# Lecture 3: Machine learning basics
# - math

기계학습개론
박상효

## 학습목표

- 기계학습 이해를 위한 기본 수학 개념 이해
    - 주요 수학 표기법 정리
    - 기본 용어 정의
- 모델 훈련 관련 주요 용어 이해

## 핵심용어

- Feature space
    - Dimension
    - Tensor
- Training
    - Underfitting, Overfitting

# 기계 학습에서 수학의 역할

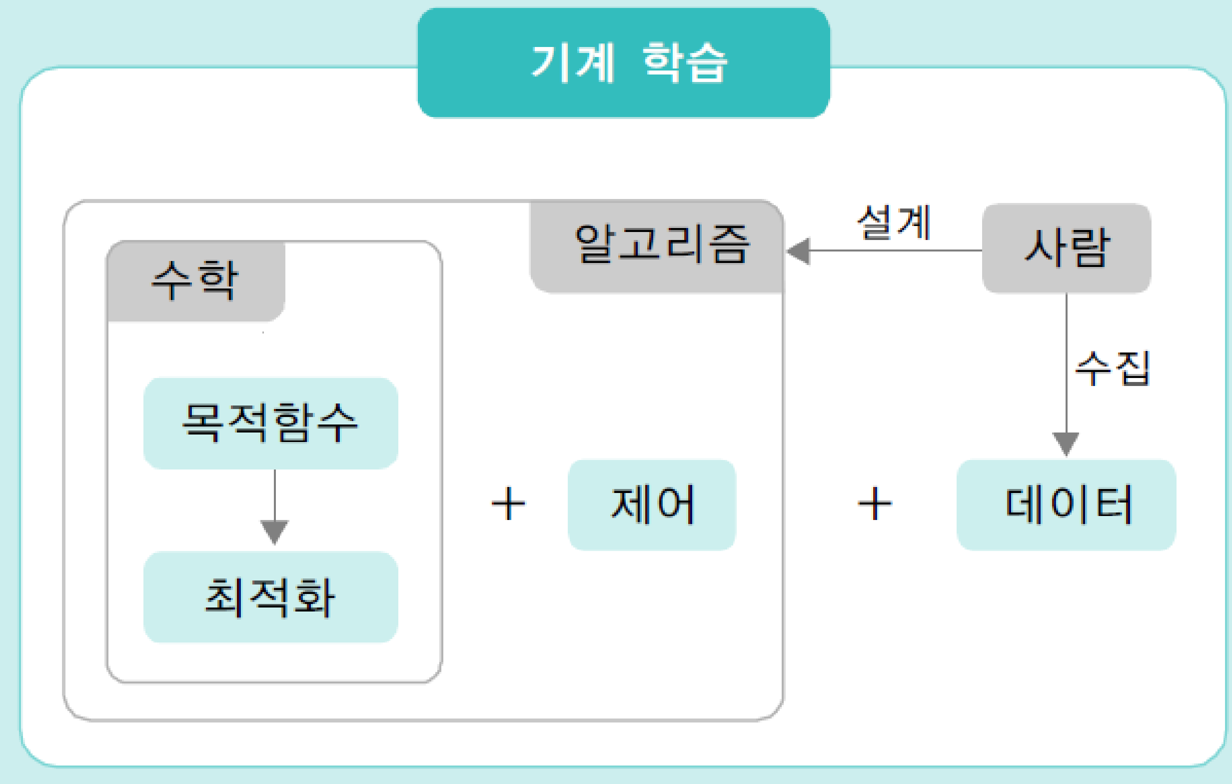
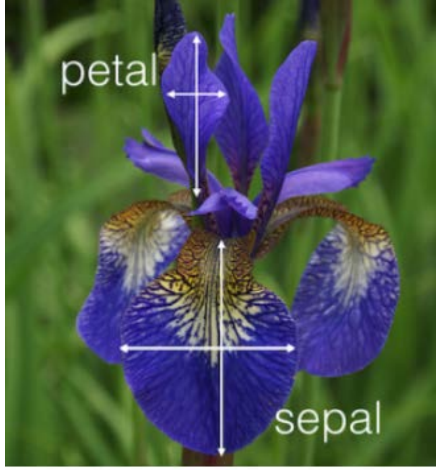



**그림 2-1 기계 학습의 개념도**

# 선형대수 : 벡터




- 벡터
  - 샘플을 특징 벡터로feature vector 표현
  - 예) Iris 데이터*에서 꽃받침의 길이, 꽃받침의 너비, 꽃잎의 길이, 꽃잎의 너비라는 4개의 특징이 각각 5.1, 3.5, 1.4, 0.2인 샘플 ➔

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 5.1 \\ 3.5 \\ 1.4 \\ 0.2 \end{pmatrix}$$

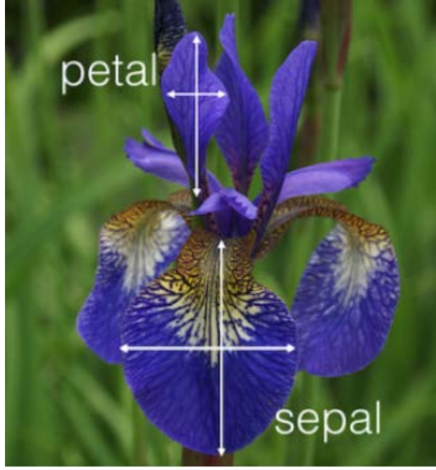
  - 여러 개의 특징 벡터를 첨자로 구분

➔

$$\mathbf{x}_1 = \begin{pmatrix} 5.1 \\ 3.5 \\ 1.4 \\ 0.2 \end{pmatrix}, \mathbf{x}_2 = \begin{pmatrix} 4.9 \\ 3.0 \\ 1.4 \\ 0.2 \end{pmatrix}, \mathbf{x}_3 = \begin{pmatrix} 4.7 \\ 3.2 \\ 1.3 \\ 0.2 \end{pmatrix}, \cdots, \mathbf{x}_{150} = \begin{pmatrix} 5.9 \\ 3.0 \\ 5.1 \\ 1.8 \end{pmatrix}$$

*Iris data set 출처 : http://archive.ics.uci.edu/ml/datasets/Iris

# 선형대수 : 행렬




- 행렬
  - 여러 개의 벡터를 담음
  - 예) Iris 데이터 * 에 있는 150개의 샘플을 설계 행렬 $\mathbf{X}$로 표현

$$\mathbf{X} = \begin{pmatrix} 5.1 & 3.5 & 1.4 & 0.2 \\ 4.9 & 3.0 & 1.4 & 0.2 \\ 4.7 & 3.2 & 1.3 & 0.2 \\ 4.6 & 3.1 & 1.5 & 0.2 \\ \vdots & \vdots & \vdots & \vdots \\ 6.2 & 3.4 & 5.4 & 2.3 \\ 5.9 & 3.0 & 5.1 & 1.8 \end{pmatrix} = \begin{pmatrix} x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} \\ x_{2,1} & x_{2,2} & x_{2,3} & x_{2,4} \\ x_{3,1} & x_{3,2} & x_{3,3} & x_{3,4} \\ x_{4,1} & x_{4,2} & x_{4,3} & x_{4,4} \\ \vdots & \vdots & \vdots & \vdots \\ x_{149,1} & x_{149,2} & x_{149,3} & x_{149,4} \\ x_{150,1} & x_{150,2} & x_{150,3} & x_{150,4} \end{pmatrix}$$

