

신경망

5.1 뉴런 모델

5.2 퍼셉트론과 다층 네트워크

5.3 오차 역전파 알고리즘



통합

Intro

신경망은 지도학습일까요? 비지도학습일까요?

5.0.1. 지도학습 vs 비지도학습

지도학습

- 학습 데이터의 입력과 출력 간의 관계를 학습하여 규칙이나 함수로 표현되는 모델을 찾음.

비지도학습

- 출력 정보가 없는 학습 데이터에 대해서 데이터의 패턴을 발견하는 학습

**입력데이터 = 독립 변수, 출력데이터 = 종속 변수

5.0.1. 신경망은 지도학습

지도학습

- 학습 데이터의 입력과 출력 간의 관계를 학습하여 규칙이나 함수로 표현되는 모델을 찾음.



분류

출력 값이 범주형 데이터인 경우

회귀

출력 값이 연속형 데이터인 경우

* 입력값은 범주형이든 연속형이든 상관 없다.

예외) SOM 신경망은 비지도학습이다. (군집화)

❖ 프로그래밍(예측방법)

input → ? → output

→ 대부분의 학습법은 coding으로 구현되지만, 신경망은 스스로 학습한다.

→ 피드백 과정: 오차 역전파 알고리즘

** 입력데이터 = 독립 변수, 출력데이터 = 종속 변수

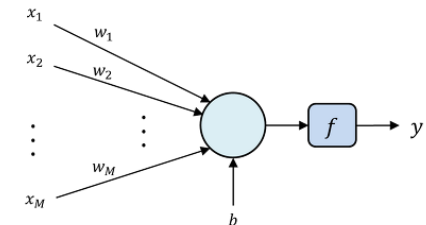
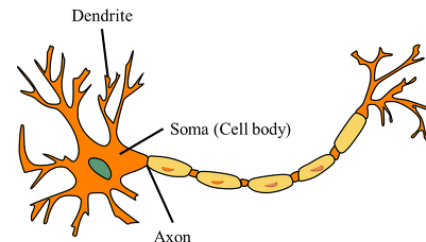
5.1.1. 신경망이란?

- 개념: 뇌신경망의 원리(뉴런을 이용한 생물학적 활동 모델)를 모방한 학습법

신경망이란, 적응성이 있는 단순 단위로 구성된 광범위하고 서로 연결된 네트워크다. 이 조직은 현실 세계 사물에 대한 생물 신경계통의 상호작용을 모방할 수 있다. (p.119, 5 line)

- 신경망의 역사: 컴퓨터의 탄생과 동시에 시작되었지만 **기술의 한계**로 구현하지 못했다. (발전-쇠퇴의 반복)
- 오늘날 신경망은 어떻게 구현할 수 있었을까요?
 1. 하드웨어의 발달 → 빠른 속도의 처리가 가능해짐
 2. 다량의 데이터 축적 및 활용 가능 → 신경망은 대량의 데이터가 있어야 함.
 3. 방법론의 발달 → 규제화기법, 드롭아웃, 최적화 기법 등의 방법론이 발달

- 뇌신경망의 기본 단위: 뉴런(인간의 뇌를 이루는 신경 세포)
- 인공신경망의 기본 단위: **퍼셉트론** perceptron
- 수상돌기: 신호 입력/ 축색돌기: 신호 출력



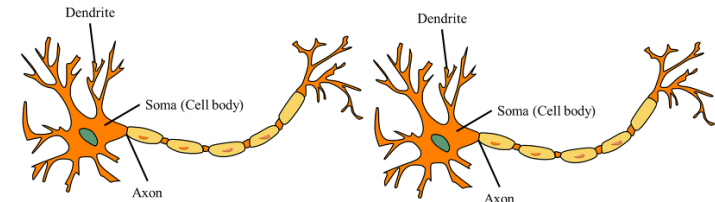
5.1.2. 퍼셉트론

- 인간의 뇌

- 뉴런: 인간의 뇌를 이루는 신경세포. 뇌 신경망의 기본 단위.
- 시냅스: 뉴런과 뉴런 사이의 연결 부위 위치. 전기적 신호 전달함.

