EURON 중급세션 1주차
김유경

신경망과 딥러닝

1. 딥러닝 소개

2. 신경망과 로지스틱 회귀

1. Intro

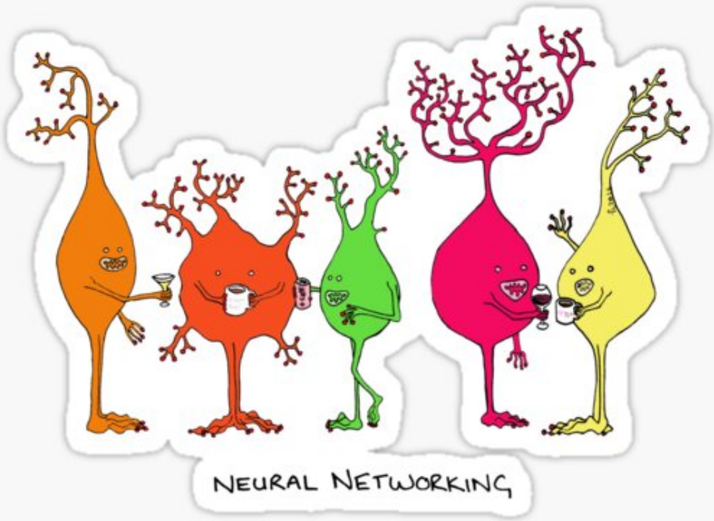
"AI is the new Electricity"

→ 인공지능은 전기처럼 앞으로 수많은 산업들을 크게 변화시킬 수 있다.

딥러닝(Deep Learning)은 인공지능의 한 부분으로, 오늘날 매우 중요해졌다.

딥러닝이란? **신경망**을 학습시키는 것!

그럼 **신경망**이란?



2. 신경망(Neural Network)이란?

Housing Price Prediction Example

6개의 house 샘플이 있고, 각 house의 size(feature)와, price(target)를 알고 있다고 하자.

✓ **선형 회귀(Linear Regression)**: 예측을 위한 어떤 직선(선형 방정식)을 학습

price는 음수가 될 수 없으니 특정 지점보다 작아지면 0으로 유지되도록 한다.

→ ReLU(Rectified Linear Unit) function

- Rectified: 0과 결괏값 중 큰 값을 취하는 것

2. 신경망(Neural Network)이란?

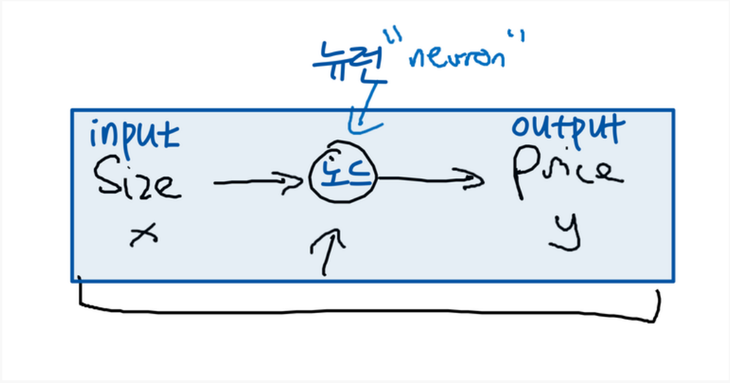
Housing Price Prediction Example

이번에는 각 house의 feature가 4개라고 하자. - size, #bedrooms, postal code, wealth

✓ 뉴런(Neuron)

