Beyond Linear Regression Part II

Intro to Regularization & Model Selection

송 준

고려대학교
통계학과 / 융합데이터과학 대학원

Bias-Variance Tradeoff

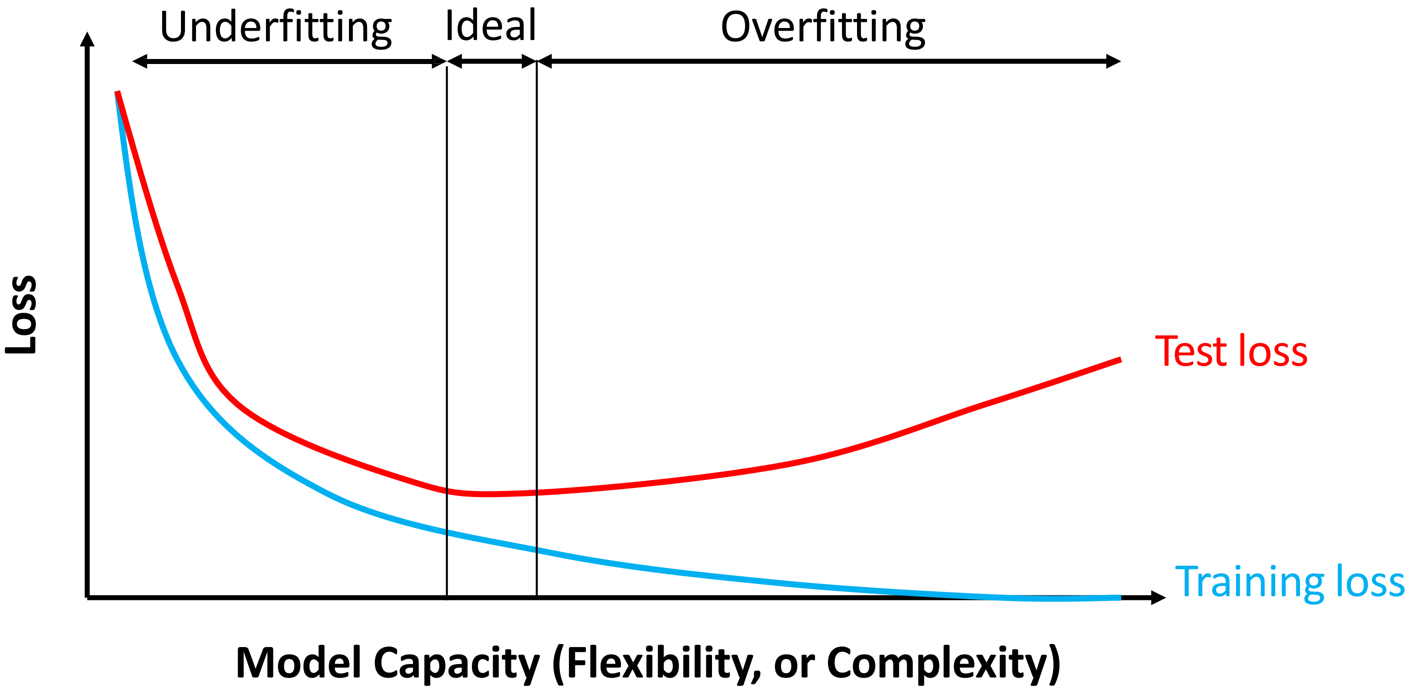
- *For linear regression, increasing feature dimension d…*
    - *Tends to* **increase capacity**
    - *Tends to decrease bias but increase variance*
- *Need to construct ϕ to balance tradeoff between bias and variance*
    - **Rule of thumb**: $n \approx d \log d$
    - *Large fraction of data science work is data cleaning + feature engineering*

Bias-Variance Tradeoff

- *Increasing number of examples n in the data…*
    - *Tends to* **increase bias** *and* **decrease variance**

- **General strategy**
    - **High bias:** *Increase model capacity d*
    - **High variance:** *Increase data size n (i.e., gather more labeled data)*

Bias-Variance Tradeoff



Slide by Padhraic Smyth, UCIrvine

Bias-Variance Tradeoff

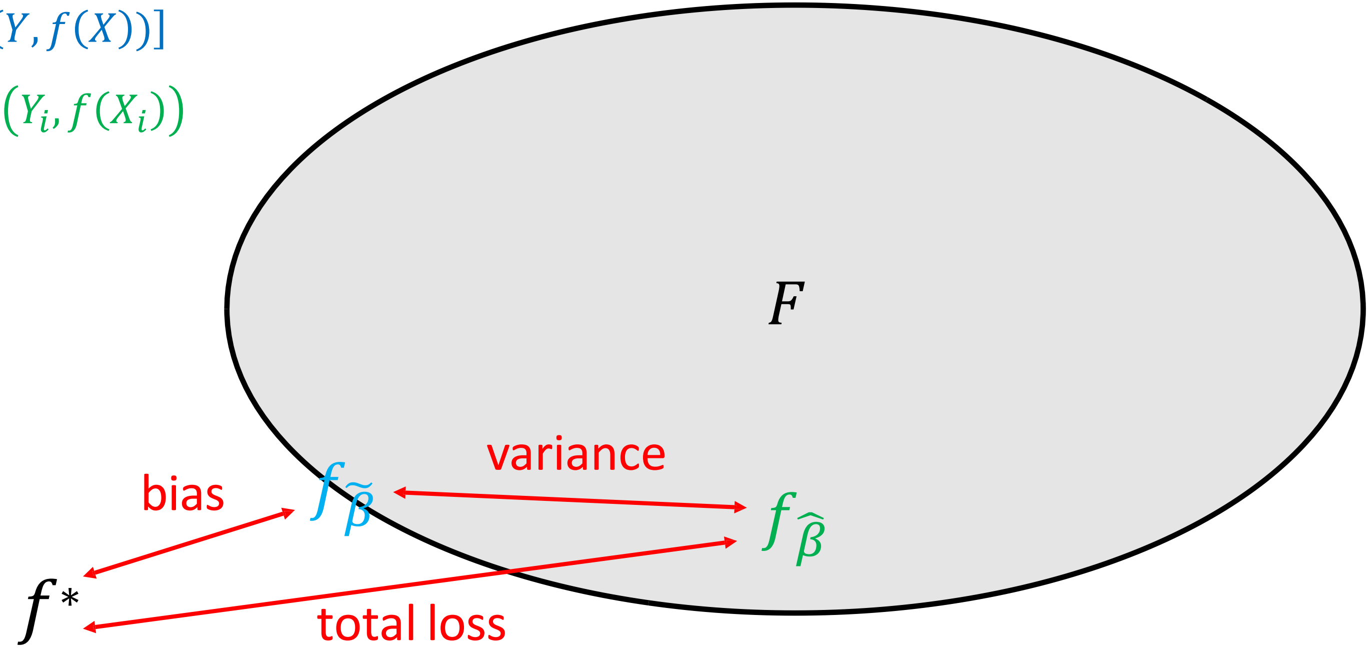
Ideal goal: *Construct prediction rule* $f^*: \mathcal{X} \to \mathcal{Y}$

$$\boldsymbol{f^* = argmin_f \mathbb{E}_{XY}[loss(Y, f(X))]}$$

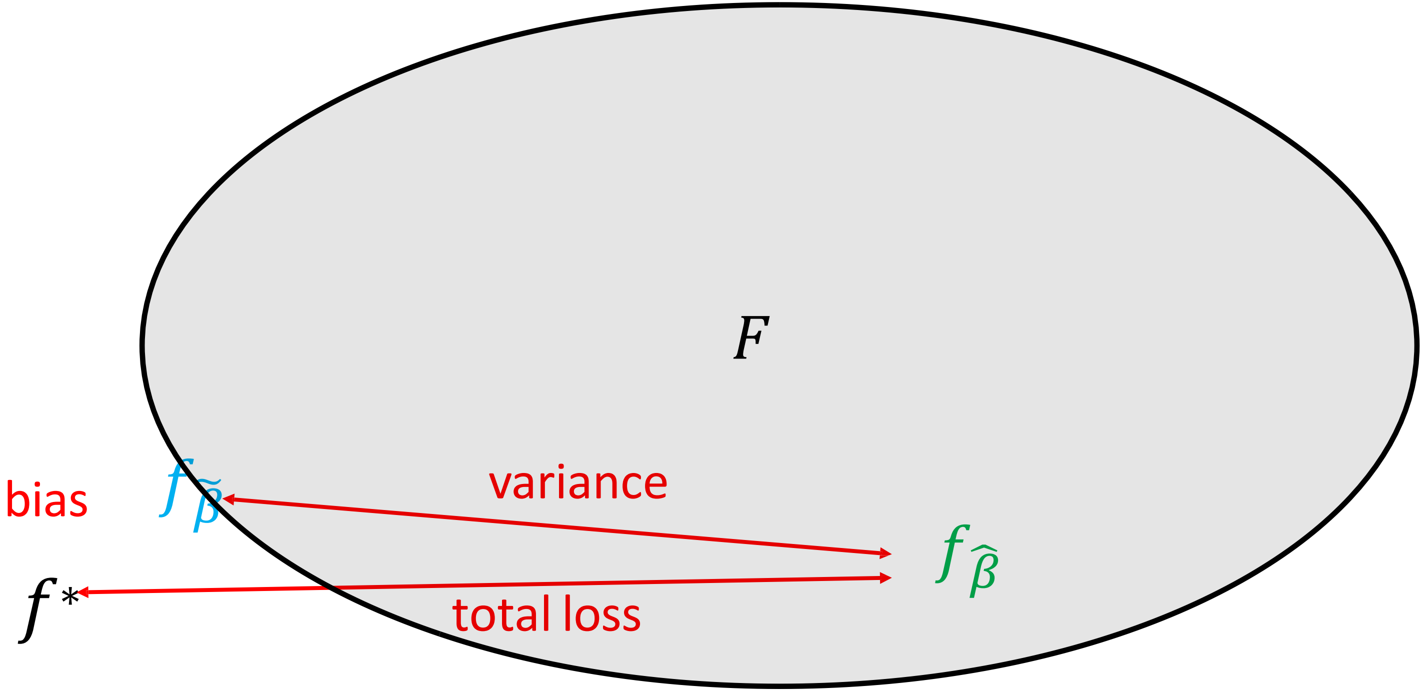
$$\tilde{\boldsymbol{f}} = \arg\min_{f \in F} \mathbb{E}_{XY}[\mathrm{loss}(Y, f(X))]$$

$$\hat{f}_n = \arg\min_{f \in F} \sum_{i=1}^{n} \mathrm{loss}\big(Y_i, f(X_i)\big)$$

