

Artificial Intelligence Practice 1

Seongho Choi

**Biointelligence Laboratory
Department of Computer Science and Engineering,
Seoul National University
March 22, 2018**

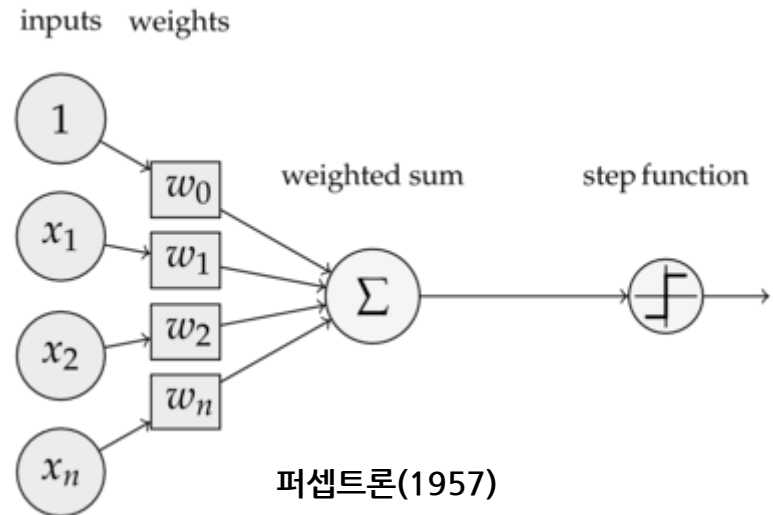
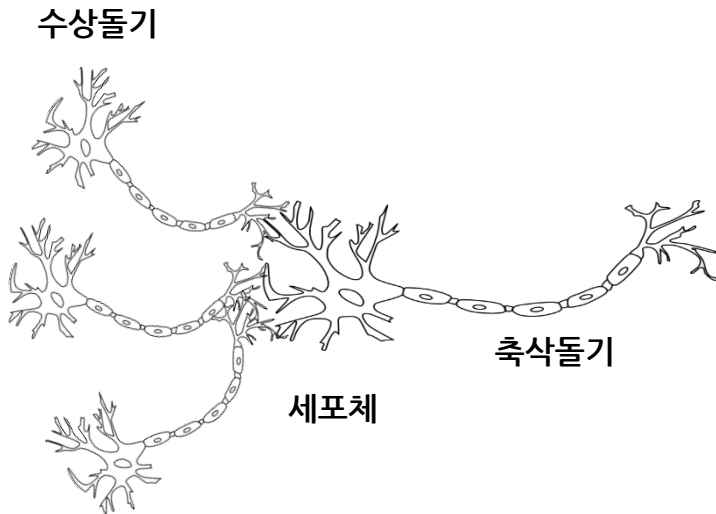


BI
BIOINTELLIGENCE

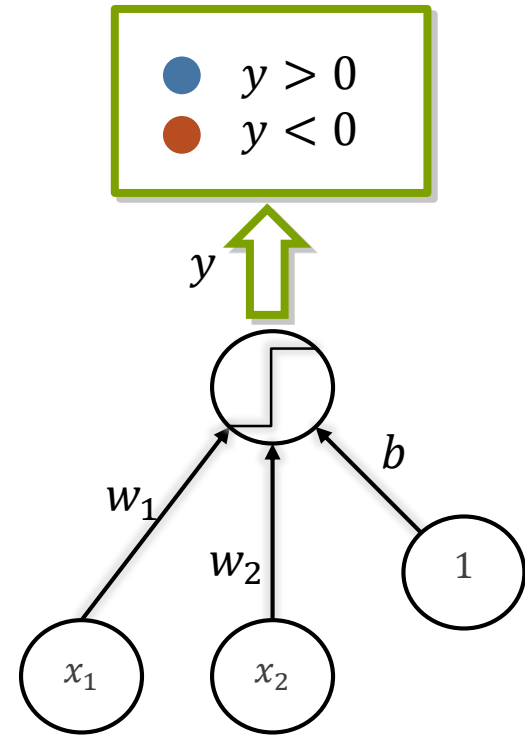
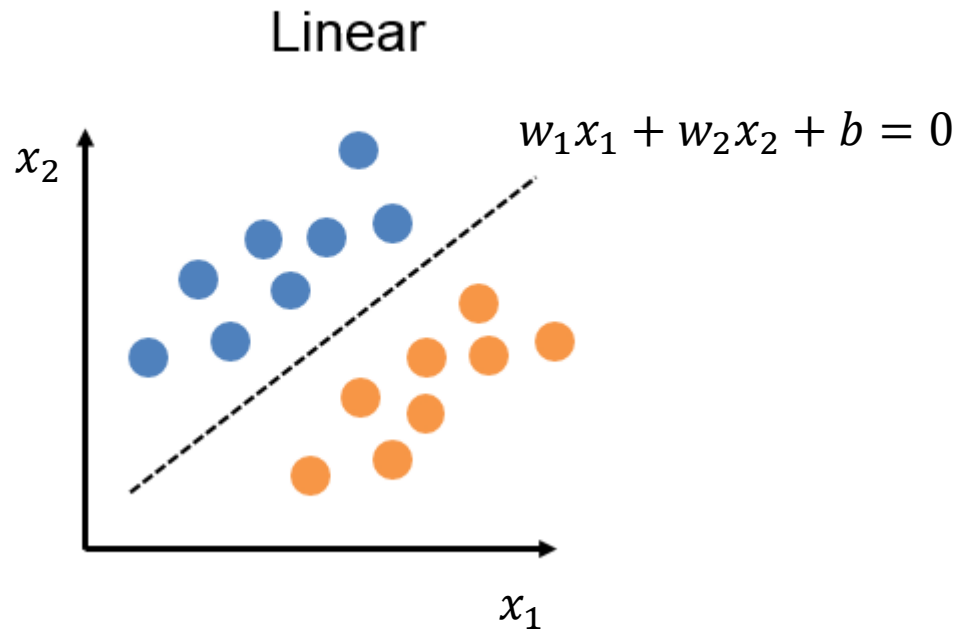
알파고 논문에 등장하는 “network”에 대해서

