

$$z_1 = f(w_{11}x_1 + w_{12}x_2 + w_{13}x_3 + b)$$

$$z_j = f(\sum_i w_{ij}x_i + b_j)$$

$$W' = \begin{pmatrix} w_{11} & w_{12} & w_{13} & \dots & w_{1d} \\ w_{21} & w_{22} & w_{23} & \dots & w_{2d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{h1} & w_{h2} & w_{h3} & \dots & w_{hd} \\ \dots & \dots & \dots & \dots & \dots \\ w_{dh} & \dots & \dots & \dots & w_{dd} \end{pmatrix}$$

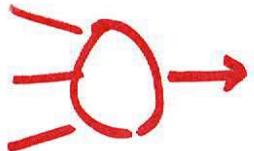
$d$   
 $h' \times d$   
matrix

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{pmatrix}$$

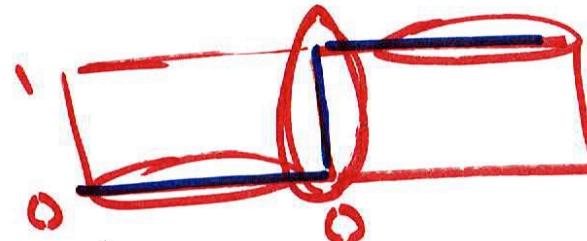
$$\hat{y} = f(-f(f(W^1)^T (W^2)^T f(W' x + b') + b^2) + b^3) \dots$$

Diagram illustrating the dimensions of the layers:

- $W'$  is a  $h' \times d$  matrix.
- $x$  is a  $d \times 1$  vector.
- The input  $x$  is multiplied by  $W'$  to produce a  $h' \times 1$  vector.
- This vector is then passed through a function  $f$  to produce a  $h' \times 1$  vector.
- The result of the first layer is a  $h' \times d$  matrix.
- This matrix is multiplied by  $W^1$  to produce a  $h^1 \times d$  matrix.
- This matrix is then passed through a function  $f$  to produce a  $h^1 \times 1$  vector.
- The result of the second layer is a  $h^1 \times d$  matrix.
- This matrix is multiplied by  $W^2$  to produce a  $h^2 \times d$  matrix.
- This matrix is then passed through a function  $f$  to produce a  $h^2 \times 1$  vector.
- The final output is a  $h^2 \times 1$  vector.

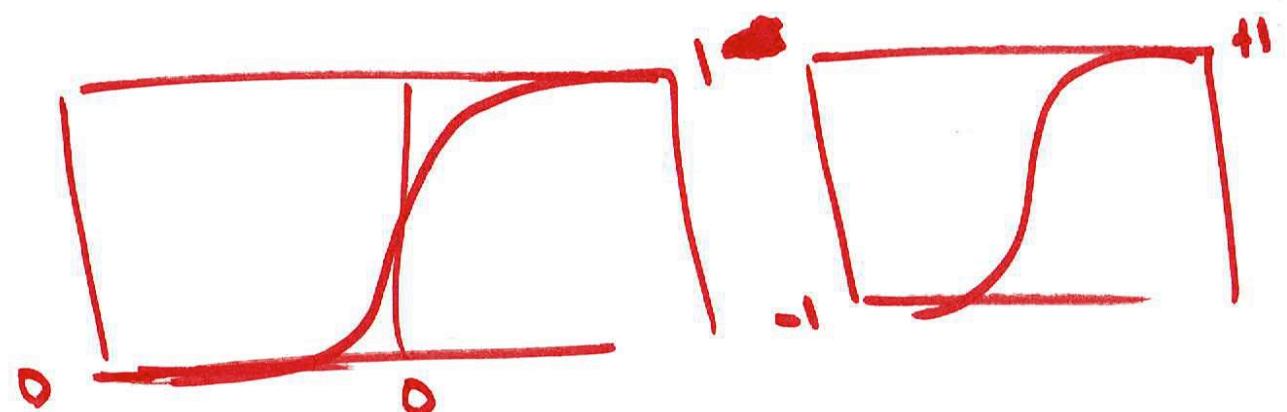


Step ( $\omega_1 x_1 + \omega_2 x_2 + \omega_3 x_3 + b$ )



