

KHOA CNTT & TRUYỀN THÔNG
BM KHOA HỌC MÁY TÍNH

MẠNG NƠ-RON NHÂN TẠO ARTIFICIAL NEURAL NETWORK

Giáo viên giảng dạy:
TS. TRẦN NGUYỄN MINH THU
tnmthu@cit.ctu.edu.vn

2

2

Giới thiệu

Mạng nơ-ron nhân tạo (Artificial neural network - ANN)

- Mô hình hoá hoạt động của hệ thần kinh con người
- Được nghiên cứu lần đầu vào năm 1943 (McCulloch và Pitts, 1943)
- Perceptron: thể hệ đầu tiên của mạng nơ-ron (Rosenblatt, 1958)
- Mô phỏng quá trình hoạt động của thị giác con người

3

3

Giới thiệu

Lịch sử

- 1943, McCulloch & Pitts đưa ra mô hình nơ-ron đầu tiên
- 1982, Mô hình mạng nơ-ron hồi tiếp của Hopfield
- 1984, Mạng nơ-ron Kohonen hay còn gọi là Bản đồ tự tổ chức (SOM)
- 1985, Mạng nơ-ron đa tầng (MLP)

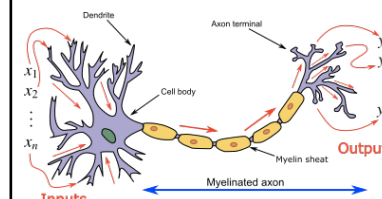
Mô hình mạng nơ-ron khác

– Mạng nơ-ron tích chập (Convolutional NN) của Yan LeCun.

4

4

Giới thiệu nơ ron sinh học



- Thân nơ-ron với nhân bên trong (gọi là soma), là nơi tiếp nhận hay phát ra các xung động thần kinh.
- Một hệ thống dạng cây các dây thần kinh vào (gọi là dendrite) để đưa tín hiệu tới nhân nơ-ron
- Đầu dây thần kinh ra (gọi là sợi trục axon) phân nhánh dạng hình cây. Chúng nối với các dây thần kinh vào hoặc trực tiếp với nhân tế bào của các nơ-ron khác thông qua các khớp nối (gọi là synapse. Có hai loại khớp nối, khớp nối kích thích (excitatory) sẽ cho tín hiệu qua nó để tới nơ-ron còn khớp nối ức chế (inhibitory) có tác dụng làm cản tín hiệu tới nơ-ron.

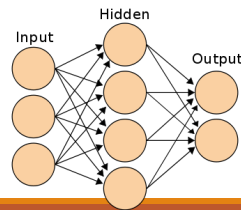
5

5

Giới thiệu

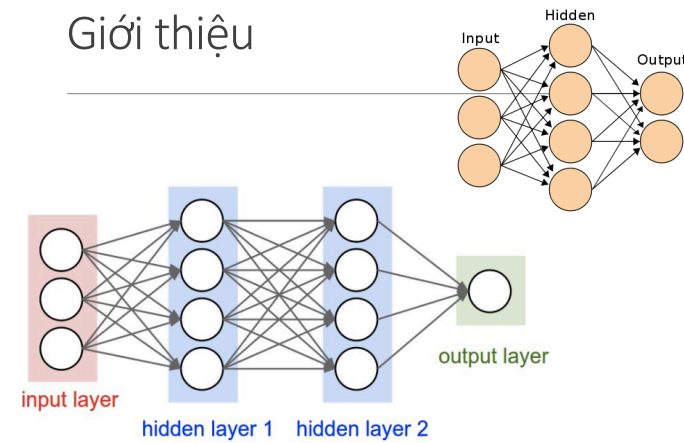
Mạng nơ-ron nhân tạo hay thường gọi ngắn gọn là **mạng nơ-ron** là một **mô hình toán học** hay **mô hình tính toán** được xây dựng dựa trên các **mạng nơ-ron sinh học**. Nó gồm có một nhóm các **nơ-ron nhân tạo** (nút) nối với nhau, và xử lý thông tin bằng cách truyền theo các kết nối và tính giá trị mới tại các nút (cách tiếp cận connectionism đối với **tính toán**).

