[SWCON253] Machine Learning – Lec.12

# Overfitting & Regularization

Fall 2025

김휘용

hykim.v@khu.ac.kr

경희대학교 KYUNG HEE UNIVERSITY

# Contents



**References**

- *기계학습* by 오일석 (한빛아카데미, http://cv.jbnu.ac.kr/index.php?mid=ml)
- *Intro to Machine Learning* by Dmitry Kobak @Tubingen Univ. (https://www.youtube.com/watch?v=brkS6rAKTl4&list=PL05umP7R6ij35ShKLDqccJSDntugY4FQT&index=3)

# 1. Overfitting & Underfitting

- ✓ Underfitting
- ✓ Overfitting
- ✓ Causes of Overfitting

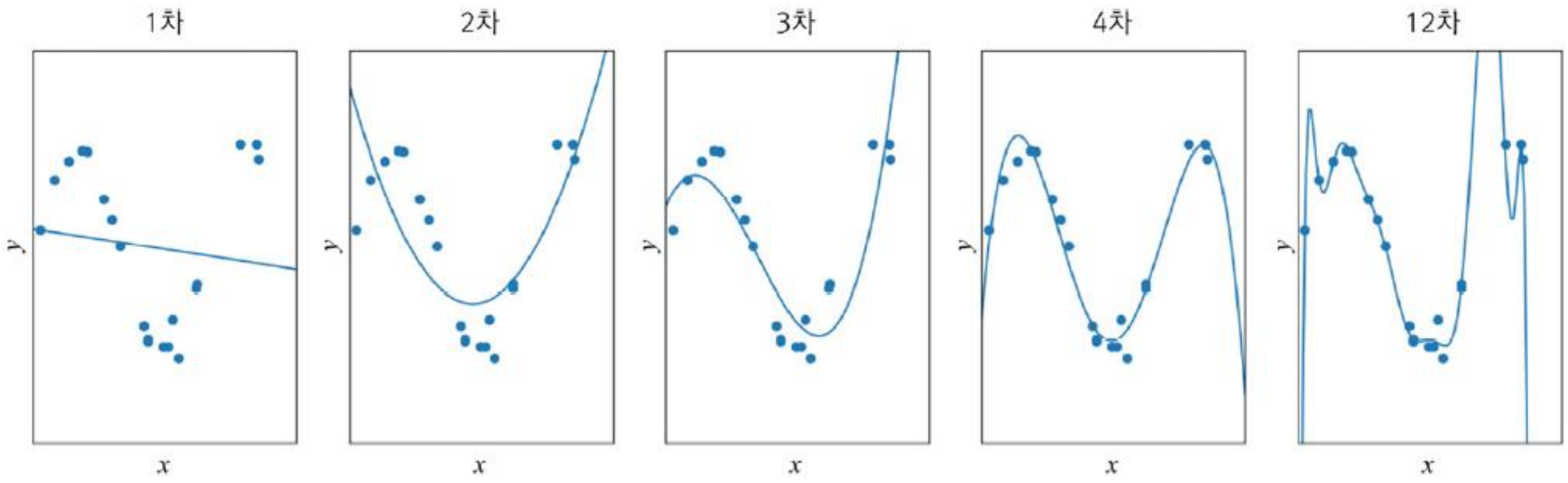
# Underfitting

## ◆ Underfitting (과소적합)과 훈련 오차

- 모델의 '용량이 너무 작아' (훈련집합에 대해서 조차) 오차가 클 수밖에 없는 현상
- 예) 아래 그림의 선형(1차 다항식) 또는 2차 다항식 모델을 사용한 경우

