MẠNG NEURON VÀ ỨNG DỤNG TRONG XỬ LÝ TÍN HIỆU

TS. TRẦN MẠNH CƯỜNG
TS. NGUYỄN THÚY BÌNH
BỘ MÔN KỸ THUẬT ĐIỆN TỬ
Email: thuybinh_ktdt@utc.edu.vn

Perceptron và thuật toán perceptron

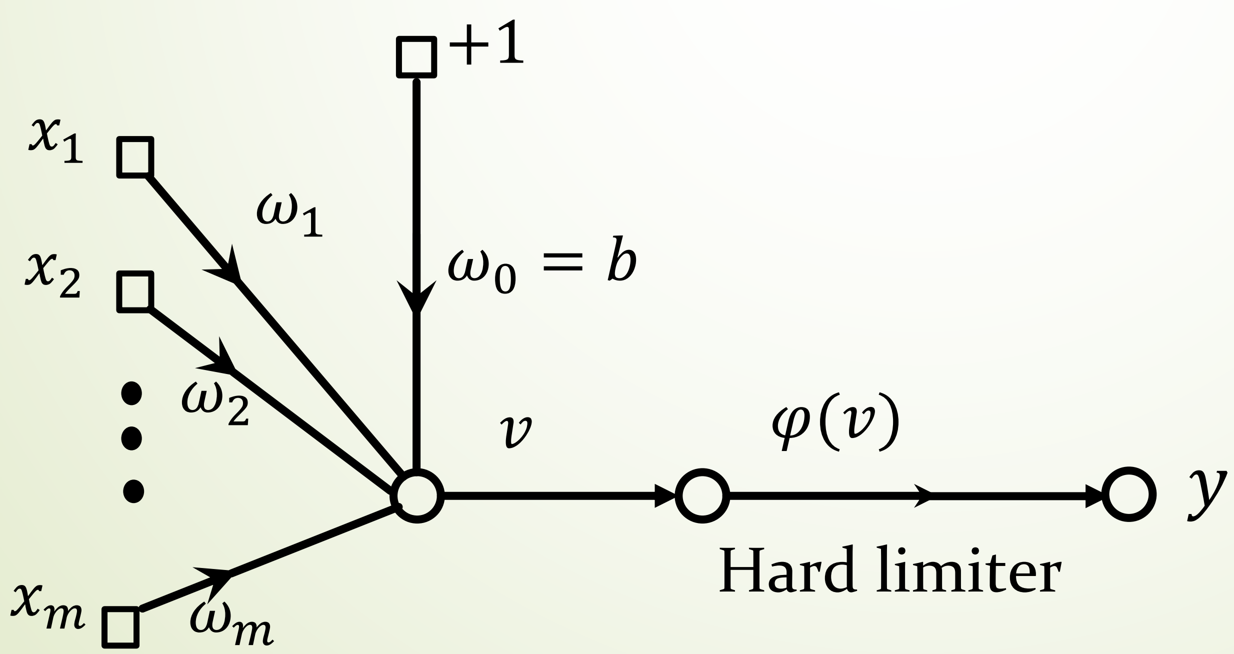
1. Giới thiệu Perceptron
2. Xây dựng lý thuyết Perceptron
3. Giải thuật Perceptron - PLA
4. Ví dụ

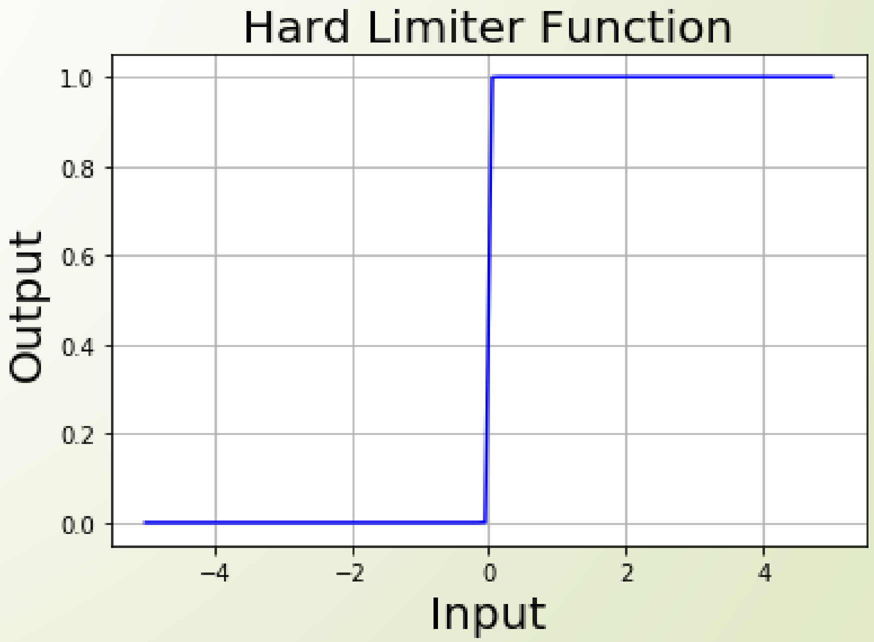
Giới thiệu Perceptron

Đề xuất bởi McCulloch - Pitts (1943)

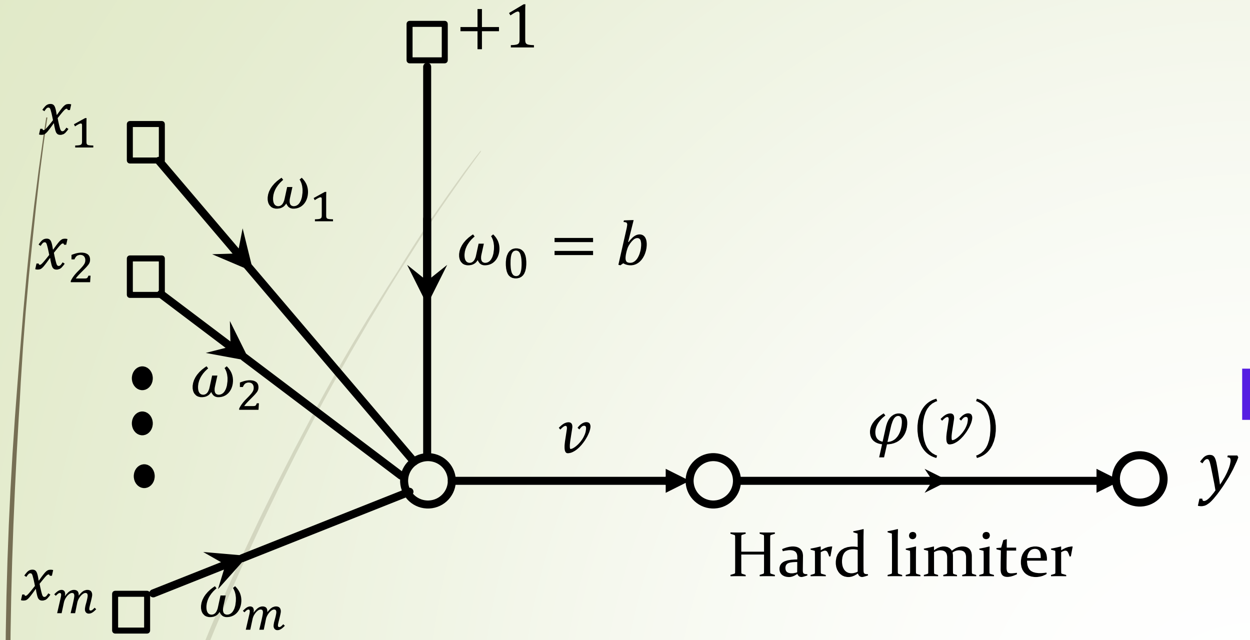
Luật tự học - Luật Hebb (1949)

Rosenblatt đề xuất mạng Perceptron với luật học có giám sát (1958)

