



17기 분석 base세션 2주차

회귀/오버, 언더피팅/정규화

발표자: 16기 분석 김영은

INDEX

001 머신 러닝

002 (Simple) Linear Regression

003 Bias & Variance

004 Multiple Linear Regression

005 Logistic Regression

006 과제 및 참고

1. 머신 러닝

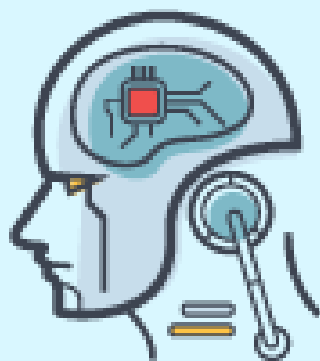
1. 머신 러닝

데이터를 반복적으로 학습해 데이터에 숨어있는 패턴을 찾아내는 것 !

Artificial Intelligence

인공지능

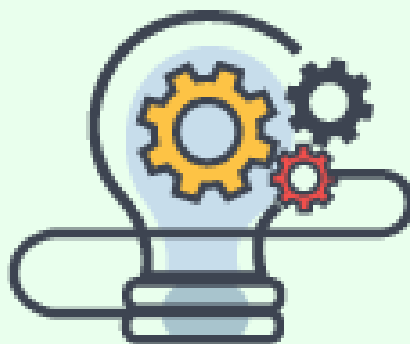
사고나 학습 등 인간이 가진
지적 능력을 컴퓨터를 통해
구현하는 기술



Machine Learning

머신러닝

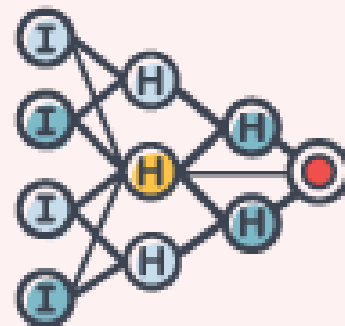
컴퓨터가 스스로 학습하여
인공지능의 성능을
향상 시키는 기술 방법



Deep Learning

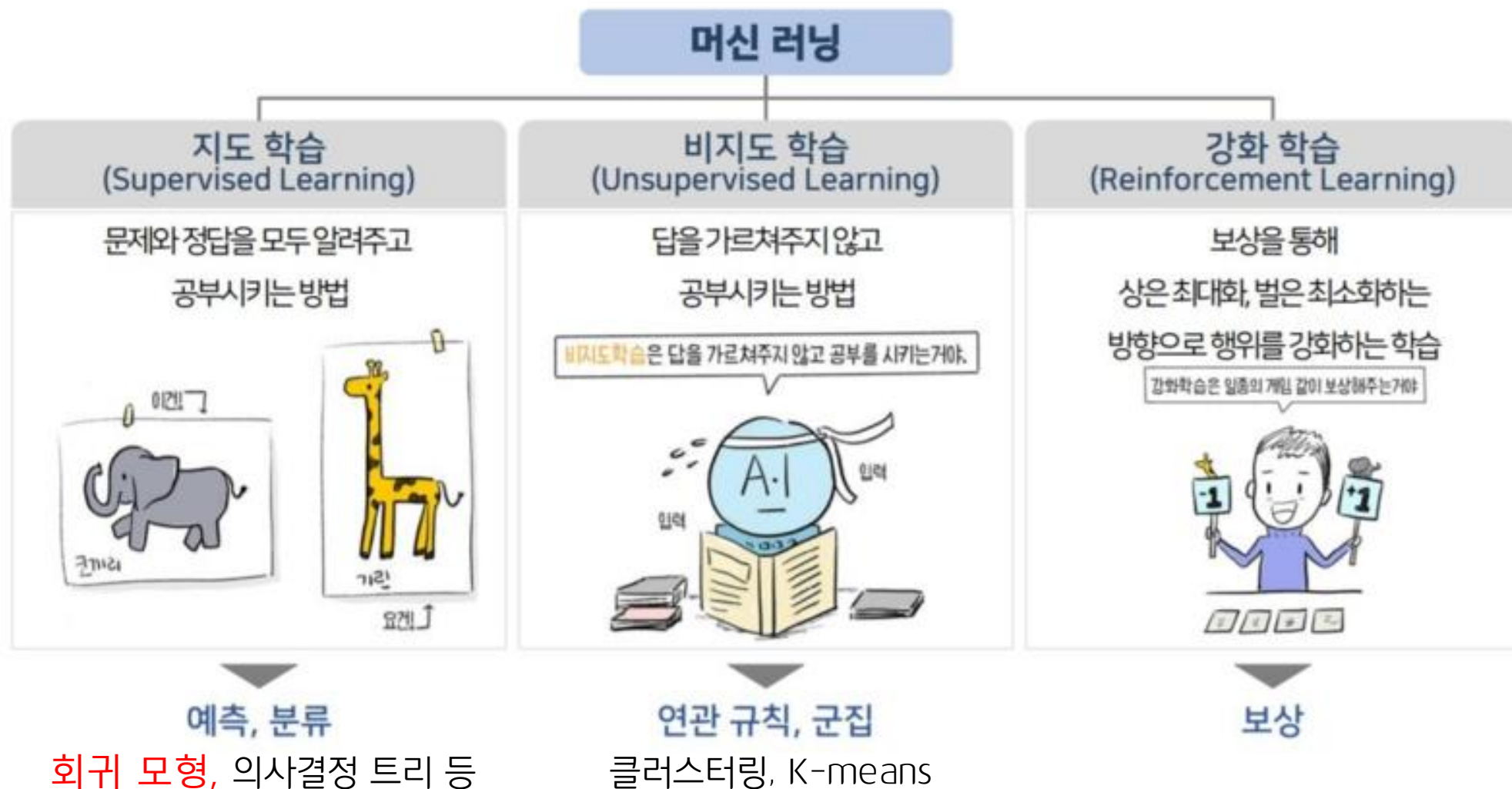
딥러닝

인간의 뉴런과 비슷한
인공신경망 방식으로
정보를 처리

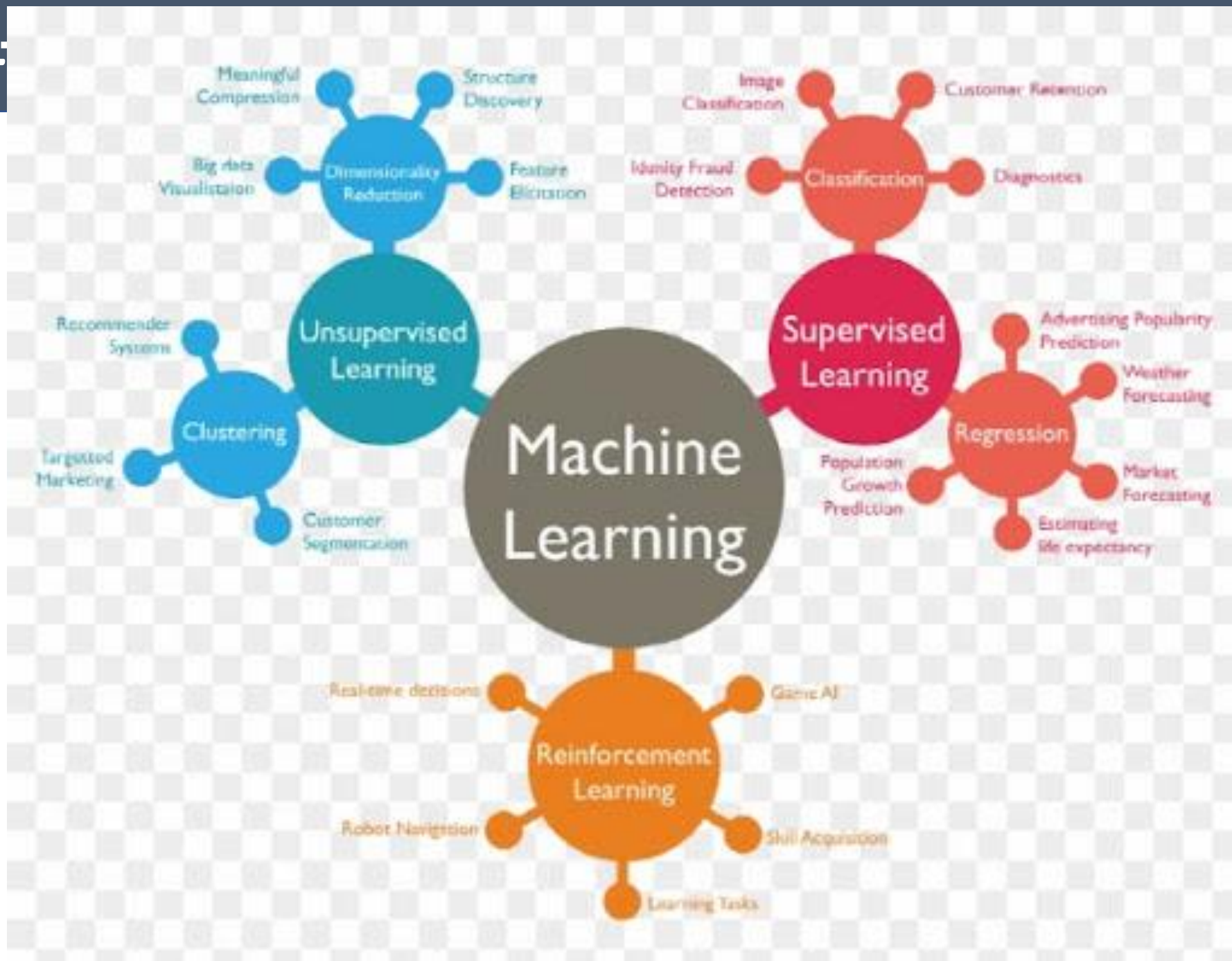


1. 머신 러닝

데이터를 반복적으로 학습해 데이터에 숨어있는 패턴을 찾아내는 것 !



