Page No.

Date 17/07/25

Numerical-1 :-

$$x = \begin{array}{c} x_1 \\ x_2 \\ x_3 \end{array}\begin{bmatrix} 2.0 \\ 1.0 \\ 0.1 \end{bmatrix}_{3\times1} \xrightarrow{\text{softmax}} s = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix}_{3\times1}$$

⇒ How much change in s_1 when there is a change in x_1 or x_2 —

1) compute softmax output; i.e; prob. scores.
2) Compute the Jacobian matrix (J); i.e; 1st order partial derivative.

Ans :- softmax expression —

$$\boxed{s_i = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}}}$$

1) $s_1 = 7.389 / 11.212 = 0.659 \approx 0.66$
$s_2 = 2.718 / 11.212 = 0.24$
$s_3 = 1.105 / 11.212 = 0.098 \approx 0.1$

$$s = \begin{bmatrix} 0.66 \\ 0.24 \\ 0.1 \end{bmatrix}$$

2) $$\boxed{\frac{\partial s_1}{\partial x_1} = s_1(1-s_1)}$$

$$\boxed{J_{ij} = \frac{\partial s_i}{\partial x_j}}$$

Case-1 :- $i=j$ (for diagonal elements)

$$\frac{\partial s_i}{\partial x_i} = s_i(1-s_i)$$

Date

Case-2 :- $i \neq j$ (for off-diagonal elements)

$$\boxed{\frac{\partial s_i}{\partial x_j} = -s_i s_j}$$

$$J = \begin{array}{cccc} a_{11} & a_{12} & a_{13} & \\ \frac{\delta s_1}{\delta x_1} & \frac{\delta s_1}{\delta x_2} & \frac{\delta s_1}{\delta x_3} & \\ \frac{\delta s_2}{\delta x_1} & \frac{\delta s_2}{\delta x_2} & \frac{\delta s_2}{\delta x_3} & = ? \\ \frac{\delta s_3}{\delta x_1} & \frac{\delta s_3}{\delta x_2} & \frac{\delta s_3}{\delta x_3} & \\ a_{31} & a_{32} & a_{33} & \end{array}$$

$i = j$

$$\frac{\delta s_1}{\delta x_1} = s_1(1-s_1) = 0.66 \times 0.34 = 0.224$$

$$\frac{\delta s_2}{\delta x_2} = s_2(1-s_2) = 0.24 \times 0.76 = 0.182$$

$$\frac{\delta s_3}{\delta x_3} = s_3(1-s_3) = 0.1 \times 0.9 = 0.09$$

$i \neq j$

$$\frac{\delta s_1}{\delta x_2} = -(s_1)(s_2) = -0.66 \times 0.24 = -0.158$$

$$\frac{\delta s_1}{\delta x_3} = -(s_1)(s_3) = -0.66 \times 0.1 = -0.066$$

$$\frac{\delta s_2}{\delta x_1} = -(s_2)(s_1) = -0.24 \times 0.66 = -0.158$$

$$\frac{\delta s_2}{\delta x_3} = -(s_2)(s_3) = -0.24 \times 0.1 = -0.024$$

Date

$$\frac{\delta s_3}{\delta x_1} = -(s_3)(s_1) = -0.1 \times 0.66 = -0.066$$

$$\frac{\delta s_3}{\delta x_2} = -(s_3)(s_2) = -0.1 \times 0.24 = -0.024$$

$$J = \begin{bmatrix} 0.224 & -0.158 & -0.066 \\ -0.158 & 0.182 & -0.024 \\ -0.066 & -0.024 & 0.09 \end{bmatrix}_{3\times 3}$$

Forward Pass

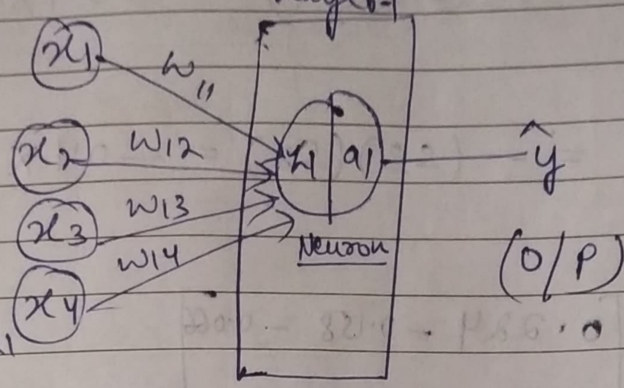
Page No.
Date 18/07/25

Numerical-2 —

x

$$x = \begin{bmatrix} 0.2 \\ -1.3 \\ 2 \\ 0.01 \end{bmatrix}_{4\times1}$$

(I/P)



Linear:- $z_1 = w_{11}x_1 + w_{12}x_2 + w_{13}x_3 + w_{14}x_4$

Non-linearity introduction: $a_1 = \text{softmax}(z_1) \rightarrow$ K/a "$\hat{y}$"

layer-1 → x_1 going to layer-1

$$\begin{bmatrix} w_{11} \\ w_{12} \\ w_{13} \\ w_{14} \end{bmatrix} = \begin{bmatrix} 0.001 \\ 0.01 \\ -0.005 \\ -1.2 \end{bmatrix}$$

Ans:- $\hat{y} = ?$

$$z_1 = 0.001 \times 0.2 + 0.01 \times (-1.3) + (-0.005) \times 2 + (-1.2) \times 0.01$$

$$z_1 = 0.0002 - 0.013 - 0.01 - 0.012$$

$$\boxed{z_1 = -0.0348}$$

$a_1 = \text{softmax}(z_1)$

$a_1 = \text{softmax}(-0.0348) \Rightarrow \boxed{z_1 = -ve \text{ so, } a_1 = 1}$

