# 딥러닝 기초

BOAZ 분석

# 목차

1. 퍼셉트론

2. 신경망

3. 신경망 학습

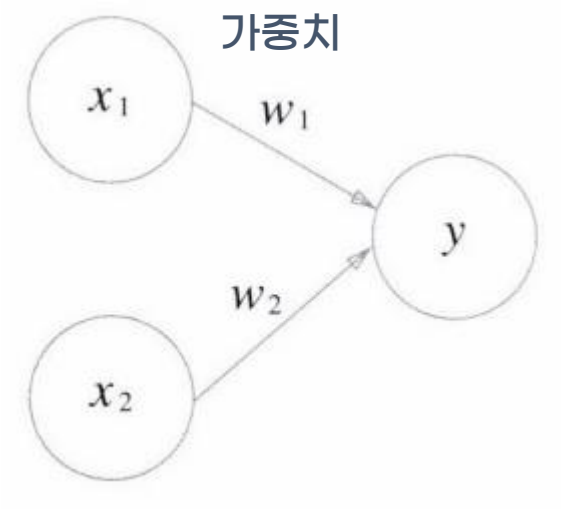
4. 오차역전파법

5. 학습 관련 기술들

## 1. 퍼셉트론 - 퍼셉트론이란?

가중치

다수의 신호 입력




하나의 신호 출력

임계값

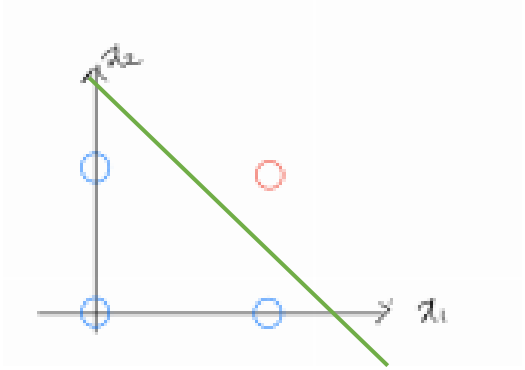
$$y = \begin{cases} 0 \ (w_1 x_1 + w_2 x_2 \leq \theta) \\ 1 \ (w_1 x_1 + w_2 x_2 > \theta) \end{cases}$$

신호의 합이 임계치보다 클 때만 출력

# 1. 퍼셉트론 - 단순한 논리회로

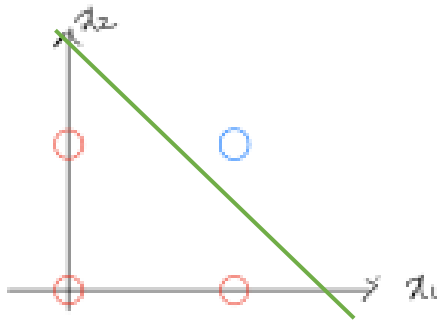
## AND

| $x_1$ | $x_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 1 | 1 |



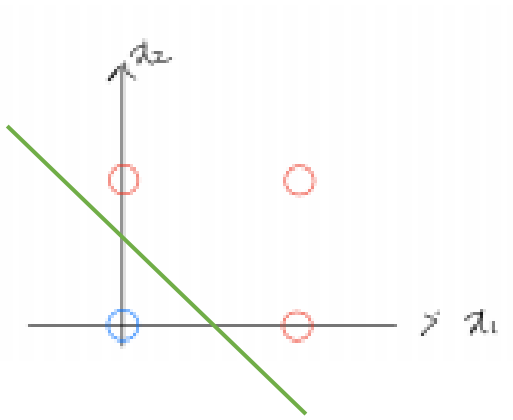

## NAND

| $x_1$ | $x_2$ | $y$ |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 1 | 0 |




## OR

| $x_1$ | $x_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 1 | 1 |



AND

| $x_1$ | $x_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 1 | 1 |

NAND

| $x_1$ | $x_2$ | $y$ |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 1 | 0 |

OR

| $x_1$ | $x_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 1 | 1 |

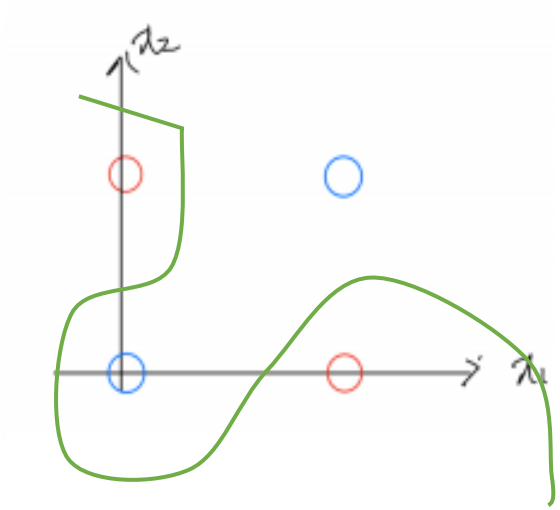
**매개변수 조정만 해주면 퍼셉트론 표현 가능**

**학습 : 적절한 매개변수 값을 정하는 작업**

# XOR 게이트

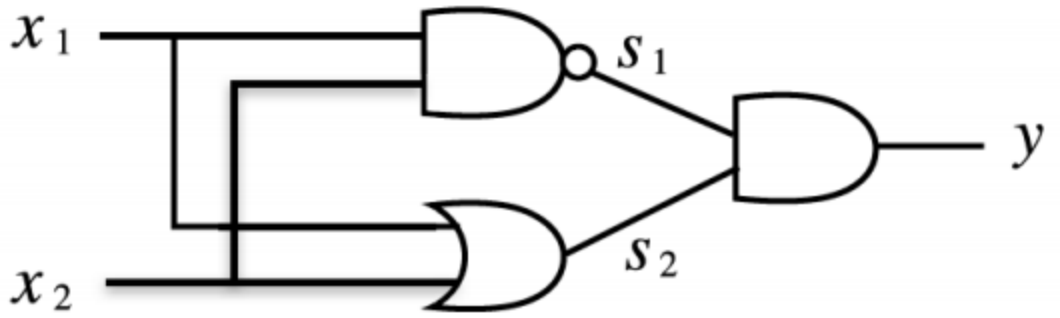
XOR

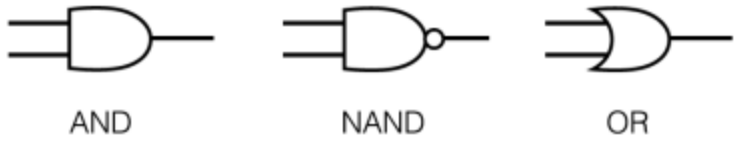
| $x_1$ | $x_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 1 | 0 |

