_to_experience`)
- Mô hình 2: Sử dụng 8 đặc trưng (`10percentage, 12percentage, collegeGPA, English, Logical, Quant, Domain, ComputerProgramming`)
- Mô hình 3: Sử dụng 9 đặc trưng (`10percentage, 12percentage, Quant, Domain, CivilEngg, ComputerProgramming, ElectronicsAndSemicon, ComputerScience, ElectricalEngg`)

Sử dụng K-fold Cross Validation với k=5 để tìm ra mô hình tốt nhất trong các mô hình. Lần lượt thử nghiệm từng mô hình trên các bộ và tính trung bình thì thu được bảng trung bình sai số tuyệt đối trung bình trên các bộ mẫu cho các mô hình như sau:

STT	Mô hình với 1 đặc trưng	MAE
1	`Mô hình 1`	`112355.820905`
2	`Mô hình 2`	`114296.115694`
3	`Mô hình 3`	`111408.234551`

Từ bảng kết quả trên, ta thấy được giá trị trung bình MAE của mô hình sử dụng 9 đặc trưng (`10percentage, 12percentage, Quant, Domain, CivilEngg, ComputerProgramming, ElectronicsA`

`ComputerScience, ElectricalEngg)` là nhỏ nhất. $\Longrightarrow$ Đây là mô hình tốt nhất để dự đoán mức lương của các kỹ sư trong số 3 mô hình.

Chọn mô hình trên để huấn luyện trên bộ dữ liệu `train` và kiểm tra trên bộ dữ liệu `test` ta được kết quả sau:

MAE: `101971.5226723628`
params: `[ 1182.019, 856.921, 248.862, 32146.875, 92.867, -69.901, -169.944,`

Công thức hồi quy:

$$\textbf{Salary} = 1182.019 \times \textit{10percentage} + 856.921 \times \textit{12percentage} + 248.862 \times \textit{Quant}$$
$$+32146.875 \times \textit{Domain} + 92.867 \times \textit{ComputerProgramming} - 69.901 \times \textit{ElectronicsAndSemicon}$$
$$-169.944 \times \textit{ComputerScience} - 143.188 \times \textit{ElectricalEngg} + 133.199 \times \textit{CivilEngg}$$

4.2 Nhận xét

Ở kết quả phân tích tại mục **4.1.2**, trung bình sai số tuyệt đối trung bình của hai đặc trưng `agreeableness` và `neuroticism` là thấp hơn so với các đặc trưng còn lại. Vẫn có khả năng xuất hiện MAE trung bình của `agreeableness` là nhỏ nhất tùy thuộc vào sự phân phối đều ngẫu nhiên của bộ dữ liệu sau khi xáo trộn, tuy nhiên xác suất xuất hiện là rất thấp. Đã thực hiện thử nghiệm 15 lần chạy phân tích, kết quả `agreeableness` chỉ xuất hiện 1 lần.

Tuy nhiên, các đặc trưng tính tính cách không ảnh hưởng quá nhiều đến mức lương của các kỹ sư. Do trung bình MAE của nó lớn hơn gấp đôi MAE của mô hình **4.1.1** cũng như mô hình đặc trưng định lượng **4.1.3**.

Không xét đến các mô hình tự xây dựng thì mô hình dựa trên 11 đặc trưng đầu tiên đề bài cung cấp (**4.1.1**) mang lại kết quả tốt nhất, ảnh hương nhiều nhất đến mức lương của các kỹ sư do độ lỗi **MAE** là nhỏ nhất.

Tuy nhiên, so với MAE của mô hình ở câu **4.1.1** thì độ lỗi của mô hình ở câu **4.1.3** cũng được xem là tốt mà chỉ cần sử dụng một đặc trưng duy nhất là `Quant` thay vì bộ 11 đặc trưng. Từ đó có thể kết luận rằng `Quant` là một đặc trưng mạnh, ảnh hưởng rất lớn đến mức lương của các kỹ sư.

