BỘ CÔNG THƯƠNG

TRƯỜNG ĐẠI HỌC KINH TẾ - KỸ THUẬT CÔNG NGHIỆP

KHOA KHOA HỌC ỨNG DỤNG

BÁO CÁO TỔNG KẾT
ĐỀ TÀI NGHIÊN CỨU KHOA HỌC SINH VIÊN
NĂM HỌC 2024 – 2025

Tên đề tài:

SỬ DỤNG CÁC MÔ HÌNH HỌC MÁY ĐỂ DỰ ĐOÁN KHẢ NĂNG PHÊ DUYỆT KHOẢN VAY

Giảng viên hướng dẫn: **Lê Hằng Anh**

Chủ nhiệm đề tài: **Trần Đức Lương** **Lớp DHKL16A2HN**

Thành viên: **Hồ Thị Minh Hằng** **Lớp DHKL16A2HN**

Phạm Thị Ngọc Tú **Lớp DHKL16A2HN**

Nguyễn Ngọc Bắc **Lớp DHKL16A2HN**

HÀ NỘI 04/2025

BỘ CÔNG THƯƠNG

TRƯỜNG ĐẠI HỌC KINH TẾ - KỸ THUẬT CÔNG NGHIỆP

KHOA KHOA HỌC ỨNG DỤNG

BÁO CÁO TỔNG KẾT
ĐỀ TÀI NGHIÊN CỨU KHOA HỌC SINH VIÊN
NĂM 2024 – 2025

Tên đề tài:

SỬ DỤNG CÁC MÔ HÌNH HỌC MÁY ĐỂ DỰ ĐOÁN KHẢ NĂNG PHÊ DUYỆT KHOẢN VAY

Giảng viên hướng dẫn: **Lê Hằng Anh**

Chủ nhiệm đề tài:	**Trần Đức Lương**	**Lớp DHKL16A2HN**
Thành viên:	**Hồ Thị Minh Hằng**	**Lớp DHKL16A2HN**
	Phạm Thị Ngọc Tú	**Lớp DHKL16A2HN**
	Nguyễn Ngọc Bắc	**Lớp DHKL16A2HN**

HÀ NỘI 04/2025

DANH SÁCH NHỮNG NGƯỜI THỰC HIỆN, GIẢNG VIÊN HƯỚNG DẪN

Sinh viên thực hiện:

STT	Họ và tên	Lớp	Khoa	Chức danh
1	Hồ Thị Minh Hằng	DHKL16A2HN	Khoa học ứng dụng	Chủ nhiệm đề tài
2	Trần Đức Lương	DHKL16A2HN	Khoa học ứng dụng	Thành viên
3	Phạm Thị Ngọc Tú	DHKL16A2HN	Khoa học ứng dụng	Thành viên
4	Nguyễn Ngọc Bắc	DHKL16A2HN	Khoa học ứng dụng	Thành viên

Giảng viên hướng dẫn: Lê Hằng Anh

- Học vị: Cử nhân Chức vụ: Giảng viên
- Điện thoại: 083 455 1996 Email: lhanh@uneti.edu.vn

MỤC LỤC

DANH MỤC BIỂU, BẢNG

DANH MỤC HÌNH VẼ

CHƯƠNG 1: MỞ ĐẦU

1.1. Tính cấp thiết của đề tài

Trong bối cảnh nền kinh tế phát triển mạnh mẽ, nhu cầu vay vốn của cá nhân và doanh nghiệp ngày càng tăng cao. Các tổ chức tài chính, ngân hàng cần một phương pháp đánh giá hiệu quả, nhanh chóng và chính xác để ra quyết định về việc cấp hay từ chối khoản vay. Tuy nhiên, việc đánh giá truyền thống dựa chủ yếu vào sự xem xét thủ công của nhân viên tín dụng tiềm ẩn nhiều rủi ro chủ quan, mất thời gian và thiếu nhất quán.

Sự phát triển vượt bậc của trí tuệ nhân tạo, đặc biệt là học máy, đã mở ra cơ hội ứng dụng các thuật toán thông minh vào quy trình phân tích và đánh giá hồ sơ vay vốn. Việc xây dựng một mô hình dự đoán khả năng được chấp nhận vay dựa trên dữ liệu lịch sử là bước tiến quan trọng nhằm nâng cao hiệu quả ra quyết định, giảm thiểu rủi ro tín dụng và tối ưu hóa hoạt động của các tổ chức tài chính.

1.2. Tình hình nghiên cứu đề tài

Trong những năm gần đây, cả ở trong nước lẫn quốc tế, các nghiên cứu đã tích cực triển khai các thuật toán học máy tiên tiến như Logistic Regression, Decision Tree, Random Forest, SVM, XGBoost,... để giải quyết những bài toán phân loại phức tạp trong lĩnh vực tài chính. Kết quả từ các nghiên cứu này cho thấy rằng học máy không chỉ có khả năng khai thác dữ liệu lịch sử mà còn vượt trội hơn các phương pháp truyền thống về độ chính xác trong dự đoán. Tuy nhiên, mỗi thuật toán đều sở hữu những thế mạnh và hạn chế riêng, điều này đặt ra thách thức trong việc lựa chọn mô hình tối ưu nhất cho từng bộ dữ liệu cụ thể. Đây vẫn là một vấn đề mở cần được nghiên cứu chuyên sâu để tận dụng triệt để tiềm năng của học máy trong lĩnh vực tài chính.

1.3. Mục đích và nhiệm vụ nghiên cứu

Mục đích: Xây dựng một mô hình học máy có khả năng dự đoán khả năng được phê duyệt khoản vay của khách hàng dựa trên các thông tin đầu vào như thu nhập, độ tuổi, lịch sử tín dụng, v.v.

Nhiệm vụ:

- Thu thập và tiền xử lý dữ liệu phù hợp với mục tiêu nghiên cứu.
- Xây dựng và huấn luyện các mô hình học máy như Logistic Regression, Random Forest, XGBoost và SVM.
- So sánh hiệu suất của các mô hình thông qua các chỉ số đánh giá (accuracy, precision, recall, F1-score...).

- Đưa ra khuyến nghị về mô hình tối ưu và khả năng ứng dụng thực tiễn trong ngành tài chính.

1.4. Đối tượng và phạm vi nghiên cứu

Về đối tượng nghiên cứu, đề tài tập trung vào các khách hàng có nhu cầu vay vốn từ các tổ chức tài chính như ngân hàng hay công ty tài chính. Nghiên cứu sẽ đi sâu vào việc phân tích dữ liệu của những khách hàng này nhằm xây dựng các mô hình dự đoán khả năng được phê duyệt khoản vay, dựa trên nhiều yếu tố đầu vào như thu nhập, độ tuổi, lịch sử tín dụng và các thông số liên quan khác.

Về phương pháp học máy, nghiên cứu sẽ xây dựng và tiến hành so sánh hiệu quả của nhiều mô hình khác nhau bao gồm Logistic Regression, Decision Tree, Random Forest, SVM và XGBoost. Các mô hình này sẽ được đánh giá một cách toàn diện thông qua các chỉ số đo lường hiệu suất như accuracy (độ chính xác), precision (độ chuẩn xác), recall (độ bao phủ) và F1-score nhằm xác định mô hình tối ưu nhất cho bài toán dự đoán.

Về quy trình nghiên cứu, đề tài sẽ thực hiện một chuỗi các hoạt động từ thu thập và tiền xử lý dữ liệu phù hợp với mục tiêu đề ra, tiến hành huấn luyện các mô hình học máy đã chọn, sau đó tiến hành so sánh, đánh giá để lựa chọn mô hình có hiệu suất tốt nhất. Cuối cùng, nghiên cứu sẽ đưa ra những khuyến nghị cụ thể về khả năng ứng dụng thực tiễn của mô hình được chọn trong ngành tài chính.

Về phạm vi ứng dụng, kết quả của nghiên cứu hướng đến việc cải thiện quy trình đánh giá tính năng tín dụng, giúp các tổ chức tài chính giảm thiểu rủi ro trong hoạt động cho vay và tối ưu hóa công tác quản lý tín dụng. Qua đó, nghiên cứu góp phần nâng cao hiệu quả ra quyết định, tạo ra một phương pháp đánh giá nhanh chóng, chính xác và nhất quán hơn so với phương pháp đánh giá thủ công truyền thống.

1.5. Cách tiếp cận và phương pháp nghiên cứu

Cách tiếp cận: Tiếp cận theo hướng xây dựng mô hình học máy từ dữ liệu thực tế, đánh giá hiệu quả mô hình và đề xuất giải pháp áp dụng.

Phương pháp nghiên cứu:

Phân tích dữ liệu: Đầu tiên, đề tài sẽ thực hiện phân tích tổng quan bộ dữ liệu, bao gồm thống kê số lượng quan sát, phân bố các biến quan trọng và mối quan hệ giữa các đặc trưng đầu vào.

Tiền xử lý dữ liệu: Tiếp theo, đề tài sẽ tiến hành tiền xử lý dữ liệu, bao gồm xử lý giá trị thiếu, mã hóa biến định tính và tạo đặc trưng mới khi cần thiết.

Trực quan hóa dữ liệu: Sử dụng các biểu đồ và đồ thị để trực quan hóa dữ liệu là một phần quan trọng trong việc phân tích và hiểu sâu hơn về mối quan hệ giữa các biến và khả năng phê duyệt khoản vay.

Xây dựng và đánh giá mô hình: Đề tài dự kiến sử dụng một số thuật toán học máy như Logistic Regression, Random Forest, và XGBoost. Việc xây dựng mô hình sẽ bao gồm huấn luyện mô hình trên dữ liệu huấn luyện và đánh giá hiệu suất của mỗi mô hình bằng các chỉ số như độ chính xác, độ nhạy, độ đặc hiệu, và AUC-ROC.

So sánh và phân tích kết quả: Sau khi có các mô hình dự đoán, đề tài sẽ tiến hành so sánh hiệu suất của chúng để xác định mô hình có độ chính xác cao nhất và thích hợp nhất cho mục đích dự đoán khả năng phê duyệt khoản vay.

Thảo luận và đánh giá kết quả: Phần này sẽ phân tích ưu nhược điểm từng mô hình, thảo luận việc kết hợp dữ liệu định tính – định lượng nhằm nâng cao độ chính xác, và đánh giá khả năng ứng dụng thực tế của hệ thống.

1.6. Kết cấu của đề tài

Đề tài được chia thành năm chương chính, sắp xếp theo trình tự logic để giải quyết vấn đề nghiên cứu một cách mạch lạc:

Chương 1: Mở đầu - Trình bày bối cảnh, tính cấp thiết của vấn đề nghiên cứu, tổng quan tình hình nghiên cứu, mục tiêu và nhiệm vụ, đối tượng – phạm vi nghiên cứu, phương pháp nghiên cứu và cấu trúc của đề tài.

Chương 2: Cơ sở lý thuyết - Giới thiệu tổng quan về học máy, các thuật toán được sử dụng trong nghiên cứu như Logistic Regression, Random Forest, Gradient Boosting và SVM. Trình bày các chỉ số đánh giá hiệu suất mô hình và tổng quan về bài toán phê duyệt khoản vay trong lĩnh vực tài chính.

Chương 3: Thực nghiệm - Mô tả bộ dữ liệu, quá trình tiền xử lý, chuẩn hóa và loại bỏ dữ liệu nhiễu. Tiến hành xây dựng, huấn luyện mô hình và điều chỉnh siêu tham số nhằm tối ưu hiệu suất.

Chương 4: Kết quả và thảo luận - Trình bày trực quan hóa dữ liệu, kết quả từ các mô hình đã xây dựng, so sánh hiệu suất mô hình và phân tích độ quan trọng của các đặc trưng. Thảo luận về ưu – nhược điểm và khả năng áp dụng thực tiễn của mô hình.

Chương 5: Kết luận và hướng phát triển - Tóm tắt kết quả nghiên cứu, nêu rõ những đóng góp, các hạn chế còn tồn tại và đề xuất những hướng nghiên cứu tiếp theo trong tương lai.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Tổng quan về học máy

Học máy (machine learning) là một lĩnh vực thuộc trí tuệ nhân tạo (AI), cho phép hệ thống học hỏi và cải thiện hiệu suất dựa trên dữ liệu và kinh nghiệm, mà không cần phải được lập trình rõ ràng cho từng nhiệm vụ cụ thể.

2.1.1. Khái niệm và phân loại

Khái niệm: Học máy tập trung vào việc xây dựng các thuật toán và mô hình giúp máy tính phân tích và dự đoán dựa trên dữ liệu. Mục tiêu chính là để hệ thống tự động nhận diện các mẫu hoặc xu hướng và sử dụng chúng để đưa ra quyết định hoặc dự đoán.

Học máy thường được chia thành các loại chính sau:

- Học có giám sát (Supervised Learning): Máy học từ dữ liệu đầu vào và đầu ra đã biết để dự đoán kết quả cho dữ liệu chưa biết. Ví dụ: phân loại hình ảnh, dự đoán giá nhà.
- Học không giám sát (Unsupervised Learning): Máy học từ dữ liệu đầu vào mà không có nhãn hoặc hướng dẫn. Ví dụ: phân cụm dữ liệu, giảm kích thước dữ liệu.
- Học bán giám sát (Semi-Supervised Learning): Kết hợp giữa học có giám sát và không giám sát, thường được dùng khi dữ liệu nhãn chỉ có một phần và không đầy đủ.
- Học tăng cường (Reinforcement Learning): Máy học thông qua việc tương tác với môi trường và nhận phần thưởng hoặc hình phạt. Đây là cách học thường áp dụng trong robot hoặc các trò chơi.

2.1.2. Các loại thuật toán trong học máy

Thuật toán trong học máy là công cụ cốt lõi giúp mô hình học từ dữ liệu và đưa ra dự đoán hoặc quyết định. Tùy theo phương pháp học, các thuật toán được chia thành các nhóm chính sau:

- Thuật toán học có giám sát: Áp dụng khi dữ liệu có nhãn. Một số thuật toán tiêu biểu gồm Hồi quy tuyến tính, Hồi quy logistic, Cây quyết định, Random Forest, SVM, và KNN.
- Thuật toán học không giám sát: Dùng cho dữ liệu không có nhãn, nhằm khám phá cấu trúc ẩn. Các thuật toán phổ biến gồm K-Means, Hierarchical Clustering, PCA và DBSCAN.
- Thuật toán học tăng cường: Dựa trên việc học thông qua tương tác với môi trường để tối đa hóa phần thưởng. Các thuật toán gồm Q-Learning, SARSA, và Deep Q-Network.

Ngoài ra, còn có các phương pháp kết hợp như học bán giám sát, học chuyển giao và học đa tác vụ, nhằm tối ưu hiệu quả trong những trường hợp dữ liệu hạn chế hoặc nhiệm vụ phức tạp.

2.2. Các mô hình học máy được sử dụng trong nghiên cứu

Để giải quyết bài toán dự đoán khả năng chấp nhận khoản vay của khách hàng, nghiên cứu này sử dụng một số mô hình học máy phổ biến và có hiệu quả cao trong các bài toán phân loại. Mỗi mô hình có những đặc điểm riêng về cách xây dựng, khả năng học dữ liệu và mức độ giải thích. Trong phần này, chúng tôi sẽ trình bày tổng quan về bốn mô hình được lựa chọn, bao gồm Logistic Regression, Random Forest, XGBoost và SVM.

2.2.1. Mô hình Logistic Regression

Logistic Regression là một mô hình tuyến tính được sử dụng phổ biến cho các bài toán phân loại nhị phân. Mô hình này dựa trên hàm sigmoid để ước lượng xác suất của một biến đầu ra thuộc về một lớp cụ thể. Về mặt toán học, mô hình này có dạng:

$$P(Y=1|X)=1+e−(β0+β1X1+β2X2+...+βnXn)1$$

Trong đó:

- $P(Y=1|X)$ là xác suất mà kết quả thuộc lớp 1 (được chấp nhận vay) với đầu vào X
- β là các hệ số cần được ước lượng trong quá trình huấn luyện
- X là các đặc trưng đầu vào (như thu nhập, độ tuổi, lịch sử tín dụng...)

Ưu điểm của Logistic Regression là đơn giản, dễ cài đặt và dễ diễn giải, đặc biệt phù hợp với dữ liệu có mối quan hệ tuyến tính giữa các đặc trưng và kết quả đầu ra. Mỗi hệ số β cho biết mức độ ảnh hưởng của đặc trưng tương ứng đến khả năng được chấp nhận vay, giúp các tổ chức tài chính dễ dàng giải thích quyết định cho khách hàng.

