

# Các hệ tri thức kết hợp

---

TS. NGUYỄN ĐÌNH HÓA

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

# Một số câu hỏi tổng hợp

---

Hệ thống nào yêu cầu người dùng cập nhật dữ liệu một cách thường xuyên?

Hệ thống nào sử dụng luật để đưa ra quyết định?

Hệ thống nào tự điều chỉnh các tham số theo dữ liệu đầu vào và dữ liệu đầu ra?

Hệ thống nào có thể liên tục mở rộng lượng thông tin chứa trong nó?

Hệ thống nào có thể mô phỏng quá trình ra quyết định của con người?

Hệ thống nào có thể học theo quá trình phân tích/xử lý với các mẫu thông tin của con người?

# Đặc điểm của mạng nơ ron

---

Mạng nơ ron chỉ hiệu quả trong các bài toán được mô tả một cách tường minh bởi con người

Mạng nơ ron sử dụng kết hợp các biểu thức toán học đơn giản để biểu diễn mối quan hệ phức tạp giữa dữ liệu đầu vào và đầu ra.

Mạng nơ ron không diễn đạt một cách rõ ràng tại sao kết luận được thực hiện trong từng trường hợp cụ thể

Mạng nơ ron yêu cầu dữ liệu huấn luyện càng nhiều càng tốt.

Mạng nơ ron có nhiều ứng dụng trong thực tế.

# Câu hỏi về mạng nơ ron

---

Giả sử chúng ta cần huấn luyện một mạng nơ ron để nhận dạng ảnh. Bộ dữ liệu gồm 150 bức ảnh khác nhau, và có 2 cách thực hiện như sau:

- Chia tập dữ liệu thành 2 phần bằng nhau. Huấn luyện mạng nơ ron bằng phần dữ liệu huấn luyện gồm 75 bức ảnh, và kiểm tra mạng nơ ron bằng phần dữ liệu gồm 75 bức ảnh còn lại.
- Sử dụng toàn bộ 150 bức ảnh để huấn luyện mạng nơ ron.

Anh/Chị hãy cho biết cách làm nào có thể tạo ra mạng nơ ron tốt nhất? Tại sao?

# Tìm hệ thống phù hợp

---

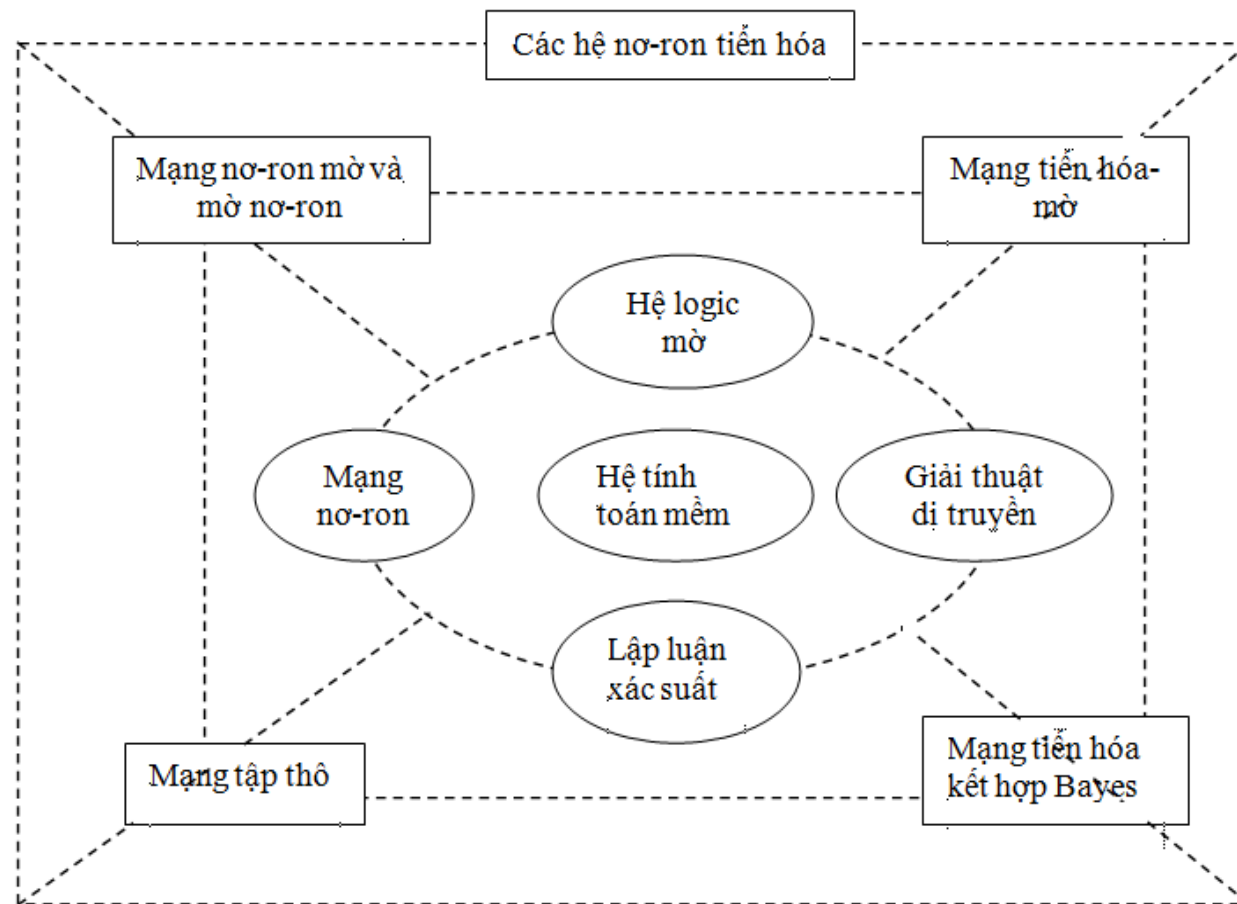
**Bài toán 1:** Một công ty sản xuất bia muốn áp dụng một hệ thống nhận diện mùi tự động để “ngửi” xem bia có mùi phù hợp hay không trước khi được đóng vào lon. Thiết bị “ngửi” này sẽ được triển khai trên hệ thống cảm biến trong dây chuyền tự động.

**Bài toán 2:** Một ngân hàng muốn triển khai một hệ thống lọc hồ sơ tự động để xác định các hồ sơ vay tiền có khả năng thanh toán hay không.

Hãy chọn hệ thống phù hợp cho từng bài toán trên:

1. Hệ thống dựa trên luật (Rule-based systems)
2. Mạng nơ ron nhân tạo
3. Hệ thống dựa trên mẫu có sẵn (Case-based systems)
4. Giải thuật di truyền.