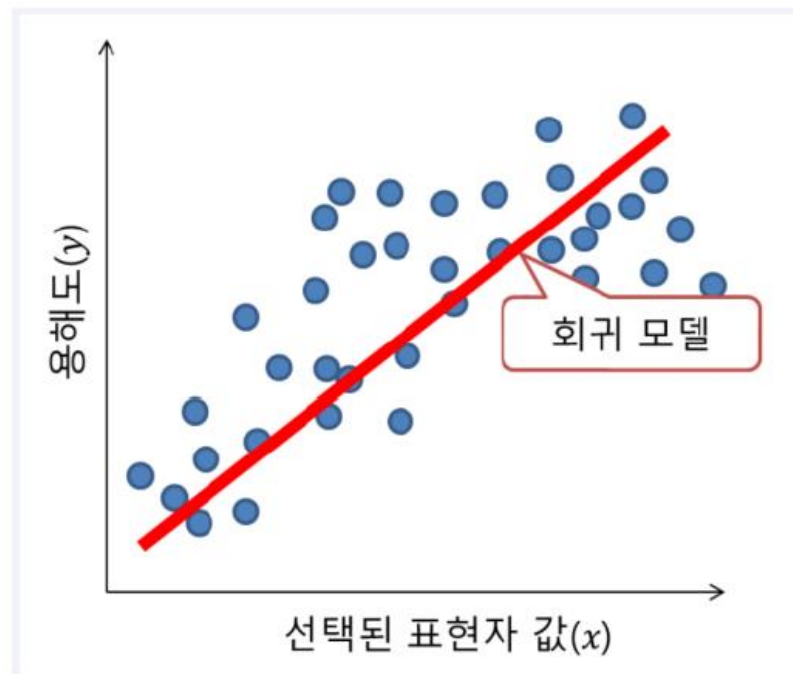
1. 머신 러



1. 머신 러닝

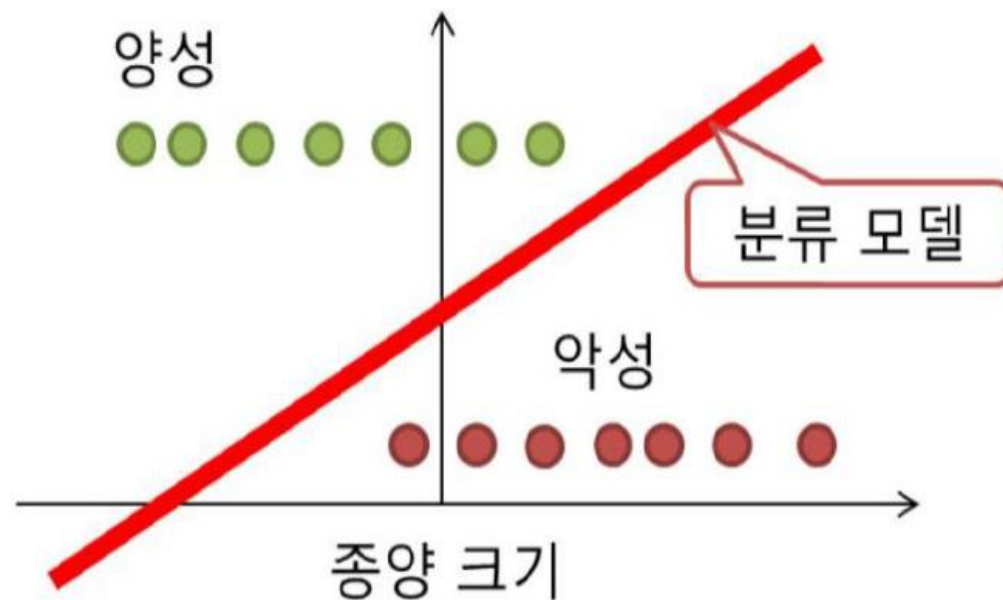
종속 변수의 형태가 무엇이나에 따라

1. Regression 회귀



예) 집값 예측, GDP 예측 등

2. Classification 분류



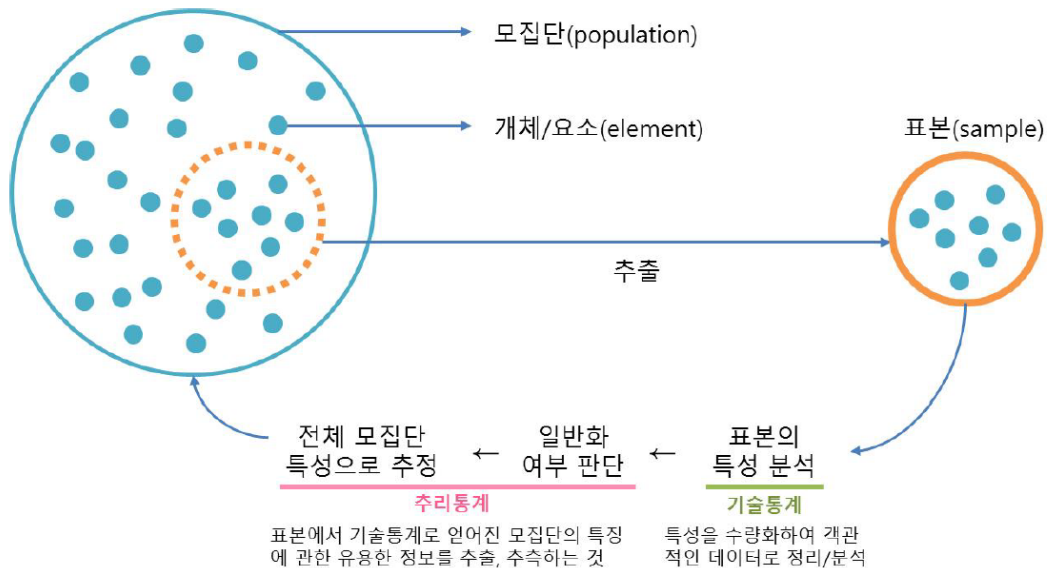
예) 스팸 분류기, 악성종양 판별 등

2. (Simple) Linear Regression

2. Linear Regression

Simple Linear Regression 단순선형회귀 : 입력 변수 X가 1개일 때

$$Y = \beta_0 + \beta_1 X + \varepsilon \quad \longrightarrow \quad \hat{y} = b_0 + b_1 x$$



<가정>

회귀 모형은 모수에 대해 선형인 모형이다
오차항의 평균은 0이고, 분산은 σ^2 이다.

$(E\varepsilon_i=0, Var(\varepsilon_i)=\sigma^2)$

오차항은 독립이다.

