ĐẠI HỌC QUỐC GIA TPHCM

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN

Project 03: Linear Regression

Môn học: Toán ứng dụng và thống kê cho CNTT

Sinh viên thực hiện:

Lý Anh Quân (22127344)

Giáo viên hướng dẫn:

ThS. Vũ Quốc Hoàng

ThS. Phan Thị Phương Uyên

ThS. Nguyễn Văn Quang Huy

ThS. Nguyễn Ngọc Toàn

Ngày 17 tháng 8 năm 2024

Mục lục

1 Chi tiết mã nguồn chương trình

1.1 Thư viện sử dụng

- **NumPy**: Thư viện Python hỗ trợ tính toán khoa học và toán học. Nó cung cấp một cấu trúc dữ liệu mảng đa chiều (ndarray) hiệu quả và các hàm toán học để thực hiện các phép toán trên mảng đó. Chúng ta có thể sử dụng các công cụ mạnh mẽ để thực hiện các phép toán ma trận, tính toán vector hóa và xử lý nhanh chóng các mảng số.

- **Seaborn**: Thư viện Seaborn của Python dựa trên matplotlib được sử dụng để trực quan hóa dữ liệu thống kê của bộ dữ liệu. Thư viện cung cấp một số loại đồ thị để biểu diễn phù hợp với bộ dữ liệu như heatmap cho đồ thị tương quan (Correlation Heatmap) hay pairplot.

- **matplotlib**: Sử dụng cho việc trực quan hóa dữ liệu, vẽ biểu đồ tương quan. Một số hàm được sử dụng như: plt.subplots(), plt.show()

- **pandas:** Thư viện Pandas hỗ trợ cung cấp cấu trúc dữ liệu và công cụ phân tích dữ liệu mạnh mẽ. Pandas giúp xử lý và làm việc với dữ liệu dạng bảng (như dữ liệu từ file CSV) dễ dàng và ta sẽ dùng pd.read_csv để đọc vào bộ dữ liệu. Thư viện còn cung cấp đối tượng DataFrame, cho phép bạn thực hiện các thao tác như lọc dữ liệu, ghép nối dữ liệu từ nhiều nguồn, và truy vấn dữ liệu theo các điều kiện. Khi xây dựng mô hình hồi quy, Pandas thường được sử dụng để nạp dữ liệu, thực hiện các phép biến đổi dữ liệu và chuẩn bị dữ liệu cho mô hình.

- **from scipy import stats:** SciPy cung cấp khá nhiều module tính toán và ở đây chúng ta sẽ dùng đến phân phối xác suất cũng như tính giá trị p_value

- **math:** Để làm tròn số thực

1.2 Mô tả hàm và thuật toán

1.2.1 Class OLSLinearRegression:

Class này sẽ cung cấp các phương thức để thực hiện việc xây dựng mô hình hồi quy tuyến tính, trả về hệ số của các đặc trưng có trong mô hình được xây dựng và cuối cùng dựa vào mô hình vừa xây dựng để thực hiện việc dự đoán thành tích học tập.

- Phương thức fit() của class OLSLinearRegression:
  - INPUT: mảng X chứa các giá trị đặc trưng (feature), mảng y chứa các giá trị mục tiêu (target).
  - OUTPUT: trả về chính đối tượng mô hình, bao gồm mảng chứa các trọng số của từng feature trong mô hình
  - Mô hình tính toán ma trận ngược đảo của X nhân với chính nó chuyển vị (X.T @ X), sau đó nhân với chuyển vị của X (X.T). Kết quả cuối cùng là ma trận pseudo-inverse của X (hoặc còn gọi là ma trận giả nghịch đảo). Tiếp theo, mô hình tính toán vector trọng số self.w bằng cách nhân ma trận giả nghịch đảo này với mảng y.
- Phương thức get_params() của class OLSLinearRegression
  - OUTPUT: trả về mảng chứa các trọng số (self.w) của từng feature trong mô hình được huấn luyện
- Phương thức predict() của class OLSLinearRegression
  - INPUT: mảng X chứa các giá trị đặc trưng (feature).
  - OUTPUT: Trả về mảng y chứa các giá trị dự đoán [6]

1.2.2 Class k_folds_cross_validation:

Class sẽ hỗ trợ các phương thức để thực hiện kĩ thuật K-fold Cross Validation, một thuật toán hỗ trợ đánh giá xem mô hình nào sẽ tốt hơn.

- **Kỹ thuật k_folds_cross_validation** [3]
  1. Thực hiện xáo trộn dữ liệu (shuffle data)
  2. Chia dữ liệu thành k fold có kích thước bằng nhau (ở đây k = 5)
  3. Thực hiện xét qua từng fold, với mỗi fold sẽ thực hiện huấn luyện nó trên tất cả các mô hình để tính giá trị mae tương ứng.
  4. Tính giá trị mae trung bình của mỗi mô hình sau khi được huấn luyện qua k fold.

5. Dựa vào giá trị mae trung bình để tìm ra mô hình tốt nhất. Mô hình tốt nhất là mô hình có giá trị mae trung bình nhỏ nhất.