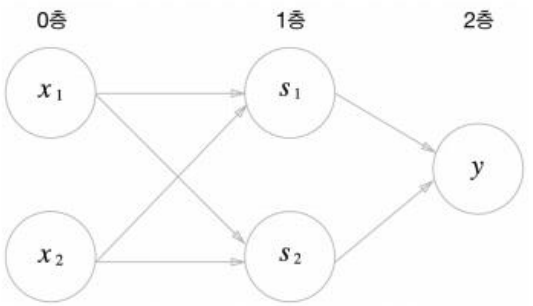


**직선 하나로 나눈 영역(=선형 영역)만 표현할 수 있다는 한계**

# ‘층을 쌓아’ 다층 퍼셉트론을 만든다.



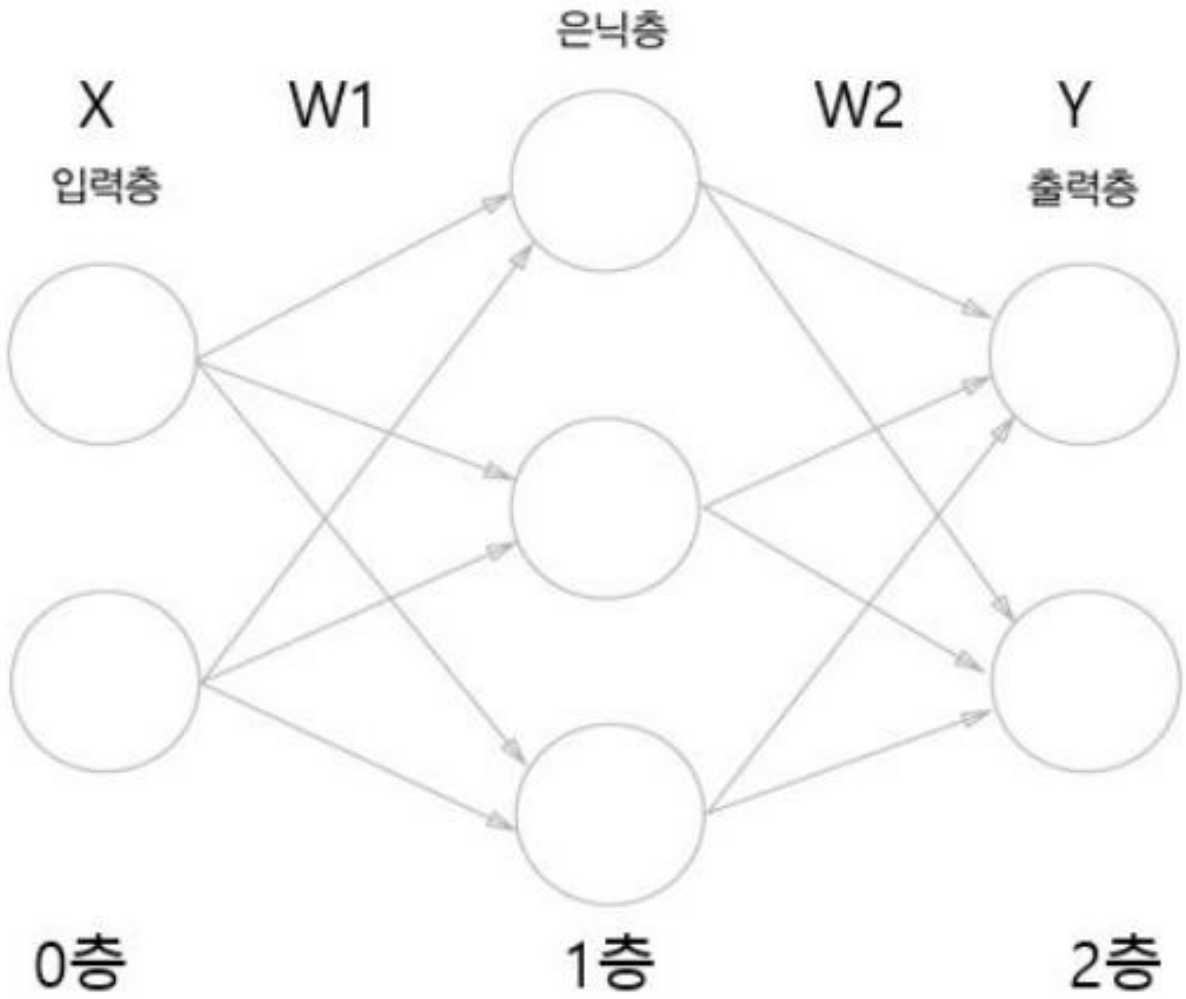



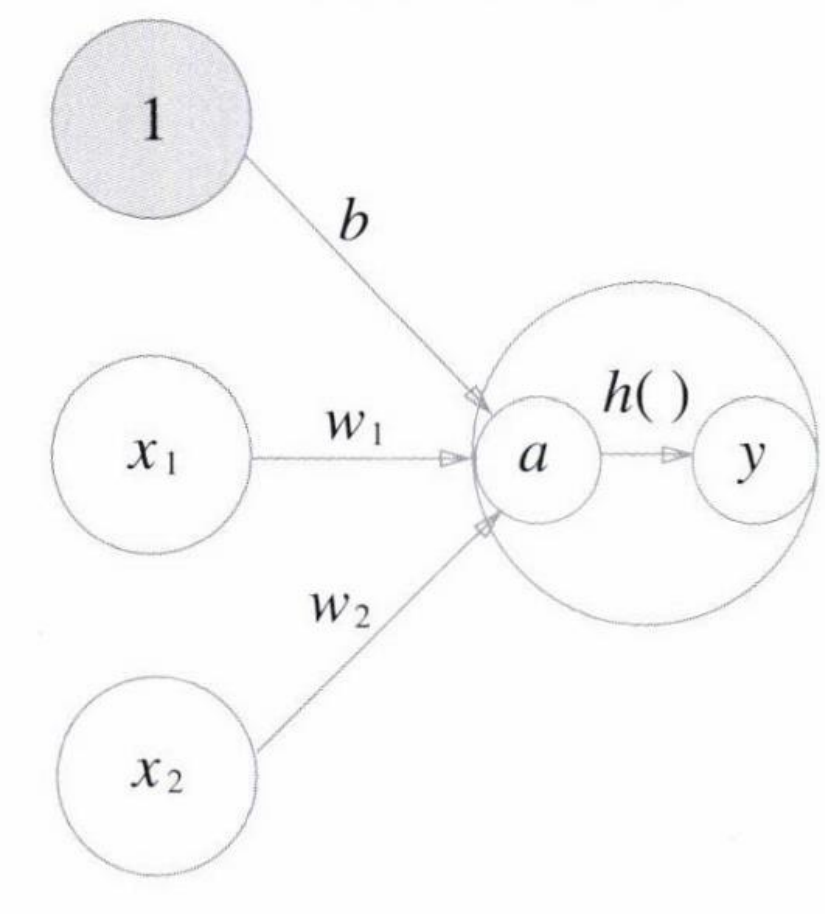

| $x_1$ | $x_2$ | $s_1$ | $s_2$ | $y$ |
|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 |

## → 선형성 극복

## 2. 신경망 - 신경망이란?



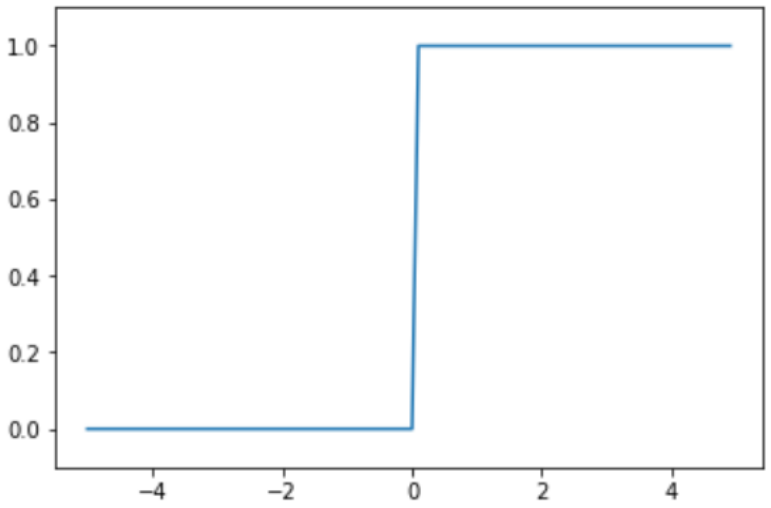
# 활성화 함수 h(x) : 신호의 총합을 출력 신호로 변환하는 함수




데이터를 **비선형**으로
바꿔주기 위해 사용!

