

**MÔN: CHUYÊN ĐỀ HỆ THỐNG THÔNG TIN**

**DỰ ĐOÁN HỦY ĐẶT PHÒNG KHÁCH SẠN**

**GVHD: TS Nguyễn Văn Tới**

**THÀNH VIÊN NHÓM STT 01**

1. **Hoàng Mậu Trung – B16DCCN370**
2. **Nguyễn Lan Anh – B16DCCN009**
3. **Nguyễn Ngọc Mai – B16DCCN225**
4. **Hoàng Đức Hải - B16DCCN122**
5. **Lê Nguyễn Ngọc Việt – B16DCCN410**
6. **Nguyễn Ngọc Ba – B16DCCN529**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG.**

**Khoa Công Nghệ Thông Tin**

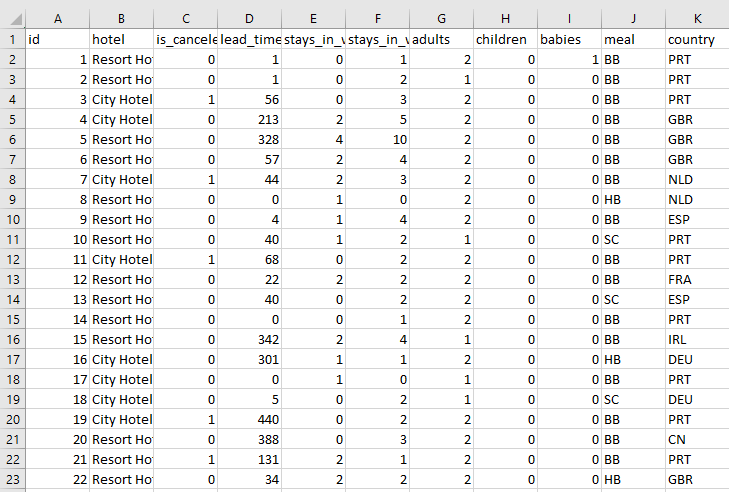
----------

1. **Đặt vấn đề bài toán**

* Dự đoán hủy đặt phòng, hãy dự đoán xem đặt phòng của Khách sạn Resort và Khách sạn City Hotel có bị hủy hay không?
* Đối với khách sạn nhà hàng, hủy đặt hẹn của khách hàng gây tổn thất lớn đến doanh số kinh doanh. Không chỉ giảm doanh thu, mà chi phí phân công và nguyên liệu để cung cấp một lịch hẹn cũng tổn thất.
* Bài toán yêu cầu đưa ra là phân loại những đặt phòng sử dụng bộ dữ liệu hơn 100 nghìn đặt phòng. Bộ dữ liệu sử dụng thông tin đặt phòng trên thực tế ở một số thành phố tại Bồ Đào Nha. Không chỉ dự đoán hủy đặt phòng đơn thuần, đây là một thử thách mang ý nghĩa đối với xã hội, có thể áp dụng rộng rãi, ví dụ như để phân tích nguyên nhân dẫn đến hủy đặt phòng trong tương lại.

1. **Bộ dữ liệu**

* Bộ dữ liệu bao gồm 40 nghìn dữ liệu của khách sạn Resort và 80 nghìn dữ liệu của khách sạn Hotel. Dữ liệu của khách sạn Resort thuộc về vùng Algarve, còn khách sạn City là của thủ đô Lisbon.
* Đây là bộ dữ liệu trích xuất từ hệ thống Quản lý khách sạn (PMS), vì thế tên khách sạn và đại lý đặt phòng sẽ được ẩn đi.