Sigmoid

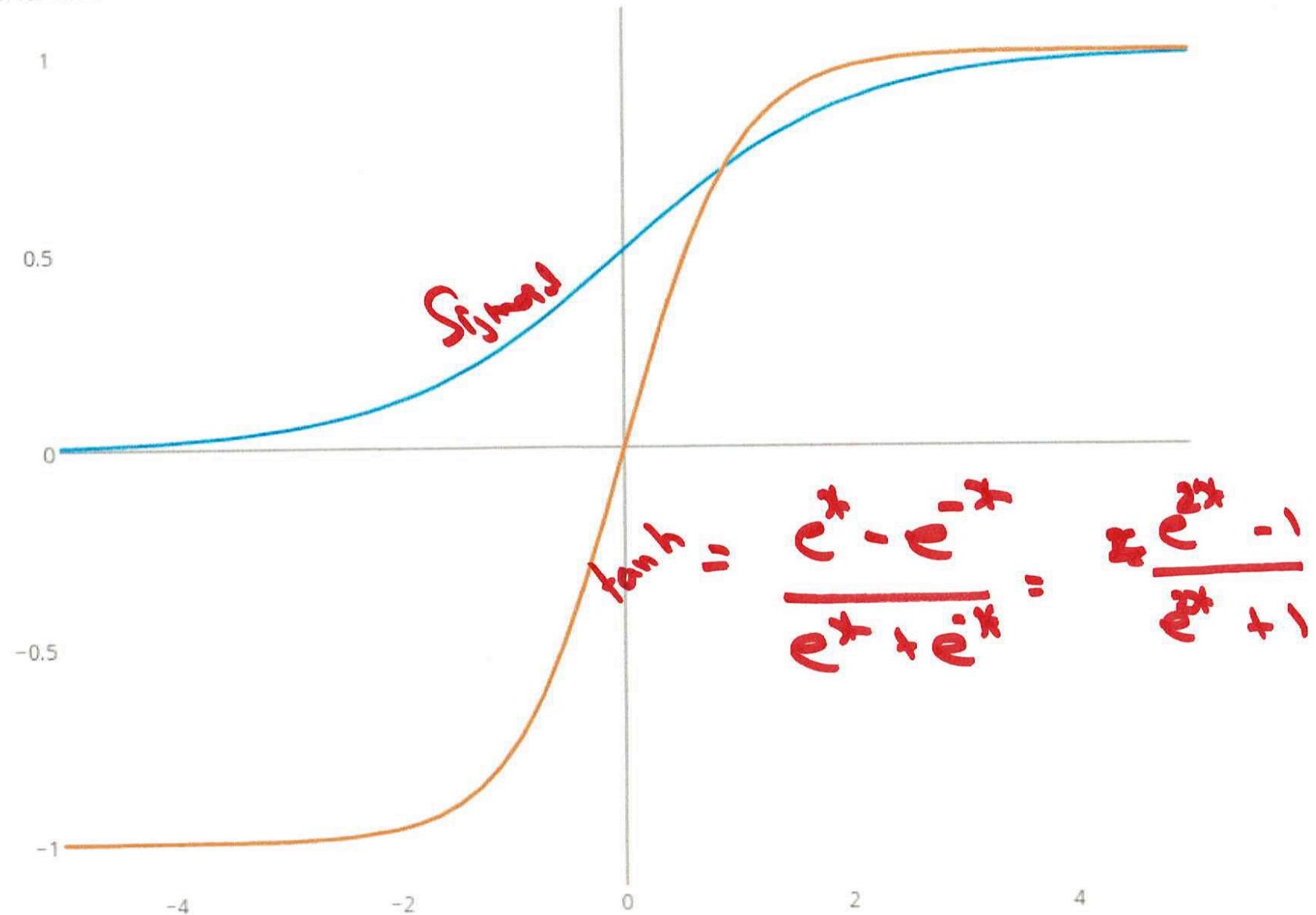
$$\sigma(x) = \frac{1}{1+e^{-x}}$$
$$\therefore \frac{e^x}{e^x + 1}$$



