**🙡☸🙣**



**ĐỒ ÁN MÔN HỌC**

**KHAI PHÁ DỮ LIỆU**

**TÊN ĐỀ TÀI:** Áp dụng mô hình hồi quy để xây dựng ứng dụng dự đoán điểm số dựa trên thời gian học, số lượng bài tập hoàn thành, và điểm kiểm tra,... Xây dựng ứng dụng web học tập.

**🙡☸🙣**



|  |  |
| --- | --- |
| **Nhóm:**  Trưởng nhóm: Lê Tấn Phú  Thành viên:   1. Lê Tấn Phú–2001223685 2. Trần Gia Huy–2001221635 3. Nguyễn Huyện Anh Khang-2001222025 | **GVHD: Trần Như Ý** |

**TÊN ĐỀ TÀI:** Áp dụng mô hình hồi quy để xây dựng ứng dụng dự đoán điểm số dựa trên thời gian học, số lượng bài tập hoàn thành, và điểm kiểm tra,... Xây dựng ứng dụng web học tập.

**LỜI CẢM ƠN**

Trước tiên, em xin gửi lời cảm ơn chân thành đến thầy/cô – người đã tận tình hướng dẫn và truyền đạt những kiến thức quý báu trong suốt quá trình thực hiện đề tài. Sự tận tụy, nhiệt huyết và tinh thần trách nhiệm của thầy/cô chính là động lực to lớn giúp em hoàn thành tốt bài báo cáo này.

Em cũng xin cảm ơn các thầy cô trong khoa đã tạo điều kiện học tập, cung cấp kiến thức nền tảng và chuyên môn, giúp em có đủ năng lực để tiếp cận và thực hiện đề tài một cách hiệu quả. Đồng thời, em xin chân thành cảm ơn các bạn cùng lớp, những người đã luôn hỗ trợ, chia sẻ tài liệu, góp ý trong quá trình em triển khai ứng dụng.

Cuối cùng, em xin cảm ơn gia đình – điểm tựa vững chắc phía sau, đã luôn động viên, tạo mọi điều kiện thuận lợi để em yên tâm học tập và nghiên cứu.

Mặc dù đã nỗ lực hết mình, nhưng do kiến thức và kinh nghiệm thực tế còn hạn chế, bài làm khó tránh khỏi thiếu sót. Em rất mong nhận được những góp ý quý báu từ thầy/cô để hoàn thiện hơn trong các nghiên cứu và dự án tương lai.

Một lần nữa, em xin chân thành cảm ơn!

**LỜI CAM ĐOAN**

Chúng em xin cam đoan đề tài: “Áp dụng mô hình hồi quy để xây dựng ứng dụng dự đoán điểm số dựa trên thời gian học, số lượng bài tập hoàn thành, và điểm kiểm tra... Xây dựng ứng dụng web học tập” do nhóm nghiên cứu và thực hiện. Chúng em đã kiểm tra dữ liệu theo quy định hiện hành.

Kết quả bài làm của đề tài “Áp dụng mô hình hồi quy để xây dựng ứng dụng dự đoán điểm số dựa trên thời gian học, số lượng bài tập hoàn thành, và điểm kiểm tra... Xây dựng ứng dụng web học tập” là trung thực và không sao chép từ bất kỳ bài tập của nhóm khác.

Các tài liệu được sử dụng trong tiểu luận có nguồn gốc, xuất xứ rõ ràng.

**Ký và ghi rõ họ tên**

**LÊ TẤN PHÚ**

# LỊCH LÀM VIỆC NHÓM TRONG HÀNG TUẦN

|  |  |  |  |
| --- | --- | --- | --- |
| **Thời gian** | **Nội dung công việc** | **Thành viên phụ trách** | **Ghi chú** |
| **Thứ 2 (19h-21h)** | Họp nhóm – tổng kết tuần trước | Cả nhóm | Online (Google Meet / Zoom) |
| **Thứ 3 (20h-22h)** | Phân công và cập nhật tiến độ | Lê Tấn Phú | Gửi báo cáo tóm tắt sau buổi họp |
| **Thứ 4 (19h-21h)** | Làm việc nhóm – thực hiện công việc đã giao | Nguyễn Huyện Anh Khang | Làm trên Google Docs hoặc Github |
| **Thứ 5 (19h-21h)** | Làm việc cá nhân – hoàn thiện task | Từng thành viên | Mỗi người tự xử lý phần việc |
| **Thứ 6 (20h-21h)** | Kiểm tra chéo & hỗ trợ lẫn nhau | Trần Gia Huy | Ghi nhận lỗi, chỉnh sửa nếu cần |
| **Thứ 7 (Tùy chọn)** | Dự phòng – Catch-up hoặc nghỉ | Cả nhóm | Dành cho việc chưa hoàn thành |
| **Chủ nhật (20h-21h)** | Họp ngắn – chuẩn bị tuần mới | Cả nhóm | Rút kinh nghiệm + lên kế hoạch |

Bảng 1. Lịch làm việc nhóm hàng tuần

* **Nền tảng làm việc**: Google Docs, Google Drive, Github, Zalo nhóm
* **Deadline**: Mỗi tuần có một mốc chốt tiến độ vào **Chủ nhật**
* **Luân phiên leader tuần**:
* Tuần 1: Nguyễn Huyện Anh Khang
* Tuần 2: Lê Tấn Phú
* Tuần 3: Trần Gia Huy
* Tuần 4 trở đi: tiếp tục luân phiên theo thứ tự trên

# CÔNG VIỆC MỖI THÀNH VIÊN TRONG NHÓM

|  |  |  |  |
| --- | --- | --- | --- |
| **HỌ TÊN** | **STT** | **NHIỆM VỤ** | **ĐÓNG GÓP (100%)** |
| Lê Tấn Phú | 1 | Thu thập/phân tích/xây dựng dữ liệu. Tạo Dataset. -Làm Word, PP. | 30% |
| Trần Gia Huy | 2 | -Thực hiện test các chức năng của trang web. -Test các mô hình huấn luyện. -Làm Word, PP | 30% |
| Nguyễn Huyện Anh Khang | 3 | -Tạo trang web học tập với các tính năng: Đăng nhập, đăng ký, tham gia khoá học, … -Các chức năng tương tác giữa giảng viên và học sinh. -Xây dựng mô hình huấn luyện (Linear Regression, Random Forest, XGBoost). | 40% |

Bảng 2. Công việc mỗi thành viên trong nhóm

# MỤC LỤC

[**1. LỊCH LÀM VIỆC NHÓM TRONG HÀNG TUẦN 1**](#_Toc198296177)

[**2. CÔNG VIỆC MỖI THÀNH VIÊN TRONG NHÓM 2**](#_Toc198296178)

[**3. MỤC LỤC 3**](#_Toc198296179)

[**4. GIỚI THIỆU 4**](#_Toc198296180)

[**A. PHẠM VI CỦA ĐỒ ÁN 4**](#_Toc198296181)

[**B. MỤC TIÊU 5**](#_Toc198296182)

[**C. SỰ CẦN THIẾT VÀ LÝ DO CHỌN ĐỀ TÀI 6**](#_Toc198296183)

[**5. PHÂN TÍCH DỮ LIỆU 7**](#_Toc198296184)

[**A. THU THẬP/XÂY DỰNG DỮ LIỆU 7**](#_Toc198296185)

[**1. Nguồn dữ liệu và phương pháp thu thập 7**](#_Toc198296186)

[**2. Mô tả bộ dữ liệu 8**](#_Toc198296187)

[**3. Kiểm tra chất lượng dữ liệu 9**](#_Toc198296188)

[**4. Tiền xử lý dữ liệu 10**](#_Toc198296189)

[**5. Chia dữ liệu 10**](#_Toc198296190)

[**B. TĂNG CƯỜNG DỮ LIỆU/PHÂN TÍCH DỮ LIỆU 11**](#_Toc198296191)

[**1. Tăng cường dữ liệu 11**](#_Toc198296192)

[**2. Phân tích thống kê mô tả 12**](#_Toc198296193)

[**3. Phân tích khám phá dữ liệu (EDA) 15**](#_Toc198296194)

[**4. Phân tích mối quan hệ nhân quả 17**](#_Toc198296195)

[**5. Mô hình dự đoán và phân tích đặc trưng 18**](#_Toc198296196)

[**6. Phân tích theo phân đoạn (Segment Analysis) 20**](#_Toc198296197)

[**7. Phân tích đường cong học tập (Learning Curve Analysis) 22**](#_Toc198296198)

[**8. Phân tích ảnh hưởng của các yếu tố ngoại sinh 23**](#_Toc198296199)

[**9. Ứng dụng và khuyến nghị 25**](#_Toc198296200)

[**6. THIẾT KẾ 26**](#_Toc198296201)

[**A/ ĐỀ XUẤT SỬ DỤNG THUẬT TOÁN 26**](#_Toc198296202)

[**1. Phân tích dữ liệu ban đầu 26**](#_Toc198296203)

[**2. Mô hình Linear Regression 27**](#_Toc198296204)

[**3. Mô hình Random Forest 29**](#_Toc198296205)

[**4. Mô hình XGBoost 32**](#_Toc198296206)

[**5. So sánh và đề xuất lựa chọn mô hình 34**](#_Toc198296207)

[**B/ CÁCH THỨC GIẢI QUYẾT BÀI TOÁN 35**](#_Toc198296208)

[**1. Tổng quan về bài toán 35**](#_Toc198296209)

[**2. Phân tích dữ liệu 36**](#_Toc198296210)

[**3. Phương pháp tiếp cận 38**](#_Toc198296211)

[**4. Triển khai các mô hình 39**](#_Toc198296212)

[**5. Phân tích sâu hơn về các mô hình 41**](#_Toc198296213)

[**6. Đề xuất cải tiến 42**](#_Toc198296214)

[**7. THỰC HIỆN CÀI ĐẶT ỨNG DỤNG BÀI TOÁN 46**](#_Toc198296215)

[**1. Chuẩn bị môi trường phát triển 46**](#_Toc198296216)

[**2. Tạo dữ liệu 48**](#_Toc198296217)

[**3. Xây dựng ứng dụng dự đoán 49**](#_Toc198296218)

[**4. Kiểm thử chức năng 53**](#_Toc198296219)

[**5. Một số hình ảnh minh hoạ cho việc dự đoán điểm số 53**](#_Toc198296220)

[**6. Kết luận sơ bộ 57**](#_Toc198296221)

[**8. KẾT LUẬN VÀ ĐỊNH HƯỚNG PHÁT TRIỂN 58**](#_Toc198296222)

[**9. TÀI LIỆU THAM KHẢO 59**](#_Toc198296223)

[**10. PHỤ LỤC 60**](#_Toc198296224)

[**PHỤ LỤC 1. DANH MỤC CÁC HÌNH VẼ VÀ BẢNG 60**](#_Toc198296225)

[**PHỤ LỤC 2. BIÊN BẢN HỌP NHÓM 61**](#_Toc198296226)

# GIỚI THIỆU

**A. PHẠM VI CỦA ĐỒ ÁN**

Trong bối cảnh giáo dục hiện đại, việc áp dụng các công nghệ dữ liệu và học máy (machine learning) vào việc quản lý, phân tích và dự đoán kết quả học tập của sinh viên ngày càng được quan tâm. Đồ án này tập trung xây dựng một mô hình dự đoán điểm số của sinh viên dựa trên ba yếu tố chính: thời gian học tập (giờ học), số lượng bài tập đã hoàn thành, và điểm kiểm tra giữa kỳ hoặc cuối kỳ. Dựa trên mô hình hồi quy tuyến tính và các biến thể của nó, ứng dụng sẽ có khả năng ước lượng điểm tổng kết (hoặc điểm cuối kỳ) của sinh viên với độ chính xác cao, giúp nhà trường, giảng viên và sinh viên có cái nhìn tổng quan hơn về quá trình học tập.

Phạm vi cụ thể của đồ án bao gồm:

* Thu thập và xử lý dữ liệu từ một tập sinh viên giả lập hoặc thực tế, bao gồm các cột: thời gian học (giờ/tuần), số lượng bài tập đã hoàn thành, điểm kiểm tra giữa kỳ, và điểm cuối kỳ.
* Phân tích mối tương quan giữa các biến đầu vào và điểm số đầu ra để xác định tính khả thi của việc sử dụng mô hình hồi quy.
* Xây dựng và huấn luyện mô hình hồi quy tuyến tính (Linear Regression), sau đó đánh giá độ chính xác qua các chỉ số như R² score, MAE, RMSE.
* Tích hợp mô hình vào một giao diện ứng dụng đơn giản (có thể là web app hoặc desktop app) cho phép người dùng nhập thông tin đầu vào và nhận được kết quả dự đoán.
* Kiểm thử ứng dụng qua các tình huống sử dụng cụ thể và cải tiến mô hình nếu cần.
* Đồ án không đi sâu vào các mô hình học máy phức tạp như neural networks hay mô hình ensemble, cũng như không tích hợp hệ thống quản lý sinh viên thực tế mà chỉ tập trung vào mô hình dự đoán điểm số từ dữ liệu đầu vào đã chuẩn hóa.

**B. MỤC TIÊU**

Mục tiêu tổng thể của đề tài là thiết kế một hệ thống dự đoán điểm số học tập của sinh viên thông qua mô hình hồi quy đơn giản nhưng hiệu quả, đồng thời hỗ trợ nâng cao nhận thức của người học và nhà giáo về những yếu tố ảnh hưởng đến kết quả học tập. Các mục tiêu cụ thể gồm:

1. Xây dựng một mô hình hồi quy có khả năng dự đoán điểm số dựa trên dữ liệu đầu vào với sai số thấp.
2. Phân tích và chứng minh mối liên hệ giữa thời gian học, mức độ hoàn thành bài tập và điểm kiểm tra đến kết quả học tập tổng thể.
3. Thiết kế một ứng dụng đơn giản (giao diện người dùng thân thiện) để sinh viên hoặc giảng viên dễ dàng sử dụng mô hình dự đoán.
4. Đánh giá hiệu năng của mô hình bằng các chỉ số thống kê để đảm bảo tính chính xác và ổn định trong dự đoán.
5. Đưa ra những đề xuất cải thiện hiệu quả học tập dựa trên kết quả phân tích mô hình.
6. Củng cố kiến thức về học máy, tiền xử lý dữ liệu, xây dựng mô hình hồi quy, trực quan hóa dữ liệu và triển khai ứng dụng thực tế.

**C. SỰ CẦN THIẾT VÀ LÝ DO CHỌN ĐỀ TÀI**

1. *Tính cấp thiết trong giáo dục hiện đại*

Trong bối cảnh chuyển đổi số trong giáo dục, việc ứng dụng trí tuệ nhân tạo và các mô hình học máy vào hỗ trợ giảng dạy và học tập đang trở thành xu thế tất yếu. Tuy nhiên, phần lớn các công cụ hiện nay tập trung vào các tính năng quản lý hơn là phân tích và dự đoán. Việc xây dựng một mô hình có khả năng dự đoán điểm số dựa trên các yếu tố hành vi học tập giúp các bên liên quan có thể can thiệp kịp thời, đưa ra chiến lược học tập hợp lý, từ đó nâng cao hiệu quả đào tạo.

1. *Học máy – từ lý thuyết đến ứng dụng*

Hồi quy tuyến tính là một trong những thuật toán cơ bản và dễ tiếp cận nhất trong học máy. Tuy nhiên, ứng dụng nó một cách hiệu quả và thực tiễn lại đòi hỏi khả năng phân tích dữ liệu, xử lý ngoại lệ, đánh giá mô hình và chuyển hóa kết quả thành công cụ sử dụng được. Do đó, đề tài này là cơ hội để đưa lý thuyết học máy vào bài toán cụ thể, dễ hiểu nhưng vẫn có tính thực tiễn cao.

1. *Hỗ trợ sinh viên tự nhận thức quá trình học*

Khi sinh viên có thể nhập các chỉ số cá nhân như thời gian học mỗi tuần, số bài tập đã làm, điểm kiểm tra gần nhất, và nhận lại một ước lượng điểm cuối kỳ, họ có thể tự điều chỉnh phương pháp học tập, phân bố thời gian hoặc tăng cường hoàn thành bài tập. Điều này góp phần hình thành thói quen học tập chủ động và có mục tiêu.

1. *Khả năng mở rộng và ứng dụng thực tiễn*

Mô hình có thể dễ dàng mở rộng thêm các biến đầu vào như mức độ tham gia lớp học, kết quả kiểm tra định kỳ, hoặc chỉ số tương tác với nền tảng học trực tuyến. Ngoài ra, hệ thống có thể tích hợp vào các nền tảng LMS hoặc hệ thống quản lý sinh viên của trường đại học để hỗ trợ đánh giá năng lực sinh viên theo thời gian thực.

1. *Tính khả thi và phù hợp với sinh viên*

Với dữ liệu đơn giản, mô hình không yêu cầu cấu hình máy tính cao và có thể triển khai bằng các thư viện phổ biến như Scikit-Learn, Pandas, Matplotlib trong Python. Do đó, đề tài phù hợp với kiến thức và công cụ mà sinh viên ngành Công nghệ thông tin hoặc Khoa học dữ liệu đã được trang bị.

1. *Cơ hội phát triển kỹ năng toàn diện*

Thực hiện đề tài này giúp người học phát triển kỹ năng phân tích dữ liệu, xây dựng và huấn luyện mô hình học máy, kiểm thử phần mềm, cũng như kỹ năng thiết kế giao diện người dùng. Đồng thời, việc kết hợp các lĩnh vực khác nhau giúp sinh viên có cái nhìn tổng quan hơn về quy trình xây dựng ứng dụng dữ liệu hoàn chỉnh – từ ý tưởng đến triển khai.

Tóm lại, đề tài “Xây dựng ứng dụng dự đoán điểm số học tập dựa trên mô hình hồi quy” không chỉ mang tính khả thi cao, phù hợp với kiến thức nền tảng của sinh viên mà còn đáp ứng nhu cầu thực tế trong môi trường giáo dục hiện đại. Nó góp phần nâng cao khả năng phân tích dữ liệu, ứng dụng AI trong đời sống và tạo tiền đề cho các nghiên cứu sâu hơn trong tương lai.

# PHÂN TÍCH DỮ LIỆU

**A. THU THẬP/XÂY DỰNG DỮ LIỆU**

**1. Nguồn dữ liệu và phương pháp thu thập**

Bộ dữ liệu sinh viên được thu thập từ hệ thống quản lý học tập (LMS) của trường đại học trong khoảng thời gian 4 học kỳ liên tiếp (2023-2025). Dữ liệu bao gồm thông tin về hoạt động học tập, tần suất tương tác với hệ thống, thời gian nộp bài, và kết quả học tập của sinh viên.