Approach-2 (P.T.O) Softmax expression $\Rightarrow S_i = \dfrac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}} = \dfrac{e^{-0.0348}}{e^{-0.0348}} = 1$

$$\Rightarrow \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \begin{bmatrix} w_{11} & w_{12} & w_{13} & w_{14} \end{bmatrix} \Rightarrow \begin{bmatrix} 0.2 \\ -1.3 \\ 2 \\ 0.01 \end{bmatrix} \begin{bmatrix} 0.001 & 0.01 & -0.005 & -1.2 \end{bmatrix}$$

$\Rightarrow w_{11} \times x_1 + w_{12} \times x_2 + w_{13} \times x_3 + w_{14} \times x_4$

$\Rightarrow 0.2 \times 0.001 + (-1.3) \times 0.01 + (-0.005) \times 2$
$+ (-1.2) \times 0.01$

$\Rightarrow \boxed{z_1 = -0.0348}$ $\Rightarrow$ Same value after converting into matrix form.

ReLU So, if z_1 is negative, i.e., < 0, then

$a_1 = \text{Relu}(-0.0348)$

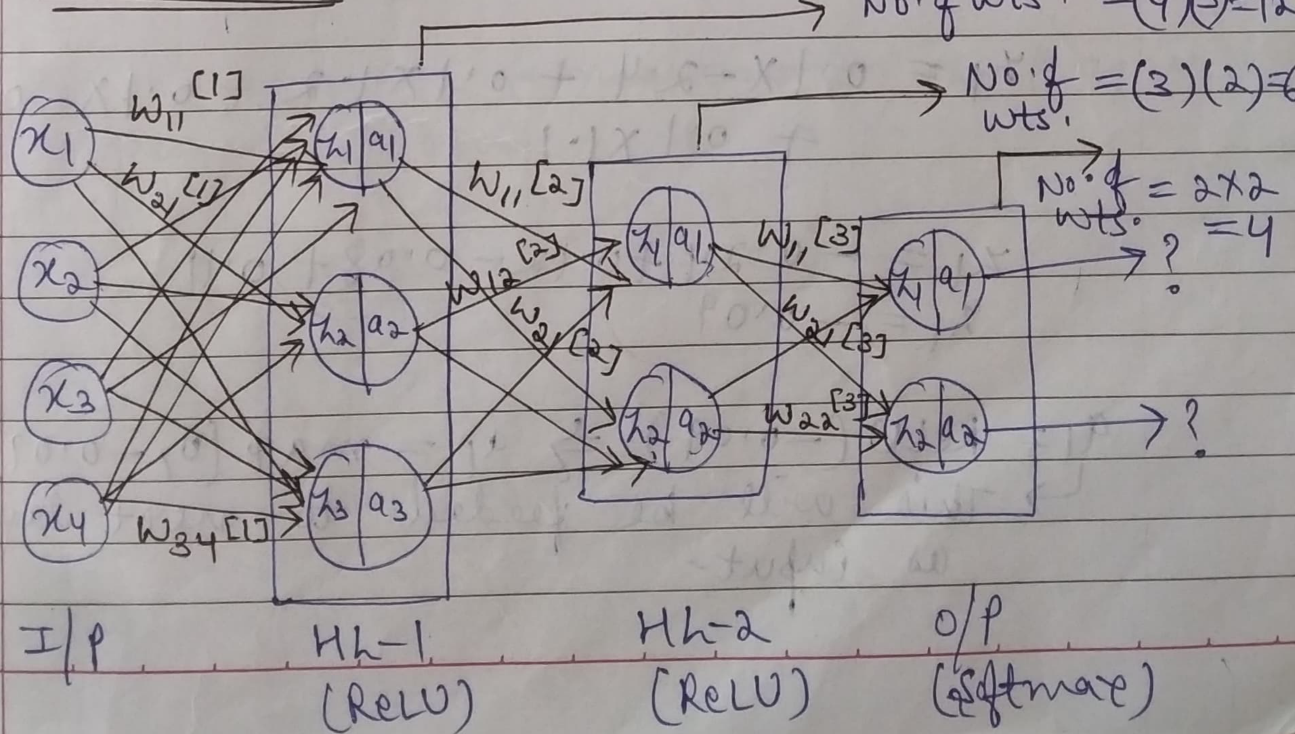
$a_1 = \max(0, z_1)$

$a_1 = \max(0, -0.0348)$

$\boxed{a_1 = 0}$

Numerical-3 — (Lab-2)

<u>Notation</u> :- $W_{(2)(x)}^{[\text{layer}]}$



To calculate no. of weights at each layer level ⇒ For example, in layer-1, $(4)(3) = 12 \leftarrow$ (no. of 'x') (neuron no.)

For layer-1 —

$$\begin{bmatrix} W_{11}^{[1]} \\ W_{21}^{[1]} \\ W_{31}^{[1]} \\ W_{41}^{[1]} \\ \vdots \end{bmatrix} \Rightarrow \begin{bmatrix} 0.1 \\ 0.1 \\ 0.1 \\ 0.1 \\ \vdots \end{bmatrix}$$

For layer-2 —

wt. ⇒ 0.001

For layer-3 —

wt. ⇒ 0.01

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \Rightarrow \begin{bmatrix} -2.4 \\ 1.2 \\ -0.8 \\ 1.1 \end{bmatrix}$$

Ans :- For layer-1 (ReLU applied) —

Neuron-1 (z_1) $\Rightarrow W_{11}^{[1]} \times x_1 + W_{12}^{[1]} \times x_2 + W_{13}^{[1]} \times x_3 + W_{14}^{[1]} \times x_4$

$$z_1 = 0.1 \times -2.4 + 0.1 \times 1.2 + 0.1 \times -0.8 + 0.1 \times 1.1$$

$$z_1 = -0.24 + 0.12 - 0.08 + 0.11$$
$$z_1 = -0.09$$

$a_1 = \text{ReLU}(-0.09) \Rightarrow a_1 = \max(0, -0.09) = 0$

↳ This will be feeded to next layer as input.

