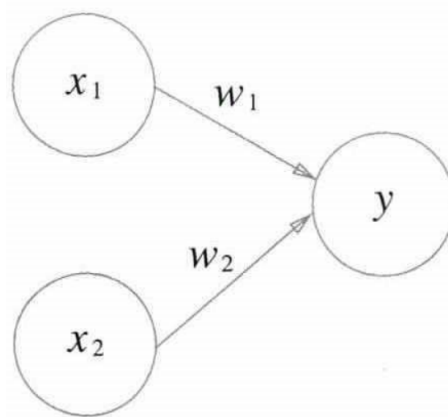


Chapter 2

퍼셉트론

2.1 퍼셉트론이란?

- 퍼셉트론 : 다수의 신호를 입력으로 받아 하나의 신호를 출력하는 시스템



$$y = \begin{cases} 0 & (w_1x_1 + w_2x_2 \leq \theta) \\ 1 & (w_1x_1 + w_2x_2 > \theta) \end{cases}$$

위의 식에서 세타(theta)는 임계값을 의미한다.

◦

2.2 단순한 논리 회로

2.2.1 AND 게이트

- AND 게이트의 진리표

해당 조건을 만족하는 매개 변수의 조합은 무수히 많음

$$EX(w_1, w_2, \theta) = (0.5, 0.5, 0.7), (0.5, 0.5, 0.8)$$

2.2.2 NAND 게이트와 OR 게이트

NAND(Not And)게이트

X_1	X_2	Y
0	0	1
1	0	1
0	1	1
1	1	0

$$EX(w_1, w_2, \theta) = (-0.5, -0.5, -0.7), (-0.5, -0.5, -0.8)$$

OR 게이트(입력 신호 중 하나 이상이 1이면 출력이 1이 되는 논리 회로)

X_1	X_2	Y
0	0	0
1	0	1
0	1	1
1	1	1

$$EX(w_1, w_2, \theta) = (0.8, 0.8, 0.3)$$

2.3 퍼셉트론 구현하기

2.3.1 간단한 구현부터

```
def AND(x1, x2):
    w1, w2, theta = 0.5, 0.5, 0.7
    tmp = x1 * w1 + x2 * w2
    if tmp <= theta:
        return 0
    elif tmp > theta:
        return 1
```

2.3.2 가중치와 편향 도입

$$y = \begin{cases} 0 & (b + w_1x_1 + w_2x_2 \leq 0) \\ 1 & (b + w_1x_1 + w_2x_2 > 0) \end{cases}$$

임계값을 b(편향)으로 바꿈

2.3.3 가중치와 편향 구현하기

- AND 게이트

```
# coding: utf-8
import numpy as np

def AND(x1, x2):
    x = np.array([x1, x2])
    w = np.array([0.5, 0.5])
    b = -0.7
    tmp = np.sum(w*x) + b
    if tmp <= 0:
        return 0
    else:
        return 1
```

- NAND 게이트

```
# coding: utf-8
import numpy as np

def NAND(x1, x2):
    x = np.array([x1, x2])
    w = np.array([-0.5, -0.5])
    b = 0.7
    tmp = np.sum(w*x) + b
    if tmp <= 0:
        return 0
    else:
        return 1
```

