ĐẠI HỌC QUỐC GIA
THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

MTH00057 - Toán ứng dụng và thống kê cho Công nghệ thông tin

BÁO CÁO ĐỒ ÁN 3

Linear Regression

Họ tên	**MSSV**
Bùi Minh Duy	23127040

Giảng viên hướng dẫn

Nguyễn Văn Quang Huy
Trần Hà Sơn
Nguyễn Đình Thúc
Nguyễn Ngọc Toàn

Ngày 14 tháng 8 năm 2025

Mục lục

1 Ý tưởng thực hiện

1.1 Tổng quan về đồ án

Đồ án 3 tập trung vào việc xây dựng mô hình **Linear Regression** để dự đoán chỉ số thành tích học tập của sinh viên (Academic Student Performance Index). Mục tiêu chính là tìm hiểu các yếu tố ảnh hưởng đến kết quả học tập và xây dựng các mô hình dự đoán hiệu quả.

1.2 Input và Output

Input:

- **Tập huấn luyện:** p03.train.csv (9000 mẫu)
- **Tập kiểm tra:** p03.test.csv (1000 mẫu)
- **Đặc trưng đầu vào:** 5 thuộc tính
  - Hours Studied: Số giờ học tập
  - Previous Scores: Điểm số các bài kiểm tra trước
  - Extracurricular Activities: Hoạt động ngoại khóa (0/1)
  - Sleep Hours: Số giờ ngủ
  - Sample Question Papers Practiced: Số bài kiểm tra mẫu đã luyện tập

Output:

- **Dự đoán Performance Index:** Chỉ số thành tích học tập được dự đoán cho từng sinh viên
- **Các mô hình hồi quy tuyến tính** với độ chính xác khác nhau
- **Đánh giá hiệu suất** bằng độ đo MSE (Mean Squared Error)
- **Công thức toán học** cho mỗi mô hình với các hệ số được tính toán

1.3 Mục tiêu

1. **Phân tích khám phá dữ liệu (EDA):** Hiểu rõ đặc điểm và mối quan hệ giữa các thuộc tính với thành tích học tập
2. **Xây dựng mô hình hồi quy tuyến tính:**
   - Mô hình sử dụng toàn bộ 5 đặc trưng
   - Mô hình sử dụng 1 đặc trưng tốt nhất
   - Mô hình tùy chỉnh sinh viên tự xây dựng
3. **So sánh và đánh giá:** Tìm ra mô hình có hiệu suất tốt nhất

1.4 Ý tưởng giải quyết

1.4.1 Phương pháp tiếp cận

1. Phân tích khám phá dữ liệu:

- Sử dụng thống kê mô tả để hiểu phân phối dữ liệu
- Trực quan hóa bằng các biểu đồ: histogram, boxplot, scatter plot, correlation matrix
- Phát hiện outliers và missing values
- Phân tích mối tương quan giữa các đặc trưng

2. Xây dựng mô hình Linear Regression:

- Cài đặt mô hình Linear Regression sử dụng Normal Equation: $\beta = (X^T X)^{-1} X^T y$
- Áp dụng k-fold Cross Validation để đánh giá độ tin cậy
- Sử dụng MSE làm độ đo đánh giá hiệu suất

3. Kỹ thuật đặc trưng:

- Tạo các đặc trưng mới từ sự kết hợp của các đặc trưng gốc
- Áp dụng đặc trưng đa thức
- Chuẩn hóa dữ liệu (chuẩn hóa Z-score)
- Tạo đặc trưng tương tác để nắm bắt mối quan hệ phức tạp

1.4.2 Kiến trúc hệ thống

1. **Module xử lý dữ liệu:**
    - Hàm đọc dữ liệu: Đọc và xử lý dữ liệu từ tập tin CSV
    - Hàm trực quan hóa EDA: Tạo 6 biểu đồ phân tích khám phá dữ liệu
    - Quản lý biểu đồ: Tự động tạo thư mục và lưu biểu đồ
2. **Module cài đặt mô hình chính:**
    - Lớp `LinearRegression`: Cài đặt mô hình hồi quy tuyến tính
    - Hàm `calculate_mse()`: Tính toán MSE với mảng numpy
    - Hàm `k_fold_cross_validation()`: Đánh giá chéo k-fold với hạt giống ngẫu nhiên
    - Hàm `display_model_formula()`: Hiển thị công thức toán học của mô hình
3. **Module kỹ thuật đặc trưng:**
    - **Mô hình 1**: Sử dụng 2 đặc trưng tốt nhất

- **Mô hình 2**: Sử dụng 3 đặc trưng tốt nhất
- **Mô hình 3**: Đặc trưng bậc 2
- **Mô hình 4**: Đặc trưng chuẩn hóa Z-score
- **Mô hình 5**: Đặc trưng tương tác và kỹ thuật