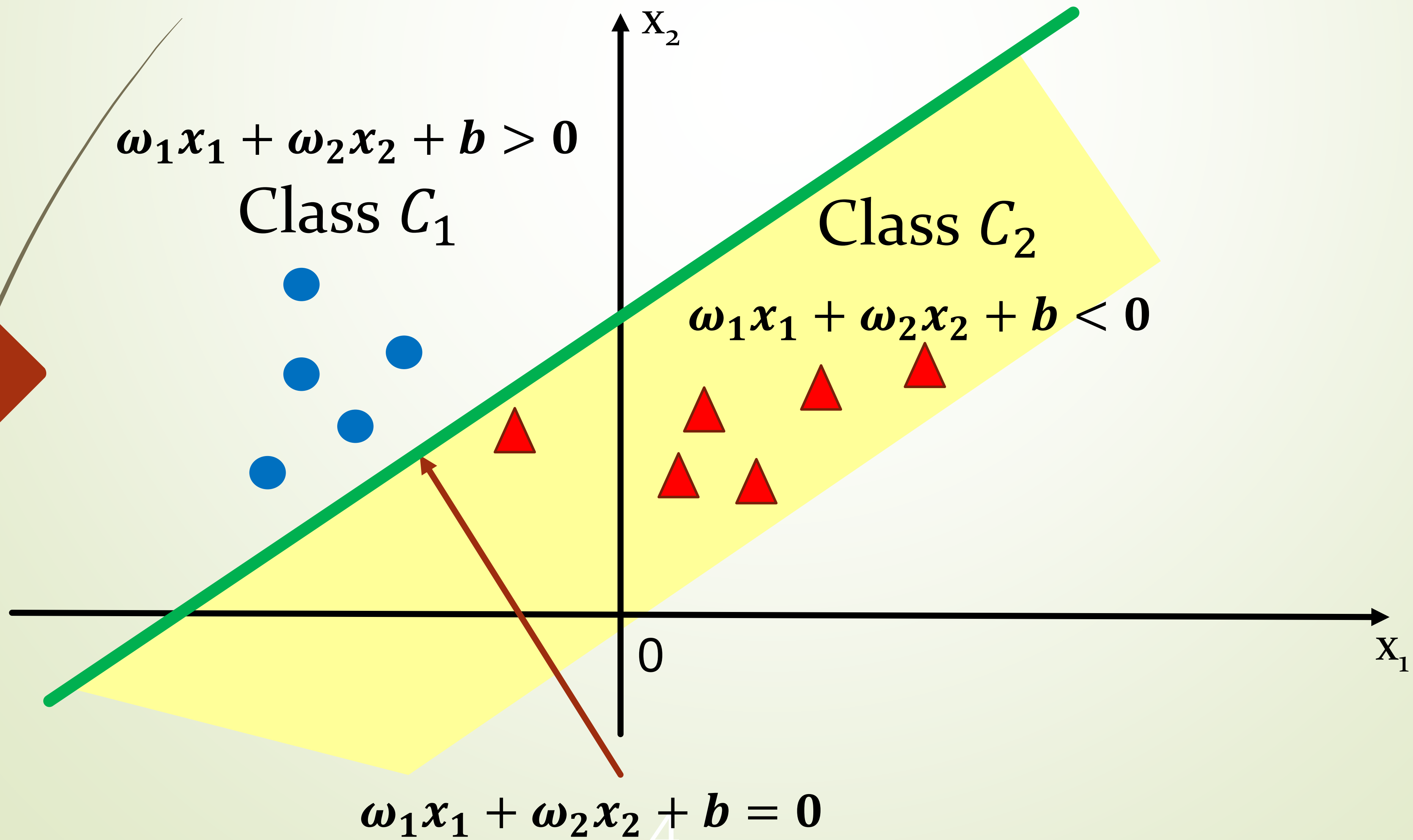




Giới thiệu Perceptron



$$v = \sum_{i=1}^{m} \omega_i x_i + b$$



Xây dựng lý thuyết Perceptron

- Vector đầu vào: $(m+1) \times 1$

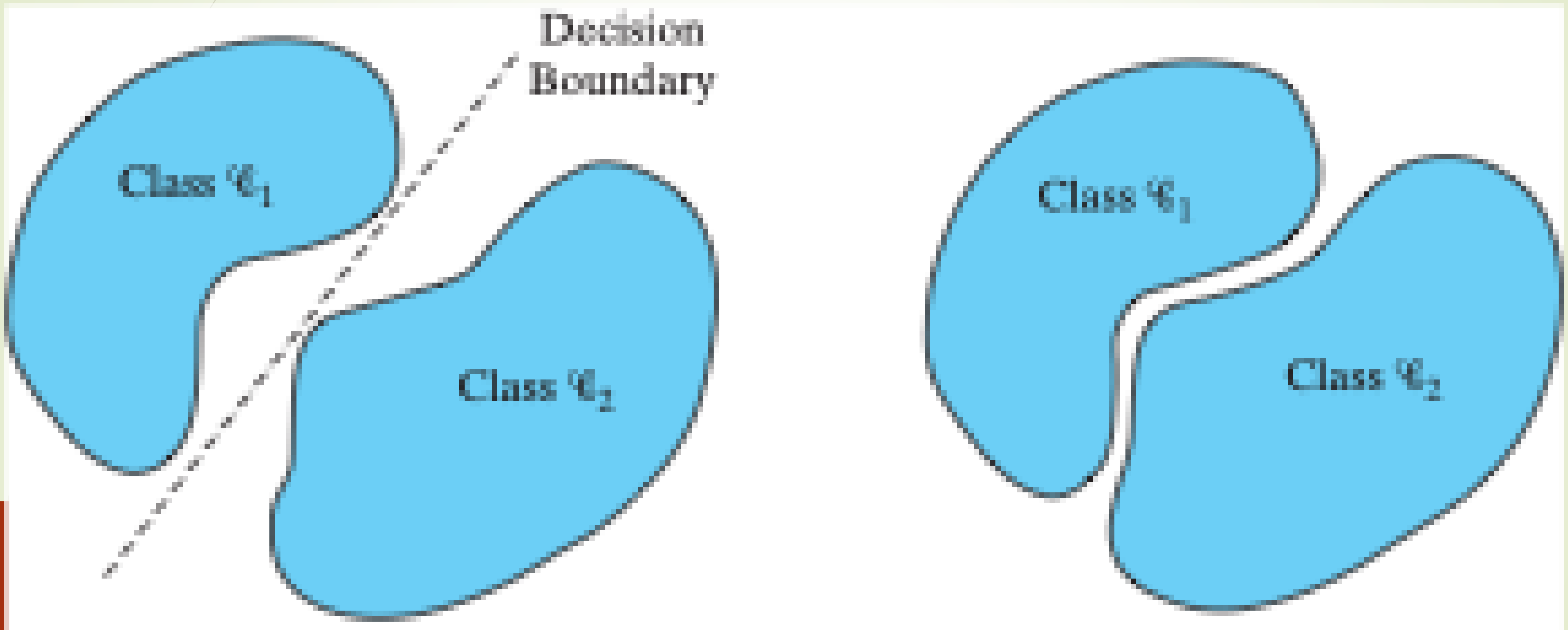
$$\mathbf{x}(\boldsymbol{n}) = [+1, x_1(n), x_2(n), \dots., x_m(n)]^T$$

- Vector trọng số: $(m+1) \times 1$

$$\mathbf{w}(\boldsymbol{n}) = [b, \omega_1(n), \omega_2(n), \dots., \omega_m(n)]^T$$

$$\boxed{v(n) = \sum_{i=0}^{m} \omega_i(n)\, x_i(n) = \mathbf{w}^T(n)\mathbf{x}(n)}$$

Giả thiết về tập đầu ra



Hai tập phân biệt rõ ràng → tồn tại một đường ranh giới

Nằm ngoài khả năng của hàm Hard Limiter

Quá trình học

$\mathcal{H}_1$ tập con của $\mathbb{C}_1$
$\mathcal{H}_2$ tập con của $\mathbb{C}_2$

$\mathbf{w}^T\mathbf{x} > 0$, với mọi $\mathbf{x}$ thuộc $\mathbb{C}_1$
$\mathbf{w}^T\mathbf{x} \le 0$, với mọi $\mathbf{x}$ thuộc $\mathbb{C}_2$

$$\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_2$$

Xét $\mathbf{x}(n)$ thuộc tập mẫu $\mathcal{H}$ và $\mathbf{w}(n)$ là vector trọng số sau n lần thực hiện quá trình học.

$\mathbf{w}(n+1) = \mathbf{w}(n)$ nếu
$$\boxed{\begin{array}{l}\mathbf{w}^T(n)\mathbf{x}(n) > 0 \text{ và } \mathbf{x}(\text{n}) \text{ thuộc } \mathbb{C}_1 \\ \mathbf{w}^T(n)\mathbf{x}(n) \le 0 \text{ và } \mathbf{x}(\text{n}) \text{ thuộc } \mathbb{C}_2\end{array}}$$

Trọng số không cần thay đổi vì ứng với mỗi giá trị đầu vào đầu ra thoả mãn tập đầu ra

Quá trình học

$\mathcal{H}_1$ tập con của $\mathbb{C}_1$
$\mathcal{H}_2$ tập con của $\mathbb{C}_2$

$\Rightarrow$

$\mathbf{w}^T\mathbf{x} > 0$, với mọi $\mathbf{x}$ thuộc $\mathbb{C}_1$
$\mathbf{w}^T\mathbf{x} \leq 0$, với mọi $\mathbf{x}$ thuộc $\mathbb{C}_2$

$\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_2$

Xét $\mathbf{x}(n)$ thuộc tập mẫu $\mathcal{H}$ và $\mathbf{w}(n)$ là vector trọng số sau n lần thực hiện quá trình học.

