**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

****

**ĐỒ ÁN**

**HỌC PHẦN MÁY HỌC ỨNG DỤNG**

**Đề tài:**

**DỰ ĐOÁN GIAN LẬN TÀI CHÍNH BẰNG PHƯƠNG PHÁP MÁY HỌC**

|  |  |  |
| --- | --- | --- |
| **Giáo viên hướng dẫn** |  | **Nhóm sinh viên thực hiện** |
| TS. Lưu Tiến Đạo |  | Lâm Bá Thành – B2207564 |
|  |  | Nguyễn Phước Phát – B2207554 |
|  |  | Phương Thành Đạt – B2207516 |

**Cần thơ, tháng 03 năm 2025**

# NHẬN XÉT CỦA GIẢNG VIÊN

*Cần Thơ, ngày…. Tháng….năm*

(Ký và ghi rõ họ tên)

# MỤC LỤC

[NHẬN XÉT CỦA GIẢNG VIÊN 2](#_Toc190556737)

[MỤC LỤC 3](#_Toc190556738)

[PHÂN CÔNG CÔNG VIỆC 4](#_Toc190556739)

[PHẦN NỘI DUNG 5](#_Toc190556740)

[1.Mô tả bài toán 5](#_Toc190556741)

[2.Mô tả dữ liệu, ý nghĩa của dữ liệu 5](#_Toc190556742)

[2.1.Mô tả dữ liệu 5](#_Toc190556743)

[2.2.Ý nghĩa của bộ dữ liệu 5](#_Toc190556744)

[3.Phân tích dữ liệu và lụa chọn mô hình 6](#_Toc190556745)

[3.1.Phân tích dữ liệu 7](#_Toc190556746)

[3.2.Tiền xử lý dữ liệu 10](#_Toc190556747)

[3.3.Lựa chọn mô hình 19](#_Toc190556748)

[4.Cấu hình máy tính 22](#_Toc190556749)

[5.Huấn luyện mô hình 22](#_Toc190556750)

[5.1.Đọc dữ liệu 22](#_Toc190556751)

[5.2.Chọn số lượng k neighbors phù hợp 22](#_Toc190556752)

[5.3.Chọn max depth phù hợp 24](#_Toc190556753)

[5.4.Chia dữ liệu 25](#_Toc190556754)

[6.Đánh giá mô hình 26](#_Toc190556755)

[6.1.Đánh giá mô hình phân lớp 26](#_Toc190556756)

[6.2.Nhận xét kết quả thực nghiệm 31](#_Toc190556757)

[PHẦN KẾT LUẬN 32](#_Toc190556758)

[1.Kết quả đạt được 32](#_Toc190556759)

[2.Hướng phát triển 32](#_Toc190556760)

[TÀI LIỆU THAM KHẢO 33](#_Toc190556761)

# PHÂN CÔNG CÔNG VIỆC

|  |  |
| --- | --- |
| **Sinh Viên** | **Công Việc** |
| Lâm Bá Thành – B2207564 |  |
| Nguyễn Phước Phát – B2207554 |  |
| Phương Thành Đạt – B2207516 |  |

# PHẦN NỘI DUNG

## Mô tả bài toán:

## Đặt vấn đề:

Gian lận tài chính là một vấn đề nghiêm trọng, gây thiệt hại hàng tỷ USD mỗi năm cho các tổ chức và nền kinh tế. Việc phát hiện sớm các hành vi gian lận trong báo cáo tài chính đóng vai trò quan trọng trong việc giảm thiểu rủi ro, bảo vệ tài sản, và tăng cường tính minh bạch của hệ thống tài chính.

## Mục tiêu:

Trong đồ án, chúng tôi sử dụng bộ dữ liệu Audit Data từ [UCI Machine Learning Repository](https://archive.ics.uci.edu/ml/datasets/Audit+Risk) để xây dựng mô hình phân loại doanh nghiệp thành hai nhóm: **Gian lận** và **Không gian lận**. Mô hình được xây dựng dựa trên các đặc trưng tài chính và lịch sử kiểm toán, nhằm hỗ trợ các tổ chức trong việc phát hiện sớm hành vi gian lận, từ đó đưa ra các biện pháp kiểm soát phù hợp.

## Phương pháp:

* Sử dụng K-Nearest Neighbors (KNN), Naïve Bayes, Decision Trees để xây dựng mô hình.
* Đánh giá mô hình bằng các chỉ số: **Precision**, **Recall**, **F1-Score**, **Accuracy**.

## Mô tả dữ liệu, ý nghĩa của dữ liệu

### Mô tả dữ liệu:

Tên tập dữ liệu: Autit\_risk.

Tập dữ liệu thể hiện các yếu tố rủi ro của các doanh nghiệp thuộc nhiều lĩnh vực khác nhau ở hiện tại và quá khứ. Thông tin về các ngành và số lượng doanh nghiệp có trong tập dữ liệu là: Irrigation (114), Public Health (77), Buildings and Roads (82), Forest (70), Corporate (47), Animal Husbandry (95), Communication (1), Electrical (4), Land (5), Science and Technology (3), Tourism (1), Fisheries (41), Industries (37), Agriculture (200).

Số lượng cột của tập dữ liệu: 27

Số lượng dòng của tập dữ liệu: 777

### Ý nghĩa của bộ dữ liệu:

**Sector\_score**: Giá trị rủi ro dựa vào lịch sử doanh nghiệp.

**LOCATION\_ID**: Mã tỉnh/thành phố.

**PARA\_A**: Mức chênh lệch giữa dữ liệu báo cáo và thực tế khi kiểm tra.

**Score\_A:** Điểm số rủi ro của **PARA\_A.**

**PARA\_B**: Mức chênh lệch giữa dữ liệu báo cáo và thực tế khi kiểm tra (giống *PARA\_A* nhưng theo tiêu chí khác).

**Score\_B:** Điểm số rủi ro của **PARA\_B.**

**Risk\_A, Risk\_B, Risk\_C, Risk\_D, Risk\_E:** Chỉ số rủi ro theo tiêu chí A, B, C, D, E.

**Score**: Điểm tổng hợp thể hiện mức độ rủi ro của doanh nghiệp.

**TOTAL**: Tổng mức chênh lệch từ tất cả các báo cáo trong tập dữ liệu.

**PROB (Probability):** Xác suất xảy ra sai lệch hoặc gian lận trong dữ liệu kiểm toán

**numbers**: Mức độ chênh lệch giữa dữ liệu kiểm tra và báo cáo trong lịch sử.

**Marks**: Điểm số tổng quát hoặc đánh giá mức độ tin cậy của dữ liệu.

**Money\_Value**: Tổng số tiền sai sót được phát hiện trong các cuộc kiểm toán trước đây.

**MONEY\_Marks**: Điểm số đánh giá mức độ nghiêm trọng của **Money\_Value**.

**District:** Loại quận/huyện nơi doanh nghiệp đặt trụ sở.

**Loss**: Doanh nghiệp có bị thua lỗ trong năm trước hay không (0: Không, 1: Có).

**LOSS\_SCORE**: Điểm số đánh giá rủi ro dựa trên **Loss**.

**History**: Tổng mức lỗ mà công ty phải chịu trong vòng 10 năm qua.

**History\_score**: Điểm số đánh giá rủi ro dựa trên **History**.

**Audit\_Risk:** Điểm tổng hợp đánh giá rủi ro kiểm toán.

**Inherent\_Risk:** Mức độ rủi ro nội tại của doanh nghiệp (do đặc điểm ngành nghề, môi trường kinh doanh).

**CONTROL\_RISK**: Mức độ sai sót tiềm ẩn trong quá trình kiểm toán tài chính.

**Detection\_Risk**: Mức độ sai sót của kiểm toán viên trong quá trình kiểm toán (khả năng không phát hiện gian lận hoặc sai sót).

**Risk:** Kết quả đánh giá về việc doanh nghiệp có gian lận hay không (Nhãn).

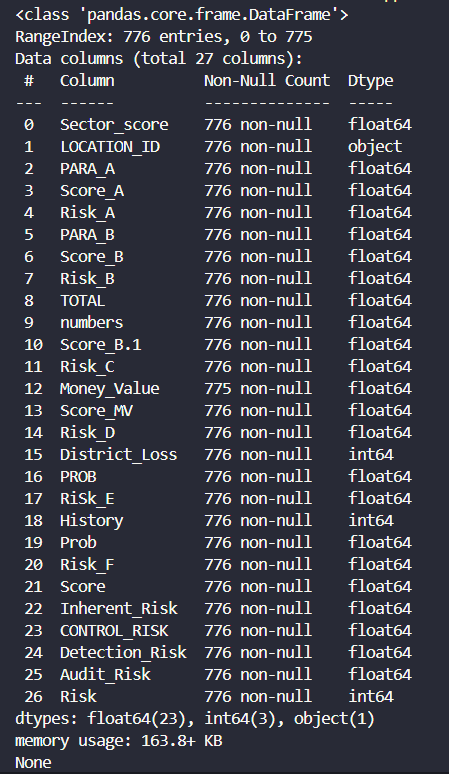
## Phân tích dữ liệu và lụa chọn mô hình:

### Phân tích dữ liệu:

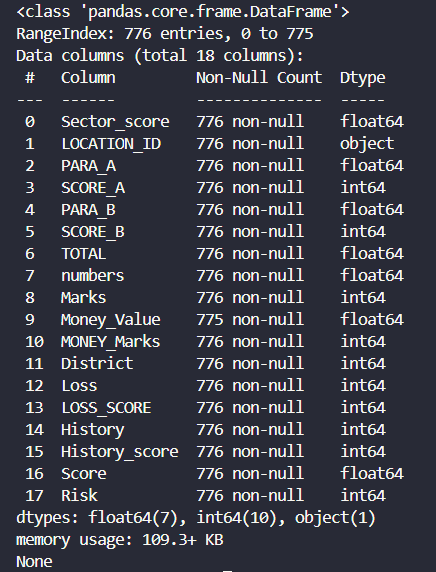
Có 2 tập dữ liệu là audit risk (27 thuộc tính) và trial (18 thuộc tính).

