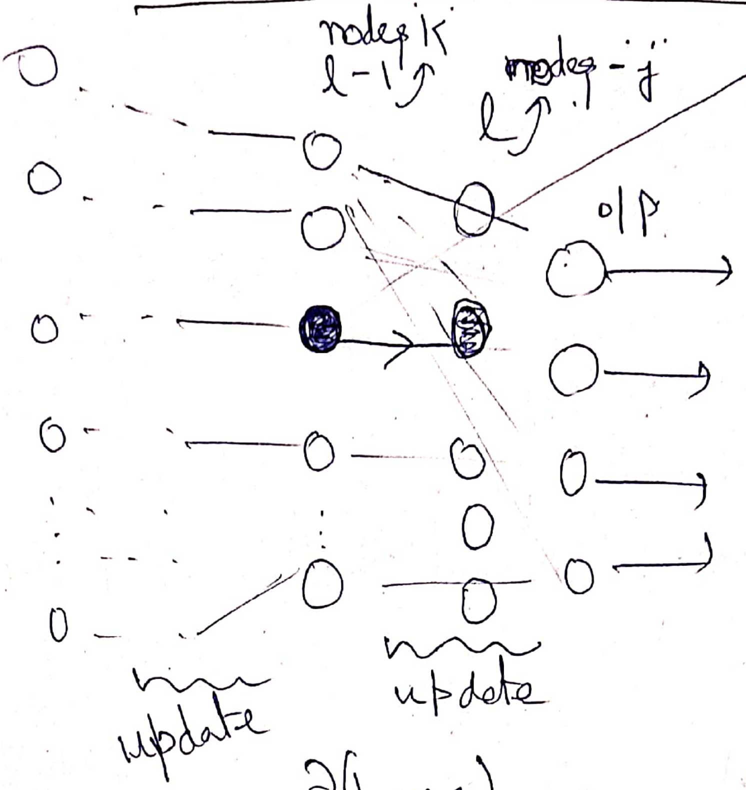
Lec 3 : Multilayer Perceptron
(MLP) → Backpropagation



$[W_{jk}^{(l)}$ = Weights of the connection that connects node k in layer l-1 to j in l.]

"[Loss calculated and updates weight in backward direction based on the amount of Loss produced]"

$\frac{\partial(\text{Loss})}{\partial(\text{weight})}$ — Avg of weight is updated in BP.

Steps:

1) passing data to model via forward propagation.
2) calculate loss on output.
3) SGD minimizes the loss.
   - Gradient is calculated via Backpropagation

W_j^l — the vector that contains all weights connect

$z_j^{(l)}$ — the input for node j in layer l.

$g^{(l)}$ — the activation function used for layer l.

$a_j^{(l)}$ — activation o/p of node j in layer l.

Scanned with CamScanner

Loss C_0: given by.

$$(a_j^{(L)} - y_j)^2$$

$a_j^{(L)}$ ↑ activation o/p @ node j

y_j ↑ desired o/p @ node j

layer 'L'

To calculate total Loss, we should sum this squared difference for all the nodes j in layer 'L'.

$$\boxed{C_0 = \sum_{j=0}^{n-1} (a_j^{(L)} - y_j)^2}$$

Input $z_j^{(l)}$:

I/p for node 'j' in layer 'l' is weighted sum of activation o/p from previous layer (l−1)

ex:

$$w_{jk}^{(l)} \, a_k^{(l-1)}$$

$w_{jk}^{(l)}$ → weights.

$a_k^{(l-1)}$ → o/p.

Input to node 'j' in layer 'l' is expressed as:

$$\boxed{z_j^{(l)} = \sum_{k=0}^{n-1} w_{jk}^{(l)} \, a_k^{(l-1)}}$$

Activation o/p: $a_j^{(l)}$

$a_j^{(l)} \rightarrow$ It is the result of passing $Z_j^{(l)}$ to whatever activation function we choose to use. Say $g^{(l)}$

$(l) \rightarrow$ many @ layer 'l'.

Activation output of node j in layer 'l' is expressed as:

$$\boxed{a_j^{(l)} = g^{(l)}\left(z_j^{(l)}\right)}$$

The i/ps for node j is a function of all weights connected to node 'j'.

