

퍼셉트론  
법

- 딥러닝(deep learning)의 시작은 1950년대부터 연구되어 온 인공 신경망(artificial neural network: ANN)이다.
- 인공 신경망 : 생물학적인 신경망에서 영감을 받아서 만들어진 컴퓨팅 구조

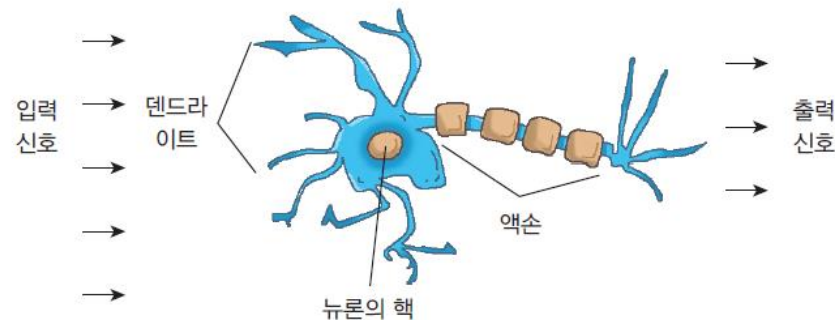


그림 5-1 뉴런의 구조

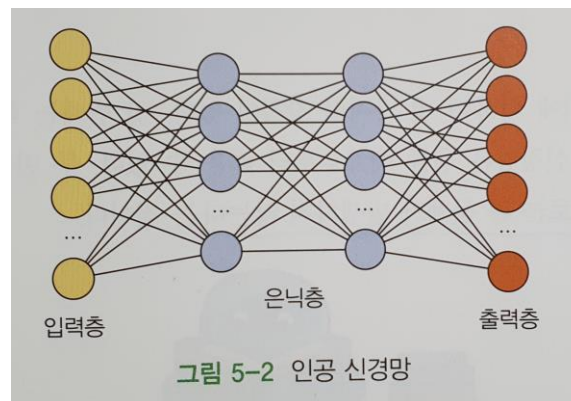
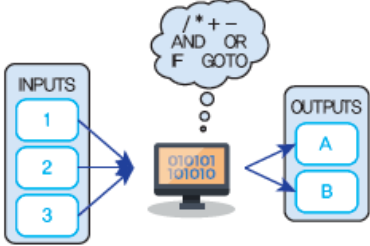



그림 5-2 인공 신경망



# 전통적인 컴퓨터 VS 인공지능망

	기존의 컴퓨터	인간의 두뇌
처리소자의 개수	$10^8$ 개의 트랜지스터	$10^{10}$ 개의 뉴런
처리소자의 속도	$10^{12}$ Hz	$10^2$ Hz
학습기능	없음	있음
계산 스타일	중앙집중식, 순차적인 처리	분산 병렬 처리
		

# 신경망의 장점

- 학습이 가능하다. 데이터만 주어진다면 신경망은 예제로부터 배울 수 있다. -> 프로그램을 작성할 필요가 없다.

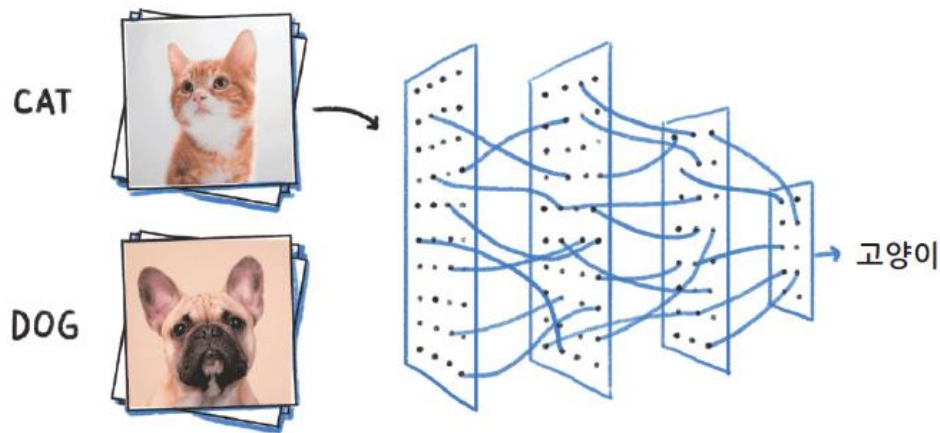


그림 5-4 신경망을 이용한 이미지 인식

- 몇 개의 소자가 오동작하더라도 전체적으로는 큰 문제가 발생하지 않는다. - 분산병렬처리



# 퍼셉트론(perceptron)

- 1957년에 로젠블라트(Frank Rosenblatt)가 고안한 인공 신경망
- 하나의 유닛(뉴런)만을 사용하는 모델.

