딥러닝의 통계적이해

3강. 딥러닝 모형의 구조와 학습(1)

1. 다층신경망의 구조
2. 활성화 함수
3. 다층신경망의 표현
4. 신경망의 학습

한국방송통신대 이긍희 교수

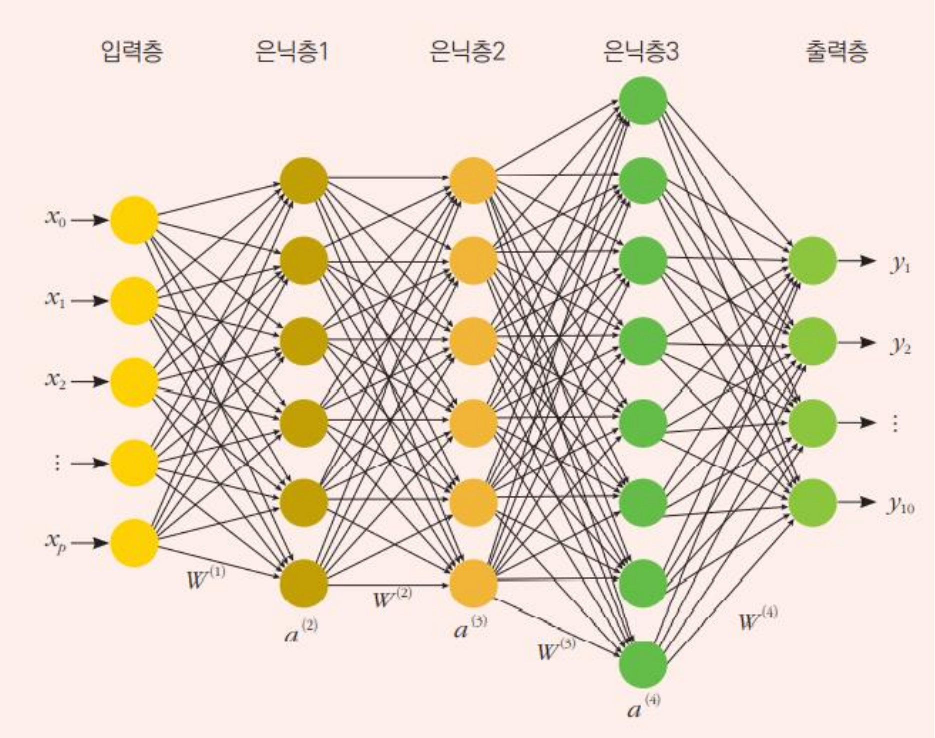
오늘의 학습목표

1. 다층신경망의 구조를 이해한다.
2. 활성화 함수를 이해한다.
3. 다층 신경망의 학습을 이해한다.

1. 다층신경망의 구조

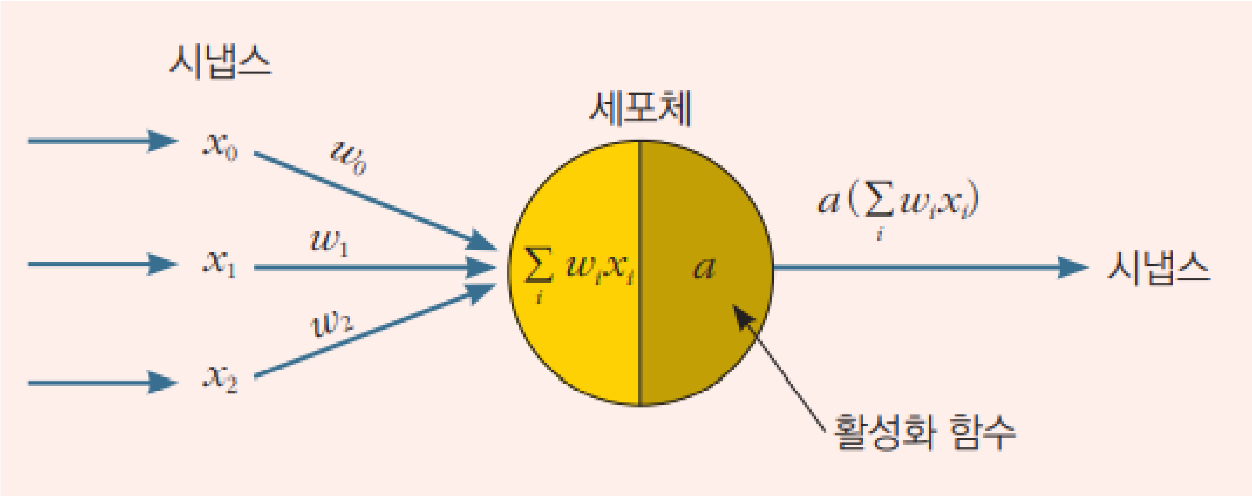
다층신경망

- 입력층, 출력층 사이 은닉층이 포함된 신경망
  → 다층 퍼셉트론(MLP) → 딥러닝(deep learning)