Hình ảnh trích xuất từ tập dữ liệu

Giải thích ý nghĩa từng cột thuộc tính

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên cột | Ý nghĩa | Phận loại dữ liệu |
| 1 | hotel | City Hotel hoặc Resort Hotel | Categorical data |
| 2 | is\_calceled | Việc đặt phòng có hủy không | Categorical data |
| 3 | lead\_time | Số ngày từ đặt đến khi nhận phòng | Numeric data |
| 4 | stays\_in  weekend\_nights | Số ngày đặt phòng và nghỉ lại (Thứ 7 và CN) | Numeric data |
| 5 | stays\_in  weekend\_nights | Số ngày đặt phòng và nghỉ lại (Thứ 2 – Thứ 6) | Numeric data |
| 6 | adults | Số lượng người lớn | Numeric data |
| 7 | children | Số trẻ em |  |
| 8 | babies | Số trẻ em | Numeric data |
| 9 | meal | Loại bữa ăn | Categorical data |
| 10 | country | Quốc gia | Categorical data |
| 11 | Market  segment | Phân loại thị trường | Categorical data |
| 12 | distribution  chanel | Kênh cung cấp | Categorical data |
| 13 | Is\_repeated  guest | Có phải khác gặp lại không? | Categorical data(binary) |
| 14 | Previous  cancellations | Đã hủy trước đây hay chưa. | Categorical data(binary) |
| 15 | Previous\_booking  not\_canceled | Chưa từng hủy phòng tước đây | Categorical data(binary) |
| 16 | Reserved room type | Loại đặt phòng | Categorical data |
| 17 | Assigned room type | Loại phòng sắp xếp | Categorical data |
| 18 | Booking changes | Số lần thay đổi phòng | Numeric data |
| 19 | agent | ẩn | Numeric data |
| 20 | Company | ẩn | Numeric data |
| 21 | Days in waitting list | Số ngày lấy xác nhận đặt phòng của khách hàng | Numeric data |
| 22 | Customer type | Loại khách hàng | Categorical data |
| 23 | adr | Tỉ giá trung bình ngày nghỉ trọ | Numeric data |
| 24 | Required car parking spaces | Yêu cầu bao nhiêu chỗ đỗ xe | Numeric data |
| 25 | Total of special requests | Số yêu cầu đặc biệt | Numeric data |
| 26 | Arrival date | Ngày đến | Date data |

1. **Phân tích dữ liệu và tiền xử lý dữ liệu**

Trong tập dữ liệu các thuộc tính thuộc 3 loại dữ liệu khách nhau gồm: numeric data (dữ liệu số), categorical data (dữ liệu phân loại) và date data (dữ liệu thời gian). Với mỗi loại dữ liệu thì có một cách xử lý khác nhau.

* Dữ liệu số (numeric data)

Dữ liệu số (numerical data) thường biểu thị dữ liệu dưới dạng các giá trị vô hướng mô tả quan sát, bản ghi hoặc thang đo. Chẳng hạn như tuổi là dạng dữ liệu số.

Đối với loại dữ liệu này thì mỗi cột, thuộc tính sẽ có tỉ lệ khác nhau, giá trị min – max cũng khác nhau. Chính lẻ đó một kĩ thuật tiền dữ lí dữ liệu tất cả các cột về một miền giá trị thông thường là (0,1) hoặc (-1,1). Điều này khiến cho sự bình đẳng của các cột với nhau trong các thuật toán hoặc máy và nó cũng không ảnh hưởng trong các nhóm thuật toán Tree Base Algorirthm.

Một phương pháp đơn giản chuẩn hóa một tập giá trị về khoảng giá trị [0,1] đó là chuẩn hóa Min-Max.

Chuẩn hóa Min-Max được thực hiện đơn giản thông qua công thức:

X = (1)

Trong sklearn phương thức này được triển khai: sklearn.preprocessing.MinMaxScaler

* Dữ liệu phân loại (categorical data)

Dữ liệu phân loại là dữ liệu mà giúp xác định loại của một thuộc tính bởi các giá trị cho trước. Chẳng hạn như thời tiết: mưa, gió, bão…. Chúng được xác định trước các giá trị của một thuộc tính, một sự vật được quan sát.

