

# 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](mailto:thuybinh_ktdt@utc.edu.vn)

# Nội dung môn học

**1. Thời lượng: 2 tín chỉ (9 buổi LT + 4 buổi BT)**

**2. Đánh giá học phần:**

➤ **Điểm danh: 15%**

➤ **Bài kiểm tra: 15%**

➤ **Thi kết thúc học phần: 70%**

Hình thức thi viết tự luận

Thời gian: 60 phút

# Nội dung môn học

## 1. Lý thuyết

- Một số khái niệm
- Một số mô hình mạng nơ-ron nhân tạo
- Một số ứng dụng sử dụng mạng nơ-ron

## 2. Bài tập+Thực hành:

- Chuẩn bị dữ liệu
- Thiết kế mô hình mạng nơ-ron đơn giản
- Huấn luyện mạng nơ-ron
- Một số bài toán về nhận dạng và phân lớp

# Các tài liệu tham khảo

## ➡ Sách:

1. “Mạng nơron và ứng dụng trong xử lý tín hiệu”, PGS.TSKH Trần Hoài Linh, NXB Bách Khoa -2015
2. “Phần IV: Neural Network”, Machine learning cơ bản, Vũ Hữu Tiệp
3. Hand book of Neural network Signal Processing, Hu and Hwang, 2002

## ➡ Một số trang web

1. <https://dominhhai.github.io/vi/2018/04/nn-intro/>
2. <https://phamdinhhkhanh.github.io/2019/09/29/OverviewObjectDetection.html>
3. <https://www.askpython.com/python/examples/neural-networks>

# Nội dung môn học

1. Tổng quan về neuron và mạng neuron
2. Perceptron và thuật toán Perceptron
3. Thuật toán hồi quy (Logistic/Softmax Regression)
4. Bài tập 1
5. Bài toán Perceptron trên ngôn ngữ Python
6. Mạng Perceptron nhiều lớp (Multiple Layer Perceptrons)
7. Feed Forward và Back Propagation
8. Bài tập 2
9. Kiểm tra giữa kì
10. Mạng PML và thuật toán BP trên Matlab
11. Bài tập 3
12. Bài tập 4



# Tổng quan về neuron và mạng neuron

1. Giới thiệu chung
2. Vai trò của mạng nơ-ron trong xử lý tín hiệu
3. Một số ứng dụng của mạng nơ-ron
4. Một số công cụ thực hành môn học

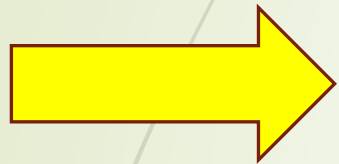
# Tổng quan về neuron và mạng neuron

- 1. Giới thiệu chung**
2. Vai trò của mạng nơ-ron trong xử lý tín hiệu
3. Một số ứng dụng của mạng nơ-ron
4. Một số công cụ thực hành môn học

# Các khái niệm

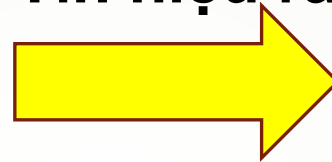
## ➤ Xử lý thông tin (Information Processing)

Tín hiệu vào



Khối xử lý thông tin

Tín hiệu ra



## ➤ Ví dụ về xử lý thông tin

- Đầu vào: Hình ảnh
- Xử lý: Phân tích, nhận dạng
- Đầu ra: Nội dung của hình ảnh là gì?

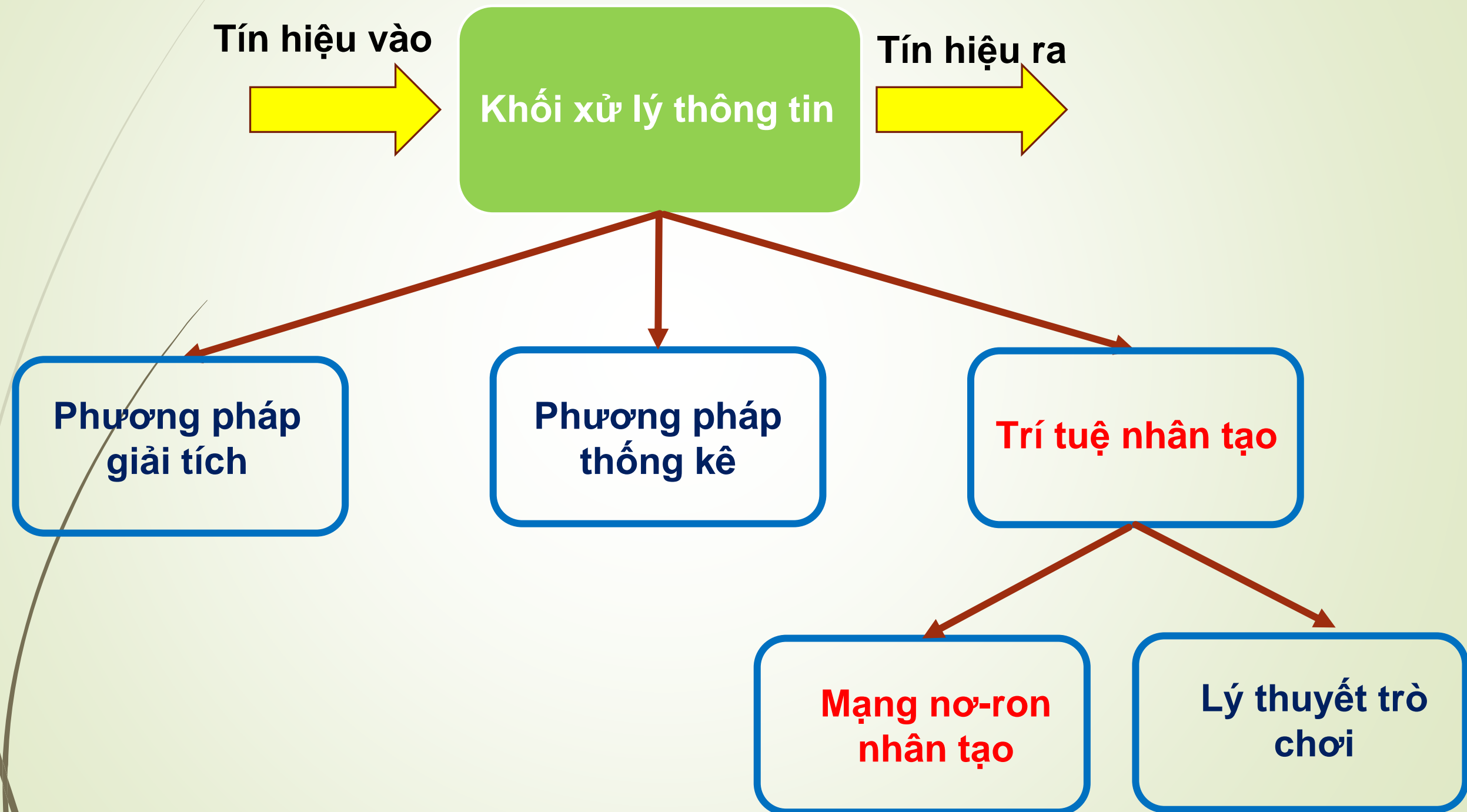


- Con người nhận biết thông qua thị giác như thế nào?
- Làm thế nào để máy tính hiểu được nội dung của hình ảnh/video?



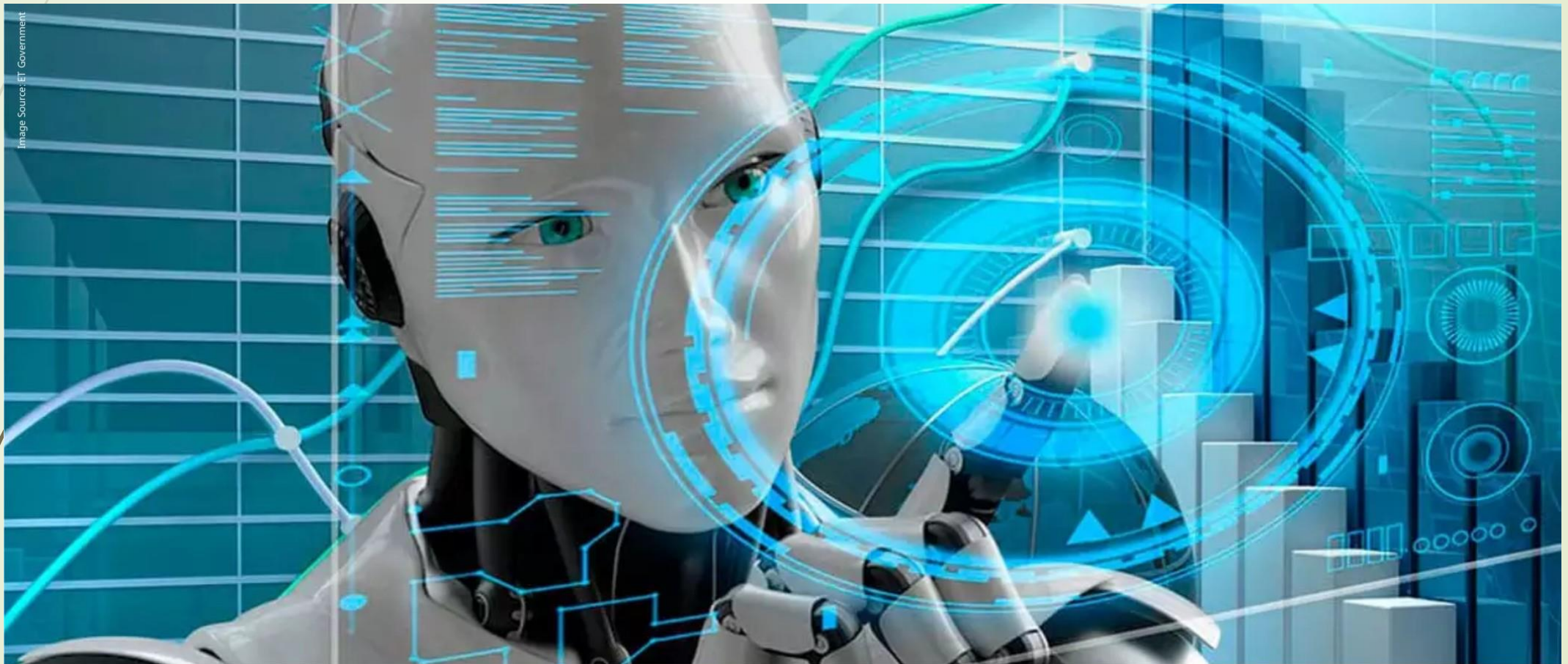


# Các khái niệm



# Trí tuệ nhân tạo

Nghiên cứu và phát triển các mô hình mô phỏng khả năng tư duy, phân tích và xử lý thông tin của con người



# Hệ trí tuệ nhân tạo

- Khả năng biểu diễn (representation): Dùng một ngôn ngữ hình thức để mô tả một vấn đề
- Khả năng suy diễn (reasoning): giải quyết một vấn đề
  - Nội suy và ngoại suy
  - Có cơ chế điều khiển để xác định phải thực hiện thao tác gì
- Khả năng học (learning): Dựa trên tri thức đã có, học các quy luật để có thể suy diễn trong tình huống tương tự hoặc dự đoán tình huống mới



# Mạng neuron nhân tạo

- Mô phỏng mạng neuron sinh học nhằm phục vụ cho một bài toán cụ thể (dự đoán, nhận dạng, phát hiện,...)
- Không hướng đến sự phức tạp của mạng neuron sinh học
- Một số điểm giống nhau giữa NN và ANN
  - Tạo ra các khối là đơn giản
  - Các liên kết trong mạng neuron cho phép xác định chức năng, nhiệm vụ của từng mạng
  - Có khả năng học và tái tạo