- Phương thức init() của class k_folds_cross_validation
  - Input: Nhận vào số lượng k, các đặc trưng sẽ dùng và các mô hình
  - Thực hiện gán các input trên vào các thuộc tính của class
- Phương thức shuffle_data của class k_folds_cross_validation
  - Cố định seed trước khi shuffle. Sau đó chuyển dữ liệu từ kiểu DataFrame sang array để thực hiện xáo trộn dữ liệu bằng hàm `numpy.random.shuffle()`. Chuyển dữ liệu đã xáo trộn lại thành Dataframe.
- Phương thức split_to_k_folds của class k_folds_cross_validation
  - Chia dữ liệu đã shuffle thành k fold bằng nhau bằng kỹ thuật list comprehension và lưu vào list self.folds.
- Phương thức cross_validation của class k_folds_cross_validation
  - Với mỗi fold, ta sẽ thực hiện đem fold đó đi huấn luyện trên tất cả mô hình và tính giá trị mae tương ứng. Lưu các giá trị mae vào 1 list sau đó tính giá trị mae trung bình của từng mô hình sau khi được huấn luyện trên k fold. fold
- Phương thức best_model của class k_folds_cross_validation
  - Trả về kết quả mae trung bình của từng mô hình sau khi thực hiện kĩ thuật k-folds cross validation và mô hình được xem là tốt nhất.

1.2.3 Các hàm hỗ trợ

- Hàm mae() [6]
  - Input: mảng chứa các giá trị dự đoán thực sự trên tập **y_test** và mảng chứa các giá trị dự đoán từ mô hình **y_hat**

  - Output: Giá trị MAE – Mean Absolute Error (Độ lỗi tuyệt đối trung bình) được tính theo công thức

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

  - Thực hiện: Chuyển hai vector dataframe y và y_hat thành numpy array
  - Sử dụng hàm ravel() của numpy để flatten hai numpy array vừa thu được và tính hiệu giữa chúng
  - Sử dụng np.abs() để tính trị tuyệt đối của hiệu vừa thu được. Sau đó tính trung bình bằng hàm np.mean().

- hàm preprocess() [6]
  - Input: ma trận X dataframe
  - Output: ma trận X sau khi đã được thêm cột 1 vào phía trước
  - Sử dụng hàm np.ones() để tạo ra ma trận toàn 1 với số dòng bằng số dòng của ma trận X và số cột là 1.
  - Sau đó dùng np.hstack để nối ma trận toàn 1 với ma trận X

- Các phương thức để hiển thị đồ thị:
  - (Pandas) .head(): hiển thị các dòng đầu của bộ dữ liệu
  - (Pandas) .describe(): đưa ra một số thống kê đơn giản như count, mean, std, min, max
  - (Pandas) .value_counts(): số lần xuất hiện của các phần tử
  - (Pandas) .corr(): Hệ số tương quan giữa các cột
  - (Pandas) .hist(): Biểu đồ tần suất để cho thấy sự thay đổi, biến động của dữ liệu
  - (Pandas) .boxplot(): Biểu đồ hộp để thể hiện sự phân bố giá trị định lượng của nhiều nhóm dữ liệu.
  - (seaborn) .scatterplot(): Thể hiện biểu đồ phân tán giữa 2 giá trị
  - (seaborn) .pairplot(): Cho ta thấy tổng quan dữ liệu và mối tương quan giữa các chiều dữ liệu theo từng cặp với nhau

2 Báo cáo và nhận xét kết quả các mô hình

2.1 Yêu cầu 1: Phân tích khám phá dữ liệu

2.1.1 Tổng thể bộ dữ liệu

Để có cái nhìn nhanh về thống kê của mỗi trường thông tin dạng số, phương thức describe() có thể được sử dụng:

	Hours Studied	Previous Scores	Extracurricular Activities	Sleep Hours	Sample Question Papers Practiced	Performance Index
count	9000.000000	9000.000000	9000.000000	9000.000000	9000.000000	9000.000000
mean	4.976444	69.396111	0.493667	6.535556	4.590889	55.136333
std	2.594647	17.369957	0.499988	1.695533	2.864570	19.187669
min	1.000000	40.000000	0.000000	4.000000	0.000000	10.000000
25%	3.000000	54.000000	0.000000	5.000000	2.000000	40.000000
50%	5.000000	69.000000	0.000000	7.000000	5.000000	55.000000
75%	7.000000	85.000000	1.000000	8.000000	7.000000	70.000000
max	9.000000	99.000000	1.000000	9.000000	9.000000	100.000000

Hình 1: Enter Caption

- Hours Studied (Giờ học) có trung bình là khoảng 5 giờ, với phần lớn sinh viên học từ 3 đến 7 tiếng mỗi ngày. Giờ học là một yếu tố quan trọng có thể ảnh hưởng trực tiếp đến chỉ số thành tích.
- Previous Scores (Điểm số học sinh đạt được trong các bài kiểm tra trước đó).Previous Scores có trung bình là 69 điểm, với phân bố khá đều từ 40 đến 99 điểm. Đây là một yếu tố quan trọng phản ánh khả năng học tập của sinh viên trước đó.
- Nhìn vào đặc trưng Extracurricular Activities (tham gia hoạt động ngoại khóa), ta thấy được có sự phân bố đều khi khoảng một nửa sinh viên tham gia hoạt động ngoại khóa, một nửa còn lại thì không. Điều này có thể ảnh hưởng đến chỉ số thành tích thông qua các kỹ năng mềm mà sinh viên phát triển.
- Sleep hours (Giờ ngủ trung bình) của sinh viên là 6.5 giờ. Phần lớn sinh viên ngủ từ 5 đến 8 giờ, và đây là yếu tố có thể ảnh hưởng đến sự tập trung và hiệu suất học tập.
- Sample Question Papers Practice (Số bài kiểm tra mẫu đã làm) có trung bình là 4.6 bài, với phần lớn sinh viên luyện tập từ 2 đến 7 bài. Việc luyện tập bài mẫu nhiều có thể giúp sinh viên làm quen với cấu trúc đề thi, qua đó cải thiện chỉ số thành tích

- Performance Index (thành tích học tập) có trung bình là 55 điểm, với sự phân tán khá rộng từ 10 đến 100 điểm. Điều này cho thấy có sự khác biệt lớn giữa các sinh viên về thành tích học tập.

