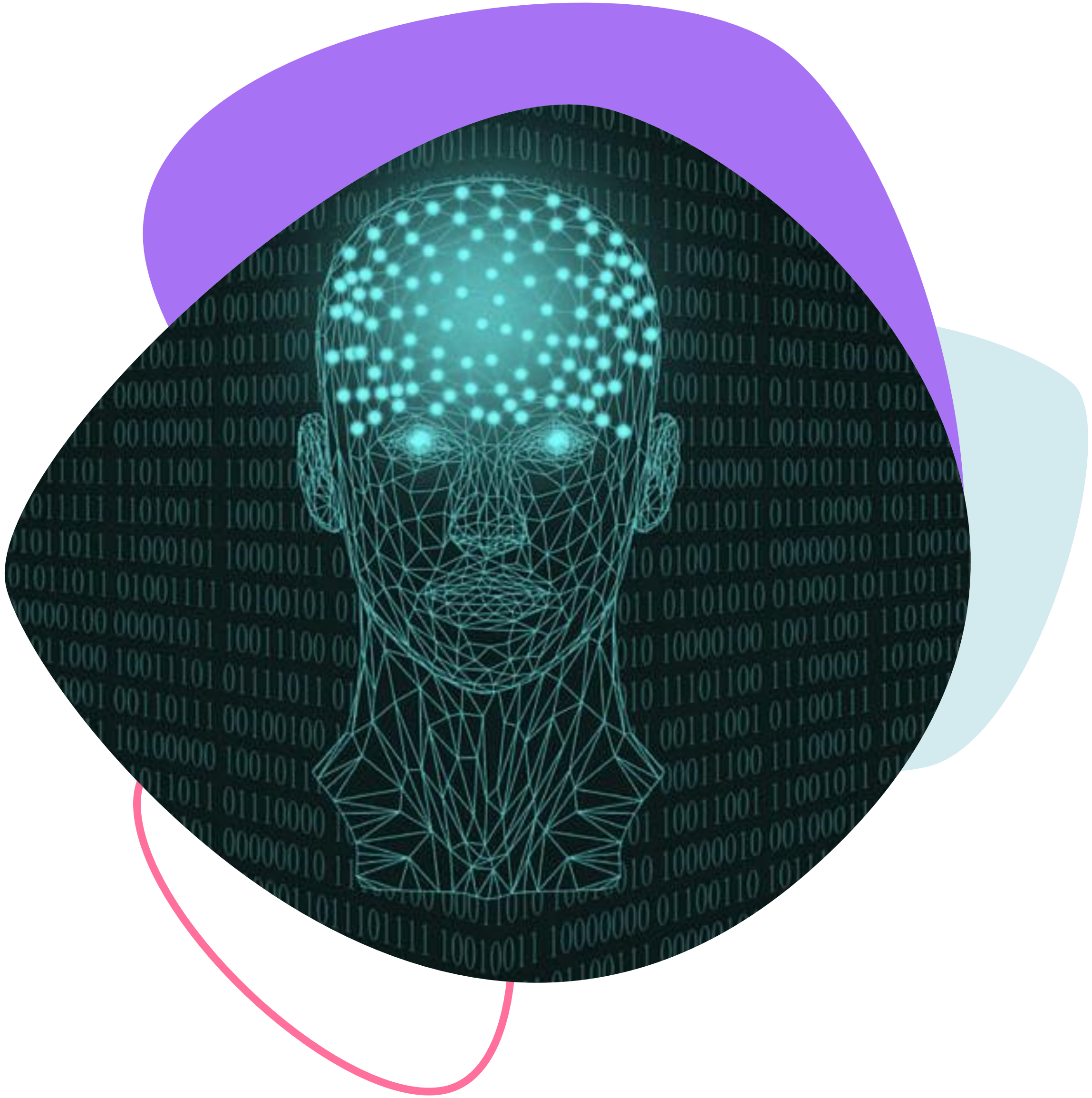
AI 신경망의 기초: MLP 완전정복

Session 4. Multilayer Perceptron



Presented by ***Yunseon Byun*** (yun-seon@kimm.re.kr)

Curriculum

Start

Day 1.
Python 설치 및 가상환경 설정

Seed

Day 1.
Python 문법 기초 숙지

Seedling

Day 2.
인공지능 개요 &
MLP 원리 이해

Sapling

Day 2.
MLP 성능 향상 & AI 모델링 결과 해석

Tree

Day 2.
MLP 회귀/분류 모델링

Apple

After today,
AI Modeling on yourself

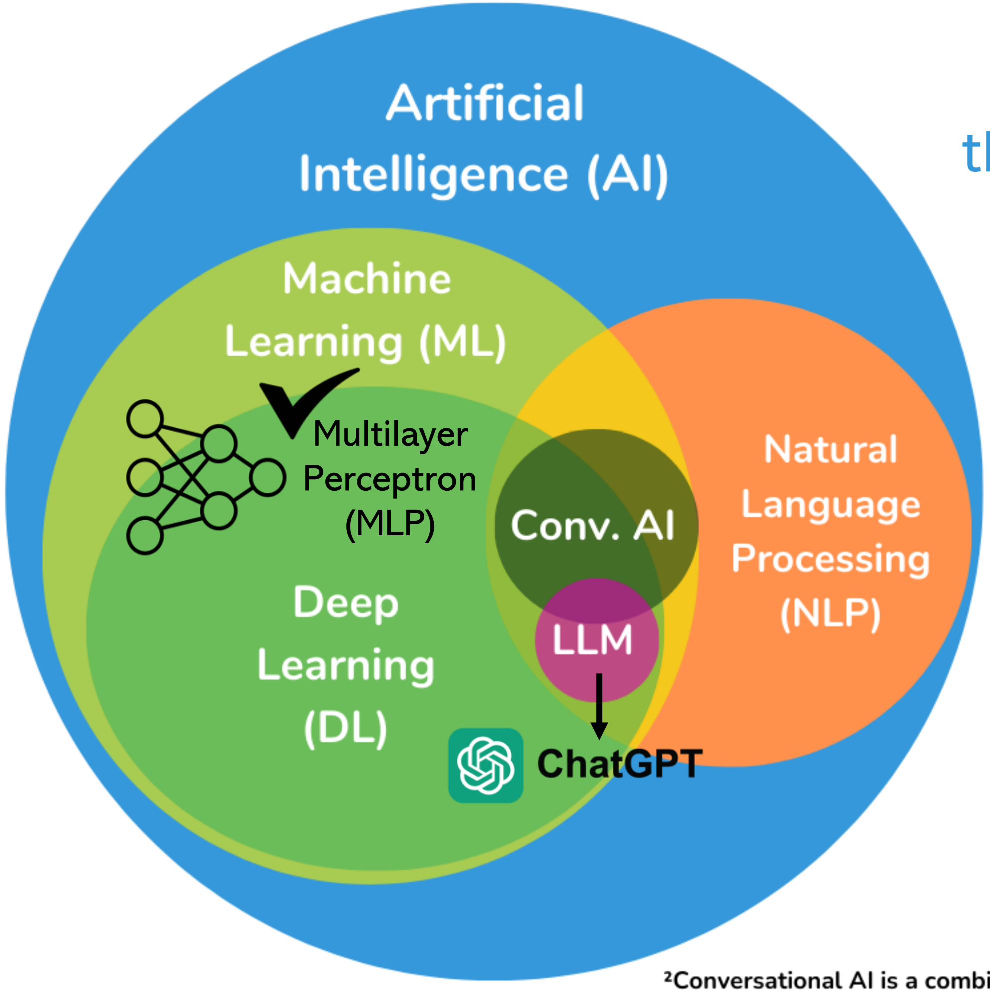
Contents

- MLP in the AI scope
- Fundamental principles of MLP
- Model training of MLP

Contents

- MLP in the AI scope
- Fundamental principles of MLP
- Model training of MLP

인공지능(Artificial Intelligence, AI)



"Artificial Intelligence (AI) is the ability for a computer to think and learn."

- Artificial Intelligence (AI)
- Machine Learning (ML)
- Deep Learning (DL)
- Natual Language Processing (NLP)
- Large Language Model (LLM)[1]
- Conversational AI (Conv. AI)[2]

[1]LLM is an intersection of DL and NLP

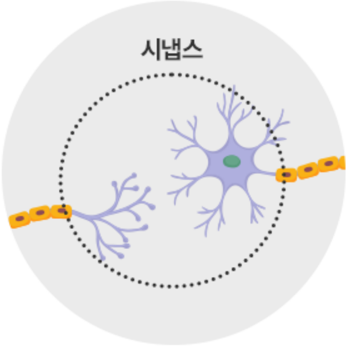
[2]Conversational AI is a combination of ML and NLP. It may include DL and LLM, but that isn't always the case.

출처: Gennadiy Bezko, Understand AI, ML & Co in Contact Centers: Definitions & Explanations, miarec, 2023.

인공지능(Artificial Intelligence, AI)

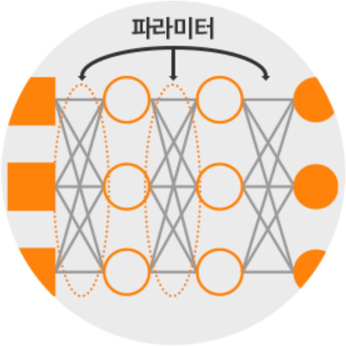
인간의 시냅스 VS AI의 파라미터

인간의 시냅스



- 뇌에는 시냅스가 있는데 정보를 학습하고 기억함
- 시냅스가 많으면 많을수록 처리할 수 있는 정보량이 많아짐

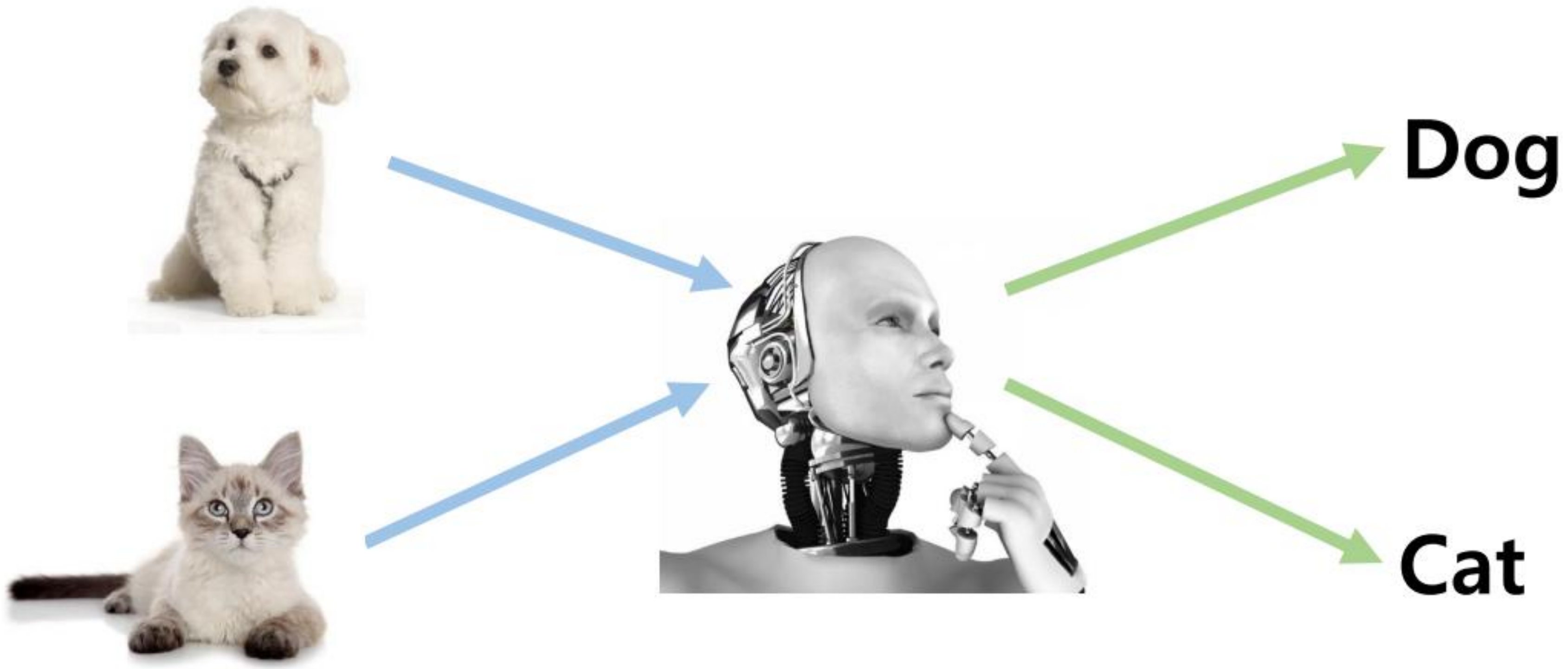
AI의 파라미터



- 인공신경망과 파라미터는 인간의 뇌와 시냅스를 본뜬 것
- 파라미터가 많을수록 정교한 학습이 가능함

출처: AI를 넘어 초거대 AI의 시대가 왔다! ChatGPT의 등장으로 주목받고 있는 초거대 AI란? | 같이 읽는 IT 트렌드, SK C&C report, 2023.

AI 기본 원리



입력이 주어지면 출력을 내보낸다.

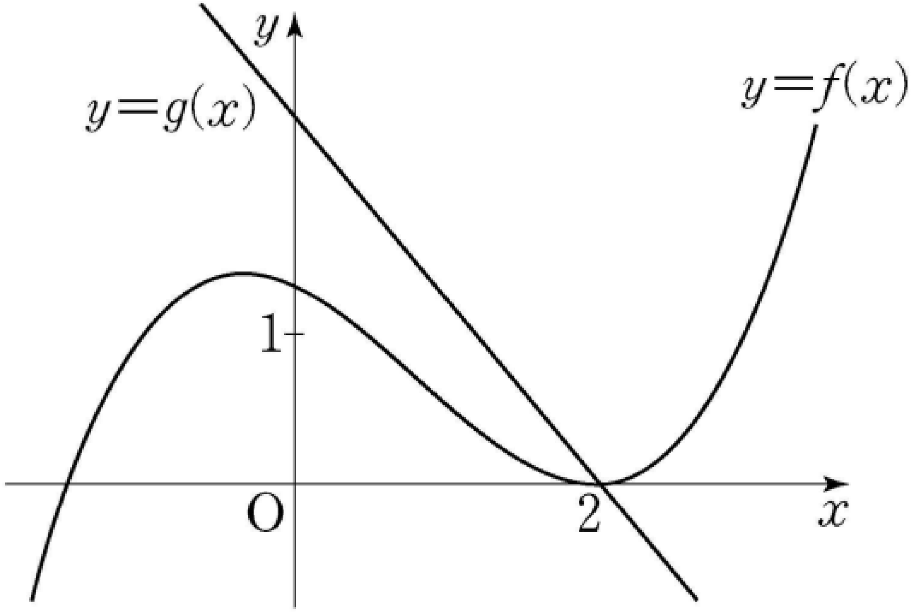
출처: 인공지능이란(Artificial Intelligence), 이수안 컴퓨터 연구소(suanlab), 2021.

AI 기본 원리

출력 입력

$$y = f(x)$$

함수(모델)



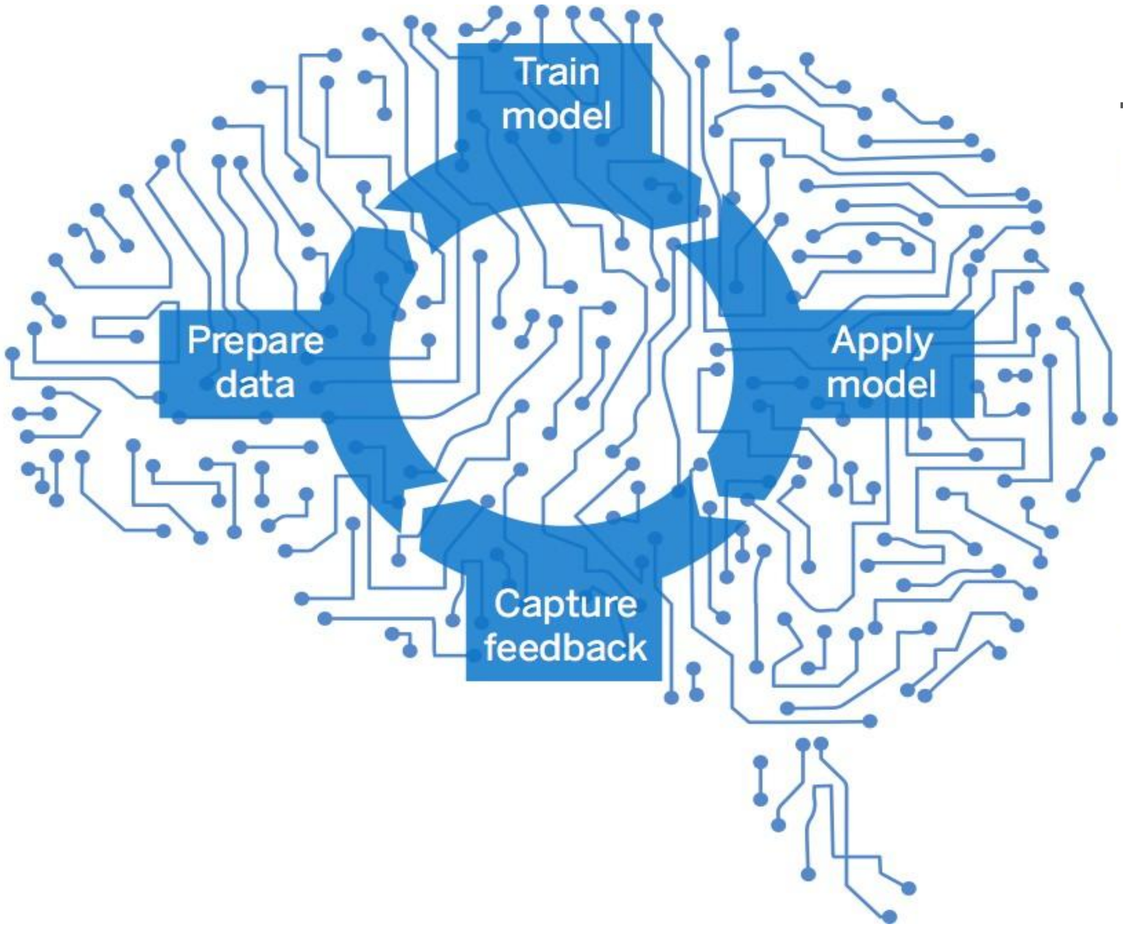
출처: 인공지능이란(Artificial Intelligence), 이수안 컴퓨터 연구소(suanlab), 2021.