Feed-forward network ⇒ bias term is added to each neuron while computing z.
Notation for bias term ⇒ $b_{(z)}^{[layer]}$

Neuron-2 (z_2) ⇒ -0.09

$a_2 = 0$

Neuron-3 (z_3) ⇒ -0.09

$a_3 = 0$

Inputs in code { I/P, HL no., activ. func, wt. (random) } ↓ O/P = ?

Lab-2

For layer-2 (ReLU applied) —

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \Rightarrow \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Neuron-1 (z_1) ⇒ $W_{11}^{[2]} \times x_1 + W_{12}^{[2]} \times x_2 + W_{13}^{[2]} \times x_3$

$z_1 \Rightarrow 0.001 \times 0 + 0.001 \times 0 + 0.001 \times 0$

$z_1 = 0$

$a_1 = \text{ReLU}(0) = \max(0, 0) = 0$

For neuron-2 ~~& neuron-3~~, z_2 & a_2 will remain same, i.e., 0.

For layer-3 (softmax applied) —

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \Rightarrow \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \Rightarrow \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Neuron-1 (z_1) & neuron-2 (z_2) ⇒ $W_{11}^{[3]} \times x_1 + W_{12}^{[3]} \times x_2$

$+ w_{21}^{[3]} \times x_1 + w_{22}^{[3]} \times x_2$

$z_1 \Rightarrow 0.01 \times 0 + 0.01 \times 0 = 0$

$\&$

z_2

$a_1 \Rightarrow \text{softmax}(z_1 \Rightarrow \frac{e^{z_1}}{e^{z_1} + e^{z_2}} = \frac{1}{1+1} = \frac{1}{2}$

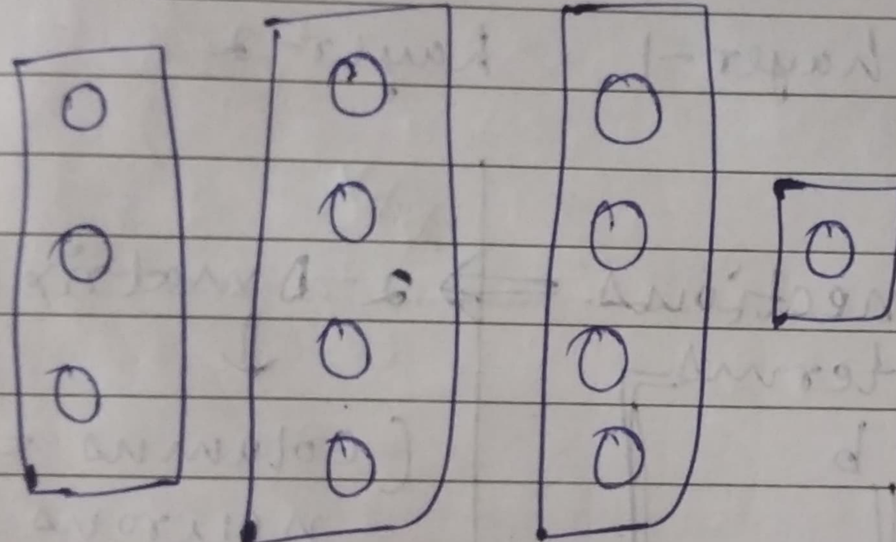
$\&$ $\quad\quad\quad\quad \&$

$a_2 \quad\quad\quad\quad z_2)$

$= 0.5$

$\Rightarrow \frac{e^{x_2}}{e^{x_2} + e^{x_1}} = \frac{1}{2}$

Q1



I/P → layer-1 → bias

- $4 \times 3 = 12 + 4 = 16$

→ bias

- I/P $4 \times 4 = 16 + 4 = 20$

→ Layer-2

- $4 \times 1 = 4$

I/P O/P

- $4 + 1 = 5$

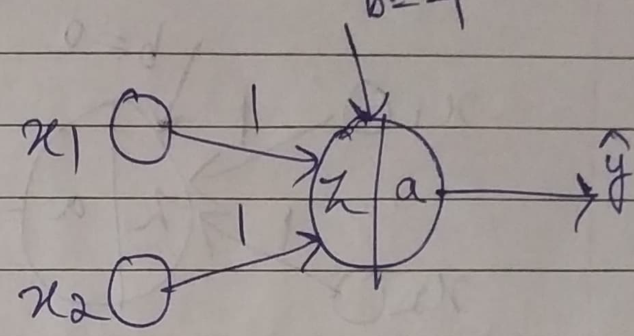
↳ bias

Total capacity ⇒ $16 + 20 + 5 = 41$

Tomodel

* AND GATE —
(using perceptron)

	x_1	x_2	y
Case-1	0	0	0
" -2	0	1	0
" -3	1	0	0
" -4	1	1	1



x_1 & x_2 are the inputs of AND gate & pass them onto the network & compute $\hat{y}$ & match $\hat{y}$ with y or is not matching?

