KUBIG 25-W
겨울방학 BASIC STUDY SESSION

# NLP SESSION
## WEEK1

KUBIG
DATA SCIENCE & AI

# CONTENTS

KUBIG
DATA SCIENCE & AI

Session 중에는 모두 캠을 켜주시기 바랍니다 :)

# 01 NLP SESSION 소개

KUBIG
DATA SCIENCE & AI

# 1-1. 분반장 소개

KUBIG DATA SCIENCE & AI

## 잘 부탁 드립니다 ! !


분반장 19기 이동주

- 데이터분석 → NLP/LLM
- 어쩌다보니 석사생
- 책모임, 연극 & 뮤지컬 토크 환영 🤗

분반장 19기 심승현

- 관심 분야: AI 응용, 시계열 처리
- 디버깅과 아직도 친해지는 중…
- 스터디와 친목 둘 다 환영

(개막하면 직관 모임도…?! ⚾ )

분반장 19기 심승현

# 1-2. 진행방식

KUBIG DATA SCIENCE & AI

| 주차 | 복습과제 | 학습내용 | 예습과제 |
|---|---|---|---|
| 1주차 | - 해당 주차에 배운 내용에 대한 코드 실습 과제 부여<br><br>- ex) week1 복습과제: deep learning reminder(pytorch basic)<br><br>- session 시작 초반부에 우수 코드 선정자가 5분 가량 코드 구현 과정 발표(별도 발표자료 없이 코드를 화면공유하여 발표)<br><br>- 마감기한: 수요일 오후 6시 | OT, DL Reminder | - 다음 주차에 배울 내용에 대한 코드 실습 과제 부여<br><br>- ex) week1 예습과제: 텍스트 전처리 pipeline 구현 코드<br><br>- week1 예습과제라 함은, week1 session이 끝나고 부여되는, week2 내용에 대한 예습과제를 의미합니다!<br><br>- 마감기한: 수요일 오후 6시 |
| 2주차 | | 텍스트 전처리, 워드 임베딩 (Word2Vec, GloVe) | |
| 3주차 | | 순환신경망: RNN, LSTM, GRU, ELMo | |
| 4주차 | | Attention, Transformer | |
| 5주차 | | BERT/GPT | |
| 6주차 | | LLM 기초: Fine-tuning, RAG | |
| 7주차 | | Toy project | |

# 1-2. 진행방식

KUBIG
DATA SCIENCE & AI

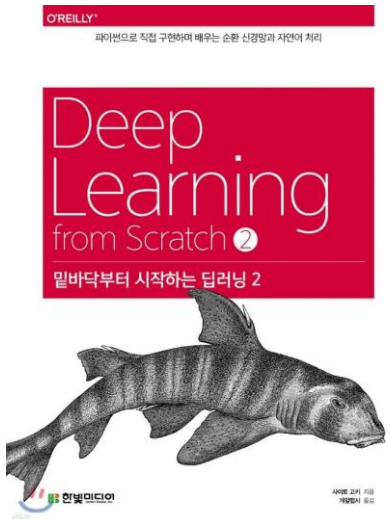
매주 목 19:00~21:00 총 2시간 진행


* 모든 문의(과제, 강의, 출결 등)는 분반장 이동주/심승현에게 슬랙 디엠 부탁드립니다

# 1-3. 타임라인

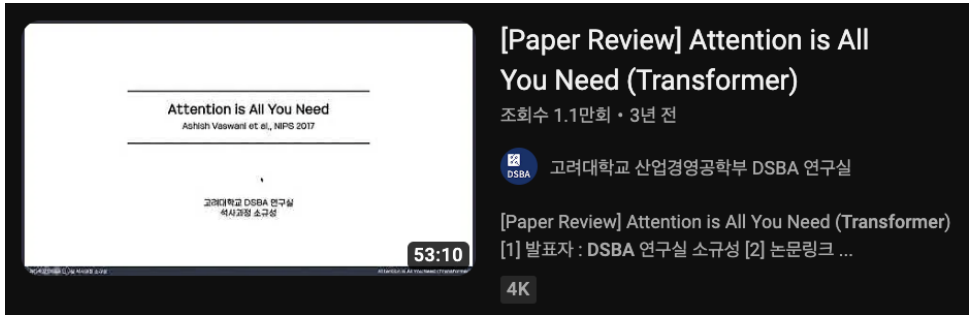

1월 9일(목)
WEEK1
1월 16일(목)
WEEK2
팀빌딩 예정
2월 20일(목)
WEEK7
팀별 중간발표 예정
(팀당 3분 내외)
2월 27일(목)
KUBIG CONTEST

# 1-4. 참고 교재




밑바닥부터 시작하는 딥러닝2


딥러닝을 이용한 자연어 처리 입문
위키독스에서 무료 이용 가능(부분 유료)




고려대학교 DSBA 연구실
유튜브 채널에서 paper review 영상 참고

# 1-5. 자기 소개

KUBIG DATA SCIENCE & AI

방학동안 함께 하실 분들 !

기광민

김재훈

이예지

강서연

김민재

김정찬

이우진

남동연

이예일

김유진

이연호

장건호

이영서

여러분을 소개해주세요!