Thuộc tính nhãn là cột **Risk** trong bảng audit risk, nhãn có dữ liệu rời rạc (0 và 1), do đó đây là bài toán **phân lớp.**

* **Tổng quan về tập dữ liệu:**
* **audit\_risk.csv:**

****

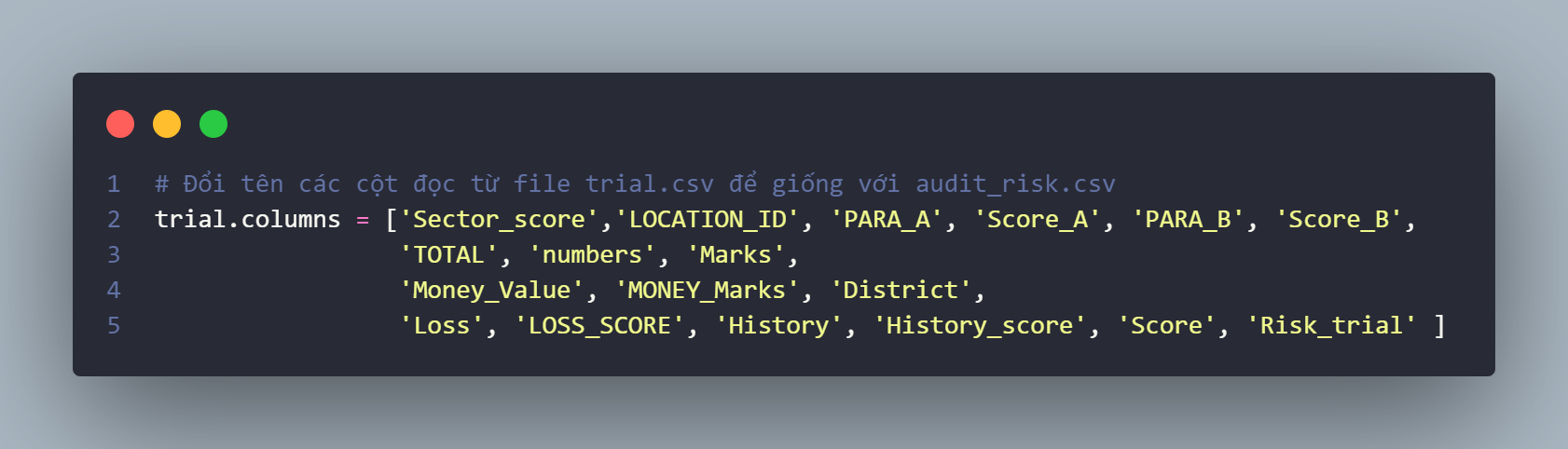
* **trial.csv:**



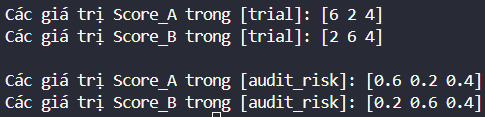
* **Các bước cho việc tiền xử lý dữ liệu:**
* Cần gộp 2 tập dữ liệu lại thành 1 tập dữ liệu để phân tích.
* Kiểm tra các giá trị bị thiếu (Null/NaN).
* Chỉnh sửa các giá trị chưa được chuẩn hóa.
* Kiểm tra kiểu dữ liệu, loại bỏ các giá trị lỗi.
* Kiểm tra sự tương quan dữ liệu, loại bỏ những trường dữ liệu tương quan cao, chỉ giữ lại 1 trường dữ liệu trong số chúng. Ưu tiên giữ lại trường dữ liệu có tồn tại trong trial.csv.
* Kiểm tra sự cân bằng nhãn.

### Tiền xử lý dữ liệu:

Gán tên các trường dữ liệu trong file trial.csv để đảm bảo chúng cùng tên với các trường dữ liệu đọc từ file audit\_risk.csv. Đổi tên trường dữ liệu Risk trong trong tập dữ liệu trial thành Risk\_trial nhằm tránh trùng với nhãn của tập dữ liệu audit\_risk.

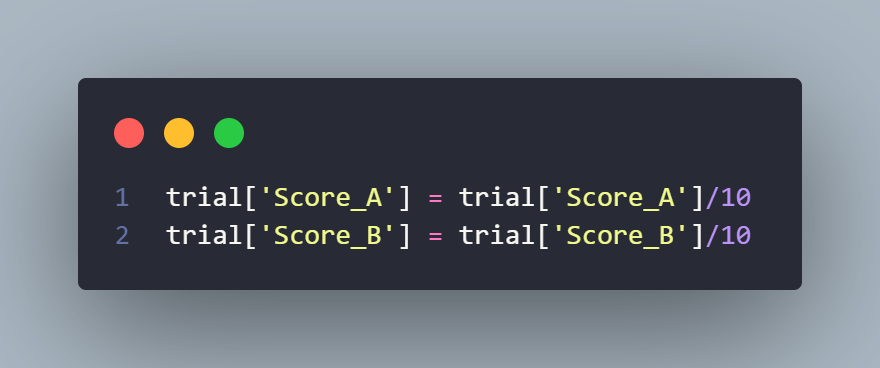


Do cột Score\_A và Score\_B trong file trial.csv được lưu trữ khác với audit\_risk.csv nên cần chuẩn hóa lại bằng các cột Score\_A và Score\_B trong audit\_risk.csv.

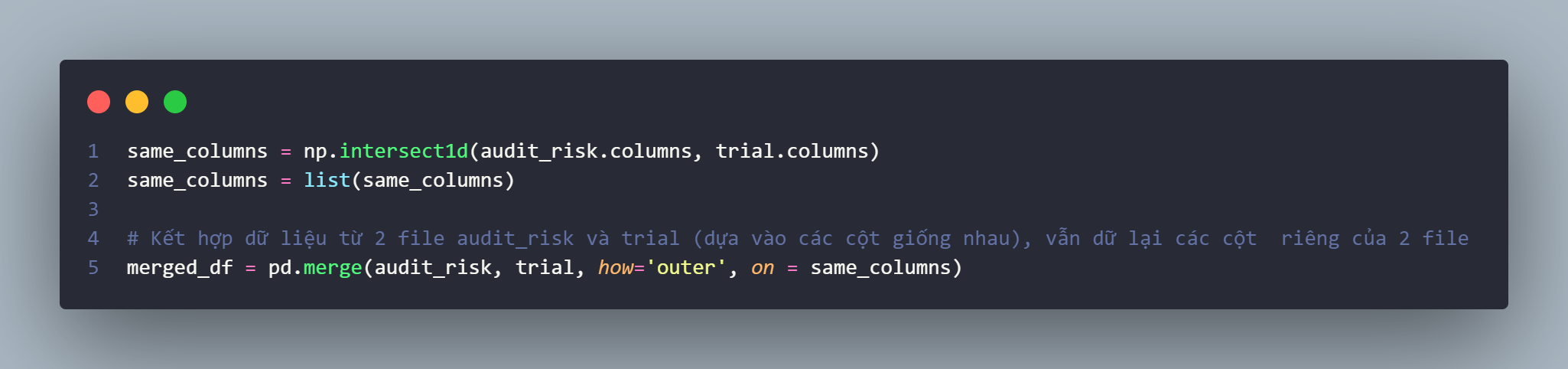


Các dữ liệu “Score\_A” và “Score\_B” trong tập dữ liệu trial có tỉ lệ giá trị gấp 10 lần trong tập dữ liệu audit\_risk nên cần điều chỉnh lại.

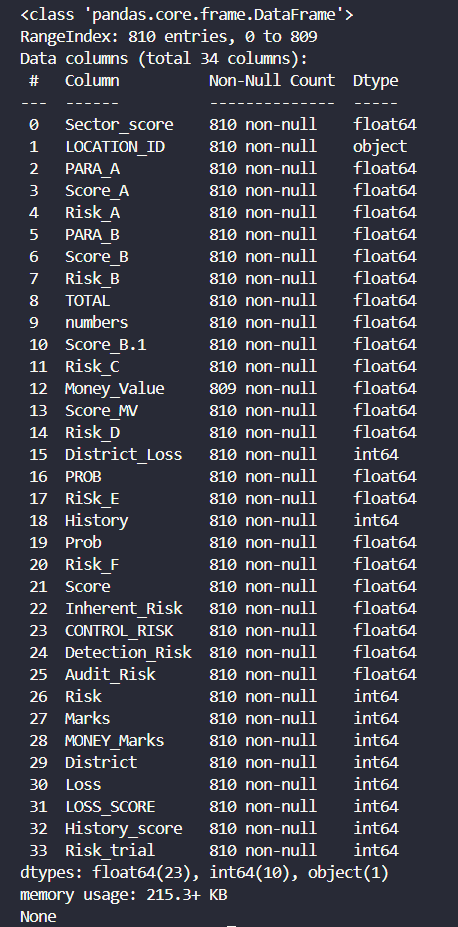
Chuẩn hóa dữ liệu:



Gộp 2 tập dữ lại dựa vào các trường dữ liệu trùng nhau giữa 2 tập dữ liệu:



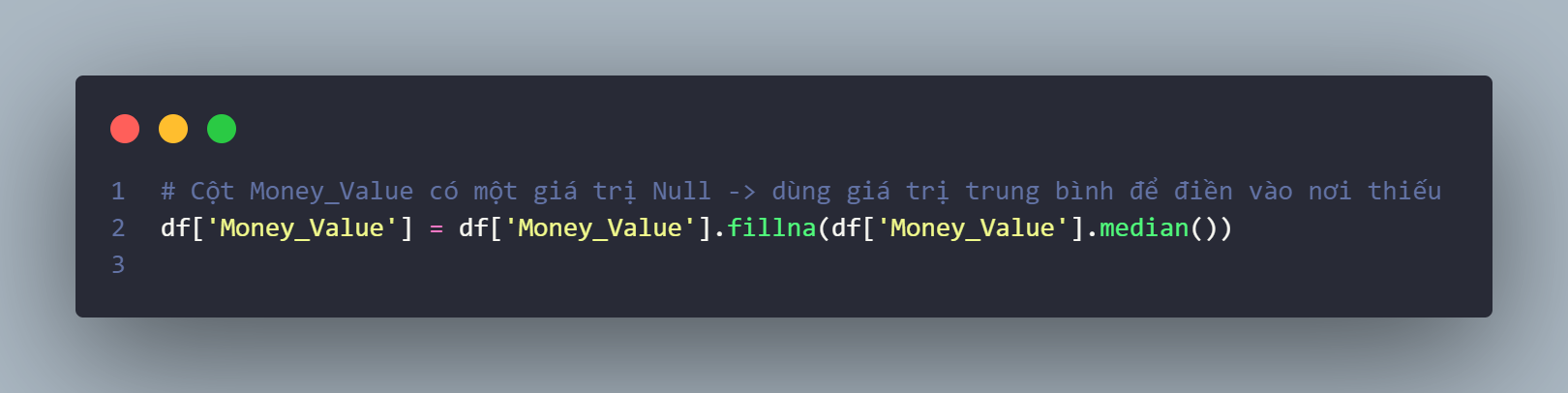
Kết quả:



Xóa nhãn của Tập dữ liệutrial (Risk\_trial) vì có thể tồn tại một số nhãn mâu thuẫn với nhãn trong Tập dữ liệuaudit\_risk.