Trong nhiều trường hợp, mạng nơ-ron nhân tạo là một **hệ thống thích ứng** (*adaptive system*) tự thay đổi cấu trúc của mình dựa trên các thông tin bên ngoài hay bên trong chảy qua mạng trong quá trình học.



6

Giới thiệu

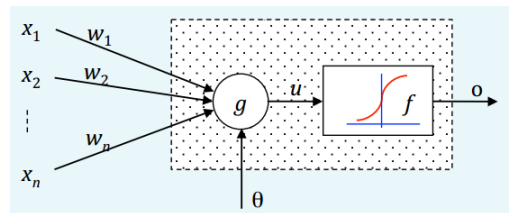


7

Mô hình nơ-ron by Warren McCulloch & Walter Pitts

Nơ-ron: đơn vị tính toán cơ bản/đơn vị của tất cả các mạng nơ-ron:

- n đầu vào, 1 tham số, 1 đầu ra
- Hàm mạng/hàm kết hợp
- Hàm kích hoạt/hàm truyền



w_i : trọng số

θ : ngưỡng (threshold), độ lệch (bias)

8

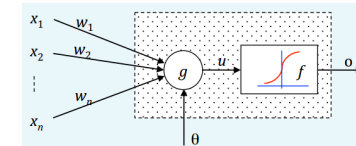
Mô hình nơ-ron McCulloch & Pitts

Nơ-ron: đơn vị tính toán cơ bản/đơn vị của tất cả các mạng nơ-ron:

- n đầu vào, 1 tham số, 1 đầu ra
- Hàm mạng/hàm kết hợp

- Hàm kích hoạt/hàm truyền

$$u = g(x) = \theta + \sum_{i=1}^n w_i x_i$$



w_i : trọng số

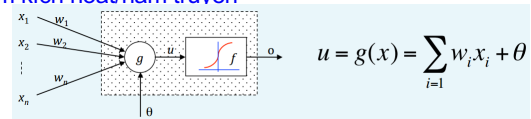
θ : ngưỡng (threshold), độ lệch (bias)

9

Mô hình nơ-ron McCulloch & Pitts

Nơ-ron: đơn vị tính toán cơ bản/đơn vị của tất cả các mạng nơ-ron:

- n đầu vào, 1 tham số, 1 đầu ra
- Hàm mạng/hàm kết hợp
- Hàm kích hoạt/hàm truyền



Đầu ra

$$o = f(u) = f(g(x))$$

$$f(u) = \frac{1}{1 + e^{-u/T}} \quad f(u)[1 - f(u)]/T$$

10

10

Hàm kích hoạt/ hàm truyền

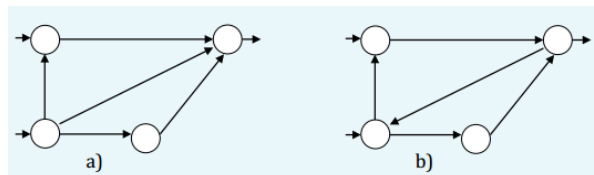
Activation function	Equation	Example	1D Graph
Unit step (Heaviside)	$\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Sign (Signum)	$\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Linear	$\phi(z) = z$	Adaline, linear regression	
Piece-wise linear	$\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$	Support vector machine	
Logistic (sigmoid)	$\phi(z) = \frac{1}{1 + e^{-z}}$	Logistic regression, Multi-layer NN	
Hyperbolic tangent	$\phi(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$	Multi-layer NN	

11

11

Kiến trúc mạng

- Mạng truyền thẳng (forward)
- Mạng hồi tiếp (feedback)



