

# 코딩은 처음이라 with 딥러닝

---

# Contents

- 
- 1장 인공지능이란**
  - 2장 딥러닝 흐름잡기**
  - 3장 기본 흐름 파악하기**
  - 4장 선형모델**
  - 5장 신경망 모델**
  - 6장 이미지 분류**
  - 7장 자연어 처리**
  - 8장 Project**
- 



# Contents



---

## 1장 인공지능이란

1.1. 인공지능: 데이터 분석을  
위한 알고리즘

1.2. 머신러닝과 딥러닝

1.3. 머신러닝 과정

1.4. kaggle 소개

---

# 인공지능이란?

## 사전

판단, 추론, 학습 등 **인간의 지능**이 가지는 기능을 갖춘  
컴퓨터 시스템

## 현실

주어진 데이터로  
주어진 모델을  
최적화하여  
새로운 데이터에 대한  
결과값을 **예측**하는 프로그램

# \*프로그램

## 시스템

외부로부터의 힘에 의해 동작하는 일련의 자동 기계 장치



## 프로그램

[전산] 어떤 문제를 해결하도록 컴퓨터에 주어지는 자료 처리 방법과 순서를 기술한 일련의 명령문의 집합체

```
1 import numpy as np
2 def softmax(a) :
3     exp_a=np.exp(a)
4     sum_a=np.sum(exp_a)
5     return exp_a/sum_a
```

# 인공지능 유형

1980년에 미국의 분석철학자인 존 설(John Rogers Searle)의 분류

강인공지능 : 인간의 마음을 복잡한 정보처리로 구현한 것

약인공지능 : 단순히 인간의 능력 일부를 시뮬레이션하거나 그런 작업을 목적으로 하는 것

**강인공지능(Strong AI)** 범용인공지능(AGI: Artificial General Intelligence)

사람처럼 지각하고 학습하고 추론하는 지능을 갖춘 컴퓨터 시스템

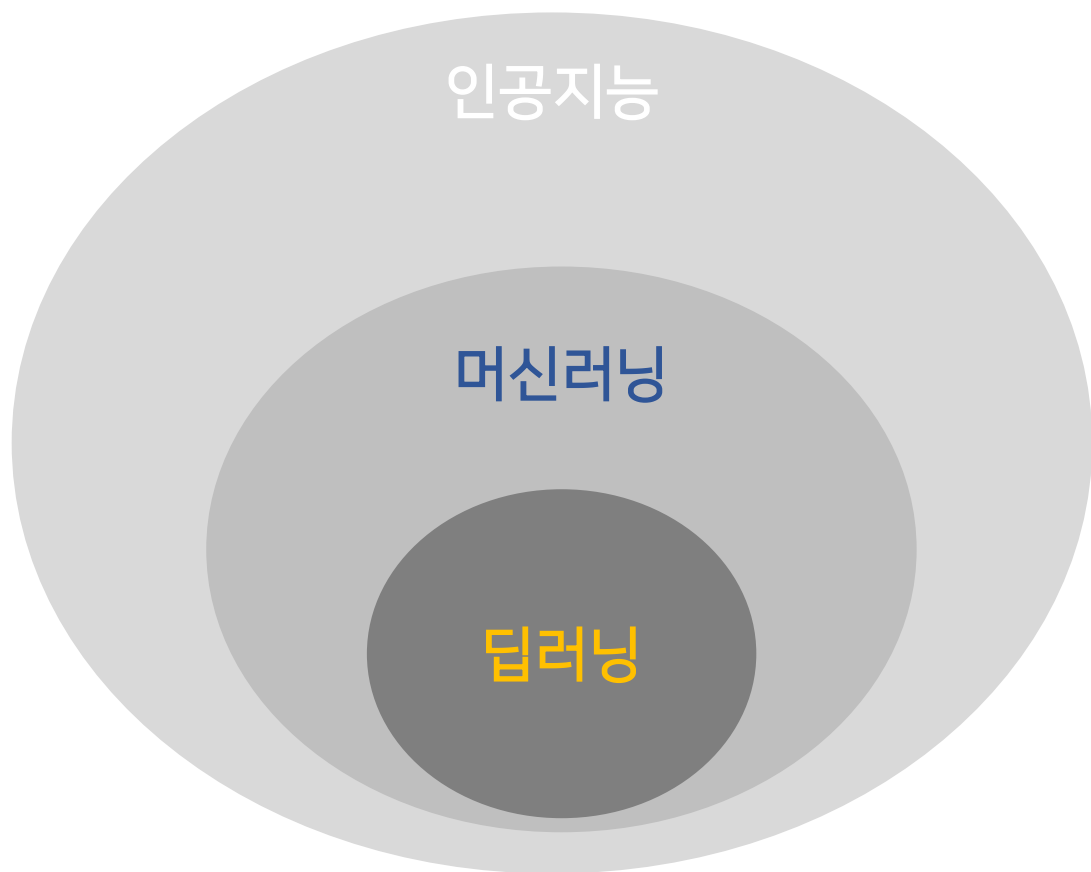
1956년 다트머스 회의에서 존 매카시(John McCarthy)가 제안한 인공지능 개념

**약인공지능(Weak AI)** ← 우리가 다룰 수 있는/다루는 인공지능의 의미

인간의 지능으로만 할 수 있던 일을 컴퓨터 작업으로 수행하는 것

지능이 있는 컴퓨터 시스템이 아닌 유용한 도구로서의 의미

# 인공지능 머신러닝 딥러닝



## 컴퓨터가 생각할 수 있는가?

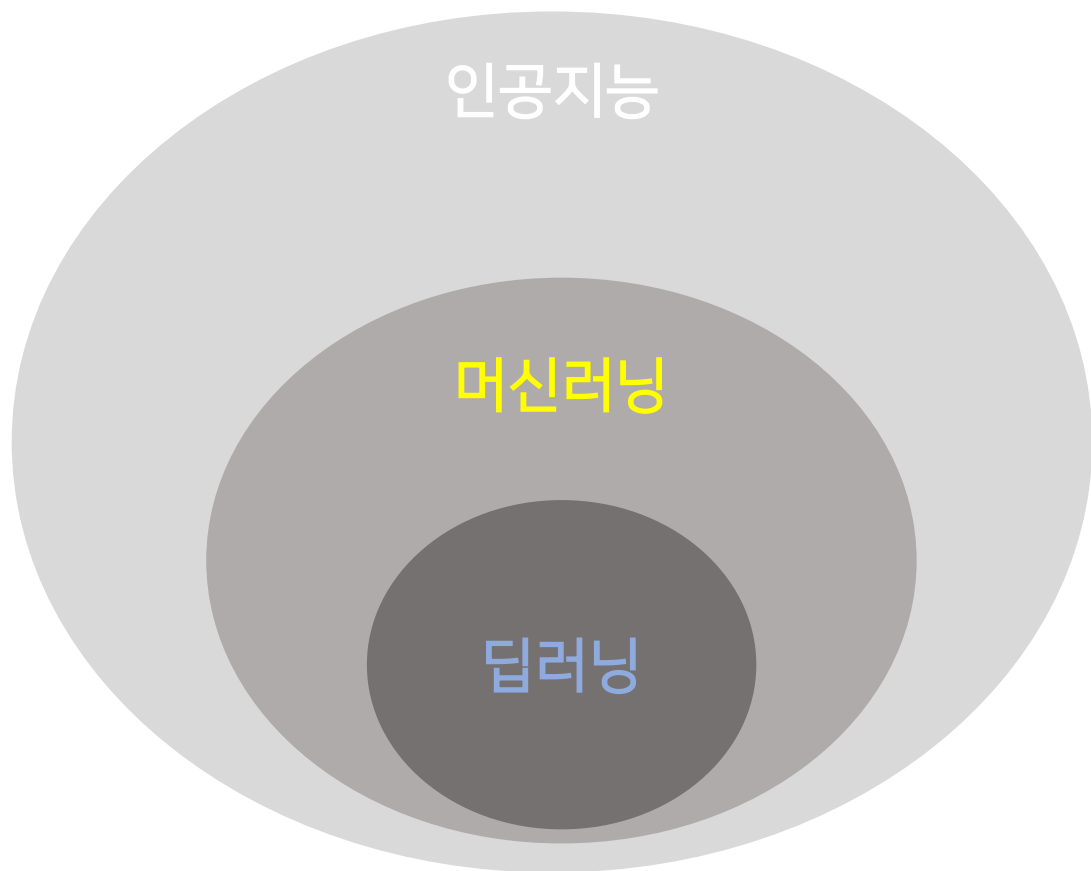
보통의 사람이 수행하는 지능적인 작업을 자동화하기 위한 연구 활동

컴퓨터가 방대한 데이터를 바탕으로 학습 모델 생성  
데이터 → 학습 → 예측

## 연속된 층을 사용한 기법

- 신경망을 기반으로 다층 구조를 통해 학습
- 정형화된 데이터를 받지 않고 필요한 데이터를 수집 처리
- 데이터로부터 표현을 학습하는 방식

# 무엇을 배우나



머신러닝

회귀 (regression)  
분류 (classification)

딥러닝

Computer Vision : CNN  
자연어 처리 : RNN, LSTM

실습

kaggle.com  
Scikit-Learn / Keras





# Contents

## 1장 인공지능이란

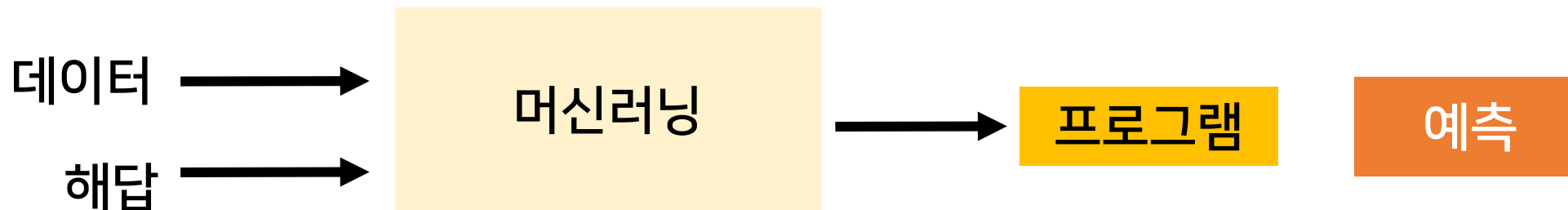
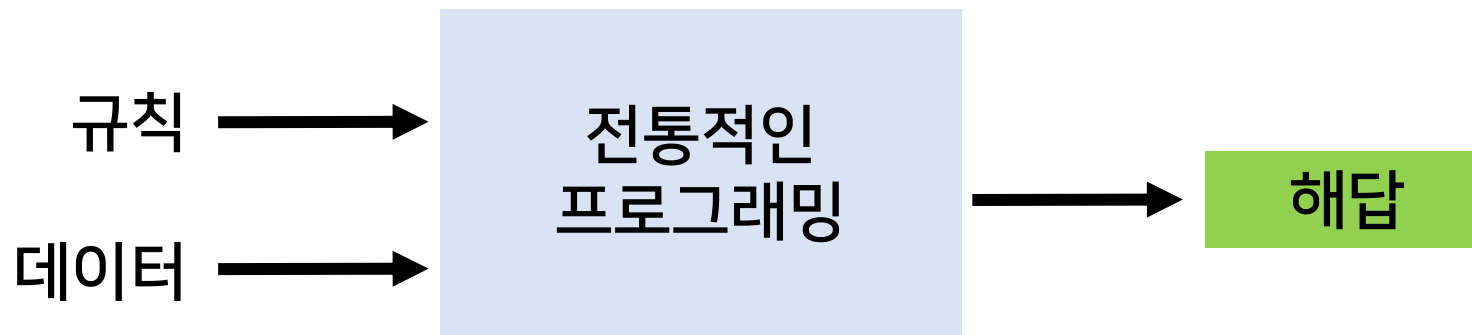
1.1. 인공지능: 데이터 분석을  
위한 알고리즘

1.2. 머신러닝과 딥러닝

1.3. 머신러닝 과정

1.4. kaggle 소개

# 머신 러닝이란



# 머신 러닝 : 딥러닝



## 머신러닝

구조적인 데이터

하나의 층

Ensemble, Boost

Sklearn

데이터

층

모델 기법

프레임워크

## 딥러닝

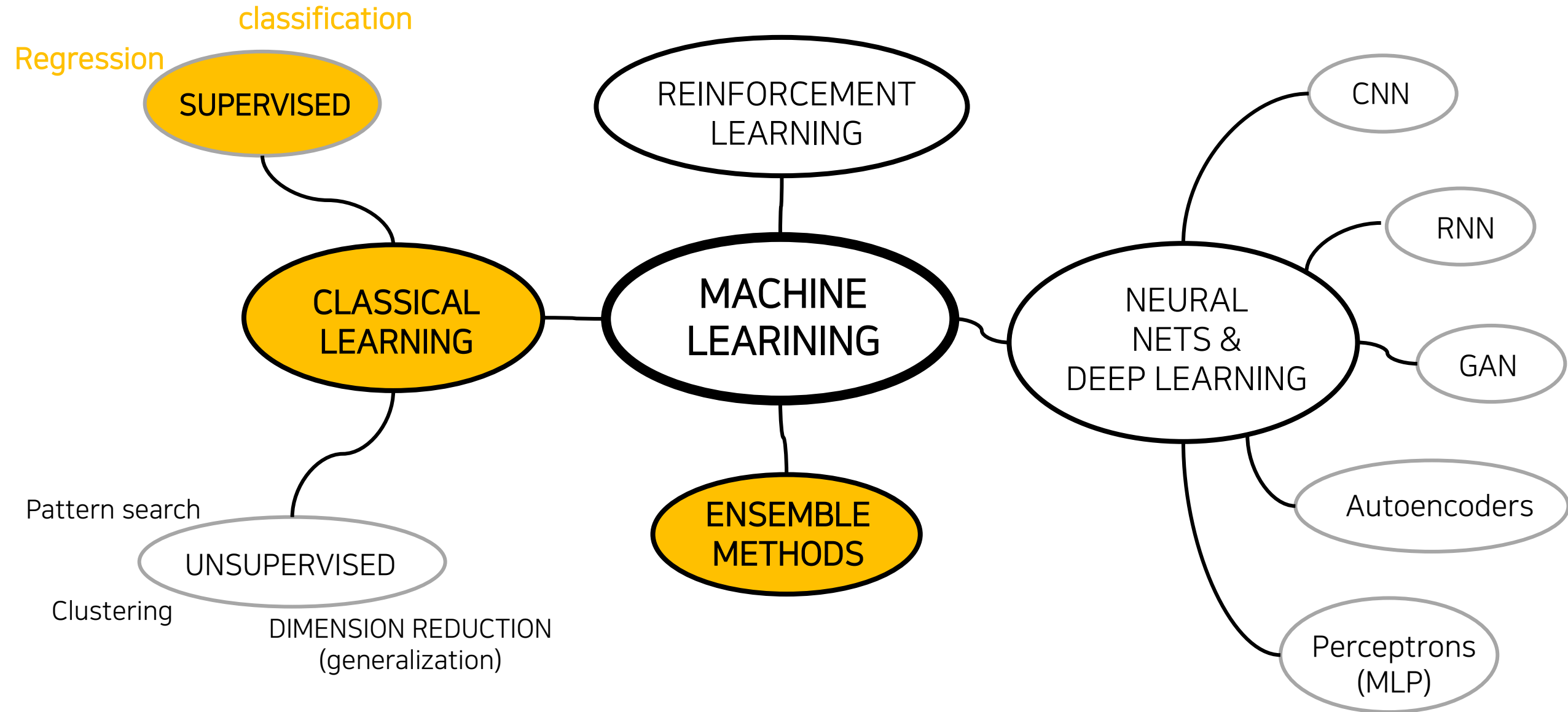
지각에 관한 데이터  
(이미지, 자연어, 시계열 등)

여러 개의 층

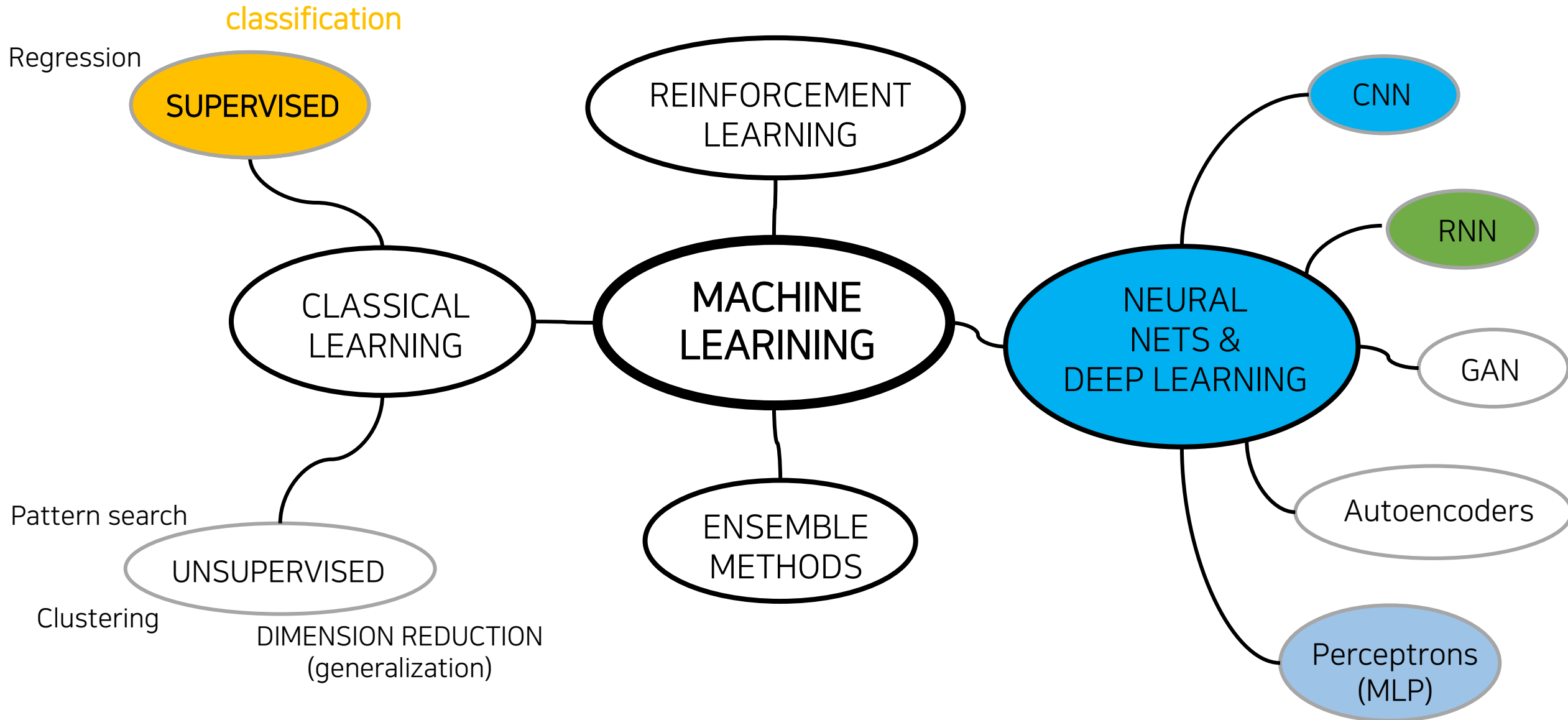
신경망, CNN, RNN

**tensorflow, keras, pytorch**

# 머신러닝



# 딥러닝



# Contents

## 1장 인공지능이란

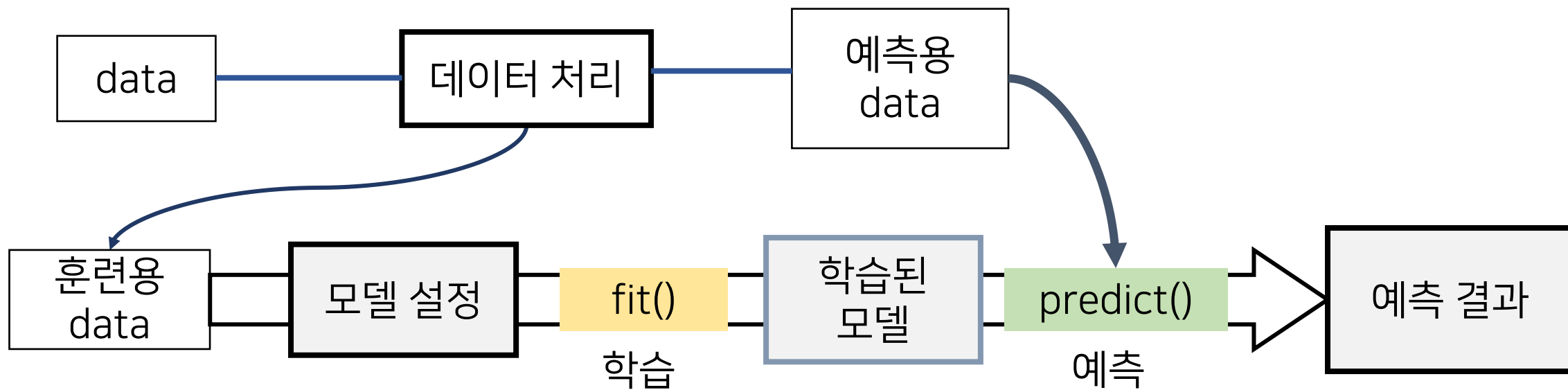
1.1. 인공지능: 데이터 분석을  
위한 알고리즘

1.2. 머신러닝과 딥러닝

1.3. 머신러닝 과정

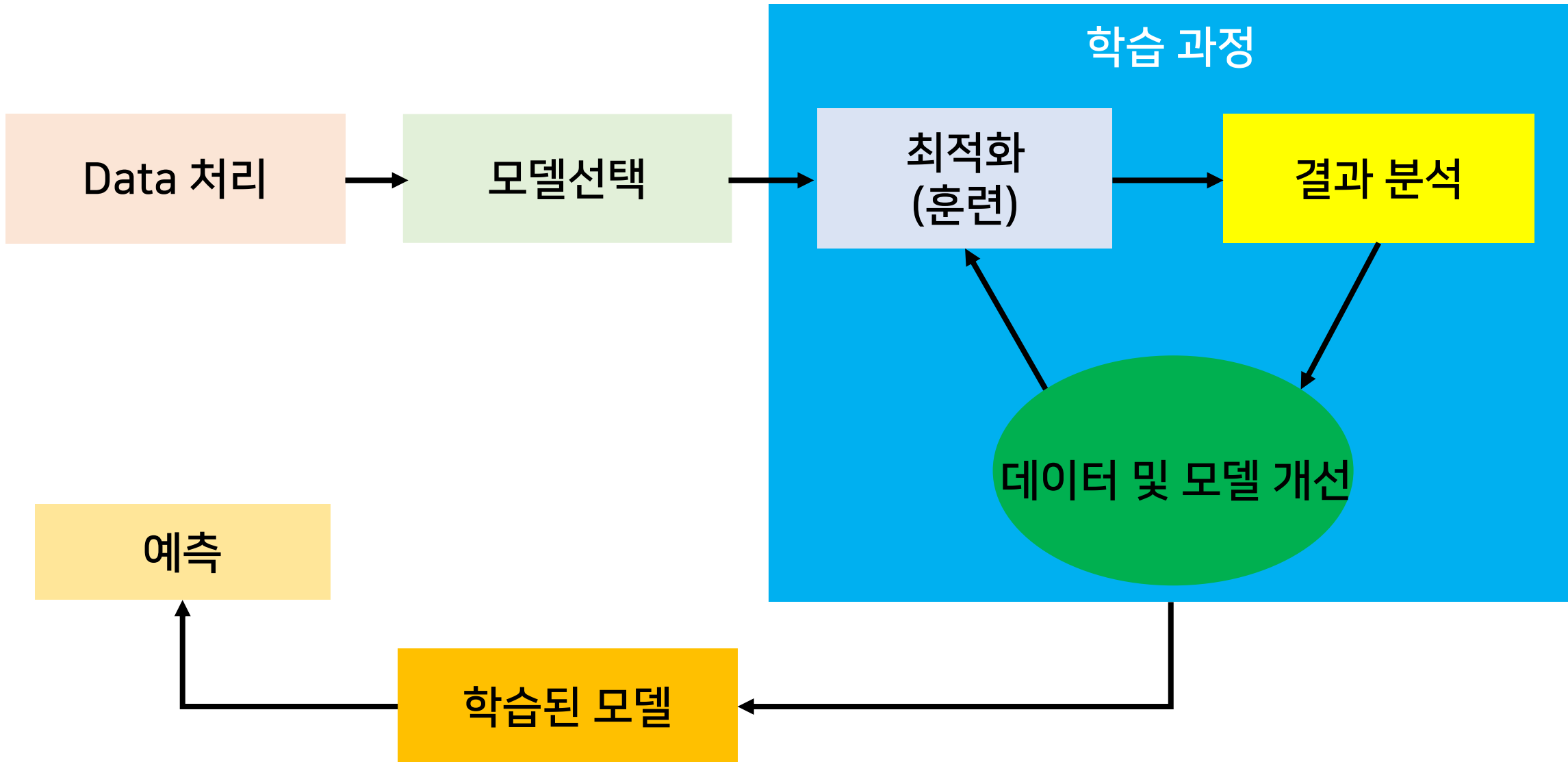
1.4. kaggle 소개

# 학습 흐름





# 머신 러닝 개요



# CIFAR10 데이터 집합

airplane



automobile



bird



cat



deer



dog



frog



horse



ship



truck



Data

32x32x3 정수 데이터(0~255)

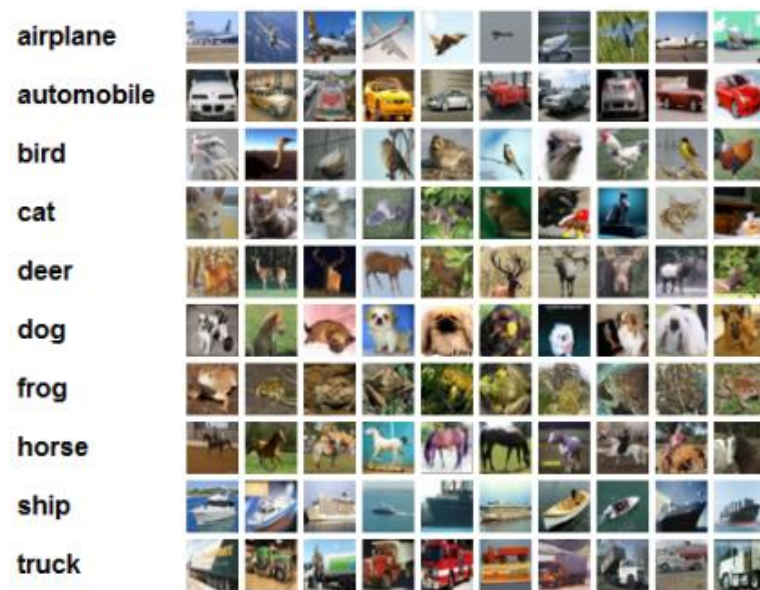
6만개

Label

0: dog, 1: cat, 2: truck, ...

10 가지

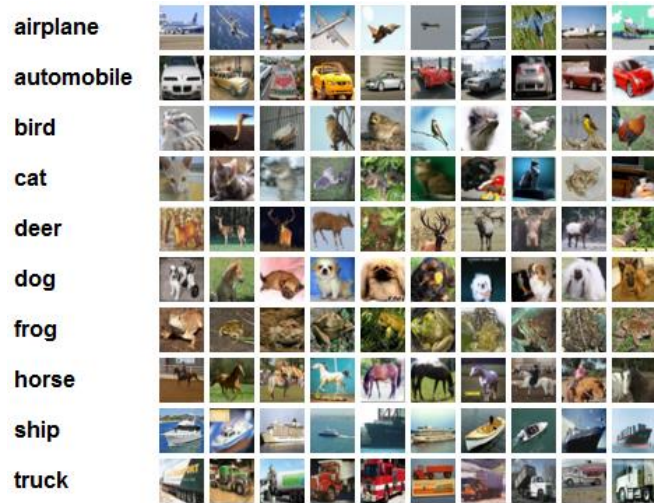
# Supervised Learning



데이터+레이블

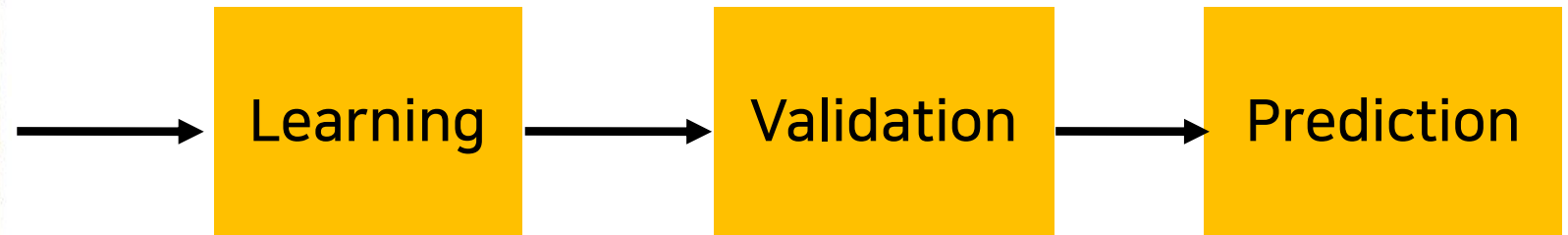


# Supervised Learning



(Data, Label)

데이터에 대한 레이블  
(Label)[target;정답]이 주어진 상  
태에서 학습



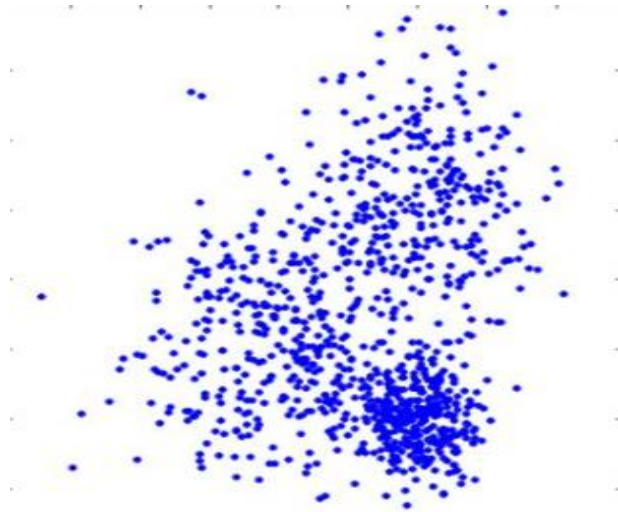
regression :  
예측 결과값이 continuous value

XGBoost  
Light GBM

classification :  
예측 결과값이 discrete value

CNN  
RNN

# Unsupervised Learning



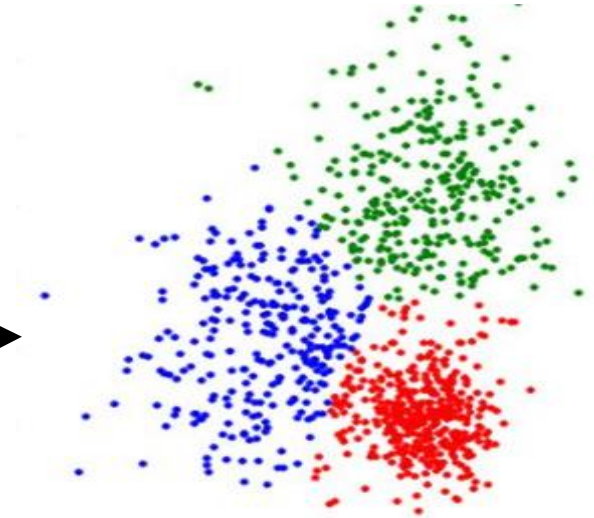
데이터



학습



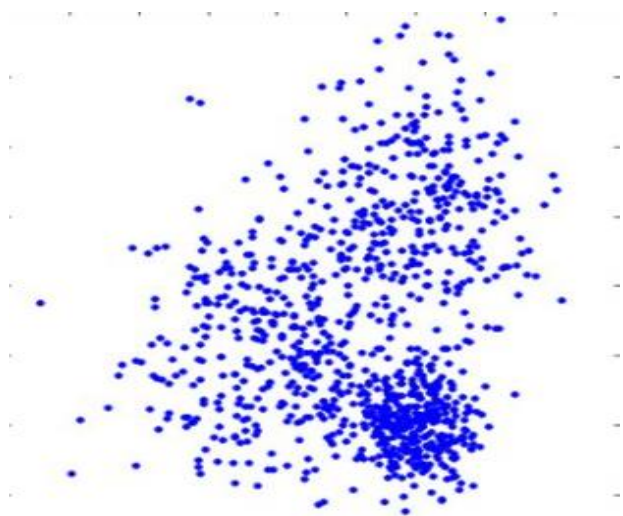
구조 파악



결과



# Unsupervised Learning



Data

데이터에 대한 Label (정답)이 주어지지 않은 상태에서 학습

데이터의 숨겨진 특징(Hidden Feature)이나 구조(Structure)를 발견하는데 사용

Learning

Finding  
Structure

## Clustering

서로 비슷해 보이는 학습 데이터를 찾아 그룹으로 묶음

## Association

데이터 샘플의 어떤 특성을 다른 특성과 연관 짓는 것

## Anomaly detection

데이터 세트에 특정 값을 표시하는데 사용

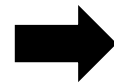
## Autoencoders

입력 데이터를 가지고 하나의 코드로 압축한 뒤 요약된 코드로부터 입력 데이터를 재생성

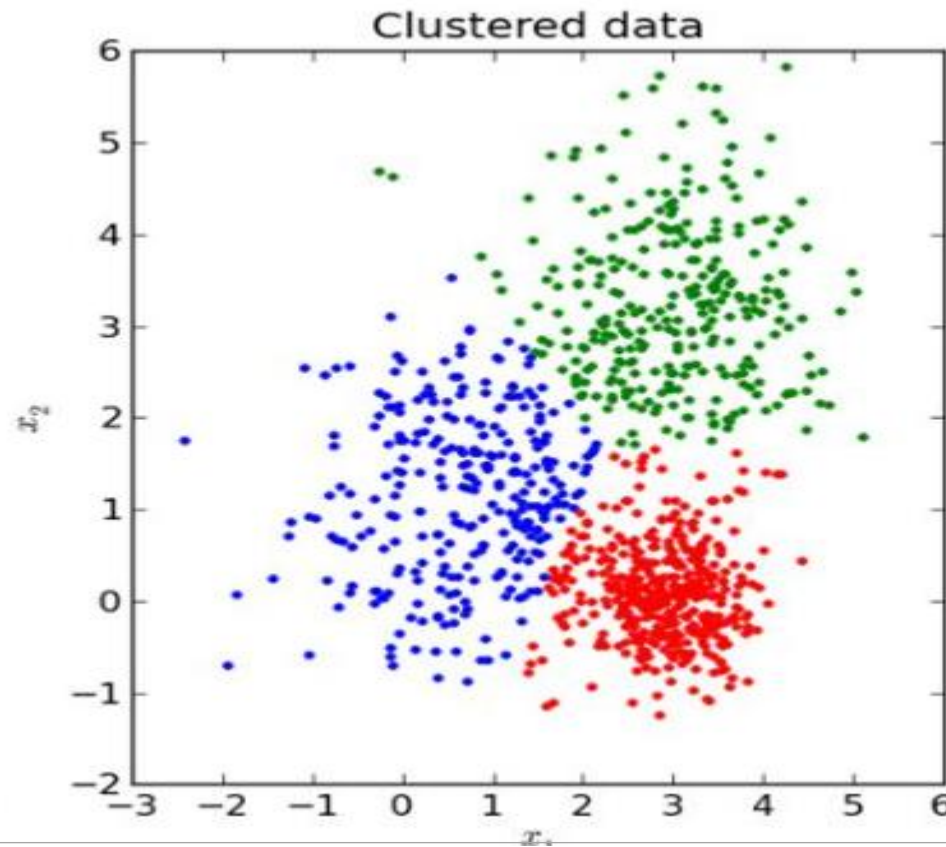
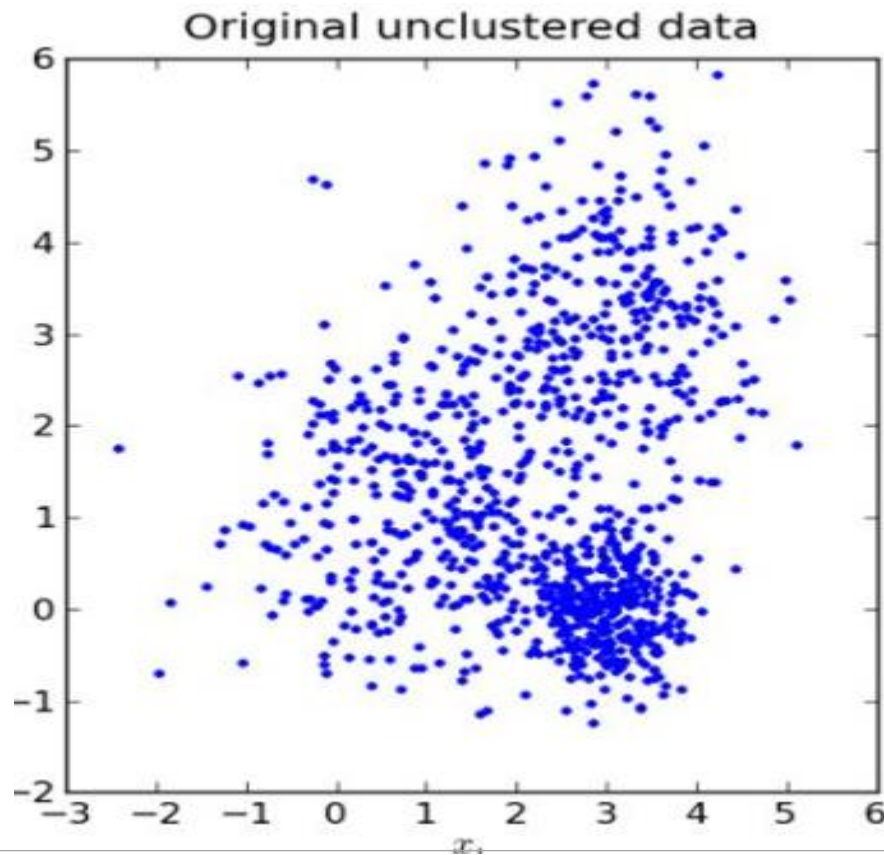
# 클러스터링(Clustering) 알고리즘



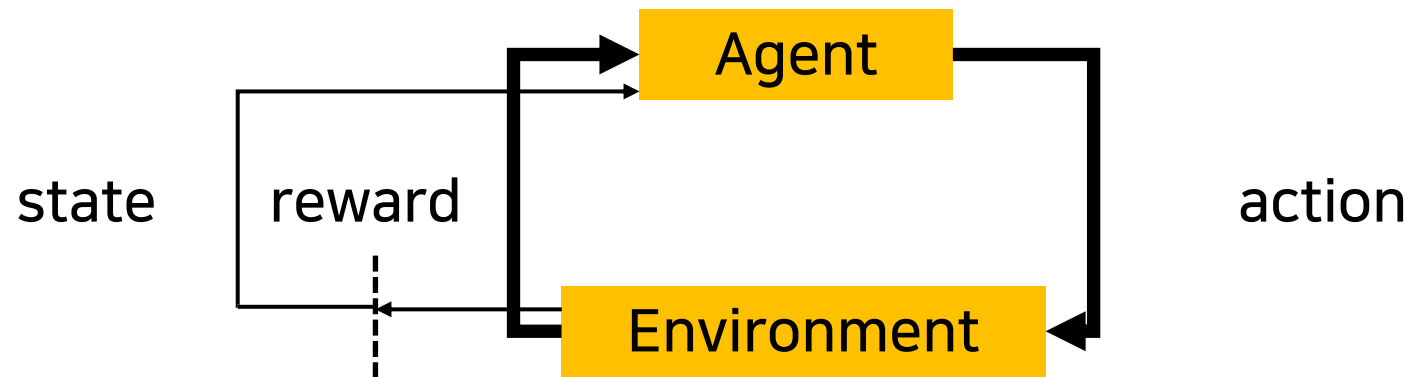
데이터가 무작위로 분포



비슷한 특성으로 묶음

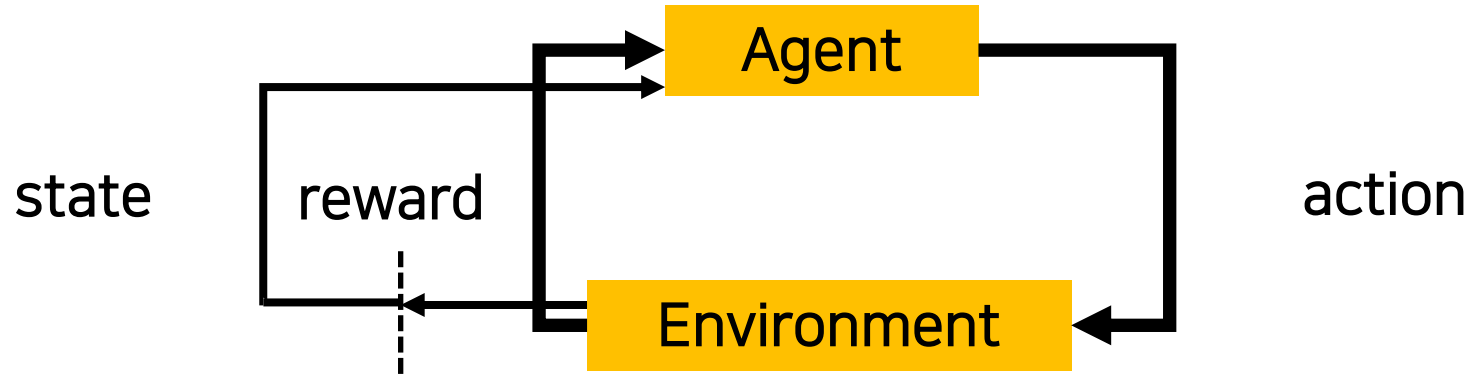


# Reinforcement Learning





# Reinforcement Learning



Agent : 상태를 관찰, 행동을 선택, 목표지향

Environment : 에이전트를 제외한 나머지

State : 현재 상황을 나타내는 정보

Action : 현재 상황에서 에이전트의 행위

Reward : 행위의 결과로 주어지는 보상

Agent : 주어진 환경(state)에 대해 행동(action)을 취하고 이로부터 보상(reward) 획득

Agent는 **보상(reward)을 최대화(maximize)**하도록 학습이 진행

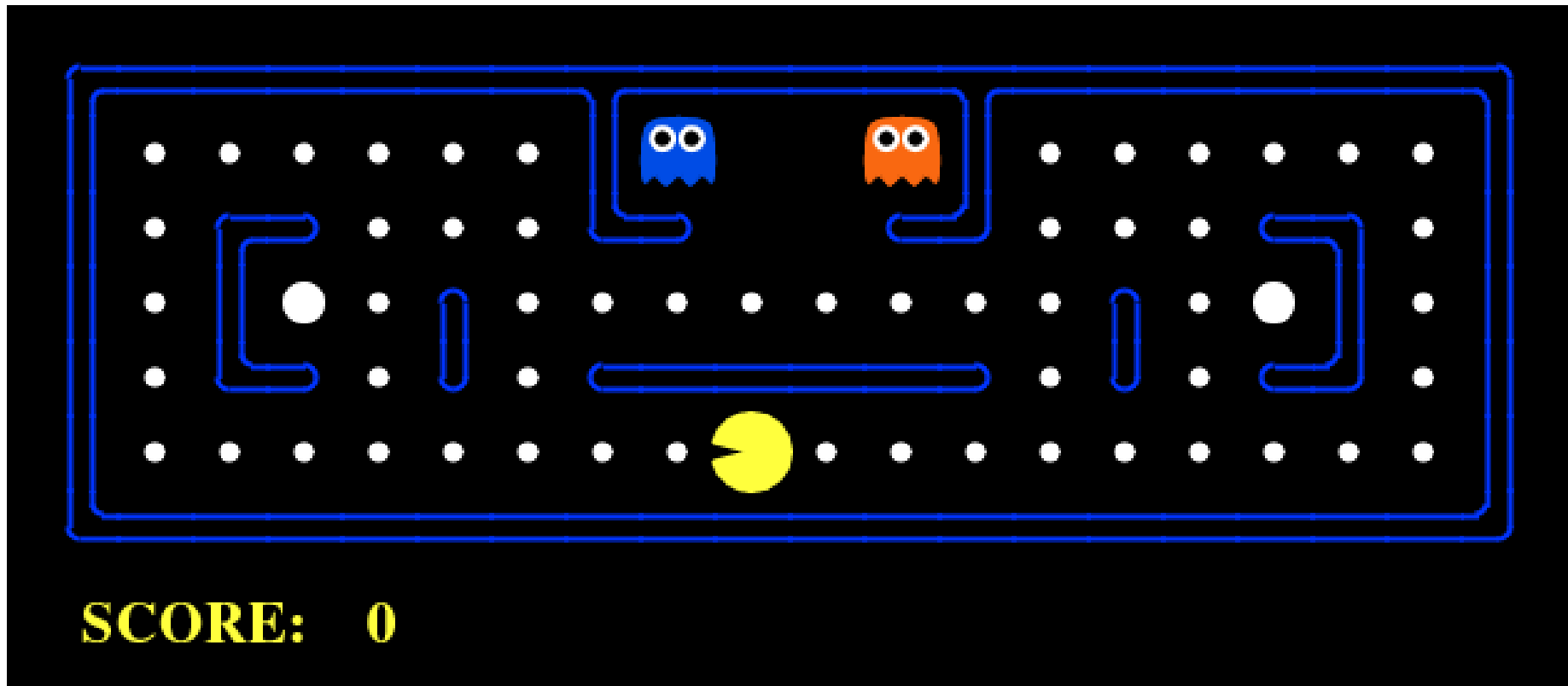
동적인 상태(dynamic environment)에서 데이터를 수집하는 과정까지 포함되어 있는 알고리즘

Alpha Go

DQN

# Reinforcement Learning

PacMan에 적용된 강화 학습의 예



<https://ichi.pro/ko/post/1612547702598>

# 학습의 종류

## Supervised

Label된 Dataset  
즉각적인 feedback  
최적화 학습

결과 예측

## Unsupervised

Label되지 않은 Dataset  
Feedback 없음

Hidden structure 발견

## Reinforcement

Markov 결정과정  
보상 시스템  
행위의 반복으로 학습

로봇 제어, 게임

# 이미지 데이터셋 : 이미지데이터+레이블



분류 레이블이 주어진 이미지 데이터셋

3채널

Data

1280x800x3 정수 데이터(0~255)

Label

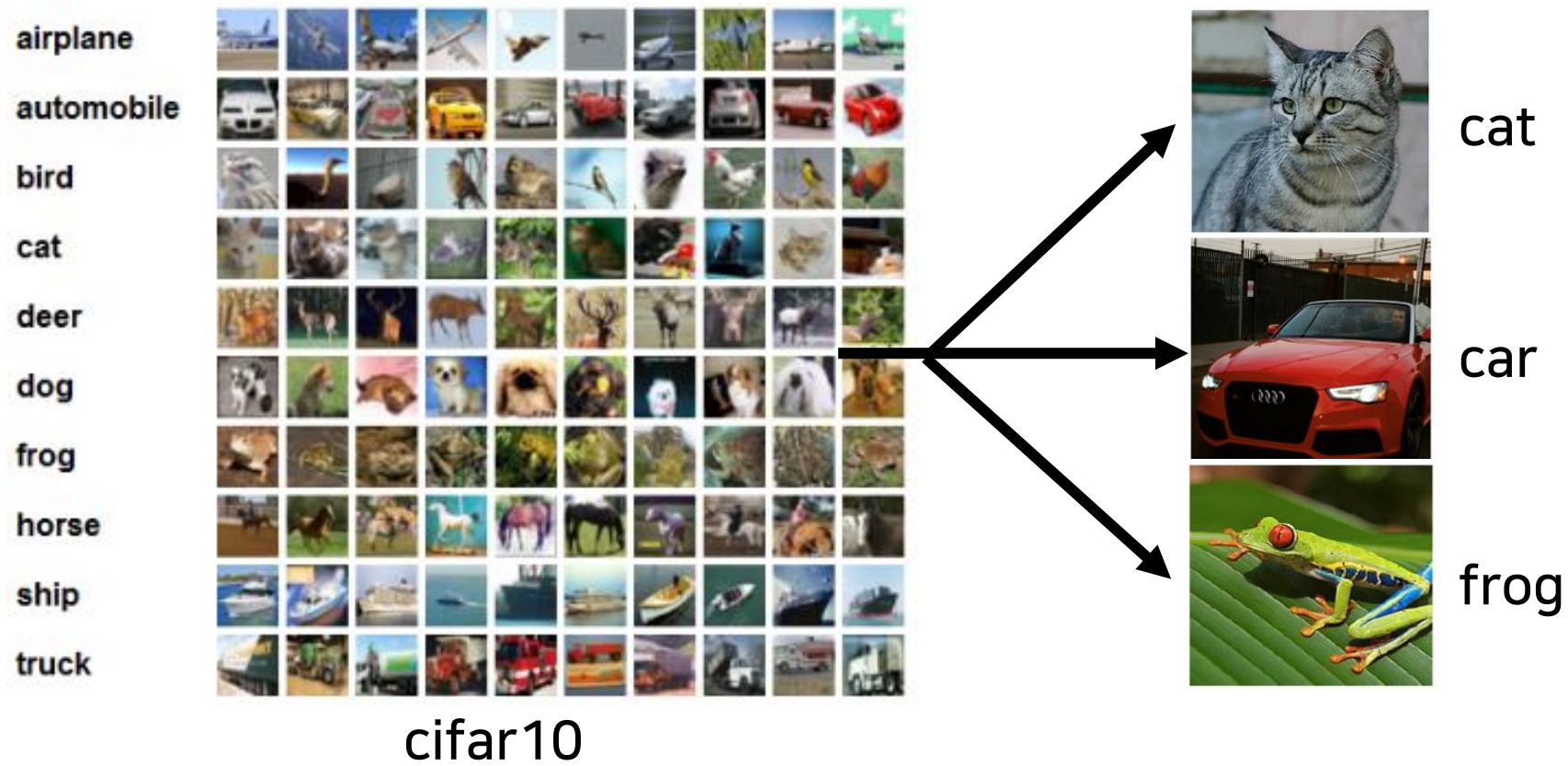
0: dog, 1: cat, 2: truck, 3: plane, ...

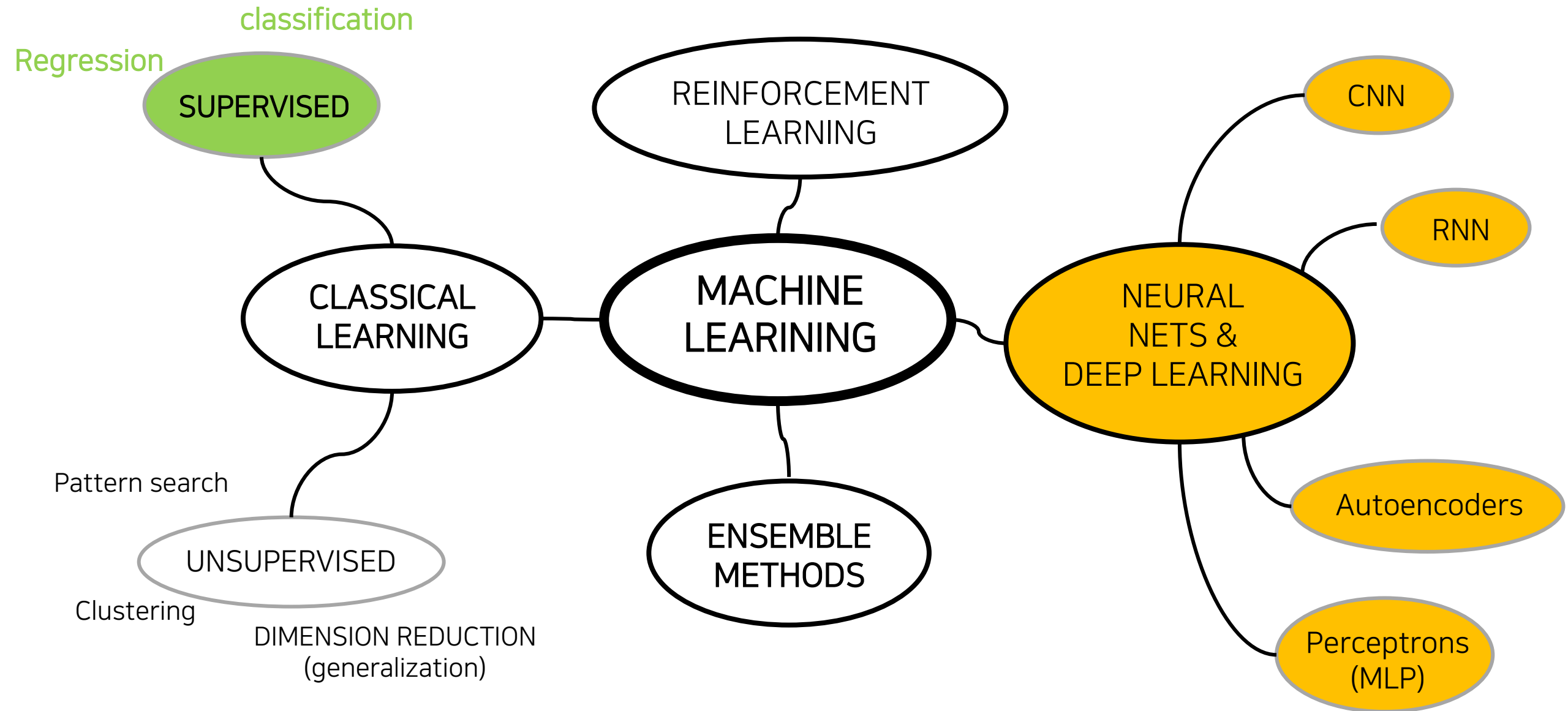
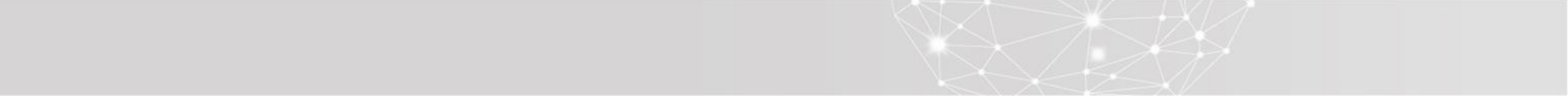
➡ 1 ➡ Cat

**이미지 분류 [Image Classification]**

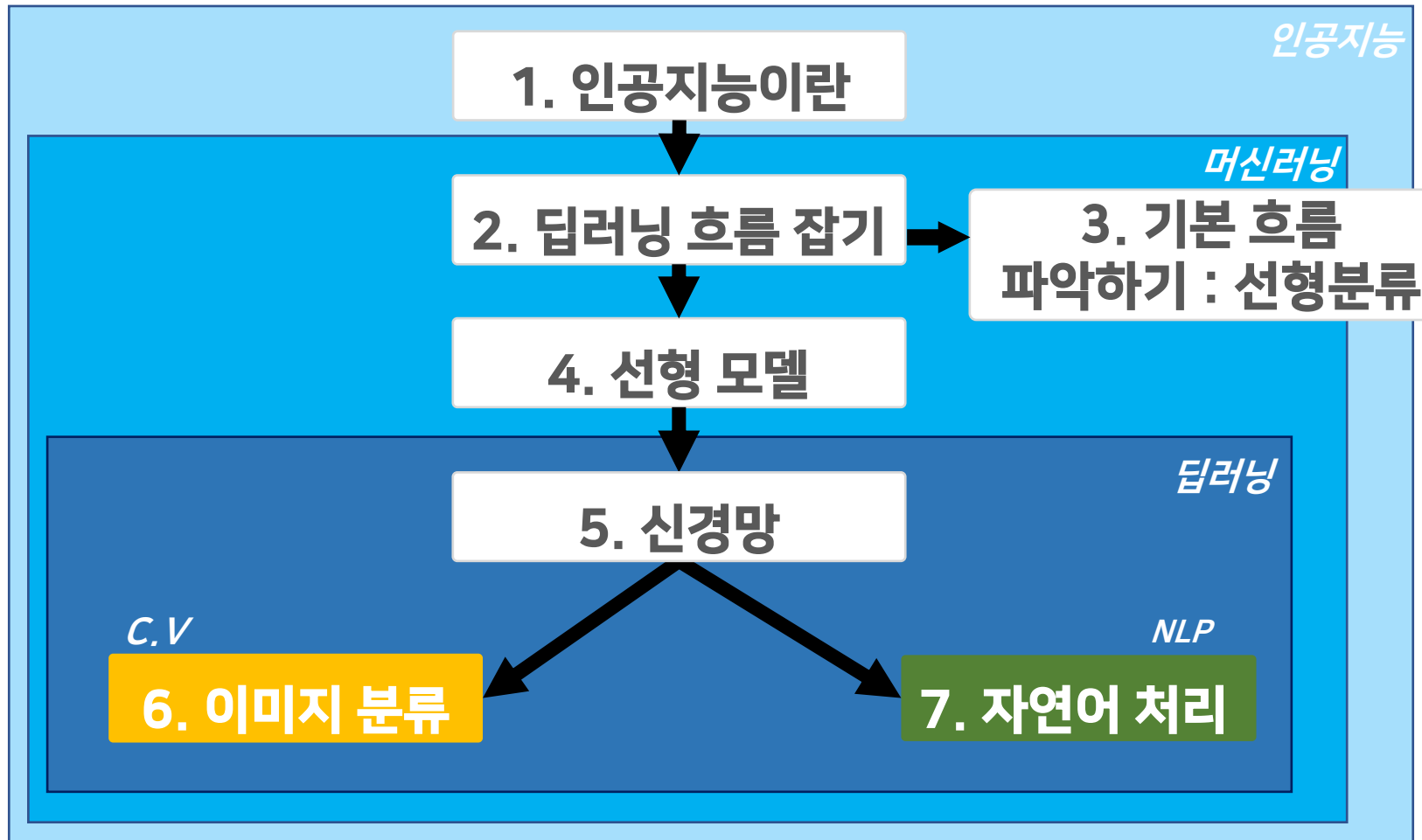
Computer Vision의 핵심 작업

# 분류Classification





# 학습 개요 Deep Learning



# Contents

## 1장 인공지능이란

1.1. 인공지능: 데이터 분석을  
위한 알고리즘

1.2. 머신러닝과 딥러닝

1.3. 머신러닝 과정

1.4. kaggle 소개



# (1) Python 소개

A. Python 언어 특징

B. library 소개

# A. Python 언어 특징

인터프리터(Interpreter) 방식 언어

인터프리터(해석기) 위에서 실행되는 언어

`interpret` : 작성된 프로그램 코드를 한 줄씩 순서대로 해석하고 실행하는 방식

`compile` : 텍스트 형태로 되어 있는 소스파일을 기계언어로 번역

➔ 작성된 부분까지 / 한 줄씩 실행가능  
오류 확인 용이

간결한 코드

코드가 간결하고 알아보기가 쉬운 고급(high-level) 프로그래밍 언어

들여쓰기로 코드 블록 구분

같은 들여쓰기로 블록을 구분

➔ 가시적으로 이해가 쉽다

# Python 언어 특징

## 객체지향 언어

물건(객체)와 물건(객체)과의 상호 관계를 중심으로 작성  
객체를 표현하고 객체 간의 상호 작용을 작성할 수 있는 방법을 제공

➔ 프로그램 모듈화, 코드 재사용, 유지보수에 용이  
체계적인 프로그램 작성  
생산성 향상

## 무료 오픈 소스

오픈소스로 개발된 언어  
module / library 무료로 사용 가능

➔ 전 세계 사용자들이 만들어 놓은 모듈을 무료로 사용가능

framework

tensorflow

pytorch

keras

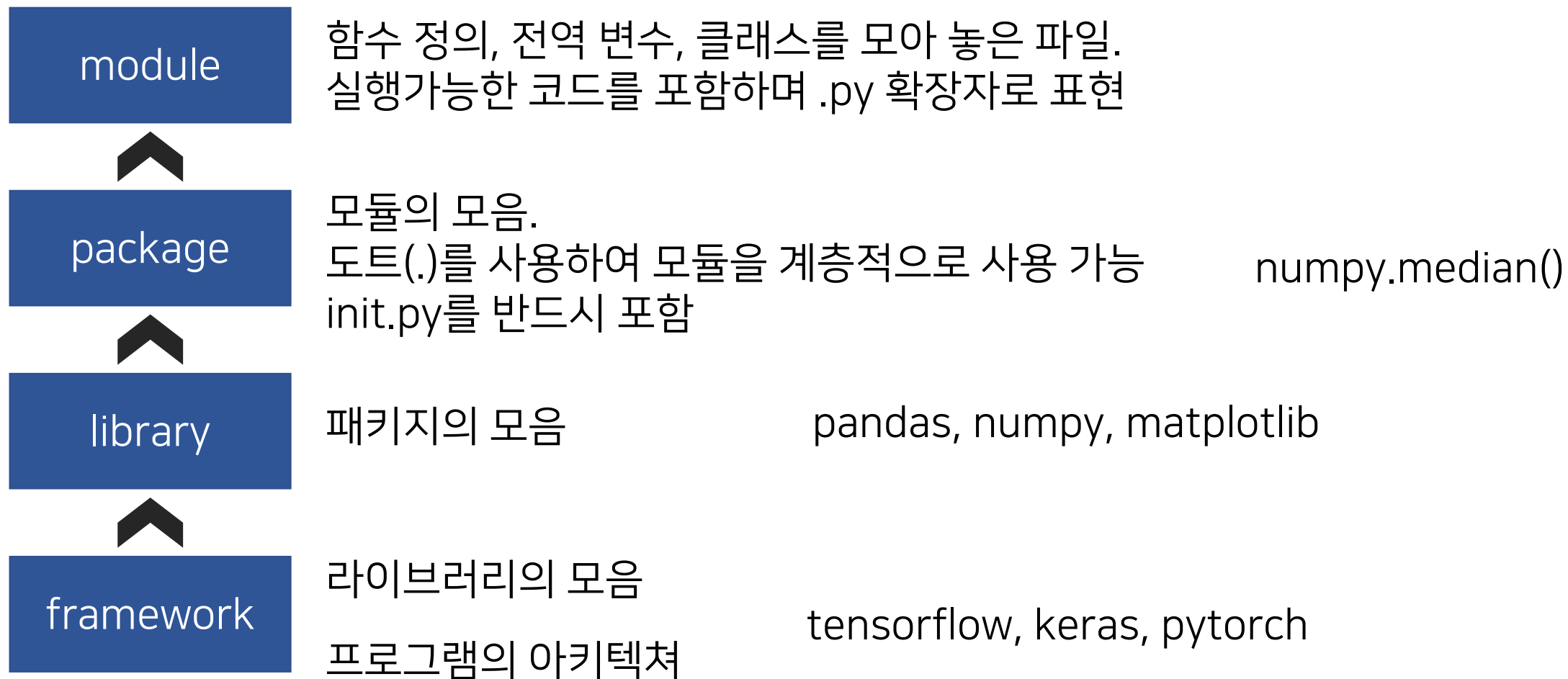
module/library

numpy

pandas

matplotlib

# Python library 구조



# Python 예제 : keras

## Softmax

```
1 import numpy as np
2 def softmax(a) :
3     exp_a=np.exp(a)
4     sum_a=np.sum(exp_a)
5     return exp_a/sum_a
```

<keras>

activation='softmax'

## Cross-Entropy Loss

```
1 import math
2 def cross_entropy_loss(a,i) :
3     return (-1)*math.log(softmax(a)[i])
```

loss='categorical\_crossentropy'

## B. library 소개

데이터 분석

pandas

numpy

데이터 시각화

matplotlib

seaborn

통계

statsmodels

머신러닝

scikit-learn

딥러닝

keras

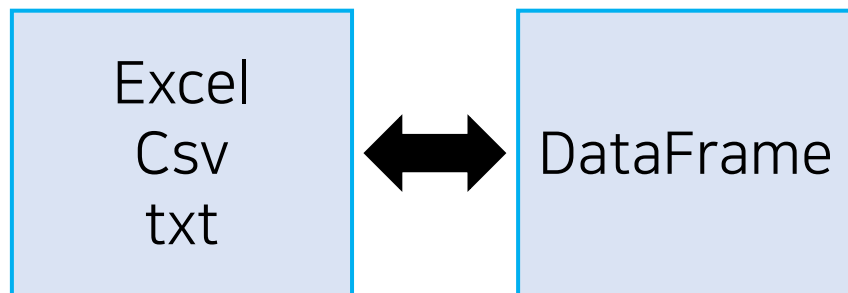
pytorch

# Pandas.DataFrame

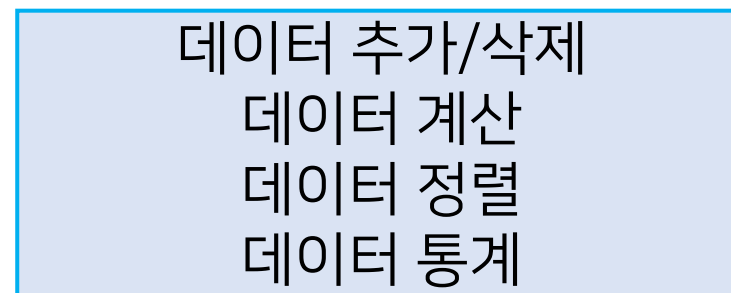
```
import pandas as pd
```

<https://pandas.pydata.org/>

문서를 데이터로



데이터 처리



<https://doorbw.tistory.com/172>

pd.loc pd.iloc pd.drop  
pd.concat pd.append  
pd.describe()  
pd.groupby()

# numpy

```
import numpy as np
```

<https://numpy.org>

행렬 계산에 특화된 라이브러리

기본적인 통계 함수 제공

데이터 생성  
데이터 사이즈 변경  
랜덤 데이터 생성

데이터 계산  
데이터 정렬  
데이터 통계

수학 함수 계산  
데이터 통계

```
np.zeros(10)  
np.ones(9)  
np.random.normal(2,3,4)
```

```
np.mean() # 평균  
np.var() # 분산  
np.std() # 표준 편차  
np.max() # 최대값
```

```
np.min() # 최소값  
np.median() # 중앙값  
np.percentile(a, 25)  
np.cos()  
np.round()  
np.argmax()
```

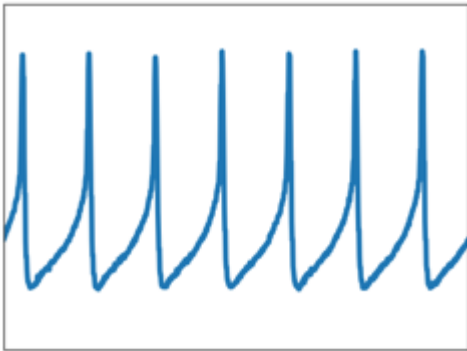
예습

<https://doorbw.tistory.com/171>

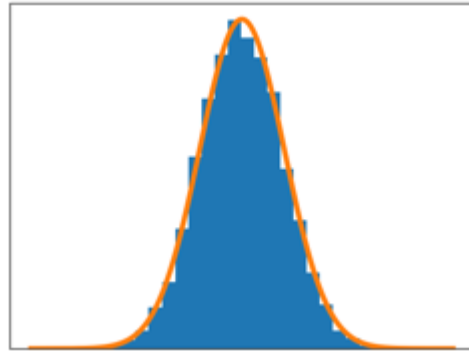


# Matplotlib: Visualization with Python

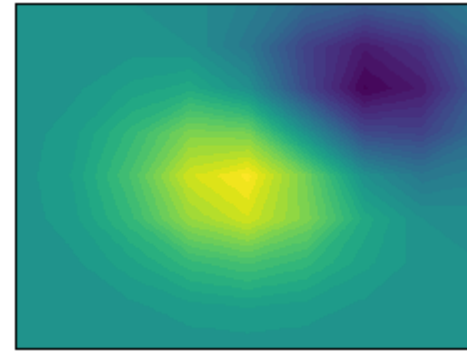
데이터 시각화



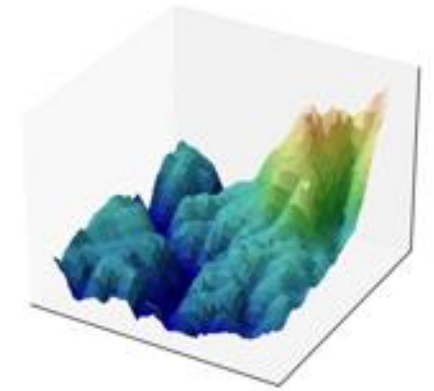
Line plot



Histograms



Image



3d plot

```
import matplotlib.pyplot as plt
```

<https://matplotlib.org/>

```
plt.figure()  
plt.xlabel
```

```
plt.plot  
plt.legend()
```

# keras : deep learning framework



Models	Sequential
Layers	Dense LSTM
Callbacks	
Data Preprocessing	
Optimizers	sgd adam
Metrics	accuracy
Losses	crossentropy

```
from keras.ooo import oooo  
from keras.models import  
Sequential
```

데이터 전처리  
딥러닝 모델  
딥러닝 학습

<https://keras.io/>

## (2) Kaggle 사용법



A. kaggle 소개

B. kaggle notebook

# A. kaggle 소개

The Kaggle logo, featuring the word "kaggle" in a blue, lowercase, sans-serif font with a small trademark symbol.

kaggle.com

회원가입 필요

GPU 지원 38시간/5일

A hamburger menu icon consisting of three horizontal lines.

kaggle



Home



Compete

데이터 사이언스 경진대회 플랫폼



Data

각종 데이터 제공/공유



Code

notebook 사용



Communities



Courses

학습용 콘텐츠 제공



More

# B. kaggle notebook

Kaggle notebook

실행

kaggle



< > Code



+ New Notebook

Your work

File



Import Notebook

파일 업로드

Link to GitHub



Download Notebook

파일 다운로드

Code

코드 작성용 라인

Markdown

메모용 라인

File Edit View Run Add-ons Help



Toggle line numbers

코드에 줄번호 표시

# kaggle notebook 사용법

code 입력

code 실행

결과 확인

Shift + Enter

실행 라인마다 결과 확인 가능

행 삽입

+ Code

+ Markdown

휴대폰 인증

GPU 사용  
인터넷 연계

라인 별 실행

vs

code 전체 실행

실행 결과에 차이 없음

a

print(a)

출력방법

print(a)

.ipynb

확장자

.py

notebook

편집기

vs code



Jupyter Notebook

colab  
kaggle



Visual Studio Code

colab

# Contents

## 1장 인공지능이란

실습예제 OR gate

# OR gate

```
def OR(x1,x2):  
    a1,a2,b = 0.3,0.3, 0.4  
    delta = 0.5  
    y = a1*x1+a2*x2+b  
    if y < delta:  
        return 0  
    else:  
        return 1
```

OR(0,0), OR(0,1), OR(1,0), OR(1,1)



0,1,1,1