case-1

Ans:- $z = x_1 \times 1 + x_2 \times 1 + (-1)$

$z = 0 + 0 - 1$

$z = -1$

$a = g(-1)$

Since z is -ve, so $\hat{y} = 0$

$\hat{y}$ matches with y

Case-2

$z = 0 + \cancel{1} - \cancel{1} = 0$

$\hat{y} = 0$

case-3

$z = \cancel{1} + 0 - \cancel{1} = 0$

$\hat{y} = 0$

case-4

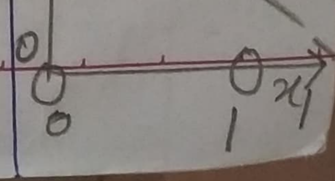
$z = 1 + \cancel{1} - \cancel{1}$

$z = 1$

$\hat{y} = 1$

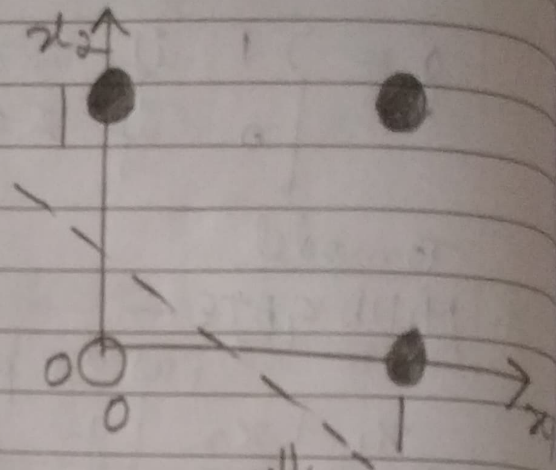
all matches

x_1 AND x_2 (able to separate)

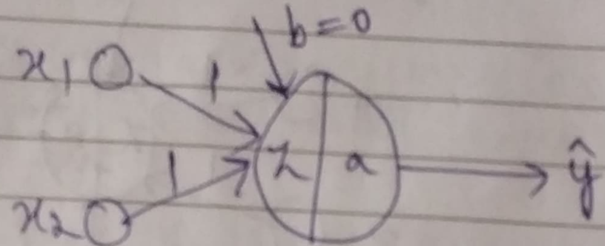


* To model OR gate —

	x_1	x_2	y
Case-1	0	0	0
" -2	0	1	1
" -3	1	0	1
" -4	1	1	1



⇓
able to separate
⇓
x_1 OR x_2



Case-1

$z = 0+0+0 = 0$

$\hat{y} = 0 \approx y$ ✓

Case-2

$z = 0+1+0 = 1$

$\hat{y} \approx y = 1$ ✓

Case-3

$z = 1+0+0 = 1$

$\hat{y} = y = 1$ ✓

Case-4

$z = 1+1+0 = 2$

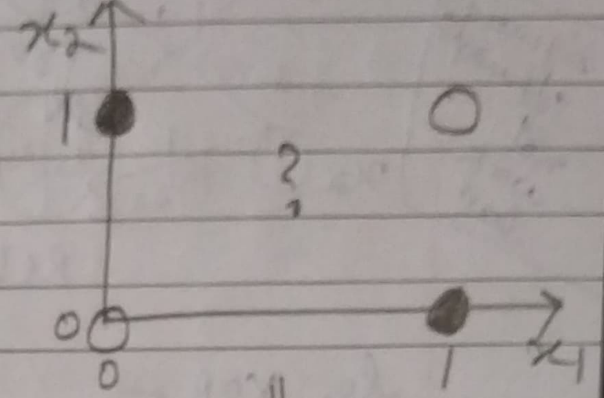
z is > 0

so $\hat{y} = 1 \approx y$ ✓

∴ Perceptron is able to model both AND & OR but not XOR

* To model XOR gate —

	x_1	x_2	y
case-1	0	0	0
" -2	0	1	1
" -3	1	0	1
" -4	1	1	$\boxed{0}$



⇓

x_1 XOR x_2

⇓

can't model

⇓

~~Perceptron~~ can't model non-linear data

⇓

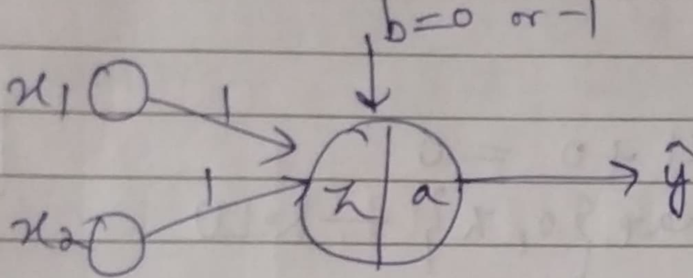
so, add a hidden layer to the perceptron

⇓

Then, it will be able to model well. (P.T.O.)

↓

bec; we transform original datapoints ~~to a new hidden~~ space to make it linear.