# Các hệ thống lai



# Hệ thống lai Nơ ron – Mờ

---

So sánh giữa mạng nơ ron và logic mờ

Đặc tính	Mạng nơ ron	Logic mờ
Khả năng biểu diễn tri thức	Không tốt	Tốt
Khả năng linh hoạt	Tốt	Tốt
Khả năng xử lý thông tin không chính xác	Tốt	Tốt
Khả năng thích nghi	Tốt	Tốt
Khả năng học	Tốt	Không tốt
Khả năng diễn đạt tri thức	Không tốt	Tốt

# Đặc điểm của logic mờ và mạng nơ ron

---

Logic mờ cho phép tạo ra các quyết định dựa trên dữ liệu và thông tin không tường minh.

Mạng nơ ron nhân tạo mô phỏng quá trình hoạt động của não người để giải quyết bài toán mà không cần dựa trên một mô hình toán học cụ thể nào

Cả logic mờ và mạng nơ ron nhân tạo đều có thể giải quyết các bài toán phi tuyến, và các bài toán không cần định nghĩa rõ ràng.

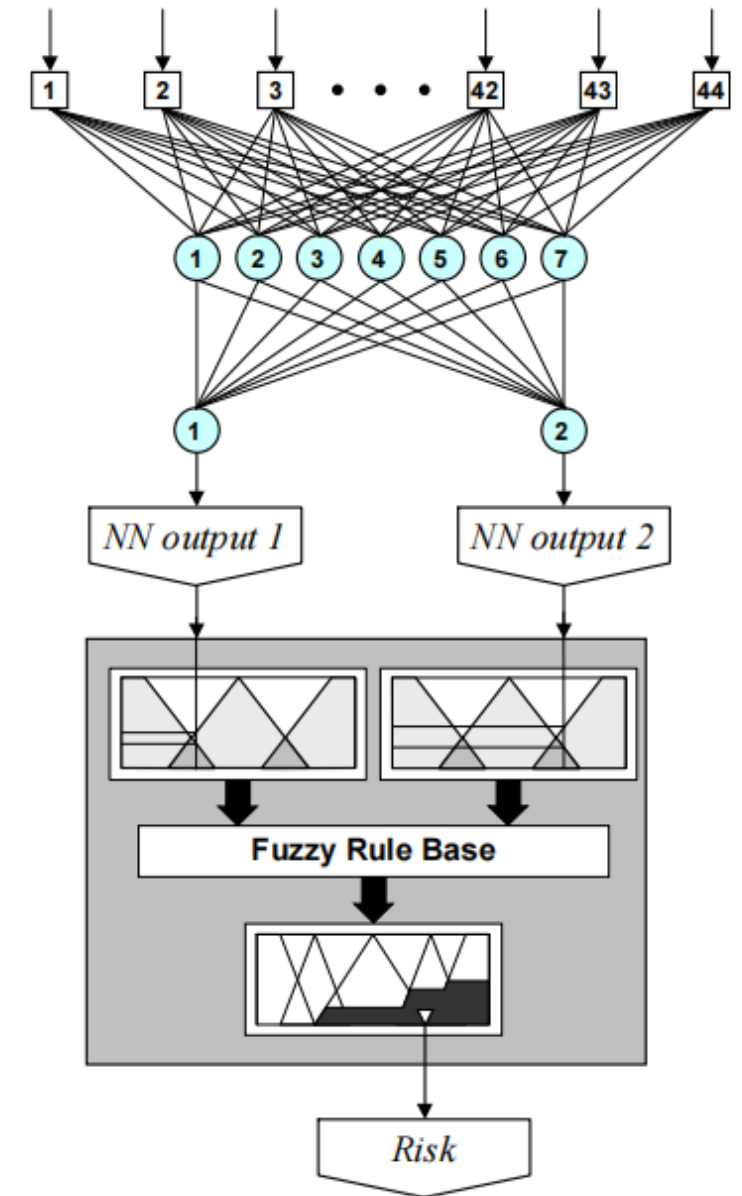
Sự kết hợp giữa mạng nơ ron và logic mờ về bản chất là sự kết hợp giữa kiến thức chuyên gia (các luật **if-then**) với các mô hình tính toán sinh học trong não người để tạo ra một hệ thống mạnh hơn.



# Hệ thống lai Nơ ron – mờ

Là sự kết hợp giữa mạng nơ ron và hệ mờ nhằm tận dụng các ưu điểm của cả hai loại hệ thống này.

1. Mạng nơ ron: có thể nhận diện các mẫu dữ liệu, và tự điều chỉnh các thông số của nó để thích ứng với các thông tin huấn luyện
2. Hệ mờ: cho phép kết hợp tri thức con người với các luật suy diễn để đưa ra các quyết định.



# Hệ nơ ron – mờ

---

Lĩnh vực ứng dụng: trong các bài toán thiên nhiều về tính toán như xử lý tín hiệu thích ứng, điều khiển thích ứng, nhận diện hệ thống phi tuyến,... Ứng dụng trong các hệ thống lớn, độ bất định cao.

Tạo ra hệ thống mà không cần các mô hình huấn luyện. Hệ thống được xây dựng dựa trên ứng dụng mục tiêu: không coi trọng quá trình tạo ra kết quả, mà chỉ quan tâm đến kết quả đạt được có đúng yêu cầu hay không.

Hệ thống mạnh về tính toán.

Hệ thống có khả năng chống lỗi tốt: hệ thống không bị hỏng nếu một vài nơ ron hoặc một vài luật bị loại bỏ mà hiệu quả của nó chỉ bị giảm đi.

# Hệ thống nơ ron – mờ

---

Là mạng nơ ron trong đó mỗi nơ ron biểu diễn một biến ngôn ngữ (logic mờ). Toàn bộ mạng nơ ron có chức năng giống một mô hình suy diễn mờ

Mạng nơ ron được huấn luyện dựa trên các luật mờ **If - Then**.

Đầu ra của mỗi nơ ron là giá trị thành viên của đầu vào tương ứng với một hàm thành viên nào đó.

