

Perceptron

from Deep Learning from Scratch
by Goki Saito

박성우

Sep 26, 2021





1. 퍼셉트론이란?
2. 퍼셉트론의 활용
3. 퍼셉트론의 한계와 해결

What is “Perceptron”?

- Definition

- ‘Threshold function’으로 불리는 이진 분류기(Binary classifier) 학습에 사용되는 알고리즘
- Threshold function (or Activation function)

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{w} \cdot \mathbf{x} \geq \theta \\ 0 & \text{o.w} \end{cases} \quad \text{or} \quad f(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{w} \cdot \mathbf{x} + \mathbf{b} \geq 0 \\ 0 & \text{o.w} \end{cases}$$

What is “Perceptron”?

- Definition

- ‘Threshold function’으로 불리는 이진 분류기(Binary classifier) 학습에 사용되는 알고리즘
- Threshold function (or Activation function)

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{w} \cdot \mathbf{x} \geq \theta \\ 0 & \text{o.w} \end{cases} \quad \text{or} \quad f(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{w} \cdot \mathbf{x} + b \geq 0 \\ 0 & \text{o.w} \end{cases}$$

Threshold

What is “Perceptron”?

- Definition

- ‘Threshold function’으로 불리는 이진 분류기(Binary classifier) 학습에 사용되는 알고리즘
- Threshold function (or Activation function)

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{w} \cdot \mathbf{x} \geq \theta \\ 0 & \text{o.w.} \end{cases} \quad \text{or} \quad f(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{w} \cdot \mathbf{x} + b \geq 0 \\ 0 & \text{o.w.} \end{cases}$$

The diagram illustrates the components of the perceptron equations. A box labeled "Weight" is connected by a line to the \mathbf{w} in the first equation. Another box labeled "Bias" is connected by a line to the b in the second equation.

What is “Perceptron”?

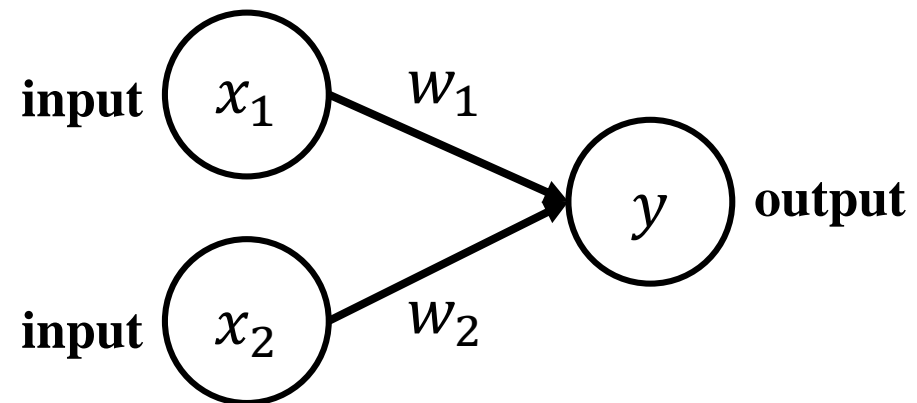
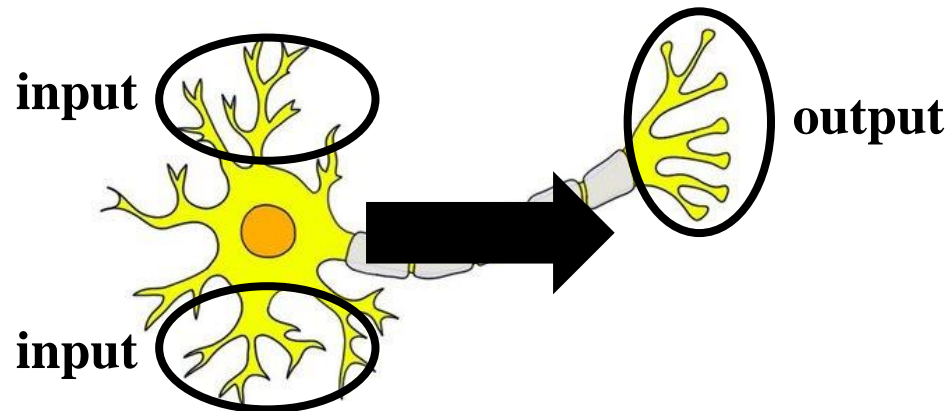
- Definition

- ‘Threshold function’으로 불리는 이진 분류기(Binary classifier) 학습에 사용되는 알고리즘
- Threshold function (or Activation function)

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{w} \cdot \mathbf{x} \geq \theta \\ 0 & \text{o.w} \end{cases} \quad \text{or} \quad f(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{w} \cdot \mathbf{x} + \mathbf{b} \geq 0 \\ 0 & \text{o.w} \end{cases}$$

- Features

- ‘선형 분류기(Linear Classifier)’
- Input은 여러 개를 받을 수 있지만, output은 0 또는 1의 값만 가질 수 있음
- 인간의 신경세포인 뉴런(neuron)과 형태적 특징이 비슷해 ‘인공 뉴런’이라고도 불림



What is “Perceptron”?

- Linear Classifier?