$\mathbf{w}(n+1) = \mathbf{w}(n) - \eta(n)\mathbf{x}(n)$ nếu $\mathbf{w}^T(n)\mathbf{x}(n) > 0$ và $\mathbf{x}$(n) thuộc $\mathbb{C}_2$

Giảm để kéo $\mathbf{w}^T(n)\mathbf{x}(n) > \mathbf{0}$ về tập $\mathbb{C}_2$ (âm)

$\mathbf{w}(n+1) = \mathbf{w}(n) + \eta(n)\mathbf{x}(n)$ nếu $\mathbf{w}^T(n)\mathbf{x}(n) \leq 0$ và $\mathbf{x}$(n) thuộc $\mathbb{C}_1$

Tăng để kéo $\mathbf{w}^T(n)\mathbf{x}(n) \leq \mathbf{0}$ về tập $\mathbb{C}_1$ (dương)

Trọng số thay đổi để đầu ra thoả mãn tập đầu ra

Tốc độ học η

- η thay đổi được → có tác động thích hợp tới vector $w(n)$
- η nhỏ → việc học diễn ra chậm
- η lớn → việc học diễn ra nhanh nhưng khó hội tụ
- $\eta = 1; w(0) = 0$:

$$w(n+1) = x(1) + x(2) + \cdots + x(n)$$

với các mẫu không thoả mãn tập điều kiện

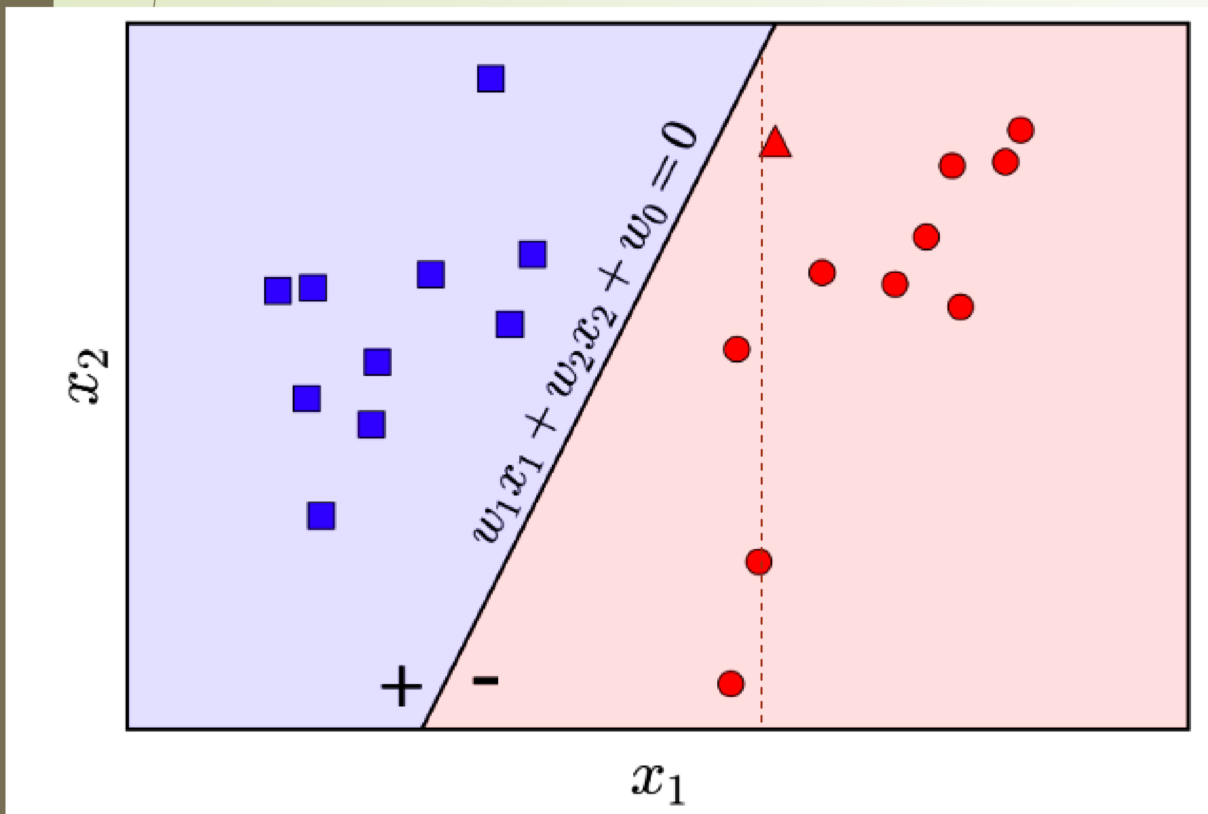
Giải thuật Perceptron

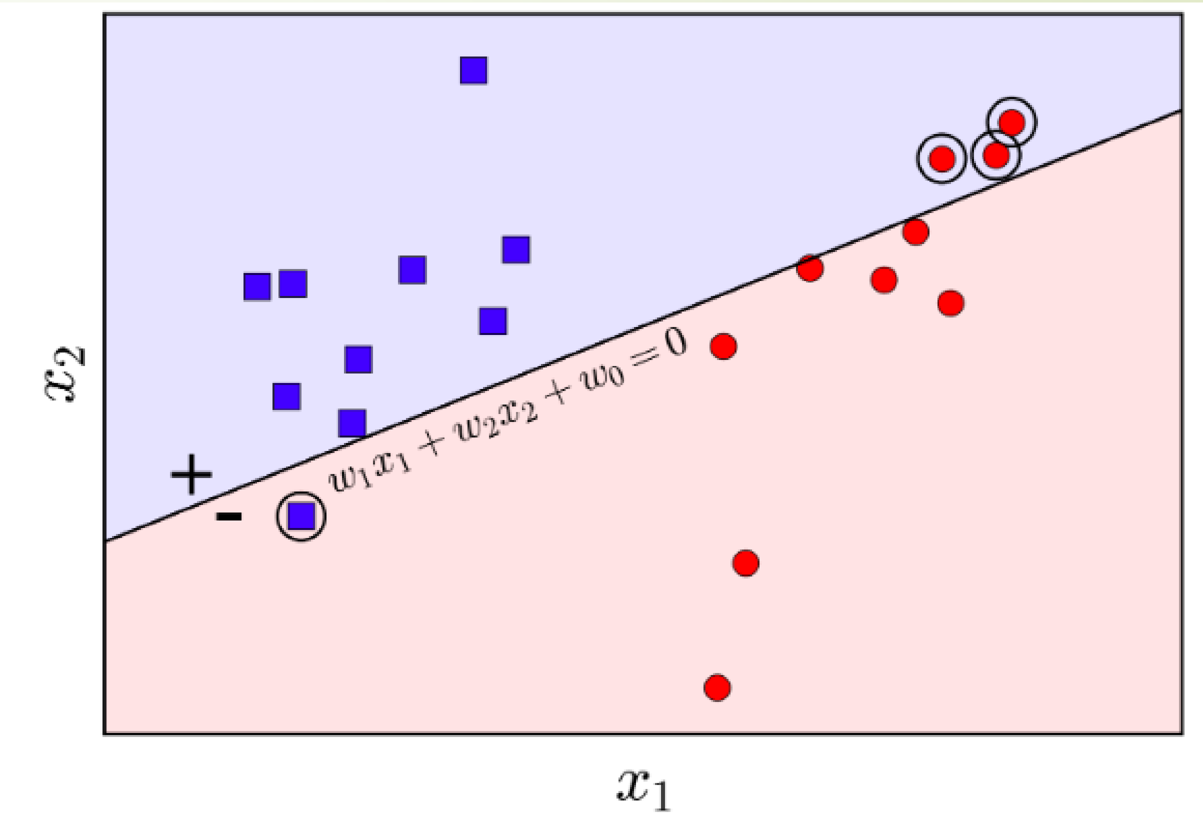
Các biến và tham số

$\mathbf{x}(\boldsymbol{n})$ là vector đầu vào lớp n	$\mathbf{y}(\boldsymbol{n})$ là đáp ứng đầu ra
$\mathbf{w}(\boldsymbol{n})$là vector trọng số	$\mathbf{d}(\boldsymbol{n})$ là đầu ra mong muốn

1. *Khởi tạo:* Đặt w(0) = 0
2. *Kích thích:* Tại bước tính thứ *n*, tính sự kích hoạt của tế bào perceptron với vector vào $\mathbf{x}(\boldsymbol{n})$
3. *Xem xét đầu ra:* Tính đáp ứng ra: $y(n) = \begin{cases} 1 & \boldsymbol{w}^T(n)\boldsymbol{x}(n) > 0 \\ 0 & \boldsymbol{w}^T(n)\boldsymbol{x}(n) \leq 0 \end{cases}$
4. *Cập nhật vector trọng số:* $\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + \eta(n)[\boldsymbol{d}(n) - \boldsymbol{y}(n)]\boldsymbol{x}(n)$
5. *Tính với giá trị n tiếp theo* $d(n) = \begin{cases} 1 \ \textit{nếu}\ x(n) \in \mathbb{C}_1 \\ 0 \ \textit{nếu}\ x(n) \in \mathbb{C}_2 \end{cases}$

Giải thuật Perceptron





Các mẫu được phân chia vào 2 tập hợp

Không tồn tại mẫu không thoả mãn

Có một vài mẫu không thoả mãn

Hàm mất mát

Perceptron cost function

Giải thuật Perceptron

Perceptron cost function

$$J(w) = \sum_{x \in \chi} [-\mathbf{w}^T \mathbf{x}(n)\mathbf{d}(n)]$$

trong đó **x** là một phần tử tập χ là tập mẫu bị phân loại sai và trọng số của nó **w**

Phân loại đúng → χ là tập rỗng → $J(w)$ =0

Phân loại sai → $\begin{cases} \mathrm{w}^T \mathrm{x}(n) > 0, \mathrm{d(n)}=0 \\ \mathrm{w}^T \mathrm{x}(n) \leq 0, \mathrm{d(n)}=1 \end{cases}$