* 신경 말단에서 자극을 받으면 시냅스에서 화학 물질이 나와 전위 변화를 일으킴.

* 전위가 임계값을 넘으면 다음 뉴런으로 신호를 전달하고, 임계값에 미치지 못하면 아무 것도 하지 않음.



- 인공신경망

- 퍼셉트론: 신경망을 이루는 가장 중요한 기본 단위.

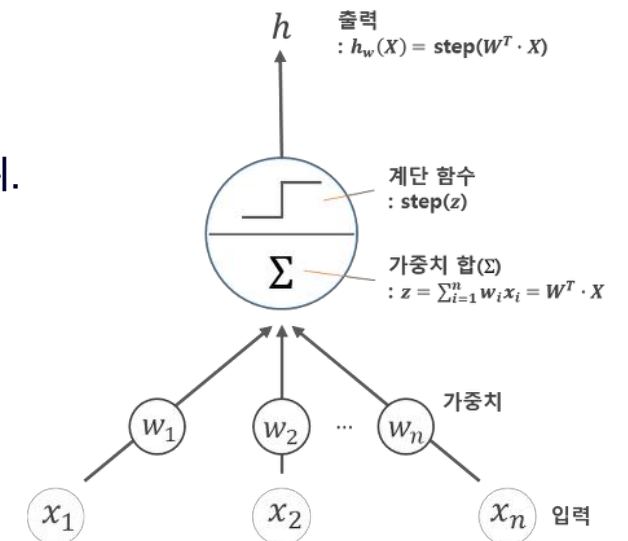
입력 값과 활성화 함수를 사용해서 출력값을 다음 값으로 넘기는 가장 작은 신경망 단위.

- **가중치(연결강도)**: 퍼셉트론과 퍼셉트론 사이를 연결. 입력값을 전달함.

$$z = \sum_{i=1}^n w_i x_i = W^T \cdot X$$

* 입력 값을 합할 때, 가중치 weight를 적용해서 입력 값을 합한다.

* 행렬이 연산이 더 빠르므로 행렬로 사용을 많이 한다.



5.1.2. 퍼셉트론

가중합(weighted sum): 입력 값과 가중치의 곱을 모두 더한 다음에 거기에 바이어스를 더한 값

$$y = \sum_i w_i x_i + b$$

가중합한 값을 어떻게 다른 뉴런에게 전달할까? 활성화 함수

- 인간의 뇌

- 뉴런: 인간의 뇌를 이루는 신경세포
- 시냅스: 뉴런과 뉴런 사이의 연결 부위 - 신호 전달함.

* 신경 말단에서 자극을 받으면 시냅스에서 화학 물질이 나와 전위 변화를 일으킴.

* 전위가 임계값을 넘으면 다음 뉴런으로 신호를 전달하고, 임계값에 미치지 못하면 아무 것도 하지 않음.

- 인공신경망

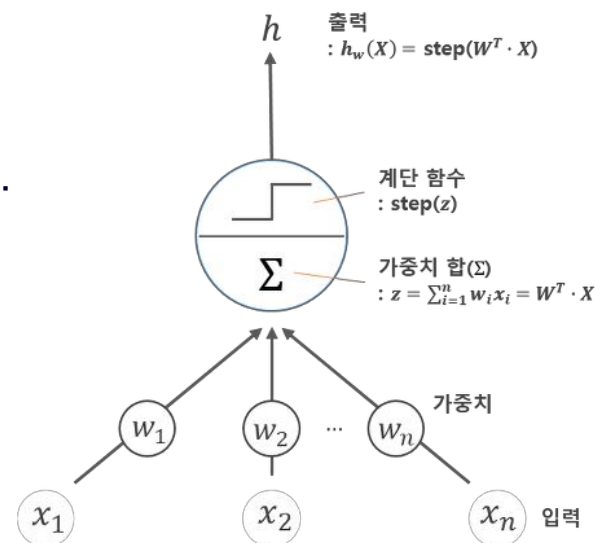
- 퍼셉트론: 신경망을 이루는 가장 중요한 기본 단위.

