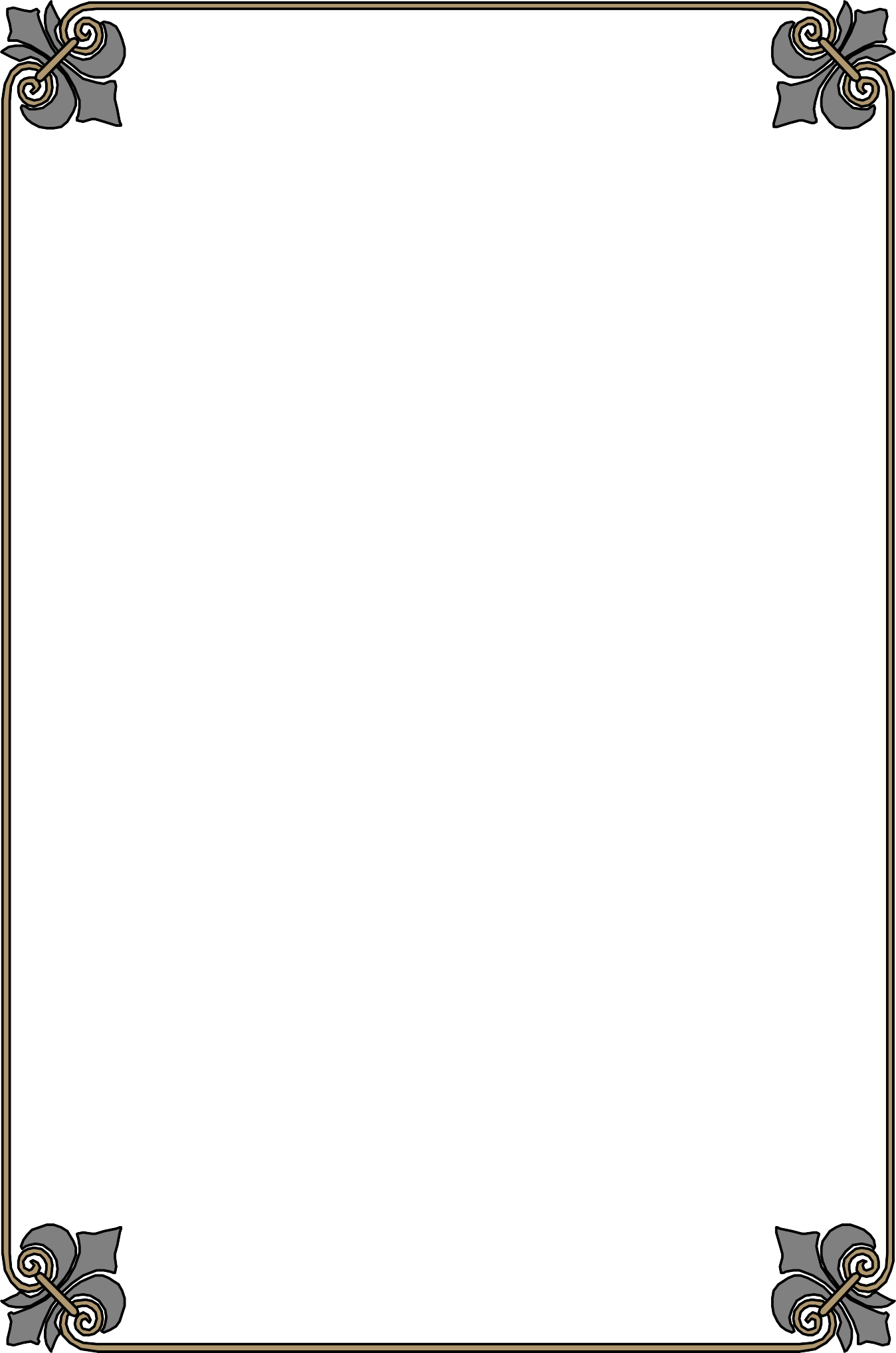
**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**



**KHOA CÔNG NGHỆ THÔNG TIN**

**BỘ MÔN HỆ THỐNG THÔNG TIN**

-----🙞🙞🙞🙞🙞-----

**BÁO CÁO CUỐI KỲ MÔN KHAI PHÁ DỮ LIỆU**

**TÊN ĐỀ TÀI:**

**ỨNG DỤNG CÁC KỸ THUẬT KHAI PHÁ DỮ LIỆU**

**ĐỂ DỰ ĐOÁN THU NHẬP CỦA NGƯỜI TRƯỞNG THÀNH**

**GVHD:** *ThS. Trần Trọng Bình*

**Lớp HP: DAMI330484\_23\_2\_01**

**Nhóm thực hiện:** *Nhóm 04*

**Học kỳ: 2**

**Năm học:** *2023 - 2024*

*Thành phố Hồ Chí Minh, tháng 5, năm 2024*

**LỜI CẢM ƠN**

Em xin gửi lời cảm ơn chân thành và sự tri ân sâu sắc đối với các thầy cô của Trường Đại học Sư phạm Kỹ thuật TP Hồ Chí Minh, đặc biệt là quý thầy cô bộ môn Hệ thống Thông tin của trường đã giúp cho chúng em trang bị các kiến thức cơ bản, các kỹ năng thực tế và tạo điều kiện để chúng em có thể hoàn thành đồ án môn học của mình.

Đặc biệt chúng em xin chân thành cảm ơn thầy Trần Trọng Bình đã nhiệt tình hướng dẫn, quan tâm truyền đạt những kiến thức và kinh nghiệm, trực tiếp hướng dẫn tận tình, sửa chữa và đóng góp ý kiến quý báu cho chúng em trong suốt thời gian học tập để chúng em có thể hoàn thành tốt môn học này.

Trong thời gian một học kỳ thực hiện đề tài, nhóm chúng em đã vận dụng những  
kiến thức nền tảng đã tích lũy đồng thời kết hợp với việc học hỏi và nghiên cứu những  
kiến thức mới. Từ đó, nhóm chúng em vận dụng tối đa những gì đã thu thập được để hoàn  
thành một báo cáo môn học tốt nhất. Tuy nhiên, trong quá trình thực hiện, nhóm chúng em không tránh khỏi những thiếu sót. Chính vì vậy, nhóm chúng em rất mong nhận được  
những sự góp ý từ phía Thầy/Cô nhằm hoàn thiện những kiến thức mà nhóm chúng em đã học tập và là hành trang để nhóm tác giả thực hiện tiếp các đề tài khác trong tương lai.

Nhóm chúng em xin chân thành cảm ơn!

***Nhóm thực hiện***

|  |  |
| --- | --- |
| BỘ GIÁO DỤC VÀ ĐÀO TẠO  **TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT**  **THÀNH PHỐ HỒ CHÍ MINH** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc lập – Tự do – Hạnh phúc** |

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

**Họ và tên sinh viên thực hiện**

* Ngô Thanh Thanh - 21110643
* Vũ Thị Bích Ngọc – 21110905
* Nguyễn Quốc Thịnh - 21110661
* Kiều Anh Thiên - 21110746

**Môn học:** Khai phá dữ liệu

**Nhận xét**

Tp HCM, / / 2024

Giảng viên hướng dẫn

(Tên và chữ ký)

Trần Trọng Bình

BẢNG PHÂN CÔNG CÔNG VIỆC

|  |  |  |
| --- | --- | --- |
| **Nội dung phân công** | **Sinh viên thực hiện** | **Mức độ hoàn thành** |
| **TÌM VÀ XỬ LÝ DỮ LIỆU** | | |
| Tìm tập dữ liệu | Vũ Thị Bích Ngọc | 100% |
| Tiền xử lý dữ liệu | Nguyễn Quốc Thịnh | 100% |
| Khám phá dữ liệu | Ngô Thanh Thanh | 100% |
| **CÀI ĐẶT CÁC KỲ THUẬT KHAI PHÁ DỮ LIỆU** | | |
| Decision tree | Vũ Thị Bích Ngọc | 100% |
| Random forest | Ngô Thanh Thanh | 100% |
| KNN | Ngô Thanh Thanh | 100% |
| Support Vector Machines | Vũ Thị Bích Ngọc | 100% |
| Logistic Regression | Nguyễn Quốc Thịnh | 100% |
| Navie Bayes | Kiều Anh Thiên | 100% |
| Feedforward Neural Network | Nguyễn Quốc Thịnh | 100% |
| **PHÂN TÍCH KẾT QUẢ - TRẢ LỜI CÂU HỎI** | | |
| Phân tích hiệu quả của các kỹ thuật | Nguyễn Quốc Thịnh | 100% |
| Đánh giá, nhận xét, trả lời câu hỏi | Tất cả | 100% |
| Viết báo cáo | Tất cả | 100% |