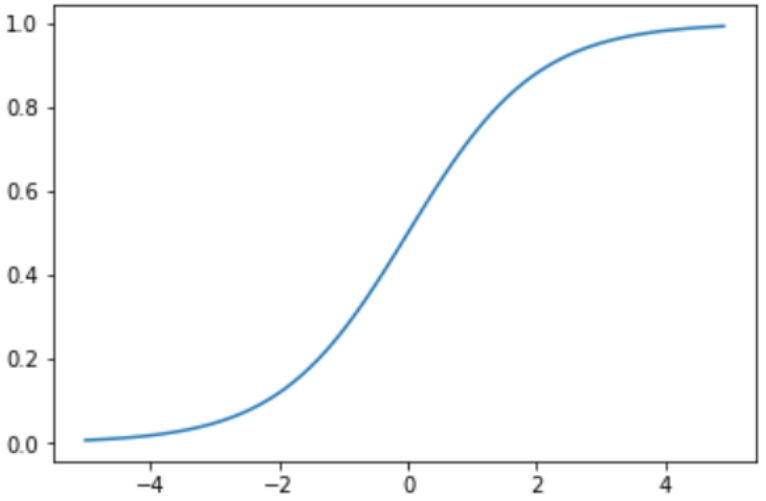
# 활성화 함수 h(x) : 신호의 총합을 출력 신호로 변환하는 함수

## 1. 계단 함수



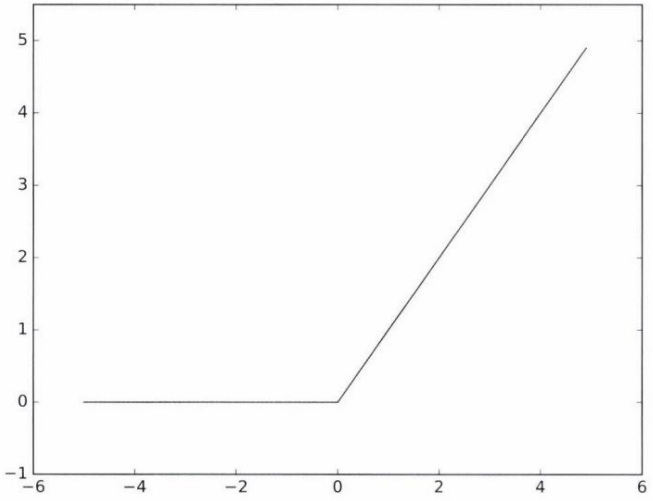

$$y = \begin{cases} 0 & (b + w_1x_1 + w_2x_2 \leq 0) \\ 1 & (b + w_1x_1 + w_2x_2 > 0) \end{cases}$$

