기계학습 수학

이현석 교수

2023-2

PREVIEW

■ 기계 학습에서 수학의 역할

- 수학은 목적함수를 정의하고, 목적함수가 최저가 되는 점을 찾아주는 최적화 이론 제공
- 최적화 이론에 규제, 모멘텀, 학습률, 멈춤조건과 같은 제어를 추가하여 알고리즘 구축
- 사람은 알고리즘을 설계하고 데이터를 수집함

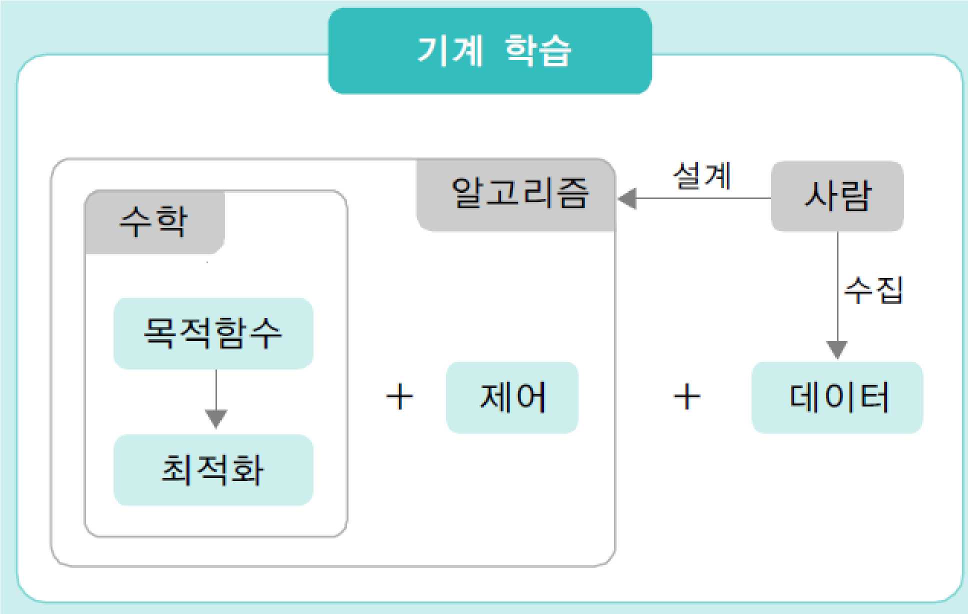


그림 2-1 기계 학습의 개념도

PREVIEW

- 선형대수: 이 분야의 개념을 이용하면 학습 모델의 매개변수집합, 데이터, 선형연산의 결합 등을 행렬 또는 텐서로 간결하게 표현할 수 있다. 데이터를 분석하여 유용한 정보를 알아내거나 특징 공간을 변환하는 등의 과업을 수행하는 데 핵심 역할을 한다.
- 확률과 통계: 데이터에 포함된 불확실성을 표현하고 처리하는 데 활용한다. 베이즈 이론과 최대 우도 기법을 이용하여 확률 추론을 수행한다.
- 최적화: 목적함수를 최소화하는 최적해를 찾는 데 활용하며, 주로 미분을 활용한 방법을 사용한다. 수학자들이 개발한 최적화 방법을 기계 학습이라는 도메인에 어떻게 효율적으로 적용할지가 주요 관심사이다.

벡터와 행렬

벡터

- 샘플을 특징 벡터로feature vector 표현
- 예) Iris 데이터에서 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

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

벡터와 행렬

행렬

- 여러 개의 벡터를 담음
- 훈련집합을 담은 행렬을 설계행렬이라 부름
- 예) Iris 데이터에 있는 150개의 샘플을 설계 행렬 $\mathbf{X}$로 표현

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

벡터와 행렬

- 행렬 $\mathbf{A}$의 전치행렬 $\mathbf{A}^{\mathrm{T}}$

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

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

- Iris의 설계 행렬을 전치행렬 표기에 따라 표현하면,

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

벡터와 행렬

- 행렬을 이용하면 수학을 간결하게 표현할 수 있음
  - 예) 다항식의 행렬 표현

$$
\begin{aligned}
f(\mathbf{x}) &= f(x_1, x_2, x_3) \\
&= 2x_1x_1 - 4x_1x_2 + 3x_1x_3 + x_2x_1 + 2x_2x_2 + 6x_2x_3 - 2x_3x_1 + 3x_3x_2 + 2x_3x_3 + 2x_1 + 3x_2 - 4x_3 + 5 \\
&= \begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix} \begin{pmatrix} 2 & -4 & 3 \\ 1 & 2 & 6 \\ -2 & 3 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 2 & 3 & -4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + 5 \\
&= \mathbf{x}^{\mathrm{T}}\mathbf{A}\mathbf{x} + \mathbf{b}^{\mathrm{T}}\mathbf{x} + c
\end{aligned}
$$

- 특수한 행렬들

정사각행렬 $\begin{pmatrix} 2 & 0 & 1 \\ 1 & 21 & 5 \\ 4 & 5 & 12 \end{pmatrix}$, 대각행렬 $\begin{pmatrix} 50 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 8 \end{pmatrix}$,

단위행렬 $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$, 대칭행렬 $\begin{pmatrix} 1 & 2 & 11 \\ 2 & 21 & 5 \\ 11 & 5 & 1 \end{pmatrix}$

벡터와 행렬

행렬 연산

- 행렬 곱셈 $\mathbf{C} = \mathbf{AB}$, 이때 $c_{ij} = \sum_{k=1,s} a_{ik} b_{kj}$ (2.1)

2*3 행렬 $\mathbf{A} = \begin{pmatrix} 3 & 4 & 1 \\ 0 & 5 & 2 \end{pmatrix}$와 3*3행렬 $\mathbf{B} = \begin{pmatrix} 2 & 0 & 1 \\ 1 & 0 & 5 \\ 4 & 5 & 1 \end{pmatrix}$을 곱하면 2*3 행렬 $\mathbf{C} = \mathbf{AB} = \begin{pmatrix} 14 & 5 & 24 \\ 13 & 10 & 27 \end{pmatrix}$

  - 교환법칙 성립하지 않음: $\mathbf{AB} \neq \mathbf{BA}$
  - 분배법칙과 결합법칙 성립: $\mathbf{A}(\mathbf{B} + \mathbf{C}) = \mathbf{AB} + \mathbf{AC}$이고 $\mathbf{A}(\mathbf{BC}) = (\mathbf{AB})\mathbf{C}$

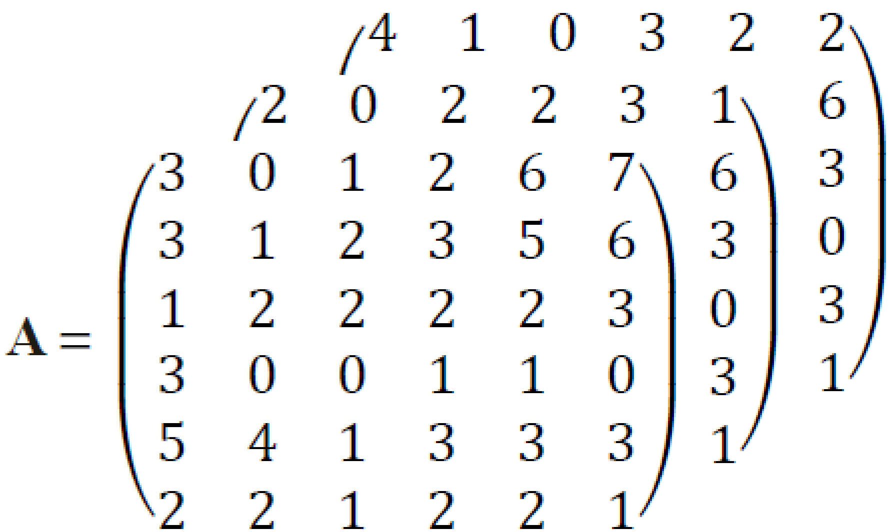
- 벡터의 내적 $\mathbf{a} \cdot \mathbf{b} = \mathbf{a}^{\mathrm{T}}\mathbf{b} = \sum_{k=1,d} a_k b_k$ (2.2)

$\mathbf{x}_1 = \begin{pmatrix} 5.1 \\ 3.5 \\ 1.4 \\ 0.2 \end{pmatrix}$와 $\mathbf{x}_2 = \begin{pmatrix} 4.9 \\ 3.0 \\ 1.4 \\ 0.2 \end{pmatrix}$의 내적 $\mathbf{x}_1 \cdot \mathbf{x}_2$는 37.49

