# 

[1. GIỚI THIỆU 2](#_Toc482643872)

[2. Một số phương pháp học máy 3](#_Toc482643873)

[2.1. Phương pháp học không giám sát (Unsupervised Learning) 3](#_Toc482643874)

[2.1.1. Khái niệm học không giám sát 3](#_Toc482643875)

[2.1.2. Nội dung phương pháp học không giám sát 4](#_Toc482643876)

[2.1.3. Một số thuật toán học không giám sát 4](#_Toc482643877)

[2.2. Phương pháp học bán giám sát (Semi-Supervised Learning) 4](#_Toc482643878)

[2.2.1 Khái niệm học bán giám sát 4](#_Toc482643879)

[2.2.2. Nội dung phương pháp học bán giám sát 5](#_Toc482643880)

[2.2.3. Một số thuật toán học bán giám sát 5](#_Toc482643881)

[2.3. Phương pháp học có giám sát (Supervised Learning) 6](#_Toc482643882)

[2.3.1. Khái niệm học có giám sát: 6](#_Toc482643883)

[2.3.2. Nội dung phương pháp học có giám sát: 6](#_Toc482643884)

[2.3.3. Một số thuật toán học có giám sát: 6](#_Toc482643885)

[3. PHÂN LỚP BẰNG PHƯƠNG PHÁP QUY NẠP CÂY QUYẾT ĐỊNH 7](#_Toc482643886)

[3.1. KHÁI NIỆM CÂY QUYẾT ĐỊNH: 7](#_Toc482643887)

[3.2. TẠO CÂY QUYẾT ĐỊNH: 7](#_Toc482643888)

[4. SỬ DỤNG CÂY QUYẾT ĐỊNH: 9](#_Toc482643889)

[5. MỘT SỐ VẤN ĐỀ CẦN GIẢI QUYẾT TRONG VIỆC PHÂN LỚP DỮ LIỆU: 10](#_Toc482643890)

[6. GIỚI THIỆU THUẬT TOÁN C4.5: 15](#_Toc482643891)

[6.1 GIỚI THIỆU PHẦN MỀM: 18](#_Toc482643892)

[6.2 CÀI ĐẶT THUẬT TOÁN: 19](#_Toc482643893)

# **GIỚI THIỆU**

Học máy, có tài liệu gọi là Máy học, ([tiếng Anh](http://vi.wikipedia.org/wiki/Ti%E1%BA%BFng_Anh): *machine learning*) là một lĩnh vực của [trí tuệ nhân tạo](http://vi.wikipedia.org/wiki/Tr%C3%AD_tu%E1%BB%87_nh%C3%A2n_t%E1%BA%A1o) liên quan đến việc nghiên cứu và xây dựng các kĩ thuật cho phép các hệ thống "học" tự động từ dữ liệu để giải quyết những vấn đề cụ thể. Ví dụ như các máy có thể "học" cách phân loại [thư điện tử](http://vi.wikipedia.org/wiki/Th%C6%B0_%C4%91i%E1%BB%87n_t%E1%BB%AD) xem có phải [thư rác (spam)](http://vi.wikipedia.org/wiki/Th%C6%B0_r%C3%A1c) hay không và tự động xếp thư vào thư mục tương ứng. Học máy rất gần với [suy diễn thống kê](http://vi.wikipedia.org/w/index.php?title=Suy_di%E1%BB%85n_th%E1%BB%91ng_k%C3%AA&action=edit&redlink=1) (statistical inference) tuy có khác nhau về thuật ngữ.

Học máy có liên quan lớn đến [thống kê](http://vi.wikipedia.org/wiki/Khoa_h%E1%BB%8Dc_Th%E1%BB%91ng_k%C3%AA), vì cả hai lĩnh vực đều nghiên cứu việc phân tích dữ liệu, nhưng khác với thống kê, học máy tập trung vào sự phức tạp của các giải thuật trong việc thực thi tính toán. Nhiều bài toán suy luận được xếp vào loại bài toán [NP-khó](http://vi.wikipedia.org/wiki/NP-kh%C3%B3), vì thế một phần của học máy là nghiên cứu sự phát triển các giải thuật suy luận xấp xỉ mà có thể xử lý được.

Học máy có hiện nay được áp dụng rộng rãi bao gồm [máy truy tìm dữ liệu](http://vi.wikipedia.org/wiki/M%C3%A1y_truy_t%C3%ACm_d%E1%BB%AF_li%E1%BB%87u), [chẩn đoán y khoa](http://vi.wikipedia.org/wiki/Ch%E1%BA%A9n_%C4%91o%C3%A1n), phát hiện [thẻ tín dụng giả](http://vi.wikipedia.org/w/index.php?title=Th%E1%BA%BB_t%C3%ADn_d%E1%BB%A5ng_gi%E1%BA%A3&action=edit&redlink=1), phân tích [thị trường chứng khoán](http://vi.wikipedia.org/wiki/Th%E1%BB%8B_tr%C6%B0%E1%BB%9Dng_ch%E1%BB%A9ng_kho%C3%A1n), phân loại các [chuỗi DNA](http://vi.wikipedia.org/w/index.php?title=Chu%E1%BB%97i_DNA&action=edit&redlink=1), [nhận dạng tiếng nói](http://vi.wikipedia.org/wiki/Nh%E1%BA%ADn_d%E1%BA%A1ng_ti%E1%BA%BFng_n%C3%B3i) và [chữ viết](http://vi.wikipedia.org/w/index.php?title=Nh%E1%BA%ADn_d%E1%BA%A1ng_ch%E1%BB%AF_vi%E1%BA%BFt_tay&action=edit&redlink=1), [dịch tự động](http://vi.wikipedia.org/wiki/D%E1%BB%8Bch_t%E1%BB%B1_%C4%91%E1%BB%99ng), [chơi trò chơi](http://vi.wikipedia.org/w/index.php?title=Tr%C3%B2_ch%C6%A1i_chi%E1%BA%BFn_l%C6%B0%E1%BB%A3c&action=edit&redlink=1) và [cử động rô-bốt](http://vi.wikipedia.org/w/index.php?title=C%E1%BB%AD_%C4%91%E1%BB%99ng_r%C3%B4-b%E1%BB%91t&action=edit&redlink=1) (*robot locomotion*).

Trong lĩnh vực máy học có nhiều phương pháp học khác nhau cụ thể (phương pháp học không giám sát, Phương pháp học bán giám sát, và Phương pháp học có giám sát ). Trong chuyên đề máy học tác giả chọn phương pháp học có giám sát sử dụng TSVM trong phân lớp dữ liệu trên web site từ đó xây dựng cây học quyết định C45 dựa trên dữu liệu đã được phân lớp và làm nền tảng cho dịch vụ điện toám đám mây.

**2. Một số phương pháp học máy**

Trong lĩnh vực học máy có nhiều phương pháp học khác nhau, trong phần này đề cập đến 3 phương pháp học được sử dụng phổ biến nhất, gồm có: học không giám sát, học bán/ nửa giám sát và học có giám sát.

**2.1. Phương pháp học không giám sát (Unsupervised Learning)**

**2.1.1. Khái niệm học không giám sát**

Học không giám sát là học với tập dữ liệu huấn luyện ban đầu hoàn toàn chưa được gán nhãn.

Học không giám sát là phương pháp học sử dụng cho lớp bài toán gom cụm, phân cụm (Clustering).

