

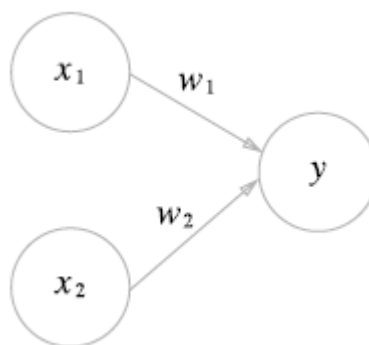
Chapter 2. 퍼셉트론

신경망(딥러닝)의 기원이 되는 알고리즘

2.1 퍼셉트론이란?

다수의 신호를 입력으로 받아 하나의 신호를 출력한다.

신호가 흐른다(1) / 안 흐른다(0)



x: 입력 신호, w: 가중치, y: 출력 신호, 원: 노드/뉴런

입력 신호가 뉴런에 보내질 때 각각 고유한 가중치가 곱해지고, 뉴런에서 보내온 신호의 총합이 정해진 한계(임계값, θ)를 넘어설 때만 1을 출력한다(뉴런이 활성화한다).

$$y = \begin{cases} 0 & (w_1 x_1 + w_2 x_2 \leq \theta) \\ 1 & (w_1 x_1 + w_2 x_2 > \theta) \end{cases}$$

* 주의) θ 초과임!

퍼셉트론은 복수의 입력 신호 각각에 고유한 가중치를 부여한다. 가중치는 각 신호가 결과에 영향력을 조절하는 요소로 작용한다. 즉, 가중치가 클수록 해당 신호가 그만큼 더 중요함을 뜻한다.

2.2 단순한 논리 회로

2.2.1 AND 게이트

AND 게이트를 퍼셉트론으로 표현하기 위

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

AND 게이트의 진리표(입력 신호와 출력 신호의 대응 표)

해 w_1, w_2, θ 의 값을 정해야 한다.

→ 매개변수 조합은 무한히 많음 ex. (0.5, 0.5, 0.7) (0.5, 0.5, 0.8) (1.0, 1.0, 1.0)

2.2.2 NAND 게이트와 OR 게이트

Not AND 게이트 : AND 게이트의 출력을 뒤집은 결과

기계학습 문제는 이 매개변수의 값을 정하는 작업을 컴퓨터가 자동으로 하도록 한다.

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

NAND 게이트, x_1 과 x_2 가 모두 1일때만 0

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

OR 게이트, 하나라도 1이면 1

2.3 퍼셉트론 구현하기

2.3.1 간단한 구현부터

```
def AND_basic(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
```

AND(0, 0)

참고) <https://excelsior-cjh.tistory.com/169>

2.3.2 가중치와 편향 도입

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

위 식의 θ 를 $-b$ 로 치환한 식이다.

여기에서 b 를 **편향**이라 한다.

2.3.3 가중치와 편향 구현하기

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

가중치와 편향은 기능이 다르다: 가중치는 각 입력 신호가 결과에 주는 영향력(**중요도**)을 조절하는 매개변수고, 편향은 **뉴런이 얼마나 쉽게 활성화**(결과로 1을 출력)하느냐를 조정하는 매개변수다.

