**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THỰC PHẨM TP.HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

-----o0o---

****

**BÀI TẬP LỚN/ BÀI TẬP DỰ ÁN HỌC PHẦN:**

**KHAI THÁC DỮ LIỆU**

**TÊN ĐỀ TÀI:**

**XÂY DỰNG CÂY QUYẾT ĐỊNH TƯ VẤN CHO VAY**

|  |  |
| --- | --- |
| Thành viên  Nguyễn Ngọc Nhung - 2001190720   1. Lê Thùy Na – 2001190679 | **GVBM: Ngô Dương Hà** |

**Thành phố Hồ Chí Minh, ngày 1 tháng 12 năm 2021**

Mục lục

[I. Giới thiệu cây quyết định 5](#_Toc89260554)

[1. Thuật toán ID3 5](#_Toc89260555)

[2. Tri thức dạng luật 6](#_Toc89260556)

[II. Tập dữ liệu 7](#_Toc89260557)

[III. Xử lý tiền dữ liệu 8](#_Toc89260558)

[1. Nạp dữ liệu 8](#_Toc89260559)

[2. Làm sạch dữ liệu 9](#_Toc89260560)

[3. Chuyển đổi dữ liệu 12](#_Toc89260561)

[4. Lọc thuộc tính 12](#_Toc89260562)

[IV. Train model 13](#_Toc89260563)

[V. Ứng dụng 16](#_Toc89260564)

[1. Công cụ 16](#_Toc89260565)

[2. Giao diện 16](#_Toc89260566)

[3. Chức năng 16](#_Toc89260567)

[4. Code 16](#_Toc89260568)

[VI. Mục tham khảo 21](#_Toc89260569)

**LỜI MỞ ĐẦU**

Sự phát triển của công nghệ thông tin và việc ứng dụng công nghệ thông tin trong nhiều lĩnh vực của đời sống, kinh tế xã hội trong nhiều năm qua cũng đồng nghĩa với lượng dữ liệu đã được các cơ quan thu thập và lưu trữ ngày một tích luỹ nhiều lên. Họ lưu trữ các dữ liệu này vì cho rằng trong nó ẩn chứa những giá trị nhất định nào đó. Tuy nhiên, theo thống kê thì chỉ có một lượng nhỏ của những dữ liệu này (khoảng từ 5% đến 10%) là luôn được phân tích, số còn lại họ không biết sẽ phải làm gì hoặc có thể làm gì với chúng nhưng họ vẫn tiếp tục thu thập rất tốn kém với ý nghĩ lo sợ rằng sẽ có cái gì đó quan trọng đã bị bỏ qua sau này có lúc cần đến nó. Mặt khác, trong môi trường cạnh tranh, người ta ngày càng cần có nhiều thông tin với tốc độ nhanh để trợ giúp việc ra quyết định và ngày càng có nhiều câu hỏi mang tính chất định tính cần phải trả lời dựa trên một khối lượng dữ liệu khổng lồ đã có. Với những lý do như vậy, các phương pháp quản trị và khai thác cơ sở dữ liệu truyền thống ngày càng không đáp ứng được thực tế đã làm phát triển một khuynh hướng kỹ thuật mới đó là Kỹ thuật phát hiện tri thức và khai phá dữ liệu (KDD - Knowledge Discovery and Data Mining).

Kỹ thuật phát hiện tri thức và khai phá dữ liệu đã và đang được nghiên cứu, ứng dụng trong nhiều lĩnh vực khác nhau ở các nước trên thế giới, tại Việt Nam kỹ thuật này tương đối còn mới mẻ tuy nhiên cũng đang được nghiên cứu và dần đưa vào ứng dụng.Phát hiện tri thức trong các cơ sở dữ liệu là một qui trình nhận biết các mẫu hoặc các mô hình trong dữ liệu với các tính năng: hợp thức, mới, khả ích, và có thể hiểu được. Còn khai thác dữ liệu là một bước trong qui trình phát hiện tri thức gồm có các thuật toán khai thác dữ liệu chuyên dùng dưới một số qui định về hiệu quả tính toán chấp nhận được để tìm ra các mẫu hoặc các mô hình trong dữ liệu. Nói một cách khác, mục đích của phát hiện tri thức và khai phá dữ liệu chính là tìm ra các mẫu và/hoặc các mô hình đang tồn tại trong các cơ sở dữ liệu nhưng vẫn còn bị che khuất bởi hàng núi dữ liệu.

Môn học “Khai phá dữ liệu” cung cấp cho sinh viên công nghệ thông tin cái nhìn tổng quan về phát hiện tri thức và khai phá dữ liệu. Với những kiến thức đã học, trong bài tiểu luận môn học em tập trung vào các thuật toán cây quyết định với bài toán cụ thể là “Quyết định cho vay”. Bài toán được thực hiện bằng ngôn ngữ python, google colab, và ứng dụng trên pycharm bộ mã nguồn mở khá hiệu quả hiện nay.

