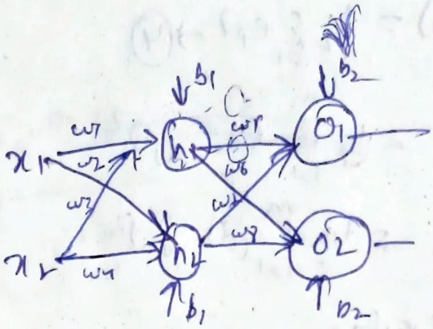
***

17/9/22

• Back Propagation Algorithm :-



It has 2 phases

① Forward phase

② Backward phase

①⇒ network fed with i/p.

- Neurons in hidden layers process the i/p & pass to next layers

②⇒ It is the error connection pass.

- error value is computed

• when there is an error, then back propagation alg uses gradient descent to minimize error value

$$w(n+1) = w(n) - \eta g(n)$$

• computing gradient of o/p layer

$$E_{Total} = E_{o_1} + E_{o_2} + \dots + E_{o_n}$$

$$E_{o_1} = \frac{1}{2}\sum_{i=1}^{m} e_i^2 \quad \text{m size of training set.}$$

$$= \frac{1}{2}\sum_{i=1}^{m} (d_{o_1} - y_{o_1})^2$$

$$E_{o_2} = \frac{1}{2}\sum_{i=1}^{m} e_i^2 = \frac{1}{2}\sum_{i=1}^{M} (d_{o_2} - y_{o_2})^2$$

20/9/22

• At o/p layer

o/p layer neuron j

$$e_j = d_j - y_j \rightarrow ①$$

$$y_j = \varphi(v_j)$$

$$y_j = \text{sigmoid}(v_j) = \frac{1}{1+e^{-v_j}} \rightarrow ② \text{ (If sigmoid act function is used.)}$$

$$v_j = \sum_{i=1}^{M} x_j w_{ij} + b_0 \rightarrow ③$$

$$E(w) = \frac{1}{2}\sum_{i=1}^{m} e_i^2 \rightarrow (4)$$

$$= \cancel{\frac{1}{2}\sum_{i=1}^{m}(d_i - y_i)}^2$$

$$= \frac{1}{2}\sum_{i=1}^{m}(d_i - y_i)^2$$

• **Update rule:**

as per gradient descent

$w_{new} = w_{old} - \eta g(n)$

(or)

$w(n+1) = w(n) - \eta g(n)$

$g(n)$ is grad. of cost function

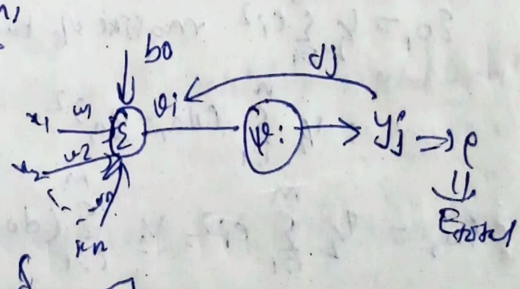
• A neural n/w with k neurons in o/p layer:

$$E_{Tot}(w) = \sum_{i=1}^{k} E_i(w)$$

$$= E_1(w) + E_2(w) + \cdots + E_k(w) \rightarrow (5)$$

• **gradient of cost function:**

$$g(n) = \frac{\partial E_{total}}{\partial w_{ji}}$$



$$\frac{\partial E_{total}}{\partial w_{ji}} = \frac{\partial E_{total}}{\partial e_j} \times \frac{\partial e_j}{\partial y_j} \times \frac{\partial y_j}{\partial v_j} \times \frac{\partial v_j}{\partial w_{ji}}$$

(by applying the chain rule)

$$\frac{\partial E_{total}}{\partial e_j} = \frac{\partial}{\partial e_j}\left(E_1(w) + E_2(w) + \cdots + E_k(w)\right)$$

$$= \frac{\partial}{\partial e_j}\left(\frac{1}{2}\sum_{i=1}^{m} e_i^2 + \frac{1}{2}\sum_{i=1}^{m} e_i^2 + \cdots + \frac{1}{2}\sum_{i=1}^{m} e_i^2 + \cdots + \frac{1}{2}\sum_{i=1}^{m} e_k^2\right)$$

$$= e_j$$

$$\frac{\partial e_j}{\partial y_j} = \frac{\partial}{\partial y_j}(d_j - y_j) = -1$$

$$\frac{\partial y_j}{\partial v_j} = \frac{\partial}{\partial v_j}\left(\frac{1}{1+e^{-v_j}}\right) = \frac{(1+e^{-v_j})(0) - 1(e^{-v_j})}{(1+e^{-v_j})^2}$$

