HO CHI MINH UNIVERSITY OF SCIENCE

FACULTY OF INFORMATION TECHNOLOGY

----------o0o----------

Applied Mathematics and Statistics for Information Technology

Project 3: Linear Regression

Student ID : 23127206
Student name : Lê Nguyễn Nhật Khánh
Class : 23CLC04

Ho Chi Minh City, August 12 2025

Mục lục

I. Ý tưởng thực hiện

1. *Tổng quan về đồ án*

- Trong lĩnh vực giáo dục, việc dự đoán chính xác hiệu suất học tập của sinh viên là một thách thức quan trọng. Hiểu được các yếu tố ảnh hưởng đến kết quả học tập không chỉ giúp các nhà giáo dục đưa ra những can thiệp kịp thời mà còn hỗ trợ sinh viên tối ưu hóa phương pháp học tập của mình.
- Đồ án này tập trung vào việc dự đoán hiệu suất học tập của sinh viên thông qua việc ứng dụng các kỹ thuật học máy, cụ thể là hồi quy tuyến tính. Với mục tiêu xây dựng và so sánh hiệu quả của các mô hình dự đoán khác nhau, đồ án thực hiện một quy trình hoàn chỉnh từ khám phá dữ liệu đến đánh giá mô hình cuối cùng.

2. *Cách thức nhập dữ liệu*

- Việc nhập dữ liệu được thực hiện thông qua thư viện pandas với hai file chính dùng để train và test là p03.train.csv và p03.test.csv.

3. *Mục tiêu và ý tưởng thực hiện*

- *Mục tiêu của đồ án*
  - Làm quen với các thao tác cơ bản trong machine learning và hồi quy tuyến tính.
  - Hiểu được bản chất của thuật toán OLS, cách mô hình hồi quy tuyến tính hoạt động từ công thức toán học cơ bản.
  - Biết cách sử dụng jupyter notebook và các thư viện của Python cho data science.
  - Rèn luyện kĩ năng lập trình python.
  - Tạo nền tảng cho việc học các kỹ thuật machine learning phức tạp hơn sau này.
- *Ý tưởng cốt lõi*
  - Đọc dữ liệu sinh viên từ file CSV và chuyển thành DataFrame, trong đó mỗi dòng chứa thông tin về một sinh viên với 5 đặc trưng đầu vào và 1 giá trị hiệu suất học tập.
  - Sử dụng thư viện của Python để thống kê dữ liệu và vẽ các biểu đồ quan hệ để thấy được mối quan hệ giữa các đặc trưng và giá trị hiệu suất học tập.
  - Gọi đến các hàm xử lý dữ liệu mà người dùng yêu cầu để train và đánh giá mô hình.
  - Dựa trên mô hình đã đánh giá trên để dự đoán các kết quả cũng như tính sai số trên tập kiểm tra.

II. Chi tiết thực hiện

1. *Các hàm và thư viện đã sử dụng*

a. *Thư viện pandas*

- Thư viện pandas đóng vai trò quan trọng và được sử dụng xuyên suốt trong đồ án với các chức năng chính sau:
  - Đọc dữ liệu từ file csv và chuyển đổi dữ liệu thành Dataframe để xử lý dễ dàng hơn với cú pháp “read_csv”.
  - Tách dữ liệu vừa đọc được để tạo thành DataFrame cho các features và Series cho target.
  - Thống kê dữ liệu bằng “describe” hay xử lý duplicate, shuffle.

⇨ Pandas là thư viện cốt lõi giúp xử lý toàn bộ quy trình từ nhập dữ liệu, xử lý, phân tích đến chuẩn bị dữ liệu cho modeling trong đồ án này.

b. *Thư viện numpy*

- Thư viện numpy là nền tảng toán học và tính toán số trong đồ án, được sử dụng chủ yếu cho các tính toán ma trận và thuật toán machine learning:

- Dùng để tính toán trong công thức của thuật toán OLS, trong thống kê dữ liệu,…
- Thực hiện phép toán với ma trận như nhân, nghịch đảo…
- Chuyển đổi dữ liệu với DataFrame/Series thành numpy array.

c. Thư viện matplotlib

- Thư viện matplotlib đóng vai trò chính trong việc visualize dữ liệu và trình bày kết quả trong đồ án, được sử dụng qua các hàm vẽ biểu đồ chuyên biệt:
  - Vẽ histogram cho từng features và target.
  - Vẽ scatter plots để phân tích mối quan hệ của từng features với target.
  - Format, tùy chỉnh và làm cho các biểu đồ dễ nhìn, hoàn thiện hơn.

d. Thư viện seaborn

- Thư viện seaborn được sử dụng một cách chuyên biệt và quan trọng trong đồ án để tạo ra correlation heatmap - một visualization chuyên nghiệp và trực quan.

e. Class OLSLinearRegression

- Mục tiêu của class: xây dựng mô hình hồi quy tuyến tính sử dụng phương pháp Ordinary Least Squares (OLS).

i. Hàm init

- Mục tiêu của hàm: khởi tạo mô hình với trọng số rỗng và bias bằng không.
- Mô tả hàm:
  - Khởi tạo thuộc tính weight để lưu trọng số khi train model.
  - Khởi tạo thuộc tính intercept để lưu bias và cho nó bằng không.
  - Chuẩn bị đối tượng mô hình sẵn sàng cho việc huấn luyện.

