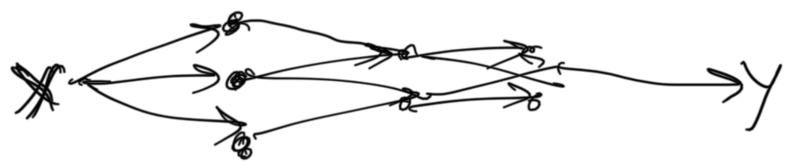
$$y = f(x)$$

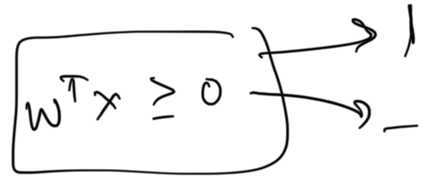
Neural Networks.

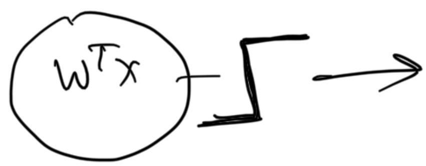


$W^T X$

Thresholded perceptron

$$w_1 x_1 + w_2 x_2 + w_3 x_3 + w_4 x_4 + w_5 x_5$$



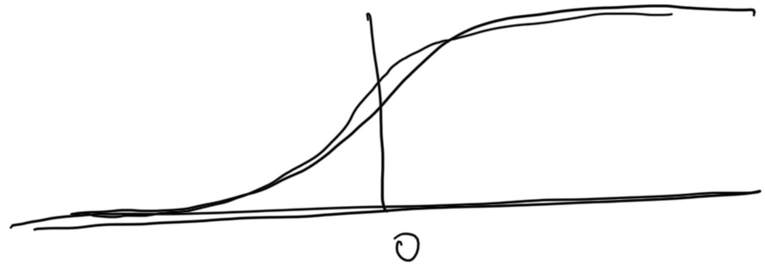


Unit

Layer

Sigmoid unit

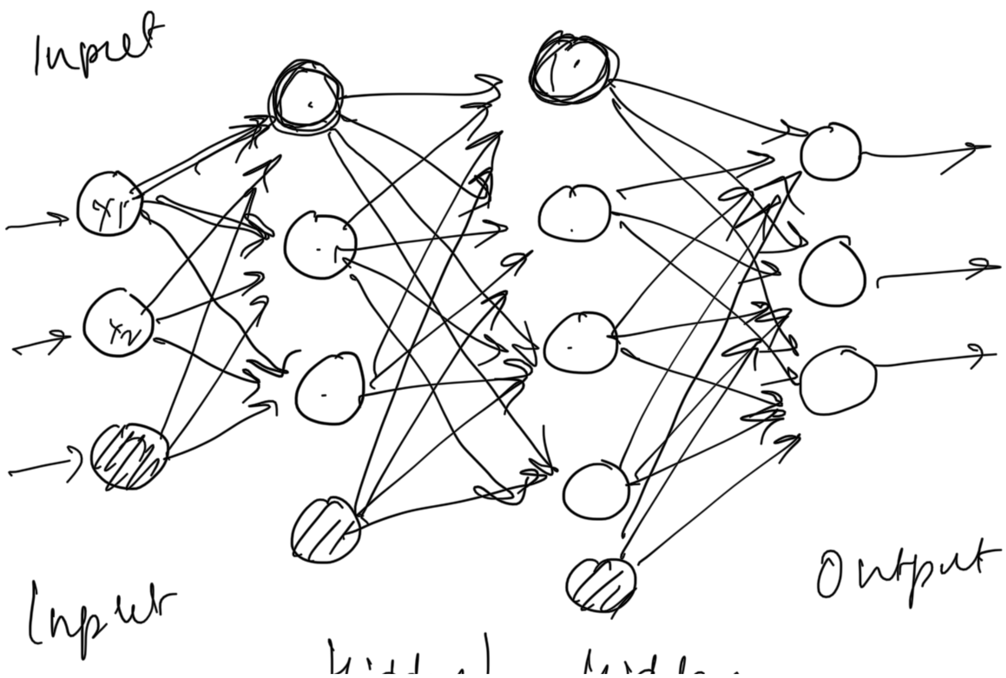
$$\boxed{\sigma(z) = \frac{1}{1+\exp(-z)}}$$