$\boxed{\mathbf{w}(\mathrm{n}+1) \text{ cần làm cho } J(w) = 0}$

Giải thuật Perceptron

Hàm **J** thay đổi theo **w**

$$\nabla J(w) = \sum_{\mathbf{x(n)} \in \chi} (-\mathbf{x}(n)\mathbf{d}(n))$$

$$\boldsymbol{\nabla} = \left[\frac{\partial}{\partial \omega_1}, \frac{\partial}{\partial \omega_2}, \frac{\partial}{\partial \omega_3}, \dots, \frac{\partial}{\partial \omega_m}\right]^T$$

w(n+1) cần làm cho $J(w)$ =0

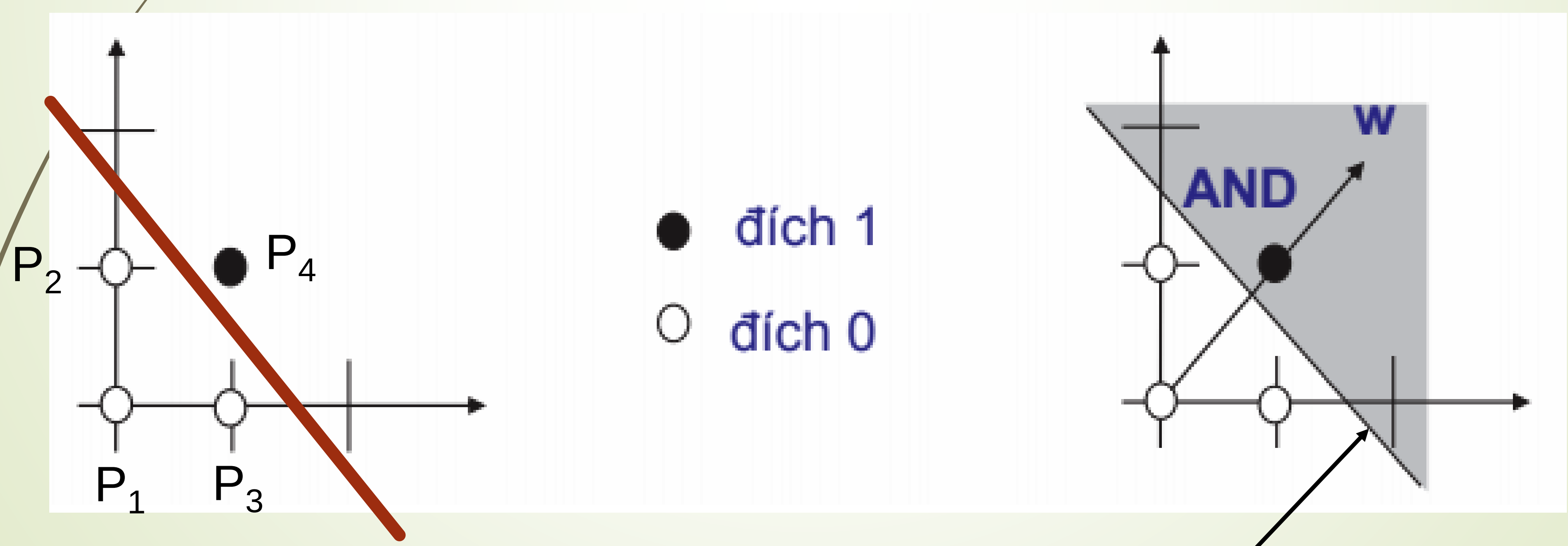
$$\begin{aligned}\mathbf{w}(n+1) &= \boldsymbol{w}(n) - \eta(n)\, \nabla J(w) \\ &= \boldsymbol{w}(n) + \eta(n) \textstyle\sum_{\mathbf{x(n)} \in \chi} (-\mathbf{x}(n)\mathbf{d}(n))\end{aligned}$$

Ví dụ

Yêu cầu: Thiết kế mạng Perceptron thực hiện hàm logic AND

Các mẫu

$$\left(P_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad d_1 = 0 \right) \left(P_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad d_2 = 0 \right) \left(P_3 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad d_3 = 0 \right) \left(P_4 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad d_1 = 1 \right)$$



Mục tiêu là xác định ma trận w để có đường ranh giới

Ví dụ

Yêu cầu: Thiết kế mạng Perceptron thực hiện hàm logic AND

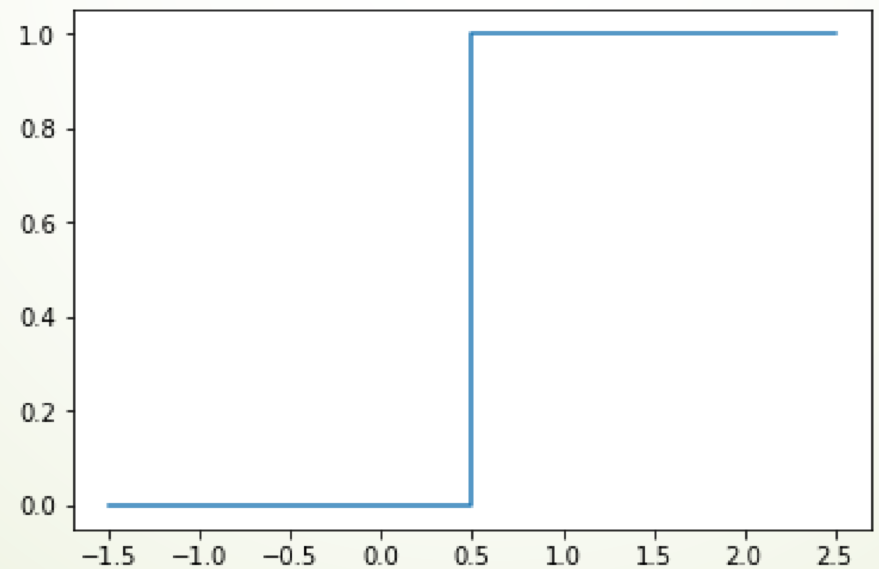
Các mẫu

$$\left(P_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad d_1 = 0 \right) \left(P_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad d_2 = 0 \right) \left(P_3 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad d_3 = 0 \right) \left(P_4 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad d_1 = 1 \right)$$

Giải bằng cách thay đổi hàm kích thích

Chọn hàm kích thích dạng step, b=0, w = [0.5 0.5]

y = 0 nếu *v* ≤0.5



$$v_1 = w_1 p_{11} + w_2 p_{12} + b = 0.5 \times 0 + 0.5 \times 0 + 0 = 0 \Longrightarrow y_1 = 0$$
$$v_2 = w_1 p_{21} + w_2 p_{22} + b = 0.5 \times 0 + 0.5 \times 1 + 0 = 0.5 \Longrightarrow y_2 = 0$$
$$v_3 = w_1 p_{31} + w_2 p_{32} + b = 0.5 \times 1 + 0.5 \times 0 + 0 = 0.5 \Longrightarrow y_3 = 0$$
$$v_4 = w_1 p_{41} + w_2 p_{42} + b = 0.5 \times 1 + 0.5 \times 1 + 0 = 1 \Longrightarrow y_4 = 1$$

Ví dụ

Yêu cầu: Thiết kế mạng Perceptron thực hiện hàm logic AND

Các mẫu

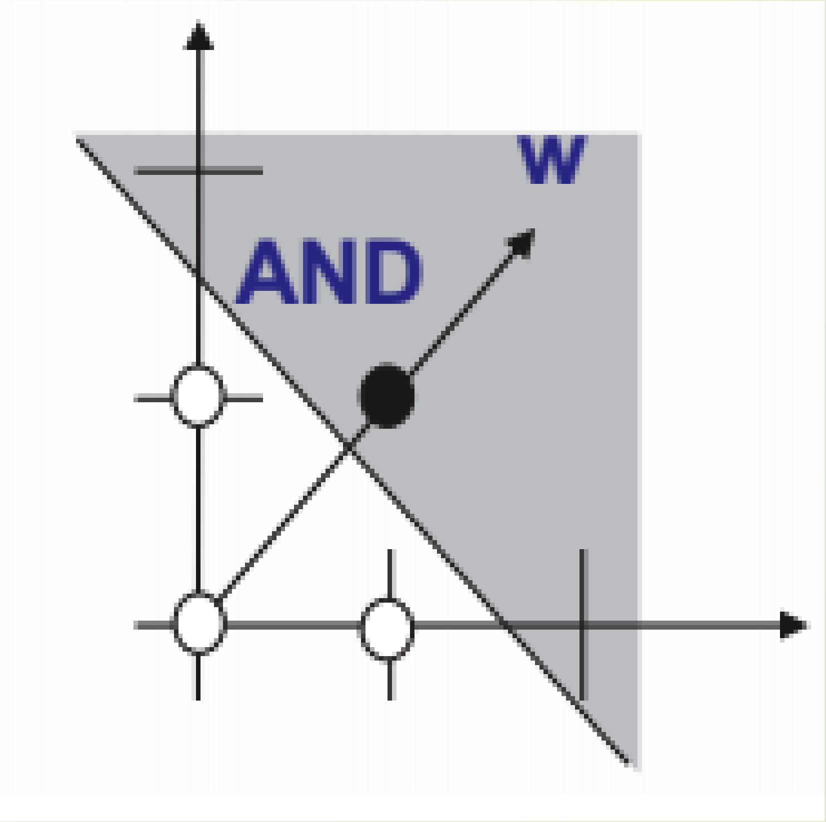
$$\left\{ P_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, d_1 = 0 \right\} \left\{ P_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, d_2 = 0 \right\} \left\{ P_3 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, d_3 = 0 \right\} \left\{ P_4 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, d_4 = 1 \right\}$$

Giải với việc cố định bias b

Chọn đường ranh giới là đường thẳng

Chọn vector trọng số vuông góc đường ranh giới

$$w = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$



$$x_1 + x_2 - 1 = 0$$

Ví dụ

Yêu cầu: Thiết kế mạng Perceptron thực hiện hàm logic AND

Giải với việc cố định bias b

- Lấy 1 điểm trên biên thoả mãn: $w^T P + b = 0$
- Giả sử chọn P_2 có $d_2 = 0$

$$\begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} + b = 0 \Rightarrow b = -1$$