$$= \frac{-e^{-v_j}}{(1+e^{-v_j})^2}$$

$$= y_j(1-y_j)$$

$$\frac{\partial v_j}{\partial w_{ji}} = \frac{\partial}{\partial w_{ji}}\left(\sum_{i=1}^{n} w_{ji} x_i + b_0\right)$$

$$= x_i$$

$$g(n) = \frac{\partial E_{total}}{\partial w_{ji}} = \overbrace{e_j \times -1 \times y_j(1-y_j)}^{\delta} \times x_i$$

$$= \overbrace{-e_j y_j (1-y_j)}^{\delta} \times x_i$$

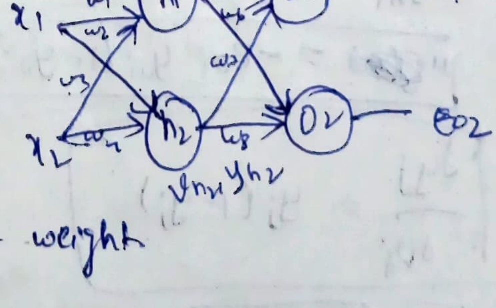
$$w(new) = w_{old} - \eta g(n)$$

$$\boxed{w_{new} = w_{old} + \eta y(1-y) e_j x_i} \rightarrow (6)$$

21/9/22

- **At hidden layer:**

- If there is an error in i/p layer then the o/p will also be effected.



g(n) = gradient of cost function wrt weight

$E_{total} = \{$ error caused by hidden layer neuron i at o/p layer neuron j

$$\boxed{g(n) = \frac{\partial E_{total}}{\partial w_{ji}}} \rightarrow (7)$$

$x \to (h_1) \xrightarrow{v_j} \boxed{\varphi} \to y_j \to E_{total}$

$\boxed{E_{total} \to y_j \to v_j \to w_{ji}}$

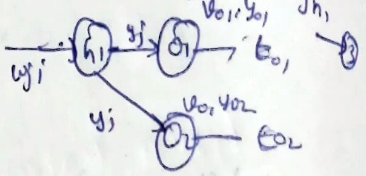
$$\frac{\partial E_{total}}{\partial w_{ji}} = \frac{\partial E_{total}}{\partial y_j} \times \frac{\partial y_j}{\partial v_j} \times \frac{\partial v_j}{\partial w_{ji}} \to (2)$$

$\cancel{\left(e_i \times -1 \times y_j(1-y_j) \right)}$

E_{oi}^2

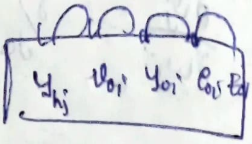
$$\frac{\partial E_{total}}{\partial y_j} = \frac{\partial E_{o1}}{\partial y_j} + \frac{\partial E_{o2}}{\partial y_j} + \dots + \frac{\partial E_{ok}}{\partial y_j}$$

$$= \frac{\partial E_{o1}}{\partial y_{h_1}} + \frac{\partial E_{o2}}{\partial y_{h_1}} + \dots + \frac{\partial E_{ok}}{\partial y_{h_1}} \to (3)$$



$$\boxed{\frac{\partial E_{o1}}{\partial y_{h_1}} = \frac{\partial E_{oi}}{\partial e_{oi}} \times \frac{\partial e_{oi}}{\partial y_{oi}} \times \frac{\partial y_{oi}}{\partial v_{oi}} \times \frac{\partial v_{oi}}{\partial y_j}} \to (4)$$

$\boxed{E_{oi} = \frac{1}{2} \sum_{i=1}^{n} e_{oi}^2}$



$\frac{\partial E_{oi}}{\partial y_{h_1}} = \cancel{\frac{\partial E_{oi}}{\partial e_{oi}}} = \cancel{\frac{\partial e}{\partial e_{oi}}}$

$$\frac{\partial E_{oi}}{\partial e_{oi}} = e_{oi} \qquad e_{oi} = d_j - y_{oi}$$

$$\frac{\partial e_{oi}}{\partial y_{oi}} = -1 \qquad y_{oi} = \varphi(v_{oi})$$

$$\frac{\partial y_{oi}}{\partial v_{oi}} = y_{oi}(1-y_{oi}) \qquad \cancel{v_{oi} = \sum_{i=1}^{n} w_{ji} y_j + b_{oi}}$$

$$\frac{\partial v_{oi}}{\partial y_j} = w_{ji} \qquad v_{oi} = \sum_{i=1}^{n} w_{ji} y_j + b_{oi}$$

$$\frac{\partial E_{oi}}{\partial y_{h_1}} = e_{oi} \times -1 \times y_{oi}(1-y_{oi}) \times w_{ji}$$

