



부산대학교

PUSAN NATIONAL UNIVERSITY

Learning from Mistakes

The Perceptron

Linear Scores, Step Predictions, and Weight Updates

BX3600087-140 / DS3600087-140 / AB3600087-140

기계학습개론

2026학년도 2학기

오용경

데이터사이언스전공

yongkyung.oh@pusan.ac.kr

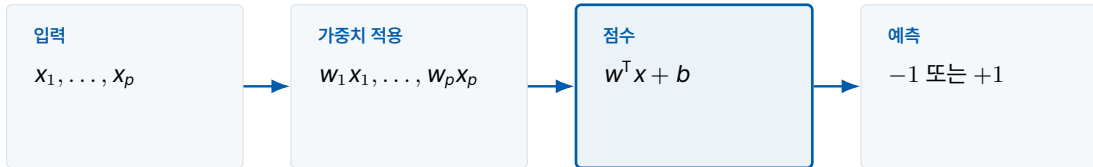
Section 1 • Learning from Mistakes

Single-Layer Model

단층 구조의 입력, 가중치, 출력을 연결한다

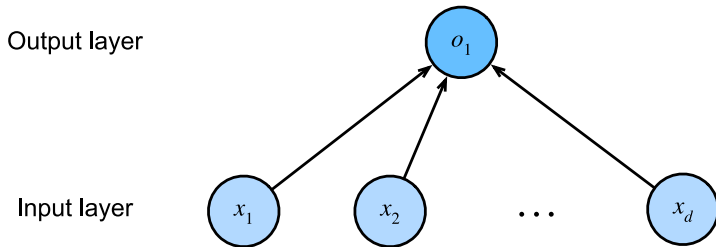
퍼셉트론의 구성

퍼셉트론(perceptron)은 선형 점수의 부호로 두 범주를 분류한다



확률을 계산하지 않고 점수를 0과 비교하여 범주를 정한다

한 층에서 입력이 출력으로 모인다



그림은 입력의 가중합 하나를 출력으로 내는 구조다. 퍼셉트론은 이 가중합의 부호로 범주를 정한다. 그림의 d 는 특성 수 p 다

가중치와 절편

선형식의 계수를 가중치(weight) w 로, 상수항을 절편 b 로 나타낸다

$$s = w_1x_1 + \cdots + w_px_p + b = w^T x + b$$

기호	역할	선형 회귀와 연결
w_j	특성(feature) x_j 의 가중치	계수 β_j
b	입력과 무관한 상수항	절편 β_0

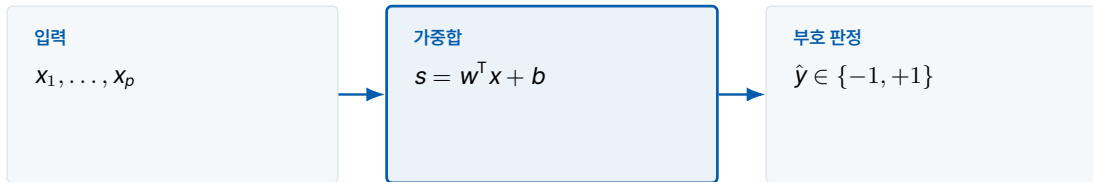
내적이 특성을 점수 하나로 줄인다

특성과 가중치의 내적에 절편을 더하면 선형 점수가 된다

$$\begin{bmatrix} w_1 & w_2 & w_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = w_1 x_1 + w_2 x_2 + w_3 x_3$$

$$w = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}, \quad x = \begin{bmatrix} 1 \\ 3 \\ 2 \end{bmatrix}, \quad b = 0 \quad \implies \quad s = 2 - 3 + 2 = 1$$

입력을 바로 출력 계산에 연결한다



입력과 출력 사이에 다른 계산 단계가 없어, 가중치를 학습하는 연결이 한 층을 이룬다

로지스틱 회귀와 공유하는 부분

두 모델은 같은 선형 점수를 쓰지만 점수 뒤의 계산이 다르다

모델	공통 점수	점수 다음
로지스틱 회귀	$w^T x + b$	sigmoid로 확률을 계산
퍼셉트론	$w^T x + b$	부호 판정으로 범주를 바로 결정