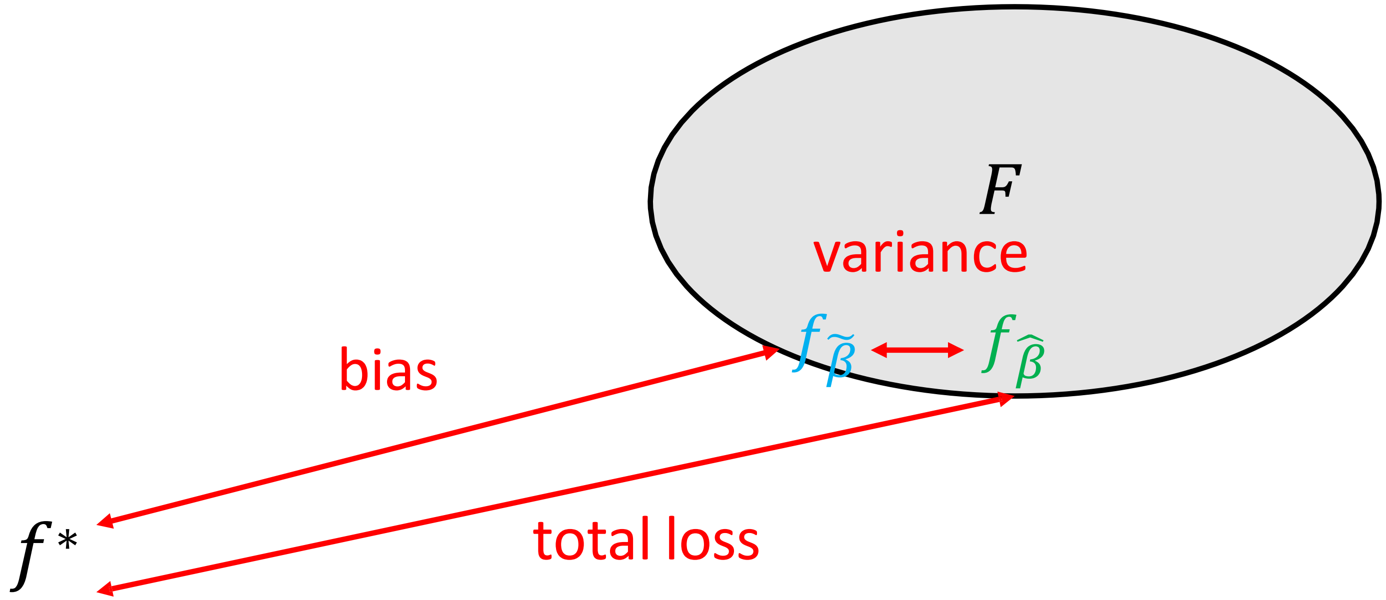
Overfitting?
Underfitting?



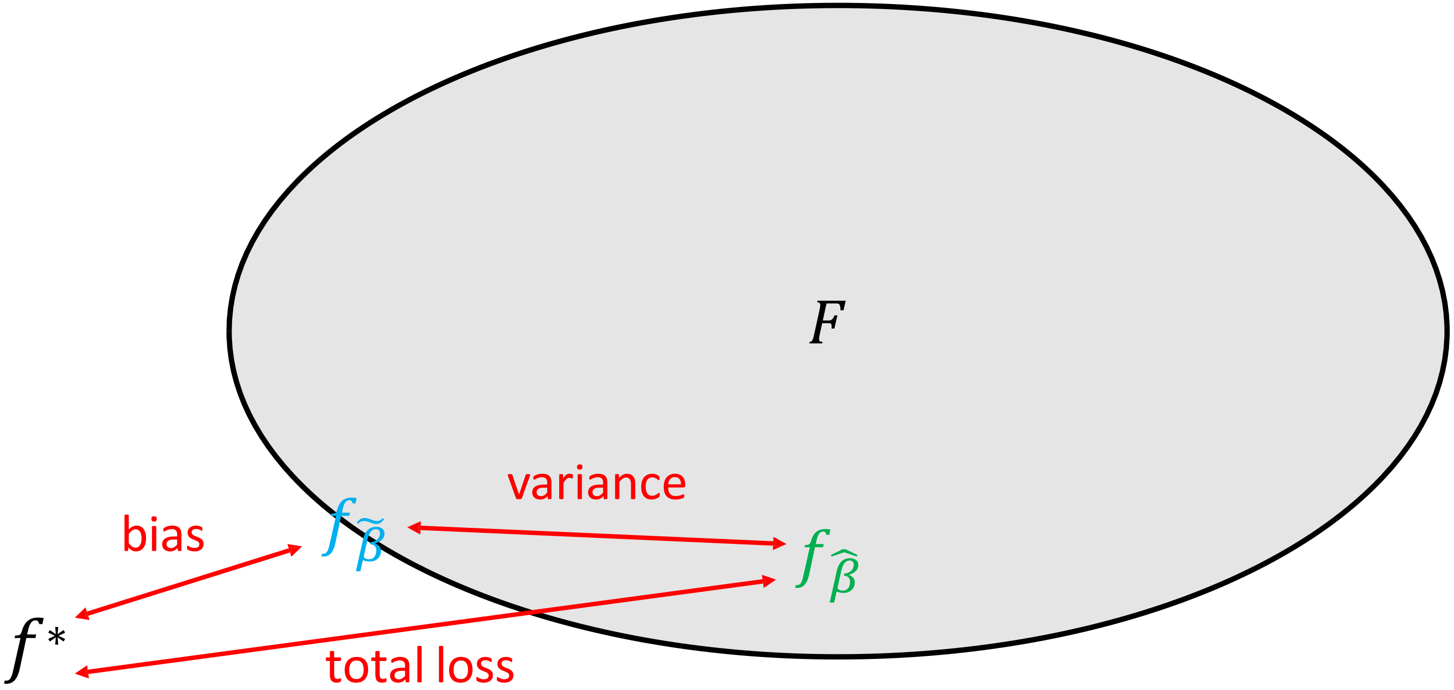
Bias-Variance Tradeoff(Overfitting)



Bias-Variance Tradeoff(Underfitting)



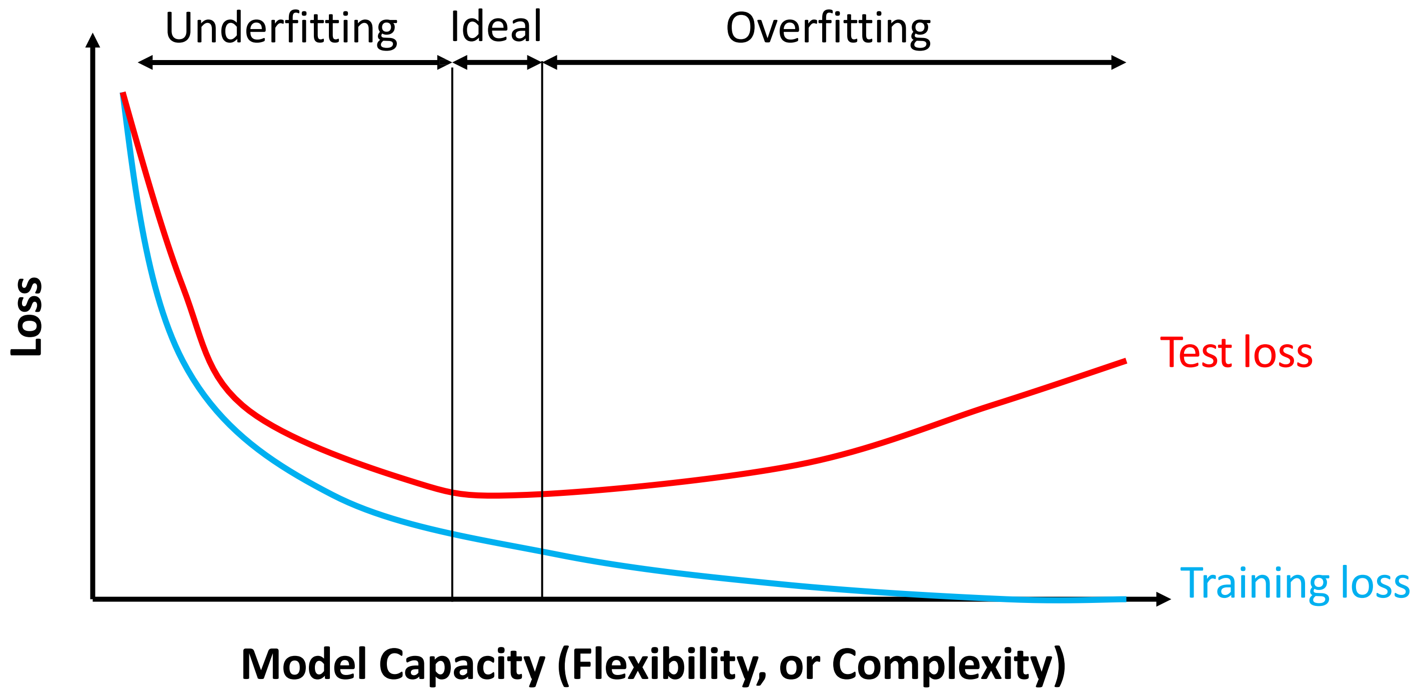
Bias-Variance Tradeoff(Ideal)



