

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

Chương 4

Phân Loại Dữ Liệu – Data Classification

TRAN MINH QUANG

quangtran@hcmut.edu.vn

<http://www.cse.hcmut.edu.vn/staff/Staff/quangtran>

<http://researchmap.jp/quang>

1

NỘI DUNG

1. Tổng quan về phân loại dữ liệu
2. Hồi qui logistic
3. Cây quyết định
4. Mạng Bayesian
5. Mạng Neural
6. Các phương pháp phân loại dữ liệu khác
7. Đánh giá và chọn mô hình phân loại
8. Tóm tắt

TÀI LIỆU THAM KHẢO

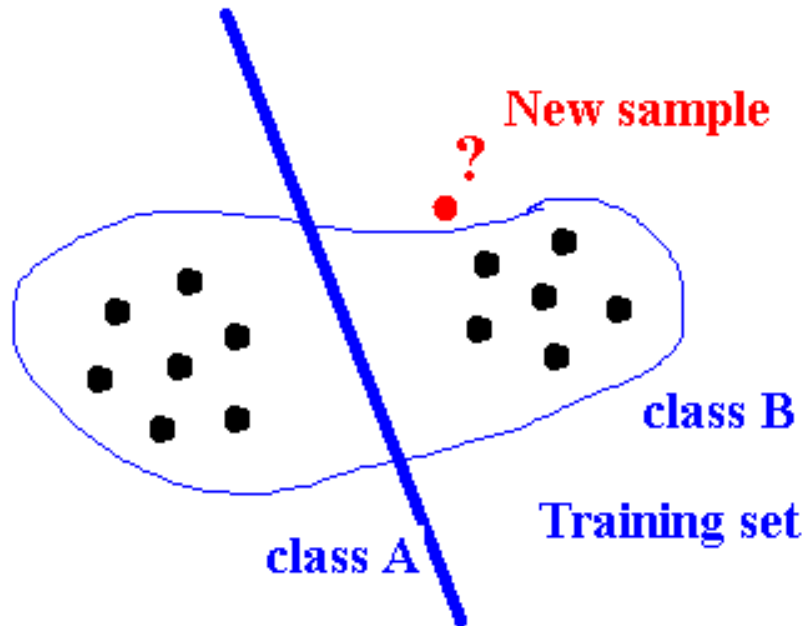
- [1] Jiawei Han, Micheline Kamber, and Jian Pei, “Data Mining: Concepts and Techniques”, 3rd Edition, Morgan Kaufmann Publishers, 2012.
- [2] David Hand, Heikki Mannila, Padhraic Smyth, “Principles of Data Mining”, MIT Press, 2001.
- [3] David L. Olson, Dursun Delen, “Advanced Data Mining Techniques”, Springer-Verlag, 2008.
- [4] Graham J. Williams, Simeon J. Simoff, “Data Mining: Theory, Methodology, Techniques, and Applications”, Springer-Verlag, 2006.
- [5] ZhaoHui Tang, Jamie MacLennan, “Data Mining with SQL Server 2005”, Wiley Publishing, 2005.
- [6] Oracle, “Data Mining Concepts”, B28129-01, 2008.
- [7] Oracle, “Data Mining Application Developer’s Guide”, B28131-01, 2008.
- [8] Ian H.Witten, Eibe Frank, “Data mining : practical machine learning tools and techniques”, 2nd Edition, Elsevier Inc, 2005.
- [9] Florent Messegli, Pascal Poncelet & Maguelonne Teisseire, “Successes and new directions in data mining”, IGI Global, 2008.
- [10] Oded Maimon, Lior Rokach, “Data Mining and Knowledge Discovery Handbook”, 2nd Edition, Springer Science + Business Media, LLC 2005, 2010.

1. TỔNG QUAN VỀ PHÂN LOẠI DL

○ Các tình huống

- Email: “spam” hay “bình thường”
- Các giao dịch trực tuyến: “gian lận” hay “thông thường”
- Y tế: “bị bệnh” hay “không bị bệnh”; khối u “lành tính” hay “ác tính”,...
- $Y \in \{0, 1\}$: 0: “negative” hay 1: “positive” classes
- $Y \in \{0, 1, 2, 3\}$: Dữ liệu thuộc nhiều lớp
- Mỗi lớp có gắn một nhãn (label): Ví dụ “spam” hay “not spam”

1. TỔNG QUAN VỀ PHÂN LOẠI DL



- Cho trước tập huấn luyện (training set), dẫn ra mô tả về class A và B
- Cho trước mẫu/đối tượng mới, xác định class mà mẫu đó thuộc về?
- Liệu class đó có thực sự phù hợp/đúng cho mẫu/đối tượng đó?

1. TỔNG QUAN VỀ PHÂN LOẠI DL

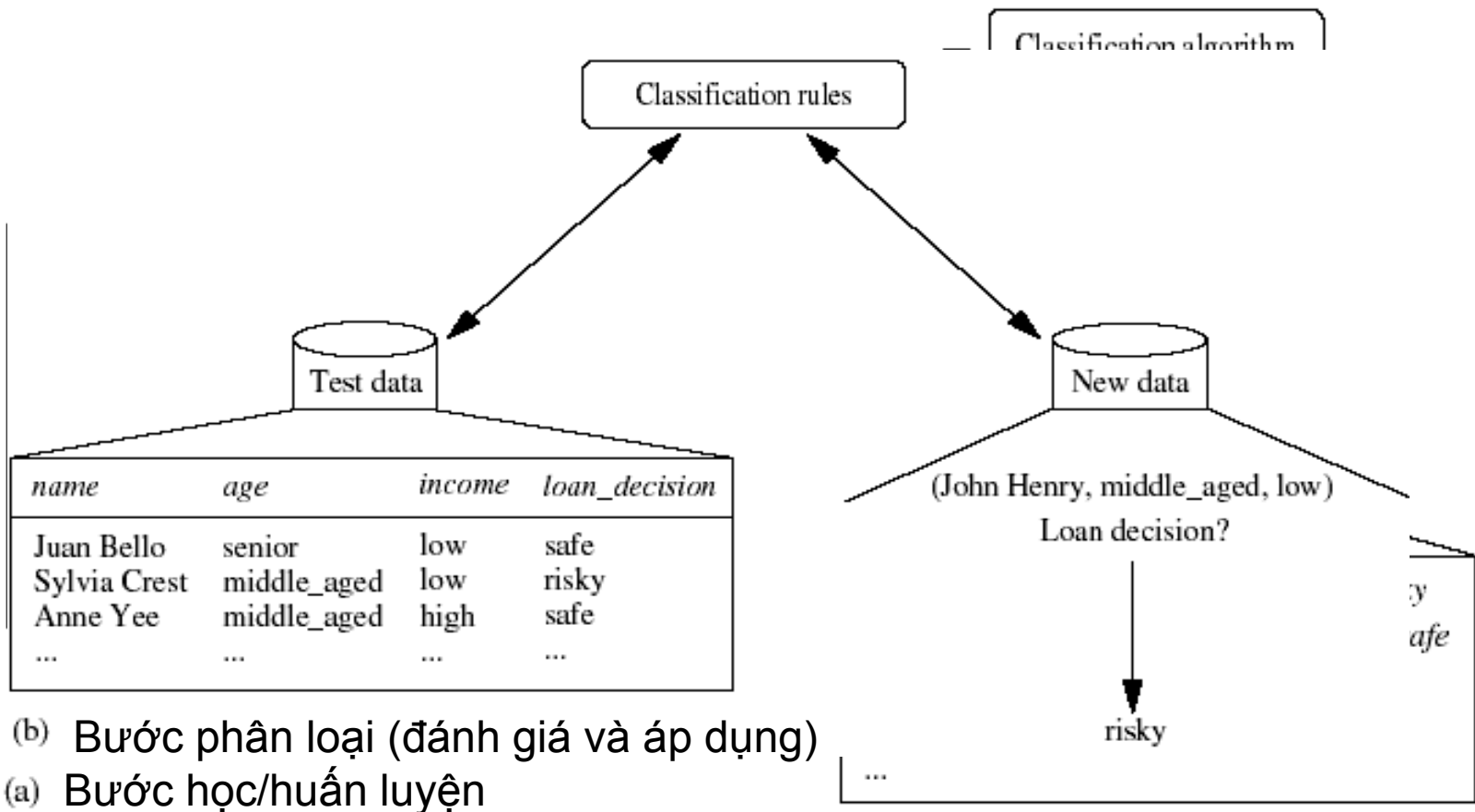
○ Phân loại dữ liệu (classification)

- 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 (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

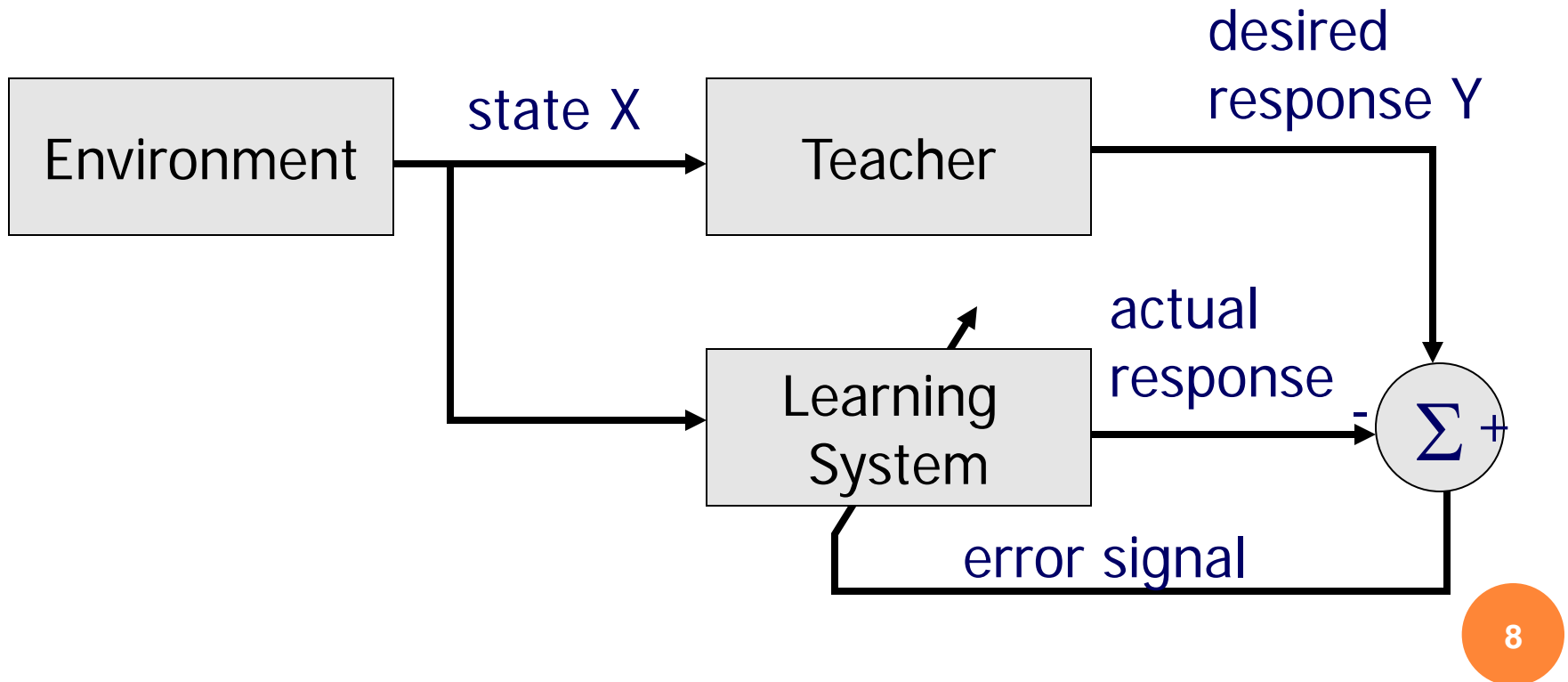
- **Bước học:** X trong tập huấn luyện, một trị y được cho trước với $X \rightarrow$ xác định f
- **Bước phân loại:** đánh giá f với (X', y') và $X' \leftrightarrow$ mọi X trong tập huấn luyện; nếu chấp nhận được thì dùng f để xác định y'' cho X'' (mới)