Đối với dữ liệu này thì có 2 dạng: dữ liệu phân loại có thứ tự và dữ liệu phân loại không có thứ tự.

Ví dụ dữ liệu có thứ tự như size quần áo: S, M, L, XL, XXL. Dữ liệu phân loại không có thứ tự như thể loại phim: phim cổ trang, phim tình cảm, phim hoạt hình, phim hành động. Sự khác biệt giữa hai loại dữ liệu này đó là tính thứ tự chúng ta biết size M thì kích thước lớn hơn size M. Nhưng thể loại phim thì không có ý nghĩa về mặt so sánh.

* Dữ liệu phân loại không có thứ tự

Một cách đơn giản để mô tả dữ liệu phân loại đó là mã hóa từng loại có trong quan sát thành các con số đặc trưng chẳng hạn như: phim cổ trang là 0, phim hành đồng là 1…

Phương thức mã hóa này được thực hiện trong: sklearn.preprocessing.LabelEncoder

* Dữ liệu phân loại có thứ tự

Dữ liệu có thứ tự chúng ta cũng hoàn toàn áp dụng các mã hóa như dữ liệu không có thứ tự tuy nhiên phương pháp kia có mặt hạn chết, chúng ta phải mã hóa chính xác và được thẩm định bằng con người. Chẳng hạn như 2 phép mã hóa size quần áo L, XL, XXL nếu được mã hóa 0,1,2 thì chúng chính xác, nhưng phép mã hóa chỉ cần điều chỉ thành 1,0,2 thì giá trị về mặt thứ tự không còn được bảo toàn.

* Dữ liệu thời gian (date data)

Thời gian là một dạng dữ liệu đặc biệt nó cung cấp một thông tin về mặt thời gian. Dữ liệu này cũng có hai dạng chính: Periodicity và Time Since.

Ví dụ về dữ liệu thời gian Periodicity: 12/10/1998, 12:58am, còn Time Since: ngày hôm qua, ngày mai, 200 năm trước, từ hôm sinh nhật.

Với loại dữ liệu Time Since thông thường nó có một cách duy nhất để xử lý đó là chuyên gia ngồi tự phân tích ước lượng thời gian cho nó. Còn đối với loại periodicity chúng ta có một khái niệm đó là diff\_time : chênh lệch thời gian so với một mốc xác định.

Điều này có nghĩa là thời gian đầu vào được tính chênh lệch thời gian với một mốc xác thời gian cố định, thông thường chúng ta chọn mốc thời gian bắt đầu trên hệ điều hành 0:00:00 01/01/1990.

Diff\_time = time - time\_root (2)

Ví dụ: Chúng ta tính cần mã hóa thời gian cho đầu vào ngày 6/7/2020

Mốc thời gian gốc được chọn là 1/7/2020 Ta được diff\_time = 5.

Từ đó mà dữ liệu thời gian hoàn toàn có thể chuyển về dữ liệu numeric data.

1. **Thuật toán áp dụng**

Trong khuôn khổ bài toán thì 4 thuật toán được lựa chọn để áp dụng cho tập dữ liệu đó là:

[1]: Logistic Regression

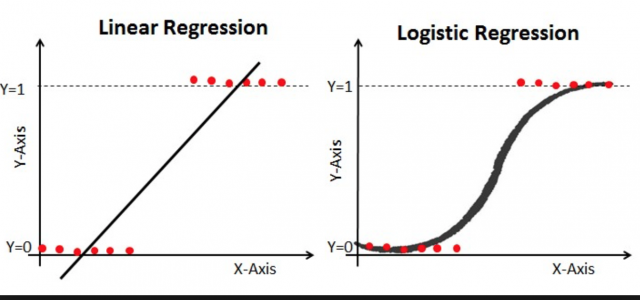
[2]: MLP Multi Layer Perceptrong

[3]: Decision Tree (CART, ID3)