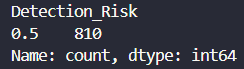


Do dữ liệu vừa gộp tại Money\_Value có 809 dòng non-null, trong khi các trường dữ liệu khác có 810 dòng non-null. Vậy có một giá trị bị thiếu tại Money\_Value. Cần xử lý bằng cách điền vào nơi bị thiếu giá trị trung bình của Money\_Value.

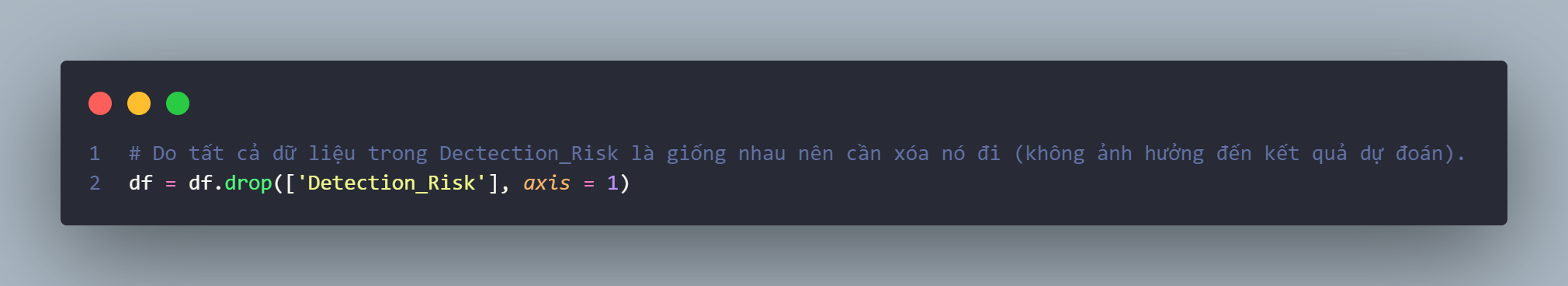


In ra các giá trị khác nhau lưu trong các trường dữ liệu:

Phát hiện tại trường dữ liệu Detection\_Risk tất cả dữ liệu bên trong đều là 0.5 nên Detection\_Risk vô ích trong việc dự đoán nên không cần giữ lại.

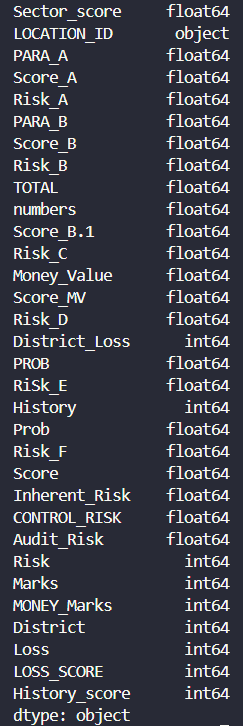


Xóa Detection\_Risk khỏi tập dữ liệu.



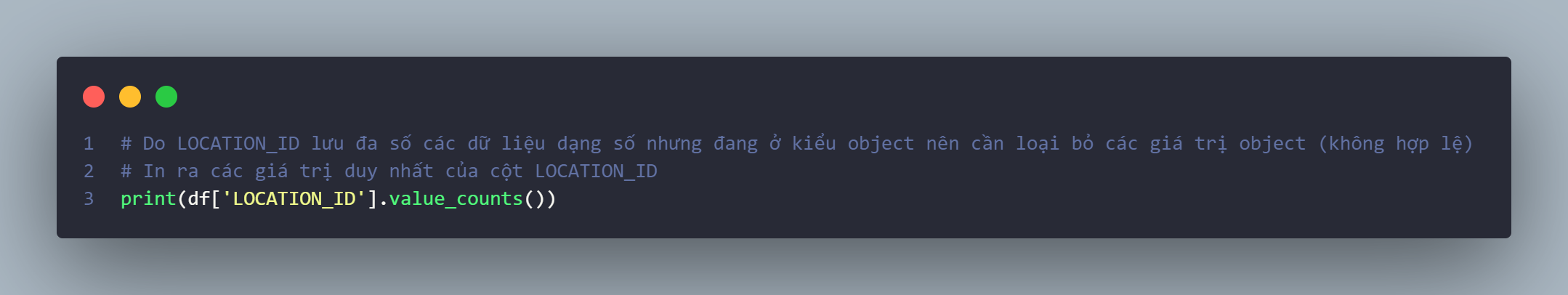
Kiểm tra kiểu dữ liệu của các trường dữ liệu:



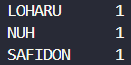


Các dữ liệu đầu vào đều là số, nhưng LOCATION\_ID có kiểu object.

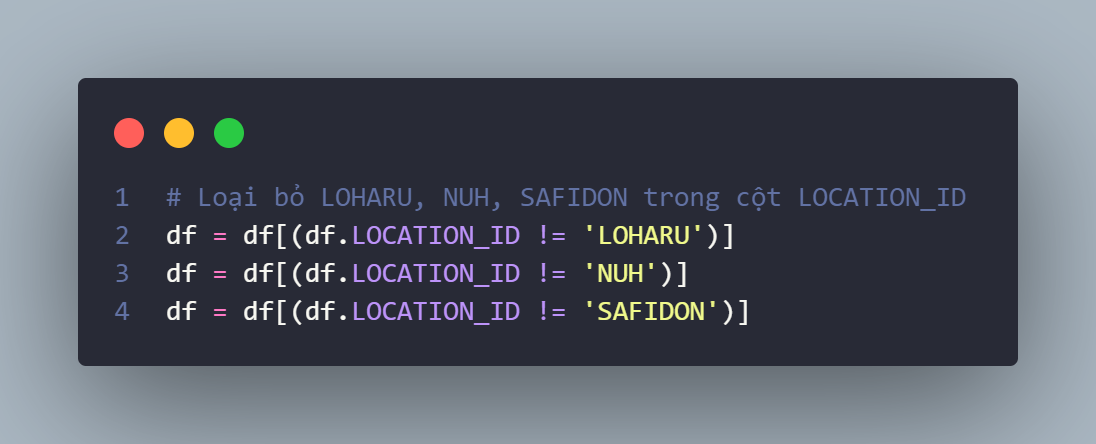
Kiểm tra các dữ liệu trong LOCATION\_ID:



Trong số 810 giá trị chỉ có 3 giá trị tương ứng với LOHARU, NUH và SAFIDON.



Vậy đó là các giá trị xuất hiện do lỗi nhập liệu cần phải xóa.