Tuy nhiên, mô hình này có nhược điểm là khó xử lý các mối quan hệ phức tạp, phi tuyến giữa các đặc trưng và đặc biệt kém hiệu quả khi dữ liệu có nhiều đặc trưng tương quan với nhau (multicollinearity).

2.2.2. Mô hình Random Forest

Random Forest là một thuật toán học máy theo kiểu ensemble, kết hợp nhiều cây quyết định để tạo ra một mô hình mạnh hơn. Mỗi cây trong rừng được huấn luyện trên một tập con của dữ liệu và đặc trưng, được chọn ngẫu nhiên bằng kỹ thuật bootstrap sampling và feature bagging. Kết quả cuối cùng là sự kết hợp (thường là trung bình) của các dự đoán từ tất cả các cây.

Quy trình xây dựng mô hình Random Forest bao gồm:

- Chọn ngẫu nhiên n mẫu từ tập dữ liệu gốc bằng kỹ thuật bootstrap

- Xây dựng cây quyết định dựa trên n mẫu này, nhưng tại mỗi nút chỉ xét một tập con ngẫu nhiên các đặc trưng
- Lặp lại bước 1 và 2 để tạo ra nhiều cây quyết định (thường từ 100 đến 500 cây)
- Đối với bài toán phân loại, kết quả dự đoán cuối cùng được xác định bằng cách bỏ phiếu đa số

Random Forest hoạt động hiệu quả trên các tập dữ liệu có nhiều đặc trưng và không đòi hỏi nhiều bước xử lý đầu vào. Mô hình này có khả năng xử lý tốt dữ liệu phi tuyến, dữ liệu nhiễu và ít bị ảnh hưởng bởi hiện tượng quá khớp (overfitting). Random Forest cũng cung cấp thông tin về tầm quan trọng của từng đặc trưng, giúp hiểu rõ hơn về các yếu tố ảnh hưởng đến quyết định cấp vay.

Nhược điểm của Random Forest là quá trình huấn luyện có thể chậm trên dữ liệu lớn và khó diễn giải chi tiết hơn so với các mô hình đơn giản như Logistic Regression.

2.2.3. Mô hình XGBoost

XGBoost (Extreme Gradient Boosting) là một biến thể nâng cao của thuật toán Gradient Boosting, được thiết kế để tối ưu về tốc độ và hiệu suất. Thuật toán này hoạt động theo nguyên tắc ensemble, xây dựng các mô hình yếu (weak learners) một cách tuần tự, mỗi mô hình mới tập trung vào việc sửa chữa lỗi của các mô hình trước đó.

Các cải tiến chính của XGBoost so với Gradient Boosting truyền thống bao gồm:

Tối ưu hóa đạo hàm bậc hai: XGBoost sử dụng cả đạo hàm bậc nhất và bậc hai của hàm mất mát để cải thiện tốc độ hội tụ và độ chính xác.

Regularization hiệu quả: Áp dụng cả kỹ thuật regularization L1 và L2 để kiểm soát độ phức tạp của mô hình, giảm thiểu overfitting.

Xử lý dữ liệu thiếu: XGBoost có cơ chế tự động xử lý giá trị thiếu bằng cách tìm hướng phân chia tối ưu.

Tính toán song song: Thuật toán được thiết kế để tận dụng khả năng tính toán song song, giúp tăng tốc quá trình huấn luyện.

Cắt tỉa cây (pruning): XGBoost tự động cắt tỉa các nhánh cây không cải thiện hiệu suất, giúp giảm độ phức tạp của mô hình.

XGBoost thường đạt hiệu suất dự đoán cao trên nhiều loại dữ liệu, đặc biệt là các bài toán cấu trúc như bài toán dự đoán khả năng vay vốn. Mô hình này cũng cung cấp thông tin về tầm quan trọng của các đặc trưng, giúp các tổ chức tài chính xác định những yếu tố quan trọng nhất ảnh hưởng đến quyết định cấp vay.

Tuy nhiên, XGBoost có thể tốn nhiều thời gian và tài nguyên để tinh chỉnh các tham số (hyperparameter tuning) và khó diễn giải hơn so với các mô hình đơn giản.

2.2.4. Mô hình Support Vector Machine (SVM)

SVM là phân loại mạnh mẽ, hoạt động bằng cách tìm siêu phẳng (hyperplane) tối ưu để phân tách các điểm dữ liệu thuộc các lớp khác nhau. Siêu phẳng tối ưu được xác định là siêu phẳng có khoảng cách (margin) lớn nhất đến các điểm dữ liệu gần nhất của mỗi lớp, được gọi là các support vectors.

Về mặt toán học, SVM giải quyết bài toán tối ưu:

$$\min_{w,b} \ \frac{1}{2}|w|^2 \tag{1}$$

Với điều kiện ràng buộc:

$$y_i(w^T x_i + b) \geq 1, \quad \forall i$$

Trong đó:

- w là vector trọng số của siêu phẳng
- b là hệ số điều chỉnh
- x_i là vector đặc trưng của mẫu thứ i
- y_i là nhãn của mẫu thứ i (+1 hoặc -1)

Đối với dữ liệu không thể phân tách tuyến tính, SVM sử dụng các hàm kernel để ánh xạ dữ liệu vào không gian đặc trưng cao hơn, nơi có thể tìm được siêu phẳng phân tách. Các kernel phổ biến bao gồm:

1. **Kernel tuyến tính**: K(x_i, x_j) = x_i^T x_j
2. **Kernel đa thức**: K(x_i, x_j) = (γx_i^T x_j + r)^d
3. **Kernel RBF (Radial Basis Function)**: K(x_i, x_j) = exp(-γ||x_i-x_j||^2)
4. **Kernel sigmoid**: K(x_i, x_j) = tanh(γx_i^T x_j + r)

SVM hiệu quả trong không gian đặc trưng cao và có khả năng xử lý dữ liệu phi tuyến thông qua các hàm kernel. Mô hình này ít bị ảnh hưởng bởi hiện tượng overfitting, đặc biệt khi số lượng đặc trưng lớn hơn số lượng mẫu dữ liệu.

Tuy nhiên, SVM có thể tốn thời gian huấn luyện trên tập dữ liệu lớn và việc lựa chọn kernel cùng các tham số phù hợp đòi hỏi quá trình thử nghiệm kỹ lưỡng. Ngoài ra, SVM không cung cấp trực tiếp xác suất dự đoán, điều này có thể là một hạn chế trong một số ứng dụng tài chính cần thông tin về mức độ tin cậy của dự đoán.

2.3. Đánh giá hiệu suất mô hình

Việc đánh giá hiệu suất của mô hình học máy là bước không thể thiếu trong quá trình xây dựng hệ thống dự đoán đáng tin cậy. Đặc biệt trong các bài toán phân loại nhị phân như dự đoán khả năng chấp nhận khoản vay, việc lựa chọn đúng chỉ số đánh giá giúp phản ánh chính xác năng lực của mô hình, đồng thời định hướng điều chỉnh và cải tiến phù hợp. Trong nghiên cứu này, chúng tôi sử dụng bộ tiêu chí đánh giá đa chiều bao gồm

Accuracy, Precision, Recall, F1-score và ROC – AUC. Những chỉ số này không chỉ giúp đo lường mức độ chính xác của dự đoán, mà còn thể hiện sự cân bằng giữa việc phát hiện đúng và tránh dự đoán sai trong bối cảnh dữ liệu có thể mất cân đối giữa các lớp.

2.3.1. Accuracy (Độ chính xác)

Accuracy là một trong những thước đo cơ bản và phổ biến nhất trong đánh giá hiệu suất của các mô hình phân loại. Độ chính xác được định nghĩa là tỷ lệ giữa số lượng dự đoán đúng trên tổng số dự đoán. Cụ thể, công thức tính accuracy được biểu diễn như sau:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

Trong đó:

- TP (True Positive) là số lượng mẫu dương được dự đoán đúng.
- TN (True Negative) là số lượng mẫu âm được dự đoán đúng.
- FP (False Positive) là số lượng mẫu âm bị dự đoán nhầm là dương.
- FN (False Negative) là số lượng mẫu dương bị dự đoán nhầm là âm.

Ưu điểm của accuracy là dễ hiểu và tính toán đơn giản. Tuy nhiên, chỉ số này có thể gây hiểu nhầm trong trường hợp dữ liệu mất cân bằng (imbalanced data), khi số lượng mẫu của một lớp chiếm đa số. Ví dụ, nếu 90% hồ sơ vay được chấp nhận, thì một mô hình luôn dự đoán "chấp nhận" sẽ có accuracy 90%, mặc dù mô hình này không có giá trị thực tế.

Trong lĩnh vực tài chính, đặc biệt là đánh giá tín dụng, accuracy cung cấp cái nhìn tổng quát về hiệu suất của mô hình nhưng cần được bổ sung bằng các chỉ số đánh giá khác để có đánh giá toàn diện hơn.

2.3.2. Precision (Độ chính xác dự báo)

Precision được sử dụng để đánh giá mức độ chính xác trong các dự đoán dương của mô hình. Nói cách khác, precision phản ánh trong số các trường hợp mà mô hình dự đoán là dương, có bao nhiêu phần trăm là chính xác:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3)$$

Trong bối cảnh đánh giá tín dụng, precision cao có nghĩa là khi mô hình dự đoán một khách hàng sẽ được chấp nhận vay, thì dự đoán này có độ tin cậy cao. Precision thấp sẽ dẫn đến việc cấp tín dụng cho những khách hàng không đáp ứng đủ điều kiện, từ đó làm tăng rủi ro tín dụng cho tổ chức tài chính.

Precision đặc biệt quan trọng khi chi phí của false positive cao. Ví dụ, trong trường hợp ngân hàng muốn giảm thiểu rủi ro tín dụng, họ sẽ ưu tiên mô hình có precision cao để hạn chế việc cấp vốn cho khách hàng không đủ điều kiện.

2.3.3. Recall (Độ nhạy)

Recall, hay còn gọi là Sensitivity hoặc True Positive Rate, là thước đo phản ánh khả năng mô hình phát hiện các trường hợp dương thực sự. Recall được định nghĩa là tỷ lệ giữa số lượng mẫu dương được dự đoán đúng trên tổng số mẫu dương thực tế:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

Trong lĩnh vực tín dụng, recall cao có nghĩa là mô hình có khả năng xác định được phần lớn khách hàng đủ điều kiện vay vốn. Recall thấp đồng nghĩa với việc nhiều khách hàng tiềm năng (có khả năng trả nợ) sẽ bị từ chối khoản vay, dẫn đến mất cơ hội kinh doanh cho tổ chức tài chính.

Recall đặc biệt quan trọng khi chi phí của false negative cao. Ví dụ, trong trường hợp ngân hàng muốn mở rộng danh mục khách hàng và tăng doanh thu, họ sẽ ưu tiên mô hình có recall cao để không bỏ lỡ khách hàng tiềm năng.

2.3.4. F1-score

F1-score là chỉ số tổng hợp giữa precision và recall, được tính theo trung bình điều hòa (harmonic mean). F1-score mang lại cái nhìn cân bằng trong trường hợp cần xem xét đồng thời cả độ chính xác của dự đoán dương (precision) và khả năng phát hiện đầy đủ các trường hợp dương (recall):

$$\text{F1score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

F1-score đạt giá trị cao nhất là 1 khi cả precision và recall đều bằng 1, và giá trị thấp nhất là 0 khi một trong hai chỉ số này bằng 0. F1-score đặc biệt hữu ích trong trường hợp dữ liệu mất cân bằng và khi cần cân nhắc đồng thời hai mục tiêu:

1. Giảm thiểu rủi ro tín dụng bằng cách không cấp vốn cho khách hàng không đủ điều kiện (precision cao)
2. Tối đa hóa cơ hội kinh doanh bằng cách xác định được nhiều khách hàng tiềm năng (recall cao)

Trong thực tế ngành tài chính, F1-score thường được sử dụng như một chỉ số tổng hợp để so sánh hiệu suất tổng thể của các mô hình khác nhau.

2.3.5. ROC và AUC

ROC (Receiver Operating Characteristic) là đường cong biểu diễn mối quan hệ giữa True Positive Rate (TPR = Recall) và False Positive Rate (FPR = FP/(FP+TN)) khi thay

đổi ngưỡng phân loại của mô hình. Đường cong ROC cho thấy sự đánh đổi (trade-off) giữa độ nhạy (sensitivity) và độ đặc hiệu (specificity) của mô hình.

AUC (Area Under the Curve) là diện tích dưới đường cong ROC, phản ánh khả năng phân biệt hai lớp của mô hình. AUC có giá trị từ 0 đến 1, với các đặc điểm:

- AUC = 1.0: Mô hình phân loại hoàn hảo, có khả năng phân biệt hoàn toàn giữa các khách hàng được chấp nhận và bị từ chối vay.
- AUC > 0.9: Mô hình có khả năng phân loại xuất sắc.
- AUC = 0.8-0.9: Mô hình có khả năng phân loại tốt.
- AUC = 0.7-0.8: Mô hình có khả năng phân loại khá.
- AUC = 0.6-0.7: Mô hình có khả năng phân loại trung bình.
- AUC = 0.5-0.6: Mô hình có khả năng phân loại kém.
- AUC = 0.5: Mô hình phân loại ngẫu nhiên, không có khả năng phân biệt.
- AUC < 0.5: Mô hình phân loại tệ hơn ngẫu nhiên, thường do lỗi cài đặt.

AUC là một chỉ số đặc biệt quan trọng trong lĩnh vực tài chính vì các lý do sau:

1. Không bị ảnh hưởng bởi sự mất cân bằng của dữ liệu.
2. Độc lập với ngưỡng phân loại, cho phép đánh giá mô hình ở tất cả các ngưỡng có thể.
3. Cung cấp thông tin về khả năng mô hình xếp hạng (ranking) tốt hay không, điều này phù hợp với nhu cầu đánh giá rủi ro tín dụng.

Trong thực tế, các tổ chức tài chính thường sử dụng AUC như một trong những chỉ số quan trọng nhất để đánh giá và so sánh các mô hình dự đoán khả năng chấp nhận vay vốn.

2.4. Tổng quan về bài toán phê duyệt khoản vay

Trong lĩnh vực tài chính ngân hàng, việc ra quyết định có phê duyệt một khoản vay hay không đóng vai trò then chốt trong việc quản trị rủi ro tín dụng. Bài toán phê duyệt khoản vay là một bài toán phân loại nhị phân, trong đó đầu vào là các đặc trưng liên quan đến người vay và điều kiện vay, đầu ra là quyết định phê duyệt (Y) hoặc từ chối (N) khoản vay.

Với sự phát triển của các hệ thống dữ liệu lớn và công nghệ học máy, việc tự động hóa và tối ưu hóa quá trình đánh giá hồ sơ vay đang ngày càng trở nên phổ biến nhằm giảm thiểu rủi ro, tiết kiệm chi phí và nâng cao hiệu quả hoạt động.

2.4.1. Các tiêu chí trong quyết định phê duyệt khoản vay

Các ngân hàng và tổ chức tín dụng thường căn cứ vào nhiều tiêu chí để đưa ra quyết định phê duyệt khoản vay:

- Thông tin cá nhân: Giới tính, tình trạng hôn nhân, số người phụ thuộc, trình độ học vấn, tình trạng làm chủ.
- Thông tin tài chính: Thu nhập của người vay và người đồng vay, số tiền vay, thời hạn vay.
- Lịch sử tín dụng và bất động sản: Lịch sử tín dụng (tốt/xấu), khu vực bất động sản.

Những tiêu chí này phản ánh khả năng trả nợ và mức độ rủi ro của từng cá nhân, từ đó hỗ trợ mô hình trong việc đưa ra quyết định hợp lý về khoản vay.

2.4.2. Ứng dụng học máy trong lĩnh vực tài chính ngân hàng

Học máy (Machine Learning) đã và đang được ứng dụng rộng rãi trong ngành tài chính – ngân hàng, đặc biệt trong các bài toán:

- Phê duyệt khoản vay: Dự đoán khả năng khách hàng sẽ được duyệt vay dựa trên dữ liệu lịch sử.
- Đánh giá rủi ro tín dụng: Phân tích và xếp hạng mức độ rủi ro của khách hàng.
- Phát hiện gian lận: Phát hiện các hành vi bất thường trong giao dịch tài chính.
- Tư vấn tài chính cá nhân: Gợi ý sản phẩm phù hợp với hồ sơ và lịch sử tài chính của khách hàng.

Việc áp dụng các mô hình học máy giúp tăng tốc độ xử lý, cải thiện độ chính xác và nâng cao trải nghiệm khách hàng, đồng thời góp phần tối ưu hóa hiệu quả kinh doanh của các tổ chức tài chính.