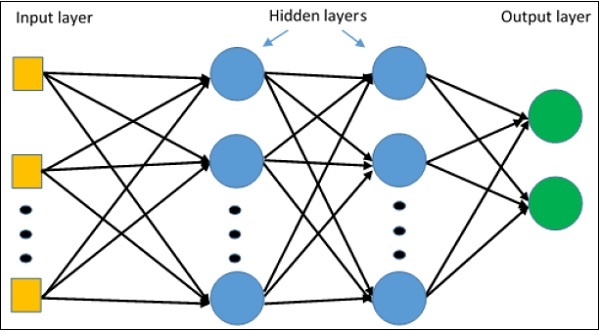
[4]: Random Forest

* Logistic Regression

Là một mô hình thống kê dạng cơ bản mà nó sử dụng một hàm logistic (là một đường cong hình chữ S) để mô hình biến phụ thuộc nhị phân, hàm logistic thông dụng là hàm sigmoid có miền giá trị thuộc [0,1]. Nó được sử dụng để xác xuất một lớp hoặc sự kiện nào đó dưới dạng nhị phân.

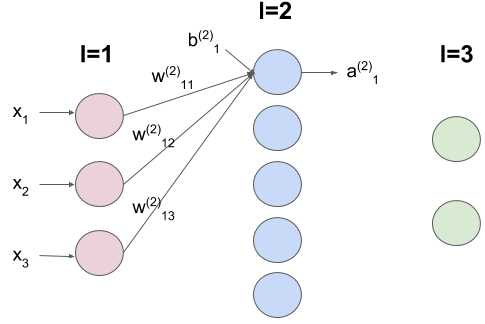


* Mutil Layer Perceptron (MLP)



Gọi là Multi-layer Perceptron (perceptron nhiều lớp) bởi vì nó là tập hợp của các perceptron chia làm nhiều nhóm, mỗi nhóm tương ứng với một layer. Trong hình trên ta có một ANN với 3 lớp: Input layer (lớp đầu vào), Output layer (lớp đầu ra) và Hidden layer (lớp ẩn). Thông thường khi giải quyết một bài toán ta chỉ quan tâm đến input và output của một model, do vậy trong MLP nói riêng và ANN nói chung ngoài lớp Input và Output ra thì các lớp neuron ở giữa được gọi chung là Hidden (ẩn không phải là không nhìn thấy mà đơn giản là không quan tâm đến).

Tư tưởng của MLP có thể được mô tả ngắn gọn như sau: Nó là sự mô tả sự lan truyền thông tin của các nơ-ron (các hình tròn trong hình). Sự truyển thông tin như các tầng khác nhau đến tầng cao hơn. Tầng cao hơn nhận được thông tin từ tầng thấp hơn và lan truyền đến tầng khác.



Trong hình vẽ trên mô tả nguyên lý hoạt động của một mạng nơ-ron đa tầng. Đầu vào là vector có 3 phần tử: [x1, x2, x3]. Thông tin đầu vào được truyển đến nơ-ron khác tại tần thứ 2 (I=2). Tại đó thông tin nhận được theo một phương trình tuyến tính:



Và cuối cùng trước khi đưa ra đầu ra nó được đưa ra qua một hàm activation (tương tự như thuật toán 1) nàm này giúp đưa ra xác xuất mà đầu vào mỗi nhóm phân loại.

Tại sao lại có sự lan truyền này?

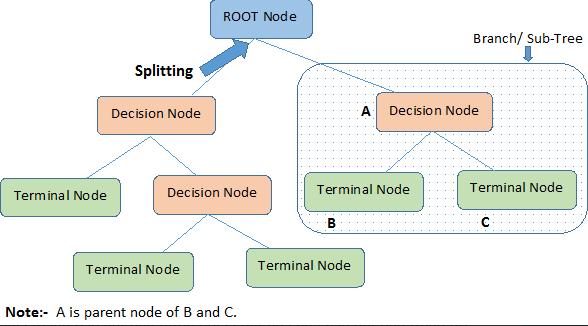
Điều này đúng đắn và được chứng minh cho việc học ánh xạ từ đầu vào thành đầu ra của tập dữ liệu. Trong quá trình huấn luyện thì các trọng số w sẽ được thay đổi và điều chỉnh sao cho việc ánh xạ đầu vào đạt hiệu quả nhất với việc đưa ra đầu ra. Điều này giống việc bạn thực hiện tìm hệ số a, b trong phương trình y = ax + b khi bạn biết x và y.

