**TRƯỜNG ĐẠI HỌC NGUYỄN TẤT THÀNH**

**KHOA CÔNG NGHỆ THÔNG TIN**



**TIỂU LUẬN HỌC MÁY VÀ ỨNG DỤNG**

**ĐỀ TÀI: XÂY DỰNG MÔ HÌNH DỰ ĐOÁN KHẢ NĂNG VỠ NỢ TỪ DỮ LIỆU TÍN DỤNG**

**Giảng viên hướng dẫn: TRẦN CHÂU THANH THIỆN**

**Sinh viên thực hiện: NGUYỄN VĂN KHA**

**MSSV: 2100011608**

**Khoá: 21**

**Ngành/ chuyên ngành: TRÍ TUỆ NHÂN TẠO**

TP. HCM, tháng 09 năm 2024

**TRƯỜNG ĐẠI HỌC NGUYỄN TẤT THÀNH**

**KHOA CÔNG NGHỆ THÔNG TIN**



**TIỂU LUẬN HỌC MÁY VÀ ỨNG DỤNG**

**ĐỀ TÀI: XÂY DỰNG MÔ HÌNH DỰ ĐOÁN KHẢ NĂNG VỠ NỢ TỪ DỮ LIỆU TÍN DỤNG**

**Giảng viên hướng dẫn: TRẦN CHÂU THANH THIỆN**

**Sinh viên thực hiện: NGUYỄN VĂN KHA**

**MSSV: 2100011608**

**Khoá: 21**

**Ngành/ chuyên ngành: TRÍ TUỆ NHÂN TẠO**

TP. HCM, tháng 09 năm 2024

# LỜI CẢM ƠN

Kính Thầy Trần Châu Thanh Thiện,

Em xin gửi lời biết ơn sâu sắc và lòng tôn kính đến Thầy về sự hỗ trợ và sự hướng dẫn tận tình của Thầy trong quá trình thực hiện tiểu luận môn học "Học máy và ứng dụng" tại trường Đại học Nguyễn Tất Thành. Sự tận tâm và kiến thức sâu rộng của Thầy đã là nguồn động viên và cổ vũ vô cùng quý báu đối với em.

Trong suốt khoảng thời gian em được làm việc cùng Thầy, em đã học được không chỉ về kiến thức chuyên ngành mà còn về sự cống hiến và đam mê trong công việc. Thầy đã luôn sẵn sàng chia sẻ kiến thức, giải đáp mọi câu hỏi của em và tạo điều kiện để em có thể phát triển tốt nhất khả năng của mình.

Em biết ơn Thầy vì tất cả những kiến thức và kỹ năng quý báu mà em đã học được dưới sự hướng dẫn của thầy. Sự tận tâm và lòng nhiệt thành của Thầy đã góp phần quan trọng vào sự thành công của em trong đồ án này.

Chúng em xin gửi lời cảm ơn chân thành nhất đến Thầy Trần Châu Thanh Thiện và mong rằng Thầy sẽ tiếp tục truyền đạt sự đam mê và kiến thức đỉnh cao của mình cho thế hệ sinh viên tiếp theo. Thầy là nguồn cảm hứng lớn lao cho chúng em.

Cảm ơn Thầy một lần nữa và chúc thầy luôn được sức khỏe, hạnh phúc và thành công trong mọi sự nghiệp.

Trân trọng,

Nguyễn Văn Kha

# LỜI MỞ ĐẦU

Trong bối cảnh kinh tế hiện nay, việc quản lý rủi ro tín dụng đóng vai trò ngày càng quan trọng đối với các tổ chức tài chính. Khả năng dự đoán chính xác khả năng vỡ nợ của khách hàng không chỉ giúp giảm thiểu tổn thất tài chính mà còn góp phần duy trì sự ổn định của toàn bộ hệ thống ngân hàng.

Nghiên cứu này tập trung vào việc xây dựng một mô hình dự đoán khả năng vỡ nợ dựa trên dữ liệu tín dụng. Bằng cách kết hợp các kỹ thuật học máy tiên tiến với bộ dữ liệu tín dụng toàn diện, mục tiêu hướng đến việc phát triển một công cụ dự báo chính xác và đáng tin cậy. Mô hình này sẽ giúp các tổ chức tài chính đánh giá rủi ro một cách hiệu quả hơn, từ đó đưa ra quyết định cho vay sáng suốt và xây dựng chiến lược quản lý danh mục đầu tư hợp lý.

Em hy vọng rằng kết quả của nghiên cứu này không chỉ có ý nghĩa học thuật mà còn có thể áp dụng được trong thực tế, góp phần nâng cao hiệu quả quản lý rủi ro tín dụng tại các tổ chức tài chính ở Việt Nam.

Em xin bày tỏ lòng biết ơn sâu sắc đến tất cả những người đã hỗ trợ và đóng góp vào việc thực hiện đồ án này, đặc biệt là giảng viên hướng dẫn và các chuyên gia trong lĩnh vực công nghệ thông tin và trí tuệ nhân tạo.

Mong rằng tiểu luận này sẽ đem lại giá trị và kiến thức hữu ích cho người đọc và là một bước tiến quan trọng trong lĩnh vực nhận diện tiền tự động.

Cảm ơn quý Thầy\Cô và các bạn đã theo dõi và ủng hộ tiểu luận của chúng em.

Nguyễn Văn Kha

BM-ChT-11

|  |  |
| --- | --- |
| TRƯỜNG ĐẠI HỌC NGUYỄN TẤT THÀNH  **TRUNG TÂM KHẢO THÍ** | **KỲ THI KẾT THÚC HỌC PHẦN**  **HỌC KỲ 3 – NĂM HỌC 2023 - 2024** |

# Phiếu chấm điểm TIỂU LUẬN HẾT MÔN

Môn thi: Học máy và ứng dụng Lớp học phần: 21DTH1C

Nhóm sinh viên thực hiện :

1.Nguyễn Văn Kha – 2100011608 Tham gia đóng góp: Xây dựng mô hình

Ngày thi: 24/09/2024 Phòng thi: L.603A

Đề tài tiểu luận: Xây dựng mô hình dự đoán khả năng vỡ nợ từ dữ liệu tín dụng.

Phần đánh giá của giảng viên (căn cứ trên thang rubrics của môn học):

|  |  |  |  |
| --- | --- | --- | --- |
| **Tiêu chí (theo CĐR HP)** | **Đánh giá của GV** | **Điểm tối đa** | **Điểm đạt được** |
| Cấu trúc của báo cáo |  | 2 |  |
| **Nội dung** | | | |
| Các nội dung thành phần |  | 4 |  |
| Lập luận |  | 2 |  |
| Kết luận |  | 1 |  |
| Trình bày |  | 1 |  |
| TỔNG ĐIỂM |  | 10 |  |

**Giảng viên chấm thi**

*(ký, ghi rõ họ tên)*

**ThS. Trần Châu Thanh Thiện**

# MỤC LỤC

[LỜI CẢM ƠN 3](#_Toc174296190)

[LỜI MỞ ĐẦU 4](#_Toc174296191)

[Phiếu chấm điểm TIỂU LUẬN HẾT MÔN 5](#_Toc174296192)

[MỤC LỤC 6](#_Toc174296193)

[CHƯƠNG 1 8](#_Toc174296194)

[Tổng quan đề tài 8](#_Toc174296195)

[1. Giới thiệu vấn đề: 8](#_Toc174296196)

[2. Mục tiêu và hướng nghiên cứu 8](#_Toc174296197)

[CHƯƠNG 2 10](#_Toc174296198)

[Cơ sở lý thuyết 10](#_Toc174296199)

[1. Học máy (Machine Learning) 10](#_Toc174296200)

[2. Các thuật toán học máy 10](#_Toc174296201)

[2.1. Hồi quy Logistic (Logistic Regression) 10](#_Toc174296202)

[2.2. Cây quyết định và Rừng ngẫu nhiên (Decision Tree and Random Forest) 10](#_Toc174296203)

[2.3. Support Vector Machine (SVM) 10](#_Toc174296204)

[2.4. XGBoost (Extreme Gradient Boosting) 10](#_Toc174296205)

[CHƯƠNG 3 11](#_Toc174296206)

[Mô hình thực nghiệm 11](#_Toc174296207)

[1. Ý tưởng thực nghiệm: 11](#_Toc174296208)

[2. Xây dựng mô hình: 11](#_Toc174296209)

[2.1. Import Libraries 11](#_Toc174296210)

[2.2. Đọc và giải thích dữ liệu (Read and Explain Dataset) 12](#_Toc174296211)

[**2.2.1. Mô tả các đặc trưng** 12](#_Toc174296212)

[**2.2.2. Định nghĩa các cột Object/ Integer** 14](#_Toc174296213)

[2.3. Khai phá dữ liệu (Exploratory Data Analysis) 14](#_Toc174296214)

[**2.3.1. Phân tích khách hàng dựa trên độ tuổi** 14](#_Toc174296215)

[**2.3.2. Phân tích khách hàng dựa trên thu nhập** 16](#_Toc174296216)

[**2.3.3. Phân tích khách hàng dựa trên số tiền vay mà cá nhân yêu cầu** 16](#_Toc174296217)

[**2.3.4. Mức độ tương quan giữa các đặc trưng (Correlation)** 17](#_Toc174296218)

[**2.3.5. Biểu đồ tròn về quyền sở hữu nhà của người vay** 18](#_Toc174296219)

[**2.3.6. Biểu đồ tròn về điểm tính dụng** 18](#_Toc174296220)

[**2.3.7. Biểu đồ thể hiện mục đích khoản vay** 19](#_Toc174296221)

[**2.3.8. Biểu đồ thể hiện lịch sử các cá nhân vỡ nợ** 19](#_Toc174296222)

[**2.3.9. Phân bố độ tuổi của khách hàng theo tình trạng vay** 20](#_Toc174296223)

[**2.3.10. Hiển thị các mối quan hệ giữa các biến và xem xét sự phân phối của các điểm dữ liệu** 20](#_Toc174296224)

[2.4. Feature Engineering 21](#_Toc174296225)

[**2.4.1. Xử lý dữ liệu thiếu** 21](#_Toc174296226)

[**2.4.2. Xử lý dữ liệu ngoại lai** 23](#_Toc174296227)

[2.5. Tiền xử lý dữ liệu (Data Preprocessing) 26](#_Toc174296228)

[**2.5.1. Áp dụng One Hot Encoder trên các cột feature có nhiều hơn 2 category** 26](#_Toc174296229)

[**2.5.2. Áp dụng Ordinal Encoder trên một cột feature chỉ có 2 category** 26](#_Toc174296230)

[**2.5.3. Cân bằng dữ liệu bằng SMOTE** 27](#_Toc174296231)

[**2.5.4. Chuẩn hóa dữ liệu** 28](#_Toc174296232)

[2.6. Các mô hình phân loại Machine Learning 28](#_Toc174296233)

[**2.6.1. Logistic Regression** 28](#_Toc174296234)

[**2.6.2. Decision Tree** 30](#_Toc174296235)

[**2.6.3. Random Forest** 32](#_Toc174296236)

[**2.6.4. XGBoost (Extreme Gradient Boosting)** 37](#_Toc174296237)

[**2.6.5. SVM (Super Vector Machine)** 38](#_Toc174296238)

[3. Đánh giá mô hình 41](#_Toc174296239)

[CHƯƠNG 4 43](#_Toc174296240)

[Kết luận và kiến nghị 43](#_Toc174296241)

[1. Kết quả đạt được: 43](#_Toc174296242)

[2. Đề xuất và kiến nghị: 43](#_Toc174296243)

[3. Kết luận: 43](#_Toc174296244)

# CHƯƠNG 1

# Tổng quan đề tài

## 1. Giới thiệu vấn đề:

Trong bối cảnh kinh tế hiện đại, các khoản vay đóng vai trò quan trọng trong việc thúc đẩy tăng trưởng và phát triển cá nhân cũng như doanh nghiệp. Tuy nhiên, quá trình cấp vay luôn tiềm ẩn rủi ro đối với các tổ chức tài chính. Các công ty cho vay, chuyên cung cấp đa dạng các loại khoản vay trên khắp các khu vực thành phố và tỉnh lẻ, đang phải đối mặt với thách thức trong việc đánh giá chính xác khả năng trả nợ của khách hàng.

Hiện tại, quy trình thẩm định khoản vay của các công ty này dựa trên nhiều yếu tố như tuổi tác, quyền sở hữu nhà, kinh nghiệm làm việc, mục đích vay, thu nhập, số tiền vay và lịch sử tín dụng. Tuy nhiên, quá trình này vẫn còn mang tính thủ công và tốn thời gian, dẫn đến hiệu quả chưa cao và khả năng xảy ra sai sót. Hơn nữa, với sự gia tăng nhanh chóng của số lượng đơn xin vay, các công ty cho vay cần một giải pháp tự động và chính xác hơn để đánh giá rủi ro tín dụng và đưa ra quyết định cho vay.

Vấn đề đặt ra là làm thế nào để xây dựng một mô hình dự đoán hiệu quả, có khả năng tự động hóa quá trình đánh giá điều kiện vay và xác định chính xác các phân khúc khách hàng đủ điều kiện cho các khoản vay. Mô hình này cần phải có khả năng xử lý lượng lớn dữ liệu, tích hợp nhiều yếu tố đánh giá, và đưa ra dự đoán chính xác về khả năng trả nợ của người vay.

Việc phát triển một mô hình dự đoán như vậy không chỉ giúp tối ưu hóa quy trình cấp vay, mà còn nâng cao hiệu quả quản lý rủi ro của công ty. Điều này sẽ giúp các công ty cho vay đưa ra quyết định cho vay nhanh chóng và chính xác hơn, cải thiện trải nghiệm khách hàng, đồng thời giảm thiểu rủi ro tín dụng.

Tuy nhiên, việc xây dựng mô hình này cũng đối mặt với nhiều thách thức. Đầu tiên là vấn đề mất cân bằng dữ liệu giữa các lớp, đòi hỏi phải áp dụng các kỹ thuật cân bằng lớp như SMOTE. Tiếp theo là việc lựa chọn và tinh chỉnh mô hình phù hợp từ nhiều thuật toán học máy khác nhau như Hồi quy Logistic, Decesion Tree, Random Forest, SVM và XGBoost. Cuối cùng, việc đảm bảo tính chính xác và khả năng giải thích của mô hình cũng là một yêu cầu quan trọng, đặc biệt trong lĩnh vực tài chính nhạy cảm.

Bằng cách giải quyết những vấn đề này, các công ty cho vay không chỉ cải thiện hiệu quả hoạt động mà còn nâng cao vị thế cạnh tranh trong thị trường cho vay ngày càng cạnh tranh.

## 2. Mục tiêu và hướng nghiên cứu

Mục tiêu chính của nghiên cứu này là phát triển một mô hình dự đoán rủi ro tín dụng tự động và chính xác, giúp tối ưu hóa quy trình cấp vay và giảm thiểu rủi ro tín dụng cho các tổ chức tài chính. Cụ thể, nghiên cứu hướng tới:

1. **Xây dựng và huấn luyện mô hình dự đoán**: Tạo ra một mô hình dự đoán khả năng vỡ nợ của khách hàng dựa trên các yếu tố đầu vào như tuổi tác, thu nhập, lịch sử tín dụng, và các thông tin tài chính cá nhân khác. Mô hình này cần có khả năng tự động đánh giá rủi ro tín dụng với độ chính xác cao.
2. **Giải quyết vấn đề mất cân bằng dữ liệu**: Ứng dụng các kỹ thuật như **SMOTE** (Synthetic Minority Over-sampling Technique) để xử lý tình trạng mất cân bằng giữa các lớp dữ liệu, đặc biệt là giữa các nhóm khách hàng có khả năng vỡ nợ và không vỡ nợ.
3. **So sánh và tối ưu hóa các mô hình Machine Learning**: Thử nghiệm và đánh giá hiệu suất của nhiều thuật toán khác nhau như Regression Logistic, **Decision Tree**, **Random Forest**, **SVM**, và **XGBoost**. Tìm kiếm mô hình tốt nhất về độ chính xác, khả năng giải thích và tính thực tiễn trong ứng dụng.
4. **Đánh giá tính khả dụng và hiệu quả của mô hình**: Phân tích khả năng áp dụng mô hình trong thực tế, đặc biệt là trong việc hỗ trợ các quyết định cho vay. Xác định mức độ cải thiện của quy trình cấp vay khi ứng dụng mô hình dự đoán.