CHƯƠNG 3: THỰC NGHIỆM

3.1. Giới thiệu bộ dữ liệu

Việc lựa chọn bộ dữ liệu phù hợp đóng vai trò then chốt trong đánh giá và triển khai các mô hình học máy. Trong bối cảnh tài chính – ngân hàng, bài toán dự đoán khả năng phê duyệt khoản vay không chỉ mang tính ứng dụng cao mà còn phản ánh nhiều yếu tố phức tạp liên quan đến khách hàng và điều kiện vay. Phân tích bài toán này sẽ giúp kiểm chứng hiệu quả của mô hình và khả năng giải thích trong các tình huống thực tế.

3.1.1. Nguồn gốc và đặc điểm bộ dữ liệu

Bộ dữ liệu Loan Approval Prediction được lấy từ nền tảng Kaggle – một kho lưu trữ dữ liệu phổ biến trong cộng đồng khoa học dữ liệu. Dữ liệu bao gồm 598 quan sát tương ứng với các khoản vay cá nhân thực tế, được mô tả thông qua 13 đặc trưng đầu vào cùng biến mục tiêu Loan_Status (Y: được duyệt, N: không được duyệt). Đây là bài toán phân loại nhị phân, phản ánh bài toán ra quyết định trong môi trường tài chính, nơi các yếu tố nhân khẩu học, kinh tế và lịch sử tín dụng ảnh hưởng trực tiếp đến khả năng được cấp vay.

Để hiểu rõ hơn về quy mô và cấu trúc dữ liệu, nhóm đã tiến hành phân tích sơ bộ với các thông tin như sau:

- Số lượng khoản vay cá nhân (số dòng dữ liệu): num_obs quan sát, tương ứng với số lượng đơn đăng ký vay trong bộ dữ liệu.
- Số lượng đặc trưng (cột): num_vars đặc trưng, bao gồm các thông tin như giới tính, thu nhập, tình trạng hôn nhân, lịch sử tín dụng, số người phụ thuộc, số tiền vay,...

3.1.2. Mô tả các biến trong bộ dữ liệu

Bộ dữ liệu gồm cả biến định tính (categorical) và định lượng (numerical). Việc phân loại giúp áp dụng các kỹ thuật xử lý và phân tích phù hợp.

- Các biến định tính gồm: Loan_ID, Gender, Married, Education, Self_Employed, Property_Area, Loan_Status. Một số đặc điểm nổi bật:
  - 64.88% người vay đã kết hôn.
  - Nam chiếm 81.44%, nữ chiếm 18.56%.
  - 77.76% có trình độ tốt nghiệp đại học.
  - 68.73% các khoản vay được phê duyệt.
- Các biến định lượng gồm: Dependents, ApplicantIncome, CoapplicantIncome, LoanAmount, Loan_Amount_Term, Credit_History. Thống kê mô tả cho thấy sự chênh lệch lớn về thu nhập và số tiền vay giữa các cá nhân.

Ví dụ, người có lịch sử tín dụng tốt (Credit_History = 1) có tới 79.48% được duyệt vay, trong khi người không có lịch sử tín dụng chỉ có 8.14% được duyệt.

Các biến trong bộ dữ liệu được chia thành ba nhóm chính:

- Thông tin cá nhân: Gender, Married, Dependents, Education, Self_Employed. Phản ánh đặc điểm nhân khẩu học của người vay.
- Thông tin tài chính: ApplicantIncome, CoapplicantIncome, LoanAmount, Loan_Amount_Term. Thể hiện khả năng tài chính và điều kiện khoản vay.
- Lịch sử tín dụng & bất động sản: Credit_History, Property_Area. Cho biết mức độ uy tín tín dụng và khu vực tài sản bảo đảm.

Biến mục tiêu Loan_Status cho biết kết quả phê duyệt khoản vay, tổng số khoản vay là 598 (411 khoản vay được chấp nhận và 187 khoản vay không được chấp nhận), với tỷ lệ được duyệt chiếm khoảng 69%. Ngoài ra, dữ liệu cũng chứa các biến định tính (như Gender, Education,...) và định lượng (như LoanAmount, ApplicantIncome,...), tạo điều kiện thuận lợi cho việc áp dụng đa dạng các thuật toán học máy và kỹ thuật giải thích mô hình.

3.2. Tiền xử lý dữ liệu

Trước khi xây dựng và huấn luyện các mô hình học máy, dữ liệu cần được xử lý và làm sạch để đảm bảo tính chính xác và hiệu quả của quá trình phân tích. Các bước tiền xử lý giúp loại bỏ nhiễu, xử lý các giá trị không hợp lệ hoặc bị thiếu, đồng thời chuẩn bị dữ liệu ở định dạng phù hợp cho thuật toán học máy. Trong phần này, chúng tôi sẽ trình bày các bước tiền xử lý đã thực hiện trên tập dữ liệu, bắt đầu với việc xử lý các giá trị bị thiếu.

3.2.1. Xử lý giá trị bị thiếu

Trong quá trình thu thập dữ liệu thực tế, việc xuất hiện các giá trị bị thiếu (missing values) là điều không thể tránh khỏi. Nếu không được xử lý hợp lý, những giá trị thiếu này có thể gây sai lệch trong việc huấn luyện mô hình và làm giảm độ chính xác của kết quả. Do đó, bước đầu tiên trong tiền xử lý là kiểm tra và xử lý các giá trị bị thiếu trong tập dữ liệu.

Tập dữ liệu được kiểm tra bằng cách sử dụng phương thức .isnull().sum() để xác định số lượng giá trị bị thiếu ở mỗi cột:

Số lượng dữ liệu bị thiếu trước khi xử lý:

```
Số lượng dữ liệu bị thiếu trước khi xử lý:
Loan_ID               0
Gender                0
Married               0
Dependents           12
Education             0
Self_Employed         0
ApplicantIncome       0
CoapplicantIncome     0
LoanAmount           21
Loan_Amount_Term     14
Credit_History       49
Property_Area         0
Loan_Status           0
dtype: int64
```

Hình 3.1. Kết quả dữ liệu thiếu trước khi xử lý

Sử dụng phương pháp điền giá trị thiếu:

- Đối với các biến số liên tục như LoanAmount, Loan_Amount_Term, ApplicantIncome, và CoapplicantIncome, thay thế giá trị bị thiếu bằng giá trị trung bình mean() của từng cột. Việc này giúp giữ nguyên phân phối dữ liệu và giảm thiểu sai lệch.
- Đối với các biến phân loại như Dependents, Credit_History, Gender, Married, Education, Self_Employed, và Property_Area, điền giá trị thiếu bằng giá trị xuất hiện nhiều nhất mode()[0]. Cách tiếp cận này giúp bảo toàn tính phổ biến của các nhóm trong dữ liệu.

Số lượng dữ liệu bị thiếu sau khi điền vào:

```
Số lượng dữ liệu bị thiếu sau khi điền vào:
Loan_ID              0
Gender               0
Married              0
Dependents           0
Education            0
Self_Employed        0
ApplicantIncome      0
CoapplicantIncome    0
LoanAmount           0
Loan_Amount_Term     0
Credit_History       0
Property_Area        0
Loan_Status          0
dtype: int64
```

Hình 3.2. Kết quả dữ liệu thiếu sau khi xử lý bằng phương pháp điền dữ liệu

Sau khi điền giá trị, dữ liệu tiếp tục được kiểm tra lại để đảm bảo không còn giá trị bị thiếu. Cuối cùng, tập dữ liệu đã được xử lý phục vụ cho các bước phân tích tiếp theo.

3.2.2. Chuẩn hóa dữ liệu

Dữ liệu có một số đặc điểm quan trọng:

- Có phân phối lệch phải ở các cột như ApplicantIncome, CoapplicantIncome, LoanAmount.
- Chứa ngoại lai, đặc biệt ở ApplicantIncome (giá trị cao nhất là 81,000) và LoanAmount (giá trị cao nhất là 650).
- Có nhiều giá trị bằng 0 ở CoapplicantIncome, dẫn đến phân phối không đồng đều

➔ Dựa vào các đặc điểm trên, ta đưa ra các phương pháp chuẩn hóa sau:

3.2.2.1. Min-Max Scaling

- **Kết quả:**

```
Min-Max Scaled Data:
 [[0.12183055 0.03998368 0.328594  ]
 [0.1569697  0.26322989 0.52448657]
 [0.07235622 0.         0.21958926]
 ...
 [0.04753247 0.07857537 0.30015798]
 [0.08884354 0.1999904  0.24960506]
 [0.05568336 0.08599131 0.20216241]]
```

Hình 3.3. Kết quả chuẩn hóa Min-Max Scaling

- **Ý tưởng:** Đưa các giá trị về một phạm vi xác định, thường là từ 0 đến 1. Điều này làm cho dữ liệu nằm gọn trong một khoảng giá trị cố định và có thể giúp mô hình học máy hội tụ nhanh hơn trong quá trình tối ưu.
- **Nguyên lý hoạt động**
  - Công thức:

$$X' = \left(X_{\max} - X_{\min}\right) / \left(X - X_{\min}\right) \quad (6)$$

  - Co dãn dữ liệu về khoảng [0,1]
  - Giá trị nhỏ nhất được đưa về 0, giá trị lớn nhất về 1.
- **Quan sát trên dữ liệu**
  - Dữ liệu sau chuẩn hóa có giá trị từ 0 đến 1, giúp giữ nguyên tỷ lệ khoảng cách giữa các giá trị.
  - Vì CoapplicantIncome có nhiều giá trị bằng 0, phần lớn dữ liệu bị dồn về phía dưới của trục số.
  - ApplicantIncome có giá trị cao nhất 81,000, nên sau khi chuẩn hóa, các giá trị thấp hơn bị nén lại gần 0.
- **Ưu điểm:** Đơn giản và trực quan, giúp mô hình hội tụ nhanh hơn.

- **Nhược điểm:** Dễ bị ảnh hưởng bởi giá trị ngoại lệ, vì nếu có một giá trị rất lớn hoặc rất nhỏ, nó sẽ làm thay đổi phạm vi của toàn bộ dữ liệu.

3.2.2.2. Robust Scaler

- **Kết quả:**

```
Robust Scaled Data:
 [[ 2.0852292   0.25312874  1.49230769]
 [ 3.04991511  4.30231799  3.4       ]
 [ 0.72699491 -0.47208619  0.43076923]
 ...
 [ 0.04550085  0.95309609  1.21538462]
 [ 1.17962649  3.15529437  0.72307692]
 [ 0.26926995  1.08760474  0.26105853]]
```

Hình 3.4. Kết quả chuẩn hóa Robust Scaled

- **Nguyên lý hoạt động**
  - Công thức:

$$X' = \left(X - median\right) / IQR \quad (7)$$

  - Sử dụng median (trung vị) thay vì mean (trung bình).
  - Chia tỷ lệ theo IQR (khoảng giữa Q1 - Q3) thay vì độ lệch chuẩn.
- **Quan sát trên dữ liệu**
  - Sau chuẩn hóa, trung vị của tất cả các cột là 0.
  - Ngoại lai bị giảm ảnh hưởng, nhưng vẫn có một số giá trị lớn hơn 1 (ví dụ: ApplicantIncome có giá trị max 26.91).
  - CoapplicantIncome có nhiều giá trị bằng 0, nên phần lớn dữ liệu vẫn tập trung gần trung vị.
- **Ý tưởng**: Sử dụng các thống kê phân vị (quartiles) thay vì giá trị trung bình và độ lệch chuẩn, giúp giảm ảnh hưởng của các giá trị ngoại lệ.
- **Ưu điểm**: Tốt hơn Min-Max Scaling và Standardization khi dữ liệu có nhiều giá trị ngoại lệ.
- **Nhược điểm**: Khi dữ liệu có phân phối rất hẹp, phương pháp này có thể không cải thiện đáng kể.

3.2.2.3. Standardization (Z-score normalization)

- **Kết quả:**

```
Standardized Data:
 [[ 0.78868524  0.0254584   0.91320682]
 [ 1.2689553   3.30405611  2.35766744]
 [ 0.11248663 -0.5617426   0.10943437]
 ...
 [-0.22679602  0.59221664  0.70352705]
 [ 0.33782981  2.37531982  0.33076302]
 [-0.1153923   0.70112727 -0.01906666]]
```

Hình 3.5. Kết quả chuẩn hóa Standardized

- **Nguyên lý hoạt động**
  - Công thức:

$$X' = (X - \mu) / \sigma \quad (8)$$

  - Dữ liệu được đưa về trung bình 0 và độ lệch chuẩn 1.
  - Giữ nguyên thông tin về phân phối và khoảng cách tương đối giữa các điểm dữ liệu.
- **Quan sát trên dữ liệu**
  - Sau chuẩn hóa, trung bình của ApplicantIncome, CoapplicantIncome, LoanAmount gần 0.
  - Độ lệch chuẩn $\approx$ 1, giúp dữ liệu có tỷ lệ hợp lý hơn giữa các đặc trưng.
  - Tuy nhiên, ngoại lai vẫn tồn tại: ApplicantIncome có giá trị cao nhất ≈ 13, cho thấy vẫn có giá trị rất lớn so với trung bình.
- **Ý tưởng:** Đưa các giá trị về dạng phân phối chuẩn với giá trị trung bình bằng 0 và độ lệch chuẩn là 1, phương pháp này hữu ích khi dữ liệu có phân phối gần giống với phân phối chuẩn.
- **Ưu điểm:** Ít bị ảnh hưởng bởi các giá trị ngoại lệ hơn so với Min-Max Scaling và phù hợp khi dữ liệu có phân phối chuẩn.
- **Nhược điểm:** Không đảm bảo dữ liệu sẽ nằm trong một khoảng cố định.

3.2.3. Loại bỏ dữ liệu ngoại lai

Trong quá trình xử lý dữ liệu vay vốn, việc phát hiện và xử lý giá trị ngoại lai đóng vai trò quan trọng nhằm nâng cao độ chính xác của mô hình dự đoán. Giá trị ngoại lai là những điểm dữ liệu khác biệt rõ rệt so với phần lớn quan sát, có thể xuất phát từ lỗi nhập liệu hoặc các trường hợp đặc biệt. Nghiên cứu tập trung xác định ngoại lai ở các biến số học gồm "ApplicantIncome", "CoapplicantIncome" và "LoanAmount" bằng phương pháp IQR. Cụ thể, các giá trị nằm ngoài khoảng (Q1 - 1.5×IQR, Q3 + 1.5×IQR) được

xem là ngoại lai và bị loại khỏi tập dữ liệu. Việc loại bỏ này giúp giảm nhiễu, tránh ảnh hưởng bất thường lên mô hình và phản ánh tốt hơn xu hướng chung của dữ liệu.

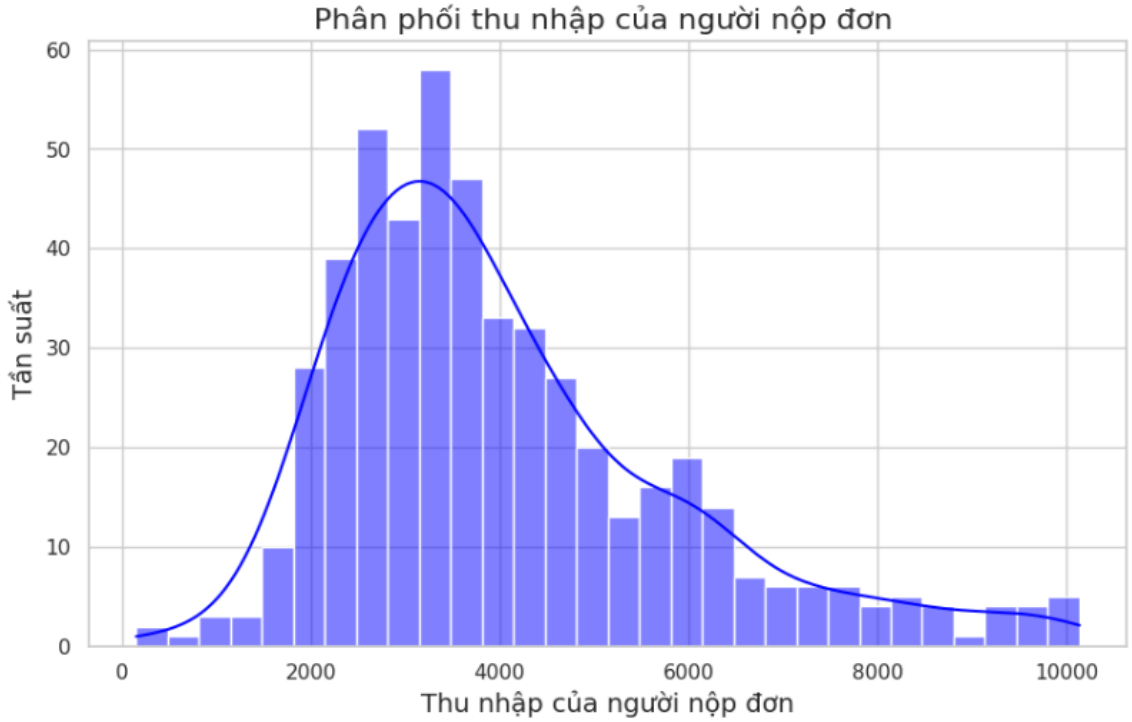
3.3. Trực quan hóa dữ liệu

Trực quan hóa dữ liệu đóng vai trò then chốt trong quá trình phân tích và xây dựng mô hình dự đoán khả năng chấp nhận vay vốn. Việc biểu diễn dữ liệu một cách trực quan không chỉ giúp nhà phân tích hiểu rõ hơn về bản chất và đặc điểm của dữ liệu, mà còn hỗ trợ phát hiện các mẫu, xu hướng và mối quan hệ tiềm ẩn giữa các biến. Trong nghiên cứu này, chúng tôi sử dụng các công cụ trực quan hóa từ thư viện Matplotlib và Seaborn để khám phá sâu hơn về dữ liệu đã qua xử lý, nhằm đạt được cái nhìn toàn diện trước khi tiến hành xây dựng các mô hình học máy.