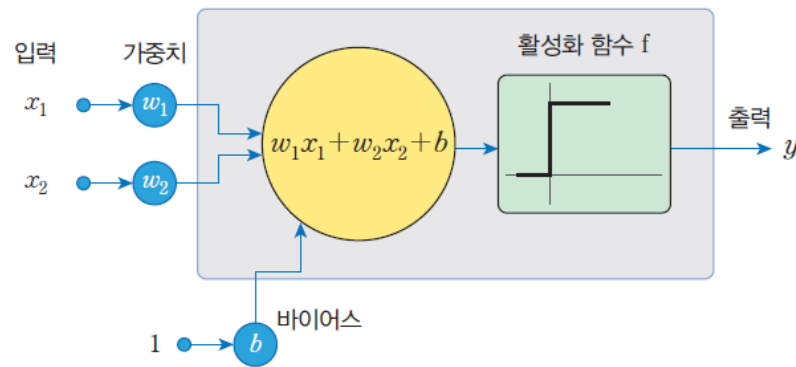


그림 5-5 퍼셉트론에서의 뉴런의 모델

- 뉴런에서는 입력 신호( $x$ )의 가중치( $w$ ) 합이 어떤 임계값( $b$ )을 넘는 경우에만 뉴런이 활성화되어서 1을 출력한다. 그렇지 않으면 0을 출력한다.

$$y = \begin{cases} 1 & \text{if } (w_1x_1 + w_2x_2 + b \geq 0) \\ 0 & \text{otherwise} \end{cases}$$



# 퍼셉트론은 논리 연산을 학습할 수 있을까

- AND 소자처럼 동작하려면

$x_1$	$x_2$	$y$
0	0	0
1	0	0
0	1	0
1	1	1

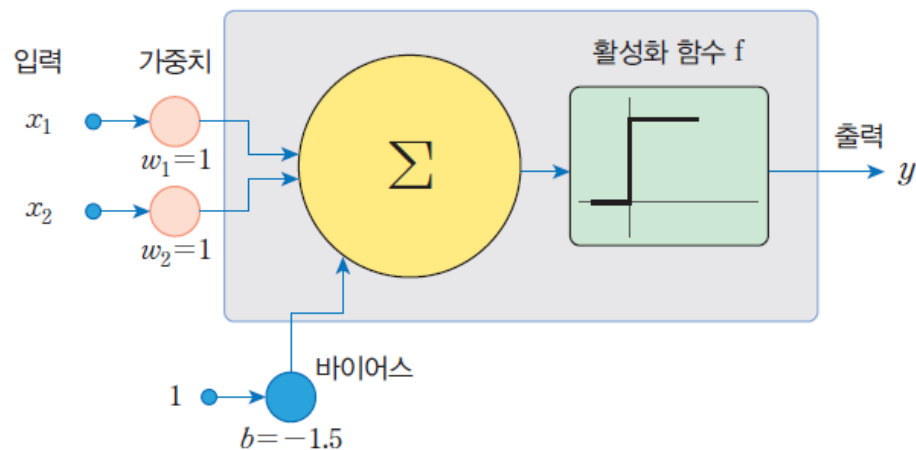


그림 5-6 논리 연산을 하는 퍼셉트론

표 5-2 퍼셉트론 출력 계산

$x_1$	$x_2$	$w_1x_1 + w_2x_2$	$b$	$y$
0	0	$1*0+1*0=0$	-1.5	0
1	0	$1*1+1*0=1$	-1.5	0
0	1	$1*0+1*1=1$	-1.5	0
1	1	$1*1+1*1=2$	-1.5	1



# 퍼셉트론은 논리 연산을 학습할 수 있을까

- 매개변수 ( $w_1, w_2, b$ ) 값이  $(1, 1, -1.5)$ 이면 퍼셉트론은 **AND** 연산을 한다.  $(0.5, 0.5, -0.7)$ 거나  $(0.5, 0.5, -0.8)$  일 때도 만족하다. 무수히 많다.
- 컴퓨터가 자동으로 매개변수의 값을 찾아야. 그것이 바로 머신러닝.