AI 기본 원리

입력

Structured and
unstructured data

- ✓ Tables (excel, csv)
- ✓ Text
- ✓ Images, audio and video
- ✓ Time series and geospatial data



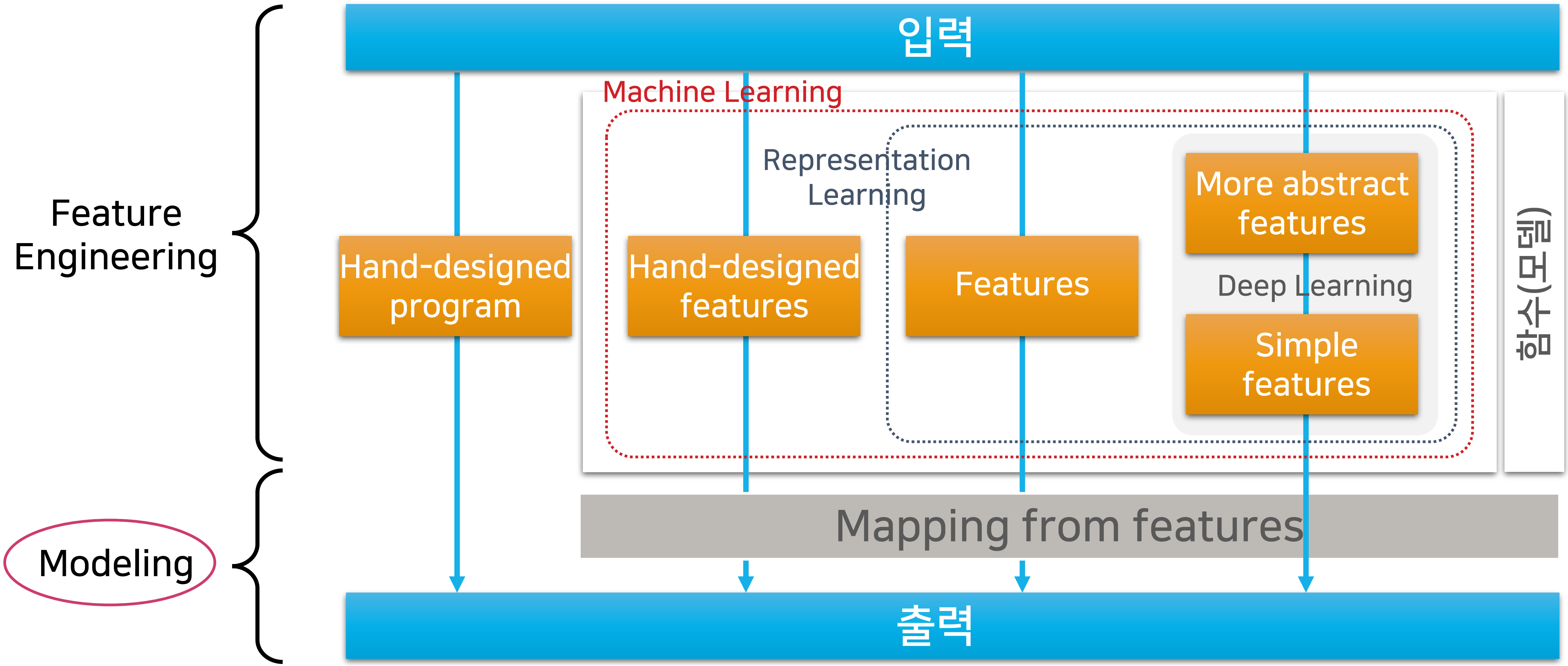
출력

from big-data
to smart data

- ✓ Regression/Predictions
- ✓ Classifications and clustering
- ✓ Recommendations
- ✓ Automation

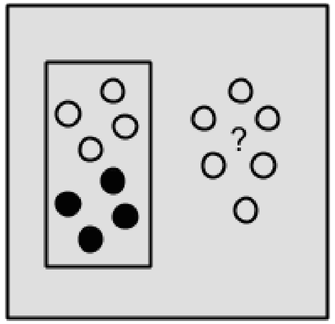
출처: 인공지능이란(Artificial Intelligence), 이수안 컴퓨터 연구소(suanlab), 2021.

AI 기본 원리



출처: 인공지능이란(Artificial Intelligence), 이수안 컴퓨터 연구소(suanlab), 2021.

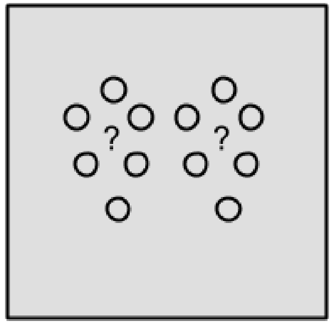
AI 모델 구분



MLP

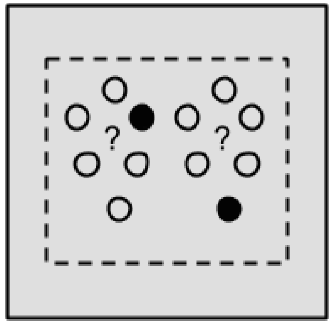
지도학습
Supervised Learning

- 입력과 출력에 매핑(mapping) 되는 일반적인 규칙을 학습
- 입력과 출력 레이블을 모델 학습에 직접적으로 사용하는 방식



비지도학습
Supervised Learning

- 사전 정의된 출력 없이 입력 데이터 사용하는 방식
- 입력 데이터의 구조나 패턴을 찾는 것을 목적으로 함

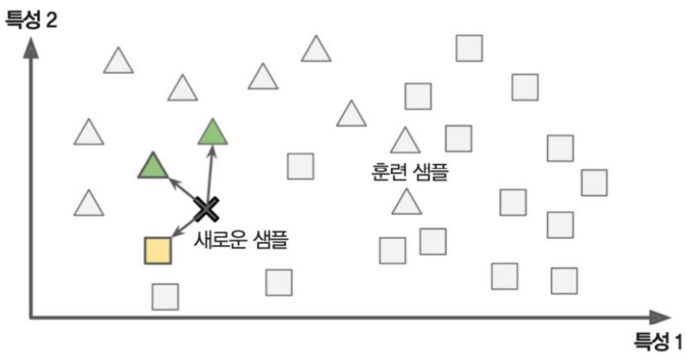


준(반)지도학습
Semi-supervised Learning

- 출력 레이블이 있는 데이터와 없는 데이터가 혼합된 경우에 사용
- 일부 데이터의 출력 레이블을 모델 학습에 직접적으로 사용하는 방식

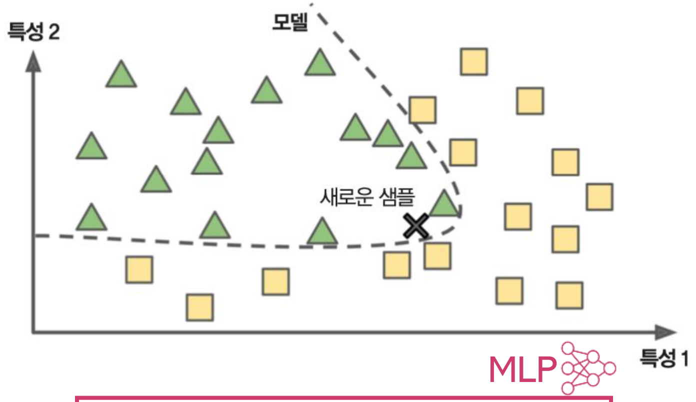
출처: 머신러닝(Artificial Intelligence), 이수안 컴퓨터 연구소(suanlab), 2021.
출처: Jason Brownlee, A Tour of Machine Learning Algorithms, 2023.

AI 모델 구분



사례 기반 학습
Instance-based Learning

- 샘플을 기억하는 방식으로 학습
- 예측을 위해 샘플 사이의 유사도를 측정한 후, 유사한 샘플과 동일하게 출력하는 방식



모델 기반 학습
Model-based Learning

- 샘플을 사용해 설계된 모델을 학습
- Train data로 학습한 모델을 사용해 Test data에 대한 출력을 예측(Regression) 및 분류(Classification)하는 방식

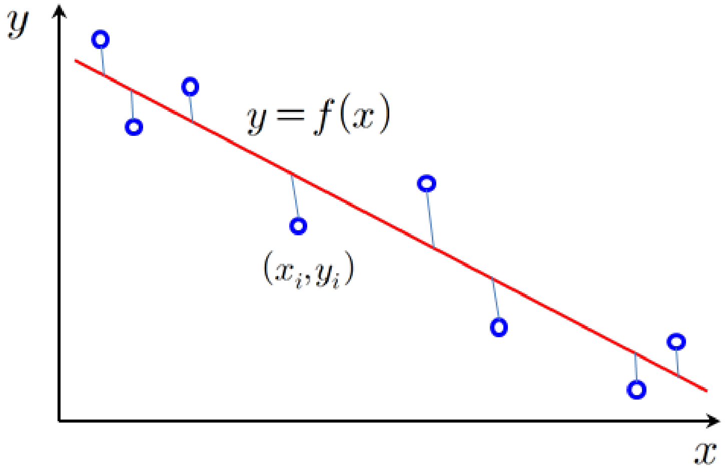
출처: 머신러닝(Artificial Intelligence), 이수안 컴퓨터 연구소(suanlab), 2021.
출처: Jason Brownlee, A Tour of Machine Learning Algorithms, 2023.

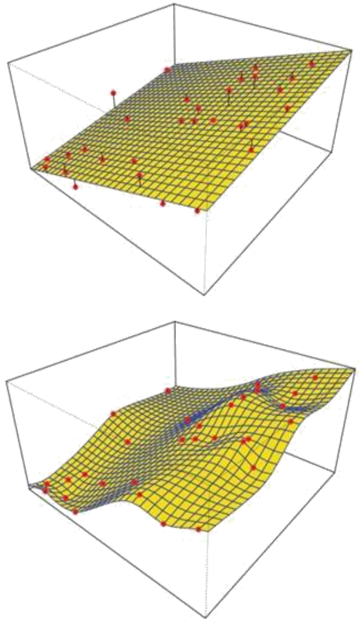
AI 모델링 목적

예측/회귀(Regression)

- 학습 데이터에 부합되는 출력 값이 실수인 함수를 찾는 문제
- 오차(예측 값과 실제 값 간의 차이)를 줄일 수 있는 함수를 찾는 것
- 모델의 종류(함수의 종류)에 영향을 받음

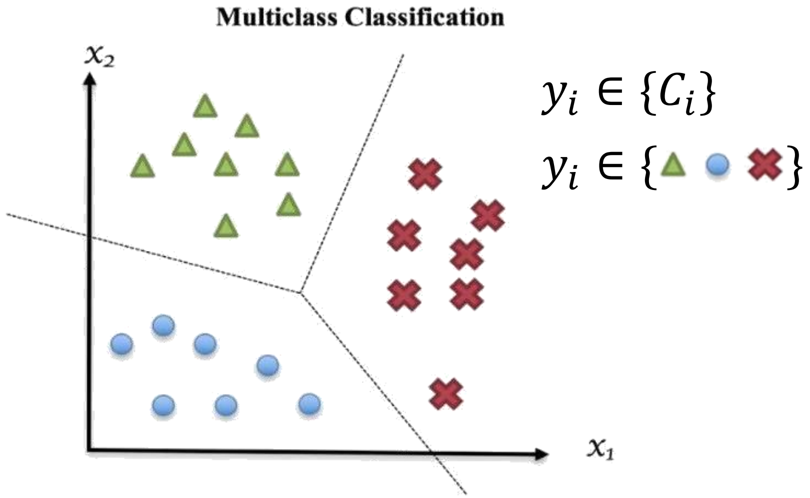
$$f^*(x) = \arg\min_f \sum_{i=1}^{n}(y_i - f(x_i))^2$$





분류(Classification)

- 데이터들을 사전에 정해진 몇 개의 부류(class, category)로 대응시키는 문제
- 학습 데이터를 잘 분류할 수 있는 함수를 찾는 것
- 함수의 형태는 수학적 함수일 수도 있고, 규칙일 수도 있음
- 이상적인 분류 모델: 학습에 사용되지 않은 데이터에 대해서 분류 성능이 높으며, 일반화(generalization) 능력이 좋은 것



출처: 머신러닝(Artificial Intelligence), 이수안 컴퓨터 연구소(suanlab), 2021.

AI 모델링 목적 및 평가

"좋은" 모델 ⇒ 일반화된 모델

① 학습(Train)/검증(Test) 데이터 나누기

독립변수 (features, x)	종속변수 (label, y)	학습 데이터
		검증 데이터

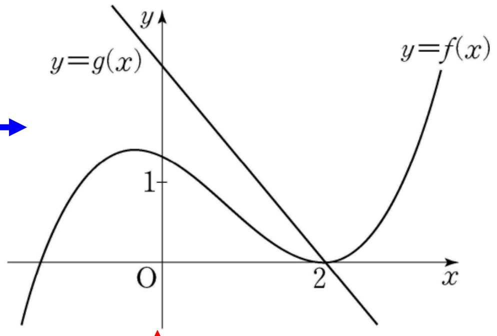
② 학습 데이터를 사용해 모델 학습 ($f(x)$ 함수 찾기)

독립변수 (features, x) | 종속변수 (label, y) — 학습 데이터 → f 함수 찾기

③ 테스트 데이터를 사용해 모델 일반화 성능 확인

검증 데이터 →

Regression model



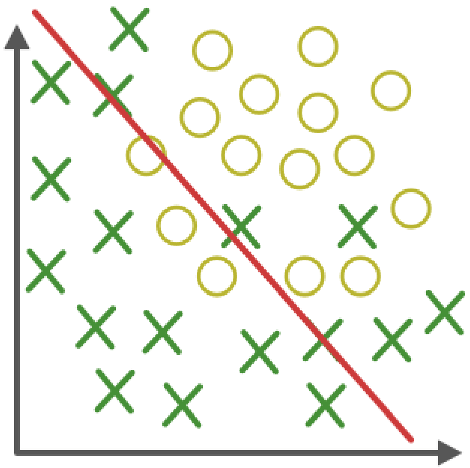
모델의 일반화(generalization) 능력을 높이기 위해,

예측 값과 실제 값 간의 오차 줄이기

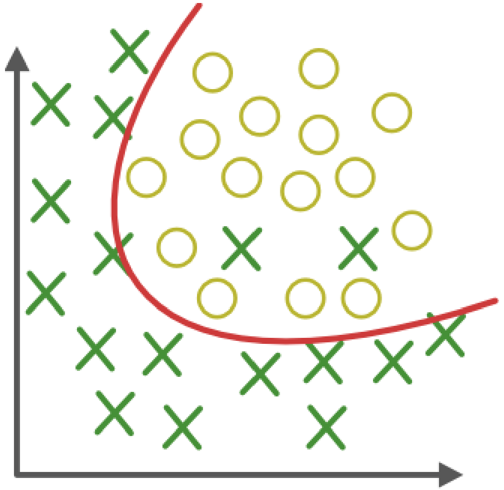
AI 모델링 목적: 분류(Classification)

지나치게 단순한 모델(함수) 사용

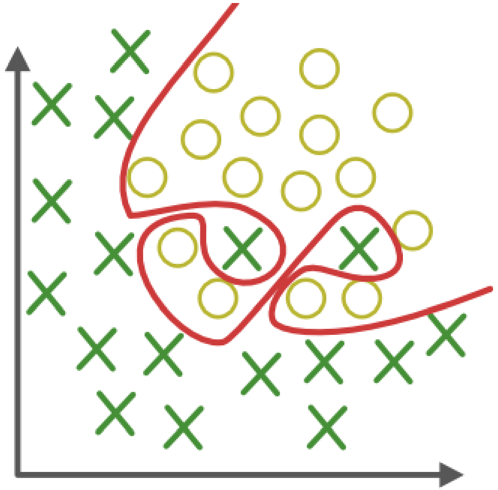
지나치게 복잡한 모델(함수) 사용



과소적합(underfitting)

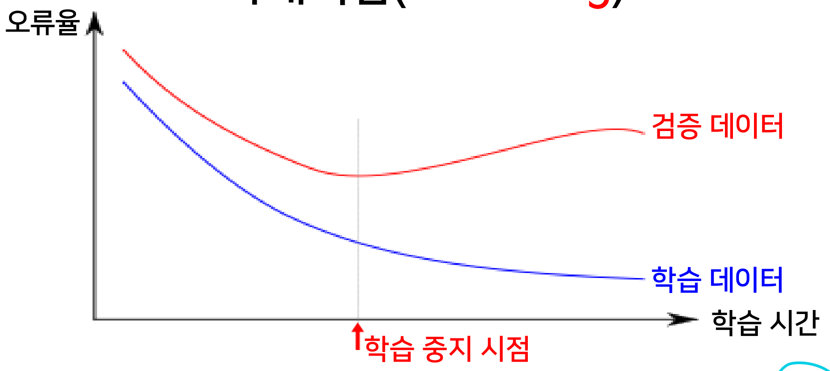


정적합(good fitting)



과대적합(overfitting)

- 분류의 과적합(overfitting) 대응 방법
  - 학습과정에서 별도의 검증 데이터(validation data)에 대한 성능 평가
  - 검증 데이터에 대한 오류가 감소하다가 증가하는 시점에 학습 중단



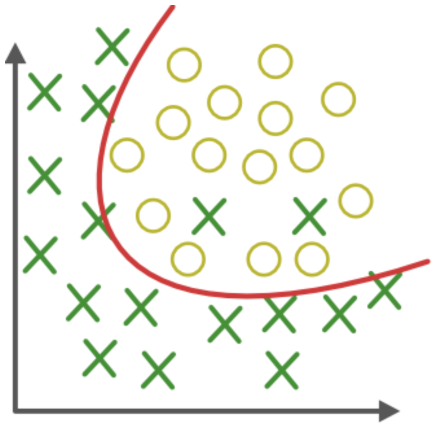
출처: PIAI, [머신러닝] 지도학습(학습/테스트/검증 데이터 분리), LCY tistory, 2022.

AI 모델링 목적: 분류(Classification)

x

입력 (독립변수)

$$y = f(x)$$



정적합(good fitting)

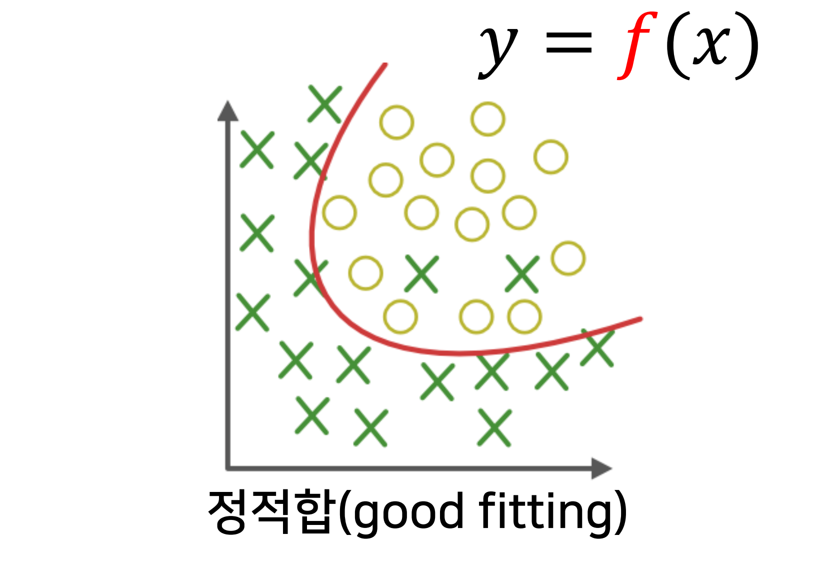
y

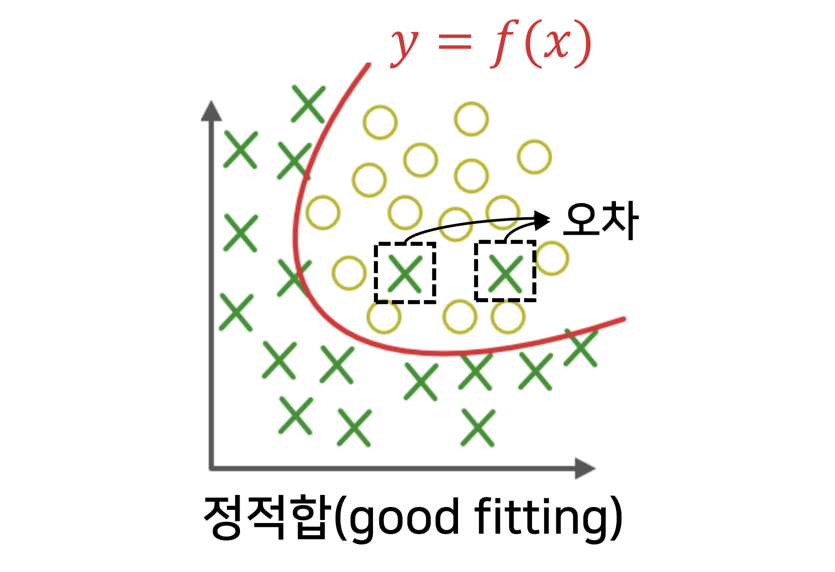
범주형 값

출력 (종속변수)

적절한 함수(모델) 찾기

AI 모델링 목적: 분류(Classification)



