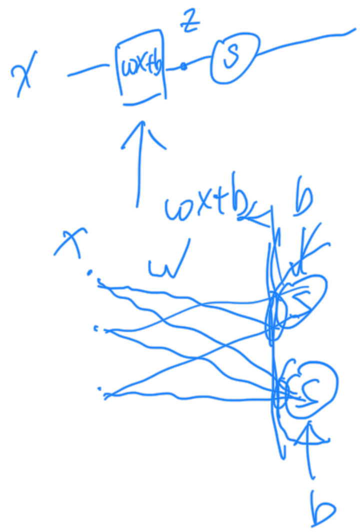
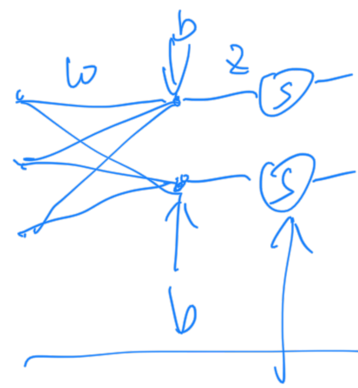
Lecture 2

1. Actuation Functions



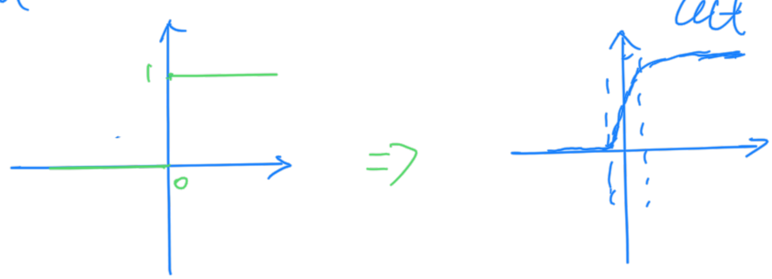
$z = wx + b$ — vector

w: matrix, x: vector



act

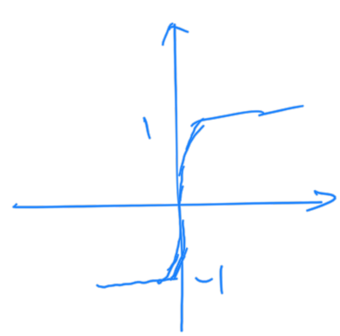
sigmod



$$g(x) = \frac{e^x}{1+e^x} \quad \text{or} \quad \frac{1}{1+e^{-x}}$$

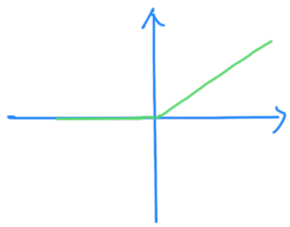
tanh

$$\tanh = 2g(2x) - 1$$



$$\tanh = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

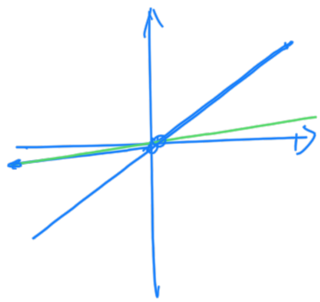
ReLU — Rectified Linear Unit



$$ReLU(x) = max(0, x)$$

$$\begin{cases} x & \text{when } x > 0 \\ 0 & \text{else} \end{cases}$$

Leaky ReLU



$$max(x, 0.1x)$$

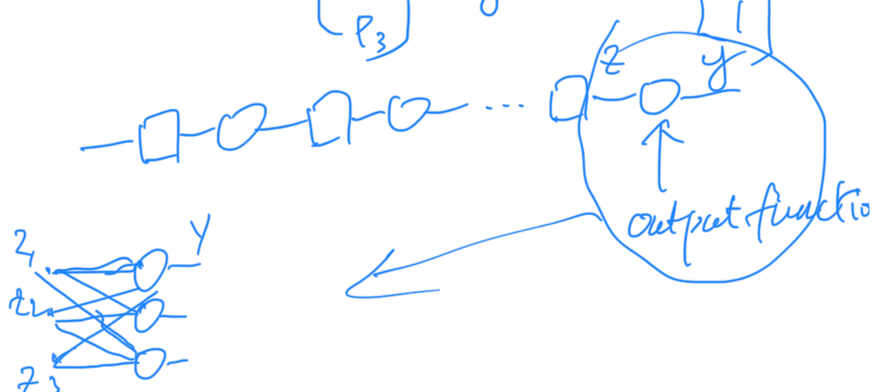
$$max(w_1 x, w_2 x)$$

2. Output function

For classification

$$\begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} \quad \text{ground truth} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