행row

열column

*Iris data set 출처 : http://archive.ics.uci.edu/ml/datasets/Iris

# 선형대수 : 행렬

- 전치행렬(Transpose)

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{pmatrix}, \quad \mathbf{A}^{\mathrm{T}} = \begin{pmatrix} a_{11} & a_{21} & \cdots & a_{n1} \\ a_{12} & a_{22} & \cdots & a_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1m} & a_{2m} & \dots & a_{nm} \end{pmatrix}$$

예를 들어, $\mathbf{A} = \begin{pmatrix} 3 & 4 & 1 \\ 0 & 5 & 2 \end{pmatrix}$라면 $\mathbf{A}^{\mathrm{T}} = \begin{pmatrix} 3 & 0 \\ 4 & 5 \\ 1 & 2 \end{pmatrix}$

# 선형대수 : 행렬

- 전치행렬(Transpose)
  - Iris dataset을 전치행렬로 표현하면,

$$\mathbf{X} = \begin{pmatrix} 5.1 & 3.5 & 1.4 & 0.2 \\ 4.9 & 3.0 & 1.4 & 0.2 \\ 4.7 & 3.2 & 1.3 & 0.2 \\ 4.6 & 3.1 & 1.5 & 0.2 \\ \vdots & \vdots & \vdots & \vdots \\ 6.2 & 3.4 & 5.4 & 2.3 \\ 5.9 & 3.0 & 5.1 & 1.8 \end{pmatrix} = \begin{pmatrix} x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} \\ x_{2,1} & x_{2,2} & x_{2,3} & x_{2,4} \\ x_{3,1} & x_{3,2} & x_{3,3} & x_{3,4} \\ x_{4,1} & x_{4,2} & x_{4,3} & x_{4,4} \\ \vdots & \vdots & \vdots & \vdots \\ x_{149,1} & x_{149,2} & x_{149,3} & x_{149,4} \\ x_{150,1} & x_{150,2} & x_{150,3} & x_{150,4} \end{pmatrix} \Rightarrow \mathbf{X} = \begin{pmatrix} \mathbf{x}_1^{\mathrm{T}} \\ \mathbf{x}_2^{\mathrm{T}} \\ \vdots \\ \mathbf{x}_{150}^{\mathrm{T}} \end{pmatrix}$$