Neural Networks

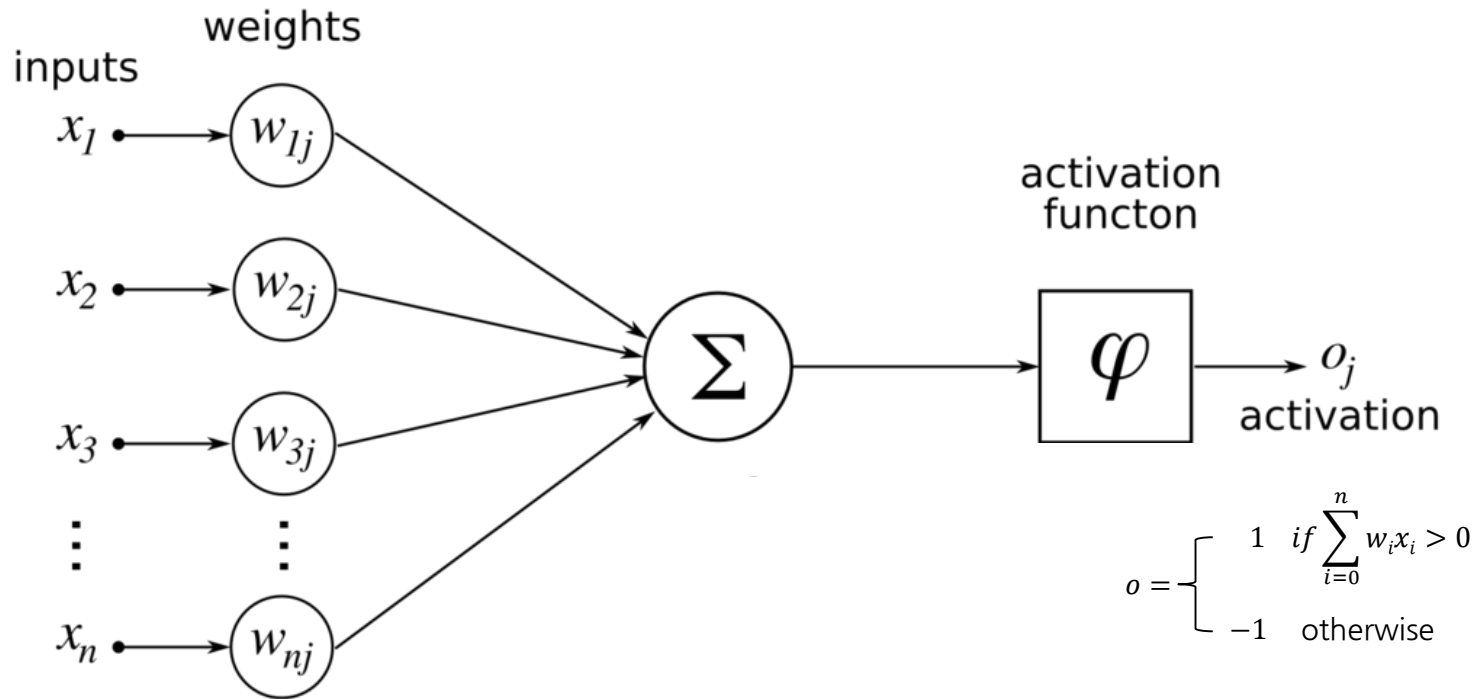
신경망의 신경생물학적 관점



신경망의 수학적 관점



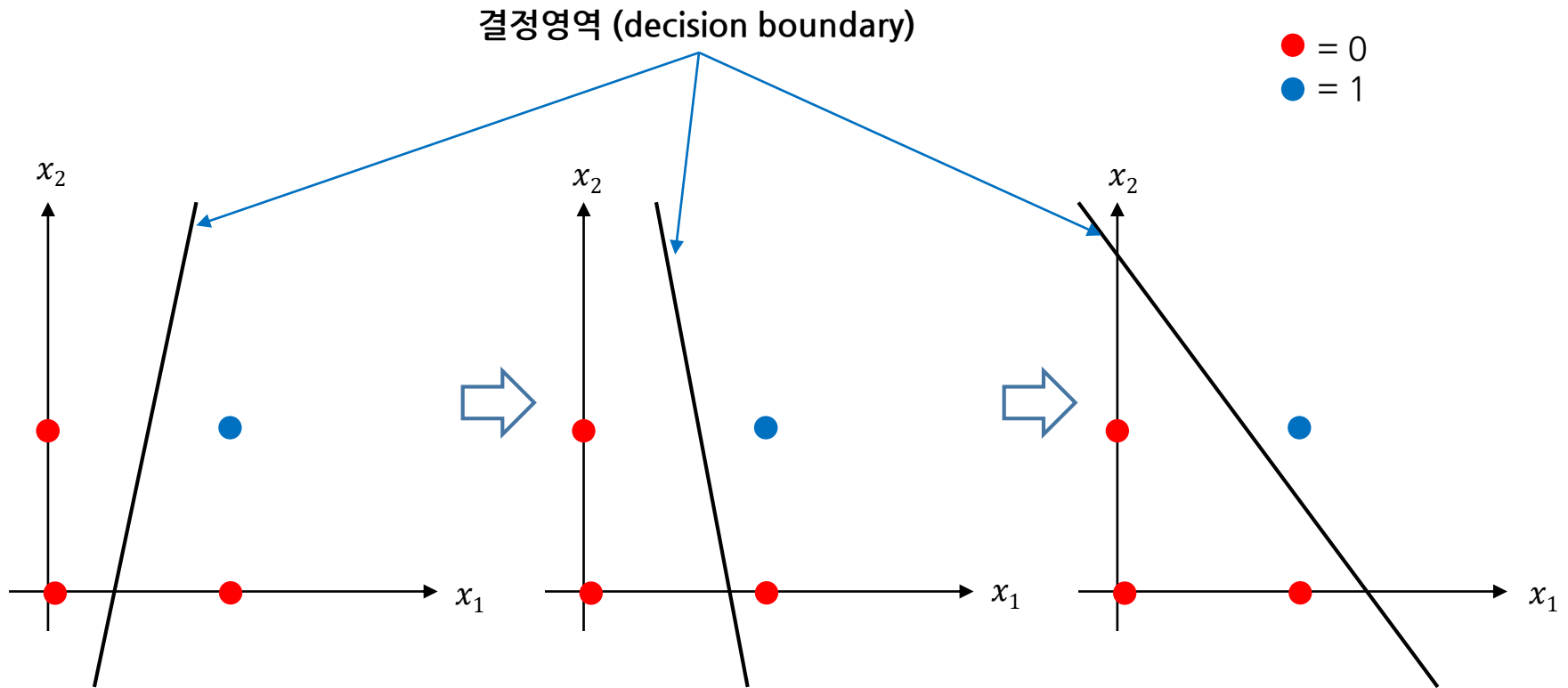
퍼셉트론 (1957)



- 입력: 실수 값을 갖는 벡터(\mathbf{x})
- 모델 구성: 연결 가중치(w) & 임계치
- 출력: 1 또는 -1
- 학습: 알맞은 연결 가중치를 탐색