# Mạng neuron

Mạng neuron nhân tạo



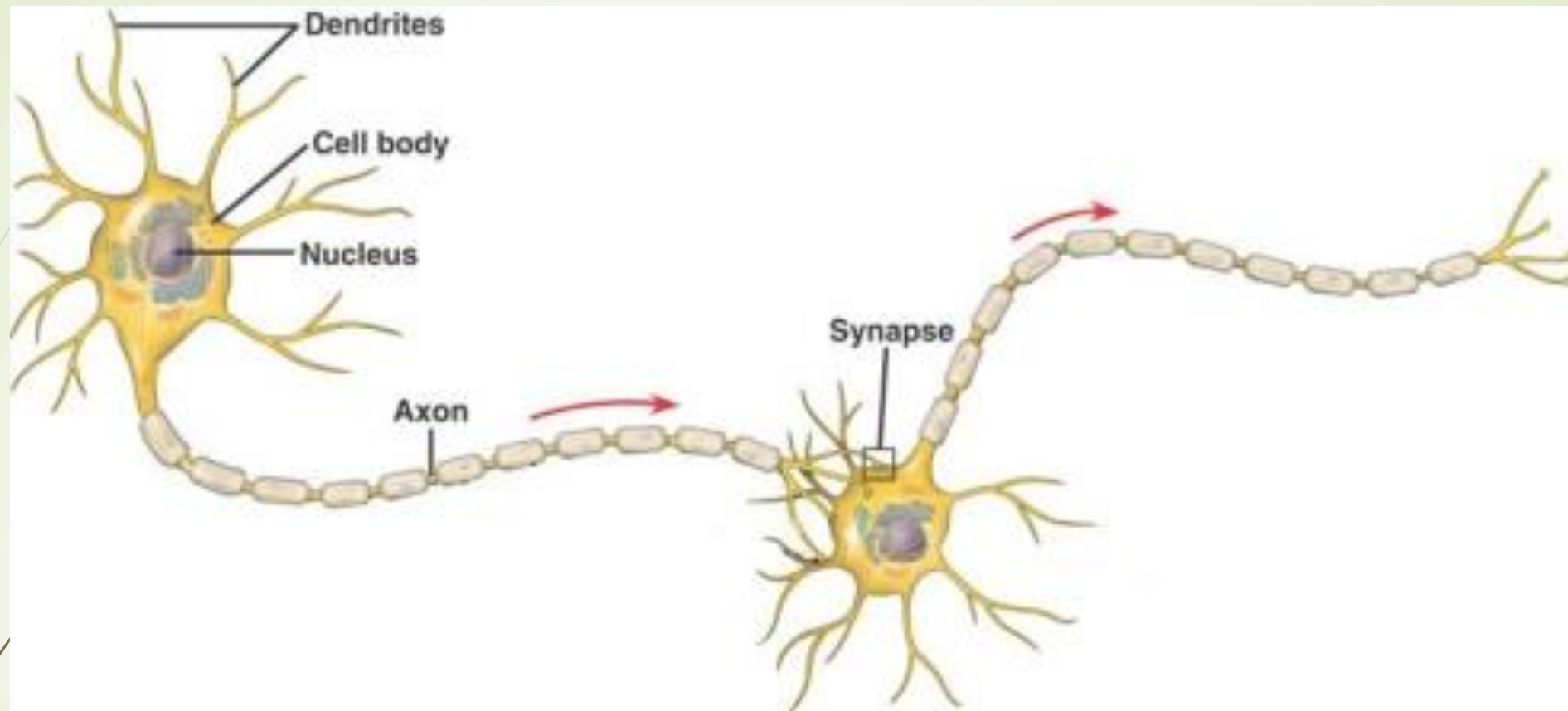
Mạng neuron

Mạng neuron nhân tạo thực chất là một máy tính được thiết kế để mô tả lại hoạt động của não người khi thực hiện một nhiệm vụ nào đó

## Ưu điểm

- Xử lý hệ thống cả tuyến tính lẫn phi tuyến
- Input-Output mapping
- Có khả năng thích nghi cao
- Đáp ứng đầu ra có độ tin cậy tốt
- Có khả năng tích hợp với mạch điện tử

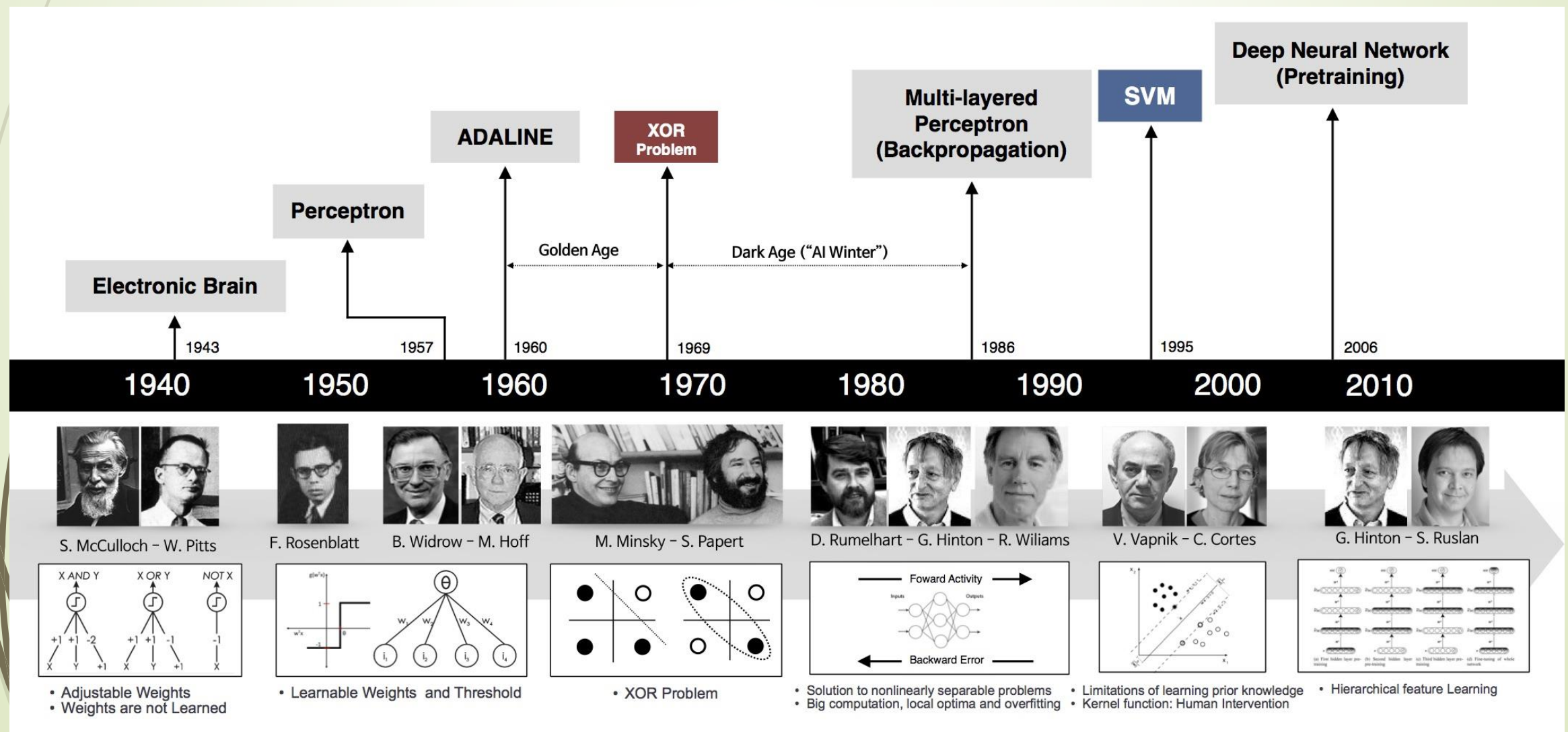
# Tế bào thần kinh



1. Tín hiệu lan truyền giữa các tế bào thần kinh là các tín hiệu điện - hoá
2. Thông tin được trao đổi ở các khớp thần kinh khi tổng các tín hiệu vào lớn hơn một ngưỡng - Tế bào được kích thích
3. Tế bào bị kích thích sẽ đưa ra các tín hiệu ra tương ứng

# Lịch sử về ANN

- Thập niên 1940s: Sự xuất hiện của neuron network cùng máy tính điện tử
- 1950-1960: Kỷ nguyên vàng (Golden age) đầu tiên của neural network
- 1970-1980: Không phát triển
- 1980s: Sự xuất hiện của các mạng lan truyền ngược





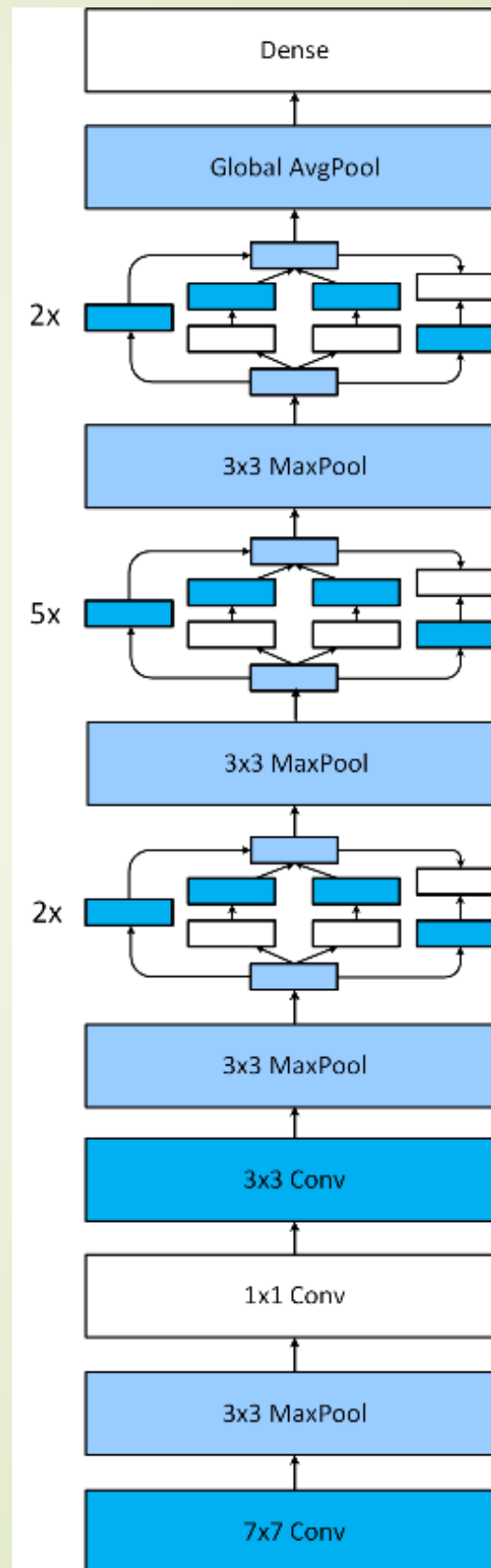
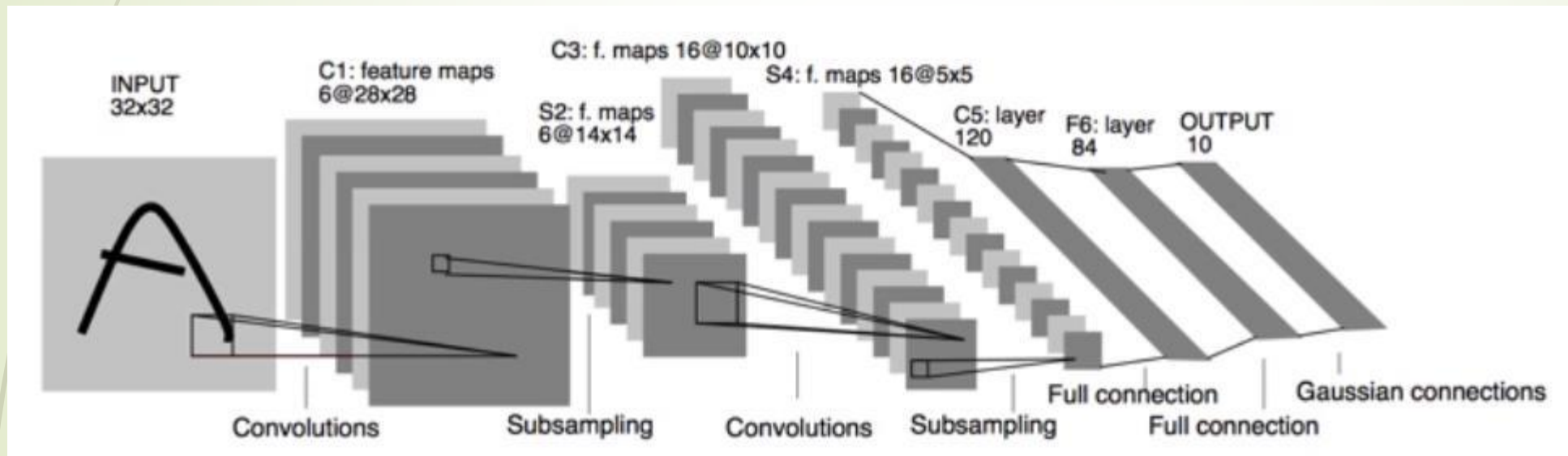
# Tổng quan về neuron và mạng neuron

1. Giới thiệu chung
2. Vai trò của mạng nơ-ron trong xử lý tín hiệu
3. Một số ứng dụng của mạng nơ-ron
4. Một số công cụ thực hành môn học