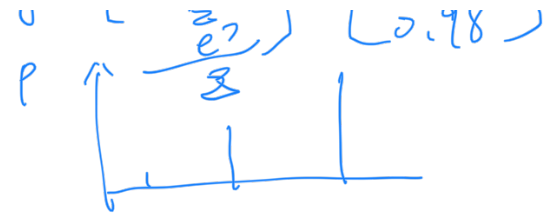
softmax



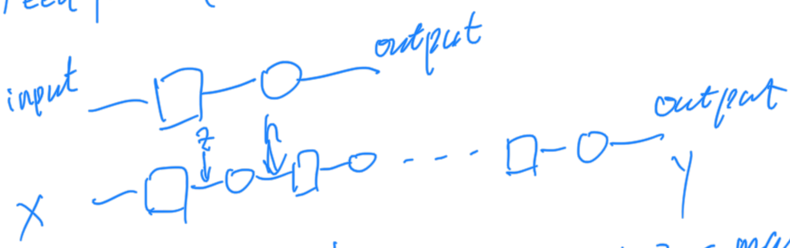
$$y_i = \frac{e^{z_i}}{\sum e^{z_i}}$$

$$z = \begin{bmatrix} 1 \\ 3 \end{bmatrix} \Rightarrow y = \begin{bmatrix} \frac{e^1}{e^1+e^3+e^?} \\ \frac{e^3}{\ldots} \end{bmatrix} = \begin{bmatrix} 0.002 \\ 0.02 \\ 0.? \end{bmatrix}$$

[7]



3. Feed forward



$z^1 = \underline{w} x + b$ $\quad$ w — matrix

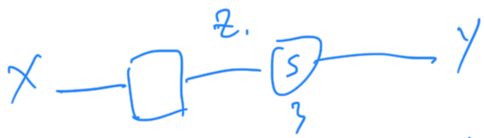
$h^1 = \text{sigmoid}(z^1)$ $\quad$ b, x — vector

$\quad$ h^i, z^i — vector

$z^2 = w h^1 + b$

$h^2 = \text{sigmoid}(z^2)$

⋮



x is a vector of five dimensions $\quad$ $x = \begin{bmatrix} x \\ \vdots \\ x \end{bmatrix}$

y — — — three — — $\quad$ $y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$

$\underset{5}{x}$ $\quad$ $\underset{3}{z}$

$\underset{3\times1}{z} = \underset{3\times5}{w}\ \underset{5\times1}{x} + \underset{3\times1}{b}$

$z = wx + \underline{b}$ $\quad$ $h = \delta(z)$

5×1

1 : $\underline{W_1}$ 10 neurons $50+10$

10×5

2 : W_2 $200+20$

20×10

3 : W_3 $60+3$

3×20

4. Back propagation

$$y = a_0 x + a_1$$

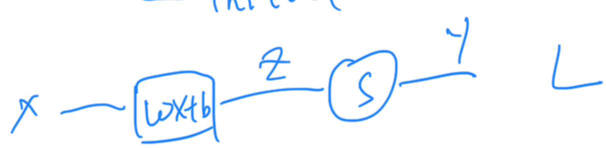
$$L = (y-\hat{y})^2$$

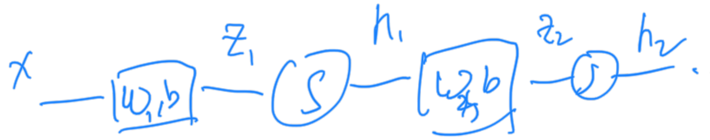
$$a_0^1 = a_0^0 - \lambda \frac{\partial L}{\partial a_0}$$

derivative

step size

initial





$$w^1 = w^0 - \lambda \frac{\partial L}{\partial w}$$

step size

initial

$$[w_1, w_2, \cdots w_n]$$

$$\begin{cases} w_1^i = w_1^{i-1} - \lambda \dfrac{\partial L(w_1^{i-1})}{\partial w_1} \\ \vdots \\ w_n^i = w_n^{i-1} - \lambda \dfrac{\partial L(w_n^{i-1})}{\partial w_n} \end{cases}$$

n layers

$$\frac{\partial L(w_n^{i-1})}{\partial w_n} = \frac{\partial L}{\partial y}\frac{\partial y}{\partial w}$$

$$= \frac{\partial L}{\partial y}\frac{\partial y}{\partial z_n}\frac{\partial z_n}{\partial w_n}$$