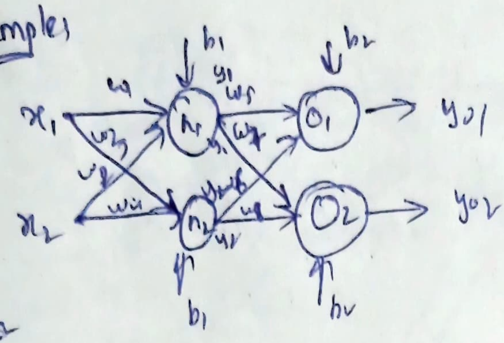
$\boxed{\text{u} \; \Rightarrow -e_{oi}\, y_{oi}(1-y_{oi})\, w_{ji}}$

$$\boxed{\frac{\partial y_j}{\partial v_j} = y_j(1-y_j)}$$

$$\boxed{\frac{\partial v_j}{\partial w_{ji}} = x_i}$$

$$\boxed{\frac{\partial E_{total}}{\partial w_{ji}} = -e_{oi}\, y_{oi}(1-y_{oi})\, w_{ji} \cdot y_j(1-y_j)\, x_i}$$

Examples

Q: 

O_1 ⇒

$v_{O1} = y_{h_1} \times w_5 + y_{h_2} \times w_6 + b_2$

$y_{O1} = \text{Sigmoid}(v_{O1}) = \dfrac{1}{1+e^{-v_{O1}}}$

$e_{O1} = d_{O1} - y_{O1}$

$E_{O1} = \frac{1}{2}\sum_{i=1}^{n} e_{O1}^2(i)$

$= \frac{1}{2}\sum_{i=1}^{n} (d_{O1}(i) - y_{O1}(i))^2$

$w_5(new) = w_5(old) - \eta g(n)$

$g(n) = -e_{O1}\, y_{O1}(1-y_{O1})\, y_{h1}$

$\boxed{w_5(new) = w_5(old) + \eta\, e_{O1}\, y_{O1}(1-y_{O1})\, y_{h1}}$

Update Rule for w_6:

$w_6(new) = w_6(old) - \eta g(n)$

$\boxed{\dfrac{\partial v_{O1}}{\partial w_6} = y_{h2}}$

$\boxed{g(n) = -e_{O1}\, y_{O1}(1-y_{O1})\, y_{h2}}$

$\boxed{w_6(new) = w_6(old) + \eta\, e_{O1}\, y_{O1}(1-y_{O1})\, y_{h2}}$

O2:-

$$v_{o2} = y_{h2} \times w_8 + y_{h1} \times w_7 + b_2$$

$$y_{o2} = \text{sigmoid}(v_{o2}) = \frac{1}{1+e^{-v_{o2}}}$$

$$e_{o2} = d_{o2} - y_{o2}$$

$$E_{o2} = \frac{1}{2}\sum_{i=1}^{n} e_{o2}^2(i)$$

$$= \frac{1}{2}\sum_{i=1}^{n} (d_{o2}(i) - y_{o2}(i))^2$$

$$w_7(new) = w_7(old) - \eta g(n)$$

$w_7(new) = 0.459$

$$g(n) = -e_{o2}\, y_{o2}(1-y_{o2})\, y_{h1}$$

$$\boxed{w_7(new) = w_7(old) + \eta\, e_{o2}\, y_{o2}(1-y_{o2})\, y_{h1}}$$

$$w_8(new) = w_8(old) - \eta g(n)$$

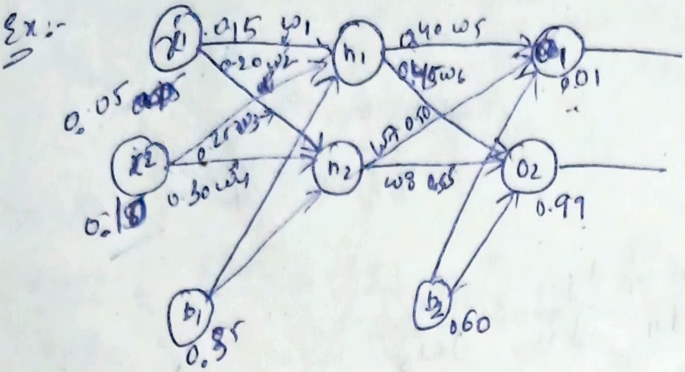
$$g(n) = -e_{o2}\, y_{o2}(1-y_{o2})\, y_{h2}$$

$$\boxed{w_8(new) = w_8(old) + \eta\, e_{o2}\, y_{o2}(1-y_{o2})\, y_{h2}}$$

$w_8(new) = 0.5117$

23/9/22

Ex:-



Iteration 1:

• Forward pass:

At h1:

$v_{h_1} = w_1 x_1 + w_3 x_2 + b_1$

$= 0.05(0.15) + 0.25 \times 0.10 + 0.35$

$\boxed{v_{h_1} = 0.3825}$

$y_{h_1} = \varphi(v_{h_1}) = \frac{1}{1+e^{-0.3825}}$

$\boxed{y_{h_1} = 0.5944}$

At h2:

$v_{h_2} = x_1 w_2 + x_2 w_4 + b_1$

$= 0.05(0.20) + 0.10(0.30) + 0.35$

$\boxed{v_{h_2} = 0.39}$

$y_{h_2} = \frac{1}{1+e^{-0.39}} = 0.5962$

$\boxed{y_{h_2} = 0.5912}$

At O1:

$v_{o_1} = y_{h_1} \times w_5 + y_{h_2} \times w_7 + b_2$

$= 0.5944(0.40) + 0.5962(0.50) + 0.60$

$= 0.2377 + 0.2981 + 0.60$

$= 1.135$

$y_{o_1} = \varphi(v_{o_1}) = \frac{1}{1+e^{-1.135}}$

$\boxed{y_{o_1} = 0.7569}$

At O2:

$v_{o_2} = y_{h_1} \times w_6 + y_{h_2} \times w_8 + b_2$

$= 0.5944(0.45) + 0.5962(0.55) + 0.60$

$= 0.2674 + 0.3279 + 0.60$

$\boxed{v_{o_2} = 1.1965}$

$y_{o_2} = \frac{1}{1+e^{-1.1965}} = 0.7679$

$\boxed{y_{o_2} = 0.7679}$

	d	y
O1	0.01	0.756 ×
O2	0.99	0.767 ×

So it has error

• Backward Pass:

$w_5(\text{new}) = w_5(\text{old}) - \eta\, g(n)$

$$E_{total} = E_{01} + E_{02}$$
$$= \frac{1}{2}\sum_{i=1}^{n} e_{01}^2 + \frac{1}{2}\sum_{i=1}^{n} e_{02}^2$$

$g(n) = \frac{\partial E_{Total}}{\partial w_5}$

$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial e_{01}} \times \frac{\partial e_{01}}{\partial y_{01}} \times \frac{\partial y_{01}}{\partial v_{01}} \times \frac{\partial v_{01}}{\partial w_5}$$

$$= e_{01} \times -1 \times y_{01}(1-y_{01}) \times y_{h_1}$$

$$= -e_{01} \times y_{01}(1-y_{01}) \times y_{h_1}$$

$$= -(d_{01}-y_{01}) \cdot (y_{01}(1-y_{01})) \cdot y_{h_1}$$

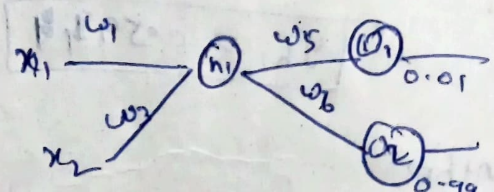
$$= -(0.01 - 0.7569)(0.7569(1-0.7569))\, 0.5944$$

$$\boxed{g(n) = 0.0817}$$

$\eta = 0.5$

$w_5(\text{new}) = 0.40 - 0.5(0.0817)$

$$\boxed{w_5(\text{new}) = 0.3591}$$



At hidden layer

update w_1

$w_1(\text{new}) = w_1(\text{old}) - \eta\, g(n)$

$g(n) = \frac{\partial E_{total}}{\partial w_1}$

$E_{tot} = E_{01} + E_{02}$

①	②	③
E_{total}	E_{01}	E_{02}
↓	↓	↓
y_{h_1}	e_{01}	e_{02}
↓	↓	↓
v_{h_1}	y_{01}	y_{02}
↓	↓	↓
w_1	v_{01}	v_{02}
	↓	↓
	y_{h_1}	y_{h_1}

$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial E_{total}}{\partial y_{h_1}} \times \frac{\partial y_{h_1}}{\partial v_{h_1}} \times \frac{\partial v_{h_1}}{\partial w_1} \rightarrow ①$$

$$\frac{\partial E_{total}}{\partial y_{h_1}} = \frac{\partial E_{01}}{\partial y_{h_1}} + \frac{\partial E_{02}}{\partial y_{h_1}} \rightarrow ②$$

$$\frac{\partial E_{01}}{\partial y_{h_1}} = \frac{\partial E_{01}}{\partial e_{01}} \times \frac{\partial e_{01}}{\partial y_{01}} \times \frac{\partial y_{01}}{\partial v_{01}} \times \frac{\partial v_{01}}{\partial y_{h_1}} \rightarrow ③$$

