

# 신경망의 기본

퍼셉트론, MLP



# 목차

- 01      퍼셉트론의 개념**
- 02      MLP의 등장**
- 03      과적합, 과소적합**
- 04      Keras 실습**



## | 퍼셉트론 개념

### 가장 기이초적인 퍼셉트론의 기능방식

입력:  $x = [x_1, x_2, x_3, \dots, x_n]$  (1차원 실수 벡터)

가중치:  $w = [w_1, w_2, w_3, \dots, w_n]$  (1차원 실수 벡터)

편향:  $b$  (스칼라)

연산:  $\sum(w_i \times x_i) + b = w_1x_1 + w_2x_2 + \dots + w_nx_n + b$

**퍼셉트론은 현대 딥러닝의 모든 모델들이  
기반으로 하는 '핵심 기술'**

MLP를 스포츠가, CNN을 승용차 등이라고 비유한다면,  
퍼셉트론은 자동차의 '바퀴'와 같은 역할

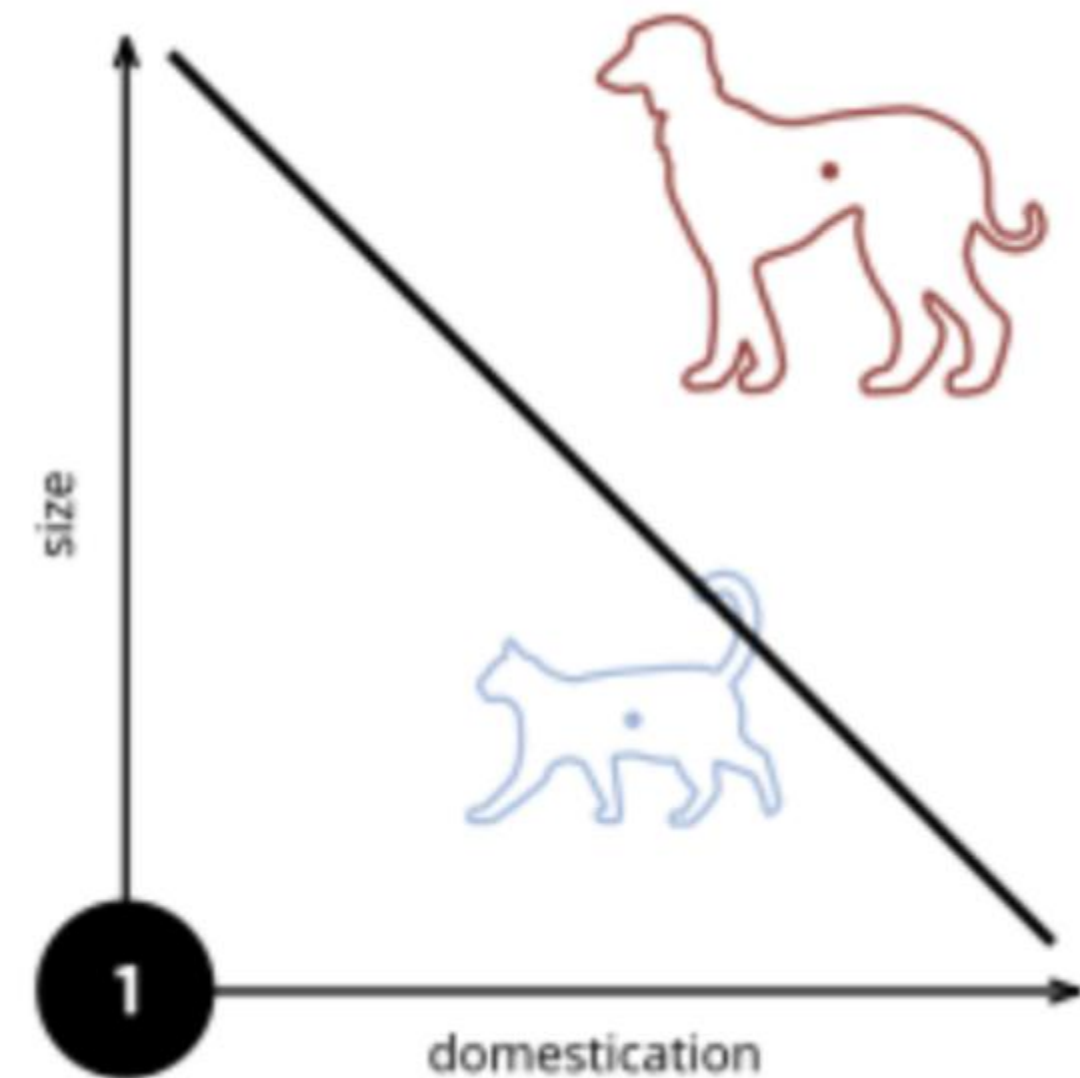
입력값을 받고 -> 가중치를 곱 -> 활성화함수 통과

- 모든 모델들이 기초로 거치는 연산

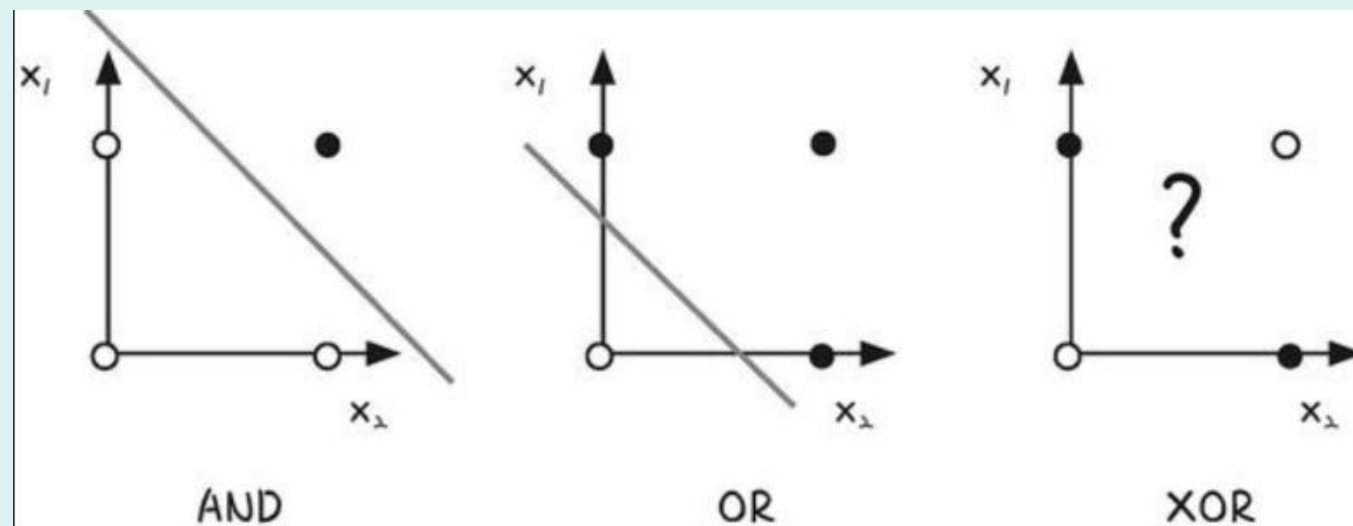
# 예시: 개 찾아내기

## 2차원 선형 분류

- 2D에서 직선 방정식:  $ax_1 + bx_2 + c = 0$ 
  - a, b는 가중치, c는 편향
- x들은 '개'를 나타내는 특징 벡터