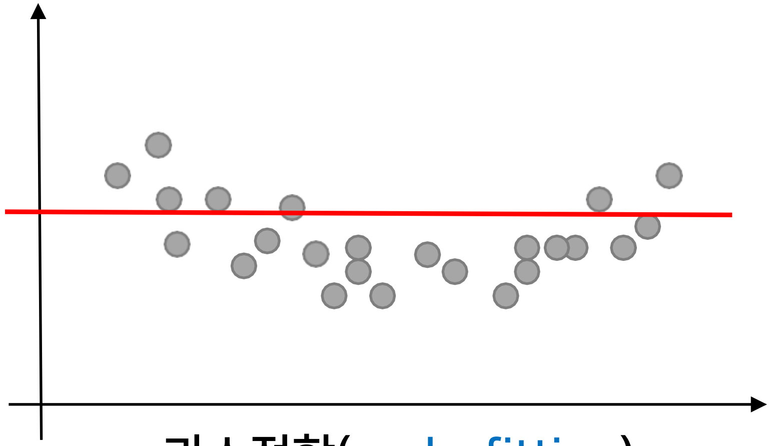


적절한 함수(모델) 찾기 ·······▸ $\hat{y}$**예측 값**과 y**실제 값** 간의 **오차** 줄이기

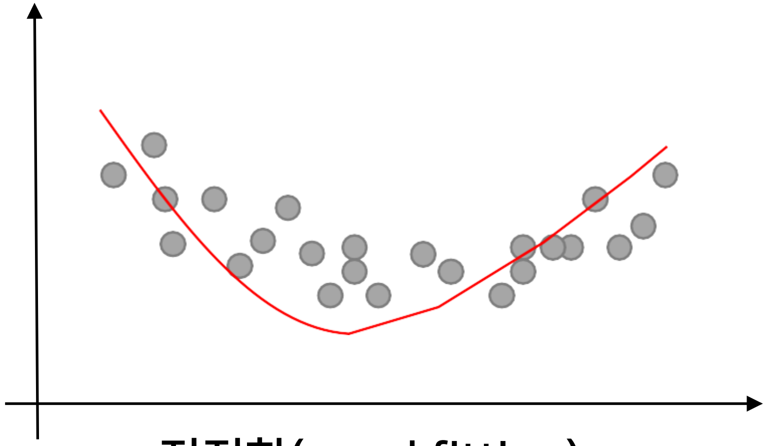
AI 모델링 목적: 회귀(Regression)

지나치게 단순한 모델(함수) 사용

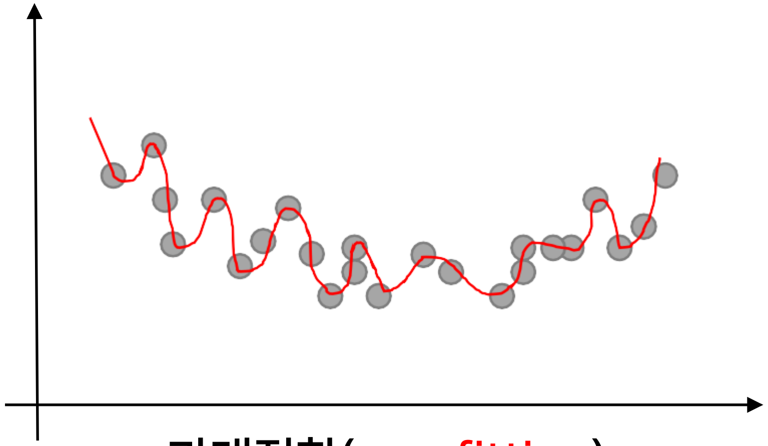
지나치게 복잡한모델(함수) 사용



과소적합(underfitting)



정적합(good fitting)



과대적합(overfitting)

- 회귀의과적합(overfitting) 대응방법
  - 모델의복잡도(model complexity)를성능평가에반영
  - 목적함수 변형(e.g. 오차의합+ (가중치) * (모델복잡도))

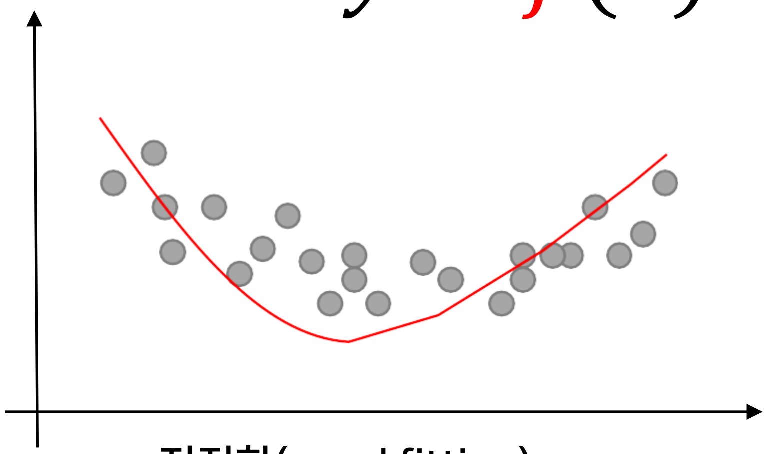
출처: 머신러닝(Artificial Intelligence), 이수안 컴퓨터 연구소(suanlab), 2021.

AI 모델링 목적: 회귀(Regression)

x

입력 (독립변수)

$$y = f(x)$$



정적합(good fitting)

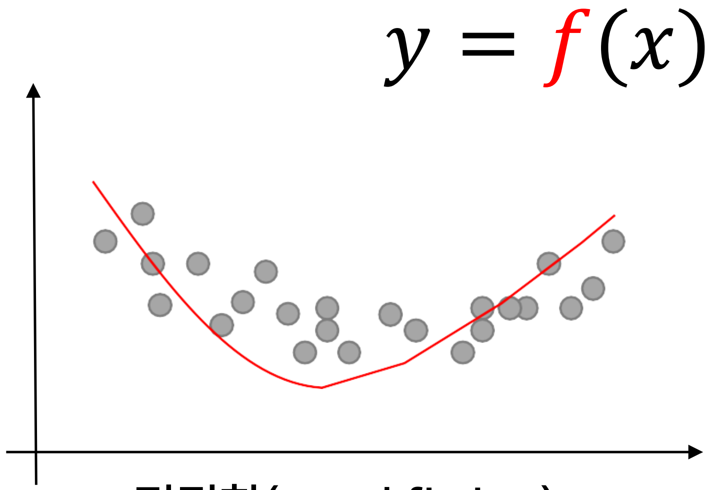
y

실수 값

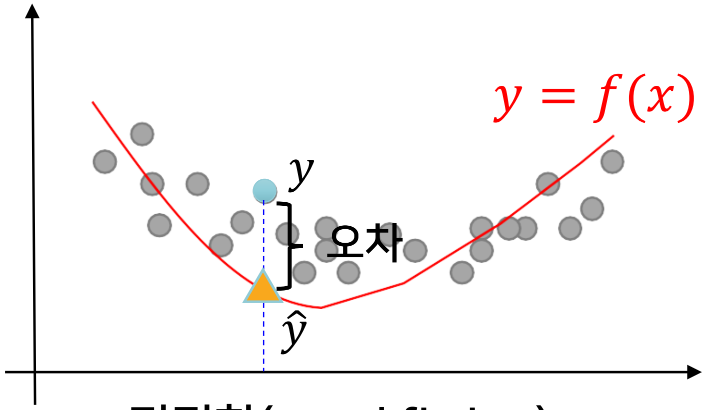
출력 (종속변수)

적절한 함수(모델) 찾기

AI 모델링 목적: 회귀(Regression)



정적합(good fitting)



정적합(good fitting)

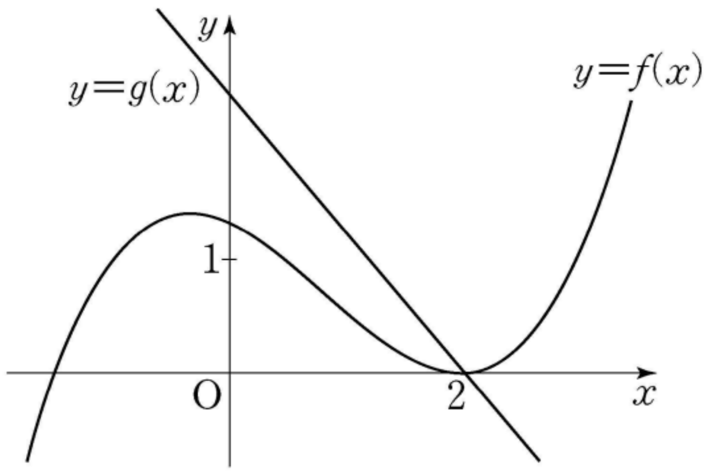
적절한 함수(모델) 찾기 ·······▸ $\hat{y}$**예측 값**과 y**실제 값** 간의 **오차** 줄이기

AI 모델링 목적: 회귀(Regression)

학습 데이터(Train data)

독립변수 (features, x)	종속변수 (label, y)

f 함수 찾기



학습 데이터(Train data)

독립변수 (features, x)		종속변수 (label, y)
x_1	x_2	y
0	2	6
1	3	9.5
2	4	13
3	5	16.5

Regression model
(estimated)

$$y = 0.5x_1 + 3x_2$$

출처: 김성범, [핵심 머신러닝] 머신러닝 모델 학습 프로세스, 고려대학교, 2019.

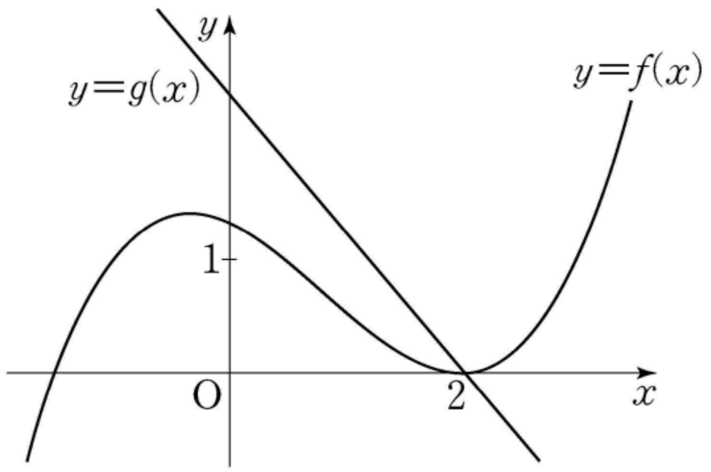
AI 모델링 목적: 회귀(Regression)

검증 데이터(Test data)

독립변수 (features, x)	종속변수 (label, y)

성능 확인

Regression model



검증 데이터 (Test data)

검증 데이터 (Test data)

독립변수 (features, x) x_1	x_2	예측 값 (predicted, $\hat{y}$) $\hat{y}$	종속변수 (label, y) y
1	4	12.5	12.5
3	3	10.5	11
5	1	5.5	6
6	3	12	12

$$y = 0.5x_1 + 3x_2$$

오차: 0.5

오차: 0.5

출처: 김성범, [핵심 머신러닝] 머신러닝 모델 학습 프로세스, 고려대학교, 2019.

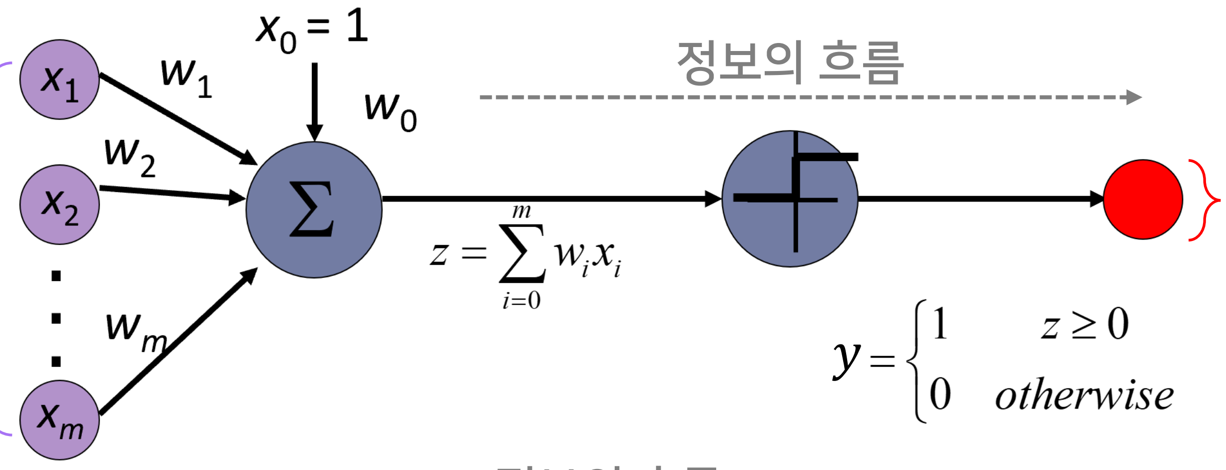
Contents

Perceptron

Perceptron

Introduced by Frank Rosenblatt (1958)

입력



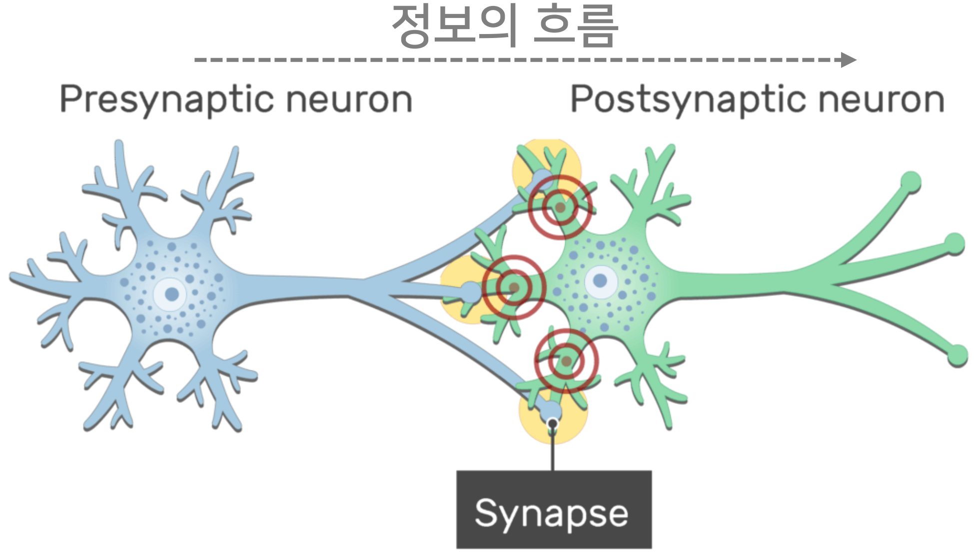
$$z = \sum_{i=0}^{m} w_i x_i$$

$$y = \begin{cases} 1 & z \geq 0 \\ 0 & otherwise \end{cases}$$

출력

인간의 시냅스/뉴런

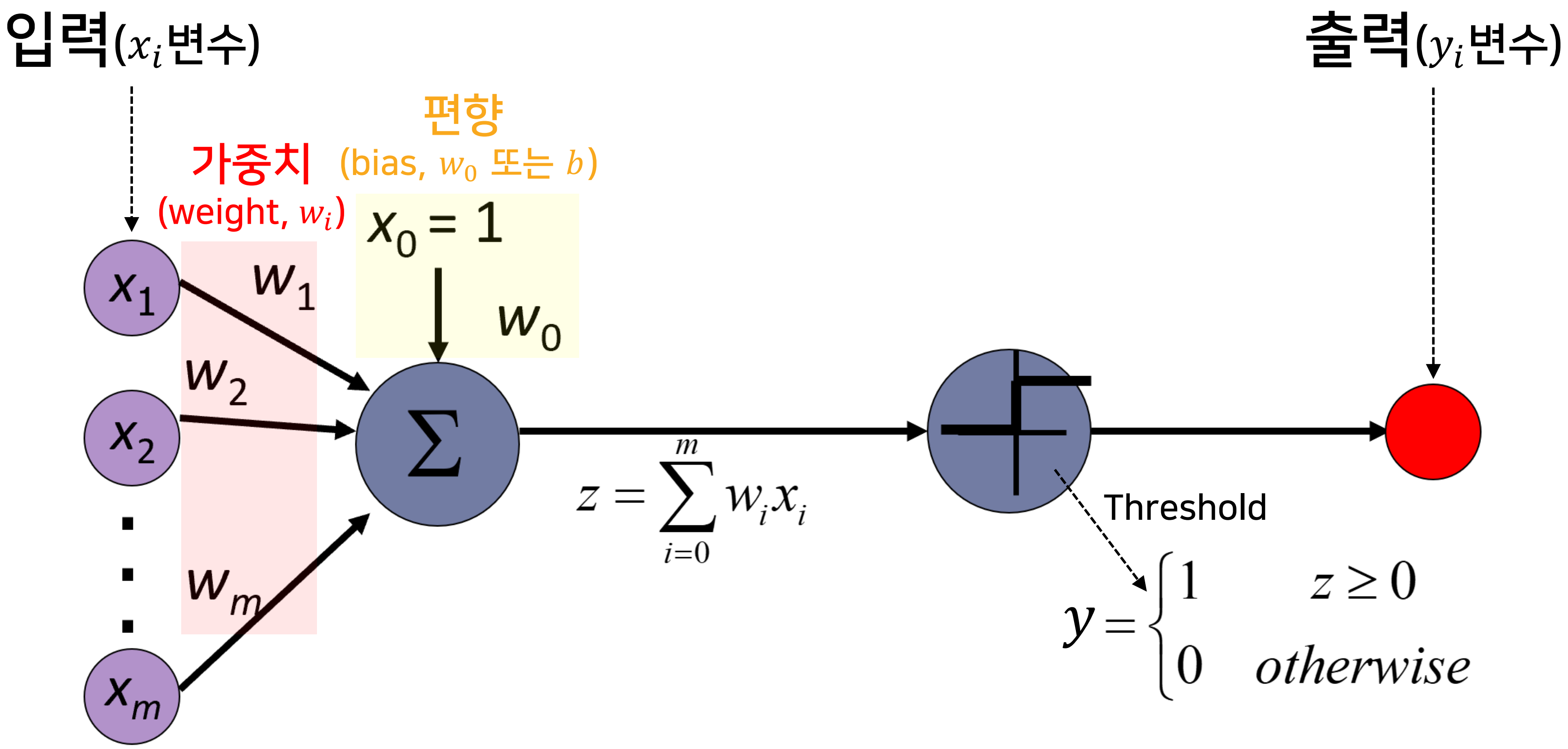
입력



출력

출처: Scott A. Sheffield MS, General Structure of a Neuron Synapse, Get Body Smart, 2022.
출처: Roi Yehoshua, Perceptrons: The First Neural Network Model, towards data science, 2023.

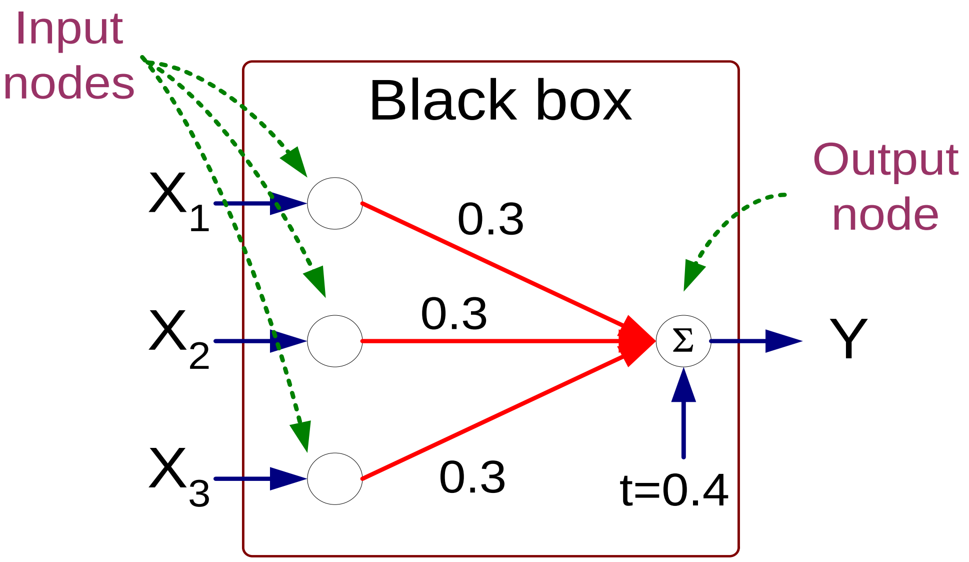
Perceptron

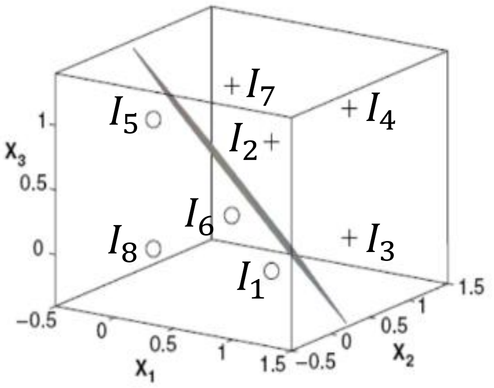


출처: Roi Yehoshua, Perceptrons: The First Neural Network Model, towards data science, 2023.

