

신경망 (chapter5)

파이널 프로젝트 7주차 스터디

2023.12.22(금) 9:30

이지훈

목차

5.1 뉴런 모델

5.2 퍼셉트론과 다층 네트워크

5.3 활성화 함수

5.4 경사하강법

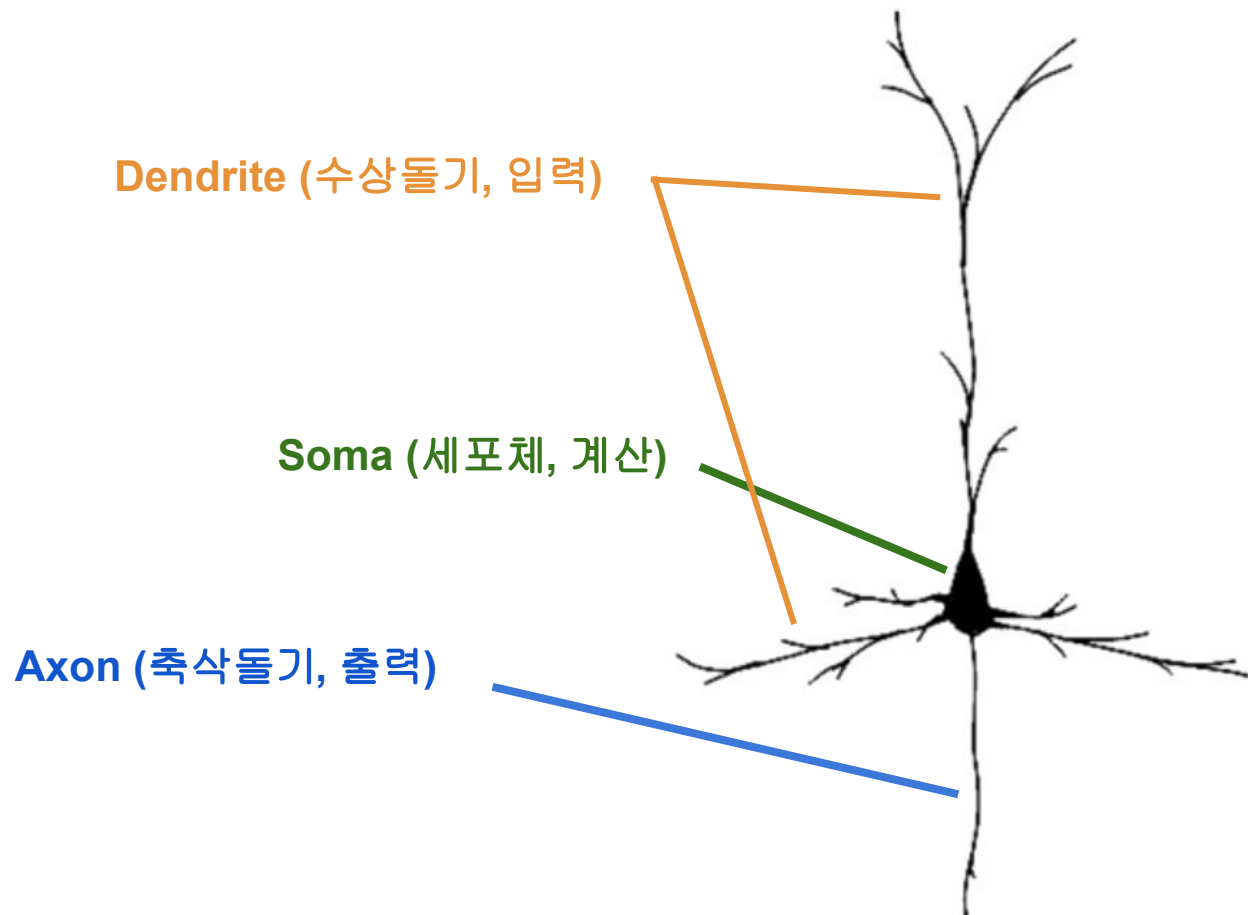
5.5 오차 역전파 알고리즘

5.6 글로벌 미니멈과 로컬 미니멈

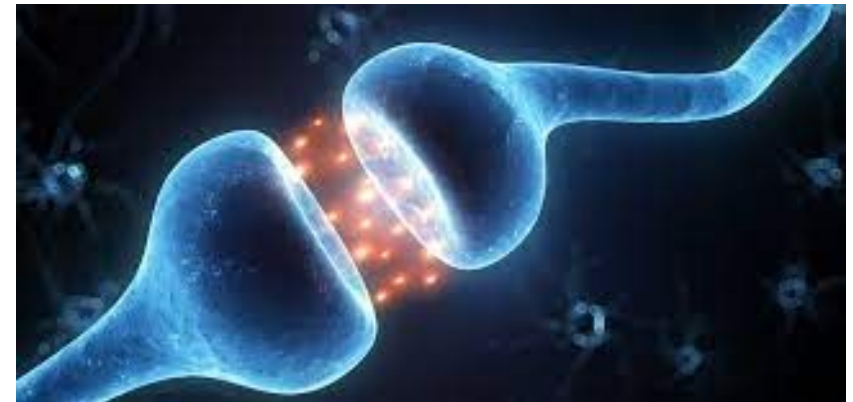
5.7 기타 신경망

5.8 딥러닝

5.1 뉴런 모델

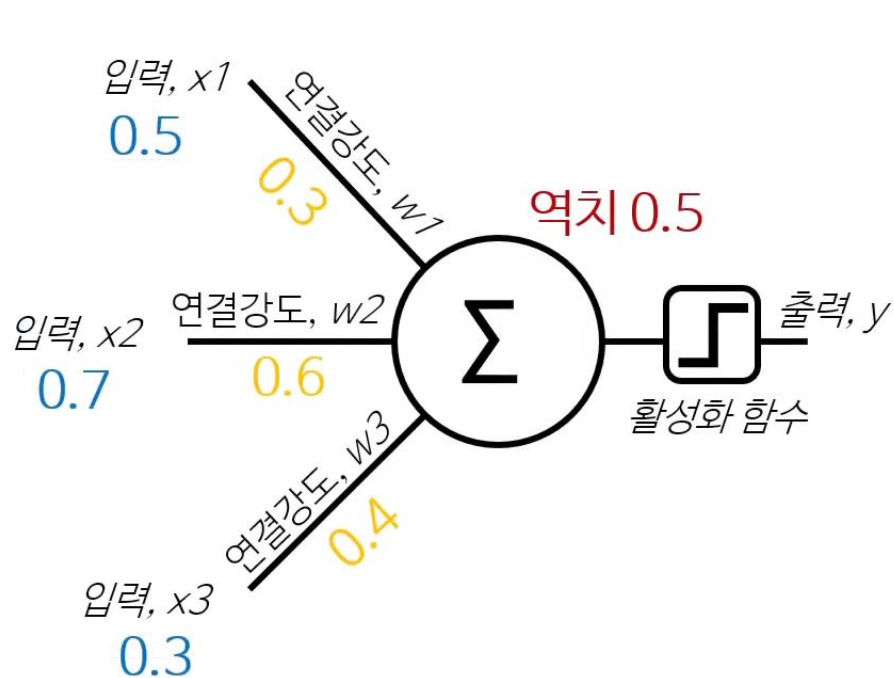


- **시냅스**
이전 뉴런의 **Axon**과 현 뉴런의 **Dendrite** 부분이 만나는 곳에 간극(시냅스) 존재.



5.1 뉴런 모델

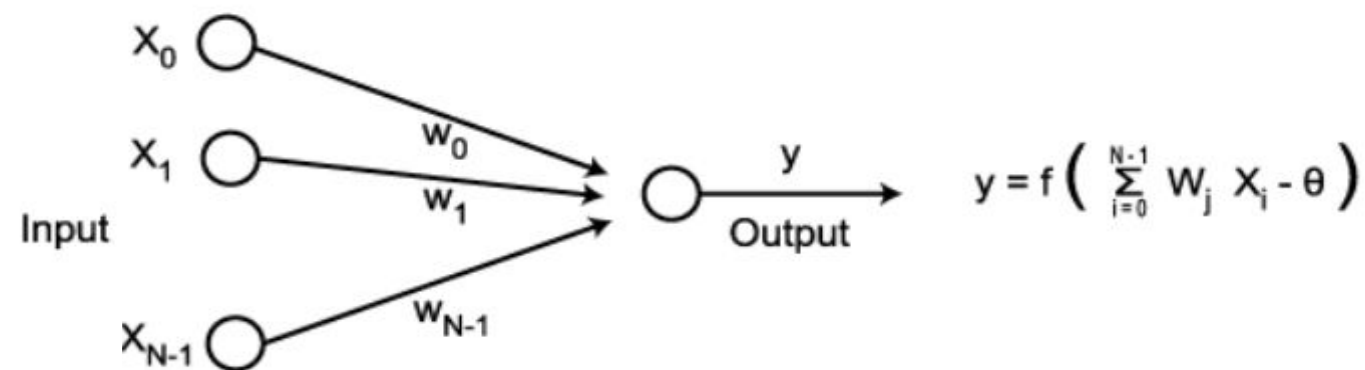
세 개의 뉴런으로 입력받아 역치를 계산



$$\begin{array}{rcl}
 0.5 \times 0.3 & = & 0.15 \\
 0.7 \times 0.6 & = & 0.42 \\
 0.3 \times 0.4 & = & 0.12 \\
 \hline
 \text{역치 } 0.5 & < & 0.69
 \end{array}$$