March 12 Friday

- PA2 is out
- Mid-term next Friday

D=2

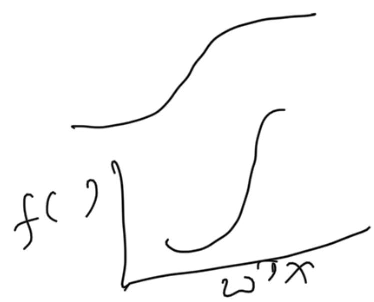


hidden1 hidden2

# of units in a layer $\equiv$ width of a layer

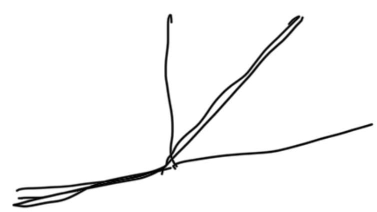
Sigmoid

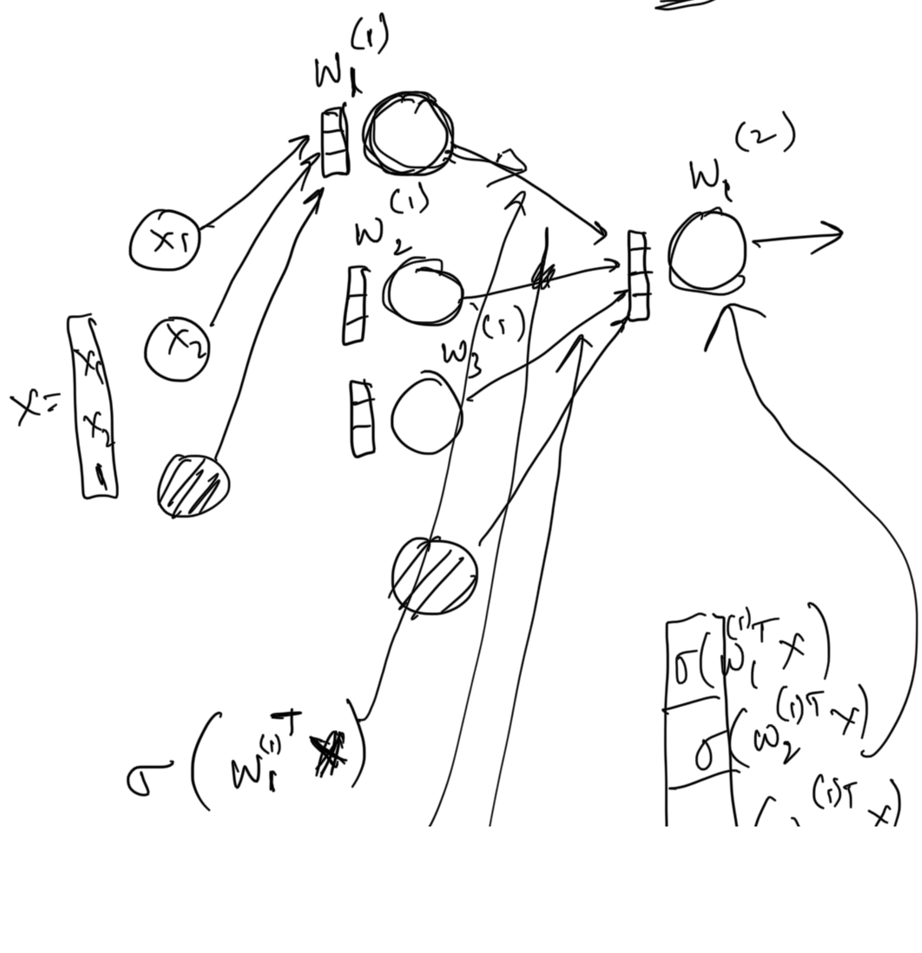
tanh



Relu

$\max(0, w^T x)$



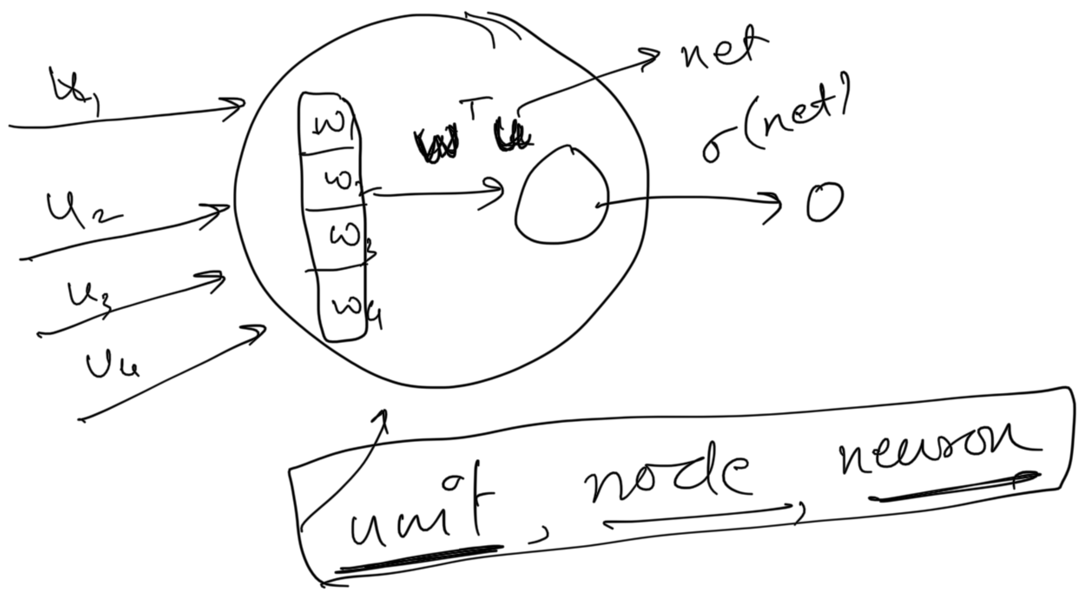


$$\sigma(w_2^{(1)T} x)$$

$$\sigma(w_3^{(1)T} x)$$

$\sigma(w_3 \ldots$

1



Sigmoid

$$sigmoid(z) = \frac{1}{1+e^{-z}}$$

Tanh

$$tanh(z) = \frac{e^{z}-e^{-z}}{e^{z}+e^{-z}}$$

Simple example

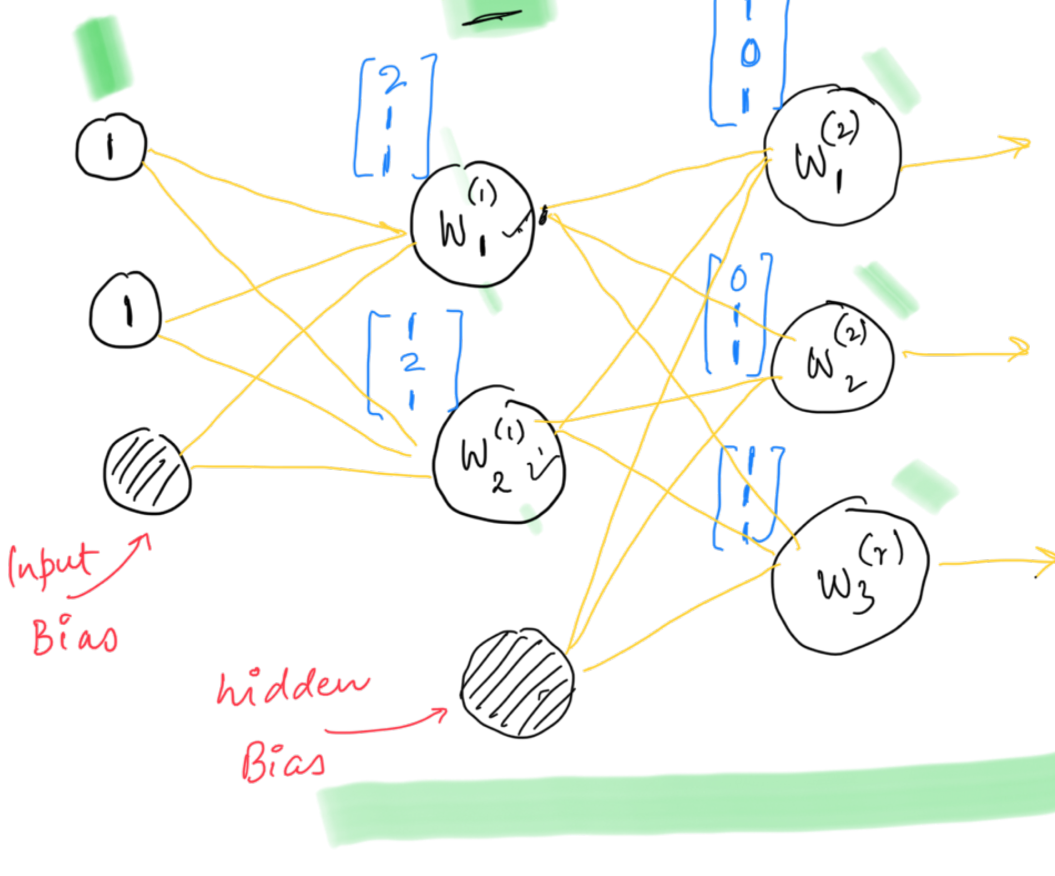
$D = 2$ (2-d data)