### 2.1.2. Nội dung phương pháp học không giám sát

- Để thực hiện phân cụm, trước tiên cần một tập dữ liệu huấn luyện (training dataset) – là một tập các ví dụ học (training examples/instances). Trong đó, mỗi ví dụ học chỉ chứa thông tin biểu diễn (ví dụ: một vector các giá trị thuộc tính), mà không có bất kỳ thông tin gì về nhãn lớp hoặc giá trị đầu ra mong muốn (expected output).

- Áp dụng một thuật toán học không có giám sát (ví dụ k-Means) để học hàm/mô hình mục tiêu (trong trường hợp này là hàm phân cụm ứng với thuật toán được chọn).

- Sử dụng một phương pháp thử nghiệm (có thể kết hợp với một tập dữ liệu có gán nhãn) để đánh giá hiệu năng/chất lượng của hàm mục tiêu học được.

### 2.1.3. Một số thuật toán học không giám sát

Có rất nhiều thuật toán học không giám sát được ra đời và phát triển nhằm giải quyết bài toán phân cụm phục vụ khai thác hiệu quả nguồn dữ liệu chưa gán nhãn nhiều và rất đa dạng. Việc lựa chọn sử dụng thuật toán nào tuỳ thuộc vào dữ liệu và mục đích của từng bài toán. Trong đó các thuật toán thường được sử dụng như: k-means, HAC (Hierarchical Agglomerative Clustering), SOM (Self-Organizing Map), DBSCAN, FCM,... (chi tiết các thuật toán này có thể tìm kiếm trên Internet)

## **2.2. Phương pháp học bán giám sát (Semi-Supervised Learning)**

Trong thực tế, để có được một tập dữ liệu có chất lượng và đã được gán nhãn của một lĩnh vực, thường được thực hiện thủ công bằng tay bởi người có nhiều kinh nghiệm về lĩnh vực đó. Vì vậy, dữ liệu đã được gán nhãn thường ít và đắt. Trong khi đó, dữ liệu chưa được gán nhãn lại rất nhiều và phong phú. Phương pháp học bán giám sát (hay học nửa giám sát) được đặt ra để tận dụng cả hai nguồn dữ liệu này.

### 2.2.1 Khái niệm học bán giám sát

Học bán giám sát là học với tập dữ liệu huấn luyện gồm cả dữ liệu đã được gán nhãn và dữ liệu chưa được gán nhãn.

Tuỳ vào từng mục đích cụ thể, học bán giám sát có thể được áp dụng cho bài toán phân lớp hoặc phân cụm.

### 2.2.2. Nội dung phương pháp học bán giám sát

**-** Nội dung chính của học bán giám sát là hệ thống sử dụng một tập học (training set) gồm 2 phần: các ví dụ học có nhãn, thường với số lượng (rất) ít, và các ví dụ học không có nhãn, thường với số lượng (rất) nhiều. Thực tế cho thấy khi sử dụng kết hợp dữ liệu không có nhãn với một lượng nhất định dữ liệu có nhãn có thể tăng độ chính xác đáng kể.

**-** Một thuật toán học bán giám sát được sử dụng (ví dụ Self-training) sẽ học các ví dụ có nhãn, sau đó tiến hành gán nhãn cho một số (có lựa chọn) các ví dụ không có nhãn - một cách hợp lý, có đánh giá chất lượng công việc hay độ chính xác. Tiếp theo, chọn các ví dụ vừa được gán nhãn có độ tin cậy cao (vượt trên một ngưỡng chọn trước) đưa vào kết hợp với tập dữ liệu có nhãn, tạo thành một tập dữ liệu huấn luyện mới.

**-** Áp dụng một phương pháp kiểm thử (có thể kết hợp với một tập dữ liệu đã biết trước nhãn) để đánh giá hiệu năng/độ chính xác của mô hình.

### 2.2.3. Một số thuật toán học bán giám sát

Một số thuật toán thường được sử dụng gồm có: thuật toán Cực đại kỳ vọng (EM - Expectation Maximization), SVM truyền dẫn (TSVM - Transductive Support Vector Machine), Self-training, Co-training và các phương pháp dựa trên đồ thị (graph-based).

Việc lựa chọn thuật toán nào dựa trên một số định hướng: nếu các lớp dữ liệu có tính phân cụm cao thì nên dùng EM với mô hình hỗn hợp sinh; nếu đã sử dụng SVM thì mở rộng thành TSVM; khi đó nâng cấp mô hình học có giám sát đã có, thì nên dùng self-training; nếu các đặc trưng của dữ liệu phân chia tự nhiên thành hai phần riêng rẽ thì nên dùng Co-training; còn nếu hai mẫu dữ liệu có đặc trưng tương tự nhau hướng tới một lớp thì sử dụng phương pháp dựa trên đồ thị.

## **2.3. Phương pháp học có giám sát (Supervised Learning)**

### 2.3.1. Khái niệm học có giám sát:

Học có giám sát là học với tập dữ liệu huấn luyện ban đầu hoàn toàn được gán nhãn từ trước.

Học có giám sát là phương pháp học sử dụng cho lớp bài toán phân lớp, phân loại (Classification).

### 2.3.2. Nội dung phương pháp học có giám sát:

**-** Để thực hiện phân lớp, trước tiên phải chuẩn bị một tập dữ liệu huấn luyện (trainning data set), để có tập dữ liệu huấn luyện phải thực hiện gán nhãn cho dữ liệu ban đầu, đây được gọi là quá trình thu thập tập huấn luyện.

**-** Lựa chọn một thuật toán phân lớp (ví dụ SVM) xây dựng bộ phân lớp để học tập dữ liệu huấn luyện. Hay nói cách khác, dùng tập dữ liệu huấn luyện để huấn luyện bộ phân lớp. Thuật ngữ học có giám sát được hiểu là học tập dữ liệu đã được gán nhãn trước (các dữ liệu kèm theo nhãn tương ứng này coi như đã được giám sát bởi người thực hiện gán nhãn).

**-** Sử dụng một tập dữ liệu kiểm tra (test data set) đã được gán nhãn trước, để kiểm tra tính đúng đắn của bộ phân lớp. Sau đó, có thể dùng bộ phân lớp để phân lớp cho các dữ liệu mới.

### 2.3.3. Một số thuật toán học có giám sát:

Một số thuật toán thường được lựa chọn khi xây dựng bộ phân lớp gồm có: máy vector hỗ trợ (Support Vector Machine – SVM); k láng giềng gần nhất (K Nearest Neighbours – KNN); tiếp cận xác suất thống kê (Naïve Bayes – NB); Cây quyết định (Decision Tree – DT); sử dụng mạng nơron (Neural Network – Nnet); dựa trên vector trọng tâm (Centroid–base vector); hay tuyến tính bình phương nhỏ nhất (Linear Least Square Fit – LLSF). (Chi tiết các thuật toán này có thể tham khảo trên Internet).

# **PHÂN LỚP BẰNG PHƯƠNG PHÁP QUY NẠP CÂY QUYẾT ĐỊNH**

* 1. **KHÁI NIỆM CÂY QUYẾT ĐỊNH:**

Cây quyết định là một cấu trúc phân cấp của các nút và các nhánh, nút bên trong được gán nhãn bằng các thuộc tính. Các nhánh bắt nguồn từ một nút có nhãn là thuộc tính A sẽ được gán nhãn bằng mỗi giá trị có thể có của thuộc tính A. Các nút lá của cây biểu diễn nhãn lớp hoặc sự phân bố của lớp. Để phân lớp một mẫu chưa biết chúng ta duyệt nó từ nút gốc đến nút lá, với mỗi thuộc tính bắt gặp nhánh tương ứng với giá trị của mẫu cho thuộc tính đó sẽ được đi theo cho đến khi gặp nút lá, phân lớp mẫu này tương ứng với nút lá đó sẽ được trả về.

