BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN

BÀI TẬP LỚN

TÌM HIỂU MÔ HÌNH LINEAR REGRESSTION, DECISION TREE, RANDOM FOREST ĐỂ DỰ ĐOÁN ĐIỂM THÔNG QUA THÓI QUEN HỌC TẬP

NGÀNH: KHOA HỌC MÁY TÍNH

CHUYÊN NGÀNH: TRÍ TUỆ NHÂN TẠO VÀ KHOA HỌC DỮ LIỆU

SINH VIÊN: NGUYỄN XUÂN ĐỨC

MÃ SINH VIÊN: 12423057

MÃ LỚP: 124231

GIẢNG VIÊN HƯỚNG DẪN: PGS. TS. NGUYỄN VĂN HẬU

HƯNG YÊN – 2025

NHẬN XÉT

Nhận xét của giảng viên hướng dẫn:

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

GIẢNG VIÊN HƯỚNG DẪN

(Ký và ghi rõ họ tên)

LỜI CAM ĐOAN

Em xin cam đoan bài tập lớn "Tìm hiểu mô hình linear regression, decision tree, random forest để dự đoán điểm thi thông qua thói quen học tập" là kết quả thực hiện của bản thân em dưới sự hướng dẫn của Thầy Nguyễn Văn Hậu và thầy Nguyễn Tuấn Anh.

Những phần sử dụng tài liệu tham khảo trong bài tập lớn đã được nêu rõ trong phần tài liệu tham khảo. Các kết quả trình bày trong bài tập lớn và chương trình xây dựng được hoàn toàn là kết quả do bản thân em thực hiện.

Nếu vi phạm lời cam đoan này, em xin chịu hoàn toàn trách nhiệm trước khoa và nhà trường.

Hưng Yên, ngày 30 tháng 12 năm 2025

SINH VIÊN

(Ký, ghi rõ họ tên)

LỜI CẢM ƠN

Để có thể hoàn thành bài tập lớn này, lời đầu tiên em xin phép gửi lời cảm ơn tới bộ môn Khoa học máy tính, Khoa Công nghệ thông tin – Trường Đại học Sư phạm Kỹ thuật Hưng Yên đã tạo điều kiện thuận lợi cho em thực hiện bài tập lớn môn học này.

Đặc biệt em xin chân thành cảm ơn Thầy Nguyễn Văn Hậu đã rất tận tình hướng dẫn, chỉ bảo em trong suốt thời gian thực hiện bài tập lớn vừa qua.

Em cũng xin chân thành cảm ơn tất cả các thầy/cô trong trường đã tận tình giảng dạy, trang bị cho em những kiến thức cần thiết, quý báu để giúp em thực hiện được bài tập lớn này.

Mặc dù em đã có cố gắng, nhưng với trình độ còn hạn chế, trong quá trình thực hiện đề tài không tránh khỏi những thiếu sót. Em hi vọng sẽ nhận được những ý kiến nhận xét, góp ý của các thầy/cô về những kết quả triển khai trong bài tập lớn.

Em xin trân trọng cảm ơn!

MỤC LỤC

DANH MỤC CÁC THUẬT NGỮ

STT	Từ viết tắt	Cụm từ tiếng Anh	Diễn giải
1	ML	Machine Learning	Học máy, lĩnh vực nghiên cứu các thuật toán cho phép máy tính học từ dữ liệu mà không cần lập trình rõ ràng từng bước.
2	LR	Logistic Regression	Mô hình hồi quy logistic, dùng cho phân loại nhị phân, dự đoán xác suất một sự kiện xảy ra.
3	DT	Decision Tree	Cây quyết định, mô hình phân loại phi tuyến dựa trên các quyết định nhánh theo giá trị biến.
4	RF	Random Forest	Rừng ngẫu nhiên, tập hợp nhiều cây quyết định để cải thiện độ chính xác và giảm overfitting.

DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ

CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

1.1. Lý do chọn đề tài

Việc lựa chọn đề tài "Dự đoán điểm thi thông qua thói quen học tập" xuất phát từ nhu cầu cấp thiết trong việc cá nhân hóa giáo dục và tối ưu hóa kết quả học tập của học sinh, sinh viên. Trong kỷ nguyên số, dữ liệu về hành vi học tập không chỉ là những con số vô tri mà còn chứa đựng quy luật về sự thành công. Đề tài này tập trung khai thác mối tương quan giữa các biến số như thời gian tự học, thói quen sinh hoạt và kết quả thi cử, từ đó đưa ra những cảnh báo sớm và định hướng điều chỉnh hành vi hiệu quả.

Về mặt kỹ thuật, việc áp dụng đồng thời ba mô hình Linear Regression, Decision Tree và Random Forest cho phép bài toán được tiếp cận một cách toàn diện. Linear Regression đóng vai trò là mô hình nền tảng giúp xác định các mối quan hệ tuyến tính đơn giản và dễ diễn giải. Trong khi đó, Decision Tree giúp hình tượng hóa các quy tắc quyết định theo từng ngưỡng thói quen cụ thể, tạo sự gần gũi và dễ hiểu cho người dùng cuối. Cuối cùng, Random Forest được lựa chọn để nâng cao độ chính xác dự báo, giảm thiểu hiện tượng quá khớp và xác định tầm quan trọng của từng nhân tố ảnh hưởng. Sự kết hợp và đối chiếu giữa các thuật toán này không chỉ mang lại kết quả dự đoán tin cậy mà còn tạo ra một cái nhìn đa chiều về cách thức máy học có thể giải quyết các vấn đề thực tiễn trong giáo dục hiện đại.1.2. Mục tiêu của đề tài

1.2.1 Mục tiêu tổng quát

Xây dựng được hệ thống dự đoán kết quả học tập dựa trên dữ liệu hành vi, nhằm cung cấp một công cụ khoa học giúp học sinh và giáo viên hiểu rõ tác động của các thói quen sinh hoạt đến điểm số. Từ đó, đề tài hướng đến việc cải thiện chất lượng giáo dục thông qua việc định hướng thay đổi lộ trình học tập dựa trên bằng chứng dữ liệu

1.2.2 Mục tiêu cụ thể

- Phân tích và xử lý dữ liệu: Thu thập, làm sạch và chuẩn hóa tập dữ liệu về thói quen học tập (thời gian học, thời gian ngủ, tần suất

tham gia hoạt động ngoại khóa, mức độ sử dụng internet,...) để đảm bảo tính phù hợp cho các mô hình máy học.Tiền xử lý và phân tích dữ liệu tín dụng, bao gồm xử lý dữ liệu thiếu, dữ liệu mất cân bằng và biến phân loại.

- Xác định các nhân tố ảnh hưởng: Sử dụng các kỹ thuật phân tích dữ liệu để tìm ra những thói quen có tác động mạnh nhất (feature importance) đến biến mục tiêu là điểm thi.
- Đánh giá và so sánh hiệu năng: Sử dụng các chỉ số đo lường như R2, MAE và RMSE để so sánh sai số giữa ba mô hình, từ đó tìm ra thuật toán tối ưu nhất cho bài toán dự đoán điểm thi.
- Triển khai và huấn luyện mô hình: Áp dụng thành công ba thuật toán Machine Learning bao gồm Linear Regression để thiết lập mối liên hệ tuyến tính cơ bản giữa thói quen và điểm số; Decision Tree nhằm xây dựng các bộ quy tắc quyết định trực quan để phân loại mức độ học tập; và Random Forest để tối ưu hóa độ chính xác cũng như xử lý các mối quan hệ phi tuyến phức tạp trong dữ liệu.