Quá trình trực quan hóa dữ liệu nhằm trả lời các câu hỏi quan trọng: Yếu tố nào ảnh hưởng nhiều nhất đến quyết định cấp vốn? Có sự khác biệt rõ rệt giữa nhóm được chấp nhận và bị từ chối không? Các biến đầu vào liên hệ như thế nào với nhau và với biến mục tiêu? Những phân tích này giúp hiểu rõ dữ liệu và hỗ trợ lựa chọn, tinh chỉnh mô hình học máy phù hợp.

3.3.1. Phân tích phân phối các biến đơn lẻ

3.3.1.1. Phân phối thu nhập ứng viên

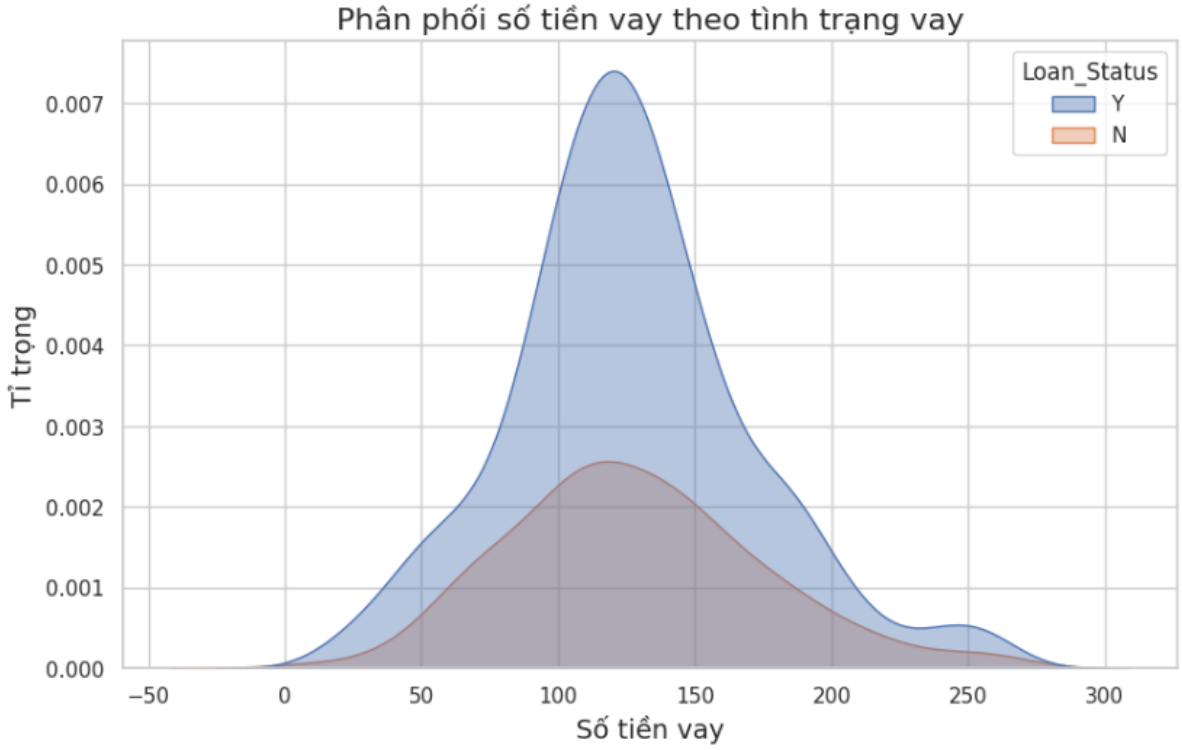


Hình 3.6. Biểu đồ phân phối thu nhập ứng viên (Applicant Income)

Phân tích biểu đồ phân phối thu nhập của ứng viên sau khi xử lý dữ liệu cho thấy đặc điểm phân bố rõ rệt về thu nhập của những người nộp đơn vay vốn. Sau khi áp dụng phương pháp loại bỏ các điểm ngoại lai, biểu đồ thể hiện phần lớn ứng viên có mức thu nhập tập trung trong khoảng từ 2,000 đến 10,000 đơn vị tiền tệ, phản ánh đúng phân khúc chính của đối tượng khách hàng mà các tổ chức tài chính đang phục vụ. Một số ít ứng viên có thu nhập vượt ngưỡng 15,000, tuy nhiên số lượng này đã giảm đáng kể sau quá trình xử lý dữ liệu ngoại lai, giúp phân phối trở nên cân đối và đại diện hơn cho tổng thể dữ liệu.

Đặc điểm nổi bật của phân phối là sự lệch phải (right-skewed), cho thấy mặc dù có một số ứng viên sở hữu mức thu nhập rất cao, nhưng đại đa số vẫn thuộc nhóm có thu nhập trung bình hoặc thấp. Điều này có ý nghĩa quan trọng trong việc đánh giá khả năng vay vốn, bởi các ứng viên có thu nhập cao thường có khả năng được phê duyệt khoản vay với số tiền lớn hơn, nhưng họ chỉ chiếm một phần nhỏ trong tổng thể dữ liệu. Thông tin này giúp các tổ chức tài chính hiểu rõ hơn về cấu trúc thu nhập của khách hàng tiềm năng, từ đó có thể thiết kế các sản phẩm vay phù hợp với từng phân khúc và đưa ra các chính sách thẩm định phù hợp.

3.3.1.2. Phân phối khoản vay theo trạng thái phê duyệt

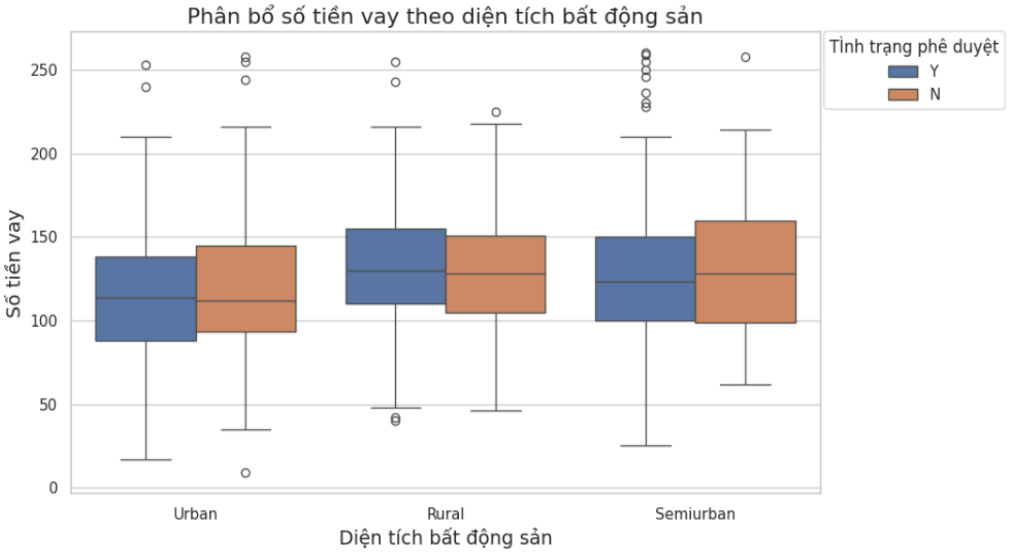


Hình 3.7. Biểu đồ phân phối khoản vay theo trạng thái phê duyệt

Phân tích biểu đồ KDE (Kernel Density Estimation) thể hiện rõ nét sự khác biệt trong phân phối số tiền vay giữa các khoản được phê duyệt và bị từ chối. Biểu đồ cho thấy những khoản vay được chấp thuận (Y) có xu hướng tập trung chủ yếu ở mức trung bình thấp, dao động trong khoảng 100-150 đơn vị tiền tệ, hình thành một đỉnh phân phối tương đối rõ ràng. Điều này phản ánh chính sách thận trọng của các tổ chức tài chính, ưu tiên cấp các khoản vay có giá trị vừa phải nhằm giảm thiểu rủi ro.

Ngược lại, các khoản vay bị từ chối (N) có phân phối rộng hơn đáng kể, trải dài trên phổ giá trị và bao gồm cả những khoản vay có giá trị cao. Đặc biệt, biểu đồ chỉ ra rằng những ứng viên yêu cầu khoản vay có giá trị lớn đối mặt với xác suất bị từ chối cao hơn. Xu hướng này có thể được giải thích bởi mức độ rủi ro tăng cao đối với các khoản vay lớn, đặc biệt khi những khoản vay này không tương xứng với khả năng tài chính của người vay hoặc thiếu các đảm bảo cần thiết. Thông tin này có giá trị quan trọng đối với cả tổ chức tài chính và khách hàng tiềm năng, giúp định hướng chiến lược thẩm định khoản vay cũng như tư vấn khách hàng về mức vay có khả năng được chấp thuận cao nhất.

3.3.1.3. Phân phối khoản vay theo khu vực tài sản



Hình 3.8. Biểu đồ phân phối khoản vay theo khu vực tài sản

Phân tích biểu đồ boxplot thể hiện mối quan hệ đa chiều giữa số tiền vay (LoanAmount), khu vực tài sản (Property_Area) và trạng thái phê duyệt khoản vay cho thấy những khác biệt đáng chú ý giữa các khu vực địa lý. Tại khu vực thành phố (Urban), phân phối khoản vay có biên độ dao động rộng hơn so với các khu vực khác, bao gồm cả những khoản vay có giá trị lớn vượt mức 200 đơn vị tiền tệ, phản ánh giá trị bất động sản và nhu cầu vốn cao hơn tại đô thị. Tuy nhiên, đáng chú ý là ngay cả ở khu vực này,

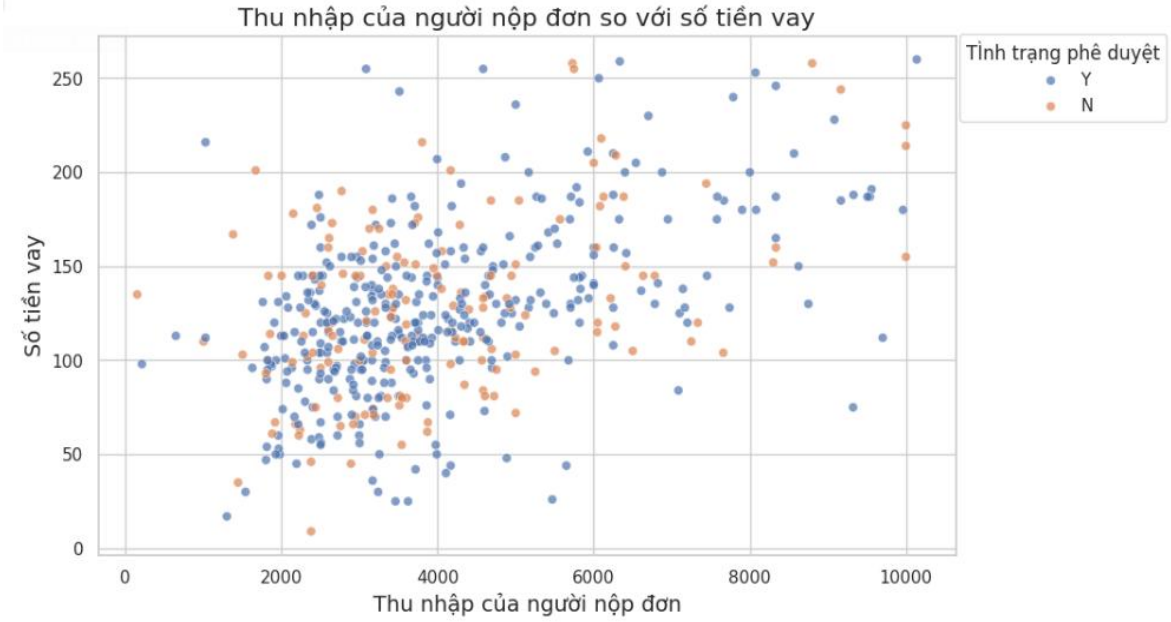
vẫn có tỷ lệ đáng kể các khoản vay bị từ chối, cho thấy giá trị khoản vay cao không phải là yếu tố duy nhất quyết định việc phê duyệt.

Khu vực bán thành thị (Semiurban) thể hiện đặc điểm nổi bật với phân phối khoản vay tập trung chủ yếu ở mức trung bình và có tỷ lệ phê duyệt cao nhất trong ba khu vực. Hiện tượng này có thể được giải thích bởi sự cân bằng lý tưởng giữa giá trị bất động sản vừa phải và khả năng trả nợ ổn định của người dân sống ở khu vực này. Ngược lại, tại khu vực nông thôn (Rural), các khoản vay chủ yếu có giá trị nhỏ (dưới 150 đơn vị tiền tệ), tương ứng với giá bất động sản thấp hơn, nhưng đi kèm với tỷ lệ từ chối cao, có thể do thu nhập thấp hơn hoặc tính ổn định của nguồn thu nhập không cao của người dân nông thôn.

Ý nghĩa quan trọng từ phân tích này là khu vực bán thành thị dường như mang lại cơ hội vay vốn thuận lợi nhất, với tỷ lệ phê duyệt cao hơn, có thể xuất phát từ điều kiện kinh tế cân bằng và khả năng tín dụng tốt của cư dân tại đây. Thông tin này hữu ích cho cả tổ chức tài chính trong việc phân bổ nguồn lực và định hướng thị trường, cũng như cho khách hàng tiềm năng trong việc đánh giá cơ hội vay vốn của mình dựa trên vị trí địa lý của tài sản.