다층신경망

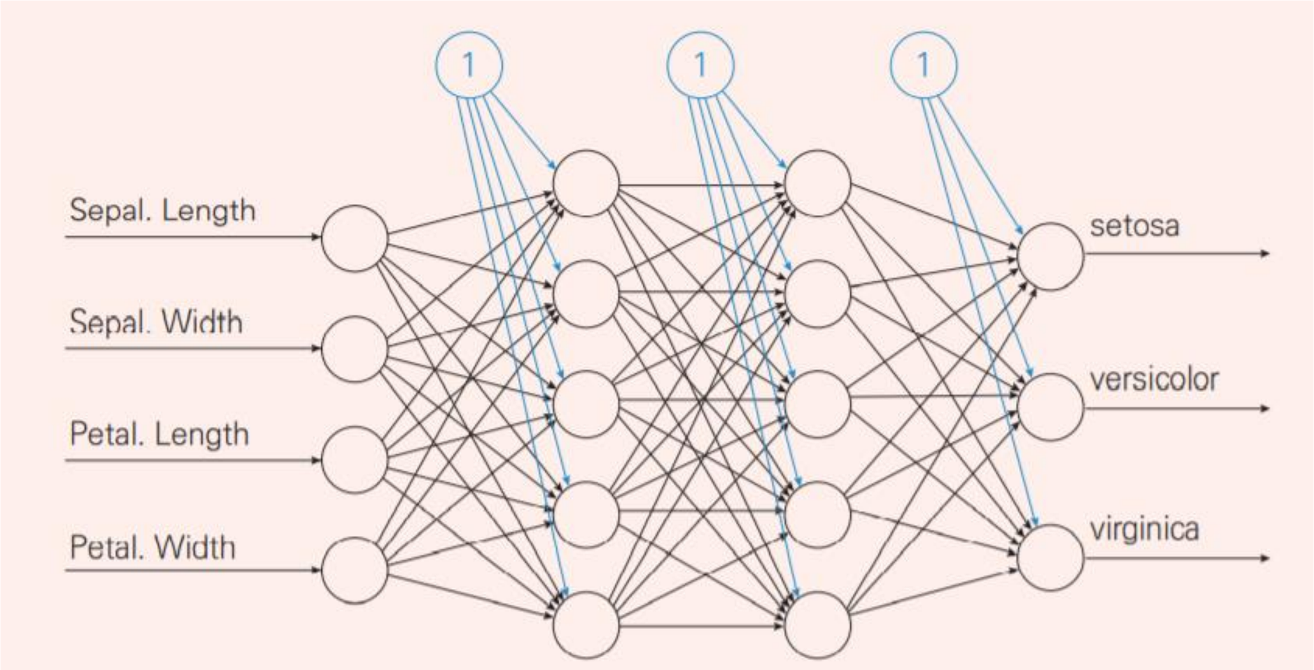
- 입력층에서 은닉층을 통과할 때마다 데이터들이 가중 결합된 후 활성화 함수가 적용되어 출력층에 이름
  - 손실함수 최소로 하는 가중치를 구해서 신경망을 작성