# Giới thiệu cây quyết định

* Cây quyết định là một kiểu mô hình dự báo
* Kỹ thuật học máy dùng trong cây quyết định được gọi là học bằng cây quyết định, hay chỉ gọi với cái tên ngắn gọn là cây quyết định
* Phương tiện có tính mô tả dành cho việc tính toán các xác suất có điều kiện
* Sự kết hợp của các kỹ thuật toán học và tính toán nhằm hỗ trợ việc mô tả, phân loại và tổng quát hóa một tập dữ liệu cho trước
* Cây quyết định là một cấu trúc phân cấp của các nút và các nhánh
* 3 loại nút trên cây:
  + Nút gốc
  + Nút nội bộ: mang tên thuộc tính của CSDL
  + Nút lá: mang tên lớp C*i*

–Nhánh: mang giá trị có thể của thuộc tính

* Cây quyết định được sử dụng trong phân lớp bằng cách duyệt từ nút gốc của cây cho đến khi đụng đến nút lá, từ đó rút ra lớp của đối tượng cần xét

## Thuật toán ID3

Giải thuật ID3 (gọi tắt là ID3) Được phát triển đồng thời bởi Quinlan trong AI và Breiman, Friedman, Olsen và Stone trong thống kê. ID3 là một giải thuật học đơn giản nhưng tỏ ra thành công trong nhiều lĩnh vực. ID3 là một giải thuật hay vì cách biểu diễn tri thức học được của nó, tiếp cận của nó trong việc quản lý tính phức tạp, heuristic của nó dùng cho việc chọn lựa các khái niệm ứng viên, và tiềm năng của nó đối với việc xử lý dữ liệu nhiễu.

ID3 biểu diễn các khái niệm (concept) ở dạng các cây quyết định (decision tree). Biểu diễn này cho phép chúng ta xác định phân loại của một đối tượng bằng cách kiểm tra các giá trị của nó trên một số thuộc tính nào đó.

Như vậy, nhiệm vụ của giải thuật ID3 là học cây quyết định từ một tập các ví dụ rèn luyện (training example) hay còn gọi là dữ liệu rèn luyện (training data).

Input: Một tập hợp các ví dụ. Mỗi ví dụ bao gồm các thuộc tính mô tả một tình huống, hay một đối tượng nào đó, và một giá trị phân loại của nó.

Output: Cây quyết định có khả năng phân loại đúng đắn các ví dụ trong tập dữ liệu rèn luyện, và hy vọng là phân loại đúng cho cả các ví dụ chưa gặp trong tương lai.

Giải thuật ID3 xây dựng cây quyết định được trình bày như sau:

Lặp:  
1. Chọn A <= thuộc tính quyết định “tốt nhất” cho nút kế tiếp  
2. Gán A là thuộc tính quyết định cho nút  
3. Với mỗi giá trị của A, tạo nhánh con mới của nút  
4. Phân loại các mẫu huấn luyện cho các nút lá  
5. Nếu các mẫu huấn luyện được phân loại hoàn toàn thì NGƯNG,  
Ngược lại, lặp với các nút lá mới.

Thuộc tính tốt nhất ở đây là thuộc tính có entropy trung bình thấp nhất theo thuộc tính kết quả với  Entropy được tính như sau:

* Gọi S là tập các mẫu huấn luyện
* Gọi p là tỷ lệ các mẫu dương trong S
* Ta có H ≡ – p.log2p – (1 – p).log2(1 – p)

Entropy trung bình của một thuộc tính bằng trung bình theo tỉ lệ của entropy các nhánh: Ảnh có chứa văn bản

Mô tả được tạo tự động

## Tri thức dạng luật

* Tri thức được biểu diễn dưới dạng luật:

IF Điều kiện 1 ^ Điều kiện 2… THEN Kết luận

* Dễ hiểu với con người, được sử dụng chủ yếu trong các  
  hệ chuyên gia
* Rút luật từ cây quyết định: đi từ nút gốc đến nút lá, lấy  
  các phép thử làm tiền đề và phân loại của nút lá làm kết quả

Lưu ý: Một phiên bản khác của thuật toán ID3 sử dụng Informatic Gain thay cho entropy để chọn thuộc tính quyết định. Công thức tính Informatic Gain như sau:

Gain(A) = Entropy(S) – Entropy(A)

Trong đó: S là tập mẫu và A là một thuộc tính. Entropy(S): độ hỗn loạn của tập S.

Entropy(A): độ hỗn loạn trung bình của thuộc tính A (cách tính như trên)

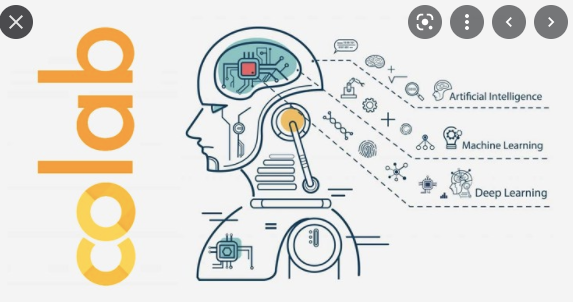
# Tập dữ liệu

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id | age | sex | region | income | married | children | car | save\_act | current\_act | mortgage | loan |