---

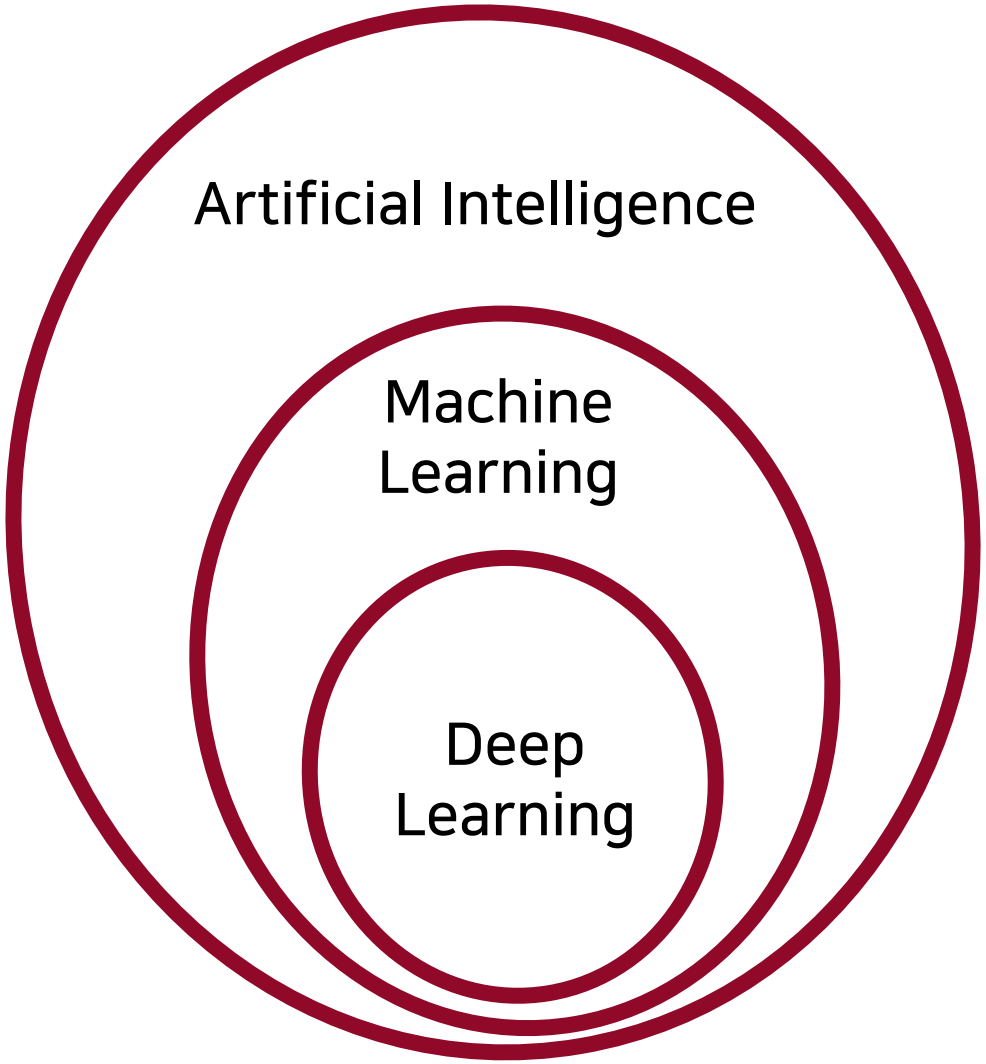
기수, 이름, 학과, 나이, 취미

NLP 경험, NLP 선택 이유, 원하는 방향성

# 02 Theoretical Background of Deep Learning

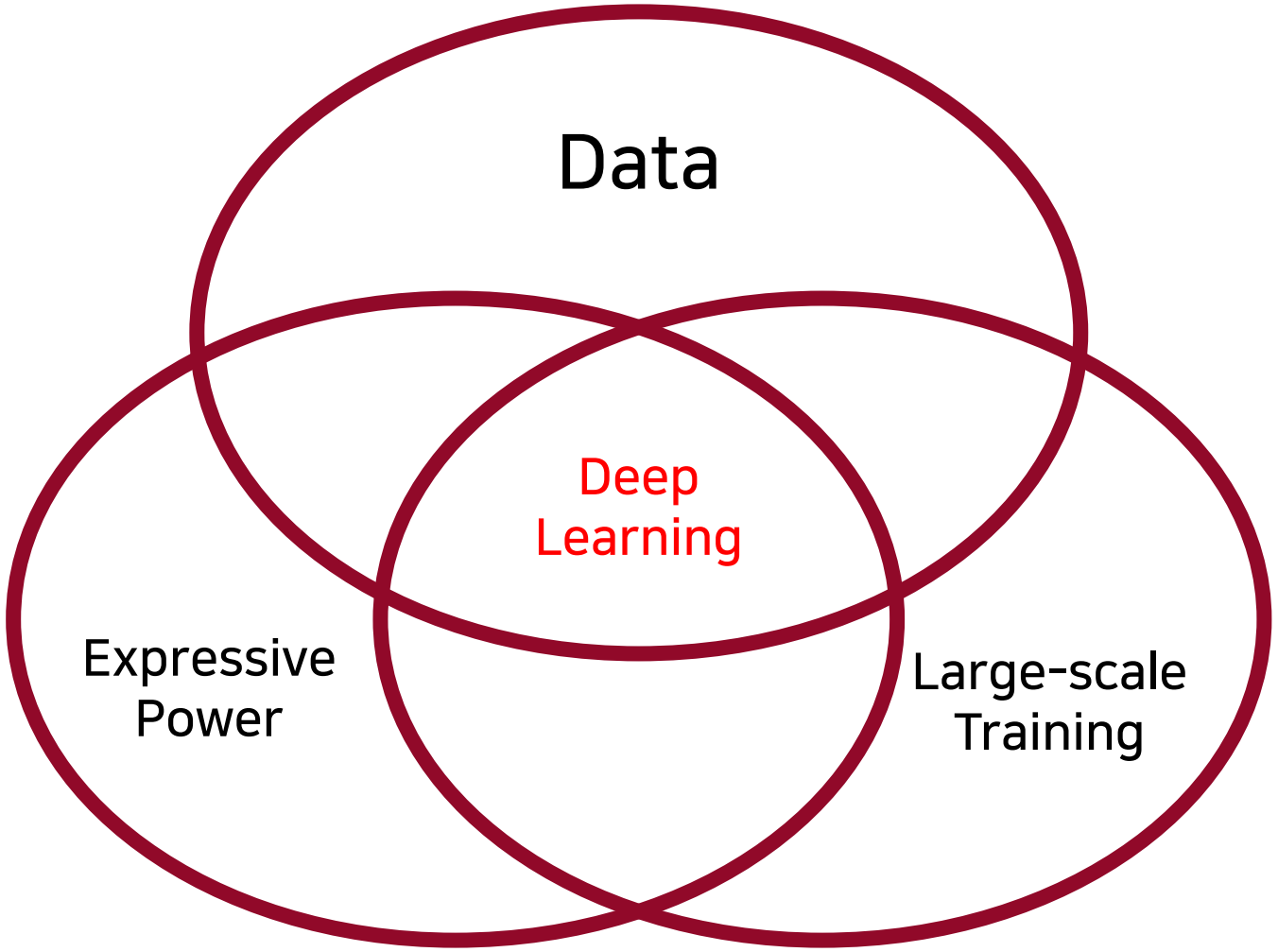
KUBIG
DATA SCIENCE & AI

# 2-0. What is Deep Learning?




출처: 딥러닝을 위한 통계적 모델링(STAT433)

# 2-0. What is Deep Learning?




출처: 딥러닝을 위한 통계적 모델링(STAT433)