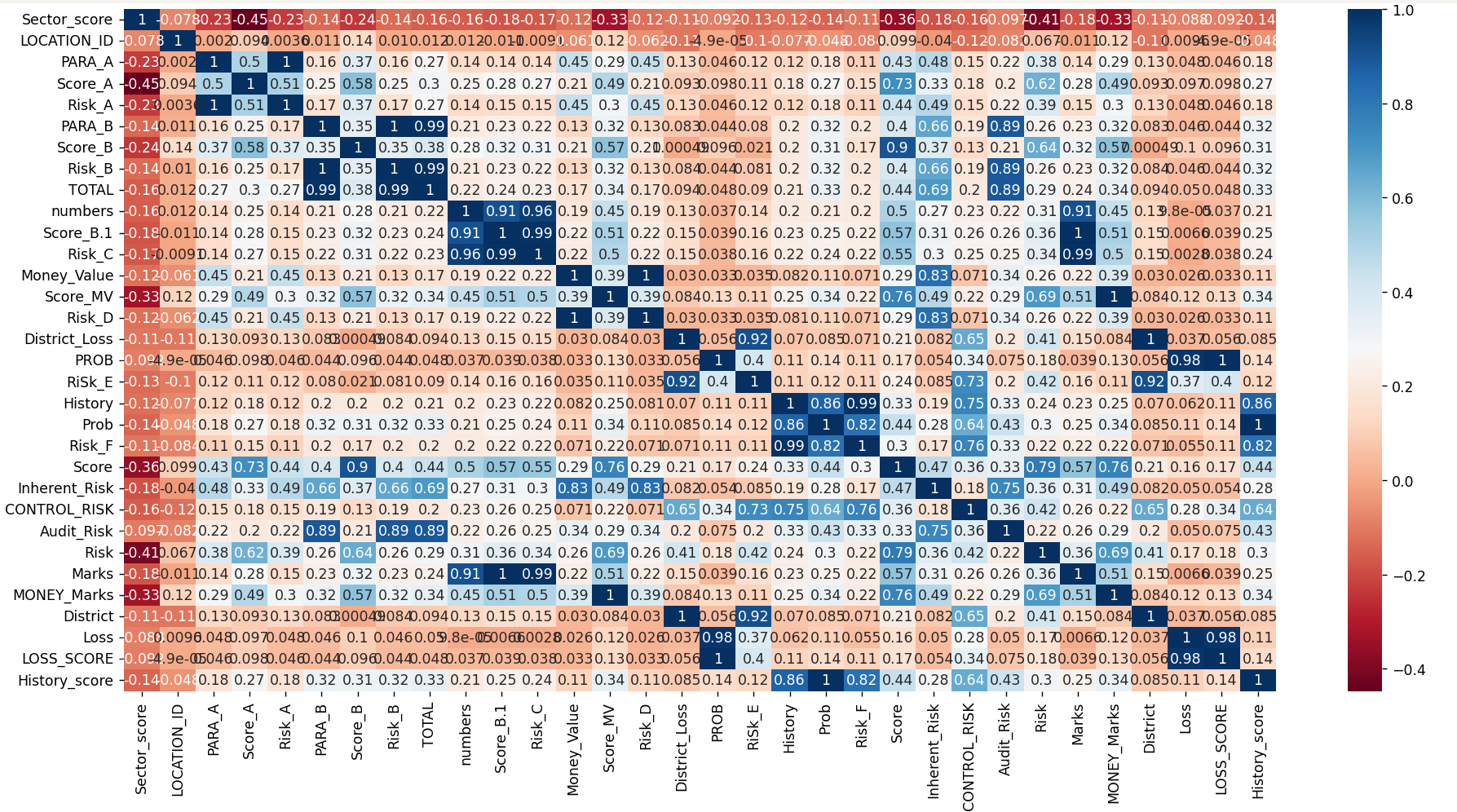


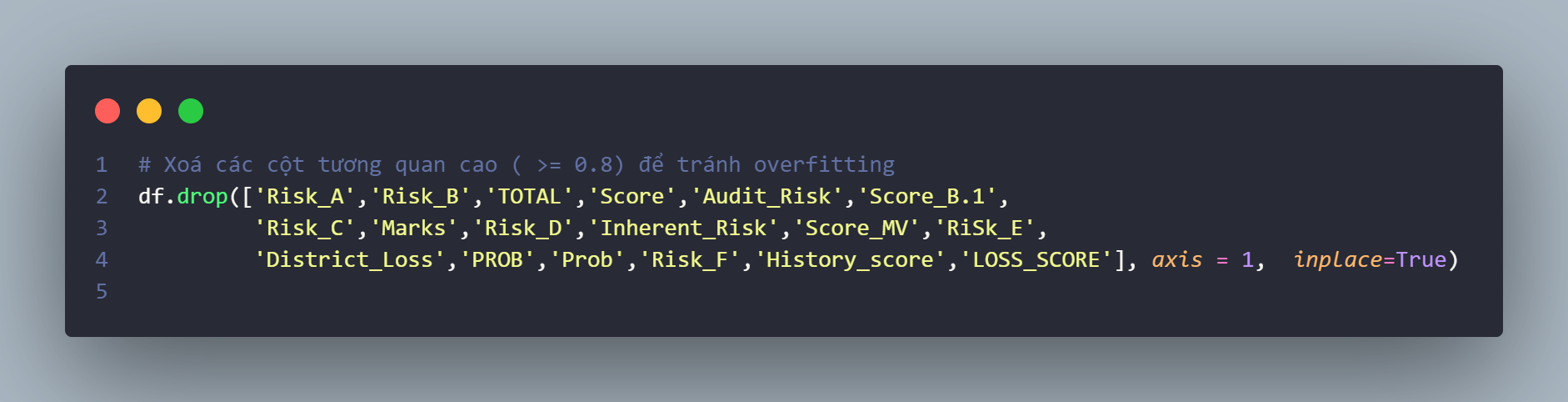
Chuyển đổi kiểu của LOCATION\_ID về dạng float cho phù hợp với các trường dữ liệu khác.



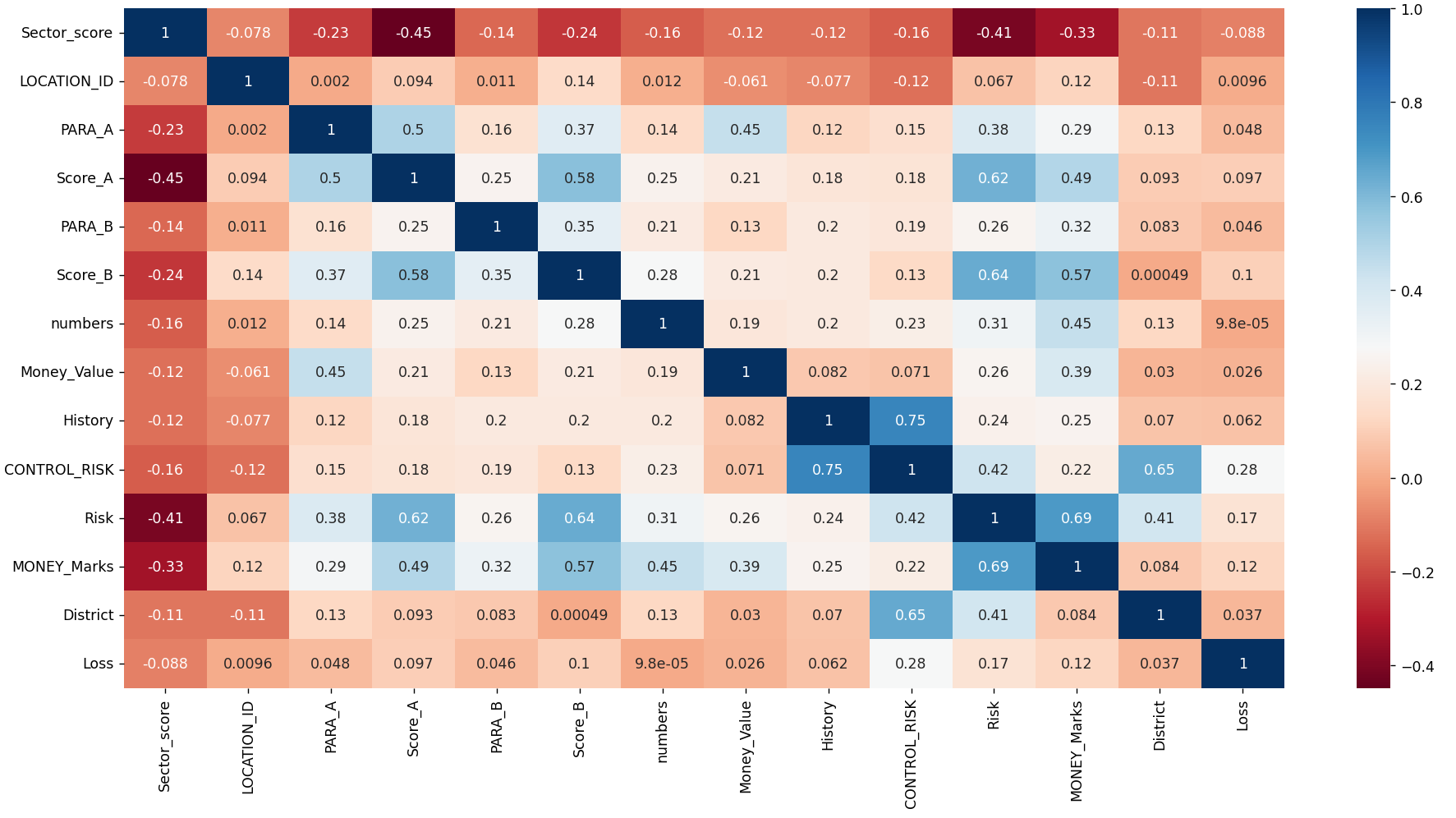
Cần loại bỏ các trường dữ liệu có giá trị tương quan cao với nhau, nghĩa là xóa các trường giá trị có thể dự đoán được từ giá trị của trường giá trị khác. Ta xây dựng ma trận thể hiện mức độ tương quan >= 0.8 (các ô màu xanh).







Kết quả:



Sau khi xem xét dữ liệu, ta thấy cột PARA\_A và PARA\_B sẽ suy ra được Score\_A và Score\_B với nguyên tắc

PARA\_A < 1 => Score\_A = 2

PARA\_A >= 1 và <= 2 => Score\_A = 4

PARA\_A > 2 => Score\_A = 6

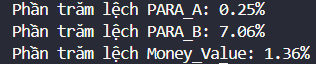
Và cột Money\_Value sẽ suy ra được MONEY\_Marks với nguyên tắc:

Money\_Value có giá trị < 6 thì Money\_Marks = 2

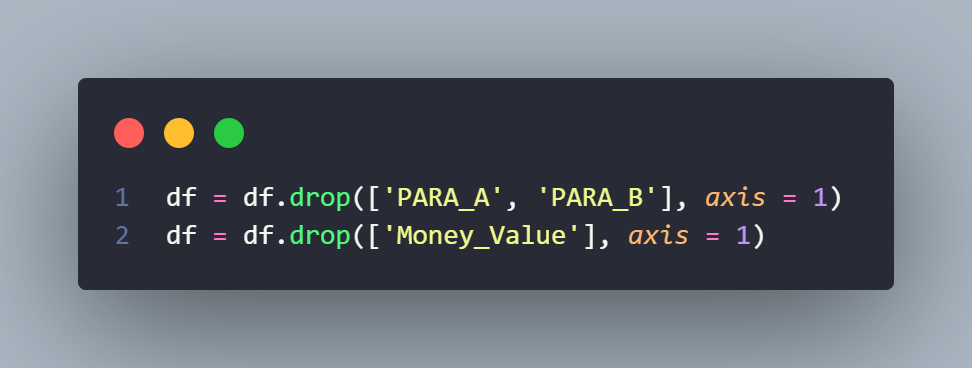
Money\_Value có giá trị >= 6 và < 10 thì Money\_Marks = 4

Money\_Value có giá trị >= 10 thì Money\_Marks = 6

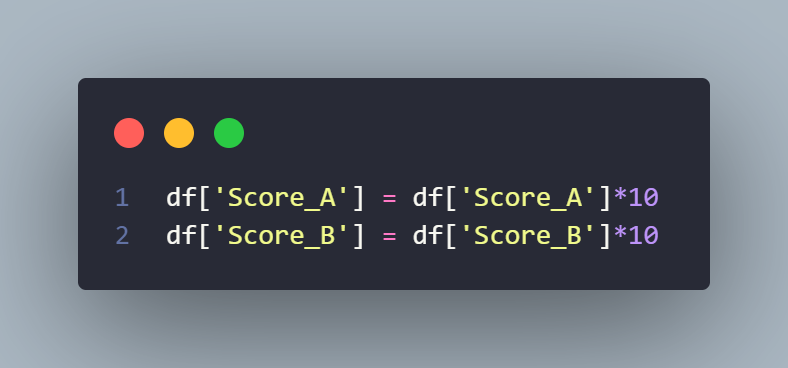
Kiểm tra phần trăm sai lệch của cá quyên tắc đó:



Ta thấy 2 nguyên tắc trên đúng lên tới hơn 90% nên ta xóa 3 cột PARA\_A, PARA\_B và Money\_Value để tránh dư thừa dữ liệu:

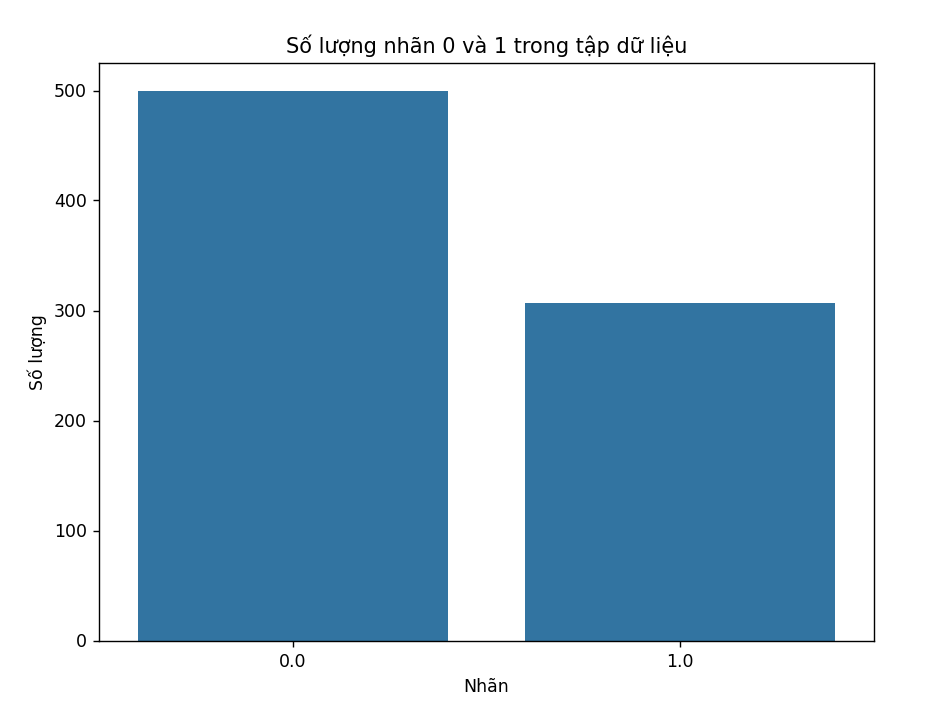


Chuẩn hóa miền giá trị lại cho 2 cột Score\_A, Score\_B:



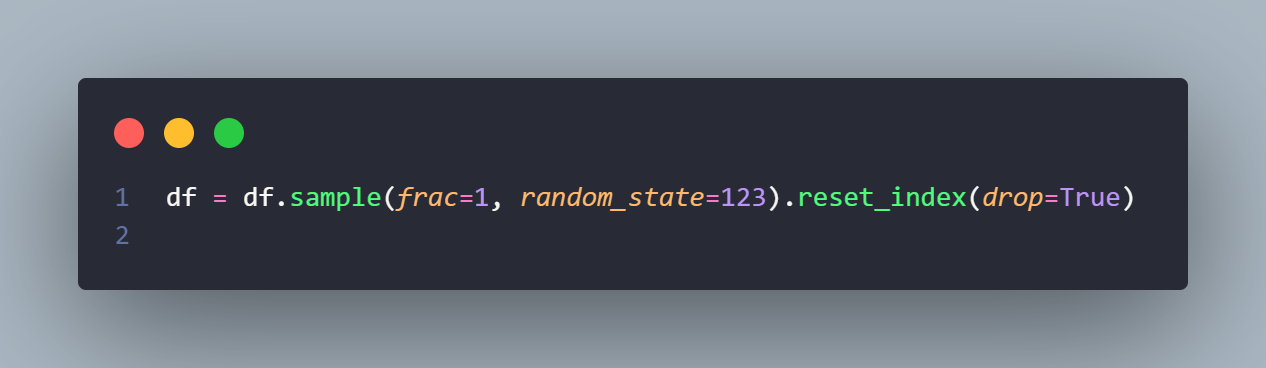
Kiểm tra nhãn của Tập dữ liệucó lệch không.





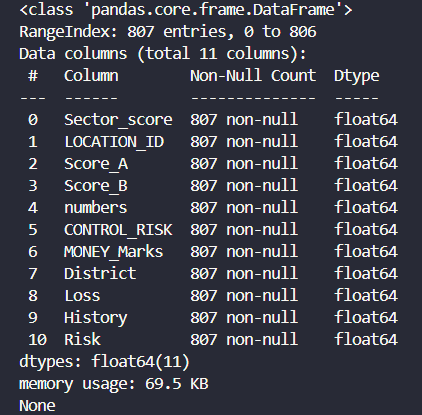
Số lượng nhãn 0 và 1 theo tỉ lệ khoảng . Tỉ lệ này vẫn tương đối cân bằng, có thể dùng xây dựng mô hình dự đoán nên không cần phải xử lý cân bằng nhãn.

Xáo trộn dữ liệu trước khi lưu lại tập dữ liệu.



Lưu dữ liệu đã xử lý vào data.csv thành tập dữ liệu dùng để xây dựng các mô hình huấn luyện khác nhau (KNN, Bayes, Decision Tree). Index = False có tác dụng loại bỏ cột số thứ tự khỏi dữ liệu.

Tập dữ liệu sau khi tiền xử lý:



### Lựa chọn mô hình:

Mô hình phù hợp với dữ liệu phân lớp: K-Nearest Neighbors, Gaussian Naive Bayes, Decision Trees.

#### K-Nearest Neighbors:

**Khái niệm:** K-Nearest Neighbors (KNN) là một thuật toán máy học thuộc nhóm học có giám sát, hoạt động dựa trên nguyên tắc bỏ phiếu đa số từ k láng giềng gần nhất trong không gian đặc trưng để dự đoán nhãn của một điểm dữ liệu mới.

**Nguyên tắc hoạt động:**

Khi một điểm dữ liệu mới xuất hiện, thuật toán sẽ:

1. Tính toán khoảng cách từ điểm dữ liệu đó đến tất cả các điểm dữ liệu đã tồn tại trong tập huấn luyện.
2. Xác định k điểm gần nhất dựa trên khoảng cách đã tính.
3. Đếm số lần xuất hiện của từng nhãn trong k điểm lân cận.
4. Gán nhãn cho điểm dữ liệu mới theo nhãn có tần suất xuất hiện cao nhất.

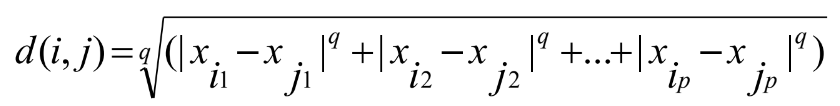
**Các tham số quan trọng:**

**n\_neighbors (k):** Số lượng láng giềng gần nhất được sử dụng để xác định nhãn. Giá trị k ảnh hưởng trực tiếp đến độ chính xác của mô hình, trong đó:

* k nhỏ có thể dẫn đến overfitting do mô hình quá nhạy với dữ liệu huấn luyện.
* k lớn có thể dẫn đến underfitting do mô hình bỏ qua chi tiết quan trọng.

**q:** Tham số xác định loại khoảng cách sử dụng để tính độ gần giữa các điểm dữ liệu. Các giá trị phổ biến:

* q = 1: Sử dụng khoảng cách Manhattan.
* q = 2: Sử dụng khoảng cách Euclidean.



**Ưu điểm**

* Đơn giản, dễ hiểu và dễ triển khai.
* Không yêu cầu quá trình huấn luyện phức tạp, chỉ cần lưu trữ dữ liệu huấn luyện.
* Hoạt động hiệu quả với tập dữ liệu có phân bố rõ ràng.

**Nhược điểm**

* Là thuật toán **lười học,** không xây dựng mô hình trước mà chỉ thực hiện tính toán khi dự đoán, dẫn đến thời gian dự đoán chậm với dữ liệu lớn.
* Nhạy cảm với dữ liệu nhiễu và ngoại lệ, ảnh hưởng đến độ chính xác.
* Độ phức tạp tính toán cao khi làm việc với tập dữ liệu lớn do phải tính khoảng cách đến toàn bộ điểm dữ liệu khác.

#### Gaussian Naïve Bayes:

**Khái niệm:** Gaussian Naïve Bayes (GNB) là một thuật toán máy học dựa trên xác suất Bayes, với giả định rằng các biến đầu vào là độc lập với nhau. GNB thường được sử dụng cho các bài toán phân loại và đôi khi có thể áp dụng cho hồi quy.

**Nguyên tắc hoạt động:**

* Tính toán xác suất có điều kiện của từng đặc trưng đầu vào dựa trên phân phối Gaussian (phân phối chuẩn).
* Áp dụng định lý Bayes để tính xác suất của từng nhãn đầu ra.
* Chọn nhãn có xác suất cao nhất làm kết quả dự đoán.

**Ưu điểm**

* Không phụ thuộc vào các tham số như số lượng láng giềng (k) hay độ sâu tối đa của cây quyết định, giúp mô hình ổn định hơn.
* Tốc độ dự đoán nhanh, phù hợp với các bài toán yêu cầu thời gian thực.
* Hoạt động tốt ngay cả với tập dữ liệu có kích thước nhỏ.

**Nhược điểm**

* Nhạy cảm với dữ liệu nhiễu, có thể bị ảnh hưởng nếu dữ liệu không tuân theo phân phối chuẩn.
* Hiệu suất kém trong trường hợp các biến đầu vào có sự phụ thuộc mạnh lẫn nhau, vì mô hình giả định chúng là độc lập.

#### Decision Trees:

**Khái niệm:** Cây quyết định (Decision Trees) là một thuật toán học máy có giám sát, được sử dụng cho cả bài toán phân loại và hồi quy. Phương pháp này hoạt động bằng cách chia dữ liệu thành các tập con theo từng điều kiện nhất định, tạo thành một cấu trúc cây giúp đưa ra quyết định dựa trên dữ liệu đầu vào.