뉴런은 예측 함수를 계산하여,

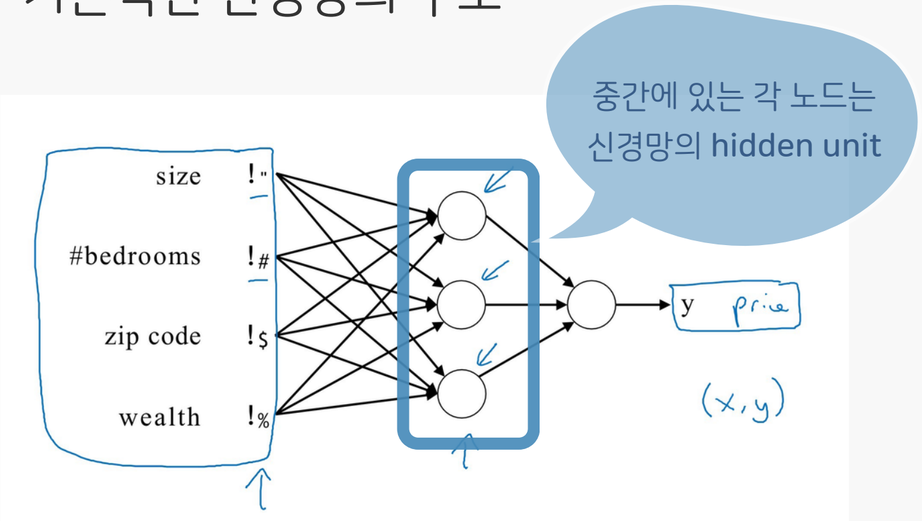
input x에 따른 output y를(예측 값) 반환한다.



✓ 신경망(Neural Network, NN)

신경망은 뉴런이 많이 쌓인 형태이다.

-기본적인 신경망의 구조-



→ 충분한 train data(x, y)를 주면 신경망은 x→y로 연결하는 함수를 스스로 알아내고 학습한다.

3. 지도학습 + 신경망

지도학습은 신경망을 학습시키는 방식이 가장 유용하고 효과적이다.

표준 신경망(Standard NN)

- house features로 price 예측 → 부동산
- 사용자 정보+광고로 사용자의 광고 클릭 예측 → 온라인 광고

합성곱 신경망(Convolutional NN, CNN)

이미지 데이터에는 CNN 사용

- 이미지가 어떤 종류인지 예측 → photo tagging

순환 신경망(Recurrent NN, RNN)

음성이나 언어 같은 시퀀스 데이터에는 RNN 사용

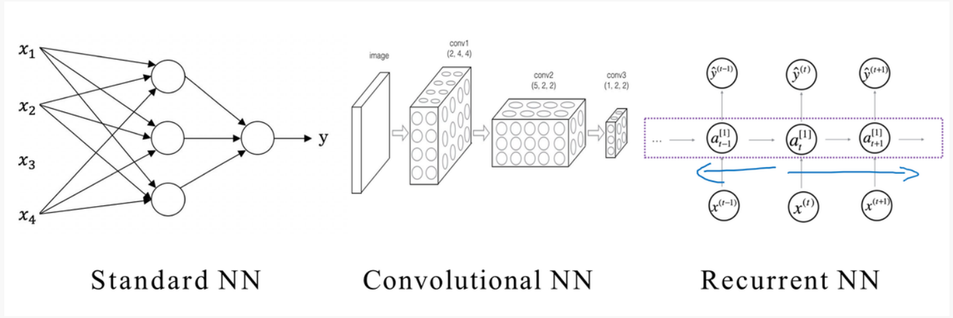
✓ 시퀀스 데이터(Sequence data): 데이터 간의 순서(의존성)이 있는 데이터

✓ 음성은 시퀀스 데이터 중에서도 시간의 흐름에 따라 재생되는 1차원 시계열 데이터(Time Series data)

