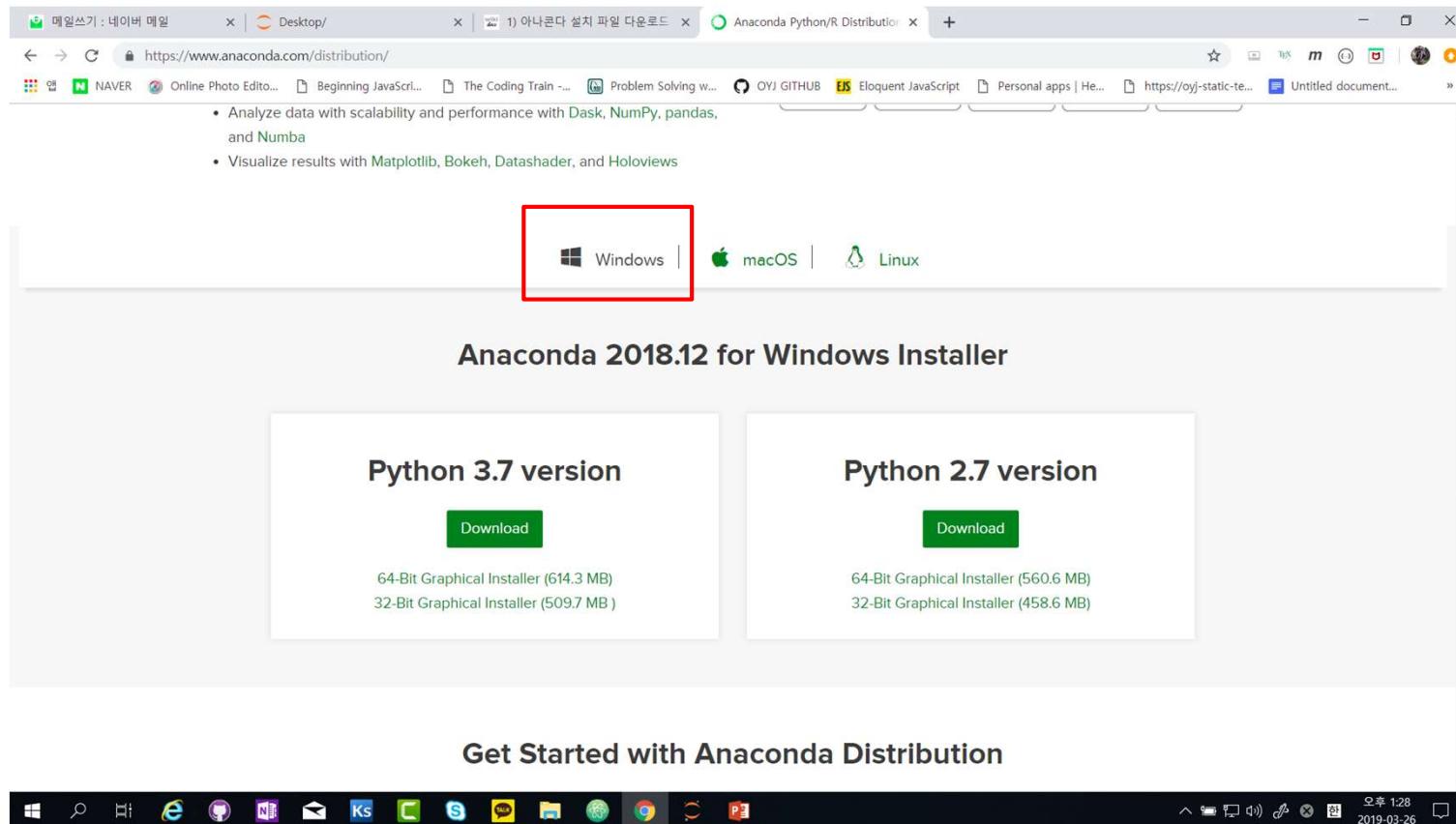


# Python 을 활용한 머신러닝 입문

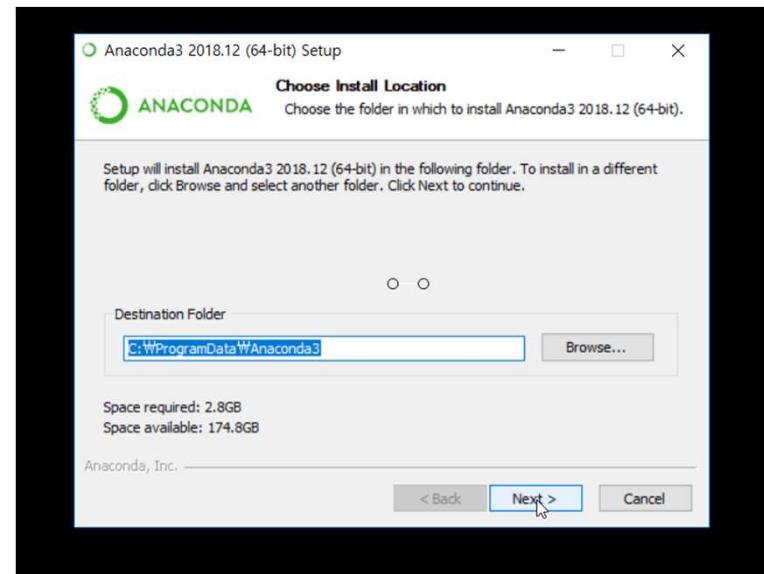
2019. 6

# Anaconda Installation

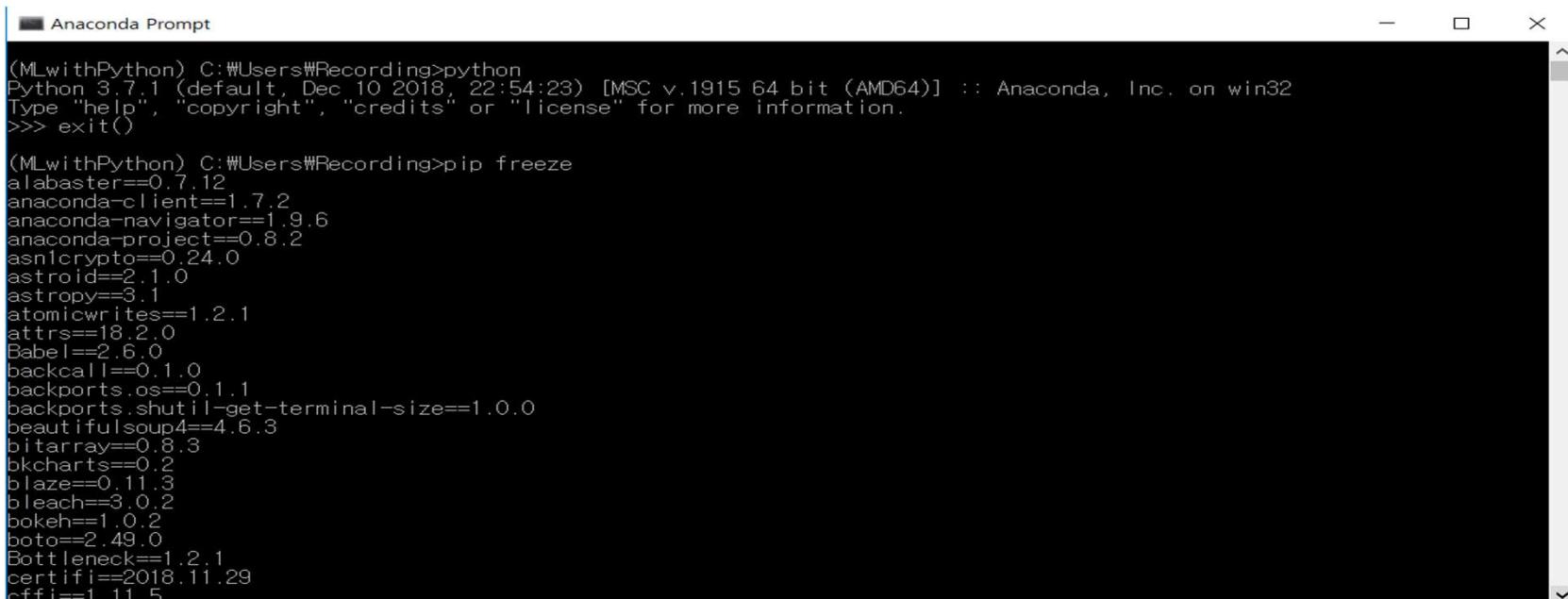
- <https://www.anaconda.com/distribution>



# Anaconda 설치



- 개인별 작업 directory 생성
- 패키지 설치 : pip install <package name>
- 설치된 Python package 확인 : pip list

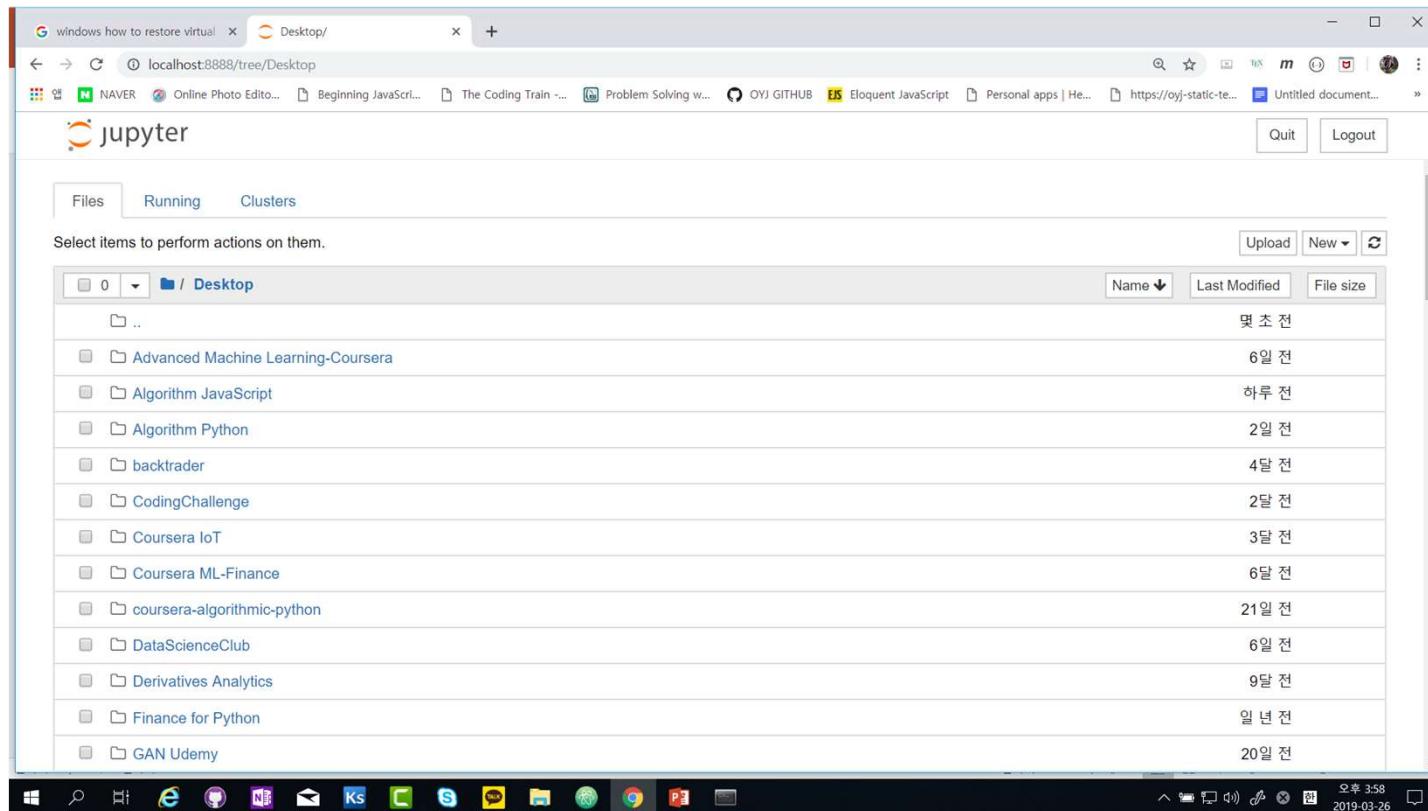


```
Anaconda Prompt

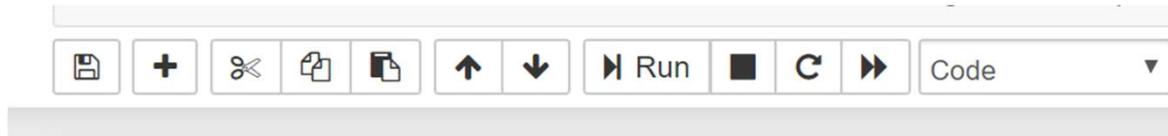
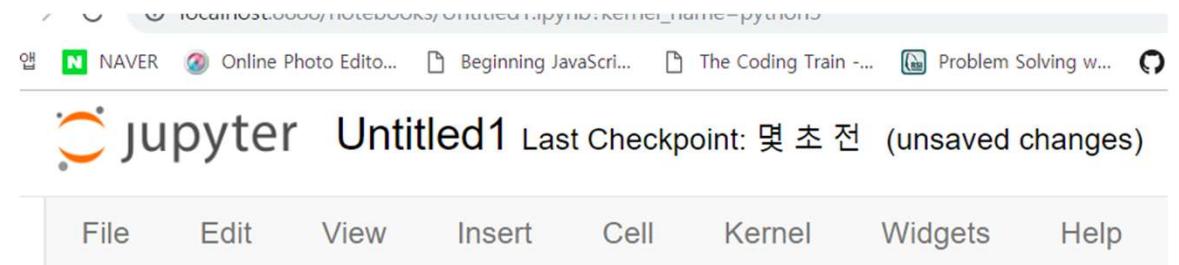
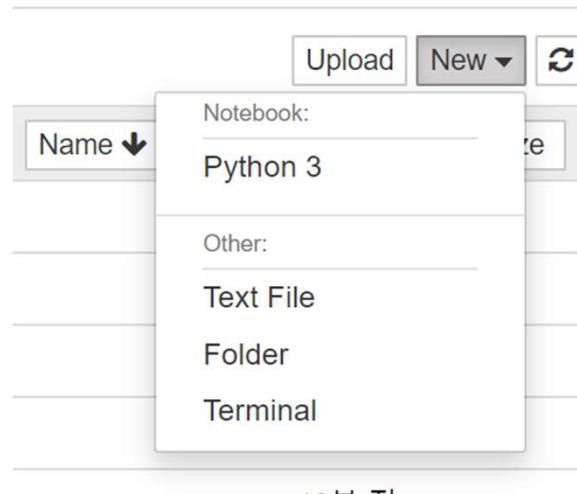
(MLwithPython) C:\Users\Recording>python
Python 3.7.1 (default, Dec 10 2018, 22:54:23) [MSC v.1915 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> exit()

(MLwithPython) C:\Users\Recording>pip freeze
alabaster==0.7.12
anaconda-client==1.7.2
anaconda-navigator==1.9.6
anaconda-project==0.8.2
asn1crypto==0.24.0
astroid==2.1.0
astropy==3.1
atomicwrites==1.2.1
attrs==18.2.0
Babel==2.6.0
backcall==0.1.0
backports.os==0.1.1
backports.shutil-get-terminal-size==1.0.0
beautifulsoup4==4.6.3
bitarray==0.8.3
bkcharts==0.2
blaze==0.11.3
bleach==3.0.2
bokeh==1.0.2
boto==2.49.0
Bottleneck==1.2.1
certifi==2018.11.29
cffi==1.11.5
```