Thuật toán điều chỉnh trọng số w đó là Backpropagation.

* Decision Tree

Cây quyết định là một loại thuật toán học tập có giám sát được sử dụng nhiều trong bài toán phân loại.

Có 2 loại cây quyết định phụ thuộc vào mục tiếp áp dụng nó: mục tiêu phân loại với mục tiêu liêu tục ở biến đầu ra.



Tư tưởng của thuật toán cây quyết định đó là xây dựng một cây với các nhánh. Một cây quyết định là một đồ thị của các quyết định và các hậu quả có thể của nó. Cây quyết định được sử dụng để xây dựng một kế hoạch nhằm đạt được mục tiêu mong muốn. Các cây quyết định được dùng để hỗ trợ quá trình ra quyết định. Cây quyết định là một dạng đặc biệt của cấu trúc cây.

Xét bài toán: Có 30 học sinh gồm 3 biến đặc trưng cơ bản:x: giới tính (Nam/Nữ) y: lớp (IX, X) z: chiều cao (5-6 ft). Trong đó có 15/30 chơi mobile ledgend bang bang trong thời gian giải trí.

Mục đích chúng ta cần phân loại nhóm sinh viên ra 2 loại chơi game MLBB tron giờ giải trí ra khi biết các đắc trưng đầu vào x, y, z với từng học sinh.

Để giải bài toán này bằng cây quyết định ta phải thử lần lượt các thuộc tính:

Xét thuộc tính x = [Nam, Nu]

Với thuộc tính này chia tập dữ liệu thành 2 nhóm:

+ Nhóm 1: Toàn sinh viên Nam (20 sinh viên, 13 người chơi game)

+ Nhóm 2: Toàn sinh viên Nữ (10 sinh viên, 2 người chơi game)

Tương tự cho 2 loại thuộc tính như mỗi thuộc tính ta đc 2 nhóm

Thuộc tính y:

+ Nhóm 1: Học sinh lớp IX (14 người, 6 người chơi game)

+ Nhóm 2: Học sinh lớp X (16 người, 9 người chơi game)

Thuộc tính z:

+ Nhóm 1: Học sinh cao trên >= 5.5ft (18 người, 10 người chơi game)

+ Nhóm 2: Học sinh thấp hơn < 5.5ft (12 người, 5 người chơi game)

Vậy thì lựa chọn thuộc tính nào để phân loại? Tiêu chí nào đánh giá thuộc tính nào phân loại tốt hơn.

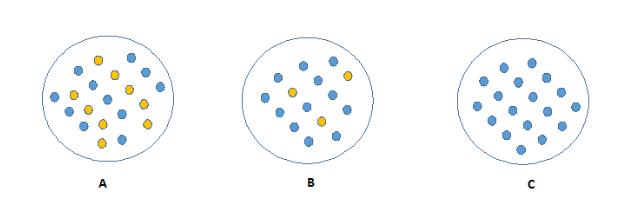
#### Gni (chỉ số Gni)

* Áp dụng cho phân chia cây dạng nhị phân
* Một tập bất kì khi ta chia 2 tập con từ tập gốc thì xác xuất mỗi tập được chia ra là như nhau và xác xuất khi tập đó đồng nhất (chỉ có 1 loại) là 1
* Chỉ số Gni càng cao thì chứng tỏ chỉ số đồng nhất càng cao tức mỗi tập đó gần giống nhau nhiều nhất