- 음성으로 Text Transcript를 예측 → 음성 인식
- 한 언어를 다른 언어로 예측 → 번역

Custom/Hybrid 신경망

- 이미지+레이더 정보로 다른 차의 위치 예측 → 자율주행



3. 지도학습 + 신경망

비정형 데이터

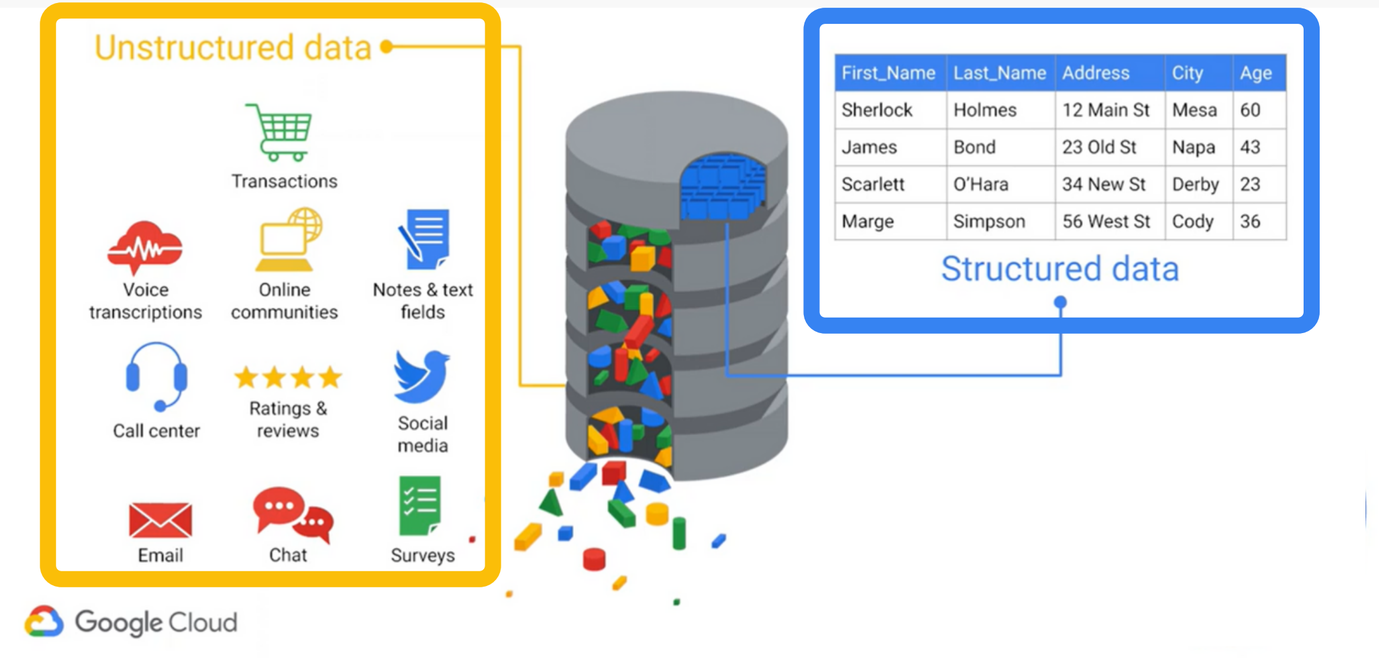
데이터베이스로 표현하기 어려운

텍스트, 이미지, 오디오 등의 데이터

정형 데이터

어떤 구조로 이루어져 csv나

데이터베이스에 저장하기 쉬운 데이터



→ 딥러닝의 발전으로 컴퓨터가 비정형 데이터를 더 잘 해석할 수 있게 되었다.

4. 딥러닝의 성장 동력

data, computation, algorithms의 “**scale**”이 딥러닝을 성장하게 한다.

(1) data

train data의 양이 적을 때: ML 알고리즘과 신경망의 성능 차이가 크지 않다. → 구현 방법에 따라 성능이 달라진다.