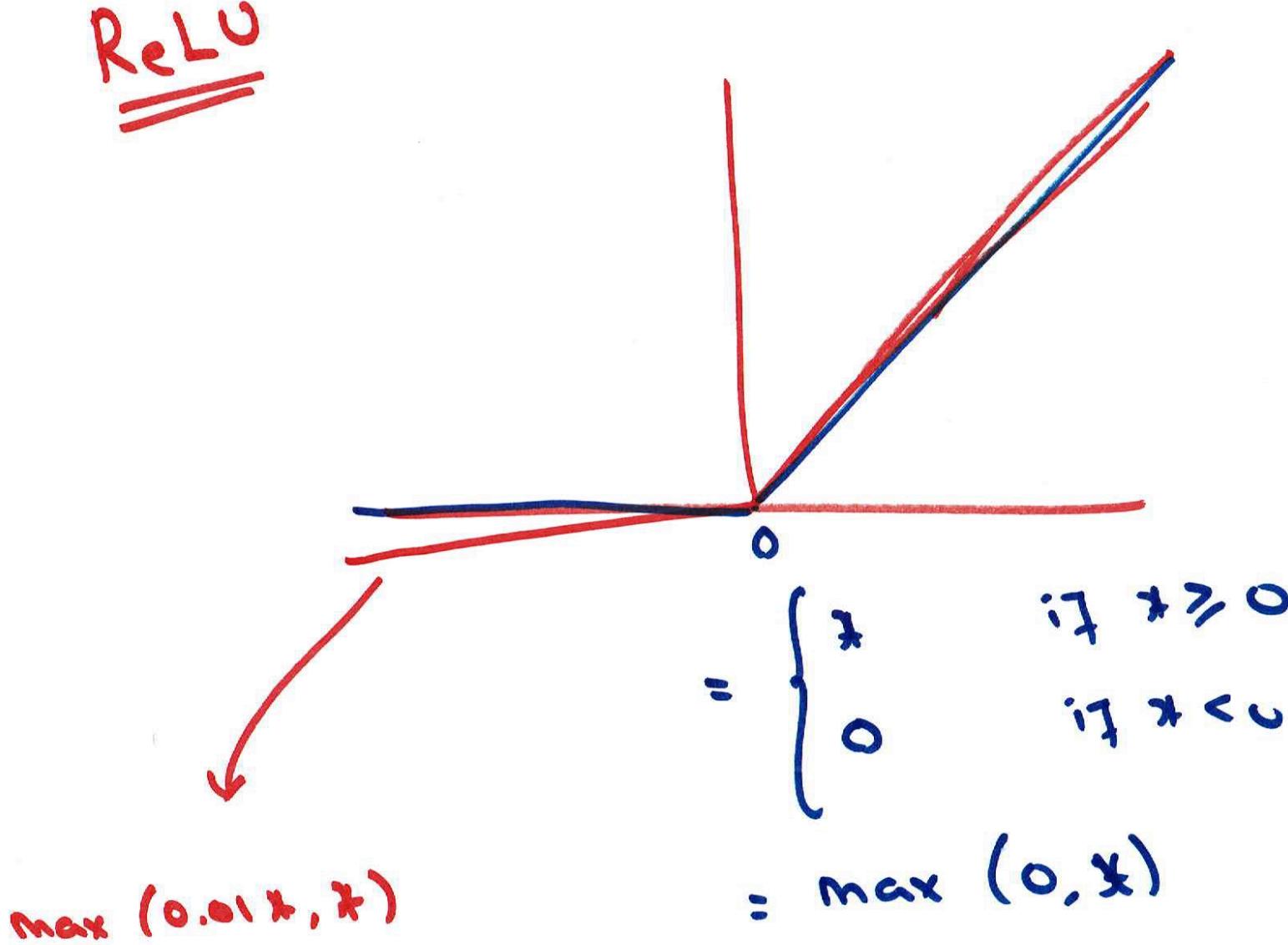
$$2\sigma(x) = 1$$

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

- Sigmoid function
- Tanh function

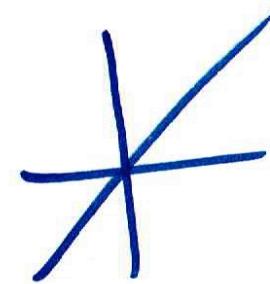