# 한계



2차원 XOR문제

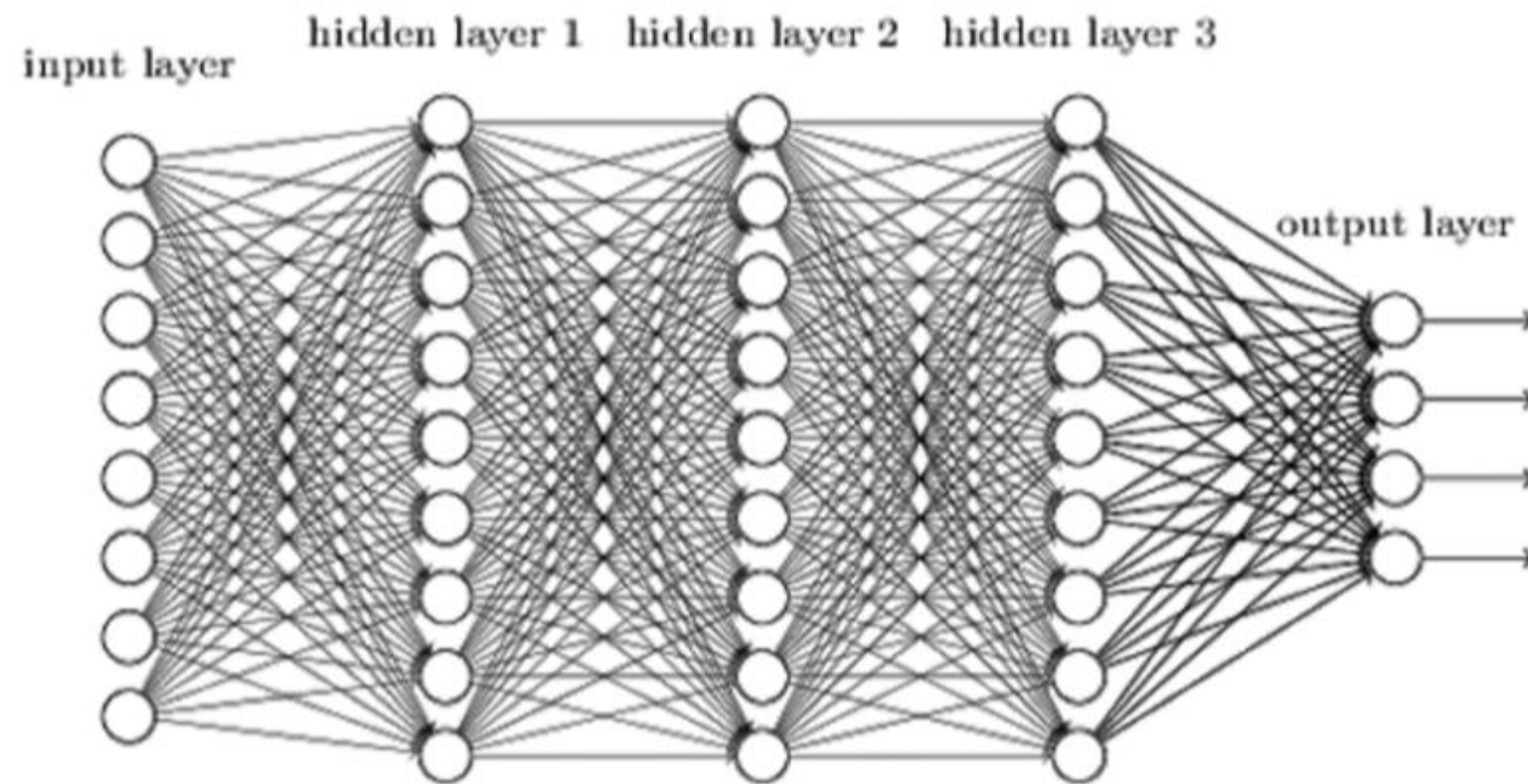
\*3차원 참고

## 선형 경계선으로 해결 불가

- 경계선에 '비선형성' 필요
- MLP의 등장, '활성화 함수'