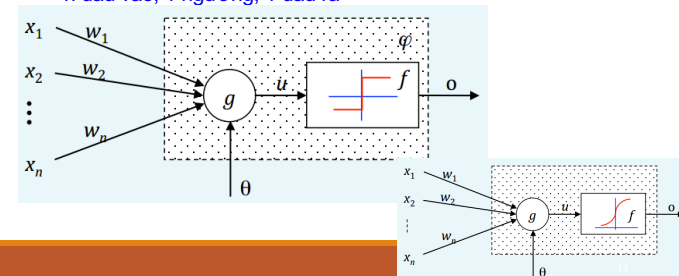
12

12

Mô hình perceptron

Mô hình perceptron

- Do Rosenblatt đề xuất năm 1958
- Tương tự như mô hình nơ-ron của McCulloch&Pitts
- Perceptron tuyến tính có ngưỡng
- n đầu vào, 1 ngưỡng, 1 đầu ra



13

Mô hình perceptron

Perceptron tuyến tính có ngưỡng

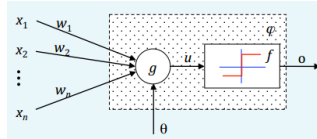
- n đầu vào, **1 ngưỡng**, 1 đầu ra
- Hàm mạng tuyến tính

$$u = g(x) = \theta + \sum_{i=1}^n w_i x_i$$

- Hàm kích hoạt là hàm ngưỡng

$$o = f(u) = f(g(x)) = \begin{cases} 1 & g(x) \geq 0 \\ 0 & g(x) < 0 \end{cases}$$

$$o = f(u) = f(g(x)) = \begin{cases} 1 & g(x) \geq 0 \\ -1 & g(x) < 0 \end{cases}$$

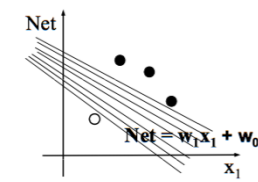
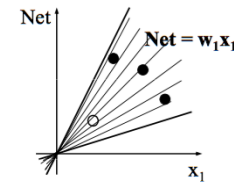


14

14

Ý nghĩa của giá trị theta

$$u = g(x) = \theta + \sum_{i=1}^n w_i x_i$$



15

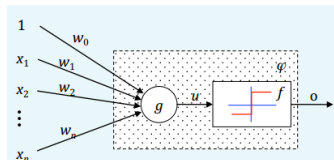
15

Mô hình perceptron

Perceptron tuyến tính không ngưỡng

- $n+1$ đầu vào, 1 đầu ra
- Đầu vào giả x_0 luôn có giá trị 1, $w_0 = \theta$
- Hàm mạng tuyến tính

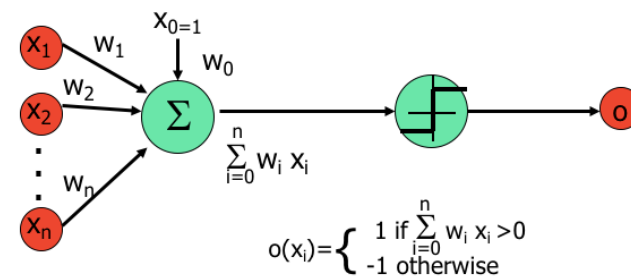
$$u = g(x) = \sum_{i=0}^n w_i x_i$$



16

16

Mô hình perceptron



$$o(x_i) = \begin{cases} 1 & \text{if } \sum_{i=0}^n w_i x_i > 0 \\ -1 & \text{otherwise} \end{cases}$$

17

17

Huấn luyện perceptron

Huấn luyện - Dạy cho perceptron

- **Tìm kiếm n tham số:** $w_0, w_1, w_2, \dots, w_n$ sao cho đầu ra của nơ-ron phù hợp với giá trị mong muốn của tất cả dữ liệu học nhất.

Dữ liệu đầu vào:

- Tập các mẫu huấn luyện
- Mỗi mẫu huấn luyện gồm: véc-tơ đặc trưng x và nhãn y .

