Phân lớp dữ liệu

TS. Nguyễn Quốc Tuấn

Nội dung

- **Tổng quan về bài toán phân lớp dữ liệu**
- Phương pháp đánh giá độ chính xác
- Cây quyết định
- Bài tập

Tổng quan về phân lớp dữ liệu

name	*age*	*income*	*loan_decision*
Sandy Jones	youth	low	risky
Bill Lee	youth	low	risky
Caroline Fox	middle_aged	high	safe
Rick Field	middle_aged	low	risky
Susan Lake	senior	low	safe
Claire Phips	senior	medium	safe
Joe Smith	middle_aged	high	safe
...	...	...	...

RID	*age*	*income*	*student*	*credit_rating*	*Class: buys_computer*
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes

Phân lớp
(**classification**)

Risky or Safe?

Yes or No?

Tổng quan về phân lớp dữ liệu

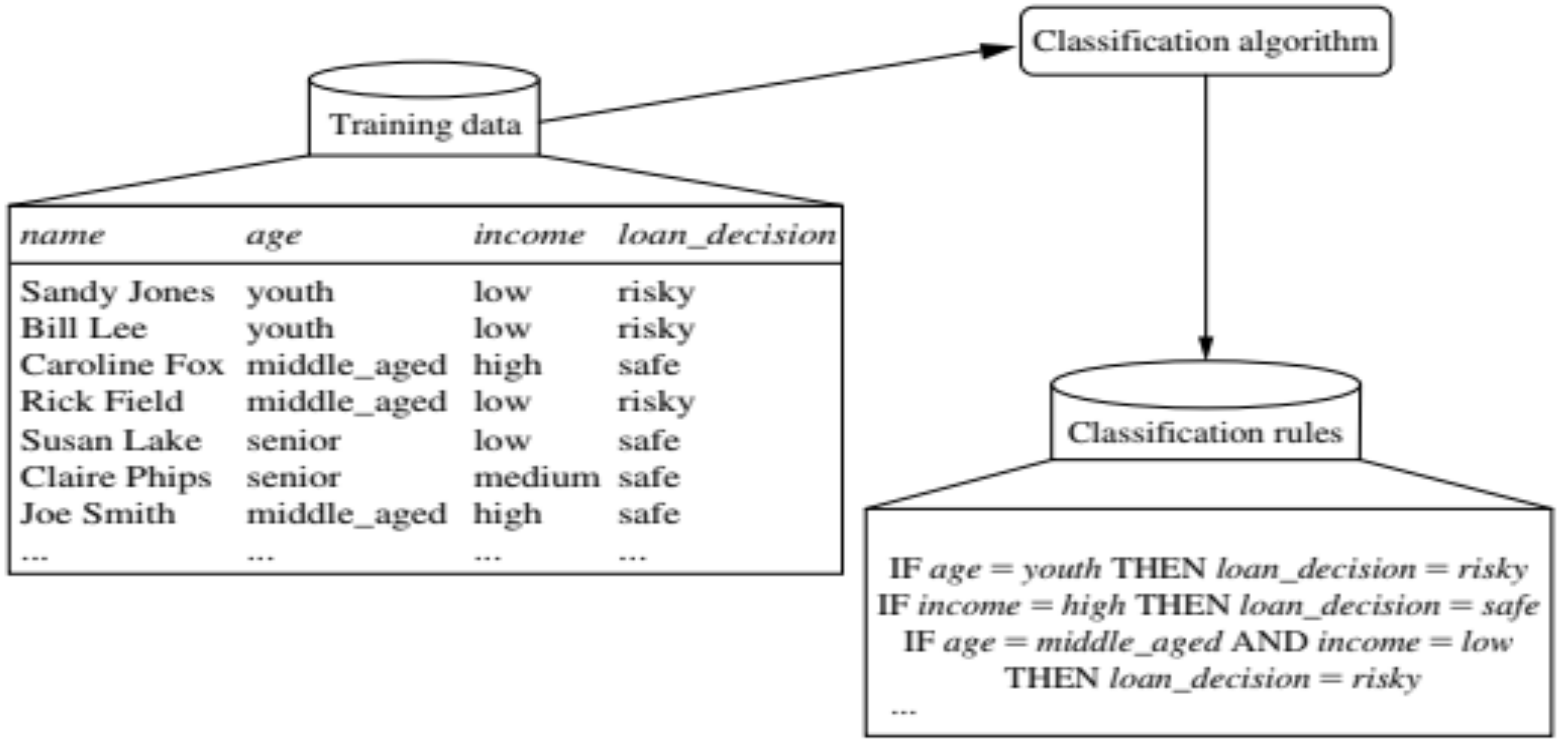
- Phân lớp dữ liệu là một dạng phân tích dữ liệu nhằm rút trích các mô hình mô tả các lớp dữ liệu quan trọng. Các mô hình như vậy được gọi là bộ Phân lớp (classifier) có nhiệm vụ dự đoán các nhãn lớp. Các nhãn lớp này thuộc loại rời rạc và thứ tự giữa các giá trị không có ý nghĩa.