Để đạt được các mục tiêu trên, nghiên cứu sẽ được tiến hành theo các hướng cụ thể sau:

1. **Thu thập và chuẩn bị dữ liệu**:

* **Dữ liệu**: Sử dụng bộ dữ liệu về tín dụng khách hàng, bao gồm các thông tin cá nhân, tài chính, lịch sử tín dụng, và trạng thái khoản vay.
* **Xử lý dữ liệu thiếu**: Sử dụng các phương pháp như điền giá trị trung bình, median, hoặc mô hình dự đoán để xử lý dữ liệu bị thiếu.
* **Chuẩn hóa và biến đổi dữ liệu**: Tiến hành chuẩn hóa dữ liệu và tạo ra các biến số mới (feature engineering) nhằm tối ưu hóa hiệu suất mô hình.

1. **Xử lý mất cân bằng dữ liệu**:

* **SMOTE**: Sử dụng kỹ thuật SMOTE để tăng số lượng mẫu ở lớp thiểu số, đảm bảo dữ liệu cân bằng hơn và mô hình học tốt hơn.
* **Undersampling và oversampling**: Thử nghiệm các kỹ thuật khác để kiểm tra và so sánh hiệu quả của chúng trong việc cải thiện mô hình.

1. **Xây dựng và huấn luyện mô hình**:

* **Thử nghiệm nhiều thuật toán:** Huấn luyện và so sánh các thuật toán học máy, bao gồm Logistic Regression, Decision Tree, Random Forest, SVM, và XGBoost.
* **Cross-validation:** Sử dụng kỹ thuật cross-validation để kiểm tra hiệu suất của các mô hình và tránh overfitting.

1. **Đánh giá mô hình**:

* **Đo lường hiệu suất:** Sử dụng các chỉ số đánh giá như Accuracy, Precision, Recall, F1-score, và ROC-AUC để so sánh hiệu quả của các mô hình.
* **Khả năng giải thích:** Đánh giá khả năng giải thích của mô hình đối với các quyết định tín dụng, đặc biệt là trong việc nhận diện các yếu tố chính dẫn đến rủi ro.

Nghiên cứu này kỳ vọng sẽ đóng góp vào việc cải tiến quy trình cấp vay, giúp các tổ chức tài chính nâng cao khả năng quản lý rủi ro, giảm thiểu tỉ lệ vỡ nợ, và đồng thời tối ưu hóa hiệu suất hoạt động kinh doanh.

# CHƯƠNG 2

# Cơ sở lý thuyết

## 1. Học máy (Machine Learning)

## 2. Các thuật toán học máy

### 2.1. Hồi quy Logistic (Logistic Regression)

### 2.2. Cây quyết định và Rừng ngẫu nhiên (Decision Tree and Random Forest)

### 2.3. Support Vector Machine (SVM)

### 2.4. XGBoost (Extreme Gradient Boosting)

# CHƯƠNG 3

# Mô hình thực nghiệm

## 1. Ý tưởng thực nghiệm:

1. Thu nhập dữ liệu

* Sử dụng dữ liệu tín dụng từ kaggle
* Dữ liệu bao gồm thông tin cá nhân, lịch sử tín dụng, thu nhập, nợ hiện tại, và trạng thái vỡ nợ của khách hàng

2. Khai phá dữ liệu ((Exploratory Data Analysis))

* Phân tích khách hàng dựa trên độ tuổi
* Phân tích khách hàng dựa trên thu nhập
* Phân tích khách hàng dựa trên số tiền vay mà cá nhân yêu cầu
* Mức độ tương quan giữa các đặc trưng (Correlation)

3. Feature Engineering

* Làm sạch dữ liệu, loại bỏ các giá trị thiếu hoặc không hợp lệ
* Chuyển đổi dữ liệu

4. Tiền xử lý dữ liệu

* Mã hóa các cột category
* Chuẩn hóa dữ liệu

5. Xây dựng và huấn luyện các mô hình

* Logistic Regression
* Decision Tree
* Random Forest
* XGBoost (Extreme Gradient Boosting)
* SVM (Super Vector Machine)

6. Đánh giá mô hình

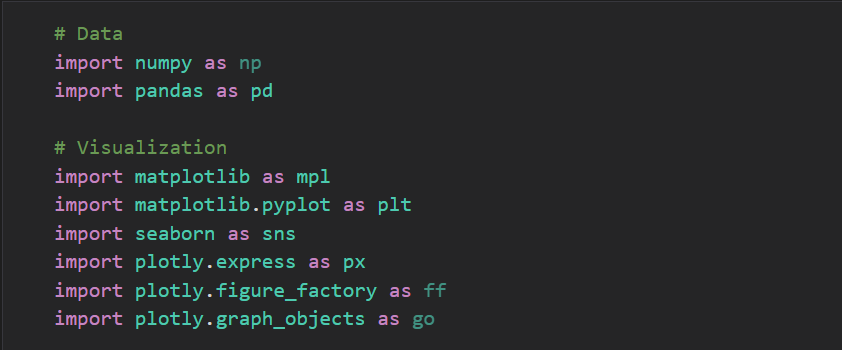
7. Lựa chọn mô hình tốt nhất

8. Kiểm tra mô hình trên tập dữ liệu độc lập

## 2. Xây dựng mô hình:

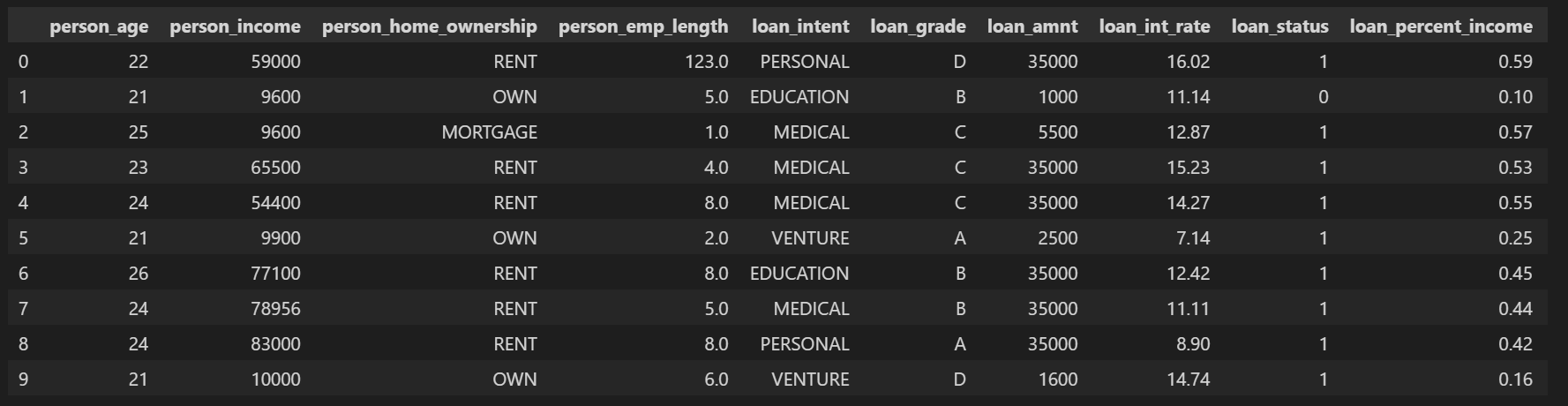
### 2.1. Import Libraries

Import một số thư viện để read và visualization data



### 2.2. Đọc và giải thích dữ liệu (Read and Explain Dataset)

Read data

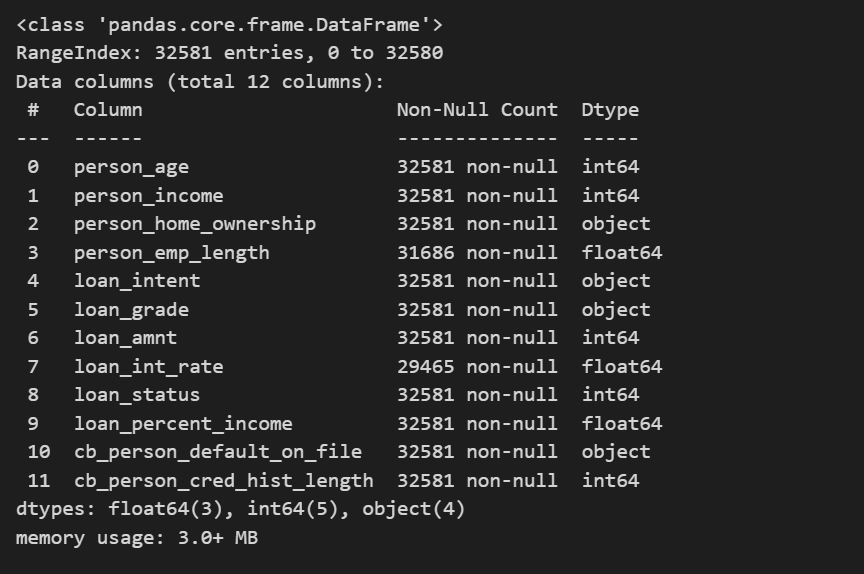


#### **2.2.1. Mô tả các đặc trưng**

|  |  |
| --- | --- |
| **person\_age** | Tuổi của người đăng ký vay |
| **person\_income** | Thu nhập hàng năm của cá nhân |
| **person\_emp\_length** | Thời gian làm việc của cá nhân tính bằng năm |
| **loan\_amnt** | Số tiền vay mà cá nhân yêu cầu |
| **loan\_int\_rate** | Lãi suất liên quan đến khoản vay |
| **loan\_status** | 0: không vỡ nợ, 1: vỡ nợ |
| **loan\_percent\_income** | Tỷ lệ thu nhập của cá nhân được phân bổ để trả nợ |
| **cb\_person\_cred\_hist\_length** | Lịch sử vỡ nợ của cá nhân theo hồ sơ của cục tín dụng |

|  |  |  |
| --- | --- | --- |
| **person\_home\_ownership** | **loan\_grade** | **cb\_person\_default\_on\_file** |
| rent: Cá nhân đang thuê nhà | A: Người vay có khả năng tín dụng cao, chỉ ra rủi ro thấp | Y: Cá nhân có lịch sử vỡ nợ trong hồ sơ tín dụng của họ |
| mortgage: Cá nhân đang thế chấp bất động sản họ sở hữu | B: Người vay có rủi ro tương đối thấp, nhưng không đáng tin cậy bằng Hạng A | N: Cá nhân không có bất kỳ lịch sử vỡ nợ nào |
| own: Cá nhân sở hữu hoàn toàn ngôi nhà của họ | C: Khả năng tín dụng của người vay ở mức trung bình |  |
| other: Các hình thức sở hữu nhà khác có thể đặc thù cho bộ dữ liệu. | D: Người vay được coi là có rủi ro cao hơn so với các hạng trước |  |
|  | E: Khả năng tín dụng của người vay thấp hơn, chỉ ra rủi ro cao hơn. |  |
|  | F: Người vay có rủi ro tín dụng đáng kể |  |
|  | G: Khả năng tín dụng của người vay là thấp nhất, biểu thị rủi ro cao nhất |  |

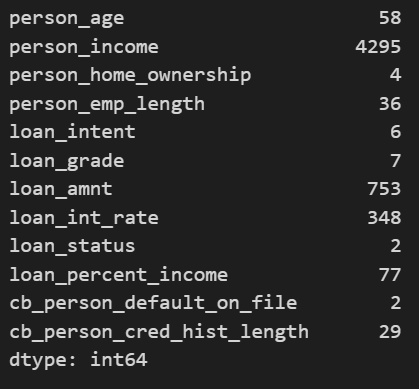
Kiểm tra thông tin của data:



Xem thống kê cơ từ các feature của dataframe:

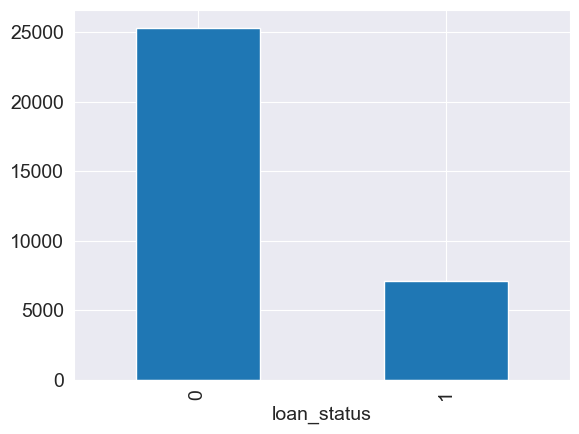


Đếm số lượng giá trị duy nhất (không trùng lặp) trong mỗi cột của DataFrame:



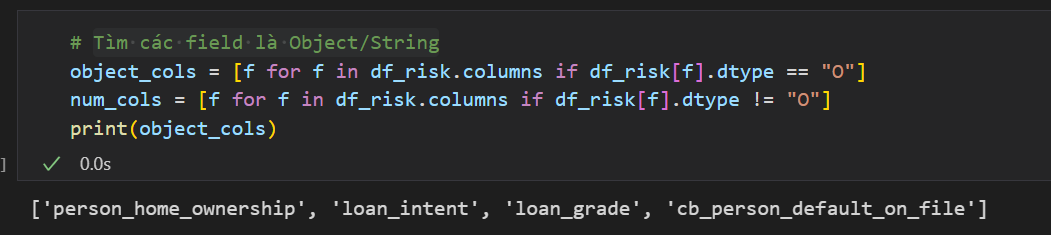
Thống kê khoản vay có hoàn trả không

* 0: Không vỡ nợ - Người vay đã trả nợ thành công theo thỏa thuận và không có vỡ nợ.
* 1: Vỡ nợ - Người vay không trả được nợ theo các điều khoản đã thỏa thuận và đã vỡ nợ.



#### **2.2.2. Định nghĩa các cột Object/ Integer**

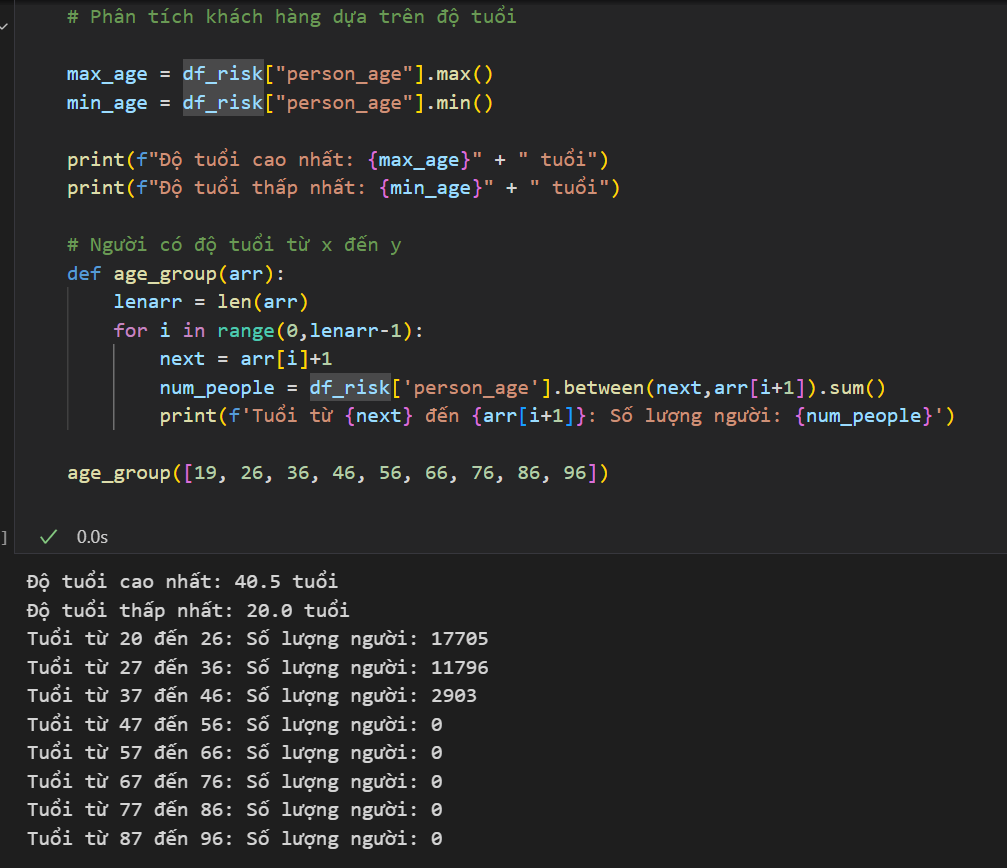
Tìm các field là Object/String:



### 2.3. Khai phá dữ liệu (Exploratory Data Analysis)

Vì bài toán của chúng ta xây dựng ở đây là dự đoán ra việc có trả nợ thành công hay là không nên chúng ta cần đi sâu vào các đặc trưng quan thiết yếu của khách hàng như độ tuổi, thu nhập, số tiền vay mà cá nhân yêu cầu.