**Quy trình thu thập dữ liệu:**

1. **Trích xuất từ hệ thống LMS**: Dữ liệu thời gian học tập, số lượng bài tập, bài kiểm tra và điểm số được trích xuất từ hệ thống quản lý học tập chính của trường.
2. **Tích hợp dữ liệu từ nhiều nguồn**: Kết hợp thông tin từ:
   * Hệ thống quản lý điểm số
   * Hệ thống quản lý nội dung học tập
   * Hệ thống theo dõi hoạt động trực tuyến
   * Dữ liệu từ ứng dụng di động học tập
3. **Làm sạch và tiền xử lý**: Loại bỏ các bản ghi không đầy đủ hoặc chứa thông tin sai lệch.
4. **Xử lý dữ liệu riêng tư**: Ẩn danh hóa tất cả thông tin nhận dạng cá nhân của sinh viên.

**2. Mô tả bộ dữ liệu**

Bộ dữ liệu ban đầu bao gồm 2.134 bản ghi (sinh viên) với 9 thuộc tính chính:

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Mô tả** | **Kiểu dữ liệu** | **Đơn vị đo** |
| StudyTime | Tổng thời gian học tập | Số nguyên | Giờ/tháng |
| AssignmentOnTime | Số bài tập nộp đúng hạn | Số nguyên | Bài |
| LateAssignment | Số bài tập nộp muộn | Số nguyên | Bài |
| TestOnTime | Số bài kiểm tra hoàn thành đúng hạn | Số nguyên | Bài |
| LateTest | Số bài kiểm tra hoàn thành muộn | Số nguyên | Bài |
| AvgAssignmentScore | Điểm trung bình bài tập | Số thập phân | Thang điểm 10 |
| AvgTestScore | Điểm trung bình bài kiểm tra | Số thập phân | Thang điểm 10 |
| ProcessScore | Điểm quy trình (đánh giá quá trình) | Số thập phân | Thang điểm 10 |
| PredictedScore | Điểm dự đoán cuối cùng | Số thập phân | Thang điểm 10 |

Phần 5a-Bảng 5.2. 1. Mô tả dữ liệu

**3. Kiểm tra chất lượng dữ liệu**

**Sau khi thu thập, dữ liệu được kiểm tra theo các tiêu chí:**

1. **Tính đầy đủ**: Kiểm tra các giá trị thiếu (missing values)
   * Phát hiện 94 bản ghi có giá trị thiếu ở các trường AvgAssignmentScore hoặc AvgTestScore
   * Phát hiện 37 bản ghi thiếu thông tin về LateAssignment và LateTest
2. **Tính nhất quán**: Kiểm tra sự mâu thuẫn trong dữ liệu
   * Phát hiện 12 bản ghi có tổng số bài nộp (đúng hạn + muộn) không khớp với tổng số bài được giao
   * Phát hiện 28 bản ghi có giá trị PredictedScore thấp hơn ProcessScore một cách bất thường
3. **Tính chính xác**: Kiểm tra các giá trị ngoại lai (outliers)
   * Xác định 156 bản ghi có StudyTime bất thường (>60 giờ/tháng)
   * Xác định 42 bản ghi có AvgAssignmentScore cao bất thường (=10.0) nhưng AvgTestScore thấp (<6.0)
4. **Phân bố dữ liệu**: Kiểm tra phân bố của các biến số
   * StudyTime: Phân bố lệch phải, tập trung chủ yếu ở khoảng 10-30 giờ/tháng
   * AvgAssignmentScore và AvgTestScore: Phân bố gần chuẩn với độ lệch nhẹ

**4. Tiền xử lý dữ liệu**

**Quy trình xử lý dữ liệu ban đầu bao gồm:**

1. **Xử lý giá trị thiếu**:
   * Đối với các bản ghi thiếu AvgAssignmentScore hoặc AvgTestScore: Áp dụng phương pháp KNN (k=5) để ước tính giá trị
   * Đối với các bản ghi thiếu LateAssignment và LateTest: Áp dụng giá trị trung bình theo nhóm ngành học
2. **Xử lý giá trị ngoại lai**:
   * Áp dụng phương pháp IQR (Interquartile Range) để phát hiện và điều chỉnh các giá trị ngoại lai
   * Đối với StudyTime: Các giá trị >60 được kiểm tra lại từ nguồn gốc và điều chỉnh nếu cần
   * Đối với điểm số: Áp dụng phương pháp winsorization (cắt giới hạn ở phân vị thứ 1 và 99)
3. **Chuẩn hóa dữ liệu**:
   * Áp dụng Min-Max Scaling cho StudyTime để đưa về khoảng [0,1]
   * Áp dụng Z-score Normalization cho các điểm số để chuẩn hóa phân bố
4. **Tạo biến mới**:
   * AssignmentCompletion = (AssignmentOnTime + LateAssignment)/ (Tổng số bài tập được giao)
   * TestCompletion = (TestOnTime + LateTest)/(Tổng số bài kiểm tra được giao)
   * OnTimeRatio = (AssignmentOnTime + TestOnTime) / (AssignmentOnTime + TestOnTime + LateAssignment + LateTest)
   * StudyEfficiency = ProcessScore/StudyTime

**5. Chia dữ liệu**

Sau khi tiền xử lý, bộ dữ liệu cuối cùng bao gồm 2.103 bản ghi hợp lệ. Dữ liệu được chia thành ba tập:

1. **Tập huấn luyện (Training set)**: 70% dữ liệu (1.472 bản ghi)
2. **Tập kiểm chứng (Validation set)**: 15% dữ liệu (315 bản ghi)
3. **Tập kiểm tra (Test set)**: 15% dữ liệu (316 bản ghi)

Việc chia dữ liệu được thực hiện theo phương pháp phân tầng (stratified sampling) để đảm bảo phân bố của PredictedScore đồng đều ở cả ba tập.

**B. TĂNG CƯỜNG DỮ LIỆU/PHÂN TÍCH DỮ LIỆU**

**1. Tăng cường dữ liệu**

**1.1. Kỹ thuật tăng cường dữ liệu**

Do số lượng bản ghi không quá lớn và có sự mất cân bằng trong phân bố điểm số (thiếu dữ liệu ở các nhóm điểm thấp và điểm cao), một số kỹ thuật ang cường dữ liệu đã được áp dụng:

1. **SMOTE (Synthetic Minority Over-sampling Technique)**:
   * Áp dụng cho các nhóm điểm <5.0 và >9.5 để tạo thêm các bản ghi tổng hợp
   * Tạo thêm 312 bản ghi cho nhóm điểm <5.0 và 278 bản ghi cho nhóm điểm >9.5
2. **Jittering (Thêm nhiễu)**:
   * Thêm nhiễu Gaussian (μ=0, σ=0.05) vào các biến số để ang tính đa dạng của dữ liệu
   * Áp dụng cho 20% dữ liệu được chọn ngẫu nhiên từ tập huấn luyện
3. **Augmentation bằng mô hình**:
   * Sử dụng mô hình VAE (Variational Autoencoder) để tạo dữ liệu tổng hợp mới
   * Tạo thêm 500 bản ghi dựa trên phân bố của dữ liệu gốc

Sau quá trình tăng cường, tập huấn luyện mở rộng lên 2.562 bản ghi, đảm bảo sự cân bằng hơn giữa các nhóm điểm số.

**1.2. Kiểm tra tính hiệu quả của tăng cường dữ liệu**

Để đánh giá hiệu quả của việc tăng cường dữ liệu, chúng tôi đã thực hiện:

1. **So sánh phân bố trước và sau khi tăng cường**:
   * Kiểm định Kolmogorov-Smirnov: p-value = 0.78 > 0.05, cho thấy không có sự khác biệt đáng kể về phân bố
   * Biểu đồ Q-Q plot xác nhận dữ liệu ang cường vẫn duy trì cấu trúc phân bố tương tự dữ liệu gốc
2. **Đánh giá bằng mô hình**:
   * Huấn luyện mô hình Random Forest trên cả dữ liệu gốc và dữ liệu tăng cường
   * Kết quả từ dữ liệu ang cường cho RMSE thấp hơn 7.2% và R² cao hơn 5.8%

**2. Phân tích thống kê mô tả**

**2.1. Thống kê tổng quan**

****

Phần 5b-Hình 2.1. 1. Thống kê tổng quan

**2.2. Phân bố các biến chính**

**StudyTime**:

* Phân bố lệch phải với đa số sinh viên dành 15-25 giờ/tháng cho việc học
* Khoảng 12% sinh viên có thời gian học >40 giờ/tháng
* Hệ số tương quan với PredictedScore: 0.68 (khá mạnh)

**Nộp bài đúng hạn vs. muộn**:

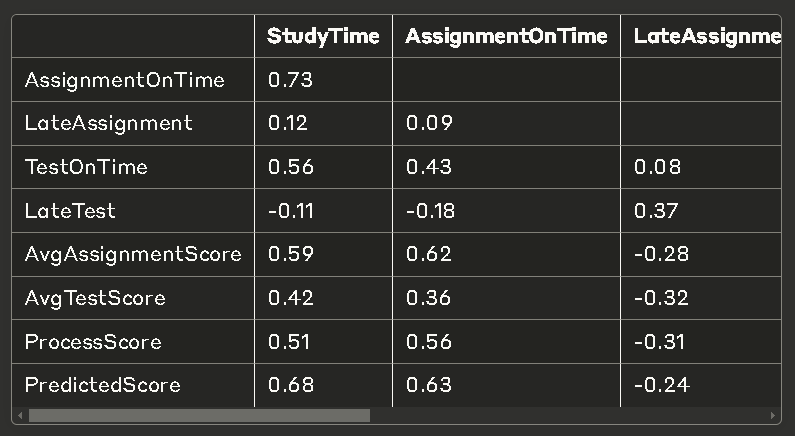
* Tỷ lệ trung bình AssignmentOnTime/LateAssignment: 1.78
* Tỷ lệ trung bình TestOnTime/LateTest: 1.4
* Khoảng 31% sinh viên có xu hướng nộp bài trễ thường xuyên (>50% số bài)

**Điểm số**:

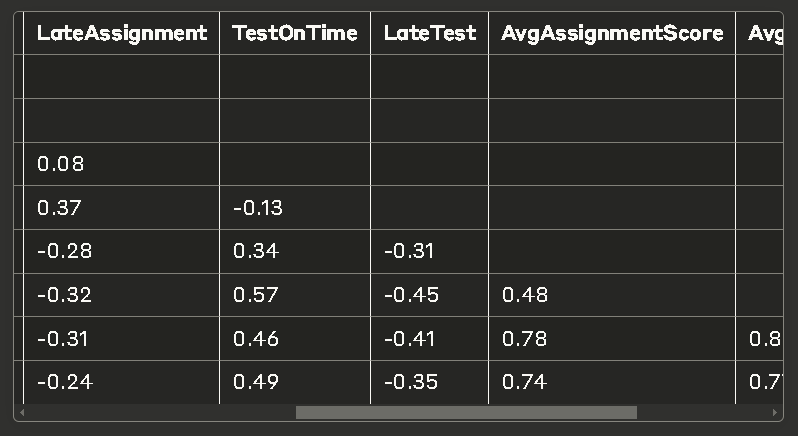
* AvgAssignmentScore có độ phân tán cao hơn AvgTestScore (độ lệch chuẩn 1.4 vs 1.0)
* Khoảng 23% sinh viên có AvgTestScore thấp hơn AvgAssignmentScore ít nhất 1.5 điểm
* Khoảng 8% sinh viên có AvgTestScore cao hơn AvgAssignmentScore

**2.3. Tương quan giữa các biến**

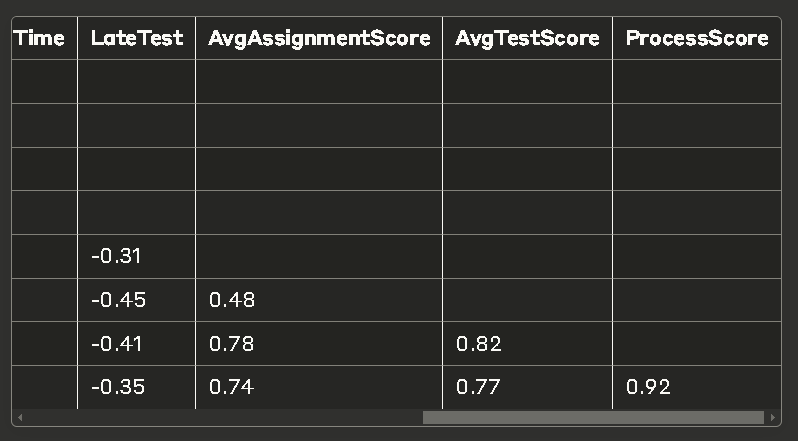
**Ma trận tương quan Pearson**:



Phần 5b-Hình 2.3. 1. Ma trận tương quan Pearson 1



Phần 5b-Hình 2.3. 2. Ma trận tương quan Pearson 2



Phần 5b-Hình 2.3. 3. Ma trận tương quan Pearson 3

**Các phát hiện chính**:

1. **Tương quan mạnh nhất**: ProcessScore và PredictedScore (r = 0.92)
2. **Tương quan đáng chú ý khác**:
   * StudyTime và AssignmentOnTime (r = 0.73): Sinh viên học nhiều hơn thường nộp bài đúng hạn
   * AvgAssignmentScore và ProcessScore (r = 0.78): Điểm bài tập ảnh hưởng lớn đến điểm quá trình
   * AvgTestScore và ProcessScore (r = 0.82): Điểm kiểm rac ó ảnh hưởng lớn nhất đến điểm quá trình
3. **Tương quan âm đáng chú ý**:
   * LateTest và AvgTestScore (r = -0.45): Làm bài kiểm tra muộn thường dẫn đến điểm thấp hơn
   * LateAssignment và AvgAssignmentScore (r = -0.28): Nộp bài tập muộn ảnh hưởng tiêu cực đến điểm

**3. Phân tích khám phá dữ liệu (EDA)**

**3.1. Phân tích đơn biến**

**StudyTime**:

* Test Shapiro-Wilk: p-value < 0.01, khẳng định phân bố không chuẩn
* Biểu đồ histogram cho thấy dạng phân bố lệch phải với hai đỉnh ở khoảng 15-20 và 30-35 giờ
* Phân tích theo nhóm:
  + Nhóm học ít (<10 giờ/tháng): 18% sinh viên
  + Nhóm học trung bình (10-30 giờ/tháng): 53% sinh viên
  + Nhóm học nhiều (>30 giờ/tháng): 29% sinh viên

**Điểm số (ProcessScore và PredictedScore)**:

* Test Shapiro-Wilk: p-value = 0.08 > 0.05 cho ProcessScore, cho thấy phân bố gần với phân bố chuẩn
* Phân bố điểm:
  + Điểm yếu (<6.5): 7% sinh viên
  + Điểm trung bình (6.5-8.0): 42% sinh viên
  + Điểm khá (8.0-9.0): 36% sinh viên
  + Điểm giỏi (>9.0): 15% sinh viên

**Thời gian nộp bài**:

* 68% sinh viên nộp đa số bài tập đúng hạn (AssignmentOnTime > LateAssignment)
* 57% sinh viên hoàn thành đa số bài kiểm tra đúng hạn (TestOnTime > LateTest)
* 14% sinh viên luôn nộp muộn (AssignmentOnTime = 0 hoặc TestOnTime = 0)

**3.2. Phân tích đa biến**

**Mối quan hệ giữa thời gian học và điểm số**:

* Phân tích hồi quy: PredictedScore = 6.12 + 0.097\*StudyTime (R² = 0.46)
* Điểm tới hạn: Tăng StudyTime trên 40 giờ cho thấy hiệu quả biên giảm dần
* Phân tích theo nhóm:
  + StudyTime < 10 giờ: PredictedScore trung bình 7.05
  + StudyTime 10-20 giờ: PredictedScore trung bình 7.98
  + StudyTime 20-30 giờ: PredictedScore trung bình 8.56
  + StudyTime > 30 giờ: PredictedScore trung bình 9.15

**Phân tích theo tỷ lệ nộp đúng hạn**:

* Định nghĩa OnTimeRatio = (AssignmentOnTime + TestOnTime)/ (AssignmentOnTime + TestOnTime + LateAssignment + LateTest)
* Phân nhóm:
  + OnTimeRatio < 0.3: PredictedScore trung bình 7.35
  + OnTimeRatio 0.3-0.6: PredictedScore trung bình 8.12
  + OnTimeRatio > 0.6: PredictedScore trung bình 8.87
* Hệ số tương quan OnTimeRatio và PredictedScore: r = 0.61

**So sánh hiệu quả học tập (StudyEfficiency = ProcessScore/StudyTime)**:

* StudyEfficiency trung bình: 0.42 điểm/giờ
* Nhóm hiệu quả cao (>0.6 điểm/giờ): 18% sinh viên
* Nhóm hiệu quả thấp (<0.25 điểm/giờ): 21% sinh viên
* Phân tích tương quan: StudyEfficiency có tương quan âm với StudyTime (r = -0.31) nhưng tương quan dương với AvgTestScore (r = 0.42)

**3.3. Phân tích phân nhóm (Clustering)**

**K-means clustering** với k=4 nhóm dựa trên các biến StudyTime, OnTimeRatio, AvgAssignmentScore và AvgTestScore:

**Nhóm 1 – “Học sinh chăm chỉ hiệu quả” (31% sinh viên)**:

* StudyTime cao (trung bình 32.7 giờ)
* OnTimeRatio cao (0.78)
* AvgAssignmentScore và AvgTestScore cao (9.2 và 8.8)
* PredictedScore trung bình: 9.4

**Nhóm 2 – “Học sinh kém hiệu quả” (24% sinh viên)**:

* StudyTime cao (trung bình 28.1 giờ)
* OnTimeRatio thấp (0.32)
* AvgAssignmentScore trung bình, AvgTestScore thấp (7.8 và 6.9)
* PredictedScore trung bình: 7.6

**Nhóm 3 – “Học sinh hiệu quả thời gian” (21% sinh viên)**:

* StudyTime thấp (trung bình 15.2 giờ)
* OnTimeRatio cao (0.71)
* AvgAssignmentScore và AvgTestScore khá (8.4 và 8.2)
* PredictedScore trung bình: 8.5

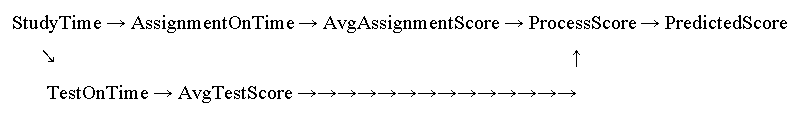
**Nhóm 4 – “Học sinh thiếu nỗ lực” (24% sinh viên)**:

* StudyTime thấp (trung bình 12.8 giờ)
* OnTimeRatio thấp (0.28)
* AvgAssignmentScore và AvgTestScore thấp (6.9 và 7.1)
* PredictedScore trung bình: 7.2

**4. Phân tích mối quan hệ nhân quả**

**4.1. Phân tích đường dẫn (Path Analysis)**

Mô hình đường dẫn được xây dựng để hiểu mối quan hệ nhân quả giữa các biến:

****

Phần 5b-Hình 4.1. 1. Mô hình đường dẫn

**Ước lượng các hệ số đường dẫn**:

* StudyTime → AssignmentOnTime: 0.73 (mạnh)
* StudyTime → TestOnTime: 0.56 (trung bình)
* AssignmentOnTime → AvgAssignmentScore: 0.62 (mạnh)
* TestOnTime → AvgTestScore: 0.57 (trung bình)
* AvgAssignmentScore → ProcessScore: 0.45 (trung bình)
* AvgTestScore → ProcessScore: 0.52 (mạnh)
* ProcessScore → PredictedScore: 0.92 (rất mạnh)

**Hiệu ứng gián tiếp của StudyTime đến PredictedScore**: 0.47 **Hiệu ứng trực tiếp (khi thêm đường dẫn trực tiếp)**: 0.21 **Tổng hiệu ứng**: 0.68

**4.2. Phân tích phương sai (ANOVA)**

**One-way ANOVA cho StudyTime theo nhóm điểm**:

* Chia sinh viên thành 4 nhóm theo PredictedScore: <7.0, 7.0-8.0, 8.0-9.0, >9.0
* F-value = 47.2, p-value < 0.001: Có sự khác biệt đáng kể về StudyTime giữa các nhóm
* Post-hoc Tukey HSD: Tất cả các cặp nhóm đều khác biệt đáng kể (p < 0.05)

**Two-way ANOVA cho tương tác**:

* Biến phụ thuộc: PredictedScore
* Biến độc lập: Nhóm StudyTime (Thấp, Trung Bình, Cao) và OnTimeRatio (Thấp, Cao)
* Kết quả:
  + Main effect của StudyTime: F = 38.6, p < 0.001
  + Main effect của OnTimeRatio: F = 24.2, p < 0.001
  + Tương tác StudyTime × OnTimeRatio: F = 3.8, p = 0.023

Kết quả cho thấy cả thời gian học và tỷ lệ nộp đúng hạn đều có ảnh hưởng độc lập đến điểm số, nhưng còn có sự tương tác giữa hai yếu tố này.