- Linear

- 일차함수 $f(x) = ax + b$ 를 영어로 ‘Linear function’; 함수 그래프가 직선 형태!
 - 이 개념을 스칼라(scalar)에서 벡터(vector)로 확장한 개념
 - Scalar: ‘방향은 가지지 않고 크기만 가지는 물리량’ 또는 쉽게 생각해 하나의 실수
 - Vector: ‘방향과 크기를 모두 가지는 물리량’ 또는 쉽게 생각해 실수들의 모음 또는 n 개의 행과 1개의 열로 이루어진 행렬 ($n \geq 1$)
 - $f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + \mathbf{b}$ where $\mathbf{w}, \mathbf{x}, \mathbf{b} \in \mathbb{R}^n$
 - 이때 $\mathbf{w} \cdot \mathbf{x}$ 의 연산법을 ‘선형 결합(Linear Combination)’이라고 하며 연산 결과를 ‘내적(Inner Product or Dot Product)’이라고 함
 - Linear combination $\mathbf{w} \cdot \mathbf{x} = w_1x_1 + w_2x_2 + \cdots + w_nx_n = \sum_{i=1}^n w_ix_i$

What is “Perceptron”?

- Linear Classifier?

- Classifier

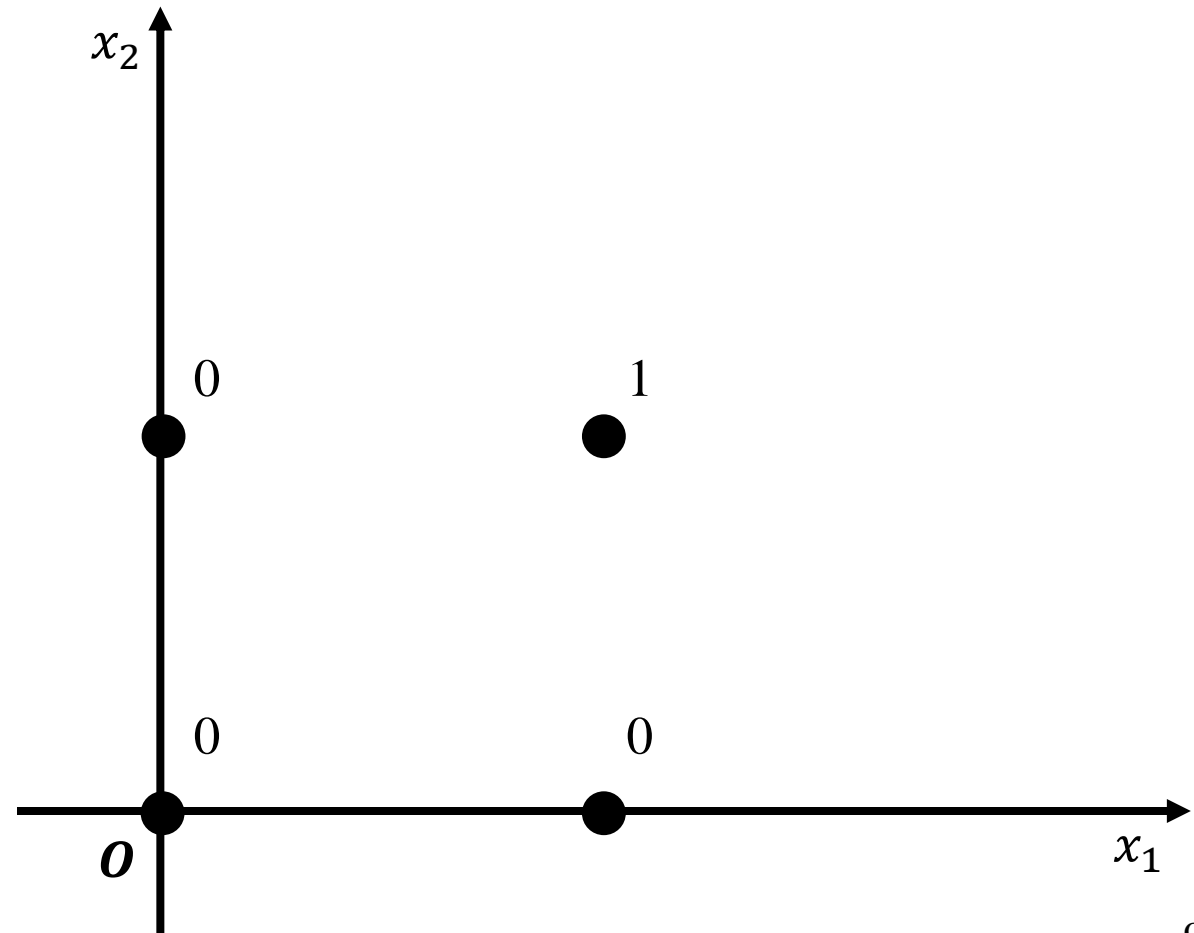
- 어떤 변수를 input으로 받았을 때, 해당 변수가 어떤 클래스에 속하는지를 도출하는 함수
 - 따라서 최종 함수의 결과값은 항상 discrete
 - Discrete: 값들이 뚝뚝 떨어져 있어서, ‘셀 수 있는’ 수들의 집합 ex) 자연수, 정수
 - Continuous: 값들을 서로 떨어뜨릴 수 없고 따라서 ‘셀 수 없는’ 수들의 집합 ex) 실수
 - 그런데 input은 continuous한 값들이 주어질 수도 있고, 특히 선형함수를 취했을 때 그 계산 결과가 항상 discrete하다고 장담할 수 없음
 - 따라서 최종 결과를 도출하기 전에, 계산 결과를 discrete하게 만들어주는 연산이 필요
 - 그 연산을 수행하는 함수를 ‘Threshold function(Activation function)’이라고 하며
 - 어떤 클래스에 속하는지 판단하는 데 사용되는 기준을 Threshold라고 함