## ◆ Underfitting 방지

- 비선형 모델 등과 같이 **용량이 더 큰 모델**을 사용한다
- 예) 아래 그림의 3차, 4차, 12차 다항식 모델의 경우



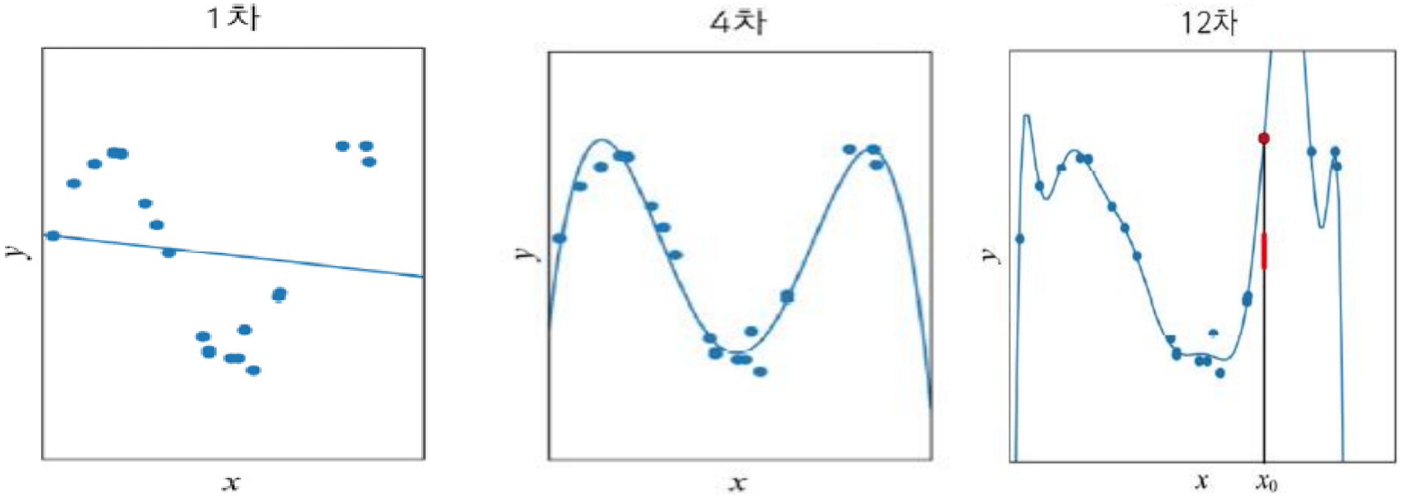
# Overfitting

## Overfitting (과잉적합)과 예측(시험) 오차

- 앞의 예에서, 12차 다항식 곡선을 채택한다면 훈련집합에 대해 거의 완벽하게 근사화함
- 하지만 '새로운' 데이터를 예측한다면 큰 문제 발생 (빨간 점)
- 이유는 '용량이 너무 크기' 때문에 학습 과정에서 잡음까지 수용 → 과잉적합 현상

## Overfitting 방지: 다양한 방법이 있음

- **적절한 용량의 모델을 선택**하는 모델 선택 작업을 수행
- 앞의 예에서는 4차 다항식을 적절한 모델로 볼 수 있음

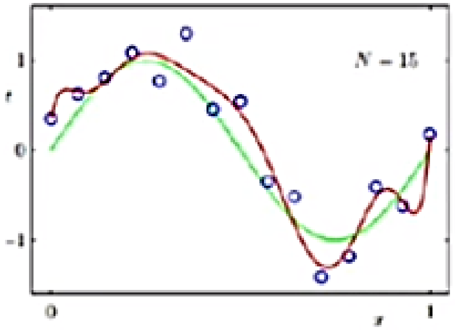



- 1차: 훈련집합과 테스트집합 모두 낮은 성능
- **12차**: 훈련집합에 높은 성능, 테스트집합에는 낮은 성능 → **낮은 일반화 능력 (과잉적합)**
- **4차**: 훈련집합에 12차보다 낮은 성능, 테스트집합에는 높은 성능 → **높은 일반화 능력**

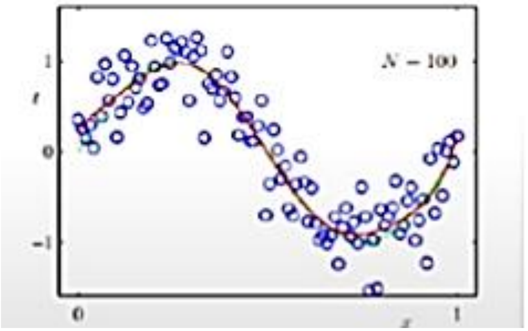
# Causes of Overfitting – *1. Data*

- **Insufficient # of Training Examples**
  - the training set may be too sparse or cannot represent the full variety of the data

