TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO CUỐI KÌ MÔN**

**NHẬP MÔN HỌC MÁY**

*Người hướng dẫn*: **PGS.TS LÊ ANH CƯỜNG**

*Người thực hiện*: **NGUYỄN THỊ THU YẾN - 52000869**

**NGUYỄN KHẮC VĂN - 52000868**

**TRƯƠNG THÔNG THẾ THÁI - 52000714**

Lớp **: 20050401**

Khoá  **: 24**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO CUỐI KÌ MÔN**

**NHẬP MÔN HỌC MÁY**

*Người hướng dẫn*: **PGS.TS LÊ ANH CƯỜNG**

*Người thực hiện*: **NGUYỄN THỊ THU YẾN - 52000869**

**NGUYỄN KHẮC VĂN - 52000868**

**TRƯƠNG THÔNG THẾ THÁI - 52000714**

Lớp **: 20050401**

Khoá  **: 24**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

LỜI CẢM ƠN

Nhóm chúng em xin gửi lời cảm ơn chân thành đến khoa công nghệ thông tin, nhà trường và đặc biệt là thầy **(PGS.TS) Lê Anh Cường** đã tận tình hướng dẫn nhóm em trong quá trình giảng dạy bộ môn Nhập môn Học máy (Machine learning)

Nhờ sự hướng dẫn và chỉ bảo tận tình của thầy, nhóm em đã có thể hoàn thành bài báo cáo của mình một cách chu đáo và đầy đủ. Nhóm em đã học hỏi được rất nhiều kiến thức và kinh nghiệm từ thầy, giúp nhóm có thêm hành trang và kỹ năng để áp dụng vào công việc sau này.

Em xin chân thành cảm ơn thầy đã dành thời gian quý báu để góp ý và chỉnh sửa bài báo cáo của nhóm. Nhóm em sẽ tiếp tục nỗ lực học tập và nghiên cứu để đạt được những thành tích tốt hơn trong tương lai.

Một lần nữa, nhóm em xin chân thành cảm ơn khoa, nhà trường và thầy.

Trân trọng!

**ĐỒ ÁN ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm đồ án của riêng tôi và được sự hướng dẫn của PGS.TS Lê Anh Cường. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày 22 tháng 10 năm 2023*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Nguyễn Khắc Văn*

*Nguyễn Thị Thu Yến*

*Trương Thông Thế Thái*

# PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

**Phần xác nhận của GV hướng dẫn**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

TÓM TẮT

**Bài 1 (3 điểm): làm riêng từng người**

Trình bày một bài nghiên cứu, đánh giá của em về các vấn đề sau:

1. Tìm hiểu, so sánh các phương pháp Optimizer trong huấn luyện mô hình học máy;
2. Tìm hiểu về Continual Learning và Test Production khi xây dựng một giải pháp học máy để giải quyết một bài toán nào đó.

**Bài 2: (7 điểm): làm chung trong nhóm**

Đưa ra một bài toán dự đoán có thể giải quyết bằng học máy (machine leanring) với các yêu cầu sau:

* Số Feature/Attribute gồm nhiều kiểu: categorial và numerical;
* Dữ liệu phải chưa được học, thực tập trên lớp và trong bài tập về nhà;

1. Phân tích thống kê trên dữ liệu, vẽ các đồ thị để hiểu bài toán, hiểu dữ liệu. Tìm hiểu các đặc trưng và đánh gía vai trò của các đặc trưng đối với mục tiêu bài toán;
2. Ứng dụng các mô hình học máy cơ bản để giải quyết bài toán, bao gồm cả các mô hình thuộc Ensemble Learing;
3. Sử dụng Feed Forward Neural Network và Reccurent Neural Network (hoặc mô thuộc loại này) để giải quyết bài toán;
4. Áp dụng các kỹ thuật tránh Overfiting trên các mô hình của câu (2) và câu (3) để giải quyết bài toán;
5. Sau khi huấn luyện xong mô hình thì muốn cải thiện độ chính xác, ta sẽ làm gì để giải quyết nó? Phân tích các trường hợp sai, đề ra giải pháp và thực hiện nó, sau đó đánh giá xem có cải tiến so với trước không.

MỤC LỤC

[LỜI CẢM ƠN 1](#_Toc154259004)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN 3](#_Toc154259005)

[TÓM TẮT 4](#_Toc154259006)

[MỤC LỤC BẢNG BIỂU, HÌNH VẼ 8](#_Toc154259007)

[PHẦN 1 : NGHIÊN CỨU CÁ NHÂN 9](#_Toc154259008)

[**1.** **Các phương pháp tối ưu hóa mô hình học máy** 9](#_Toc154259009)

[**1.1** **Khái niệm** 9](#_Toc154259010)

[**1.2** **Cách hoạt động** 9](#_Toc154259011)

[**1.3** **Các phương pháp tối ưu hóa phổ biến trong Machine Learning** 11](#_Toc154259012)

[**1.4** **So sánh các thuật toán tối ưu** 18](#_Toc154259013)

[**1.5** **Xây dụng mô hình học máy và sử dụng các trình tối ưu** 19](#_Toc154259014)

[**2.** **Continual Learning** 22](#_Toc154259015)

[**2.1** **Continual Learning** 22](#_Toc154259016)

[**2.2** **Các yếu tố chính trong quá trình học liên tục** 22](#_Toc154259017)

[**2.3** **Một số phương pháp học liên tục (Continual learning)** 23](#_Toc154259018)

[**2.4** **Lợi ích của Continual learning** 24](#_Toc154259019)

[**2.5** **Những hạn chế** 25](#_Toc154259020)

[**3.** **Test Production** 26](#_Toc154259021)

[**3.1** **Khái niệm** 26](#_Toc154259022)

[**3.2** **Các lợi ích** 26](#_Toc154259023)

[**4.** **Xây dụng mô hình học máy sử dụng Continual Learning và Test Production** 27](#_Toc154259024)

[PHẦN 2 – CÂU 2 : XÂY DỰNG MÔ HÌNH HỌC MÁY – DỰ ĐOÁN PHÊ DUYỆT THẺ TÍN DỤNG TỰ ĐỘNG 31](#_Toc154259025)

[**1.** **Giới thiệu dữ liệu sử dụng cho bài toán** 31](#_Toc154259026)

[**1.1 Dữ liệu: Credit Card Approval Prediction** 31](#_Toc154259027)

[**1.2** **Đánh giá các feature của dữ liệu** 41](#_Toc154259028)

[2. Xây dựng mô hình học máy 42](#_Toc154259029)

[**2.1 Các mô hình sử dụng** 42](#_Toc154259030)

[**2.2 Phòng tránh Overfitting** 43](#_Toc154259031)

[**2.** **Thực hiện bài toán với các mô hình học máy** 44](#_Toc154259032)

[PHẦN 3 – KẾT LUẬN 76](#_Toc154259033)

[**1.** **Kiến thức sau nghiên cứu, tìm hiểu** 76](#_Toc154259034)

[**2.** **Những điều cần lưu ý** 76](#_Toc154259035)

[TÀI LIỆU THAM KHẢO 78](#_Toc154259036)

# MỤC LỤC BẢNG BIỂU, HÌNH VẼ

# 

[Hình 1 Dữ liệu chung của tệp application\_record.csv 36](#_Toc154258963)

[Hình 2 Dữ liệu chung của tệp credit\_record.csv 37](#_Toc154258964)

[Hình 3 Thống kê số lượng khách hàng 38](#_Toc154258965)

[Hình 4 Số liệu giới tính và sở hữu ô tô 38](#_Toc154258966)

[Hình 5 Số liệu sở hữu tài sản 38](#_Toc154258967)

[Hình 6 Thống kê số lượng trẻ em 39](#_Toc154258968)

[Hình 7 Thống kê thu nhập hàng năm 39](#_Toc154258969)

[Hình 8 Thống kê loại thu nhập 39](#_Toc154258970)

[Hình 9 Thống kê trình độ học vấn 40](#_Toc154258971)

[Hình 10 Thống kê số lượng khách hàng 40](#_Toc154258972)

[Hình 11 Thống kê theo tháng 40](#_Toc154258973)

[Hình 12 Thống kê theo trạng thái ghi nợ 41](#_Toc154258974)

[Hình 13 Kết quả học mô hình Logistic Regression 62](#_Toc154258975)

[Hình 14 Kết quả học mô hình Decision Tree 63](#_Toc154258976)

[Hình 15 Kết quả học mô hình Random Forests 65](#_Toc154258977)

[Hình 16 Kết quả học với KNN 67](#_Toc154258978)

[Hình 17 Mô hình Random forest phòng tránh Overfitting 70](#_Toc154258979)

[Hình 18 Tạo mô hình FNN có sử dụng tránh Overfiting 71](#_Toc154258980)

[Hình 19 Tạo mô hình RNN có sử dụng tránh Overfiting 71](#_Toc154258981)

# PHẦN 1 : NGHIÊN CỨU CÁ NHÂN

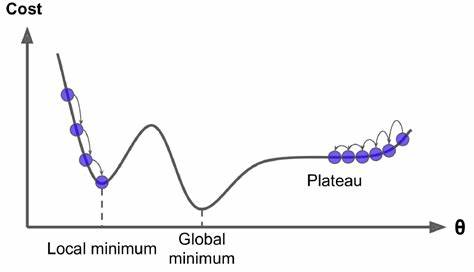
1. **Các phương pháp Optimize (tối ưu hóa) mô hình học máy**
   1. **Khái niệm**
   * Trong lĩnh vực học máy, tối ưu hóa là một thành phần quan trọng được sử dụng để điều chỉnh các tham số của mô hình nhằm giảm thiểu lỗi hoặc tối đa hóa hiệu suất của thuật toán. Nó đóng một vai trò quan trọng trong quá trình training mô hình học máy bằng cách tối ưu hóa trọng số và độ lệch của mô hình dựa trên dữ liệu đầu vào.
   * Mục tiêu của trình tối ưu hóa là tìm ra bộ tham số tối ưu giúp giảm thiểu hàm mất mát, định lượng lỗi giữa các giá trị dự đoán của mô hình và giá trị thực tế. Bằng cách điều chỉnh lặp đi lặp lại các tham số, trình tối ưu hóa sẽ điều khiển mô hình theo hướng hội tụ, trong đó sai số được giảm thiểu và mô hình đưa ra dự đoán chính xác nhất.
   * Quá trình tối ưu hóa có thể được coi là điều hướng một cảnh quan phức tạp với nhiều đỉnh và valleys, trong đó mỗi đỉnh biểu thị một cấu hình khác nhau của các tham số của mô hình. Công việc của trình tối ưu hóa là xác định mức tối thiểu toàn cục hoặc cục bộ, đại diện cho cấu hình tốt nhất cho mô hình.
   * Có nhiều thuật toán tối ưu hóa khác nhau, mỗi thuật toán đều có điểm mạnh và điểm yếu riêng. Các thuật toán này được thiết kế để tìm kiếm hiệu quả không gian tham số và hội tụ đến một giải pháp tối ưu. Các trình tối ưu hóa dựa trên độ dốc, chẳng hạn như Gradient Descent và các biến thể của nó, thường được sử dụng trong học máy do tính đơn giản và hiệu quả của chúng..
   1. **Cách hoạt động**

* Trình tối ưu hóa trong học máy hoạt động bằng cách điều chỉnh lặp đi lặp lại các tham số của mô hình để giảm thiểu lỗi hoặc tối đa hóa hiệu suất của thuật toán học. Nó tuân theo thuật toán tối ưu hóa được xác định trước để điều hướng không gian tham số và tìm cấu hình tối ưu cho mô hình.
* Khi bắt đầu quá trình huấn luyện, mô hình được khởi tạo với các giá trị tham số ngẫu nhiên. Sau đó, trình tối ưu hóa sẽ đánh giá hiệu suất hiện tại của mô hình bằng cách tính hàm mất mát, hàm này định lượng sai số giữa giá trị dự đoán và giá trị thực tế.
* Dựa trên tổn thất được tính toán, trình tối ưu hóa sẽ xác định hướng và cường độ của các cập nhật tham số. Nó thực hiện những điều chỉnh nhỏ về trọng số và độ lệch của mô hình để giảm thiểu tổn thất qua mỗi lần lặp. Quá trình này được lặp lại cho đến khi mô hình hội tụ về một giải pháp tối ưu, trong đó tổn thất được giảm thiểu và mô hình đạt được hiệu suất thỏa đáng.
* Trình tối ưu hóa sử dụng khái niệm gradient, là dẫn xuất của hàm mất mát đối với các tham số. Bằng cách tính toán độ dốc, trình tối ưu hóa sẽ xác định độ dốc của hàm mất mát và cập nhật các tham số tương ứng. Đây được gọi là thuật toán giảm độ dốc.
* Có nhiều biến thể khác nhau của trình tối ưu hóa dựa trên độ dốc, chẳng hạn như Stochastic gradient Descent (SGD), Adam, RMSprop và AdaGrad, áp dụng các chiến lược khác nhau để điều chỉnh các tham số. SGD chọn ngẫu nhiên một tập hợp con các mẫu huấn luyện cho mỗi lần lặp, giúp nó đạt hiệu quả tính toán cao. Adam kết hợp các ưu điểm của cả Ước tính thời điểm thích ứng (Adam) và Tuyên truyền bình phương trung bình gốc (RMSprop) để cải thiện khả năng hội tụ. RMSprop điều chỉnh tốc độ học cho từng tham số dựa trên mức trung bình động của gradient bình phương. AdaGrad điều chỉnh tốc độ học tập dựa trên độ dốc lịch sử cho từng tham số.
* Bằng cách cập nhật hiệu quả các tham số bằng các thuật toán tối ưu hóa này, trình tối ưu hóa sẽ giúp mô hình học hỏi từ dữ liệu huấn luyện và khái quát hóa thành dữ liệu chưa nhìn thấy. Nó đóng một vai trò quan trọng trong sự thành công của các mô hình học máy bằng cách tinh chỉnh các tham số và tối ưu hóa hiệu suất
  1. **Các phương pháp tối ưu hóa phổ biến trong Machine Learning**

Trong lĩnh vực học máy, có một số trình tối ưu hóa phổ biến được sử dụng rộng rãi để huấn luyện các mô hình và tối ưu hóa hiệu suất của chúng. Các trình tối ưu hóa này khác nhau về cách tiếp cận và chiến lược để điều chỉnh các tham số của mô hình. Dưới đây là một số trình tối ưu hóa được sử dụng phổ biến nhất:.

* + 1. **Gradient Descent**

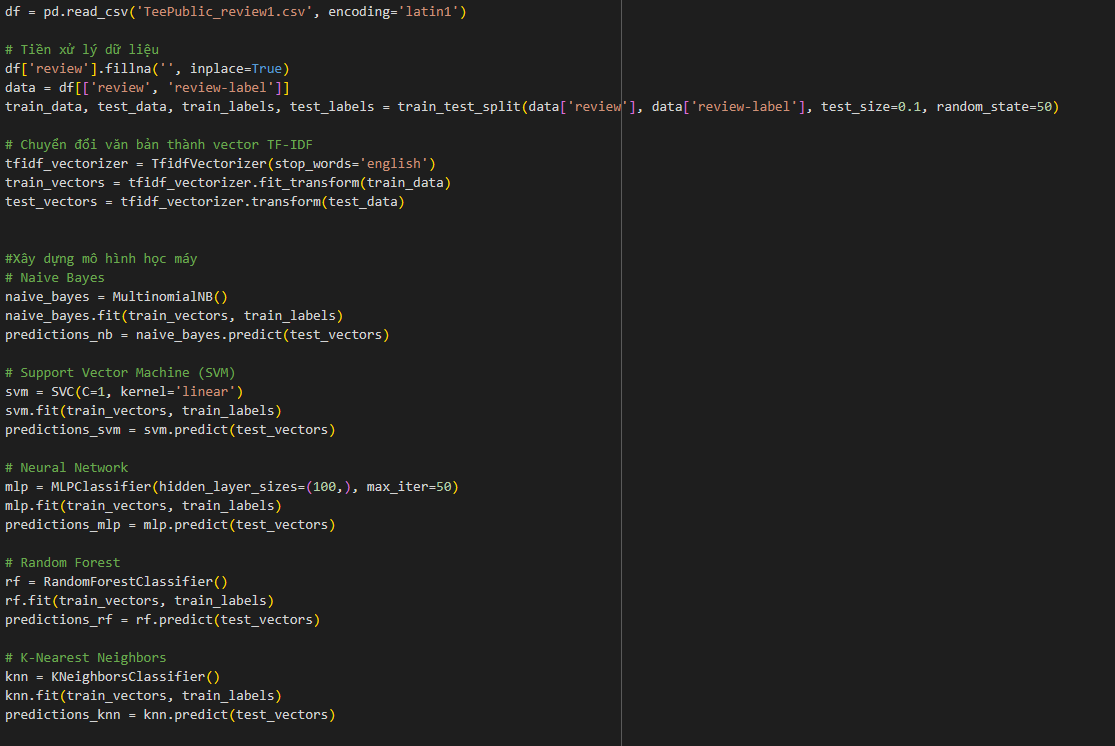
**Gradient Descent** là một thuật toán tối ưu hóa được sử dụng rộng rãi để điều chỉnh các tham số theo hướng ngược lại với độ dốc của hàm mất mát. Nó bắt đầu với các giá trị ban đầu ngẫu nhiên cho các tham số và cập nhật chúng lặp đi lặp lại dựa trên độ dốc được tính toán. Gradient Descent có thể được phân loại thêm thành các biến thể khác nhau, chẳng hạn như Gradient Descent hàng loạt, Gradient Descent hàng loạt nhỏ và Gradient Descent ngẫu nhiên (

