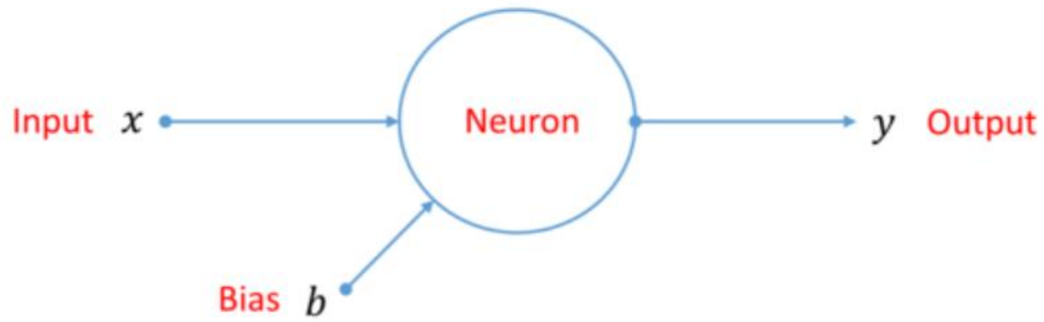
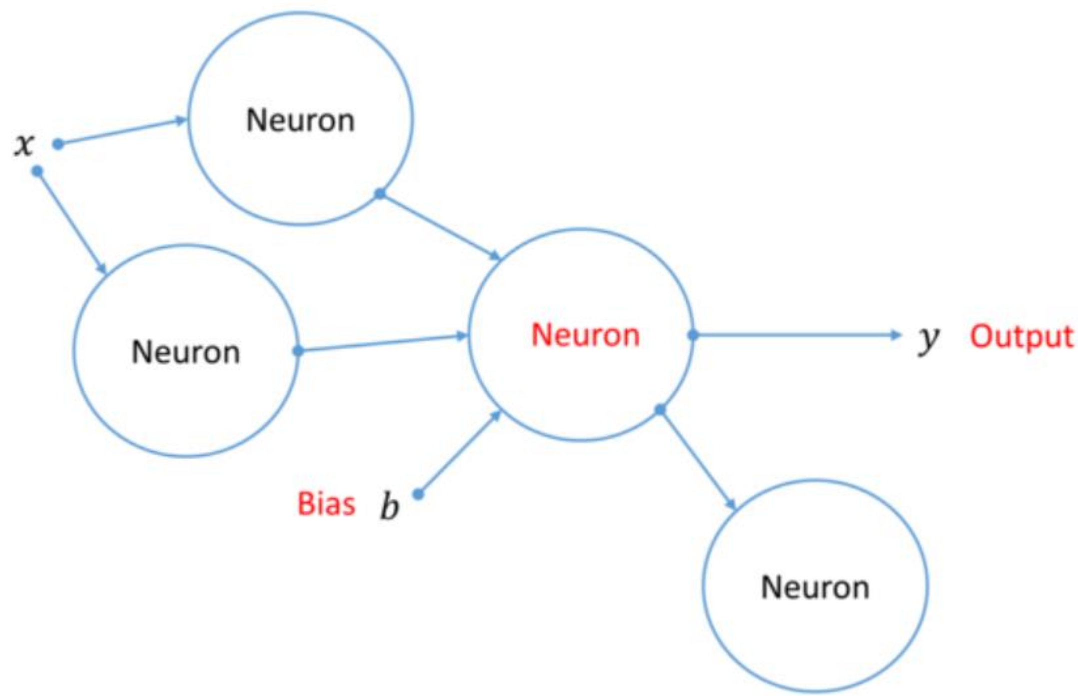
# 인공 신경망
# 퍼셉트론의 이해

# 인공 신경 세포(Artificial Neuron)

- **뉴런**
  - 입력
  - 편향(bias)

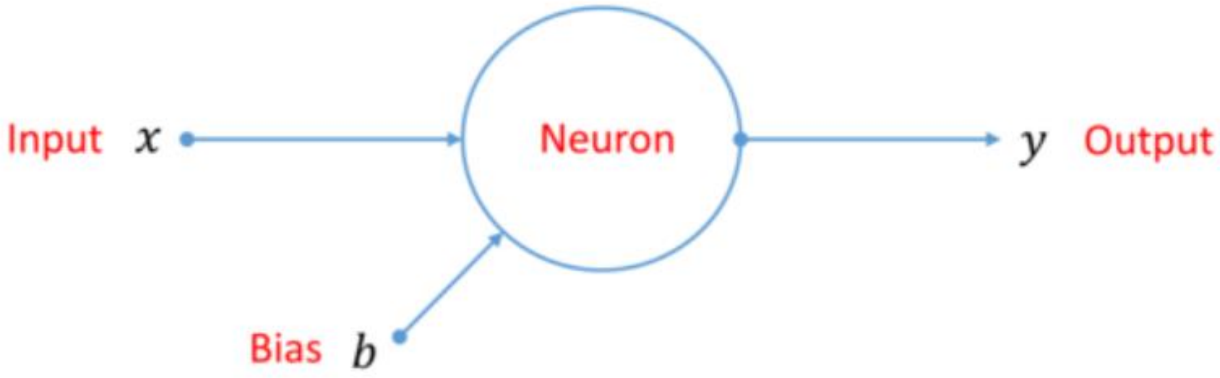



- **신경망(network)**
  - 뉴런의 연결



# 입력과 출력

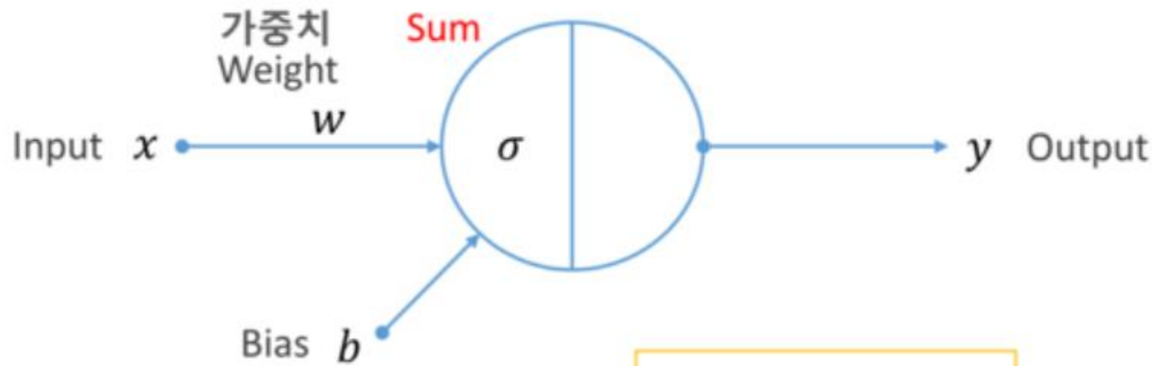
- **편향(bias)**
  - 편향을 조정해 출력을 맞춤




| **Input $x$** | **Output $y$** |
|---|---|
| Size of house | Price |
| Time spent for studying | Score in exam |