3.3.2. Phân tích mối quan hệ giữa các biến

3.3.2.1. Biểu đồ phân tán của Thu nhập ứng viên (Applicant Income) và Số tiền vay (Loan Amount)



Hình 3.9. Biểu đồ thu nhập ứng viên so với khoản vay

Phân tích biểu đồ tán xạ thể hiện mối tương quan giữa thu nhập của ứng viên (ApplicantIncome) và số tiền vay yêu cầu (LoanAmount), với trạng thái phê duyệt

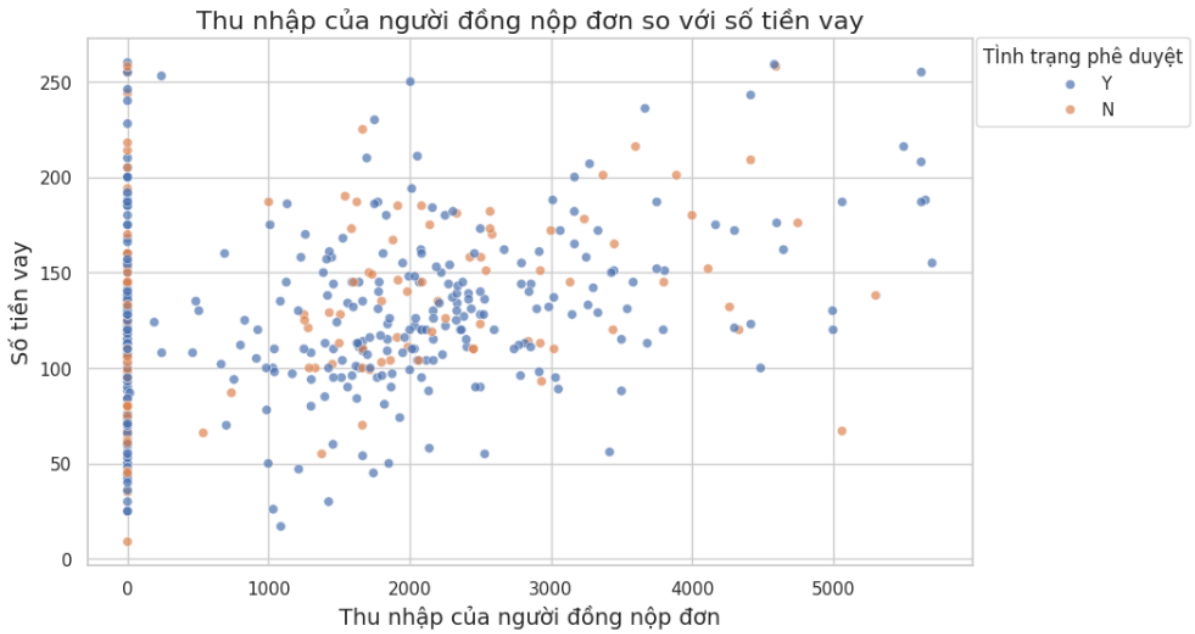
(Loan_Status) được phân biệt bằng màu sắc, cung cấp những hiểu biết sâu sắc về quá trình ra quyết định tín dụng. Dữ liệu cho thấy một xu hướng rõ ràng: các khoản vay có giá trị nhỏ (dưới 200 đơn vị tiền tệ) thường nhận được tỷ lệ phê duyệt cao hơn, phản ánh mức độ rủi ro thấp hơn mà các tổ chức tài chính gắn với những khoản vay này.

Biểu đồ cho thấy mối quan hệ tích cực giữa thu nhập và số tiền vay, với xu hướng tăng từ trái dưới lên phải trên. Tuy nhiên, thu nhập cao và khoản vay lớn không đồng nghĩa với việc được phê duyệt, thể hiện qua nhiều điểm bị từ chối ở vùng thu nhập và khoản vay cao.

Biểu đồ cho thấy một số ứng viên thu nhập cao vẫn bị từ chối vay, cho thấy thẩm định tín dụng là quá trình đánh giá đa chiều. Ngoài thu nhập, các yếu tố như lịch sử tín dụng, tỷ lệ nợ, khu vực tài sản và các biến định tính khác cũng ảnh hưởng đáng kể đến quyết định cho vay.

Ý nghĩa sâu sắc của phân tích này là ngay cả khi ứng viên sở hữu mức thu nhập đáng kể, khả năng được phê duyệt khoản vay vẫn phụ thuộc vào một tập hợp phức tạp các yếu tố. Điều này nhấn mạnh tầm quan trọng của việc xây dựng các mô hình học máy đa biến có khả năng nắm bắt những tương tác phức tạp này để dự đoán chính xác kết quả phê duyệt khoản vay.

3.3.2.2. Thu nhập người cùng vay (Coapplicant Income) so với khoản vay



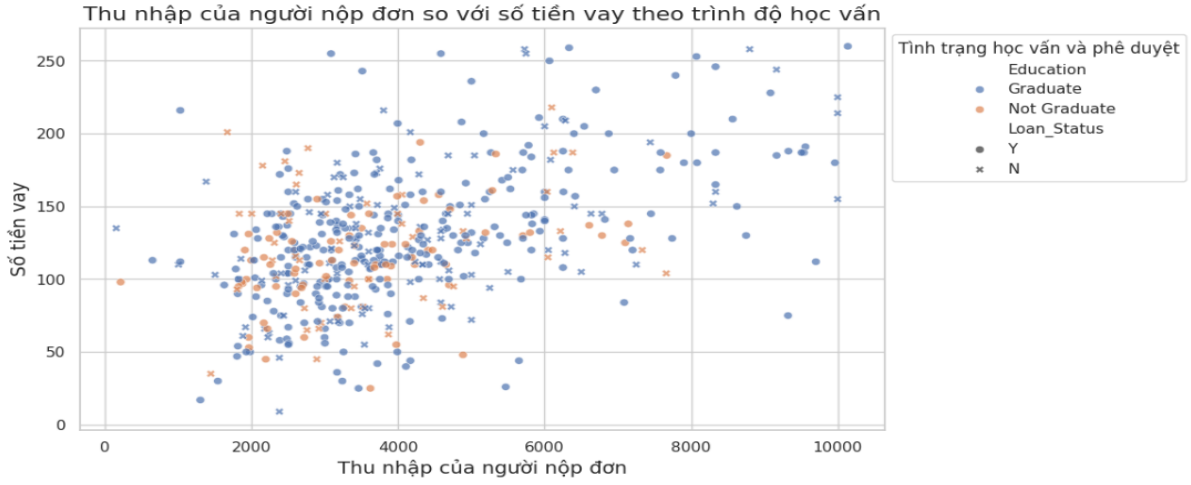
Hình 3.10. Biểu đồ thu nhập người cùng vay (Coapplicant Income) so với khoản vay

Phân tích biểu đồ tán xạ thể hiện mối quan hệ giữa thu nhập của người đồng vay (CoapplicantIncome) và số tiền vay (LoanAmount), với phân loại theo trạng thái phê duyệt, cho thấy những thông tin giá trị về cấu trúc hồ sơ vay và tác động của người đồng vay đến quyết định cấp tín dụng. Biểu đồ chỉ ra rằng phần lớn các hồ sơ vay có giá trị thu nhập của người đồng vay bằng 0, cho thấy đa số ứng viên nộp đơn vay vốn mà không có người đồng vay. Đáng chú ý, ngay cả trong nhóm này, vẫn có một tỷ lệ đáng kể được phê duyệt khoản vay, phản ánh thực tế rằng việc không có người đồng vay không nhất thiết là bất lợi trong quá trình thẩm định tín dụng nếu các yếu tố khác đạt yêu cầu.

Khi xem xét các trường hợp có người đồng vay với thu nhập cao (trên 1,500 đơn vị tiền tệ), dữ liệu thể hiện một xu hướng tích cực rõ rệt: số tiền vay được yêu cầu thường lớn hơn và tỷ lệ phê duyệt cũng cao hơn đáng kể. Điều này minh chứng rằng thu nhập của người đồng vay đóng vai trò như một yếu tố bổ sung quan trọng, tăng cường khả năng trả nợ tổng thể của khoản vay và do đó làm giảm rủi ro tín dụng mà tổ chức tài chính phải đối mặt. Người đồng vay có thu nhập cao tỏ ra là một yếu tố hỗ trợ đặc biệt hiệu quả trong việc nâng cao khả năng phê duyệt khoản vay, đặc biệt là đối với những khoản vay có giá trị lớn. Hiện tượng này có thể được giải thích bởi việc thu nhập kết hợp tạo ra một lá chắn bảo vệ mạnh mẽ hơn trước các rủi ro tài chính như mất việc làm hoặc giảm thu nhập của người vay chính.

Ý nghĩa quan trọng của phân tích này là việc có người đồng vay với thu nhập ổn định có thể cải thiện đáng kể cơ hội được phê duyệt khoản vay, đặc biệt đối với những khoản vay giá trị cao. Đây là thông tin hữu ích cho khách hàng tiềm năng khi xem xét chiến lược nộp đơn vay vốn, cũng như cho các tổ chức tài chính trong việc phát triển các sản phẩm tín dụng có sự tham gia của nhiều bên vay.

3.3.2.3. Thu nhập ứng viên so với số tiền vay theo trình độ học vấn



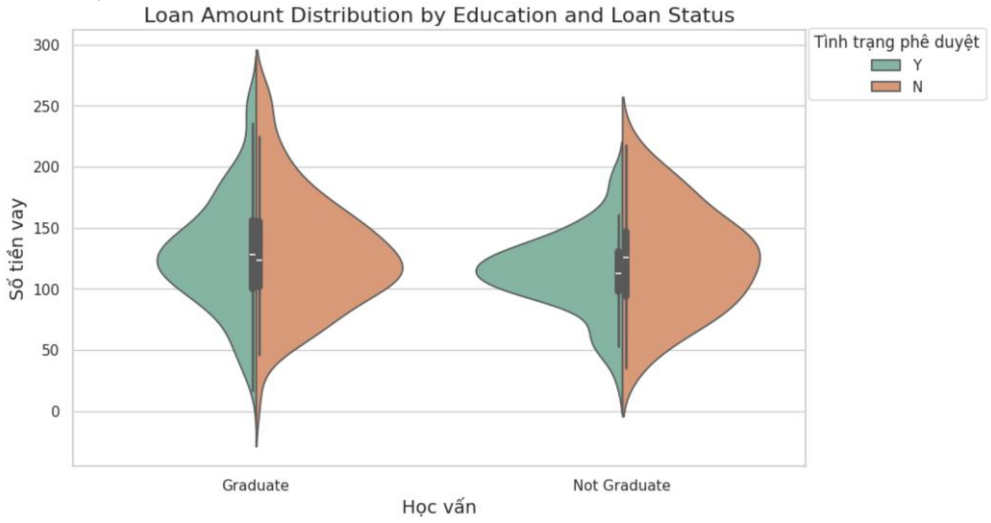
Hình 3.11. Biểu đồ thu nhập ứng viên so với số tiền vay theo trình độ học vấn

Phân tích biểu đồ tán xạ thể hiện mối quan hệ giữa thu nhập ứng viên (ApplicantIncome) và số tiền vay (LoanAmount), kết hợp với phân loại theo trình độ học vấn (Education) và trạng thái phê duyệt khoản vay (Loan_Status), cho thấy những khác biệt đáng chú ý giữa các nhóm ứng viên. Ứng viên có trình độ đại học (Graduate) biểu hiện xu hướng tiếp cận các khoản vay có giá trị cao hơn, đồng thời cũng nhận được tỷ lệ phê duyệt khá thuận lợi cho những khoản vay này. Điều này phản ánh mối tương quan tích cực giữa trình độ học vấn cao và khả năng vay vốn, có thể xuất phát từ niềm tin của các tổ chức tài chính vào tính ổn định việc làm và tiềm năng thu nhập lâu dài của nhóm đối tượng này.

Ngược lại, nhóm ứng viên không có bằng đại học (Not Graduate) thường yêu cầu các khoản vay có giá trị thấp hơn, phù hợp với mức thu nhập trung bình của họ. Tuy nhiên, đáng chú ý là nhóm này phải đối mặt với tỷ lệ từ chối cao hơn, ngay cả đối với những khoản vay giá trị thấp. Hiện tượng này có thể phản ánh nhận định của các tổ chức tài chính về tính ổn định và khả năng tăng trưởng thu nhập hạn chế hơn của nhóm không có bằng đại học, dẫn đến đánh giá rủi ro cao hơn.

Biểu đồ cho thấy thu nhập tuy ảnh hưởng đến kết quả phê duyệt khoản vay, nhưng không phải yếu tố quyết định. Nhiều trường hợp ứng viên thu nhập cao vẫn bị từ chối, dù có hoặc không có bằng đại học. Điều này phản ánh quá trình thẩm định tín dụng mang tính đa chiều, phụ thuộc vào nhiều yếu tố khác như lịch sử tín dụng, tỷ lệ nợ, thời gian làm việc và các chỉ số tài chính liên quan.

3.3.2.4. Số tiền vay (Loan Amount) theo Trạng thái vay (Loan Status) và Trình độ học vấn (Education)



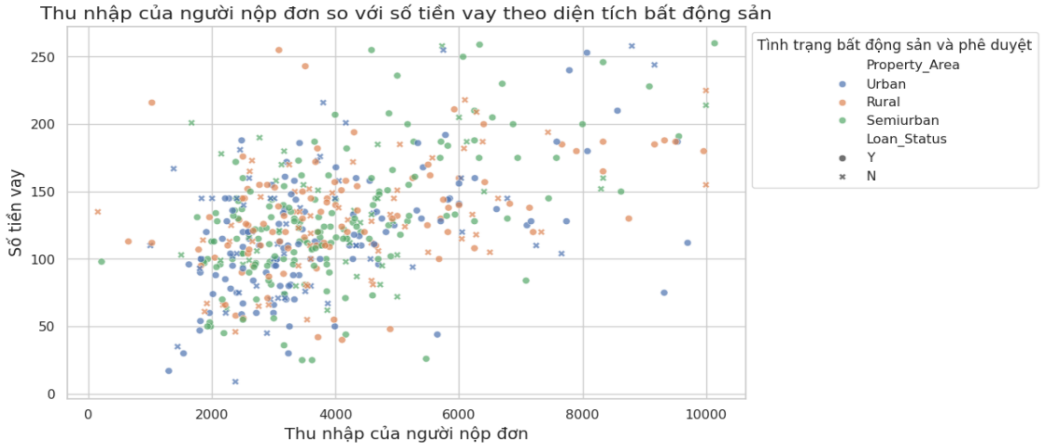
Hình 3.12. Violin Plot của Số tiền vay (Loan Amount) theo Trạng thái vay (Loan Status) và Trình độ học vấn (Education)

Biểu đồ violin plot thể hiện phân phối số tiền vay (Loan Amount) theo trình độ học vấn (Education) và trạng thái phê duyệt khoản vay (Loan Status) cung cấp những thông tin giá trị về các xu hướng cho vay. Qua biểu đồ, có thể thấy rõ sự khác biệt trong phân phối số tiền vay giữa nhóm tốt nghiệp đại học (Graduate) và nhóm chưa tốt nghiệp đại học (Not Graduate), cũng như giữa các khoản vay được chấp thuận (Y) và bị từ chối (N).

Đối với nhóm ứng viên tốt nghiệp đại học, phân phối số tiền vay có dải giá trị rộng hơn và có xu hướng tập trung ở mức cao hơn so với nhóm chưa tốt nghiệp đại học. Đặc biệt, khoản vay được chấp thuận (màu xanh) của nhóm Graduate có phân phối dày hơn ở phần giữa, cho thấy có nhiều khoản vay ở mức trung bình được phê duyệt. Trong khi đó, khoản vay bị từ chối (màu cam) của nhóm này có xu hướng tập trung ở các giá trị cao và thấp hơn, tạo thành hình dạng hai đỉnh, gợi ý rằng cả khoản vay quá nhỏ và quá lớn đều có rủi ro bị từ chối cao hơn. Với nhóm chưa tốt nghiệp đại học, phân phối số tiền vay có phạm vi hẹp hơn, phản ánh khả năng tài chính hạn chế hơn của nhóm này. Đáng chú ý, phân phối khoản vay bị từ chối (màu cam) của nhóm Not Graduate có đỉnh cao hơn và trải dài hơn so với khoản vay được chấp thuận, cho thấy tỷ lệ từ chối cao hơn ở nhiều mức số tiền vay khác nhau, đặc biệt ở các giá trị cao.

Quan sát này gợi ý rằng trình độ học vấn có thể ảnh hưởng đáng kể đến khả năng được phê duyệt khoản vay, với ứng viên có bằng đại học thường được đánh giá là có khả năng trả nợ tốt hơn ở cùng mức vay so với những người không có bằng đại học. Thông tin này có thể giúp các tổ chức tài chính tinh chỉnh các tiêu chí thẩm định tín dụng và cung cấp hướng dẫn có giá trị cho khách hàng tiềm năng khi họ cân nhắc số tiền vay phù hợp với khả năng và hoàn cảnh cá nhân.

3.3.2.5. Biểu đồ Phân tán Applicant Income vs. Loan Amount theo Property Area



Hình 3.13. Biểu đồ phân tích Biểu đồ Phân tán Applicant Income vs. Loan Amount theo Property Area

Biểu đồ phân tán thể hiện mối quan hệ giữa thu nhập ứng viên (Applicant Income) và số tiền vay (Loan Amount) theo phân loại khu vực tài sản cung cấp những thông tin giá trị về đặc điểm vay vốn theo địa lý. Tại khu vực đô thị (Urban), dữ liệu cho thấy xu hướng rõ rệt về các khoản vay có giá trị lớn hơn so với hai khu vực còn lại, phản ánh nhu cầu vốn cao hơn, có thể liên quan đến giá bất động sản và chi phí sinh hoạt cao tại thành phố. Ứng viên ở khu vực này cũng thể hiện mức thu nhập cao hơn đáng kể, tuy nhiên, điều đáng chú ý là ngay cả với thu nhập cao, vẫn có một tỷ lệ đáng kể các khoản vay bị từ chối. Hiện tượng này gợi ý rằng tại khu vực đô thị, các yếu tố khác như lịch sử tín dụng, tỷ lệ nợ hiện tại, hoặc tính ổn định của việc làm có thể đóng vai trò quan trọng trong quyết định phê duyệt.