1. TỔNG QUAN VỀ PHÂN LOẠI DL



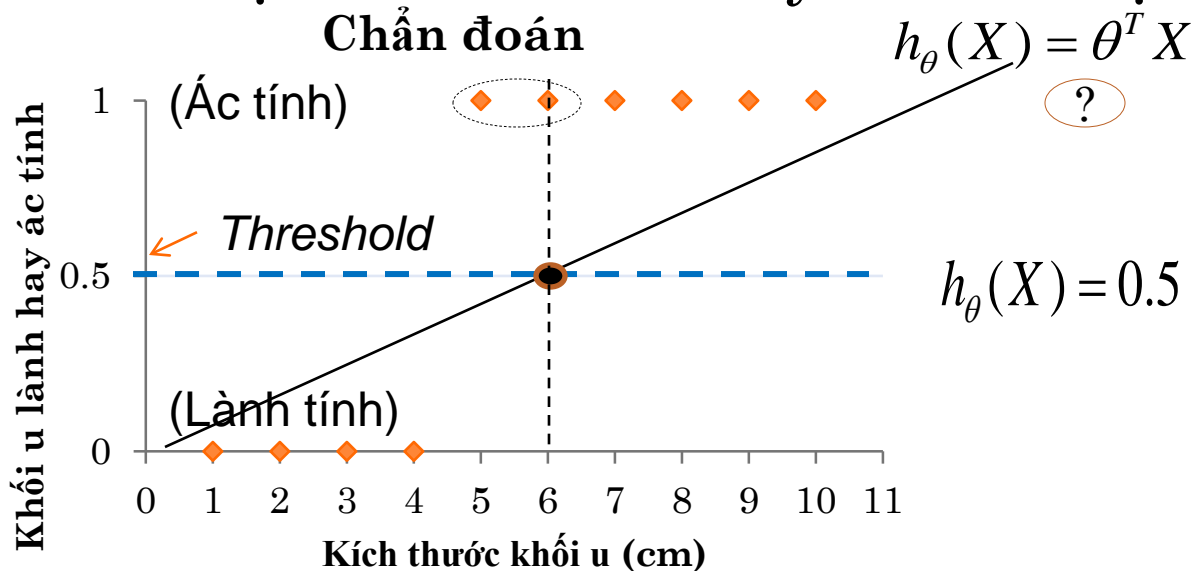
1. TỔNG QUAN VỀ PHÂN LOẠI DL

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



1. TỔNG QUAN VỀ PHÂN LOẠI DL

- Phân loại khối u lành hay ác tính dựa vào kích thước



- Nếu $h_{\theta}(X) \geq 0.5$ thì dự đoán: “Y=1” ngược lại “Y=0”
- Thực tế, $h_{\theta}(X) > 1$ hoặc $h_{\theta}(X) < 0$
- Hồi qui logistic (Logistic regression) $0 \leq h_{\theta}(X) \leq 1$

=> Phân loại (Classification)

1. TỔNG QUAN VỀ PHÂN LOẠI DL

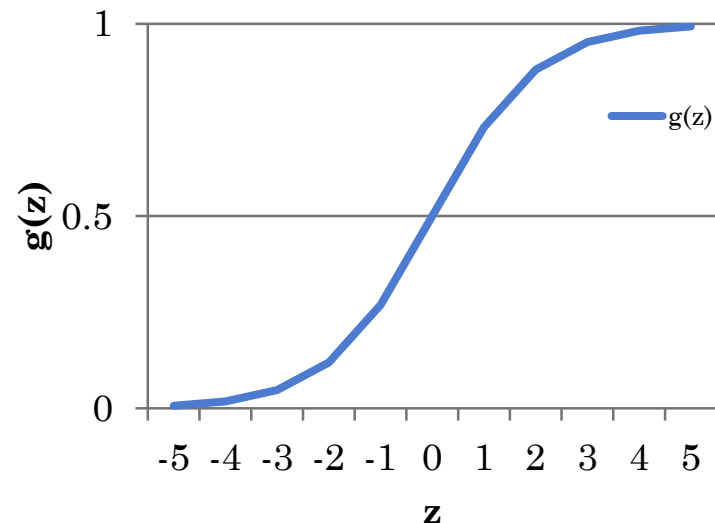
- Các giải thuật phân loại dữ liệu
 - Hồi qui logistic (logistic regression)
 - Cây quyết định (decision tree)
 - Mạng Bayesian
 - Mạng neural nhân tạo (ANN)
 - Phân loại với k phần tử cận gần nhất (k-nearest neighbor)
 - Phân loại với suy diễn dựa trên tình huống (case-based reasoning)
 - Phân loại 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) ...

2. PHÂN LOẠI DL VỚI HỒI QUI LOGISTIC

- $h_{\theta}(X) = \theta^T X$ (có thể lớn hơn 1 hoặc nhỏ hơn 0)
- Cần có $h_{\theta}(X)$ sao cho $0 \leq h_{\theta}(X) \leq 1$
- Mô hình lại: $h_{\theta}(X) = g(\theta^T X)$ với $g(z) = \frac{1}{1 + e^{-z}}$

$$h_{\theta}(X) = \frac{1}{1 + e^{-\theta^T X}}$$

- Sigmoid function
hay Logistic function



Liên quan đến bộ tham số θ

2. PHÂN LOẠI DL VỚI HỒI QUI LOGISTIC

○ Giải thích giá trị của $h_{\theta}(X) = \frac{1}{1 + e^{-\theta^T X}}$

● Là xác suất dự đoán rằng “ $y=1$ ” với input là x

● Ex.,
$$x = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 1 \\ kt.khoi_u \end{bmatrix}$$

$$h_{\theta}(x) = 0.7$$

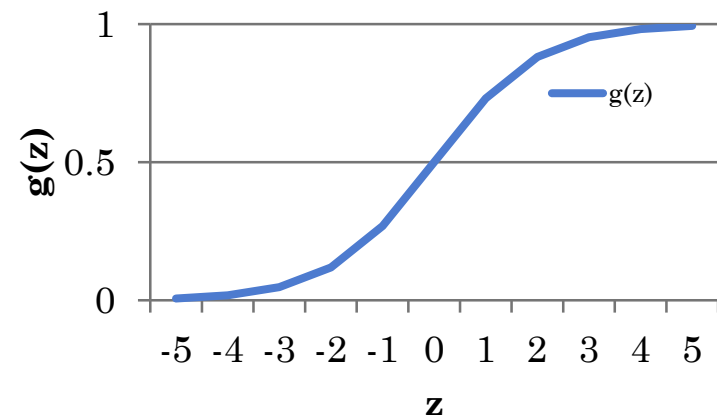
⇒ 70% khối u với k.thước đã cho có thể là ác tính

⇒ $h_{\theta}(x) = P(y=1 | x; \theta)$ (xác suất $y=1$, với x cho trước và được thông số hóa bởi θ)

2. PHÂN LOẠI DL VỚI HỒI QUI LOGISTIC

- Chú ý $h_{\theta}(X) = g(\theta^T X)$ với $g(z) = \frac{1}{1 + e^{-z}}$ hay

$$h_{\theta}(X) = \frac{1}{1 + e^{-\theta^T X}}$$



- $g(z) \geq 0.5$ khi $z \geq 0$
- $g(z) < 0.5$ khi $z < 0$
- dự đoán $y=1$ khi $h_{\theta}(X) \geq 0.5$ hay $\theta^T X \geq 0$
- dự đoán $y=0$ khi $h_{\theta}(X) < 0.5$ hay $\theta^T X < 0$

2. PHÂN LOẠI DL VỚI HỒI QUI LOGISTIC

- Đường phân lớp (decision boundary)

- $h_{\theta}(X) = g(\theta^T X) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$

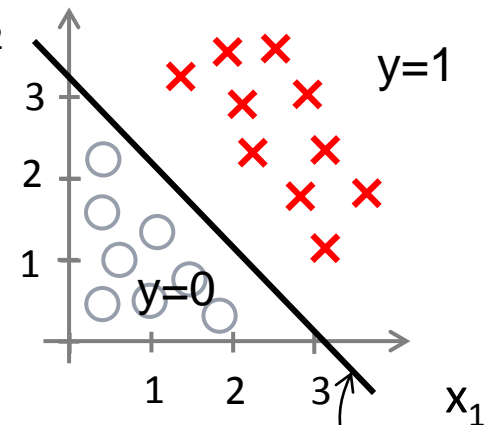
- Chọn

$$\theta = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix}$$

- Dự đoán “ $y=1$ ” nếu $\theta^T X \geq 0$

hay $-3 + x_1 + x_2 \geq 0$

$\Rightarrow x_1 + x_2 \geq 3$



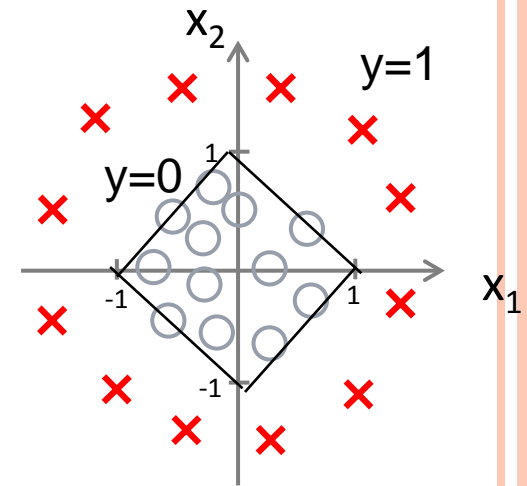
Decision boundary

2. PHÂN LOẠI DL VỚI HỒI QUI LOGISTIC

- Đường phân lớp (decision boundary)

- $h_{\theta}(X) = g(\theta^T X) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2)$

- Dự đoán “ $y=1$ ” nếu $\theta^T X \geq 0$
hay $-1 + x_1^2 + x_2^2 \geq 0$



2. PHÂN LOẠI DL VỚI HỒI QUI LOGISTIC

- Cost function của hồi qui logistic

- Training set; $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(N)}, y^{(N)})\}$

- N examples

$$x = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}; x_0 = 1; y \in \{0, 1\}$$

$$h_{\theta}(X) = \frac{1}{1 + e^{-\theta^T X}}$$

- Làm sao để chọn bộ thông số θ ?

2. PHÂN LOẠI DL VỚI HỒI QUI LOGISTIC

- Liên hệ với hồi qui tuyến tính $J(\theta) = \frac{1}{2N} \sum_{i=1}^N (h_{\theta}(x^{(i)}) - y^{(i)})^2$

- Trong hồi qui phi tuyến

$$J(\theta) = \text{cost}(h_{\theta}(x), y)$$

$$\text{cost}(h_{\theta}(x^{(i)}), y^{(i)}) = \frac{1}{2} (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

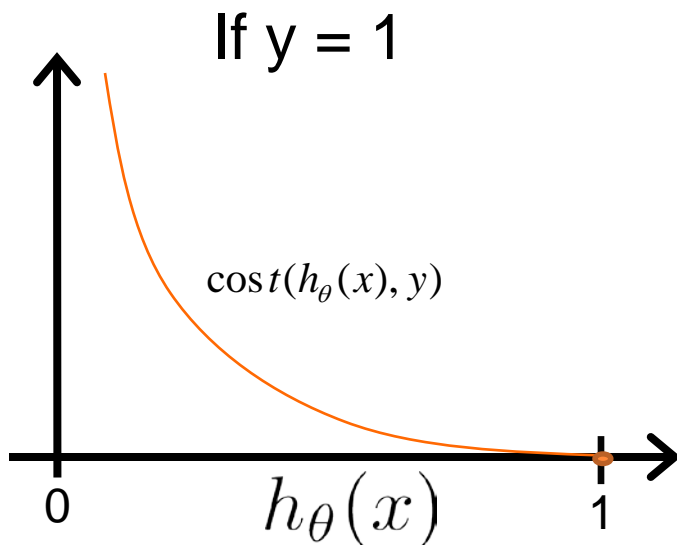
Để đơn giản hóa ta ghi

$$\text{cost}(h_{\theta}(x), y) = \frac{1}{2} (h_{\theta}(x) - y)^2$$

2. PHÂN LOẠI DL VỚI HỒI QUI LOGISTIC

- Cost function của hồi qui logistic

$$\text{cost}(h_{\theta}(x), y) = \begin{cases} -\log(h_{\theta}(x)) : y = 1 \\ -\log(1 - h_{\theta}(x)) : y = 0 \end{cases}$$



- Cost = 0 nếu $y=1$, $h_{\theta}(x)=1$
- Khi $h_{\theta}(x) \rightarrow 0$ thì cost $\rightarrow \infty$

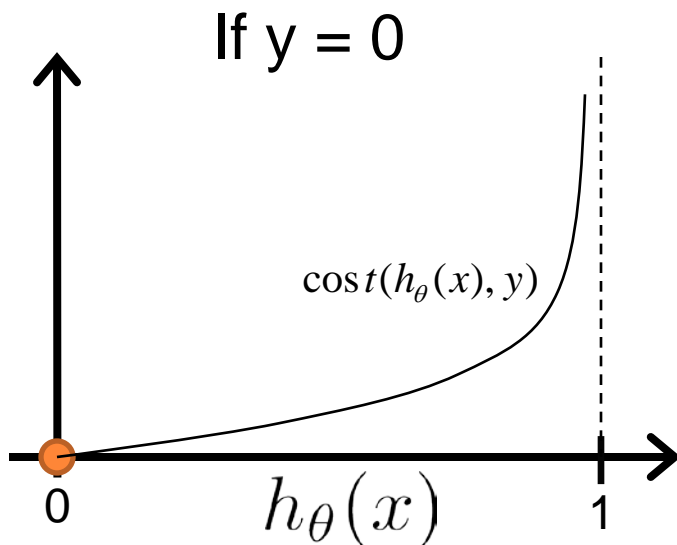
\Rightarrow Khi $h_{\theta}(x)=0$ có nghĩa là ta dự đoán rằng:

$P(y=1|x, \theta)=0$, trong khi đó $y=1$, do vậy chi phí của giải thuật trong trường hợp này là rất lớn

2. PHÂN LOẠI DL VỚI HỒI QUI LOGISTIC

- Cost function của hồi qui logistic

$$\text{cost}(h_{\theta}(x), y) = \begin{cases} -\log(h_{\theta}(x)) : y = 1 \\ -\log(1 - h_{\theta}(x)) : y = 0 \end{cases}$$



- Cost = 0 nếu $y=0$, $h_{\theta}(x)=0$
- Khi $h_{\theta}(x) \rightarrow 1$ thì cost $\rightarrow \infty$

\Rightarrow Khi $h_{\theta}(x)=1$ có nghĩa là ta dự đoán rằng:

$P(y=1|x, \theta)=1$, trong khi đó $y=0$, do vậy chi phí của giải thuật trong trường hợp này là rất lớn

2. PHÂN LOẠI DL VỚI HỒI QUI LOGISTIC

- Đơn giản hàm chi phí và giải thuật gradient descent

$$\text{cost}(h_{\theta}(x), y) = \begin{cases} -\log(h_{\theta}(x)) : y = 1 \\ -\log(1 - h_{\theta}(x)) : y = 0 \end{cases}$$

- Do $y=0|1$, nên hàm chi phí có thể đơn giản hóa như sau

$$\begin{aligned} \text{cost}(h_{\theta}(x), y) &= -y \log(h_{\theta}(x)) - (1-y) \log(1-h_{\theta}(x)) \\ J(\theta) &= \frac{1}{N} \sum_{i=1}^N \text{cost}(h_{\theta}(x^{(i)}) - y^{(i)}) = -\frac{1}{N} \sum_{i=1}^N y^{(i)} \log h_{\theta}(x^{(i)}) + (1-y) \log(1-h_{\theta}(x^{(i)})) \end{aligned}$$

- Thực hiện tìm $\min_{\theta} J(\theta)$ ta sẽ tìm được bộ thông số θ (giải thuật gradient descent)

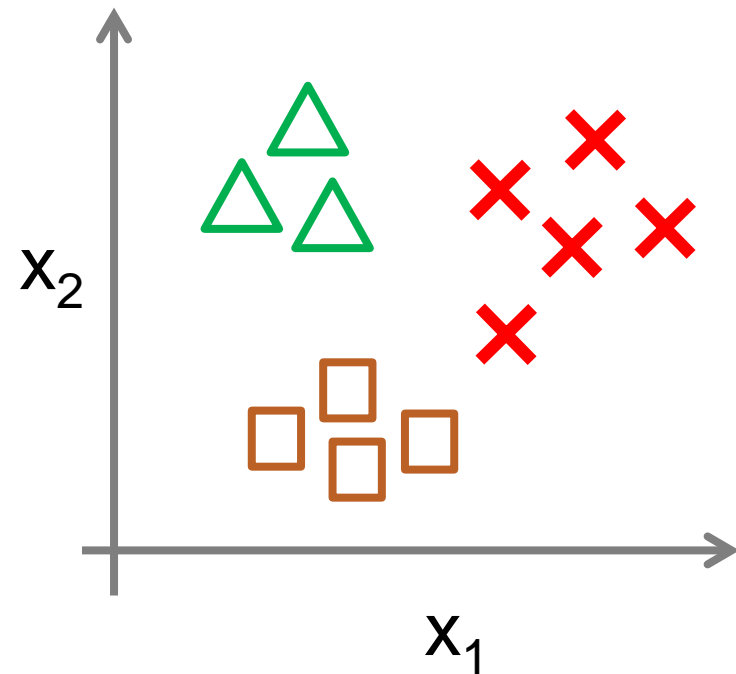
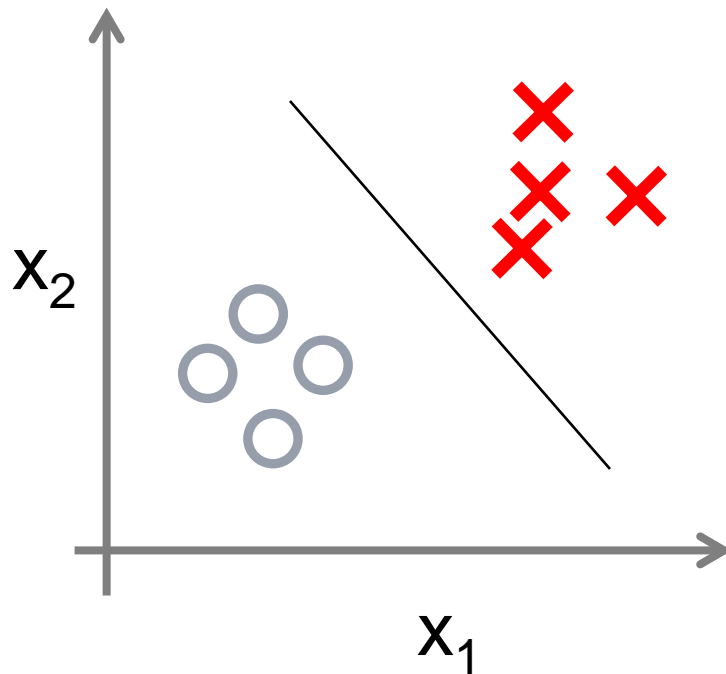
- Để dự đoán giá trị của y dựa vào giá trị x (mới đưa vào):
$$h_{\theta}(X) = \frac{1}{1 + e^{-\theta^T X}}$$

2. PHÂN LOẠI DL VỚI HỒI QUI LOGISTIC

- Hồi qui logistic để phân lớp tập dữ liệu có nhị nguyên (thuộc nhị nguyên lớp):
 - Thư mục email: “business”, “friend”, “family”, “hobby” ($y=\{1,2,3,4\}$)
 - Chẩn đoán: “cảm cúm”, “sốt siêu vi”, “rubella” ($y=\{1,2,3\}$)
 - Dự báo thời tiết: “nắng”, “nhị nguyên mây”, “mưa” ($y=\{1,2,3\}$)

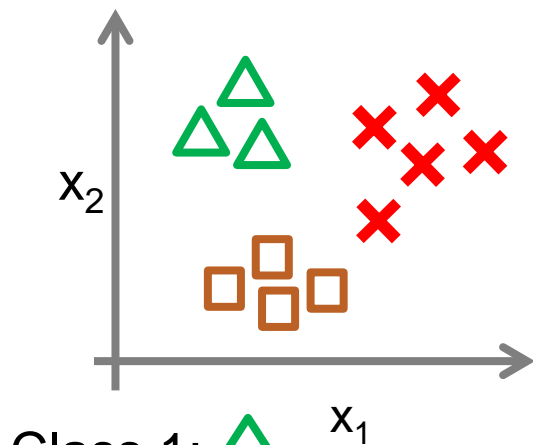
2. PHÂN LOẠI DL VỚI HỒI QUI LOGISTIC




- Dữ liệu đa lớp (multi-class)

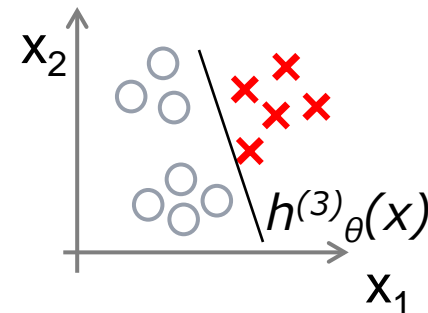
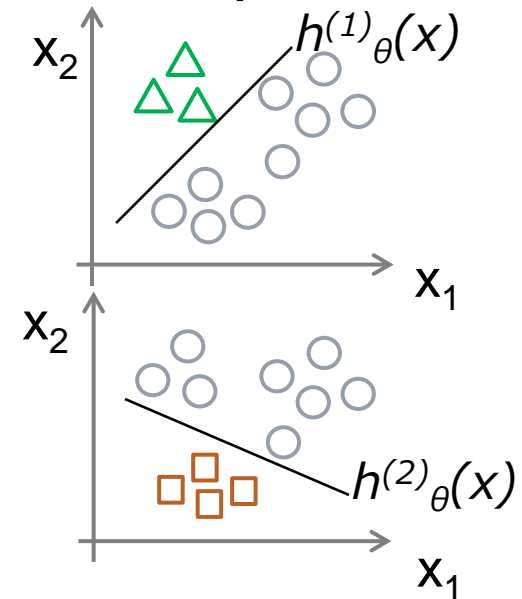


2. PHÂN LOẠI DL VỚI HỒI QUI LOGISTIC

- Dữ liệu đa lớp (multi-class): một và phần còn lại



Class 1:  x_1
Class 2: 
Class 3: 



$$h^{(i)}_{\theta}(x) = P(y=1|x; \theta) \text{ với } (i=1,2,...k), k \text{ là số lớp}$$

2. PHÂN LOẠI DL VỚI HỒI QUI LOGISTIC

- Huấn luyện bộ phân lớp hồi qui logistic $h^{(i)}_{\theta}(x)$ cho mỗi lớp i
- Với một phần tử x mới đưa vào, ta dự đoán y bằng cách chọn lớp i sao cho $h^{(i)}_{\theta}(x)$ là lớn nhất

$$h^{(i)}_{\theta}(x) = P(y=1 | x; \theta) \quad \text{với } (i=1,2,..k), k \text{ là số lớp}$$

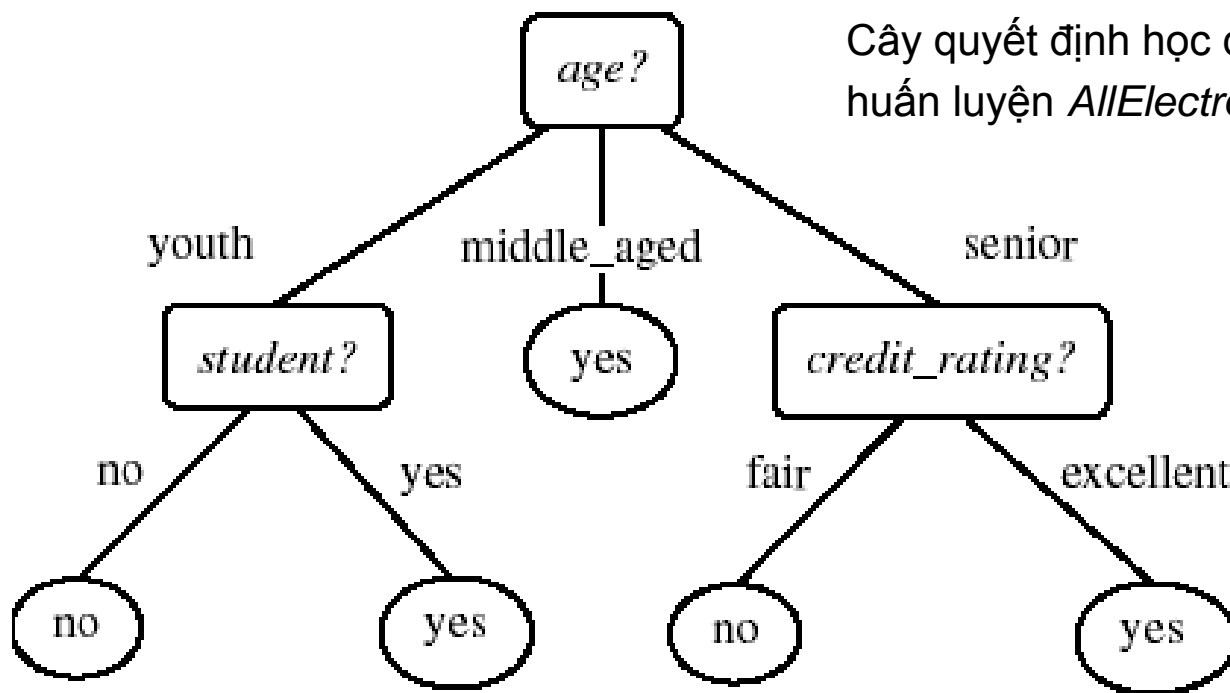
3. PHÂN LOẠI VỚI CÂY QUYẾT ĐỊNH

<i>RID</i>	<i>age</i>	<i>income</i>	<i>student</i>	<i>credit_rating</i>	<i>Class: buys_computer</i>
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

3. PHÂN LOẠI 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*

3. PHÂN LOẠI 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.

3. PHÂN LOẠI 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) $attribute_list \leftarrow attribute_list - splitting_attribute$; // 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 ;

3. PHÂN LOẠI VỚI CÂY QUYẾT ĐỊNH

○ Đặc điểm của giải thuật

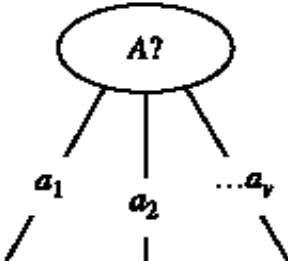

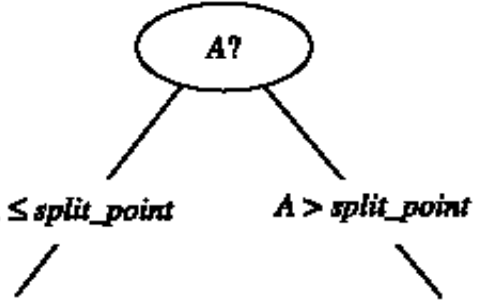
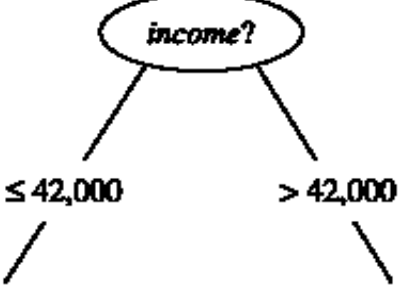
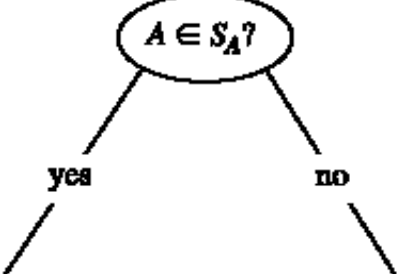
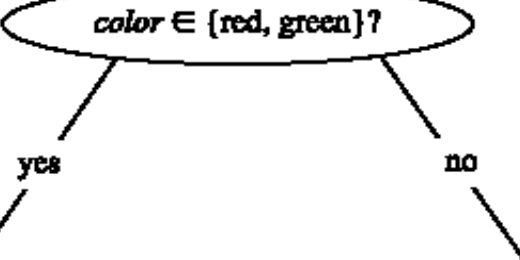
- Giải thuật tham lam (không có quay lui), chia để trị, đệ qui, từ trên xuống
- Độ phức tạp: $O(n * |D| * \log |D|)$
 - Mỗi thuộc tính ứng với mỗi mức (level) của cây
 - Ở mỗi mức của cây, $|D|$ phần tử huấn luyện được duyệt qua
 - In-memory $\rightarrow ???$

3. PHÂN LOẠI VỚI CÂY QUYẾT ĐỊNH

○ Attribute_selection_method

- 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**

3. PHÂN LOẠI VỚI CÂY QUYẾT ĐỊNH

	Partitioning Scenarios	Examples
a)		
b)		
c)		

A là thuộc tính phân tách (splitting attribute).

