Statistical Machine Learning

4주차

담당: 18기 신인수

3주차 과제 우수자

김재훈님☺

코드를화면공유해주시고, 간단히설명해주세요!

리마인드: **매주 수요일 22시까지** 과제를 제출해주셔야 합니다! (github)

과제제출시, 파일명은 기존.ipynb 파일이름을 유지하되, '_이름'만 추가해주세요 ☺ (ML_week4_HW_방서연)

0. Conditional Expectation

1. Regression

2. Linear Regression

3. Regularization

4. Regression Diagnostics

0. Conditional Expectation

Conditional Expectation

Conditional Distribution (조건부 분포)

- Recap) 조건부 확률: B라는 조건이 주어졌을 때 A의 확률

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

- Y라는 조건이 주어졌을 때, X의 분포

$$\boldsymbol{f(X|Y) = \frac{f(x,y)}{f(y)}} = \frac{P(X \cap Y)}{P(Y)}$$

Conditional Expectation

Conditional Expectation (조건부 기댓값)

- Y라는 조건이 주어졌을 때, X의 기댓값

$$E(X|Y) = \int_{-\infty}^{\infty} x\, f(x|y)dx, X \text{ is continuous random variable}$$

$$E(X|Y) = \sum_x xf(x|y), X \text{ is discrete random variable}$$

Conditional Variance (조건부 분산)

- Y라는 조건이 주어졌을 때, X의 분산 (역시 편차 제곱의 평균)

$$Var(X|Y) = E[\left(X - E(X|Y)\right)^2|Y]$$

Conditional Expectation

Properties

$$E(X) = E[E(X|Y)]$$

$$Var(X) = Var[E(X|Y)] + E[Var(X|Y)]$$

1. Regression

Supervised Learning

Objective

Regresion

지도학습: y가 연속형

Classification

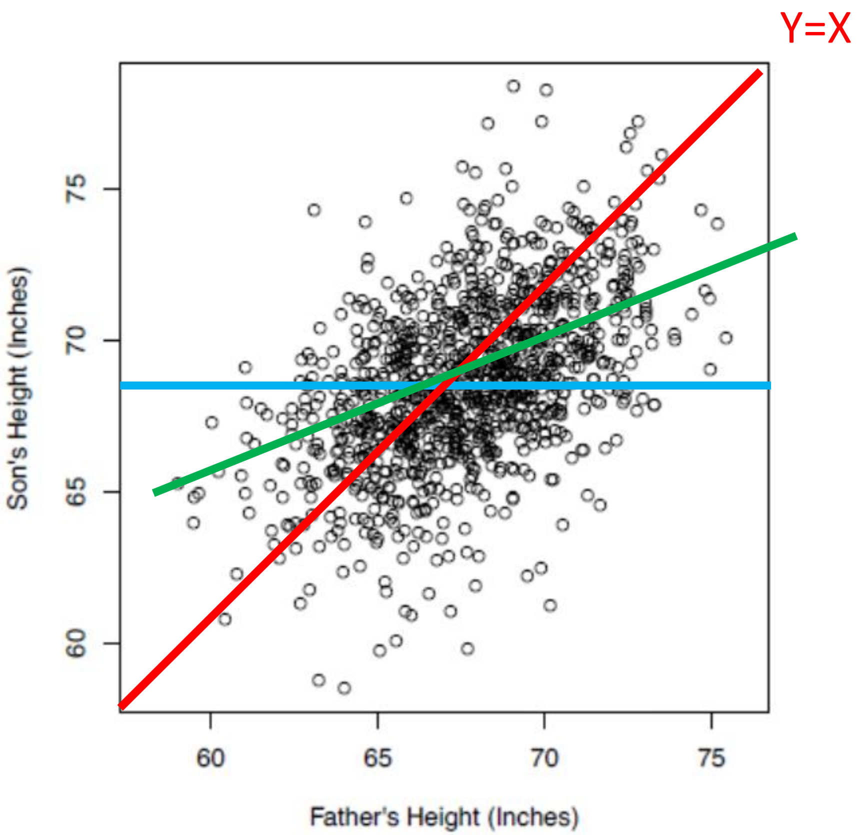
지도학습: y가 범주형

Regression

Galton의 height 데이터.
X : 아버지의 키
Y : 아들의 키

두 변수의 관계를
Y = aX + b
로 표현하려고 할 때, a와 b를
어떻게 정할 것인가?

→ **원칙이 필요하다**



이런 식으로 아무렇게나
그어버리면 곤란하다.

여전히 우리는 조건부
기댓값을 모델링하고 싶다.

Regression

Recap: Bias and Variance tradeoff & MSE

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

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

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

$$= [Var(\hat{f}(x)) + Bias(\hat{f}(x))^2] + Var(\epsilon)$$

$$= Reducible\ Error + Irreducible\ Error$$

- 원칙1: **Bias = 0으로 만들고(unbiased estimator)**, 분산이 최소인 직선을(추정량) 구하겠다 → Linear Regression (1장)
- 원칙2: Bias가 있더라도 전체 MSE를 최소화시키는 직선을(추정량) 구하겠다 → LASSO, Ridge 등의 regularization (2장)

Regression

어떤 원칙으로 직선을 만들게 되는가?

$$SST = SSR + SSE$$

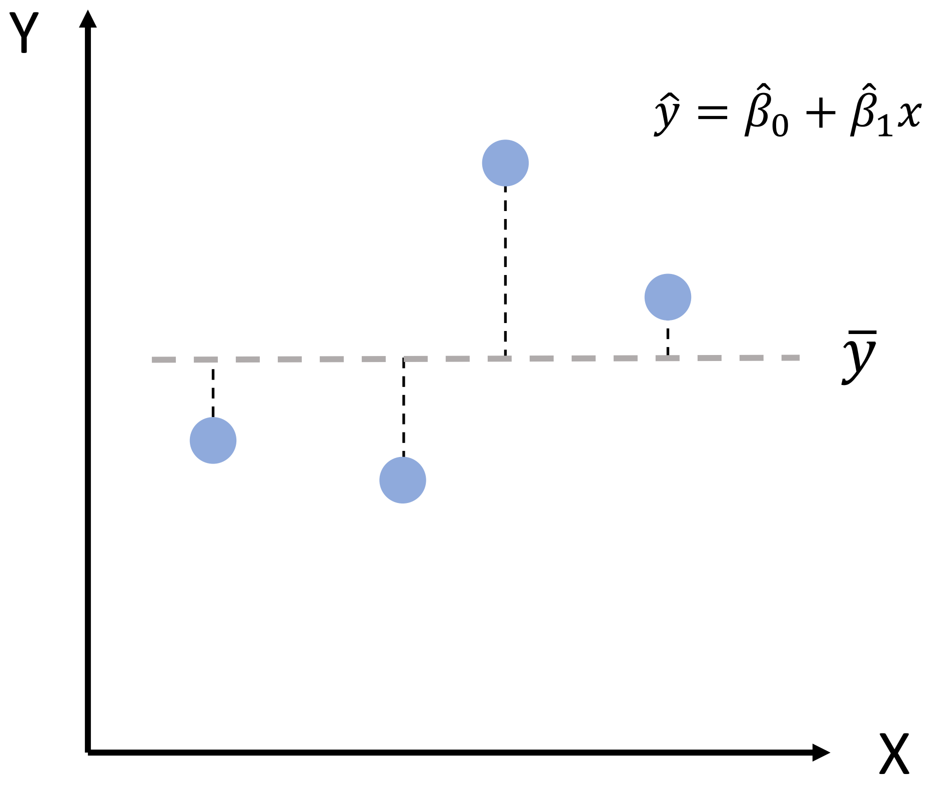
Sum of squares total = sum of squares regression + sum of squares error

