

# 데이터마이닝팀

## 4팀

김현우  
김준서  
서희나  
김수빈  
변석주



# CONTENTS



1. 데이터마이닝

2. 모델링

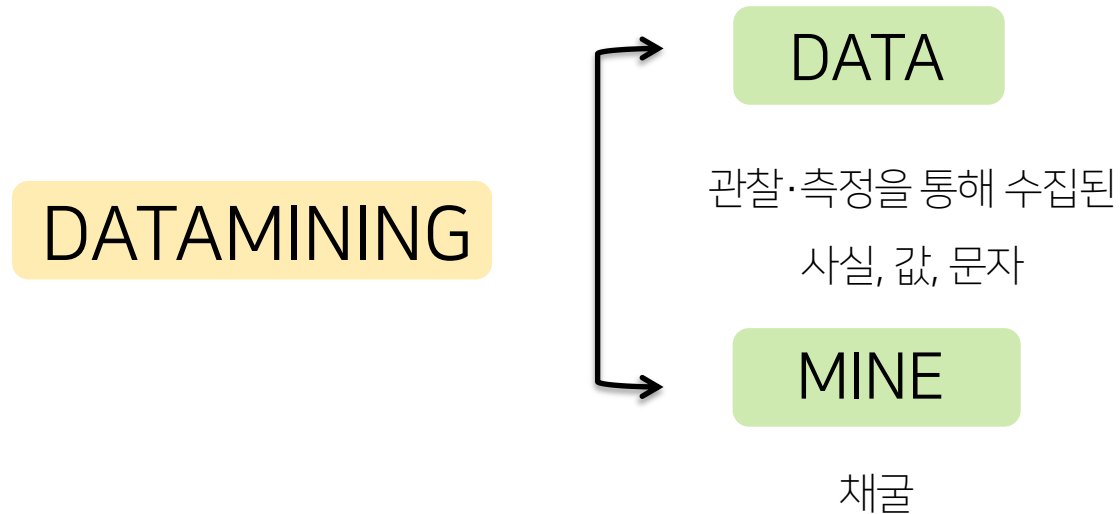
3. 모델링 전략

1

데이터 마이닝

## 데이터 마이닝의 정의

데이터 마이닝의 어원



대량의 데이터로부터 **유용한 정보와 패턴을 추출**해내는 과정

## 데이터 마이닝의 정의

데이터 마이닝의 일반적인 과정

- ① 데이터를 전처리
- ② 데이터로부터 패턴을 찾아냄
- ③ 패턴을 바탕으로 예측 진행  
→ 새로운 정보 얻어냄



DIKW 피라미드

DIKW 구조에서 데이터를 가공하여 현실 문제의 해결을 위해  
상황과 맥락에 맞추어 적용할 수 있는 지혜로 도달하는 과정과 동일

## 데이터 마이닝의 정의

데이터 마이닝의 일반적인 과정

- ① 데이터를 전처리
- ② 데이터로부터 패턴을 찾아냄
- ③ 패턴을 바탕으로 예측 진행  
→ 새로운 정보 얻어냄



DIKW 피라미드

DIKW 구조에서 데이터를 가공하여 현실 문제의 해결을 위해  
상황과 맥락에 맞추어 적용할 수 있는 지혜로 도달하는 과정과 동일

## 데이터 마이닝의 정의

데이터 마이닝의 일반적인 과정



### 데이터마이닝이란?

- 주어진 데이터를 바탕으로 **지혜**를 얻어내기 위한 방법
  - ① 데이터를 정리
  - ② 데이터로부터 패턴을 찾아냄
  - **머신러닝**: 정보를 효율적으로, 잘 설명할 수 있는
  - ③ 패턴을 바탕으로 예측 진행
- 새로운 방식으로 얻어낼 수 있도록 도와줌



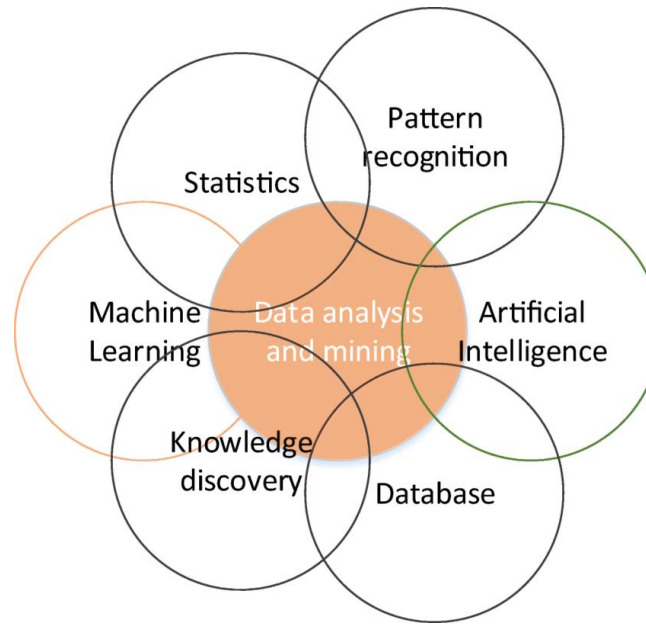
회귀 분류 클러스터링



DIKW 구조에서 데이터를 가공하여 현실 문제의 해결  
상황과 맥락에 맞추어 적용할 수 있는 지혜로 도달하는 고

YES답 머신러닝 3형제

## 데이터 마이닝의 간학문적 성격



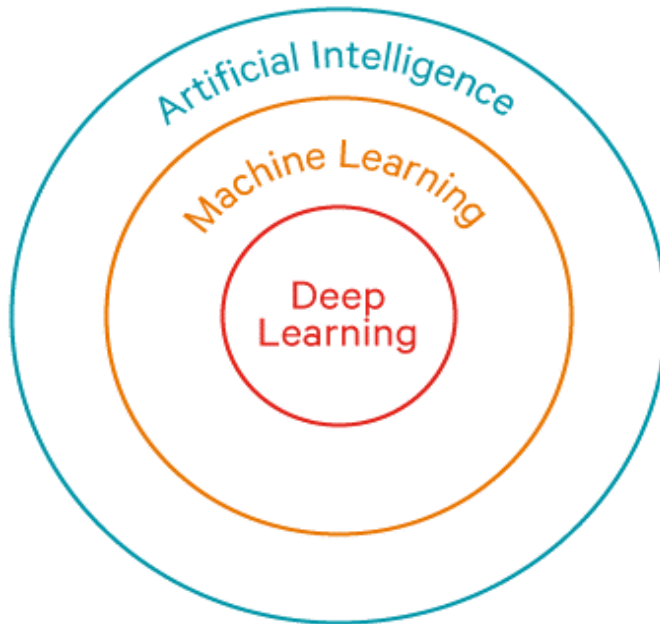
- 데이터 마이닝은 인공지능, 머신러닝, 딥러닝 등 **여러 학문의 교집합**에 위치
  - 데이터 처리, 모델링 학습, 평가 → 여러 학문들의 경계 넘나들  
(통계학, 수학, 컴퓨터 공학 등등..)



## 인공지능, 머신러닝, 딥러닝?

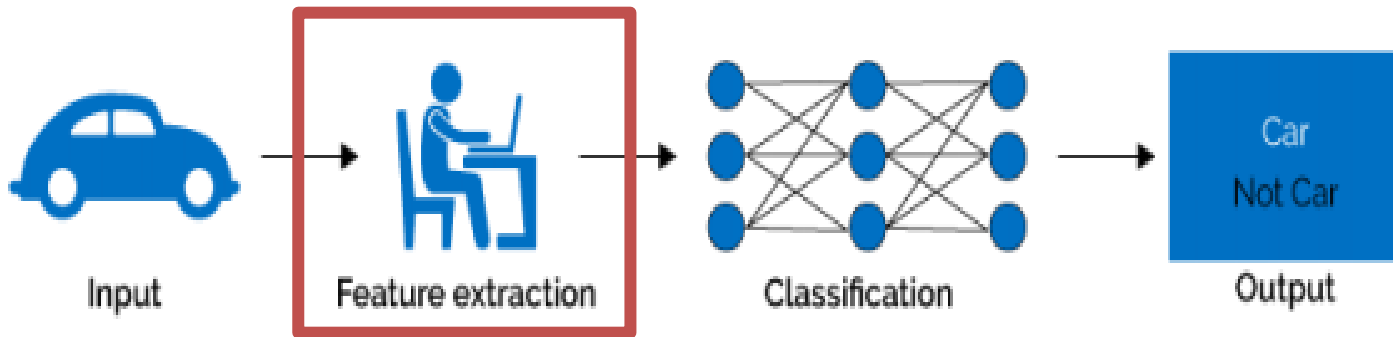
### 인공지능(AI ,Artificial Intelligent)

- 컴퓨팅을 이용한 학습 과정을 모두 포함하는 포괄적인 개념
  - 머신러닝과 딥러닝을 모두 포함하는 개념



알잘딱깔센 계산하겠다 기다려라 인간

## 인공지능, 머신러닝, 딥러닝?



### 기계학습(Machine Learning)

- 사람의 개입이 **최소화**된 학습 수행 방법
- 수행 목적(분류/회귀)에 적절한 모델 선정 → 컴퓨터 데이터 학습 후 결과 도출

## 인공지능, 머신러닝, 딥러닝?



### 딥러닝(Deep Learning)

- 사람의 **신경망**과 유사한 학습체계 구축해 목적 달성을 위한 과정 수행
  - 간단한 모델을 **여러 겹** 쌓아 매우 **복잡**하게 구성
    - 해석 어려움 (a.k.a '블랙박스 모델')



녀석들은 대체 뭘 본 거지?



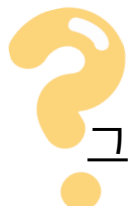
녀석들에게 보여준 걸  
나한테도 보여줘라!

## 데이터 마이닝의 목표

Feat. 통계야 도와줘~~



머신 러닝과 딥러닝이 "수행 과정"에 초점을 둔다면  
데이터 마이닝은 이에 더해 "인사이트를 얻어내는 것"을 목표로!



모델 선택이 적절했는가?

그 모델이 목표를 얼마나 잘 달성했는가?

선택한 모델의 성능을 높이는 방법은 무엇인가?



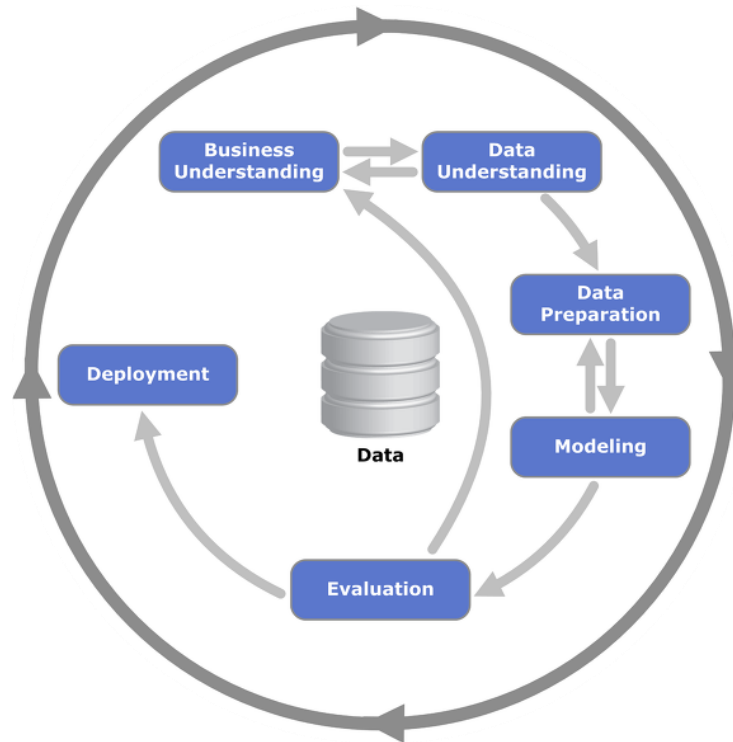
질문의 해답은...

# 통계학!



## 방법론 : CRISP\_DM

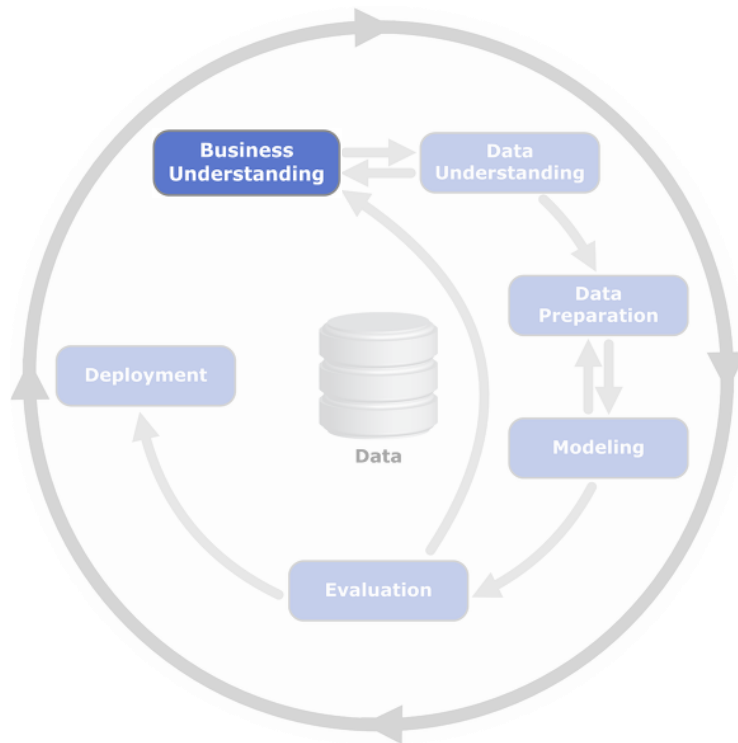
Cross-Industry Standard Process for Data Mining