N=15




N=100




N : # of traing examples

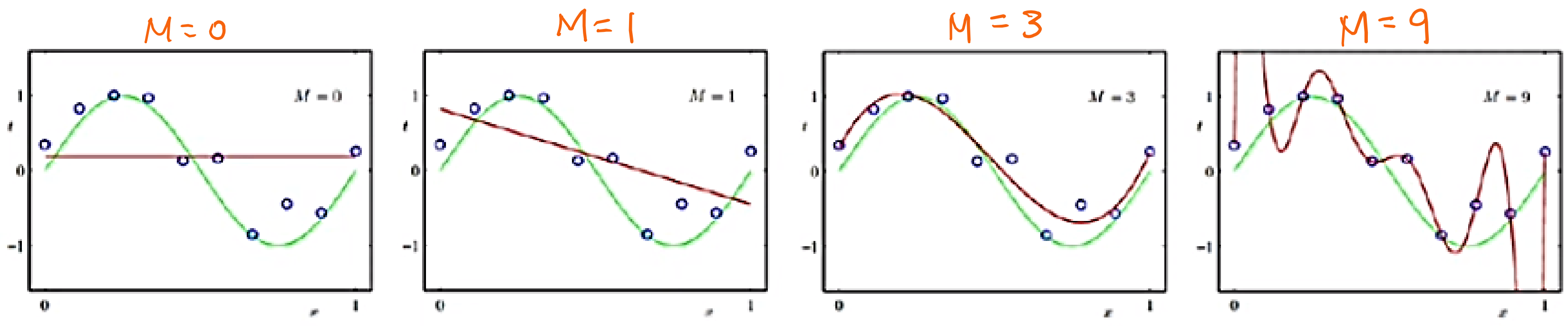
  - **해결책:** **충분히 많은 Training Data** 사용

    ★ Cf.) 데이터 증대(Data Augmentation) 기법 등을 통해 기존 Training Data 증대 가능

# Causes of Overfitting – 2. *Model*

- **Too Large # of Parameters (Model Capacity)**
  - the model is relatively too flexible for the dataset
  - the resulting parameters tend to have large values

M: the highest order of the polynomial
( M+1 : the # of parameters )




| | $M = 0$ | $M = 1$ | $M = 3$ | $M = 9$ |
|---|---|---|---|---|
| $w_0^\star$ | 0.19 | 0.82 | 0.31 | 0.35 |
| $w_1^\star$ | | -1.27 | 7.99 | 232.37 |
| $w_2^\star$ | | | -25.43 | -5321.83 |
| $w_3^\star$ | | | 17.37 | 48568.31 |
| $w_4^\star$ | | | | -231639.30 |
| $w_5^\star$ | | | | 640042.26 |
| $w_6^\star$ | | | | -1061800.52 |
| $w_7^\star$ | | | | 1042400.18 |
| $w_8^\star$ | | | | -557682.99 |
| $w_9^\star$ | | | | 125201.43 |

→ Magnitudes of parameters are very large!

Figures from: Bishop, "*Pattern Recognition and Machine Learning*"

# Causes of Overfitting – 2. *Model* (cont'd)

◆ 해결책 1: 검증집합(Validation Set)을 이용한 모델선택(Model Selection)

- 훈련집합과 테스트집합과 다른 별도의 **검증집합**을 준비한다.
- **모델집합**에 속한 각각의 모델에 대해 **훈련집합**으로 학습시킨다. (훈련 성능)
  - ★ 앞의 예에서는 서로 다른 차수의 다항식의 집합(서로 다른 용량)이 모델집합인 셈
- **검증집합**에 대해 최고의 성능을 보인 모델을 선택한다. (검증 성능) → ***Overfitting 방지***

→ *Lecture 13 'Model Evaluation' 에서 배울 예정*

◆ 해결책 2: 규제(Regularization)

- **용량이 충분히 큰 모델 + 다양한 규제(*Regularization*) 기법을 적용**
  - ★ 예시: Weight Penalty, Drop-out...
  - ★ Overfitting을 방지하기 위한 기술을 통칭하여 '규제'라고 부르기도 함
  - ★ **Regularization Parameter들은 *Validation Set*을 이용하여 결정 가능 (model selection)**

→ *다음 슬라이드부터 배울 예정*

# 2. Regularization by Weight Penalty

- ✓ Regularization
- ✓ Regularization by Weight Penalty
- ✓ L1 Norm vs. L2 Norm
- ✓ Selecting Lambda
- ✓ Do not penalize the bias!
- ✓ Example: Linear Regression

# Regularization (규제)

- 규제는 오래 전부터 수학과 통계학에서 연구해온 주제
  - 모델 용량에 비해 데이터가 부족한 경우의 불량 문제를ill-posed problem 푸는 데 사용

$$\underbrace{J_{regularized}(\Theta)}_{\text{규제를 적용한 목적함수}} = \underbrace{J(\Theta)}_{\text{목적함수}} + \lambda \underbrace{R(\Theta)}_{\text{규제 항}}$$