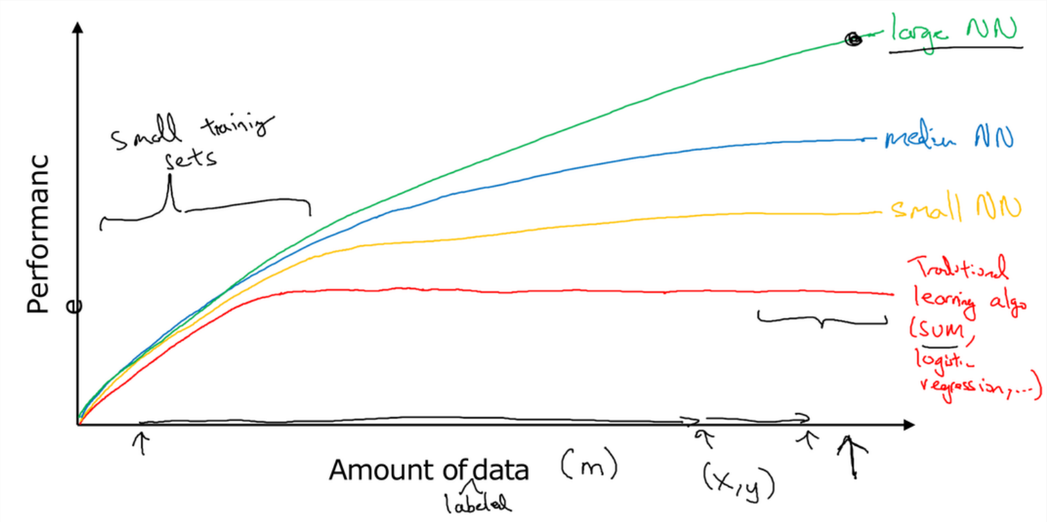
train data의 양이 커질 수록: large NN의 성능이 높아진다.

(2) computation

GPU 같이 전문화된 하드웨어나 더 빠른 네트워크 → 빠른 속도

(3) algorithms

알고리즘 자체의 혁신 → 빠른 속도



신경망을 학습시키는 과정은 반복적이므로 계산 속도가 빠를수록 좋다.

오늘날...

많은 양의 data → 신경망의 성능이 높아진다.
computation, algorithms의 발전 → 학습 속도가 빨라진다.
→ 더 많이 테스트해 보고, 더 좋은 방법을 채택할 수 있다.

⇒ 지금까지도 딥러닝은 계속 발전하고 있다!

※ Notation

m : 전체 dataset의 샘플 개수

- m_{train}: train data의 샘플 개수
- m_{test}: test data의 샘플 개수

n = n_x : input feature 값의 개수(차원)

n_y: output 값의 개수

$x^{(i)}$: i번째 샘플의 x 값

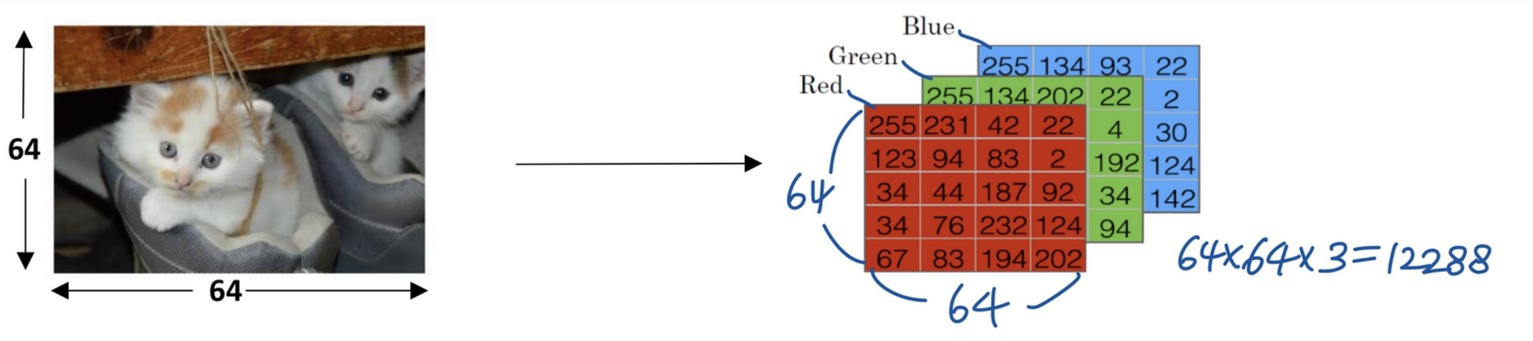
$y^{(i)}$: i번째 샘플의 y 값

$\hat{y}$: y의 예측 값

5. 이진 분류

Cat(1) vs Non-Cat(0)