# MLP의 등장



## 입력층

- 각 뉴런마다 원래의 데이터를 수치화한 특성값으로서 실수 형태로 하나씩 들어감
- 이미지 기준 총 8개의 특성값으로 데이터가 나타내어진 것

## 은닉층

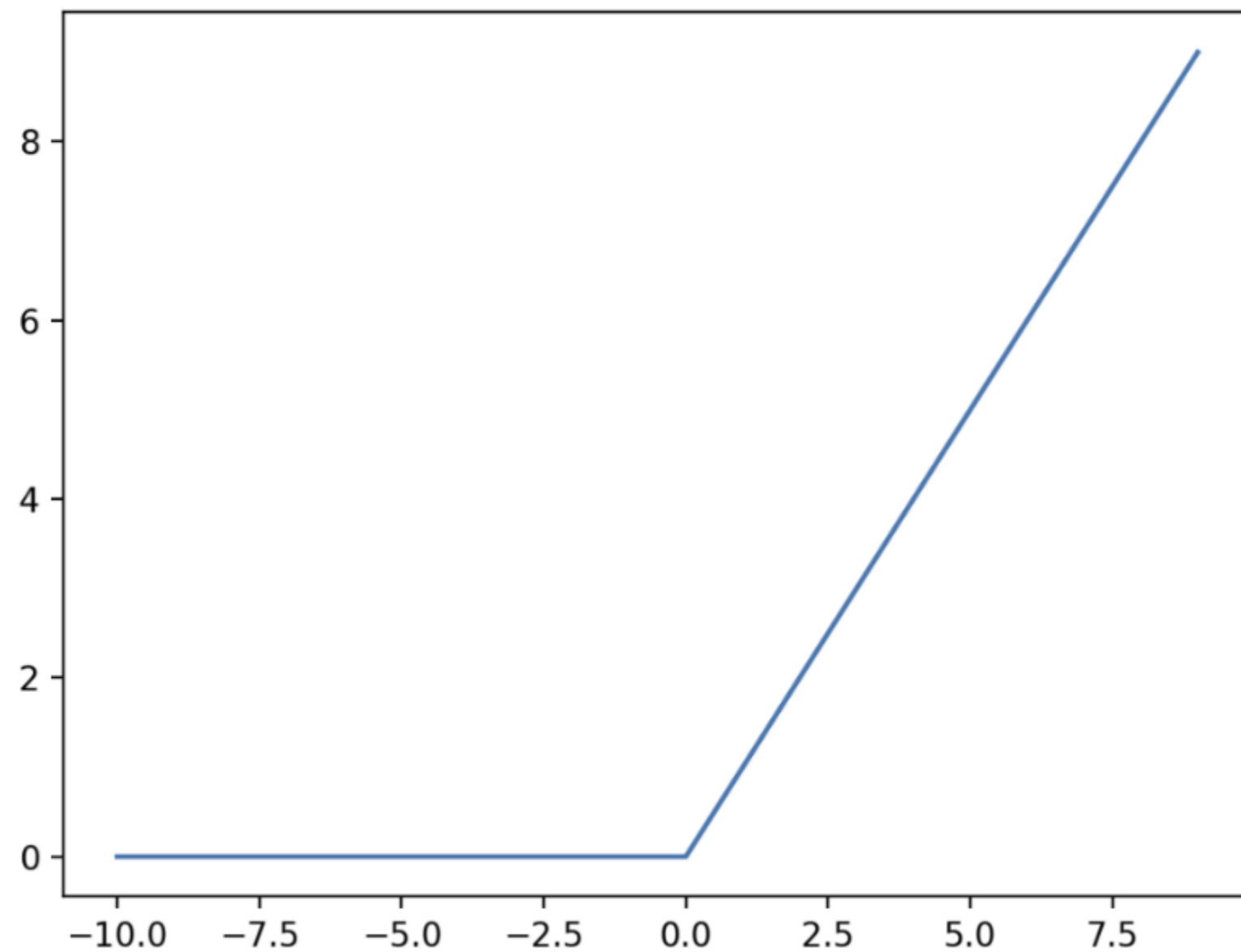
- 여러개의 층이, 층마다 여러개의 뉴런이 존재
- 층이 깊어질수록 더 구체적이고 자세한 특성 학습
- 뉴런 하나하나가 아까 봤던 퍼셉트론의 기능을 함

## 출력층

- 최종 예측값을 알아내는
- 분류문제라면 분류 클래스만큼 뉴런 수를 가짐  
(10클래스의 이미지분류라면 출력층의 뉴런은 10개)