* 1. **TẠO CÂY QUYẾT ĐỊNH:**

Việc tạo cây quyết định bao gồm 2 giai đoạn: Tạo cây và tỉa cây

* *Tạo cây:*ở thời điểm bắt đầu tất cả những mẫu huấn luyện đều ở gốc, sau đó phân chia mẫu dựa trên các thuộc tính được chọn, với việc kiểm tra các thuộc tính được chọn dựa trên một độ đo thống kê hoặc heuristic.
* *Tỉa cây:* là xác định và xóa những nhánh mà có phần tử hỗn loạn hoặc những phần tử nằm ngoài các lớp (những phần tử không thể phân vào một lớp nào đó*Ví dụ*: Tạo cây quyết định theo bảng dữ liệu D sau: *Bảng dữ liệu 2.1*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Stt** | **Age** | **Income** | **Student** | **Credit\_rating** | **Buy\_computer** |
| 1 | <=30 | High | No | Fair | No |
| 2 | <=30 | High | No | Fair | No |
| 3 | 31..40 | High | No | Fair | Yes |
| 4 | >40 | Medium | No | Fair | Yes |
| 5 | >40 | Low | Yes | Fair | Yes |
| 6 | >40 | Low | Yes | Excellent | No |
| 7 | 31..40 | Low | Yes | Excellent | Yes |
| 8 | <=30 | Medium | No | Fair | No |
| 9 | <=30 | Low | Yes | Fair | Yes |
| 10 | >40 | Medium | Yes | Fair | Yes |
| 11 | <=30 | Medium | Yes | Excellent | Yes |
| 12 | 31..40 | Medium | No | Excellent | Yes |
| 13 | 31..40 | High | Yes | Fair | Yes |
| 14 | >40 | Medium | No | Excellent | No |

Bảng dữ liệu này nhằm mô tả việc mua máy tính hay không dựa vào các thuộc tính tuổi (age), mức thu nhập (income), sinh viên (student), tỷ lệ tín dụng (credit\_rating) và thuộc tính nhãn lớp mô tả việc mua máy tính hay không (Buy\_Computer)

Cây quyết định thu được với bảng dữ liệu trên

Credit\_rating

Student

Age

yes

no

fair

Excellent

>40

31..40

<=30

*Hình 2.2:Biểu diễn cây quyết định*

1. **SỬ DỤNG CÂY QUYẾT ĐỊNH:**

Kiểm tra những giá trị thuộc tính của từng nút bắt đầu từ nút gốc của cây quyết định. Từ các nhánh chứa các giá trị của thuộc tính, ta tìm lần đến một phân lớp cuối cùng và từ đây ta có thể suy ra các luật tương ứng để mô tả cho quá trình khám phá tri thức từ các mẫu dữ liệu.

1. Mỗi một đường dẫn từ gốc đến lá trong cây tạo thành một luật.
2. Mỗi cặp giá trị thuộc tính trên một đường dẫn tạo nên một sự liên kết.
3. Nút lá giữ quyết định phân lớp dự đoán.
4. Các luật tạo được dễ hiểu hơn các cây

*Ví dụ:* Suy ra các luật tương ứng theo cây quyết định có được từ hình 2.2 như sau:

Nếu (Age<=30) và (Student=yes) Thì buy\_computer=yes

Nếu (Age từ 31..40) Thì buy\_computer=yes

Nếu (Age >40) và (Credit\_rating=fair) Thì buy\_computer=yes

* **Thuật toán quy nạp cây quyết định:**

Quy nạp là kiểu suy luận cho phép rút ra các luật tổng quát từ các tri thức trong một tập hữu hạn các mẫu.

Cây quyết định là mô hình biểu diễn cho kết quả quy nạp trên một tập mẫu.

*- Giải thuật cơ bản (giải thuật tham lam) được chia thành các bước như sau:*

1. Cây được xây dựng đệ quy từ trên xuống dưới (top - down) và theo cách thức chia để trị.
2. Ở thời điểm bắt đầu, tất cả những mẫu huấn luyện ở gốc.
3. Thuộc tính được phân loại (nếu là giá trị liên tục chúng được rời rạc hóa).
4. Những mẫu huấn luyện được phân chia đệ quy dựa trên thuộc tính mà nó chọn lựa.

Kiểm tra những thuộc tính được chọn dựa trên nền tảng của heuristic hoặc một định lượng thống kê.

1. **MỘT SỐ VẤN ĐỀ CẦN GIẢI QUYẾT TRONG VIỆC PHÂN LỚP DỮ LIỆU:**

* **Việc chọn thuộc tính nào để phân chia các mẫu?**

Ta có thể chọn bất kỳ thuộc tính nào làm nút của cây, điều này có khả năng xuất hiện nhiều cây quyết định khác nhau cùng biểu diễn một tập mẫu, có cây xuất hiện nhiều nút hoặc cây đơn giản, điều quan trọng là chọn thuộc tính nào để có thể phân lớp tốt dữ liệu sau này, một cách trực quan là ta nên chọn thuộc tính có độ phân biệt cao lên gần với nút gốc của cây, tức là chọn thuộc tính cho cây quyết định nhỏ nhất theo các cách sau:

* Tiêu chuẩn phân chia: tạo ra các nhóm sao cho một lớp chiếm ưu thế trong từng nhóm.
* Thuộc tính được chọn là thuộc tính cho độ đo tốt nhất, có lợi nhất cho quá trình phân lớp.
* Độ đo để đánh giá chất lượng phân chia là độ đo đồng nhất.
* Entropy (Information Gain)
* Information Gain Ratio
* Gini Index
* X2 – số thống kê bảng ngẫu nhiên (ccontingency table statistic) G – thống kê (statistic)
* **Điều kiện để dừng việc phân chia:**

1. Tất cả những mẫu huấn luyện thuộc về cùng một lớp.
2. Không còn thuộc tính còn lại nào để phân chia tiếp.
3. Không còn mẫu nào còn lại.

Các thuật toán trên cây quyết định có nhiều dạng khác nhau như: ID3. C4.5, CART, CHAID,...điểm khác biệt chính của các thuật toán này là tiêu chuẩn phân chia thuộc tính, độ đo để chọn lựa.