ii. Hàm fit

- Mục tiêu của hàm: tính toán vector trọng số tối ưu cho mô hình hồi quy tuyến tính dựa trên dữ liệu huấn luyện.
- Mô tả hàm:
  - Kiểm tra và chuyển đổi dữ liệu X, y sang numpy array.
  - Áp dụng công thức để tính vector trọng số.

$$w = (X^T * X)^{-1} * X^T * Y$$

  - Lưu trọng số và thuộc tính w của đối tượng.
- Input của hàm:
  - X: DataFrame chứa dữ liệu đầu vào.
  - y: Series chứa dữ liệu đầu ra mục tiêu.
- Output của hàm: trả về đối tượng self với trọng số đã được tính toán và lưu trữ.

iii. Hàm predict

- Mục tiêu của hàm: dự đoán giá trị đầu ra dựa trên dữ liệu đầu vào và trọng số đã học được.
- Mô tả hàm:
  - Đầu tiên chúng ta cần kiểm tra xem mô hình đã được huấn luyện chưa dựa vào trọng số.
  - Chuyển đổi dữ liệu X sang numpy array.
  - Tính toán kết quả dự đoán bằng cách nhân hai ma trận X và vector trọng số w.
- Input của hàm:
  - X: DataFrame chứa dữ liệu cần dự đoán.
- Output của hàm: mảng numpy chứa các giá trị dự đoán tương ứng với từng mẫu dữ liệu đầu vào

iv. Hàm get_params

- Mục tiêu của hàm: lấy ra vector trọng số đã học được của mô hình.
- Mô tả hàm:
  - Trả về trọng số w của mô hình.
- Output của hàm: mảng numpy chứa các trọng số của mô hình đã học được.

f. Hàm calculate_mse

- Mục tiêu của hàm: tính toán sai số bình phương trung bình giữa giá trị thực tế và giá trị dự đoán để đánh giá hiệu suất mô hình.
- Mô tả hàm:
  - Nhận đầu vào là hai mảng y_true và y_pred đại diện cho giá trị thực tế và giá trị dự đoán.
  - Kiểm tra và chuyển đổi dữ liệu từ pandas Series sang numpy array nếu cần thiết.
  - Tính vector sai số residuals bằng phép trừ: y_true - y_pred.
  - Tính bình phương từng phần tử trong vector sai số và lấy trung bình tất cả các giá trị bình phương để có MSE cuối cùng.
  - Chuyển đổi kết quả về kiểu float và trả về.
- Input của hàm:
  - y_true: pandas Series chứa giá trị thực tế của biến mục tiêu.
  - y_pred: pandas Series chứa giá trị dự đoán từ mô hình.
- Output của hàm: giá trị MSE – chỉ số đánh giá độ sai sót của một mô hình hồi quy.

g. Hàm process_train_data

- Mục tiêu của hàm: tiền xử lý dữ liệu huấn luyện bằng cách loại bỏ các dòng trùng lặp và xáo trộn dữ liệu.
- Mô tả hàm:
  - Nhận vào DataFrame gốc chứa dữ liệu huấn luyện.
  - Kiểm tra và đếm số lượng các dòng dữ liệu bị lặp trong bộ dữ liệu.
  - Loại bỏ các dòng dữ liệu bị lặp bằng “drop_duplicates” để đảm bảo dữ liệu không bị trùng lặp khi đem đi train.
  - Xáo trộn dữ liệu một cách ngẫu nhiên với “random_state = 42” để đảm bảo có khả năng phục hồi lại dữ liệu cũ sau này.
  - Cuối cùng là tách dữ liệu đã làm sạch thành hai phần, một là tập đặc trưng đầu (X_train) và biến mục tiêu (y_train).
- Input của hàm:
  - Train: pandas DataFrame chứa dữ liệu huấn luyện ban đầu với tất cả features và biến target.
- Output của hàm:
  - train_new: DataFrame dữ liệu huấn luyện sau khi đã được loại trùng và xáo trộn.
  - X_train: DataFrame chỉ chứa các features.
  - y_train: Series chứa giá trị của biến mục tiêu.

h. Hàm plot_train_histograms

- Mục tiêu của hàm: vẽ biểu đồ histogram để thể hiện phân phối tần số của tất cả các đặc trưng và biến mục tiêu trong tập dữ liệu huấn luyện nhằm khám phá và hiểu được phân bố dữ liệu.
- Mô tả hàm:
  - Nhận đầu vào gồm DataFrame chứa các đặc trưng (X_train) và Series chứa biến mục tiêu (y_train).
  - Tạo một figure với layout 2 hàng, 3 cột để đồng thời hiển thị tất cả biểu đồ về sự phân bố.

  - Vẽ từng histogram cho từng feature với màu xanh da trời và cho biến mục tiêu với màu đỏ nhạt để làm nổi bật target.
  - Thêm tiêu đề cho các biểu đồ, tiêu đề tổng thể và tối ưu hóa bố cục.
- Input của hàm:
  - X_train: pandas DataFrame chứa các đặc trưng đầu vào của 5 features.
  - y_train: pandas Series chứa biến mục tiêu.
- Output của hàm: hiển thị biểu đồ histogram cho tất cả các biến trong tập huấn luyện, giúp người dùng quan sát trực quan phân bố dữ liệu, phát hiện outliers và hiểu đặc điểm thống kê của từng biến.