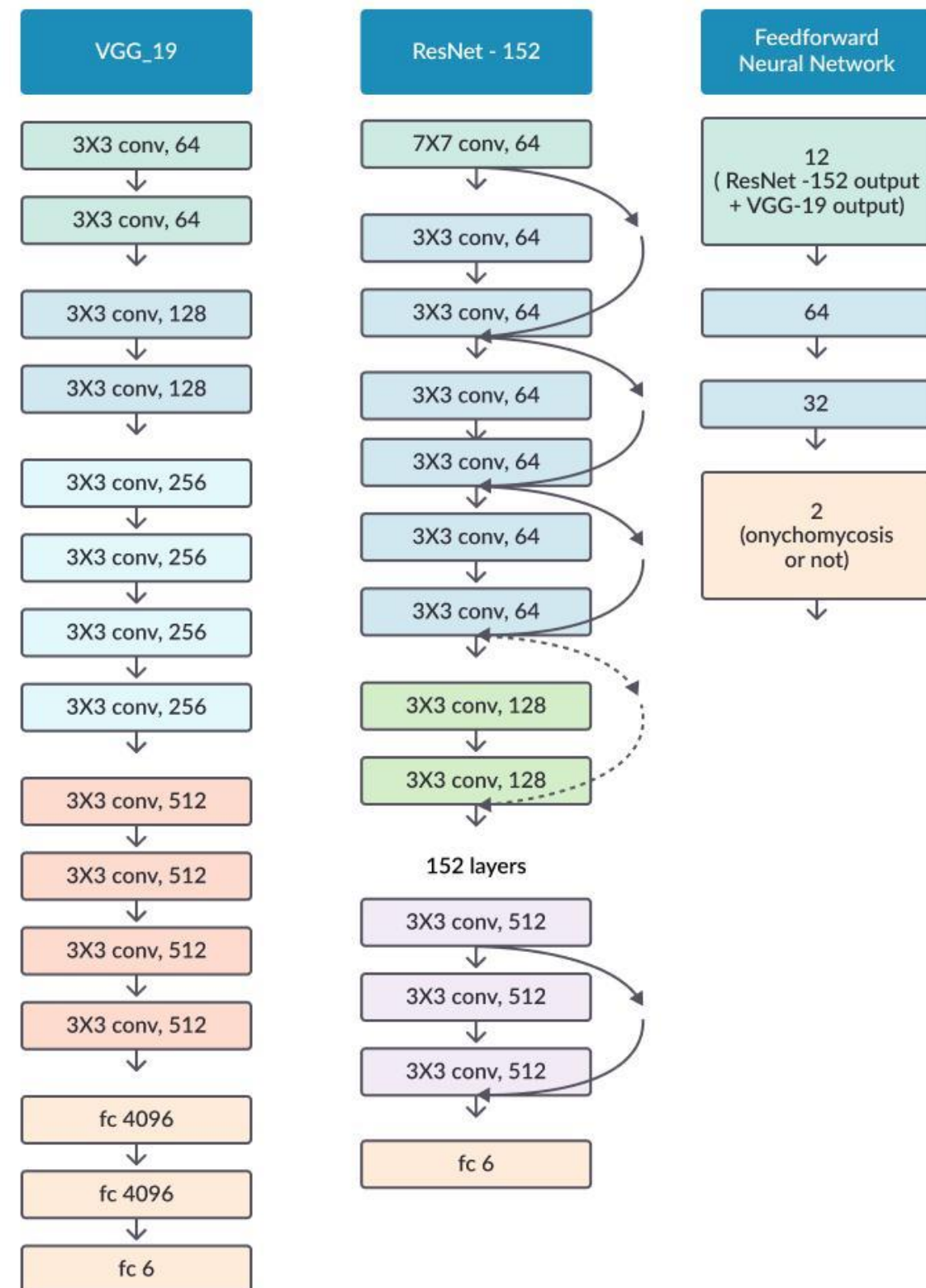
# Một số mạng DNN

## GoogleNet



# Một số mạng DNN

## VGG-19 ResNet-152



# Tổng quan về neuron và mạng neuron

1. Giới thiệu chung
2. Vai trò của mạng nơ-ron trong xử lý tín hiệu
3. Một số ứng dụng của mạng nơ-ron
4. Một số công cụ thực hành môn học

# Các ứng dụng của ANN

- Quân sự, quốc phòng:
  - Chỉ huy vũ khí, theo dõi mục tiêu
  - Ngăn chặn và tìm kiếm tội phạm
- Ngân hàng
  - Đánh giá các hoạt động tín dụng, đưa ra các dự báo
  - Đo lường rủi ro tín dụng
  - Phân loại các công ty
- Giao thông Vận tải
  - Máy bay không người lái (UAV)
  - Hệ thống dẫn đường tự động
- Phân tích âm thanh, hình ảnh
- Công nghiệp giải trí
- Công nghiệp Robot



# Tổng quan về neuron và mạng neuron

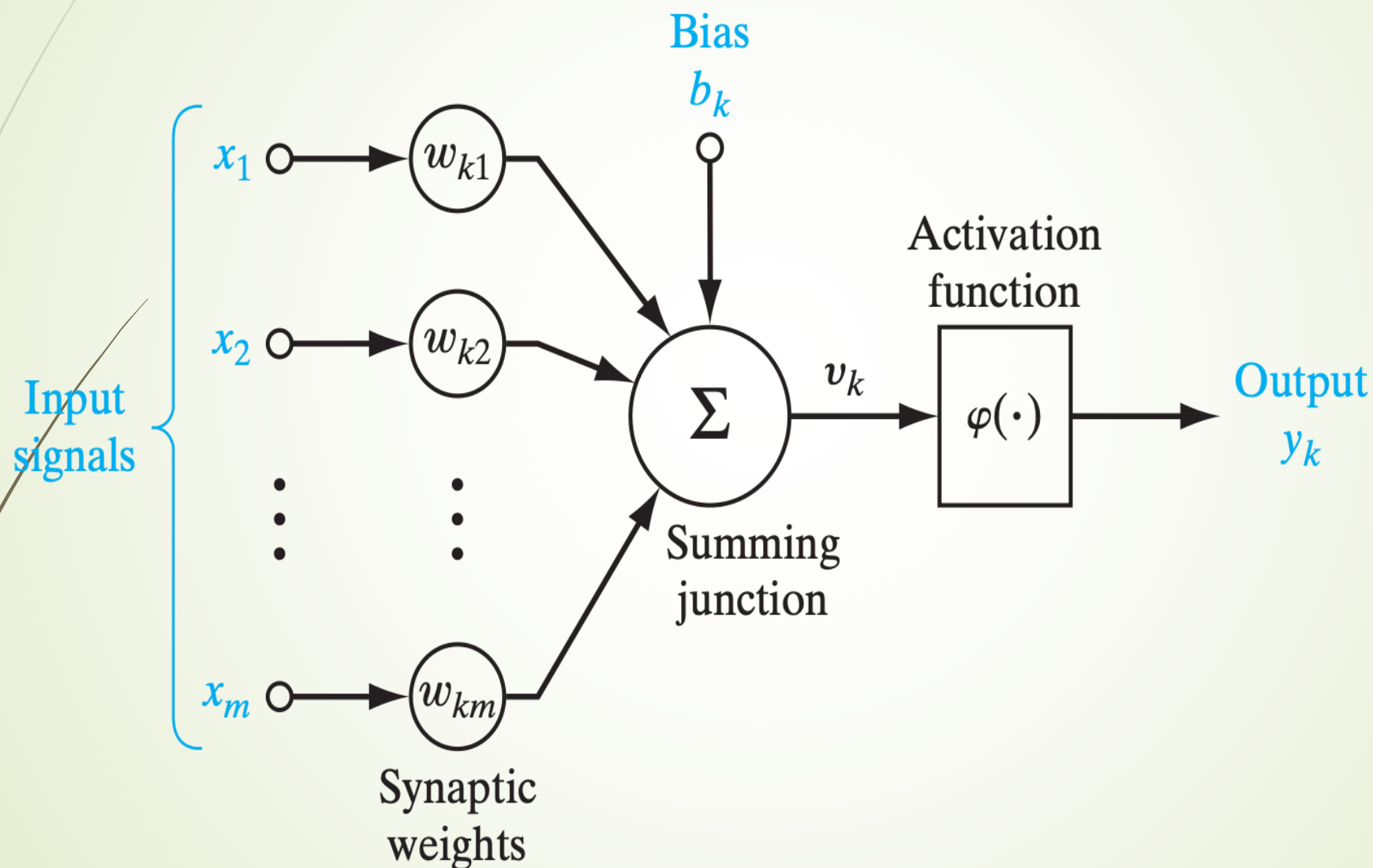
1. Giới thiệu chung
2. Vai trò của mạng nơ-ron trong xử lý tín hiệu
3. Một số ứng dụng của mạng nơ-ron
4. Một số công cụ thực hành môn học

# Công cụ thực hành môn học

- Ngôn ngữ lập trình Matlab
- Ngôn ngữ lập trình Python
- Lập trình trên Google Colab
- Công cụ gán nhãn (LabelMe, Label Img)