같은 결정 경계를 나타낼 수 있어도, 출력의 의미와 학습 규칙은 다르다

두 벡터의 성분 수가 같아야 내적을 계산할 수 있다

w 와 x 는 모두 성분이 p 개인 벡터여야 한다

대상	배열 모양	계산 결과
x	$p \times 1$	한 관측의 특성 벡터
w	$p \times 1$	성분 수가 같은 가중치 벡터
$w^T x$	1×1	점수 하나
b	스칼라	점수에 더하는 절편

특성이 늘면 가중치도 같은 수만큼 필요하다

Checkpoint

Single-Layer Model

- 퍼셉트론의 입력에서 예측까지 어떤 계산을 거치는가
- 가중치 w_j 와 절편 b 는 점수에서 각각 어떤 역할을 하는가
- 로지스틱 회귀와 퍼셉트론이 공유하는 계산은 무엇인가

Section 2 • Learning from Mistakes

Score and Step

선형 점수의 부호로 범주를 예측하고 결정 경계를 읽는다

선형 분류기의 점수

선형 분류기(linear classifier)는 $w^T x + b$ 하나로 두 범주를 가른다

$$s(x) = w^T x + b$$

점수	예측	공간의 위치
$s(x) \geq 0$	+1	양의 쪽 또는 경계 위
$s(x) < 0$	-1	음의 쪽

Source: MIT 6.390 (2026), Classification.

두 점의 점수 계산

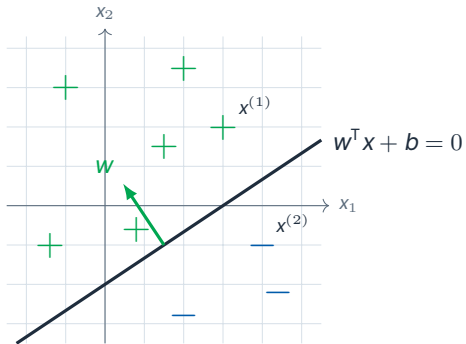
$w = [-1, 1.5]^T, b = 3$ 인 선형 분류기에 두 점을 넣는다

입력	점수	예측
$x^{(1)} = [3, 2]^T$	$-3 + 1.5(2) + 3 = 3$	+1
$x^{(2)} = [4, -1]^T$	$-4 + 1.5(-1) + 3 = -2.5$	-1

점수가 양수이면 +1, 음수이면 -1로 예측한다

Source: MIT 6.390 (2026), Classification.

결정 경계와 법선 벡터

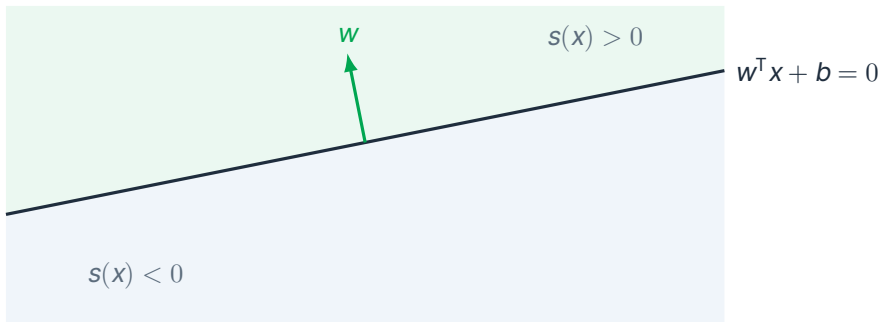


$w = (-1, 1.5)$, $b = 3$ 인 결정 경계(decision boundary)다. w 는 점수가 0인 경계에 수직이고 양의 쪽을 가리킨다

결정 경계는 점수가 0인 점의 집합이다

$w \neq 0$ 이면 2차원에서는 직선, p 차원에서는 초평면이다

$$\{x : w^T x + b = 0\}$$



점수의 부호를 두 출력값으로 바꾼다

step function은 점수가 0 이상이면 1, 음수이면 0을 내는 함수다

$$\text{step}(s) = \begin{cases} 1, & s \geq 0, \\ 0, & s < 0, \end{cases} \quad \hat{y} = 2 \text{step}(s) - 1 \in \{-1, +1\}$$

$$s = -0.2 \Rightarrow \text{step}(s) = 0, \hat{y} = -1 \quad s = 2.3 \Rightarrow \text{step}(s) = 1, \hat{y} = +1$$