case-1

$z = 0+0+0 = 0$

$\hat{y} = 0$ ✓

case-2

$z = 0+1+0 = 1$

$\hat{y} = 1$ ✓

case-3

$z = 1+0+0 = 1$

$\hat{y} = 1$

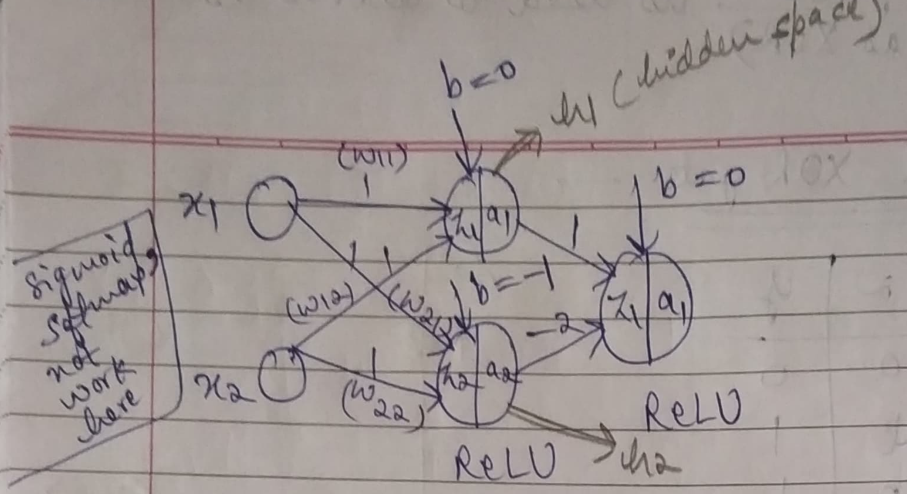
case-4

$z = 1+1+0 = 2$

$z > 0$, so $\hat{y} = 1$

but $y = 0$

(not matching)



So, XOR —

⇓

Layer-1 :-

case-1 :- $z = 0\times1 + 0\times1 + 0 = 0$

(Neuron-1) $g(z) = \max\{0, z\}$ — ReLU

$g(0) = \max\{0, 0\} = 0$

$\hat{y} = y = 0$

case-1 :- $z = 0\times1 + \cancel{0\times1} + \cancel{(-1)} = -1$

(Neuron-2)

$g(z) = \max\{0, -1\} = 0$

$\hat{y} = y = 0$

$\rightarrow x_1 \times w_{11} + x_2 \times w_{12} + b$

case-2 :- $z = 0\times1 + 1\times1 + 0 = 1$

(Neuron-1)

$g(z) = \max\{0, 1\}$

$\hat{y} = y = 1$

case-2 :- $z = 0\times\cancel{1} + 1\times\cancel{1} + \cancel{(-1)} = 0$

(Neuron-2) $g(0) = \max\{0, 0\}$

$\hat{y} \neq y = 0$

case-3 :-

(Neuron-1) $z = 1\times1 + 0\times1 + 0 = 1$

$g(z) = \max\{0, 1\} = 1$

$\hat{y} = y = 1$

Case-3 :- $z = 1\times1 + 0\times1 + (-1) = 0$
(Neuron-2) $g(0) = max\{0, 0\} = 0$
$\hat{y} \neq y = 0$

Case-4 :- $z = 1\times1 + 1\times1 + 0 = 2$
(Neuron-1) $\hat{y} \neq y = 2$

Case-4 :- $z = 1\times1 + \cancel{1\times1} + \cancel{(-1)} = 1$
(Neuron-2) $\hat{y} \neq y = 1$

Layer-2 :-

$a_1^{[1]} = 0$, $a_2^{[1]} = 0$ Case-1 :- $z = 0\times1 + 0\times(-2) + 0 = 0$
$g(0) = max\{0, 0\} = 0$
$\hat{y} = y = 0$ ✓

$a_1^{[1]} = 1$, $a_2^{[1]} = 0$ Case-2 :- $z = 1\times1 + 0\times(-2) + 0 = 1$
$g(1) = max\{0, 1\} = 1$ ✓

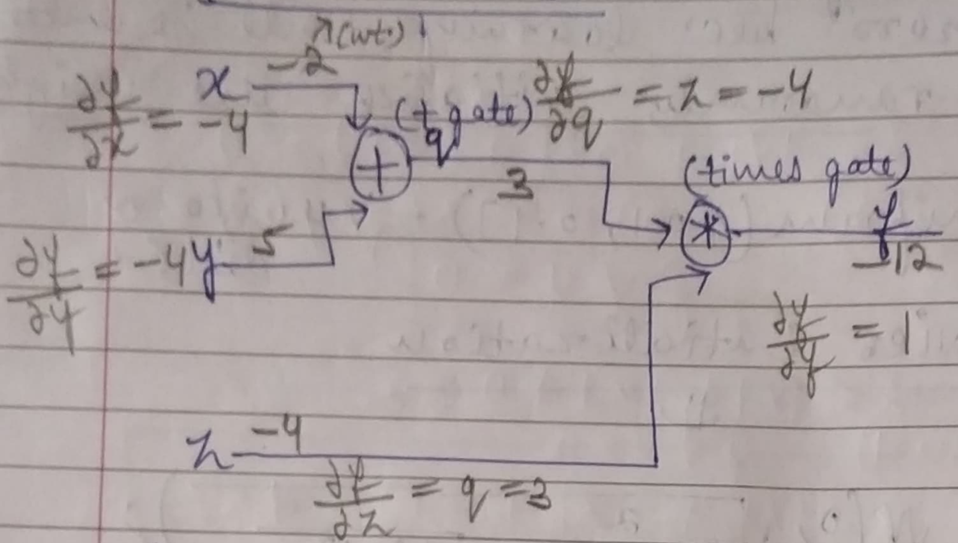
$a_1^{[1]} = 1$, $a_2^{[1]} = 0$ Case-3 :- $z = 1\times1 + 0\times(-2) + 0 = 1$
$\hat{y} = y = 1$ ✓

$a_1^{[1]} = 2$, $a_2^{[1]} = 1$ Case-4 :- $z = 2\times1 + 1\times(-2) + 0 = 0$
$g(0) = max\{0, 0\} = 0$
$\hat{y} = y = 0$ ✓

$\therefore$ Introducing the additional layers in the network can model the XOR gate.

If data is transformed from original space (x_1, x_2) to new space (h_1, h_2) where it is linearly separable.