Mỗi thuật toán có độ đo khác nhau để lựa chọn thuộc tính phân lớp, ở đây chúng ta trình bày về thuật toán C4.5 nên cần tìm hiểu độ lợi thông tin để chọn lựa thuộc tính phân lớp

* **Độ lợi thông tin (Information Gain) trong cây quyết định:**

Information Gain là đại lượng được sử dụng để lựa chọn thuộc tính có độ lợi thông tin lớn nhất để phân lớp dữ liệu. Giả sử cho P, N là hai lớp và S là tập dữ liệu chứa p phần tử của lớp P và n phần tử của lớp N. Khối lượng của thông tin cần để quyết định một mẫu tùy ý trong S thuộc về lớp P hoặc N được định nghĩa như sau:

 (công thức 2.3)

Giả sử rằng sử dụng thuộc tính A để phân hoạch tập hợp S thành những tập hợp {S1, S2, ... ,Sv}. Nếu Si chứa những pi mẫu của lớp P và ni mẫu của N, entropy hay thông tin mong đợi cần để phân lớp những đối tượng trong tất cả các cây con Si là:

 (công thức 2.4)

Độ lợi thông tin nhận được bởi việc phân nhánh trên thuộc tính A là:

** (công thức 2.5)

*Ví dụ*: Tính độ lợi thông tin theo bảng dữ liệu 2.1

Thừa nhận:

* Lớp P: Buy\_computer = “yes”
* Lớp N: Buy\_computer = “no”
* Thông tin cần thiết để phân lớp một mẫu được cho là:

= 0.940

Sau đó ta tính entropy (thông tin mong đợi cần để phân lớp những đối tượng trong tất cả các cây con) cho từng thuộc tính của bảng dữ liệu trên:



Do đó: Gain(Age)=I(9,5)-E(Age)=0.246

Tương tự:

Gain(Income)= 0.029

Gain(Student)= 0.151

Gain(credit\_rating)=0.048

Như vậy, độ lợi thông tin của thuộc tính tuổi là lớn nhất, ta chọn thuộc tính này để phân lớp dữ liệu.Ta nhận thấy độđo Gain có xu hướng chọn các thuộc tính có nhiều giá trị, tuy nhiên thuộc tính có nhiều giá trị không phải lúc nào cũng cho việc phân lớp tốt nhất, vì vậy ta cần chuẩn hóa độ đo Gain, việc chọn thuộc tính không chỉ dựa vào độ đo Gain mà còn phụ thuộc vào độ đo Gain Ration:

GainRatio(A) = Gain(A) / SplitInfo(A) (công thức 2.6)

Trong đó: SplitInfo(A) =  (*v*: số giá trị cùa thuộc tính A) ­­­­­­­­­là công thức tính thông tin để phân nhánh trên cây quyết định.

Đây là công thức tính độ đo Gain Ratio cho thuộc tính A trên cơ sở dữ liệu D, sau đó ta chọn thuộc tính nào có độ đo Gain Ratio lớn nhất để phân lớp theo thuộc tính đó.

Lấy ví dụ trên, ta tính độ đo SplitInfo và Gain Ratio cho các thuộc tính:

SplitInfo(Age) = 

Tương tự:

SplitInfo(Income) = 1.555

SplitInfo(Student) = 1

SplitInfo(Credit-rating) = 0.984

Do đó:

Gain Ratio (Age) = Gain(Age)/SplitInfo(Age) = 0.246 / 1.576 = 0.156

Gain Ratio(Income)= 0.019

Gain Ratio(Student)= 0.151

Gain Ratio(Credit\_rating) = 0.049

Độ đo Gain Ratio của thuộc tính Age có giá trị lớn nhất nên ta chọn thuộc tính này để phân lớp dữ liệu.

* **Vấn đề quá khớp trong phân lớp**

Cây được tạo ra có thể quá khớp với dữ liệu huấn luyện. Việc quá khớp có thể xảy ra trong những trường hợp sau đây:

* Khi có độ nhiễu của dữ liệu, một số có thể phản ánh sự dị thường vì những phần tử hỗn loạn hoặc những phần tử nằm ngoài phân lớp, dữ liệu huấn luyện quá ít tạo thành các giá trị tối đa cục bộ trong tìm kiếm tham lam, đôi lúc mỗi mẫu dữ liệu cho ta một khái niệm. Kết quả nhận được cây quá nhiều nhánh, thiếu chính xác đối với những mẫu chưa biết. Vì thế, trong thuật toán qui nạp chúng ta đã dùng các độ đo để chọn thuật tính phân lớp cây vừa đủ sâu và cho kết quả phân lớp tốt nhất. Tuy nhiên, ngay cả sử dụng các độ đo vẫn còn có quá nhiều các khái niệm nhất quán với không gian giả thuyết trên tập huấn luyện, có những trường hợp chỉ có vài mẫu dữ liệu cho một khái niệm, như vậy cây quyết định được kết hợp để phân lớp tất cả các trường hợp của tập huấn luyện một cách chính xác có thể thực hiện một cách nghèo nàn trên các mẫu mới mà đã không được sử dụng để xây dựng cây quyết định, ta nói đây là trường hợp quá khớp với dữ liệu huấn luyện bởi vì dữ liệu huấn luyện chỉ là một tập con của tất cả các mẫu trong kho dữ liệu. Vấn đề làm thế nào xây dựng được mô hình tốt nhất dự đoán cho các mẫu chưa biết.

Có hai cách tiếp cận để tránh quá khớp dữ liệu như sau:

* *Tỉa trước***:** cách tiếp cận này dùng để dừng sự tăng trưởng của cây sớm hơn trước khi nó tìm đến một điểm mà tại đó phân lớp hoàn hảo tập dữ liệu huấn luyện. Điều đó có nghĩa là không tiếp tục phân vùng một nút nếu điều này tạo kết quả ở dưới một ngưỡng theo một hệ đánh giá nhất định. Khi dừng lại một nút thì nút đó trở thành nút lá và nó có thể chứa hầu hết tần suất xuất hiện các lớp giữa các tập con của mẫu hoặc phân phối xác suất của toàn bộ mẫu. Khó khăn nhất trong việc tỉa trước là tạo ra một ngưỡng thích hợp để dừng việc phân chia tại một nút.
* *Tỉa sau***:** đây là cách tiếp cận phổ biến nhất, cho phép một cây tăng trưởng đầy đủ, sau đó ta mới tiến hành cắt tỉa bằng cách duyệt từ dưới lên. Tại mỗi nút trong của cây, ta tính tỉ lệ sai số kỳ vọng khi nó bị cắt bỏ và khi chưa cắt. Tỉ lệ sai số khi nó bị cắt được tính dựa vào hợp nhất các thể hiện ở các nhánh con của nó. Tỉ lệ sai số khi nó chưa bị cắt được tính theo tỉ lệ sai số ở mỗi nhánh kết hợp với trọng số của mỗi nhánh. Nếu việc cắt bỏ một nút dẫn đến tỉ lệ sai số trông đợi lớn hơn thì nút đó được giữ lại, ngược lại thì cắt bỏ. Nút bị cắt bỏ sẽ trở thành nút lá và nhãn lớp được thay bằng hầu hết tần suất xuất hiện giữa các lớp trong các nhánh tạo thành nó.
* **Vấn đề phân lớp cây quyết định trong cơ sở dữ liệu lớn:**

Sự phân lớp là một vấn đề cổ điển được nghiên cứu một cách mở rộng bởi những nhà thống kê và những nhà nghiên cứu máy học, chúng có tính co dãn vì vậy phân lớp các tập dữ liệu có hàng triệu mẫu và hàng trăm thuộc tính với tốc độ chấp nhận được.

Quy nạp cây quyết định được đánh giá cao trong khai phá dữ liệu lớn vì những nguyên nhân sau:

* Tốc độ học tương đối nhanh so với những phương pháp phân loại khác.
* Có thể hoán chuyển được thành những luật phân lớp đơn giản và dễ hiểu.
* Có thể sử dụng truy vấn SQL để truy xuất cơ sở dữ liệu.
* Sự chính xác phân lớp có thể so sánh được với những phương pháp khác.

Giải thuật cây quyết định là mô hình dạng cây. Không có giới hạn cho khối lượng dữ liệu đầu vào cũng như số lượng thuộc tính được gán vào giải thuật, khối lượng còn lớn cây càng lớn – càng rộng và càng sâu hơn.