벡터와 행렬

텐서

- 3차원 이상의 구조를 가진 숫자 배열
- 예) 3차원 구조의 RGB 컬러 영상



놈과 유사도

벡터와 행렬의 크기를 놈으로 측정

- 벡터의 p차 놈

$$p\text{차 놈: } \|\mathbf{x}\|_p = \left(\sum_{i=1,d} |x_i|^p\right)^{\frac{1}{p}} \quad (2.3)$$

$$\text{최대 놈: } \|\mathbf{x}\|_\infty = \max(|x_1|, |x_2|, \cdots, |x_d|) \quad (2.4)$$

  - 예) $\mathbf{x} = (3 \ \ -4 \ \ 1)$ 일 때, 2차 놈은 $\|\mathbf{x}\|_2 = (3^2 + (-4)^2 + 1^2)^{1/2} = 5.099$

- 행렬의 프로베니우스 놈 $\|\mathbf{A}\|_F = \left(\sum_{i=1,n} \sum_{j=1,m} a_{ij}^2\right)^{\frac{1}{2}} \quad (2.6)$

예를 들어, $\left\|\begin{pmatrix} 2 & 1 \\ 6 & 4 \end{pmatrix}\right\|_F = \sqrt{2^2 + 1^2 + 6^2 + 4^2} = 7.550$

놈과 유사도

유사도와 거리

- 벡터를 기하학적으로 해석

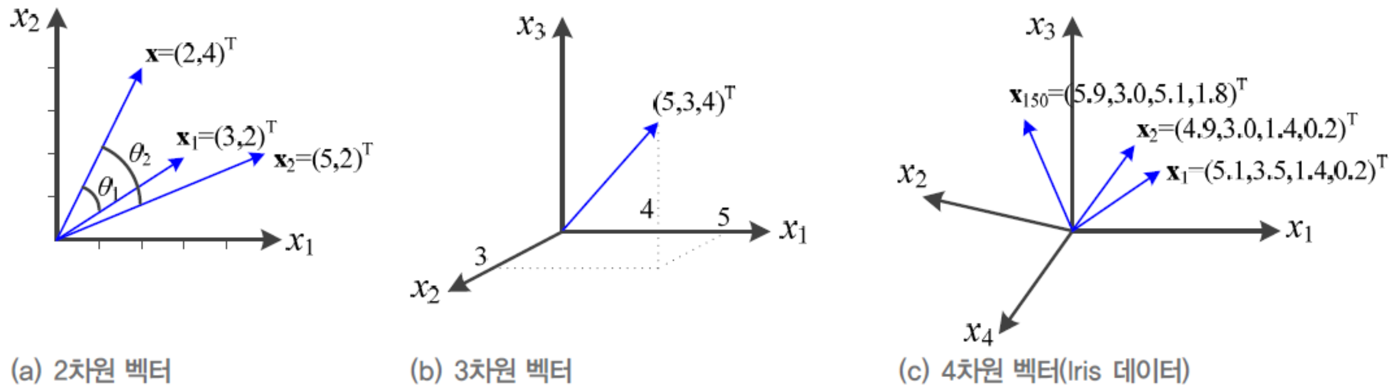


그림 2-2 벡터를 기하학적으로 해석

- 코사인 유사도

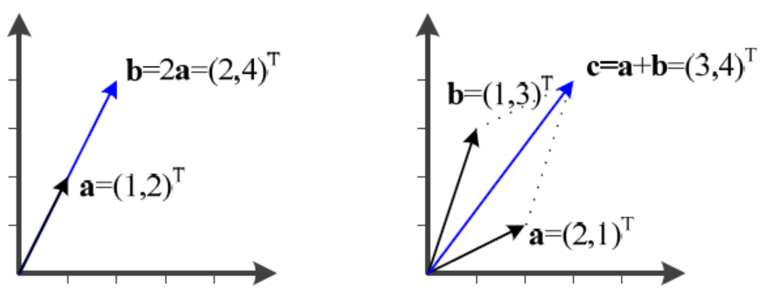
$$cosine\_similarity(\mathbf{a},\mathbf{b}) = \frac{\mathbf{a}}{\|\mathbf{a}\|}\cdot\frac{\mathbf{b}}{\|\mathbf{b}\|} = cos(\theta) \tag{2.7}$$

선형결합과 벡터공간

- 벡터
  - 공간상의 한 점으로 화살표 끝이 벡터의 좌표에 해당
- 선형결합이 만드는 벡터공간
  - 기저벡터 **a**와 **b**의 선형결합

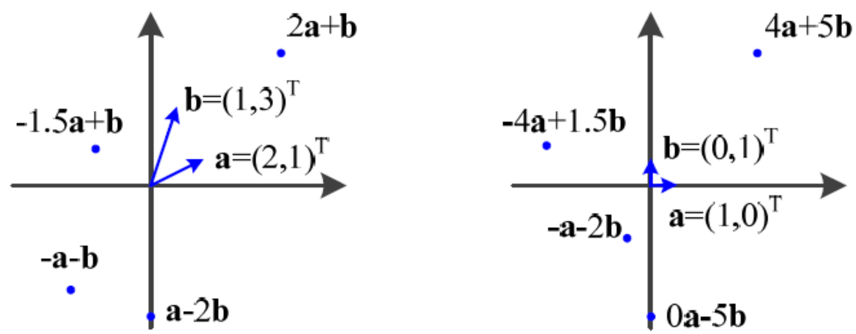
$$\mathbf{c} = \alpha_1 \mathbf{a} + \alpha_2 \mathbf{b} \qquad (2.12)$$

  - 선형결합으로 만들어지는 공간을 벡터공간이라 부름



(a) 벡터에 스칼라 곱 (b) 두 벡터의 덧셈

그림 2-6 벡터의 연산

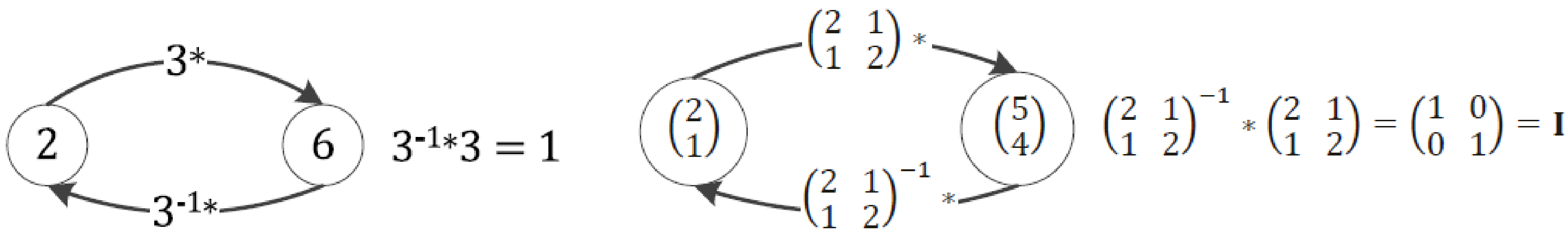


(a) 기저 벡터와 벡터공간 (b) 정규직교 기저 벡터

그림 2-7 벡터공간

역행렬

역행렬의 원리



(a) 역수의 원리 (b) 역행렬의 원리

그림 2-9 역행렬

- 정사각행렬 $\mathbf{A}$의 역행렬 $\mathbf{A}^{-1}$

$$\mathbf{A}^{-1}\mathbf{A} = \mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$$

- 예를 들어, $\begin{pmatrix}2 & 1\\6 & 4\end{pmatrix}$의 역행렬은 $\begin{pmatrix}2 & -0.5\\-3 & 1\end{pmatrix}$

역행렬

정리

정리 2-1 다음 성질은 서로 필요충분조건이다.

- $\mathbf{A}$는 역행렬을 가진다. 즉, 특이행렬이 아니다.
- $\mathbf{A}$는 최대계수를 가진다.
- $\mathbf{A}$의 모든 행이 선형독립이다.
- $\mathbf{A}$의 모든 열이 선형독립이다.
- $\mathbf{A}$의 행렬식은 0이 아니다.
- $\mathbf{A}^{\mathrm{T}}\mathbf{A}$는 양의 정부호positive definite 대칭 행렬이다.
- $\mathbf{A}$의 고윳값은 모두 0이 아니다.

행렬 분해

분해란?