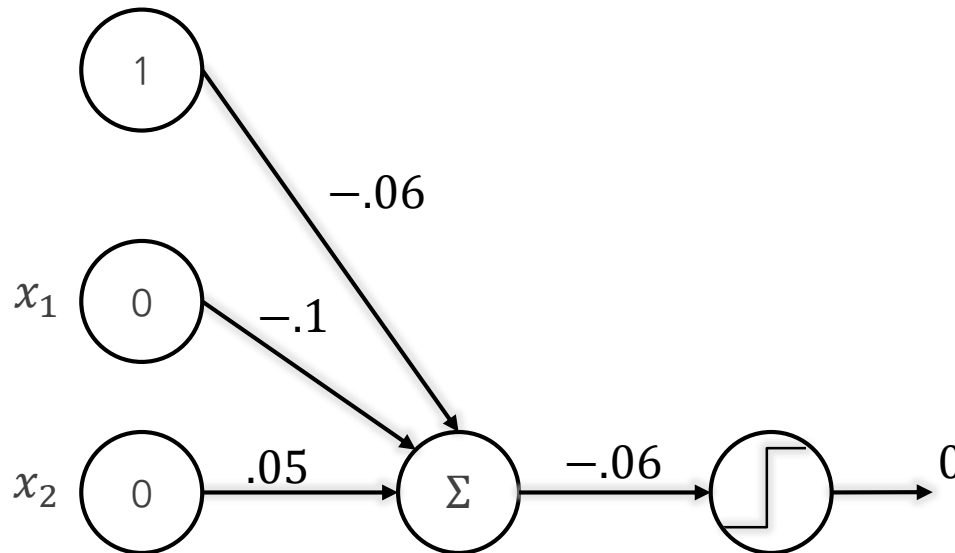
퍼셉트론의 학습 방법

- 퍼셉트론의 학습은 “적절한 연결 가중치”를 찾는 것
 - 적절한 연결 가중치를 가지게 되면 결정영역을 바꿀 수 있음



퍼셉트론의 학습 방법

- 퍼셉트론의 학습은 “적절한 연결 가중치”를 찾는 것

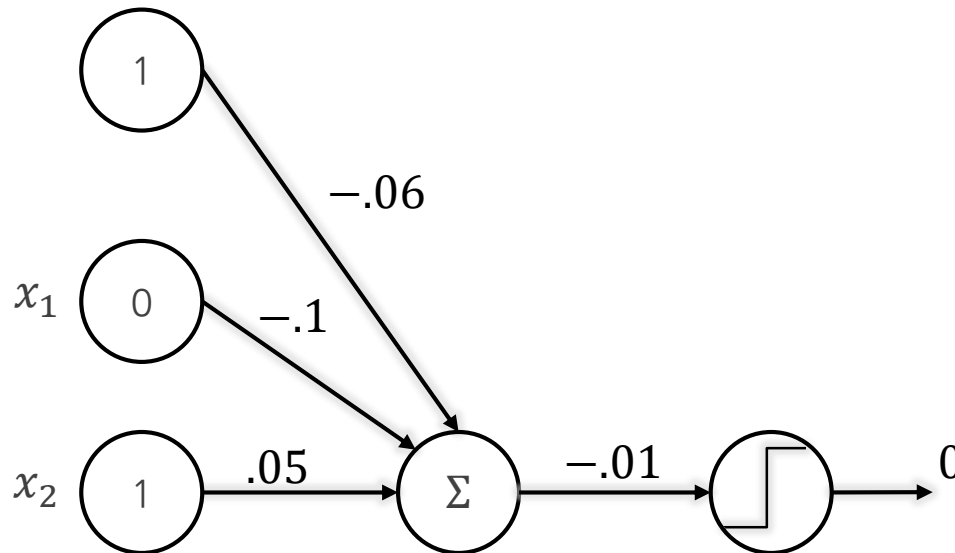


AND 연산

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

퍼셉트론의 학습 방법

- 퍼셉트론의 학습은 “적절한 연결 가중치”를 찾는 것

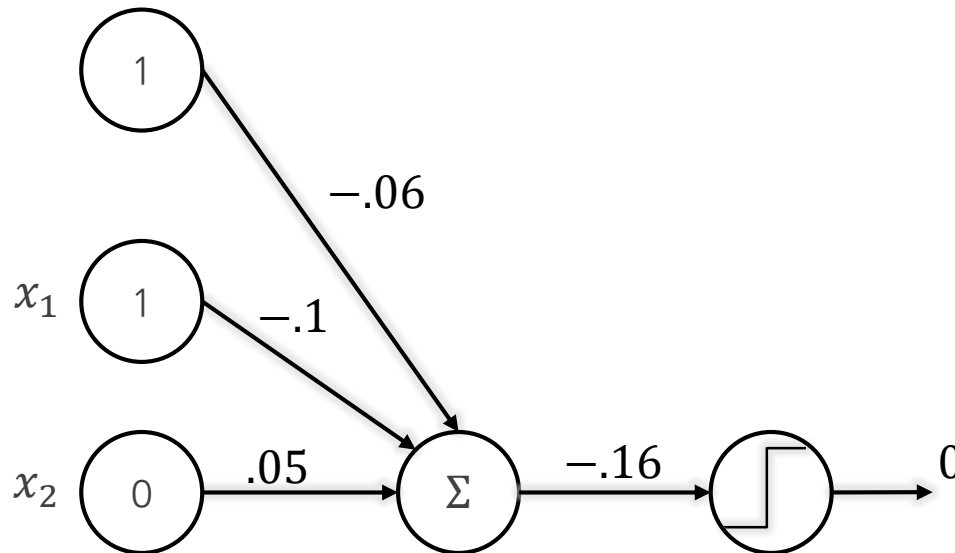


AND 연산

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

퍼셉트론의 학습 방법

- 퍼셉트론의 학습은 “적절한 연결 가중치”를 찾는 것

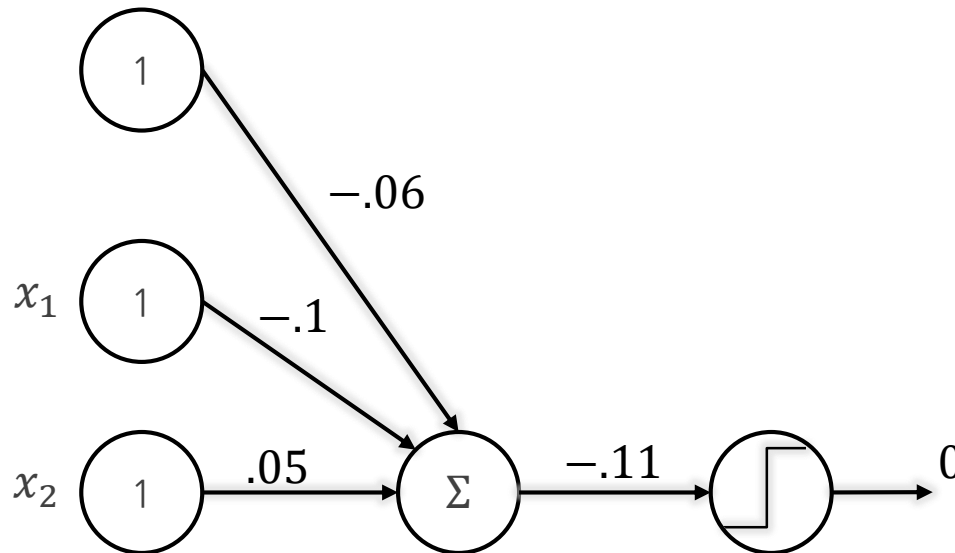


AND 연산

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

퍼셉트론의 학습 방법