Perceptron 예제

	X_1	X_2	X_3	Y
I_1	1	0	0	0
I_2	1	0	1	1
I_3	1	1	0	1
I_4	1	1	1	1
I_5	0	0	1	0
I_6	0	1	0	0
I_7	0	1	1	1
I_8	0	0	0	0





$$Y = I(0.3X_1 + 0.3X_2 + 0.3X_3 - 0.4 > 0)$$

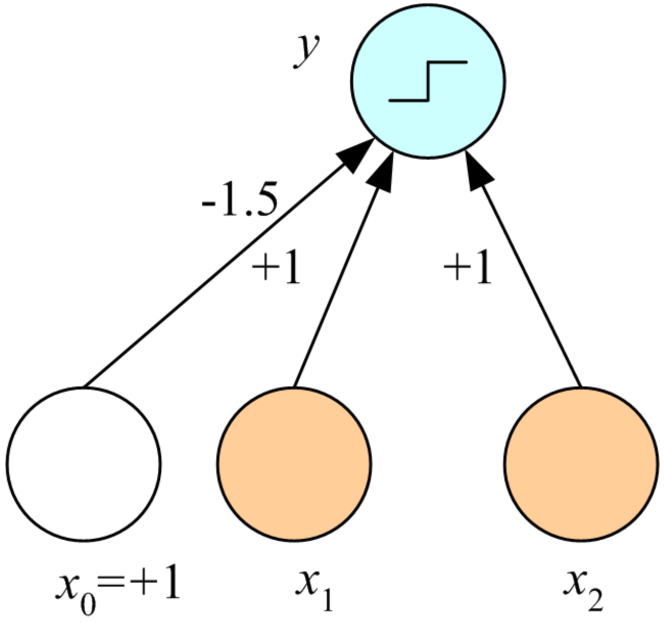
$$\text{where } I(z) = \begin{cases} 1 & \text{if } z \text{ is true} \\ 0 & \text{otherwise} \end{cases}$$

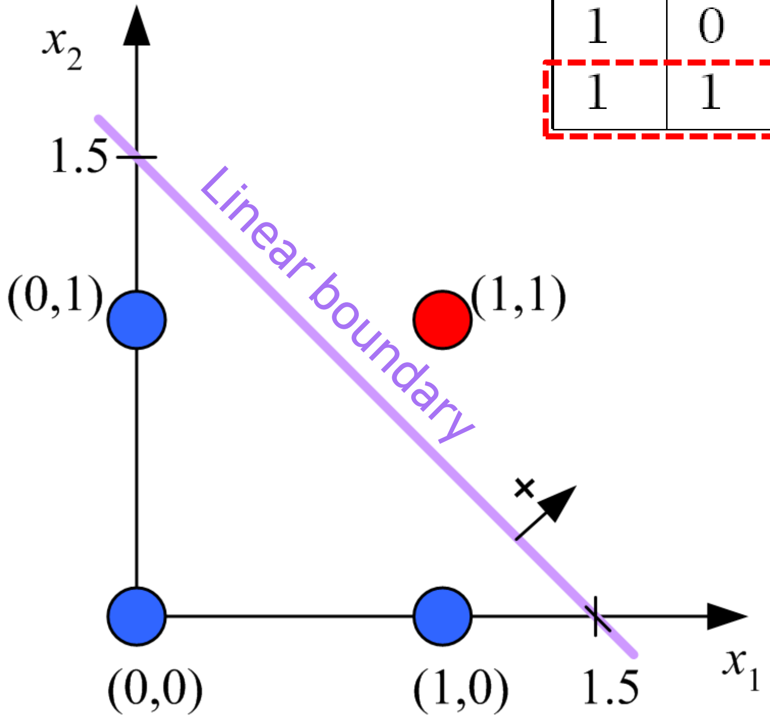
출처: P. Tan, M. Steinbach, A. Karpatne, and V. Kumar, Introduction to Data Mining, 2nd Edition.

Perceptron 예제

Boolean AND 조건

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

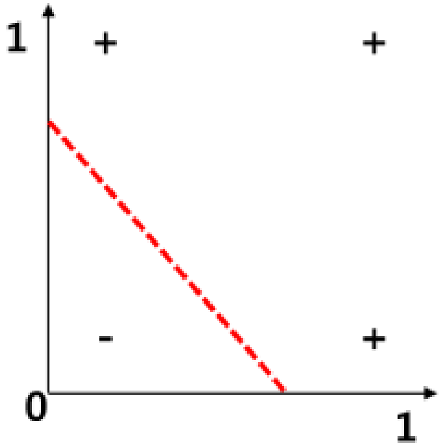




출처: P. Tan, M. Steinbach, A. Karpatne, and V. Kumar, Introduction to Data Mining, 2nd Edition.

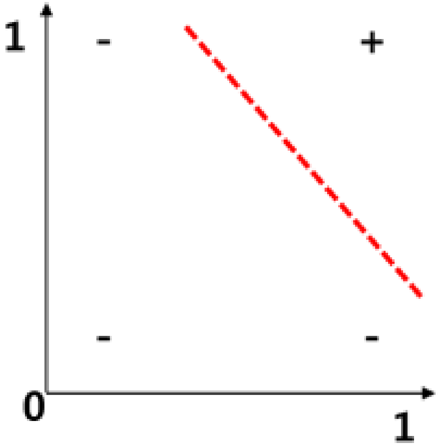
Perceptron 한계점

OR



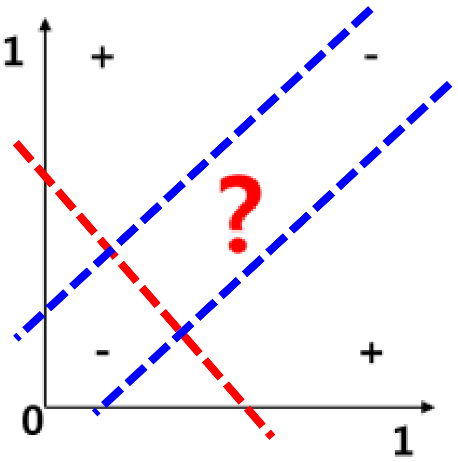
x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	1

AND



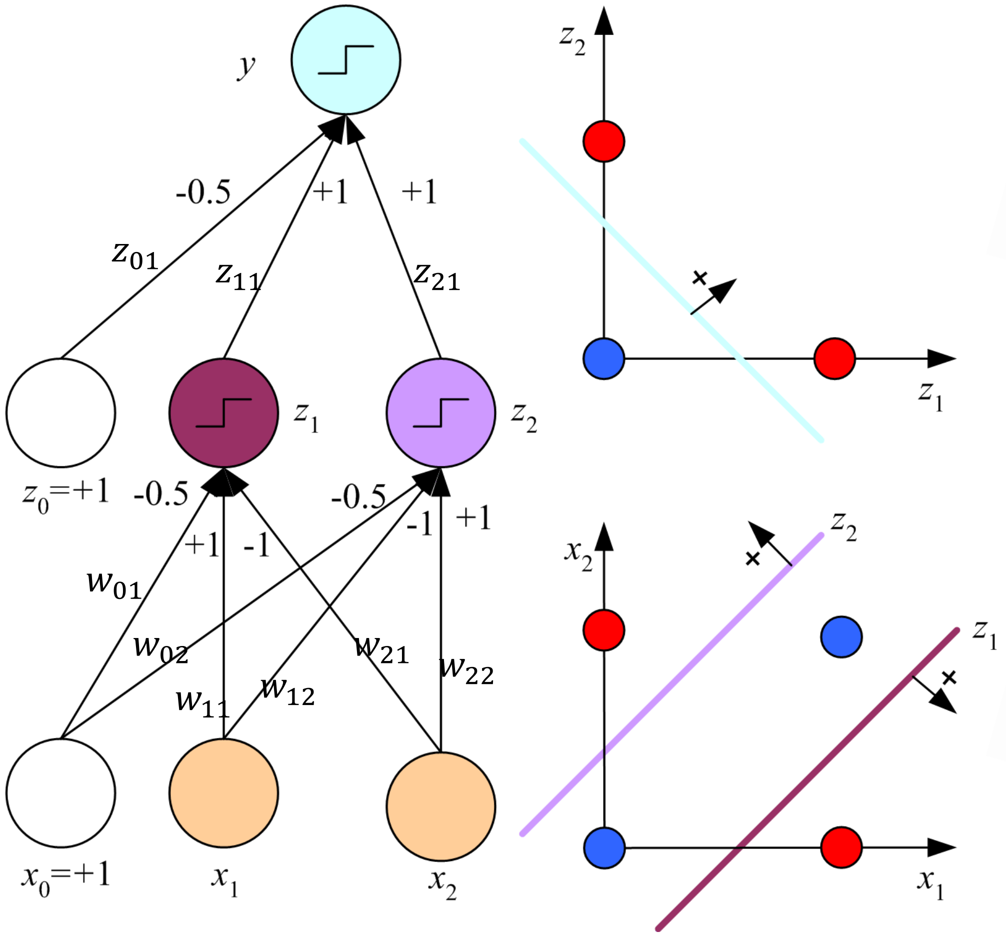
x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

XOR



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

Multilayer Perceptron 필요성



One perceptron

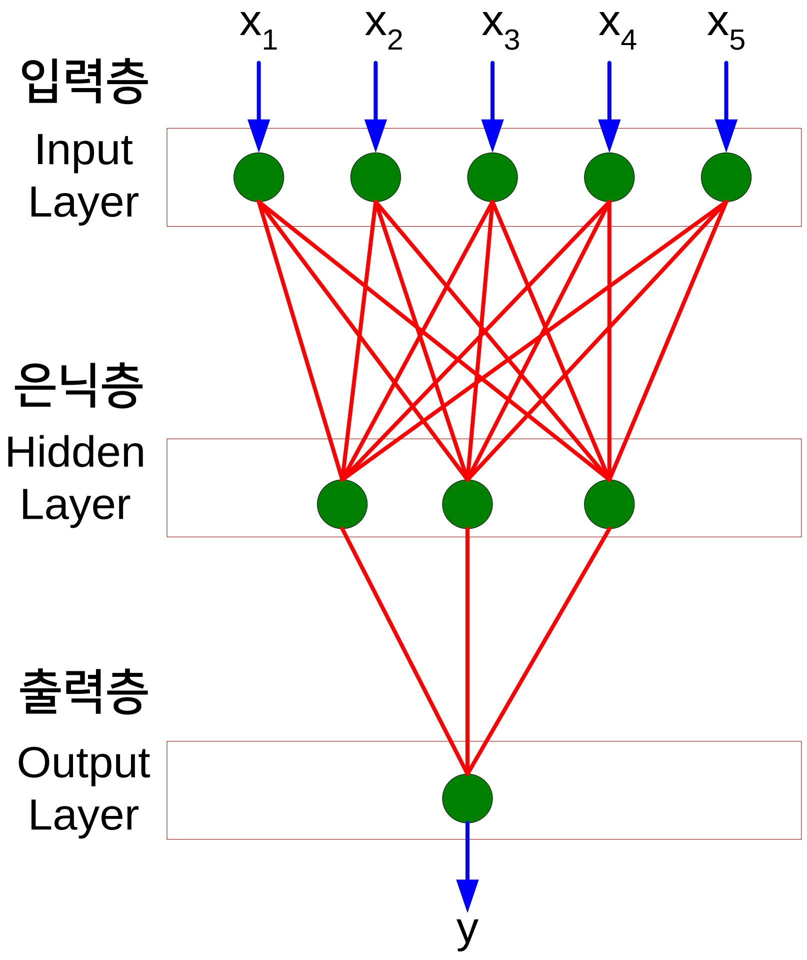
Two-input perceptron

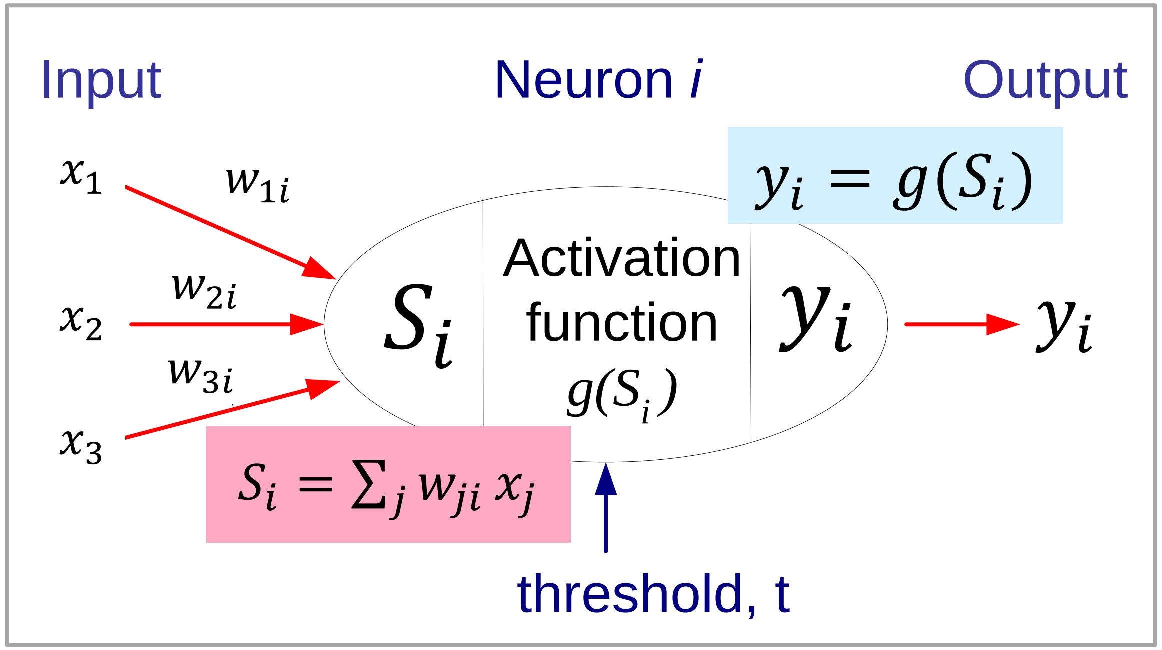
(Multilayer perceptron, MLP)

x_1 XOR x_2 = (x_1 AND ~x_2) OR (~x_1 AND x_2)

출처: P. Tan, M. Steinbach, A. Karpatne, and V. Kumar, Introduction to Data Mining, 2nd Edition.

Multilayer Perceptron 구조

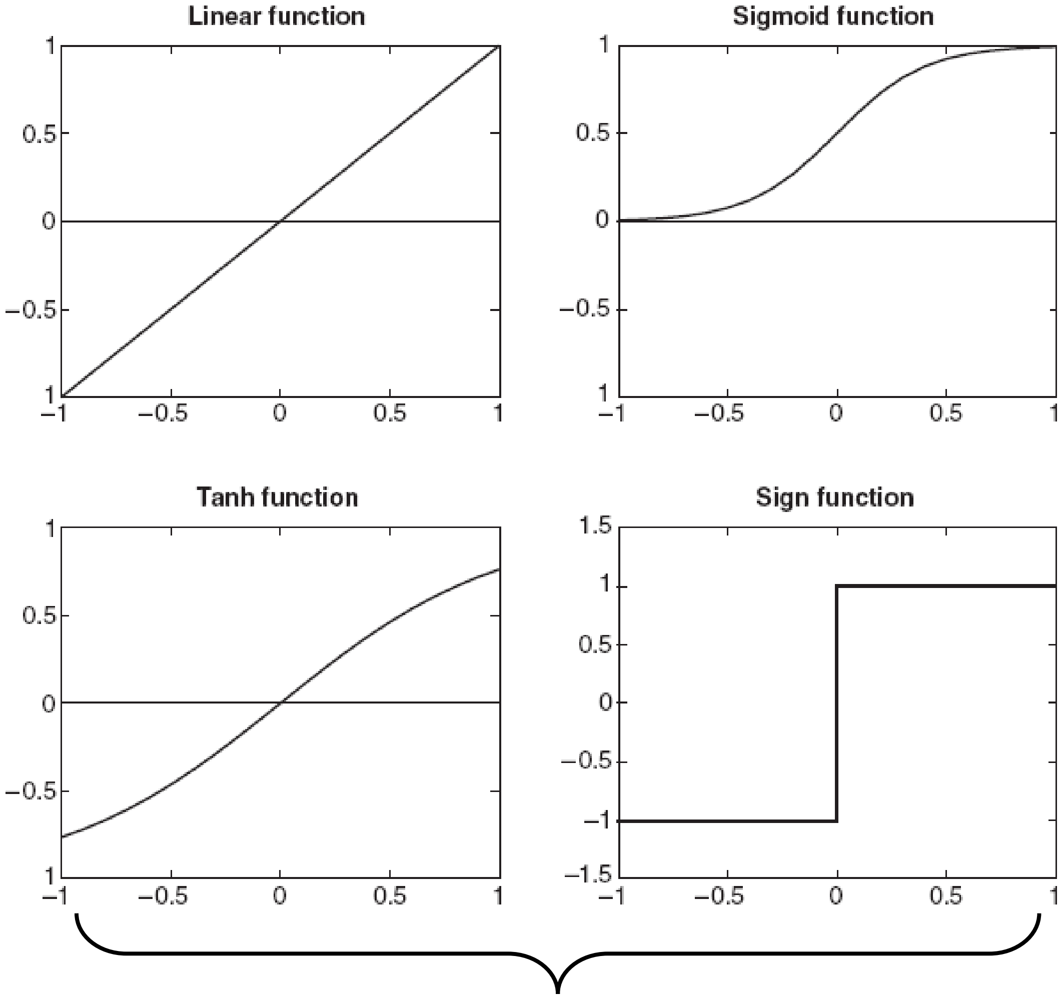




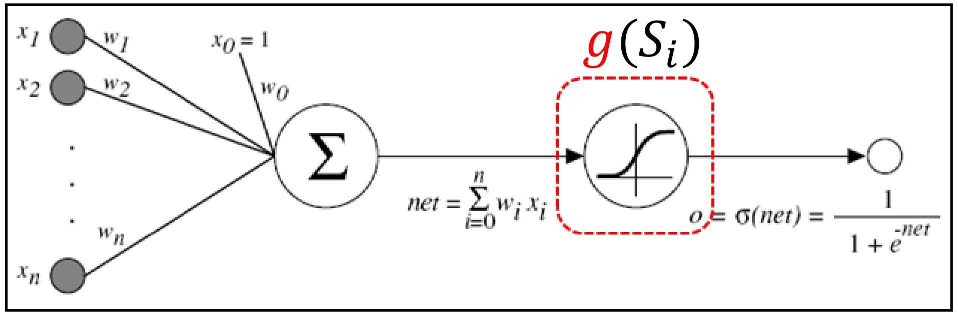
Training MLP means learning the weights

출처: P. Tan, M. Steinbach, A. Karpatne, and V. Kumar, Introduction to Data Mining, 2nd Edition.