- 정수 3717은 특성이 보이지 않지만, 3*3*7*59로 소인수 분해를 하면 특성이 보이듯이, 행렬도 분해하면 여러모로 유용함

고윳값과 고유 벡터

- 고유 벡터 $\mathbf{v}$와 고윳값 λ

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$$

- 예를 들어, $\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}\begin{pmatrix} 1 \\ 1 \end{pmatrix} = 3\begin{pmatrix} 1 \\ 1 \end{pmatrix}$이고 $\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}\begin{pmatrix} 1 \\ -1 \end{pmatrix} = 1\begin{pmatrix} 1 \\ -1 \end{pmatrix}$이므로, $\lambda_1 = 3, \lambda_2 = 1$이고 $\mathbf{v}_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$, $\mathbf{v}_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$

행렬 분해

■ 고윳값과 고유 벡터의 기하학적 해석

예제 2-5

[그림 2-12]의 반지름이 1인 원 위에 있는 4개의 벡터 $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4$가 $\mathbf{A} = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$에 의해 어떻게 변환되는지 살펴보자. 변환 후의 벡터를 각각 $\mathbf{x}'_1, \mathbf{x}'_2, \mathbf{x}'_3, \mathbf{x}'_4$로 표기한다.

$$\mathbf{x}'_1 = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}\begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix} = \begin{pmatrix} 3/\sqrt{2} \\ 3/\sqrt{2} \end{pmatrix}$$
$$\mathbf{x}'_2 = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$
$$\mathbf{x}'_3 = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}\begin{pmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{pmatrix} = \begin{pmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{pmatrix}$$
$$\mathbf{x}'_4 = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}\begin{pmatrix} 0 \\ -1 \end{pmatrix} = \begin{pmatrix} -1 \\ -2 \end{pmatrix}$$

눈 여겨 볼 점은 $\mathbf{A}$의 고유 벡터 $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$, $\begin{pmatrix} 1 \\ -1 \end{pmatrix}$과 방향이 같은 $\mathbf{x}_1$과 $\mathbf{x}_3$이다. 이들은 변환 때문에 길이가 달라지더라도 방향은 그대로 유지한다. 식 (2.20)을 충실히 따르고 있다. 이때 길이의 변화는 고윳값 λ에 따른다. 즉, $\mathbf{x}_1$은 3배만큼, $\mathbf{x}_3$은 1배만큼 길이가 변한다. 나머지 $\mathbf{x}_2$와 $\mathbf{x}_4$는 길이와 방향이 모두 변한다. 파란 원 위에 있는 모든 점을 변환하면 빨간색의 타원이 된다. 파란 원 위에 존재하는 무수히 많은 점(벡터) 중에 방향이 바뀌지 않는 것은 고유 벡터에 해당하는 $\mathbf{x}_1$과 $\mathbf{x}_3$뿐이다.

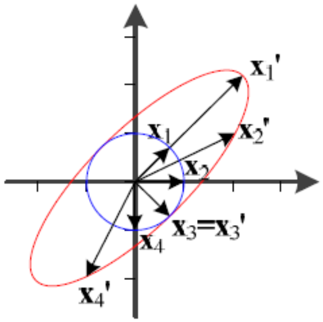


그림 2-12 고유 벡터의 공간 변환

행렬 분해

고윳값 분해eigen value decomposition

$$\mathbf{A} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{-1} \qquad (2.21)$$

- $\mathbf{Q}$는 $\mathbf{A}$의 고유 벡터를 열에 배치한 행렬이고 $\boldsymbol{\Lambda}$는 고윳값을 대각선에 배치한 대각행렬
- 예를 들어, $\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 3 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{pmatrix}$
- 고윳값 분해는 정사각행렬에만 적용 가능한데, 기계 학습에서는 정사각행렬이 아닌 경우의 분해도 필요하므로 고윳값 분해는 한계를 가짐

행렬 분해

- $n*m$ 행렬 **A**의 특잇값 분해SVD(singular value decomposition)

$$\mathbf{A} = \mathbf{U\Sigma V}^{\mathrm{T}} \tag{2.22}$$

  - 왼쪽 특이행렬 **U**는 $\mathbf{AA}^{\mathrm{T}}$의 고유 벡터를 열에 배치한 $n*n$ 행렬
  - 오른쪽 특이행렬 **V**는 $\mathbf{A}^{\mathrm{T}}\mathbf{A}$의 고유 벡터를 열에 배치한 $m*m$ 행렬
  - $\mathbf{\Sigma}$는 $\mathbf{AA}^{\mathrm{T}}$의 고윳값의 제곱근을 대각선에 배치한 $n*m$ 대각행렬

예를 들어, **A**를 4*3 행렬이라고 했을 때 다음과 같이 특잇값 분해가 된다.

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 2 \\ 3 & 1 & 0 \\ 1 & 0 & 2 \end{pmatrix} = \begin{pmatrix} -0.1914 & -0.2412 & 0.1195 & -0.9439 \\ -0.5144 & 0.6990 & -0.4781 & -0.1348 \\ -0.6946 & -0.6226 & -0.2390 & 0.2697 \\ -0.4651 & 0.2560 & 0.8367 & 0.1348 \end{pmatrix}$$

$$\begin{pmatrix} 3.7837 & 0 & 0 \\ 0 & 2.7719 & 0 \\ 0 & 0 & 1.4142 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} -0.7242 & -0.4555 & -0.5177 \\ -0.6685 & 0.2797 & 0.6891 \\ 0.1690 & -0.8452 & 0.5071 \end{pmatrix}$$

확률과 통계

- 기계 학습이 처리할 데이터는 불확실한 세상에서 발생하므로, 불확실성을 다루는 확률과 통계를 잘 활용해야 함

확률 기초

- 확률변수random variable
  - 예) 윷

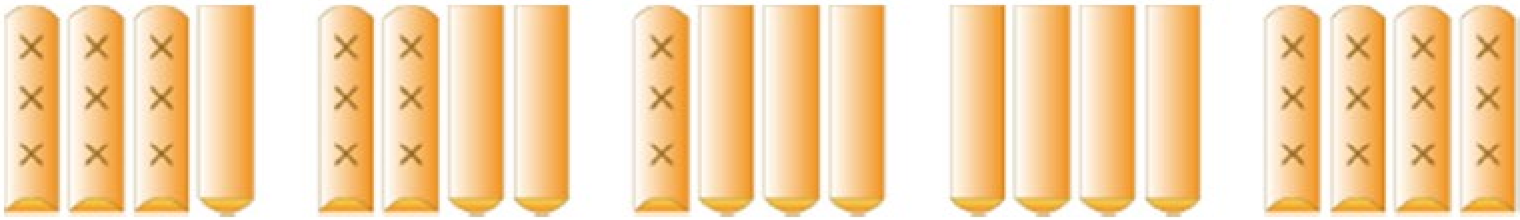


그림 2-13 윷을 던졌을 때 나올 수 있는 다섯 가지 경우(왼쪽부터 도, 개, 걸, 윷, 모)

  - 다섯 가지 경우 중 한 값을 갖는 확률변수 x
  - x의 정의역은 {도, 개, 걸, 윷, 모}

확률 기초

확률분포

$$P(x=\text{도})=\frac{4}{16}, P(x=\text{개})=\frac{6}{16}, P(x=\text{걸})=\frac{4}{16}, P(x=\text{윷})=\frac{1}{16}, P(x=\text{모})=\frac{1}{16}$$

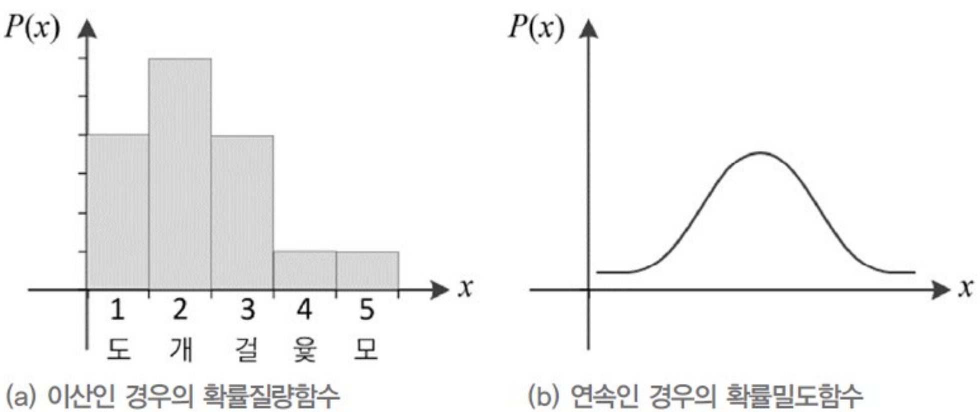


그림 2-14 확률분포

확률벡터random vector

- 예) Iris에서 확률벡터 $\mathbf{x}$는 4차원 $\mathbf{x}=(x_1,x_2,x_3,x_4)^{\mathrm{T}}=$ (꽃받침 길이, 꽃받침 너비$_1$, 꽃잎 길이, 꽃잎 너비$)^{\mathrm{T}}$