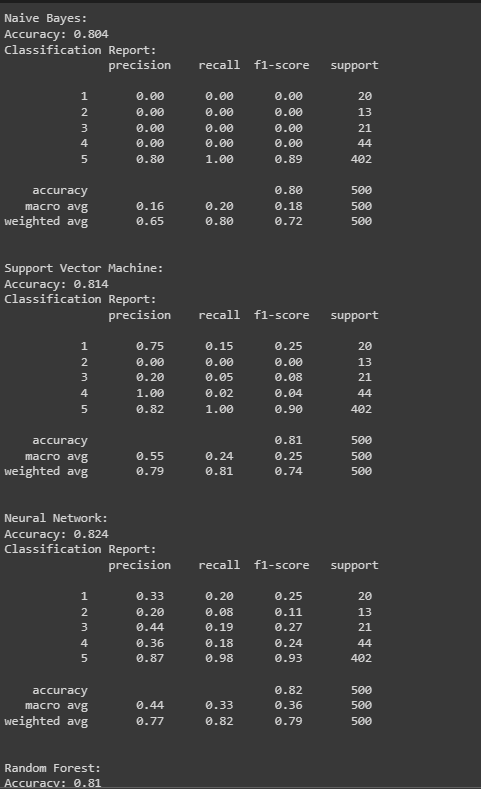


* + 1. **Stochastic Gradient Descent (SGD)**
* Stochastic Gradient Descent ngẫu nhiên (SGD) là một biến thể của Gradient Descent chọn ngẫu nhiên một tập hợp con hoặc các ví dụ trainning cho mỗi lần lặp. Nó cập nhật các tham số dựa trên độ dốc được tính toán từ tập hợp con này. SGD có hiệu quả tính toán và thường hội tụ nhanh hơn so với Gradient Descent truyền thống
* Ưu điểm chính của SGD so với gradient Descent truyền thống là hiệu quả tính toán của nó. Thay vì đánh giá toàn bộ tập dữ liệu huấn luyện trong mỗi lần lặp, SGD chỉ xem xét một tập hợp con nhỏ các ví dụ, thường được gọi là một lô nhỏ. Điều này làm cho nó đặc biệt hữu ích cho các tập dữ liệu lớn không vừa với bộ nhớ.
* Ở mỗi lần lặp, SGD chọn ngẫu nhiên một loạt mẫu huấn luyện nhỏ và tính toán độ suy hao và độ dốc dựa trên tập hợp con này. Sau đó, cập nhật các tham số của mô hình bằng cách sử dụng độ dốc được tính toán từ lô nhỏ. Bằng cách thực hiện liên tục các cập nhật này, SGD hướng mô hình tới mức tối thiểu của hàm mất mát.
* Một đặc điểm quan trọng của SGD là tính chất ngẫu nhiên của nó. Do tính ngẫu nhiên trong việc chọn các lô nhỏ, việc cập nhật tham số trong SGD có thể bị nhiễu và dẫn đến sự biến động của hàm mất mát. Tuy nhiên, tính chất ngẫu nhiên này cũng có thể giúp SGD thoát khỏi cực tiểu cục bộ và khám phá các vùng khác nhau của không gian tham số, dẫn đến khả năng khái quát hóa tốt hơn.
  + 1. **Adam**
* Adam, viết tắt của  **Adaptive Moment Estimation**, là một thuật toán tối ưu hóa tốc độ học thích ứng kết hợp các chiến lược của cả RMSprop và Momentum. Nó điều chỉnh tốc độ học cho từng tham số dựa trên khoảnh khắc bậc nhất (giá trị trung bình) và khoảnh khắc bậc hai (phương sai không tập trung) của độ dốc. Adam được biết đến với đặc tính hội tụ tốt và độ bền cao đối với độ dốc nhiễu hoặc thưa thớt.
* Trình tối ưu hóa Adam điều chỉnh tốc độ học cho từng tham số dựa trên khoảnh khắc bậc một (giá trị trung bình) và khoảnh khắc bậc hai (phương sai không tập trung) của độ dốc. Nó tính toán mức trung bình giảm dần theo cấp số nhân của độ dốc trong quá khứ và giá trị bình phương của chúng, ước tính hiệu quả khoảnh khắc đầu tiên (trung bình) và khoảnh khắc thứ hai (phương sai) của độ dốc.
* Thuật toán sử dụng ước tính giá trị trung bình và phương sai này để cập nhật các tham số. Nó kết hợp các lợi ích của AdaGrad, giúp điều chỉnh tỷ lệ học tập dựa trên độ dốc lịch sử và RMSprop, điều chỉnh tốc độ học tập một cách thích ứng dựa trên mức trung bình di chuyển của độ dốc bình phương.
* Ưu điểm chính của Adam là khả năng xử lý độ dốc thưa thớt và các mục tiêu không cố định. Bằng cách duy trì tốc độ học riêng biệt cho từng tham số, nó có thể cập nhật các tham số một cách thích ứng dựa trên mức độ liên quan của chúng với quá trình tối ưu hóa. Điều này khiến Adam đặc biệt hiệu quả trong các tình huống trong đó các thông số khác nhau có thể có yêu cầu cập nhật khác nhau.
* Ngoài ra, Adam còn giúp khắc phục một số hạn chế của các thuật toán tối ưu hóa khác. Nó thể hiện các đặc tính hội tụ tốt và mạnh mẽ đối với độ dốc nhiễu hoặc thưa thớt. Việc điều chỉnh tốc độ học thích ứng cho phép Adam hội tụ nhanh chóng và đạt được hiệu suất tốt trên nhiều nhiệm vụ học máy.
* Một tính năng quan trọng khác của Adam là bước hiệu chỉnh sai lệch. Vì Adam tính toán khoảnh khắc thứ nhất và thứ hai của độ dốc bằng cách sử dụng đường trung bình động hàm mũ nên các ước tính có độ lệch về 0, đặc biệt là trong giai đoạn đầu đào tạo. Để điều chỉnh sai lệch này, Adam áp dụng hệ số hiệu chỉnh cho các ước tính đã điều chỉnh sai lệch. Điều này giúp cải thiện độ chính xác của ước tính thời điểm và nâng cao hơn nữa hiệu suất của trình tối ưu hóa.
  + 1. **RMSprop**
* Root Mean Square Propagation (RMSprop) là một thuật toán tối ưu hóa điều chỉnh tốc độ học cho từng tham số dựa trên mức trung bình di chuyển của độ dốc bình phương. Nó chia tốc độ học cho giá trị trung bình đang chạy của lịch sử các gradient bình phương để tăng tốc độ hội tụ. RMSprop đặc biệt hiệu quả trong việc xử lý dữ liệu thưa thớt hoặc các mục tiêu không cố định
* Mục tiêu chính của RMSprop là tăng tốc độ hội tụ bằng cách điều chỉnh tốc độ học dựa trên mức trung bình động của gradient bình phương. Nó chia tốc độ học cho căn bậc hai của mức trung bình giảm dần theo cấp số nhân của gradient bình phương. Việc chuẩn hóa này ngăn tốc độ học tập trở nên quá lớn khi độ dốc luôn lớn và ngược lại khi độ dốc luôn nhỏ.
* Trực giác đằng sau RMSprop là nó điều chỉnh tốc độ học khác nhau cho từng tham số dựa trên lịch sử độ dốc gần đây của chúng. Độ dốc có cường độ lớn hơn sẽ dẫn đến tốc độ học nhỏ hơn, trong khi độ dốc có cường độ nhỏ hơn sẽ dẫn đến tốc độ học lớn hơn. Sự điều chỉnh thích ứng tốc độ học này giúp cải thiện sự hội tụ và giảm bớt vấn đề dao động hoặc phân kỳ có thể xảy ra với tốc độ học cố định.
* Ngoài ra, RMSprop còn hiệu quả trong việc xử lý dữ liệu thưa thớt hoặc các mục tiêu không cố định. Bằng cách duy trì mức trung bình động của các gradient bình phương, nó điều chỉnh tốc độ học theo các đặc điểm của gradient gặp phải trong quá trình huấn luyện. Điều này làm cho RMSprop đặc biệt hữu ích trong các tình huống trong đó độ dốc thay đổi đáng kể giữa các tham số hoặc điểm dữ liệu khác nhau.
* Một tính năng quan trọng của RMSprop là khả năng xử lý các tốc độ học khác nhau cho các tham số khác nhau. Nó cho phép thuật toán điều chỉnh tốc độ học riêng cho từng tham số, điều này có thể có lợi khi một số tham số nhất định yêu cầu cập nhật chậm hơn hoặc nhanh hơn so với các tham số khác.
  + 1. **AdaGrad**
* AdaGrad, viết tắt của Adaptive gradient, là một thuật toán tối ưu hóa điều chỉnh tốc độ học tập cho từng tham số dựa trên lịch sử của độ dốc. Nó chỉ định tốc độ học khác nhau cho từng tham số, với tốc độ học giảm dần đối với các tham số xuất hiện thường xuyên. AdaGrad có hiệu quả trong việc xử lý dữ liệu thưa thớt và thường được sử dụng trong các tác vụ xử lý ngôn ngữ tự nhiên
* Ý tưởng chính đằng sau AdaGrad là chỉ định tốc độ học khác nhau cho từng tham số dựa trên độ lớn lịch sử của độ dốc. Nó bắt đầu bằng cách tích lũy tổng bình phương gradient cho mỗi tham số theo thời gian. Sự tích lũy này hoạt động như một thước đo tần suất một tham số được cập nhật, mang lại nhiều trọng số hơn cho các tham số được cập nhật không thường xuyên và ít trọng số hơn cho các tham số được cập nhật thường xuyên.
* AdaGrad đạt được sự điều chỉnh này bằng cách chia tốc độ học cho căn bậc hai của tổng bình phương gradient tích lũy. Các tham số có độ dốc lớn hơn sẽ có tốc độ học nhỏ hơn, trong khi các tham số có độ dốc nhỏ hơn sẽ có tốc độ học lớn hơn. Cách tiếp cận này cho phép AdaGrad thay đổi tốc độ học tập một cách thích ứng cho từng tham số, giải quyết vấn đề biến mất hoặc bùng nổ độ dốc.
* Một ưu điểm chính của AdaGrad là khả năng xử lý dữ liệu thưa thớt. Trong nhiều tác vụ học máy, dữ liệu đầu vào thưa thớt, nghĩa là chỉ một phần nhỏ các tính năng hoặc đầu vào có giá trị khác 0. Các thuật toán tối ưu hóa truyền thống xử lý tất cả các kích thước như nhau có thể gán các bản cập nhật quá mức cho các kích thước khác 0, dẫn đến hiệu suất kém. AdaGrad giảm thiểu vấn đề này bằng cách tăng tỷ lệ học tỷ lệ nghịch với căn bậc hai của tổng bình phương gradient tích lũy, nhấn mạnh hơn vào các tính năng hoặc đầu vào hiếm khi xảy ra.
* Tuy nhiên, một nhược điểm tiềm ẩn của AdaGrad là tốc độ học có xu hướng trở nên quá nhỏ khi quá trình tích lũy gradient bình phương tiếp tục diễn ra. Điều này có thể cản trở việc học tập trong các giai đoạn đào tạo sau này. Để giải quyết vấn đề này, các thuật toán tối ưu hóa khác như RMSprop và Adam đã được phát triển, sử dụng các chiến lược khác nhau để điều chỉnh tốc độ học.
  1. **So sánh các thuật toán tối ưu**

| **Thuật Toán** | **Ưu Điểm** | **Nhược Điểm** | **Dữ liệu phù hợp** | **Tính Chính Xác** |
| --- | --- | --- | --- | --- |
| Stochastic Gradient Descent (SGD) | - Dễ triển khai và hiểu. | - Có thể chậm hội tụ, đặc biệt trên dữ liệu lớn. | - Phù hợp với dữ liệu lớn và khi bộ nhớ hạn chế.  - Sử dụng trong các mô hình có tính chất trơn và không nhiễu nhiều. | Phổ biến, có thể hoạt động tốt nhưng cần điều chỉnh hyperparameter một cách cẩn thận. |
| Gradient Descent (GD) | - Hội tụ ổn định trên dữ liệu trơn. | - Cần nhiều bộ nhớ, chậm trên dữ liệu lớn. | - Phù hợp với dữ liệu nhỏ và khi bộ nhớ không là vấn đề. - Sử dụng khi muốn đảm bảo hội tụ chắc chắn đến điểm tối ưu. | -Hiệu suất tốt nhưng có thể chậm trên dữ liệu lớn. |
| AdaGrad | - Thích hợp với dữ liệu có đặc trưng thưa. | - Learning rate giảm quá nhanh, có thể dẫn đến hội tụ sớm. | - Phù hợp với bài toán có dữ liệu có đặc trưng thưa. | Có thể cho kết quả tốt trên dữ liệu có đặc trưng thưa, nhưng cần điều chỉnh learning rate. |
| RMSprop | - Thích ứng với biến động của gradient. | - Cần chọn tham số learning rate. | - Phù hợp với nhiều loại dữ liệu, đặc biệt là khi có biến động nhanh trong quá trình học. - Thường là một lựa chọn tốt giữa SGD và Adam. | Hiệu suất tốt, có thể tự điều chỉnh learning rate, thích ứng với biến động của gradient. |
| Adam (Adaptive Moment Estimation) | - Kết hợp lợi ích của Momentum và RMSprop. | - Có thể nhạy cảm với outliers. | - Phổ biến và thường hoạt động tốt trên nhiều loại dữ liệu. - Đặc biệt phù hợp khi không rõ thuật toán nào phải chọn. | Thường cho hiệu suất tốt, không yêu cầu nhiều điều chỉnh. |

* 1. **Xây dụng mô hình học máy và sử dụng các trình tối ưu**
* Chuẩn bị dữ liệu:
* Dataset: ***TeePublic\_review.csv*** (sẽ được giới thiệu trong phần code).
* Xác định nhãn cho mô hình phân loại. Trong trường hợp này, đó là cột review-label với giá trị từ 1 đến 5.
* Tách tập dữ liệu thành tập huấn luyện và tập kiểm thử để đánh giá hiệu suất của mô hình.
* Tiền xử lý dữ liệu:
* Loại bỏ các cột không cần thiết hoặc không ảnh hưởng đến việc phân loại (ví dụ: reviewer\_id, store\_location, latitude, longitude).
* Tiền xử lý văn bản trong cột review, bao gồm việc loại bỏ stop words, chuyển đổi về dạng lowercase, và có thể sử dụng các kỹ thuật như TF-IDF hoặc Word Embeddings.
* Xây dựng mô hình:
* Chọn một mô hình học máy phù hợp cho bài toán phân loại, chẳng hạn như Support Vector Machines (SVM), Naive Bayes, hoặc mô hình Deep Learning như mạng nơ-ron hồi quy.
* Huấn luyện mô hình trên tập dữ liệu huấn luyện.
* Đánh giá mô hình:
* Sử dụng tập kiểm thử để đánh giá hiệu suất của mô hình. Các metric như accuracy, precision, recall, và F1-score có thể được sử dụng để đo lường hiệu suất.





1. **Continual Learning** 
   1. **Continual Learning**

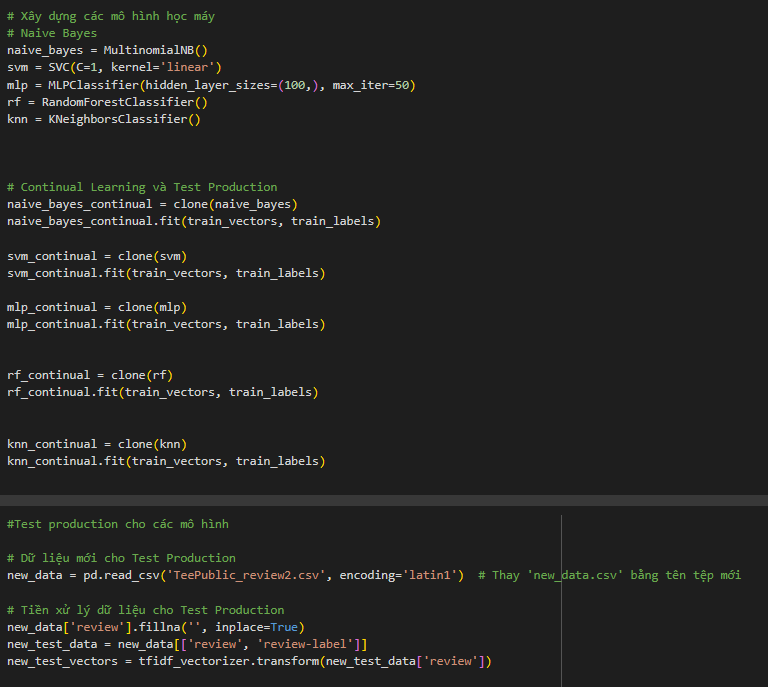
* Continual Learning (học liên tục) là một mô hình hiện đại trong lĩnh vực học máy, có thể tạo ra các mô hình có khả năng phát triển và biến đổi không ngừng.
* Không giống như những mô hình học máy thông thường thu thập kiến ​​thức về các chiến lược có xu hướng có sự hiểu biết cố định, việc học tập liên tục cho phép các mô hình phù hợp với thời gian, thu thập số liệu thống kê và năng lực mới mà không xóa bỏ những gì đã học được trong quá khứ của chúng
* Vấn đề quan trọng được giải quyết bằng cách học tập thường xuyên là sự “quên lãng”, trong đó các mô hình truyền thống thường có xu hướng mất đi sự nhận biết trong các nhiệm vụ đã học trước đó khi tiếp xúc với các nhiệm vụ mới. Bằng cách giảm thiểu việc “quên lãng” này, liên tục học sẽ giúp các hệ thống AI luôn có khả năng áp dụng và phát triển
  1. **Các yếu tố chính trong quá trình học liên tục**
* Học tập tăng dần(Incremental Learning): Việc tiếp thu kiến ​​thức liên tục bao gồm việc đào tạo một mô hình về các sự kiện mới qua nhiều năm, thường là theo cách tăng dần. Điều này có nghĩa là phiên bản cần phải thích ứng với số liệu thống kê mới mà không cần đào tạo lại toàn bộ tập dữ liệu.
* Bộ nhớ và sự quên(Memory and Forgetting): Các mô hình trong quá trình nghiên cứu liên tục muốn có các cơ chế không quên và lưu giữ kiến ​​thức quan trọng từ những lần xem xét trước đây, cũng như các kỹ thuật để tránh sự quên lãng nghiêm trọng, trong đó họ mất hiệu suất tổng thể đối với các nghĩa vụ đã phát hiện trước đó trong khi tiếp thu kiến ​​thức về những nghĩa vụ mới.
* Trình tự nhiệm vụ(Task Sequences): Việc tiếp thu kiến ​​thức liên tục về các tình huống có thể khác nhau tùy theo các cụm từ trong chuỗi nhiệm vụ mà bạn gặp phải. Một số cũng có thể liên quan đến một trật tự nghĩa vụ cố định, ngay cả khi một số khác cũng có thể có một trật tự năng động hơn hoặc không thể đoán trước.
* Chính quy hóa và ổn định(Regularization and Stabilization): Các kỹ thuật như hợp nhất trọng số đàn hồi (EWC) và synaptic intelligence (SI) được sử dụng để chính quy hóa và ổn định trọng số mô hình nhằm giúp bạn tiết kiệm được những sửa đổi mạnh mẽ trong khi tìm hiểu các nghĩa vụ mới, hỗ trợ duy trì sự hiểu biết trước đó.
* Phát lại và Trải nghiệm Phát lại(Replay and Experience Repla): Cơ chế phát lại liên quan đến việc xem lại và đào tạo lại định kỳ về các hồ sơ hoặc câu chuyện ngoài hồ sơ để nâng cao và củng cố sự hiểu biết có được trong quá trình thực hiện các nhiệm vụ trước đó.
* Chuyển giao học tập(Transfer Learnin): Tận dụng kiến ​​thức chuyên môn từ các nhiệm vụ trước đây để hỗ trợ việc tiếp thu kiến ​​thức về các trách nhiệm mới là vấn đề cơ bản của việc nắm vững liên tục.
  1. **Một số phương pháp học liên tục (Continual learning)**
* Task-based Continual Learning (học liên tục dựa trên nhiệm vụ): Trong phương pháp này, một phiên bản học một chuỗi các nghĩa vụ riêng biệt qua nhiều năm. Mục tiêu của mô hình là phù hợp với từng cam kết mới trong khi vẫn lưu giữ kiến ​​thức về các nghĩa vụ đã được xác định trước đó. Các kỹ thuật bao gồm Hợp nhất trọng lượng đàn hồi (EWC) và Mạng thần kinh tiến bộ (PNN) thuộc loại này.
* Class-incremental Learning (học tăng dần theo lớp): Làm chủ theo lớp tăng dần chuyên quản lý các lớp học mới hoặc các lớp thông tin qua nhiều năm trong khi vẫn duy trì sự hiểu biết về các bài học đã thấy trước đây. Điều này phổ biến trong các gói như nhận dạng hình ảnh, trong đó việc đào tạo đối tượng mới được thực hiện định kỳ. Các phương pháp như iCaRL (Học phân loại tăng dần và biểu diễn) được sử dụng để thành thạo theo từng lớp.
* Domain-incremental Learning(học tăng dần theo miền): Tích lũy kiến ​​thức theo miền tăng dần về các vấn đề liên quan đến việc thích ứng với việc phân phối bản ghi mới hoặc tên miền. Ví dụ, trong chế tạo robot tự cung tự cấp, robot có thể muốn thích nghi với các môi trường khác nhau. Các kỹ thuật thay đổi diện tích và học tăng dần theo diện tích được sử dụng để giải quyết tình trạng này.
  1. **Lợi ích của Continual learning**
* Khả năng thích ứng: Cho phép các mô hình thích ứng và phát triển theo thời gian để làm cho chúng phù hợp với các ứng dụng trong môi trường năng động và thay đổi. Khả năng thích ứng này rất quan trọng trong các lĩnh vực như robot tự động và hiểu ngôn ngữ tự nhiên.
* Hiệu quả: Thay vì đào tạo lại các mô hình từ đầu, mỗi khi dữ liệu hoặc nhiệm vụ mới xuất hiện, nó cho phép cập nhật gia tăng giúp tiết kiệm tài nguyên và thời gian tính toán.
* Ghi nhớ kiến ​​thức: Nó giảm thiểu vấn đề quên lãng nghiêm trọng, cho phép các mô hình lưu giữ kiến ​​thức về các nhiệm vụ hoặc kinh nghiệm trong quá khứ. Điều này có giá trị khi xử lý vấn đề lưu giữ bộ nhớ dài hạn trong hệ thống AI.
* Giảm lưu trữ dữ liệu: Các kỹ thuật như phát lại làm giảm nhu cầu lưu trữ và quản lý các tập dữ liệu lịch sử lớn, giúp việc triển khai học tập liên tục trong các cài đặt có nguồn lực hạn chế trở nên khả thi hơn.
* Tính linh hoạt: Nó được áp dụng cho nhiều lĩnh vực bao gồm xử lý ngôn ngữ tự nhiên, thị giác máy tính, hệ thống đề xuất khiến nó trở thành một cách tiếp cận linh hoạt trong AI.
  1. **Những hạn chế**
* Sự quên lãng: Mặc dù đã cố gắng giảm thiểu, các mô hình học tập liên tục vẫn có thể gặp phải tình trạng quên lãng, dẫn đến mất dần hiệu suất đối với các nhiệm vụ trước đây khi học những nhiệm vụ mới.
* Điều chỉnh quá mức cho dữ liệu cũ: Một số phương pháp học tập liên tục có thể quá phù hợp với dữ liệu cũ, điều này có thể khiến mô hình khó khái quát hóa hơn cho các nhiệm vụ hoặc lĩnh vực mới.
* Độ phức tạp: Việc triển khai các kỹ thuật học tập liên tục có thể phức tạp và đòi hỏi phải điều chỉnh và thiết kế cẩn thận. Sự phức tạp này có thể hạn chế việc áp dụng chúng trong một số ứng dụng.
* Khả năng mở rộng: Khi mô hình tích lũy nhiều kiến ​​thức hơn, khả năng mở rộng có thể trở thành một thách thức. Kích thước của mô hình và yêu cầu tính toán có thể tăng đáng kể theo thời gian.
* Thay đổi phân phối dữ liệu: Khi các nhiệm vụ hoặc miền mới có cách phân phối dữ liệu khác biệt đáng kể so với trước đây, các mô hình học tập liên tục có thể gặp khó khăn trong việc thích ứng một cách hiệu quả.
* Cân bằng giữa kiến ​​thức cũ và kiến ​​thức mới: Tạo ra sự cân bằng phù hợp giữa kiến ​​thức cũ và kiến ​​thức mới có thể là một thách thức. Người mẫu cần quyết định điều gì