- 퍼셉트론의 학습은 “적절한 연결 가중치”를 찾는 것

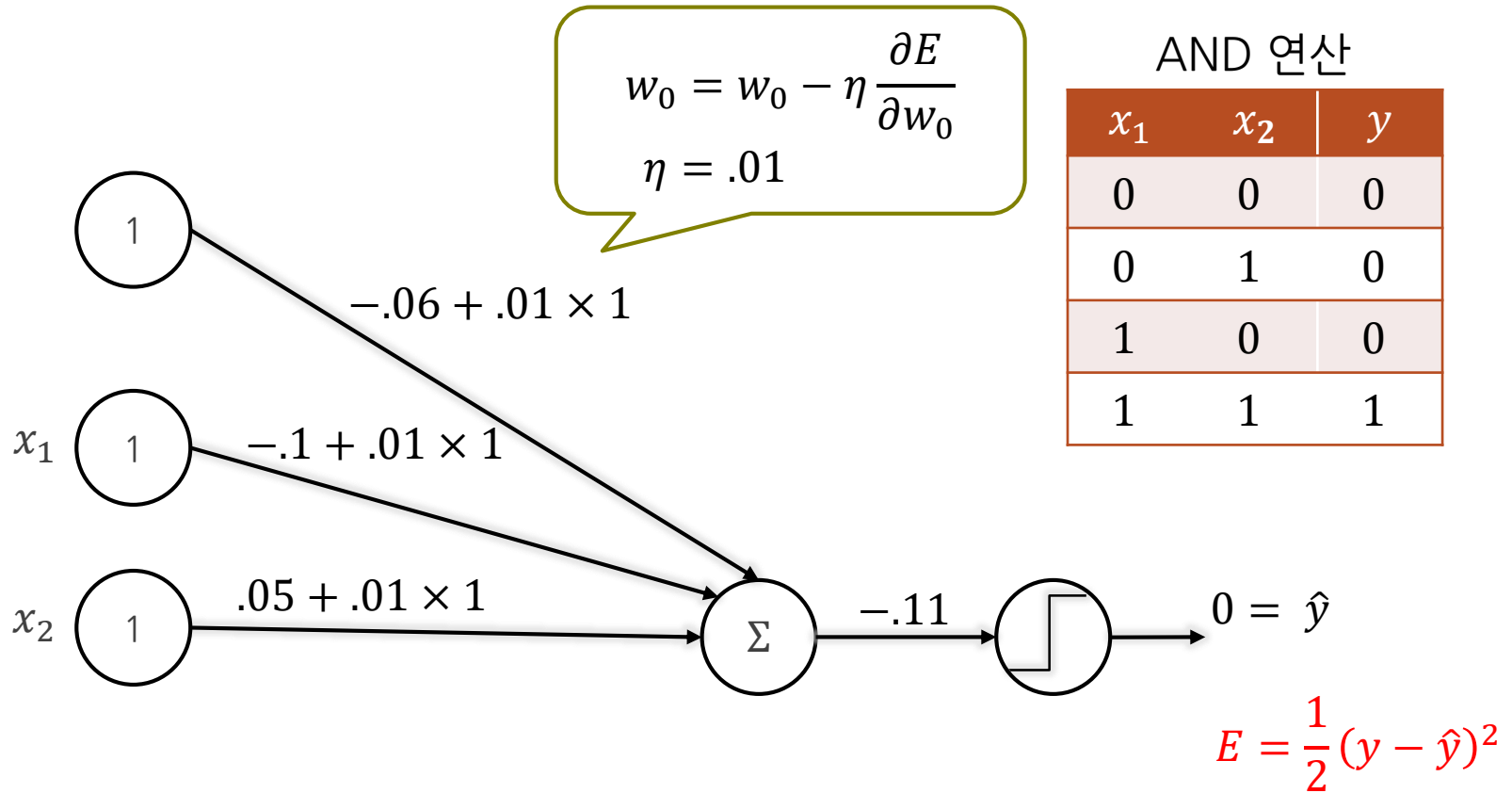


AND 연산

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

퍼셉트론의 학습 방법

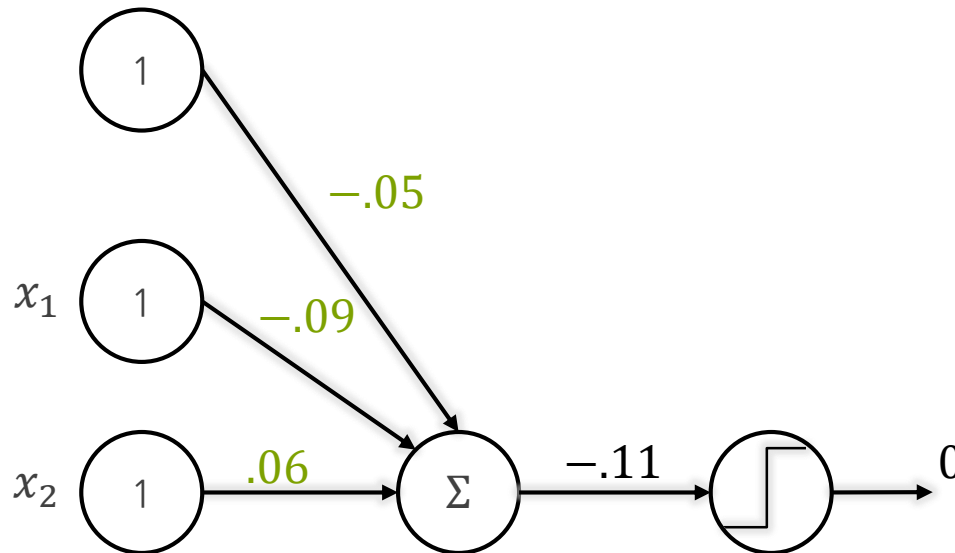
- 퍼셉트론의 학습은 “적절한 연결 가중치”를 찾는 것



$$E = \frac{1}{2} (y - \hat{y})^2$$

퍼셉트론의 학습 방법

- 퍼셉트론의 학습은 “적절한 연결 가중치”를 찾는 것

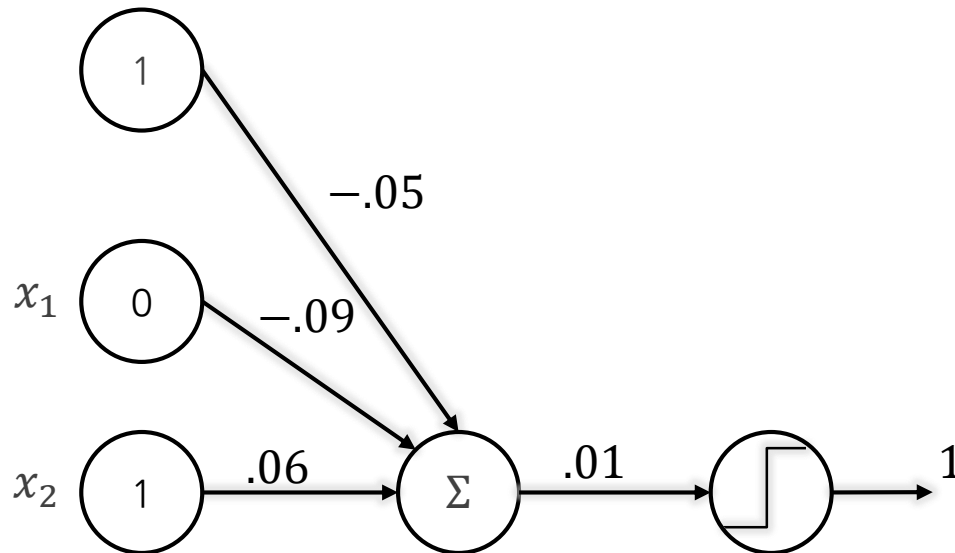


AND 연산

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

퍼셉트론의 학습 방법

- 퍼셉트론의 학습은 “적절한 연결 가중치”를 찾는 것

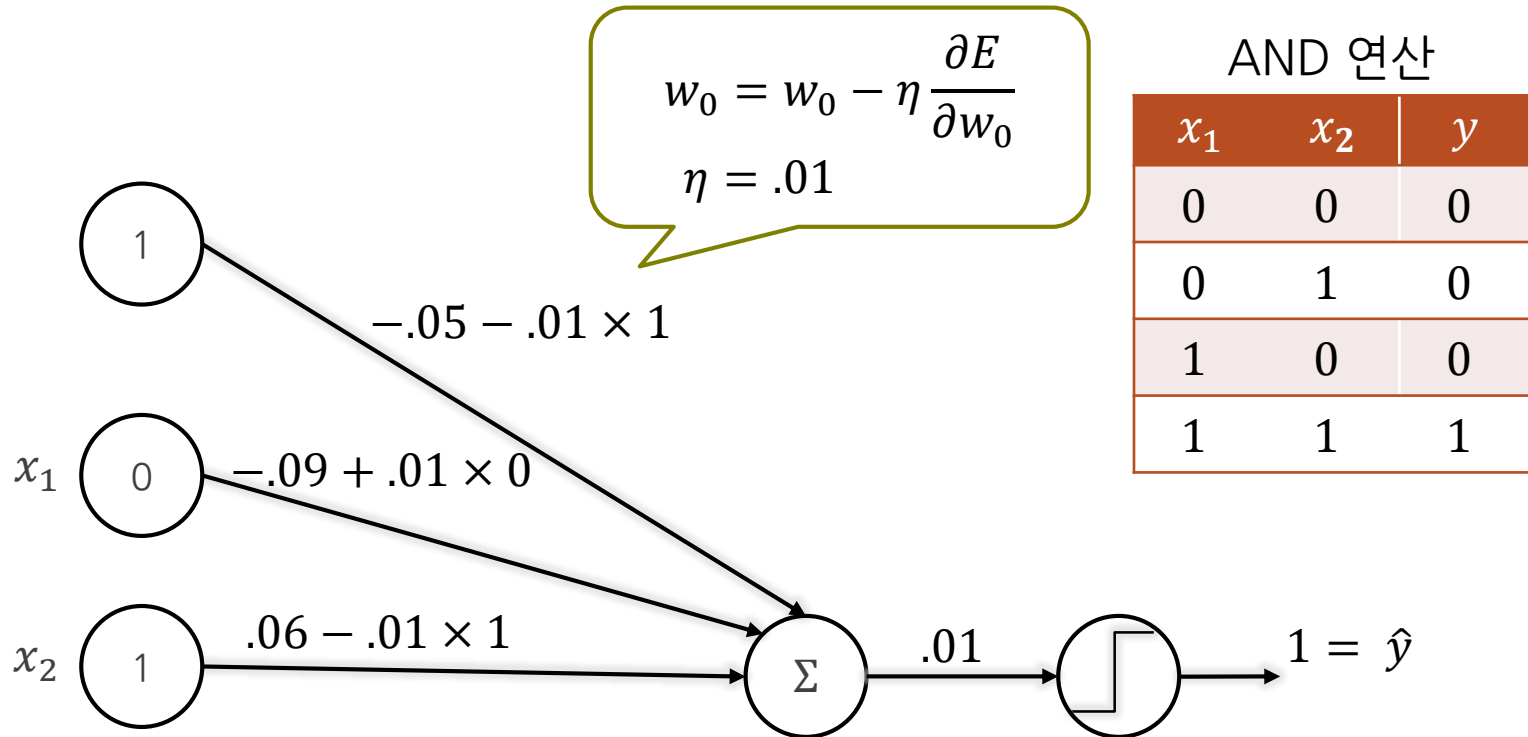


AND 연산

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

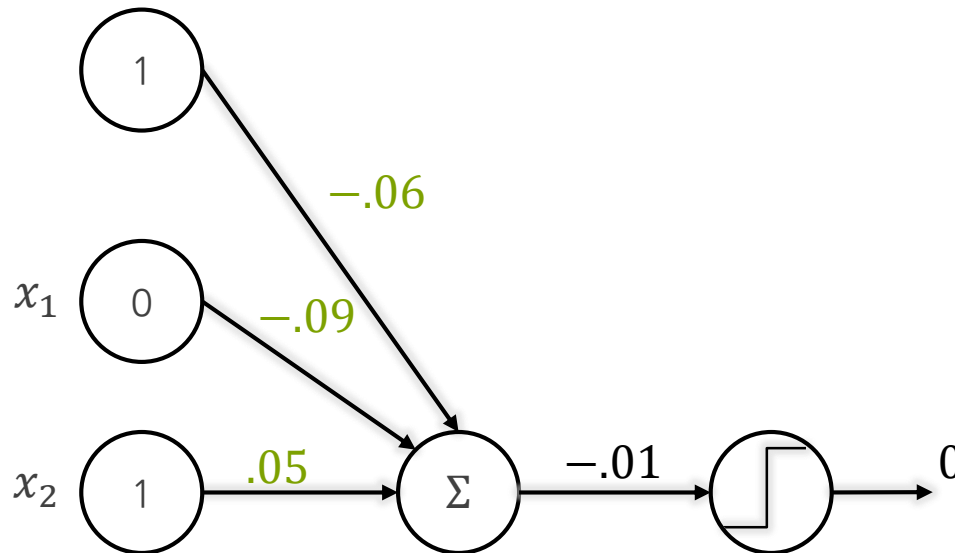
