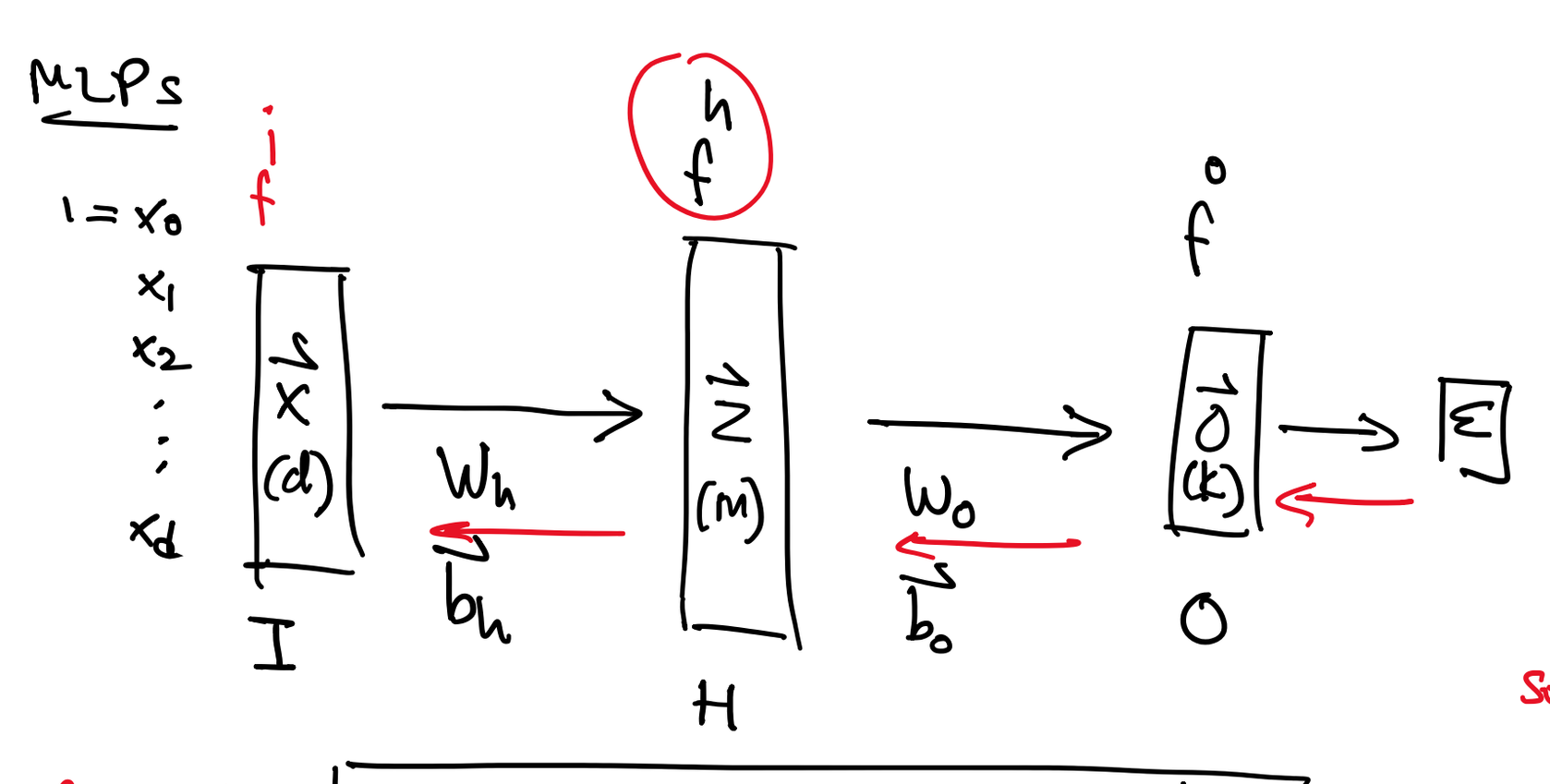
Lecture 20

Monday, November 18, 2024 9:57 AM

MLPs



$k = 1$

linear $f^o \Leftarrow\Rightarrow$ linear regression

sigmoid $f^o \Leftarrow\Rightarrow$ binary logistic

or general k

softmax $f^o \Leftarrow\Rightarrow$ multiclass logistic

linear $f^o \Leftarrow\Rightarrow$ multivariate regression

$f^i \Leftrightarrow$ linear / identity

$$W_h \in \mathbb{R}^{d\times m} \quad W_o \in \mathbb{R}^{m\times k}$$
$$\vec{b}_h \in \mathbb{R}^m \quad \vec{b}_o \in \mathbb{R}^k$$

Single neuron perspective

$1 = x_0, x_1, \dots, x_d \to z_i$

$$net_i = \omega_0 + \omega_{1i} x_1 + \omega_{2i} x_2 + \dots + \omega_{di} x_d = \omega_0 + \vec{\omega}_i^T \vec{x}$$
$$z_i = f^h(net_i)$$

I) forward pass

0) input $\vec{x}$

1) Compute $\vec{net}_z = \begin{pmatrix} net_1 \\ net_2 \\ \vdots \\ net_m \end{pmatrix}$ $\qquad \vec{net}_z = W_h^T \vec{x} + \vec{b}_h$

2) $\vec{z} = f^h(\vec{net}_z) = \begin{pmatrix} f^h(net_1) \\ f^h(net_2) \\ \vdots \\ f^h(net_m) \end{pmatrix}$ $\qquad \vec{z} = f^h(\vec{net}_z) = f^h(W_h^T \vec{x} + \vec{b}_h)$