ĐÁNH GIÁ THÀNH VIÊN NHÓM

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tiêu chí đánh giá** | Phân công & đóng góp (30) | Kỹ năng làm việc nhóm (20) | Tính kịp thời (20) | Chất lượng công việc (30) | **Tổng quan (100)** |
| **Ngô Thanh Thanh**  **21110643** | 25 | 20 | 15 | 25 | **85** |
| **Vũ Thị Bích Ngọc 21110905** | 25 | 15 | 20 | 25 | **85** |
| **Nguyễn Quốc Thịnh**  **21110661** | 25 | 20 | 15 | 25 | **85** |
| **Kiều Anh Thiên 21110746** | 25 | 15 | 15 | 25 | **80** |

MỤC LỤC

[**PHẦN 1. PHÂN TÍCH DỮ LIỆU** 8](#_Toc167402992)

[**1. Phát biểu về dữ liệu** 8](#_Toc167402993)

[**1.1 Mô tả về dữ liệu** 8](#_Toc167402994)

[**1.2 Các trường của dữ liệu** 8](#_Toc167402995)

[**1.3 Đặc điểm của dữ liệu** 11](#_Toc167402996)

[**2. Ứng dụng tập dữ liệu** 17](#_Toc167402997)

[**PHẦN 2: PHÂN TÍCH CÁC KỸ THUẬT KHAI PHÁ ĐỐI VỚI TẬP DỮ LIỆU** 19](#_Toc167402998)

[**2.1 Decision tree** 19](#_Toc167402999)

[**2.1.1 Sơ lược về lý thuyết** 19](#_Toc167403000)

[**2.1.2 Ưu điểm** 20](#_Toc167403001)

[**2.1.3 Nhược điểm** 20](#_Toc167403002)

[**2.2 Random forest** 21](#_Toc167403003)

[**2.2.1 Sơ lược về lý thuyết** 21](#_Toc167403004)

[**2.2.2 Ưu điểm** 22](#_Toc167403005)

[**2.2.3 Nhược điểm** 22](#_Toc167403006)

[**2.3 KNN** 23](#_Toc167403007)

[**2.3.1 Sơ lược về lý thuyết** 23](#_Toc167403008)

[**2.3.2 Ưu điểm** 23](#_Toc167403009)

[**2.3.3 Nhược điểm** 24](#_Toc167403010)

[**2.4 Support Vector Machines** 24](#_Toc167403011)

[**2.4.1 Sơ lược về lý thuyết** 24](#_Toc167403012)

[**2.4.2 Ưu điểm** 25](#_Toc167403013)

[**2.5 Logistic Regression** 26](#_Toc167403014)

[**2.5.1 Sơ lược về lý thuyết** 26](#_Toc167403015)

[**2.5.2 Ưu điểm** 27](#_Toc167403016)

[**2.5.3 Nhược điểm** 27](#_Toc167403017)

[**2.6 Navie Bayes** 27](#_Toc167403018)

[**2.6.1 Sơ lược về lý thuyết** 27](#_Toc167403019)

[**2.6.2 Ưu điểm** 28](#_Toc167403020)

[**2.6.3 Nhược điểm** 28](#_Toc167403021)

[**2.7 Feedforward neural network** 28](#_Toc167403022)

[**2.7.1 Sơ lược về lý thuyết** 28](#_Toc167403023)

[**2.7.2 Ưu điểm** 29](#_Toc167403024)

[**2.7.3 Nhược điểm** 29](#_Toc167403025)

[**PHẦN 3: KẾT QUẢ KHAI PHÁ** 31](#_Toc167403026)

[**3.1 Đánh giá – nhận xét – trả lời câu hỏi** 31](#_Toc167403027)

[**3.2 Ý nghĩa** 33](#_Toc167403028)

[**PHẦN 4 : KẾT LUẬN** 35](#_Toc167403029)

[**4.1 Kết quả đạt được** 35](#_Toc167403030)

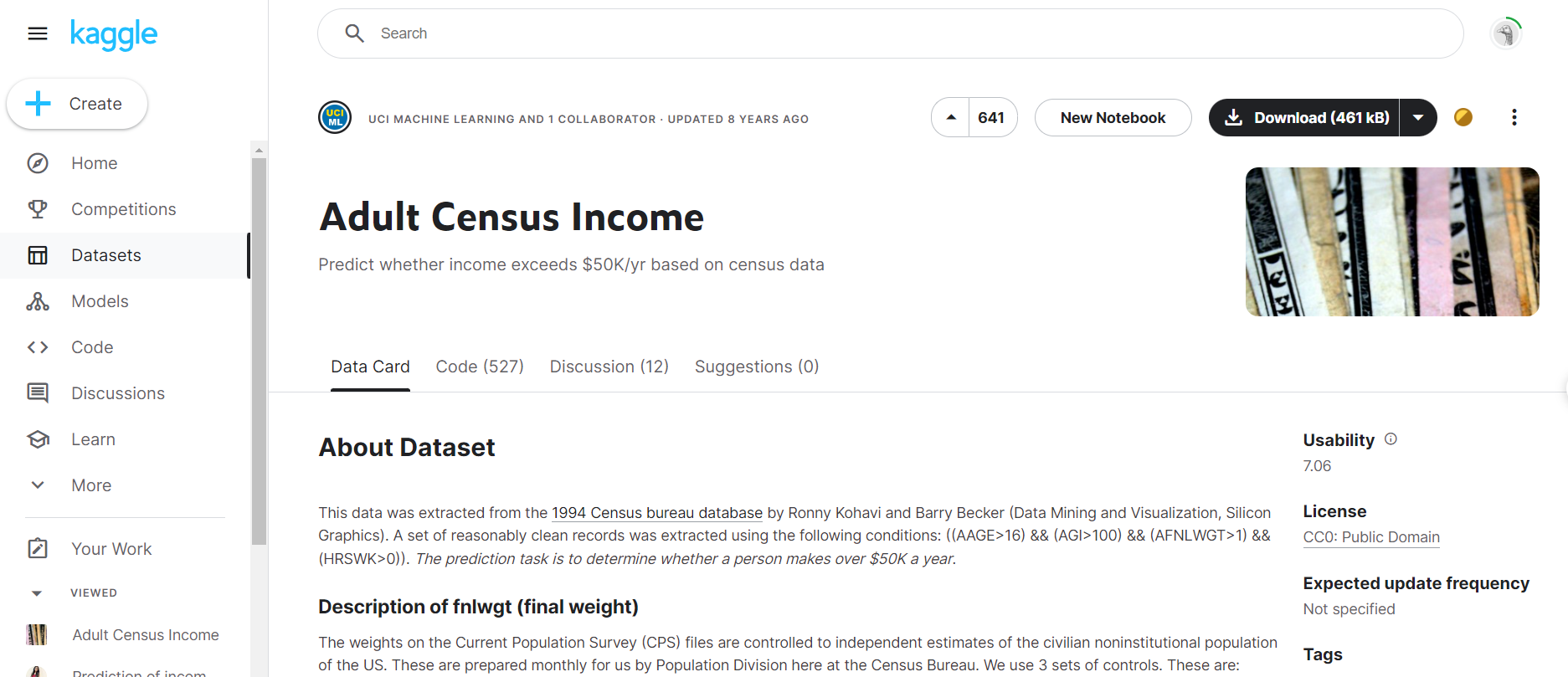
[**4.2 Những hạn chế** 35](#_Toc167403031)

[**4.3 Cách cải tiến** 36](#_Toc167403032)

*PHẦN 1. PHÂN TÍCH DỮ LIỆU*

**1. Phát biểu về dữ liệu**

### 1.1 Mô tả về dữ liệu

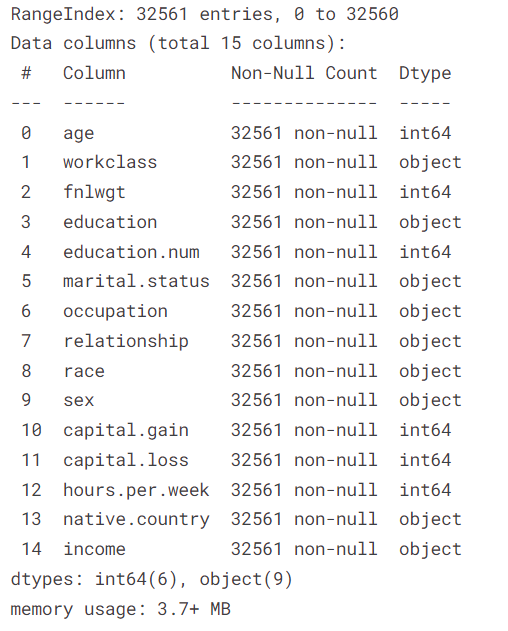


* Dữ liệu này được trích xuất từ ​​cơ sở dữ liệu của Cục Điều tra Dân số năm 1994 bởi Ronny Kohavi và Barry Becker (Data Mining and Visualization, Silicon Graphics)
* Bộ dữ liệu bao gồm một danh sách các bản ghi, mỗi bản ghi giải thích các đặc điểm khác nhau của một người cùng với thu nhập mỗi năm của người đó.
* Dữ liệu bao gồm 15 cột và 32561 dòng
* Link dữ liệu gốc: <https://www.kaggle.com/datasets/uciml/adult-census-income>