Tham số:

- Tốc độ học: η (đọc là eta)

Về mặt hình học:

- Tìm siêu phẳng tách dữ liệu thành 2 lớp sao cho mỗi lớp về 1 phía của siêu phẳng này.

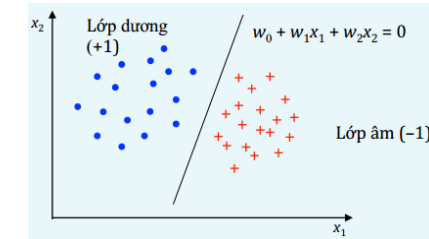
18

Mô hình perceptron

Ý nghĩa hình học

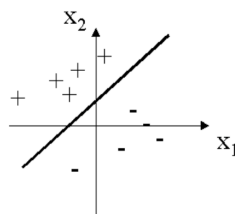
- Phương trình $u = g(x) = 0$ là phương trình của 1 siêu phẳng trong không gian n chiều.

$$u = g(x) = \sum_{i=0}^n w_i x_i = 0$$

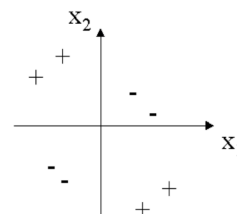


19

Dữ liệu khả tách tuyến tính?



Linearly Separable



Not Linearly Separable

20

20

Huấn luyện perceptron

Trường hợp dữ liệu khả tách tuyến tính

- Khởi tạo ngẫu nhiên các w
- Đưa từng mẫu học qua perceptron và quan sát giá trị đầu ra
- Nếu giá trị đầu ra khác với giá trị mong muốn, cập nhật lại các trọng số theo công thức:

$$w_j = w_j + \eta \cdot (y_i - o_i) \cdot x_{ij}, \forall j = 0..n$$

- Nếu giá trị **output bằng giá trị mong muốn**: trọng số không thay đổi do $y - o = 0$
- Nếu giá trị **output nhỏ hơn giá trị mong muốn**: các trọng số sẽ được tăng một lượng tỉ lệ thuận với thành phần x_i của vectơ đặc trưng đang xét
- Nếu giá trị **output lớn hơn giá trị mong muốn**: các trọng số sẽ giảm đi một lượng tỉ lệ thuận với đầu vào

21

21

Bài tập

x1	x2	Y
0	0	0
0	1	0
1	0	0
1	1	1

Thiết kế perceptron cho dữ liệu trong bảng
 Với các trọng số $w_0 = -0.2$, $w_1 = 0.5$, $w_2 = 0.5$
 Tốc độ học: $\eta = 0.15$

22

Bài tập

x1	x2	Y
0	0	0
0	1	0
1	0	0
1	1	1

Đầu vào: x1, x2

Đầu ra: y – có 2 giá trị 0-1,
 sử dụng hàm kích hoạt ngưỡng {0,1}
 Hàm mạng

$$\varphi(x_i) = g(x_i) = \sum_{j=0}^n w_j \cdot x_{ij}$$

Thiết kế perceptron cho dữ liệu trong bảng
 Hàm kích hoạt hay hàm ngưỡng

Với các trọng số $w_0 = -0.2$,
 $w_1 = 0.5$, $w_2 = 0.5$
 Tốc độ học: $\eta = 0.15$

$$o = f(u) = f(g(x)) = \begin{cases} 1 & g(x) \geq 0 \\ 0 & g(x) < 0 \end{cases}$$

23

Bài tập

x1	x2	Y
0	0	0
0	1	0
1	0	0
1	1	1

Đầu vào: x1, x2

Đầu ra: y – có 2 giá trị 0-1,
 sử dụng hàm kích hoạt ngưỡng {0,1}

Xác định dữ liệu có khả năng tách tuyến tính?