- Huấn luyện mạng

Mẫu P_1 $\begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + (-1) = -1 \Rightarrow y_1 = 0 \Rightarrow$ Thoả mãn

Mẫu P_2 $\begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} + (-1) = 0 \Rightarrow y_2 = 0 \Rightarrow$ Thoả mãn

Mẫu P_3 $\begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + (-1) = 0 \Rightarrow y_3 = 0 \Rightarrow$ Thoả mãn

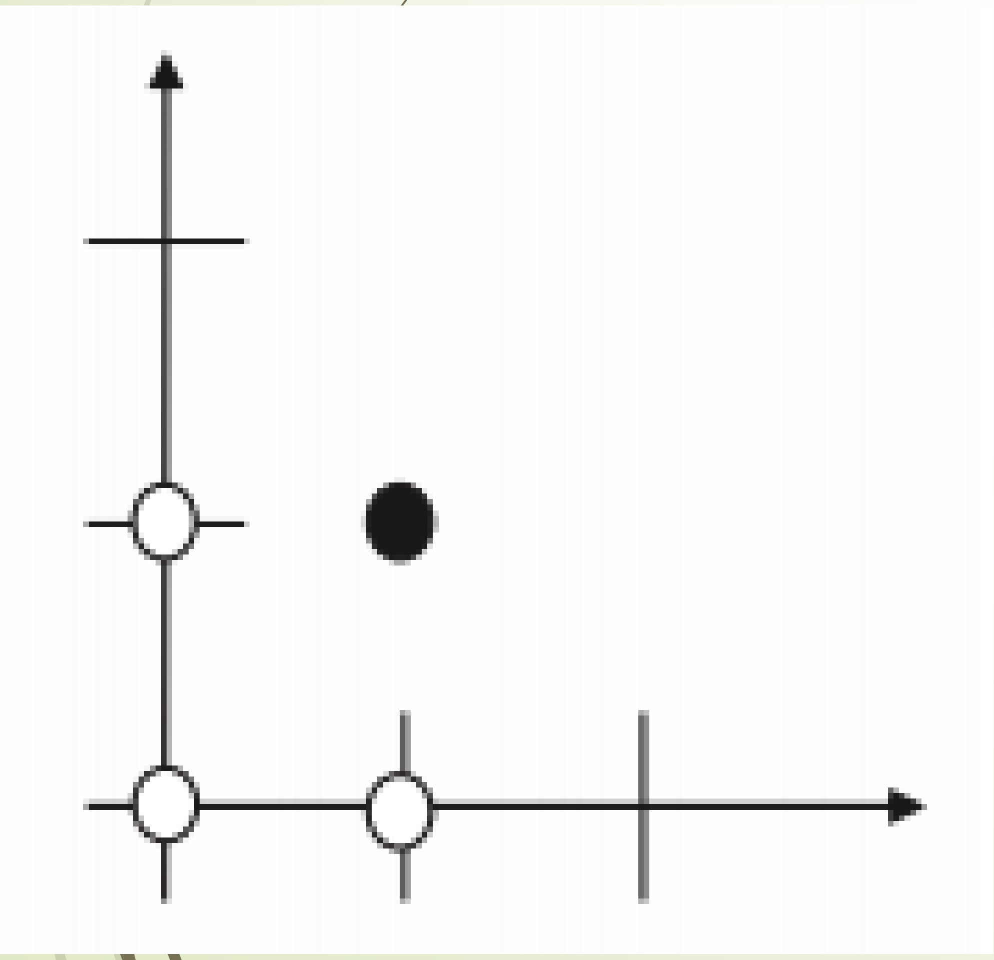
Mẫu P_4 $\begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} + (-1) = 1 \Rightarrow y_4 = 1 \Rightarrow$ Thoả mãn

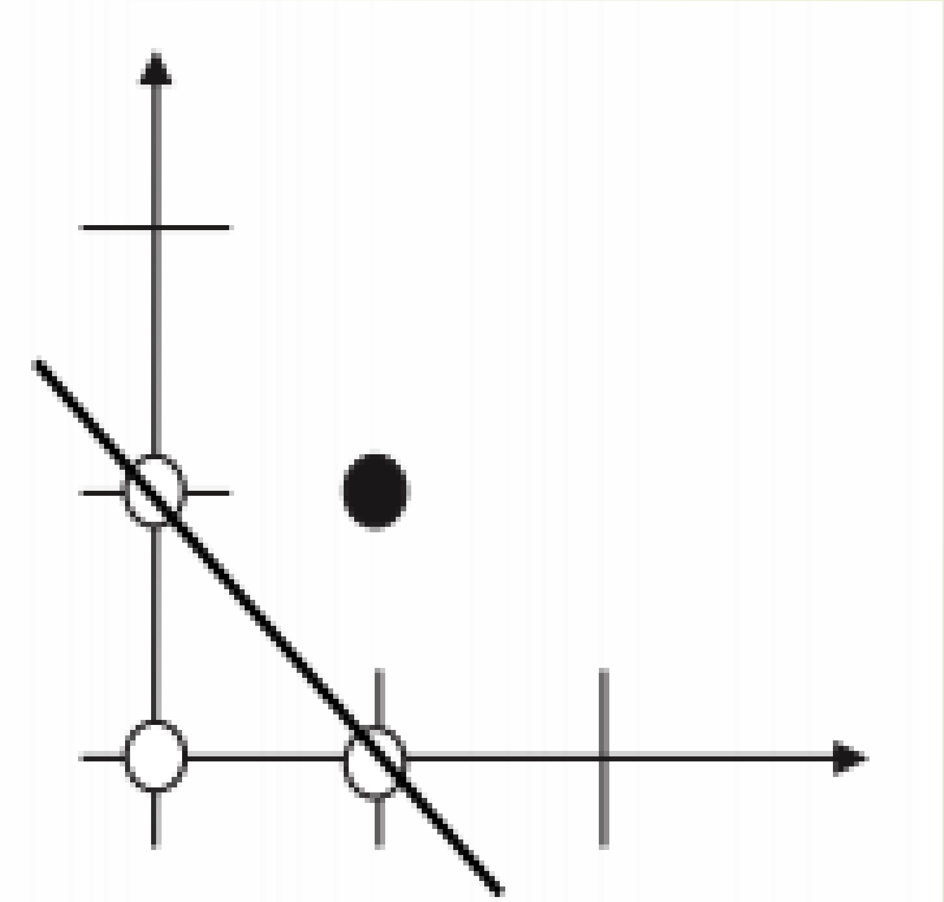
Ví dụ

Yêu cầu: Thiết kế mạng Perceptron thực hiện hàm logic AND

Giải với việc cố định bias b

$$\mathbf{w} = \begin{bmatrix} \mathbf{1} \\ \mathbf{1} \end{bmatrix} \text{ và } \boldsymbol{b} = -\mathbf{1}$$



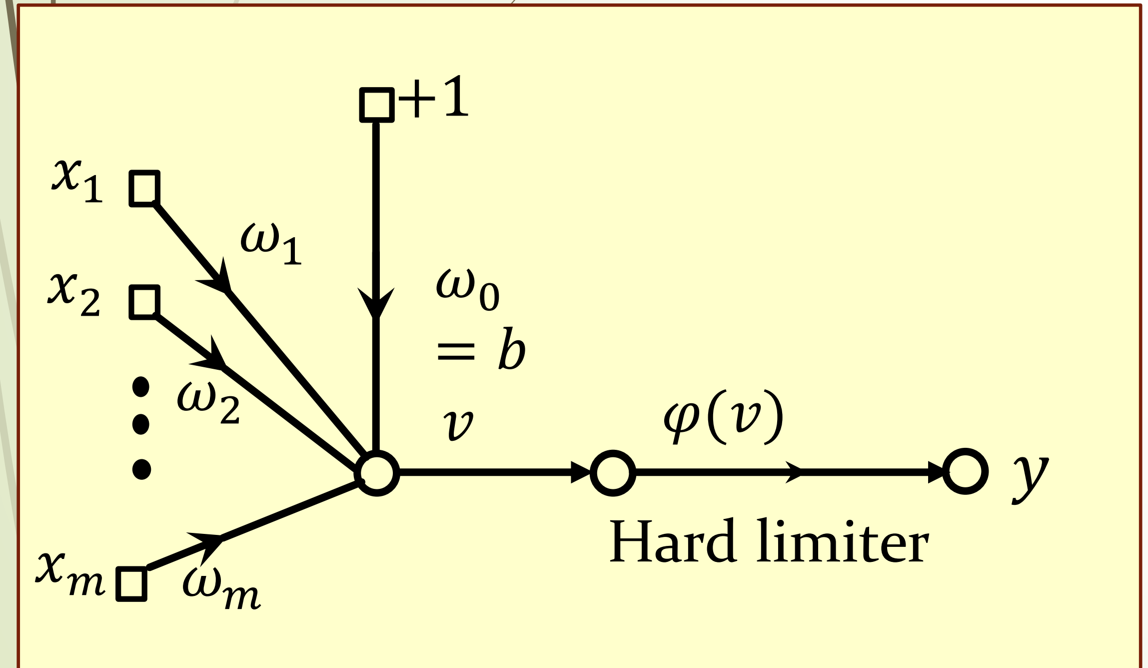


Ví dụ

Yêu cầu: Thiết kế mạng Perceptron thực hiện hàm logic AND

Giải với bias b cũng là một tham số học

$$\left(P_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad d_1 = 0 \right) \left(P_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad d_2 = 0 \right) \left(P_3 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad d_3 = 0 \right) \left(P_4 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad d_1 = 1 \right)$$