  - 현대 기계 학습도 규제를 널리 사용함

- 『Deep Learning』 책의 규제 정의
  - "… any modification we make to a learning algorithm that is intended to *reduce its generalization error* …" (일반화 오류를 줄이려는 의도를 가지고 학습 알고리즘을 수정하는 방법 모두)

- 명시적 규제와 암시적 규제
  - **명시적 규제**: 가중치 감쇠나 드롭아웃처럼 목적함수나 신경망 구조를 직접 수정하는 방식
  - **암시적 규제**: 조기 멈춤, 데이터 증대, 잡음 추가, 앙상블처럼 간접적으로 영향을 미치는 방식

# Regularization by Weight Penalty (가중치 감쇠)

◆ Regularized Cost Function

$$\underbrace{J_{regularized}(\Theta;\mathbb{X},\mathbb{Y})}_{\text{규제를 적용한 목적함수}} = \underbrace{J(\Theta;\mathbb{X},\mathbb{Y})}_{\text{목적함수}} + \lambda \underbrace{R(\Theta)}_{\text{규제 항}}$$

- **규제항**은 훈련집합과 무관하며, 데이터 생성 과정에 내재한 **사전 지식**에 해당

◆ 규제항 $R(\Theta)$로 무엇을 사용할 것인가? → **가중치 감쇠 (가중치 벌칙)**

- 큰 가중치($\Theta$)에 벌칙을 가해 작은 가중치를 유지. 주로 $L2$ 놈이나 $L1$ 놈을 사용
  - ★ L2 norm 사용: $R(\Theta) = \|\Theta\|_2^2$
  - ★ L1 norm 사용: $R(\Theta) = \|\Theta\|_1$
  - ★ 최종해를 원점 가까이 당기는 효과 (즉 가중치를 작게 유지함)
- 가중치 감쇠는 모델의 구조적 용량을 충분히 크게 하고 모델의 **'수치적 용량'을 제한**하는 규제 기법임

# Regularization – *L2 Norm*

## Regularized Cost & Gradient

$$\underbrace{J_{regularized}(\mathbf{\Theta};\mathbb{X},\mathbb{Y})}_{\text{규제를 적용한 목적함수}} = \underbrace{J(\mathbf{\Theta};\mathbb{X},\mathbb{Y})}_{\text{목적함수}} + \lambda \underbrace{\|\mathbf{\Theta}\|_2^2}_{\text{규제 항}}$$

$$\nabla J_{regularized}(\mathbf{\Theta};\mathbb{X},\mathbb{Y}) = \nabla J(\mathbf{\Theta};\mathbb{X},\mathbb{Y}) + 2\lambda\mathbf{\Theta}$$

## Parameter Update

$$\mathbf{\Theta} = \mathbf{\Theta} - \rho\nabla J_{regularized}(\mathbf{\Theta};\mathbb{X},\mathbb{Y})$$

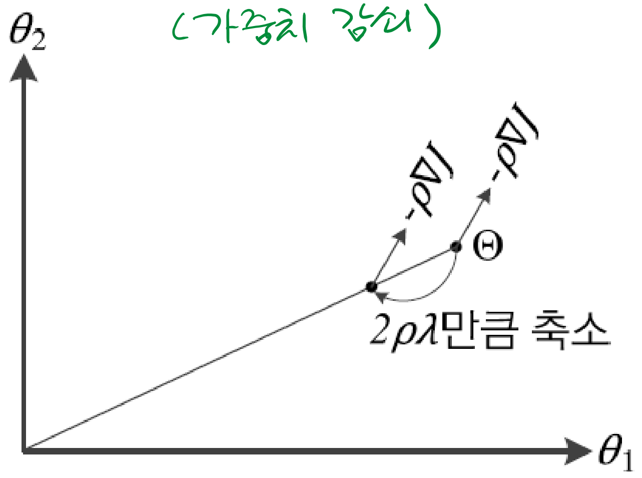
$$= \mathbf{\Theta} - \rho(\nabla J(\mathbf{\Theta};\mathbb{X},\mathbb{Y}) + 2\lambda\mathbf{\Theta})$$

$$= \boxed{(1-2\rho\lambda)}\mathbf{\Theta} - \rho\nabla J(\mathbf{\Theta};\mathbb{X},\mathbb{Y})$$

- L2 규제는 $\mathbf{\Theta}$를 $2\rho\lambda$의 비율로 줄인 후 업데이트 하는 셈

★ 즉, 가중치 감소 정도가 현재 가중치 크기에 비례함

"weight decay"
(가중치 감쇠)