- 각각의 뉴런으로부터 오는 정보 x_1, x_2, x_3
- 시냅스 역할 연결강도 w_1, w_2, w_3
- 노드에서 세 개의 뉴런값을 입력받아 더함
- 역치 보다 크면 발화

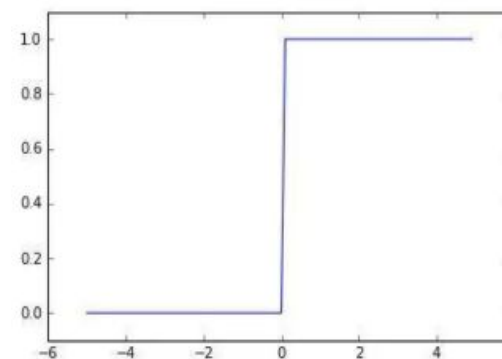
5.1 뉴런 모델



맥컬로치-피츠 뉴런

- 입력 신호의 가중합을 계산하고
이 값이 특정 임계치를 넘으면 출력 신호가 활성화되는 것

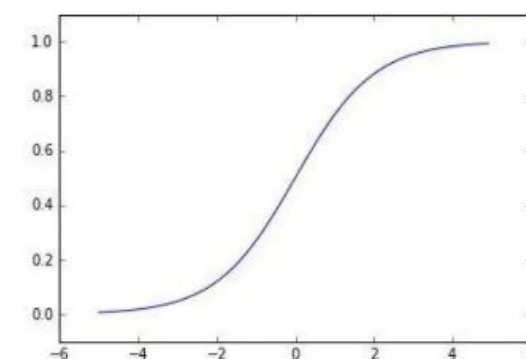
계단 함수, 시그모이드 함수



계단 함수

$$h(x) = \begin{cases} 0 & (x \leq 0) \\ 1 & (x > 0) \end{cases}$$

- 비선형함수
- 역치 이상이면 1, 미만이면 0
- 출력값이 0 또는 1

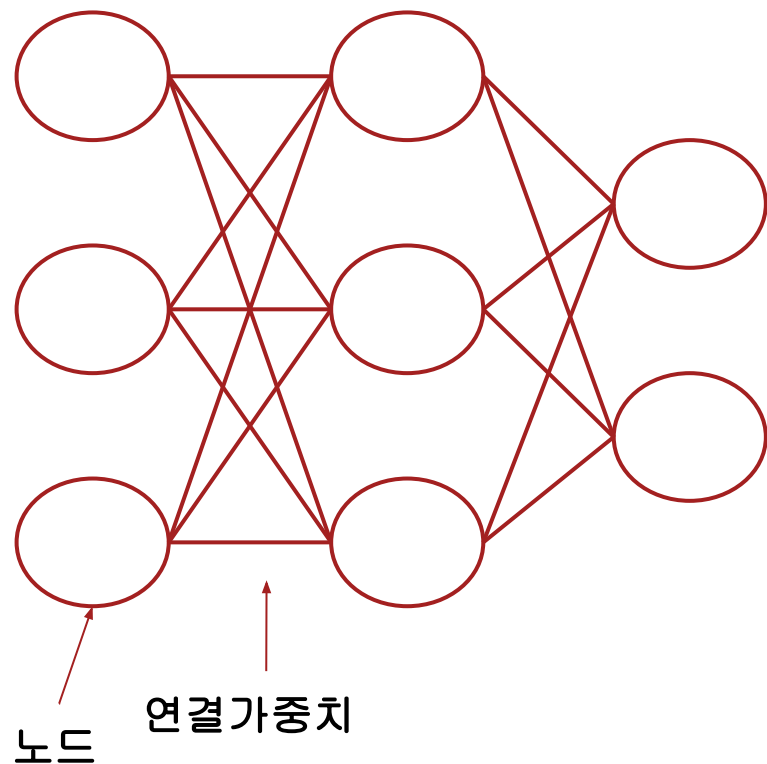


시그모이드 함수

$$h(x) = \frac{1}{1 + e^{-x}}$$

- 미분 가능
- 출력값이 0 과 1 사이

5.2 퍼셉트론과 다층 네트워크



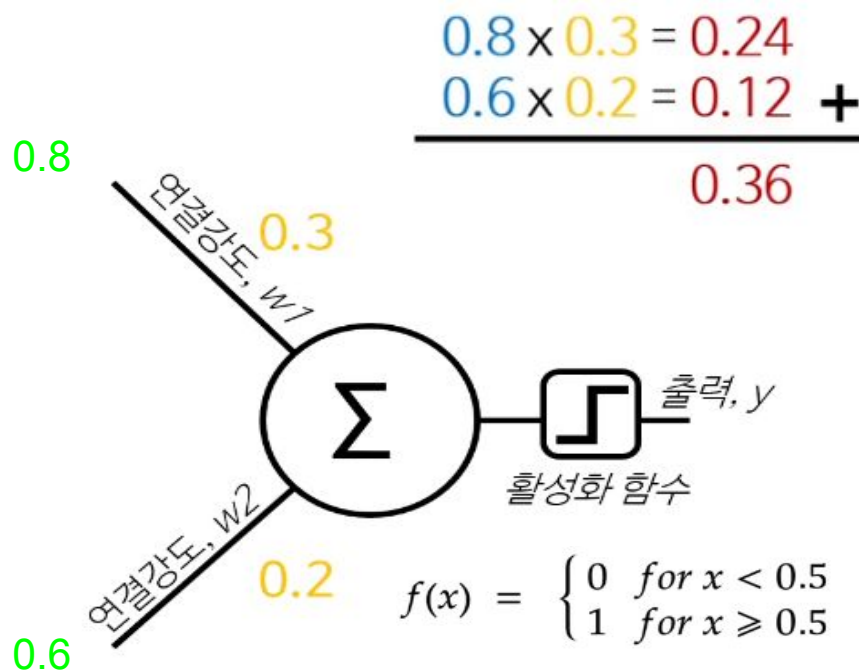
- 원은 노드라고 하고, 선은 연결 가중치를 의미
- 인공신경망을 학습시킨다는 의미는 연결가중치(w)를 조절하여 제대로 된 출력값을 나오게 하는 것
- 모든 신경망 학습은 연결가중치를 변화

5.2 퍼셉트론과 다층 네트워크

단층 퍼셉트론 계산
어떻게 연결 가중치가 변화하는지

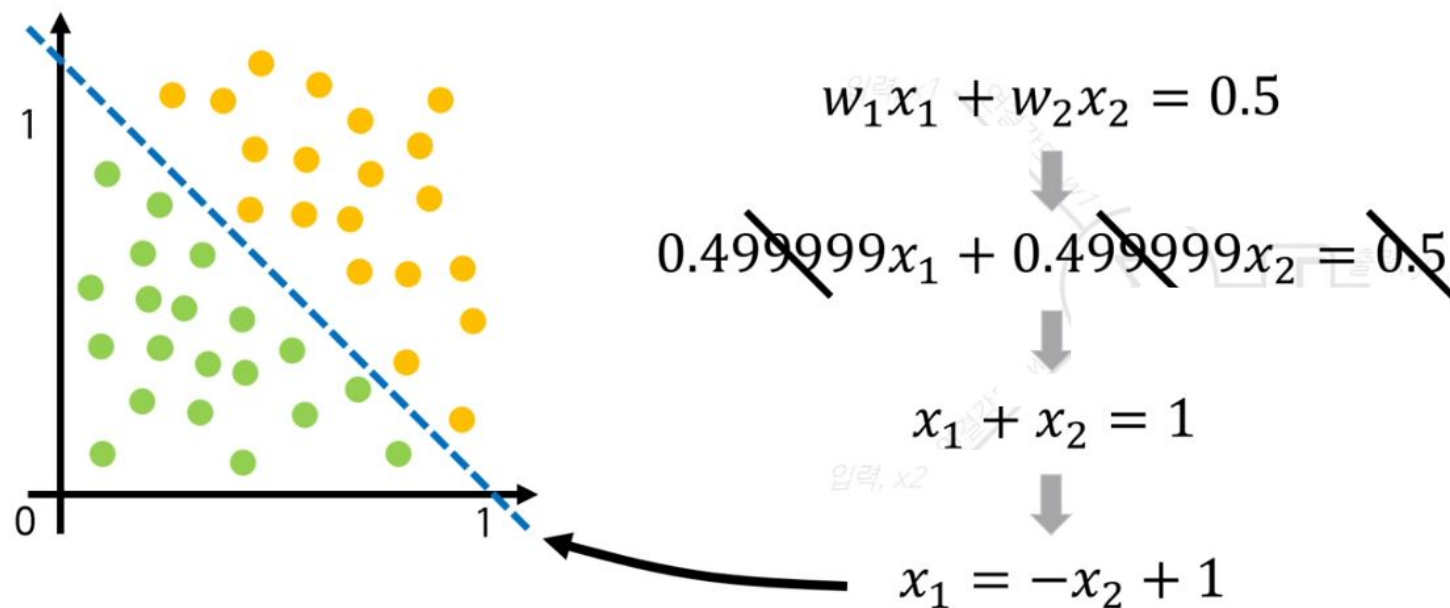
새 연결강도 = 현 연결강도 + 현 입력값 x 오차 x 학습률

x1	x2
0.8	0.6
0.6	0.9
0.1	0.2
0.3	0.1
0.6	0.6
0.4	0.3
0.1	0.2



y	y'
1	0
1	0
0	0
0	0
1	0
0	0
0	0

5.2 퍼셉트론과 다층 네트워크



- 오차 계산하고 연결강도 점진적 변화시키면 w_1 과 w_2 는 0.5에 근사적으로 도달
- 0.5로 다 나뉘어서 일차원 함수가 나옴
- 선형 분리기로서 데이터 분류 가능