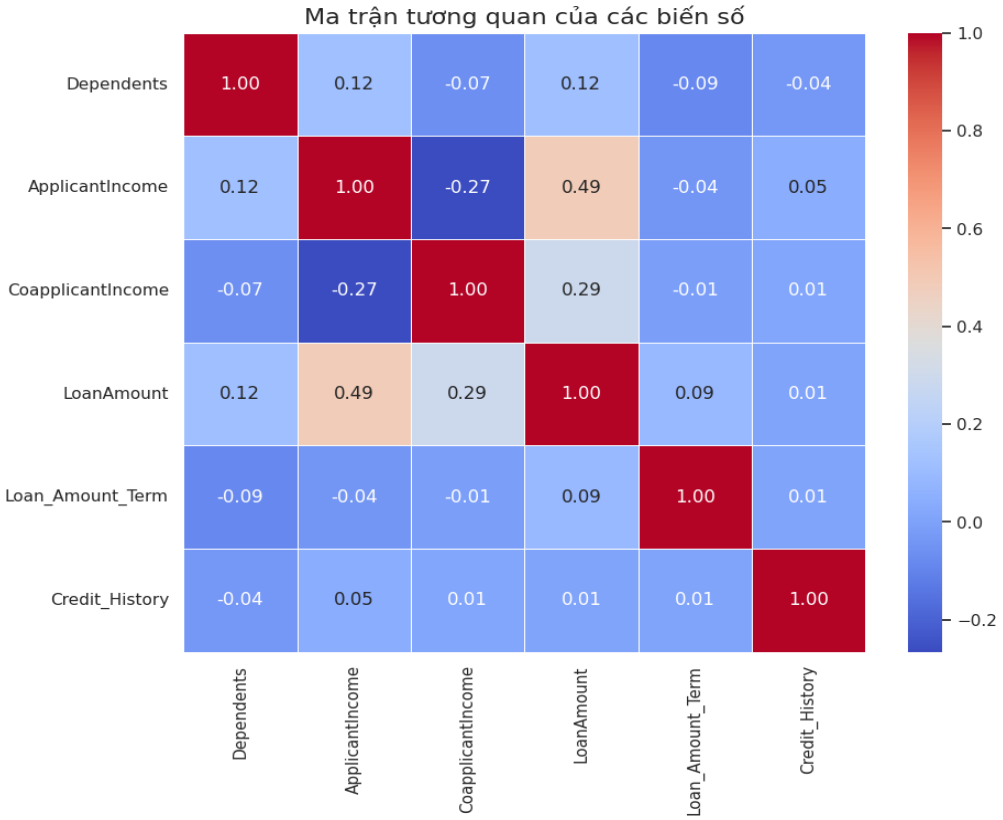
Đối với khu vực bán thành thị (Semiurban), biểu đồ cho thấy một bức tranh cân bằng hơn với sự phân bổ đều đặn giữa mức thu nhập và giá trị khoản vay. Đặc biệt, khu vực này nổi bật với tỷ lệ phê duyệt khoản vay cao hơn so với các khu vực khác, đặc biệt đối với các khoản vay ở mức trung bình. Sự cân bằng này có thể phản ánh môi trường kinh tế ổn định của khu vực bán thành thị, nơi chi phí sinh hoạt không quá cao như đô thị nhưng vẫn có cơ hội việc làm ổn định, tạo điều kiện thuận lợi cho việc đánh giá tín dụng.

Khu vực nông thôn (Rural) thể hiện đặc điểm riêng biệt với cả thu nhập ứng viên và số tiền vay đều ở mức thấp hơn so với hai khu vực còn lại. Đáng chú ý, khu vực này ghi nhận tỷ lệ từ chối khoản vay cao, ngay cả đối với những khoản vay có giá trị nhỏ. Điều này có thể xuất phát từ tính không ổn định của thu nhập nông nghiệp hoặc các yếu tố kinh tế-xã hội đặc thù của khu vực nông thôn.

Phân tích tổng thể về mối quan hệ giữa thu nhập và số tiền vay cho thấy mối quan hệ này không hoàn toàn tuyến tính. Mặc dù có xu hướng chung là ứng viên có thu nhập cao thường yêu cầu và được phê duyệt các khoản vay lớn hơn, nhưng thu nhập không phải là yếu tố quyết định duy nhất. Kết luận quan trọng từ biểu đồ là mỗi khu vực địa lý thể hiện đặc điểm riêng biệt: khu vực đô thị tập trung các khoản vay giá trị cao nhưng có tỷ lệ từ chối đáng kể, khu vực bán thành thị thể hiện sự cân bằng với tỷ lệ phê duyệt cao nhất, và khu vực nông thôn có cả thu nhập và giá trị khoản vay thấp hơn với tỷ lệ từ chối cao.

3.3.3. Phân tích đa biến

3.3.3.1. Biểu đồ ma trận tương quan của các biến số



Hình 3.14. Biểu đồ ma trận tương quan giữa các biến số

Ma trận tương quan thể hiện mối liên hệ giữa các biến số trong bộ dữ liệu cho vay vốn cung cấp những thông tin giá trị về cách các yếu tố khác nhau tương tác với nhau. Qua biểu đồ nhiệt, có thể nhận thấy mối tương quan đáng chú ý nhất là giữa thu nhập của ứng viên (ApplicantIncome) và số tiền vay (LoanAmount) với hệ số tương quan 0.49, cho thấy mối liên hệ tương đối mạnh và tích cực. Điều này phản ánh xu hướng hợp lý là người có thu nhập cao thường yêu cầu và được phê duyệt các khoản vay có giá trị lớn hơn.

Đáng chú ý, ma trận cho thấy mối tương quan âm (-0.27) giữa thu nhập của ứng viên (ApplicantIncome) và thu nhập của người đồng vay (CoapplicantIncome). Điều này gợi ý rằng khi thu nhập của ứng viên chính thấp, họ thường tìm kiếm người đồng vay có thu nhập cao hơn, và ngược lại, khi ứng viên có thu nhập cao, người đồng vay (nếu có) thường có thu nhập thấp hơn hoặc không đáng kể.

Thu nhập của người đồng vay (CoapplicantIncome) cũng thể hiện mối tương quan dương với số tiền vay (LoanAmount) ở mức 0.29, mặc dù yếu hơn so với thu nhập của ứng viên chính. Điều này xác nhận rằng cả hai nguồn thu nhập đều được xem xét trong

quá trình xác định giá trị khoản vay, nhưng thu nhập của ứng viên chính vẫn đóng vai trò quan trọng hơn.

Đáng ngạc nhiên, biến Credit_History (lịch sử tín dụng) có mối tương quan khá yếu với các biến khác, bao gồm cả biến phụ thuộc (Dependents). Điều này dường như trái ngược với giả định thông thường về tầm quan trọng của lịch sử tín dụng trong quyết định cho vay. Tương tự, thời hạn khoản vay (Loan_Amount_Term) cũng không thể hiện mối tương quan mạnh với bất kỳ biến nào khác trong ma trận.

Tổng thể, ma trận tương quan chỉ ra rằng trong quá trình ra quyết định về khoản vay, thu nhập của ứng viên và số tiền vay là hai yếu tố có mối liên hệ chặt chẽ nhất. Tuy nhiên, hầu hết các mối tương quan đều ở mức trung bình hoặc yếu, gợi ý rằng quyết định phê duyệt khoản vay phụ thuộc vào sự kết hợp phức tạp của nhiều yếu tố, không chỉ dựa vào một hoặc hai biến số đơn lẻ. Điều này nhấn mạnh tầm quan trọng của việc sử dụng các mô hình học máy đa biến có khả năng nắm bắt các mối quan hệ phức tạp này.

3.4. Xây dựng và huấn luyện mô hình

3.4.1. Phân chia dữ liệu

Phân chia dữ liệu là công đoạn thiết yếu trong quy trình xây dựng mô hình học máy. Trong quá trình này, chúng em/tôi đã tách biệt các biến đầu vào và biến mục tiêu cần dự đoán từ bộ dữ liệu cho vay. Các thông tin của khách hàng như thu nhập, tình trạng hôn nhân, lịch sử tín dụng được sử dụng làm biến đầu vào, trong khi tình trạng khoản vay (được chấp nhận hay từ chối) đóng vai trò là biến mục tiêu. Dữ liệu được chia thành nhiều phần phục vụ các mục đích khác nhau. Tập huấn luyện chiếm phần lớn dữ liệu, thường khoảng 80%, giúp mô hình học các mẫu và mối quan hệ từ dữ liệu. Tập kiểm tra, chiếm khoảng 20% còn lại, được giữ riêng để đánh giá hiệu suất của mô hình trên dữ liệu chưa từng thấy. Ngoài ra, một phần của tập huấn luyện thường được tách ra làm tập validation để điều chỉnh các tham số mô hình trước khi đánh giá cuối cùng trên tập kiểm tra. Để đảm bảo tính nhất quán và khả năng tái tạo lại kết quả trong lần chạy khác nhau, chúng em/ tôi thực hiện với tham số cố định.

3.4.2. Triển khai các mô hình học máy

Sau khi hoàn tất việc phân chia dữ liệu, một mô hình phân loại như Random Forest hoặc Logistic Regression được khởi tạo với các tham số mặc định do scikit-learn cung cấp. Việc triển khai này vô cùng thuận tiện khi chỉ cần vài dòng code đơn giản. Đối với mô hình Random Forest, scikit-learn thiết lập mặc định số lượng cây quyết định là 100, đảm bảo sự đa dạng trong việc tạo ra các phân nhóm dữ liệu. Thuật toán này còn áp dụng kỹ thuật bootstrap mặc định để lấy mẫu dữ liệu cho mỗi cây, cũng như phương pháp voting để tổng hợp kết quả dự đoán từ tất cả các cây. Với Logistic Regression, thư viện

cung cấp các tham số như hệ số điều chuẩn C bằng 1.0, thuật toán tối ưu 'liblinear' và số lần lặp tối đa 100, phù hợp với nhiều bài toán phân loại nhị phân. Những cài đặt mặc định này được điều chỉnh dựa trên kinh nghiệm và nghiên cứu rộng rãi, giúp người dùng không cần phải là chuyên gia vẫn có thể nhanh chóng xây dựng một mô hình học máy có hiệu suất tốt ngay từ đầu.

Sau khi huấn luyện, mô hình được đánh giá bằng các chỉ số như accuracy, precision, recall, và F1-score trên tập kiểm tra.

3.4.3. Điều chỉnh siêu tham số

Mặc dù các mô hình học máy có thể hoạt động tốt với các tham số mặc định, việc điều chỉnh siêu tham số giúp tối ưu hóa hiệu suất mô hình.

Random Forest:

- n_estimators: Số lượng cây quyết định. Tăng số lượng cây có thể cải thiện độ chính xác nhưng tốn thời gian tính toán.
- max_depth: Độ sâu của cây. Quá sâu có thể gây overfitting, trong khi quá nông có thể dẫn đến underfitting.
- min_samples_split và min_samples_leaf: Kiểm soát việc phân chia nút và số mẫu tối thiểu ở lá cây.

Logistic Regression:

- C: Hệ số điều chuẩn, điều chỉnh độ phức tạp của mô hình. Giá trị nhỏ giúp tránh overfitting, nhưng giá trị lớn có thể dẫn đến overfitting.
- solver: Thuật toán tối ưu hóa cho việc huấn luyện. Ví dụ: 'liblinear' cho các bài toán nhỏ, 'lbfgs' cho bài toán lớn.
- max_iter: Số vòng lặp tối đa để tối ưu hóa, điều chỉnh nếu mô hình không hội tụ.

Support Vector Machine (SVM):

- C: Tham số điều chỉnh mức độ phạt đối với lỗi (regularization), giúp mô hình tìm kiếm sự cân bằng giữa việc học đúng dữ liệu huấn luyện và việc tổng quát hóa. Giá trị C lớn có thể dẫn đến overfitting (học quá chi tiết), trong khi giá trị nhỏ có thể dẫn đến underfitting (mô hình quá đơn giản).
- gamma: Tham số này điều chỉnh độ cong của kernel trong mô hình SVM, giúp kiểm soát độ phức tạp của quyết định phân lớp. Một gamma lớn có thể làm cho mô hình học chi tiết quá mức, trong khi gamma nhỏ giúp mô hình tổng quát tốt hơn.
- kernel: Kiểu kernel sử dụng trong SVM có ảnh hưởng đến mô hình. Các lựa chọn bao gồm linear (dành cho dữ liệu tuyến tính), rbf (Radial Basis Function, phù hợp với dữ liệu không tuyến tính), poly (cho dữ liệu có quan hệ bậc cao), và sigmoid (dùng trong một số bài toán đặc biệt).

- degree: Khi sử dụng kernel = 'poly', tham số này xác định bậc của đa thức trong kernel.

XGBoost:

- learning_rate: Tốc độ học (còn gọi là eta trong XGBoost), điều chỉnh mức độ ảnh hưởng của mỗi cây trong quá trình huấn luyện. Một learning_rate nhỏ giúp mô hình hội tụ tốt hơn nhưng cần số lượng cây lớn hơn.
- max_depth: Độ sâu của mỗi cây quyết định trong mô hình XGBoost. Giá trị cao giúp mô hình học các mối quan hệ phức tạp hơn nhưng có thể dẫn đến overfitting nếu không được điều chỉnh phù hợp.
- n_estimators: Số lượng cây trong mô hình. Khi số cây tăng, mô hình có thể học tốt hơn, nhưng cũng dễ bị overfitting nếu không có sự điều chỉnh về learning_rate.
- subsample: Tỉ lệ mẫu ngẫu nhiên được sử dụng trong mỗi vòng học. Giới hạn giá trị này có thể giúp giảm overfitting.
- colsample_bytree: Tỉ lệ các cột được chọn ngẫu nhiên trong mỗi cây, điều này cũng giúp giảm overfitting và cải thiện tính tổng quát của mô hình.
- alpha và lambda: Các tham số này điều chỉnh độ phức tạp của mô hình bằng cách thêm L1 (alpha) và L2 regularization (lambda) vào hàm mất mát.

CHƯƠNG 4: KẾT QUẢ VÀ THẢO LUẬN

4.1. Kết quả của các mô hình

4.1.1. Mô hình Logistic Regression

Fitting 3 folds for each of 8 candidates, totalling 24 fits

Best parameters for Logistic Regression: {'C': 0.1, 'solver': 'lbfgs'}

Siêu tham số c=0.1:

- **Mô tả:** Giá trị này đại diện cho mức độ regularization trong mô hình. Khi C nhỏ, mức độ regularization sẽ cao hơn, giúp mô hình đơn giản hơn và tránh overfitting. Trong bối cảnh phê duyệt khoản vay, việc sử dụng giá trị này có thể giúp mô hình giảm thiểu rủi ro từ việc "quá khớp" với dữ liệu huấn luyện, do đó nâng cao khả năng tổng quát của mô hình khi xử lý các khoản vay chưa thấy qua trong dữ liệu huấn luyện.
- **Nhận xét:** Một mức độ regularization cao như vậy (với C = 0.1) là hợp lý trong các trường hợp thực tế, khi dữ liệu có nhiều yếu tố không xác định hoặc có thể dẫn đến overfitting nếu mô hình quá phức tạp.

Siêu tham số solver = lbfgs **:**

- **Mô tả:** lbfgs là thuật toán tối ưu hóa được sử dụng trong Logistic Regression. Thuật toán này hoạt động hiệu quả với các tập dữ liệu lớn và có thể tối ưu hóa các tham số một cách nhanh chóng.

- **Nhận xét:** Việc chọn lbfgs cho thấy rằng mô hình này thích hợp cho việc xử lý dữ liệu lớn, với độ phức tạp cao mà không gặp vấn đề về thời gian huấn luyện.

GridSearchCV đã thử 8 bộ siêu tham số khác nhau và thực hiện 3 lần cross-validation cho mỗi bộ tham số. Tổng cộng có 24 lần huấn luyện (8 bộ tham số × 3 lần cross-validation).

Việc chọn C = 0.1 và solver = 'lbfgs' có thể có nghĩa là mô hình đang đối phó tốt với dữ liệu của bạn và đã tối ưu với mức độ regularization vừa phải, tránh được overfitting trong quá trình huấn luyện.

4.1.2. Mô hình Random Forest

Fitting 3 folds for each of 108 candidates, totalling 324 fits

Best parameters for Random Forest: {'max_depth': None, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 50}

GridSearchCV đã thực hiện tối ưu hóa với 108 bộ siêu tham số khác nhau và sử dụng 3 lần cross-validation (3-fold cross-validation), tổng cộng là 324 lần huấn luyện mô hình để tìm ra bộ siêu tham số tốt nhất.