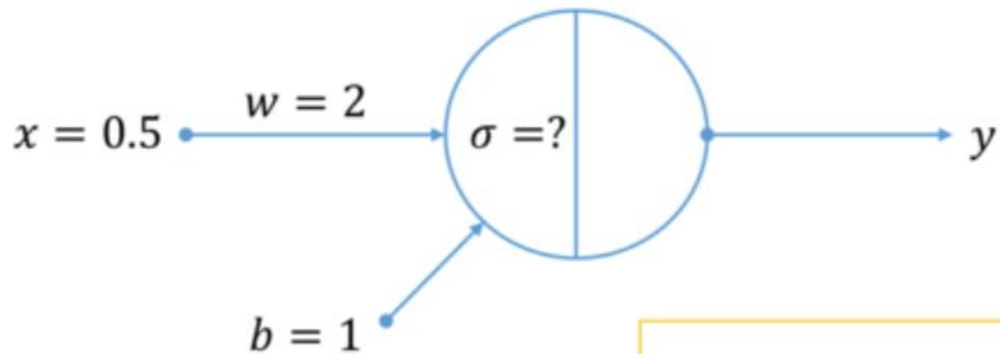
# 뉴런 연산

- **뉴런 식**



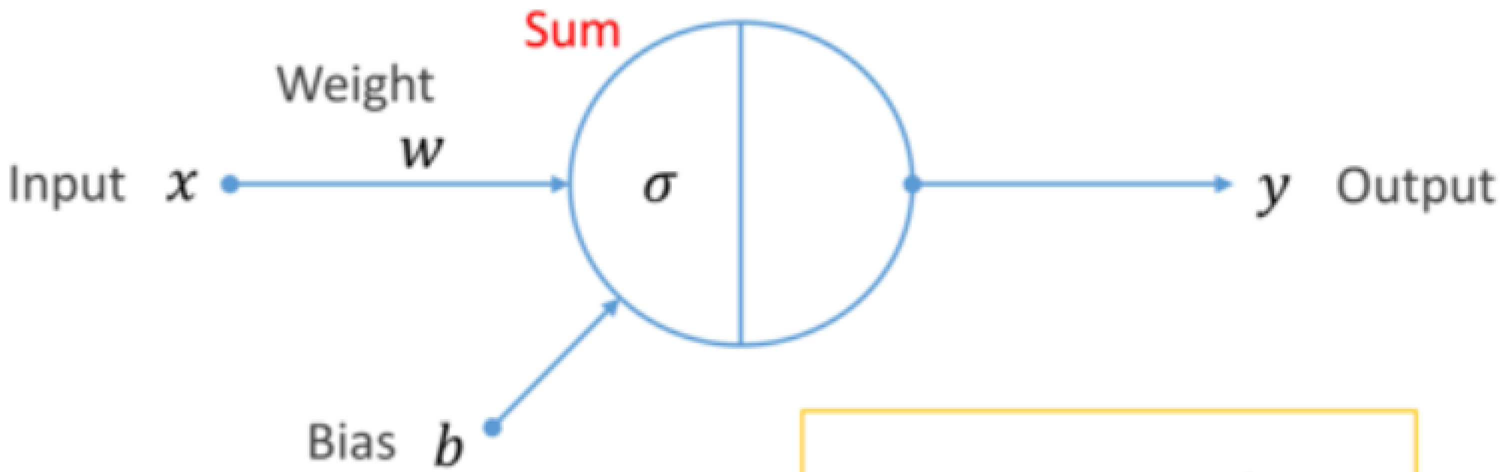

$$\sigma = w \cdot x + b$$

- **가중치와 편향**




$$\begin{aligned}\sigma &= w \cdot x + b \\ &= 2 \cdot 0.5 + 1 \\ &= 2\end{aligned}$$

# 행렬 곱 연산

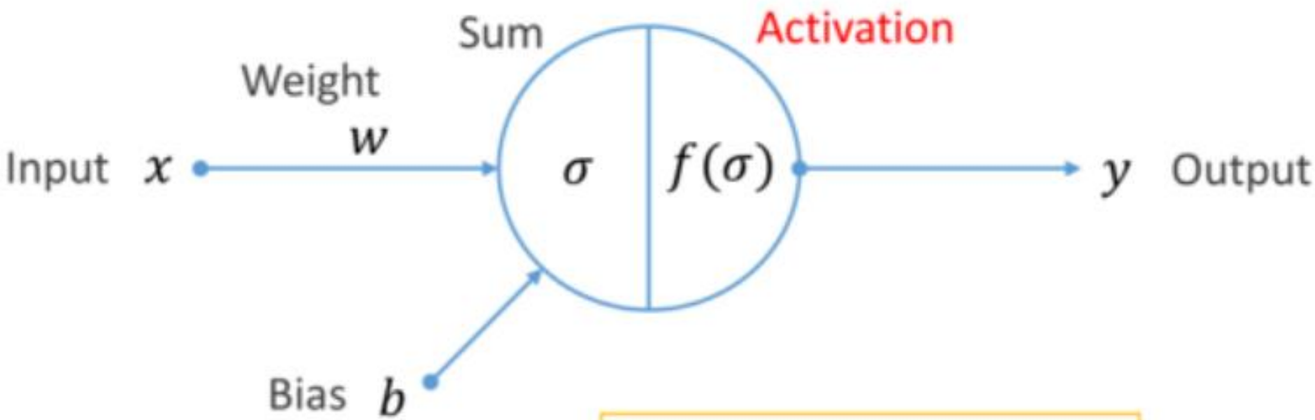



$$\begin{aligned}\sigma &= w \cdot x + b \\ &= [w\ b]\begin{bmatrix} x \\ 1 \end{bmatrix}\end{aligned}$$