### **1.2 Các trường của dữ liệu**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Ý nghĩa | Giá trị |
| 1 | Age | int64 | Tuổi của các cá nhân được khảo sát. |  |
| 2 | WorkClass | object | Loại hình công việc | -Private: Làm việc cho công ty tư nhân.  -Self-emp-not-inc" : Tự làm chủ nhưng không đăng ký thành lập công ty.  - "Self-emp-inc": Tự làm chủ và đã đăng ký thành lập công ty.  - "Federal-gov": Làm việc cho chính phủ liên bang.  -"Local-gov" : Làm việc cho chính quyền địa phương.  - "State-gov" : Làm việc cho chính quyền tiểu bang.  - ? : thiếu hoặc không xác |
| 3 | Fnlwgt | int64 | Số cuối cùng của mỗi cá nhân được khảo sát. |  |
| 4 | Education | object | Mức độ học vấn |  |
| 5 | Education.num | int64 | Một biến số thay thế cho cột "education" để biểu thị số năm học chính thức mà một người đã hoàn thành |  |
| 6 | Marital.status | object | Trạng thái hôn nhân | - "Married-civ-spouse": Kết hôn hợp pháp  - "Divorced": ly hôn  - "Never-married": chưa bao giờ kết hôn.  - "Separated": ly thân.  - "Widowed": góa  - "Married-spouse-absent": kết hôn nhưng vợ/chồng vắng.  - "Married-AF-spouse": kết hôn với người trong lực lượng vũ trang. |
| 7 | Occupation | object | Nghề nghiệp của mỗi người. |  |
| 8 | Relationship | object | ối quan hệ của mỗi cá nhân trong gia đình | - Head of household: Chủ hộ  - Wife/Husband: Vợ/chồng  -Own-child: Con  - Not in family: Không có quan hệ  - Other relative: Khác  - Unmarried: Chung sống |
| 9 | Race | object | Nguồn gốc dân tộc |  |
| 10 | Sex | object | Mô tả giới tính | -Male: nam  -Female: |
| 11 | Capital.gain | int64 | Lợi nhuận từ vốn của mỗi cá nhân |  |
| 12 | Capital.loss | int64 | Số tiền mất mát về vốn |  |
| 13 | Hours.per.week | int64 | Số giờ làm việc hàng tuần |  |
| 14 | Native.country | object | Quốc gia nguyên quán |  |
| 15 | Income | object | Thu nhập cá nhân |  |

### **1.3 Đặc điểm của dữ liệu**



Tóm tắt thống kê của tập dữ liệu cho biết

- Số lượng giá trị không null (count)

- Số lượng giá trị duy nhất (unique) (chỉ cho kiểu object)

- Giá trị phổ biến nhất (top) (chỉ cho kiểu object)

- Tần số của giá trị phổ biến nhất (freq) (chỉ cho kiểu object)

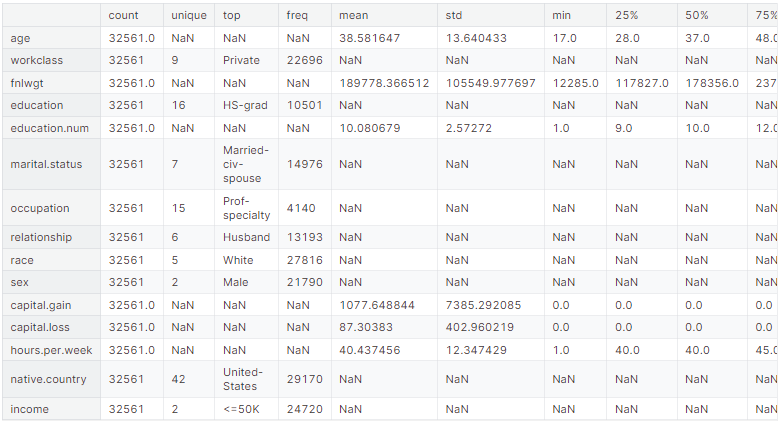
- Giá trị trung bình (mean) (chỉ cho kiểu số)

- Độ lệch chuẩn (std) (chỉ cho kiểu số)

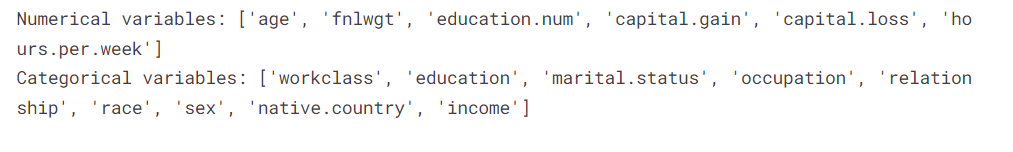
- Giá trị nhỏ nhất (min) (chỉ cho kiểu số)

- Các phần trăm vị trí (25%, 50%, 75%) (chỉ cho kiểu số)

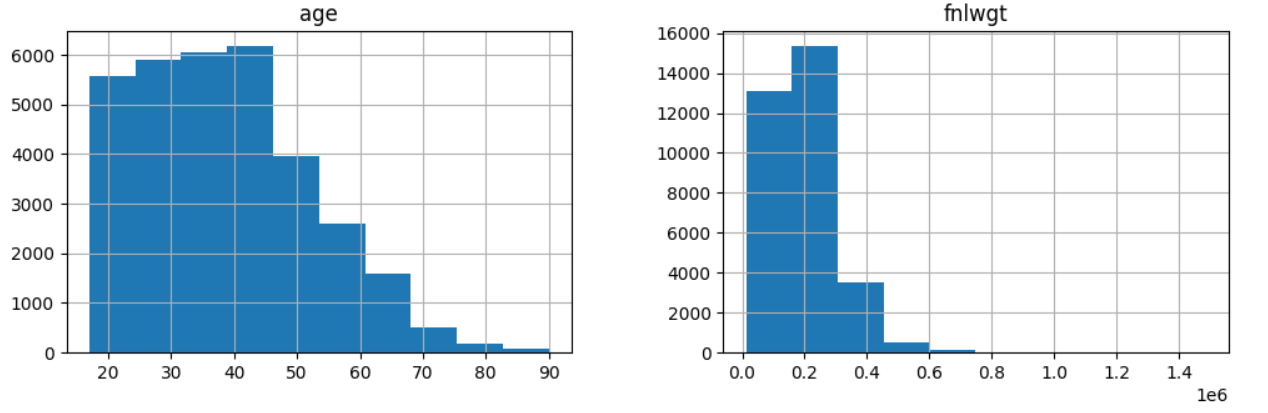
- Giá trị lớn nhất (max) (chỉ cho kiểu số)



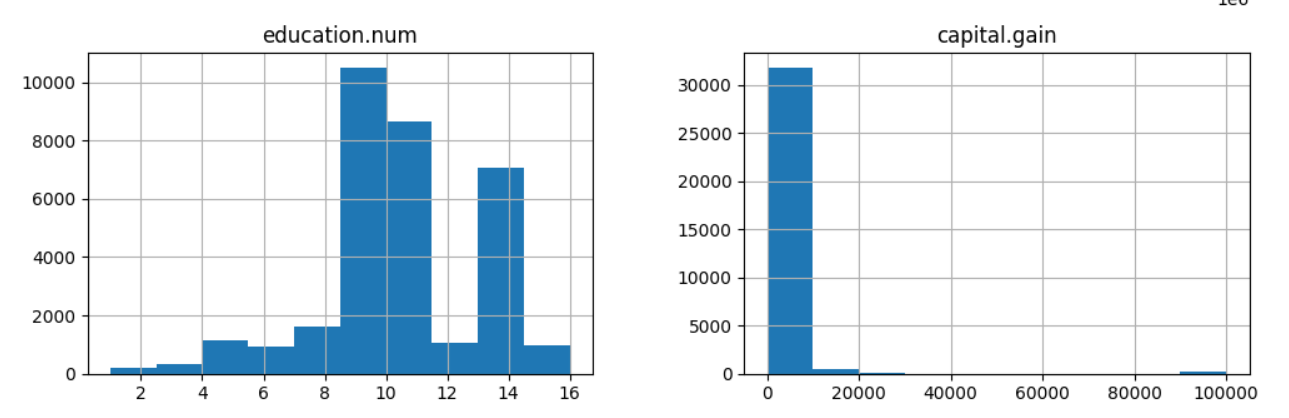
Danh sách các biến số và các biến phân loại



Trực quan hóa các đặc trưng số (numerical features) bằng cách sử dụng các biểu đồ histogram Mục đích của việc này là để phân tích phân phối của các đặc trưng đó trong tập dữ liệu.