퍼셉트론의 학습 방법

- 퍼셉트론의 학습은 “적절한 연결 가중치”를 찾는 것



퍼셉트론의 학습 방법

- 퍼셉트론의 학습은 “적절한 연결 가중치”를 찾는 것

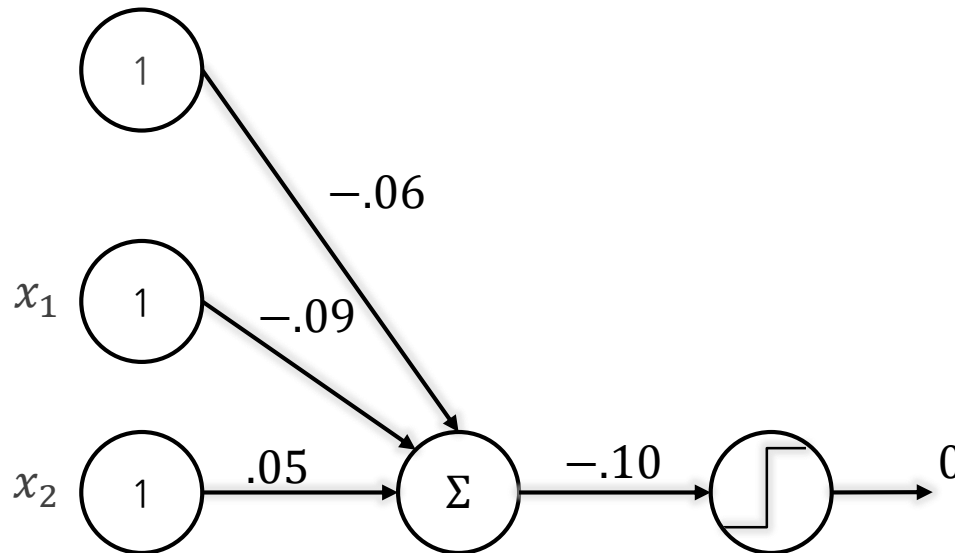


AND 연산

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

퍼셉트론의 학습 방법

- 퍼셉트론의 학습은 “적절한 연결 가중치”를 찾는 것

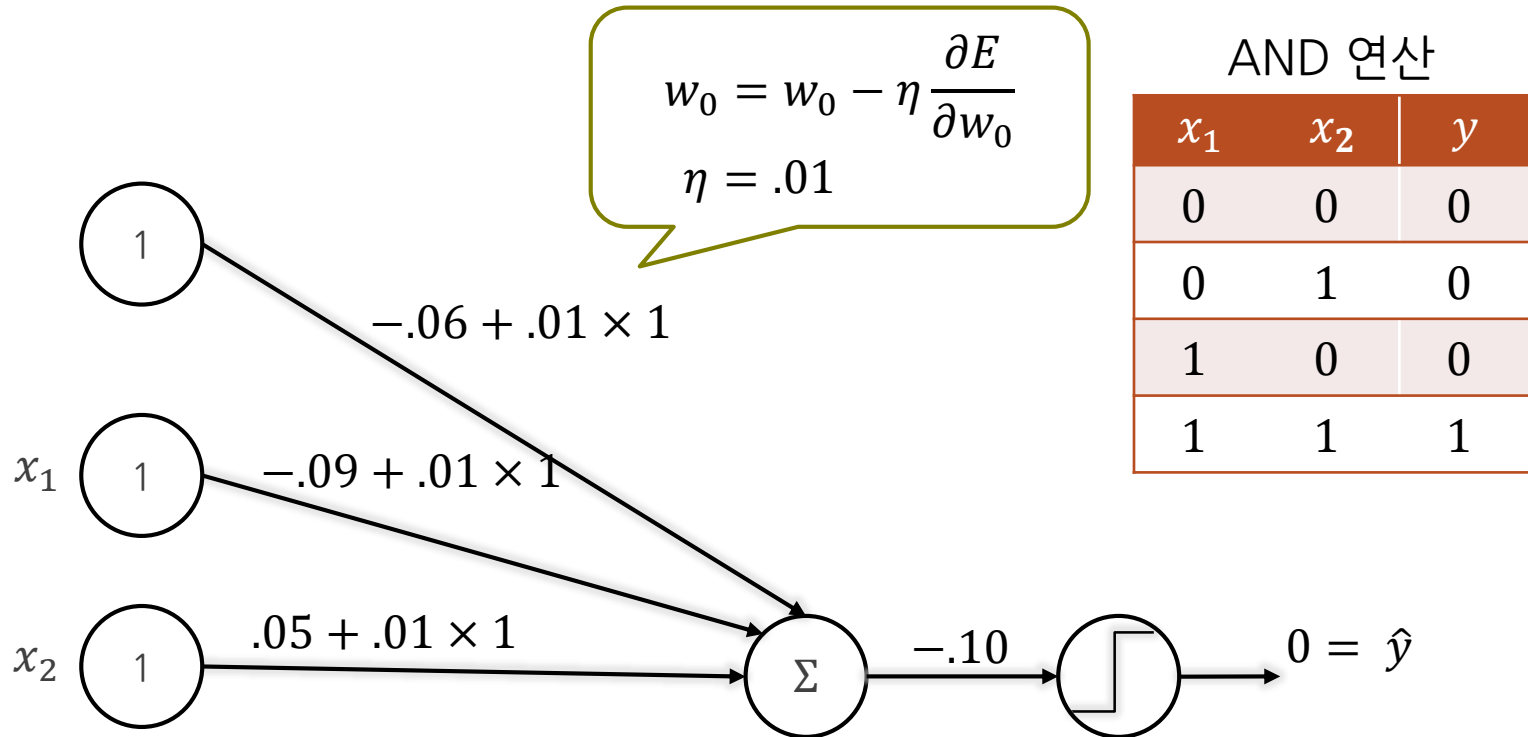


AND 연산

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

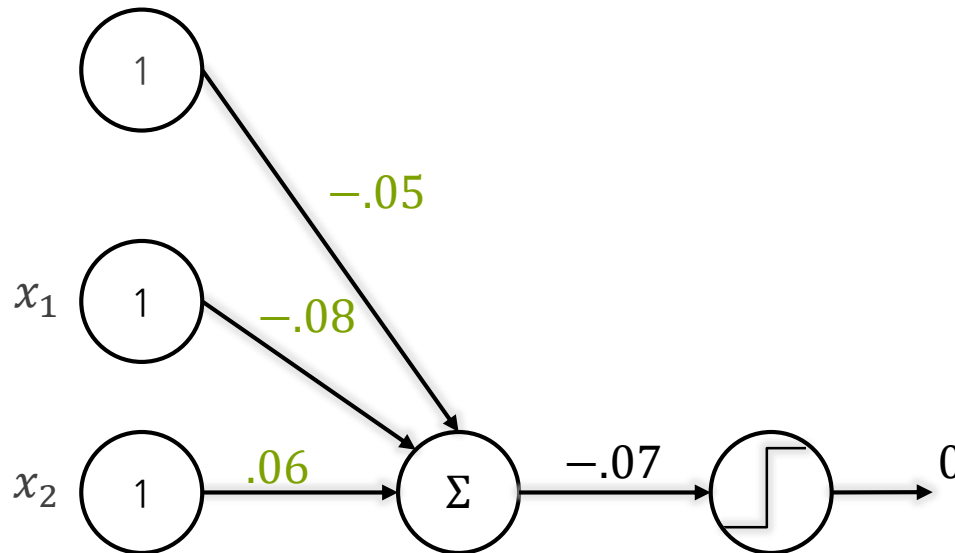
퍼셉트론의 학습 방법

- 퍼셉트론의 학습은 “적절한 연결 가중치”를 찾는 것



퍼셉트론의 학습 방법

- 퍼셉트론의 학습은 “적절한 연결 가중치”를 찾는 것



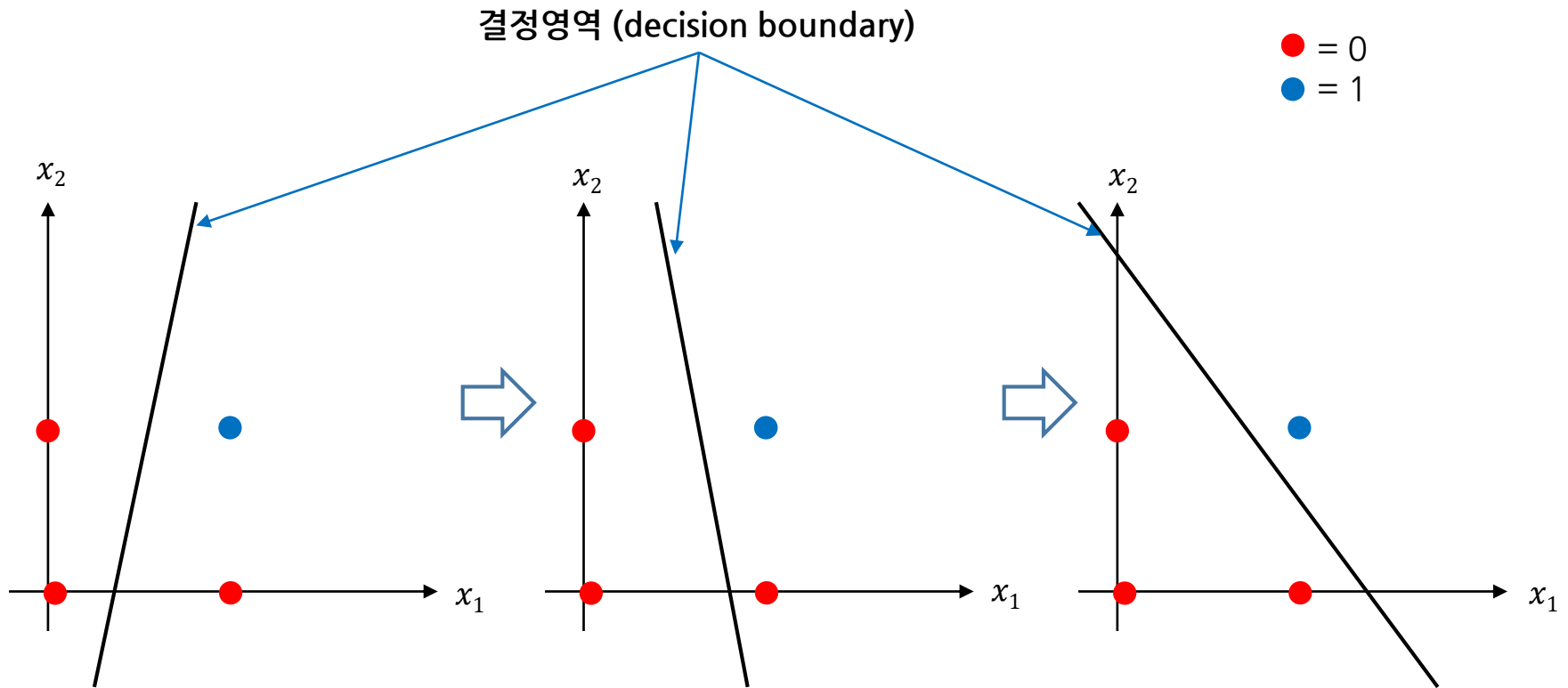
AND 연산

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

수렴할때까지 계속...

