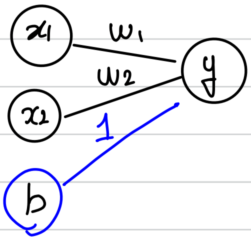
(w·b 같은 매개변수를 데이터로부터
Parameter
자동으로 학습. 설정)

2021 / 1 / 4

Perceptron (단층)

(의미있는 값을 얻지못함)
- 비선형 표현 불가. (복잡한 함수 표현 불가)
- w. b를 직접 손으로 정해줘야함.
- activation function 을 Step function

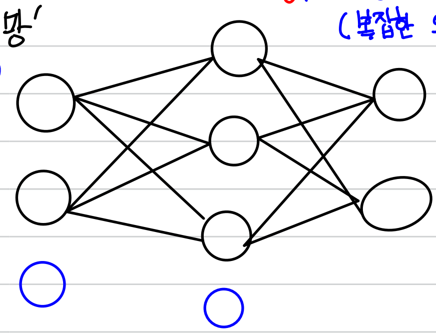


x_1 : 입력
y_1 : 출력
w : 가중치 (신호의 영향력 제어)
b : 편향 (뉴런, 노드가 얼마나 쉽게 활성화되는냐)

다층 퍼셉트론
신경망

- multi layer Perceptron 으로 복잡한 함수표현 가능
- 여러가지 activation function 사용 (비선형 표현 가능)
(복잡한 의미 가능)

'2층 신경망'
(가중치 층이 2개니까)



입력층 input layer | 은닉층 hidden layer | 출력층 output layer

$y_1 \cdot w_{11} + y_1 \cdot w_{12}$

$$y = \begin{cases} 1 & (x_1 \cdot w_1 + x_2 \cdot w_2 + b > 0) \\ 0 & (x_1 \cdot w_1 + x_2 \cdot w_2 + b \le 0) \end{cases}$$

$$y = h(x_1 \cdot w_1 + x_2 \cdot w_2 + b)$$

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

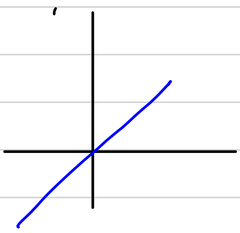
1) why activation function을 선형이 아닌 비선형으로 해야하나. ⇒ 층을 쌓는 의미가 없어진다.

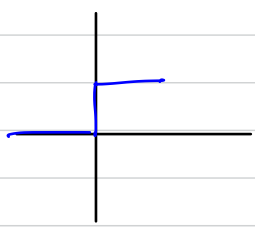
2) 풀고싶은 문제 골라가기.

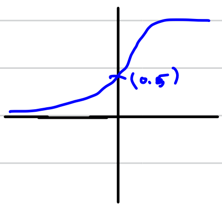
입력 신호의 총합을 출력신호로 변환하는 함수
= 활성화 함수
= activation function

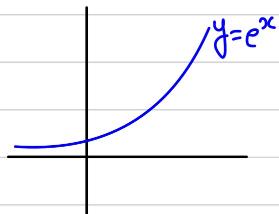
linear function	step function	ReLU function	Sigmoid function	Softmax function	...
$h(x) = x$	$h(x) = \begin{cases} 1 & (x>0) \\ 0 & (x \le 0) \end{cases}$	$h(x) = \begin{cases} x & (x>0) \\ 0 & (x \le 0) \end{cases}$	$h(x) = \frac{1}{1+e^{-x}}$	$y_k = \frac{e^{(a_k)}}{\sum_{i=1}^{n} e^{(a_i)}}$	

n = 출력층의 node 개수
y_k = 그중 k번째 출력









linear function:
- 예측. 회귀 문제
- 선형 함수 (직선)

step function:
- 비선형 (한 개의 직선으로 그릴수없는)
- 0 or 1 만 출력

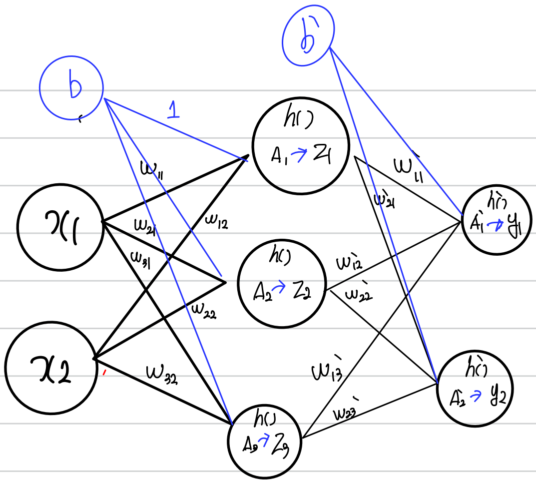
ReLU function:
- hidden layer

Sigmoid function:
- 이진 분류 문제
- 'S자 모양' 이란 뜻
- 비선형
- 연속값 (0, 0.284, 0.67, 1)