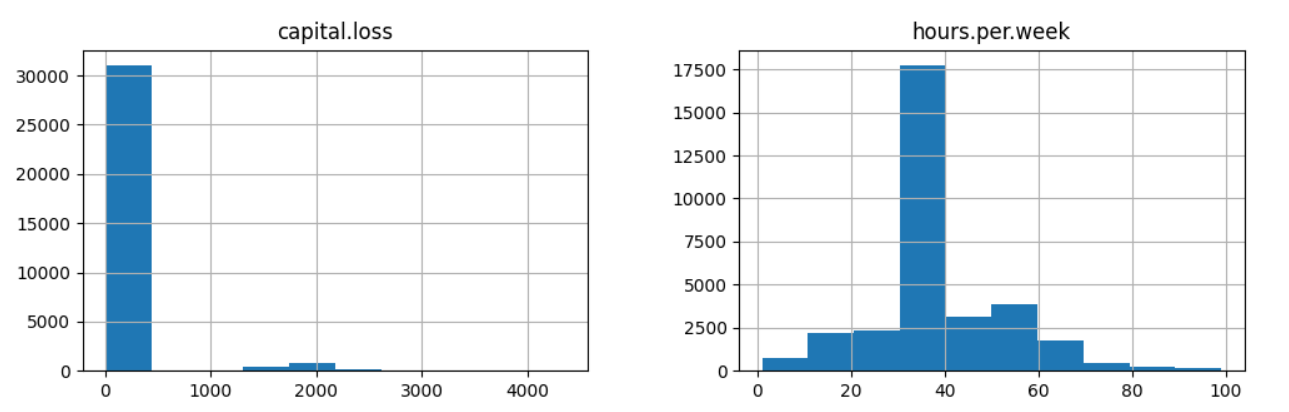


Độ tuổi (Age) thấp nhất là 17 tuổi và caoa nhất là 90 tuổi. Trong tập dữ liệu, độ tuổi chiếm nhiều nhất là từ 17 – 45. Độ tuổi chiếm ít nhất là 60-80

Trọng số cuối cùng - finalweight (thường được viết tắt là fnlwgt) là một giá trị số được gán cho mỗi bản ghi trong tập dữ liệu để cho biết mức độ đại diện của bản ghi đó đối với toàn bộ tập hợp

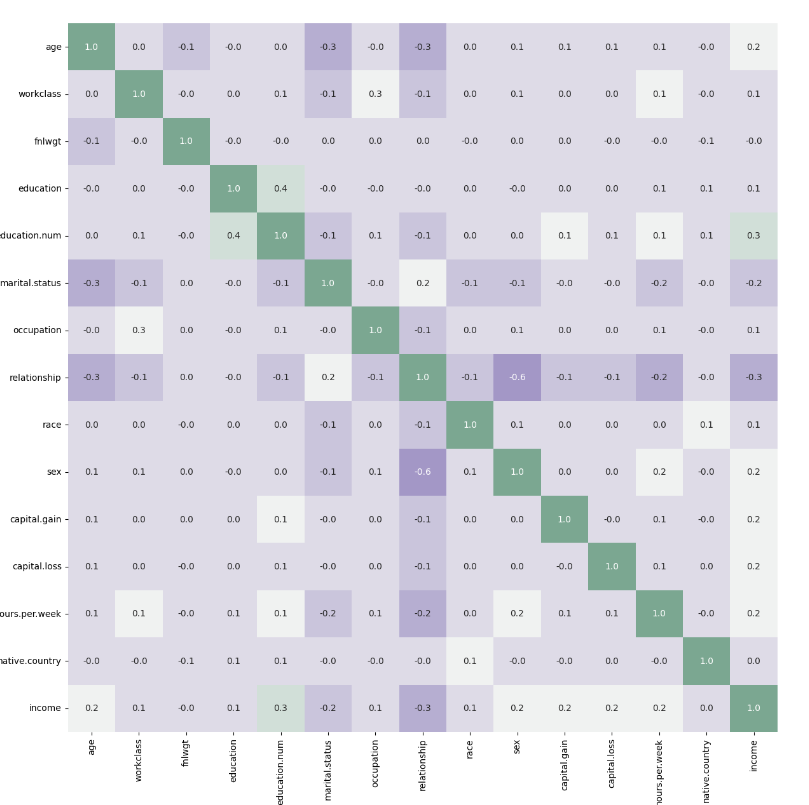
Số năm học tập của tập dữ liệu khảo sát nhiều nhất là từ hơn 8 năm đến hơn 11 năm và từ hơn 12 năm đến hơn 14 năm

Lợi tức vốn đa số là từ 0 – 10 000 USD trong thời gian khảo sát và một số rất ít là 100 000 USD

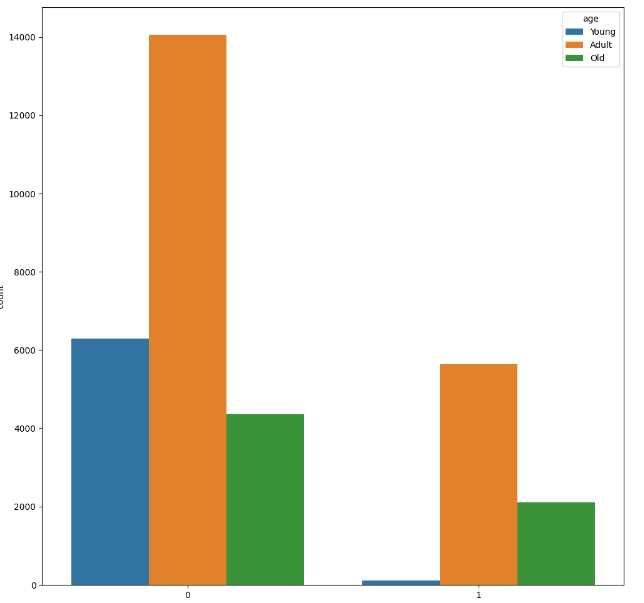


Tương tự, khoản lỗ vốn cũng đa số là từ 0 – 10 000 USD

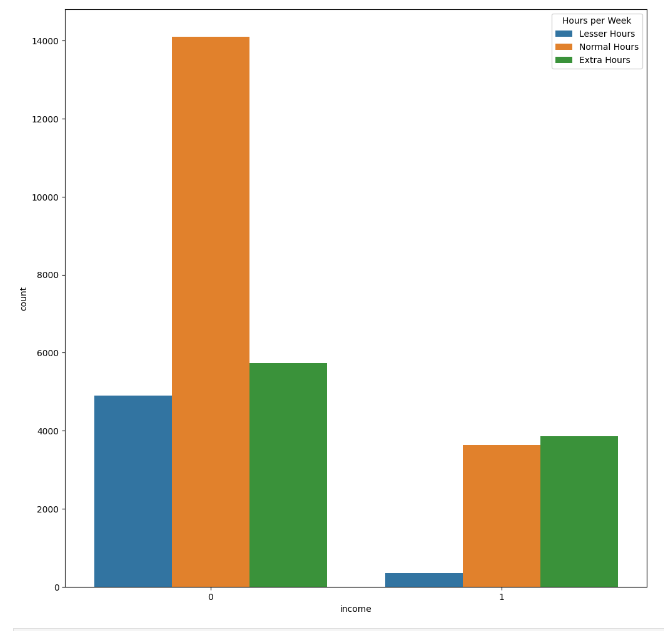
Số giờ làm việc mỗi tuần của một người đa số là là từ khoảng 30-40 tiếng, một số rất ít là từ 80 đến gần 100 tiếng



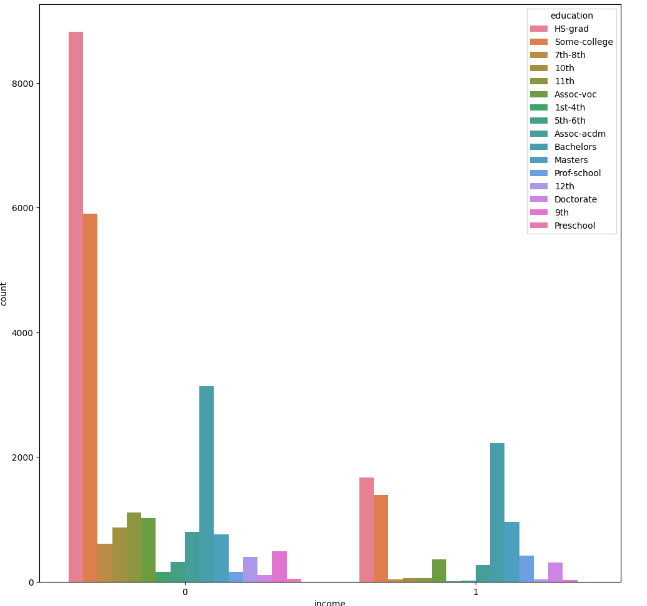
Biểu đồ thể hiện mức độ tương quan cho thấy số năm học và trình độ học vấn khá tương quan với nhau, khoảng 0.4 .Ngoài ra, thu nhập (income) và số năm học có mức độ tương quan 0.3, và mức độ tương quan giữa income và fnlwgt là 0 nên khi tiền xử lý dữ liệu có thể bỏ cột fnlwgt



Dữ liệu ở cột age sẽ được phân chia thành ba độ tuổi young, adult, old. Biểu đồ trên thể hiện tỉ lệ người ở ba độ tuổi có thu nhập dưới 50 000 USD (0) và thu nhập trên 50 000 USD (1). Có thể thấy, ở cả ba độ tuổi số lượng người có thu nhập trên 50 000 USD cao hơn số lượng người có mức thu nhập dưới 50 000 USD. Đa số những người ở độ tuổi young có mức thu nhập dưới 50 000 USD, trong khi một số rất ít người ở độ tuổi young có mức thu nhập trên 50 000 USD.