다층신경망의 예

◆ 붓꽃 종의 분류를 다층신경망을 이용

- 동그란 부분이 뉴런(또는 노드), 연결된 선이 네트워크



순방향신경망

- 순방향신경망(Feedforward Neural Network, FNN) : 같은 층 내에서는 연결되지 않고 앞의 층으로만 연결

- 뉴런이 위층의 뉴런과 모두 연결됨 → 완전연결망 (Fully Connected Network, FCN)이라고 부름

다층신경망과 합성함수

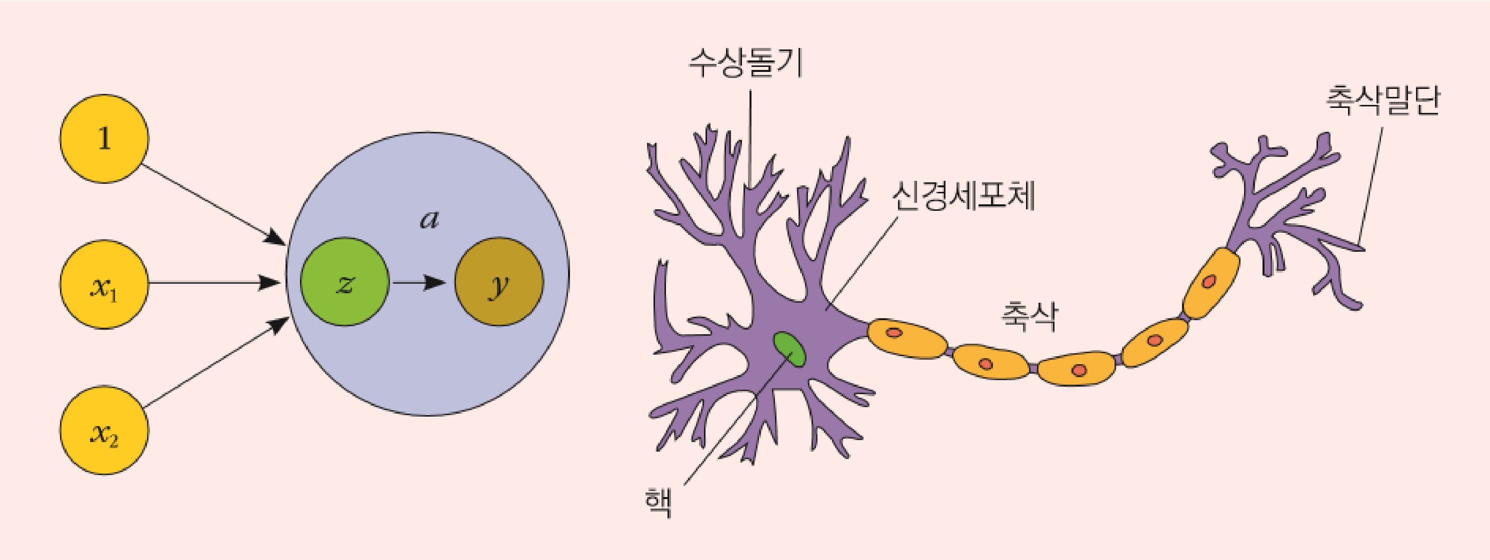
- 다층신경망 : 합성함수로 표현
  - 함수의 합성이 반복 → 신경망의 목적 함수의 표현력이 좋아짐

$$y = f^{(3)}(f^{(2)}(f^{(1)}(x)))$$

2. 활성화 함수

활성화 함수

- 활성화 함수(activation function) : 시냅스 구현 함수
  - 뉴런 정보가 시냅스로 이동 → 화학물질 이용, 전기 정보가 임계값을 넘었을 때 활성화 → 다른 뉴런으로 전달



