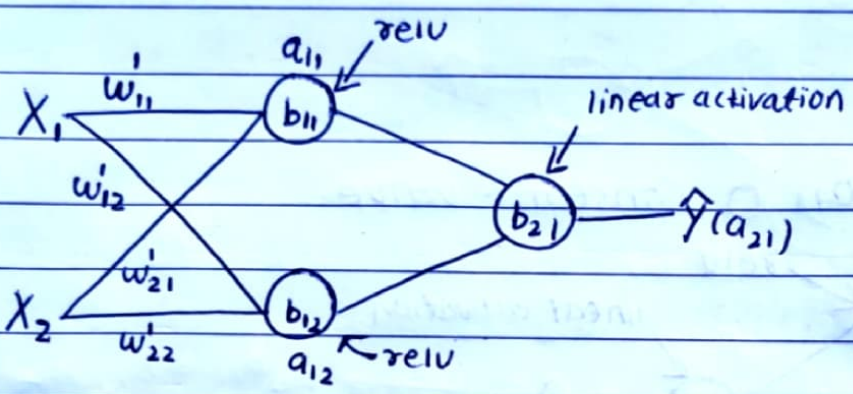
Weight Initialization Techniques:

Wrong weight Initialization Techniques:

1) Case 1 ⇒ Zero Initialization



let's calculate a_{11} (relu activation) ⇒ $a_{11} = \max(0, z_{11})$

where,

$z_{11} = w'_{11} \times X_1 + w'_{21} \times X_2 + b_{11}$

let's calculate a_{12} ⇒ $a_{12} = \max(0, z_{12})$

where,

$z_{12} = w'_{12} \times X_1 + w'_{22} \times X_2 + b_{12}$

Now, if we initilize $w = 0$ and $b = 0$

$z_{11} = 0 X_1 + 0 X_2 + 0 = 0$

$z_{12} = 0 X_1 + 0 X_2 + 0 = 0$

$\therefore a_{11} = \max(0, 0) = 0$

$a_{12} = \max(0, 0) = 0$

Now, during back propogation $\boxed{w'_{11} = w'_{11} - \alpha \frac{dL}{dw'_{11}}}$ we have to do $\frac{dL}{dw'_{11}} \cdot \frac{dl}{dw'_{11}}$ garda yo all ra a_{12} activation ko pani derivative garna parxa so, 0 aauxa $\frac{dl}{dw'_{11}}$ ko vale

So, $\boxed{w'_{11} = w'_{11} - \alpha 0} \Rightarrow w'_{11} = w'_{11}$ weight never get updated. It's not for only w'_{11}. It is for all weight)

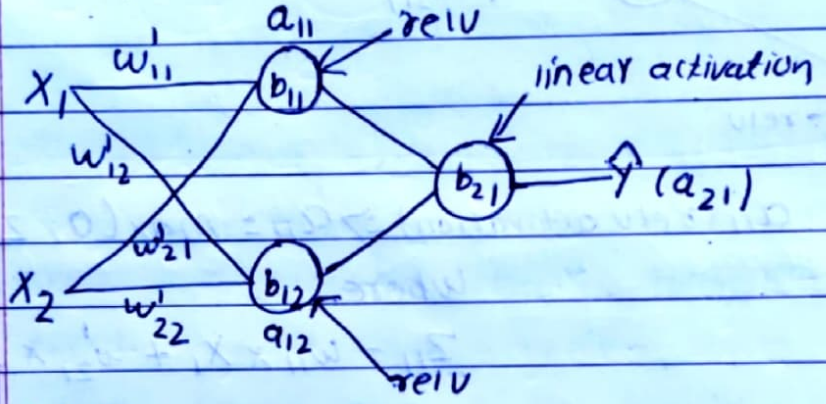
we if initilize all weights 0 and bias = 0

for tanh

$$a_{11} = \frac{e^{z_{11}} - e^{-z_{11}}}{e^{z_{11}} + e^{-z_{11}}} = \frac{e^0 - e^{-0}}{e^0 + e^0} = 0$$

for similary like relu tanh also give activation valve 0 so, no weight updation takes place.

2) Case 2 → Non 0 constant valve.



now,

$a_{11} = \max(0, z_{11})$

$z_{11} = w'_{11} x_1 + w'_{21} x_2 + b_{11}$

result: $z_{11} \geq 0$, $a_{11} \geq 0$ (z_{11} ra a_{11} ko valve non zero aauxa)

$a_{12} = \max(0, z_{12})$

$z_{12} = w'_{12} x_1 + w'_{22} x_2 + b_{12}$

result: $z_{12} \geq 0$, $a_{12} \geq 0$ (z_{12} ra a_{12} ko valve non zero aauxa)

let's initilize $w = 0.5$ and $b = 0.5$

$z_{11} = 0.5 x_1 + 0.5 x_2 + 0.5$

$z_{12} = 0.5 x_1 + 0.5 x_2 + 0.5$

$a_{11} = \max(0, 0.5) = 0.5$

$a_{12} = \max(0, 0.5) = 0.5$

$\therefore \boxed{a_{11} = a_{12}}$

$$\frac{dz_{11}}{dw'_{11}} = \frac{d(w'_{11}x_1 + w'_{21}x_2 + b_1)}{dw'_{11}} = x_1$$

Now, let's do the weight update and calculate $\frac{dL}{dw}$ first.

