**Khoa Khoa Học & Kỹ Thuật Máy Tính**
**Trường Đại Học Bách Khoa Tp. Hồ Chí Minh**

# Chương 4: Phân loại dữ liệu

Khai phá dữ liệu

(Data mining)



Học kỳ 1 – 2021-2022

# Nội dung

- 4.1. Tổng quan về phân loại dữ liệu
- 4.2. Phân loại dữ liệu với cây quyết định
- 4.3. Phân loại dữ liệu với mạng Bayesian
- 4.4. Phân loại dữ liệu với mạng Neural
- 4.5. Các phương pháp phân loại dữ liệu khác
- 4.6. Tóm tắt

# Tài liệu tham khảo

- **[1] Jiawei Han, Micheline Kamber, "Data Mining: Concepts and Techniques", Second Edition, Morgan Kaufmann Publishers, 2006.**

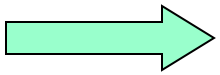
- Classification by Decision Tree Induction (291 -> 306)

- Bayesian Classification (310 -> 315)

- Classification by Backpropagation ( 327 -> 334)

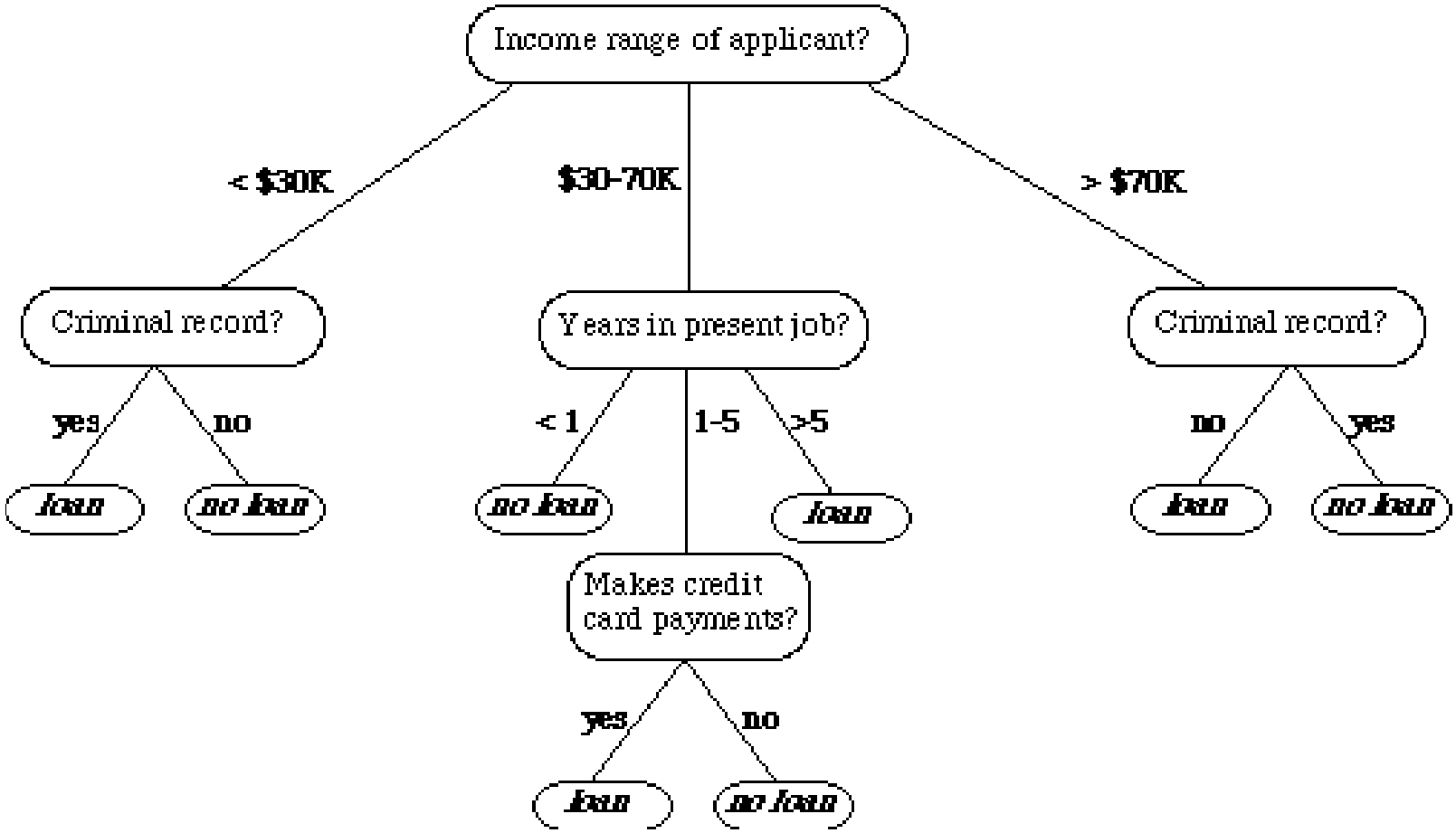
# 4.0. Tình huống 1

| *Tid* | Refund | Marital Status | Taxable Income | Evade |
|---|---|---|---|---|
| 1 | Yes | Single | 125K | No |
| 2 | No | Married | 100K | No |
| 3 | No | Single | 70K | No |
| 4 | Yes | Married | 120K | No |
| 5 | No | Divorced | 95K | Yes |
| 6 | No | Married | 60K | No |
| 7 | Yes | Divorced | 220K | No |
| 8 | No | Single | 85K | Yes |
| 9 | No | Married | 75K | No |
| 10 | No | Single | 90K | Yes |



Ông A (Tid = 100) có khả năng trốn thuế???

# 4.0. Tình huống 2



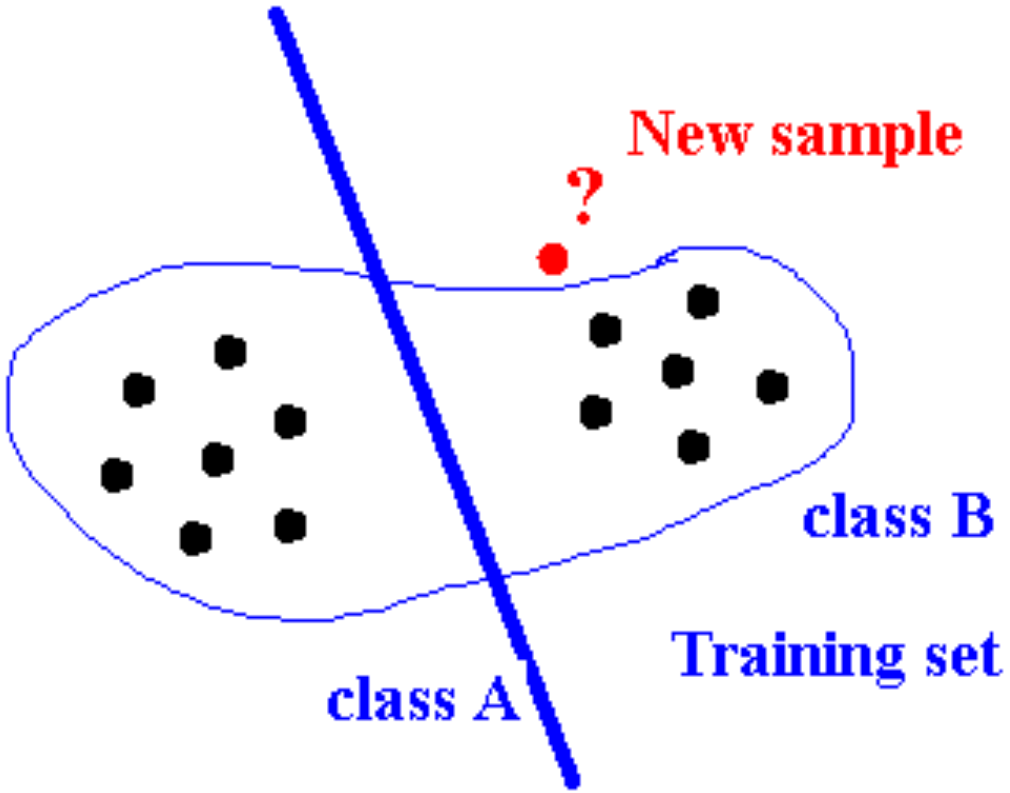

Với thông tin của một applicant A, xác định liệu ngân hàng có cho A vay không?

# 4.0. Tình huống 3

| Khóa | MãSV | MônHọc1 | MônHọc2 | ... | TốtNghiệp |
|---|---|---|---|---|---|
| 2004 | 1 | 9.0 | 8.5 | … | Có |
| 2004 | 2 | 6.5 | 8.0 | … | Có |
| 2004 | 3 | 4.0 | 2.5 | … | Không |
| 2004 | 8 | 5.5 | 3.5 | … | Không |
| 2004 | 14 | 5.0 | 5.5 | … | Có |
| … | … | … | … | … | … |
| 2005 | 90 | 7.0 | 6.0 | … | Có |
| 2006 | 24 | 9.5 | 7.5 | … | Có |
| 2007 | 82 | 5.5 | 4.5 | … | Không |
| 2008 | 47 | 2.0 | 3.0 | … | Không |
| … | … | … | … | … | … |

Làm sao xác định liệu sinh viên A sẽ tốt nghiệp?

# 4.0. Tình huống …




Cho trước tập huấn luyện (training set), dẫn ra mô tả về class A và class B?

Cho trước mẫu/đối tượng mới, làm sao xác định class cho mẫu/đối tượng đó?

Liệu class đó có thực sự phù hợp/đúng cho mẫu/đối tượng đó?