5.2 퍼셉트론과 다층 네트워크

x_1	x_2	y
True	True	True
True	False	False
False	True	False
False	False	False

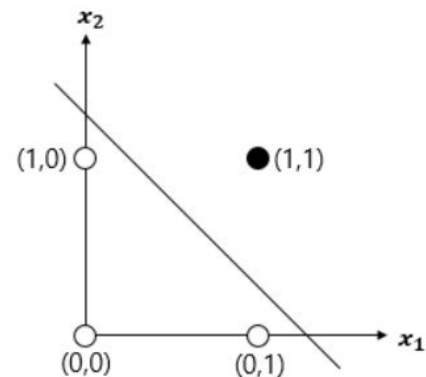
x_1	x_2	y
1	1	1
1	0	0
0	1	0
0	0	0

x_1	x_2	y
True	True	False
True	False	True
False	True	True
False	False	True

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

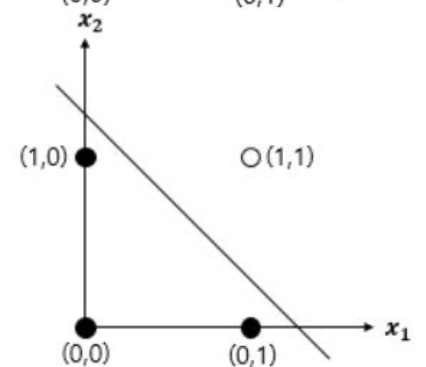
x_1	x_2	y
True	True	True
True	False	True
False	True	True
False	False	False

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



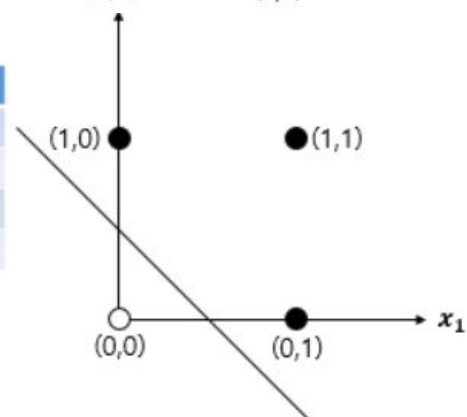
and

T 그리고 T = T
T 그리고 F = F
F 그리고 F = F



nand
(and가 not)

T nand T = F
T nand F = T
F nand F = T



or

T 또는 T = T
T 또는 F = T
F 또는 F = F

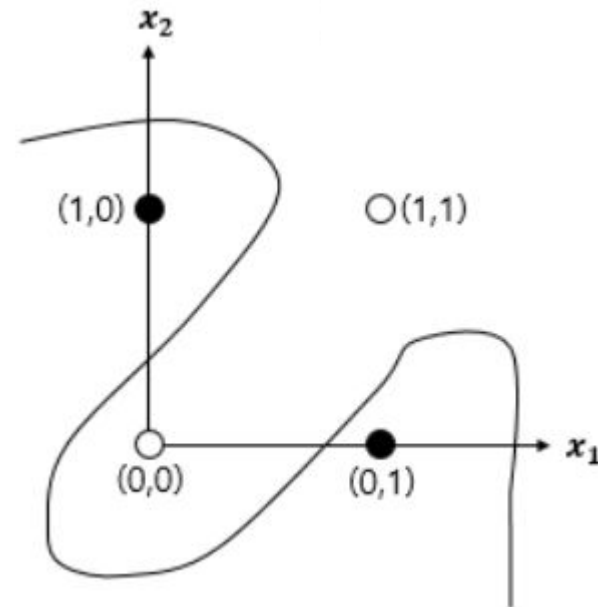
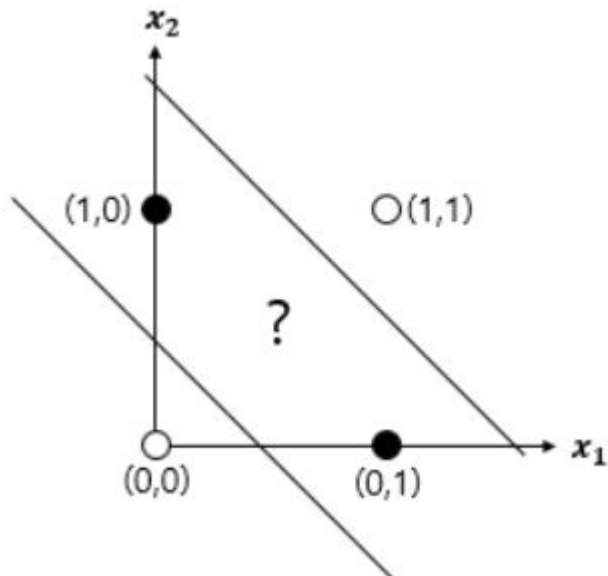
5.2 퍼셉트론과 다층 네트워크

x_1	x_2	y
True	True	False
True	False	True
False	True	True
False	False	False

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

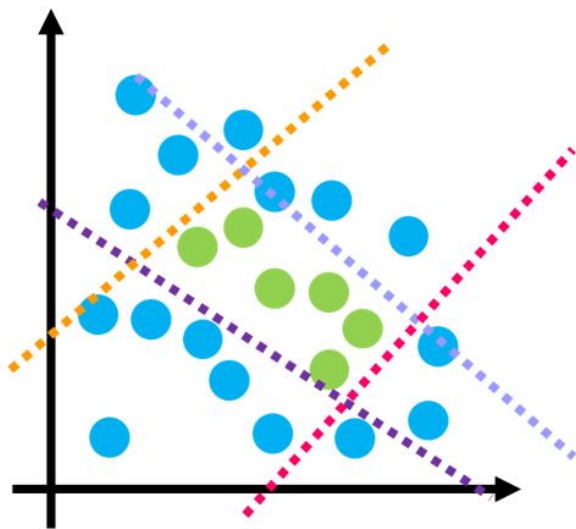
xor

$T \text{ xor } T = F$
 $T \text{ xor } F = T$
 $F \text{ xor } T = T$
 $F \text{ xor } F = F$