$M = 2$

$K = 3$ (3-outputs / classes)

Assume sigmoid activation

Data: $\mathbf{x} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$



At hidden unit 1

$$net_1^{(1)} = w_1^{(1)T} \mathbf{x}$$

$$= [2 \quad 1 \quad 1] \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = 4$$

$$z_1 = \sigma(net_1^{(1)}) = \sigma(4)$$

$$= 0.98$$

<u>At hidden unit 2</u>

$$net_2^{(1)} = w_2^{(1)} X$$

$$= [1 \quad 2 \quad 1] \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = 4$$

$$z_2 = \sigma(net_2^{(1)}) = \sigma(4)$$

$$= 0.98$$

$$\mathbf{Z} = \begin{bmatrix} z_1 \\ z_2 \\ 1 \end{bmatrix}$$

<u>At output unit 1</u>

$$net_1^{(2)} = w_1^{(2)T} \mathbf{Z}$$

$$= [1 \quad 0 \quad 1] \begin{bmatrix} 0.98 \\ 0.98 \\ 1 \end{bmatrix} = 1.98$$

$$o_1 = \sigma(1.98) = \frac{1}{1 + \exp(-1.98)}$$

$$= 0.88$$

<u>At output unit 2</u>

$$net^{(2)} \quad w^{(2)T} \quad Z$$

$$net_2 = w_2 \; z$$

$$= \begin{bmatrix} 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0.98 \\ 0.98 \\ 1 \end{bmatrix} = 1.98$$

$$O_2 = \sigma(1.98) = 0.88$$

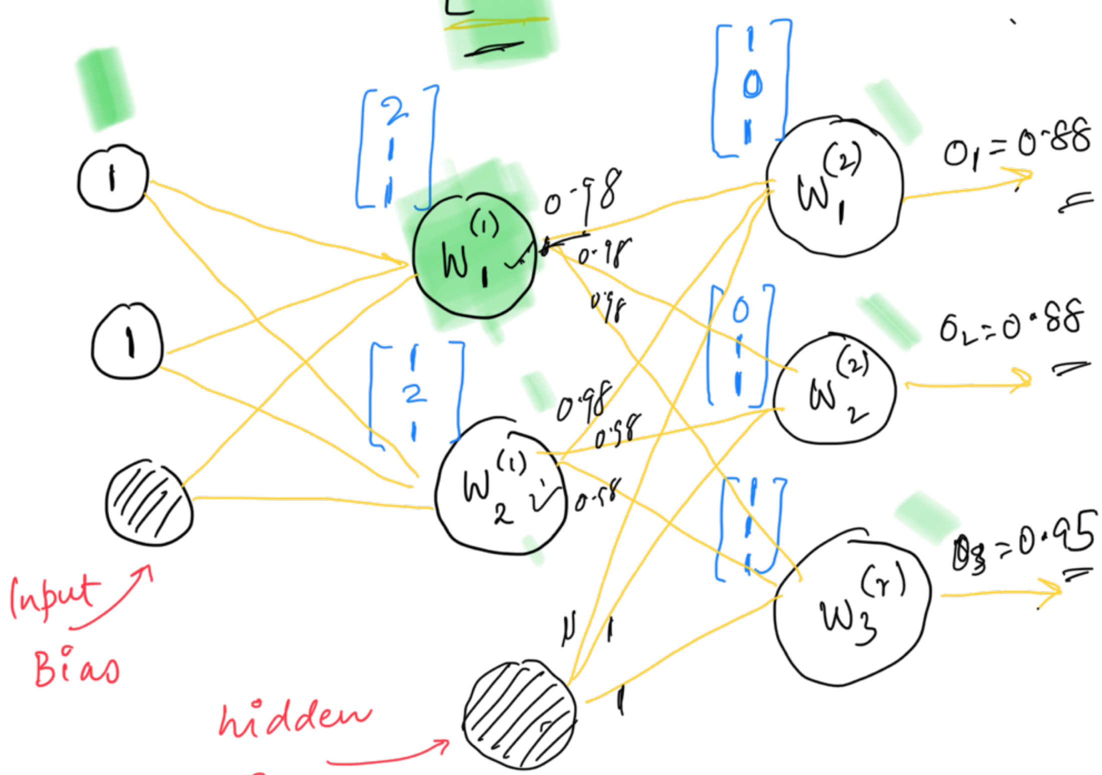
<u>At output unit 3</u>

$$net_3^{(2)} = W_3^{(2)T} z = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0.98 \\ 0.98 \\ 1 \end{bmatrix}$$