3) $\vec{z}$ is the "input" to the $\vec{o}$ layer

$\vec{net}_o = \begin{pmatrix} net_{o_1} \\ net_{o_2} \\ \vdots \\ net_{o_k} \end{pmatrix}$ $\qquad \vec{net}_o = W_o^T \vec{z} + \vec{b}_o$

4) $\vec{o} = f^o(\vec{net}_o) = f^o(W_o^T \vec{z} + \vec{b}_o)$

5) Error or Loss Computation

$$\mathcal{E} = \frac{1}{2}\|\vec{y} - \vec{o}\|^2 \quad \text{(squared error)}$$

$\vec{y}$: target vector; $\vec{o}$: predicted vector

$(\vec{x}, \vec{y})$ → $\vec{o}$

II) Back propagation of error

$$\vec{o} = f^o\Big(W_o^T\big(f^h(\underbrace{W_h^T \vec{x} + \vec{b}_h}_{\vec{net}_z})\big) + \vec{b}_o\Big)$$

(the whole argument of f^o: $\vec{net}_o$)

$\nabla_{W_h} = \frac{\partial \mathcal{E}}{\partial W_h}$ (matrix $d\times m$) $\quad \nabla_{b_h} = \frac{\partial \mathcal{E}}{\partial b_h}$ (vector m) $\quad \nabla_{W_o} = \frac{\partial \mathcal{E}}{\partial W_o}$ (matrix $m\times k$) $\quad \nabla_{b_o} = \frac{\partial \mathcal{E}}{\partial b_o}$ (vector k)

gradient descent:

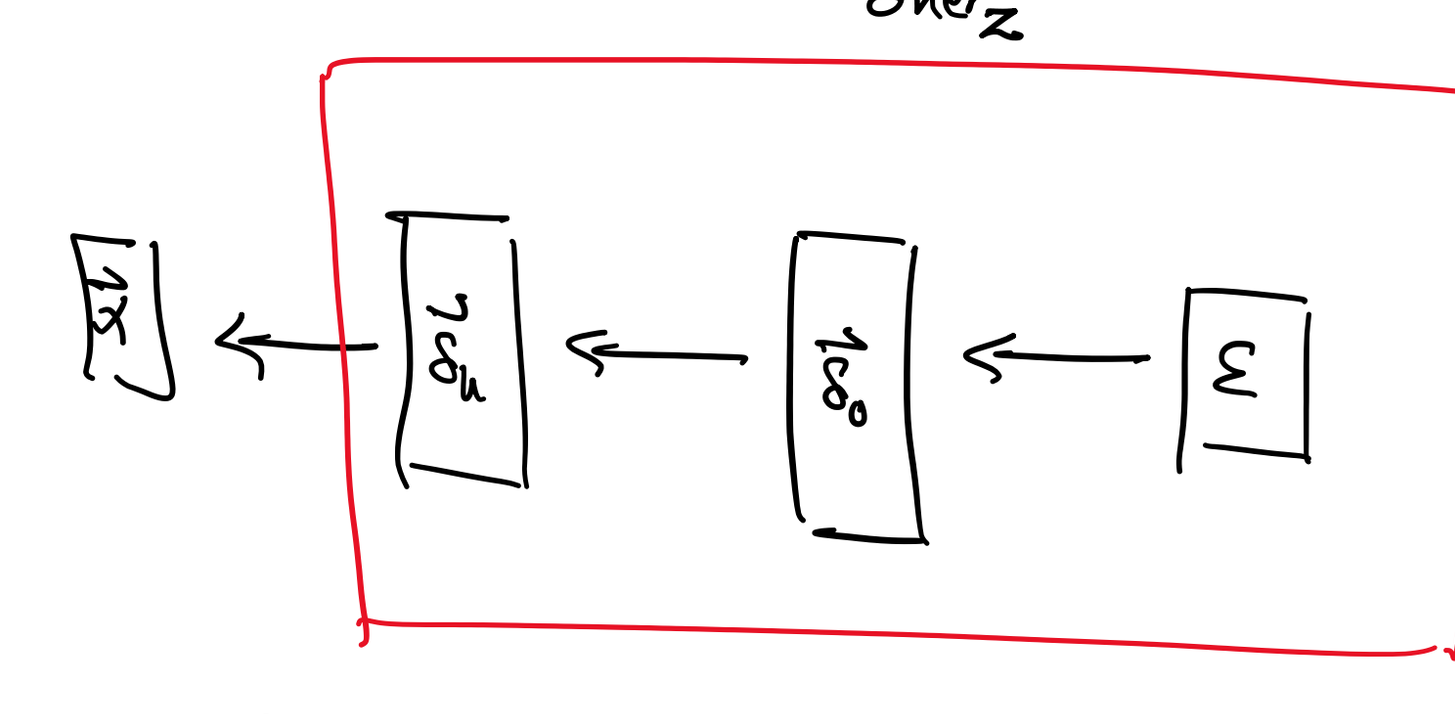
$$W_h = W_h - \eta \cdot \nabla_{W_h} \qquad W_o = W_o - \eta \cdot \nabla_{W_o}$$
$$b_h = b_h - \eta \cdot \nabla_{b_h} \qquad b_o = b_o - \eta \cdot \nabla_{b_o}$$

$$\vec{o} = f^o(W_o^T \vec{z} + \vec{b}_o) = f^o(\vec{net}_o)$$

$$\mathcal{E} = \frac{1}{2}\|\vec{y} - \vec{o}\|^2$$

$\vec{\delta}_o \equiv$ net gradient vector for output $= \frac{\partial \mathcal{E}}{\partial \vec{net}_o}$