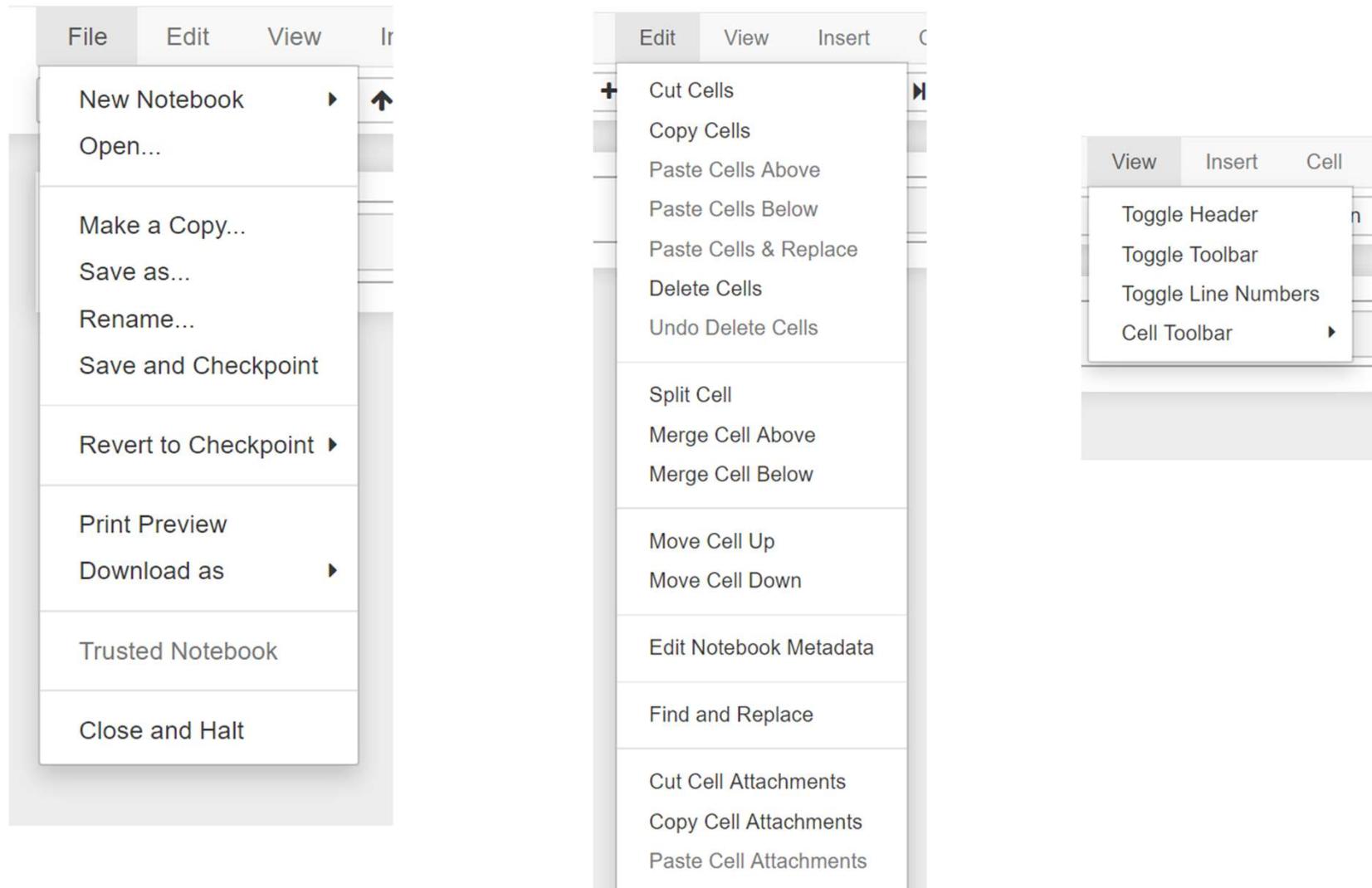
# >jupyter notebook



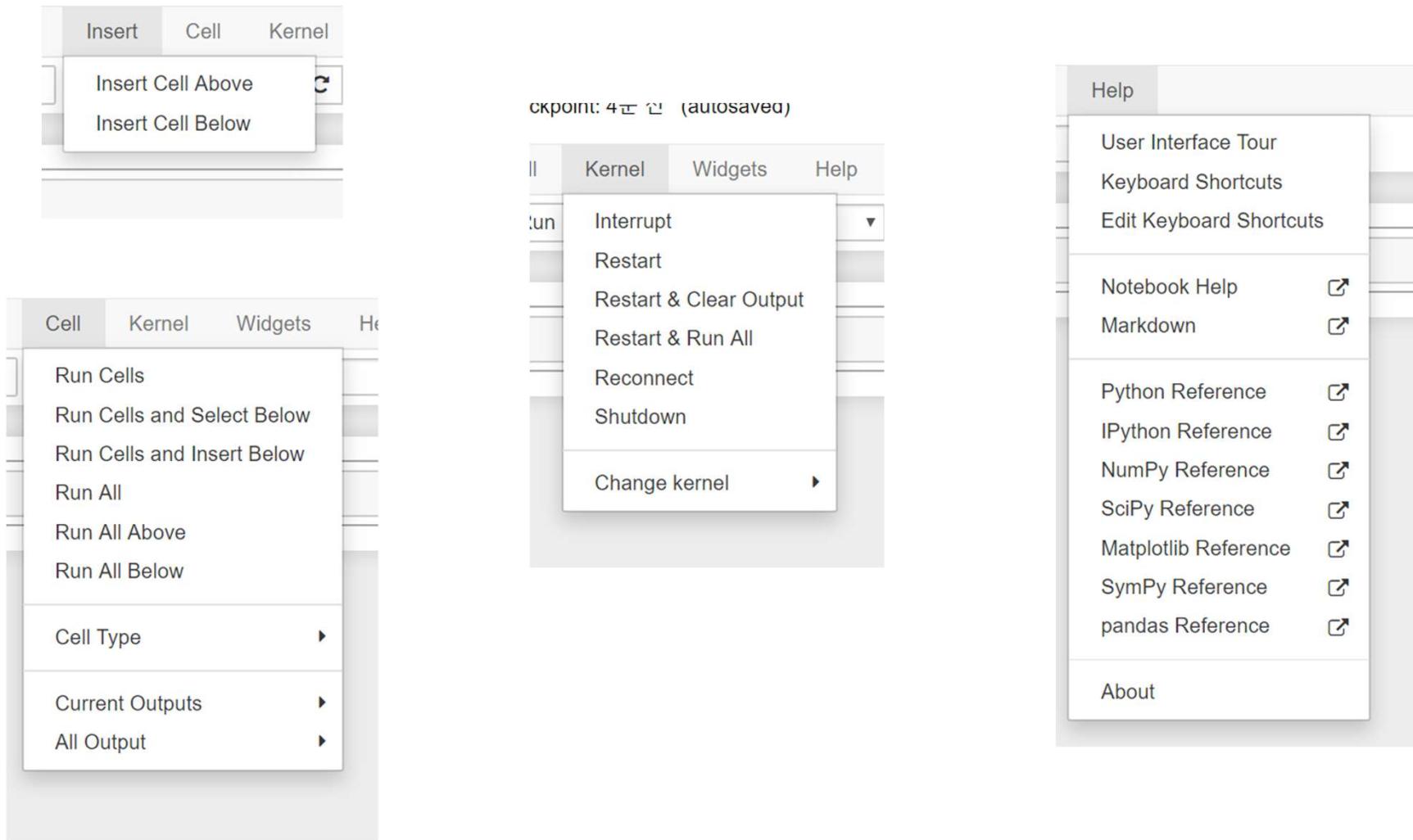
# Jupyter Notebook 사용방법



# Jupyter Notebook 사용방법

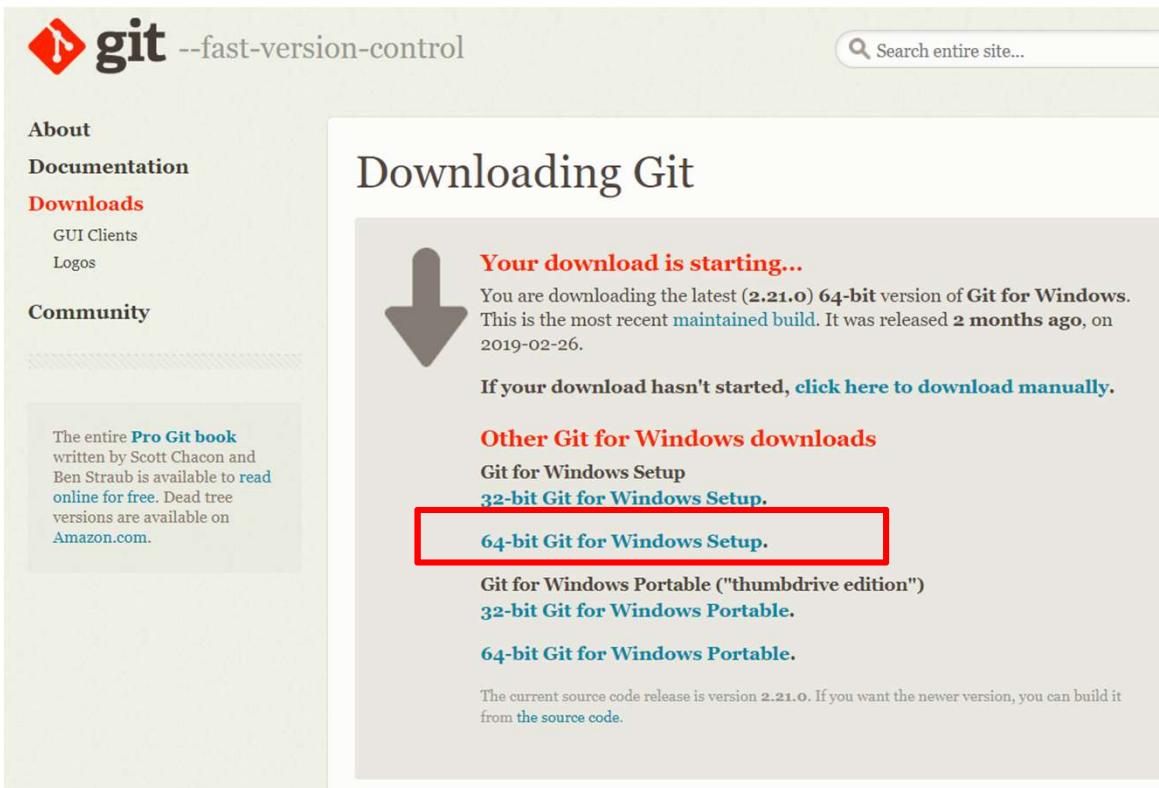


# Jupyter Notebook 사용방법



# git Installation

<https://git-scm.com/download/win>



The image shows a screenshot of the Git website. The header features the Git logo and the tagline "fast-version-control". A search bar is located in the top right corner. The left sidebar contains links for "About", "Documentation", "Downloads" (which is highlighted in red), and "Community". A sidebar box promotes the "Pro Git book". The main content area is titled "Downloading Git". It displays a large downward arrow icon and the text "Your download is starting...". Below this, it says "You are downloading the latest (2.21.0) 64-bit version of Git for Windows. This is the most recent maintained build. It was released 2 months ago, on 2019-02-26." It also includes a link "click here to download manually". A red box highlights the "64-bit Git for Windows Setup." link in the "Other Git for Windows downloads" section. The section also lists "Git for Windows Setup", "32-bit Git for Windows Setup.", and "64-bit Git for Windows Portable." A note at the bottom states: "The current source code release is version 2.21.0. If you want the newer version, you can build it from the source code."

# Github Repository

[\*\*https://github.com/ironmanciti/machineLearningBasic\*\*](https://github.com/ironmanciti/machineLearningBasic)

The screenshot shows a GitHub repository page for the user 'ironmanciti' with the repository name 'machineLearningBasic'. The page includes a navigation bar with 'Code' (selected), 'Issues 0', 'Pull requests 0', 'Projects 0', 'Wiki', 'Insights', and 'Settings'. Below the navigation bar, there is a summary section with metrics: 1 commit, 1 branch, 0 releases, and 1 contributor. The 'Branch: master' dropdown is set to 'master'. There are buttons for 'New pull request', 'Create new file', 'Upload files', 'Find File', and a prominent green 'Clone or download' button. The repository's README.md file is displayed, showing the text '파이썬을 활용한 머신러닝 입문 과정' (Introduction to Machine Learning with Python). The repository has 0 stars, 0 forks, and 0 issues. The 'Clone with HTTPS' and 'Use SSH' options are shown, along with the URL <https://github.com/ironmanciti/machineLearningBasic>. There are also 'Open in Desktop' and 'Download ZIP' buttons.

ironmanciti / machineLearningBasic

Watch 0 | Star 0 | Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

파이썬을 활용한 머신러닝 입문 과정

Edit

Manage topics

1 commit 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find File Clone or download

ironmanciti Initial commit

README.md Initial commit

README.md

Clone with HTTPS Use SSH

Use Git or checkout with SVN using the web URL.

<https://github.com/ironmanciti/machineLearningBasic>

Open in Desktop Download ZIP

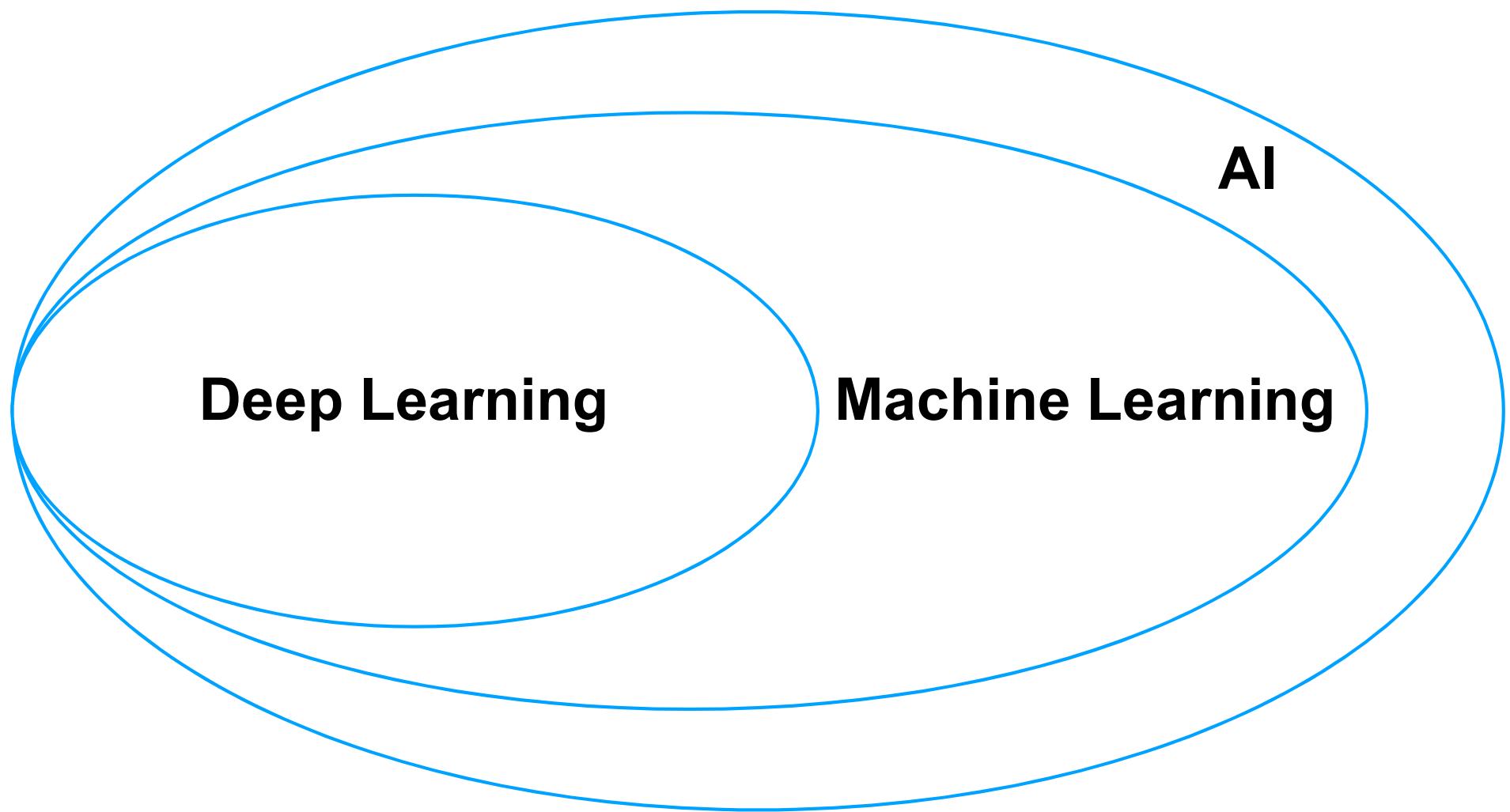
## machineLearningBasic

파이썬을 활용한 머신러닝 입문 과정

# Machine Learning

## 개요

# AI vs ML vs DL



# History of Machine Learning

- **탄생 [1950년대] : 1958**
  - Frank Rosenblatt 가 인간의 뇌신경을 본떠 Perceptron 고안
- **AI의 첫번째 Winter [1970년대] :**  
Marvin Minsky 가 Perceptron 은 XOR 문제를 해결할 수 없음을 수학적으로 증명
- **중흥기 [1980년대] : Expert System**
- **AI의 두번째 Winter [1987 – 1993]**
  - Jeffery Hinton (Toronto 대학) back-propagation algorithm 개발

- **IBM Deep Blue** 가 **Garry Kasparov** 에 승리 - 1996
- **Google Brain** 이 최초로 인간 얼굴 인식 - 2012
- ~ 상업적 대 폭발기
- **AlphaGo** 이세돌에 승리 – 2016



**Andrew Ng.**  
Stanford Univ.  
Google/Baidu



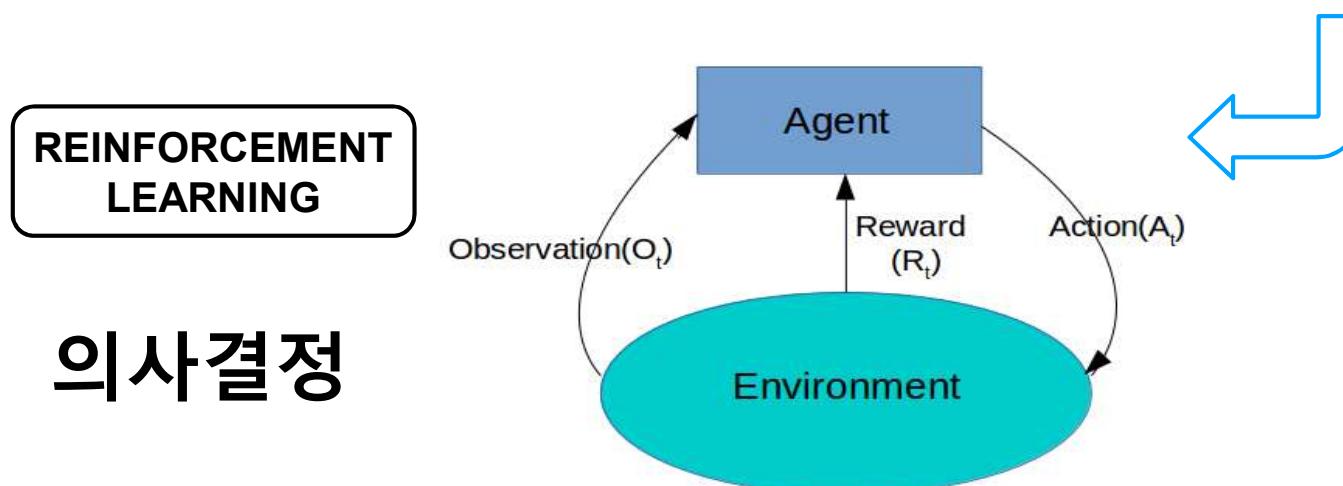
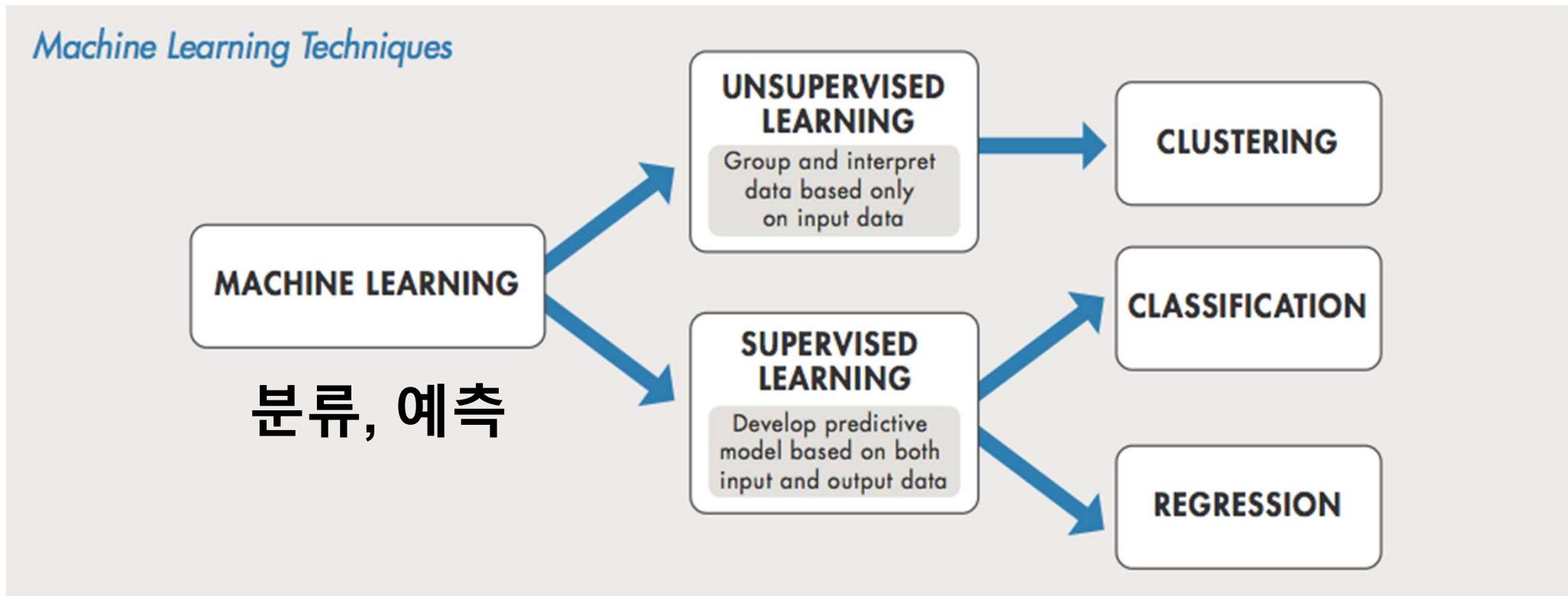
**Jeffery Hinton**  
Toronto Univ.  
Google