i. Hàm plot_feature_vs_target_scatter

- Mục tiêu của hàm: trực quan hóa mối quan hệ giữa từng đặc trưng đầu vào và biến mục tiêu thông qua các biểu đồ scatter plot để phân tích tương quan và pattern trong dữ liệu.
- Mô tả hàm:
  - Sử dụng biến toàn cục X_train và y_train để lấy dữ liệu đặc trưng và biến mục tiêu.
  - Tạo một figure với layout 2 hàng, 3 cột để đồng thời hiển thị tất cả biểu đồ.
  - Lặp qua từng đặc trưng trong 5 đặc trưng đầu vào để tạo từng biểu đồ scatter.
  - Với mỗi đặc trưng, vẽ scatter plot các điểm dữ liệu với màu xanh da trời và viền đen.
  - Tính toán và vẽ đường hồi quy tuyến tính bậc nhất bằng “np.polyfit” và “np.poly1d” với màu đỏ đứt nét.
  - Tìm hệ số tương quan bằng “np.corrcoef” và hiển thị trong text box ở góc trên bên trái của mỗi biểu đồ.
  - Thiết lập tiêu đề, nhãn trục và format lại cho đẹp.
- Output của hàm: hiển thị 5 biểu đồ scatter plot cùng với trend line và hệ số tương quan, giúp người dùng trực quan hóa và phân tích mối quan hệ giữa từng đặc trưng với biến mục tiêu Performance Index.

j. Hàm plot_correlation_heatmap

- Mục tiêu của hàm: trực quan hóa ma trận tương quan giữa tất cả các biến số trong dữ liệu huấn luyện thông qua biểu đồ heatmap để phân tích mối quan hệ tương quan giữa các đặc trưng và biến mục tiêu.
- Mô tả hàm:
  - Sử dụng biến toàn cục train chứa toàn bộ dữ liệu huấn luyện.
  - Tính toán ma trận tương quan giữa các biến số bằng phương thức ” .corr(numeric_only=True)”.
  - Tạo figure để hiển thị heatmap rõ ràng và dùng “sns.heatmap” để tùy chỉnh các tham số sao cho biểu đồ hiển thị rõ ràng và trực quan nhất.
- Output của hàm: hiển thị biểu đồ heatmap chuyên nghiệp thể hiện ma trận tương quan giữa tất cả các biến, giúp người dùng nhanh chóng nhận biết mức độ tương quan và phát hiện các mối quan hệ quan trọng trong dữ liệu để hỗ trợ quá trình phân tích và lựa chọn đặc trưng.

k. Hàm k_fold_cross_validation_unified

- Mục tiêu của hàm: thực hiện K-fold Cross Validation cho nhiều mô hình cùng lúc rồi chọn mô hình có Mean MSE thấp nhất.
- Mô tả hàm:
  - Nhận models_dict chứa các tên model và dataframe dùng để train.
  - Xác định kích thước của fold, sau đó tuần tự chia dữ liệu thành k đoạn liên tiếp.
  - Với mỗi mô hình, chúng ta sẽ:
    - Lặp qua từng fold, khởi tạo tập validation và tập train của nó.

    - Khởi tạo mô hình bằng cách tạo object của class rồi đem nó huấn luyện trên tập train vừa tạo được ở trên.
    - Đem đi dự đoán và tính MSE với tập validation và lưu MSE đó vào biến chứa danh sách các MSE của 1 mô hình.
  - Tính mean MSE của từng mô hình rồi đem đi sắp xếp theo tăng dần để lấy ra được mô hình tốt nhất.
  - In bảng thống kê vừa tìm được ra màn hình và thông báo mô hình tốt nhất.
  - Trả về các giá trị cần thiết để tiếp tục với bước tiếp theo là huấn luyện trên full data.
- Input của hàm:
  - models_dict: dictionary có dạng {model name: X_dataframe} chứa các bộ đặc trưng đã thiết kế trước đó.
  - y_train: pandas Series giá trị của target tương ứng.
  - k: số lượng fold.
- Output của hàm:
  - best_model_name: tên của mô hình có mean MSE nhỏ nhất
  - best_X_data: DataFrame đặc trưng tương ứng dùng để cho bước huấn luyện sau đó.

l. Hàm create_test_data

- Mục tiêu của hàm: tạo dữ liệu đầu vào phù hợp cho tập test dựa trên mô hình tốt nhất được chọn từ quá trình cross-validation, đảm bảo format đầu vào nhất quán với mô hình đã huấn luyện.
- Mô tả hàm:
  - Nhận đầu vào là DataFrame chứa dữ liệu test gốc và tên mô hình được chọn.
  - Dựa vào tên model được truyền vào mà trả về DataFrame tương ứng của tập test để phục vụ cho việc train và tính MSE sau này.
- Input của hàm:
  - X_test: pandas DataFrame chứa dữ liệu test gốc với tất cả 5 đặc trưng ban đầu.
  - Model_name: tên của mô hình được chọn làm mô hình tốt nhất.
- Output của hàm: pandas DataFrame chứa tập đặc trưng được biến đổi phù hợp với kiến trúc của mô hình được chọn, sẵn sàng làm input cho việc dự đoán trên tập test với mô hình đã huấn luyện.