3. PHÂN LOẠI 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
 - N: node hiện tại cần phân hoạch các phần tử trong D
 - Splitting attribute đảm bảo sự trùng lặp (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

3. PHÂN LOẠI 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

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

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

3. PHÂN LOẠI 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 $\{D_1, D_2, \dots, D_j, \dots, D_v\}$
 - Mỗi phân hoạch D_j gồm $|D_j|$ phần tử trong D
 - Lượng thông tin này sẽ cho biết mức độ trùng lặp 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**

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

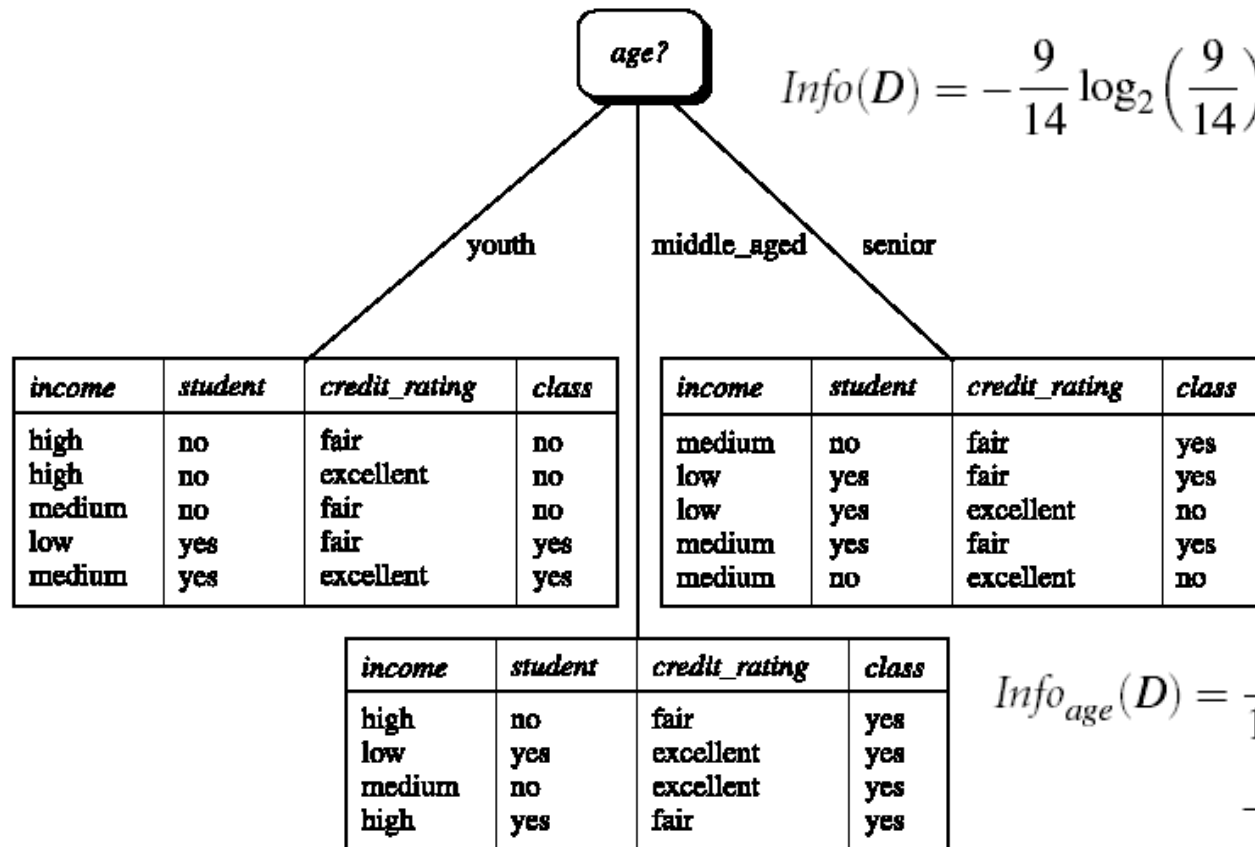
3. PHÂN LOẠI VỚI CÂY QUYẾT ĐỊNH

○ 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)

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

3. PHÂN LOẠI VỚI CÂY QUYẾT ĐỊNH



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

Gain(age)=0.246 bits

Gain(income)?

Gain(student)?

Gain(credit_rating)?

→ Splitting attribute?

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

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

3. PHÂN LOẠI 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

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

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

3. PHÂN LOẠI VỚI CÂY QUYẾT ĐỊNH

$$\begin{aligned}\text{SplitInfo}_{\text{income}}(D) &= -\frac{4}{14} \times \log_2\left(\frac{4}{14}\right) - \frac{6}{14} \times \log_2\left(\frac{6}{14}\right) - \frac{4}{14} \times \log_2\left(\frac{4}{14}\right) \\ &= 0.926.\end{aligned}$$

38

$$\text{Gain}(\text{income}) = 0.029$$

$$\text{GainRatio}(\text{income}) = 0.029/0.926 = 0.031$$

GainRatio(age)?

GainRatio(student)?

GainRatio(credit_rating)?

→ Splitting attribute?



3. PHÂN LOẠI VỚI CÂY QUYẾT ĐỊNH

○ Độ đo Gini Index

- Dùng với CART
- Sự phân tách nhị phân (binary split) cho mỗi t.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 lặp giữa các phân hoạch

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

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

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

3. PHÂN LOẠI VỚI CÂY QUYẾT ĐỊNH

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

$$\begin{aligned} Gini_{income \in \{low, medium\}}(D) &= \frac{10}{14} Gini(D_1) + \frac{4}{14} Gini(D_2) \\ &= \frac{10}{14} \left(1 - \left(\frac{6}{10}\right)^2 - \left(\frac{4}{10}\right)^2 \right) + \frac{4}{14} \left(1 - \left(\frac{1}{4}\right)^2 - \left(\frac{3}{4}\right)^2 \right) \\ &= 0.450 \\ &= Gini_{income \in \{high\}}(D). \end{aligned}$$

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

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

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

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

$$Gini_{student} = 0.367$$

$$Gini_{credit_rating} = 0.429$$

→ Splitting attribute?

3. PHÂN LOẠI 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. PHÂN LOẠI 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



Reverend Thomas Bayes
(1702-1761)

4. PHÂN LOẠI 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. PHÂN LOẠI VỚI MẠNG BAYESIAN

○ Định lý Bayes

- $P(H | X)$: posterior probability
 - Ví dụ: $P(\text{buys_computer}=\text{yes} | \text{age}=\text{young}, \text{income}=\text{high})$ 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
 - Ví dụ: $P(\text{age}=\text{young}, \text{income}=\text{high} | \text{buys_computer}=\text{yes})$ là xác suất khách hàng đã mua máy tính có tuổi “young” và thu nhập “high”
 - $P(\text{age}=\text{young}, \text{income}=\text{high} | \text{buys_computer}=\text{yes}) = 0$
 - $P(\text{age}=\text{young}, \text{income}=\text{high} | \text{buys_computer}=\text{no}) = 2/5 = 0.4$

4. PHÂN LOẠI VỚI MẠNG BAYESIAN

○ Định lý Bayes

- $P(H)$: prior probability, Xác suất của H
 - Ví dụ: $P(\text{buys_computer}=\text{yes})$ là xác suất mua máy tính của khách hàng nói chung
 - $P(\text{buys_computer}=\text{yes}) = 9/14 = 0.643$
 - $P(\text{buys_computer}=\text{no}) = 5/14 = 0.357$
- $P(X)$: prior probability, Xác suất của X
 - Ví dụ: $P(\text{age}=\text{young}, \text{income}=\text{high})$ là xác suất khách hàng có tuổi “young” và thu nhập “high”
 - $P(\text{age}=\text{young}, \text{income}=\text{high}) = 2/14 = 0.143$

4. PHÂN LOẠI VỚI MẠNG BAYESIAN

○ Định lý Bayes

- $P(H)$, $P(X | H)$, $P(X)$ được tính từ tập dữ liệu cho trước
- $P(H | X)$ được tính từ định lý Bayes

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

$P(\text{buys_computer=yes} | \text{age=young, income=high}) = P(\text{age=young, income=high} | \text{buys_computer=yes})P(\text{buys_computer=yes}) / P(\text{age=young, income=high}) = 0$

$P(\text{buys_computer=no} | \text{age=young, income=high}) = P(\text{age=young, income=high} | \text{buys_computer=no})P(\text{buys_computer=no}) / P(\text{age=young, income=high}) = 0.4 * 0.357 / 0.143 = 0.9986$

4. PHÂN LOẠI 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 $X = (x_1, x_2, \dots, x_n)$ với mạng Bayesian như sau:

X được phân loại vào C_i nếu và chỉ nếu

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

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

→ Tối đa hóa $P(C_i | X)$ (i.e. chọn C_i nếu $P(C_i | X)$ là trị lớn nhất)

→ Tối đa hóa $P(X | C_i)P(C_i)$

→ $P(C_1) = P(C_2) = \dots = P(C_m)$ hoặc $P(C_i) = |C_{i,D}| / |D| \dots$

4. PHÂN LOẠI VỚI MẠNG BAYESIAN

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

- $P(X|C_i)$ 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. PHÂN LOẠI VỚI MẠNG BAYESIAN

○ Nếu $P(\mathbf{x}_k | C_i) = 0$ thì $P(X | C_i) = 0!!!$

- Ban đầu

- $P(\mathbf{x}_k | C_i) = |\{X' | \mathbf{x}'_k = \mathbf{x}_k \wedge X' \in C_i\}| / |C_{i,D}|$

- Laplace (Pierre Laplace, nhà toán học Pháp, 1749-1827)

- $P(\mathbf{x}_k | C_i) = (|\{X' | \mathbf{x}'_k = \mathbf{x}_k \wedge X' \in C_i\}| + 1) / (|C_{i,D}| + m)$

Trong đó, m : số lượng giá trị phân biệt trong miền trị của thuộc tính A_k

- z-estimate

- $P(\mathbf{x}_k | C_i) = (|\{X' | \mathbf{x}'_k = \mathbf{x}_k \wedge X' \in C_i\}| + z * P(\mathbf{x}_k)) / (|C_{i,D}| + z)$

4. PHÂN LOẠI VỚI MẠNG BAYESIAN

$X = (\text{age} = \text{youth}, \text{income} = \text{medium}, \text{student} = \text{yes}, \text{credit_rating} = \text{fair})$

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

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

$$P(\text{age} = \text{youth} | \text{buys_computer} = \text{yes}) = 2/9 = 0.222$$

$$P(\text{age} = \text{youth} | \text{buys_computer} = \text{no}) = 3/5 = 0.600$$

$$P(\text{income} = \text{medium} | \text{buys_computer} = \text{yes}) = 4/9 = 0.444$$

$$P(\text{income} = \text{medium} | \text{buys_computer} = \text{no}) = 2/5 = 0.400$$

$$P(\text{student} = \text{yes} | \text{buys_computer} = \text{yes}) = 6/9 = 0.667$$

$$P(\text{student} = \text{yes} | \text{buys_computer} = \text{no}) = 1/5 = 0.200$$

$$P(\text{credit_rating} = \text{fair} | \text{buys_computer} = \text{yes}) = 6/9 = 0.667$$

$$P(\text{credit_rating} = \text{fair} | \text{buys_computer} = \text{no}) = 2/5 = 0.400$$

$$\begin{aligned} P(X | \text{buys_computer} = \text{yes}) &= P(\text{age} = \text{youth} | \text{buys_computer} = \text{yes}) \times \\ &\quad P(\text{income} = \text{medium} | \text{buys_computer} = \text{yes}) \times \\ &\quad P(\text{student} = \text{yes} | \text{buys_computer} = \text{yes}) \times \\ &\quad P(\text{credit_rating} = \text{fair} | \text{buys_computer} = \text{yes}) \\ &= 0.222 \times 0.444 \times 0.667 \times 0.667 = 0.044. \end{aligned}$$

$$P(X | \text{buys_computer} = \text{no}) = 0.600 \times 0.400 \times 0.200 \times 0.400 = 0.019.$$

$$P(X | \text{buys_computer} = \text{yes})P(\text{buys_computer} = \text{yes}) = 0.044 \times 0.643 = 0.028$$

$$P(X | \text{buys_computer} = \text{no})P(\text{buys_computer} = \text{no}) = 0.019 \times 0.357 = 0.007$$

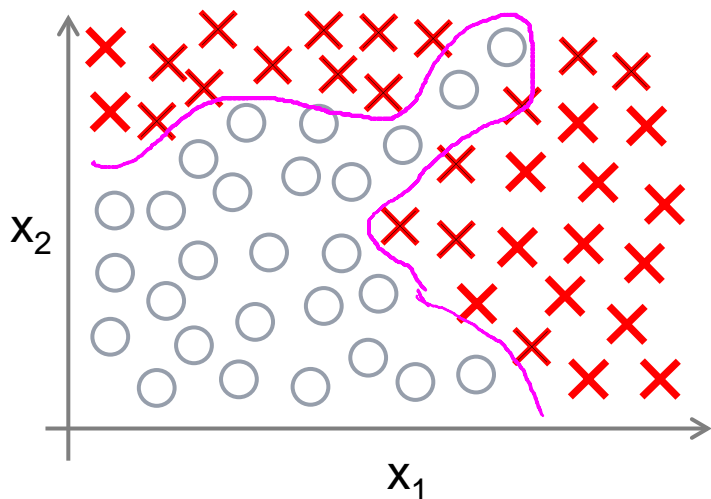
$$P(\text{buys_computer} = \text{yes}) = 9/14 = 0.643$$

$$P(\text{buys_computer} = \text{no}) = 5/14 = 0.357$$

$\rightarrow X \in C_1$

4. PHÂN LOẠI VỚI MẠNG NEURAL

○ Non-linear Classification



x1: kích thước
x2: số p. ngủ
x3: số tầng
x4: tuổi
....
X100:.....