1. **Test Production**
   1. **Khái niệm**

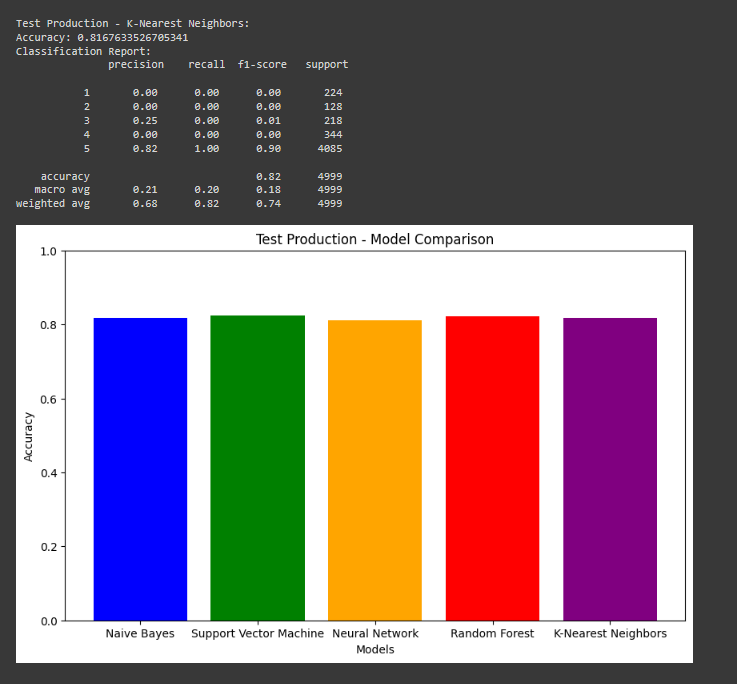
* Kiểm thử mô hình máy học trong môi trường sản xuất là quá trình đảm bảo rằng mô hình máy học, sau khi đã được đào tạo và kiểm thử trong môi trường phát triển, có thể hoạt động hiệu quả và ổn định khi triển khai vào môi trường thực tế. Quá trình này đặt ra các thách thức và yêu cầu cụ thể để đảm bảo mô hình không chỉ đáp ứng các tiêu chí hiệu suất mà còn duy trì được độ tin cậy trong điều kiện thực tế.
  1. **Các lợi ích**
* Thay Đổi Dữ Liệu: liên tục theo dõi thay đổi dữ liệu để đảm bảo phân phối của dữ liệu đầu vào không thay đổi đáng kể theo thời gian. Xác định và xác nhận tầm quan trọng của các đặc trưng, đặc biệt là khi có đặc trưng mới được giới thiệu.
* Xác định hiệu suất: xác định và theo dõi các chỉ số hiệu suất phù hợp cho trường hợp sử dụng cụ thể của bạn thường xuyên đánh giá độ chính xác, precision, recall, F1 score, hoặc bất kỳ chỉ số nào khác có liên quan.
* Thử Nghiệm A/B: 7hực hiện thử nghiệm A/B để so sánh hiệu suất của mô hình mới với mô hình hiện tại hoặc các biến thể khác. Triển khai mô hình mới dần dần cho một tập con nhỏ người dùng để giảm thiểu các rủi ro có thể phát sinh.
* Sự Mạnh Mẽ và Bảo Mật: kiểm thử độ mạnh mẽ của mô hình đối với các cuộc tấn công từ phía người tấn công và sự thay đổi đột ngột của đầu vào.Đánh giá các biện pháp bảo mật để bảo vệ mô hình và dữ liệu khỏi những rủi ro có thể phát sinh.
* Khả Năng Mở Rộng và Latency: đánh giá khả năng mở rộng của mô hình để xử lý công việc tăng cường. Giám sát và tối ưu hóa độ trễ để đảm bảo dự đoán kịp thời.
* Khả Năng Giải Thích Mô Hình: triển khai các kỹ thuật giải thích mô hình để hiểu và giải thích dự đoán, đặc biệt là trong các ứng dụng quan trọng.
* Theo Dõi Liên Tục: thiết lập hệ thống theo dõi liên tục để theo dõi hiệu suất của mô hình theo thời gian. Đặt cảnh báo cho những sự chệch đáng kể từ hành vi dự kiến.

1. **Xây dụng mô hình học máy sử dụng Continual Learning và Test Production**

* Dữ liệu: sử dụng dataset 'TeePublic\_review1.csv' với các thuộc tính như reviewer\_id, store\_location, latitude, longitude, date, month, year, title, review, và review-label.
* Chuẩn Bị Dữ Liệu: tiền xử lý dữ liệu bằng cách điền giá trị trống trong cột 'review' và chọn chỉ các cột quan trọng: 'review' và 'review-label'. Chuyển đổi văn bản thành vector TF-IDF sử dụng TfidfVectorizer.
* Các mô hình học máy: xây dựng các mô hình học máy như Naive Bayes, SVM, Neural Network, Random Forest, và K-Nearest Neighbors.
* Continual Learning và Test Production: thực hiện Continual Learning cho mỗi mô hình để giữ thông tin đã học trước đó.
* Sử dụng dữ liệu mới từ 'TeePublic\_review2.csv' cho Test Production và đánh giá hiệu suất của mỗi mô hình.
* Hiệu Suất và vẽ biểu đồ so sánh: in kết quả hiệu suất, bao gồm độ chính xác và báo cáo phân loại, cho mỗi mô hình trên dữ liệu mới.Vẽ biểu đồ so sánh hiệu suất của các mô hình trên dữ liệu mới
* Mã code:



* Kết quả:

****

# PHẦN 2 – CÂU 2 : XÂY DỰNG MÔ HÌNH HỌC MÁY – DỰ ĐOÁN PHÊ DUYỆT THẺ TÍN DỤNG TỰ ĐỘNG

1. **Giới thiệu dữ liệu sử dụng cho bài toán**

**1.1 Dữ liệu: Credit Card Approval Prediction**

* **Lý do thu thập dữ liệu :**

Thẻ điểm tín dụng là một phương pháp kiểm soát rủi ro phổ biến trong ngành tài chính. Nó sử dụng thông tin và dữ liệu cá nhân do người đăng ký thẻ tín dụng gửi để dự đoán khả năng vỡ nợ và vay thẻ tín dụng trong tương lai. Ngân hàng có thể quyết định có cấp thẻ tín dụng cho người nộp đơn hay không. Điểm tín dụng có thể định lượng một cách khách quan mức độ rủi ro.  
   
Nói chung, thẻ điểm tín dụng dựa trên dữ liệu lịch sử. Một khi gặp phải những biến động lớn về kinh tế. Các mô hình trong quá khứ có thể mất đi khả năng dự đoán ban đầu. Mô hình logistic là một phương pháp phổ biến để chấm điểm tín dụng. Bởi vì Logistic phù hợp với các nhiệm vụ phân loại nhị phân và có thể tính toán các hệ số của từng đặc điểm. Để thuận tiện cho việc hiểu và vận hành, thẻ điểm sẽ nhân hệ số hồi quy logistic với một giá trị nhất định (chẳng hạn như 100) và làm tròn số đó.  
   
Hiện nay với sự phát triển của các thuật toán học máy. Các phương pháp dự đoán khác như Tăng cường, Rừng ngẫu nhiên và Máy vectơ hỗ trợ đã được đưa vào tính điểm thẻ tín dụng. Tuy nhiên, những phương pháp này thường không có tính minh bạch tốt. Có thể khó đưa ra lý do từ chối hoặc chấp nhận cho khách hàng và cơ quan quản lý.

* **Mô tả**

Tập dữ liệu "Credit Card Approval Prediction" là một tập dữ liệu phổ biến trong lĩnh vực học máy và dự đoán tín dụng. Tập dữ liệu này thường được sử dụng để dự đoán xem một đơn vay mượn thẻ tín dụng sẽ được phê duyệt hay từ chối dựa trên các đặc trưng khách hàng và lịch sử tài chính.

Thông thường, tập dữ liệu này chứa các thông tin như:

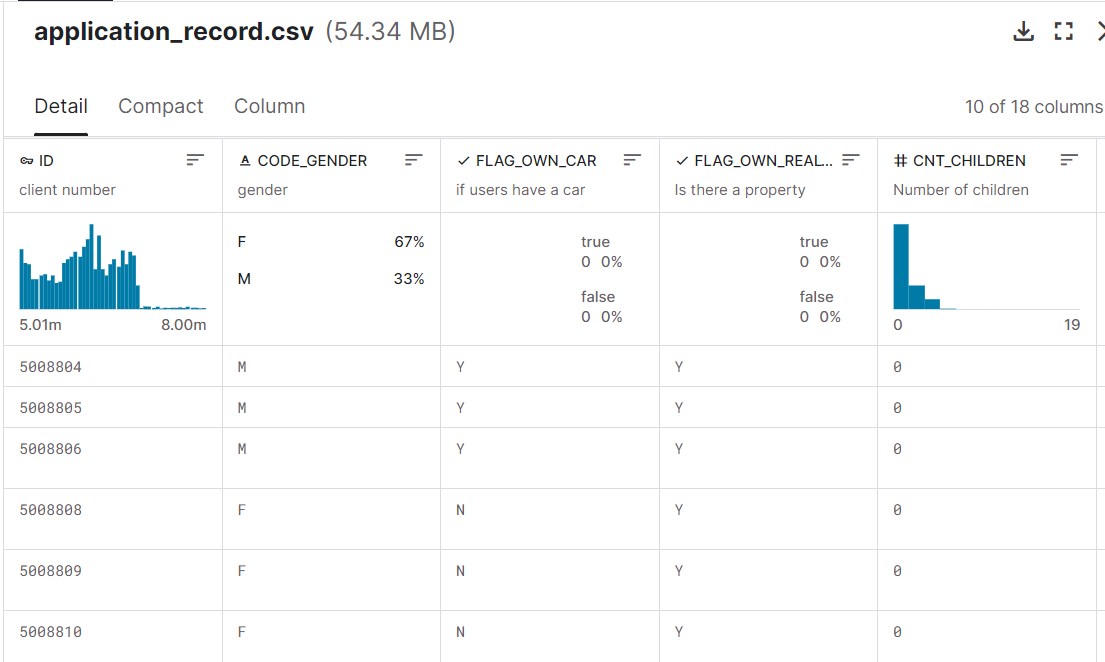
* Đặc trưng của khách hàng: Như tuổi, giới tính, thu nhập, nghề nghiệp, địa chỉ, học vấn, số lượng người phụ thuộc, v.v.
* Thông tin tài chính: Như số lượng thẻ tín dụng hiện có, lịch sử thanh toán, tỷ lệ nợ, điểm tín dụng, số tiền vay muốn, mục đích vay, v.v.
* Nhãn: Ghi chú về việc đơn vay mượn được phê duyệt (1) hoặc từ chối (0)
* **Nội dung và giải thích:**

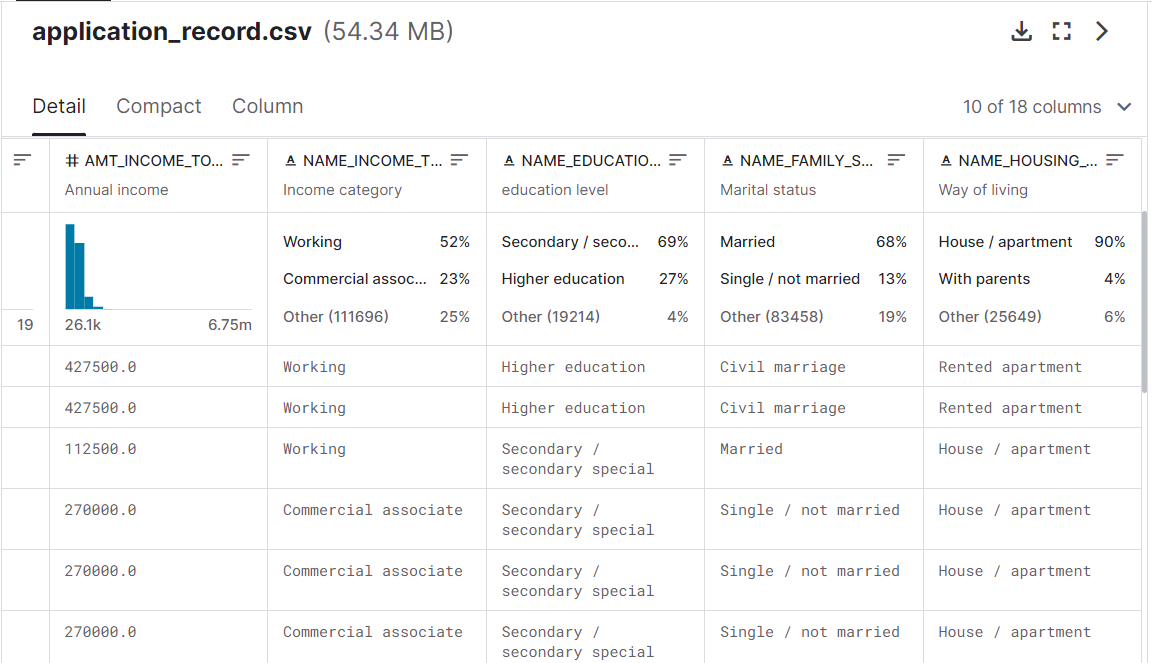
Có hai bảng được kết nối bằng ID

* application\_record.csv chứa thông tin cá nhân của người đăng ký mà bạn có thể sử dụng làm tính năng dự đoán.
* credit\_record.csv ghi lại hành vi sử dụng thẻ tín dụng của người dùng.