- OR 게이트

```
# coding: utf-8
import numpy as np

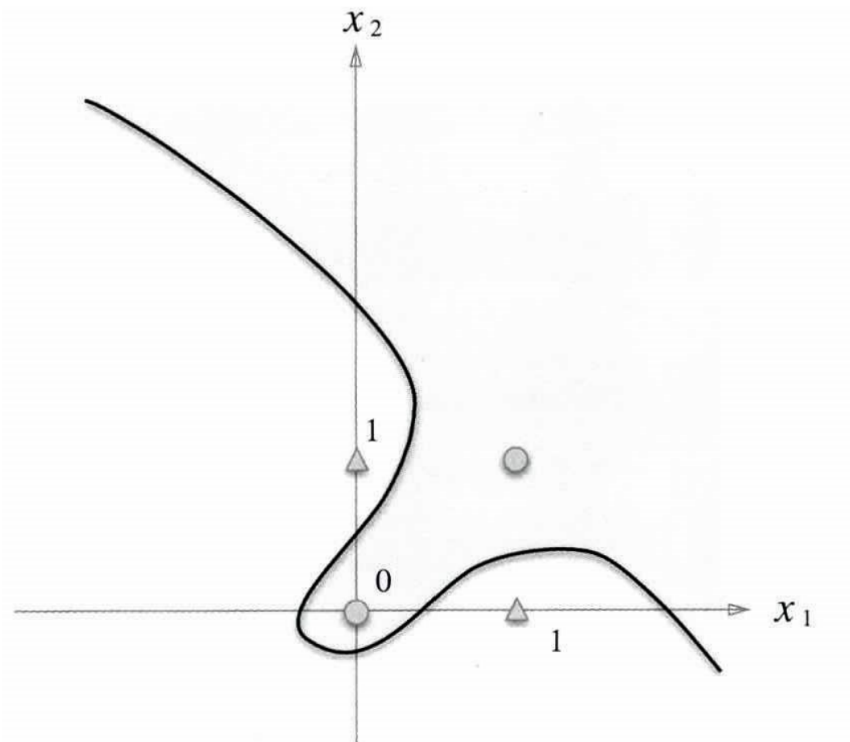
def OR(x1, x2):
    x = np.array([x1, x2])
    w = np.array([0.5, 0.5])
    b = -0.2
    tmp = np.sum(w*x) + b
    if tmp <= 0:
        return 0
    else:
        return 1
```

2.4 퍼셉트론의 한계

2.4.1 도전! XOR 게이트

XOR 게이트는 베타적 논리합이라는 논리회로이다.

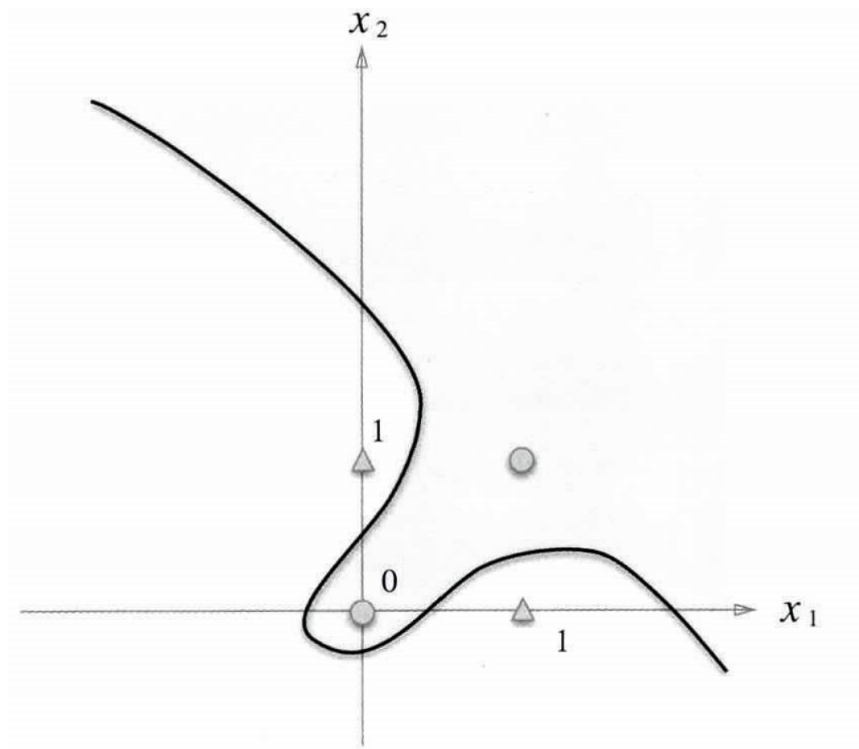
x_1	x_2	Y
0	0	0
1	0	1
0	1	1
1	1	0



- 가중치 매개변수로는 해당 식을 만족하는 조합식을 찾을 수 없음

위의 그림처럼 동그라미와 세모를 직선 하나로 나누는 방법은 불가능하다!