# 4.1. Tổng quan về phân loại dữ liệu

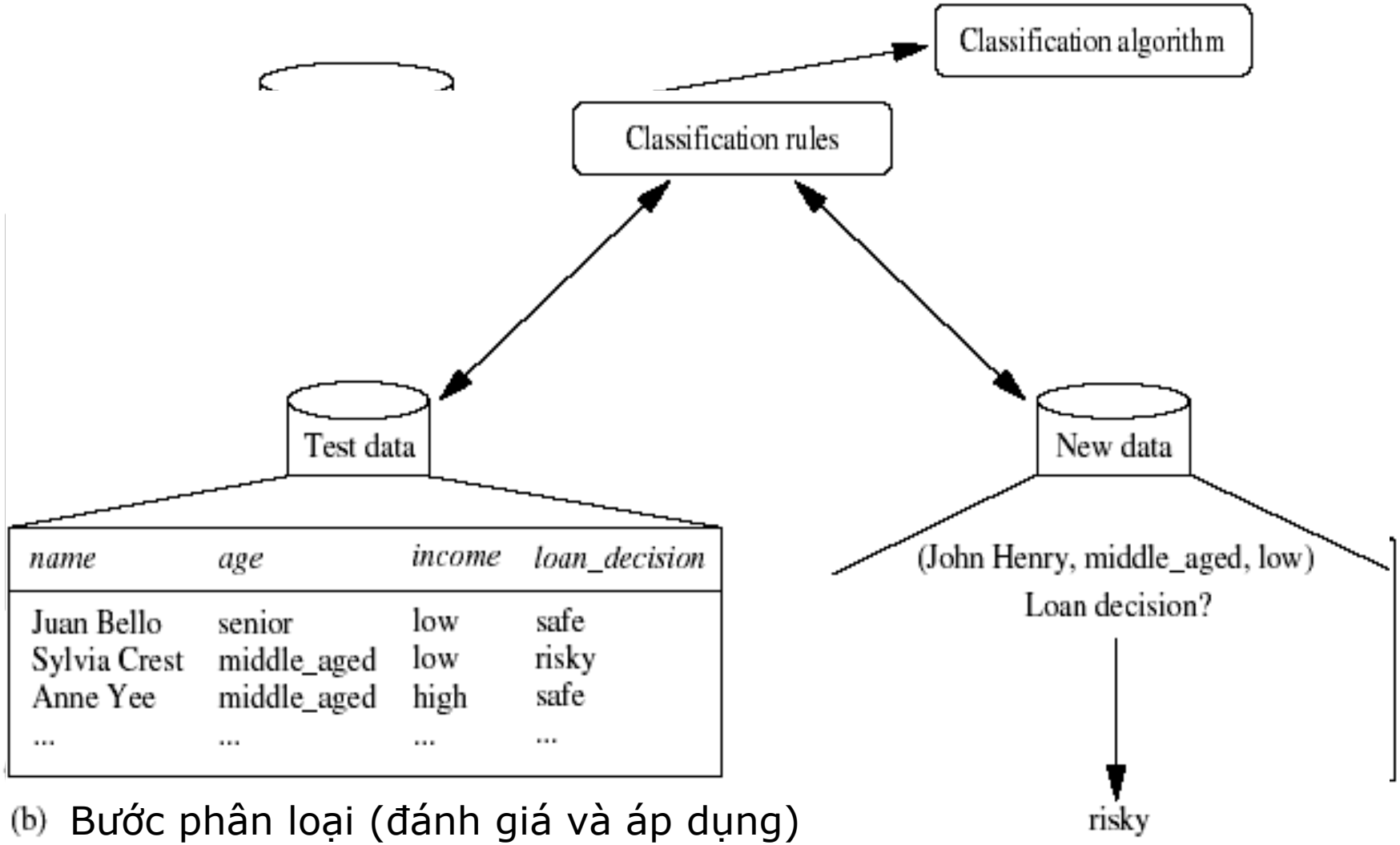
- Phân loại dữ liệu (classification)
  - Dạng phân tích dữ liệu nhằm rút trích các mô hình mô tả các lớp dữ liệu hoặc dự đoán xu hướng dữ liệu
  - Quá trình gồm hai bước:
    - Bước học (giai đoạn huấn luyện): xây dựng bộ phân loại (classifier) bằng việc phân tích/học tập huấn luyện
    - Bước phân loại (classification): phân loại dữ liệu/đối tượng mới nếu độ chính xác của bộ phân loại được đánh giá là có thể chấp nhận được (acceptable)

**y = f (X)** với y là nhãn (phần mô tả) của một lớp (class) và X là dữ liệu/đối tượng

- Bước học: X trong tập huấn luyện, một trị y được cho trước với X → xác định f

- Bước phân loại: đánh giá f với (X', y') và X' <> mọi X trong tập huấn luyện; nếu acceptable thì dùng f để xác định y'' cho X'' (mới)

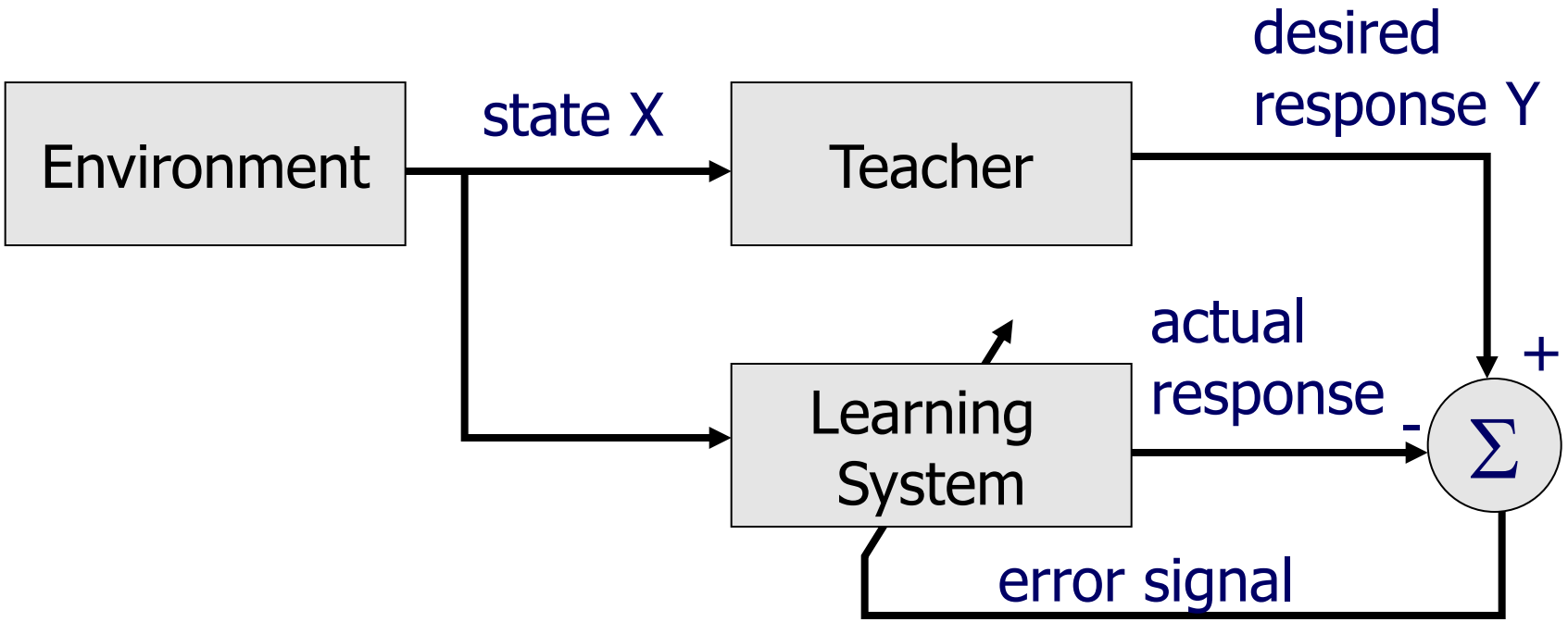
# 4.1. Tổng quan về phân loại dữ liệu




(b) Bước phân loại (đánh giá và áp dụng)

# 4.1. Tổng quan về phân loại dữ liệu

- Phân loại dữ liệu
  - Dạng học có giám sát (supervised learning)



# 4.1. Tổng quan về phân loại dữ liệu

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

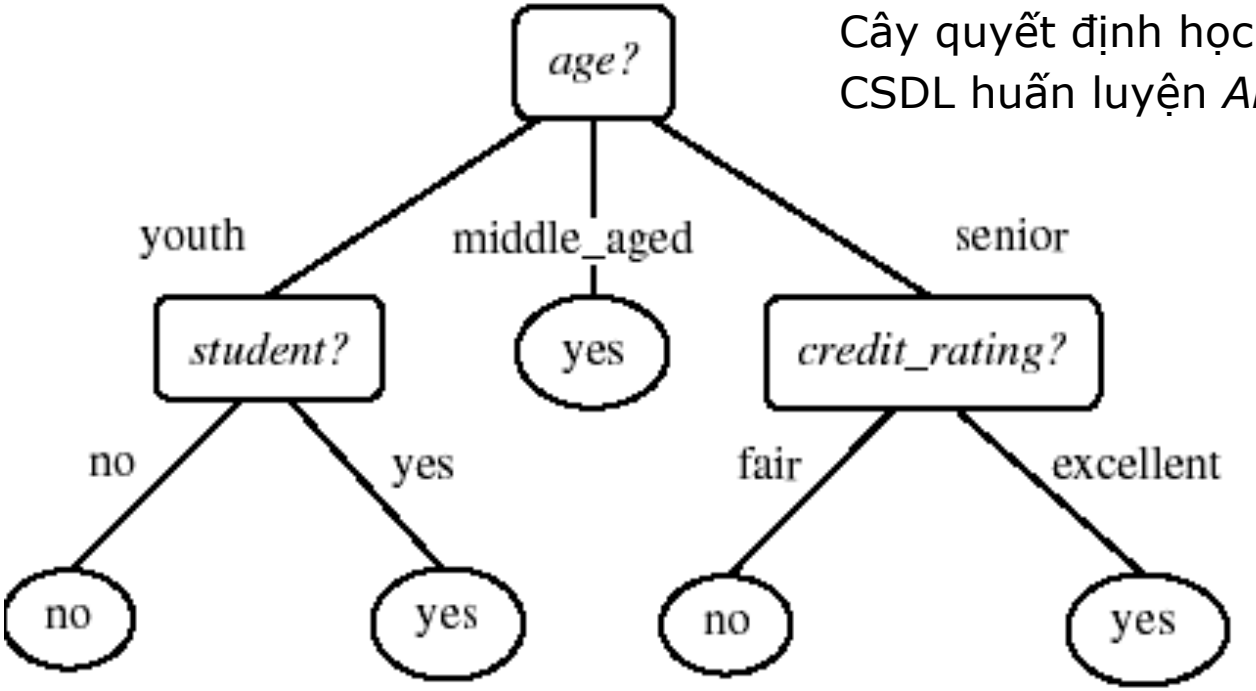
# 4.2. Phân loại dữ liệu với cây quyết định

| 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 |

Cơ sở dữ liệu khách hàng *AllElectronics* dùng cho bước học

# 4.2. Phân loại dữ liệu với cây quyết định

- Cây quyết định (decision tree) – mô hình phân loại
  - Node nội: phép kiểm thử (test) trên một thuộc tính
  - Node lá: nhãn/mô tả của một lớp (class label)
  - Nhánh từ một node nội: kết quả của một phép thử trên thuộc tính tương ứng