Gates & circuits —



Let $q = x + y$; so $\frac{\partial q}{\partial x} = 1$ and $\frac{\partial q}{\partial y} = 1$

$f = qz$; $\frac{\partial f}{\partial q} = z$ and $\frac{\partial f}{\partial z} = q$

How changes in input affect the change in function? ⇒ gradient

How much loss func. gets affected with change in wts.?

$q = -2 + 5$

$q = 3$

$f = 3 \times -4$

$f = -12$

To compute gradients —

$\frac{\partial f}{\partial x}$, $\frac{\partial f}{\partial y}$, $\frac{\partial f}{\partial z} = ?$

Always start from right to left (backpropagate) while computing the gradients.

Now, for —

chain rule :— $\frac{\partial f}{\partial x} = \underbrace{\frac{\partial f}{\partial q}}_{\text{global gradient}} \cdot \underbrace{\frac{\partial q}{\partial x}}_{\text{local gradient}} = -4 \times 1 = -4$

29/07/25

Similarly,

$$\frac{\partial f}{\partial y} = \frac{\partial f}{\partial q} \cdot \frac{\partial q}{\partial y} = -4 \times 1 = -4$$

$\therefore$ we can say that, q is negatively impacting f by 4 times.

---

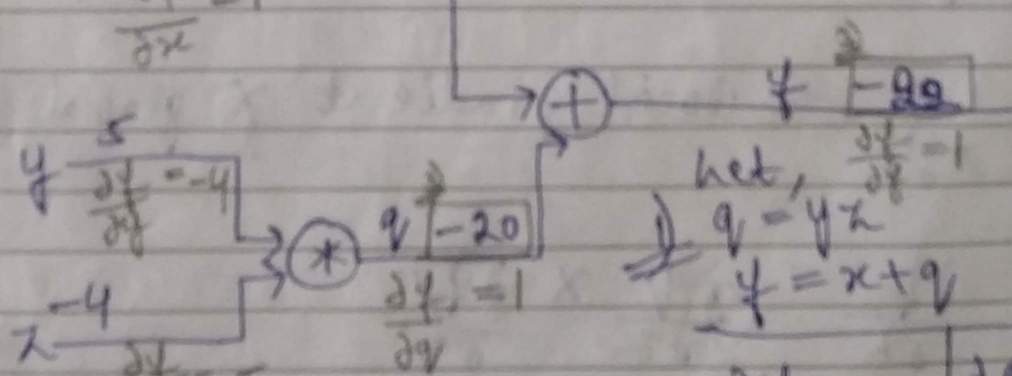
wt. → top
Gradient → below the line

Q2 $f(x, y, z) = x + yz$

$x = -2,\ y = 5,\ z = -4$

Hw:- 1) Identify additional functions from the given expression & compute local derivatives
2) Draw a computational graph
3) Perform forward pass
4) Perform backward pass starting from the end of the circuit.

2)



Let,
1) $q = yz$
$f = x + q$

$\frac{\partial f}{\partial x} = 1$	$\frac{\partial q}{\partial y} = z$
$\frac{\partial f}{\partial q} = 1$	$\frac{\partial q}{\partial z} = y$

3) Forward pass —

$q = yz = 5 \times (-4) = -20$

$f = x + q$

$f = (-2) * (-20)$

$f = -22$

4) **Backward pass** –

$\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} = ?$

$\frac{\partial f}{\partial x} = 1$ (already computed in step-1)

Right ~~from~~ to left –

- $\frac{\partial f}{\partial f} = 1$

- $\frac{\partial f}{\partial x} = 1$

- $\frac{\partial f}{\partial q} = 1$ (already computed in step-1)

- $\frac{\partial f}{\partial z} = ?$

$\frac{\partial f}{\partial z}$ = local gradient × global gradient

$\frac{\partial f}{\partial z} = \frac{\partial q}{\partial z} \times \underbrace{\frac{\partial f}{\partial q}}_{1}$

$\frac{\partial f}{\partial z} = y$

$\boxed{\frac{\partial f}{\partial z} = 5}$ ⇒ For small change in z, there is the impact on f by 5.

Date

- $\frac{\partial f}{\partial y} = \frac{\partial q}{\partial y} \times \frac{\cancel{\partial f}}{\cancel{\partial q}}$

$\frac{\partial f}{\partial y} = z$

$$\boxed{\frac{\partial f}{\partial y} = -4}$$

Q3 $\underset{\text{(sigmoid)}}{f(w, x)} = \frac{1}{1+e^{-(w_0x_0 + w_1x_1 + w_2)}}$

$w_0 = 2$

$x_0 = -1$

$w_1 = -3$

$x_1 = -2$

$w_2 = -3$

1) Simplify the expression — ↳ (every gate should have 2 inputs)