Cách tính chỉ số Gni:

Tính toán Gini cho các nút phụ, sử dụng tổng công thức bình phương xác suất thành công và thất bại (p ^ 2 + q ^ 2). Tính toán Gini cho phân chia bằng cách sử dụng điểm Gini có trọng số của từng nút của phân chia đó

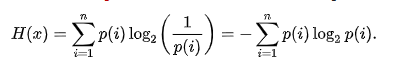
#### Entropy



Theo trực giác quan sát được hình sẽ ít tạp chất hơn 2 hình A và B. Do tập C có nhiều các points tương đồng. Từ đó ta suy ra được C cần ít thông tin hơn để mô tả nó.

Lý thuyết thông tin là một biện pháp để xác định mức độ vô tổ chức này trong một hệ thống được gọi là Entropy. Nếu mẫu hoàn toàn đồng nhất, thì entropy bằng 0 và nếu mẫu được chia đều

(50% - 50%) thì nó có entropy là 1. Công thức tính entropy là:



* Random Forest

Random Forest là một phương pháp học máy đa năng có khả năng thực hiện cả các nhiệm vụ hồi quy và phân loại. Nó cũng thực hiện các phương pháp giảm kích thước, xử lý các giá trị còn thiếu, giá trị ngoại lệ và các bước thiết yếu khác của khám phá dữ liệu và thực hiện công việc khá tốt. Nó là một loại phương pháp học tập đồng bộ, trong đó một nhóm các mô hình yếu kết hợp với nhau để tạo thành một mô hình mạnh mẽ.

Nó hoạt động ra sao?

Ý tưởng: Đối với thuật toán cây quyết định chúng ta chỉ dùng một cây duy nhất cho tác vụ phân loại hoặc hồi quy, trong rừng ngẫu nhiên chúng ta trồng nhiều cây một cách ngẫu nhiên. Mỗi cây đưa ra một phân loại và nói rằng cây đã đưa ra 1 phiếu bầu cho lớp đó và tổng số phiếu bầu lớp nào cao nhất là đầu ra dự đoán. Công nhận cũng hay.

Vậy trồng cây thế nào?

1. Gỉa sử có n mẫu dữ liệu cho huấn luyện, ta lấy ngẫu nhiên tập con trong số n mẫu huấn luyện đã cho và có thay thế mẫu con trong quá trình huấn luyện.
2. Gỉa sử có M giá trị đầu vào, ta sử dụng m giá trị (m < M) để xây dựng cây như cây quyết định
3. Mỗi cây được trồng kích thước lớn nhất và không cắt tỉa.
4. Dự đoán là tổng hợp dữ đoán của tất cả các cây.

Ưu điểm của rừng ngẫu nhiên?

* Thuật toán này giải quyết được cả vấn đề hồi quy và phân loại.
* Một trong những lợi ích của Rừng ngẫu nhiên làm tôi phấn khích nhất là, sức mạnh xử lý tập dữ liệu lớn với tính chiều cao hơn. Nó có thể xử lý hàng ngàn biến đầu vào và xác định hầu hết các biến quan trọng để nó được coi là một trong những phương pháp giảm kích thước.
* Nó có một phương pháp hiệu quả để ước tính dữ liệu bị thiếu và duy trì độ chính xác khi thiếu một tỷ lệ lớn dữ liệu.
* Nó có các phương thức để cân bằng các lỗi trong các tập dữ liệu trong đó các lớp bị mất cân bằng. Các khả năng của ở trên có thể được mở rộng thành dữ liệu chưa được gắn nhãn, dẫn đến phân cụm không được giám sát, xem dữ liệu và phát hiện ngoại lệ.
* Random Forest bao gồm lấy mẫu dữ liệu đầu vào với sự thay thế được gọi là lấy mẫu bootstrap. Ở đây một phần ba dữ liệu không được sử dụng cho đào tạo và có thể được sử dụng để kiểm tra. Chúng được gọi là mẫu ra khỏi túi. Lỗi ước tính trên các mẫu trong túi này được gọi là lỗi túi. Nghiên cứu ước tính lỗi bằng Out of bag, đưa ra bằng chứng cho thấy ước tính hết túi là chính xác như sử dụng một bộ kiểm tra có cùng kích thước với tập huấn luyện. Do đó, việc sử dụng ước tính lỗi xuất túi sẽ loại bỏ sự cần thiết của một bộ kiểm tra dành riêng.