활성화함수를 통한 비선형 변환



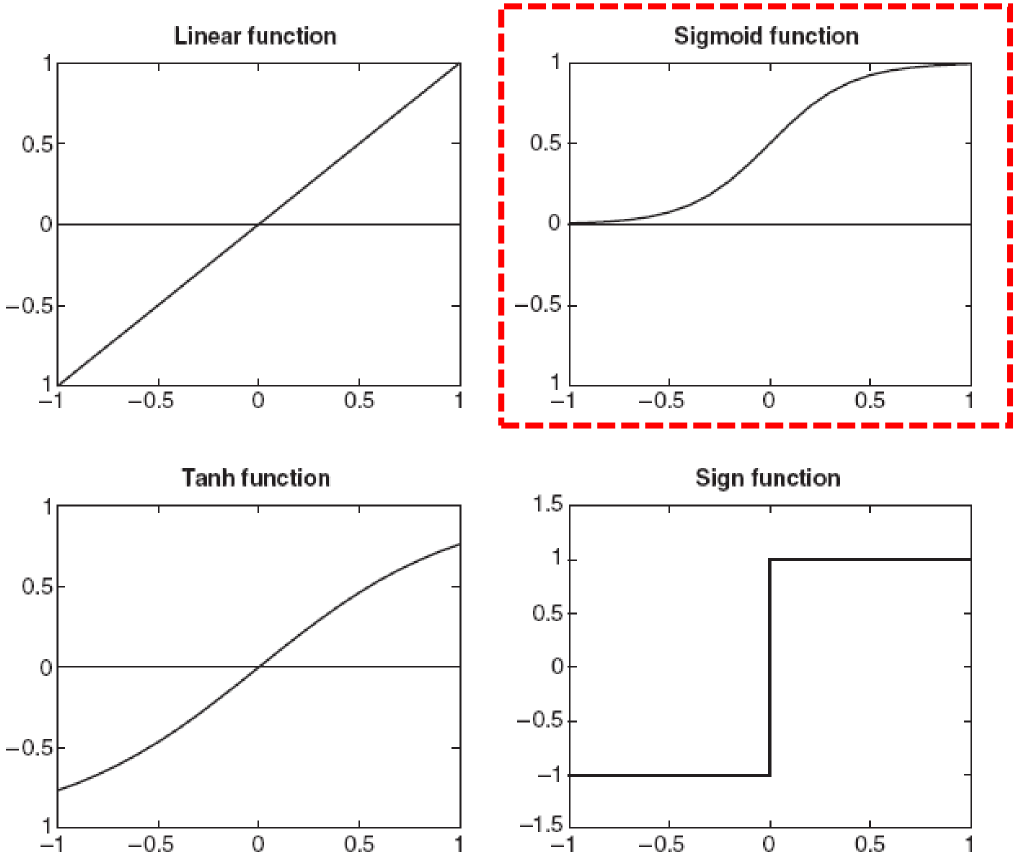
활성화 함수(Activation function)의 변형



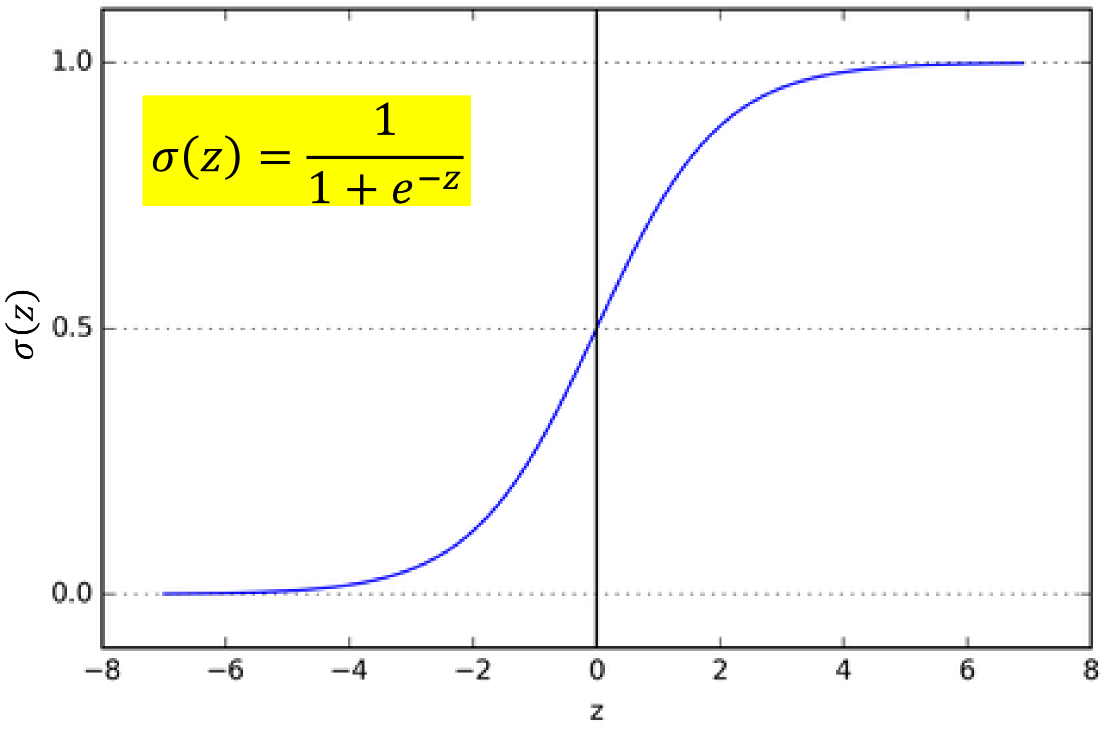
$$S_i = \sum_j w_{ji}\, x_j$$

출처: P. Tan, M. Steinbach, A. Karpatne, and V. Kumar, Introduction to Data Mining, 2nd Edition.

활성화함수를 통한 비선형 변환



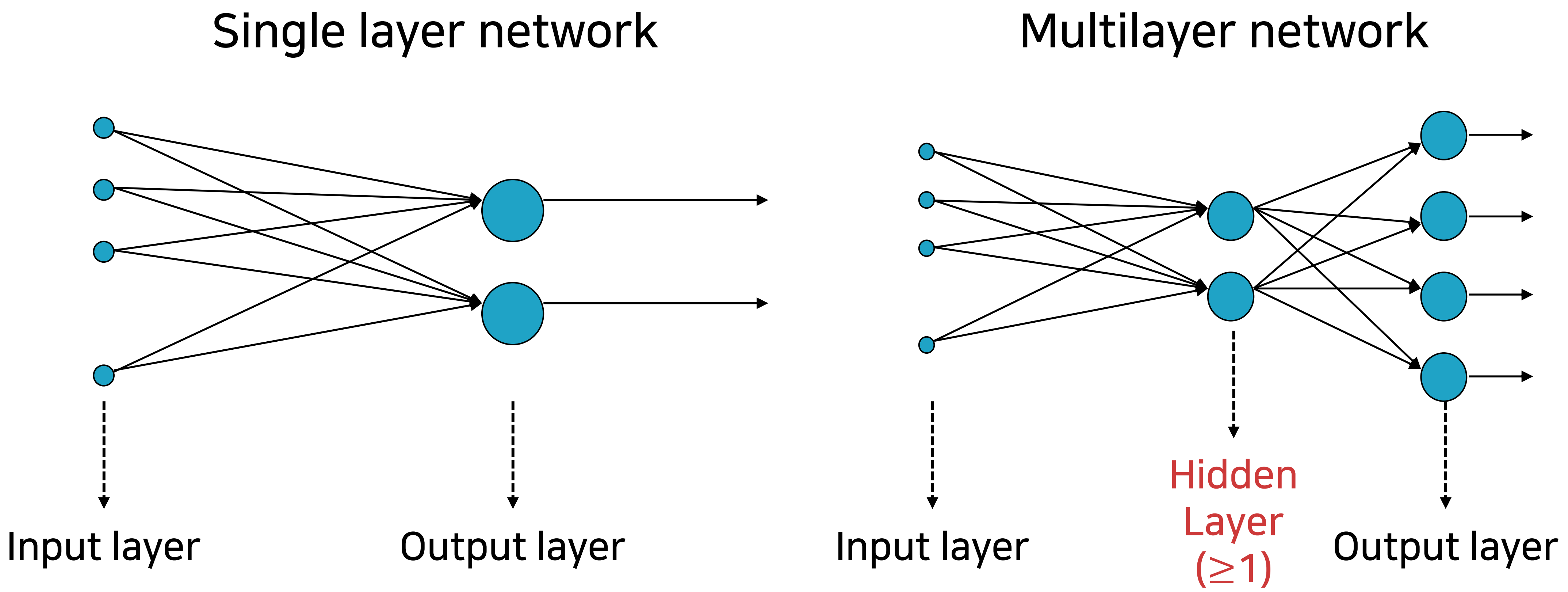
e.g. 시그모이드(Sigmoid) 함수



- 비선형 mapping function
- 큰 입력 값으로부터 작은 출력 값(0에서 1사이) 도출

출처: P. Tan, M. Steinbach, A. Karpatne, and V. Kumar, Introduction to Data Mining, 2nd Edition.

MLP 계층 구조



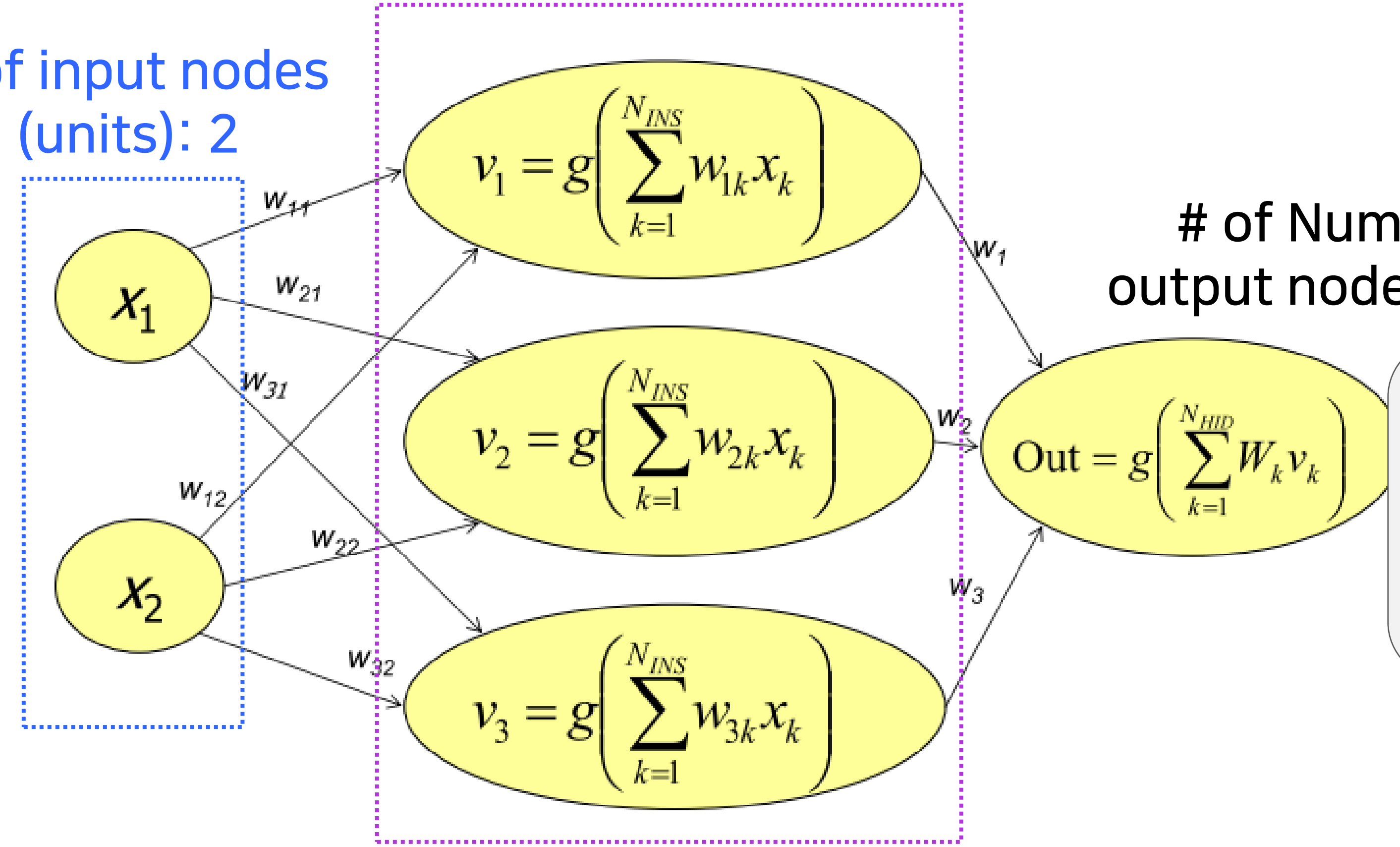
출처: Tom Mitchell, Han and Kamber, Machine Learning: Neural Networks, 1986.

MLP 계층 구조

of hidden nodes
(units) : 3

# of input nodes
(units): 2

# of Number of
output node (unit): 1



I. 예측/회귀 문제:
y_i가 수치형 값일 경우
$y_i \in R^{n\times1}$

II. 분류 문제:
y_i가 이산형 범주 값일 경우
$y_i \in \{0, 1\}$

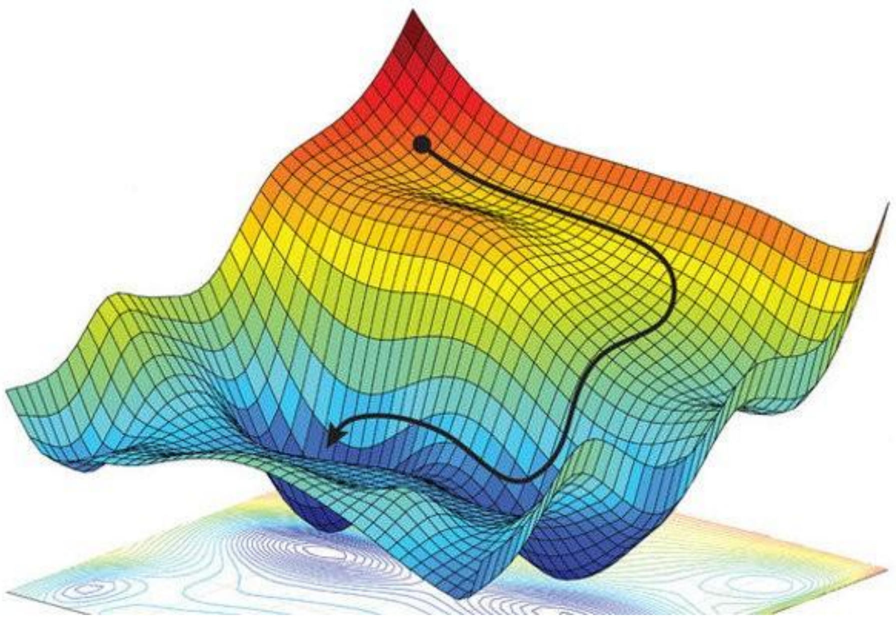
출처: Tom Mitchell, Han and Kamber, Machine Learning: Neural Networks, 1986.

Contents

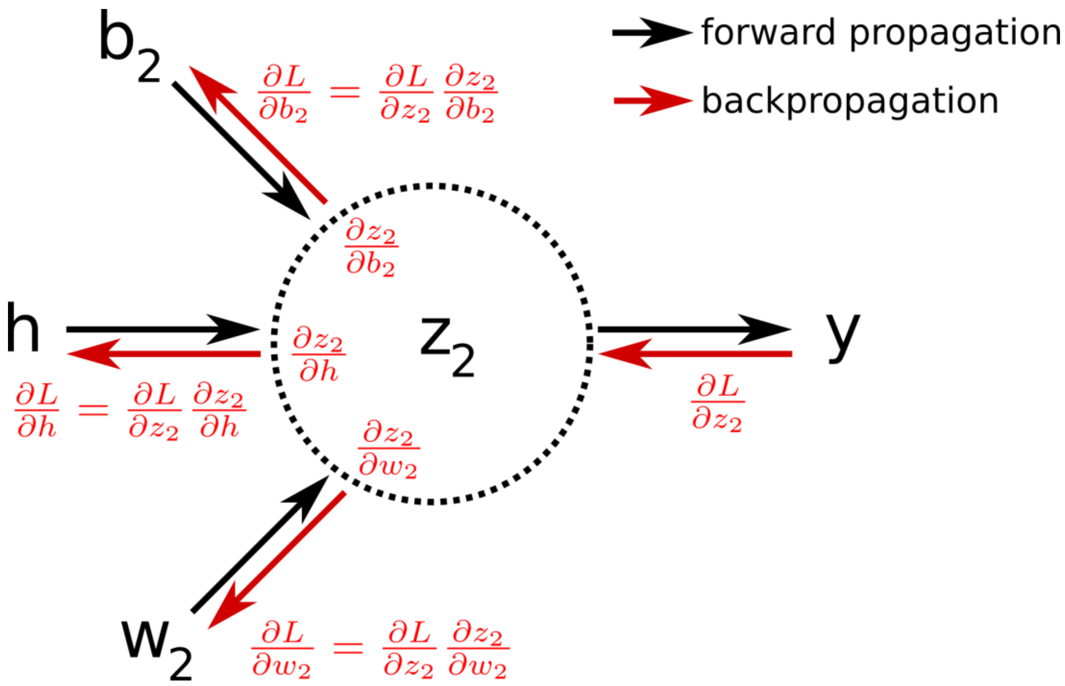
- MLP in the AI scope
- Fundamental principles of MLP
- Model training of MLP

MLP 학습 요소

손실함수
(Loss/Cost function)

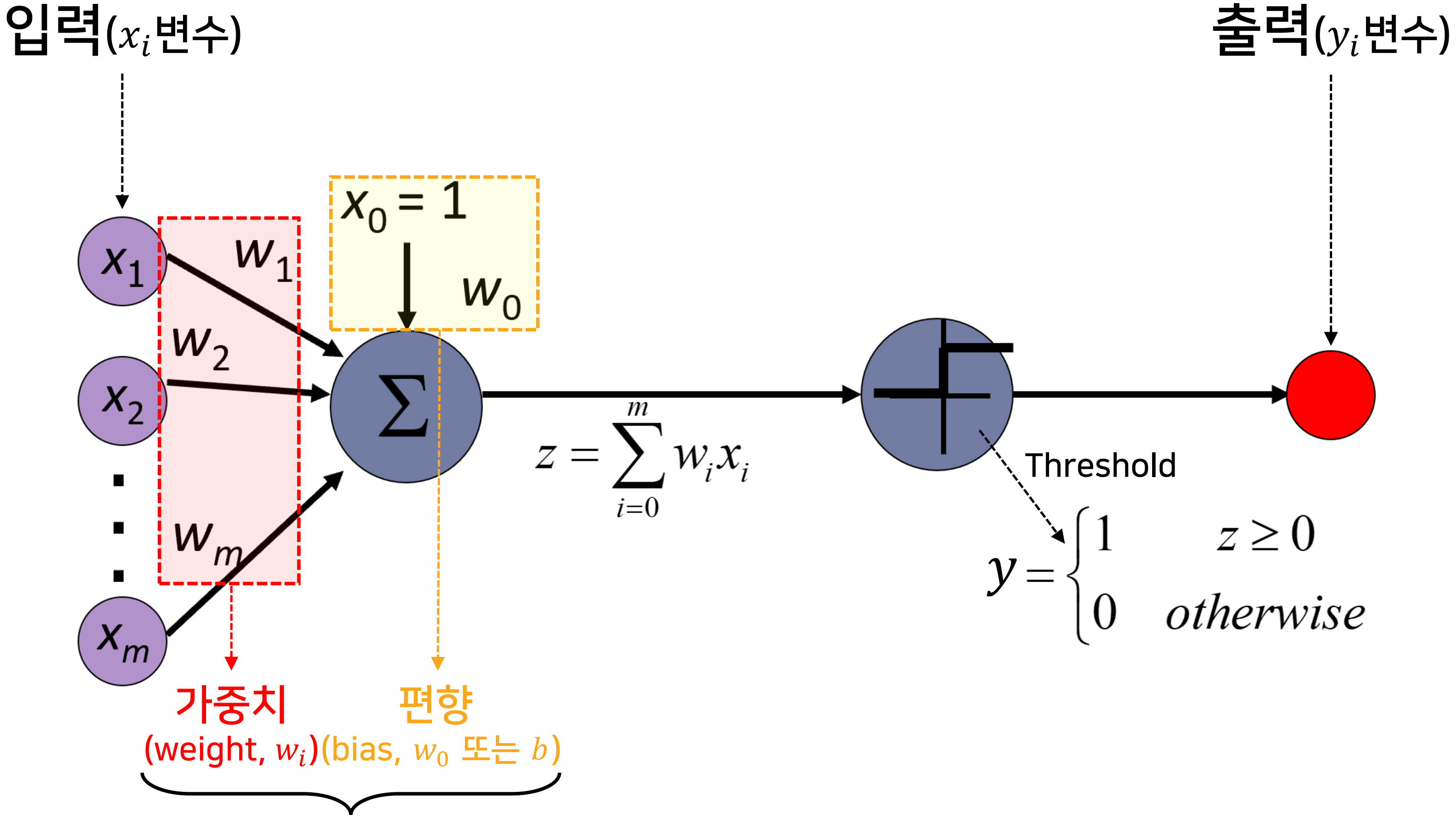


역전파
(Back propagation)



출처: [AI 이론] 손실 함수 한방에 끝내기 (Loss Function), kay Story, 2022.
출처: Romain Thalineau, An introduction to backpropagation, qwertee, 2018.