데이터마이닝의 대표적인 분석 방법론  
크게 6개의 과정으로 이루어져 있음

## CRISP\_DM

Cross-Industry Standard Process for Data Mining

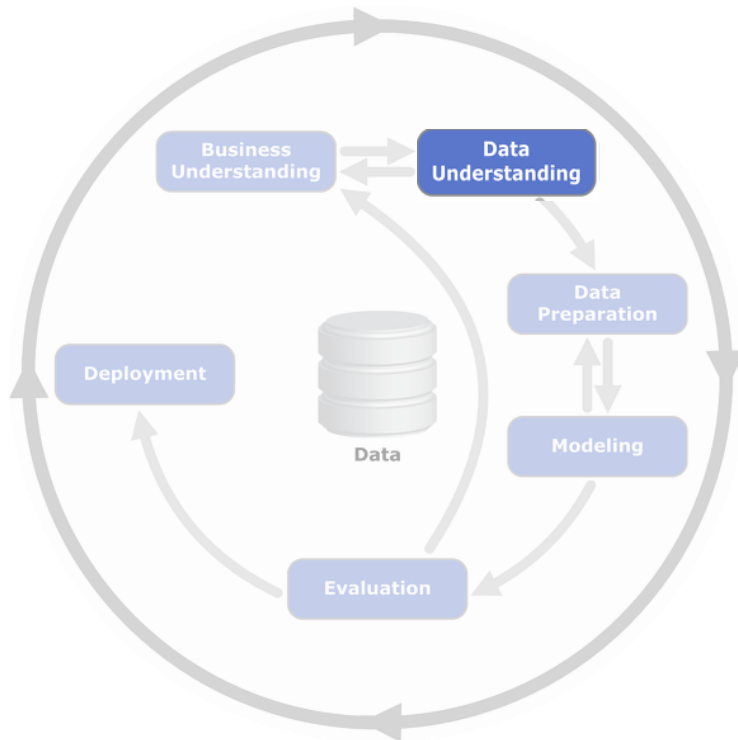


### ① 비즈니스 문제 이해

- 과제의 목적과 요구사항 이해
  - 도메인 지식을 활용
    - 초기 프로젝트 계획 수립
  - 주어진 데이터에 대한 **사전 지식**을 통해 데이터를 이해

## CRISP\_DM

Cross-Industry Standard Process for Data Mining

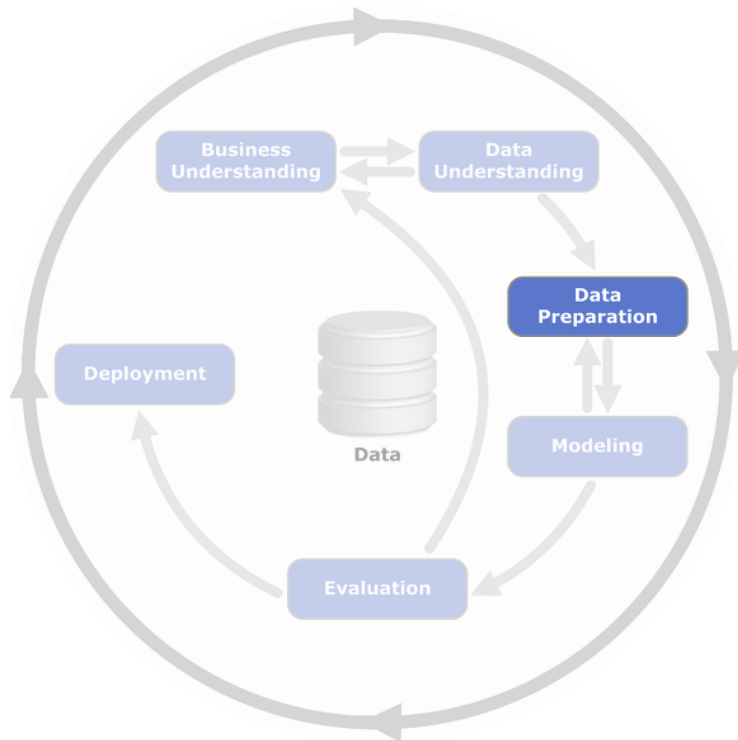


### ② 데이터 이해 (EDA)

- 해당 데이터를 수집 및 이해
  - 주어진 데이터를 직접 확인하는 것으로 **데이터를 이해**
- 변수 분포, 추이, 상관관계 시각화
- 이상치(outlier)와 결측치 확인

## CRISP\_DM

Cross-Industry Standard Process for Data Mining



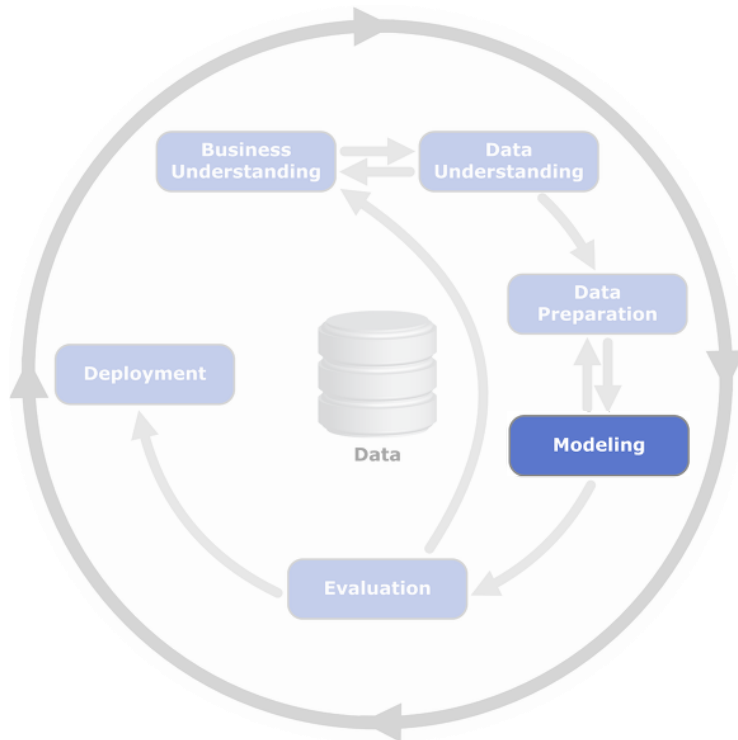
### ③ 데이터 준비

- 데이터 전처리 과정  
ex) 파생변수 생성 등
- 전처리 진행에 따라  
모델 성능이 달라짐



## CRISP\_DM

Cross-Industry Standard Process for Data Mining

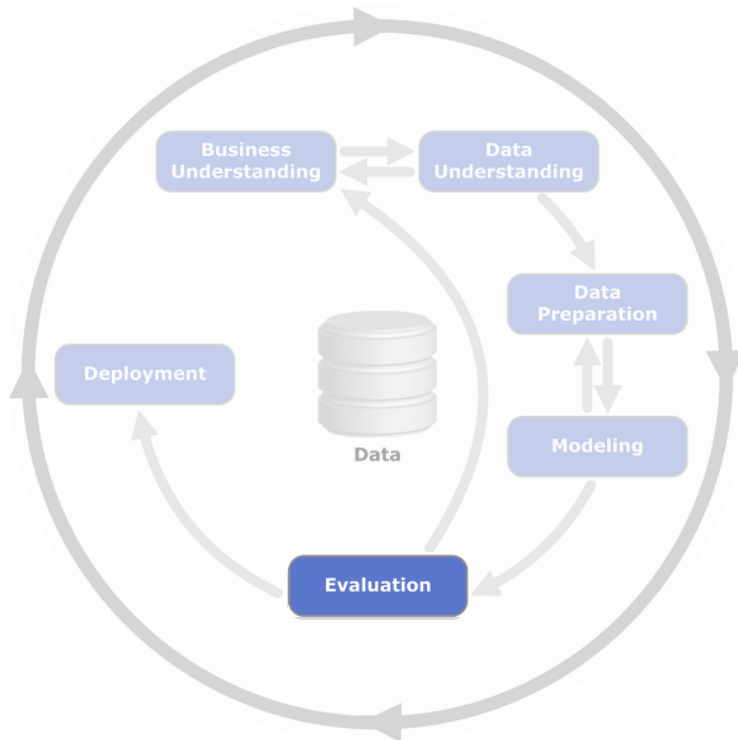


### ④ 분석 & 모델링

- 모델링 과정 수행 & 파라미터 최적화
- 모델링 기법 선택  
모델 테스트 계획 / 설계  
모델 작성과 평가

## CRISP\_DM

Cross-Industry Standard Process for Data Mining



### ⑤ 분석 모델 평가

- 모델링 성과 평가

→ 과제 목적에 맞추어 설정

- 분류모델

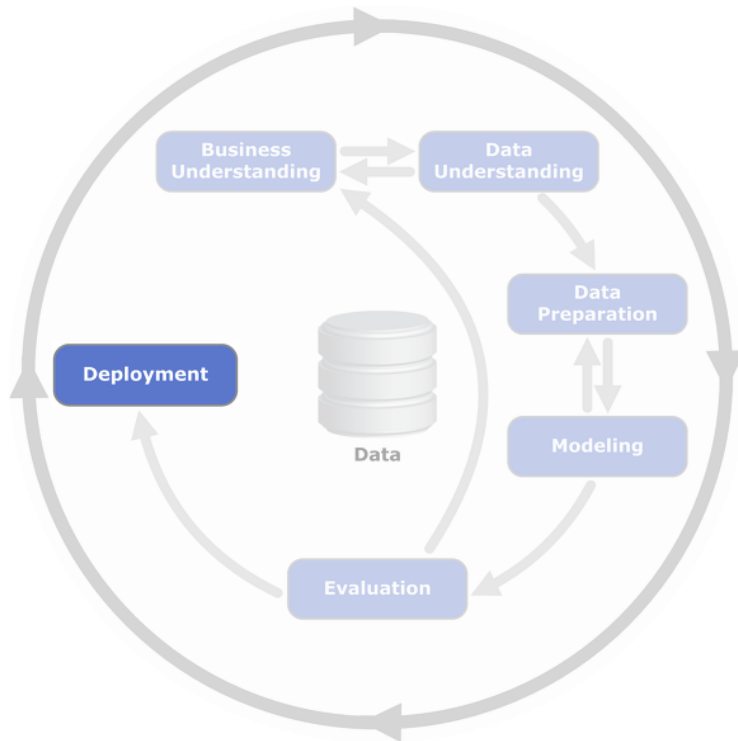
ex) Misclassification Rate

- 회귀모델

ex) RMSE, MAE

## CRISP\_DM

Cross-Industry Standard Process for Data Mining



### ⑥ 분석 결과 적용


- 분석 결과를 실제 서비스에 접목
  - 유의미한 결과를 도출함  
ex) 머신러닝 기법을 적용한 스팸메일 필터링 서비스 런칭

2

모델링

## Train Data & Test Data

ex) 집 데이터

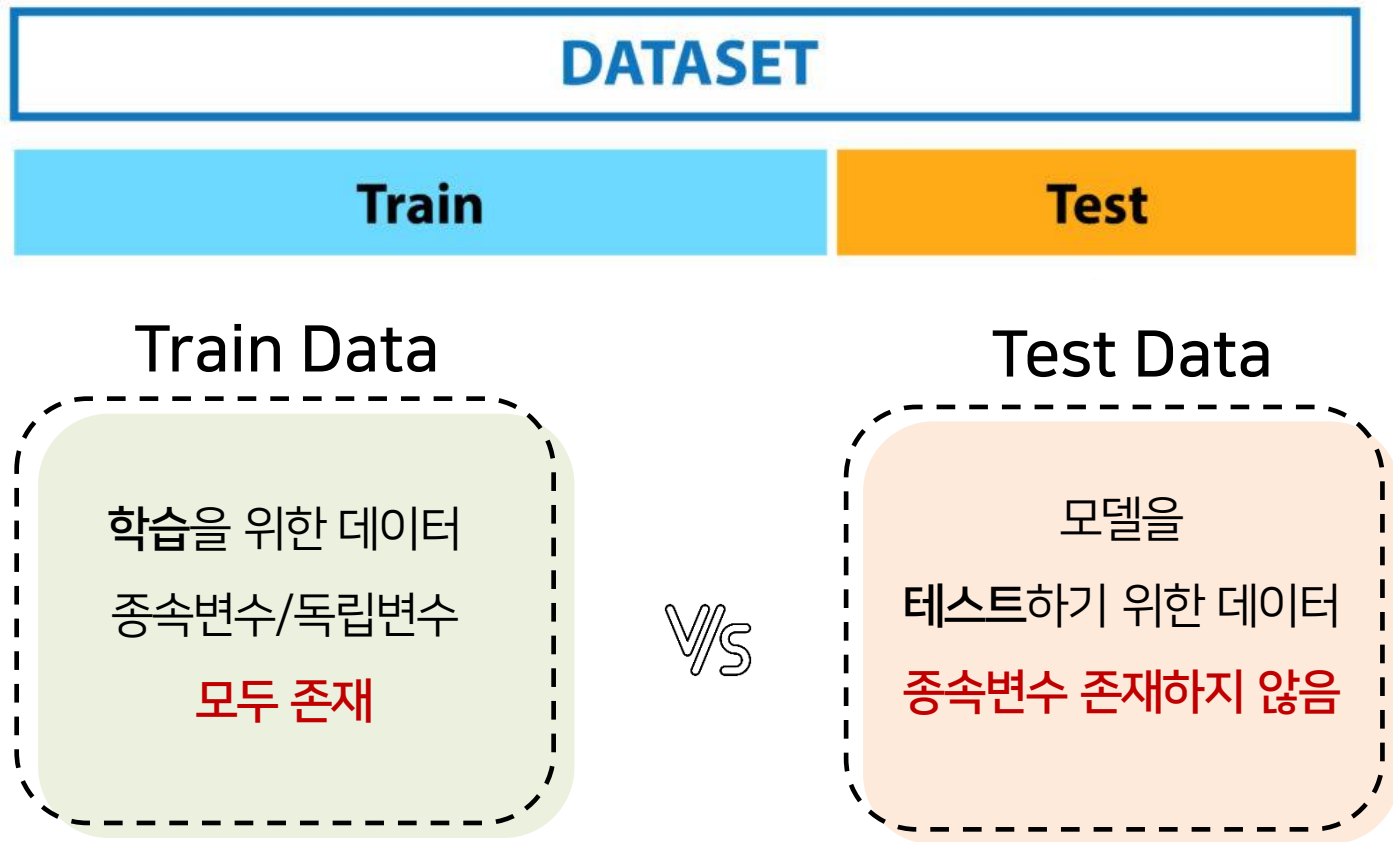


Bedroom	Sq.foot	Neighborhood	Sales Price
3	2000	Normaltown	\$250,,000
2	800	Hipstertown	\$300,000
2	850	Normaltown	\$150,000
1	550	Normaltown	\$78,000
4	2000	Hipstertown	\$150,000

독립변수와 종속변수로 이루어진 데이터를 바탕으로 **학습** 후  
 학습된 모델을 바탕으로 독립변수가 입력되면 종속변수를 **예측**

## Train Data & Test Data

Definition of Train Data & Test Data



## Train Data & Test Data

ex) 집 데이터

Bedroom	Sq.feet	Neighborhood	Sales Price
3	2000	Normaltown	\$250,,000
2	800	Hipstertown	\$300,000
2	850	Normaltown	\$150,000
1	550	Normaltown	\$78,000
4	2000	Hipstertown	\$150,000

[Train data]



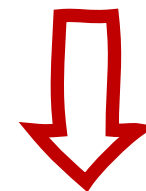
predict

Bedroom	Sq.feet	Neighborhood	Sales Price
3	2000	Hipstertown	???