#### **2.3.1. Phân tích khách hàng dựa trên độ tuổi**



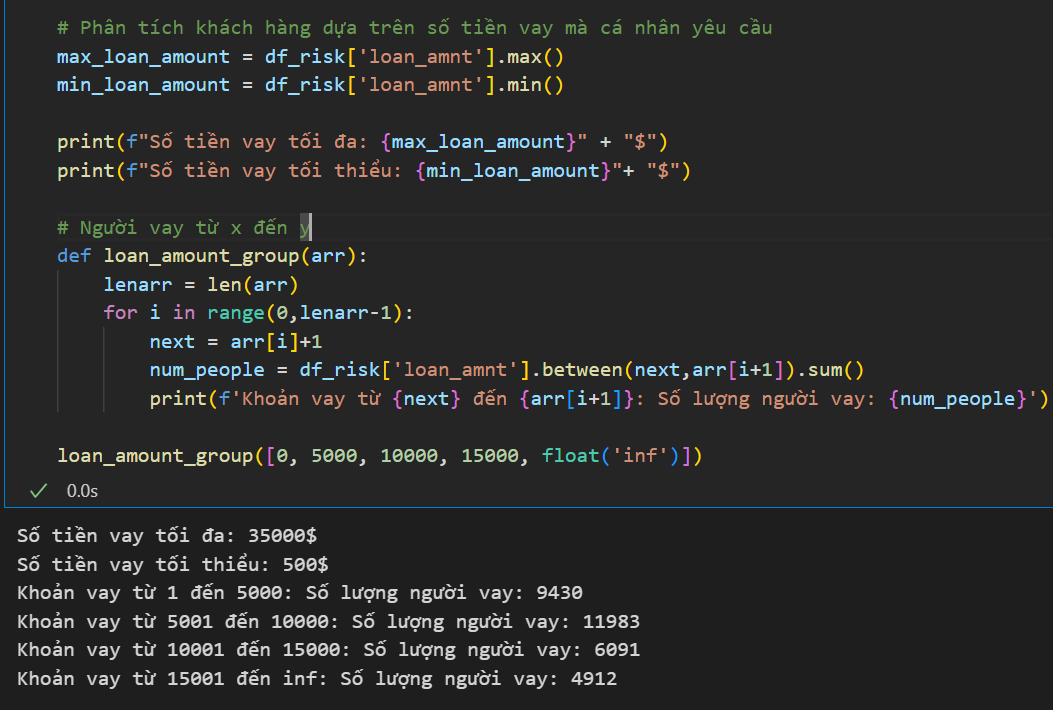
Phần lớn các cá nhân xin vay tiền là những người tương đối trẻ, với hơn 15.000 người nộp đơn trong độ tuổi từ 20 đến 26. Ngược lại, số lượng người nộp đơn giảm đáng kể khi tuổi tăng lên, với chỉ một vài người nộp đơn trong độ tuổi từ 77 đến 86 và không có ai trong độ tuổi từ 87 đến 96. Vì chỉ có bốn cá nhân trên 76 tuổi, sự hiện diện của họ có thể giới thiệu sự sai lệch vào tập dữ liệu. Do đó, sẽ là hợp lý nếu loại bỏ họ.

#### **2.3.2. Phân tích khách hàng dựa trên thu nhập**

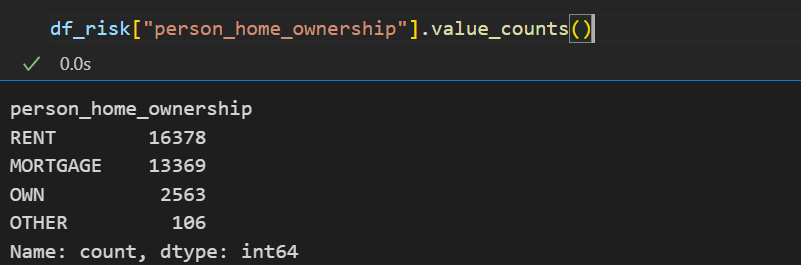


Phân bố thu nhập của những người xin vay cho thấy phần lớn có thu nhập từ $25,001 đến $50,000, với 11549 cá nhân nằm trong khoảng này. Số lượng người nộp đơn giảm dần khi thu nhập tăng, với ít người hơn có thu nhập dưới $25,000 và trên $75,000.

#### **2.3.3. Phân tích khách hàng dựa trên số tiền vay mà cá nhân yêu cầu**

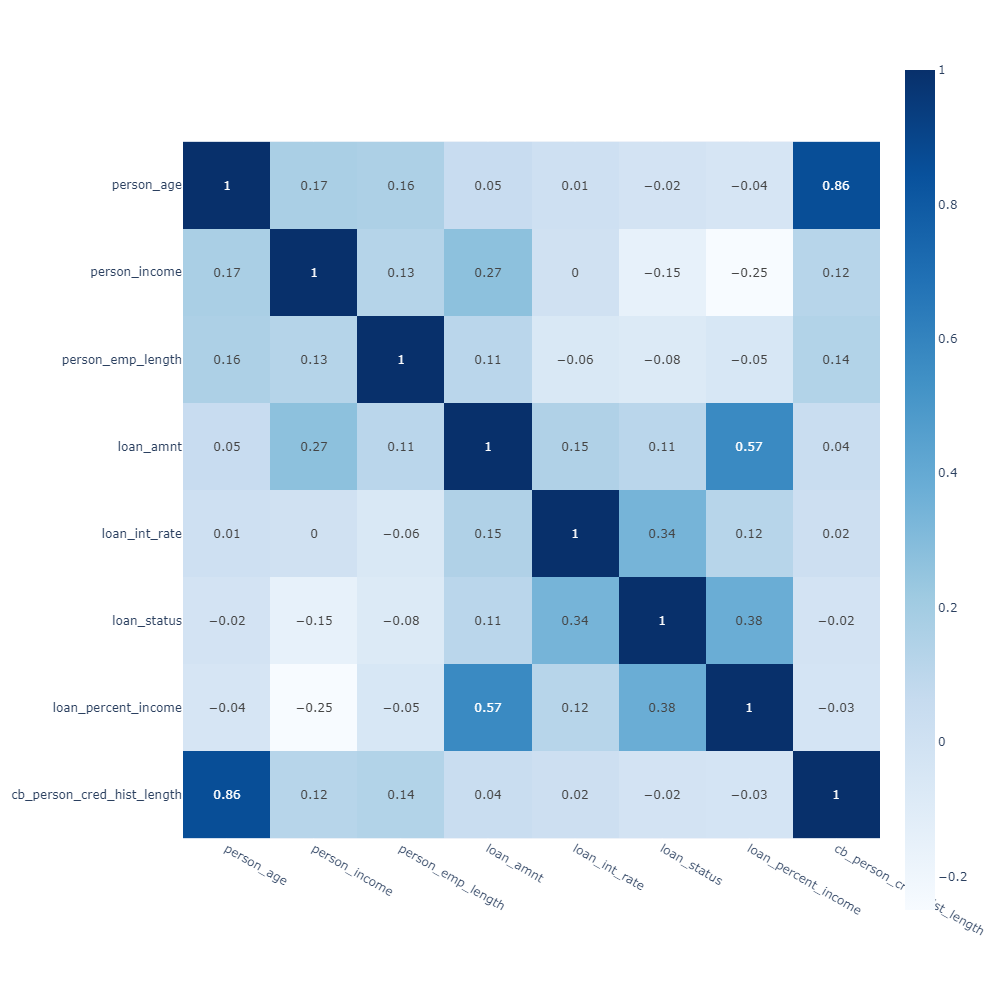


Phân bố số tiền vay cho thấy một phần lớn các cá nhân xin vay từ $5,001 đến $10,000, với 11,983 cá nhân nằm trong khoảng này. Số lượng người nộp đơn giảm dần khi số tiền vay tăng vượt quá $10,000, với số tiền vay tối thiểu là $500 và số tiền vay tối đa là $35,000.

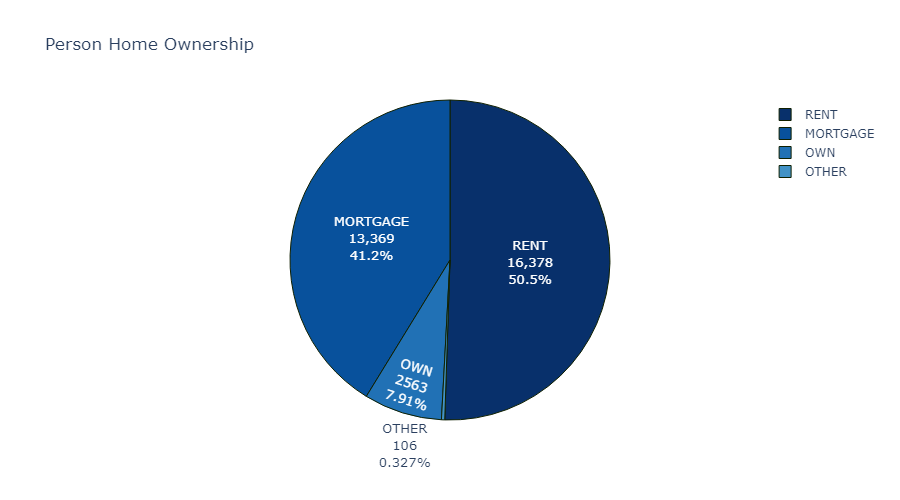


Phân bố tình trạng sở hữu nhà của các cá nhân cho thấy phần lớn (16,378 người) là người thuê nhà, tiếp theo là những người có thế chấp (13,369 người). Có ít ứng viên sở hữu nhà của mình (2,563 người), trong khi một số rất nhỏ (106 người) thuộc vào nhóm "Khác".

#### **2.3.4. Mức độ tương quan giữa các đặc trưng (Correlation)**

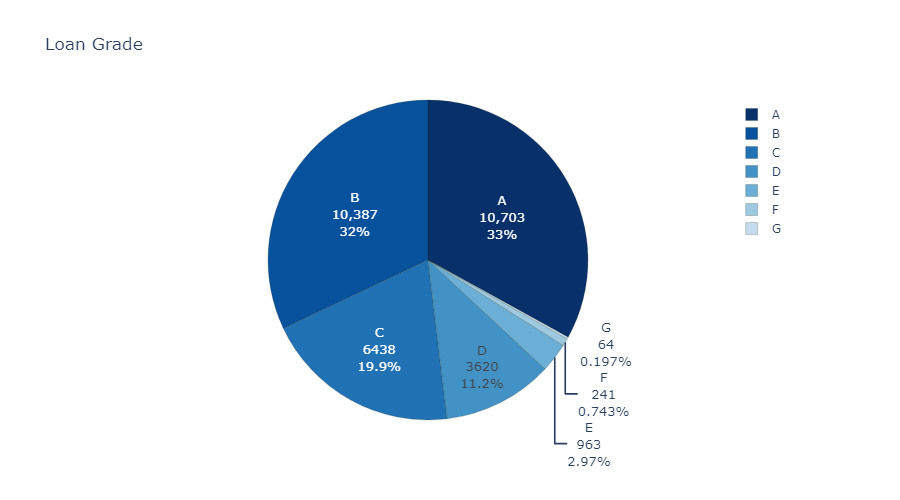


#### **2.3.5. Biểu đồ tròn về quyền sở hữu nhà của người vay**



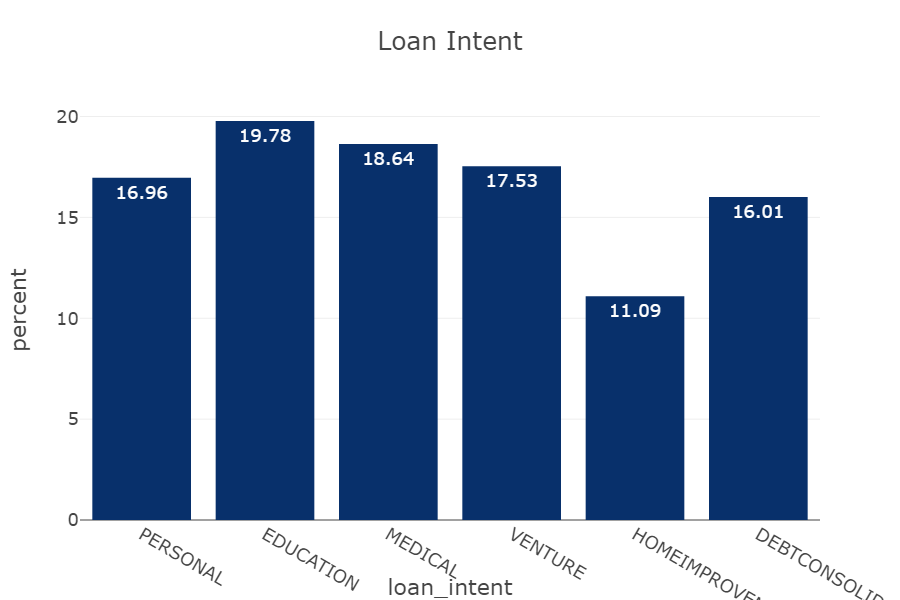
Phần lớn các cá nhân trong tập dữ liệu chiếm khoảng một nửa, sống trong các chỗ ở thuê thay vì sở hữu nhà riêng. Hơn nữa, hơn 41% cá nhân có nhà bị thế chấp.

#### **2.3.6. Biểu đồ tròn về điểm tính dụng**



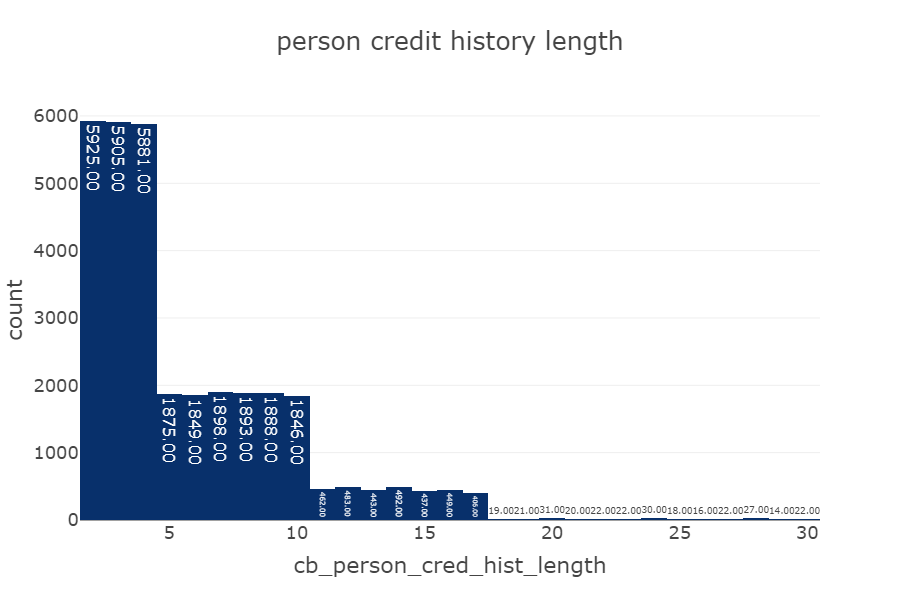
Phân phối điểm tín dụng nổi bật sự tập trung chủ yếu ở các điểm A và B, với lần lượt là 10,703 và 10,387 trường hợp, cho thấy hồ sơ rủi ro thấp hơn đối với các khoản vay này. Điểm C theo sát với 6,438 trường hợp. Khi các điểm giảm dần, số lượng các khoản vay giảm, cho thấy xu hướng giảm dần về chất lượng khoản vay. Các điểm D, E, F và G đại diện cho số lượng trường hợp ngày càng ít hơn, với lần lượt là 3620, 963, 241 và 64 khoản vay, phản ánh tiềm năng rủi ro cao hơn trong các danh mục này.