행렬의 곱

# 활성화

- **활성화 함수**
  - 뉴런의 출력 값을 정하는 함수

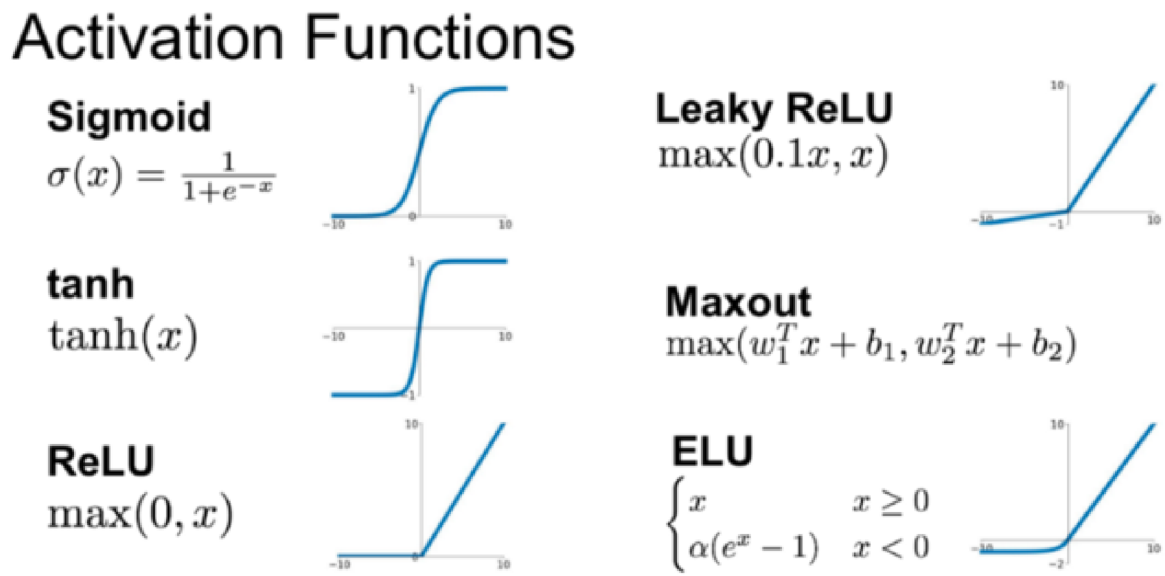



$$\sigma = w \cdot x + b$$

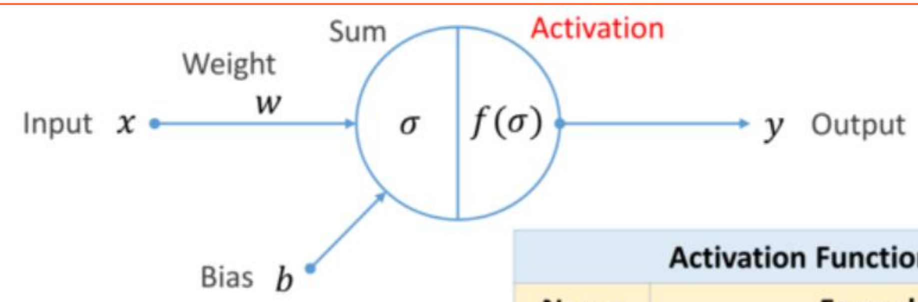
$$f(\sigma) = f(w \cdot x + b)$$

# 활성화 함수 종류

- **ReLU(교재 p43)**
  - Rectified(정류된) Linear Unit(선형 함수, y=x를 의미)
    - **선형 함수를 정류하여 0 이하는 모두 0으로 한 함수**
    - **Max(x, 0)**
      - **양수만 사용**
  - 2010년 이후
    - **층이 깊어질수록(deep) 많이 활용**
      - **양수를 그대로 반환하므로 값의 왜곡이 적어지는 효과**
    - **토론토 대학 힌트 교수**
- **Sigmoid**
  - s자 형태의 곡선이라는 의미
    - **예전에 많이 사용**




Different Activation Functions and their Graphs




| Activation Functions | |
|---|---|
| **Name** | **Formula** |
| Identity | $f(x) = x$ |
| ReLU | $f(x) = \max(x, 0)$ |
| Sigmoid | $f(x) = \frac{1}{1+e^{-x}}$ |

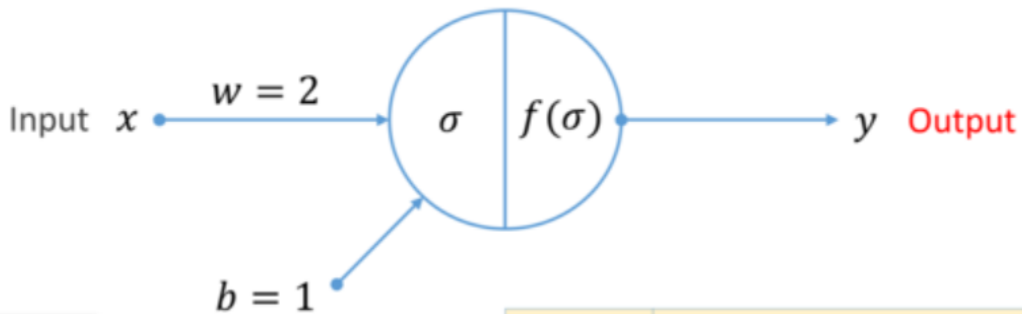
*ReLU: Rectified Linear Unit

# 입출력의 예

- **출력함수로**
  - 동일(identity) 함수를 적용

$$y = f(\sigma) = f(w \cdot x + b) = w \cdot x + b$$

with **identity** (or **linear**) activation functions

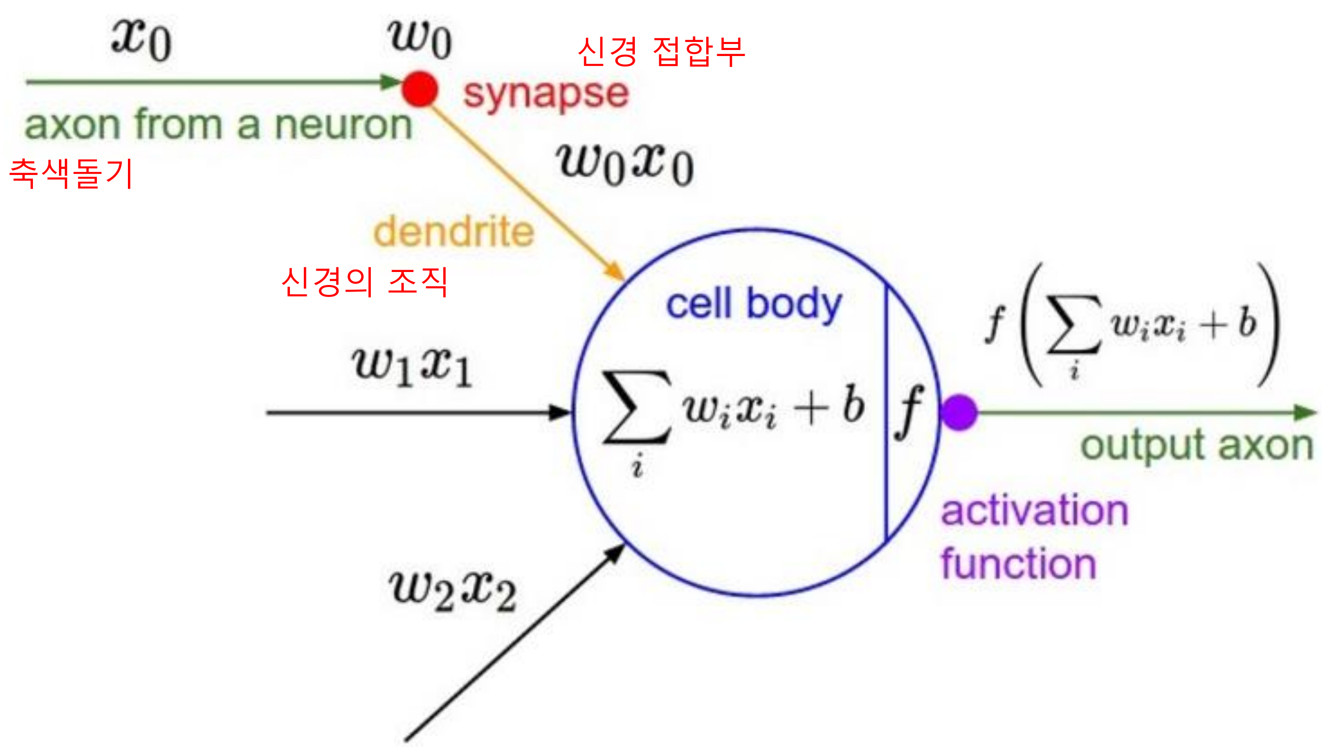




$f(x) = x$

| Input $x$ | Output $y$ |
|---|---|
| 0 | $y = f(2 \cdot 0 + 1) = 1$ |
| 1 | $y = f(2 \cdot 1 + 1) = 3$ |
| 2 | $y = f(2 \cdot 2 + 1) = 5$ |

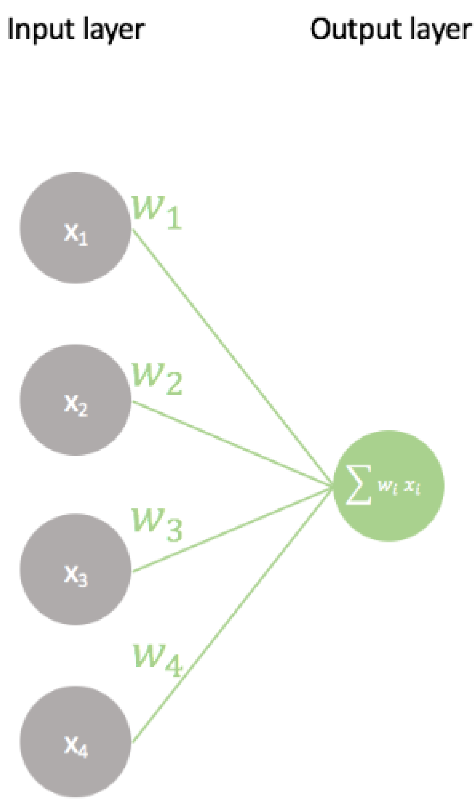
# 일반화된 인공신경망



$x_0$
$w_0$
신경 접합부
synapse
axon from a neuron
축색돌기
$w_0 x_0$
dendrite
신경의 조직
cell body
$\sum_i w_i x_i + b$
$f$
$f\left(\sum_i w_i x_i + b\right)$
$w_1 x_1$
output axon
activation
function
$w_2 x_2$