퍼셉트론은 step function의 0과 1을 -1 과 $+1$ 로 바꾼 값을 예측값 $\hat{y}$ 로 쓴다

학습 전에 정답 범주의 표기를 맞춘다

퍼셉트론 갱신에서는 두 범주를 -1 과 $+1$ 로 나타낸다

원래 범주 표기	학습에 쓰는 값	변환
0	-1	$y_{\pm} = 2y_{01} - 1$
1	$+1$	$y_{\pm} = 2y_{01} - 1$

$ys(x) > 0$ 이면 올바른 분류, < 0 이면 오분류다. $s(x) = 0$ 은 경계 위의 경우다

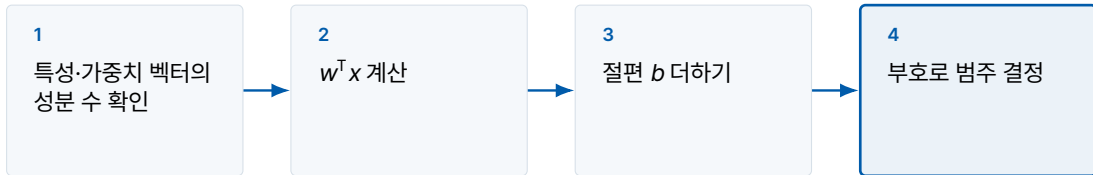
임계값을 절편에 포함할 수 있다

$w^T x$ 를 임계값(threshold) c 와 비교하는 식은 절편 $b = -c$ 를 두어 나타낸다

$$w^T x \geq c \iff w^T x - c \geq 0$$

임계값 표현	절편 표현
$w^T x \geq c$	$w^T x + b \geq 0, b = -c$

예측 계산 순서



$$x \rightarrow s(x) = w^T x + b \rightarrow \hat{y}$$

Checkpoint

Score and Step

- $w^T x + b = 0$ 인 점들은 공간에서 무엇을 만드는가
- 점수가 음수일 때 부호 판정은 어느 범주인가
- $y s(x)$ 가 양수·음수·0인 경우를 각각 어떻게 해석하는가

Section 3 • Learning from Mistakes

Learning from Mistakes

경계 위·오분류 관측에서 매개변수를 갱신하는 과정을 계산한다

정답 범주와 점수의 부호를 비교한다

정답 범주 y_i 와 점수 s_i 의 부호가 다르면 예측이 틀렸다

$$s_i = w^T x_i + b, \quad \begin{cases} y_i s_i > 0 & \text{올바른 분류,} \\ y_i s_i \leq 0 & \text{오분류 또는 경계 위.} \end{cases}$$

y_i	s_i	$y_i s_i$	판정
+1	2	2	올바른 분류
-1	2	-2	오분류

x_i 는 i 번째 관측의 특성 벡터로, $x^{(i)}$ 로도 쓴다

오분류하거나 경계 위에 놓인 관측에서 갱신한다

$y_i s_i \leq 0$ 인 관측의 점수가 정답 부호 쪽으로 이동하도록 갱신한다

$$w \leftarrow w + y_i x_i, \quad b \leftarrow b + y_i \quad \text{if } y_i(w^T x_i + b) \leq 0$$

관측 한 건	가중치 변화	절편 변화
$y_i = +1, s_i \leq 0$	$+x_i$	$+1$
$y_i = -1, s_i \geq 0$	$-x_i$	-1
$y_i(w^T x_i + b) > 0$	변화 없음	변화 없음

정답 쪽에 놓인 관측에서는 w 와 b 를 유지한다

$y_i s_i > 0$ 일 때만 (w, b) 를 그대로 두므로, 경계 위의 점은 예측 범주와 관계없이 갱신한다

$$m_i = y_i(w^T x_i + b)$$

$$(w, b) \leftarrow \begin{cases} (w, b), & m_i > 0 \\ (w + y_i x_i, b + y_i), & m_i \leq 0 \end{cases}$$

$$y_i s_i^{\text{new}} = y_i((w + y_i x_i)^T x_i + b + y_i) = y_i s_i + \|x_i\|^2 + 1$$