Application – AND / NAND / OR Gate

- AND Gate

- 두 개의 변수에 대해서, 두 값이 서로 1일 때에만 1(또는 True)을 도출하며, 그렇지 않은 경우 0 (또는 False)를 도출하는 논리 연산자

x_1	x_2	$x_1 \wedge x_2$
0	0	0
0	1	0
1	0	0
1	1	1

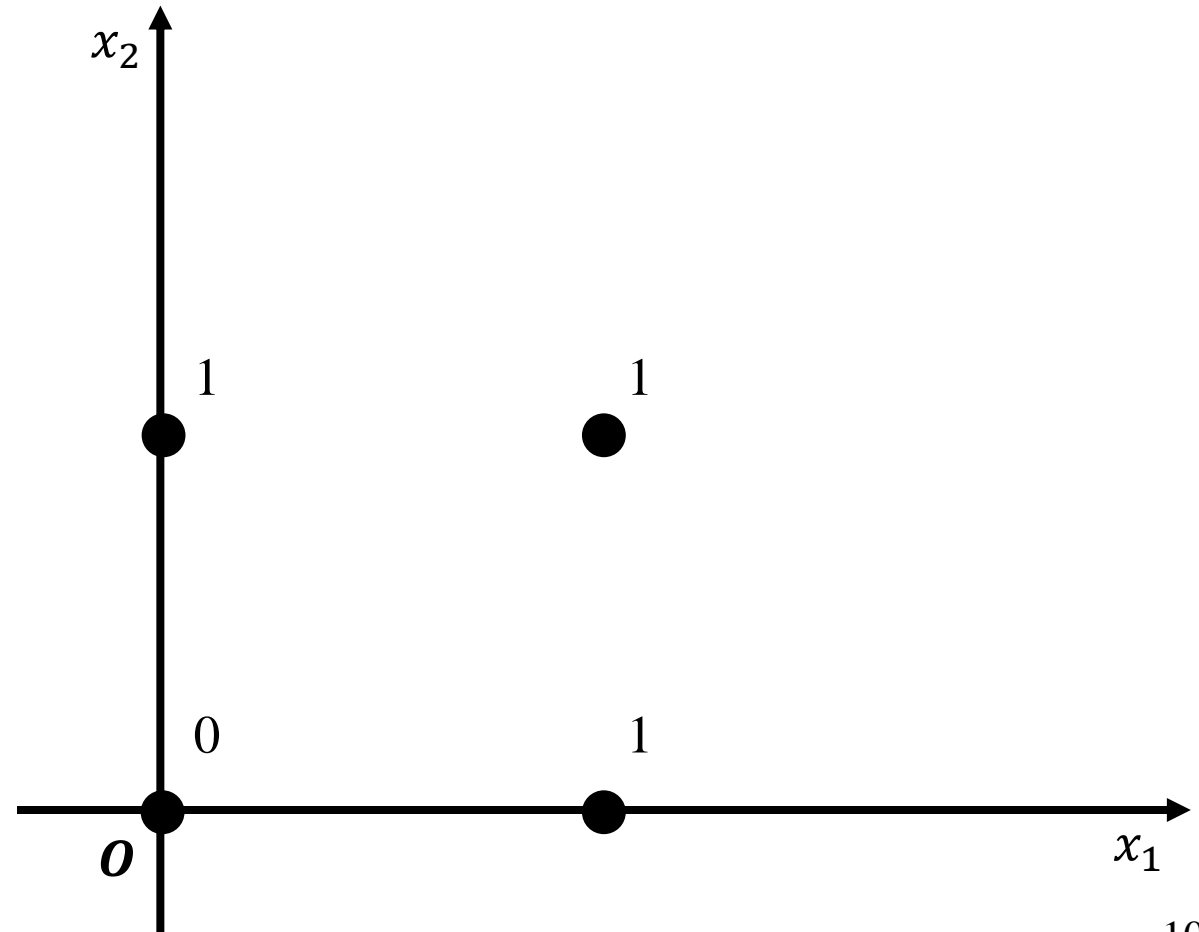


Application – AND / NAND / OR Gate

- OR Gate