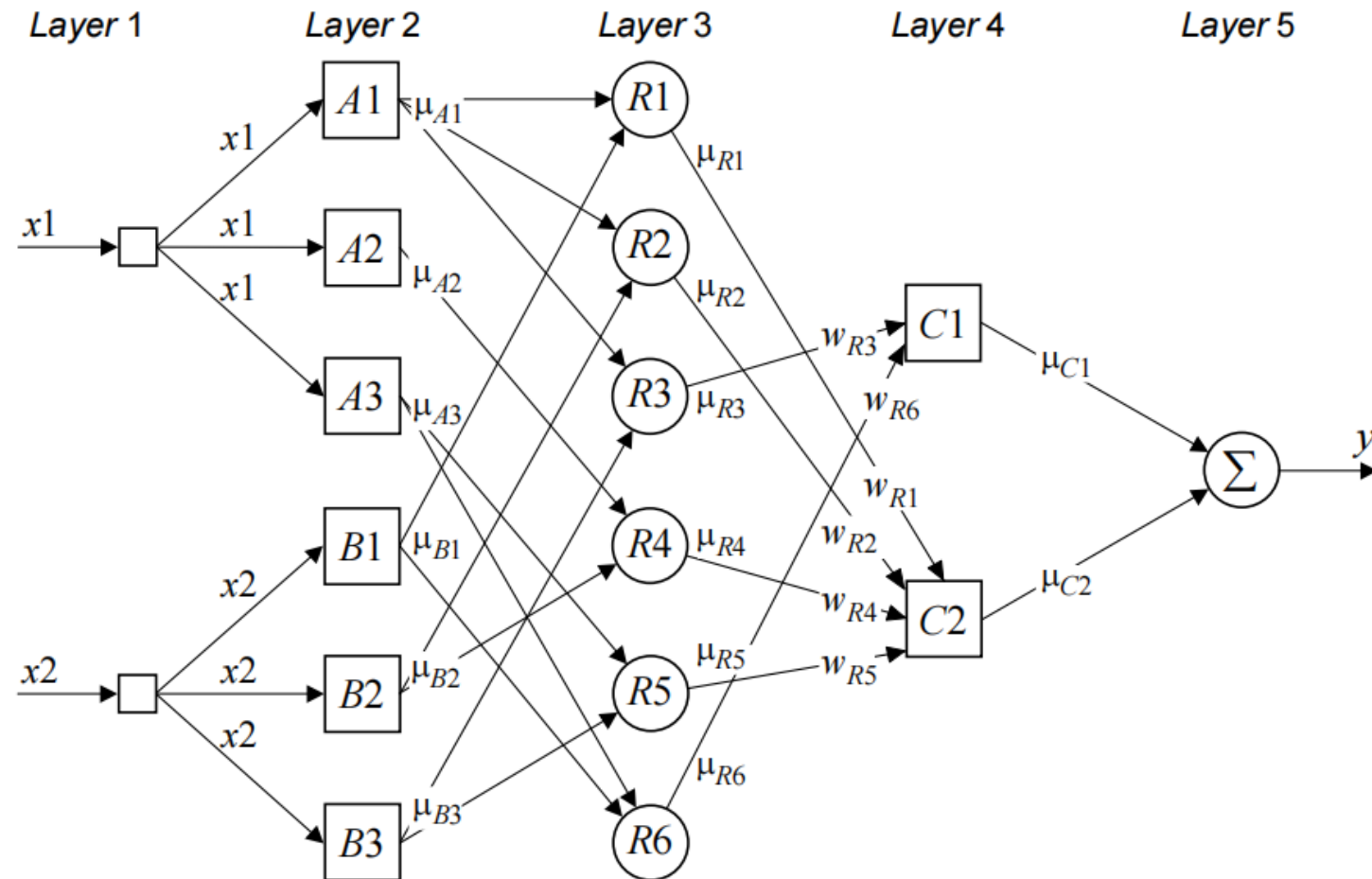
Các tri thức chuyên gia có thể được dễ dàng tích hợp vào hệ thống nơ ron – mờ này

Đầu ra của lớp nơ ron này sẽ là đầu vào của lớp tiếp theo. Điều này giúp tránh được việc sử dụng cơ chế suy diễn mờ.

Cấu trúc hệ thống giống như một mạng nơ ron nhiều lớp.

Hệ thống gồm 1 lớp đầu vào, 1 lớp đầu ra, và 3 lớp ẩn.

# Hệ nơ ron – mờ



# Hệ nơ ron – mờ

---

Lớp 1: là lớp đầu vào

Lớp này truyền trực tiếp giá trị đầu vào đến lớp tiếp theo

$$y_i^{(1)} = x_i^{(1)}$$

Lớp 2: là lớp mờ hóa.

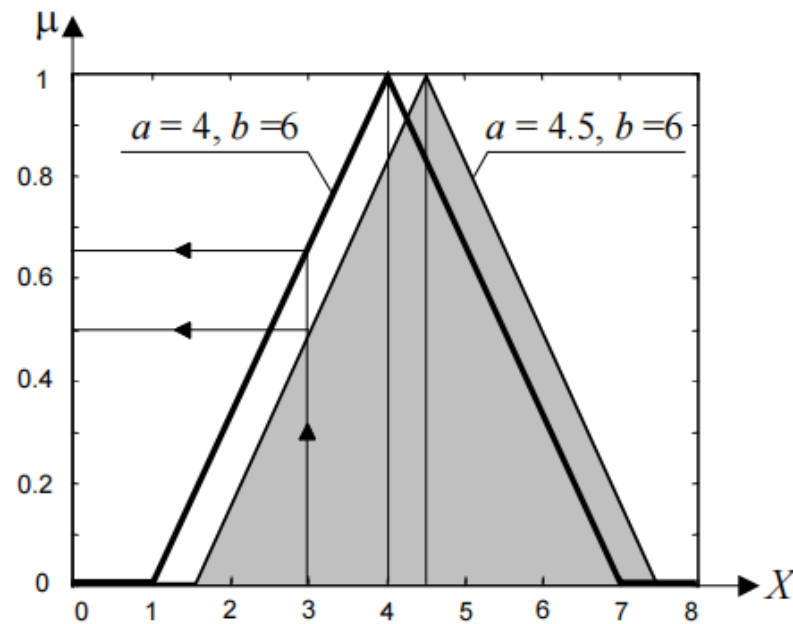
Các nơ ron của lớp này biểu diễn các tập mờ trong luật mờ. Đầu ra của các nơ ron là giá trị hàm thành viên của dữ liệu đầu vào tương ứng với từng tập mờ.

Hàm truyền của mỗi nơ ron là hàm thành viên của một tập mờ nào đó. Hàm thành viên thường được chọn là hàm tam giác.

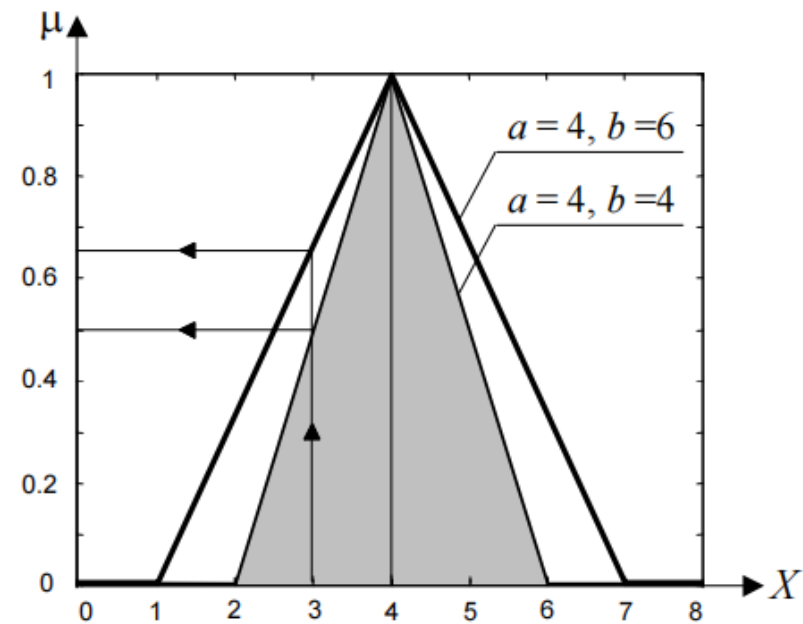
$$y_i^{(2)} = \begin{cases} 0, & \text{if } x_i^{(2)} \leq a - \frac{b}{2} \\ 1 - \frac{2|x_i^{(2)} - a|}{b}, & \text{if } a - \frac{b}{2} < x_i^{(2)} < a + \frac{b}{2} \\ 0, & \text{if } x_i^{(2)} \geq a + \frac{b}{2} \end{cases}$$

# Hệ nơ ron – mờ

Tham số của mỗi nơ ron thuộc lớp 2 là  $a$  và  $b$ . Quá trình huấn luyện nơ ron cũng là xác định các giá trị tham số  $a$  và  $b$ .