Cây quyết định học được từ CSDL huấn luyện *AllElectronics*

# 4.2. Phân loại dữ liệu với cây quyết định

- Giải thuật xây dựng cây quyết định
  - ID3, C4.5, CART (Classification and Regression Trees – binary decision trees)

**Algorithm: Generate_decision_tree.** Generate a decision tree from the training tuples of data partition *D*.

**Input:**

- Data partition, *D*, which is a set of training tuples and their associated class labels;
- *attribute_list*, the set of candidate attributes;
- *Attribute_selection_method*, a procedure to determine the splitting criterion that "best" partitions the data tuples into individual classes. This criterion consists of a *splitting_attribute* and, possibly, either a *split point* or *splitting subset*.

**Output:** A decision tree.

# 4.2. Phân loại dữ liệu với cây quyết định

**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 $D_j$ be the set of data tuples in $D$ satisfying outcome $j$; // a partition
(12) if $D_j$ 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**($D_j$, *attribute_list*) to node $N$;
endfor
(15) return $N$;

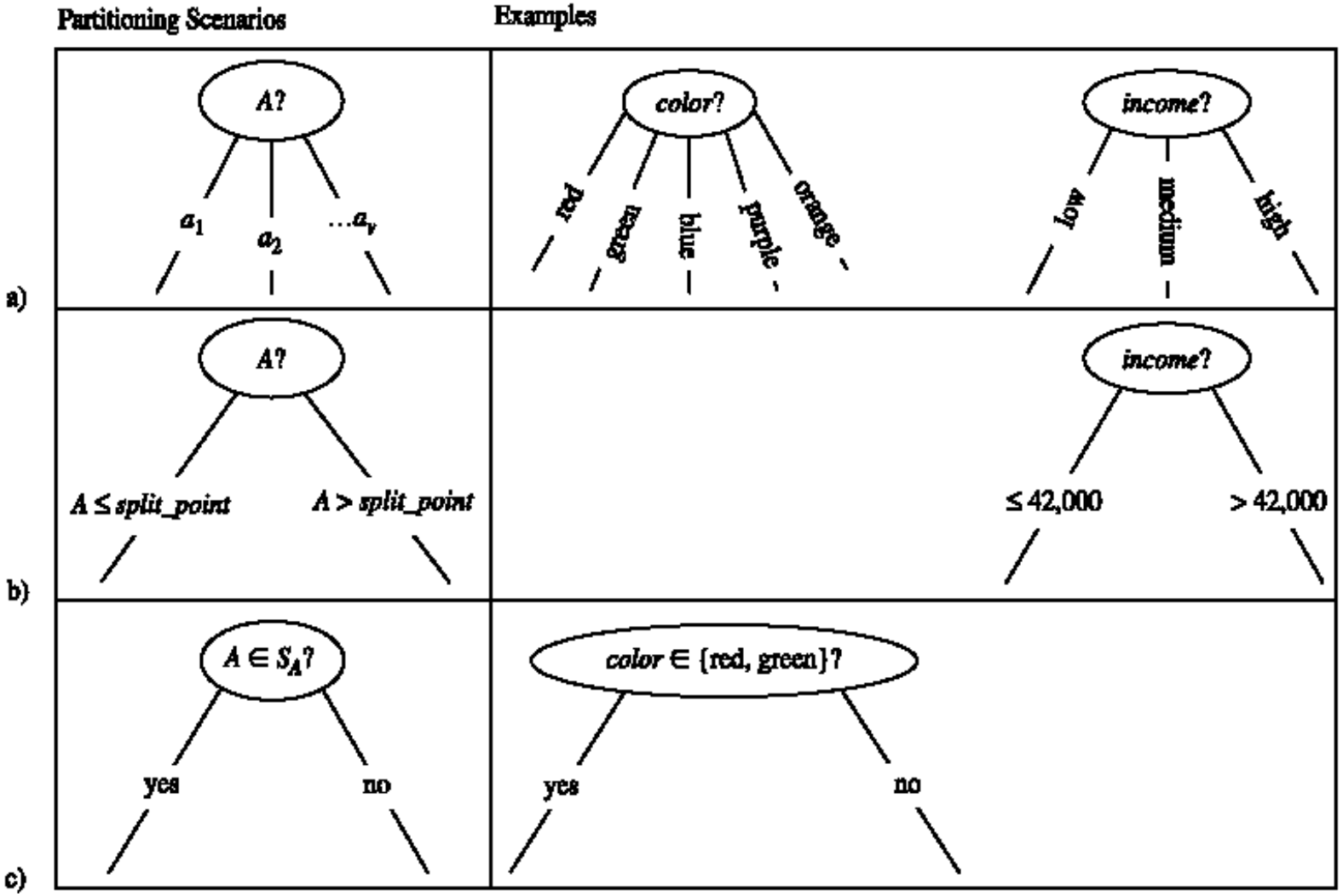
# 4.2. Phân loại dữ liệu với cây quyết định

- Đặc điểm của giải thuật
  - Giải thuật tham lam (không có quay lui), chia để trị, đệ qui, từ trên xuống
  - Độ phức tạp với tập huấn luyện ***D*** gồm |***D***| phần tử (đối tượng), mỗi phần tử gồm ***n*** thuộc tính
    - O($n$*|$D$|*$log$|$D$|)
      - Mỗi thuộc tính ứng với mỗi mức (level) của cây.
      - Cho mỗi mức của cây, |D| phân tử huấn luyện được duyệt qua.
  - In-memory → ???

# 4.2. Phân loại dữ liệu với cây quyết định

- Attribute_selection_method
  - Phương thức dùng heuristic để chọn tiêu chí rẽ nhánh tại một node, i.e. phân hoạch tập huấn luyện *D* thành các phân hoạch con với các nhãn phù hợp
    - Xếp hạng mỗi thuộc tính
    - Thuộc tính được chọn để rẽ nhánh là thuộc có trị số điểm (score) lớn nhất
    - Độ đo chọn thuộc tính phân tách (splitting attribute): information gain, gain ratio, gini index

# 4.2. Phân loại dữ liệu với cây quyết định




A là thuộc tính phân tách (splitting attribute).

# 4.2. Phân loại dữ liệu với cây quyết định

- Độ đo Information Gain
  - Dựa trên lý thuyết thông tin (information theory) của Claude Shannon về giá trị (nội dung thông tin) của tin
  - Thuộc tính tương ứng với information gain lớn nhất sẽ được chọn làm splitting attribute cho node N.
    - Node N là node hiện tại cần phân hoạch các phần tử trong D.
    - Splitting attribute đảm bảo sự trùng lắp (impurity)/ngẫu nhiên (randomness) ít nhất giữa các phân hoạch tạo được.
    - Cách tiếp cận này giúp tối thiểu số phép thử (test) để phân loại một phần tử.

# 4.2. Phân loại dữ liệu với cây quyết định

- Độ đo Information Gain
  - Lượng thông tin cần để phân loại một phần tử trong D (= Entropy của D): Info(D)
    - $p_i$: 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

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

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

# 4.2. Phân loại dữ liệu với cây quyết định

- Độ đo Information Gain
  - 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.
    - Lượng thông tin này sẽ cho biết mức độ trùng lắp giữa các phân hoạch, nghĩa là một phân hoạch chứa các phần tử từ một lớp hay nhiều lớp khác nhau.
    - Mong đợi: $Info_A(D)$ càng nhỏ càng tốt.

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

# 4.2. Phân loại dữ liệu với cây quyết định

- Độ đo Information Gain
  - Information gain chính là độ sai biệt giữa trị thông tin Info(D) ban đầu (trước phân hoạch) và trị thông tin mới $Info_A(D)$ (sau phân hoạch với A).