30/9/22

$$\frac{\partial E_{o2}}{\partial y_{h_1}} = \frac{\partial E_{o2}}{\partial e_{o2}} \times \frac{\partial e_{o2}}{\partial y_{o2}} \times \frac{\partial y_{o2}}{\partial \vartheta_{o2}} \times \frac{\partial \vartheta_{o2}}{\partial y_{h_1}} \quad -(9)$$

from (8)

$$\frac{\partial E_{o1}}{\partial y_{h_1}} = \frac{\partial E_{o1}}{\partial e_{o1}} \times \frac{\partial e_{o1}}{\partial y_{o1}} \times \frac{\partial y_{o1}}{\partial \vartheta_{o1}} \times \frac{\partial \vartheta_{o1}}{\partial y_{h_1}}$$

$\vartheta_{o1} = y_{h_1} \times w_5 + y_{h_2} \times w_7 + b_2$

$= e_{o1} \times (-1) \times y_{o1}(1-y_{o1}) \times w_5$

$\frac{\partial \vartheta_{o1}}{\partial y_{h_1}} = w_5$

$$\boxed{\frac{\partial E_{o1}}{\partial y_{h_1}} = -e_{o1}\, y_o(1-y_o)\, w_5}$$

$= -(d_{o1} - y_{o1}) \times y_{o1} \times (1-y_{o1}) \times w_5$

$= -(0.01 - 0.7569) \times 0.7569(1-0.7569) \times 0.40$

$= 0.7469 \times 0.7569(0.2431) \times 0.40$

$= 0.0549$

from (9),

$$\frac{\partial E_{o2}}{\partial y_{h_1}} = \frac{\partial E_{o2}}{\partial e_{o2}} \times \frac{\partial e_{o2}}{\partial y_{o2}} \times \frac{\partial y_{o2}}{\partial \vartheta_{o2}} \times \frac{\partial \vartheta_{o2}}{\partial y_{h_1}}$$

$= e_{o2} \times (-1) \times y_{o2}(1-y_{o2}) \times w_6$

$$\boxed{\frac{\partial E_{o2}}{\partial y_{h_1}} = -e_{o2}\,(y_{o2}(1-y_{o2}))\, w_6}$$

$= -(0.99 - 0.7679) \times (0.7679)(1-0.7679) \times 0.45$

$= -0.0178$

$$\boxed{\frac{\partial E_{total}}{\partial y_{h_1}} = -e_{o1}\, y_{o1}(1-y_{o1})\, w_5 - e_{o2}\, y_{o2}(1-y_{o2})\, w_6} \rightarrow (1) \text{ update (1).}$$

$\therefore \frac{\partial E_{total}}{\partial y_{h_1}} = 0.0549 - 0.0178$

$= 0.0371$

$\frac{\partial E_{total}}{\partial w_1} = 0.0371 \times y_{h_1}(1-y_{h_1}) \times x_1$

$= 0.0371 \times 0.5944(1-0.5944)\, 0.05$

$\eta = 0.5$ $\boxed{g(n) = 0.00044}$

$w_1(new) = 0.15 - 0.5(0.00044)$

$$\boxed{w_1(new) = 0.1497}$$

update w_3

$$g(n) = \frac{\partial E_{total}}{\partial w_3} = \frac{\partial E_{total}}{\partial y_{h1}} \times \frac{\partial y_{h1}}{\partial v_{h1}} \times \frac{\partial v_{h1}}{\partial w_3} \rightarrow (1)$$

$$\frac{\partial E_{total}}{\partial y_{h1}} = \frac{\partial E_{o1}}{\partial y_{h1}} + \frac{\partial E_{o2}}{\partial y_{h1}} \rightarrow (2)$$

$$\frac{\partial E_{o1}}{\partial y_{h1}} = \frac{\partial E_{o1}}{\partial e_{o1}} \times \frac{\partial e_{o1}}{\partial y_{o1}} \times \frac{\partial y_{o1}}{\partial v_{o1}} \times \frac{\partial v_{o1}}{\partial y_{h1}} \rightarrow (3)$$

$$\frac{\partial E_{o2}}{\partial y_{h1}} = \frac{\partial E_{o2}}{\partial e_{o2}} \times \frac{\partial e_{o2}}{\partial y_{o2}} \times \frac{\partial y_{o2}}{\partial v_{o2}} \times \frac{\partial v_{o2}}{\partial y_{h1}} \rightarrow (4)$$

from (3),

$$\frac{\partial E_{o1}}{\partial y_{h1}} = e_{o1} \times (-1) \times y_{o1}(1-y_{o1}) \times w_5$$