Mô hình được chọn ở **4.1.4** có độ lỗi **MAE** bé nhất trong các mô hình được thể hiện trong đồ án này.

5 Giả thuyết

Thực hiện scatter và plot[5] mô hình bằng thư viện `matplotlib` trên bộ dữ liệu `test` lên biểu đồ.

5.1 `best_personality_feature_model`

5 đặc trưng tính cách được phân tích thuộc mô hình **Big Five personality traits**[6]. Lewis Goldberg là người tiên phong trong ý tưởng rút gọn số lượng tính cách còn 5 nhóm chính từ 16 tính cách trong danh sách của Raymond Cattell. Nghiên cứu của ông được mở rộng bởi McCrae & Costa – những người đã xác minh tính chính xác và cung cấp mô hình nền tảng của bài kiểm tra Big Five hiện nay.[12] 5 đặc điểm tính cách này bao gồm:

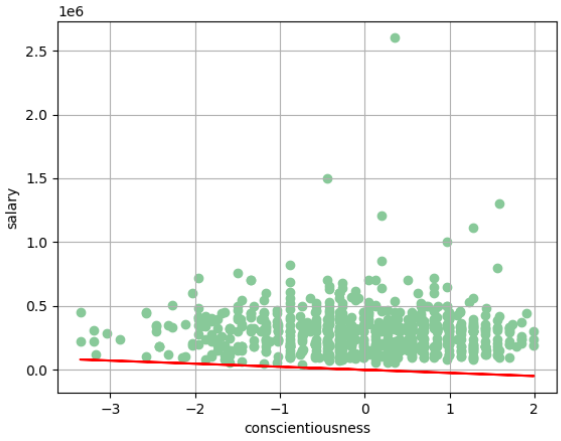
- Dễ chịu (Agreeableness)
- Tự chủ, tận tâm (Conscientiousness)
- Nhạy cảm, bất ổn cảm xúc (Neuroticism)
- Hướng ngoại (Extraversion)
- Cởi mở (Openness)

`best_personality_feature_model` là mô hình được huấn luyện dựa trên đặc trưng được chọn là `nueroticism` (sự bất ổn cảm xúc) sau khi kiểm tra trên 5 mẫu sử dụng phương pháp **K-Fold Cross Validation**. Tính cách này được xem là một đặc điểm mô tả sự ổn định cảm xúc tổng thể của một cá nhân. Sự bất ổn cảm xúc cho thấy khả năng cân bằng cảm xúc của một cá nhân qua cách họ nhìn nhận thế giới. Khía cạnh này cũng sẽ phản ánh xu hướng trải nghiệm cảm xúc tiêu cực của người đó. Người với điểm bất ổn cảm xúc cao sẽ thường lo âu, bất an và tự ti. Họ cũng dễ mất bình tĩnh trong tình huống hỗn loạn. Nhóm người này có nguy cơ đối mặt với trầm cảm và các chứng rối loạn tâm lý cao hơn. Người với điểm bất ổn cảm xúc thấp sẽ bình tĩnh, an toàn và hài lòng với bản thân nhiều hơn. Họ là người có tâm lý vững vàng, chịu áp lực tốt và có độ tự tin cao.[12]

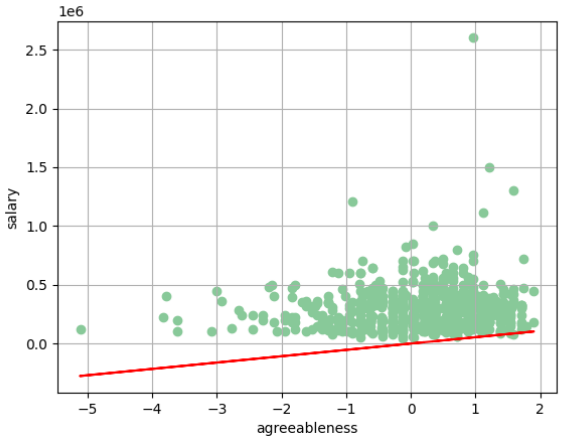
Hay hiểu đơn giản là: nếu bạn có chỉ số cao về đặc điểm tính cách này, bạn có khuynh hướng buồn bã, ủ rũ và cảm xúc không ổn định (cáu kỉnh) hay suy nghĩ tiêu cực. Ngược lại, bạn có điểm số thấp về đặc điểm này được coi là ổn định, kiên cường hơn về mặt cảm xúc.

