Regression models

- Part 2

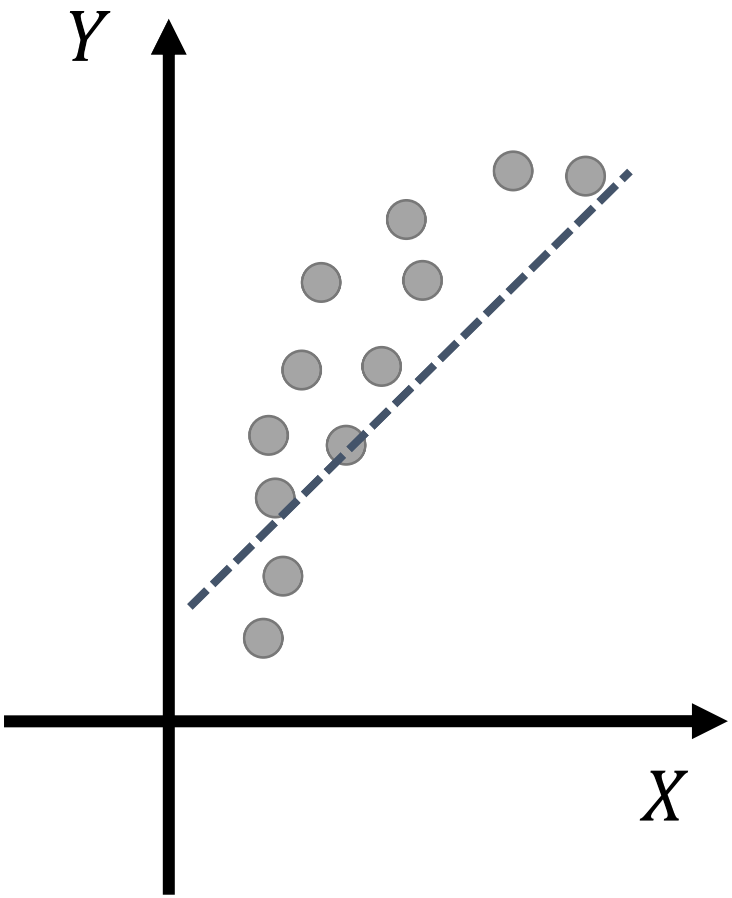
Taehoon Ko (thoon.koh@gmail.com)

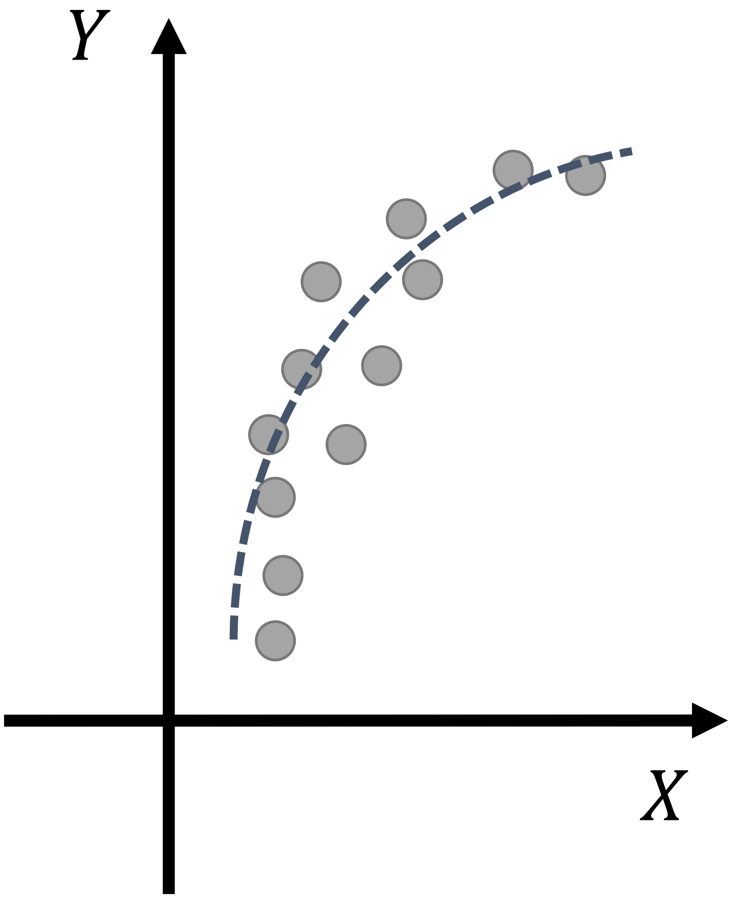
다항회귀모델

(Polynomial regression)

만약 X와 Y의 관계가 선형이 아니라면?