ReLU



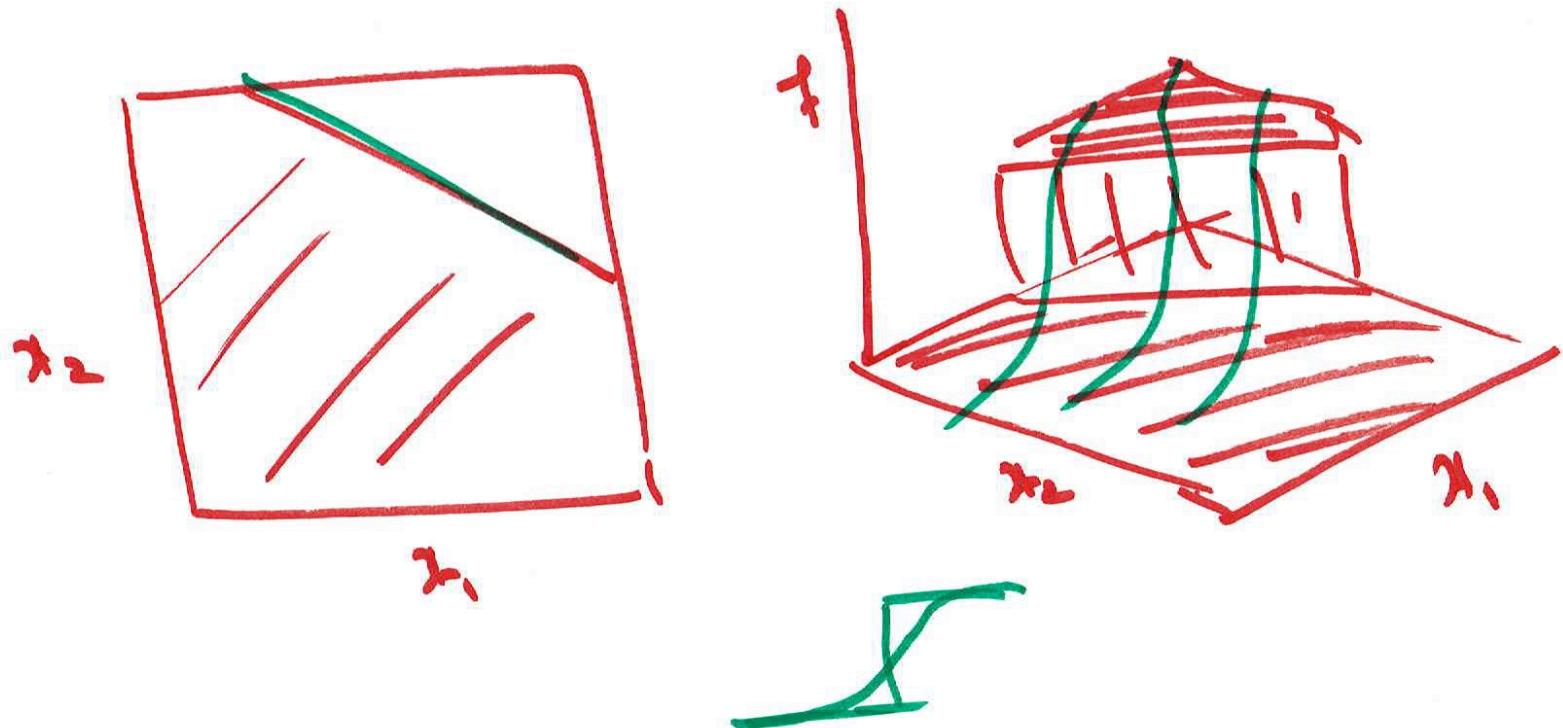
linear

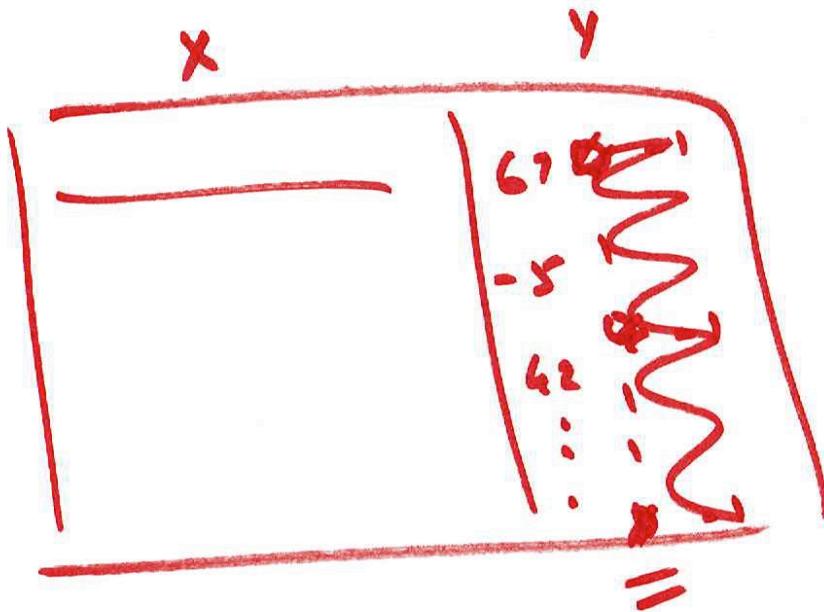
= \*



$$\text{Step } = f(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$$