# 활성화 함수와 편향

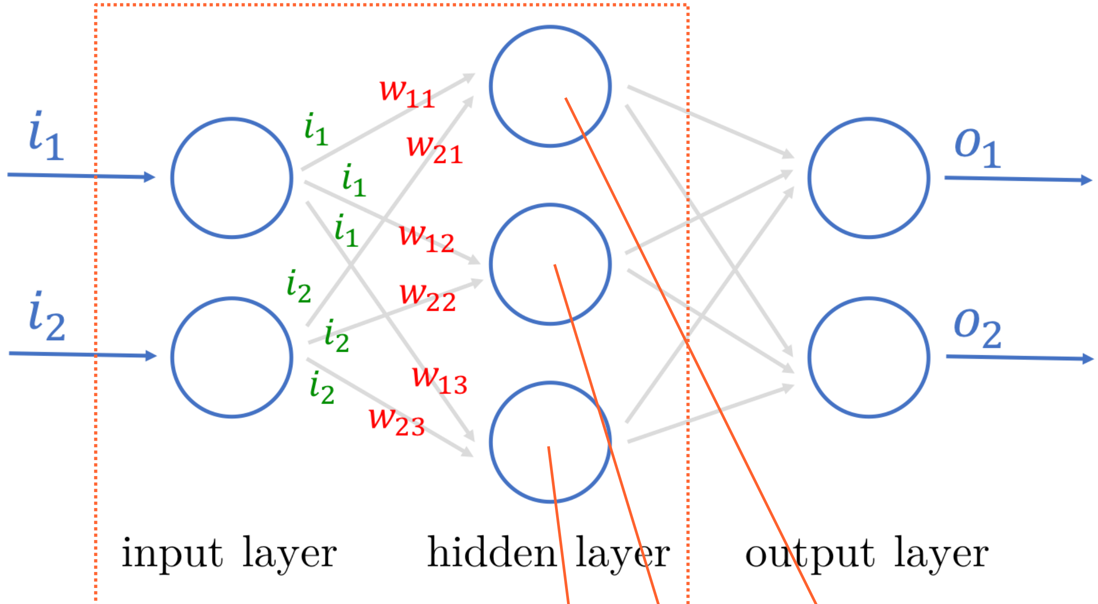
- **결과 값이 임계 값 역할**
  - 결과가 임계 값 이상이면 활성화
  - 결과가 임계 값 미만이면 비활성화




## Perceptron Unit

$$f_w(x) = \begin{cases} \sum w_i x_i \geq \theta \rightarrow \text{neuron fires} \\ \sum w_i x_i < \theta \rightarrow \text{neuron doesn't fire} \end{cases}$$

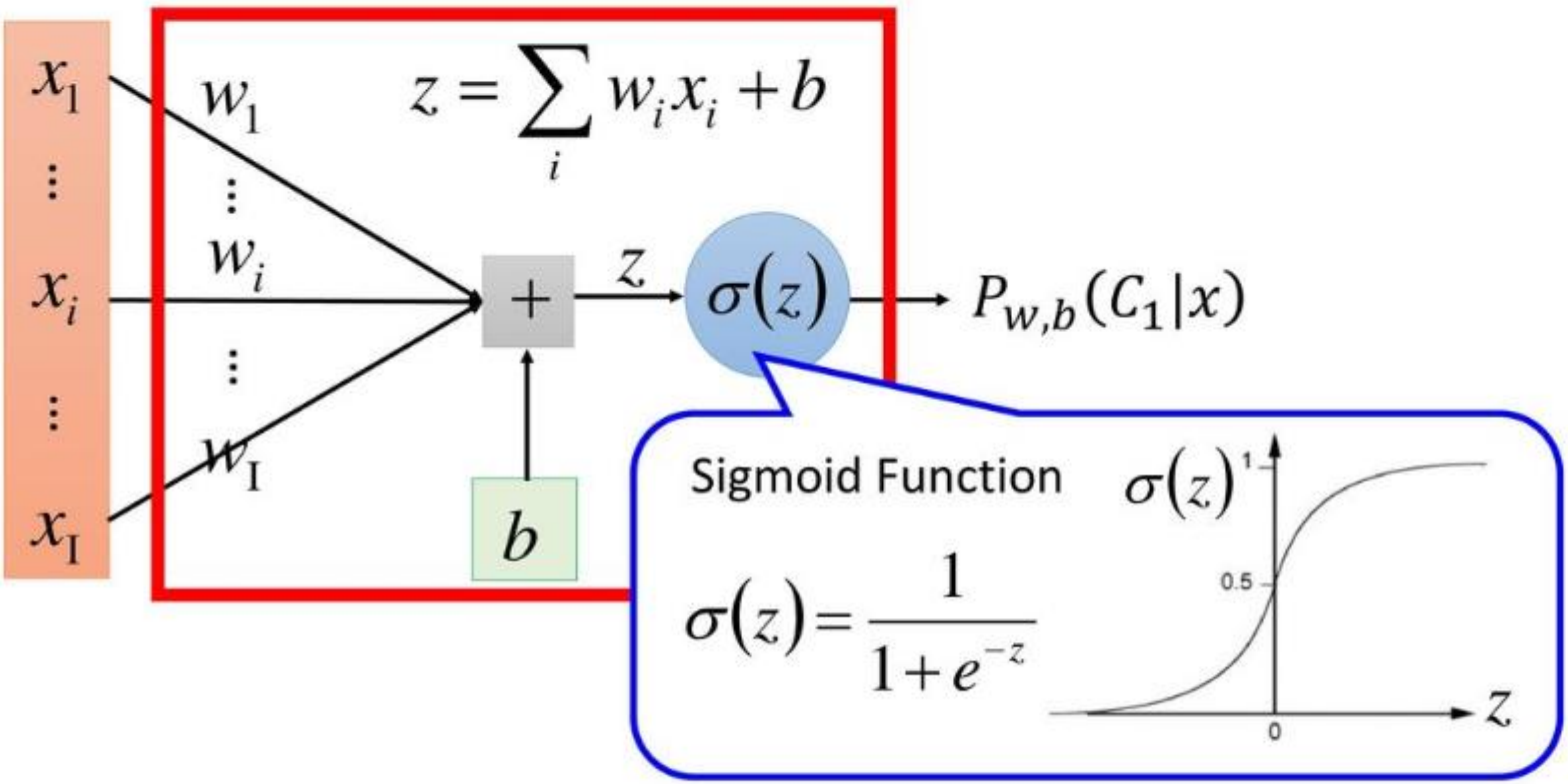
# 입력 2개, 출력 3개인 신경망 연산




$$\begin{bmatrix} w_{11} & w_{21} \\ w_{12} & w_{22} \\ w_{13} & w_{23} \end{bmatrix} \cdot \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} = \begin{bmatrix} (w_{11} \times i_1) + (w_{21} \times i_2) \\ (w_{12} \times i_1) + (w_{22} \times i_2) \\ (w_{13} \times i_1) + (w_{23} \times i_2) \end{bmatrix}$$