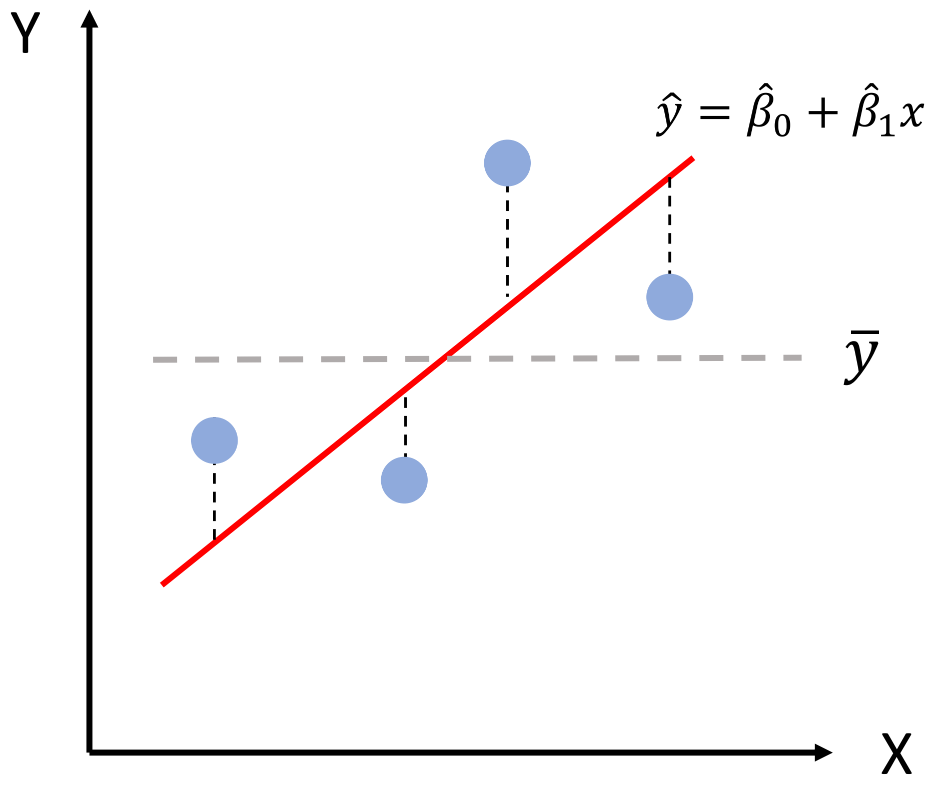
$$\sum_{i=1}^{n}(y_i - \bar{y})^2 = \sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2 + \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

데이터의 분산 = 회귀모형이 설명 가능한 부분 + 불가능한 부분

→ SSE를 최소화하는 직선 찾아라!

Regression





→ SSE를 최소화하는 직선 찾기 → **LSE (Least Squares Estimate)**

2. Linear Regression

Linearity & Linear Model

- Linearity?

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \cdots + \beta_p X_{pi} + \epsilon_i$$

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \beta_3 X_{1i} X_{2i} + \epsilon_i$$

$$Y_i = \beta_0 + \beta_1 X_i + \beta_2 X_i^2 + \cdots + \beta_p X_i^p + \epsilon_i$$

Linear Regression

예전에 사용하던 슬라이드
중요한 부분 짚어가면서 배워봅시다!

$r = f(x) + \varepsilon$

Estimator this one directly! = $g(x|w) = w_1x_1+\cdots+w_dx_d+w_0 = w^T x + w_0$

Assume as Linear model

[Assumptions of error] : Normality & Homoscedasticity & independent → 회귀진단에 이용!

$\varepsilon \sim N(0, \sigma^2)$

$r \sim N(g(x|w), \sigma^2)$

$$\boxed{log \prod_{t=1}^{N} p(r^t|x^t) = log \prod_{t}^{N} \frac{1}{\sqrt{2\pi}\sigma} exp\left[-\frac{(r^t - g(x^t|w))^2}{2\sigma^2}\right] \rightarrow \text{Maximize!}}$$

→ **갑자기 MLE??? → 다음 슬라이드**

Minimize : Loss function E(w|x) = $\frac{1}{2}\sum_{t=1}^{N}[(r^t - g(x^t|w))^2]$ → MSE 최소화 $\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$

Linear Regression

- 분명 MLE와 LSE는 원칙이 다르다.
  - MLE: likelihood을 최대로 만드는 추정량 (estimator)
  - LSE: MSE, 혹은 SSE를 최소로 만드는 추정량
- **그러나 오차의 정규성, 등분산성, 독립성을 만족할 때 두 추정량은 같다.**

$$\mathbf{y} = \mathbf{X}\beta + \epsilon \qquad \epsilon \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$$

MLE

$$L(\beta, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_i - \mathbf{x}_i^T \beta)^2}{2\sigma^2}\right)$$

Maximize $\ell(\beta, \sigma^2) = -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \mathbf{x}_i^T \beta)^2$

Minimize $-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \mathbf{x}_i^T \beta)^2$

Equivalent to $S(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta)$

LSE

Minimize $S(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta)$

$$S(\beta) = \mathbf{y}^T\mathbf{y} - 2\mathbf{y}^T\mathbf{X}\beta + \beta^T\mathbf{X}^T\mathbf{X}\beta$$

$$\frac{\partial S(\beta)}{\partial \beta} = -2\mathbf{X}^T\mathbf{y} + 2\mathbf{X}^T\mathbf{X}\beta$$

Let $-2\mathbf{X}^T\mathbf{y} + 2\mathbf{X}^T\mathbf{X}\beta = 0$ → $\mathbf{X}^T\mathbf{X}\beta = \mathbf{X}^T\mathbf{y}$

$$\beta = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

Therefore $\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$

Gauss-Markov Theorem

$\hat{\beta}$가 LSE이면서 비편향 추정량(unbiased estimator)일때, 임의의 비편향추정량 $\tilde{\beta}$에 대해 다음을 만족

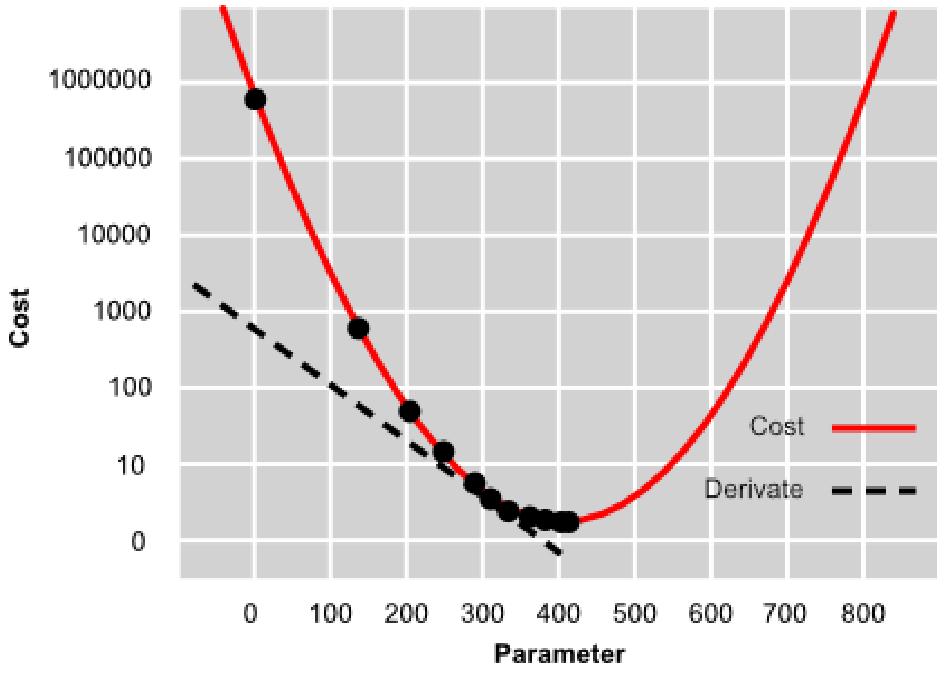
$$Var[\hat{\beta}|X] \leq Var[\tilde{\beta}|X]$$

따라서 LSE가 BLUE (best linear unbiased estimator)를 만족!

Gradient Descent

Minimize : Loss function E(w|x) = $\frac{1}{N}\sum_{t=1}^{N}[(r^t - g(x^t|w))^2] = MSE(Mean\ Squared\ Error)$

$w^* = argmin_w E(w|x)$ $\qquad w_{j+1} \leftarrow w_j - \eta\frac{\partial E}{\partial w_j}$ iteratively



- 컴퓨터가 어떻게 MSE를 최소화 하는 추정량을 찾는가 → Gradient Descent
- MSE가 2차함수이기 때문에 금방 찾을 수 있다.
- 다른 Loss 함수를 사용하는 경우도 사용 가능
  - Ex) MAE, RMSE, Huber 등등

Least Square Estimation

Minimize : Loss function E(w|x) = $\frac{1}{2}\sum_{t=1}^{N}[(r^t - g(x^t|w))^2]$