-이미지 데이터-



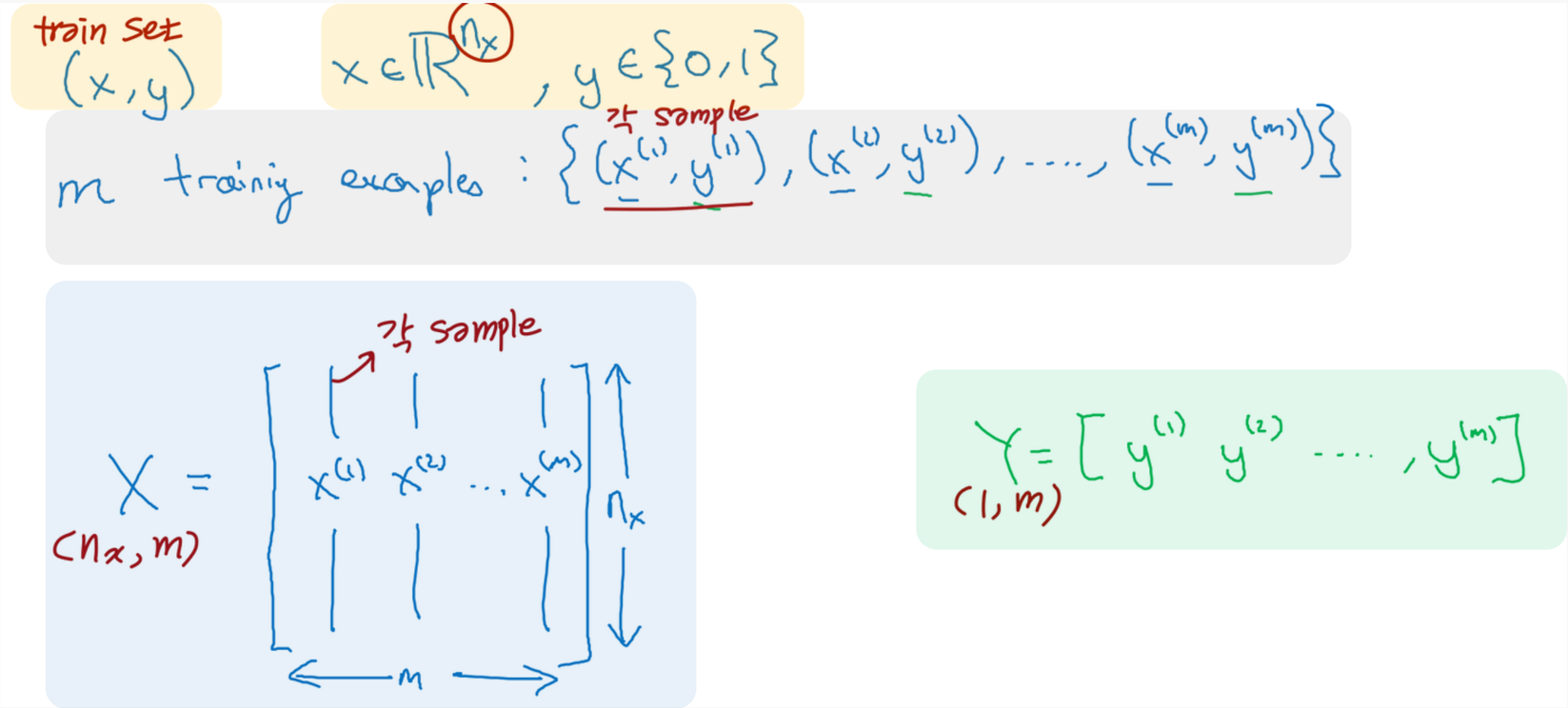
$$x = \begin{bmatrix} 255 \\ 231 \\ 42 \\ \vdots \\ 255 \\ 134 \\ 202 \\ \vdots \\ 255 \\ 134 \\ 93 \\ \vdots \end{bmatrix} \begin{matrix} \text{red } 64\times64 \\ \\ \\ \\ \text{green } 64\times64 \\ \\ \\ \\ \text{blue } 64\times64 \\ \\ \\ \end{matrix}$$