[Test data]

Train data를 통해

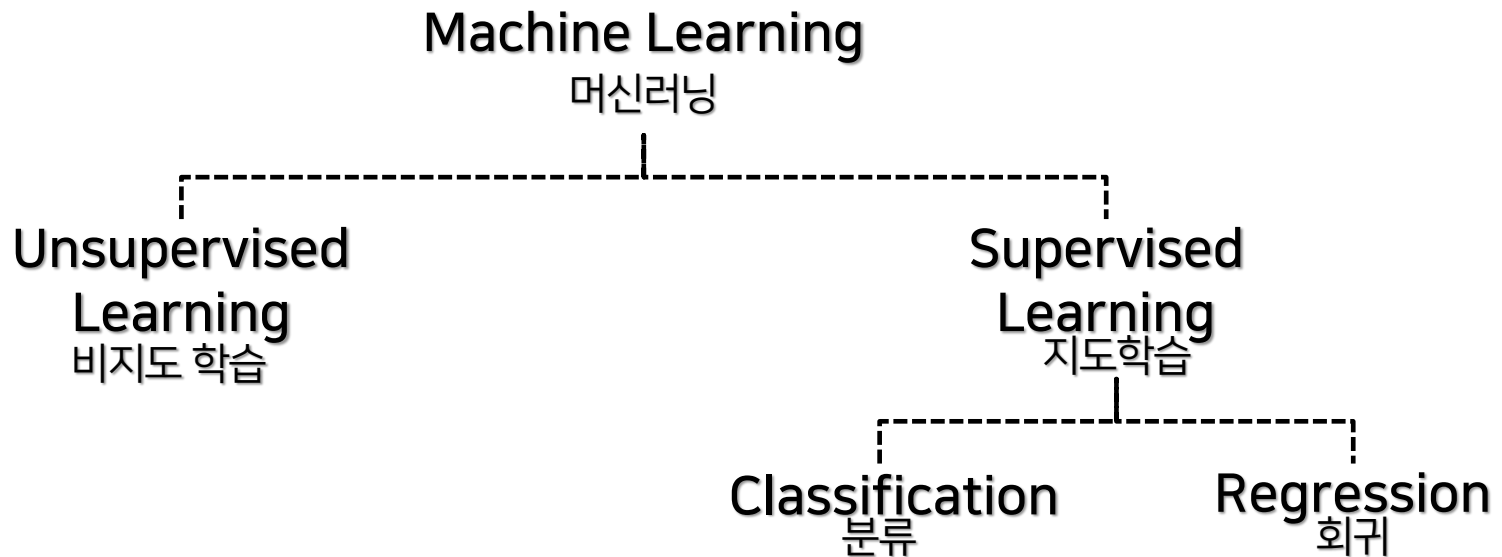
모델 학습



Test Data의

종속변수 예측

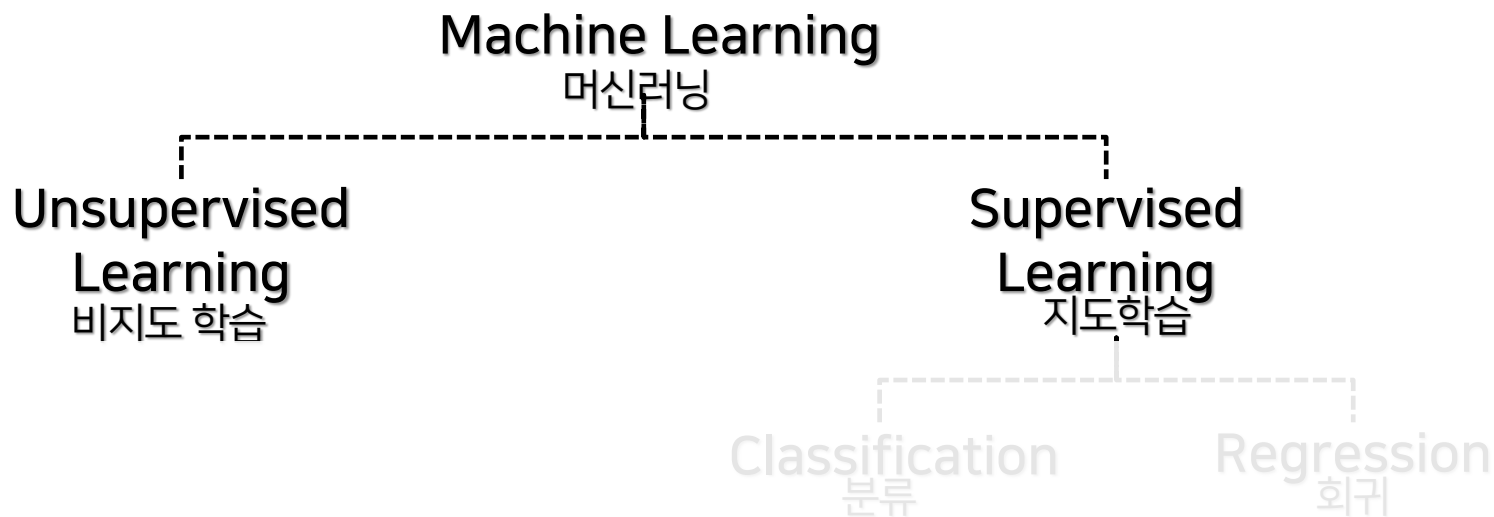
## 머신러닝의 종류



머신러닝은 학습 방식에 따라 지도학습과 비지도학습,  
지도학습은 목적에 따라 분류와 회귀로 나뉨

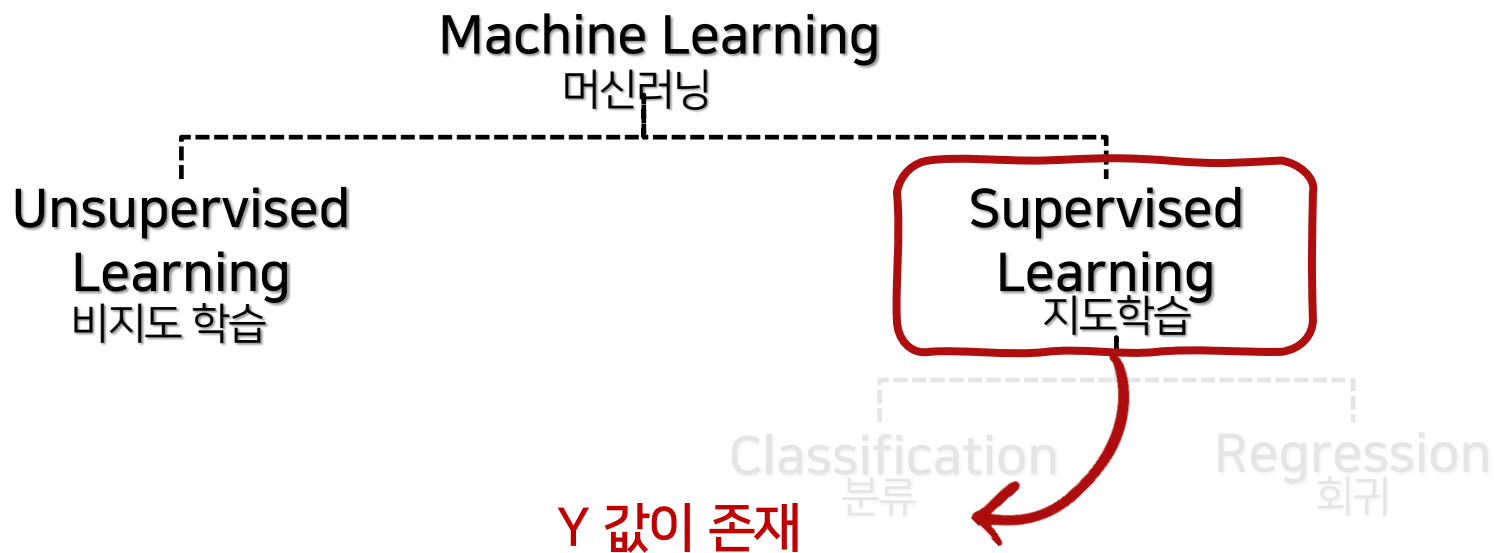


## 머신러닝의 종류



지도학습과 비지도학습은  
데이터의 라벨  
즉, **Y 값의 존재 여부**에 따라 구분

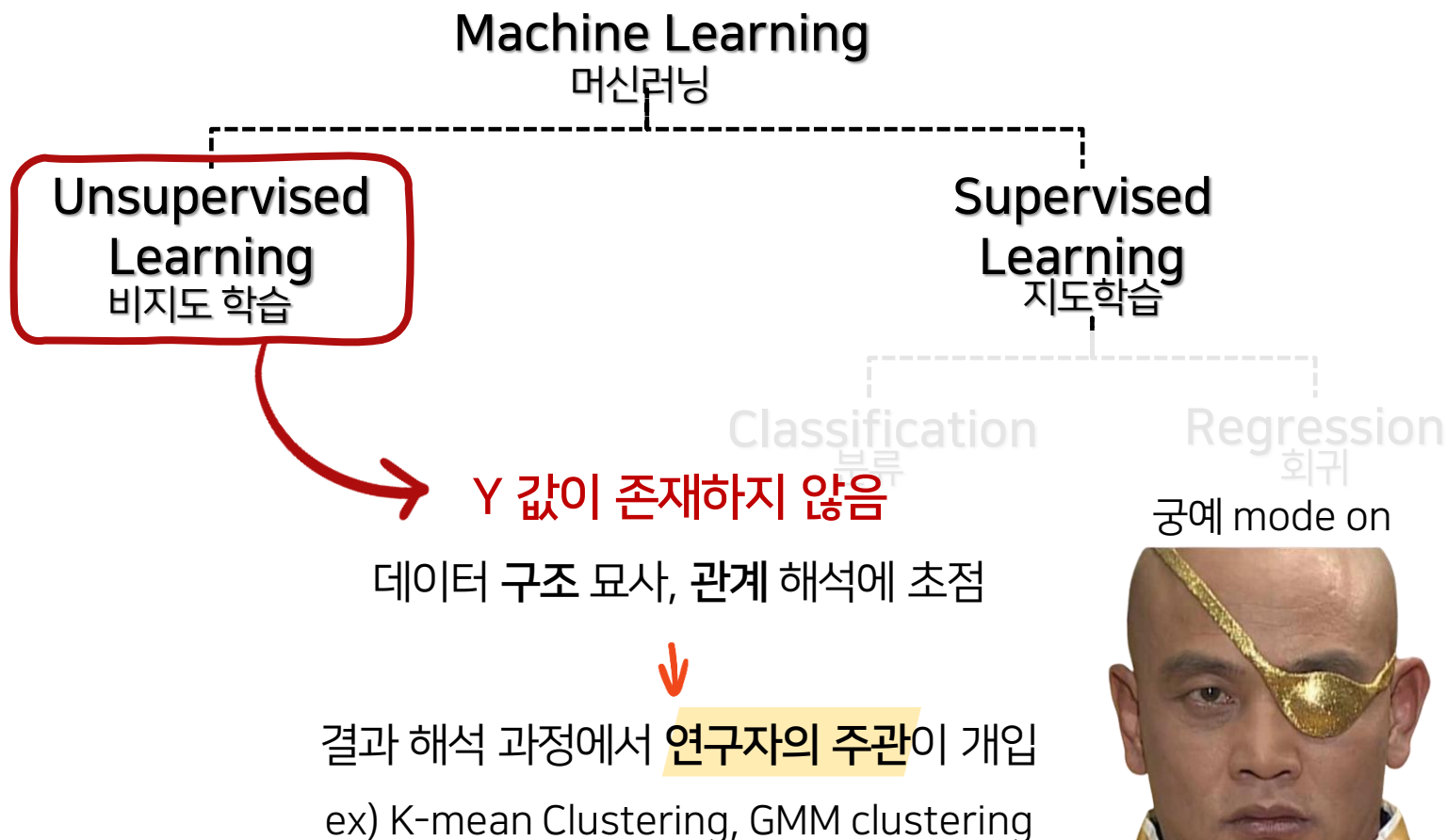
## 머신러닝의 종류



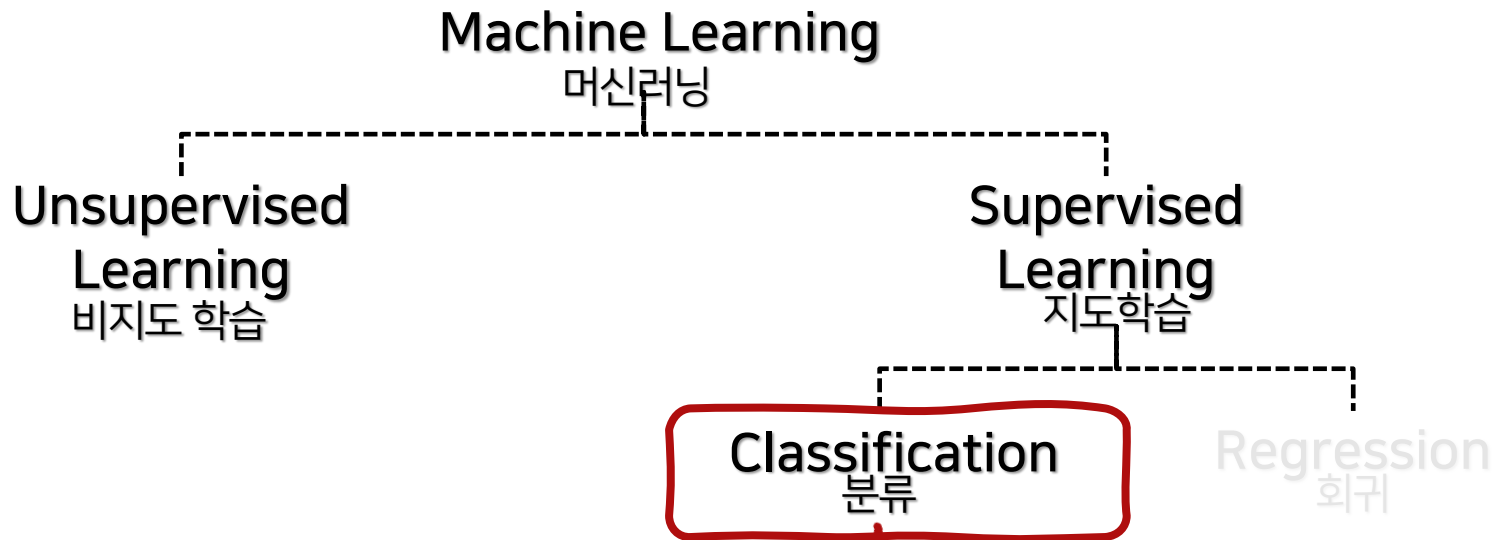
문제를 수행하고 이에 대한 답을 확인 가능

ex) Linear Regression, DecisionTree, Naive Bayes

## 머신러닝의 종류



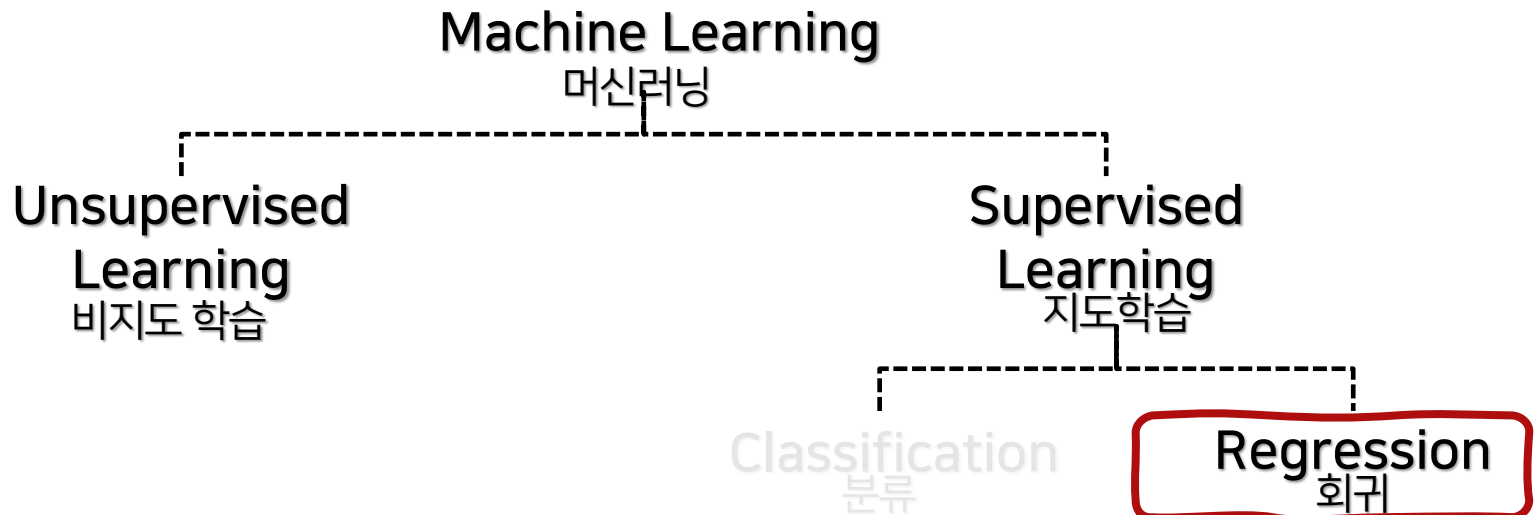
## 머신러닝의 종류



종속변수가 어떤 **카테고리**에 들어가는지 예측

$$P(y = j|X) = E(I(y = j)|X)$$

## 머신러닝의 종류

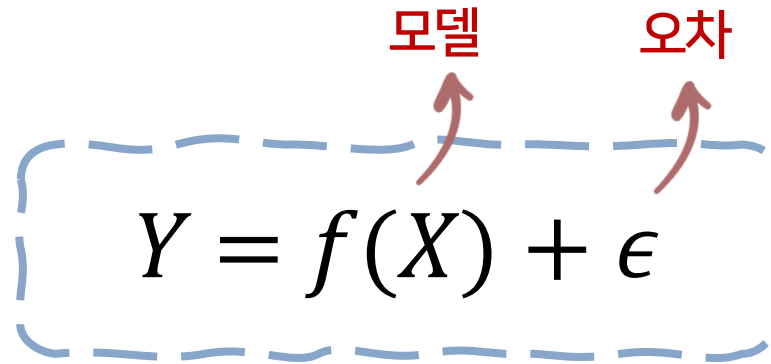


연속적인 형태를 띠는 종속변수의 값을 예측

$$E(y|X)$$

## Supervised Learning

지도 학습


$$Y = f(X) + \epsilon$$

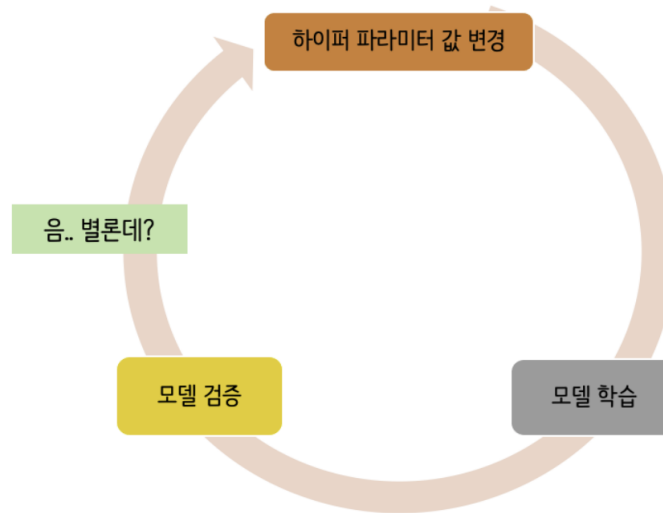
수학적 모델은 실제 데이터의 관계를 완벽하게 설명 못함

→ 대신, 여러 모델의 **하이퍼파라미터**를 조절하여

**실제 y값에 근접한 추정치**를 예측하는 모델 사용!

## Supervised Learning

지도 학습



### 하이퍼파라미터

모델의 성능을 결정하는 모델의 **매개변수**

→ 최적의 하이퍼파라미터를 찾아

모델 성능을 비약적으로 높일 수 있음!

## Supervised Learning

### MSE Decomposition

MSE (Mean Squared Error)

$$E[(y - \hat{y})^2]$$



회귀 모델의 성능을 평가

MSE를 줄여 모델의 성능을 높일 수 있음  
어떻게 줄일 수 있을까? → MSE Decomposition





## Supervised Learning

### MSE Decomposition

유도 과정:

$$\begin{aligned}
 E[(y - \hat{f})^2] &= E[(f + \varepsilon - \hat{f})^2] \\
 &= E[(f + \varepsilon - \hat{f} + E[\hat{f}] - E[\hat{f}])^2] \\
 &= E[(f - E[\hat{f}])^2] + E[\varepsilon^2] + E[(E[\hat{f}] - \hat{f})^2] + 2E[(f - E[\hat{f}])\varepsilon] + 2E[\varepsilon(E[\hat{f}] - \hat{f})] + 2E[(E[\hat{f}] - \hat{f})(f - E[\hat{f}])] \\