1. **GIỚI THIỆU THUẬT TOÁN C4.5:**

Chuyên đề này mô tả hệ thống dự báo ra quyết định cho dịch vụ điện toán đám mây sử dụng thuật toán C4.5. Trong chuyên đề này nêu lên giải pháp ảo hóa mềm nhằm cung cấp đa dịch vụ trong các lĩnh vực kinh doanh, thương mại, giáo dục, y tế, du lịch,...vv, bằng các phương pháp thu thập, phân tích dữ liệu trên mạng internet thông qua máy học SVM, từ đó xây dựng cây học quyết định nhằm đưa ra dự báo cho người sử dụng thông qua thuật toán C4.5. Chuyên đề này chủ yếu làm nổi bật phương pháp và quyết định khác nhau trong hệ thống bằng các tiêu chí, chia nhỏ, cắt tỉa, xử lý và các tính năng khác trên thuật toán C4.5. Chuyên đề mô tả thuật toán C4.5 là tối ưu thông qua các tiêu chuẩn phân chia thuộc tính phân lớp trong đó bài báo sử dụng việc tính độ lợi thông tin (Information Gain) làm tiêu chi đánh giá cho thuật toán C4.5.

Nhiệm vụ của giải thuật C4.5 là học cây quyết định từ một tập các dữ liệu huấn luyện bằng cách xét từng thuộc tính của tập dữ liệu huấn luyện để tìm ra thuộc tính có độ lợi thông tin cao nhất và phân nhánh cho thuộc tính đó. Biểu diễn này cho phép chúng ta xác định phân loại một đối tượng bằng cách kiểm tra các giá trị của nó trên một số thuộc tính nào đó.

**Ý tưởng giải thuật C4.5 như sau:**

Đầu vào: Một tập hợp các mẫu huấn luyện. Mỗi mẫu huấn luyện bao gồm các thuộc tính mô tả một tình huống, hay một đối tượng nào đó, và một giá trị phân loại của nó.

Đầu ra: Cây quyết định có khả năng phân loại đúng đắn các mẫu huấn luyện trong tập dữ liệu rèn luyện, và phân loại đúng cho cả các bộ chưa gặp trong tương lai.

Giải thuật:

**Function induce\_tree** (tập\_mẫu\_huấn\_luyện, tập\_thuộc\_tính)

**begin**

**if** mọi mẫu trong tập\_mẫu\_huấn\_luyện đều nằm trong cùng một lớp **then**

**return** một nút lá được gán nhãn bởi lớp đó

**else if** tập\_thuộc\_tính là rỗng **then**

return nút lá được gán nhãn bởi tuyển của tất cả các lớp trong tập\_mẫu\_huấn\_luyện

**else**

**begin** chọn một thuộc tính P, lấy nó làm gốc cho cây hiện tại;

xóa P ra khỏi tập\_thuộc\_tính;

với mỗi giá trị V của P

**begin** tạo một nhánh của cây gán nhãn V;

Đặt vào phân\_vùngV các mẫu trong tập\_mẫu\_huấn\_luyện có giá trị V tại thuộc tính P; Gọi **induce\_tree**(phân\_vùngV, tập\_thuộc\_tính), gắn kết quả vào nhánh V

**end**

**end**

**end**

Để xây dựng cây quyết định, tại mỗi nút của cây thì thuật toán đều tính lượng thông tin nhận được trên các thuộc tính và chọn thuộc tính có lượng thông tin tốt nhất làm nút phân tách trên cây.

Information Gain là đại lượng được sử dụng để lựa chọn thuộc tính có độ lợi thông tin lớn nhất để phân lớp dữ liệu. Giả sử cho P, N là hai lớp và S là tập dữ liệu chứa p phần tử của lớp P và n phần tử của lớp N. Khối lượng thông tin cần để quyết định một mẫu tùy ý trong S thuộc về lớp P hoặc N được định nghĩa như sau:



Entropy là khái niệm để đo tính thuần nhất của một tập huấn luyện. Một tập huấn luyện là thuần nhất nếu như tất cả các phần tử của tập huấn luyện đều thuộc cùng một loại, hay nói cách khác tập huấn luyện này có độ pha trộn là thấp nhất.

Giả sử rằng sử dụng thuộc tính A để phân hoạch tập hợp S thành những tập hợp {S1, S2, ... ,Sv}. Nếu Si chứa những pi mẫu của lớp P và ni mẫu của N, entropy hay thông tin mong đợi cần để phân lớp những đối tượng trong tất cả các cây con Si là:



Độ lợi thông tin nhận được bởi việc phân nhánh trên thuộc tính A là:

**

Tuy nhiên thuộc tính có nhiều giá trị không phải lúc nào cũng cho việc phân lớp tốt nhất, vì vậy ta cần chuẩn hóa độ đo Gain.

Tính thông tin trung bình của từng thuộc tính, để hạn chế xu hướng chọn thuộc tính có nhiều giá trị, thông tin trung bình của thuộc tính A được tính như sau

SplitInfo(A) = 

Việc chọn thuộc tính để phân nhánh dựa vào độ đo Gain Ration được tính như sau:

GainRatio(A) = Gain(A) / SplitInfo(A)

* 1. **GIỚI THIỆU PHẦN MỀM:**

Ngành nghề ẩm thực ở Việt Nam đang có sự phát triển mạnh mẽ cùng với sự đổi mới và hội nhập của thế giới. Quy mô của các nhà hàng hiện nay đà và đang lớn dần lên, phù hợp với nhu cầu của xã hội. Chính sự phát triển đó mà việc quản lý nhà hàng cũng trở nên khó khăn và phức tạp hơn nếu như vẫn sử dụng phương thức quản lý trên giấy tờ. Trước đây, khi máy tính chưa được ứng dụng rộng rãi, các công việc Quản Lý Nhà Hàng đều được làm thủ công nên mất rất nhiều thời gian và tốn kém về nhân lực như tài chính. Ngày nay, với sự phát triển của công nghệ thông tin mà máy tính đã được sử dụng rộng rãi trong các cơ quan, nhà máy, trường học…. giúp cho công việc được tốt hơn. Việc sử dụng máy tính vào công tác Quản lý Nhà Hàng là một yêu cầu cần thiết nhằm xóa bỏ những phương thức lạc hậu, lỗi thời gây tốn kém về nhiều mặt.

Đối với việc kinh doanh trong lĩnh vực ẩm thực ngoài việc phải quảng bá giới thiệu cho mọi người biết về nhà hàng, còn phải có sự tổ chức quản lý các nghiệp vụ trong nhà hàng, điều đó góp phần làm cho nhà hàng có sự chuyên nghiệp, đáp ứng yêu cầu ngày càng khắt khe của khách hàng, nó cũng góp phần tạo sự rành mạch trong việc quản lý từng loại nghiệp vụ. Điều này cũng giúp nhà quản lý dễ dàng hơn trong việc quản lý, đồng thời họ có nhiều thời gian hơn để đưa ra những kế hoạch trong kinh doanh.

Không chỉ dừng lại ở việc giúp các nhà quản lý quản lý hệ thống của mình dễ dàng, mà nó cũng giúp cho các nhân viên người trực tiếp thao tác với phần mềm làm việc nhanh hơn, hiệu quả hơn, tránh nhầm lẫn trong việc tính toán nhập xấu hàng.