확률 기초

간단한 확률실험 장치

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

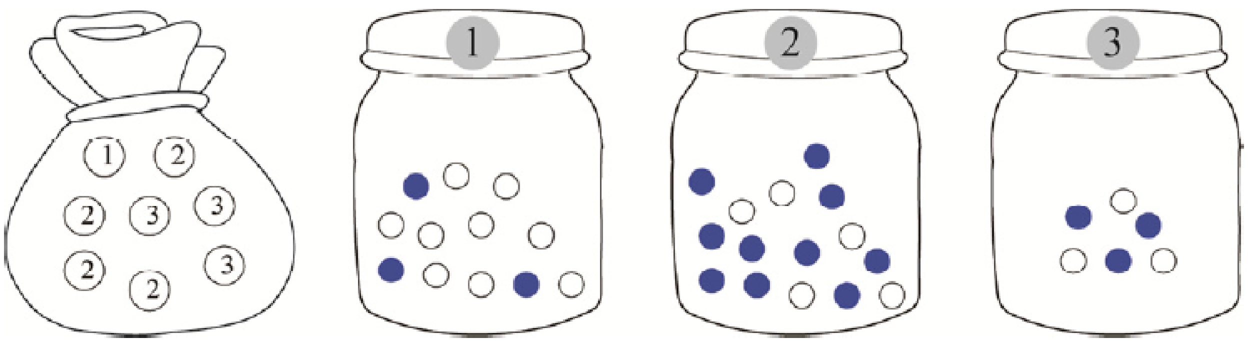


그림 2-15 확률 실험

확률 기초

곱 규칙과 합 규칙

- ①번 카드를 뽑을 확률은 $P(y=①)=P(①)=1/8$
- 카드는 ①번, 공은 하양일 확률은 $P(y=①,x=\text{하양})=P(①,\text{하양})$ ← 결합확률

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

- 곱 규칙 $P(y,x)=P(x|y)P(y)$ (2.23)
- 하얀 공이 뽑힐 확률

$$\begin{aligned}P(\text{하양})&=P(\text{하양}|①)P(①)+P(\text{하양}|②)P(②)+P(\text{하양}|③)P(③)\\&=\frac{9}{12}\frac{1}{8}+\frac{5}{15}\frac{4}{8}+\frac{3}{6}\frac{3}{8}=\frac{43}{96}\end{aligned}$$

- 합 규칙 $P(x)=\sum_{y}P(y,x)=\sum_{y}P(x|y)P(y)$ (2.24)

베이즈 정리와 기계 학습

- 베이즈 정리 (식 (2.26))

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

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

  - 다음 질문을 식 (2.27)로 쓸 수 있음

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

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

베이즈 정리와 기계 학습

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,

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

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

③번 병일 확률이 가장 높음

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

베이즈 정리의 해석

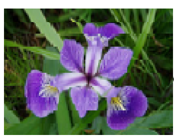
$$\overbrace{P(y|x)}^{\text{사후확률}} = \frac{\overbrace{P(x|y)}^{\text{우도}}\ \overbrace{P(y)}^{\text{사전확률}}}{P(x)}$$

베이즈 정리와 기계 학습

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)

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



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

그림 2-16 붓꽃의 부류 예측 과정

- 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 6.4절의 밀도 추정 기법으로 추정

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

최대 우도

- 매개변수 Θ를 모르는 상황에서 매개변수를 추정하는 문제

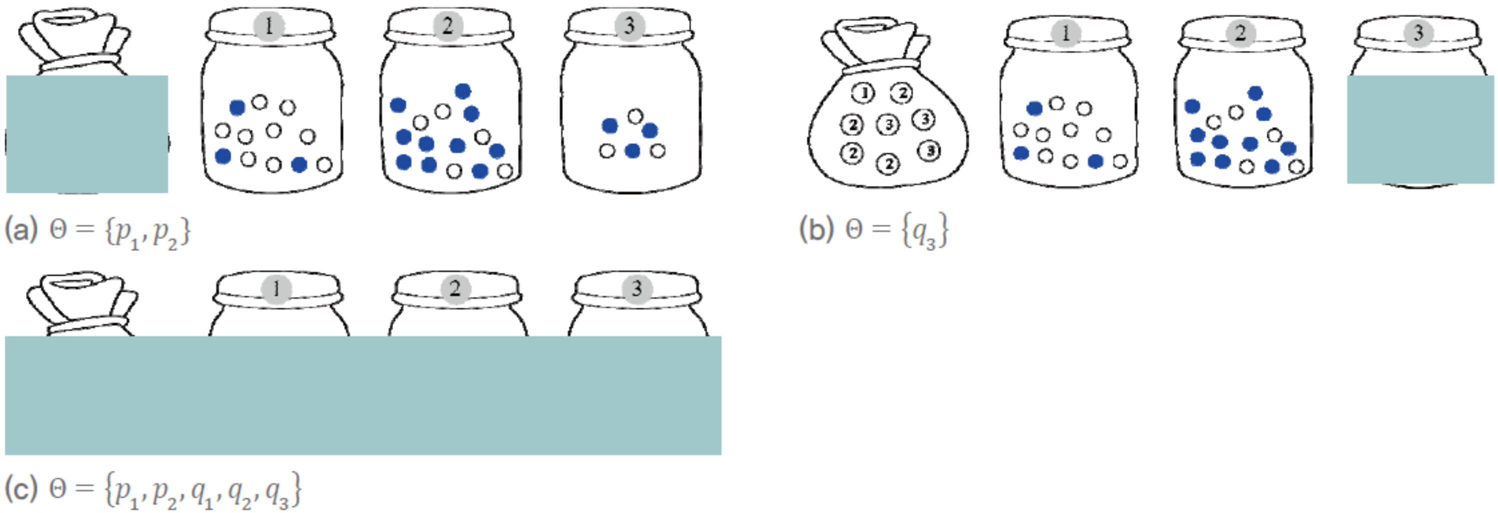


그림 2-17 매개변수가 감추어진 여러 가지 상황

- 예) [그림 2-17(b)] 상황

데이터집합 $\mathbb{X}$={●○○●○●○○●●○○}

"데이터 $\mathbb{X}$가 주어졌을 때, $\mathbb{X}$를 발생시켰을 가능성을 최대로 하는 매개변수 $\Theta = \{q_3\}$의 값을 찾아라."

최대 우도

최대 우도법

- [그림 2-17(b)] 문제를 수식으로 쓰면,

$$\hat{q}_3 = \underset{q_3}{\operatorname{argmax}} P(\mathbb{X}|q_3) \tag{2.31}$$

- 일반화 하면,

$$\text{최대 우도 추정:} \quad \hat{\Theta} = \underset{\Theta}{\operatorname{argmax}} P(\mathbb{X}|\Theta) \tag{2.32}$$

- 수치 문제를 피하기 위해 로그 표현으로 바꾸면,

$$\text{최대 로그우도 추정:} \quad \hat{\Theta} = \underset{\Theta}{\operatorname{argmax}} \log P(\mathbb{X}|\Theta) = \underset{\Theta}{\operatorname{argmax}} \sum_{i=1}^{n} \log P(\mathbf{x}_i|\Theta) \tag{2.34}$$

평균과 분산

- 데이터의 요약 정보로서 평균과 분산

$$\left.\begin{aligned} \text{평균} \quad \mu &= \frac{1}{n}\sum_{i=1}^{n} x_i \\ \text{분산} \quad \sigma^2 &= \frac{1}{n}\sum_{i=1}^{n} (x_i - \mu)^2 \end{aligned}\right\} \qquad (2.36)$$

- 평균 벡터와 공분산 행렬

$$\boldsymbol{\mu} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i \qquad (2.37)$$

$$\boldsymbol{\Sigma} = \frac{1}{n}\sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^{\mathrm{T}} \qquad (2.39)$$

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1d} \\ \sigma_{21} & \sigma_{22} & & \sigma_{2d} \\ \vdots & & \ddots & \vdots \\ \sigma_{d1} & \sigma_{d2} & \cdots & \sigma_{dd} \end{pmatrix} = \begin{pmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1d} \\ \sigma_{21} & \sigma_2^2 & & \sigma_{2d} \\ \vdots & & \ddots & \vdots \\ \sigma_{d1} & \sigma_{d2} & \cdots & \sigma_d^2 \end{pmatrix}$$

평균과 분산

평균 벡터와 공분산 행렬 예제

예제 2-7

Iris 데이터베이스의 샘플 중 8개만 가지고 공분산 행렬을 계산하자.