1.3. Giới hạn và phạm vi của đề tài

1.3.1 Đối tượng nghiên cứu

Đối tượng nghiên cứu: Nghiên cứu tập trung vào mối quan hệ giữa các thói quen học tập, sinh hoạt và kết quả điểm thi của học sinh, sinh viên thông qua việc phân tích các biến số định lượng và định tính như thời gian tự học, thời gian ngủ, tần suất chuyên cần, mức độ sử dụng mạng xã hội, hoạt động ngoại khóa cùng các phương pháp học tập cụ thể. Đồng thời, đối tượng nghiên cứu cũng bao gồm việc ứng dụng và đánh giá hiệu năng của các thuật toán Machine Learning như Linear Regression, Decision Tree, Random Forest cùng các chỉ số đo lường liên quan để xây dựng mô hình dự báo tối ưu.

1.3.2 Phạm vi nghiên cứu

- Về nội dung: Tập trung vào việc phân tích dữ liệu hành vi để dự đoán điểm số. Đề tài không đi sâu vào các yếu tố tâm lý chuyên sâu

hay các điều kiện kinh tế - xã hội quá phức tạp mà chỉ tập trung vào những thói quen có thể đo lường được.

- kết quả mong đợi không chỉ là một con số dự đoán, mà là một bảng đánh giá toàn diện về sự hiệu quả của các thói quen học tập.
- Đề tài tập trung nghiên cứu và so sánh hiệu năng của ba thuật toán cụ thể là Linear Regression, Decision Tree và Random Forest để có thể đưa ra dự đoán chính xác nhất

1.4. Nội dung thực hiện

Nội dung chính của đề tài bao gồm các bước sau:

- Khảo sát và nghiên cứu tổng quan về bài toán dự đoán điểm thi thông qua thói quen học tập.
- Thu thập, khám phá và tiền xử lý dữ liệu.
- Xây dựng và huấn luyện các mô hình Machine Learning.
- Đánh giá, so sánh và phân tích kết quả của các mô hình.
- Lựa chọn mô hình tối ưu và triển khai thử nghiệm ứng dụng dự đoán.
- Tổng kết kết quả đạt được, đánh giá hạn chế và đề xuất hướng phát triển.

1.5. Phương pháp tiếp cận

Đề tài sử dụng kết hợp các phương pháp nghiên cứu định lượng và kỹ thuật học máy:

- Phương pháp phân tích lý thuyết: Nghiên cứu các tài liệu giáo dục học để xác định các yếu tố thói quen nào có khả năng ảnh hưởng đến kết quả học tập nhất, tạo cơ sở cho việc chọn biến..
- Phương pháp học máy có giám sát (Supervised Learning): Đây là phương pháp chủ đạo. Tiếp cận bài toán theo hướng Hồi quy (Regression) vì biến mục tiêu (điểm thi) là một giá trị liên tục..
- Sử dụng quy trình Machine Learning chuẩn gồm các bước: tiền xử lý dữ liệu – huấn luyện mô hình – đánh giá – triển khai.

- Đánh giá và đối chiếu: Kiểm thử mô hình trên tập dữ liệu mới và so sánh hiệu suất dựa trên các thang đo khoa học

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Tổng quan bài toán

Dự đoán điểm thi là bài toán ước lượng kết quả học tập của người học thông qua việc phân tích các đặc trưng hành vi và thói quen sinh hoạt tích lũy trong quá trình đào tạo. Trong bối cảnh giáo dục hiện đại, kết quả thi không chỉ phụ thuộc vào năng lực tư duy mà còn hệ quả của một chuỗi các quyết định về phân bổ thời gian và cường độ học tập. Việc dự báo sớm điểm số đóng vai trò quan trọng trong việc đưa ra các biện pháp can thiệp sư phạm kịp thời, giúp tối ưu hóa lộ trình cá nhân hóa và giảm thiểu tỷ lệ học sinh không đạt yêu cầu.

Về phương diện học máy, dự đoán điểm thi được cấu trúc dưới dạng bài toán hồi quy (regression). Mục tiêu của mô hình là tìm ra một hàm ánh xạ f: Rd→R, trong đó đầu vào là vector đặc trưng biểu diễn thói quen học tập x=(x1,x2,…,x d) — bao gồm các biến định lượng như số giờ tự học, thời gian ngủ, tỷ lệ chuyên cần và các biến định tính đã được mã hóa. Đầu ra là giá trị mục tiêu y∈[0,10] đại diện cho điểm số thực tế. Mô hình hướng tới việc tối thiểu hóa hàm tổn thất (loss function), thường là Sai số bình phương trung bình:

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

Dữ liệu thói quen học tập thường mang đặc trưng bởi tính đa chiều, sự tương quan phức tạp giữa các biến (như mối quan hệ phi tuyến giữa thời gian ngủ và sự tập trung) và sự tồn tại của các dữ liệu nhiễu do sai lệch trong quá trình tự kê khai. Do đó, bài toán đòi hỏi các mô hình không chỉ có khả năng dự báo chính xác mà còn phải cung cấp khả năng giải thích (interpretability) để xác định trọng số tác động của từng thói quen. Việc sử dụng kết hợp các thuật toán từ tuyến tính đến phi tuyến và mô hình tập hợp (Ensemble

Learning) cho phép kiểm chứng giả thuyết về sự tác động của thói quen lên kết quả học tập một cách toàn diện và ổn định nhất.

2.2. Tiền xử lý dữ liệu

Tiền xử lý dữ liệu là bước quan trọng nhằm nâng cao chất lượng dữ liệu, giúp mô hình học hiệu quả và giảm nhiễu. Các bước chính bao gồm xử lý dữ liệu thiếu, mã hóa biến phân loại, chuẩn hóa biến số và xử lý mất cân bằng lớp.

2.2.1. Xử lý dữ liệu thiếu

Dữ liệu thiếu có thể phát sinh từ học sinh không cung cấp thông tin, lỗi trong quá trình ghi nhận hành vi học tập hoặc các chỉ số không được ghi chép đồng nhất. Nếu không xử lý, dữ liệu thiếu có thể làm sai kết quả huấn luyện hoặc gây lỗi cho thuật toán.

Trong đề tài này, các biến số thiếu được điền bằng giá trị trung vị (median) của biến:

$$x_i = \begin{cases} x_i, & \text{nếu không thiếu} \\ \text{median}(X), & \text{nếu thiếu} \end{cases}$$

Trung vị được chọn vì ít bị ảnh hưởng bởi outliers và phù hợp với dữ liệu tài chính thường phân phối lệch.

2.2.2. Mã hóa biến phân loại

a) Mã hóa nhị phân (Binary Encoding)

Áp dụng cho các biến có hai giá trị (Yes/No, Y/N):

$$x = \begin{cases} 1 & \text{Yes} \\ 0 & \text{No} \end{cases}$$

Ưu điểm:

- Giữ nguyên ý nghĩa logic
- Phù hợp với mọi mô hình

b) Mã hóa có thứ tự (Ordinal Encoding)

Áp dụng cho các biến phản ánh mức độ hoặc tần suất hành vi có tính tăng dần, ví dụ như Tần suất học nhóm (Hiếm khi – Bình thường – Thường xuyên)

Hiếm khi→0, Bình Thường→1, Thường Xuyên→2

Việc giữ thứ tự giúp mô hình:

- Hiểu được mức độ tác động của thói quen tăng dần đối với kết quả thi.
- Học được quan hệ tuyến tính giữa sự nỗ lực và điểm số (ví dụ: học càng thường xuyên điểm càng cao).

c) One-Hot Encoding

Áp dụng cho các biến phân loại không có thứ tự tự nhiên, ví dụ như Phương pháp học tập chủ đạo. Với biến có k giá trị, One-Hot Encoding sẽ tạo ra k biến nhị phân tương ứng.