- 두 개의 변수에 대해서, 두 값 중 하나만 1이면 1(또는 True)를 도출하는 논리 연산자

x_1	x_2	$x_1 \vee x_2$
0	0	0
0	1	1
1	0	1
1	1	1

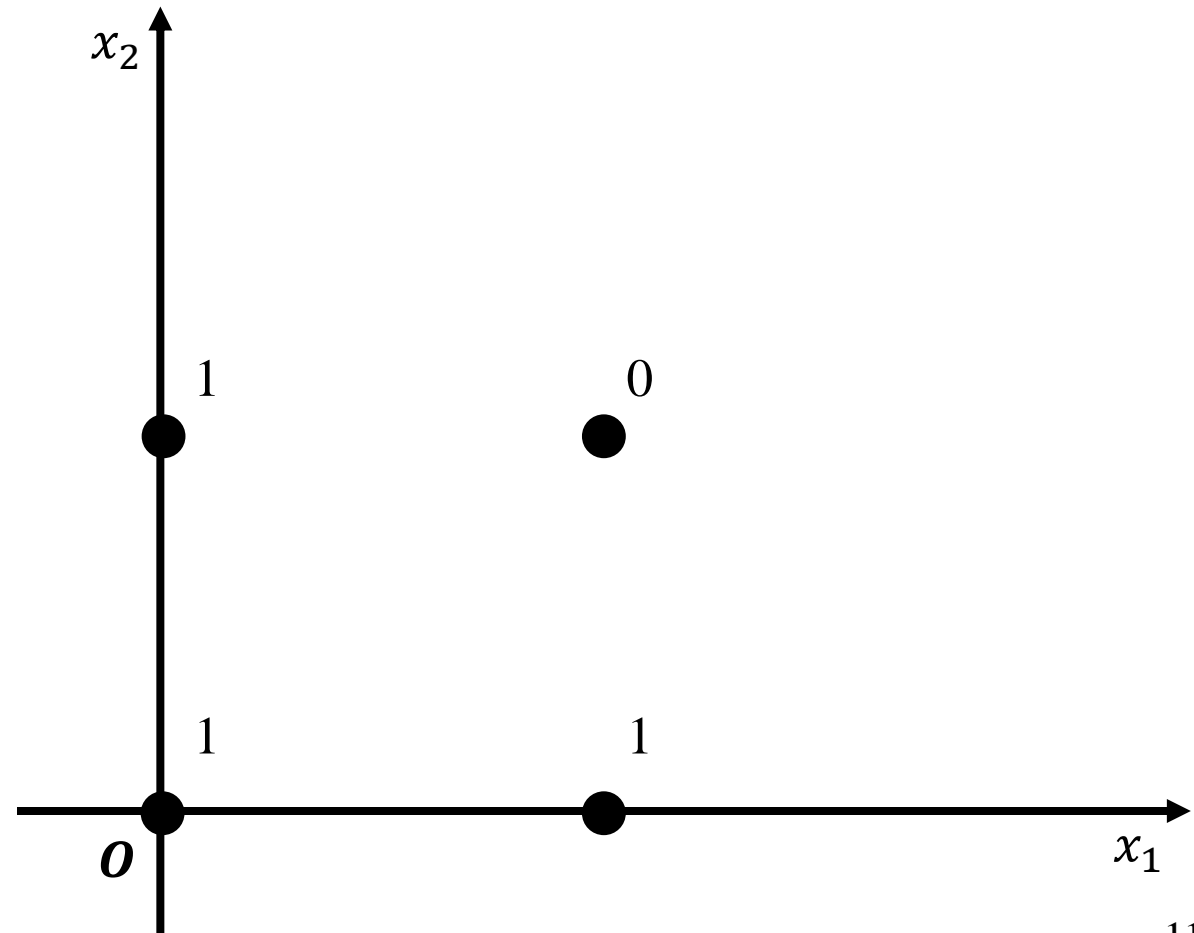


Application – AND / NAND / OR Gate

- NAND Gate

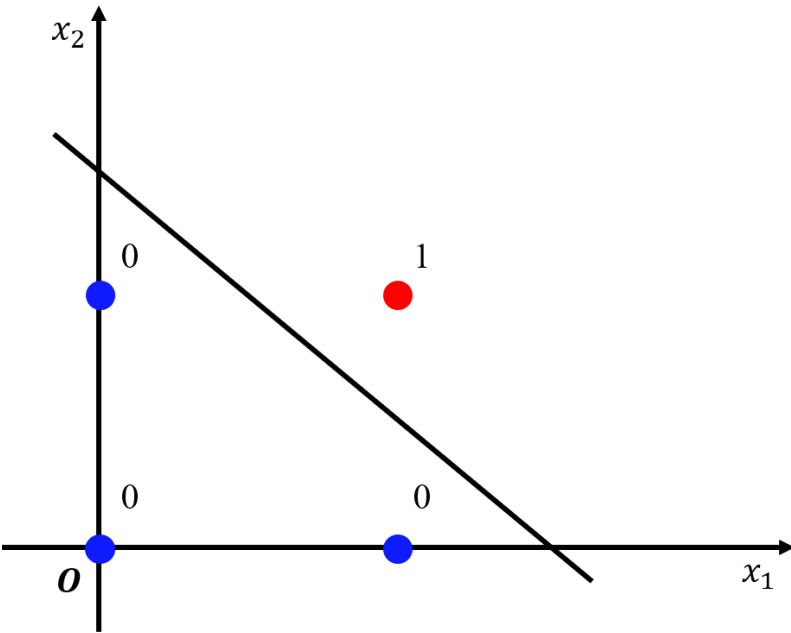
- Not AND, 즉 AND 연산자의 정확히 반대 결과를 도출하는 연산자