벡터의 배열

1차원 특징 공간

2차원 특징 공간

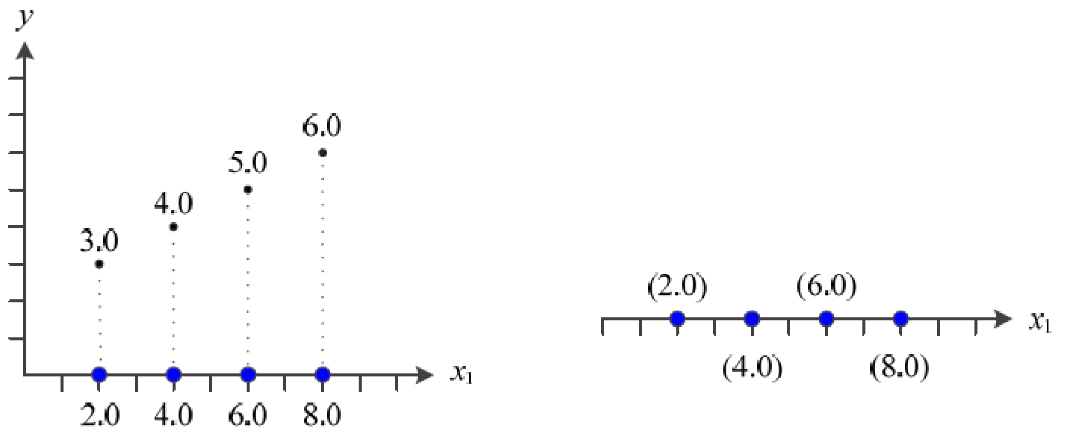
특징 벡터 표기

$\mathbf{x}=(x_1,x_2)^T$

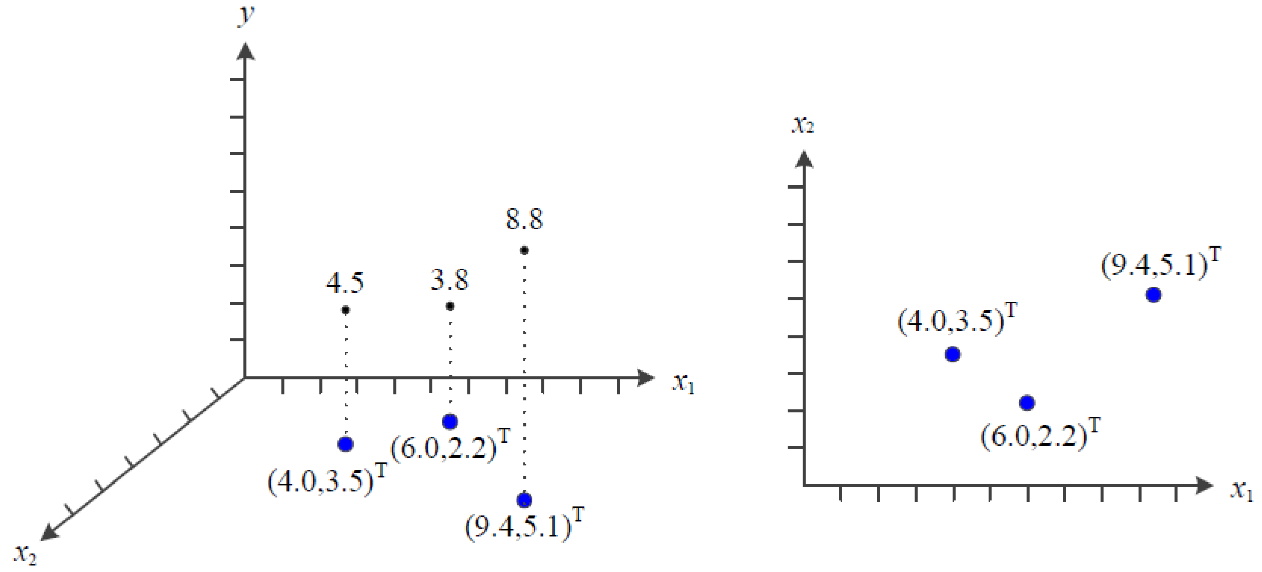
예시

$\mathbf{x}$=(몸무게,키)$^T$, $y$=장타율

$\mathbf{x}$=(체온,두통)$^T$, $y$=감기 여부



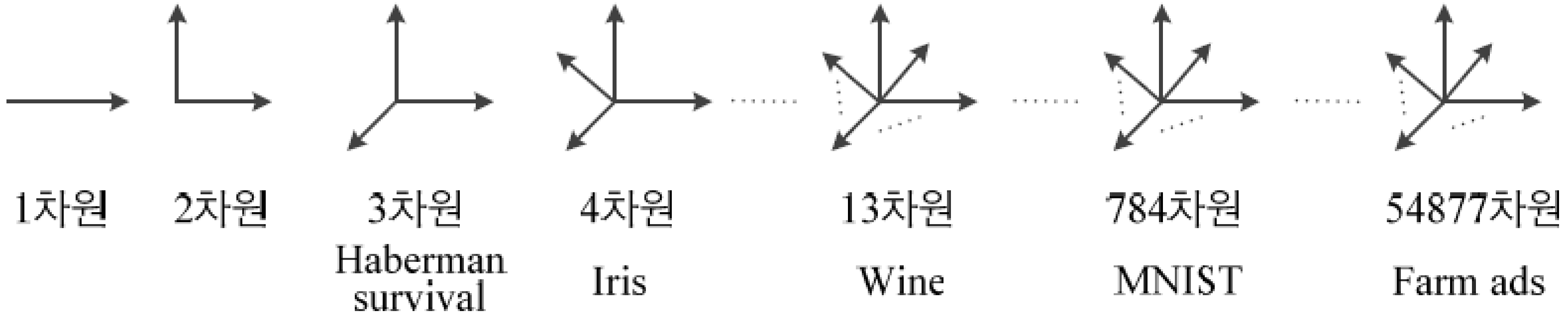

(a) 1차원 특징 공간(왼쪽: 특징과 목푯값을 축으로 표시, 오른쪽: 특징만 축으로 표시)




(b) 2차원 특징 공간(왼쪽: 특징 벡터와 목푯값을 축으로 표시, 오른쪽: 특징 벡터만 축으로 표시)

**그림 1-5 특징 공간과 데이터의 표현**

# 다차원 특징공간




Haberman survival: $\mathbf{x} = (\text{나이, 수술년도, 양성 림프샘 개수})^{\mathrm{T}}$

Iris: $\mathbf{x} = (\text{꽃받침 길이, 꽃받침 너비, 꽃잎 길이, 꽃잎 너비})^{\mathrm{T}}$

Wine: $\mathbf{x} = (\text{Alcohol, Malic acid, Ash, Alcalinity of ash, Magnesium, Total phenols, Flavanoids, Nonflavanoid phenols}$
$\text{Proanthocyanins, Color intensity, Hue, OD280 / OD315 of diluted wines, Proline})^{\mathrm{T}}$

MNIST: $\mathbf{x} = (\text{화소1, 화소2}, \cdots, \text{화소784})^{\mathrm{T}}$

Farm ads: $\mathbf{x} = (\text{단어1, 단어2}, \cdots, \text{단어54877})^{\mathrm{T}}$

**그림 1-6 다차원 특징 공간**

# 다차원 공간 → 행렬, 텐서

- 벡터의 배열 ➔ 행렬(matrix) 또는 2D 텐서(Tensor)임

```
>>> x = np.array([[5, 78, 2, 34, 0],
                  [6, 79, 3, 35, 1],
                  [7, 80, 4, 36, 2]])
>>> x.ndim
2
```

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^{\mathrm{T}} \\ \mathbf{x}_2^{\mathrm{T}} \\ \vdots \\ \mathbf{x}_{150}^{\mathrm{T}} \end{pmatrix}$$

벡터의 배열

- 텐서는 임의의 차원 개수를 가지는 행렬의 일반화된 모습
  (텐서에서는 차원(dimension)을 종종 축(axis)이라고 부름)

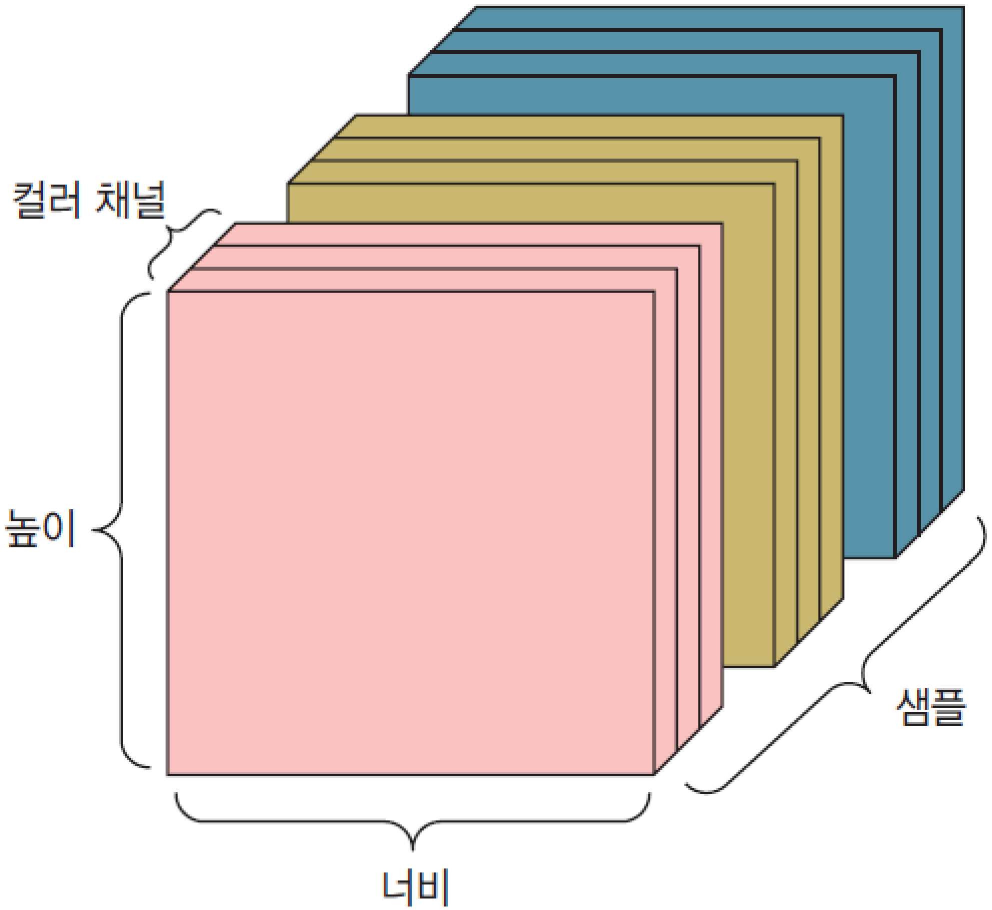
# 텐서(Tensor)