Ưu điểm:

- Không tạo ra giả định sai lầm về thứ tự giữa các phương pháp học tập khác nhau.
- Đặc biệt phù hợp với Linear Regression (tránh gây nhiễu trọng số) và các mô hình dựa trên cây như Decision Tree, Random Forest.

2.2.3. Chuẩn hóa dữ liệu số

a) Mục đích chuẩn hóa

Chuẩn hóa dữ liệu là bước kỹ thuật quan trọng nhằm đưa các biến số hành vi về cùng một thang đo, triệt tiêu sự chênh lệch về đơn vị đo lường. Việc này giúp tránh trường hợp các biến có biên độ giá trị lớn (ví dụ: số phút tự học lên đến hàng trăm) chi phối hoàn toàn mô hình so với các biến có biên độ nhỏ (ví dụ: số lần đi muộn). Đối với các mô hình dựa trên tối ưu hóa như Linear Regression, chuẩn hóa giúp quá trình hạ độ dốc (Gradient Descent) diễn ra ổn định, giảm số bước lặp cần thiết để đạt cực tiểu hàm mất mát và ngăn chặn tình trạng mất cân bằng trọng số giữa các đặc trưng.

b) Standardization (Z-score Normalization)

Một phương pháp phổ biến là chuẩn hóa Z-score, trong đó mỗi giá trị của biến được trừ đi trung bình và chia cho độ lệch chuẩn của biến:

$$x' = \frac{x - \mu}{\sigma}$$

Trong đó:

- μ: trung bình
- σ: độ lệch chuẩn

Phương pháp này đặc biệt phù hợp với các đặc trưng hành vi có xu hướng phân phối gần chuẩn. Trong đề tài, Z-score được sử dụng làm phương pháp chuẩn hóa chính cho các biến số liên tục (như thời gian học tập), vì nó hỗ trợ tốt cho mô hình Linear Regression và giúp duy trì được thông tin về các giá trị ngoại lai (outliers) mà không làm mất đi đặc trưng phân phối ban đầu của dữ liệu học sinh.

c) Min-Max Normalization

Phương pháp chuẩn hóa Min-Max thực hiện biến đổi dữ liệu để đưa giá trị của mỗi biến về một khoảng xác định, thông thường là [0,1]:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

Min-Max Normalization đặc biệt hữu ích khi cần giữ tỷ lệ giữa các giá trị và khi mô hình yêu cầu dữ liệu trong một phạm vi xác định. Tuy nhiên, phương pháp này khá nhạy cảm với outliers: các giá trị ngoại lai có thể làm co cụm phần lớn dữ liệu vào một phạm vi hẹp, ảnh hưởng đến hiệu quả học của mô hình. Trong đề tài, phương pháp này có thể áp dụng cho các biến số mà outliers đã được xử lý hoặc biến có phân phối gần đều.

2.2.4. Xử lý mất cân bằng lớp

Với dữ liệu tín dụng, số khách hàng vỡ nợ thường ít hơn nhiều so với khách hàng tốt. Nếu không xử lý, mô hình có thể đạt accuracy cao nhưng bỏ sót nhiều khách hàng rủi ro. Phương pháp phổ biến:

- Class Weight: Gán trọng số wyi cho từng nhãn trong hàm mất mát:

$$Loss = \sum_{i=1}^{N} w_{y_i} \cdot \ell(y_i, \hat{y}_i)$$

Nhãn ít mẫu có trọng số cao, ảnh hưởng lớn hơn khi tính loss, giúp mô hình học chú ý đến lớp thiểu số.

2.3. Các mô hình học máy áp dụng

2.3.1. Linear Regression

Công thức:

$$\hat{y} = wx + b$$

Trong đó:

- x: biến độc lập đặc trưng
- y: biến phụ thuộc
- w là vector trọng số, b là bias

Đặc điểm:

- Là mô hình học có giám sát dùng cho bài toán dự đoán giá trị liên tục (điểm thi).
- Giả định mối quan hệ tuyến tính giữa biến đầu vào và đầu ra (điểm thi).
- Các hệ số *wi* thể hiện mức độ ảnh hưởng của từng thói quen học tập đến điểm thi.
- Dễ huấn luyện, thường dùng làm baseline model trong các bài toán dự đoán.

Lý do chọn:

- Điểm thi là biến số liên tục, rất phù hợp với hồi quy tuyến tính.
- Các yếu tố học tập thường có xu hướng tăng → điểm tăng, gần tuyến tính trong nhiều trường hợp.

Ưu nhược điểm:

- Ưu điểm: Đơn giản, dễ giải thích, tính toán nhanh.
- Nhược điểm: Không xử lí tốt mối quan hệ phức tạp , nhạy cảm với outlier

2.3.2. Decision Tree

Công thức:

Cây quyết định dựa trên việc chọn split tối ưu để giảm hàm mất mát. Ví dụ sử dụng Entropy / Gini Index:

- Entropy:

$$H(S) = -\sum_{i=1}^{c} p_i \log_2 p_i$$

- Gain thông tin khi chia tập S theo feature A:

$$\text{Information Gain}(S, A) = H(S) - \sum_{v \in \text{values}(A)} \frac{|S_v|}{|S|} H(S_v)$$

Đặc điểm:

- Mô hình phi tuyến, trực quan.
- Dễ xác định các ngưỡng quyết định quan trọng.

Lý do chọn:

- Trực quan, dễ giải thích, phù hợp trình bày cho người không chuyên.
- Phát hiện tương tác phi tuyến giữa các biến.

Ưu nhược điểm:

- Ưu điểm: Trực quan, dễ giải thích, nhận biết các yếu tố quyết định rủi ro.
- Nhược điểm: Dễ overfitting với dữ liệu nhỏ hoặc nhiều biến.

2.3.3. Random Forest

Công thức tổng quát:

$$\hat{y} = \text{majority\_vote}(T_1(\mathbf{x}), T_2(\mathbf{x}), ..., T_n(\mathbf{x}))$$

Trong đó Ti(x) là dự đoán của cây thứ i.

Đặc điểm:

- Tập hợp nhiều cây quyết định (ensemble) để giảm variance.
- Kỹ thuật Bagging: bootstrap các tập con dữ liệu để huấn luyện từng cây.

Lý do chọn:

- Giảm overfitting so với cây đơn.
- Hoạt động tốt với dữ liệu tabular nhiều biến và nhiễu.

Ưu nhược điểm:

- Ưu điểm: Độ chính xác cao, robust với dữ liệu lớn và phức tạp.
- Nhược điểm: Ít trực quan, tốn tài nguyên hơn LR hoặc DT.

2.4. Các metric đánh giá mô hình

Để đánh giá hiệu quả của mô hình dự đoán điểm thi thông qua thói quen học tập, đề tài sử dụng nhiều chỉ số đánh giá. Mỗi metric cung cấp một góc nhìn khác nhau về khả năng dự đoán của mô hình, đặc biệt trong bối cảnh dữ liệu mất cân bằng.

2.4.1. MAE(MEAN ABSOLUTE ERROR)

MAE đo độ lệch trung bình tuyệt đối giữa giá trị dự đoán và giá trị thực tế.

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

- Ý nghĩa: Phản ánh trung bình dự đoán sai lệch khoảng ,dễ hiểu , ít bị ảnh hưởng bởi các Outlier
- Hạn chế: Không phạt mạnh các lỗi lớn.