Bias b cũng được coi là một tham số trong quá trình học

$$\mathbf{w} = \begin{bmatrix} \boldsymbol{b} \\ \boldsymbol{w_1} \\ \boldsymbol{w_2} \end{bmatrix}$$

$$\left(x_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad d_1 = 0 \right) \left(x_2 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \quad d_2 = 0 \right) \left(x_3 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \quad d_3 = 0 \right) \left(x_4 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad d_4 = 1 \right)$$

Ví dụ

Yêu cầu: Thiết kế mạng Perceptron thực hiện hàm logic AND

Khởi đầu chọn $\mathbf{w}(0) = \begin{bmatrix} b \\ w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$

Học mẫu 1 $\boldsymbol{w}^T \boldsymbol{x}(1) = [0 \quad 0 \quad 0] \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = 0 \Rightarrow y_1 = 0$ **Thoả mãn**

Học mẫu 2 $\boldsymbol{w}^T \boldsymbol{x}(2) = [0 \quad 0 \quad 0] \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} = 0 \Rightarrow y_2 = 0$ **Thoả mãn**

Học mẫu 3 $\boldsymbol{w}^T \boldsymbol{x}(3) = [0 \quad 0 \quad 0] \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = 0 \Rightarrow y_3 = 0$ **Thoả mãn**

Học mẫu 4 $\boldsymbol{w}^T \boldsymbol{x}(4) = [0 \quad 0 \quad 0] \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = 0 \Rightarrow y_4 = 0$ **Không thoả mãn**

Ví dụ

Cập nhật vector trọng số: **w**

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + \eta(n)[\boldsymbol{d}(n) - \boldsymbol{y}(n)]\boldsymbol{x}(n)$$

Chọn tốc độ học (learning rate): $\eta = 1$

$$\mathbf{w(1)} = \begin{bmatrix}\mathbf{0}\\\mathbf{0}\\\mathbf{0}\end{bmatrix} + \mathbf{1} \times (\mathbf{1} - \mathbf{0}) \times \begin{bmatrix}\mathbf{1}\\\mathbf{1}\\\mathbf{1}\end{bmatrix} = \begin{bmatrix}\mathbf{1}\\\mathbf{1}\\\mathbf{1}\end{bmatrix}$$

Học mẫu : $\boldsymbol{w}^T\boldsymbol{x}(\mathbf{1}) = [\mathbf{1} \quad \mathbf{1} \quad \mathbf{1}]\begin{bmatrix}\mathbf{1}\\\mathbf{0}\\\mathbf{0}\end{bmatrix} = \mathbf{1} \Rightarrow \boldsymbol{y}_\mathbf{1} = \mathbf{1}$ **Không thoả mãn**

Cập nhật vector trọng số: **w**

$$\mathbf{w(2)} = \begin{bmatrix}\mathbf{1}\\\mathbf{1}\\\mathbf{1}\end{bmatrix} + \mathbf{1} \times (\mathbf{0} - \mathbf{1}) \times \begin{bmatrix}\mathbf{1}\\\mathbf{0}\\\mathbf{0}\end{bmatrix} = \begin{bmatrix}\mathbf{0}\\\mathbf{1}\\\mathbf{1}\end{bmatrix}$$

Học mẫu 1 $\boldsymbol{w}^T\boldsymbol{x}(\mathbf{1}) = [\mathbf{0} \quad \mathbf{1} \quad \mathbf{1}]\begin{bmatrix}\mathbf{1}\\\mathbf{0}\\\mathbf{0}\end{bmatrix} = \mathbf{0} \Rightarrow \boldsymbol{y}_\mathbf{1} = \mathbf{0}$ **Thoả mãn**

Ví dụ

Cập nhật vector trọng số: w

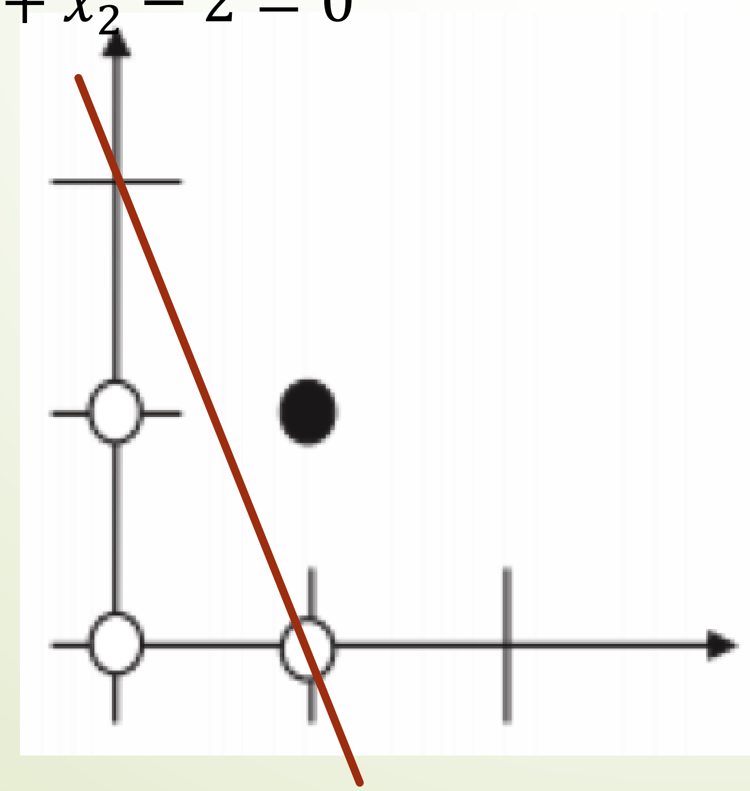
$$\mathbf{w(3)} = \begin{bmatrix} \mathbf{0} \\ \mathbf{1} \\ \mathbf{1} \end{bmatrix} + \mathbf{1} \times (\mathbf{0} - \mathbf{1}) \times \begin{bmatrix} \mathbf{1} \\ \mathbf{0} \\ \mathbf{1} \end{bmatrix} = \begin{bmatrix} \mathbf{-1} \\ \mathbf{1} \\ \mathbf{0} \end{bmatrix}$$

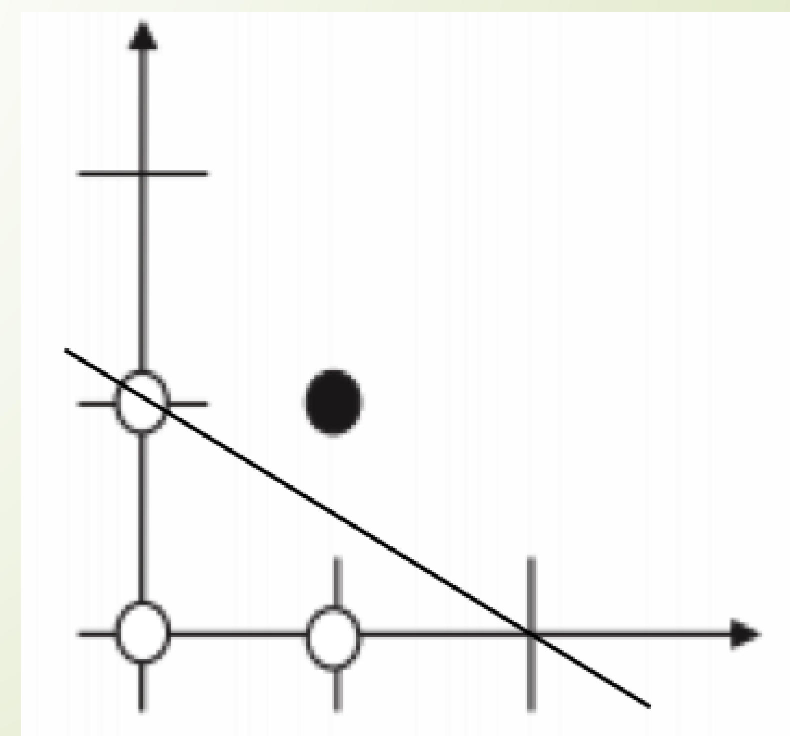
Sau mỗi bước tính *w(n)* ta tiến hành cho mạng học lại các mẫu

Tới bước tính n=10 thì tất cả các mẫu đều thoả mãn

$2x_1 + x_2 - 2 = 0$

$$\mathbf{w(10)} = \begin{bmatrix} \mathbf{-2} \\ \mathbf{2} \\ \mathbf{1} \end{bmatrix}$$