2.4.2 선형과 비선형



따라서 위의 그림처럼 비선형 관계로만 가능하다

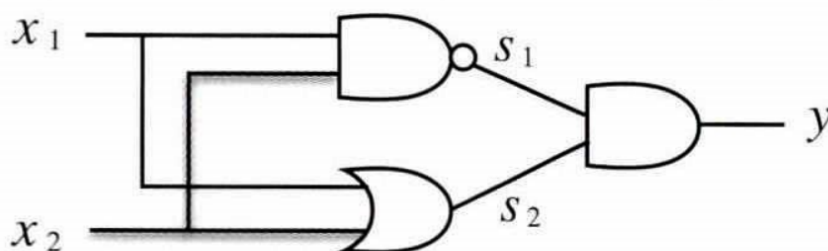
2.5 다층 퍼셉트론이 출동한다면

- XOR 게이트 다층 퍼셉트론으로 해결 가능하다

2.5.1 기존 게이트 조합하기

- AND, NAND, OR 게이트를 조합해 구현한 XOR 게이트

그림 2-11 AND, NAND, OR 게이트를 조합해 구현한 XOR 게이트



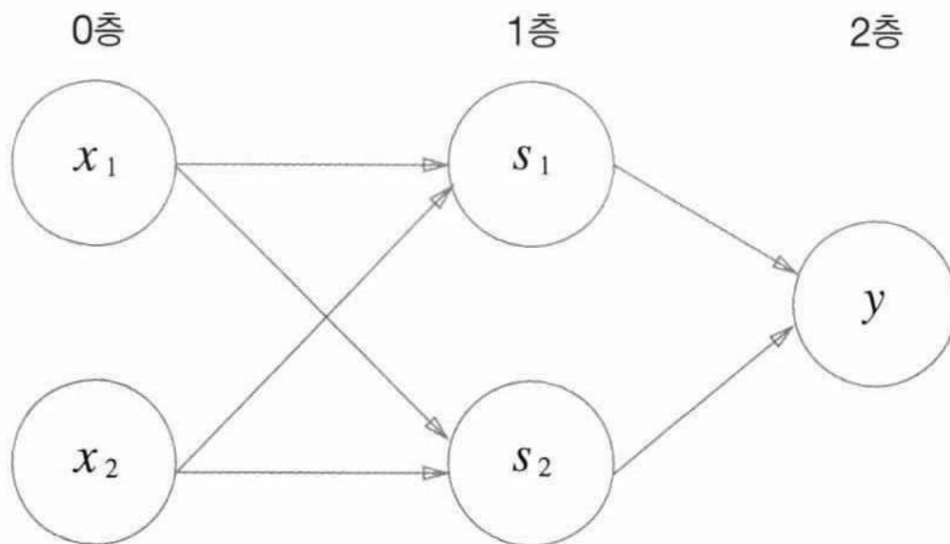
x_1	x_2	s_1	s_2	y
0	0	1	0	0
1	0	1	1	1

0	1	1	1	1
1	1	0	1	0

2.5.2 XOR 게이트 구현하기

```
def XOR(x1, x2):
    s1 = NAND(x1, x2)
    s2 = OR(x1, x2)
    y = AND(s1, s2)
    return y
```

그림 2-13 XOR의 퍼셉트론



- 위와 같이 2층 퍼셉트론으로 시각화가 가능하고 이를 다층 퍼셉트론이라 한다.

2.6 NAND에서 컴퓨터까지

- 신기하게 이론상 2층 퍼셉트론이면 컴퓨터를 만들 수 있다!
- 물론 비선형인 시그모이드 함수를 활성화 함수로 이용하면 임의의 함수를 표현할 수 있다는 증명으로 입증됨
- 결론 : 퍼셉트론은 층을 거듭 쌓으면 비선형적인 표현도 가능하고, 이론상 컴퓨터가 수행하는 처리도 모두 표현 가능하다!