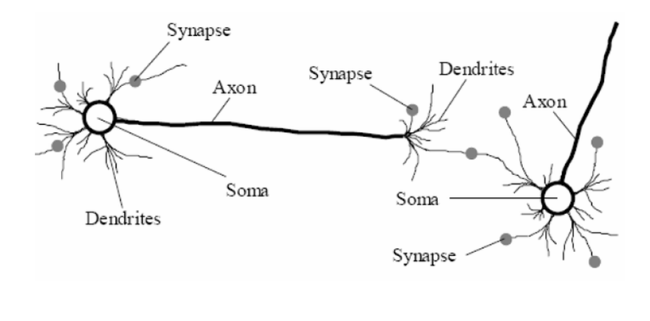
**신경망(Neural Network)**

정확히는 인공신경망ANN(artificial neural network)

줄여서 신경망 NN이라고 많이 부른다.

<**생물학적 신경망**의 구조>

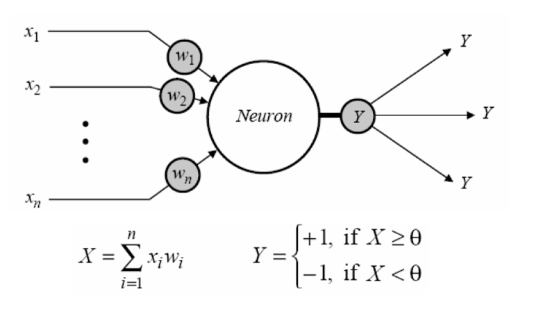
수상돌기(Dendrites) : 외부 신경자극 받아들이는 역할

축삭돌기(Axon) : 다른 뉴런의 신호를 전달하는 기능

신경세포체(Soma) : 외부자극에 대한 판정을 하여 다른 뉴런으로 신호를 전달할 것인지를 최종 결정을 한다.

시냅스(Synapse) : 다른 뉴런의 축삭돌기 말단과 수상돌기의 연결 부위

<**인공 신경망**의 구조>



생물학적 뉴런의 구조를 모방하여 만들어졌다.

수상 돌기에 해당되는 입력(x1,x2,...., xn)에 시냅스에 해당하는 입력의 가중치(w1,w2,...,wn)를 곱하고 이것들의 총합이 신경세포체(Soma) 부분으로 전달이되면

신경세포체에서는 활성함수(activation function)에 따라 최종 출력 Y가 결정이 된다.

활성 함수가 특정 경계값과 비교를 하여 같거나 크면 +1을 출력하고 작으면 –1을 출력한다.

인공신경망은 보통 이런 뉴런들을 multi-layer 로 구성을 하며 일종의 훈련(training) 데이터를 통한 반복 훈련을 통해 가중치(w1,w2,...,wn)의 최적값이 정해지게 된다.

<Backpropagation (역전파) >

Backpropagation의 알고리즘의 기원

신경망에서 가장 중요한 개념 중 하나가 역전파(backpropagation)이다.

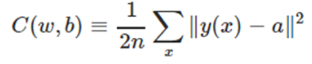
이 역전파를 통해 “역방향으로 에러를 전파(backward propagation of error)” 시키면서 최적의 학습 결과를 찾아가는 것이 가능해졌다.

<Backpropagtion 이해를 위한 사전 지식>

1. Cost Function (Loss Function)

역전파에 대한 본격적인 설명에 앞서,

cost function 혹은 loss function 의 개념을 먼저 파악해야 할 필요가 있다.



여기에 n은 훈련에 사용하는 입력의 수를 나타내고,

y(x)는 입력 x를 가했을 때의 기대출력을 나타내며,

a는 입력 x를 신경망에 넣었을 때의 실제출력이다.

cost function이란 신경망에 훈련 데이터 x를 가하고

실제출력과 기대출력간의 차에 대한 MSE(Mean Square Error)를 구하는 것임을 알 수가 있다.

y(x)와 a의 차가 작아질수록 신경망이 학습이 잘 된다고 볼 수 있으며, 훈련 데이터를 이용해 가중치(w)와 바이어스(b)를 변화시키는 과정을 반복적으로 수행하여 cost function이 최소값이 되도록 하는 것이 신경망 학습의 목표이다.