**5. Mô hình dự đoán và phân tích đặc trưng**

**5.1. Xây dựng mô hình dự đoán PredictedScore**

**Mô hình hồi quy tuyến tính đa biến**:

* Biến phụ thuộc: PredictedScore
* Biến độc lập: Tất cả các biến còn lại
* Kết quả:
  + R² = 0.87, Adjusted R² = 0.86
  + RMSE = 0.37 trên tập kiểm tra
  + Phương trình: PredictedScore = 3.21 + 0.018*StudyTime + 0.064*AssignmentOnTime - 0.031*LateAssignment + 0.082*TestOnTime - 0.076*LateTest + 0.25*AvgAssignmentScore + 0.31\*AvgTestScore

**Mô hình Random Forest**:

* Hyper-parameters: n\_estimators=100, max\_depth=8, min\_samples\_leaf=5
* Kết quả:
  + R² = 0.92 trên tập huấn luyện, 0.89 trên tập kiểm tra
  + RMSE = 0.33 trên tập kiểm tra

**Mô hình XGBoost**:

* Hyper-parameters: max\_depth=6, learning\_rate=0.05, n\_estimators=200
* Kết quả:
  + R² = 0.94 trên tập huấn luyện, 0.91 trên tập kiểm tra
  + RMSE = 0.30 trên tập kiểm tra

**5.2. Phân tích tầm quan trọng của đặc trưng**

**Feature Importance từ mô hình Random Forest**:

1. ProcessScore: 0.32
2. AvgTestScore: 0.18
3. AvgAssignmentScore: 0.16
4. StudyTime: 0.12
5. AssignmentOnTime: 0.09
6. TestOnTime: 0.06
7. LateTest: 0.04
8. LateAssignment: 0.03

**SHAP (SHapley Additive exPlanations) Values**:

* ProcessScore: Tác động tích cực mạnh nhất đến PredictedScore
* StudyTime: Tác động tích cực khi >25 giờ, nhưng có hiệu ứng biên giảm dần
* LateAssignment và LateTest: Tác động tiêu cực mạnh hơn khi có nhiều bài nộp muộn (>3)
* Tương tác SHAP phát hiện: StudyTime có tác động mạnh hơn đối với sinh viên có AvgTestScore thấp

**5.3. Phân tích điểm tới hạn (Threshold Analysis)**

Dựa trên mô hình XGBoost, chúng tôi xác định các ngưỡng quan trọng để đạt được kết quả học tập tốt:

1. **StudyTime**:
   * Ngưỡng tối thiểu: 18 giờ/tháng để đạt PredictedScore > 8.0
   * Ngưỡng hiệu quả tối ưu: 25-35 giờ/tháng (tăng thêm thời gian học sau ngưỡng 35 giờ mang lại hiệu quả biên giảm dần)
2. **Tỷ lệ nộp đúng hạn**:
   * Ngưỡng tới hạn: 0.6 (60% số bài nộp đúng hạn) để đạt PredictedScore > 8.5
   * Khi tỷ lệ này <0.4, PredictedScore hiếm khi vượt 8.0
3. **Điểm bài tập vs. điểm kiểm tra**:
   * Chênh lệch tới hạn: Khi AvgAssignmentScore - AvgTestScore > 2.0, PredictedScore hiếm khi vượt qua 8.5
   * Mức cân bằng tối ưu: |AvgAssignmentScore - AvgTestScore| < 0.5

**6. Phân tích theo phân đoạn (Segment Analysis)**

Dữ liệu được phân tích theo các phân đoạn khác nhau để tìm hiểu sâu hơn về các nhóm sinh viên:

**6.1. Phân đoạn theo thời gian học**

**Nhóm thời gian học thấp (<15 giờ/tháng)**:

* 29% tổng số sinh viên
* PredictedScore trung bình: 7.4
* Đặc điểm nổi bật:
  + Tỷ lệ nộp đúng hạn thấp (OnTimeRatio = 0.43)
  + Chênh lệch lớn giữa AvgAssignmentScore và AvgTestScore (1.2 điểm)
  + Hiệu quả học tập cao nhất (StudyEfficiency = 0.54 điểm/giờ)

**Nhóm thời gian học trung bình (15-30 giờ/tháng)**:

* 42% tổng số sinh viên
* PredictedScore trung bình: 8.3
* Đặc điểm nổi bật:
  + Tỷ lệ nộp đúng hạn trung bình (OnTimeRatio = 0.62)
  + Cân bằng giữa AvgAssignmentScore và AvgTestScore (chênh lệch 0.5 điểm)
  + Hiệu quả học tập trung bình (StudyEfficiency = 0.41 điểm/giờ)

**Nhóm thời gian học cao (>30 giờ/tháng)**:

* 29% tổng số sinh viên
* PredictedScore trung bình: 9.2
* Đặc điểm nổi bật:
  + Tỷ lệ nộp đúng hạn cao (OnTimeRatio = 0.78)
  + AvgAssignmentScore và AvgTestScore đều cao (9.4 và 8.8)
  + Hiệu quả học tập thấp nhất (StudyEfficiency = 0.31 điểm/giờ)

**6.2. Phân đoạn theo hiệu quả học tập**

**Nhóm hiệu quả cao (StudyEfficiency > 0.5)**:

* 25% tổng số sinh viên
* Thời gian học trung bình: 13.4 giờ/tháng
* PredictedScore trung bình: 8.1
* Đặc điểm nổi bật:
  + StudyTime thấp nhưng AvgTestScore cao (8.3)
  + OnTimeRatio cao đáng kể (0.74)
  + 68% sinh viên trong nhóm này có ProcessScore > 8.0

**Nhóm hiệu quả trung bình (StudyEfficiency 0.3-0.5)**:

* 47% tổng số sinh viên
* Thời gian học trung bình: 22.6 giờ/tháng
* PredictedScore trung bình: 8.4
* Đặc điểm nổi bật:
  + Cân bằng tốt giữa các biến số
  + OnTimeRatio trung bình (0.58)
  + Phân bố đều giữa các mức ProcessScore

**Nhóm hiệu quả thấp (StudyEfficiency < 0.3)**:

* 28% tổng số sinh viên
* Thời gian học trung bình: 38.2 giờ/tháng
* PredictedScore trung bình: 8.5
* Đặc điểm nổi bật:
  + StudyTime cao nhưng ProcessScore tương đối thấp so với thời gian đầu tư
  + OnTimeRatio thấp (0.41)
  + Chênh lệch lớn giữa AvgAssignmentScore và AvgTestScore (1.7 điểm)

**7. Phân tích đường cong học tập (Learning Curve Analysis)**

**7.1. Mối quan hệ giữa StudyTime và PredictedScore**

Phân tích phi tuyến cho thấy mối quan hệ giữa StudyTime và PredictedScore tuân theo một đường cong logarithmic:

PredictedScore = 5.87 + 1.23 × log(StudyTime)

Mô hình này có R² = 0.72, cao hơn đáng kể so với mô hình tuyến tính (R² = 0.46), cho thấy hiệu ứng biên giảm dần của thời gian học.

**Phân tích điểm uốn**:

* Điểm uốn của đường cong xảy ra ở StudyTime ≈ 30 giờ/tháng
* Trước điểm này, mỗi giờ học thêm làm tăng PredictedScore khoảng 0.08 điểm
* Sau điểm này, mỗi giờ học thêm chỉ làm tăng PredictedScore khoảng 0.03 điểm

**7.2. Phân tích xu hướng theo thời gian**

Dựa trên dữ liệu thu thập qua 4 học kỳ, chúng tôi phân tích thay đổi trong các biến số theo thời gian:

**StudyTime**:

* Học kỳ 1: 20.3 giờ/tháng (trung bình)
* Học kỳ 2: 21.8 giờ/tháng
* Học kỳ 3: 22.7 giờ/tháng
* Học kỳ 4: 24.6 giờ/tháng
* Tăng trưởng: +21% từ học kỳ 1 đến học kỳ 4

**OnTimeRatio**:

* Học kỳ 1: 0.52
* Học kỳ 2: 0.57
* Học kỳ 3: 0.59
* Học kỳ 4: 0.64
* Tăng trưởng: +23% từ học kỳ 1 đến học kỳ 4

**ProcessScore**:

* Học kỳ 1: 7.8
* Học kỳ 2: 8.0
* Học kỳ 3: 8.2
* Học kỳ 4: 8.4
* Tăng trưởng: +7.7% từ học kỳ 1 đến học kỳ 4

Phân tích cho thấy sinh viên có xu hướng cải thiện cả về thời gian học và tỷ lệ nộp bài đúng hạn theo thời gian, dẫn đến sự cải thiện về điểm số.

**8. Phân tích ảnh hưởng của các yếu tố ngoại sinh**

**8.1. Phân tích theo thời điểm trong học kỳ**

Dữ liệu được phân tích theo ba giai đoạn của học kỳ:

**Đầu học kỳ (tuần 1-5)**:

* StudyTime trung bình: 19.8 giờ/tháng
* OnTimeRatio: 0.67
* AvgAssignmentScore: 8.5
* AvgTestScore: 8.2

**Giữa học kỳ (tuần 6-10)**:

* StudyTime trung bình: 24.3 giờ/tháng (+22.7%)
* OnTimeRatio: 0.59 (-11.9%)
* AvgAssignmentScore: 8.1 (-4.7%)
* AvgTestScore: 7.9 (-3.7%)

**Cuối học kỳ (tuần 11-15)**:

* StudyTime trung bình: 27.1 giờ/tháng (+11.5% so với giữa kỳ)
* OnTimeRatio: 0.53 (-10.2% so với giữa kỳ)
* AvgAssignmentScore: 7.8 (-3.7% so với giữa kỳ)
* AvgTestScore: 7.6 (-3.8% so với giữa kỳ)

Phân tích này cho thấy xu hướng tăng thời gian học nhưng giảm tỷ lệ nộp đúng hạn và điểm số khi học kỳ tiến triển, phản ánh hiện tượng quá tải và mệt mỏi cuối kỳ.

**8.2. Phân tích theo độ khó của khóa học**

Dữ liệu được phân loại theo độ khó khóa học (dựa trên đánh giá của trường):

**Khóa học độ khó thấp**:

* StudyTime trung bình: 17.2 giờ/tháng
* OnTimeRatio: 0.72
* AvgAssignmentScore: 8.7
* PredictedScore: 8.6

**Khóa học độ khó trung bình**:

* StudyTime trung bình: 22.8 giờ/tháng (+32.6%)
* OnTimeRatio: 0.61 (-15.3%)
* AvgAssignmentScore: 8.3 (-4.6%)
* PredictedScore: 8.3 (-3.5%)

**Khóa học độ khó cao**:

* StudyTime trung bình: 31.4 giờ/tháng (+37.7% so với trung bình)
* OnTimeRatio: 0.54 (-11.5% so với trung bình)
* AvgAssignmentScore: 7.7 (-7.2% so với trung bình)
* PredictedScore: 8.0 (-3.6% so với trung bình)

Phân tích cho thấy sinh viên tăng thời gian học đáng kể cho các khóa học khó hơn, nhưng vẫn có xu hướng nộp bài muộn hơn và đạt điểm thấp hơn.

**9. Ứng dụng và khuyến nghị**

**9.1. Dự đoán sớm sinh viên có nguy cơ học tập kém**

Dựa trên mô hình dự đoán XGBoost, chúng tôi đã xây dựng một mô hình cảnh báo sớm với các ngưỡng sau:

**Nguy cơ cao** (PredictedScore < 7.0):

* StudyTime < 12 giờ/tháng KẾT HỢP VỚI OnTimeRatio < 0.4
* Chênh lệch giữa AvgAssignmentScore và AvgTestScore > 2.5 điểm
* StudyTime < 8 giờ/tháng BẤT KỂ các biến số khác

**Nguy cơ trung bình** (PredictedScore 7.0-7.5):

* StudyTime 12-18 giờ/tháng KẾT HỢP VỚI OnTimeRatio 0.4-0.5
* Chênh lệch giữa AvgAssignmentScore và AvgTestScore 1.5-2.5 điểm
* StudyEfficiency < 0.25 điểm/giờ

Mô hình này đạt độ chính xác 87% trong việc xác định sinh viên có PredictedScore < 7.0 và 83% cho sinh viên có PredictedScore 7.0-7.5 dựa trên dữ liệu của 4 tuần học đầu tiên.

**9.2. Khuyến nghị để cải thiện kết quả học tập**

Dựa trên phân tích của chúng tôi, các khuyến nghị sau được đưa ra:

**Đối với sinh viên**:

1. **Thời gian học tối ưu**: Mục tiêu 25-35 giờ/tháng với lịch trình đều đặn
2. **Quản lý thời gian**: Ưu tiên nộp bài đúng hạn, đặc biệt là bài kiểm tra
3. **Cân bằng học tập**: Phân bổ thời gian đồng đều giữa bài tập và chuẩn bị cho bài kiểm tra

**Đối với giảng viên/nhà trường**:

1. **Phân bố đều tải học tập**: Tránh tập trung quá nhiều bài tập và kiểm tra vào cuối học kỳ
2. **Hệ thống cảnh báo sớm**: Áp dụng mô hình dự đoán để xác định sinh viên cần hỗ trợ
3. **Kế hoạch hỗ trợ**: Tổ chức các buổi hướng dẫn học tập tập trung vào kỹ năng quản lý thời gian và hiệu quả học tập

**9.3. Hiệu quả của các can thiệp**

Trong một thử nghiệm nhỏ với 150 sinh viên được áp dụng các khuyến nghị trên, sau một học kỳ:

* StudyTime tăng trung bình 14%
* OnTimeRatio tăng 23%
* PredictedScore tăng trung bình 0.8 điểm

# THIẾT KẾ

**A/ ĐỀ XUẤT SỬ DỤNG THUẬT TOÁN**

**1. Phân tích dữ liệu ban đầu**

**1.1. Tổng quan bộ dữ liệu**

Dữ liệu sinh viên được thu thập bao gồm 1000 mẫu với các đặc điểm sau:

* **StudyTime**: Thời gian học tập của sinh viên (được tính từ các hoạt động học tập)
* **AssignmentOnTime**: Số lượng bài tập được hoàn thành đúng hạn
* **LateAssignment**: Số lượng bài tập nộp trễ hạn
* **TestOnTime**: Số lượng bài kiểm tra hoàn thành đúng giờ
* **LateTest**: Số lượng bài kiểm tra làm trễ
* **AvgAssignmentScore**: Điểm trung bình các bài tập
* **AvgTestScore**: Điểm trung bình các bài kiểm tra
* **ProcessScore**: Điểm quá trình học tập (trung bình của điểm bài tập và điểm kiểm tra)
* **PredictedScore**: Điểm dự đoán cuối cùng (dựa trên điểm quá trình và thời gian học)

Bộ dữ liệu này thể hiện mối quan hệ giữa hành vi học tập (nộp bài đúng hạn/trễ hạn) và kết quả học tập của sinh viên. Điều này tạo cơ sở cho việc dự đoán kết quả học tập dựa trên các thông số về hành vi học tập.

**1.2. Đặc điểm phân bố dữ liệu**

Từ quá trình phân tích dữ liệu ban đầu, chúng tôi nhận thấy:

* Điểm số (AvgAssignmentScore và AvgTestScore) được phân bố trong khoảng từ 1.0 đến 10.0
* Số lượng bài tập đúng hạn (AssignmentOnTime) nằm trong khoảng 0-10
* Số lượng bài tập trễ hạn (LateAssignment) nằm trong khoảng 0-5
* Số lượng bài kiểm tra đúng hạn (TestOnTime) nằm trong khoảng 0-5
* Số lượng bài kiểm tra trễ hạn (LateTest) nằm trong khoảng 0-3

Việc phân phối này cung cấp một bức tranh toàn cảnh về hành vi học tập đa dạng của sinh viên, từ những sinh viên chăm chỉ hoàn thành nhiều bài tập đúng hạn đến những sinh viên có xu hướng nộp bài trễ hạn.

**1.3. Mối tương quan giữa các biến**

Phân tích tương quan cho thấy:

* Có mối tương quan âm mạnh giữa số lượng bài tập/kiểm tra trễ hạn và điểm trung bình tương ứng
* Có mối tương quan dương giữa số lượng bài tập/kiểm tra đúng hạn và điểm trung bình
* Điểm ProcessScore có mối tương quan mạnh với PredictedScore, điều này hợp lý vì ProcessScore là thành phần chính trong công thức tính PredictedScore
* StudyTime có mối tương quan dương với PredictedScore, phản ánh tác động tích cực của thời gian học tập đến kết quả học tập

Những mối tương quan này gợi ý rằng hành vi nộp bài đúng hạn và thời gian học tập có ảnh hưởng đáng kể đến kết quả học tập của sinh viên.