2. Phân tích các yêu cầu của bài toán

a. Phân tích khám phá dữ liệu

i. Thống kê dữ liệu

- Dựa vào việc thống kê tập train, tôi đã có được các thông số sau:
  - Hours Studied: trung bình 4.98 giờ/ngày (1-9 giờ), độ lệch chuẩn 2.59. Phân phối khá đồng đều với median = 5 giờ.
  - Previous Scores: trung bình 69.4 điểm (40-99), độ lệch chuẩn 17.35. Phân phối gần chuẩn với median = 69 điểm.
  - Extracurricular Activities: 49.4% học sinh tham gia hoạt động ngoại khóa.
  - Sleep Hours: trung bình 6.54 giờ/đêm (4-9 giờ), độ lệch chuẩn 1.70. Median = 7 giờ.
  - Sample Question Papers Practiced: trung bình 4.59 đề (0-9 đề), độ lệch chuẩn 2.86. Phân phối khá rộng.
  - Performance Index: trung bình 55.14 điểm (10-100), độ lệch chuẩn 19.19. Phân phối gần chuẩn với median = 55 điểm.
- Đánh giá về bộ dữ liệu:
  - Không có giá trị thiếu trong tất cả các biến.

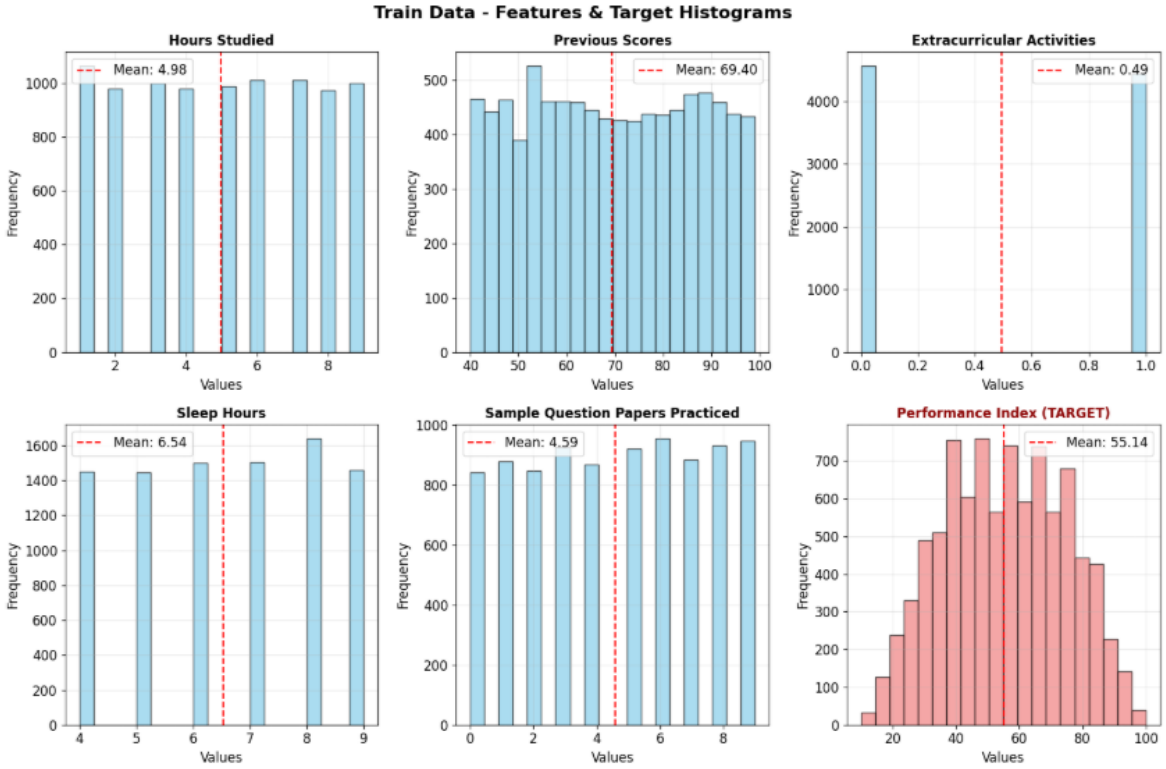
  - Tất cả các giá trị đều nằm trong phạm vi thực tế.
  - Dataset đủ lớn để đảm bảo tính đại diện.
- Phân phối dữ liệu:
  - Previous Scores và Performance Index có phân phối gần chuẩn nên thuận lợi cho mô hình hồi quy tuyến tính.
  - Sleep Hours tập trung ở 6-8 giờ (75% học sinh), điều này là hoàn toàn phù hợp khuyến nghị y học hiện nay.
  - Sample Question Papers có phân phối rộng (0-9 đề). Điều này thể hiện đa dạng về cường độ luyện tập.
- Tính chất của biến mục tiêu:
  - Có phạm vi rộng: 10-100 điểm, điều này cho phép đánh giá toàn diện hiệu suất học tập.
  - Có phân phối chuẩn với mean khoảng bằng 55, điều này thuận lợi cho mô hình tuyến tính.
  - Độ biến thiên hợp lý: đủ lớn để phân biệt nhưng không quá phân tán.
  - Tất cả giá trị đều hợp lý, không có outlier cực đoan.
- Mối quan hệ tiềm năng với Performance Index:
  - Previous Scores có khả năng là predictor mạnh nhất do cùng bản chất đánh giá học tập.
  - Hours Studied có thể có tương quan tích cực nhưng có thể phi tuyến.
  - Sample Question Papers khả năng tương quan mạnh với việc chuẩn bị thi cử.
  - Sleep Hours có thể có mối quan hệ phi tuyến.
- Đặc điểm nổi bật của bộ dữ liệu trên:
  - Dataset sạch và chất lượng, sẵn sàng cho modeling.
  - Phân phối gần chuẩn của target variable thuận lợi cho linear regression.