- 3D 텐서와 고차원 텐서
  - numpy에서 3D 텐서를 나타내면,

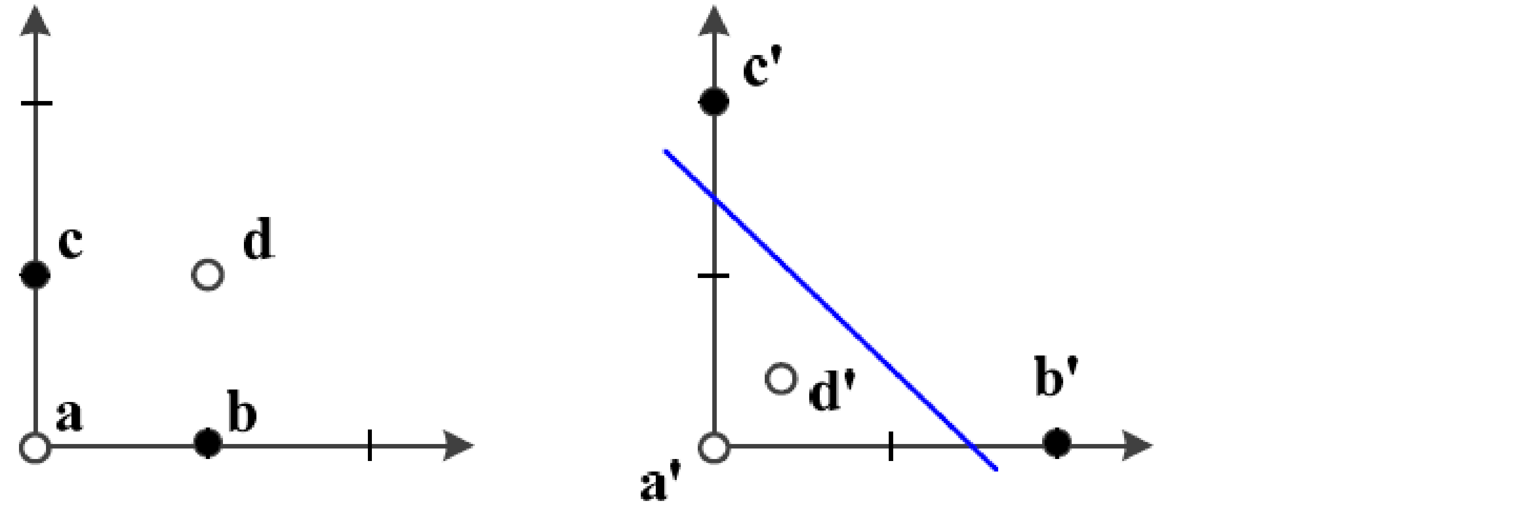
```
>>> x = np.array([[[5, 78, 2, 34, 0],
                   [6, 79, 3, 35, 1],
                   [7, 80, 4, 36, 2]],
                  [[5, 78, 2, 34, 0],
                   [6, 79, 3, 35, 1],
                   [7, 80, 4, 36, 2]],
                  [[5, 78, 2, 34, 0],
                   [6, 79, 3, 35, 1],
                   [7, 80, 4, 36, 2]]])
>>> x.ndim
3
```

# 4D Tensor

- 4D 이미지 데이터 텐서(채널 우선 표기)



# 선형 분리 불가능linearly non-separable한 경우




(a) 원래 특징 공간 (b) 분류에 더 유리하도록 변환된 새로운 특징 공간

**그림 1-7 특징 공간 변환**

# 확률과 통계

기|2.2

# 확률 기초

- 간단한 확률실험 장치
  - 주머니에서 번호를 뽑은 다음, 번호에 따라 해당 병에서 공을 뽑고 색을 관찰함
  - 번호를 $y$, 공의 색을 $x$라는 확률변수로 표현하면 정의역은 $y \in \{①,②,③\}$, $x \in \{$파랑, 하양$\}$

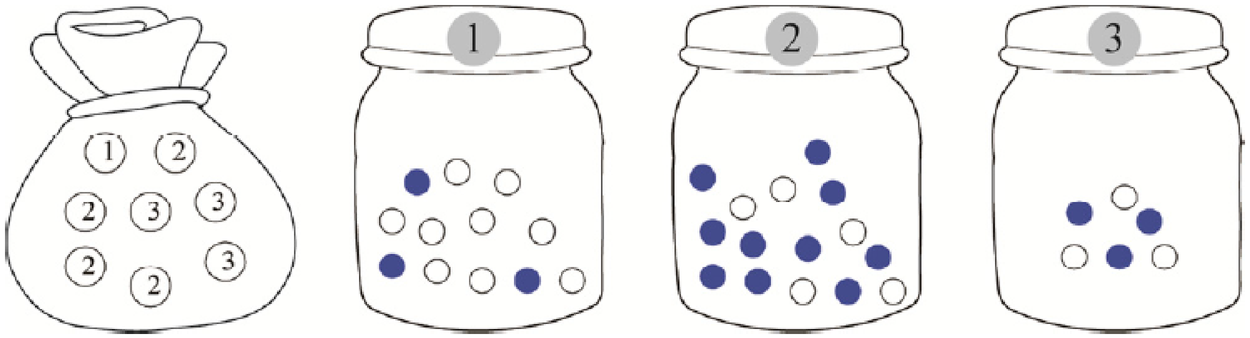



그림 2-15 확률 실험

# 확률 기초

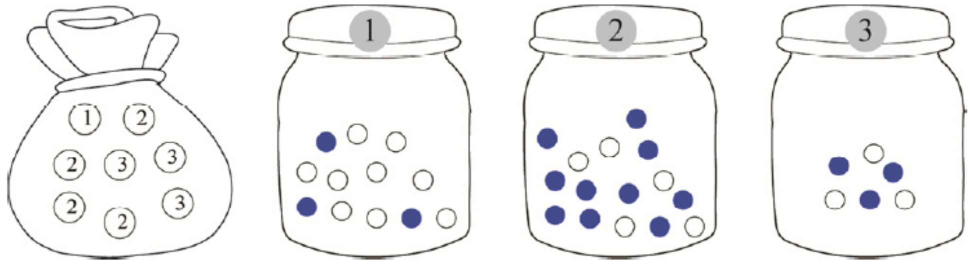


그림 2-15 확률 실험

- 카드는 ①번, 공은 하양일 확률은 $P(y=①,x=\text{하양})=P(①,\text{하양})$

➔결합확률(joint prob.)

$$P(y=①, x=\text{하양}) = P(x=\text{하양}|y=①)P(y=①) = \frac{9}{12}\frac{1}{8} = \frac{3}{32}$$

곱규칙

$$\text{곱 규칙: } P(y,x) = P(x|y)P(y) \tag{2.23}$$

# 베이즈 정리

- 베이즈 정리 (식 (2.26))

$$P(y,x) = P(x|y)P(y) = P(x,y) = P(y|x)P(x)$$

$$\longrightarrow \quad P(y|x) = \frac{P(x|y)P(y)}{P(x)} \tag{2.26}$$

- 다음 질문을 식 (2.27)로 쓸 수 있음

"하얀 공이 나왔다는 사실만 알고 어느 병에서 나왔는지 모르는데, 어느 병인지 추정하라."