Khoảng thời gian làm một tuần của mỗi người được chia thành 3 mức, ít hơn giờ làm thông thường, bình thường và nhiều hơn giờ làm bình thường. Biểu đồ thể hiện mối tương quan giữa giờ làm một tuần và thu nhập cho thấy đa số những người có giờ làm ít hơn hoặc trung bình đều có mức thu nhập dưới 50 000 USD và khoảng chênh lệch của những người có giờ làm nhiều hơn bình thường giữa mức thu nhập thấp và cao không nhiều.



Dựa vào biểu đồ tương quan giữa thu nhập và trình độ học vấn có thể thấy đa số những người có mức thu nhập trên 50 000 USD là tốt nghiệp phổ thông, cao đẳng hoặc có bằng cử nhân.

**2. Ứng dụng tập dữ liệu**

Tập dữ liệu này sẽ được dùng để dự đoán xem thu nhập hàng năm của một cá nhân có vượt quá 50,000 đô la không dựa trên các hồ sơ điều tra dân số. Bằng cách sử dụng các kỹ thuật khai phá dữ liệu như hồi quy logistic, rừng ngẫu nhiên, cây quyết định.

Dự án này sẽ giúp cung cấp những hiểu biết sâu sắc có giá trị về các yếu tố liên quan đến việc tăng thu nhập và giúp giải quyết sự thiên vị trong hoạt động tài chính. Khi đã có được mô hình dự đoán tốt nhất thì ta cũng có thể ứng dụng mô hình này trong các lĩnh vực như ngân hàng, đầu tư,…

Các bước thực hiện:

- Tiền xử lý dữ liệu: Tiền xử lý dữ liệu bằng cách xử lý các giá trị thiếu, mã hóa biến phân loại và chuẩn hóa các đặc trưng số.

- Khám phá dữ liệu: Khám phá tập dữ liệu để có cái nhìn sâu hơn và thực hiện lựa chọn đặc trưng để xác định các biến ảnh hưởng.

- Huấn luyện mô hình phân loại: Huấn luyện một mô hình phân loại bằng các thuật toán như Hồi quy Logistic, Cây quyết định, Rừng ngẫu nhiên,…

- Điều chỉnh siêu tham số của mô hình: Điều chỉnh các siêu tham số của mô hình bằng các kỹ thuật như tìm kiếm lưới hoặc tìm kiếm ngẫu nhiên.

- Đánh giá hiệu suất của mô hình: Đánh giá hiệu suất của mô hình bằng các chỉ số như độ chính xác, precision, recall và F1-score.

- Phân tích các đặc trưng quan trọng: Phân tích các đặc trưng quan trọng đóng góp vào dự đoán và tạo dự đoán trên dữ liệu dân số mới.

**PHẦN 2: PHÂN TÍCH CÁC KỸ THUẬT KHAI PHÁ ĐỐI VỚI TẬP DỮ LIỆU**

**2.1 Decision tree**

### **2.1.1 Sơ lược về lý thuyết**

#### 2.1.1.1. Định nghĩa:

[Decision tree](https://amis.misa.vn/73005/decision-tree/) hay còn gọi là Cây quyết định là một dạng sơ đồ phân cấp dựa theo quy luật của mối quan hệ nguyên nhân – kết quả, cho phép người sử dụng dự báo được kết quả của các lựa chọn.

#### 2.1.1.2. Cấu trúc:

Nodes: đó là điểm mà cây phân chia theo giá trị của một số thuộc tính/tính năng của tập dữ liệu. Có những nodes có đầu vào và đầu ra nhưng có những node có đầu vào không có đầu ra.

Edges: (Cạnh) Nó hướng kết quả của việc phân chia tới nút tiếp theo.

Root: node đầu tiên sẽ được gọi là root

Leaves (node lá): là những node có đầu vào không có đầu ra.

#### 2.1.1.3. Các loại:

***a. Cây phân loại***(*Classification tree*)

Cây phân loại: Trong khi xây dựng cây quyết định, điều chính là chọn thuộc tính tốt nhất từ danh sách tính năng tổng thể của tập dữ liệu cho nút gốc cũng như cho các nút phụ. Việc lựa chọn các thuộc tính tốt nhất đang đạt được với sự trợ giúp của một kỹ thuật được gọi là Biện pháp lựa chọn thuộc tính (ASM).

Entropy là một khái niệm khá thông dụng được thể hiện mức độ hỗn loạn hay độ nhiễu của data. Độ giảm của entropy được gọi là information gain.

Gini Impurity là chỉ số thể hiện mức độ phân loại sai khi ta chọn ngẫu nhiên một phần tử từ tập data.

***b. Cây hồi quy***(*Regression tree*)

Cây hồi quy ước lượng các hàm giá có giá trị là số thực thay vì được sử dụng cho các nhiệm vụ phân loại

### **2.1.2 Ưu điểm**

Rất đơn giản để hiểu và giải thích : có cấu trúc đơn giản và rõ ràng, dễ dàng để trực quan hóa và diễn giải kết quả. Mỗi quyết định trong cây với tập dữ liệu được thêm vào sẽ tương ứng với một điều kiện kiểm tra đơn giản, dễ hiểu lý do vì sao dữ liệu được phân loại theo cách đó.

Dữ liệu đầu vào cho sơ đồ Decision tree không cần chuẩn hóa: dữ liệu đầu vào từ tập dữ có dữ liệu giá trị bằng số, tên, thể loại.

Cho kết quả nhanh chóng từ các giá trị nhập vào, tiết kiệm được thời gian ra quyết định

### **2.1.3 Nhược điểm**

Phụ thuộc rất lớn vào dữ liệu đầu vào được nhập, cấu trúc mô hình của cây quyết định có thể bị thay đổi hoàn toàn bởi một thay đổi nhỏ trên tập dữ liệu ban đầu.

Kết quả dự báo chỉ mang tính chất dự báo, không bao giờ đảm bảo được 100% chính xác bởi chỉ cần dữ liệu đầu vào bị thay đổi nhỏ có thể dẫn tới kết quả dự báo mới. Nếu có các giá trị sai lệch hoặc thiếu trong các thuộc tính như "workclass" và "occupation" như ký tự "?", mô hình có thể tạo ra các quyết định không chính xác.

## **2.2 Random forest**

### **2.2.1 Sơ lược về lý thuyết**

#### 2.2.1.1. Định nghĩa

Random forest là một phương pháp thống kê mô hình hóa bằng máy (machine learning statistic)dùng để phục vụ các mục đích phân loại, tính hồi quy và các nhiệm vụ khác bằng cách xây dựng nhiều cây quyết định (Decision tree).

#### 2.2.1.2. Đặc điểm của Random forest

Random Forest là một thuật toán học có giám sát, được sử dụng để giải quyết bài toán phân loại và hồi quy, một số đặc điểm nổi bật:

- Ensemble Learning: kết hợp nhiều bộ phân loại để giải quyết vấn đề phức tạp và cải thiện hiệu suất mô hình

- Xây dựng nhiều cây quyết định: Thuật toán Random Forest xây dựng nhiều cây quyết định (Decision Tree). Mỗi cây quyết định được xây dựng từ một tập dữ liệu khác nhau và có yếu tố ngẫu nhiên.

Nguyên tắc ngẫu nhiên:

- Lấy ngẫu nhiên dữ liệu: lấy ngẫu nhiên một tập dữ liệu từ bộ dữ liệu gốc bằng kỹ thuật Bootstrapping (random sampling with replacement). Tập dữ liệu mới có thể chứa các dữ liệu trùng nhau.

- Lấy ngẫu nhiên các thuộc tính: trên tập dữ liệu mới, chọn ngẫu nhiên một số thuộc tính để xây dựng cây quyết định.

#### 2.2.1.3. Cách hoạt động của Random forest

Bước 1: Xây dựng tập dữ liệu ngẫu nhiên từ training set:

* Dùng kỹ thuật Bootstrapping (random sampling with replacement)
* Tập dữ liệu có thể xuất hiện dữ liệu trùng nhau
* Tập dữ liệu ngẫu nhiên không lấy toàn bộ dữ liệu trong tập dữ liệu ban đầu

Bước 2: Xây dựng Decision Tree (cây quyết định) từ tập dữ liệu ngẫu nhiên bên trên

* Lựa chọn ngẫu nhiên một số thuộc tính để xây dựng cây quyết định
* Dự đoán trên từng cây quyết định.