## 2. 시그모이드 함수




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

## 3. ReLU 함수



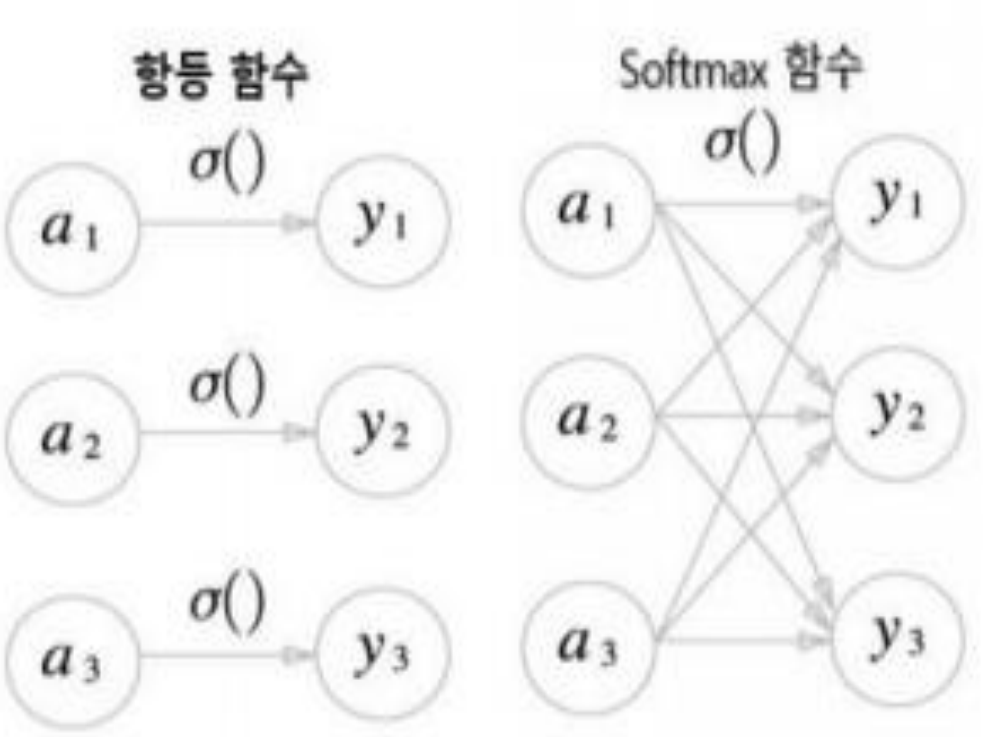

$$h(x) = \begin{cases} x & (x > 0) \\ 0 & (x \leq 0) \end{cases}$$

# 활성화 함수 h(x) : 신호의 총합을 출력 신호로 변환하는 함수

출력층의 활성화 함수는 풀고자 하는 문제의 성질에 맞게 정의

## 4. 항등 함수

: 출력 범위에 제한이 없고 연속적이기 때문에, 연속적인 수치를 예측하는 **회귀 문제를 다룰 때 적합**



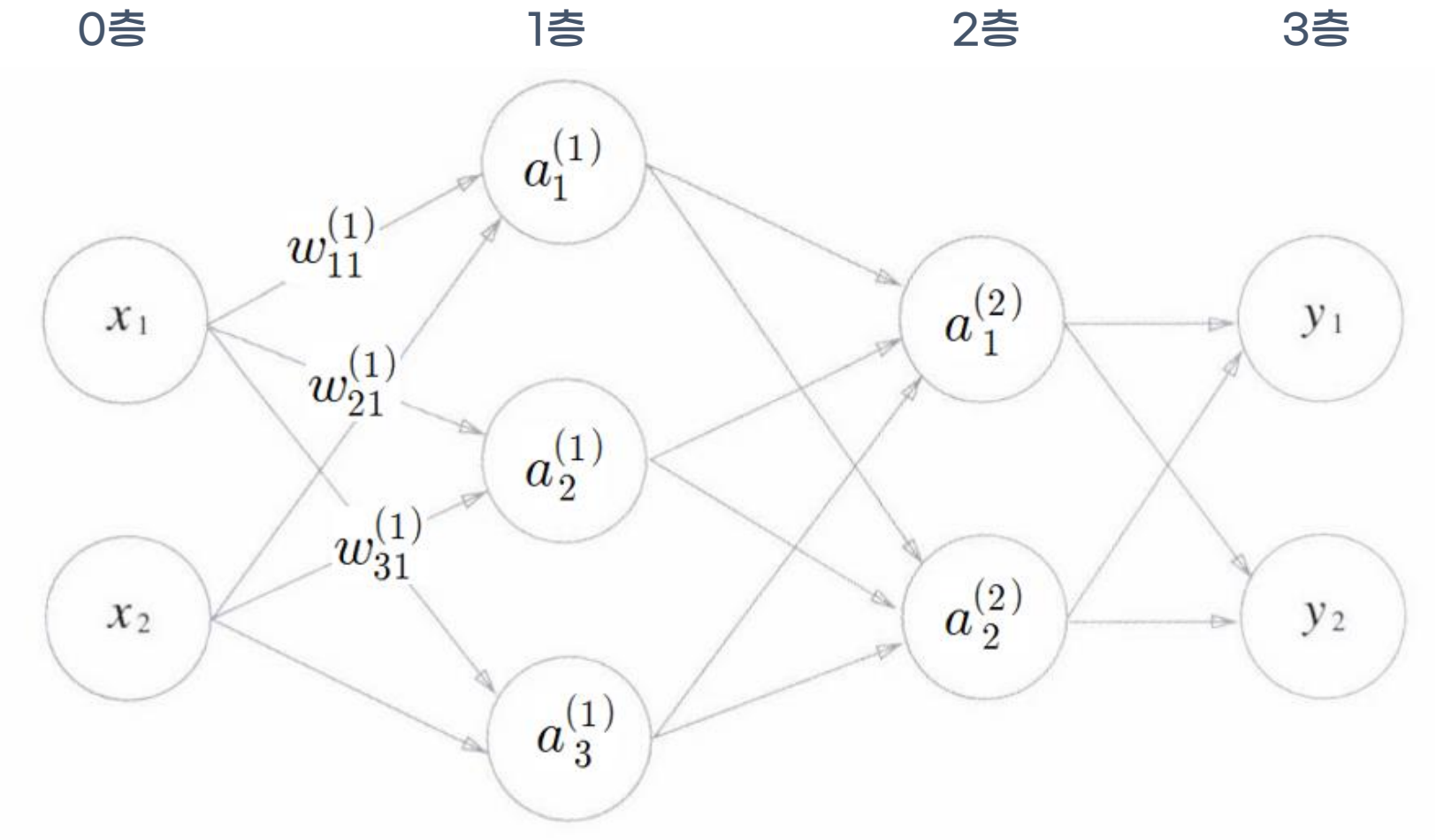

## 5. Softmax 함수

: 출력값을 양수로 바꾸고 모든 출력값들의 합이 1이 되도록 nomalize

: 확률로 해석 가능

: **분류 문제를 다룰 때 적합**

## 2. 신경망 - 3층 신경망 구현하기




$w_{12}^{(1)}$

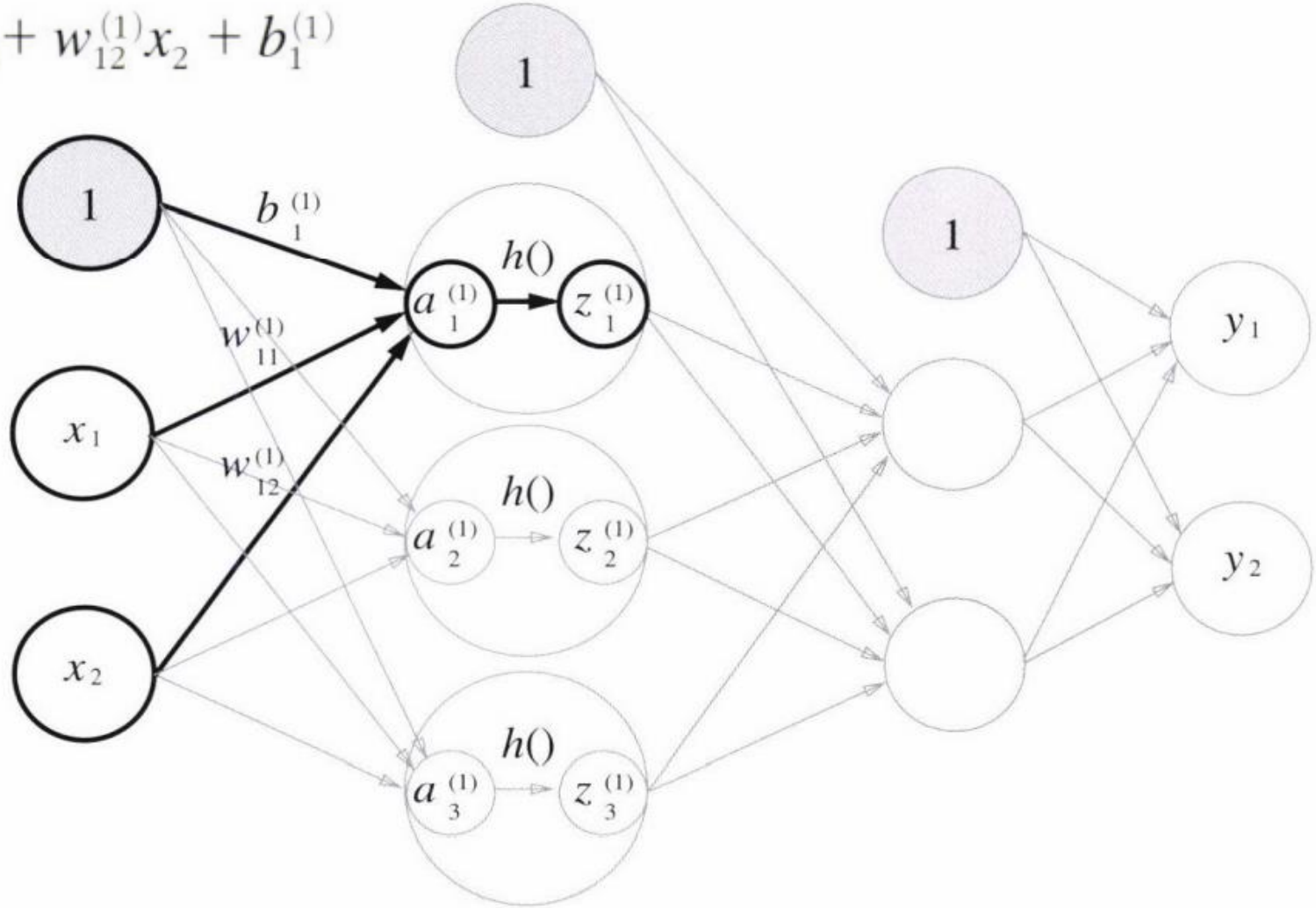
- 1층의 가중치
- 앞 층의 2번째 뉴런
- 다음 층의 1번째 뉴런

$a_1^{(2)}$ 2층의 뉴런

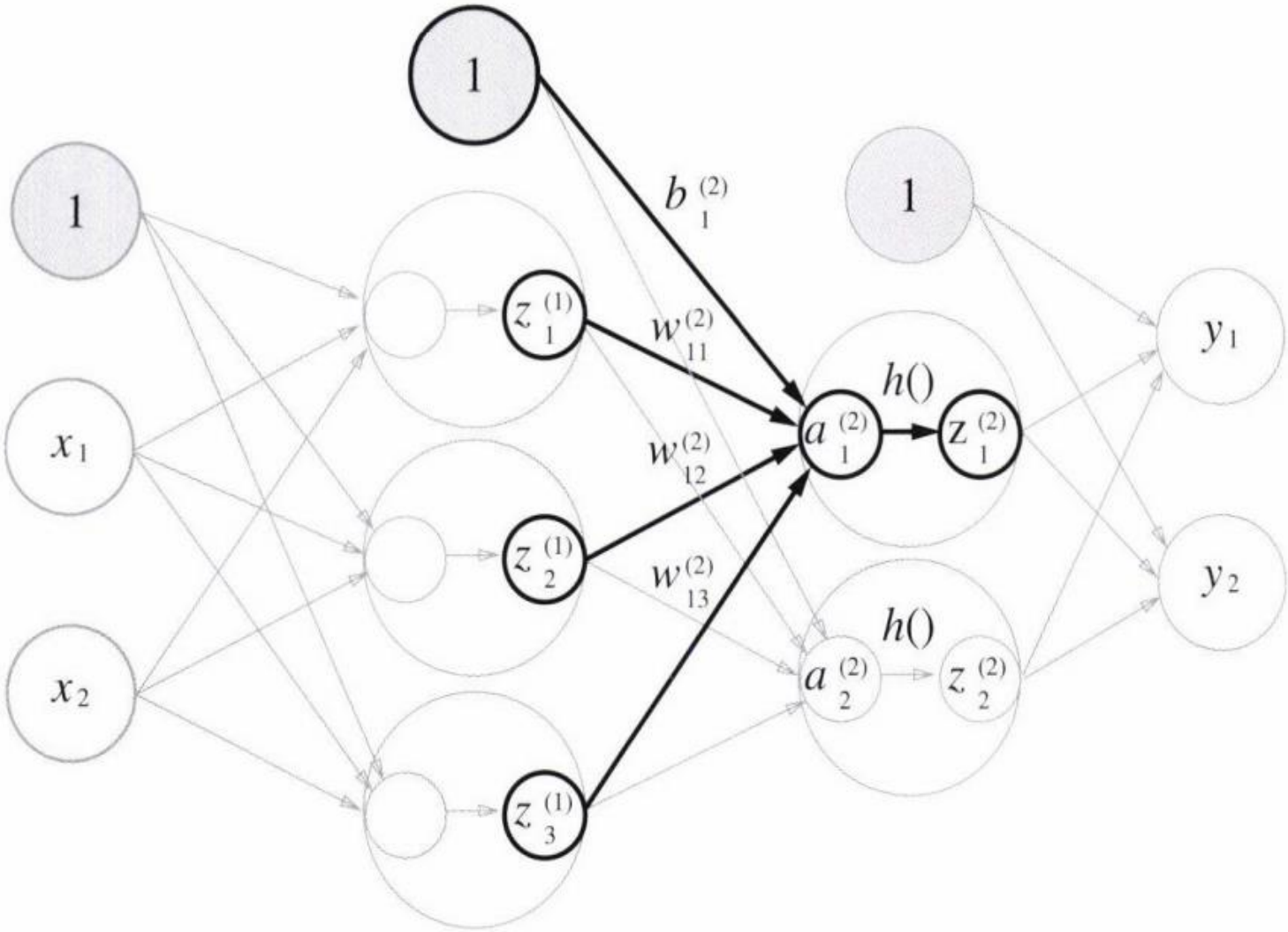
(편향 제외)
1번째 노드

## 2. 신경망 - 3층 신경망 구현하기

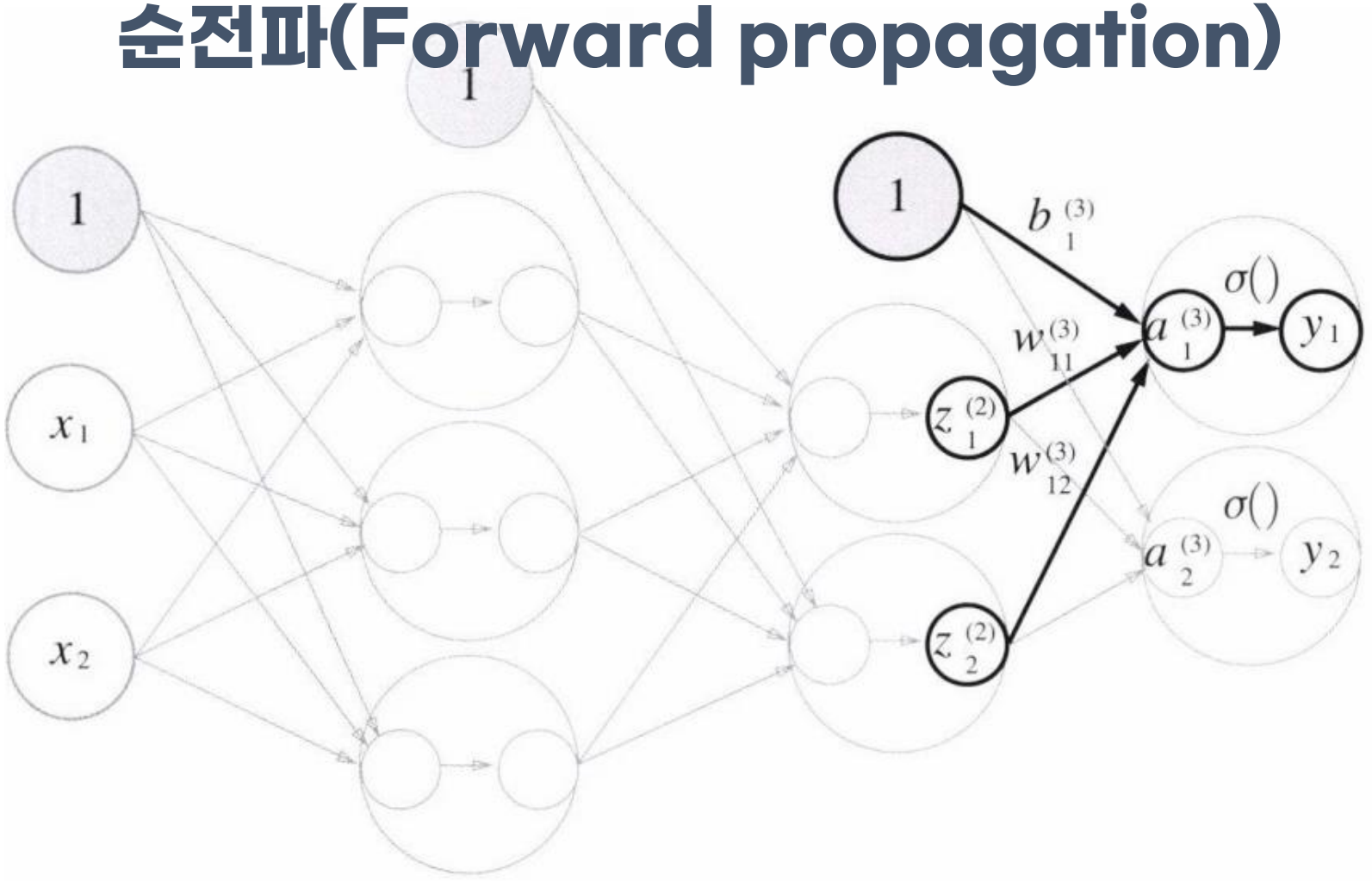
$$a_1^{(1)} = w_{11}^{(1)} x_1 + w_{12}^{(1)} x_2 + b_1^{(1)}$$



## 2. 신경망 - 3층 신경망 구현하기



# 순전파(Forward propagation)



# 손실 함수(Loss function) : 신경망이 학습할 수 있도록 해주는 지표

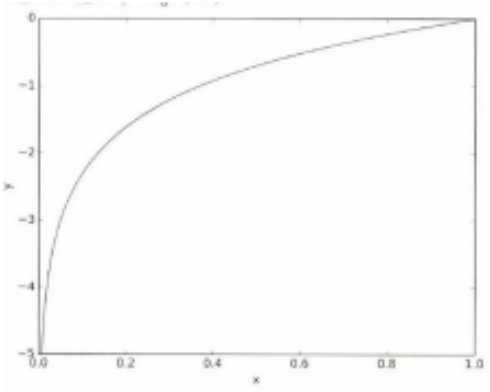