Nhược điểm:

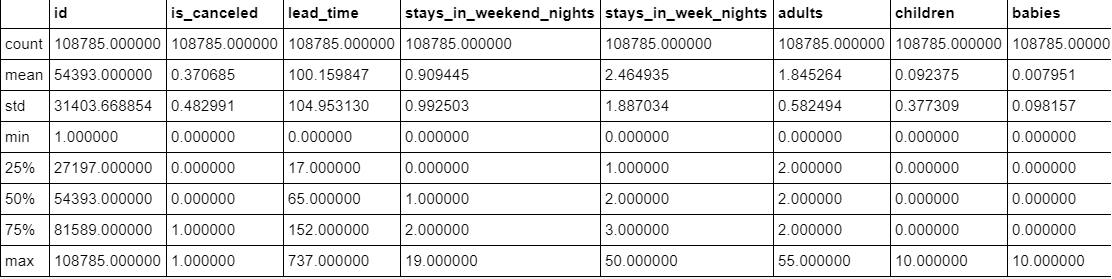
Rất tốt cho nhiệm vụ phân loại, không tốt cho nhiệm vụ hồi quy.

1. **Cài đặt**

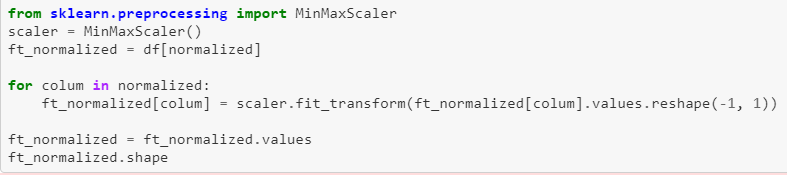
* **Phân loại các dạng dữ liệu của các cột**