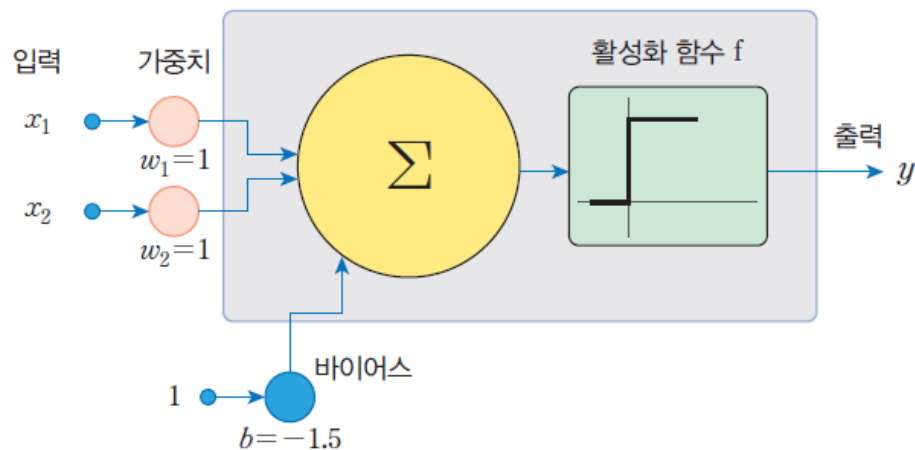


그림 5-6 논리 연산을 하는 퍼셉트론

- 1. 퍼셉트론이 **OR** 소자처럼 동작하려면 가중치와 임계값은 어떤 값이어야 할까 ?
- 2. 퍼셉트론이 **NAND** 소자처럼 동작하려면 가중치와 임계값은 어떤 값이어야 할까 ?

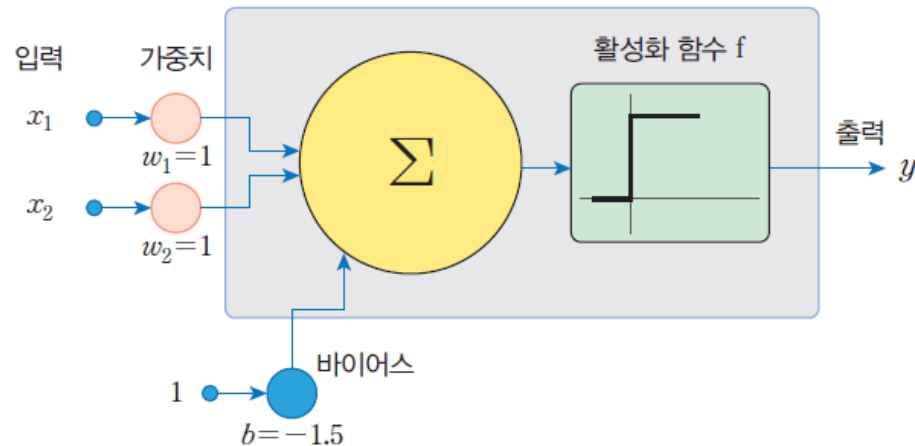


그림 5-6 논리 연산을 하는 퍼셉트론



- 계단 함수

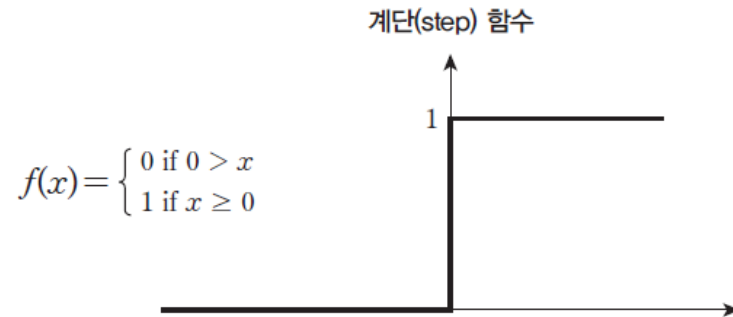


그림 5-7 퍼셉트론에서의 활성화 함수

- 참고사항 (p171)
  - 가중치 - 입력 신호가 출력에 미치는 중요도를 조절하는 역할
  - 바이어스 - 얼마나 쉽게 활성화되느냐를 결정하는 역할