$\vec{\delta}_h \equiv$ net gradient for hidden $= \frac{\partial \mathcal{E}}{\partial \vec{net}_z}$



$$\mathcal{E} = \frac{1}{2}\|\vec{y} - \vec{o}\|^2 = \frac{1}{2}\|\vec{y} - f^o(\vec{net}_o)\|^2$$

$\vec{o} = f^o(\vec{net}_o)$

$$\frac{\partial \mathcal{E}}{\partial \vec{net}_o} = \frac{\partial \mathcal{E}}{\partial \vec{o}} \times \frac{\partial \vec{o}}{\partial \vec{net}_o}$$

$$= \frac{1}{2}\cdot 2(\vec{y} - \vec{o}) \times -1 = (\vec{o} - \vec{y})$$

$\frac{\partial f(\vec{net}_o)}{\partial net_o}$ — e.g. f^o is sigmoid: $f^o(net_o)\cdot(1 - f^o(\vec{net}_o))$ = $\vec{o} \odot (1 - \vec{o})$

$$\vec{\delta}_o = \frac{\partial \mathcal{E}}{\partial \vec{net}_o} = (\vec{o} - \vec{y}) \odot \vec{o} \odot (1 - \vec{o})$$

element-wise multiplication

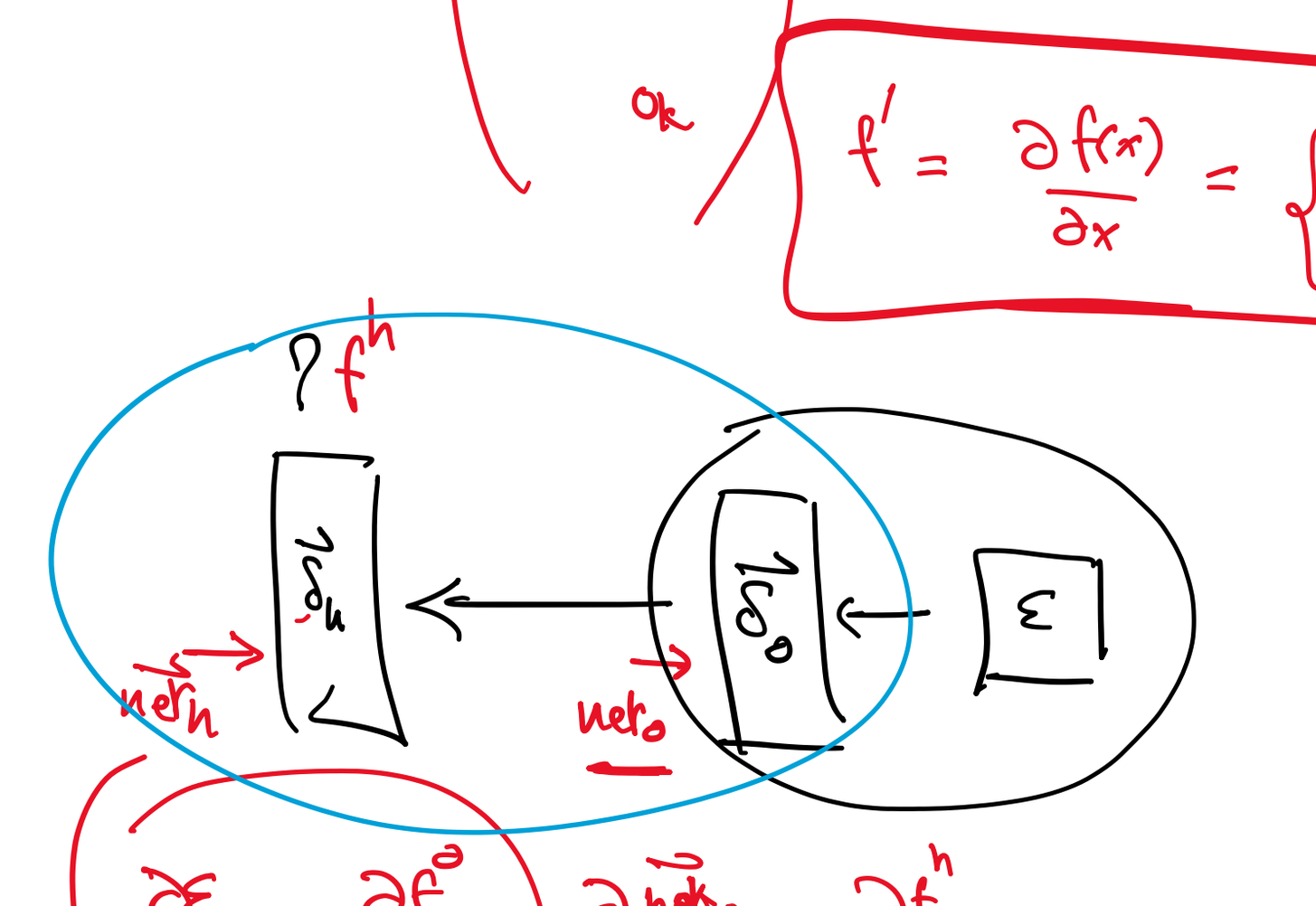
$$\left(\frac{\partial \mathcal{E}}{\partial o} \odot \frac{\partial f^o}{\partial net_o}\right)$$

e.g. f^o is tanh: $\frac{\partial f(\vec{net}_o)}{\partial \vec{net}_o} = 1 - \vec{net}_o^{\,2}$ (element wise)