Bước 3: Lặp lại bước 1, bước 2

Bước 4: Tổng hợp kết quả: Kết quả cuối cùng được tổng hợp từ các cây quyết định. Dự đoán cuối cùng được quyết định bằng cách lấy mode (đối với bài toán phân loại) hoặc trung bình (đối với bài toán hồi quy) từ tất cả cây quyết định.

### **2.2.2 Ưu điểm**

- Random Forest xây dựng nhiều cây quyết định dựa trên các mẫu ngẫu nhiên từ tập dữ liệu huấn luyện và kết hợp các kết quả của chúng. Điều này giúp giảm thiểu overfitting.

- Tập dữ liệu của có nhiều giá trị thiếu (ký tự "?"), đặc biệt trong các thuộc tính như "workclass" và "occupation". Random Forest có khả năng xử lý tốt các trường hợp dữ liệu thiếu thông qua quá trình xây dựng nhiều cây, mỗi cây có thể sử dụng các phần khác nhau của dữ liệu có sẵn.

- Tăng độ chính xác hơn cho mô hình : tổng hợp dự đoán từ nhiều cây quyết định giúp tăng độ chính xác của mô hình so với một cây quyết định đơn lẻ.

### **2.2.3 Nhược điểm**

- Tốn Nhiều Bộ Nhớ và Thời Gian: mô hình xây dựng nhiều cây quyết định, đòi hỏi nhiều tài nguyên tính toán và bộ nhớ hơn, với các tập dữ liệu lớn đưa vào quá trình huấn luyện và dự đoán có thể trở nên tốn kém về mặt tài nguyên.

- Khó diễn giải về mô hình sinh ra sau khi khai phá tập dữ liệu: có nhiều cây quyết định trong mô hình khiến cho việc giải thích chi tiết trở nên khó khăn hơn.

- Việc huấn luyện lâu hơn so với mô hình decision tree: bởi vì xây dựng nhiều cây quyết định hơn cho tập dữ liệu ban đầu cho nên việc xử lý dữ liệu sẽ trở nên tốt thời gian hơn.

## **2.3 KNN**

### **2.3.1 Sơ lược về lý thuyết**

KNN là viết tắt của "K-nearest neighbors" (K láng giềng gần nhất). Đây là một phương pháp trong machine learning được sử dụng cho cả các bài toán phân loại (classification) và dự đoán (regression).

Ý tưởng cơ bản của thuật toán KNN là dự đoán nhãn hoặc giá trị của một điểm dữ liệu mới dựa trên những điểm dữ liệu gần nhất với nó trong không gian đặc trưng. K trong KNN đề cập đến số lượng các láng giềng gần nhất mà bạn muốn sử dụng để đưa ra dự đoán.

Thuật toán hoạt động như sau:

- Tính khoảng cách giữa điểm dữ liệu mới và tất cả các điểm dữ liệu trong tập huấn luyện.

- Chọn ra K điểm dữ liệu gần nhất.

- Dùng đa số phiếu bầu (với phân loại) hoặc trung bình (với regression) của nhãn của các điểm láng giềng để dự đoán nhãn hoặc giá trị cho điểm dữ liệu m

### **2.3.2 Ưu điểm**

- Dễ hiểu và triển khai: KNN là một thuật toán đơn giản và dễ hiểu. Nó không yêu cầu giả định về phân phối của dữ liệu và có thể được triển khai một cách nhanh chóng.

- Khả năng xử lý dữ liệu phức tạp: KNN có thể xử lý cả dữ liệu có cấu trúc và không cấu trúc, cũng như dữ liệu có nhiều biến số và đặc trưng.

- Khả năng dự đoán linh hoạt: KNN không tạo ra một mô hình cụ thể mà chỉ lưu trữ dữ liệu đào tạo. Do đó, nó có thể dự đoán các điểm dữ liệu mới mà không cần phải huấn luyện lại mô hình.

- Hiệu suất tốt với dữ liệu nhỏ: KNN thường hoạt động tốt với các tập dữ liệu nhỏ hoặc có kích thước nhỏ vì không yêu cầu nhiều tài nguyên tính toán.

### **2.3.3 Nhược điểm**

- Nhạy cảm với dữ liệu nhiễu: KNN có thể bị ảnh hưởng bởi dữ liệu nhiễu hoặc outliers, đặc biệt là khi số lượng hàng xóm gần (k) nhỏ.

- Yêu cầu xử lý dữ liệu: KNN yêu cầu phải chuẩn hóa dữ liệu trước khi sử dụng để đảm bảo rằng mọi đặc trưng có trọng số như nhau.

- Tính toán phức tạp: Trong quá trình dự đoán, KNN cần phải tính toán khoảng cách giữa điểm dữ liệu mới và tất cả các điểm trong tập dữ liệu huấn luyện. Điều này có thể là một vấn đề với các tập dữ liệu lớn.

- Chọn giá trị k phù hợp: Việc chọn giá trị k phù hợp có thể ảnh hưởng đến hiệu suất của mô hình. Giá trị k quá nhỏ có thể dẫn đến overfitting, trong khi giá trị k quá lớn có thể dẫn đến underfitting.

- Khả năng xử lý dữ liệu không đồng nhất: KNN không xử lý tốt với dữ liệu có độ khác biệt lớn giữa các đặc trưng hoặc dữ liệu không đồng nhất.

## **2.4 Support Vector Machines**

### **2.4.1 Sơ lược về lý thuyết**

#### 2.4.1.1. Định nghĩa.

Support Vector là một thuật toán giám sát, sử dụng chủ yếu cho việc phân loại. Thuật toán sẽ thực hiện vẽ đồ thị dữ liệu là các điểm trong n chiều với giá trị của mỗi tính năng sẽ là một phần liên kết. Sau đó thực hiện tìm hyper-plane phân chia các lớp.

#### 2.4.1.2. Siêu phẳng (hyper-plane)

Trong không gian hai chiều, siêu phẳng là một đường thẳng. Trong không gian ba chiều, siêu phẳng là một mặt phẳng. Trong không gian có nhiều chiều hơn, siêu phẳng là một không gian chiều thấp hơn.

#### 2.4.1.3. Hàm mất mát và Margin

Khoảng cách giữa siêu phẳng và các điểm dữ liệu gần nhất (vector hỗ trợ) là yếu tố quyết định hiệu quả của SVM. Khoảng cách này gọi là margin, và SVM tối ưu hóa để làm cho margin này càng lớn càng tốt.

#### 2.4.1.4. Hàm Kernel

SVM có thể mở rộng để giải quyết các bài toán phi tuyến bằng cách sử dụng các hàm kernel để ánh xạ dữ liệu vào không gian chiều cao hơn

### **2.4.2 Ưu điểm**

- Ưu điểm của SVM là hiệu quả trong không gian chiều cao, với tập dữ liệu nhiều thuộc tính như tập dữ liệu đầu vào “Adult income dataset” thì SVM có khả năng tìm ra siêu phẳng tối ưu trong không gian này.

- Thích ứng với dữ liệu mất cân bằng: SVM có thể điều chỉnh tham số C (regularization parameter) để đối phó tốt hơn với sự mất cân bằng giữa các lớp, ví dụ như khi số lượng mẫu “income >50K” khi bị chỉnh sửa ít hơn nhiều so với “income <=50K”

- Tiền xử lý linh hoạt: Với các giá trị thiếu như “?” trong thuộc tính “workclass” và “occupation”, SVM có thể kết hợp với các kỹ thuật tiền xử lý dữ liệu để đảm bảo dữ liệu sạch và sẵn sàng cho việc huấn luyện mô hình.

**2.4.3 Nhược điểm**

- SVM nhạy cảm với dữ liệu nhiễu và outliers. Các điểm dữ liệu nằm xa siêu phẳng phân loại có thể ảnh hưởng đáng kể đến vị trí của siêu phẳng này. Trong tập dữ liệu ban đầu, có thể có các giá trị ("?") và các outliers có thể làm giảm hiệu suất của mô hình

- Tập dữ liệu ban đầu có có giá trị “?“ trong các thuộc tính "workclass" và "occupation". SVM không thể xử lý trực tiếp các giá trị thiếu này, do đó cần tiền xử lý dữ liệu kỹ lưỡng trước khi huấn luyện mô hình, điều này có thể làm tăng thêm độ phức tạp.

- Thời gian huấn luyện dài: Với các tập dữ liệu như Adult Income Dataset, SVM có thể mất nhiều thời gian để huấn luyện. Việc tìm kiếm siêu phẳng phân loại tốt nhất yêu cầu nhiều tính toán, đặc biệt khi sử dụng các hàm kernel phi tuyến.

## **2.5 Logistic Regression**

### **2.5.1 Sơ lược về lý thuyết**

#### 2.5.1.1. Khái niệm.

Logistic Regression (hồi quy logistic) là một phương pháp phân tích thống kê được sử dụng để mô hình hóa mối quan hệ giữa một biến phụ thuộc nhị phân (binary dependent variable) và một hoặc nhiều biến độc lập (independent variables). Khác với hồi quy tuyến tính, logistic regression được thiết kế để dự đoán các giá trị phân loại (category outcomes) thay vì các giá trị liên tục.