**HEROs**

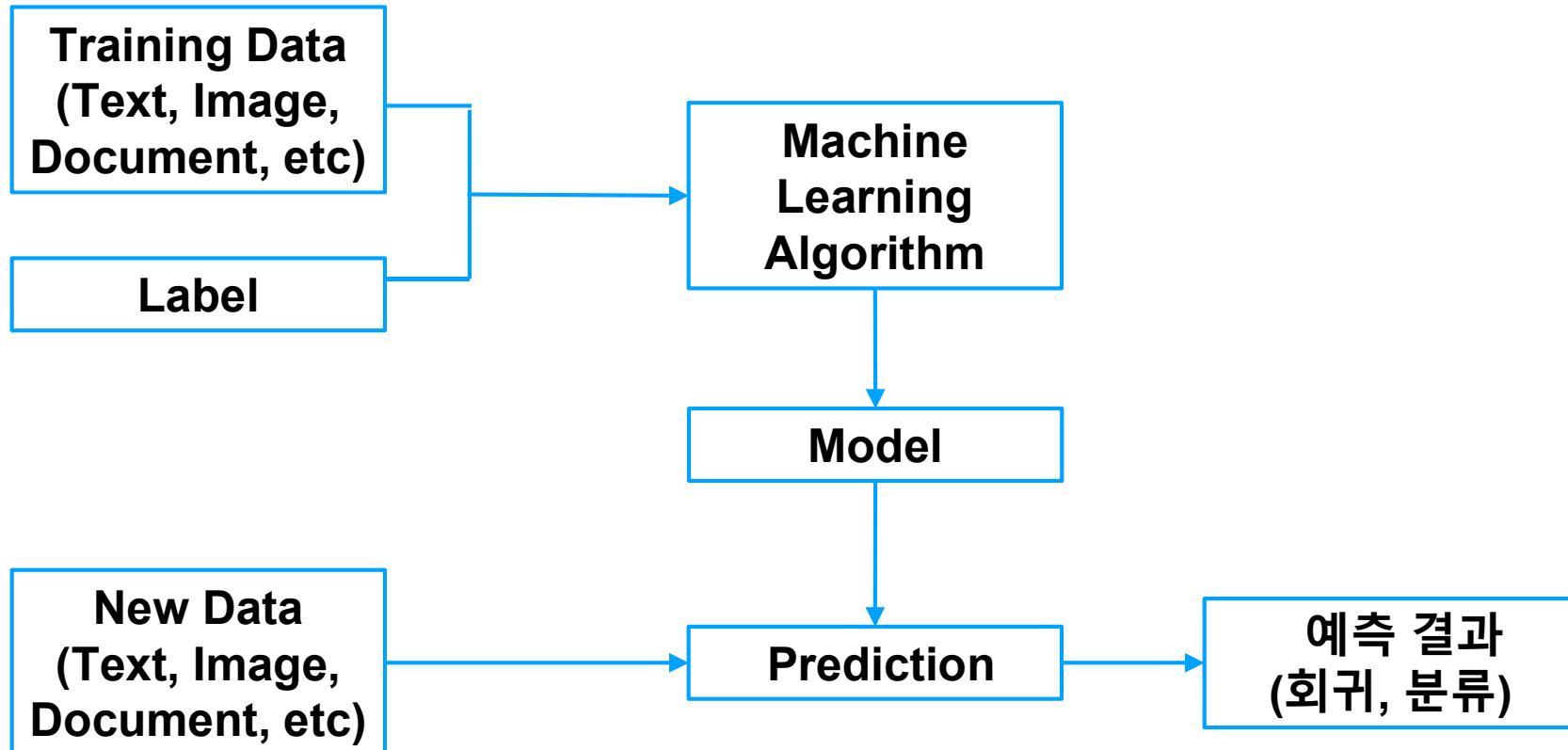


**Yan LeCun**  
New York Univ.  
Facebook

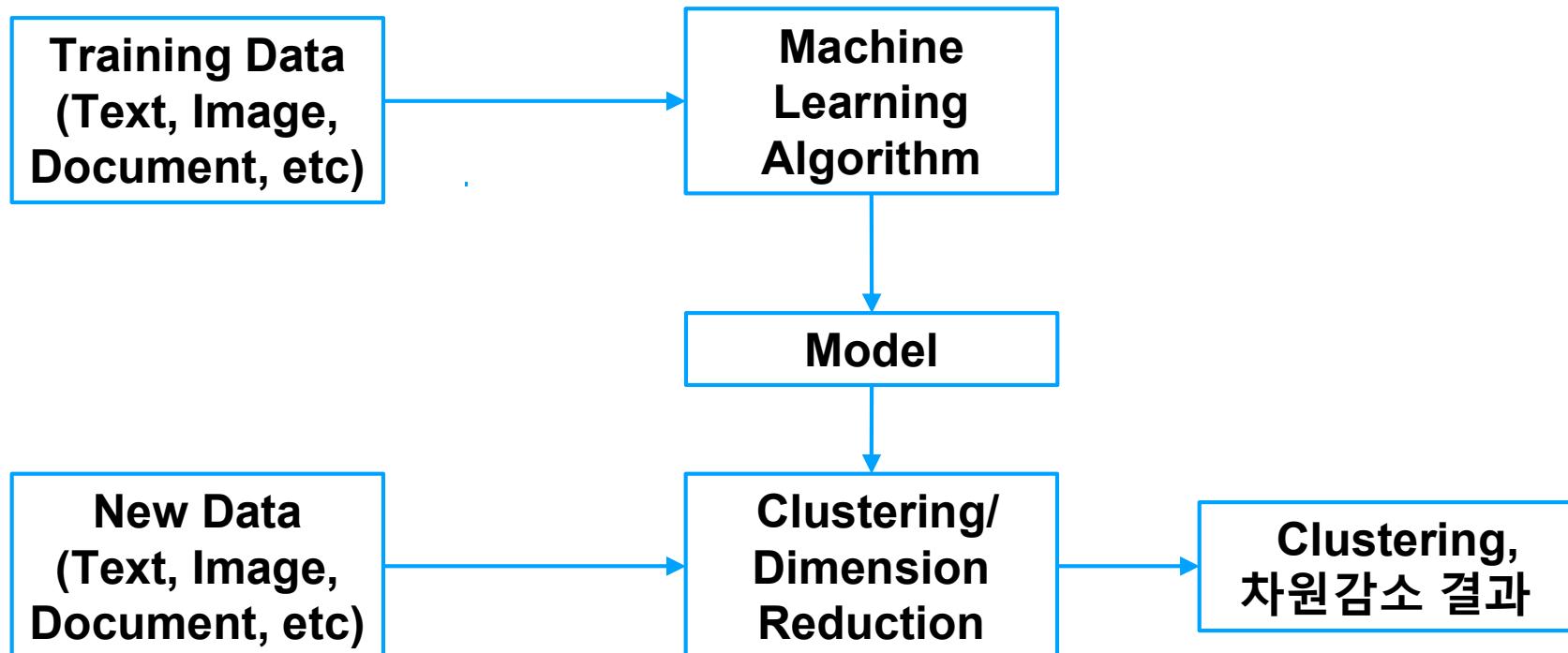
# Machine Learning 의 종류



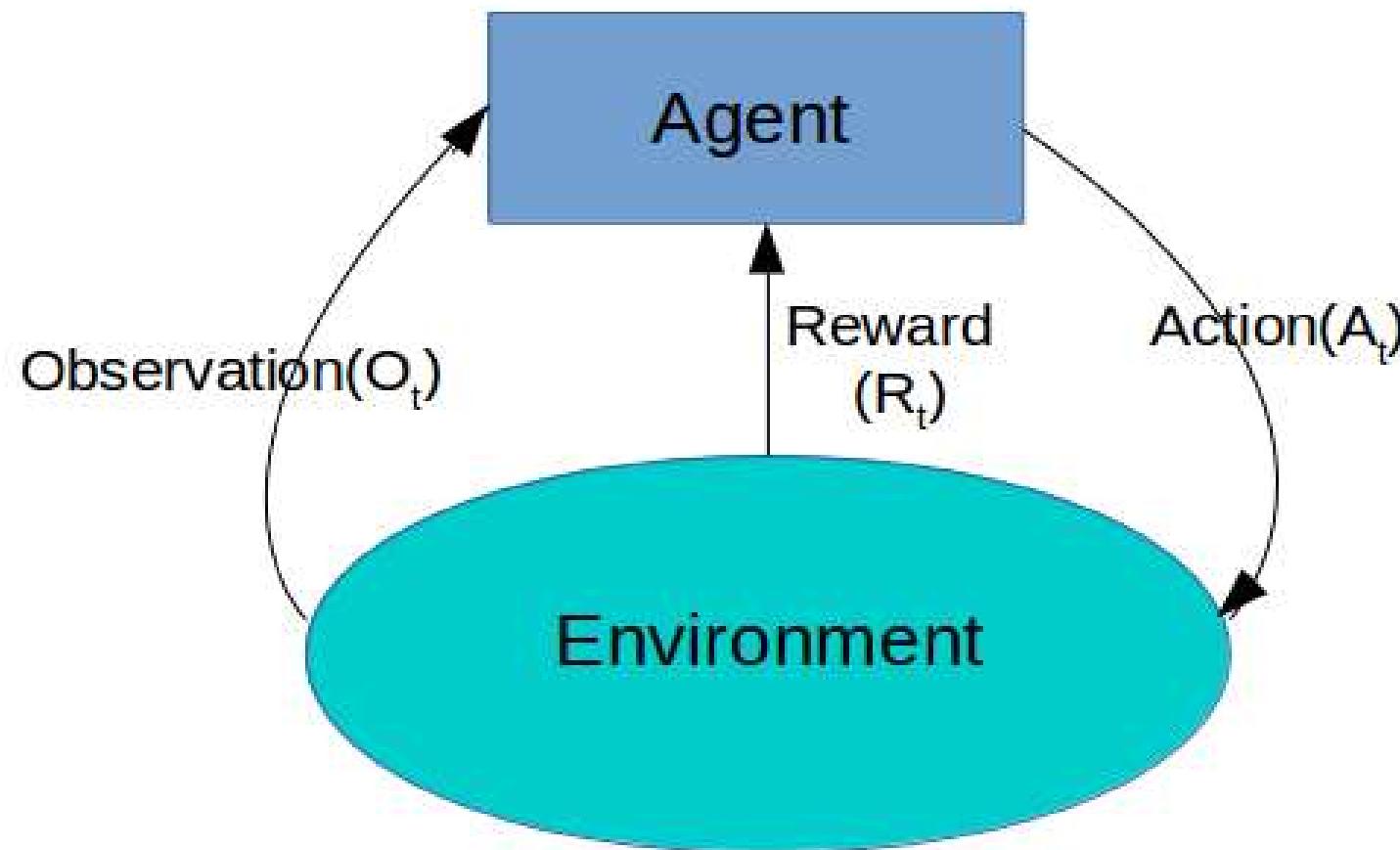
# Supervised Learning (지도학습)



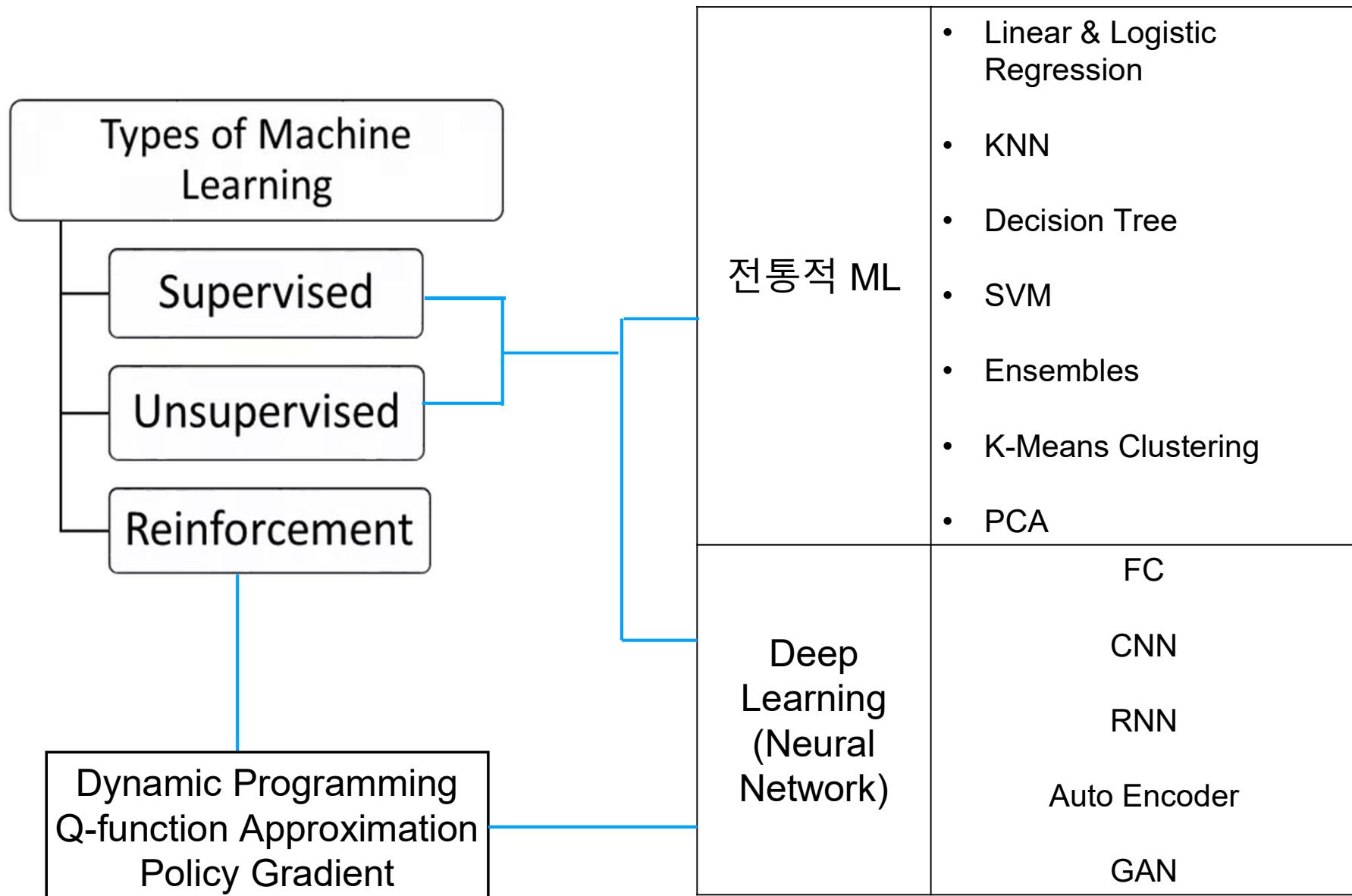
# Unsupervised Learning (비지도학습)



# Reinforcement Learning (강화학습)



# Machine Learning 기법의 종류



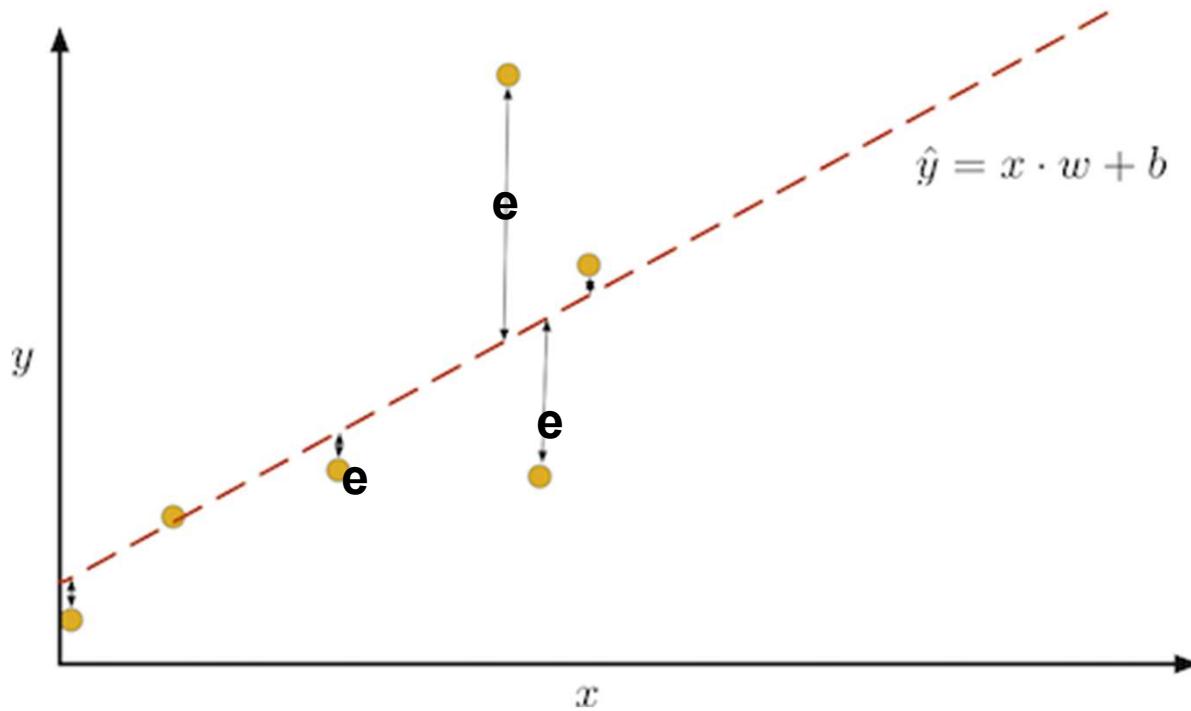
# 전통적 ML의 기법

종류	용도
Linear / Logistic Regression	선형회귀/분류
KNN (K-Nearest Neighbor)	분류
Decision Tree (결정나무)	분류
SVM (Support Vector Machine)	분류, 회귀
Ensemble (Random Forest, XGBoost, etc)	분류, 회귀
K-Means Clustering (K-평균 군집화)	군집화
PCA (Principal Component Analysis)	차원 축소