2.1.2 Mối tương quan giữa các đặc trưng

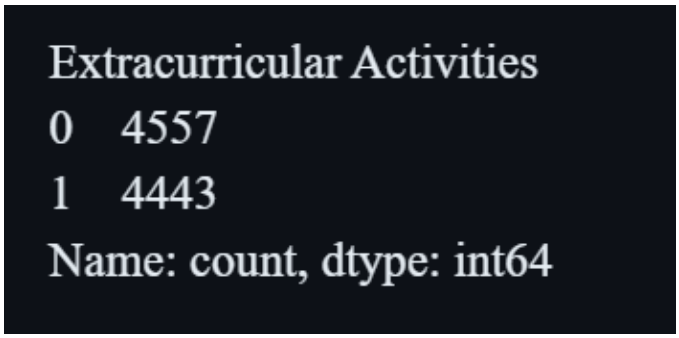
Chúng ta sẽ muốn biết đặc trưng nào có mối quan hệ tương quan nhất với thành tích học tập của sinh viên.

	Hours Studied	Previous Scores	Extracurricular Activities	Sleep Hours	Sample Question Papers Practiced	Performance Index
Hours Studied	1.000000	-0.018463	0.004511	-0.000694	0.015852	0.369148
Previous Scores	-0.018463	1.000000	0.009533	0.002802	0.006417	0.914775
Extracurricular Activities	0.004511	0.009533	1.000000	-0.020773	0.008199	0.025637
Sleep Hours	-0.000694	0.002802	-0.020773	1.000000	0.005054	0.043980
Sample Question Papers Practiced	0.015852	0.006417	0.008199	0.005054	1.000000	0.041088
Performance Index	0.369148	0.914775	0.025637	0.043980	0.041088	1.000000

Bảng 1: Correlation Matrix

Phân tích bảng cho thấy có mối tương quan tuyến tính dương giữa Performance Index và một số đặc trưng, bao gồm Hours Studied, Previous Scores, Extracurricular Activities, Sleep Hours, và Sample Question Papers Practiced, tất cả đều thể hiện xu hướng tăng lên. Đáng chú ý, đặc trưng Previous Scores thể hiện mối tương quan tuyến tính lớn với Performance Index, nhấn mạnh tầm quan trọng của các điểm số gần đây sinh viên đạt được trong việc ảnh hưởng đến hiệu suất học tập của sinh viên.

2.1.3 Chênh lệch giữa các giá trị



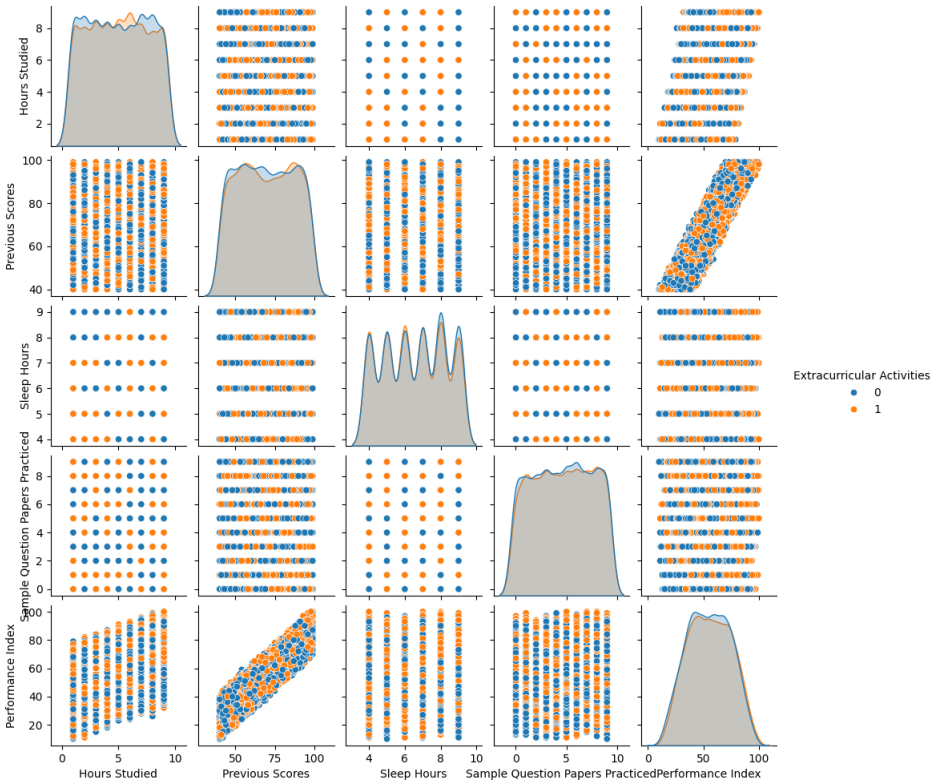
Hình 2: Chênh lệch theo số lượng

```
Extracurricular Activities
0    54.650647
1    55.634481
Name: Performance Index, dtype: float64
```

Hình 3: Chênh lệch theo Performance Index

Từ dữ liệu trên, ta có thể trong các đặc trưng thì đặc trưng Extracurricular Activities có sự chênh lệch giữa 2 giá trị (yes-no) là nhỏ.