Intro to Feature Selection

Intro to Feature Selection

- *To avoid overfitting, Select some features to reduce the model capacity*

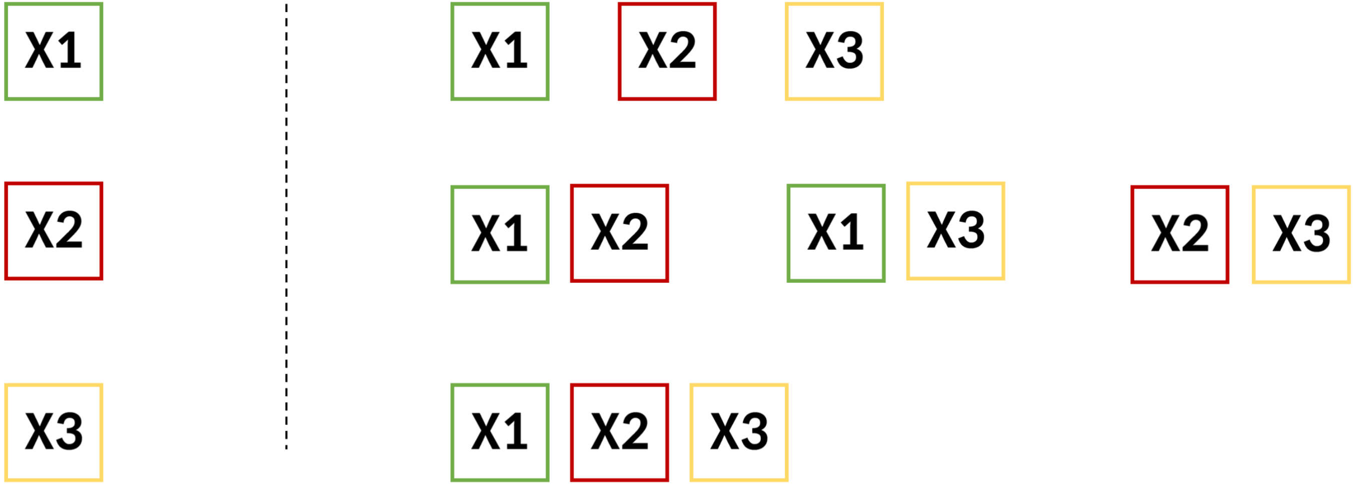


Exhaustive Search

- *Best combination of features*
- 경우의 수: $2^p - 1$ - 시간이 너무 많이 걸림
- 예제: *p=3*

3 Variables

총 7개 가능 Subsets 존재



Sequential Selection

- *Forward Selection (Addition)*
    - *0 variable*부터 시작. 하나씩 추가
    - 한번 추가된 변수는 다시 지우지 않음

- *Backward Selection (Elimination)*
    - *Full model*에서 시작. 하나씩 제거
    - 한번 제거된 변수는 다시 추가하지 않음

- *Step-wise Selection*
    - 위 기법 혼합

Forward Selection

- *Forward Selection (Addition)*
  - *0 variable*부터 시작. 하나씩 추가. **적절한 Performance Measure 활용**
  - 한번 추가된 변수는 다시 지우지 않음
- 예제: *8 variables. 1st step (Find the best 1-variable model)*

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 \, , R^2_{adj} = 0.48$$

$$\boxed{\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 \, , R^2_{adj} = 0.56}$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_3 x_3 \, , R^2_{adj} = 0.51$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_4 x_4 \, , R^2_{adj} = 0.50$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_5 x_5 \, , R^2_{adj} = 0.38$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_6 x_6 \, , R^2_{adj} = 0.32$$

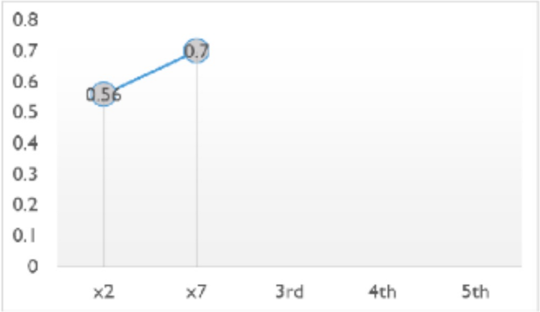
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_7 x_7 \, , R^2_{adj} = 0.50$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_8 x_8 \, , R^2_{adj} = 0.19$$

Forward Selection

- *Forward Selection (Addition)*
  - *0 variable*부터 시작. 하나씩 추가. **적절한 Performance Measure 활용**
  - 한번 추가된 변수는 다시 지우지 않음
- 예제: *8 variables. 2nd step (Find the best 2-variable model, including x2)*

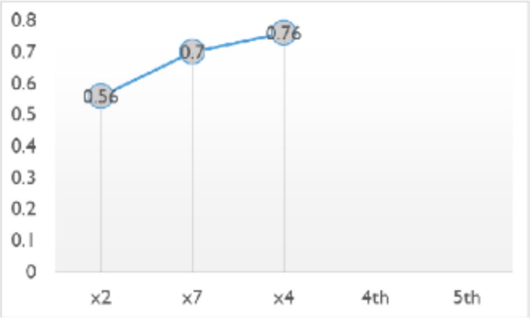
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_1 x_1 , R^2_{adj} = 0.60$$
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_3 , R^2_{adj} = 0.64$$
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_4 x_4 , R^2_{adj} = 0.58$$
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_5 x_5 , R^2_{adj} = 0.61$$
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_6 x_6 , R^2_{adj} = 0.57$$
$$\boxed{\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7 , R^2_{adj} = 0.70}$$
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_8 x_8 , R^2_{adj} = 0.56$$



Forward Selection

- *Forward Selection (Addition)*
  - *0 variable*부터 시작. 하나씩 추가. **적절한 Performance Measure 활용**
  - 한번 추가된 변수는 다시 지우지 않음
- 예제: *8 variables. 3rd step (Find the best 3-variable model, including x2,x7)*

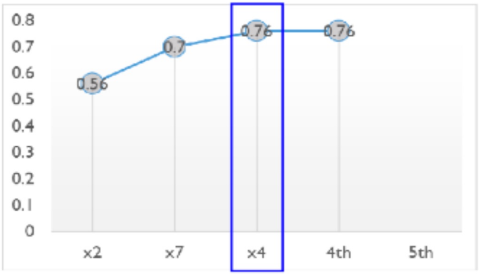
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7 + \hat{\beta}_1 x_1 \,, R^2_{adj} = 0.71$$
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7 + \hat{\beta}_3 x_3 \,, R^2_{adj} = 0.72$$
$$\boxed{\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7 + \hat{\beta}_4 x_4 \,, R^2_{adj} = 0.76}$$
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7 + \hat{\beta}_5 x_5 \,, R^2_{adj} = 0.73$$
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7 + \hat{\beta}_6 x_6 \,, R^2_{adj} = 0.69$$
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7 + \hat{\beta}_8 x_8 \,, R^2_{adj} = 0.70$$



Forward Selection

- *Forward Selection (Addition)*
  - *0 variable*부터 시작. 하나씩 추가. **적절한 Performance Measure 활용**
  - 한번 추가된 변수는 다시 지우지 않음
- 예제: *8 variables. Stop the procedure when a certain criteria is satisfied.*

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7 + \hat{\beta}_4 x_4 + \hat{\beta}_1 x_1 \,, R^2_{adj} = 0.76$$
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7 + \hat{\beta}_4 x_4 + \hat{\beta}_3 x_3 \,, R^2_{adj} = 0.76$$
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7 + \hat{\beta}_4 x_4 + \hat{\beta}_5 x_5 \,, R^2_{adj} = 0.75$$
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7 + \hat{\beta}_4 x_4 + \hat{\beta}_6 x_6 \,, R^2_{adj} = 0.76$$
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7 + \hat{\beta}_4 x_4 + \hat{\beta}_8 x_8 \,, R^2_{adj} = 0.75$$



정확도의 변화가 없으면 STOP

- Final model $\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7 + \hat{\beta}_4 x_4 \,, R^2_{adj} = 0.76$

Backward Selection

- *Backward Selection (Elimination)*
  - *Full model*에서 시작. 하나씩 제거. **적절한 Performance Measure 활용**
  - 한번 제거된 변수는 다시 추가하지 않음
- 예제: *8 variables. 1st step: fit the full model.*

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_3 + \hat{\beta}_4 x_4 + \hat{\beta}_5 x_5 + \hat{\beta}_6 x_6 + \hat{\beta}_7 x_7 + \hat{\beta}_8 x_8, \qquad R^2_{adj} = 0.77$$

8개 Variables 다 넣고 시작

Backward Selection

- *Backward Selection (Elimination)*
  - *Full model*에서 시작. 하나씩 제거. **적절한 Performance Measure 활용**
  - 한번 제거된 변수는 다시 추가하지 않음
- 예제: *8 variables. next step: Find the best 7-variable model – do this until …*

$$\hat{y} = \hat{\beta}_0 + \qquad + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_3 + \hat{\beta}_4 x_4 + \hat{\beta}_5 x_5 + \hat{\beta}_6 x_6 + \hat{\beta}_7 x_7 + \hat{\beta}_8 x_8,\ R^2_{adj} = 0.65$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \qquad + \hat{\beta}_3 x_3 + \hat{\beta}_4 x_4 + \hat{\beta}_5 x_5 + \hat{\beta}_6 x_6 + \hat{\beta}_7 x_7 + \hat{\beta}_8 x_8,\ R^2_{adj} = 0.60$$

$$\boxed{\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 +. \qquad + \hat{\beta}_4 x_4 + \hat{\beta}_5 x_5 + \hat{\beta}_6 x_6 + \hat{\beta}_7 x_7 + \hat{\beta}_8 x_8,\ R^2_{adj} = 0.77}$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_3 +. \qquad + \hat{\beta}_5 x_5 + \hat{\beta}_6 x_6 + \hat{\beta}_7 x_7 + \hat{\beta}_8 x_8,\ R^2_{adj} = 0.62$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_3 + \hat{\beta}_4 x_4 + \qquad + \hat{\beta}_6 x_6 + \hat{\beta}_7 x_7 + \hat{\beta}_8 x_8,\ R^2_{adj} = 0.73$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_3 + \hat{\beta}_4 x_4 + \hat{\beta}_5 x_5 + \qquad + \hat{\beta}_7 x_7 + \hat{\beta}_8 x_8,\ R^2_{adj} = 0.71$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_3 + \hat{\beta}_4 x_4 + \hat{\beta}_5 x_5 + \hat{\beta}_6 x_6 + \qquad + \hat{\beta}_8 x_8,\ R^2_{adj} = 0.61$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_3 + \hat{\beta}_4 x_4 + \hat{\beta}_5 x_5 + \hat{\beta}_6 x_6 + \hat{\beta}_7 x_7 + \qquad ,\ R^2_{adj} = 0.74$$

Stepwise Selection

- *Step-wise Selection*
  - 위 기법 혼합 *(Forward/Backward* 번갈아 수행)
  - 시간은 더 오래 걸릴 수 있지만 보다 좋은 모델 선택 가능성이 있음.