#### **2.3.7. Biểu đồ thể hiện mục đích khoản vay**



Dữ liệu cho thấy tỷ lệ cao nhất của các cá nhân đã chọn vay học tập, tiếp theo gần sau là vay y tế. Thêm vào đó, có vẻ như tỷ lệ giữa vay cá nhân và vay đầu tư là tương đương, cho thấy mức độ quan tâm tương tự trong hai danh mục này.

#### **2.3.8. Biểu đồ thể hiện lịch sử các cá nhân vỡ nợ**



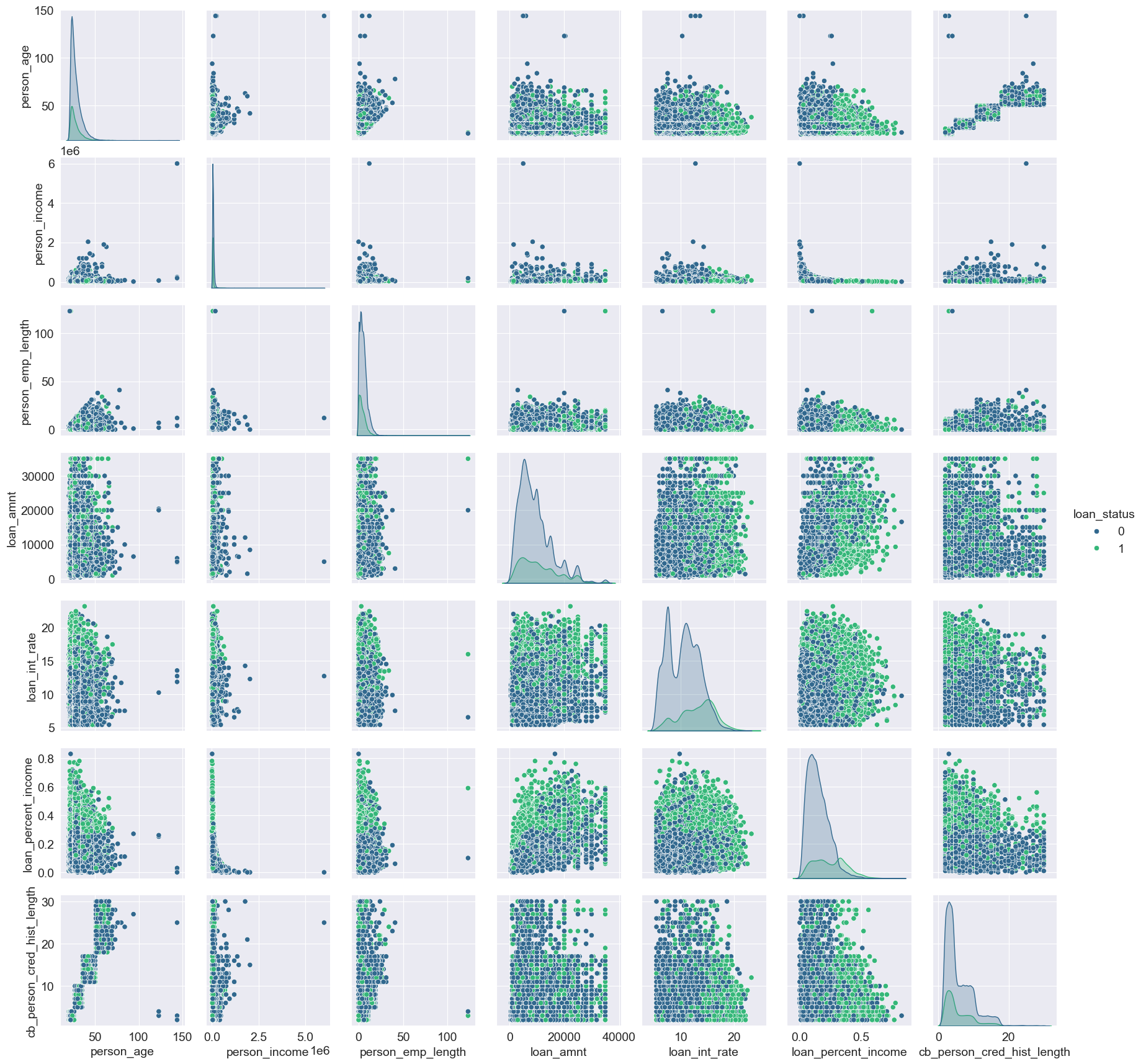
Phần lớn các cá nhân trong tập dữ liệu không bị vỡ nợ khoản vay nào. Tuy nhiên, có một nhóm nhỏ hơn các cá nhân đã vỡ nợ từ 3 đến 8 lần.

#### **2.3.9. Phân bố độ tuổi của khách hàng theo tình trạng vay**



Phần lớn các cá nhân nằm trong độ tuổi từ 20 đến 35 tuổi, và có vẻ như hầu hết các khoản vay đều được các cá nhân trong độ tuổi này thực hiện.

#### **2.3.10. Hiển thị các mối quan hệ giữa các biến và xem xét sự phân phối của các điểm dữ liệu**

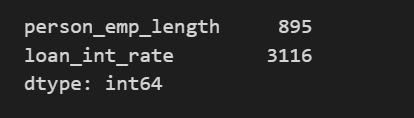


### 2.4. Feature Engineering

#### **2.4.1. Xử lý dữ liệu thiếu**

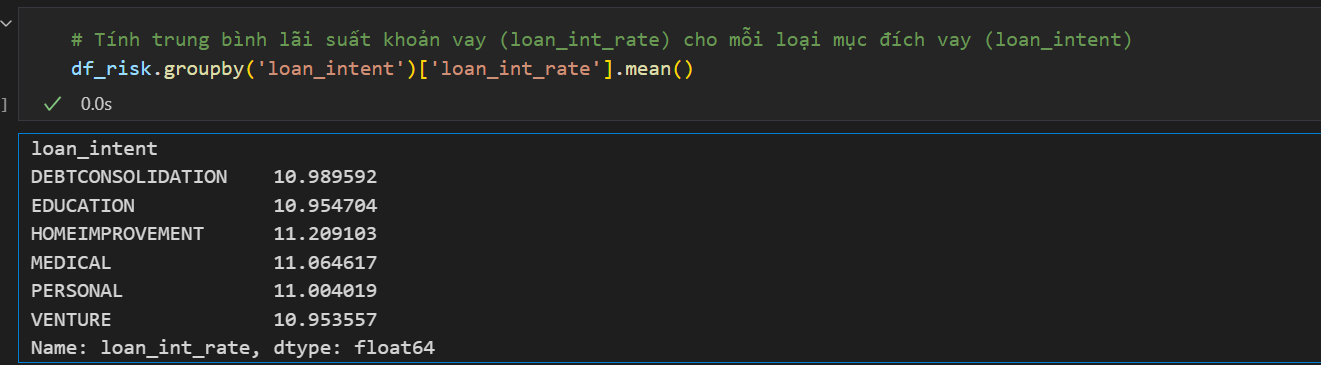
Xử lý dữ liệu thiếu bằng cách tính trung bình các feature null ("person\_emp\_length", "loan\_int\_rate") dựa trên mục đích vay (loan\_intent)

Kiểm tra dữ liệu thiếu trong data:

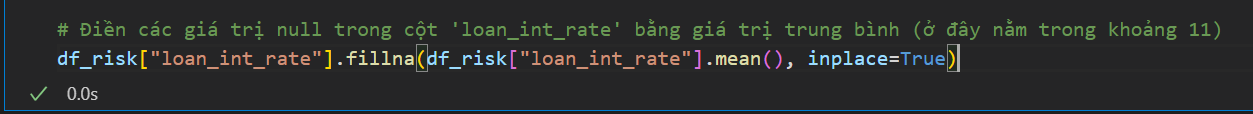
**

Cả hai cột đều có dữ liệu là Integer nên giá trị null có thể được điền theo giá trị trung bình dựa trên mục đích vay.

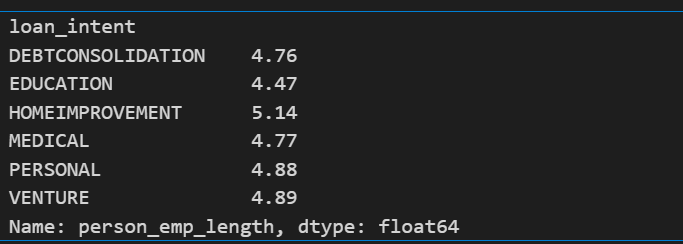
Tính trung bình lãi suất khoản vay (loan\_int\_rate) cho mỗi loại mục đích vay (loan\_intent):



Chúng ta có thể thay thế các giá trị null trong cột "loan interest rate" bằng giá trị trung bình là 11. Giá trị trung bình này được tính toán từ tập dữ liệu, có tính đến tất cả các danh mục.



Tính trung bình thời gian làm việc (person\_emp\_length) cho mỗi mục đích vay (loan\_intent) làm tròn đến 2 chữ số thập phân:



Để tính số ngày trong 0.01 năm, chúng ta cần thực hiện một phép tính đơn giản:

\* Một năm thông thường có 365 ngày.

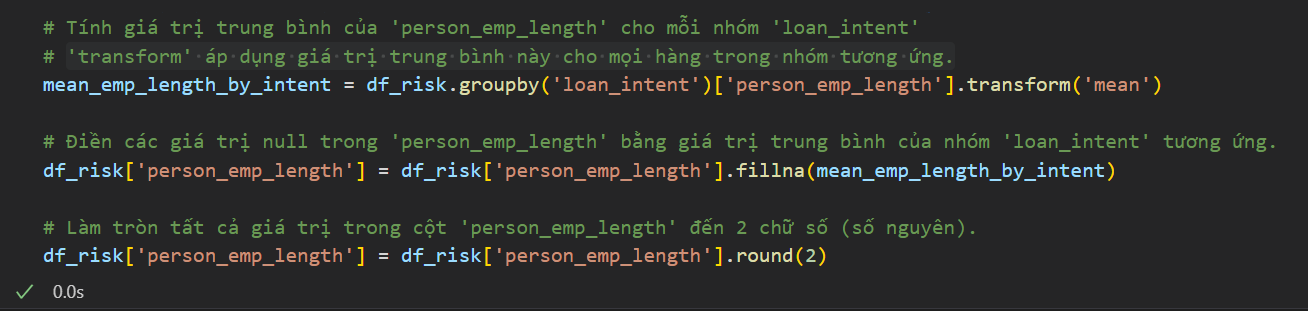
0.01 năm = 365 \* 0.01 = 3.65 ngày

\* Làm tròn kết quả, chúng ta có:

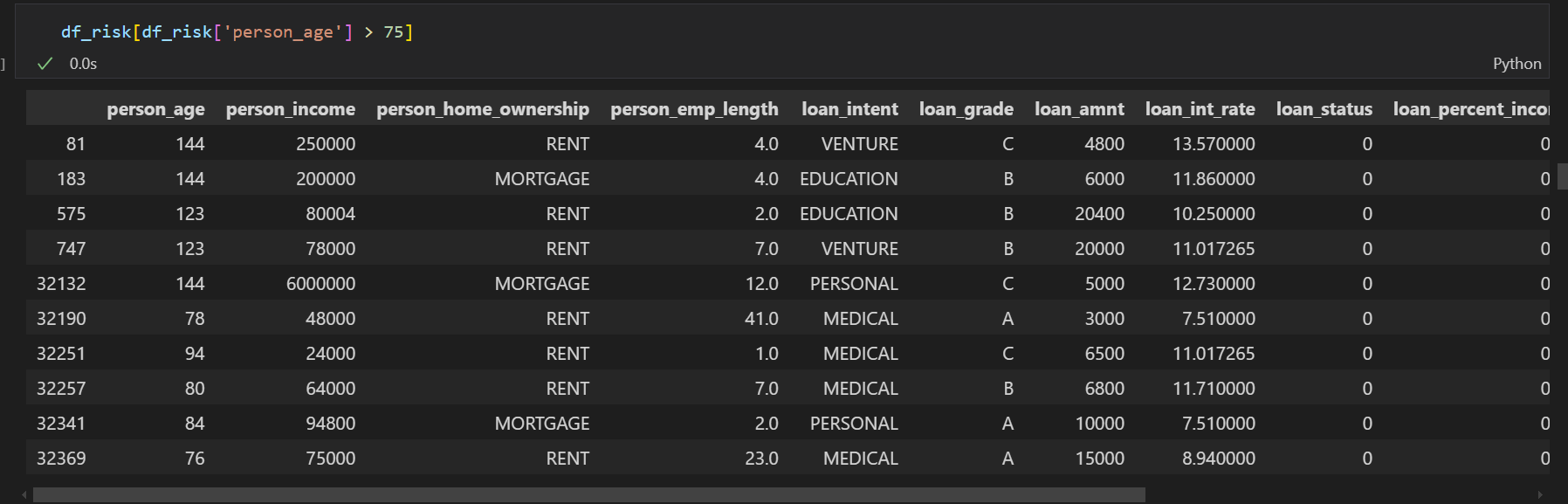
0.01 năm ≈ 3.65 ngày ≈ 3 ngày 15 giờ 36 phút

Vậy 0.01 năm tương đương với khoảng 3.65 ngày, hay gần đúng là 3 ngày rưỡi.

Tính giá trị trung bình của 'person\_emp\_length' cho mỗi nhóm 'loan\_intent', 'transform' áp dụng giá trị trung bình này cho mọi hàng trong nhóm tương ứng.



**Xử lý dữ liệu không cần thiết:**



Sau khi kiểm tra cột tuổi người đăng ký vay ta thấy hầu hết đều trả nợ đầy đủ (loan\_status = 0) nên tiến hành xóa để dữ liệu tốt hơn cho quá trình training

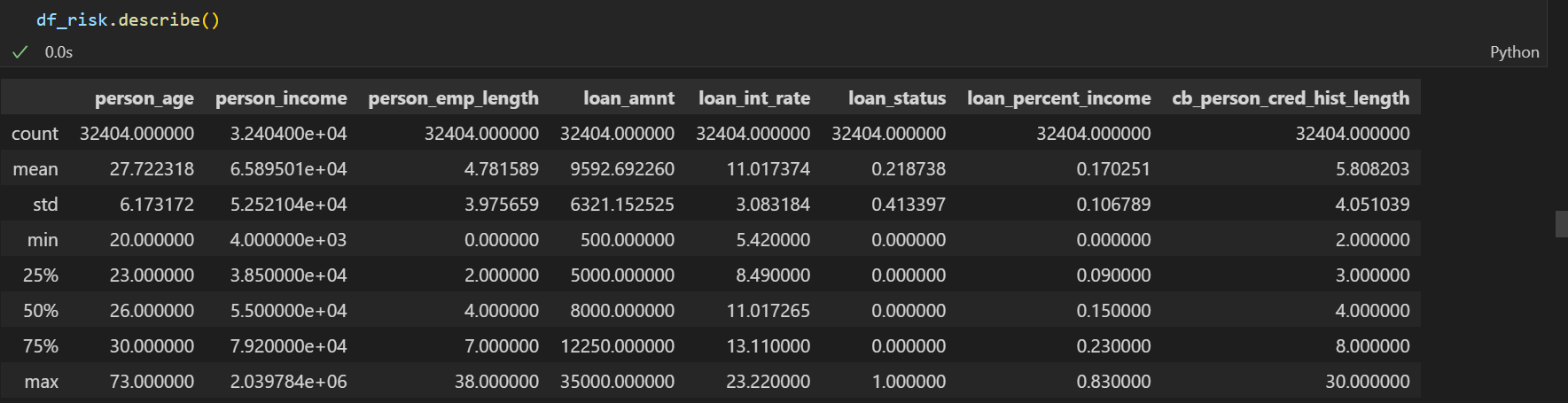
Kiểm tra thời gian làm việc của các cá nhân:



Bỏ hai hàng này đi vì nếu một người mới 22 tuổi thì làm sao người đó có thể làm việc trong 123 năm được.