So, $z_j^{(L)} \rightarrow \text{func}(W_j^{(L)})$.

$$z_j^{(L)}(w_j^{(L)}).$$

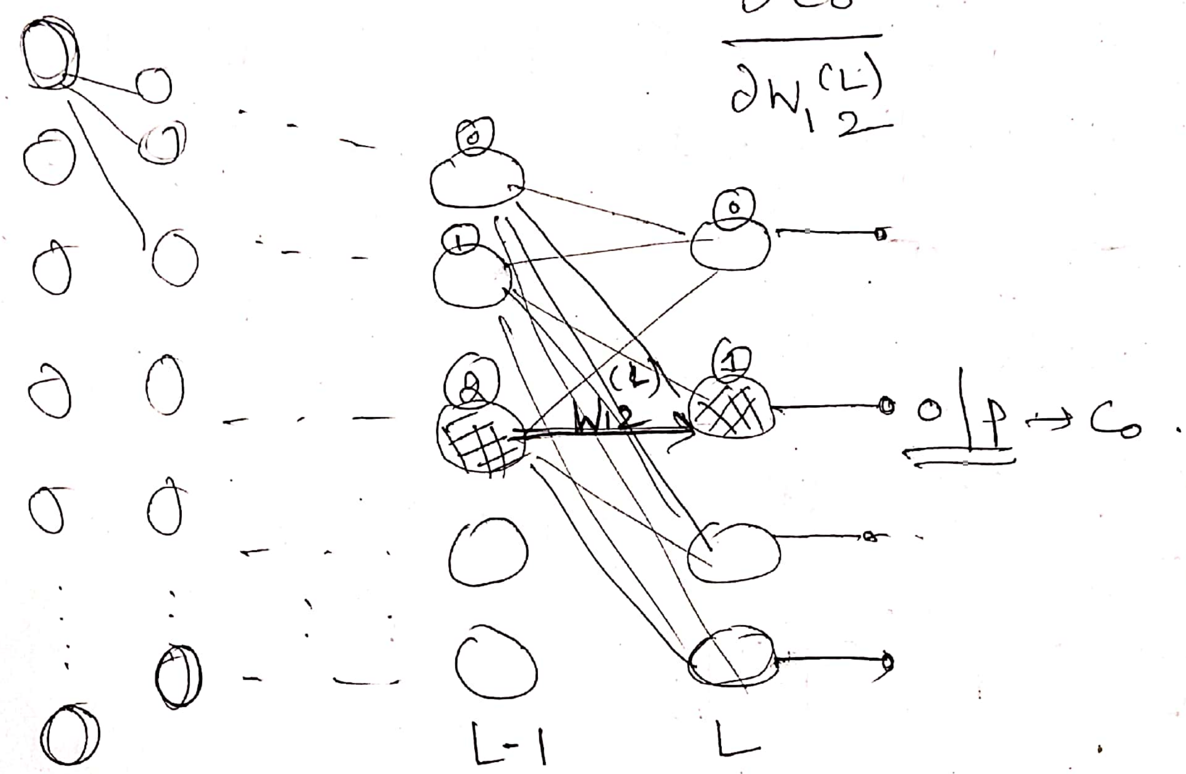
$$\boxed{\therefore C_{0j} = C_{0j}\left(a_j^{(L)}\left(z_j^{(L)}\left(w_j^{(L)}\right)\right)\right)}$$

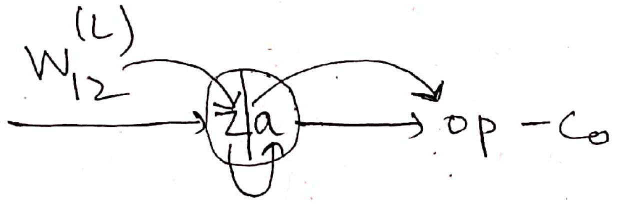
Loss @ node j

$$\boxed{C_0 = \sum_{j=0}^{n-1} C_{0j}}$$

Calculations: Derivative of Loss w.r.t weights. 4

$$\frac{\partial C_0}{\partial W_{12}^{(L)}}$$





C_0 depends on $a_1^{(l)}$
$a_1^{(l)}$ depends on $z_1^{(l)}$
$z_1^{(l)}$ depends on $W_{12}^{(L)}$

} composition of functions

$$\frac{\partial C_0}{\partial W_{12}^{(L)}} = \underset{①}{\left(\frac{\partial C_0}{\partial a_1^{(L)}}\right)} \cdot \underset{②}{\left(\frac{\partial a_1^{(L)}}{\partial z_1^{(L)}}\right)} \cdot \underset{③}{\left(\frac{\partial z_1^{(L)}}{\partial W_{12}^{(L)}}\right)}$$