Tổng quan về phân lớp dữ liệu

- Quá trình phân lớp dữ liệu gồm 2 bước:
  - Bước học (Training) : xây dựng bộ Phân lớp (classifier), quá trình học nhằm xây dựng một mô hình mô tả một tập các lớp dữ liệu hay các khái niệm định trước.
  - Bước phân lớp (Classification) : phân lớp dữ liệu/đối tượng mới nếu độ chính xác của bộ Phân lớp được đánh giá là có thể chấp nhận được (acceptable).

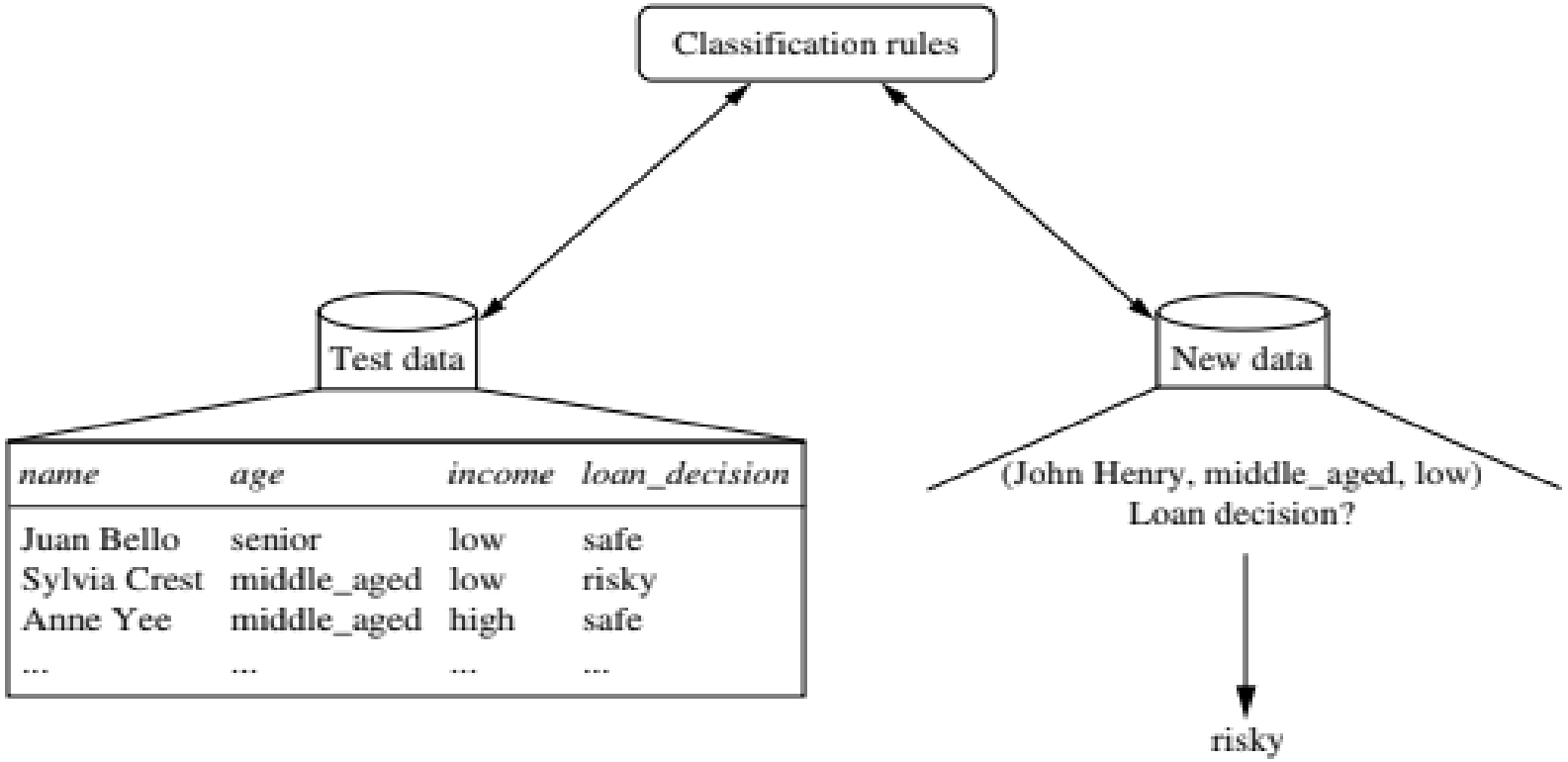
Tổng quan về phân lớp dữ liệu

- Training



Tổng quan về phân lớp dữ liệu

- Classification



Tổng quan về phân lớp dữ liệu

- Các thuật toán sử dụng trong phân lớp dữ liệu:
  - Phân lớp với cây quyết định (decision tree)
  - Phân lớp với mạng Bayesian
  - Phân lớp với mạng neural
  - Phân lớp với k phần tử cận gần nhất (k-nearest neighbor)
  - Phân lớp với suy diễn dựa trên tình huống (case based reasoning)
  - Phân lớp dựa trên giải thuật tiến hóa (genetic algorithms)
  - Phân lớp với lý thuyết tập thô (rough sets)
  - Phân lớp với lý thuyết tập mờ (fuzzy sets)