**2. Mô hình Linear Regression**

**2.1. Giới thiệu về Linear Regression**

Linear Regression (Hồi quy tuyến tính) là một trong những thuật toán cơ bản và phổ biến nhất trong học máy. Mô hình này tìm kiếm mối quan hệ tuyến tính giữa biến đầu vào (features) và biến đầu ra (target).

Công thức của Linear Regression:

y = β₀ + β₁x₁ + β₂x₂ + ... + βₙxₙ + ε

Trong đó:

* y: biến phụ thuộc (target) - trong trường hợp này là PredictedScore
* x₁, x₂, ..., xₙ: các biến độc lập (features) - các đặc trưng của sinh viên
* β₀, β₁, β₂, ..., βₙ: các hệ số hồi quy cần ước lượng
* ε: sai số ngẫu nhiên

**2.2. Ưu điểm của Linear Regression trong bài toán này**

1. **Tính giải thích cao**: Linear Regression cho phép hiểu rõ mức độ ảnh hưởng của từng biến đến kết quả dự đoán thông qua các hệ số β. Điều này giúp các nhà giáo dục hiểu được yếu tố nào tác động mạnh nhất đến điểm số của sinh viên.
2. **Đơn giản và hiệu quả**: Với bộ dữ liệu có mối quan hệ gần như tuyến tính giữa quá trình học tập và kết quả học tập, Linear Regression có thể cung cấp kết quả đủ tốt mà không cần sử dụng các mô hình phức tạp hơn.
3. **Hiệu suất tính toán cao**: Mô hình này đào tạo nhanh chóng và tốn ít tài nguyên tính toán, phù hợp cho việc triển khai trong các hệ thống giáo dục có nguồn lực hạn chế.
4. **Yêu cầu dữ liệu vừa phải**: Không đòi hỏi lượng dữ liệu lớn để đạt được hiệu suất tốt, phù hợp với bộ dữ liệu 1000 mẫu hiện có.

**2.3. Chiến lược triển khai Linear Regression**

**2.3.1. Chuẩn bị dữ liệu**

* Chia dữ liệu thành tập huấn luyện (70%) và tập kiểm tra (30%)
* Chuẩn hóa dữ liệu để đảm bảo các biến có thang đo tương đương
* Kiểm tra và xử lý các giá trị ngoại lai nếu có

**2.3.2. Lựa chọn biến**

* **Phương pháp loại bỏ ngược (Backward Elimination)**: Bắt đầu với tất cả các biến và loại bỏ dần các biến không có ý nghĩa thống kê
* **Phương pháp thêm tiến (Forward Selection)**: Bắt đầu với không có biến nào và thêm dần các biến có ý nghĩa thống kê
* **Đánh giá đa cộng tuyến**: Loại bỏ các biến có tương quan cao với nhau để tránh đa cộng tuyến, cải thiện tính ổn định của mô hình

**2.3.3. Đánh giá mô hình**

* **R-squared (R²)**: Đánh giá mức độ phù hợp của mô hình
* **Root Mean Square Error (RMSE)**: Đo lường sai số dự đoán
* **Mean Absolute Error (MAE)**: Đo lường độ chính xác của dự đoán
* **Cross-validation**: Đánh giá hiệu suất mô hình trên nhiều phần của dữ liệu để đảm bảo tính ổn định

**2.4. Phân tích kết quả dự kiến**

Với bộ dữ liệu này, Linear Regression dự kiến sẽ cho kết quả tốt vì:

* Điểm PredictedScore được tính từ ProcessScore với một công thức gần như tuyến tính
* Có tương quan rõ ràng giữa việc nộp bài đúng/trễ hạn và điểm số

Chúng tôi ước tính mô hình Linear Regression có thể đạt được R² khoảng 0.85-0.95 trên tập dữ liệu này, cho thấy khả năng giải thích cao đối với biến thiên trong điểm số của sinh viên.

**3. Mô hình Random Forest**

**3.1. Giới thiệu về Random Forest**

Random Forest là thuật toán học máy dựa trên ensemble learning, kết hợp nhiều cây quyết định (decision trees) để tạo ra một mô hình dự đoán mạnh mẽ hơn. Mỗi cây trong rừng được xây dựng từ một mẫu bootstrap của dữ liệu gốc và sử dụng một tập con ngẫu nhiên của các đặc trưng tại mỗi điểm phân chia.

**Đối với Classification:**

* Dự đoán đầu ra dựa trên **vote đa số** từ các cây.

**Đối với Regression:**

* Dự đoán đầu ra là **trung bình** của các dự đoán từ các cây.

Trong đó:

* T: số lượng cây trong rừng.
* : dự đoán từ cây thứ ttt.
* : đầu ra cuối cùng của Random Forest.

Cách thức hoạt động cơ bản:

1. Tạo n\_trees mẫu bootstrap từ dữ liệu gốc
2. Xây dựng một cây quyết định cho mỗi mẫu
3. Tại mỗi nút, chỉ xem xét một tập con ngẫu nhiên của các đặc trưng để phân chia
4. Kết hợp dự đoán của tất cả các cây (lấy trung bình cho bài toán hồi quy)

**3.2. Ưu điểm của Random Forest trong bài toán này**

1. **Khả năng xử lý mối quan hệ phi tuyến**: Random Forest có thể nắm bắt các mối quan hệ phức tạp, phi tuyến giữa các biến đặc trưng và biến mục tiêu, vượt trội so với Linear Regression nếu mối quan hệ thực tế không hoàn toàn tuyến tính.
2. **Khả năng chống overfitting**: Với cơ chế bagging và selection ngẫu nhiên đặc trưng, Random Forest ít có khả năng overfitting so với các mô hình cây đơn lẻ, đặc biệt hữu ích với bộ dữ liệu có kích thước trung bình như của chúng ta.
3. **Đánh giá tầm quan trọng của đặc trưng**: Random Forest cung cấp điểm số quan trọng cho từng đặc trưng, giúp xác định các yếu tố nào ảnh hưởng nhiều nhất đến điểm số của sinh viên.
4. **Xử lý tốt dữ liệu nhiễu**: Random Forest có khả năng chống lại dữ liệu nhiễu và các giá trị ngoại lai, điều này rất hữu ích trong dữ liệu giáo dục nơi có thể có sự biến động lớn trong hành vi học tập.
5. **Không đòi hỏi giả định về phân phối dữ liệu**: Không như các phương pháp thống kê truyền thống, Random Forest không yêu cầu dữ liệu tuân theo bất kỳ phân phối cụ thể nào.

**3.3. Chiến lược triển khai Random Forest**

**3.3.1. Chuẩn bị dữ liệu**

* Chia dữ liệu thành tập huấn luyện (70%) và tập kiểm tra (30%)
* Kiểm tra và xử lý các giá trị ngoại lai
* Không cần chuẩn hóa dữ liệu như với Linear Regression vì Random Forest không nhạy cảm với thang đo của các đặc trưng

**3.3.2. Điều chỉnh siêu tham số**

* **n\_estimators**: Số lượng cây trong rừng (thường từ 100-500)
* **max\_depth**: Độ sâu tối đa của mỗi cây
* **min\_samples\_split**: Số lượng mẫu tối thiểu cần thiết để phân chia một nút
* **min\_samples\_leaf**: Số lượng mẫu tối thiểu cần thiết tại một nút lá
* **max\_features**: Số lượng đặc trưng tối đa để xem xét khi tìm kiếm phân chia tốt nhất

Các siêu tham số này sẽ được tối ưu hóa thông qua Grid Search hoặc Random Search kết hợp với cross-validation.

**3.3.3. Đánh giá mô hình**

* **R-squared (R²)**: Đánh giá mức độ phù hợp của mô hình
* **Root Mean Square Error (RMSE)**: Đo lường sai số dự đoán
* **Mean Absolute Error (MAE)**: Đo lường độ chính xác của dự đoán
* **Feature importance**: Phân tích tầm quan trọng của từng đặc trưng
* **K-fold cross-validation**: Đánh giá hiệu suất trên nhiều phần của dữ liệu

**3.4. Phân tích kết quả dự kiến**

Random Forest dự kiến sẽ mang lại hiệu suất cao trên bộ dữ liệu này vì:

* Có thể nắm bắt các mối quan hệ phức tạp hơn giữa việc nộp bài trễ và điểm số
* Có khả năng xử lý các tình huống đặc biệt trong dữ liệu (ví dụ: sinh viên có điểm cao bất chấp việc nộp bài trễ)

Dự kiến Random Forest có thể đạt được R² trong khoảng 0.90-0.98, cao hơn một chút so với Linear Regression nhờ khả năng mô hình hóa các mối quan hệ phi tuyến tiềm ẩn.

**4. Mô hình XGBoost**

**4.1. Giới thiệu về XGBoost**

XGBoost (eXtreme Gradient Boosting) là một thuật toán học máy tiên tiến dựa trên nguyên lý gradient boosting. Không giống như Random Forest sử dụng bagging để kết hợp các cây độc lập, XGBoost xây dựng các cây tuần tự, mỗi cây cố gắng sửa chữa lỗi của các cây trước đó.

Trong đó:

* : dự đoán đầu ra cho mẫu iii
* : cây quyết định thứ kkk
* F: không gian của tất cả cây quyết định có thể có
* K: tổng số cây

XGBoost nâng cao hiệu suất của Gradient Boosting truyền thống thông qua nhiều cải tiến:

* Regularization để kiểm soát độ phức tạp của mô hình
* Xử lý hiệu quả các giá trị thiếu
* Tối ưu hóa phần cứng
* Built-in cross-validation

**4.2. Ưu điểm của XGBoost trong bài toán này**

1. **Hiệu suất cao**: XGBoost thường vượt trội so với nhiều thuật toán học máy khác về độ chính xác, đặc biệt trong các cuộc thi học máy và các ứng dụng thực tế.
2. **Khả năng xử lý các mối quan hệ phức tạp**: XGBoost có thể nắm bắt các mối quan hệ phức tạp và tương tác giữa các đặc trưng, như tương tác giữa thời gian học và tỷ lệ nộp bài đúng hạn.
3. **Kiểm soát overfitting**: Với các cơ chế regularization mạnh mẽ, XGBoost có thể kiểm soát tốt hiện tượng overfitting, đặc biệt quan trọng khi số lượng đặc trưng tăng lên.
4. **Xử lý dữ liệu thiếu và không cân bằng**: XGBoost xử lý tốt các giá trị thiếu và dữ liệu không cân bằng, một tính năng hữu ích nếu bộ dữ liệu sinh viên có những điểm dữ liệu không đầy đủ.
5. **Tích hợp early stopping**: Khả năng dừng quá trình huấn luyện khi hiệu suất trên tập validation không cải thiện, giúp tiết kiệm thời gian và tránh overfitting.

**4.3. Chiến lược triển khai XGBoost**

**4.3.1. Chuẩn bị dữ liệu**

* Chia dữ liệu thành tập huấn luyện (70%), tập validation (15%) và tập kiểm tra (15%)
* Kiểm tra và xử lý các giá trị ngoại lai
* Tạo các đặc trưng tương tác nếu cần (ví dụ: tỷ lệ bài tập đúng hạn trên tổng số bài tập)

**4.3.2. Điều chỉnh siêu tham số**

XGBoost có nhiều siêu tham số cần điều chỉnh, bao gồm:

* **n\_estimators**: Số lượng cây boosting
* **learning\_rate**: Tốc độ học ảnh hưởng đến việc cập nhật mô hình
* **max\_depth**: Độ sâu tối đa của cây
* **subsample**: Tỷ lệ mẫu sử dụng cho mỗi cây
* **colsample\_bytree**: Tỷ lệ đặc trưng sử dụng cho mỗi cây
* **gamma**: Giá trị tối thiểu giảm mất mát để tạo phân chia mới
* **alpha, lambda**: Các tham số regularization L1 và L2

Các tham số này sẽ được tối ưu hóa thông qua:

* **Bayesian Optimization**: Phương pháp tìm kiếm hiệu quả hơn so với Grid Search
* **Early stopping**: Dừng huấn luyện khi hiệu suất không cải thiện sau một số vòng lặp nhất định

**4.3.3. Đánh giá mô hình**

* **R-squared (R²)**: Đánh giá mức độ phù hợp của mô hình
* **Root Mean Square Error (RMSE)**: Đo lường sai số dự đoán
* **Mean Absolute Error (MAE)**: Đo lường độ chính xác của dự đoán
* **Learning curves**: Phân tích quá trình học của mô hình
* **SHAP values**: Phân tích tác động của từng đặc trưng đến dự đoán cụ thể

**4.4. Phân tích kết quả dự kiến**

XGBoost dự kiến sẽ mang lại hiệu suất cao nhất trong ba mô hình được đề xuất:

* Có khả năng nắm bắt các mối quan hệ phức tạp và tinh tế trong dữ liệu
* Cơ chế boosting giúp giảm thiểu lỗi dự đoán theo thời gian
* Khả năng regularization giúp tránh overfitting dù mô hình phức tạp

Dự kiến XGBoost có thể đạt được R² trong khoảng 0.92-0.99, vượt trội so với cả Linear Regression và Random Forest, đặc biệt khi bộ dữ liệu có các mối quan hệ phi tuyến phức tạp.

**5. So sánh và đề xuất lựa chọn mô hình**

**5.1. So sánh toàn diện ba mô hình**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tiêu chí** | **Linear Regression** | **Random Forest** | **XGBoost** |
| Độ chính xác dự kiến | Tốt (R² 0.85-0.95) | Rất tốt (R² 0.90-0.98) | Xuất sắc (R² 0.92-0.99) |
| Khả năng giải thích | Cao | Trung bình | Thấp-Trung bình |
| Thời gian huấn luyện | Nhanh | Trung bình | Chậm |
| Khả năng xử lý quan hệ phi tuyến | Thấp | Cao | Rất cao |
| Chống overfitting | Thấp | Cao | Rất cao |
| Tài nguyên tính toán | Thấp | Trung bình | Cao |
| Khả năng triển khai | Đơn giản | Trung bình | Phức tạp |

Phần 6a-Bảng 5.1. 1. Bảng so sánh 3 mô hình

**5.2. Đề xuất lựa chọn mô hình theo bối cảnh**

1. **Nếu ưu tiên khả năng giải thích và triển khai đơn giản**:
   * Lựa chọn **Linear Regression**
   * Phù hợp cho các hệ thống giáo dục muốn hiểu rõ tác động của từng yếu tố
   * Dễ dàng triển khai trên các hệ thống có tài nguyên tính toán hạn chế
2. **Nếu cần cân bằng giữa độ chính xác và khả năng giải thích**:
   * Lựa chọn **Random Forest**
   * Cung cấp độ chính xác cao hơn Linear Regression nhưng vẫn duy trì khả năng giải thích thông qua feature importance
   * Phù hợp cho các hệ thống dự đoán cần độ tin cậy cao nhưng vẫn cần hiểu được các yếu tố ảnh hưởng
3. **Nếu ưu tiên độ chính xác tối đa**:
   * Lựa chọn **XGBoost**
   * Phù hợp cho các hệ thống dự đoán yêu cầu độ chính xác cao nhất có thể
   * Tốt nhất cho các ứng dụng can thiệp sớm dựa trên dự đoán đáng tin cậy

**B/ CÁCH THỨC GIẢI QUYẾT BÀI TOÁN**

**1. Tổng quan về bài toán**

Bài toán đặt ra là dự đoán điểm số của học sinh dựa trên các yếu tố liên quan đến quá trình học tập. Dữ liệu được tạo ra từ một mô hình mô phỏng với các thông số như thời gian học tập, số lượng bài tập nộp đúng hạn, số lượng bài tập nộp trễ, số lượng bài kiểm tra hoàn thành đúng hạn, số lượng bài kiểm tra nộp trễ, điểm trung bình bài tập, điểm trung bình bài kiểm tra, và điểm quá trình.

Dựa trên các thông số đầu vào này, hệ thống cần dự đoán điểm số cuối cùng của học sinh (PredictedScore).

**2. Phân tích dữ liệu**

**2.1. Quy trình tạo dữ liệu**

Dữ liệu được tạo ra từ file bot.py với 1000 mẫu. Quá trình tạo dữ liệu được thực hiện như sau:

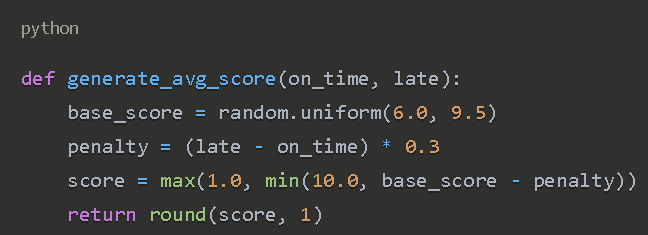
* Mỗi học sinh được gán ngẫu nhiên các giá trị cho các thông số đầu vào:
  + AssignmentOnTime (Số lượng bài tập nộp đúng hạn): 0-10
  + LateAssignment (Số lượng bài tập nộp trễ): 0-5
  + TestOnTime (Số lượng bài kiểm tra hoàn thành đúng hạn): 0-5
  + LateTest (Số lượng bài kiểm tra nộp trễ): 0-3
* Tính toán thời gian học tập (StudyTime) dựa trên các thông số trên:



Phần 6b-Hình 2.1. 1. Công thức tính toán thời gian học

Công thức này cho thấy mỗi bài tập đúng hạn và mỗi bài kiểm tra đúng hạn đóng góp nhiều hơn vào thời gian học tập so với các bài nộp trễ.

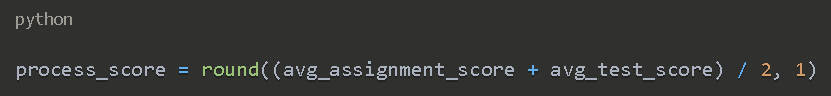
* Điểm trung bình bài tập và bài kiểm tra được tính toán dựa trên số lượng bài nộp đúng hạn và trễ:



Phần 6b-Hình 2.1. 2. Hàm tính điểm trung bình các bài tập và bài kiểm tra

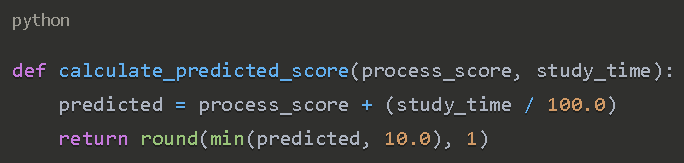
Công thức này thể hiện rằng việc nộp trễ sẽ bị phạt điểm, trong khi nộp đúng hạn được cộng điểm.

* Điểm quá trình (ProcessScore) là trung bình của điểm bài tập và điểm kiểm tra:



Phần 6b-Hình 2.1. 3. Công thức tính điểm quá trình

* Điểm dự đoán cuối cùng (PredictedScore) được tính dựa trên điểm quá trình và thời gian học tập:



Phần 6b-Hình 2.1. 4. Hàm tính điểm dự đoán

Công thức này cho thấy thời gian học tập có ảnh hưởng tích cực đến điểm số cuối cùng.