문맥에 따라 w 와 b 모두를 가중치라고 할 때도 있다.

```
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
```

```
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 XOR 게이트

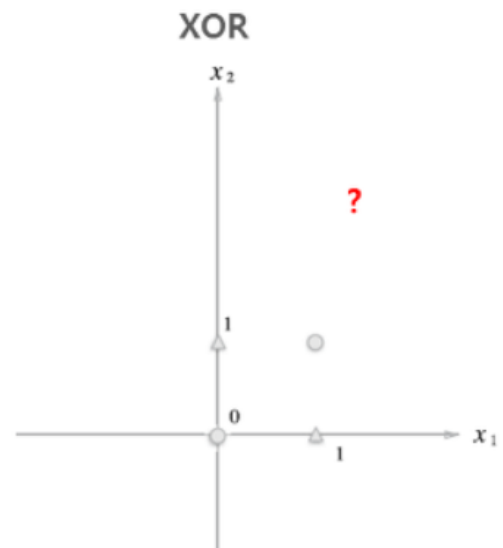
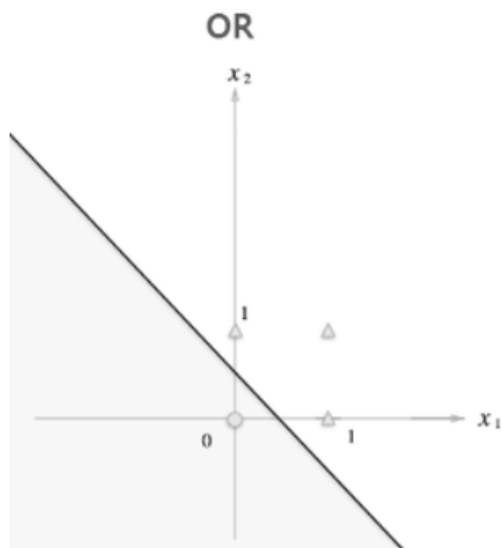
배타적 논리합이라는 논리 회로이다. x_1 과 x_2 중 한 쪽이 1일 때만 1을 출력한다.

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

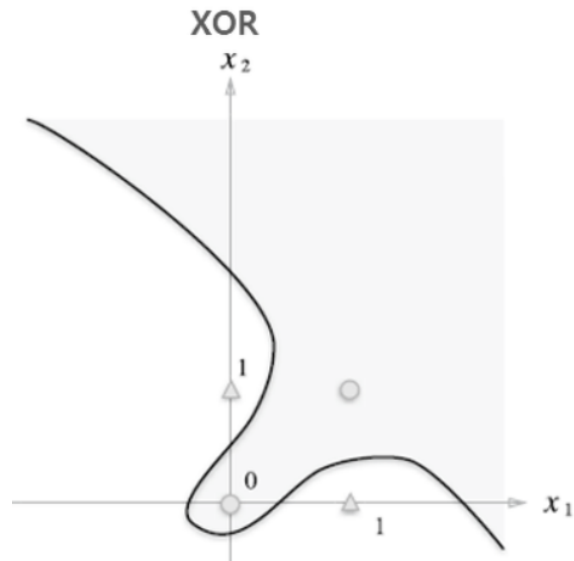
‘배타적’이란 자기 외에는 거부한다는 뜻

지금까지 본 퍼셉트론으로는 이 XOR게이트를 구현할 수 없다.

밑의 그림을 보면, XOR게이트는 하나의 직선으로 구분할 수 없다.



2.4.2 선형과 비선형

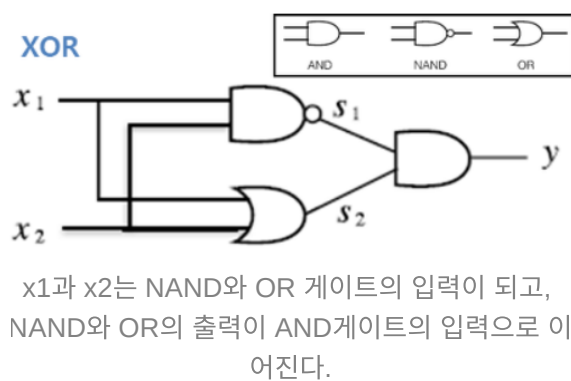


위와 같은 곡선의 영역을 비선형 영역이라고 한다.

2.5 다층 퍼셉트론이 출동한다면

2.5.1 기존 게이트 조합하기

XOR 게이트는 AND, NAND, OR 게이트를 조합해서 표현할 수 있다.



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

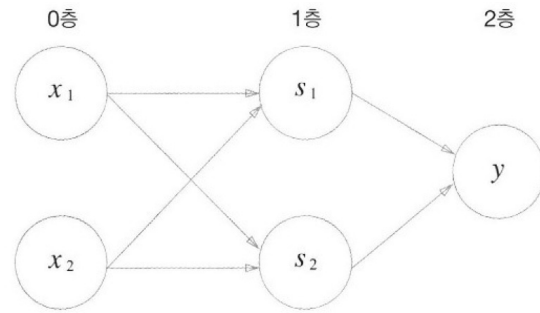
NAND의 출력이 s_1 , OR의 출력이 s_2 이다.

다층 퍼셉트론으로 비선형적인 표현이 가능하다.

2.5.2 XOR 게이트 구현하기

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

```
y = AND(s1, s2)
return y
```



XOR는 다층 구조의 네트워크, 즉 **층이 여러 개인 퍼셉트론: 다층 퍼셉트론**이다.

2.6 NAND부터 컴퓨터까지

NAND 게이트만으로 컴퓨터를 만들 수 있다.

2.7 정리

- 퍼셉트론은 입출력을 갖춘 알고리즘이다. 입력을 주면 정해진 규칙에 따른 값을 출력한다.
- 퍼셉트론에서는 ‘가중치’와 ‘편향’을 매개변수로 설정한다.
- 퍼셉트론으로 AND, OR 게이트 등의 논리 회로를 표현할 수 있다.
- XOR 게이트는 단층 퍼셉트론으로는 표현할 수 없다.
- 2층 퍼셉트론을 이용하면 XOR 게이트를 표현할 수 있다.
- 단층 퍼셉트론은 직선형 영역만 표현할 수 있고, 다층 퍼셉트론은 비선형 영역도 표현할 수 있다.
- 다층 퍼셉트론은 (이론상) 컴퓨터를 표현할 수 있다.