Tổng quan về phân lớp dữ liệu

- Một số ứng dụng phân lớp dữ liệu:
  - Tín dụng
  - Kinh doanh
  - Tiếp thị
  - Chuẩn đoán y khoa
  - Phân tích hiệu quả điều trị
  - Phân lớp văn bản
  - Lọc thư rác

Nội dung

- Tổng quan về bài toán phân lớp dữ liệu
- **Phương pháp đánh giá độ chính xác**
- Cây quyết định
- Bài tập

Các phương pháp đánh giá

- Hold-out (Splitting)
- Cross-validation
  - k-fold

Các phương pháp đánh giá

- Hold-out (Spliting)
  - Toàn bộ tập ví dụ D được chia thành 2 tập con không giao nhau
    - Tập huấn luyện D_train – để huấn luyện hệ thống
    - Tập kiểm thử D_test – để đánh giá hiệu năng của hệ thống đã học
    - D = D_train ∪ D_test, và thường là |D_train| >> |D_test|
  - Yêu cầu:
    - Bất kỳ dữ liệu nào thuộc vào tập kiểm thử D_test đều không được sử dụng trong quá trình huấn luyện hệ thống
    - Bất kỳ dữ liệu nào được sử dụng trong giai đoạn huấn luyện hệ thống đều không được sử dụng trong giai đoạn đánh giá hệ thống
    - Các lựa chọn thường gặp: |D_train|=(2/3).|D|, |D_test|=(1/3).|D|
  - Phù hợp khi ta có tập ví dụ D có kích thước lớn.

Các phương pháp đánh giá

- Cross-validation
  - Để tránh việc trùng lặp giữa các tập kiểm thử (một số ví dụ cùng xuất hiện trong các tập kiểm thử khác nhau)
  - k-fold cross-validation
    - Tập toàn bộ các dữ liệu D được chia thành k tập con không giao nhau (gọi là "fold") có kích thước xấp xỉ nhau
    - Mỗi lần (trong số k lần) lặp, một tập con được sử dụng làm tập kiểm thử, và (k-1) tập con còn lại được dùng làm tập huấn luyện
    - k giá trị lỗi (mỗi giá trị tương ứng với một fold) được tính trung bình cộng để thu được giá trị lỗi tổng thể
  - Các lựa chọn thông thường của k: 10, hoặc 5
  - Thông thường, mỗi tập con (fold) được lấy mẫu phân tầng (xấp xỉ phân bố lớp) trước khi áp dụng quá trình đánh giá Cross-validation
  - Phù hợp khi ta có tập ví dụ D vừa và nhỏ

Tính chính xác

- Giá trị (kết quả) đầu ra của hệ thống là một giá trị định danh

$$Accuracy = \frac{1}{|D\_test|} \sum_{x \in D\_test} Identical(o(x), c(x)); \qquad Identical(a,b) = \begin{cases} 1, \text{if} & (a = b) \\ 0, \text{if otherwise} \end{cases}$$

  - x là 1 ví dụ trong tập thử nghiệm (D_test); o(x) Giá trị đầu ra (phân lớp) bởi hệ thống đối với ví dụ x
  - c(x): Phân lớp thực sự (đúng) đối với ví dụ x

Ma trận nhầm lẫn (Confusion Matrix)

- ***TP$_i$***: Số lượng các ví dụ thuộc lớp c_i được phân loại chính xác vào lớp c_i
- ***FP$_i$***: Số lượng các ví dụ không thuộc lớp c_i bị phân loại nhầm vào lớp c_i
- ***TN$_i$***: Số lượng các ví dụ không thuộc lớp c_i được phân loại (chính xác)
- ***FN$_i$***: Số lượng các ví dụ thuộc lớp c_i bị phân loại nhầm (vào các lớp khác c_i)

Lớp c_i		Được phân lớp **bởi hệ thống**	
		Thuộc	Ko thuộc
Phân lớp **thực sự (đúng)**	Thuộc	**TP$_i$**	**FN$_i$**
	Ko thuộc	**FP$_i$**	**TN$_i$**

Precision và Recall

- **Accuracy** độ chính xác $$\text{Accuracy}(c_i) = \frac{TP_i + TN_i}{TP_i + FP_i + TN_i + FN_i}$$

- **Precision** đối với lớp c_i
  - Tổng số các ví dụ thuộc lớp c_i được Phân lớp chính xác chia cho tổng số các ví dụ được Phân lớp vào lớp c_i

$$Precision(c_i) = \frac{TP_i}{TP_i + FP_i}$$

- **Recall** đối với lớp c_i
  - Tổng số các ví dụ thuộc lớp c_i được Phân lớp chính xác chia cho tổng số các ví dụ thuộc lớp c_i