$f(w_0 + w_1 x_1 + w_2 x_2 + b)$





Output nodes

---

hidden layer

Classification

Sigmoid, tanh  
Softmax

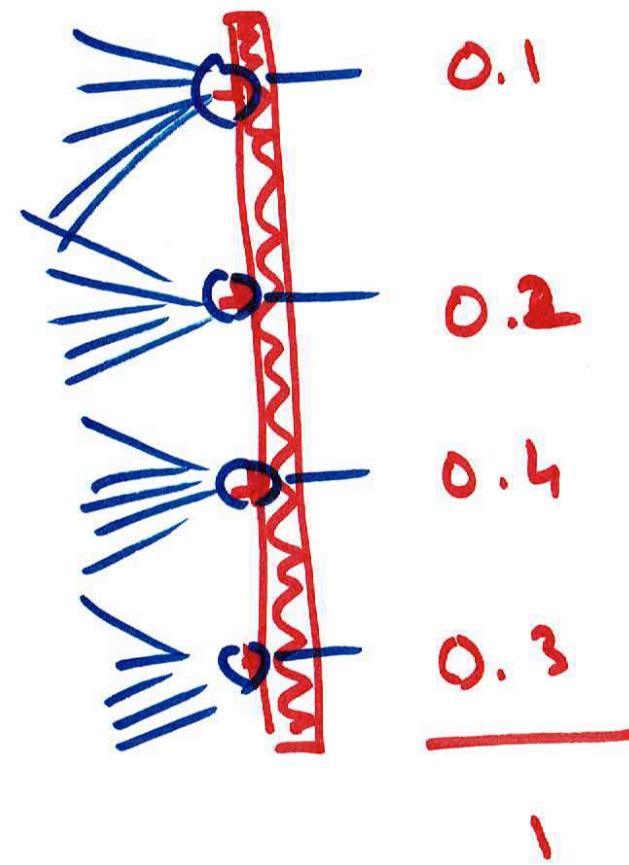
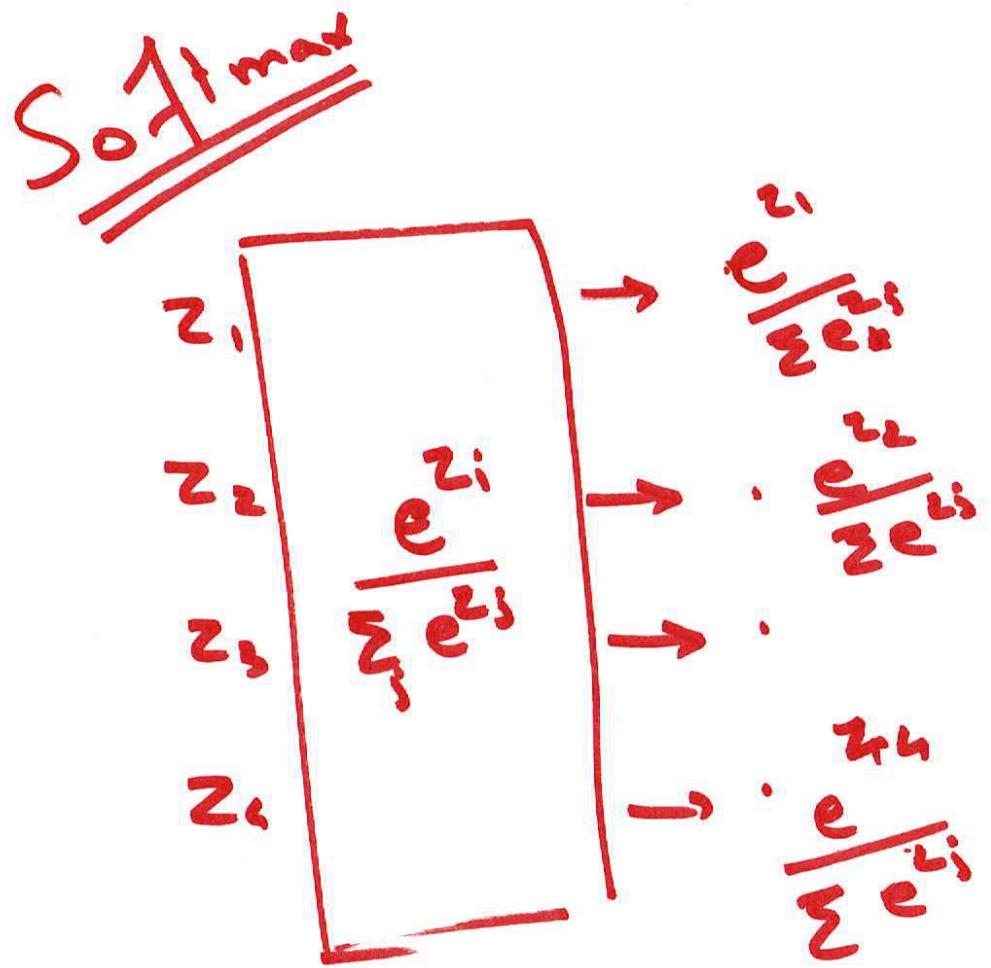
Sigmoid ✓  
tanh ✓