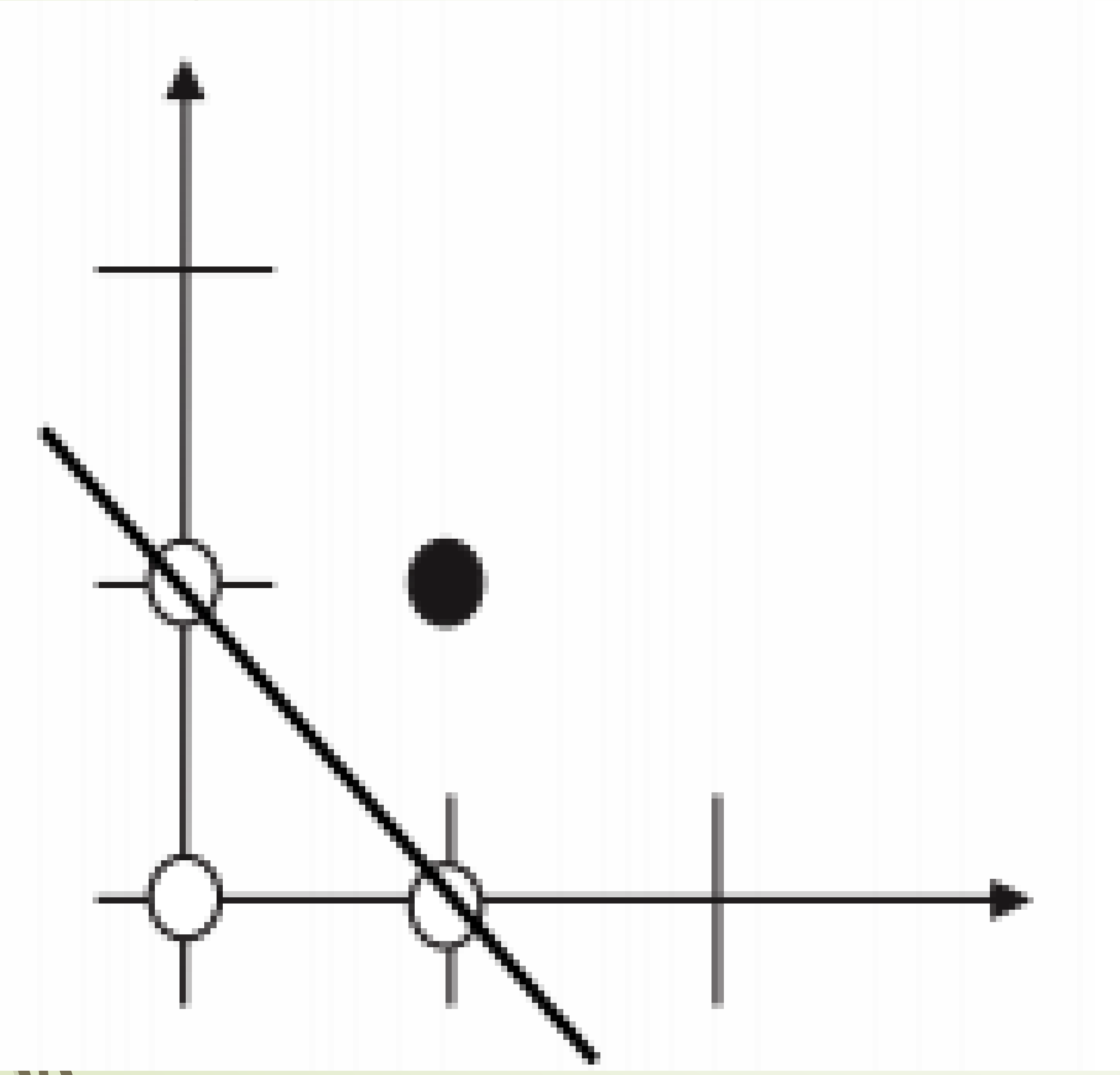


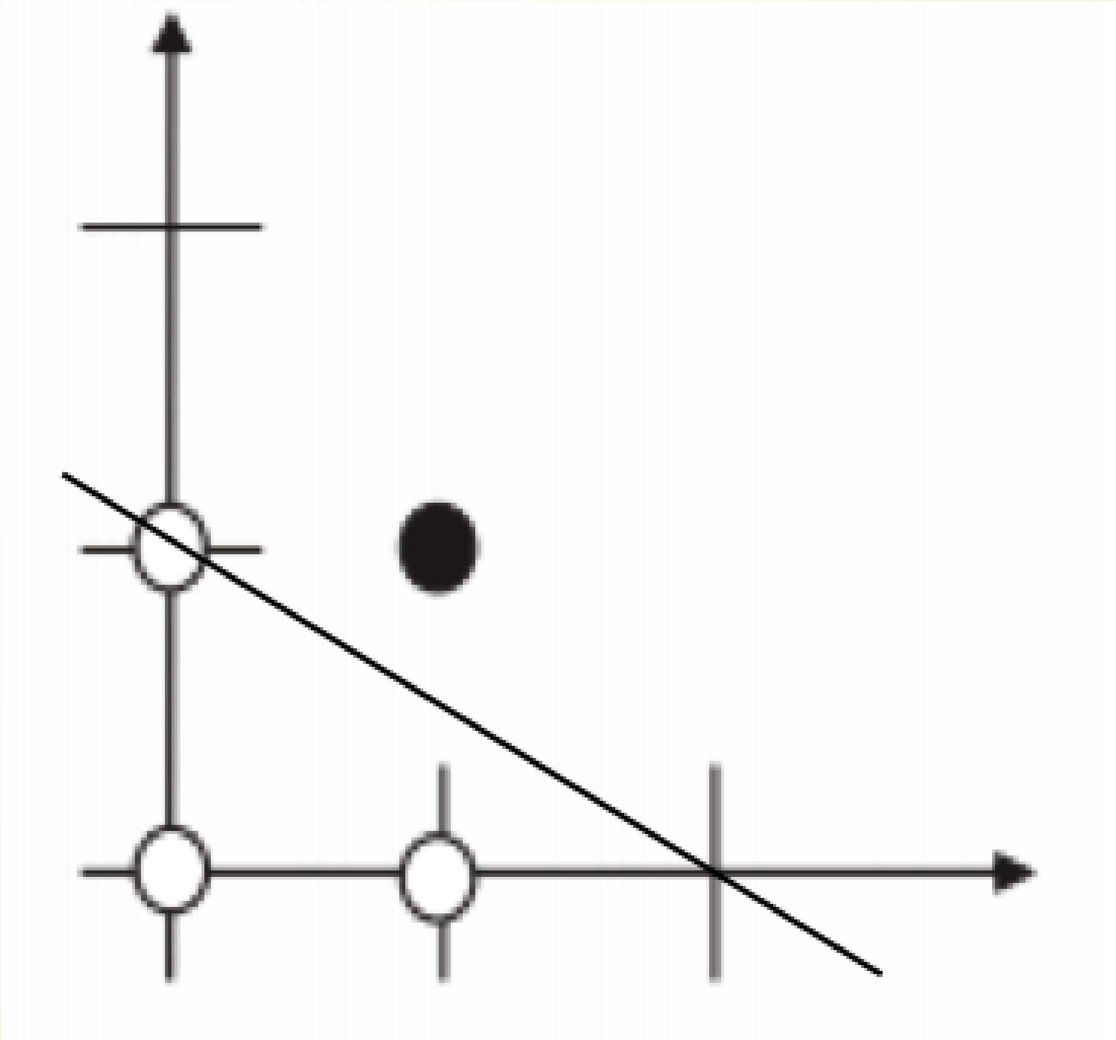


Ví dụ

$\mathbf{w} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ và $b = -1$

$\mathbf{w}(10) = \begin{bmatrix} -2 \\ 2 \\ 1 \end{bmatrix}$

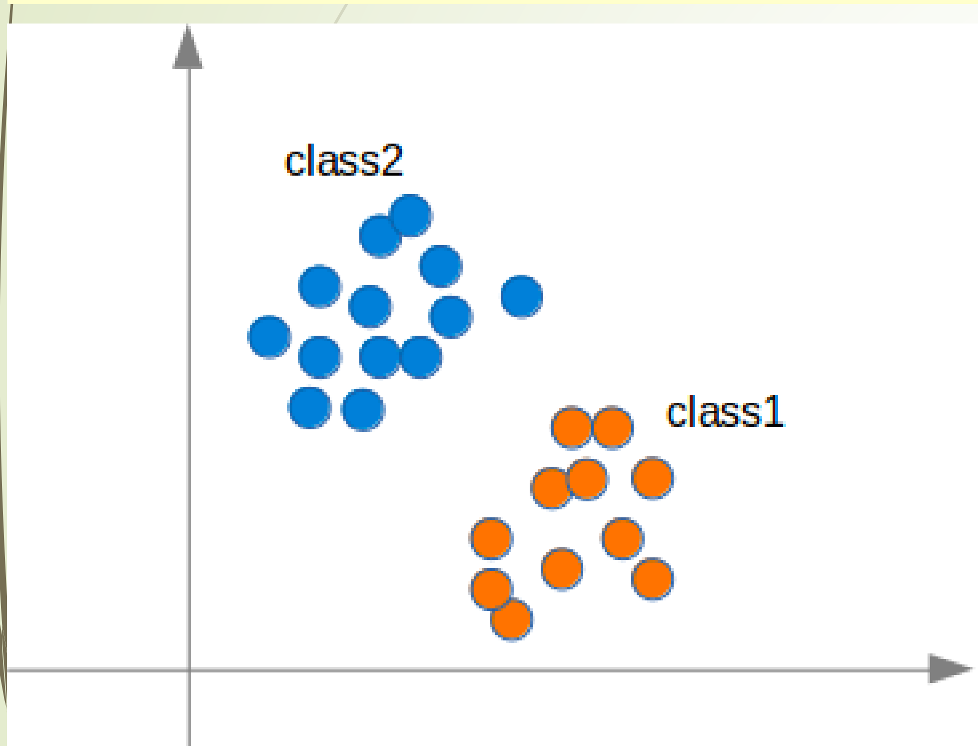


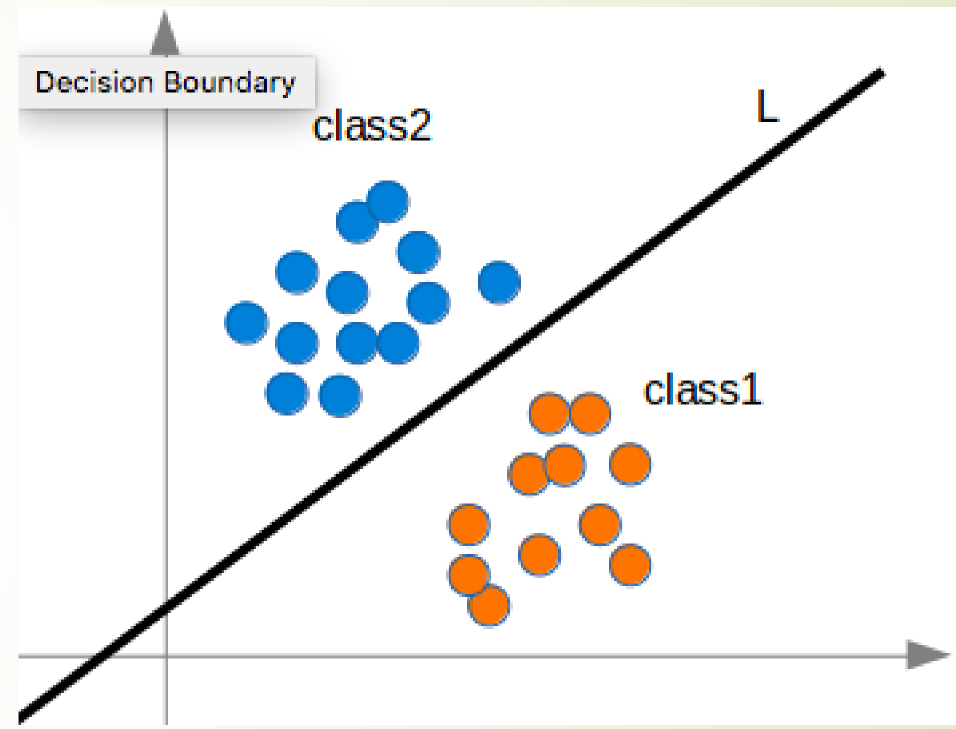


Cùng một tập mẫu có thể có nhiều đường ranh giới

Sự hội tụ của Perceptron

Luật học perceptron đảm bảo hội tụ về một lời giải sau một số hữu hạn các bước tính, nếu tồn tại lời giải