# 2-0. What is Deep Learning?

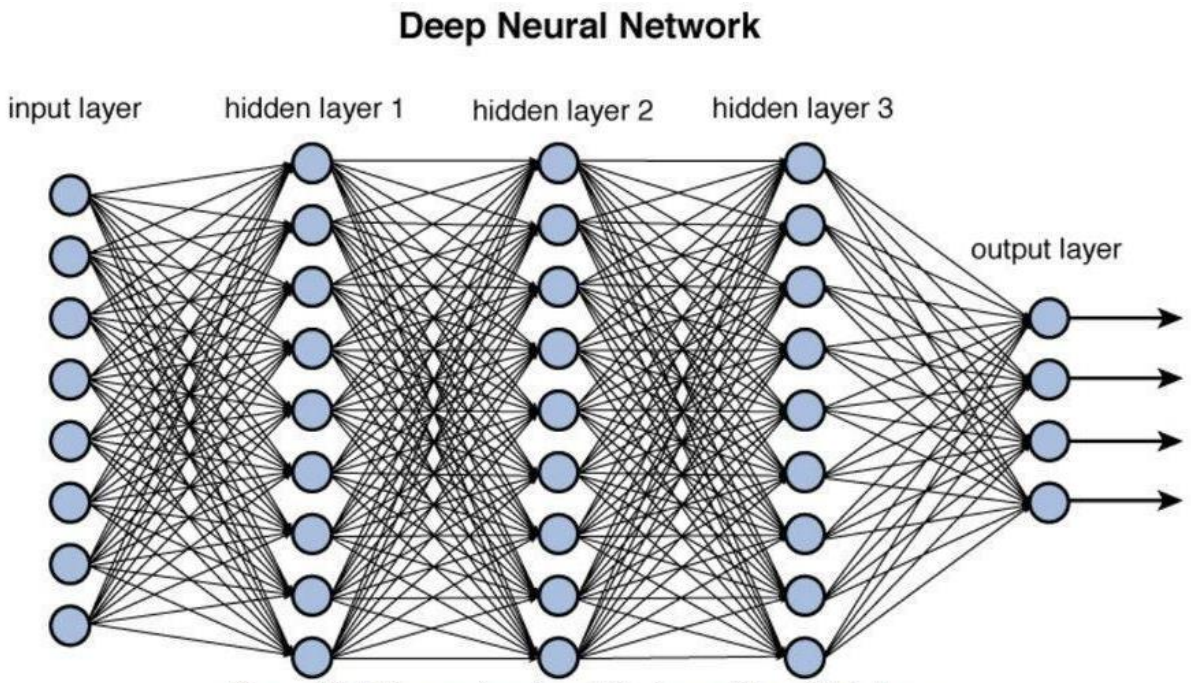


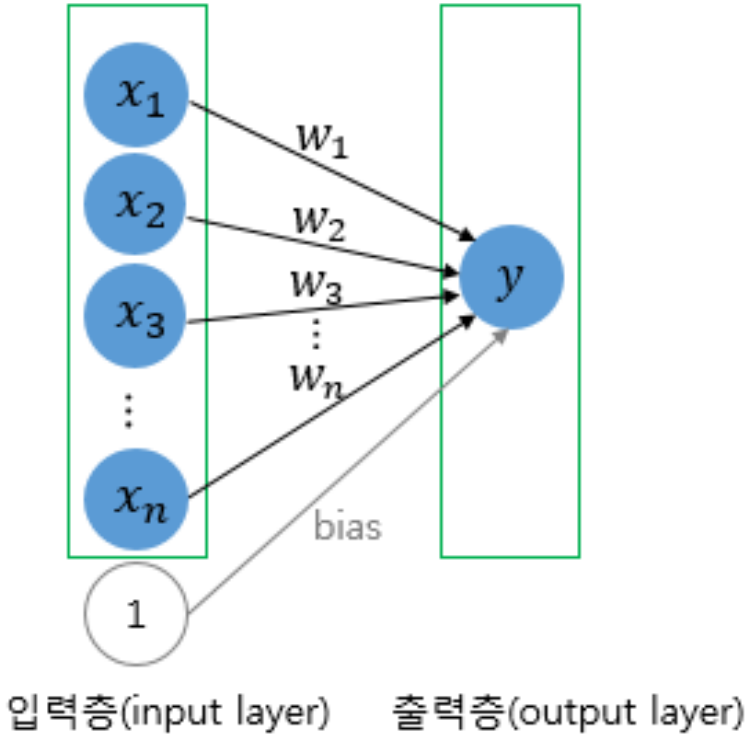

Figure 12.2 Deep network architecture with multiple layers.

Deep Learning = Deep Neural Network

# 2-1. Single-Layer Perceptron

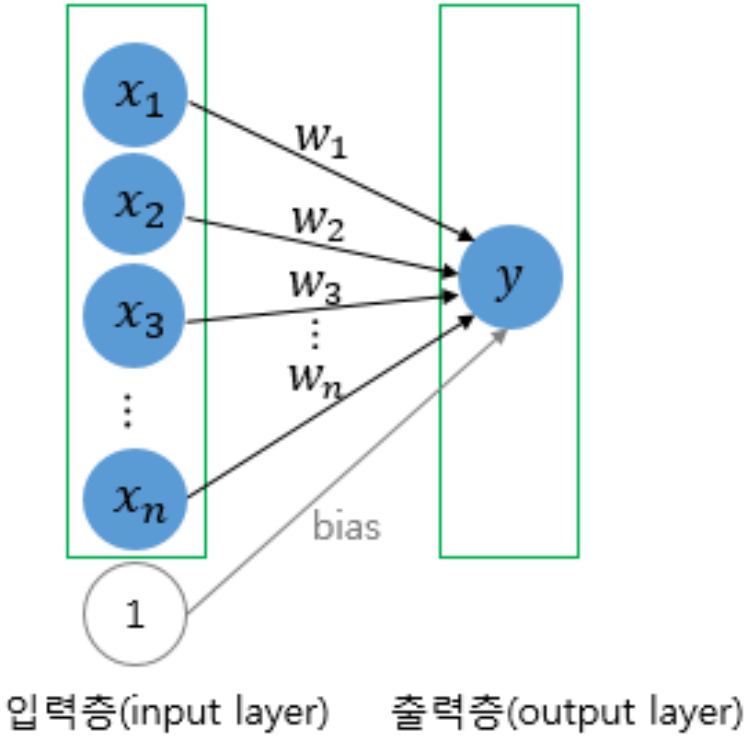
KUBIG DATA SCIENCE & AI

single-layer perceptron




출처: https://wikidocs.net/24958

# 2-1. Single-Layer Perceptron




When activation function is step function..

$$if \sum_{i}^{n} w_i x_i + b \geq 0 \rightarrow y = 1$$

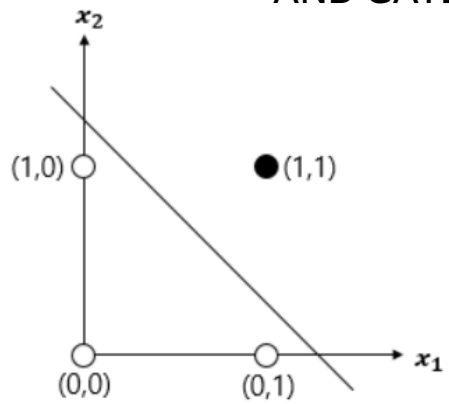
$$if \sum_{i}^{n} w_i x_i + b < 0 \rightarrow y = 0$$

출처: https://wikidocs.net/24958

# 2-2. The XOR Gate Problem

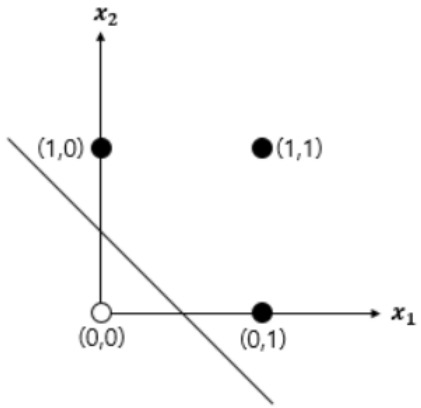
Separate B/W by a single line *(in 2-dimension space)*

AND GATE



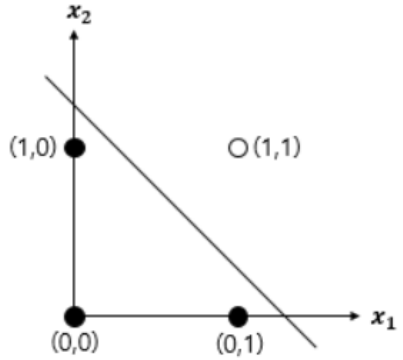

| $x_1$ | $x_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

OR GATE




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

NAND GATE



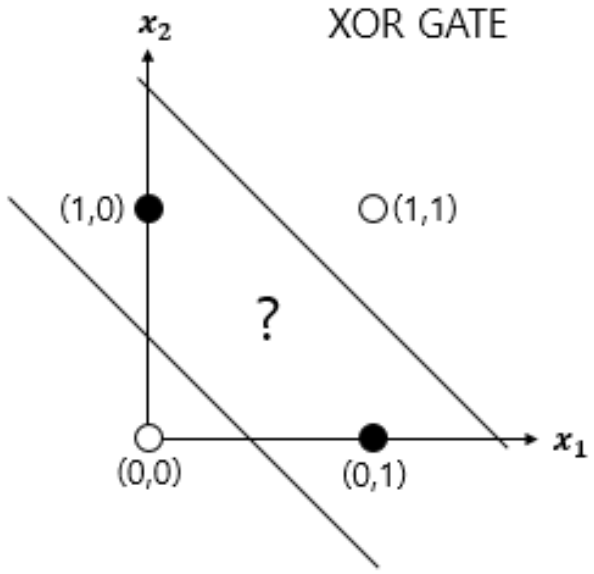

| $x_1$ | $x_2$ | $y$ |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

출처: https://wikidocs.net/24958

KUBIG DATA SCIENCE & AI

# XOR problem

Can you separate B/W by a single line *(in 2-dimension space)*?