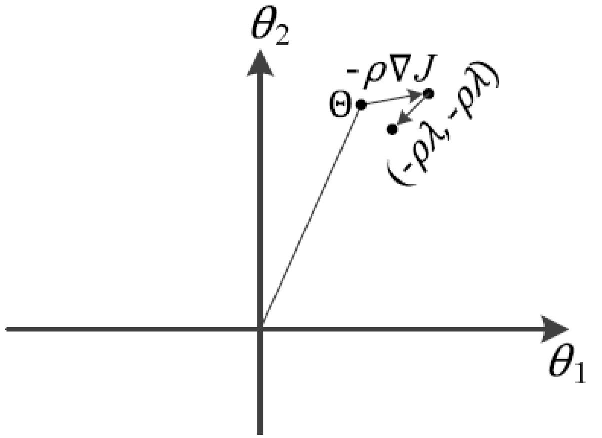
# Regularization – *L1 Norm*

## Regularized Cost & Gradient

$$\underbrace{J_{regularized}(\boldsymbol{\Theta}; \mathbb{X}, \mathbb{Y})}_{\text{규제를 적용한 목적함수}} = \underbrace{J(\boldsymbol{\Theta}; \mathbb{X}, \mathbb{Y})}_{\text{목적함수}} + \lambda \underbrace{\|\boldsymbol{\Theta}\|_1}_{\text{규제 항}}$$

$$\nabla J_{regularized}(\boldsymbol{\Theta}; \mathbb{X}, \mathbb{Y}) = \nabla J(\boldsymbol{\Theta}; \mathbb{X}, \mathbb{Y}) + \underline{\lambda \mathbf{sign}(\boldsymbol{\Theta})}$$

sign(Θ) : Θ의 부호벡터
(1,-1)

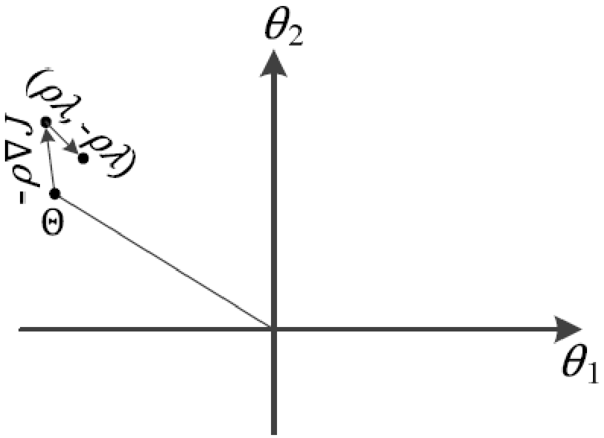



(a) $\mathbf{sign}(\boldsymbol{\Theta}) = (1,1)^{\mathrm{T}}$인 경우

## Parameter Update

$$\begin{aligned}\boldsymbol{\Theta} &= \boldsymbol{\Theta} - \rho \nabla J_{regularized}(\boldsymbol{\Theta}; \mathbb{X}, \mathbb{Y}) \\ &= \boldsymbol{\Theta} - \rho(\nabla J(\boldsymbol{\Theta}; \mathbb{X}, \mathbb{Y}) + \lambda \mathbf{sign}(\boldsymbol{\Theta})) \\ &= \boldsymbol{\Theta} - \rho \nabla J(\boldsymbol{\Theta}; \mathbb{X}, \mathbb{Y}) \boxed{- \rho\lambda \mathbf{sign}(\boldsymbol{\Theta})}\end{aligned}$$

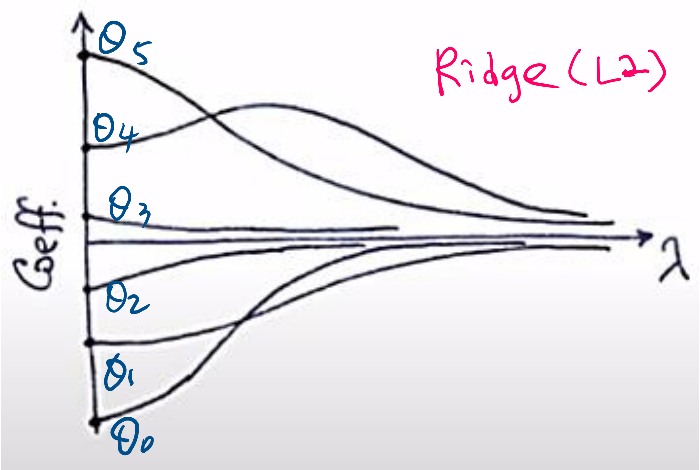
- L1 규제는 $\boldsymbol{\Theta}$를 $\rho\lambda$(고정값)만큼 줄인 후 업데이트 하는 셈
- L1 규제의 희소성(Sparse) 효과: 0이 되는 가중치가 많이 발생
  ★ 선형 회귀에 적용하면 특징 선택 효과