$$\hat{y} = \underset{y}{\operatorname{argmax}} P(y|x) \tag{2.27}$$

# 베이즈 정리

- 베이즈 정리 (식 (2.26))
  - 베이즈 정리를 적용하면, $\hat{y} = \underset{y}{\operatorname{argmax}} P(y|x = \text{하양}) = \underset{y}{\operatorname{argmax}} \frac{P(x = \text{하양}|y)P(y)}{P(x = \text{하양})}$
  - 세 가지 경우에 대해 확률을 계산하면,

$$P(①|\text{하양}) = \frac{P(\text{하양}|①)P(①)}{P(\text{하양})} = \frac{\frac{9}{12}\frac{1}{8}}{\frac{43}{96}} = \frac{9}{43}$$

$$P(②|\text{하양}) = \frac{P(\text{하양}|②)P(②)}{P(\text{하양})} = \frac{\frac{5}{15}\frac{4}{8}}{\frac{43}{96}} = \frac{16}{43}$$

$$P(③|\text{하양}) = \frac{P(\text{하양}|③)P(③)}{P(\text{하양})} = \frac{\frac{3}{6}\frac{3}{8}}{\frac{43}{96}} = \frac{18}{43}$$

⟶ ③번 병일 확률이 가장 높음

# 베이즈 정리

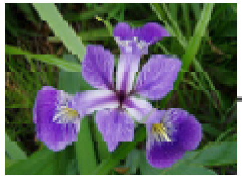
$$\overbrace{P(y|x)}^{\text{사후확률}} = \frac{\overbrace{P(x|y)}^{\text{우도}}\ \overbrace{P(y)}^{\text{사전확률}}}{P(x)}$$

- 용어
  - 사후확률(posterior prob.)
  - 우도(likelihood)
  - 사전확률(prior prob.)

# 베이즈 정리

- 기계 학습에 적용
  - 예) Iris 데이터 분류 문제
    - 특징 벡터 $\mathbf{x}$, 부류 $y \in$ {setosa, versicolor, virginica}
    - 분류 문제를 argmax로 표현하면 식 (2.29)

$$\hat{y} = \underset{y}{\operatorname{argmax}}\, P(y|\mathbf{x}) \tag{2.29}$$



$$\xrightarrow{\text{특징추출}} \mathbf{x} = (7.0, 3.2, 4.7, 1.4)^{\mathrm{T}} \xrightarrow[\text{추정}]{\text{사후확률}} \begin{array}{l} P(\text{setosa}|\mathbf{x}) = 0.18 \\ P(\text{versicolor}|\mathbf{x}) = 0.72 \\ P(\text{virginica}|\mathbf{x}) = 0.10 \end{array} \xrightarrow{\text{argmax}} \text{versicolor}$$

그림 2-16 붓꽃의 부류 예측 과정

# 베이즈 정리

- 기계 학습에 적용

    - 사후확률 $P(y|\mathbf{x})$를 직접 추정하는 일은 아주 단순한 경우를 빼고 불가능
    - 따라서 베이즈 정리를 이용하여 추정함
        - 사전확률은 식 (2.30)으로 추정
        - 우도는 밀도 추정 기법으로 추정(기6.4, GMM 참고)

$$\text{사전확률:} \quad P(y = c_i) = \frac{n_i}{n} \tag{2.30}$$

# 정보이론

- 메시지가 지닌 정보를 수량화할 수 있나?
  - "고비 사막에 눈이 왔다"와 "대관령에 눈이 왔다"라는 두 메시지 중 어느 것이 더 많은 정보를 가지나?
  - 정보이론의 기본 원리 → 확률이 작을수록 많은 정보

# 정보이론

- 자기 정보self information
  - 사건(메시지) $e_i$의 정보량 (단위: bit)

$$h(e_i) = -\log_2 P(e_i) \quad \text{또는} \quad h(e_i) = -\log_e P(e_i) \tag{2.44}$$

- 엔트로피
  - 확률변수 $x$의 불확실성을 나타내는 엔트로피

$$\text{이산 확률분포} \quad H(x) = -\sum_{i=1,k} P(e_i)\log_2 P(e_i) \quad \text{또는} \quad H(x) = -\sum_{i=1,k} P(e_i)\log_e P(e_i) \tag{2.45}$$

$$\text{연속 확률분포} \quad H(x) = -\int_{\mathbb{R}} P(x)\log_2 P(x) \quad \text{또는} \quad H(x) = -\int_{\mathbb{R}} P(x)\log_e P(x) \tag{2.46}$$

# 정보이론

- 자기 정보와 엔트로피 예제

**예제 2-8**

윷을 나타내는 확률변수를 *x*라 할 때 *x*의 엔트로피는 다음과 같다.

$$H(x) = -\left(\frac{4}{16}\log_2\frac{4}{16} + \frac{6}{16}\log_2\frac{6}{16} + \frac{4}{16}\log_2\frac{4}{16} + \frac{1}{16}\log_2\frac{1}{16} + \frac{1}{16}\log_2\frac{1}{16}\right) = 2.0306\text{비트}$$

주사위는 눈이 6개인데 모두 1/6이라는 균일한 확률을 가진다. 이 경우 엔트로피를 계산하면 다음과 같다.

$$H(x) = -\left(\frac{1}{6}\log_2\frac{1}{6} + \frac{1}{6}\log_2\frac{1}{6} + \frac{1}{6}\log_2\frac{1}{6} + \frac{1}{6}\log_2\frac{1}{6} + \frac{1}{6}\log_2\frac{1}{6} + \frac{1}{6}\log_2\frac{1}{6}\right) = 2.585\text{ 비트}$$

# 최적화

기|2.3

# 최적화

- 순수 수학 최적화와 기계 학습 최적화의 차이
  - 순수 수학의 최적화 예) $f(x_1, x_2) = -\left(\cos\left(x_1{}^2\right) + \sin\left(x_2{}^2\right)\right)^2$의 최저점을 찾아라.
  - 기계 학습의 최적화는 단지 훈련집합이 주어지고, 훈련집합에 따라 정해지는 목적함수의 최저점을 찾아야 함
    - 데이터로 미분하는 과정 필요 → 오류 역전파 알고리즘 (기3.4)
    - 주로 스토캐스틱 경사 하강법(SGD) 사용