$$\mathbb{X} = \left\{\mathbf{x}_1 = \begin{pmatrix}5.1\\3.5\\1.4\\0.2\end{pmatrix}, \mathbf{x}_2 = \begin{pmatrix}4.9\\3.0\\1.4\\0.2\end{pmatrix}, \mathbf{x}_3 = \begin{pmatrix}4.7\\3.2\\1.3\\0.2\end{pmatrix}, \mathbf{x}_4 = \begin{pmatrix}4.6\\3.1\\1.5\\0.2\end{pmatrix}, \mathbf{x}_5 = \begin{pmatrix}5.0\\3.6\\1.4\\0.2\end{pmatrix}, \mathbf{x}_6 = \begin{pmatrix}5.4\\3.9\\1.7\\0.4\end{pmatrix}, \mathbf{x}_7 = \begin{pmatrix}4.6\\3.4\\1.4\\0.3\end{pmatrix}, \mathbf{x}_8 = \begin{pmatrix}5.0\\3.4\\1.5\\0.2\end{pmatrix}\right\}$$

먼저 평균벡터를 구하면 $\boldsymbol{\mu} = (4.9125,\ 3.3875,\ 1.45,\ 0.2375)^{\mathrm{T}}$이다. 첫 번째 샘플 $\mathbf{x}_1$을 식 (2.39)에 적용하면 다음과 같다.

$$(\mathbf{x}_1 - \boldsymbol{\mu})(\mathbf{x}_1 - \boldsymbol{\mu})^{\mathrm{T}} = \begin{pmatrix}0.1875\\0.1125\\-0.05\\-0.0375\end{pmatrix}(0.1875 \quad 0.1125 \quad -0.05 \quad -0.0375)$$

$$= \begin{pmatrix}0.0325 & 0.0211 & -0.0094 & -0.0070\\0.0211 & 0.0127 & -0.0056 & -0.0042\\-0.0094 & -0.0056 & 0.0025 & 0.0019\\-0.0070 & -0.0042 & 0.0019 & 0.0014\end{pmatrix}$$

나머지 7개 샘플도 같은 계산을 한 다음, 결과를 모두 더하고 8로 나누면 다음과 같은 공분산 행렬을 얻는다.

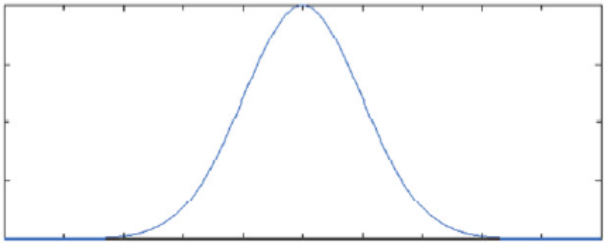
$$\boldsymbol{\Sigma} = \begin{pmatrix}0.0661 & 0.0527 & 0.0181 & 0.0083\\0.0527 & 0.0736 & 0.0181 & 0.0130\\0.0181 & 0.0181 & 0.0125 & 0.0056\\0.0083 & 0.0130 & 0.0056 & 0.0048\end{pmatrix}$$

유용한 확률분포

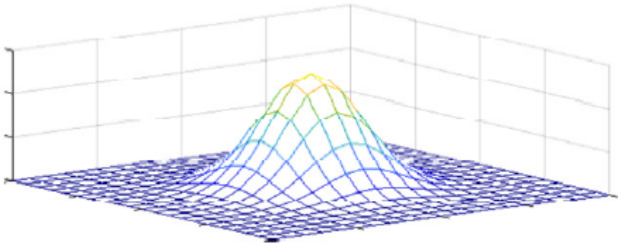
가우시안 분포

- 평균 μ와 분산 σ^2으로 정의

$$N(x;\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)$$



(a) 1차원



(b) 2차원

그림 2-19 가우시안 분포

- 다차원 가우시안 분포: 평균벡터 $\boldsymbol{\mu}$와 공분산행렬 $\boldsymbol{\Sigma}$로 정의

$$N(\mathbf{x};\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{\sqrt{|\boldsymbol{\Sigma}|}\sqrt{(2\pi)^d}}\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

유용한 확률분포

베르누이 분포

- 성공(x=1) 확률 p이고 실패(x=0) 확률이 1-p인 분포

$$Ber(x;p) = p^x(1-p)^{1-x} = \begin{cases} p, & x = 1\text{일 때} \\ 1-p, & x = 0\text{일 때} \end{cases}$$

이항 분포

- 성공 확률이 p인 베르누이 실험을 m번 수행할 때 성공할 횟수의 확률분포

$$B(x;m,p) = C_m^x p^x (1-p)^{m-x} = \frac{m!}{x!\,(m-x)!} p^x (1-p)^{m-x}$$

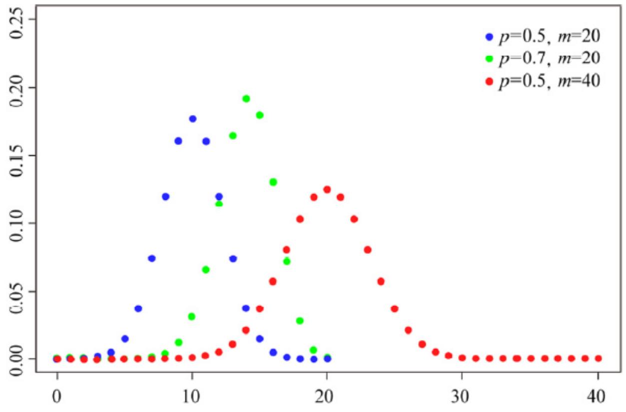


그림 2-20 이항 분포

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
- 자기 정보self information
  - 사건(메시지) e_i의 정보량 (단위: 비트 또는 나츠)

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

- 엔트로피
  - 확률변수 x의 불확실성을 나타내는 엔트로피

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

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

자기 정보와 엔트로피 예제

예제 2-8

윷을 나타내는 확률변수를 x라 할 때 x의 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.

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

- 주사위가 윷보다 엔트로피가 높은 이유는?

정보이론

- 교차 엔트로피cross entropy
  - 두 확률분포 P와 Q 사이의 교차 엔트로피

$$H(P,Q) = -\sum_{x} P(x)\log_2 Q(x) = -\sum_{i=1,k} P(e_i)\log_2 Q(e_i) \tag{2.47}$$

  - 식을 전개하면,

$$\begin{aligned} H(P,Q) &= -\sum_{x} P(x)\log_2 Q(x) \\ &= -\sum_{x} P(x)\log_2 P(x) + \sum_{x} P(x)\log_2 P(x) - \sum_{x} P(x)\log_2 Q(x) \\ &= H(P) + \underbrace{\sum_{x} P(x)\log_2 \frac{P(x)}{Q(x)}}_{\text{KL 다이버전스}} \end{aligned}$$

정보이론

KL 다이버전스

- 식 (2.48)은 P와 Q 사이의 KL 다이버전스
- 두 확률분포 사이의 거리를 계산할 때 주로 사용

$$KL(P \parallel Q) = \sum_{x} P(x)\log_2 \frac{P(x)}{Q(x)} \tag{2.48}$$

교차 엔트로피와 KL 다이버전스의 관계

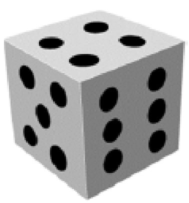
$$\begin{aligned} P\text{와 } Q\text{의 교차 엔트로피 } H(P,Q) &= H(P) + \sum_{x} P(x)\log_2 \frac{P(x)}{Q(x)} \\ &= P\text{의 엔트로피} + P\text{와 } Q \text{ 간의 } KL \text{ 다이버전스} \end{aligned} \tag{2.49}$$

정보이론

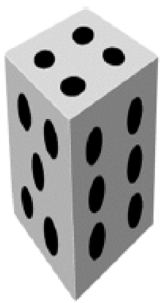
예제 2-9

[그림 2-21]과 같이 정상적인 주사위와 찌그러진 주사위가 있는데, 정상적인 주사위의 확률분포는 P, 찌그러진 주사위의 확률분포는 Q를 따르며, P와 Q가 다음과 같이 분포한다고 가정하자.

$$P(1)=\frac{1}{6},\ P(2)=\frac{1}{6},\ P(3)=\frac{1}{6},\ P(4)=\frac{1}{6},\ P(5)=\frac{1}{6},\ P(6)=\frac{1}{6}$$
$$Q(1)=\frac{3}{12},\ Q(2)=\frac{1}{12},\ Q(3)=\frac{1}{12},\ Q(4)=\frac{1}{12},\ Q(5)=\frac{3}{12},\ Q(6)=\frac{3}{12}$$



(a) 정상 주사위



(b) 찌그러진 주사위

그림 2-21 확률분포가 다른 두 주사위

확률분포 P와 Q 사이의 교차 엔트로피와 KL 다이버전스는 다음과 같다.

$$H(P,Q)=-\left(\frac{1}{6}\log_2\frac{3}{12}+\frac{1}{6}\log_2\frac{1}{12}+\frac{1}{6}\log_2\frac{1}{12}+\frac{1}{6}\log_2\frac{1}{12}+\frac{1}{6}\log_2\frac{3}{12}+\frac{1}{6}\log_2\frac{3}{12}\right)=2.7925$$
$$KL(P\parallel Q)=\frac{1}{6}\log_2\frac{2}{3}+\frac{1}{6}\log_2 2+\frac{1}{6}\log_2 2+\frac{1}{6}\log_2 2+\frac{1}{6}\log_2\frac{2}{3}+\frac{1}{6}\log_2\frac{2}{3}=0.2075$$

[예제 2-8]에서 P의 엔트로피 $H(P)$는 2.585이었다. 따라서 식 (2.49)가 성립함을 알 수 있다.