2.4.2. RMSE(ROOT MEAN SQUARED ERROR)

RMSE đo căn bậc hai của sai số bình phương trung bình, nhấn mạnh các dự đoán sai lệch lớn.

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

- Ý nghĩa: Khi mô hình dự đoán sai nặng ở một số học sinh thì RMSE sẽ cao , phù hợp khi muốn hạn chế các trường hợp dự đoán lệch nhiều điểm
- Hạn chế : Nhạy cảm với Outlier, khó diễn giải hơn MAE

2.4.3. R^2 (HỆ SỐ XÁC ĐỊNH)

R^2 đo mức độ giải thích của mô hình đối với sự biến thiên của biến mục tiêu (điểm thi).

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}$$

- Ý nghĩa: R^2=0.85: mô hình giải thích được 85% sự biến thiên của điểm thi.
- R^2 càng gần 1 thì mô hình càng tốt
- $R^2 < 0$: mô hình kém hơn dự đoán trung bình
- Ưu điểm : Đánh giá tổng quát chất lượng mô hình, dễ so sánh giữa các mô hình
- Nhược điểm : Không phản ánh trực tiếp mức sai lệch dự đoán, có thể gây hiểu lầm nếu dùng riêng lẻ

2.4.4. Tổng kết lựa chọn metric

- MAE: Ưu tiên để đánh giá mức sai lệch trung bình giữa điểm dự đoán và điểm thực tế, dễ diễn giải theo đơn vị điểm thi.

- RMSE: Ưu tiên nhằm phát hiện và phạt mạnh các trường hợp dự đoán sai lệch lớn, đảm bảo mô hình không mắc lỗi nghiêm trọng với một số học sinh.
- R^2: Dùng để đánh giá khả năng giải thích tổng thể của mô hình đối với sự biến thiên điểm thi, phục vụ so sánh hiệu quả giữa các mô hình khác nhau.
- Kết hợp MAE – RMSE – R^2: Đảm bảo đánh giá mô hình toàn diện, vừa phản ánh độ chính xác, mức độ ổn định, vừa thể hiện khả năng giải thích của mô hình.

CHƯƠNG 3: CÀI ĐẶT VÀ THỰC NGHIỆM

3.1. Môi trường cài đặt và dữ liệu thực nghiệm

Môi trường cài đặt:

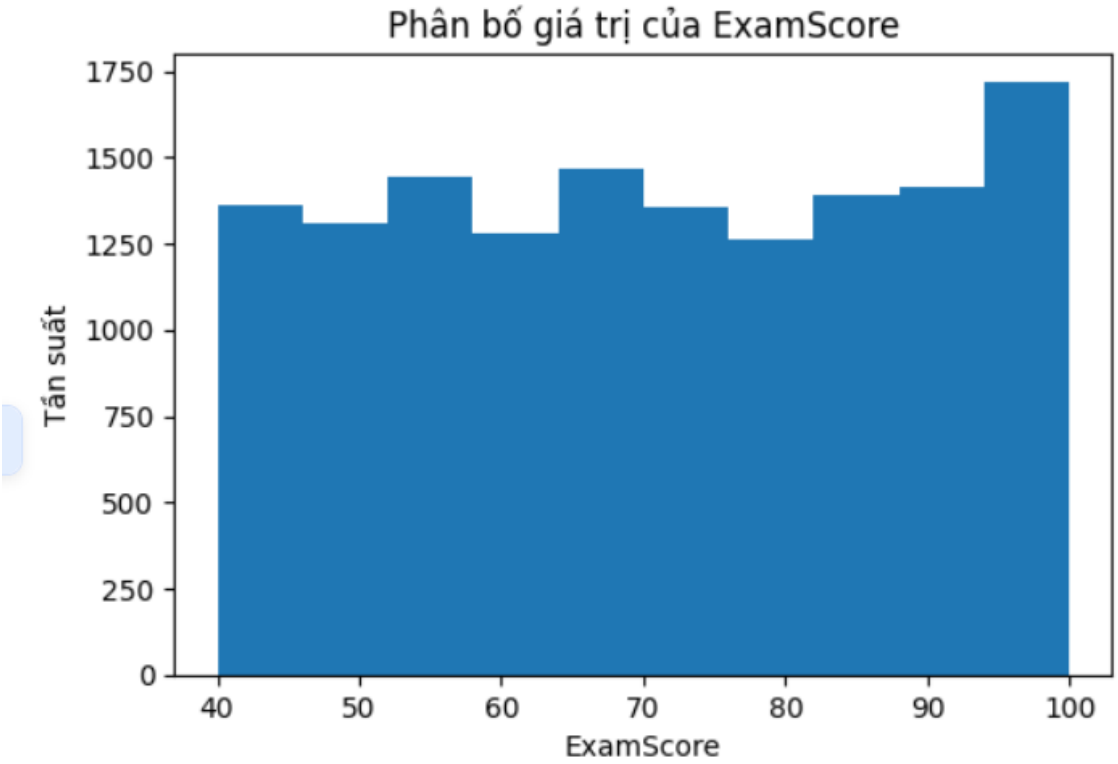
- Ngôn ngữ: Python 3.12.7
- Thư viện chính: pandas, numpy, scikit-learn, imblearn, matplotlib, seaborn
- Công cụ: google colab

Dữ liệu thực nghiệm:

- Tập dữ liệu: student_performance.csv
- Số lượng bản ghi: 14,004 dòng
- Số lượng đặc trưng: 16 cột
  + StudyHours: Số giờ tự học trung bình.
  + Attendance: Tỷ lệ đi học chuyên cần (%)
  + Resources: Mức độ tiếp cận tài nguyên học tập.
  + Extracurricular: Tham gia hoạt động ngoại khóa (1: Có, 0: Không).
  + Motivation: Mức độ động lực học tập.
  + Internet: Khả năng tiếp cận Internet (1: Có, 0: Không).
  + Gender: Giới tính học sinh/sinh viên.
  + Age: Độ tuổi của đối tượng nghiên cứu.
  + LearningStyle: Phong cách học tập chủ đạo.
  + OnlineCourses: Số lượng khóa học trực tuyến đã tham gia.
  + Discussions: Tần suất tham gia thảo luận nhóm.
  + AssignmentCompletion: Tỷ lệ hoàn thành bài tập về nhà
  + EduTech: Sử dụng công nghệ trong học tập.
  + StressLevel: Mức độ căng thẳng trong học tập
  + ExamScore: Điểm số bài thi (biến mục tiêu cho bài toán Hồi quy).
  + FinalGrade: Xếp loại học tập cuối kỳ

- Biến mục tiêu: ExamScore: Điểm số bài thi

Phân tích biến mục tiêu:



Hình 3.1 Biểu đồ phân bố giá trị của examscore

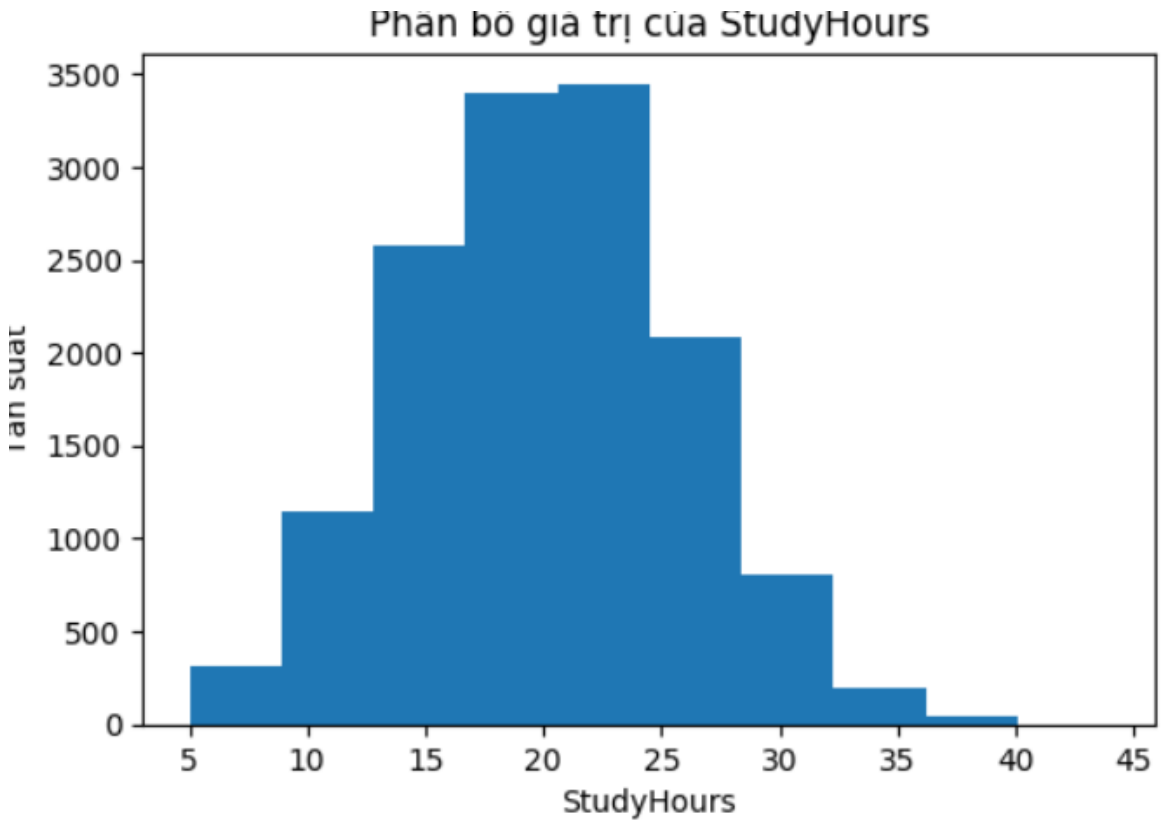
- Phạm vi điểm số: Dao động từ 40 đến 100 điểm.
- Đặc điểm nổi bật: * Tần suất cao nhất nằm ở nhóm điểm 95 - 100 (hơn 1.700 trường hợp). Cho thấy 1 lượng lớn sinh viên được xuất sắc
- Kết luận: Dữ liệu có độ phân hóa rộng, không có hiện tượng "lệch" quá mức về phía điểm thấp, phản ánh một kỳ thi có tỉ lệ sinh viên đạt điểm giỏi khá cao

Phân phối biến phân loại (Categorical EDA):

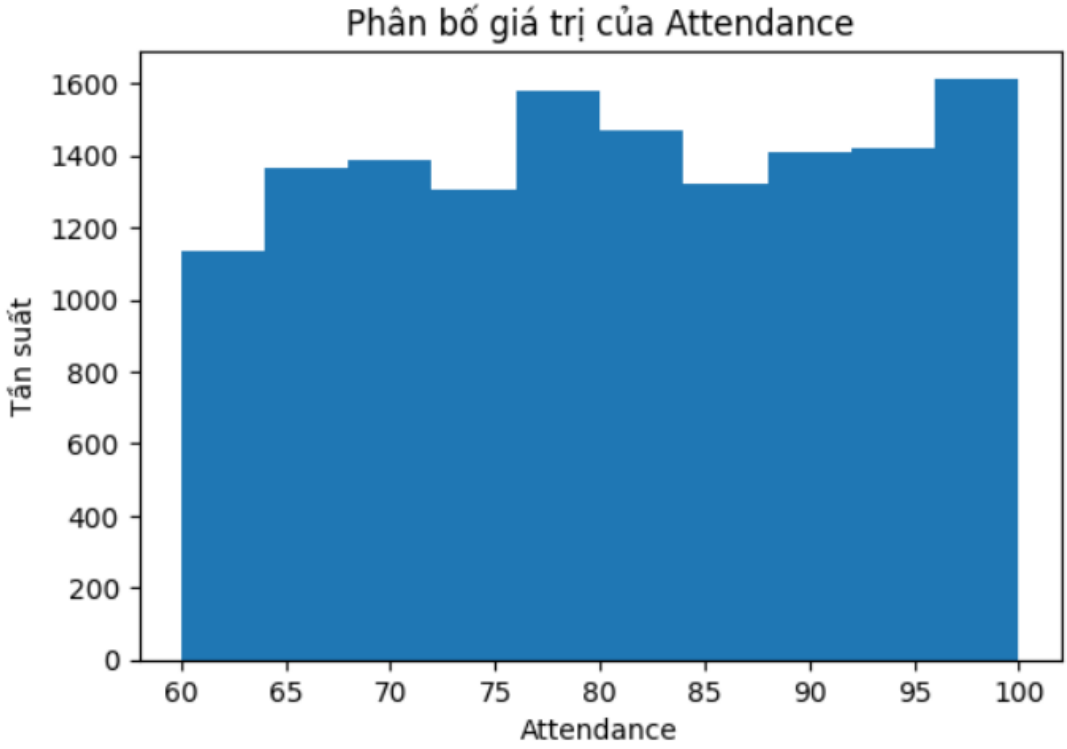
Các biến: **learningstyle, stresslevel, finalgrade.**

Phân phối biến số (Numerical EDA):

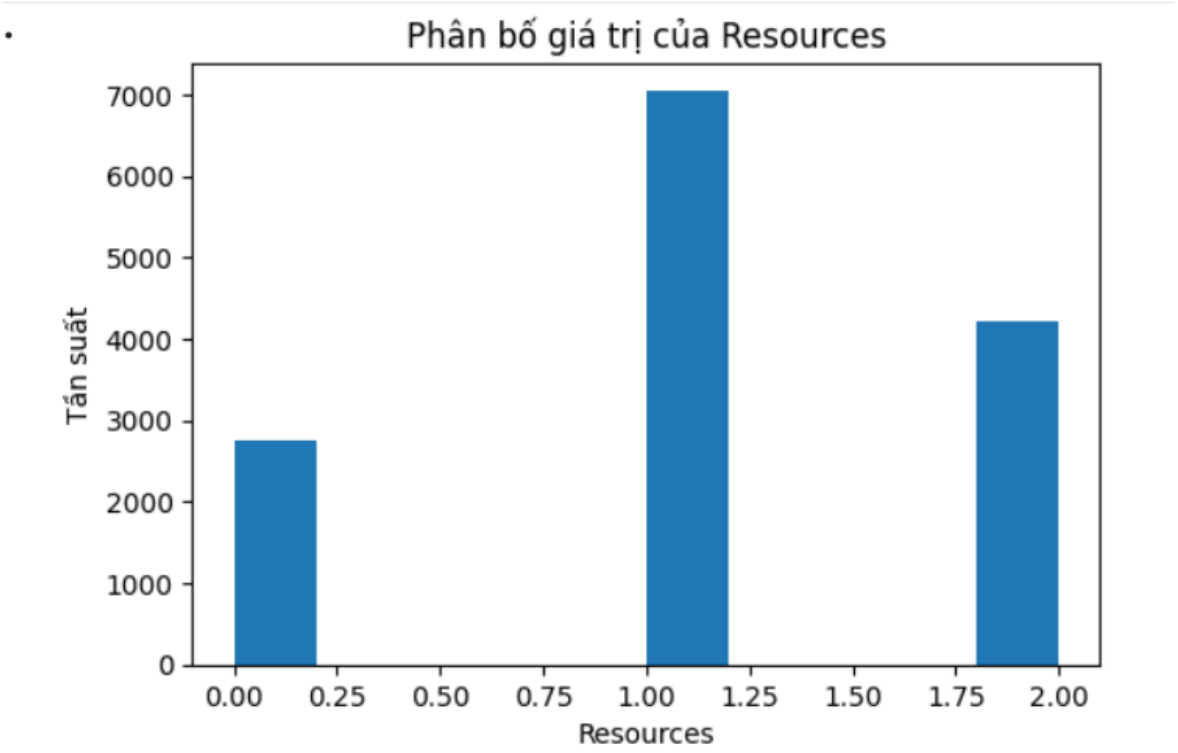
Các biến số: StudyHours, Attendance, Resources, Motivation, Age, OnlineCourses, AssignmentCompletion, ExamScore.