x_1	x_2	$\neg(x_1 \wedge x_2)$
0	0	1
0	1	1
1	0	1
1	1	0



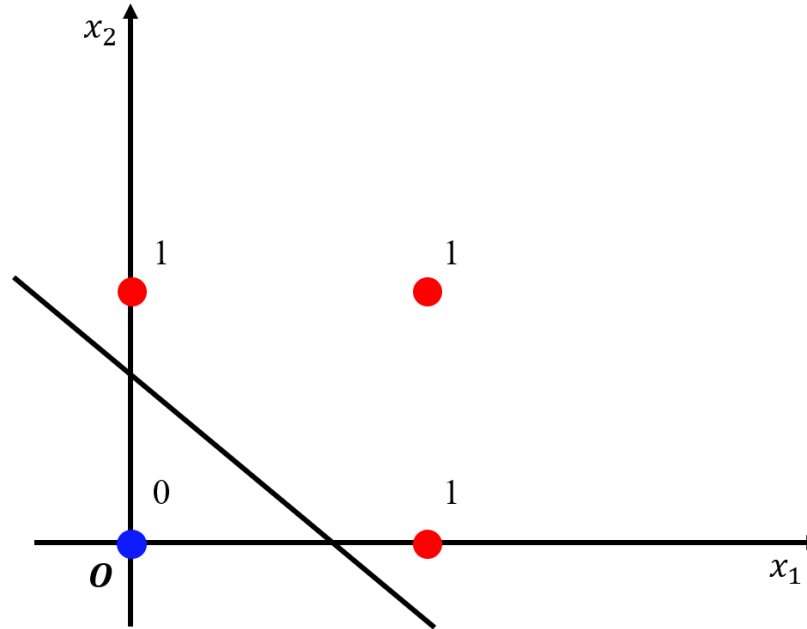
Application – AND / NAND / OR Gate

- 모두 Perceptron (Linear Classifier)으로 해결할 수 있음!



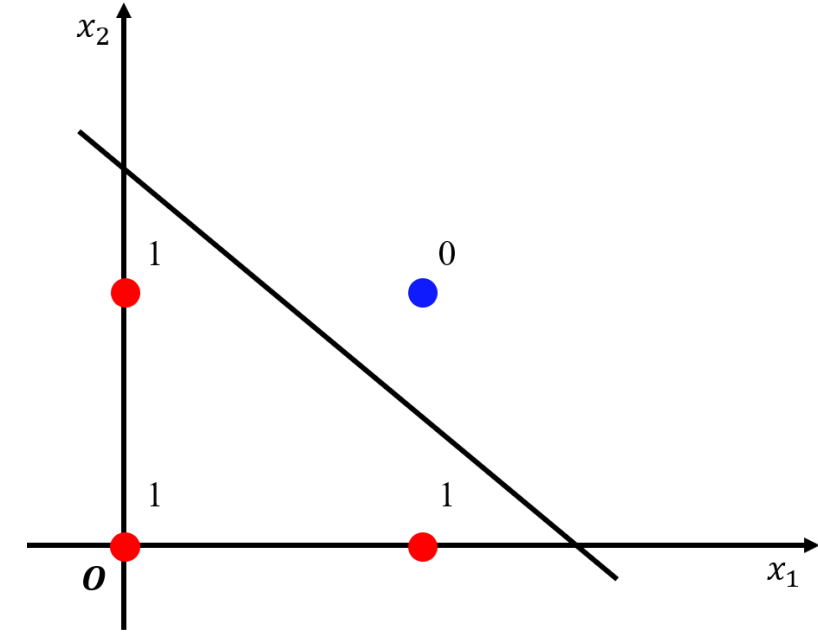
AND

$$0.5x_1 + 0.5x_2 \geq 0.7$$
$$w = \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}, \theta = 0.7$$



OR

$$0.6x_1 + 0.4x_2 \geq 0.3$$
$$w = \begin{pmatrix} 0.6 \\ 0.4 \end{pmatrix}, \theta = 0.3$$



NAND

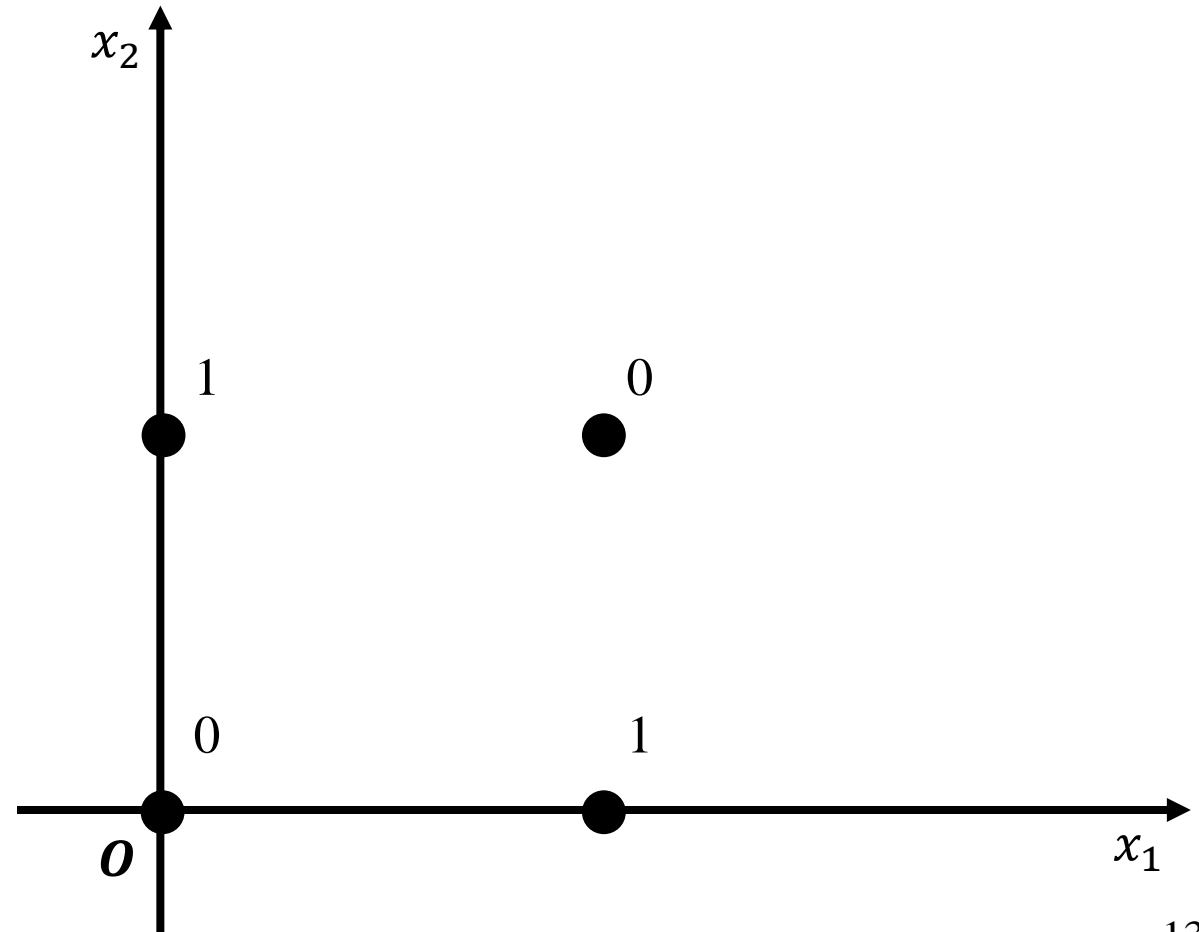
$$-0.5x_1 - 0.5x_2 \geq -0.7$$
$$w = \begin{pmatrix} -0.5 \\ -0.5 \end{pmatrix}, \theta = -0.7$$