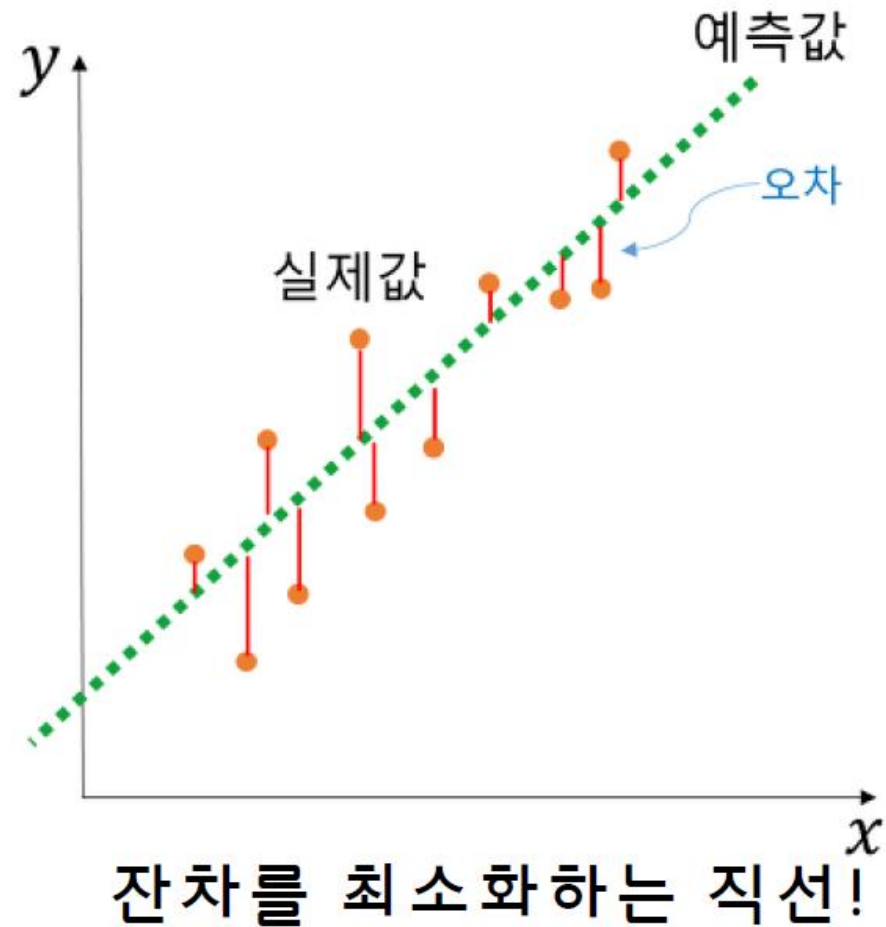
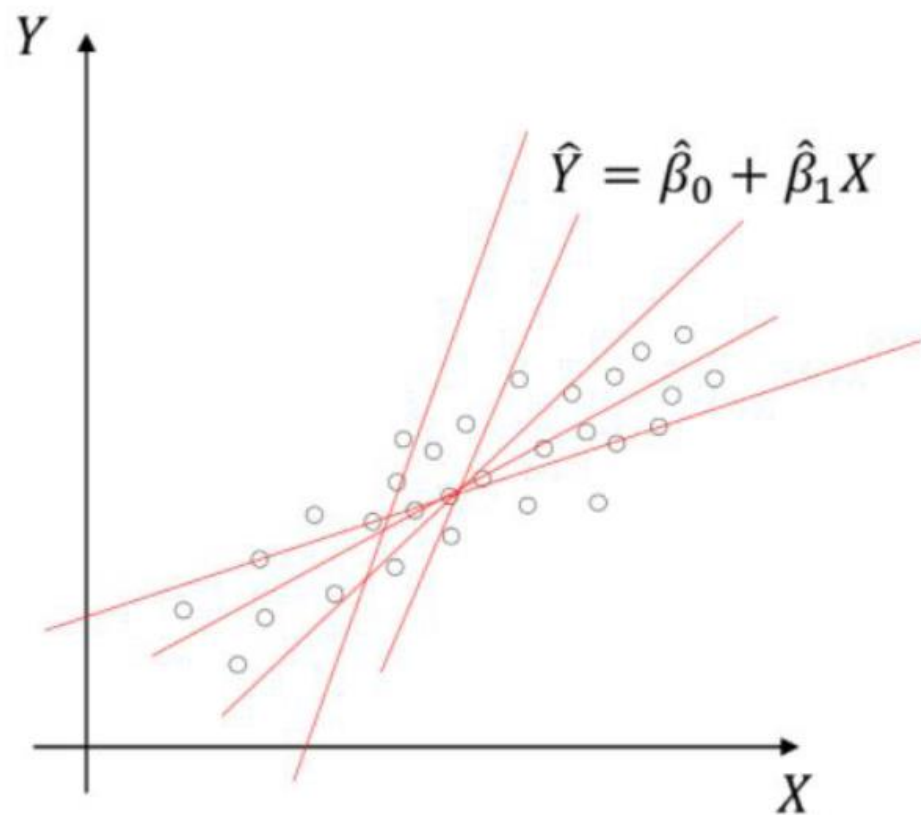
$Cov(\varepsilon_i, \varepsilon_j)=0, i \neq j$

오차항은 정규분포를 따른다.

독립변수 X는 비 확률(nonstochastic) 변수이다.

2. Linear Regression

수많은 직선 중 무엇을 골라야 할까?



2. Linear Regression

Ordinary Least Square Error(OLSE) 최소 제곱법

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \quad i=1, \dots, n, \quad (x_i, y_i) : i\text{th 관측값.}$$

$(y - \hat{y})$ 차이 작게 해주는 line

$$\sum_{i=1}^n \varepsilon_i^2 = \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i)^2$$

$$\text{" } S(\beta), \quad \beta = (\beta_0, \beta_1)$$

$$\hat{\beta} = \arg \min_{\beta} S(\beta), \quad \text{where } \hat{\beta} = (\hat{\beta}_0, \hat{\beta}_1)$$

2. Linear Regression

Ordinary Least Square Error(OLSE) 최소 제곱법

$$\hat{\beta}_1 = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{n \sum x_i^2 - (\sum x_i)^2} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2} = \frac{S_{xy}}{S_{xx}} \quad (\hat{\beta}_1 \text{의 분산} = \text{corr 분산})$$
$$\hat{\beta}_0 = \frac{1}{n} \sum y_i - \frac{1}{n} \sum x_i \hat{\beta}_1 = \bar{y} - \hat{\beta}_1 \bar{x}.$$

$$\sum e_i = \sum (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) = 0$$

$$\sum x_i e_i = \sum x_i (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) = 0$$

$$\hat{\beta}_1 \sim N\left(\beta_1, \frac{\sigma^2}{S_{xx}}\right)$$

$$\hat{\beta}_0 \sim N\left(\beta_0, \sigma^2 \left(\frac{1}{n} + \frac{\bar{x}^2}{S_{xx}}\right)\right)$$

- $S(\beta_0, \beta_1)$ 을 최소로 하는 회귀 계수 확인
- 잔차들의 합과 잔차*x들의 합이 0이라는 식을 유도할 수 있다.
- 추정된 β_0, β_1 의 평균과 표준오차를 구하여 분포를 알면, 검정통계량을 구할 수 있다.

2. Linear Regression

선형 회귀 식 정확도 평가 방법

1. $MSE(\text{Mean Squared Error}) = \frac{SSE}{n-2}$

2. $RMSE(\text{Mean Squared Error}) = \sqrt{MSE}$

3. $R\text{-Squared} = \frac{SSR}{SST}$

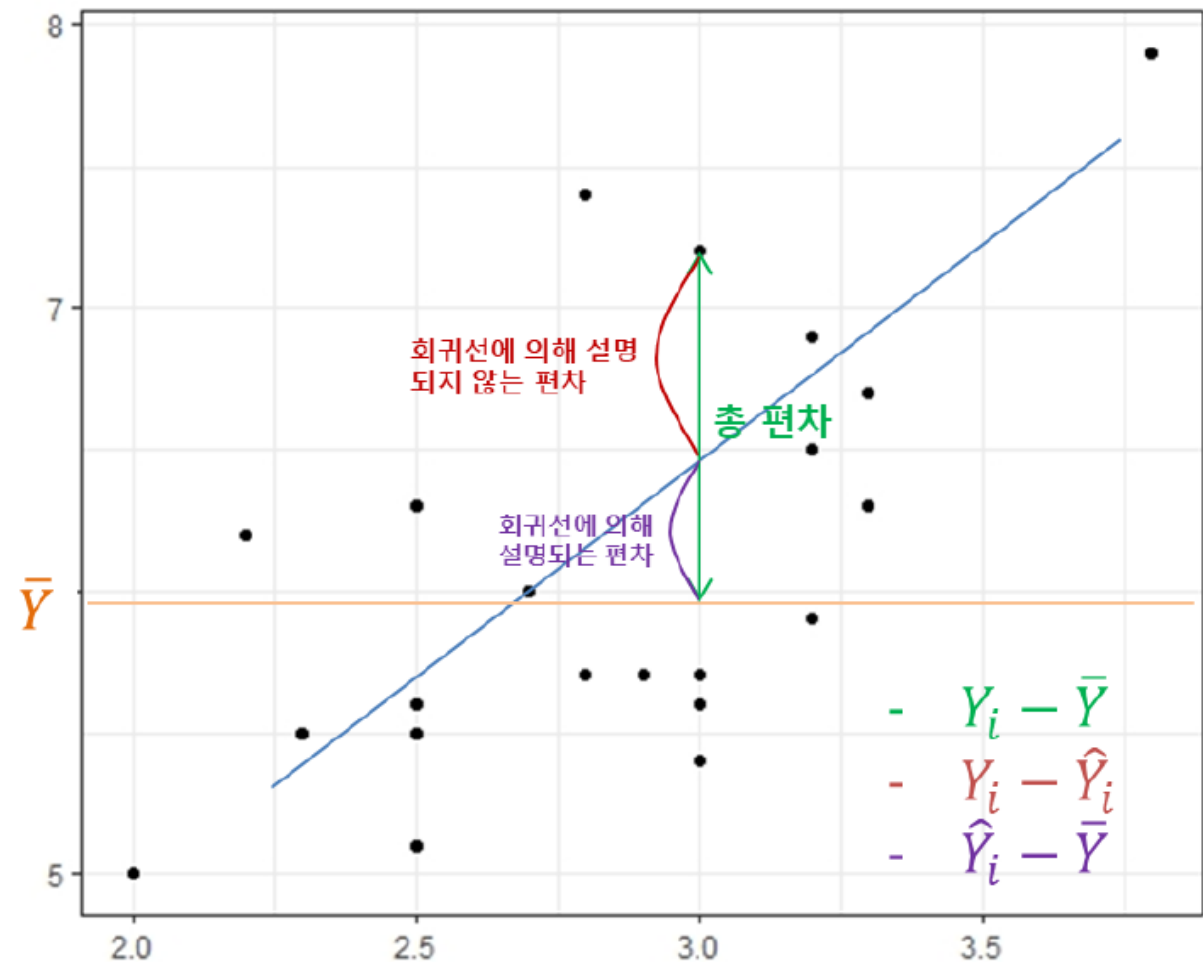
$$y_i - \bar{y} = y_i - \hat{y}_i + \hat{y}_i - \bar{y}$$