Softmax function:
- 다중 분류 문제
- 총합 1로서, 전체 확률로 볼수있음

ex)

관계형 데이터(table) = 열 vector (abc)
기계학습 = 컬럼 Vector $\begin{pmatrix}a\\b\\c\end{pmatrix}$



(for에 T필요, back에 T불필요)

w_{ij}

$$\begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \end{pmatrix}^T \begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix}$$

3X2 3X1

(for에 T불필요, back에 필요)

w_{ji}

$$\begin{pmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix}$$

2X3 (3X1)

$$(x_1\ x_2)\begin{pmatrix} w_{11} & w_{21} & w_{31} \\ w_{12} & w_{22} & w_{32} \end{pmatrix} + (b_1\ b_2\ b_3) = h((A_1 \cdot A_2 \cdot A_3))$$

1X2 2X3 1X3

$X \quad W \quad + B = A$

$\downarrow$

$$(z_1\ z_2\ z_3)\begin{pmatrix} w'_{11} & w'_{21} \\ w'_{12} & w'_{22} \\ w'_{13} & w'_{23} \end{pmatrix} + (b'_1\ b'_2) = h'((A'_1\ A'_2)) = (y_1, y_2)$$

1X3 3X2

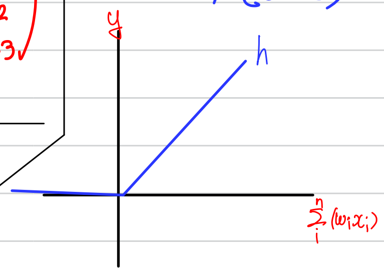
$h(A) = Z \cdot W' + b' = A'$

$h'(A') = y$

$Z = Wx + b$ (W_{ji} 표기법)

relu (Rectified Linear Unit, 경사 함수)

$h = \max(0, x)$



○ $h' = \begin{cases} 1 & (x > 0) \\ 0 & (x \le 0) \end{cases}$ Vanishing Gradient Problem solve!!

○ easy! 연산 비용이 적고, 구현이 매우 간단

○ 0 이하보다 작으면 죽는다.

Sigmoid

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

$$h' = \frac{0\cdot(1+e^{-x}) - 1(1+e^{-x})'}{(1+e^{-x})^2}$$

$$= \frac{e^{-x}}{(1+e^{-x})^2}$$

$$= \frac{1+e^{-x}}{(1+e^{-x})^2} - \frac{1}{(1+e^{-x})^2}$$

$$= \frac{1}{(1+e^{-x})} - \frac{1}{(1+e^{-x})^2}$$

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

$$= h(1-h)$$

확률 $0 < P < 1$

$odds(P) = \frac{P}{1-P}$ $\quad 0 < odds(P) < \infty$

$lg(odds(P)) = wx+b$ $\quad -\infty < lg(odds(P)) < \infty$

실수 범위이므로 선형 회귀분석 의미있음

$e^{wx+b} = odds(P)$

$e^{wx+b} = \frac{P}{1-P}$ $\quad = wx+b$

$(1-P)e^{wx+b} = P$

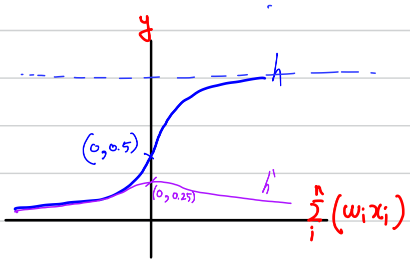
$e^{wx+b} - P\,e^{wx+b} - P = 0$

$P(1+e^{wx+b}) = e^{wx+b}$

$P = \frac{e^{wx+b}}{1+e^{wx+b}}$

$P = \frac{1}{1+e^{-(wx+b)}}$ → $\sum(w_i x_i)$

node 1개,



(실수값)

○ $(-\infty, \infty)$ 값을 $(0, 1)$로 표현 가능 → 확률표현 가능

○ Sigmoid 층을 여러 번 쓰다면 기울기의 최대값이 0.25 이므로 지나갈때 마다 미분값이 작아짐

⇒ 오차율을 계산하기 어려워짐

⇒ Vanishing Gradient Problem (작아지거나 0이되면 안됨) step function

나중에 back Propagation에서 구체적으로

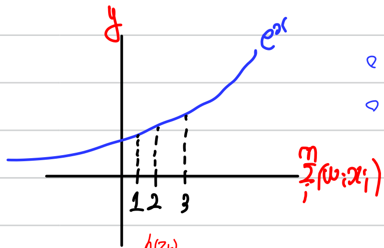
Binary classification에서 0.5 기준으로 넘으면 a, 안넘으면 b 2개 가능.