퍼셉트론의 학습 방법

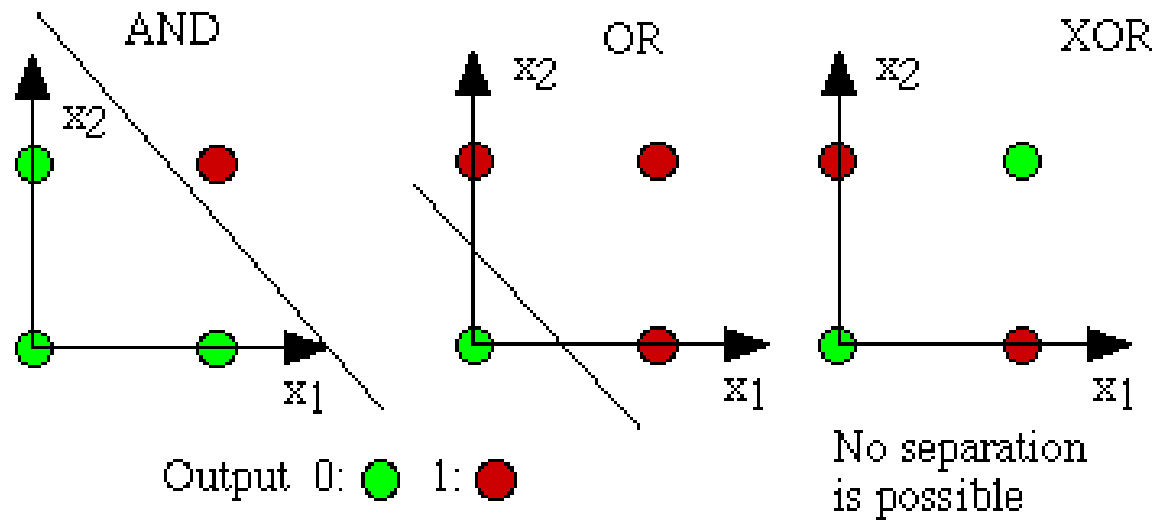
- 퍼셉트론의 학습은 “적절한 연결 가중치”를 찾는 것
 - 적절한 연결 가중치를 가지게 되면 결정영역을 바꿀 수 있음



퍼셉트론의 한계점

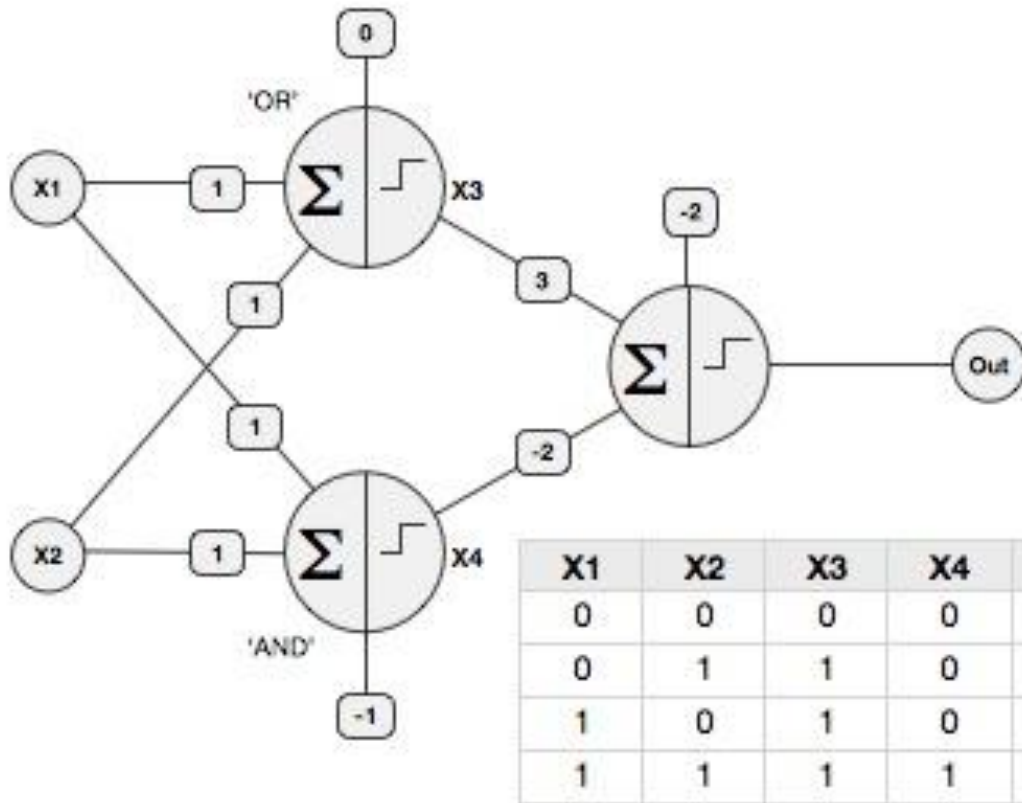
■ 퍼셉트론의 한계점은

- 비선형 분류 문제(XOR 분류와 같은 문제)를 풀 수 없다는 것.

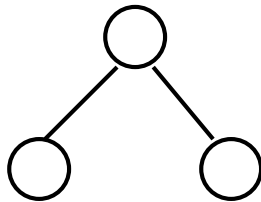
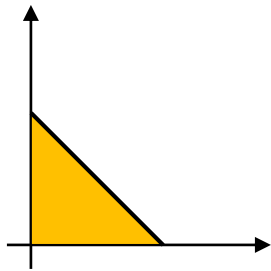


퍼셉트론의 한계 극복방법

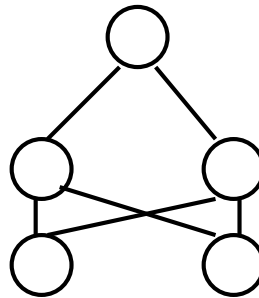
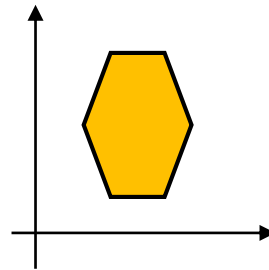
- 만약 Hidden layer가 있다면
 - 비선형 문제(XOR 분리)를 해결할 수 있게 된다.



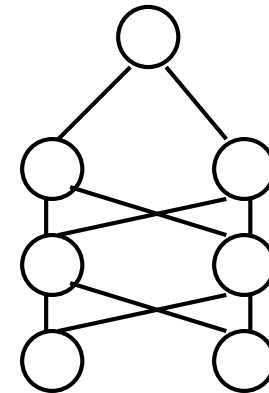
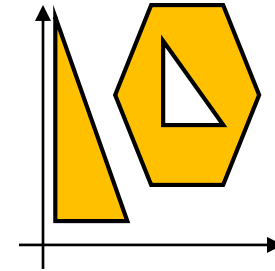
Hidden Layer의 역할



첫번째 레이어는
선형 결정 영역을 만듦

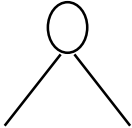
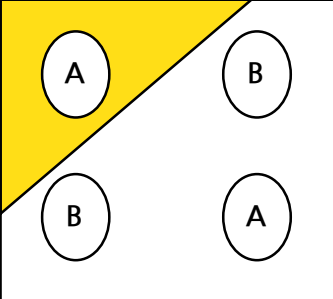
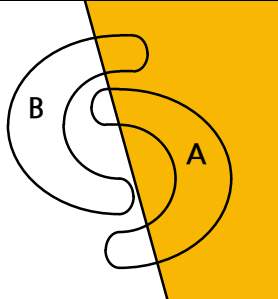
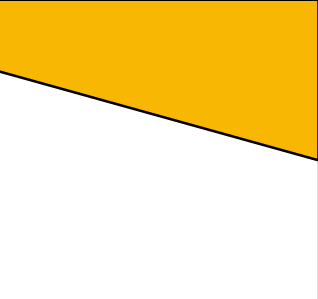
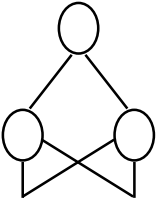
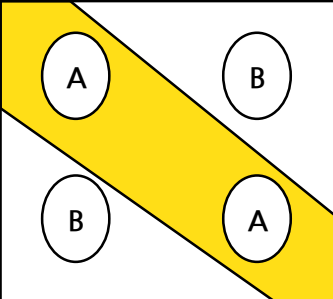
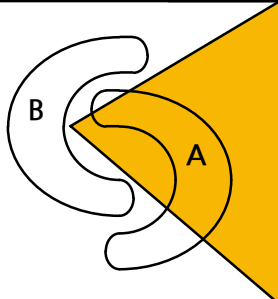
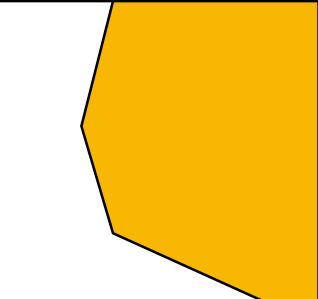
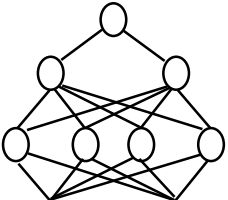
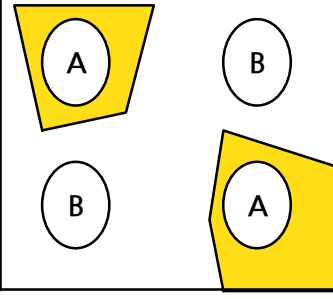
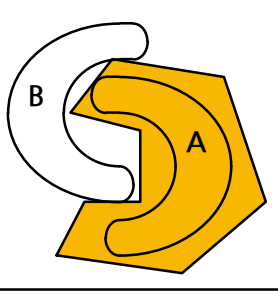
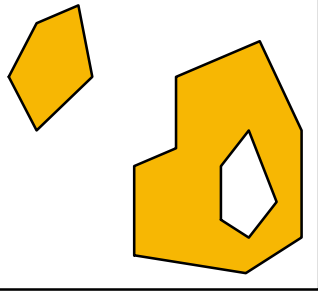