* bank-data .csv : gồm 615 dòng và 12 cột
* id một mã số nhận dạng duy nhất tuổi của khách hàng trong năm
* Giới tính NAM / NỮ ​​
* Khu vực nội thành / nông thôn / ngoại thành / thị trấn
* Thu nhập của khách hàng đã kết hôn là khách hàng đã kết hôn (CÓ / KHÔNG)
* Con số con
* Xe khách hàng sở hữu một chiếc ô tô (CÓ / KHÔNG)
* Tiết kiệm acct khách hàng có tài khoản tiết kiệm (CÓ / KHÔNG)
* Acct hiện tại khách hàng có tài khoản vãng lai (CÓ / KHÔNG)
* Thế chấp khách hàng có thế chấp (CÓ / KHÔNG)
* Loan có đồng ý cho vay (CÓ / KHÔNG)

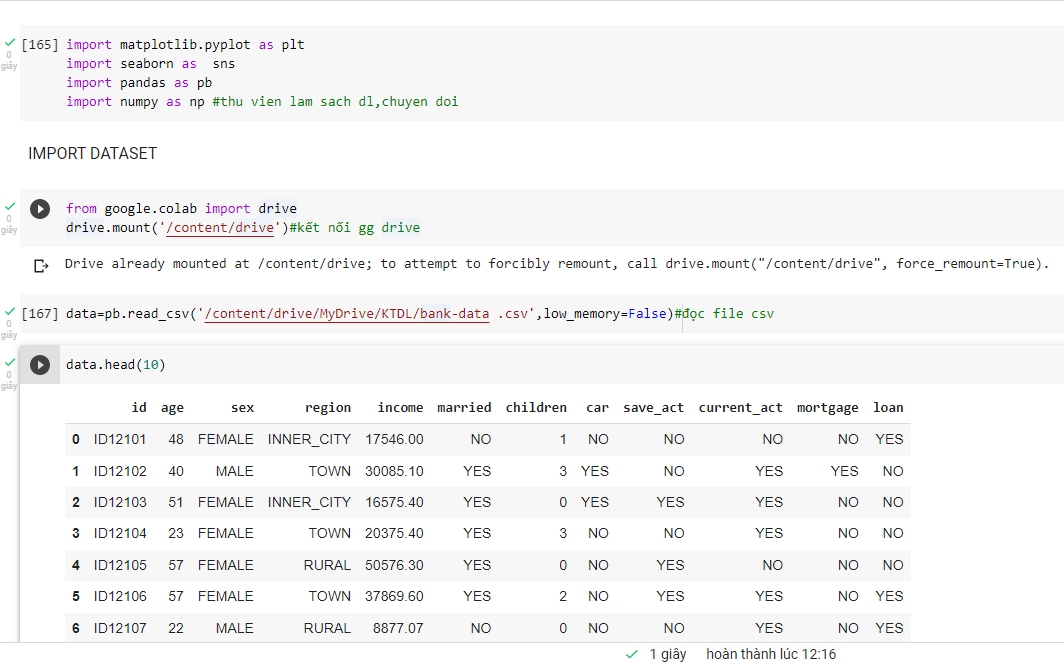
# Xử lý tiền dữ liệu

* Sử dụng ngôn ngữ Python trên Google Colab



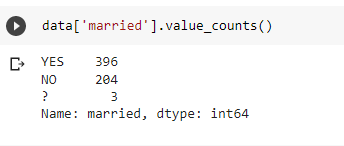
## Nạp dữ liệu

* Tải file chứa dữ liệu bank-data .csv trên thư mục KTDL trên Google Drive .
* Kết nối Google Colab với Google Drive
* Đọc file theo đường dẫn có trong Google Drive trên Colab

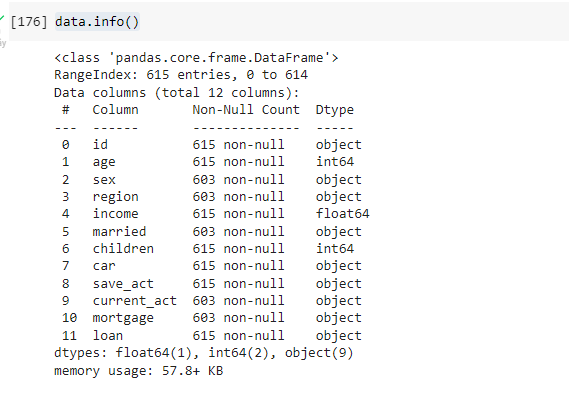


## Làm sạch dữ liệu

* Kiểm tra xem dữ liệu có yêu cầu làm sạch không (kiểm tra các phép đo của tất cả các biến để đảm bảo chúng chính xác)
* Kiểm tra xem có bất kỳ giá trị ngoại lệ hoặc thiếu nào trong tập dữ liệu hay không.
* Giá trị “?” cần làm sạch

****

* Các dòng của dữ liệu rỗng ở cột sex, region, married, current\_act, mortgage

****

* Tính tỷ lệ các giá trị rỗng của từng cột trên toàn bộ dữ liệu
* Vẽ hình minh họa dựa vào tỷ lệ vừa tính