** $g(x^t|w) = w_1x^t + w_0$: 1st order

$$w^* = argmin_w E(w|x) \rightarrow \frac{\partial E}{\partial w_1} = 0 \;\&\; \frac{\partial E}{\partial w_0} = 0$$

$$A = \begin{bmatrix} N & \sum_t x^t \\ \sum_t x^t & \sum_t (x^t)^2 \end{bmatrix} \quad \mathrm{w} = \begin{bmatrix} w_0 \\ w_1 \end{bmatrix} \quad y = \begin{bmatrix} \sum_t y \\ \sum_t r^t x^t \end{bmatrix}$$

$$w^* = A^{-1}y$$

~~Multivariate Regression~~ Multiple Regression

$$r^t = g(\mathbf{x}^t | w_0, w_1, \ldots, w_d) + \epsilon = w_0 + w_1 x_1^t + w_2 x_2^t + \cdots + w_d x_d^t + \epsilon$$

$$E(w_0, w_1, \ldots, w_d | \mathcal{X}) = \frac{1}{2} \sum_t (r^t - w_0 - w_1 x_1^t - w_2 x_2^t - \cdots - w_d x_d^t)^2$$

$$\begin{aligned}
\sum_t r^t &= N w_0 + w_1 \sum_t x_1^t + w_2 \sum_t x_2^t + \cdots + w_d \sum_t x_d^t \\
\sum_t x_1^t r^t &= w_0 \sum_t x_1^t + w_1 \sum_t (x_1^t)^2 + w_2 \sum_t x_1^t x_2^t + \cdots + w_d \sum_t x_1^t x_d^t \\
\sum_t x_2^t r^t &= w_0 \sum_t x_2^t + w_1 \sum_t x_1^t x_2^t + w_2 \sum_t (x_2^t)^2 + \cdots + w_d \sum_t x_2^t x_d^t \\
&\vdots \\
\sum_t x_d^t r^t &= w_0 \sum_t x_d^t + w_1 \sum_t x_d^t x_1^t + w_2 \sum_t x_d^t x_2^t + \cdots + w_d \sum_t (x_d^t)^2
\end{aligned}$$

22

$$\mathbf{X} = \begin{bmatrix} 1 & x_1^1 & x_2^1 & \cdots & x_d^1 \\ 1 & x_1^2 & x_2^2 & \cdots & x_d^2 \\ \vdots & & & & \\ 1 & x_1^N & x_2^N & \cdots & x_d^N \end{bmatrix}, \boldsymbol{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_d, \end{bmatrix}, \boldsymbol{r} = \begin{bmatrix} r^1 \\ r^2 \\ \vdots \\ r^N \end{bmatrix}$$

Then the normal equations can be written as

$$\mathbf{X}^T \mathbf{X} \boldsymbol{w} = \mathbf{X}^T \boldsymbol{r}$$

$$\boldsymbol{w} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \boldsymbol{r}$$

$$\boxed{w^* = A^{-1} y}$$

※주의

Multiple Linear Regression
종속 변수 1개

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \epsilon$$

Multivariate Linear Regression
종속변수 여러 개

$$Y_1 = \beta_{01} + \beta_{11} X_1 + \beta_{21} X_2 + \cdots + \beta_{p1} X_p + \epsilon_1$$

$$Y_2 = \beta_{02} + \beta_{12} X_1 + \beta_{22} X_2 + \cdots + \beta_{p2} X_p + \epsilon_2$$

$$Y_3 = \beta_{03} + \beta_{13} X_1 + \beta_{23} X_2 + \cdots + \beta_{p3} X_p + \epsilon_3$$

Polynomial Regression

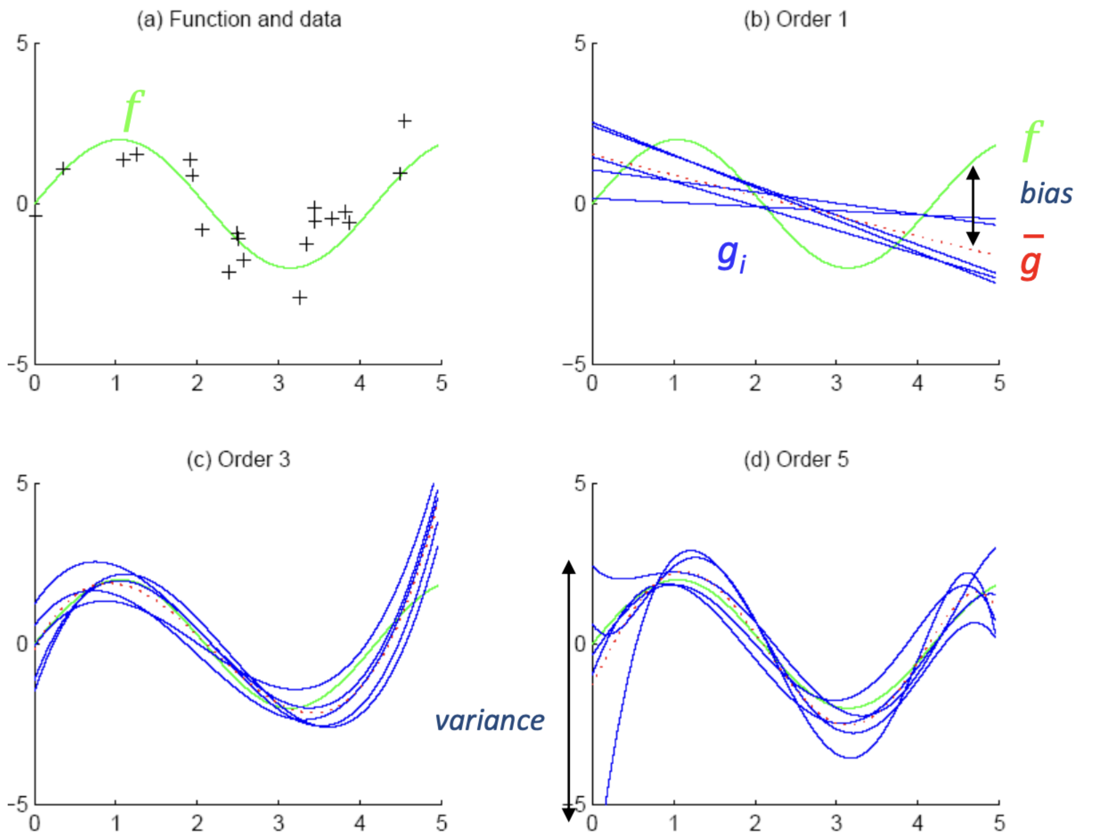
$$g\left(x^t \mid w_k, \ldots, w_2, w_1, w_0\right) = w_k\left(x^t\right)^k + \cdots + w_2\left(x^t\right)^2 + w_1 x^t + w_0$$

$$\mathbf{A} = \begin{bmatrix} N & \sum_t x^t & \sum_t (x^t)^2 & \cdots & \sum_t (x^t)^k \\ \sum_t x^t & \sum_t (x^t)^2 & \sum_t (x^t)^3 & \cdots & \sum_t (x^t)^{k+1} \\ \vdots & & & & \\ \sum_t (x^t)^k & \sum_t (x^t)^{k+1} & \sum_t (x^t)^{k+2} & \cdots & \sum_t (x^t)^{2k} \end{bmatrix}$$

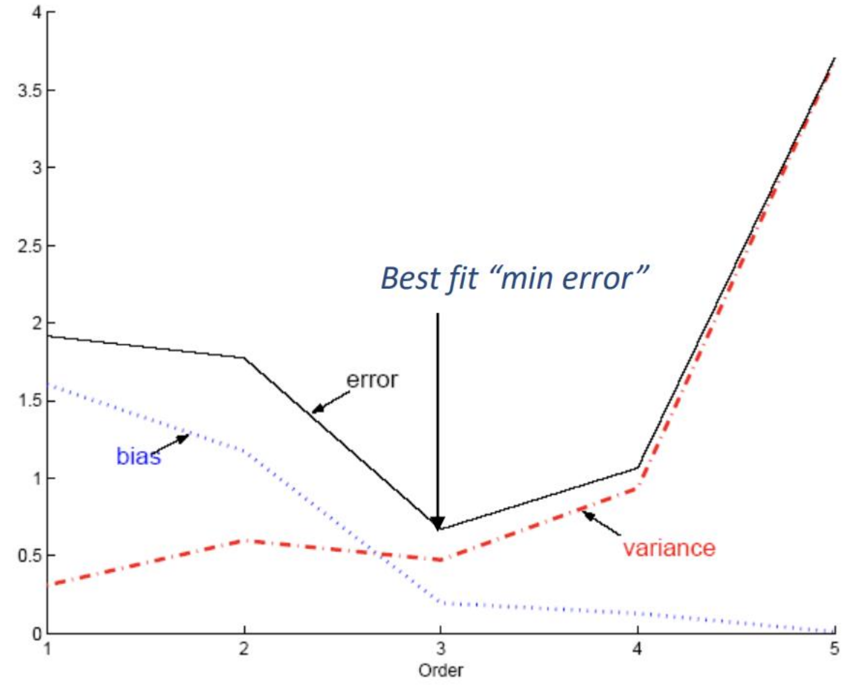
$$\boldsymbol{w} = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \\ w_k \end{bmatrix}, \quad \boldsymbol{y} = \begin{bmatrix} \sum_t r^t \\ \sum_t r^t x^t \\ \sum_t r^t (x^t)^2 \\ \vdots \\ \sum_t r^t (x^t)^k \end{bmatrix}$$