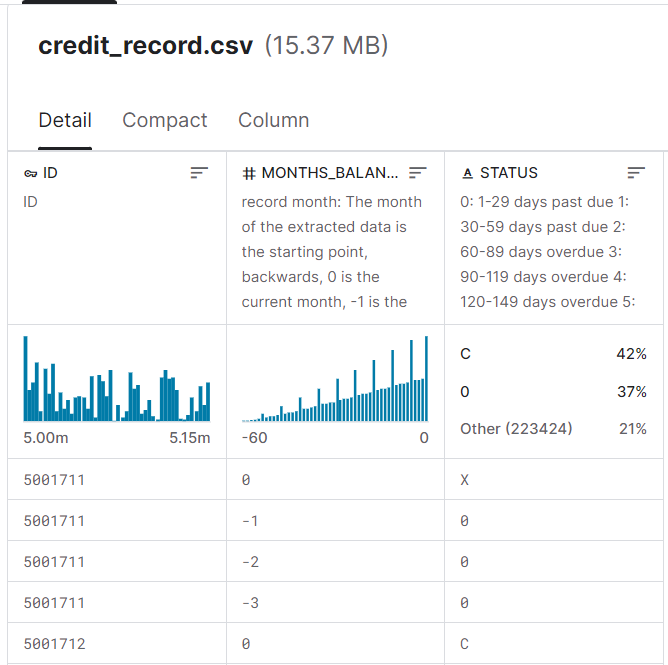
|  |  |  |
| --- | --- | --- |
| application\_record.csv | | |
| Tên tính năng | Giải trình | Bình luận |
| ID | Số lượng khách hàng |  |
| CODE\_GENDER | Giới tính |  |
| FLAG\_OWN\_CAR | Có ô tô không |  |
| FLAG\_OWN\_REALTY | Có tài sản nào không |  |
| CNT\_CHILDREN | số lượng trẻ em |  |
| AMT\_INCOME\_TOTAL | Thu nhập hàng năm |  |
| NAME\_INCOME\_TYPE | Loại thu nhập |  |
| NAME\_EDUCATION\_TYPE | Trình độ học vấn |  |
| NAME\_FAMILY\_STATUS | Trình trạng hôn nhân |  |
| NAME\_HOUSING\_TYPE | Cách sống |  |
| DAYS\_BIRTH | Sinh nhật | Đếm ngược từ ngày hiện tại (0), -1 nghĩa là ngày hôm qua |
| DAYS\_EMPLOYED | Ngày bắt đầu làm việc | Đếm ngược từ ngày hiện tại (0). Nếu tích cực, nó có nghĩa là người hiện đang thất nghiệp. |
| FLAG\_MOBIL | Có điện thoại di động không |  |
| FLAG\_WORK\_PHONE | Có điện thoại cơ quan không |  |
| FLAG\_PHONE | Có điện thoại không |  |
| FLAG\_EMAIL | Có email không |  |
| OCCUPATION\_TYPE | Nghề nghiệp |  |
| CNT\_FAM\_MEMBERS | Quy mô gia đình |  |

|  |  |  |
| --- | --- | --- |
| credit\_record.csv | | |
| Tên tính năng | Giải trình | Bình luận |
| ID | Số lượng khách hàng |  |
| MONTHS\_BALANCE | Ghi tháng | Tháng của dữ liệu được trích xuất là điểm bắt đầu, ngược lại, 0 là tháng hiện tại, -1 là tháng trước đó, v.v. |
| TATUS | Trạng thái | 0: Quá hạn 1-29 ngày  1: Quá hạn 30-59 ngày  2: Quá hạn 60-89 ngày  3: Quá hạn 90-119 ngày  4: Quá hạn 120-149 ngày 5: Nợ quá hạn hoặc nợ xấu, xóa nợ trên 150 ngày C: trả hết tháng đó X: Không vay trong tháng |



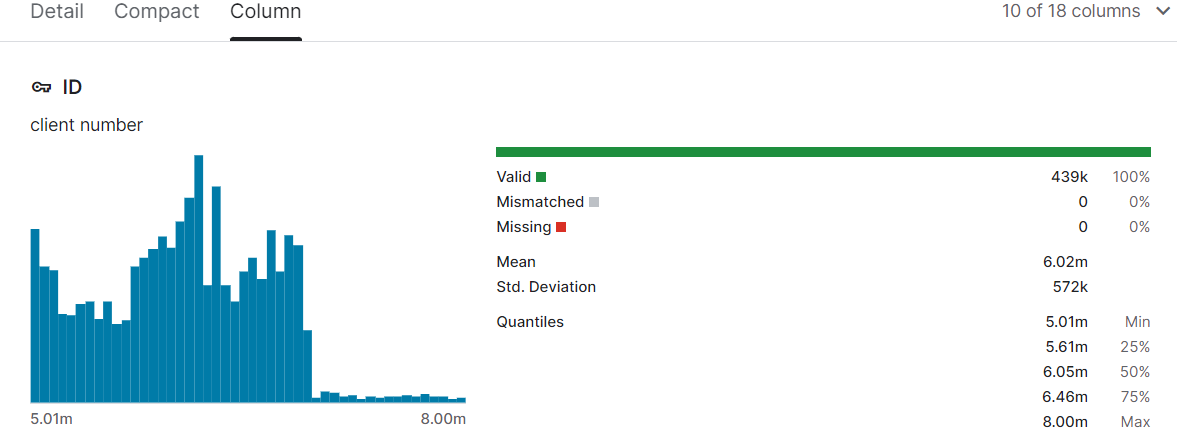


Hình Dữ liệu chung của tệp application\_record.csv

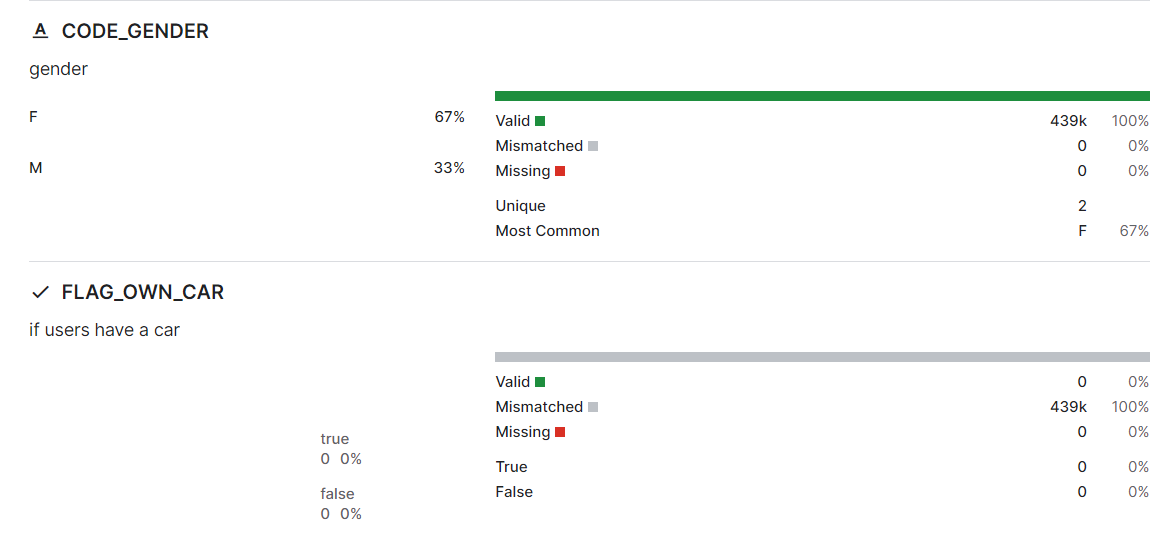


Hình 2 Dữ liệu chung của tệp credit\_record.csv

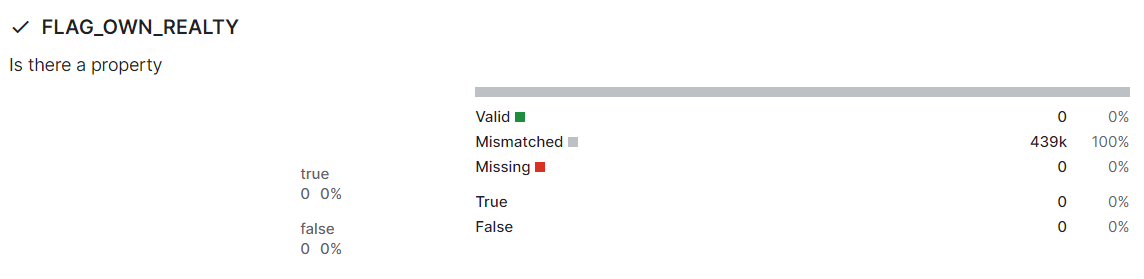
Các dữ liệu thống kê của application\_record được thể hiện trong các hình sau:



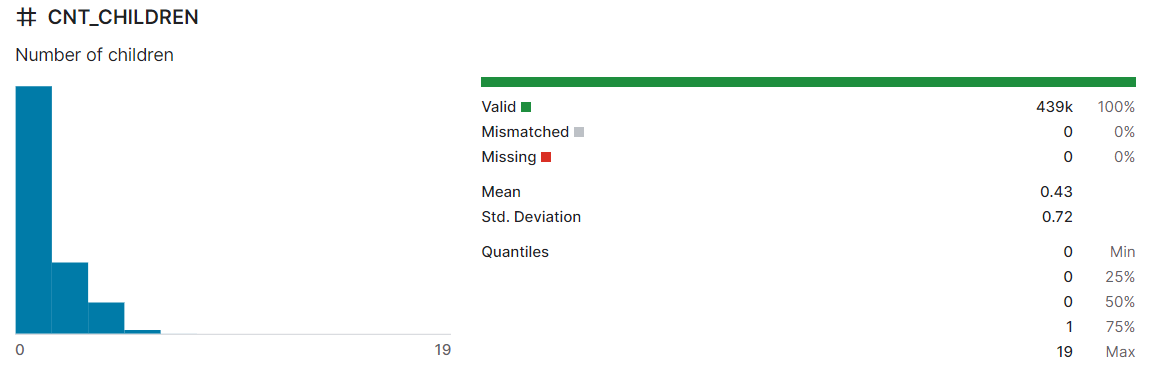
Hình Thống kê số lượng khách hàng



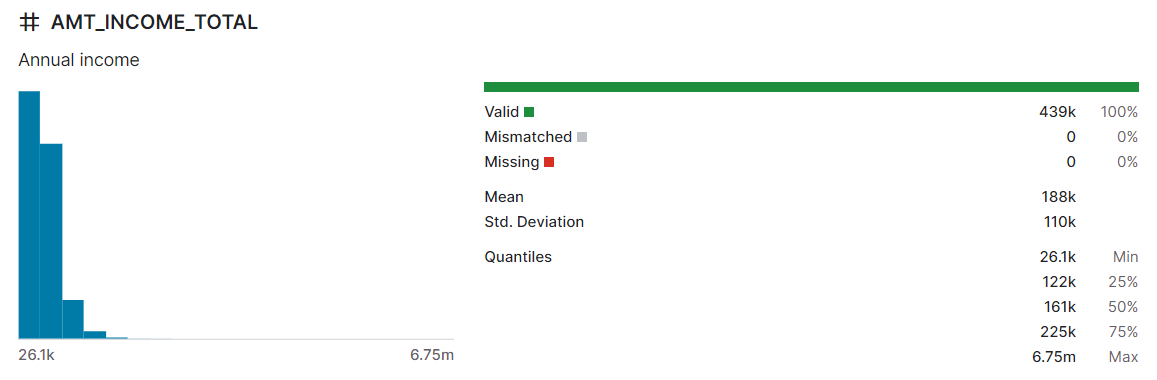
Hình Số liệu giới tính và sở hữu ô tô



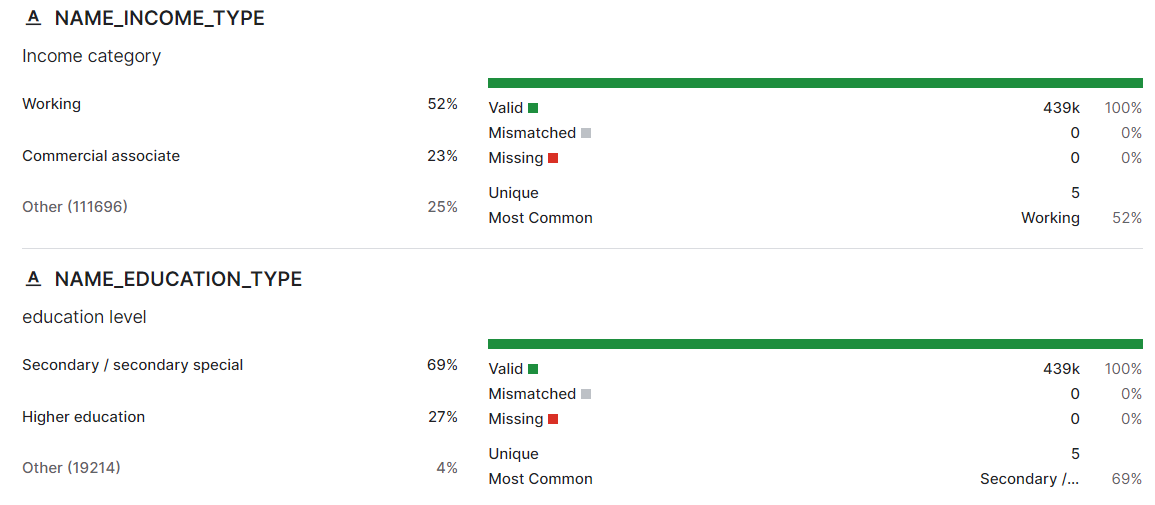
Hình Số liệu sở hữu tài sản



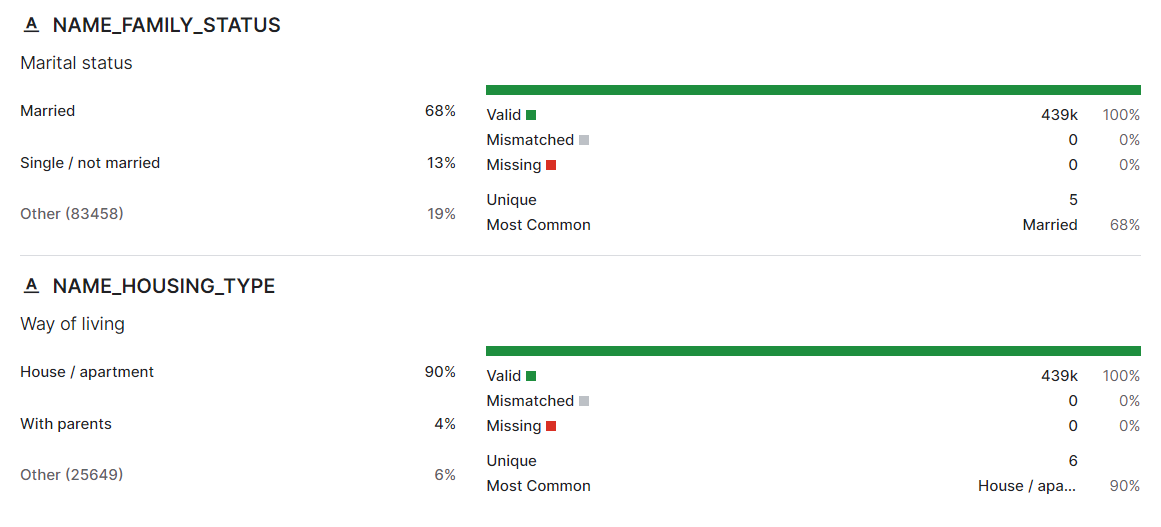
Hình Thống kê số lượng trẻ em



Hình Thống kê thu nhập hàng năm

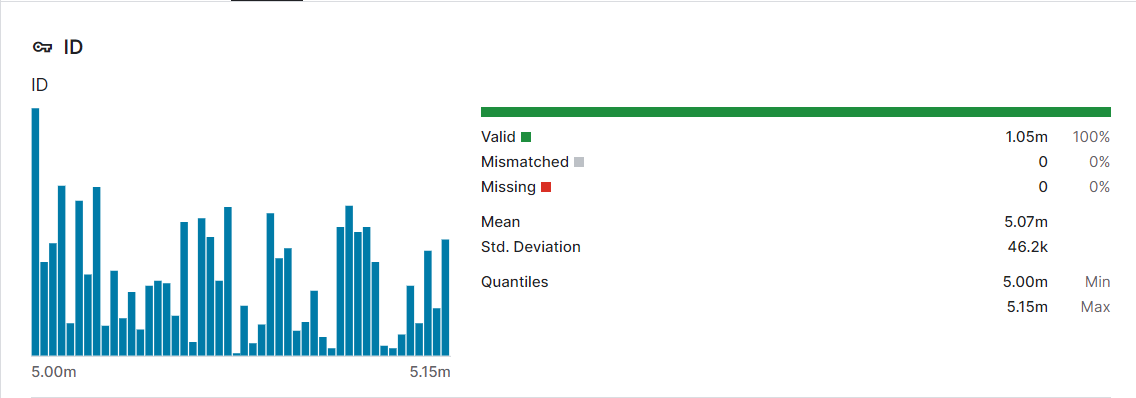


Hình Thống kê loại thu nhập

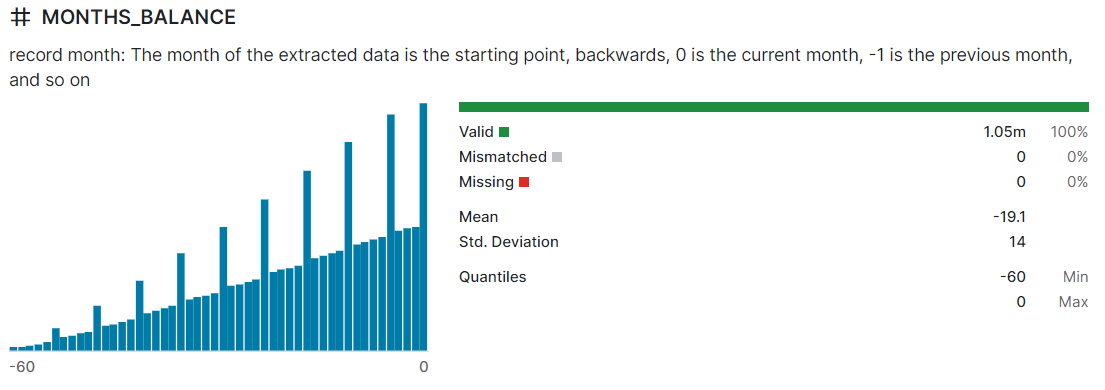


Hình Thống kê trình độ học vấn

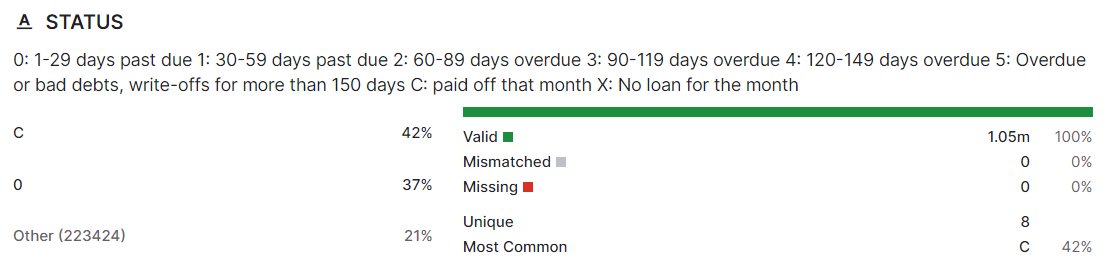
Các dữ liệu thống kê của credit\_record được thể hiện qua các biểu đồ dưới đây:



Hình Thống kê số lượng khách hàng



Hình Thống kê theo tháng



Hình Thống kê theo trạng thái ghi nợ

* 1. **Đánh giá các feature của dữ liệu**

**Dữ liệu: Credit Card Approval Prediction**

Để đánh giá vai trò của các cột trong dữ liệu đối với bài toán Dự đoán Phê duyệt thẻ tín dụng tự động, bạn có thể tập trung vào các cột có thể ảnh hưởng đến quyết định phê duyệt của hệ thống. Dưới đây là một số cột quan trọng mà bạn có thể xem xét:

• AMT\_INCOME\_TOTAL (Thu nhập hằng năm):Số tiền thu nhập hàng năm của khách hàng có thể là một yếu tố quan trọng để đánh giá khả năng thanh toán và đáng giá để cân nhắc trong quyết định phê duyệt.