활성화 함수의 종류

◆ 활성화 함수의 종류

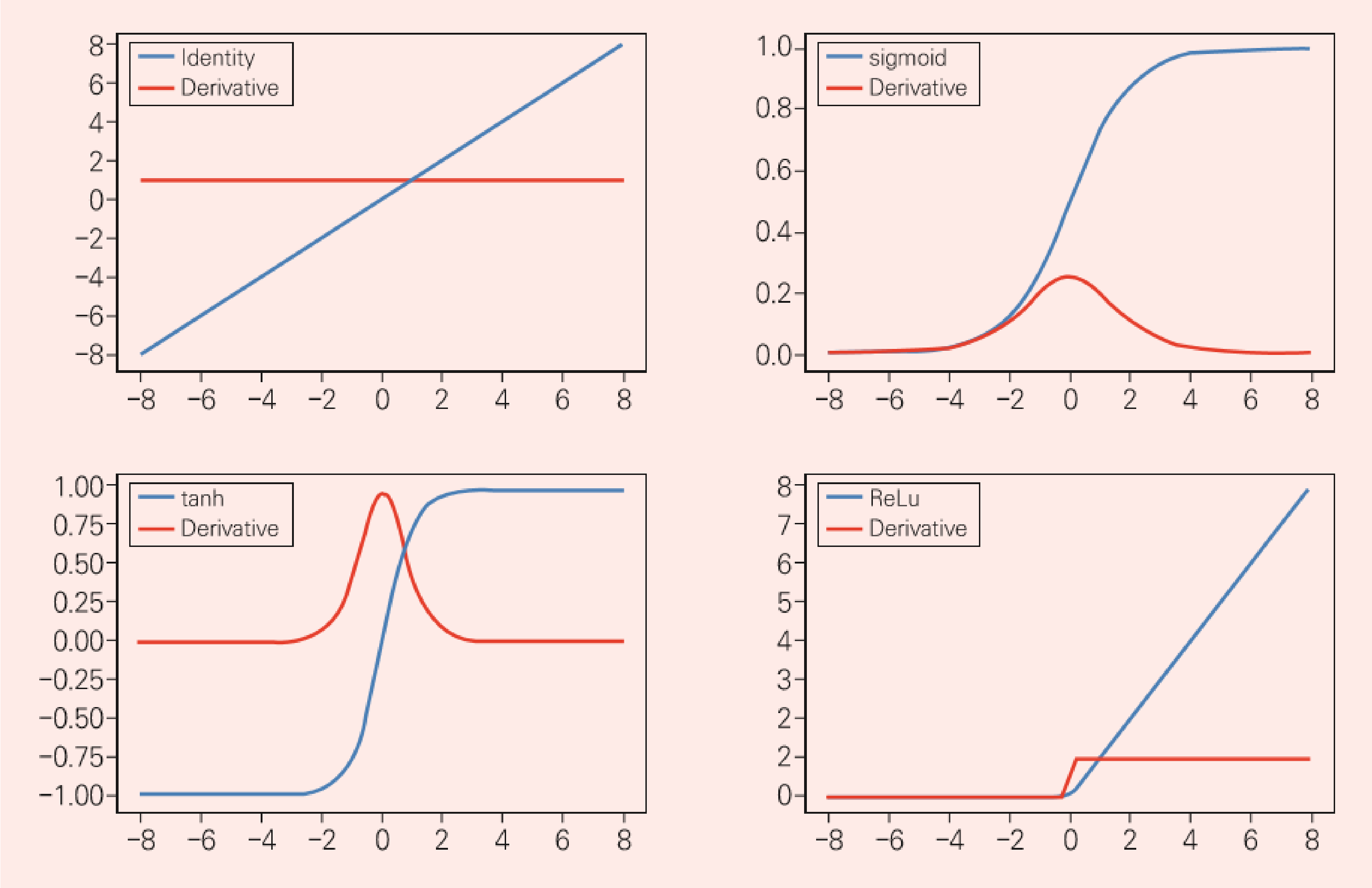
항등함수 : $a(x) = x$

시그모이드 함수 : $a(x) = \dfrac{1}{1+e^{-x}}$

tanh 함수 : $a(x) = \dfrac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$

ReLU 함수 : $a(x) = \max(x, 0)$

활성화 함수의 종류


Identity
Derivative
sigmoid
Derivative
tanh
Derivative
ReLu
Derivative
−8 −6 −4 −2 0 2 4 6 8
1.0 0.8 0.6 0.4 0.2 0.0
1.00 0.75 0.50 0.25 0.00 −0.25 −0.50 −0.75 −1.00
8 7 6 5 4 3 2 2 0