2.1.4 Đặc trưng Extracurricular Activities kết hợp với các đặt trưng khác



Hình 4: Các biểu đồ với thống kê những người tham gia EA hoặc không

Vì đặc trưng Extracurricular Activities trước khi được làm sạch không phải là 1 biến giá trị số nên ta có thể kết hợp nó với các đặc trưng khác để làm rõ hơn về độ ảnh hưởng của việc tham gia hoạt động ngoại khóa. Dựa vào các đồ thị và kiến thức xác suất thống kê, ta có thể đặt ra các giả thuyết và phân tích như sau:

- H_0: Thành tích học tập trung bình của số sinh viên có tham gia Hoạt động ngoại khóa bằng với số sinh viên không tham gia.
- H_1: Thành tích học tập trung bình của số sinh viên tham gia Hoạt động ngoại khóa cao hơn số sinh viên không tham gia.

Với mức ý nghĩa (alpha) là 0,05

np.var(data_extracurricular), np.var(data_not_extracurricular)

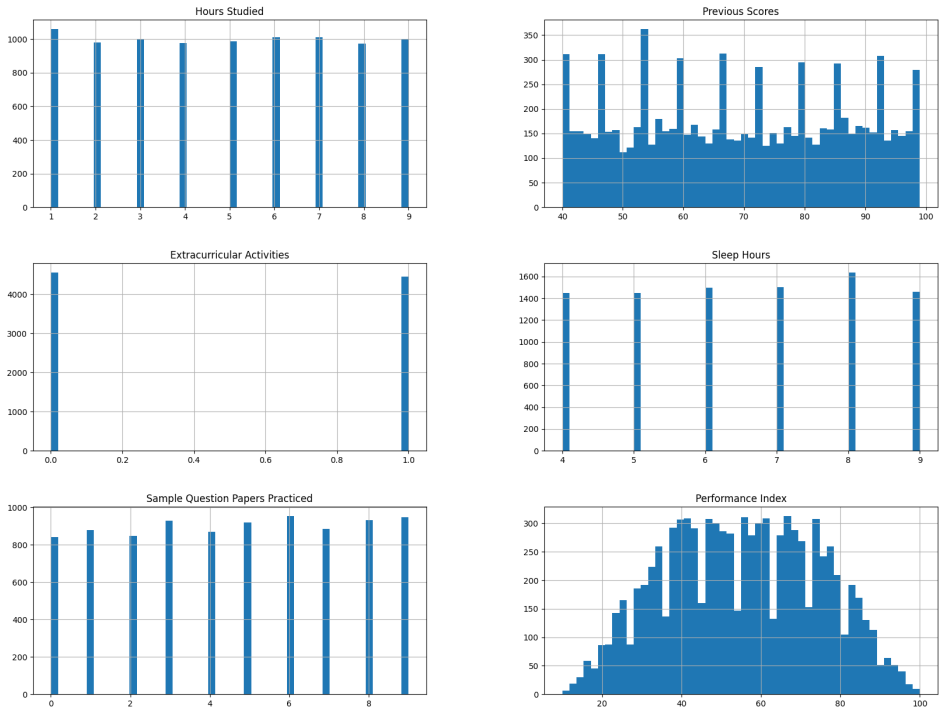
(370.874513922569, 366.5309625072948)

p-value = 0.007508665475760223

If the p-value 0.007508665475760223 is less than 0.05, then reject the null hypothesis

Từ kết quả trên, ta biết rằng giá trị p là 0.0075, do đó H_0 bị bác bỏ. Điều này có nghĩa là có đủ bằng chứng để kết luận rằng có sự khác biệt đáng kể giữa các nhóm được kiểm tra, cho thấy thành tích học tập trung bình của số sinh viên tham gia Hoạt động ngoại khóa cao hơn số sinh viên không tham gia.[2]

2.1.5 biểu đồ phân bố tần số



Hình 5: Biểu đồ histogram

- Histogram của số giờ học cho thấy một sự phân bố khá đồng đều trên các mức giá trị khác nhau, nghĩa là không có sự tập trung đáng kể ở bất kỳ một khoảng thời gian học nào. Điều này có thể ám chỉ rằng sinh viên có các thói quen học tập khác nhau, từ việc học ít đến học nhiều giờ. Sự đa dạng này có thể phản ánh rằng số giờ học không phải là yếu tố quyết định duy nhất đến kết quả học tập. Có thể có các yếu tố khác như chất lượng học tập, phương pháp học, hoặc các yếu tố khác đóng vai trò quan trọng hơn
- Ở biểu đồ Previous Scores cho thấy điểm số trước đây tập trung chủ yếu ở khoảng từ 50 đến 80. Điều này cho thấy phần lớn sinh viên có điểm số nằm ở mức trung bình, với ít sinh viên có điểm số rất cao hoặc rất thấp. Previous Scores có thể là một yếu tố dự đoán khá tốt cho kết quả học tập hiện tại. Những sinh viên có điểm số trung bình trước đó có thể duy trì mức thành tích tương tự, trong khi các sinh viên có điểm số cao có khả năng đạt kết quả tốt hơn
- Ở biểu đồ hoạt động ngoại khóa chỉ có 2 giá trị 0 và 1 tương ứng với có hoặc không tham gia. Số lượng của 2 cột này là sấp sỉ bằng nhau nhưng rất có thể khi kết hợp với các đặc trưng khác thì sẽ có kết quả khác nhau.
- Đa số sinh viên ngủ từ 5 đến 8 giờ mỗi đêm, điều này phù hợp với khuyến nghị về giấc ngủ cho người trưởng thành.
- Biểu đồ về số lượng làm đề thi thử phân bố khá đồng đều cho thấy sinh viên có mức độ làm đề thi thử khác nhau, từ rất ít đến rất nhiều. Số lượng làm đề thi thử có thể phản ánh mức độ chuẩn bị của sinh viên cho các kỳ thi. Những sinh viên làm nhiều đề thi thử có thể có khả năng đạt kết quả cao hơn do đã quen với cấu trúc và nội dung của kỳ thi.
- Chỉ số thành tích tập trung chủ yếu ở khoảng từ 20 đến 80, tương tự như phân bố của Previous Scores.