$$= 0.0549$$

(4), $$\frac{\partial E_{o2}}{\partial y_2} = -0.0178$$

(2), $$\frac{\partial E_{total}}{\partial y_{h1}} = 0.0549 - 0.0178$$

$$= 0.00371$$

(1), $$\frac{\partial E_{total}}{\partial w_3} = 0.0371 \times y_{h1}(1-y_{h1}) \times x_2$$

$$= 0.0371 \times (0.5944)(1-0.5944) \times 0.10.$$

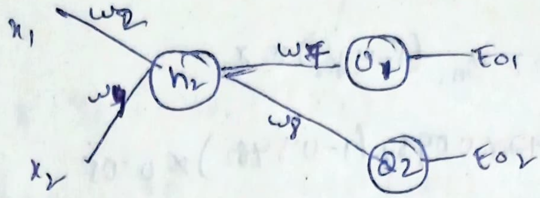
$$\boxed{g(n) = 0.00089}$$

$$w_3(new) = w_3(old) - \eta g(n)$$

$$= 0.25 - 0.5(0.00089)$$

$$= 0.249$$

At Hidden layer 1, neuron 2:



updation for W_2

$$w_2(new) = w_2(old) - \eta g(n)$$

$$g(n) = \frac{\partial E_{total}}{\partial w_2}$$

$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial E_{total}}{\partial y_{h2}} \times \frac{\partial y_{h2}}{\partial v_{h2}} \times \frac{\partial v_{h2}}{\partial w_2} \to (1)$$

$$\frac{\partial E_{total}}{\partial y_{h2}} = \frac{\partial E_{o1}}{\partial y_{h2}} + \frac{\partial E_{o2}}{\partial y_{h2}} \to (2)$$

$$\frac{\partial E_{o1}}{\partial y_{h2}} = \frac{\partial E_{o1}}{\partial e_{o1}} \times \frac{\partial e_{o1}}{\partial y_{o1}} \times \frac{\partial y_{o1}}{\partial v_{o1}} \times \frac{\partial v_{o1}}{\partial y_{h2}} \to (3)$$

$$\frac{\partial E_{o2}}{\partial y_{h2}} = \frac{\partial E_{o2}}{\partial e_{o2}} \times \frac{\partial e_{o2}}{\partial y_{o2}} \times \frac{\partial y_{o2}}{\partial v_{o2}} \times \frac{\partial v_{o2}}{\partial y_{h2}} \to (4)$$

from (3),

$$\frac{\partial E_{o1}}{\partial y_{h2}} = e_{o1} \times -1 \times y_{o1}(1-y_{o1})\, w_7$$

$$= -(0.01-0.7567)\, 0.7567(1-0.7567)(0.50)$$

$$= 0.0687$$

From (4),

$$\frac{\partial E_{o2}}{\partial y_{h2}} = e_{o2} \times -1 \times y_{o2}(1-y_{o2})\, w_8$$

$$= -(0.99-0.7679)\, 0.7679(1-0.7679)(0.55)$$

$$= -0.02177$$

$$\boxed{\frac{\partial E_{total}}{\partial y_{h2}} = 0.047}$$

$$\frac{\partial E_{total}}{\partial w_2} = 0.047 \times y_{h_2}(1-y_{h_2}) \times x_1$$

$$= 0.047 \times 0.5962(1-0.5962) \times 0.05$$

$$= 0.00056$$

$$w_2(new) = w_2(old) - \eta\, g(n)$$

$$= 0.20 - 0.5(0.00056)$$

$$= \cancel{0.0997} = 0.9997$$

<u>update w_4</u>

$$w_4(new) = w_4(old) - \eta\, g(n)$$

$$g(n) = \frac{\partial E_{total}}{\partial w_4}$$

$$\frac{\partial E_{total}}{\partial w_4} = \frac{\partial E_{total}}{\partial y_{h_2}} \times \frac{\partial y_{h_2}}{\partial v_{h_2}} \times \frac{\partial v_{h_2}}{\partial w_4} \rightarrow (1)$$

$$\frac{\partial E_{total}}{\partial y_{h_2}} = \frac{\partial E_{O1}}{\partial y_{h_2}} + \frac{\partial E_{O2}}{\partial y_{h_2}} \rightarrow (2)$$

$$\frac{\partial E_{O1}}{\partial y_{h_2}} = \frac{\partial E_{O1}}{\partial e_{O1}} \times \frac{\partial e_{O1}}{\partial y_{O1}} \times \frac{\partial y_{O1}}{\partial v_{O1}} \times \frac{\partial v_{O1}}{\partial y_{h_2}} \rightarrow (3)$$

$$\frac{\partial E_{O2}}{\partial y_{h_2}} = \frac{\partial E_{O2}}{\partial e_{O2}} \times \frac{\partial e_{O2}}{\partial y_{O2}} \times \frac{\partial y_{O2}}{\partial v_{O2}} \times \frac{\partial v_{O2}}{\partial y_{h_2}} \rightarrow (4)$$

same as before

$$\boxed{\frac{\partial E_{total}}{\partial y_{h_2}} = 0.047}$$

$\frac{\partial E_{total}}{\partial w_4} = 0.047 \times y_{h_2}(1-y_{h_2}) \cdot (x_2)$