Limitation and Improvement

- XOR Gate

- 두 개의 변수에 대해서, 서로 다르면 1(또는 True), 서로 같으면 0(또는 False)을 도출하는 논리 연산자

x_1	x_2	$x_1 \oplus x_2$
0	0	0
0	1	1
1	0	1
1	1	0

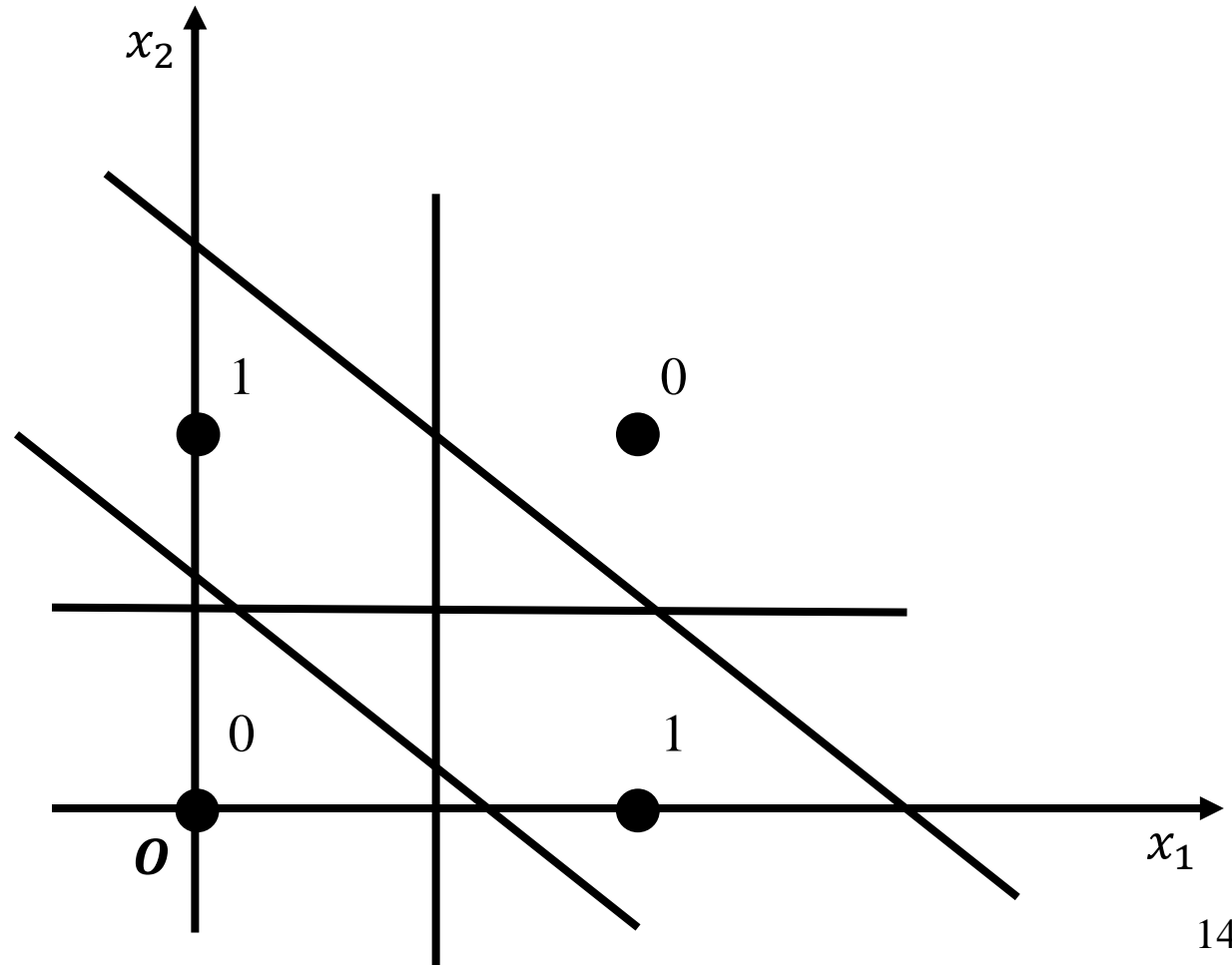


Limitation and Improvement

- XOR Gate

- 하나의 Perceptron(Single-layer Perceptron)으로 해결할 수 없음!