# Mô hình perceptron

Đề xuất bởi McCulloch - Pitt năm 1943



# Mô hình perceptron

24

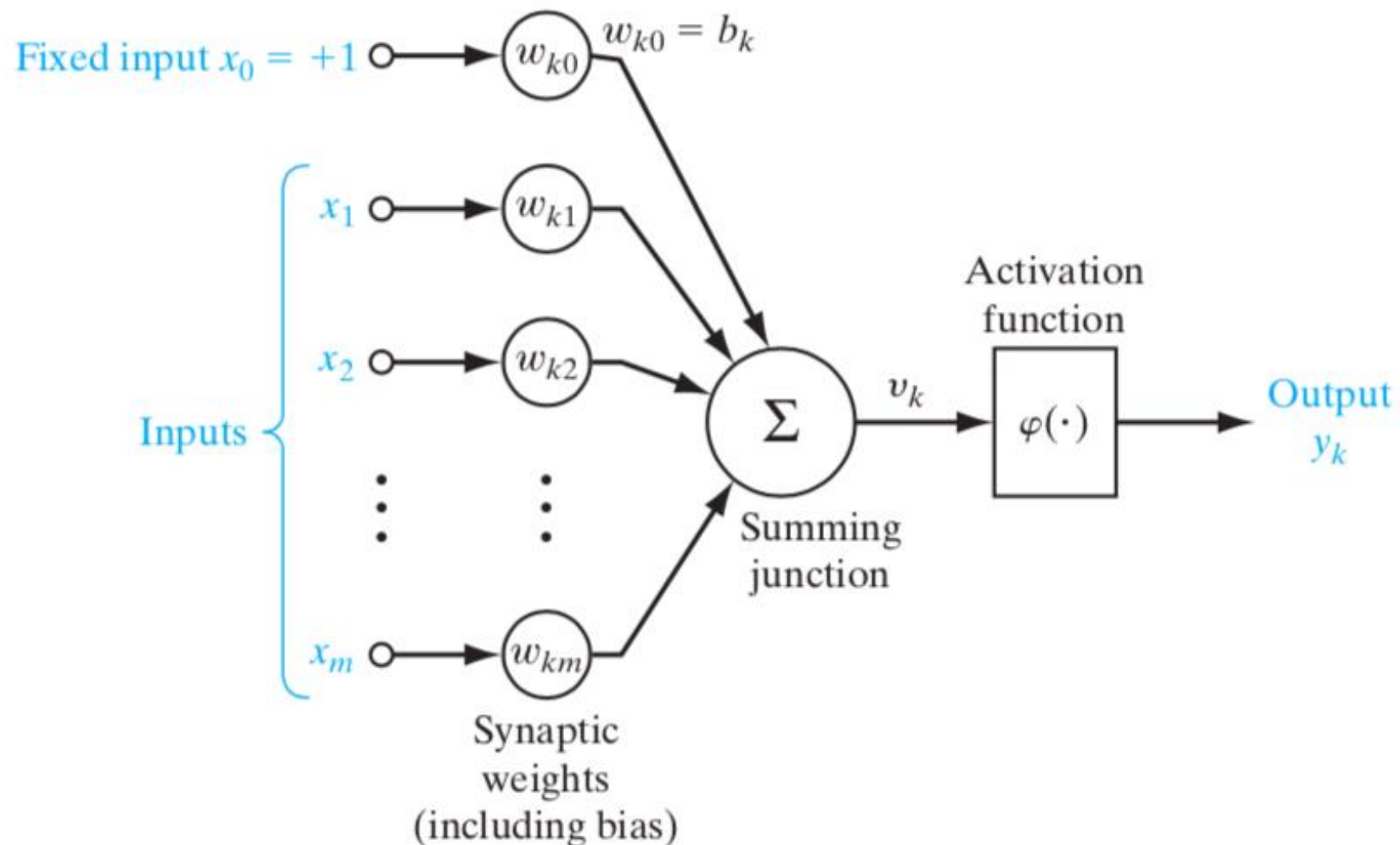
Trọng số      Tín hiệu vào

$$u_k = \sum_{j=1}^m w_{kj} x_j$$
$$y_k = \varphi(u_k + b_k)$$

$$v_k = u_k + b_k$$

$$x_0 = +1$$
$$w_{k0} = b_k$$

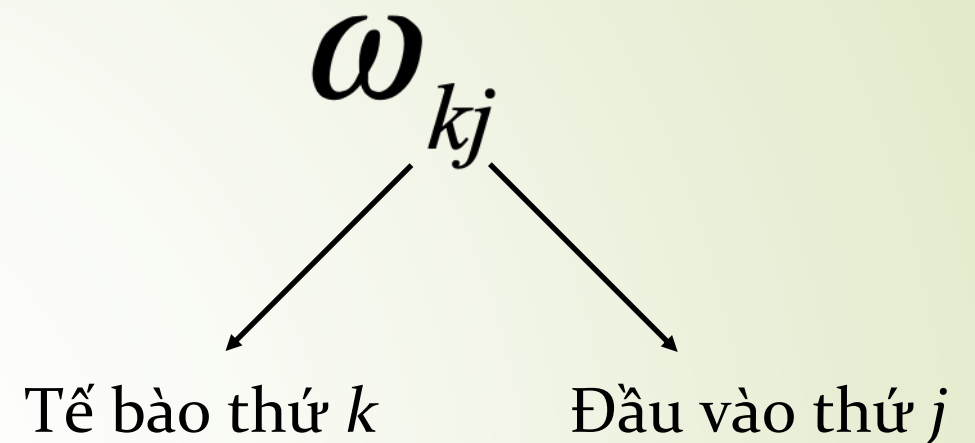
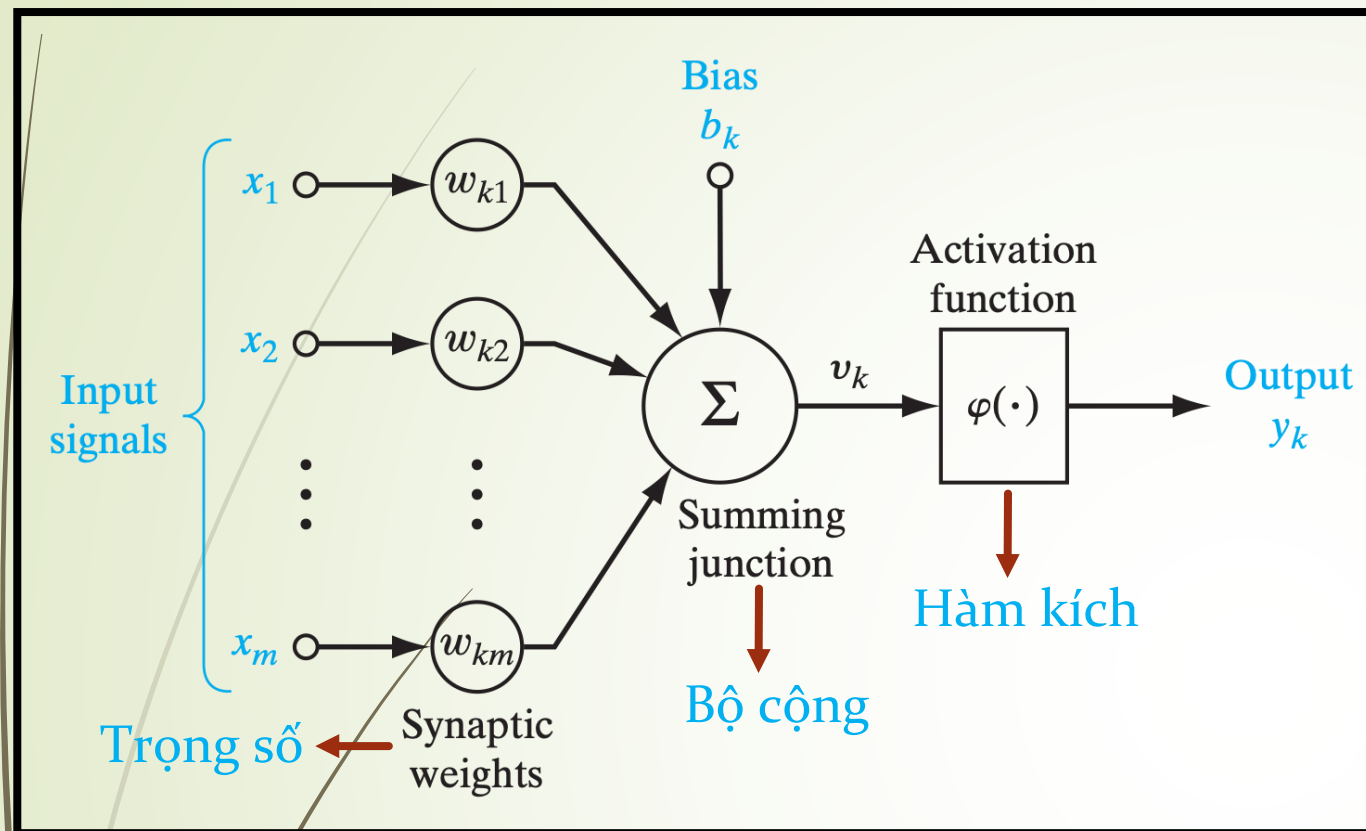
$$v_k = \sum_{j=0}^m w_{kj} x_j$$
$$y_k = \varphi(v_k)$$





# Mô hình perceptron

25

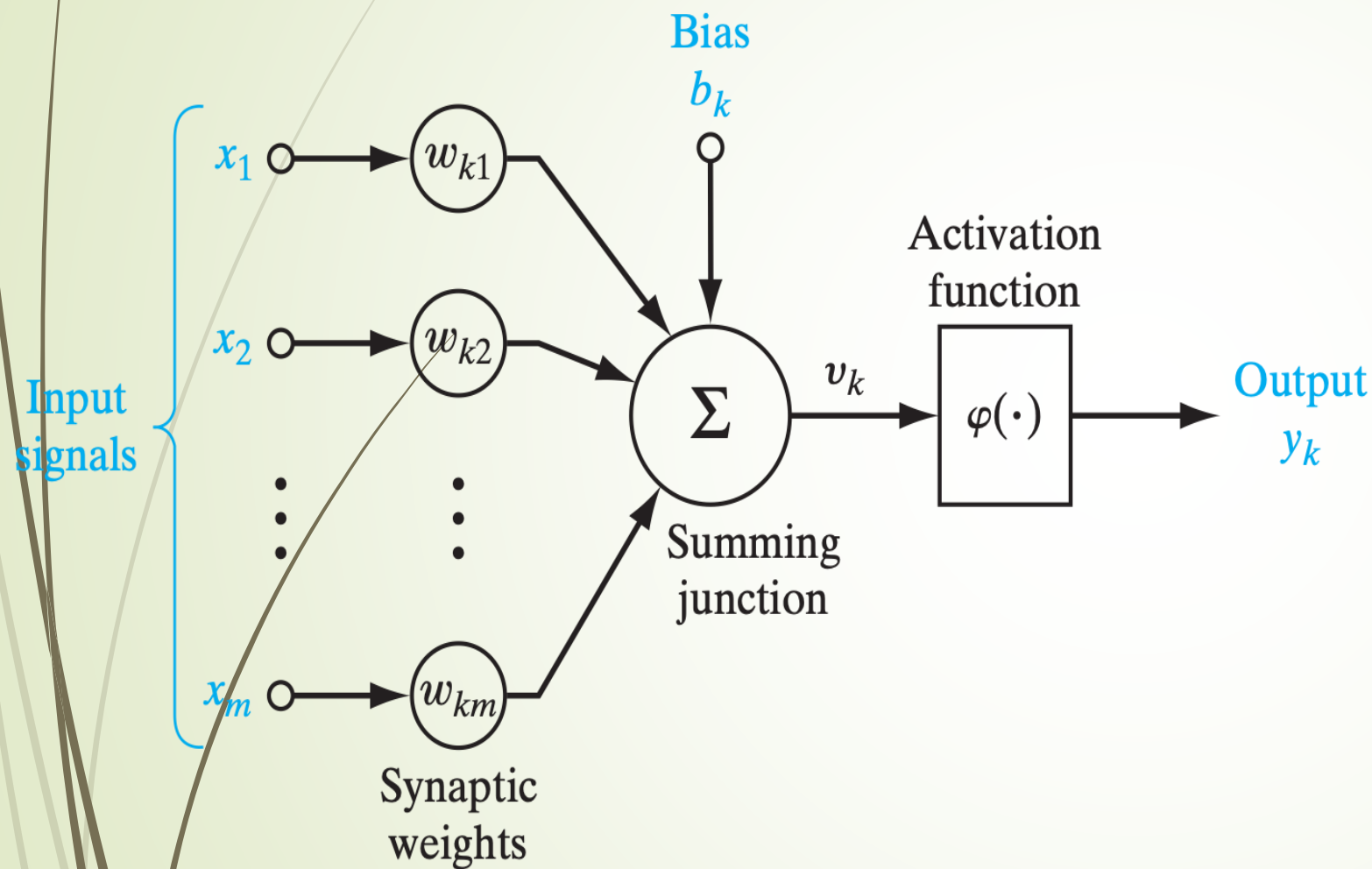


1. Tập hợp các liên kết, mỗi liên kết có một trọng số tương ứng
2. Bộ cộng tín hiệu
3. Hàm kích thích - giới hạn tín hiệu ra của một neuron
4. Độ lệch Bias  $b_k$  được dùng để tăng hoặc giảm ngưỡng tác động lên hàm kích thích

# Mô hình perceptron

26

Đề xuất bởi McCulloch - Pitt năm 1943



Hàm kích hoạt là hàm ngưỡng

$$y_k = \begin{cases} 0 & \text{nếu } \sum x_i w_{ki} \leq \text{ngưỡng} \\ 1 & \text{nếu } \sum x_i w_{ki} > \text{ngưỡng} \end{cases}$$



$b_k = -\text{ngưỡng}$

$$y_k = \begin{cases} 0 & \text{nếu } \sum x_i w_{ki} + b_k \leq 0 \\ 1 & \text{nếu } \sum x_i w_{ki} + b_k > 0 \end{cases}$$

Một nơ-ron có thể nhận nhiều đầu vào (dạng nhị phân) và đưa một kết quả đầu ra duy nhất (dạng nhị phân)

# Mô hình perceptron

**Ví dụ:** Việc đi học của một sinh viên nam phụ thuộc vào các yếu tố sau

1. Giáo viên có điểm danh hay không? ( $x_1; w_{k1} = 0,1$ )
2. Trời hôm nay có nắng hay không? ( $x_2; w_{k2} = 0,2$ )
3. Bạn gái có đi học hay không? ( $x_3; w_{k3} = 0,4$ )
4. Môn học có hữu ích hay không? ( $x_4; w_{k4} = 0,3$ )



Việc “Bạn gái có đi học hay không” ảnh hưởng tới 40% quyết định đi học; “Giáo viên có điểm danh hay không” chỉ ảnh hưởng tới 10% quyết định đó.

$$y_k = \begin{cases} 0 & \text{nếu } \sum x_i w_{ki} + b_k \leq 0 \\ 1 & \text{nếu } \sum x_i w_{ki} + b_k > 0 \end{cases}$$

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$
$$\mathbf{w}_k = \begin{bmatrix} w_{k1} \\ w_{k2} \\ w_{k3} \\ w_{k4} \end{bmatrix}$$