AI 학습의 목적



최적화 대상 ⟶ “최적”의 기준은?

출처: Roi Yehoshua, Perceptrons: The First Neural Network Model, towards data science, 2023.

AI 학습의 목적

"좋은" 모델 ⇒ 일반화된 모델

① 학습(Train)/검증(Test) 데이터 나누기

독립변수 (features, x)	종속변수 (label, y)	학습 데이터
		검증 데이터

② 학습 데이터를 사용해 모델 학습 ($f(x)$ 함수 찾기)

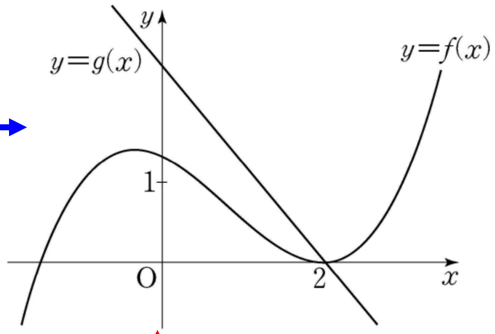
독립변수 (features, x)	종속변수 (label, y)

학습 데이터 → f 함수 찾기

③ 테스트 데이터를 사용해 모델 일반화 성능 확인

검증 데이터

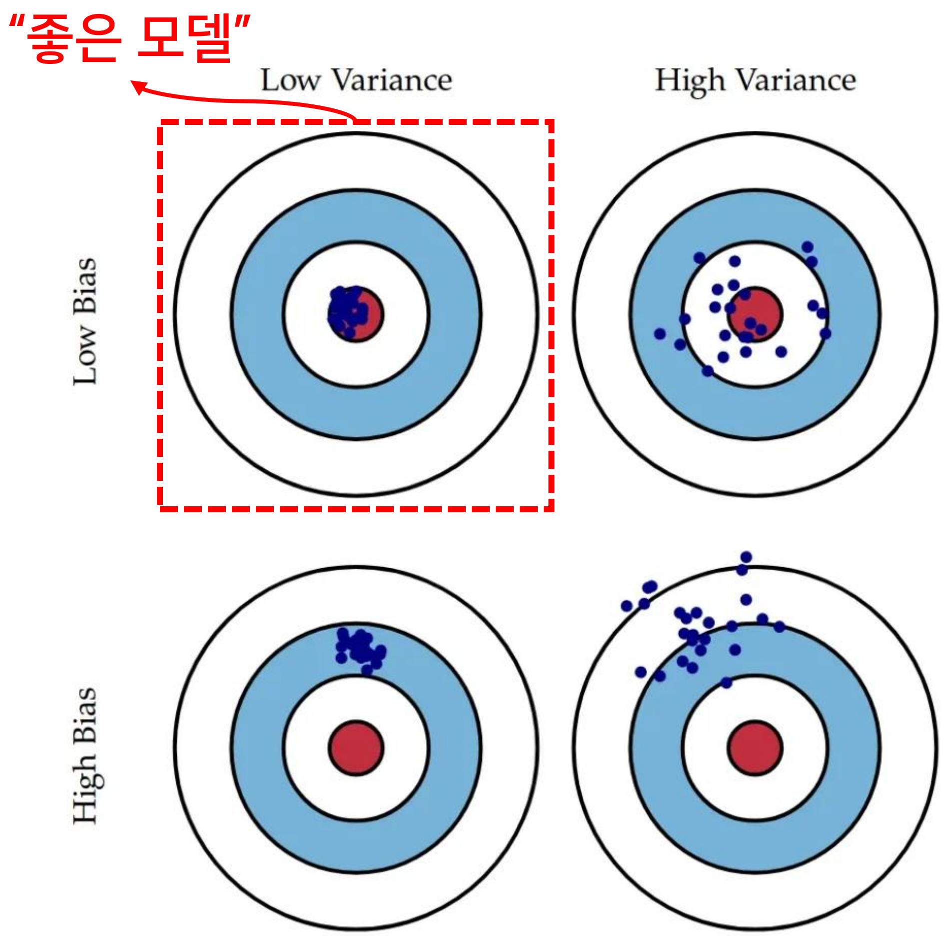
Regression model

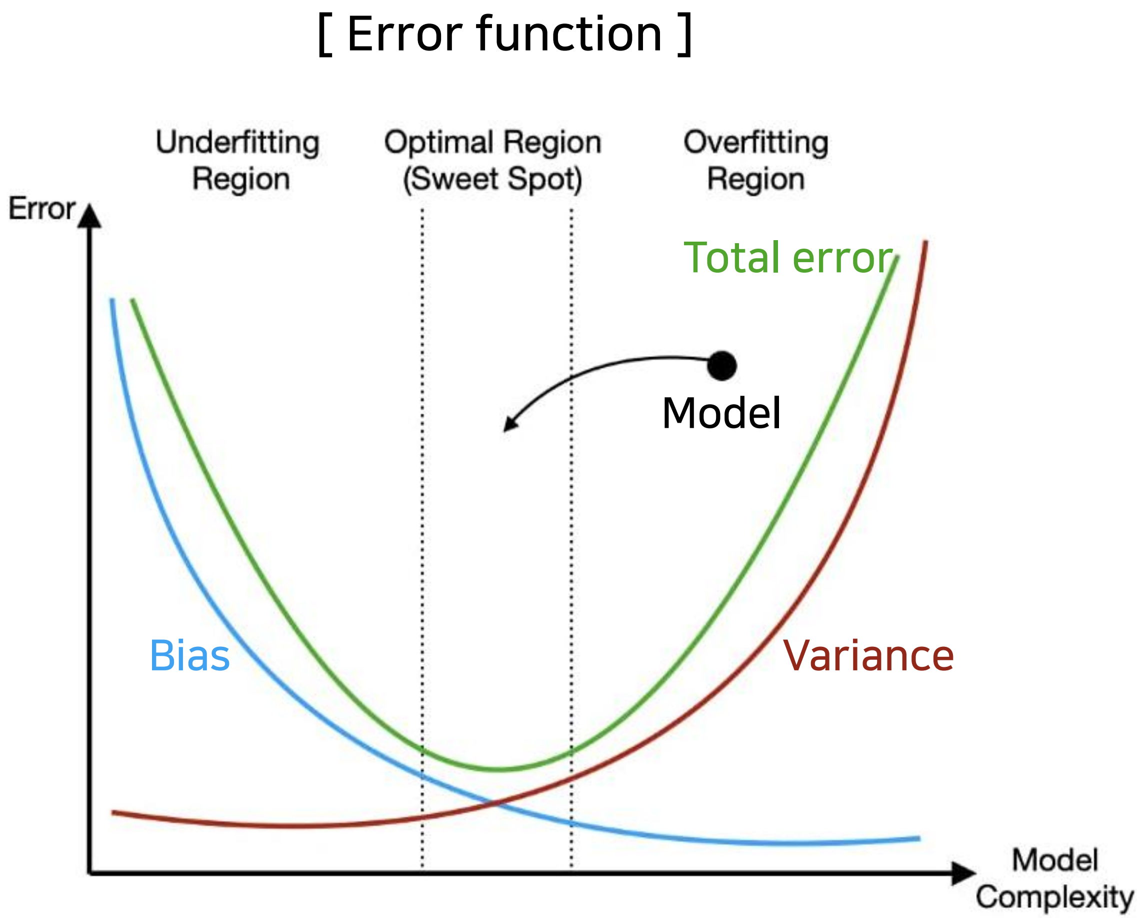


모델의 일반화(generalization) 능력을 높이기 위해,

예측 값과 실제 값 간의 오차 줄이기

AI 학습의 목적





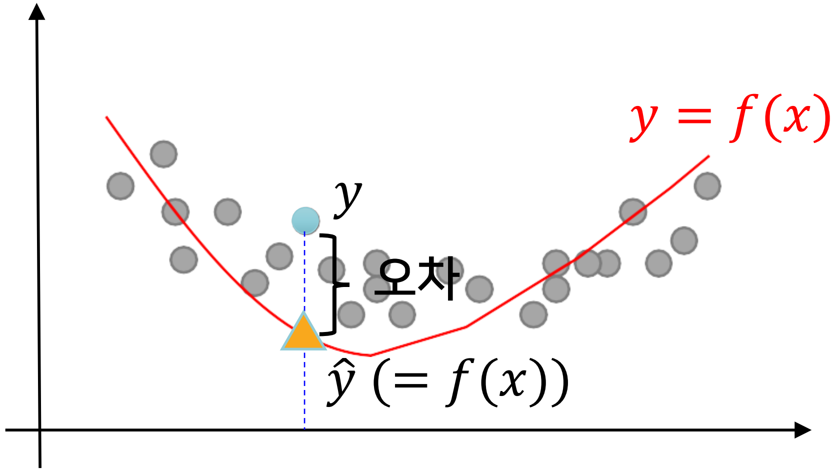
출처: Mohit Uniyal, Bias and Variance in Machine Learning, Applied ROOTS, 2024.
출처: Edwin Dong, Regularization in Deep Learning, 2024.

손실함수 정의

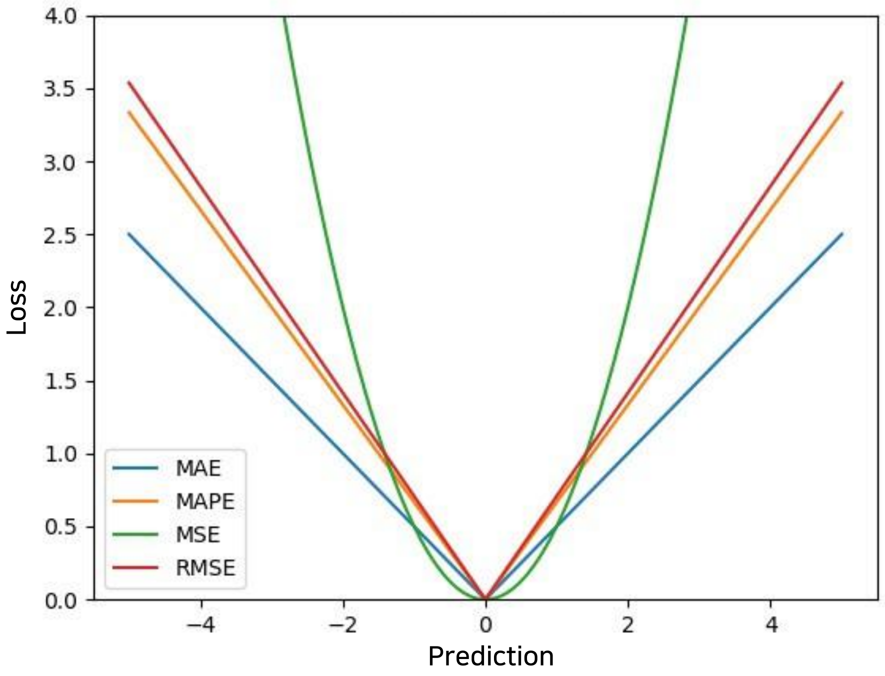
Error function
Loss function
Cost function
Objective function

[회귀/예측 문제]

목적: 예측 값과 실제 값 간의 오차 줄이기



수치 값의 차이가 중요!



Mean Absolute Error (MAE)	$\mathcal{L}_{MAE} = \frac{1}{N}\sum_{i=1}^{N}\lvert y_i - f(x_i)\rvert$
Mean Squared Error (MSE)	$\mathcal{L}_{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - f(x_i))^2$

출처: Rohan Hirekerur, A Comprehensive Guide To Loss Functions — Part 1 : Regression, Analytics Vidhya, 2020.

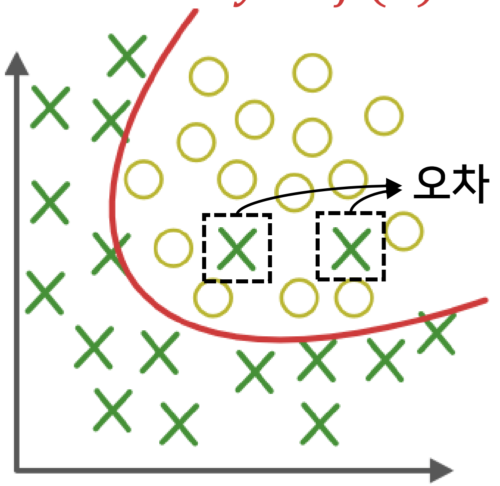
손실함수 정의

Error function
Loss function
Cost function
Objective function

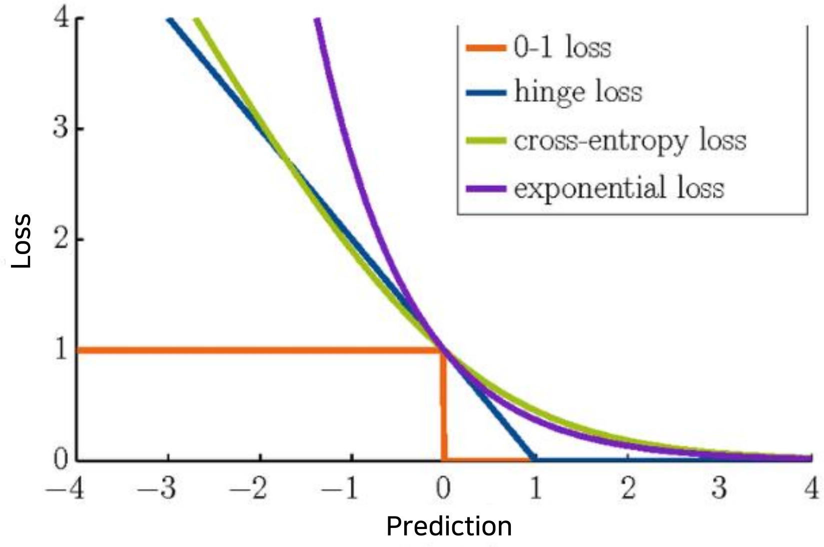
[분류 문제]

목적: 예측 값과 실제 값 간의 오차 줄이기

$y = f(x)$



★명목형 값의 차이(count)가 중요!



Binary Cross Entropy (BCE)	$\mathcal{L}_{BCE} = \frac{1}{N}\sum_{i=1}^{N} y_i log(p(x_i)) + (1-y_i) log(1-p(x_i))$
Cross Entropy (CE)	Samples # N, M Classes # $\mathcal{L}_{CE} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij} log(f(x_{ij}))$

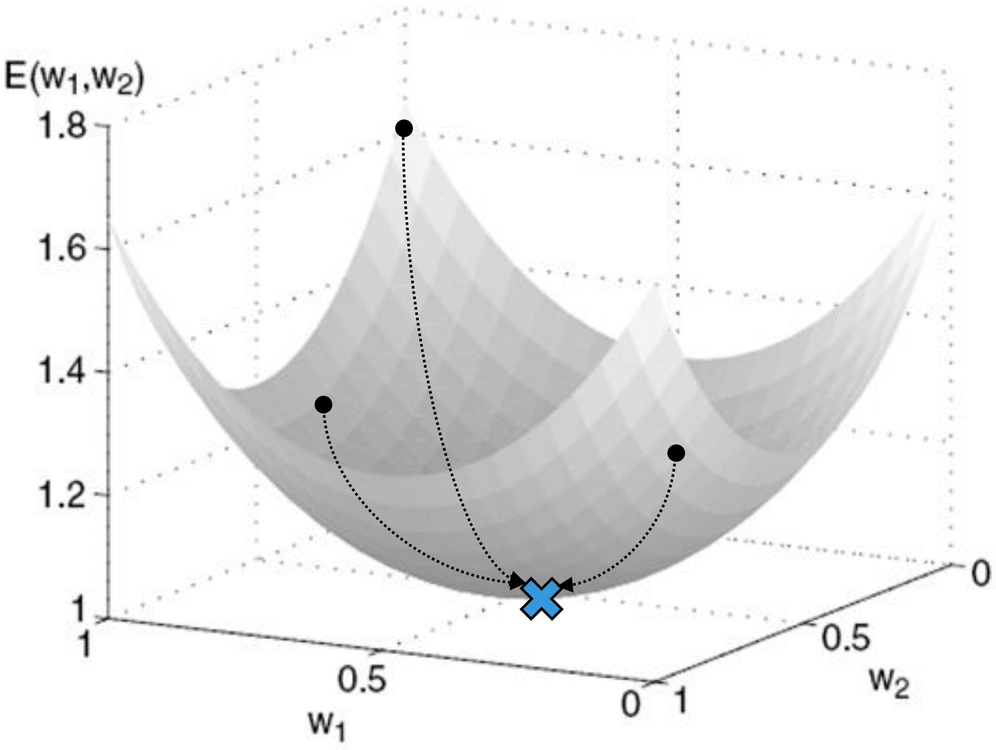
출처: Aditya Naik, Understanding Different Loss Functions for Dense Neural Networks, LinkedIn, 2023.