x_1	x_2	$x_1 \oplus x_2$
0	0	0
0	1	1
1	0	1
1	1	0



Limitation and Improvement

- XOR Gate

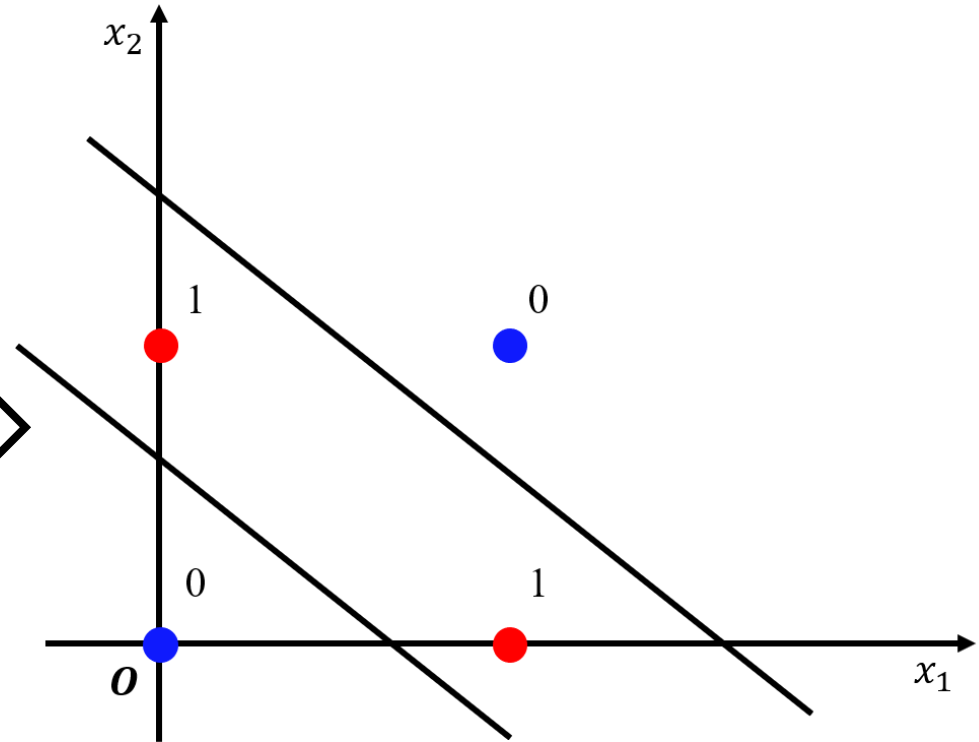
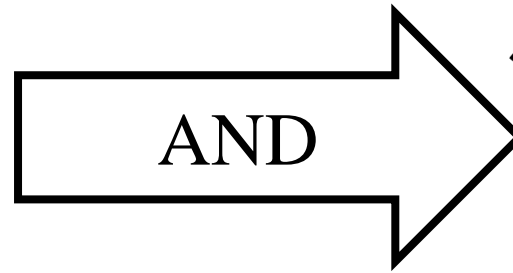
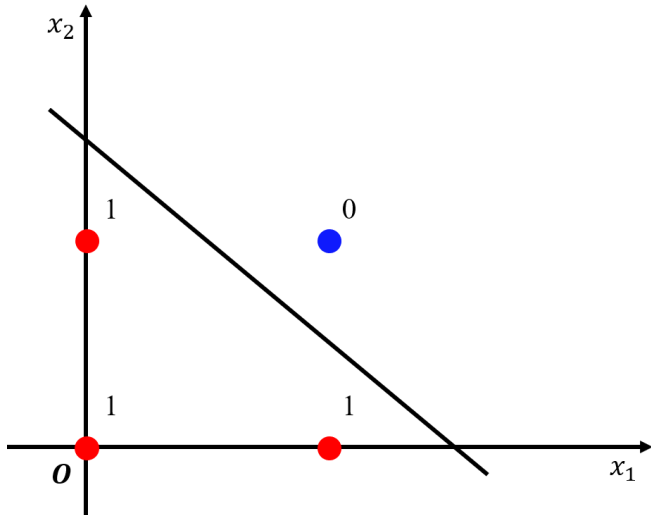
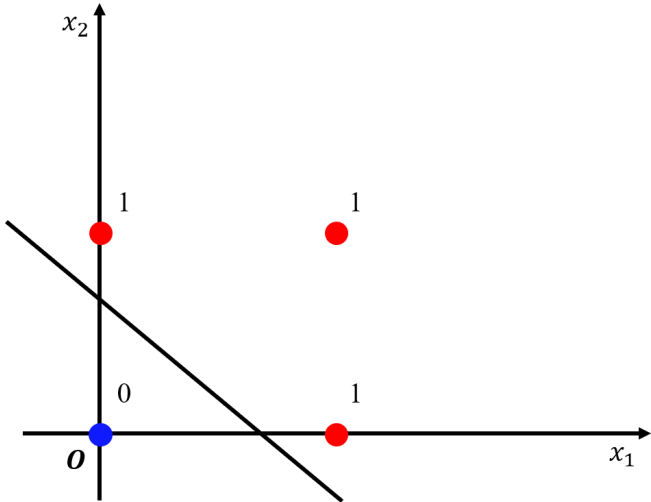
- 두 개의 Perceptron을 사용함으로써 해결 가능!