# 인공신경망의 시그모이드 함수

- **활성화 함수의 예**
  - 시그모이드 함수
    - **출력 값이 (0~1)**



# 가중치

- **3 × 3의 가중치 실수**

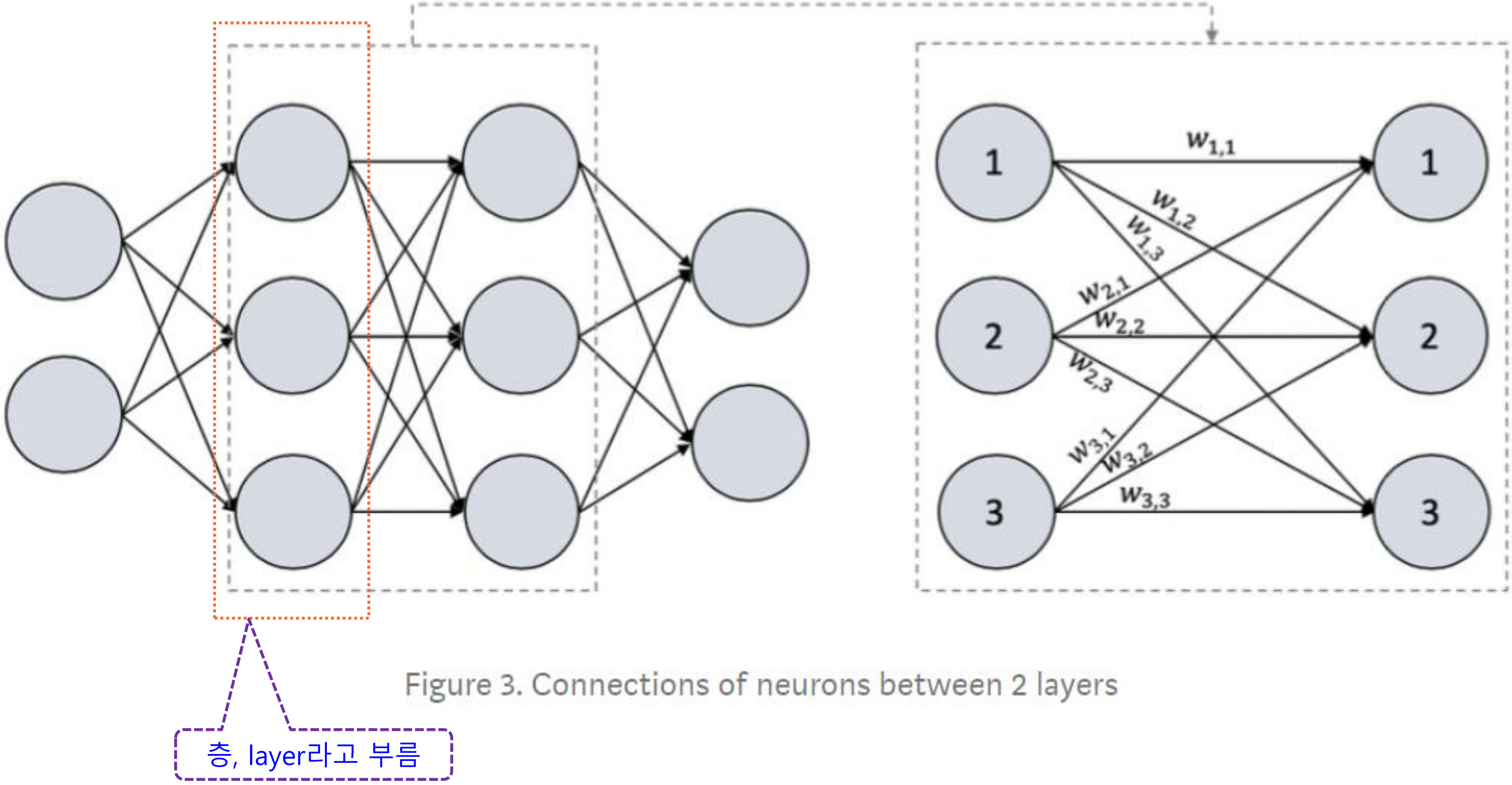



Figure 3. Connections of neurons between 2 layers

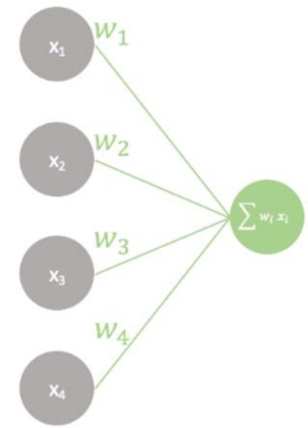


https://towardsdatascience.com/neural-networks-i-notation-and-building-blocks-817b1d2ea04b

# 인공 신경망
# 행렬 연산

# 입력의 특징($x_1$ $x_2$ $x_3$ ... $x_i$)과 입력의 자료 수

- **특징 n개가 있는 뉴런 신경망에서 하나의 출력 계산**




✓ **샘플 수 1개에 대한 계산**

Geektail

$$\boldsymbol{x} = [x_1 \quad x_2 \quad \cdots \quad x_n] \qquad \boldsymbol{w}^T = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix}$$

$(1 \times n)$ $(n \times 1)$

$$\boldsymbol{x}\boldsymbol{w}^T = w_1 x_1 + w_2 x_2 + \cdots w_n x_n$$
$$= z$$

스칼라

$(1 \times 1)$

n: MNIST 손글씨에서 손글씨 이미지 하나의 펙셀 수인 786(28 x 28)을 의미

✓ **샘플 수 s개에 대한 계산**

$$X = \overset{n}{\begin{bmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{s1} & \cdots & x_{sn} \end{bmatrix}} s \qquad \boldsymbol{w}^T = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix}$$

$(s \times n)$ $(n \times 1)$

$$X\boldsymbol{w}^T = \begin{bmatrix} w_1 x_{11} + w_2 x_{12} + \cdots + w_n x_{1n} \\ w_1 x_{21} + w_2 x_{22} + \cdots + w_n x_{2n} \\ \vdots \\ w_1 x_{s1} + w_2 x_{s2} + \cdots + w_n x_{sn} \end{bmatrix}$$
$$= \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_s \end{bmatrix} = \boldsymbol{z}$$