4. **Module đánh giá và báo cáo:**
    - So sánh cross-validation
    - Tính toán MSE trên tập kiểm tra
    - Lựa chọn mô hình tốt nhất dựa trên điểm số cross-validation
    - Hiển thị công thức với ký hiệu toán học
    - Phân tích hiệu suất và nhận xét kết quả

1.4.3 Chiến lược lựa chọn đặc trưng

Dựa trên phân tích EDA, đề xuất các chiến lược:

- **Đặc trưng hàng đầu:** Kết hợp 2-3 đặc trưng có tương quan cao nhất
- **Phương pháp đa thức:** Sử dụng Previous Scores và bình phương của nó
- **Chuẩn hóa:** Chuẩn hóa để cân bằng tầm quan trọng các đặc trưng
- **Đặc trưng tương tác:** Tạo đặc trưng mới phản ánh mối quan hệ phức tạp
- **Đặc trưng kỹ thuật:** Tạo chỉ số tổng hợp như "hiệu quả học tập", "chỉ số cân bằng"

2 Chi tiết thực hiện

2.1 Cấu trúc chương trình

2.1.1 Thư viện sử dụng

Chương trình sử dụng các thư viện Python sau:

- **pandas**: Xử lý và thao tác dữ liệu dạng DataFrame, đọc tập tin CSV
- **numpy**: Tính toán số học, đại số tuyến tính và các phép toán ma trận
- **matplotlib.pyplot**: Tạo biểu đồ và trực quan hóa dữ liệu
- **seaborn**: Tạo biểu đồ thống kê chuyên sâu và ma trận tương quan
- **os**: Quản lý thư mục, đường dẫn tập tin và hệ thống

2.2 Module xử lý dữ liệu

2.2.1 Đọc và xử lý dữ liệu

- Sử dụng `pd.read_csv()` để đọc tập tin 'p03.train.csv' (9000 mẫu) và 'p03.test.csv' (1000 mẫu)
- Lưu trữ dữ liệu gốc trong `train_raw` và `test_raw`
- Sử dụng `drop_duplicates(keep='first')` để loại bỏ dữ liệu trùng lặp
- Báo cáo chi tiết số lượng dòng ban đầu, số dòng trùng lặp và số dòng còn lại
- Lưu dữ liệu đã được làm sạch vào `train` và `test`

2.2.2 Phân tích khám phá dữ liệu (EDA)

Thực hiện phân tích toàn diện trên tập huấn luyện:

Thống kê mô tả:

- Sử dụng `train.info()` để kiểm tra kiểu dữ liệu và missing values
- Sử dụng `train.describe()` để có thống kê tổng quan
- Kiểm tra `train.isnull().sum()` để xác nhận không có giá trị thiếu

Trực quan hóa dữ liệu - Tạo 6 biểu đồ chính:

- **Figure 1**: Distribution of Performance Index - Biểu đồ phân phối của biến mục tiêu
- **Figure 2**: Boxplots for numerical features - Biểu đồ hộp cho các đặc trưng số
- **Figure 3**: Distribution of Extracurricular Activities - Phân phối hoạt động ngoại khóa
- **Figure 4**: Correlation Matrix - Ma trận tương quan giữa các đặc trưng

- **Figure 5**: Scatter plots relationships - Biểu đồ phân tán mối quan hệ với Performance Index
- **Figure 6**: Performance comparison by Extracurricular - So sánh hiệu suất theo hoạt động ngoại khóa

2.3 Module cài đặt mô hình chính

2.3.1 Lớp `LinearRegression`

Mục đích: Cài đặt hoàn chỉnh mô hình hồi quy tuyến tính sử dụng Normal Equation.

Thuộc tính:

- `coefficients`: Mảng các hệ số $\beta_1, \beta_2, ..., \beta_n$
- `intercept`: Hệ số chặn β_0
- `feature_names`: Danh sách tên các đặc trưng

Phương thức `fit(X, y)`:

1. Chuyển đổi dữ liệu đầu vào thành numpy arrays
2. Lưu trữ tên đặc trưng nếu X là DataFrame
3. Thêm cột bias (cột toàn số 1) cho intercept: `X_with_bias`
4. Áp dụng Normal Equation: $\beta = (X^T X)^{-1} X^T y$
    - Tính `XtX = np.dot(X_with_bias.T, X_with_bias)`
    - Tính `XtX_inv = np.linalg.inv(XtX)`
    - Tính `Xty = np.dot(X_with_bias.T, y)`
    - Kết quả: `beta = np.dot(XtX_inv, Xty)`
5. Xử lý ngoại lệ ma trận suy biến bằng pseudo-inverse
6. Tách intercept và coefficients từ vector beta

Phương thức `predict(X)`:

- Áp dụng công thức: $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$
- Sử dụng `np.dot(X, self.coefficients)` để tính tích vô hướng
- Trả về mảng dự đoán

2.3.2 Hàm `calculate_mse()`

Mục đích: Tính Mean Squared Error để đánh giá hiệu suất mô hình.