Cập nhật lại các giá trị w khi giá trị đầu ra khác
 với giá trị mong muốn

$$w_j = w_j + \eta \cdot (y_i - o_i) \cdot x_{ij}, \forall j = 0..n$$

24

24

Bài tập

x1	x2	Y
0	0	0
0	1	0
1	0	0
1	1	1

Lần lặp 1:

Xáo trộn thứ tự tập huấn luyện: 1,3,4,2

Xét phần tử 1 có vector đặc trưng (0,0) –
 $y_1 = 0$

Hàm mạng

$$\varphi(x_i) = g(x_i) = \sum_{j=0}^n w_j \cdot x_{ij}$$

$$U_1 = w_0 + w_1 \cdot x_1 + w_2 \cdot x_2 = -0.2 + 0.5 \cdot 0 + 0.5 \cdot 0 = -0.2$$

Do $U_1 < 0 \Rightarrow \text{theta} = 0$

Giống với đầu ra thực tế \Rightarrow giữ nguyên

25

25

Bài tập

x1	x2	Y
0	0	0
0	1	0
1	0	0
1	1	1

Lần lặp 1:

Xáo trộn thứ tự tập huấn luyện: 1,3,4,2

Xét phần tử 3 có vector đặc trưng (1,0) – y3=0

Hàm mạng $\varphi(x_i) = g(x_i) = \sum_{j=0}^n w_j \cdot x_{ij}$

$$U_1 = w_0 + w_1 \cdot x_1 + w_2 \cdot x_2 = -0.2 + 0.5 \cdot 1 + 0.5 \cdot 0 = 0.3$$

Do $U_1 > 0 \Rightarrow \text{theta} = 1$

Khác với đầu ra thực tế (y3=0) \Rightarrow cập nhật lại trọng số

$$w_0 = w_0 + 0.15 \cdot (0-1) \cdot 1 = -0.2 - 0.15 = -0.35$$

$$w_1 = w_1 + 0.15 \cdot (0-1) \cdot 1 = 0.5 - 0.15 = 0.35$$

$$w_2 = w_2 + 0.15 \cdot (0-1) \cdot 0 = 0.5 + 0 = 0.5$$

$$w_j = w_j + \eta \cdot (y_i - o_i) \cdot x_{ij}, \forall j = 0..n$$

26

Huấn luyện perceptron

$$E(w) = E(w_0, w_1, \dots, w_n) = \frac{1}{2} \sum_{i=1}^m (y^{(i)} - g(x^{(i)}))^2$$

Trường hợp dữ liệu **không** khả tách tuyến tính

- Cố gắng tìm một siêu phẳng “tốt” nhất
- **Tốt = lỗi (trên tập học) nhỏ nhất có thể**
- Định nghĩa hàm **lỗi E(w)** theo các trọng số **w** trên tất cả các phần tử của tập học:

$$E(w) \equiv E(w_0, w_1, \dots, w_n) = \frac{1}{2} \sum_{i=1}^m (y_i - \varphi(x_i))^2$$

- **Bài toán huấn luyện trở thành tìm w sao cho E(w) nhỏ nhất**

27

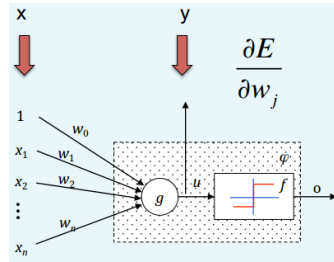
Huấn luyện perceptron

Chú ý:

- Hàm kích hoạt trong trường hợp này được thay bằng hàm tuyến tính $f(u) = u$ hay:

$$\varphi(x_i) = g(x_i) = \sum_{j=0}^n w_j \cdot x_{ij}$$

Ta tối ưu E(w) bằng phương pháp gradient descent



$$o = f(u) = f(g(x)) = \begin{cases} 1 & g(x) \geq 0 \\ 0 & g(x) < 0 \end{cases}$$

28

28

Huấn luyện perceptron

Ta tối ưu E(w) bằng phương pháp gradient descent

Giải thuật:

- Khởi động ngẫu nhiên w
- Lặp cho đến khi điều kiện dừng thỏa mãn:
- **$w = w - \eta \Delta E(w)$**
 $= w + \Delta w$ (Với $\Delta w = -\eta \Delta E(w)$)

Gradient của E(w):

$$\frac{\partial E}{\partial w_j} = - \sum_{i=1}^m (y_i - g(x_i)) \cdot x_{ij}$$

$$\nabla E(w) = \begin{pmatrix} \frac{\partial E}{\partial w_0} \\ \frac{\partial E}{\partial w_1} \\ \vdots \\ \frac{\partial E}{\partial w_n} \end{pmatrix}$$

29

29

Chương 2. Mạng nơ-ron nhân tạo Đỗ Thanh Nghị - Phạm Nguyễn Khang

$$\begin{aligned}\frac{\partial E}{\partial w_j} &= \frac{\partial}{\partial w_j} \left(\frac{1}{2} \sum_{i=1}^m (y_i - \phi(x_i))^2 \right) \\ \frac{\partial E}{\partial w_j} &= \frac{1}{2} \sum_{i=1}^m \left\{ \frac{\partial}{\partial w_j} (y_i - \phi(x_i))^2 \right\} \\ \frac{\partial E}{\partial w_j} &= \frac{1}{2} \sum_{i=1}^m \left\{ 2(y_i - \phi(x_i)) \cdot \frac{\partial}{\partial w_j} (y_i - \phi(x_i)) \right\} \\ \frac{\partial E}{\partial w_j} &= \frac{1}{2} \sum_{i=1}^m \left\{ 2(y_i - \phi(x_i)) \cdot \frac{\partial}{\partial w_j} (-\phi(x_i)) \right\} \\ \frac{\partial E}{\partial w_j} &= \sum_{i=1}^m \left\{ (y_i - \phi(x_i)) \cdot \frac{\partial}{\partial w_j} (-\phi(x_i)) \right\} \\ \frac{\partial E}{\partial w_j} &= \sum_{i=1}^m \left\{ (y_i - \phi(x_i)) \cdot \frac{\partial}{\partial w_j} (f(g(x_i))) \right\} \\ \frac{\partial E}{\partial w_j} &= - \sum_{i=1}^m \left\{ (y_i - \phi(x_i)) \cdot \frac{\partial f(g(x_i))}{\partial g(x_i)} \cdot \frac{\partial g(x_i)}{\partial w_j} \right\} \\ \frac{\partial E}{\partial w_j} &= - \sum_{i=1}^m \left\{ (y_i - g(x_i)) \cdot f'(g(x_i)) \cdot \frac{\partial}{\partial w_j} \left(\sum_{k=0}^n w_k \cdot x_k \right) \right\} \\ \frac{\partial E}{\partial w_j} &= - \sum_{i=1}^m \left\{ (y_i - g(x_i)) \cdot f'(g(x_i)) \cdot x_{ij} \right\}\end{aligned} \quad (2.14)$$

ong đó, f' là đạo hàm cấp 1 của f theo g .

30

Huấn luyện perceptron

$$\frac{\partial E}{\partial w_j} = - \sum_{i=1}^m \left\{ (y_i - g(x_i)) \cdot f'(g(x_i)) \cdot x_{ij} \right\}$$

trong đó, f' là đạo hàm cấp 1 của f theo g .

Từ đây ta có thể suy ra:

$$\Delta w_j = \eta \cdot \sum_{i=1}^m \left\{ (y_i - g(x_i)) \cdot f'(g(x_i)) \cdot x_{ij} \right\} \quad (2.15)$$

Vì hàm kích hoạt f là hàm tuyến tính! $f(u) = u$, ta có $f'(u) = df/du = 1$.
Lượng cập nhật cho trọng số w_j là:

$$\Delta w_j = \eta \cdot \sum_{i=1}^m \left\{ (y_i - g(x_i)) \cdot x_{ij} \right\} \quad (2.16)$$

31