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 (3.4절)
  - 주로 SGD(스토캐스틱 경사 하강법) 사용

최적화 이론

- 최적의 선택을 찾는 것
  - 무엇이 “최적”인가?
  - 모든 가능한 $\mathbf{x} \in \Omega$ 중 함수 $f(\mathbf{x})$를 최소화 (혹은 최대화)하는 $\mathbf{x}$를 찾음
- 최적화 문제의 수학적 표현

$$\begin{array}{cc} \text{minimize } or \text{ maximize} & f(\mathbf{x}) \\ \text{subject to} & \mathbf{x} \in \Omega \end{array}$$

  - $f(\mathbf{x})$: 목적함수 (objective function)
  - $\mathbf{x} = [x_1, \dots, x_n]^T$: 결정변수벡터 (vector of decision variables)
  - $\Omega \subset \mathbb{R}^n$: 가능해집합 (feasible set)
  - 목적함수 f를 최소화(최대화)하는 $\mathbf{x}$를 최적해 (optimal solution) 라 함

최적화 이론

- 예1)

$$\begin{aligned}\text{minimize} \quad & x^2 - 2x + 1 \\ \text{subject to} \quad & x \in [-2,2]\end{aligned}$$

- 예2)

$$\begin{aligned}\text{minimize} \quad & x^2 - 2x + 1 \\ \text{subject to} \quad & x \in [-6,-2]\end{aligned}$$

최적화 이론

예3) 선형회귀

- n 개의 점 $(x_1, y_1), \dots, (x_n, y_n)$
- 위의 점을 가장 적은 오차로 표현하는 직선 $y = ax + b$ 를 찾고 싶을 경우
- 목적 → 평균제곱오차를 최소화하는 직선의 매개변수 a, b를 찾는 최적화 문제

$$\underset{a,b}{\text{minimize}} \frac{1}{n} \sum_{i=1}^{n} (ax_i + b - y_i)^2$$

  - n, x_i, y_i: 상수
  - a, b: 결정변수

매개변수 공간의 탐색

학습 모델의 매개변수 공간

- 높은 차원에 비해 훈련집합의 크기가 작아 참인 확률분포를 구하는 일은 불가능함
- 따라서 기계 학습은 적절한 모델을 선택하고, 목적함수를 정의하고, 모델의 매개변수 공간을 탐색하여 목적함수가 최저가 되는 최적점을 찾는 전략 사용 → 특징 공간에서 해야 하는 일을 모델의 매개변수 공간에서 하는 일로 대치한 셈
- [그림 2-22]는 여러 예제 (Θ는 매개변수, $J(\Theta)$는 목적함수)

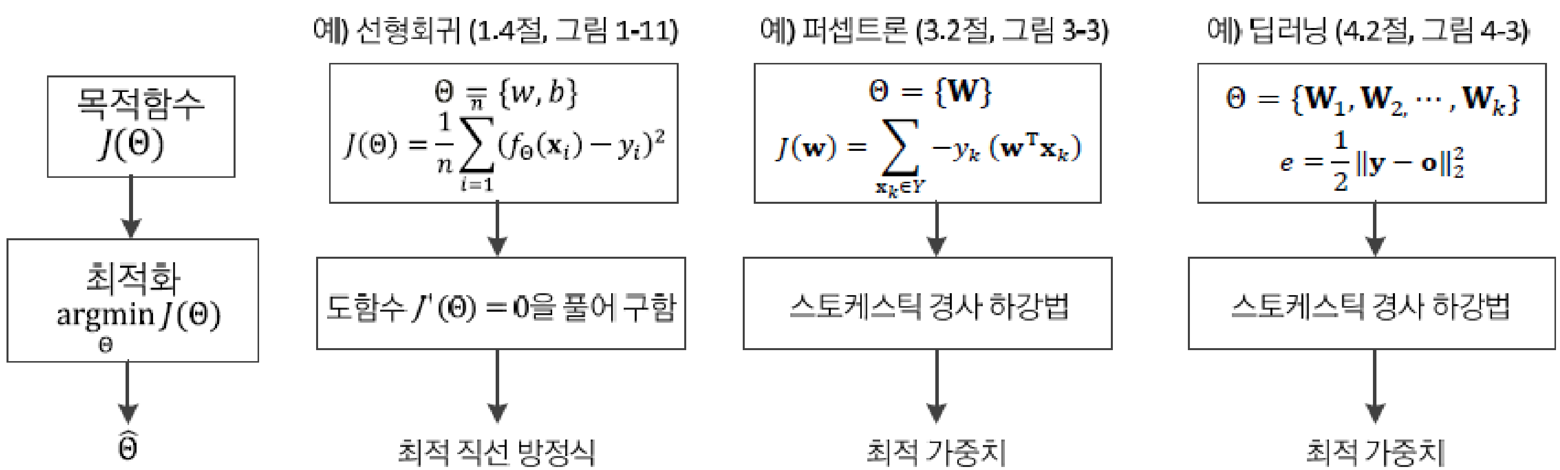


그림 2-22 최적화를 이용한 기계 학습의 문제풀이 과정

매개변수 공간의 탐색

- 학습 모델의 매개변수 공간
  - 특징 공간보다 수 배~수만 배 넓음
    - [그림 2-22]의 선형회귀에서는 특징 공간은 1차원 , 매개변수 공간은 2차원
    - MNIST 인식하는 딥러닝 모델은 784차원 특징 공간, 수십만~수백만 차원의 매개변수 공간
  - [그림 2-23] 개념도의 매개변수 공간: $\hat{x}$은 전역 최적해, x_2와 x_4는 지역 최적해
  - x_2와 같이 전역 최적해에 가까운 지역 최적해를 찾고 만족하는 경우 많음

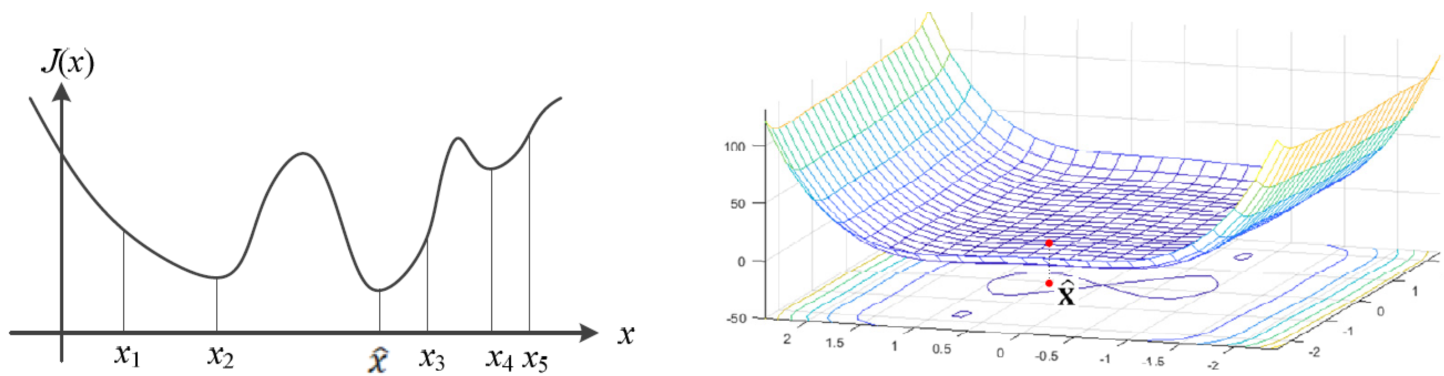


그림 2-23 최적해 탐색

- 기계 학습이 해야 할 일을 식으로 정의하면,

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

매개변수 공간의 탐색

최적화 문제 해결

- 낱낱탐색exhaustive search 알고리즘
  - 차원이 조금만 높아져도 적용 불가능
  - 예) 4차원 Iris에서 각 차원을 1000구간으로 나눈다면 총 1000^4개의 점을 평가해야 함