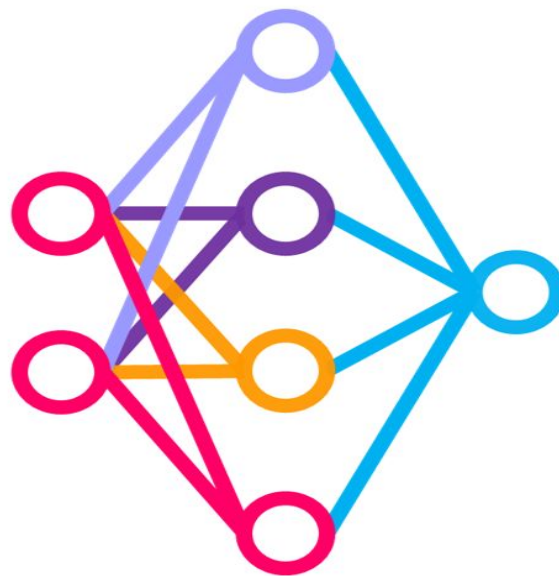


5.2 퍼셉트론과 다층 네트워크

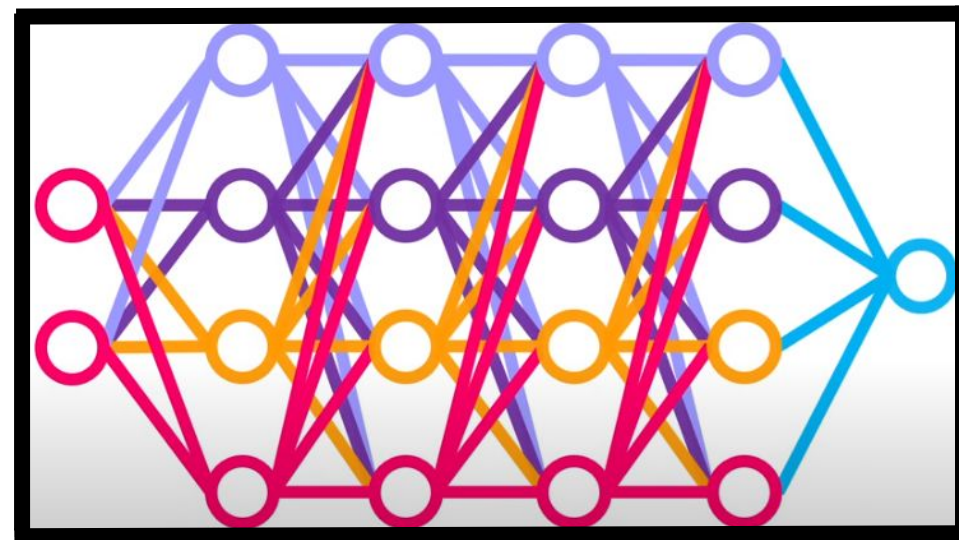
다층 네트워크



- 선을 여러개 더하여 데이터를 분리



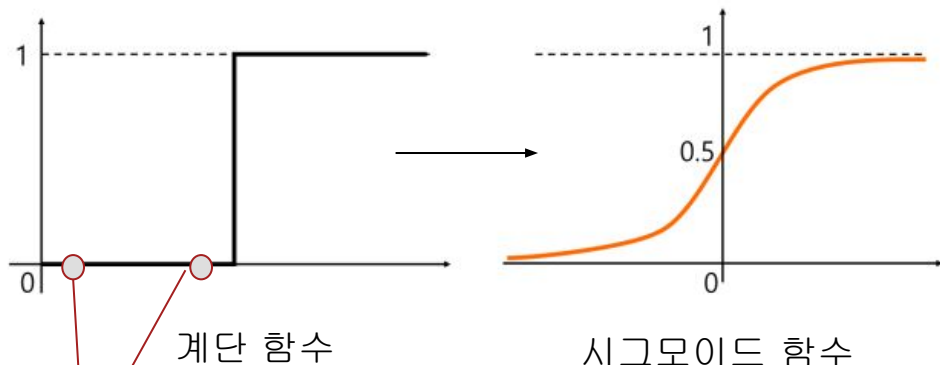
- 다층 신경망은 하나의 입력층, 한 개 이상의 은닉층, 출력층으로 이루어짐.



- 층이 많다는 것이 **deep**하다고 표현

5.3 활성화 함수

계단 함수와 시그모이드 함수



출력값은 0

0.001값과 0.499값 모두
출력값이 0

실제값이 1이라고
가정하면 똑같이 오차가
1이 됨

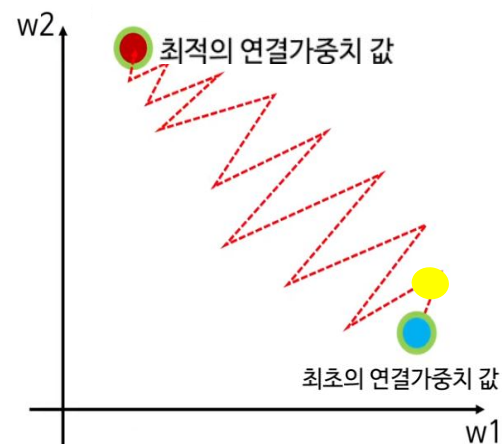
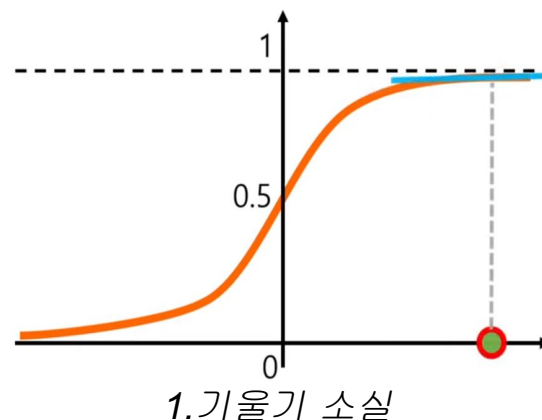
학습의 비효율

미분 가능

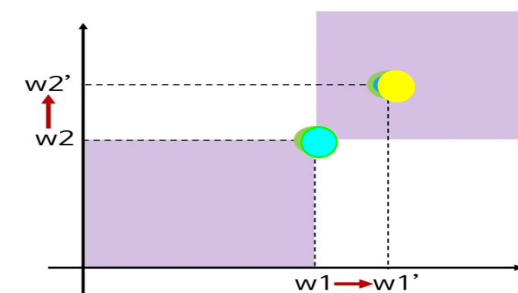
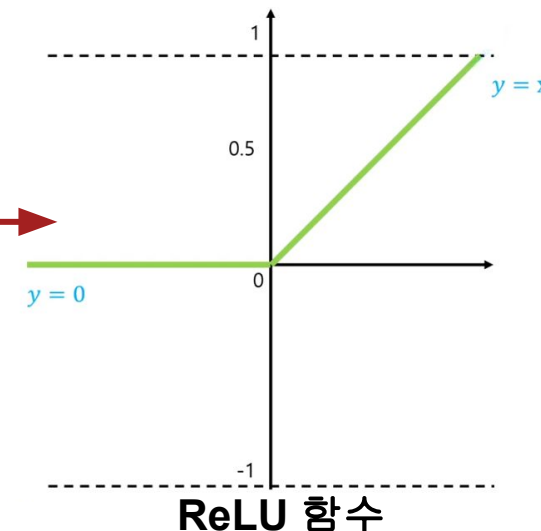
0과 1 사이 값 출력
-> 훨씬 다양한 오차 계산
가능

출력값이 의미 있음

시그모이드의 한계



2. 학습 가중치 변화 중에
나타나는 지그재그 현상



5.4 경사하강법

경사하강법이란?

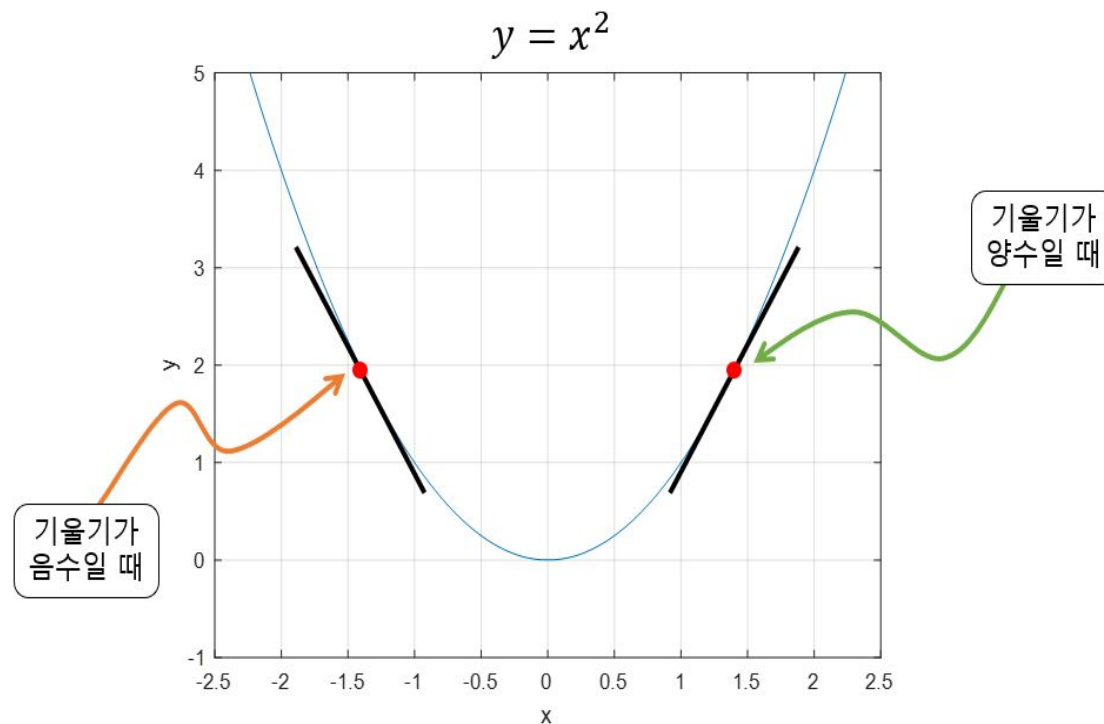
: 주어진 손실함수에서 모델의 파라미터의 최적의 값을 찾는 머신러닝과 딥러닝의 최적화 알고리즘 중 하나

$$MSE = \frac{1}{n} \sum_i^n (y_i - \hat{y}_i)^2$$

MSE : 예측값과 실제값 차이를 측정하는 함수

$$y = \frac{1}{1} \sum_1^1 (x - a)^2$$

n이 1일 때 2차원 함수로 나타나는 것을 볼 수 있음



기울기가 음수면 x 를 증가

기울기가 양수면 x 를 감소

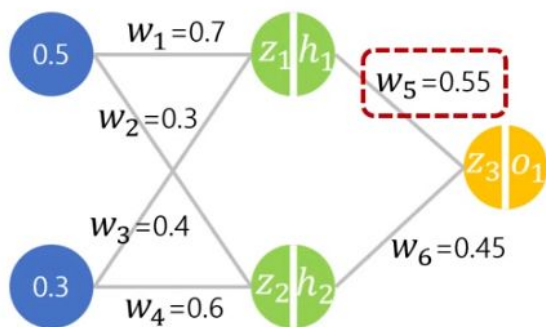
-> 기울기가 0에 수렴 할 때 까지 점진적으로 하강

5.5 오차 역전파 알고리즘

1단계: feedforward 순전파

2단계: 손실계산

3단계: backpropagation 역전파



경사하강법을 이용한 가중치 학습법:

새 연결강도 = 현 연결강도 + $-\frac{\partial C}{\partial w_5}$ x 오차 x 학습률

$$\frac{\partial C}{\partial w_5} = \frac{\partial C}{\partial o_1} \cdot \frac{\partial o_1}{\partial z_3} \cdot \frac{\partial z_3}{\partial w_5}$$

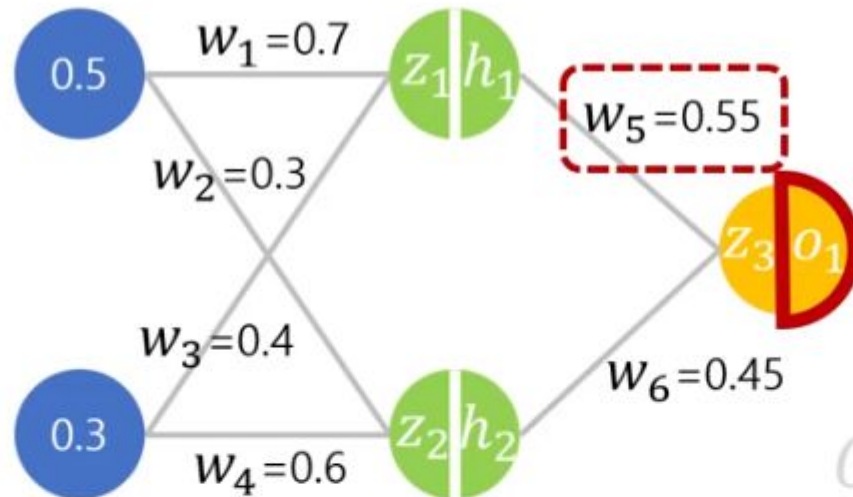
체인룰(연쇄법칙)

어떤 두 변수의 미분 값을 구하려 하는데 관계를 모르면 각각 아는 미분 값으로 연쇄시켜 확장시켜 나가면 부분들을 해결하면서 전체를 해결할 수 있다.

$$\frac{d\text{치타}}{d\text{사람}} = \frac{d\text{치타}}{d\text{사자}} \cdot \frac{d\text{사자}}{d\text{곰}} \cdot \frac{d\text{곰}}{d\text{사람}}$$

$$6 = 2 \times 2 \times 1.5$$

5.5 오차 역전파 알고리즘



$$o_1 = \text{sigmoid}(z_3) = \mathbf{0.645}$$

$$C = \frac{1}{n} \sum_i^n (y_i - \hat{y}_i)^2$$

$$C = (y - o_1)^2$$

$$\frac{\partial C}{\partial w_5} = \frac{\partial C}{\partial o_1} \cdot \frac{\partial o_1}{\partial z_3} \cdot \frac{\partial z_3}{\partial w_5}$$

$$\frac{\partial C}{\partial o_1} = -2(y - o_1)^{2-1}$$

$$\frac{\partial C}{\partial o_1} = -2(1 - 0.645)$$

5.5 오차 역전파 알고리즘

$$S(x) = \frac{1}{1 + e^{-x}}$$

$$O(z) = \frac{1}{1 + e^{-z}}$$

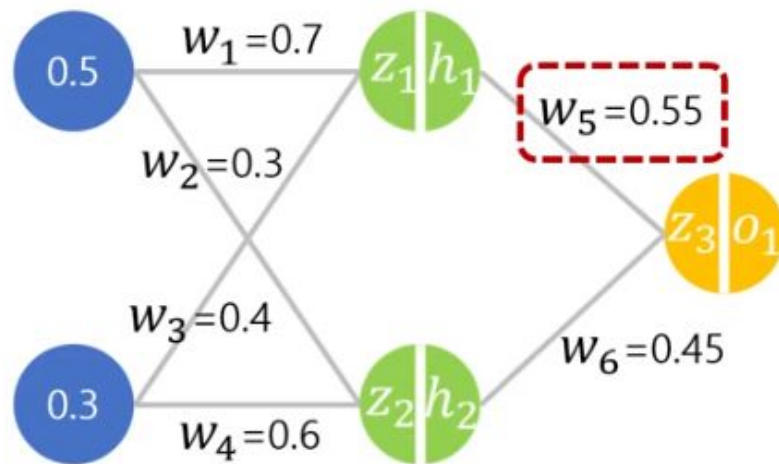
$$\frac{\partial O}{\partial z} = \frac{\partial}{\partial z} \left(\frac{1}{1 + e^{-z}} \right) = O(z)(1 - O(z))$$

$$o_1 = \text{sigmoid}(z_3) = \mathbf{0.645}$$

$$0.645(1 - 0.645) = \mathbf{0.229}$$

$$\frac{\partial C}{\partial w_5} = -0.71 \cdot \mathbf{0.229} \cdot \frac{\partial z_3}{\partial w_5}$$

5.5 오차 역전파 알고리즘



$$z_3 = h_1 w_5 + h_2 w_6$$

$$\frac{\partial z_3}{\partial w_5} = h_1 \cancel{w_5} + \cancel{h_2 w_6}$$

$$= h_1 = 0.615$$

$$\frac{\partial C}{\partial w_5} = -0.71 \cdot 0.2229 \cdot \frac{\partial z_3}{\partial w_5}$$

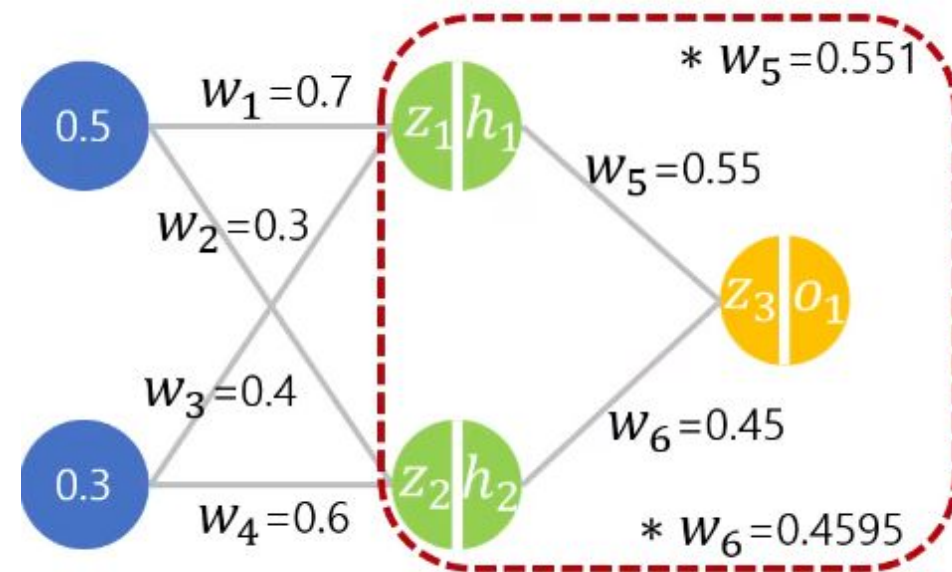
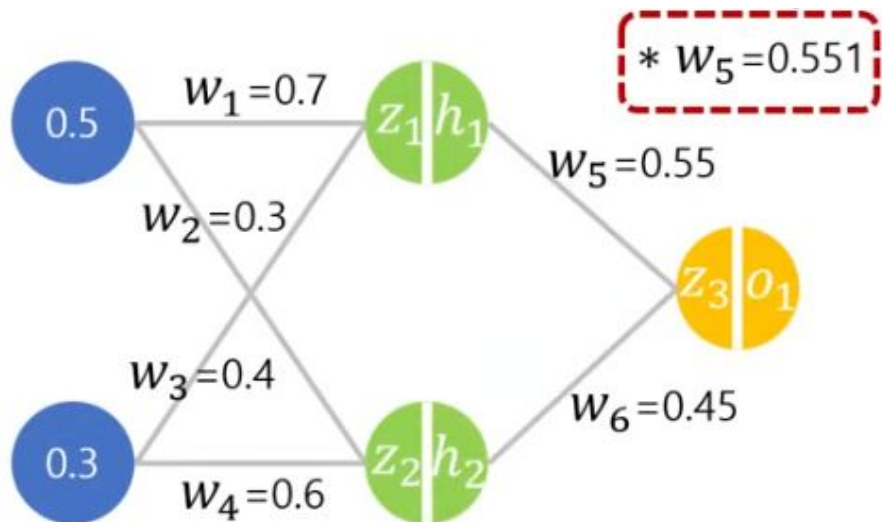
$$\frac{\partial C}{\partial w_5} = -0.71 \cdot 0.2229 \cdot 0.615 = -0.01$$