2.2 Yêu cầu 2a: Xây dựng mô hình sử dụng toàn bộ 5 đặc trưng đề bài cung cấp

- Yêu cầu:
  - Sử dụng toàn bộ 5 đặc trưng đề bài cung cấp: Hours Studied,Previous Scores, Extracurricular Activities, Sleep Hours, Sample Question Papers Practiced để xây dựng mô hình

hồi quy tuyến tính dự đoán các yếu tố ảnh hưởng đến thành tích học tập của sinh viên.

- Huấn luyện 1 lần duy nhất cho 5 đặc trưng trên cho toàn bộ tập huấn luyện (train.csv).
- Thể hiện công thức cho mô hình hồi quy theo 5 đặc trưng.
- Báo cáo kết quả MAE trên tập kiểm tra (test.csv) cho mô hình vừa huấn luyện được.

- Cách triển khai:
  - Mục tiêu chúng ta là cần tìm các vector trọng số ứng với các đặc trưng để hình thành công thức hồi quy tuyến tính. $StudentPerformance = w_0 + w_1 \times$ Hour Studied $+$ w_2 Previous Scores $+ w_3 \times$ Extracurricular Activities
    $+ w_4 \times$ Sleep Hours $+ w_5 \times$ Question Paper
  - Từ tập train ta sẽ lấy ra được x_train và sau đó dùng hàm **preprocess()** cho x_train để thêm một cột 1 vào trước x.
  - Sau đó ta tạo mô hình với 5 đặc trưng bằng cách **OLSLinearRegression().fit** X_train_pre và y_train
  - Sau khi đã có mô hình, chúng ta sẽ có được các trọng số bằng phương thức **get_params()** và dùng hàm round() để làm tròn đến 3 chữ số thập phân.
  - Với x_test_pre đã preprocess, ta tính giá trị dự đoán của tập kiểm tra dựa vào mô hình vừa có nhờ phương thức predict() của class OLSLinearRegression và từ đó tính giá trị mae (mean absolute error) của mô hình.

- Kết quả:
  - Công thức hồi quy của mô hình:

Student Performance $= -33.969 + 2.852 \times$ **Hour Studied** $+ 1.018 \times$ **Previous Scores** $+ 0.604 \times$ **Extracurricular Activities** $+ 0.474 \times$ **Sleep Hours** $+ 0.192 \times$ **Question Paper**

  - Giá trị **MAE** của mô hình:

$$\textbf{MAE} = 1.596$$

- Nhận xét:

  - Vì mô hình trên là mô hình sử dụng đủ các đặc trưng của bộ dữ liệu nên nó sẽ không đảm bảo về vấn đề chính xác và tối ưu. Nhưng với giá trị MAE khá nhỏ thì ta cũng thấy được là mô hình này cho kết quả khá tốt.

2.3 Yêu cầu 2b: Xây dựng mô hình sử dụng duy nhất 1 đặc trưng, tìm mô hình cho kết quả tốt nhất

K-fold Cross-validation (xác thực chéo hoặc kiểm chứng chéo) là khi chúng ta chia nhỏ dataset để kiểm chứng hiệu quả của mô hình trên từng phần riêng biệt. K đại diện cho số nhóm mà dữ liệu sẽ được chia ra

- Xáo trộn (shuffle) dataset một cách ngẫu nhiên
- 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
  - Đánh giá và sau đó hủy mô hình
- Tổng hợp hiệu quả của mô hình dựa từ các số liệu đánh giá

Lưu ý quan trọng là mỗi mẫu chỉ được gán cho duy nhất một nhóm và phải ở nguyên trong nhóm đó cho đến hết quá trình. Chỉ được thực hiện trên tập huấn luyện đã được chia chứ không được thực hiện trên toàn bộ dataset.

- **Cách triển khai:**
  - Lưu các đặc trưng sẽ sử dụng vào features_2b.
  - Với mỗi mô hình sẽ có đặc trưng nào thì sẽ lưu vào 1 list để quản lí, tất cả mô hình sẽ được lưu vào list models_2b.
  - Sử dụng các phương thức **shuffle_data()**, **split_to_k_folds**, **cross_validation()** của class k_folds_cross_validation để thực hiện xáo trộn dữ liệu, chia thành k fold như nhau và huấn luyện 5 mô hình trên từng fold.

- Cuối cùng tính giá trị mae trung bình của từng mô hình và tìm ra mô hình tốt nhất nhờ phương thức best_model().
- Có được mô hình tốt rồi thì ta sẽ làm lại như câu 2a đó chính là đem mô hình best_personality_feature_model đi huấn luyện trên toàn bộ tập train. Tìm ra được công thức hồi quy tuyến tính của mô hình tốt nhất và cuối cùng tính ra được giá trị mae của mô hình này. [6]

- **Kết quả:**

STT	Mô hình với 1 đặc trưng	MAE
1	Hours Studied	15.444
2	**Previous Scores**	**6.617**
3	Extracurricular Activities	16.181
4	Sleep Hours	16.177
5	Sample Question Papers Practiced	16.166

Bảng 2: MAE cho 5 mô hình từ K-fold Cross Validation

- **Nhận xét chung:**
  - Từ 5 kết quả mae trung bình có từ k-folds cross validation, ta có thể nhận ra được đặc trưng Previous Scores là đặc trưng ảnh hưởng nhất tới Performance Index của học sinh và kết quả trên phản ánh khá chính xác vì ta thấy chênh lệch giữa mae trung bình của 4 đặc trưng còn lại với Previous Scores là rất cao. Nhìn chung, những chỉ số mae này có thể coi là khá cao, kể cả là với đặc trưng cho mô hình tốt nhất là Previous Scores. Từ đó có thể nhận xét chung là những đặc trưng này chưa thể kết luận được sự ảnh hưởng đến thành tích của sinh viên. Vì thế mức độ ảnh hưởng của những đặc trưng này lên giá trị mục tiêu khi đứng riêng lẻ là chưa thực sự đáng tin.

