

# 연구지도 및 연구윤리 5주차 보고서

## (Chapter2. AND,NAND,OR,XOR 구현)

경북대학교 전자공학부  
2016113566 김남영

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



```
>>> AND(0,0)
0
>>> AND(0,1)
0
>>> AND(1,0)
0
>>> AND(1,1)
1
```

AND gate 구현

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



```
>>> NAND(0,0)
1
>>> NAND(0,1)
1
>>> NAND(1,0)
1
>>> NAND(1,1)
0
```

NAND gate 구현

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



```
>>> OR(0,0)
0
>>> OR(0,1)
1
>>> OR(1,0)
1
>>> OR(1,1)
1
```

OR gate 구현

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



```
>>> XOR(0,0)
0
>>> XOR(0,1)
1
>>> XOR(1,0)
1
>>> XOR(1,1)
0
>>>
```

XOR gate 구현

AND, NAND, OR gate를 도식적으로 표현해보면, 진리표를 만족하게끔 공간을 구분하는 직선을 그을 수 있다. 이때 이 직선을 표현하는 수식은 gate를 구현할 때 이용한 가중치와 편향식이다. 반면, XOR gate의 경우 도식적으로 (1,0)과 (0,1)이 속한 공간만 구분할 수 있는 직선이 존재하지 않는다. 하지만 위 그림처럼 NAND, OR, AND를 이용하여 구현할 수 있고, 이는 선형 영역의 조합을 통해 비선형 영역을 표현할 수 있다는 말이 된다.

이처럼 퍼셉트론을 다층으로 쌓은 경우, 비선형 영역을 표현할 수 있다는 것이 다층 퍼셉트론의 핵심이다. 데이터를 직선으로 분류하는 것보다 곡선으로 분류하는 것이 정확도가 당연히 높다. 따라서 다층 퍼셉트론을 이용한다면 높은 정확도의 데이터 분류, 학습, 예측이 가능할 것이다. 추가로, 위 gate들은 출력이 계단함수이므로 출력이 0과 1로 나타나지만, 이를 시그모이드 함수로 나타내면 0부터 1 사이의 모든 실수 값들이 출력 값으로 나타날 수 있으므로 좀 더 유연하게 데이터를 학습시킬 수 있다.