- 어떤 모델이 학습 데이터를 잘 설명하는가?





다항회귀모델 (Polynomial regression)

기존 다중선형회귀모델에서 입력변수들의 조합 및 제곱 변수를 생성하여 모델을 설계하면 된다.

- (입력 변수가 2개일 때) 선형회귀모델

"입력변수들을 추가하는 효과"

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2$$

- (입력 변수가 2개일 때) 2차 다항회귀모델
  - 각 입력 변수의 제곱항: X_1^2, X_2^2
  - 그리고 두 입력 변수의 교호작용 (interaction) 을 나타내는 항: $X_1 X_2$

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1^2 + \beta_4 X_2^2 + \beta_5 X_1 X_2$$

다항변수의 생성 (Generating polynomial features)

- sklearn.preprocessing.PolynomialFeatures 로 선언

- (꼭 회귀분석에서만 쓰는 것이 아니라) 여러 모델에서 사용 가능
  - 다항변수를 생성하는 것은, 하나의 데이터를 더 풍부한 표현으로 만드는 것이 가능함

- 너무 큰 차수로 설정하면 입력변수의 수가 매우 늘어나므로 주의

단계적 변수선택 회귀모델

(Stepwise regression)

어떻게 입력변수 집합을 결정할 것인가?

- **모델에서 이용하는 입력 변수의 집합이 달라지면, 모델의 성능이 달라진다.**
  - 어떤 입력 변수 집합이 가장 좋은 성능을 보일 것인가?
  - 이를 feature subset selection (변수 부분집합 선택)이라고 한다.

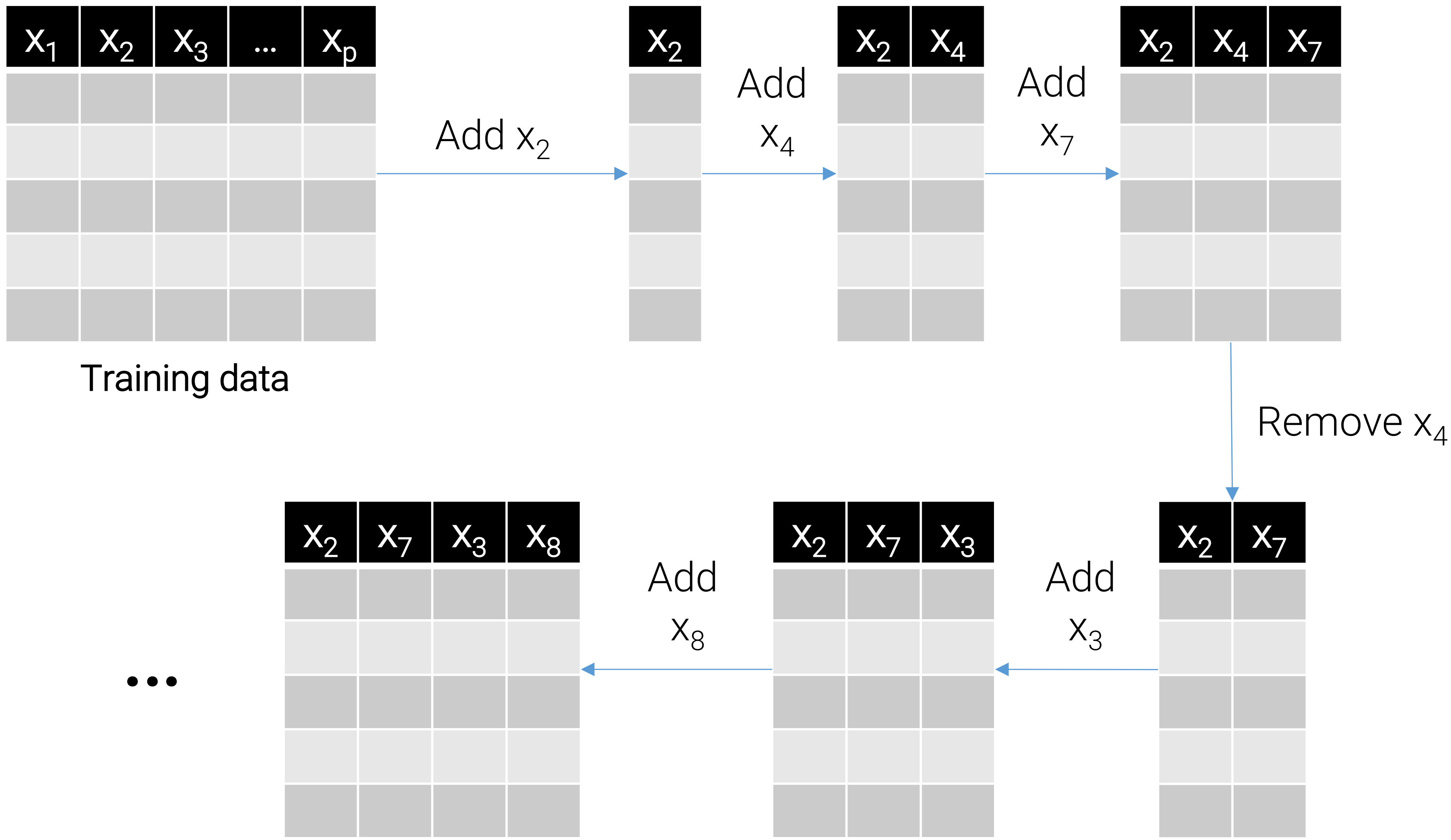
- Exhaustive search (전역 탐색)
  - The simplest method for finding an optimal feature subset
    : 모든 변수 집합을 탐색
  - But, we need too much time.
    : 변수의 수가 n개이면, 가능한 모든 부분집합의 수가 $2^n - 1$
  - (참고) Exhaustive search = Brute-force search