$$\sum_{i=1}^n (y_i - \bar{y})^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \sum_{i=1}^n (\hat{y}_i - \bar{y})^2$$

SST
(Total sum of squares) SSE
(Error sum of squares) SSR
(Regression sum of squares)

$$R^2 = \frac{SST - SSR}{SST} = 1 - \frac{SSE}{SST} = \frac{SSR}{SST} \quad \text{where } SST = \sum_{i=1}^n (y_i - \bar{y})^2$$

입력 변수로 설명할 수 없는 변동 비율



3. Bias & Variance

3. Bias & Variance

1. Bias

실제 값에서 멀어진 척도
예측 값과 실제 값 간의 발생하는 차이

$$Bias = E[f^{pred}(x)] - f(x)$$

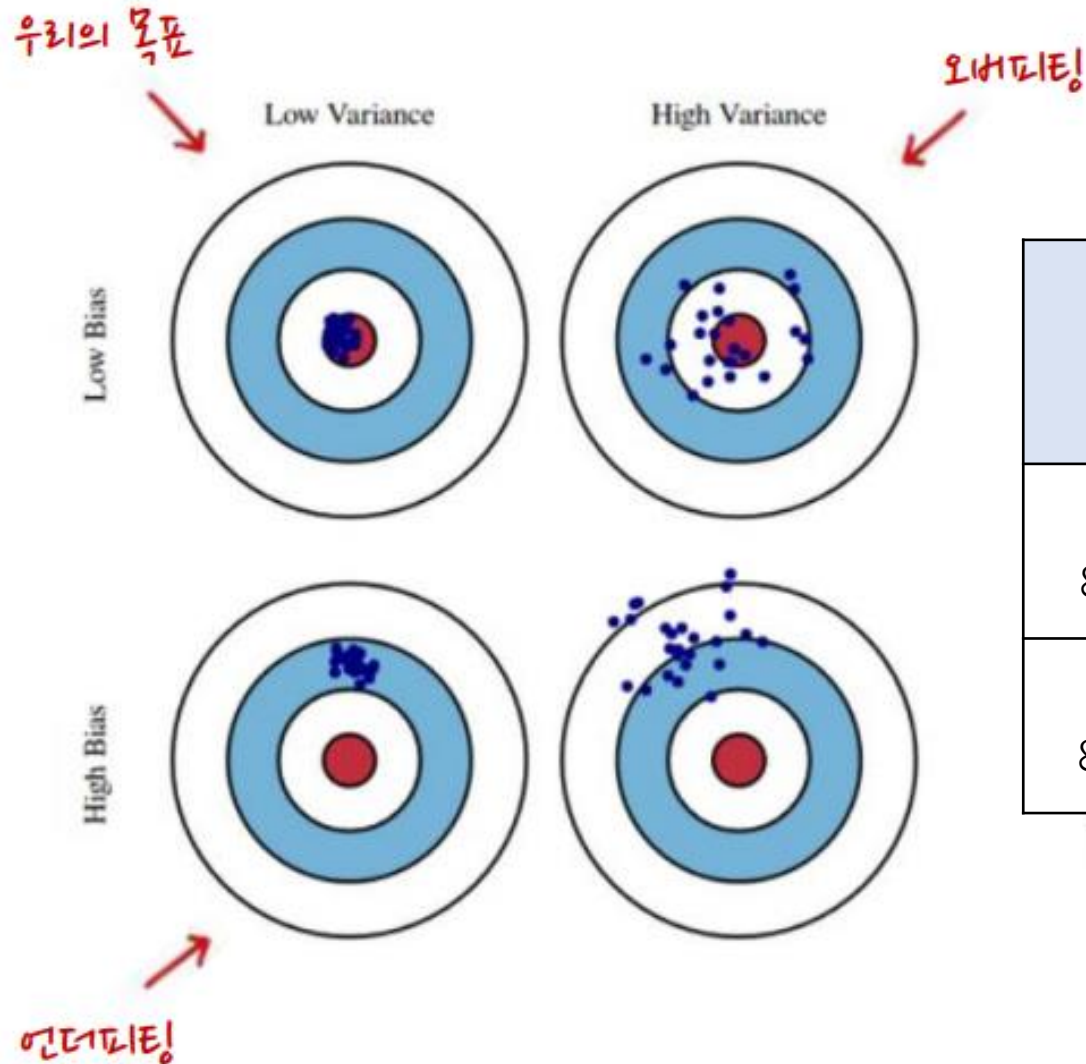
2. Variance

예측값끼리 서로 얼마나 떨어져 있는가
추정 값들의 흩어진 정도

$$Variance = E[f^{pred}(x) - E[f^{pred}(x)]]^2$$

3. Bias & Variance

3. 오버피팅과 언더피팅

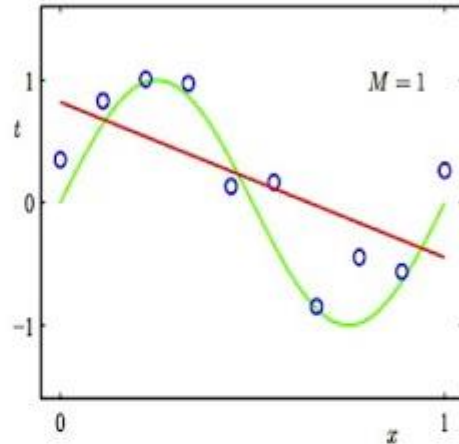


구분	Flexibility	Fitting
Low Bias & High Variance	Flexible	Overfitting 오버피팅
High Bias & Low Variance	Inflexible	Underfitting 언더피팅

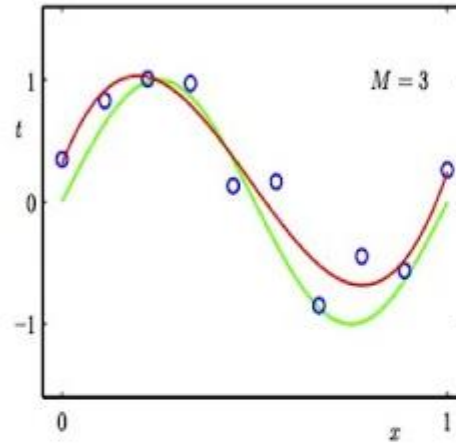
Under- and Over-fitting examples

High bias & Low variance

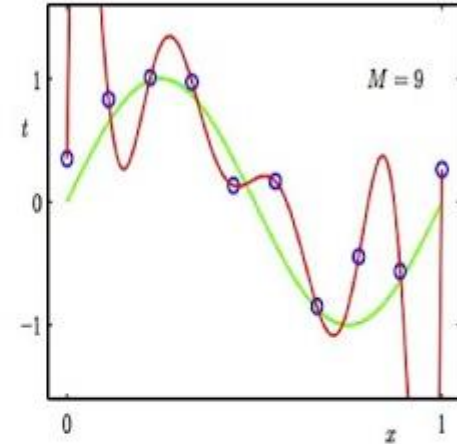
Regression:



predictor too inflexible:
cannot capture pattern

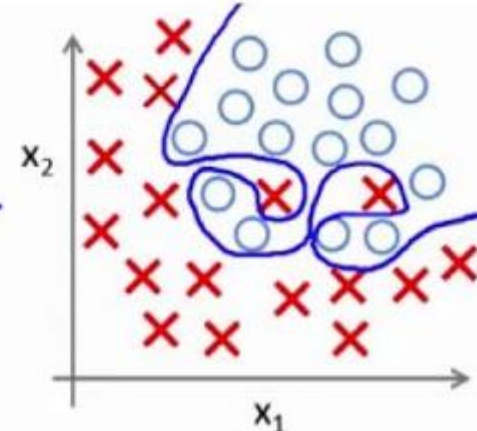
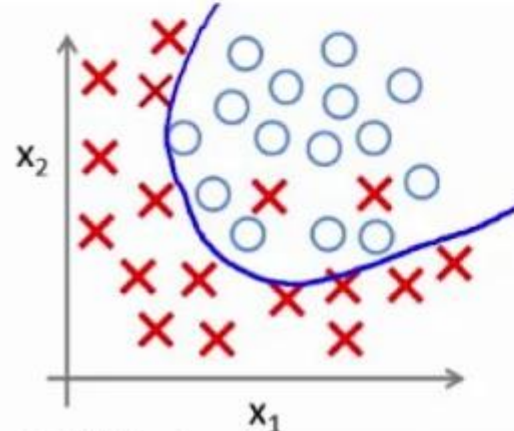
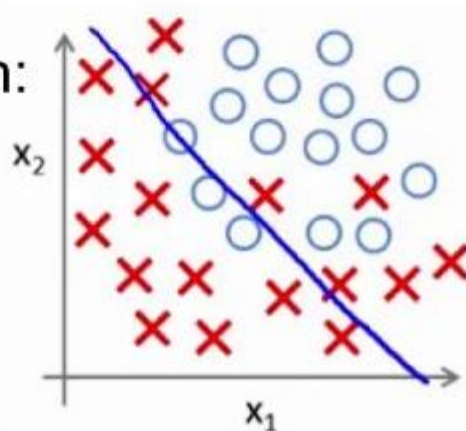


Low bias & High variance



predictor too flexible:
fits noise in the data

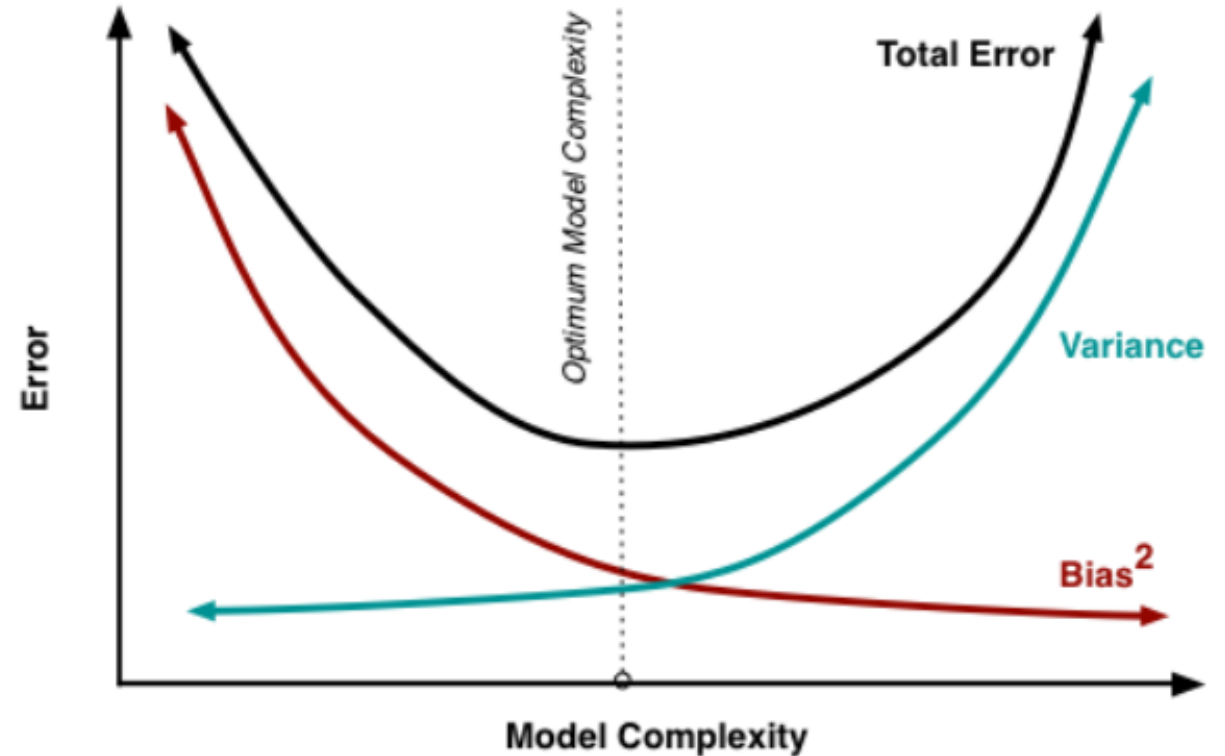
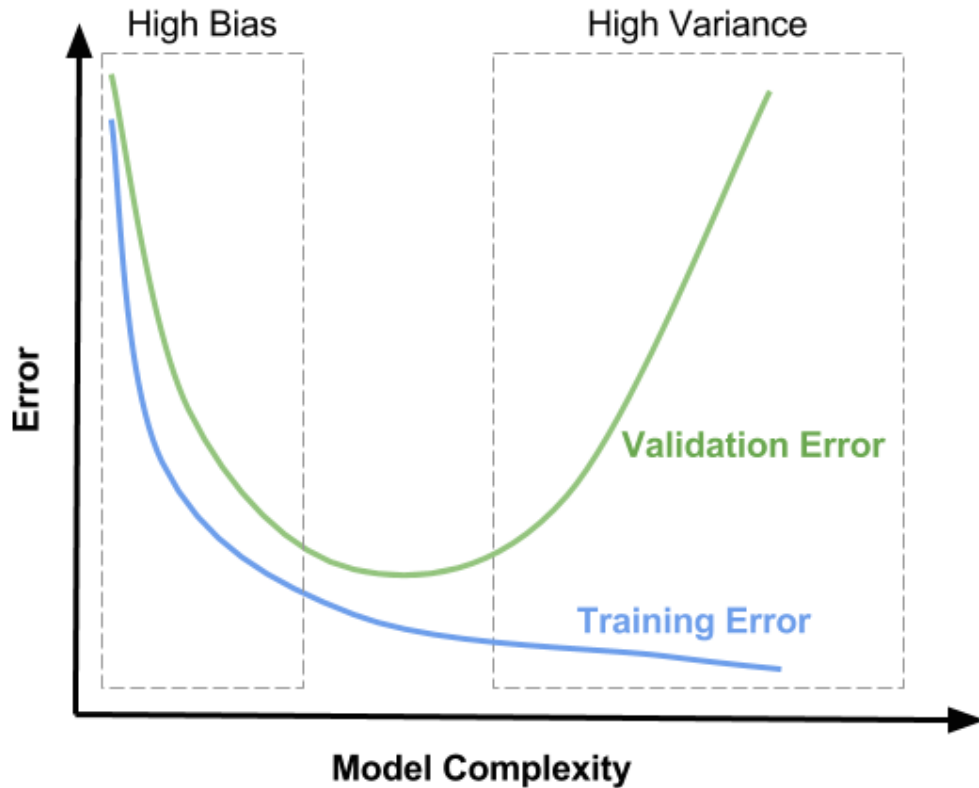
Classification:



3. Bias & Variance

4. 우리의 목표 !

$\text{COST(MSE)} = \text{bias}^2 + \text{Variance}$ 이고 Bias와 Variance는 trade off
COST가 최저가 되기 위해서는 bias와 Variance가 균형을 이루어야 한다



3. Bias & Variance

4. 우리의 목표 !