## 손실 함수의 결과값을 가장 적게 만드는 가중치 매개 변수를 찾는 것이 학습의 목표

1. 평균 제곱 오차
   (MSE, mean squared error)

$$E = \frac{1}{2}\sum_k (y_k - t_k)^2$$

2. 교차 엔트로피 오차
   (CEE, cross entropy error)

$$E = -\sum_k t_k \log y_k$$



# 손실 함수(Loss function) : 신경망이 학습할 수 있도록 해주는 지표

Q. 왜 '정확도' 대신 '손실 함수'를 신경망 학습 기준 지표로 사용할까?

A. 정확도는 불연속적인 수치 & 미소한 변화에는 반응 X -> 정확도를 지표로 하면 매개변수의 미분이 대부분의 장소에서 0이 되기 때문!

이는 신경망에서 대부분의 장소에서 미분값이 0이 되는 계단 함수를 활성화 함수로 사용하지 않고, 매끄러운 함수를 사용하는 것과 같은 이유

## 신경망의 손실함수를 작게 만드는 기법으로 함수의 기울기를 활용

# 경사법 : 현 위치에서 기울어진 방향으로 일정 거리만큼 이동하여 함수의 값(=손실 함수 값)을 점차 줄이는 방법

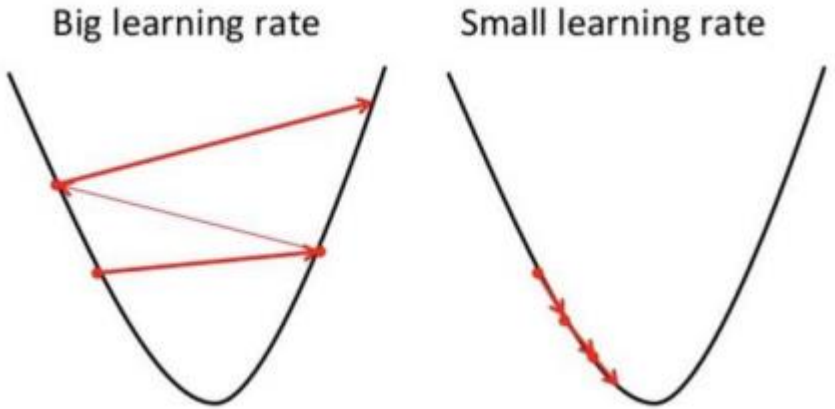
경사 하강법(gradient descent method) : 최소값 찾기

경사 상승법(gradient ascent method) : 최대값 찾기

$$x_0 = x_0 - \eta \frac{\partial f}{\partial x_0}$$

$$x_1 = x_1 - \eta \frac{\partial f}{\partial x_1}$$

* 에타(η) = 학습률(learning rate) : 한번에 얼마나 학습해야 할지. 매개변수 값을 얼마나 갱신해야 할지를 나타냄



# 계산 그래프 : 계산 과정을 그래프(노드, 엣지)로 표현한 것

사용하는 이유?