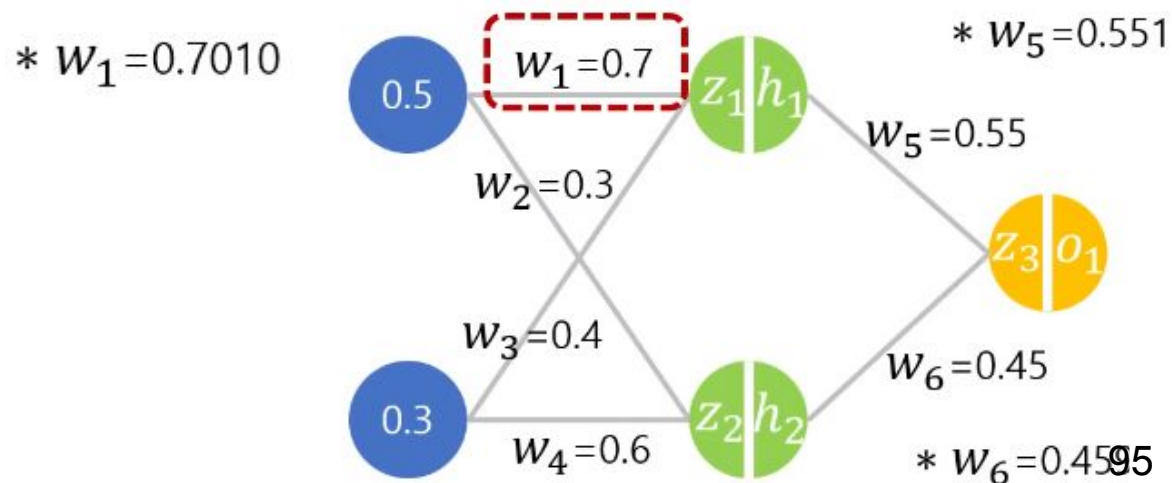
5.5 오차 역전파 알고리즘

새 연결강도 = 현 연결강도 + **0.01** 오차 x 학습률

$$0.551 = 0.55 + \mathbf{0.01} \text{ 오차} \times 0.1$$



5.5 오차 역전파 알고리즘

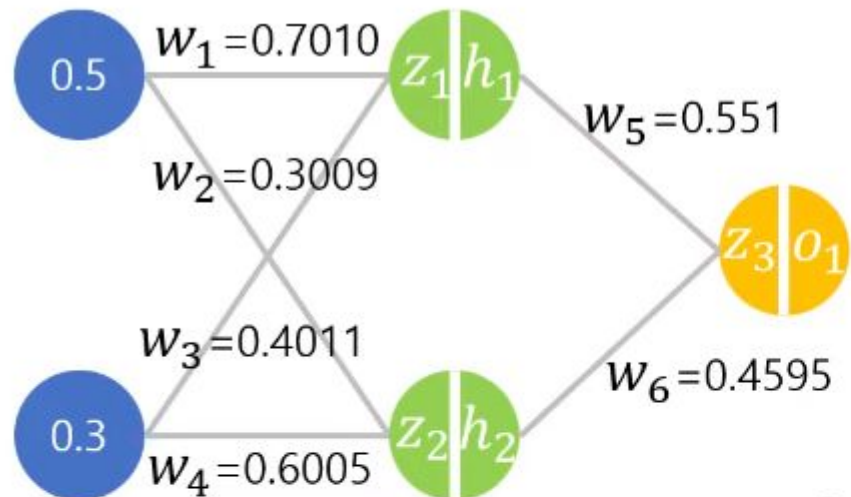


$$\frac{\partial C}{\partial w_1} = \frac{\partial C}{\partial o_1} \cdot \frac{\partial o_1}{\partial z_3} \cdot \frac{\partial z_3}{\partial h_1} \cdot \frac{\partial h_1}{\partial z_1} \cdot \frac{\partial z_1}{\partial w_1}$$

-0.0106 $-0.71 \cdot 0.229$ 0.55 0.237 0.5

$$\begin{aligned} \text{새 연결강도} &= 0.7 + 0.0106 \times 0.1 \\ &= 0.7010 \end{aligned}$$

5.5 오차 역전파 알고리즘



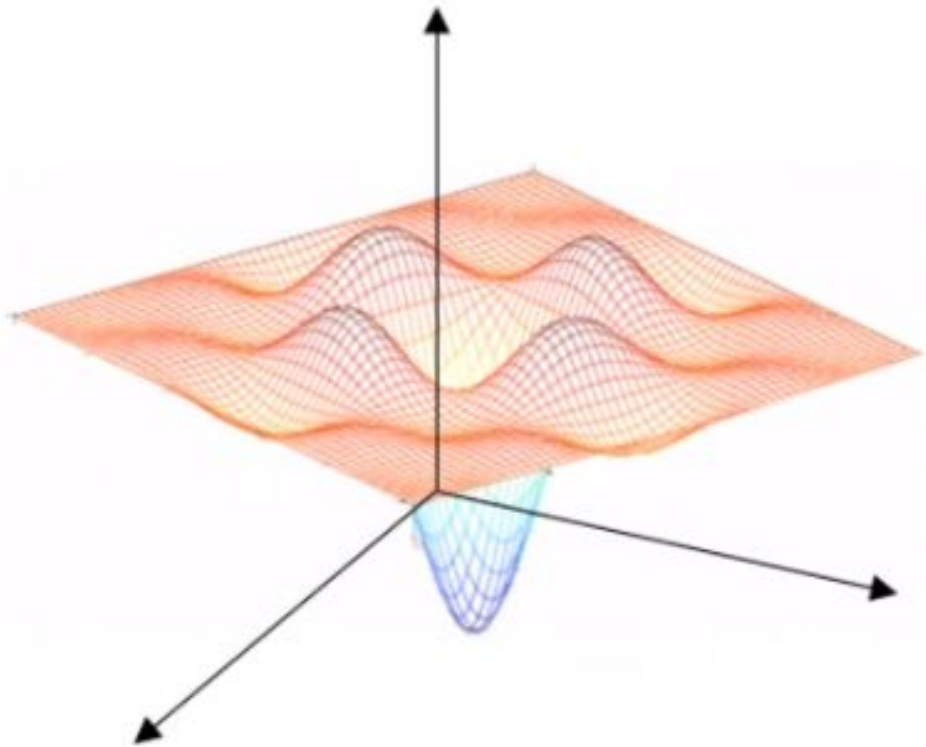
$$o_1 = \text{sigmoid}(z_3) = 0.6443$$

$$C = \frac{1}{1} \sum_i (1 - 0.6472)^2$$

$$C = \mathbf{0.1245}$$

$$C = \mathbf{0.126} \text{ (이전오차)}$$

5.6 글로벌 미니멈과 로컬 미니멈



- 오차가 가장 작은 해를 최종 파라미터로 선정
-> 시작점을 바꾸면서 서로 다른 로컬 미니멈에 빠지며 최대한 글로벌 미니멈에 근접한 결과
- 담금질 기법
- 스토캐스틱 경사하강법

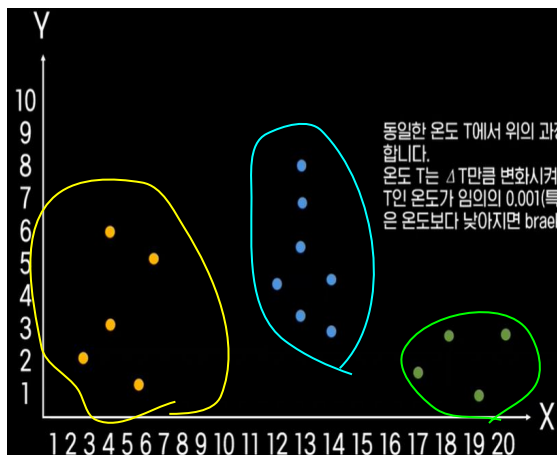
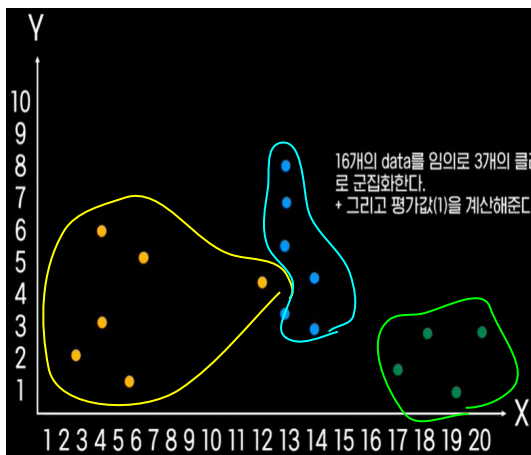
5.6 글로벌 미니멈과 로컬 미니멈