# 퍼셉트론 구현 #1 (순수 파이썬 사용)





# 퍼셉트론 구현 #2 (넘파이 사용)





# 퍼셉트론 학습 알고리즘

- 학습 = 신경망이 스스로 가중치를 자동으로 설정해주는 알고리즘

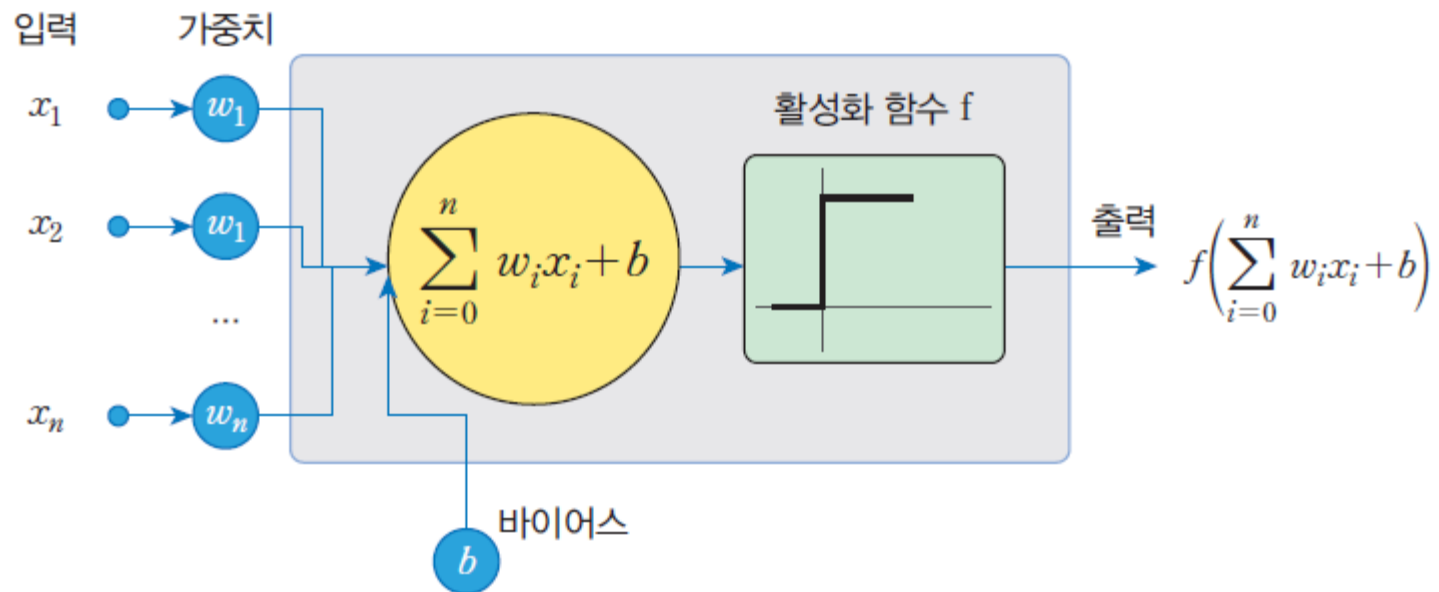


그림 5-8 퍼셉트론



# 퍼셉트론 학습 알고리즘

input: 학습 데이터  $(x^1, d^1), \dots, (x^m, d^m)$

- ① 모든  $w$ 와 바이어스  $b$ 를 0 또는 작은 난수로 초기화한다.
- ② while (가중치가 변경되지 않을 때까지 반복)
- ③     for 각 학습 데이터  $x^k$ 와 정답  $d^k$
- ④          $y^k(t) = f(w(t) \cdot x^k)$
- ⑤         모든 가중치  $w_i$ 에 대하여  $w_i(t+1) = w_i(t) + \eta \cdot (d^k - y^k(t)) \cdot x_i^k$



# 노리 연산자 학습 과정

$$w_i(t+1) = w_i(t) + \eta \cdot (d^k - y^k(t)) \cdot x_i^k$$

- 퍼셉트론이 1을 0으로 잘못 식별했다고 하자. 가중치의 변화량은  $\eta * (1-0) * x_i^k$  가 된다. 따라서 가중치는 증가된다. 가중치가 증가되면 출력도 증가되어서 출력이 0에서 1이 될 가능성이 있다.
- 반대로 0을 1로 잘못 식별했다고 하자. 가중치의 변화량은  $\eta * (0-1) * x_i^k$  가 된다. 따라서 가중치는 줄어든다. 가중치가 줄어들면 출력도 감소되어서 출력이 1에서 0이 될 가능성이 있다.