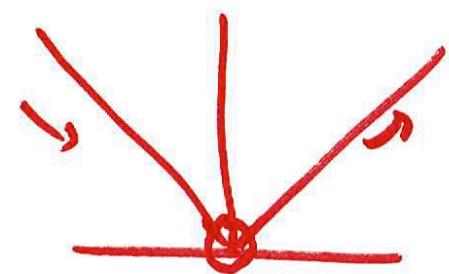
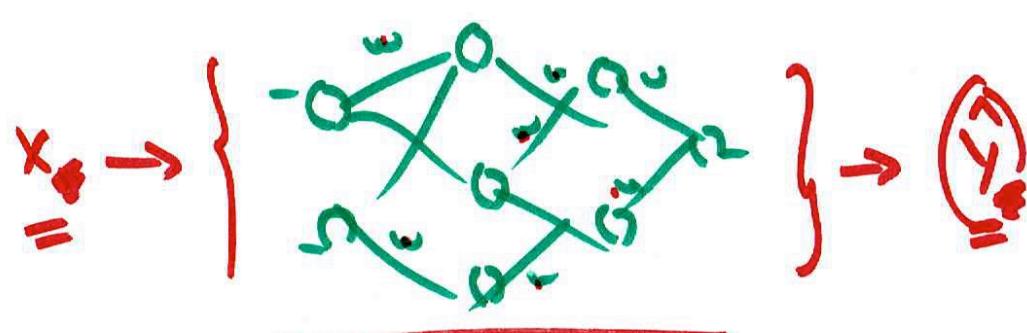
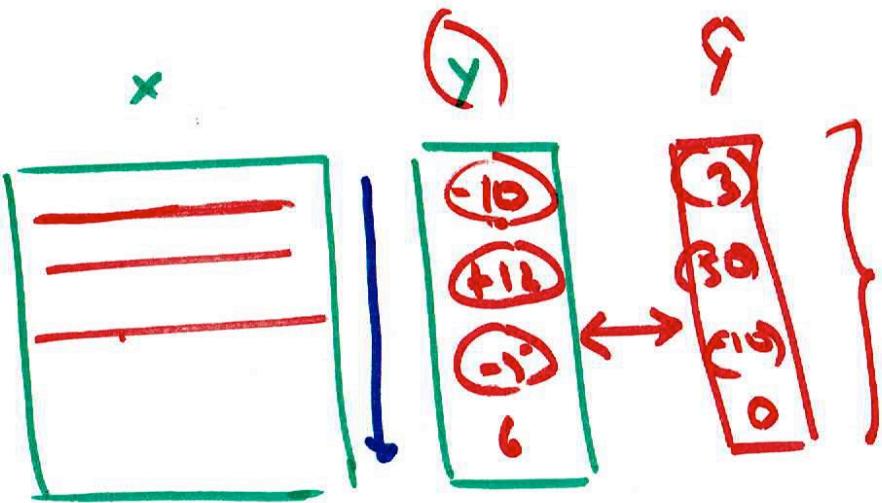
ReLU ✓

linear

Reg  $\rightarrow$  linear

$$\hat{a} + \hat{b}(a + b x)$$





Loss function

$$L(y, \hat{y}) = \frac{1}{n} \sum_i (y_i - \hat{y}_i)^2$$

Reg

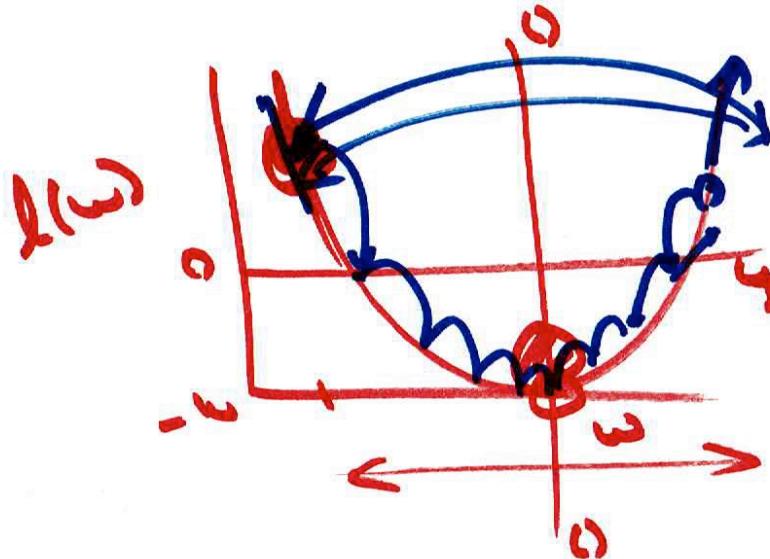
$$L(y, \hat{y}) = L(w)$$

$L_2$  loss  
MSE  
SSE

$$\text{Classification Loss} \quad L(y, \hat{y}) = -[y_i \log(\hat{y}_i) + (1-y_i) \log(1-\hat{y}_i)]$$

Cross entropy loss

How  $\min \underline{L(y, \hat{y})}$  by changing  
by my  $w^1, w^2, \dots, w^n$



