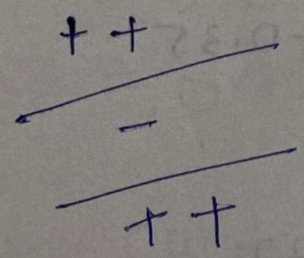
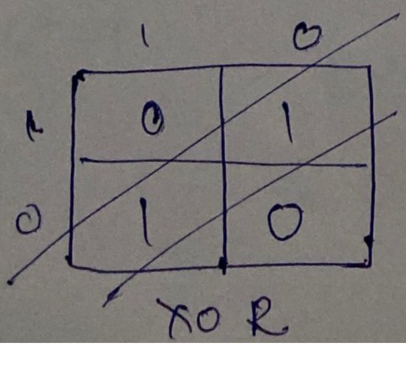
unit 4

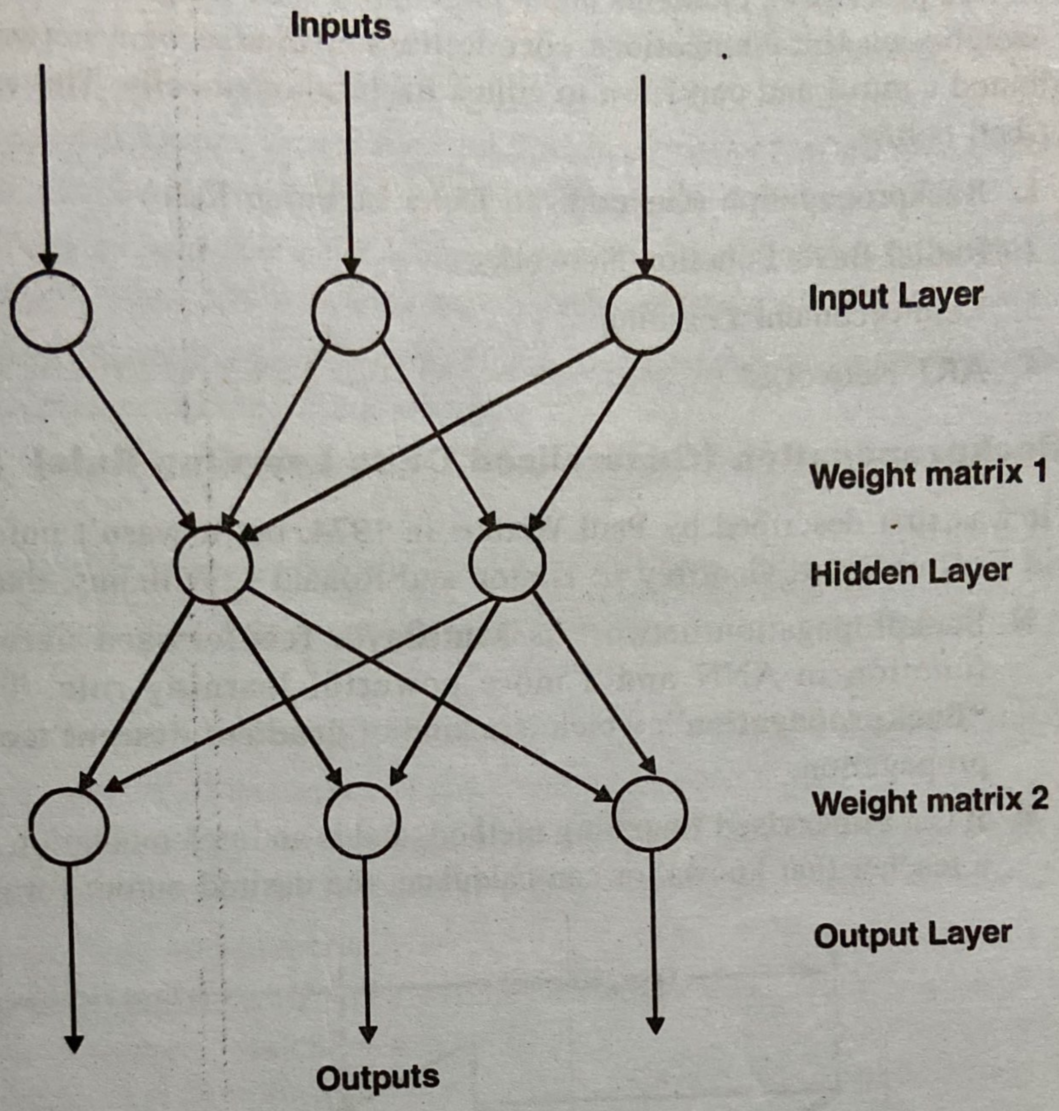
Multilayer Perceptron

Ans MLP is a network of simple neurons called perceptrons. A multilayer perceptron is a feed forward neurel network. MLP consists of at least three layers of nodes: an input layer, a hidden layer and an output layer. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. MLP uses supervised learning. It uses backpropagation for training. Its multiple layers & non linear activation distinguish MLP from a linear ~~activation~~ perceptron. It can distinguish data that is not linearly separable.