#### 2.5.1.2. Hàm Logistic (Logistic Function).

Hồi quy logistic sử dụng hàm logistic (còn gọi là hàm sigmoid) để biến đổi giá trị đầu ra của hồi quy tuyến tính thành một giá trị xác suất nằm trong khoảng từ 0 đến 1.

#### 2.5.1.3. Hàm tổn thất (Loss Function) và Tối ưu hóa.

- Hàm tổn thất (Log-Loss): Trong logistic regression, hàm tổn thất thường được gọi là Log-Loss hoặc Binary Cross-Entropy. Nó đo lường sự khác biệt giữa các giá trị dự đoán và giá trị thực tế.

- Tối ưu hóa: Hồi quy logistic sử dụng các thuật toán tối ưu hóa như Gradient Descent để tìm ra các giá trị tối ưu của các hệ số.

### **2.5.2 Ưu điểm**

- Logistic Regression là một mô hình thống kê tương đối đơn giản, dễ hiểu và dễ triển khai. Sử dụng hàm logistic (sigmoid) để ánh xạ mọi giá trị đầu vào thành xác suất, từ đó dự đoán nhãn đầu ra.

- Logistic Regression đặc biệt hữu ích cho các bài toán phân loại nhị phân, tức là khi dự đoán một trong hai lớp (ví dụ: có bệnh hoặc không có bệnh).

- Các hệ số của Logistic Regression cung cấp thông tin về tầm quan trọng của từng đặc trưng (feature) đối với xác suất của nhãn. Điều này giúp dễ dàng giải thích và hiểu được cách mô hình ra quyết định.

### **2.5.3 Nhược điểm**

- Logistic Regression giả định rằng mối quan hệ giữa các biến độc lập và log-odds (logarithm của tỷ lệ odds) là tuyến tính. Điều này có thể không đúng với các tập dữ liệu phức tạp và có mối quan hệ phi tuyến tính giữa các biến.

- Logistic Regression hoạt động tốt khi các lớp có thể phân tách bằng một đường thẳng (hoặc siêu phẳng trong không gian nhiều chiều). Nếu dữ liệu không thể phân tách tuyến tính, hiệu suất của mô hình có thể giảm đáng kể.

- Nhạy cảm với outliers: Các giá trị ngoại lai có thể ảnh hưởng lớn đến mô hình Logistic Regression vì các giá trị này có thể làm sai lệch đường phân tách.

## **2.6 Navie Bayes**

### **2.6.1 Sơ lược về lý thuyết**

#### 2.6.1.1. Khái niệm.

- Là một thuật toán học máy phân loại được sử dụng rộng rãi trong các bài toán phân loại văn bản và dữ liệu. Thuật toán này dựa trên định lý Bayes và giả định "ngây thơ" (naive) rằng các đặc trưng đầu vào độc lập với nhau.

#### 2.6.1.2. Định lý Bayes.

- Là một công cụ cơ bản trong thống kê và xác suất, cho phép tính xác suất của một sự kiện dựa trên các thông tin liên quan đến sự kiện đó. Trong bài toán phân loại, chúng ta muốn tính xác suất của một lớp dựa trên các đặc trưng của mẫu dữ liệu

#### 2.6.1.3. Giả định "ngây thơ" của Naive Bayes.

- Là một giả định đơn giản nhưng mạnh mẽ, giả định rằng các đặc trưng của một mẫu dữ liệu là độc lập.

### **2.6.2 Ưu điểm**

- Naive Bayes dựa trên giả định cơ bản các biến độc lập với nhau giúp làm giảm độ phức tạp của mô hình.

- Naive Bayes tập trung vào việc phân loại dữ liệu một cách nhanh chóng và hiểu quả

### **2.6.3 Nhược điểm**

- Giả định độc lập có điều kiện: Naive Bayes giả định rằng các đặc trưng đầu vào là độc lập có điều kiện và việc giả định này dẫn đến kết quả dự đoán không chính xác.

- Naive Bayes có thể không hoạt động tốt nếu có dữ liệu mới nằm ngoài phạm vi của dữ liệu huấn luyện. Chỉ dựa vào việc tính toán xác suất từ dữ liệu huấn luyện và gặp vấn đề khi dữ liệu thử nghiệm khác biệt quá nhiều.

- Tập dữ liệu có một đặc trưng không xuất hiện trong tập huấn luyện, Naive Bayes gán xác suất 0 cho lớp đó (zero frequency). Dẫn đến dự đoán không chính xác khi có dữ liệu mới mà có đặc trưng chưa xuất hiện trong tập huấn luyện.

## **2.7 Feedforward neural network**

### **2.7.1 Sơ lược về lý thuyết**

Mạng nơron truyền thẳng nhiều lớp, hay còn được gọi là mạng nơron nhân tạo đa tầng (Multilayer Perceptron - MLP), là một loại kiến trúc mạng nơron sâu phổ biến trong lĩnh vực học máy và deep learning.

Mạng nơron truyền thẳng nhiều lớp bao gồm ít nhất ba loại lớp:

- Lớp vào (input layer): Là lớp đầu tiên của mạng, nhận dữ liệu đầu vào và truyền nó qua mạng. Mỗi nút trong lớp này biểu diễn một thuộc tính hoặc đặc trưng của dữ liệu.

- Các lớp ẩn (hidden layers): Đây là các lớp nằm giữa lớp vào và lớp ra. Mỗi lớp ẩn có nhiều nút nơron, mỗi nút liên kết với tất cả các nút trong lớp trước đó và lớp sau nó. Các lớp ẩn này giúp mạng học được các biểu diễn phức tạp của dữ liệu.

- Lớp ra (output layer): Là lớp cuối cùng của mạng, đưa ra dự đoán hoặc giá trị đầu ra. Số lượng nút trong lớp này thường phụ thuộc vào loại bài toán, ví dụ, một nút cho mỗi lớp phân loại trong bài toán phân loại, hoặc một nút cho mỗi chiều trong bài toán dự đoán.

### **2.7.2 Ưu điểm**

- Khả năng học được các biểu diễn phức tạp: Mạng nơron truyền thẳng có khả năng học được các biểu diễn phức tạp của dữ liệu, đặc biệt là khi sử dụng nhiều lớp ẩn. Điều này cho phép nó thích hợp với các bài toán có dữ liệu phức tạp và không gian đặc trưng lớn.

- Hiệu suất cao khi được huấn luyện đúng cách: Nếu được huấn luyện đúng cách trên tập dữ liệu đủ lớn và đại diện, mạng nơron truyền thẳng có thể đạt được hiệu suất cao và đưa ra các dự đoán chính xác.

- Linh hoạt trong thiết kế kiến trúc: Mạng nơron truyền thẳng có thể được thiết kế với nhiều lớp ẩn và số lượng nút nơron khác nhau, cho phép tối ưu hóa cho từng bài toán cụ thể.

### **2.7.3 Nhược điểm**

- Dễ bị overfitting: Mạng nơron truyền thẳng có khả năng cao bị overfitting, đặc biệt là khi có nhiều lớp ẩn và số lượng tham số lớn. Điều này đặc biệt đúng khi dữ liệu huấn luyện không đủ lớn hoặc không đủ đa dạng.

- Đòi hỏi nhiều dữ liệu huấn luyện: Mạng nơron truyền thẳng thường yêu cầu một lượng lớn dữ liệu huấn luyện để đạt được hiệu suất tốt, đặc biệt là đối với các mô hình phức tạp.

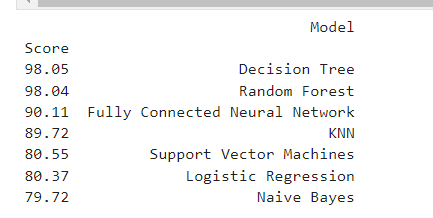
- Đòi hỏi nhiều tính toán: Khi mạng có nhiều lớp và nút nơron, quá trình huấn luyện và dự đoán có thể tốn kém về mặt tính toán, đặc biệt là trên các tập dữ liệu lớn.

