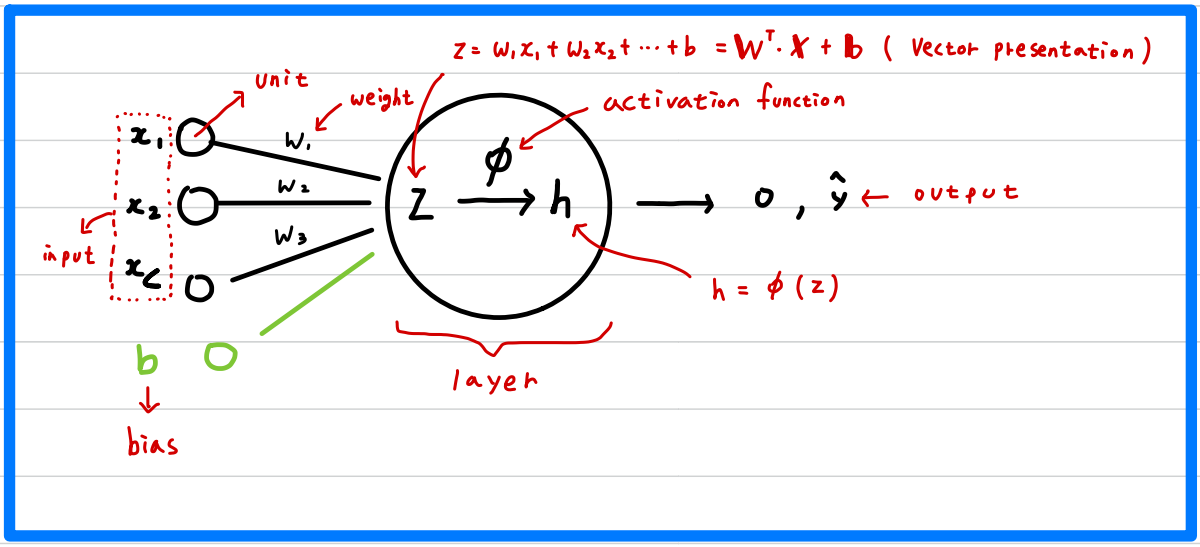
신경망 5분 리뷰

생김새, 용어



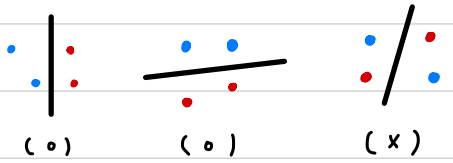
why activation function ?

↳ 신호가 다음 노드에 전달할만큼 강한 (의미있는 = 임계치를 넘은) 신호인지 검사하는 도구.

의사결정 나무의 조건식이 수행하는 역할과 의미상으로 비슷 (같은건 x)

why non-linear activation function ?

↳ 그림으로 이해 :



⇒ 선형 활성화 함수는 비선형적 검사를 진행할 수 없다.

↳ 수식 이해 : $\phi(z) = 10z$ 인 활성화 함수가 있을때

3개의 은닉층을 만들어 통과시킨다고 하자. $x \xrightarrow{w} \phi \rightarrow \phi \rightarrow \phi \rightarrow y$

결국 $y = \phi(\phi(\phi(wx))) = \phi(\phi(10 \cdot wx))$

$= \phi(10 \cdot 10 \cdot wx) = 10 \cdot 10 \cdot 10 \cdot wx$ 곱의 연산이 이루어 질텐데,

이는 $\phi_2(z) = 1000z$ 처럼 생긴 활성화 함수를 사용하여

하나의 은닉층을 쓰는것과 "학습" 측면에서 차이가 없다.

즉, 은닉층이 없는 단순한 케이스에 대해서만 학습이 가능하다.

↳ 이게 사실 Linear Regression 입니다.

How NN "learn" ?

↳ 문제에 맞는 손실을 정의하고 손실을 최소화 하는 방향으로 가중치를 갱신 !

↳ 값이 OO 만큼 차이가 난다,
△△개 중 □□개 맞혔네 ?
(MSE, Cross Entropy, ...)

But How ?

Back Propagation VS Optimizer

↳ 안다고 생각했는데 막상 말하려니 모르겠더라.

역전파는 NN 만의 특징이며 각 가중치가 어떤 값으로 갱신되어야 하는지 를 설명하는 수식* 이다

옵티마이저는 가중치 갱신이 어떤 방식* 으로 이루어져야 하는지에 대해 만들어 놓은 도구이다.

↳ 매 데이터마다 갱신할 것인가 ? (BGD, SGD, mini-BGD ...)
learning rate 를 다르게 할것인가 ? (Adagrad)
몇가지 트릭, 혹은 조합 (RMS prop, Adam)

Gradient vanishing ? Gradient exploding ?