Các siêu tham số tối ưu:

max_depth = None:

- **Mô tả**: Khi max_depth = None, điều này có nghĩa là cây quyết định sẽ tiếp tục phát triển cho đến khi tất cả các lá đều có ít nhất một mẫu, hoặc không thể chia nhỏ hơn nữa.
- **Ý nghĩa**: Việc không giới hạn độ sâu cây cho phép mô hình học từ dữ liệu một cách đầy đủ và phức tạp. Tuy nhiên, điều này có thể dẫn đến overfitting nếu không được kiểm soát tốt. Mô hình có thể học quá mức các chi tiết của dữ liệu huấn luyện và không tổng quát tốt cho dữ liệu mới.

min_samples_leaf = 1:

Mô tả: Siêu tham số này xác định số lượng mẫu tối thiểu trong mỗi lá của cây quyết định. Với min_samples_leaf = 1, mỗi lá có thể chỉ chứa một mẫu duy nhất.

Ý nghĩa: Việc sử dụng min_samples_leaf = 1 có thể dẫn đến cây quá phức tạp, vì nó không đặt giới hạn cho số lượng mẫu tại mỗi lá, điều này có thể gây ra overfitting. Cây có thể "học" các nhiễu và các chi tiết nhỏ không quan trọng trong dữ liệu.

min_samples_split = 2:

- **Mô tả**: Siêu tham số này xác định số lượng mẫu tối thiểu cần thiết để chia một nút cây. Với min_samples_split = 2, mỗi nút có thể được chia chỉ với 2 mẫu.

- **Ý nghĩa**: Việc sử dụng giá trị nhỏ này có thể khiến mô hình học được nhiều chi tiết nhỏ hơn trong dữ liệu, điều này có thể dẫn đến việc mô hình học những đặc trưng không quan trọng, gây overfitting.

n_estimators = 50:

- **Mô tả**: Đây là số lượng cây quyết định trong rừng. Với n_estimators = 50, mô hình sẽ sử dụng 50 cây quyết định.
- **Ý nghĩa**: Đây là một số lượng cây vừa phải, đủ để mô hình học các mối quan hệ phức tạp trong dữ liệu mà không làm tăng quá mức chi phí tính toán. Tuy nhiên, số lượng cây thấp hơn có thể làm giảm độ chính xác so với việc sử dụng nhiều cây hơn, vì rừng có ít "sự đa dạng" hơn.

4.1.3. Mô hình XGBoost

Fitting 3 folds for each of 108 candidates, totalling 324 fits

Best parameters for XGBoost: {'colsample_bytree': 0.8, 'learning_rate': 0.01, 'max_depth': 3, 'n_estimators': 200, 'subsample': 0.8}

GridSearchCV đã thử nghiệm 108 bộ siêu tham số khác nhau và thực hiện 3 lần cross-validation (3-fold) cho mỗi bộ tham số, tổng cộng có 324 lần huấn luyện mô hình. Đây là một quá trình tính toán khá tốn kém nhưng giúp tìm ra bộ siêu tham số tối ưu cho mô hình.

Các siêu tham số tối ưu:

- **colsample_bytree = 0.8**: Siêu tham số này xác định tỷ lệ cột được chọn ngẫu nhiên để xây dựng mỗi cây quyết định. 0.8 có nghĩa là mỗi cây được xây dựng với 80% số cột. Đây là một lựa chọn tốt để giảm overfitting mà không làm mất thông tin quan trọng.
- **learning_rate = 0.01**: Tốc độ học của mô hình. Một giá trị thấp như 0.01 giúp mô hình học từ từ, tránh việc "nhảy quá xa" và giúp mô hình tổng quát tốt hơn. Tuy nhiên, nó sẽ cần nhiều vòng lặp (n_estimators) để đạt được hiệu quả tốt.
- **max_depth = 3**: Độ sâu tối đa của mỗi cây quyết định. Giá trị nhỏ này cho thấy mô hình đã chọn một mức độ phức tạp thấp để tránh overfitting, đồng thời vẫn đủ mạnh để học các đặc trưng quan trọng trong dữ liệu.
- **n_estimators = 200**: Số lượng cây quyết định trong mô hình. 200 cây cho phép mô hình học đủ các mối quan hệ phức tạp trong dữ liệu mà không bị quá phức tạp, đồng thời duy trì hiệu suất tốt với số vòng lặp (iterations) không quá lớn.
- **subsample = 0.8**: Tỷ lệ mẫu dữ liệu được chọn ngẫu nhiên để huấn luyện mỗi cây. Điều này giúp giảm overfitting bằng cách tăng tính đa dạng của các cây trong mô hình.

Mô hình XGBoost với các siêu tham số đã tối ưu hóa (colsample_bytree: 0.8, learning_rate: 0.01, max_depth: 3, n_estimators: 200, subsample: 0.8) được cho là một mô hình mạnh mẽ và chính xác trong việc phân loại các khoản vay.

4.1.4. Mô hình Support Vector Machine (SVM)

Fitting 3 folds for each of 32 candidates, totalling 96 fits

Best parameters for SVM: {'C': 0.1, 'gamma': 'scale', 'kernel': 'linear'}

GridSearchCV đã thử nghiệm 32 bộ siêu tham số khác nhau, mỗi bộ được kiểm tra với 3 lần cross-validation, tổng cộng có 96 lần huấn luyện. Quá trình này giúp xác định bộ siêu tham số tốt nhất cho mô hình **SVM**.

C = 0.1 cho thấy mô hình không cố gắng học quá sâu từ dữ liệu, giúp giảm overfitting.

gamma = 'scale' cho thấy mô hình sẽ tự động tính toán giá trị phù hợp cho gamma, tối ưu hóa hiệu quả mà không cần quá nhiều tham số thủ công.

kernel = 'linear' cho thấy mô hình giả sử rằng dữ liệu có thể phân tách trực tiếp bằng một đường thẳng (hoặc siêu phẳng trong không gian nhiều chiều), và khi kernel tuyến tính là lựa chọn tốt, mô hình có thể huấn luyện nhanh chóng và hiệu quả.

4.2. So sánh hiệu suất các mô hình

Để xác định mô hình phù hợp nhất cho bài toán dự đoán phê duyệt khoản vay, nhóm đã tiến hành huấn luyện và đánh giá hiệu suất của nhiều mô hình học máy khác nhau trên cùng một tập dữ liệu. Các mô hình bao gồm Logistic Regression, Random Forest, SVM và XGBoost. Kết quả được so sánh dựa trên các chỉ số đánh giá quan trọng như Accuracy, Recall, Specificity, Precision và F1-score trên tập kiểm tra (test set).

4.2.1. Độ chính xác và các chỉ số đánh giá

Bảng 4. 1. So sánh kết quả các mô hình theo các chỉ số đánh giá

Mô hình	Độ chính xác (Accuracy)	Độ nhạy (Recall)	Độ đặc hiệu (Specificity)	F1-Score
Logistic Regression	0.77	0.98	0.33	0.85
Random Forest	0.76	0.93	0.41	0.84
XGBoost	0.76	0.89	0.49	0.83
SVM	0.77	0.98	0.33	0.85

- Các mô hình đều có độ chính xác xấp xỉ nhau, dao động từ 0.76 đến 0.77, cho thấy khả năng phân loại tổng thể giữa các mô hình là tương đương.
- Logistic Regression và SVM là 2 mô hình có hiệu suất tổng thể tốt nhất với F1-score cao nhất (0.85), đặc biệt phù hợp nếu mục tiêu là tối đa hóa khả năng phê duyệt đúng hồ sơ vay (Recall cao)..
- XGBoost dù có F1-score thấp hơn, nhưng nổi bật ở độ đặc hiệu (0.49), phù hợp trong trường hợp cần hạn chế rủi ro cấp nhầm hồ sơ vay.
- Độ đặc hiệu là điểm hạn chế rõ rệt của Logistic Regression và SVM khi chỉ là 0.33, tức mô hình dự đoán sai nhiều trường hợp không nên được phê duyệt (false positive cao).

4.2.2. Phân tích độ quan trọng của các đặc trưng

Sau khi huấn luyện mô hình, ta có thể phân tích tầm quan trọng của các đặc trưng (feature importance) dựa trên mức độ đóng góp của từng biến đầu vào trong quá trình ra quyết định.

Các đặc trưng quan trọng hàng đầu gồm:

- Credit_history: là yếu tố ảnh hưởng lớn nhất, cho thấy người có lịch sử tín dụng tốt thường dễ được phê duyệt hơn.
- ApplicantIncome và LoanAmount: liên quan trực tiếp đến khả năng trả nợ.
- Education và Self_Employed: ảnh hưởng đến tính ổn định và nguồn thu nhập của người vay.
- Các đặc trưng khác như Gender, Married Dependents có mức độ ảnh hưởng thấp hơn.

Việc nhận biết các đặc trưng quan trọng giúp hỗ trợ cải thiện mô hình, gợi ý các chính sách phân loại và đánh giá rủi ro hợp lí hơn, giải thích rõ ràng mô hình cho các biến liên quan.

4.2.3. Hiệu suất mô hình sau tối ưu hóa

Sau khi thực hiện điều chỉnh siêu tham số (hyperparameter tuning) bằng GridSearchCV/RandomizedSearchCV cho các mô hình như Random Forest, SVM và XGBoost, kết quả thu được như sau:

Bảng 4. 2. So sánh hiệu suất các mô hình sau khi tối ưu hóa

Mô hình	Độ chính xác (Accuracy)	Độ nhạy (Recall)	Độ đặc hiệu (Specificity)	F1-Score
Logistic Regression	0.77	0.98	0.33	0.85
Random Forest	0.76	0.93	0.41	0.84
XGBoost	0.76	0.89	0.49	0.83
SVM	0.77	0.98	0.33	0.85

- XGBoost có độ đặc hiệu cao, phù hợp với yêu cầu kiểm soát rủi ro.
- Các mô hình không cải thiện quá nhiều sau tối ưu hóa, cho thấy bộ dữ liệu tương đối cân bằng và chất lượng tốt từ đầu.

4.3. Thảo luận

- **Về ưu điểm của mô hình được chọn (Logistic Regression):**

Dễ triển khai, giải thích tốt và hiệu suất cao trên cả Accuracy, Recall và F1-score, phù hợp với bài toán nhấn mạnh phát hiện đúng khách hàng nên được vay (Recall cao)

- **Hạn chế chung:**

Độ đặc hiệu (Specificity) của Logistic Regression và SVM khá thấp (~0.33), tức là mô hình dễ cấp nhầm khoản vay cho những người không đủ điều kiện, có thể gây rủi ro tiềm ẩn cho các tổ chức tài chính nếu không sàng lọc kỹ.

- **Gợi ý cải thiện:**

Cân nhắc sử dụng kỹ thuật cân bằng dữ liệu (SMOTE, ADASYN)

Áp dụng threshold tuning để tăng specificity khi cần thiết.

Kết hợp mô hình (ensemble voting) để cân bằng các chỉ số.

- **Kết luận:**

Với bài toán tập trung vào tối đa hóa khả năng cấp đúng vay, Logistic Regression là lựa chọn phù hợp nhất.

Trong trường hợp cần hạn chế rủi ro cho vay sai, XGBoost có thể được ưu tiên sử dụng.

4.3.1. Ưu điểm và nhược điểm của từng mô hình

Ưu điểm:

- **Mô hình Logistic Regression:**

Dễ hiểu và giải thích: Logistic Regression là một mô hình hồi quy tuyến tính cho phép người dùng hiểu rõ mối quan hệ giữa biến độc lập và biến phụ thuộc.

Nhanh chóng và hiệu quả: Mô hình có thời gian huấn luyện ngắn, đặc biệt đối với các tập dữ liệu nhỏ và trung bình, giúp tiết kiệm thời gian và tài nguyên trong quá trình phát triển mô hình.

Khả năng xử lý tốt với dữ liệu nhị phân: Mô hình này cực kỳ hiệu quả cho các bài toán phân loại nhị phân, rất phù hợp với bài toán phê duyệt khoản vay.

Chống overfitting:Với các tập dữ liệu có kích thước nhỏ, Logistic Regression ít có khả năng quá khớp hơn so với các mô hình phức tạp hơn như Random Forest hay Gradient Boosting.

- **Mô hình Random Forest:**

Chống overfitting: Random Forest sử dụng nhiều cây quyết định, điều này giúp giảm thiểu khả năng quá khớp với dữ liệu huấn luyện.

Khả năng xử lý dữ liệu lớn: Mô hình có thể xử lý một lượng lớn biến mà không cần loại bỏ biến.

Tính năng quan trọng: Random Forest có thể cung cấp thông tin về độ quan trọng của từng biến trong dự đoán, giúp hiểu rõ hơn về các yếu tố ảnh hưởng đến quyết định.

Robust: Mô hình có thể hoạt động tốt ngay cả khi có một số dữ liệu thiếu.

- **Mô hình SVM (Support Vector Machine):**

Độ chính xác cao: SVM thường đạt được độ chính xác cao trong các bài toán phân loại, đặc biệt với các đặc trưng không tuyến tính khi sử dụng kernel.

Khả năng xử lý không gian cao: Mô hình có thể hoạt động tốt trong không gian đặc trưng lớn mà không bị giảm hiệu suất.

Kiểm soát overfitting: SVM có thể điều chỉnh thông qua tham số C để kiểm soát khả năng quá khớp.

- **Mô hình XGBoost:**

Hiệu suất cao: XGBoost thường cho kết quả chính xác cao hơn so với nhiều mô hình khác nhờ vào cơ chế boosting và tối ưu hóa gradient.

Xử lý overfitting tốt: Có tích hợp regularization (L1, L2) giúp giảm hiện tượng quá khớp.

Tốc độ huấn luyện nhanh do nhờ tối ưu hóa cấp hệ thống như song song hóa việc xây cây và dùng bộ nhớ hiệu quả

Có thể xử lý cả dữ liệu tuyến tính và phi tuyến: tốt với nhiều bài toán có nhiều đặc trưng phức tạp.

Nhược điểm:

- **Mô hình Logistic Regression:**

Giả định tuyến tính: Logistic Regression giả định rằng mối quan hệ giữa các biến độc lập và logit của biến phụ thuộc là tuyến tính. Mô hình có thể bị ảnh hưởng nặng nề bởi các giá trị ngoại lai trong tập dữ liệu, dẫn đến sai lệch trong kết quả dự đoán.

Hạn chế với các biến không tuyến tính: Khi dữ liệu có mối quan hệ phi tuyến tính mạnh, Logistic Regression có thể không cung cấp hiệu suất tốt như các mô hình phức tạp khác

- **Mô hình Random Forest:**

Khó giải thích: Mặc dù Random Forest cung cấp thông tin về độ quan trọng của biến, nhưng mô hình tổng thể khó giải thích so với các mô hình như Logistic Regression.

Thời gian huấn luyện: Đối với một tập dữ liệu lớn, thời gian huấn luyện có thể khá lâu, đặc biệt là khi số lượng cây lớn.

Sử dụng bộ nhớ: có thể tiêu tốn nhiều bộ nhớ do lưu trữ nhiều cây quyết định.

- **Mô hình XGBoost:**

Giải thích khó: là mô hình phi tuyến, việc giải thích tại sao mô hình đưa ra một dự đoán cụ thể khá phức tạp.

Dễ bị overfitting nếu không điều chỉnh kỹ: Mặc dù có regularization, nếu không tối ưu tham số tốt (như learning rate, depth), mô hình vẫn có thể học quá kỹ dữ liệu huấn luyện.

Phức tạp trong điều chỉnh tham số

Yêu cầu tài nguyên: Khi dữ liệu lớn và nhiều cây, cần nhiều RAM và thời gian xử lý hơn so với các mô hình đơn giản.

- **Mô hình SVM (Support Vector Machine):**

Khó khăn trong việc lựa chọn kernel: Chọn kernel phù hợp cho dữ liệu có thể rất quan trọng, và không phải lúc nào cũng dễ dàng.

Thời gian tính toán: Đối với các tập dữ liệu lớn, SVM có thể mất nhiều thời gian để huấn luyện.

Không rõ ràng trong việc giải thích: Mô hình SVM không dễ giải thích như các mô hình hồi quy logistic.