# ReLU



수식:  $f(x) = \max(0, x)$

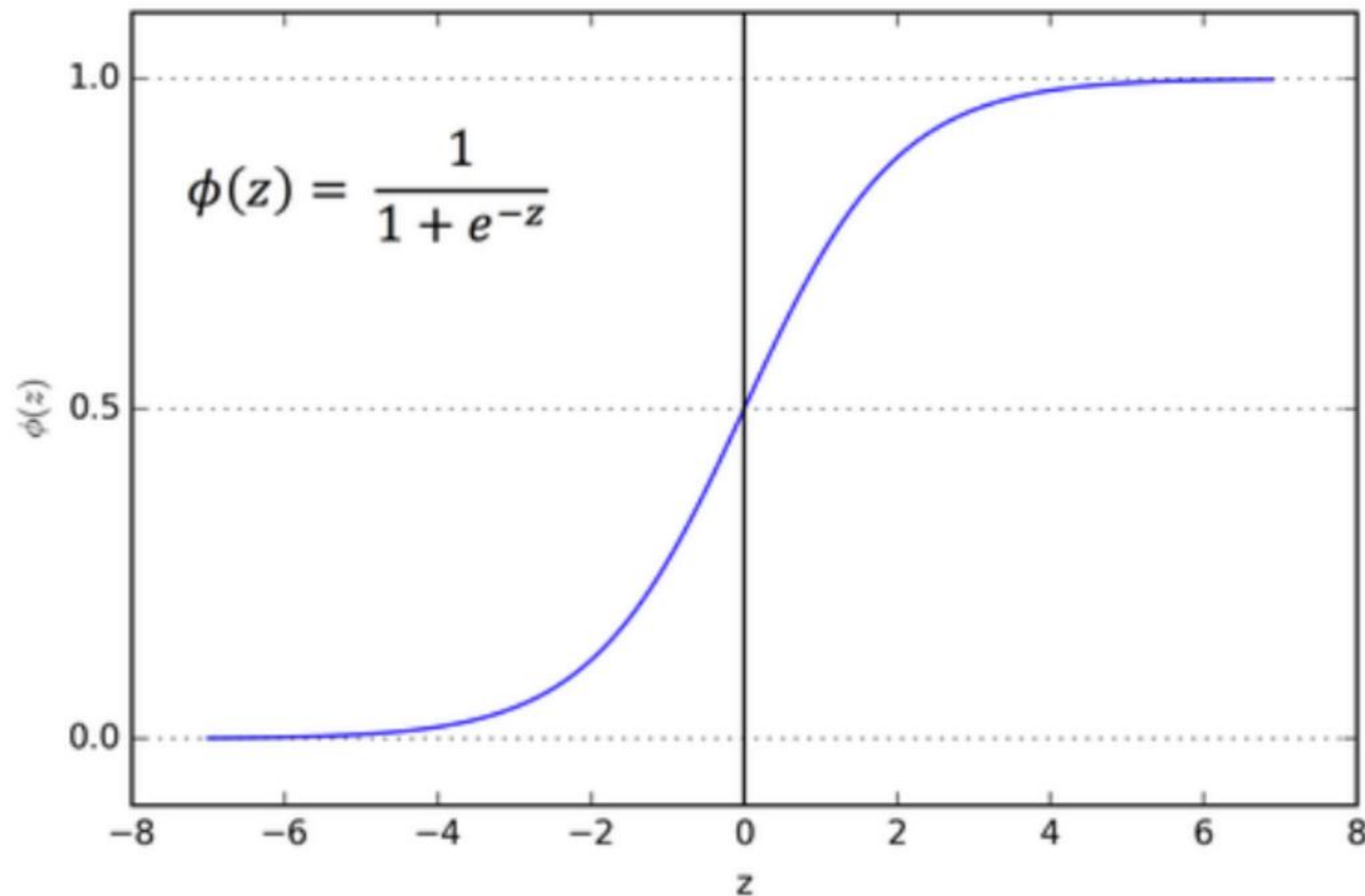
- x가 양수면 x를 그대로 출력
- x가 0 이하면 0을 출력

특징

- 은닉층에서 주로 사용: 현재 가장 표준적인 선택
- CNN, RNN 등 모든 신경망 구조에서 널리 활용
- 출력층에서는 거의 사용하지 않음
  - 음수 값을 출력할 수 없기 때문