___THỐNG KÊ DỮ LIỆU CỦA TRAIN___

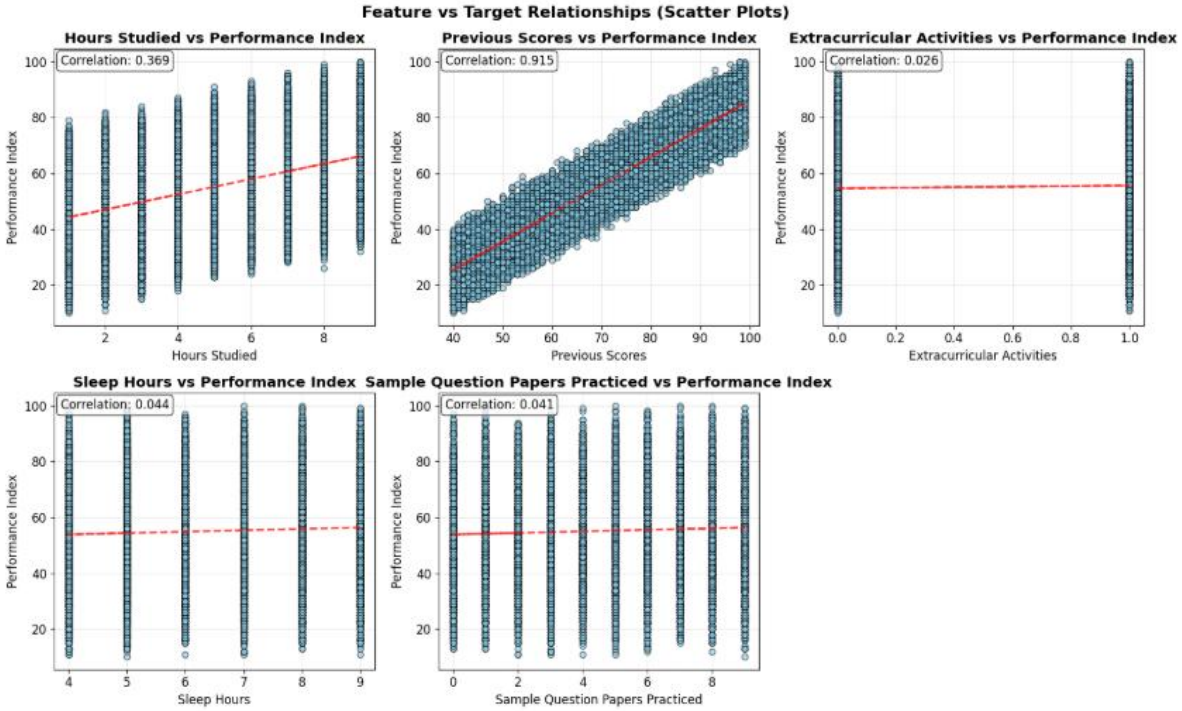
	Hours Studied	Previous Scores	Extracurricular Activities	\
count	9000.000000	9000.000000	9000.000000	
mean	4.976444	69.396111	0.493667	
std	2.594647	17.369957	0.499988	
min	1.000000	40.000000	0.000000	
25%	3.000000	54.000000	0.000000	
50%	5.000000	69.000000	0.000000	
75%	7.000000	85.000000	1.000000	
max	9.000000	99.000000	1.000000	

	Sleep Hours	Sample Question Papers Practiced	Performance Index
count	9000.000000	9000.000000	9000.000000
mean	6.535556	4.590889	55.136333
std	1.695533	2.864570	19.187669
min	4.000000	0.000000	10.000000
25%	5.000000	2.000000	40.000000
50%	7.000000	5.000000	55.000000
75%	8.000000	7.000000	70.000000
max	9.000000	9.000000	100.000000