# 최적화

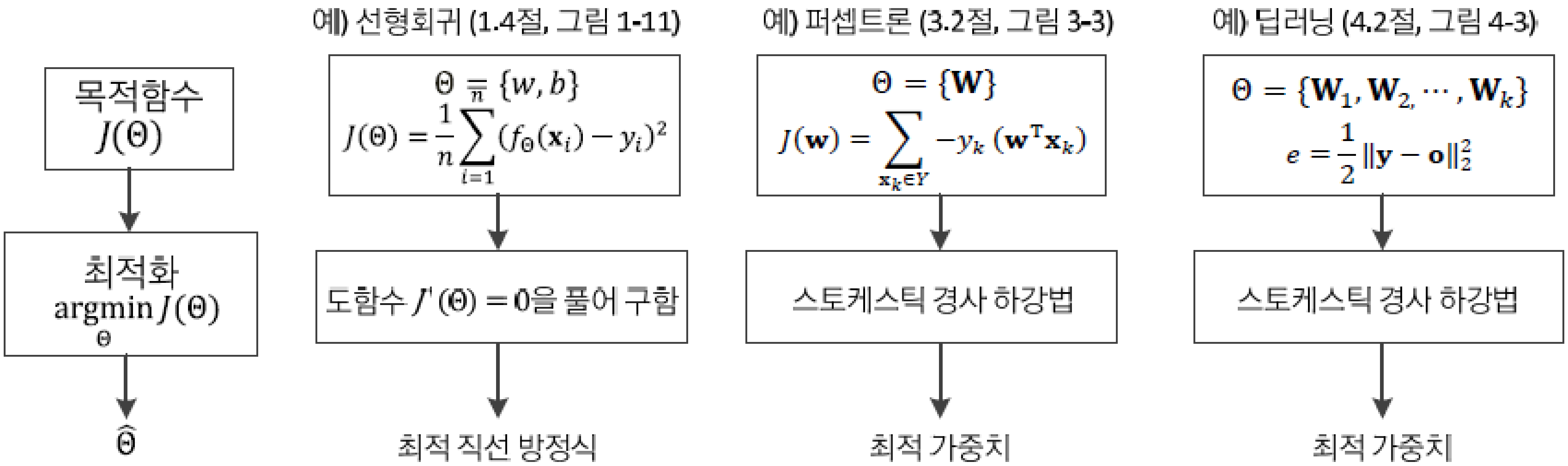
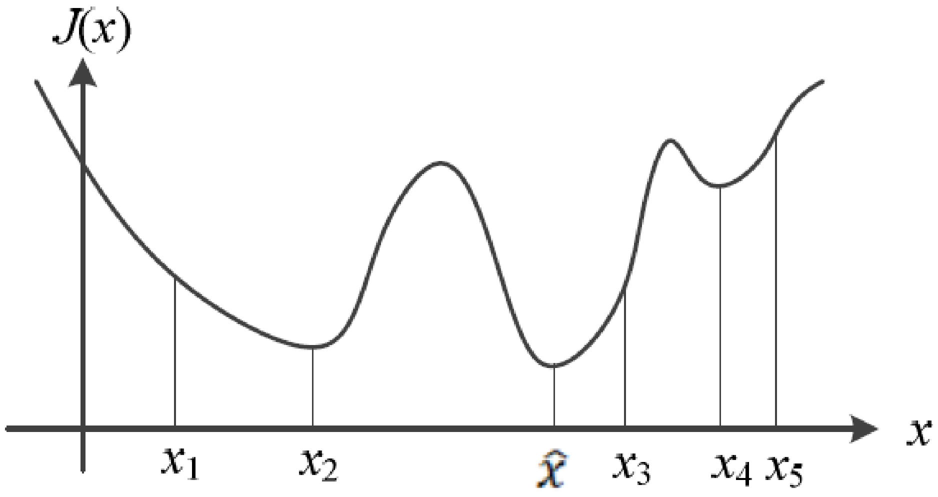



**그림 2-22 최적화를 이용한 기계 학습의 문제풀이 과정**



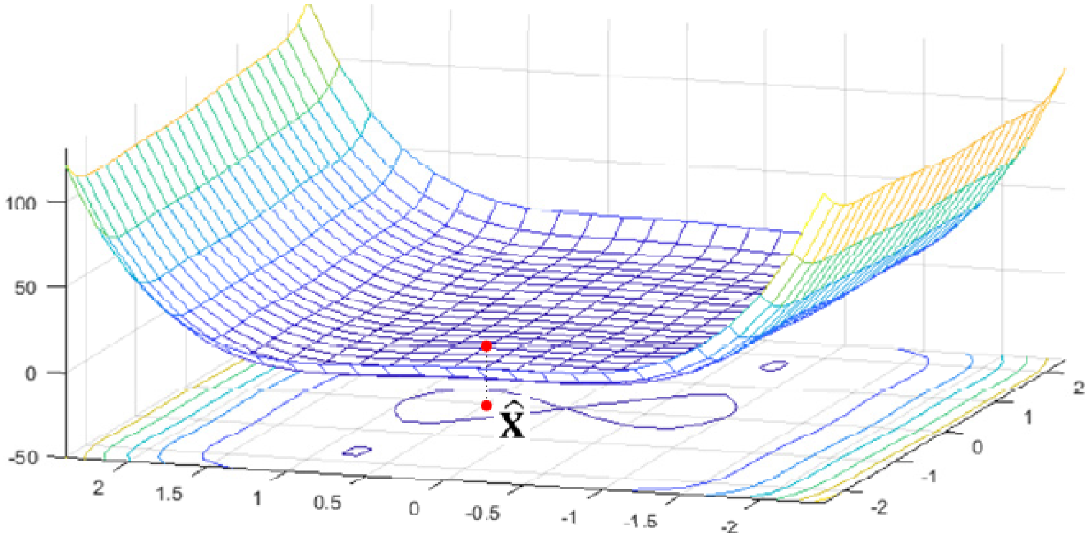




**그림 2-23 최적해 탐색**

- $\hat{x}$은 전역 최적해, $x_2$와 $x_4$는 지역 최적해

• 기계 학습이 해야 할 일을 식으로 정의하면,

$$J(\mathbf{\Theta})\text{를 최소로 하는 최적해 }\hat{\mathbf{\Theta}}\text{을 찾아라. 즉, }\hat{\mathbf{\Theta}} = \underset{\mathbf{\Theta}}{\operatorname{argmin}}\, J(\mathbf{\Theta}) \tag{2.50}$$

$J(\boldsymbol{\Theta})$를 최소로 하는 최적해 $\hat{\boldsymbol{\Theta}}$을 찾아라. 즉, $\hat{\boldsymbol{\Theta}} = \underset{\boldsymbol{\Theta}}{\operatorname{argmin}} J(\boldsymbol{\Theta})$ (2.50)

**알고리즘 2-3 기계 학습이 사용하는 전형적인 탐색 알고리즘(1장의 [알고리즘 1-1]과 같음)**

**입력 :** 훈련집합 $\mathbb{X}$와 $\mathbb{Y}$

**출력 :** 최적해 $\hat{\boldsymbol{\Theta}}$

```
난수를 생성하여 초기해 Θ을 설정한다.
repeat
    J(Θ)가 작아지는 방향 dΘ를 구한다.
    Θ = Θ + dΘ
until(멈춤 조건)
Θ̂ = Θ
```