Một thống kê trên The Ascent về chủ đề tương tự với mức thu nhập, người viết nhận định về đặc trưng bất ổn cảm xúc như sau: *"Những người có thể chịu sự bất ổn cảm xúc ở mức độ thấp có mức lương trung bình $46.200, còn những người có thể chịu sự bất ổn cảm xúc ở mức độ cao có mức lương trung bình $55.200. Những người kiếm được nhiều tiền nhất là những người có thể kiểm soát, cân bằng cảm xúc tiêu cực."*[11]

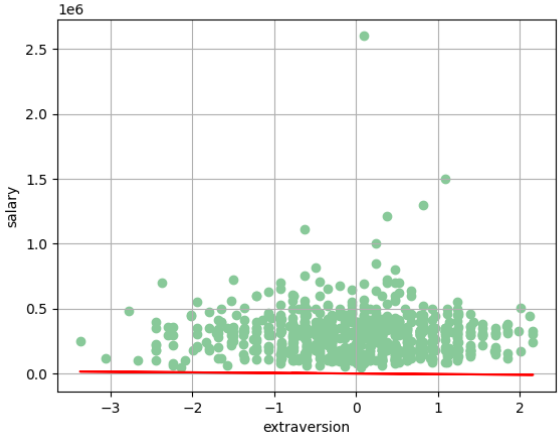
Ngoài ra, trong một nghiên cứu của KSE (Kyiv School of Economics) về mối quan hệ giữa tính cách và mức lương ở các nước Anh, Đức và Hà Lan, tác giả đã chỉ ra rằng: *"Mặc dù các đặc điểm tính cách khác ảnh hưởng đến tiền lương một cách khác nhau ở các quốc gia khác nhau nhưng hai đặc điểm (**`agreeableness`** và **`neuroticism`**) đều ảnh hưởng nhiều đến tiền lương ở tất cả các quốc gia nơi nghiên cứu được tiến hành. Các đặc điểm tính cách khác cũng ảnh hưởng đến tiền lương mặc dù không cho thấy sự thống nhất ở tất cả các quốc gia như hai đặc điểm này. Tình hình trên là hợp lý với yếu tố **`neuroticism`**. Nhưng **`agreeableness`** thì sao? Tất cả chúng ta đều muốn làm việc với những người hợp tác và không ích kỷ. Đây là một tính cách tốt, xứng đáng được tôn trọng. Nhưng không có nghĩa rằng bạn sẽ tự động được khen thưởng tài chính cho việc đó."*[15] Phát biểu trên khớp với kết quả phân tích ở mục **4.1.2**, tuy nhiên `neuroticism` được chọn bởi như quan điểm của tác giả, tính cách kia chỉ nên được xem