} n=100

$$g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1 x_2 + \theta_4 x_1^2 x_2 + \theta_5 x_1^3 x_2 + \theta_6 x_1 x_2^2 + \dots)$$

$x_1^2, x_1 x_2, x_1 x_3, \dots, x_1 x_{100},$
 $x_2^2, x_1 x_3 \dots$

\Rightarrow **5000 features ($\sim O(n^2)$ parameters)**

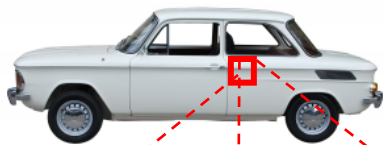
$x_1^2, x_2^2, x_3^2, \dots, x_{10}^2,$
 $x_1 x_2 x_3, x_1^2 x_2, \dots$

\Rightarrow **$O(n^3)$ parameters**

4. PHÂN LOẠI VỚI MẠNG NEURAL

What is this?

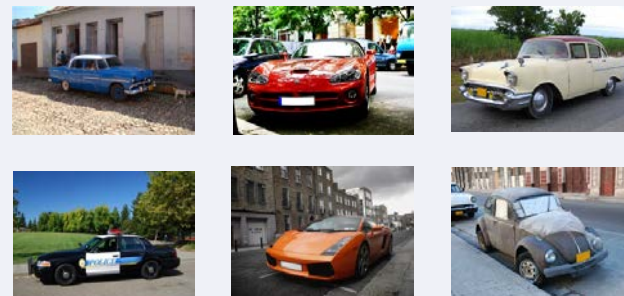
Con người: nhìn thấy hình dạng chiếc xe:



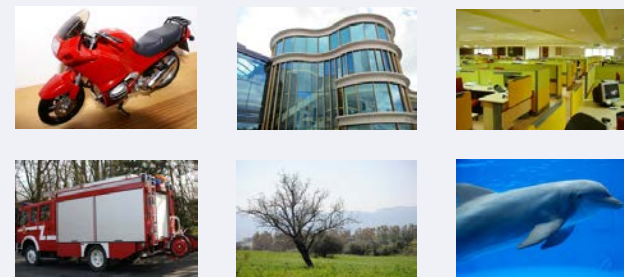
Camera đọc được các pixels:

194	210	201	212	199	213	215	195	178	158	182	209
180	189	190	221	209	205	191	167	147	115	129	163
114	126	140	188	176	165	152	140	170	106	78	88
87	103	115	154	143	142	149	153	173	101	57	57
102	112	106	131	122	138	152	147	128	84	58	66
94	95	79	104	105	124	129	113	107	87	69	67
68	71	69	98	89	92	98	95	89	88	76	67
41	56	68	99	63	45	60	82	58	76	75	65
20	43	69	75	56	41	51	73	55	70	63	44
50	50	57	69	75	75	73	74	53	68	59	37
72	59	53	66	84	92	84	74	57	72	63	42
67	61	58	65	75	78	76	73	59	75	69	50

Training:



Cars

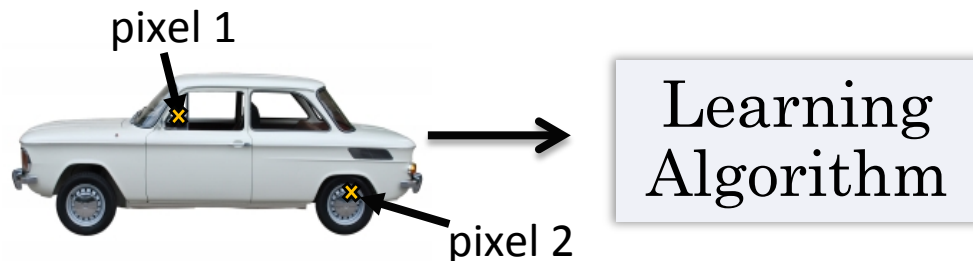


Not a car

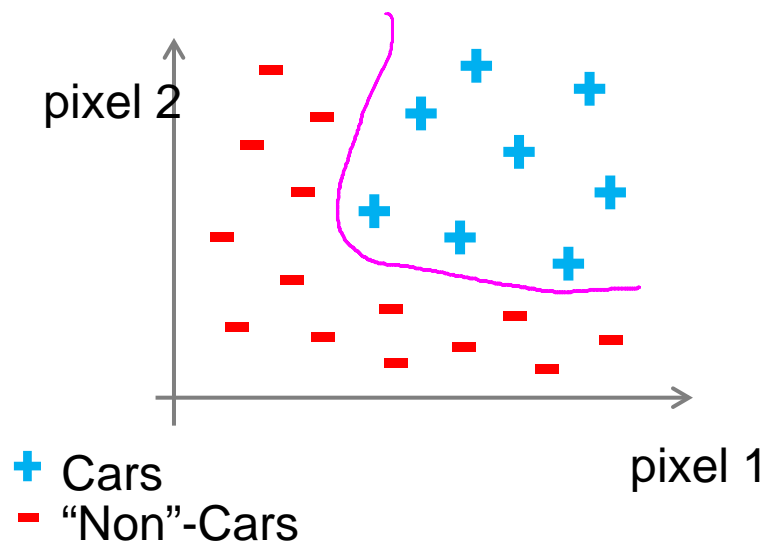


Testing: What is this?

4. PHÂN LOẠI VỚI MẠNG NEURAL



Hình 50 x 50 pixels \rightarrow 2500 pixels
($n=2500$) (7500 if RGB)



$$x = \begin{bmatrix} \text{pixel 1 intensity} \\ \text{pixel 2 intensity} \\ \vdots \\ \text{pixel 2500 intensity} \end{bmatrix}$$

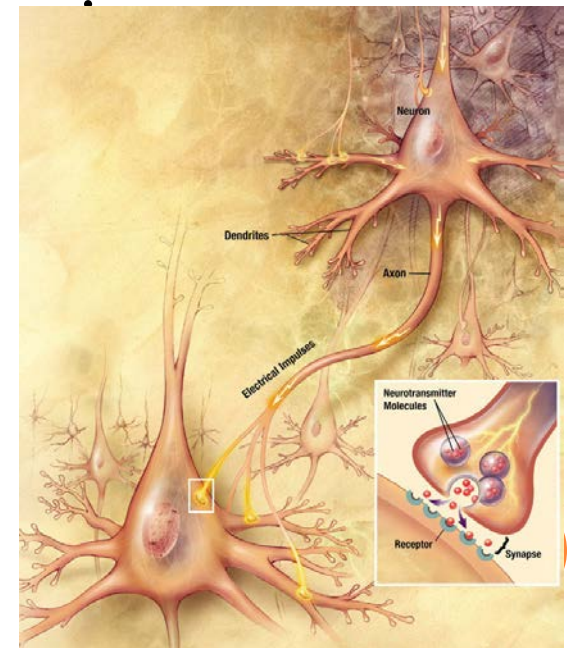
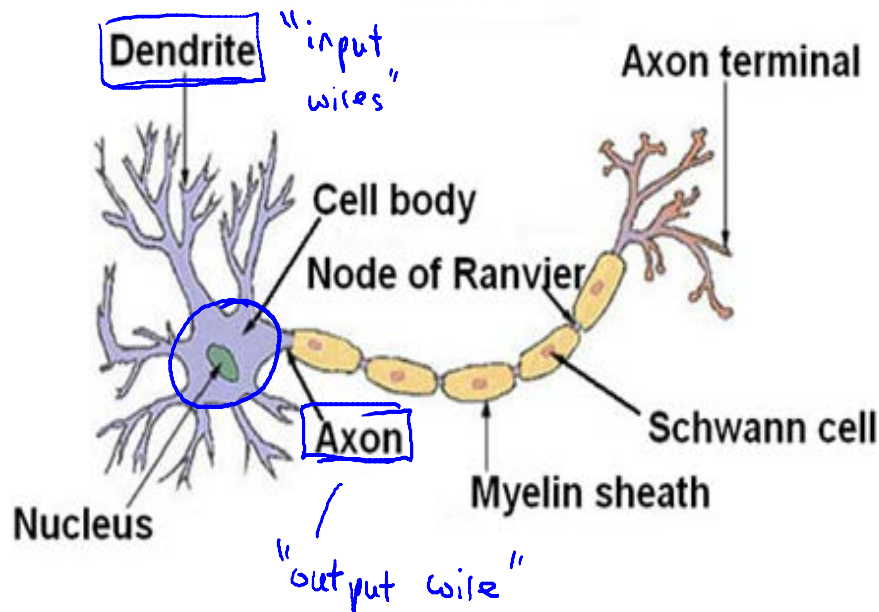
Handwritten note: 0-255

Kết hợp hàm bậc 2 ($x_i * x_j$): $\approx 3M$ features

4. PHÂN LOẠI VỚI MẠNG NEURAL

- Các giải thuật mô phỏng theo sự hoạt động của bộ não người
- Phổ biến từ những năm 80, đầu những năm 90
- Hiện tại được sử dụng ở nhiều lĩnh vực khác nhau