#### **2.4.2. Xử lý dữ liệu ngoại lai**

**Xem lại thống kê:**



**- Dựa trên thống kê mô tả, đây là cách xử lý ngoại lai cho từng biến:**

- person\_age:

Giới hạn dưới: max(20, Q1 - 1.5 \* IQR) = max(20, 23 - 1.5 \* (30 - 23)) = 20

Giới hạn trên: min(73, Q3 + 1.5 \* IQR) = min(73, 30 + 1.5 \* (30 - 23)) = 40.5

- person\_income:

Giới hạn dưới: max(4000, Q1 - 1.5 \* IQR) = max(4000, 38500 - 1.5 \* (79200 - 38500)) = 38500

Giới hạn trên: Q3 + 1.5 \* IQR = 79200 + 1.5 \* (79200 - 38500) = 140250

- person\_emp\_length:

Giới hạn dưới: max(0, Q1 - 1.5 \* IQR) = max(0, 2 - 1.5 \* (7 - 2)) = 0

Giới hạn trên: Q3 + 1.5 \* IQR = 7 + 1.5 \* (7 - 2) = 14.5

- loan\_amnt:

Giới hạn dưới: max(500, Q1 - 1.5 \* IQR) = max(500, 5000 - 1.5 \* (12250 - 5000)) = 500

Giới hạn trên: min(35000, Q3 + 1.5 \* IQR) = min(35000, 12250 + 1.5 \* (12250 - 5000)) = 23125

- loan\_int\_rate:

Giới hạn dưới: max(5.42, Q1 - 1.5 \* IQR) = max(5.42, 8.49 - 1.5 \* (13.11 - 8.49)) = 5.42

Giới hạn trên: min(23.22, Q3 + 1.5 \* IQR) = min(23.22, 13.11 + 1.5 \* (13.11 - 8.49)) = 20.04

loan\_status: Đây là biến nhị phân (0 hoặc 1), không cần xử lý ngoại lai.

- loan\_percent\_income:

Giới hạn dưới: max(0, Q1 - 1.5 \* IQR) = max(0, 0.09 - 1.5 \* (0.23 - 0.09)) = 0

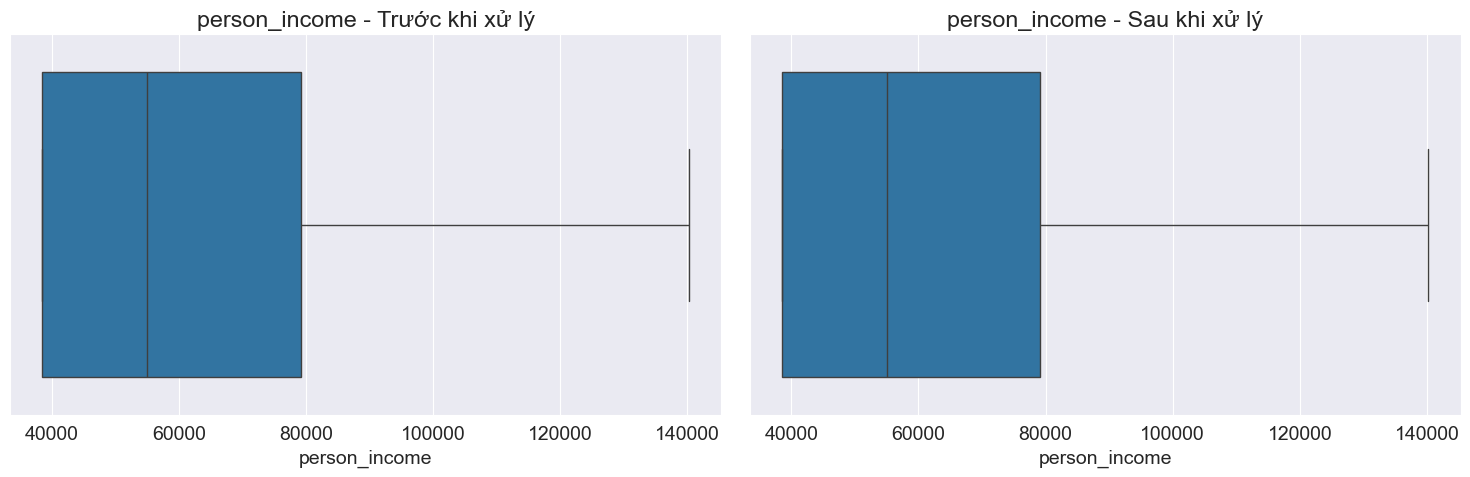
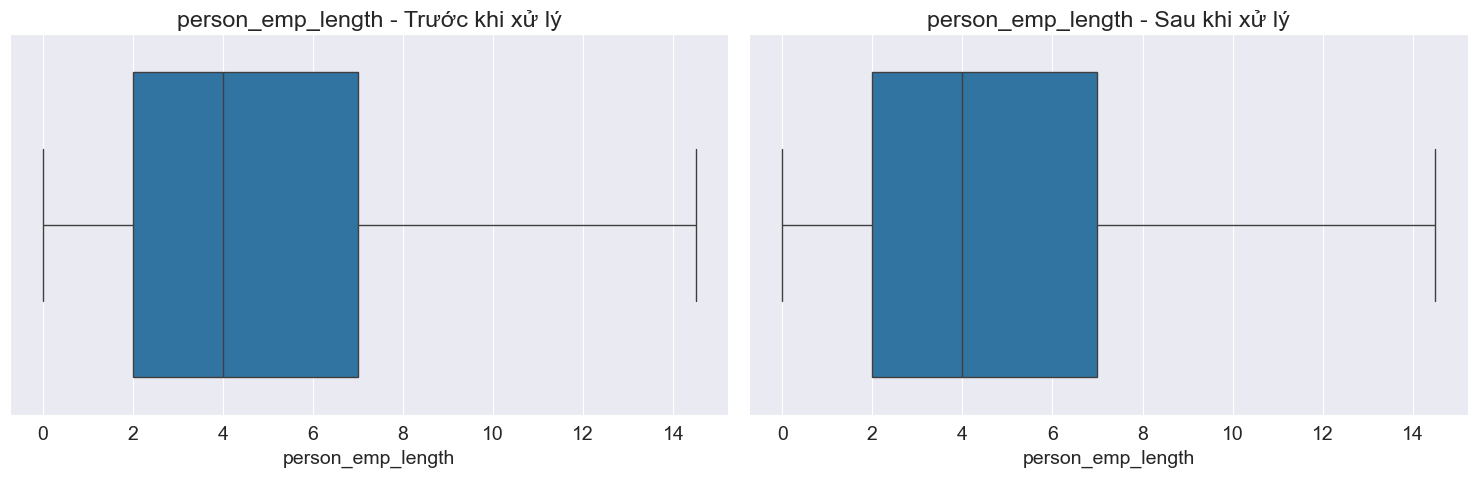
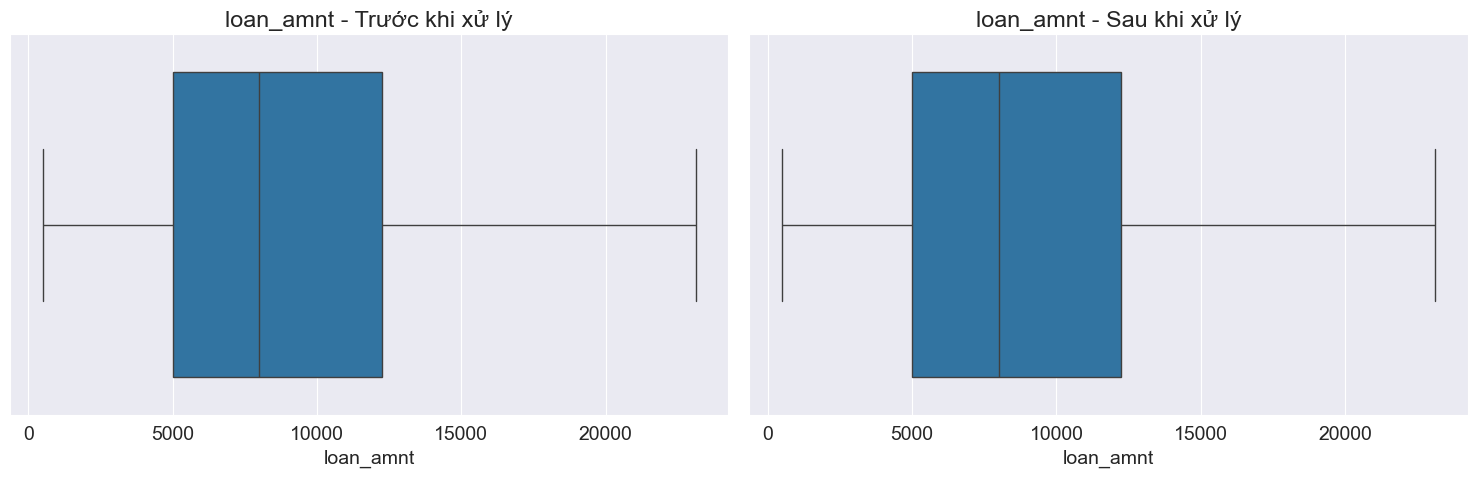
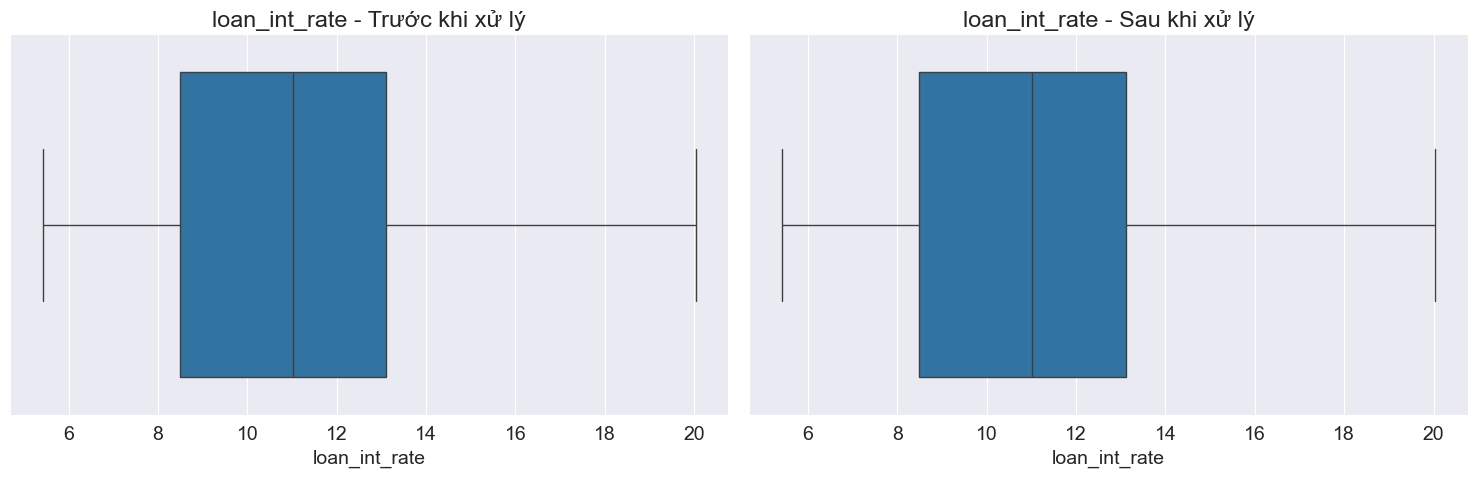
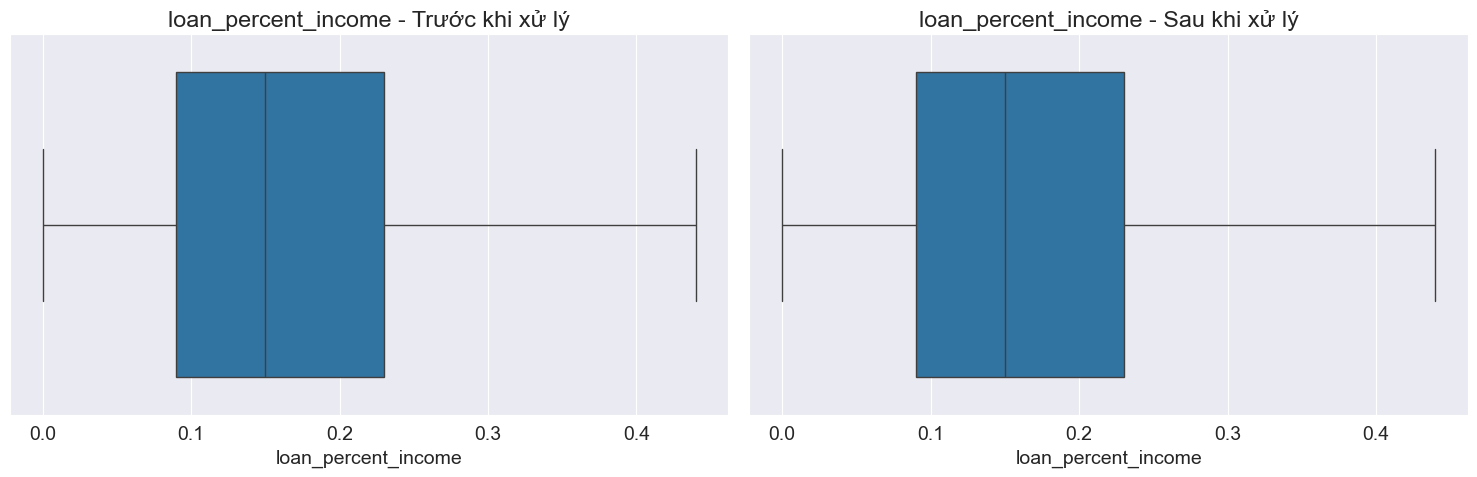
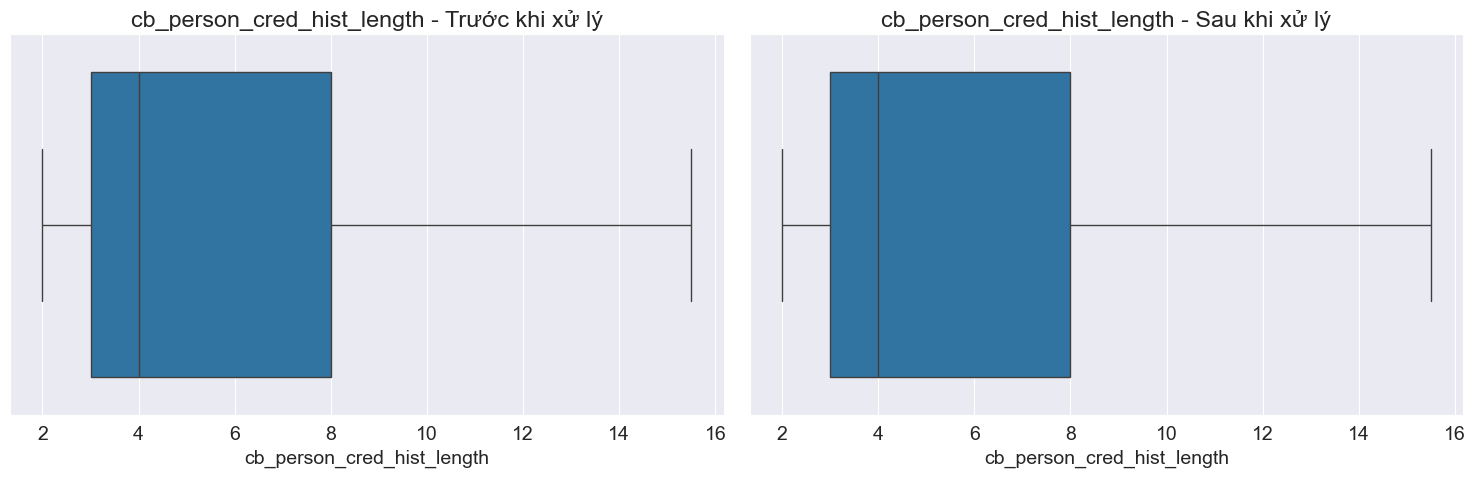
Giới hạn trên: min(0.83, Q3 + 1.5 \* IQR) = min(0.83, 0.23 + 1.5 \* (0.23 - 0.09)) = 0.44