$\frac{f(\vec{net}_o)}{\partial \vec{net}_o} = \begin{pmatrix} 1/0 \text{ whether } net_{o_1} > 0 \\ \vdots \\ \text{ok} \end{pmatrix}$

f^o = relu

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{otherwise} \end{cases}$$
$$f' = \frac{\partial f(x)}{\partial x} = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{otherwise} \end{cases}$$



$$\frac{\partial \mathcal{E}}{\partial net_z} = \frac{\partial \mathcal{E}}{\partial o} \cdot \frac{\partial f^o}{\partial net_o} \cdot \frac{\partial net_o}{\partial f^h} \cdot \frac{\partial f^h}{\partial net_z}$$

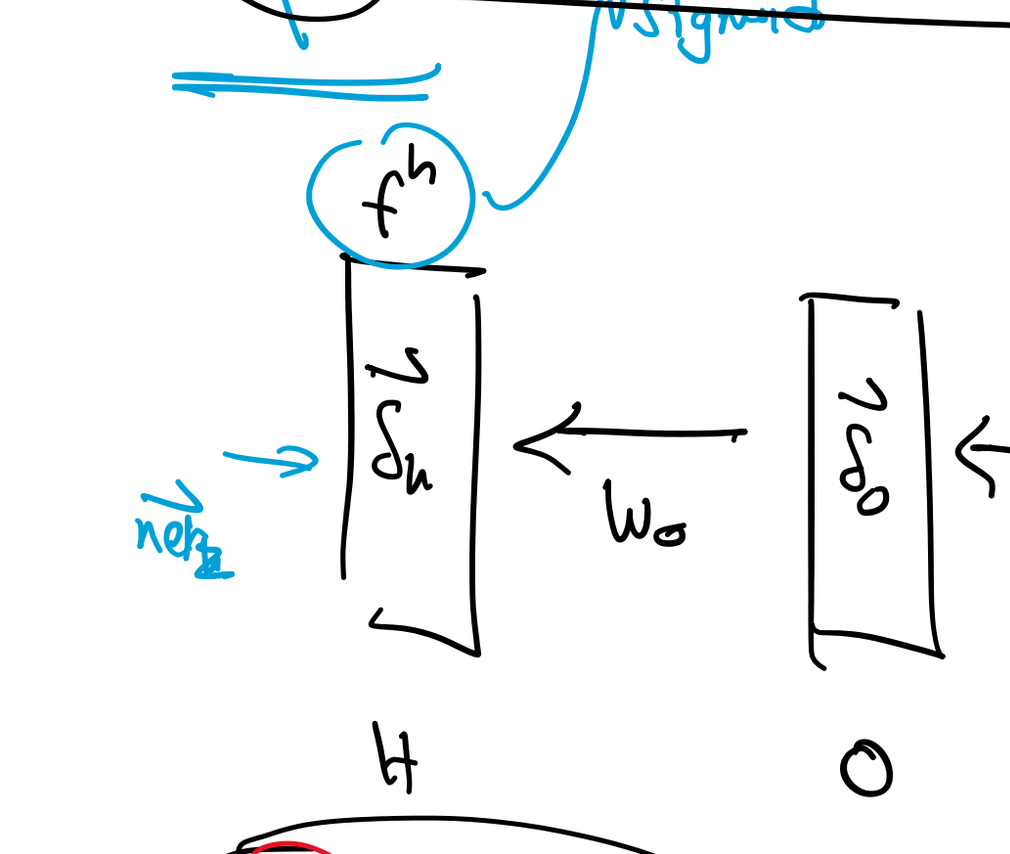
$$\frac{\partial \mathcal{E}}{\partial net_z} = \vec{\delta}^o \frac{\partial \vec{net}_o}{\partial \vec{z}} \cdot \frac{\partial \vec{z}}{\partial \vec{net}_z}$$

$$\vec{\delta}_h = (W_o \cdot \vec{\delta}_o) \odot \vec{z} \odot (1 - \vec{z})$$

(sigmoid f^h)

$\vec{net}_o = W_o^T \vec{z} + \vec{b}_o$

$\vec{z} = f^h(\vec{net}_h)$



$$\nabla_{W_o} = \frac{\partial \mathcal{E}}{\partial W_o} = \frac{\partial \mathcal{E}}{\partial net_o} \cdot \frac{\partial net_o}{\partial W_o}$$