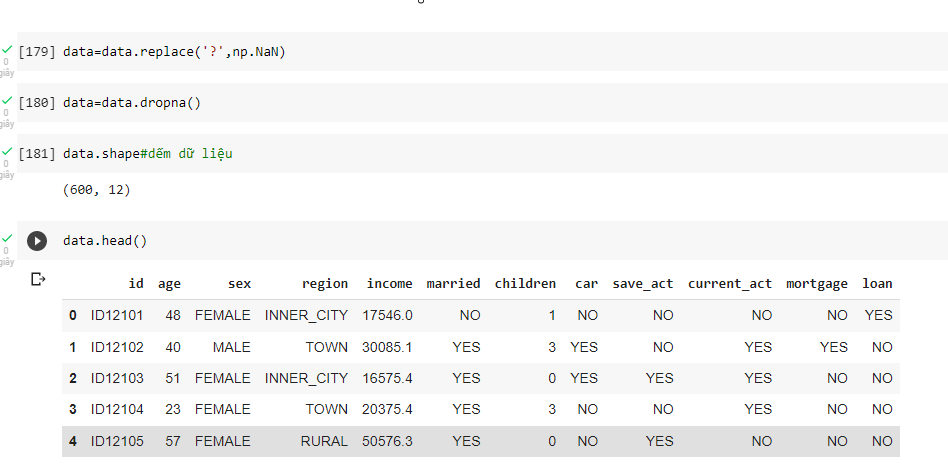
****

* Thực hiện làm sạch

+ Thay các giá trị = ‘?’ thành giá trị NaN

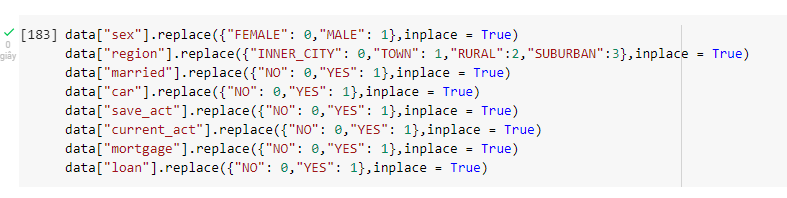
+ Xóa tất cả các giá trị = NaN

+ Đếm lại dữ liệu ta có được 600 dòng x 12 cột đã làm sạch



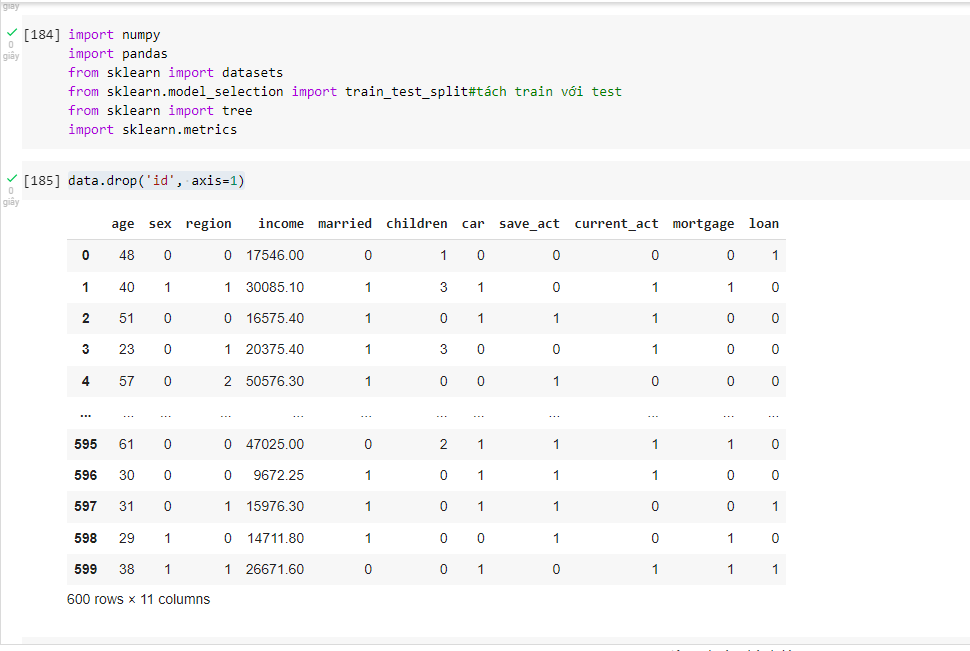
## Chuyển đổi dữ liệu

* Mã hóa số - Numeric Encoding
* Mã hóa số là việc ta gán mỗi giá trị của đăng trưng dạng nhóm thành một số bất kỳ và khác nhau từng đôi một. Ví dụ như trong thuộc tính sex, ta có sex=”Female”là 0, “Male” là 1
* Phương pháp này cũng còn được gọi là mã hóa nhãn (label encoding) hoặc mã hóa số nguyên (integer encoding).