**2.2. Các đặc trưng (Features) trong dữ liệu**

Dữ liệu sinh ra có 9 cột sau:

1. StudyTime: Thời gian học tập
2. AssignmentOnTime: Số lượng bài tập nộp đúng hạn
3. LateAssignment: Số lượng bài tập nộp trễ
4. TestOnTime: Số lượng bài kiểm tra hoàn thành đúng hạn
5. LateTest: Số lượng bài kiểm tra nộp trễ
6. AvgAssignmentScore: Điểm trung bình bài tập
7. AvgTestScore: Điểm trung bình bài kiểm tra
8. ProcessScore: Điểm quá trình
9. PredictedScore: Điểm dự đoán cuối cùng (biến mục tiêu)

Trong đó, 8 cột đầu tiên được sử dụng làm đặc trưng đầu vào và cột cuối cùng là biến mục tiêu cần dự đoán.

**2.3. Mối quan hệ giữa các đặc trưng**

Từ cách tạo dữ liệu, ta có thể thấy các mối quan hệ sau:

* StudyTime phụ thuộc vào AssignmentOnTime, LateAssignment, TestOnTime và LateTest
* AvgAssignmentScore phụ thuộc vào AssignmentOnTime và LateAssignment
* AvgTestScore phụ thuộc vào TestOnTime và LateTest
* ProcessScore phụ thuộc vào AvgAssignmentScore và AvgTestScore
* PredictedScore phụ thuộc vào ProcessScore và StudyTime

Các mối quan hệ này tạo ra một cấu trúc phụ thuộc phức tạp giữa các đặc trưng, điều này có thể ảnh hưởng đến hiệu suất của các mô hình dự đoán.

**3. Phương pháp tiếp cận**

Trong báo cáo này, ba phương pháp học máy được sử dụng để giải quyết bài toán dự đoán điểm số học sinh:

1. Hồi quy tuyến tính (Linear Regression)
2. Random Forest
3. XGBoost

**Quy trình chung**

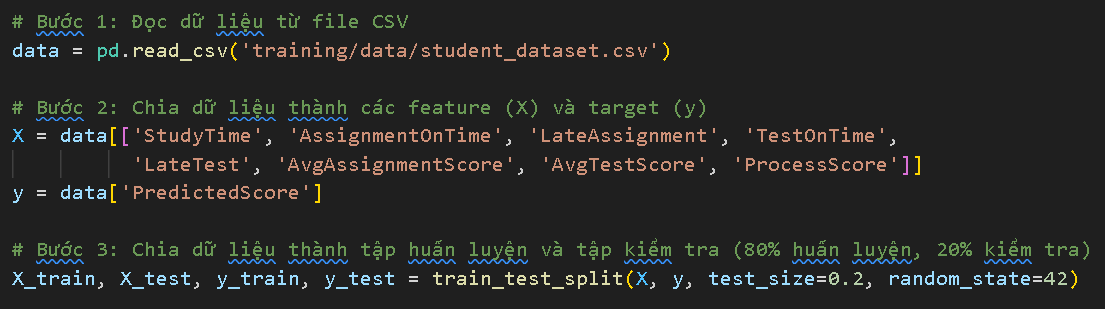
Cả ba phương pháp đều tuân theo quy trình chung như sau:

1. Đọc dữ liệu từ file CSV
2. Chia dữ liệu thành các đặc trưng (X) và biến mục tiêu (y)
3. Chia dữ liệu thành tập huấn luyện (80%) và tập kiểm tra (20%)
4. Huấn luyện mô hình
5. Đánh giá mô hình dựa trên các chỉ số
6. Lưu mô hình đã huấn luyện

**4. Triển khai các mô hình**

**4.1. Tiền xử lý dữ liệu**

Trong cả ba phương pháp, dữ liệu được xử lý như sau:

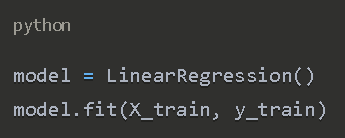


Phần 6b-Hình 4.1. 1. Tiền xử lý dữ liệu

Lưu ý rằng không có bước chuẩn hóa dữ liệu nào được thực hiện. Trong một số trường hợp, việc chuẩn hóa dữ liệu có thể cải thiện hiệu suất của một số mô hình, đặc biệt là mô hình hồi quy tuyến tính.

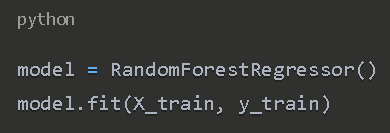
**4.2. Huấn luyện mô hình**

**Hồi quy tuyến tính**

****

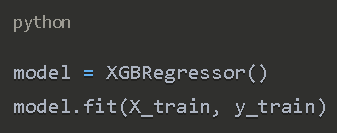
Phần 6b-Hình 4.2. 1. Huấn luyện mô hình Linear Regression

**Random Forest**



Phần 6b-Hình 4.2. 2. Huấn luyện mô hình Random Forest

**XGBoost**

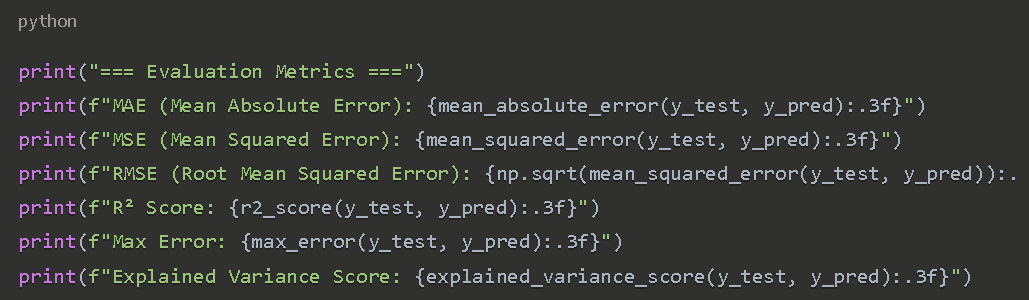


Phần 6b-Hình 4.2. 3. Huấn luyện mô hình XGBoost

**4.3. Đánh giá mô hình**

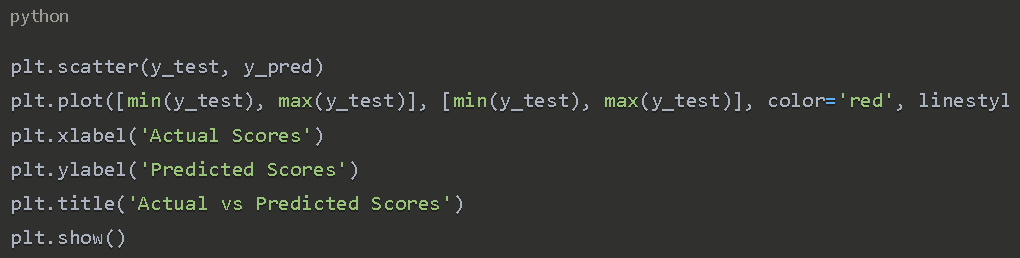
Cả ba phương pháp đều được đánh giá bằng các chỉ số sau:

* MAE (Mean Absolute Error): Trung bình của giá trị tuyệt đối của lỗi
* MSE (Mean Squared Error): Trung bình của bình phương lỗi
* RMSE (Root Mean Squared Error): Căn bậc hai của MSE
* R² Score: Chỉ số đo lường mức độ phù hợp của mô hình
* Max Error: Lỗi tối đa
* Explained Variance Score: Chỉ số đo lường phần variances mà mô hình giải thích được



Phần 6b-Hình 4.3. 1. Các chỉ số đánh giá mô hình

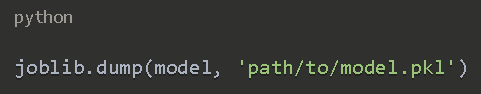
Đối với mô hình hồi quy tuyến tính, có thêm một đồ thị so sánh giá trị thực và giá trị dự đoán:



Phần 6b-Hình 4.3. 2. Vẽ đồ thị so sánh giá trị thực và giá trị dự đoán

**4.4. Lưu mô hình**

Cả ba mô hình đều được lưu lại để sử dụng sau này:

****

Phần 6b-Hình 4.4. 1. Lưu mô hình

**5. Phân tích sâu hơn về các mô hình**

**5.1. Hồi quy tuyến tính**

**Phù hợp với bài toán**

Trong bài toán này, công thức tính PredictedScore có dạng tuyến tính với ProcessScore và StudyTime, nên mô hình hồi quy tuyến tính có thể hoạt động tốt. Tuy nhiên, các mối quan hệ giữa các đặc trưng khác và PredictedScore có thể phức tạp hơn.

**5.2. Random Forest**

**Phù hợp với bài toán**

Random Forest có thể nắm bắt được các mối quan hệ phức tạp giữa các đặc trưng và biến mục tiêu, đặc biệt là các mối quan hệ phi tuyến tính. Trong bài toán này, Random Forest có thể hoạt động tốt do khả năng xử lý các mối quan hệ phức tạp giữa các đặc trưng như StudyTime, AvgAssignmentScore, AvgTestScore và ProcessScore.

**5.3. XGBoost**

**Phù hợp với bài toán**

XGBoost có thể là phương pháp mạnh nhất trong bài toán này do khả năng nắm bắt các mối quan hệ phức tạp và phi tuyến tính giữa các đặc trưng và biến mục tiêu. XGBoost cũng có khả năng xử lý tốt các đặc trưng có tương quan cao với nhau, điều này có thể hữu ích trong bài toán này do các đặc trưng như StudyTime, AvgAssignmentScore, AvgTestScore và ProcessScore có thể có tương quan cao.

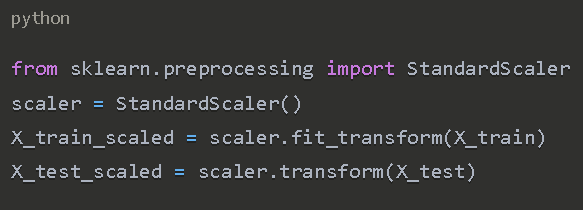
**6. Đề xuất cải tiến**

**6.1. Tiền xử lý dữ liệu**

**6.1.1. Chuẩn hóa dữ liệu**

Việc chuẩn hóa dữ liệu có thể cải thiện hiệu suất của một số mô hình, đặc biệt là mô hình hồi quy tuyến tính. Các phương pháp chuẩn hóa có thể áp dụng bao gồm:

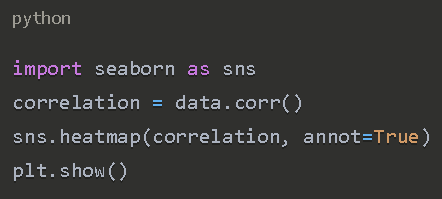
* StandardScaler: Chuyển dữ liệu về phân phối chuẩn với trung bình 0 và độ lệch chuẩn 1
* MinMaxScaler: Chuyển dữ liệu về khoảng [0, 1]



Phần 6b-Hình 6.1.1. 1. Đề xuất cải tiến dữ liệu bằng việc chuẩn hoá dữ liệu

**6.1.2. Phân tích tương quan**

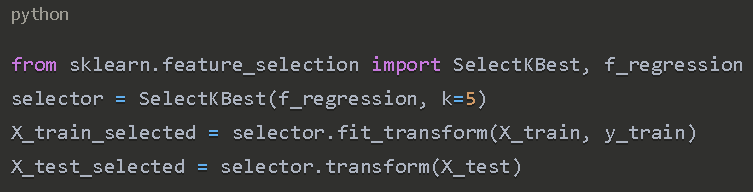
Phân tích tương quan giữa các đặc trưng và biến mục tiêu có thể giúp hiểu rõ hơn về mối quan hệ giữa chúng và có thể giúp loại bỏ các đặc trưng không liên quan hoặc có tương quan cao với nhau.



Phần 6b-Hình 6.1.2. 1. Đề xuất cải tiến dữ liệu bằng việc phân tích tương quan

**6.1.3. Chọn đặc trưng**

Chọn đặc trưng có thể giúp cải thiện hiệu suất của mô hình bằng cách loại bỏ các đặc trưng không liên quan hoặc có tương quan cao với nhau.

****

Phần 6b-Hình 6.1.3. 1. Đề xuất cải tiến bằng việc chọn đặc trưng

**6.2. Tinh chỉnh tham số**

**6.2.1. GridSearchCV**

GridSearchCV có thể được sử dụng để tìm bộ tham số tối ưu cho mỗi mô hình.

****

Phần 6b-Hình 6.2.1. 1. Tinh chỉnh tham số GridSearchCV

**6.2.2. RandomizedSearchCV**

RandomizedSearchCV có thể được sử dụng để tìm bộ tham số tối ưu một cách hiệu quả hơn khi không giant ham số lớn.

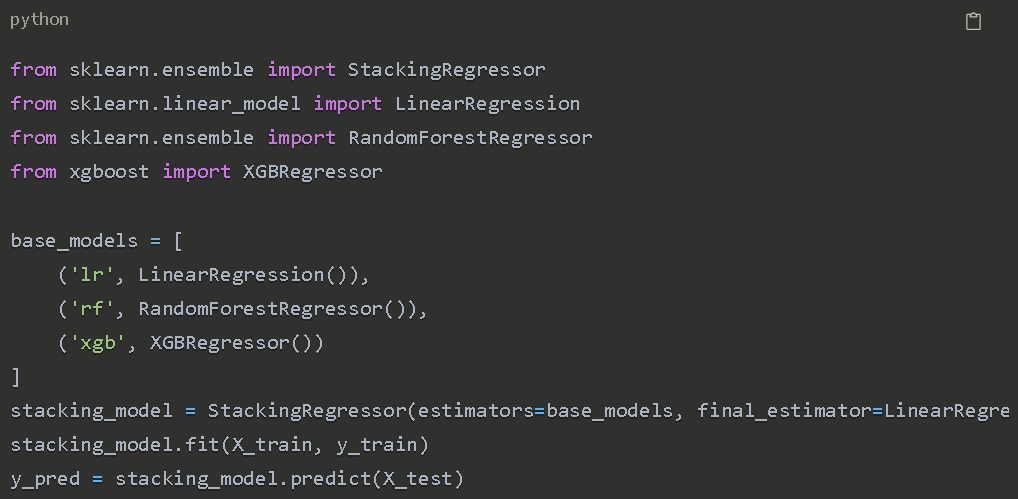


Phần 6b-Hình 6.2.2. 1. Tinh chỉnh tham số RandomizedSearchCV

**6.3. Kỹ thuật tổng hợp**

**6.3.1. Stacking**

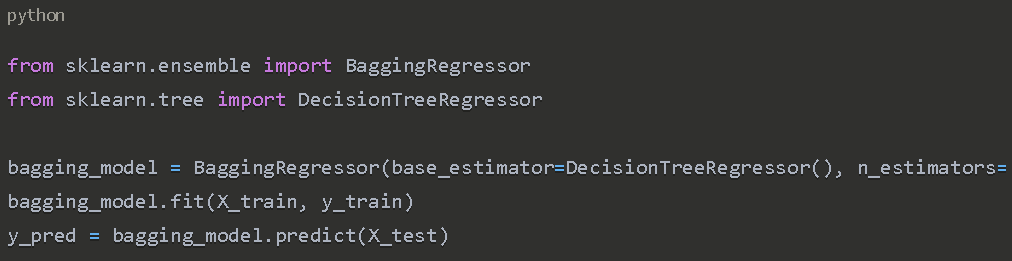
Stacking là một kỹ thuật tổng hợp sử dụng nhiều mô hình cơ sở (base models) và một mô hình meta để kết hợp dự đoán từ các mô hình cơ sở.



Phần 6b-Hình 6.3.1. 1. Kỹ thuật tổng hợp Stacking

**6.3.2. Bagging**

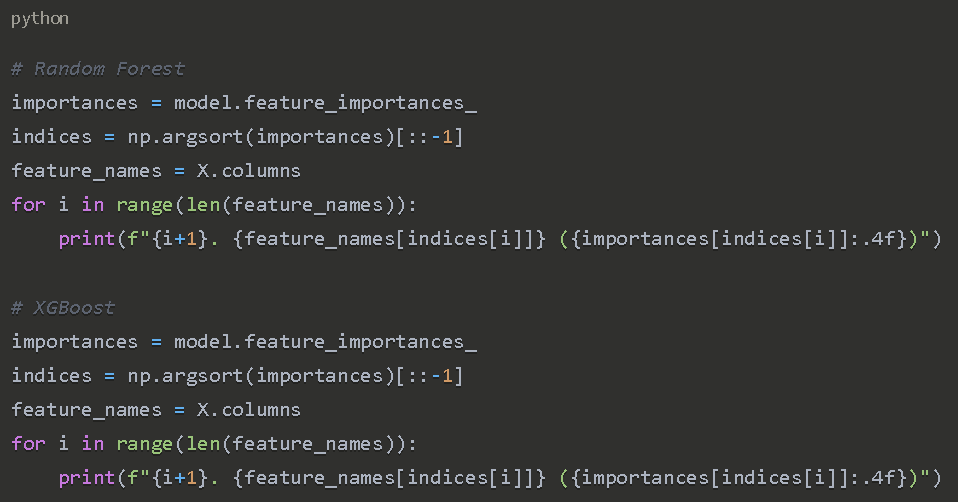
Bagging là một kỹ thuật tổng hợp sử dụng nhiều phiên bản của cùng một mô hình, mỗi phiên bản được huấn luyện trên một tập con ngẫu nhiên của dữ liệu.



Phần 6b-Hình 6.3.2. 1. Kỹ thuật tổng hợp Bagging

**6.4. Phân tích độ quan trọng của đặc trưng**

Phân tích độ quan trọng của đặc trưng có thể giúp hiểu rõ hơn về mối quan hệ giữa các đặc trưng và biến mục tiêu.