두번째 레이어는
결정 영역들을 합침

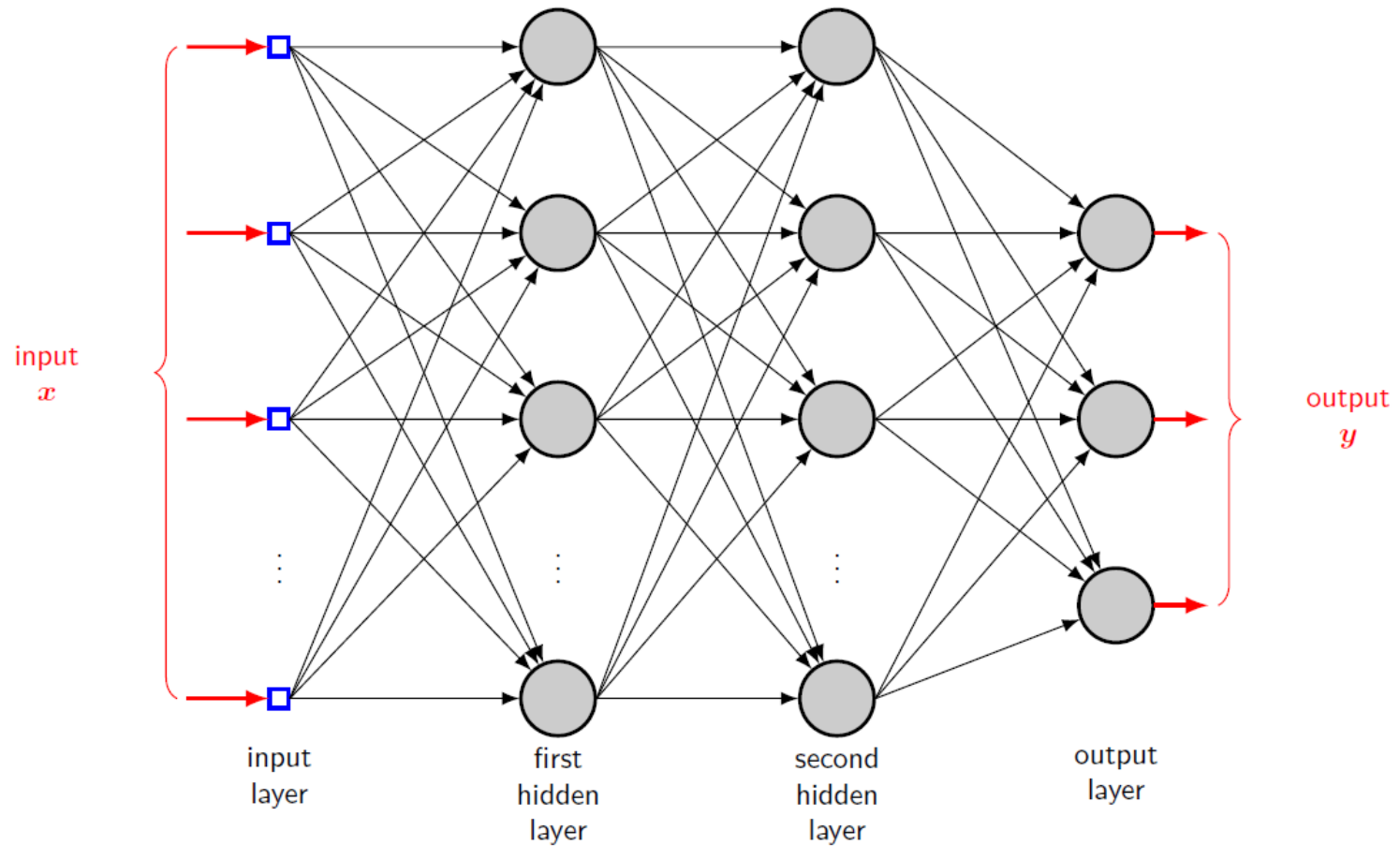


세번째 레이어는
임의의 결정 영역들을 생성

Hidden Layer의 역할

구조	결정 영역 형태	Exclusive-OR 문제	얽힌 결정 영역을 갖는 클래스들	대부분의 결정 영역들
<p>단층</p> 	<p>초평면에 의해 나뉘어지는 반평면 (Hyper plane)</p>			
<p>두 개층</p> 	<p>볼록한 모양 또는 닫힌 영역</p>			
<p>세 개층</p> 	<p>임의의 형태 (노드들의 개수에 따라 복잡도가 결정됨)</p>			