## **Lọc thuộc tính**

* Trong file dữ liệu bank-data.csv, mỗi khách hàng được xác định duy nhất bởi thuộc tính id.
* Chúng ta cần loại bỏ thuộc tính này trước khi thực hiện các bước tiếp theo:

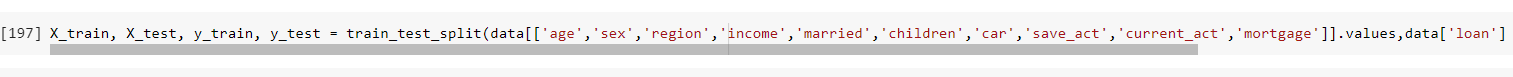


* Sau khi xử lý dữ liệu ta tiến hành lưu lại file đã làm sạch để phục vụ cho việc viết ứng dụng phía sau.

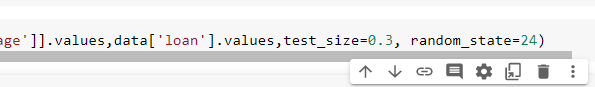


# Train model

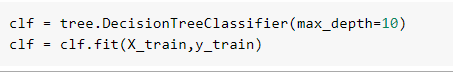
* Sử dụng 70% bộ dữ liệu để đào tạo và 30% để kiểm tra. Tạo mô hình cây quyết định



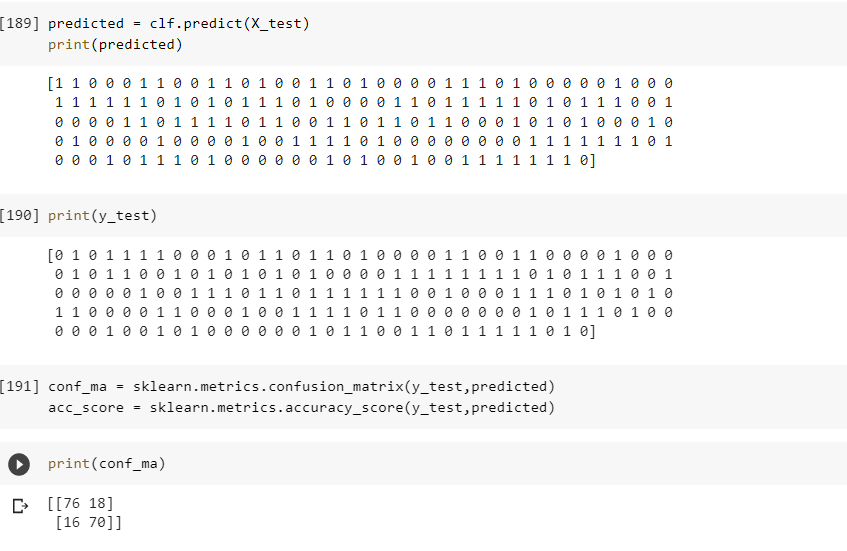
+



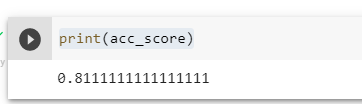
* Với độ sâu 10



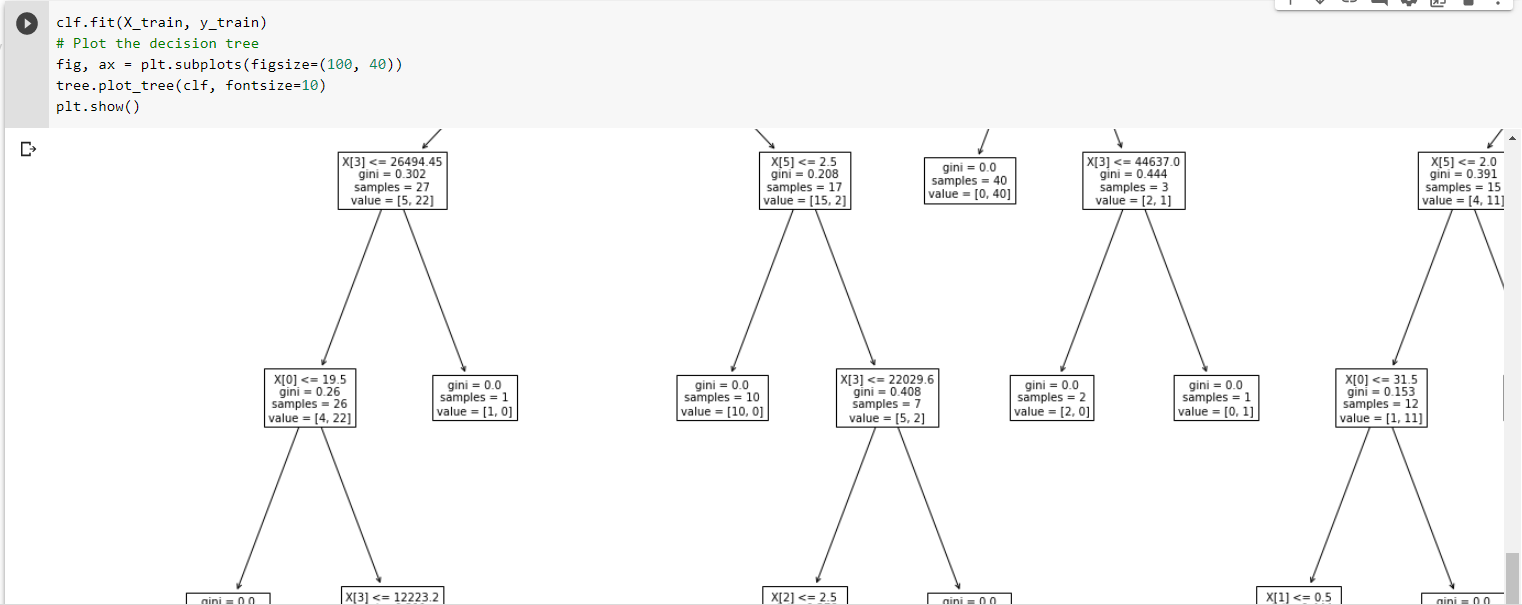
* Xác định tỷ lệ phân loại sai mô hình của bạn và vẽ ma trận nhầm lẫn trên dữ liệu thử nghiệm.