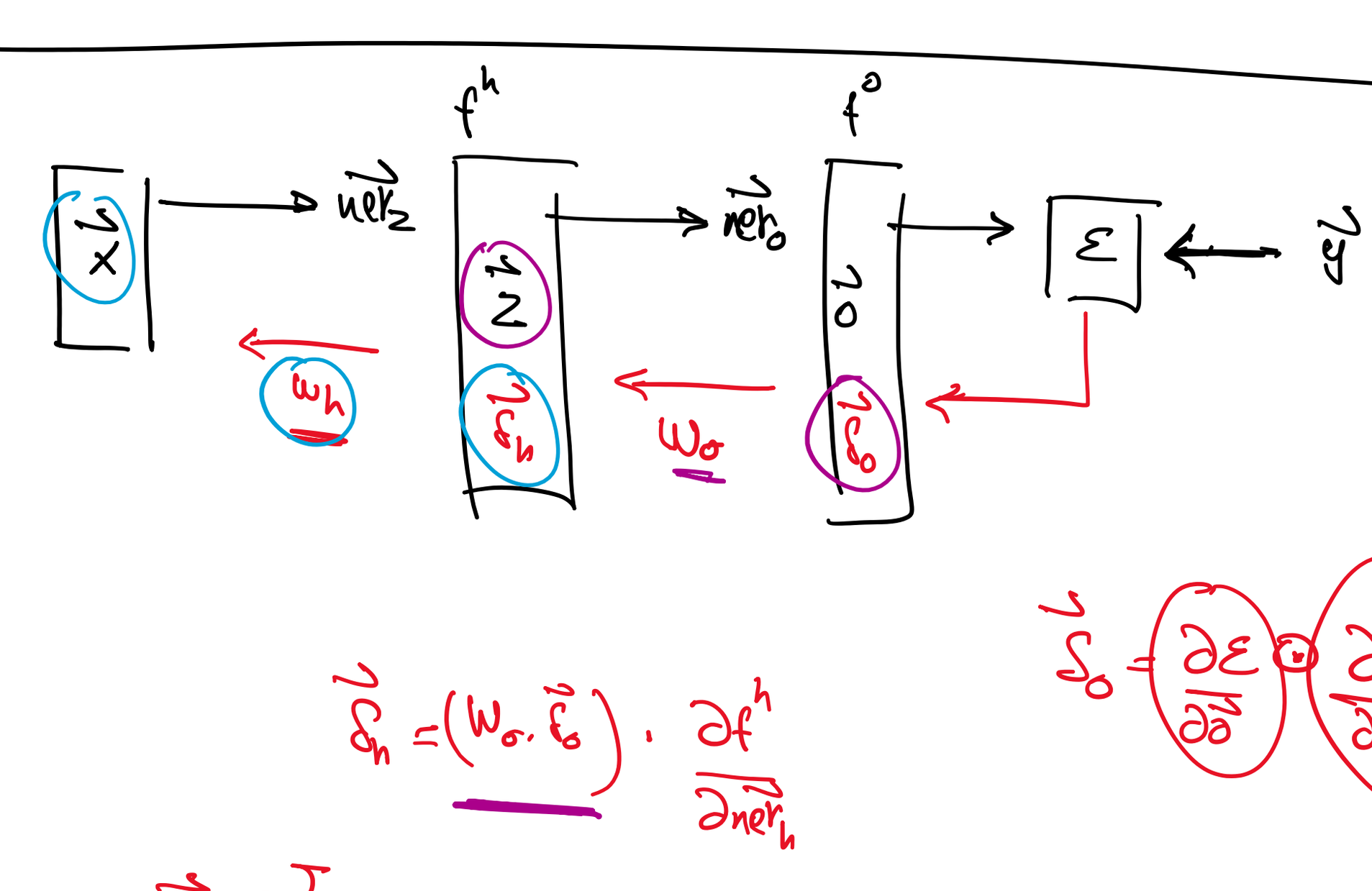
$$\nabla_{W_o} = \underbrace{\vec{z} \cdot \vec{\delta}_o^T}_{\text{outer product}}$$

$\vec{net}_o = W_o^T \vec{z} + \vec{b}_o$

$\frac{\partial \vec{net}_o}{\partial W_o} = \vec{z}$

$\frac{\partial \vec{net}_o}{\partial b_o} = \vec{1}$

$$\nabla_{b_o} = \frac{\partial \mathcal{E}}{\partial net_o} \cdot \frac{\partial net_o}{\partial b_o}$$
$$\nabla_{b_o} = \vec{\delta}_o$$
$$\nabla_{W_h} = \vec{x} \cdot \vec{\delta}_h^T$$
$$\nabla_{b_h} = \vec{\delta}_h$$