$$y = x^2 - 10 = -10$$

$$\frac{dy}{dx} = 2x = 0$$

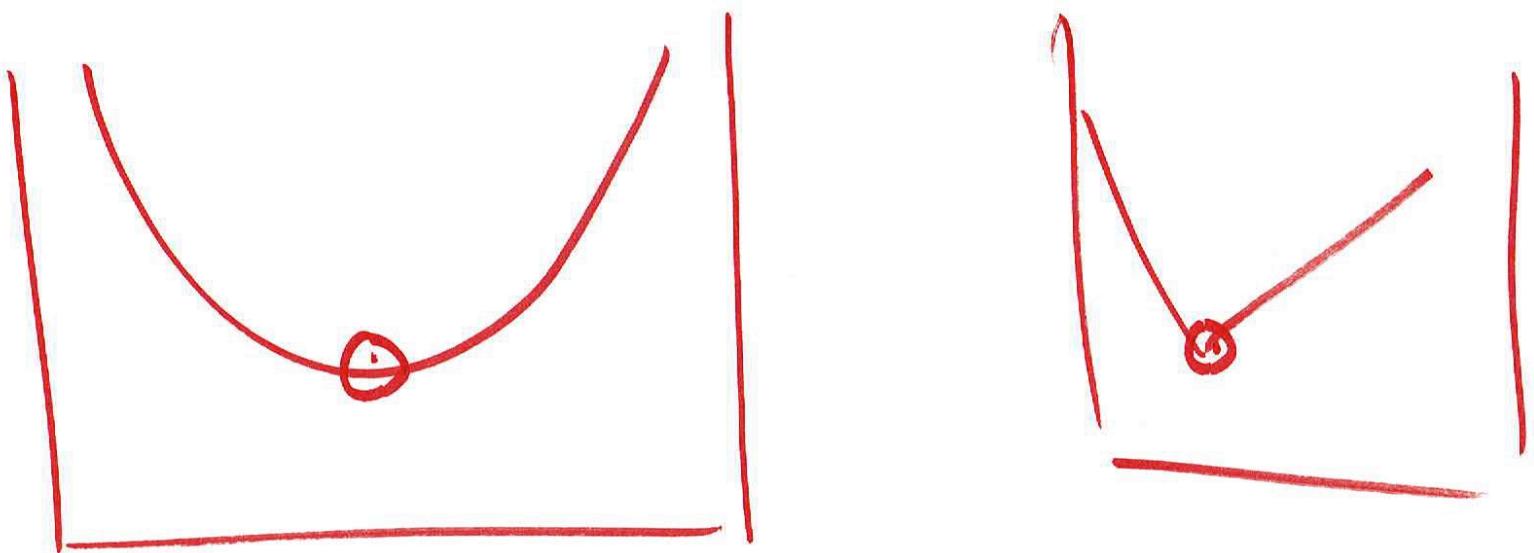
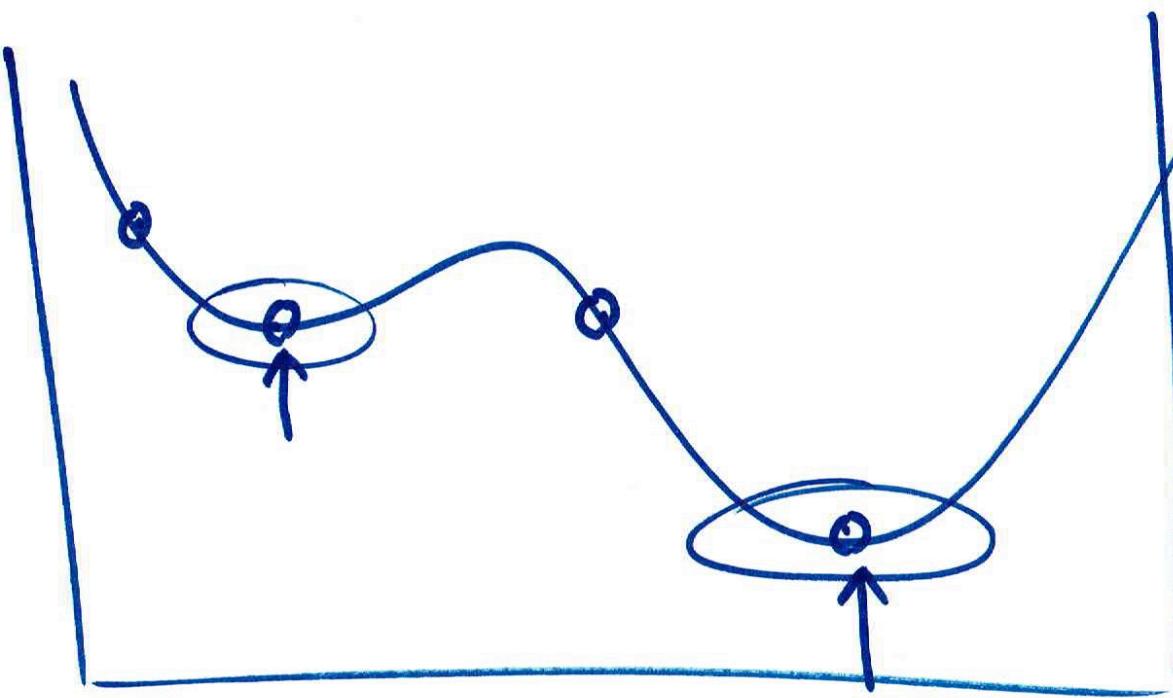
$$x = 0$$