 &= (f - E[\hat{f}])^2 + E[\varepsilon^2] + E[(E[\hat{f}] - \hat{f})^2] + 2(f - E[\hat{f}])E[\varepsilon] + 2E[\varepsilon]E[E[\hat{f}] - \hat{f}] + 2E[E[\hat{f}] - \hat{f}](f - E[\hat{f}]) \\
 &= (f - E[\hat{f}])^2 + E[\varepsilon^2] + E[(E[\hat{f}] - \hat{f})^2] \\
 &= (f - E[\hat{f}])^2 + \text{Var}[\varepsilon] + \text{Var}[\hat{f}] \\
 &= \text{Bias}[\hat{f}]^2 + \text{Var}[\varepsilon] + \text{Var}[\hat{f}] \\
 &= \text{Bias}[\hat{f}]^2 + \sigma^2 + \text{Var}[\hat{f}].
 \end{aligned}$$

## Supervised Learning

### MSE Decomposition

$$E[(y - \hat{f})^2] = E[(f + \varepsilon - \hat{f})^2]$$

$\vdots$

$$= \text{Bias}[\widehat{f}]^2 + \text{Var}[\varepsilon] + \text{Var}[\hat{f}]$$

$$= \boxed{\text{Bias}[\widehat{f}]^2 + \text{Var}[\hat{f}]} + \boxed{\sigma^2}$$

Reducible Error

Irreducible Error

# Supervised Learning

## MSE Decomposition

**Bias**  $E[(y - \hat{f})^2] = E[(f + \epsilon - \hat{f})^2]$  추정된 모델이 실제 모델을 얼마나 잘 설명하는지와 관련된 지표

**Variance**

사용한 모델로 다른 데이터셋을 학습했을 때  
모델이 얼마나 달라지는지와 관련된 지표

$$= \underbrace{Bias[\hat{f}]^2}_{\text{Bias}} + \underbrace{Var[\hat{f}]}_{\text{Variance}} + \underbrace{\sigma^2}_{\text{Noise}}$$

좋은 모델일수록 Error가 낮음



Reducible Error인 Bias와 variance를 동시에 줄이면 되지 않을까?

Reducible Error

Irreducible Error

## Supervised Learning

## MSE Decomposition

**Bias(편차)와 Variance(분산)을**

$$E[(y - \hat{f})^2] = E[(f + \varepsilon - \hat{f})^2]$$

동시에 줄이는 것은 어려움!

$$= \text{Bias}[\widehat{f}]^2 + \text{Var}[\varepsilon] + \text{Var}[\hat{f}]$$

$$= \boxed{\text{Bias}[\widehat{f}]^2 + \text{Var}[\hat{f}]} + \boxed{\sigma^2}$$