입력 값과 활성화 함수를 사용해서 출력값을 다음 값으로 넘기는 가장 작은 신경망 단위.

- 가중치(연결강도): 퍼셉트론과 퍼셉트론 사이를 연결.

$$z = \sum_{i=1}^n w_i x_i = W^T \cdot X$$

- 입력 값을 합 할때, 가중치 weight를 적용해서 입력 값을 합한다.
- 행렬이 연산이 더 빠르므로 행렬로 사용을 많이 한다.



5.1.3. 활성화함수

가중합(weighted sum): 입력 값과 가중치의 곱을 모두 더한 다음에 거기에 바이어스를 더한 값

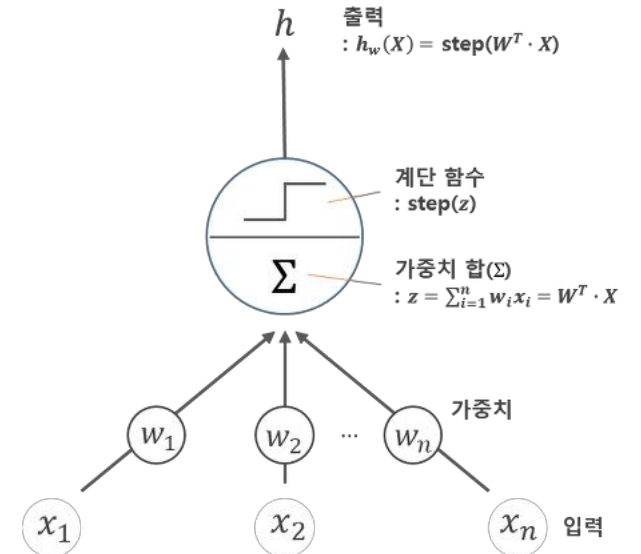
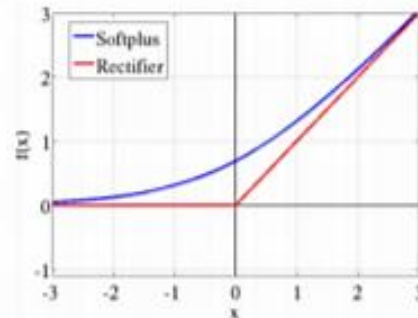
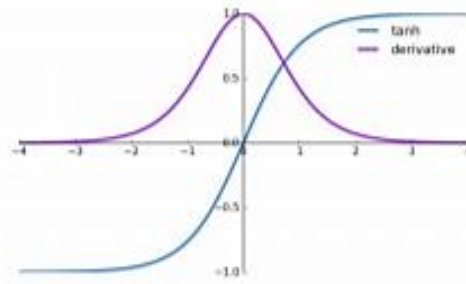
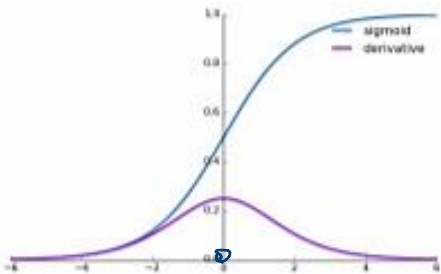
$$y = \sum_i w_i x_i + b$$

- 활성화 함수: 가중합의 결과를 놓고 0 또는 1을 출력하도록 만드는 함수

* 로지스틱 회귀 분석 시에 배웠던 시그모이드 함수가 대표적인 활성화 함수 중 하나임.