$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2,\qquad R^2_{adj} = 0.56.$

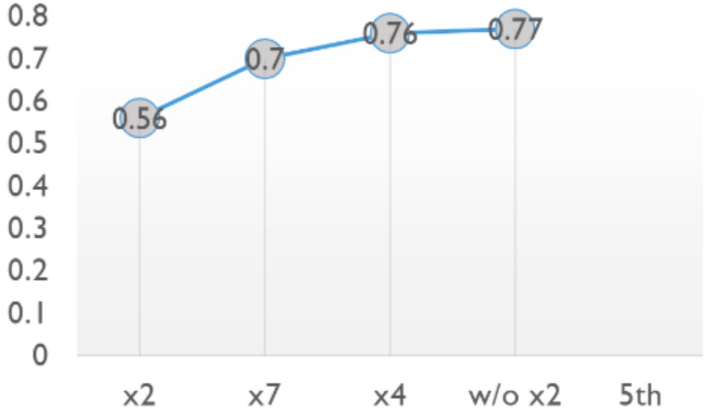
$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7\,,\qquad R^2_{adj} = 0.70$

$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7 + \hat{\beta}_4 x_4, R^2_{adj} = 0.76$

$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_4 x_4\,,\qquad R^2_{adj} = 0.58$

$\hat{y} = \hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_7 x_7\,,\qquad R^2_{adj} = 0.70$

$\boxed{\hat{y} = \hat{\beta}_0 + \hat{\beta}_4 x_4 + \hat{\beta}_7 x_7\,,\qquad R^2_{adj} = 0.77}$



Regularization

Recall (다중공선성, Multicollinearity)

Suppose that there are $c_0, c_1, \ldots, c_n \in \mathbb{R}$, *such that*

$$X_j = c_0 1_n + c_1 X_1 + \cdots + c_{j-1} X_{j-1} + c_{j+1} X_{j+1} + \cdots + c_p X_p + \delta,$$

If δ = 0,

- *X* 변수들이 서로 상관관계가 *1*이면
  - $(\boldsymbol{X}^T \boldsymbol{X})\hat{\beta} = \boldsymbol{X}^T \boldsymbol{Y}$: 해가 존재하지 않음.

- *X* 변수들이 서로 상관관계가 매우 높다면
  - $\hat{\beta} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{Y}$ <- 이 값이 매우 불안정함 (**분산이 매우 높음**!)
  - 계산이 가능하더라도 결과값에 대한 신뢰도는 매우 낮게 됨

- 통계모델 학습 자체는 가능. 하지만 결과값이 *j*번째 변수에 의한 것인지 아니면 다른 변수에 의한 것인지 판단이 어려움.

High Variance in Linear Regression

- *Multicollinearity*
  - $\hat{\beta} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}$ <- 이 값이 매우 불안정함 (**분산이 매우 높음**!)
  - 계산이 가능하더라도 결과값에 대한 신뢰도는 매우 낮게 됨

- *High-dimensional data (n < p)*
  - $\hat{\beta} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}$ <- 이 값이 매우 불안정함 (**분산이 매우 높음**!)

Regularization

- **Regularization**
  - *Strategy to address bias-variance tradeoff*
  - **Start with** *Linear regression with L_2 regularization*

Regularization

- **Regularization**
  - *Strategy to address bias-variance tradeoff*
  - **Start with** *Linear regression with L_2 regularization*

- *Why Linear?*
  - *Simple:* **inferences**, *its* **interpretability** *and often shows good predictive performance.*

- *Improve the linear model, by replacing the least square fitting with some alternative fitting procedure.*

Recall : Mean Squared Error Loss

- *Mean squared error loss for linear regression:*

$$L(\beta; Z) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta^T x_i)^2$$

Linear Regression with L_p Regularization

- **Original loss** + **regularization**:

$$L(\beta; Z) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta^T x_i)^2 + \lambda \cdot \|\beta\|_p$$

- λ *is a* **hyperparameter** *that must be tuned (satisfies* $\lambda \geq 0$*)*

L_p Norm?

- *Norm: Informally, a norm of a vector represents* **how large** *the vector is* (원점으로부터의 거리)

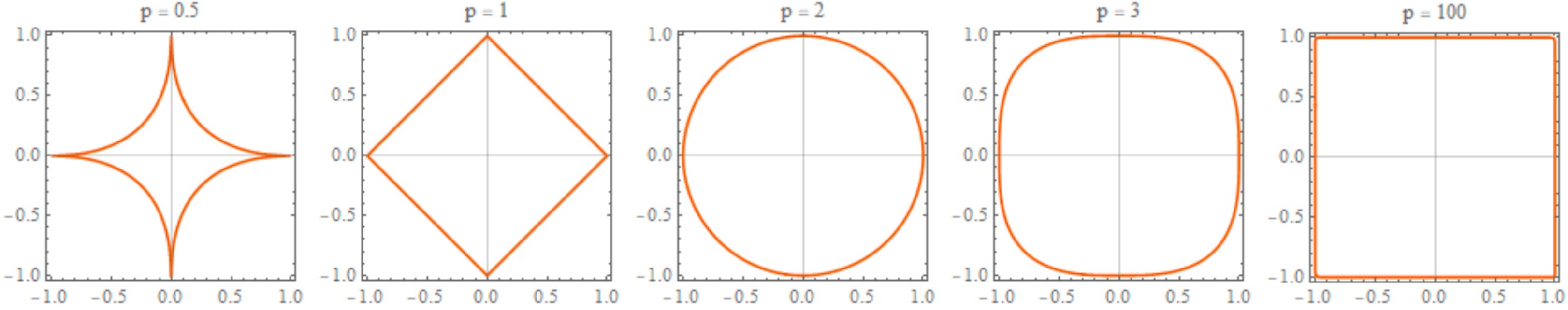
The Vector B -> A
How Large?

$$||x||_2 = \sqrt{x_1^2 + \cdots + x_n^2}$$

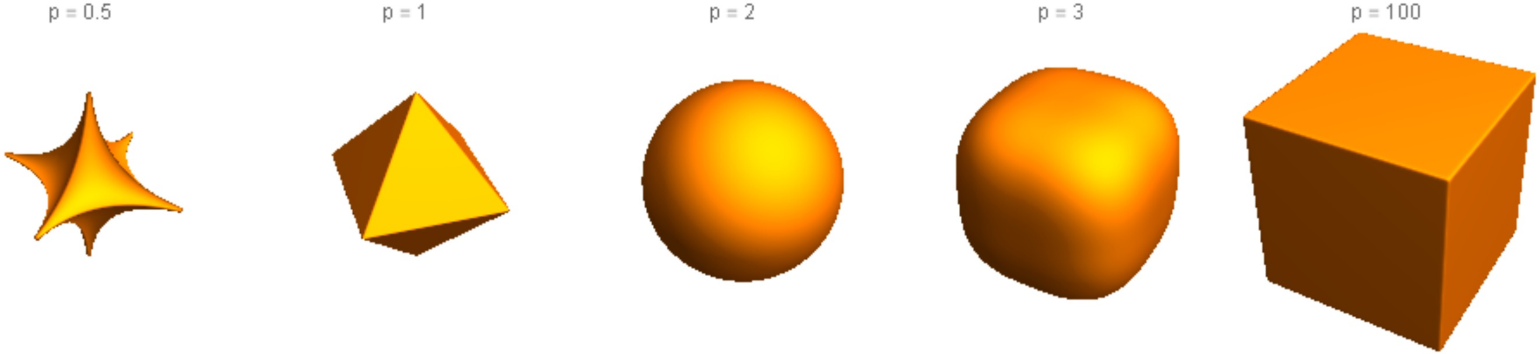
$$||x||_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}}$$

L_p Norm?