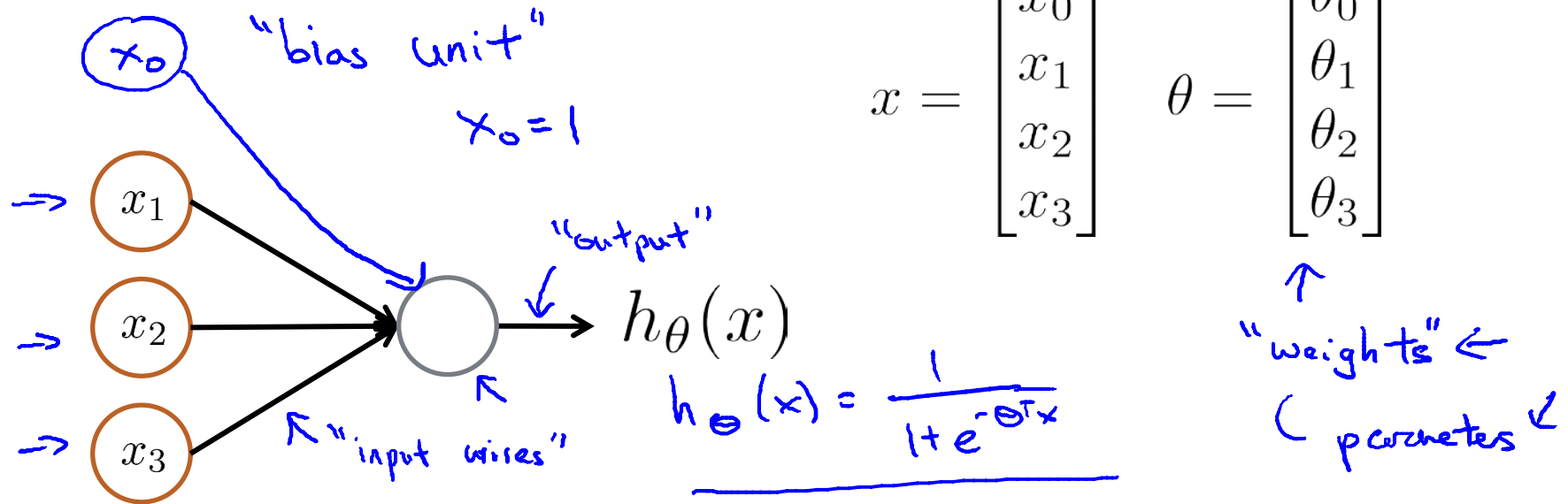
Neuron in the brain



Source: Andrew Ng

4. PHÂN LOẠI VỚI MẠNG NEURAL

Mô hình neuron: Logistic unit

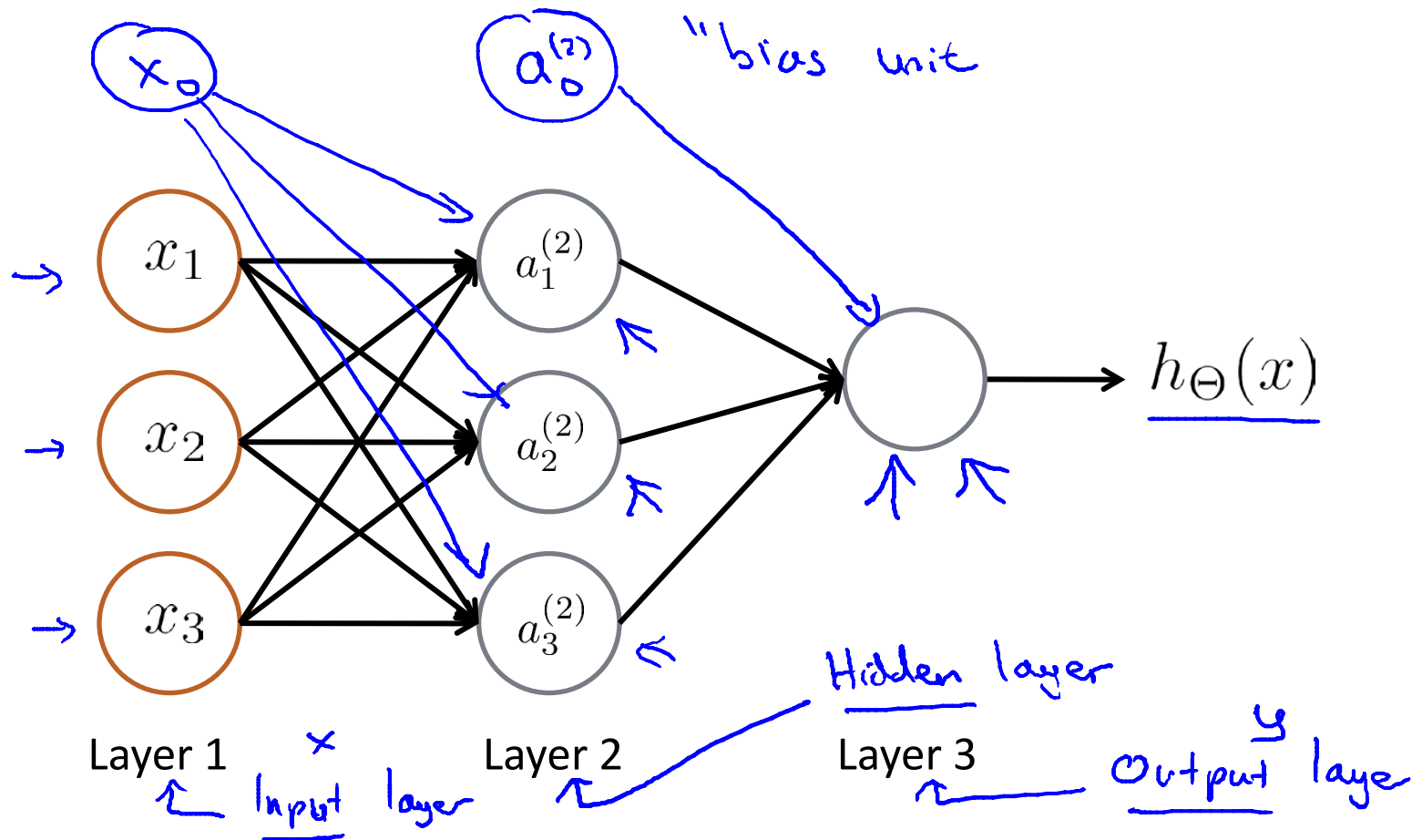


Sigmoid (logistic) activation function.

$$g(z) = \frac{1}{1 + e^{-z}}$$

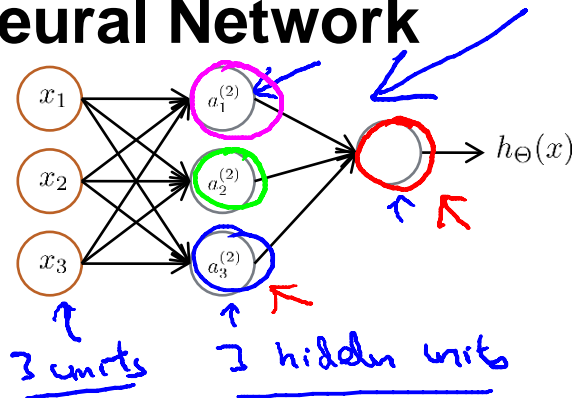
4. PHÂN LOẠI VỚI MẠNG NEURAL

Neural Network



4. PHÂN LOẠI VỚI MẠNG NEURAL

Neural Network



→ $a_i^{(j)}$ = “activation” of unit i in layer j

→ $\Theta^{(j)}$ = matrix of weights controlling function mapping from layer j to layer $j+1$

$$\Theta^{(1)} \in \mathbb{R}^{3 \times 4}$$

$$h_{\Theta}(x)$$

$$a_1^{(2)} = g(\Theta_{10}^{(1)} x_0 + \Theta_{11}^{(1)} x_1 + \Theta_{12}^{(1)} x_2 + \Theta_{13}^{(1)} x_3)$$

$$a_2^{(2)} = g(\Theta_{20}^{(1)} x_0 + \Theta_{21}^{(1)} x_1 + \Theta_{22}^{(1)} x_2 + \Theta_{23}^{(1)} x_3)$$

$$a_3^{(2)} = g(\Theta_{30}^{(1)} x_0 + \Theta_{31}^{(1)} x_1 + \Theta_{32}^{(1)} x_2 + \Theta_{33}^{(1)} x_3)$$

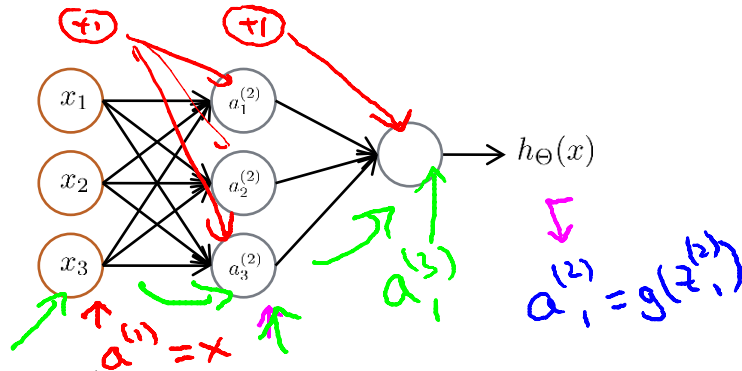
$$h_{\Theta}(x) = a_1^{(3)} = g(\Theta_{10}^{(2)} a_0^{(2)} + \Theta_{11}^{(2)} a_1^{(2)} + \Theta_{12}^{(2)} a_2^{(2)} + \Theta_{13}^{(2)} a_3^{(2)})$$

$\Theta^{(2)}$

Nếu mạng có s_j nodes (units) ở lớp j và s_{j+1} nodes ở lớp $j+1$, thì số chiều của $\Theta^{(j)}$ sẽ là $S_{j+1} \times (S_j + 1)$

4. PHÂN LOẠI VỚI MẠNG NEURAL

ANN: Feed forward (Forward propagation)



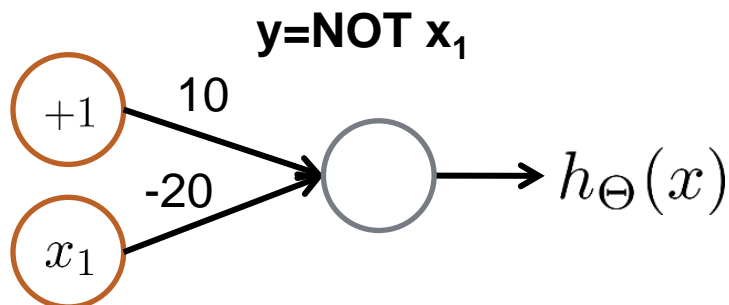
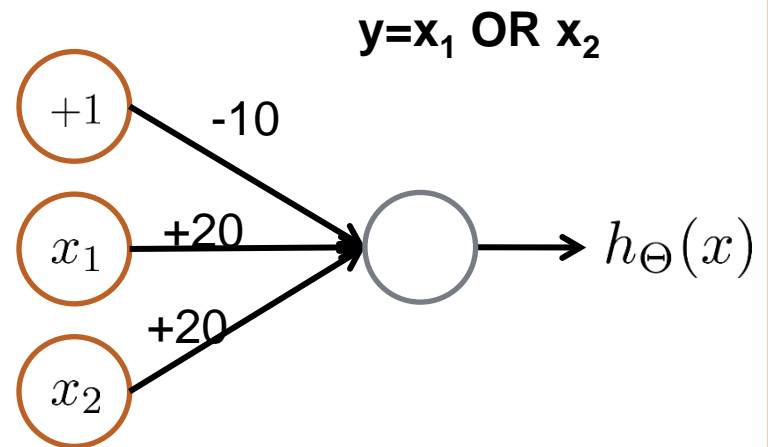
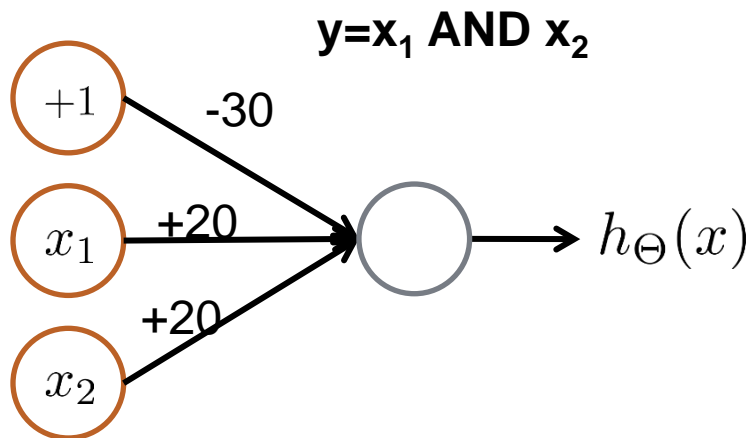
$$x = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad z^{(2)} = \begin{bmatrix} z_1^{(2)} \\ z_2^{(2)} \\ z_3^{(2)} \end{bmatrix}$$

$$\begin{aligned} a_1^{(2)} &= g(\Theta_{10}^{(1)} x_0 + \Theta_{11}^{(1)} x_1 + \Theta_{12}^{(1)} x_2 + \Theta_{13}^{(1)} x_3) & z^{(2)}_1 \\ a_2^{(2)} &= g(\Theta_{20}^{(1)} x_0 + \Theta_{21}^{(1)} x_1 + \Theta_{22}^{(1)} x_2 + \Theta_{23}^{(1)} x_3) & z^{(2)}_2 \\ a_3^{(2)} &= g(\Theta_{30}^{(1)} x_0 + \Theta_{31}^{(1)} x_1 + \Theta_{32}^{(1)} x_2 + \Theta_{33}^{(1)} x_3) & z^{(2)}_3 \\ h_{\Theta}(x) &= a_1^{(3)} = g(\Theta_{10}^{(2)} a_0^{(2)} + \Theta_{11}^{(2)} a_1^{(2)} + \Theta_{12}^{(2)} a_2^{(2)} + \Theta_{13}^{(2)} a_3^{(2)}) \\ & & a^{(3)} = g(z^{(3)}) \end{aligned}$$

$$\begin{aligned} z^{(2)} &= \Theta^{(1)} a^{(1)} \\ a^{(2)} &= g(z^{(2)}) \\ \text{Thêm } a^{(2)}_0 &= 1 \text{ vào } a^{(2)} \\ z^{(3)} &= \Theta^{(2)} a^{(2)} \\ h_{\theta}(x) &= a^{(3)} = g(z^{(3)}) \end{aligned}$$

4. PHÂN LOẠI VỚI MẠNG NEURAL

- Biểu diễn ANN cho các phép toán logic cơ bản



Hãy kiểm chứng bằng bảng logic cho các trường hợp trên!