손실 및 가중치 Update

Model parameter (가중치, 편향) 최적화란, 함수의 최솟값 혹은 최댓값을 찾는 것

(Weight: 가중치, Bias: 편향)

Error surface E($\boldsymbol{w_1}$, $\boldsymbol{w_2}$) for a two-parameter model



최적화 알고리즘

(e.g. Gradient descent method)

weight update formula

$$w_{t+1} = w_t \underbrace{- \alpha \frac{\partial E(\mathbf{w})}{\partial w}}_{\Delta w}$$

α: Learning rate

$0 < \alpha < 1$

출처: P. Tan, M. Steinbach, A. Karpatne, and V. Kumar, Introduction to Data Mining, 2nd Edition

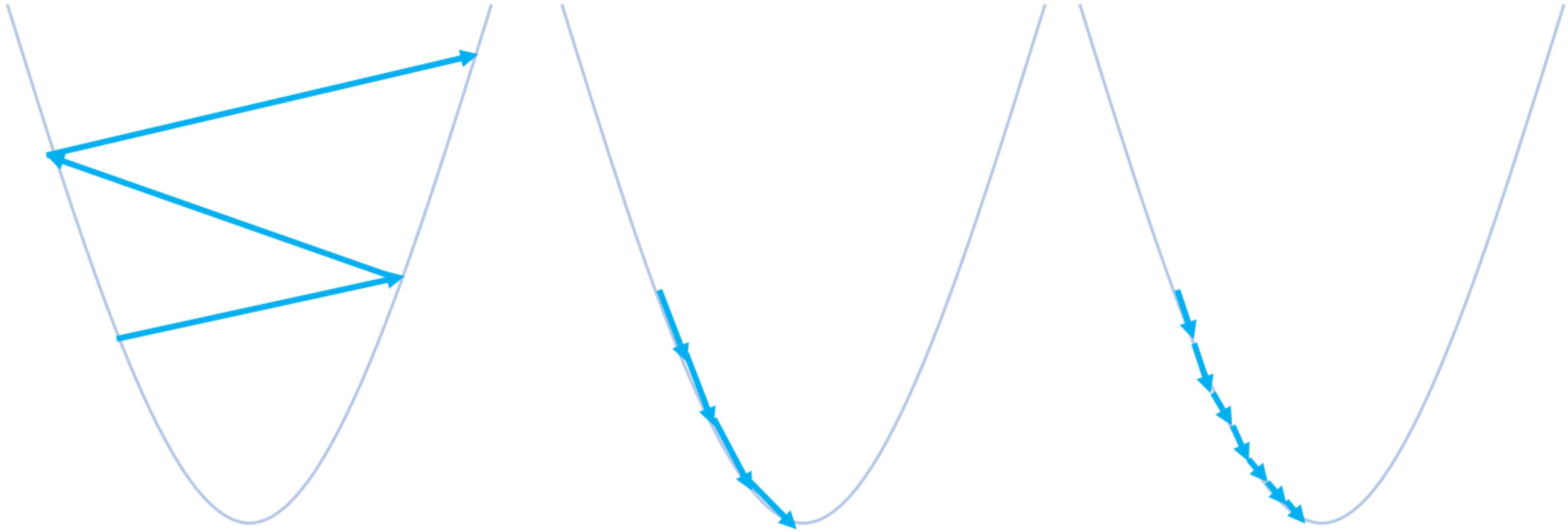
손실 및 가중치 Update

Large learning rate

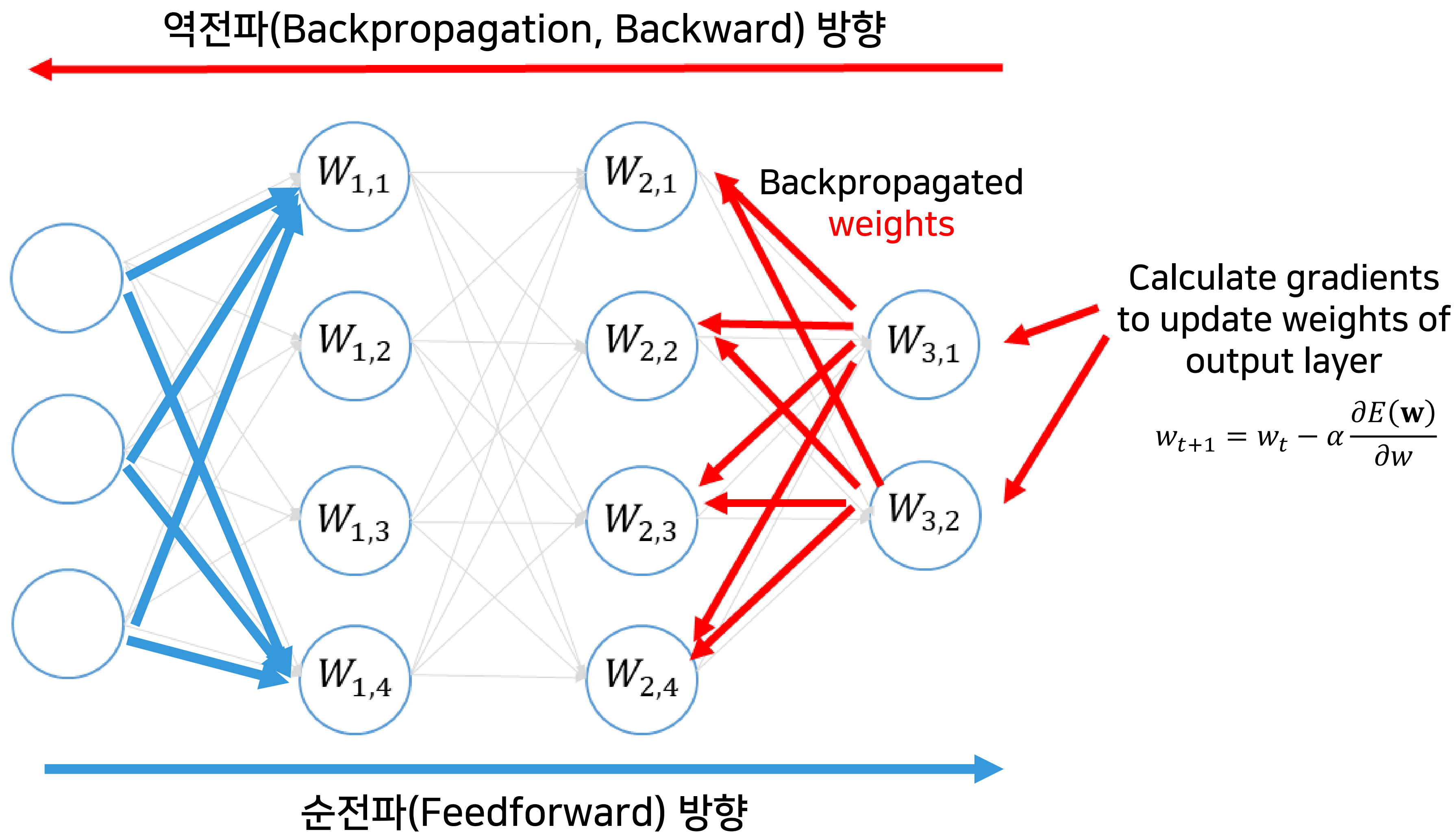
Appropriate learning rate

사용자가 지정하는 hyperparameter

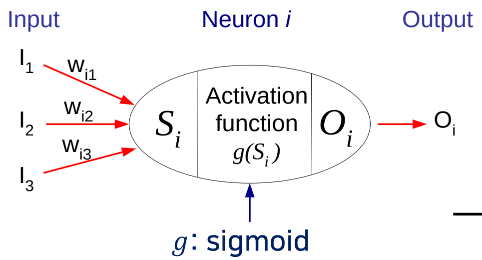
Small learning rate



순전파와 역전파

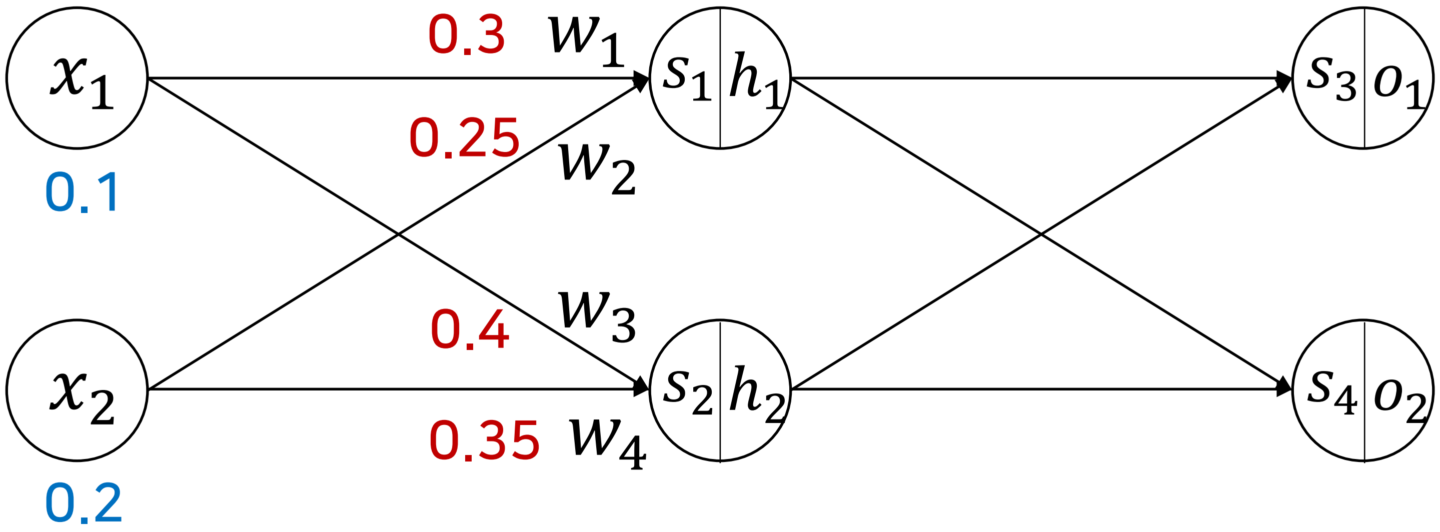


출처: Adil Lheureux, Feed-forward vs feedback neural networks, DigitalOcean, 2024.



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

순전파 예제

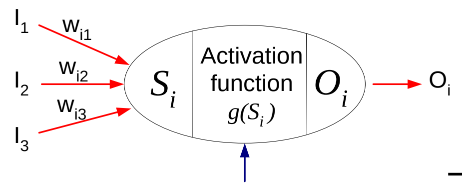


$$s_1 = w_1 x_1 + w_2 x_2 = 0.3 \times 0.1 + 0.25 \times 0.2 = 0.08$$

$$s_2 = w_3 x_1 + w_4 x_2 = 0.4 \times 0.1 + 0.35 \times 0.2 = 0.11$$

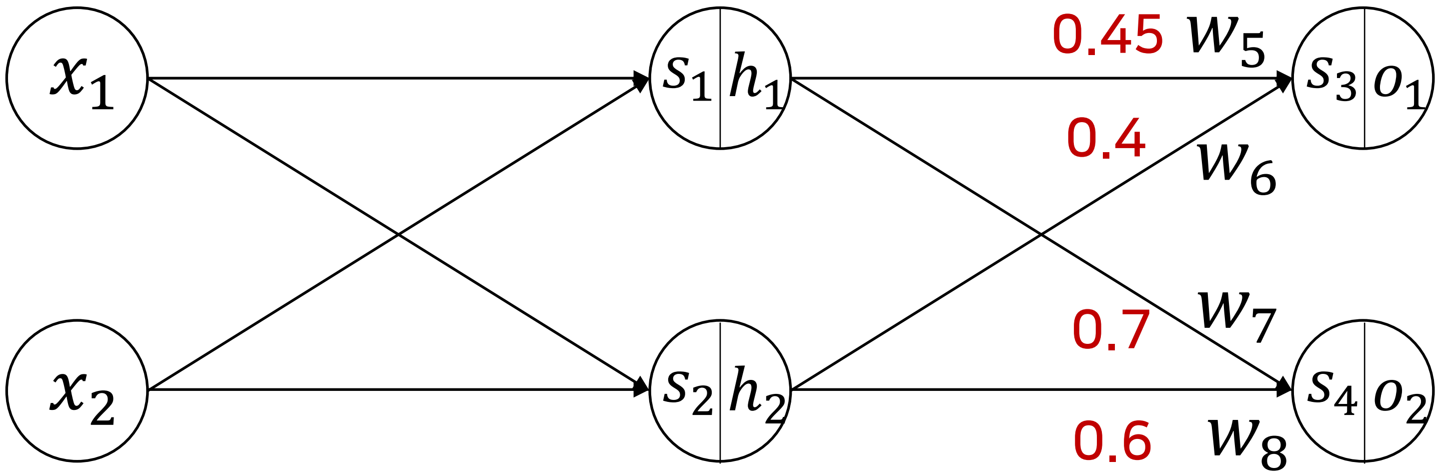
$$h_1 = sigmoid(s_1) = 0.520 \qquad h_2 = sigmoid(s_2) = 0.527$$

출처: 비전공자가 알려주는 자연어처리 기초 특강 – 12강. 순전파, 역전파, wikidocs, 2023.



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

순전파 예제

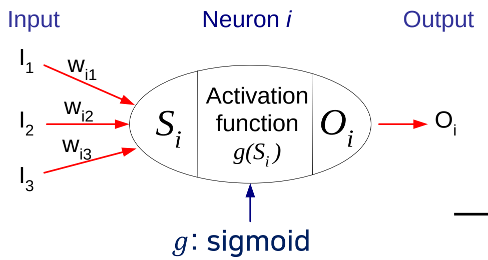


$$s_3 = w_5 h_1 + w_6 h_2 = 0.45 \times 0.520 + 0.4 \times 0.527 = 0.445$$

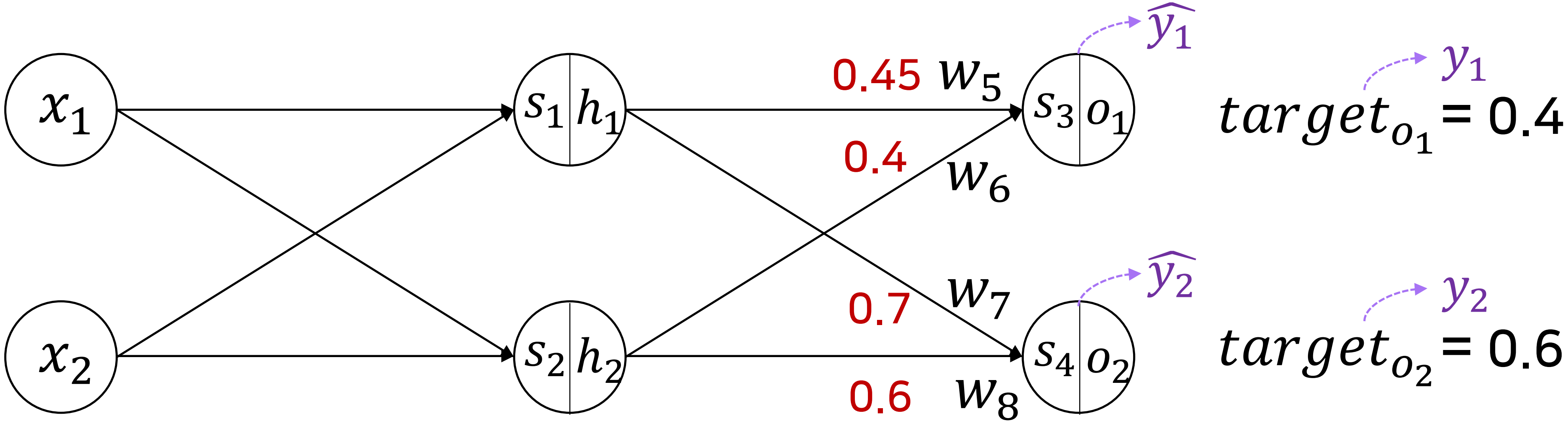
$$s_4 = w_7 h_1 + w_8 h_2 = 0.7 \times 0.520 + 0.6 \times 0.527 = 0.680$$

$$o_1 = sigmoid(s_3) = 0.609 \qquad o_2 = sigmoid(s_4) = 0.664$$

출처: 비전공자가 알려주는 자연어처리 기초 특강 – 12강. 순전파, 역전파, wikidocs, 2023.



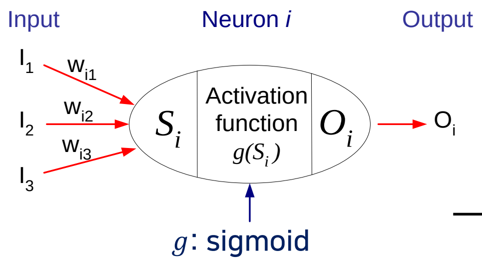
순전파 예제



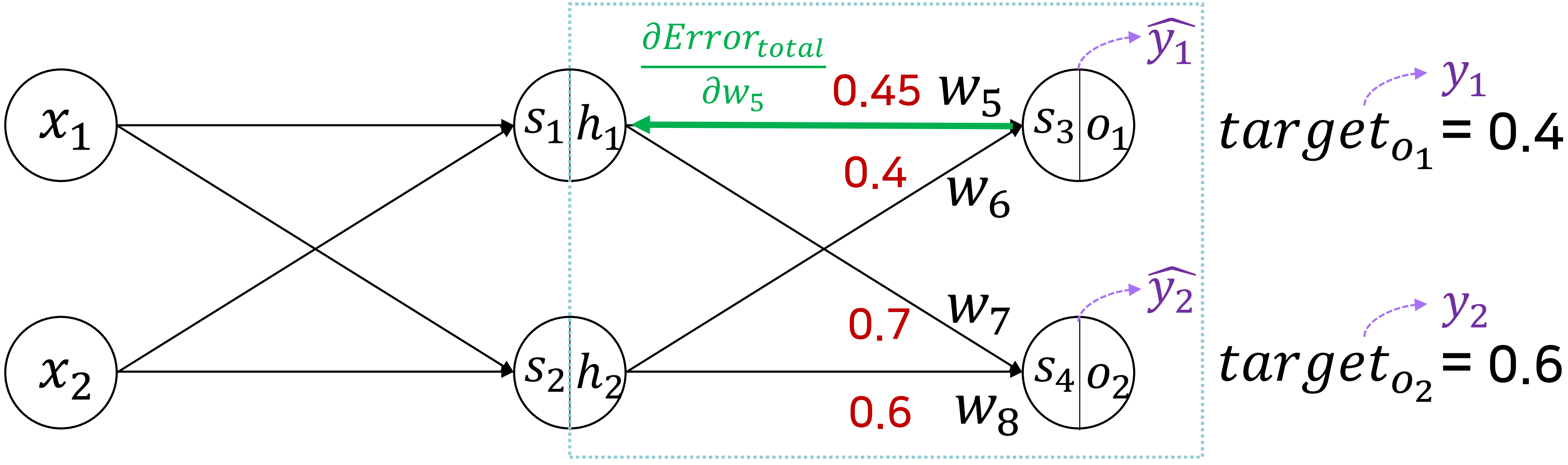
$$Error_{o_1} = \frac{1}{2}\left(target_{o_1} - output_{o_1}\right)^2 = \frac{1}{2}(0.4 - 0.609)^2 = 0.022$$

$$Error_{o_2} = \frac{1}{2}\left(target_{o_2} - output_{o_2}\right)^2 = \frac{1}{2}(0.6 - 0.664)^2 = 0.002$$

$$Error_{total} = Error_{o_1} + Error_{o_2} = 0.024$$

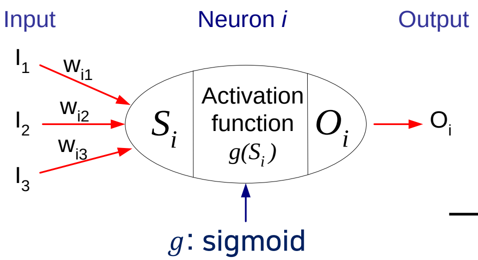


역전파 예제

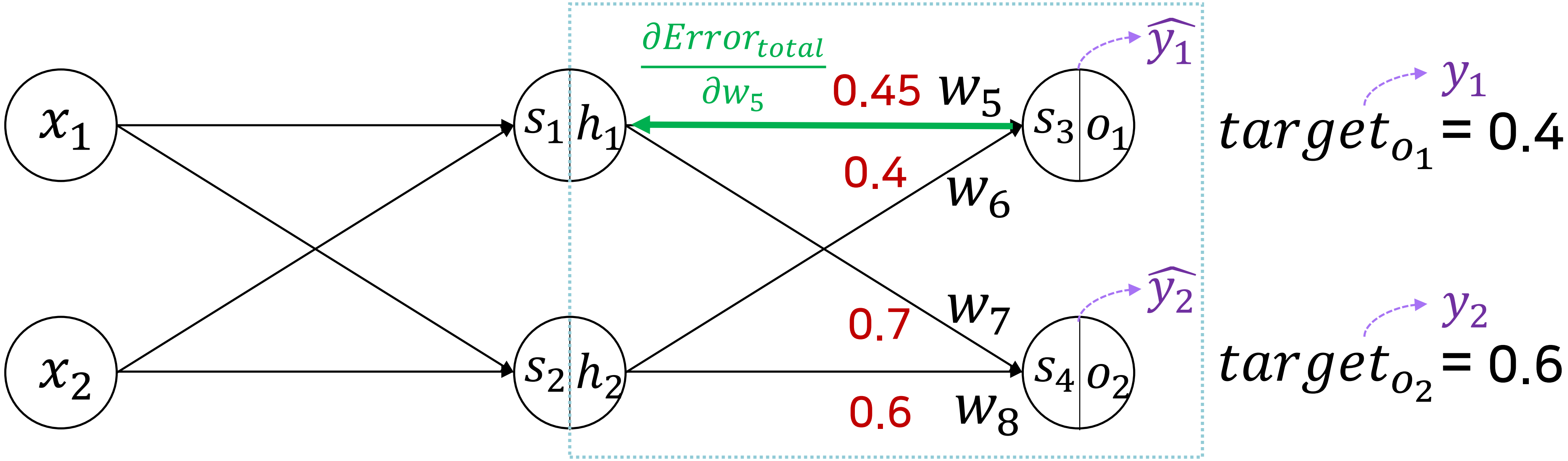


$$\frac{\partial Error_{total}}{\partial w_5} = \frac{\partial Error_{total}}{\partial o_1} \times \frac{\partial o_1}{\partial s_3} \times \frac{\partial s_3}{\partial w_5}$$

출처: 비전공자가 알려주는 자연어처리 기초 특강 – 12강. 순전파, 역전파, wikidocs, 2023.