$= 0.047 \times 0.5912(1-0.5912)(0.10)$

$= 0.00011$

$w_4(new) = w_4(old) - \eta g(n)$

$= 0.30 - 0.5(0.00011)$

$= 0.299$

<u>after iteration 1:</u>

$w_1 = 0.1497$ $w_5 = 0.3591$ $x_1 = 0.05$

$w_2 = 0.799$ $w_6 = 0.5617$ $x_2 = 0.10$

$w_3 = 0.249$ $w_7 = 0.459$

$w_4 = 0.299$ $w_8 = 0.5617$

* Pratical & Design issues of Back Propagation algorithm ?

① Rate of learning :-

- specifies a step size taken by learning algorithm
- takes a value in b/w 0-1, $0 \leq \eta \leq 1$.

<u>η is very small</u>

$w(new) = w(old) - \underbrace{\eta}_{\text{learning rate}} g(n)$

↳ slowly converges to global minimum.

- It increases no. of iterations required for converging.
- Algorithm can trapped into local minimum points.

<u>η = very large</u>

- Algorithm can overshoot global minimum point can oscillate

• Solution:

momentum:

- $\Delta w_{ji}(n) = \alpha \Delta w_{ji}(n-1) + \Delta w_{ji}(n)$ ⇒ $\alpha = 0.01$ (or) $\alpha = 0.001$
- provides solution to problems associated with learning rate
- Adds small amt of update from prev iteration controlled by momentum term(α)

• Sequential or Batch mode:

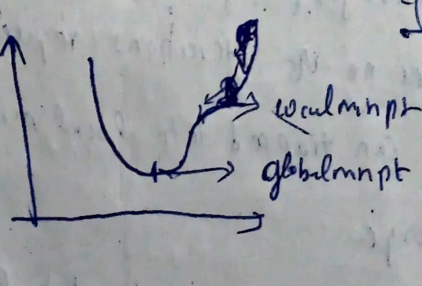
epoch
↓
presentation of complete set of examples or samples

- also called online mode
- update for weight vector computed sequentially by considering only one sample at a time
- It requires less memory space as it slowly converges towards global minima pt.

• Drawback:
 • It is theoretically difficult to determine criteria.

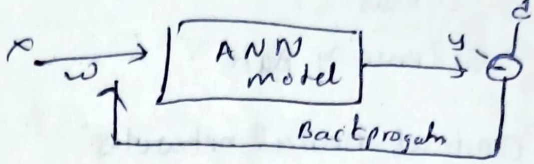
• Design issues in Back-Propagation Algorithm: 11/10/22

• To get rid of local minimum pt we will use momentum to push the object from local min pt.



② Stopping criteria:-

- Used to minimize error produced by ANN model.
- To stop the ANN model, if error ~~the~~ is present, then we update the weights.



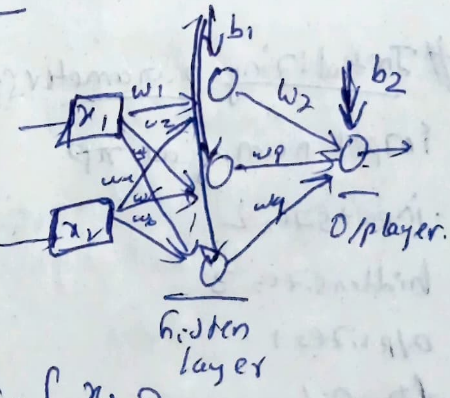
- ~~when~~ we stop the model when the value is approxly equal to (target o/p) the actual o/p.
- when euclidean norm value of the predicted - actual o/p reaches to min value then we can stop.

③ New sample not there in training set:-

It will goes a process called generalization, it try to match with actual o/p and gives the conclusions.

- Implementation of Backpropagation algorithm:-

$$hw = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \end{bmatrix}_{6\times 1} \qquad ho = \begin{bmatrix} w_7 \\ w_8 \\ w_9 \end{bmatrix}_{3\times 1}$$



$$\Rightarrow \begin{bmatrix} w_1 & w_4 \\ w_2 & w_5 \\ w_3 & w_6 \end{bmatrix} \qquad x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$hw \times x$

$$= \begin{bmatrix} w_1 & w_4 \\ w_2 & w_5 \\ w_3 & w_6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$= \begin{bmatrix} w_1 x_1 + x_2 w_4 \\ w_2 x_1 + w_5 x_2 \\ w_3 x_1 + w_6 x_2 \end{bmatrix}$$

- Implementing Backpropagation:

① Initialize parameters

(1) weights.