- cb\_person\_cred\_hist\_length:

Giới hạn dưới: max(2, Q1 - 1.5 \* IQR) = max(2, 3 - 1.5 \* (8 - 3)) = 2

Giới hạn trên: min(30, Q3 + 1.5 \* IQR) = min(30, 8 + 1.5 \* (8 - 3)) = 15.5

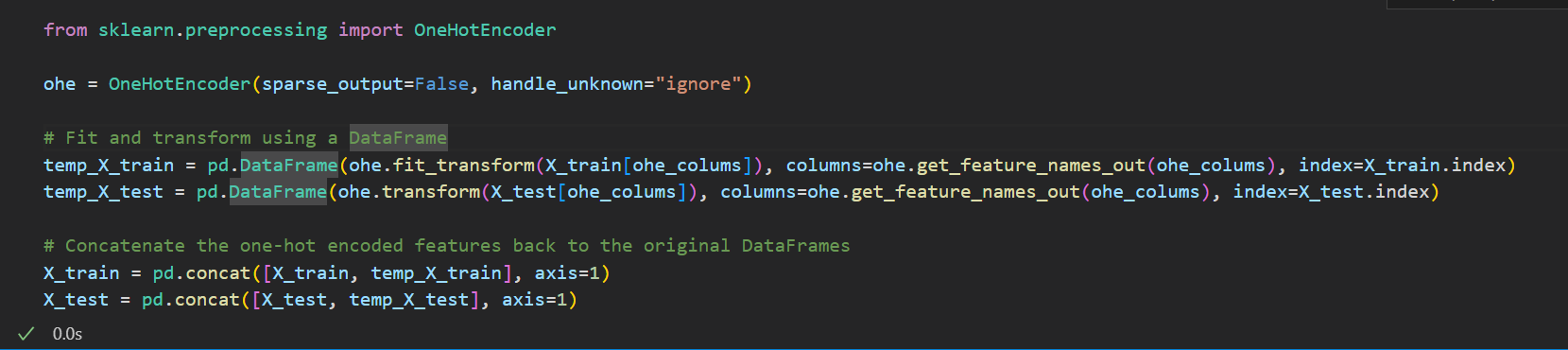
**Visualize các feature:**

**** ****    

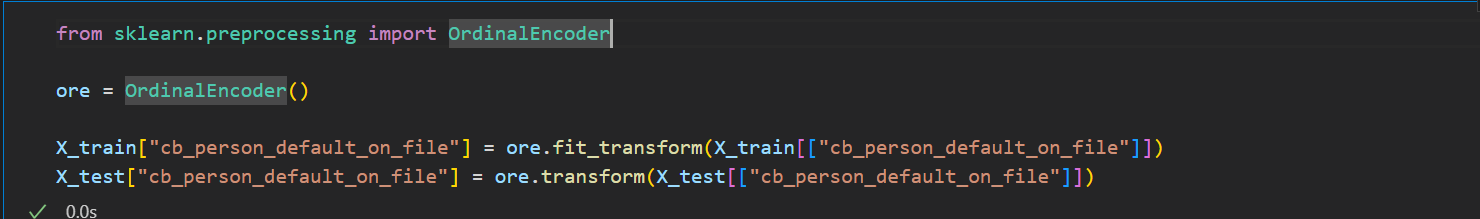
Tổng kết: Ở đây tất cả các dữ liệu đều không có outliers nên không cần phải xử lý.

### 2.5. Tiền xử lý dữ liệu (Data Preprocessing)

#### **2.5.1. Áp dụng One Hot Encoder trên các cột feature có nhiều hơn 2 category**



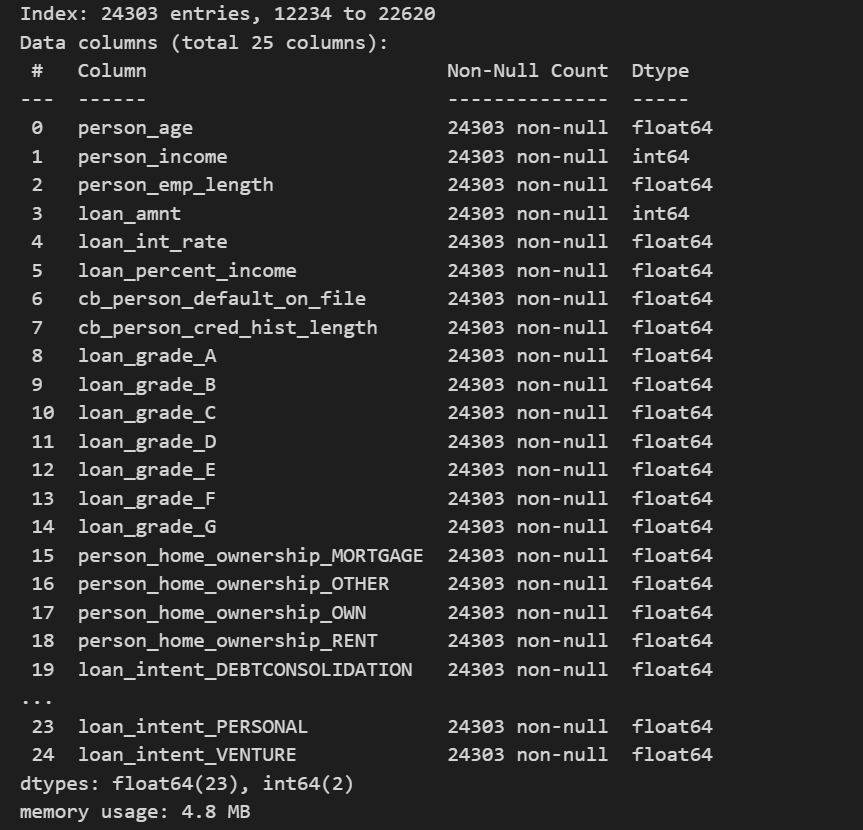
#### **2.5.2. Áp dụng Ordinal Encoder trên một cột feature chỉ có 2 category**



Sự khác biệt giữa OneHotEncoder và OrdinalEncoder:

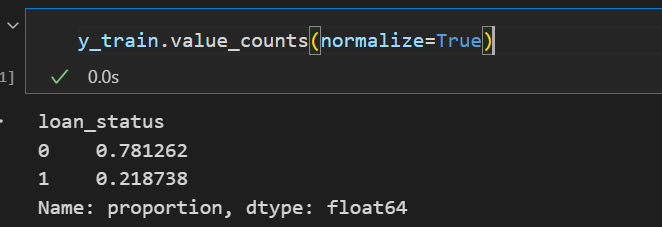
* OneHotEncoder: Chuyển đổi các giá trị phân loại thành các vector nhị phân (binary vectors). Với mỗi giá trị phân loại, OneHotEncoder tạo ra một cột riêng biệt. Nếu giá trị xuất hiện, cột tương ứng sẽ có giá trị là 1, còn lại sẽ là 0.
* OrdinalEncoder: Chuyển đổi các giá trị phân loại thành các số nguyên. Mỗi giá trị phân loại được ánh xạ đến một số nguyên duy nhất, theo một thứ tự cố định.

**View lại dataframe:**

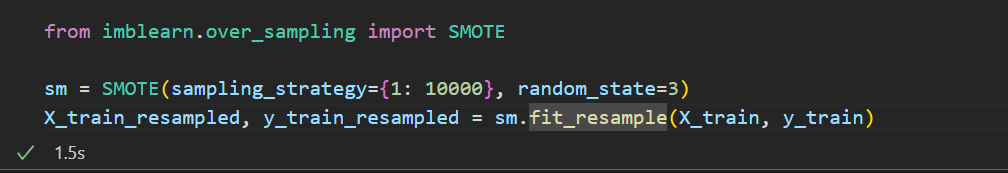
****

#### **2.5.3. Cân bằng dữ liệu bằng SMOTE**

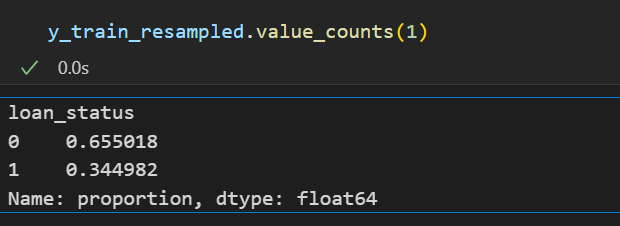
Xem tần suất xuất hiện các lớp trong y\_train:



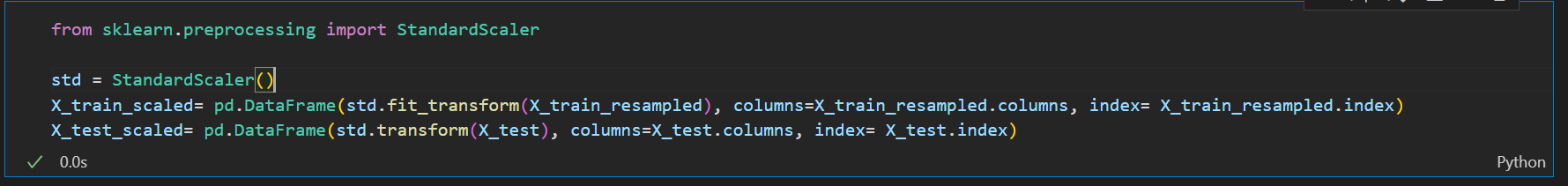
Vì dữ liệu ở đây bị imbalanced, nên ở đây mình sẽ sử dụng Kỹ thuật lấy mẫu quá mức thiểu số tổng hợp (SMOTE) để giải quyết vấn đề này. Hiện tại, tỷ lệ giữa các lớp đa số và thiểu số là 78:22. Sau khi áp dụng SMOTE, tỷ lệ này dự kiến ​​sẽ được điều chỉnh thành khoảng 66:34.



Vì sao ở đây mình lấy lớp 1 thêm 10000 mẫu tại vì không phải lúc nào chia 50 50 là cũng tốt mà cần phải dựa trên tập dữ liệu nếu chia 50 50 có thể dẫn đến sự sai lệch phân bố thực tế và tăng nguy cơ overfiting.



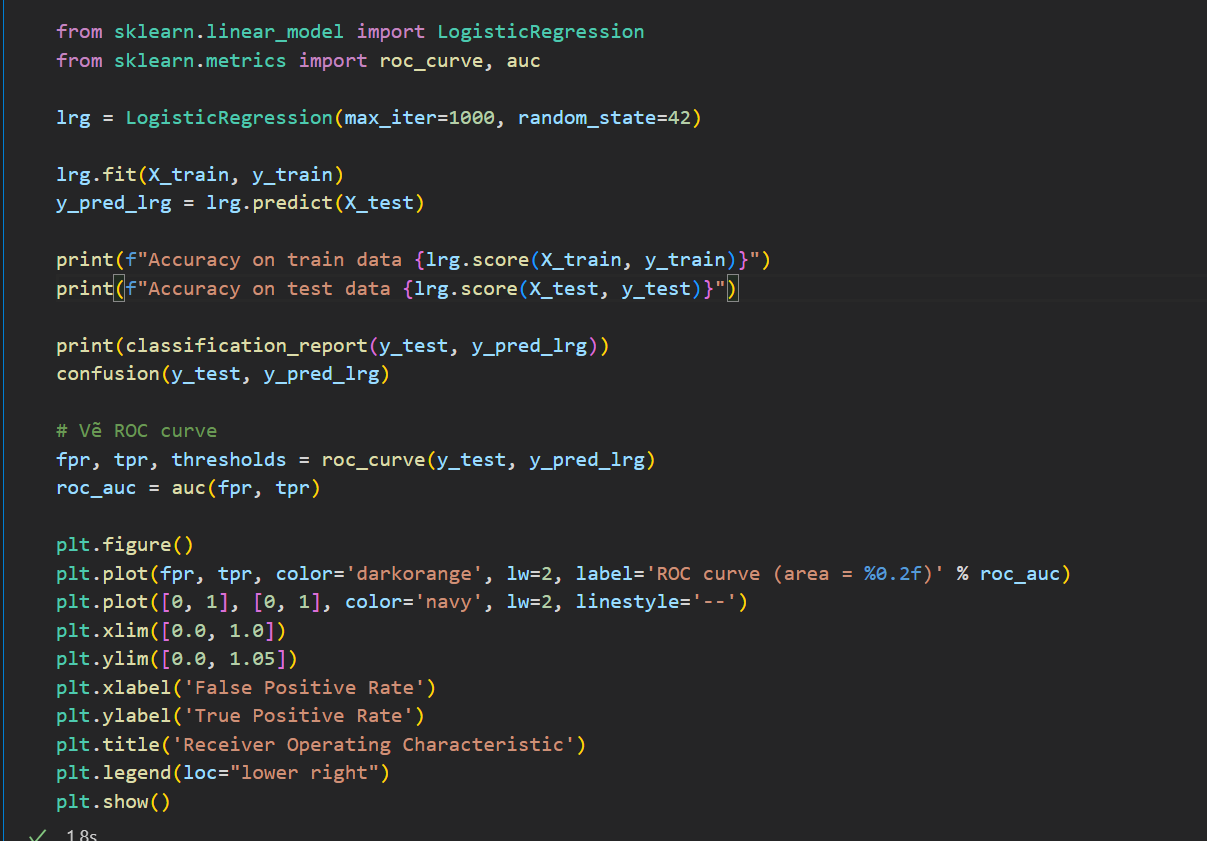
#### **2.5.4. Chuẩn hóa dữ liệu**



### 2.6. Các mô hình phân loại Machine Learning

#### **2.6.1. Logistic Regression**

**Mô hình Logistic Regression trước khi áp dụng SMOTE**

****

Accuracy on train data 0.8406781055836728

Accuracy on test data 0.8423651401061597

precision recall f1-score support

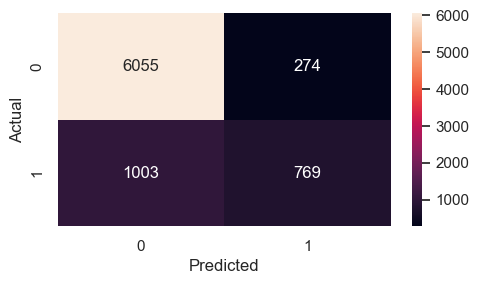
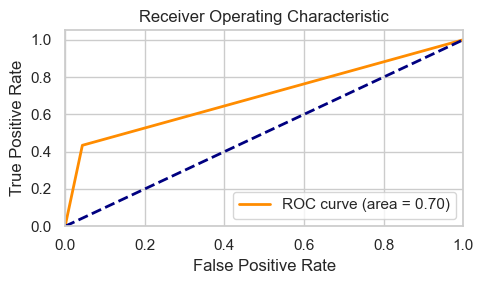
0 0.86 0.96 0.90 6329