알고리즘 2-1 낱낱탐색 알고리즘

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

가능한 해를 모두 생성하여 집합 S에 저장한다.
min을 충분히 큰 값으로 초기화한다.
for (S에 속하는 각 점 $\mathbf{\Theta}_{current}$에 대해)
    if($J(\mathbf{\Theta}_{current}) < min$) $min = J(\mathbf{\Theta}_{current})$, $\mathbf{\Theta}_{best} = \mathbf{\Theta}_{current}$
$\hat{\mathbf{\Theta}} = \mathbf{\Theta}_{best}$

- 무작위 탐색 알고리즘
  - 아무 전략이 없는 순진한 알고리즘

알고리즘 2-2 무작위 탐색 알고리즘

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

min을 충분히 큰 값으로 초기화한다.
repeat
    무작위로 해를 하나 생성하고 $\mathbf{\Theta}_{current}$라 한다.
    if($J(\mathbf{\Theta}_{current}) < min$) $min = J(\mathbf{\Theta}_{current})$, $\mathbf{\Theta}_{best} = \mathbf{\Theta}_{current}$
until(멈춤 조건)
$\hat{\mathbf{\Theta}} = \mathbf{\Theta}_{best}$

매개변수 공간의 탐색

- [알고리즘 2-3]은 기계 학습이 사용하는 전형적인 알고리즘
    - 라인 3에서는 목적함수가 작아지는 방향을 주로 미분으로 찾아냄

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

입력 : 훈련집합 $\mathbb{X}$와 $\mathbb{Y}$

출력 : 최적해 $\hat{\boldsymbol{\Theta}}$

1 난수를 생성하여 초기해 $\boldsymbol{\Theta}$을 설정한다.

2 repeat

3 $\quad J(\boldsymbol{\Theta})$가 작아지는 방향 $d\boldsymbol{\Theta}$를 구한다.

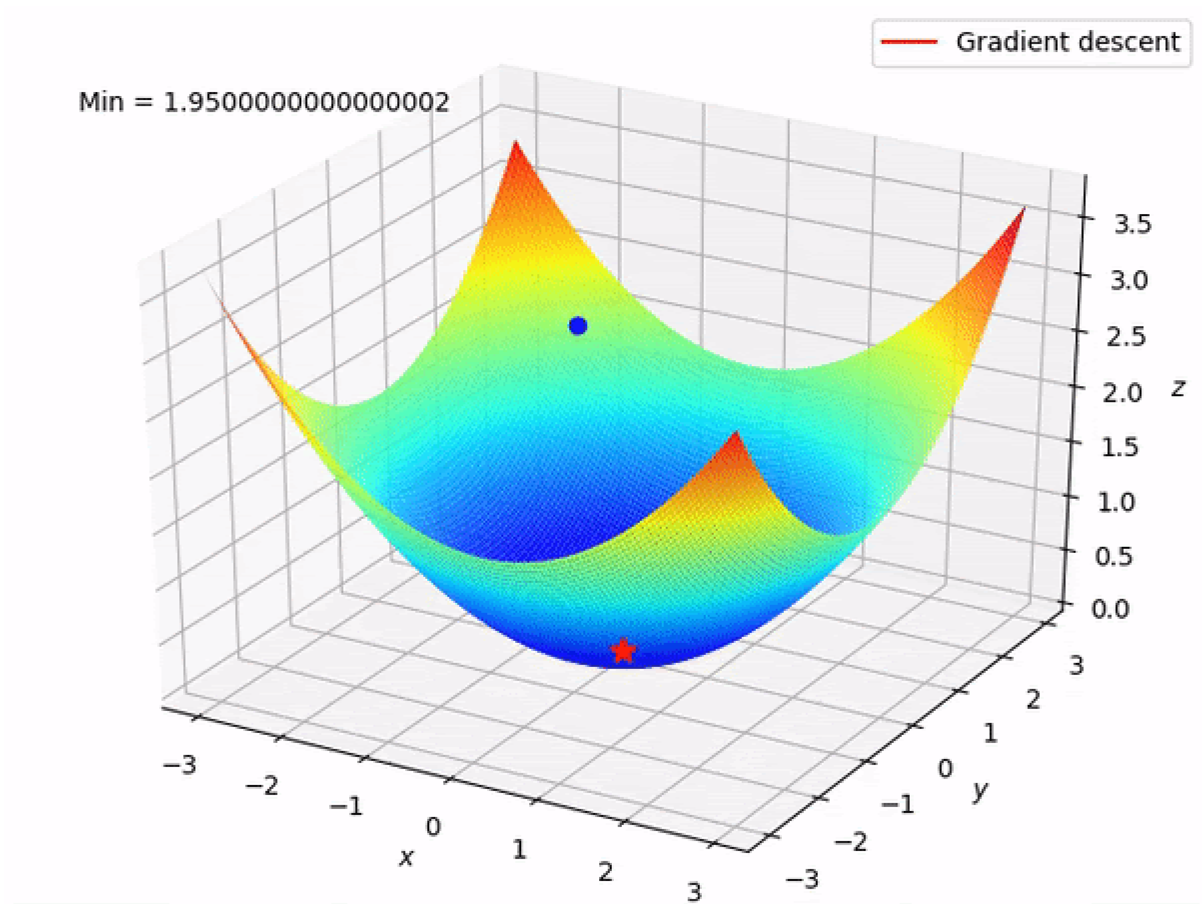
4 $\quad \boldsymbol{\Theta} = \boldsymbol{\Theta} + d\boldsymbol{\Theta}$

5 until(멈춤 조건)

6 $\hat{\boldsymbol{\Theta}} = \boldsymbol{\Theta}$

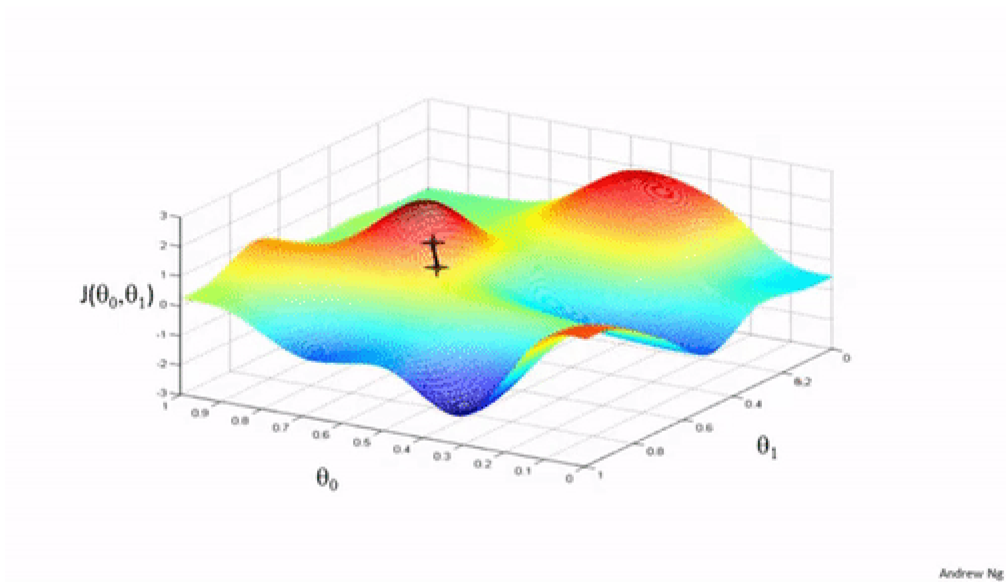
매개변수 공간의 탐색

■ 미분을 이용한 경사하강법



매개변수 공간의 탐색

- 미분을 이용한 경사하강법



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

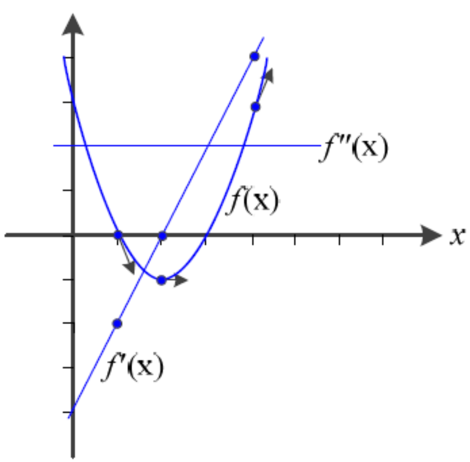


그림 2-24 간단한 미분 예제

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

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

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)

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

기계 학습에서 편미분

- 매개변수 집합 Θ에 많은 변수가 있으므로 편미분을 많이 사용

편미분으로 얻은 그레이디언트에 따라 최저점을 찾아가는 예제

예제 2-10

초기점 $\mathbf{x}_0=(-0.5,0.5)^T$라고 하자. $\mathbf{x}_0$에서의 그레이디언트는 $f'(\mathbf{x}_0) = (-2.5125, -2.5)^T$ 즉, $\nabla f|_{\mathbf{x}_0} = (-2.5125, -2.5)^T$이다. [그림 2-25]는 $\mathbf{x}_0$에서 그레이디언트를 화살표로 표시하고 있어, $-f'(\mathbf{x}_0)$은 최저점의 방향을 제대로 가리키는 것을 확인할 수 있다. 하지만 얼마만큼 이동하여 다음 점 $\mathbf{x}_1$로 옮겨갈지에 대한 방안은 아직 없다. 2.3.3절에서 공부하는 경사 하강법은 이에 대한 답을 제공한다.