↳ 위에서 빠르게 리뷰한 NN 에서 결국 어떤 한 가중치 W_i 는
어떤 옵티마이저를 통하든 기본적인 아이디어는 다음과 같다. $W_i := W_i - \alpha \cdot \frac{\partial E}{\partial W_i}$ (learning-rate)
여기서 E 는 모델의 총 오차를 뜻하며 $\frac{\partial E}{\partial W_i}$ 는 E 함수에 대한 W_i 편미분 값이다.

기울기 소실, 증폭 문제에서 "기울기" 는 이러한 편미분 값을 뜻한다. (W_i 변화가 E 오차에 얼마나 영향을 주었는가 ?)
즉 $\frac{\partial E}{\partial W_i} = 0$ or $\frac{\partial E}{\partial W_i} = \infty$ 와 같은 기울기 값을 가지는 것이 소실, 증폭 문제이다.

이러한 문제가 발생하면 1. $W_i^{new} = W_i^{old} + \alpha \cdot \frac{\partial E}{\partial W_i^{old}} = W_i^{old} + \alpha \cdot 0 = W_i^{old}$ (갱신 X)

2. $W_i^{new} = W_i^{old} + \alpha \cdot \frac{\partial E}{\partial W_i^{old}} = W_i^{old} + \alpha \cdot \infty = \infty$ (시스템 타아, $1 \div 0$ 과 같은)

과 같은 결과를 얻게된다.

Why does this happen ?

Common : while back propagating

- 1. activation function used in network (appears on DNN)
- 2 different type of back propagation , BPTT (appears on RNN)

1. Activation function used

$z = w_1 x + b$

$h = \phi(z)$

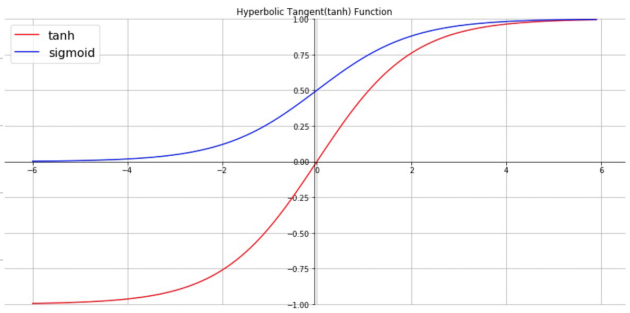
$o = w_2 h$

$f = err(y, o)$

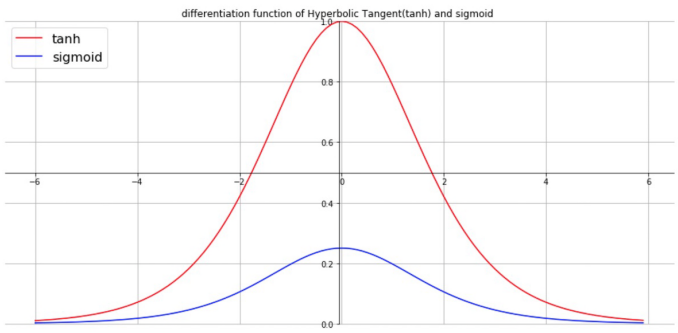
$$\Longrightarrow \frac{\partial f}{\partial w_1} = \frac{\partial f}{\partial o} \cdot \frac{\partial o}{\partial h} \cdot \frac{\partial h}{\partial z} \cdot \frac{\partial z}{\partial w}$$

$$= err'(y, o) \cdot w_2 \cdot \phi'(z) \cdot x$$

what if ϕ = sigmoid or tanh ?



$\frac{d}{dx}$ →

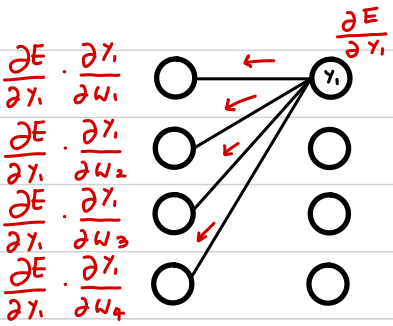


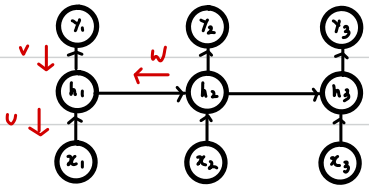
- sigmoid' : (0, 0.25]
- tanh' : (0, 1]

즉, sigmoid, tanh 와 같은 활성화 함수를 사용하는 경우
은닉층의 깊이가 조금만 깊어져도 $\phi'(z)$ 가 zero 에 가깝다.

ex) 10 깊이의 은닉층을 가진 신경망에서 각 층에 유닛이 1개씩만 있다 하더라도
sigmoid 를 쓸 경우 입력과 연결된 가중치에 전달되는 그래디언트는
"최대" $(\frac{1}{4})^{10} = \frac{1}{2^{20}}$ = wow

2. Different type of BP : BPTT





설명할 수 있을까..

DNN

각 가중치는 다른 오차(그래디언트)를 역전파 받고

업데이트 시점만 동시에 일어날뿐 값은 출력층에서부터

차례차례 이미 계산이 끝난다.