(a) Effect of parameter  $a$ .



(b) Effect of parameter  $b$ .

# Hệ nơ ron – mờ

---

Lớp 3: lớp luật mờ (thay thế cho suy diễn mờ trong Hệ mờ).

Mỗi nơ ron thuộc lớp này tương ứng với một luật mờ nào đó là sự kết hợp thông tin từ nhiều tập mờ khác nhau.

Đầu vào của mỗi nơ ron là giá trị thành viên của nhiều tập mờ. Các thông tin đầu vào này có thể được kết hợp với nhau bằng toán tử nhân.

$$y_i^{(3)} = x_{1i}^{(3)} \times x_{2i}^{(3)} \times \dots \times x_{ki}^{(3)}$$

Giả sử mơ ron R1 tương ứng với luật 1, nhận đầu vào từ nơ ron A1 và B1

$$y_{R1}^{(3)} = \mu_{A1} \times \mu_{B1} = \mu_{R1}$$

# Hệ nơ ron – mờ

---

Lớp 4: là lớp quyết định tập thành viên của dữ liệu đầu vào.

Các nơ ron trong lớp này biểu diễn tập luật mờ.

Mỗi nơ ron sẽ kết hợp toàn bộ các thông tin đầu vào bằng toán tử Hợp (Union), hoặc toán tử OR

$$y_i^{(4)} = x_{1i}^{(4)} \oplus x_{2i}^{(4)} \oplus \dots \oplus x_{li}^{(4)}$$

Ví dụ:  $y_{C1}^{(4)} = \mu_{R3} \oplus \mu_{R6} = \mu_{C1}$

Giá trị  $\mu_{C1}$  biểu diễn sự kết hợp các luật từ nơ ron R3 và R6



# Hệ nơ ron – mờ

---

Lớp 5: là lớp giải mờ

Mỗi nơ ron trong lớp này biểu diễn một kết quả đầu ra của hệ nơ ron – mờ.

Mỗi nơ ron sử dụng các tập luật mờ được biểu diễn từ sự kết hợp các giá trị đầu ra của các luật và kết hợp chúng để tạo ra một kết quả là một tập mờ nào đó.

$$y = \frac{\mu_{C1} \times a_{C1} \times b_{C1} + \mu_{C2} \times a_{C2} \times b_{C2}}{\mu_{C1} \times b_{C1} + \mu_{C2} \times b_{C2}}$$

Về cơ bản, kết quả đầu ra là trung bình có trọng số của các trọng tâm các hàm thành viên.

# Hệ nơ ron – mờ

---

Hệ nơ ron – mờ có thể được huấn luyện bằng các thuật toán giống như ở mạng nơ ron, kể cả việc sử dụng thuật toán lan truyền ngược.

Dữ liệu được cho vào đầu vào, hệ thống sẽ tính ra kết quả đầu ra. Sai số đầu ra sẽ được lan truyền ngược từ lớp đầu ra về lớp đầu vào. Quá trình lan truyền này sẽ làm thay đổi các tham số của hàm truyền trong mỗi nơ ron dựa trên đạo hàm của các hàm truyền đó.

# Hệ nơ ron – mờ

---

Việc kết hợp mạng nơ ron với hệ mờ giúp tạo ra một hệ thống mạnh, cho phép biểu diễn tri thức dễ dàng hơn.

Các tri thức chuyên gia có thể được nạp vào hệ nơ ron - mờ dưới dạng các biến ngôn ngữ và các luật mờ.

Hệ nơ ron – mờ giúp chuyển hóa các tri thức chuyên gia thành một tập các luật If...Then...

# ANFIS (Adaptive Neuro Fuzzy Inference System)

---

Là một hệ nơ ron mờ biểu diễn mô hình mờ Sugeno

“If  $x$  is  $A$  and  $y$  is  $B$  then  $z = f(x,y)$ ”

Trong đó  $A$  và  $B$  là các tập mờ cho trước.

$z = f(x,y)$  là hàm kết hợp, thường ở dạng đa thức, biểu diễn sự kết hợp giữa  $x$  và  $y$ .