로컬 미니멈의 함정에서 탈출하여 글로벌 미니멈을 찾고자 하는 방법

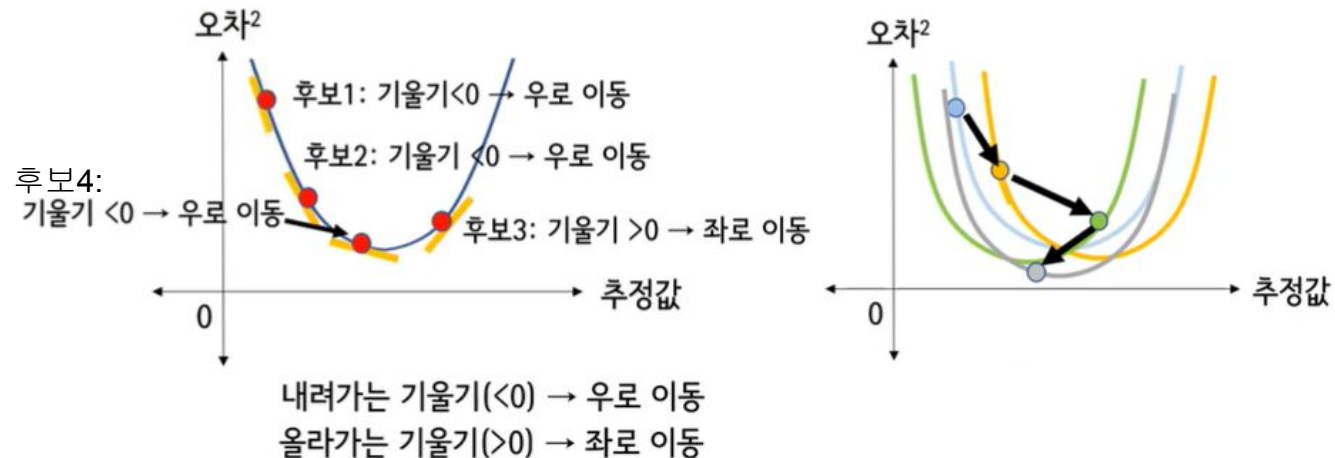
초기값 설정

- 다양한 파라미터 조합 값을 사용해 신경망을 여러 차례 초기화 하고, 일반적 방법으로 훈련한 후에 오차가 가장 작은 해를 최종 파라미터로 선정하는 방법

담금질 기법



스토캐스틱 경사하강법



- 기울기를 계산할 때 랜덤 요소를 추가
기울기가 음수이면 오른쪽으로 이동,
기울기가 양수이면 왼쪽으로 이동.
이 과정을 계속 반복하다 보면 최저점에 도달할 수 있다.

5.7 기타 신경망

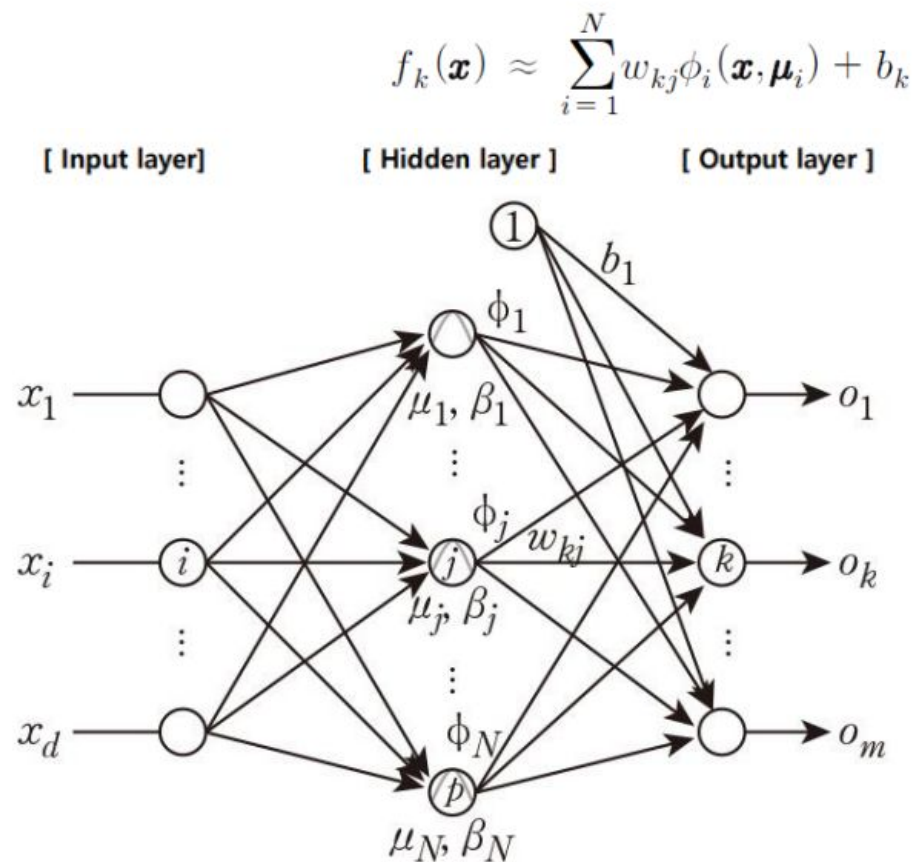
신경망 모델과 알고리즘

1. RBF 신경망
2. ART 신경망
3. SOM 신경망
4. 중첩된 상호연관 신경망
5. 엘만 네트워크
6. 볼츠만 머신

5.7 기타 신경망

1. RBF 신경망

- 어떤 함수 $f_k(\mathbf{x})$ 를 다음과 같이 **RBF 함수들의 선형 결합 형태로** 근사시키는 모델



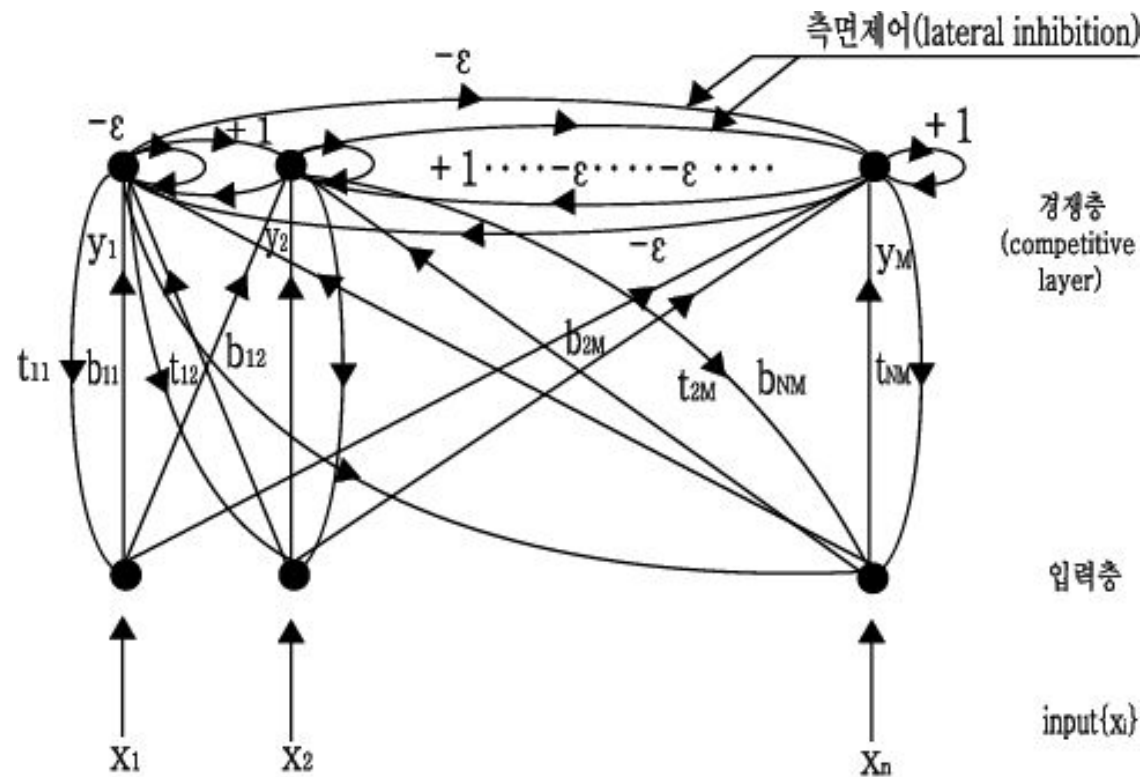
$$\phi_j = \exp(-\beta_j \|\mathbf{x} - \mu_j\|^2)$$

$$o_k = \sum_{j=1}^N w_{kj} \phi_j + b_k$$

- 1개의 입력 (input) 층 , 숨겨진 (hidden) 층, 그리고 1개의 출력(output) 층이다
- 출력층은 선형이다. Linear output layer
- 숨겨진 층과, 입력 층 사이에는 가중치 (weight)가 없다.
- 숨겨진 계층에서는 방사형 기반 함수를 사용한다
- 거리 측정 방식을 통해 연산한다.