**Variance-Bias Trade-off**

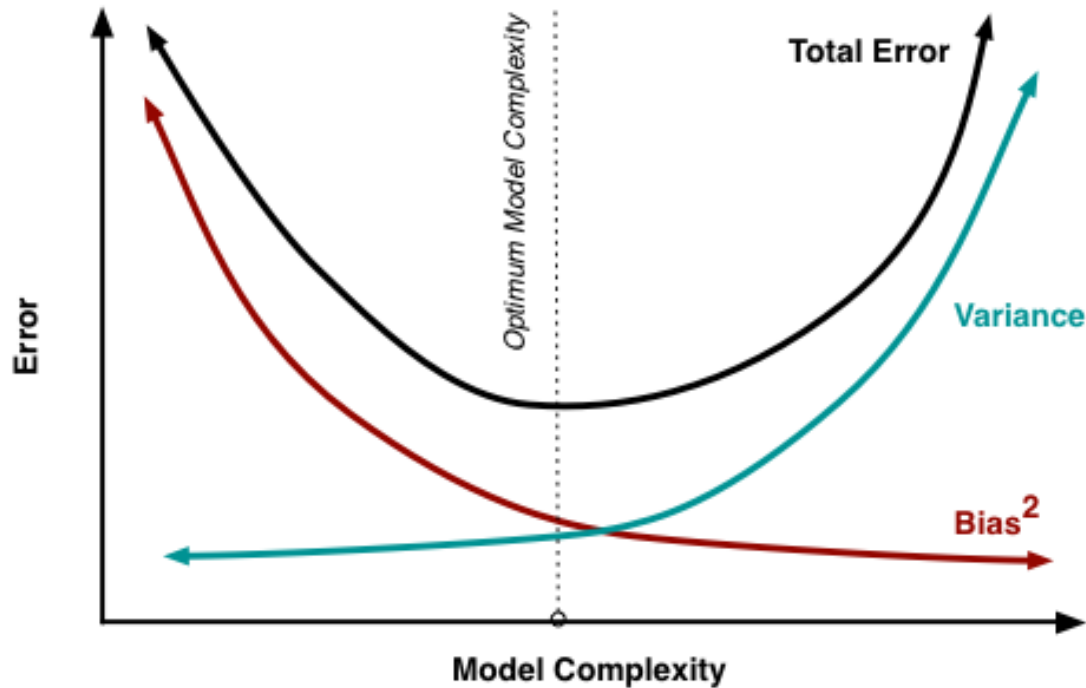
Reducible Error

Irreducible Error



## Supervised Learning

### Variance-Bias Trade-off

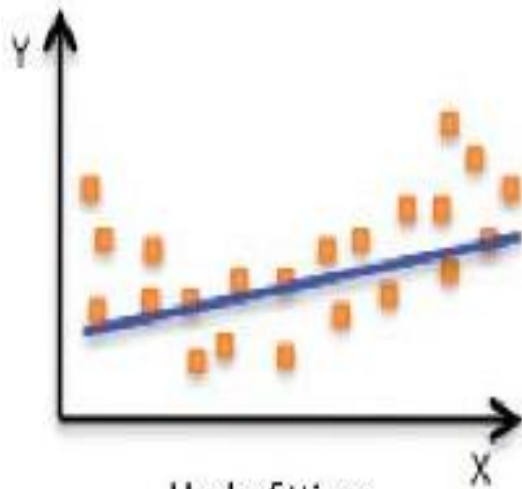


모델이 복잡해질수록 편차가 감소하지만 분산은 증가하고 있고,  
전체 오차가 점점 감소하다가 일정 부분 지나면 다시 증가하는 추세

## Supervised Learning

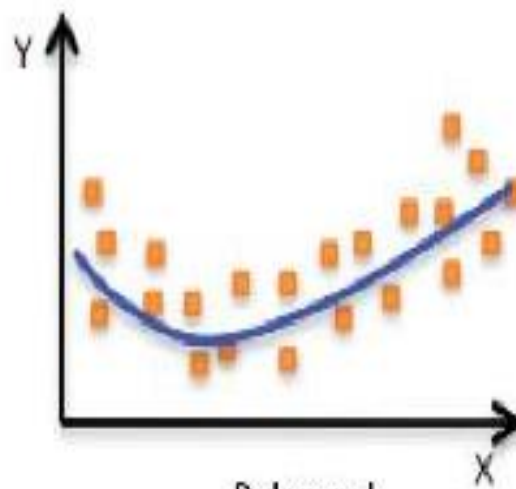
### Variance-Bias Trade-off

①



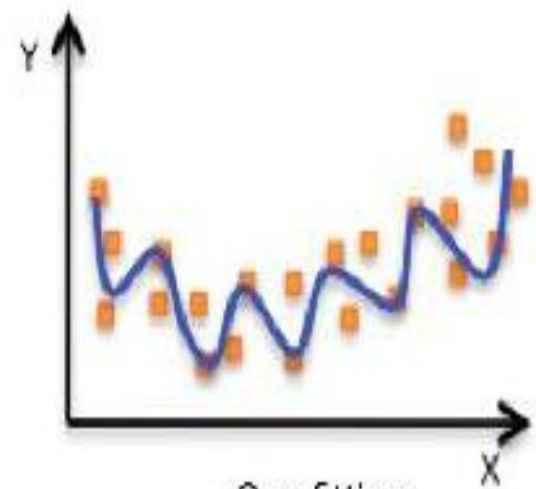
Underfitting

②



Balanced

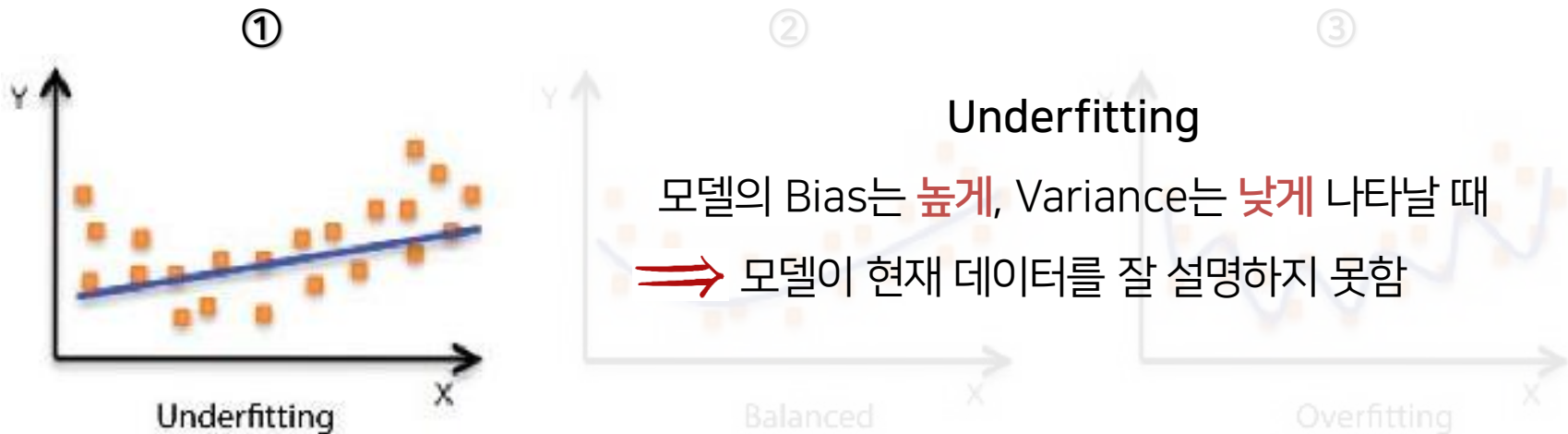
③



Overfitting

## Supervised Learning

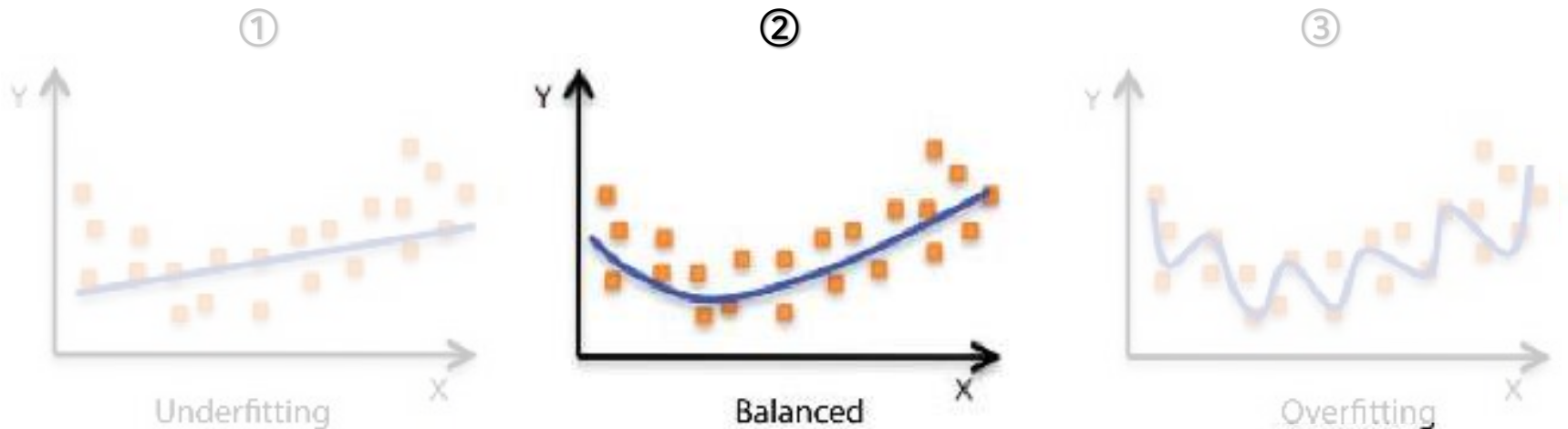
### Variance-Bias Trade-off



- ① Bias는 **높게**, Variance는 **낮게** 나타나고 있음
- ② Bias와 Variance가 **균형**을 이루고 **오차가 제일 적게** 나타나고 있음
- ③ Bias는 **낮게**, Variance는 **높게** 나타나고 있음

## Supervised Learning

### Variance-Bias Trade-off

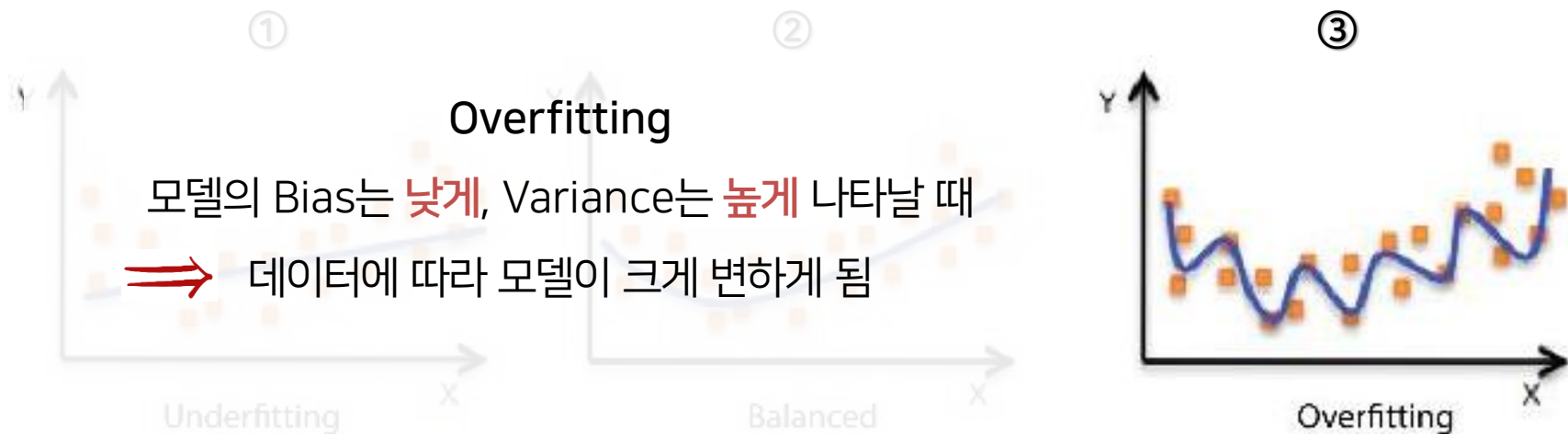


- ① Bias는 **높게**, Variance는 **낮게** 나타나고 있음
- ② Bias와 Variance가 **균형**을 이루고 **오차가 제일 적게** 나타나고 있음
- ③ Bias는 **낮게**, Variance는 **높게** 나타나고 있음



## Supervised Learning

### Variance-Bias Trade-off



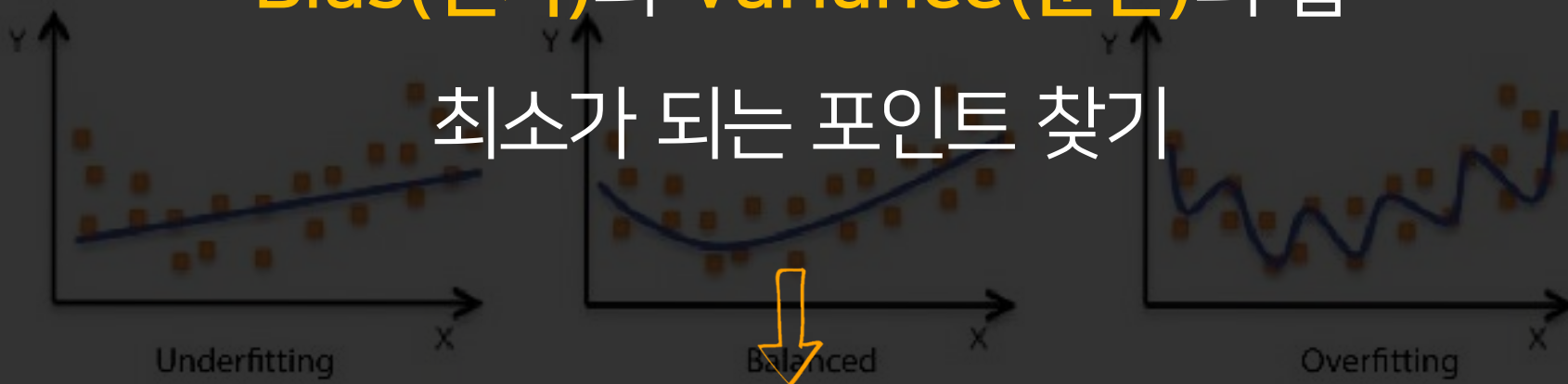
- ① Bias는 높게, Variance는 낮게 나타나고 있음
- ② Bias와 Variance가 균형을 이루고 오차가 제일 낮게 나타나고 있음
- ③ Bias는 낮게, Variance는 높게 나타나고 있음