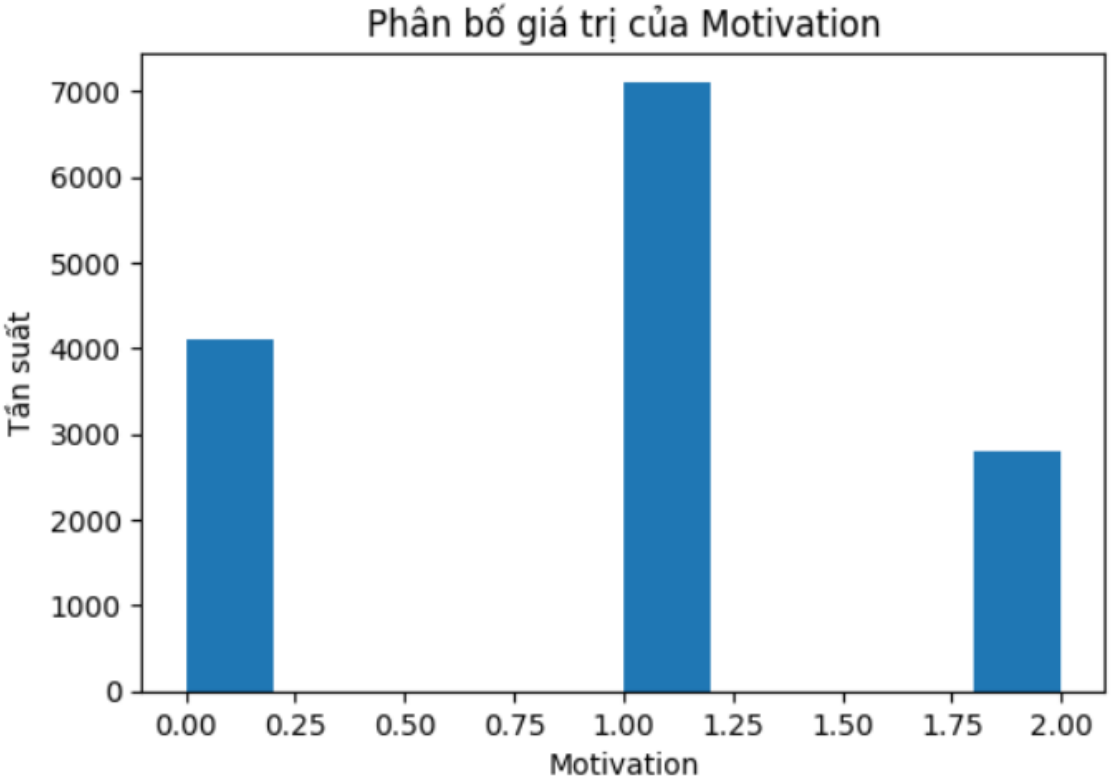
Hình 3.2 Phân bố giá trị giờ học



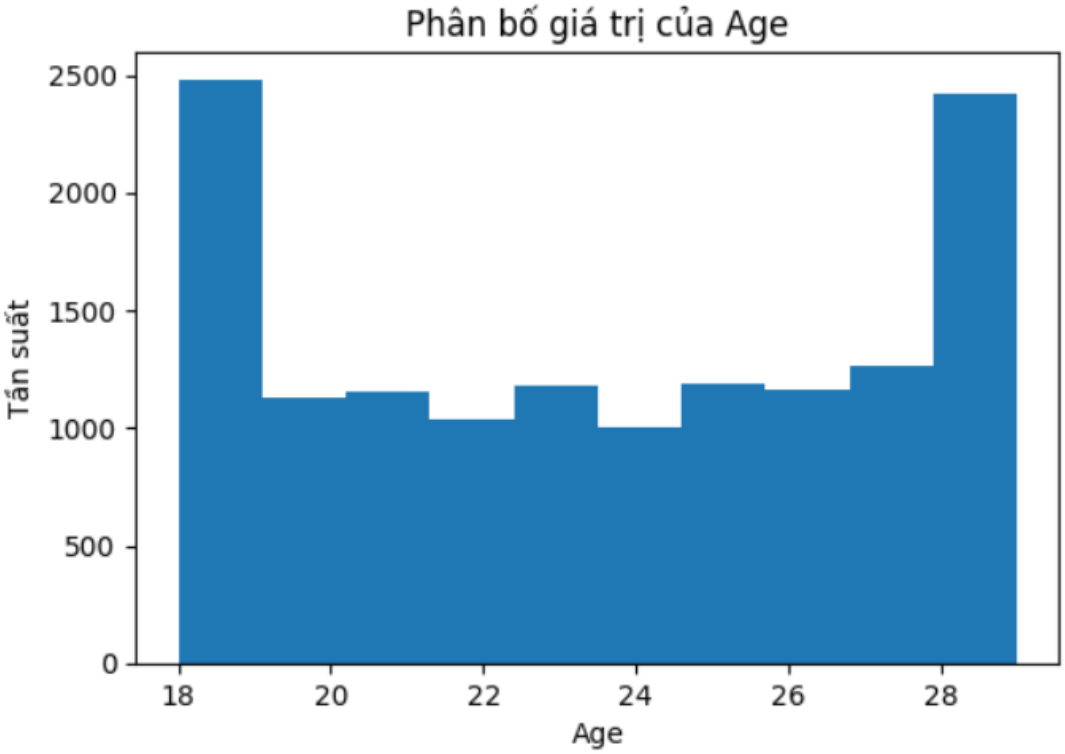
Hình 3.3 Phân bố giá trị tỉ lệ chuyên cần