Công thức toán học:

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

Thực hiện:

- Chuyển đổi `y_true` và `y_pred` thành numpy arrays
- Tính số lượng mẫu: `n = len(y_true)`
- Áp dụng công thức: `mse = np.sum((y_true - y_pred) ** 2) / n`

2.3.3 Hàm `k_fold_cross_validation()`

Mục đích: Thực hiện đánh giá chéo k-fold để đánh giá độ tin cậy mô hình.

Tham số:

- `X`: Ma trận đặc trưng
- `y`: Vector mục tiêu
- `k=5`: Số fold (mặc định 5) - Cân bằng giữa độ chính xác và chi phí tính toán
- `random_state=42`: Hạt giống ngẫu nhiên (giá trị phổ biến trong Machine Learning) để đảm bảo tái tạo kết quả

Quy trình thực hiện:

1. Thiết lập hạt giống ngẫu nhiên: `np.random.seed(random_state)`
2. Tạo và xáo trộn chỉ số mẫu: `np.random.shuffle(indices)`
3. Chia dữ liệu thành k folds với `fold_size = n_samples // k`
4. Với mỗi fold i:
    - Xác định chỉ số test: từ `i * fold_size` đến `(i+1) * fold_size`
    - Chỉ số train: tất cả mẫu còn lại
    - Chia dữ liệu theo chỉ số
    - Huấn luyện mô hình LinearRegression trên fold train
    - Dự đoán trên fold test và tính MSE
5. Trả về MSE trung bình: `np.mean(mse_scores)`

2.3.4 Hàm `display_model_formula()`

Mục đích: Hiển thị công thức hồi quy dưới dạng toán học dễ đọc.

Thực hiện:

- Bắt đầu với intercept: `f"Student Performance = {model.intercept:.3f}"`
- Duyệt qua từng đặc trưng và hệ số tương ứng
- Xử lý dấu "+/-"tự động dựa trên giá trị hệ số
- Định dạng: `f"{sign}{coef:.3f}×{feature}"`
- In ra công thức hoàn chỉnh

2.4 Module kỹ thuật đặc trưng

2.4.1 Thiết kế 5 mô hình tùy chỉnh

Mô hình 1: Top 2 đặc trưng

- **Hàm**: `create_model1_features(X)`
- **Đặc trưng**: Hours Studied + Previous Scores
- **Cơ sở**: Dựa trên phân tích tương quan cao nhất từ EDA
- **Thực hiện**: `return X[['Hours Studied', 'Previous Scores']]`

Mô hình 2: Top 3 đặc trưng

- **Hàm**: `create_model2_features(X)`
- **Đặc trưng**: Hours Studied + Previous Scores + Extracurricular Activities
- **Cơ sở**: Mở rộng từ Model 1 với thêm yếu tố hoạt động ngoại khóa
- **Thực hiện**: `return X[['Hours Studied', 'Previous Scores', 'Extracurricular Activities']]`

Mô hình 3: Đặc trưng đa thức

- **Hàm**: `create_model3_features(X)`
- **Đặc trưng**: Previous Scores + Previous Scores^2
- **Cơ sở**: Nắm bắt mối quan hệ phi tuyến
- **Thực hiện**:
  - `X_new = X[['Previous Scores']].copy()`
  - `X_new['Previous_Scores_squared'] = X['Previous Scores'] ** 2`

Mô hình 4: Chuẩn hóa Z-score

- **Hàm**: `create_model4_features(X)`
- **Đặc trưng**: Hours Studied và Previous Scores được chuẩn hóa
- **Công thức chuẩn hóa**: $z = \frac{x-\mu}{\sigma}$

- **Thực hiện**:
  - Tính mean và std cho mỗi đặc trưng
  - `Hours_Normalized = (Hours - mean) / std`
  - `Previous_Normalized = (Previous - mean) / std`

Mô hình 5: Đặc trưng tương tác và kỹ thuật

- **Đặc trưng kỹ thuật:**
  - **Hours_x_Previous**: `X['Hours Studied'] * X['Previous Scores']`
  - **Study_Efficiency**: `X['Hours Studied'] * X['Sample Question Papers Practiced']`
  - **Total_Capability**: `X['Previous Scores'] + X['Extracurricular Activities'] * 10`
  - **Balance_Index**: `X['Previous Scores'] / (X['Hours Studied'] + 1)`
- **Ý nghĩa**:
  - Tương tác giữa số giờ học và điểm số trước đó
  - Hiệu quả học tập qua việc luyện tập
  - Tổng năng lực với bonus từ hoạt động ngoại khóa (nhân 10 để tạo trọng số có ý nghĩa so với Previous Scores)
  - Chỉ số cân bằng đánh giá hiệu quả (tránh chia 0 bằng cách cộng thêm 1 vào Hours Studied)