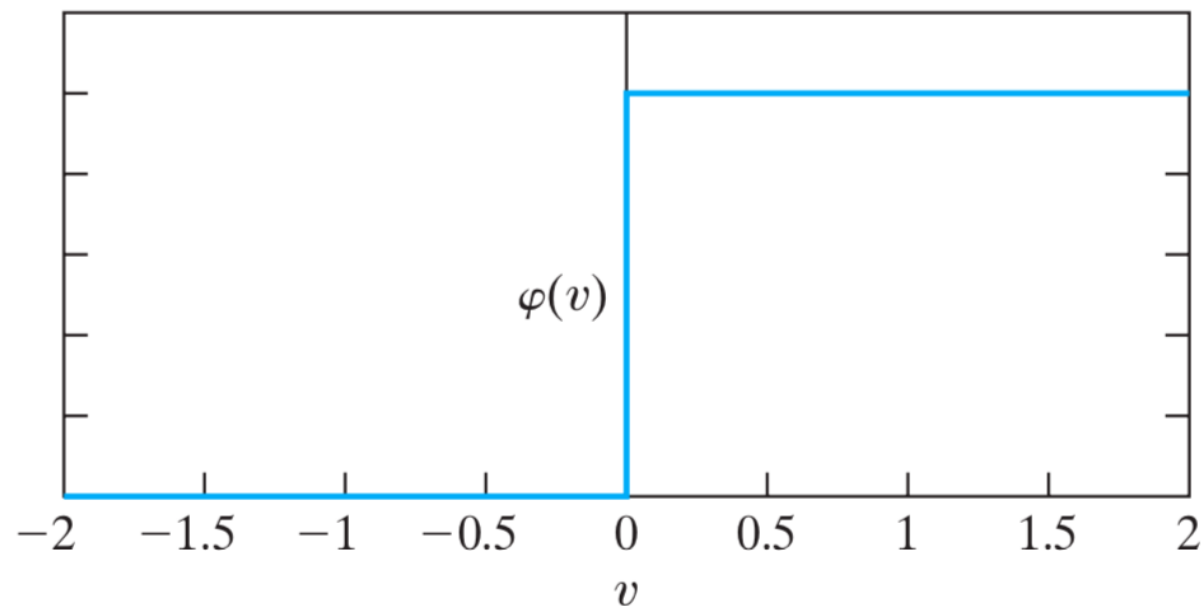


$$y_k = \begin{cases} 0 & \text{nếu } \mathbf{w}_k^T \mathbf{x} + b_k \leq 0 \\ 1 & \text{nếu } \mathbf{w}_k^T \mathbf{x} + b_k > 0 \end{cases}$$

# Hàm kích thích

28

## Hàm ngưỡng Threshold



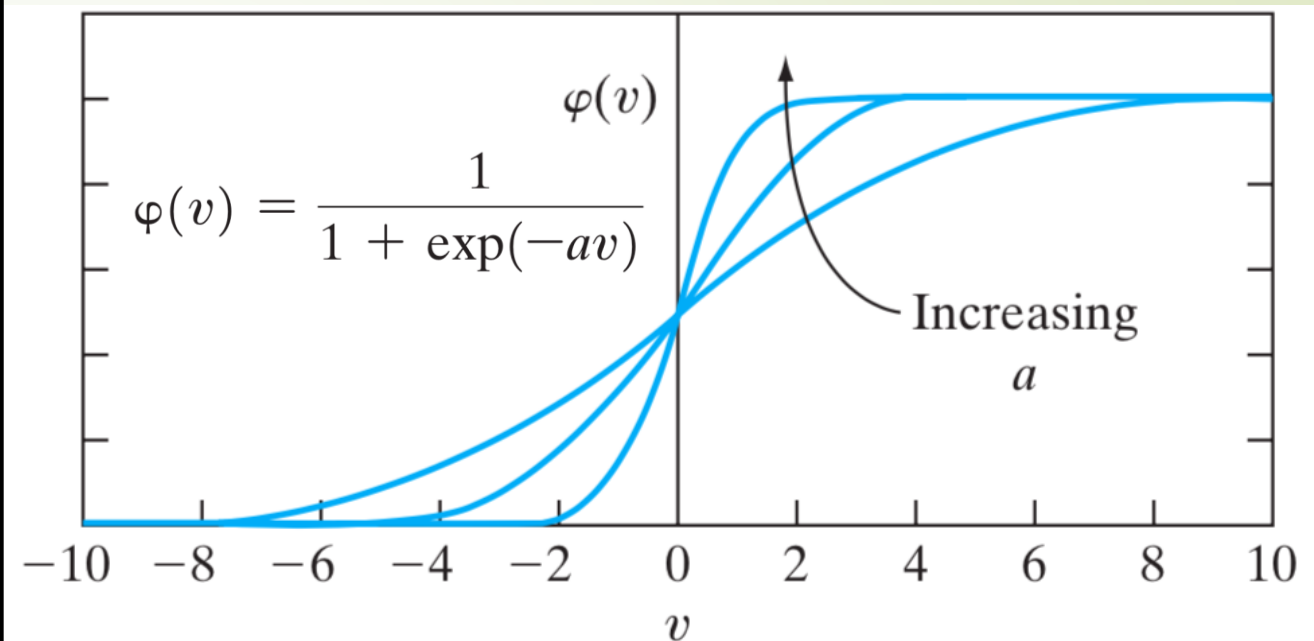
$$\varphi(v) = \begin{cases} 1 & \text{if } v \geq 0 \\ 0 & \text{if } v < 0 \end{cases}$$



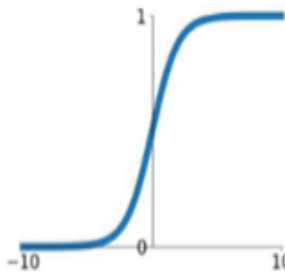
$$y_k = \begin{cases} 1 & \text{if } v_k \geq 0 \\ 0 & \text{if } v_k < 0 \end{cases}$$

$$v_k = \sum_{j=1}^m w_{kj} x_j + b_k$$

## Hàm Sigmoid/Logistic



Function	Equation	Range	Derivative
Sigmoid (Logistic)	$f(x) = \frac{1}{1 + e^{-x}}$	0,1	$f'(x) = f(x)(1 - f(x))$



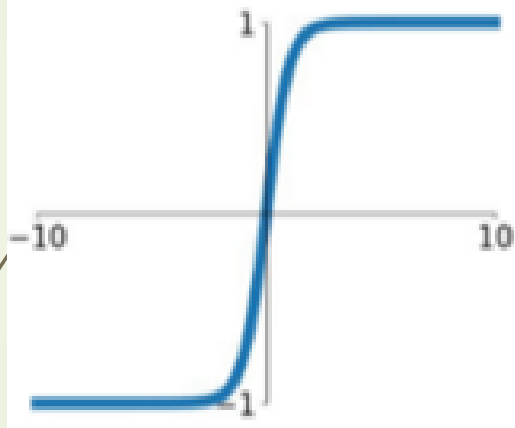
- Đầu ra nằm trong khoảng [0,1] → phân lớp đối tượng
- Đòi hỏi nhiều tính toán → hội tụ chậm
- Vấn đề bão hòa



# Hàm kích thích

## Hàm Tanh

Function	Equation	Range	Derivative
<b>Tanh</b> (Hyperbolic tangent)	$f(x) = \frac{2}{1+e^{-2x}} - 1$	$[-1, 1]$	$f'(x) = 1 - f(x)^2$

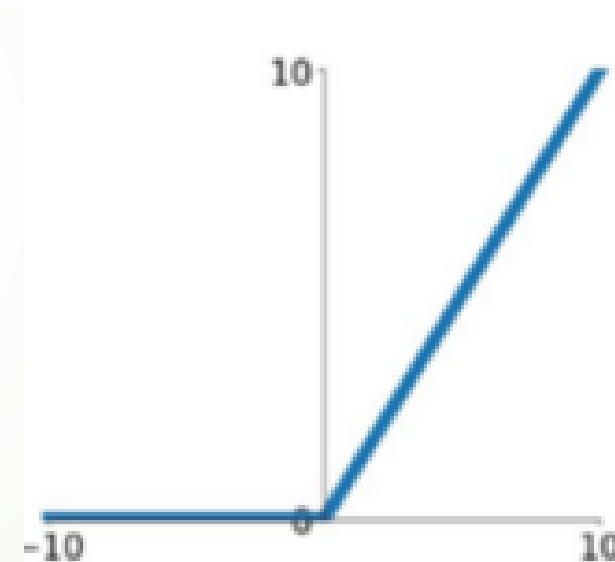


$$f(x) = \frac{2}{1+e^{-2x}} - 1 = \frac{1-e^{-2x}}{1+e^{-2x}} = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

- Đầu ra nhận giá trị trong khoảng  $[-1, 1]$
- Đòi hỏi nhiều tính toán
- Gặp phải vấn đề bão hòa
- Tâm đối xứng là gốc tọa độ  $\rightarrow$  tối ưu hàm mất mát trở nên dễ dàng hơn

## Hàm ReLu

Function	Equation	Range	Derivative
<b>ReLu</b> (Rectified Linear Unit)	$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$[0, +\infty)$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$



- Được sử dụng nhiều nhất
- Dễ dàng tính toán để mạng neuron hội tụ nhanh
- Không gặp phải vấn đề bão hòa đối với các neuron có giá trị dương
- Độ bão hòa và độ dốc biến mất chỉ xảy ra khi các giá trị âm được đưa tới hàm kích hoạt

# Ví dụ

Ví dụ 1: Cho neuron với 2 đầu vào:  $\mathbf{x} = \begin{bmatrix} 5 \\ -3 \end{bmatrix}$ ; trọng số tương ứng:

$\mathbf{w} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$ ; độ lệch (bias)  $b = 0,5$ . Xác định đầu ra của neuron với hàm kích hoạt

1. Hàm ngưỡng (threshold)
2. Hàm sigmoid
3. Hàm tank

**Giải:**

$$v = \mathbf{w}^T \mathbf{x} + b = [1 \ 2] \begin{bmatrix} 5 \\ -3 \end{bmatrix} + 0.5 = 1.5 + 2.(-3) + 0.5 = -0.5$$

1. Hàm ngưỡng (threshold):  $v = -0.5 < 0 \rightarrow y = f(v) = 0$
2. Hàm sigmoid:

$$f(v) = \frac{1}{1 + e^{0.5}} = 0.38$$

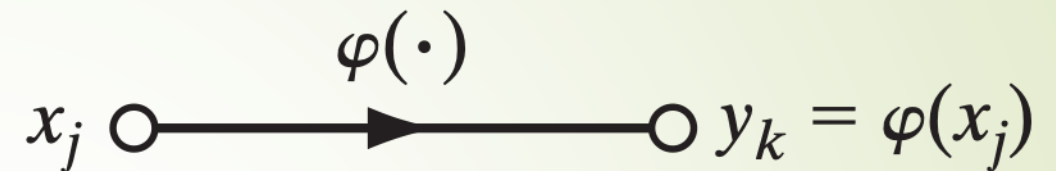
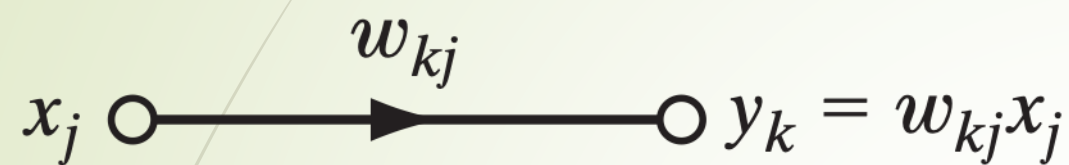
1. Hàm tank:

$$f(v) = \frac{2}{1 + e^{2.0.5}} - 1 = \frac{2}{1 + e} - 1 = -0.46$$

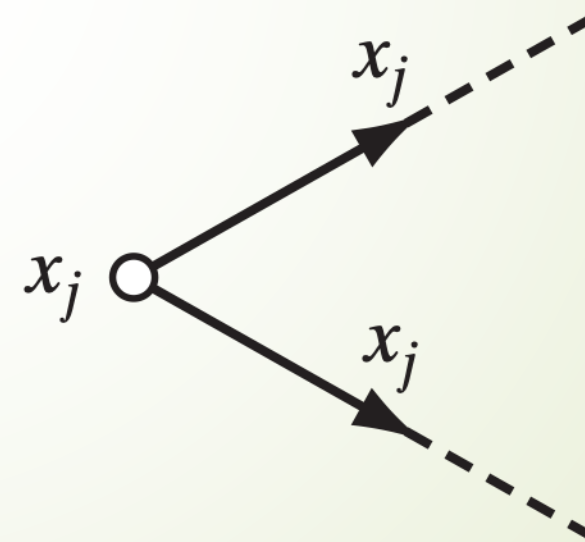
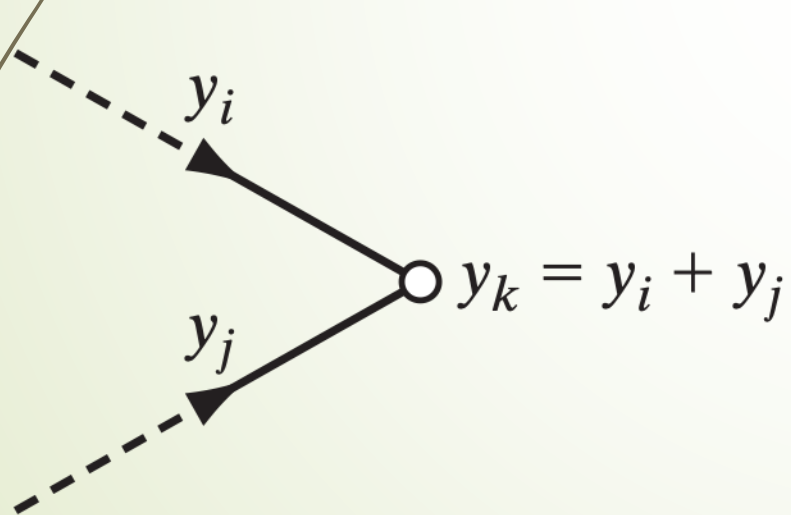
# Mạng neuron dưới dạng graph