$$= 2.96$$

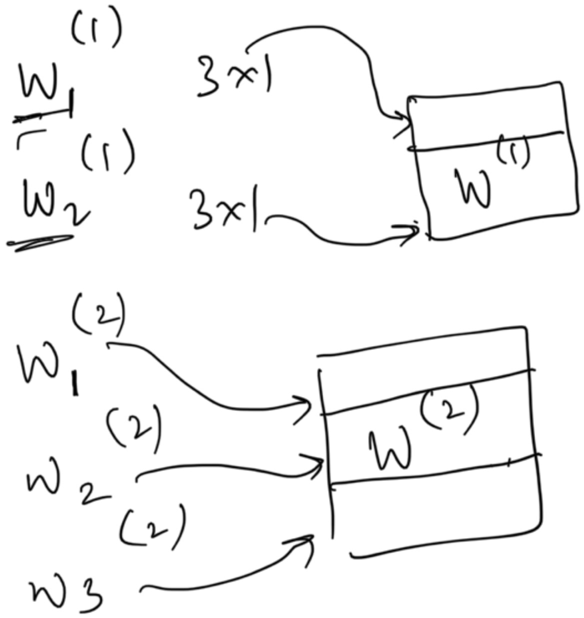
$$O_3 = \sigma(2.96) = 0.95$$

Data: $\mathbf{x} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$



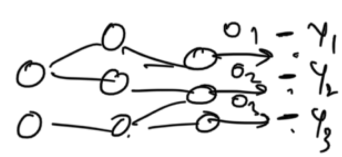
Bias

softmax output : $\boxed{\frac{O_1}{O_1+O_2+O_3}, \frac{O_2}{O_1+O_2+O_3}, \frac{O_3}{O_1+O_3+O_3}}$



$$\sigma\Big(\underbrace{W^{(1)}}_{2\times 3}\ \underbrace{x}_{3\times 1}\Big) = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \qquad z = \begin{bmatrix} z_1 \\ z_2 \\ 1 \end{bmatrix}$$

$$\sigma\left(\underline{W^{(2)} z}\right) = \begin{bmatrix} o_1 \\ o_2 \\ o_3 \end{bmatrix}$$



$$J(w^{(1)}, \ldots, w^{(2)}, \ldots)$$

$$= \sum^{N} J_i$$

$$\boxed{J_i = \frac{1}{2}\sum_{l=1}^{K}(y_{il} - O_{il})^2}$$

$O_{il} \rightarrow$ output for the i^{th} training example at output unit l

$y_{il} \rightarrow$ true output for the i^{th} example at l^{th} output unit.

$$\boxed{J = \frac{1}{2}\sum_{i=1}^{N}\sum_{l=1}^{K}(y_{il} - O_{il})^2}$$

E.g. 3 class classifier.

x $D=2$	y
3.7, 4.8	2
3.4, 1.2	1

One-of-K encoding

Dummy encoding

Let $K=3$

$2 \longrightarrow$ [0 | 1 | 0]

$1 \longrightarrow$ [1 | 0 | 0]

$3 \longrightarrow$ [0 | 0 | 1]

Gradient Descent

$J \to$ is a function of all the weights.

∂J

Wednesday March 17

Notation

Subscripts

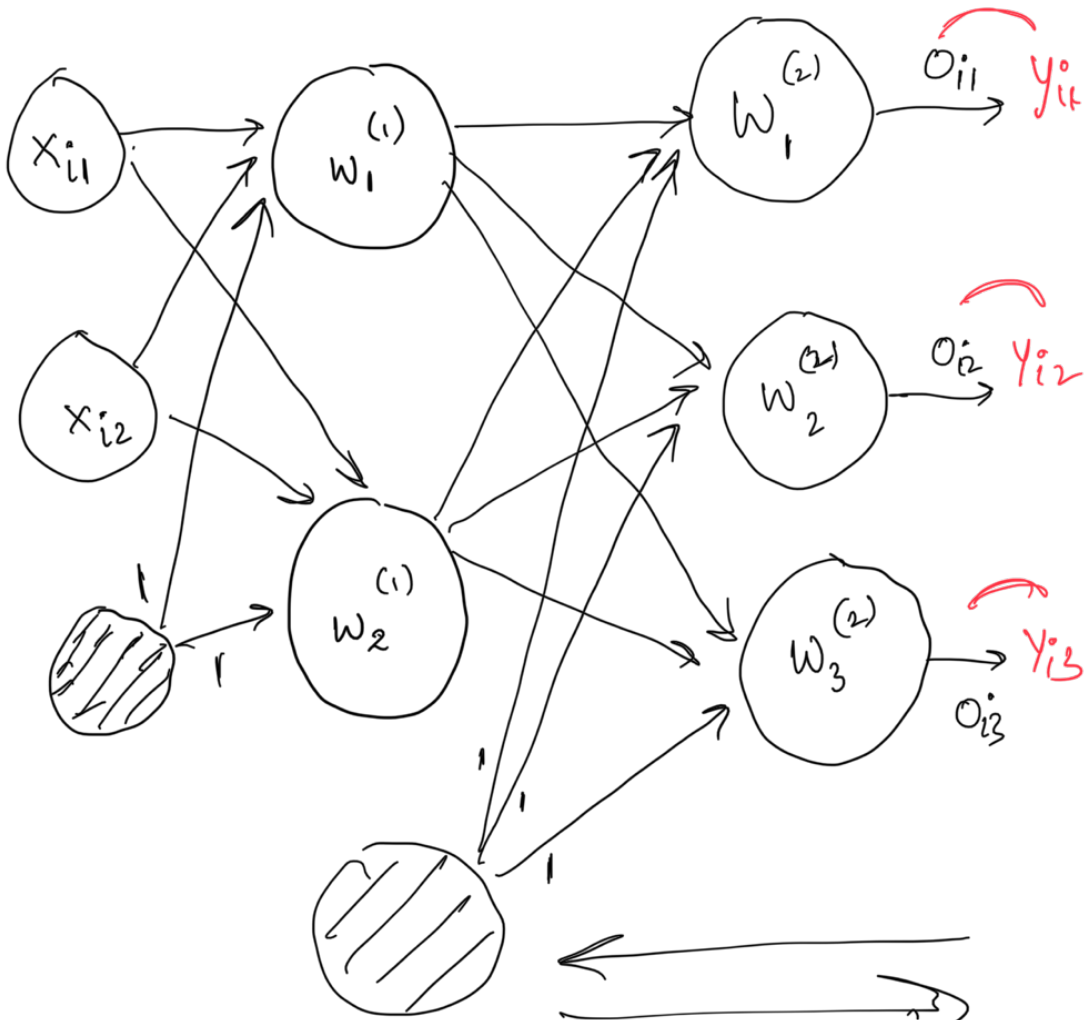
i	Training example	x_i
p	Feature	x_{ip}
j	Hidden layer unit	$w_j^{(1)}$
l	Output layer unit	$w_l^{(2)}$