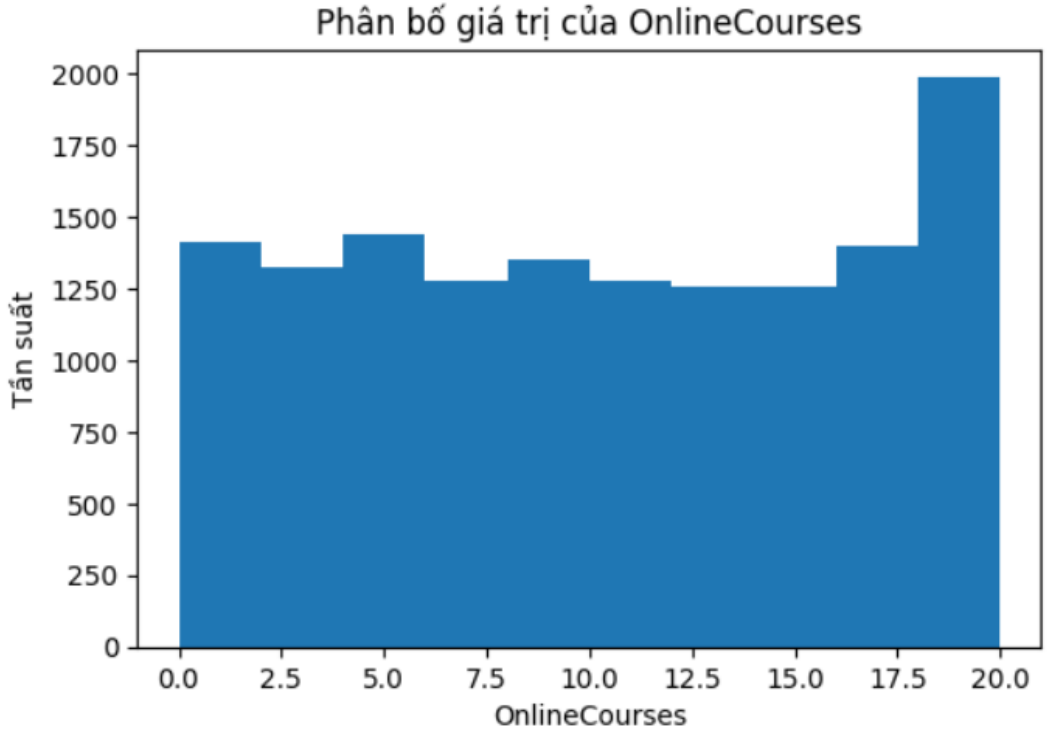
Hình 3.4 Phân bố giá trị của tham khảo tài liệu



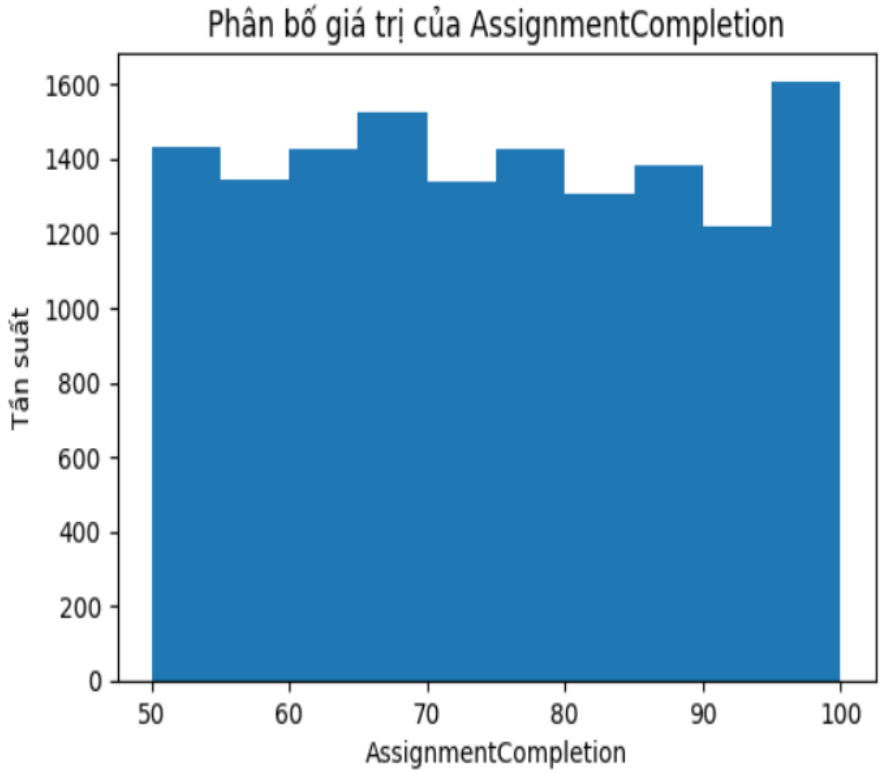
Hình 3.5 Phân bố giá trị của động lực học tập



Hình 3.6 Phân bố giá trị của tuổi



Hình 3.7 Phân bố giá trị của khóa học onl



Hình 3.8 Phân bố giá trị của hoàn thành bài tập

3.2. Quy trình xử lý dữ liệu (Data Pipeline)

Làm sạch dữ liệu: Tập dữ liệu hoàn chỉnh, không có giá trị thiếu (Missing values), giúp đảm bảo tính toàn vẹn và không gây mất mát thông tin trong quá trình huấn luyện.

Kiểm tra và loại bỏ các bản ghi trùng lặp (nếu có).

3.2.1. Mã hóa biến phân loại (Encoding)

Biến nhị phân:

- Chuyển đổi các biến có 2 giá trị (Gender, Internet,...) về định dạng 0 và 1.
- Giúp mô hình hiểu rõ lớp

Biến đa nhóm:

- Thực hiện One-Hot Encoding cho các biến danh mục không có thứ bậc (LearningStyle) để tránh gán nhầm trọng số cho các nhãn.

3.2.2. Chuẩn hóa dữ liệu số

- Sử dụng StandardScaler để đưa các biến số (StudyHours, Attendance, Age...) về cùng một thang đo với trung bình (μ) bằng 0 và độ lệch chuẩn (σ) bằng 1.

3.2.3. Chia tập dữ liệu

Tỷ lệ chia:

- Train: 70%: Dùng để huấn luyện mô hình.
- Validation: 15%: Dùng để tối ưu hóa tham số và lựa chọn mô hình.
- Test: 15%: Dùng để đánh giá khách quan hiệu suất cuối cùng.

Khi chia dữ liệu:

- Đảm bảo tính đại diện: Giữ nguyên tỷ lệ phân bổ của các nhóm điểm (hoặc xếp loại FinalGrade) trong từng tập con tương ứng với tập dữ liệu gốc.
- Ổn định hóa đánh giá: Tránh tình trạng tập Test hoặc Validation chứa quá nhiều hoặc quá ít các trường hợp đặc biệt (như sinh viên xuất sắc hoặc sinh viên trượt môn), từ đó giúp kết quả đánh giá mô hình phản ánh đúng thực tế năng lực của toàn bộ tập sinh viên.

3.3. Cài đặt các mô hình dự đoán

Trong đề tài này, ba mô hình học máy được lựa chọn để thử nghiệm và so sánh hiệu quả dự đoán điểm số: Linear Regression, Decision Tree, Random Forest. Mỗi mô hình được cấu hình với các tham số (hyperparameters) phù hợp với đặc trưng dữ liệu và mục tiêu dự đoán.

3.3.1. Linear Regression (LR)

Cấu hình chính:

- Sử dụng hàm mất mát MSE (Mean Squared Error) để tối ưu hóa đường thẳng tiệm cận dữ liệu.

Giải thích:

- Linear Regression dự đoán dựa trên công thức:

$$\hat{y} = wx + b$$

- Ưu điểm: Đơn giản, tốc độ huấn luyện và dự báo cực nhanh..
- Cung cấp khả năng giải thích mô hình tốt: Nhìn vào hệ số (w), ta biết được biến nào tác động tích cực hay tiêu cực đến điểm số.

3.3.2. Decision Tree (DT)

Cấu hình chính:

- Sử dụng tiêu chí squared_error để chia nhánh và tham số max_depth để kiểm soát độ sâu của cây.