31

Luật 1: Tín hiệu trên một khớp nối có chiều theo chiều mũi tên



Luật 2: Tín hiệu tại một nút bằng tổng đại số các tín hiệu đi vào nút

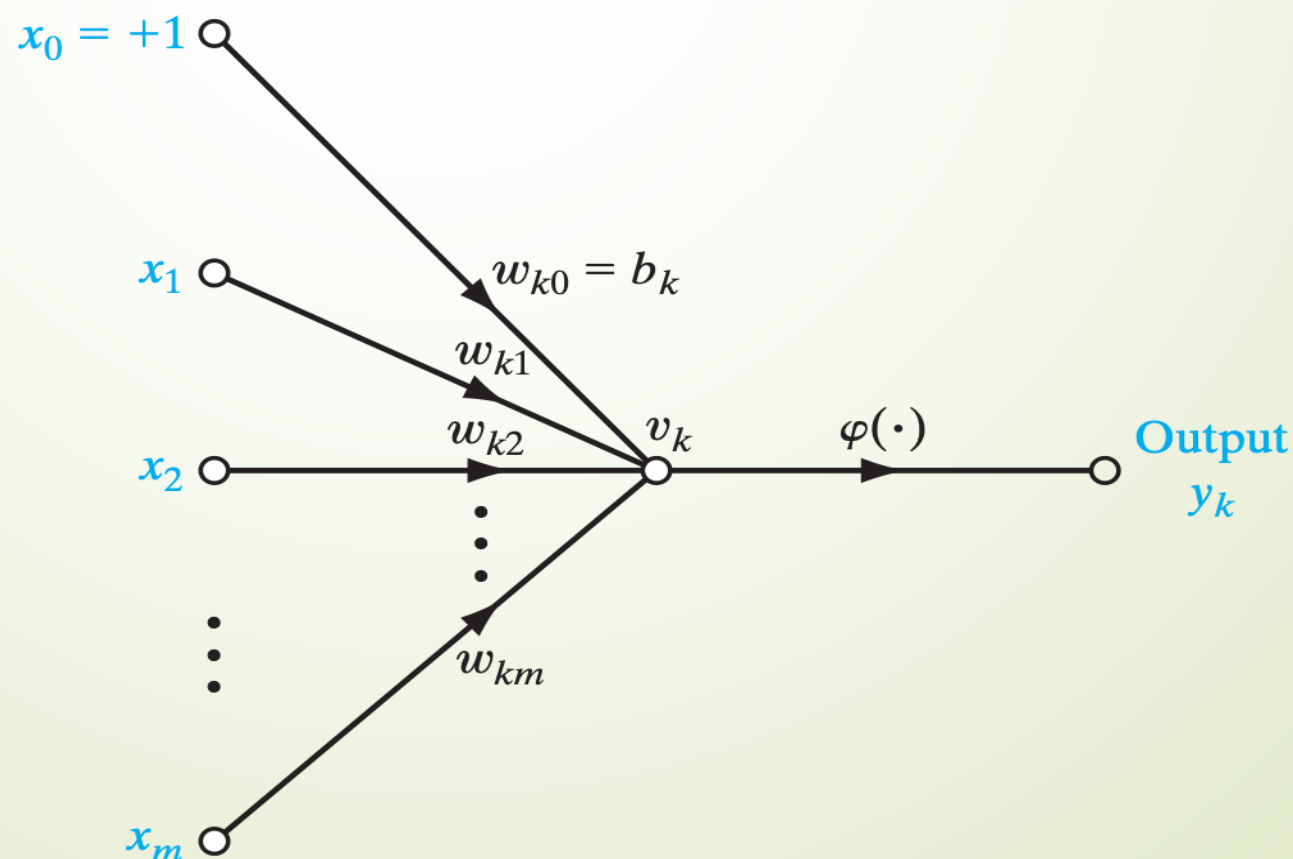


Luật 3: Tín hiệu đi ra tại một nút bằng nhau ở tất cả các nhánh

# Mạng neuron dưới dạng graph

32

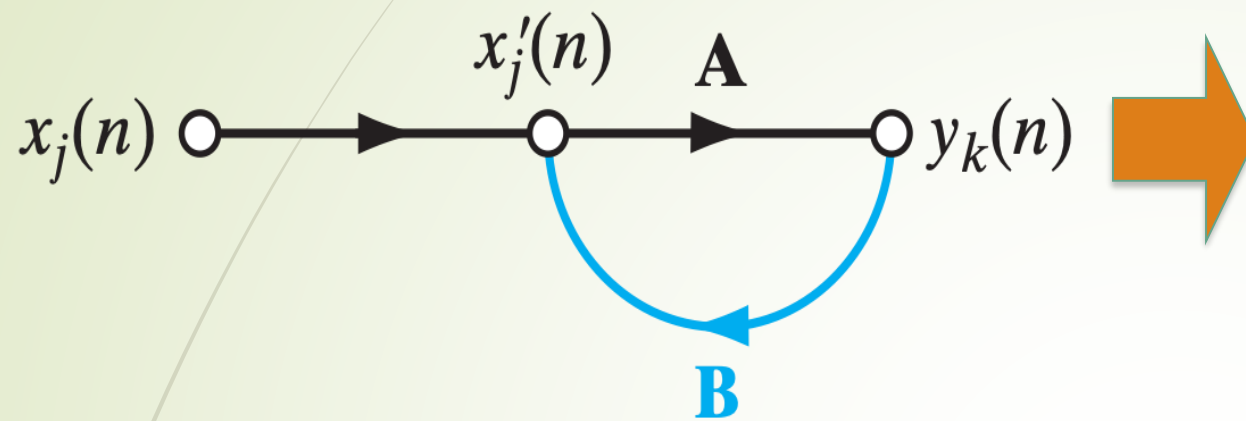
- Mỗi neuron được biểu diễn bằng một tập các liên kết đầu vào, bao gồm cả *bias*, và một liên kết đầu ra
- Các liên kết được thể hiện thông qua các trọng số  $w_{kj}$
- Tổng các giá trị đầu vào gồm trọng số sẽ quyết định kích hoạt liên kết đầu ra của tế bào neuron
- Liên kết được kích hoạt sẽ quyết định giá trị ra Output:  $y_k$





# Hồi quy (Recurrent)

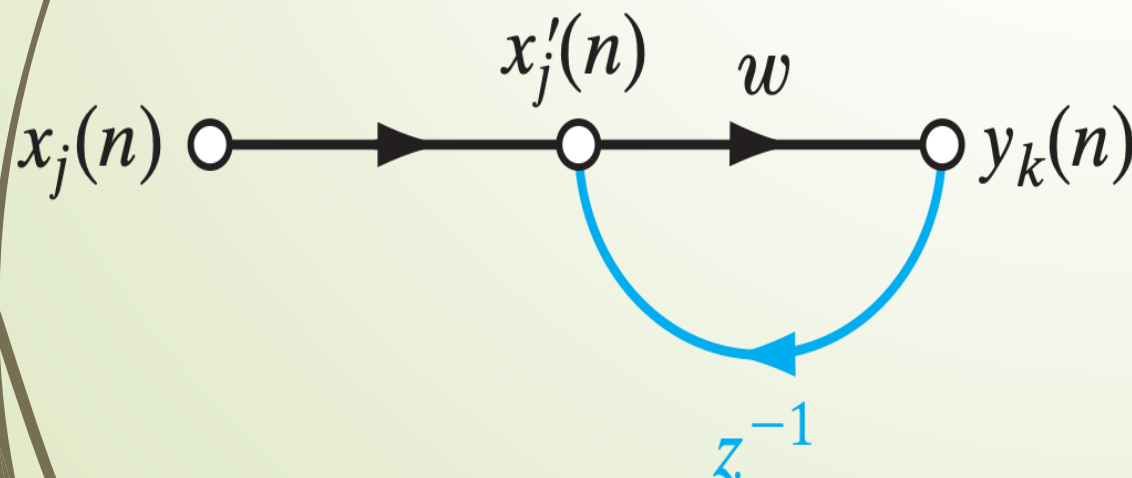
33



$$y_k(n) = \mathbf{A}[x'_j(n)]$$

$$x'_j(n) = x_j(n) + \mathbf{B}[y_k(n)]$$

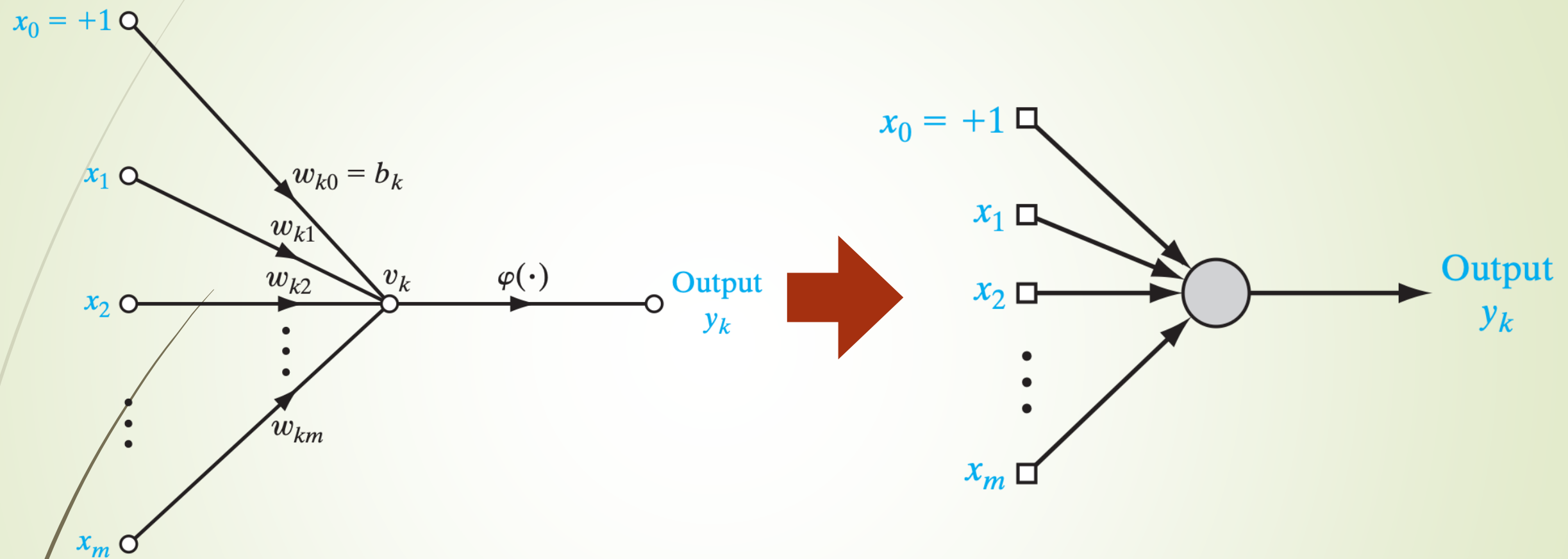
$$y_k(n) = \frac{\mathbf{A}}{1 - \mathbf{A}\mathbf{B}} [x_j(n)]$$



$$\begin{aligned} \frac{\mathbf{A}}{1 - \mathbf{A}\mathbf{B}} &= \frac{w}{1 - wz^{-1}} \\ &= w(1 - wz^{-1})^{-1} \end{aligned}$$