- **Kết quả mô hình tốt nhất:**
  - Mô hình tốt nhất là mô hình với đặc trưng: Previous Scores
  - Công thức hồi quy:

$$\textbf{Student Performance} = -14.989 + 1.011 \times PreviousScrores$$

– Độ lỗi tuyệt đối trung bình MAE trên tập kiểm tra của mô hình tốt nhất ở câu 2b:

$$\textbf{MAE} = 6.544$$

- **Nhận xét và nêu giả thuyết:**

  – Kết quả mae của mô hình cho thấy mô hình này không phải là một mô hình tốt, nó có độ lỗi lớn hơn nhiều so với mô hình 5 đặc trưng ở câu 2a, từ đó ta có thể thấy được mô hình một đặc trưng cho ra kết quả không tốt như thế nào. Ngoài ra trọng số của Previous Scores còn cho thấy đây là một đặc trưng ảnh hưởng mạnh mẽ theo hướng tích cựcc đến thành tích học tập Performance Index.

  – Có rất nhiều giả thuyết để giải thích cho lý do trong 5 đặc trưng trên thì **Previous Scores** lại là đặc trưng có tầm ảnh hưởng lớn nhất đối với thành tích học tập của sinh viên.

  – Dựa vào công thức của mô hình trên, với trọng số của đặc trưng Previous Scores thì ta có thể thấy các sinh viên có số điểm những kỳ thi trước đó cao (hoặc thấp) thì học sẽ tiếp tục duy trì số điểm đó ở các kỳ thi trước. Có thể nói nó phản ánh tâm lý của một sinh viên, khi họ đạt điểm cao thì họ sẽ có hứng thú hơn với các kỳ thi sau để đạt được số điểm như vậy hoặc ngược lại khi họ có số điểm thấp ở các kỳ trước đó thì kỳ vọng của họ sẽ thấp hơn và ảnh hưởng đến thành tích học tập.

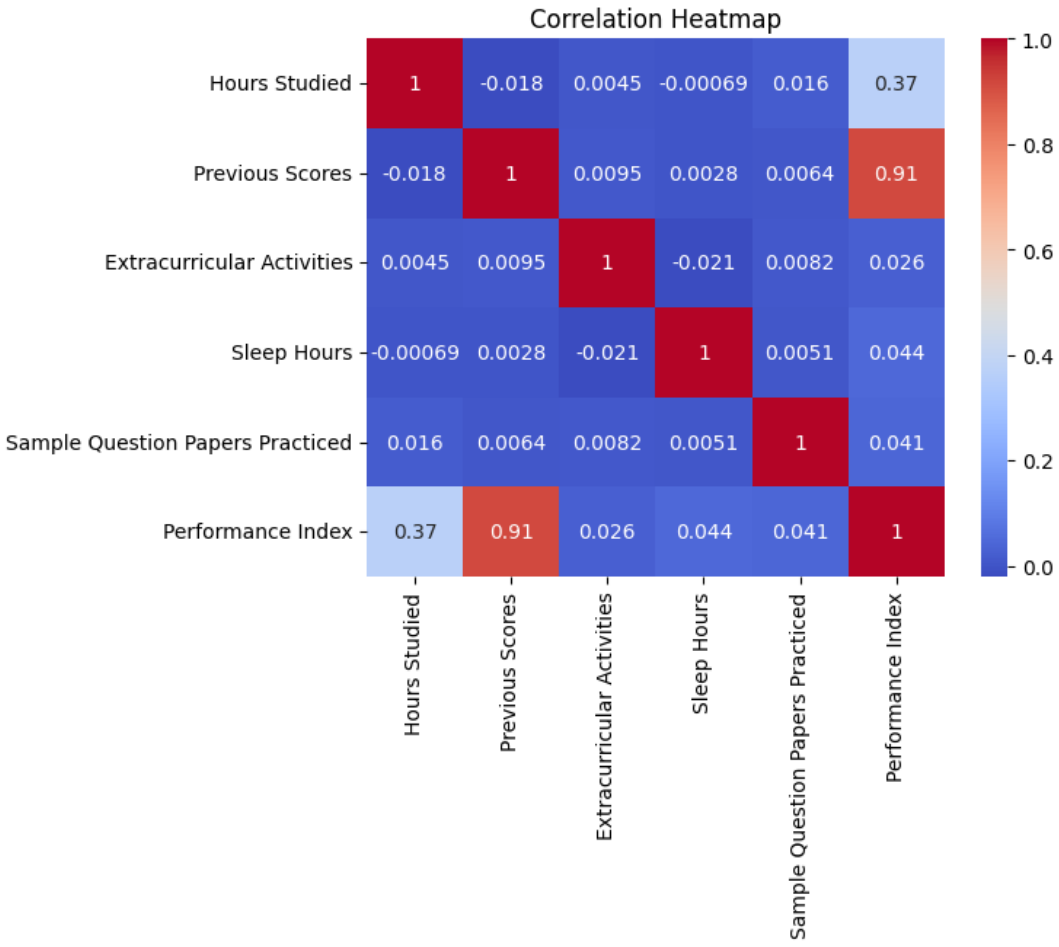
  – Áp lực điểm số cũng là một phần nguyên nhân chủ yếu dẫn đến kết quả này. Dựa theo số liệu thống kê được thì có đến hơn 60% sinh viên phải liên tục đối diện với áp lực điểm số. Khi sinh viên không đạt được những điểm số cao hoặc thậm chí là tuyệt đối thì sẽ được xem là người kém cỏi ảnh hưởng không nhỏ tới sức khỏe tinh thần. Cũng chính vì thế mà nhiều học sinh luôn bị ám ảnh về những điểm số, sinh viênluôn cố gắng học tập ngày đêm để giành lấy những điểm số cao. [1]