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

# Ví dụ:

| Outlook | Temperature | Humidity | Windy | Decision |
|---|---|---|---|---|
| sunny | hot | high | false | n |
| sunny | hot | high | true | n |
| overcast | hot | high | false | p |
| rain | mild | high | false | p |
| rain | cool | normal | false | p |
| rain | cool | normal | false | n |
| overcast | cool | normal | true | p |
| sunny | mild | high | false | p |
| sunny | mild | normal | true | p |
| rain | mild | normal | false | p |
| sunny | mild | normal | true | p |
| overcast | mild | high | true | p |
| overcast | hot | normal | false | p |
| rain | mild | high | true | n |

- Outlook: sunny, overcast , rain
- Temperature: hot , mild, cood
- Humidity: high,normal
- Windy: true,false
- **Decision: n(negative), p(positive)**

# Ví dụ:

- Tạo nút gốc( rootNode) , chứa đựng toàn bộ learning set như là những tập hợp con của chúng (subset) sau đó tính:

Entropy(rootNode.subset)= $-(9/14)\log_2(9/14) - (5/14)\log_2(5/14) = 0.940$

- Tính toán thông tin nhận được cho mỗi thuộc tính:

Gain(S,Windy)= Entropy(S)$-(8/14)$Entropy($S_{false}$) – $(6/14)$Entropy($S_{true}$) = 0.048

Gain(S,Humidity) = 0.151

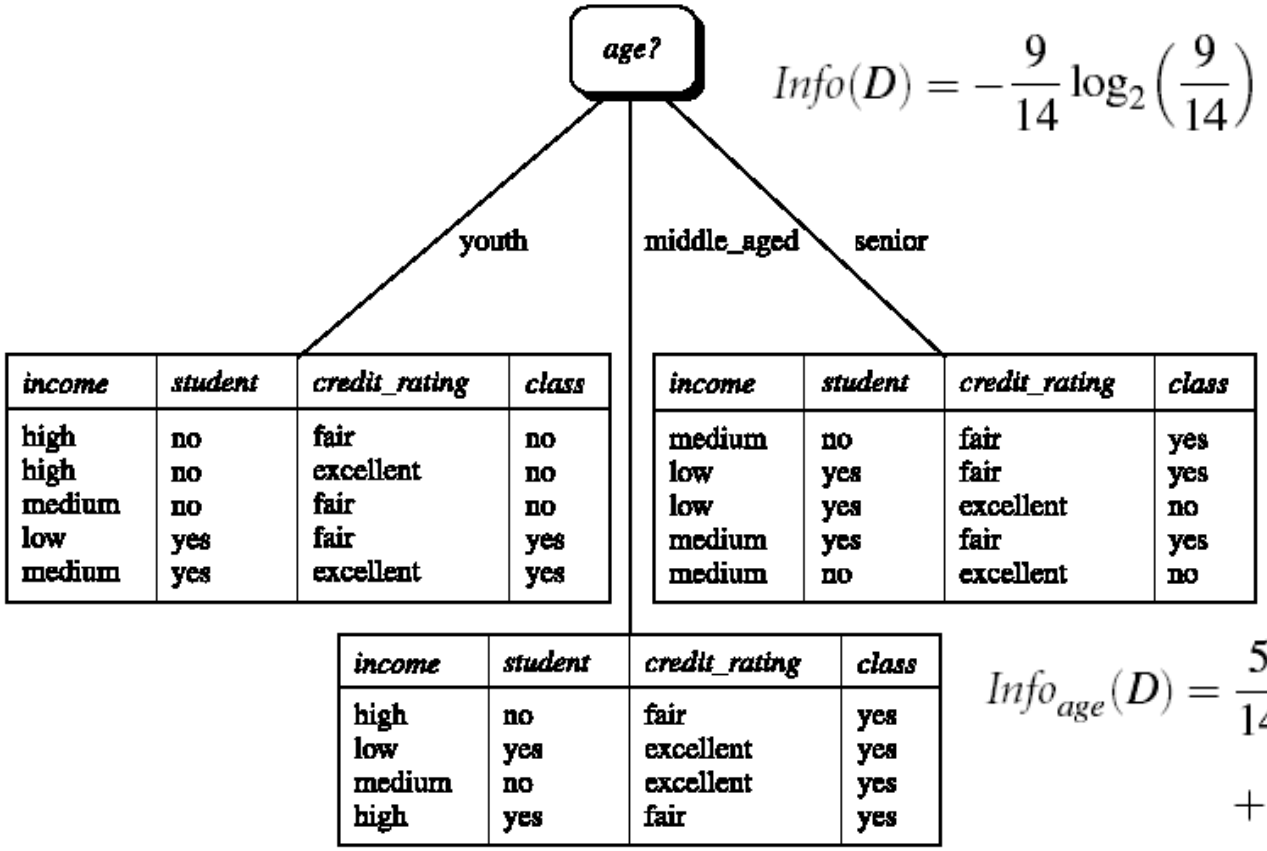
Gain(S,Temperature) = 0.029

Gain(S,Outlook) = **0.246**

- Chọn lựa những thuộc tính với thông tin nhận được tối đa , đó chính là sự phân chia theo thuộc tính “outlook”.

- Áp dụng ID3 cho mỗi nút con của nút gốc này , cho đến khi đạt đến nút lá hoặc nút có entropy = 0.

# 4.2. Phân loại dữ liệu với cây quyết định




| 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 |

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

Gain(age)=0.246 bits

Gain(income)?

Gain(student)?

Gain(credit_rating)?

→ Splitting attribute?

$$Info_{age}(D) = \frac{5}{14} \times \left(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}\right) + \frac{4}{14} \times \left(-\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4}\right) + \frac{5}{14} \times \left(-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}\right) = 0.694 \text{ bits.}$$

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

# 4.2. Phân loại dữ liệu với cây quyết định

- Độ đo Gain Ratio: GainRatio(A)
  - Dùng với C4.5
  - 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 trị thông tin phân tách (split information): $SplitInfo_A(D)$
  - Splitting attribute A tương ứng với trị GainRatio(A) là trị lớn nhất.

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

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

# 4.2. 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?

# 4.2. Phân loại dữ liệu với cây quyết định

- Độ đo Gini Index
  - Dùng với CART
  - Sự phân tách nhị phân (binary split) cho mỗi thuộc tính A
    - $A \in S_A$?
    - $S_A$ là một tập con gồm một hay v-1 trị thuộc tính A.
  - Gini index của một thuộc tính là trị nhỏ nhất tương ứng với một tập con $S_A$ từ $2^v$ – 2 tập con.
  - Splitting attribute tương ứng với gini index nhỏ nhất để tối đa hóa sự suy giảm về độ trùng lắp giữa các phân hoạch.

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2$$

$$Gini_A(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2)$$

$$\Delta Gini(A) = Gini(D) - Gini_A(D)$$

# 4.2. Phân loại dữ liệu với cây quyết định

$$Gini(D) = 1 - \left(\frac{9}{14}\right)^2 - \left(\frac{5}{14}\right)^2 = 0.459$$

$$Gini_{income \in \{low,medium\}}(D)$$

$$= \frac{10}{14} Gini(D_1) + \frac{4}{14} Gini(D_2)$$

$$= \frac{10}{14}\left(1 - \left(\frac{6}{10}\right)^2 - \left(\frac{4}{10}\right)^2\right) + \frac{4}{14}\left(1 - \left(\frac{1}{4}\right)^2 - \left(\frac{3}{4}\right)^2\right)$$

$$= 0.450$$

$$= Gini_{income \in \{high\}}(D).$$

$Gini_{income \in \{low,high\}} = Gini_{income \in \{medium\}} = 0.315$

$Gini_{income \in \{medium,high\}} = Gini_{income \in \{low\}} = 0.300$

→ $Gini_{income \in \{medium,high\}/\{low\}} = 0.300$

$Gini_{age \in \{youth,senior\}/\{middle\_aged\}} = 0.375$

$Gini_{student} = 0.367$

$Gini_{credit\_rating} = 0.429$

→ Splitting attribute?

# 4.2. Phân loại dữ liệu với cây quyết định

- Xây dựng cây quyết định từ cơ sở dữ liệu huấn luyện AllElectronics
  - Dùng độ đo Information Gain
  - Dùng độ đo Gain Ratio
  - Dùng độ đo Gini Index
  - → Các cây quyết định học được giống nhau???
  - → Tiến hành đánh giá và phân loại với các cây quyết định học được

# 4.3. Phân loại dữ liệu với mạng Bayesian

- Dựa trên định lý của Bayes
  - Phân loại Naïve Bayesian
    - Giả định: độc lập có điều kiện lớp (class conditional independence)
  - Phân loại Bayesian belief networks
- Phương pháp phân loại dựa trên xác suất

# 4.3. Phân loại dữ liệu với mạng Bayesian

Reverend Thomas Bayes (1702-1761)

# 4.3. Phân loại dữ liệu với mạng Bayesian

- Định lý Bayes
  - X: một tuple/đối tượng (evidence)
  - H: giả thuyết (hypothesis)
    - X thuộc về lớp C.

Cho một RID, RID thuộc về lớp "yes" (buys_computer = yes)

X →

X được xác định bởi trị của các thuộc tính.

| 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 |

# 4.3. Phân loại dữ liệu với mạng Bayesian

- Định lý Bayes
  - P(H|X): posterior probability H khi X xảy ra
    - Xác suất có điều kiện của H đối với X.
    - Ví dụ: P(buys_computer=yes|age=young, income=high) là xác suất mua máy tính của khách hàng có tuổi "young" và thu nhập "high".
  - P(X|H): posterior probability
    - Xác suất có điều kiện của X đối với H. X khi H xảy ra
    - Ví dụ: P(age=young, income=high|buys_computer=yes) là xác suất khách hàng mua máy tính có tuổi "young" và thu nhập "high".
      - P(age=young, income=high|buys_computer=yes) = 0
      - P(age=young, income=high|buys_computer=no) = 2/5 = 0.4

# 4.3. Phân loại dữ liệu với mạng Bayesian

- Định lý Bayes
  - P(H): prior probability xác suất độc lập
    - Xác suất của H
    - Ví dụ: P(buys_computer=yes) là xác suất mua máy tính của khách hàng nói chung.
      - P(buys_computer=yes) = 9/14 = 0.643
      - P(buys_computer=no) = 5/14 = 0.357
  - P(X): prior probability
    - Xác suất của X
    - Ví dụ: P(age=young, income=high) là xác suất khách hàng có tuổi "young" và thu nhập "high".
      - P(age=young, income=high) = 2/14 = 0.143

# 4.3. Phân loại dữ liệu với mạng Bayesian

- Định lý Bayes
  - P(H), P(X|H), P(X) có thể được tính từ tập dữ liệu cho trước.
  - P(H|X) được tính từ định lý Bayes.

$$P(H \mid X) = \frac{P(X \mid H)P(H)}{P(X)}$$

P(buys_computer=yes|age=young, income=high) = P(age=young, income=high|buys_computer=yes)P(buys_computer=yes)/P(age=young, income=high) = 0

P(buys_computer=no|age=young, income=high) = P(age=young, income=high|buys_computer=no)P(buys_computer=no)/P(age=young, income=high) = 0.4*0.357/0.143 = 0.9986