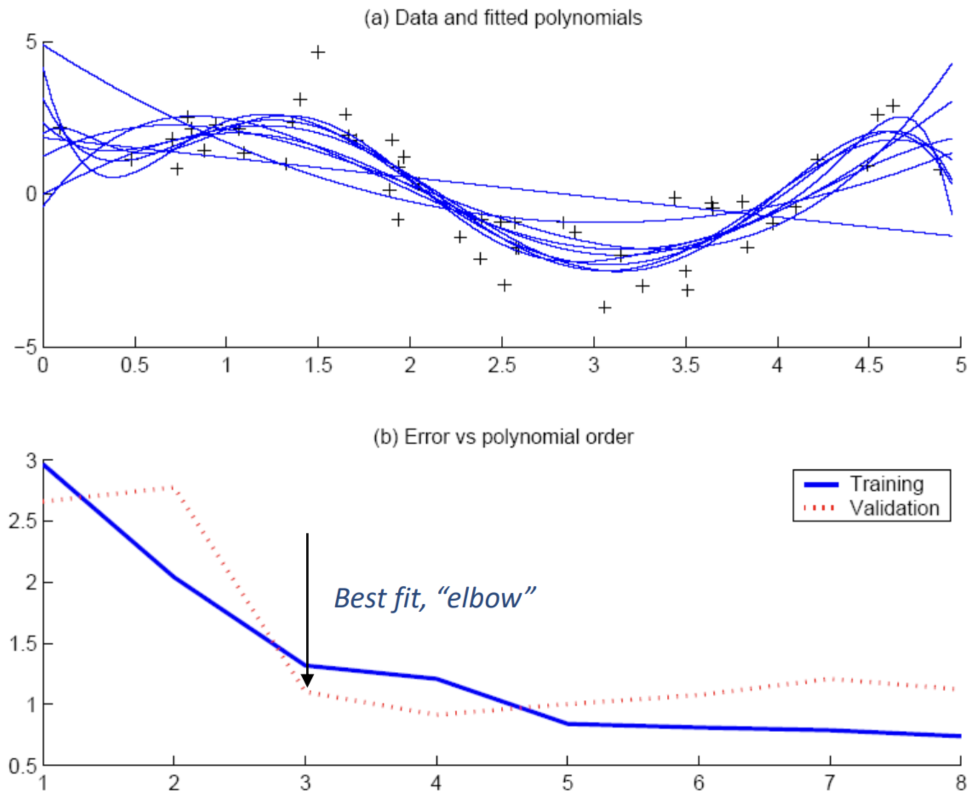
$$\boxed{w^* = A^{-1} y}$$

Polynomial Regression



Model Selection





Cross Validation

To estimate generalization error, we need data unseen during training.

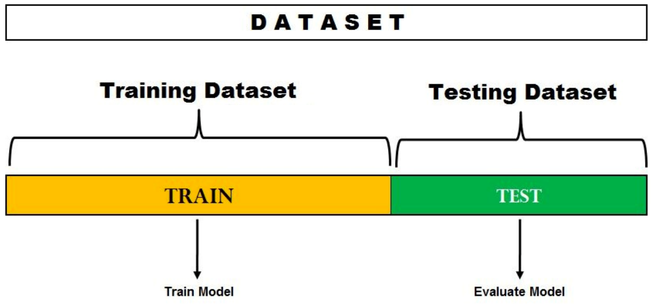
= Test Error 추정

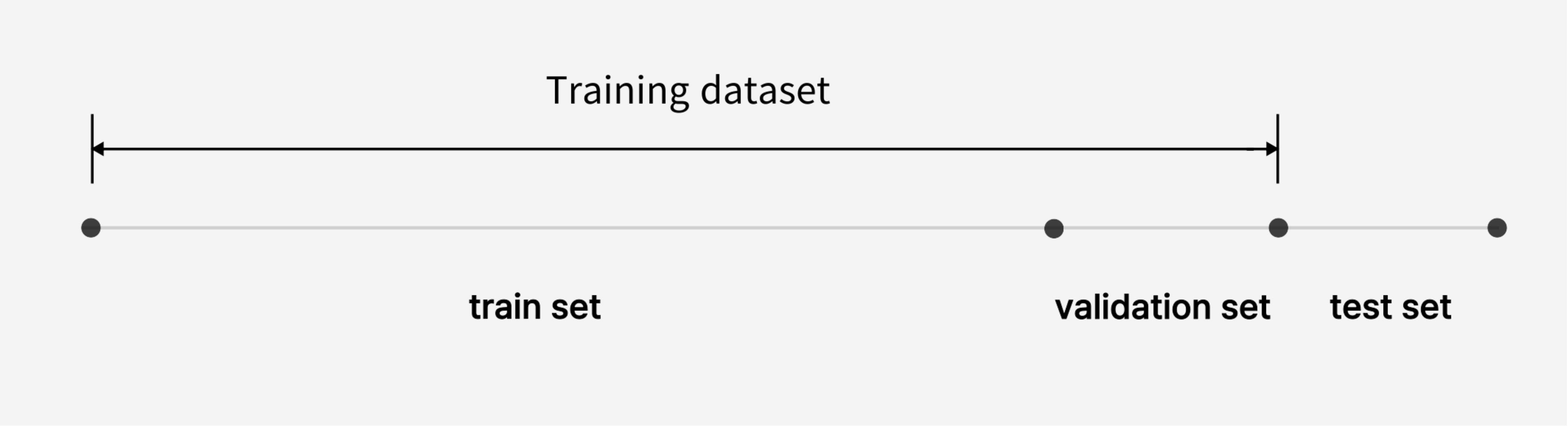
We split the data as

- – Training set (50%)
- – Validation set (25%)
- – Test (publication) set (25%)

Measure generalization accuracy by testing on data unused during training

Hold out





Regularization

Penalize complex models

- E'=error on data + λ*model complexity

* If λ increases, variance decreases, but bias increases

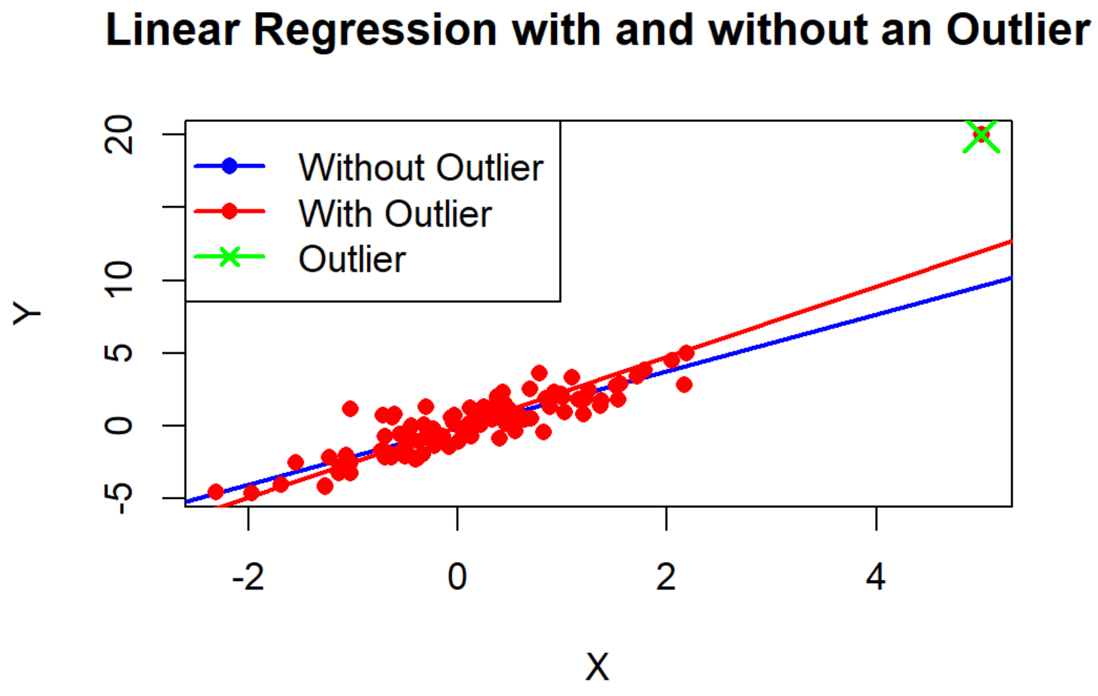
→ Penalty term이 생기는 순간
unbiased 포기 (Gauss-Markov Theorem)

In regression...