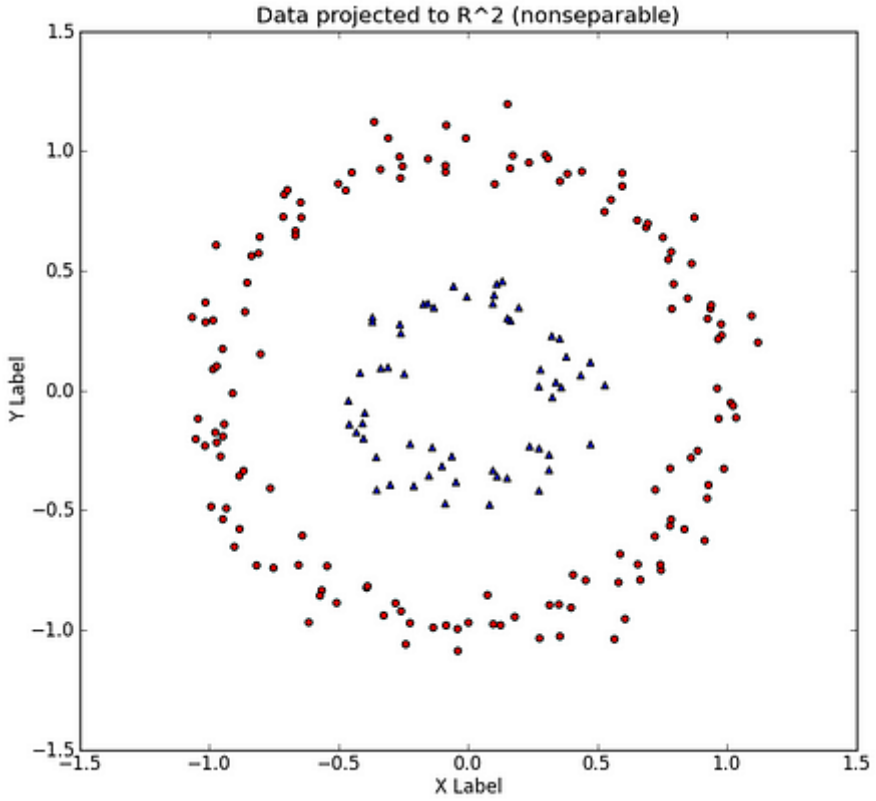

| $x_1$ | $x_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

출처: https://wikidocs.net/24958

KUBIG DATA SCIENCE & AI

# Linearly Non-Separable

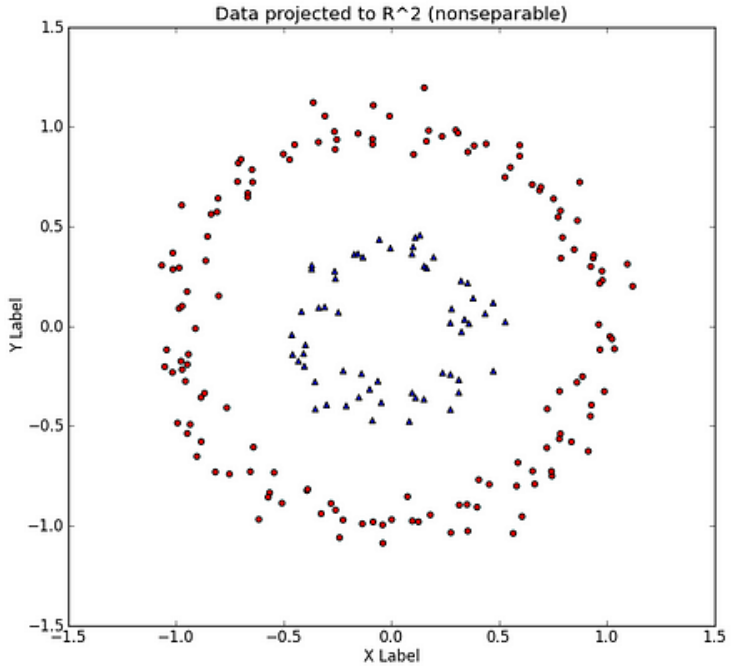
*(in 2-dimension space)*



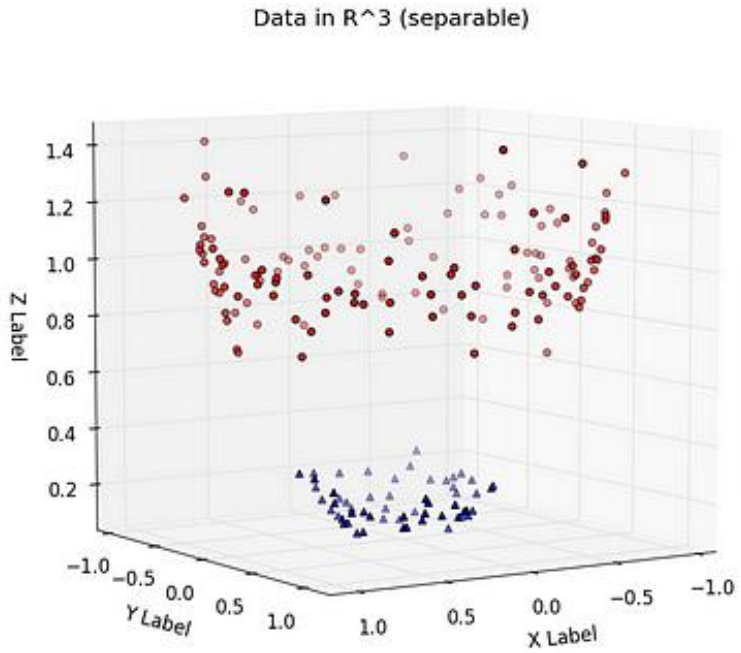

출처: https://vivek-yadav.medium.com/

# Linearly Non-Separable

*(in 2-dimension space)*

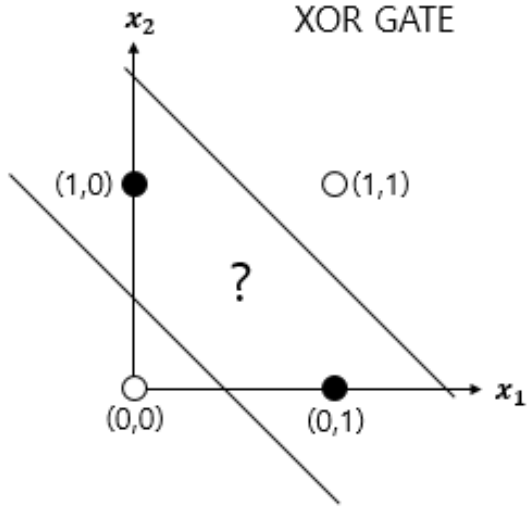



High-dimensional Space



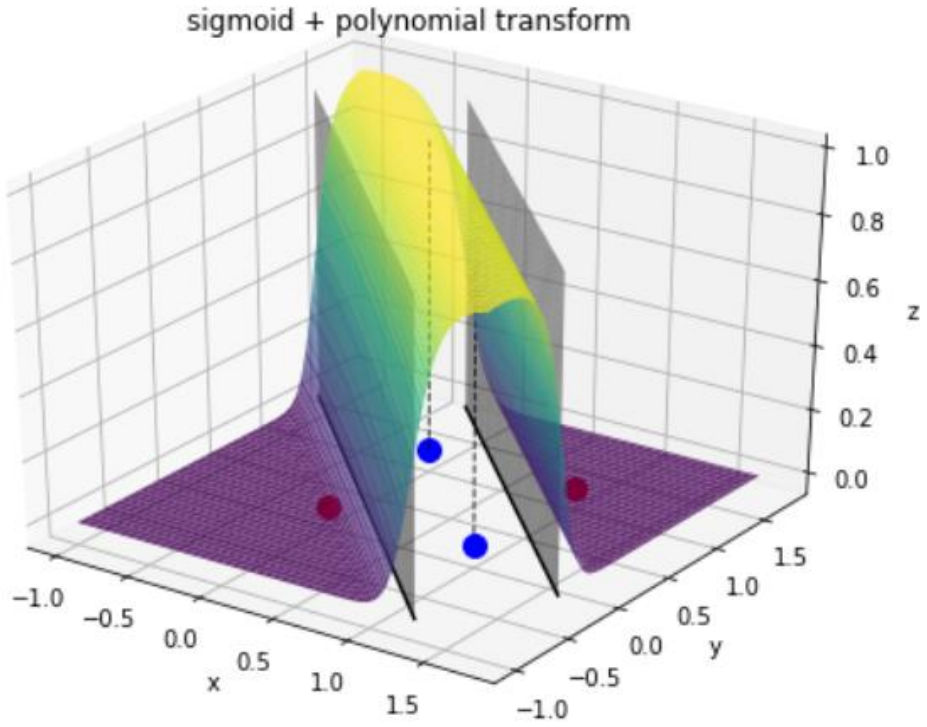

출처: https://vivek-yadav.medium.com/

# 2-3. Dimensional transform




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

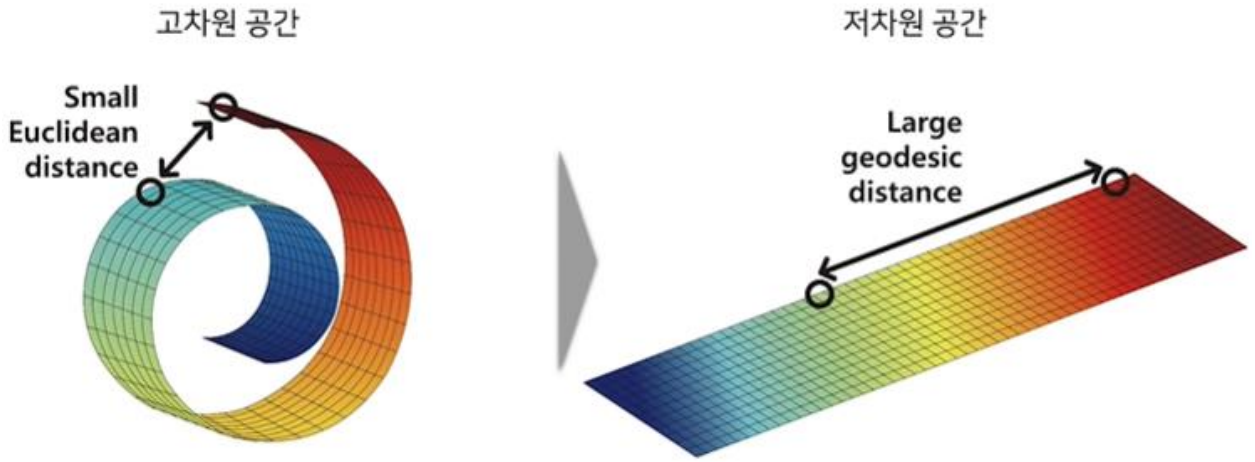
High-dimensional Space




출처: https://datascience.stackexchange.com/questions/11589/creating-neural-net-for-xor-function

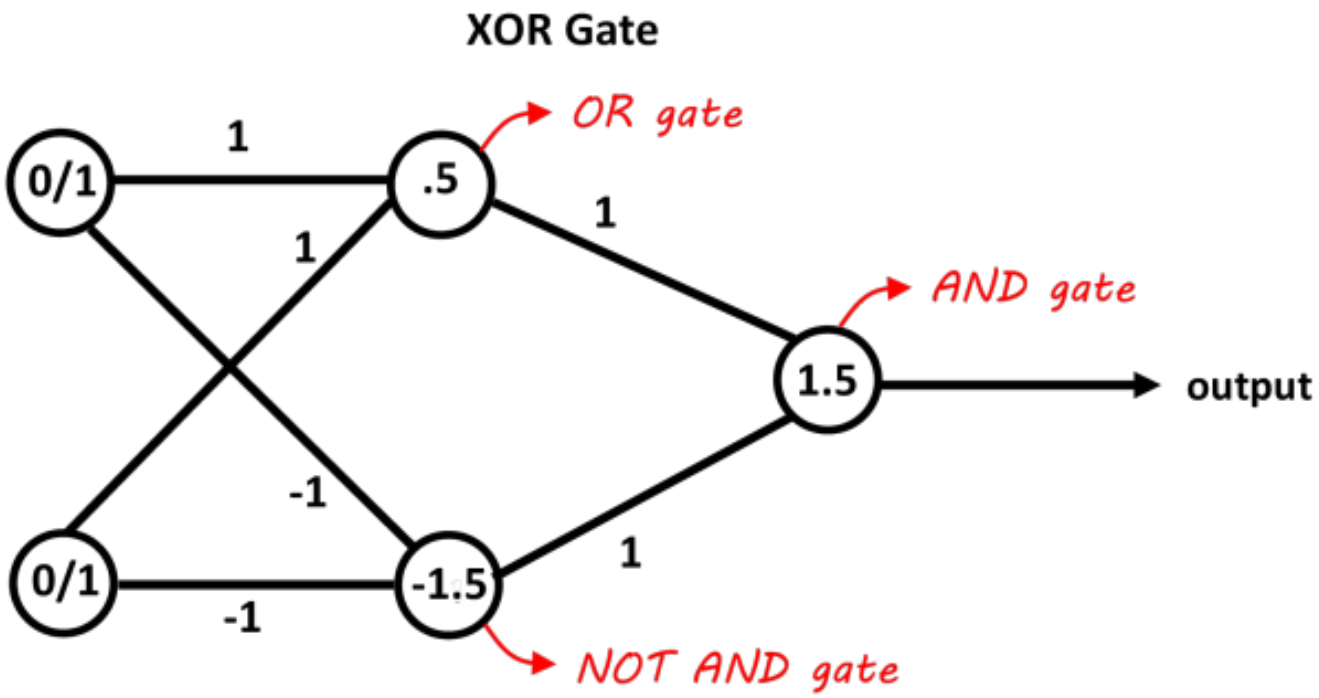
KUBIG DATA SCIENCE & AI

# Manifold Hypothesis




- Hypothesis: High-dimensional data tend to lie in the vicinity of a low-dimensional manifold