$y_i^2 = 1$ 이므로 갱신은 그 관측의 $y_i s_i$ 를 늘린다. 다만 한 번의 갱신으로 양수가 된다는 보장은 없다

$y_i = -1$ 이면 x_i 를 뺀다

$y_i x_i = -x_i$ 이므로 가중치가 입력 벡터의 반대 방향으로 움직인다

$$x_i = \begin{bmatrix} -1 \\ 2 \end{bmatrix}, \quad y_i = -1, \quad w = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad b = 0$$

$$w_{\text{new}} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} + (-1) \begin{bmatrix} -1 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 \\ -2 \end{bmatrix}, \quad b_{\text{new}} = -1$$

갱신 방향은 정답 범주와 특성 벡터가 함께 정한다

계산할 관측과 초기값

네 관측을 주어진 순서로 한 번씩 보고 학습률 1로 갱신한다

순서	x_i	y_i	위치
1	(2, 1)	+1	오른쪽, +1 범주
2	(-1, 2)	-1	왼쪽, -1 범주
3	(2, 2)	+1	오른쪽, +1 범주
4	(-2, 1)	-1	왼쪽, -1 범주

초기값은 $w^{(0)} = (0, 0)$, $b^{(0)} = 0$ 이다. $y_i s_i \leq 0$ 이면 경계 위도 갱신한다

첫 두 관측에서의 갱신

첫 관측은 점수가 0이라 경계 위에 있어 갱신하고, 두 번째 관측은 첫 갱신 뒤 오분류되어 다시 갱신한다

i	s_i	$y_i s_i$	판정	갱신 뒤 (w, b)
1	0	0	경계 위: 갱신	$((2, 1), 1)$
2	1	-1	오분류: 갱신	$((3, -1), 0)$

$$(0, 0), 0 \xrightarrow{(2,1), +1} (2, 1), 1 \xrightarrow{(-1,2), -1} (3, -1), 0$$

나머지 두 관측의 판정

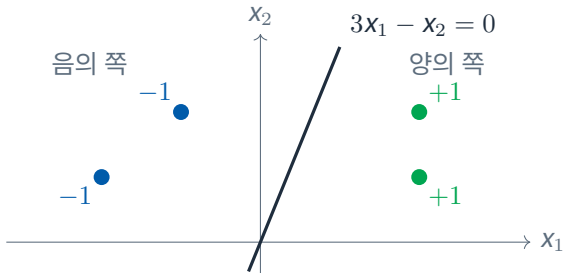
두 번 갱신한 경계는 세 번째와 네 번째 관측을 모두 올바르게 분류한다

i	s_i	$y_i s_i$	판정	갱신 뒤 (w, b)
3	$3(2) - 2 = 4$	4	올바른 분류	$((3, -1), 0)$
4	$3(-2) - 1 = -7$	7	올바른 분류	$((3, -1), 0)$

두 관측 모두 $y_i s_i > 0$ 이므로 $w = (3, -1)$, $b = 0$ 을 유지한다

첫 에포크 뒤의 결정 경계

$w = (3, -1)$, $b = 0$ 은 $3x_1 - x_2 = 0$ 을 결정 경계로 만든다



두 번의 갱신 결과 네 관측이 모두 정답 쪽에 놓인다

Checkpoint

Learning from Mistakes

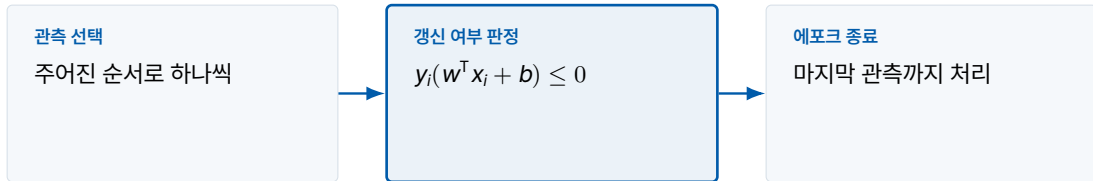
- $y_i(w^\top x_i + b) \leq 0$ 은 어떤 상황을 뜻하는가
- 갱신할 관측의 정답이 +1일 때와 -1일 때 w 는 어떻게 변하는가
- 경계 위의 관측도 갱신하는 이유는 무엇인가

Section 4 • Learning from Mistakes

Perceptron in Code

관측 순회, 갱신 횟수, 에포크를 코드와 연결한다

한 에포크에서 모든 관측을 한 번씩 확인한다



가중치와 절편은 학습 시작 때 초기화하고, 에포크 사이에는 유지한다

NumPy로 구현한 퍼셉트론 학습

오분류한 관측과 경계 위의 관측에서만 가중치와 절편을 갱신한다

```
1 w = np.zeros(X.shape[1])
2 b = 0.0
3 for _ in range(num_epochs):
4     updates = 0
5     for x_i, y_i in zip(X, y):
6         score = w @ x_i + b
7         if y_i * score <= 0:
8             w += y_i * x_i
9             b += y_i
10            updates += 1
11    if updates == 0:
12        break
```

Adapted from Bishop (2006), Section 4.1.7.