2.5 Module đánh giá và báo cáo

2.5.1 Yêu cầu 2a: Mô hình 5 đặc trưng

Quy trình thực hiện:

1. Khởi tạo mô hình: `model_2a = LinearRegression()`
2. Huấn luyện: `model_2a.fit(X_train, y_train)`
3. Dự đoán: `y_pred_2a = model_2a.predict(X_test)`
4. Đánh giá: `mse_2a = calculate_mse(y_test, y_pred_2a)`
5. Hiển thị công thức: `display_model_formula(model_2a, X_train.columns)`

2.5.2 Yêu cầu 2b: Đặc trưng tốt nhất

Quy trình so sánh:

1. Định nghĩa danh sách 5 đặc trưng

2. Với mỗi đặc trưng:
    - Tạo `X_single = X_train[[feature]]`
    - Thực hiện 5-fold cross validation
    - Lưu kết quả MSE vào dictionary
3. Tìm đặc trưng tốt nhất: `min(cv_results, key=cv_results.get)`
4. Huấn luyện lại mô hình với đặc trưng tốt nhất
5. Đánh giá trên tập test

2.5.3 Yêu cầu 2c: Mô hình tùy chỉnh

Quy trình đánh giá:

1. Định nghĩa dictionary chứa 5 mô hình và hàm tạo đặc trưng
2. Với mỗi mô hình:
    - Gọi hàm tạo đặc trưng: `X_features = feature_func(X_train)`
    - Thực hiện 5-fold cross validation
3. Lựa chọn mô hình tốt nhất dựa trên MSE thấp nhất
4. Huấn luyện lại trên toàn bộ tập train
5. Đánh giá trên tập test

3 Kết quả và Kết luận

3.1 Kết quả Phân tích Khám phá Dữ liệu

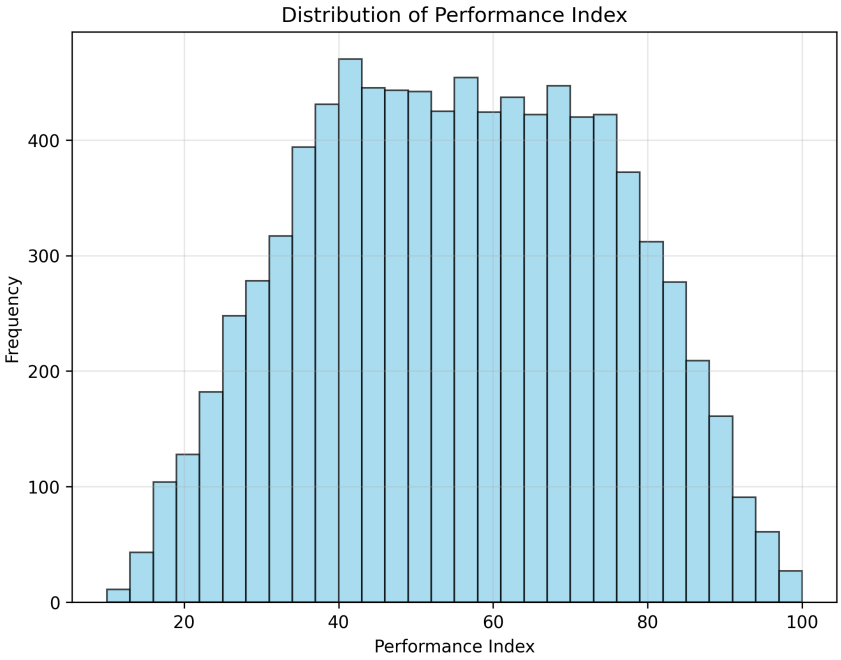
3.1.1 Thông tin tổng quan về dữ liệu

Sau quá trình đọc và làm sạch dữ liệu:

- **Tập huấn luyện**: Còn lại 8896 mẫu (đã loại 103 mẫu trùng lặp)
- **Tập kiểm tra**: Còn lại 999 mẫu (đã loại 1 mẫu trùng lặp)
- **Đặc trưng**: 5 đặc trưng đầu vào và 1 biến mục tiêu (Performance Index)
- **Chất lượng dữ liệu**: Không có giá trị thiếu (missing values)