$$Recall(c_i) = \frac{TP_i}{TP_i + FN_i}$$

Nội dung

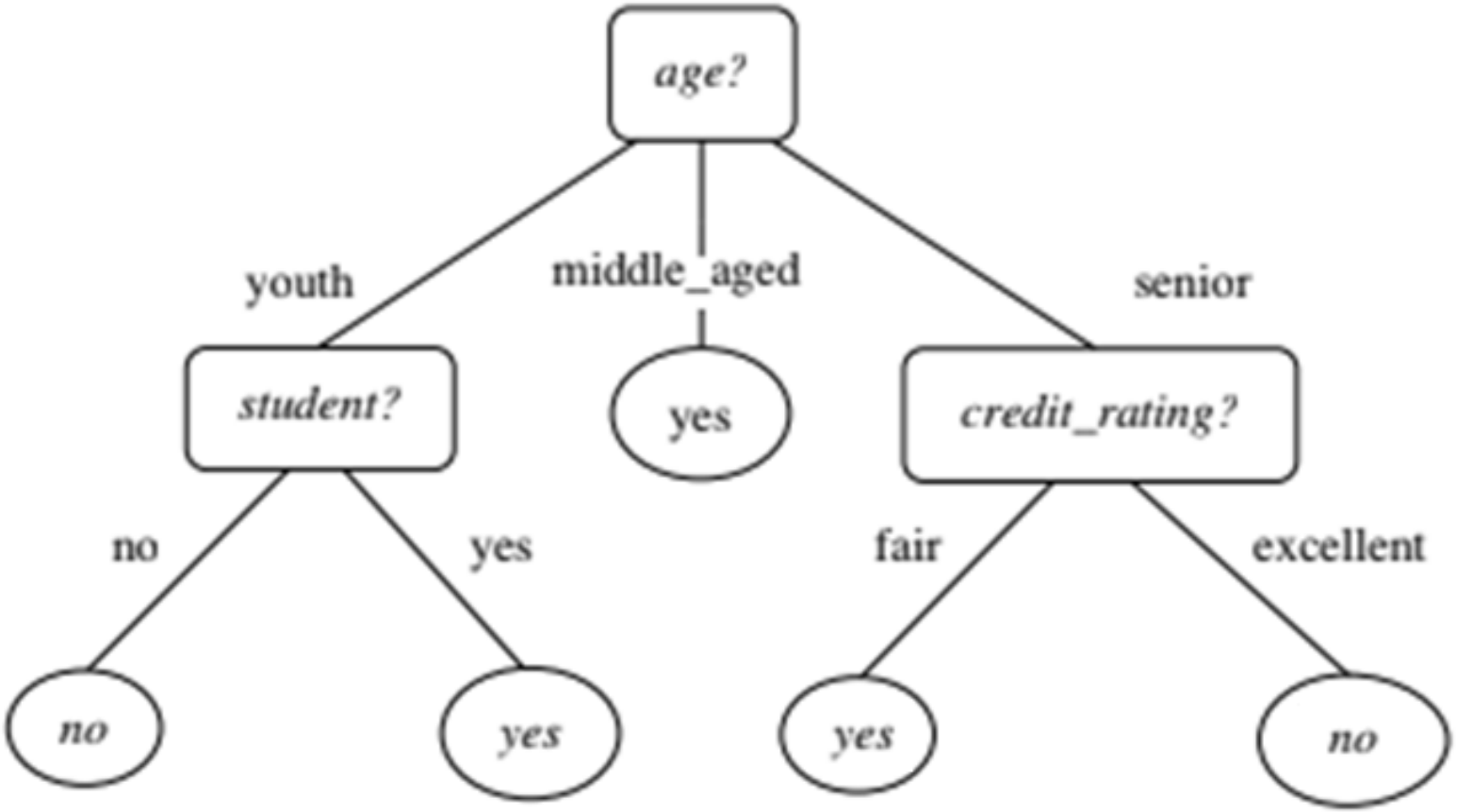
- Tổng quan về bài toán phân lớp dữ liệu
- Phương pháp đánh giá độ chính xác
- **Cây quyết định**
- Bài tập

Cây quyết định (Decision tree)

- Cây quyết định
  - Để học (xấp xỉ) một hàm mục tiêu có giá trị rời rạc – hàm phân lớp
  - Hàm phân lớp được biểu diễn bởi một cây quyết định
- Một cây quyết định có thể được biểu diễn (diễn giải) bằng một tập các luật IF-THEN (dễ đọc và dễ hiểu)
- Học cây quyết định có thể thực hiện ngay cả với các dữ liệu có chứa nhiễu/lỗi (noisy data)
- Là một trong các phương pháp học quy nạp (inductive learning) được dùng phổ biến nhất
- Được áp dụng thành công trong rất nhiều bài toán ứng dụng trong thực tế.

Cây quyết định (Decision tree)

- Ví dụ:



Cây quyết định (Decision tree)

- Biểu diễn cây quyết định:
  - Mỗi nút trong (internal node) biểu diễn một thuộc tính cần kiểm tra giá trị đối với các ví dụ
  - Mỗi nhánh (branch) từ một nút sẽ tương ứng với một giá trị có thể của thuộc tính gắn với nút đó
  - Mỗi nút lá (leaf node) biểu diễn một phân lớp
  - Một cây quyết định học được sẽ phân lớp đối với một ví dụ, bằng cách duyệt cây từ nút gốc đến một nút lá

    → Nhãn lớp gắn với nút lá đó sẽ được gán cho ví dụ cần phân lớp

Cây quyết định (Decision tree)

- Các giải thuật xây dựng cây quyết định
  - ID3 (Iterative Dichotomiser 3)
  - C4.5 (a successor of ID3)
  - CART (Classification and Regression Trees)

Thuật toán ID3 (Iterative Dichotomiser 3)