# Neural Network 의 종류

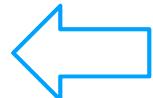
종류	용도
FC (Fully Connected Neural Network)	분류, 회귀
CNN (Convolutional Neural Network)	Image 인식
RNN (Recurrent Neural Network)	시계열 인식, 자동번역, 감성분석 등
AE (Auto Encoder)	비지도 학습, 차원축소
GAN (Generative Adversarial Nets)	적대적 생성모델, 이미지 위조

# Linear Regression



Ex)  $x$  : 강우량

Machine Learning 은  
주어진 data 에 대하여  
Parameter 를  
inference 하는 작업



$y$  : 실제우산 판매량 (label)

$\hat{y}$  : 모델에 의한 예측 값

$w, b$  : 학습에 의한 model parameter

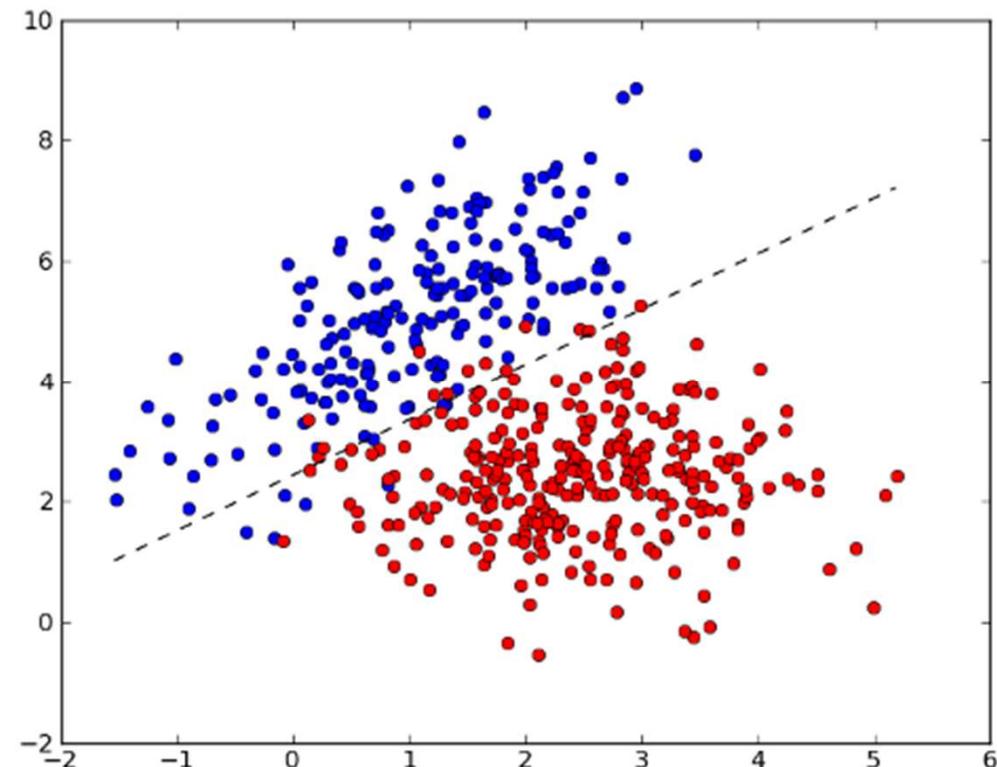
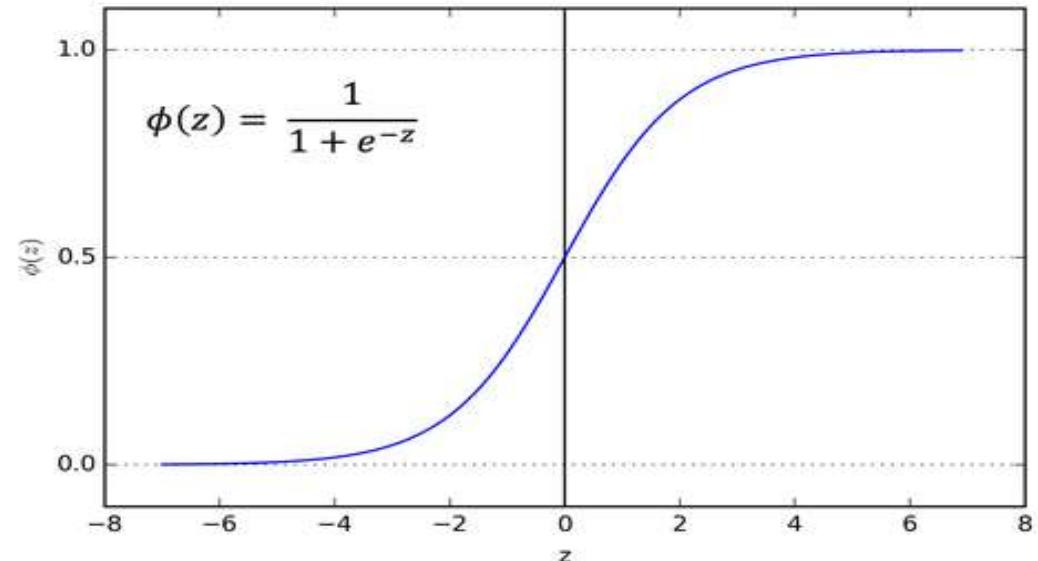
$e$  : 예측 값과 실제 값의 차이

# Logistic Regression

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

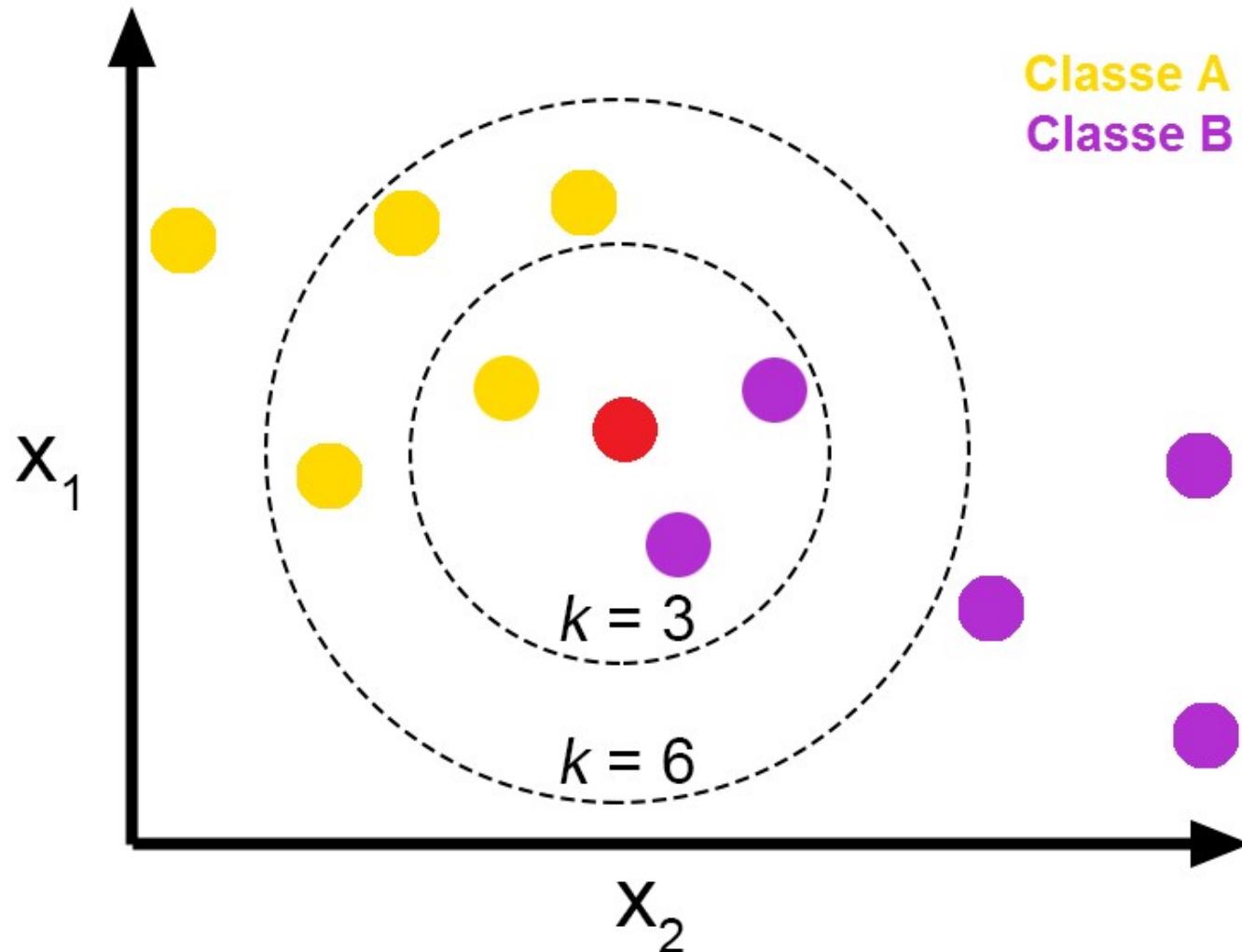
$$z : \mathbf{w}\mathbf{x} + b$$

If  $\Phi(z) > 0.5 \rightarrow$  blue  
 $< 0.5 \rightarrow$  red

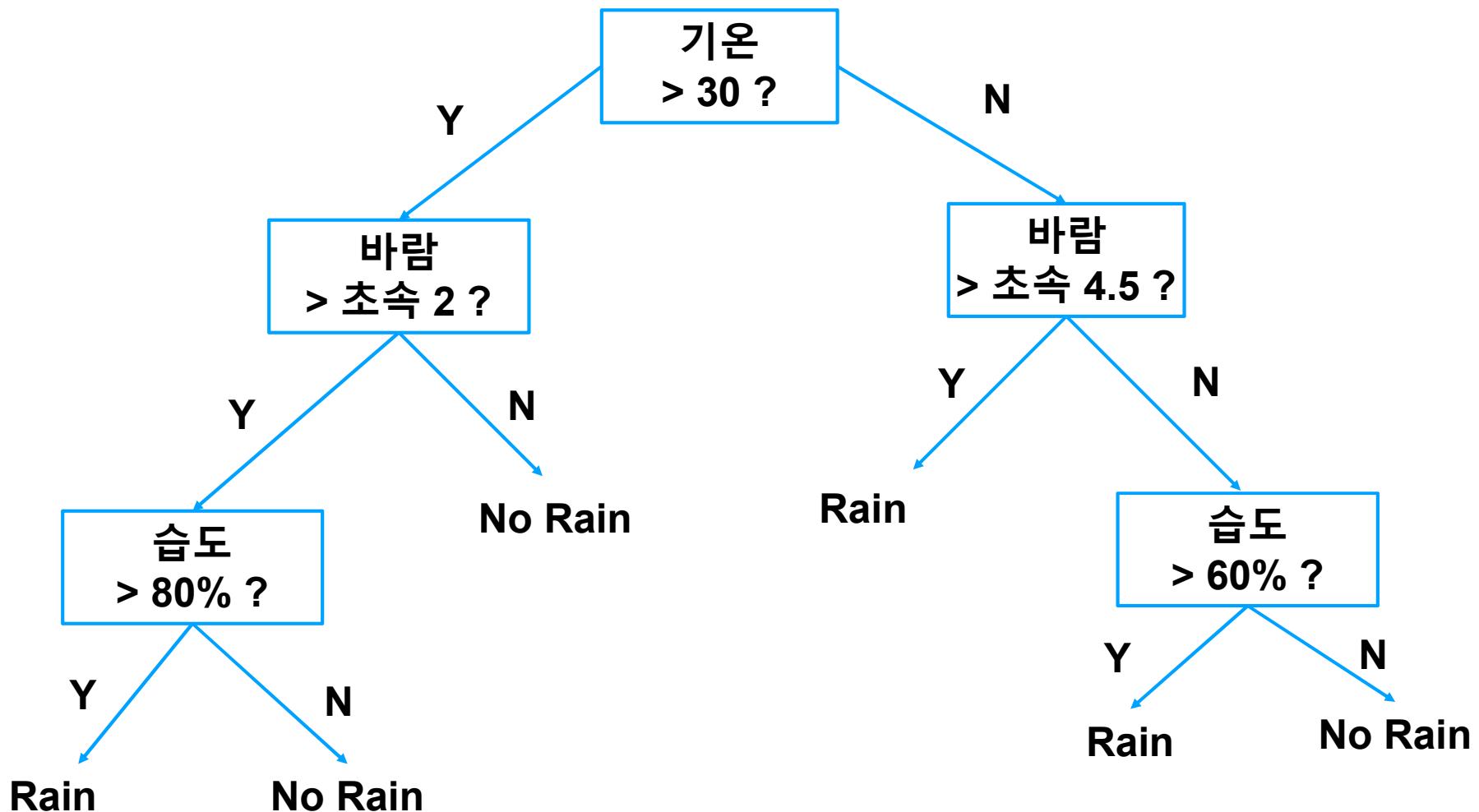


# KNN (K-Nearest Neighbor)

If  $K=3 \rightarrow$  Class B  
If  $K=6 \rightarrow$  Class A



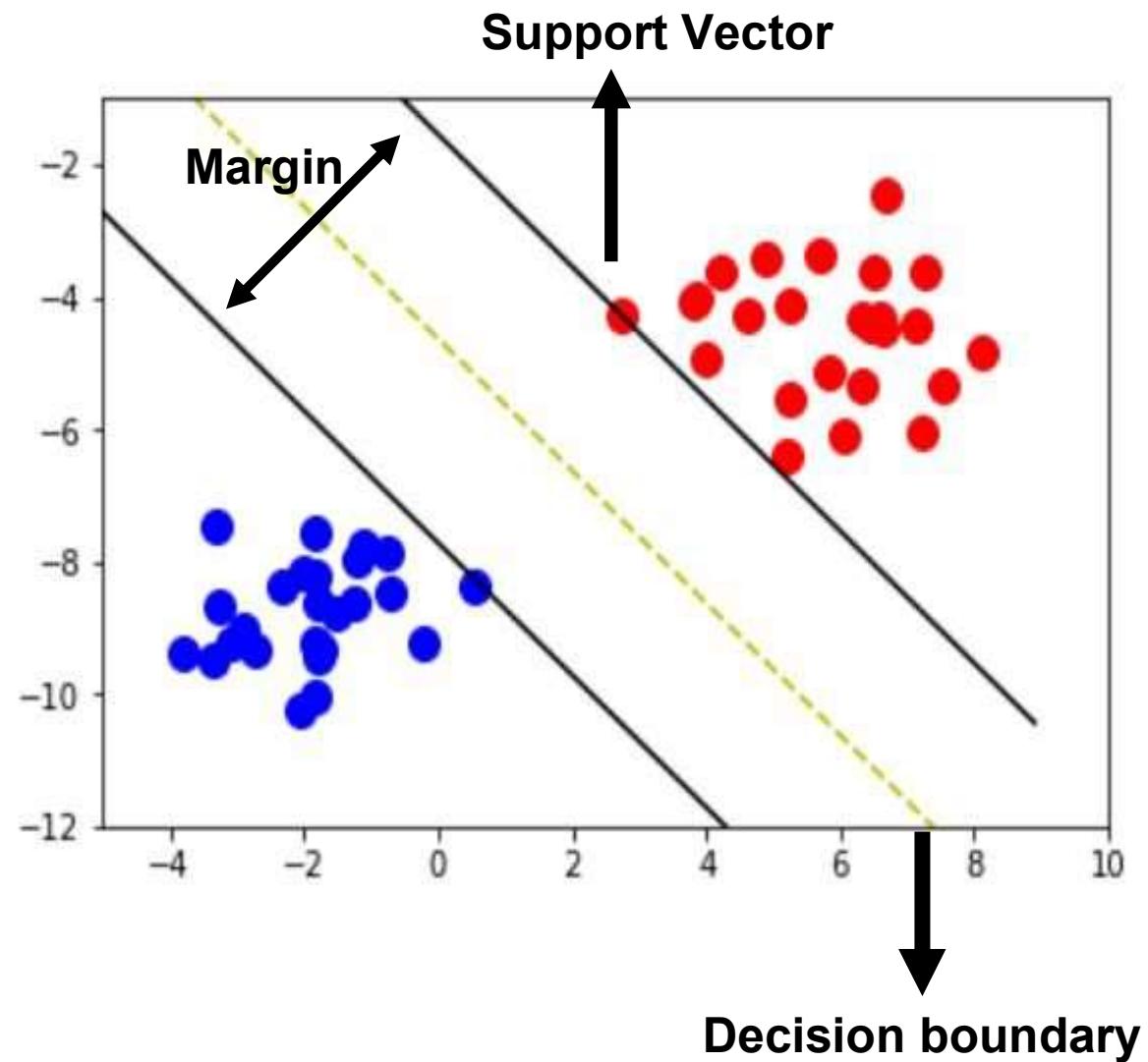
# Decision Tree (결정나무)