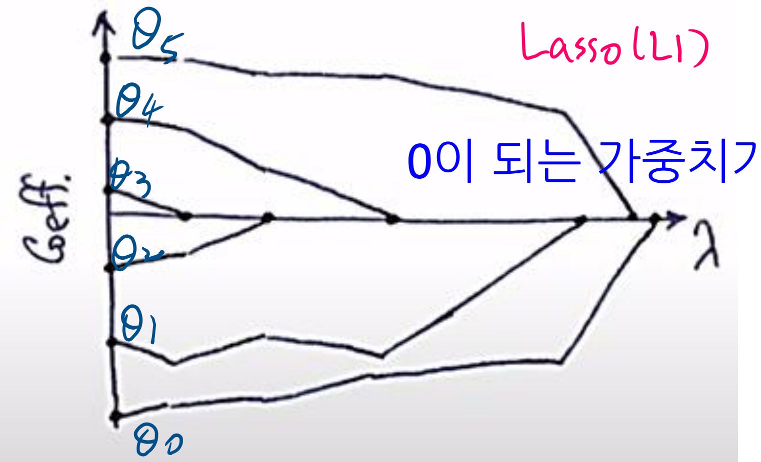



(b) $\mathbf{sign}(\boldsymbol{\Theta}) = (-1,1)^{\mathrm{T}}$인 경우

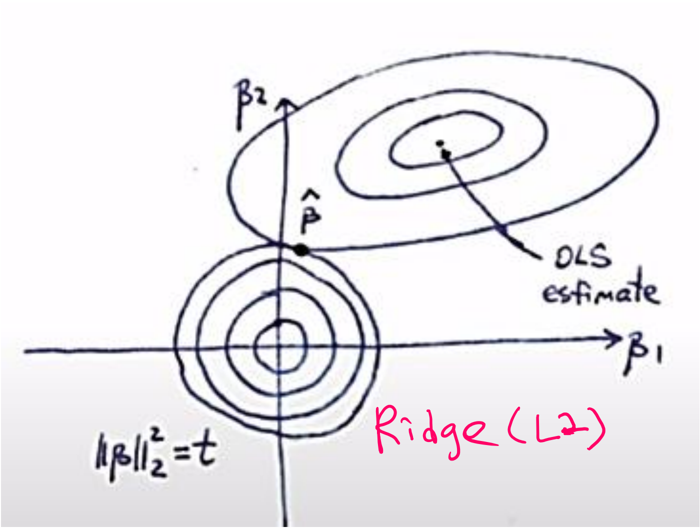
# Regularization – *L1 norm vs. L2 norm*

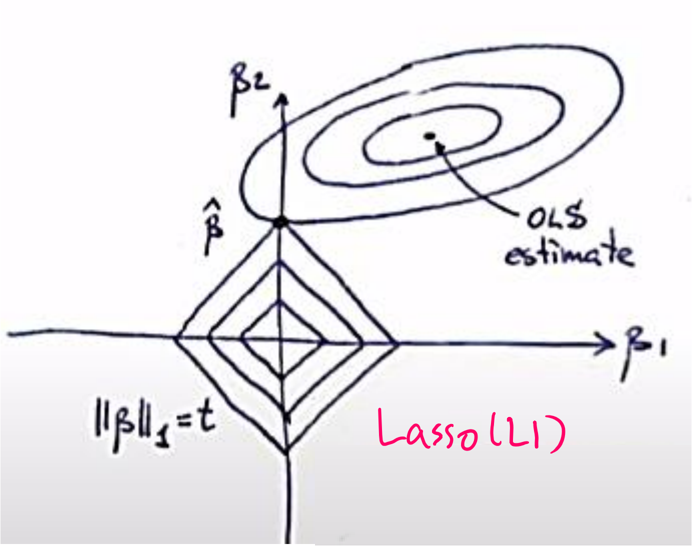
◆ Ridge (L2) vs. Lasso (L1) Regression (Linear Regression with L2-/L1-Norms)