# ANFIS

---

Ví dụ về mô hình mờ Sugeno order 1

If  $x$  is **small** then  $y = 0.1x + 6.4$

If  $x$  is **median** then  $y = -0.5x + 4$

If  $x$  is **large** then  $y = x - 2$

Ví dụ về mô hình mờ Sugeno order 2

If  $x$  is **small** and  $y$  is **small** then  $z = -x + y + 1$

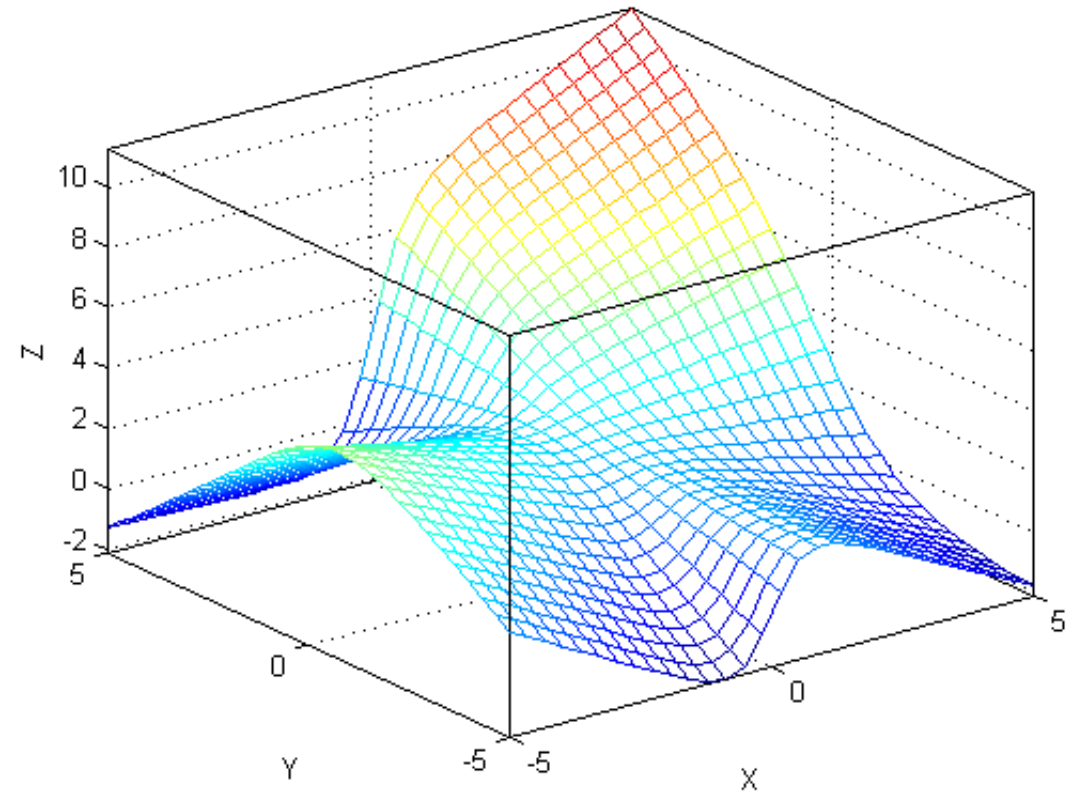
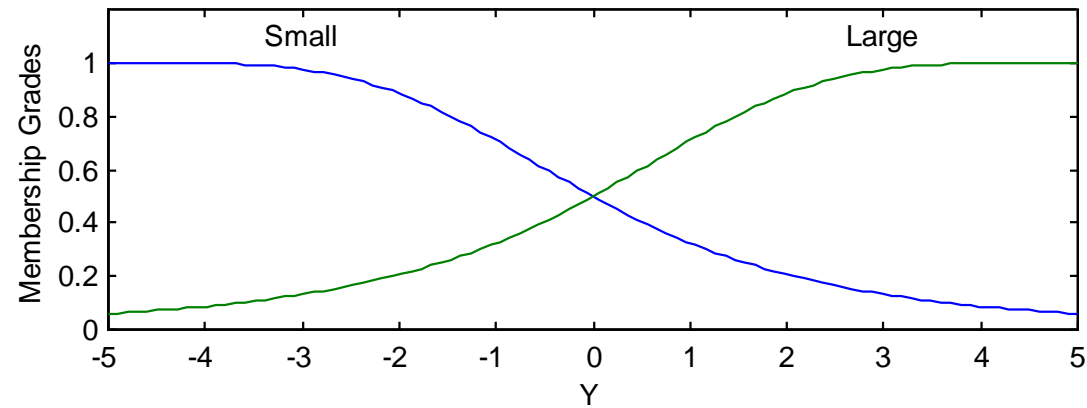
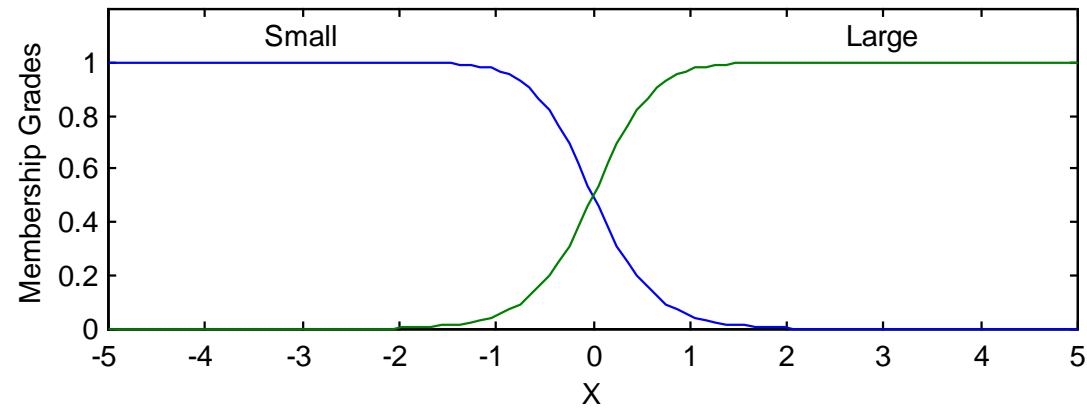
If  $x$  is **small** and  $y$  is **large** then  $z = -y + 3$