(2) Bias.

(3) learning rate

② Construct neural networks

→ no. of inputs.

→ no. of hiddenlayers.

→ no. of neurons

→ no. of neurons in o/p. layer.

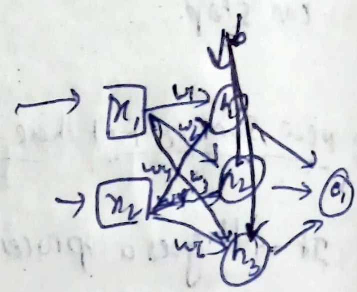
③ compute error

$(e = d - y)$

Updateweights,

$w_1(new) = w_1(old) - \eta\, g_h(n)$

$w_2(new) = w_2(old) - \eta\, g_o(n)$



$g_o(n) = \frac{\partial E}{\partial w_2}$

$= \frac{\partial E}{\partial e_o} \times \frac{\partial e_o}{\partial y_o} \times \frac{\partial y_o}{\partial v_o} \times \frac{\partial v_o}{\partial w_2}$

$\boxed{g_o(n) = \underbrace{e_o \times -1 \times y_o(1-y_o)}_{\delta}\, y_h}$

$g_h(n) = \frac{\partial E}{\partial w_1} = \frac{\partial E}{\partial y_h} \times \frac{\partial y_h}{\partial v_h} \times \frac{\partial v_h}{\partial w_1}$

$\frac{\partial E}{\partial y_h} = \frac{\partial E}{\partial e_o} \times \frac{\partial e_o}{\partial y_o} \times \frac{\partial y_o}{\partial v_o} \times \frac{\partial v_o}{\partial y_h}$

$= \delta \times w_2$

Initializing parameters:

```
import numpy as np.
inputsize=2
hiddensize=3
o/psize=1
lr=0.1
w1=np.random.randn(inputsize, hiddensize)*0.01
w2=np.random.randn(hiddensize, o/psize)*0.01
b1=np.random.randn(hiddensize,1)*0.01
b2=np.random.randn(o/psize,1)*0.01
```

defining activation functions

```
def sigm(v):
    y=1/(1+np.exp(-v))
```

Ex:	hours Studied	hours enjoyed	Result
	7	2	80
	2	6	50
	9	1	95

```
    return y.

def derivation(y):
    return(yx(1-y))

def meansqerror(yp,y):
    MSE = ((yp-y)**2).sum()/2
    return E

#define a dataset
    x= np.array([[7,2],[2,6],[9,1]])

    y=np.array([[80],[50],[95]])

#normalize
    x=x/np.amax(x,axis=0)

    y=y/100

#create a neural N/w & train:

def train(x,y):
    global w1,w2,b1,b2,lr.

#forward phase:
    vh= np.dot(x,w1)+b1 #or x @ w1 can also be used
    yh = sigm(vh)
    vo= np.dot(yh,w2)+b2
    yo= sigm(vo)
    return yo.

train(x,y)
```

back propagation phase

→

$e1 = y - y_0$

$delta_0 = e1 \times y_0 \times (1 - y_0) \times$ _

$e2 = delta\_0 @ w2.T$

$delta\_h = e2 \times yh \times (1 - yh)$

$go = yh.T @ delta\_0$

$gh = x.T @ delta\_h$

$w1 = w1 - \eta \times gh$ $(\eta = lr)$

$w2 = w2 - \eta \times go$

$b1 = b1 - \eta \times delta\_h$

$b2 = b2 - \eta \times delta\_0$

$$\frac{\partial E}{\partial w_2} = \frac{\partial E}{\partial e_0} \times \frac{\partial e_0}{\partial y_0} \times \frac{\partial y_0}{\partial v_0} \times \frac{\partial v_0}{\partial w_2}$$

$$= e \times y_0(1 - y_0) \times$$

$$\frac{\partial E}{\partial y_h} = \underbrace{\frac{\partial E}{\partial e_0} \times \frac{\partial e_0}{\partial y_0} \times \frac{\partial y_0}{\partial v_0}}_{delta\_0} \times \frac{\partial v_0}{\partial y_{h_1}}$$

$= e_2$

$\frac{\partial y_0}{\partial w_1}$ =

```
→ for i in range(2000):
      train(x, y)

→ def forwardtest(x, y):
      global w1, w2, b1, b2,
      vh = np.dot(x, w1) + b1
      yh = sigm(vh)
      vo = np.dot(yh, w2) + b2
      yo = sigm(vo)
      return yo.

→ forward test(x, y)
```