4. PHÂN LOẠI VỚI MẠNG NEURAL

- Vd dùng ANN để biểu diễn một sự kết hợp phức tạp hơn của các phép toán logic: $x_1 \text{ NOR } x_2$
- $x_1 \text{ NOR } x_2 = \text{NOT } x_1 \text{ XOR } x_2$:

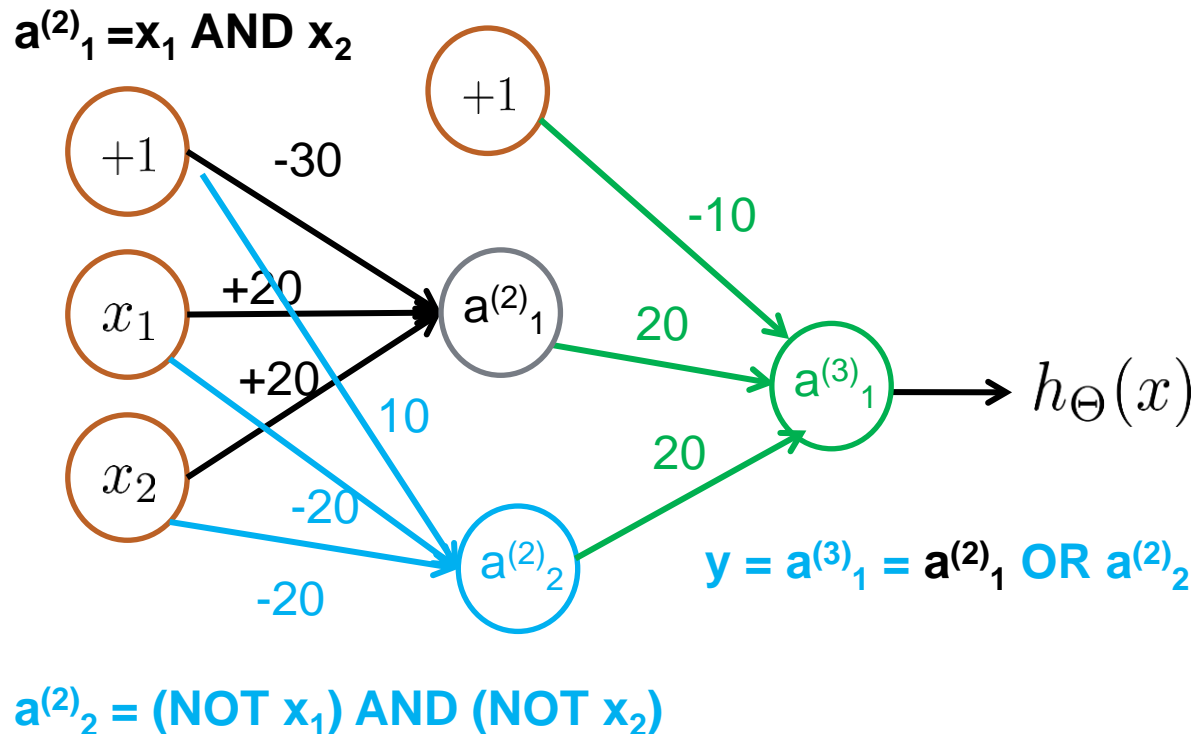
x_1	x_2	$x_1 \text{ XOR } x_2$	$x_1 \text{ NOR } x_2$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1

$\Rightarrow x_1 \text{ NOR } x_2 = (x_1 \text{ AND } x_2) \text{ OR } (\text{NOT } x_1 \text{ AND } \text{NOT } x_2)$

\Rightarrow Kết hợp các biểu diễn ANN cơ bản để biểu diễn biểu thức này

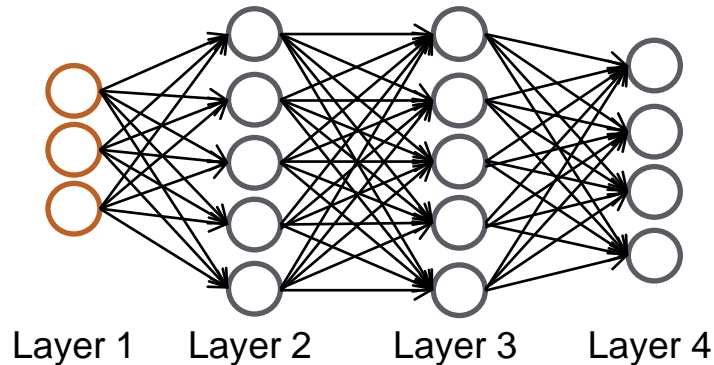
4. PHÂN LOẠI VỚI MẠNG NEURAL

- Biểu diễn $x_1 \text{ NOR } x_2 = (x_1 \text{ AND } x_2) \text{ OR } (\text{NOT } x_1 \text{ AND NOT } x_2)$



4. PHÂN LOẠI VỚI MẠNG NEURAL

○ Hàm chi phí (Cost function) trong ANN



$$\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(m)}, y^{(m)})\}$$

L = Tổng số lớp (layer) trong mạng

s_l = Số nodes (không bao gồm bias node) trong lớp l

Binary classification

$$y = 0 \text{ or } 1$$

1 output unit

Phân loại đa lớp (K classes)

$$y \in \mathbb{R}^K$$

E.g. $\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$, $\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$, $\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$, $\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$
pedestrian car motorcycle truck

K output units

4. PHÂN LOẠI VỚI MẠNG NEURAL

○ Hàm chi phí (Cost function) trong ANN

Logistic regression:

$$J(\theta) = -\frac{1}{m} \left[\sum_{i=1}^m y^{(i)} \log h_{\theta}(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_{\theta}(x^{(i)})) \right] + \frac{\lambda}{2m} \sum_{j=1}^n \theta_j^2$$

Neural network:

$$h_{\Theta}(x) \in \mathbb{R}^K \quad (h_{\Theta}(x))_i = i^{th} \text{ output}$$

$$J(\Theta) = -\frac{1}{m} \left[\sum_{i=1}^m \sum_{k=1}^K y_k^{(i)} \log(h_{\Theta}(x^{(i)}))_k + (1 - y_k^{(i)}) \log(1 - (h_{\Theta}(x^{(i)}))_k) \right] \\ + \frac{\lambda}{2m} \sum_{l=1}^{L-1} \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} (\Theta_{ji}^{(l)})^2$$

4. PHÂN LOẠI VỚI MẠNG NEURAL

- Tối thiểu hóa chi phí (Minimizing cost) trong ANN: Phương pháp lan truyền ngược - Backpropagation

$$J(\Theta) = -\frac{1}{m} \left[\sum_{i=1}^m \sum_{k=1}^K y_k^{(i)} \log h_{\theta}(x^{(i)})_k + (1 - y_k^{(i)}) \log(1 - h_{\theta}(x^{(i)})_k) \right] \\ + \frac{\lambda}{2m} \sum_{l=1}^{L-1} \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} (\Theta_j^{(l)})^2$$

$\min_{\Theta} J(\Theta)$

Cần phải tính:

- $J(\Theta)$
- $\frac{\partial}{\partial \Theta_{ij}^{(l)}} J(\Theta)$

$$\Theta_{ij}^{(l)} \in \mathbb{R}$$

4. PHÂN LOẠI VỚI MẠNG NEURAL

- Tính độ biến thiên của đạo hàm hàm chi phí khi thay đổi thông số (gradient computation)
 - Cho tập huấn luyện (x, y) , feed forward trong ANN

$$a^{(1)} = x$$

$$z^{(2)} = \Theta^{(1)} a^{(1)}$$

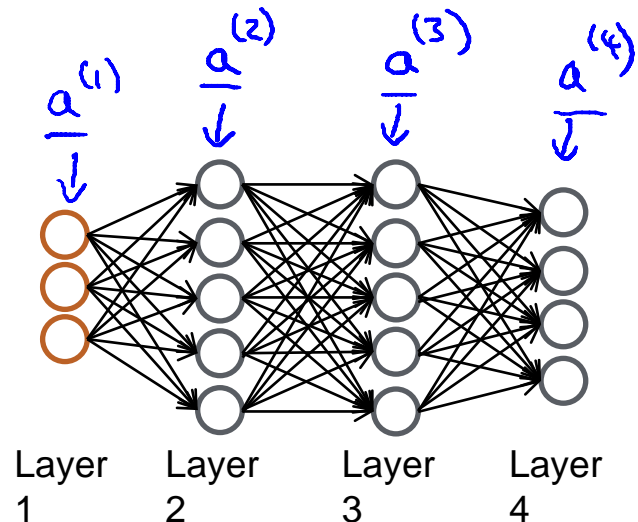
$$a^{(2)} = g(z^{(2)}) \quad (\text{add } a_0^{(2)})$$

$$z^{(3)} = \Theta^{(2)} a^{(2)}$$

$$a^{(3)} = g(z^{(3)}) \quad (\text{add } a_0^{(3)})$$

$$z^{(4)} = \Theta^{(3)} a^{(3)}$$

$$a^{(4)} = h_{\Theta}(x) = g(z^{(4)})$$



4. PHÂN LOẠI VỚI MẠNG NEURAL

○ Gọi $\delta^{(l)}_j$ là “error” do node j ở lớp l gây ra

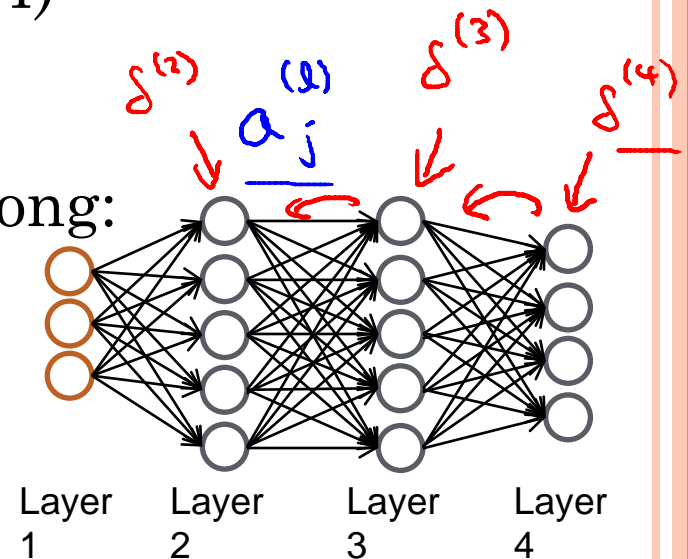
- Với mỗi node ở lớp output ($l=L=4$)

$$\delta^{(4)}_j = a^{(4)}_j - y_j \quad (\delta^{(4)} = a^{(4)} - y)$$

Tính “errors” của các node bên trong:

$$\delta^{(3)} = (\Theta^{(3)})^T \delta^{(4)} \cdot g'(z^{(3)})$$

$$\delta^{(2)} = (\Theta^{(2)})^T \delta^{(3)} \cdot g'(z^{(2)})$$



Chú ý: không tính $\delta^{(1)}$ và

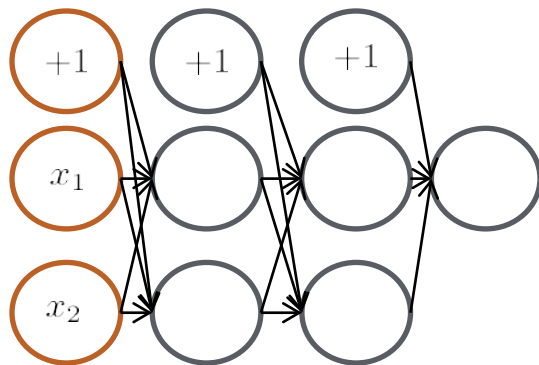
$$\frac{\partial}{\partial \theta_{ij}^{(l)}} J(\theta) = a_j^{(l)} \delta_i^{(l+1)}$$

4. PHÂN LOẠI VỚI MẠNG NEURAL

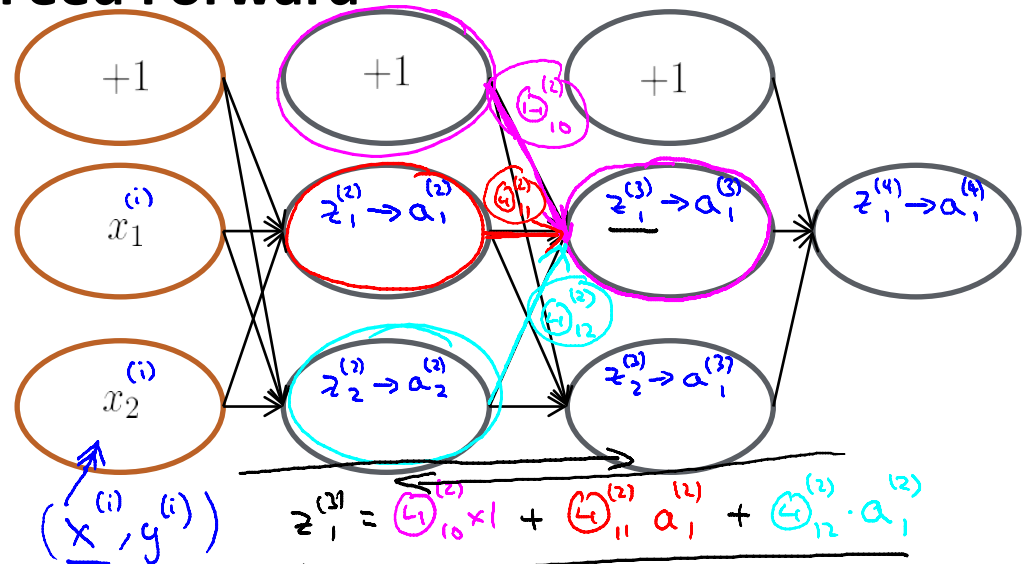
- Giải thuật lan truyền ngược (backpropagation)
 - Cho tập huấn luyện $\{(\mathbf{x}^{(1)}, y^{(1)}), \dots, (\mathbf{x}^{(N)}, y^{(N)})\}$
 - Gán $\Delta_{ij}^{(l)} = 0$ (với mọi i, j, l)
 - For $i=1$ to N ($N = |\mathcal{D}|$)
 - $\mathbf{a}^{(i)} := \mathbf{x}^i$ thực hiện feed forward tính $\mathbf{a}^{(l)}$ ($l=1,2,3,\dots,L$)
 - Dùng $y^{(i)}$ để tính $\delta^{(L)} = \mathbf{a}^{(L)} - y$
 - Tính $\delta^{(L-1)}, \delta^{(L-2)}, \dots, \delta^{(2)}$
 - Tính $\Delta_{ij}^{(l)} := \Delta_{ij}^{(l)} + a_{(j)}^{(l)} \delta_{(i)}^{(l+1)}$
 - Gán
$$\begin{cases} D_{ij}^{(l)} := \frac{1}{N} \Delta_{ij}^{(l)} + \lambda \theta_{ij}^{(l)}, j \neq 0 \\ D_{ij}^{(l)} := \frac{1}{N} \Delta_{ij}^{(l)}, j = 0 \end{cases}$$
$$\frac{\partial}{\partial \theta_{ij}^{(l)}} J(\theta) = D_{ij}^{(l)}$$