Trong đồ án em xây dựng là những giải pháp gợi ý món ăn cho khách hàng. Phần mềm này giúp khách hàng đưa ra được những lựa chọn hợp lý khi gọi món, phù hợp với giới tính và tình trạng sức khỏe của mỗi người.

* 1. **CÀI ĐẶT THUẬT TOÁN:**

Từ dữ liệu lưu trữ ta rút trích các mẫu dữ liệu theo bảng sau:

*Bảng 3.3 Một số mẫu dữ liệu trong cơ sở dữ liệu*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Monan | Gioitinh | tinhtrang | xuhuong | tinhhieuqua |
| 1 | Canh bắp hầm sườn non | Nữ | Bình thường | Tăng cân | Có |
| 2 | Cá hộp | Nam | Bình Thường | Giảm cân | Không |
| 3 | Cua rang tiêu | Nam | Bình Thường | Giữ cân | Có |
| 4 | Đậu hấp tôm thịt | Nam/Nữ | Bình Thường | Giảm cân | Có |
| 5 | Cháo gạo lứt, đậu đỏ và ngô | Nam/Nữ | Bệnh cao huyết áp | Tăng cân | Có |
| 6 | Súp bột mì và rong biển | Nam/Nữ | Bệnh cao huyết áp | Giữ cân | Có |
| 7 | Thịt chó | Nam | Bệnh cao huyết áp | Giữ cân | Không |
| 8 | Gà hầm khoai tây | Nam | Bệnh cao huyết áp | Giữ cân | Không |
| 9 | Khổ qua xào trứng | Nam/Nữ | Bệnh Viêm gan B | Tăng cân | Không |
| 10 | Lẩu lòng heo | Nam/Nữ | Bệnh Viêm gan B | Giữ cân | Không |
| 11 | Lòng lợn non luộc | Nam | Bệnh Viêm gan B | Giữ cân | Không |
| 12 | Bông cải luộc | Nam | Bệnh Viêm dạ dày | Giữ cân | Có |
| 13 | Canh bắp cải | Nam/Nữ | Bệnh Viêm dạ dày | Giữ cân | Có |
| 14 | Phở chua | Nữ | Bệnh Viêm dạ dày | Giảm cân | Không |
| 15 | Thịt trâu lúc lắc xào rau muống | Nữ | Bệnh Viêm dạ dày | Giữ cân | Không |
| 16 | Cá hồi kho tộ | Nữ | Bệnh thiếu máu | Giảm cân | Có |
| 17 | Salad cá hồi xông khói | Nam | Bệnh thiếu máu | Giữ cân | Có |
| 18 | Trứng hấp kiểu nhật | Nam/Nữ | Bệnh thiếu máu | Giữ cân | Không |
| 19 | Cá chép hầm đậu đỏ | Nam | Bệnh tiểu đường | Giữ cân | Có |
| 20 | Ba ba hầm bắp nếp | Nữ | Bệnh tiểu đường | Tăng cân | Có |
| 21 | Canh trai nấu hẹ | Nam | Bệnh tiểu đường | Tăng cân | Không |

Ta áp dụng tính độ đo GainRatio cho các thuộc tính theo bảng dữ liệu mẫu trên để xác định thuộc tính nào được chọn trong quá trình tạo cây quyết định.

Bộ mẫu dữ liệu của chúng ta có 02 miền giá trị {c, k} (c ứng với “Món đó có dùng được” và k ứng với “không dùng được”)

Trước tiên, ta tính lượng thông tin trên tất cả mẫu dữ liệu S theo bảng trên:

I(S) = - -  = 0.998

*Bảng 3.4 Tính Entropy cho thuộc tính Tình trạng*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tình trạng | ci | ki | I(ci,ki) |
| 1 | Bình thường | 3 | 1 | 0.811 |
| 2 | Bệnh cao huyết áp | 2 | 2 | 1 |
| 3 | Bệnh viêm gan B | 0 | 3 | 0 |
| 4 | Bệnh Viêm dạ dày | 2 | 2 | 1 |
| 5 | Bệnh thiếu máu | 2 | 1 | 0.918 |
| 6 | Bệnh tiểu đường | 2 | 1 | 0.918 |

Ta có:

E(Tình trạng) = \*I(c­­1,­k­­1) + \*I(c2,­­­­­­­­­­k­2) + \*I(c­3,­k­­3) + \*I(c­4,­k­­4)

+ \*I(c­5,­k­­5) + \*I(c­6,­k­­6) = 0,798

Trong đó:

I(c­­1,­k­­1) = - -  = 0.811

I(c2,­­­­­­­­­­k­2) = - -  = 1

I(c3,­­­­­­­­­­k­3) = - -  = 0

I(c­4,­k­­4) = - -  =1

I(c­5,­k­­5) = - -  =0.918

I(c­6,­k­­6) = - -  =0.918

Do đó:

Gain(Tình trạng) = I(S) – E(Tình trạng) = 0,988 – 0,798 = 0,19

Tính độ đo SplitInfo cho thuộc tính Tình trạng:

SplitInfo(Tình trạng) = 

Vậy ta tính được độ đo GainRatio cho thuộc tình trạng:

GainRatio(Tình trạng)= Gain(Tình trạng)/ SplitInfo(Tình trạng) =0,19/2.57=0.074

Tương tự:

*Bảng 3.5: Tính Entropy cho thuộc tính Giới tính*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Giới tính | ci | ki | I(ci,ki) |
| 1 | Nam | 4 | 5 | 0.991 |
| 2 | Nữ | 3 | 2 | 0.971 |
| 3 | Nam/Nữ | 4 | 3 | 0.985 |

E(Gioitinh) = \*I(c­­1,­k­­1) + \*I(c2,­­­­­­­­­­k­2) + \*I(c­3,­k­­3)= 0.984

Gain(Gioitinh) = 0.998 – 0.984 =0.014

SplitInfo(Gioitinh)= 

GainRatio(Gioitinh)=0.014/1.545=0.009

*Bảng 3.6: Tính Entropy cho thuộc tính Xu hướng*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Xu hướng | ci | ki | I(ci,ki) |
| 1 | Giảm cân | 2 | 2 | 1 |
| 2 | Tăng cân | 3 | 2 | 0.971 |
| 3 | Giữ cân | 6 | 6 | 1 |

E(*Xu hướng*) = \*I(c­­1,­k­­1) + \*I(c2,­­­­­­­­­­k­2) + \*I(c­3,­k­­3)=0,993

Gain(*Xu hướng*) = 0,998 – 0,993 = 0.005

SplitInfo(*Xu hướng*)=

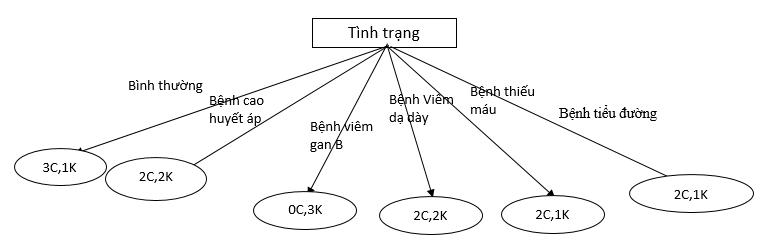
GainRatio(*Xu hướng*)= 0.005/1.41=**0.004**

*Bảng 3.7:Độ đo GainRatio của các thuộc tính được sắp xếp giảm dần*

|  |  |  |
| --- | --- | --- |
| STT | Thuộc tính | Độ đo GainRatio |
| 1 | Tình trạng | 0.074 |
| 2 | Xu hướng | 0.009 |
| 3 | Giới tính | 0.004 |

Như vậy thuộc tính có độ đo GainRatio lớn nhất là “Tình trạng”.