단계적 변수선택 회귀모델 (Stepwise regression)

- **단계적 선택법**
  - 입력변수 집합에 변수를 하나씩 추가하거나(전진선택법: Forward selection) 하나씩 제거하는(후진소거법: Backward elimination) 과정을 반복함
  - 입력변수 집합이 생성될 때마다 선형회귀모델을 학습하고 이를 평가하여 최적의 입력변수 집합을 탐색
  - 한 번 선택되거나 제거된 변수가 다시 선택/제거될 수 있음

단계적 변수선택 회귀모델 (Stepwise regression)

Example



Stepwise regression: Algorithm

- Initialize:
  - Start with model with no input variables.
  - *Selected* = null
- Loop
  - For each variable which is not in ***Selected***:
    - ***Selected*** = ***Selected*** + candidate variable
    - Build submatrix of X using ***Selected***
    - Train a regression model and evaluate it.
  - Find the best model and responding ***Selected***.

Forward selection phase

  - For each variable which is in ***Selected***:
    - ***Selected*** = ***Selected*** - candidate variable
    - Build submatrix of X using *Selected*
    - Train a regression model and evaluate it.
  - Find the best model and responding ***Selected***.

Backward elimination phase

Stepwise regression

매번 선택된 변수집합 (*Selected*) 으로 만들어진 모델을 어떤 지표로 평가? ➔ 앞서 배운 회귀모델 평가지표를 활용

- **기존 통계학에서의 접근**
  - Akaike Information Criteria (AIC)
  - Bayesian Information Criteria (BIC)
  - Adjusted-R^2: 기존의 R^2에 변수의 수를 고려
  - Mallow's C_k

$$\text{AIC} = n \cdot \ln\left(\frac{\text{SSE}_k}{n}\right) + 2k$$

$$\text{BIC} = n \cdot \ln\left(\frac{\text{SSE}_k}{n}\right) + k \cdot \ln(n)$$

$$\text{Adjusted-}R^2 = 1 - \left(\frac{n-1}{n-k-1}\right)(1-R^2)$$

$$C_k = \frac{SSE_k}{s^2} - (n-2k)$$

n : number of samples

k : number of selected variables

SSE_k : sum of squared error of regression model with k variables

s : sum of squared error of full regression model

Stepwise regression

매번 선택된 변수집합 (*Selected*) 으로 만들어진 모델을 어떤 지표로 평가? ➔ 앞서 배운 회귀모델 평가지표를 활용

- Train error나 test error도 사용 가능함.
  - Using **train error**
    - 앞에서의 AIC, BIC, Mallow's C_k, Adjusted-R^2 와 마찬가지로 Regression model이 학습데이터에 잘 적합했는가를 살펴보는 지표
  - Using **test error**
    - Regression model이 앞으로 새롭게 발생하는 데이터의 Y를 얼마나 잘 예측할 것인가를 살펴보는 지표