4. PHÂN LOẠI VỚI MẠNG NEURAL

- Ví dụ về backpropagation:

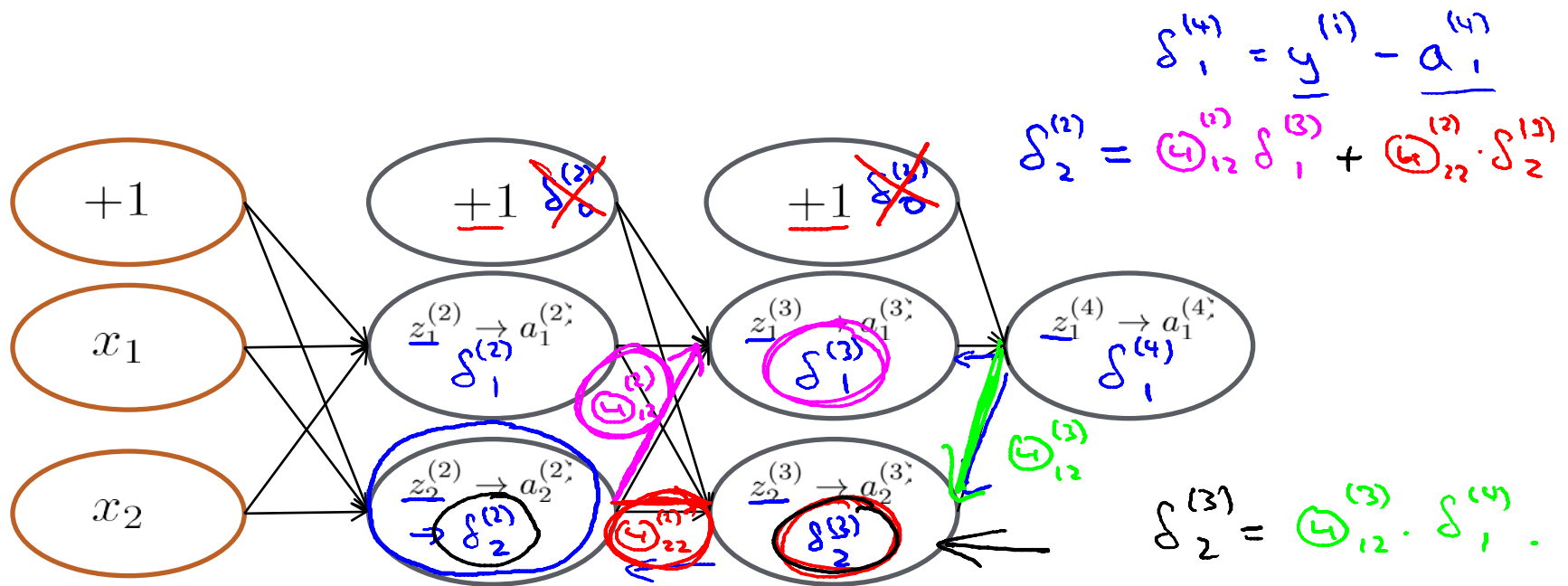


Feed Forward



4. PHÂN LOẠI VỚI MẠNG NEURAL

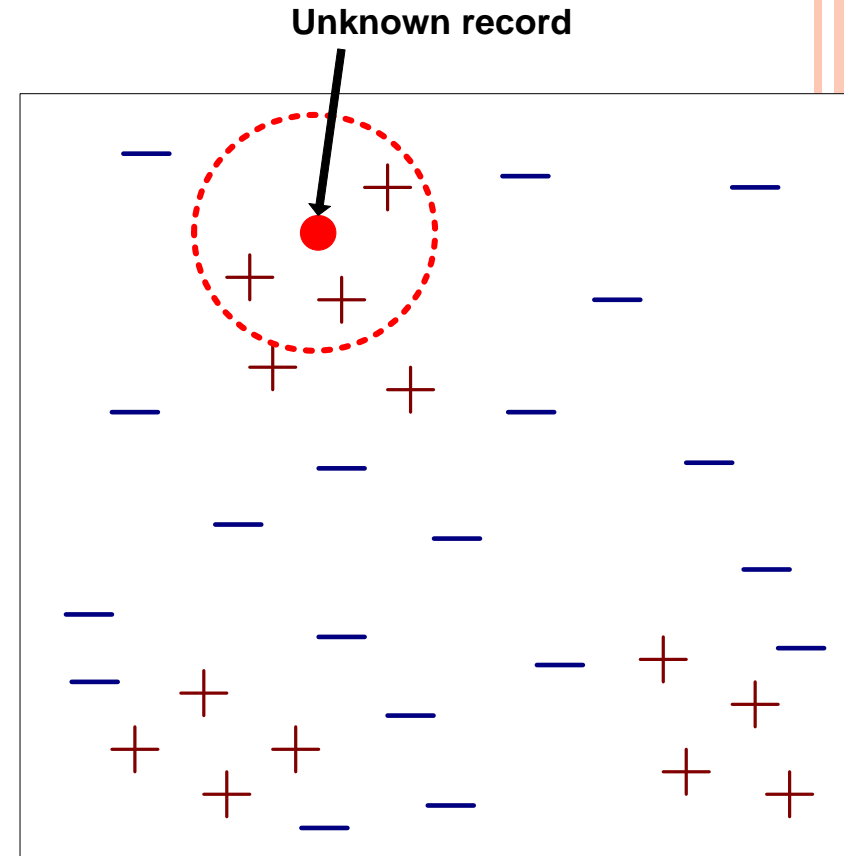
- Ví dụ về backpropagation:



5. CÁC PP PHÂN LOẠI DL KHÁC

○ PP 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
→ $k \leq |D|^{1/2}$



5. CÁC PP PHÂN LOẠI DL KHÁC

○ Chọn độ đo

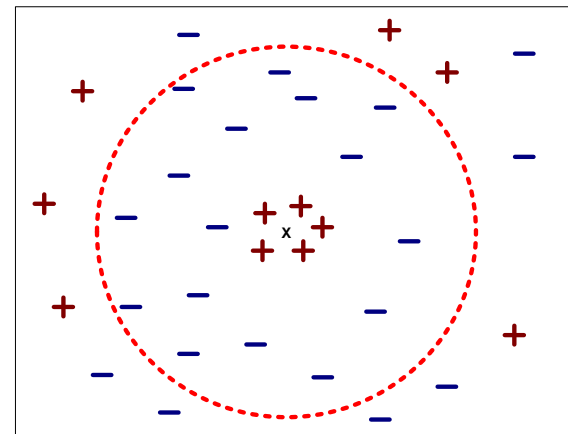
- Độ đo Euclidean

$$d(p, q) = \sqrt{\sum_i (p_i - q_i)^2}$$

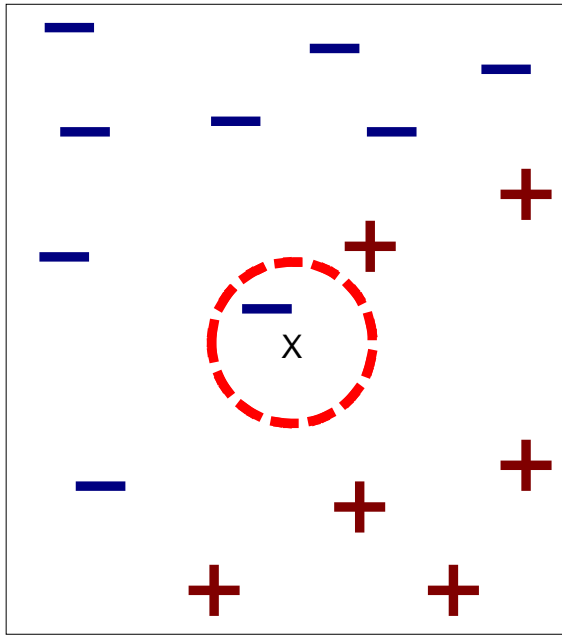
○ Chọn trị k

- 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!

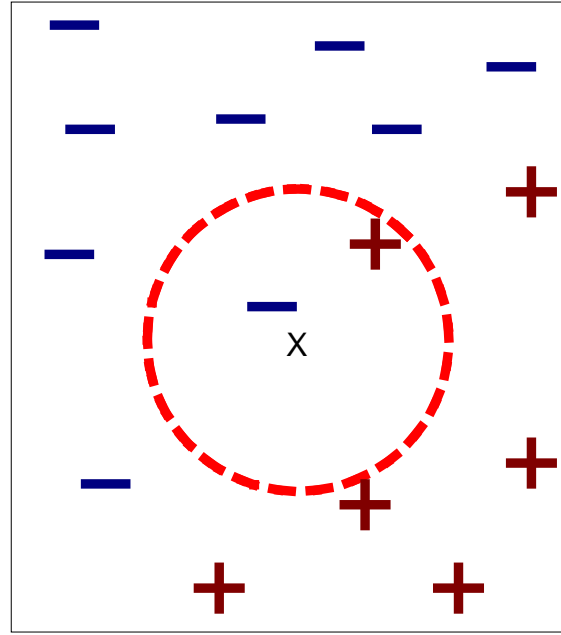


5. CÁC PP PHÂN LOẠI DL KHÁC



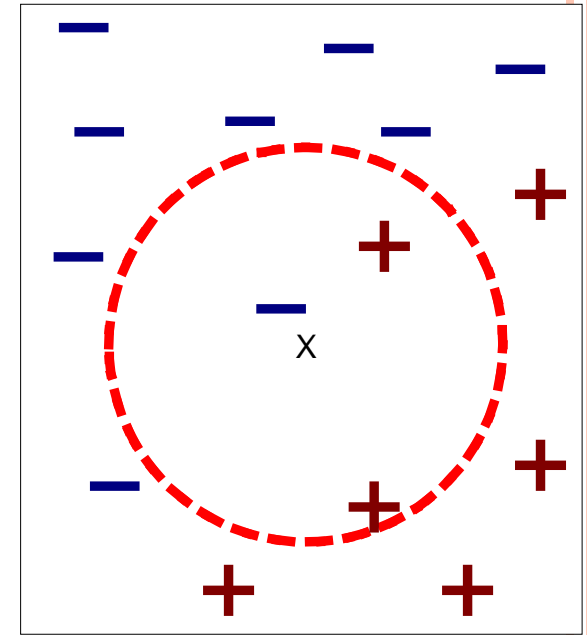
(a) 1-nearest neighbor

$X \in \text{MINUS}$



(b) 2-nearest neighbor

$X \in \text{MINUS}$
hay
 $X \in \text{PLUS} ?$



(c) 3-nearest neighbor

$X \in \text{PLUS}$

6. ĐÁNH GIÁ VÀ CHỌN MÔ HÌNH P.LỚP

○ Tiêu chí đánh giá

- Độ chính xác (accuracy)
 - Thể hiện bộ phân lớp (classifier) có thể nhận biết các đối tượng khác nhau trong tập dữ liệu tốt như thế nào
- Tốc độ (Speed)
 - Chi phí tính toán trong việc xây dựng và sử dụng bộ phân lớp
- Tính mạnh mẽ (Robustness)
 - Khả năng bộ phân lớp có thể làm việc tốt với tập dữ liệu chứa nhiều nhiễu hoặc khuyết nhiều giá trị
- Khả năng mở rộng (Scalability)
 - Khả năng xây dựng được bộ phân lớp với số lượng rất lớn dữ liệu
- Khả năng lý giải được (Interpretability)
 - Khả năng hiểu được nguyên lý hoạt động của bộ phân lớp

6. ĐÁNH GIÁ VÀ CHỌN MÔ HÌNH P.LỚP

- Tiêu chí: High Precision (P) and high Recall (R)

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN} \quad F - Score = \frac{2 * P * R}{P + R}$$

TP: True positive; FP: False positive; FN: False negative

Fact Classified	X	!X
X	TP	FP
!X	FN	TN

Precision points to the FP cell (blue box).

Recall points to the FN cell (red box).

- E.x: Dataset has 9 BG and 4 FG flows (total: 13 flows)

The classifier picks up 7 (4 BG and 3 FG) flows as BG flows. $\Rightarrow P=4/(3+4)=4/7$; $R=4/(4+5)=4/9$

6. ĐÁNH GIÁ VÀ CHỌN MÔ HÌNH PHÂN LỚP

- Đánh giá tính đúng đắn (accuracy) của bộ phân lớp
 - Holdout method: Chia tập dữ liệu đã cho (D) làm 2 tập riêng biệt một cách ngẫu nhiên:
 - Training set: (e.g., $2/3$) để xây dựng mô hình
 - Test set (e.g., $1/3$): để kiểm tra mô hình
 - Cross validation
 - Chia D làm k ($k=10$) phần riêng biệt có kích thước tương đương nhau
 - Ở vòng lặp thứ i , sử dụng D_i làm tập kiểm tra và các phần còn lại là tập huấn luyện
 - Các chỉ số đo đạt tính đúng đắn của bộ phân lớp là sự tổng hợp (trung bình) của các lần thử

7. 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
- Đánh giá và chọn mô hình phân lớp
 - Tiêu chí, độ đo, và phương pháp ước lượng

Q&A

quangtran@hcmut.edu.vn

2015/10/13

77