# 미분

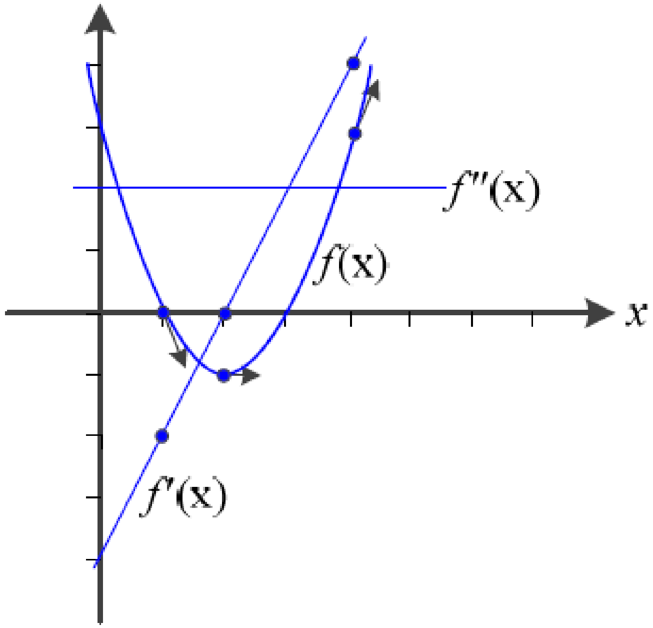
- 미분에 의한 최적화
  - 미분의 정의

$$f'(x) = \lim_{\Delta x \to 0} \frac{f(x+\Delta x) - f(x)}{\Delta x}, \qquad f''(x) = \lim_{\Delta x \to 0} \frac{f'(x+\Delta x) - f'(x)}{\Delta x} \tag{2.51}$$

  - 1차 도함수 $f'(x)$는 함수의 기울기, 즉 값이 커지는 방향을 지시함

# 미분

- 미분에 의한 최적화
  - 따라서 $-f'(x)$ 방향에 목적함수의 최저점이 존재
  - [알고리즘 2-3]에서 $d\mathbf{\Theta}$로 $-f'(x)$를 사용함← 경사 하강 알고리즘의 핵심 원리




$$y = f(x) = x^2 - 4x + 3$$

$$y' = f'(x) = 2x - 4$$

그림 2-24 간단한 미분 예제

# 미분

- 편미분
  - 변수가 여러 개인 함수의 미분
  - 미분값이 이루는 벡터를 그레이디언트라 부름
  - 여러 가지 표기: $\nabla f, \frac{\partial f}{\partial \mathbf{x}}, \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}\right)^{\mathrm{T}}$
  - 예)

$$
\left.\begin{aligned}
f(\mathbf{x}) = f(x_1, x_2) &= \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 + (-4 + 4x_2^2)x_2^2 \\
\nabla f = f'(\mathbf{x}) = \frac{\partial f}{\partial \mathbf{x}} &= \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}\right)^{\mathrm{T}} = (2x_1^5 - 8.4x_1^3 + 8x_1 + x_2, 16x_2^3 - 8x_2 + x_1)^{\mathrm{T}}
\end{aligned}\right\} \quad (2.52)
$$

# 경사하강알고리즘

- 식 (2.58)은 경사 하강법이 낮은 곳을 찾아가는 원리
  - $\mathbf{g} = d\boldsymbol{\Theta} = \frac{\partial J}{\partial \boldsymbol{\Theta}}$이고, $\rho$는 학습률

$$\boldsymbol{\Theta} = \boldsymbol{\Theta} - \rho\mathbf{g} \tag{2.58}$$

- 배치 경사 하강 알고리즘
  - 샘플의 그레이디언트를 평균한 후 한꺼번에 갱신

**알고리즘 2-4 배치 경사 하강 알고리즘(BGD)**

**입력:** 훈련집합 $\mathbb{X}$와 $\mathbb{Y}$, 학습률 $\rho$
**출력:** 최적해 $\widehat{\boldsymbol{\Theta}}$

```
난수를 생성하여 초기해 Θ를 설정한다.
repeat
    X에 있는 샘플의 그레이디언트 ∇_1, ∇_2, ⋯, ∇_n을 계산한다.
    ∇_total = (1/n) Σ_{i=1,n} ∇_i   // 그레이디언트 평균을 계산
    Θ = Θ − ρ∇_total
until(멈춤 조건)
Θ̂ = Θ
```

# 경사하강알고리즘

- 스토캐스틱 경사 하강SGD(stochastic gradient descent) 알고리즘
  - 한 샘플의 그레이디언트를 계산한 후 즉시 갱신
  - 라인 3~6을 한 번 반복하는 일을 한 세대라 부름

**알고리즘 2-5 스토케스틱 경사 하강 알고리즘(SGD)**

**입력 :** 훈련집합 $\mathbb{X}$와 $\mathbb{Y}$, 학습률 $\rho$
**출력 :** 최적해 $\hat{\mathbf{\Theta}}$

1 난수를 생성하여 초기해 $\mathbf{\Theta}$를 설정한다.
2 repeat
3 　$\mathbb{X}$의 샘플의 순서를 섞는다.
4 　for ($i$=1 to $n$)
5 　　　$i$번째 샘플에 대한 그레이디언트 $\mathbf{\nabla}_i$를 계산한다.
6 　　　$\mathbf{\Theta} = \mathbf{\Theta} - \rho\mathbf{\nabla}_i$
7 until(멈춤 조건)
8 $\hat{\mathbf{\Theta}} = \mathbf{\Theta}$

**알고리즘 2-3 기계 학습이 사용하는 전형적인 탐색 알고리즘(1장의 [알고리즘 1-1]과 같음)**

**입력 :** 훈련집합 $\mathbb{X}$와 $\mathbb{Y}$
**출력 :** 최적해 $\hat{\boldsymbol{\Theta}}$

```
난수를 생성하여 초기해 Θ을 설정한다.
repeat
    J(Θ)가 작아지는 방향 dΘ를 구한다.
    Θ = Θ + dΘ
until(멈춤 조건)
Θ̂ = Θ
```

**알고리즘 2-4 배치 경사 하강 알고리즘(BGD)**

**입력 :** 훈련집합 $\mathbb{X}$와 $\mathbb{Y}$, 학습률 $\rho$
**출력 :** 최적해 $\hat{\boldsymbol{\Theta}}$

```
난수를 생성하여 초기해 Θ를 설정한다.
repeat
    X에 있는 샘플의 그레이디언트 ∇_1, ∇_2, ⋯, ∇_n을 계산한다.
    ∇_total = (1/n) Σ_{i=1,n} ∇_i   // 그레이디언트 평균을 계산
    Θ = Θ − ρ∇_total
until(멈춤 조건)
Θ̂ = Θ
```

**알고리즘 2-5 스토케스틱 경사 하강 알고리즘(SGD)**

**입력 :** 훈련집합 $\mathbb{X}$와 $\mathbb{Y}$, 학습률 $\rho$
**출력 :** 최적해 $\hat{\boldsymbol{\Theta}}$

```
난수를 생성하여 초기해 Θ를 설정한다.
repeat
    X의 샘플의 순서를 섞는다.
    for (i=1 to n)
        i번째 샘플에 대한 그레이디언트 ∇_i를 계산한다.
        Θ = Θ − ρ∇_i
until(멈춤 조건)
Θ̂ = Θ
```