단계적 변수선택 (Stepwise feature selection)

- (꼭 회귀분석에서만 쓰는 것이 아니라) 여러 모델에서 사용 가능
  - 그러나 하나의 모델 학습이 오래 걸리는 경우에는 매우 부적절한 선택

    (Example) Support Vector Machine (SVM), Neural Network (NN)

- Scikit-learn에서는 단계적 변수선택을 제공하지 않음
  - [참고] Scikit-learn에서 제공하는 변수선택법

    ➔ http://scikit-learn.org/stable/modules/feature_selection.html
  - 수업 후반부에 '차원 축소 방법'에 대해 자세히 다룰 예정

Ridge regression

&

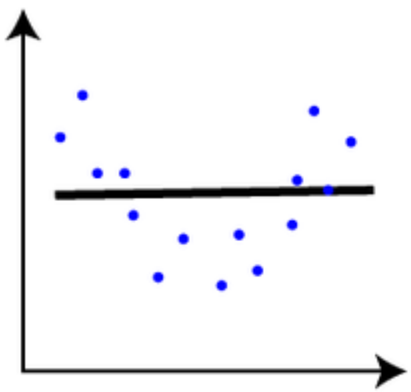
Lasso (Least Absolute Shrinkage and Selection Operator)

Motivation: 입력변수의 수와 과적합 문제 (Overfitting)

(데이터 샘플 수에 비해) 입력변수의 수가 너무 많아지면, 현재 데이터의 적합도는 매우 좋음에도 불구하고 **예측 성능이 떨어지는 문제**가 발생한다.

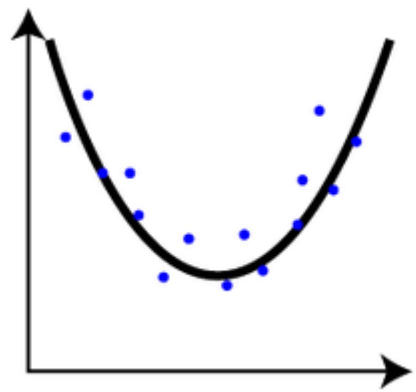
- 예: 다항회귀모델

● (모델이 본) 학습 데이터포인트



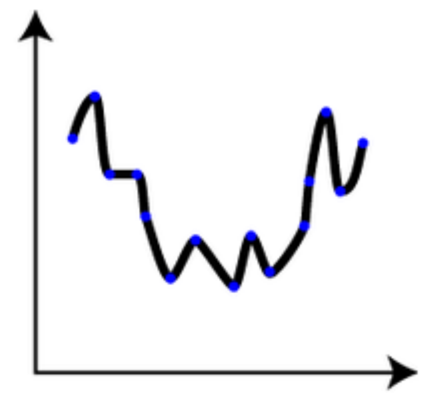
$$Y = a$$

Training RMSE: 10



$$Y = aX^2 + bX + c$$

Training RMSE: 5



$$Y = aX^{10} + bX^9 + \cdots + jX^{10} + k$$

Training RMSE: 0.1

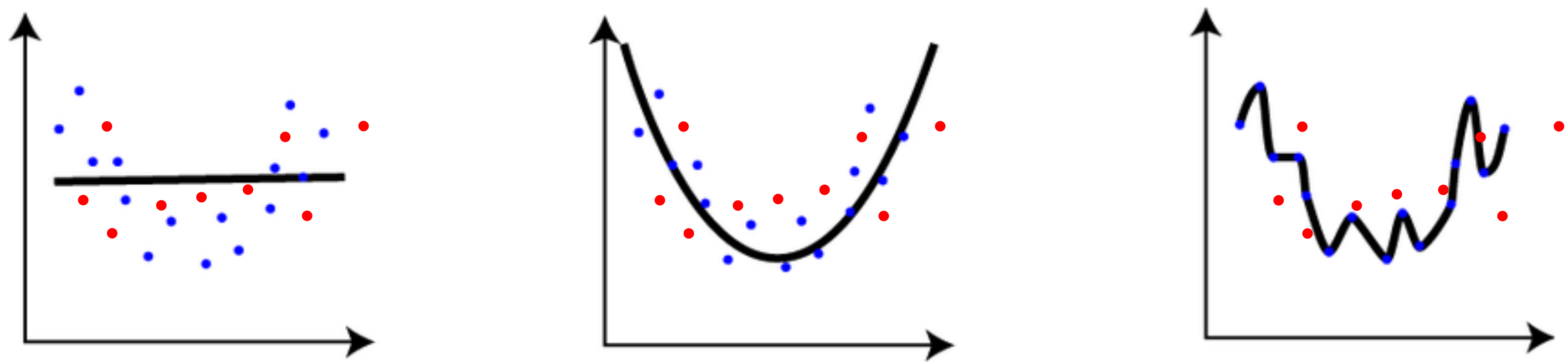
Motivation: 입력변수의 수와 과적합 문제 (Overfitting)