Consider this: First term

$$C_0 = \sum_{j=0}^{n-1} (a_j^{(L)} - y_j)^2 \Rightarrow \frac{\partial C_0}{\partial a_j^{(L)}} = \frac{\partial}{\partial a_j}\left((a_0^{(L)} - y_0)^2 + (a_1^{(L)} - y_0)^2 + (a_2^{(L)} - y_2)^2 + (a_3^{(L)} - y_3)^2\right)$$

derivative of sum = sum of derivatives:

$$\left[\frac{\partial(\_+\_+\_)}{\partial x} = \frac{\partial(\,)}{\partial x} + \frac{\partial(\,)}{\partial x} + \frac{\partial(\,)}{\partial x}\right]$$

$$\frac{\partial C_0}{\partial a_1^{(L)}} = \cancel{\frac{\partial}{\partial a_1^{(L)}}(a_0^{(L)}-y_0)^2} + \cancel{\frac{\partial}{\partial a_1^{(L)}}(a_1^{(L)}-y_1)^2} + \cancel{\frac{\partial}{\partial a_1^{(L)}}(a_2^{(L)}-y_2)^2} + \cancel{\frac{\partial}{\partial a_2^{(L)}}(a_3^{(L)}-y_3)^2}$$

0 0 0

$$\boxed{\frac{\partial C_0}{\partial a_1^{(L)}} \Rightarrow 2(a_1^{(L)}-y_1)} \rightarrow 1\text{st}$$

Second term in $\frac{\partial C_0}{\partial w_{12}^{(L)}}$ 2

We know:

$$a_j^{(L)} = g^{(L)}(z_j^{(L)}) \rightarrow \text{general}$$

$$a_1^{(L)} = g^{(L)}(z_1^{(L)}) \rightarrow \text{for node 'i' in layer L}$$

$$\frac{\partial a_1^{(L)}}{\partial z_1^{(L)}} = \frac{\partial}{\partial z_1^{(L)}} g^{(L)}(z_1^{(L)})$$

$$\boxed{\frac{\partial a_1^{(L)}}{\partial z_1^{(L)}} = g'^{(L)}(z_1^{(L)})} \rightarrow 2\text{nd}$$

Third term: $\frac{\partial z_1^{(L)}}{\partial w_{12}^{(L)}}$

We know,

$$z_j^{(L)} = \sum_{k=0}^{n-1} w_{jk}^{(L)} a_k^{(L-1)}$$

$\underline{j=1}$,

$$z_1^{(L)} = \sum_{k=0}^{n-1} w_{1k}^{(L)} a_k^{(L-1)}.$$

$$\frac{\partial\left(\sum_{k=0}^{n-1} w_{1k}^{(L)} a_k^{(L-1)}\right)}{\partial w_{12}^{(L)}} = \cancel{\frac{\partial\left(w_{10}^{(L)} a_0^{(L-1)}\right)}{\partial w_{12}^{(L)}}}^{0} + \cancel{\frac{\partial\left(w_{11}^{(L)} a_1^{(L-1)}\right)}{\partial w_{12}^{(L)}}}^{0}$$

expand summation

$$\left(+ \frac{\partial\left(w_{12}^{(L)} a_2^{(L-1)}\right)}{\partial w_{12}^{(L)}} + \right) \cdots$$

$+ \cdots$ Rest will be zero

only this will remain.

$$\boxed{\frac{\partial z_1^{(L)}}{\partial w_{12}^{(L)}} = a_2^{(L-1)}} \rightarrow 3^{rd}$$