# 4.3. Phân loại dữ liệu với mạng Bayesian

- Cho trước tập dữ liệu huấn luyện D với mô tả (nhãn) của các lớp $C_i$, i=1..m, quá trình phân loại một tuple/đối tượng $X = (x_1, x_2, \ldots, x_n)$ với mạng Bayesian như sau:
  - X được phân loại vào $C_i$ nếu và chỉ nếu

$$P(C_i|X) > P(C_j|X) \text{ với } 1<=j<=m, j<>i$$

$$P(C_i \mid X) = \frac{P(X \mid C_i)P(C_i)}{P(X)}$$

  - → Tối đa hóa $P(C_i|X)$ (i.e. chọn $C_i$ nếu $P(C_i|X)$ là trị lớn nhất)
  - → Tối đa hóa $P(X|C_i)P(C_i)$
    - → $P(C_1) = P(C_2) = .. = P(C_m)$ hoặc $P(C_i) = |C_{i,D}|/|D|$ …

# 4.3. Phân loại dữ liệu với mạng Bayesian

độc lập

$$P(X \mid C_i) = \prod_{k=1}^{n} P(x_k \mid C_i) = P(x_1 \mid C_i) * P(x_2 \mid C_i) * .. * P(x_n \mid C_i)$$

- $P(X|C_i)$ được tính với giả định class conditional independence.
- $x_k$, k = 1..n: trị thuộc tính $A_k$ của X
- $P(x_k|C_i)$ được tính như sau:
  - $A_k$ là thuộc tính rời rạc.
    - $P(x_k|C_i) = |\{X'|x'_k = x_k \wedge X' \in C_i\}|/|C_{i,D}|$
  - $A_k$ là thuộc tính liên tục.
    - $P(x_k|C_i)$ tuân theo một phân bố xác suất nào đó (ví dụ: phân bố Gauss).

# 4.3. Phân loại dữ liệu với mạng Bayesian

- Nếu $P(x_k|C_i) = 0$ thì $P(X|C_i) = 0$!!!
  - Ban đầu
    - $P(x_k|C_i) = |\{X'|x'_k = x_k \wedge X' \in C_i\}|/|C_{i,D}|$
  - Laplace (Pierre Laplace, nhà toán học Pháp, 1749-1827)
    - $P(x_k|C_i) = (|\{X'|x'_k = x_k \wedge X' \in C_i\}| \mathbf{+1})/(|C_{i,D}| \mathbf{+ m})$
  - z-estimate
    - $P(x_k|C_i) = (|\{X'|x'_k = x_k \wedge X' \in C_i\}| \mathbf{+ z*P(x_k)})/(|C_{i,D}| \mathbf{+ z})$

# 4.3. Phân loại dữ liệu với mạng Bayesian

$$\boldsymbol{X} = (age = youth,\ income = medium,\ student = yes,\ credit\_rating = fair)$$

$C_1$ = {X'|X'.buys_computer = yes}

$C_2$ = {X''|X''.buys_computer = no}

$$P(age = youth \mid buys\_computer = yes) = 2/9 = 0.222$$
$$P(age = youth \mid buys\_computer = no) = 3/5 = 0.600$$
$$P(income = medium \mid buys\_computer = yes) = 4/9 = 0.444$$
$$P(income = medium \mid buys\_computer = no) = 2/5 = 0.400$$
$$P(student = yes \mid buys\_computer = yes) = 6/9 = 0.667$$
$$P(student = yes \mid buys\_computer = no) = 1/5 = 0.200$$
$$P(credit\_rating = fair \mid buys\_computer = yes) = 6/9 = 0.667$$
$$P(credit\_rating = fair \mid buys\_computer = no) = 2/5 = 0.400$$

$$P(buys\_computer = yes) = 9/14 = 0.643$$
$$P(buys\_computer = no) = 5/14 = 0.357$$

$$\begin{aligned} P(\boldsymbol{X} \mid buys\_computer = yes) = {} & P(age = youth \mid buys\_computer = yes) \times \\ & P(income = medium \mid buys\_computer = yes) \times \\ & P(student = yes \mid buys\_computer = yes) \times \\ & P(credit\_rating = fair \mid buys\_computer = yes) \\ = {} & 0.222 \times 0.444 \times 0.667 \times 0.667 = 0.044. \end{aligned}$$

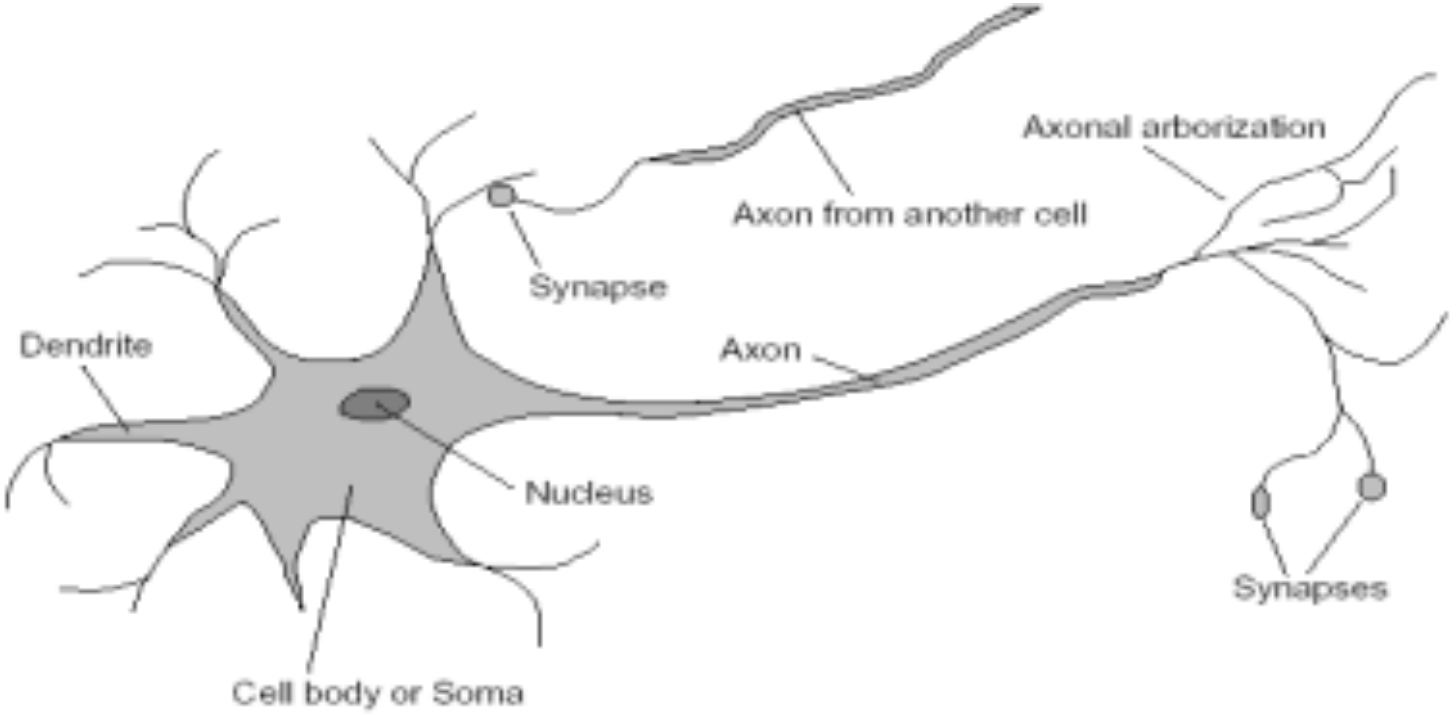
$$P(\boldsymbol{X} \mid buys\_computer = no) = 0.600 \times 0.400 \times 0.200 \times 0.400 = 0.019.$$

$$P(\boldsymbol{X} \mid buys\_computer = yes) P(buys\_computer = yes) = 0.044 \times 0.643 = 0.028$$
$$P(\boldsymbol{X} \mid buys\_computer = no) P(buys\_computer = no) = 0.019 \times 0.357 = 0.007$$

→ $X \in C_1$

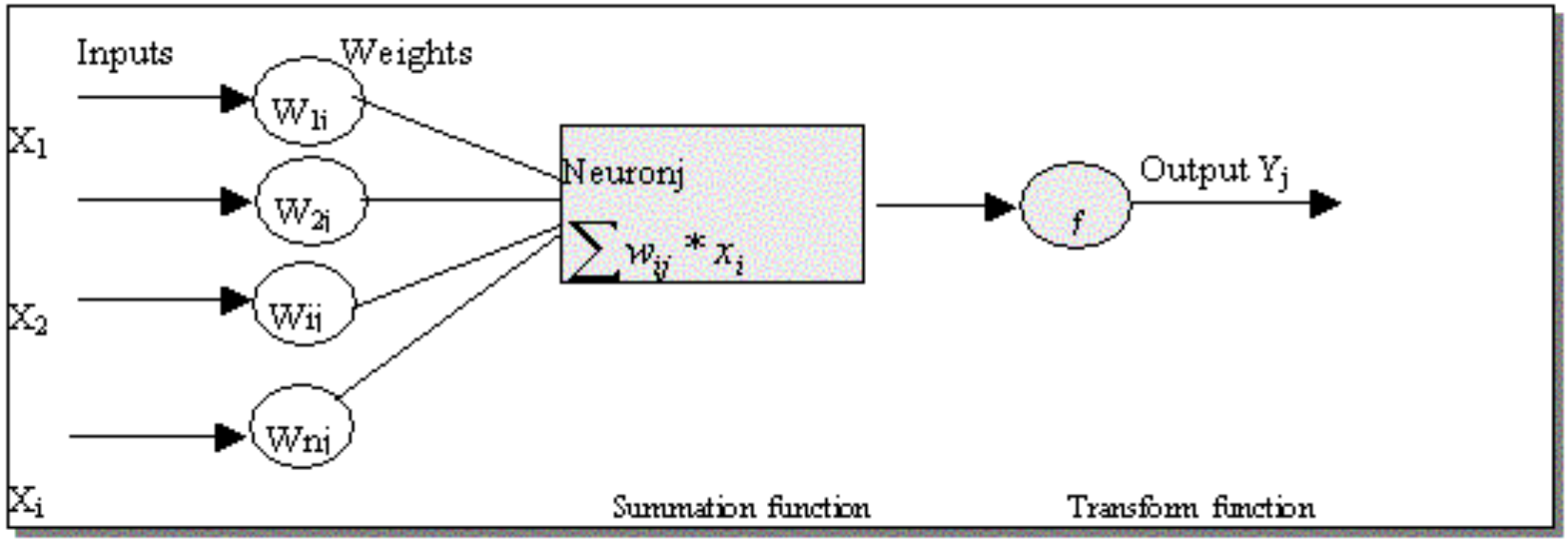
# 4.4. Phân loại dữ liệu với mạng Neural