- We can map data to a high-dimensional space through a smooth representation if the manifold hypothesis holds

출처: https://velog.io/@sheoyonj/NN-Network-3

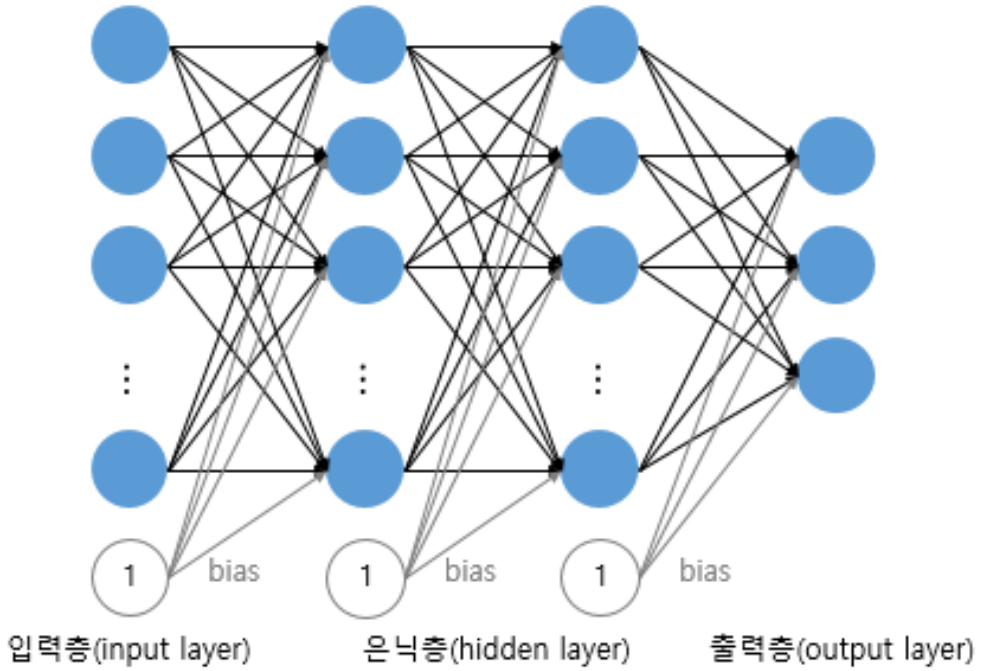



출처: https://datascience.stackexchange.com/questions/11589/creating-neural-net-for-xor-function

KUBIG DATA SCIENCE & AI

# Multi-Layer Perceptron

(More Layers, much more parameters)




출처: https://wikidocs.net/24958

KUBIG DATA SCIENCE & AI

# Universal Approximation Theorem

## Why deep learning is powerful

**Universal approximation theorem** — Let $C(X, \mathbb{R}^m)$ denote the set of continuous functions from a subset $X$ of a Euclidean $\mathbb{R}^n$ space to a Euclidean space $\mathbb{R}^m$. Let $\sigma \in C(\mathbb{R}, \mathbb{R})$. Note that $(\sigma \circ x)_i = \sigma(x_i)$, so $\sigma \circ x$ denotes $\sigma$ applied to each component of $x$.