• NAME\_EDUCATION\_TYPE (Loại hình giáo dục):Trình độ giáo dục của khách hàng có thể ảnh hưởng đến khả năng thanh toán và sự ổn định tài chính, có thể là một yếu tố quan trọng.

• NAME\_FAMILY\_STATUS (Tình trạng hôn nhân):Tình trạng hôn nhân của khách hàng có thể ảnh hưởng đến ổn định gia đình và khả năng chi trả nghĩa vụ tài chính.

• DAYS\_BIRTH (Tuổi):Độ tuổi của khách hàng có thể đóng vai trò trong đánh giá tính ổn định tài chính và khả năng thanh toán.

• DAYS\_EMPLOYED (Thời gian làm việc):

• Thời gian làm việc của khách hàng có thể là một yếu tố quan trọng, ảnh hưởng đến khả năng chi trả và ổn định tài chính.

• FLAG\_OWN\_CAR (Có xe ô tô):Việc sở hữu ô tô có thể là một yếu tố ảnh hưởng đến khả năng chi trả và tình trạng tài chính tổng thể.

• FLAG\_OWN\_REALTY (Có bất động sản):Việc sở hữu bất động sản cũng có thể là một chỉ số cho sự ổn định tài chính và đáng giá để xem xét.

==> Từ đó chúng ta có thể xác định các feature quan trọng của dữ liệu để xây dựng mô hình học máy giải quyết bài toán.

## 2. Xây dựng mô hình học máy

**2.1 Các mô hình sử dụng**

* Decision Tree (Cây Quyết định):
  + 1. Ưu Điểm:Dễ hiểu và trực quan: Cây quyết định có thể được biểu diễn dễ hiểu, giúp giải thích quyết định của mô hình một cách rõ ràng.
    2. Khả năng xử lý dữ liệu phi cấp: Cây quyết định có thể xử lý dữ liệu không cần phải chia thành các phần tử cấp.
    3. Ứng Dụng:Phù hợp cho việc tìm hiểu cấu trúc quyết định và giảm độ phức tạp của mô hình.
* K-Nearest Neighbors (KNN):
  + 1. Ưu Điểm:Đơn giản và dễ triển khai: KNN không đòi hỏi giả định về phân phối dữ liệu và là một mô hình dễ triển khai.
    2. Ứng Dụng:Hiệu quả cho các bài toán phân loại đơn giản và khi dữ liệu có cấu trúc lân cận.
* Gradient Boosting:
  + 1. Ưu Điểm: Hiệu suất cao: Gradient Boosting thường cho kết quả rất tốt và có khả năng xử lý các tập dữ liệu lớn.
    2. Ứng Dụng: Phù hợp cho các bài toán phức tạp, đặc biệt là khi có sự tương tác phức tạp giữa các đặc trưng.
* Random Forest:
  + 1. Ưu Điểm: Ổn định và chống overfitting: Random Forest có khả năng giảm nguy cơ overfitting do sự đa dạng của các cây quyết định.
    2. Ứng Dụng: Hiệu quả trong việc xử lý các tập dữ liệu lớn và nhiễu.
* Logistic Regression:
  + 1. Ưu Điểm: Dễ hiểu và dễ triển khai: Logistic Regression là mô hình đơn giản nhưng mạnh mẽ, thường được sử dụng như một điểm xuất phát cho các mô hình phức tạp hơn.
    2. Ứng Dụng: Phù hợp cho các bài toán phân loại nhị phân và thường được sử dụng trong các hệ thống đánh giá rủi ro tín dụng.
* Feedforward Neural Network (FNN):

Ưu Điểm khi Áp Dụng:

* + 1. Khả Năng Học Phi Tuyến: FNN có khả năng học được các biểu diễn phi tuyến tính và phi tuyến tính, giúp nắm bắt được các mối quan hệ phức tạp trong dữ liệu.
    2. Phù Hợp Cho Dữ Liệu Phi Tuyến: Nếu dữ liệu của bạn có các mối quan hệ phi tuyến, FNN có thể là một lựa chọn tốt.
    3. Ứng Dụng: Phù hợp cho các bài toán phức tạp và dữ liệu có tính phi tuyến.
  1. Recurrent Neural Network (RNN):

Ưu Điểm khi Áp Dụng:

* + 1. Xử Lý Dữ Liệu Chuỗi Thời Gian: RNN được thiết kế để xử lý dữ liệu chuỗi thời gian, có thể hữu ích nếu dữ liệu của bạn có yếu tố thời gian.
    2. Chia Sẻ Trọng Số: RNN có khả năng chia sẻ trọng số qua các bước thời gian, giúp nắm bắt thông tin từ quá khứ.
    3. Ứng Dụng: Phù hợp cho các bài toán liên quan đến chuỗi thời gian, ví dụ như dự đoán sự phê duyệt thẻ tín dụng dựa trên dữ liệu lịch sử.