(데이터 샘플 수에 비해) 입력변수의 수가 너무 많아지면, 현재 데이터의 적합도는 매우 좋음에도 불구하고 **예측 성능이 떨어지는 문제**가 발생한다.

- 예: 다항회귀모델

● (모델이 본) 학습 데이터포인트

● (모델이 보지 못한) 테스트 데이터포인트



$Y = a$

$Y = aX^2 + bX + c$

$Y = aX^{10} + bX^9 + \cdots + jX^{10} + k$

Training RMSE: 10 | Training RMSE: 5 | Training RMSE: 0.1

Test RMSE: 12 | Test RMSE: 6 | Test RMSE: 20

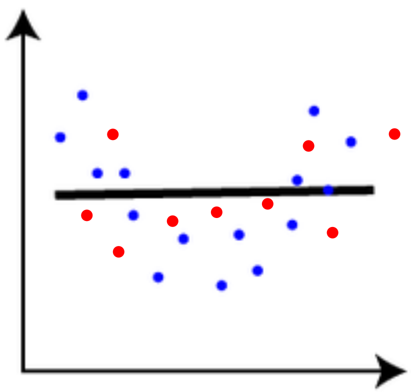
Motivation: 입력변수의 수와 과적합 문제 (Overfitting)

(데이터 샘플 수에 비해) 입력변수의 수가 너무 많아지면, 현재 데이터의 적합도는 매우 좋음에도 불구하고 **예측 성능이 떨어지는 문제**가 발생한다.

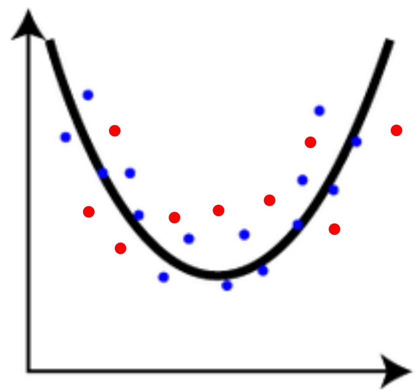
- 예: 다항회귀모델

● (모델이 본) 학습 데이터포인트

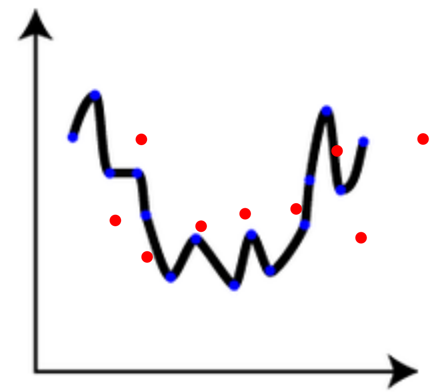
● (모델이 보지 못한) 테스트 데이터포인트



$Y = a$



$Y = aX^2 + bX + c$



$Y = aX^{10} + bX^9 + \cdots + jX^{10} + k$

과소적합
(Underfitting)

과적합
(Overfitting)

Motivation: 계수의 크기

오버피팅된 선형회귀모델은 계수의 크기가 큰 경향이 있다.

- 계수가 크면 해당 변수의 변화에 매우 민감한 모델이 되며,
  이는 새로운 데이터에 대한 예측 성능의 하락으로 이어진다.
- 예) 같은 학습 데이터에서 다음과 같은 두 모델이 도출되었을 때,
  - 1번 모델: $Y = 1 + 2X_1 + \cdots$
  - 2번 모델: $Y = 0.1 + 100X_1 + \cdots$

  - 1번 모델은 X_1의 값이 1 증가할 때, Y가 1이 증가
  - 2번 모델은 X_1의 값이 1 증가할 때, Y가 100이 증가
    => 값 변화에 매우 민감한 모델 => 예측 성능 하락 예상

Regularization (제약)

Regularization은 모델의 복잡도에 대한 제약(혹은 penalty)을 학습 과정에 반영하는 것. (매우 중요!)