When $x \in \mathbb{R}^2$, $\{x \in \mathbb{R}^2 : \|x\|_p = 1\}$ is



When $x \in \mathbb{R}^3$, $\{x \in \mathbb{R}^2 : \|x\|_p = 1\}$ is



Linear Regression with L_p Regularization

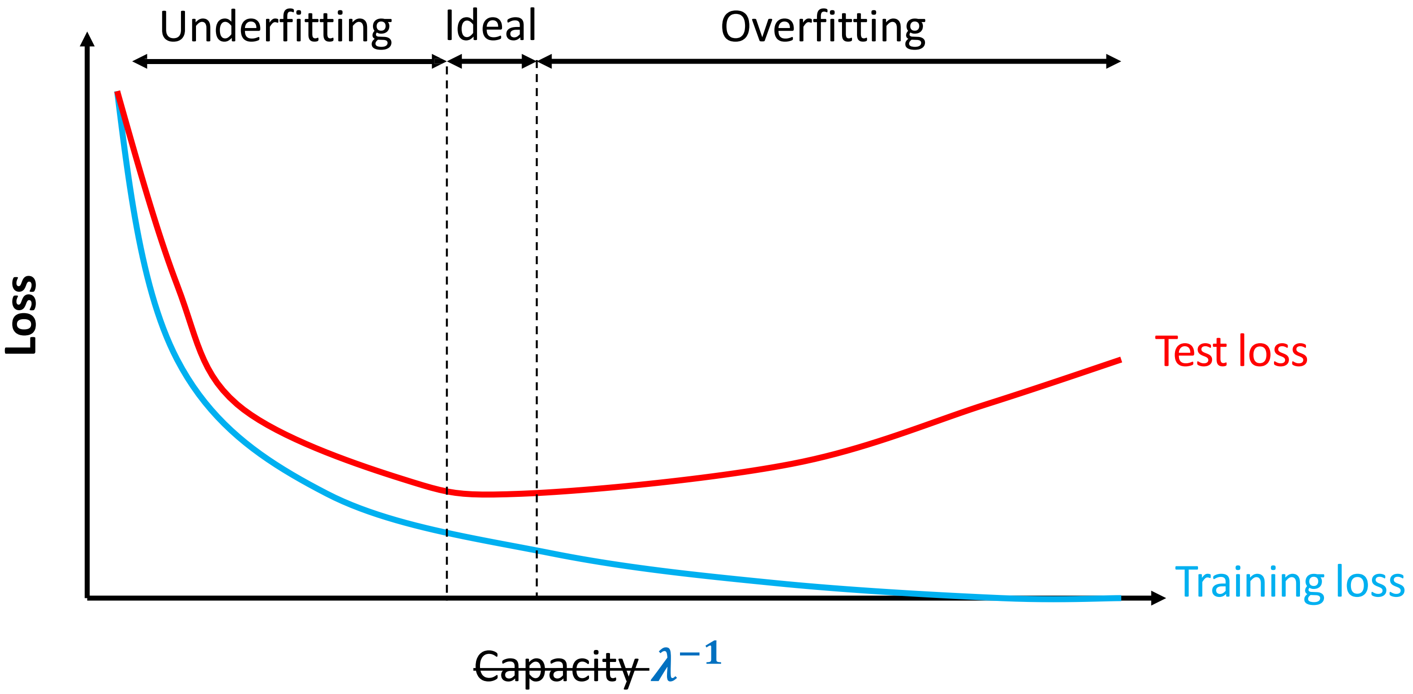
- **Original loss** + **regularization**:

$$L(\beta; Z) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta^T x_i)^2 + \lambda \cdot \|\beta\|_p$$

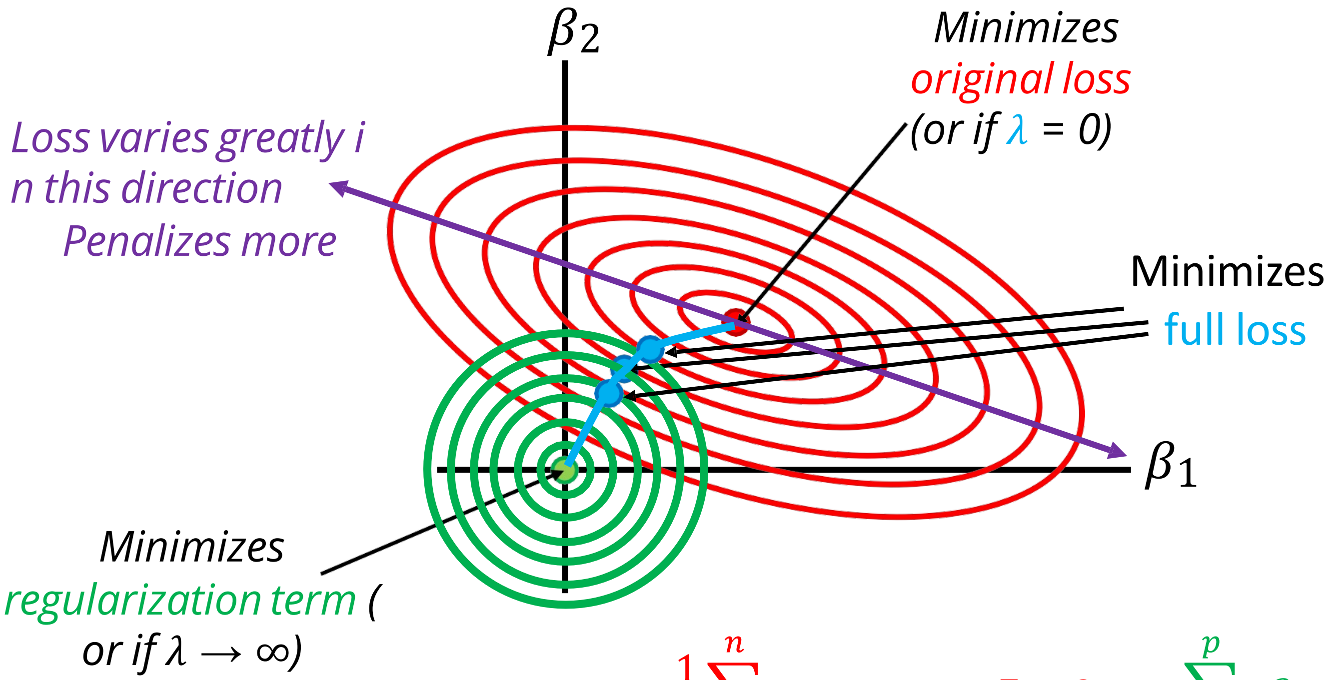
- λ *is a* **hyperparameter** *that must be tuned (satisfies* $\lambda \geq 0$*)*

- *Penalty term: we want to reduce the loss. If* λ *is large, more penalty on* $\|\beta\|_p$
  - *A large* λ *encourages "simple" function.*
  - *Tuning* λ = *Tuning* **bias-variance tradeoff**

Bias-Variance Tradeoff for Regularization



Intuition L_2 Regularization



- At this point, the gradients are **equal**
- (with opposite sign)
- Tradeoff depends on ch oice of λ

$$L(\beta; Z) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta^T x_i)^2 + \lambda \sum_{j=1}^{p}\beta_j^2$$

L_2 Regularization: Optimization

Recall the Lagrangian multiplier method.

- *Minimize*

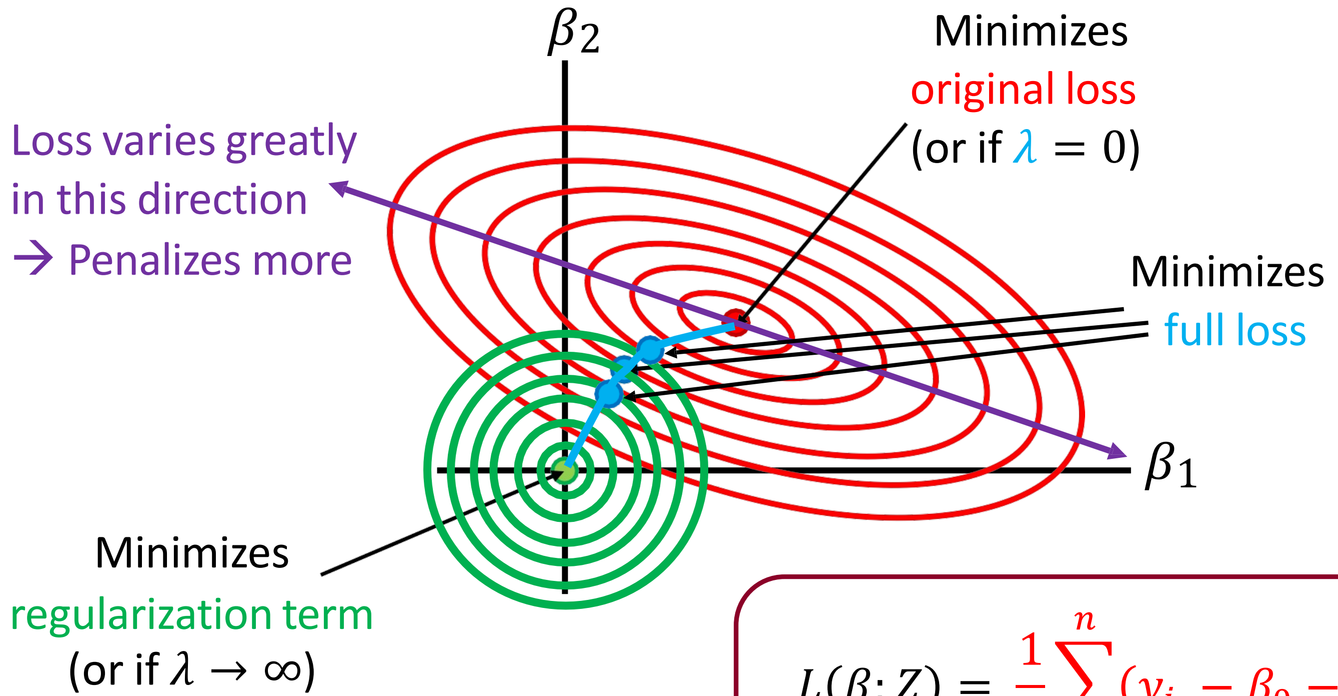
$$L(\beta; Z) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta^T x_i)^2 + \lambda \cdot \|\beta\|_p$$

- *Minimize*

$$L(\beta; Z) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta^T x_i)^2$$

subject to $\|\beta\|_p \leq c$

Intuition L_2 Regularization



- At this point, the gradients are **equal**
- (with opposite sign)
- Tradeoff depends on choice of λ

$$L(\beta; Z) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta^T x_i)^2$$

subject to $\|\beta\|_p \leq c$

Ridge Regression

Ridge Regression *is the linear regression with L2 penalty*

- *Minimize*

$$\hat{\beta}^{Ridge} = \arg\min_{\beta \in \mathbb{R}^p} L(\beta; Z) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta^T x_i)^2 + \lambda \cdot \|\beta\|_p$$