* Tỷ lệ train model 81% và sẽ thay đổi mỗi lần run vì lấy dữ liệu ngẫu nhiên.



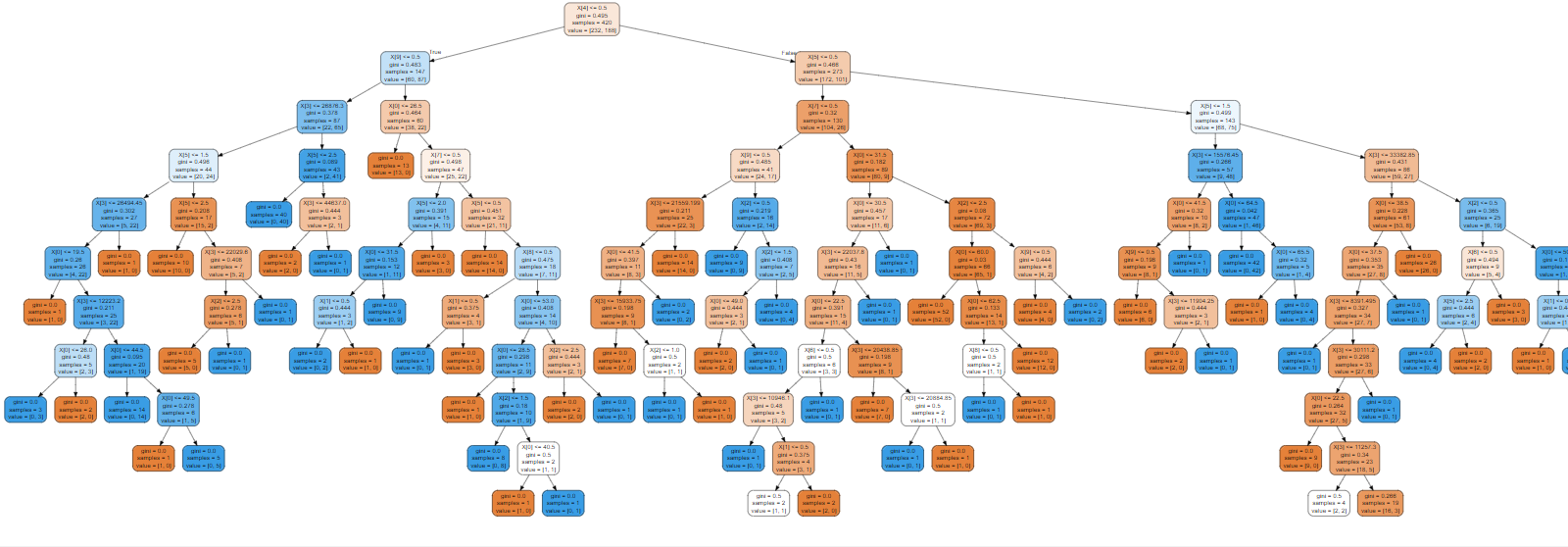
* Vẽ cây quyết định trên Colab (Không chụp hết cây)