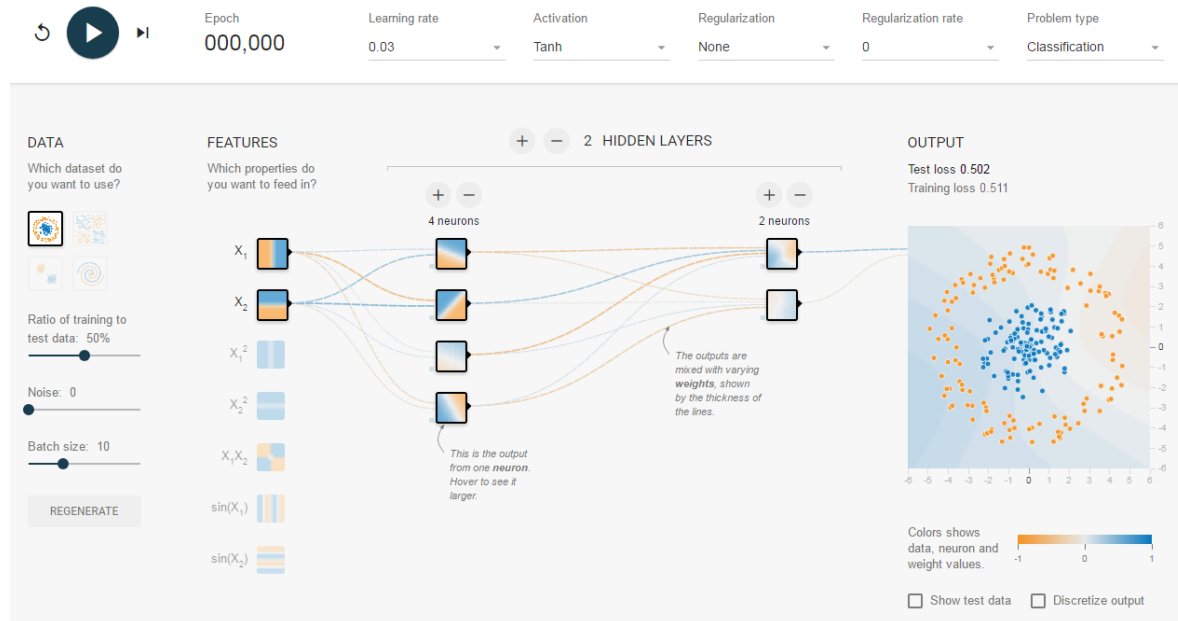
Multi-Layer Perceptron



Neural network playground

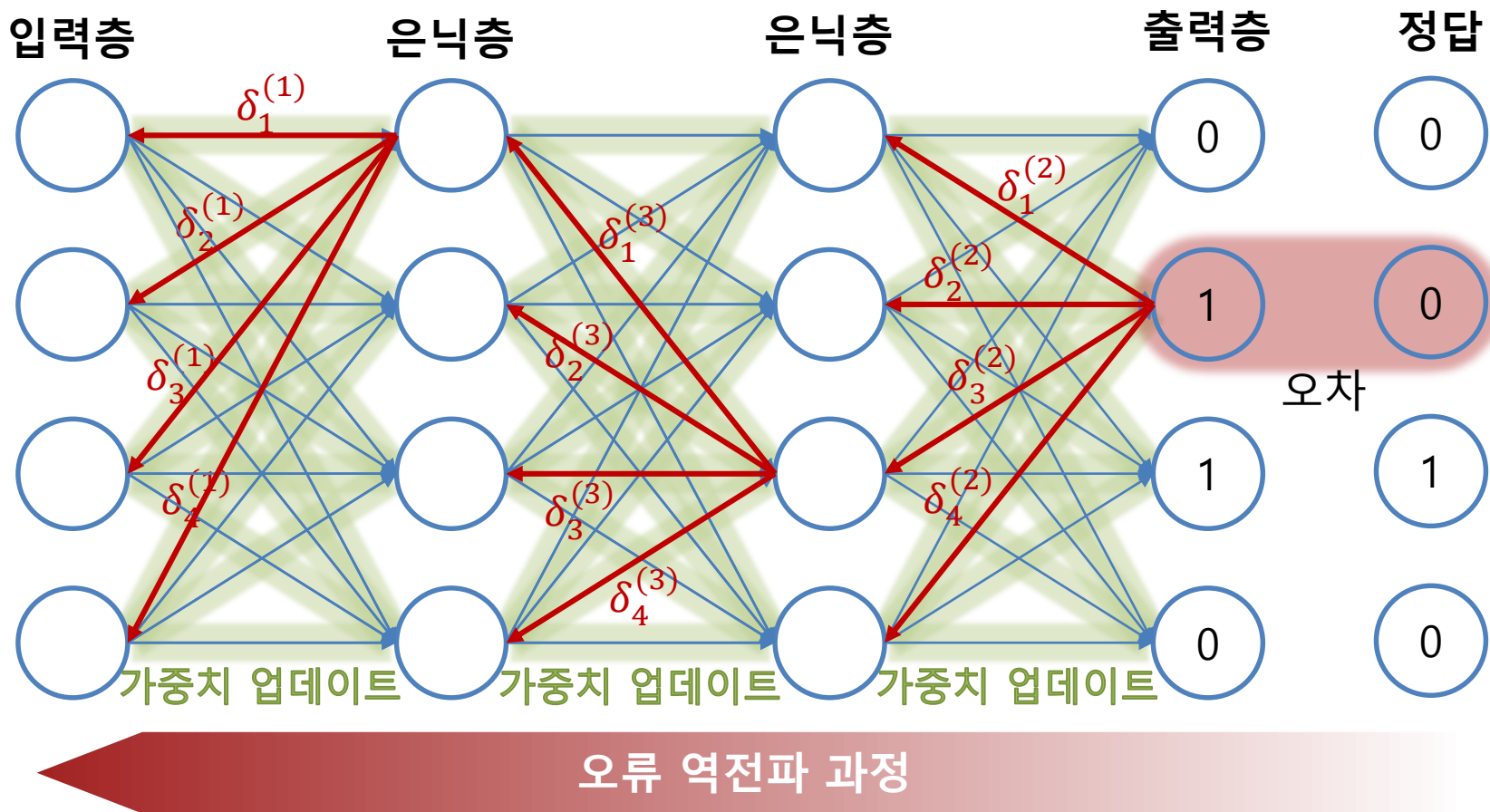
■ <http://playground.tensorflow.org>

- 신경망의 노드와 레이어를 추가한 뒤 어떻게 결정 영역이 학습되는지
- 네 가지의 서로 다른 Dataset으로 확인



MLP 학습 방법

Backpropagation Algorithm



Gradient Descent 알고리즘

■ Gradient Descent

- Neural network의 weight을 조절하는 과정.

$$\theta = \theta - \eta \nabla_{\theta} J(\theta)$$

- 네트워크의 parameter(weight) : θ
- Loss function(네트워크의 결과값과 실제값 사이의 차이) : $J(\theta)$
- step size 또는 learning rate : η

- 그러나 Loss function계산 시 전체 train data를 모두 사용한다. 하지만, 이를 계산하기 위해선 매우 많은 계산량이 필요하다.

Stochastic Gradient Descent 알고리즘

■ Stochastic Gradient Descent

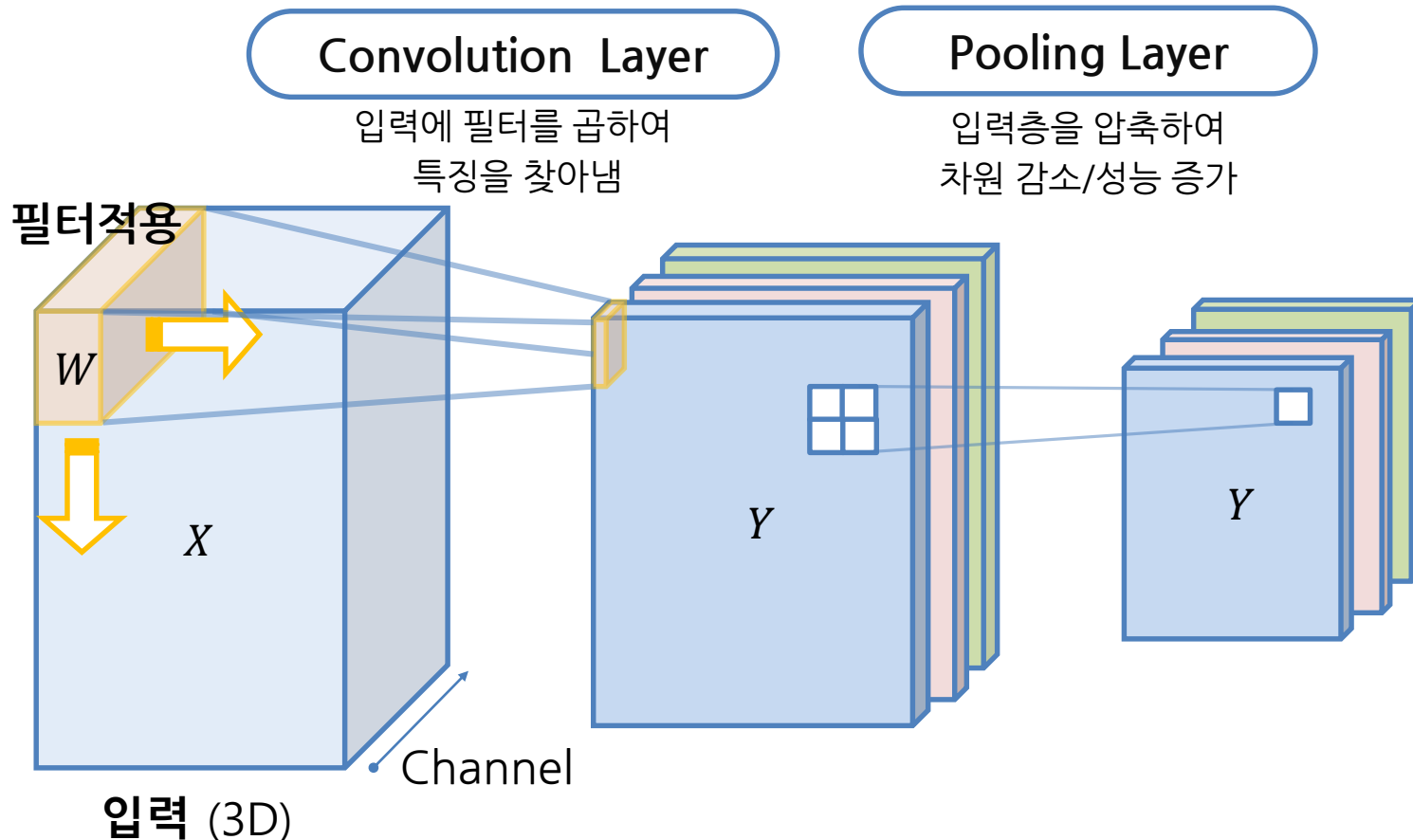
- Loss function을 계산할 때 전체 데이터(batch) 대신 mini-batch(전체 중 일부 조그마한 데이터의 모음)에 대해서만 Loss function을 계산
- Gradient Descent보다 계산 속도가 훨씬 빠르며
- SGD와 GD는 보통 같은 결과로 수렴한다.

$$J_{train}(\theta) \approx J_{mini-batch}(\theta)$$

CNN 이란?

- Convolutional Neural Network

- 평면 형태의 입력(Image)을 처리할 때 주로 사용되는 알고리즘.
- 얼굴인식, 이미지 분류에서 탁월한 성능을 보이고 있다.



Supplementary Material

- <http://cs231n.stanford.edu> (Stanford 강의노트)
- <https://wikidocs.net/4213> (Andrew Ng 머신러닝 설명)
- <http://www0.cs.ucl.ac.uk/staff/d.silver/web/Teaching.html> (강화학습)
- http://www.modulabs.co.kr/RL_library/2136 (강화학습 설명)
- **Google** (dqn, alphago model, ...)

Thank you!

made by Seongho choi