- Ross Quinlan
- Xây dựng một cây quyết định theo chiến lược top-down, bắt đầu từ nút gốc
- Ở mỗi nút, thuộc tính kiểm tra là thuộc tính có khả năng Phân lớp tốt nhất đối với các ví dụ học gắn với nút đó.
- Tạo mới một cây con (sub-tree) của nút hiện tại cho mỗi giá trị có thể của thuộc tính kiểm tra, và tập học sẽ được tách ra (thành các tập con) tương ứng với cây con vừa tạo
- Mỗi thuộc tính chỉ được phép xuất hiện tối đa 1 lần đối với bất kỳ một đường đi nào trong cây
- Quá trình phát triển cây quyết định sẽ tiếp tục cho đến khi:
  - Cây quyết định Phân lớp hoàn toàn các ví dụ học, hoặc
  - Tất cả các thuộc tính đã được sử dụng

Thuật toán ID3 (Iterative Dichotomiser 3)

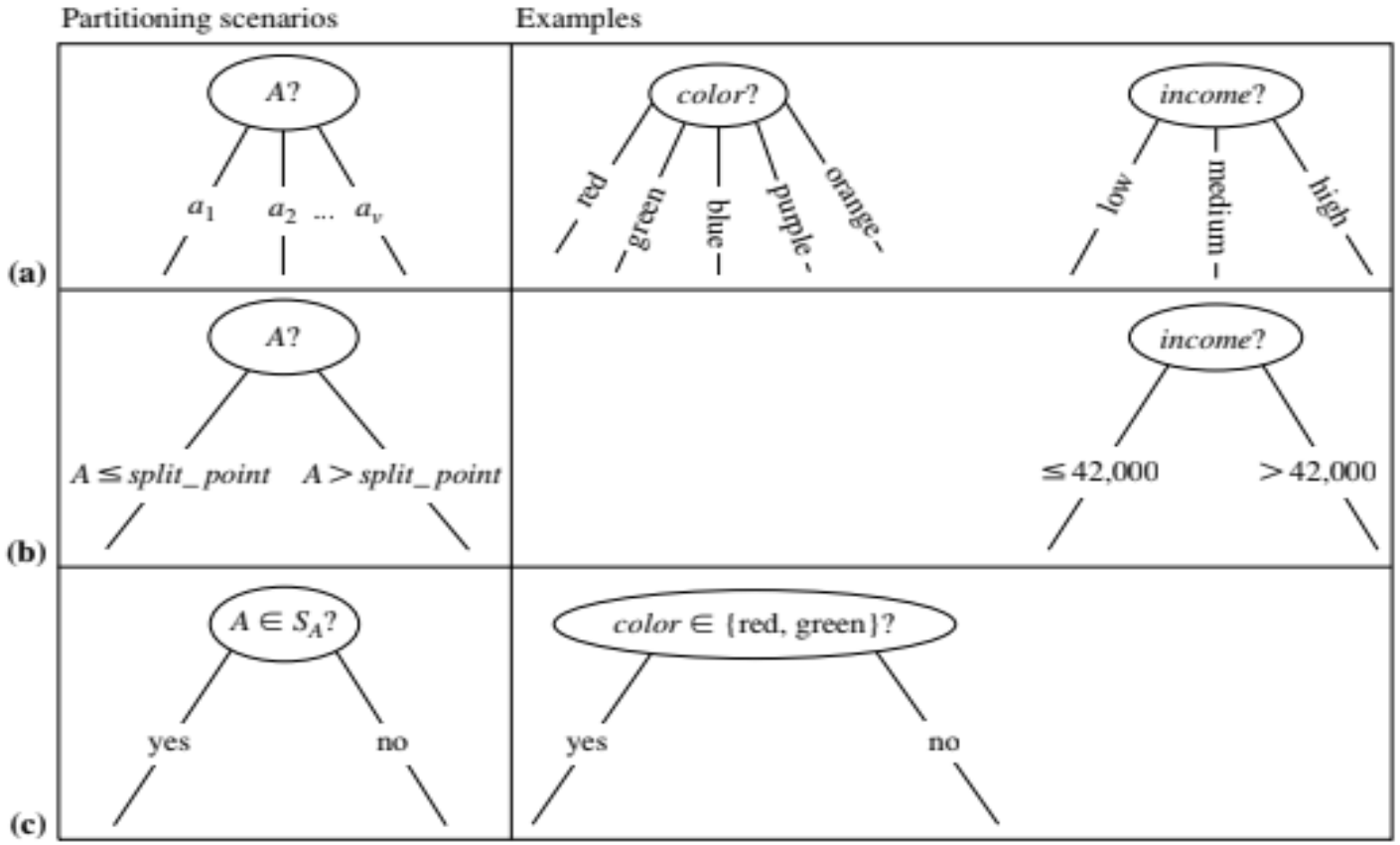
- Mô tả thuật toán

Method:

```
(1)  create a node N;
(2)  if tuples in D are all of the same class, C, then
(3)      return N as a leaf node labeled with the class C;
(4)  if attribute_list is empty then
(5)      return N as a leaf node labeled with the majority class in D; // majority voting
(6)  apply Attribute_selection_method(D, attribute_list) to find the "best" splitting_criterion;
(7)  label node N with splitting_criterion;
(8)  if splitting_attribute is discrete-valued and
         multiway splits allowed then // not restricted to binary trees
(9)      attribute_list ← attribute_list − splitting_attribute; // remove splitting_attribute
(10) for each outcome j of splitting_criterion
     // partition the tuples and grow subtrees for each partition
(11)     let Dj be the set of data tuples in D satisfying outcome j; // a partition
(12)     if Dj is empty then
(13)         attach a leaf labeled with the majority class in D to node N;
(14)     else attach the node returned by Generate_decision_tree(Dj, attribute_list) to node N;
     endfor
(15) return N;
```