# Cấu trúc mạng neuron

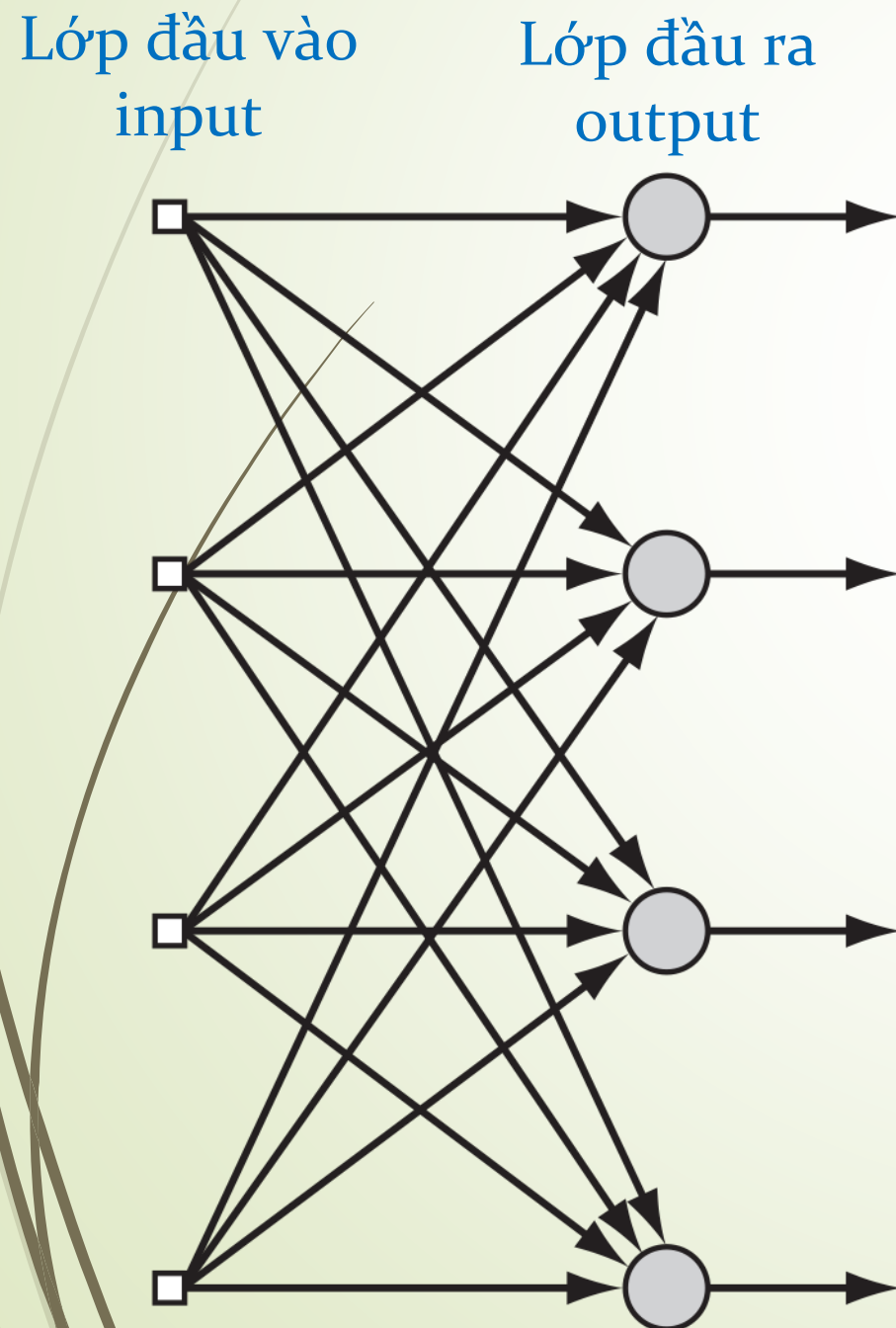
34



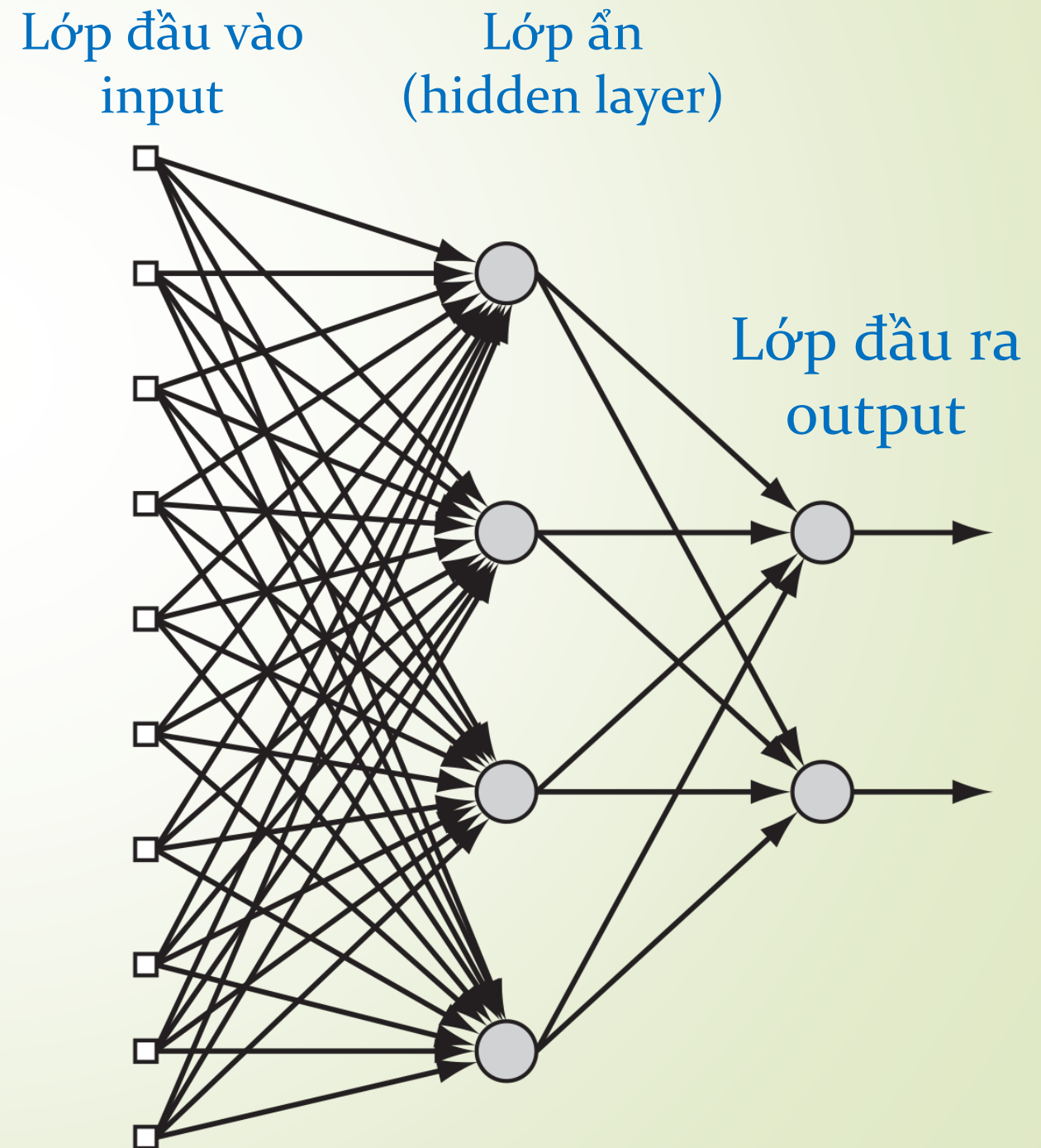
# Cấu trúc mạng neuron

35

Mạng neuron 1 lớp truyền thẳng



Mạng neuron nhiều lớp truyền thẳng



# Các quá trình học của mạng neuron

## ➤ Phân chia theo cách xử lý tín hiệu

- Học ngoại tuyến (offline)
- Học trực tuyến (online)
- Học phối hợp (online+offline)

## ➤ Phân chia theo dạng dữ liệu

- Học có giám sát (Supervised learning)
- Học không giám sát (Unsupervised learning)
- Học tăng cường (Reinforce learning)



# Học có giám sát

37

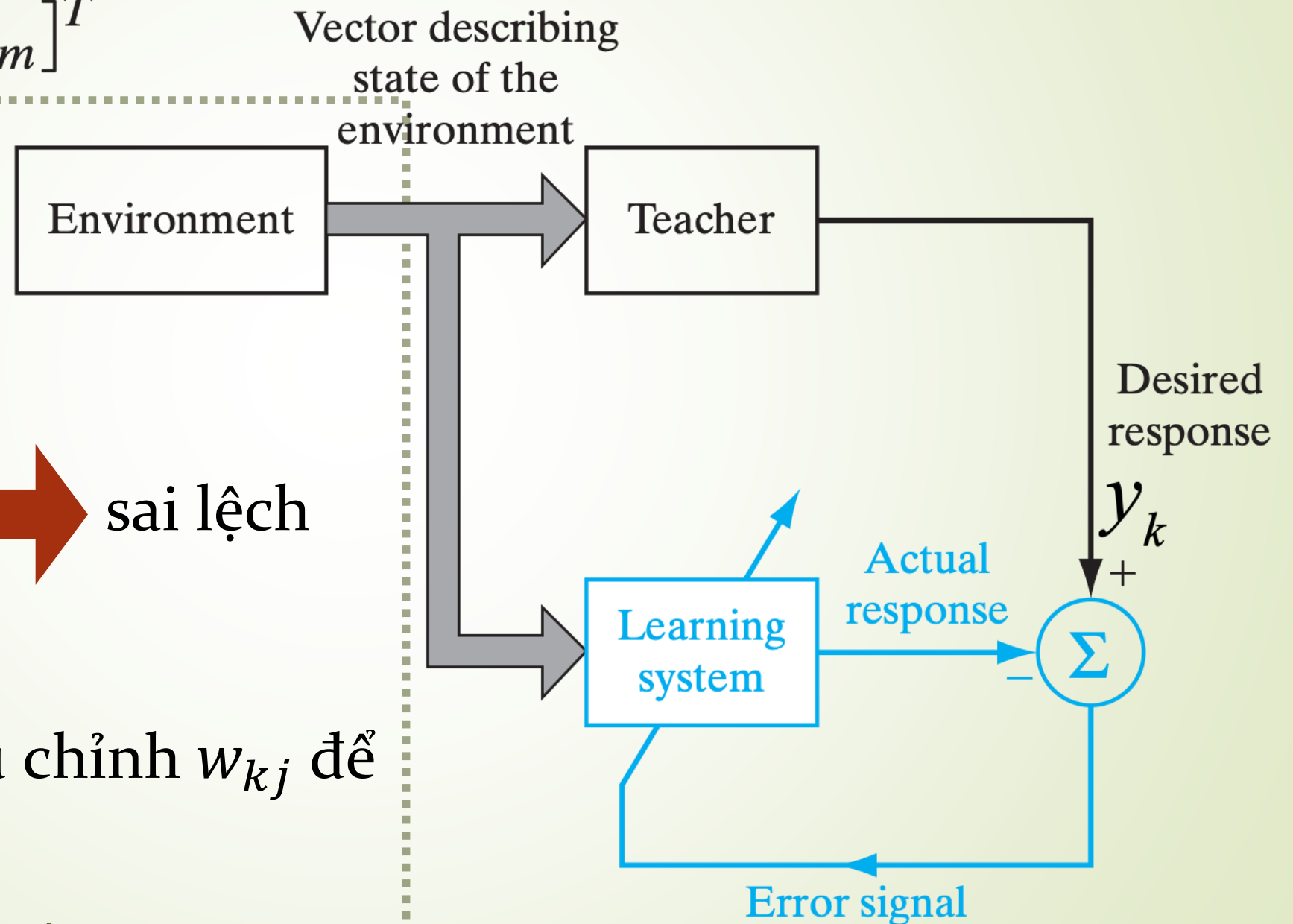
$$\mathbf{x}_i = [x_{i1}, x_{i2}, \dots, x_{im}]^T$$