- 하는 이유?
  - 학습 데이터에 너무 과적합(overfitting)하여, 새롭게 등장하는 데이터에 대한 예측 성능이 떨어지는 것을 방지 ➔ "Generalization"
  - 더 자세한 내용은 추후 [편향-분산 트레이드오프 (Bias-variance tradeoff)]에서 더욱 자세히 다룰 예정

선형회귀모델에서의 제약

손실함수 (Mean Squared Error) 를 너무 줄이면
과적합이 발생한다.

따라서 과적합이 발생하지 않도록, **계수 크기**에 **제약**을
두자.

총 비용함수 = 손실함수 + 계수 크기에 대한 페널티

Trade-off 관계

Ridge and Lasso

Ridge와 Lasso는 기존 다중선형회귀모델의 손실함수에 계수 크기에 대한 페널티를 더한 함수를 사용하여 모델을 학습

Multiple Linear Regression

$$\min \sum_{i=1}^{n} \{y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \cdots + \hat{\beta}_p x_{ip})\}^2$$

λ : A parameter to control effectiveness of penalty

Lasso

$$\min \sum_{i=1}^{n} \{y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \cdots + \hat{\beta}_p x_{ip})\}^2 + \lambda \sum_{j=1}^{p} |\beta|$$

L1 penalty

Ridge regression

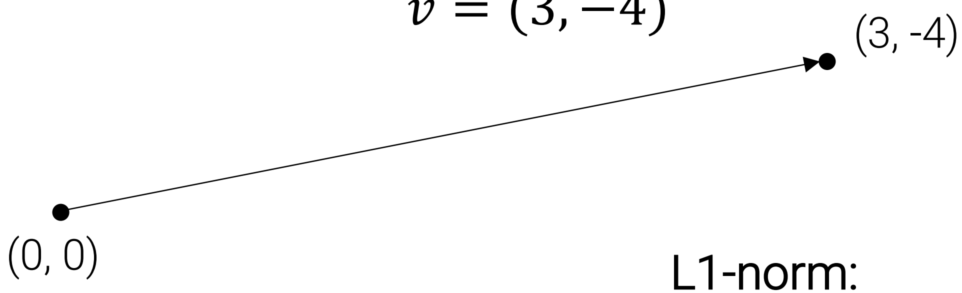
$$\min \sum_{i=1}^{n} \{y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \cdots + \hat{\beta}_p x_{ip})\}^2 + \lambda \sum_{j=1}^{p} \beta^2$$

L2 penalty

(참고) L1-norm, L2-norm

Norm은 벡터의 크기를 나타내는 함수이며, 절대값과 의미가 유사함.

$\vec{v} = (3, -4)$



L1-norm: $\|\vec{v}\|_1 = |3| + |-4| = 7$

L2-norm: $\|\vec{v}\|_2 = \sqrt{3^2 + (-4)^2} = 5$

⋮

Lp-norm: $\|\vec{v}\|_p = (|3|^p + |-4|^p)^{\frac{1}{p}}$

Ridge regression (능형회귀분석)

- Ridge regression의 회귀계수

$$\hat{\boldsymbol{\beta}}_{ridge} = \arg\min_{\boldsymbol{\beta}} \left\{ \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|^2 + \lambda \|\boldsymbol{\beta}\|^2 \right\}$$
$$= ((\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{Y}$$

$$\min_{\boldsymbol{\beta}} \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|^2 \quad \text{subject to } \|\boldsymbol{\beta}\|^2 = \sum_{j=1}^{p} \beta_j^{\,2} \le s$$

  - 계수의 크기에 대한 L2-norm penalty를 부여하여 모델의 overfitting을 방지
  - Ridge regression은 전체 계수의 크기를 최대한 작게 만드는 동시에 회귀 모델의 성능을 올림