# Sigmoid



수식:  $\sigma(x) = 1 / (1 + e^{(-x)})$

- 지수계산 때문에 연산비용이 높음

특징

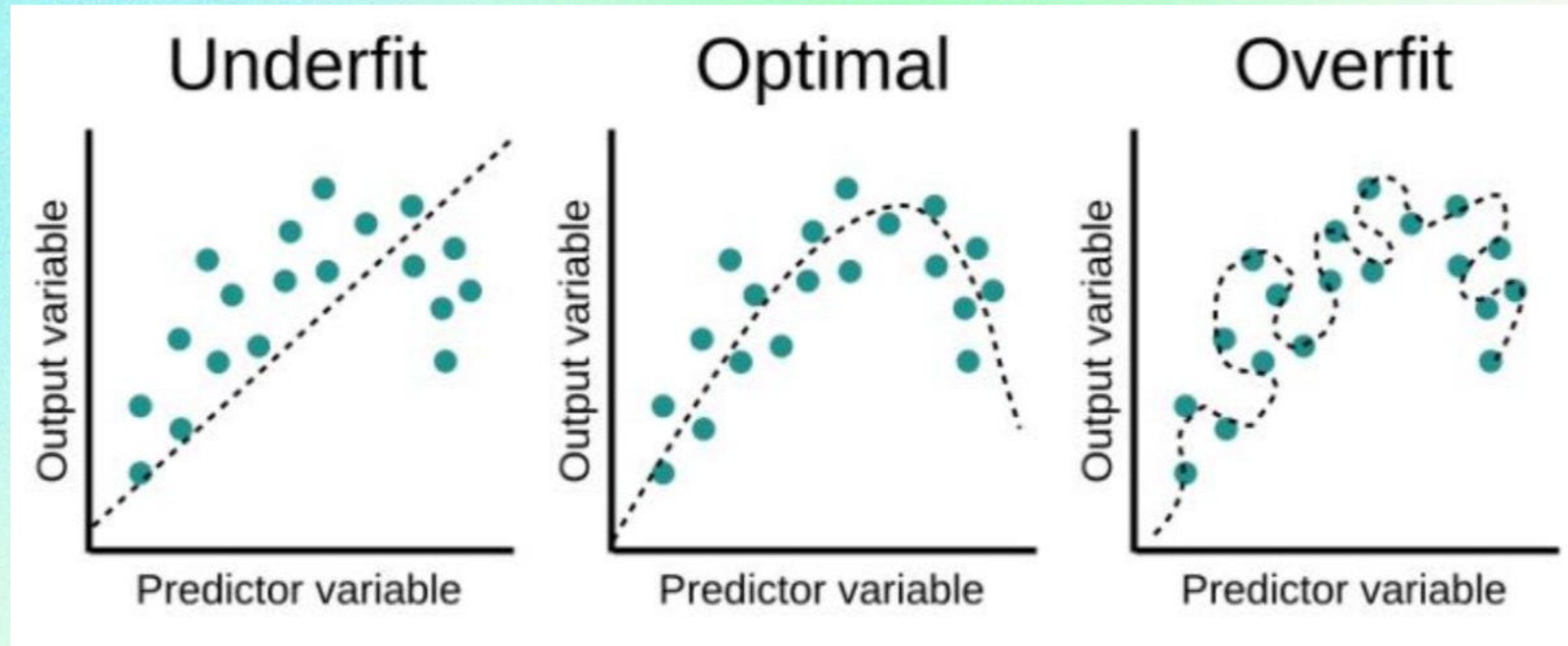
- 출력값의 범위가 정해져있음(0-1) - 수치적 안정성

- 이진분류 문제에 용이하게 사용됨 - 0~1의 값을 확률로서 해석이 가능

- 은닉층에선 많이 안쓰임(ReLU 많이씀)



# 과적합, 과소적합



## 과적합(overfitting)

- 모델이 엄청 유연해서 학습 데이터는 귀신같이 잘 분류하지만, 다른 데이터를 넣어봤을 때는 제대로 성능을 발휘하지 못하는 경우
- 너무나 학습데이터에만 완벽한 선형모델

## 과소적합(underfitting)

- 모델의 학습이 부족하고, 제대로 학습되지 못해서 모델의 학습결과가 너무 단순한 경우
- 그냥 분류를 잘 못하는 비 선형적이고 복잡하지 않은 모델



# **Keras - XOR 문제 해결, MNIST 실습**