# 모델 훈련

# Underfitting

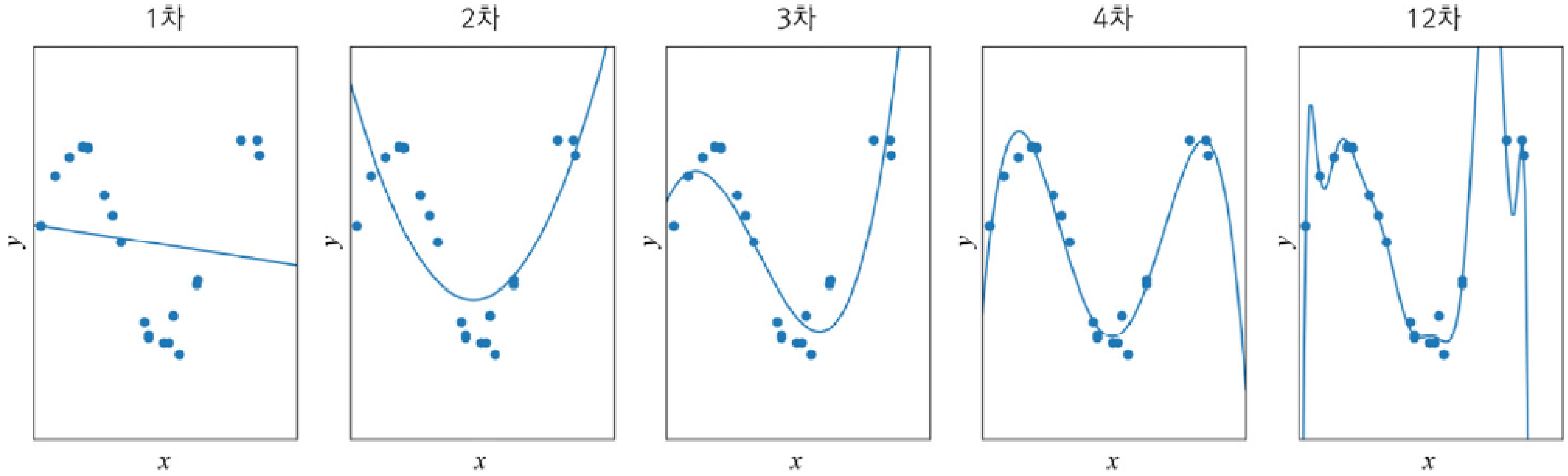



**그림 1-13** 과소적합과 과잉적합 현상

# Overfitting

- 12차 다항식 곡선을 채택한다면 **훈련집합**에 대해 거의 완벽하게 근사화함
- 하지만 '새로운' 데이터를 예측한다면 큰 문제 발생
  - $x_0$에서 빨간 막대 근방을 예측해야 하지만 빨간 점을 예측

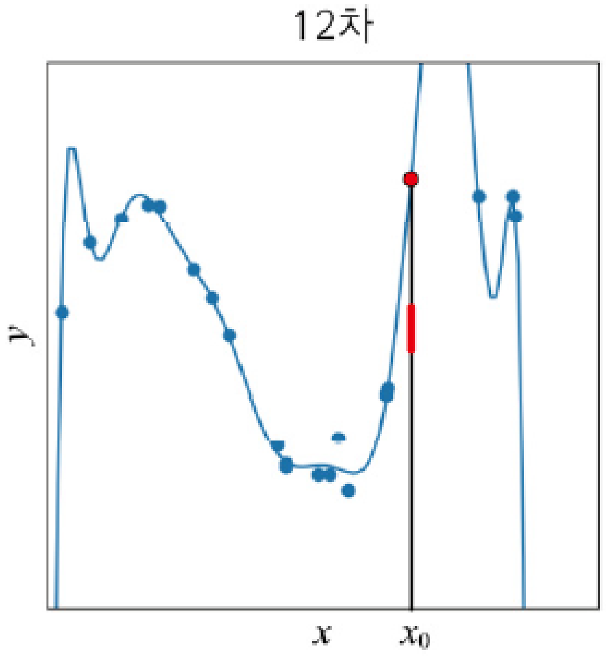



그림 1-14 과잉적합되었을 때 부정확한 예측 현상

# 검증집합

- 훈련집합과 테스트집합과 다른 별도의 검증집합을 가진 상황

**알고리즘 1-2 검증집합을 이용한 모델 선택**

**입력 :** 모델집합 Ω, 훈련집합, 검증집합, 테스트집합
**출력 :** 최적 모델과 성능

```
for (Ω에 있는 각각의 모델)
        모델을 훈련집합으로 학습시킨다.
        검증집합으로 학습된 모델의 성능을 측정한다.   // 검증 성능 측정
가장 높은 성능을 보인 모델을 선택한다.
테스트집합으로 선택된 모델의 성능을 측정한다.
```

# 모델 선택의 한계

- SVM, Decision tree, GMM, PCA, MLP, CNN, RNN, GAN, etc.
  - 현실에서는 모델의 종류가 아주 많음

- 현실에서는 경험으로 큰 틀 선택한 후 [경험적 접근방법]
  - 모델 선택 알고리즘으로 세부 모델 선택하는 전략 사용
  - 예) CNN을 사용하기로 정한 후, 은닉층 개수, 활성함수, 모멘툼 계수 등을 정하는데 모델 선택 알고리즘을 적용함

"To some extent, we are always trying to fit a square peg(the data generating process) into a round hole(our model family). 어느 정도 우리가 하는 일은 항상 둥근 홈(우리가 선택한 모델)에 네모 막대기(데이터 생성 과정)를 끼워 넣는 것이라고 말할 수 있다[Goodfellow2016(222쪽)]." *출처 : Deep Learning,

# Summary

- 수학 용어
  - 행렬
  - 베이즈정리(Bayes’ theorem)
  - 엔트로피(entropy)
  - 편미분(partial derivative)
- 특징공간
  - 차원, 텐서
- 훈련
  - 과소적합
  - 과대적합
  - 모델 선택과 훈련의 한계

# In the next lecture…

- 데이터 처리방식
  - Numpy, pandas,
  - 데이터 중요성
  - 시각화를 통한 분석
- 과제 #1
  - Colab 활용
  - 영상 분석

**기** : 기계학습, 오일석, 2017
**핸** : 핸즈온머신러닝, 2/E, 2020 (번역)
**모** : 모두의 딥러닝, 2/E, 2020
**케** : 케라스 창시자에게 배우는..., 2018 (번역)
**머** : 머신러닝 도감 그림으로..., 2019 (번역)
**파** : Python machine learning, 2/E, 2019 (번역) → "머신러닝 교과서 with 파이썬, ..." 2019

# 참고자료

- 기1.2, 1.5
- 기2
- 케1

# 퀴즈1(높은배점)

응시기한 9.8 ~ 10