1 0.74 0.43 0.55 1772

accuracy 0.84 8101

macro avg 0.80 0.70 0.73 8101

weighted avg 0.83 0.84 0.83 8101

**** ****

**Mô hình Logistic Regression sau khi áp dụng SMOTE**

Accuracy on train data 0.799137187230371

Accuracy on test data 0.8291568942105912

precision recall f1-score support

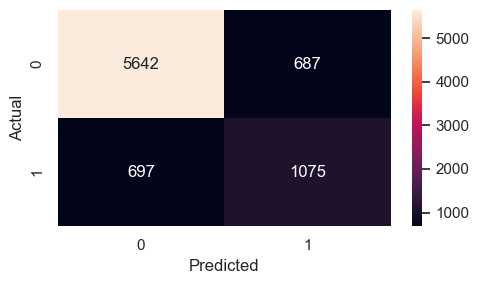
0 0.89 0.89 0.89 6329

1 0.61 0.61 0.61 1772

accuracy 0.83 8101

macro avg 0.75 0.75 0.75 8101

weighted avg 0.83 0.83 0.83 8101

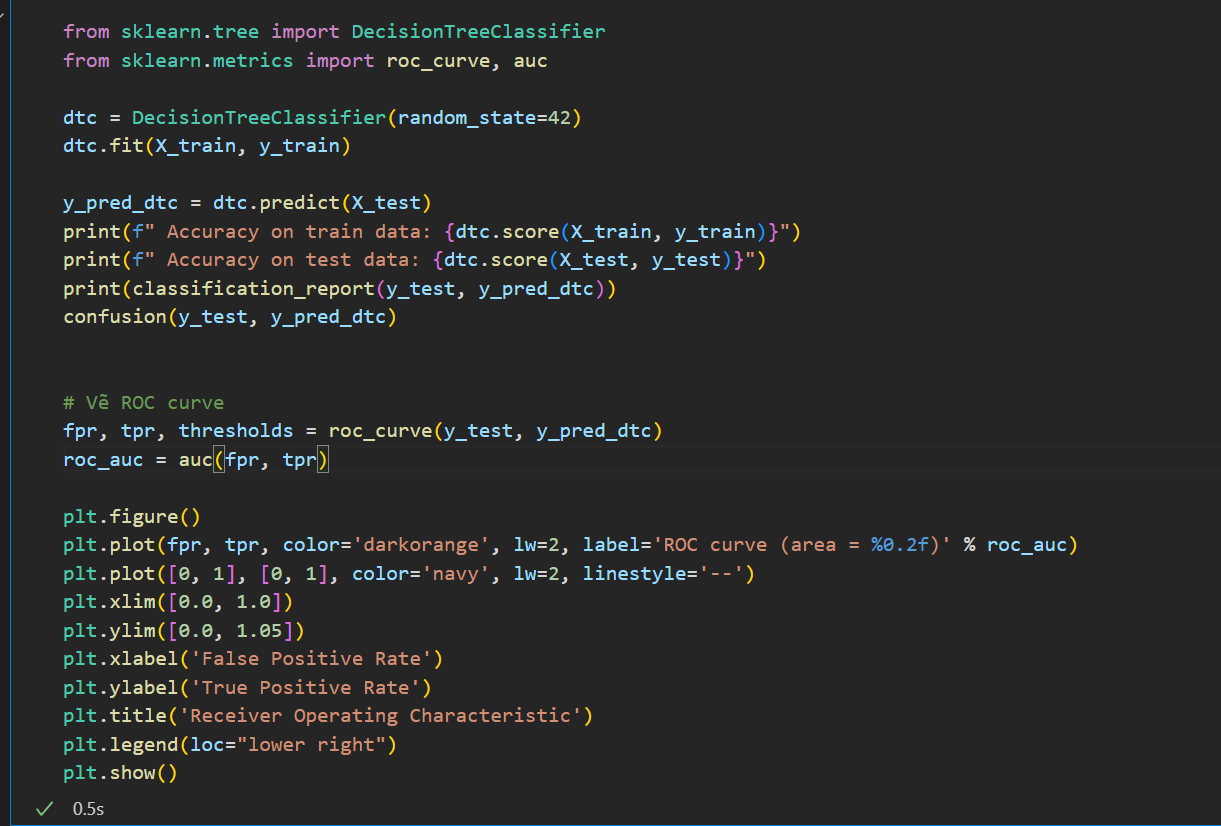


Đánh giá:

* Trước khi áp dụng SMOTE, mô hình đạt độ chính xác khoảng 84% trên dữ liệu thử nghiệm, với độ chính xác là 0,74 đối với lớp 1 và 0,86 đối với lớp 0. Tuy nhiên, sau khi áp dụng SMOTE, độ chính xác giảm xuống khoảng 82% trên dữ liệu thử nghiệm, với độ chính xác là 0,61 đối với lớp 1 và 0,89 đối với lớp 0.
* Ngoài ra, khả năng thu hồi và điểm F1 đối với lớp 1 bị giảm sau khi áp dụng SMOTE, cho thấy hiệu suất kém trong việc xác định các trường hợp của lớp thiểu số.

#### **2.6.2. Decision Tree**

**Mô hình Decision Tree trước khi áp dụng SMOTE**

****

Accuracy on train data: 1.0

Accuracy on test data: 0.8844587088013826

precision recall f1-score support

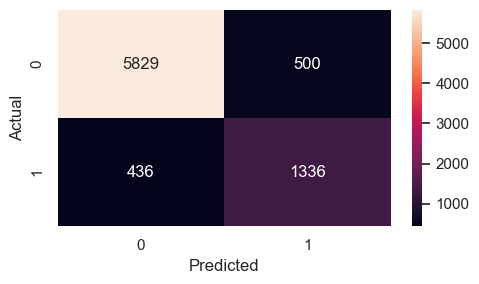
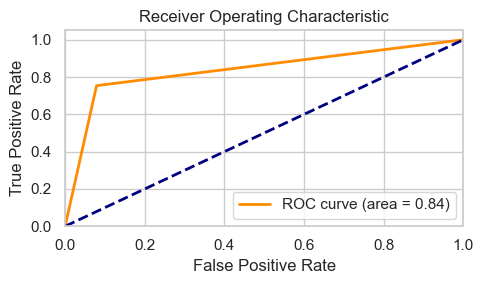
0 0.93 0.92 0.93 6329

1 0.73 0.75 0.74 1772

accuracy 0.88 8101

macro avg 0.83 0.84 0.83 8101

weighted avg 0.89 0.88 0.89 8101

**** ****

**Mô hình Decision Tree sau khi áp dụng SMOTE**

Accuracy on train data: 1.0

Accuracy on test data: 0.8793976052339217

precision recall f1-score support

0 0.93 0.91 0.92 6329

1 0.71 0.75 0.73 1772

accuracy 0.88 8101

macro avg 0.82 0.83 0.83 8101

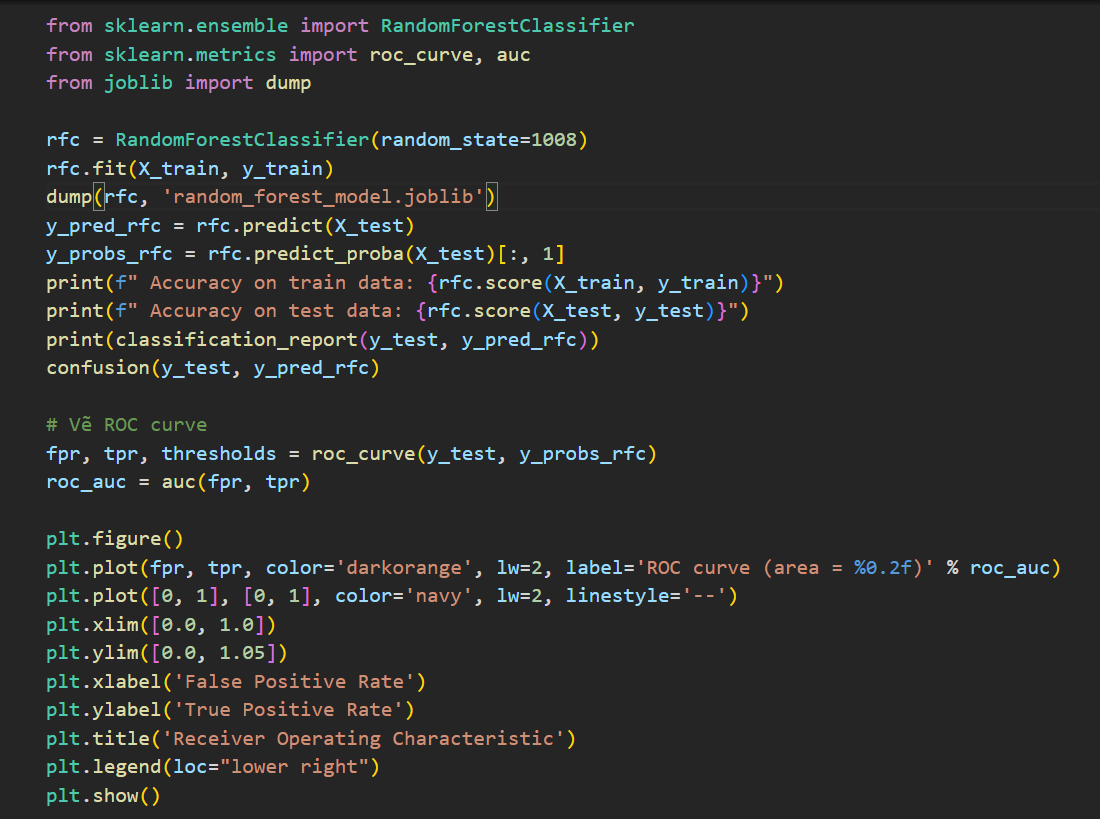
weighted avg 0.88 0.88 0.88 8101

Đánh giá:

* Mô hình cây quyết định cho thấy dấu hiệu rõ ràng của tình trạng quá khớp (overfitting), như thấy ở độ chính xác cao trên dữ liệu đào tạo nhưng hiệu suất thấp hơn trên dữ liệu thử nghiệm.
* Ngay cả sau khi giải quyết mất cân bằng lớp với SMOTE, vấn đề quá khớp vẫn tồn tại.
* Do đó, chỉ dựa vào mô hình cây quyết định có thể không phải là cách tiếp cận phù hợp.

#### **2.6.3. Random Forest**

**Mô hình Random Forest trước khi áp dụng SMOTE**



Accuracy on train data: 1.0

Accuracy on test data: 0.9253178619923467

precision recall f1-score support

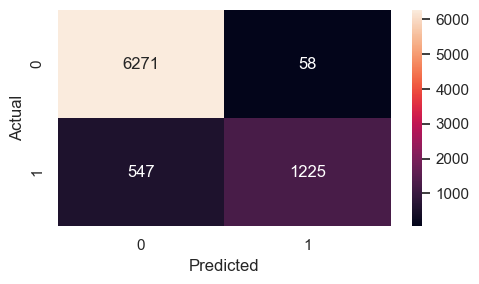
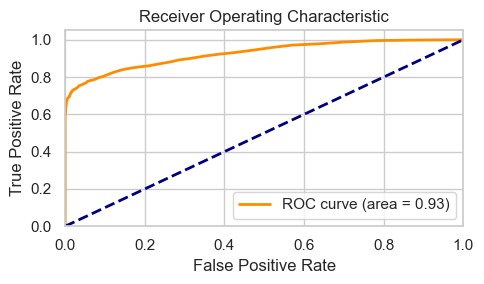
0 0.92 0.99 0.95 6329

1 0.95 0.69 0.80 1772

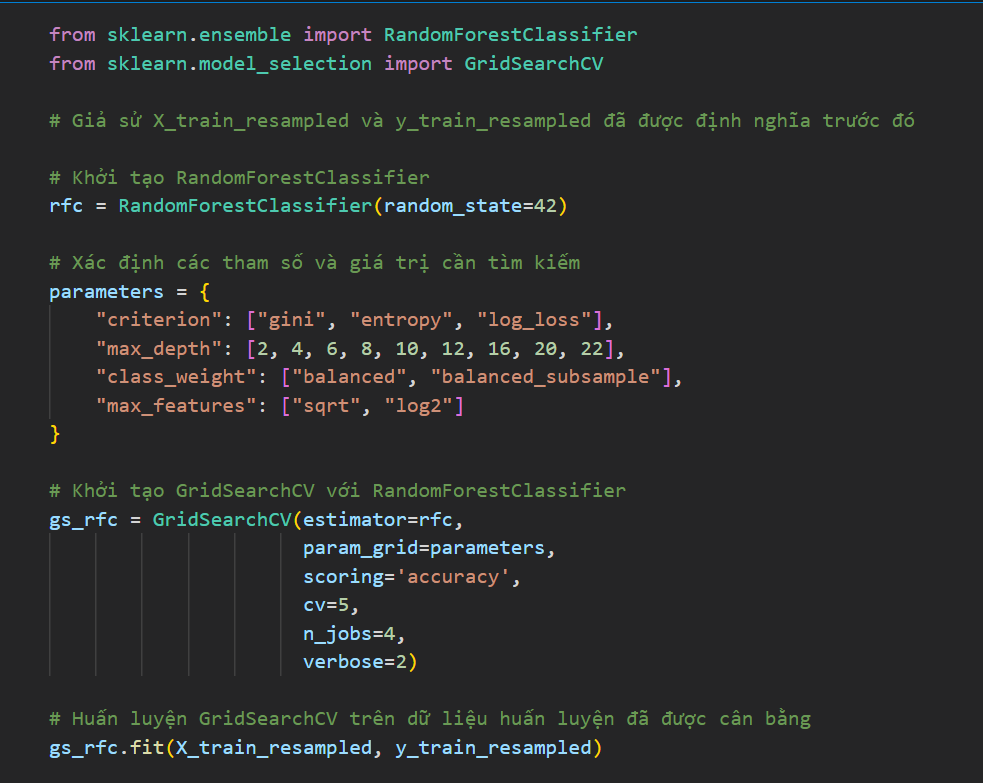
accuracy 0.93 8101

macro avg 0.94 0.84 0.88 8101

weighted avg 0.93 0.93 0.92 8101

**Tìm kiếm các tham số tối ưu cho mô hình Random Forest dựa trên phương pháp GridSearchCV**

****

Fitting 5 folds for each of 108 candidates, totalling 540 fits

Best parameters found: {'class\_weight': 'balanced', 'criterion': 'entropy', 'max\_depth': 22, 'max\_features': 'sqrt'}