- *The objective function has a closed-form solution (analytic solution) as below*

$$\hat{\beta}^{Ridge} = \boxed{\left(X^T X + \lambda I_p\right)^{-1}} X^T Y$$

inverse is stable

- *Remark: if the predictors are orthonormal, (variables are not correlated), it has a form of*

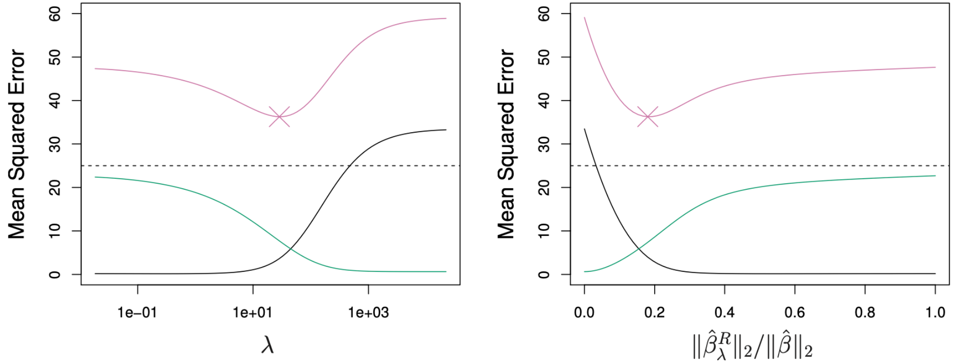
$$\hat{\beta}^{Ridge} = \boxed{\frac{\hat{\beta}}{1+\lambda}}$$

coefficients are shrunken

Why does Ridge Regression Improve over LSE?

The Bias-Variance Tradeoff:

Squred bias(black), Variance(green), and Test MSE(purple).



We can find λ>0 such that

$$MSE_{test}\left(\hat{\beta}_\lambda^{Ridge}\right) < MSE_{test}\left(\hat{\beta}^{OLS}\right)$$

Feature Standardization

- **Unregularized linear regression is invariant to feature scaling**
  - *Suppose we scale $x_{ij} \leftarrow 2x_{ij}$ for all examples x_i*
  - *Without regularization, simply use $\beta_j \leftarrow \beta_j/2$ to obtain equivalent solution*
  - In particular $\frac{\beta_j}{2} \cdot 2x_{ij} = \beta_j \cdot x_{ij}$
- *Not true for regularized regression!*
  - *Penalty $(\beta_j/2)^2$ is scaled by 1/4 (not cancelled out!)*

$$L(\beta; Z) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta^T x_i)^2 + \lambda \sum_{j=2}^{d} \beta_j^2$$

Feature Standardization

- **Unregularized linear regression is invariant to feature scaling**
  - *Suppose we scale* $x_{ij} \leftarrow 2x_{ij}$ *for all examples* x_i
  - *Without regularization, simply use* $\beta_j \leftarrow \beta_j/2$ *to obtain equivalent solution*
  - *In particular* $\sum_{j=1}^{d} \frac{\beta_j}{2} \cdot 2x_{ij} = \sum_{j=1}^{d} \beta_j \cdot x_{ij}$

- *Not true for regularized regression!*
  - *Penalty* $\left(\beta_j/2\right)^2$ *is scaled by 1/4 (not cancelled out!)*

$$L(\beta; Z) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \beta^T x_i)^2 + \lambda\left(\beta_2^2 + \cdots + \beta_j^2 + \cdots + \beta_d^2\right)$$

Feature Standardization

- **Solution:** *Rescale features to zero mean and unit variance*

$$x_{i,j} \leftarrow \frac{x_{i,j} - \mu_j}{\sigma_j} \qquad \mu_j = \frac{1}{N}\sum_{i=1}^{N} x_{i,j} \qquad \sigma_j = \frac{1}{N}\sum_{i=1}^{N}\left(x_{i,j} - \mu_j\right)^2$$

  - **Note:** *When using intercept term, do not rescale $x_1 = 1$*

- **Must use same transformation during training and for prediction**
  - *Compute on standardization on training data and use on test data*

General Regularization Strategy

- **Original loss** + **regularization**:

$$L_{new}(\beta; Z) \qquad = L(\beta; Z) \qquad + \lambda \cdot R(\beta)$$

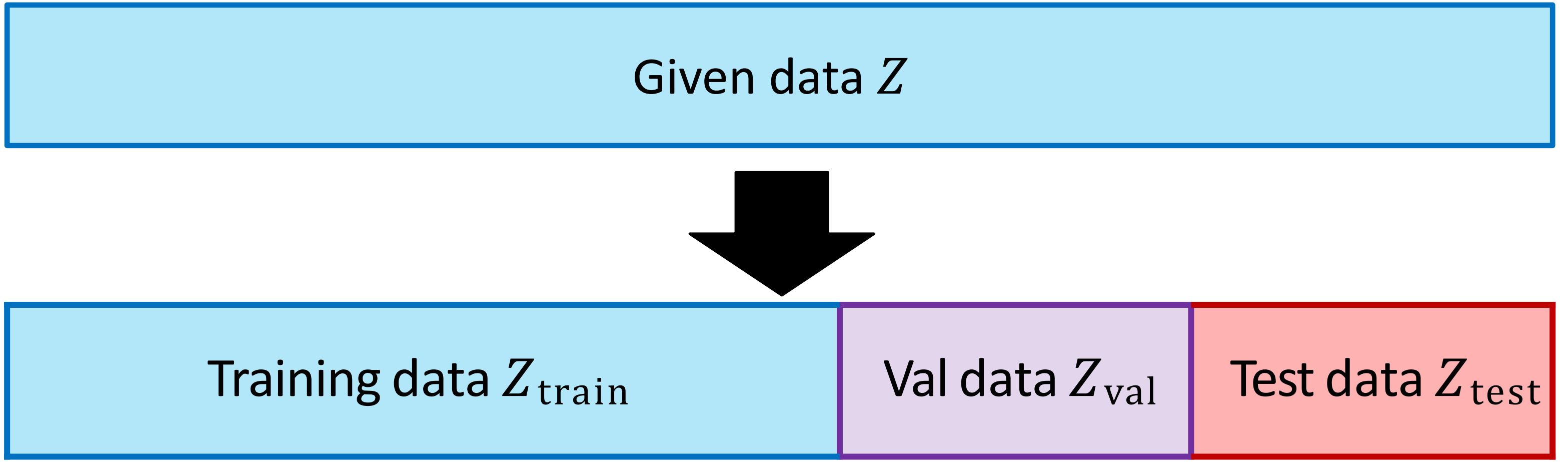
  - *Offers a way to express a preference "simpler" functions in family*
  - *Typically, regularization is independent of data*

Hyperparameter Tuning

- λ *is a* **hyperparameter** *that must be tuned (satisfies* $\lambda \geq 0$*)*
- **Naïve strategy:** *Try a few different candidates* λ_t *and choose the one that minimizes the test loss*
- **Problem:** *We may overfit the test set!*
  - *Major problem if we have more hyperparameters*

Training/Val/Test Split

- **Goal:** *Choose best hyperparameter* λ
  - *Can also compare different model families, feature maps, etc.*
- **Solution:** *Optimize* λ *on a* **held-out validation data**
  - **Rule of thumb:** *60/20/20 split*