Feature Scaling = Data normalization

	표준화(standardization)	정규화(normalization)
공통점	데이터 rescaling	
정의 & 목적	데이터가 <u>평균으로부터 얼마나 떨어져있는지</u> 나타내는 값으로, 특정 범위를 벗어난 데이터는 outlier로 간주, 제거	데이터의 <u>상대적 크기</u> 에 대한 영향을 줄이기 위해 데이터범위를 0~1로 변환
값의 범위	± 1.96 (또는 ± 2) 데이터만 선택	0~1
공식	$Z = \frac{X - \bar{X}}{\sigma}$ <p>(분모가 표준편차)</p>	$X_{new} = \frac{X - X_{min}}{X_{max} - X_{min}}$ <p>(분모가 max값)</p>

3. Bias & Variance

4. 우리의 목표 ! 언더피팅, 오버피팅 해결하기

언더피팅 해결하기

1. Feature 수 늘리기. 더 많이 반영 필요
2. Variance 높이기

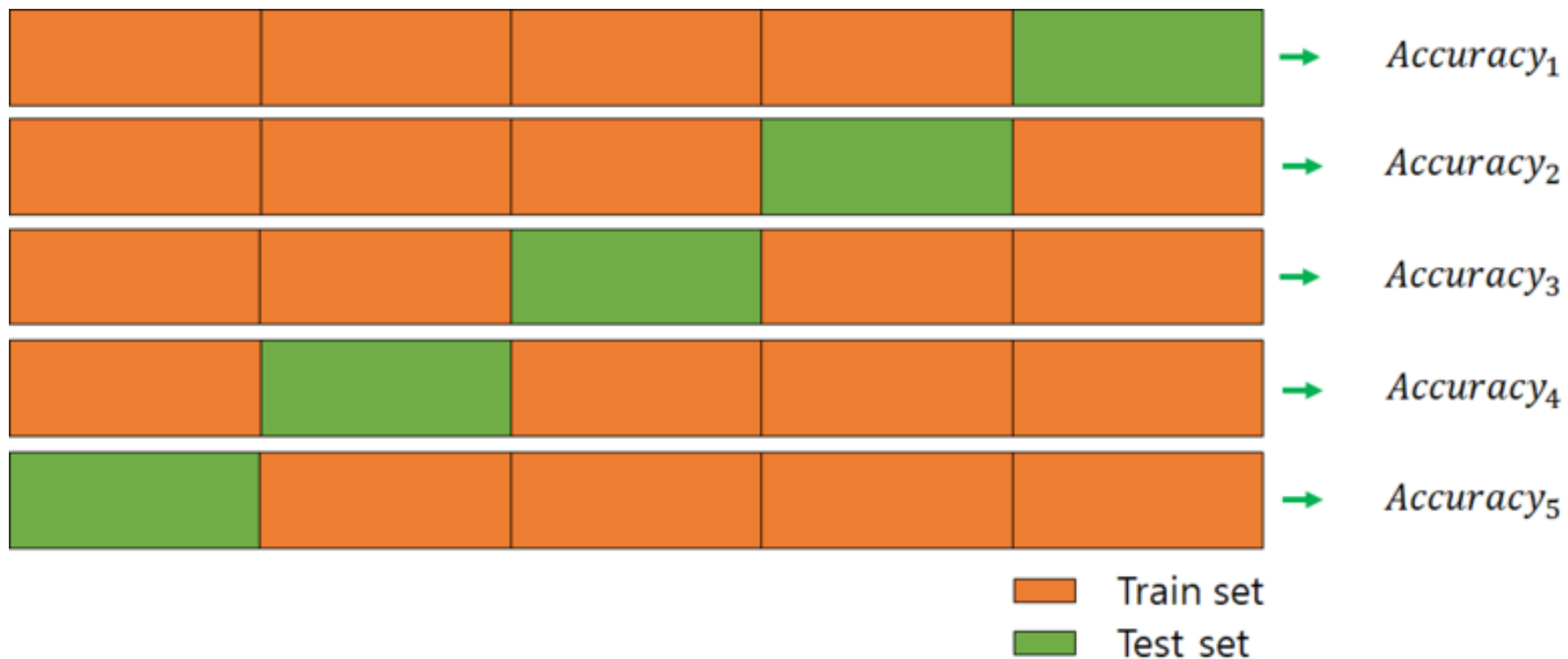
오버피팅 해결하기

1. Feature 수 줄이기
2. 더 많은 데이터 모으기
3. Cross Validation 사용하기
4. Early Stopping, Dropout(딥러닝)
5. Model에 제약 걸기(L1 / L2 regularization)

3. Bias & Variance

4. 우리의 목표 !

Regularization – Cross Validation 교차 검증



장점

1. 모든 데이터 셋을 평가에 활용한다
2. 모든 데이터 셋을 훈련에 활용한다.

단점

1. Iteration 횟수가 많아서 모델 훈련/평가 시간이 오래 걸린다.

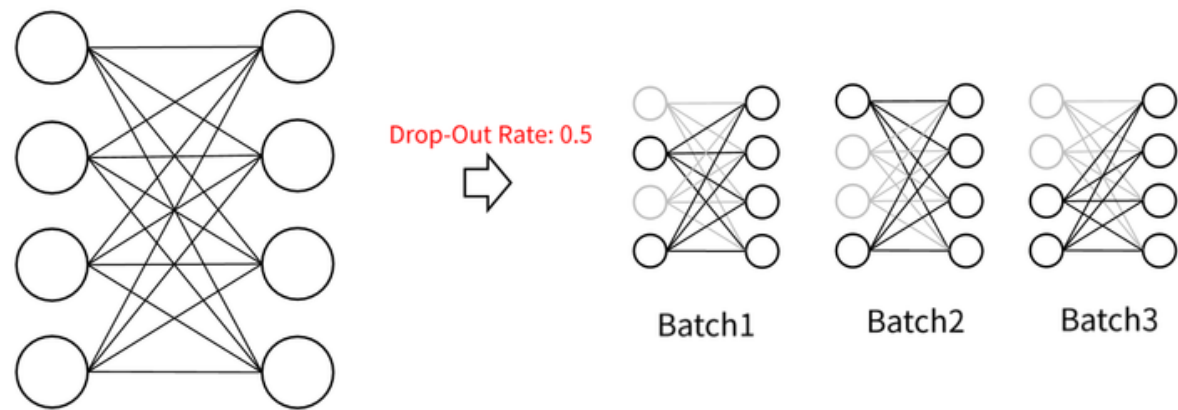
3. Bias & Variance

4. 우리의 목표 !

Regularization – Early Stopping, Dropout (딥러닝)



Validation set의 accuracy가 더 이상 올라가지 않을 때 stop한다.



전체 train 데이터 중 일부를 drop-out 하고 train한다.

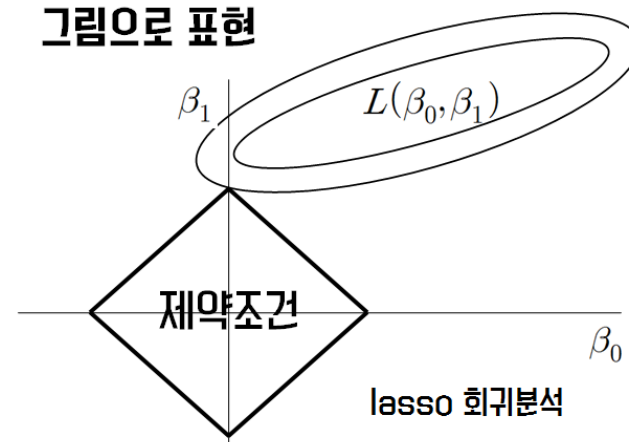
3. Bias & Variance

4. 우리의 목표 !

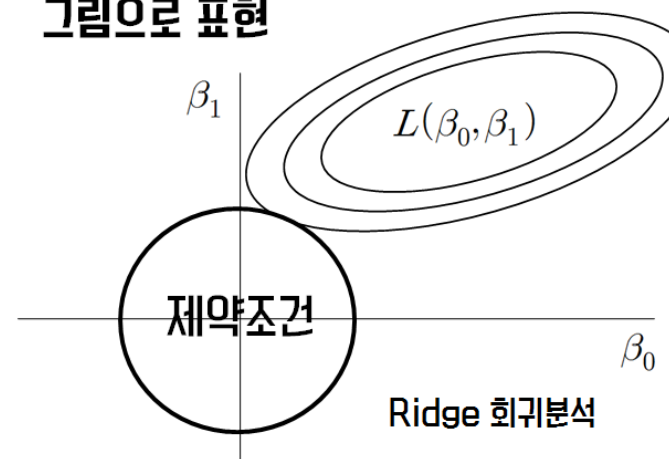
Regularization – L1(Lasso), L2(Ridge)

$$\sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^p |\beta_j| = \text{RSS} + \lambda \sum_{j=1}^p |\beta_j|.$$