Then $\sigma$ is not polynomial if and only if for every $n \in \mathbb{N}$, $m \in \mathbb{N}$, compact $K \subseteq \mathbb{R}^n$, $f \in C(K, \mathbb{R}^m), \varepsilon > 0$ there exist $k \in \mathbb{N}$, $A \in \mathbb{R}^{k \times n}$, $b \in \mathbb{R}^k$, $C \in \mathbb{R}^{m \times k}$ such that

$$\sup_{x \in K} \|f(x) - g(x)\| < \varepsilon$$

where $g(x) = C \cdot (\sigma \circ (A \cdot x + b))$

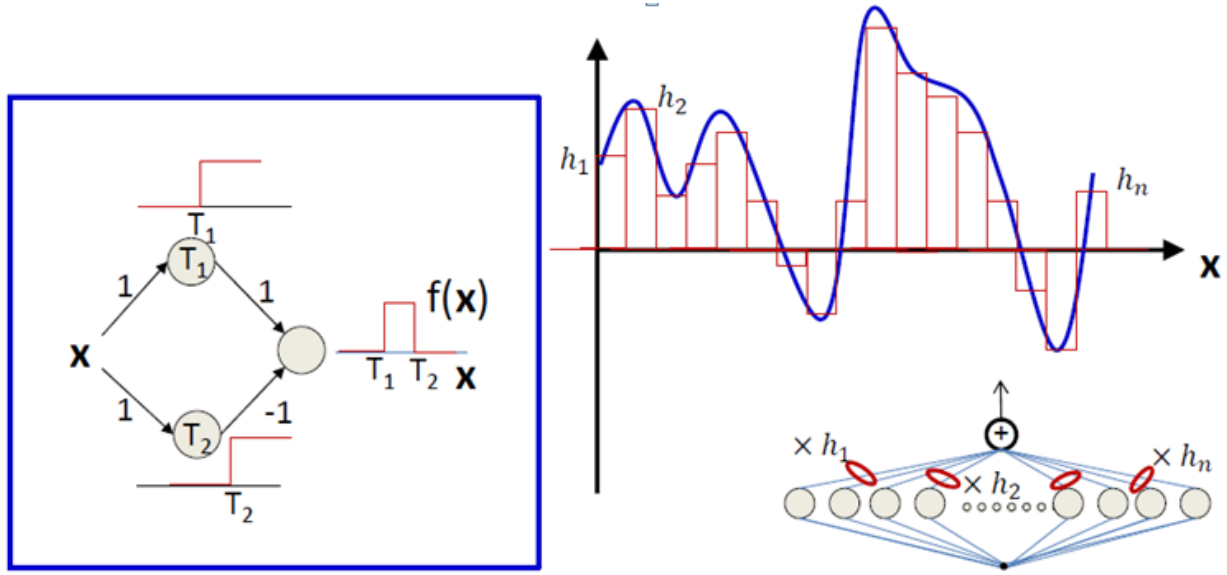
(Universal Approximation Theorem) For a given arbitrary continuous function on a bounded domain and an error bound, there always exists a one-hidden-layer neural network that can approximate the given continuous function within the error bound.

출처: https://en.wikipedia.org/wiki/Universal_ approximation_ theorem

KUBIG DATA SCIENCE & AI

# Universal Approximation Theorem

Why deep learning is powerful



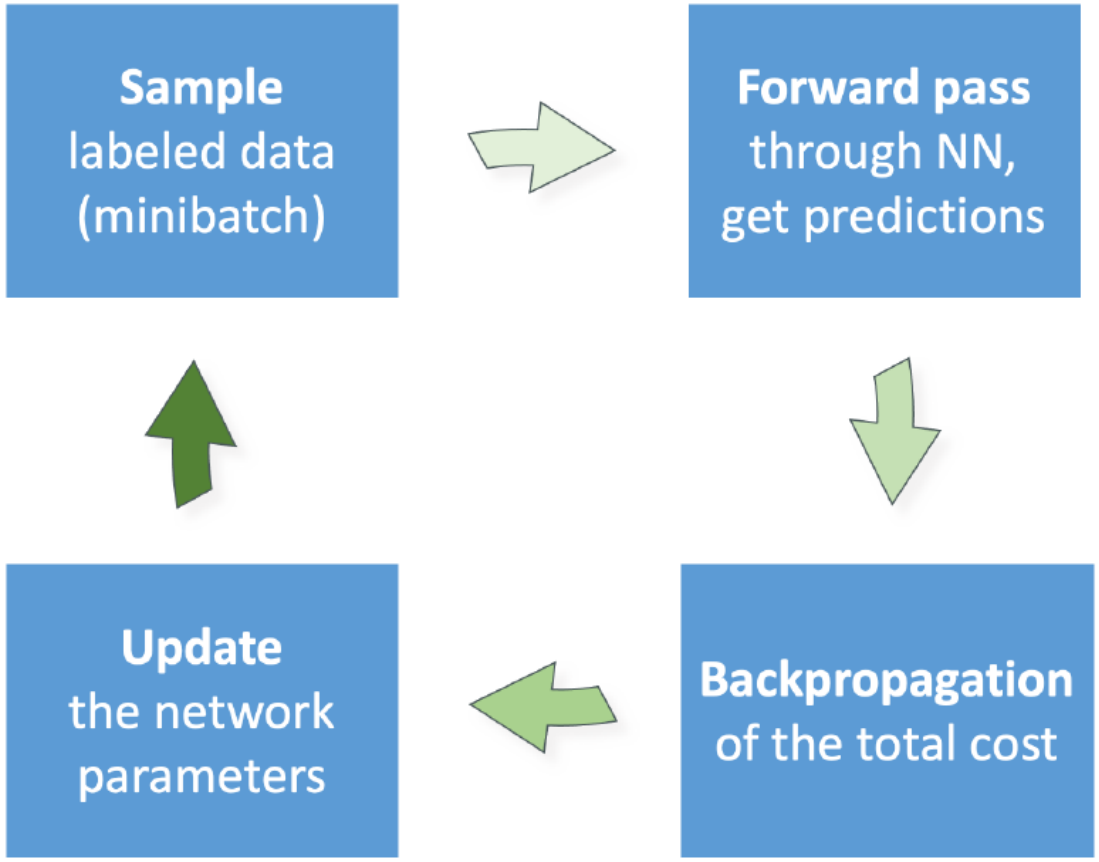

Increase in # of hidden units -> increase in # of parameters
-> increase in # of regions -> increase in # of patterns a function can represent