$f_1 = w_0 x_0$

$f_2 = w_1 x_1$

$f_3 = f_1 + f_2 \cancel{+ w_2}$

$f_4 = \cancel{f_3}\ f_3 + w_2$; $f_5 = -f_4$

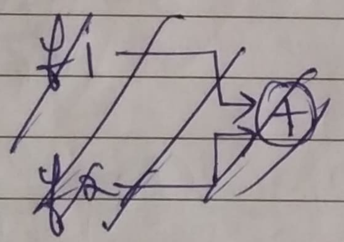
$f_6 = e^{f_5}$

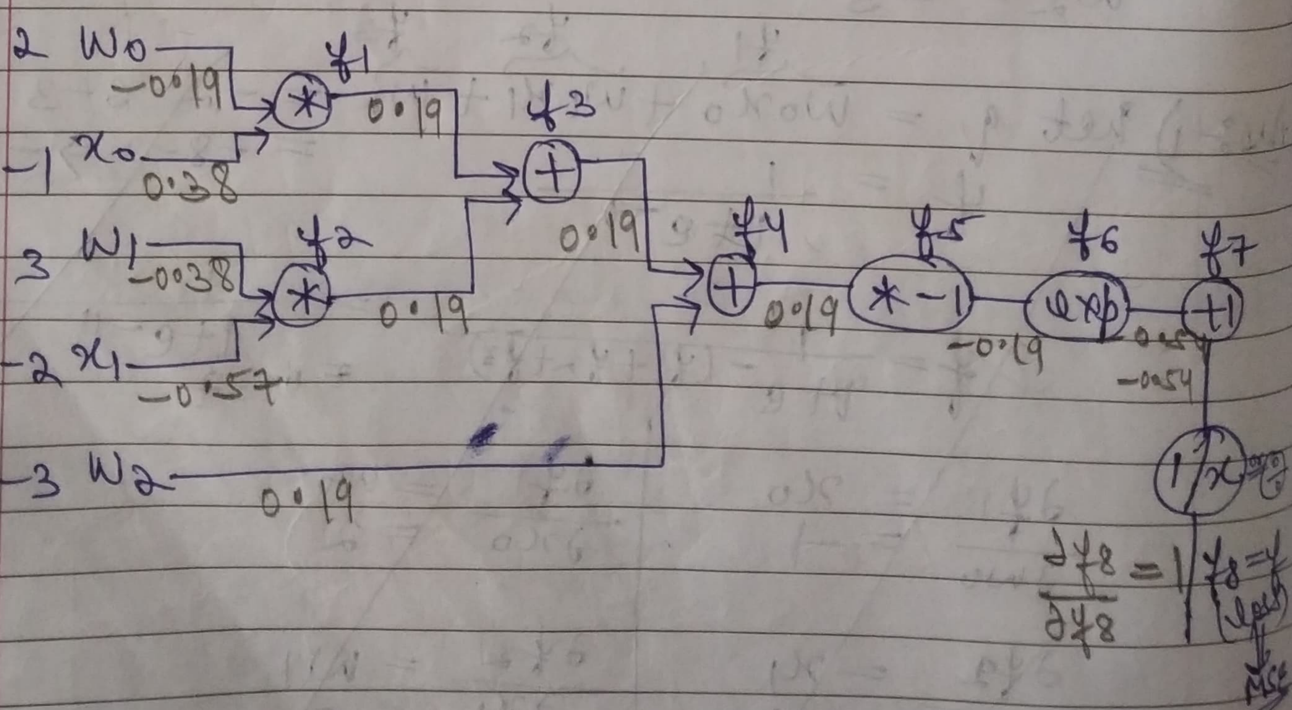
$f_7 = 1 + f_6$

$f_8 = \frac{1}{f_7}$

$\boxed{f = f_8}$

2) Computational graph —





3) Forward pass —

$f_1 = -2$

$f_2 = +6$

$f_3 = (-2) + (+6) = 4$

$f_4 = (4) + (-3) = +1$

$f_5 = -(+1) = -1$

$f_6 = e^{-1} = \cancel{59874.14172}\ 0.36$

$f_7 = \cancel{59875.14}\ 1.36$

$f_8 = 0.73$

4) Local gradients —

$\frac{\partial f_1}{\partial w_0} = x_0, \quad \frac{\partial f_1}{x_0} = w_0$

$\frac{\partial f_2}{\partial w_1} = x_1, \quad \frac{\partial f_2}{\partial x_1} = w_1$

$\frac{\partial f_3}{\partial f_1} = 1, \quad \frac{\partial f_3}{\partial f_2} = 1$

$\frac{\partial f_4}{\partial f_3} = 1, \quad \frac{\partial f_4}{\partial w_2} = 1$

$\frac{\partial f_5}{\partial f_4} = -1$

$\frac{\partial f_6}{\partial f_5} = \frac{\partial (e^{f_5})}{\partial f_5} = e^{f_5}$

$\frac{\partial f_7}{\partial f_6} = 1$

$\frac{\partial f_8}{\partial f_7} = -\frac{1}{f_7^2}$ (nx^{n-1})

5) Backward pass —
(right → left)

$\frac{\partial f_8}{\partial f_8} = 1$

(loss w.r.t. every variable)

$\frac{\partial f_8}{\partial f_7}$ = local × global (prev.)

$= -\frac{1}{f_7^2} \times 1$

$= -\frac{1}{(1.36)^2}$

$= -0.54$

$\frac{\partial f_8}{\partial f_6}$ = local × global

$= \frac{\partial f_7}{\partial f_6} \times -0.54$

$= 1 \times -0.54$

$= -0.54$

$$\frac{\partial f_8}{\partial f_5} = \text{local} \times \text{global}$$

$$= \frac{\partial f_6}{\partial f_5} \times -0.54$$

$$= e^{f_5} \times -0.54$$

$$= e^{-1} \times -0.54$$

$$= -0.19$$

$$\frac{\partial f_8}{\partial f_4} = \text{local} \times \text{global}$$

$$= \frac{\partial f_5}{\partial f_4} \times -0.19$$

$$= -1 \times -0.19$$

$$= 0.19$$

$\cancel{\frac{\partial f_8}{\partial f_3} = \text{local} \times \text{global}}$

(local) (global)

$$\frac{\partial f_8}{\partial w_2} = \frac{\partial f_4}{\partial w_2} \times 0.19$$

$$= 1 \times 0.19$$

$$= 0.19$$

$$\frac{\partial f_8}{\partial f_3} = \text{local} \times \overset{\text{prev.}}{\text{global}}$$