역전파 예제

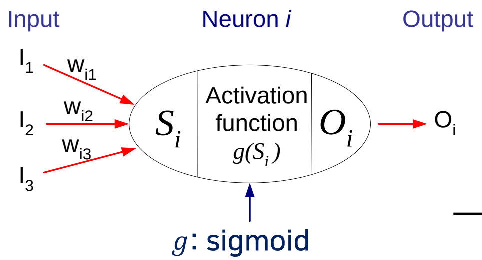


$$Error_{total} = \frac{1}{2}(target_{o_1} - output_{o_1})^2 + \frac{1}{2}(target_{o_2} - output_{o_2})^2$$

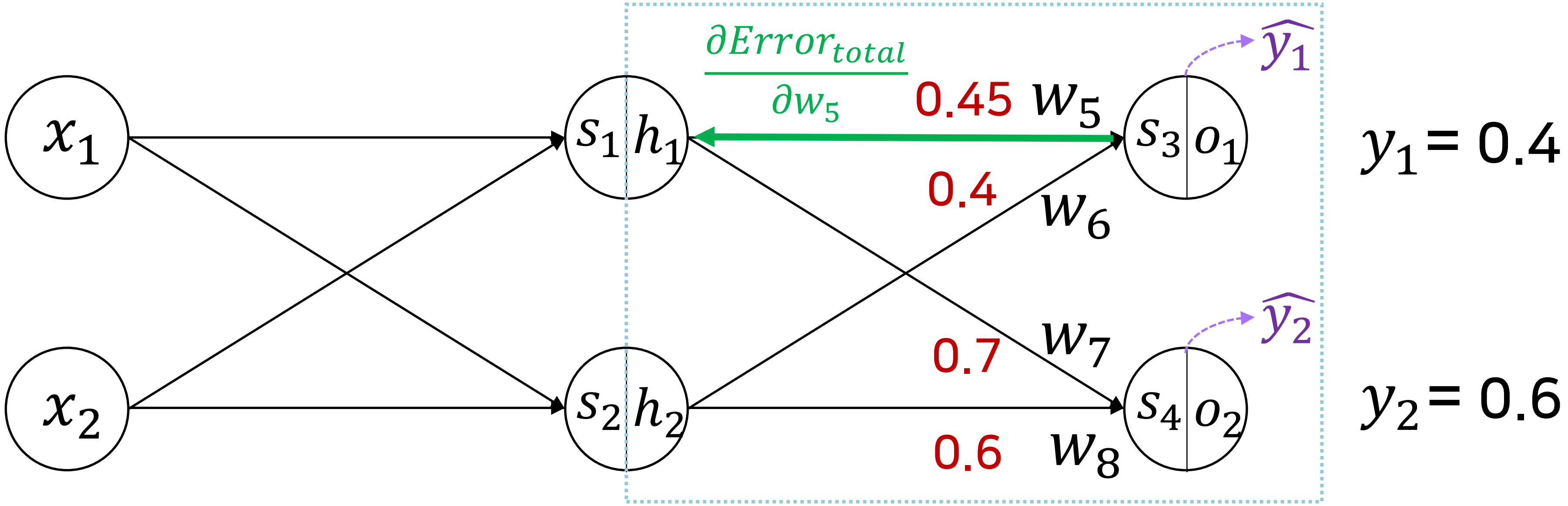
$$\frac{\partial Error_{total}}{\partial o_1} = 2 \times \frac{1}{2}(target_{o_1} - output_{o_1})^1 \times (-1) + 0$$

$$= -(0.4 - 0.609) = 0.209$$

출처: 비전공자가 알려주는 자연어처리 기초 특강 – 12강. 순전파, 역전파, wikidocs, 2023.



역전파 예제

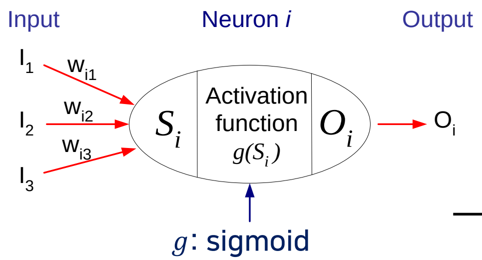


$$\frac{\partial o_1}{\partial s_3} = \{Sigmoid\text{의 미분}\}$$

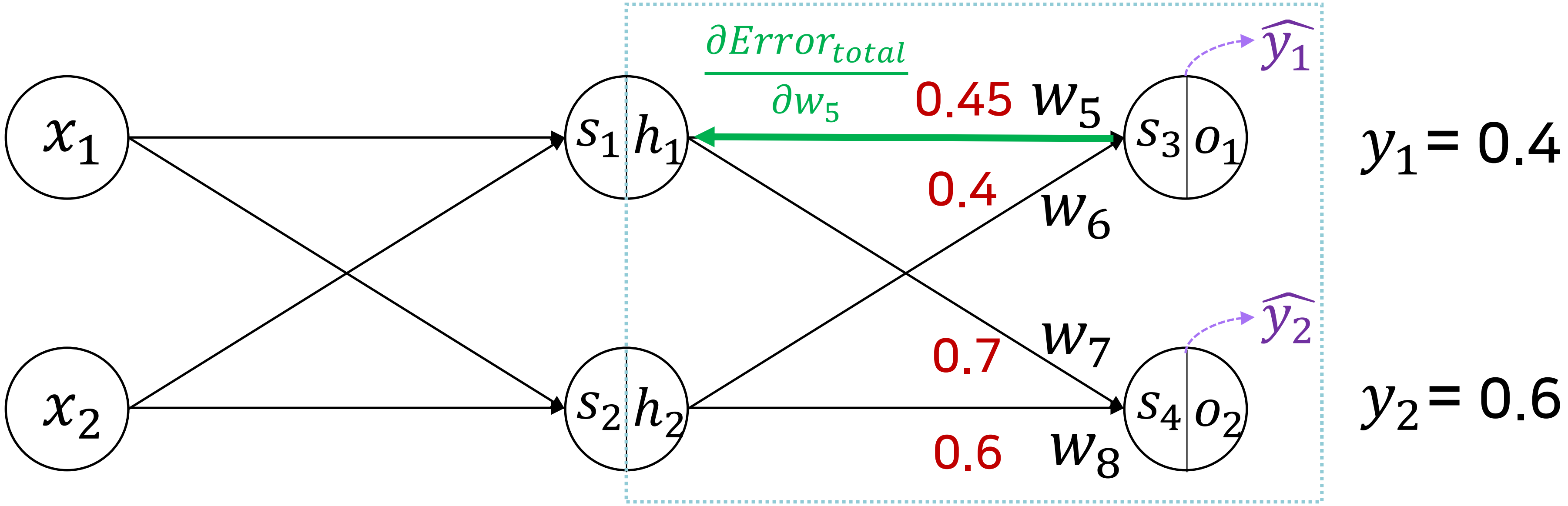
$$f(x) = \frac{1}{1+e^{-x}} \qquad f'(x) = f(x)\cdot(1-f(x))$$

$$\frac{\partial o_1}{\partial s_3} = o_1 \times (1 - o_1) = 0.609 \times (1 - 0.609) = 0.238$$

출처: 비전공자가 알려주는 자연어처리 기초 특강 – 12강. 순전파, 역전파, wikidocs, 2023.



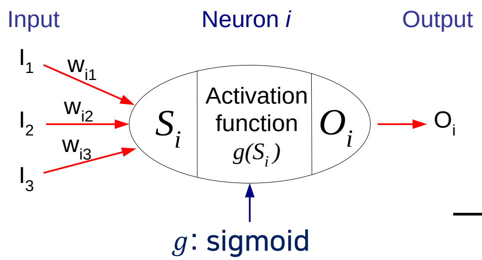
역전파 예제



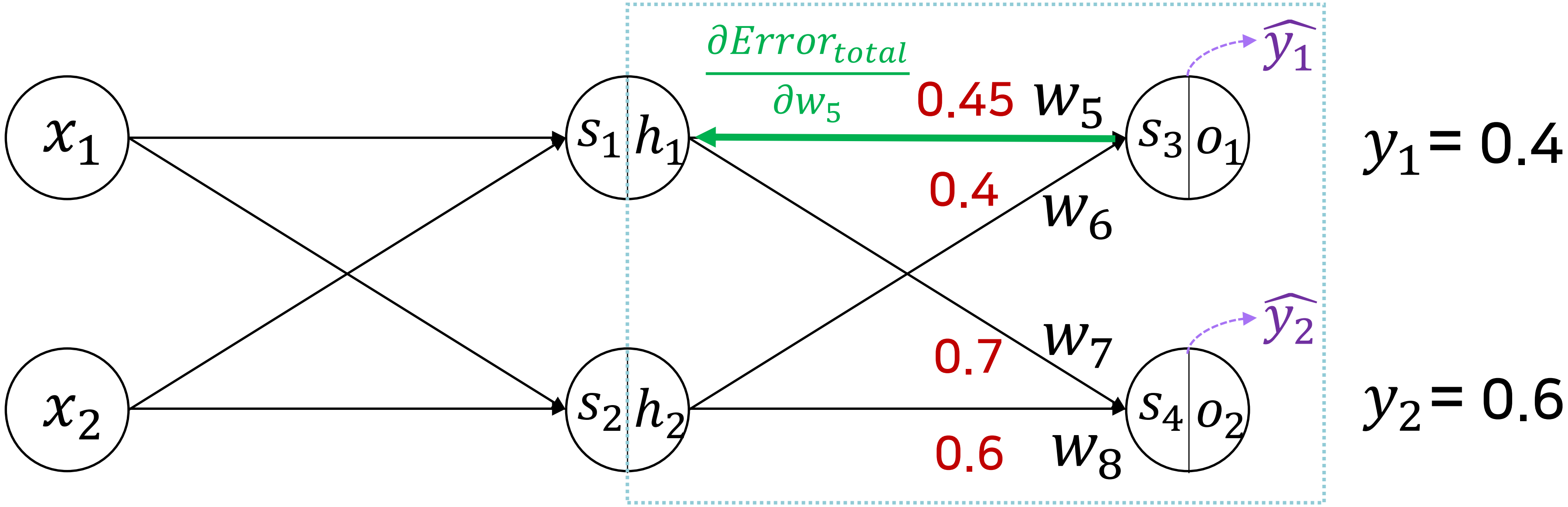
$$s_3 = w_5 \times h_1 + b_3$$

$$\frac{\partial s_3}{\partial w_5} = h_1 = 0.520$$

출처: 비전공자가 알려주는 자연어처리 기초 특강 – 12강. 순전파, 역전파, wikidocs, 2023.



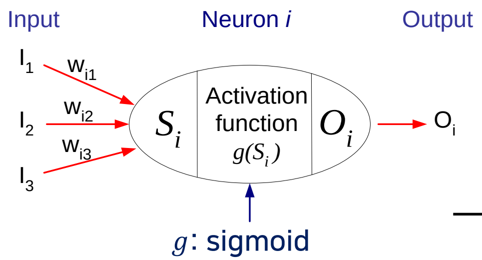
역전파 예제



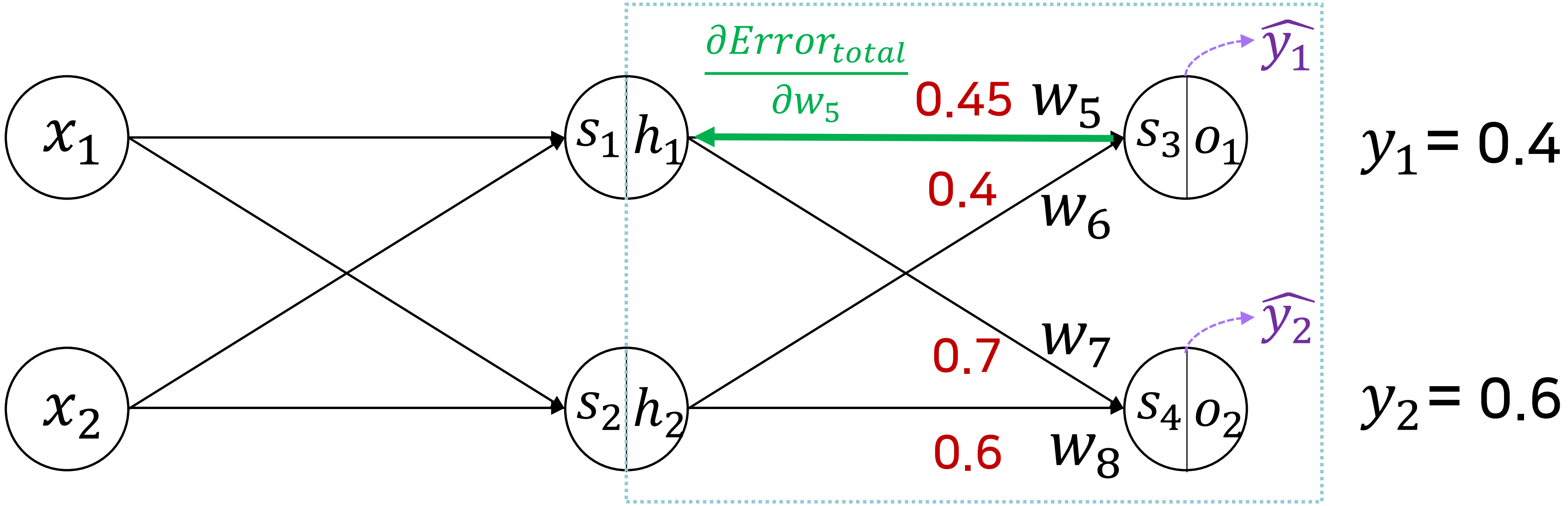
$$\frac{\partial Error_{total}}{\partial w_5} = \frac{\partial Error_{total}}{\partial o_1} \times \frac{\partial o_1}{\partial s_3} \times \frac{\partial s_3}{\partial w_5}$$

$$= 0.209 \times 0.238 \times 0.520 = 0.026$$

출처: 비전공자가 알려주는 자연어처리 기초 특강 – 12강. 순전파, 역전파, wikidocs, 2023.



역전파 예제



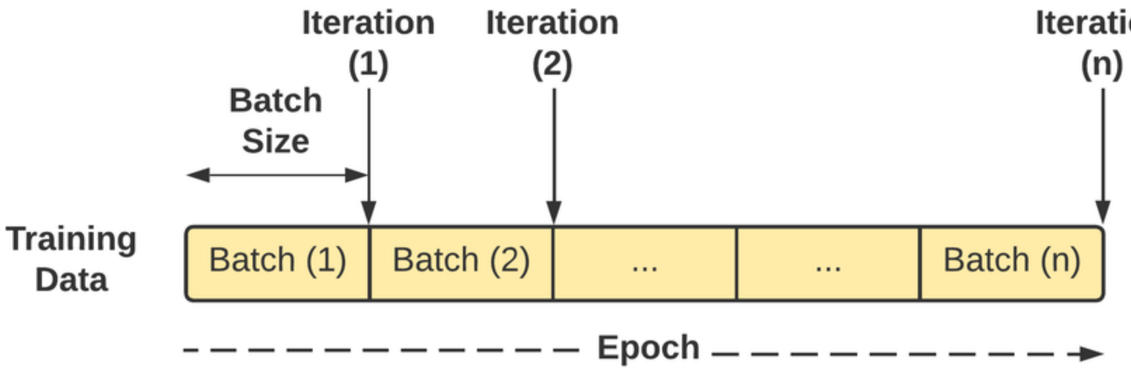
Updated weight

$$w_5^+ = w_5 - \alpha \times \frac{\partial Error_{total}}{\partial w_5} = 0.45 - 0.5 \times 0.026 = 0.437$$

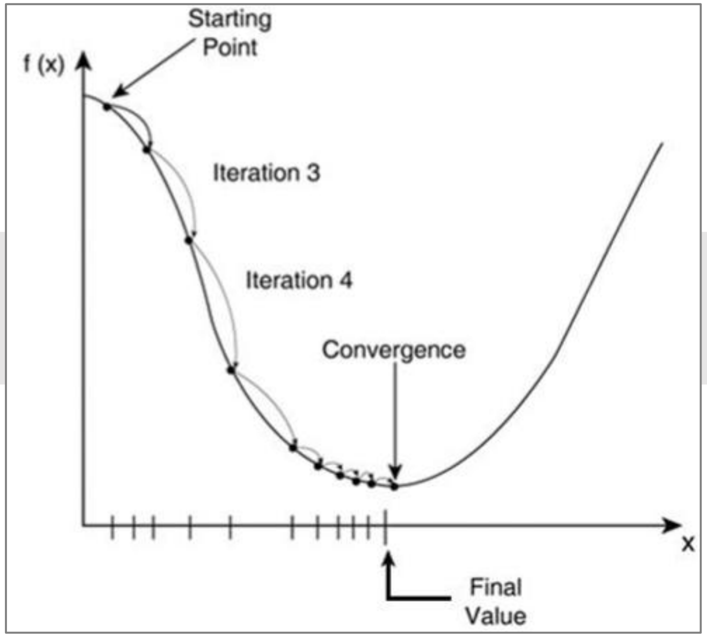
Learning rate
(hyperparameter)

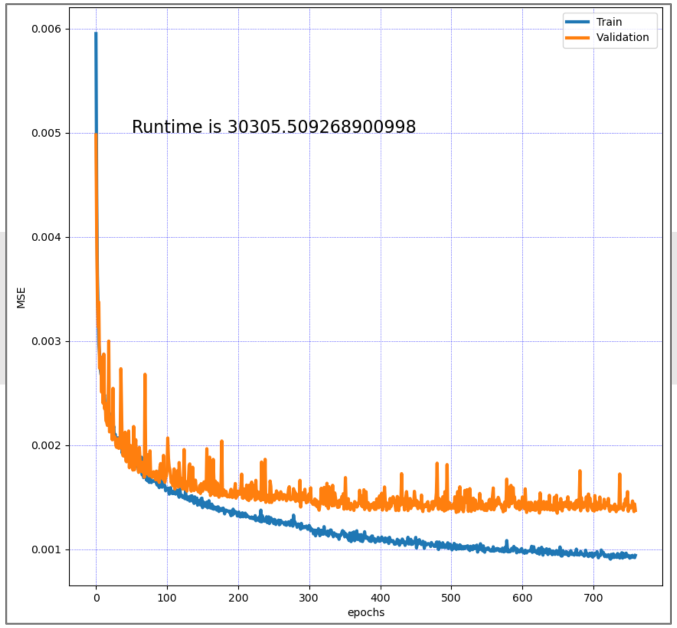
출처: 비전공자가 알려주는 자연어처리 기초 특강 – 12강. 순전파, 역전파, wikidocs, 2023.

AI 모델 학습 프로세스



데이터 학습 시작
(가중치
업데이트 시작)





학습 종료
(가중치
최적화 완료)

출처:CS 179: Lecture 13 – Intro to machine learning, 2017.
출처: Thanapol, P., Lavangnananda, K., Leprévost, F., Glad, A., Schleich, J., & Bouvry, P. (2024). Round-based mechanism and job packing with model-similarity-based policy for Scheduling DL Training in GPU Cluster. Applied Sciences, 14(6), 2349.

Thank you

Yunseon Byun 변윤선, Ph.D

나노융합본부, 이차전지장비연구실

yun-seon@kimm.re.kr