- Mạng Neural sinh học



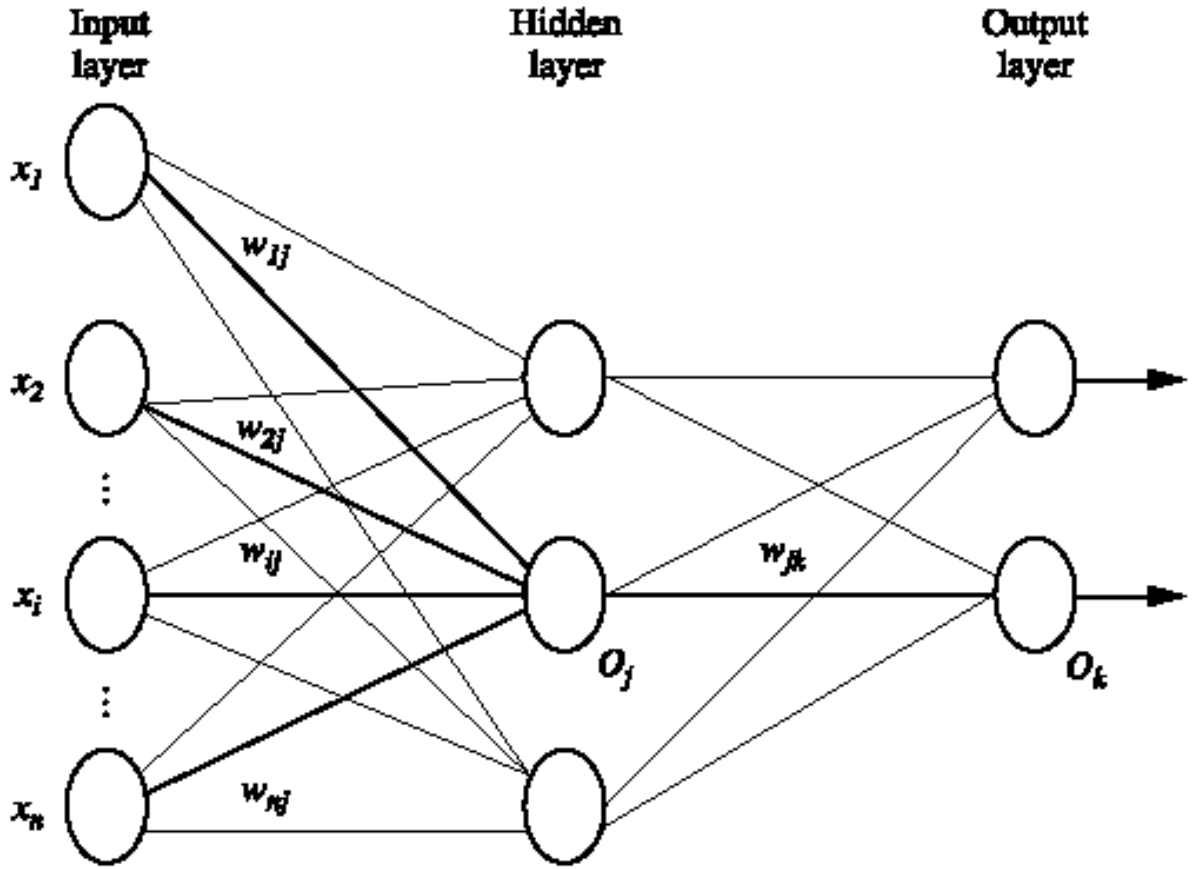
# 4.4. Phân loại dữ liệu với mạng Neural

- Quá trình xử lý thông tin tại một neuron của mạng Neural nhân tạo

# 4.4. Phân loại dữ liệu với mạng Neural

- Mạng neural feed-forward đa tầng



# 4.4. Phân loại dữ liệu với mạng Neural

- Giải thuật học lan truyền ngược (Backpropagation) có giám sát

**Algorithm: Backpropagation.** Neural network learning for classification or prediction, using the backpropagation algorithm.

Input:

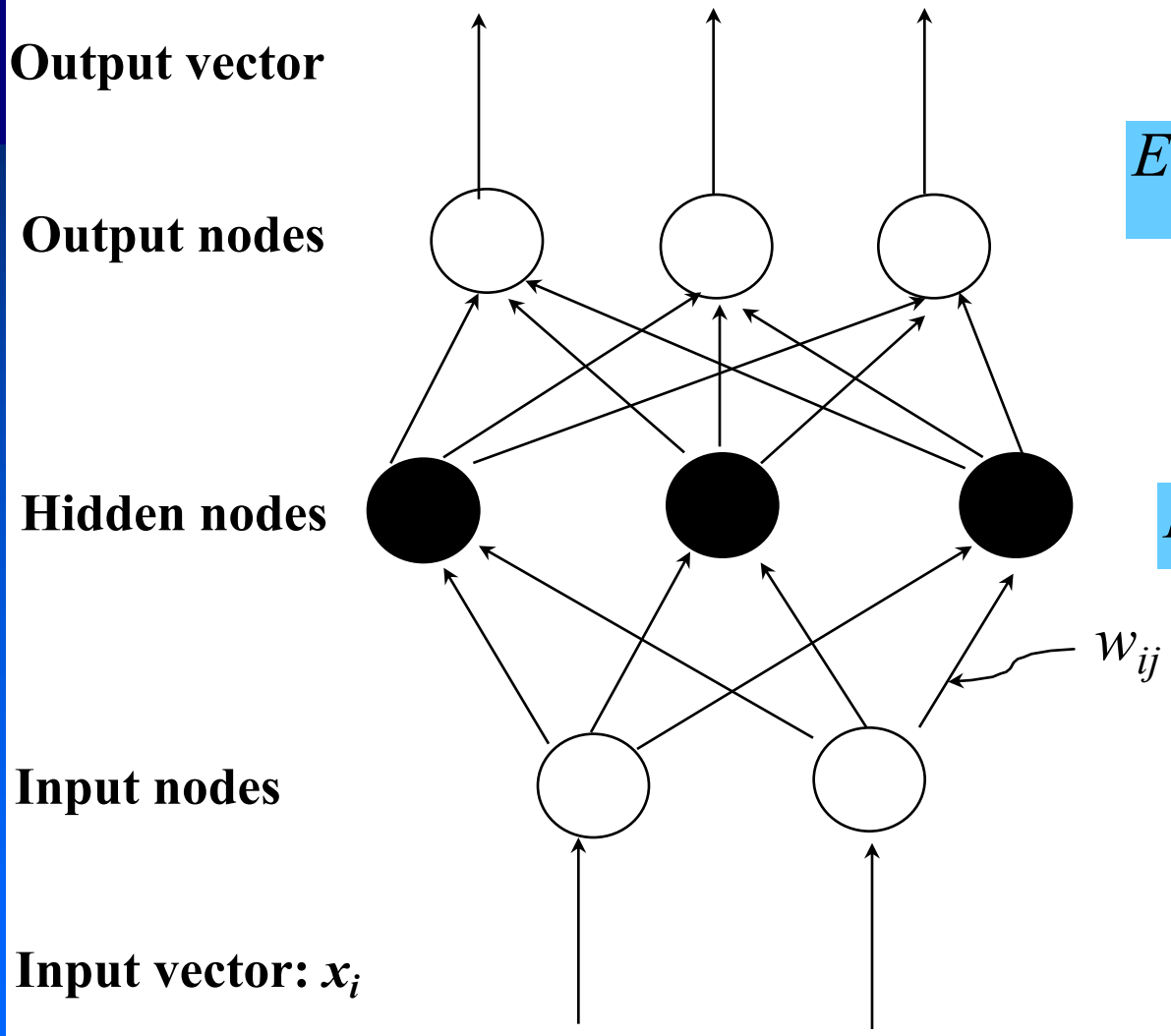
- *D*, a data set consisting of the training tuples and their associated target values;
- *l*, the learning rate;
- *network*, a multilayer feed-forward network.

