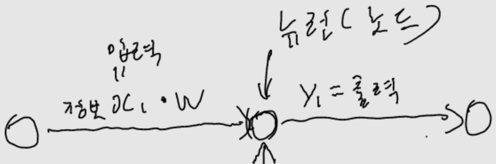
DL : perceptron ⇒ multi layer perceptron (MLP)

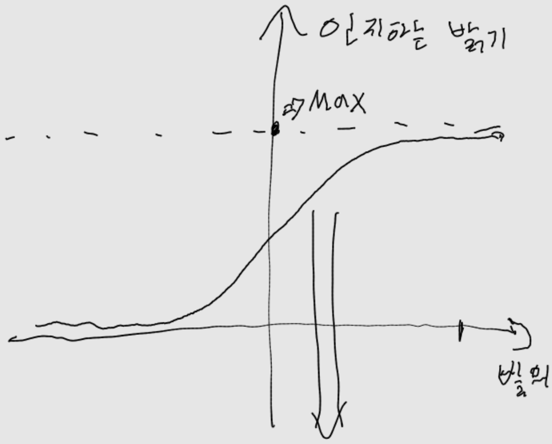
알고리즘의 이름 (다수의 입력 → 하나의 출력)



더해지는 값 : 바이어스 (bias) : b

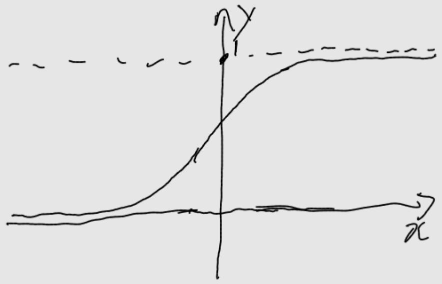
$w \cdot x_1 + b$

∴ $y_1 = A(w_1 x_1 + b)$ ⇒ 노드와 노드 간의 프로세스



활성화 함수

(Activation function)

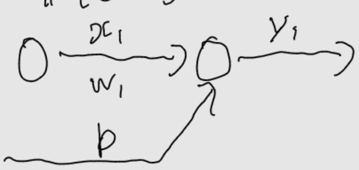


⇒ 활성화 함수

ex) sigmoid 함수

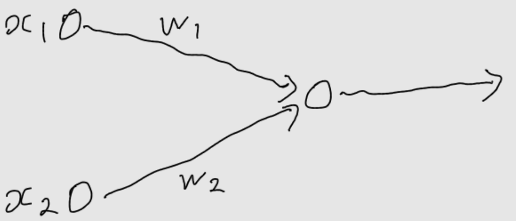
$$\rightarrow \underset{y축}{\sigma(x)} = \frac{1}{1+e^{-x}} \Rightarrow \begin{cases} x \to \infty \Rightarrow \sigma(x) = 1 \\ x \to -\infty \Rightarrow \sigma(x) = 0 \end{cases}$$

• 뉴런(노드) process



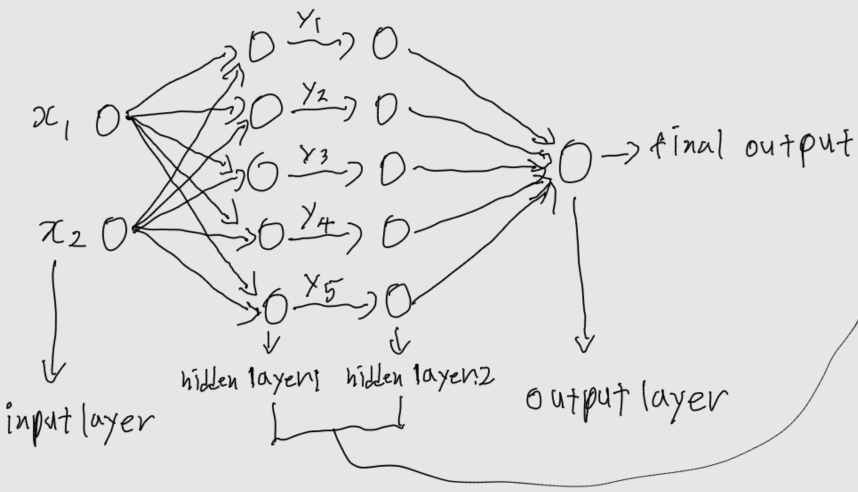
$y_1 = w_1 x_1 + b \in R$

• 퍼셉트론 (다수개의 입력 → 하나의 출력)



$y_1 = A(w_1 x_1 + w_2 x_2 + b)$ ⇒ $\sigma(x) = \frac{1}{1+e^{-x}}$

•



• hidden layer 가 전부 연결되어 있는 경우

⇒ Fully Connected Layer (FC Layer)