1. 복잡한 문제를 자신과 관계된 정보만으로 결과를 출력 가능
2. 중간 계산 결과를 모두 보관할 수 있음
3. 역전파를 통해 미분을 효율적으로 계산

# 계산 그래프 : 계산 과정을 그래프(노드, 엣지)로 표현한 것

1. 덧셈 노드의 역전파

: 입력 값을 그대로 다음 노드로 흘려 보냄

2. 곱셈 노드의 역전파

: 상류의 값에 순전파 때의 입력 신호들을 서로 바꾼 값을 곱해 하류로 보냄

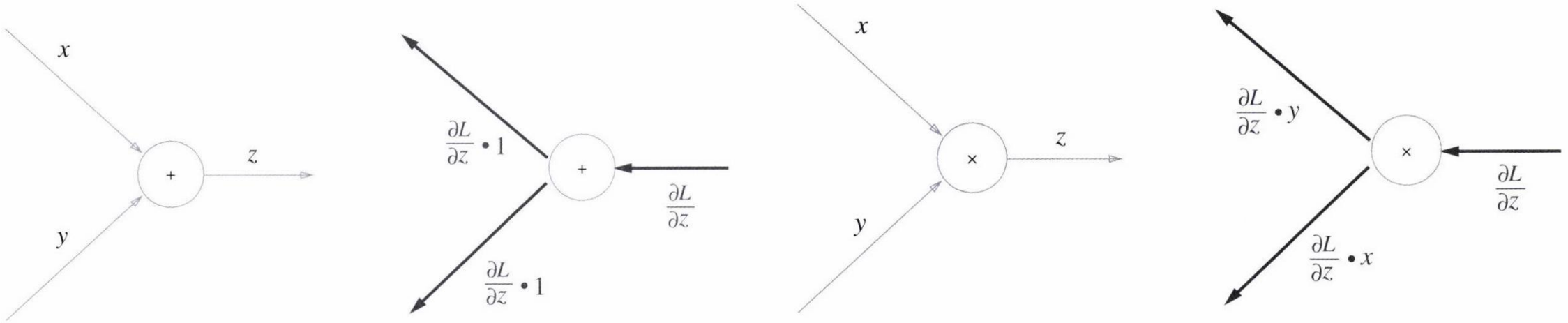
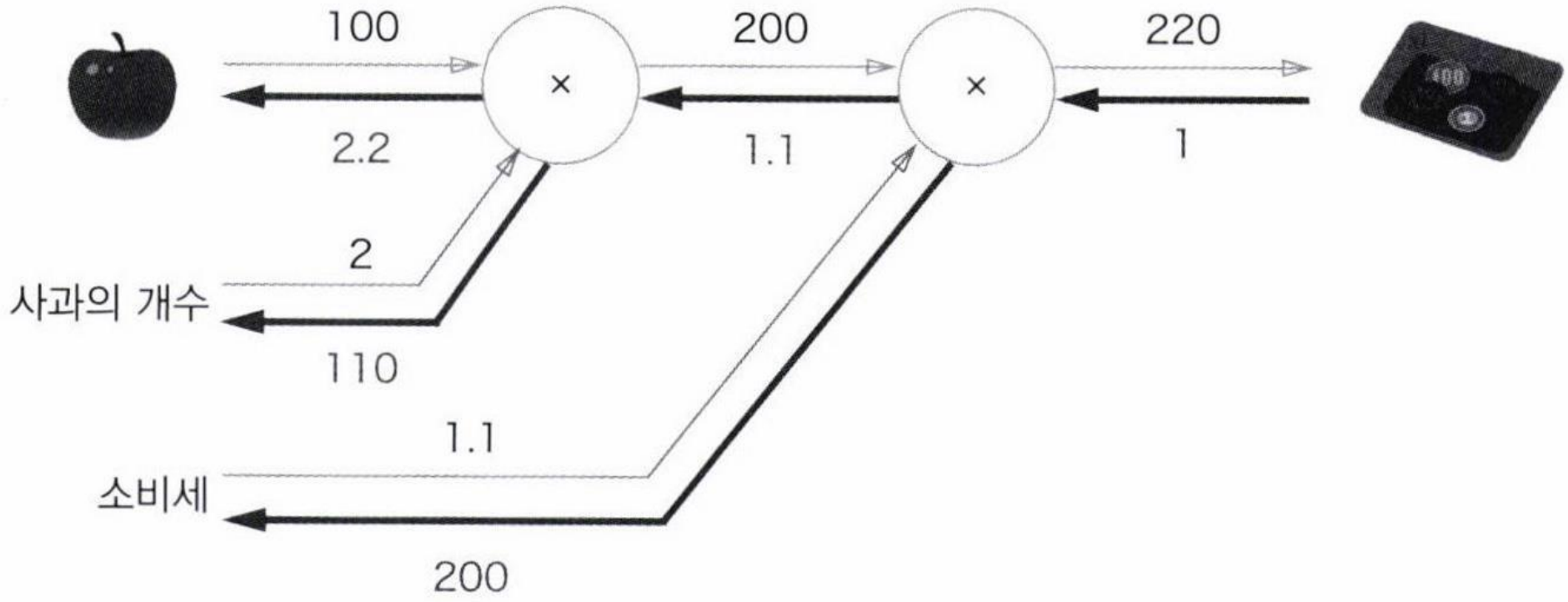


**그림 5-14** 사과 쇼핑의 역전파 예

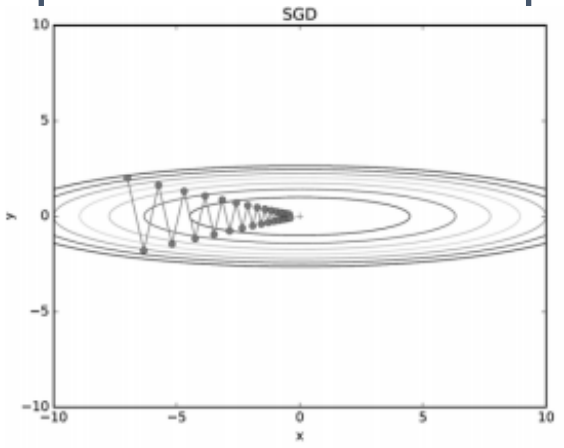


# 5. 학습 관련 기술들 - 매개변수 갱신

## ① 확률적 경사 하강법 (SGD)

$$\mathbf{W} \leftarrow \mathbf{W} - \eta \frac{\partial L}{\partial \mathbf{W}}$$

- 데이터를 무작위로 선정하여 경사 하강법을 적용하는 매개변수 갱신 방법
- 추출된 데이터 한 개에 대해서 그라디언트를 계산
- 단점 : 비등방성 함수에서 탐색 경로가 비효율적