활성화 함수의 미분

◆ 활성화 함수의 미분

항등함수 : $a(x)=x$

시그모이드 함수 : $a(x)=\dfrac{1}{1+e^{-x}}$

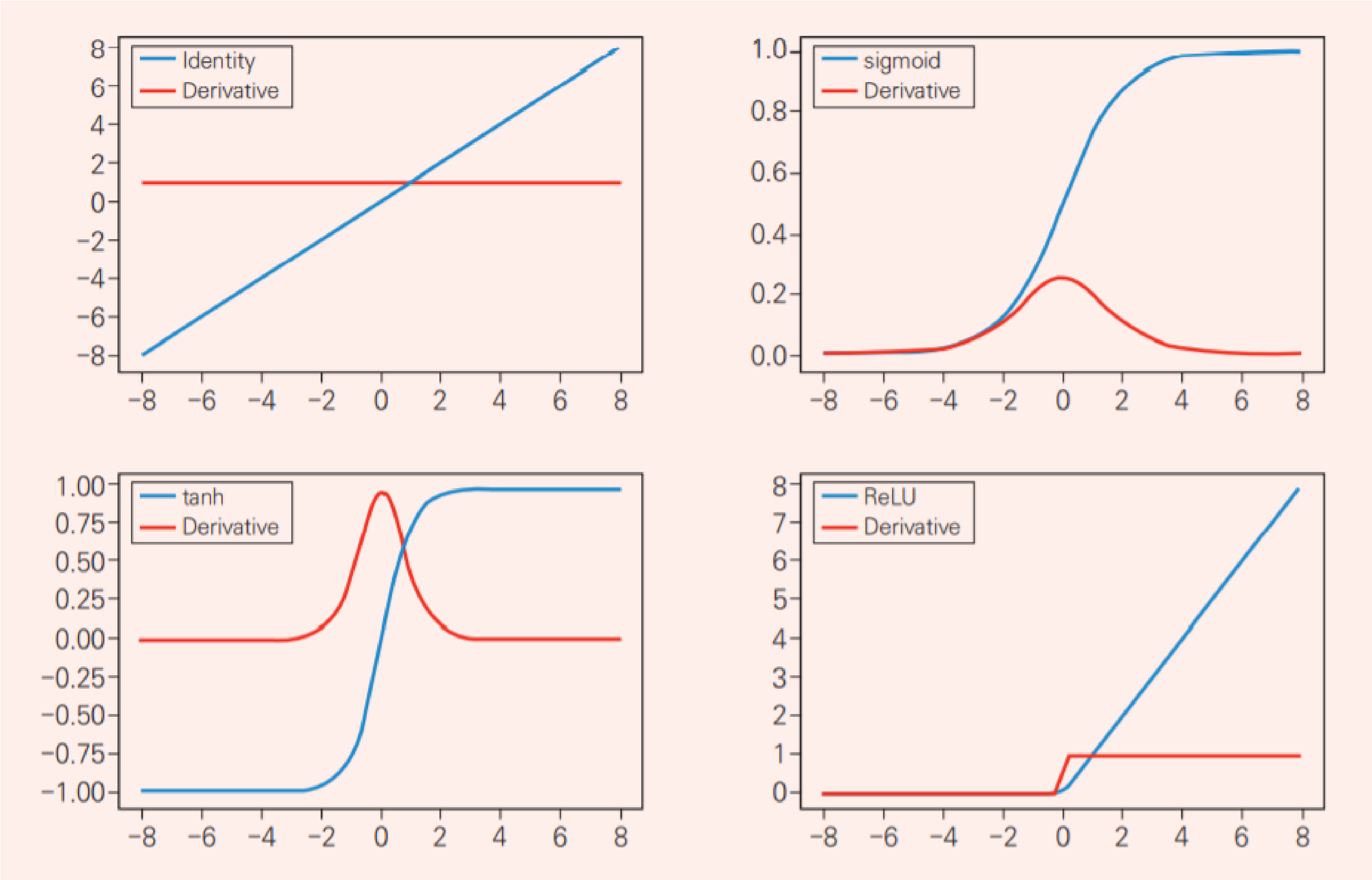
활성화 함수의 미분

◆ 활성화 함수의 미분

tanh 함수 : $a(x)=\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$

ReLU 함수 : $a(x)=\max(x, 0)$

활성화 함수의 미분



Identity
Derivative
sigmoid
Derivative
tanh
Derivative
ReLU
Derivative

활성화 함수의 미분

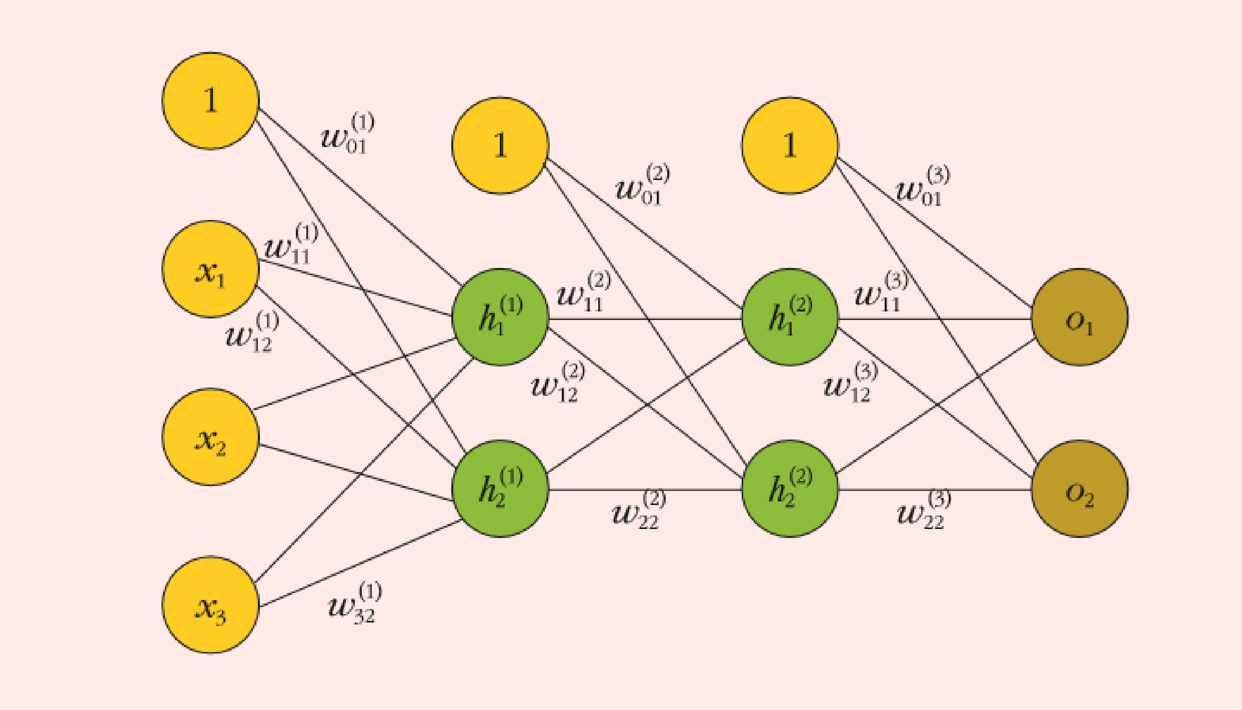
- 선형 함수 연속 적용 → 활성화 함수를 중복 적용해도 선형 함수
  → 시그모이드, ReLU 함수 비선형 함수로 표현가능