Thuật toán ID3 (Iterative Dichotomiser 3)

- Lựa chọn thuộc tính phân lớp



Thuật toán ID3 (Iterative Dichotomiser 3)

- Entropy
  - Một đánh giá thường được sử dụng trong lý thuyết thông tin, sử dụng để đánh giá mức độ hỗ tạp của một tập.
  - Lượng thông tin cần để phân loại một phần tử trong D (Entropy của D): Info (D)

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

$$p_i = | C_{i,D} | / | D |$$

  - Trong đó :
    - D : là tập huấn luyện
    - p_i : xác suất để một phần tử bất kỳ trong D thuộc về lớp C_i với i = 1..m
    - $C_{i,D}$: tập các phần tử của lớp C_i trong D

Thuật toán ID3 (Iterative Dichotomiser 3)

- Entropy
  - Lượng thông tin cần để phân loại một phần tử trong D dựa trên thuộc tính A: $Info_A(D)$
    - Thuộc tính A dùng phân tách D thành v phân hoạch $\{D_1, D_2, \ldots, D_j, \ldots, D_v\}$.
    - Mỗi phân hoạch D_j gồm $|D_j|$ phần tử trong D.
    - $Info_A(D)$ càng nhỏ càng tốt.

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} * Info(D_j)$$

Thuật toán ID3 (Iterative Dichotomiser 3)

- Information Gain
  - Là độ sai biệt giữa trị thông tin Info(D) ban đầu (trước phân hoạch) và trị thông tin mới InfoA(D) (sau phân hoạch với A).

$$Gain(A) = Info(D) - Info_A(D)$$

  - Thuộc tính được chọn để phân lớp là thuộc tính có giá trị Information Gain cao nhất.

RID	age	income	student	credit_rating	Class: buys_computer
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

Thuật toán ID3 (Iterative Dichotomiser 3)

- Tính Entropy
  - $|D| = 14$
  - $m = 2$
  - C_1 = yes
  - C_2 = no
  - $|C_{1,D}| = 9$
  - $|C_{2,D}| = 5$

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

$$p_i = |C_{i,D}| / |D|$$

$$Info(D) = -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right) = 0.940 \text{ bits.}$$

Thuật toán ID3 (Iterative Dichotomiser 3)

- Tính $Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} * Info(D_j)$
  - Thuộc tính age ={“youth”, “middle_aged”, “senior”}

$$Info_{age}(D) = \frac{5}{14} \times \left(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}\right)$$
$$+ \frac{4}{14} \times \left(-\frac{4}{4}\log_2\frac{4}{4}\right)$$
$$+ \frac{5}{14} \times \left(-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}\right)$$
$$= 0.694 \text{ bits.}$$