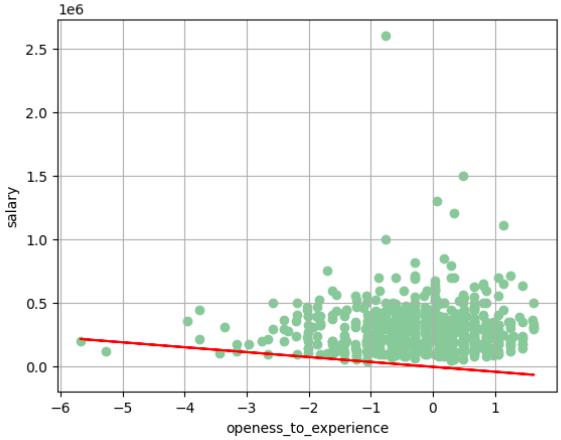
(a) Đặc trưng `conscientiousness`



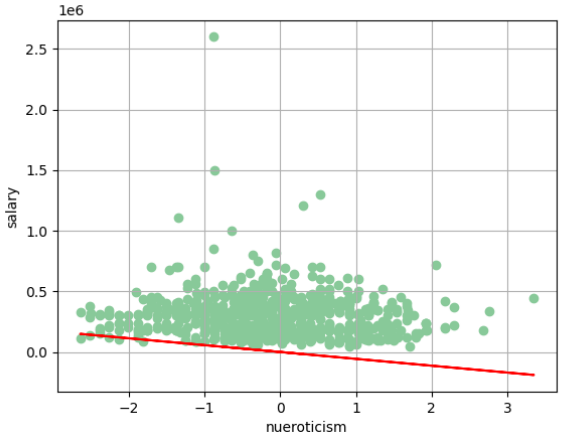
(b) Đặc trưng `agreeableness`



(c) Đặc trưng `extraversion`

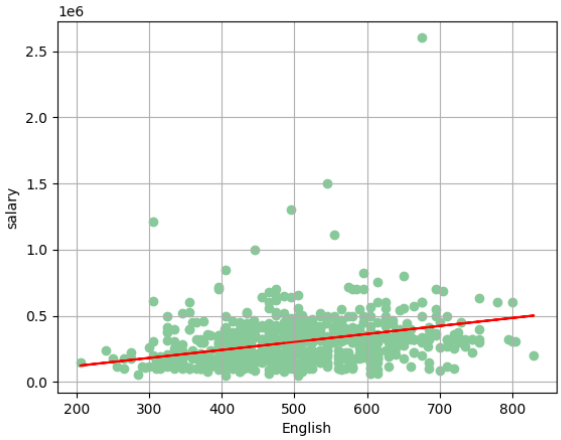


(d) Đặc trưng `openess_to_experience`

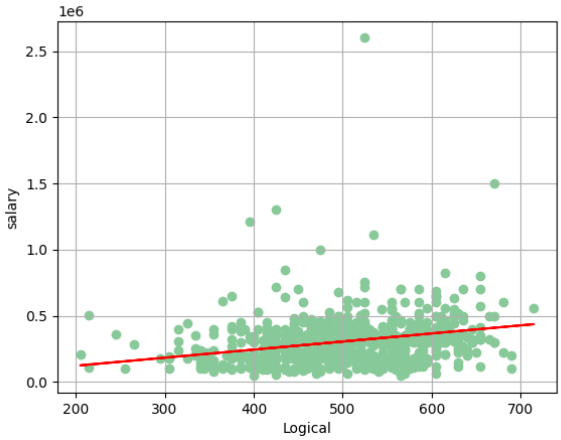


(e) Đặc trưng `nueroticism`

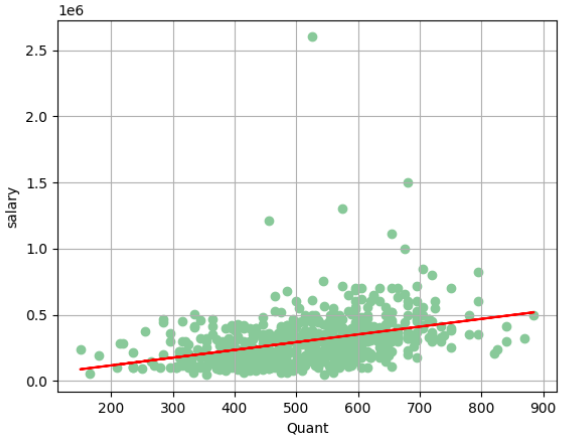
như một điểm cộng (có thể xét trong việc thăng tiến,...) chứ không thật sự liên quan hay ảnh hưởng quá nhiều đến mức lương.