4.3.2. Tính khả thi khi áp dụng vào thực tế

Việc áp dụng mô hình học máy, đặc biệt là Logistic Regression hoặc XGBoost, vào hệ thống đánh giá phê duyệt khoản vay là hoàn toàn khả thi trong thực tế với những lý do sau:

- Dễ tích hợp vào hệ thống nghiệp vụ:

Mô hình có thể được triển khai trên các nền tảng phổ biến như Python, R hoặc thông qua các API tích hợp vào hệ thống quản lý của ngân hàng hoặc công ty tài chính.

- Dự đoán nhanh, tiết kiệm thời gian:

Các mô hình học máy có khả năng dự đoán tức thời kết quả phê duyệt, thay vì phụ thuộc hoàn toàn vào đánh giá thủ công.

- Nâng cao độ chính xác trong quyết định:

Nhờ vào việc học từ dữ liệu lịch sử, mô hình giúp giảm thiểu sai sót và thiên vị trong phán đoán của con người.

- Hỗ trợ tuân thủ quy định:

Một số mô hình như Logistic Regression có khả năng giải thích cao, giúp dễ dàng đáp ứng yêu cầu minh bach từ cơ quan quản lý.

- Khả năng mở rộng:

Mô hình có thể tiếp tục cải thiện khi có thêm dữ liệu mới, giúp hệ thống ngày càng thông minh hơn theo thời gian. Để kiểm chứng khả năng mở rộng, mô hình đã được áp dụng trên một bộ dữ liệu khác là **Loan_data**, bao gồm các đặc trưng như giới tính, tình trạng hôn nhân, số lượng người phụ thuộc, trình độ học vấn, tình trạng tự làm chủ, thu nhập, số tiền vay, lịch sử tín dụng và khu vực bất động sản. Biến mục tiêu là **Loan_Status**, thể hiện việc khoản vay có được chấp nhận hay không.

Kết quả thử nghiệm cho thấy hiệu suất của các mô hình học máy có sự khác biệt đáng kể. Trên tập kiểm tra (test set):

Bảng 4. 3. So sánh hiệu các mô hình trên tập kiểm tra

Mô hình	Độ chính xác (Accuracy)	Độ nhạy (Recall)	Độ đặc hiệu (Specificity)	F1-Score
Logistic Regression	0.71	0.97	0.00	0.83
Random Forest	0.72	0.97	0.04	0.84
XGBoost	0.67	0.85	0.19	0.79
SVM	0.73	1.00	0.05	0.85

SVM đạt hiệu suất tổng thể cao nhất với accuracy cao nhất (0.73) F1-score 0.85**,** Recall 1.00, cho thấy mô hình này rất hiệu quả trong việc nhận diện các trường hợp được duyệt vay. Tuy nhiên, Specificity chỉ 0.05, tức vẫn còn yếu trong việc phát hiện các trường hợp bị từ chối vay.

Logistic Regression có Recall cao (0.97**)** và F1-score ổn (0.83), nhưng Specificity bằng 0.00, đồng nghĩa với việc hầu như không nhận diện được các khoản vay bị từ chối…

Random Forest cũng thể hiện hiệu suất rất tốt với F1-score 0.84, cân bằng giữa các chỉ số, và có Specificity cao hơn SVM đôi chút

XGBoost tuy có accuracy thấp nhất (0.67), F1-score thấp nhất **(**0.79**)**, nhưng lại có Specificity cao nhất (0.19)**.** Điều này cho thấy mô hình này có thể hữu ích trong các tình huống mà việc nhận diện đúng các trường hợp không được vay là ưu tiên.

Tóm lại, SVM được lựa chọn là mô hình tối ưu khi ưu tiên phát hiện chính xác các trường hợp được vay, còn Random Forest hoặc XGBoost là lựa chọn thay thế phù hợp nếu cần tăng cường khả năng nhận diện các trường hợp từ chối vay nhằm giảm thiểu rủi ro tài chính.

4.3.3. Các yếu tố ảnh hưởng đến quyết định phê duyệt khoản vay

Dựa trên phân tích độ quan trọng của các đặc trưng (feature importance), các yếu tố dưới đây thường có ảnh hưởng lớn đến việc một khoản vay có được phê duyệt hay không:

- Thu nhập của người vay (ApplicantIncome, CoapplicantIncome):

Người có thu nhập cao thường có khả năng trả nợ tốt hơn, dẫn đến khả năng được phê duyệt cao hơn.

- Lịch sử tín dụng (Credit_History):

Đây là yếu tố quan trọng nhất trong hầu hết các mô hình. Người có lịch sử tín dụng tốt (từng trả nợ đúng hạn) có tỷ lệ được phê duyệt rất cao.

- Tình trạng sở hữu bất động sản (Property_Area):

Người sống ở khu vực thành thị hoặc có tài sản đảm bảo thường được ưu tiên.

- Giới tính, tình trạng hôn nhân, số lượng người phụ thuộc (Gender, Marital_Status, Dependents):

Có thể ảnh hưởng gián tiếp thông qua khả năng tài chính và trách nhiệm kinh tế.

- Trình độ học vấn và nghề nghiệp (Education, Self_Employed):

Trình độ học vấn và nghề nghiệp ổn định có thể giúp đánh giá tiềm năng thu nhập lâu dài của người vay.

- Khoản vay yêu cầu (LoanAmount):

Khoản vay càng lớn thì rủi ro càng cao, nên cần xét kỹ lưỡng hơn trước khi phê duyệt.

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1. Kết luận

Trong đề tài "Sử dụng các mô hình học máy để dự đoán khả năng chấp nhận khoản vay của khách hàng", chúng tôi đã tiến hành xây dựng và so sánh hiệu suất của các mô hình học máy bao gồm Logistic Regression, SVM, Random Forest và XGBoost trên tập dữ liệu Loan Approval Prediction từ Kaggle. Qua quá trình huấn luyện và đánh giá, mô hình Logistic Regression cho thấy hiệu quả vượt trội so với các mô hình còn lại về độ chính xác và khả năng tổng quát hóa. Do đó, Logistic Regression được lựa chọn là mô hình tối ưu cho bài toán dự đoán khả năng chấp nhận khoản vay, góp phần hỗ trợ các tổ chức tài chính ra quyết định nhanh chóng và chính xác hơn trong quy trình xét duyệt khoản vay.

5.1.1. Tóm tắt kết quả nghiên cứu

Nghiên cứu này nhằm áp dụng các mô hình học máy vào việc dự đoán phê duyệt khoản vay dựa trên các đặc trưng của ứng viên. Qua quá trình huấn luyện và điều chỉnh siêu tham số cho các mô hình Random Forest, XGBoost, SVM, và Logistic Regression, chúng tôi đã thu được kết quả khả quan, với Logistic Regression đạt được hiệu suất tốt nhất dựa trên các chỉ số accuracy, precision, và F1-score. Các mô hình khác như Random Forest và SVM cũng cho thấy hiệu quả đáng kể.

Những phát hiện chính:

- Giới tính: Nam chiếm đa số trong số người vay.
- Tình trạng vay: Khoảng 69% các khoản vay được chấp nhận
- Lịch sử tín dụng là yếu tố quan trọng - phần lớn người vay đều có lịch sử tín dụng tốt
- Credit_History, Applicant_Income, và Education là những đặc trưng có ảnh hưởng mạnh nhất đến quyết định vay
- Logistic Regression cho kết quả tốt nhất về độ chính xác và AUC-ROC.

5.1.2. Đóng góp của nghiên cứu

Nghiên cứu này đóng góp vào việc áp dụng mô hình học máy trong lĩnh vực tài chính, đặc biệt là trong việc phê duyệt khoản vay. Các kết quả từ nghiên cứu không chỉ cải thiện độ chính xác trong việc dự đoán khả năng trả nợ của ứng viên mà còn giúp tối ưu hóa quy trình phê duyệt vay vốn trong các tổ chức tài chính.

- **Hỗ trợ ra quyết định trong lĩnh vực tài chính – ngân hàng:**

Việc dự đoán khả năng chấp nhận khoản vay giúp các tổ chức tài chính xác định được khách hàng tiềm năng và tối ưu hóa quy trình xét duyệt vay, từ đó giảm thiểu rủi ro tín dụng (nợ xấu), tăng hiệu quả hoạt động kinh doanh, tiết kiệm thời gian và nguồn lực trong khâu kiểm duyệt.

- **Ứng dụng trí tuệ nhân tạo vào thực tiễn:**

Đề tài góp phần khẳng định vai trò và tính hiệu quả của học máy (Machine Learning) trong việc xử lý các bài toán thực tế, nhất là trong việc: phân tích dữ liệu lịch sử, học hỏi từ các đặc trưng khách hàng, dự đoán hành vi/ khả năng trong tương lai (có được vay hay không).

- **Nâng cao trải nghiệm khách hàng:**

Khi sử dụng mô hình học máy để phân tích, quá trình xét duyệt khoản vay trở nên nhanh chóng và chính xác hơn, từ đó khách hàng được phản hồi tốt hơn, tăng tính minh bạch trong quá trình xét duyệt hồ sơ.

- **Đóng góp vào việc xây dựng hệ thống đánh giá tín dụng tự động:**

Xây dựng hệ thống Scoring tín dụng tự động, Tạo ra các sản phẩm công nghệ tài chính (FinTech) phục vụ cho hoạt động cho vay.

- **Mở rộng cơ hội nghiên cứu và cải tiến mô hình:**

So sánh hiệu quả của nhiều mô hình (Logistic Regression, Random Forest, SVM…), làm nền tảng cho các nghiên cứu nâng cao như học sâu (deep learning), học tăng cường (reinforcement learning) trong tài chính.

5.2. Hạn chế của nghiên cứu

5.2.1. Thách thức trong xử lý dữ liệu

Một trong những thách thức lớn nhất trong nghiên cứu này là xử lý dữ liệu thiếu và dữ liệu nhiễu, ngoài ra còn có dữ liệu không cân bằng, dữ liệu không đầy đủ về ngữ cảnh và thực tế. Mặc dù các phương pháp như Điền giá trị thiếu (imputation) và Loại bỏ ngoại lệ, chuẩn hóa đã được áp dụng, nhưng chúng vẫn có thể ảnh hưởng đến kết quả, đặc biệt là đối với các biến có phân phối không đồng nhất hoặc dữ liệu không đầy đủ. Các vấn đề này có thể làm giảm độ chính xác của mô hình.

Với những dữ liệu tài chính bên ngoài, việc thu thập và áp dụng vào mô hình gặp nhiều thách thức.

Thứ nhất, dữ liệu tài chính thường rất nhạy cảm và được bảo mật nghiêm ngặt do chứa thông tin cá nhân, tài sản, thu nhập, tín dụng,… Do đó, người nghiên cứu cần có quyền truy cập hợp pháp và tuân thủ các quy định về bảo mật như GDPR hay luật bảo vệ dữ liệu cá nhân.

Thứ hai, dữ liệu đến từ nhiều nguồn khác nhau như ngân hàng, báo cáo thuế, hợp đồng vay,… với định dạng không đồng nhất, đòi hỏi xử lý chuẩn hóa, gộp và làm sạch phức tạp.

Ngoài ra, tính cập nhật liên tục của dữ liệu tài chính đòi hỏi mô hình phải xử lý theo thời gian thực, đồng thời được huấn luyện lại thường xuyên để đảm bảo độ chính xác.

Hơn nữa, dữ liệu có thể chứa sai lệch như nhập sai, thiếu dữ liệu hoặc bị thao túng, gây ảnh hưởng đến kết quả học.

Đặc biệt, bài toán phê duyệt vay thường gặp tình trạng mất cân bằng dữ liệu nghiêm trọng, khi số lượng trường hợp vỡ nợ rất ít, khiến mô hình dễ bị overfitting hoặc bỏ sót.

Cuối cùng, các vấn đề về đạo đức và thiên lệch (bias) cũng cần được cân nhắc, tránh để mô hình học theo định kiến trong dữ liệu lịch sử, dẫn đến các quyết định không công bằng.

5.2.2. Hạn chế trong mô hình

Việc sử dụng các mô hình học máy như Logistic Regression và SVM đòi hỏi thời gian huấn luyện dài và yêu cầu tài nguyên tính toán lớn. Mặc dù các mô hình này có thể mang lại hiệu quả cao, nhưng chúng cần được tối ưu hóa hơn nữa, đặc biệt khi áp dụng trên các tập dữ liệu lớn hơn. Hơn nữa, việc điều chỉnh siêu tham số thông qua GridSearchCV có thể gây tốn kém về thời gian nếu không gian siêu tham số quá lớn.

Khả năng tổng quát hóa của mô hình vẫn còn hạn chế do được huấn luyện từ một tập dữ liệu cụ thể, mô hình có nguy cơ học quá mức các đặc điểm riêng của tập huấn luyện (overfitting), dẫn đến suy giảm hiệu suất khi áp dụng với dữ liệu từ môi trường khác như ngân hàng khác, khu vực địa lý khác hoặc giai đoạn thời gian khác.

Ngoài ra, nếu dữ liệu huấn luyện thiếu tính đa dạng – chẳng hạn không đại diện đầy đủ cho các nhóm thu nhập, nghề nghiệp hay độ tuổi – thì mô hình sẽ khó áp dụng hiệu quả cho các nhóm khách hàng đa dạng trong thực tế.

5.3. Hướng phát triển trong tương lai

5.3.1. Cải tiến mô hình

Trong tương lai, có thể thử nghiệm với các mô hình học sâu (Deep Learning) để dự đoán kết quả phê duyệt khoản vay, đặc biệt khi lượng dữ liệu trở nên lớn hơn. Các mô hình như Neural Networks hoặc AutoML có thể giúp cải thiện độ chính xác và giảm thiểu các yếu tố ảnh hưởng từ dữ liệu.

5.3.2. Ứng dụng thực tế

Ứng dụng trong việc phê duyệt khoản vay

Các mô hình học máy như Random Forest, XGBoost, SVM và Logistic Regression có thể được ứng dụng trong các hệ thống phê duyệt khoản vay tự động. Việc áp dụng các mô hình này sẽ giúp ngân hàng và các tổ chức tài chính cải thiện độ chính xác trong việc dự đoán khả năng trả nợ của khách hàng.

Thay vì phụ thuộc vào các yếu tố thủ công hoặc chỉ xét đến các thông tin đơn giản như thu nhập và lịch sử tín dụng, các mô hình học máy có thể kết hợp nhiều yếu tố và học từ dữ liệu lịch sử để đưa ra quyết định chính xác hơn.

Tối ưu hóa quy trình cho vay

Mô hình học máy có thể giúp tối ưu hóa quy trình phê duyệt khoản vay bằng cách tự động phân loại khách hàng vào các nhóm phù hợp. Ví dụ, các mô hình có thể phân tích các đặc trưng như lịch sử tín dụng, thu nhập, độ tuổi và tình trạng hôn nhân của ứng viên vay, từ đó xác định mức độ rủi ro và đưa ra quyết định về việc chấp nhận hay từ chối khoản vay. Điều này không chỉ giúp tiết kiệm thời gian mà còn giảm thiểu sai sót do yếu tố con người.

Quản lý rủi ro tín dụng

Một ứng dụng quan trọng khác là trong việc quản lý rủi ro tín dụng. Các mô hình học máy có thể giúp các tổ chức tài chính xác định các khách hàng có nguy cơ vỡ nợ cao, từ đó đưa ra các biện pháp giảm thiểu rủi ro như điều chỉnh lãi suất hoặc yêu cầu các biện pháp bảo đảm thêm. Với khả năng phân tích các đặc trưng phức tạp và mối quan hệ giữa các yếu tố, mô hình học máy giúp các ngân hàng phát hiện sớm các dấu hiệu cảnh báo rủi ro.

Cải thiện dịch vụ khách hàng

Các mô hình học máy còn có thể được ứng dụng để cải thiện trải nghiệm khách hàng trong lĩnh vực tài chính. Bằng cách phân tích các hành vi và yêu cầu vay của khách hàng, ngân hàng có thể cung cấp các sản phẩm tài chính phù hợp với nhu cầu của từng nhóm khách hàng. Ví dụ, các mô hình có thể dự đoán các sản phẩm vay phù hợp với khách hàng dựa trên lịch sử giao dịch, mức thu nhập và các yếu tố cá nhân khác.

Áp dụng vào các ngành khác ngoài tài chính

Mặc dù nghiên cứu này tập trung vào lĩnh vực tài chính, nhưng các mô hình phân loại học máy này cũng có thể được áp dụng trong các ngành khác như bảo hiểm, chăm sóc sức khỏe, và bán lẻ. Ví dụ, trong ngành bảo hiểm, các mô hình có thể giúp dự đoán khả năng yêu cầu bảo hiểm của khách hàng hoặc xác định các yếu tố rủi ro để điều chỉnh phí bảo hiểm.

TÀI LIỆU THAM KHẢO