## Supervised Learning

Variance-Bias Trade-off

① Bias(편차)와 Variance(분산)의 합

최소가 되는 포인트 찾기



① Bias는 크게, Variance는 낮게 나타나고 있음

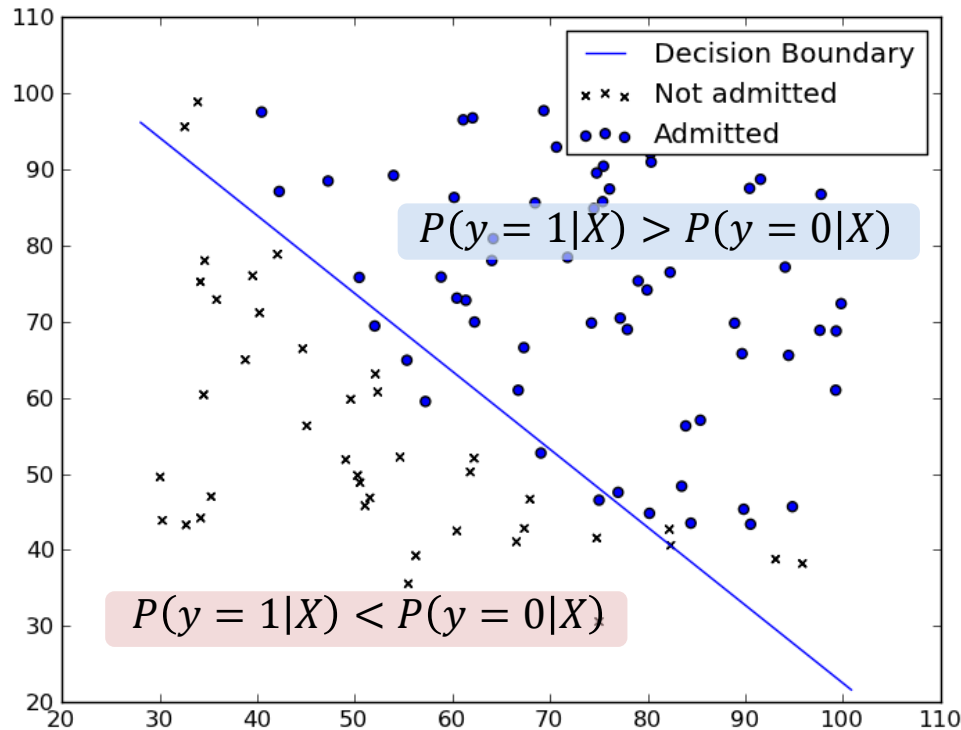
## Optimum Model Complexity

② Bias와 Variance가 균형을 이루고 오차가 제일 낮게 나타나고 있음

③ Bias는 작게, Variance는 높게 나타나고 있음

## Various Types of Models

판별 모델 vs. 생성 모델

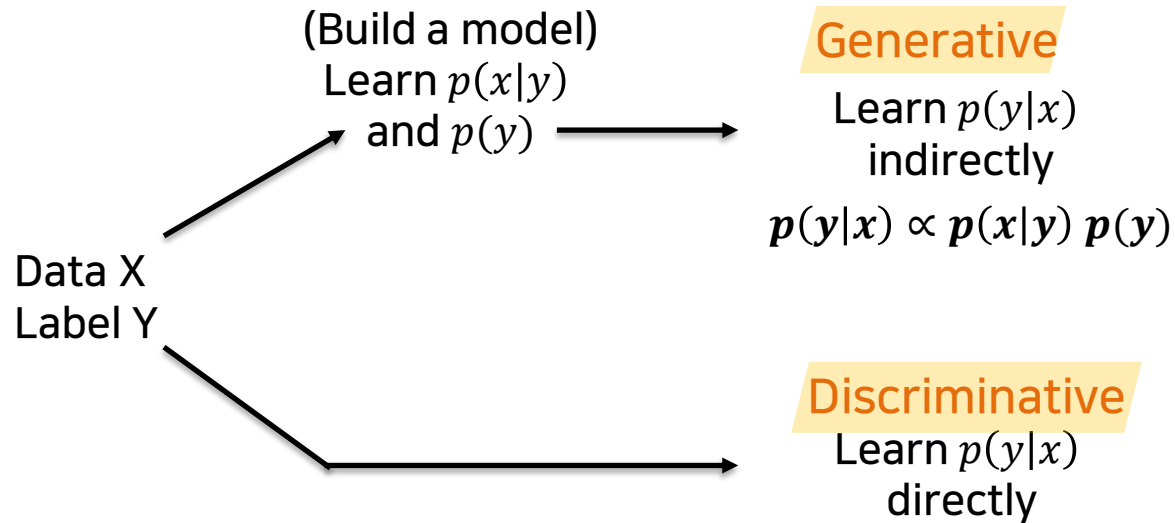


대부분의 분류 모델은 분류를 위해

$P(y = j|X) = E(I(y = j)|X)$ 를 계산

## Various Types of Models

판별 모델 vs. 생성 모델

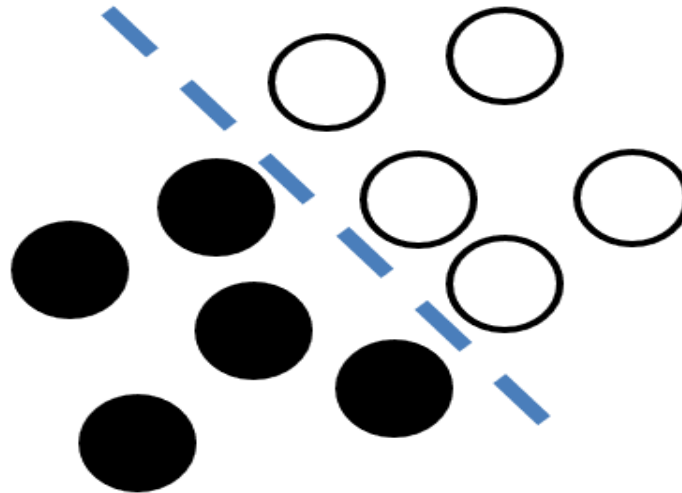


$P(y = j|X) = E(I(y = j)|X)$ 를 구하는 방식에 따라  
판별 모델과 생성 모델로 구별 가능

## Various Types of Models

판별 모델 vs. 생성 모델

### Discriminative Modeling



Class의 차이에 주목하여 데이터  $X$ 가 주어졌을 때  $y$ 가  $j$ 일 확률,

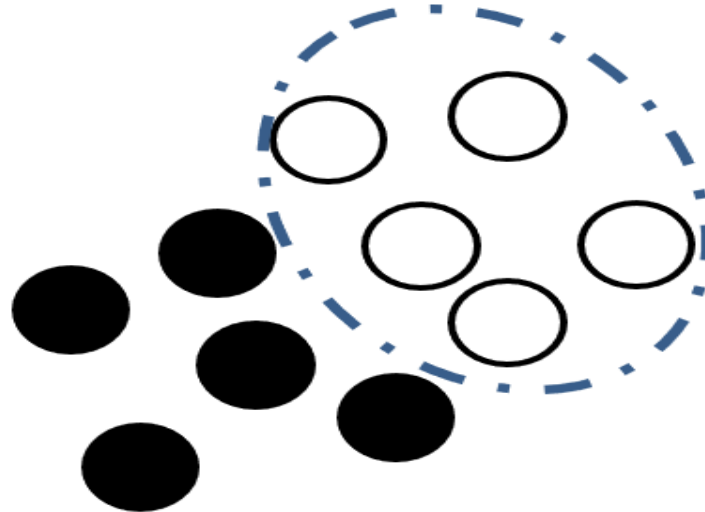
즉  $P(y = j|X)$ 를 직접 학습하여 분류를 진행함

⇒ 우리가 알고 있는 대부분의 모델은 판별 모델에 속함!

## Various Types of Models

판별 모델 vs. 생성 모델

### Generative Modeling



Class의 분포에 주목하여  $P(y = j|X)$ 를  
 $P(y = j)$ 와  $P(X|y = j)$ 를 통해 **간접적**으로 구함

## Various Types of Models

판별 모델 vs. 생성 모델

### Generative Modeling

어떻게  $P(y = j)$ 와  $P(X|y = j)$ 를 통해  
**간접적으로**  $P(y = j|X)$ 를 계산할까?



**베이즈 정리**(Bayes' Theorem)을 사용!

Class의 분포에 주목하여  $P(y = j|X)$ 를

$P(y = j)$ 와  $P(X|y = j)$ 를 통해 **간접적으로** 구함

## Various Types of Models

판별 모델 vs. 생성 모델

$$P(B_J | A) = \frac{P(A | B_J) P(B_J)}{\sum_{i=1}^n P(A | B_i) P(B_i)}$$

베이즈 정리를 사용하면  $P(y = j|X)$ 를  
 $P(y = j)$ 와  $P(X|y = j)$ 를 통해 간접적으로 구할 수 있음  
이를 위해  $y, X|y$ 의 분포를 가정해줘야 함



## Various Types of Models

판별 모델 vs. 생성 모델

### 장점 ①

실제 데이터의 분포가 모델에서 가정한 분포와 일치할 때 좋은 성능을 보임

이 경우에는 적은 데이터를 학습하더라도

판별 모델에 비해 훨씬 좋은 성능을 보임

### 장점 ②

$P(y = j|X)$ 를 간접적으로 구하는 방식의 모델임

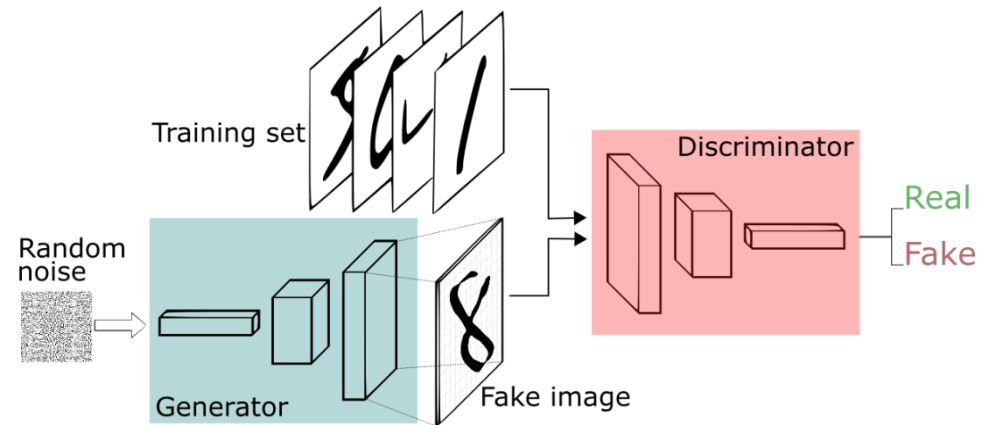
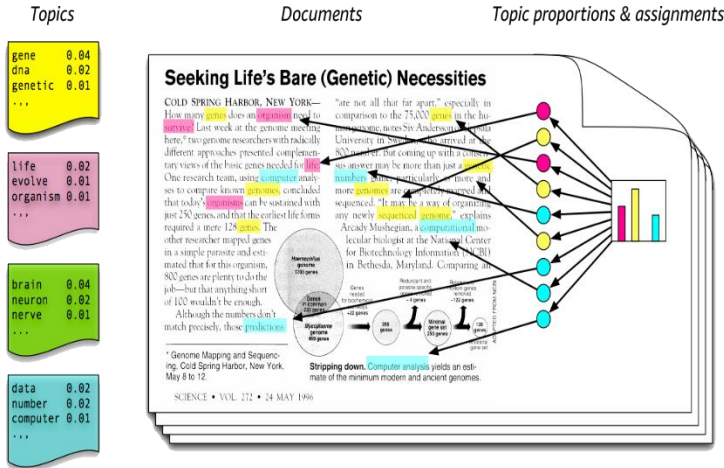
데이터 셋에서 종속 변수  $Y$ 가 주어지지 않더라도

모델이  $P(y = j|X)$ 를 간접적으로 구할 수 있음

⇒ 지도학습 외에도 비지도학습에도 사용 가능!

# Various Types of Models

## 판별 모델 vs. 생성 모델



이런 생성 모델은 머신러닝과  
딥러닝 분야에서 활발하게 사용되고 있음

## Various types of models

모수적인 모델 vs. 비모수적인 모델

### 모수적인 모델

파라미터를 가지고 있고, 모델의 학습 과정에서  
해당 파라미터를 추정하는 모델

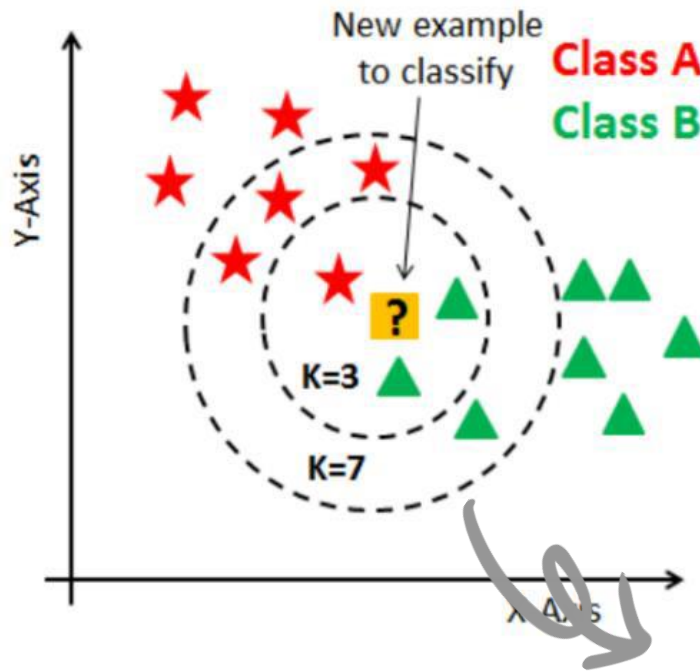
### 비모수적인 모델

알고리즘을 통해 바로 예측 값을 출력하는 모델  
예) KNN(K-Nearest Neighbor)모델

## Various types of models

모수적인 모델 vs. 비모수적인 모델

### KNN(K-Nearest Neighbor) 모델



#### KNN Classifier

$k$ 의 개수 따라  
decision boundary가 달라짐

$k$  : "Hyperparameter"

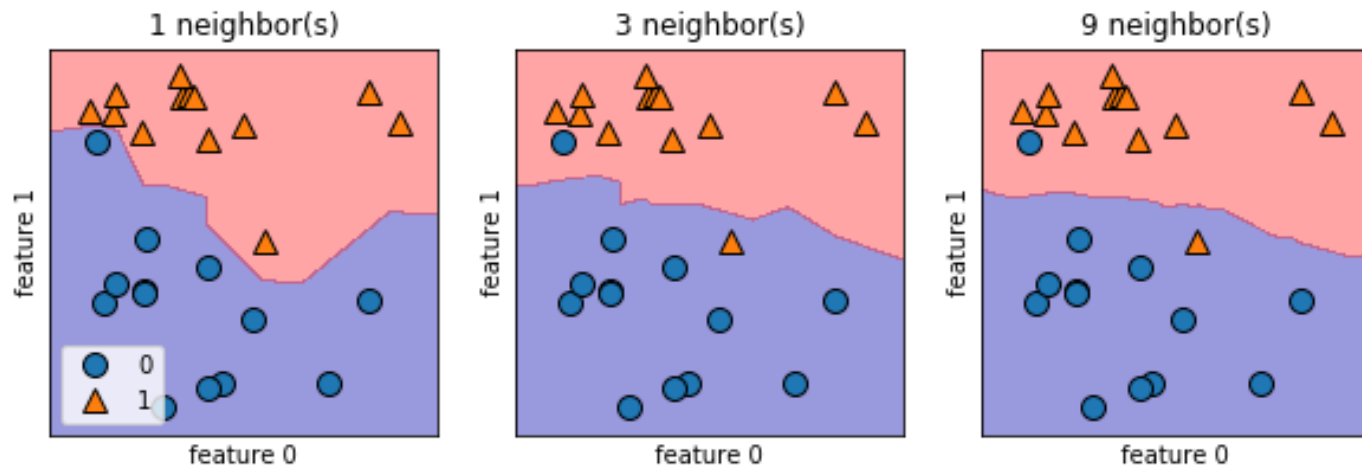
비슷한 데이터는 비슷한 결과값을 가진다

## Various types of models

모수적인 모델 vs. 비모수적인 모델

### KNN(K-Nearest Neighbor) 모델

Decision boundaries created by the nearest neighbors model for different values of  $n\_neighbor$



$k$  개수가 많아질수록 모델의 **decision boundary** 크게 바뀌지 않음

## Various types of models

모수적인 모델 vs. 비모수적인 모델

KNN(K-Nearest Neighbor) 모델

### KNN Regression

회귀문제에도 적용 가능

→ 타겟 데이터의 Y값을 타겟 데이터와 가까이 있는

k개 데이터의 Y값의 평균을 예측 값으로 사용!



$E(Y|X)$

# 3

## 모델링 전략

## 교차 검증 (Cross Validation)

### 교차 검증 (Cross Validation)이란?

분석 과정에서 주어진 train data를  
다시 **Train set**와 **Validation set**로 나누어  
모델의 적절성을 평가하는 방법

### Why CV?

- ① 과대적합 방지
- ② 모델의 성능 정확하게 판단



## 교차 검증 (Cross Validation)

Hold-out: Train-Test Split



### Train Set

Validation set이 제외된 train data로만 모델의 학습 진행

## 교차 검증 (Cross Validation)

Hold-out: Train-Test Split



### Validation Set

모델을 학습하는데 이용하지 않은 새로운  
데이터로 모델의 성능 측정을 위한 데이터 셋

## 교차 검증 (Cross Validation)

Hold-out: Train-Test Split



Test data로 모델 성능 예측 불가능한 이유

Test data는 **y값**이 존재하지 않기 때문에!

Validation Set

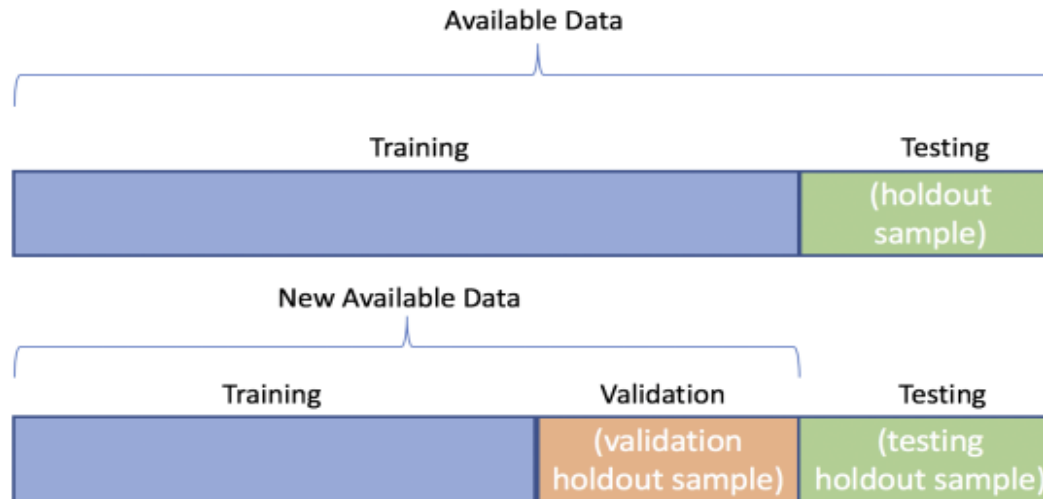
모델을 학습하는데 이용하지 않은 새로운  
데이터로 모델의 성능 측정을 위한 데이터 셋

아니 없어요 그냥



## 교차 검증 (Cross Validation): Hold-out

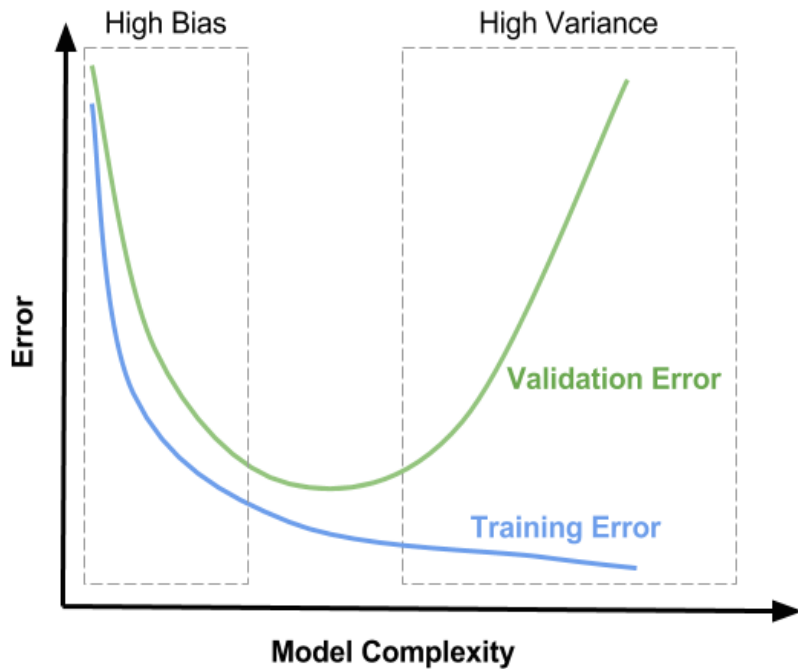
Train-Test Split을 통해 단일한 검증 데이터셋 생성



- ① 새로운 train, validation set 만들어냄
- ② Validation set이 제외된 train data로만 모델 학습
- ③ Validation data 이용하여 성능 예측

## 교차 검증 (Cross Validation): Hold-out

Train-Test Split



단점

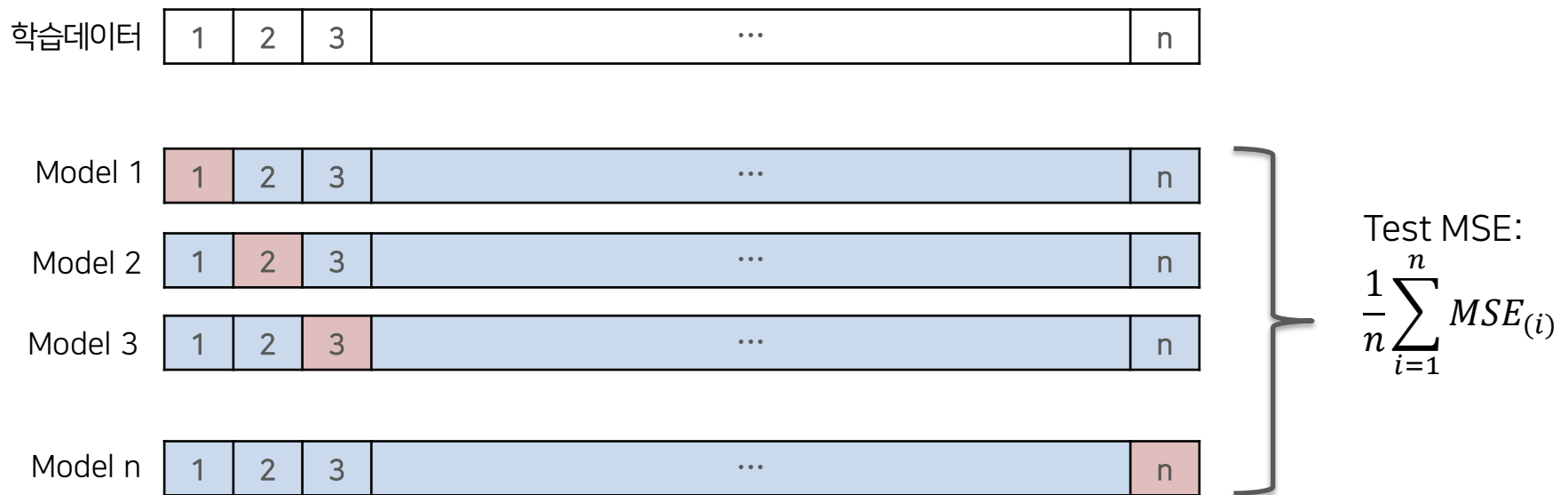
- Validation set이 전체 데이터의 경향성 포함한다는 보장 없음
- 이상치들의 집합으로 구성될 가능성
  - 작은 데이터 셋의 한계



LOOCV, k-fold CV

## 교차 검증 (Cross Validation)

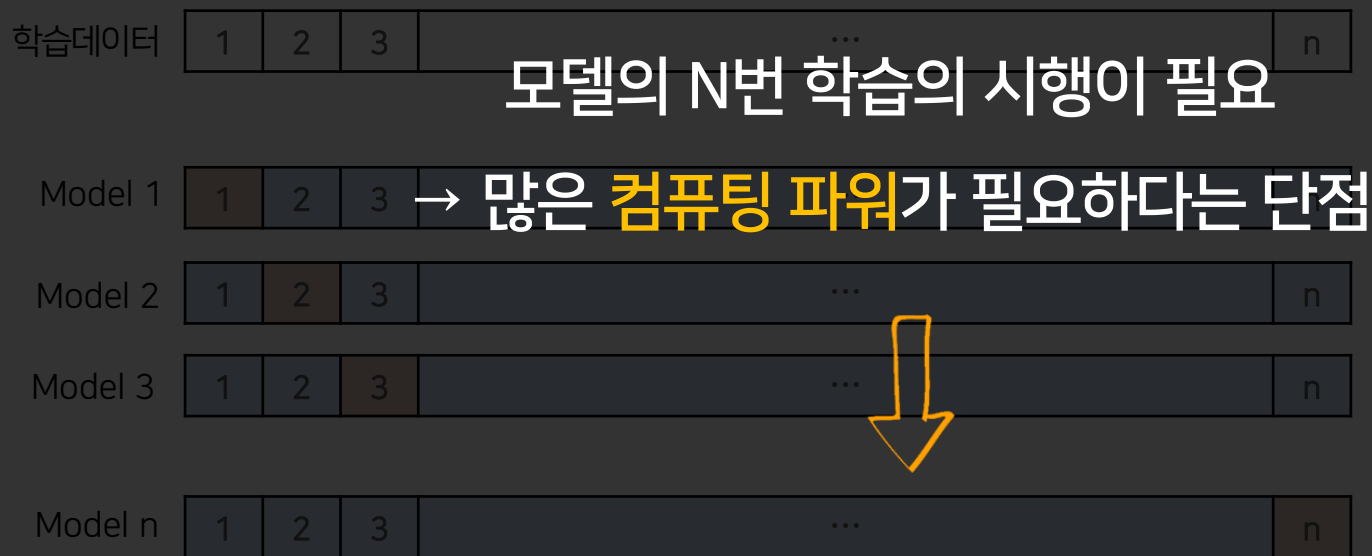
LOOCV(Leave-one-out CV)



n개의 전체 데이터에서 **한 개의 데이터를 검증 데이터**로,  
 나머지 **n-1개의 데이터를 학습데이터**로 사용하여 n번의 검증을 시행하는 방식

## 교차 검증 (Cross Validation)

LOOCV(Leave-one-out CV)



Test MSE:

$$\frac{1}{n} \sum_{i=1}^n MSE_{(i)}$$

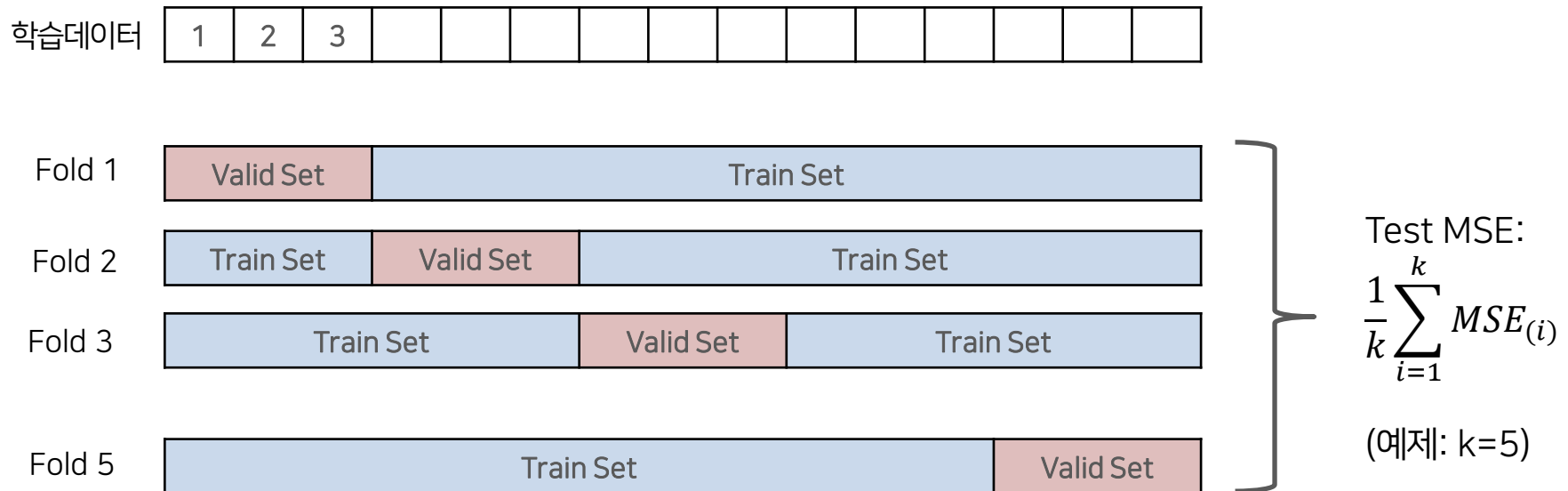
그러나 데이터셋이 매우 작은 경우에는

모든 교차검증 방법 중에서는 가장 효과적으로 모델의 성능을 평가

나머지  $n-1$ 개의 데이터를 학습데이터로 사용하여  $n$ 번의 검증을 시행하는 방식

## 교차 검증 (Cross Validation)

### K-Fold 교차검증(K-Fold CV)



전체 데이터를 K 개의 집합(set, fold)으로 분할, 하나의 집합을 검증 데이터셋으로,  
나머지 K-1개의 집합을 학습 데이터로 사용하여 총 K번의 검증을 시행하는 방식



## 교차 검증 (Cross Validation)

### K-Fold 교차검증(K-Fold CV)



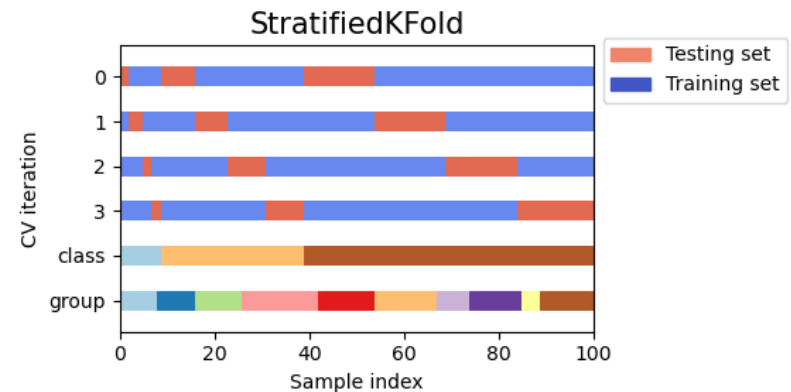
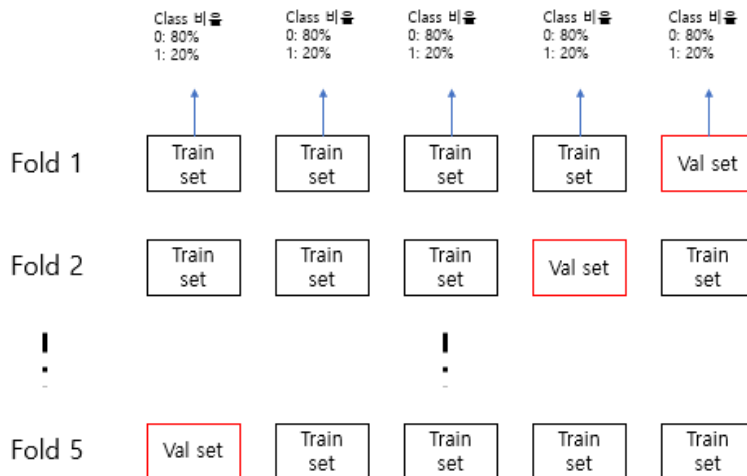
#### 장점

- LOOCV보다  
컴퓨팅 파워를 잡아먹지 않음
- 단순 Train-Test Split과 달리  
교차검증 과정에서  
전체 데이터를 전부 활용 가능

⇒ '과적합 여부 판단'

## 교차 검증 (Cross Validation)

층화 K-Fold 교차 검증(Stratified K-Fold CV)



K- Fold CV를 통해 더욱 정확하게 모델의 성능을 측정할 수 있지만  
여전히 검증 데이터셋이 전체 데이터의 경향을 반영하지 못함

## 교차 검증 (Cross Validation)



층화 K-Fold 교차 검증(Stratified K-Fold CV)

데이터를 분할할 때,

전체 데이터의 분포를 고려하여 분배!!



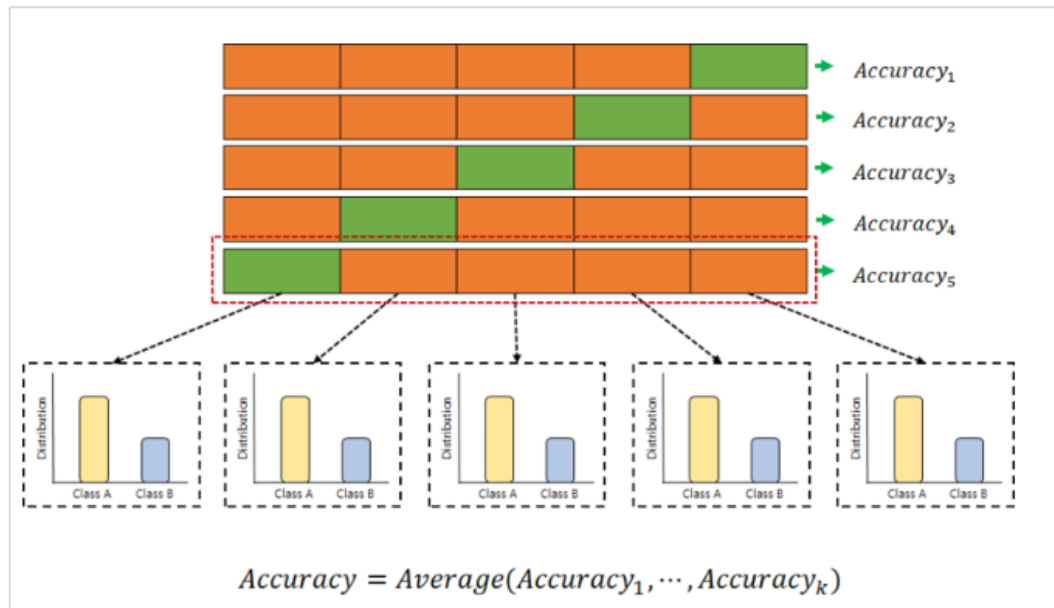
"Stratified K-Fold CV"

(층화 K-Fold 교차 검증)

K- Fold CV를 통해 더욱 정확하게 모델의 성능을 측정할 수 있지만  
여전히 검증 데이터셋이 전체 데이터의 경향을 반영하지 못함

## 교차 검증 (Cross Validation)

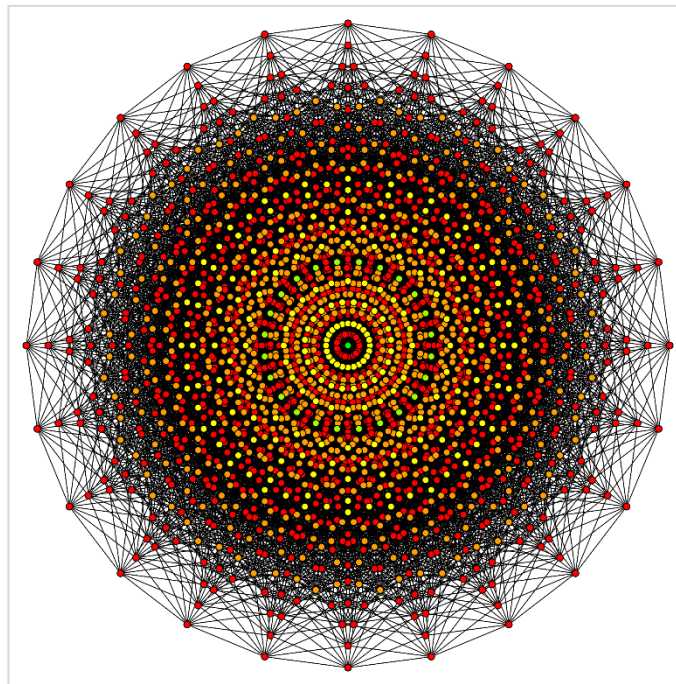
층화 K-Fold 교차 검증(Stratified K-Fold CV)



- 전체 데이터의 분포를 고려하여 학습 데이터셋과 검증 데이터셋을 분배
  - 불균형한 데이터를 사용하는 모델 성능을 측정하는데 용이

## 차원의 저주(Curse of Dimensionality)

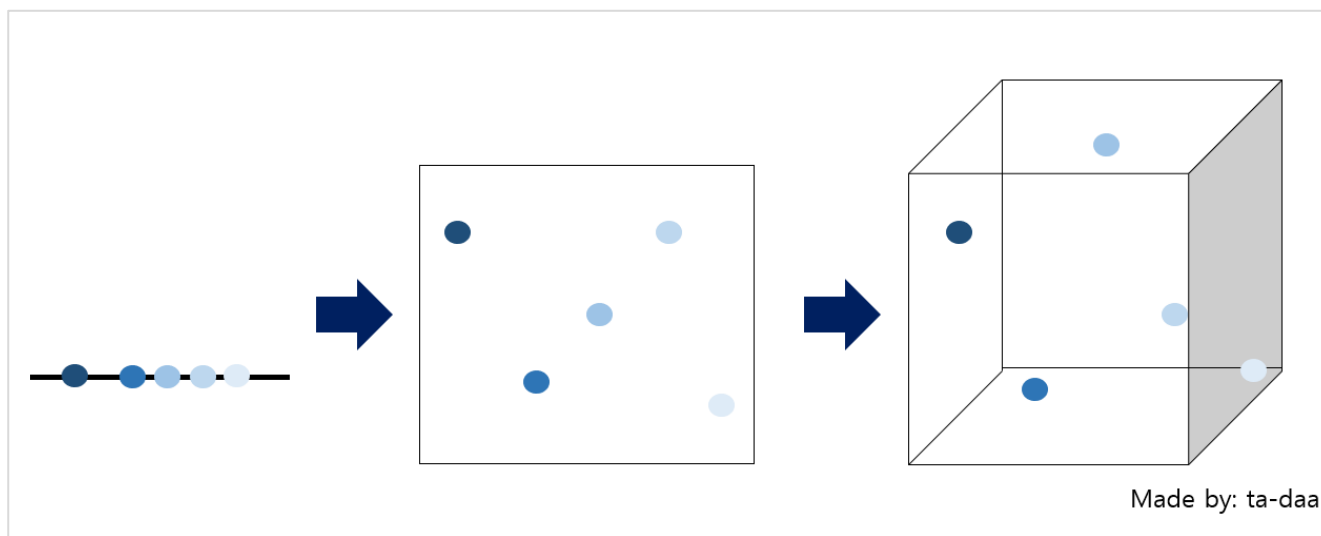
과적합이 발생하는 이유??



독립 변수가 많아 모델에서 고려하는 변수가 많은 경우,  
데이터의 차원이 높은 경우에 과적합이 발생

## 차원의 저주(Curse of Dimensionality)

차원의 저주란??



- 차원이 너무 많아서, 알고리즘 성능이 저하되는 현상
- 데이터셋이 고차원 공간을 갖고 있다면 데이터 간 거리가 멀어져 비슷한 특징을 가진 패턴을 찾기 어려움

## 차원의 저주(Curse of Dimensionality)

차원의 저주란??



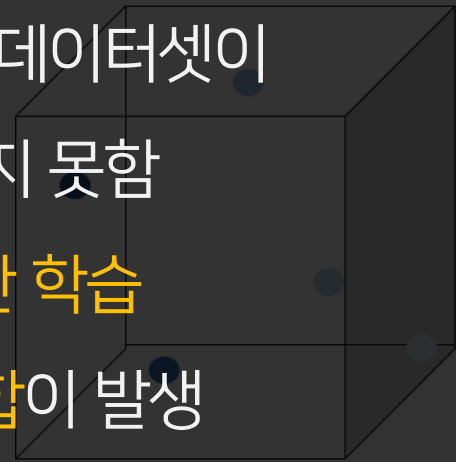
데이터가 너무 **고차원**이라 훈련 데이터셋이

충분히 전체 공간을 나타내지 못함

→ 학습 모델이 **특정부분만 학습**

→ 특정 부분에 대해서 **과적합**이 발생

→ **예측력 감소**



Made by: ta-daa

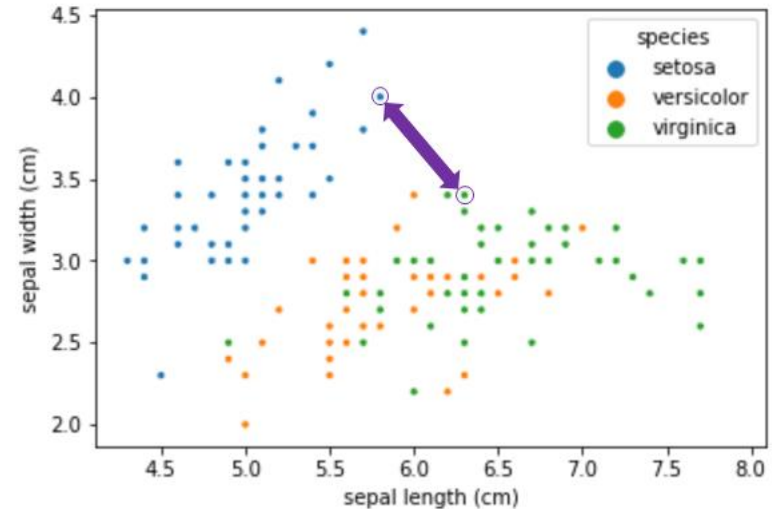
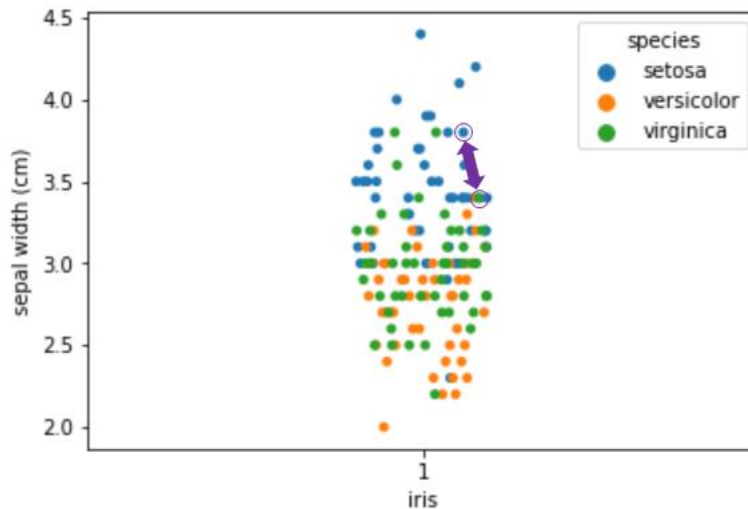
차원이 너무 많아서, 알고리즘 성능이 저하되는 현상

데이터셋이 고차원 공간을 갖고 있다면 데이터 간 거리가 멀어져

비슷한 특징을 가진 패턴을 찾기 어려움

## 차원의 저주(Curse of Dimensionality)

KNN의 예시: Iris Dataset



차원이 커지면 근접한 이웃(데이터)의 거리가 점점 멀어짐

→ 차원이 매우 커질 경우, 데이터 간의 거리는 상당히 증가



## 차원축소

Feature Selection과 Feature Extraction

### 변수선택

Feature Selection

데이터의 특성을  
가장 잘 설명하는  
변수를 **추가**하거나  
**제거**해가며 모델을 적합시킴

- Feedforward Selection
- Backward Elimination
- Stepwise Selection



### 변수추출

Feature Extraction

데이터의 차원을  
고차원에서 **저차원**으로  
**변환**함으로써  
모델을 적합시킴

- PCA(Principal Component Analysis)
- LDA(Linear Discriminant Analysis)
- SVD(Singular Value Decomposition)

## 차원축소

Feature Selection과 Feature Extraction

### 변수선택

Feature Selection

데이터의 특성을  
가장 잘 설명하는  
변수를 **추가**하거나  
**제거**해가며 모델을 적합시킴

V/S

### 변수추출

Feature Extraction

데이터의 차원을  
고차원에서 **저차원**으로  
**변환**함으로써  
모델을 적합시킴

• Feedforward Selection

• Backward Elimination

• Stepwise Selection

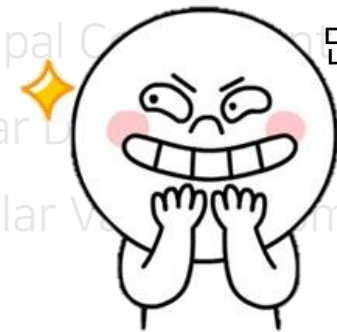
자세한 내용은 클린업 기간동안

회귀분석팀, 선형대수학팀에서 자세히 다룰 예정!

• PCA(Principal Component Analysis) 많은관부~

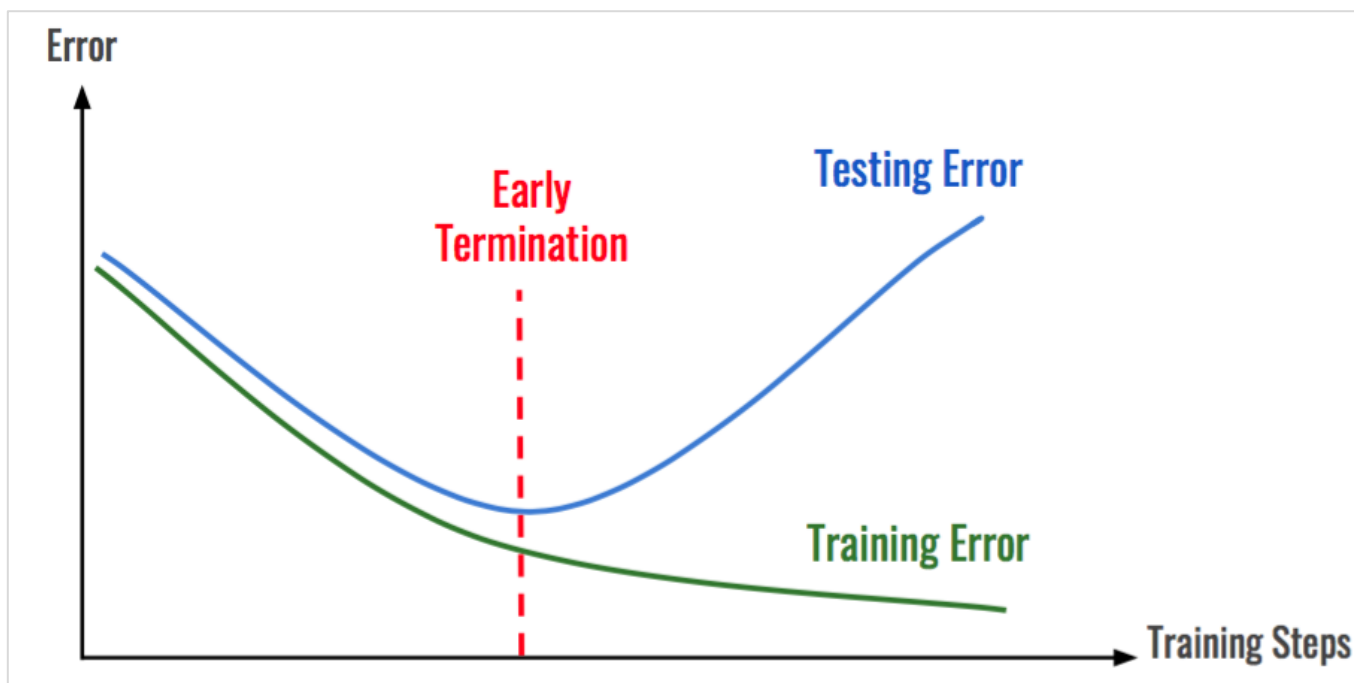
• LDA(Linear Discriminant Analysis)

• SVD(Singular Value Decomposition)



## 학습 관점에서의 해결책

Early Stopping: 학습 조기 종료



학습을 진행할 때 소요되는 시간에 제한,  
혹은 모델의 성능이 일정 수준에 도달하게 되면 학습을 조기에 종료

THANK YOU