X — $\boxed{w_1}$ z_1 ○ h_1 $\boxed{w_2}$ ○ h_2 … $\boxed{w_{n-1}}$ z_{n-1} ○ h_{n-1} $\boxed{w_n}$ z_n ○ y L

no w_n

$$\frac{\partial L}{\partial w_n} = \frac{\partial L}{\partial y}\frac{\partial y}{\partial z_n}\underbrace{\frac{\partial z_n}{\partial w_n}}_{h_{n-1}}$$

$$\frac{\partial z_n}{\partial w_n} = \frac{\partial (w_n h_{n-1} + \cancel{b_n})}{\partial w_n} = h_{n-1}$$

$L = (y-\hat{y})^2 \quad y =$

$$\frac{\partial L}{\partial w_{n-1}} = \underbrace{\frac{\partial L}{\partial y}\frac{\partial y}{\partial z_n}}_{\Delta}\underbrace{\frac{\partial z_n}{\partial h_{n-1}}}_{w_n^{i-1}}\frac{\partial h_{n-1}}{\partial z_{n-1}}\underbrace{\frac{\partial z_{n-1}}{\partial w_{n-1}}}_{h_{n-1}}$$

training data set

$\{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$ m samples.

~~迭代~~ initialization

n layers

$\{w_1^0, w_2^0, \ldots w_n^0\}$

$$\vdots \qquad \vdots \qquad \circ$$

it = 1

to get $\{ W_1^1, W_2^1, \cdots W_n^1 \}$

$$W_n^1 = W_n^0 - \lambda \frac{\partial L}{\partial W_n}(W_n^0)$$

$$\frac{\partial L}{\partial W_n} = \frac{\partial L}{\partial y} \frac{\partial y}{\partial z_n} \frac{\partial z_n}{\partial W_n}$$

feed forward to compute L

$$W_{n-1}^1 = W_{n-1}^0 - \lambda \frac{\partial L}{\partial W_{n-1}}(W_1^0, W_2^0, \cdots W_n^0)$$

$$\vdots$$

$$W_1^1 = W_1^0 - \lambda \frac{\partial L}{\partial W_1}(W_1^0, \cdots W_n^0)$$

5. Derivatives of Activation Functions

sigmoid $\quad y = \frac{1}{1+e^{-z}}$ or $\frac{e^z}{1+e^z}$

$$\frac{\partial y}{\partial z} = \frac{-1(-e^{-z})}{(1+e^{-z})^2}$$

$$= \underbrace{\frac{1}{1+e^{-z}}}_{y} \cdot \underbrace{\frac{e^{-z}}{1+e^{-z}}}_{1-y}$$

$$= y(1-y)$$

softmax $\quad y_i = \frac{e^{z_i}}{\sum e^{z_i}} = \frac{e^{z_i}}{c+e^{z_i}}$

$$\frac{\partial Y_i}{\partial z_i} = Y_i(1-Y_i) \qquad \sum_{i}^{\uparrow} e^{z_k}$$

6. Loss functions

For regression

$$L = \frac{1}{2}(y-\hat{y})^2$$

$$= (y-\hat{y})^T(y-\hat{y})$$

$$\frac{\partial L}{\partial y} = -(y-\hat{y})$$

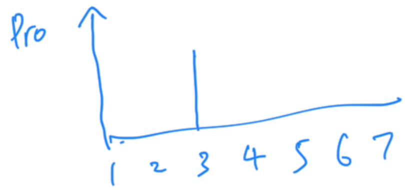
Huber

$$L = \begin{cases} \frac{1}{2}(y-\hat{y})^2 & \text{if } |y-\hat{y}| < c \\ c\left(|y-\hat{y}| - \frac{1}{2}c\right) & \end{cases}$$

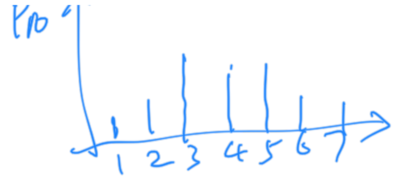
Pseudo Huber

$$L = c^2\left(\sqrt{1+\left(\frac{y-\hat{y}}{c}\right)^2} - 1\right)$$

cross entropy Loss



ground truth



Estimation result

$\Sigma = 1$

$$L = -\sum_i t_i \log(y_i)$$

target/label

estimate

ground truth $\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$, estimated $\begin{bmatrix} 0.002 \\ 0.02 \\ 0.98 \end{bmatrix}$

$$L = -\left(0 + 0 + \log(0.98) \right) = 0.02$$

log — natural log

if estimated is $\begin{bmatrix} 0.33 \\ 0.33 \\ 0.33 \end{bmatrix}$

$$L = -\left(0 + 0 + \log(0.33) \right) = 1.1$$