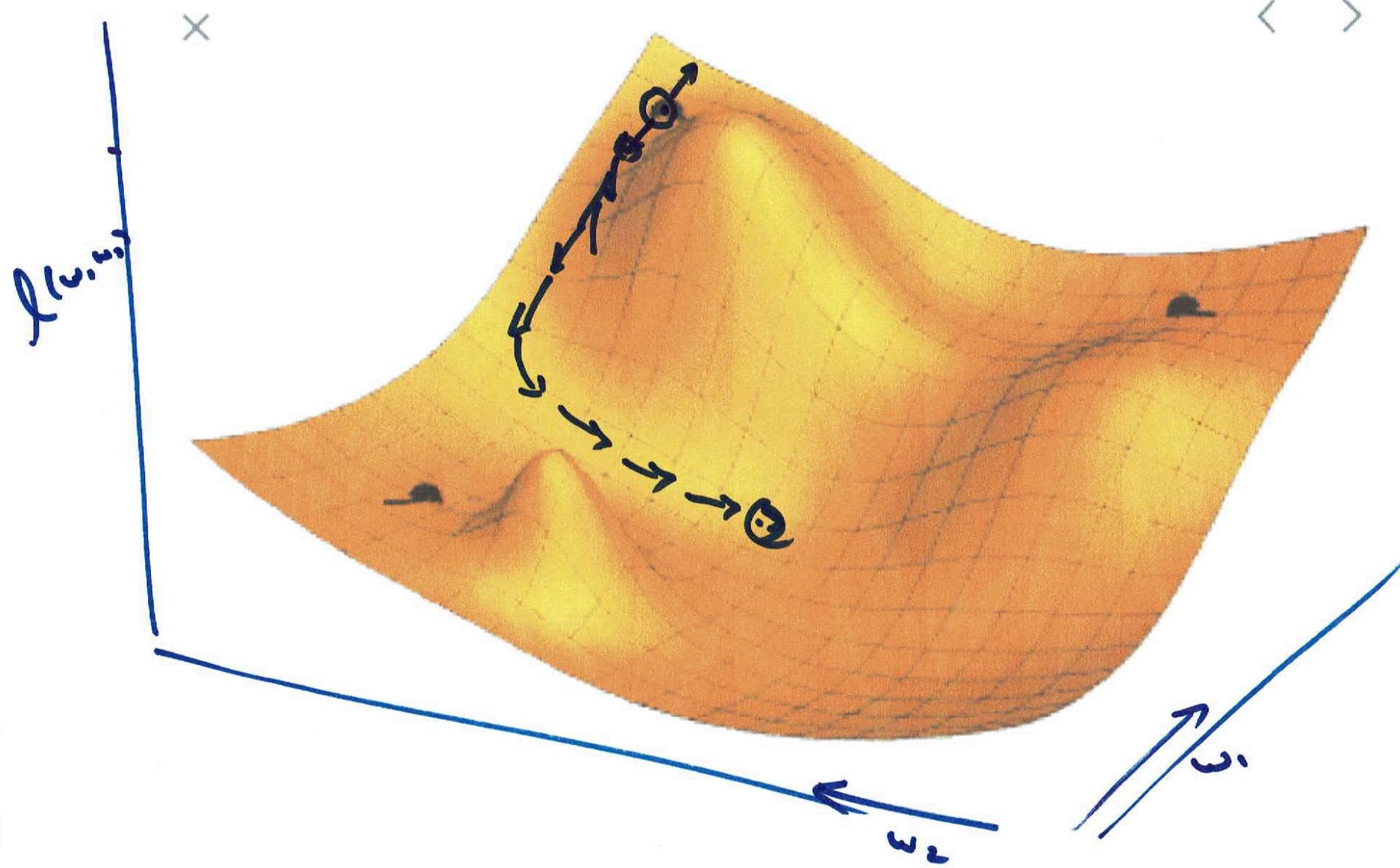
$$\frac{dL}{dw} = \boxed{\text{---}} = 0$$

$$\frac{dl}{dw}$$

$$w^{new} = w - \eta \nabla_w l$$

↑ learning rate





$$\text{if } \omega^{\text{new}} = \omega^{\text{old}} - \frac{\eta \nabla_{\omega} l(\omega)}{N}$$

$$= \omega^{\text{old}} - \frac{1}{N} \sum_i \nabla_{\omega} l_i(\omega) \leftarrow$$

(S&D)

$$\omega^{\text{new}} = \omega^{\text{old}} - \eta \nabla_{\omega} l_i(\omega) \leftarrow$$

$$\boxed{\omega^{\text{new}} = \omega^{\text{old}} - \frac{1}{N} \eta \sum_i \nabla_{\omega} l_i(\omega)}$$

↓  
over  
a min  
batch

Loss

function of w ( $l(w)$ )

$$L = \frac{1}{N} \sum_i \ell(y_i - f(\dots, f(w^2 f'(w^1 x + b) + b^2), \dots))$$

Chain Rule

$$f(g(h(x)))$$

$$\frac{df}{dx} = \boxed{\frac{df}{dg} \cdot \frac{dg}{dh} \cdot \frac{dh}{dx}}$$

Back propagation