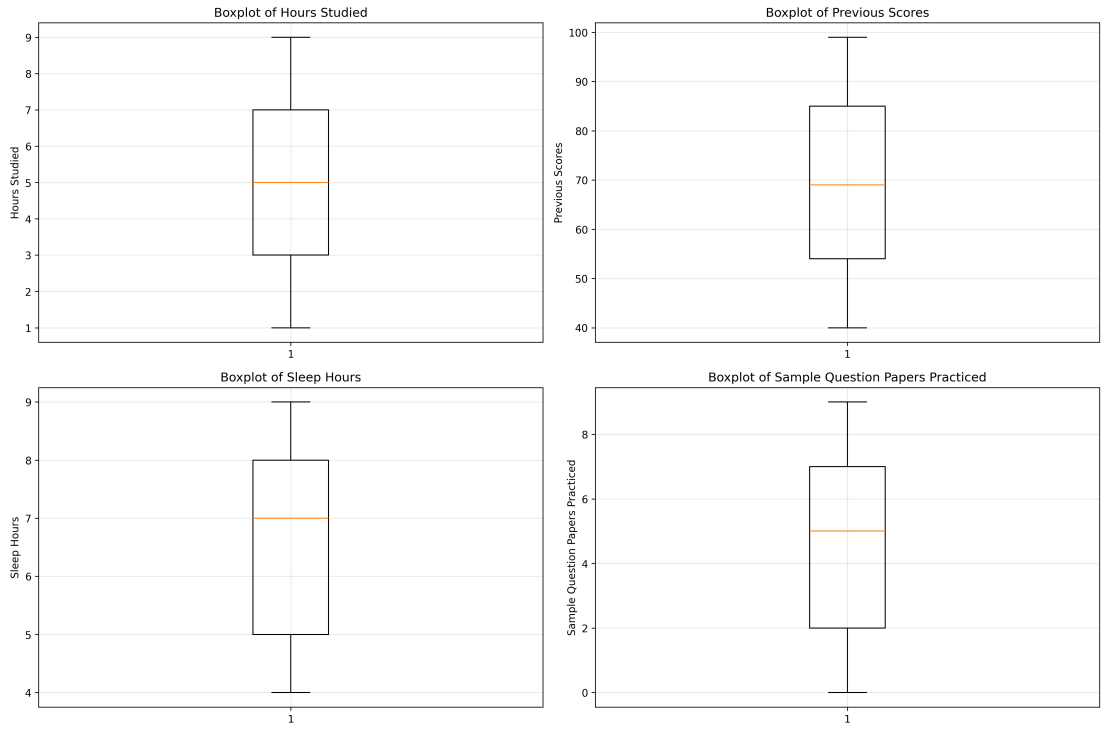
3.1.2 Phân tích trực quan hóa dữ liệu

Từ 6 biểu đồ được tạo ra trong quá trình EDA:



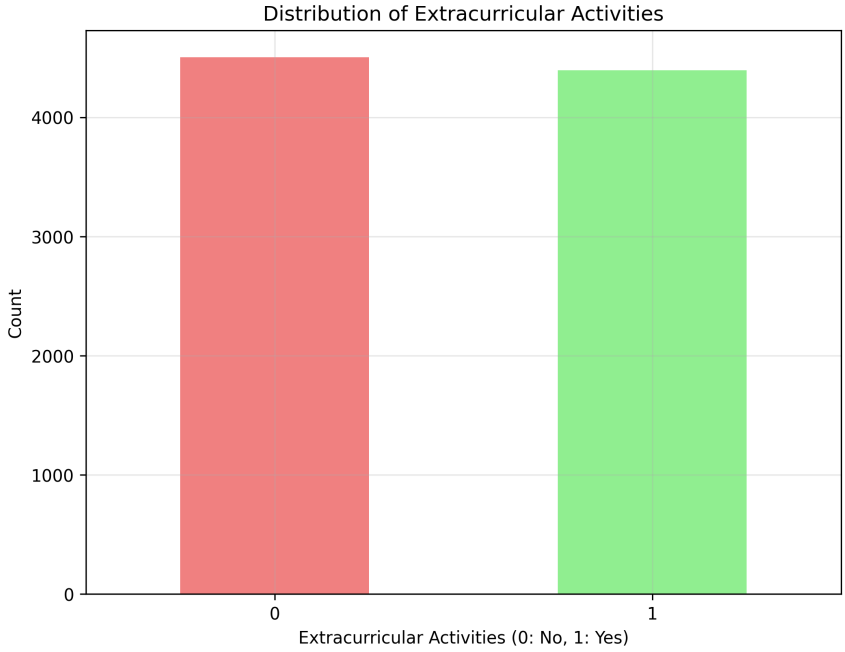
Hình 1: Phân phối Performance Index

- **Nhận xét:**
  - **Phân phối gần chuẩn**: Biểu đồ histogram cho thấy Performance Index có phân phối gần như chuẩn với đỉnh tại khoảng 40-45 điểm.
  - **Tính đối xứng**: Phân phối tương đối đối xứng với độ lệch nhẹ về phía phải.
  - **Phạm vi rộng**: Dữ liệu phân bố từ khoảng 10 đến gần 100 điểm, cho thấy sự đa dạng lớn trong hiệu suất học tập.
  - **Tập trung chính**: Phần lớn sinh viên có điểm số trong khoảng 30-70, với mật độ cao nhất ở khoảng 40-50 điểm.