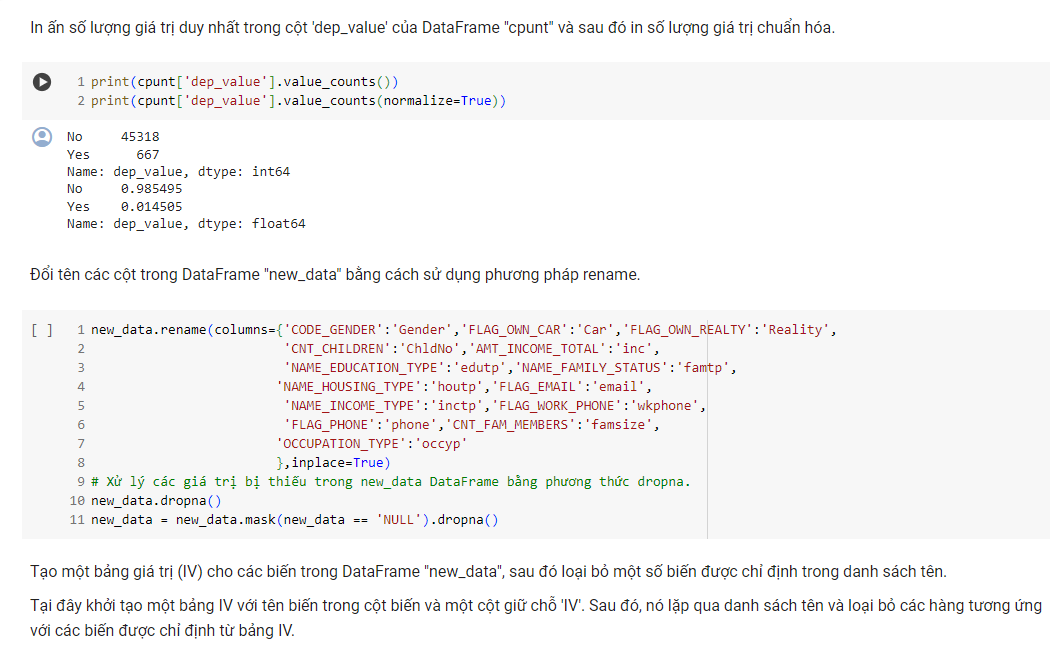
**2.2 Phòng tránh Overfitting**

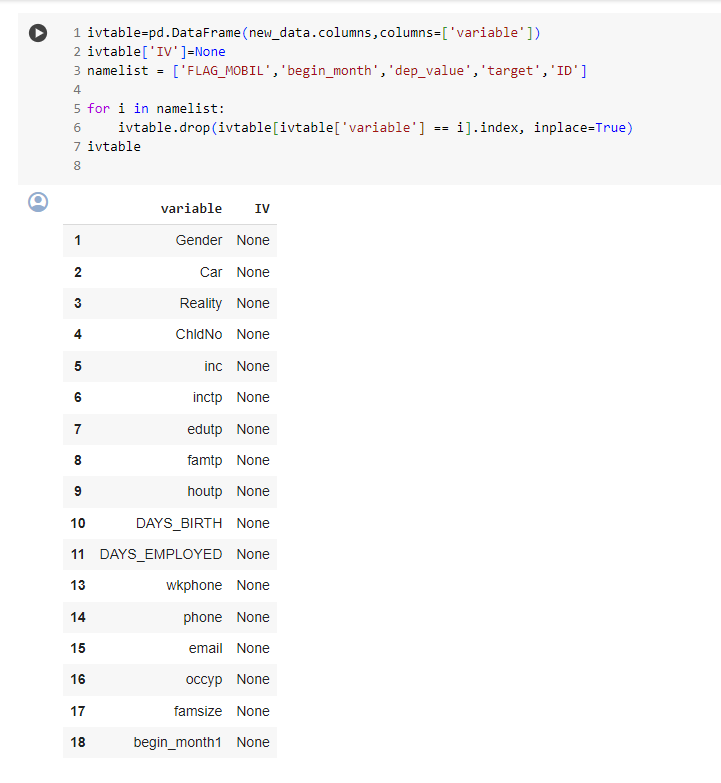
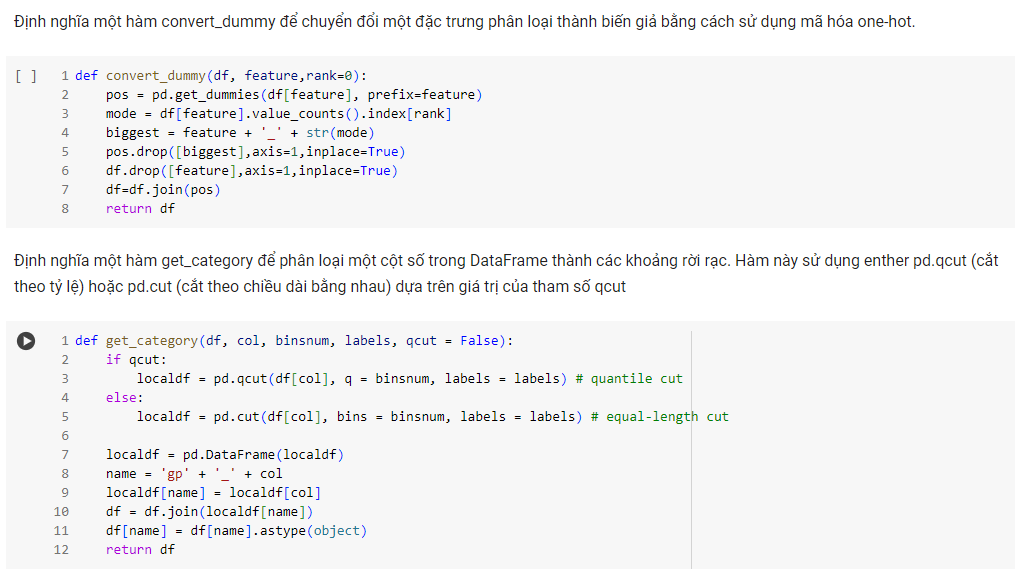
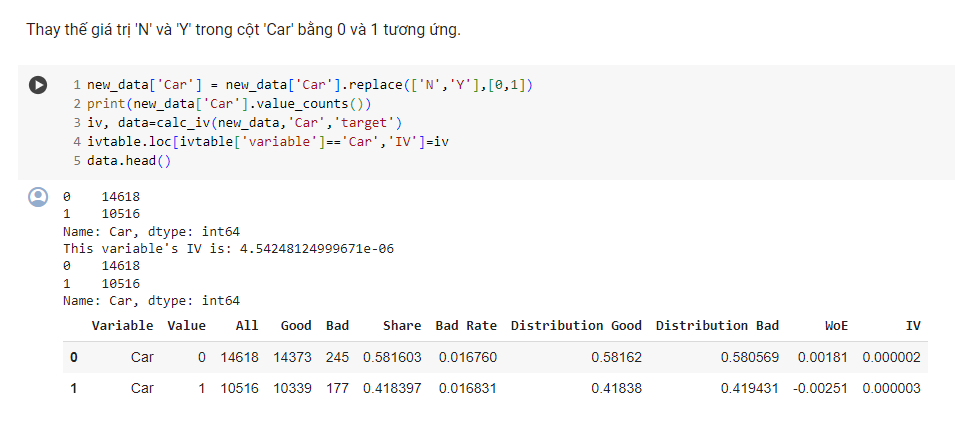
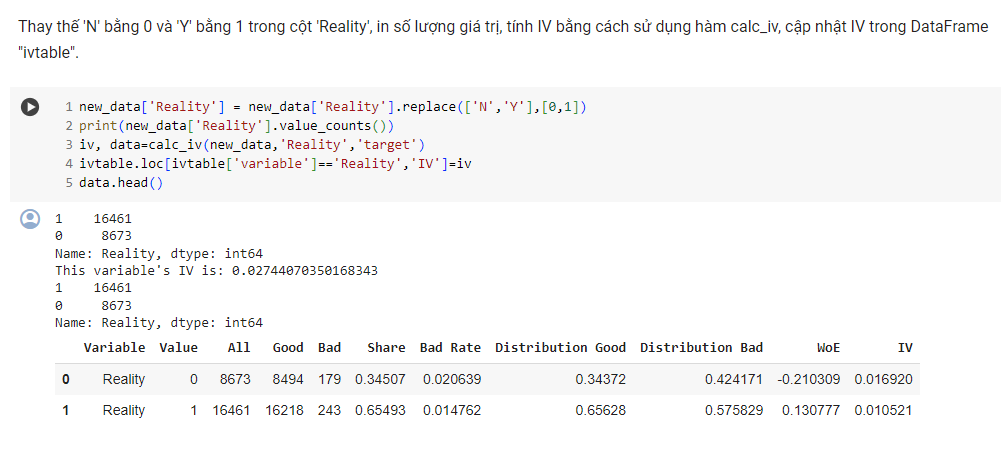
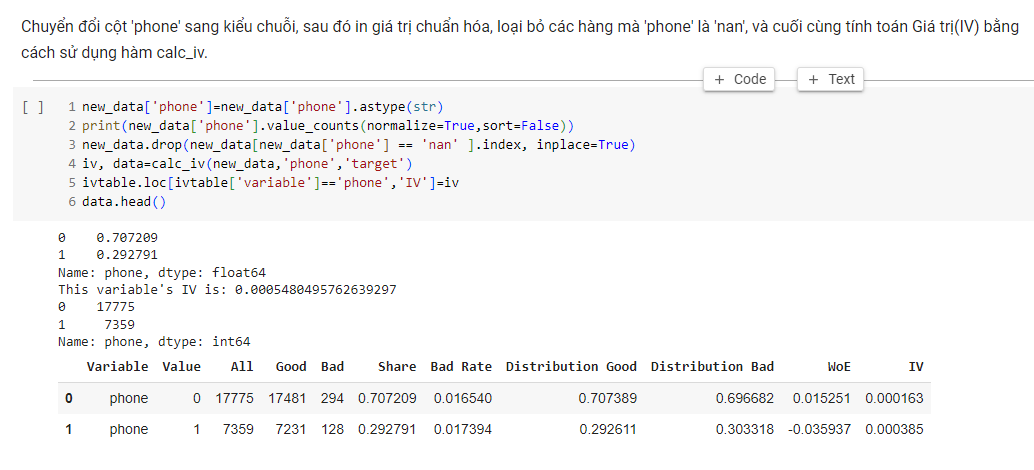
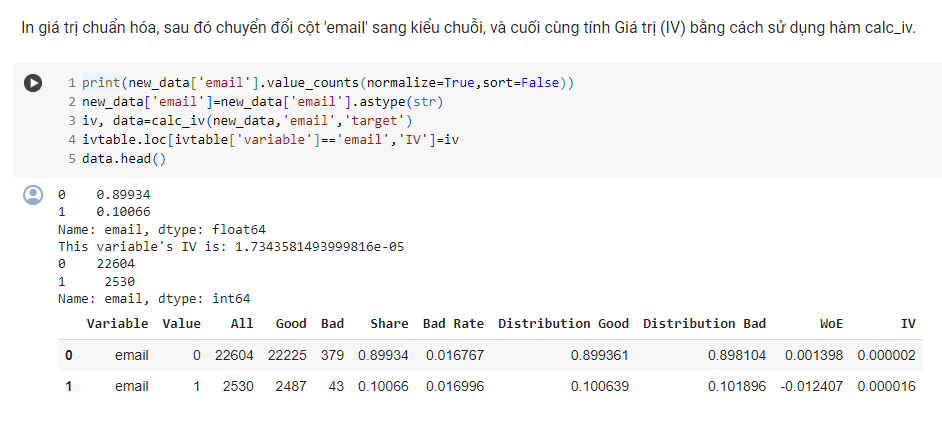
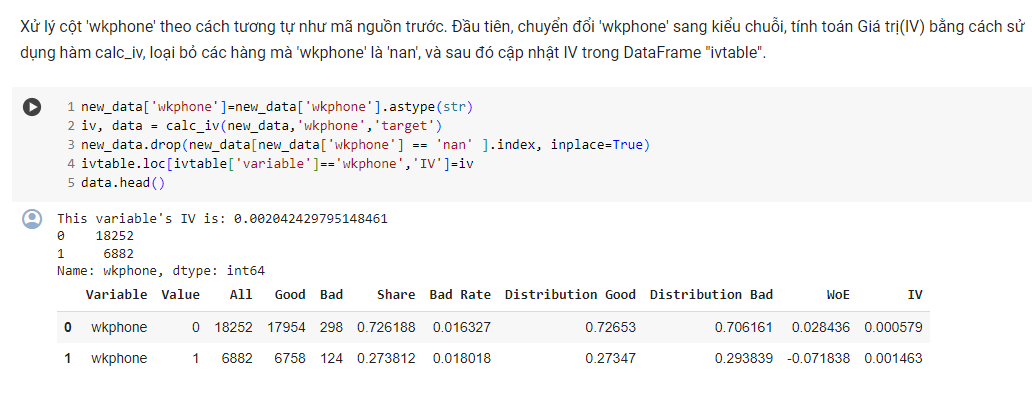
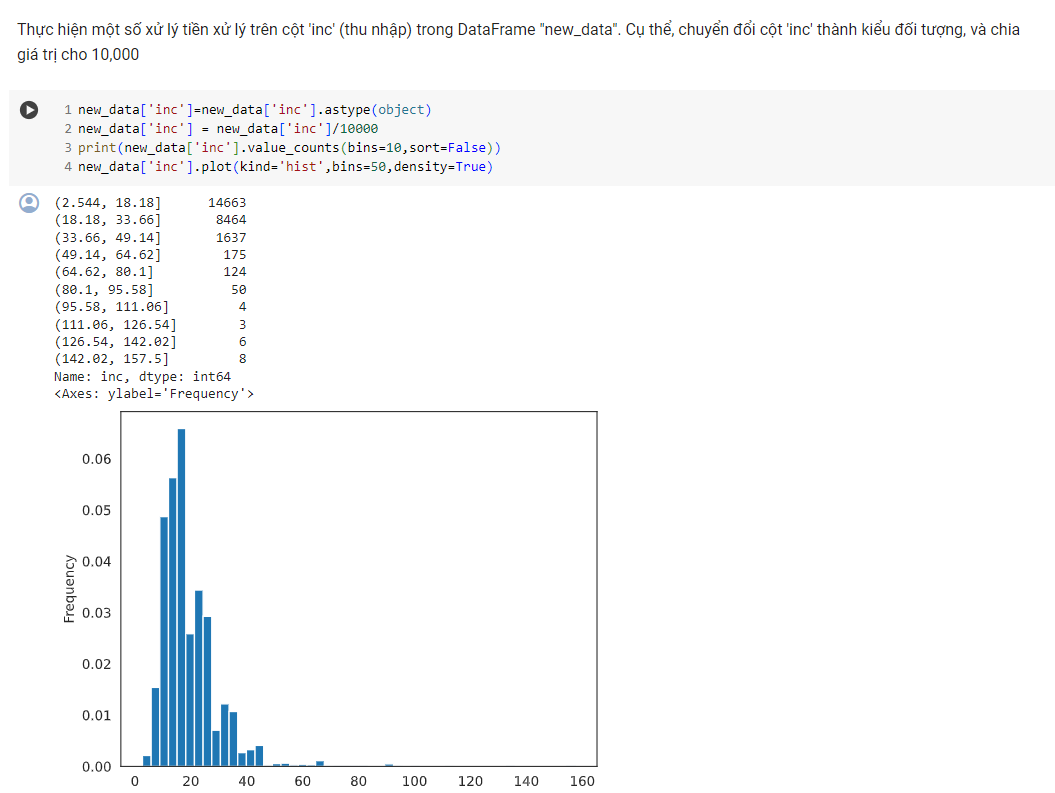
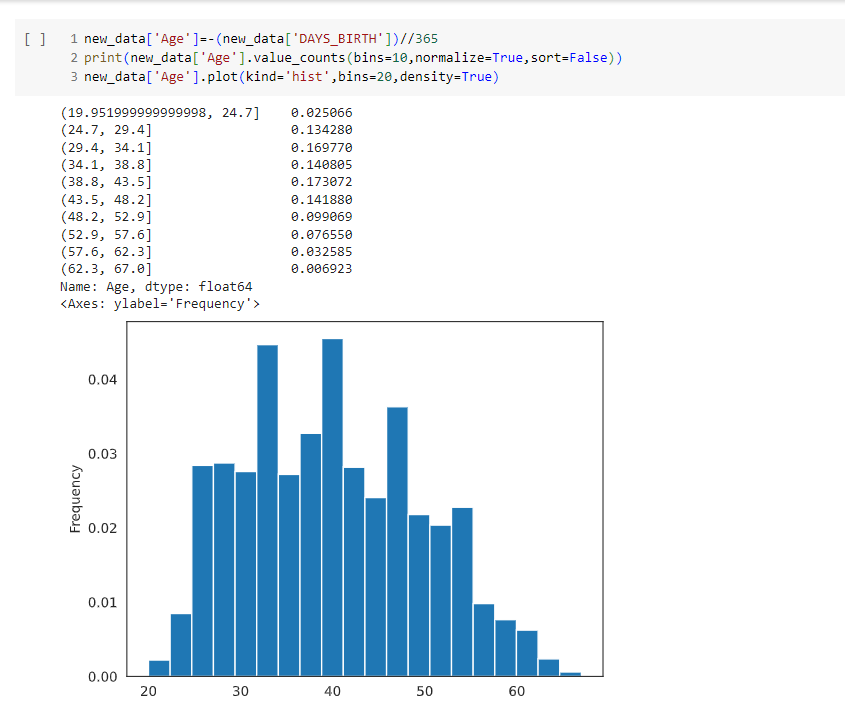
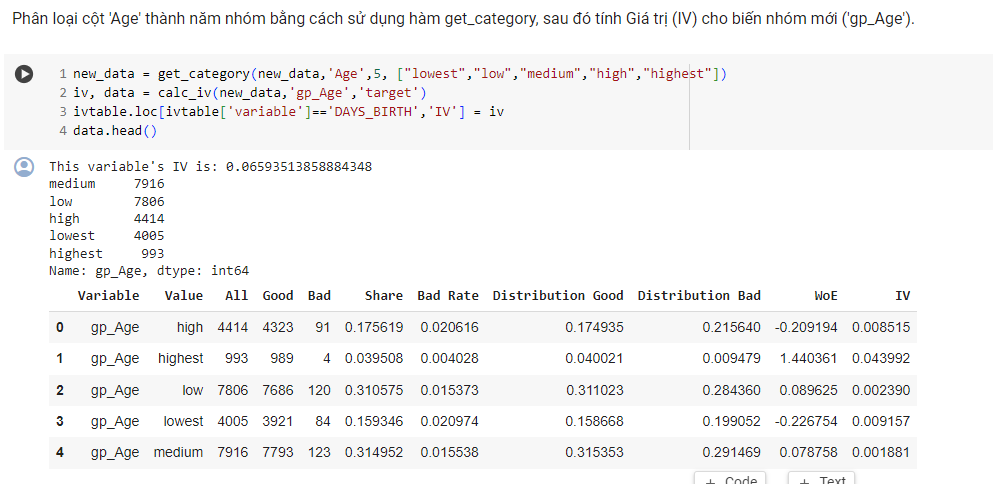
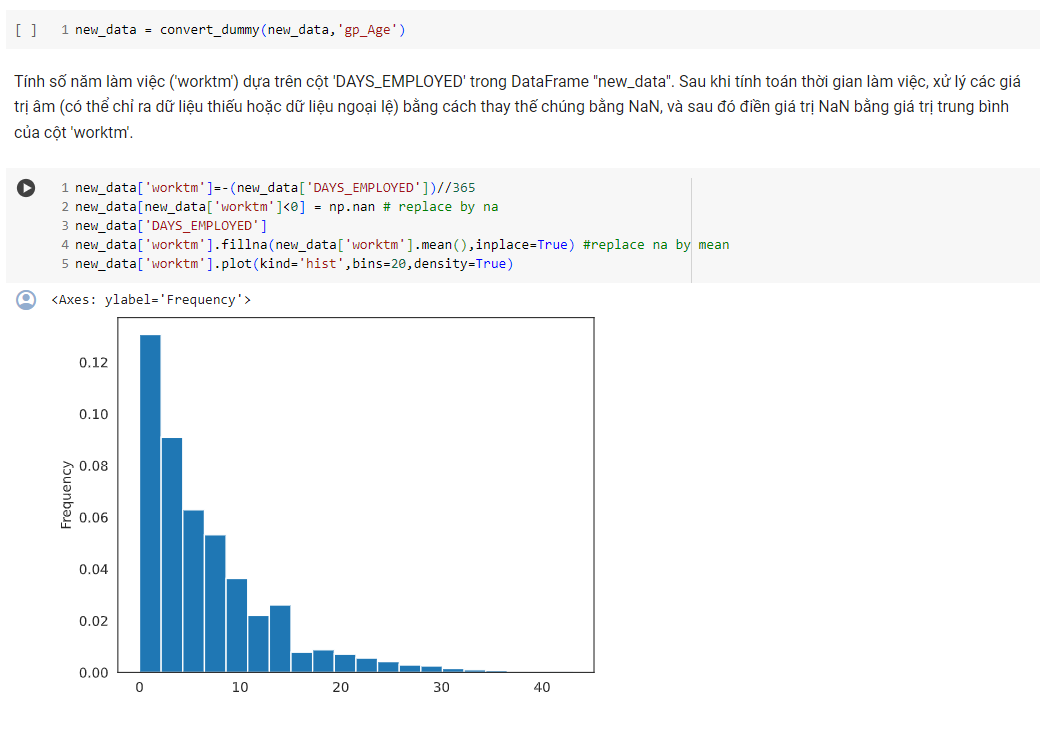
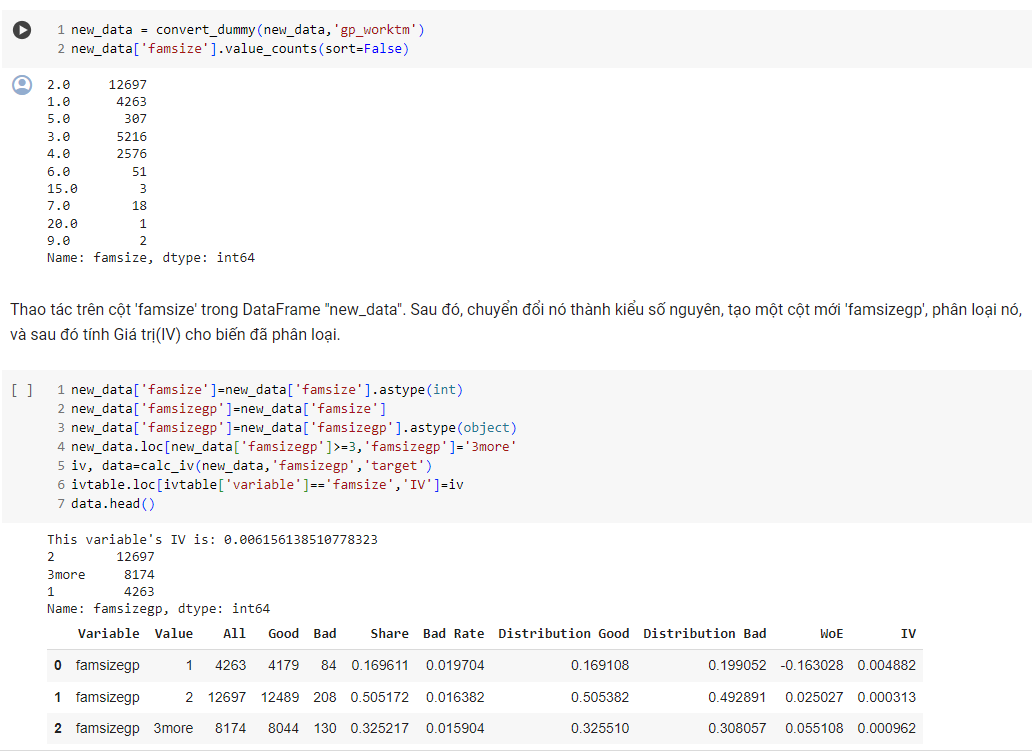
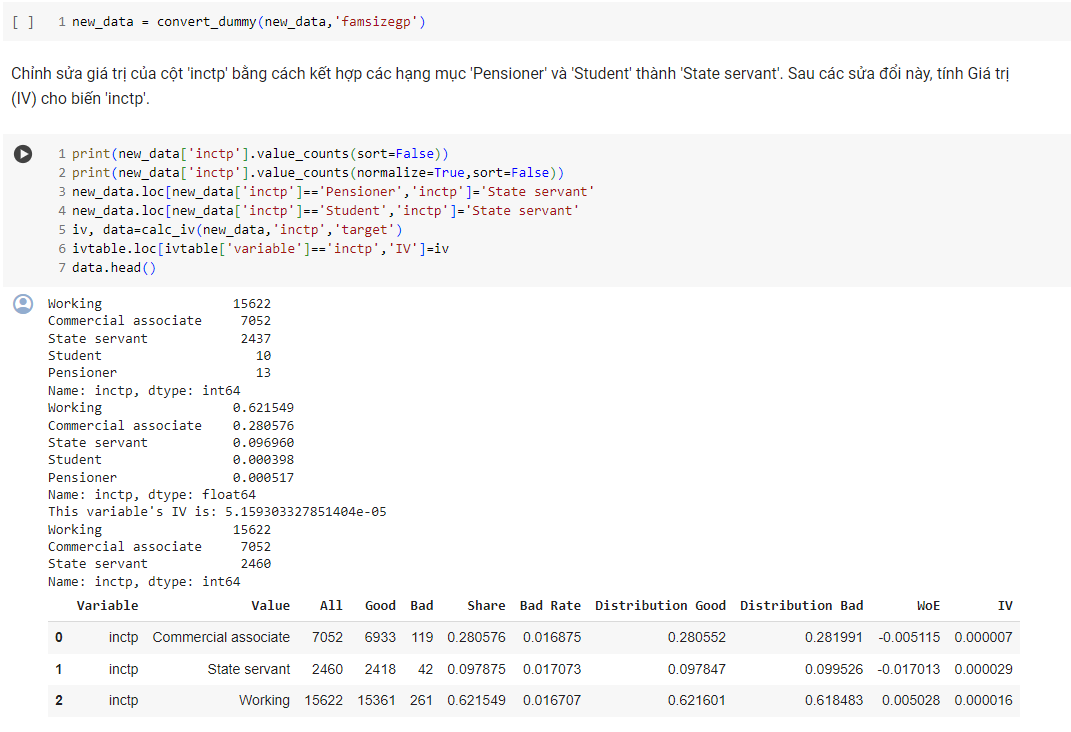
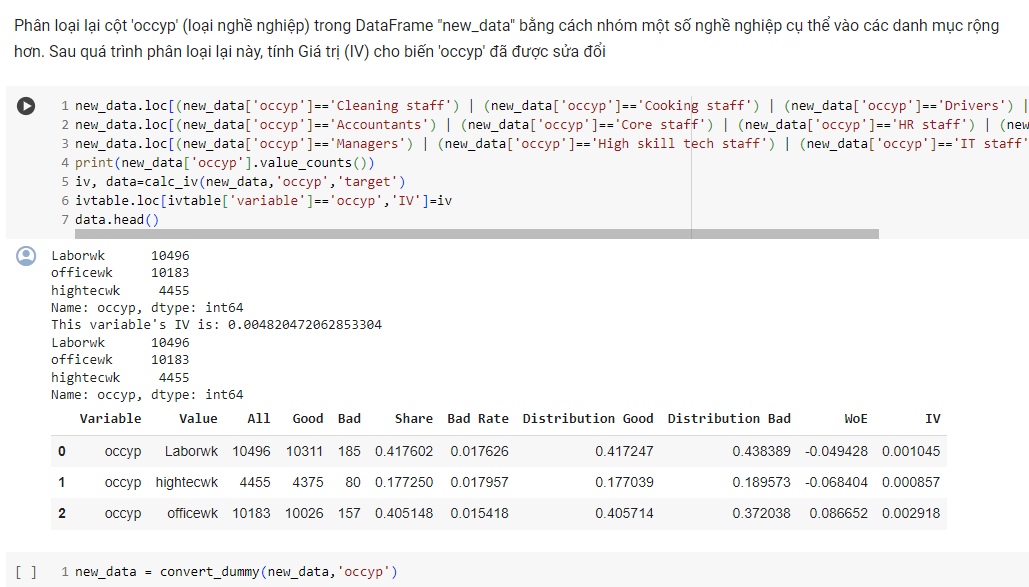
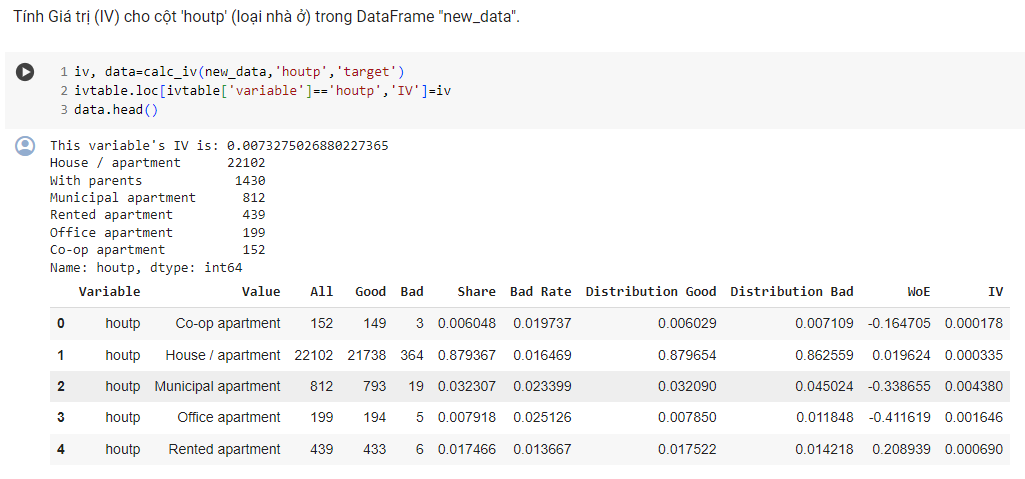
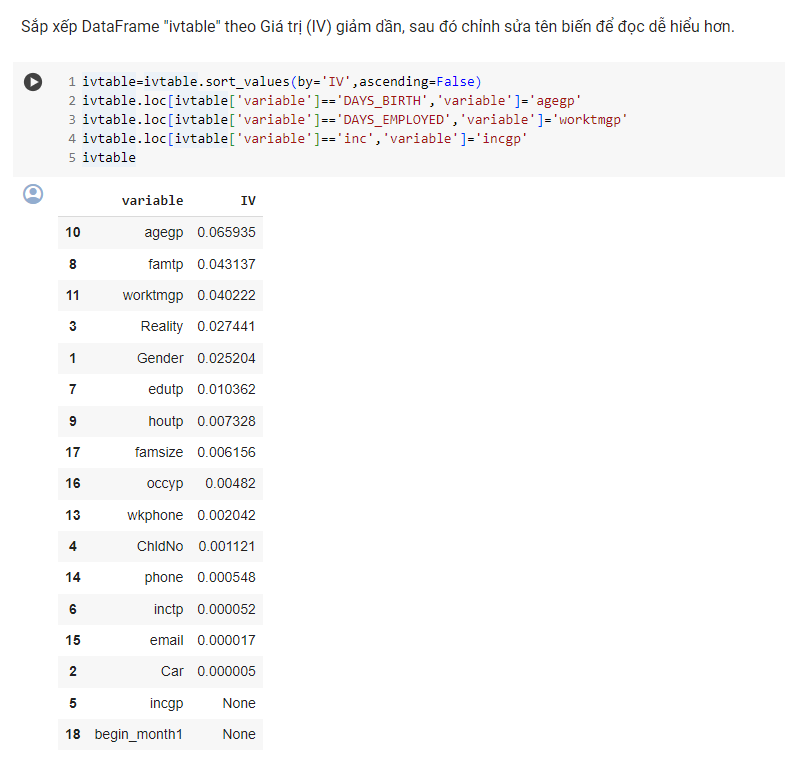
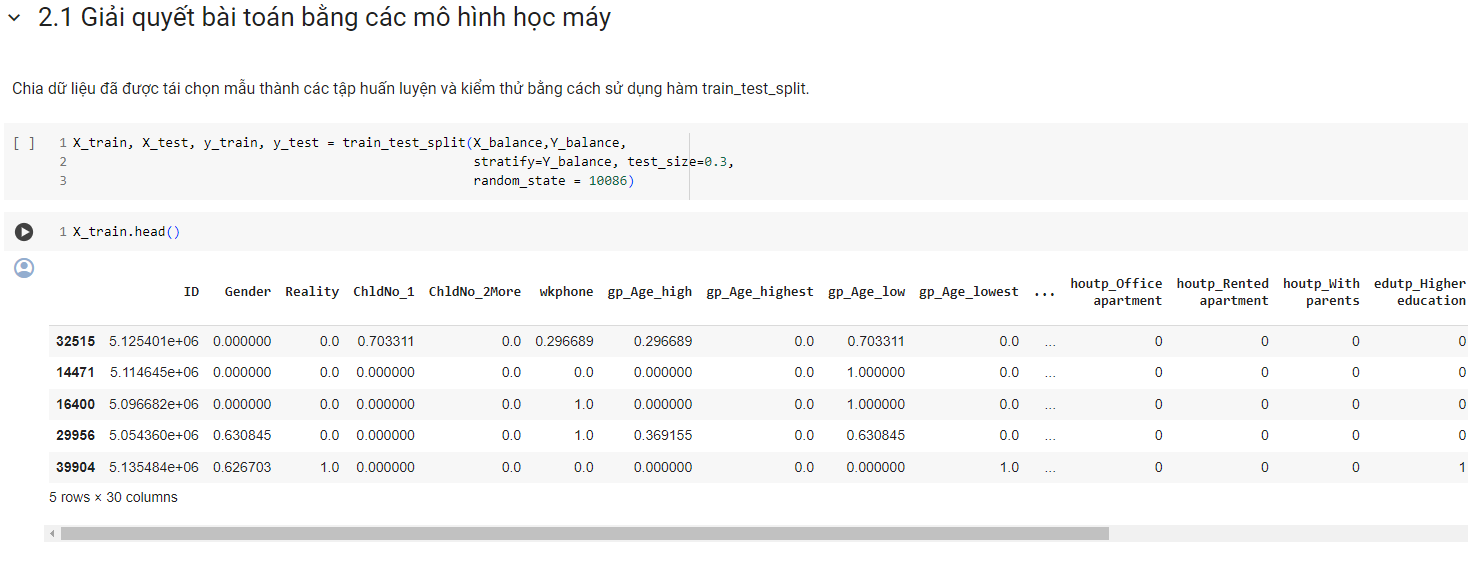
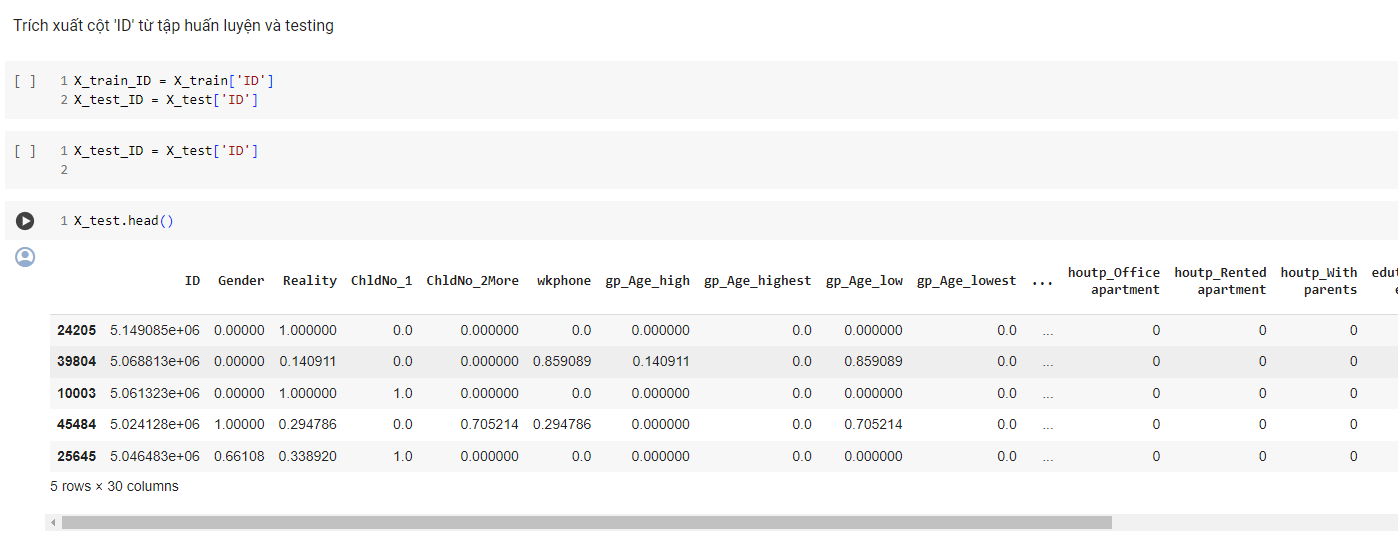
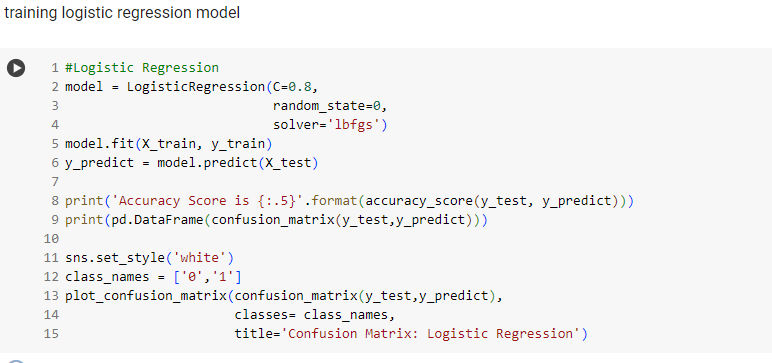
Sau khi đã xây dựng các mô hình học máy, tiến hành áp dụng một số kỹ thuật để phòng tránh Overfitting tăng độ chính xác cho mô hình:

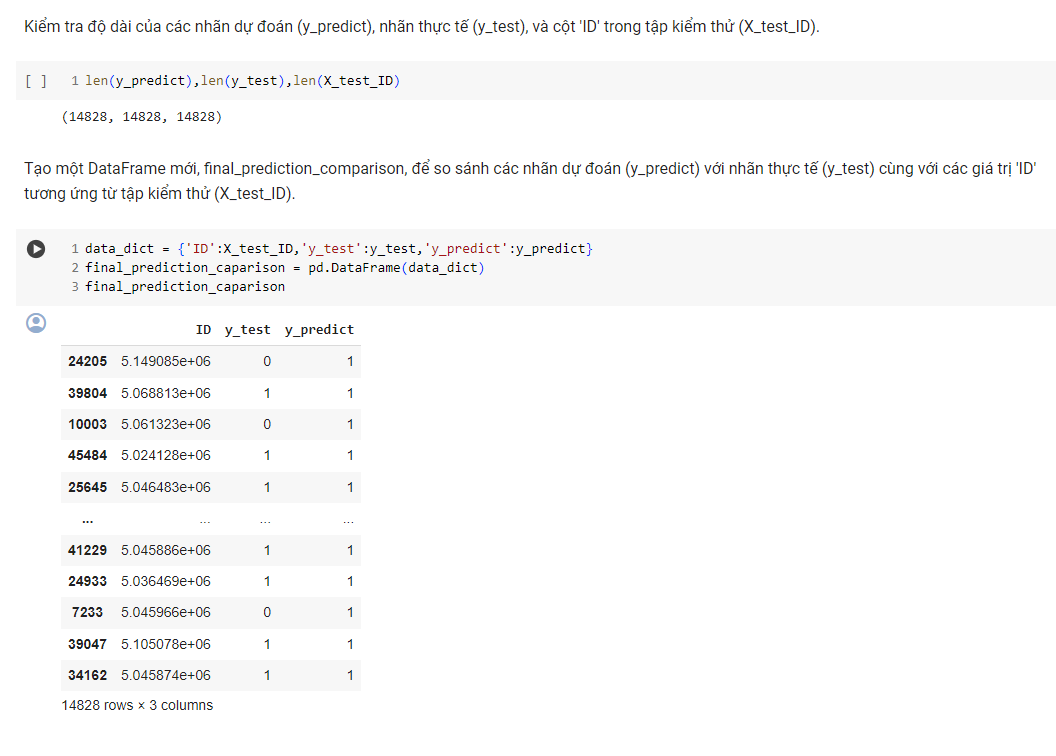
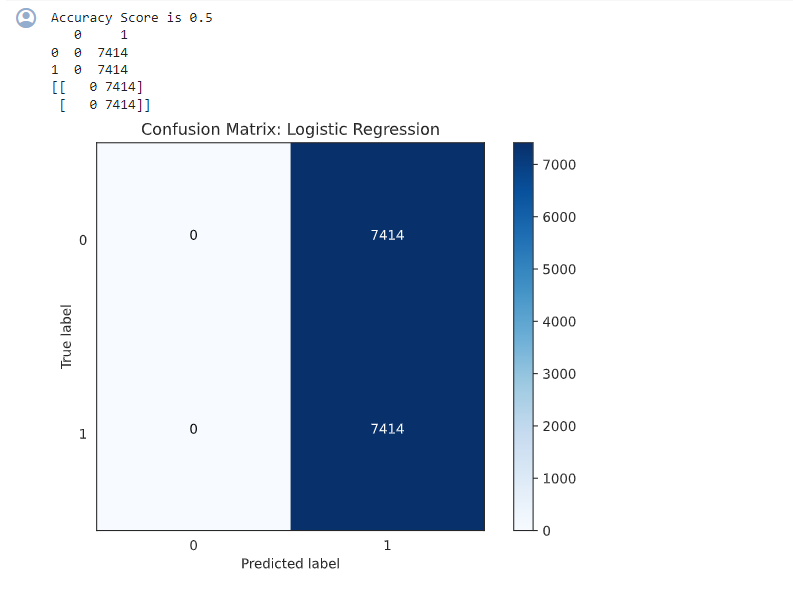
* + 1. Giảm Độ Phức Tạp của Mô Hình: Giảm độ sâu của cây quyết định, số lượng lớp và đơn vị ẩn trong các mô hình neural network có thể giúp giảm độ phức tạp của mô hình và nguy cơ overfitting.
    2. Dropout (Cho Neural Networks): Sử dụng kỹ thuật dropout trong các mạng neural. Dropout là quá trình ngẫu nhiên loại bỏ một số lượng đơn vị trong quá trình huấn luyện, giúp ngăn chặn overfitting.
    3. Tinh chỉnh Hyperparameters: Thực hiện tinh chỉnh hyperparameters một cách cẩn thận để điều chỉnh mô hình sao cho nó không quá phức tạp hoặc quá đơn giản
    4. Early Stopping:Sử dụng kỹ thuật early stopping để dừng quá trình huấn luyện khi hiệu suất trên tập validation không còn cải thiện nữa, giảm nguy cơ overfitting.

1. **Thực hiện bài toán với các mô hình học máy**

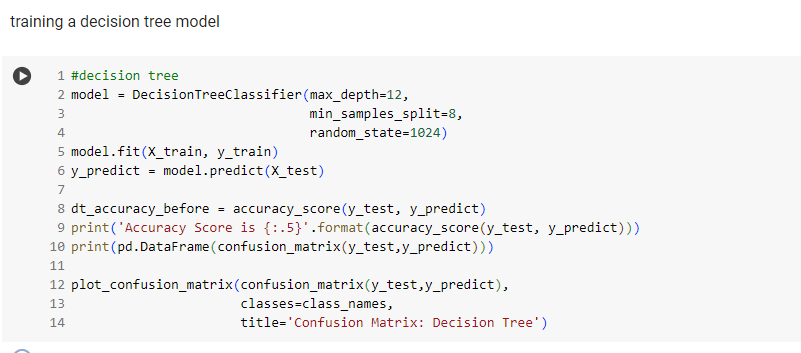


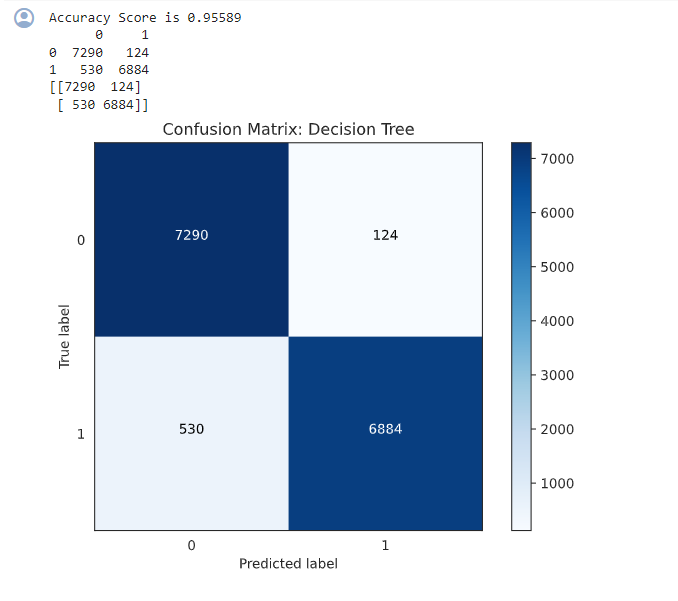


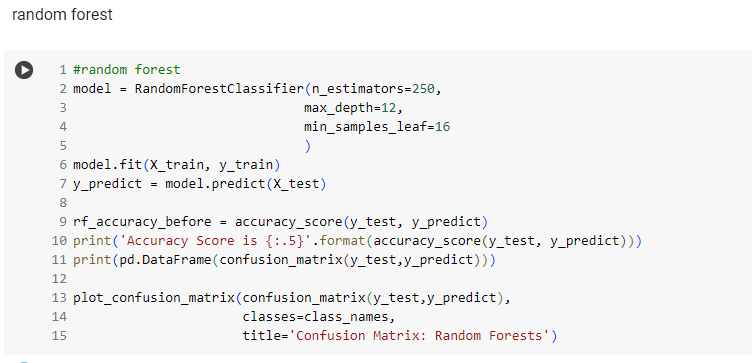


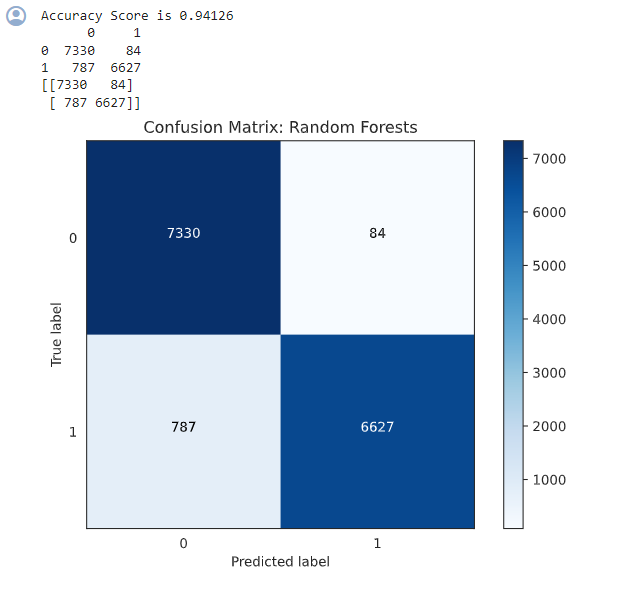
Hình 13 Kết quả học mô hình Logistic Regression



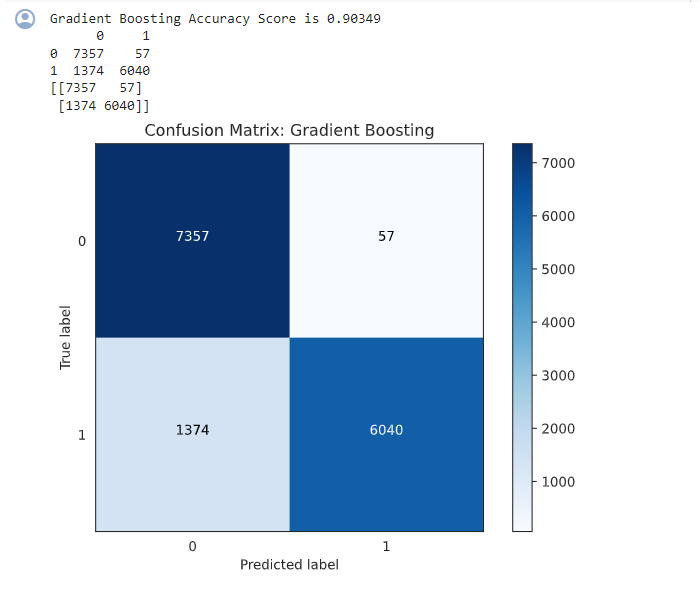


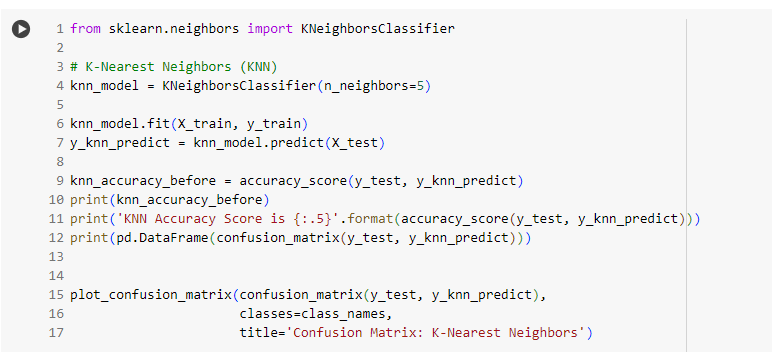
Hình 14 Kết quả học mô hình Decision Tree