그림으로 표현



그림으로 표현



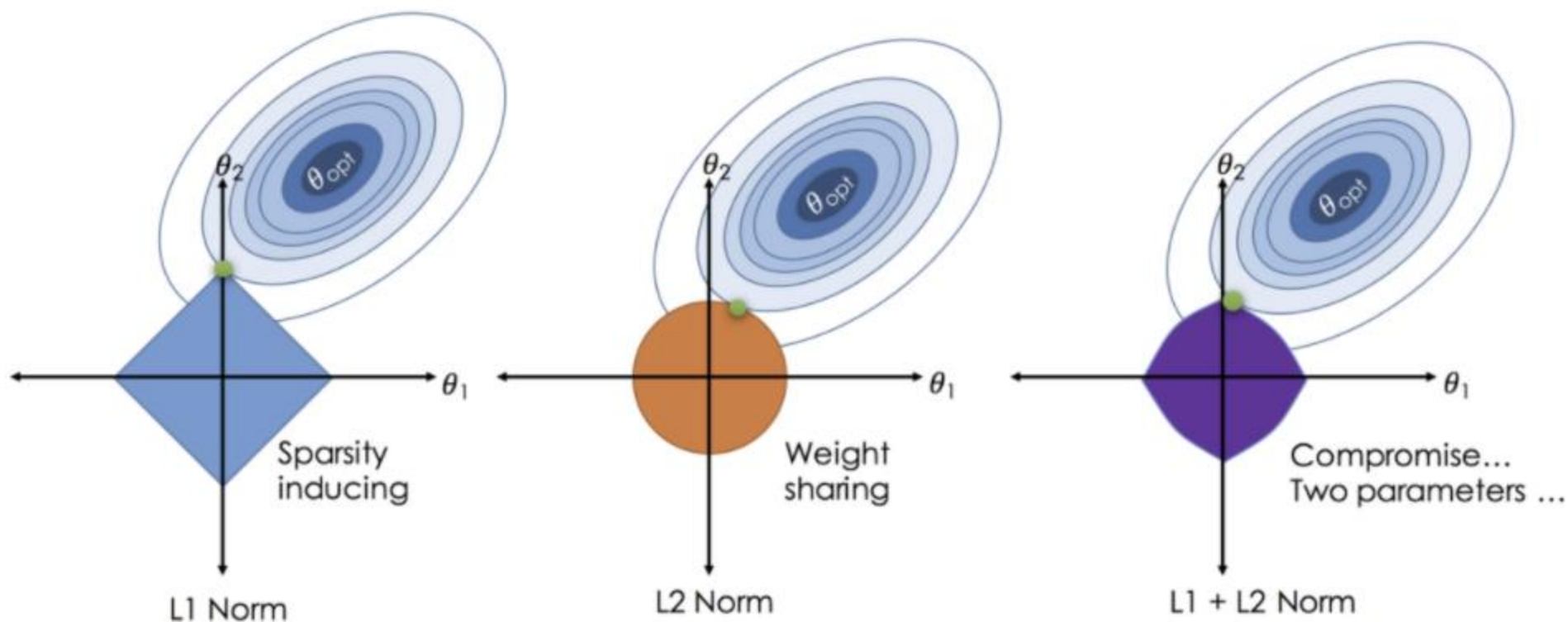
$$\sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^p \beta_j^2 = \text{RSS} + \lambda \sum_{j=1}^p \beta_j^2$$

3. Bias & Variance

4. 우리의 목표 !

Regularization – L1, L2

$$\text{Elastic Net Regression} = \text{RSS}(\beta) + \lambda_1 \sum_{j=1}^p \beta_j^2 + \lambda_2 \sum_{j=1}^p |\beta_j|$$



3. Bias & Variance

4. 우리의 목표 !

Regularization – L1, L2

Lasso L1-norm regularization	Ridge L2-norm regularization	Elastic net Lasso + Ridge
변수 선택 가능	변수 선택 불가능 독립 변수들 간 variance 감소	변수 선택 가능 독립 변수들 간 variance 감소
Closed form solution 존재 X (numerical optimization 이용)	Closed form solution 존재 O (미분으로 구함)	
변수 간 상관관계가 높은 상황에서 릿지에 비해 상대적으로 예측 성능이 떨어짐	변수 간 상관관계가 높은 상황에서 좋은 예측 성능	변수 간 상관관계 반영
크기가 큰 변수 먼저 줄이기	중요하지 않은 변수 먼저 없애기	모두 가능

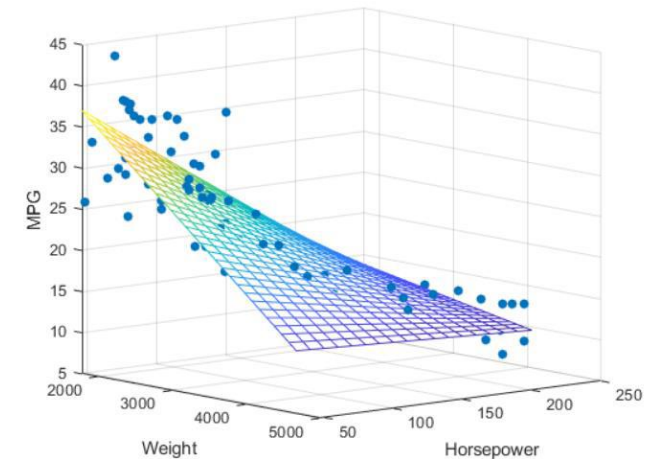
4. Multiple Linear Regression

4. Multiple Linear Regression

Multiple Linear Regression 다중선형회귀 : 입력변수 X 가 여러 개일 때

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p + \varepsilon$$
$$Y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \quad X = \begin{pmatrix} 1 & x_{11} & \dots & x_{1p} \\ \vdots & x_{21} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \dots & x_{np} \end{pmatrix} \quad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} \quad \varepsilon = \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

$= [1 \ x_1 \ \dots \ x_p]$
design matrix



<가정>

추정치는 선형관계여야 한다
독립변수간 다중공선성은 없어야 한다
자기상관이 없어야 한다
등분산이어야 한다
잔차의 가정은 단순선형회귀와 동일

4. Multiple Linear Regression

다중공선성 (Multicollinearity) : 독립변수들 간에 강한 상관관계가 나타나는 문제

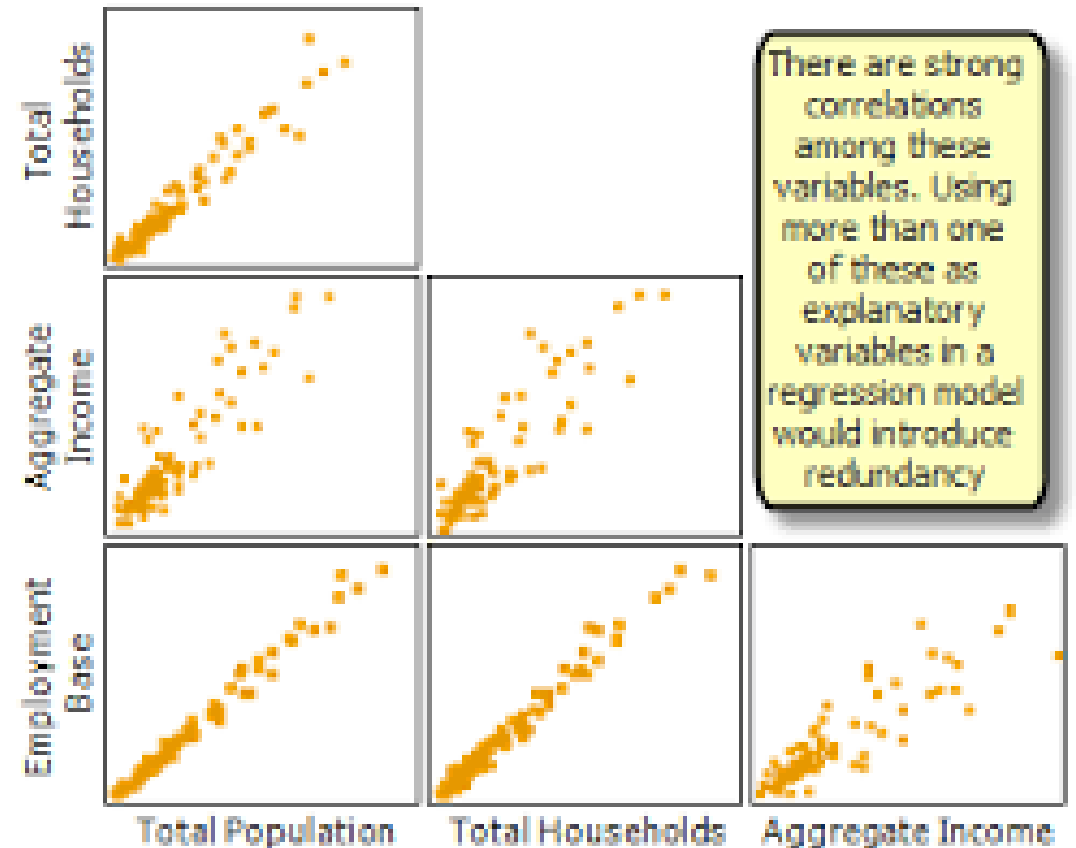
(예)

x1 월평균 음주량

x2 혈중 알코올 농도

Y 학업 성취도

- > X1과 x2는 독립적이라고 보기 어렵다
- > 회귀선의 판단 능력 저하!
- > 회귀 계수에 대한 분산 증가!