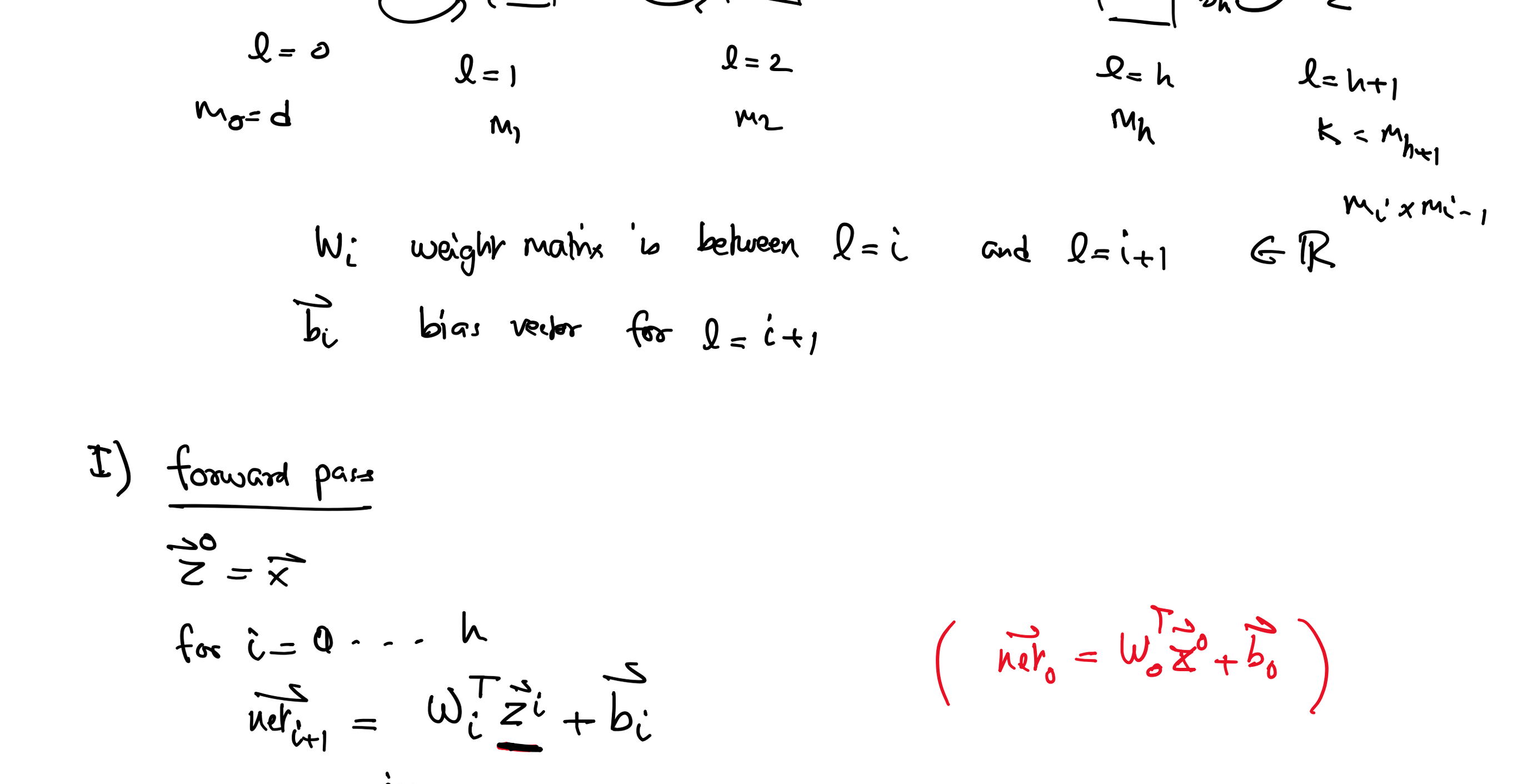
$$\vec{\delta}_o = \frac{\partial \mathcal{E}}{\partial \vec{o}} \odot \frac{\partial f^o}{\partial net_o}$$

$$\vec{\delta}_h = (W_o \cdot \vec{\delta}_o) \cdot \frac{\partial f^h}{\partial net_h}$$

$$\nabla W_o = \vec{z} \cdot \vec{\delta}_o^T \qquad \nabla_{b_o} = \vec{\delta}_o$$
$$\nabla W_h = \vec{x} \cdot \vec{\delta}_h^T \qquad \nabla_{b_h} = \vec{\delta}_h$$

deep MLP

h: # of hidden layers (info bottleneck)