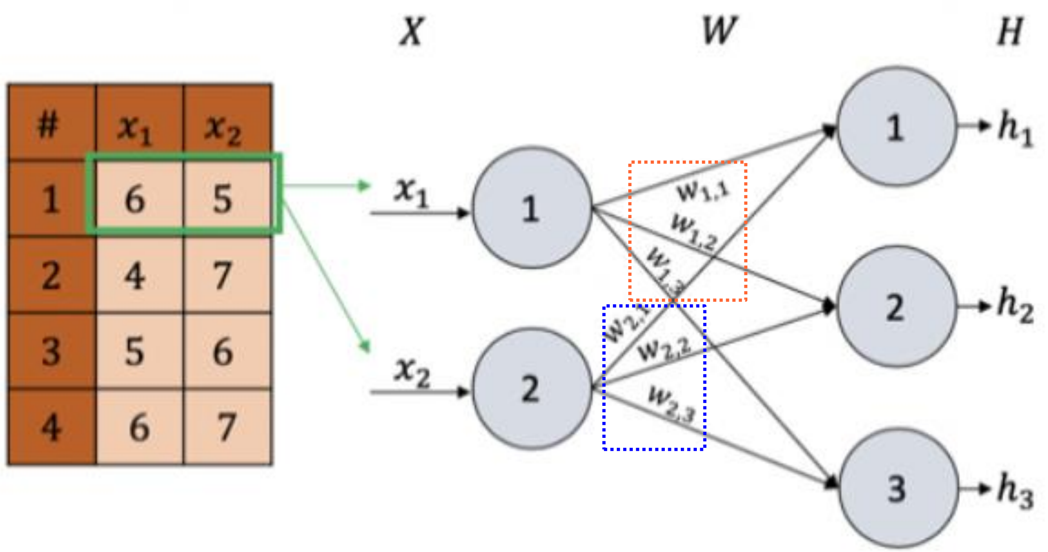
벡터

$(s \times 1)$

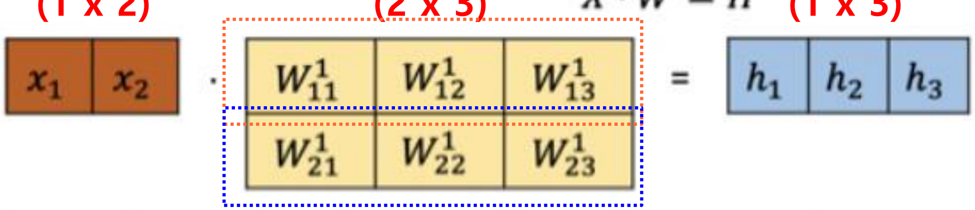
s: MNIST 손글씨에서 훈련 데이터 6만개에서 6만을 의미

# 신경망 행렬 계산

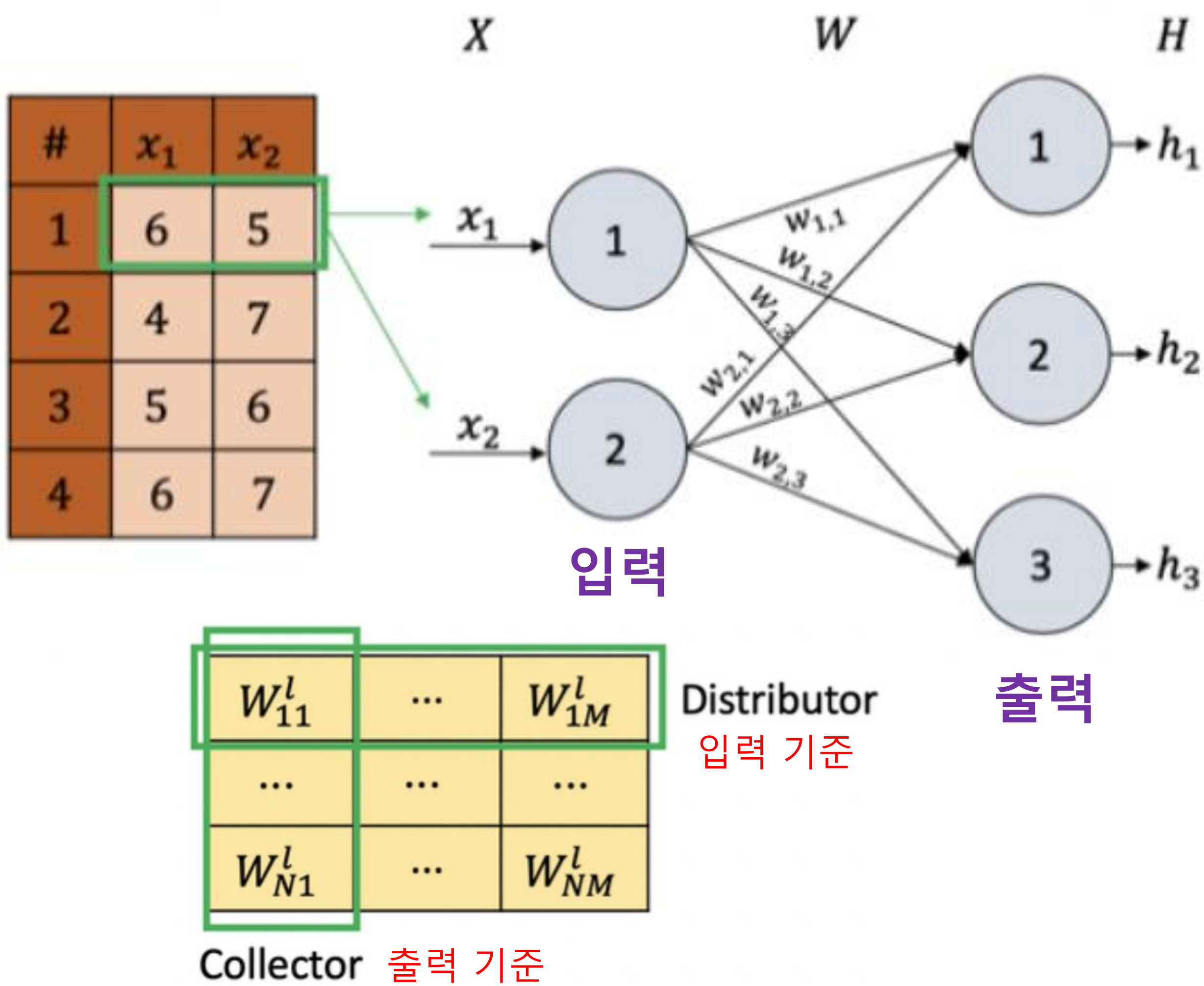
- **특징 2개**
  - 샘플 수 4



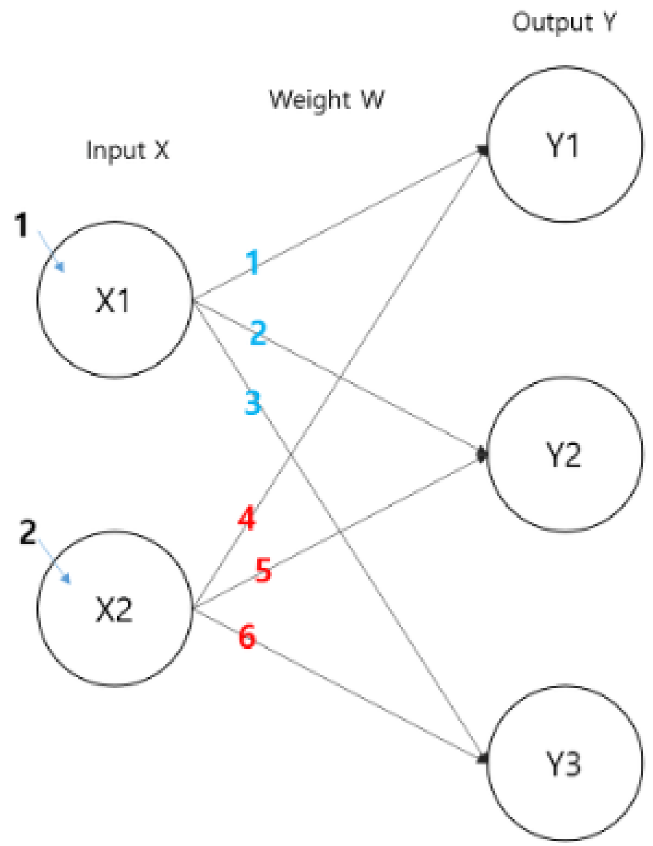

(1 x 2) (2 x 3) $X \cdot W = H$ (1 x 3)



# 뉴런 계산


X
W
H
#
$x_1$
$x_2$
1
6
5
2
4
7
3
5
6
4
6
7
$x_1$
$x_2$
1
2
$w_{1,1}$
$w_{1,2}$
$w_{1,3}$
$w_{2,1}$
$w_{2,2}$
$w_{2,3}$
1
2
3
$h_1$
$h_2$
$h_3$
입력
출력
$W_{11}^l$
…
$W_{1M}^l$
…
…
…
$W_{N1}^l$
…
$W_{NM}^l$
Distributor
입력 기준
Collector
출력 기준

# 계산 사례




| X | * | W | = | Y |
|---|---|---|---|---|
| 1 x 2 | * | 2 x 3 | = | 1 x 3 |

$$\begin{pmatrix} 1 & 2 \end{pmatrix} \quad \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} \quad \begin{pmatrix} 9 & 12 & 15 \end{pmatrix}$$