Lasso regression

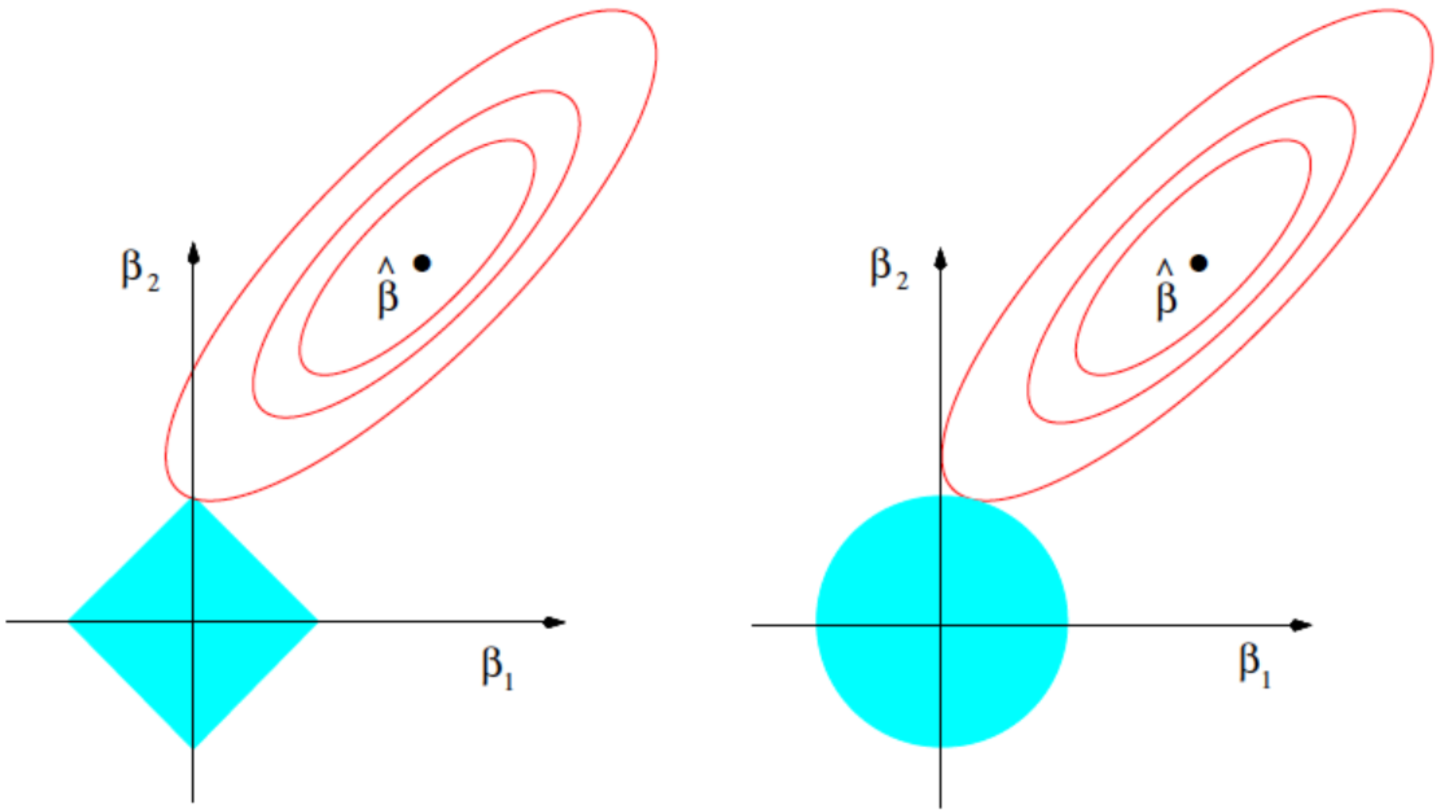
- Least absolute shrinkage and selection operator (LASSO) regression

$$\hat{\boldsymbol{\beta}}_{lasso} = \arg\min_{\boldsymbol{\beta}} \left\{ \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|^2 + \lambda \|\boldsymbol{\beta}\| \right\}$$

$$\min_{\boldsymbol{\beta}} \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|^2 \quad \text{subject to} \quad \|\boldsymbol{\beta}\| = \sum_{j=1}^{p} |\beta_j| \le t$$

  - 계수의 크기에 대한 L1-norm penalty를 부여하여 모델의 overfitting을 방지
  - Lasso regression은 전체 입력변수 계수 중 일부를 0으로 만들어 입력변수를 선택하는 효과가 있음 ➔ Sparse modeling

Lasso regression vs. Ridge regression



T. Hastie, R. Tibshirani, J. Friedman. *The elements of statistical learning : data mining, inference , and prediction*. Springer, 2011