2.4 Yêu cầu 2c: Sinh viên tự xây dựng/thiết kế mô hình, tìm mô hình cho kết quả tốt nhất

- Ý tưởng xây dựng mô hình:

– Để xây dựng một mô hình tốt, trước hết ta cần phải chọn lọc được các đặc trưng tốt để đưa vào mô hình và loại bỏ các đặc trưng ít ảnh hưởng. Ta có thể sử dụng phương pháp chọn lọc như phương pháp chọn lọc dựa trên hệ số tương quan Pearson (Pearson Correlation Coefficient) – một phương pháp thuộc Filter Methods

– Sự tương quan là thước đo của mối quan hệ tuyến tính giữa hai hoặc nhiều biến. Thông qua độ tương quan, ta có thể dự đoán một biến từ biến khác. Các đặc trưng được chọn là những đặc trưng có độ tương quan cao nhất với giá trị mục tiêu. Nếu như hai biến tương quan mạnh với nhau, ta có thể dự đoán một biến từ biến còn lại. Do đó, mô hình có thể chỉ cần một trong hai chứ không cần cả hai. [7]

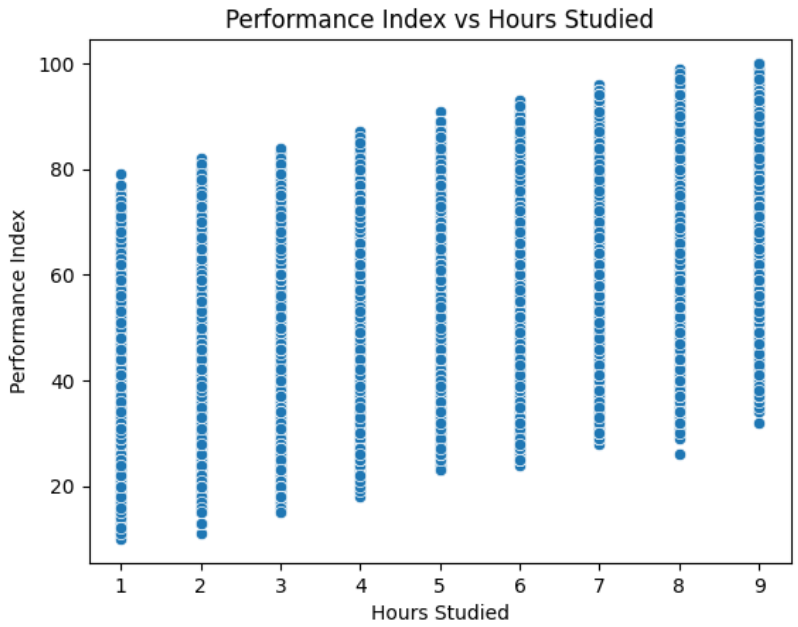


Hình 6: Biểu đồ trực quan hóa ma trận tương quan bằng heatmap

- Nhìn vào dòng cuối thì ta có thể dễ dàng nhận ra 2 đặc trưng tương quan nhất với thành tích học tập đó chính là Hours Studied và Previous Scores. Các đặc trưng còn lại có độ tương

quan không cao với lại thành tích học tập khi đứng riêng lẻ.

Ta xét thêm về đồ thị phân tán của 2 đặc trưng trên so với Performance Index



Hình 7: Biểu đồ phân tán giữa hours studied và Performance Index

Hình 8: Biểu đồ phân tán giữa Previous Scores và Performance Index

Mô hình 1 của chúng ta sẽ sử dụng 2 đặc trưng là Hour Studied và Previous Scores cho vào mô hình hồi quy tuyến tính. Khi từ đồ thị trên ta có thể thấy được hình dáng đồ thị là tuyến tính

dương, có nghĩa là khi x càng lớn thì y càng cao. Kết hợp 2 đặc trưng này có thể cho chúng ta mô hình tốt với bộ dữ liệu này.

- **Mô hình 1:**

$$\textbf{Student Performance} = w_0 + w_1 \times \textbf{Hour Studied} + w_2 \times \textbf{Previous Scores}$$

Để xây dựng mô hình 2, ta lại một lần nữa nhìn các chỉ số tương quan ở đồ thị 6. Ta có thể thấy ngoài 2 đặc trưng Hours Studied và Previous Scores có mối tương quan với nhau khá đậm thì còn có sleep hours và Extracurricular Activities. Nhưng đây là mối tương quan âm, tức 2 dữ liệu sẽ không có mối quan hệ tuyến tính rõ ràng. Ta lại nhìn vào đồ thị 8 thì ta có thể phán đoán là có thể đặc trưng Previous Scores sẽ có dạng phi tuyến tính (dạng cong). Ta có thể kết hợp các đặc trưng để tìm ra mô hình 2:

- **Mô hình 2:**

$$\textbf{Student Performance} = w_0 + w_1 \times \textbf{Hour Studied} + w_2 \times \textbf{Previous Scores}^2 + w_3 \times \textbf{Extracurricular Activities} + w_4 \times \textbf{Sleep Hours}$$