- 이미지 데이터는 픽셀로 이루어진다. 각 픽셀에는 (R, G, B) 값이 포함된다.
  - E.g. 64*64 크기의 이미지에는 총 64*64*3=12288개의 픽셀 값이 있다.

- 이미지 데이터의 픽셀 값을 하나의 열로 나열한다.
- input feature의 차원 = n = 12288

5. 이진 분류

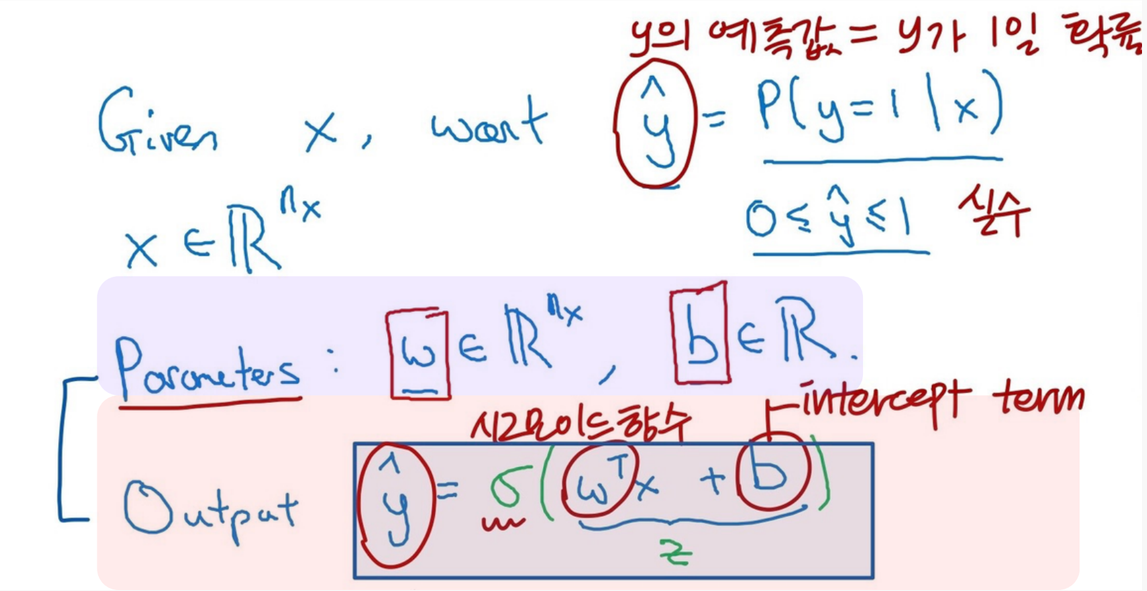
Cat(1) vs Non-Cat(0)