**Nguyên tắc hoạt động:**

* Thuật toán chọn một thuộc tính làm điểm phân chia dựa trên một chỉ số đánh giá như **Gini impurity** hoặc **Entropy**.
* Tập dữ liệu được chia thành các nhánh con dựa trên giá trị của thuộc tính đó.
* Quá trình chia tiếp tục diễn ra đệ quy cho đến khi không thể phân nhánh thêm (ví dụ: khi đạt đến số lượng mẫu tối thiểu hoặc max depth được thiết lập).
* Kết quả cuối cùng là một cây quyết định, trong đó mỗi nút lá đại diện cho một nhãn đầu ra hoặc giá trị dự đoán.

**Ưu điểm**

* Hiệu quả cao trong nhiều bài toán phân loại và hồi quy.
* Dễ hiểu, trực quan, có thể diễn giải bằng quy tắc logic rõ ràng.
* Xây dựng và dự đoán nhanh, không yêu cầu quá nhiều dữ liệu huấn luyện.

**Nhược điểm**

* Dễ bị **overfitting**, đặc biệt khi cây có quá nhiều nhánh.
* Độ chính xác phụ thuộc vào tham số **max depth** do người dùng thiết lập, có thể dẫn đến mô hình quá phức tạp hoặc quá đơn giản.

## Cấu hình máy tính

|  |  |
| --- | --- |
| **Sinh Viên** | **Cấu hình máy tính** |
| Lâm Bá Thành – B2207564 | CPU: Intel® Core™ i5-10300H  RAM: 8 Gb  SSD: 512 Gb |
| Nguyễn Phước Phát – B2207554 | CPU: Intel® Core™ i5-12500H  RAM: 24 Gb  SSD: 512 Gb |
| Phương Thành Đạt – B2207516 | CPU: Intel® Core™ i5-1135G7H  RAM: 8 Gb  SSD: 512 Gb |

## Huấn luyện mô hình:

### Đọc dữ liệu:

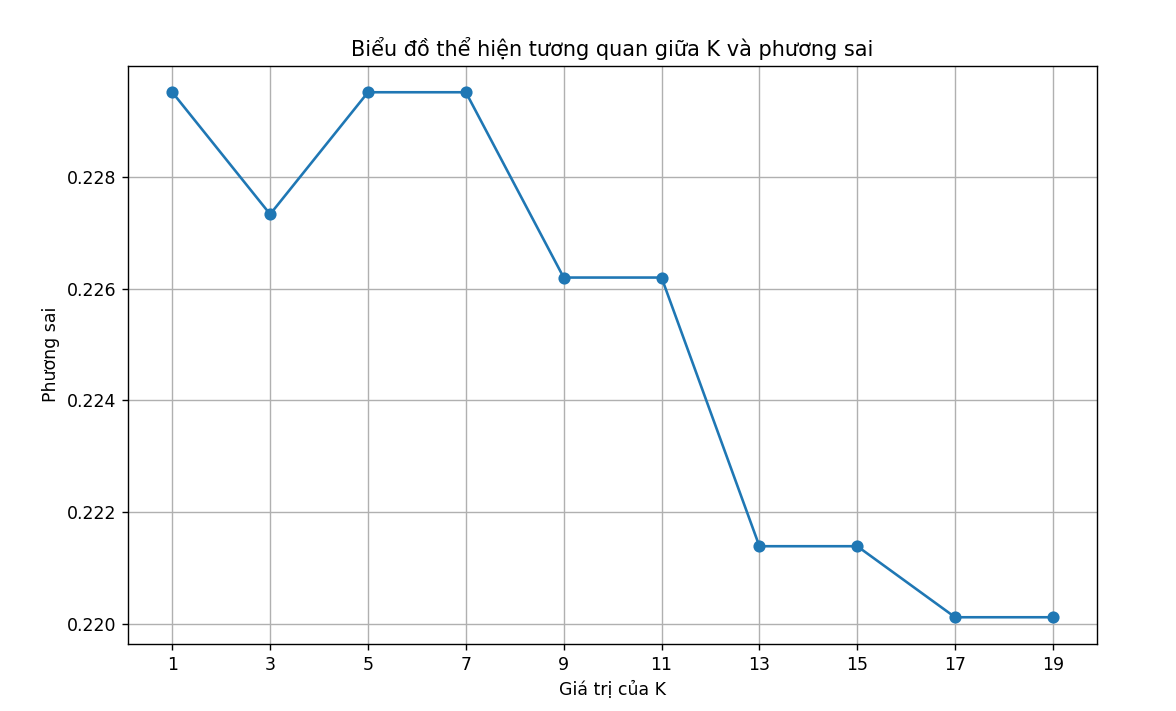
Đọc dữ liệu từ file data.csv đã được xử lý bên trên. Lưu các đặc trưng vào biến X và nhãn vào biến y.



### Chọn số lượng k neighbors phù hợp:

Thử nghiệm giá trị k từ 1 đến 21 để tìm ra k tối ưu, sao cho không quá nhỏ để tránh overfitting, nhưng cũng không quá lớn để làm giảm đáng kể độ chính xác mà không mang lại lợi ích đáng kể trong việc giảm phương sai.



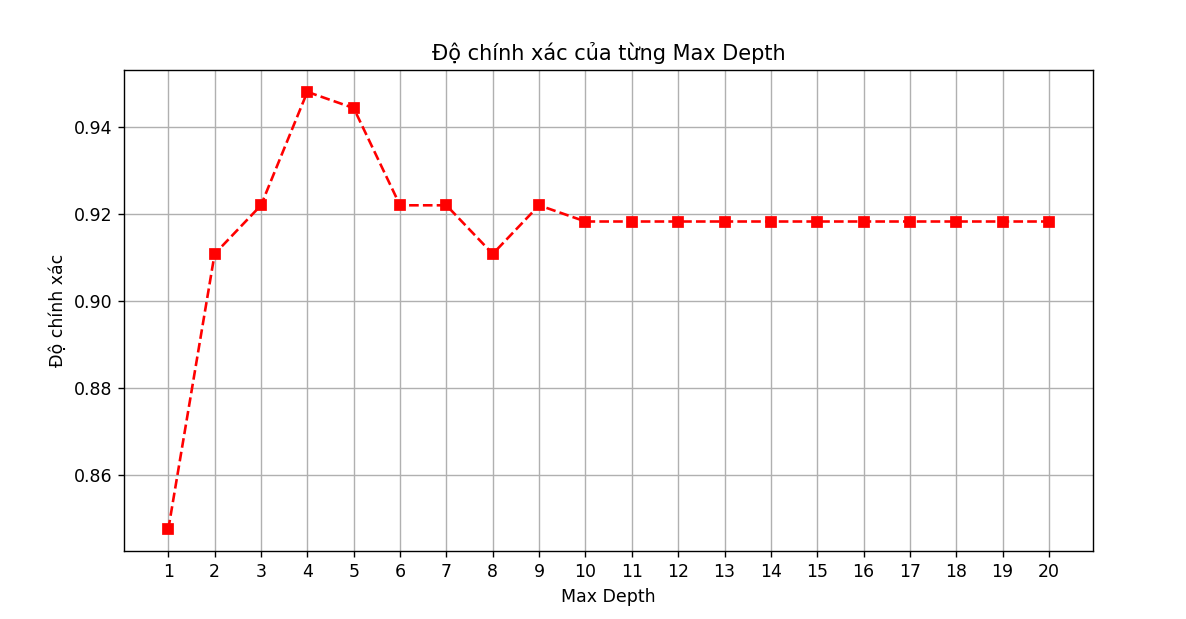


Chọn k = 13

### Chọn max depth và min samples leaf phù hợp:

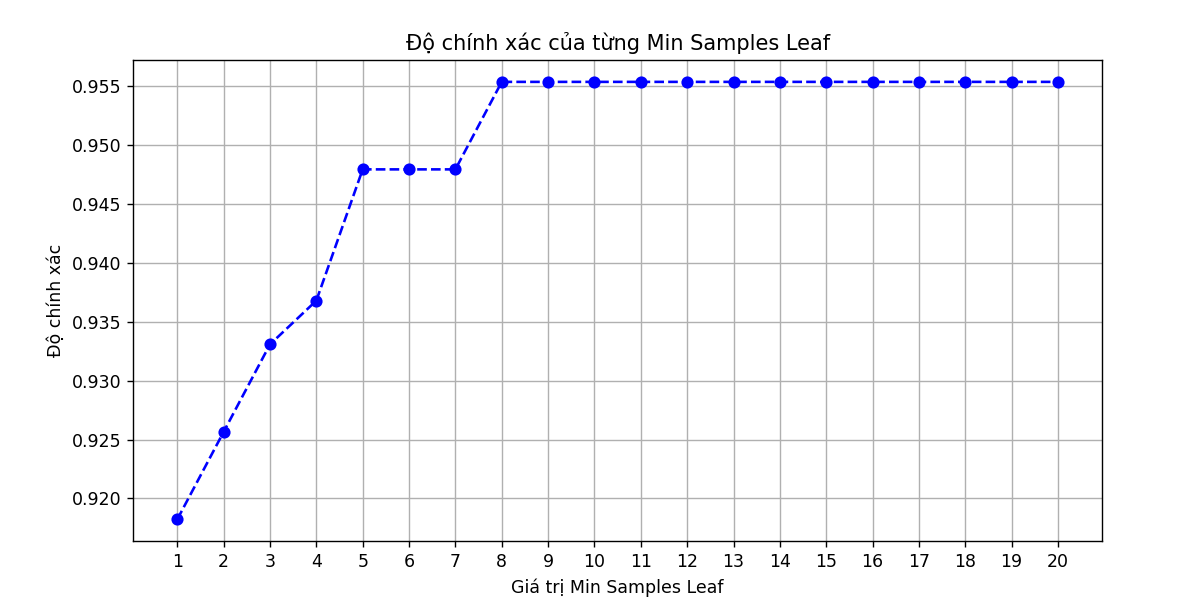
Max depth:





Chọn max\_depth = 9

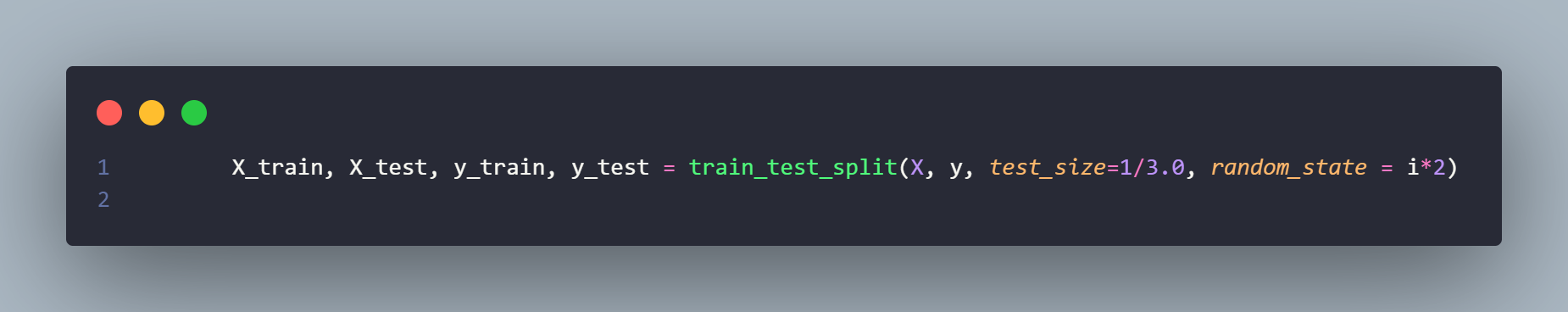




Chọn min samples leaf = 8

### Chia dữ liệu:

Sử dụng 2 phần dữ liệu để huấn luyện và 1 phần dữ liệu để kiểm tra.