x_1	x_2	$s_1 = (x_1 \vee x_2)$	$s_2 = \neg(x_1 \wedge x_2)$	$s_1 \wedge s_2 = x_1 \oplus x_2$
0	0	0	1	0
0	1	1	1	1
1	0	1	1	1
1	1	1	0	0

Limitation and Improvement

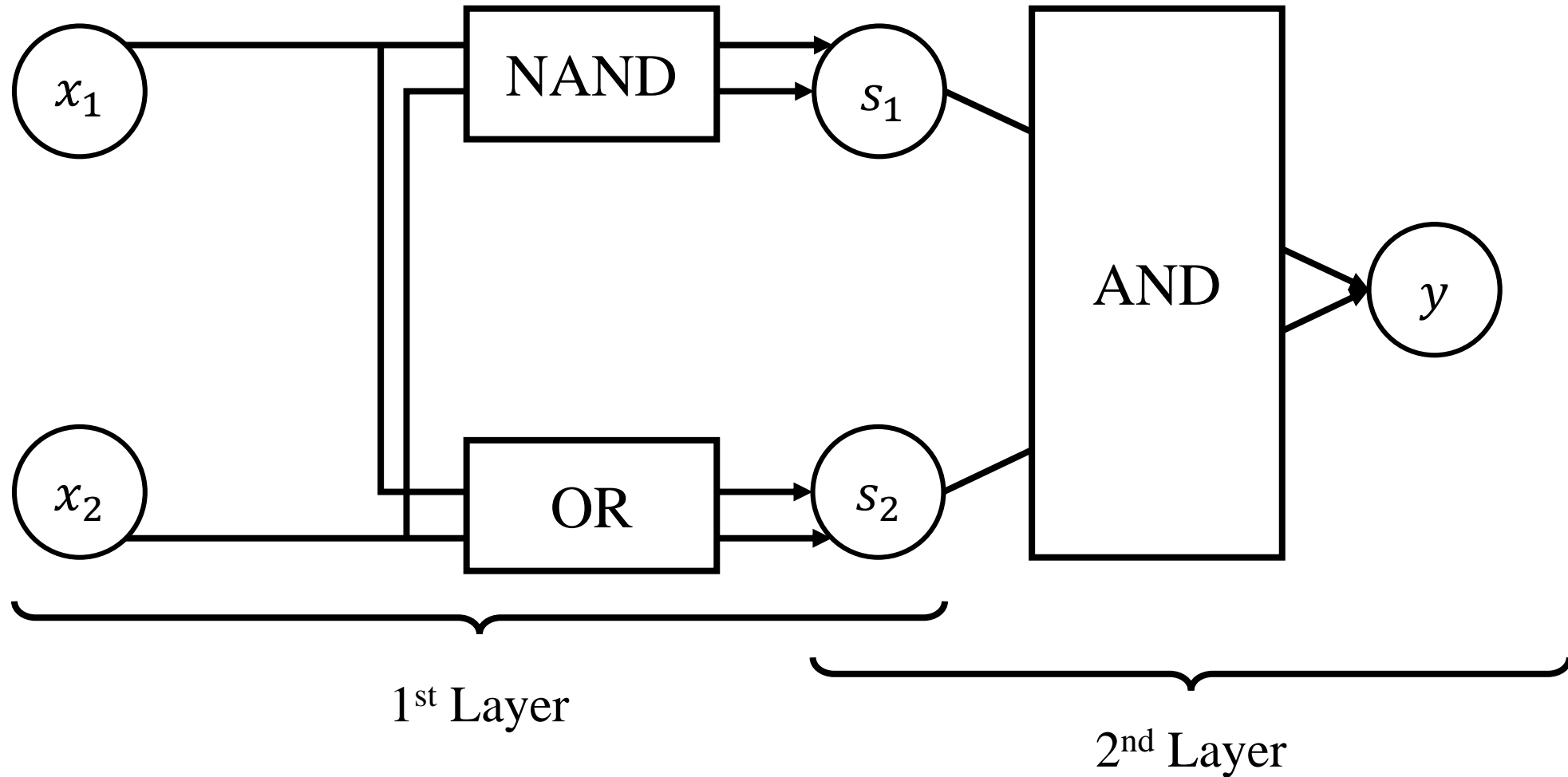
- XOR Gate

- 두 개의 Perceptron을 사용함으로써 해결 가능!



Limitation and Improvement

- XOR Gate
 - 두 개의 Perceptron을 사용함으로써 해결 가능!



Conclusion

- Perceptron은 다수의 input을 받아 해당 input을 0 또는 1의 값으로 할당하는 이진 분류기 (Binary Classifier)이다.
- 이때 input을 그대로 사용하는 것이 아니라 weight이라는 벡터와 선형결합한 형태로 변환 (transform)하여 사용하며
 - 때로는 이 선형결합에 bias라고 하는 term을 더해주기도 한다.
- 그리고 이렇게 선형결합된 값을 0 또는 1이라는 값으로 mapping해주는 함수로 Threshold function이라는 함수를 사용하며,
 - 0 또는 1로 분류해주는 기준을 threshold라고 한다.
- Perceptron은 여러 층으로 쌓을 수 있으며, 다층으로 쌓은 Multi-layer Perceptron은 단층인 Single-layer Perceptron이 해결할 수 없는 여러 문제를 해결할 수 있다.
 - 그리고 Neural Network를 비롯한 모든 Deep Learning은 이 Perceptron으로부터 출발한다.

Thank You