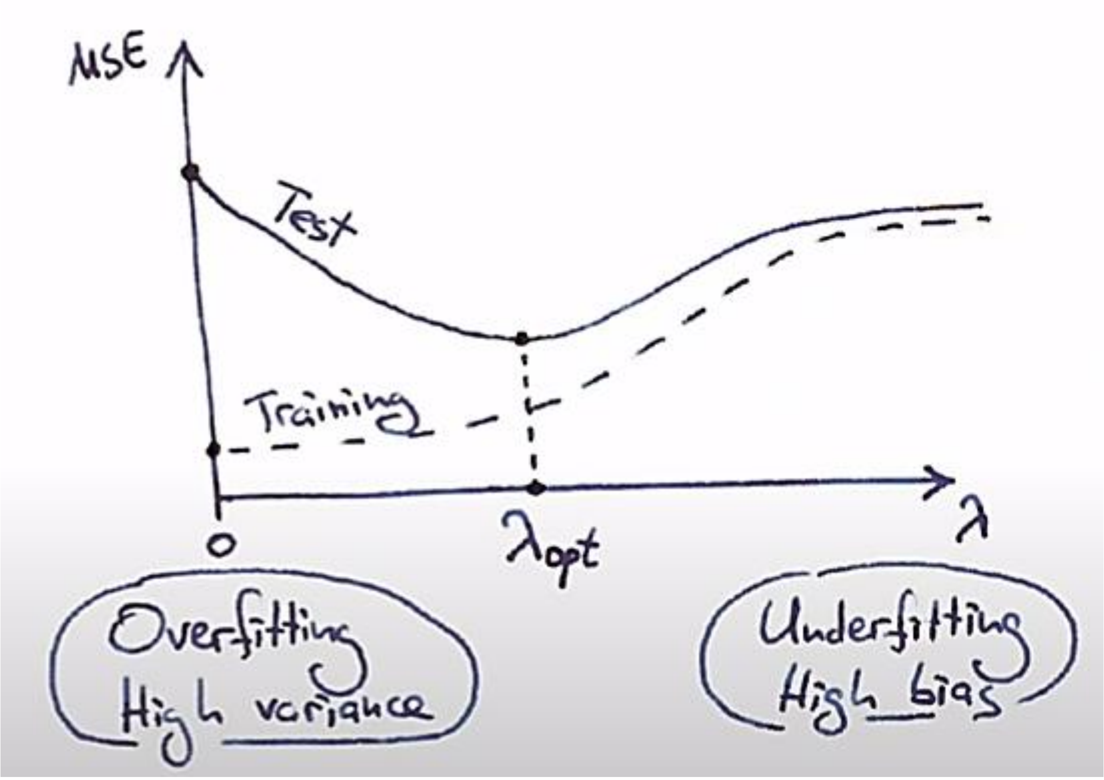

0이 되는 가중치가 많이 발생



# Regularization – *Selecting Lambda*

- Test Error가 가장 작게 되는 $\lambda$가 최적
  - 그러나 학습시에는 test set에 접근할 수 없으므로, validation set을 이용하여 최적의 $\lambda$를 선택함



# Regularization – *Do Not Penalize Bias!*

- **For Centered Dataset (when both *x* and *y* have zero mean)**
  - No problem even if we have zero bias (i.e., $\beta_0 = 0$).

$$\mathcal{L} = \frac{1}{n}\|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda\|\beta\|^2$$

- **For Non-centered Dataset (the general case)**
  - Penalizing bias often leads to bad performance.
  - Thus we need to ***exclude the bias ($\beta_0$) from the regularization term***:

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j X_{ij}\right)^2 + \lambda\sum_{j=1}^{p}\beta_j^2.$$

# Regularization – *Example: Linear Regression*

## ■ 선형 회귀에 적용

- 선형 회귀는 훈련집합 $\mathbb{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$, $\mathbb{Y} = \{y_1, y_2, \cdots, y_n\}$이 주어지면, 식 (5.24)를 풀어 $\mathbf{w} = (w_1, w_2, \cdots, w_d)^{\mathrm{T}}$를 구하는 문제. 이때 $\mathbf{x}_i = (x_{i1}, x_{i2}, \cdots, x_{id})^{\mathrm{T}}$

$$w_1 x_{i1} + w_2 x_{i2} \cdots + w_d x_{id} = \mathbf{x}_i^{\mathrm{T}} \mathbf{w} = y_i, \qquad i = 1,2,\cdots,n \tag{5.24}$$