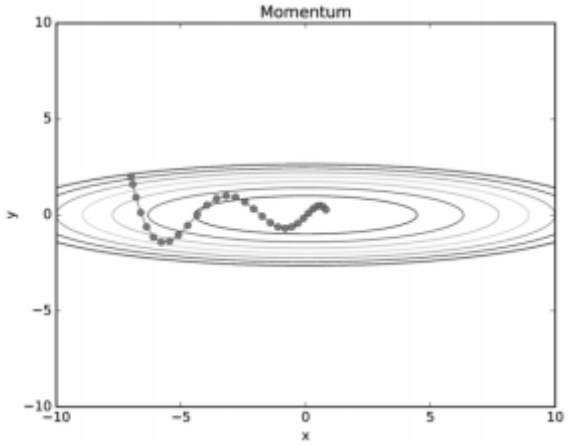



## ② Momentum

$$\mathbf{v} \leftarrow \alpha\mathbf{v} - \eta \frac{\partial L}{\partial \mathbf{W}}$$

$$\mathbf{W} \leftarrow \mathbf{W} + \mathbf{v}$$

- 확률적 경사 하강법에 속도의 개념의 더함
- SGD 단점 개선

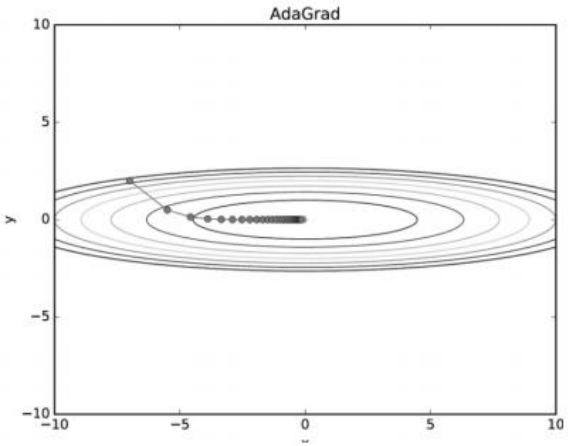



## ③ AdaGrad

$$\mathbf{h} \leftarrow \mathbf{h} + \frac{\partial L}{\partial \mathbf{W}} \odot \frac{\partial L}{\partial \mathbf{W}}$$

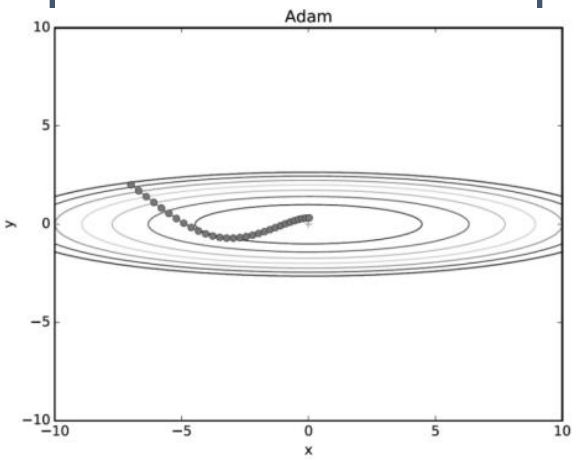
$$\mathbf{W} \leftarrow \mathbf{W} - \eta \frac{1}{\sqrt{\mathbf{h}}} \frac{\partial L}{\partial \mathbf{W}}$$

- 학습을 진행하면서 학습률을 점차 줄이는 '학습률 감소 기법'을 적용
- 개별 매개변수에 적응적으로 학습률을 조정하면서 학습 진행




## ④ Adam

- momentum + AdaGad

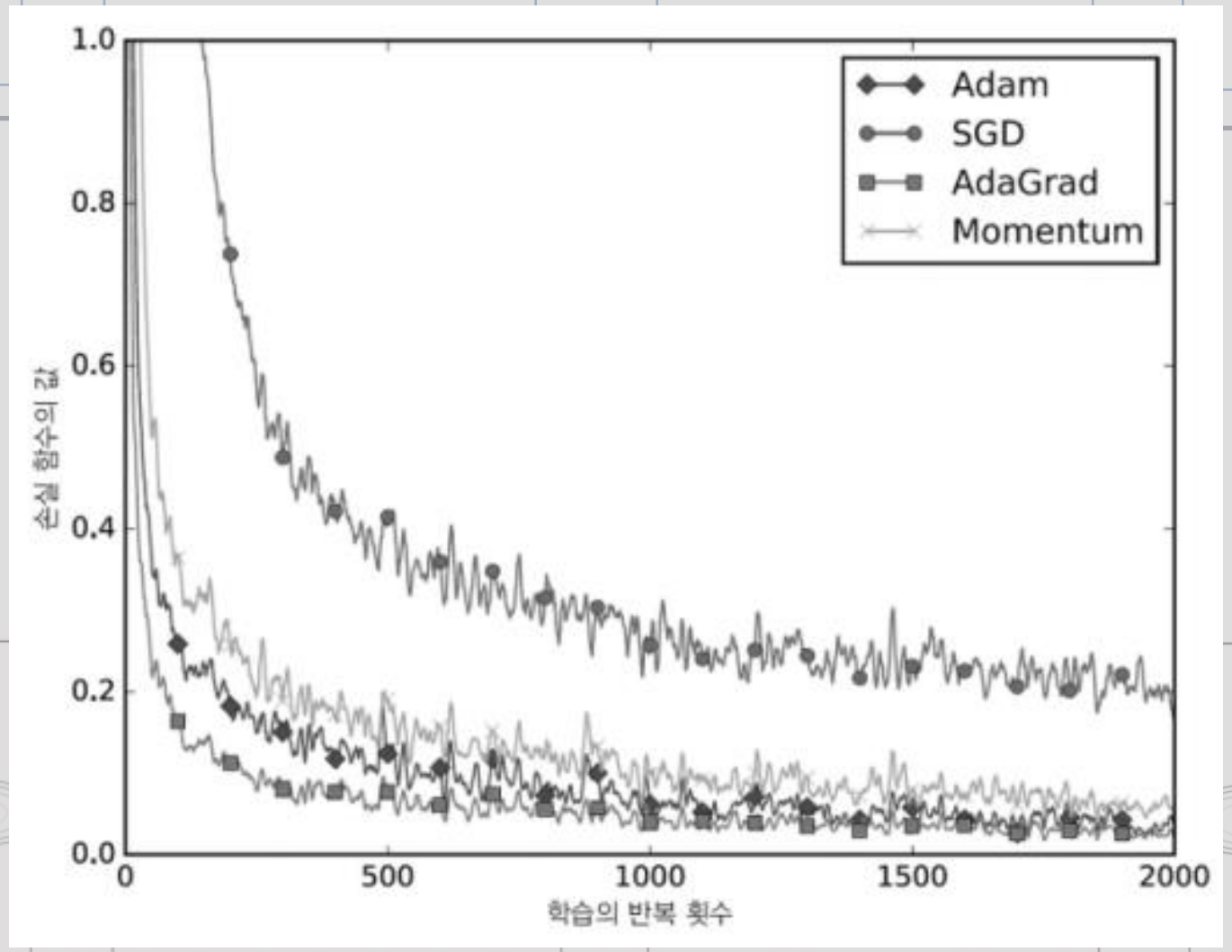


# 5. 학습 관련 기술들 - 매개변수 갱신

① 확률적 경사 하강법 (SGD)

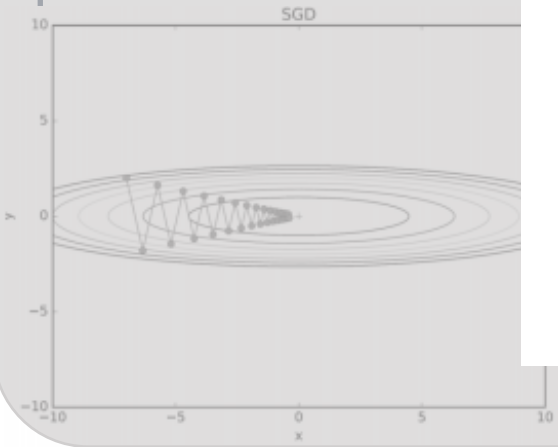
$$W \leftarrow W - \eta \frac{\partial L}{\partial W}$$

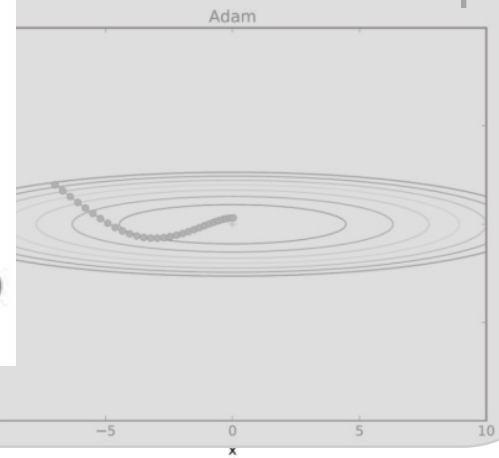
- 데이터를 무작위로 선정하여 경사 하강법을 적용하는 매개변수 갱신 방법
- 추출된 데이터 한 개에 대해서 그라디언트를 계산
- 단점 : 비등방성 함수에서 탐색 경로가 비효율적




④ Adam

momentum + AdaGad



- 가중치의 초기값은 무작위로!