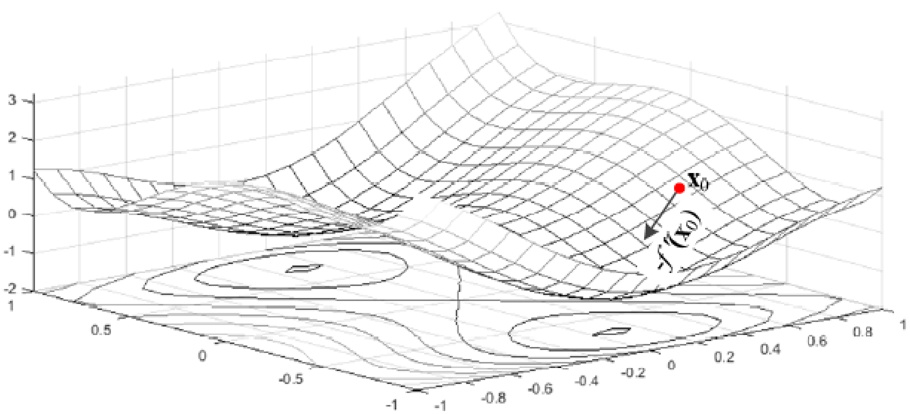


그림 2-25 그레이디언트는 최저점으로 가는 방향을 알려 줌

미분

독립변수와 종속변수의 구분

- 식 (1.2)에서 x는 독립변수, y는 종속변수

$$y = wx + b \tag{1.2}$$

- 기계 학습에서 이런 해석은 무의미 (왜냐하면 예측 단계를 위한 해석에 불과)
- 최적화는 예측 단계가 아니라 학습 단계에 필요
  - 식 (1.8)에서 Θ가 독립변수이고 $e = J(\Theta)$라 하면 e가 종속변수임

$$J(\Theta) = \frac{1}{n}\sum_{i=1}^{n}\left(f_\Theta(\mathbf{x}_i) - y_i\right)^2 \tag{1.8}$$

  - [그림 2-22]는 여러 가지 사례를 보여줌

미분

연쇄법칙

- 합성함수 $f(x) = g(h(x))$의 미분

$$\left.\begin{aligned} f'(x) &= g'\big(h(x)\big)h'(x) \\ f'(x) &= g'\big(h(i(x))\big)h'(i(x))i'(x) \end{aligned}\right\} \qquad (2.53)$$

- 예) $f(x) = 3(2x^2-1)^2 - 2(2x^2-1) + 5$ 일 때 $h(x) = 2x^2 - 1$ 로 두면,

$$f'(x) = \underbrace{(3 * 2(2x^2-1) - 2)}_{g'(h(x))}\underbrace{(2 * 2x)}_{h'(x)} = 48x^3 - 32x$$

다층 퍼셉트론은 합성함수

- $\frac{\partial o_i}{\partial u_{23}^1}$를 계산할 때 연쇄법칙 적용
- 3.4절(오류 역전파)에서 설명

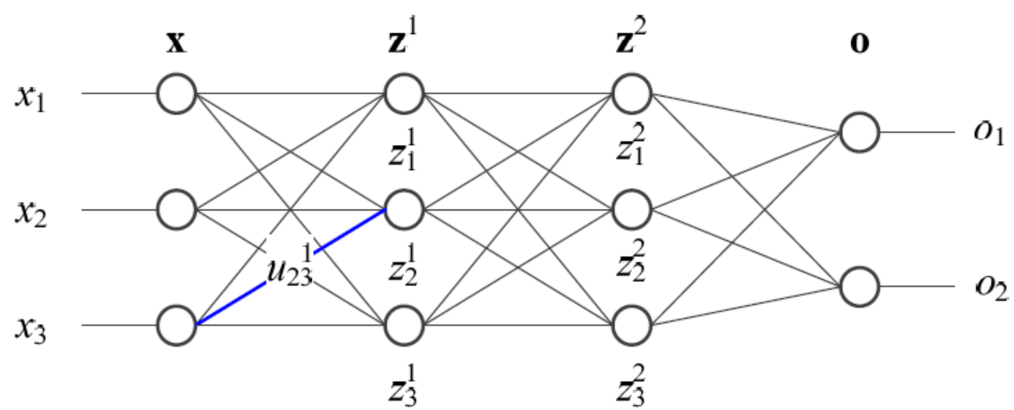


그림 2-26 다층 퍼셉트론은 합성함수

미분

야코비언 행렬

- 함수 $\mathbf{f}: \mathbb{R}^d \mapsto \mathbb{R}^m$을 미분하여 얻은 행렬

$$\text{야코비안 행렬 } \mathbf{J} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_d} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_d} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_d} \end{pmatrix}$$

예)

$\mathbf{f}: \mathbb{R}^2 \mapsto \mathbb{R}^3$인 $\mathbf{f}(\mathbf{x}) = (2x_1 + x_2^2, -x_1^2 + 3x_2, 4x_1x_2)^{\mathrm{T}}$

$$\mathbf{J} = \begin{pmatrix} 2 & 2x_2 \\ -2x_1 & 3 \\ 4x_2 & 4x_1 \end{pmatrix} \qquad \mathbf{J}|_{(2,1)^{\mathrm{T}}} = \begin{pmatrix} 2 & 2 \\ -4 & 3 \\ 4 & 8 \end{pmatrix}$$

헤시안 행렬

- 2차 편도함수

$$\text{헤시안 행렬 } \mathbf{H} = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1 x_1} & \frac{\partial^2 f}{\partial x_1 x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 x_n} \\ \frac{\partial^2 f}{\partial x_2 x_1} & \frac{\partial^2 f}{\partial x_2 x_2} & \cdots & \frac{\partial^2 f}{\partial x_2 x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n x_1} & \frac{\partial^2 f}{\partial x_n x_2} & \cdots & \frac{\partial^2 f}{\partial x_n x_n} \end{pmatrix}$$

예)

$$f(\mathbf{x}) = f(x_1, x_2)$$
$$= \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 + (-4 + 4x_2^2)x_2^2$$

$$\mathbf{H} = \begin{pmatrix} 10x_1^4 - 25.2x_1^2 + 8 & 1 \\ 1 & 48x_2^2 - 8 \end{pmatrix}$$

$$\mathbf{H}|_{(0,1)^{\mathrm{T}}} = \begin{pmatrix} 8 & 1 \\ 1 & 40 \end{pmatrix}$$

경사 하강 알고리즘

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

$$\boxed{\boldsymbol{\Theta} = \boldsymbol{\Theta} - \rho\mathbf{g}} \qquad (2.58)$$

- 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 ρ
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
```

훈련집합

$$\mathbb{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$$

$$\mathbb{Y} = \{y_1, y_2, \cdots, y_n\}$$

경사 하강 알고리즘

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 스토캐스틱 경사 하강 알고리즘(SGD)

입력 : 훈련집합 $\mathbb{X}$와 $\mathbb{Y}$, 학습률 ρ
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   i번째 샘플에 대한 그레이디언트 $\mathbf{\nabla}_i$를 계산한다.
6       $\mathbf{\Theta} = \mathbf{\Theta} - \rho\mathbf{\nabla}_i$
7 until(멈춤 조건)
8 $\hat{\mathbf{\Theta}} = \mathbf{\Theta}$

- 다른 방식의 구현

3 $\mathbb{X}$에서 임의로 샘플 하나를 뽑는다.
4 뽑힌 샘플의 그레이디언트 $\mathbf{\nabla}$를 계산한다.
5 $\mathbf{\Theta} = \mathbf{\Theta} - \rho\mathbf{\nabla}$