- 활성화 함수 종류

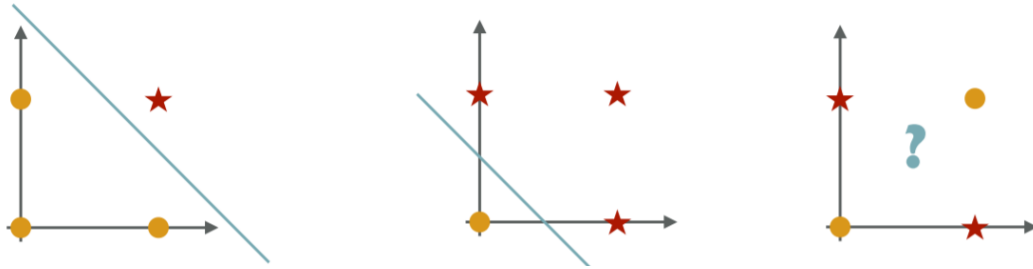
- 시그모이드 함수: $f(x) = \frac{1}{1 + e^{-x}}$
- 하이퍼볼릭 탄젠트 (hyperbolic tangent): $f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
- 렐루(ReLU): $f(x) = \max\{0, x\}$
- 소프트플러스(softplus): $f(x) = \log(1 + e^x)$
- 여러 은닉층을 거치고 곱해지더라도 맨 처음 층까지 사라지지 않고 남아있도록 할 수 있음



5.1.4. 퍼셉트론 모형의 한계

- XOR 문제

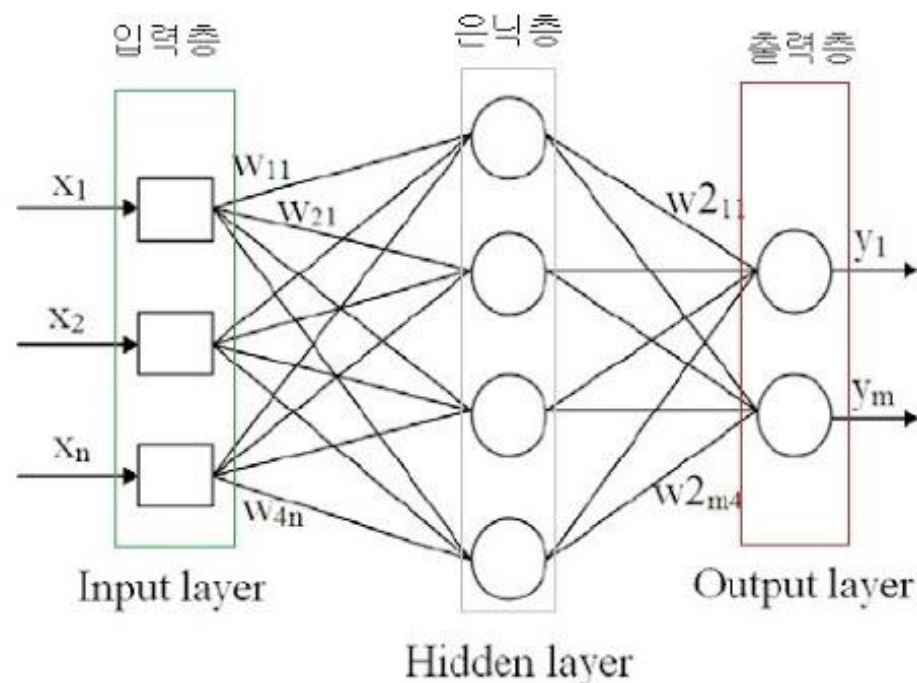
- XOR 게이트의 경우에는 여러 개의 선을 아무리 그려 보아도 하나의 직선으로는 두 종류의 점을 완벽하게 구분할 수 없음
- 이후 인공지능 연구가 한동안 침체기를 겪음
- 약 10년 후, 다층 퍼셉트론의 개념을 통해서 이 문제가 해결됨.



5.1.5. 다층퍼셉트론

- 은닉층 (hidden layer)를 추가해서 은닉층에 존재하는 노드들의 값들을 계산한 후에 이를 다음 층의 입력으로 사용함
- 간단한 다층 퍼셉트론의 예 (입력 변수: 2개 x_1, x_2)
 - ✓ 한 개의 은닉층을 가지고 있고, 해당 은닉층이 2개의 노드를 가지고 있다고 가정하면 은닉층의 노드 n_1, n_2 는 다음과 같이 계산할 수 있음
 - ✓ $n_1 = \sigma(x_1 w_{11}^{(1)} + x_2 w_{21}^{(1)} + b_1^{(1)})$, $n_2 = \sigma(x_1 w_{12}^{(1)} + x_2 w_{22}^{(1)} + b_2^{(1)})$
 - ✓ 최종 출력 값은 은닉층의 노드들로부터 다음과 같이 계산됨
 - ✓ $y_{out} = \sigma(n_1 w_1^{(2)} + n_2 w_2^{(2)} + b^{(2)})$
- XOR 문제의 해결
 - ✓ $w^{(1)} = [-2, 2; -2, 2]$, $b^{(1)} = [3, -1]$, $w^{(2)} = [1, 1]$, $b^{(2)} = -1.3$