Substitute $1^{st}, 2^{nd}, 3^{rd}$ in Loss derivative,

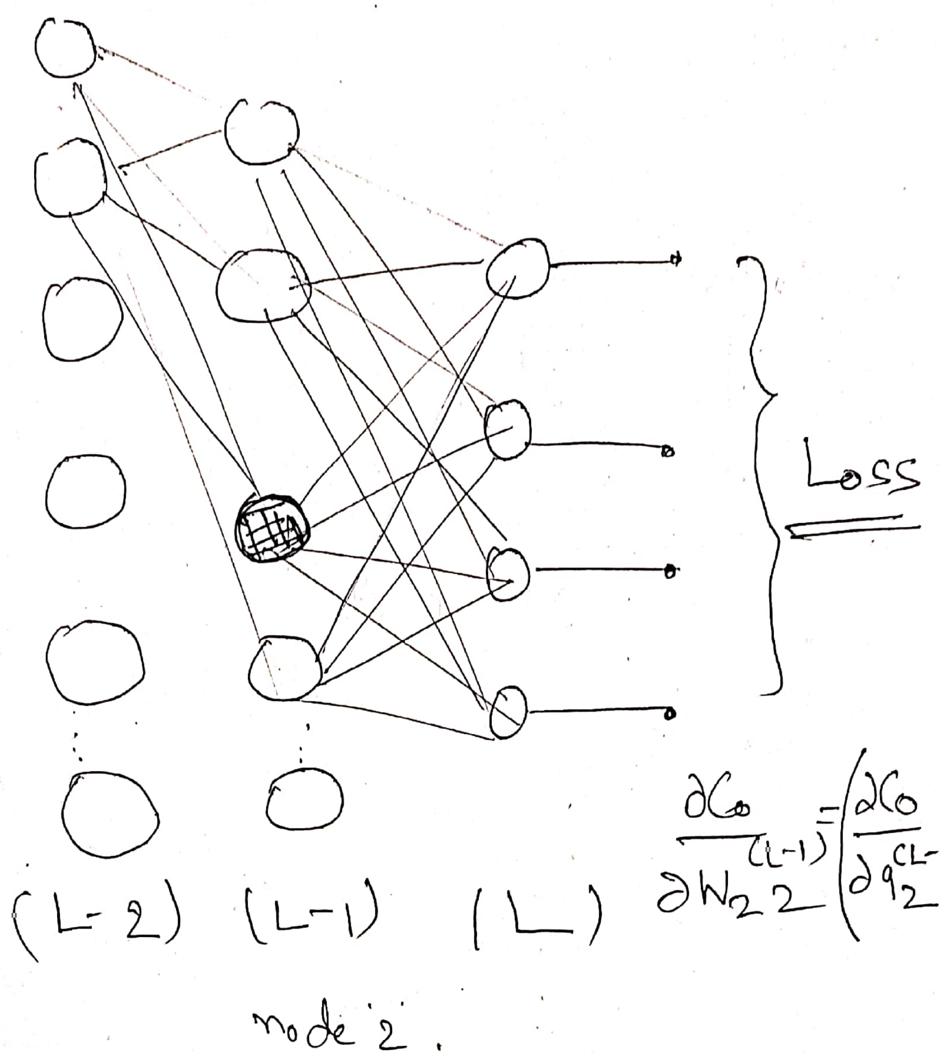
$$\boxed{\frac{\partial C_0}{\partial w_{12}^{(L)}} = \left[2\left(a_1^{(L)} - y_1\right)\right] \cdot \left[g'^{(L)} \cdot \left(z_1^{(L)}\right)\right] \cdot \left[a_2^{(L-1)}\right]}$$

$\hookrightarrow$ This is Loss Calculation w.r.t weight w_{12} for 1 training sample.

To consider all the training samples, we have to Avg:

$$\boxed{\frac{\partial C_0}{\partial w_{12}^{(L)}} = \frac{1}{n} \sum_{i=0}^{n-1} \frac{\partial C_i}{\partial w_{12}^{(L)}}}$$

$\rightarrow$ We can calculate similarly for all other weights in Network or Layer.



$$\frac{\partial C_0}{\partial w_{22}^{(L-1)}} = \left(\frac{\partial C_0}{\partial a_2^{(L-1)}}\right) \cdot \left(\frac{\partial a_2^{(L-1)}}{\partial z_2^{(L-1)}}\right) \cdot \left(\frac{\partial z_2^{(L-1)}}{\partial w_{22}^{(L-1)}}\right)$$

$$\boxed{\frac{\partial C_0}{\partial a_2^{(L-1)}}}$$

$\boxed{\text{In some books } a = h}$

→ To find this we need to find the product of the derivatives of the composed functions.

$$\underbrace{z_j^{(L)}}_{\downarrow} = \sum_{k=0}^{n-1} w_{jk}^{(L)} a_k^{(L-1)}$$

$$\boxed{\therefore \frac{\partial z_j^{(L)}}{\partial a_2^{(L-1)}} = \frac{\partial}{\partial a_2^{(L-1)}} \sum_{k=0}^{n-1} w_{jk}^{(L)} a_k^{(L-1)}}$$ → expand.