Hình 2: Biểu đồ hộp các đặc trưng số

- **Nhận xét:**
  - **Hours Studied**: Phân phối đối xứng với trung vị khoảng 5 giờ, IQR từ 3-7 giờ, phạm vi từ 1-9 giờ. Không có outliers đáng kể.
  - **Previous Scores**: Phân phối đối xứng với trung vị khoảng 70 điểm, IQR từ 55-85 điểm, phạm vi từ 40-100 điểm. Phân phối cân bằng và ổn định.
  - **Sleep Hours**: Phân phối đối xứng với trung vị khoảng 7 giờ, IQR từ 5-8 giờ, phạm vi từ 4-9 giờ. Phân bố khá tập trung.
  - **Sample Question Papers**: Phân phối đối xứng với trung vị khoảng 5 đề, IQR từ 2-7 đề, phạm vi từ 0-9 đề. Có một số sinh viên không luyện đề (giá trị 0).

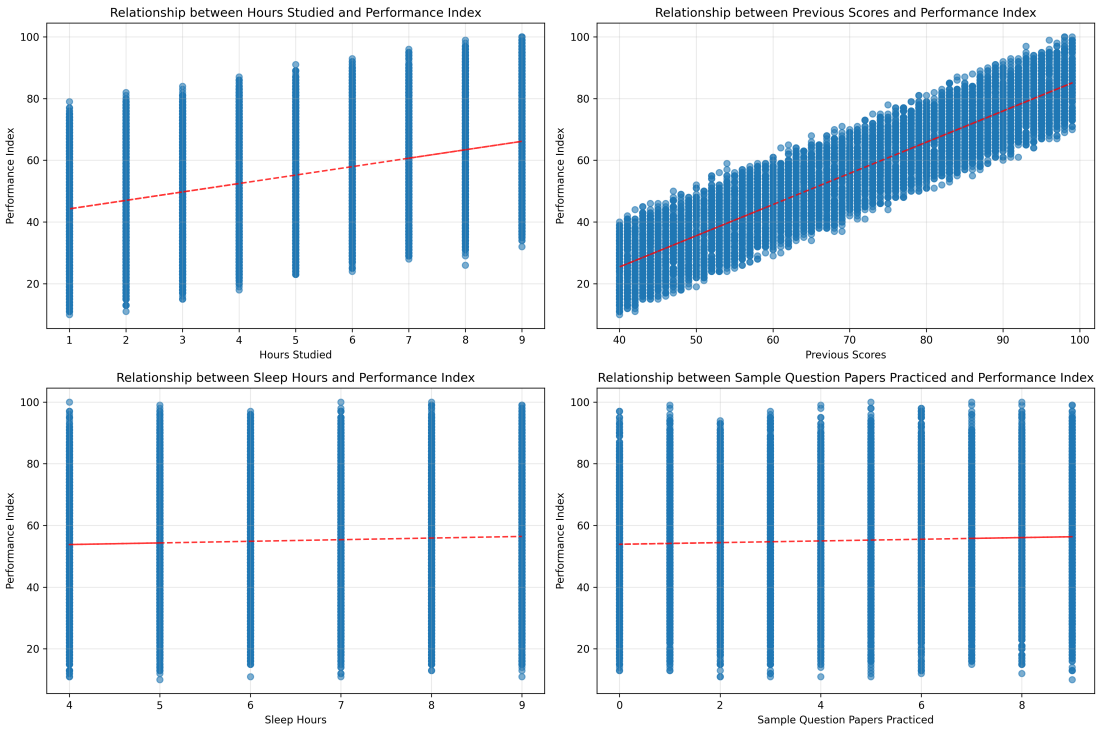


Hình 3: Phân phối Hoạt động Ngoại khóa

- **Nhận xét:**
  - **Phân bố cân bằng**: Số lượng sinh viên tham gia và không tham gia hoạt động ngoại khóa gần như bằng nhau.
  - **Tính đại diện**: Sự cân bằng này đảm bảo dữ liệu có tính đại diện tốt, tránh thiên lệch trong phân tích.

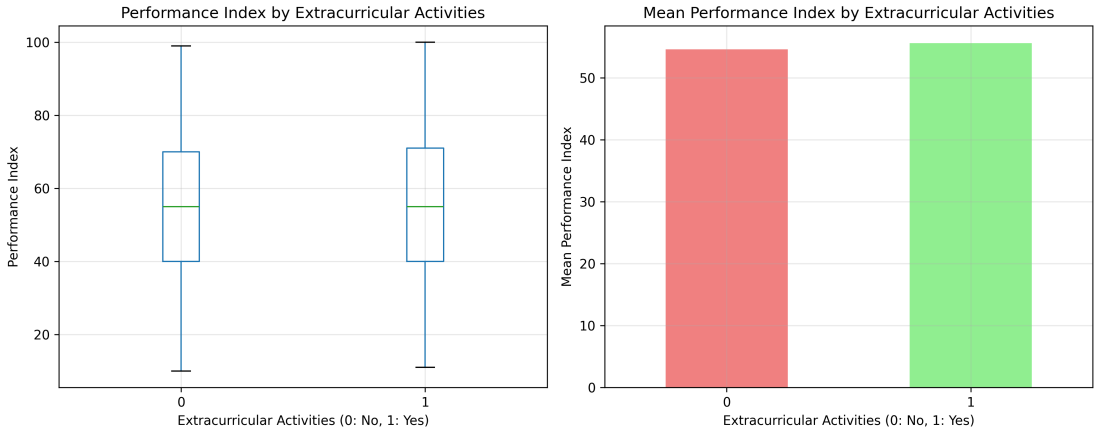
Hình 4: Ma trận tương quan giữa các đặc trưng