- 미분의 용이성으로 시그모이드 함수와 tanh 함수를 오랫동안 이용 → 경사 소실(gradient vanishing) 문제
  - 최근 딥러닝에서는 ReLU 함수가 주로 이용
    - · 단, ReLU 함수는 0에서 미분이 불가능하다는 제약

3. 다층신경망의 표현

다층신경망

- 뉴런이 3개인 입력층, 뉴런이 2개인 은닉층이 2층, 뉴런이 2개인 출력층으로 구성된 다층 신경망



다층신경망의 표현

- 뉴런이 3개인 입력층, 뉴런이 2개인 은닉층이 2층, 뉴런이 2개인 출력층으로 구성된 다층 신경망

가중합 : $z_j^{(1)} = w_{0j}^{(1)} + w_{1j}^{(1)} x_1 + w_{2j}^{(1)} x_2 + w_{3j}^{(1)} x_3 = \sum_{i=0}^{3} w_{ij}^{(1)} x_i$

활성화 함수 적용 : $h_j^{(1)} = a(z_j^{(1)}) = a(\sum_{i=0}^{3} w_{ij}^{(1)} x_i)$

다층신경망의 표현

- 뉴런이 3개인 입력층, 뉴런이 2개인 은닉층이 2층, 뉴런이 2개인 출력층으로 구성된 다층 신경망

제1 은닉층 : $h_{j_1}^{(1)} = a\left(\sum_{i=0}^{3} w_{ij_1}^{(1)} x_i\right),\ j_1 = 1, 2$

제2 은닉층 : $h_{j_2}^{(2)} = a\left(\sum_{j_1=0}^{2} w_{j_1 j_2}^{(2)} h_{j_1}^{(1)}\right),\ j_2 = 1, 2$

출력층 : $o_{j_3} = a_o\left(\sum_{j_2=0}^{2} w_{j_2 j_3}^{(3)} h_{j_2}^{(2)}\right),\ j_3 = 1, 2$

다층신경망의 행렬 표현

- 뉴런이 3개인 입력층, 뉴런이 2개인 은닉층이 2층, 뉴런이 2개인 출력층으로 구성된 다층 신경망

$$X=[1, x_1, x_2, x_3],\ h^{(l)}=[1, h_1^{(l)}, h_2^{(l)}],\ O=[o_1, o_2]$$

$$W^{(1)}=\begin{bmatrix} w_{01}^{(1)} & w_{02}^{(1)} \\ w_{11}^{(1)} & w_{12}^{(1)} \\ w_{21}^{(1)} & w_{22}^{(1)} \\ w_{31}^{(1)} & w_{32}^{(1)} \end{bmatrix},\ W^{(2)}=\begin{bmatrix} w_{01}^{(2)} & w_{02}^{(2)} \\ w_{11}^{(2)} & w_{12}^{(2)} \\ w_{21}^{(2)} & w_{22}^{(2)} \end{bmatrix},\ W^{(3)}=\begin{bmatrix} w_{01}^{(3)} & w_{02}^{(3)} \\ w_{11}^{(3)} & w_{12}^{(3)} \\ w_{21}^{(3)} & w_{22}^{(3)} \end{bmatrix}$$

다층신경망의 행렬 표현

- 뉴런이 3개인 입력층, 뉴런이 2개인 은닉층이 2층, 뉴런이 2개인 출력층으로 구성된 다층 신경망

제1 은닉층 : $Z^{(1)}=XW^{(1)}$,

$$h^{(1)}=[1, a(Z^{(1)})]=[1, h_1^{(1)}, h_2^{(1)}]$$

제2 은닉층 : $Z^{(2)}=h^{(1)}W^{(2)}$,

$$h^{(2)}=[1, a(Z^{(2)})]=[1, h_1^{(2)}, h_2^{(2)}]$$

출력층 : $Z^{(3)}=h^{(2)}W^{(3)}$, $O=[a_o(Z^{(3)})]=[o_1, o_2]$

출력층의 활성화 함수

- 회귀모형 : 항등함수
- 이진분류 : 시그모이드 함수
- 다중분류 : 소프트맥스 함수

신경망의 구조 설계

◆ 은닉층의 수와 각 층별 뉴런의 수를 정하는 것
- 층이 깊어질수록 신경망이 보다 추상적으로 입력 데이터를 파악→ 데이터의 표현력이 좋아짐
- 층별 뉴런의 수를 늘려도 신경망의 성과가 크게 높아지지는 않음