W_i: weight matrix is between $l = i$ and $l = i+1$ $\in \mathbb{R}^{m_i \times m_{i+1}}$

b_i: bias vector for $l = i+1$

I) forward pass

$\vec{z}^0 = \vec{x}$

for $i = 0 \dots h$

$$\vec{net}_{i+1} = W_i^T \vec{z}^i + \vec{b}_i$$
$$\vec{z}^{i+1} = f^{i+1}(\vec{net}_{i+1})$$

$(\vec{net}_o = W_o^T \vec{z}^o + \vec{b}_o)$

II) Error

$$\vec{o} = \vec{z}^{h+1} \in \mathbb{R}^k$$

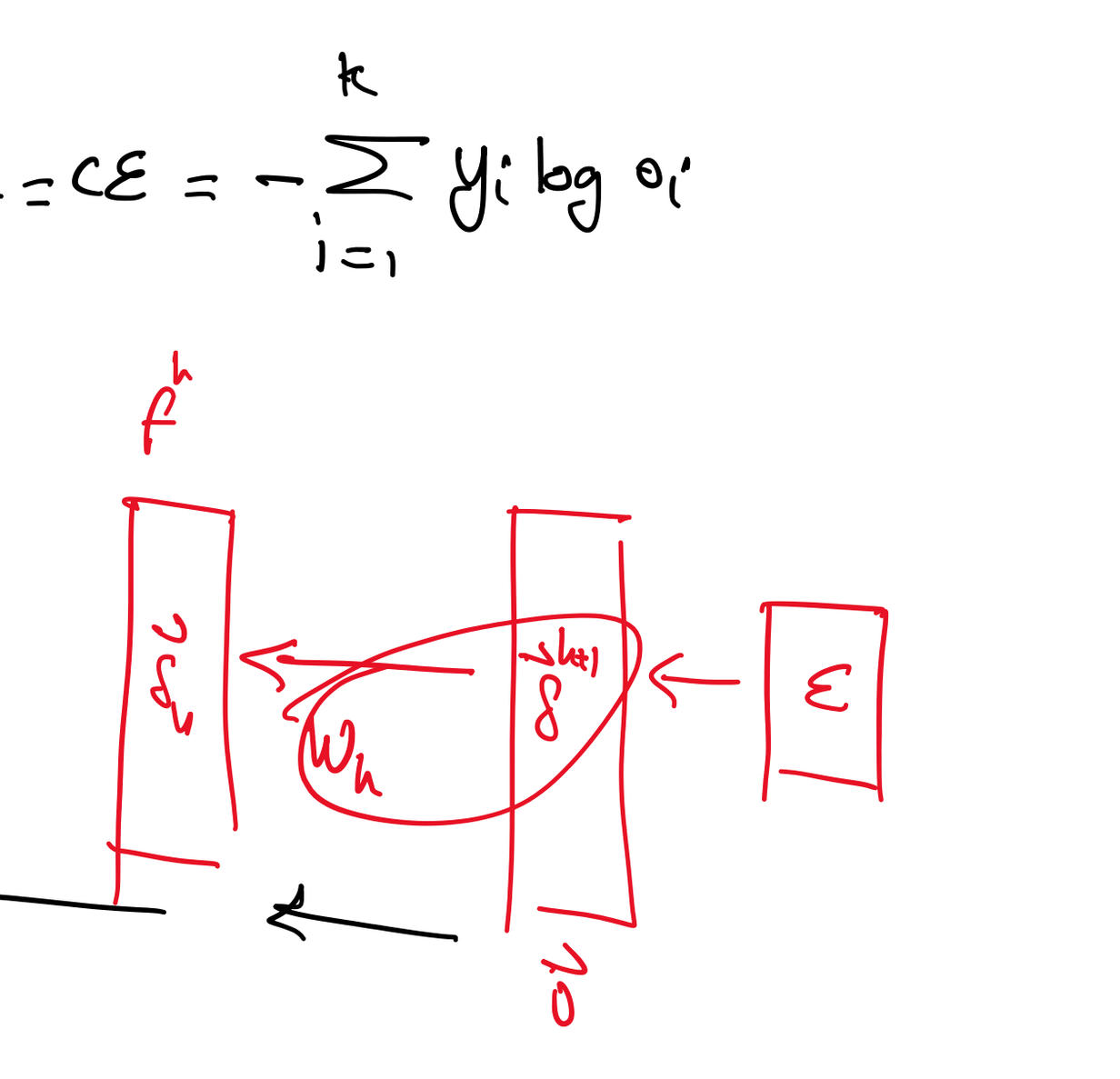
$$\mathcal{E} = \frac{1}{2}\|\vec{y} - \vec{o}\|^2 \quad \text{or} \quad \mathcal{E} = CE = -\sum_{i=1}^{k} y_i \log o_i$$