Giải thích:

- Decision Tree xây dựng dựa trên các split của biến nhằm tối đa hóa thông tin thu được (information gain) hoặc giảm Gini impurity:

$$Gini(D) = 1 - \sum_{i=1}^{C} p_i^2$$

- Thuận tiện trực quan hóa, dễ giải thích các quyết định rủi ro.
- Ưu điểm: * Có khả năng bắt được các mối quan hệ phi tuyến tính phức tạp giữa các biến.

3.3.3. Random Forest (RF)

Cấu hình chính:

- Tập hợp (Ensemble) nhiều cây quyết định độc lập. Các tham số quan trọng gồm n_estimators (số lượng cây) và max_features (số biến chọn ngẫu nhiên cho mỗi cây).
- Giải thích: Thay vì tin vào một cây duy nhất, mô hình lấy trung bình cộng dự báo từ hàng trăm cây khác nhau để đưa ra kết quả cuối cùng.

- Ưu điểm: Độ chính xác rất cao và ổn định hơn nhiều so với một cây đơn lẻ.
- Hạn chế tối đa hiện tượng Overfitting (quá khớp) bằng cách lấy mẫu ngẫu nhiên dữ liệu và biến số.
- Xử lý rất tốt dữ liệu có sự tập trung mật độ cao như dải điểm 95-100 trong bài.

3.4. Đánh giá và so sánh kết quả

3.4.1. Tổng quan kết quả trên Train và Validation

```
========== Comparison of Models on Train & Validation ==========
```

	Model	Dataset	MAE	RMSE	R2
0	Linear Regression	Train	0.063480	0.073460	0.937670
1	Linear Regression	Validation	0.063310	0.072910	0.939210
3	Decision Tree	Train	0.000000	0.000000	1.000000
4	Decision Tree	Validation	0.011660	0.038460	0.983080
6	Random Forest	Train	0.007030	0.012220	0.998280
7	Random Forest	Validation	0.017800	0.031590	0.988590

Hình 3.9 Đánh giá trên tập train/val

Nhận xét:

- Logistic Regression
  + Đơn giản, hiệu suất ở mức khá với R^2 đạt ~0.939 trên tập Validation.
  + Sai số MAE (~0.063) và RMSE (~0.073) rất ổn định giữa tập Train và Validation, cho thấy mô hình không bị hiện tượng quá khớp (overfitting).
  + Đóng vai trò là mô hình nền tảng (benchmark) để so sánh hiệu quả với các thuật toán phức tạp hơn.
- Decision Tree

+ Xảy ra hiện tượng Overfitting nghiêm trọng: R^2 trên tập Train đạt tuyệt đối (1.00) và sai số bằng 0, nhưng hiệu suất giảm rõ rệt trên tập Validation (R^2 còn 0.983).
+ Dù R^2 vẫn ở mức cao, nhưng sự chênh lệch lớn giữa hai tập dữ liệu cho thấy mô hình đang "học thuộc lòng" dữ liệu thay vì học quy luật chung .

- Random Forest
  + Hiệu suất tốt nhất: Đạt R^2 cao nhất trên tập Validation (~0.988) và sai số RMSE thấp nhất (~0.031).
  + Khắc phục hiệu quả lỗi Overfitting của Decision Tree nhờ cơ chế Ensemble (lấy trung bình dự báo từ nhiều cây), giúp mô hình ổn định và bền vững hơn

3.4.2. Đánh giá trên tập Test

```
========== Comparison of Models on Test Set ==========
```

	Model	Dataset	MAE	RMSE	R2
2	Linear Regression	Test	0.063830	0.073780	0.938070
5	Decision Tree	Test	0.012340	0.038120	0.983470
8	Random Forest	Test	0.018540	0.031860	0.988450

Hình 3.10 Đánh giá trên tập test

Nhận xét:

- Linear Regression Hiệu suất ổn định nhưng thấp nhất và đóng vai trò nền tảng, sai số RMSE cao nhất trong 3 mô hình.
- Decision tree kết quả tốt tuy nhiên vẫn tồn tại khoảng cách lớn so với kết quả hoàn hảo, cho thấy dấu hiệu overfitting.
- Random Forest ổn định hiệu suất cao nhất , khả năng tổng quát hóa tốt trên dữ liệu thực tế

KẾT LUẬN

Kết quả đạt được

Qua quá trình thực nghiệm với 3 mô hình học máy (Linear Regression, Decision Tree, Random Forest), kết quả cho thấy:

- Random Forest đạt hiệu suất tốt, giảm overfitting so với Decision Tree đơn lẻ, đồng thời robust với dữ liệu nhiễu và nhiều biến số.
- Decision Tree trực quan, dễ giải thích nhưng có xu hướng overfitting khi cây quá sâu.
- Linear Regresstion đơn giản, tốc độ xử lý nhanh, phù hợp làm mô hình nền tảng (Benchmark) để đánh giá các thuật toán phức tạp hơn.

Hạn chế của đề tài

- Dữ liệu hạn chế: Tập dữ liệu chỉ gồm 14004 bản ghi và 16 đặc trưng, có thể chưa phản ánh đầy đủ sự đa dạng của sinh viên
- Hiện tượng Overfitting: Mô hình Decision Tree chưa được tối ưu sâu về các tham số cắt tỉa (Pruning), dẫn đến việc học thuộc lòng dữ liệu huấn luyện.
- Phạm vi dữ liệu: Đề tài chủ yếu khai thác các biến hành vi học tập, chưa tích hợp được các yếu tố tâm lý xã hội hoặc hoàn cảnh gia đình để có cái nhìn đa chiều hơn về kết quả học tập.

Hướng phát triển của đề tài

- Mở rộng dữ liệu: Thu thập thêm các biến số định tính về tâm lý học đường, hoàn cảnh gia đình và điều kiện kinh tế để tăng tính đa chiều cho mô hình dự báo.
- Bổ sung dữ liệu về các nhóm sinh viên có mức điểm trung bình và thấp để cân bằng lại phân phối của ExamScore, giúp mô hình dự báo nhạy bén hơn ở mọi dải điểm.
- Triển khai các mô hình Boosting mạnh mẽ như XGBoost, LightGBM hoặc CatBoost để so sánh hiệu suất với Random Forest.

- Sử dụng công cụ SHAP hoặc LIME để phân tích sâu mức độ ảnh hưởng của từng biến (ví dụ: StudyHours tác động cụ thể bao nhiêu % đến ExamScore).
- Giúp các nhà quản lý giáo dục hiểu rõ "tại sao" một sinh viên được dự báo đạt điểm cao hay thấp để có biện pháp can thiệp kịp thời.
- Ứng dụng thực tiễn: Phát triển một hệ thống cảnh báo sớm (Early Warning System) tích hợp vào cổng thông tin sinh viên để đưa ra khuyến cáo học tập dựa trên dữ liệu thời gian thực.

TÀI LIỆU THAM KHẢO

[1] Tiệp, V. H. (2018). Machine Learning Cơ bản [Ebook]. GitHub repository. https://github.com/tiepvupsu/ebookMLCB

[2] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830. https://arxiv.org/abs/1201.0490

[3] Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems (2nd ed.). O'Reilly Media.

[4] Brownlee, J. (2020). Machine Learning Mastery with Python: Understand Your Data, Create Accurate Models, and Work Projects End-to-End. Machine Learning Mastery.

[5] Raschka, S., & Mirjalili, V. (2019). Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2 (3rd ed.). Packt Publishing.

[6] TiepVuPSU. (2018). Machine Learning Cơ bản – Mã nguồn minh họa và ví dụ [GitHub repository]. https://github.com/tiepvupsu/ebookMLCB