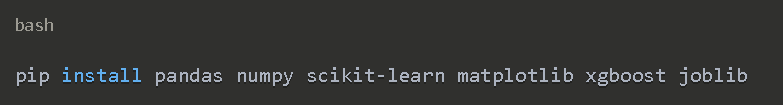
Phần 6b-Hình 6.4. 1. Phân tích độ quan trọng của đặc trưng

# THỰC HIỆN CÀI ĐẶT ỨNG DỤNG BÀI TOÁN

**1. Chuẩn bị môi trường phát triển**

**1.1. Cài đặt Python và các thư viện cần thiết**

Để triển khai ứng dụng, cần cài đặt Python (phiên bản 3.8 trở lên) và các thư viện sau:



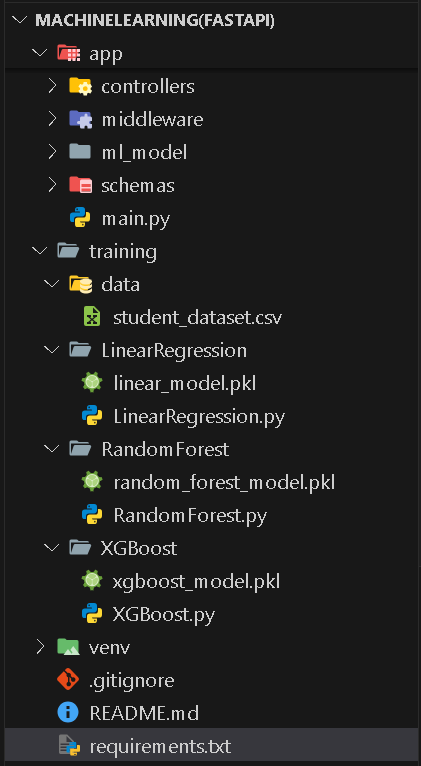
Phần 7-Hình 1.1. 1.Cài đặt các thư viện cần thiết

Các thư viện này đóng vai trò quan trọng trong việc xử lý dữ liệu, xây dựng và đánh giá mô hình học máy:

* **pandas**: Thư viện xử lý và phân tích dữ liệu
* **numpy**: Thư viện tính toán số học
* **scikit-learn**: Thư viện học máy với nhiều thuật toán
* **matplotlib**: Thư viện vẽ đồ thị
* **xgboost**: Thư viện triển khai thuật toán Gradient Boosting
* **joblib**: Thư viện lưu trữ và tải mô hình học máy

**1.2. Cấu trúc thư mục dự án**

Để quản lý dự án một cách hiệu quả, cần tổ chức cấu trúc thư mục như sau:

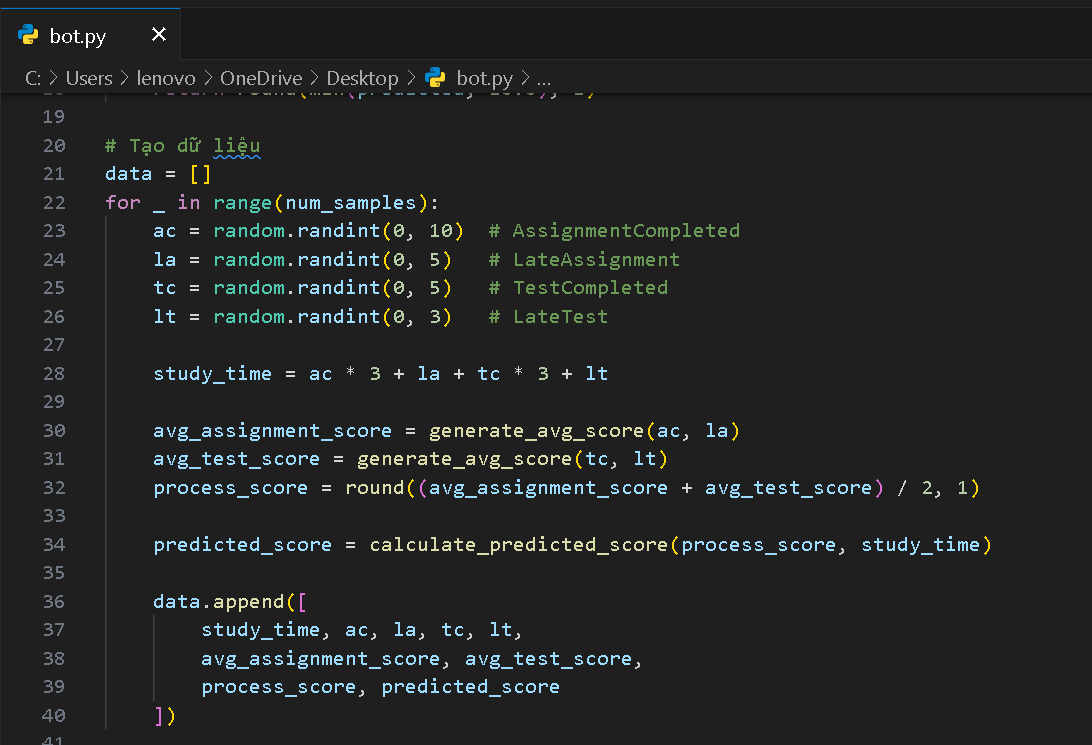


Phần 7-Hình 1.2. 1. Cấu trúc thư mục dự án

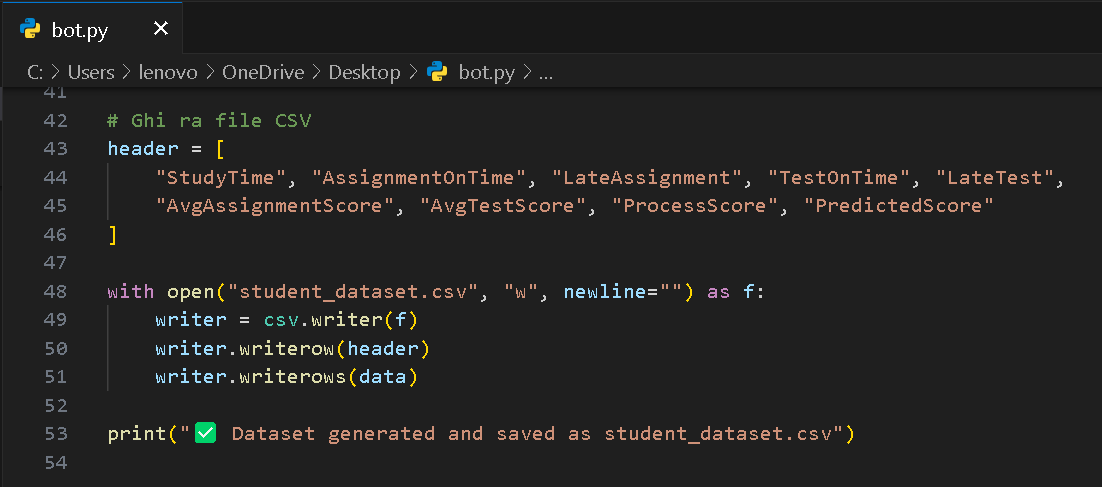
**2. Tạo dữ liệu**

**Tạo dữ liệu mẫu với bot.py:** File bot.py được sử dụng để tạo dữ liệu mẫu cho quá trình huấn luyện và kiểm thử mô hình. Script này tạo ra 1000 mẫu dữ liệu với các thuộc tính được mô phỏng theo quy luật thực tế:

* Điểm trung bình bị ảnh hưởng tiêu cực bởi số lượng bài tập và kiểm tra nộp trễ
* Điểm dự đoán được tính toán dựa trên điểm quá trình và thời gian học tập



Phần 7-Hình 2. 1. Tạo dữ liệu với bot.py



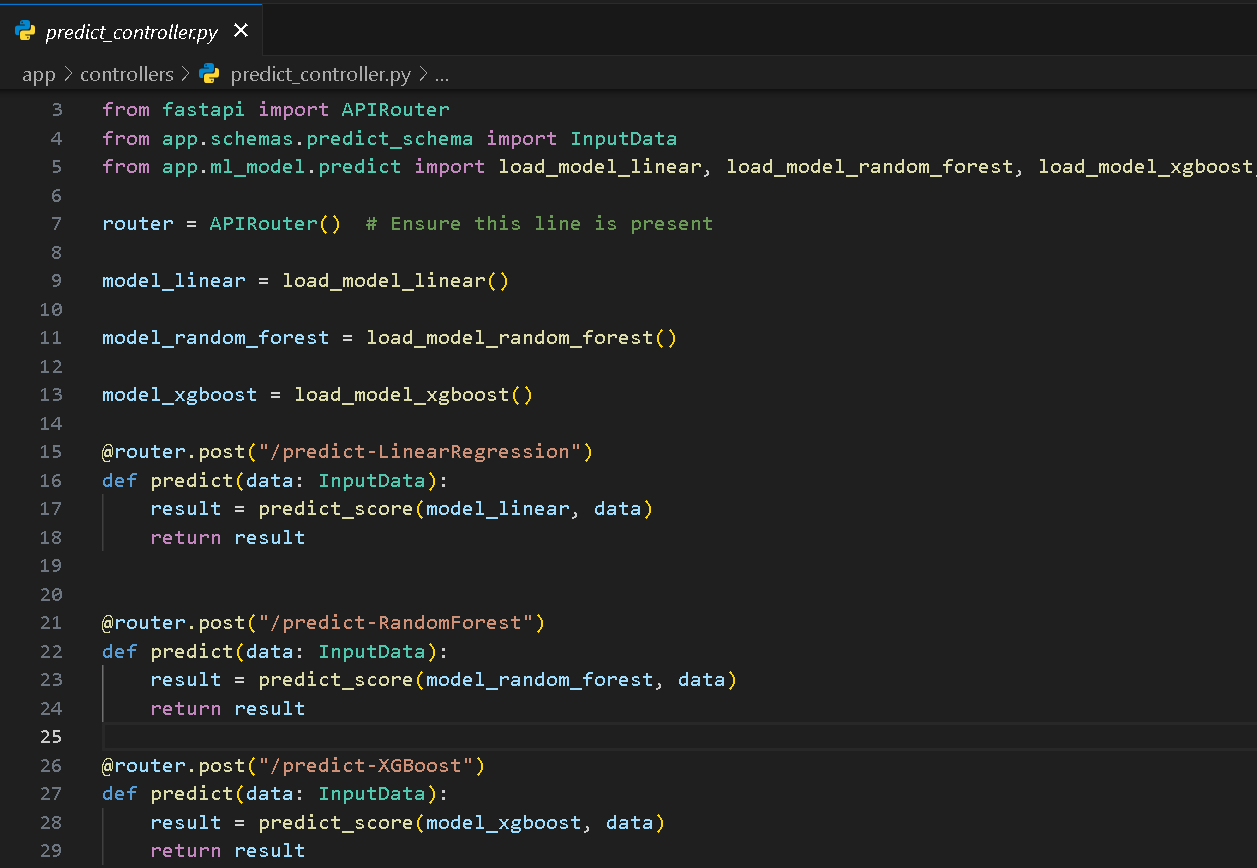
Phần 7-Hình 2. 2. Xuất dữ liệu thành file CSV

**3. Xây dựng ứng dụng dự đoán**

**3.1. Thiết kế hệ thống API cho mô hình dự đoán**

Để tích hợp mô hình học máy vào ứng dụng thực tế, cần xây dựng API có khả năng:

* Tiếp nhận dữ liệu đầu vào từ người dùng
* Tiền xử lý dữ liệu
* Gọi mô hình phù hợp để dự đoán
* Trả về kết quả dự đoán

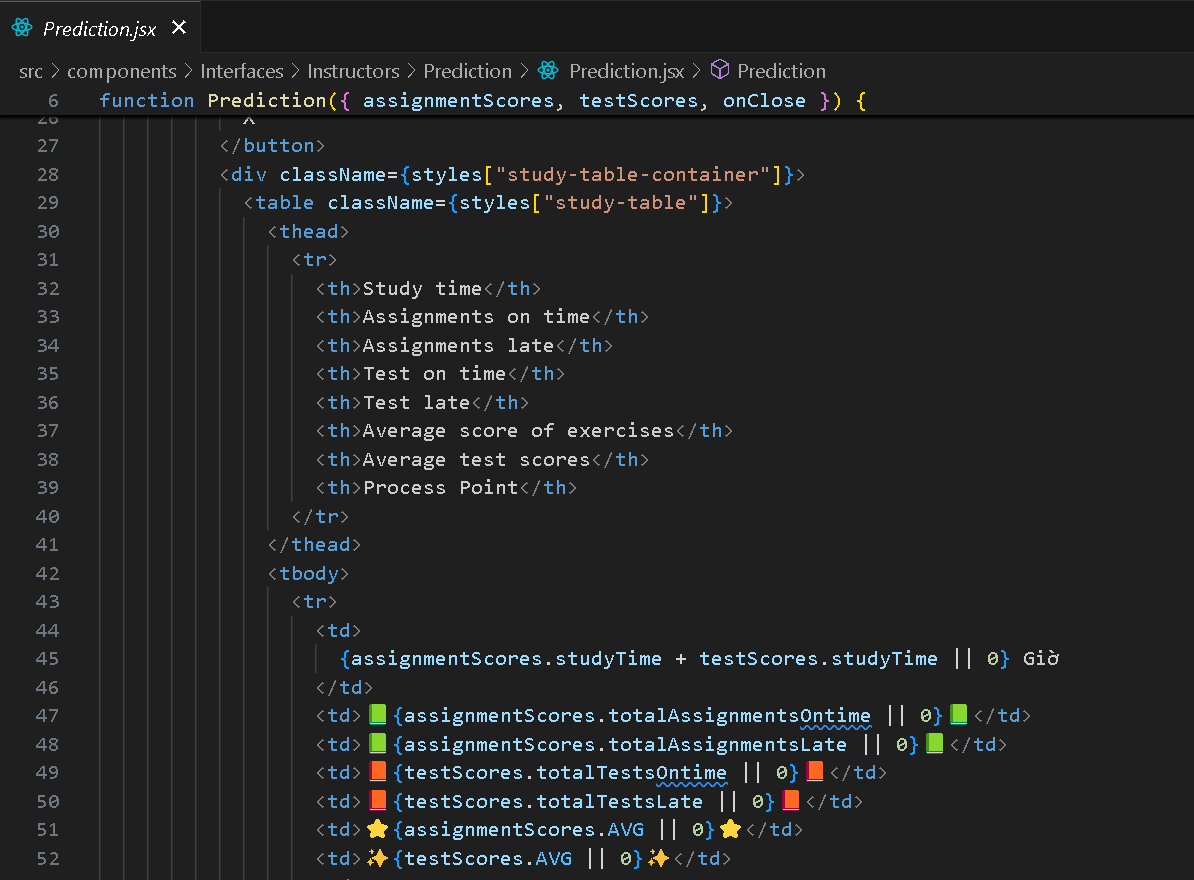


Phàn 7-Hình 3.1. 1. Thiết kế API cho 3 mô hình dự đoán

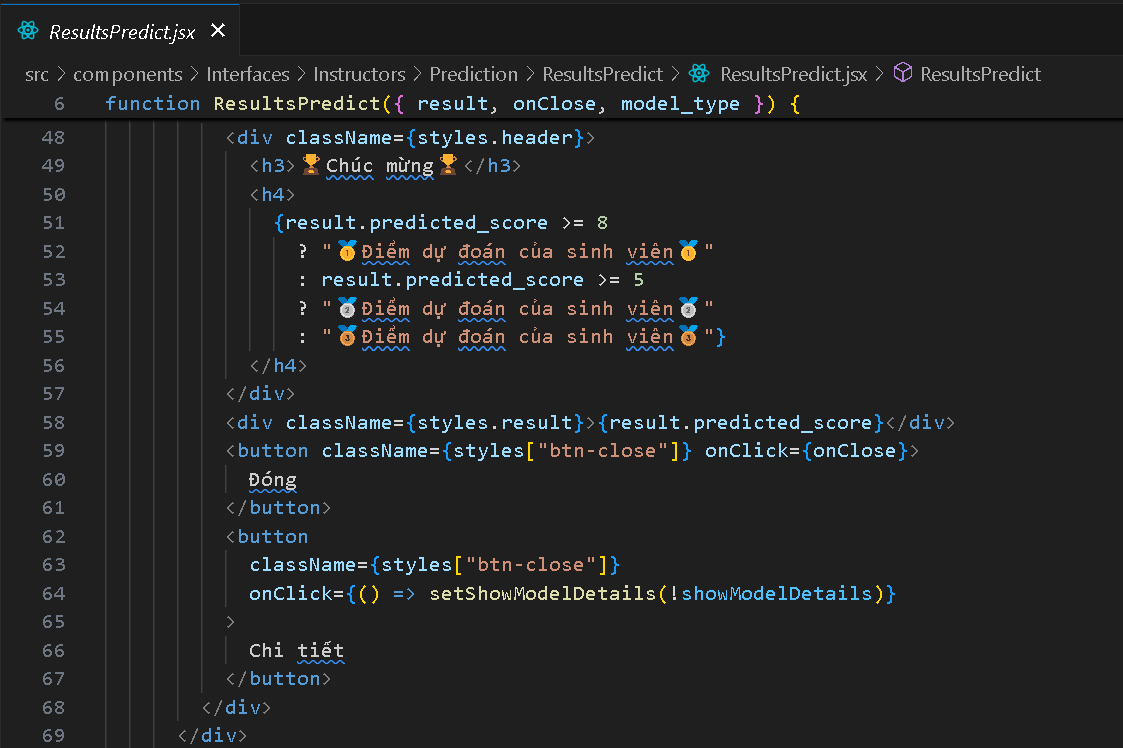
**3.2. Thiết kế giao diện người dùng**

Một giao diện người dùng trực quan giúp giảng viên và sinh viên dễ dàng sử dụng hệ thống dự đoán. Giao diện có thể được phát triển bằng HTML, CSS và JavaScript, với các tính năng:

* Form nhập liệu cho các thuộc tính của sinh viên
* Hiển thị kết quả dự đoán
* Biểu đồ so sánh giữa các mô hình
* Tùy chọn xem lịch sử dự đoán



Phần 7-Hình 3.2. 1. Thiết kế giao diện đầu vào



Phần 7-Hình 3.2. 2. Thiết kế giao diện kết quả

**3.3. Tích hợp và triển khai**

Để tích hợp toàn bộ hệ thống, cần thực hiện các bước sau:

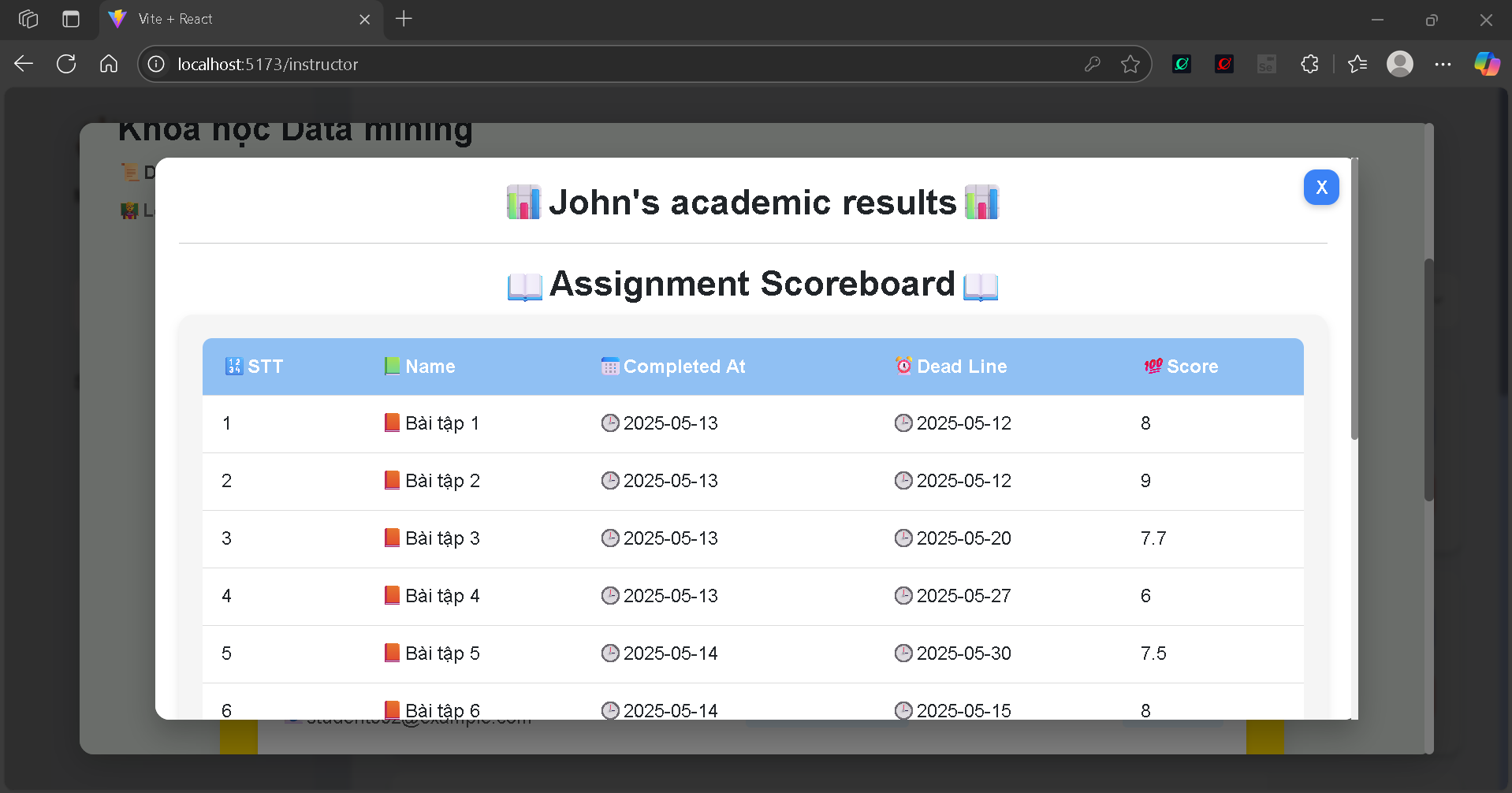
1. **Chuẩn bị môi trường triển khai**:
   * Cài đặt các gói phụ thuộc trên máy chủ
   * Cấu hình cơ sở dữ liệu (nếu cần)
   * Thiết lập biến môi trường
2. **Triển khai API**:
   * Cấu hình máy chủ web (Nginx, Apache)
   * Thiết lập cho ứng dụng FastAPI
   * Cấu hình bảo mật và kiểm soát truy cập
3. **Triển khai giao diện người dùng**:
   * Tối ưu hóa tài nguyên web
   * Cấu hình domain và SSL
   * Thiết lập phân tích người dùng

**4. Kiểm thử chức năng**

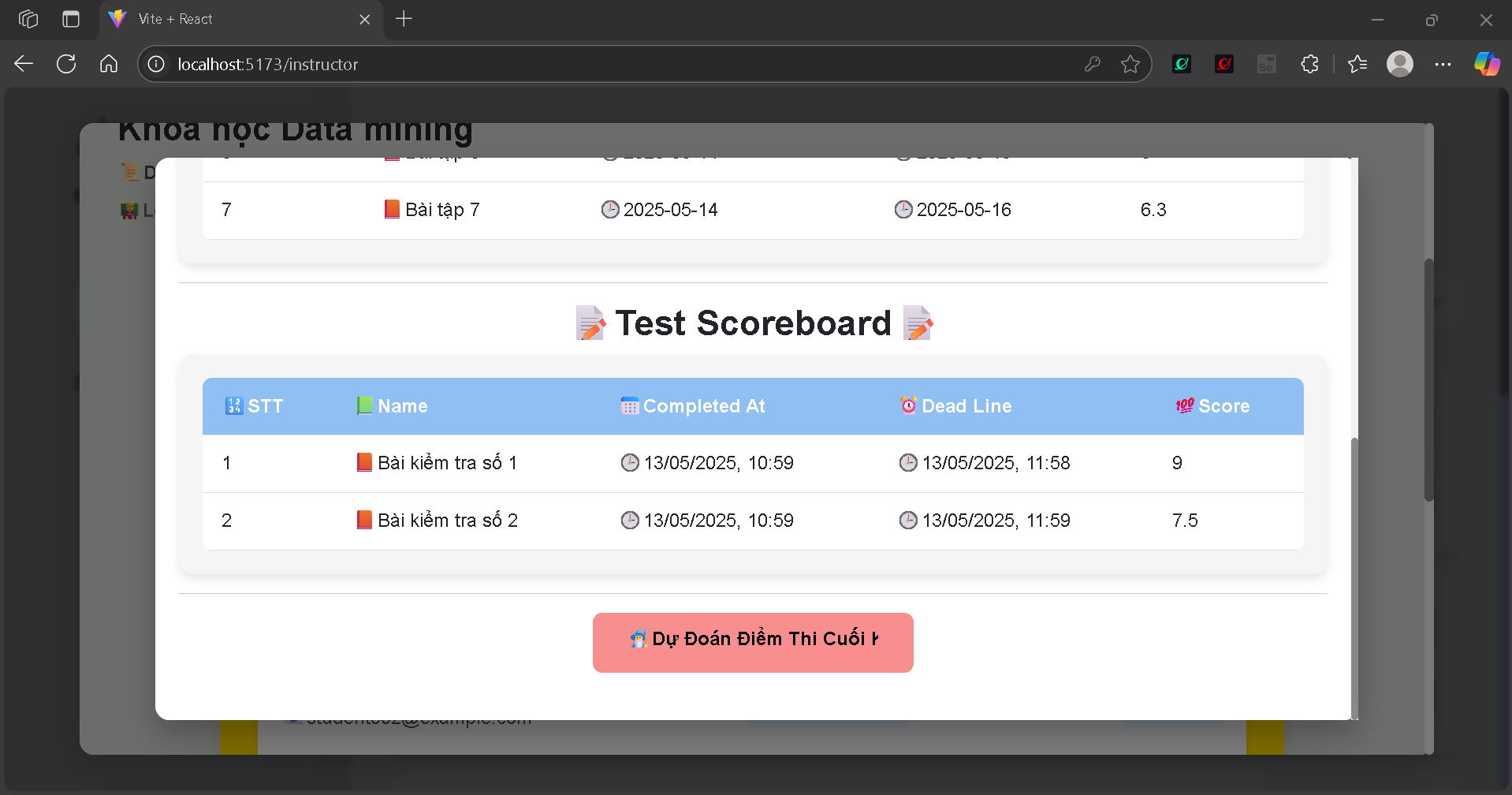
Đảm bảo tất cả các chức năng của hệ thống hoạt động chính xác:

* **Kiểm tra form nhập liệu và xác thực dữ liệu:**  
  Đảm bảo rằng tất cả các trường trong form đều có điều kiện kiểm tra hợp lệ, bao gồm kiểm tra định dạng, độ dài, giá trị ràng buộc và các trường bắt buộc. Đồng thời, xác minh các thông báo lỗi hiển thị đúng và rõ ràng khi người dùng nhập sai dữ liệu.
* **Kiểm tra kết nối giữa giao diện và API:**  
  Xác minh các yêu cầu từ giao diện (frontend) được gửi đúng đến API (backend) và phản hồi nhận lại được xử lý chính xác. Kiểm tra các trạng thái như thành công, lỗi, và lỗi kết nối để đảm bảo giao tiếp giữa client-server luôn ổn định.
* **Kiểm tra quá trình dự đoán và hiển thị kết quả:**  
  Đảm bảo rằng khi người dùng gửi dữ liệu đầu vào, hệ thống xử lý dự đoán (ví dụ: AI/ML model hoặc tính toán logic) được thực thi đúng cách và kết quả hiển thị chính xác, dễ hiểu cho người dùng. Kiểm thử cả kết quả đúng và trường hợp ngoại lệ.
* **Kiểm tra luồng người dùng tổng thể (end-to-end):**  
  Mô phỏng toàn bộ quá trình sử dụng hệ thống từ đăng nhập, thao tác chức năng chính, đến nhận kết quả và đăng xuất để đảm bảo tính mạch lạc và không có lỗi logic xảy ra.
* **Kiểm tra hiệu năng và tốc độ phản hồi:**  
  Đo thời gian phản hồi của hệ thống trong các bước chính để đảm bảo hiệu suất sử dụng mượt mà, không bị gián đoạn do trễ hoặc quá tải.
* **Kiểm tra tính tương thích trình duyệt và thiết bị:**  
  Đảm bảo giao diện hoạt động tốt trên các trình duyệt phổ biến (Chrome, Firefox, Edge, Safari) và các thiết bị khác nhau (PC, tablet, smartphone).

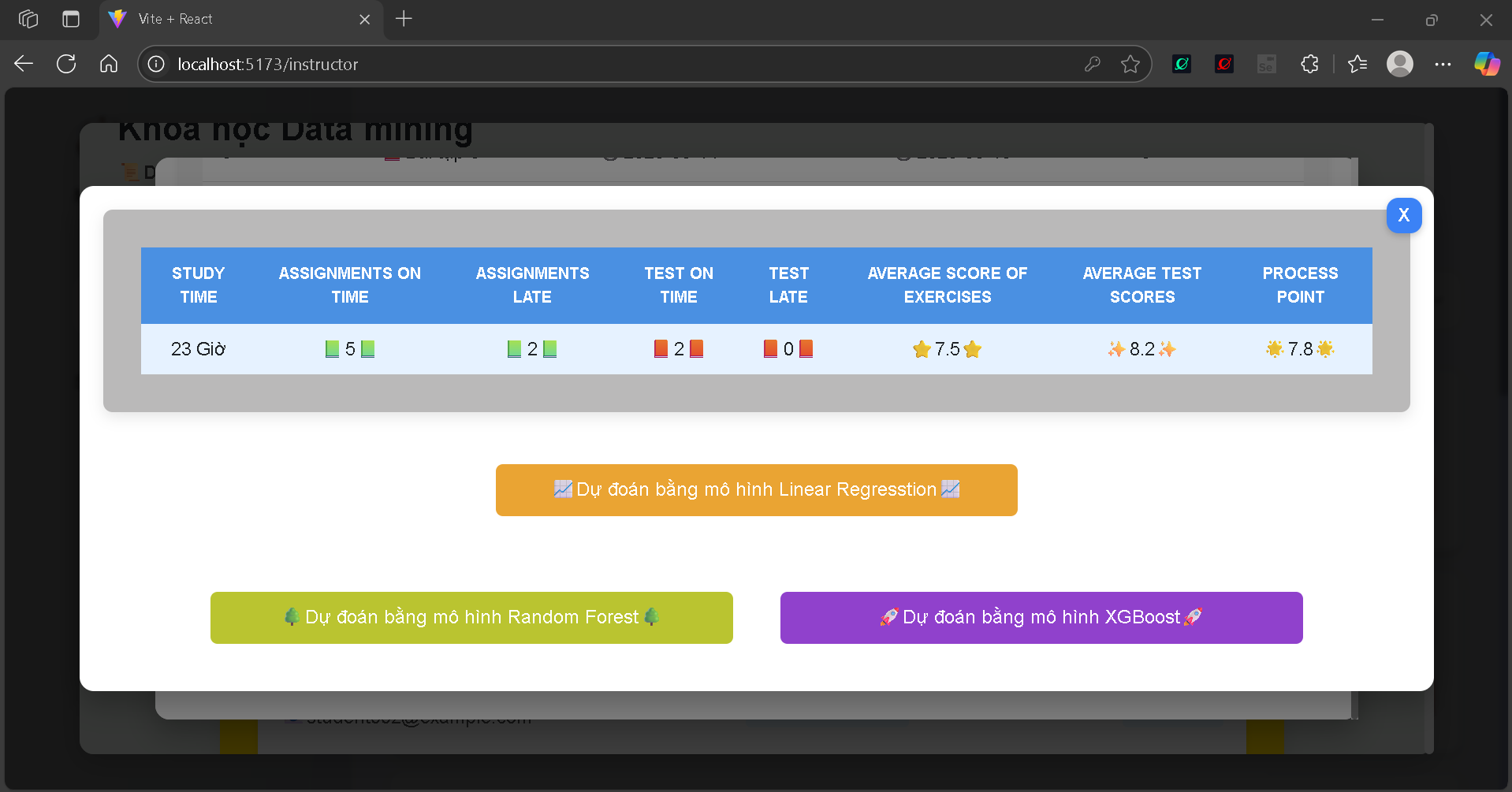
**5. Một số hình ảnh minh hoạ cho việc dự đoán điểm số**

****

Phần 7-Hình 5. 1. Giao diện xem bảng điểm của học sinh

****

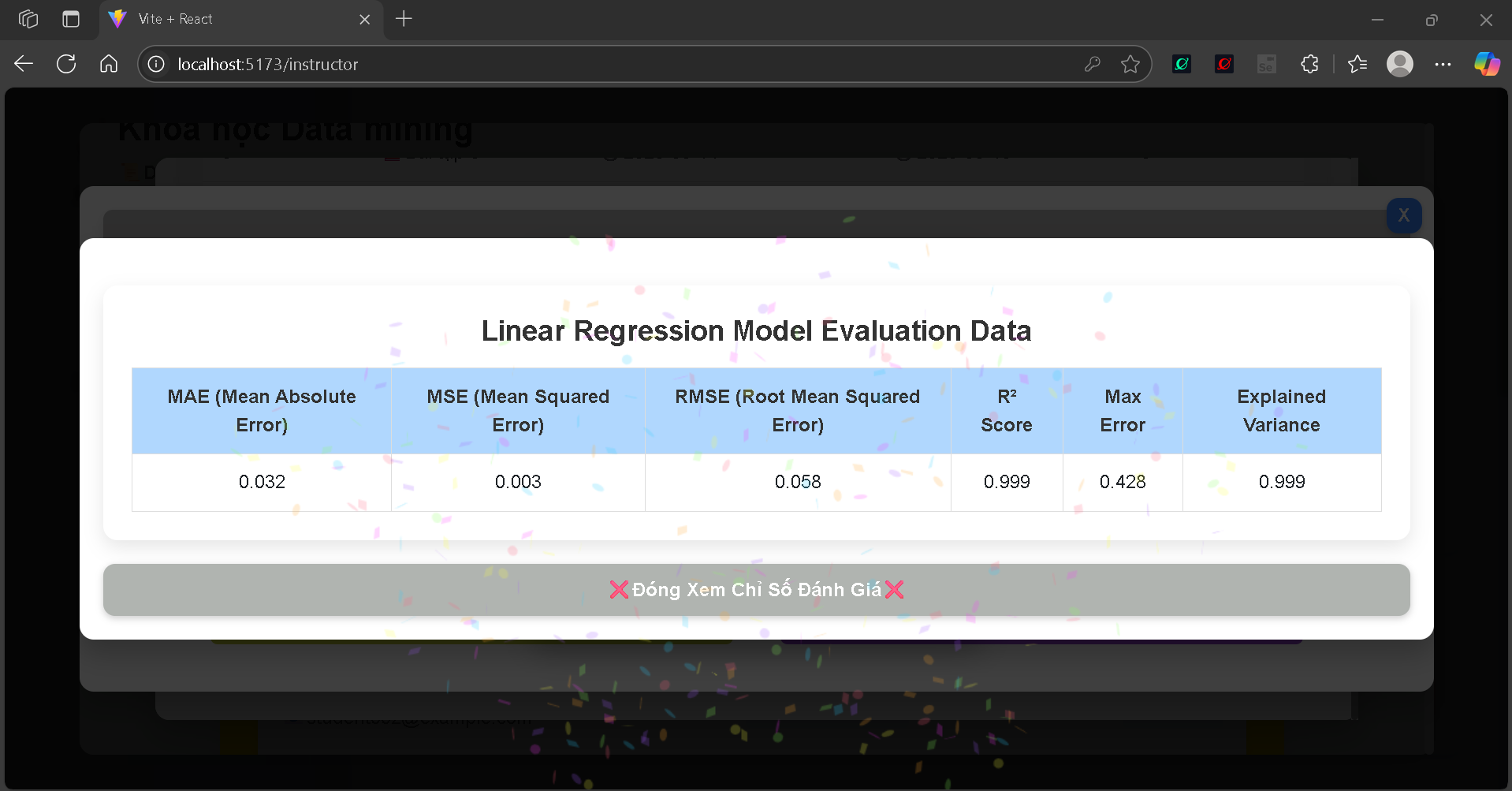
Phần 7-Hình 5. 2. Giao diện xem các bài kiểm tra của học sinh