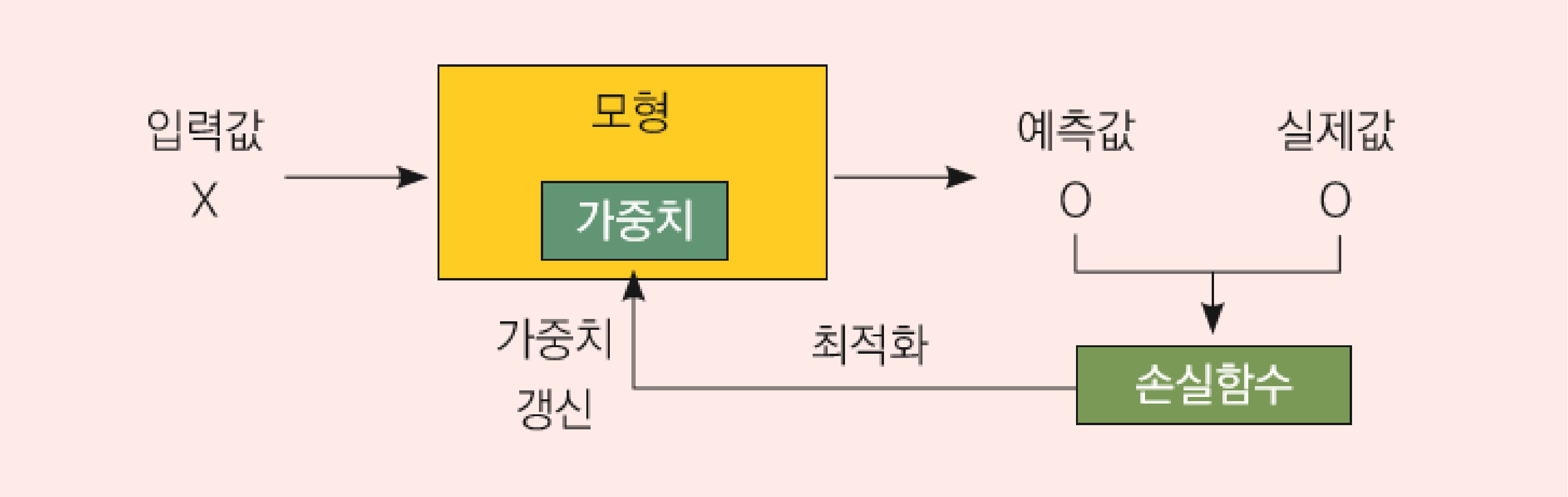
일반근사정리

- 충분한 크기의 뉴런을 가진 은닉층이 한 개 이상인 다층 신경망은 모든 유계의 연속함수를 근사할 수 있다.
  - 충분히 많은 뉴런을 가진 은닉층 1개인 신경망으로도 모든 함수를 표현
  - 신경망이 딥러닝 모형으로 발전하는데 방해 요인

4. 신경망의 학습

신경망의 작성과정

◆ 신경망의 작성과정



신경망의 작성과정

- 순방향 : 입력층 뉴런 → 은닉층 뉴런 → … → 출력 뉴런
  - 역방향 : 손실함수 $J(w)$ 기반 경사하강법을 통해 반복적으로 가중치들 갱신