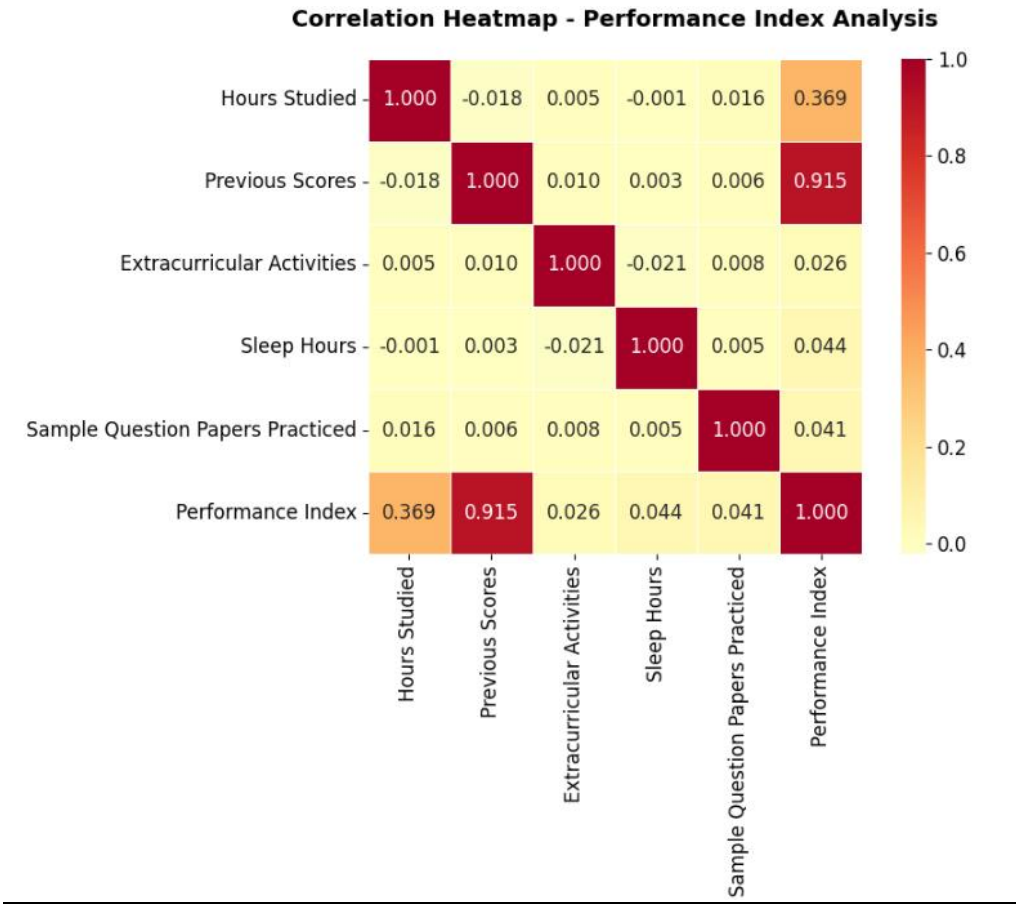
ii. *Xây dựng các biểu đồ*



- Đối với Hours Studied, ta có thể thấy được mean là 4.98 và median là 5. Điều đó cho thấy dữ liệu phân phối đối xứng với các giá trị phân bố đều từ 1-9 giờ. Không có xu hướng lệch, thể hiện sự cân bằng tốt trong thói quen học tập.
- Đối với Previous Scores, ta có mean là 69.4 và median là 70. Có phân phối lệch trái, gần chuẩn nhưng có đuôi kéo dài về phía điểm thấp. Thể hiện rằng có một số sinh viên với điểm nền tảng thấp kéo trung bình xuống, nhưng phần lớn tập trung ở mức điểm khá và giỏi.
- Extracurricular Activities, phân phối tương đối cân bằng với 49% sinh viên tham gia hoạt động ngoại khóa.
- Sleepy Hours, phân phối lệch phải, tập trung mạnh ở 7-8 giờ. Cho thấy đa số sinh viên có thói quen ngủ đủ giấc, chỉ một số ít thiếu ngủ nghiêm trọng.
- Sample Question Papers Practiced, có phân phối gần đối xứng. Thể hiện sự đa dạng cân bằng trong cường độ luyện tập, từ ít đến nhiều.
- Performance Index, ta có phân phối đối xứng khá hoàn hảo. Điều này thể hiện sự phù hợp cho mô hình hồi quy tuyến tính.



- Hours Studied vs Performance Index, có mối quan hệ tương quan dương tuyến tính mạnh. Thể hiện xu hướng rõ ràng, càng học nhiều giờ thì hiệu suất càng cao. Độ tập trung dữ liệu tương đối đồng đều và ít outlier.
- Previous Scores vs Performance Index, tương quan dương rất mạnh và gần như là tuyến tính hoàn hảo. Đường thẳng rõ ràng và cao nhất trong tất cả các biến. Thể hiện rõ đây là feature tốt nhất dùng để dự đoán target.
- Extracurricular Activities vs Performance Index, tham gia hoạt động ngoại khóa có tác động tích cực lên hiệu suất học tập. Điều này cho thấy việc phát triển toàn diện không chỉ không làm giảm mà còn hỗ trợ tích cực cho kết quả học tập.
- Sleep Hours vs Performance Index, tương quan phi tuyến với dạng cong lên. Hiệu suất tốt nhất là ở 6-8 giờ ngủ và giảm về 2 đầu. Thể hiện cho thấy đây mà một mối quan hệ phức tạp, quá ít hoặc quá nhiều đều không tốt.
- Sample Question Papers Practiced vs Performance Index, có mối quan hệ tương quan dương mạnh. Tăng nhanh từ 0-5 đề và tăng chậm hơn về sau. Cho thấy luyện tập có mang lại lợi ích nhưng lợi ích biên giảm sau một ngưỡng nhất định.



- Previous Scores là feature có tương quan mạnh nhất với Performance Index nên được xem là yếu tố quyết định chính.
- Hour Studied có tác động rõ rệt nhưng không quyết định.
- Extracurricular Activities có tác động tích cực nhỏ.
- Còn những features còn lại có tính tương quan tuyến tính rất yếu.

b. *<u>Xây dựng mô hình sử dụng toàn bộ 5 đặc trưng</u>*

- Trong phần này chúng ta sẽ tiến hành xây dựng mô hình hồi quy tuyến tính dự đoán Performance Index sử dụng tất cả 5 đặc trưng đầu vào để đánh giá khả năng dự đoán tổng thể khi tận dụng toàn bộ thông tin có sẵn.
- Chúng ta sẽ sử dụng thuật toán OLS đã được cài đặt bằng class để huấn luyện mô hình với toàn bộ features.
- Kết quả đạt được khái quát như sau:
  - Hours Studied sẽ có hệ số lớn nhất.
  - Previous Scores có hệ số dương do tương quan mạnh.
  - Ba đặc trưng còn lại có hệ số nhỏ do tương quan yếu với target.
- Ý nghĩa: xây dựng được baseline model sử dụng toàn bộ thông tin có sẵn và kết quả được dùng để làm chuẩn cho các phép só sánh giữa các mô hình khác sau này. Đồng thời cũng giúp đánh giá được sự đóng góp của từng feature trong bối cảnh tổng thể.

c. *<u>Xây dựng mô hình sử dụng duy nhất 1 đặc trưng, tìm mô hình cho kết quả tốt nhất</u>*

- Mục tiêu: xây dựng 5 mô hình hồi quy tuyến tính riêng biệt, mỗi mô hình sử dụng duy nhất 1 đặc trưng để dự đoán Performance Index. So sánh MSE để xác định đặc trưng đơn lẻ có

khả năng dự đoán tốt nhất, đồng thời cũng đánh giá được tầm quan trọng của từng feature khi hoạt động độc lập.