4. Multiple Linear Regression

다중공선성 판단하기 - VIF (Variance Inflation Factor) 사용

X_i 를 제외한 다른 X 변수들이 X_i 를 잘 설명하는 지를 평가한다.
VIF가 10 이상인 경우, 다중공선성이 있다고 판단한다.

$$\underline{x_1 = \beta_2 X_2 + \beta_3 X_3 + \cdots + \beta_p X_p + \varepsilon}$$


$$R_1^2$$


$$VIF_1 = \frac{1}{1 - R_1^2}$$

$$VIF_i > 10 \Leftrightarrow \frac{1}{1 - r_i} > 10$$

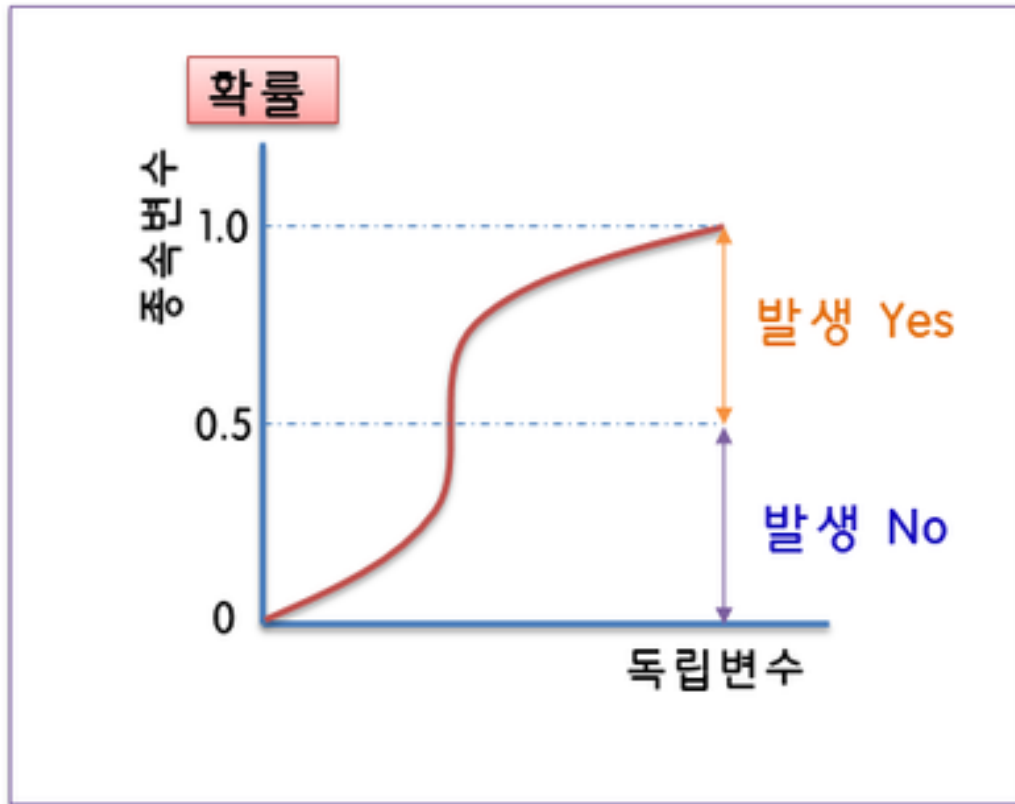
$$1 > 10 - 10r_i$$

$$r_i > 0.9$$

5. Logistic Regression

5. Logistic Regression

로지스틱 회귀 : 종속변수가 범주형 일 때 확률 값을 구하고 label을 예측한다.



로지스틱 회귀분석

$$p(X) = P(\text{success}|X_1, \dots, X_k) = \beta_0 + \beta_1 X_1 + \dots + \beta_k X_k + \text{random error}(\varepsilon)$$



로짓변환

$$\ln(p(X)) = \beta_0 + \beta_1 X_1 + \dots + \beta_k X_k + \varepsilon$$

$$\text{logit} = \ln\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \beta_1 X_1 + \dots + \beta_k X_k + \varepsilon$$

$$Y = p(X) = \frac{e^{\beta_0 + \beta_1 X + \dots + \beta_k X_k}}{1 + e^{\beta_0 + \beta_1 X + \dots + \beta_k X_k}}$$

5. Logistic Regression

로지스틱 회귀식 파라미터 추정 방법

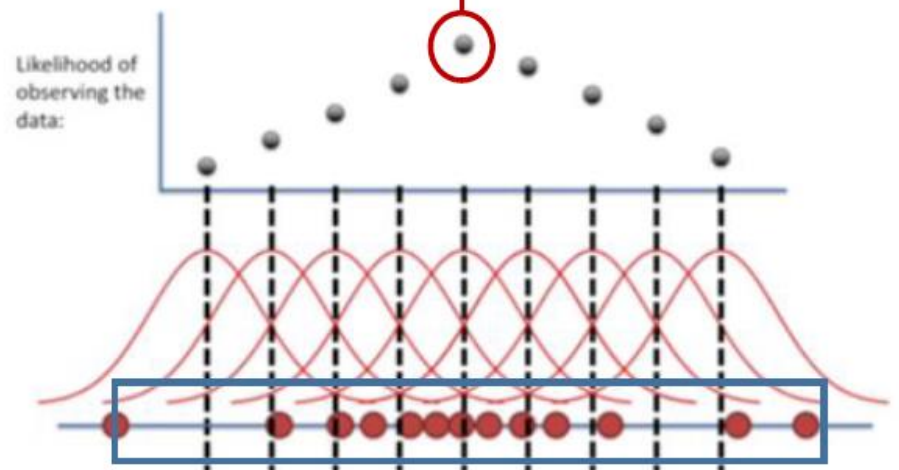
: MLE (Maximum Likelihood Estimator, 최대 우도 추정법)

어떤 모수가 주어졌을 때 원하는 값들이 나올 **가능도**를 최대로 만드는 모수를 추정하는 점 추정 방법

여기서 우도, 가능도, Likelihood란? 관측 값이 어떤 분포에 해당 할 확률 !

$$\hat{\theta} = \operatorname{argmax} L(\theta)$$

<-> 확률 : 모수로부터 다음과 같이 관측될 확률 !




고정된 n 개의 x (관측값)

5. Logistic Regression

로지스틱 회귀식 파라미터 추정 방법

Likelihood function : 전체 표본 집합의 결합확률밀도 함수

$$P(x|\theta) = \prod_{k=1}^n P(x_k|\theta)$$
$$L(\theta|x) = \log P(x|\theta) = \sum_{i=1}^n \log P(x_i|\theta)$$

$$\frac{\partial}{\partial \theta} L(\theta|x) = \frac{\partial}{\partial \theta} \log P(x|\theta) = \sum_{i=1}^n \frac{\partial}{\partial \theta} \log P(x_i|\theta) = 0$$

5. Logistic Regression

로지스틱 회귀식 mle 유도 예시 (정규분포)

정규분포의 pdf

$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \longrightarrow$$

양변에 로그

$$\mathcal{L}(\theta) = \prod_i f_{\mu,\sigma}(x_i) = \prod_i \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x_i - \mu)^2}{2\sigma^2}\right)$$
$$\mathcal{L}^*(\theta) = -\frac{n}{2} \log 2\pi - n \log \sigma - \frac{1}{2\sigma^2} \sum_i (x_i - \mu)^2$$

σ 에 대해 편미분

$$\frac{\partial}{\partial \sigma} \mathcal{L}^*(\theta) = -\frac{n}{\sigma} + \frac{1}{\sigma^3} \sum_i (x_i - \mu)^2 = 0$$
$$\therefore \sigma^2 = \sum_i (x_i - \mu)^2 / n = \text{MSE}$$

6. 과제 및 참고

6. 과제

이번주 과제는 오늘 배운 내용의 코드를 구글링해서 직접 수행하고 결과를 정리, 해석하기입니다 😊

선형, 다중선형, 로지스틱회귀 예제를 한 개씩 총 3개를 정리하시면 됩니다. 최대한 다른 분들과 안 겹치는 것을 해주세요!

예를 들어 임의의 X data를 만들거나 가져와서 LinearRegression을 수행한 다음, 추정회귀식을 구하고 회귀직선도 그려보고, 변수 간 상관관계의 유무 확인 & 오버피팅, 언더피팅은 없는지 등의 분석을 수행하고 해석도 해주시면 됩니다!

6. 참고

<https://m.blog.naver.com/x3x1121/222139532976>

<https://www.youtube.com/watch?v=dBLZg-RqoLg>

<https://m.blog.naver.com/ckdgus1433/221599517834>

<https://m.blog.naver.com/jeonghj66/222004874975>

<https://m.blog.naver.com/PostView.nhn?blogId=wjddudwo209&logNo=220177096998&proxyReferer=https:%2F%2Fwww.google.com%2F>

<https://soobarkbar.tistory.com/30>

<https://laoonlee.tistory.com/12>

감사합니다 😊