# 층과 가중치

- **뉴런 층과 가중치 층**

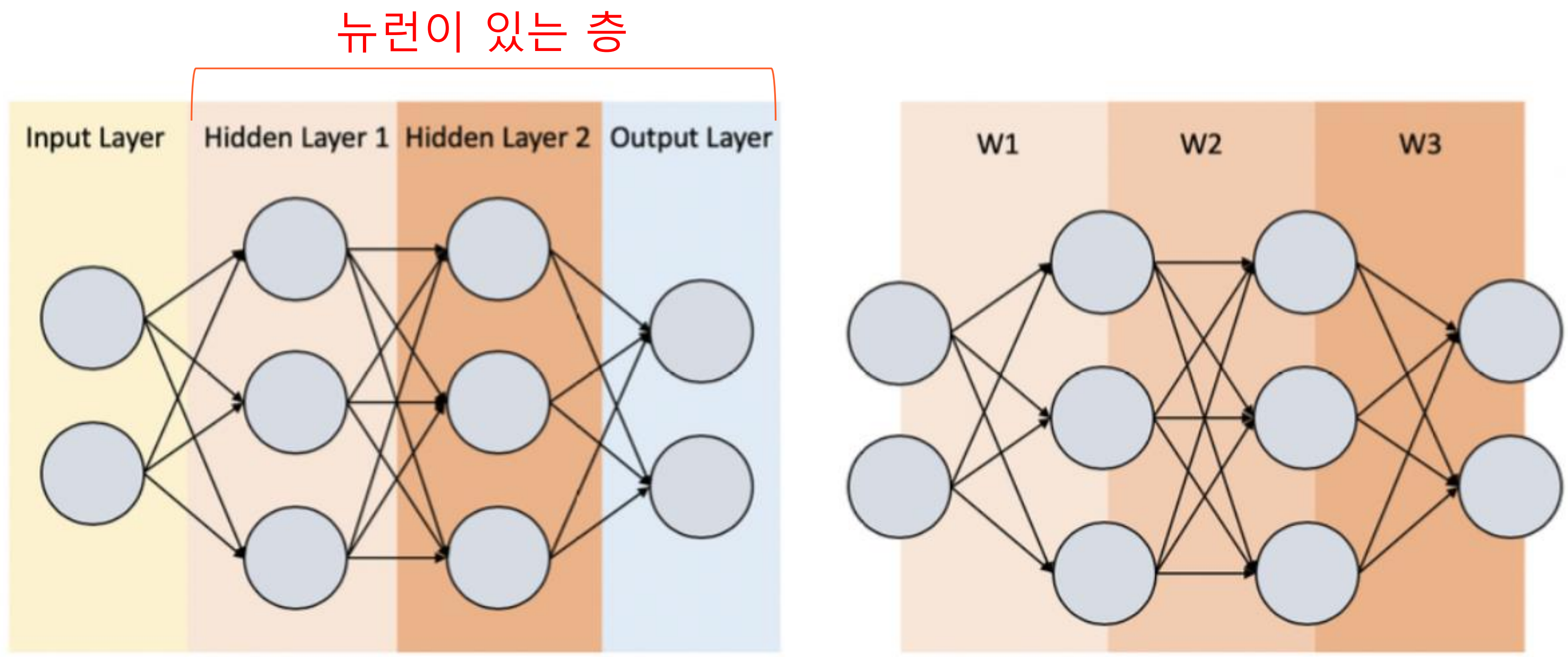
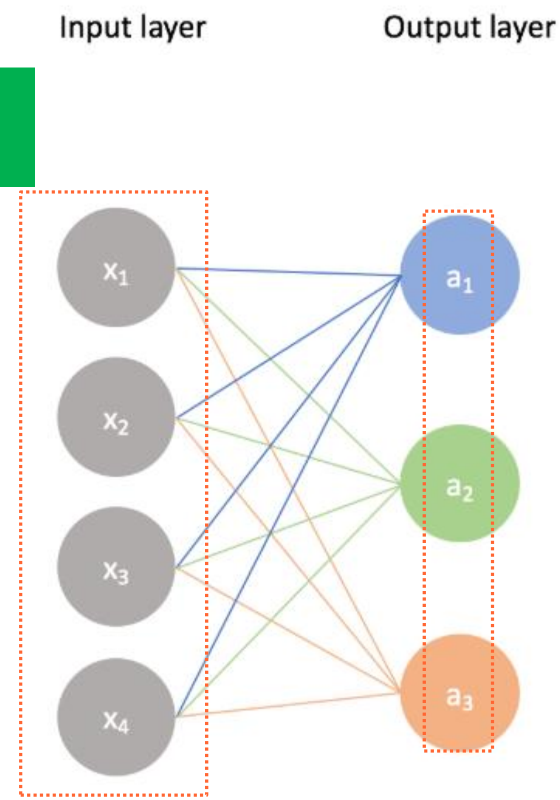



Figure 7. Layers of neuron vs Layers of weights

# 행렬의 다른 표현

- 입력을 오른쪽 행렬에 배치
- 가중치는 왼쪽 행렬에 배치
- 곱의 순서도 변환




## Using multiple observations

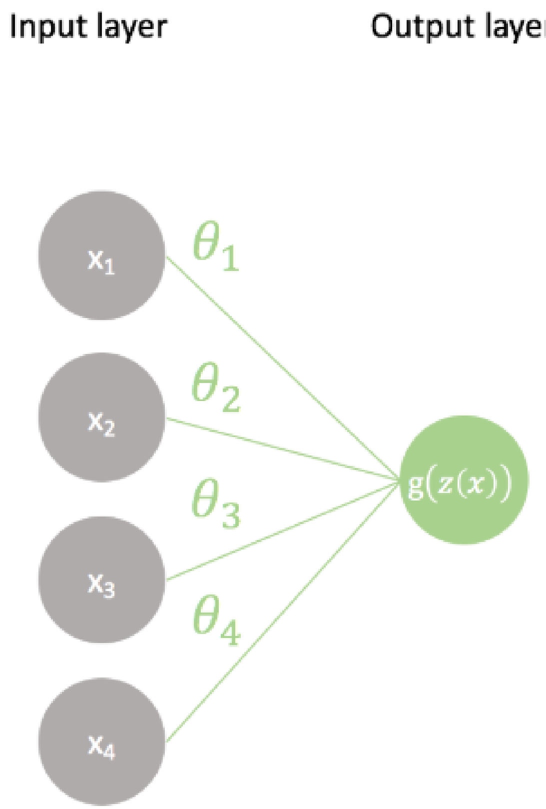
$$\begin{bmatrix} w_1 & w_2 & w_3 & w_4 \\ w_1 & w_2 & w_3 & w_4 \\ w_1 & w_2 & w_3 & w_4 \end{bmatrix} \begin{array}{cccc} \text{Observation 1} & \text{Observation 2} & \text{Observation 3} & \text{Observation 4} \\ \hline \end{array} \begin{bmatrix} x_1 & x_1 & x_1 & x_1 \\ x_2 & x_2 & x_2 & x_2 \\ x_3 & x_3 & x_3 & x_3 \\ x_4 & x_4 & x_4 & x_4 \end{bmatrix} + \begin{bmatrix} b \\ b \\ b \end{bmatrix} \xrightarrow[\textit{activation}]{} \begin{array}{cccc} \text{Observation 1} & \text{Observation 2} & \text{Observation 3} & \text{Observation 4} \\ \hline \end{array} \begin{bmatrix} a_1 & a_1 & a_1 & a_1 \\ a_2 & a_2 & a_2 & a_2 \\ a_3 & a_3 & a_3 & a_3 \end{bmatrix}$$