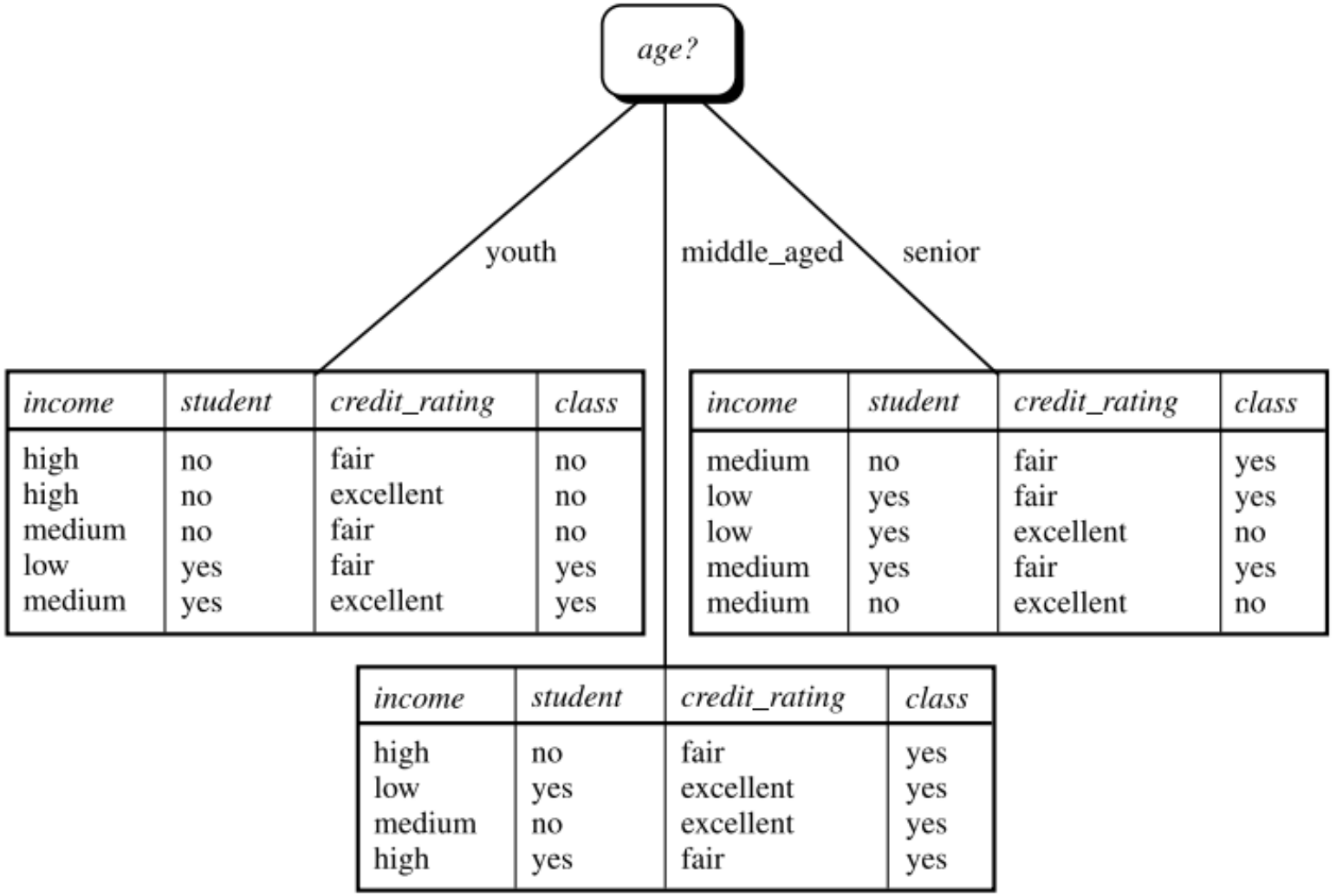
Thuật toán ID3 (Iterative Dichotomiser 3)

- Tính $Gain(A) = Info(D) - Info_A(D)$
  - Thuộc tính age

$$Gain(age) = Info(D) - Info_{age}(D) = 0.940 - 0.694 = 0.246 \text{ bits.}$$

  - Gain(income) = 0.029 bits.
  - Gain(credit rating) = 0.048 bits.
  - Gain(student) = 0.151 bits.

Thuật toán ID3 (Iterative Dichotomiser 3)