(a) Đặc trưng `English`



(b) Đặc trưng `Logical`



(c) Đặc trưng `Quant`

5.2 `best_skill_feature_model`

Trong 3 đặc trưng, `best_skill_feature_model` là mô hình được huấn luyện dựa trên đặc trưng `Quant` (kỹ năng định lượng) sau khi kiểm tra trên 5 mẫu sử dụng phương pháp **K-Fold Cross Validation**. Kỹ năng định lượng tập trung vào việc sử dụng dữ liệu số để hiểu một chủ đề, giải quyết một vấn đề hoặc đưa ra đề xuất. Trong khi một số kỹ năng định lượng yêu cầu tính toán toán học, những kỹ năng khác liên quan đến việc đưa các con số vào bối cảnh thực tế. Dưới đây là một số kỹ năng định lượng quan trọng[14]:

- Kỹ năng nghiên cứu
- Tính toán toán học và lập luận
- Mô hình định lượng
- Kỹ năng phân tích
- Kỹ năng khảo sát

Đây là kỹ năng quan trọng đối với nhiều ngành nghề. Ví dụ như tài chính, khoa học, chuỗi cung ứng, ... và đặc biệt là nhóm ngành công nghệ (lập trình máy tính và thống kê, ...). Như bài viết trên của trang chuyên phân tích việc làm, nghề nghiệp **Indeed**[14], nhóm tác giả chỉ ra

rằng: *"Kỹ năng định lượng có thể giúp các lập trình viên máy tính, nhà phát triển và lập trình viên tạo ra các sản phẩm mới và cải thiện các sản phẩm hiện có. Các lập trình viên thường sử dụng toán học nhị phân, đại số và các khái niệm thống kê trong công việc của họ. Ngoài ra, việc xây dựng các kỹ năng định lượng có thể cải thiện khả năng giải quyết vấn đề của lập trình viên và đáp ứng nhanh chóng với việc thay đổi mục tiêu dự án. Tuỳ thuộc vào dự án họ đang làm việc, các lập trình viên và nhà phát triển có thể tiến hành nghiên cứu định lượng để quyết định các tính năng và chức năng của chương trình. Họ thường sử dụng các kỹ thuật hồi quy thống kê để làm nổi bật các xu hướng dữ liệu và rút ra kết luận."*

Trong một bài viết trên **NCESC.com**[13], tác giả chỉ ra những lợi ích khi một cá nhân có kỹ năng định lượng cao là như sau:

"

- *Tăng cơ hội nghề nghiệp và tiềm năng thu nhập cao hơn.*
- *Có khả năng đưa ra quyết định sáng suốt dựa trên dữ liệu.*
- *Có khả năng xác định các mẫu và xu hướng trong dữ liệu, cho phép các cá nhân tạo ra các mô hình dự đoán.*
- *Trở thành một tài sản có giá trị cho các tổ chức bằng cách phân tích dữ liệu để cải thiện hiệu suất và hiệu quả.*

"

Vì vậy có thể kết luận khả năng định lượng rất quan trọng và có ảnh hưởng đến mức lương, đặc biệt khi xét đến nhóm ngành công nghệ.

5.3 `my_best_model`

Mô hình trên chủ yếu sử dụng các đặc trưng liên quan tới điểm số học tập và điểm chuyên môn để đánh giá[7].

Mô hình sử dụng `10percentage` và `12percentage` tương ứng với điểm số lớp 10 và lớp 12, tức giai đoạn thi cấp 3 và thi đại học, 2 giai đoạn có nhiều ảnh hưởng tới con đường học vấn của một người, quyết định đến bằng cấp và hiển nhiên là có quyết định đến con đường phát triển sự nghiệp. Điểm số chuyên ngành thể hiện được thế mạnh của bản thân, những gì về chuyên môn, kỹ năng cứng của nhân lực. Tuy nhiên, điểm số có thể nói lên rất nhiều nhưng không phải là tất cả, bởi những kĩ năng mềm cũng đóng góp vai trò không kém.