$x_i$ : được đưa vào cả Teacher lẫn mạng neuron

$y_k$  được so sánh  $\rightarrow$  sai lệch

Mạng neuron tự hiệu chỉnh  $w_{kj}$  để sai lệch là nhỏ nhất

*Error-correction learning*



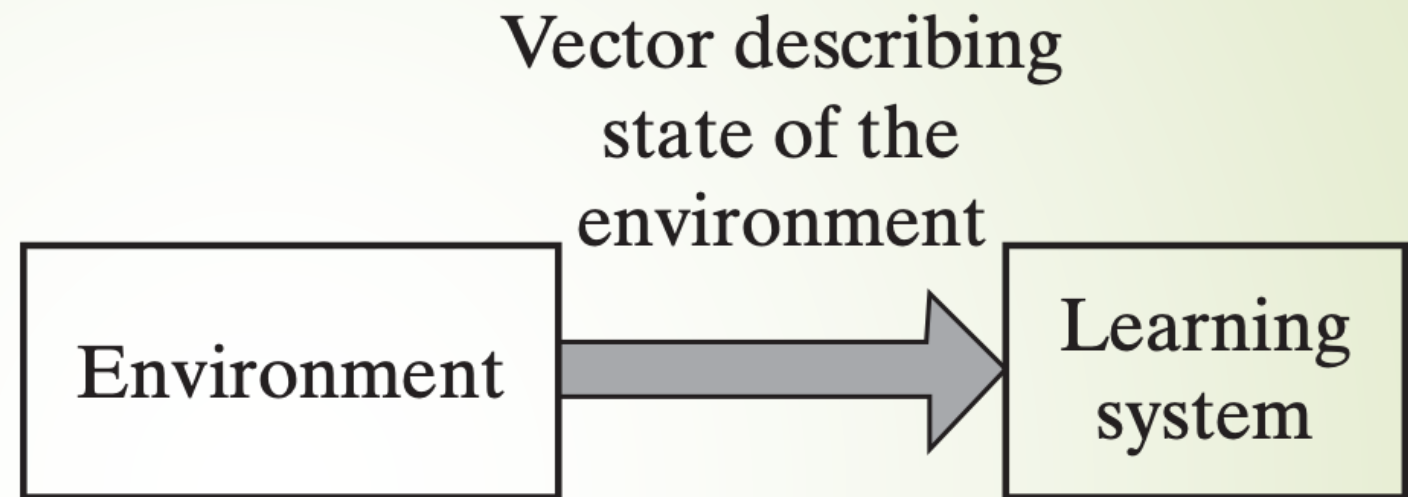
Mean square error MSE

Sum of square error SSE

# Học không giám sát

38

Tri thức được đưa  
trực tiếp vào mạng



# Ý nghĩa, mục đích việc học

39

## Lưu trữ tri thức vào mạng neuron

Một lượng tri thức được mô tả bởi  $q$  bài học, mỗi bài học có đầu vào  $\mathbf{x}$  và đầu ra  $\mathbf{y}$

$$\mathbf{x}_k \rightarrow \mathbf{y}_k, \quad k = 1, 2, \dots, q$$

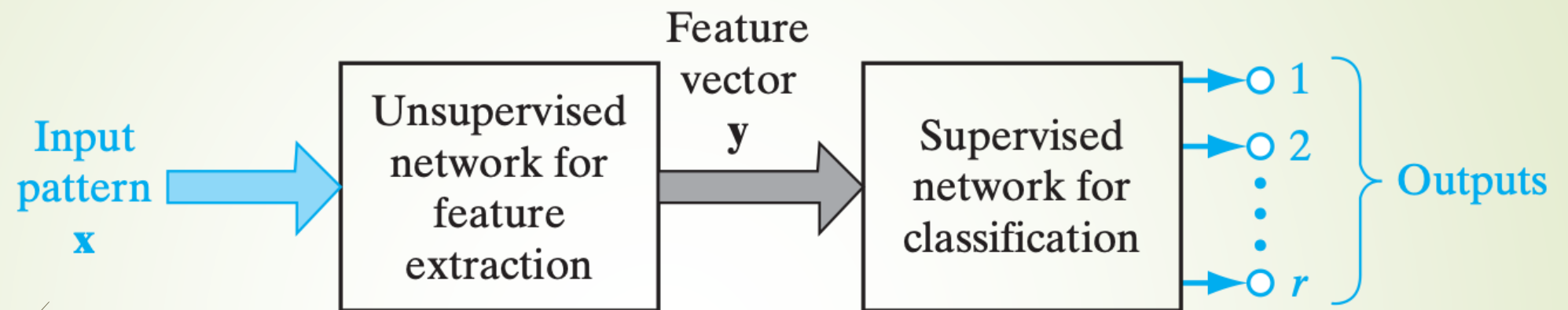
- Pha lưu tri thức: quá trình huấn luyện mạng neuron
- Pha sử dụng tri thức

Khi đưa  $\mathbf{x}$  vào mạng thì hệ thống dựa vào tri thức được huấn luyện để xác định giá trị  $\mathbf{y}$  ra tương ứng

# Ý nghĩa, mục đích việc học

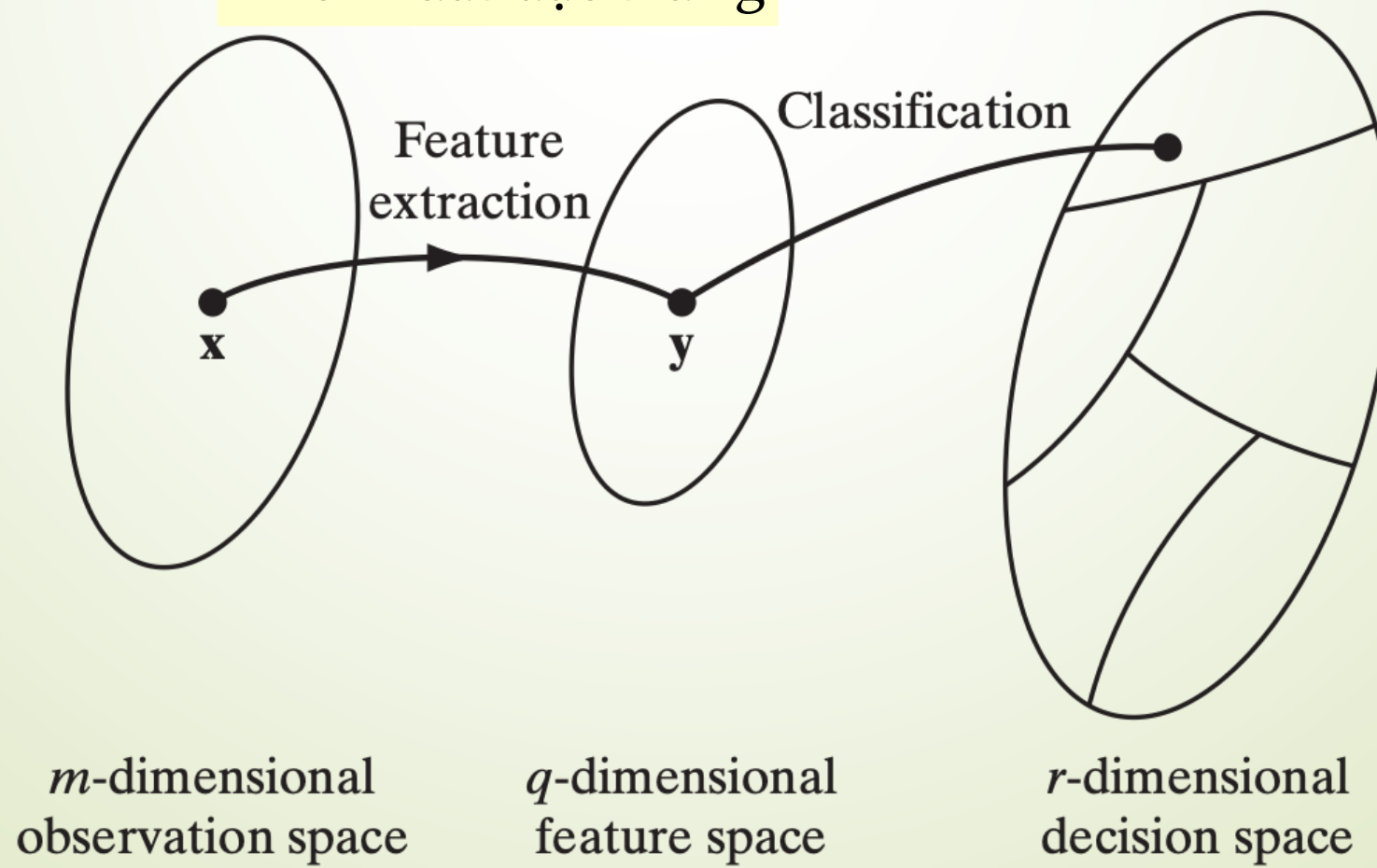
40

## Nhận dạng và phân loại



Step 1:  
Trích xuất đặc trưng

Step 2: Phân loại



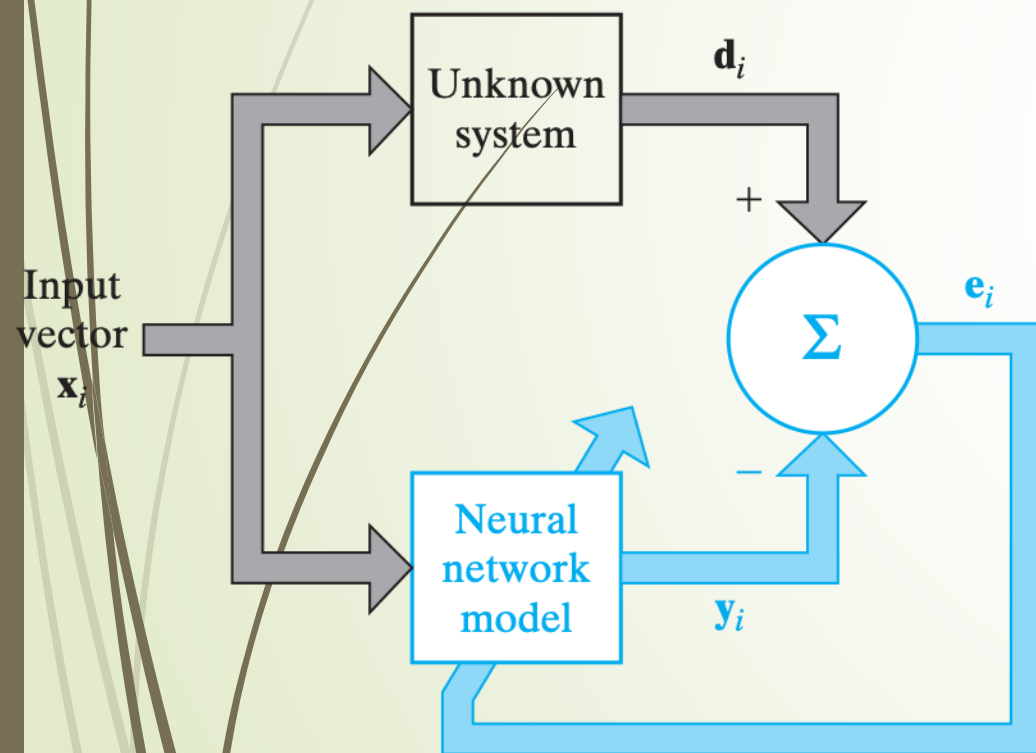


# Ý nghĩa, mục đích việc học

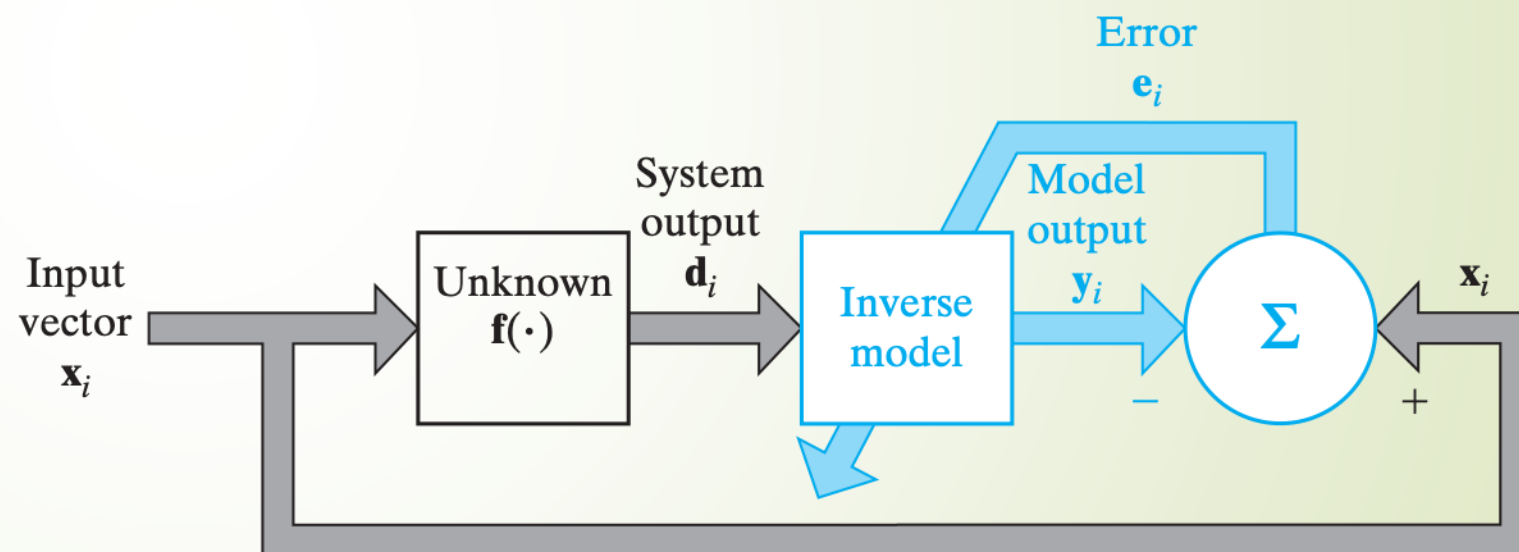
41

## Nhận dạng hàm - hệ thống

Nhận dạng hệ thống



Nhận dạng hệ thống nghịch đảo



# Ý nghĩa, mục đích việc học

## Điều khiển

Nhận dạng hệ thống