# 퍼셉트론 학습 알고리즘 구현하기





# sklearn으로 퍼셉트론 학습하기







# 퍼셉트론의 한계점

- XOR 연산

x1	x2	y
0	0	0
1	0	1
0	1	1
1	1	0

...  
0 0 -> 1  
0 1 -> 1  
1 0 -> 0  
1 1 -> 0

원하는 출력이 나오지 않는다.



# 선형 분류 가능 문제

- 패턴 인식 측면에서 보면 퍼셉트론은 직선을 이용하여 입력 패턴을 분류하는 선형 분류자(linear classifier)의 일종이라고 말할 수 있다.

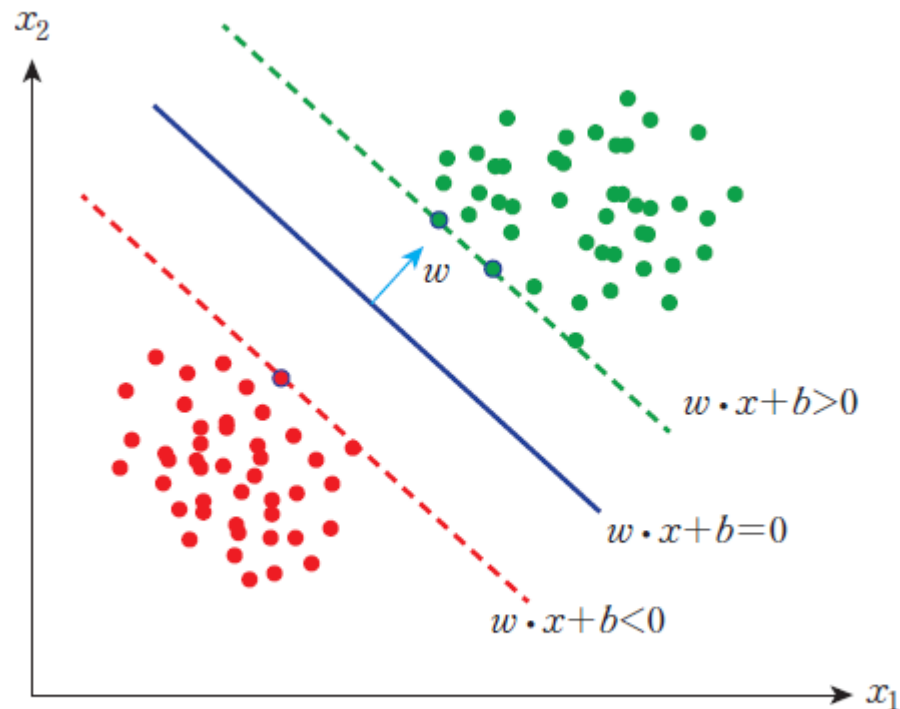


그림 5-10 선형 분류자



# 선형 분리 가능 문제

- Minsky와 Papert는 1969년에 발간된 책 “Perceptrons”에서 1개의 레이어(layer, 계층)으로 구성된 퍼셉트론은 XOR 문제를 학습할 수 없다는 것을 수학적으로 증명 -> 신경망의 첫번째 암흑기

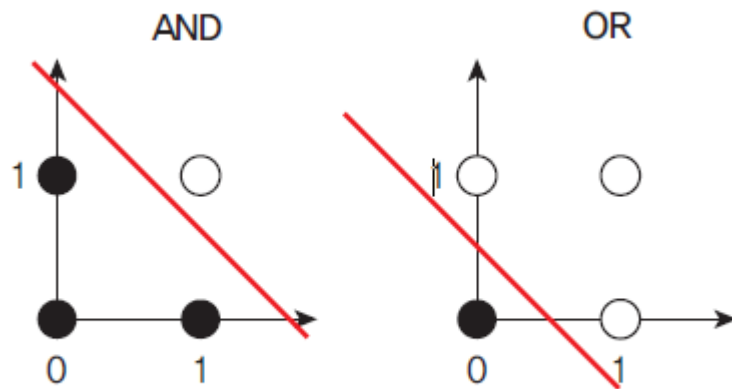


그림 5-11 선형 분리 가능한 문제

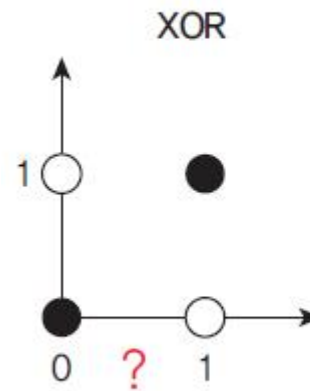


그림 5-12 선형 분리 불가능한 문제



어떻게 하나의 직선으로 분류할 수 있을까?