Variables

D+1	x_i	Input vector for i^{th} training example
D+1	$w_j^{(1)}$	weight vector at j^{th} hidden unit
M+1	$w_l^{(2)}$	weight vector at l^{th} output unit
	z_j	Output of the j^{th} hidden unit
	o_l	output the l^{th} output unit
K	y_i	1-of-K true output for i^{th} training example

i^{th} weight vector

$\mathbf{W}_r$	a general unit weight vector
$\mathbf{u}_r$	input to a general unit
V_r	output of a general unit
net_r	intermediate output: $W_r^T \mathbf{u}_r$



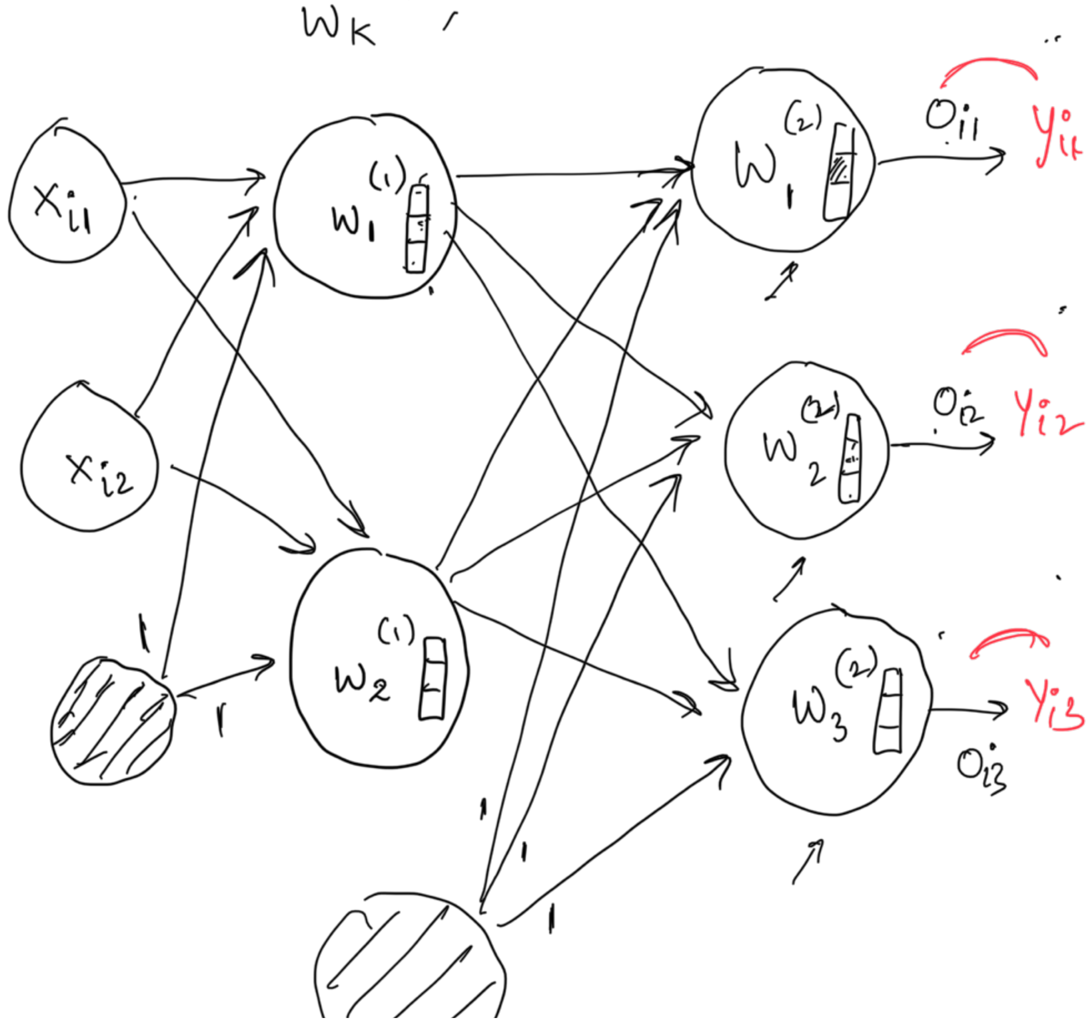
Training data

X_1	Y_1	1	[1 0 0]
X_2	Y_2	1	[1 0 0]
X_3	Y_3	2	[0 1 0]

x_i $\quad$ y_i $\quad 3$ $\quad [0 \;\; 0 \;\; 1]$

x_N $\quad$ y_N $\quad 1$

$$J\left(w_1^{(1)}, w_2^{(1)}, \ldots, w_M^{(1)}, w_1^{(2)}, w_2^{(2)}, \ldots, w_K^{(2)}\right) \qquad \nabla$$



$J =$

$$\boxed{\frac{\partial J}{\partial w_{jp}^{(1)}} \qquad \frac{\partial J}{\partial w_{\ell j}^{(2)}}}$$

scalar

$$w_{jp}^{(1)} \leftarrow w_{jp}^{(1)} - \eta \frac{\partial J}{\partial w_{jp}^{(1)}}$$

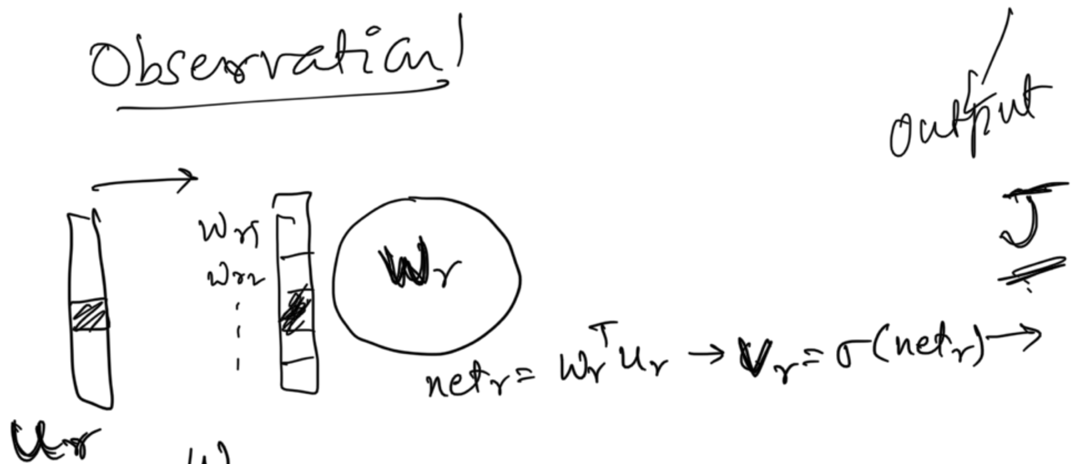
$$w_{\ell j}^{(2)} \leftarrow w_{\ell j}^{(2)} - \eta \frac{\partial J}{\partial w_{\ell j}^{(2)}}$$