If  $x$  is **large** and  $y$  is **small** then  $z = -x + 3$

If  $x$  is **large** and  $y$  is **large** then  $z = x + y + 2$

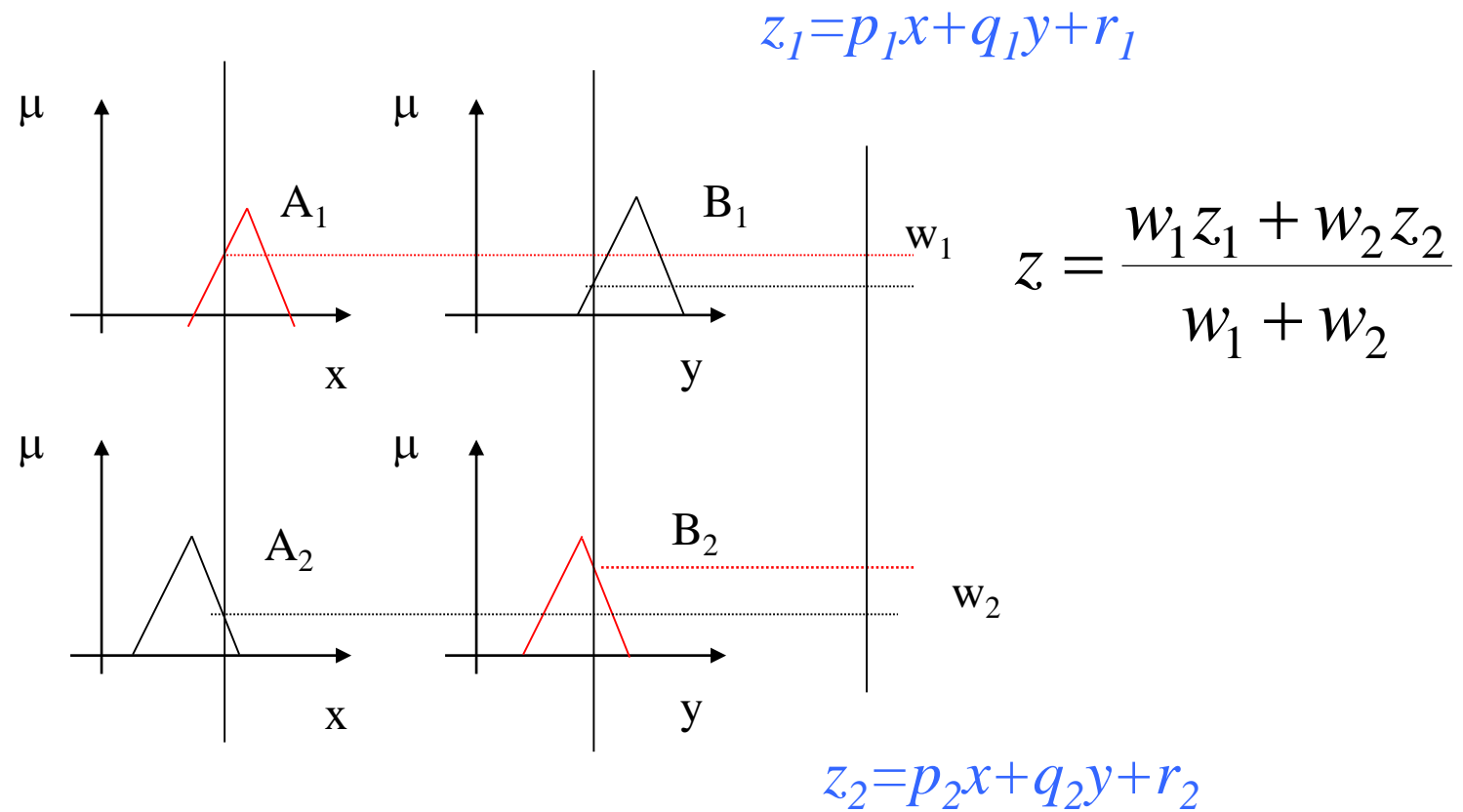
# ANFIS

---



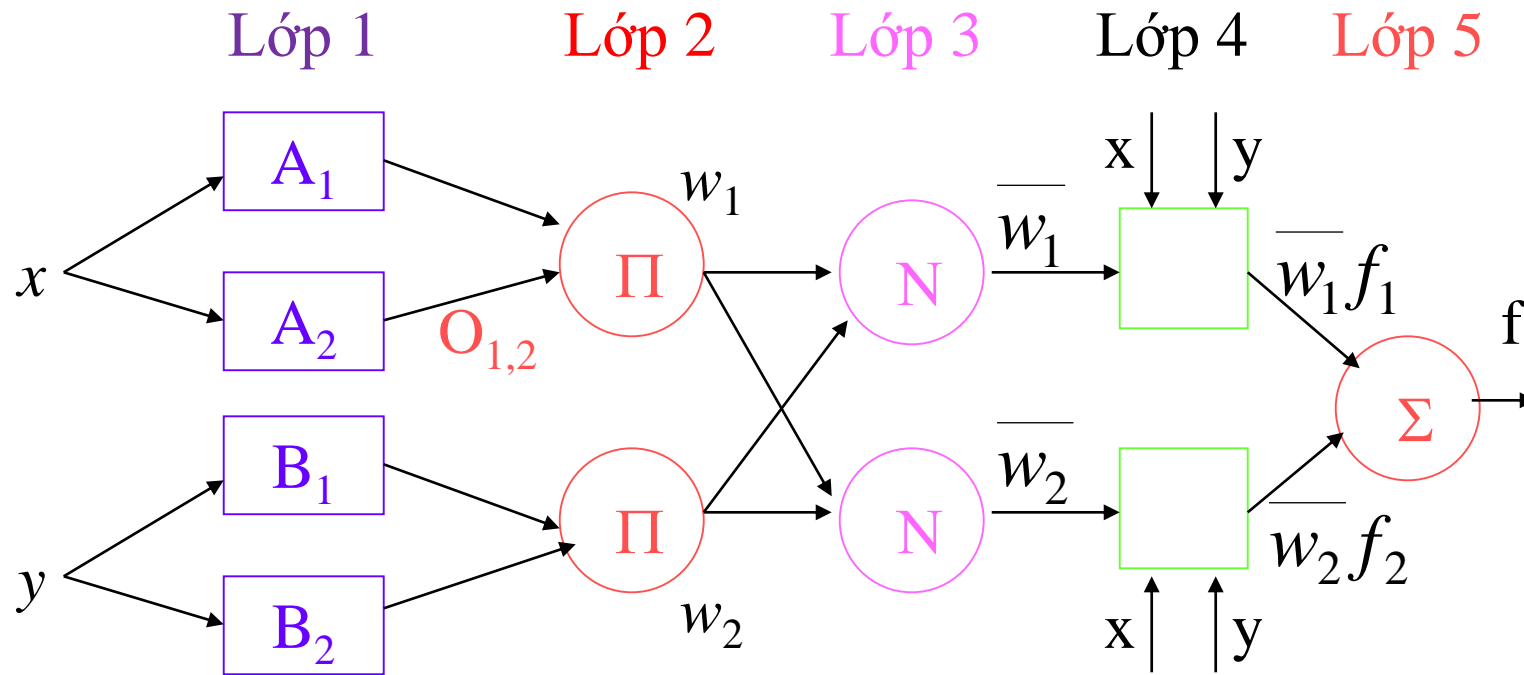
# ANFIS

---



# Cấu trúc mạng ANFIS

Đầu ra của nút  $i$  tại lớp  $j$  được ký hiệu là  $O_{j,i}$





# ANFIS

---

**Lớp 1:** gồm các nơ ron có hàm truyền là  $O_{1,i} = \mu_{A_i}(x)$ , for  $i = 1,2$  or  $O_{1,i} = \mu_{B_{i-2}}(y)$  for  $i = 3,4$

Biểu diễn giá trị thành viên của các tập mờ  $A_i$  và  $B_j$  cho các đầu vào  $x$  và  $y$ .

**Lớp 2:** gồm các nơ ron có giá trị đầu ra được xác định bởi:

$$O_{2,i} = w_i = \mu_{A_i}(x) \text{ T } \mu_{B_i}(y), \text{ for } i = 1,2$$

Trong đó T là toán tử T-norm, hoặc toán tử AND biểu diễn hàm MIN hoặc hàm nhân.

Các nơ ron trong lớp này được cố định.

# ANFIS

---

**Lớp 3:** mỗi nơ ron của lớp này sẽ tính toán tỉ lệ đầu ra của nó so với tổng đầu ra của tất cả các nơ ron trong lớp 3

$$O_{3,i} = \overline{w_i} = \frac{w_i}{w_1 + w_2} \quad \text{for } i = 1, 2$$

Kết quả của lớp 3 còn được gọi là giá trị mờ chuẩn hóa.