Regularization (L2): $$E(\mathbf{w}|\mathcal{X})=\frac{1}{2}\sum_{t=1}^{N}\left[r^t-g\left(x^t|\mathbf{w}\right)\right]^2+\boxed{\lambda\sum_i w_i^2}$$

Loss Function인 MSE에 Penalty Term
추가 → **그런데 이게 왜 필요?**
→ **다음 슬라이드**

Linear Regression: Limitations



- Linear Regression은 outlier에 취약!
- 이를 완화하기 위해 Regularization 사용
  - Ridge, LASSO

3. Regularization

Distance

- Distance measure

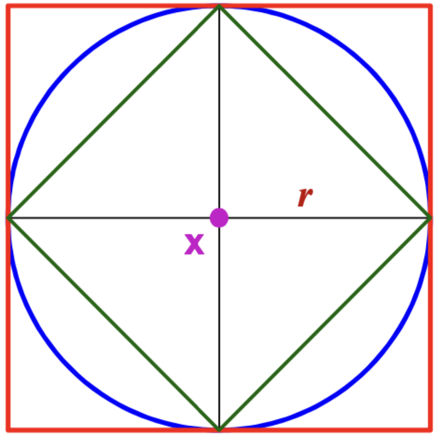
$$d(\mathbf{u},\mathbf{v}) = (\sum|u_i - v_i|^2)^{\frac{1}{2}} = ||\mathbf{u}-\mathbf{v}||_2$$ *Euclidean (L2 norm)*

$$d(\mathbf{u},\mathbf{v}) = \sum|u_i - v_i| = ||\mathbf{u}-\mathbf{v}||_1$$ *Manhattan (L1 norm)*

$$d(\mathbf{u},\mathbf{v}) = (\sum|u_i - v_i|^p)^{\frac{1}{p}} = ||\mathbf{u}-\mathbf{v}||_p$$ *Minkowski (Lp norm)*

$$d(\mathbf{u},\mathbf{v}) = \sqrt{(\mathbf{x}-\boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x}-\boldsymbol{\mu})}$$ *Mahalanobis Distance*

Distance



- Green: All points **y** at distance $\mathbf{L_1(x, y) = r}$ from point **x**
- Blue: All points **y** at distance $\mathbf{L_2(x, y) = r}$ from point **x**
- Red: All points **y** at distance $\mathbf{L_\infty(x, y) = r}$ from point **x**

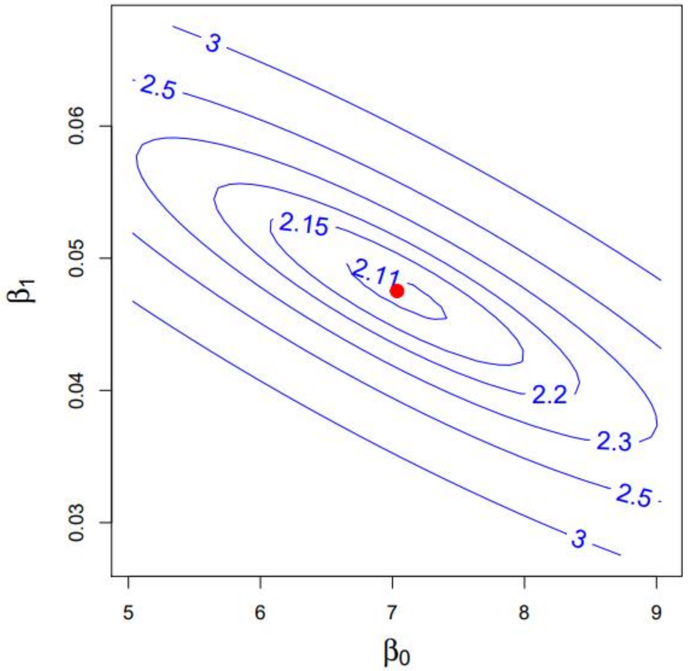
$$\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i|$$

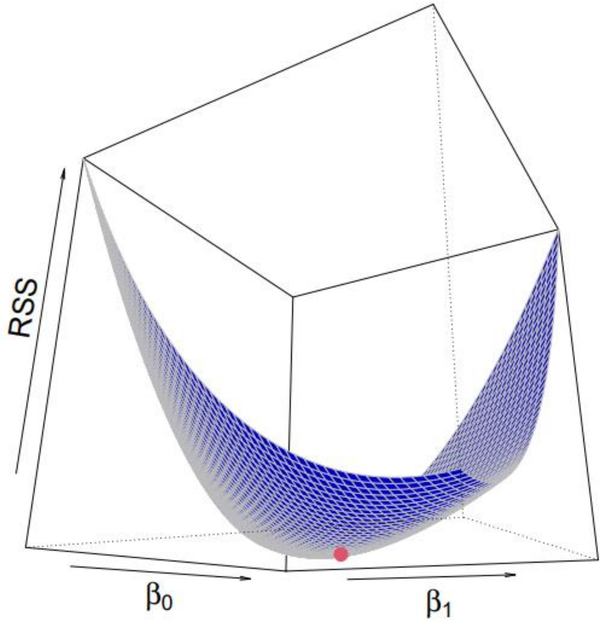
$$\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$$

$$\|\mathbf{x}\|_\infty = \max_{i=1,\ldots,n} |x_i|$$

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p}$$

SSE와 추정량의 contour

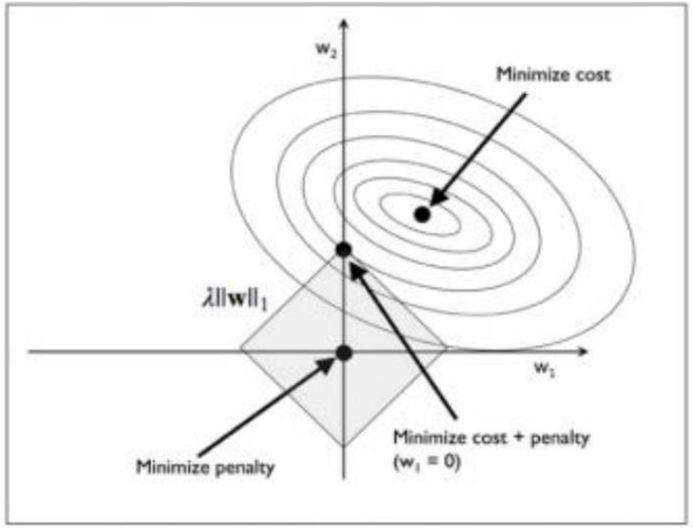




- 실제로 시뮬레이션을 해보면 이차 곡면이 나온다.
- 붉은 점이 LSE
- RSS = residual sum of square = SSE

→ 다음 슬라이드에 LASSO, Ridge regression 설명을 위함!

Lasso Regression



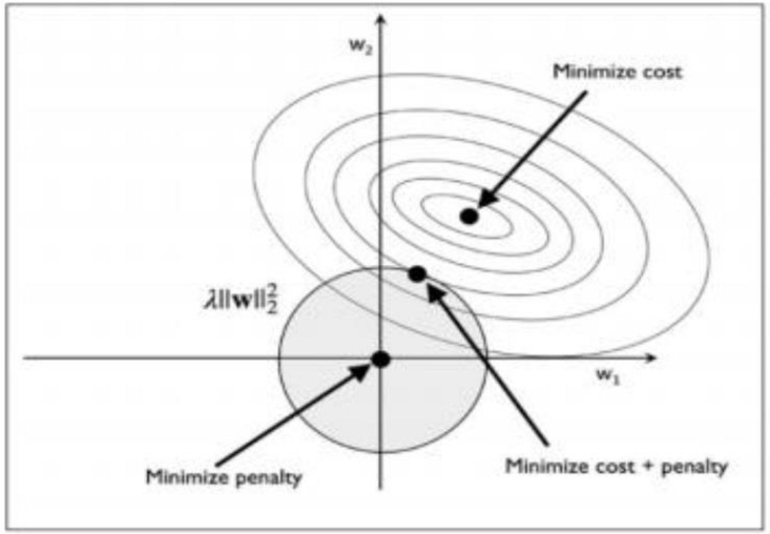
- LASSO (Least Absolute Shrinkage and Selection Operator)

$$(\hat{\boldsymbol{\beta}}^{\lambda,1} =)\ \hat{\boldsymbol{\beta}}_{LASSO} = \underset{\boldsymbol{\beta}}{argmin}\ (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \lambda\,||\boldsymbol{\beta}||_1$$

where $||\boldsymbol{\beta}||_1 = \sum_j^p |\beta_j|$ → Lagrangian Form

$$\Leftrightarrow \hat{\boldsymbol{\beta}}_{LASSO} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}(\boldsymbol{Y} - \boldsymbol{X\beta})^T(\boldsymbol{Y} - \boldsymbol{X\beta})\text{ , with constraint } \sum_{i=1}^p |\beta_i| \le t$$