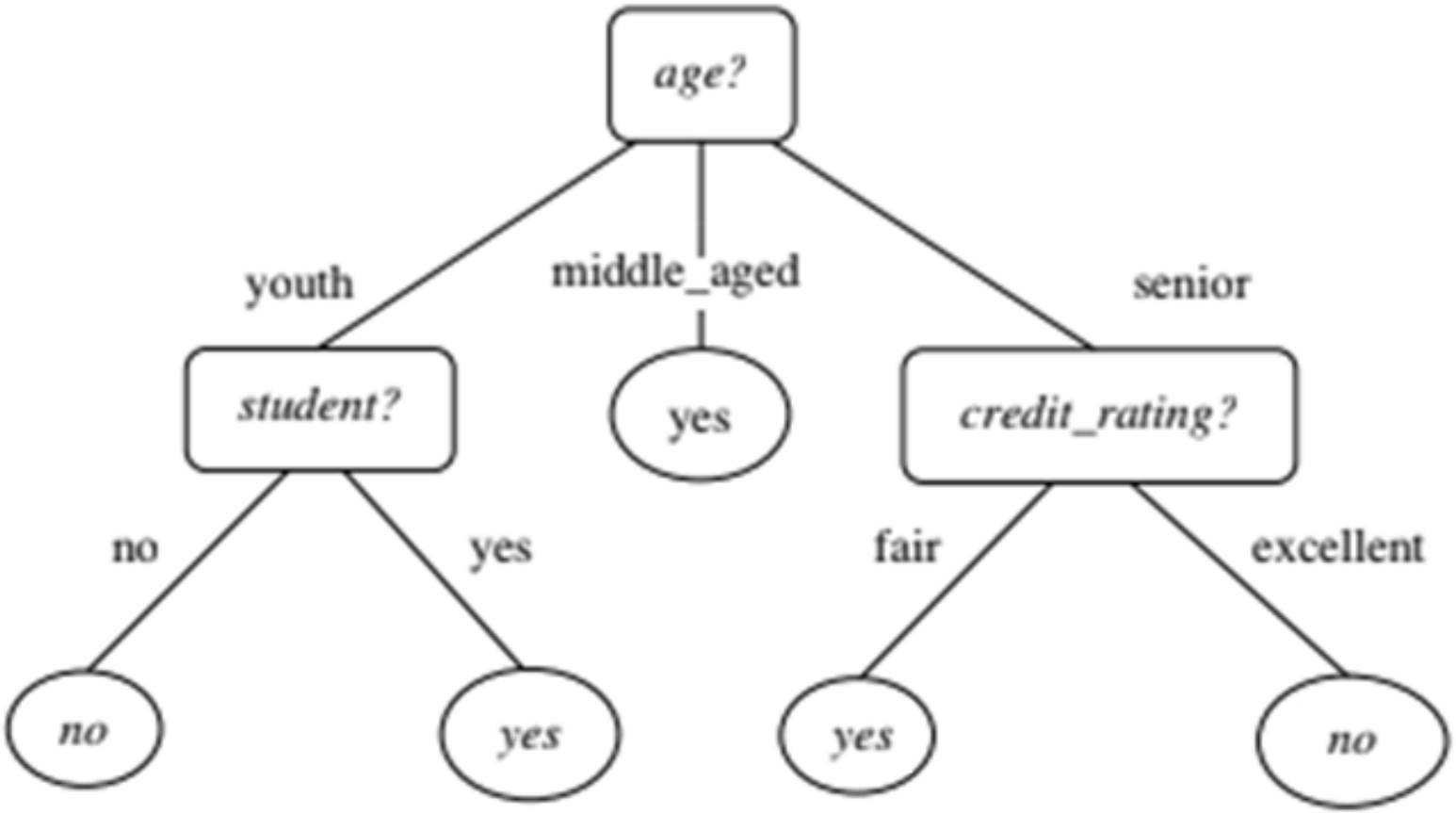


income	*student*	*credit_rating*	*class*
high	no	fair	no
high	no	excellent	no
medium	no	fair	no
low	yes	fair	yes
medium	yes	excellent	yes

income	*student*	*credit_rating*	*class*
medium	no	fair	yes
low	yes	fair	yes
low	yes	excellent	no
medium	yes	fair	yes
medium	no	excellent	no

income	*student*	*credit_rating*	*class*
high	no	fair	yes
low	yes	excellent	yes
medium	no	excellent	yes
high	yes	fair	yes

Thuật toán ID3 (Iterative Dichotomiser 3)



Phân loại dữ liệu với cây quyết định (C4.5)

- Độ đo Gain Ratio: GainRatio(A)
  - Giải quyết vấn đề một thuộc tính được dùng tạo ra rất nhiều phân hoạch (thậm chí mỗi phân hoạch chỉ gồm 1 phần tử).
  - Chuẩn hoá information gain với giá trị thông tin phân tách (split information): $\text{SplitInfo}_A(D)$
  - Splitting attribute A tương ứng với trị GainRatio(A) là trị lớn nhất.

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} * \log_2\left(\frac{|D_j|}{|D|}\right)$$

$$GainRatio(A) = \frac{Gain(A)}{SplitInfo_A(D)}$$

Phân loại dữ liệu với cây quyết định

$$\text{SplitInfo}_{\text{income}}(D) = -\frac{4}{14} \times \log_2\left(\frac{4}{14}\right) - \frac{6}{14} \times \log_2\left(\frac{6}{14}\right) - \frac{4}{14} \times \log_2\left(\frac{4}{14}\right)$$

$$= 0.926.$$

Gain(income) = 0.029

GainRatio(income) = 0.029/0.926 = 0.031

GainRatio(age)?

GainRatio(student)?

GainRatio(credit_rating)?

→ Splitting attribute?

Cây quyết định (Decision tree)

- Ưu điểm
  - Mô hình sinh ra các quy tắc dễ hiểu cho người đọc, tạo ra bộ luật với mỗi nhánh lá là một luật của cây.
  - Dữ liệu đầu vào có thể là là dữ liệu missing, không cần chuẩn hóa hoặc tạo biến giả
  - Có thể làm việc với cả dữ liệu số và dữ liệu Phân lớp
  - Có thể xác thực mô hình bằng cách sử dụng các kiểm tra thống kê
  - Có khả năng là việc với dữ liệu lớn
- Nhược điểm
  - Mô hình cây quyết định phụ thuộc rất lớn vào dữ liệu của bạn. Thậm chí, với một sự thay đổi nhỏ trong bộ dữ liệu, cấu trúc mô hình cây quyết định có thể thay đổi hoàn toàn.
  - Cây quyết định hay gặp vấn đề overfitting

Nội dung

- Tổng quan về bài toán phân lớp dữ liệu
- Phương pháp đánh giá độ chính xác
- Cây quyết định
- **Bài tập**

- Bài 1: Sử dụng thuật toán ID3 xây dựng cây quyết định cho tập dữ liệu chơi tennis như sau

Outlook	Temperature	Humidity	Wind	Play
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Mild	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes
Overcast	Mild	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Mild	High	Strong	No

- Áp dụng cây quyết định với tập mẫu mới:

Outlook	Temperature	Humidity	Windy	Play
Sunny	Hot	High	Strong	?

Một số bài tập về cây quyết định

- Bài 1: Cho tập dữ liệu huấn luyện sau, hãy dùng giải thuật ID3 để xây dựng cây quyết định với thuộc tính phân lớp là Species

	Species	Green	Legs	Height	Smelly
1)	M	N	3	S	Y
2)	M	Y	2	T	N
3)	M	Y	3	T	N
4)	M	N	2	S	Y
5)	M	Y	3	T	N
6)	H	N	2	T	Y
7)	H	N	2	S	N
8)	H	N	2	T	N
9)	H	Y	2	S	N
10)	H	N	2	T	Y