**Lớp 4:** mỗi nơ ron của lớp này được gọi là nơ ron thích ứng, có đầu ra là

$$O_{4,i} = \overline{w_i} f_i = \overline{w_i} (p_i x + q_i y + r_i) \quad \text{for } i = 1, 2$$

$p_i, q_i, r_i$  là các tham số suy diễn, và  $x, y$  là các thông tin đầu vào kết hợp với nhau theo logic Sugeno

# ANFIS

---

**Lớp 5:** chỉ có một nơ ron, tổng hợp toàn bộ các thông tin truyền đến theo hàm bình quân trọng số.

$$O_{5,1} = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i}$$

# ANFIS

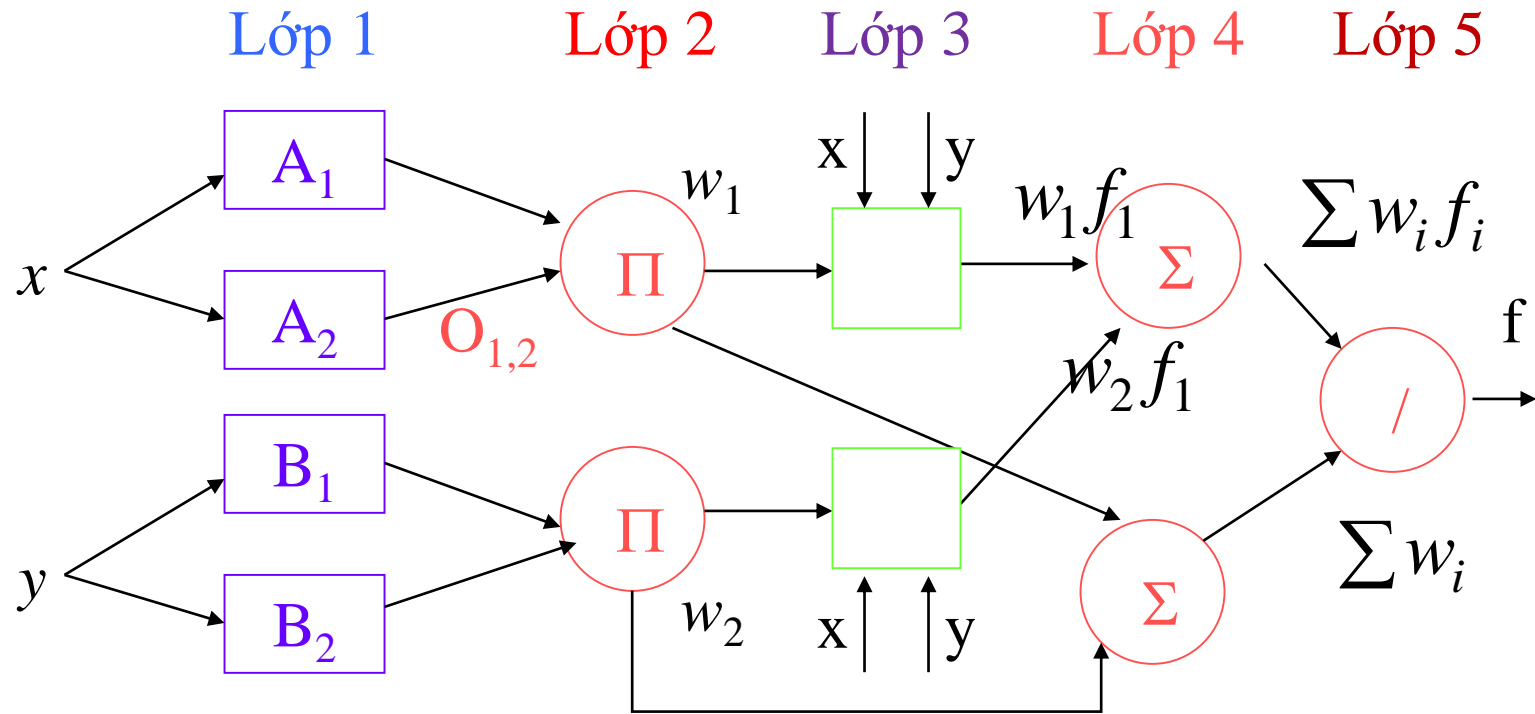
---

Cấu trúc của một hệ thống ANFIS không cố định, có thể thay đổi tùy từng bài toán khác nhau.

- Các lớp 3 và 4 có thể kết hợp với nhau tạo thành một lớp chung.
- Quá trình chuẩn hóa có thể được thực hiện tại lớp cuối cùng.

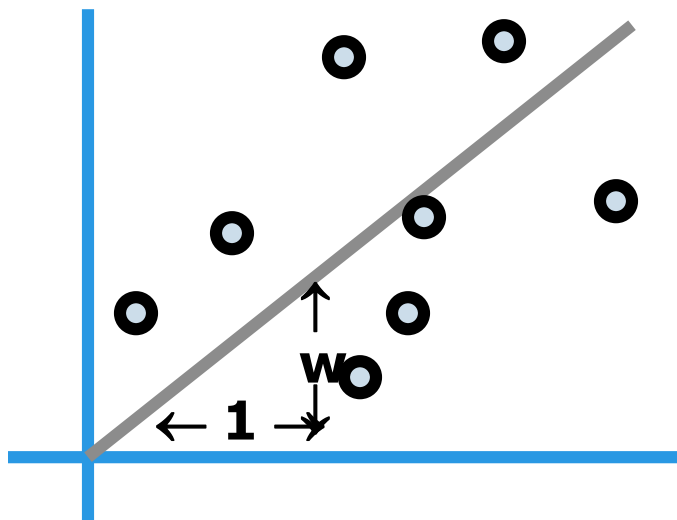
# ANFIS

Ví dụ về một cấu trúc khác cho mạng ANFIS



# Bayesian Neural Networks

Mạng nơ ron và bài toán Linear Regression



## Dữ liệu huấn luyện

inputs	outputs
$x_1 = 1$	$y_1 = 1$
$x_2 = 3$	$y_2 = 2.2$
$x_3 = 2$	$y_3 = 2$
$x_4 = 1.5$	$y_4 = 1.9$
$x_5 = 4$	$y_5 = 3.1$

# Linear Regression

---

Tìm mối quan hệ tuyến tính giữa dữ liệu đầu vào và dữ liệu đầu ra

$$y = \mathbf{w} \cdot \mathbf{x}$$

Với  $\mathbf{w}$  là tham số của hệ thống. Huấn luyện hệ thống là quá trình tìm  $\mathbf{w}$  để biểu diễn đúng mối quan hệ tuyến tính giữa đầu vào và đầu ra dựa trên dữ liệu huấn luyện.

# Bayesian Linear Regression

---

Giả sử, giá trị đầu ra của một mạng nơ ron với một bộ dữ liệu đầu vào  $x_i$  được tính như sau:

$$y_i = \mathbf{w} \cdot \mathbf{x}_i + \mathbf{noise}_i$$

Trong đó,  $\mathbf{noise}_i$  là một thành phần độc lập với quá trình xử lý của nơ ron

$\mathbf{noise}_i$  có phân bố Gauss,  $\mathcal{N}(0, \sigma^2)$

Như vậy  $y_i$  cũng sẽ có phân bố Gauss,  $\mathcal{N}(\mathbf{w} \cdot \mathbf{x}_i, \sigma^2)$

$$P(y_i | \mathbf{w} \cdot \mathbf{x}_i) \sim \mathcal{N}(\mathbf{w} \cdot \mathbf{x}_i, \sigma^2)$$



# Bayesian Linear Regression

---

Giả sử ta có các dữ liệu huấn luyện  $(\mathbf{x}_1, y_1) (\mathbf{x}_2, y_2) \dots (\mathbf{x}_n, y_n)$  được dùng để ước tính giá trị  $\mathbf{w}$

Khi đó, giá trị của  $\mathbf{w}$  có thể được ước tính bằng cách tính xác suất có điều kiện:

$$P(\mathbf{w} | \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_n, y_1, y_2, \dots, y_n)$$

Luật Bayes có thể giúp chúng ta làm được điều này.