Softmax

$$h_k = \frac{\exp(a_k)}{\sum_{i=1}^{n} \exp(a_i)}$$

n: 출력층 node 개수
구분할 class의 개수

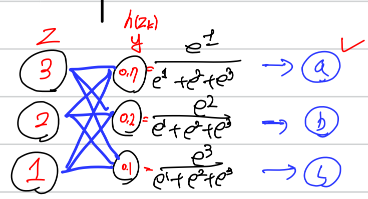
k: 출력층 node의 k번째.



○ 지수함수이기에 w가 더 커지는 효과

○ 각 node 값의 합이 1이기에 확률표현 가능.

○ 단조증가 함수이기에 대소 관계는 그대로임!



ⓐ 겠구나!!

$x \cdot w + b = Z = (z_1, z_2, z_3)$

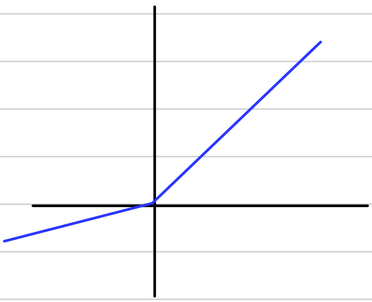
$$Softmax(z) = \left(\frac{e^{z_1}}{\sum_i^3 e^{z_i}}, \frac{e^{z_2}}{\sum_i^3 e^{z_i}}, \frac{e^{z_3}}{\sum_i^3 e^{z_i}}\right)$$

$= (P(a), P(b), P(c))$ $\quad (P(a)+P(b)+P(c) = 1)$

$= (0.7, 0.2, 0.1)$

Leaky ReLU

$h = \max(0.01x, x)$

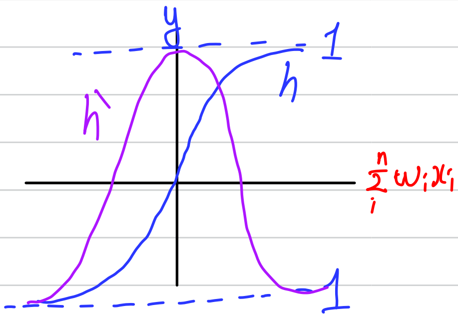


○ ReLu의 문제점 해결
($x<0$ 는 기울기가 0이 되어
뉴런. 노드가 죽는 현상)

Tanh

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

$h' = (1-\tanh(x))(1+\tanh(x))$



○ 기울기의 최대값이 1로 고쳐졌지만
여전히 낮은 값들로 인해 층을 지날수록
Vanishing gradient Problem이 생길 수 있음

○ 수치형 자료 → 연속형 자료 ex) 키. 몸무게
이산형 자료 ex) abs(키. 몸무게), 교통사고 건수.(16회)

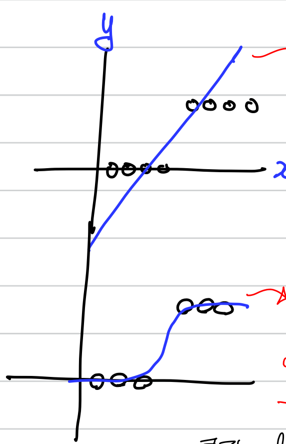
○ 범주형 자료. → 순위형 자료 순서대로 나열 ex) 학점 (A=2, B=4, C=6..)
명목형 자료 ex) 성별 (남=1, 여=2), 종교 (기독교=1, 불교=2, 천주교=3)
ex) 혈액형

How to seperate Linear regression and Logistic Regression?

⇒ what is data! (종속변수 y)

if) data is numerical data ⇒ Linear regression $(-\infty, \infty)$

if) data is categorical data ⇒ Logistic Regression (0. 1)
false True



$Y = ax+b$

Linear regression 으로는 0과 1로 이루어진 categorical data엔 안 맞지?

(가정)
$P(Y=1) = ax+b$

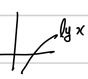
로짓 변환을 통해
$0 < P(Y=1) < 1$ 을
$-\infty < P(Y=1) < \infty$ 로!

로짓 = log + odds의 약자
(= 실패 비율 대비 성공비율)
($\frac{P}{1-P}$ (P=성공확률))

$0 < P < 1$

$0 < \frac{P}{1-P} < \infty$

$-\infty < \log\left(\frac{P}{1-P}\right) < \infty$



$\log\left(\frac{P}{1-P}\right) = ax+b$

$e^{ax+b} = \frac{P}{1-P}$

$P = \frac{1}{1+e^{-(ax+b)}}$

P에 대해 역변환.
sigmoid function

∴ 그래서 이진분류에 능함!