- Chúng ta cũng sẽ sử dụng thuật toán OLS để huấn luyện mô hình với từng feature.
- Kết quả đạt được:
  - Previous Scores có MSE thấp nhất, điều đó cho thấy nó là mô hình tốt nhất.
  - Hours Studied có MSE trung bình thấp thứ hai sau Previous Scores.
  - Các features còn lại có MSE cao do tương quan yếu.
- Ý nghĩa: xác định thứ tự quan trọng của từng đặc trưng khi hoạt động độc lập, đánh giá hiệu quả của mô hình đơn giản nhất. Từ kết quả so sánh của toàn bộ features, chúng ta có thể xác định được feature nào có thể đứng độc lập nếu cần thiết trong thực tế.

d. Tự xây dựng/ thiết kế mô hình và tìm mô hình cho kết quả tốt nhất

- Chúng ta sẽ thiết kế 5 mô hình tùy chỉnh bằng việc kết hợp các features để cải thiện khả năng dự đoán. Mục đích của việc làm này là để tạo ra các đặc trưng mới có ý nghĩa và so sánh hiệu suất giữa các mô hình để tìm ra mô hình dự đoán tốt nhất.
- Một số mô hình mà tôi đã thiết kế được:
  - Mô hình Polynomial: xây dựng mô hình với hàm bậc 3 của Previous Scores - feature tốt nhất mà ta đã tìm ra được từ câu 2b. Mô hình này được xây dựng bởi sự tương quan mạnh mà nó đem lại. Cho thấy sinh viên giỏi có thể có hiệu suất tăng theo cấp số nhân.
  - Mô hình Study-Health Balance: được xây dựng dựa trên việc kết hợp sự nỗ lực và sự thoải mái cho hiệu suất bền vững. Với Hours Studied có tác động tích cực rõ rệt và Sleep Hours có hiệu ứng vùng tối ưu từ biểu đồ phân tán.
  - Mô hình Learning Interaction: xây dựng dựa trên Hours Study và Sample Question Papers Practiced. Hai features trên có hiệu ứng tương tác với nhau, giờ học sẽ có chất lượng khi kết hợp với việc luyện tập.
  - Mô hình Learning Efficiency: thể hiện sự hiệu quả trong việc học tập bằng cách kết hợp giữa năng lực học tập trên mỗi đơn vị thời gian đầu tư và cường độ luyện tập tương đối với thời gian học.
  - Mô hình Academic Profile: kết hợp nền tảng, luyện tập và kĩ năng mềm. Thúc đẩy cho sự phát triển một cách toàn diện.
- Ý nghĩa: chúng ta có thể tự tay thiết kế các mô hình dựa trên sự hiểu biết về dữ liệu. Tìm thấy các model cho hiệu suất tối ưu nhất, từ đó áp dụng vào thực tế trong nền giáo dục.

III. Kết quả và kết luận

1. *Kết quả*

a. Với mô hình sử dụng toàn bộ 5 đặc trưng

- Hệ số và công thức của model:

```
HỆ SỐ
------------------------------------------------------------
   w1 | Hours Studied                    :     2.194027
   w2 | Previous Scores                  :     0.817060
   w3 | Extracurricular Activities       :    -1.150036
   w4 | Sleep Hours                      :    -1.500658
   w5 | Sample Question Papers Practiced :    -0.269084
```

```
Performance Index = 2.194 × Hours Studied + 0.817 × Previous Scores
                  - 1.150 × Extracurricular Activities
                  - 1.501 × Sleep Hours
                  - 0.269 × Sample Question Papers Practiced
```

- Kết quả độ đo MSE trên tập test = 36.580184
- Nhận xét: mô hình sử dụng toàn bộ 5 đặc trưng đạt MSE = 36.58 trên tập test, trong đó Hours Studied và Previous Scores là hai yếu tố tích cực chính, trong khi các hệ số âm bất ngờ của Sleep Hours và Extracurricular Activities gợi ý rằng trong bối cảnh tuyến tính, chất lượng thời gian học tập tập trung quan trọng hơn số lượng hoạt động và có thể tồn tại các mối quan hệ phi tuyến phức tạp mà mô hình chưa nắm bắt được.

b. Với mô hình sử dụng duy nhất 1 đặc trưng

- Bảng kết quả cross-validation và đặc trưng tốt nhất

```
BẢNG KẾT QUẢ K-FOLD CROSS VALIDATION
======================================================================
STT Mô hình                                  Mean MSE
----------------------------------------------------------------------
1   Previous Scores                          73.397387
2   Sleep Hours                              535.915510
3   Hours Studied                            686.114715
4   Sample Question Papers Practiced         1180.868888
5   Extracurricular Activities               1877.421092
----------------------------------------------------------------------

KẾT QUẢ TỐT NHẤT
========================================
   • Mô hình tốt nhất: Previous Scores
   • Mean MSE: 73.397387
```