출처: https://medium.com/analytics-vidhya/

# 03 Factors of Deep Learning

KUBIG
DATA SCIENCE & AI

# 3-1. Deep Learning Training Cycle

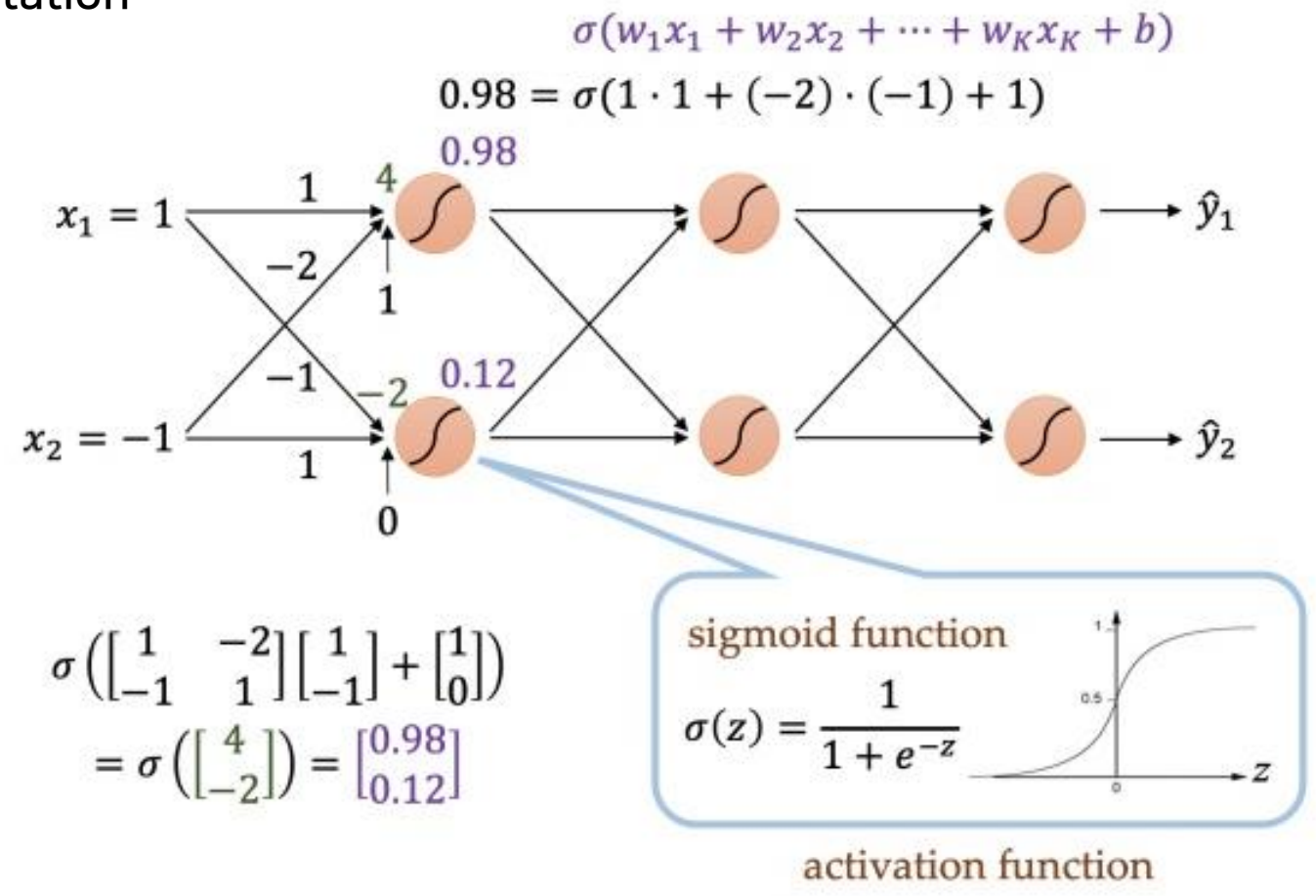



출처: 딥러닝과수학 Seungsang Oh

# 3-2. Forward Pass

KUBIG DATA SCIENCE & AI

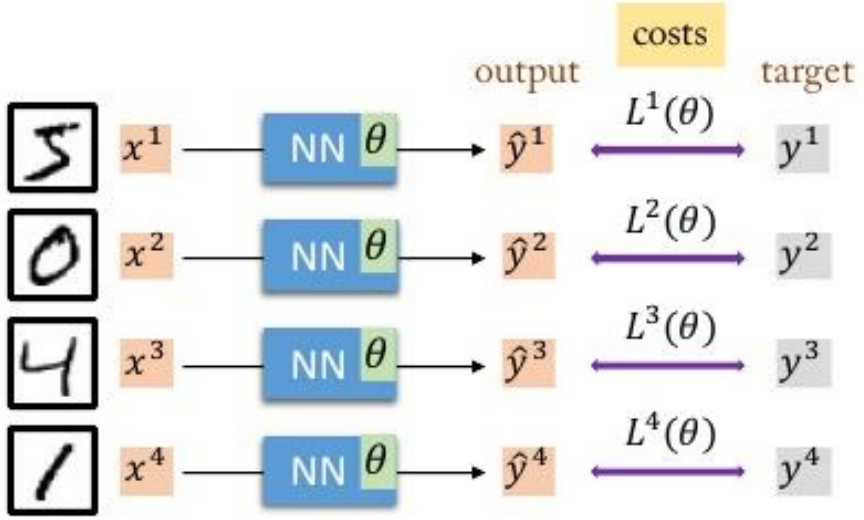
## Forward Pass computation




출처: 딥러닝과수학 Seungsang Oh

KUBIG DATA SCIENCE & AI

## Total Cost



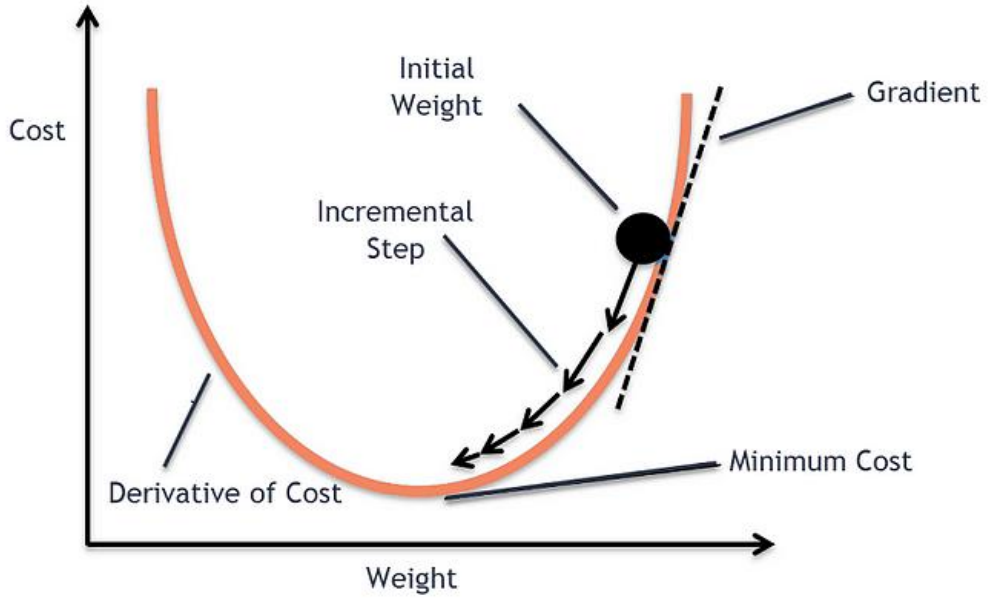

**Total Cost**

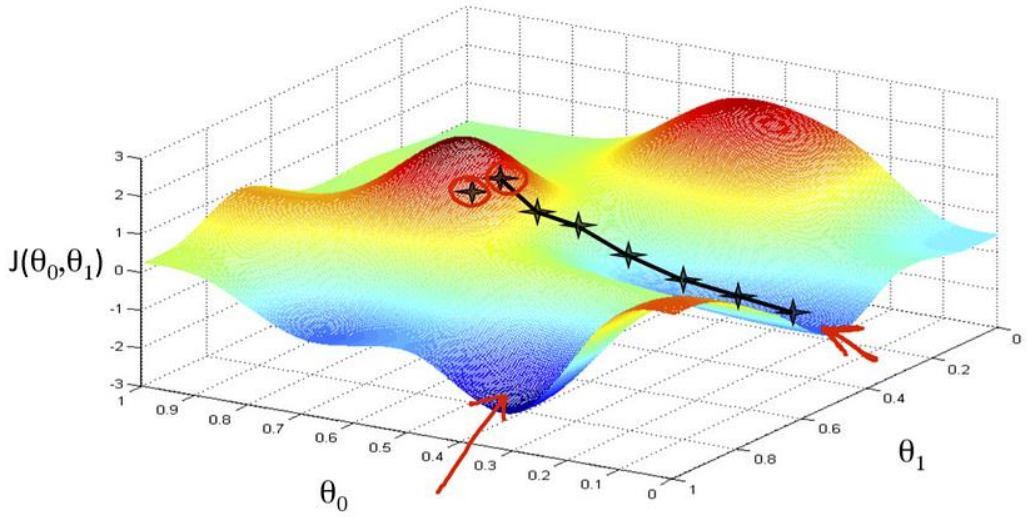
$$C(\theta) = \sum_{n=1}^{N} L^n(\theta)$$