Cây phân nhánh theo thuôc tính “Tình trạng” có hình dạng như sau:



*Hình 3.3. Cây quyết định tại thuộc tính tình trạng*

**Nhận xét**:

Sau khi phân nhánh cây theo thuộc tính “Tình trạng”, ở các nút con vẫn chưa nút nào có tất cả các mẫu thuộc về một lớp. Vì vậy ta lập bảng dữ liệu phân theo giá trị tương ứng theo từng nút và tiếp tục phân nhánh cây quyết định theo từng nút này. Ứng với sức khỏe bình thường “Bình thường” có 03 phần tử có giá trị C (món dùng được) và 01 phần tử có giá trị K (không dùng được), với sức khỏe “Bệnh cao huyết áp” có 02 phần tử có giá trị C (món dung được) và 02 phần tử có giá trị K (không dung được), với sức khỏe “Bệnh tiểu đường” có 02 phần tử có giá trị C (món dung được) và 01phần tử có giá trị K (không dung được), với sức khỏe “Bệnh thiếu máu” có 02 phần tử có giá trị C (món dung được) và 01 phần tử có giá trị K (không dung được), với sức khỏe “Bệnh viêm dạ dày” có 02 phần tử có giá trị C (món dung được) và 02 phần tử có giá trị K (không dung được)

Tiếp tục áp dụng thuật toán C4.5 cho từng nút tương ứng bảng dữ liệu sau:

**Các mẫu dữ liệu có tình trạng “Bình thường”**

*Bảng3.8 Mẫu dữ liệu với thuộc tính Tình trạng có giá trị Bình thường(S1).*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Monan | Gioitinh | tinhtrang | xuhuong | tinhhieuqua |
| 1 | Canh bắp hầm sườn non | Nữ | Bình thường | Tăng cân | Có |
| 2 | Cá hộp | Nam | Bình Thường | Giảm cân | Không |
| 3 | Cua rang tiêu | Nam | Bình Thường | Giữ cân | Có |
| 4 | Đậu hấp tôm thịt | Nam/Nữ | Bình Thường | Giảm cân | Có |

Tính lượng thông tin trên tất cả mẫu dữ liệu S1:

I(S1) = - -  = 0.811

*Bảng 3.9: Xét thuộc tính Giới tính:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Giới tính | ci | ki | I(ci,ki) |
| 1 | Nam | 1 | 1 | 1 |
| 2 | Nữ | 1 | 0 | 0 |
| 3 | Nam/Nữ | 1 | 0 | 0 |

E(*Giới tính*) =\* I(c1,k1)+ \* I(c2,k2)+ \* I(c3,k3) = 0.5

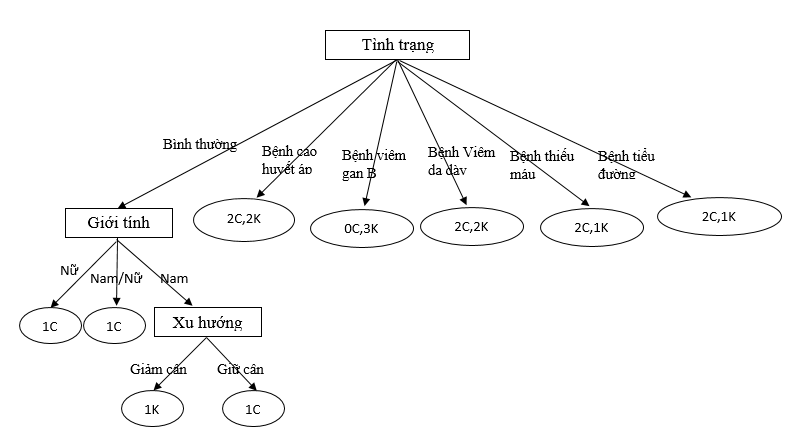
Gain(*Giới tính*) = 0.811 – 0.5 = 0.311

SplitInfo(*Giới tính*) = - - - = 0.5

GainRatio(*Giới tính*)=0.311/0.5 = 0.207

*Bảng 3.10: Xét thuộc tính Xu Hướng:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Xu hướng | ci | ki | I(ci,ki) |
| 1 | Giảm câm | 0 | 1 | 0 |
| 2 | Giữ cân | 1 | 0 | 0 |



**Các mẫu dữ liệu có tình trạng “Bệnh cao huyết áp”**

*Bảng3.11 Mẫu dữ liệu với thuộc tính Tình trạng có giá trị Bệnh cao huyết áp(S2).*

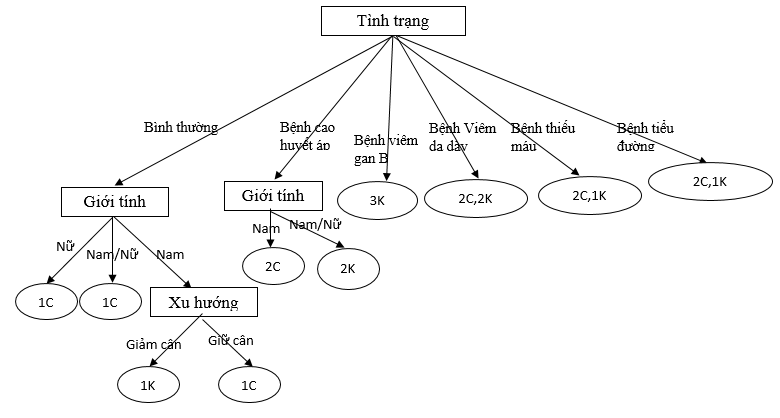
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Monan | Gioitinh | tinhtrang | xuhuong | tinhhieuqua |
| 1 | Cháo gạo lứt, đậu đỏ và ngô | Nam/Nữ | Bệnh cao huyết áp | Tăng cân | Có |
| 2 | Súp bột mì và rong biển | Nam/Nữ | Bệnh cao huyết áp | Giữ cân | Có |
| 3 | Thịt chó | Nam | Bệnh cao huyết áp | Giữ cân | Không |
| 4 | Gà hầm khoai tây | Nam | Bệnh cao huyết áp | Giữ cân | Không |

Tính lượng thông tin trên tất cả mẫu dữ liệu S2:

I(S2) = - -  = 1

*Bảng 3.12: Xét thuộc tính Giới tính:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Giới tính | ci | ki | I(ci,ki) |
| 1 | Nam | 2 | 0 | 0 |
| 2 | Nam/Nữ | 0 | 2 | 0 |



**Các mẫu dữ liệu có tình trạng “Bệnh viêm gan B”**

*Bảng 3.13: Mẫu dữ liệu với thuộc tính Tình trạng có giá trị Bệnh viêm gan B(S3).*