- Trọng số của mô hình khi train feature tốt nhất với toàn bộ tập train

```
Previous Scores là feature tốt nhất.
Trọng số: 0.8071871221778 21
```

$$\text{Student Performance} = 0.807 \times \text{Previous Scores}$$

- Kết quả độ đo MSE trên tập test = 74.983
- Previous Scores là đặc trưng đơn lẻ tốt nhất với MSE = 73.397, thể hiện sự vượt trội áp đảo so với tất cả các đặc trưng khác với khoảng cách chênh lệch hơn 7 lần về độ chính xác. Kết quả này khẳng định mạnh mẽ nguyên lý liên tục trong học tập - hiệu suất trong quá khứ là yếu tố dự đoán đáng tin cậy nhất cho thành tích tương lai, phản ánh tính bền vững và tích lũy của quá trình giáo dục.
- Mô hình trên không chỉ đặt được hiệu quả dự đoán cao mà còn sở hữu những ưu thế vượt trội về mặt thực tiễn như: độ phức tạp thấp, tính diễn giải hoàn hảo và khả năng ứng dụng rộng rãi trong các hệ thống đánh giá giáo dục.

c. Các mô hình tự xây dựng khác

- Bảng kết quả cross-validation và mô hình tốt nhất

```
BẢNG KẾT QUẢ K-FOLD CROSS VALIDATION
======================================================================
STT Mô hình                                  Mean MSE
----------------------------------------------------------------------
1   Polynomial                               60.206078
2   Academic Profile                         72.794092
3   Study-Health Balance                     391.359037
4   Learning Efficiency                      1217.013632
5   Learning Interaction                     1411.657772
----------------------------------------------------------------------

KẾT QUẢ TỐT NHẤT
========================================
   • Mô hình tốt nhất: Polynomial
   • Mean MSE: 60.206078
```

- Trọng số và mô hình tốt nhất trong những mô hình mà tôi thiết kế

```
Model tốt nhất là: Polynomial
w1: 0.315213
w2: 0.010310
w3: -0.000049
```

$$\text{Student Performance} = 0.315 \times \text{Previous Scores} + 0.010 \times \text{Previous Scores}^2 - 0.000 \times \text{Previous Scores}^3$$

- Kết quả độ đo MSE trên tập test = 58.932
- Mô hình Polynomial đạt hiệu suất vượt trội với MSE = 58.932, cải thiện 17.7% so với mô hình đơn đặc trưng tốt nhất (Previous Scores: 73.40), chứng minh rằng kỹ thuật tạo đặc trưng có căn cứ khoa học có thể mang lại giá trị thực tiễn đáng kể. Hệ số bậc hai với giá trị 0.010 thành công trong việc nắm bắt mối quan hệ phi tuyến của quá trình học tập, trong đó sinh viên có nền tảng kiến thức vững chắc thể hiện hiệu ứng tăng tốc theo cấp số nhân thay vì tăng trưởng tuyến tính đơn thuần.
- Kết quả này không chỉ tái khẳng định sự áp đảo tuyệt đối của Previous Scores trong dự đoán hiệu suất học tập mà còn tiết lộ bản chất phi tuyến sâu sắc của quá trình giáo dục, trong đó nền tảng kiến thức vững chắc tạo ra hiệu ứng domino tích cực - sinh viên có điểm cao trước đó không chỉ duy trì được thành tích mà còn gia tăng động lực học tập theo cấp số nhân. Điều này phản ánh một thực tế quan trọng trong tâm lý học giáo dục: thành công nuôi dưỡng thành công, khi sự tự tin và động lực từ những thành tích trước đó khuếch đại khả năng tiếp thu kiến thức mới, tạo ra một vòng tròn tích cực mà các mô hình tuyến tính truyền thống không thể nắm bắt được.

2. *Nhận xét chung về đồ án*
   - Đồ án cho thấy tầm quan trọng và khả năng của machine learning trong việc khám phá những pattern ẩn sâu trong dữ liệu giáo dục, biến thông tin thô thành những hiểu biết có giá trị để cải thiện hiệu quả học tập và ra quyết định trong giáo dục.
   - Thông qua đồ án, tôi đã hiểu hơn về quy trình data science hoàn chỉnh từ khám phá dữ liệu, xây dựng giả thuyết, thiết kế thí nghiệm đến đánh giá kết quả.
   - Đồ án khẳng định rằng data science là công cụ mạnh mẽ để giải quyết các vấn đề thực tiễn trong xã hội, đặc biệt trong lĩnh vực giáo dục, giúp chuyển đổi dữ liệu thành những hiểu biết có thể tác động tích cực đến việc nâng cao chất lượng học tập và phát triển con người.

IV. Tài liệu tham khảo

- "Students Performance - Multiple Linear Regression" - SANJAY BORA – URL: https://www.kaggle.com/code/sanjaybora143/students-performance-multiple-linear-regression
- "StudentPerformance_Regression" – ESMATIN – URL: https://www.kaggle.com/code/esmatn/studentperformance-regression
- "[Week 08] Lab 4: OLS Linear Regression" – Class OLSLinearRegression

V. Acknowledgement

- Đồ án có sự giúp đỡ của bạn Đạt trong việc so đáp án cuối cùng. Có sự gợi ý của bạn Long trong phần thiết kế mô hình.
- Có sự giúp đỡ của Claude/ChatGPT trong việc:
  - Tạo ra các hàm vẽ biểu đồ.
  - Tìm hiểu về cách hoạt động của K-fold cross validation.
  - Gợi ý các mô hình có ý nghĩa trong cuộc sống.
  - Viết report phần phân tích dữ liệu, giải thích và nhận xét về các kết quả đạt được.