- X: 각 샘플을 열로 나열한 행렬
- Y: 각 샘플의 target 값을 나열한 1차원 배열

6. Logistic Regression

✓ 이진 분류를 위한 **로지스틱 회귀(Logistic Regression)** 알고리즘



y의 예측 값(y hat): y가 1일 확률

파라미터

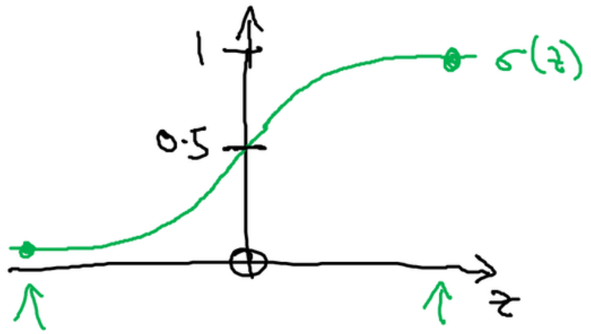
- w: 기울기, weight
- b: 절편, threshold

Logistic Regression의 목표는 파라미터 w, b를 잘 학습시켜 **y의 예측 값(y hat)을 y에 가깝게 예측**하는 것이다.

직선 함수에 **시그모이드 함수**를 적용하여 결과값을 예측한다.

- 직선 함수 결과 값은 0~1 사이로 나오지 않아 이진 분류에 적절하지 않다.

6. Logistic Regression



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

If z large $\sigma(z) \approx \frac{1}{1+0} = 1$

If z large negative number (small)

$\sigma(z) = \frac{1}{1+e^{-z}} \approx \frac{1}{1+\text{Bignum}} \approx 0$

시그모이드 함수는 직선 함수 z(직선 함수에서의 예측 값)이 커질수록 1에 수렴하고, 작아질수록 0에 수렴한다.

→ 예측 **값을 0~1 사이로 만든다.**

6. Logistic Regression

$Given\ x\ ,\ \hat{y} = P(y = 1|x), \text{where } 0 \leq \hat{y} \leq 1$

The input features vector: $x \in \mathbb{R}^{n_x}$, where n_x is the number of features

The training label: $y \in 0,1$

The weights: $w \in \mathbb{R}^{n_x}$, where n_x is the number of features

The threshold: $b \in \mathbb{R}$

The output: $\hat{y} = \sigma(w^T x + b)$

Sigmoid function: s = $\sigma(w^T x + b)$ = $\sigma(z) = \frac{1}{1+e^{-z}}$

❓ 직선 함수의 w^T(transpose of w)는 가중치 w를 x에 곱하기 위해 행으로 바꾸는 것

6. Logistic Regression

파라미터 w,b를 잘 학습시킨다는 건 어떤 의미일까?

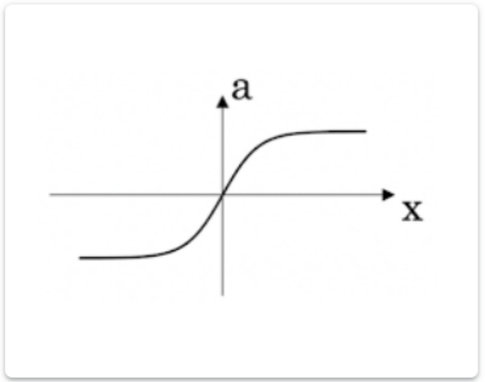
좋은 w,b는 어떻게 찾아 나갈까?

→ Loss function, Cost function

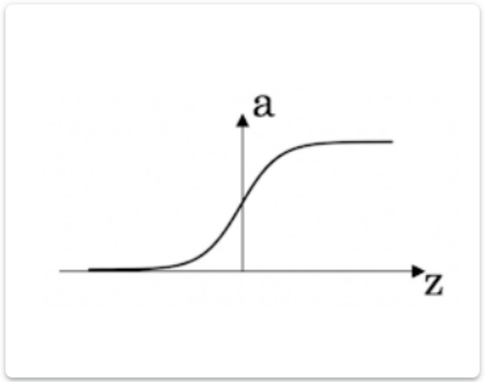
→ Gradient Descent

퀴즈 1.