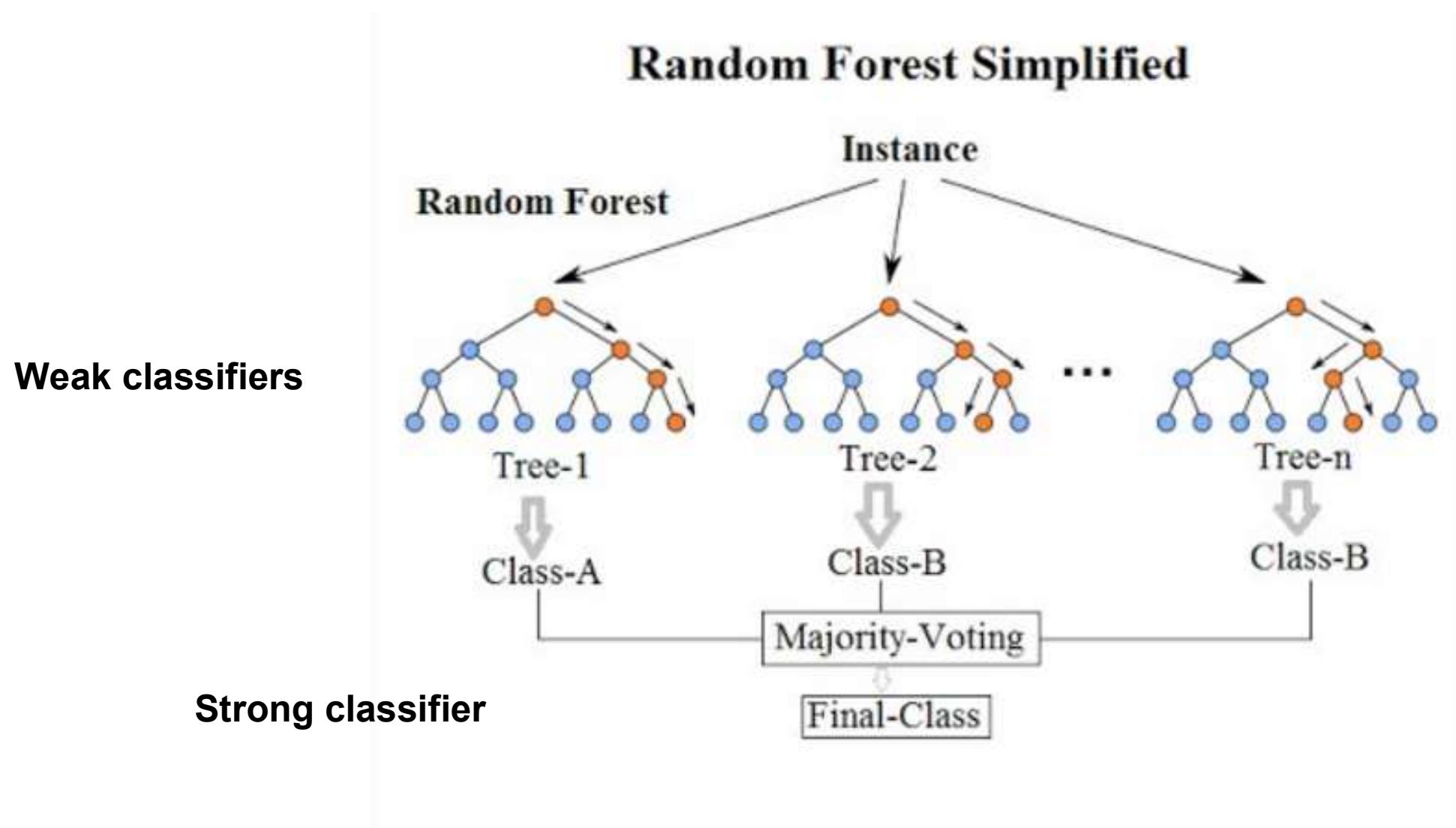
# SVM (Support Vector Machine)

$$f(x) = \text{sgn}(\mathbf{w} \cdot \mathbf{x} + b)$$

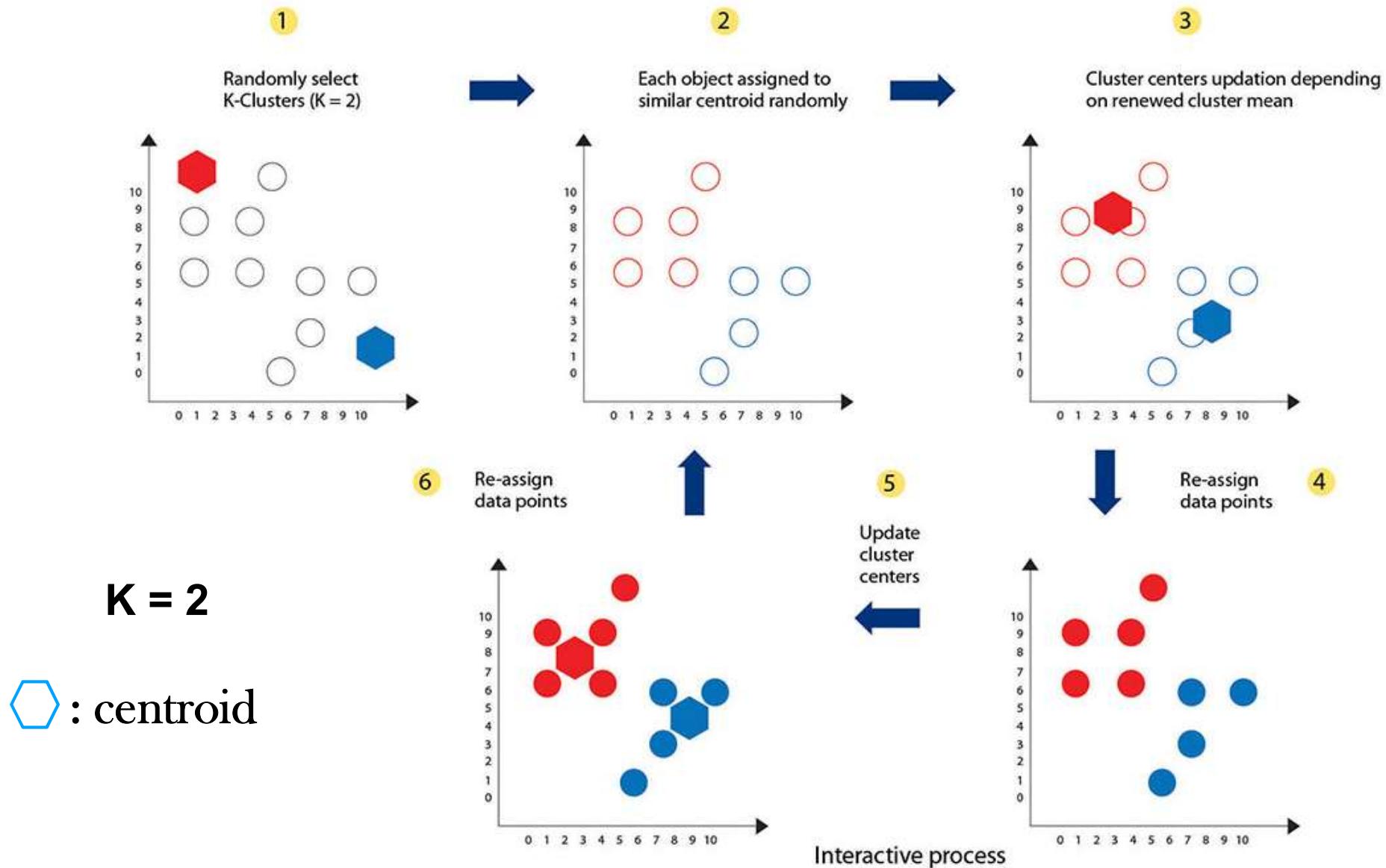
$\hat{y} = 0$  if  $f(x) < 0$   
 $1$  if  $f(x) \geq 0$



# Ensemble Random Forest



# K-Means Clustering (K-평균 군집화)



# PCA (Principal Component Analysis) (주성분 분석)

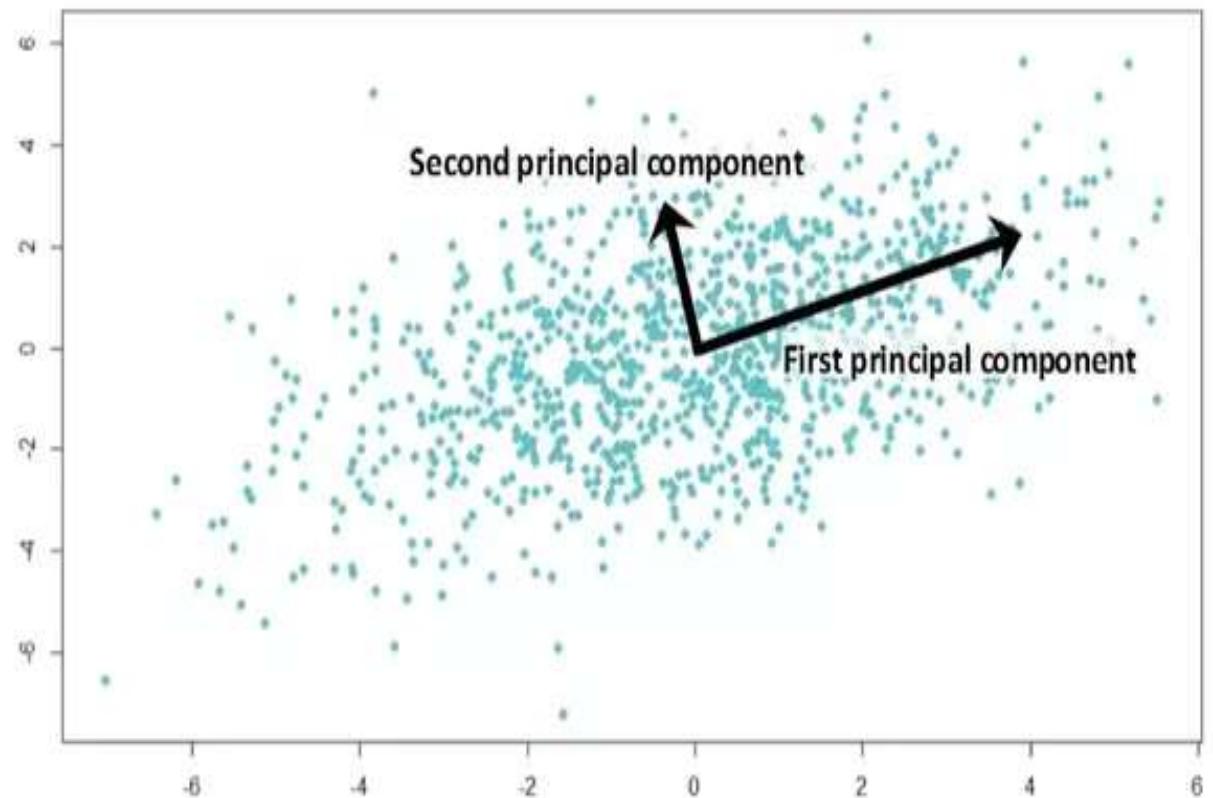
- 선형대수학의 SVD(Singular Value Decomposition) 이용
- 어떤  $m \times n$  행렬  $A$ 는 다음과 같은 형태의 세가지 행렬의 곱으로 분해할 수 있다.