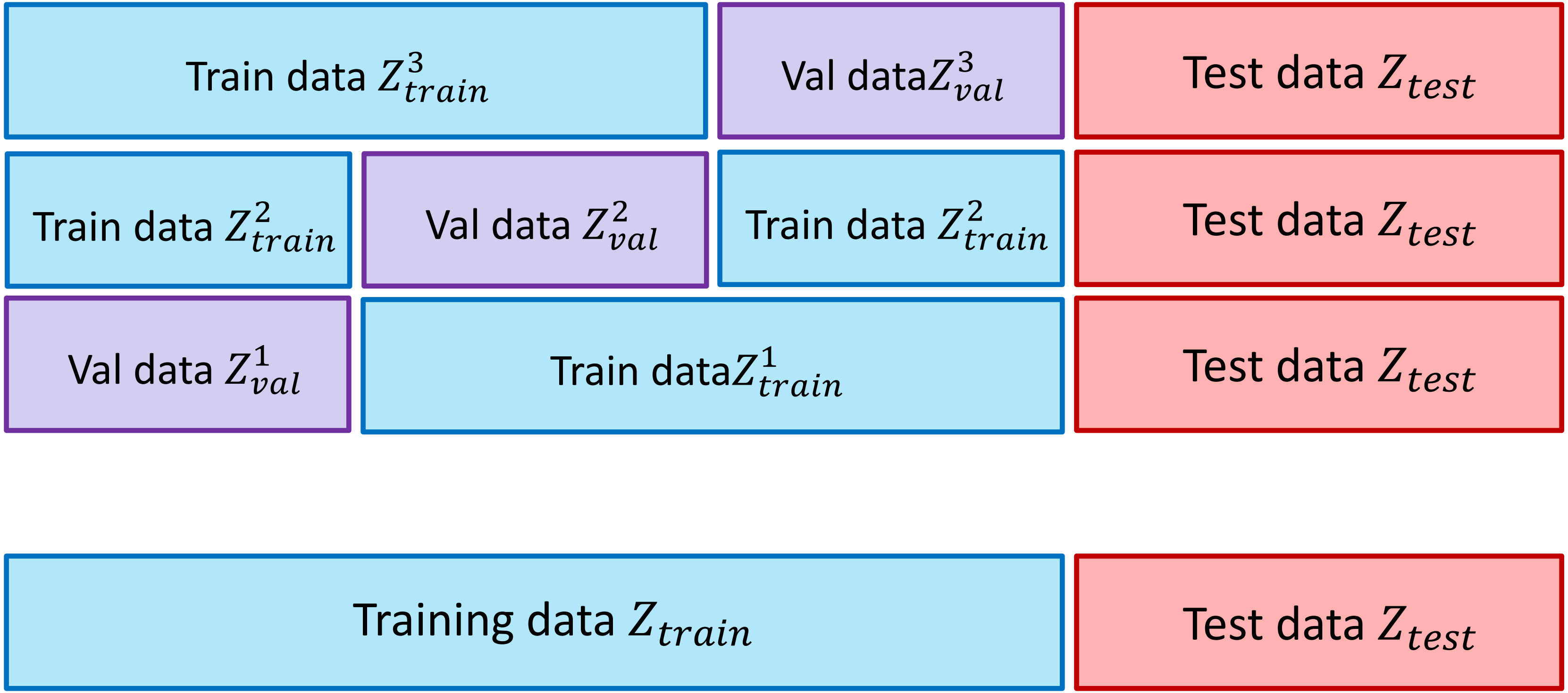
Basic Cross Validation Algorithm

- **Step 1:** *Split Z into Z_{train}, Z_{val}, and Z_{test}*

Training data Z_{train}	Val data Z_{val}	Test data Z_{test}

- **Step 2:** *For $t \in \{1, \ldots, h\}$:*
  - **Step 2a:** *Run linear regression with Z_{train} and λ_t to obtain $\hat{\beta}(Z_{train}, \lambda_t)$*
  - **Step 2b:** Evaluate validation loss $L_{val}^{t} = L\left(\hat{\beta}(Z_{train}, \lambda_t); Z_{val}\right)$
- **Step 3:** *Use best λ_t*
  - *Choose $t' = \arg\min L_{val}^{t}$ with lowest validation loss*
  - Re-run linear regression with Z_{train} and $\lambda_{t'}$ to obtain $\hat{\beta}(Z_{train}, \lambda_{t'})$

Example : 3-Fold Cross Validation



Regularization

Feature Selection

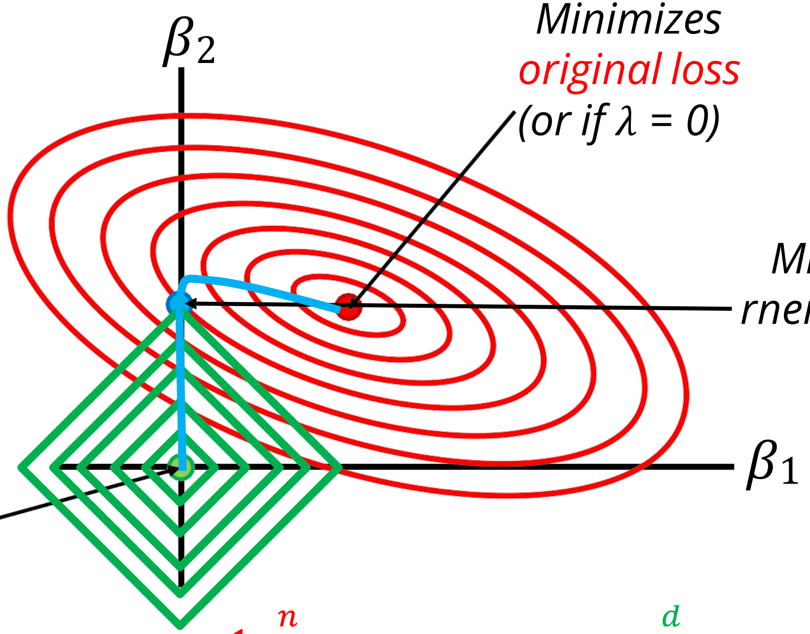
Feature Selection via L_1 Regularization

- 전통적인 *Sequential Feature Selection* 방법은 *High-dimension* 문제에서 시간이 너무 오래 걸리거나 *Full model* 계산이 불가능

- L_1 **Regularization:** *Model Estimation* 과정에서 동시에 *Feature Selection*

- 다른 *performance measure* 기반의 선택이 아닌 *model train* 과정에서 자체 *Feature* 학습

Intuition on L_1 Regularization



Minimizes
original loss
(or if $\lambda = 0$)

Minimizer of full loss at corner sparse ($\beta_1 = 0$)!

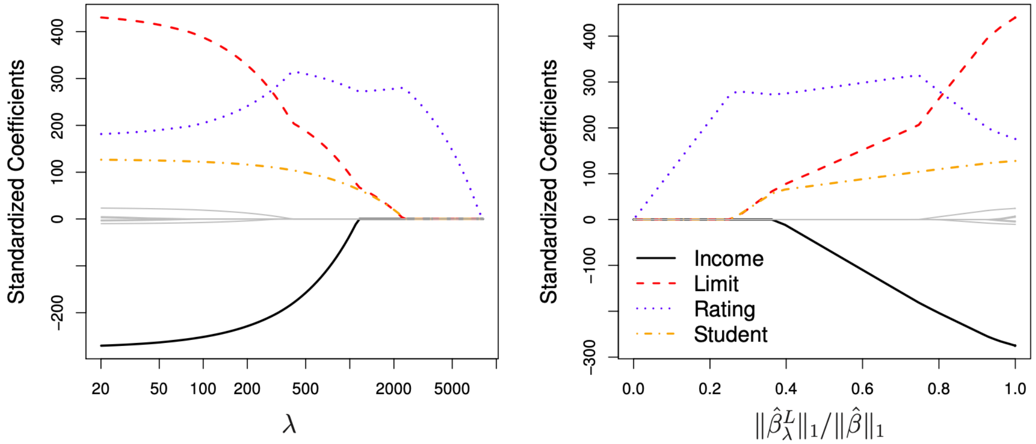
Minimizes
regularization term
(or if $\lambda \to \infty$)

$$L(\beta; Z) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta^T x_i)^2 + \lambda \sum_{j=1}^{d}|\beta_j|$$

L_1 Regularization for Feature Selection

- **Step 1:** *Construct a lot of features and add to feature map*
- **Step 2:** *Use L_1 regularized regression to "select" subset of features*
  - *I.e., coefficient $\beta_j \neq 0$ feature j is selected)*
- **Optional:** *Remove unselected features from the feature map and run vanilla linear r egression (a.k.a. ordinary least squares)*

Example : Lasso Solution Path



Given $\lambda > 0$, we can find the solution to the optimization problem, $\hat{\beta}^1_\lambda, \ldots, \hat{\beta}^p_\lambda$.

The Oracle Property

- *Original Model:*

$$Y_i = \beta_0 + \sum_{j=1}^{p} \beta_j X_{ij} + \epsilon_i, \qquad i = 1, \dots, n$$

- *Sparse Model: some of β_j is 0.*

$$Y_i = \beta_0 + \sum_{j \in \mathcal{A}} \beta_j X_{ij} + \epsilon_i, \qquad i = 1, \dots, n$$

$\mathcal{A} \subseteq \{1,2, \dots, p\}$: *Active set*

Feature Selection 이 얼마나 잘 되었나?

Limitation of Lasso

- *The estimate can be used for variable selection but it is biased.*
- *If n < p, lasso can select at most n and the solution is not unique.*
- *If two or more variables are highly correlated, lasso will choose one or a few of them randomly and shrink the rest to 0.*
- *To prove the selection consistency, it requires a very strong assumption and it is not easy to verify but it is easy to be violated.*
- *See the reference below for a more detailed review.*

Freijeiro-Gonz´alez, L., Febrero-Bande, M., and Gonz´alez-Manteiga, W. (2022) A Critical Review of LASSO and Its Derivatives for Variable Selection Under Dependence Among Covariates. International Statistical Review, 90: 118– 145.

The Oracle Property

- *Original Model:*

$$Y_i = \beta_0 + \sum_{j=1}^{p} \beta_j X_{ij} + \epsilon_i, \qquad i = 1, \dots, n$$

- *Sparse Model: some of β_j is 0.*

$$Y_i = \beta_0 + \sum_{j \in \mathcal{A}} \beta_j X_{ij} + \epsilon_i, \qquad i = 1, \dots, n$$

$\mathcal{A} \subseteq \{1,2, \dots, p\}$: *True Active set*

$\hat{\mathcal{A}} = \{j \subseteq \{1,2, \dots, p\}: \hat{\beta}_j = 0\}$: *Estimated Active set*

Feature Selection 이 얼마나 잘 되었나?

The Oracle Property

$\mathcal{A} \subseteq \{1,2,\dots,p\}$: *True Active set*

$\hat{\mathcal{A}} = \left\{j \subseteq \{1,2,\dots,p\}: \hat{\beta}_j = 0\right\}$: *Estimated Active set*

The Oracle Property

1. (Selection Consistency)

$$P\left(\hat{\mathcal{A}} = \mathcal{A}\right) \to 1$$

2. (Asymptotic Normality)

$$n^{-1/2}(\hat{\beta}_j - \beta_j) \xrightarrow{D} N(0, I_S^{-1})$$

3. (Estimation Consistency): weaker version of asymptotic normality

$$\hat{\beta}_j \xrightarrow{P} \beta_j$$

Methods with The Oracle Property

- **Smoothly clipped absolution deviation (SCAD)**

Fan & Li (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. JASA.

- **Adaptive Lasso**

Hui Zou (2006). The Adaptive Lasso and Its Oracle Properties. JASA.

- **Minimax Concave Penalty (MCP)**

Zhang (2010). Nearly unbiased variable selection under minimax concave penalt y. The Annals of Statistics.