x1	x2	n1	n2	y(out)	예측 값
0	0	$s(3)=0.95$	$s(-1)=0.27$	$s(-0.01)=0.48 < 0.5$	0
0	1	$s(1)=0.73$	$s(1)=0.73$	$s(0.16) = 0.54 > 0.5$	1
1	0	$s(1)=0.73$	$s(1)=0.73$	$s(0.16) = 0.54 > 0.5$	1
1	1	$s(-1)=0.27$	$s(3)=0.95$	$s(-0.01) = 0.48 < 0.5$	0



5.2.1. 오차 역전파 알고리즘

- 신경망 목적: 가중치, 임계값 구하기
- 에포크 epoch: 모든 데이터를 한 번 학습하는 것
 - * 에포크 횟수가 많다 = 학습량이 많다.

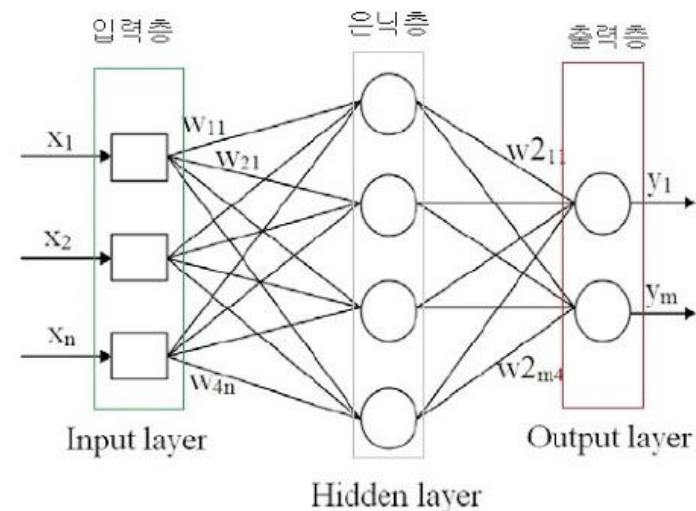
- 오차 역전파 알고리즘

: 출력층의 오차를 가지고 앞 층의 오차를 역추적하는 과정

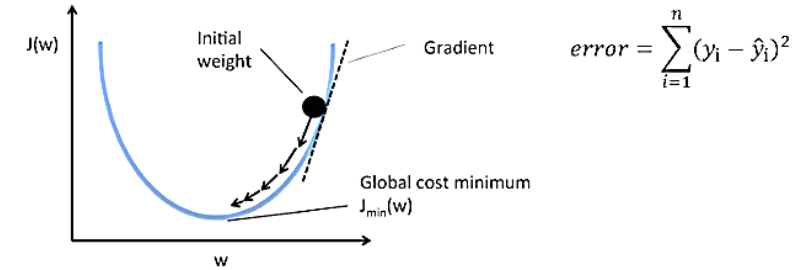
은닉층의 가중치에 영향이 끼친 후 전파되어 입력층의 가중치에 영향을 끼치게 됨.

- 오차 역전파 알고리즘 구동 방식

1. 임의의 초기 가중치와 임계값을 준 뒤에 결과 값을 계산
2. 계산 결과과 우리가 원하는 값 사이의 오차를 계산
3. 경사 하강법을 이용해 바로 앞 가중치를 와가 작아지는 방향으로 업데이트함
4. 위의 과정을 오차가 더 이상 줄어들지 않을 때까지 반복