Drop i ⇒ Assume we have only 1 training example.

$$J = \frac{1}{2} \sum_{i=1}^{N} \sum_{l=1}^{K} (y_{il} - O_{il})^2$$

$$\boxed{J = \frac{1}{2} \sum_{l=1}^{K} (y_l - O_l)^2}$$

Observation!



w_{rq}

$$net_r = w_r^T u_r$$

$$= \sum_q w_{rq} u_{rq}$$

$$\frac{\partial J}{\partial w_{rq}} = \frac{\partial J}{\partial net_r} \frac{\partial net_r}{\partial w_{rq}}$$

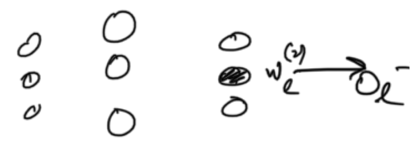
chain rule

$$\boxed{\frac{\partial net_r}{\partial w_{rq}} = u_{rq}}$$

derivations

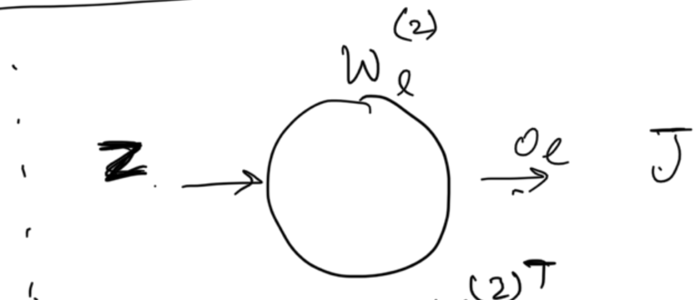
$$\boxed{\frac{\partial J}{\partial w_{rq}} = \frac{\partial J}{\partial net_r} u_{rq}}$$

Observation 2



For l^{th} output unit :

$$\boxed{\frac{\partial J}{\partial net_l} = \frac{\partial J}{\partial O_l} \frac{\partial O_l}{\partial net_l}}$$



$$net_l = w_l^{(2)T} \mathbf{z}$$

$$O_l = \sigma(net_l)$$

$$J = \frac{1}{2} \sum_{l=1}^{K} (y_l - O_l)^2$$

What is $\frac{\partial J}{\partial O_l}$?

$$J = \frac{1}{2}\Big[(y_1 - O_1) + (y_2 - O_2) + (y_3 - O_3) + \cdots + (y_\ell - O_\ell)^2 + \cdots\Big]$$

$$\boxed{\begin{aligned}\frac{\partial J}{\partial O_\ell} &= \frac{1}{\cancel{2}}\,\cancel{2}\,(y_\ell - O_\ell)\,(-1)\\ &= -(y_\ell - O_\ell)\end{aligned}}$$

#1

What is $\dfrac{\partial O_\ell}{\partial net_\ell}$

$$O_\ell = \sigma(net_\ell) = \frac{1}{1+\exp(-net_\ell)}$$

$$\frac{\partial O_\ell}{\partial net_\ell} = \frac{\partial}{\partial net_\ell}\left[\frac{1}{1+e^{-net_\ell}}\right]$$

$$= -\frac{1}{(1+e^{-net_\ell})^2}\left(-e^{-net_\ell}\right)$$

$$\boxed{\frac{\partial O_\ell}{\partial net_\ell} = \frac{e^{-net_\ell}}{(1+e^{-net_\ell})^2} = \underline{O_\ell(1-O_\ell)}}$$

#2

Combining #1 & #2:

$$\frac{\partial J}{\;} = -O_\ell(1-O_\ell)(y_\ell - O_\ell)$$

∂net_ℓ

Let $\boxed{\delta_\ell = O_\ell(1-O_\ell)(y_\ell - O_\ell)}$

$$\frac{\partial J}{\partial w_{\ell j}^{(2)}} = \frac{\partial J}{\partial net_j} z_j$$

$$= -O_\ell(1-O_\ell)(y_\ell - O_\ell) z_j$$

$$= -\delta_\ell z_j$$

Update rule for output layer.

$$\boxed{w_{\ell j}^{(2)} \leftarrow \underset{\text{old}}{w_{\ell j}^{(2)}} + \eta\, \delta_\ell z_j}$$

Preview :