Khi tìm hiểu về bộ dữ liệu thì em nhận thấy rằng đặc trưng của Previous Scores và Hour Studied cũng có thể có hình dạng của đồ thị căn bậc 2. Xét thêm độ tương quan của 2 đặc trưng này với kết quả đích thì mô hình tạo nên từ việc căn bậc 2 các đặc trưng Previous Scores và Hour Studied cộng với 2 đặc trưng có mối quan hệ là sleep hours và Extracurricular Activities sẽ cho chúng ta **mô hình 3**

- **Mô hình 3:**

$$\textbf{Student Performance} = w_0 + w_1 \times \sqrt{\textbf{Hour Studied}} + w_2 \times \sqrt{\textbf{Previous Scores}} + w_3 \times \textbf{Extracurricular Activities} + w_4 \times \textbf{Sleep Hours}$$

- Kết quả có được từ k-fold cross-validation

STT	Mô hình	MAE
1	**Mô hình 1 với 2 đặc trưng HS và PS**	**1.816**
2	Mô hình 2 với 4 đặc trưng và PS bình phương	2.321
3	Mô hình 3 với 4 đặc trưng và HS, PS căn bậc 2	2.065

Bảng 3: Kết quả MAE của các mô hình

  - Mô hình tốt nhất là mô hình 1 với việc kết hợp 2 đặc trưng có độ tương quan cao nhất.
  - Công thức hồi quy tuyến tính của mô hình tốt nhất:

$$\textbf{Student Performance} = -29.747 + 2.856 \times \textbf{Hour Studied} + 1.018 \times \textbf{Previous Scores}$$

$$MAE = 1.839$$

- Nhận xét:
  - Từ mae trung bình có được nhờ kĩ thuật k-folds cross validation thì mô hình 1 được xem là mô hình tốt nhất trong 3 mô hình ở câu 2c này. Nếu so với kết quả ở câu 2a thì ta thấy MAE cũng không chênh lệch nhiều. Điều đó chứng minh rằng mối quan hệ tuyến tính của 2 đặc trưng có tương quan cao với thành tích học tập của sinh viên và chiếm phần quan trọng trong việc dự đoán thành tích sinh viên.
  - Giả thuyết cho việc mô hình 1 là mô hình tốt nhất là vì mô hình được tạo nên từ những đặc trưng có hệ số tương quan lớn. Mô hình tương đối đầy đủ, các đặc trưng thì có hệ số tương quan tốt, mô hình không bị Overfitting.
  - Mô hình chỉ có 2 đặc trưng nên. Dù có tương quan lớn với biến mục tiêu, 2 đặc trưng có thể không mang đến đủ thông tin hoặc khả năng dự đoán cho mô hình.

- Giả thuyết cho mô hình tốt nhất:
  - Ở mô hình 1, ta loại bỏ đi đặc trưng về Extracurricular Activities, Sleep Hours và Sample Question praticed.
  - Đầu tiên ta sẽ xét về việc tham gia hoạt động ngoại khóa (Extracurricular Activities), việc tham gia hoạt động ngoại khóa đúng là có ảnh hưởng đến thành tích của sinh viên như đã chứng minh ở biểu đồ 5 - các sinh viên tham gia hoạt động ngoại khóa thường có thành tích cao hơn các sinh viên không.
  - Theo tạp chí thiết bị giáo dục [5], có 92,8% sinh viên nhận thức được tầm quan trọng của hoạt động ngoại khóa, có ảnh hưởng tích cực tới thành tích. Nhưng trong dữ liệu của ta chỉ có 2 trạng thái là 0 và 1 thì sự ảnh hưởng của Extracurricular Activities gần như là không nhiều.

- Tiếp đến là 2 đặc trưng có độ tương quan với Performance Index gần như bằng nhau là Sleep Hours và Sample Question practiced. Cả 2 cũng có độ tương quan nhỏ với nhau chứng tỏ cả 2 hầu như không liên hệ với nhau chặt chẽ. Có nghĩa là học sinh có thể có nhiều hoặc ít thời gian ngủ mà không ảnh hưởng đến việc thực hành các đề thi. Điều này là hợp lý vì thời gian ngủ bị rất nhiều yếu tố chi phối và nó còn không ảnh hưởng chính đến thành tích học tập của các sinh viên.[4]
- 2 đặc trưng được sử dụng có mối liên hệ chặt chẽ đến thành tích học tập. Previous Scores ảnh hưởng đến thành tích như đã được trình bày ở câu 2b, còn Hour Studied ảnh hưởng chỉ kém mỗi Previous Scores. Điều này cũng phản ánh đúng hiện thực khi thời gian đầu tư cho việc học càng nhiều thì điểm số sẽ càng tăng cao. Kết hợp 2 giá trị có đồ thị tuyến tính dương này lại chúng ta sẽ có được mô hình có độ lỗi thấp nhất trong 3 mô hình.

Tài liệu

[1] Stress for student. https://research.com/education/student-stress-statistics.

[2] Seaborn document. 2018.

[3] Nguyễn Thanh Bình. Project 03: Linear regression. 2022.

[4] Bao Nhan Dan. Tầm quan trọng của giấc ngủ. https://daibieunhandan.vn/song-khoe/tac-dong-cua-giac-ngu-doi-voi-hoc-tap-va-tri-nho-i305388/.

[5] Pham Duc Thanh. Tạp chí giáo dục. https://csdlkhoahoc.hueuni.edu.vn/data/2022/1/Pham_Duc_Thanh.pdf.

[6] Phan Thị Phương uyên. Lab 04 - linear regression.

[7] Vidhya. Features selection in ml. https://www.analyticsvidhya.com/blog/2020/10/feature-selection-techniques-in-machine-learning/#What_Is_Feature_Selection_in_Machine_Learning.