5.7 기타 신경망

2. ART 신경망



- 기존에 학습되었던 것이
새로운 학습에 의해 지워지지 않도록 새로운
지식을 자동적으로 전체 지식 베이스에 일관성
있는 방법으로 통합

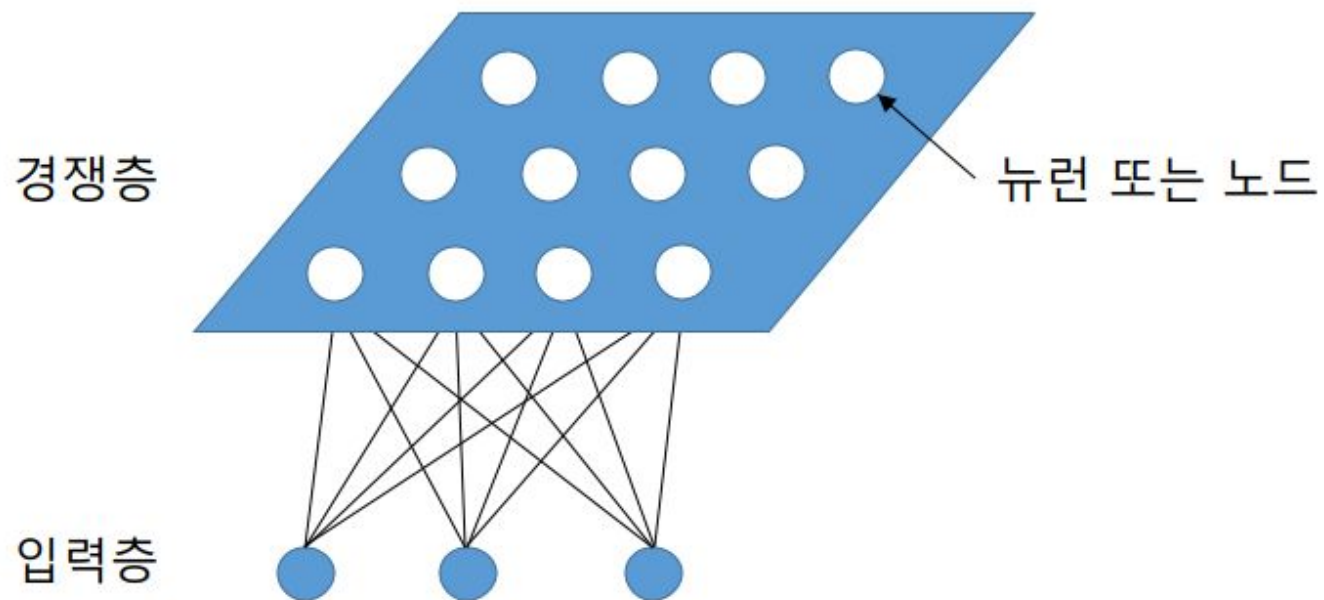
- 첫번째 입력을 첫번째 클러스터의
대표패턴으로 선택

-> 다음 입력값이 들어오면 첫번째 대표
패턴과 비교

-> 거리가 임계값보다 작으면 첫번째
클러스터로 분류, 아니면 새로운 클러스터
생성

5.7 기타 신경망

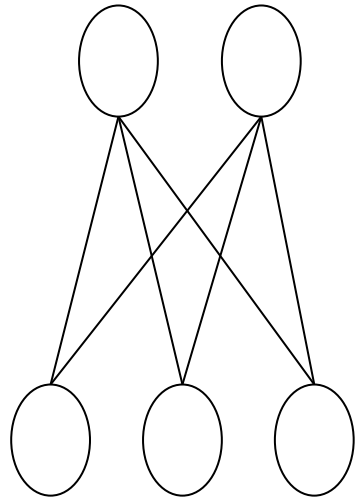
3. SOM 신경망



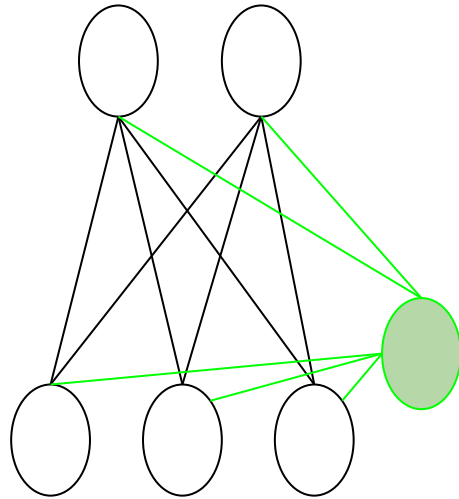
- 비지도 신경망으로 고차원의 데이터를 이해하기 쉬운 저차원의 뉴런으로 정렬하여 지도의 형태로 형상화
- 이러한 형상화는 입력 변수의 위치 관계를 그대로 보존
=> 실제 공간의 입력 변수가 가까이 있으면 지도상에서도 가까운 위치에 있게 된다.

5.7 기타 신경망

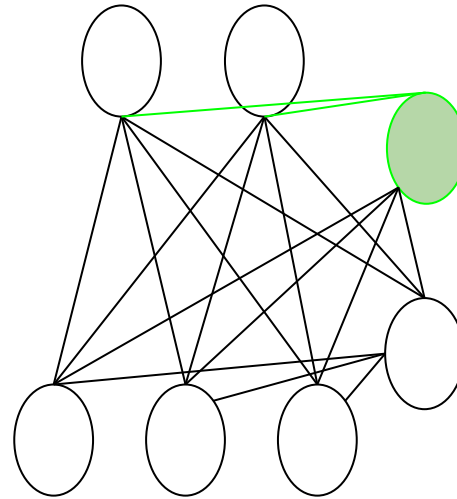
4. 중첩된 상호연관 신경망



초기 상태



하나의 은닉 노드를 추가

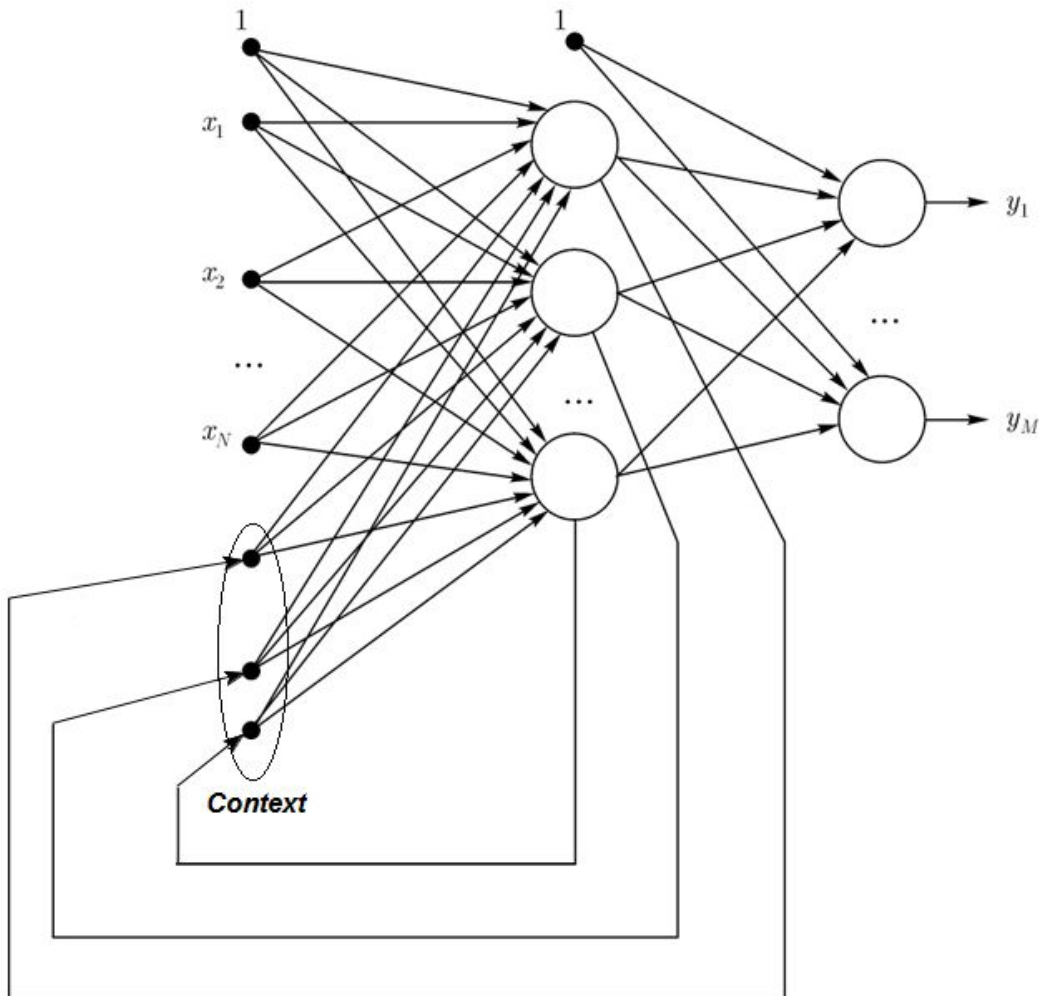


두 번째 은닉 노드 추가

- 구조를 학습 목적으로 선정
- 초기에는 최소 구조인
입력층과 출력층 존재
- **Training** 을 진행하며 **Hidden node** 를
계속 추가하는 형식
- 새로운 **Hidden node** 가 추가되면,
나머지 입력단에 존재하는
연결가중치는 고정됨

5.7 기타 신경망

5. 엘만 네트워크



- RNN의 가장 대표적인 네트워크
- 은닉층 뉴런의 출력이 다시 돌아와 다음 시간대의 입력층 뉴런에 제공되는 신호와 함께 다음 시간대의 입력이 되어 은닉층으로 흘러 들어감
- 활성화 함수는 시그모이드 사용
- context 뉴런의 개수가 은닉뉴런의 개수로 정의되어 신경망이 보다 유연
- 은닉 뉴런 쉽게 추가하거나 삭제 가능

감사합니다.