그림 6-10 가중치를 표준편차가 1인 정규분포로 초기화할 때의 각 층의 활성화값 분포

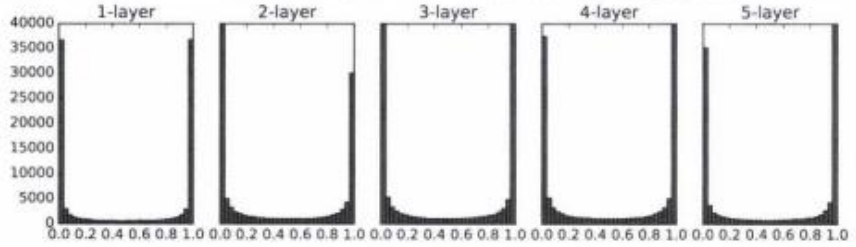
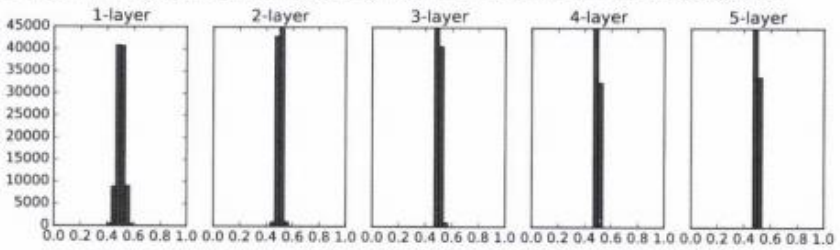
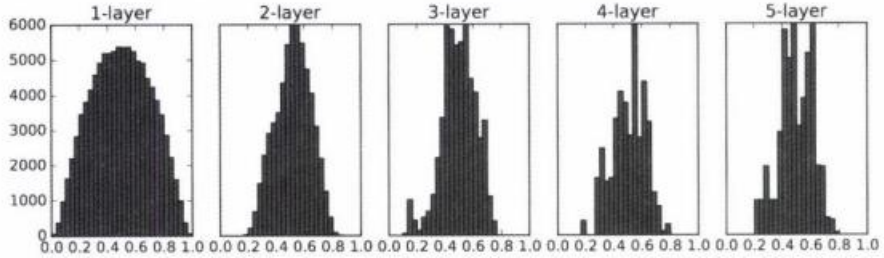



그림 6-11 가중치를 표준편차가 0.01인 정규분포로 초기화할 때의 각 층의 활성화값 분포

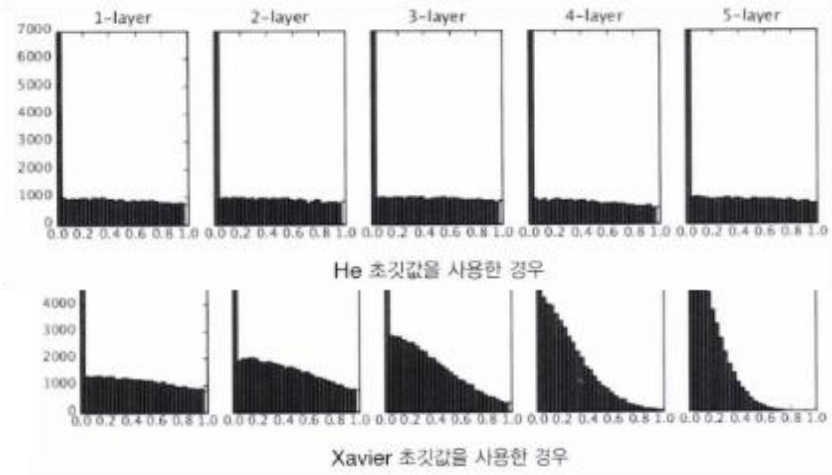



- Xavier 초깃값

그림 6-13 가중치의 초깃값으로 'Xavier 초깃값'을 이용할 때의 각 층의 활성화값 분포




- He 초깃값



# Batch Normalization

**Input:** Values of $x$ over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: $\gamma, \beta$
**Output:** $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma^2_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma^2_{\mathcal{B}} + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma \widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

- 학습하는 과정을 전체적으로 안정화 시키는 방법

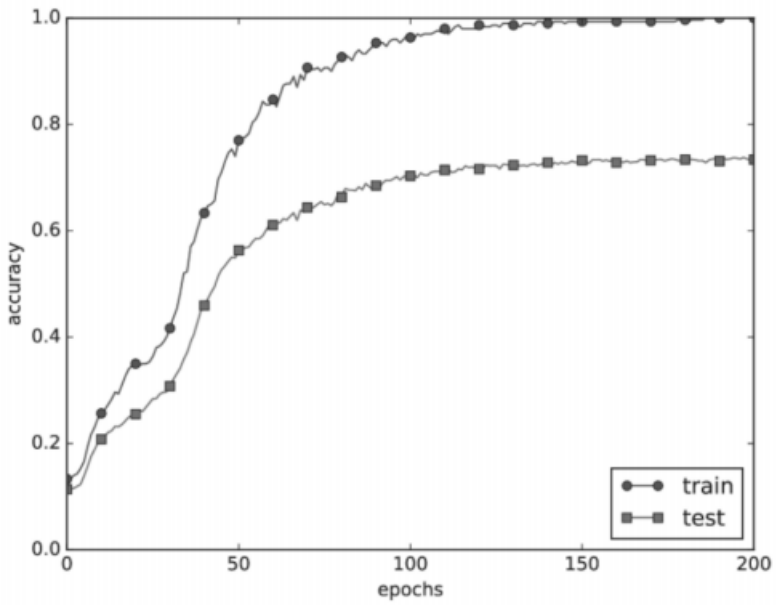
- 높은 학습률, 빠른 속도

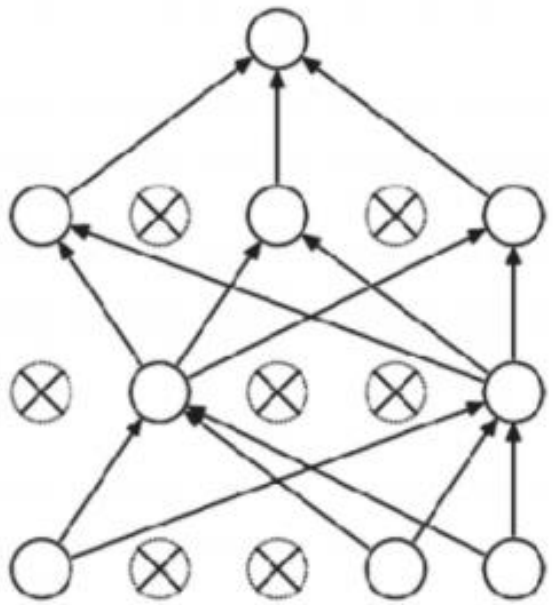
- 초기값 영향 감소

- 규제의 효과 -> 오버피팅 억제

- 감마(scale)와 베타(shift) 조정이 가능

: 비선형성 유지, saturation 현상 조절



**- 오버피팅**

: 매개변수가 많고 표현력이 높을 때, 훈련 데이터가 적을 때 발생

: 해결 방법) ① **가중치 감소**

: 큰 가중치에 대해 큰 패널티 부과

② **드롭 아웃**

: 뉴런을 임의로 삭제하며 학습

- 훈련 데이터 : 매개변수 학습 /

검증 데이터 : 하이퍼파라미터 성능 평가 /

시험 데이터 : 신경망의 범용 성능 평가

- 하이퍼파라미터의 최적값이 존재하는 범위를 조금씩 줄여나감

# 감사합니다