Ridge Regression

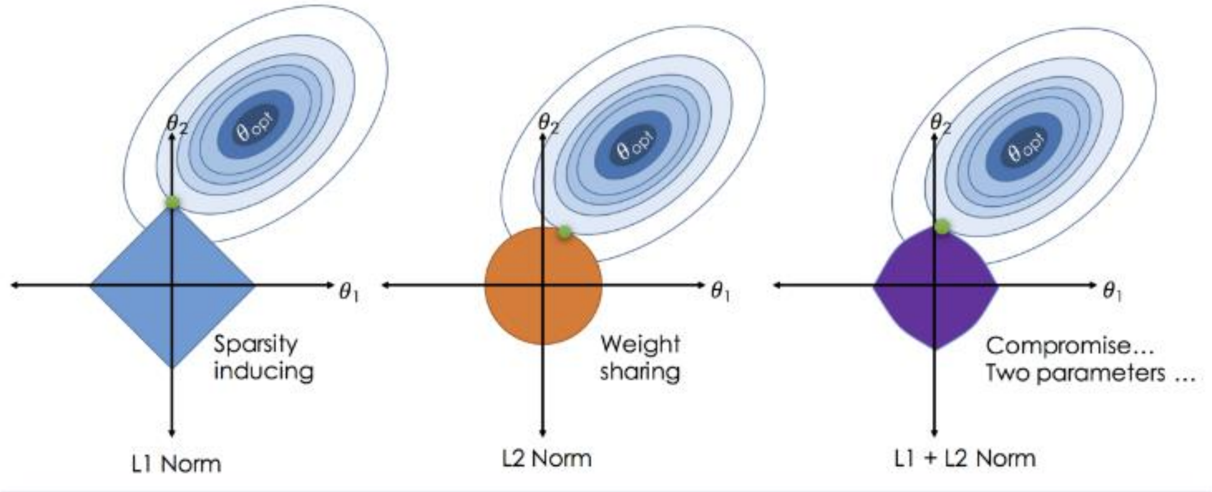


Ridge Regression solves → Lagrangian Form

$$\min_{\boldsymbol{\beta}} (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \lambda||\boldsymbol{\beta}||_2^2 \qquad (L2\ penalty)$$

$\Leftrightarrow \hat{\boldsymbol{\beta}}_{Ridge} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}(\boldsymbol{Y} - \boldsymbol{X\beta})^T(\boldsymbol{Y} - \boldsymbol{X\beta})$, with constraint $\sum_{i=1}^{p} \beta_i^2 \leq t$

Elastic-Net Regression



$$\frac{\sum_{i=1}^{n}(y_i - x_i^J \hat{\beta})^2}{2n} + \lambda\left(\frac{1-\alpha}{2}\sum_{j=1}^{m}\hat{\beta}_j^2 + \alpha\sum_{j=1}^{m}|\hat{\beta}_j|\right)$$

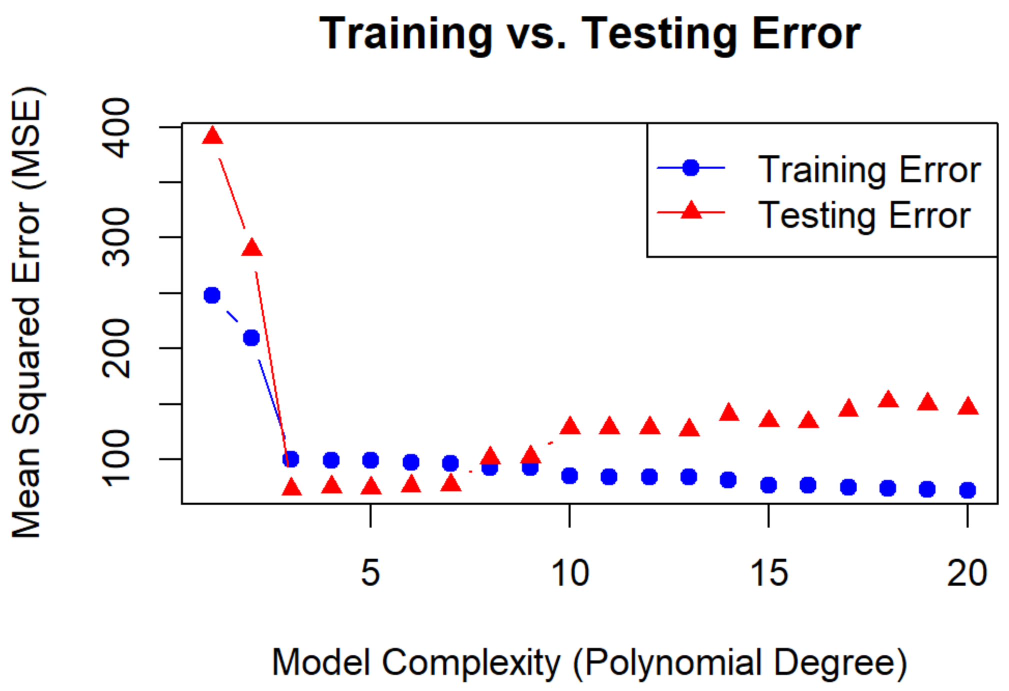
4. Regression Diagnostics

Regression Diagnostics (회귀 진단)

종류

1. Optimism based method → AIC, BIC
2. Test error based → Cross Validation, Bootstrap
3. Multicollinearity
4. Residual plot: 오차의 iid 가정 체크 (독립, 등분산, 정규성)
5. Goodness of Fit Test

Optimism of Training Error



- **우리는 test error을 최소화 하고 싶다!**
- 그러나 우리가 알고 있 수 있는 것은 training error 뿐
- 그러나 training error은 test error보다 낮다.
- Train error: 단조 감소
- Test error: 감소 → 증가

$$op = Err_{in} - \overline{err}$$

- op: optimism
- Err_{in}: in-sample error
- $\overline{err}$: training error

Optimism of Training Error

1. Optimism based method → AIC, BIC

AIC와 BIC는 optimism (op)를 추정
이 둘을 최소화하는 모델 찾기

$$\mathrm{AIC} = 2k - 2\ln(L)$$

k = number of estimated parameters
L = maximum likelihood of model

$$\mathrm{BIC} = \ln(n)k - 2\ln(L)$$

k = number of estimated parameters
L = maximum likelihood of model
n = sample size

2. Test error based → Cross Validation, Bootstrap

두 방법은 test error을 직접 추정

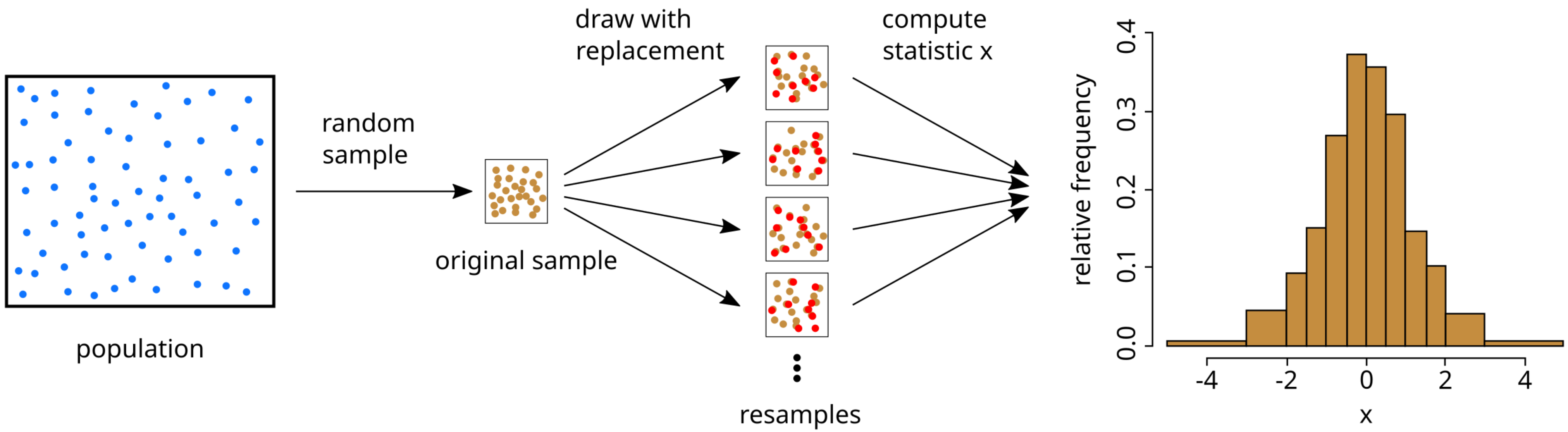
Cross validation: sample size N, fold K일 때
$\kappa: \{1, \cdots N\} \rightarrow \{1, \cdots, K\}$: indexing function
$\hat{f}^{-k}(x)$: fitted function with kth chunk left out

$$\widehat{test\ error} = CV(\hat{f}) = \frac{1}{N}\sum_{i=1}^{N} L(y_i, \hat{f}^{-\kappa(i)}(x_i))$$

Bootstrapping: B 가 replication 수 일때
$\hat{f}^{*b}(x_i)$: predicted value of bth fitted bootstrap dataset

$$\widehat{test\ error} = \frac{1}{B}\frac{1}{N}\sum_{b=1}^{B}\sum_{i=1}^{N} L(y_i, \hat{f}^{*b}(x_i))$$

Bootstrapping



Sample size가 N, replication 수를 B 라고 하자.