$$\Rightarrow \frac{dL}{dw'_{11}} = \boxed{\frac{dL}{d\hat{y}} \times \frac{d\hat{y}}{da_{11}} \times \frac{da_{11}}{dz_{11}} \times \frac{dz_{11}}{dw'_{11}}}$$

$$\Rightarrow \frac{dL}{dw'_{12}} = \frac{dL}{d\hat{y}} \times \frac{d\hat{y}}{da_{12}} \times \frac{da_{12}}{dz_{12}} \times \frac{dz_{12}}{dw'_{12}}$$

$$\Rightarrow \frac{dL}{dw'_{21}} = \frac{dL}{d\hat{y}} \times \frac{d\hat{y}}{da_{11}} \times \frac{da_{11}}{dz_{11}} \times \frac{dz_{11}}{dw'_{21}}$$

$$\Rightarrow \frac{dL}{dw'_{22}} = \boxed{\frac{dL}{d\hat{y}} \times \frac{d\hat{y}}{da_{12}} \times \frac{da_{12}}{dz_{12}}} \times \frac{dz_{12}}{dw'_{22}}$$

Since $a_{11} = a_{12}$, $z_{11} = z_{12}$ so, $\frac{d\hat{y}}{da_{11}} = \frac{d\hat{y}}{da_{12}}$ and

$$\frac{da_{11}}{dz_{11}} = \frac{da_{12}}{dz_{12}}$$

Out comes

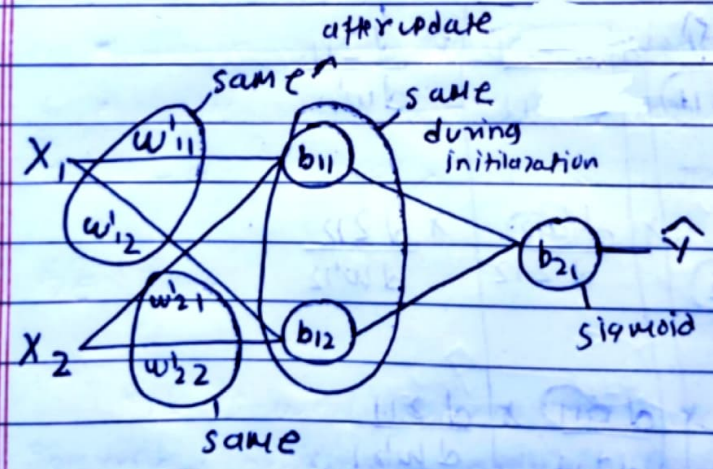
$$\frac{dL}{dw'_{11}} = \boxed{\frac{dL}{d\hat{y}} \times \frac{d\hat{y}}{da_{11}} \times \frac{da_{11}}{dz_{11}} \times x_1}$$

$$\frac{dL}{dw'_{12}} = \boxed{\frac{dL}{d\hat{y}} \times \frac{d\hat{y}}{da_{12}} \times \frac{da_{12}}{dz_{12}} \times x_1} \quad \text{— same}$$

$$\frac{dL}{dw'_{21}} = \boxed{\frac{dL}{d\hat{y}} \times \frac{d\hat{y}}{da_{11}} \times \frac{da_{11}}{dz_{11}} \times x_2}$$

$$\frac{dL}{dw'_{22}} = \boxed{\frac{dL}{d\hat{y}} \times \frac{d\hat{y}}{da_{12}} \times \frac{da_{12}}{dz_{12}} \times x_2} \quad \text{— same}$$

So, from Here we can conclude that
$w'_{11} = w'_{12}$ and $w'_{21} = w'_{22}$

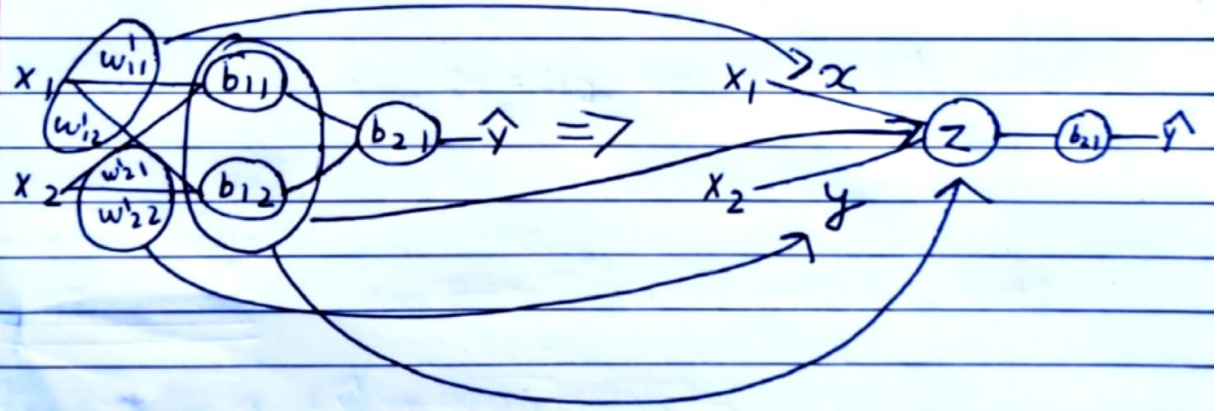


Here,
$w'_{11} = w'_{12}$ xa ra $w'_{21} = w'_{22}$ xa ra

$b_{11} = b_{12}$

Here,
$w'_{11} = w'_{12}$, $w'_{21} = w'_{22}$, $b_{11} = b_{12}$
let x y z

tei vayera x_1 bata euta weight matra ra x_2 bata ne euta weight matra niskeko act garxa neural network ley. b_1 ra b_2 ne equal vara euta matra hidden node act garxa.



So, It will act like perceptron so, non linearity capture hunna. linear model jasto act hunxa.

Conclusion: we need initilize random small numbers for weights and bias.

Weight Initialization in keras:

① Xavier (Glorot) initialization

- It is used when tanh activation function is used (practically proved it works good with tanh)

② He Init

- It is used when relu activation function is used (practically proved it works good with relu).