Accuracy on test data: 0.9254413035427725

precision recall f1-score support

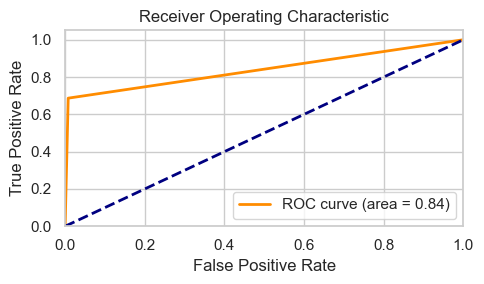
0 0.92 0.99 0.95 6329

1 0.96 0.69 0.80 1772

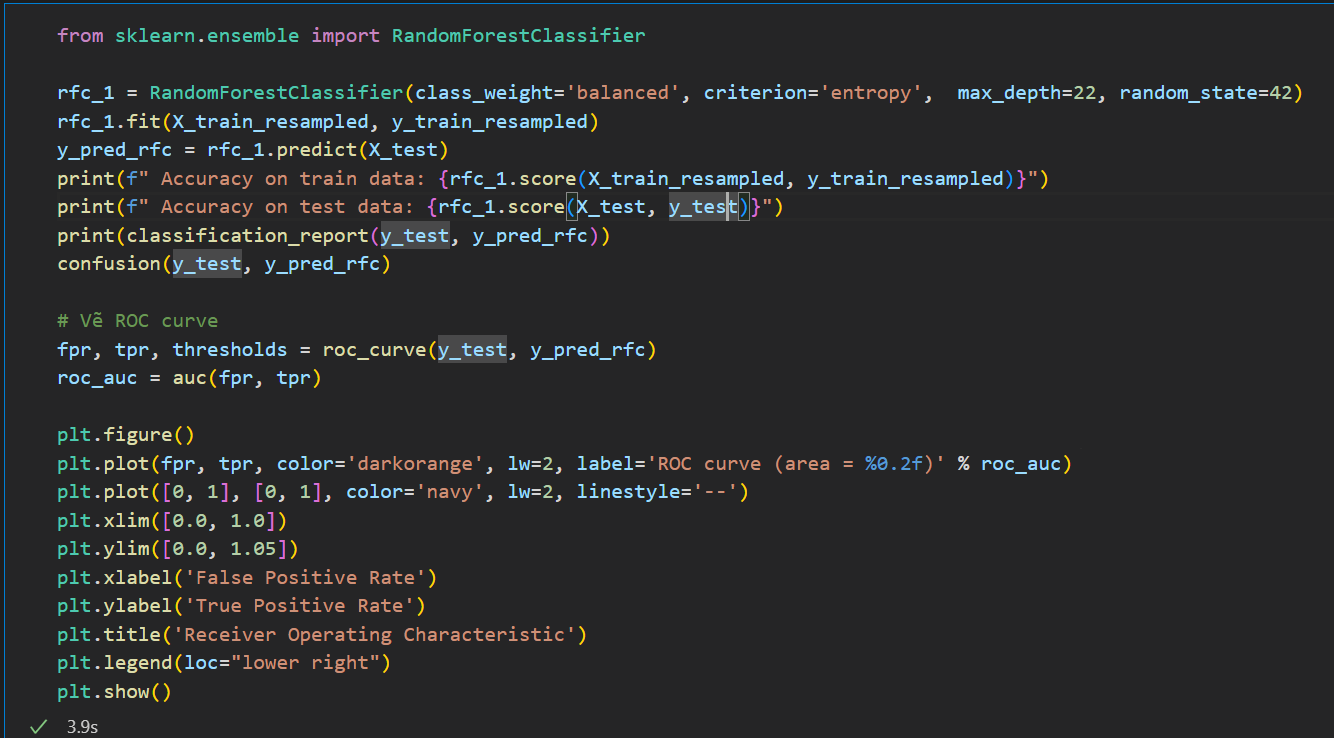
accuracy 0.93 8101

macro avg 0.94 0.84 0.88 8101

weighted avg 0.93 0.93 0.92 8101

****

**Sử dụng siêu tham số train mô hình Random Forest**

****

Accuracy on train data: 0.9930974978429681

Accuracy on test data: 0.9254413035427725

precision recall f1-score support

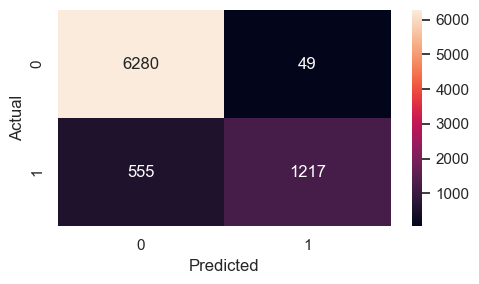
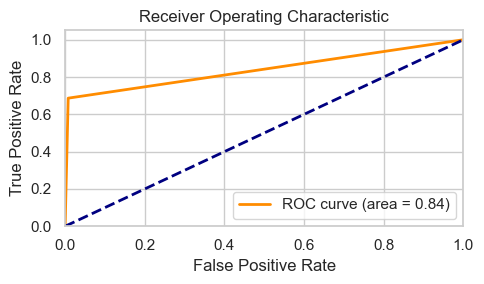
0 0.92 0.99 0.95 6329

1 0.96 0.69 0.80 1772

accuracy 0.93 8101

macro avg 0.94 0.84 0.88 8101

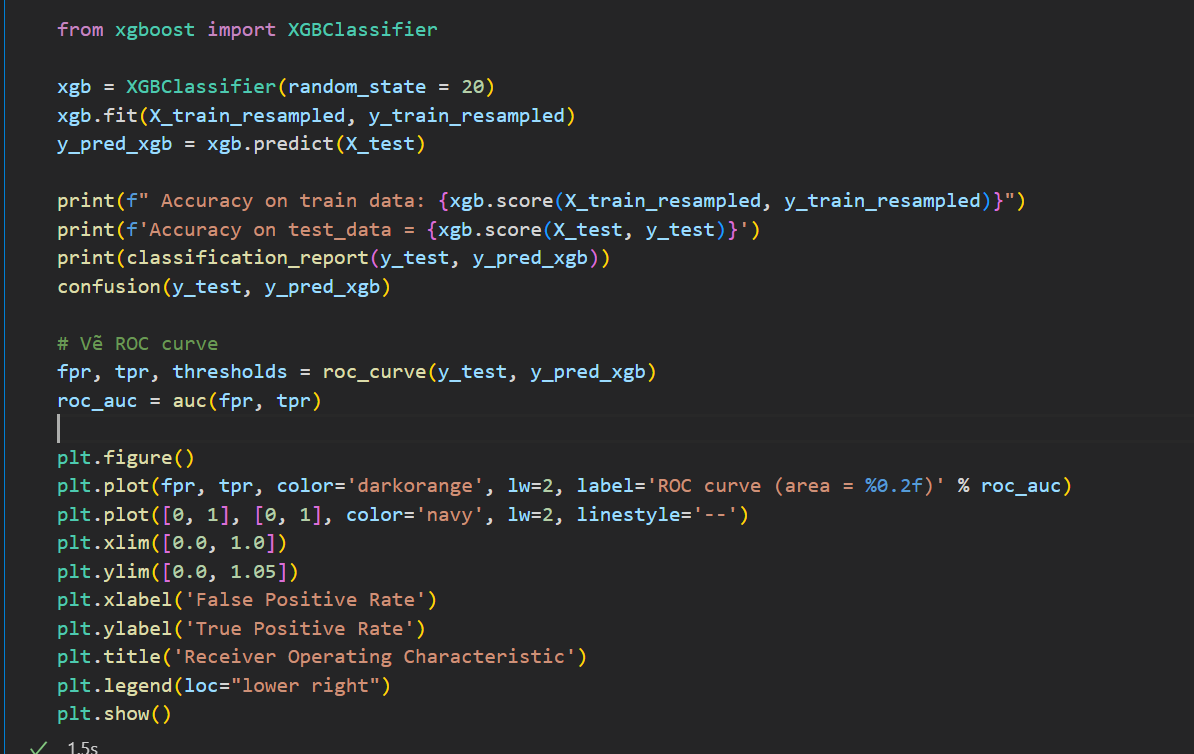
weighted avg 0.93 0.93 0.92 8101

Đánh giá:

* Random Forest được áp dụng cho dữ liệu trước khi SMOTE cho thấy sự quá khớp (Overfitting) trên dữ liệu đào tạo.
* Sử dụng siêu tham số với mô hình Random Forest trên dữ liệu được xử lý bằng SMOTE mang lại kết quả tương tự như mô hình trước đó, với độ chính xác giảm nhẹ trên dữ liệu đào tạo so với dữ liệu thử nghiệm. Tuy nhiên, Overfittinh vẫn là một mối quan tâm.

#### **2.6.4. XGBoost (Extreme Gradient Boosting)**



Accuracy on train data: 0.961518550474547

Accuracy on test\_data = 0.931860264164918

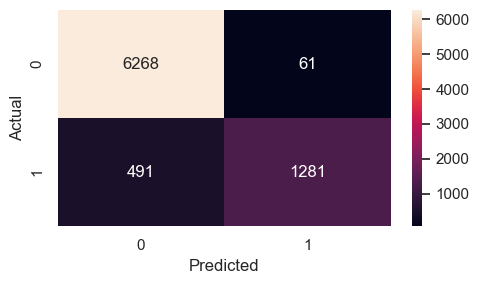
precision recall f1-score support

0 0.93 0.99 0.96 6329

1 0.95 0.72 0.82 1772

accuracy 0.93 8101

macro avg 0.94 0.86 0.89 8101

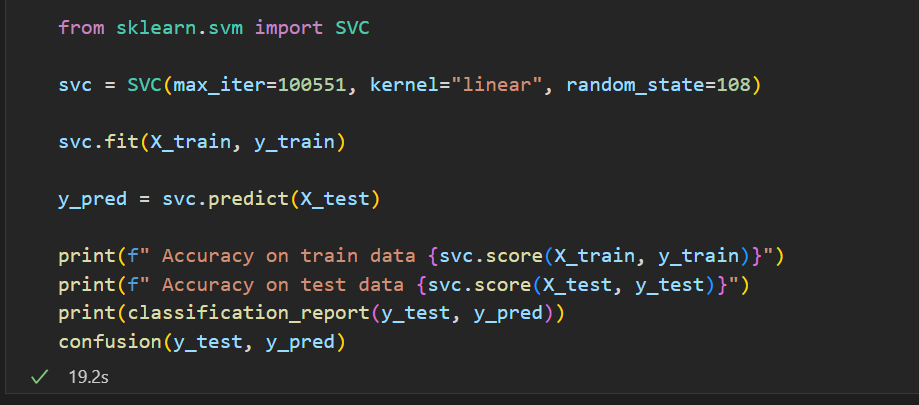
weighted avg 0.93 0.93 0.93 8101  

Đánh giá:

* Mô hình có accuracy cao trên cả dữ liệu huấn luyện và kiểm tra, cho thấy mô hình hoạt động khá tốt và không bị overfitting nghiêm trọng. Tuy nhiên, accuracy cao trên tập huấn luyện so với tập kiểm tra có thể chỉ ra một chút overfitting, nhưng mức độ là tương đối thấp.

#### **2.6.5. SVM (Super Vector Machine)**

**Mô hình SVM trước khi áp dụng SMOTE**

****

Accuracy on train data 0.4904332798419948

Accuracy on test data 0.4798173065053697

precision recall f1-score support

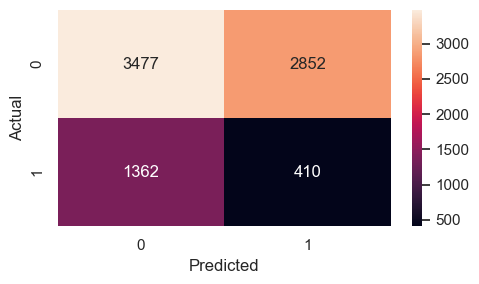
0 0.72 0.55 0.62 6329

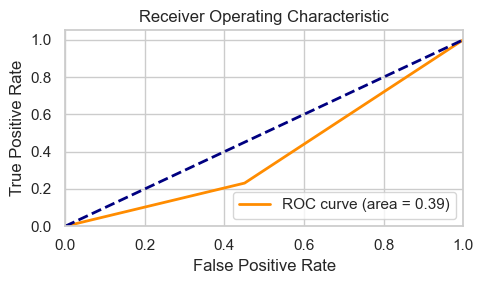
1 0.13 0.23 0.16 1772

accuracy 0.48 8101

macro avg 0.42 0.39 0.39 8101

weighted avg 0.59 0.48 0.52 8101





**Mô hình SVM sau khi áp dụng SMOTE**

Accuracy on train data 0.5420880069025021

Accuracy on test data 0.21873842735464757

precision recall f1-score support

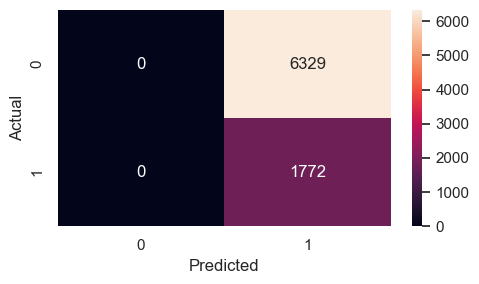
0 0.00 0.00 0.00 6329

1 0.22 1.00 0.36 1772

accuracy 0.22 8101

macro avg 0.11 0.50 0.18 8101

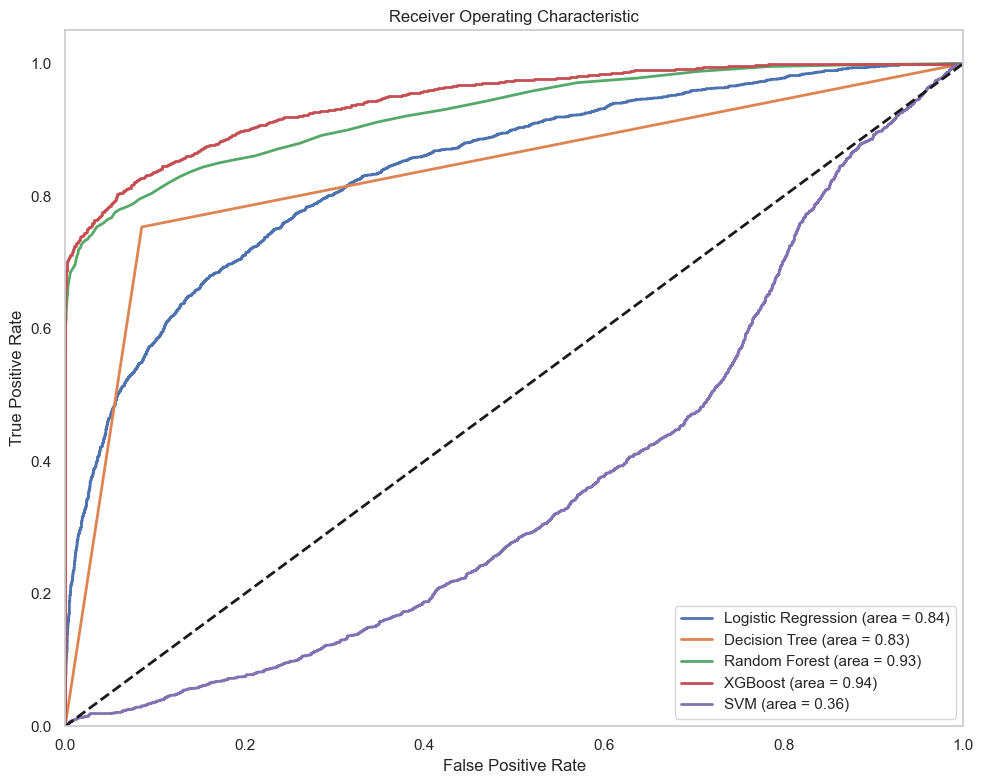
weighted avg 0.05 0.22 0.08 8101



Đánh giá:

* Thuật toán của không hội tụ trong số lần lặp tối đa cho phép. Thuật toán này không phù hợp cho bài toán này

## 3. Đánh giá mô hình



1. Trục tọa độ:

* Trục X (False Positive Rate - Tỷ lệ dương tính giả): Tỷ lệ các trường hợp âm tính bị phân loại sai thành dương tính.
* Trục Y (True Positive Rate - Tỷ lệ dương tính thật): Tỷ lệ các trường hợp dương tính được phân loại chính xác.

1. Các đường cong:

* **Logistic Regression** (diện tích = 0.84): Mô hình này có diện tích dưới đường cong (AUC) là 0.84, cho thấy hiệu suất khá tốt.
* **Decision Tree** (diện tích = 0.83): Mô hình này có AUC là 0.83, hơi thấp hơn một chút so với Logistic Regression.
* **Random Forest** (diện tích = 0.93): Mô hình này có AUC cao là 0.93, cho thấy hiệu suất rất tốt.
* **XGBoost** (diện tích = 0.94): Đây là mô hình có hiệu suất tốt nhất với AUC là 0.94.
* **SVM** (diện tích = 0.36): Mô hình này có AUC rất thấp, chỉ 0.36, cho thấy hiệu suất kém.

Nhận xét chung:

* Mô hình **XGBoost** và **Random Forest** là hai mô hình có hiệu suất cao nhất trong nhóm này.
* **SVM** có hiệu suất kém hơn hẳn so với các mô hình khác.
* **Logistic Regression** và **Decision Tree** có hiệu suất tương đối tương đồng, nhưng thấp hơn so với **XGBoost** và **Random Forest**.

# CHƯƠNG 4

# Kết luận và kiến nghị

## 1. Kết quả đạt được:

## 2. Đề xuất và kiến nghị:

## 3. Kết luận: