

# 데이터마이닝팀

4팀

김수빈  
조건우  
김보현  
이지원  
조성우

# CONTENTS

1. 데이터마이닝

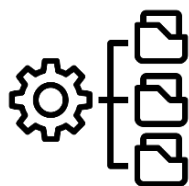
2. 모델링

3. 과적합 방지법

1

데이터마이닝

## 데이터마이닝의 어원



DATA

관찰·측정을 통해 수집된 사실, 값, 문자



MINING

"광물을 캐다"라는 의미로 중요한 정보를 채굴한다는 의미

즉, 대량의 데이터로부터

**유용한 정보와 패턴을 추출**해내는 과정

## 데이터마이닝의 과정



### Exploration

데이터를 분석 가능한 형태로 **전처리**



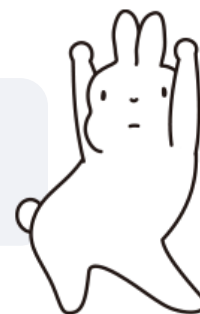
### Pattern Identification

데이터로부터 **패턴**을 찾아냄

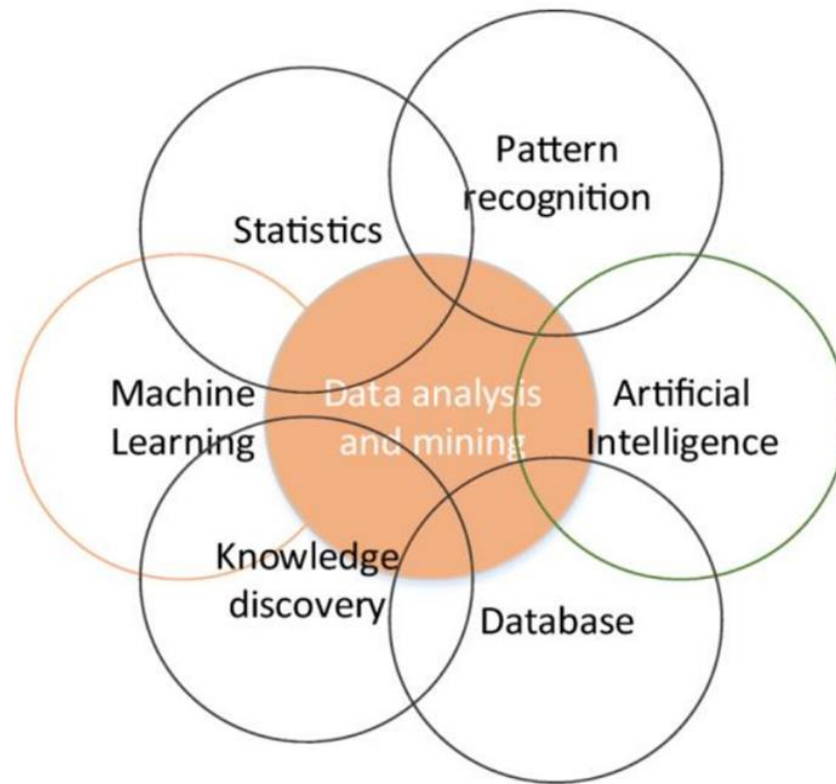


### Deployment

추출해낸 패턴을 이용하여 목적에 맞게 활용하여 **새로운 정보** 발견



## 데이터마이닝의 간학문적 성격



데이터마이닝은 **여러 학문들의 경계**를 넘나들며  
데이터 전처리, 모델링 학습, 평가 등을 진행

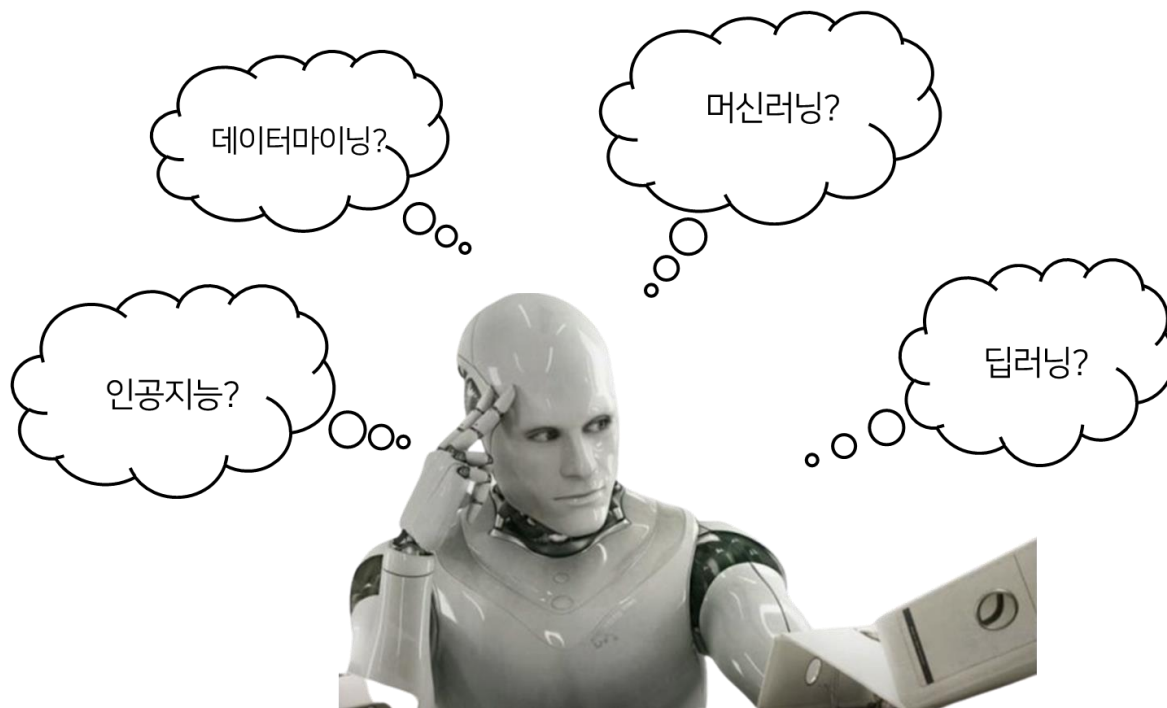


## 인공지능? 머신러닝? 딥러닝?



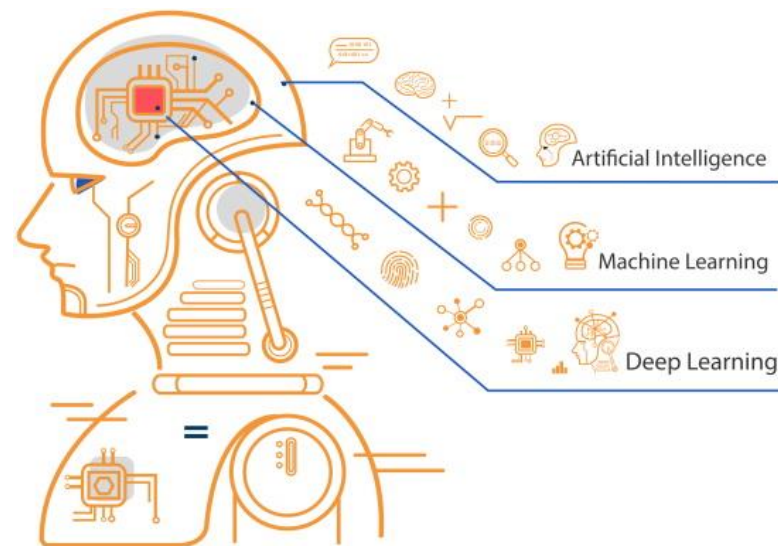
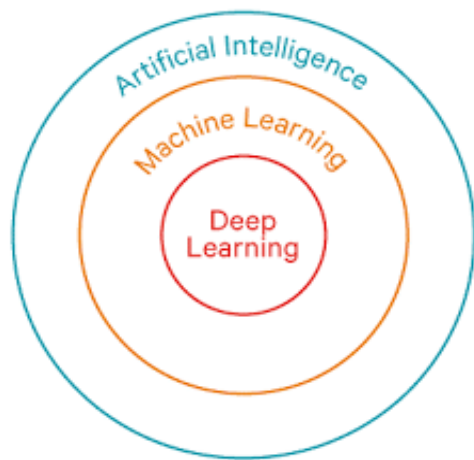
데이터마이닝의 간학문적 성격으로 인해

**다른 개념과의 혼동** 존재



## 인공지능? 머신러닝? 딥러닝?

인공지능(Artificial Intelligence)



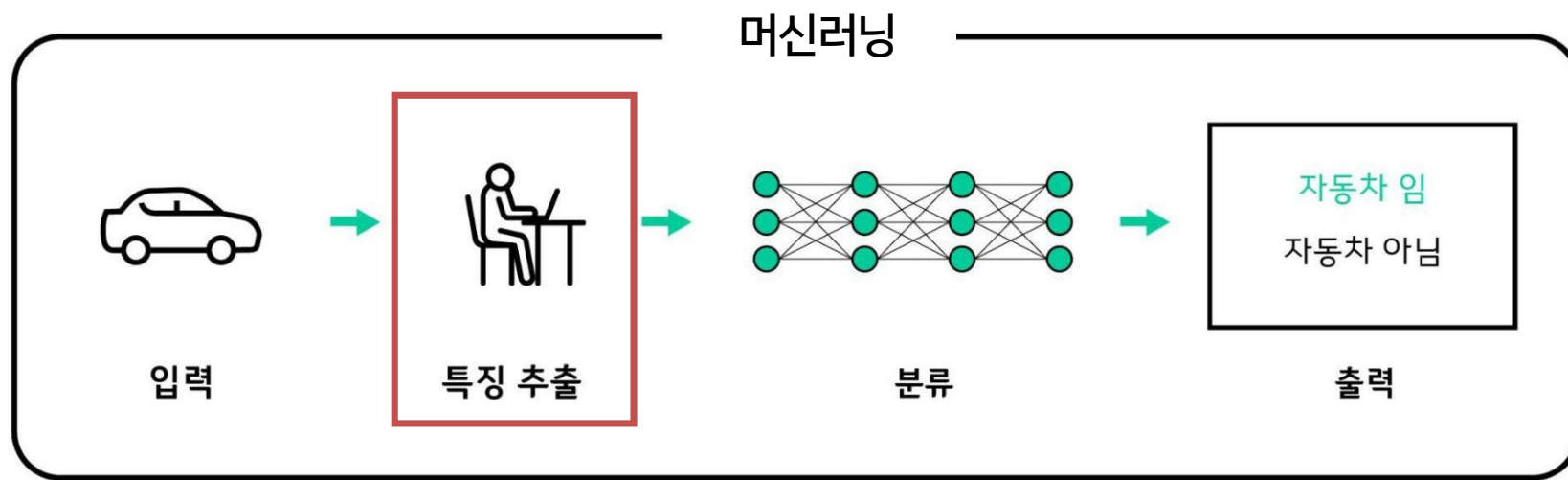
컴퓨팅을 이용한 학습과정을 모두 포함하는 **포괄적인 개념**으로  
머신러닝과 딥러닝을 모두 포함





## 인공지능? 머신러닝? 딥러닝?

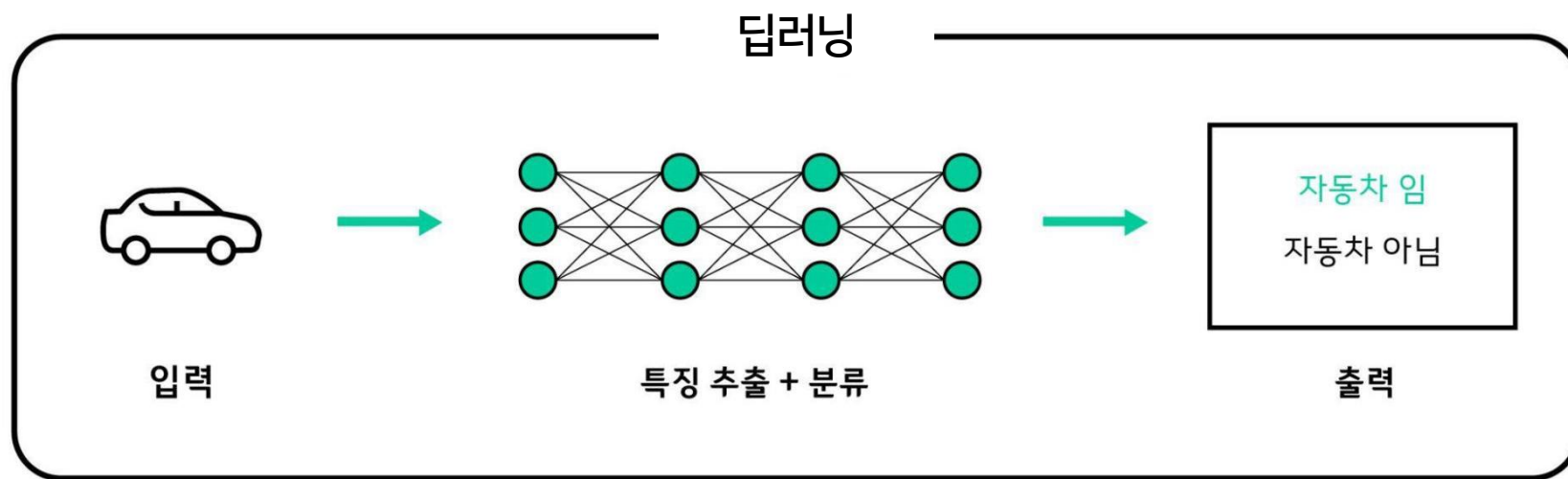
머신러닝(Machine Learning)



**사람의 개입이 최소화**된 학습 수행 방법으로  
적절한 모델을 선정하면 컴퓨터가 스스로 데이터를 학습 후 결과 도출

## 인공지능? 머신러닝? 딥러닝?

딥러닝(Deep Learning)



사람의 **신경망과 유사**한 학습 체계를 구축해 목적 달성을 위한 과정 수행  
도출된 결과 해석이 어려워 '블랙박스 모델'이라 불림

**한블리**  
한문철의 블랙박스 리뷰



## 데이터마이닝의 목표

데이터마이닝 vs ML/DL



머신러닝과 딥러닝이 "수행 과정"에 초점을 둔다면  
데이터마이닝은 이에 더해 "인사이트를 얻어내는 것"을 목표

## 데이터마이닝의 목표

데이터마이닝 vs ML/DL



머신러닝과 딥러닝이 "수행 과정"에 초점을 둔다면  
데이터마이닝은 이에 더해 "인사이트를 얻어내는 것"을 목표



인사이트를 잘 얻어내기 위해선 어떻게 해야 할까



## 데이터마이닝의 목표

데이터마이닝 vs ML/DL



머신러닝과 딥러닝이 "수행 과정"에 초점을 둔다면  
데이터마이닝은 이에 더해 "인사이트를 얻어내는 것"을 목표



인사이트를 잘 얻어내기 위해선 어떻게 해야 할까



모델 선택이 적절했는가?

그 모델이 목표를 얼마나 잘 달성했는가?

선택한 모델의 성능을 높이는 방법은 무엇인가?

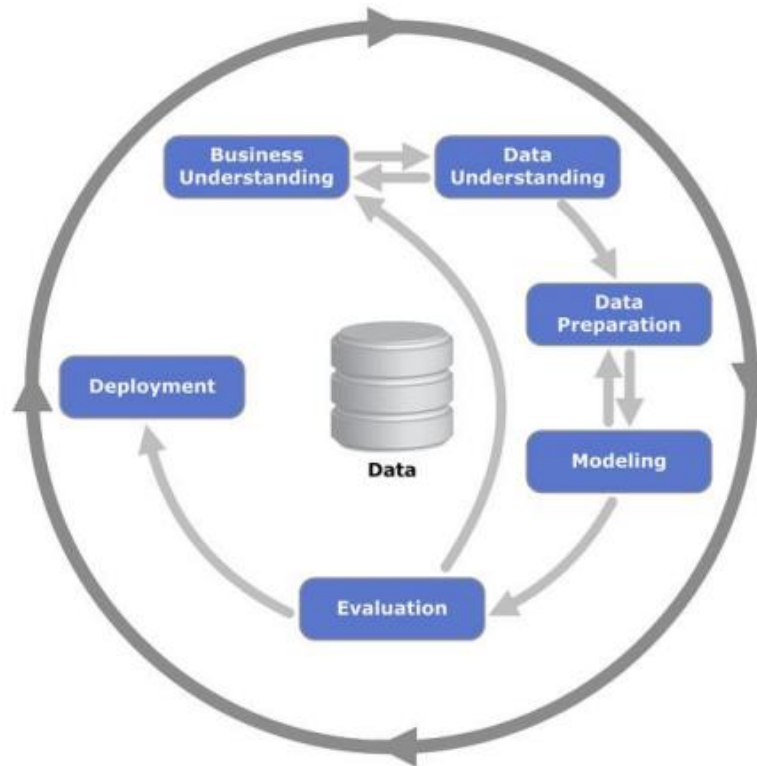


질문의 해답은...  
통계학!

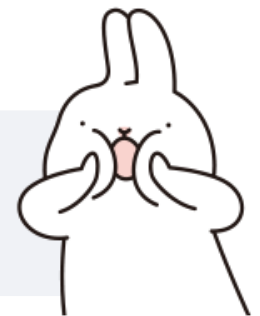


## CRISP-DM 방법론

CRISP-DM이란



데이터마이닝의 대표적인 분석 과정으로 크게 **6단계**로 구성



## CRISP-DM 방법론

### 1단계) 비즈니스 문제 이해

과제의 목적과 요구사항을 이해하는 과정으로  
도메인 지식을 활용해 초기 프로젝트 계획을 수립하는 단계

→ 배경지식 조사 & 평가 기준 설정 필요

### 2단계) 데이터 이해(EDA)

분석을 위해 데이터를 수집하고  
이에 대한 직관적인 이해가 수반되는 단계



각 변수들의 통계량 및 이상치, 결측치 확인을 통해  
변수 분포, 추이, 상관관계 등 시각화

## CRISP-DM 방법론

### 1단계) 비즈니스 문제 이해

과제의 목적과 요구사항을 이해하는 과정으로  
도메인 지식을 활용해 초기 프로젝트 계획을 수립하는 단계

→ 배경지식 조사 & 평가 기준 설정 필요

### 2단계) 데이터 이해(EDA)

분석을 위해 데이터를 수집하고  
이에 대한 직관적인 이해가 수반되는 단계





각 변수들의 통계량 및 **이상치**, **결측치** 확인을 통해  
변수 분포, 추이, 상관관계 등 시각화



## CRISP-DM 방법론

### 3단계) 데이터 준비

수집한 데이터를 분석 목적에 맞게 데이터 전처리하는 단계

전처리에 따라 모델 성능이 달라지므로 상당히 중요  

### 4단계) 분석 및 모델링

모델링 과정을 수행하고  
파라미터를 최적화해 나가는 단계



모델링 기법 선택, 모델 테스트 계획 설계 등이 이루어짐

## CRISP-DM 방법론

### 3단계) 데이터 준비

수집한 데이터를 분석 목적에 맞게 데이터 **전처리**하는 단계

전처리에 따라 모델 성능이 달라지므로 상당히 중요



### 4단계) 분석 및 모델링

모델링 과정을 수행하고  
파라미터를 최적화해 나가는 단계



모델링 기법 선택, 모델 테스트 계획 설계 등이 이루어짐

## CRISP-DM 방법론

### 5단계) 평가

모델링이 잘 되었는지 평가하는 단계로  
모델에 따라, 목적에 따라 각기 다른 **평가지표**로 평가

과제 목적에 알맞은 평가 지표를 사용하는 것이 중요 

ex) 분류문제 - F1 score / 회귀문제 - RMSE

### 6단계) 전개

분석한 결과를 적용하여 유의미한 결론을 이끌어내는 과정

분석 내용을 바탕으로 현실 사회의 문제에 해결책 제시 

## CRISP-DM 방법론

### 5단계) 평가


모델링이 잘 되었는지 평가하는 단계로  
모델에 따라, 목적에 따라 각기 다른 평가지표로 평가

과제 목적에 알맞은 평가 지표를 사용하는 것이 중요 

ex) 분류문제 - F1-score / 회귀문제 - RMSE

### 6단계) 전개

분석한 결과를 적용하여 **유의미한 결론**을 이끌어내는 과정


분석 내용을 바탕으로 현실 사회의 문제에 해결책 제시 

2

모델링

## Train Data & Test Data

독립변수와 종속변수

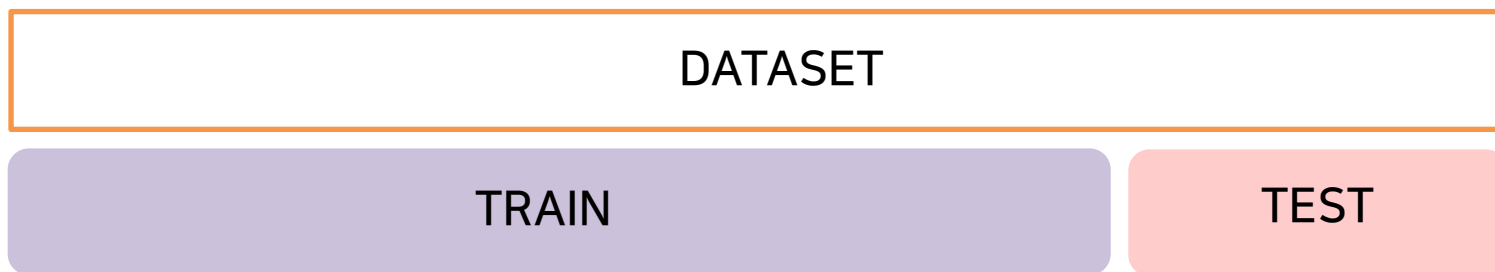


Bedrooms	Sq. feet	Neighborhood	Sales Price
3	2000	Normaltown	\$250,000
2	800	Hipsterton	\$300,000
2	850	Normaltown	\$150,000
1	550	Normaltown	\$78,000
4	2000	Skid Row	\$150,000

독립변수, 종속변수로 이루어진 데이터를 바탕으로 학습 진행하여  
 학습된 모델들은 독립변수가 입력으로 들어오면 종속변수 예측하게끔 설계

## Train Data & Test Data

Definition



Train Data

모델의 학습을 위한 데이터  
종속변수 & 독립변수 모두 존재

V/S

Test Data

목적 달성을 위한 데이터  
종속변수 존재하지 않음

## Train Data & Test Data

Ex. 주택 데이터

[Train data]

Bedrooms	Sq. feet	Neighborhood	Sales Price
3	2000	Normaltown	\$250,000
2	800	Hipsterton	\$300,000
2	850	Normaltown	\$150,000

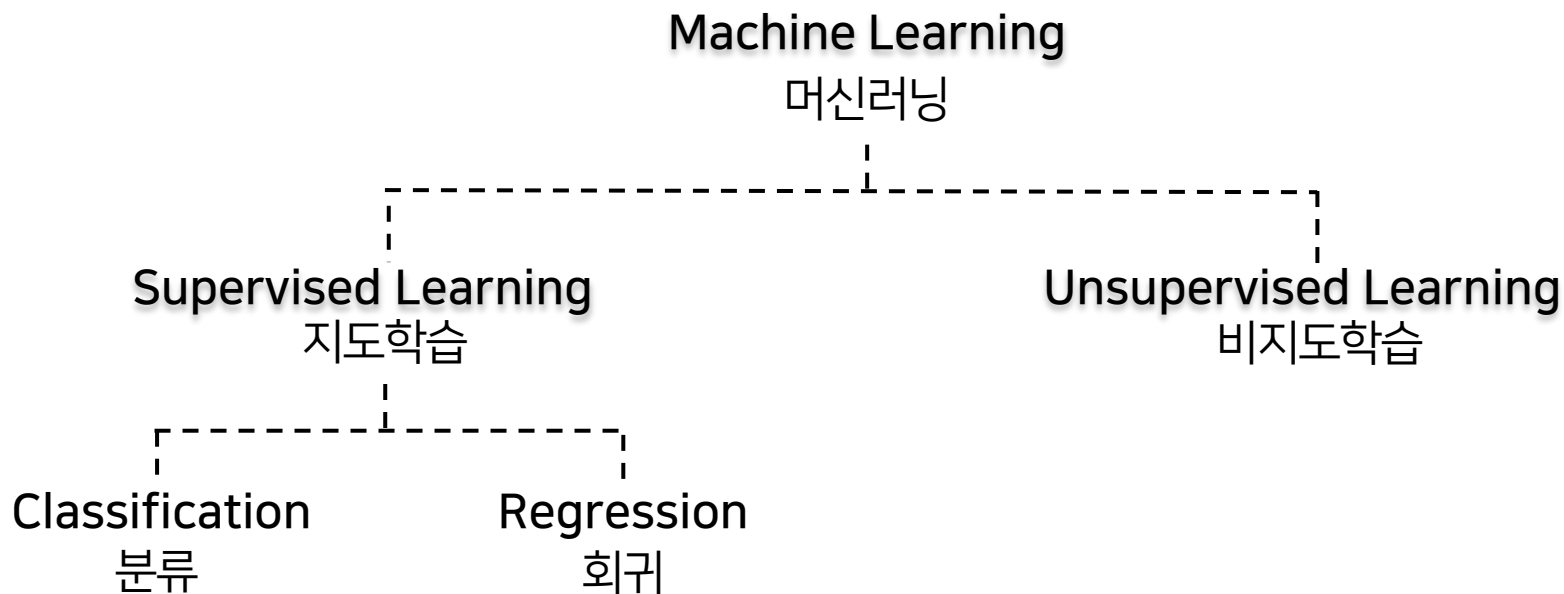
[Test data]

Bedrooms	Sq. feet	Neighborhood	Sales Price
3	2000	Hipsterton	???

Train Data를 통해 학습한 모델로 Test Data의 Sales Prices 예측

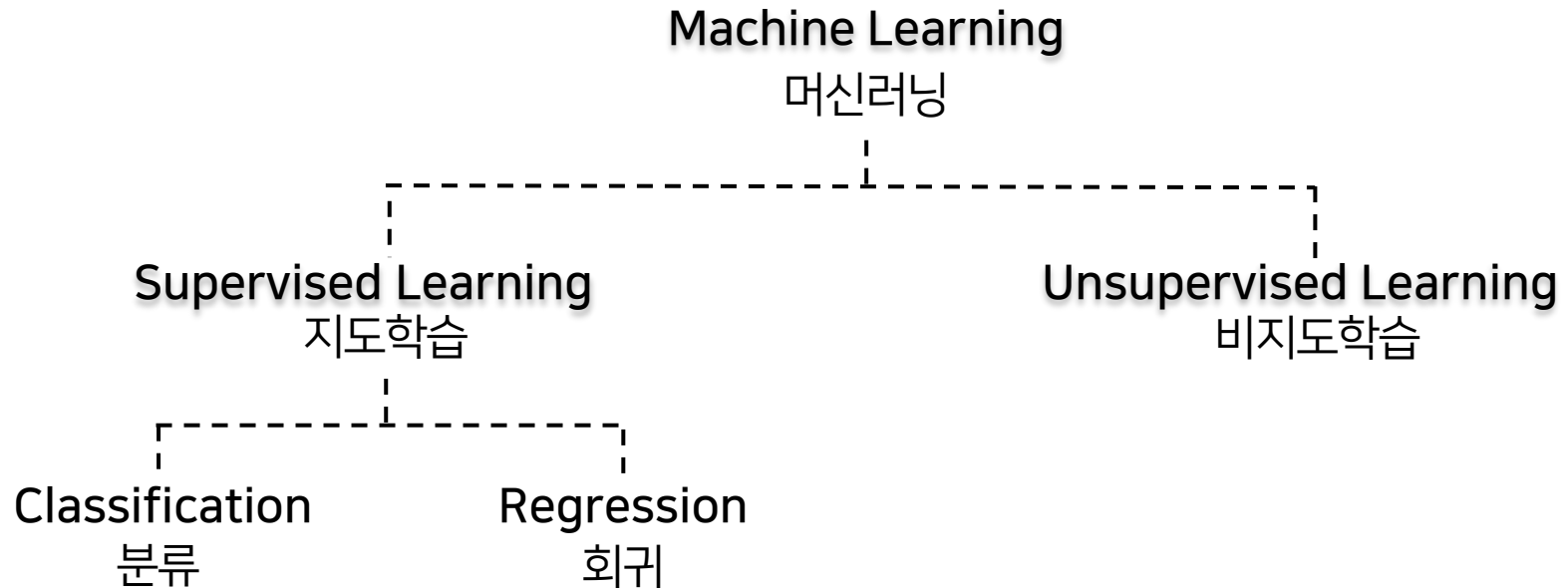


## 머신러닝의 종류



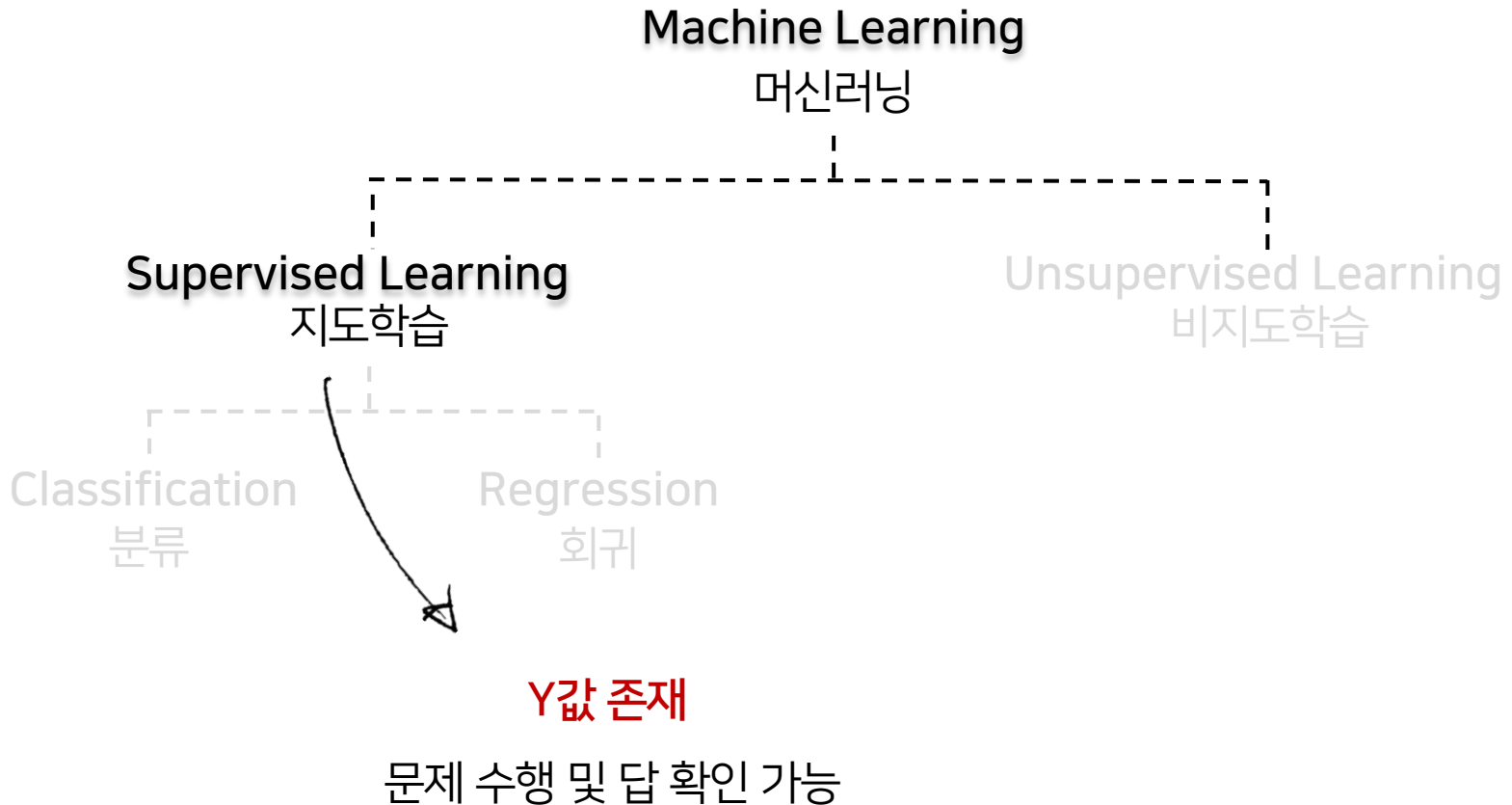
머신러닝은 지도학습과 비지도학습으로 나뉘며  
이 중 지도학습은 목적에 따라 분류와 회귀로 나뉨

## 머신러닝의 종류

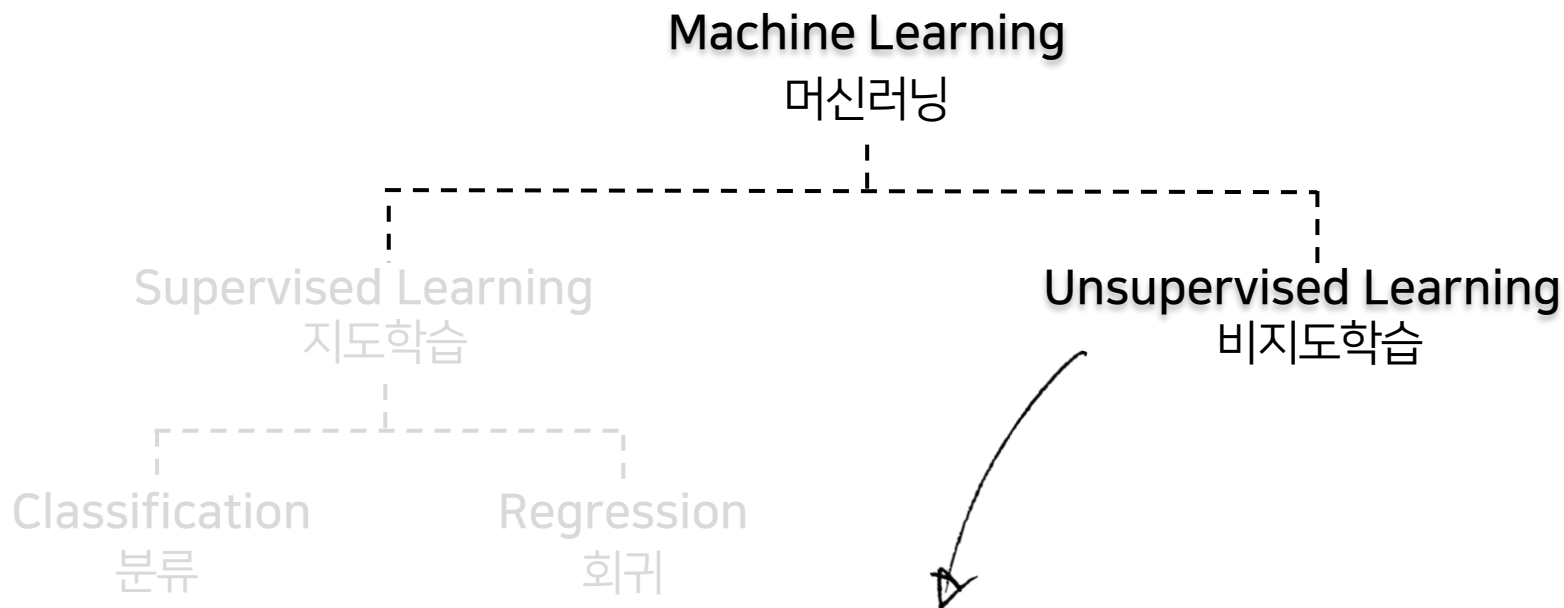


지도학습과 비지도학습은 **데이터라벨(Y값)** 존재 여부에 따라 구분

## 머신러닝의 종류



## 머신러닝의 종류



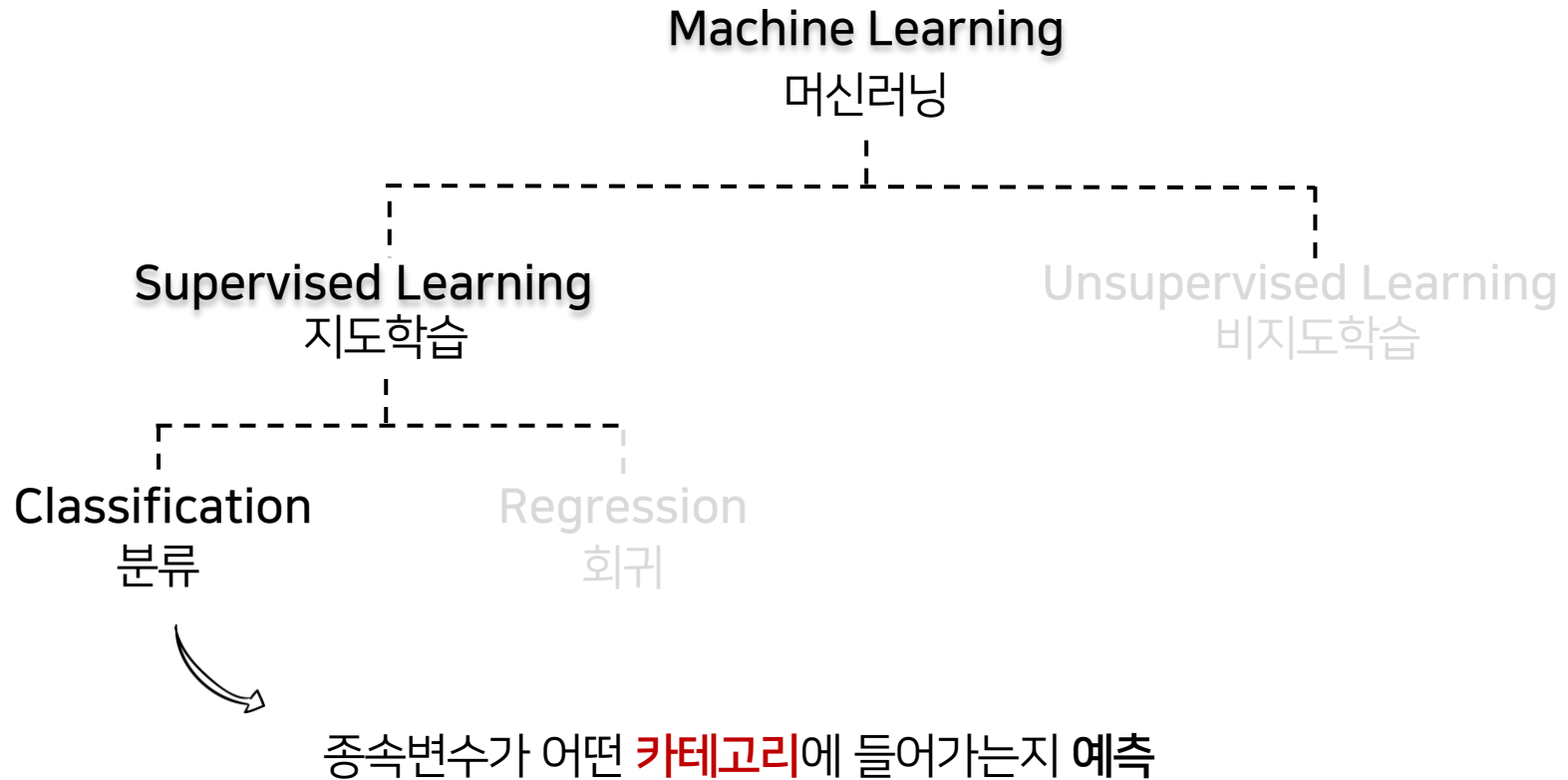
**Y값 존재 X**

데이터의 구조를 묘사하고 관계를 해석하는 데 초점

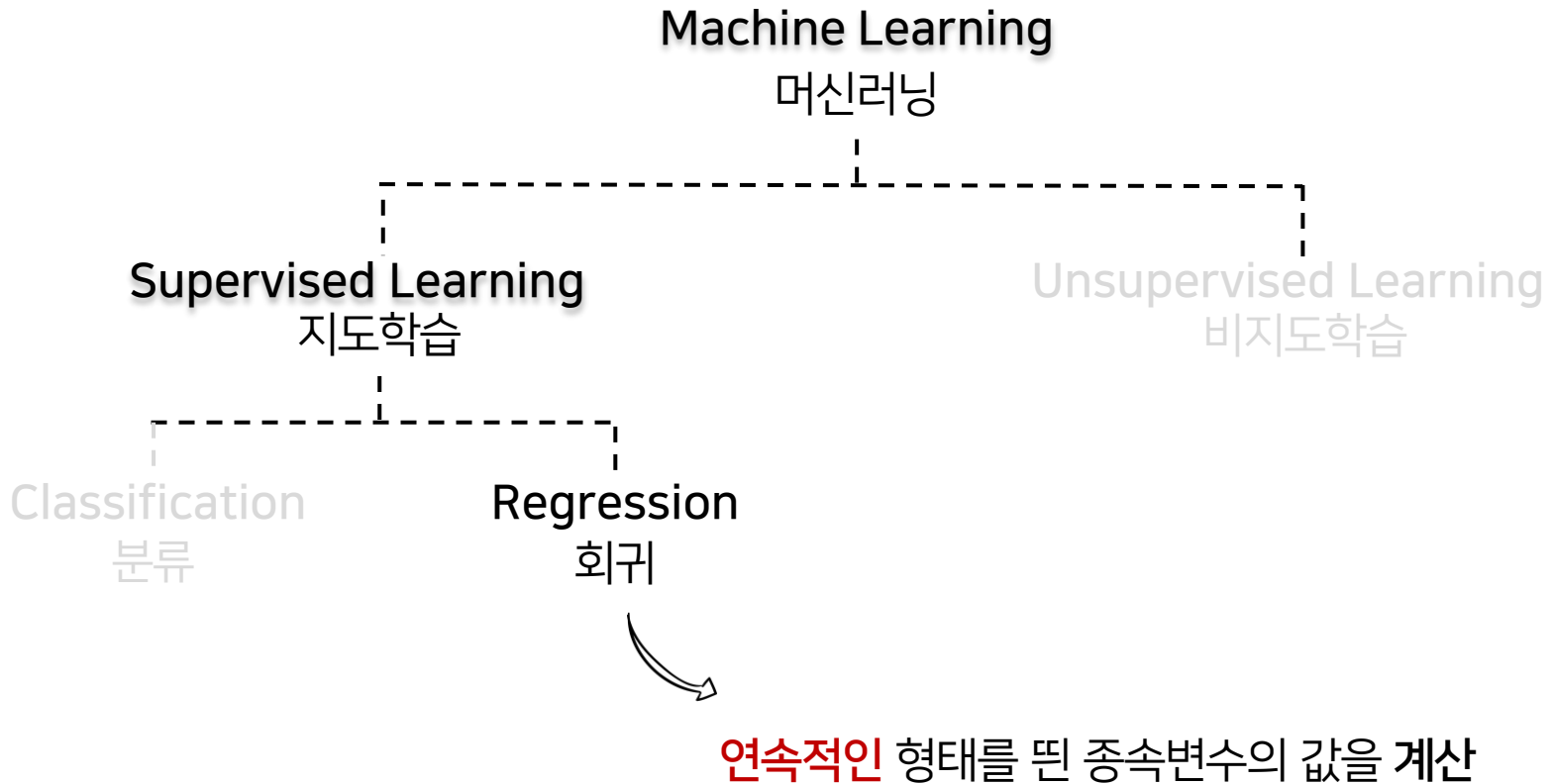
**연구자의 주관 개입 여지 큼**

Ex) 클러스터링, 주성분 분석(PCA)

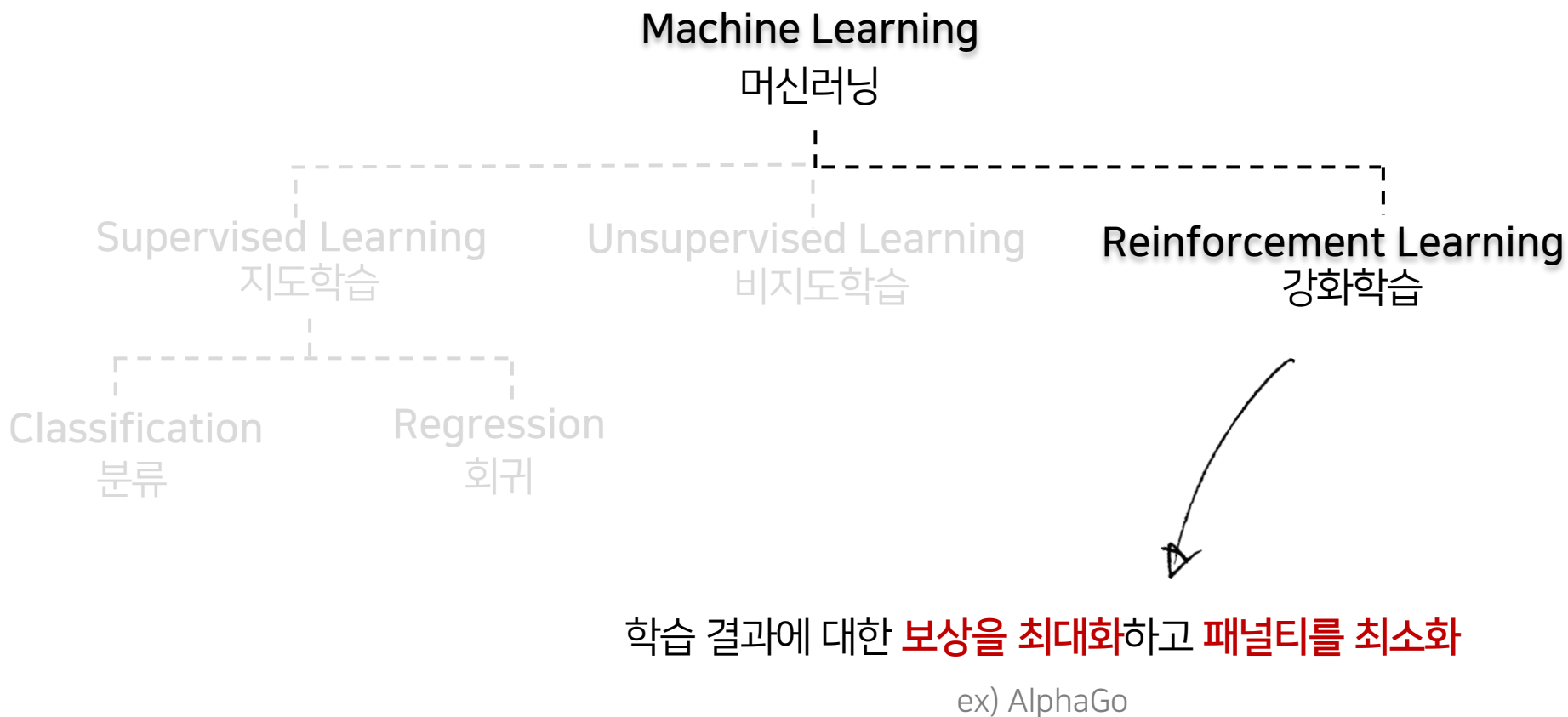
## 머신러닝의 종류



## 머신러닝의 종류



## 머신러닝의 종류



## 지도학습 (Supervised learning)

지도 학습의 기본 형태

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

Model term

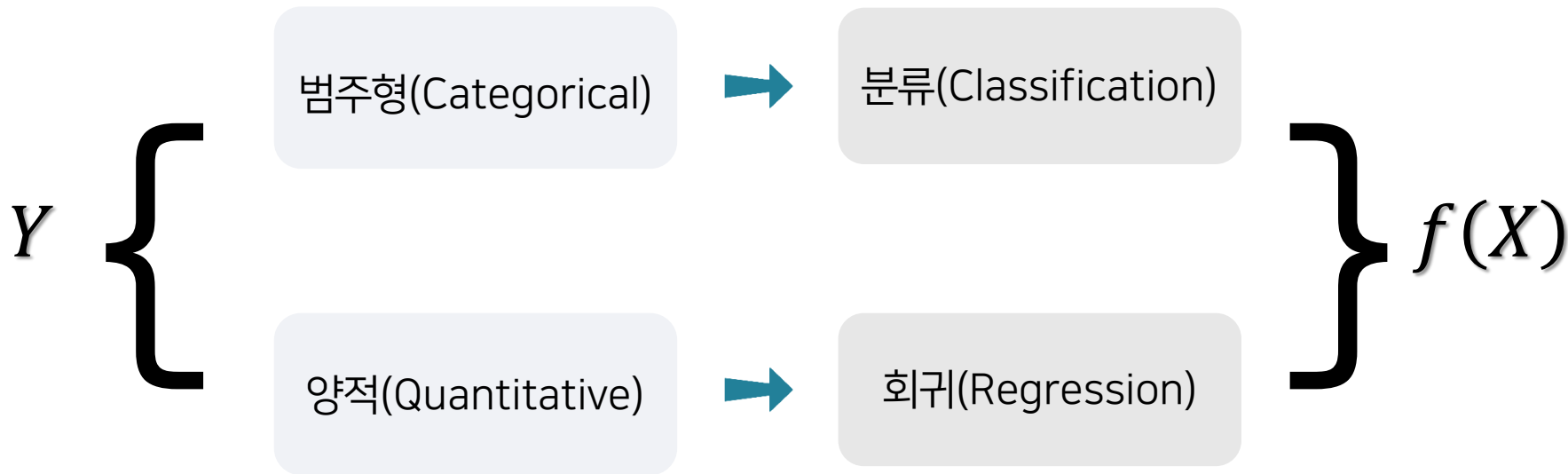
Error term

이런 수학적 모델은 실제 데이터의 관계를 완벽하게 설명하지 못하지만,  
여러모델을 적용하여 실제  $Y$ 값에 근접한 추정치  $\hat{Y}$  을 주는 모델을 선택



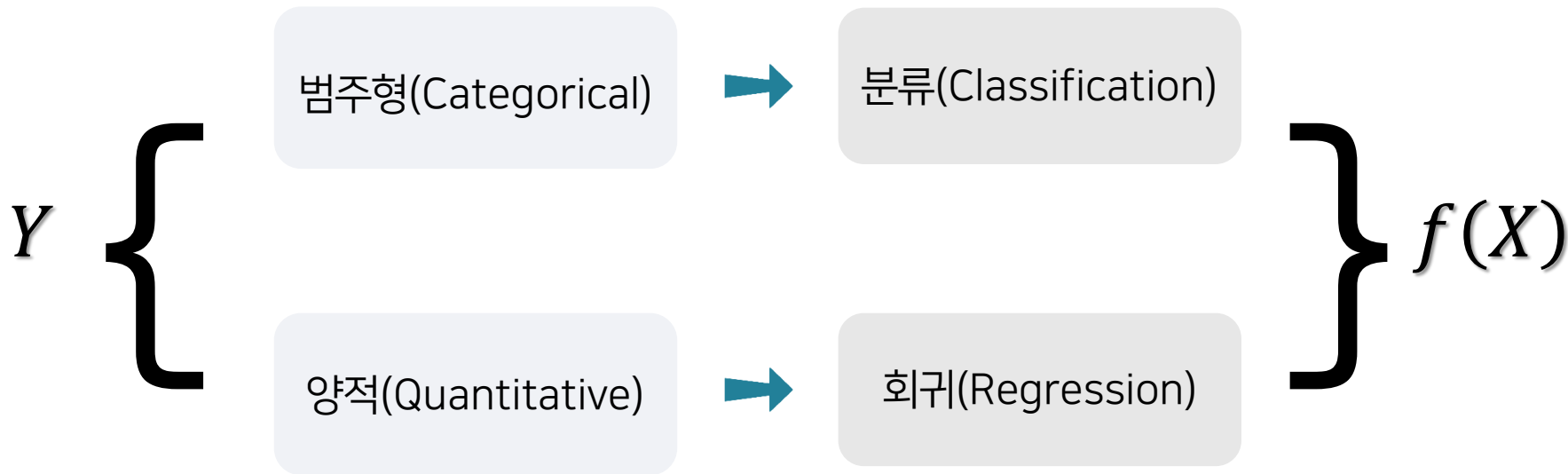
## 지도학습 (Supervised learning)

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



## 지도학습 (Supervised learning)

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



우리의 목표는 실제  $Y$ 값에  
가까운 예측을 하는 모델을 찾는 것



Hyperparameter Tuning!

## 지도학습 (Supervised learning)

### Parameter

주어진 데이터에 대한 특성을 보여주는 모수를 의미

### Hyperparameter

모델의 매개변수이며 어떤 값을 넣느냐에 따라 모델의 성능이 달라짐



Hyperparameter Tuning을 통해 최적의 모델 판단 필요

## 지도학습 (Supervised learning)


MSE decomposition

----- MSE(Mean Squared Error) -----

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

↓

MSE를 줄여 모델의 성능을 높일 수 있음




## 지도학습 (Supervised learning)

MSE decomposition

MSE(Mean Squared Error)

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

MSE를 줄여 모델의 성능을 높일 수 있음



왜 MSE인가?

(실제값 - 추정값)을 사용할 경우 음수 값이 나올 수 있고  
잔차의 합은 0이므로 오차의 제곱인 MSE 사용


## 지도학습 (Supervised learning)

MSE decomposition

----- MSE(Mean Squared Error) -----

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

MSE를 줄여 모델의 성능을 높일 수 있음



MSE 줄이는 방법?

MSE Decomposition을 통해 알아보자! 

## 지도학습 (Supervised learning)

MSE decomposition

$$\begin{aligned} E[(y - \hat{f})^2] &= E[(f + \epsilon - \hat{f})^2] = E[(f + \epsilon - \hat{f} + E[\hat{f}] - E[\hat{f}])^2] \\ &= E\left[\left((E[\hat{f}] - \hat{f}) + (f - E[\hat{f}]) + \epsilon\right)^2\right] \\ &= E\left[(f - E[\widehat{f}])^2\right] + E[\epsilon^2] + E\left[(E[\hat{f}] - \hat{f})^2\right] + 2E[(f - E[\hat{f}])\epsilon] \\ &\quad + 2E[\epsilon(E[\hat{f}] - \hat{f})] + 2E[(E[\hat{f}] - \hat{f})(f - E[\hat{f}])] \\ &= (f - E[\widehat{f}])^2 + E[\epsilon^2] + E\left[(E[\hat{f}] - \hat{f})^2\right] \\ &= (f - E[\widehat{f}])^2 + \text{Var}[\epsilon] + \text{Var}[\hat{f}] = \text{Bias}[\widehat{f}]^2 + \text{Var}[\epsilon] + \text{Var}[\hat{f}] \\ &= \text{Bias}[\widehat{f}]^2 + \sigma^2 + \text{Var}[\hat{f}] \end{aligned}$$

## 지도학습 (Supervised learning)

MSE decomposition

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



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

복잡한 유도과정은 너굴맨이 처리했으니 안심하라구~



## 지도학습 (Supervised learning)

MSE decomposition

$$E[(y - \hat{y})^2] = (Bias[\widehat{f}]^2 + var[\hat{f}]) + \sigma^2$$

## 지도학습 (Supervised learning)

MSE decomposition

$$E[(y - \hat{y})^2] = (Bias[\widehat{f}]^2 + var[\hat{f}]) + \sigma^2$$



Irreducible Error

표본 추출 과정 등  
randomness에 의해 발생하는 오차

## 지도학습 (Supervised learning)

MSE decomposition

$$E[(y - \hat{y})^2] = (Bias[\widehat{f}]^2 + var[\hat{f}]) + \sigma^2$$

Reducible Error

모델의 Bias와 모델의 Variance으로 이루어짐



## 지도학습 (Supervised learning)

MSE decomposition

$$E[(y - \hat{y})^2] = (Bias[\widehat{f}]^2 + var[\hat{f}]) + \sigma^2$$

### Bias

추정된 모델이 실제 모델을 얼마나 잘 설명하는지를 의미

### Variance

추정된 모델이 다른 데이터셋을 적합했을 때 얼마나 달라지는지를 의미

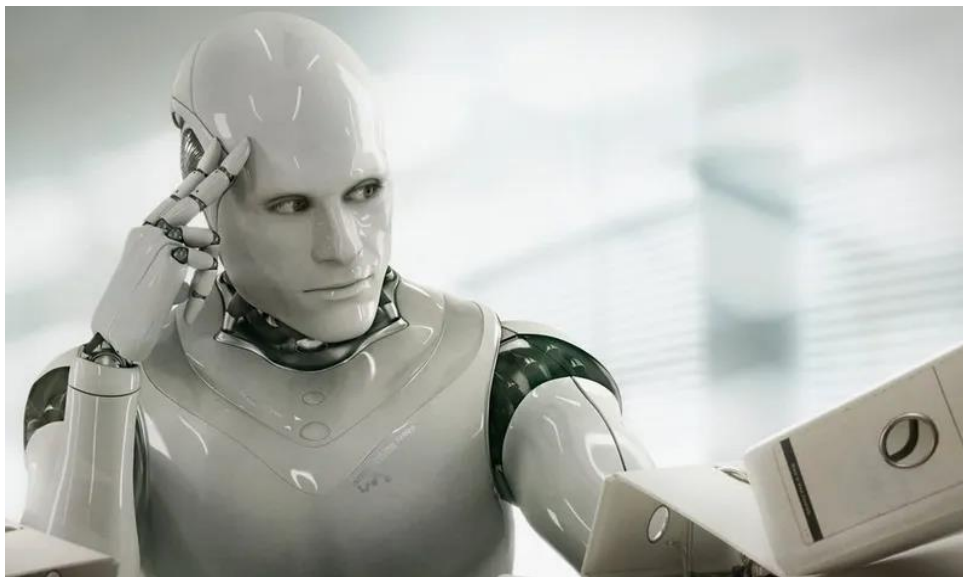


## 지도학습 (Supervised learning)

Variance-Bias Tradeoff



그렇다면 통제가능한 Bias와 Variance를 같이 줄이면  
최적의 모델을 만들 수 있지 않을까?



## 지도학습 (Supervised learning)

Variance-Bias Tradeoff



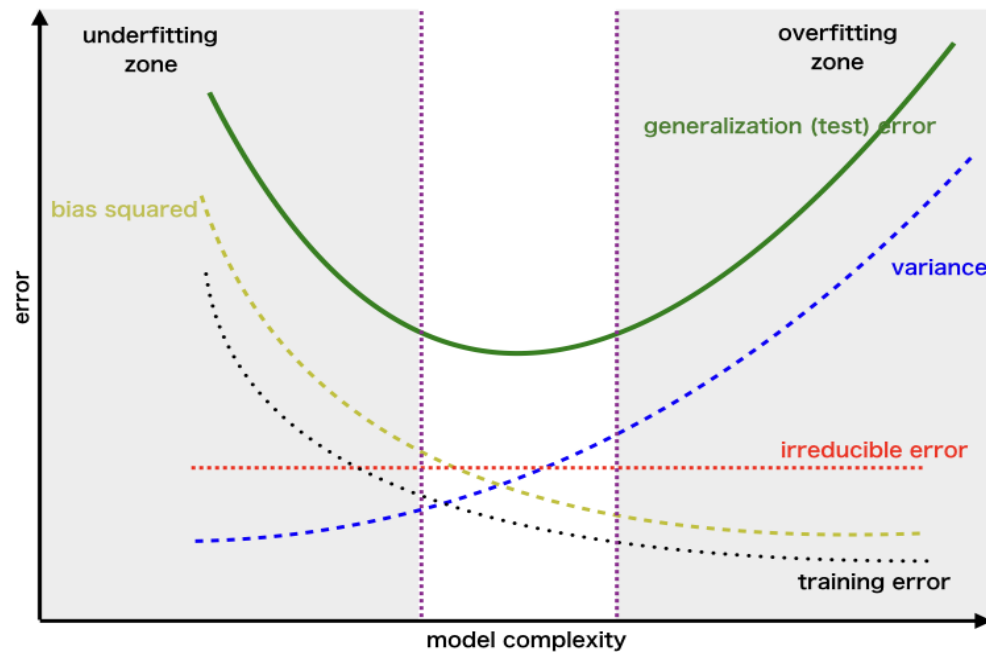
그렇다면 통제가능한 Bias와 Variance를 같이 줄이면  
최적의 모델을 만들 수 있지 않을까?

**그게 가능했으면 이미 그렇게 했겠죠?**



## 지도학습 (Supervised learning)

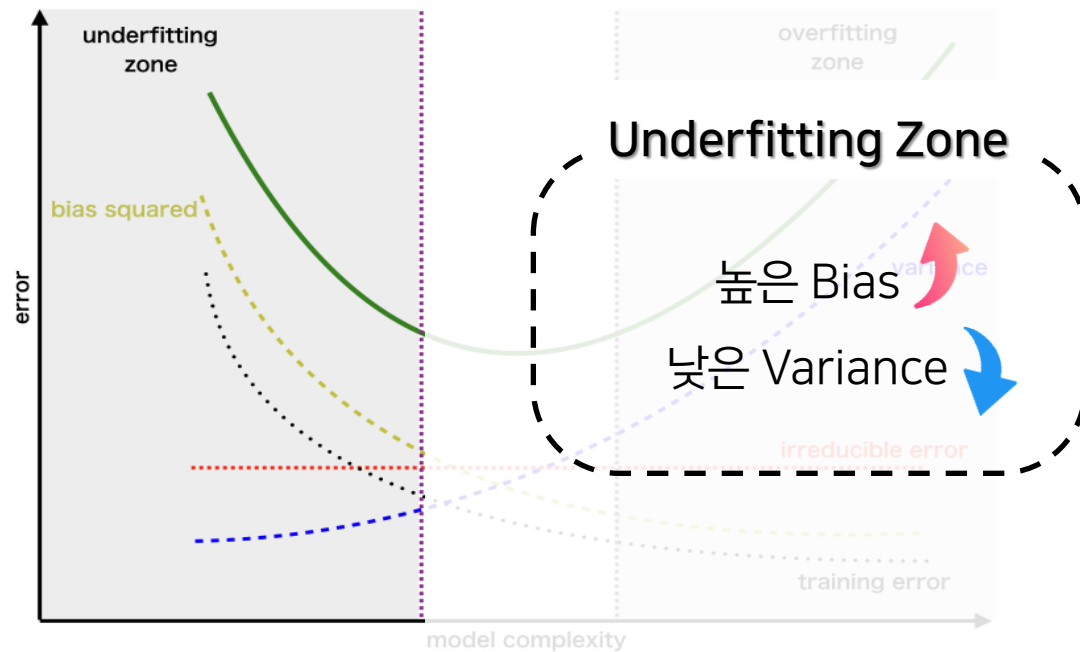
### Variance-Bias Trade-off



모델의 Bias와 Variance는 서로 반대 방향으로 움직이며  
이를 **Trade-Off** 관계라 표현

## 지도학습 (Supervised learning)

### Variance-Bias Trade-off

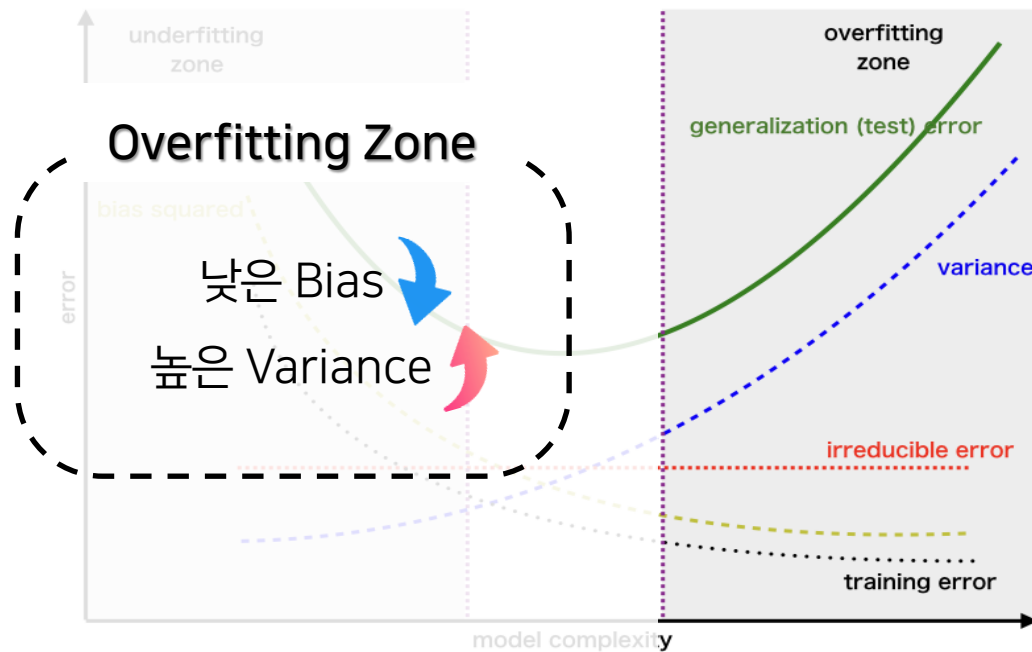


모델의 Bias가 높고 Variance가 낮은 경우,  
과소적합(Underfitting) 발생



## 지도학습 (Supervised learning)

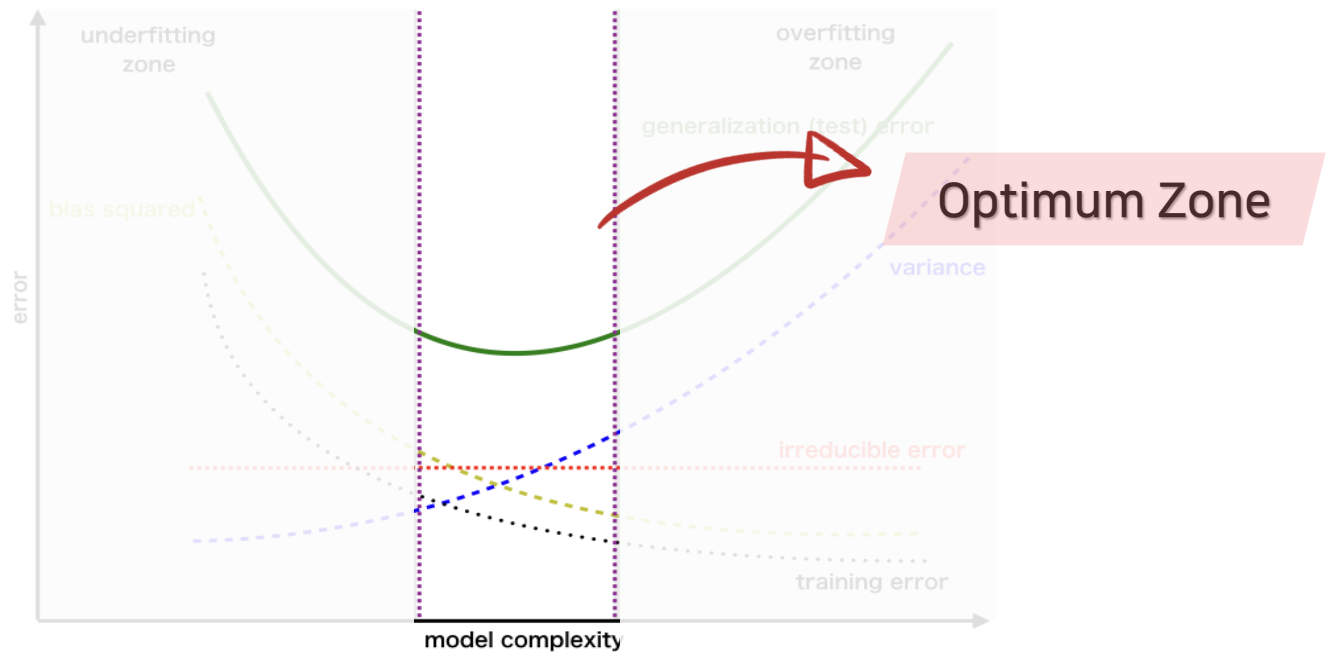
### Variance-Bias Trade-off




모델의 Bias가 낮고 Variance가 높은 경우,  
과대적합(Overfitting) 발생

## 지도학습 (Supervised learning)

### Variance-Bias Trade-off



Bias와 Variance가 적당히 작아

**MSE가 최소**가 되는 model을 찾아내는 것이 관건! 

## 지도학습 (Supervised learning)

KNN(K-Nearest-Neighbor)

모수적 모델(parametric model) VS 비모수적 모델(nonparametric model)

### 모수적 모델

주어진 데이터의 확률분포를 기반으로  
모수를 추정하는 과정이  
포함되어 있는 모델

ex) 단순선형회귀

### 비모수적 모델

모수를 추정하지 않고  
다양한 알고리즘을 사용하는 모델

ex) KNN

## 지도학습 (Supervised learning)

KNN(K-Nearest-Neighbor)

모수적 모델(parametric model) VS 비모수적 모델(nonparametric model)

### 모수적 모델

주어진 데이터의 확률분포를 기반으로  
모수를 추정하는 과정이  
포함되어 있는 모델

ex) 단순선형회귀

### 비모수적 모델

모수를 추정하지 않고  
다양한 알고리즘을 사용하는 모델

ex) KNN

## 지도학습 (Supervised learning)

KNN(K-Nearest-Neighbor)

모수적 모델(parametric model) VS 비모수적 모델(nonparametric model)

### 모수적 모델

주어진 데이터의 확률분포를 기반으로  
모수를 추정하는 과정이  
포함되어 있는 모델  
ex) 단순선형회귀

### 비모수적 모델

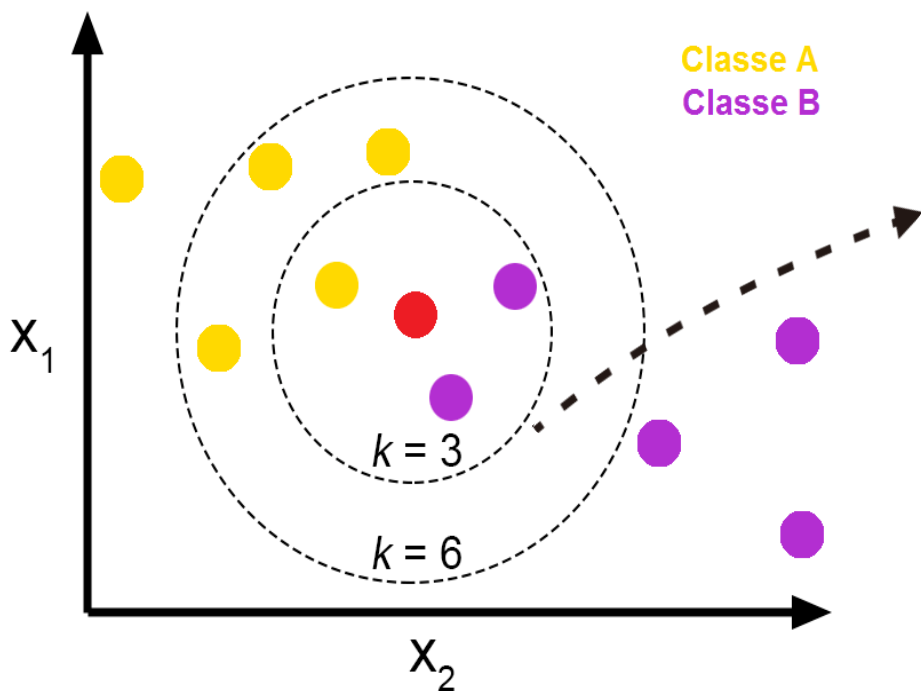
모수를 추정하지 않고  
다양한 알고리즘을 사용하는 모델  
ex) KNN

## 지도학습 (Supervised learning)

KNN(K-Nearest-Neighbor)

KNN(K-Nearest-Neighbor)

대표적인 비모수적인 모델로 K개의 가까운 이웃데이터들 중 다수결로 예측



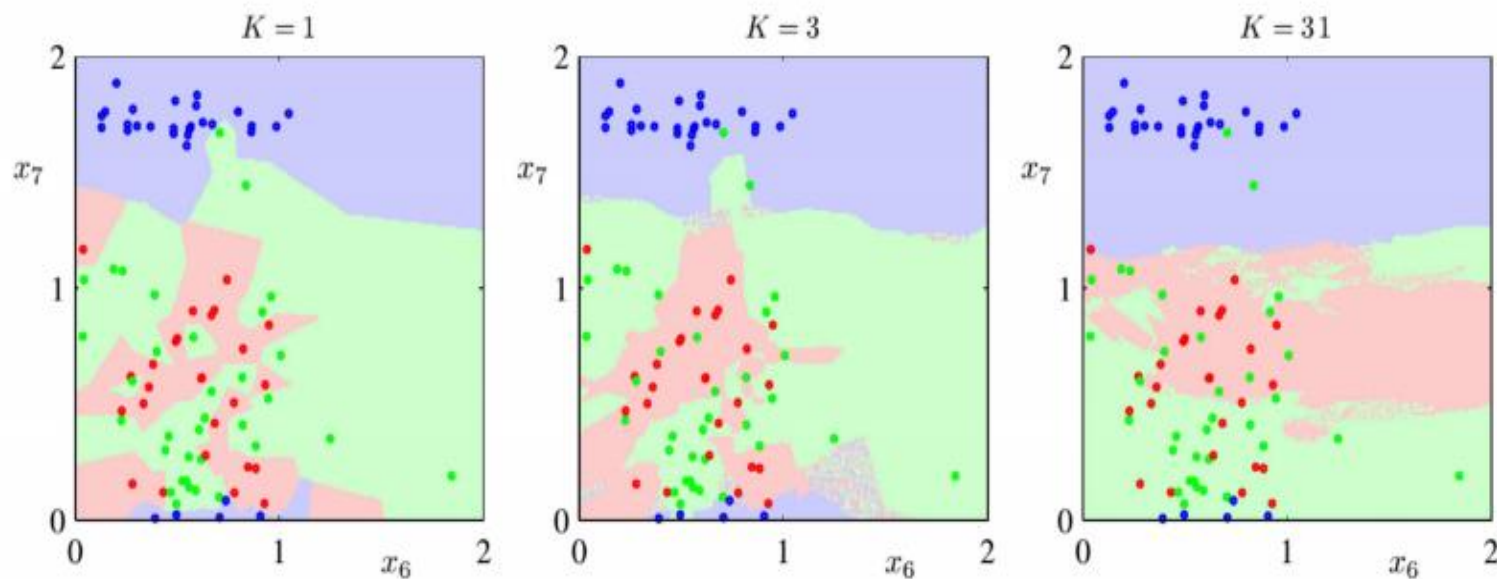
K : Hyperparameter



K의 값에 따라  
Decision Boundary 변화!

## 지도학습 (Supervised learning)

KNN(K-Nearest-Neighbor)

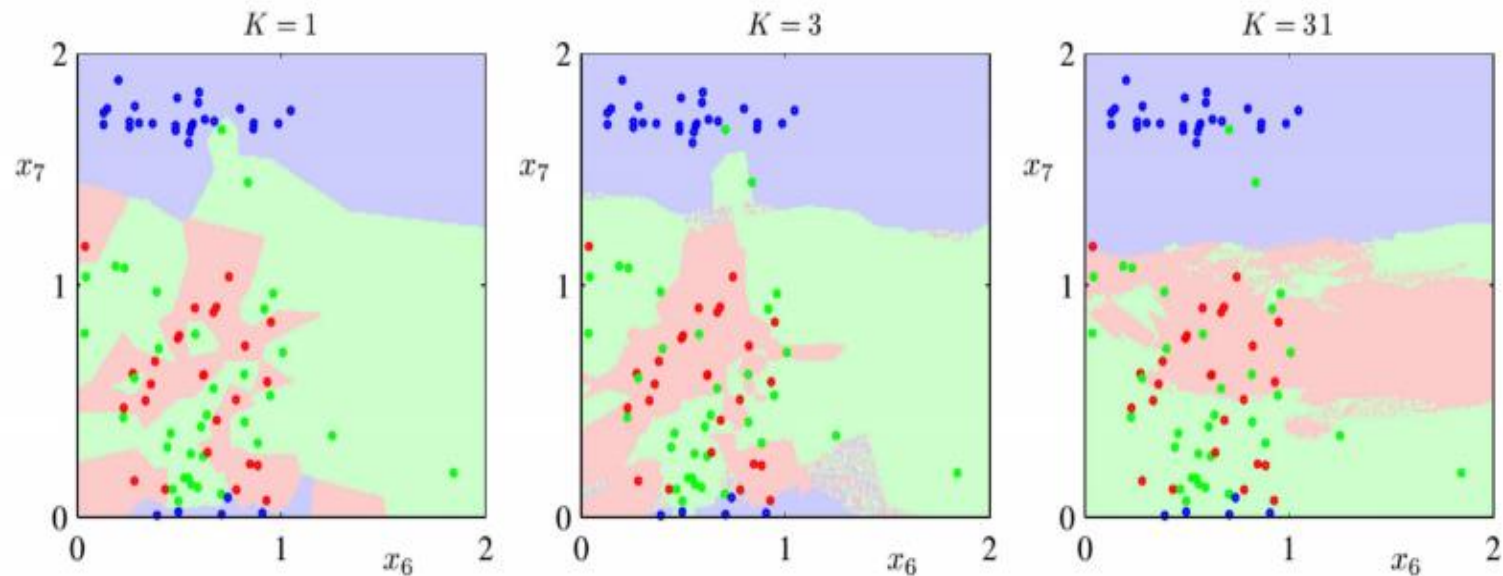


K가 작으면 적은 이웃을 반영하여 예측을 진행하기 때문에

Decision boundary가 복잡해짐 🚨

## 지도학습 (Supervised learning)

KNN(K-Nearest-Neighbor)

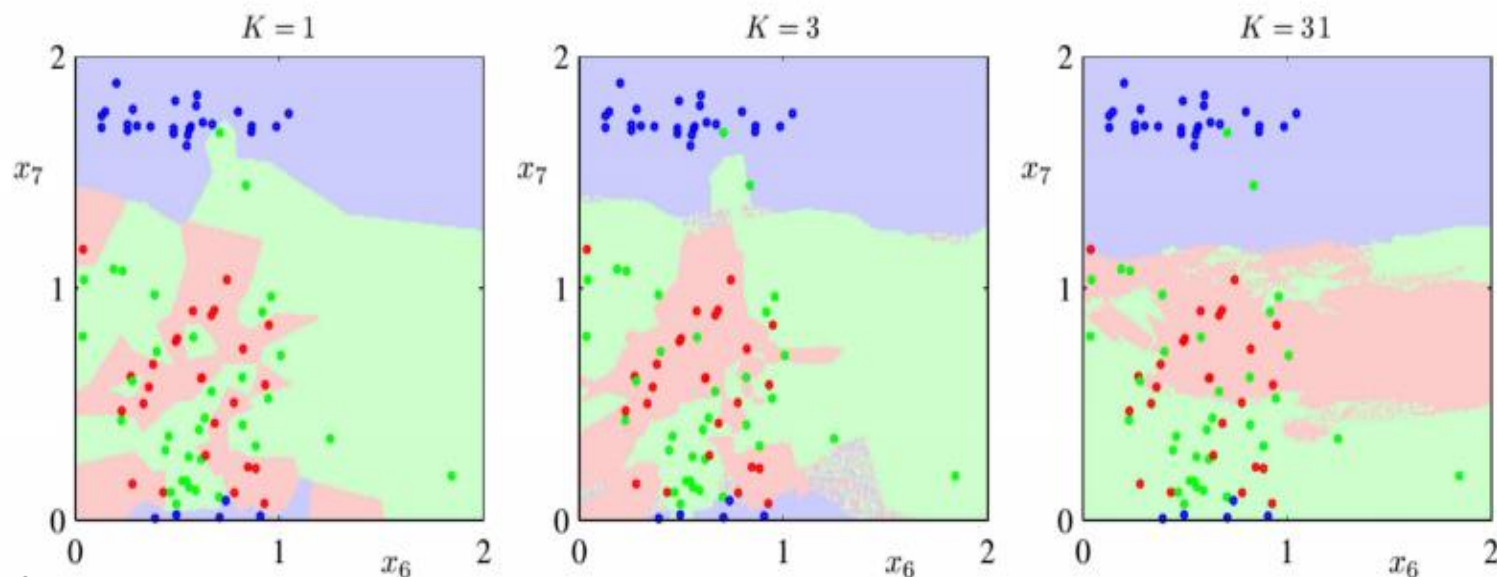


K의 값이 커질수록 Decision boundary가 안정적으로 변화



## 지도학습 (Supervised learning)

KNN(K-Nearest-Neighbor)



K의 값  $\nearrow$   $\rightarrow$  Model Complexity 감소  $\rightarrow$  High Bias & Low Variance

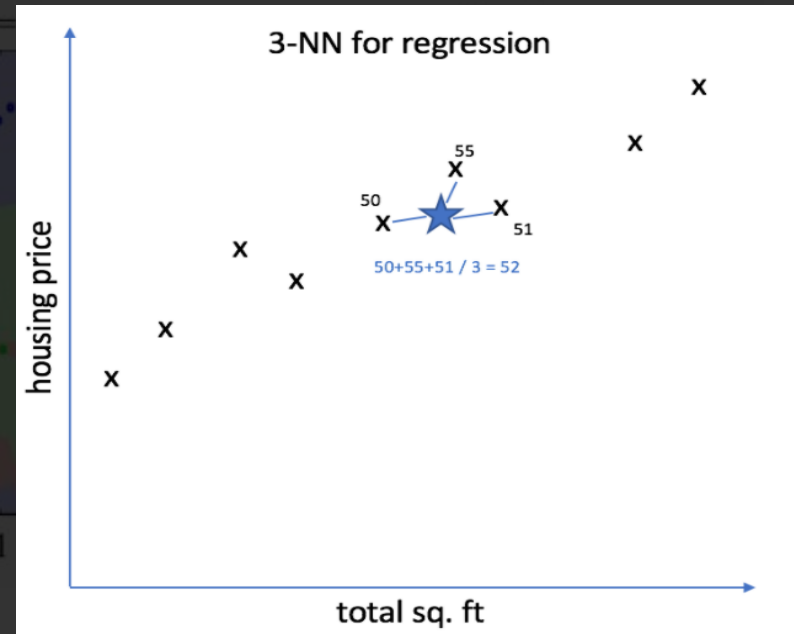
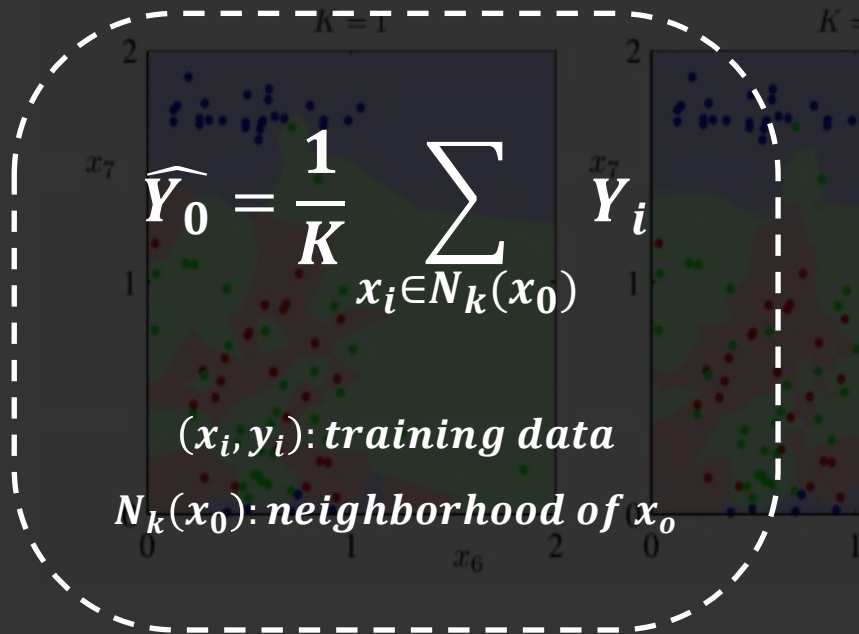
K의 값  $\searrow$   $\rightarrow$  Model Complexity 증가  $\rightarrow$  Low Bias & High Variance



## 지도학습 (Supervised learning)

### KNN Regression

KNN(K-Nearest-Neighbor)

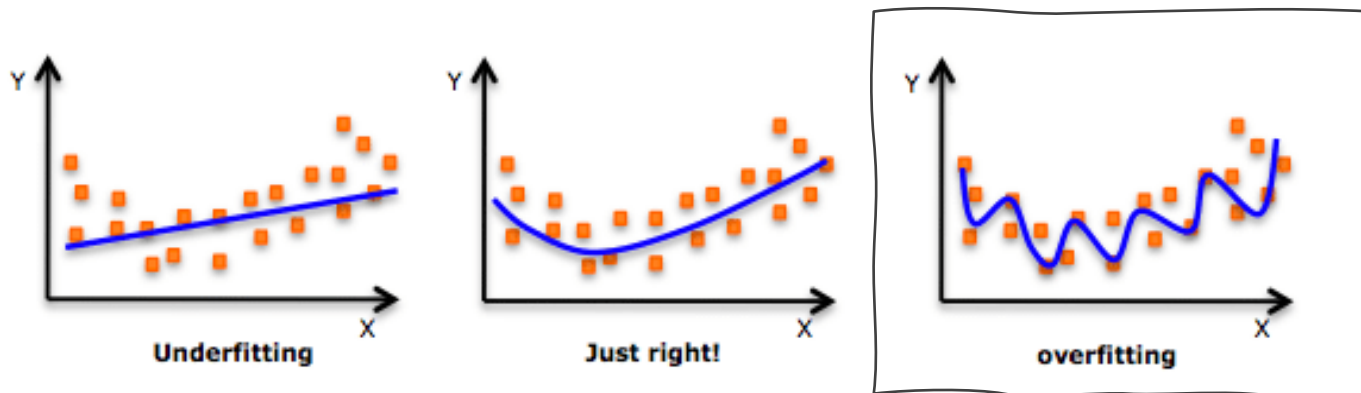


K개의 근접이웃들의 **Y값 평균**을 활용하여 예측

# 3

## 과적합 방지법

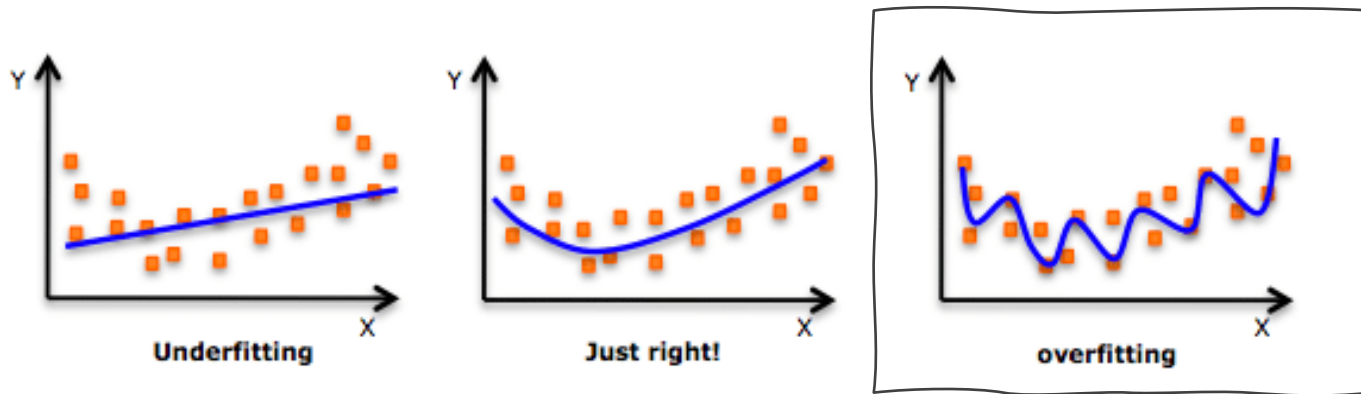
## 과적합 (Overfitting)



과적합이란?

Train data에 대해 설명력이 높아도 실제로 예측해야 하는  
Test data에 대해 설명을 못하는 현상

## 과적합 (Overfitting)



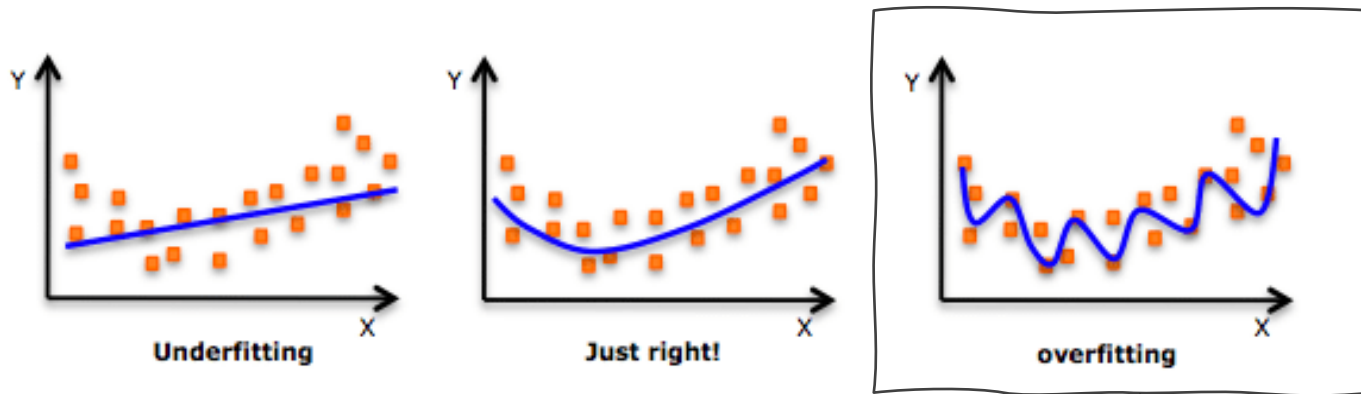
과적합이란?

Train data에 대해 설명력이 높아도 실제로 예측해야 하는  
Test data에 대해 설명을 못하는 현상



Train MSE가 작아도 Test MSE가 높을 수 있음

## 과적합 (Overfitting)



과적합이란?

Train data에 대해 설명력이 높아도 실제로 예측해야 하는  
Test data에 대해 설명을 못하는 현상

이를 방지하기 위해 여러 도구들이 등장

Cross Validation, Feature Reduction

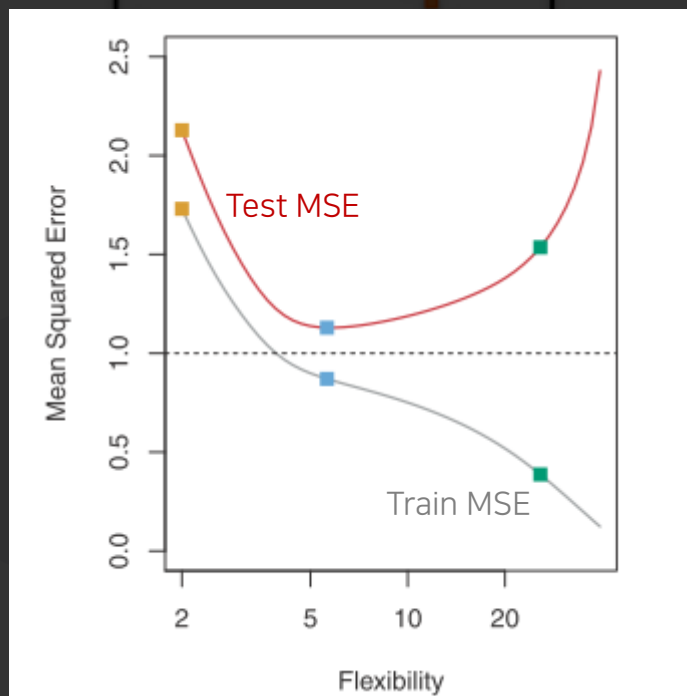


잠깐



## 과적합 (Overfitting)

“왜 Train MSE를 모델 평가 기준으로 삼을 수 없을까?”



Train MSE를 사용한다면, 가장 낮은 값을  
도출해내는 **제일 복잡한 모델**을 채택

력이 높아도 실제로 예측해야 하는  
하지만 분산이 높아지므로,  
해 설명을 못하는 현상  
**모델 예측력은 떨어짐**

모델 복잡도가 높아질수록 Train MSE는 하락함

**Test MSE**

## 교차 검증(Cross validation)

교차 검증이란?

### 교차 검증 (Cross Validation)

분석 과정에서 주어진 Train data를

다시 **Train data**와 **Validation data**로 나누어 모델의 적절성을 평가하는 방법

### Why CV?

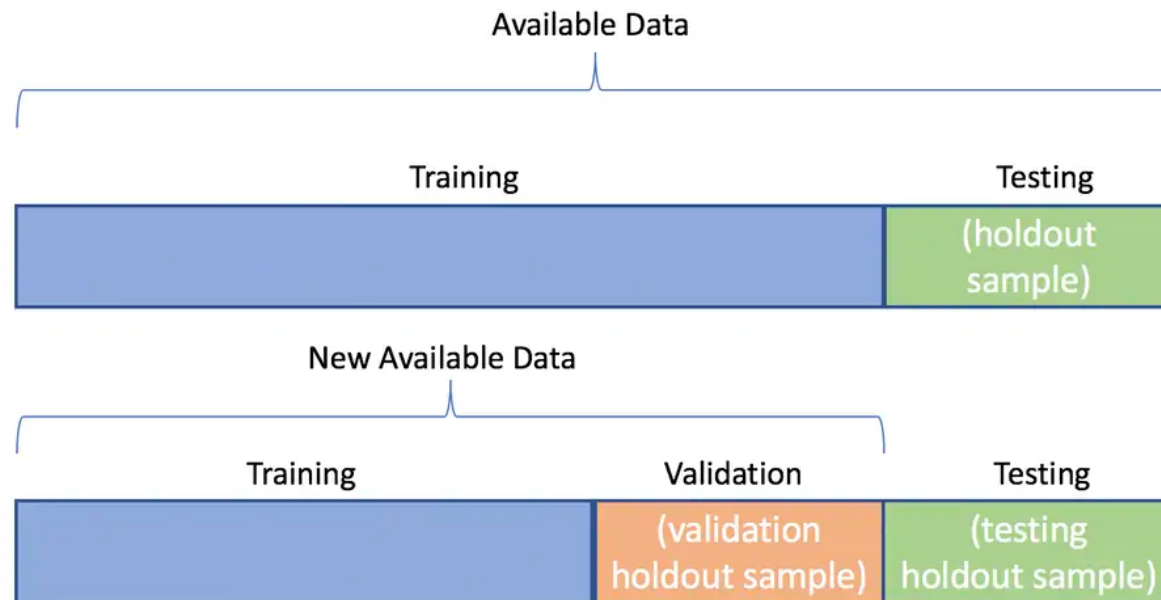
**과대적합을 방지**하고 모델의 성능 정확하게 판단 가능





## 교차 검증(Cross validation)

Hold-Out(Train-Test Split)



기존의 Train data를 둘로 쪼개는 방식으로  
일반적으로 7:3 혹은 8:2 비율로 Train-Test Split 진행

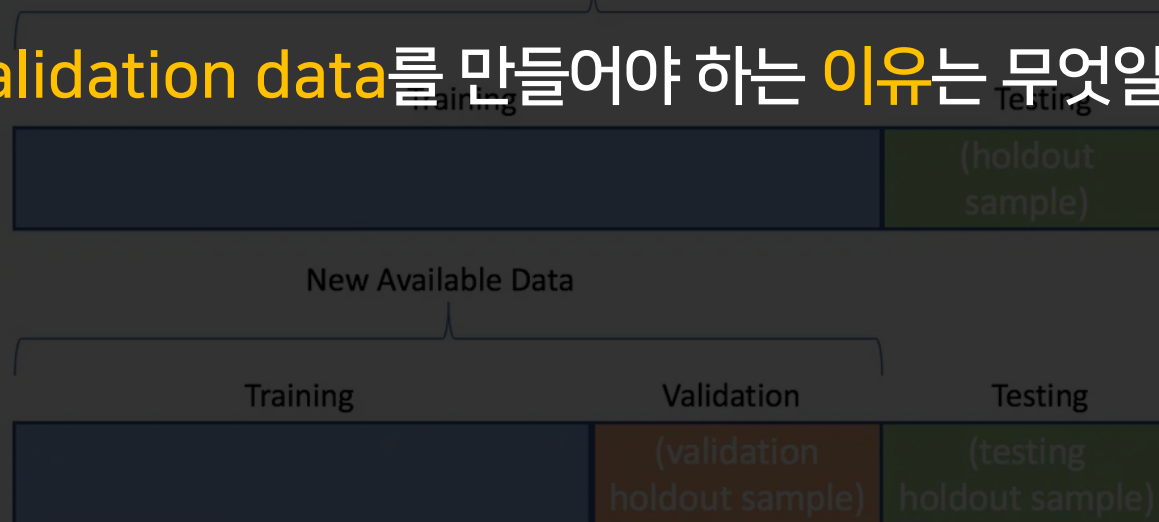
## 교차 검증(Cross validation)

Hold-Out(Train-Test Split)



그렇다면 이미 Test data가 존재하는데

Validation data를 만들어야 하는 이유는 무엇일까?



기존의 Train data를 둘로 쪼개는 방식으로  
일반적으로 7:3 혹은 8:2 비율로 Train-Test Split 진행

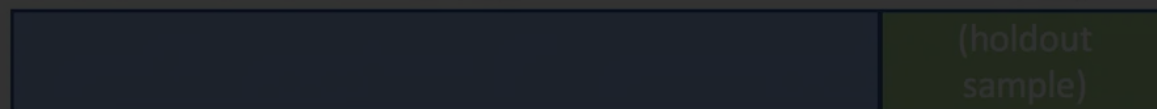
## 교차 검증(Cross validation)

Hold-Out(Train-Test Split)

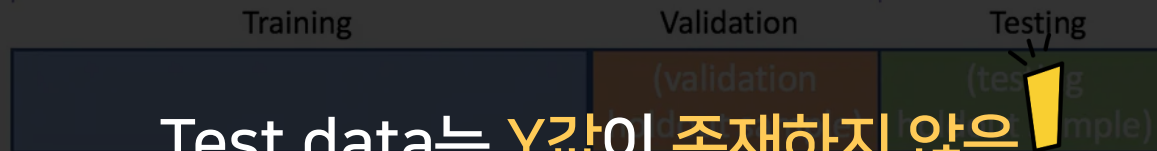


그렇다면 이미 Test data가 존재하는데

Validation data를 만들어야 하는 이유는 무엇일까?



New Available Data



Test data는 Y값이 존재하지 않음!

기존의 Train data를 둘로 쪼개는 방식으로  
일반적으로 7:3 혹은 8:2 비율로 Train-Test Split 진행

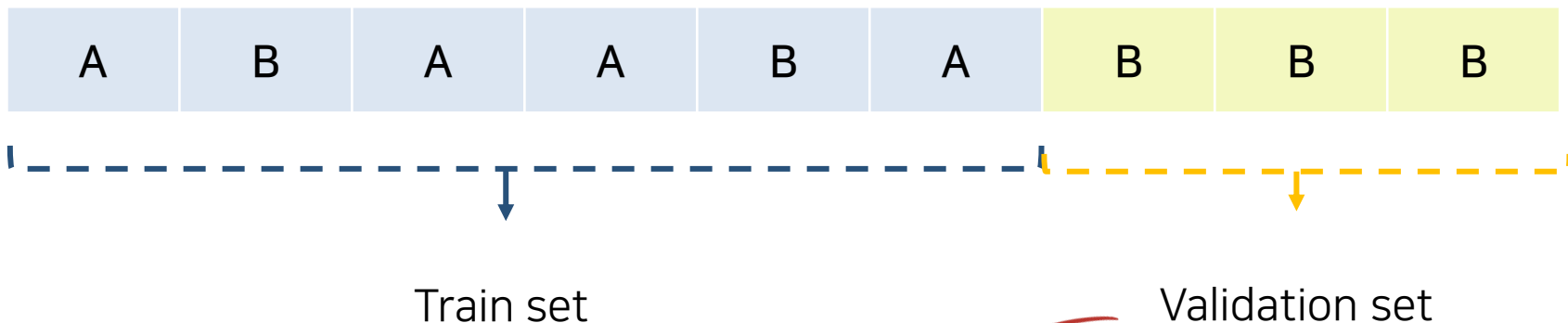
아니 없어요 그냥



## 교차 검증(Cross validation)

Hold-Out(Train-Test Split)

EX) A 와 B를 분류해주는 모델



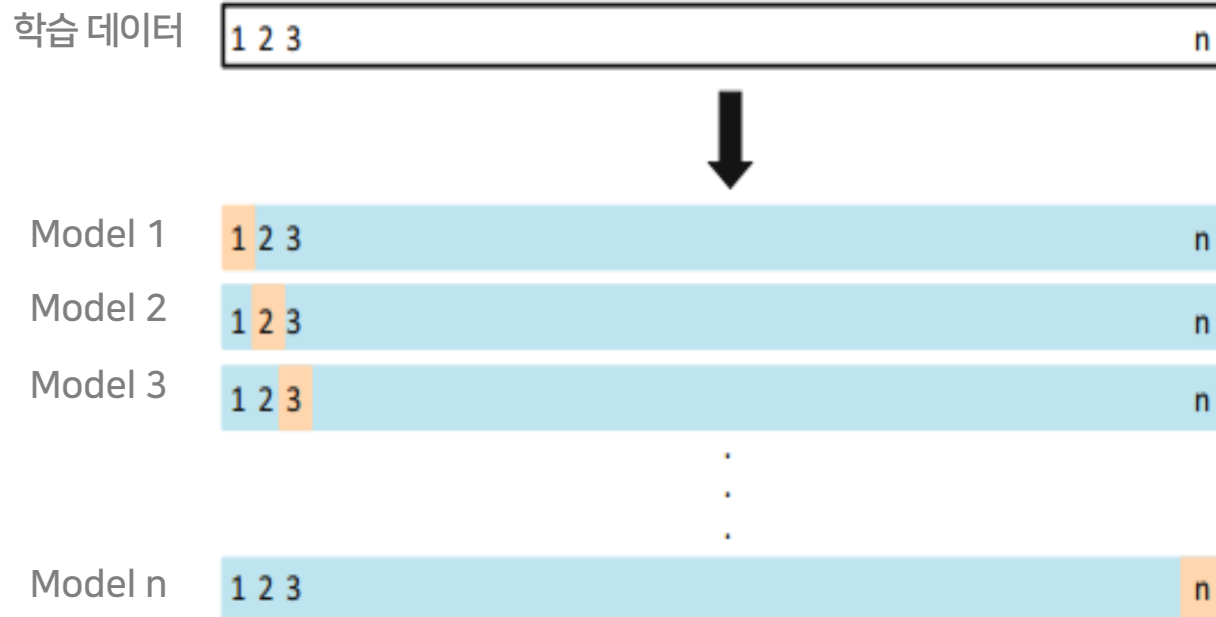
Validation set이 데이터 전체의 경향성을 보여주지 못하거나  
이상치들이 모여 있을 경우 **왜곡된 모델**을 설계한다는 한계점 존재



LOOCV, K-Fold CV 사용

## 교차검증(Cross Validation)

LOOCV(Leave-One-Out CV)



전체  $n$ 개의 학습 데이터에서 **한 개의 데이터를 검증 데이터**로,  
나머지  **$n-1$ 개의 데이터를 학습데이터**로 사용하여  $n$ 번의 검증을 진행

## 교차검증(Cross Validation)

LOOCV(Leave-One-Out CV)



학습 데이터

1 2 3

n

모델을 총  $n$ 번 학습 시켜야 하기에

Model 1

1 2 3

n

많은 컴퓨팅 파워가 요구된다는 단점

Model 2

1 2 3

n

Model 3

1 2 3

n



Model n

1 2 3

매우 작은 데이터셋에 사용하는 것을 권장



전체  $n$ 개의 학습 데이터에서 한 개의 데이터를 검증 데이터로,  
나머지  $n-1$ 개의 데이터를 학습데이터로 사용하여  $n$ 번의 검증을 진행

## 교차검증(Cross Validation)

## K-Fold CV

학습 데이터

1	2	3	...	n
---	---	---	-----	---

Fold 1

Valid Set	Train Set
-----------	-----------

Fold 2

Train Set	Valid Set	Train Set
-----------	-----------	-----------

⋮

⋮

Fold 5

Train Set	Valid Set
-----------	-----------

Test Error:

$$\frac{1}{k} \sum_{i=1}^k Error_{(i)}$$

전체 데이터를 **K개의 그룹**으로 나눈 후, **하나의 그룹을 검증 데이터셋**으로,  
나머지 **K-1개의 그룹은 학습데이터셋**으로 사용하여 K번의 검증을 진행

→ K개의 그룹이 각각 한 번씩 검증 데이터셋이 되게끔 반복 !

## 교차검증(Cross Validation)

K-Fold CV

장점

LOOCV보다 컴퓨팅 파워를 잡아먹지 않으며,

교차검증 과정에서 전체 데이터 활용 가능

→ 모델의 **과적합 여부**를 판단하는 과정에 **많이 사용**

한계점

여전히 검증 데이터셋이 전체 데이터의 경향을 **반영하지 못함**

→ K개의 그룹이 각각 한 번씩 검증 데이터셋이 되게끔 반복 !



## 교차검증(Cross Validation)

K-Fold CV

장점



데이터를 나눌 때,  
LOOCV보다 컴퓨터 자원을 잡아 먹지 않으며,  
**전체 데이터의 분포를 고려하여 분배**  
→ 모델의 **과적합 여부**를 판단하는 과정에 **많이 사용**

한계점

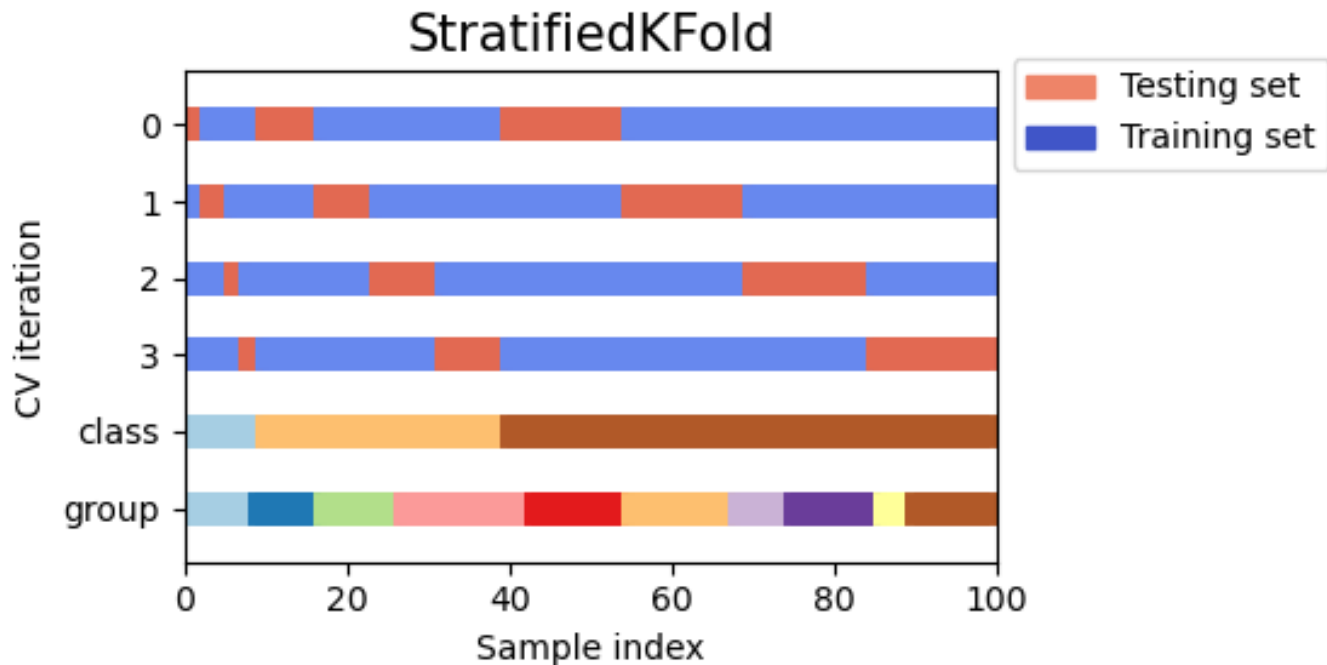
“**Stratified K-Fold CV**”

여전히 검증 데이터셋이 전체 데이터의 경향을 반영하지 못함

→ K개의 그룹이 각각 K-1개의 학습 데이터셋과 1개의 검증 데이터셋이 적게는 반복됨

## 교차검증(Cross Validation)

Stratified K-Fold CV



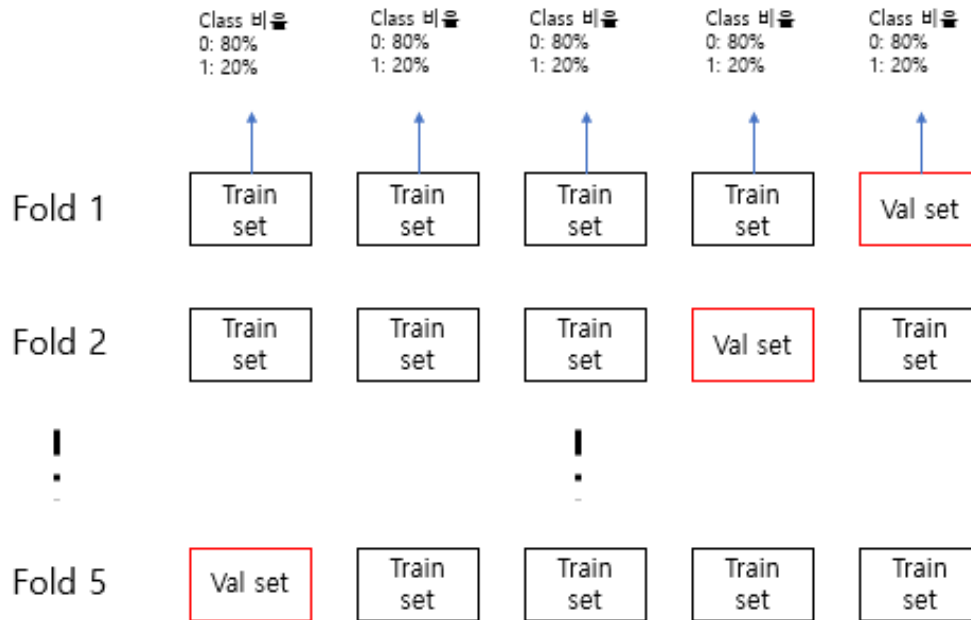
기존 K-Fold 이용 시 검증 데이터셋에 특정 클래스가 과하게 분포할 수 있음

예) 데이터셋에 빨강 1000개 파랑 10개

**불균형한 분포**를 지닌 클래스 데이터 집합을 위한 K-Fold 방식

## 교차검증(Cross Validation)

### Stratified K-Fold CV

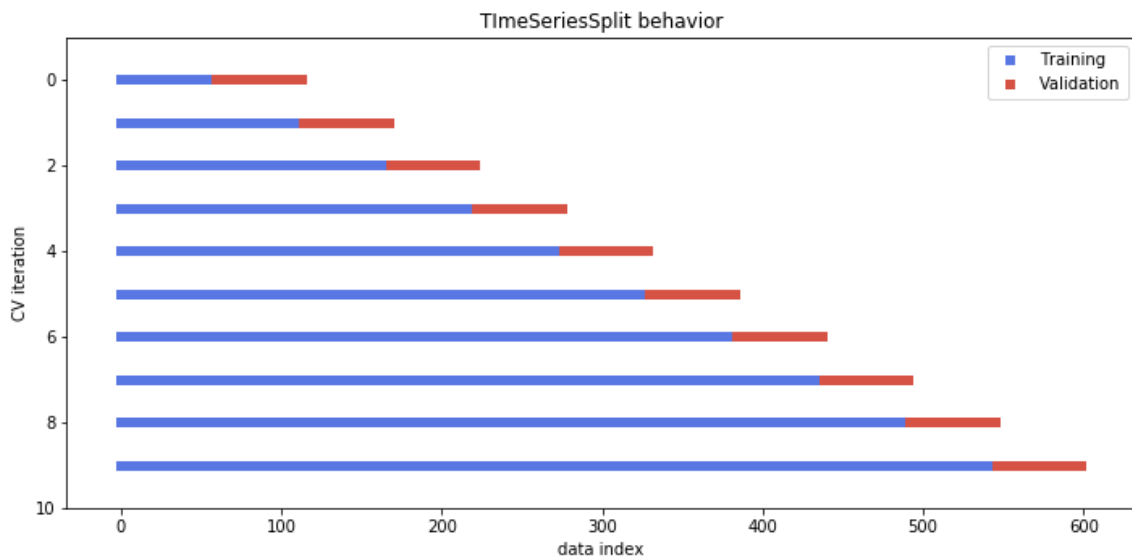


전체 데이터에 따라 학습 및 데이터셋을  
클래스 0은 80%, 클래스 1은 20%로 분배 !

전체 데이터의 분포를 고려하여 학습 데이터셋과 검증 데이터셋을 분배하므로  
 ✨ 불균형한 데이터를 사용하는 모델 성능을 측정하는데 용이함

## 교차검증(Cross Validation)

### Time Series CV



시계열 데이터는 전후 데이터 사이의 상관관계가 존재하므로

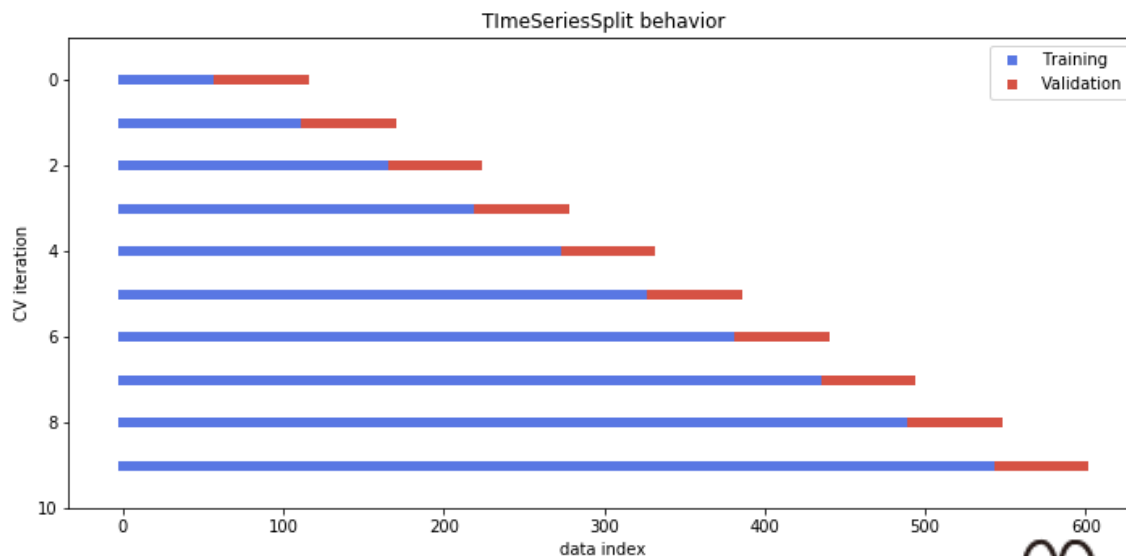
기존 교차검증 방법 **적용 불가**



학습 데이터를 항상 검증 데이터 이전으로 할당

## 교차검증(Cross Validation)

## Time Series CV



시계열 데이터는 전후 데이터 사이의 상관관계가 존재한다



시계열 클린업에서 3주차에서  
더 자세히 다룰 예정 !



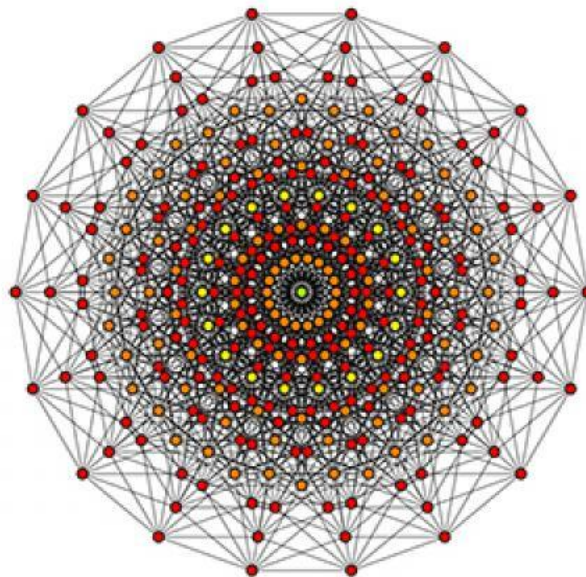
학습 데이터를 항상 검증 데이터 이전으로 한다



두 귀가 쫑긋

## 차원의 저주(Curse of Dimensionality)

과적합의 발생 원인



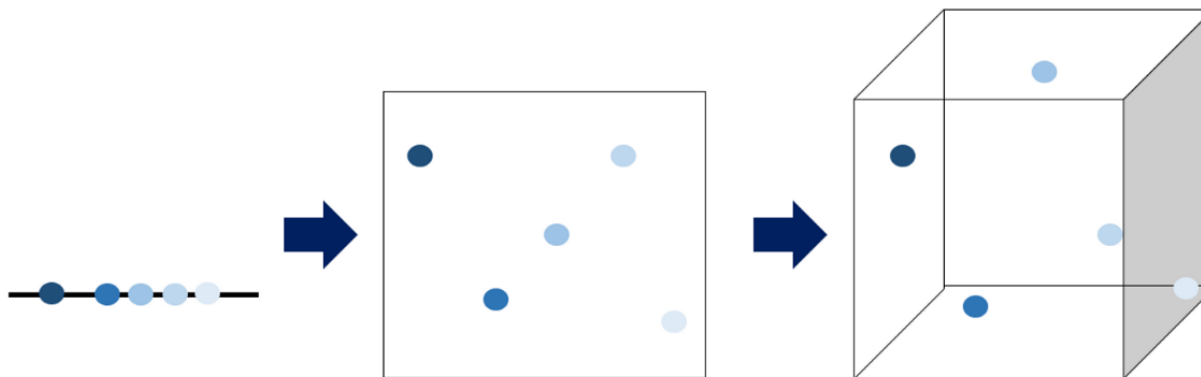
독립 변수의 개수가 많은 경우  
즉, **데이터의 차원이 높은 경우**에 과적합이 발생



## 차원의 저주(Curse of Dimensionality)란?



차원의 수가 늘어남으로써 데이터 수 증가,  
데이터의 특징이 너무 많아서 모델의 성능이 저하되는 현상

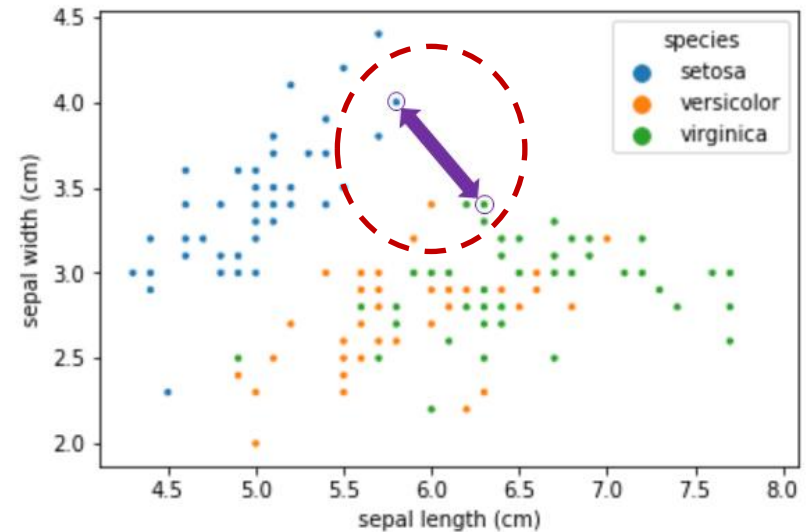
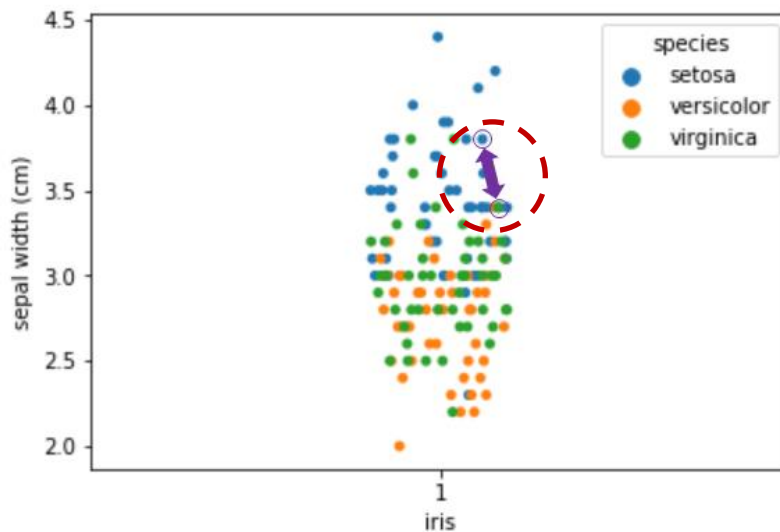


Made by: ta-daa

데이터셋이 고차원의 공간을 갖고 있다면 **데이터 간 거리가 멀어져**  
비슷한 패턴을 찾기 어려워짐

## 차원의 저주(Curse of Dimensionality)

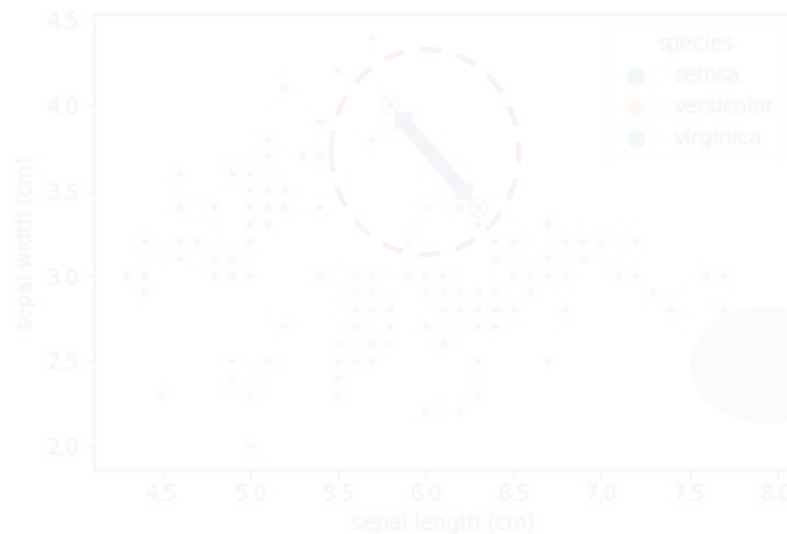
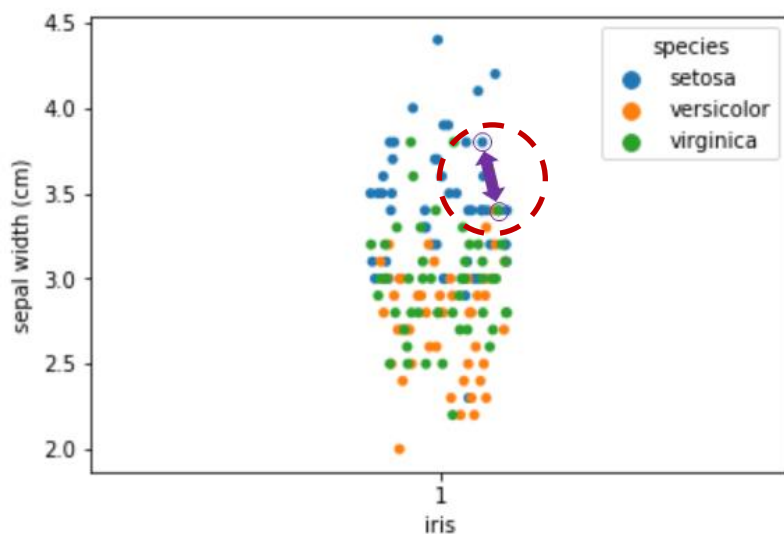
Ex) KNN을 활용한 Iris 데이터 분류 예측





## 차원의 저주(Curse of Dimensionality)

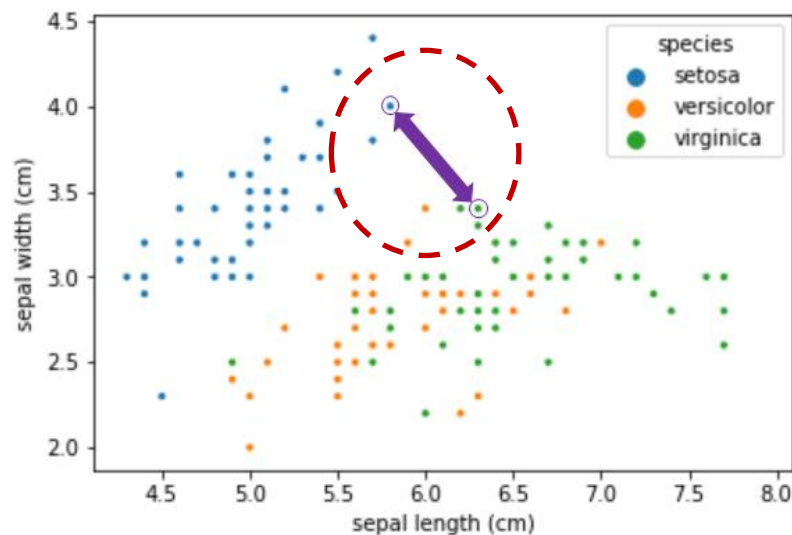
Ex) KNN을 활용한 Iris 데이터 분류 예측



Sepal Width 하나의 특성을 사용하면 **근처 데이터**들이 많아 분류하기 쉬움

## 차원의 저주(Curse of Dimensionality)

Ex) KNN을 활용한 Iris 데이터 분류 예측



그러나 Sepal Length 특성을 추가하여 분류하면 데이터 간 거리가 **멀어짐**

## 차원의 저주(Curse of Dimensionality)

Ex) KNN을 활용한 Iris 데이터 분류 예측

데이터가 너무 고차원이라 데이터 간 간격이 멀어질 경우

빈 공간에 대해 컴퓨터는 '관측값이 없다'고 인식하여  
데이터셋이 전체 공간을 나타내지 못함

특정 부분만 학습되면서 그 부분에 대해 과적합 발생

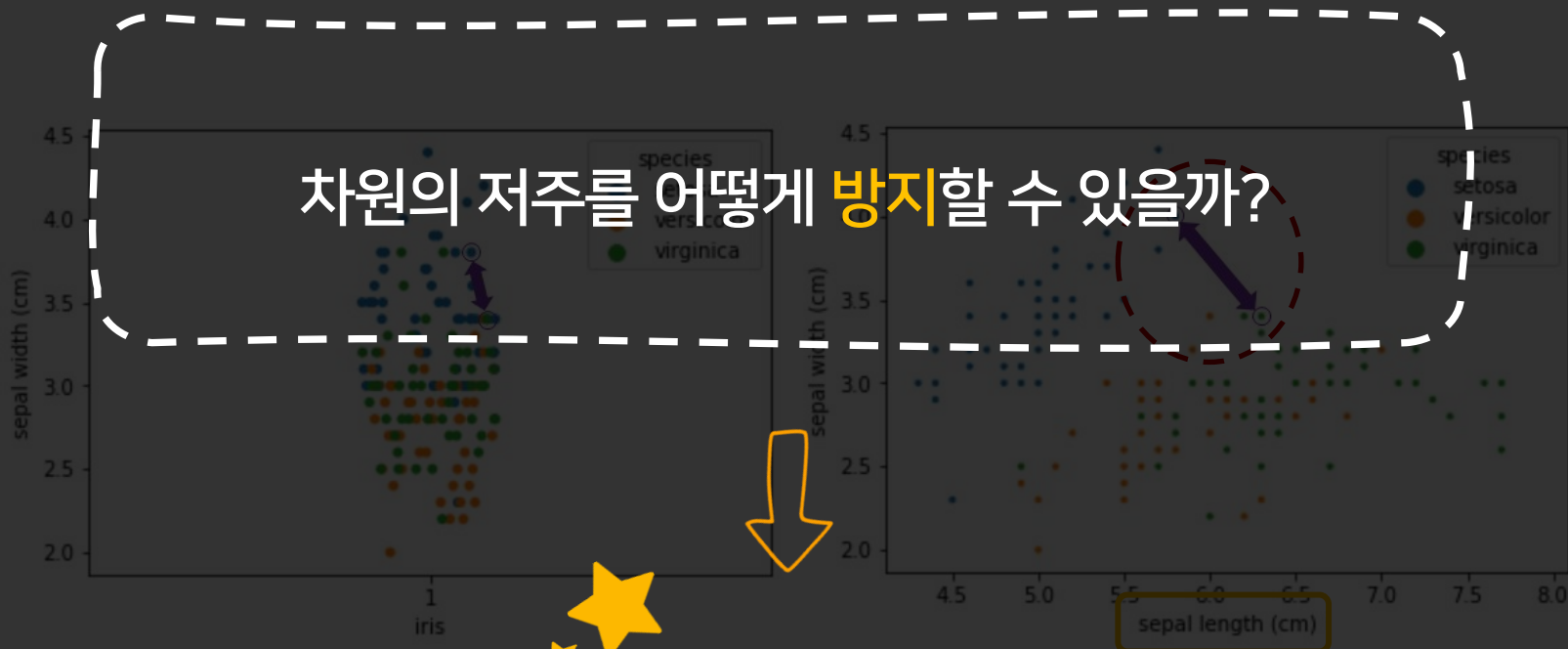
**"설명력 감소"**



그러나 Sepal Length 특성을 추가하여 분류하면 데이터 간 거리가 멀어짐

## 차원의 저주(Curse of Dimensionality)

예. KNN을 활용한 Iris 데이터 분류 예측



### 차원 축소

그러나 Sepal Length 특성을 추가하여 분류하면

데이터 간 거리가 멀어짐

## 차원 축소

### 변수선택법

Feature Selection

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

Forward Selection

Backward Elimination

Step-wise Selection

V/S

### 변수추출법

Feature Extraction

데이터의 차원을  
고차원에서 **저차원으로**  
**변환**함으로써  
모델을 적합시킴

PCA(Principal Component Analysis)

## 차원 축소

### 변수선택법

Feature Selection

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

Forward Selection

Backward Elimination

Step-wise Selection

V/S

### 변수추출법

Feature Extraction

데이터의 차원을  
고차원에서 **저차원으로**  
**변환**함으로써  
모델을 적합시킴

PCA(Principal Component Analysis)

## 차원 축소

### 변수선택법

#### Feature Selection

데이터의 특성을  
가장 잘 설명하는 변수를  
추가하거나 제거해가며  
모델을 적합시킴

Forward Selection

Backward Elimination

Step-wise Selection

V/S

### 변수추출법

#### Feature Extraction

데이터의 차원을  
고차원에서 **저차원으로**  
**변환**함으로써  
모델을 적합시킴

PCA(Principal Component Analysis)

## 차원 축소

변수선택법

Feature Selection

변수추출법

Feature Extraction



V/S

데이터의 차원을  
고차원에서 **저차원으로**  
**변환**함으로써  
모델을 적합시킴

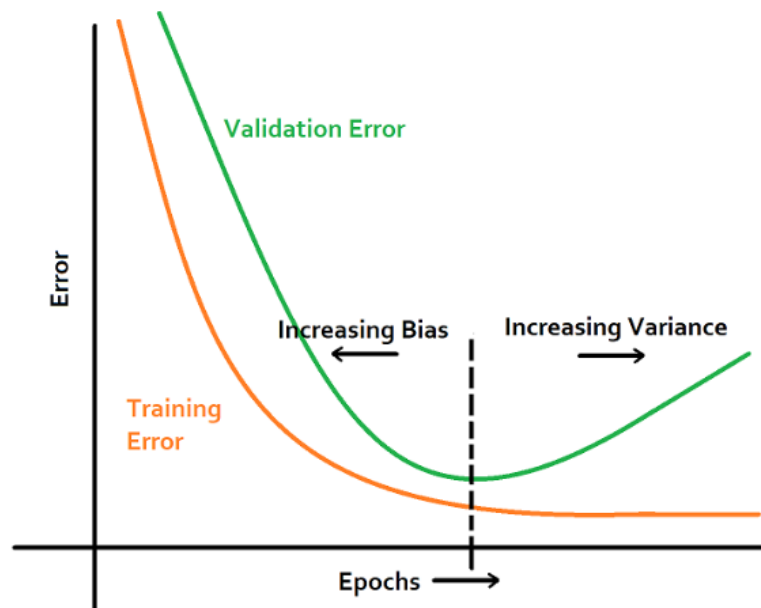
회귀분석팀과 선형대수학팀 클린업에서  
자세히 다룰 예정!

PCA(Principal Component Analysis)

Step-wise Selection



## Early Stopping



학습 관점에서의 과적합 방지법으로  
학습 소요 시간에 제한을 두거나, 모델 성능이 일정 수준 이상이 되면 학습을 종료



과적합 방지 가능

감사 감사



THANK YOU