- **Nhận xét:**
  - Previous Scores có tương quan rất mạnh với Performance Index (0.91) → Điểm số trước đây dự đoán tốt hiệu suất.
  - Hours Studied có tương quan dương vừa phải với Performance Index (0.37) → Học nhiều giờ giúp cải thiện hiệu suất, nhưng không mạnh bằng điểm số trước.
  - Các biến khác (Extracurricular Activities, Sleep Hours, Sample Question Papers Practiced) hầu như không có tương quan đáng kể với Performance Index ($<$ 0.05).



Hình 5: Biểu đồ phân tán mối quan hệ với Performance Index

- **Nhận xét:**
  - **Previous Scores**: Có mối quan hệ tuyến tính mạnh và rõ ràng với Performance Index. Đường hồi quy cho thấy correlation coefficient cao, với sự phân bố dữ liệu khá sát đường thẳng.
  - **Hours Studied**: Mối quan hệ tuyến tính yếu hơn nhưng vẫn rõ ràng. Có xu hướng tăng nhẹ theo thời gian học, nhưng độ phân tán lớn hơn Previous Scores.
  - **Sleep Hours**: Mối quan hệ rất yếu hoặc không có correlation rõ ràng. Đường hồi quy gần như nằm ngang, cho thấy Sleep Hours ít ảnh hưởng đến Performance Index.
  - **Sample Question Papers Practiced**: Tương tự Sleep Hours, mối quan hệ rất yếu với đường hồi quy gần như nằm ngang, cho thấy việc luyện tập đề mẫu không có tác động mạnh đến hiệu suất tổng thể.



Hình 6: So sánh hiệu suất theo hoạt động ngoại khóa

- **Nhận xét:**
  - **Phân phối tương tự nhau**: Từ biểu đồ boxplot, cả hai nhóm (có và không có hoạt động ngoại khóa) đều có phân phối Performance Index khá tương đồng với median gần như bằng nhau (khoảng 55 điểm).
  - **Độ biến thiên tương đương**: Khoảng tứ phân vị (IQR) của cả hai nhóm gần như bằng nhau, cho thấy độ phân tán dữ liệu tương tự.
  - **Chênh lệch mean nhẹ**: Biểu đồ cột cho thấy sinh viên tham gia hoạt động ngoại khóa (nhóm 1) có điểm trung bình cao hơn một chút so với nhóm không tham gia (nhóm 0), nhưng sự khác biệt không lớn (khoảng 1-2 điểm).
  - **Tác động hạn chế**: Mặc dù có sự khác biệt về mean, nhưng sự chồng lấn lớn giữa hai phân phối cho thấy hoạt động ngoại khóa có tác động tích cực nhưng không quá mạnh đến hiệu suất học tập.

3.2 Kết quả các Mô hình Hồi quy Tuyến tính

3.2.1 Yêu cầu 2a: Mô hình 5 đặc trưng

Công thức mô hình:

$$\begin{aligned}\text{Performance} = &-33.961 + 2.852 \times \text{Hours Studied} \\ &+ 1.018 \times \text{Previous Scores} \\ &+ 0.606 \times \text{Extracurricular Activities} \\ &+ 0.473 \times \text{Sleep Hours} \\ &+ 0.192 \times \text{Sample Question Papers}\end{aligned} \tag{1}$$

Kết quả đánh giá:

- **MSE trên tập kiểm tra**: 4.092356
- **Nhận xét**: Mô hình sử dụng toàn bộ 5 đặc trưng cho kết quả khá tốt, với Previous Scores và Hours Studied có trọng số cao nhất.

3.2.2 Yêu cầu 2b: Mô hình đặc trưng tốt nhất

Kết quả k-fold Cross Validation:

STT	Mô hình với 1 đặc trưng	MSE
1	Hours Studied	317.510929
2	Previous Scores	60.141086
3	Extracurricular Activities	367.670027
4	Sleep Hours	367.287602
5	Sample Question Papers Practiced	367.485440

Bảng 1: Kết quả Cross Validation cho mô hình 1 đặc trưng

Đặc trưng tốt nhất: Previous Scores với MSE = 60.141086

Công thức mô hình:

$$\text{Performance} = -15.015 + 1.011 \times \text{Previous Scores} \tag{2}$$

Kết quả đánh giá:

- **MSE trên tập kiểm tra**: 58.906757
- **Giải thích**: Previous Scores là đặc trưng quan trọng nhất vì nó phản ánh trực tiếp năng lực học tập trước đó của sinh viên, có tương quan mạnh với kết quả hiện tại.