- sample에서 N개를 복원 추출 (중복된 값 허용)
- B번 반복
- B개의 sample을 통해 통계량 계산 → bootstrap mean, bootstrap variance, bootstrap sd 등등

Multicollinearity (다중공선성)

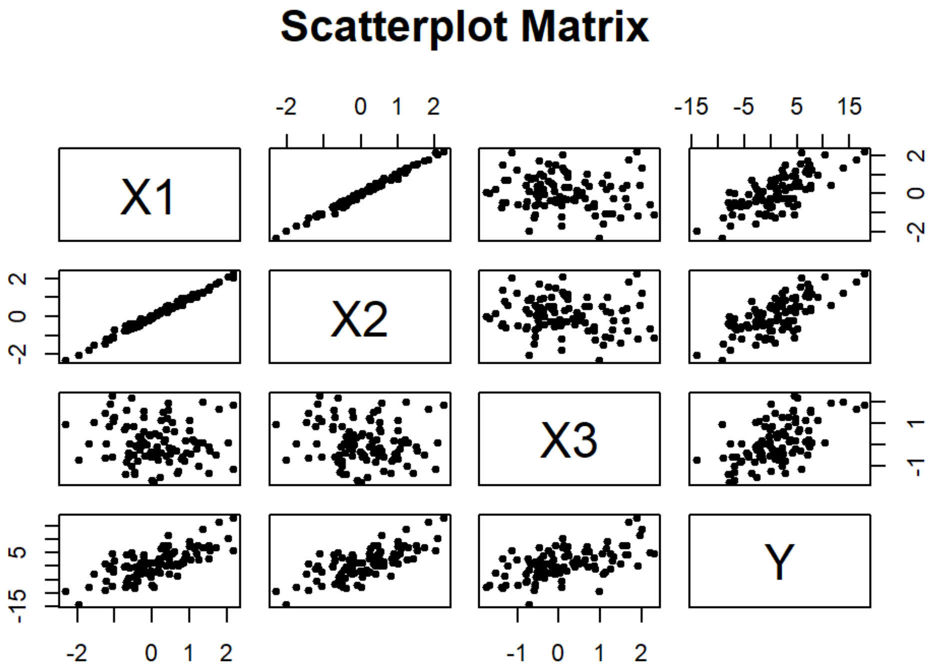
종속변수 Y, 독립변수를 X1, X2라고 하자

- Y와 X1의 correlation이 높으면 (좋다/나쁘다)
- X1과 X2의 correlation이 높으면 (좋다/나쁘다)

Multicollinearity

파악하는 방법

1. Scatterplot matrix: X1과 X2 다중공선성



2. VIF (variance inflation factor)

$$VIF_j = \frac{1}{1 - R_j^2}$$

R_j^2: R^2 calculated by jth predictor regressed on other predictors
→ 보통 cutoff 3 또는 5로 잡는다

→ R^2에 대해서는 뒤에 등장

```
> print(vif_values)
       X1        X2        X3
89.523514 89.447291  1.017576
```

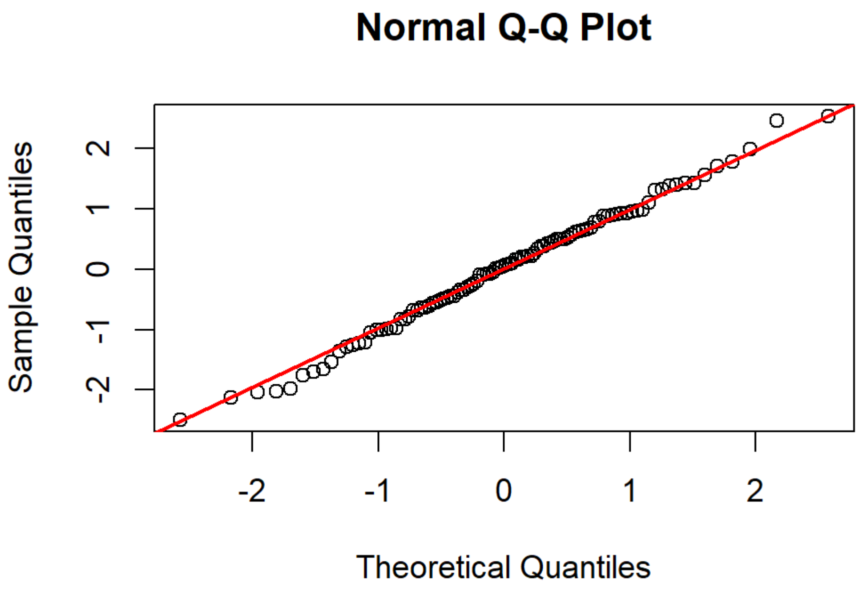
Multicollinearity

처리하는 방법

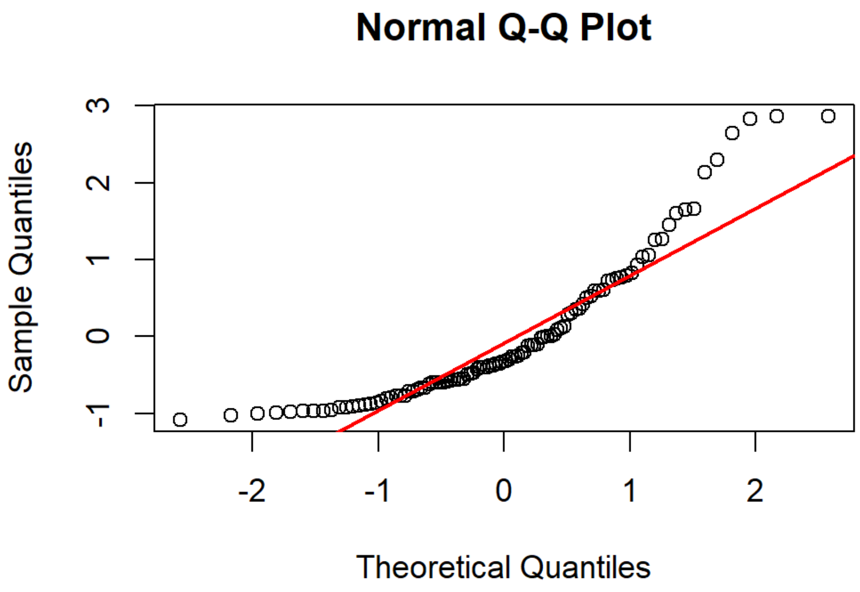
1. **변수 제거 (X2 제거)**
2. Ridge regression, LASSO regression → robust to multicollinearity
3. 파생변수로 만들기