에포크별 갱신 횟수를 기록한다

각 에포크에서 매개변수가 몇 번 바뀌었는지 기록한다

에포크	갱신 횟수	해석
1	2	경계 위·오분류 관측에서 갱신
2	0	모든 관측에서 $y_i s_i > 0$
3	실행하지 않음	종료 조건 만족

한 에포크의 갱신이 0회이면 모든 관측이 경계 밖 정답 쪽에 놓여 있다

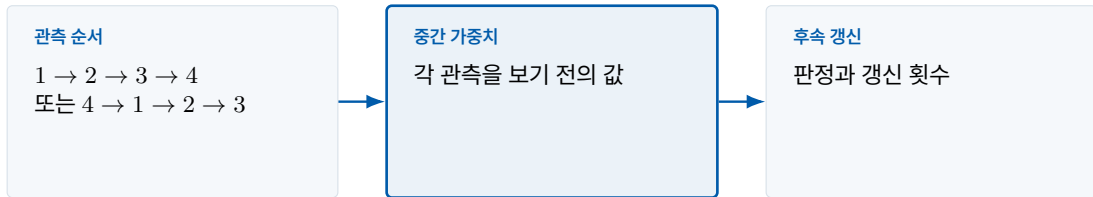
종료 조건과 반복 상한

갱신 횟수가 0이 되지 않는 경우에 대비해 최대 에포크 수도 정한다

조건	멈추는 때	의미
갱신 횟수 = 0	한 에포크에서 갱신 없음	모든 관측에서 $y_i s_i > 0$
최대 에포크 수 도달	정한 반복 상한에 도달	계산을 끝내고 결과 확인

관측 순서가 갱신 경로를 바꾼다

같은 자료라도 관측 순서에 따라 중간 가중치, 갱신 횟수, 최종 경계가 달라질 수 있다



$1 \rightarrow 2 \rightarrow 3 \rightarrow 4$ 는 2회 갱신해 $w = (3, -1)$, $b = 0$ 에서, $4 \rightarrow 1 \rightarrow 2 \rightarrow 3$ 은 1회 갱신해 $w = (2, -1)$, $b = -1$ 에서 멈춘다

재현하려면 초기값뿐 아니라 관측을 처리한 순서도 기록한다

경사하강법과 퍼셉트론 갱신

두 방법 모두 매개변수를 반복해서 바꾸지만, 갱신 기준과 시점이 다르다

비교	경사하강법	퍼셉트론
갱신 기준	목적함수의 그래디언트	현재 관측의 $y_i s_i \leq 0$
갱신 시점	각 반복	오분류 또는 경계 위
방향	$-\nabla J$	$y_i x_i$
곱하는 계수	학습률 η	학습률 1

오분류 관측에서 퍼셉트론 갱신은 손실 $-y_i s_i$ 에 학습률 1을 쓴 SGD 형태의 한 단계와 같다

퍼셉트론 구현의 출발점

퍼셉트론 학습 규칙을 구현하고 매개변수와 갱신 횟수를 기록한다

단계	구현	확인
1	점수와 예측	범주 표기와 경계 위 예측 규칙
2	갱신 조건	오분류와 경계 위를 모두 처리하는가
3	w, b 갱신	손계산 결과와 같은가
4	에포크 반복	에포크별 갱신 횟수를 기록하는가

Checkpoint

Perceptron in Code

- 한 에포크와 한 번의 갱신은 각각 무엇을 뜻하는가
- 한 에포크의 갱신 횟수가 0이면 어떤 상태인가
- 경사하강법과 퍼셉트론 갱신은 방향을 무엇에서 얻는가

What If No Line Can Separate the Data?

Next Lecture: Linear Separability

- 선형 분리 경계와 경계 양쪽의 공간
- 선형 분리가 가능한 자료
- XOR와 단층 구조의 한계