3.2.3 Yêu cầu 2c: Mô hình tùy chỉnh

Các mô hình được thiết kế:

STT	Mô hình	MSE
1	2 đặc trưng tốt nhất	5.220514
2	3 đặc trưng tốt nhất	5.136711
3	Đặc trưng đa thức	60.180196
4	Đặc trưng chuẩn hóa z-score	5.220514
5	Đặc trưng tương tác và kỹ thuật	18.505961

Bảng 2: Kết quả Cross Validation cho các mô hình tùy chỉnh

Mô hình tốt nhất: Model 2 (Top 3 features) với MSE = 5.136711

Công thức mô hình:

$$\begin{aligned}\text{Performance} = &-30.019 + 2.855 \times \text{Hours Studied} \\ &+ 1.018 \times \text{Previous Scores} \\ &+ 0.580 \times \text{Extracurricular Activities}\end{aligned} \tag{3}$$

Kết quả đánh giá:

- **MSE trên tập kiểm tra**: 5.273269
- **Giải thích**: Mô hình kết hợp 3 đặc trưng quan trọng nhất cho kết quả tốt nhất, loại bỏ được nhiễu từ các đặc trưng ít quan trọng.

3.3 Phân tích và So sánh các Mô hình

3.3.1 Bảng tổng hợp kết quả

Mô hình	MSE Cross Validation	MSE Test
Mô hình 5 đặc trưng (2a)	-	4.092356
Đặc trưng tốt nhất (2b)	60.141086	58.906757
Mô hình tùy chỉnh tốt nhất (2c)	5.136711	5.273269

Bảng 3: So sánh kết quả các mô hình

3.3.2 Nhận xét về hiệu suất

Thứ tự hiệu suất (từ tốt đến kém):

1. **Mô hình 5 đặc trưng (2a)**: MSE = 4.092356
2. **Mô hình tùy chỉnh tốt nhất (2c)**: MSE = 5.273269
3. **Đặc trưng tốt nhất (2b)**: MSE = 58.906757

Phân tích hiệu suất:

- Mô hình 5 đặc trưng (2a) đạt hiệu suất tốt nhất với MSE thấp nhất, cho thấy sự đóng góp tích cực của tất cả các đặc trưng
- Mô hình tùy chỉnh (2c) có hiệu suất tốt thứ hai, chứng tỏ việc lựa chọn đặc trưng thông minh có thể cạnh tranh với mô hình đầy đủ
- Mô hình 1 đặc trưng (2b) có hiệu suất thấp nhất do hạn chế thông tin đầu vào

3.4 Kết luận cuối cùng

Những yếu tố quan trọng ảnh hưởng đến thành tích học tập theo thứ tự ưu tiên:

1. **Previous Scores**: Yếu tố quan trọng nhất, phản ánh nền tảng kiến thức
2. **Hours Studied**: Thời gian học tập có tác động mạnh đến kết quả
3. **Extracurricular Activities**: Hoạt động ngoại khóa góp phần nâng cao hiệu suất học tập
4. **Sleep Hours**: Có tác động nhưng không mạnh bằng các yếu tố trên
5. **Sample Question Papers Practiced**: Tác động thấp nhất trong các yếu tố được khảo sát

4 Acknowledgement

Đồ án có sự hỗ trợ của **GitHub Copilot** trong việc:

- Viết docstrings cho các hàm Python
- Tối ưu hóa code và xử lý lỗi
- Tạo comments giải thích cho các đoạn code phức tạp
- Hỗ trợ viết báo cáo LaTeX với định dạng chuẩn
- So sánh các mô hình

Lưu ý: Tất cả logic chính, thuật toán Linear Regression, k-fold Cross Validation và Feature Engineering đều được tự thiết kế và cài đặt. AI chỉ hỗ trợ về mặt cú pháp và định dạng.

Tài liệu

[1] Boyd, S. & Vandenberghe, L., *Introduction to Applied Linear Algebra – Vectors, Matrices, and Least Squares*, Chương 13: Least squares, Cambridge University Press, 2018.
https://web.stanford.edu/~boyd/vmls/vmls.pdf

[2] Scikit-learn developers, *Cross-validation: evaluating estimator performance*, scikit-learn Documentation.
https://scikit-learn.org/stable/modules/cross_validation.html

[3] Andrew Ng, *Machine Learning Course - Normal Equation*, Stanford CS229 Lecture Notes.
https://cs229.stanford.edu/notes2020fall/notes2020fall/cs229-notes1.pdf

[4] Wikipedia, *Mean squared error*, truy cập ngày 13/08/2025.
https://en.wikipedia.org/wiki/Mean_squared_error