Total cost를 minimize하는 network parameter 찾기

출처: 딥러닝과수학 Seungsang Oh

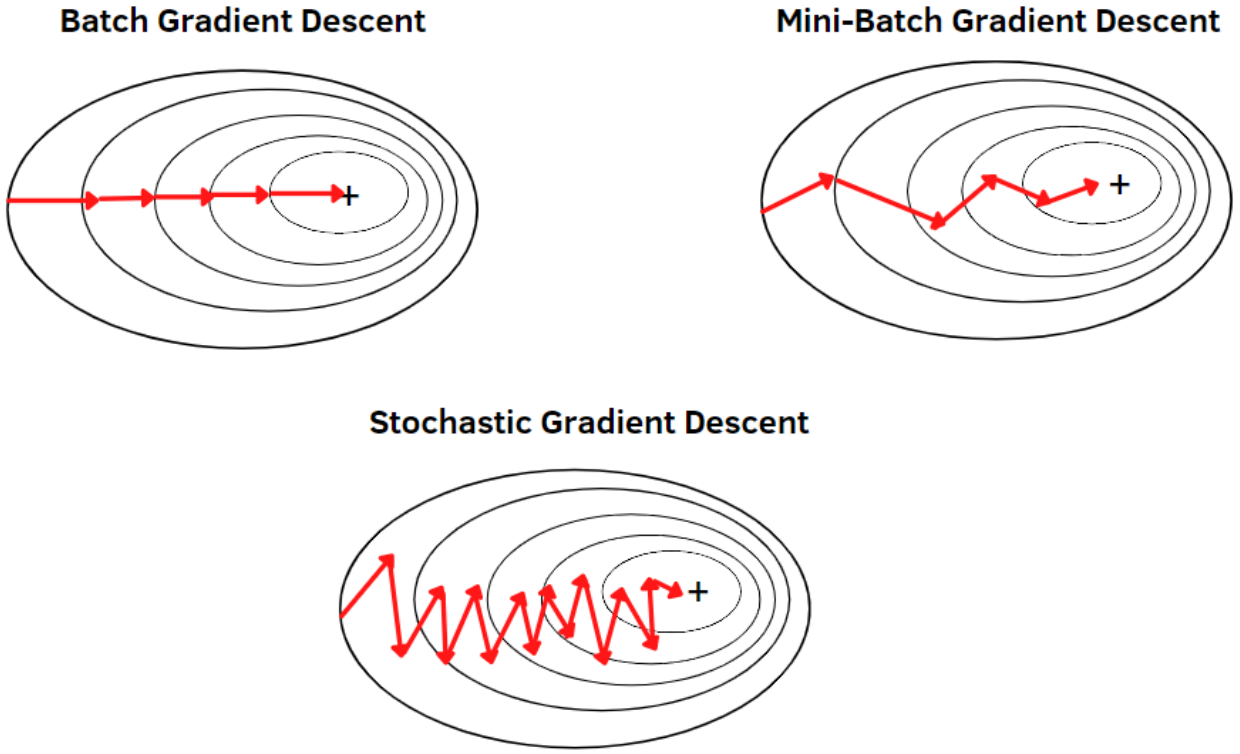
# 3-4. Gradient Descent




출처: https://towardsdatascience.com/an-intuitive-explanation-of-gradient-descent-83adf68c9c33
https://www.analyticsvidhya.com/blog/2022/07/gradient-descent-and-its-types/

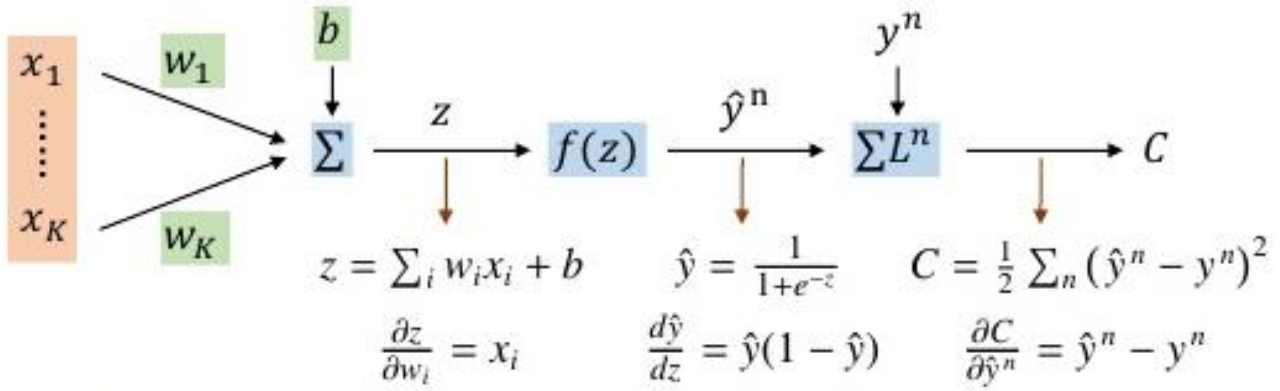
# 3-4. Gradient Descent




출처: https://towardsdatascience.com/an-intuitive-explanation-of-gradient-descent-83adf68c9c33
https://www.analyticsvidhya.com/blog/2022/07/gradient-descent-and-its-types/

# 3-5. Backpropagation

$$w_i(t+1) = w_i(t) - \eta \frac{\partial C}{\partial w_i}$$




$$\frac{\partial C}{\partial w_i} = \sum_{n=1}^{N} \frac{\partial z^n}{\partial w_i} \frac{d\hat{y}^n}{dz^n} \frac{\partial C}{\partial \hat{y}^n} = \sum_{n=1}^{N} x_i^n \hat{y}^n (1-\hat{y}^n)(\hat{y}^n - y^n)$$

Chain rule

출처: 딥러닝과수학 Seungsang Oh

# 04 과제 및 팀 빌딩 안내

KUBIG
DATA SCIENCE & AI

# 4-1. 예습과제, 복습과제

KUBIG
DATA SCIENCE & AI

| 1주차 복습과제1 | 1주차 복습과제2 | 1주차 예습과제1 | 1주차 예습과제2 |
| --- | --- | --- | --- |
| Pytorch 기본 익히기 code | 내용 복습 / 추론 | 텍스트 전처리 + 워드클라우드 code | Word2vec code |

코드과제의 파일형식은 ipynb로, KUBIG 25-1 Github repo에 업로드 될 예정입니다!
Colab 환경에서 제작된 과제들이므로 google colab에서 실행하시는 것을 권장드립니다.

## 📢 KUBIG Contest Team Build

20기 기광민 김민재 김유진 김재훈 김정찬 이연호 이예지 이우진 장건호
21기 강서연 남동연 이영서 이예일

금일 세션이 종료된 후, 관심 분야 투표 공지 예정.
구글 폼으로 수요 조사 후 관심분야에 따라 팀 분할.

WEEK2 세션 시간에 팀 빌딩 결과 공지 예정.

수고하셨습니다 !