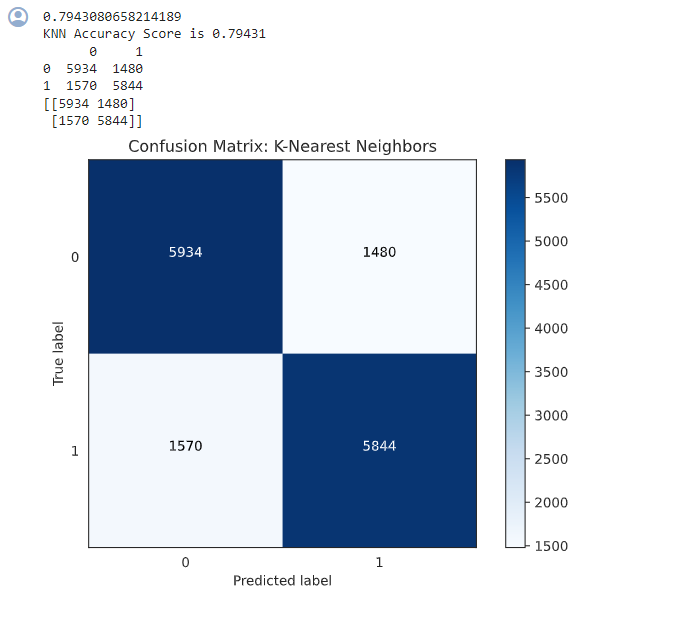


Hình 15 Kết quả học mô hình Random Forests

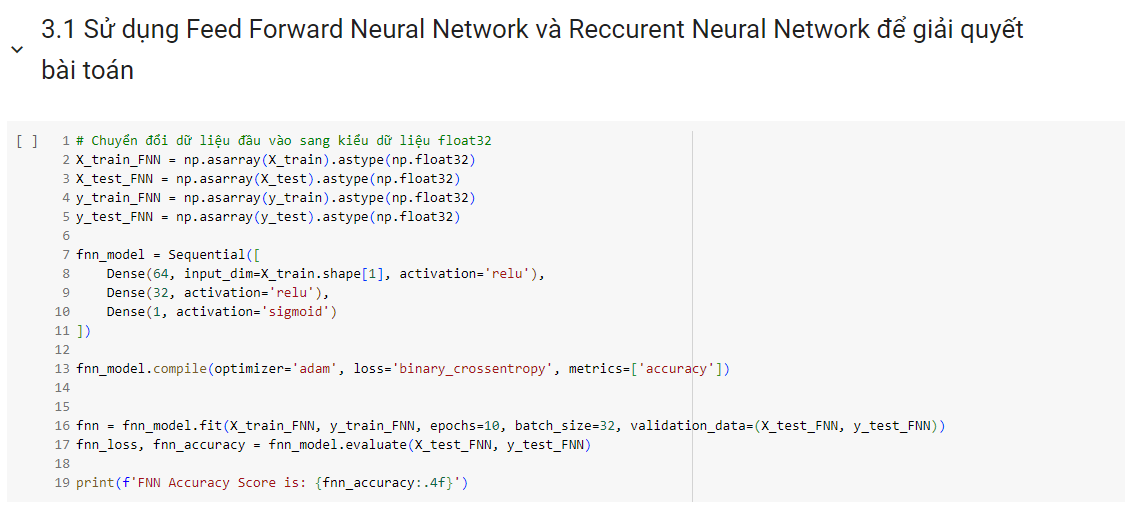


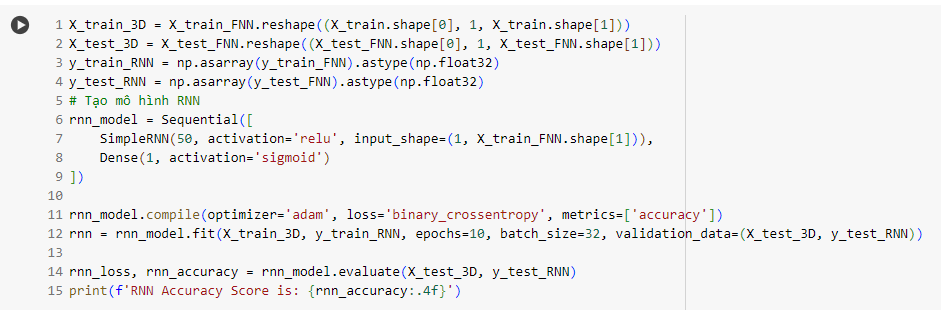


Xây dựng mô hình học máy KNN



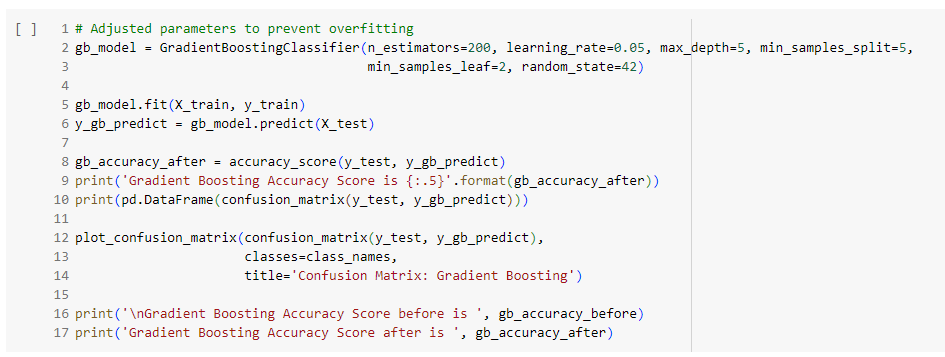
Hình 16 Kết quả học với KNN



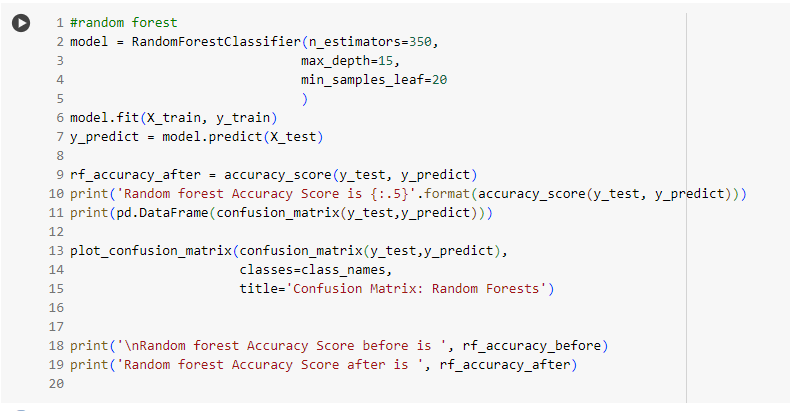




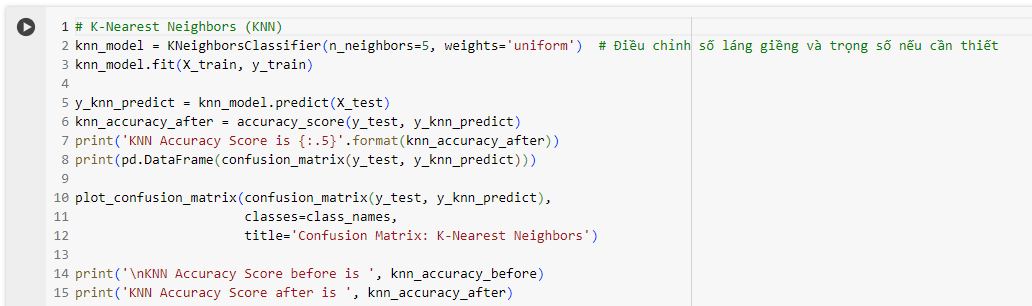
Decision Tree



Gradient Boosting phòng tránh Overfitting

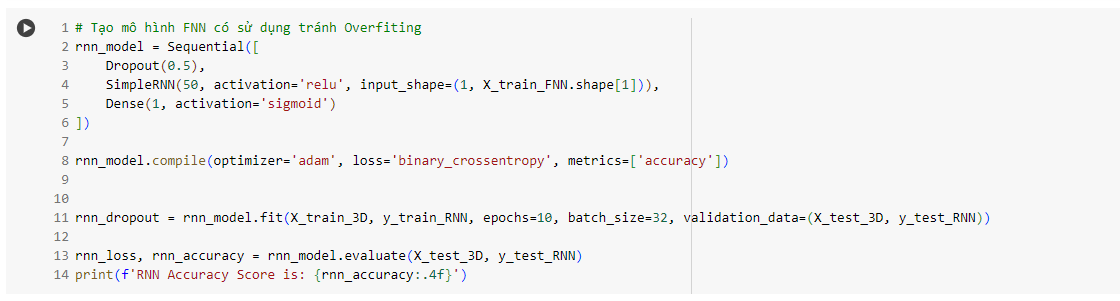


Hình 17 Mô hình Random forest phòng tránh Overfitting

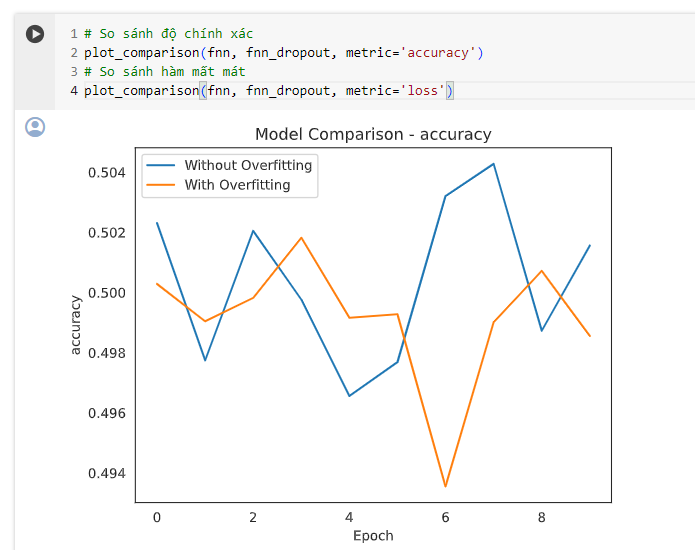


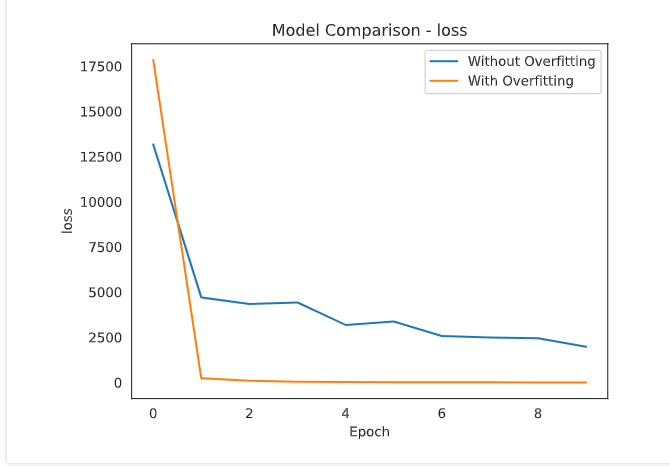


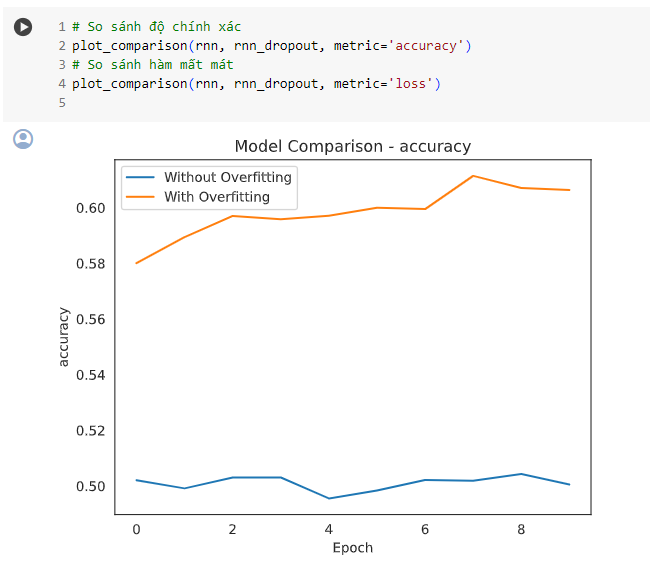
Hình 18 Tạo mô hình FNN có sử dụng tránh Overfiting

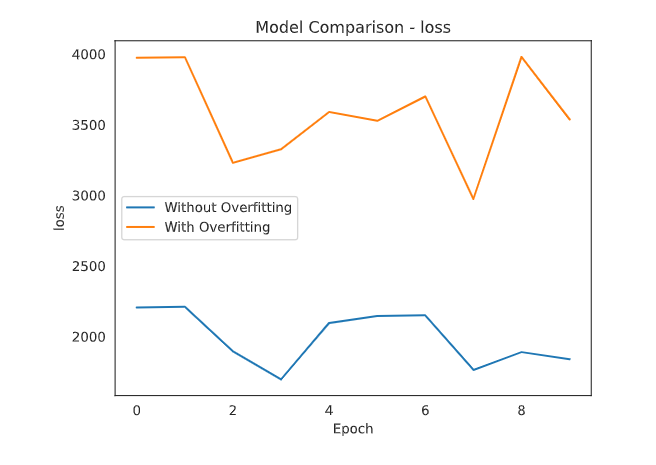


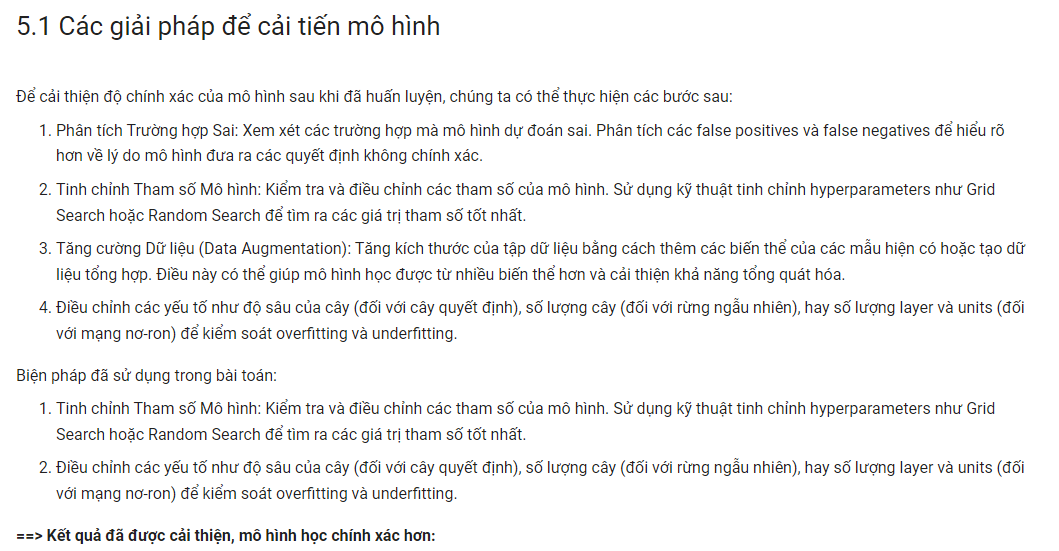
Hình 19 Tạo mô hình RNN có sử dụng tránh Overfiting











# PHẦN 3 – KẾT LUẬN

* 1. **Kiến thức sau nghiên cứu, tìm hiểu**

Qua việc nghiên cứu và áp dụng các mô hình tối ưu để giải quyết các vấn đề trong bài toán học máy, ta đã khám phá và thực hiện một loạt các khía cạnh quan trọng về Machine Learning, và sau đây là những điểm quan trọng mà nhóm em đã tìm hiểu được:

* Khái niệm, ứng dụng của các phương pháp tối ưu:
  + - Tối ưu hóa mô hình: Optimizer được sử dụng để điều chỉnh các tham số của mô hình máy học để mô hình có thể học được từ dữ liệu huấn luyện và dự đoán tốt trên dữ liệu mới.
    - Học máy sâu: Trong các mô hình học sâu như mạng nơ-ron sâu (deep neural networks), optimizer giúp điều chỉnh hàng trăm hoặc hàng nghìn tham số mô hình để tối ưu hóa hiệu suất.
    - Cải thiện hàm mất mát: Một số optimizer như Adam, SGD (Stochastic Gradient Descent), RMSprop, ... được thiết kế để tối ưu hóa hàm mất mát, giúp mô hình học nhanh hơn và tránh các điểm tối ưu cục bộ.
* Phương pháp, giải thuật để tối ưu mô hình học máy: Chúng ta đã tìm hiểu về các phương pháp tối ưu (Optimizer) như Gradient Descent, Adam, SGD (Stochastic Gradient Descent), RMSprop,... Mỗi mô hình có cách tiếp cận riêng và yêu cầu hiểu biết cụ thể về cách chúng hoạt động.
* Phân tích, so sánh các phương pháp tối ưu: Trong việc so sánh, phải xem xét mục tiêu, ưu điểm, nhược điểm của từng phương pháp. Việc so sánh này giúp chúng ta chọn ra mô hình tốt nhất cho từng tình huống cụ thể.

Ngoài ra nhóm đã giải quyết bài toán học máy dựa trên tiệp dữ liệu Credit Card Approval Prediction bao gồm thống kê, phân tích, ứng dụng các mô hình cơ bản…

* 1. **Những điều cần lưu ý**
* Sự quan trọng của việc tiền xử lý dữ liệu: Dữ liệu sạch và chuẩn hoá là yếu tố quyết định cho hiệu suất của mô hình.
* Khả năng chọn mô hình phù hợp: Việc hiểu rõ loại bài toán và dữ liệu giúp chọn ra mô hình phù hợp.
* Quá trình đánh giá và điều chỉnh mô hình: Kiểm tra hiệu suất và tối ưu hóa mô hình là một phần quan trọng của quá trình học máy.
* Overfitting và giải quyết nó: Overfitting là một vấn đề thường gặp và cần được kiểm soát thông qua các biện pháp như Regularization.
* Những kiến thức và kinh nghiệm thu thập từ môn học này sẽ có giá trị lớn khi áp dụng vào các dự án thực tế và nghiên cứu tiếp theo trong lĩnh vực Machine Learning.

# TÀI LIỆU THAM KHẢO

**Tiếng Việt**

* + **Slide môn Nhập môn học máy của trường đại học Tôn Đức Thắng**
  + [**https://viblo.asia/p/optimizer-hieu-sau-ve-cac-thuat-toan-toi-uu-gdsgdadam-Qbq5QQ9E5D8**](https://viblo.asia/p/optimizer-hieu-sau-ve-cac-thuat-toan-toi-uu-gdsgdadam-Qbq5QQ9E5D8)

**Tiếng Anh**

**[1] D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning representations by back propagating errors, nature 323 (1986) 533536.**

**[2] Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, L. D. Jackel, Handwritten digit recognition with a back-propagation network, in: Advances in neural information processing systems, pp. 396404.**

**[3] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (1997) 17351780.**

**[4] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770778.**

**[5] G. Huang, Z. Liu, L. Van Der Maaten, K. Q. Weinberger, Densely connected convolu tional networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 47004708**