* Lưu model



* Có thể dùng file vừa lưu vẽ cây trên web : *http://webgraphviz.com/*



# Ứng dụng

## Công cụ

* Python version 3.9.9
* Pycharm Community Edition version 2021.2.3

## Giao diện

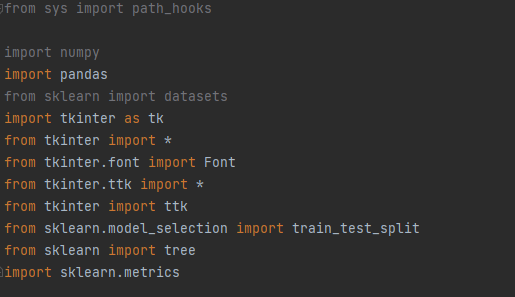


## Chức năng

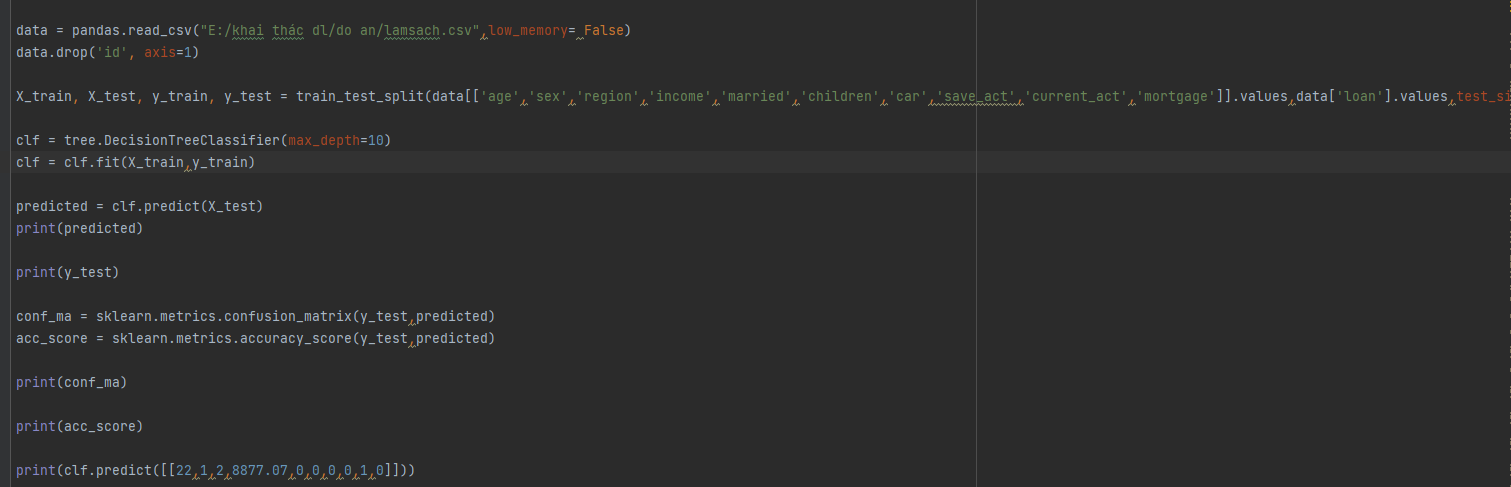
* Dựa vào các tham số truyền vào cho một người, trả về quyết định có cho họ vay hay không.