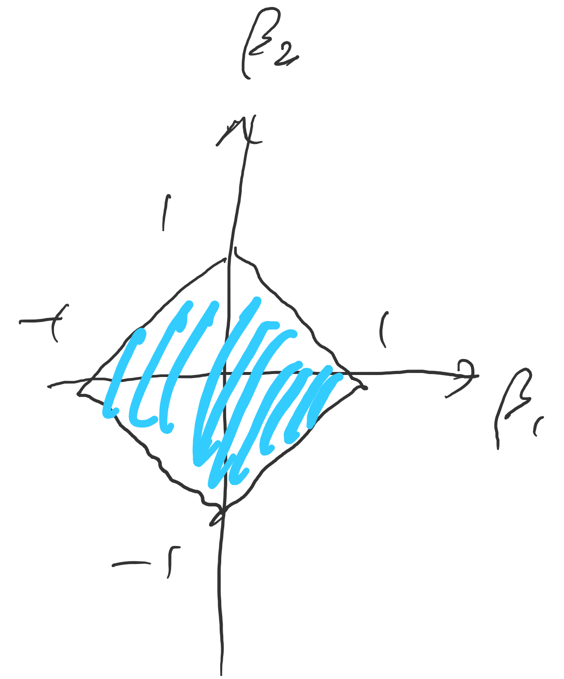
$|\beta_1| + |\beta_2| \leq 1$

i) if $\beta_1 \geq 0, \beta_2 \geq 0$, then $\beta_1 + \beta_2 \leq 1$

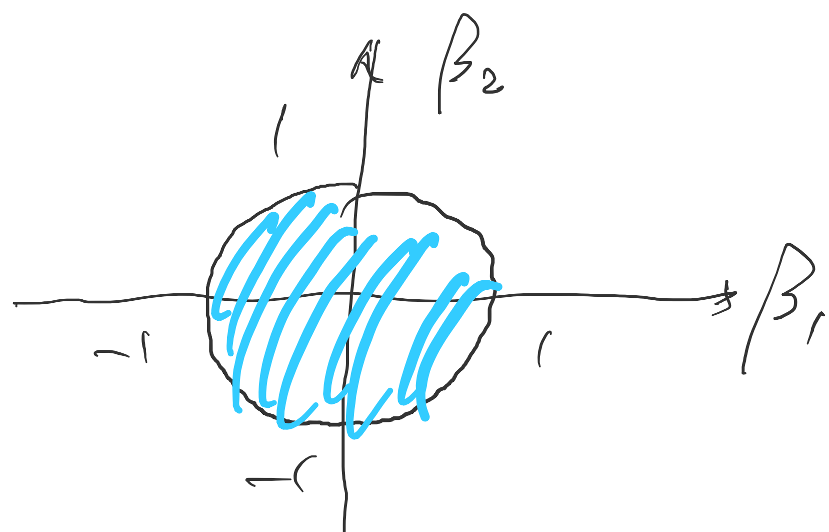
ii) if $\beta_1 \geq 0, \beta_2 < 0$, then $\beta_1 - \beta_2 \leq 1$

iii) if $\beta_1 < 0, \beta_2 \geq 0$, then $-\beta_1 + \beta_2 \leq 1$

iv) if $\beta_1 < 0, \beta_2 < 0$, then $-\beta_1 - \beta_2 \leq 1$



$\beta_1^2 + \beta_2^2 \leq 1$



$$\text{loss} = \frac{1}{n}\sum_{i=1}^{n}\left\{ y_i - \left(\cancelto{0}{\beta_0} + \beta_1 x_{i1} + \beta_2 x_{i2}\right)\right\}^2$$

(가정)

$$= \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \beta_1 x_{i1} - \beta_2 x_{i2}\right)^2$$

$$= \frac{1}{n}\sum_{i=1}^{n}\left(y_i^2 + \beta_1^2 x_{i1}^2 + \beta_2 x_{i2}^2 - 2\beta_1 x_{i1} y_i - 2\beta_2 x_{i2} y_i + 2\beta_1\beta_2 x_{i1} x_{i2}\right)$$

$$= \cdots = A\cdot\beta_1^2 + B\cdot\beta_2^2 + C\beta_1 + D\beta_2 + E\beta_1\beta_2$$

$$(A > 0,\ B > 0)$$