# 다층 퍼셉트론으로 XOR 문제를 해결

- 입력층과 출력층 사이에 은닉층을 두면 XOR 문제를 해결할 수 있다.  
-> 다층 퍼셉트론

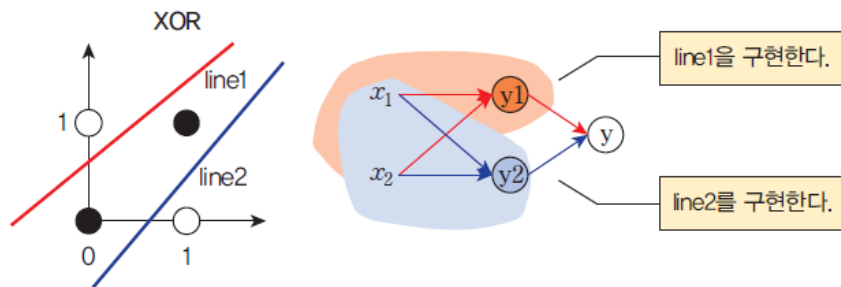


그림 5-13 다층을 사용하는 퍼셉트론

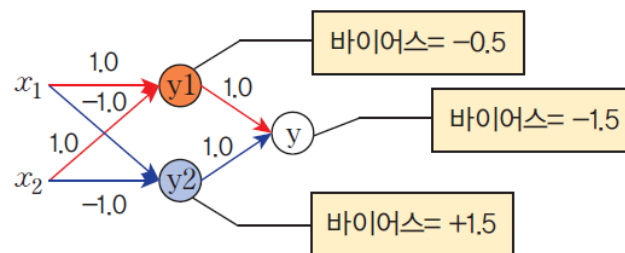


그림 5-14 다층 퍼셉트론에서 XOR 문제 해결

$x_1$	$x_2$	$y_1$	$y_2$	$y$	XOR 출력
0	0	0	1	0	0
1	0	1	1	1	1
0	1	1	1	1	1
1	1	1	0	0	0



# Mini Project: 퍼셉트론으로 분류

- 대학생들의 신장과 체중을 받아서 성별을 출력하는 퍼셉트론을 만들어보자.

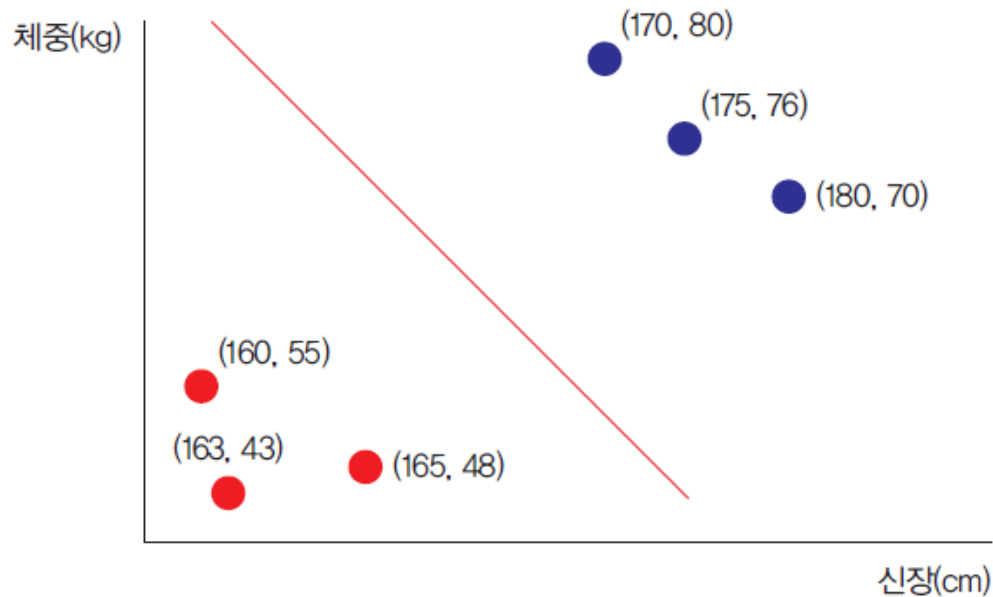


그림 5-15 신장과 체중으로 남녀를 구분하는 문제