2. Gradient-descent 방법에 기반한 학습

신경망의 뉴런의 개수가 많아지고, 입력의 수가 많아질수록 더욱 더 해를 구하기가 어려워진다.

이때 필요한 개념이 gradient-descent 방법이다.

훈련 데이터를 입력으로 가하고, cost function이 최소가 되도록 w와 b값들을 반복적으로 변화시켜 가다보면 결국 최소 값에 이를 수 있게 된다.

위 그림은 gradient-descent를 설명할 때 흔히 사용되는 그림으로, 어느 위치에서 시작을 할지라도 gradient가 음의 최대값 쪽으로 움직이게 되면, 결국 최소값에 도달할 수 있다.

단, 단점은 경사가 큰 경우는 빠른 속도로 수렴을 하지만,

거의 바닥에 오게 되면, 기울기가 작아지기 때문에 수렴 속도가 현저히 느려진다는 점이다.

**<역전파(backpropagation) 기본 개념>**

신경망의 크기가 커지고, 입력이나 출력의 개수가 많아지면,

변수들이 너무 많기 때문에 이 작업은 매우 곤혹스러운 작업이 된다.

“가중치나 바이어스 값을 아주 작게 변화를 시키면, 즉 편미분을 시키면, 출력 쪽에서 생기는 변화 역시 매우 작은 변화가 생기며,

작은 구간만을 보았을 때는 선형적인 관계가 있다”

이 말을 곰곰이 생각을 해보면,

작은 변화의 관점에서는 선형적인 관계이기 때문에,

출력에서 생긴 오차를 반대로 입력 쪽으로 전파시키면서 w와 b등을 갱신하면 된다는 뜻이 된다. w와 b 값들을 무작위로 변화시키는 것이 아니라,

cost function이 결국 w와 b의 함수로 이루어졌기 때문에,

출력 부분부터 시작해서 입력 쪽으로, 즉 역 방향으로, 순차적으로 cost function에 대한 편미분을 구하고,

얻은 편미분 값을 이용해 w와 b의 값을 갱신시킨다.

모든 훈련 데이터에 대해서 이 작업을 반복적으로 수행을 하다 보면,

훈련 데이터에 최적화된 w와 b 값들을 얻을 수 있다.

역전파(backpropagation)란 용어는 출력부터 반대 방향으로 순차적으로 편미분을 수행해가면서

w와 b값들을 갱신시켜간다는 뜻에서 만들어진 것이다.

<역전파의 과정 종합>

앞서 설명한 것처럼, 역전파는 feed forward와 backpropagation의 2단계로 구성이 된다.

Feed forward 단계에서는 훈련데이타(x)를 신경망에 인가하고 출력단에서의 에러와 cost function 을 구한다.

신경망이 충분히 학습이 되지 못한 경우는 오차가 클 것이며,

이 큰 오차값을 backpropagation 시키면서 가중치와 바이어스 값을 갱신한다.

훈련 데이터에 대해서 반복적으로 이 과정을 거치게 되면,

가중치와 바이어스는 훈련데이타에 최적화 된 값으로 바뀌게 된다.

좋은 학습 결과를 얻으려면,

이 훈련 데이터가 한쪽으로 치우치지 않고 범용성을 가져야 한다

<Bonus - 요약 정보>

1. Neural Network – Backpropagation은?

- 최적의 학습 결과를 찾기 위해 역방향으로 에러를 전파(backward propagation of error)

2. Backpropagation 이해를 위한 Cost function 개념

- 입력된 훈련 데이터에 관한 실제 출력과 기대 출력간의 차이

- cost function이 최소값이 되도록(실제 출력과 기대 출력의 차이가 없도록) Backpropagation

3. Backpropagation 이해를 위한 Gradient-descent 개념

- cost function이 최소가 되도록 하는 방법

- cost function의 인자인 가중치(w)와 바이어스(b)를 조절함

4. Backpropagation의 두 단계

- feed forward: 입력 -> 출력(최종 출력단에서 에러와 cost function을 구함)

- backpropagation: 출력부터 반대 방향으로 순차적으로 편미분을 수행해가면서

뉴런의 가중치(w)와 바이어스(b) 값을 갱신