Maximum Likelihood Estimation cũng có thể được sử dụng để tính giá trị của  $\mathbf{w}$

# Maximum Likelihood Estimation

---

Giá trị nào của  $\mathbf{w}$  để các dữ liệu vào-ra có khả năng xuất hiện đúng nhất?

Giá trị nào của  $\mathbf{w}$  để  $P(y_1, y_2 \dots y_n | \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots \mathbf{x}_n, \mathbf{w})$  mang giá trị cao nhất?

Giá trị nào của  $\mathbf{w}$  để tối đa hóa biểu thức:

$$\prod_{i=1}^n P(y_i | \mathbf{w}, \mathbf{x}_i) = \prod_{i=1}^n \exp \left( -\frac{1}{2} \left( \frac{y_i - \mathbf{w} \mathbf{x}_i}{\sigma} \right)^2 \right)$$

Giá trị nào của  $\mathbf{w}$  để tối đa hóa biểu thức

$$\sum_{i=1}^n -\frac{1}{2} \left( \frac{y_i - \mathbf{w} \mathbf{x}_i}{\sigma} \right)^2$$

Giá trị nào của  $\mathbf{w}$  để tối thiểu hóa biểu thức

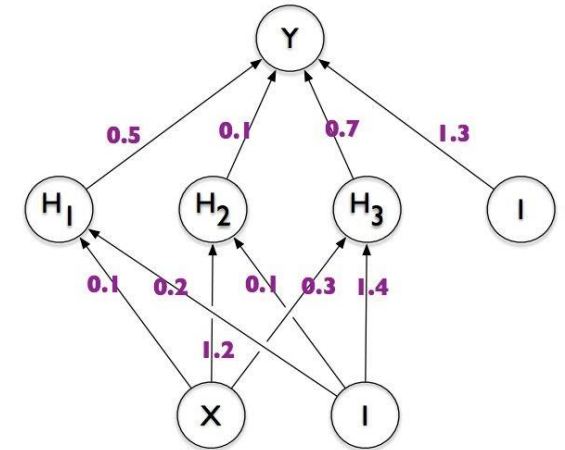
$$\sum_{i=1}^n (y_i - \mathbf{w} \mathbf{x}_i)^2$$

# Bayesian Neural Network

Mạng nơ ron thông thường:

Các tham số của mạng được biểu diễn bởi các giá trị cố định, đơn lẻ

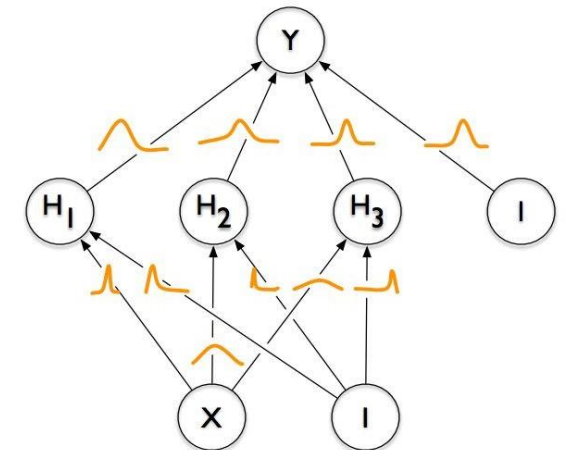
Quá trình huấn luyện dựa trên các thuật toán ước tính trọng số và bias



Bayesian Neural Network:

Các tham số được biểu diễn bởi các hàm phân bố xác suất

Quá trình huấn luyện dựa trên phân bố tiên định  $P(w)$  để từ đó ước tính xác suất  $P(w|D)$  sử dụng quá trình học Bayes



# Ước tính dựa trên Bayes

---

## Tối thiểu hóa:

Squared error (no regularization)

$$\sum_{i=1}^N (y(x^{(i)}; \mathbf{w}) - t^{(i)})^2$$



## Tương ứng với:

Maximum likelihood estimation

$$\mathbf{w}_{MLE} = \arg \max_{\mathbf{w}} P(\mathcal{D} | \mathbf{w})$$

Squared error (+  $L^2$  regularization)

$$\beta \sum_{i=1}^N (y(x^{(i)}; \mathbf{w}) - t^{(i)})^2 + \alpha \|\mathbf{w}\|^2$$



MAP estimation with a Gaussian prior

$$\mathbf{w}_{MAP} = \arg \max_{\mathbf{w}} P(\mathbf{w} | \mathcal{D})$$

$$\text{where } \mathbf{w} \sim \mathcal{N}(0, \sigma_w^2)$$

# Bayesian methods

---

Ước tính phân bố xác suất đầu ra của mạng

$$P(y \mid x, \mathcal{D}) = \int P(y \mid \mathbf{w}, x) P(\mathbf{w} \mid \mathcal{D}) d\mathbf{w}$$

Tính toán các thông số quan hệ đầu vào – đầu ra

$$P(\mathcal{D} \mid \mathcal{H}_i) = \int P(\mathcal{D} \mid \mathbf{w}, \mathcal{H}_i) P(\mathbf{w} \mid \mathcal{H}_i) d\mathbf{w}$$

Các phép tính tích phân thường được ước lượng dựa trên dữ liệu hoặc các mô hình phân bố xác suất giả định (Gauss).

# Bayesian Neural Networks

---

Xác suất tiên định về dữ liệu đầu ra  $p(t|\mathbf{x}, \mathbf{w}, \beta) = \mathcal{N}(t|y(\mathbf{x}, \mathbf{w}), \beta^{-1})$ .

Xác suất tiên định về các trọng số:  $p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbf{I})$

Hàm likelihood giả định:  $p(\mathcal{D}|\mathbf{w}, \beta) = \prod_{n=1}^N \mathcal{N}(t_n|y(\mathbf{x}_n, \mathbf{w}), \beta^{-1})$

Ước tính giá trị trọng số:  $p(\mathbf{w}|\mathcal{D}, \alpha, \beta) \propto p(\mathbf{w}|\alpha)p(\mathcal{D}|\mathbf{w}, \beta)$

Xác suất này không tuân theo phân bố Gauss vì hàm truyền trong mạng nơ ron không tuyến tính.

Quá trình tính toán sử dụng hàm log để tính trọng số đầu ra vì có thể xấp xỉ Gauss.