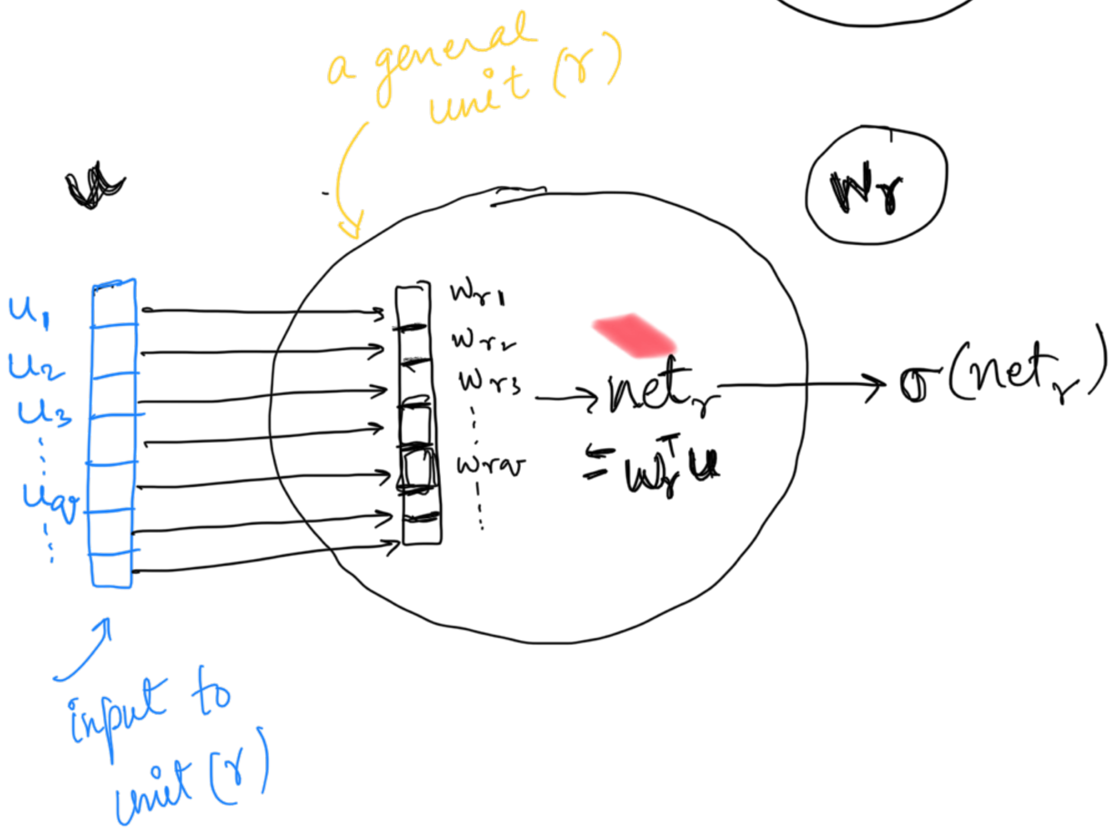
$$\boxed{w_{jp}^{(1)} \leftarrow \underset{\text{old}}{w_{jp}^{(1)}} + \eta\, \delta_j x_u}$$

δ_j = function of δ_ℓs.

Monday Mar 22

What do we need?

$$\frac{\partial J}{\partial w_{rq}}$$



Observation 1

$$\boxed{\frac{\partial J}{\partial w_{rq}} = \frac{\partial J}{\partial net_r}\,\frac{\partial net_r}{\partial w_{rq}} = \frac{\partial J}{\partial net_r}\,u_q}$$

Observation 2

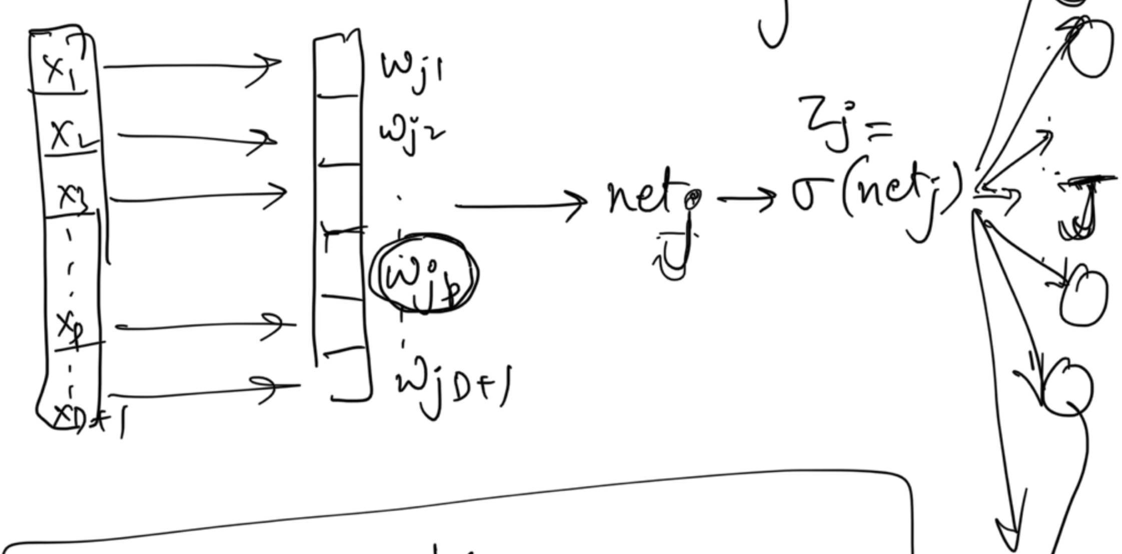
For weight on an output unit:

$$\partial J \quad = -\delta_\ell z_j$$

$\partial w_{\ell j}$

$$\boxed{\delta_\ell = (y_\ell - O_\ell)\, O_\ell (1 - O_\ell)}$$

For a hidden unit w_j



$$\boxed{\frac{\partial J}{\partial net_j} = \sum_{\ell=1}^{K} \frac{\partial J}{\partial net_\ell} \frac{\partial net_\ell}{\partial net_j}}$$

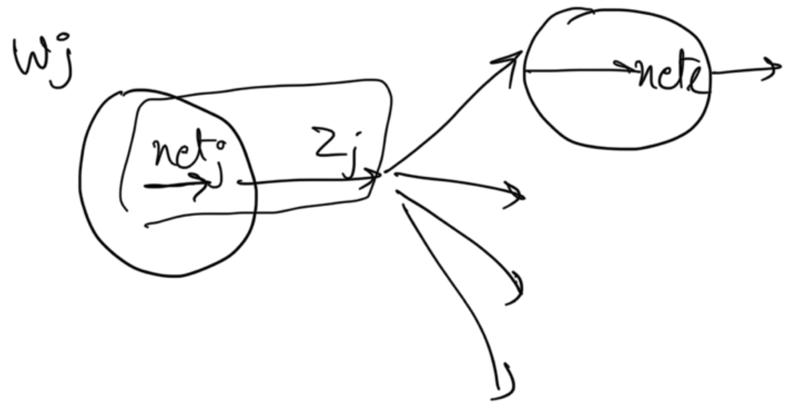
We already know: $\dfrac{\partial J}{\partial net_\ell} = -\delta_\ell$

Check ~~week~~ wed. derivation

$$\frac{\partial J}{\partial net_j} = -\sum^{K} \delta_\ell \frac{\partial net_\ell}{\partial net_j}$$