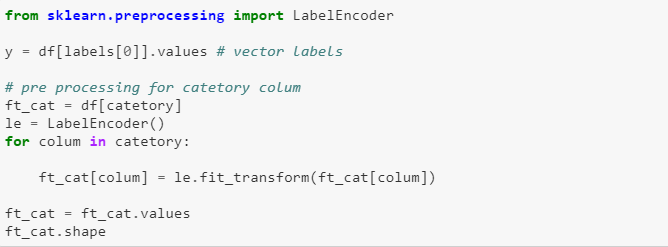
****

* **Quan sát dữ liệu**

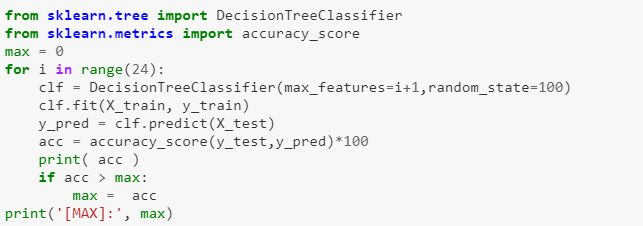
****

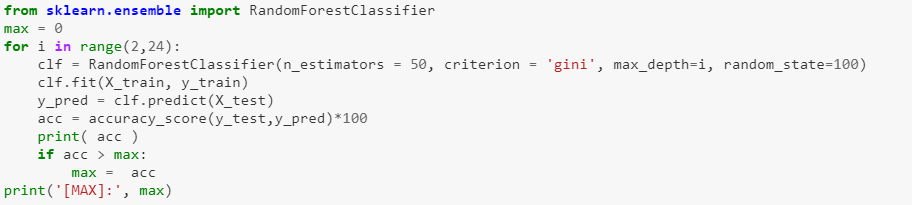
* **Chuẩn hóa dữ liệu**

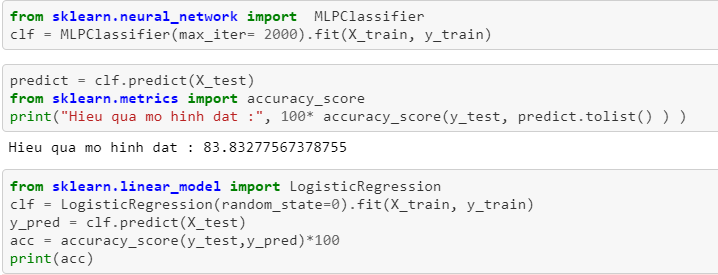
****

****

* **Thử nghiệm mô hình**

****

****

****

1. **Đánh giá kết quả và nhận xét**

Dữ liệu được phân chi tập train và test rõ ràng, kết quả huấn luyện mô hình được thử nghiệm với nhiều tham số Learning Rate, Số hiden layer hay số đặc trưng cho việc xây dựng cây quyết định, rừng ngẫu nhiên.

Bảng kết quả thử nghiệm thuật toán

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Tên thuật toán** | **Hiệu quả (acc)** | **Ghi chú** |
| 1 | Logistic Regression | 78,47% |  |
| 2 | MLP Classifier | 83,83% |  |
| 3 | Decision Tree CART | 86,11% |  |
| 4 | Decision Tree I3D | 86,26% |  |
| **5** | **Random Forest (CART)** | **89,42%** | **Best Acc** |
| 6 | Random Forest (I3D) | 89,39% |  |

Nhận xét:

* Sự đóng góp hiệu quả của mạng rơn cho tác vụ phân loại phức tạp đã được công nhận đối với các dữ liệu phức tạp như âm thanh, hình ảnh hay text. Tuy nhiên với dạng dữ liệu dạng thống kê và tính hiệu năng thì mô hình dựa trên cây có chiếm ưu thế.
* Sức mạnh thuật toán machine learning cơ bản có thể so sánh được với thuật toán machine learning của dựa trên tư tưởng mạng nơ-ron đối với trường hợp dữ liệu đơn giản và đem lại tốc độ cũng như dễ dạng sử dụng hơn.
* Code được triển khai tại: <https://github.com/ngoctuhan/CHUY-N-HTTT/blob/master/Untitled.ipynb>

1. Phân công công việc và đánh giá công việc thành viên

Nhóm trưởng: Hoàng Mậu Trung

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thành viên** | **Phân công công việc** | **Nhận xét** | **Điểm** |
| Nguyễn Lan Anh | Tìm hiểu thử nghiệm thuật toán MLP | Hoàn thành nhiệm vụ đúng thời gian  Năng nổ, quan tâm bài tập  Tham gia đầy đủ buổi họp nhóm | 8.5 |
| Nguyễn Ngọc Mai | Tìm hiểu thử nghiệm thuật toán Logistic Regression | Hoàn thành đầy đủ các nhiệm vụ được giao  Tham gia đầy đủ buổi họp nhóm  Chưa hòa đồng và năng nổ | 7.5 |
| Hoàng Đức Hải | Tìm hiểu thử nghiệm thuật toán CART | Hoàn thành các nhiệm vụ được giao  Hòa đồng, tích cực đóng góp ý kiến | 8 |
| Lê Nguyễn Ngọc Việt | Tìm hiểu thử nghiệm thuật toán  I3D | Hoàn thành nhiệm vụ được giao | 7.5 |
| Nguyễn Ngọc Ba | Các kĩ thuật tiền xử lí dữ liệu | Không tham gia đầy đủ các buổi trao đổi.  Chưa hoàn thiện công việc | 5 |