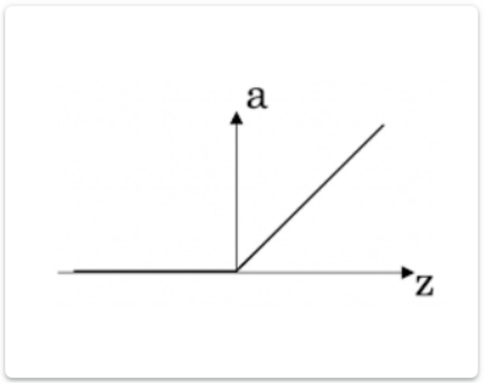
1. 다음 중 ReLU function은 무엇인가요? * 1점



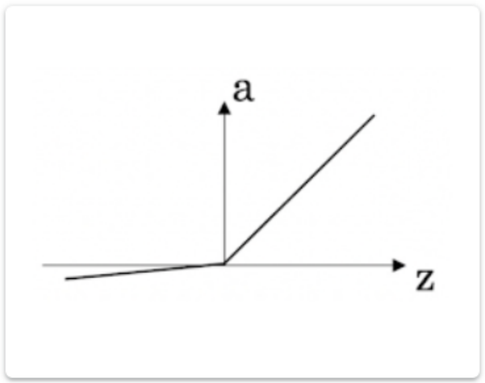
옵션 1



옵션 2



옵션 3



옵션 4

-Remind-

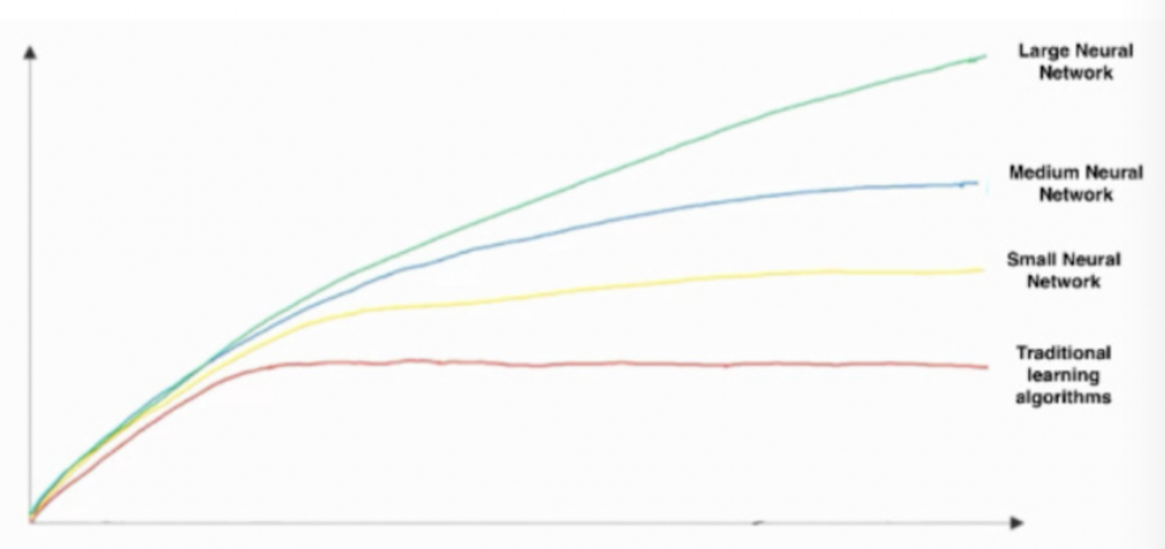
ReLU(Rectified Linear Unit) function

- Rectified: 0과 결괏값 중 큰 값을 취하는 것

퀴즈 7.

7. 강의에서 보았던 아래 그림의 x,y 축은 각각 어떤 것인가요? *

1점



- x: 알고리즘의 성능, y: 데이터의 양
- x: 데이터의 양, y: 모델의 크기
- x: 알고리즘으로의 input, y: output
- x: 데이터의 양, y: 알고리즘의 성능

-Remind-

train data의 양이 적을 때: ML 알고리즘과 신경망의 성능 차이가 크지 않다. → 구현 방법에 따라 성능이 달라진다.

train data의 양이 커질 수록: large NN의 성능이 높아진다.