$\partial net_j \qquad l=1 \qquad \partial net_j$

$$= -\sum_{l=}^{K}\left[\delta_l\left(\frac{\partial net_l}{\partial z_j}\right)\left(\frac{\partial z_j}{\partial net_j}\right)\right]$$



$z_j = \sigma(net_j)$

$$\Rightarrow \boxed{\frac{\partial z_j}{\partial net_j} = z_j(1-z_j)}$$

$$\frac{\partial J}{\partial net_j} = -\sum_{l=1}^{K}\left[\delta_l\left(\frac{\partial net_l}{\partial z_j}\right)\right]z_j(1-z_j)$$

$$\frac{\partial net_l}{\partial z_j}$$

Recall that $net_l = \sum^{M+1} W_{lj} z_j$

$j=1$

$$\frac{\partial net_\ell}{\partial z_j} = w_{\ell j}$$

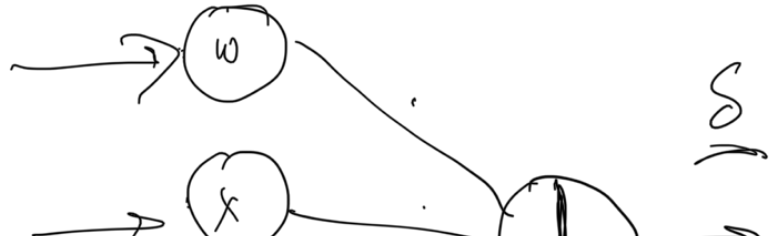
$$\Rightarrow \frac{\partial J}{\partial net_j} = -z_j(1-z_j)\left(\sum_{\ell=1}^{K} \delta_\ell w_{\ell j}\right)$$

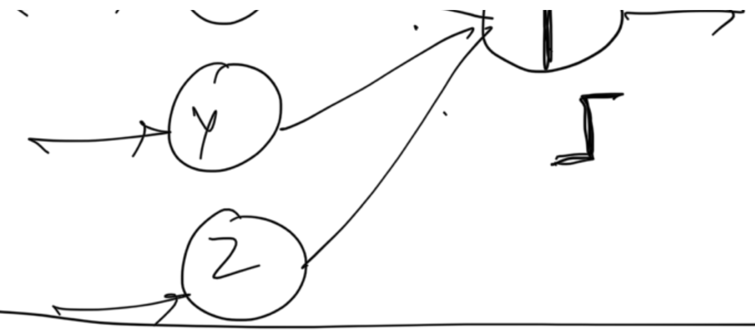
$$\frac{\partial J}{\partial w_{jp}} = -z_j(1-z_j)\left(\sum_{\ell=1}^{K} \delta_\ell w_{\ell j}\right) x_p$$

$$\delta_j = \cancel{\theta}\, z_j(1-z_j)\left(\sum_{\ell=1}^{K} \delta_\ell w_{\ell j}\right)$$

$$\boxed{\frac{\partial J}{\partial w_{jp}} = -\delta_j x_p}$$

$$\boxed{\begin{aligned} w_{jp} &= w_{jp} + \eta\, \underline{\delta_j}\, x_p \\ w_{\ell j} &= w_{\ell j} + \eta\, \underline{\delta_\ell}\, z_j \end{aligned}}$$

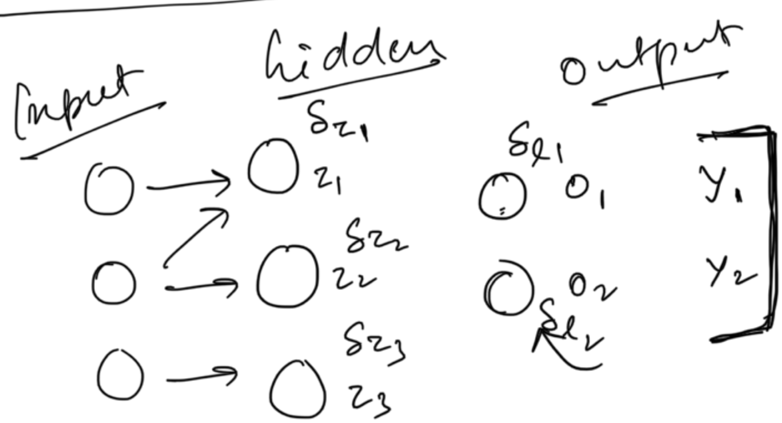




Wed Mar 24

Announcements

- Mid-term grading questions
  - Chandola Office Hours
    3–5 PM on Friday
- PA1 Grading – TAs
- Gradiance 7 – delayed
- Friday lecture "Cancelled"
  - ↳ Deen's office hours.
    1.50 PM – 2.40 PM
    No code
    PA2 only



$\longrightarrow$

Feed

First calculate all δs (δ_1s & δ_2s)

Then update all the weights.

objective fn. / loss function

error (δ) $\to$ unit

Momentum

$$\underbrace{W_{jp}^{(1)}}_{new} \leftarrow \underbrace{W_{jp}^{(1)}}_{old} - \eta \frac{\partial J}{\partial w_{jp}}$$

$$\underbrace{w_{jp}^{(1)}}_{new\_2} = \alpha \underbrace{W_{jp}^{(1)}}_{old} + \underbrace{(1-\alpha)\, W_{jp}^{(1)}}_{new}$$

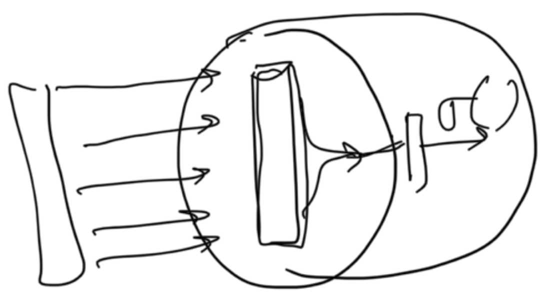
$$\boxed{0 \leq \alpha \leq 1}$$

Universal function approximator

$$x \xrightarrow{f(\,)} y$$

$$\underline{x \quad y}$$

$$\frac{\partial \tilde{J}}{\partial w_{jp}^{(l)}} = \frac{\partial J}{\partial w_{jp}^{(l)}} + \frac{\lambda}{\cancel{2}N} \cancel{2} w_{jp}^{(l)}$$



$\longrightarrow$ MLP

convolutional neural networks.