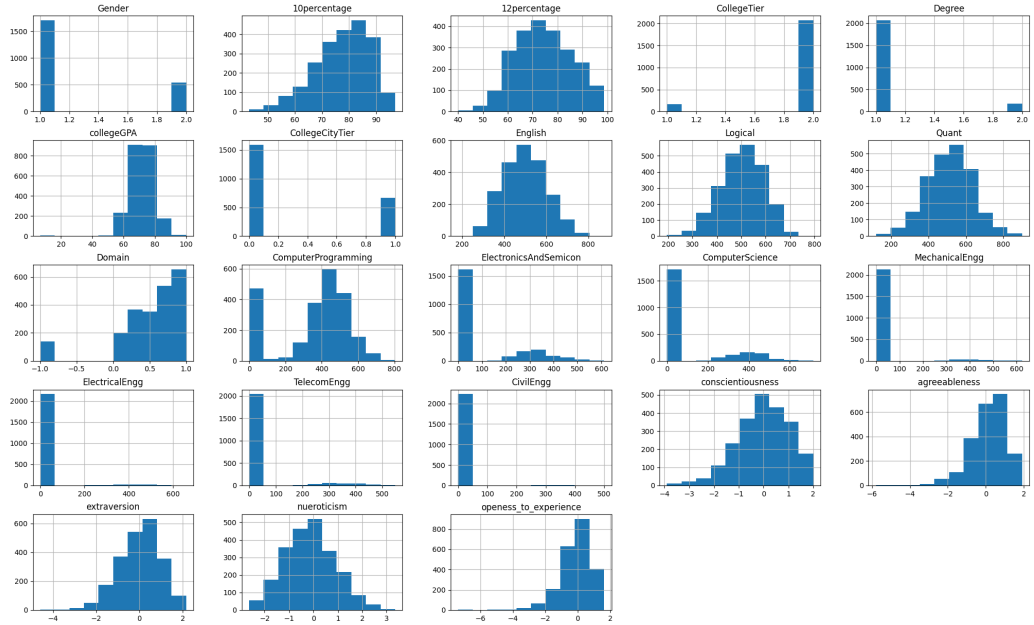
Chính vì vậy, mô hình sử dụng thêm đặc trưng được tìm kiếm ở trên mà liên quan mật thiết đến môi trường thực tế hơn: `Quant`, khả năng định lượng giúp giải quyết nhiều vấn đề tính toán thực tiễn ứng dụng trong nhiều ngành nghề, lĩnh vực đặc biệt là Công nghệ Thông tin.

6 Quá trình tìm mô hình tốt nhất

6.1 Mô hình 1

Dùng phương pháp đơn giản nhất để xử lý constant features[8][9] (chỉ hiển thị một giá trị cho tất cả quan sát trong tập dữ liệu). Ta sẽ đặt ra một ngưỡng (threshold) cho phương sai, features nào không đáp ứng ngưỡng đó thì sẽ bị loại bỏ. Nó sẽ trả ra một dãy bool, ta chỉ cần quan sát vào đó sẽ biết những feature nào không thỏa mãn điều kiện.

Tạo một đối tượng `Variancethreshold` với ngưỡng phương sai là `threshold=1` (chỉ các biến phương sai lớn hơn 1 được chọn). Gọi `fit_transform()` và truyền vào tập dữ liệu huấn luyện để chọn các đặc trưng.



6.2 Mô hình 2

Sử dụng phương pháp định lượng mối quan hệ giữa hai biến sử dụng hệ số tương quan **Pearson**[10], đo lường sự liên kết tuyến tính giữa hai biến. Hệ số này luôn luôn có giá trị giữa -1 và 1 trong đó:

- -1: chỉ ra một tương quan tuyến tính hoàn toàn tiêu cực
- 0: chỉ ra không có tương quan tuyến tính
- 1: chỉ ra một tương quan tuyến tính hoàn toàn tích cực

Hay có thể hiểu là hệ số tương quan càng gần bằng 1 thì mối liên hệ giữa hai biến càng chặt chẽ. Chạy phân tích ta được bảng hệ số tương quan giữa các đặc trưng với mức lương là như sau:

Sau khi sắp xếp giảm dần hệ số tương quan của đặc trưng, thực hiện lọc và chỉ lấy các đặc trưng có hệ số tương quan từ ngưỡng `0.1` trở lên. Từ đó được 8 đặc trưng đầu như trên bảng.