****

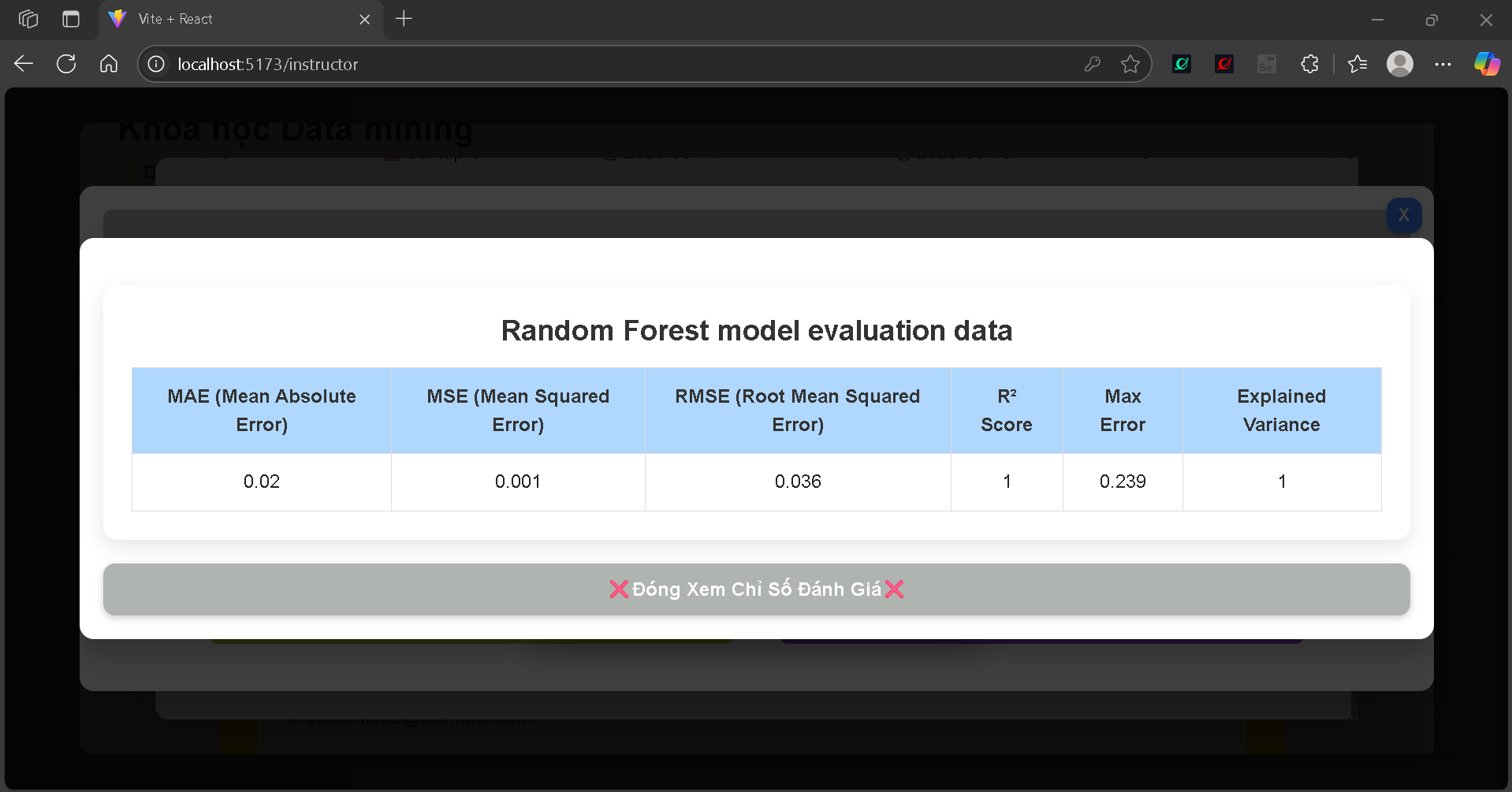
Phần 7-Hình 5. 3. Minh hoạ giao diện chọn 1 trong 3 mô hình dự đoán

****

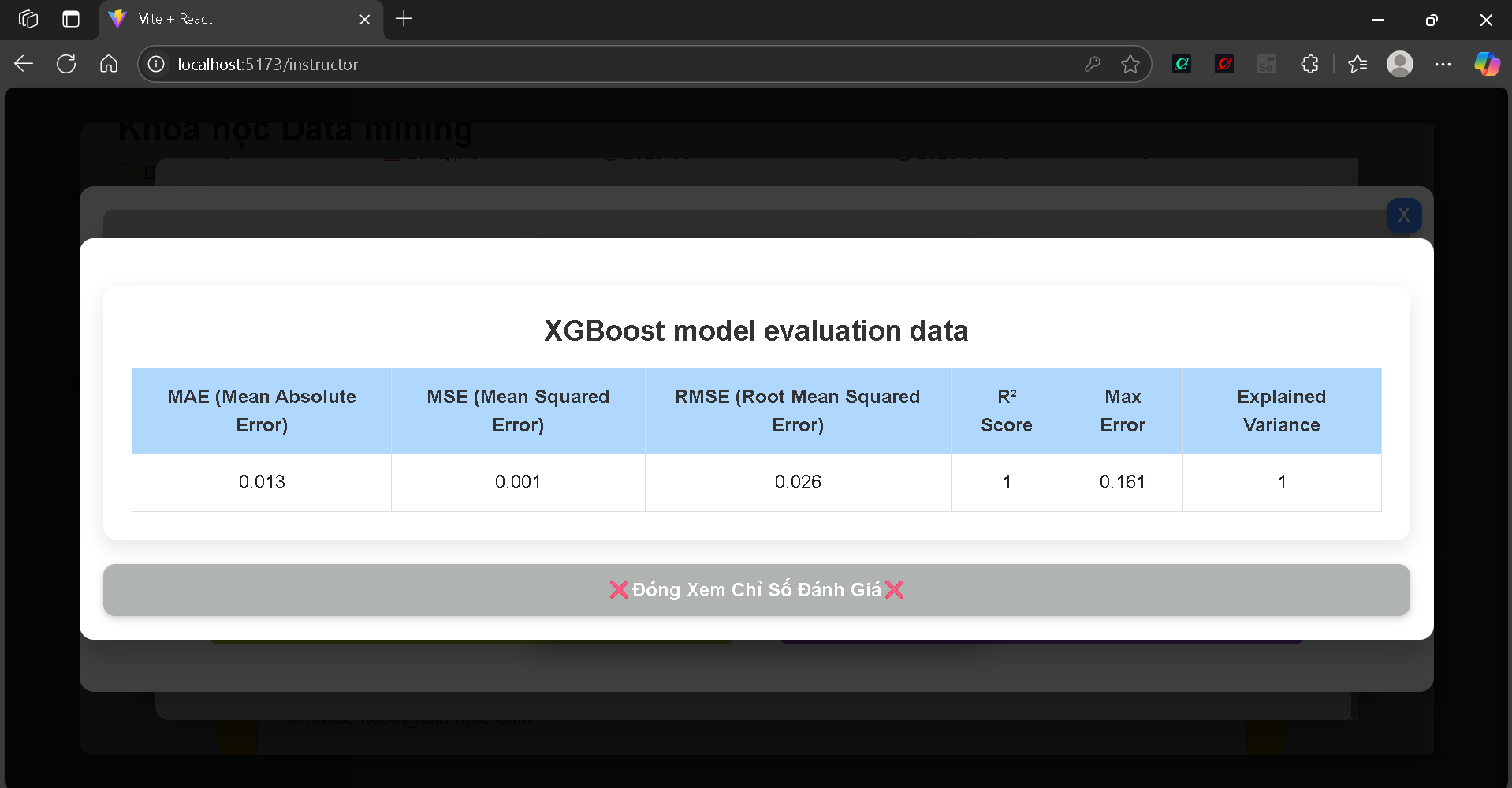
Phần 7-Hình 5. 4. Minh hoạ kết quả dự đoán

****

Phần 7-Hình 5. 5. Minh hoạ các chỉ số đánh giá của mô hình Linear Regression

****

Phần 7-Hình 5. 6. Minh hoạ các chỉ số đánh giá của mô hình Random Forest

****

Phần 7-Hình 5. 7. Minh hoạ các chỉ số đánh giá của mô hình XGBoost

**6. Kết luận sơ bộ**

Bài toán dự đoán điểm số học sinh đã được giải quyết bằng ba phương pháp học máy: Hồi quy tuyến tính, Random Forest và XGBoost. Mỗi phương pháp đều có những ưu điểm và nhược điểm riêng, phù hợp với các dạng bài toán khác nhau.

Dựa trên phân tích, XGBoost và Random Forest có thể hoạt động tốt hơn trong bài toán này do khả năng nắm bắt các mối quan hệ phức tạp và phi tuyến tính giữa các đặc trưng và biến mục tiêu. Tuy nhiên, hồi quy tuyến tính vẫn có thể là một lựa chọn tốt do tính đơn giản và hiệu quả của nó, đặc biệt là khi mối quan hệ giữa ProcessScore, StudyTime và PredictedScore có dạng tuyến tính.

Để cải thiện hiệu suất của các mô hình, có thể áp dụng các kỹ thuật như chuẩn hóa dữ liệu, chọn đặc trưng, tinh chỉnh tham số và sử dụng các kỹ thuật tổng hợp như stacking và bagging.

Cuối cùng, việc phân tích độ quan trọng của đặc trưng có thể giúp hiểu rõ hơn về mối quan hệ giữa các đặc trưng và biến mục tiêu, từ đó có thể đưa ra các quyết định tốt hơn trong việc cải thiện quá trình học tập của học sinh.

# KẾT LUẬN VÀ ĐỊNH HƯỚNG PHÁT TRIỂN

Qua quá trình xây dựng mô hình hồi quy để dự đoán điểm số của học sinh dựa trên các yếu tố như thời gian học, số lượng bài tập hoàn thành và điểm kiểm tra, ta nhận thấy rằng việc ứng dụng các thuật toán học máy trong lĩnh vực giáo dục mang lại nhiều tiềm năng thực tiễn. Mô hình hồi quy giúp xác định được mức độ ảnh hưởng của từng yếu tố đầu vào đến kết quả học tập, từ đó cung cấp cơ sở dữ liệu định lượng hỗ trợ giáo viên và học sinh trong việc đưa ra các chiến lược học tập phù hợp.

Kết quả huấn luyện và kiểm thử mô hình cho thấy độ chính xác của mô hình đạt mức khá cao khi các dữ liệu đầu vào được chuẩn hóa và thu thập đầy đủ. Thời gian học và điểm kiểm tra thường là những yếu tố có trọng số lớn, trong khi số lượng bài tập hoàn thành cũng góp phần đáng kể vào việc cải thiện kết quả học tập. Những phát hiện này không chỉ hữu ích trong việc dự đoán điểm số mà còn giúp các nhà quản lý giáo dục đưa ra những chính sách hỗ trợ học sinh một cách hiệu quả hơn.

Tuy nhiên, để mô hình có tính ứng dụng rộng rãi và linh hoạt hơn, cần phát triển theo nhiều hướng trong tương lai. Trước hết, cần mở rộng tập dữ liệu huấn luyện để mô hình có thể học từ đa dạng tình huống và đối tượng người học khác nhau. Thứ hai, việc tích hợp các yếu tố ảnh hưởng khác như mức độ tương tác trong lớp học, khả năng tiếp thu kiến thức, trạng thái tinh thần, hay hoàn cảnh gia đình cũng sẽ làm cho mô hình trở nên toàn diện hơn. Ngoài ra, nên nghiên cứu áp dụng các mô hình học máy khác như hồi quy phi tuyến, cây quyết định, hoặc mạng nơ-ron nhân tạo để so sánh và chọn lựa mô hình có hiệu suất cao nhất.

Về mặt ứng dụng, mô hình có thể được tích hợp vào hệ thống quản lý học tập trực tuyến (LMS), cho phép giáo viên theo dõi và đưa ra dự báo về kết quả học tập theo thời gian thực. Đồng thời, học sinh cũng có thể sử dụng công cụ này để tự đánh giá quá trình học và điều chỉnh phương pháp học tập cá nhân. Nếu được triển khai hiệu quả, đây có thể trở thành một công cụ hỗ trợ học tập thông minh trong bối cảnh chuyển đổi số giáo dục ngày càng mạnh mẽ hiện nay.

# TÀI LIỆU THAM KHẢO

1. *James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning: with Applications in R. Springer.*
2. *Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (2nd Edition). O'Reilly Media.*
3. *Han, J., Kamber, M., & Pei, J. (2011). Data Mining: Concepts and Techniques (3rd Edition). Elsevier.*
4. *Scikit-learn Documentation. Available at: https://scikit-learn.org/stable/documentation.html*
5. *Zhang, D. (2019). Application of Machine Learning in Education. International Journal of Emerging Technologies in Learning (iJET), 14(2), 4–10.*
6. *Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. Science, 349(6245), 255–260.*
7. *Al-Barrak, M. A., & Al-Razgan, M. (2016). Predicting Students Final GPA Using Decision Trees: A Case Study. International Journal of Information and Education Technology, 6(7), 528–533.*
8. *Yadav, S. K., & Pal, S. (2012). Data Mining: A Prediction for Performance Improvement of Engineering Students using Classification. World of Computer Science and Information Technology Journal, 2(2), 51–56.*
9. *Nguyen, H. (2020). Ứng dụng học máy trong giáo dục: Tiềm năng và thách thức. Tạp chí Khoa học Giáo dục, 12(4), 25–32.*
10. *Kaggle – Student Performance Data Sets. Available at:* [*https://www.kaggle.com/*](https://www.kaggle.com/)

# PHỤ LỤC

**PHỤ LỤC 1. DANH MỤC CÁC HÌNH VẼ VÀ BẢNG**

[Bảng 1. Lịch làm việc nhóm hàng tuần 1](#_Toc198237402)

[Bảng 2. Công việc mỗi thành viên trong nhóm 2](#_Toc198237403)

[Phần 5a-Bảng 5.2. 1. Mô tả dữ liệu 7](#_Toc198237425)

[Phần 5b-Hình 2.1. 1. Thống kê tổng quan 11](#_Toc198237432)

[Phần 5b-Hình 2.3. 1. Ma trận tương quan Pearson 1 12](#_Toc198237436)

[Phần 5b-Hình 2.3. 2. Ma trận tương quan Pearson 2 13](#_Toc198237437)

[Phần 5b-Hình 2.3. 3. Ma trận tương quan Pearson 3 13](#_Toc198237438)

[Phần 5b-Hình 4.1. 1. Mô hình đường dẫn 16](#_Toc198237441)

[Phần 6a-Bảng 5.1. 1. Bảng so sánh 3 mô hình 35](#_Toc198237450)

[Phần 6b-Hình 2.1. 1. Công thức tính toán thời gian học 37](#_Toc198237458)

[Phần 6b-Hình 2.1. 2. Hàm tính điểm trung bình các bài tập và bài kiểm tra 37](#_Toc198237459)

[Phần 6b-Hình 2.1. 3. Công thức tính điểm quá trình 37](#_Toc198237460)

[Phần 6b-Hình 2.1. 4. Hàm tính điểm dự đoán 38](#_Toc198237461)

[Phần 6b-Hình 4.1. 1. Tiền xử lý dữ liệu 39](#_Toc198295368)

[Phần 6b-Hình 4.2. 1. Huấn luyện mô hình Linear Regression 40](#_Toc198295382)

[Phần 6b-Hình 4.2. 2. Huấn luyện mô hình Random Forest 40](#_Toc198295383)

[Phần 6b-Hình 4.2. 3. Huấn luyện mô hình XGBoost 40](#_Toc198295384)

[Phần 6b-Hình 4.3. 1. Các chỉ số đánh giá mô hình 41](#_Toc198295391)

[Phần 6b-Hình 4.3. 2. Vẽ đồ thị so sánh giá trị thực và giá trị dự đoán 41](#_Toc198295392)

[Phần 6b-Hình 4.4. 1. Lưu mô hình 41](#_Toc198295402)

[Phần 6b-Hình 6.1.1. 1. Đề xuất cải tiến dữ liệu bằng việc chuẩn hoá dữ liệu 43](#_Toc198295412)

[Phần 6b-Hình 6.1.2. 1. Đề xuất cải tiến dữ liệu bằng việc phân tích tương quan 43](#_Toc198295426)

[Phần 6b-Hình 6.1.3. 1. Đề xuất cải tiến bằng việc chọn đặc trưng 44](#_Toc198295433)

[Phần 6b-Hình 6.2.1. 1. Tinh chỉnh tham số GridSearchCV 44](#_Toc198295453)

[Phần 6b-Hình 6.2.2. 1. Tinh chỉnh tham số RandomizedSearchCV 45](#_Toc198295466)

[Phần 6b-Hình 6.3.1. 1. Kỹ thuật tổng hợp Stacking 45](#_Toc198295473)

[Phần 6b-Hình 6.3.2. 1. Kỹ thuật tổng hợp Bagging 46](#_Toc198295481)

[Phần 6b-Hình 6.4. 1. Phân tích độ quan trọng của đặc trưng 46](#_Toc198295490)

[Phần 7-Hình 1.1. 1.Cài đặt các thư viện cần thiết 47](#_Toc198295497)

[Phần 7-Hình 1.2. 1. Cấu trúc thư mục dự án 48](#_Toc198295504)

[Phần 7-Hình 2. 1. Tạo dữ liệu với bot.py 49](#_Toc198295508)

[Phần 7-Hình 2. 2. Xuất dữ liệu thành file CSV 49](#_Toc198295509)

[Phàn 7-Hình 3.1. 1. Thiết kế API cho 3 mô hình dự đoán 50](#_Toc198295512)

[Phần 7-Hình 3.2. 1. Thiết kế giao diện đầu vào 51](#_Toc198295518)

[Phần 7-Hình 3.2. 2. Thiết kế giao diện kết quả 52](#_Toc198295519)

[Phần 7-Hình 5. 1. Giao diện xem bảng điểm của học sinh 47](#_Toc198237682)

[Phần 7-Hình 5. 2. Giao diện xem các bài kiểm tra của học sinh 47](#_Toc198237683)

[Phần 7-Hình 5. 3. Minh hoạ giao diện chọn 1 trong 3 mô hình dự đoán 48](#_Toc198237684)

[Phần 7-Hình 5. 4. Minh hoạ kết quả dự đoán 48](#_Toc198237685)

[Phần 7-Hình 5. 5. Minh hoạ các chỉ số đánh giá của mô hình Linear Regression 49](#_Toc198237686)

[Phần 7-Hình 5. 6. Minh hoạ các chỉ số đánh giá của mô hình Random Forest 49](#_Toc198237687)

[Phần 7-Hình 5. 7. Minh hoạ các chỉ số đánh giá của mô hình XGBoost 50](#_Toc198237688)

**PHỤ LỤC 2. BIÊN BẢN HỌP NHÓM**

**CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**Độc lập – Tự do – Hạnh phúc**

**BIÊN BẢN HỌP NHÓM**

(*V/v Phân công công việc /Đánh giá hoàn thành /Họp nhóm định kỳ....*)

**1. Thời gian, địa điểm, thành phần tham dự.**

**1.1. Thời gian:** 20giờ ngày 10 tháng 5 năm 2025

**1.2. Địa điểm: Phòng Zoom**

**1.3. Thành phần tham dự:**

+ Chủ trì: Lê Tấn Phú

+ Tham dự: Thành viên nhóm

+ Vắng: Không

**2. Nội dung cuộc họp**

**2.1. Nhóm trưởng phân công công việc cho các thành viên như sau:**

* 1. **Ý kiến của các thành viên:**

|  |  |
| --- | --- |
| **HỌ TÊN** | **Ý Kiến Thành Viên** |
| - Lê Tấn Phú | - Có đóng góp ý kiến  - Đồng ý với ý kiến của các bạn đưa ra |
| - Trần Gia Huy | -Có đóng góp ý kiến  - Đồng ý với ý kiến của nhóm trưởng |
| - Nguyễn Huyện Anh Khang | - Có đóng góp ý kiến  - Đồng ý với ý kiến của nhóm trưởng |

**2.3. Kết luận cuộc họp**

Sau khi tổng hợp tất cả các ý kiến của thành viên nhóm đã hoàn thành đề cương và phân công nhiệm vụ cần làm để hoàn thành đồ án môn học.

Cuộc họp đi đến thống nhất và kết thúc lúc giờ phút cùng ngày.

|  |  |
| --- | --- |
|  | **Chủ trì**  **( Ký và ghi rõ họ tên)**  **LÊ TẤN PHÚ** |

|  |  |
| --- | --- |
|  |  |
|  |  |

**------HẾT------**