$$= \frac{\partial f_4}{\partial f_3} \times 0.19$$

$$= 1 \times 0.19$$

$$= 0.19$$

Date

$$\frac{\partial f_8}{\partial f_2} = \text{local} \times \text{global}$$

$$= \frac{\partial f_3}{\partial f_2} \times 0.19$$

$$= 1 \times 0.19$$

$$= 0.19$$

$$\frac{\partial f_8}{\partial f_1} = \text{local} \times \text{global}$$

$$= \frac{\partial f_3}{\partial f_1} \times 0.19$$

$$= 1 \times 0.19$$

$$\frac{\partial f_8}{\partial x_1} = \frac{\partial f_2}{\partial x_1} \times 0.19$$

$$= w_1 \times 0.19$$

$$= -3 \times 0.19$$

$$= -0.57$$

$$\frac{\partial f_8}{\partial w_1} = \frac{\partial f_2}{\partial w_1} \times 0.19$$

$$= x_1 \times 0.19$$

$$= -2 \times 0.19$$

$$= -0.38$$

$$\frac{\partial f_8}{\partial w_0} = \frac{\partial f_1}{\partial w_0} \times 0.19$$

$$= x_0 \times 0.19$$

$$= -0.19$$

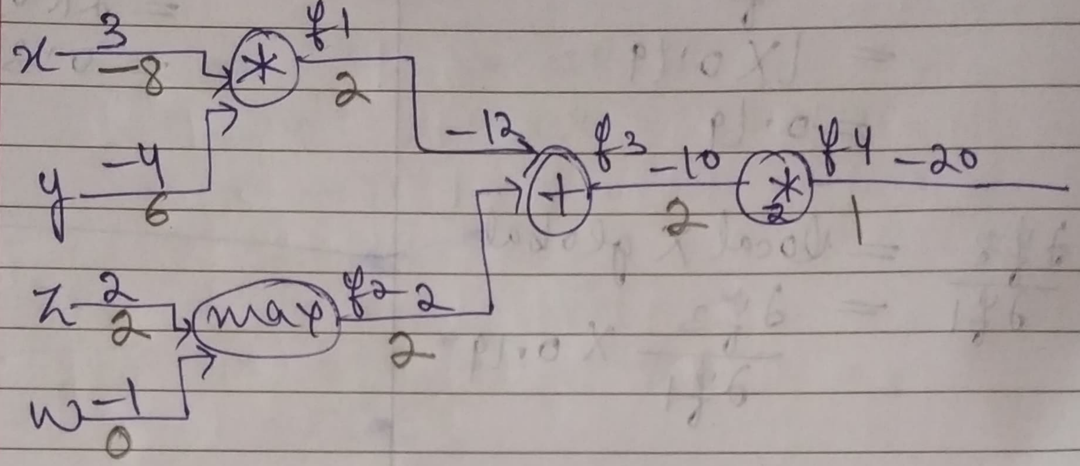
$$\frac{\partial f_8}{\partial x_0} = \frac{\partial f_1}{\partial x_0} \times 0.19$$

$$= w_0 \times 0.19$$

$$= 2 \times 0.19$$

$$= 0.38$$

Q4 Forward pass (already done) —



Do backpropagation.

$f_2 = max(z, w)$ — ReLU

Ans:- $f_1 = xy$

$f_3 = f_1 + f_2$

$f_4 = f_3 \times 2$

$\frac{\partial f_1}{\partial x} = y$, $\frac{\partial f_1}{\partial y} = x$

$\frac{\partial f_3}{\partial f_1} = 1$, $\frac{\partial f_3}{\partial f_2} = 1$

Local gradients

$\frac{\partial f_4}{\partial f_3} = 2$, $\frac{\partial f_2}{\partial z} = 1$ → If $z > w$

$\frac{\partial f_2}{\partial w} = 1$ → If $w > z$

Backward pass — (P.T.O.)

$\frac{\partial f_2}{\partial z} = 0$, $\frac{\partial f_2}{\partial w} = 0$

If $z < w$ or $w < z$

Also, $\frac{\partial f_2}{\partial z}$ = undefined (If $z = w$), $\frac{\partial f_2}{\partial w}$ = undefined (If $w = z$) arbitrary

Backward pass ⇒

$\frac{\partial f_4}{\partial f_4} = 1$

$\frac{\partial f_4}{\partial f_3}$ ⇒ local × global ⇒ $\frac{\partial f_4}{\partial f_3} \times 1$

$\Rightarrow 2 \times 1 = 2$

$\frac{\partial f_4}{\partial f_1}$ = local × global

$= \frac{\partial f_3}{\partial f_1} \times 2$

$= 1 \times 2$

$= 2$

$\frac{\partial f_4}{\partial x}$ = local × global

$= \frac{\partial f_1}{\partial x} \times 2$

$= y \times 2$

$= -4 \times 2$

$= -8$

$\frac{\partial f_4}{\partial y} \Rightarrow \frac{\partial f_1}{\partial y} \times 2$

$\Rightarrow 2x$

$\Rightarrow 2 \times 3 = 6$

$\frac{\partial f_4}{\partial f_2} = \frac{\partial f_3}{\partial f_2} \times 2$

$= 1 \times 2$

$= 2$

Since, $z > w$

$\frac{\partial f_4}{\partial z}$ = local × global

$= \frac{\partial f_2}{\partial z} \times 2$

$= 1 \times 2$

$= 2$

Since $w < z$

$\frac{\partial f_4}{\partial w}$ = local × global

$= \frac{\partial f_2}{\partial w} \times 2$

$= 0 \times 2$

$= 0$