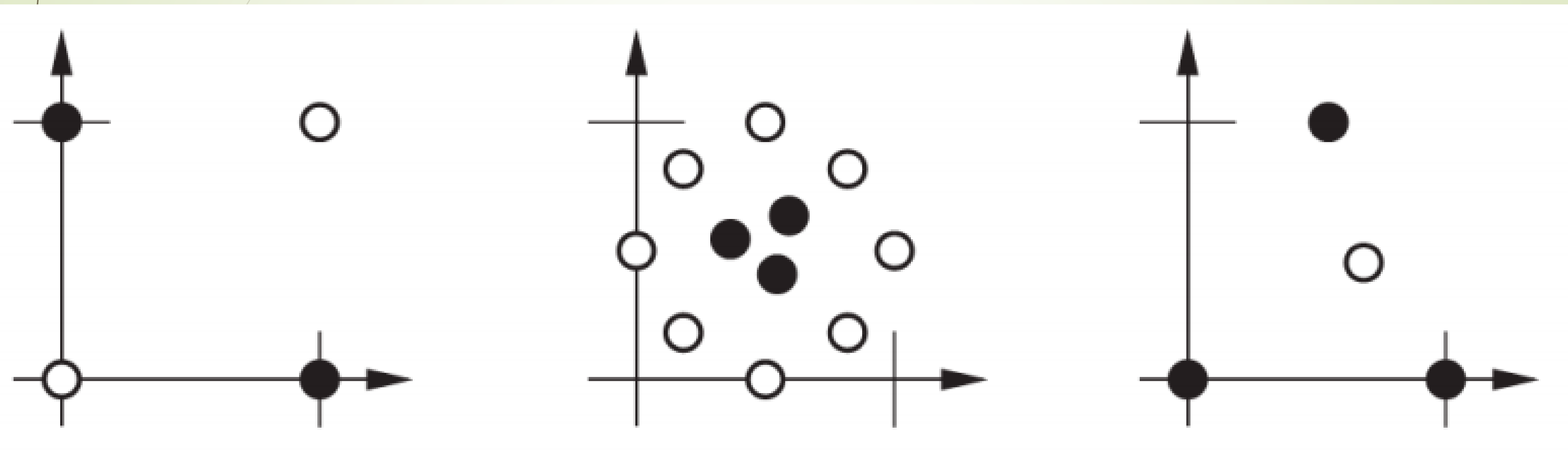




Giải quyết tốt bài toán khả tách tuyến tính

Hạn chế

Không giải quyết được bài toán không phải là "khả tách tuyến tính"



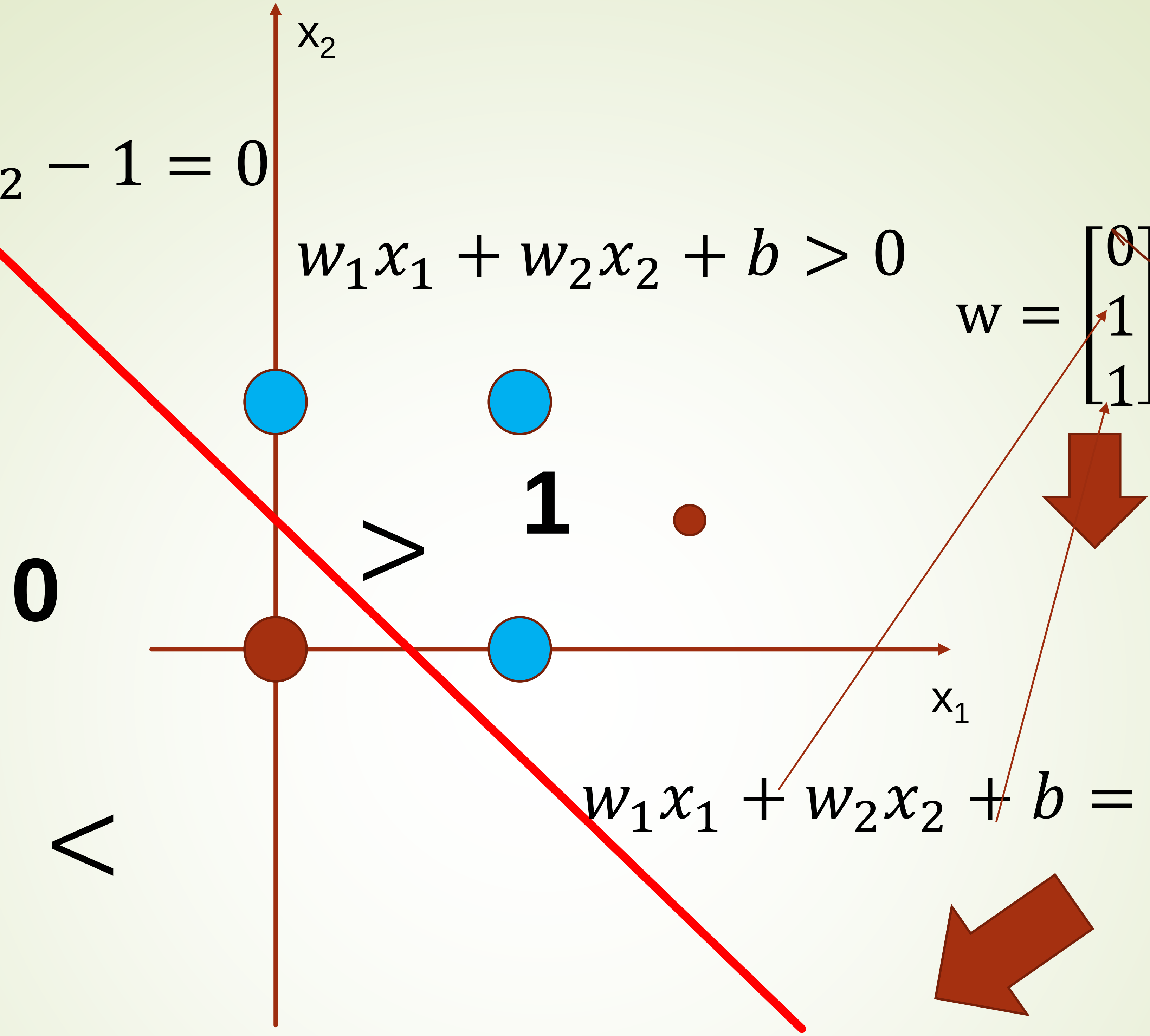
Sử dụng mạng nhiều lớp để thực hiện

$$2x_1 + 2x_2 - 1 = 0$$

$$w = \begin{bmatrix} -1 \\ 2 \\ 2 \end{bmatrix}$$

$$w_1x_1 + w_2x_2 + b > 0$$

$$w = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$



$$w_1x_1 + w_2x_2 + b = 0$$

$$w_1x_1 + w_2x_2 + b < 0\text{c}$$