$$w := w - \eta \frac{\partial}{\partial w} J(w)$$

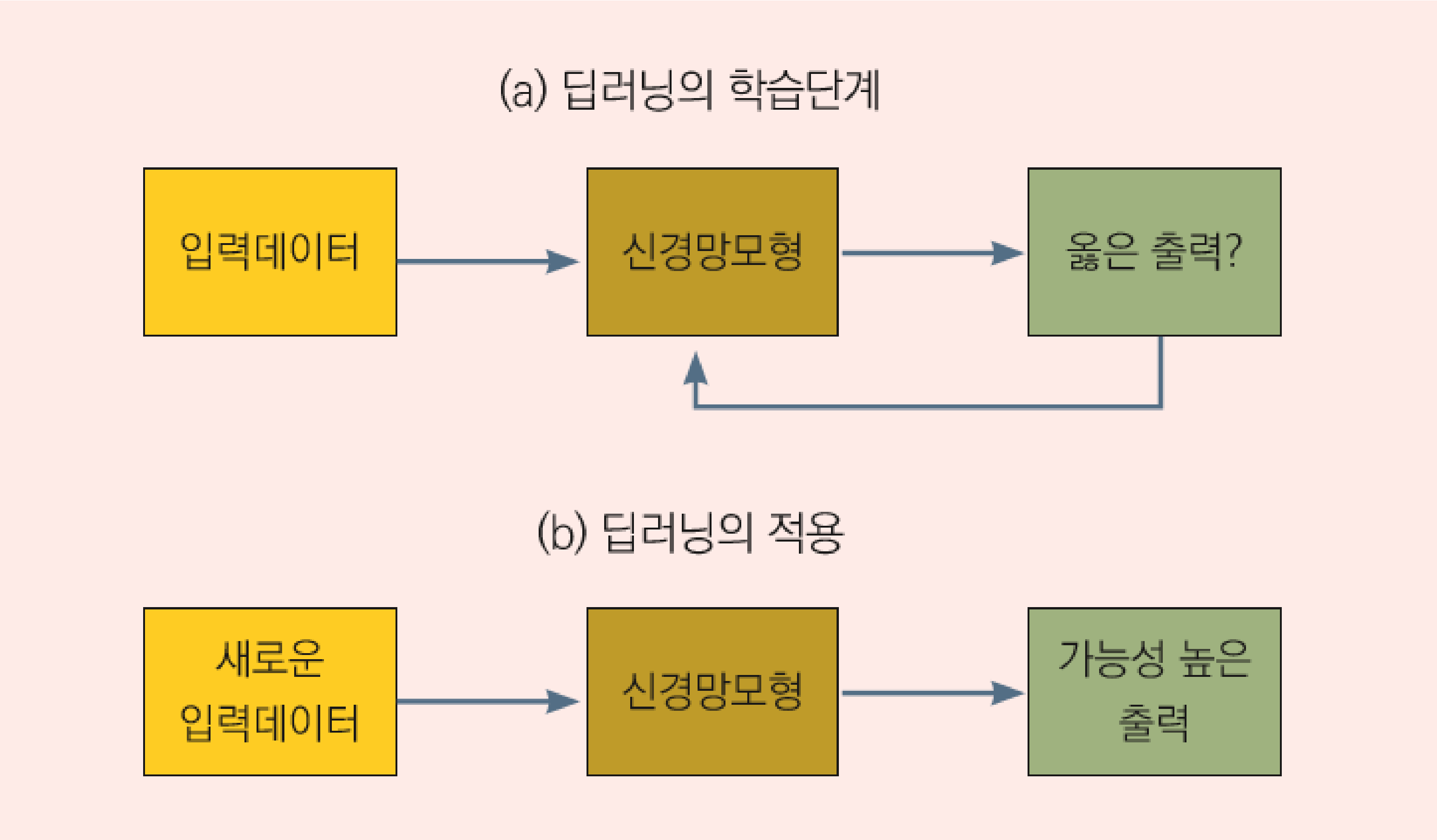
  - 가중치들이 갱신되더라도 손실함수가 더 이상 줄지 않는다면 그 가중치들을 최적값으로 판단

    → 그 가중치들의 신경망을 활용하여 예측

신경망의 학습 데이터

- 전체 데이터를 훈련(training), 검증(validation), 시험(test) 데이터를 나눔
  - 훈련 데이터 : 신경망의 학습
  - 검증 데이터 : 신경망의 선택
  - 시험 데이터 : 신경망의 성과 확인
  - 60%, 20%, 20%로 분할

신경망의 학습과 적용


(a) 딥러닝의 학습단계
입력데이터
신경망모형
옳은 출력?
(b) 딥러닝의 적용
새로운 입력데이터
신경망모형
가능성 높은 출력

손실함수

- 손실함수 $J(w)$: 출력층 값과 실제 값을 비교해서 구함

$$J(w) = \frac{1}{2n}\sum_{i=1}^{n}(y_i - o_i)^2$$

$$J(w) = -\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K} y_{ik}\log(o_{ik})$$

학습(learning)

◆ 손실함수를 최소화 하는 방향으로 경사하강법을 통해 경사에 따라 일정한 학습률로 가중치를 갱신시켜나가는 과정

- 가중치의 초깃값을 정하는 것이 학습속도를 높이는데 중요
  - · 초깃값은 주로 0 근처의 작은 값에서 시작
  - · 초깃값을 너무 큰 값
    → 활성화함수가 제대로 작동하지 않음

학습(learning)

- 1 에포크(epoch) : 하나의 데이터셋 전체를 1회 학습
  - 전체 훈련 데이터를 미니배치로 나누고, 이 조각을 모두 합쳐서 1번 훈련시킨 것도 1 에포크
  - 에포크 수 증가
    → 훈련 데이터의 정확도 증가, 손실함수 감소

학습정리

- ✓ 신경망은 함수가 연속적으로 적용되는 합성함수 또는 행렬로 표현된다.
- ✓ 은닉층이 한 개 이상 있는 신경망은 유계의 연속함수를 근사할 수 있다.
- ✓ 신경망의 학습은 경사하강법으로 손실함수를 작게 만드는 가중치를 구하는 것이다.

다음시간안내

4강. 딥러닝 모형의 구조와 학습(2)