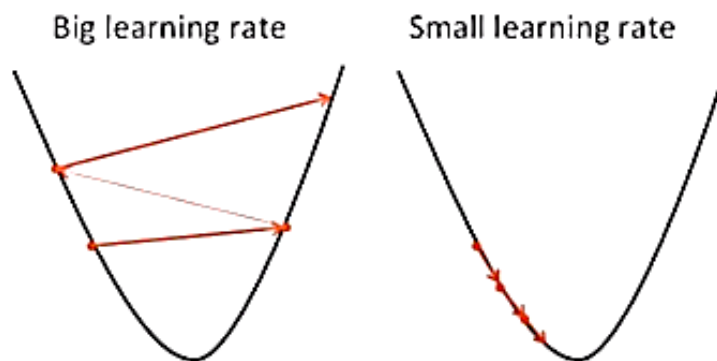
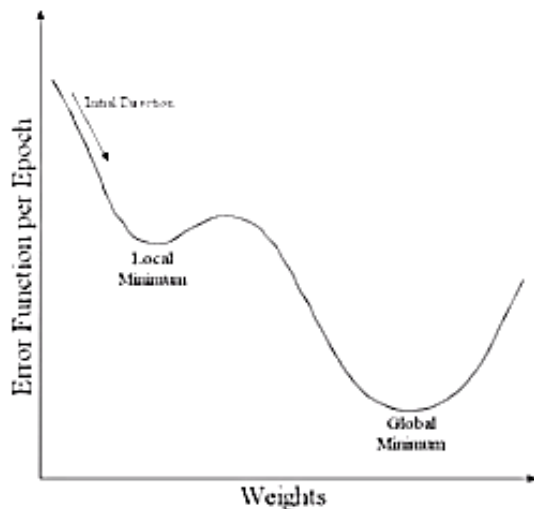
5.2.2. 경사하강법



- 방법: 손실함수를 이용하여 에러를 최소화 하도록 가중치를 조정
- 경사하강법: 미분을 이용하여 기울기가 최소인 지점 찾기
- 모멘텀 momentum: 오차를 수정하기 전 바로 앞 수정 값과 방향을 참고하여 같은 방향으로 비율만 수정되게 하는 방법
 - * 이전 값을 고려하므로 수정 방향이 급격하게 바뀌는 것을 줄일 수 있음.
- 학습률 learning rate: 새로운 정보를 얼마나 반영할지를 조절. 변화량 조절.
 - 다층 퍼셉트론으로 해결하고자 하는 문제의 종류에 따라서 최소화 하게 되는 손실 함수의 종류가 달라짐
 - 분류 문제
 - ✓ 이진 분류 문제: 로지스틱 손실 함수 사용 (binary crossentropy) $\rightarrow y_{true} \log \sigma(\hat{y}) + (1 - y_{true}) \log(1 - \sigma(\hat{y}))$ 값들의 평균 값을 오차 함수로 함
 - ✓ 다중 분류 문제: 범주형 교차 엔트로피 사용 (categorical crossentropy) $\rightarrow - \sum_{c=1}^C y_c \log \hat{a}_c$ 값들의 평균 값을 오차 함수로 함
 - 회귀 문제
 - ✓ 제곱 오차 함수: $(y_{true} - \hat{y})^2$ 값들의 평균 값을 오차 함수로 함
 - ✓ 절대 오차 함수: $|y_{true} - \hat{y}|$ 값들의 평균 값을 오차 함수로 함

4.2.2. 경사하강법

- 모멘텀 momentum: 오차를 수정하기 전 바로 앞 수정 값과 방향을 참고하여 같은 방향으로 비율만 수정되게 하는 방법
 - * 이전 값을 고려하므로 수정 방향이 급격하게 바뀌는 것을 줄일 수 있음.
- 학습률 learning rate: 새로운 정보를 얼마나 반영할지를 조절. 변화량 조절.



연습문제

5.1 선형 함수 $f(x) = w^T x$ 를 신경망 활성화 함수로 사용했을 때의 결함에 대해 설명하라.

⇒ 선형분리 불가능한 문제들 (ex) XOR 문제)를 해결할 수 없다.

은닉층이 존재해도 해결불가능

$h(x) = ax \rightarrow$ 은닉층 추가 $h(h(x)) = ax^2$: 똑같이 선형함수