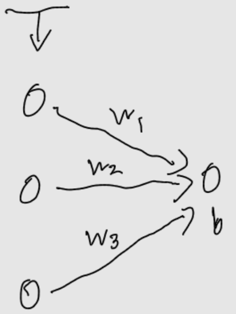
→ 노드수 +1 개의 b

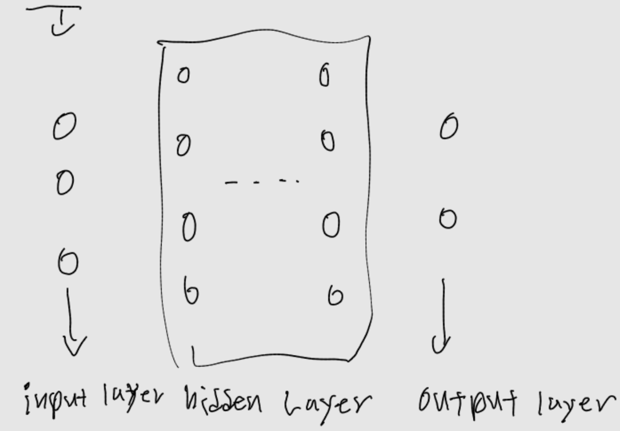
→ 연결마다 고유의 w

hidden layer 가 여러개

⇒ Multi Layer Perceptron (MLP)

• SLP vs MLP



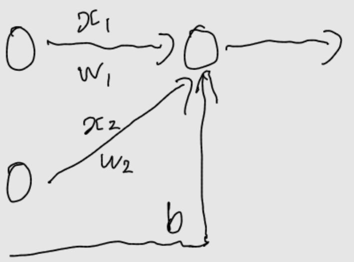


∴ SLP는 hidden layer 존재X, MLP는 존재

• input과 output 사이의 hidden layer가 수없이 많은 경우 (Deep)하다고 표현

→ 우리가 hidden layer 를 훈련시킴 (learning)

⇒ Deep Learning = Deep neural network (DNN)

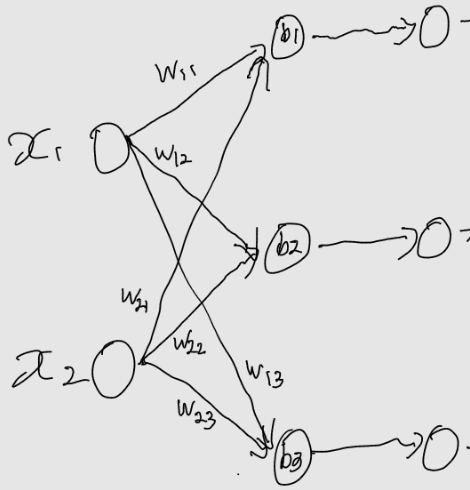
• 

$$y = A(\underbrace{x_1 w_1 + x_2 w_2 + b}_{Z})$$

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2, \quad w = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \in \mathbb{R}^2$$

$$Z = w^T \cdot x \in \mathbb{R}$$

$(1\times2)\ (2\times1) \Rightarrow (1\times1)$

• 

$\rightarrow W_{11}x_1 + W_{21}x_2 + b_1$

$\rightarrow W_{12}x_1 + W_{22}x_2 + b_2$

$\rightarrow W_{13}x_1 + W_{23}x_2 + b_3$

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2$$

$$b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \in \mathbb{R}^3$$

$$w = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \end{bmatrix} \in \mathbb{R}^{2\times3}$$

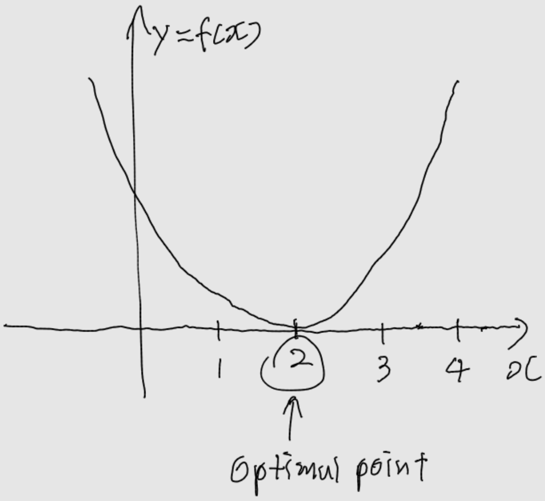
$$A(W^T \cdot x + b) = Y \in \mathbb{R}^3$$