$$A_{m \times n} = U_{n \times n} \Sigma_{n \times m} V^T_{m \times m}$$

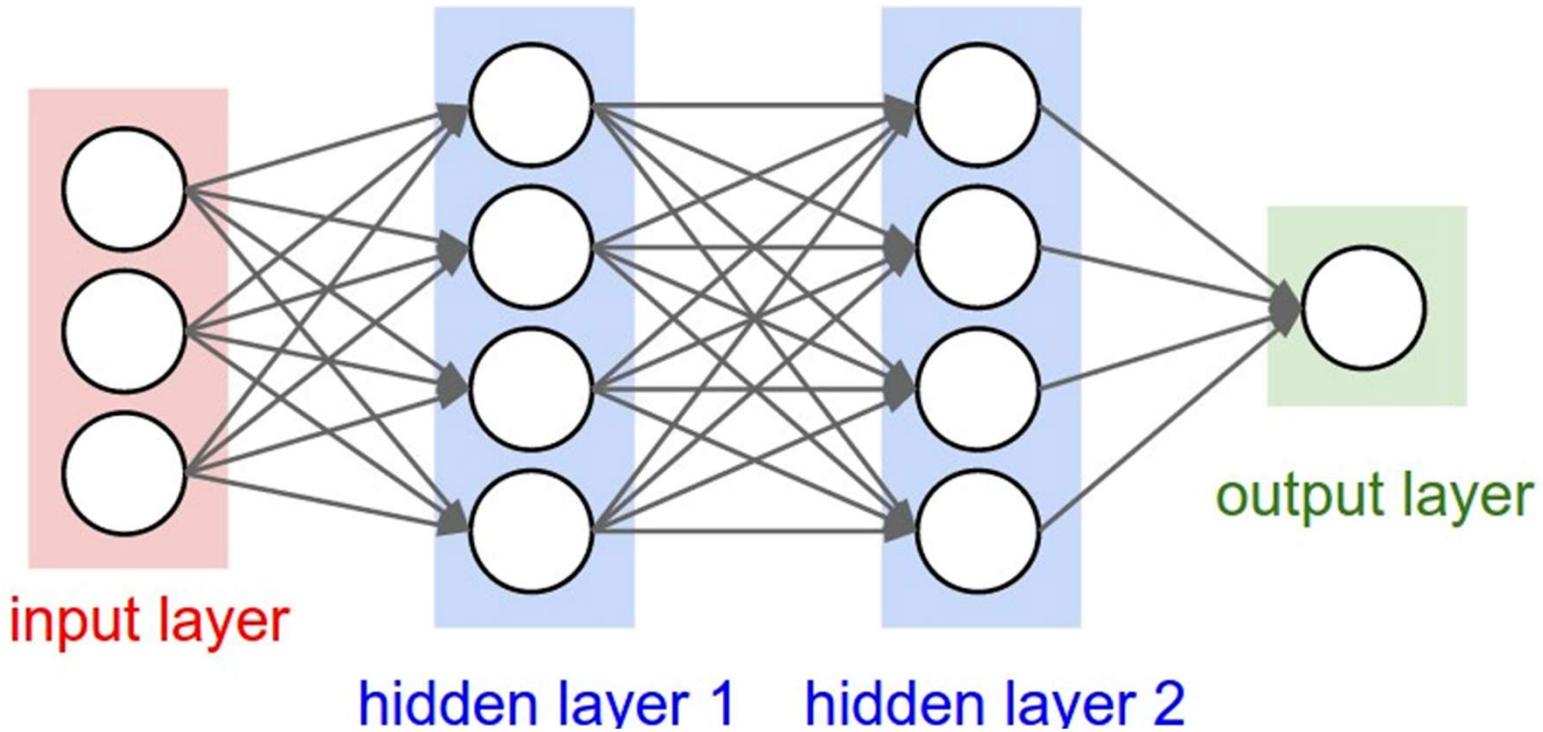
$U$  :  $n \times n$  직교행렬

$V$  :  $m \times m$  직교행렬

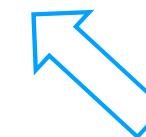
$\Sigma$  :  $n \times m$  직사각대각행렬



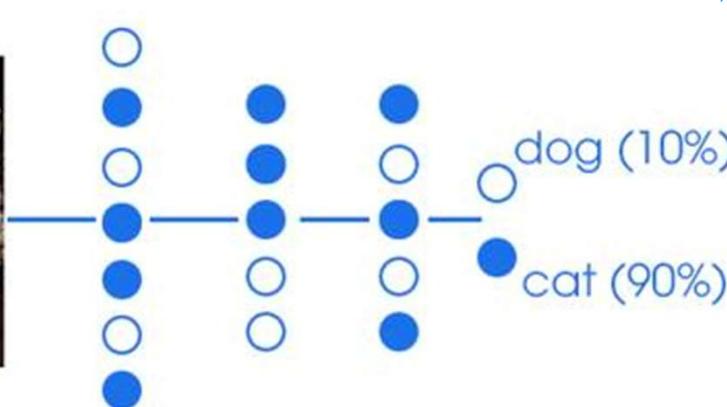
# Fully Connected Neural Network



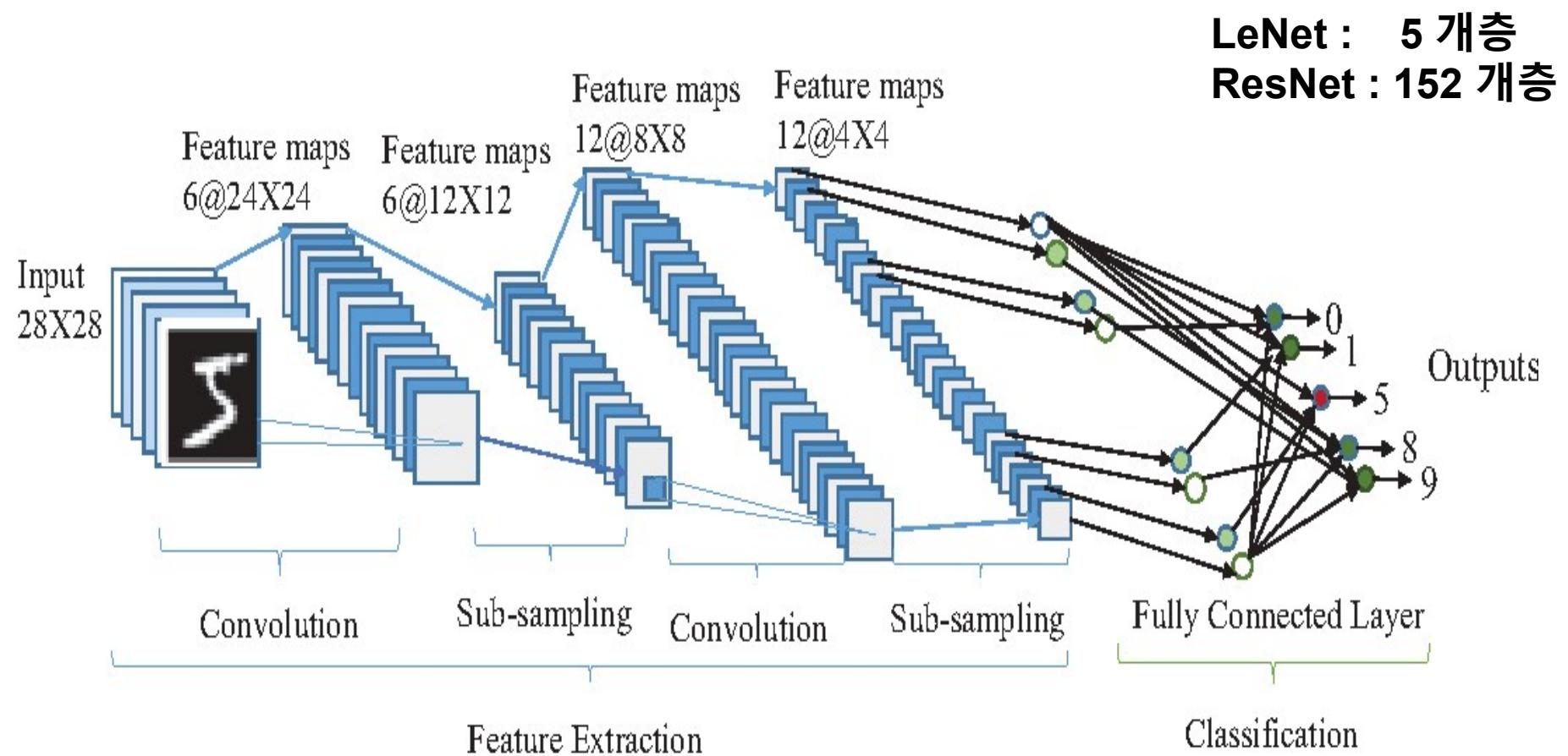
hidden layer 1    hidden layer 2



**HOW DEEP ?**



# CNN (Convolutional Neural Network, 합성곱 신경망)



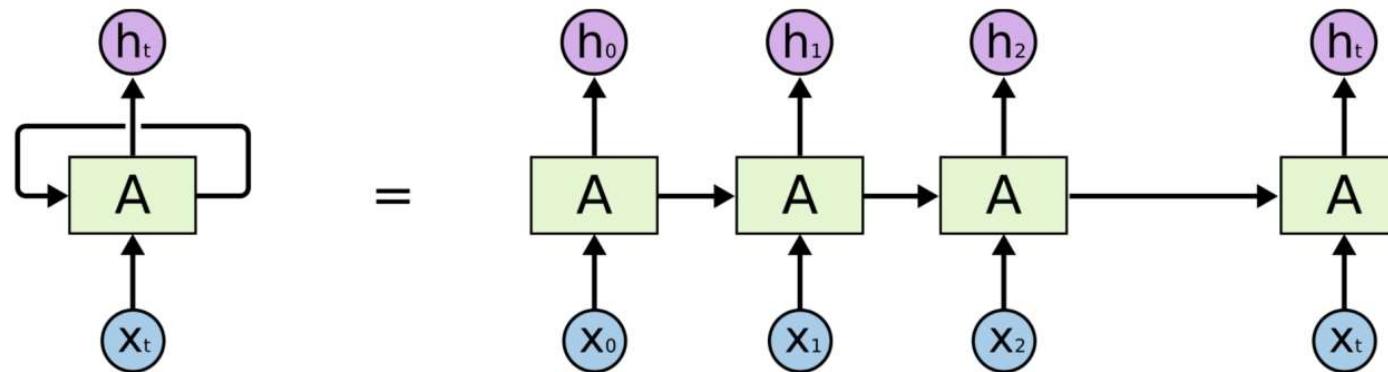
입력

합성곱

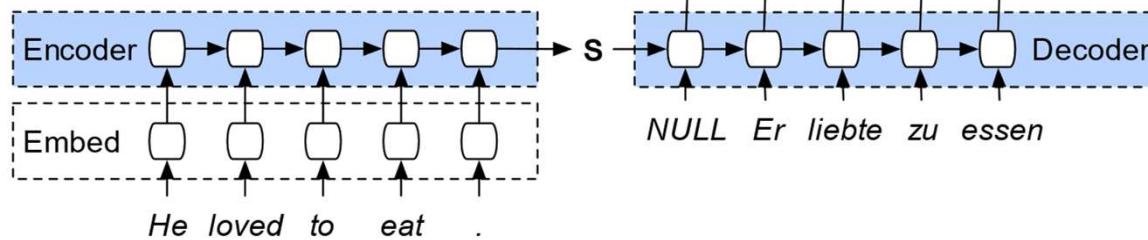
풀링

완전연결

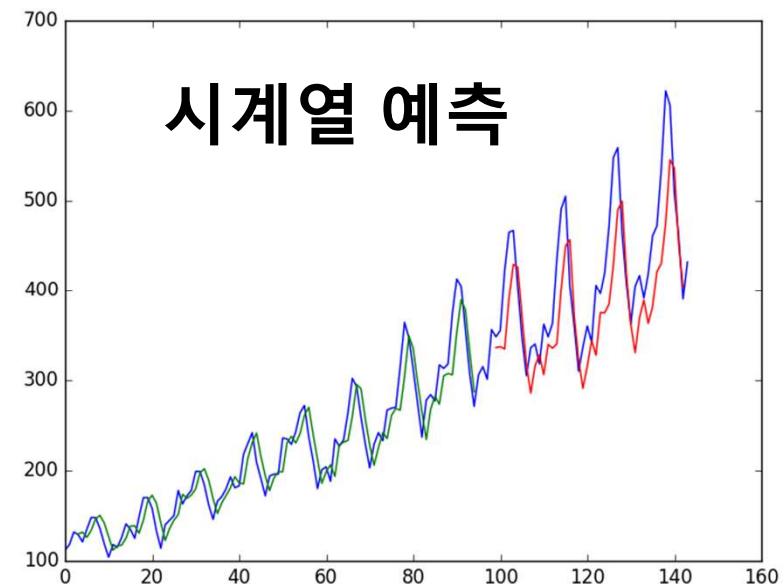
# RNN (Recurrent Neural Network, 순환 신경망)



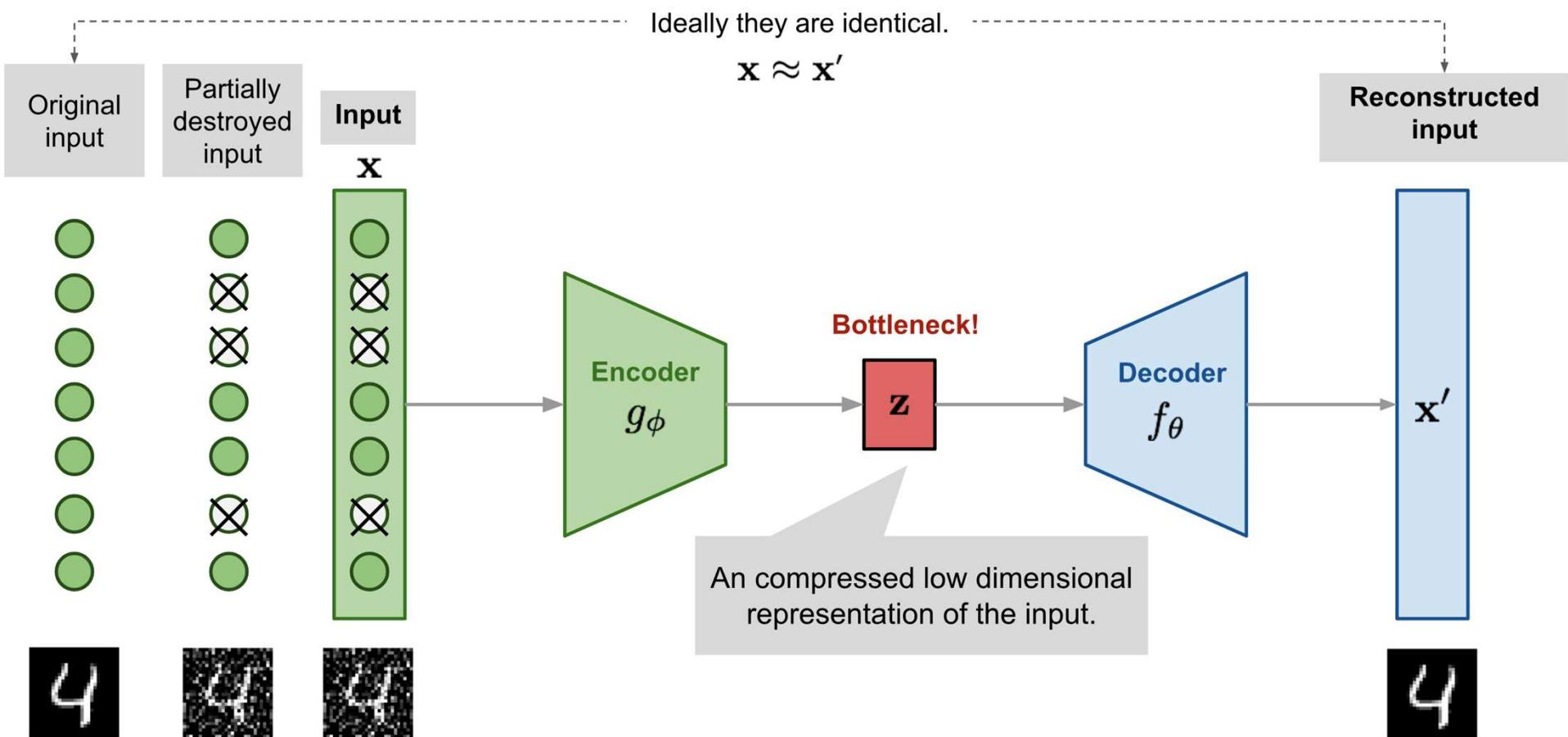
## 자연어 처리



시계열 예측

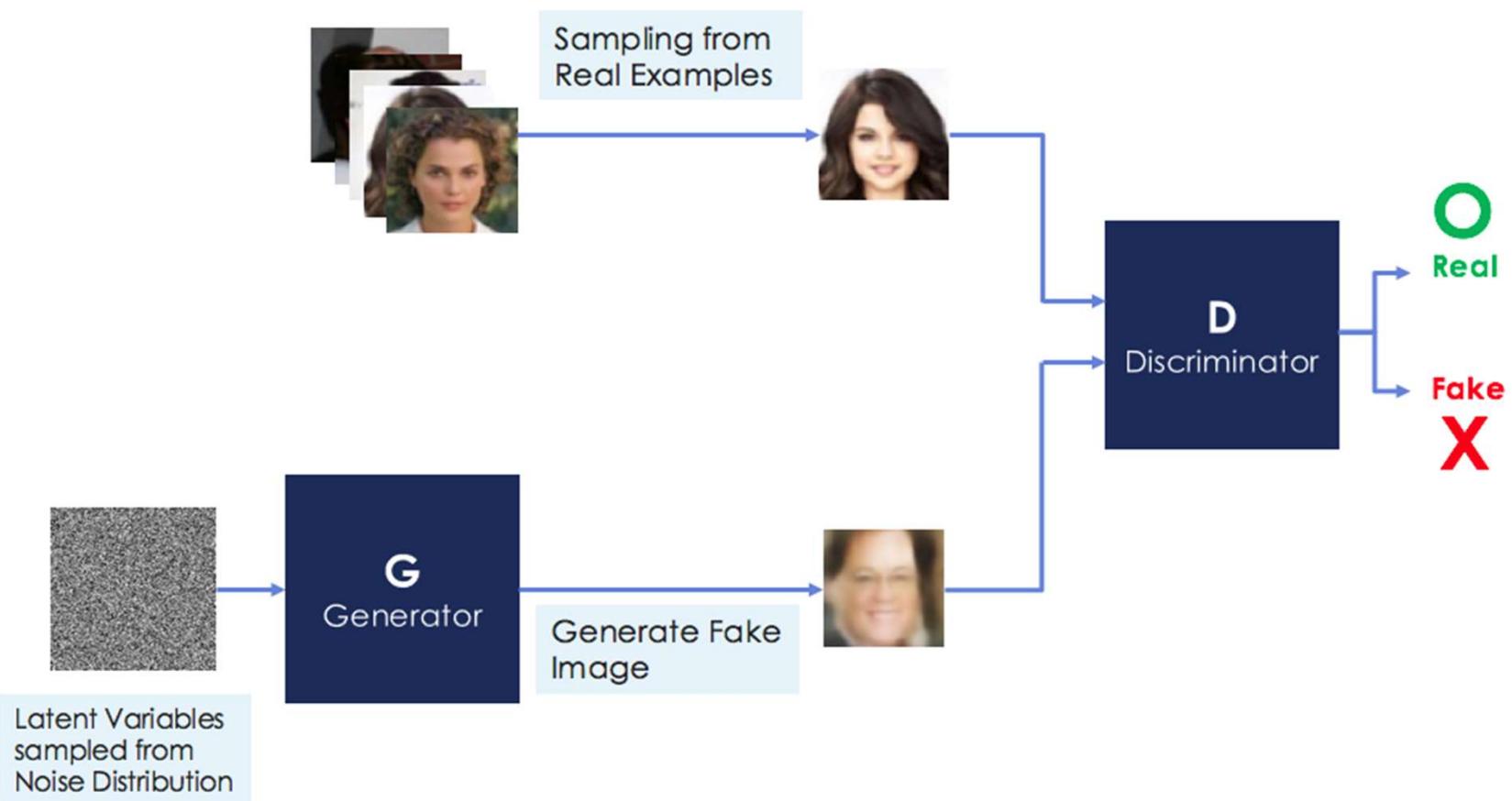


# Autoencoder (특성추출, 차원감소)



# GAN(적대적 생성모델)

## Generative Adversarial Networks(GAN)

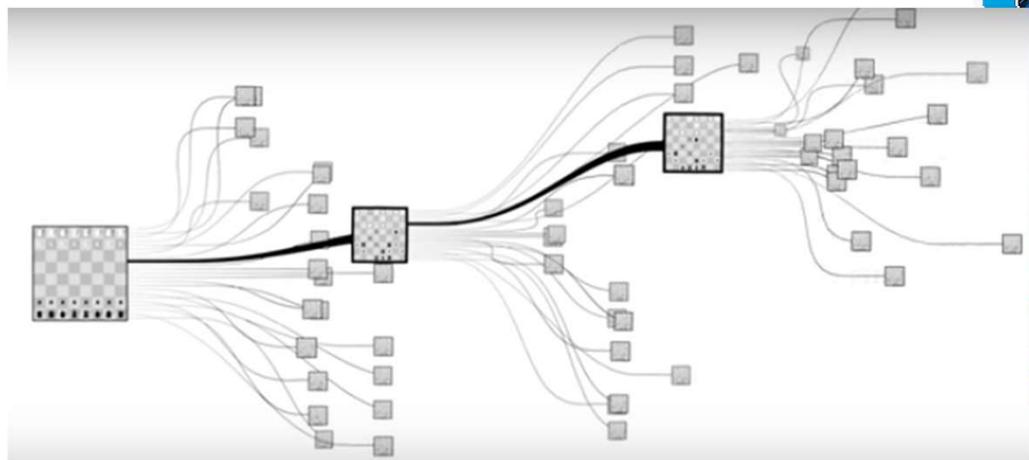




Alphago-Lee

Alphago-Master

Deep Learning 과  
강화학습의 만남

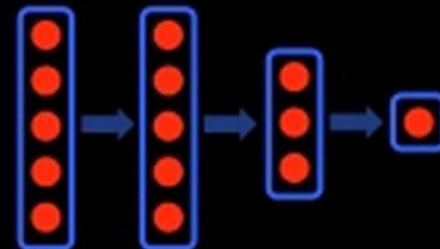


## Why is Deep Learning taking off?



Engine

Fuel



Large neural networks

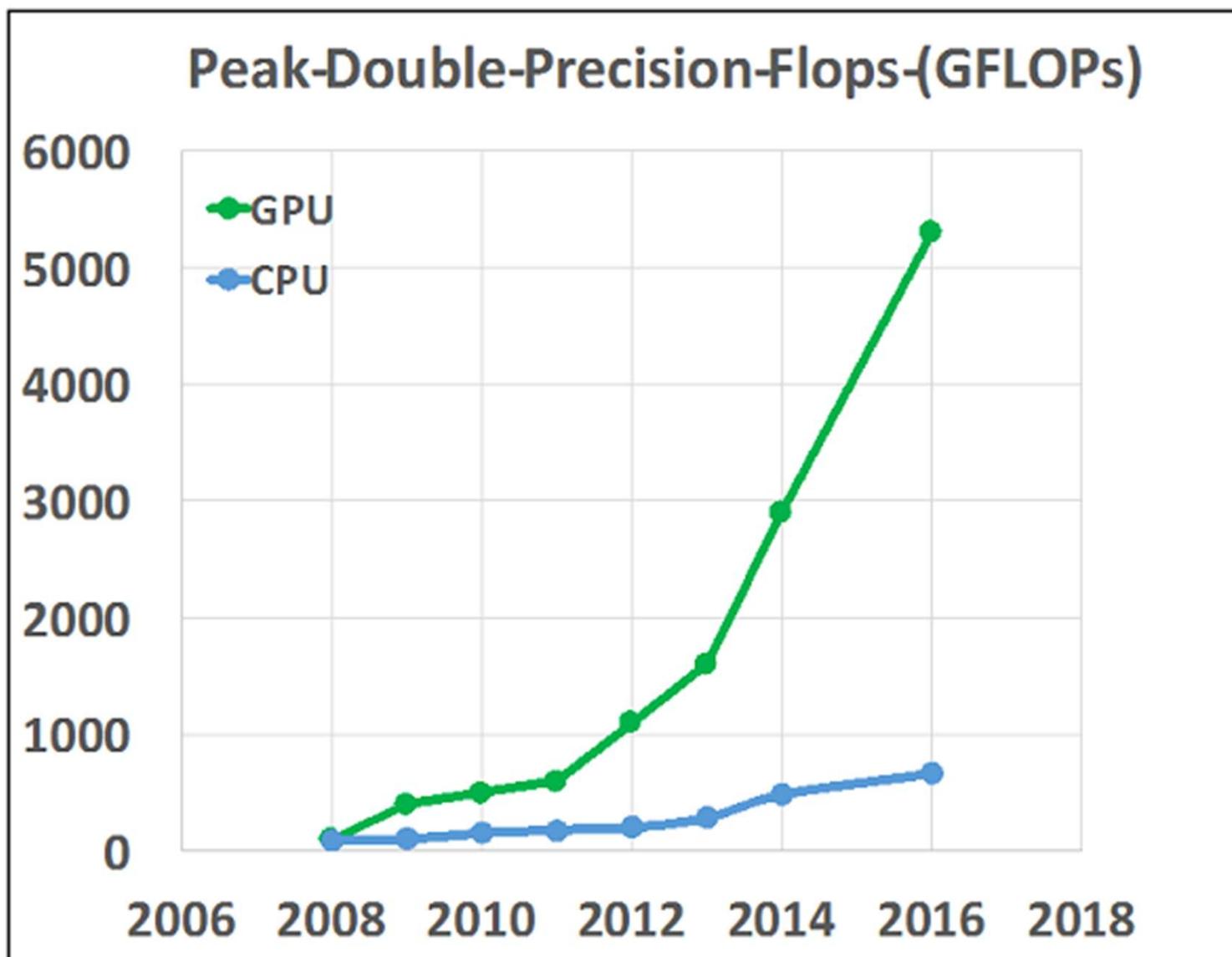


Labeled data  
( $x, y$  pairs)

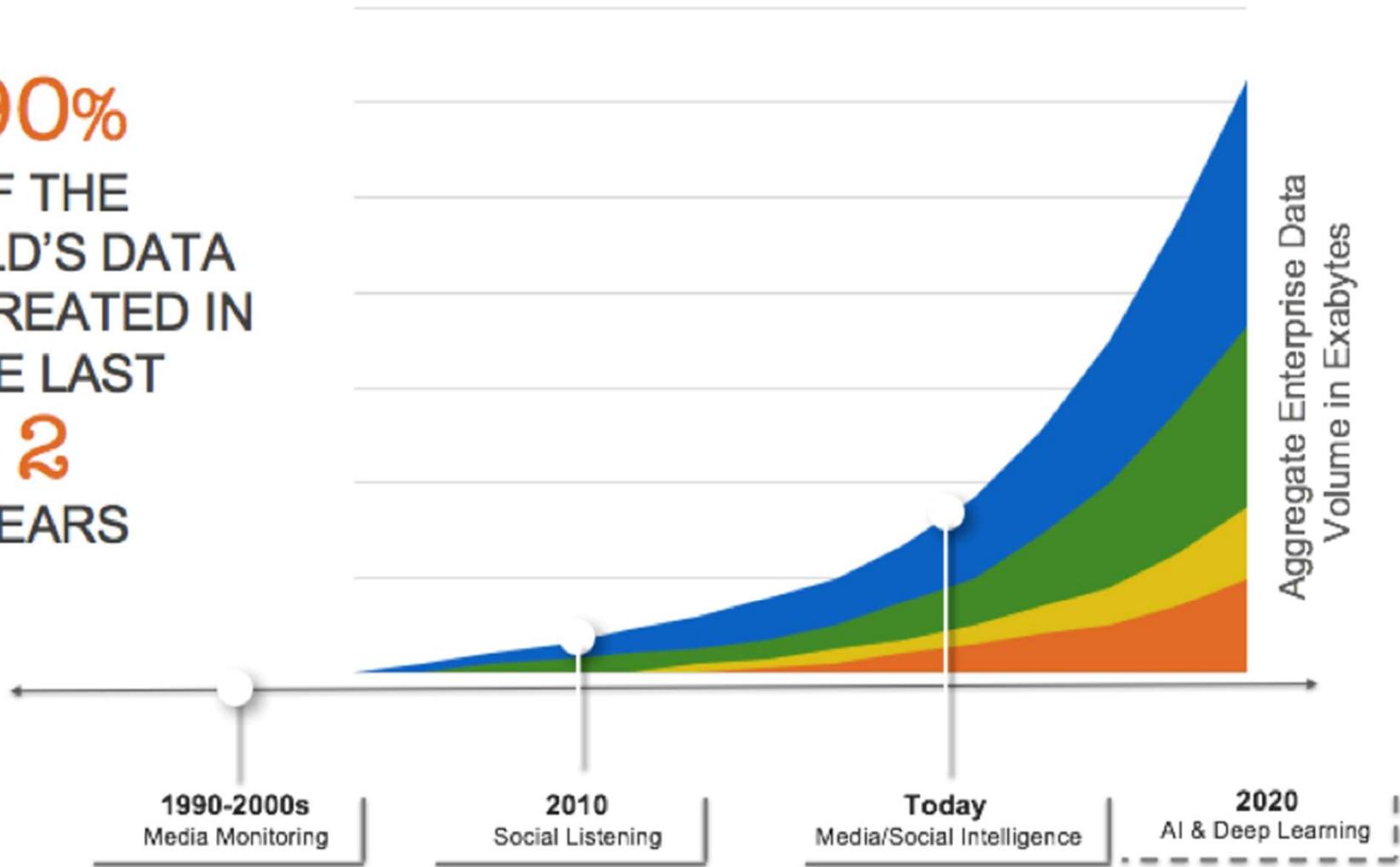
**Rocket Engine : NVIDIA + Deep Learning Algorithm**

**Fuel : Data (25,000 pictures for cat)**

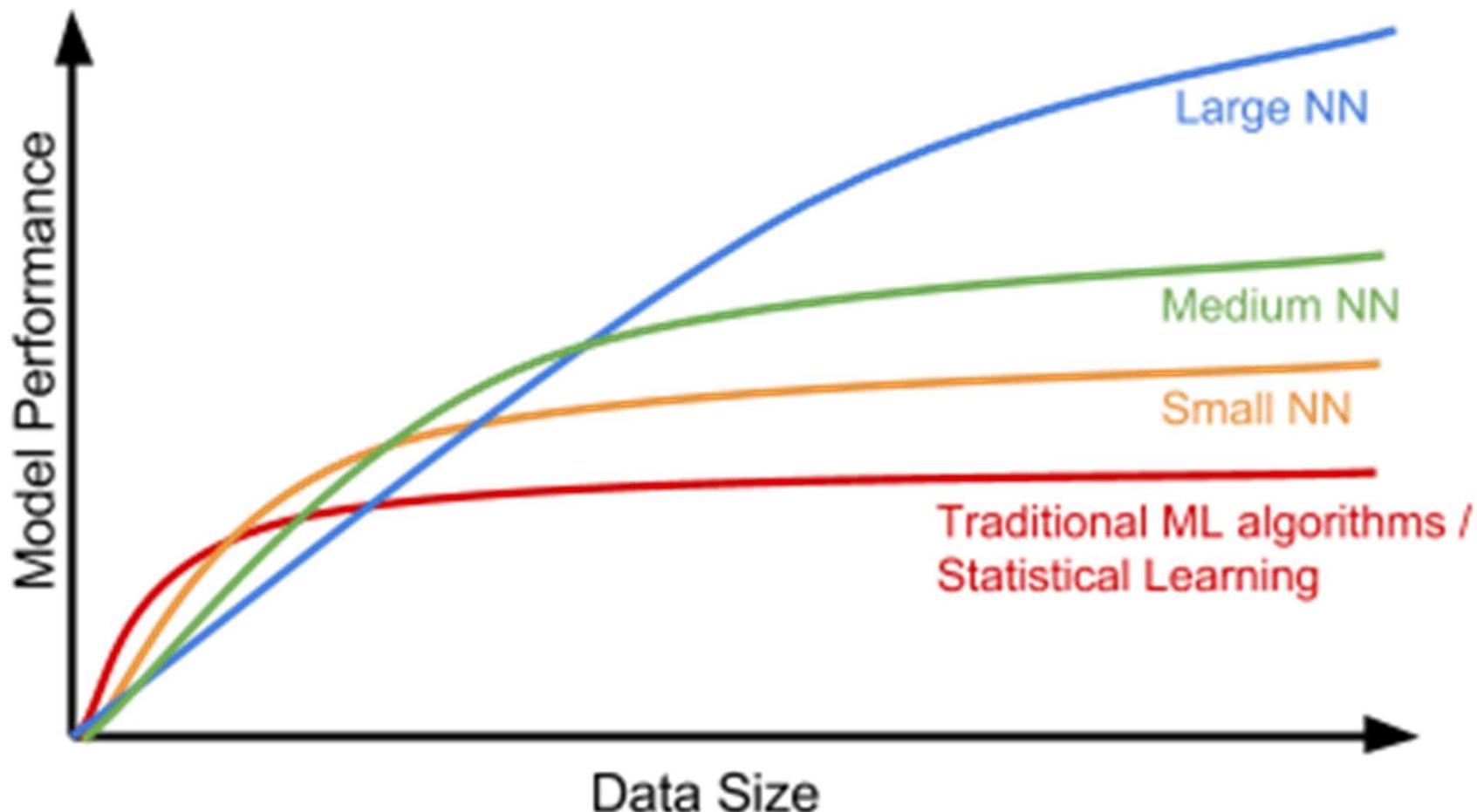
# CPU vs GPU Performance



90%  
OF THE  
WORLD'S DATA  
WAS CREATED IN  
THE LAST  
2  
YEARS



# 전통적 Machine Learning vs. Deep Neural Network

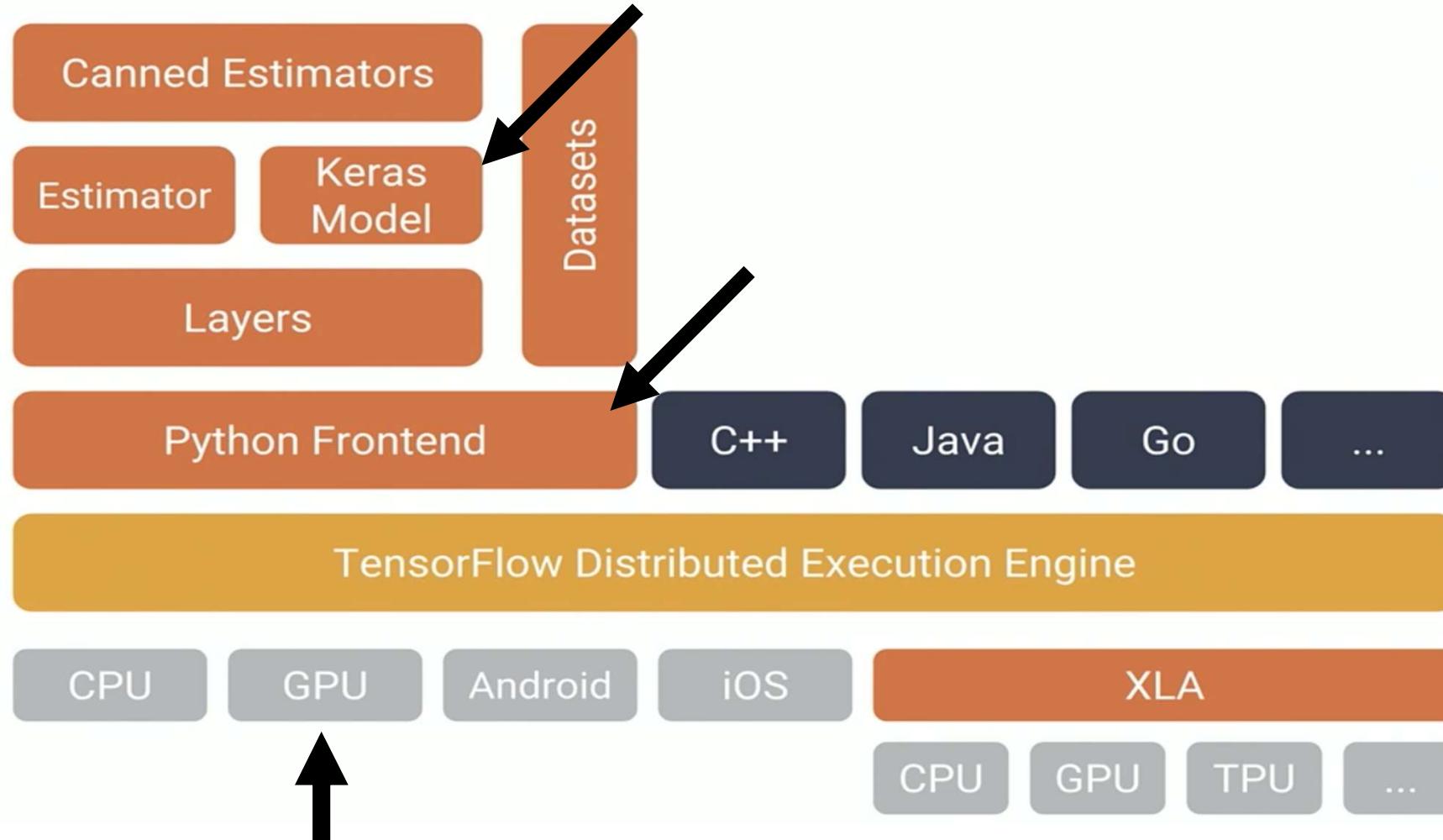




# TensorFlow

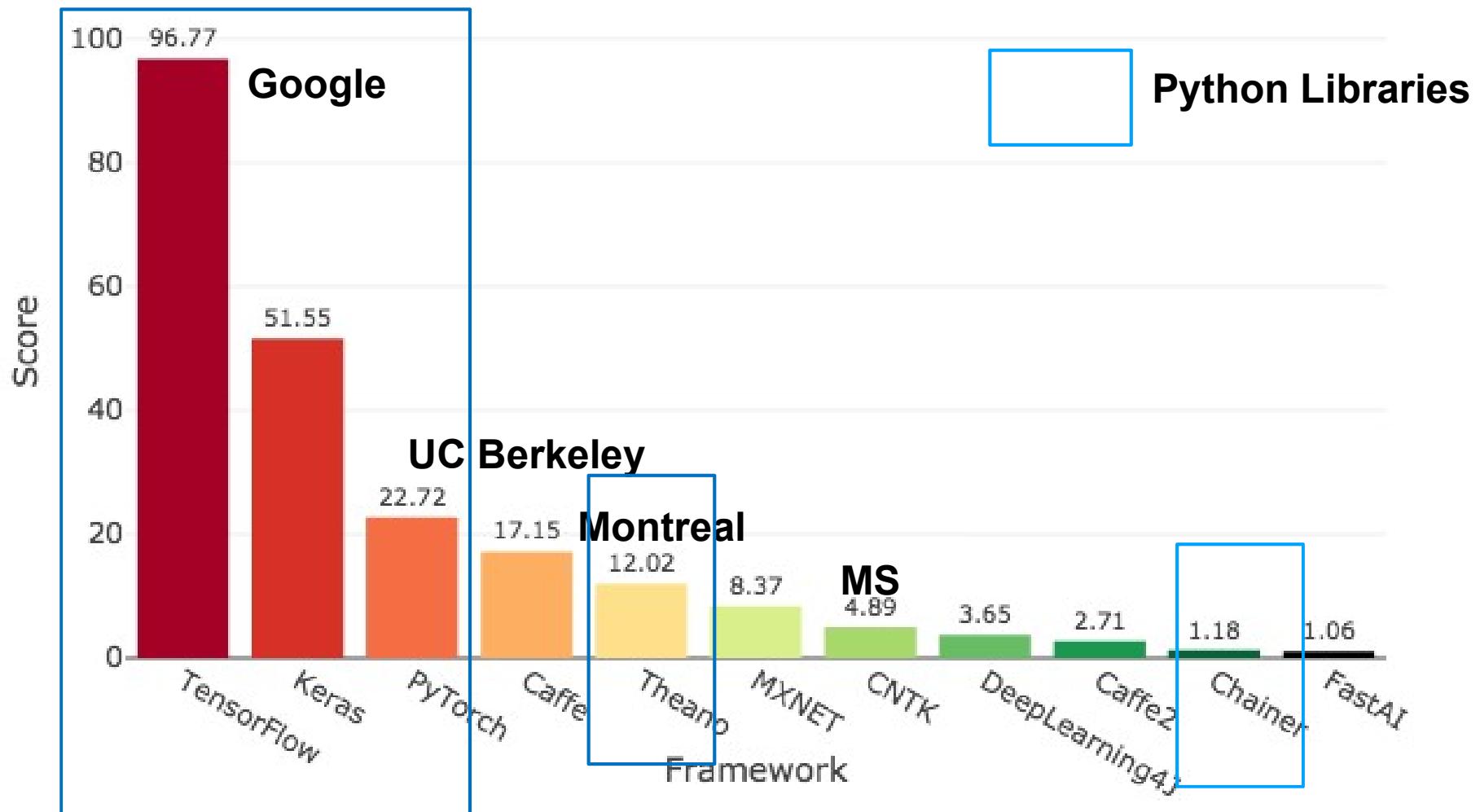
Google open source (2015년 11 월)

Version 2.1 : Keras 공식 포함

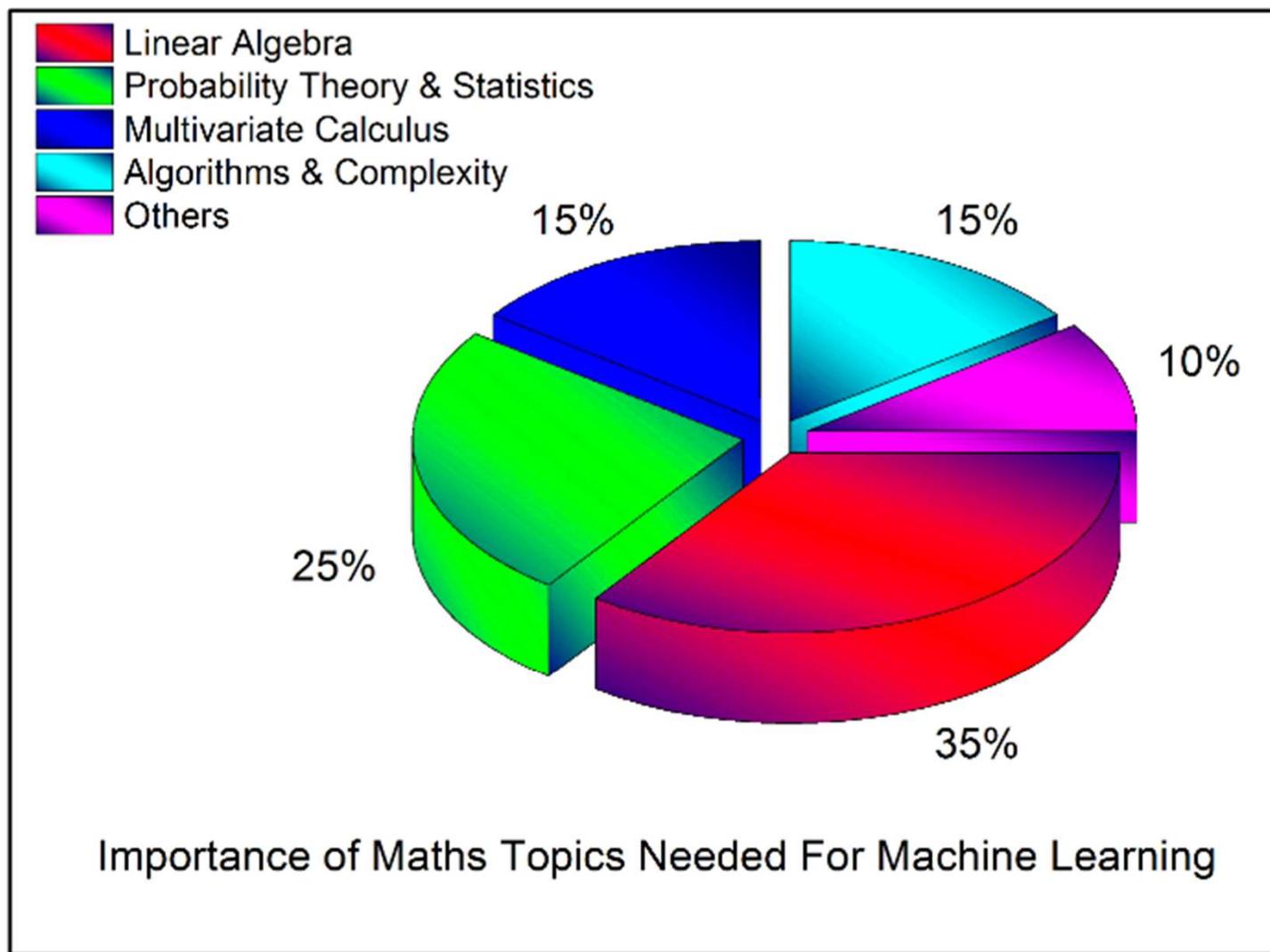


# Kaggle 사용 순위

Deep Learning Framework Power Scores 2018



# Machine Learning 학습에 필요한 수학 지식



# Linear Algebra (선형대수학)

We have

$$a_1 = f(W_{11}x_1 + W_{12}x_2 + W_{13}x_3 + b_1)$$

$$a_2 = f(W_{21}x_1 + W_{22}x_2 + W_{23}x_3 + b_2)$$

etc.

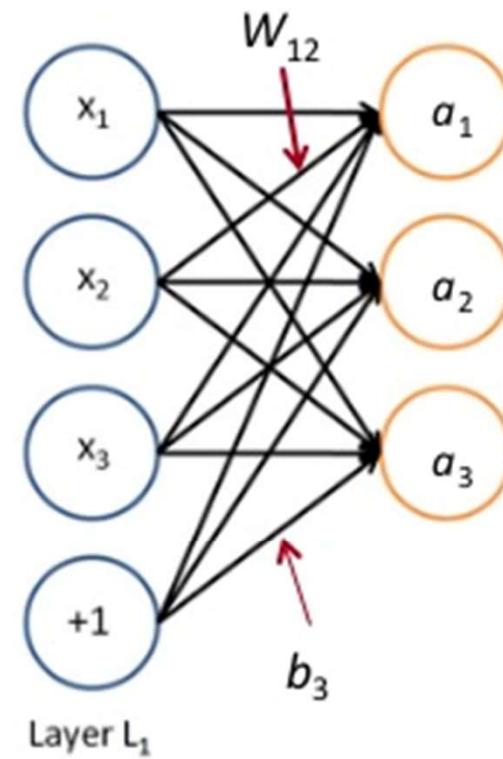
In matrix notation

$$z = Wx + b$$

$$a = f(z)$$

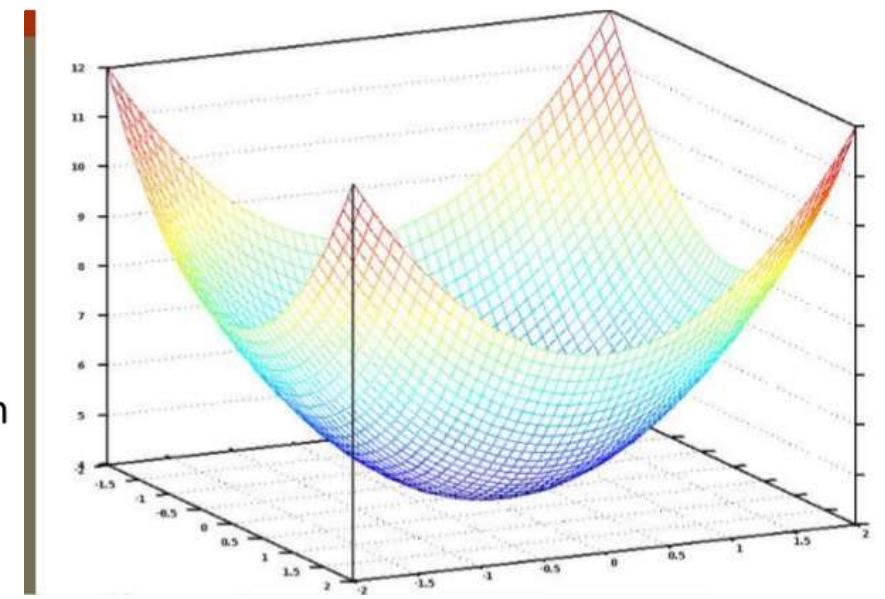
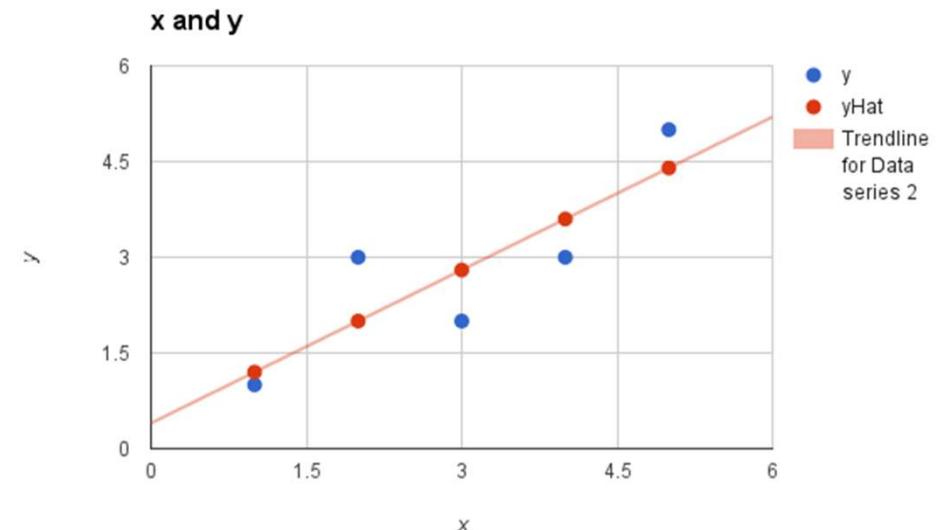
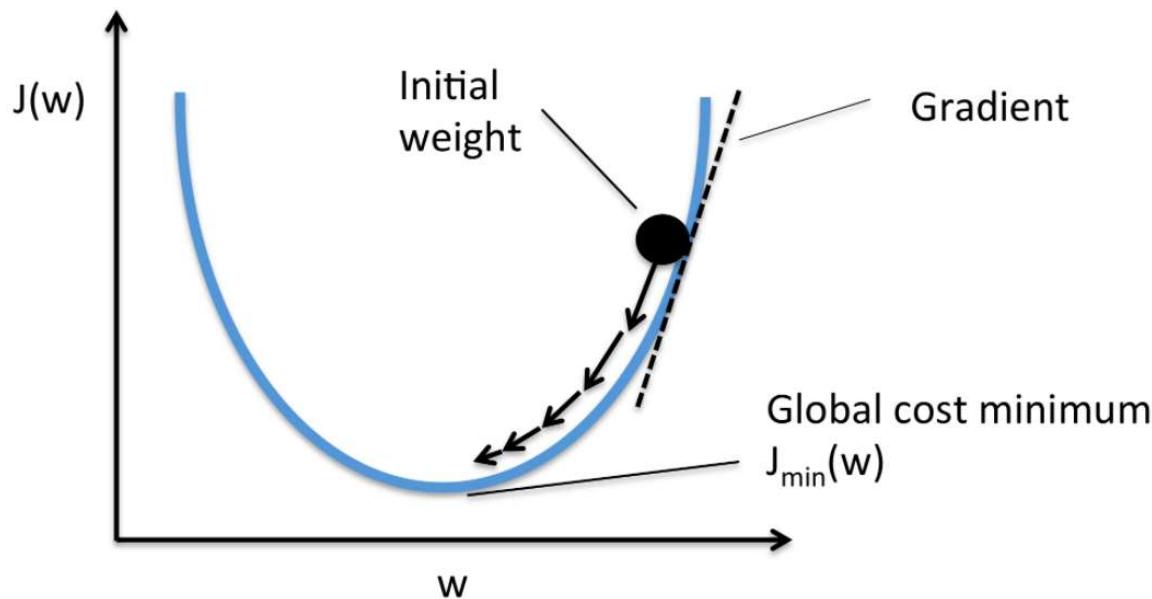
where  $f$  is applied element-wise:

$$f([z_1, z_2, z_3]) = [f(z_1), f(z_2), f(z_3)]$$

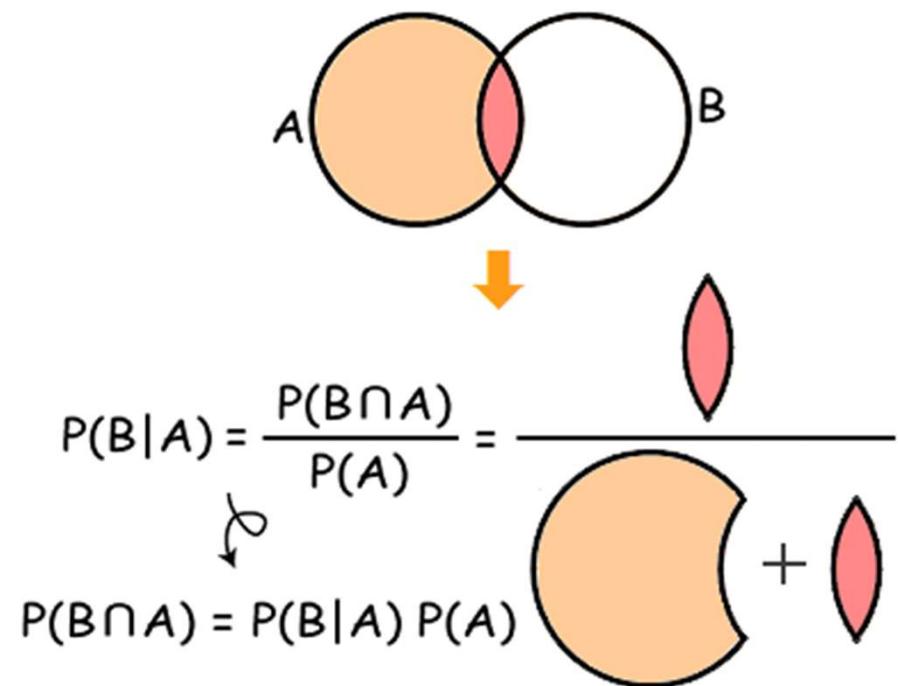
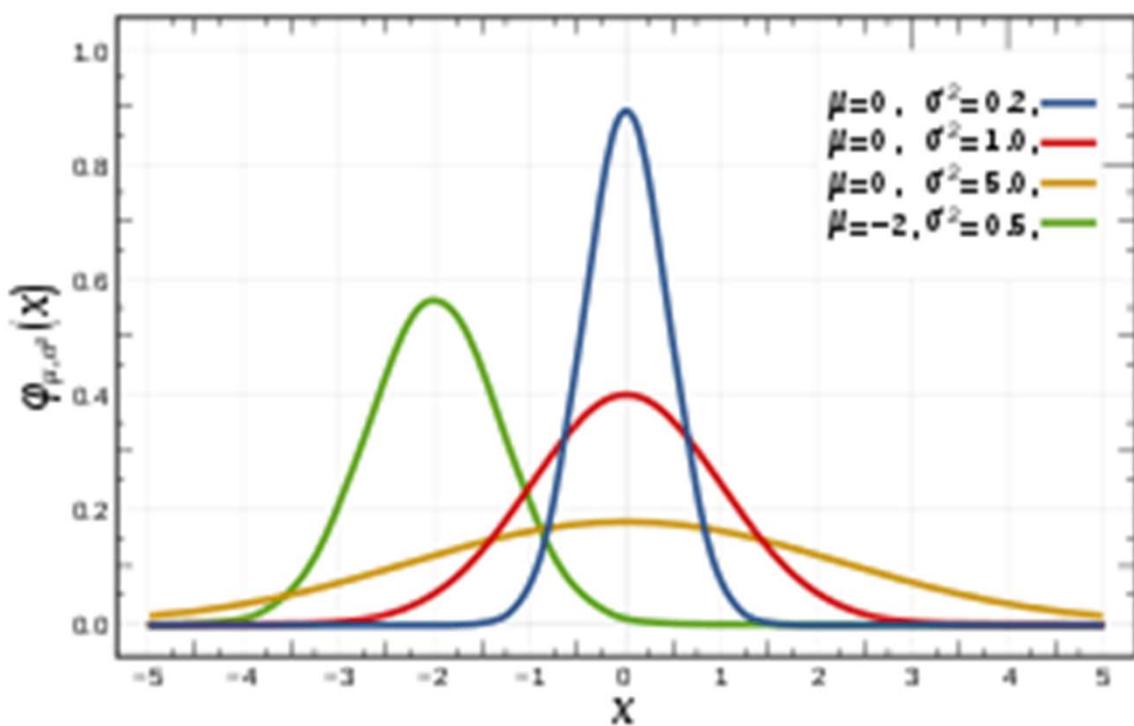


# Multivariate Calculus (다변수미분)

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (\hat{Y}_i - Y_i)^2$$



# Probability/Statistics (확률/통계)



# Python Libraries – ML 관련



IP[y]: IPython  
Interactive Computing



## 1. Scientifics Computing Libraries



### Pandas

(Data structures & tools)



### NumPy

(Arrays & matrices)



### SciPy

(Integrals, solving differential equations, optimization)

## 2. Visualization Libraries



### **Matplotlib**

(plots & graphs, most popular)

### **Seaborn**

(plots : heat maps, time series, violin plots)

### 3. Algorithmic libraries



#### **Scikit-learn**

(Machine Learning : regression, classification,... )

#### **Statsmodels**

(Explore data, estimate statistical models, and perform statistical tests.)

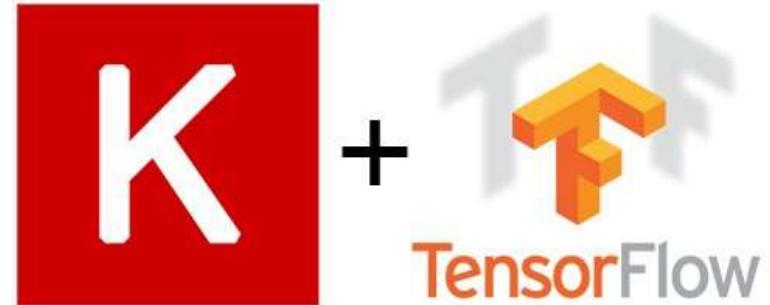
# Machine Learning



전통적 Machine Learning Tool:

- 벤치마크용 데이터셋 예제
- 데이터 전처리(preprocessing)
- 지도 학습(Supervised learning)
- 비지도 학습(Unsupervised learning)
- 모형 평가 및 선택 (evaluation and selection)

# Deep Learning



Deep Learning Tool:

- Pre-train model, Sample Dataset 제공
- Deep Learning 에 필요한 각종 함수 제공
- GPU support
- 각종 language 지원 API 제공

28 37 20<sup>2</sup>  
1 19 10 29 38 11  
12 30  
39 3 21  
22 40 5 23  
31 13 32 4  
24 15  
33 6  
16<sup>25</sup> 17 35 8  
36 9  
27 18  
34 7 26

<sup>28</sup> 1 <u>19</u> <sup>10</sup>	<sup>37</sup> <sup>2</sup> <sup>20</sup> <sub>29</sub> <sup>38</sup> <sub>11</sub>	<sup>12</sup> <sup>39</sup> <sub>3</sub> <sup>30</sup> <sup>21</sup>
<sup>22</sup> <b>31</b> <sup>40</sup> <sup>4</sup> <b>13</b>	<sup>5</sup> <sub>23</sub> <sup>32</sup> <sub>14</sub>	<b>24</b> <sup>15</sup> <sup>33</sup> <sup>6</sup>
<sup>25</sup> <b>16</b> <sub>7</sub> <u><b>34</b></u>	<sup>26</sup> <b>17</b> <b>35</b> <sup>8</sup>	<sup>36</sup> <sup>9</sup> <b>27</b> <b>18</b>

# 과정 SCOPE

- Crash Course – Numpy, Pandas, Matplotlib
- Linear Regression (선형회귀)
- non-Linear Regression (비선형회)
- KNN (K-Nearest Neighbor)
- Decision Tree
- Logistic Regression
- SVM (Support Vector Machine)
- Random Forest

- K-Means Clustering
- Tensorflow 소개
- Keras
- Simple Neural Network
- CNN (Convolutional Neural Network)
- RNN
- 실습문제

# Crash Course

1. Numpy
2. Pandas
3. Matplotlib

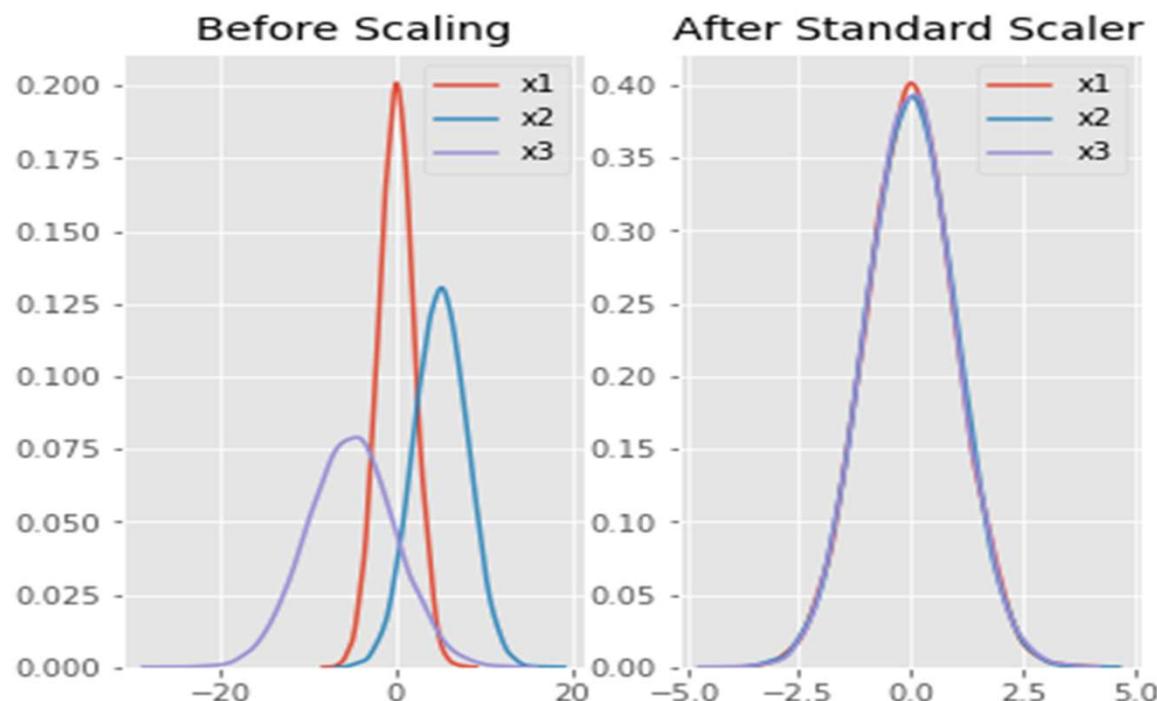
# Feature Scaling

- Raw data 를 전 처리하여 input data 의 구간을 표준화.
- Standard Scaling :

$$z = (x - \mu) / s \quad (\mu : \text{평균}, s : \text{표준편차})$$

- Minmax Scaling :

$$x_{\text{new}} = \frac{x_i - \min(X)}{\max(x) - \min(X)}$$



# 실습: sklearn 을 이용한 scaling

- Standard Scaling
- MinMax Scaling