	Feature	Coef		Feature	Coef
1	Quant	0.205358	12	CollegeCityTier	0.004575
2	Logical	0.188416	13	extraversion	-0.002661
3	English	0.169293	14	openess_to_experience	-0.007814
4	10percentage	0.155174	15	ElectronicsAndSemicon	-0.009292
5	12percentage	0.149531	16	Degree	-0.017602
6	ComputerProgramming	0.125866	17	Gender	-0.036183
7	collegeGPA	0.122469	18	TelecomEngg	-0.040415
8	Domain	0.122022	19	ElectricalEngg	-0.041217
9	agreeableness	0.068623	20	conscientiousness	-0.057699
10	MechanicalEngg	0.028854	21	nueroticism	-0.073401
11	CivilEngg	0.01615	22	ComputerScience	-0.095507
			23	CollegeTier	-0.174824

6.3 Mô hình 3

Thực hiện scatter, plot biểu đồ lên và nhận xét độ phân tán thì lọc ra được các đặc trưng sau có khả năng liên quan đến mức lương: `10percentage , 12percentage, collegeGPA, Quant, Domain, ComputerProgramming, ElectronicsAndSemicon, ComputerScience, MechanicalEngg, ElectricalEngg, TelecomEngg, CivilEngg, nueroticism` (lấy đặc trưng `Quant` và `nueroticism` từ **1b, 1c** và các điểm số đánh giá chuyên môn).

Chương trình thực hiện vét cạn với ý tưởng truy xuất tổ hợp dãy nhị phân từ toàn 0 thành toàn 1, mỗi ký tự đại diện cho việc có tiến hành chọn đặc trưng đó hay không. Tiến hành lưu tổ hợp có MAE bé hơn min và set min về giá trị bé nhất đó.

Tài liệu

[1] URL: https://tinyurl.com/bwxehjtp.

[2] URL: https://tinyurl.com/bxkethtj.

[3] URL: https://tinyurl.com/4wf7tppj.

[4] URL: https://tinyurl.com/3kkb3frp.

[5] URL: https://www.geeksforgeeks.org/python-linear-regression-using-sklearn/.

[6] URL: https://en.wikipedia.org/wiki/Big_Five_personality_traits.

[7] URL: https://tinyurl.com/y3brsjnj.

[8] URL: https://web888.vn/phuong-phap-lua-chon-feature-trong-machine-learning/.

[9] URL: https://tinyurl.com/2a2dn5fb.

[10] URL: https://tinyurl.com/2p8zvjy3.

[11] Lyle Daly. "Here's How Your Personality Type May Affect Your Income". **in** *The Ascent*: (2021. Ngày truy cập: 20/08/2023). URL: https://www.fool.com/the-ascent/banks/articles/heres-how-your-personality-type-may-affect-your-income/.

[12] Hà Phạm. "Big Five: Trắc nghiệm tính cách biết thế mạnh và hạn chế của bạn?" **in** *Vietcetera*: (2020. Ngày truy cập: 19/08/2023). URL: https://vietcetera.com/vn/big-five-trac-nghiem-tinh-cach-biet-the-manh-va-han-che-cua-ban.

[13] Carmen Smith. "What Are Quantitative Skills?" **in** *NCESC.com*: (Ngày truy cập: 21/08/2023). URL: https://tinyurl.com/4wwe4ce9.

[14] Indeed Editorial Team. "FAQ: What Are Quantitative Skills?" **in** *Indeed*: (2022. Ngày truy cập: 21/08/2023). URL: https://www.indeed.com/career-advice/career-development/quantitative-skills.

[15] Ph.D. Olha Verkhohliad. "Relationship between Your Personality and Your Salary Level". **in** *KSE*: (2019. Ngày truy cập: 20/08/2023). URL: https://kse.ua/kse-research/relationship-between-your-personality-and-your-salary-level/.