- 식 (5.24)를 행렬식으로 바꿔 쓰면,

$$\mathbf{X}\mathbf{w} = \mathbf{y} \tag{5.25}$$

- 가중치 감쇠를 적용한 목적함수

$$J_{regularized}(\mathbf{w}) = \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2 + \lambda\|\mathbf{w}\|_2^2 = \frac{1}{2}(\mathbf{X}\mathbf{w} - \mathbf{y})^{\mathrm{T}}(\mathbf{X}\mathbf{w} - \mathbf{y}) + \lambda\|\mathbf{w}\|_2^2 \tag{5.27}$$

$\nabla_{\underline{x}}\, \underline{x}^T \underline{x} = 2\underline{x}$

$\nabla_x (X\underline{w} - \underline{y})^T (X\underline{w} - \underline{y})$

$= 2X^T (X\underline{w} - \underline{y})$

# Regularization – *Example: Linear Regression (cont'd)*

- 식 (5.27)을 미분하여 0으로 놓으면,

$$\frac{\partial J_{regularized}}{\partial \mathbf{w}} = \mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{w} - \mathbf{X}^{\mathrm{T}}\mathbf{y} + 2\lambda\mathbf{w} = \mathbf{0} \quad \Longrightarrow \quad (\mathbf{X}^{\mathrm{T}}\mathbf{X} + 2\lambda\mathbf{I})\mathbf{w} = \mathbf{X}^{\mathrm{T}}\mathbf{y} \qquad (5.28)$$

- 식 (5.28)을 정리하면,

$$\hat{\mathbf{w}} = (\mathbf{X}^{\mathrm{T}}\mathbf{X} + 2\lambda\mathbf{I})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y} \qquad (5.29)$$

→ Normal Eq. for Ridge Regression

  - 공분산 행렬 $\mathbf{X}^{\mathrm{T}}\mathbf{X}$의 대각 요소가 $2\lambda$만큼씩 증가 → 역행렬을 곱하므로 가중치를 축소하여 원점으로 당기는 효과 ([그림 5-21])

- 예측 단계에서는.

$$y = \mathbf{x}^{\mathrm{T}}\hat{\mathbf{w}} \qquad (5.30)$$

# Regularization – *Example: Linear Regression (cont'd)*

**예제 5-1** **리지 회귀**

훈련집합 $\mathbb{X} = \{\mathbf{x}_1 = \begin{pmatrix}1\\1\end{pmatrix}, \mathbf{x}_2 = \begin{pmatrix}2\\3\end{pmatrix}, \mathbf{x}_3 = \begin{pmatrix}3\\3\end{pmatrix}\}$, $\mathbb{Y} = \{y_1 = 3.0,\ y_2 = 7.0, y_3 = 8.8\}$이 주어졌다고 가정하자. 특징 벡터가 2차원이므로 $d=2$이고 샘플이 3개이므로 $n=3$이다. 훈련집합으로 설계행렬 $\mathbf{X}$와 레이블 행렬 $\mathbf{y}$를 다음과 같이 쓸 수 있다.

$$\mathbf{X} = \begin{pmatrix}1 & 1\\2 & 3\\3 & 3\end{pmatrix}, \qquad \mathbf{y} = \begin{pmatrix}3.0\\7.0\\8.8\end{pmatrix}$$

이 값들을 식 (5.29)에 대입하여 다음과 같이 $\hat{\mathbf{w}}$을 구할 수 있다. 이때 $\lambda = 0.25$라 가정하자.

$$\hat{\mathbf{w}} = \left(\begin{pmatrix}1 & 2 & 3\\1 & 3 & 3\end{pmatrix}\begin{pmatrix}1 & 1\\2 & 3\\3 & 3\end{pmatrix} + \begin{pmatrix}0.5 & 0\\0 & 0.5\end{pmatrix}\right)^{-1}\begin{pmatrix}1 & 2 & 3\\1 & 3 & 3\end{pmatrix}\begin{pmatrix}3.0\\7.0\\8.8\end{pmatrix} = \begin{pmatrix}1.4916\\1.3607\end{pmatrix}$$

따라서 하이퍼 평면은 $y = 1.4916x_1 + 1.3607x_2$이다. 새로운 샘플로 $\mathbf{x} = (5 \quad 4)^{\mathrm{T}}$가 입력되면 식 (5.30)을 이용하여 12.9009를 예측한다.

# Q & A