**Output:** A trained neural network.

# 4.4. Phân loại dữ liệu với mạng Neural

(1) Initialize all weights and biases in *network*;
(2) **while** terminating condition is not satisfied {
(3) **for** each training tuple $\boldsymbol{X}$ in $D$ {
(4) // Propagate the inputs forward:
(5) **for** each input layer unit $j$ {
(6) $O_j = I_j$; // output of an input unit is its actual input value
(7) **for** each hidden or output layer unit $j$ {
(8) $I_j = \sum_i w_{ij} O_i + \theta_j$; //compute the net input of unit $j$ with respect to the previous layer, $i$
(9) $O_j = \frac{1}{1+e^{-I_j}}$; } // compute the output of each unit $j$
(10) // Backpropagate the errors:
(11) **for** each unit $j$ in the output layer
(12) $Err_j = O_j(1 - O_j)(T_j - O_j)$; // compute the error
(13) **for** each unit $j$ in the hidden layers, from the last to the first hidden layer
(14) $Err_j = O_j(1 - O_j)\sum_k Err_k w_{jk}$; // compute the error with respect to the next higher layer, $k$
(15) **for** each weight $w_{ij}$ in *network* {
(16) $\Delta w_{ij} = (l) Err_j O_i$; // weight increment
(17) $w_{ij} = w_{ij} + \Delta w_{ij}$; } // weight update
(18) **for** each bias $\theta_j$ in *network* {
(19) $\Delta \theta_j = (l) Err_j$; // bias increment
(20) $\theta_j = \theta_j + \Delta \theta_j$; } // bias update
(21) } }

# 4.4. Phân loại dữ liệu với mạng Neural




$$Err_j = O_j(1-O_j)\sum_k Err_k w_{jk}$$

$$\theta_j = \theta_j + (l)Err_j$$

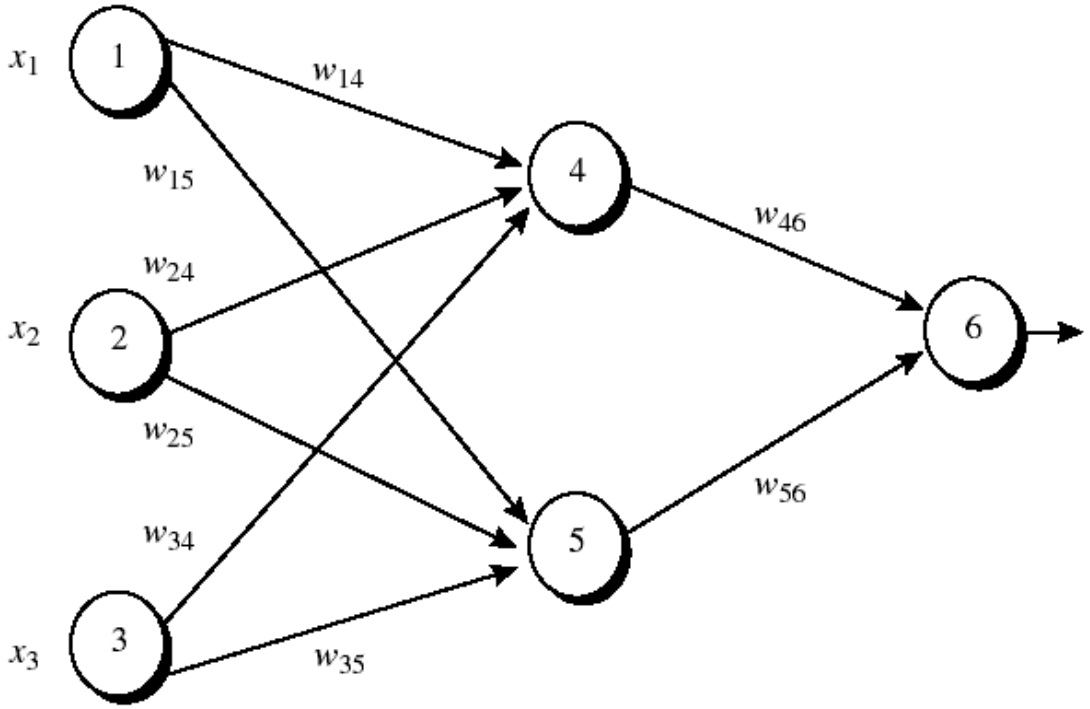
$$w_{ij} = w_{ij} + (l)Err_j O_i$$

$$Err_j = O_j(1-O_j)(T_j - O_j)$$

$$O_j = \frac{1}{1+e^{-I_j}}$$

$$I_j = \sum_i w_{ij} O_i + \theta_j$$

# 4.4. Phân loại dữ liệu với mạng Neural




Initial input, weight, and bias values.

| $x_1$ | $x_2$ | $x_3$ | $w_{14}$ | $w_{15}$ | $w_{24}$ | $w_{25}$ | $w_{34}$ | $w_{35}$ | $w_{46}$ | $w_{56}$ | $\theta_4$ | $\theta_5$ | $\theta_6$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0.2 | −0.3 | 0.4 | 0.1 | −0.5 | 0.2 | −0.3 | −0.2 | −0.4 | 0.2 | 0.1 |

# 4.4. Phân loại dữ liệu với mạng Neural

The net input and output calculations.

| *Unit j* | *Net input, $I_j$* | *Output, $O_j$* |
|---|---|---|
| 4 | $0.2+0-0.5-0.4=-0.7$ | $1/(1+e^{0.7})=0.332$ |
| 5 | $-0.3+0+0.2+0.2=0.1$ | $1/(1+e^{-0.1})=0.525$ |
| 6 | $(-0.3)(0.332)-(0.2)(0.525)+0.1=-0.105$ | $1/(1+e^{0.105})=0.474$ |

Calculation of the error at each node.

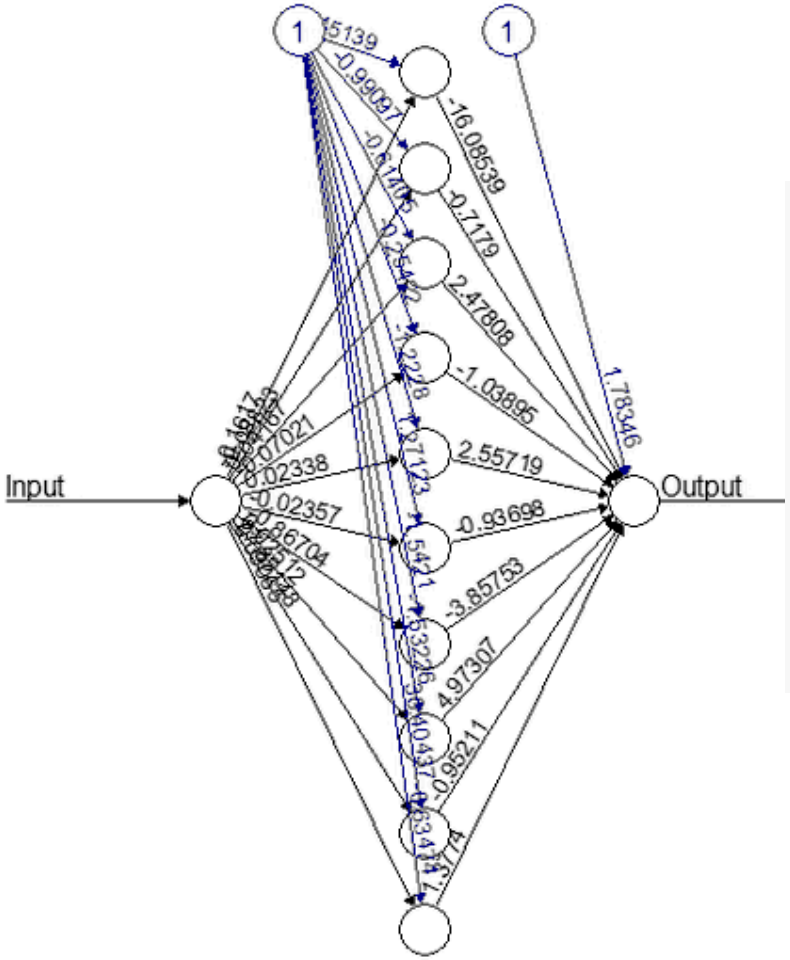
| *Unit j* | *Err$_j$* |
|---|---|
| 6 | $(0.474)(1-0.474)(1-0.474)=0.1311$ |
| 5 | $(0.525)(1-0.525)(0.1311)(-0.2)=-0.0065$ |
| 4 | $(0.332)(1-0.332)(0.1311)(-0.3)=-0.0087$ |

# 4.4. Phân loại dữ liệu với mạng Neural

Calculations for weight and bias updating.

| *Weight or bias* | *New value* |
|---|---|
| $w_{46}$ | $-0.3 + (0.9)(0.1311)(0.332) = -0.261$ |
| $w_{56}$ | $-0.2 + (0.9)(0.1311)(0.525) = -0.138$ |
| $w_{14}$ | $0.2 + (0.9)(-0.0087)(1) = 0.192$ |
| $w_{15}$ | $-0.3 + (0.9)(-0.0065)(1) = -0.306$ |
| $w_{24}$ | $0.4 + (0.9)(-0.0087)(0) = 0.4$ |
| $w_{25}$ | $0.1 + (0.9)(-0.0065)(0) = 0.1$ |
| $w_{34}$ | $-0.5 + (0.9)(-0.0087)(1) = -0.508$ |
| $w_{35}$ | $0.2 + (0.9)(-0.0065)(1) = 0.194$ |
| $\theta_6$ | $0.1 + (0.9)(0.1311) = 0.218$ |
| $\theta_5$ | $0.2 + (0.9)(-0.0065) = 0.194$ |
| $\theta_4$ | $-0.4 + (0.9)(-0.0087) = -0.408$ |