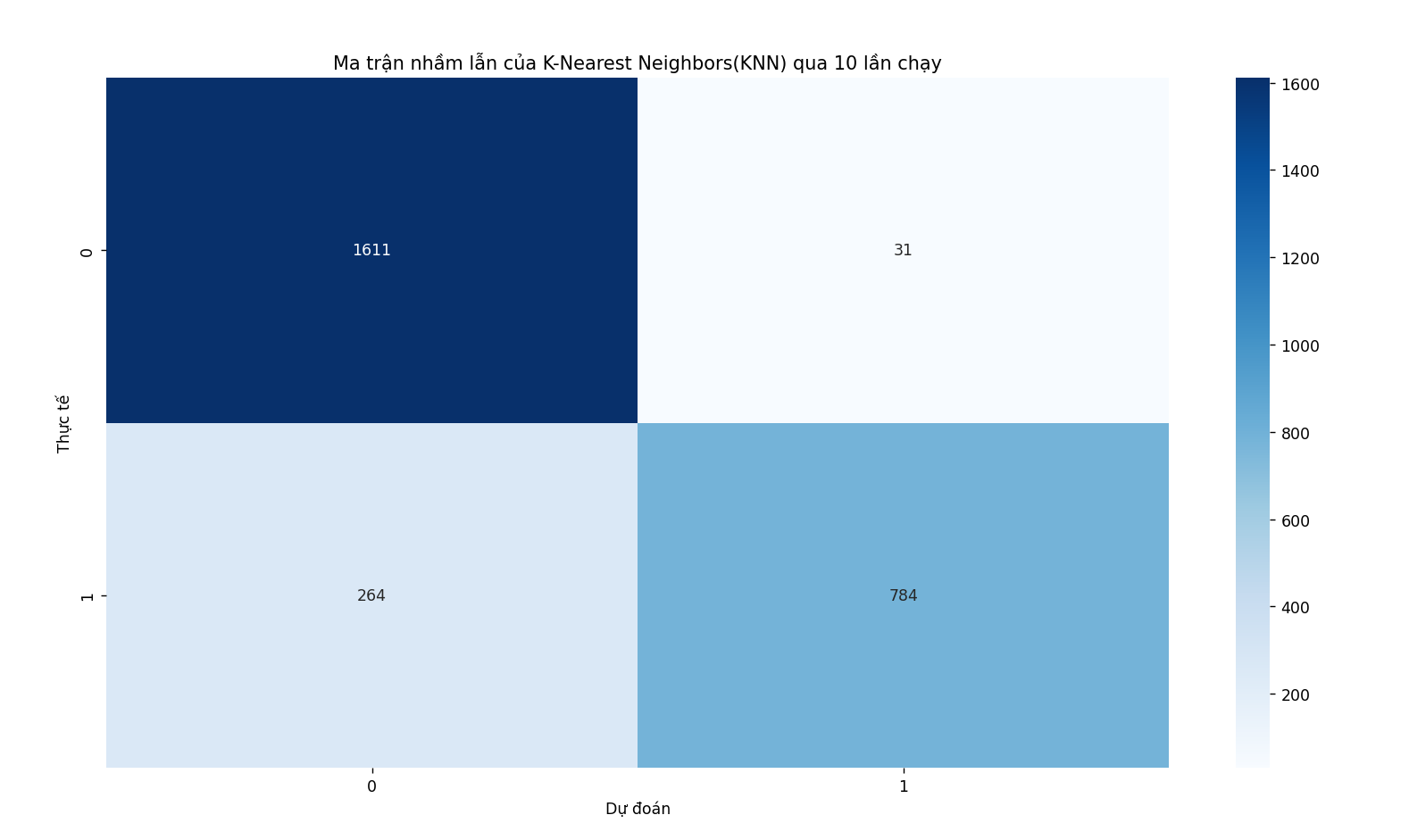


## Đánh giá mô hình:

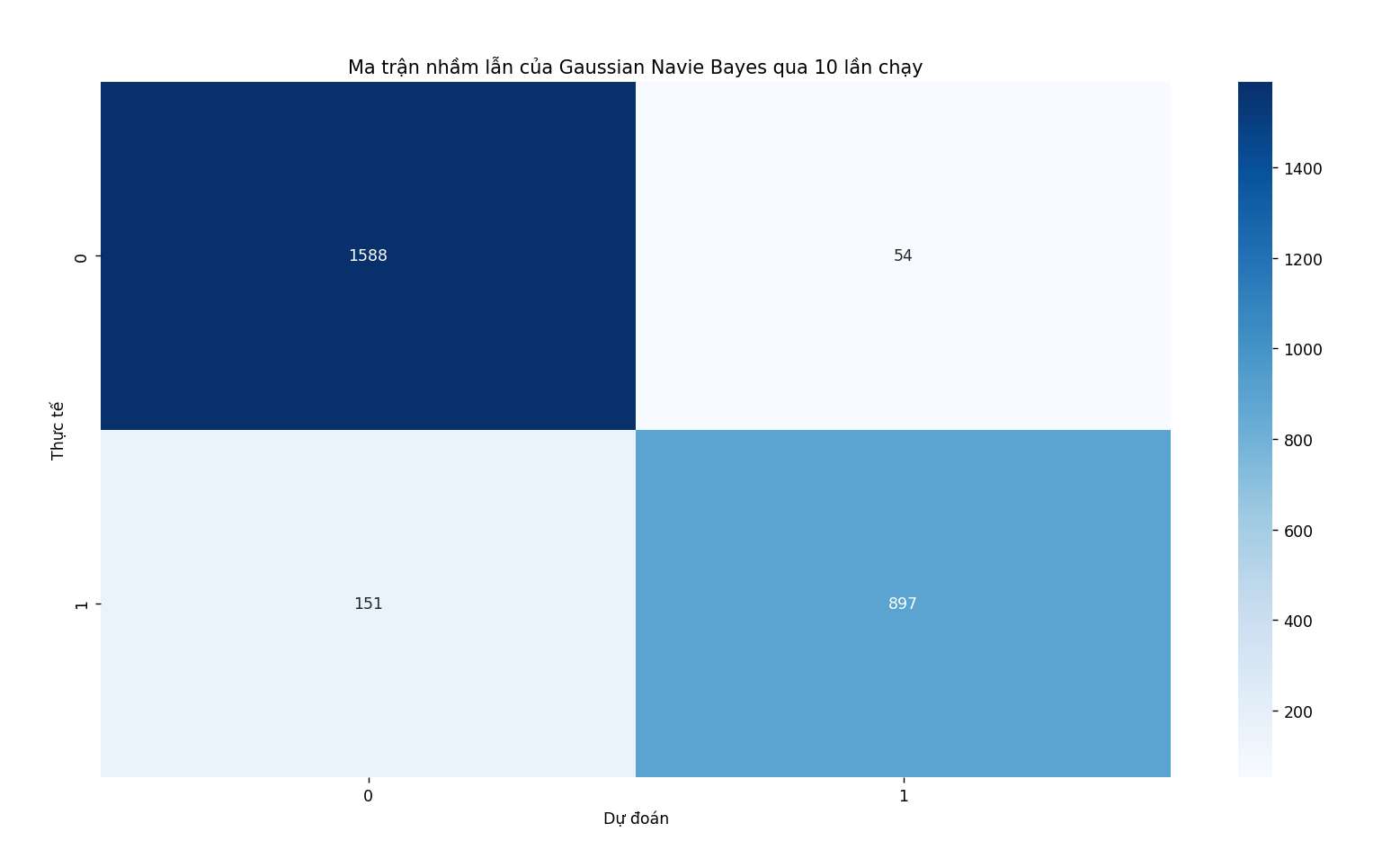
### Đánh giá mô hình phân lớp / Regression / Cluster:

**Ma trận nhầm lẫn:**

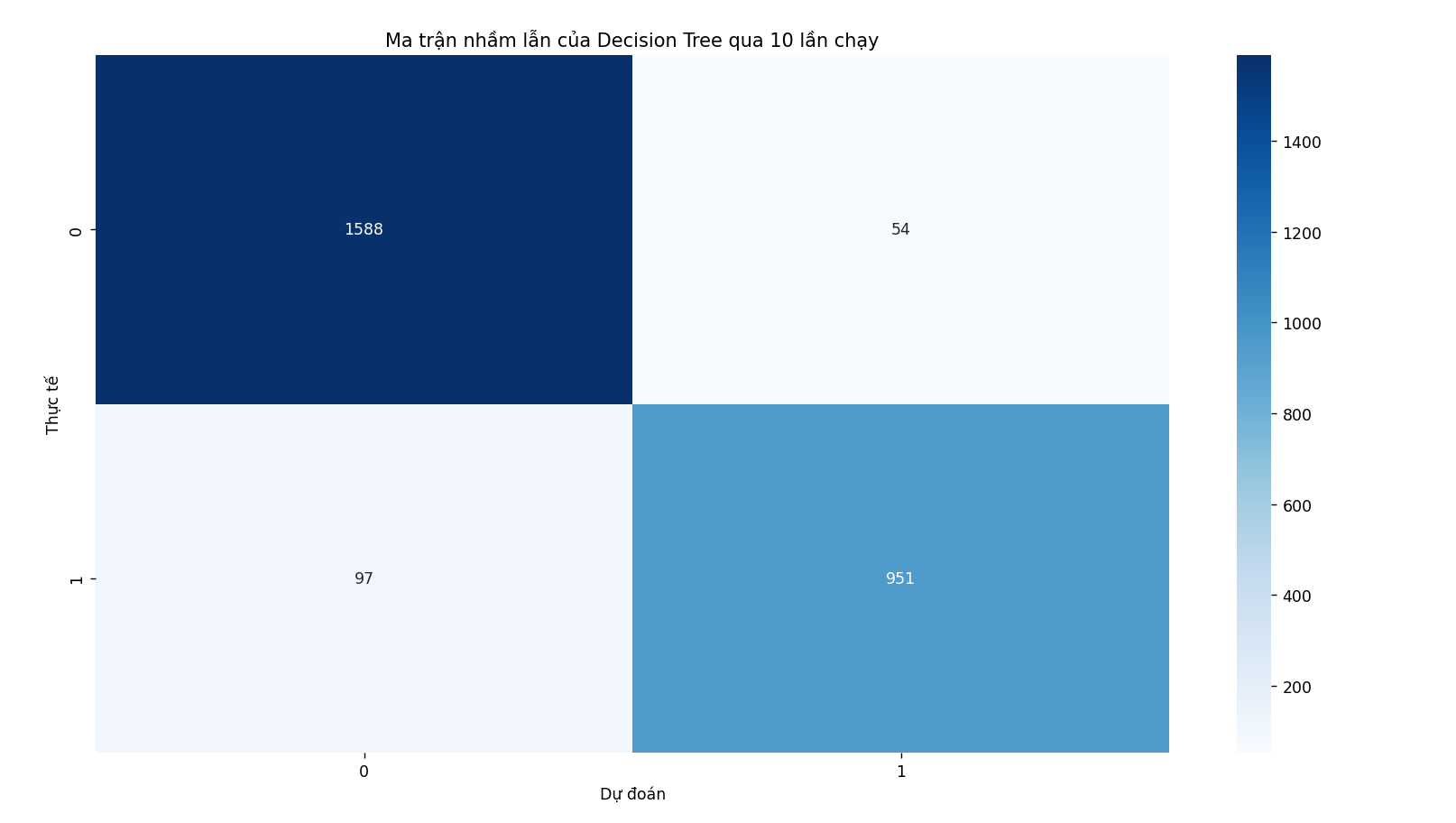
**K-Nearest Neighbors**

****

**Gaussian Naive Bayes**

****

**Decision Tree**

****

**Kết quả của các chỉ số:**

**K-Nearest Neighbors:**

**Với k = 13 như đã tìm được bên trên**

**và p = 2 (Sử dụng khoảng cách Euclide)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Lần chạy** | **Accuracy**  **(%)** | **Precision**  **(%)** | **Recall**  **(%)** | **F1**  **(%)** |
| **1** | **95.54** | **100.00** | **88.12** | **93.68** |
| **2** | **95.17** | **99.00** | **89.19** | **93.84** |
| **3** | **93.68** | **98.97** | **85.71** | **91.87** |
| **4** | **93.68** | **98.78** | **83.50** | **90.50** |
| **5** | **93.3** | **99.00** | **85.34** | **91.67** |
| **6** | **96.28** | **98.94** | **91.18** | **94.90** |
| **7** | **95.53** | **97.85** | **90.10** | **93.81** |
| **8** | **94.80** | **97.92** | **88.68** | **93.07** |
| **9** | **95.54** | **100.00** | **88.24** | **93.75** |
| **10** | **93.30** | **100.00** | **82.00** | **90.11** |
| **AVG** | **94.68** | **99.05** | **87.21** | **92.72** |

**Gaussian Navie Bayes**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Lần chạy** | **Accuracy (%)** | **Precision (%)** | **Recall**  **(%)** | **F1(%)** |
| **1** | **92.94** | **90.20** | **91.09** | **90.64** |
| **2** | **94.42** | **96.15** | **90.09** | **93.02** |
| **3** | **91.07** | **96.80** | **81.25** | **88.35** |
| **4** | **91.82** | **89.47** | **87.62** | **88.54** |
| **5** | **92.94** | **94.50** | **88.79** | **91.56** |
| **6** | **91.45** | **89.90** | **87.25** | **88.56** |
| **7** | **93.68** | **93.75** | **89.11** | **91.37** |
| **8** | **92.93** | **97.80** | **83.96** | **90.36** |
| **9** | **91.45** | **93.41** | **83.33** | **88.08** |
| **10** | **91.82** | **95.35** | **82.00** | **88.17** |
| **AVG** | **92.45** | **93.73** | **86.45** | **89.86** |

**Decesion Tree:**

**Với max depth = 9 và min samples leaf = 8**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Lần chạy** | **Accuracy**  **(%)** | **Precision**  **(%)** | **Recall**  **(%)** | **F1**  **(%)** |
| **1** | **95.54** | **100.00** | **88.12** | **93.68** |
| **2** | **96.28** | **97.20** | **93.69** | **95.41** |
| **3** | **92.57** | **97.92** | **83.93** | **90.38** |
| **4** | **94.42** | **95.56** | **88.66** | **91.98** |
| **5** | **94.42** | **99.02** | **87.93** | **93.15** |
| **6** | **93.68** | **89.00** | **95.10** | **91.94** |
| **7** | **95.91** | **96.88** | **92.08** | **94.42** |
| **8** | **94.42** | **97.89** | **87.74** | **92.54** |
| **9** | **95.91** | **97.89** | **91.18** | **94.42** |
| **10** | **95.54** | **97.83** | **90.00** | **93.75** |
| **AVG** | **94.87** | **96.92** | **89.84** | **93.17** |

### Nhận xét kết quả thực nghiệm:

Các chỉ số được sử dụng để đánh giá hiệu suất mô hình bao gồm **Accuracy, Precision, Recall** và **F1 Score**. Mục tiêu của đề tài là xác định mô hình hoạt động hiệu quả nhất trên tập dữ liệu đã cho.

**Nhận xét:**

* **KNN:** Có độ chính xác (Accuracy) khá cao trong ba mô hình, trung bình đạt **94.68%**. Precision cũng rất cao (**99.05%**), nhưng Recall lại thấp nhất (**87.21%**). Điều này cho thấy mô hình có xu hướng ưu tiên dự đoán đúng các mẫu thuộc một lớp cụ thể, nhưng có thể bỏ sót một số trường hợp khác.
* **Naive Bayes:** Các chỉ số đánh giá ổn định, với Accuracy trung bình đạt **92.45%**, Precision **93.73%**, và Recall **86.45%**. Dù có độ chính xác thấp hơn KNN, sự chênh lệch giữa Precision và Recall không quá lớn, cho thấy mô hình hoạt động cân bằng hơn.
* **Decision Tree:** Đạt giá trị cao nhất ở hầu hết các chỉ số, với Accuracy **94.87%**, Precision **96.92%**, Recall **89.84%**, và F1 Score **93.17%**. Với sự cân bằng tốt giữa Precision và Recall, Decision Tree có hiệu suất tổng thể tối ưu nhất trong ba mô hình được so sánh.

**Kết luận:** Decision Tree không chỉ có độ chính xác cao **94.87%** mà còn lại mang đến sự cân bằng tốt hơn giữa các chỉ số đánh giá, giúp mô hình hoạt động hiệu quả và ổn định hơn trên tập dữ liệu.

# PHẦN KẾT LUẬN

## Kết quả đạt được:

* Việc áp dụng ba mô hình phân lớp (**KNN, Naive Bayes, Decision Tree**) trên tập dữ liệu **Audit Risk** cho thấy hiệu suất tương đối tốt.
* Quá trình thực hiện đồ án đã giúp nhóm rèn luyện kỹ năng phân tích dữ liệu, từ **tiền xử lý dữ liệu** đến **xây dựng và đánh giá mô hình**.
* Nhóm đã áp dụng các phương pháp đánh giá hiệu suất mô hình như **ma trận nhầm lẫn (Confusion Matrix), độ chính xác (Accuracy), Precision, Recall** và **F1 Score**, giúp hiểu rõ hơn về ý nghĩa và vai trò của các chỉ số này trong thực tế.

## Hướng phát triển:

* **Mở rộng mô hình:** Thử nghiệm thêm các thuật toán khác như **Support Vector Machines (SVM), Neural Network, Random Forest**... để tìm ra mô hình có hiệu suất tối ưu hơn.
* **Triển khai thực tế:** Tích hợp mô hình vào **API web**, cải thiện **giao diện** và **tối ưu hiệu suất server** nhằm nâng cao trải nghiệm người dùng.
* **Mở rộng tập dữ liệu:** Thu thập dữ liệu thực tế theo thời gian, bổ sung vào tập dữ liệu hiện tại hoặc xây dựng cơ chế dự đoán dựa trên dữ liệu người dùng nhập.

# TÀI LIỆU THAM KHẢO

[1]. Slide máy học ứng dụng (PGS. TS. Đỗ Thanh Nghị - TS. Trần Nguyễn Minh Thư).

[2]. [Audit Data - UCI Machine Learning Repository](https://archive.ics.uci.edu/dataset/475/audit+data)

[3]. Abhishek Girish Patil, Audit Risk Classification, GitHub. Truy cập tại: [https://github.com/abhilampard/Audit-Risk-Classification/blob/master/AuditRisk\_ Classification.ipynb](https://github.com/abhilampard/Audit-Risk-Classification/blob/master/AuditRisk_%20Classification.ipynb).

[4]. TS. Lưu Tiến Đạo, Hướng dẫn xây dựng trang web dự đoán loại hoa Iris sử dụng Flask và Decision Tree, GitHub. Truy cập tại: https://github.com/ltdaovn/dataset/blob/master/docs/irisApp.md