Residual Plot

Residual QQ Plot
오차의 정규성 체크: 잔차의 qqplot이 직선을 따르면 정규성 만족



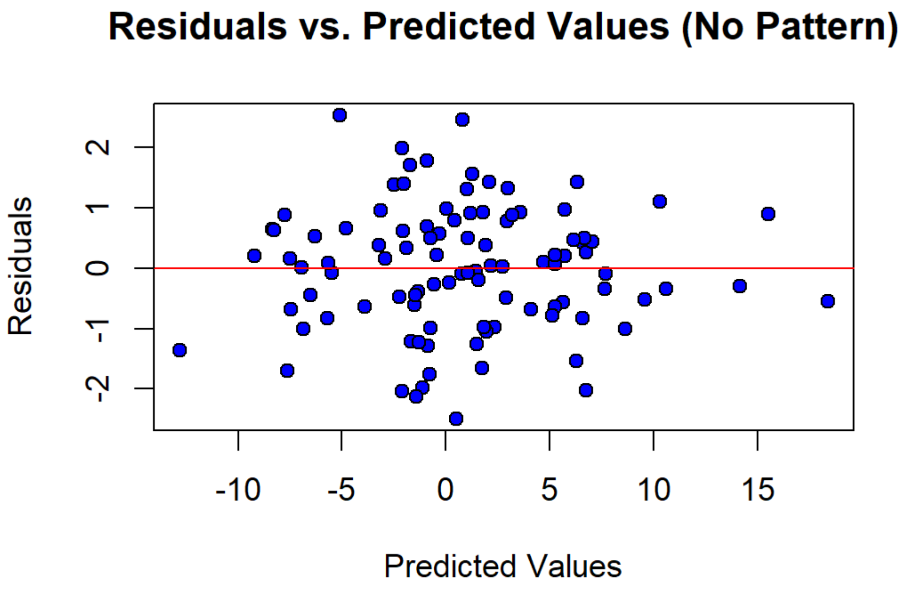
정규 가정 만족



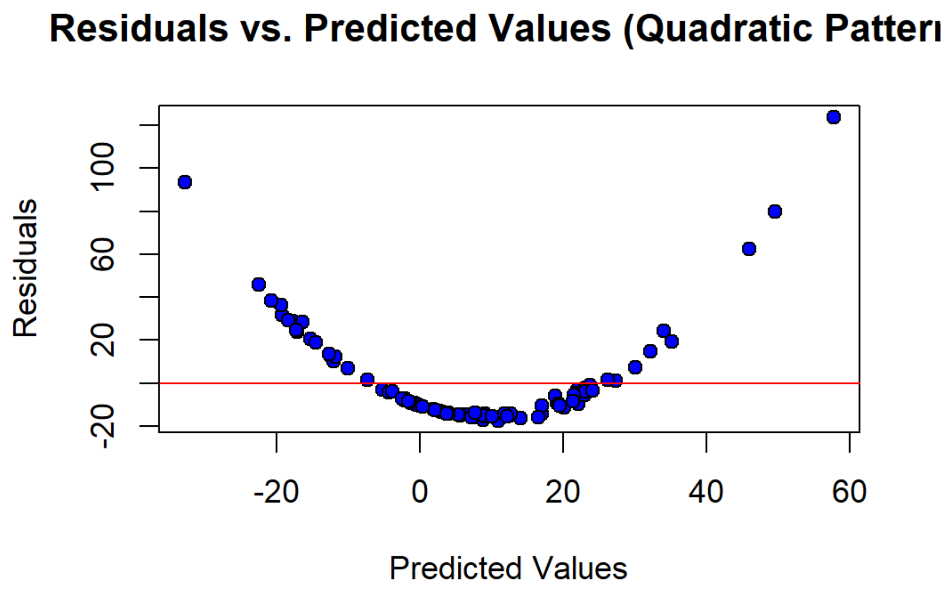
정규 가정 불만족

Residual Plot

Residual vs Predicted Values Plot
오차의 독립성 체크: predicted value vs residual 산점도 패턴 없어야 만족



독립 가정 만족

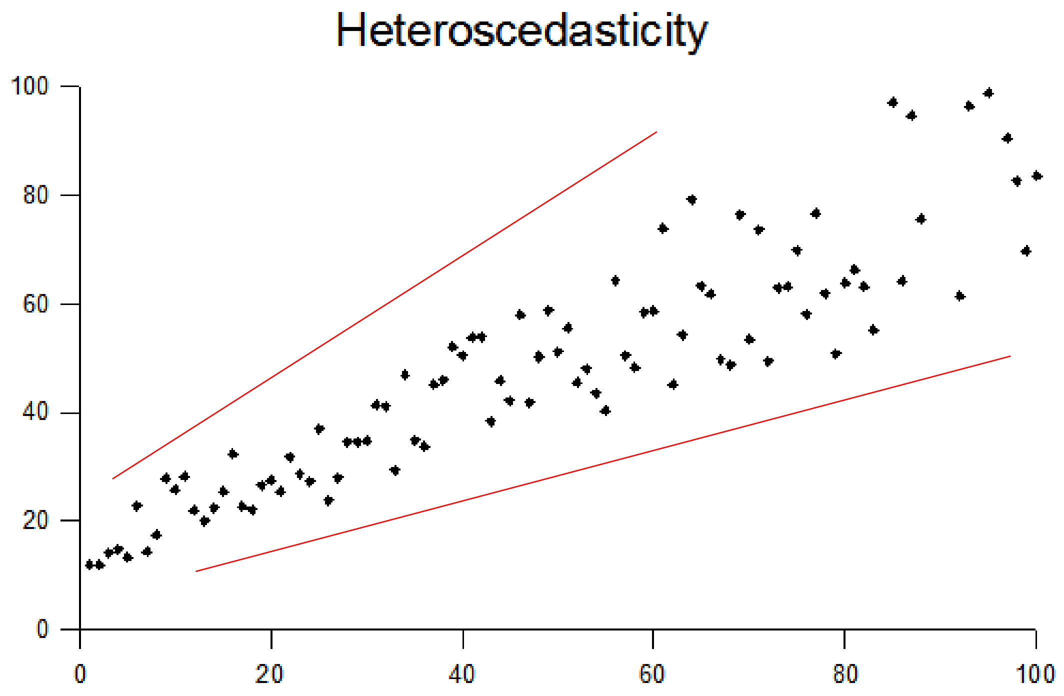


독립 가정 불만족

Residual Plot

Residual vs Predicted Values Plot
오차의 등분산성 체크: predicted value vs residual 퍼져있는 정도가 일정하면 만족

★



점점 퍼지는 중
→ 오차 등분산 불만족

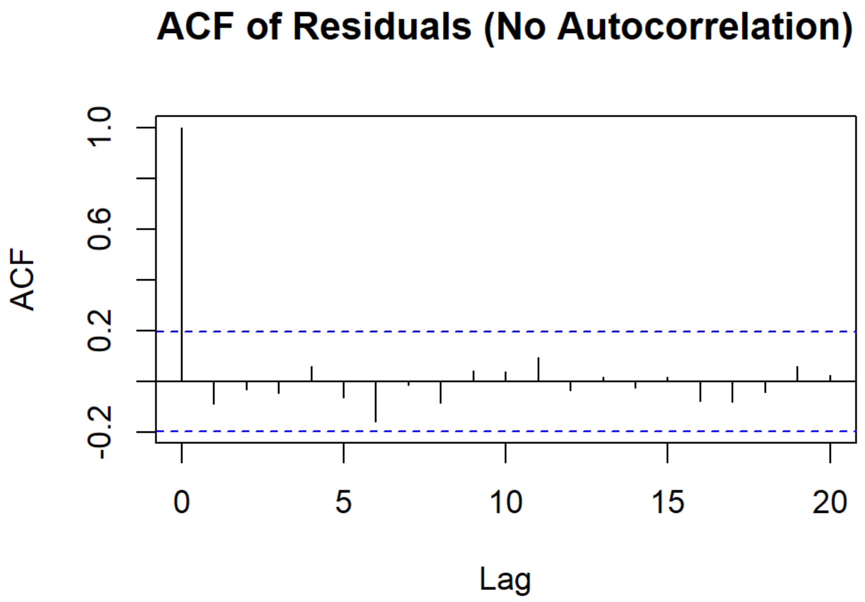
해결 방법:
Variance stabilizing transformation(VST) → 분산을 줄이는 변수변환 이용

X → $\sqrt{X}, \log X, \arcsin X$ 등으로 변환

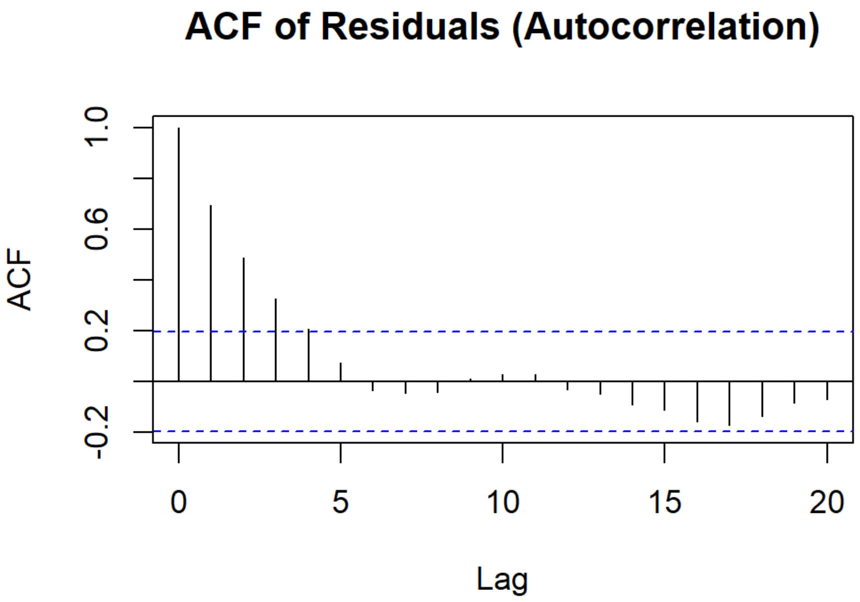
오차의 정규성을 포기한 회귀 모델이 많은데, 등분산은 유지하려 하는 경향은 있다!

Residual Plot

ACF (autocorrelation function) Plot
오차의 독립성 체크: ACF 값이 전체적으로 낮으면 만족. 첫번째 값은 높음



독립 가정 만족



독립 가정 불만족

Goodness of Fit Test

Coefficient of Determination (R^2)

$$R^2 = \frac{SSR}{SST} = \frac{SST - SSE}{SST} = 1 - \frac{SSE}{SST} \leq 1$$

- 전체 변동 (SST) 중 모델이 설명하는 변동 (SSR)의 비율이 얼만큼 되는가.
- 1에 가까워질수록 모델의 설명력이 좋다고 할 수 있다.

수고하셨습니다!

해당 세션자료는 KUBIG Github에서 보실 수 있습니다!

다음은 이번 주차 과제 설명이 있습니다!