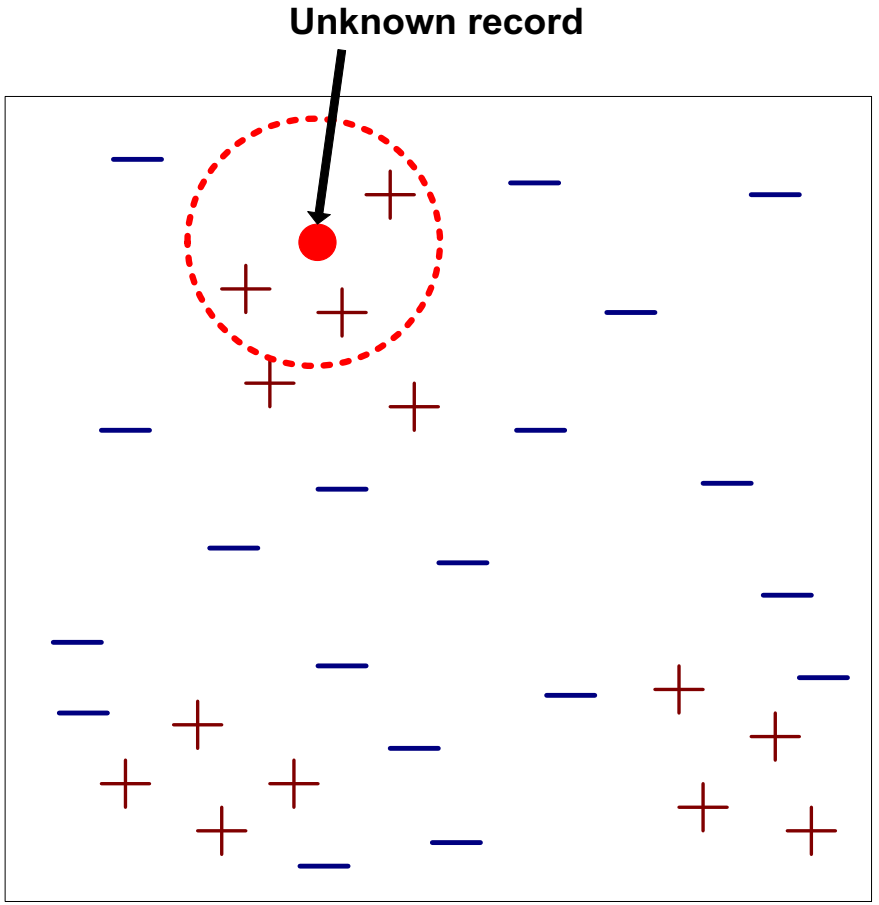
# 4.4. Phân loại dữ liệu với mạng Neural




| Input | Expected Output | Neural Net Output |
|---|---|---|
| 1 | 1 | 0.9623402772 |
| 4 | 2 | 2.0083461217 |
| 9 | 3 | 2.9958221776 |
| 16 | 4 | 4.0009548085 |
| 25 | 5 | 5.0028838579 |
| 36 | 6 | 5.9975810435 |
| 49 | 7 | 6.9968278722 |
| 64 | 8 | 8.0070028670 |
| 81 | 9 | 9.0019220736 |
| 100 | 10 | 9.9222007864 |

# 4.5. Các phương pháp phân loại dữ liệu khác

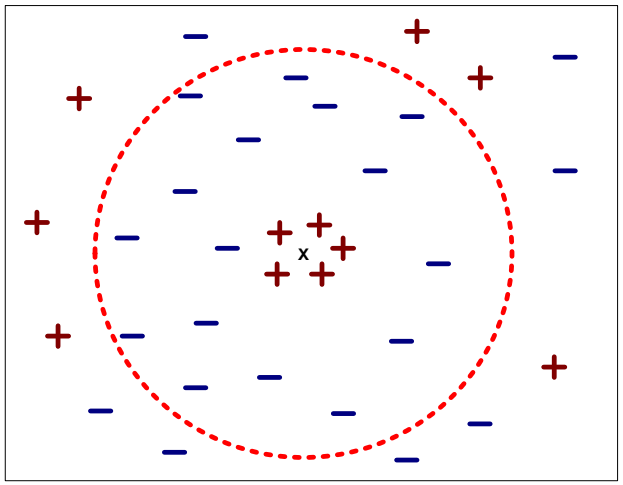
- Phân loại k-nn (k-nearest neighbor)
  - Cho trước tập dữ liệu huấn luyện D với các lớp, phân loại record/object X vào các lớp dựa vào k phần tử tương tự với X nhất (dùng luật số đông: majority vote)
  - Phụ thuộc
    - Độ đo khoảng cách để xác định sự tương tự.
    - Trị k, số phần tử láng giềng → $k <= |D|^{1/2}$



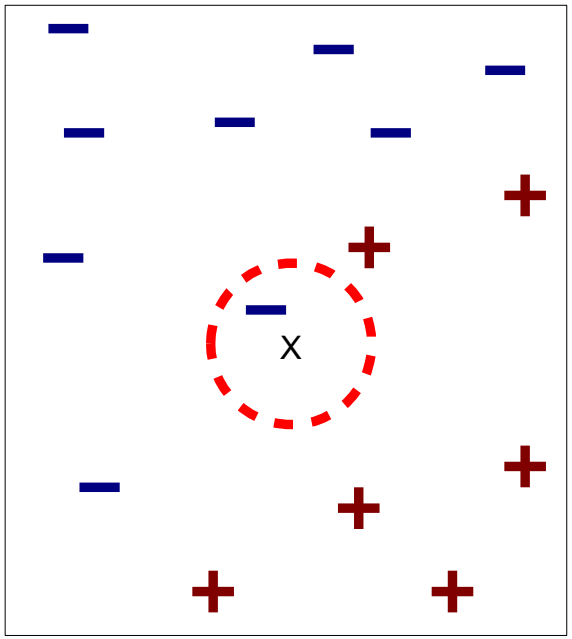
# 4.5. Các phương pháp phân loại dữ liệu khác

- Chọn độ đo
  - Độ đo Euclidean $$d(p,q) = \sqrt{\sum_i (p_i - q_i)^2}$$
- Chọn trị k
  - Nếu k quá nhỏ thì kết quả dễ bị ảnh hưởng bởi nhiễu.
  - Nếu k quá lớn thì nhiều phần tử láng giềng chọn được có thể đến từ các lớp khác.

k quá lớn!

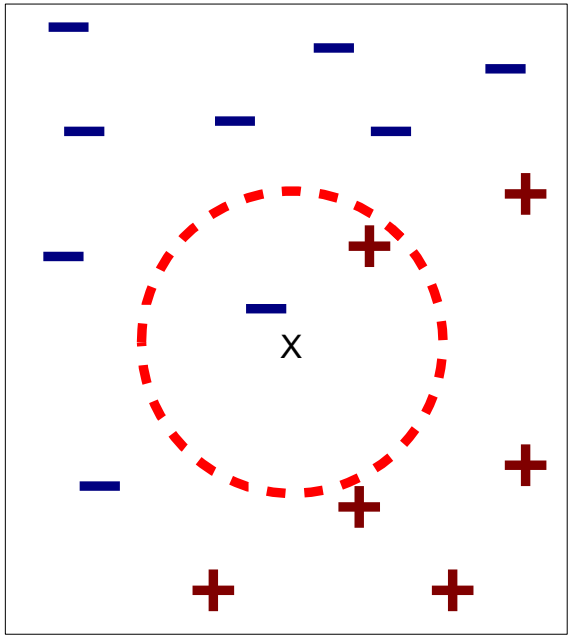


# 4.5. Các phương pháp phân loại dữ liệu khác



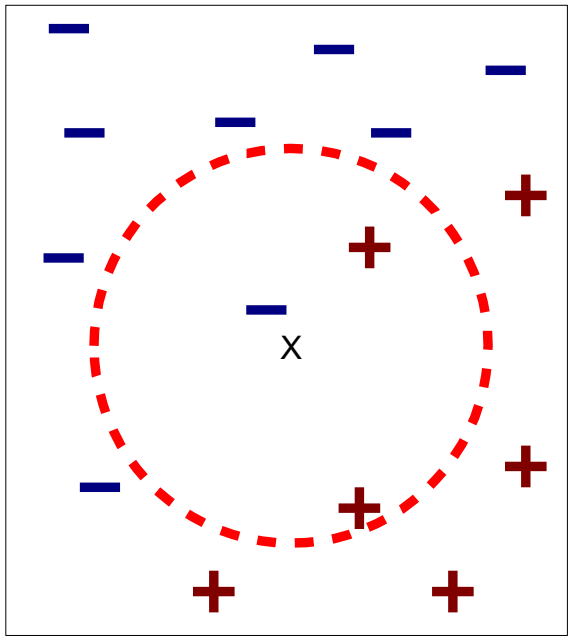

(a) 1-nearest neighbor

X ∈ MINUS




(b) 2-nearest neighbor

X ∈ MINUS
hay
X ∈ PLUS ?




(c) 3-nearest neighbor

X ∈ PLUS

# 4.6. Tóm tắt

- Classification với Decision trees
  - ID3, C4.5, CART
- Classification với mạng Bayesian
  - Dựa trên lý thuyết xác suất thống kê
- Classification với mạng Neural
- K-nn classification
  - Dựa trên khoảng cách

1. **x'**=(Outlook=*Sunny*, Temperature=*Cool*, Humidity=*High*, Wind=*Strong*)

*PlayTennis*: training examples

| Day | Outlook | Temperature | Humidity | Wind | PlayTennis |
|---|---|---|---|---|---|
| D1 | Sunny | Hot | High | Weak | No |
| D2 | Sunny | Hot | High | Strong | No |
| D3 | Overcast | Hot | High | Weak | Yes |
| D4 | Rain | Mild | High | Weak | Yes |
| D5 | Rain | Cool | Normal | Weak | Yes |
| D6 | Rain | Cool | Normal | Strong | No |
| D7 | Overcast | Cool | Normal | Strong | Yes |
| D8 | Sunny | Mild | High | Weak | No |
| D9 | Sunny | Cool | Normal | Weak | Yes |
| D10 | Rain | Mild | Normal | Weak | Yes |
| D11 | Sunny | Mild | Normal | Strong | Yes |
| D12 | Overcast | Mild | High | Strong | Yes |
| D13 | Overcast | Hot | Normal | Weak | Yes |
| D14 | Rain | Mild | High | Strong | No |

# Decision tree

| Thời_tiết | Nhiệt_độ | Độ_Ẩm | Gió | Đi bơi |
|---|---|---|---|---|
| Nắng | Nóng | Cao | Không | Không |
| Nắng | Nóng | Cao | Có | Không |
| U ám | Nóng | Cao | Không | Có |
| Mưa | Ấm áp | Cao | Không | Có |
| Mưa | Mát | Vừa | Không | Có |
| Mưa | Mát | Vừa | Có | Không |
| U ám | Mát | Vừa | Có | Có |
| Nắng | Ấm áp | Cao | Không | Không |
| Nắng | Mát | Vừa | Không | Có |
| Mưa | Ấm áp | Vừa | Không | Có |
| Nắng | Ấm áp | Vừa | Có | Có |
| U ám | Ấm áp | Cao | Có | Không |

Dùng độ đo information gain để xây dựng cây quyết định đến cấp thứ 2.