**PHẦN 3: KẾT QUẢ KHAI PHÁ**

## **3.1 Đánh giá – nhận xét – trả lời câu hỏi**

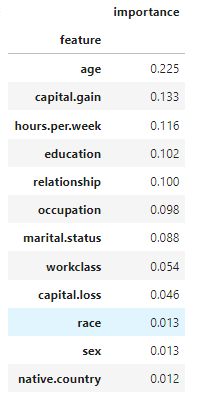
Tập dữ liệu ban đầu sau khi đã hoàn thành xử lý xong sẽ được chia ra 70% dữ liệu để training và 30% dữ liệu còn lại dùng để test

Dùng thư viện scikit-learn để áp dụng các kỹ thuật khai phá dữ liệu được giới thiệu ở phía trên vào mô hình, sau đó thực hiện so sánh đánh giá hiệu quả mô hình, ta được kết quả như hình dưới

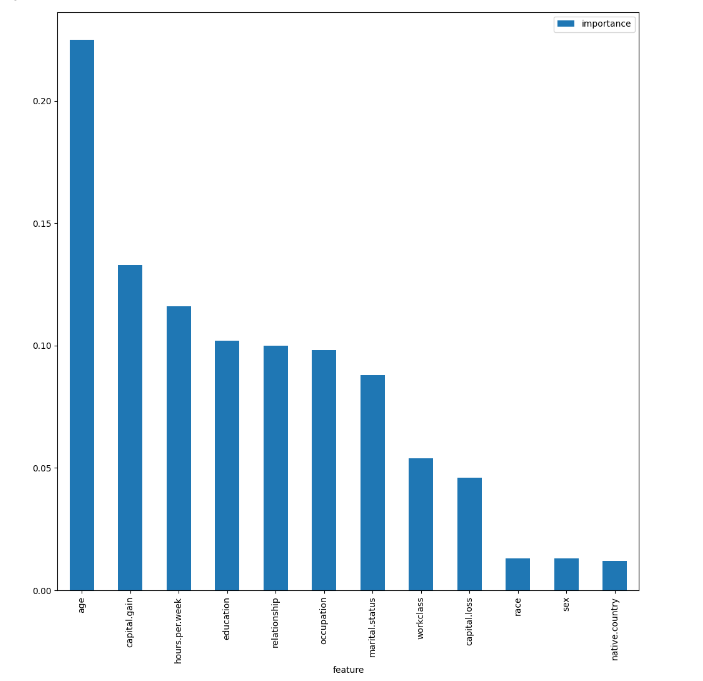


Có thể thấy, mô hình đạt hiệu quả cao nhất là decision tree, random forest, neural network (đều trên 90%) kế tiếp là KNN.

Hình ảnh dưới đây thể hiện tầm quan trọng của các đặc trưng (features) đối với income trong mô hình Random Forest đã được huấn luyện



Biểu đồ thể hiện tầm quan trọng của các đặc trưng



## **3.2 Ý nghĩa**

- Dự đoán thu nhập: Một trong những ứng dụng chính của dự án này là dự đoán liệu một người trưởng thành có thu nhập trên 50000 USD hay không dựa trên các đặc trưng có sẵn trong tập dữ liệu. Điều này có thể hữu ích cho các tổ chức hoạt động trong lĩnh vực tài chính, ngân hàng, bảo hiểm, hoặc thậm chí là chính phủ để hiểu và dự đoán xu hướng thu nhập của công dân.

- Segmentation và Marketing: Khi bạn hiểu được nhóm người có khả năng thu nhập cao và nhóm người có khả năng thu nhập thấp, bạn có thể tạo ra các chiến lược marketing phù hợp để tiếp cận và phục vụ khách hàng mục tiêu của mình. Điều này có thể giúp tối ưu hóa chi phí marketing và tăng cường hiệu suất chiến dịch quảng cáo.

- Quản lý rủi ro và tài chính: Dự án này cũng có thể hữu ích trong việc quản lý rủi ro và tài chính, bằng cách đánh giá rủi ro của các khoản vay hoặc các giao dịch tài chính dựa trên dự đoán thu nhập của cá nhân.

- Nghiên cứu và phân tích xã hội: Dự án này cũng có thể được sử dụng trong nghiên cứu xã hội để hiểu sâu hơn về mối quan hệ giữa các yếu tố như tuổi tác, giáo dục, nghề nghiệp và thu nhập.

- Tối ưu hóa chính sách công cộng: Chính phủ có thể sử dụng thông tin về thu nhập để định hình các chính sách về thuế, bảo hiểm xã hội, giáo dục và các lĩnh vực khác nhằm hỗ trợ các nhóm dân số có thu nhập thấp.

- Đối với doanh nghiệp và doanh nhân: Dự án này cung cấp thông tin cần thiết để phát triển sản phẩm hoặc dịch vụ đích thực cho các nhóm khách hàng mục tiêu, từ đó tạo ra cơ hội kinh doanh mới hoặc tối ưu hóa sản phẩm/dịch vụ hiện tại.

**PHẦN 4 : KẾT LUẬN**

## **4.1 Kết quả đạt được**

Thực hiện ứng dụng kỹ thuật khai phá dữ liệu vào bài toán thực tế giúp chúng em hiểu sâu hơn về kiến thức cũng như ý nghĩa của môn học Khai phá dữ liệu và thành thạo hơn trong việc ứng dụng các mô hình, thuật toán khai phá dữ liệu cũng như cách sử dụng các thư viện học máy scikit-learning, pandas, seaborn,…. Nâng cao kỹ năng nhìn nhận và tiếp thu, xử lý thông tin đầu vào, đặt những câu hỏi tiền đề và thảo luận hướng đi để giải quyết và tìm kiếm câu trả lời cho câu hỏi. Ngoài ra, nhóm chúng em còn được rèn luyện và kỹ năng làm việc nhóm, khả năng trình bày, tư duy giải quyết vấn đề

## **4.2 Những hạn chế**

Nhóm chỉ mới sử dụng được các kỹ thuật khai phá dữ liệu đã nêu trên để áp dụng cho tập dữ liệu ban đầu. Điều này khiến việc đánh giá kết quả cuối cùng thiếu đi sự đa dạng trong việc đánh giá các kỹ thuật khai pháp đã được học để áp dụng trên tập dữ liệu đã được lựa chọn ban đầu; ngoài ra việc áp dụng ít kỹ thuật khiến cho chúng em khó có thể đánh giá một cách toàn diện và chính xác hoàn toàn tất cả các phương pháp khai phá dữ liệu có thể sử dụng trên tập dữ liệu đã được lựa chọn. Với số lượng kỹ thuật khai phá dữ liệu hạn chế, sẽ khó để tìm ra giải pháp tối ưu nhất hoặc phương án tốt nhất để áp dụng trên tập dữ liệu mà sẽ chỉ dựa lên những kỹ thuật khai phá dữ liệu đã được chọn mà đưa ra kết quả. Điều này có thể làm giảm hiệu quả của dự án. Ngoài ra, một số mô hình có hiệu quả còn chưa cao nên sẽ dẫn đến kết quả cuối cùng không quá mức tối ưu nên sẽ không có quá nhiều sự lựa chọn mô hình khai phá phù hợp với tập dữ liệu ban đầu.

## **4.3 Cách cải tiến**

Để có thể cải tiến các hạn chế trên việc sử dụng các kỹ thuật khai phá dữ liệu, nhóm chúng em sẽ thực hiện nhiều biện pháp để cải tiến hơn:

Nâng cao hơn kiến thức và kỹ năng của nhóm trong việc hiểu và áp dụng các mô hình đã được học trên các tập dữ liệu để có thể hiểu rõ và hiểu sâu hơn về các kỹ thuật khai phá dữ liệu

Mở rộng phạm vi đánh giá và thử nghiệm các kỹ thuật trên tập dữ liệu: không chỉ tập trung vào một số ít kỹ thuật, nhóm sẽ mở rộng phạm vi đánh giá và thử nghiệm bằng cách thử nghiệm nhiều phương pháp khác nhau. Điều này bao gồm việc áp dụng các kỹ thuật mới, thử nghiệm các mô hình khác nhau, và so sánh hiệu suất của chúng trên tập dữ liệu.

Cải thiện các mô hình để khai phá dữ liệu đã được áp dụng trên tập dữ liệu để độ hiểu quả sẽ cao hơn và mô hình sẽ trở nên tối ưu hơn so với ban đầu.

Tạo ra một kế hoạch phát triển dự án dài hạn, định rõ các bước tiến hành, mục tiêu ngắn hạn và dài hạn, đo lường hiệu suất đảm bảo rằng các mô hình được triển khai một cách có hệ thống và hiệu quả.

TÀI LIỆU THAM KHẢO

[1] scikit-learn: machine learning in Python — scikit-learn 1.5.0 documentation. (n.d.). <https://scikit-learn.org/stable/> <https://scikit-learn.org/stable/> *Accessed 15/05/2024*

[2] CodeLearn. (n.d.). [https://codelearn.io/sharing/thuat-toan-k-nearest-neighbors-knn *Accessed 16/05/2024*](https://codelearn.io/sharing/thuat-toan-k-nearest-neighbors-knn%20Accessed%2016/05/2024) *Accessed 16/05/2024*

[3] Saini, A. (2024, May 22). Guide on Support Vector Machine (SVM) Algorithm. Analytics Vidhya. <https://www.analyticsvidhya.com/blog/2021/10/support-vector-machinessvm-a-complete-guide-for-beginners/> *Accessed 17/05/2024*

[4] Decision Tree algorithm — Machine Learning cho dữ liệu dạng bảng. (n.d.). <https://machinelearningcoban.com/tabml_book/ch_model/decision_tree.html> *Accessed 17/05/2024*