https://www.jeremyjordan.me/intro-to-neural-networks/

# 하나의 출력 뉴런 연산

- **활성화 함수로 시그모이드 함수 적용**



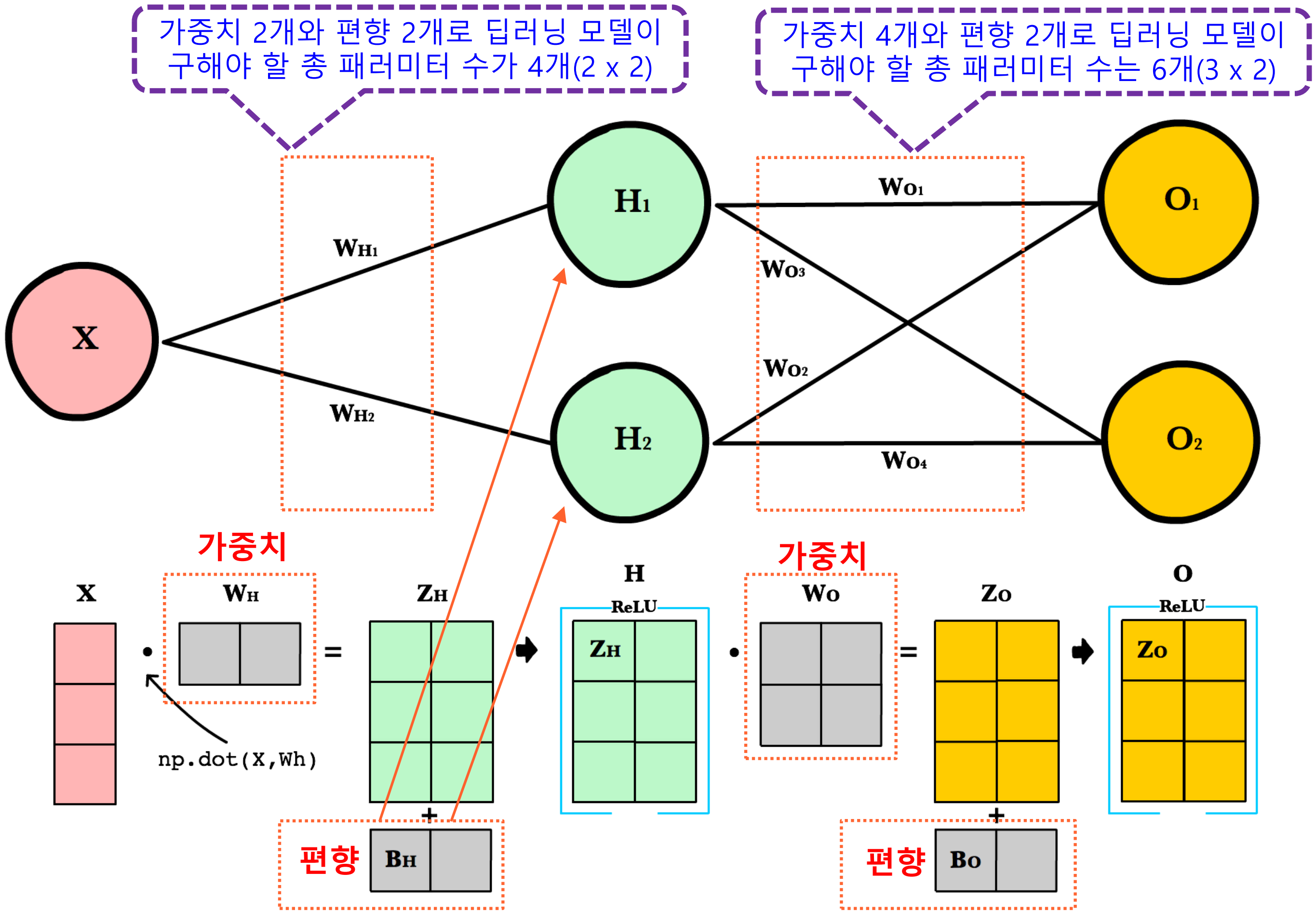

## Logistic Regression

$$\boxed{z(x)} = w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 + b$$

$$g(z) = \frac{1}{1+e^{\boxed{-z}}}$$

# 입력이 하나 중간층과 출력층으로 구성

- **중간층 뉴런 수가 2인 경우**
  - 샘플 수 3

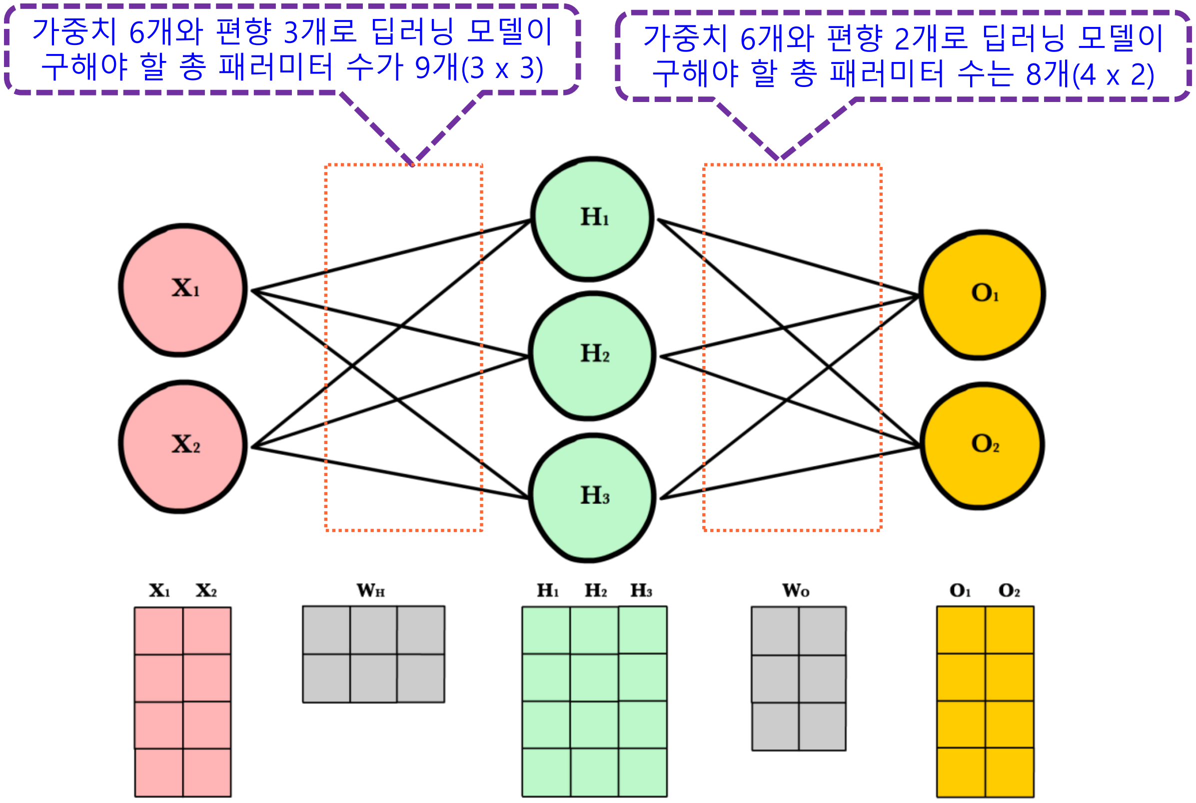





https://ml-cheatsheet.readthedocs.io/en/latest/forwardpropagation.html

# 입력 특징이 2개, 중간층과 출력층

- **중간층의 뉴런 수가 3인 경우**
  - 샘플 수 4






https://ml-cheatsheet.readthedocs.io/en/latest/forwardpropagation.html

# 특징이 2인 이진 분류

- 샘플 수 4