## Code

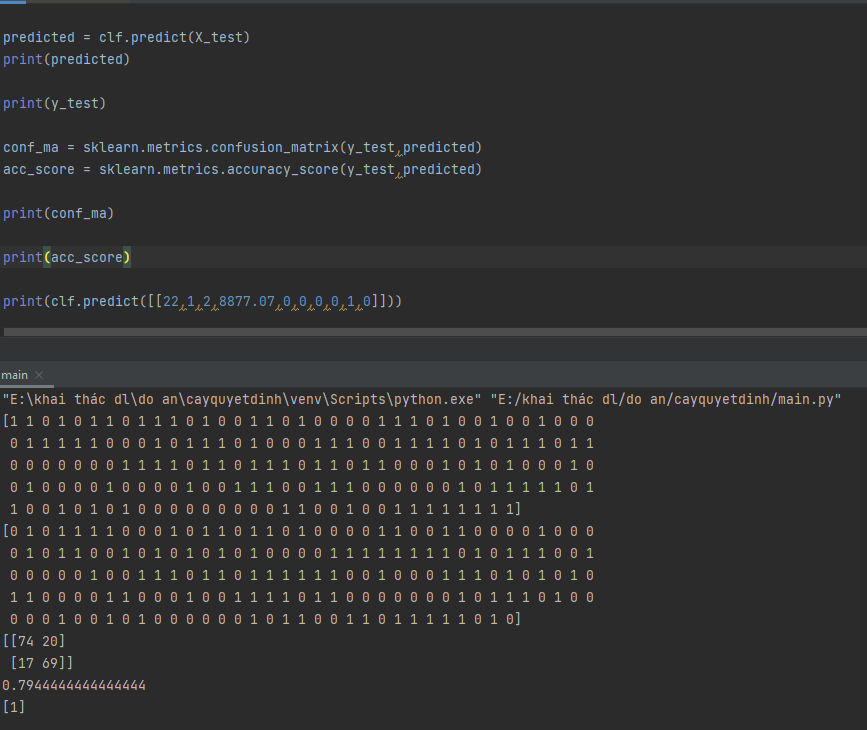
* Cài các thư viện cần thiết



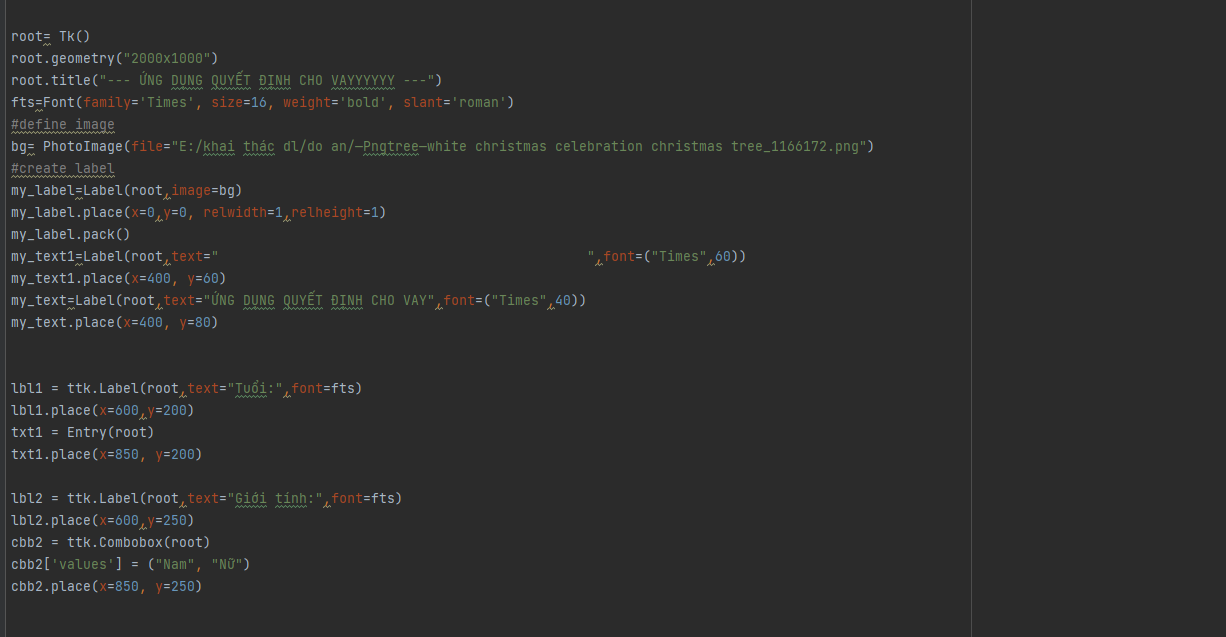
* Nạp file đã qua xử lý dữ liệu theo đường dẫn cục bộ trên máy
* Train model



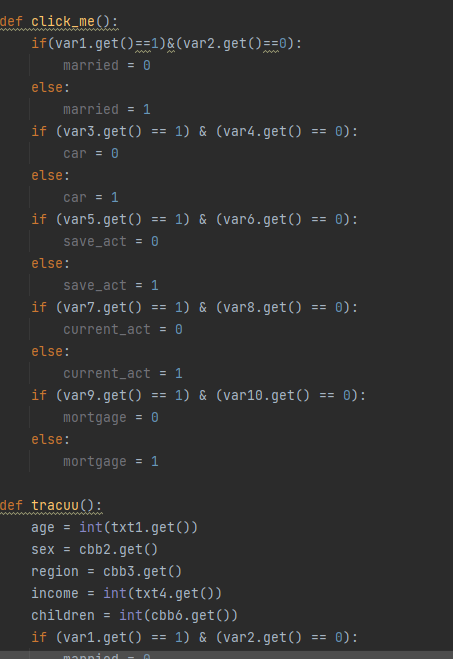
* Sử dụng model để test
* Kết quả

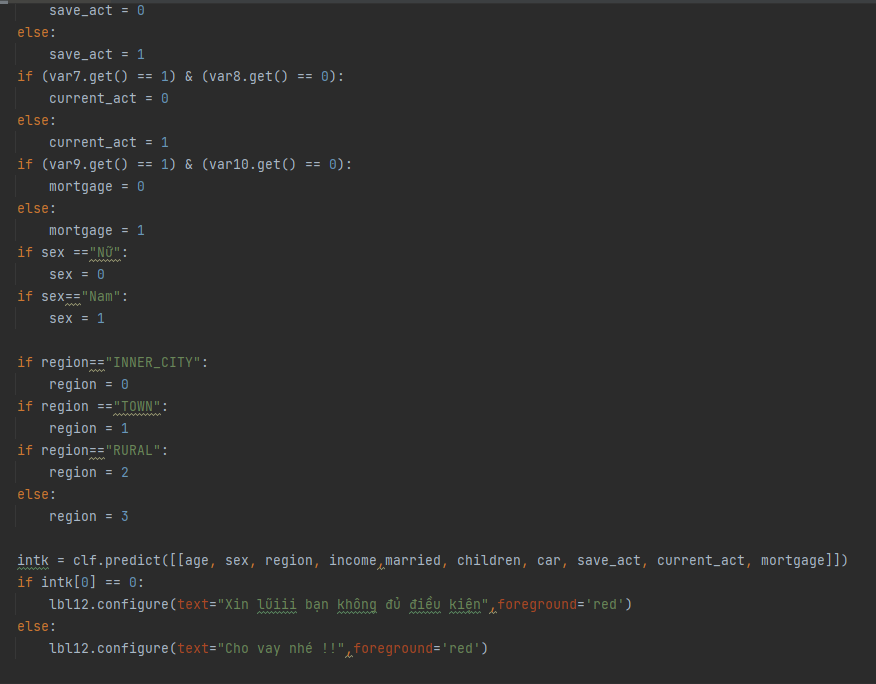


* Giao diện



* Hàm xử lý để lấy dữ liệu





# Mục tham khảo

<https://www.youtube.com/watch?v=TEHnfsqpUPc>

<https://vimentor.com/vi/lesson/tien-xu-ly-du-lieu-trong-linh-vuc-hoc-may-phan-1>