Adaptive Lasso

- **Lasso:** *Minimize*

$$L(\beta; Z) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta^T x_i)^2 + \lambda \sum_{j=1}^{p}|\beta_j|$$

- **Adaptive Lasso:** *Minimize*

$$L(\beta; Z) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta^T x_i)^2 + \lambda \sum_{j=1}^{p} w_j|\beta_j|$$

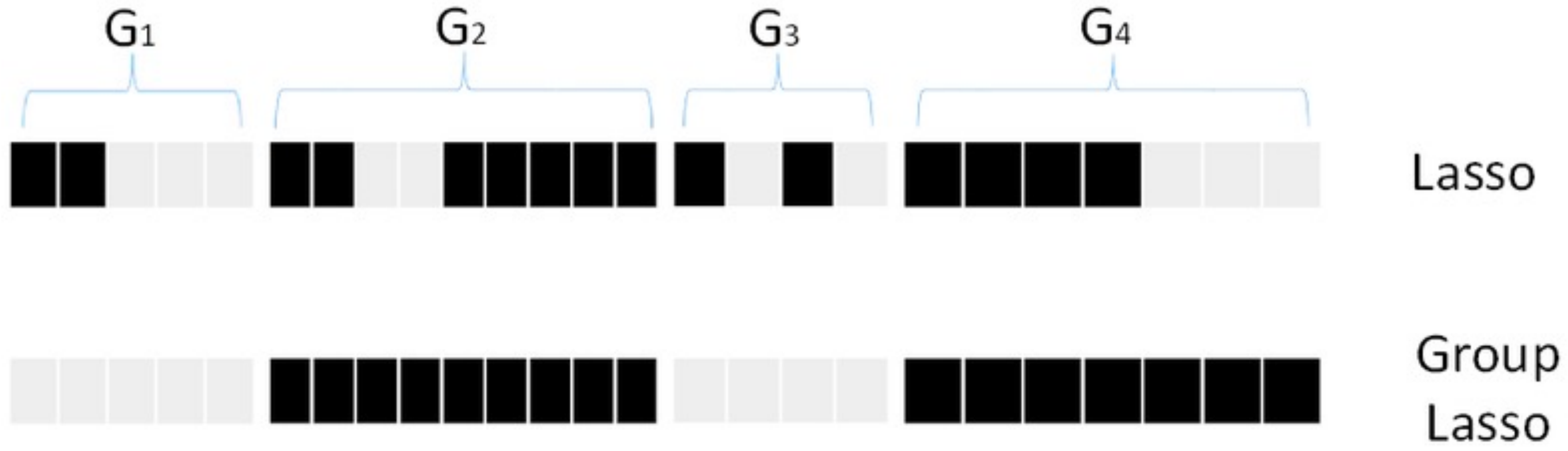
$$w_j = \frac{1}{|\hat{\beta}_j|}$$

More penalty if $\hat{\beta}_j$ is small

Less penalty if $\hat{\beta}_j$ is large

Group Lasso

- *When predictors are grouped, how does variable selection method work?*
- 같은 그룹끼리는 다 같이 *0* 혹은 *nonzero* 값이 되도록 하고싶다면?



Group Lasso

x1, (x2,x3) group

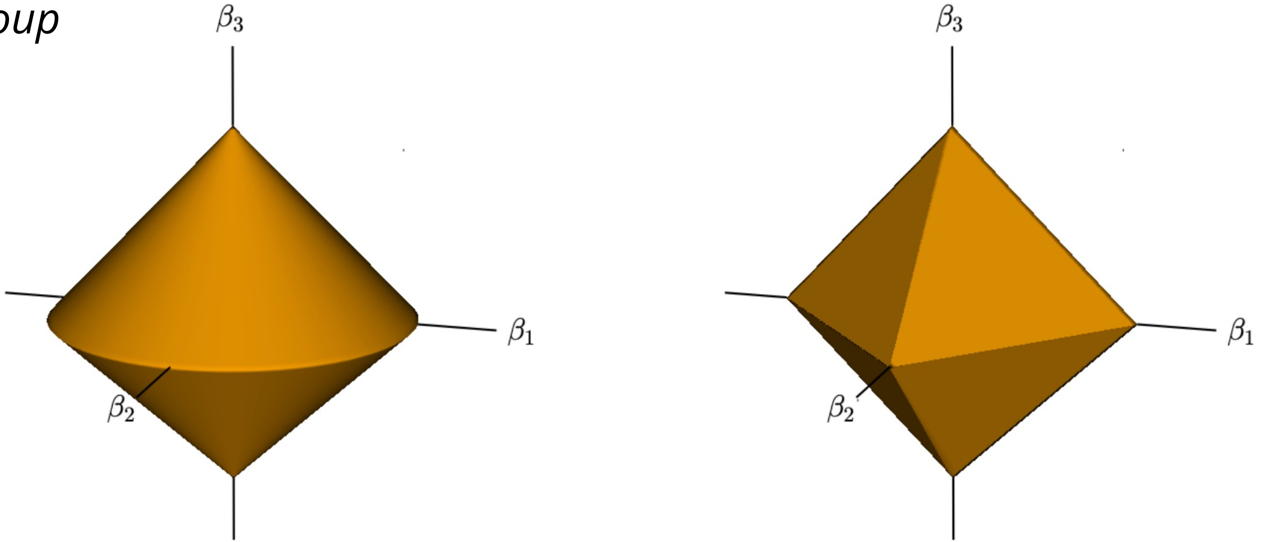


Figure 4.3 *The group lasso ball (left panel) in* $\mathbb{R}^3$*, compared to the* ℓ_1 *ball (right panel). In this case, there are two groups with coefficients* $\theta_1 = (\beta_1, \beta_2) \in \mathbb{R}^2$ *and* $\theta_2 = \beta_3 \in \mathbb{R}^1$.

Group Lasso

- *Assume we have a group structure on* X:

$$\boldsymbol{X} = (\boldsymbol{X}_1, \dots, \boldsymbol{X}_J) \text{ and } \boldsymbol{\beta} = (\boldsymbol{\beta}_1^T, \dots, \boldsymbol{\beta}_J^T)$$

With $\beta_j \in \mathbb{R}^{p_j}$, $j = 1, \dots, p$; *and* $\sum_{j=1}^{J} p_j = p$.

- *Centering both response and predictors assume* X_j *is orthonormalized to* Z_j *(i.e. ,* $Z_j^T Z_j = I_{p_j}$*), group LASSO solves*

$$\boldsymbol{min_{\beta} \frac{1}{2} \parallel Y - \sum_{j=1}^{J} Z_j \beta_j \parallel^2 + \lambda \sum_{j=1}^{J} \parallel \beta_j \parallel_2}$$

Where $\parallel \beta \parallel_2 = \sqrt{\beta_1^2 + \dots + \beta_p^2}$

Yian & Lin (2006) Model selection and estimation in regression with grouped variables JRSS-B, 68(1), 49-67

Qualitative Predictors

Extra notes on feature mapping

Feature Mapping for Qualitative Predictor

- **Example:** *investigate differences in credit card balance between* **males** *and* **females***, ignoring the other variables. We create a new variable*

$$x_i = \begin{cases} 1 & if\ ith\ person\ is\ female \\ 0 & if\ ith\ person\ is\ male. \end{cases}$$

Feature Mapping for Qualitative Predictor

- **Example:** *investigate differences in credit card balance between* **males** *and* **females**, *ignoring the other variables. We create a new variable*

$$x_i = \begin{cases} 1 & if\ ith\ person\ is\ female \\ 0 & if\ ith\ person\ is\ male. \end{cases}$$

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

$$y_i = \begin{cases} \beta_0 + \beta_1 + \epsilon_i & if\ ith\ person\ is\ female \\ \beta_0 + \epsilon_i & if\ ith\ person\ is\ male. \end{cases}$$

Feature Mapping for Qualitative Predictor

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

$$y_i = \begin{cases} \beta_0 + \beta_1 + \epsilon_i & if\ ith\ person\ is\ female \\ \beta_0 + \epsilon_i & if\ ith\ person\ is\ male. \end{cases}$$

- *β_1: difference of E(Y|X) male v.s. female*

More than two levels

- *With more than two levels for each variable, we create additional* **dummy variables.**

- $Y \sim X$
  - Y: *credit card balance*
  - X: *ethnicities (Asian, Caucasian, African American)*

More than two levels

- $Y \sim X$
  - Y: *credit card balance*
  - X: *ethnicities (Asian, Caucasian, African American)*

$$x_{i1} = \begin{cases} 1 & if\ \ ith\ \ person\ \ is\ Asian \\ 0 & if\ \ ith\ \ person\ \ is\ not\ Asian \end{cases}$$

And the second could be

$$x_{i2} = \begin{cases} 1 & if\ ith\ person\ is\ Caucasian, \\ 0 & if\ ith\ person\ is\ notCaucasian. \end{cases}$$

More than two levels

- $Y \sim X$
  - Y: *credit card balance*
  - X: *ethnicities (Asian, Caucasian, African American)*

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \varepsilon_i$$

$$y_i = \begin{cases} \beta_0 + \beta_1 + \varepsilon_i & if\ ith\ person\ is\ Asian, \\ \beta_0 + \beta_2 + \varepsilon_i & if\ ith\ person\ is\ Caucasian, \\ \beta_0 + \varepsilon_i & if\ ith\ person\ is\ African\ American. \end{cases}$$

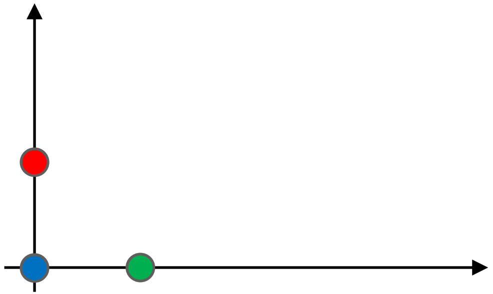
- **Baseline** *category: African American*
- β_1: *difference of E(Y|X) between African American and Asian*
- β_2: *difference of E(Y|X) between African American and Caucasian*

Dummy v.s. One-Hot Encoding

Color
Red
Green
Blue

Dummy encoding

d1	d2
1	0
0	1
0	0



Color
Red
Green
Blue

One-hot encoding

d1	d2	d3
1	0	0
0	1	0
0	0	1