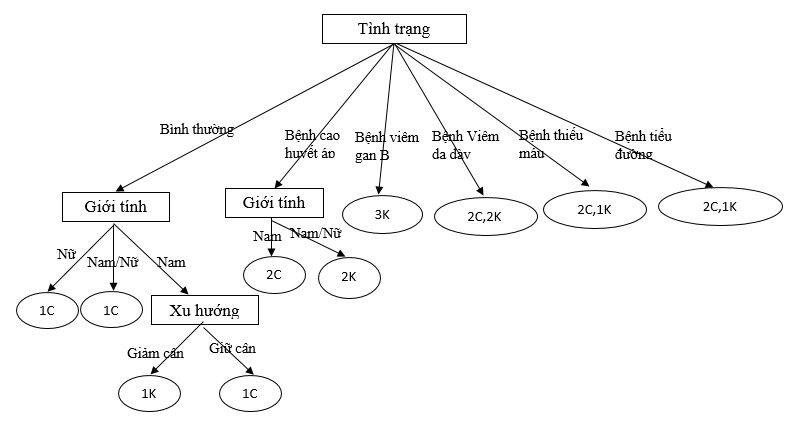
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Monan | Gioitinh | tinhtrang | xuhuong | tinhhieuqua |
| 9 | Khổ qua xào trứng | Nam/Nữ | Bệnh Viêm gan B | Tăng cân | Không |
| 10 | Lẩu lòng heo | Nam/Nữ | Bệnh Viêm gan B | Giữ cân | Không |
| 11 | Lòng lợn non luộc | Nam | Bệnh Viêm gan B | Giữ cân | Không |

Tính lượng thông tin trên tất cả mẫu dữ liệu S3:

I(S3) = - -  = 0.981

*Bảng 3.14: Xét thuộc tính Giới tính:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Giới tính | ci | ki | I(ci,ki) |
| 1 | Nam | 0 | 1 | 0 |
| 2 | Nam/Nữ | 0 | 2 | 0 |



**Các mẫu dữ liệu có tình trạng “Bệnh viêm dạ dày”**

*Bảng 3.15 Mẫu dữ liệu với thuộc tính Tình trạng có giá trị Bệnh viêm dạ dày(S4).*

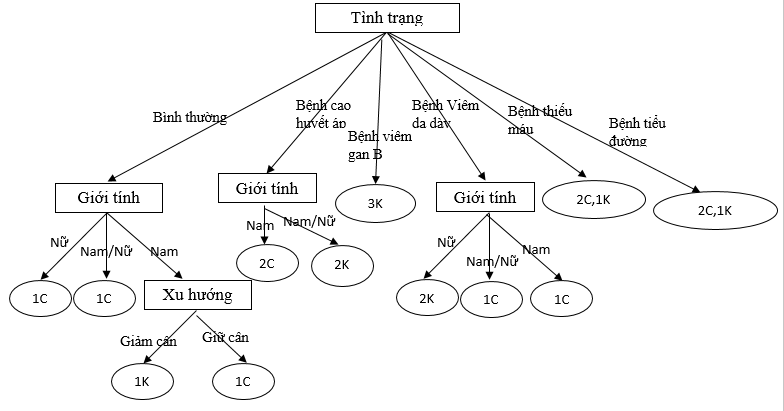
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Monan | Gioitinh | tinhtrang | xuhuong | tinhhieuqua |
| 1 | Bông cải luộc | Nam | Bệnh Viêm dạ dày | Giữ cân | Có |
| 2 | Canh bắp cải | Nam/Nữ | Bệnh Viêm dạ dày | Giữ cân | Có |
| 3 | Phở chua | Nữ | Bệnh Viêm dạ dày | Giảm cân | Không |
| 4 | Thịt trâu lúc lắc xào rau muống | Nữ | Bệnh Viêm dạ dày | Giữ cân | Không |

Tính lượng thông tin trên tất cả mẫu dữ liệu S4:

I(S4) = -- -  = 1.5

*Bảng 3.16: Xét thuộc tính Giới tính*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Giới tính | ci | ki | I(ci,ki) |
| 1 | Nam | 1 | 0 | 0 |
| 2 | Nữ | 0 | 2 | 0 |
| 3 | Nam/Nữ | 1 | 0 | 0 |



**Các mẫu dữ liệu có tình trạng “Bệnh thiếu máu”**

*Bảng3.17: Mẫu dữ liệu với thuộc tính Tình trạng có giá trị Bệnh thiếu máu(S5).*

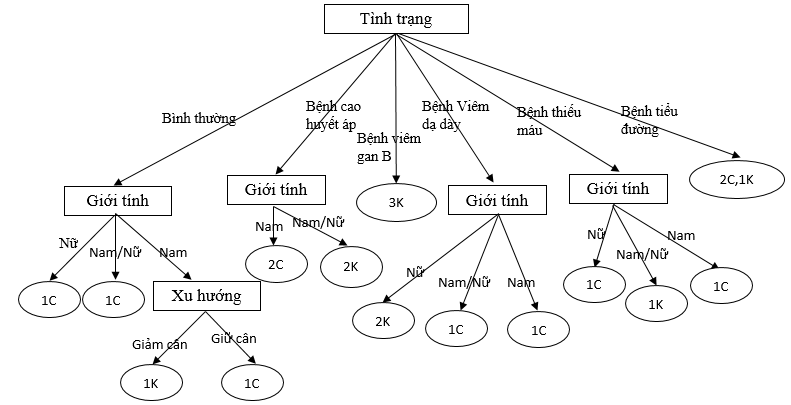
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Monan | Gioitinh | tinhtrang | xuhuong | tinhhieuqua |
| 16 | Cá hồi kho tộ | Nữ | Bệnh thiếu máu | Giảm cân | Có |
| 17 | Salad cá hồi xông khói | Nam | Bệnh thiếu máu | Giữ cân | Có |
| 18 | Trứng hấp kiểu nhật | Nam/Nữ | Bệnh thiếu máu | Giữ cân | Không |

Tính lượng thông tin trên tất cả mẫu dữ liệu S5:

I(S5) = -- -  = 1.585

*Bảng 3.18: Xét thuộc tính Giới tính*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Giới tính | ci | ki | I(ci,ki) |
| 1 | Nam | 1 | 0 | 0 |
| 2 | Nữ | 1 | 0 | 0 |
| 3 | Nam/Nữ | 0 | 1 | 0 |



**Các mẫu dữ liệu có tình trạng “Bệnh tiểu đường”**

*Bảng 3.19 Mẫu dữ liệu với thuộc tính Tình trạng có giá trị Bệnh thiếu máu(S6).*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stt | Monan | Gioitinh | tinhtrang | xuhuong | tinhhieuqua |
| 1 | Cá chép hầm đậu đỏ | Nam | Bệnh tiểu đường | Giữ cân | Có |
| 2 | Ba ba hầm bắp nếp | Nữ | Bệnh tiểu đường | Tăng cân | Có |
| 3 | Canh trai nấu hẹ | Nam | Bệnh tiểu đường | Tăng cân | Không |

Tính lượng thông tin trên tất cả mẫu dữ liệu S6:

I(S6) = -- = 1.585

*Bảng 3.20: Xét thuộc tính Giới tính:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Giới tính | ci | ki | I(ci,ki) |
| 1 | Nam | 1 | 1 | 1 |
| 2 | Nữ | 1 | 0 | 0 |

E(*Giới tính*) =\* I(c1,k1)+ \* I(c2,k2) = 0.667

Gain(*Giới tính*) = 1.585 – 0.667 = 0.918

SplitInfo(*Giới tính*) = - - = 0.918

GainRatio(*Giới tính*)=0.918/0.918 = 1

*Bảng 3.21: Xét thuộc tính Xu hướng*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Xu hướng | ci | ki | I(ci,ki) |
| 1 | Tăng cân | 0 | 1 | 0 |
| 2 | Giữ cân | 1 | 0 | 0 |