III) backprop

$$\delta^{h+1} = \frac{\partial \mathcal{E}}{\partial o} \odot \frac{\partial f^o}{\partial net_o}$$

$$\delta^h = W_h \cdot \delta^{h+1} \odot \frac{\partial f^h}{\partial net_h}$$

$\vdots$



for $l = h, h-1, \dots, 1$ (reverse)

$$\vec{\delta}^l = \partial f^l \odot (W_l \cdot \vec{\delta}^{l+1})$$

$$\partial f^l = f'^l = \frac{\partial f^l(\vec{net}_l)}{\partial \vec{net}_l}$$

IV gradient for parameters

for $l = 0, 1, \dots, h$

$$\nabla_{W_l} = z^l \cdot (\vec{\delta}^{l+1})^T \quad \text{outer product of two vectors}$$
$$\nabla_{b_l} = \vec{\delta}^{l+1}$$

V gradient descent

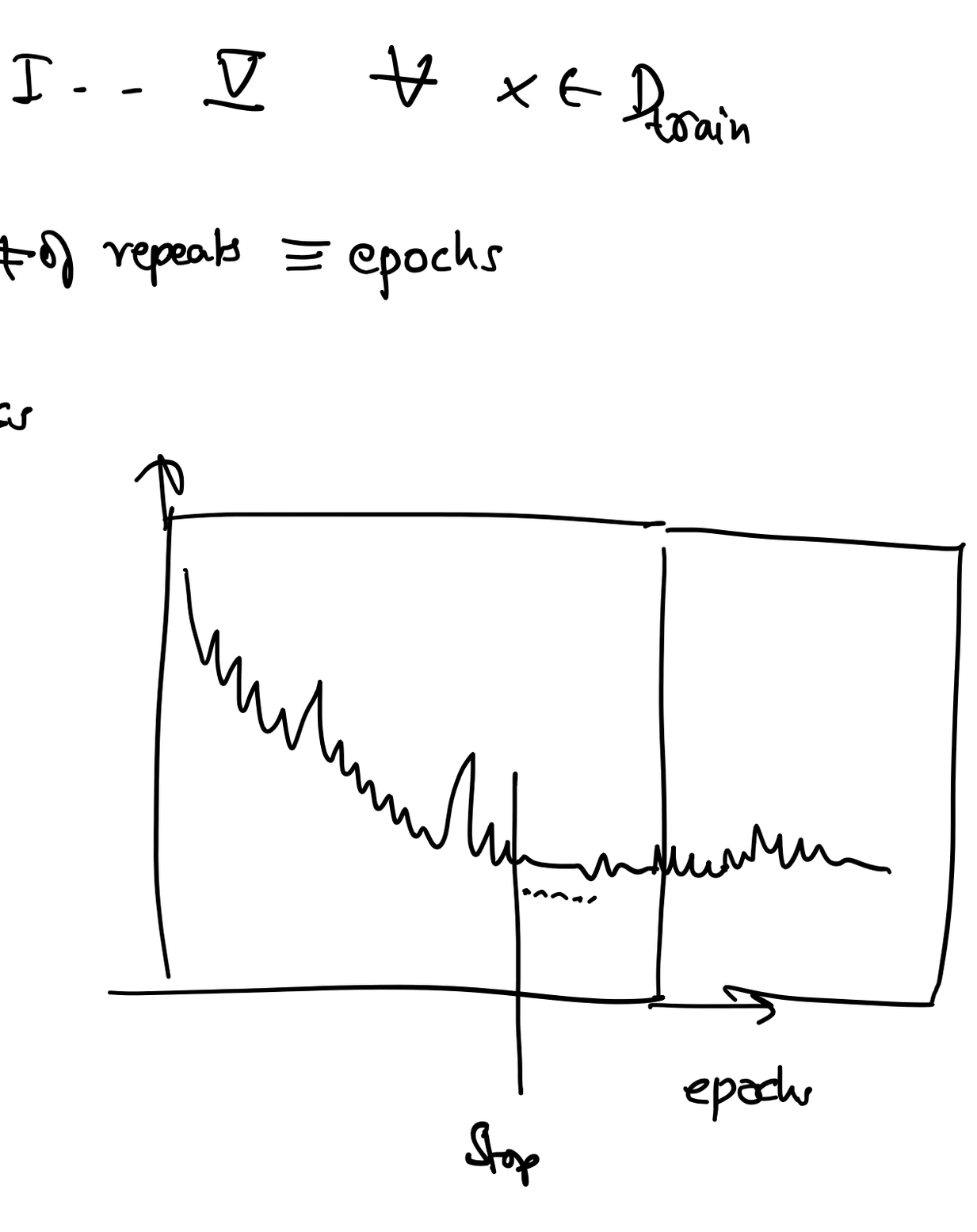
for $l = 0, 1, \dots, h$

$$W_l = W_l - \eta \cdot \nabla_{W_l}$$
$$b_l = b_l - \eta \cdot \nabla_{b_l}$$

repeat I - - V $\forall\, x \in D_{train}$

\# of repeats $\equiv$ epochs

for $e = 1 \dots$ epochs
- I
- II
- III
- IV
- V



Stopping criteria

1) fixed # of epochs
2) monitor the loss

derivatives

sigmoid $\quad f(z) = \frac{1}{1 + e^{-z}} \qquad f' = \vec{z} \odot (\vec{1} - \vec{z}) = \vec{net} \odot (1 - \vec{net})$

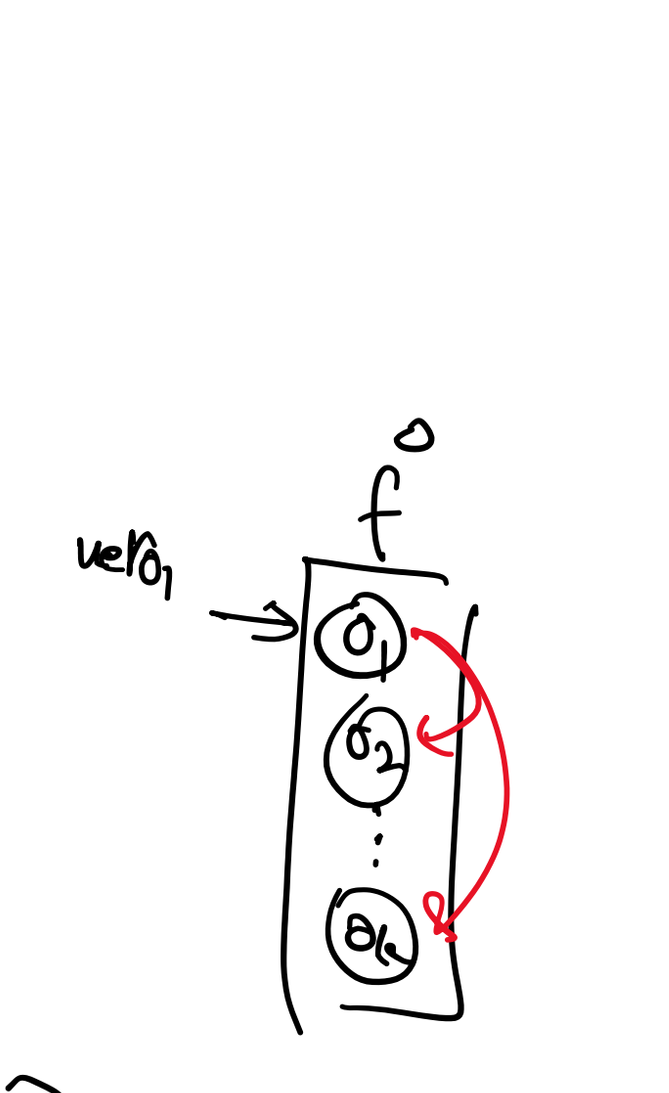
tanh $\quad f(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \qquad f' = \vec{1} - \vec{z}^2 = \vec{1} - \vec{net} \odot \vec{net}$

relu $\quad f(z) = \begin{cases} z & \text{if } z > 0 \\ 0 & \text{else} \end{cases} \qquad f' = \begin{cases} 1 & \text{if } z > 0 \\ 0 & \text{else} \end{cases}$

Softmax $(net_i \mid net_1, net_2, \dots, net_k) = \frac{e^{net_i}}{\sum_{j=1}^{k} e^{net_j}}$

$$\frac{\partial \mathcal{E}}{\partial o} \cdot \frac{\partial f^o}{\partial net_i}$$

→ for softmax



$$\begin{pmatrix} \frac{\partial o_1}{\partial net_1} & \frac{\partial o_2}{\partial net_2} & \cdots & \frac{\partial o_1}{\partial net_k} \\ \frac{\partial o_2}{\partial net_1} & & & \\ \vdots & & & \\ \frac{\partial o_k}{\partial net_1} & \cdots & & \frac{\partial o_k}{\partial net_k} \end{pmatrix}$$