$(3\times2)\ (2\times1)$

$\Downarrow \qquad \Downarrow$

$(3\times1) \qquad (3\times1)$

• Gradient Decent Method (GD Method)



$y = f(x) = (x-2)^2$

$y' = f'(x) = 2(x-2)$ ⇒ 도함수

$$\underbrace{y'(0) = -4 \quad y'(1) = 2 \quad y'(2) = 0 \quad y'(3) = 2 \quad y'(4) = 4}_{\text{특정한 순간의 기울기 (Gradient) : 미분계수}}$$

$$w = w - \underbrace{\nabla f(w)}_{-4}$$

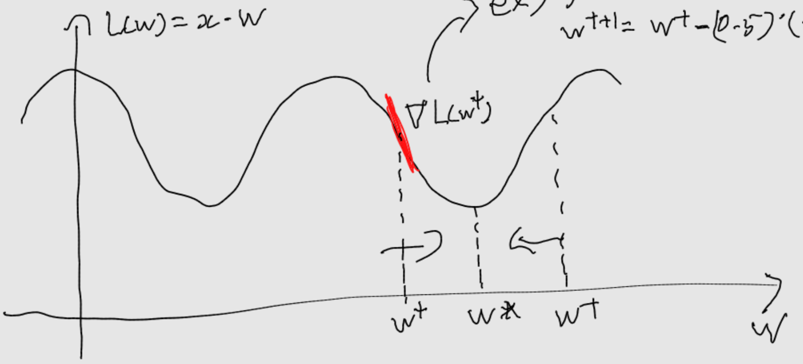
$$w = \underbrace{w + 4}_{\text{update}}$$

w: 기울기 $\nabla f(w)$: 미분계수 η: 학습률

$$\therefore w = w - \eta \cdot \nabla f(w)$$

ex) $g = -4$, $\eta = 0.5$

$w^{t+1} = w^t - (0.5) \cdot (-4) = w^t + 2$



• $w^{t+1} = w^t - \eta \cdot \nabla L(w^t)$

$\nabla L(w^t) < 0$ 일 경우, $w^{t+1} > w^t$

$\nabla L(w^t) > 0$ 〃 , $w^{t+1} < w^t$

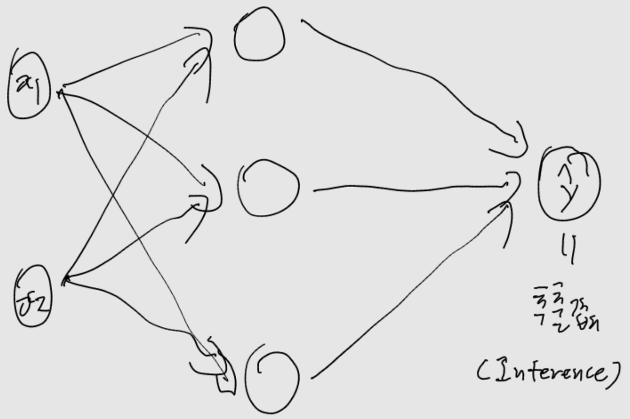
$x_1 \xrightarrow{w_1} y_1 \xrightarrow{w_1 x}$ 작을수록 좋다고 가정 ⇒ Loss Function (L) ≈ cost

입력(상수)

• MLP에서 트레이닝은 W 값을 조절해 L을 작게 하는 것

$L(w) = x \cdot w$

• 역전파



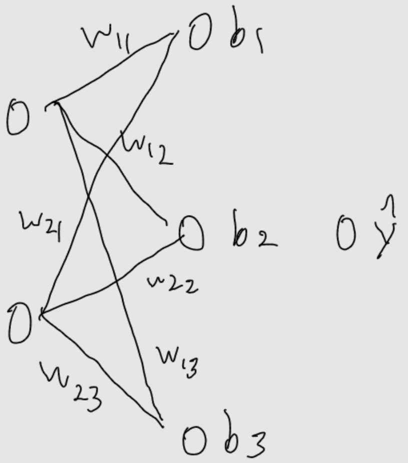
• Lable : 정답

	입력	정답
x^1 =	(x_1^1, x_2^1)	(y_1)
x^2 =	(x_1^2, x_2^2)	(y_2)
	(x_1^3, x_2^3)	(y^3)
	⋮	⋮

Training set

$$Loss = \sum_{i=1}^{3} |\hat{y} - y| = L_1 \, Loss$$

$$or = \sqrt{\sum_{i=1}^{3} (\hat{y} - y)} = L_2 \, Loss$$



• 오차 = Loss Function (작을수록 좋음)

• 그러므로 w와 b를 update

• 나머지는 강의자료 참고