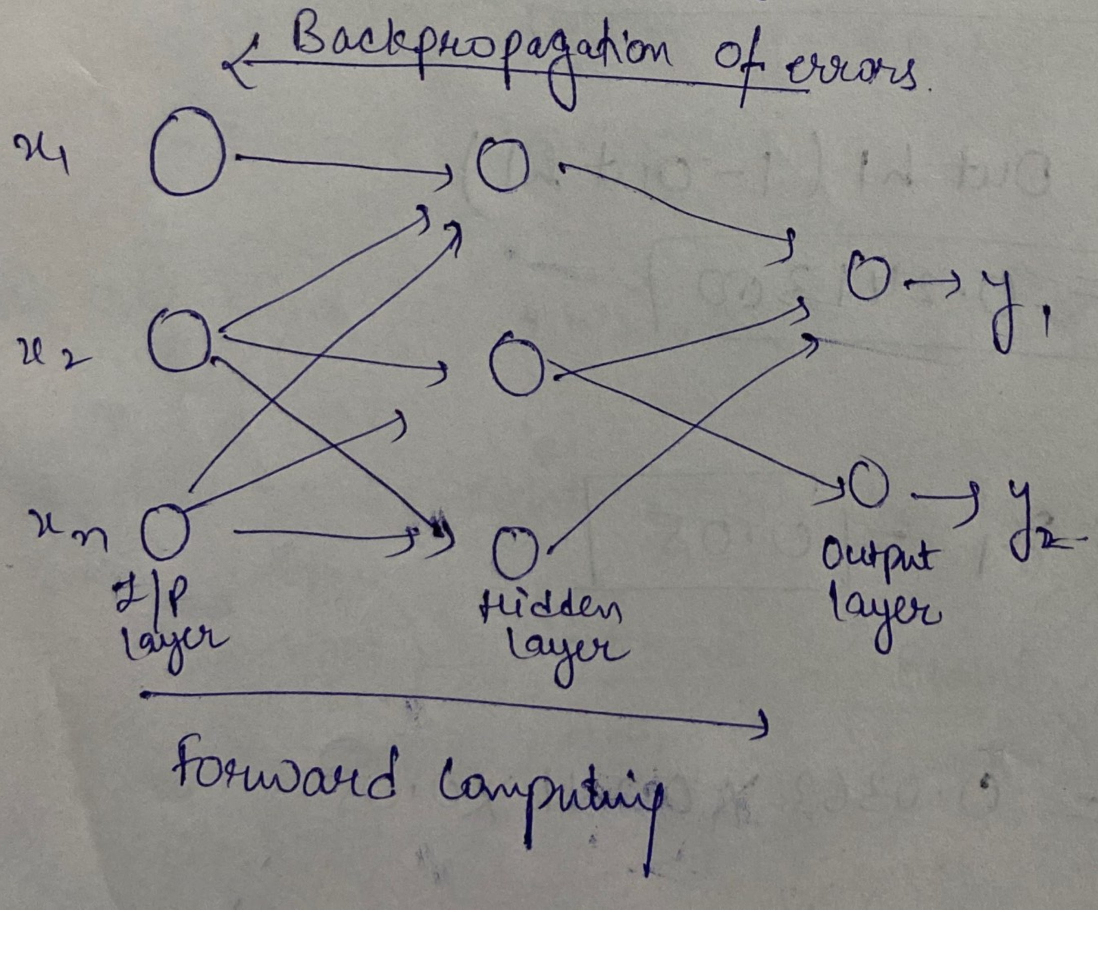


Multilayer-Perceptron Characteristics

<table>
<tr><td>Sample Structure</td><td>
</td></tr>
<tr><td>Type</td><td>Feed-forward</td></tr>
<tr><td>Neuron Layers</td><td>1 input layer
1 or more hidden layers
1 output layer</td></tr>
<tr><td>Input value types</td><td>Binary</td></tr>
<tr><td>Activation Functions</td><td>Hard limiter/sigmoid</td></tr>
<tr><td>Learning Method</td><td>Supervised</td></tr>
<tr><td>Learning algorithm</td><td>Delta learning rule
Back-propagation (mostly used)</td></tr>
</table>

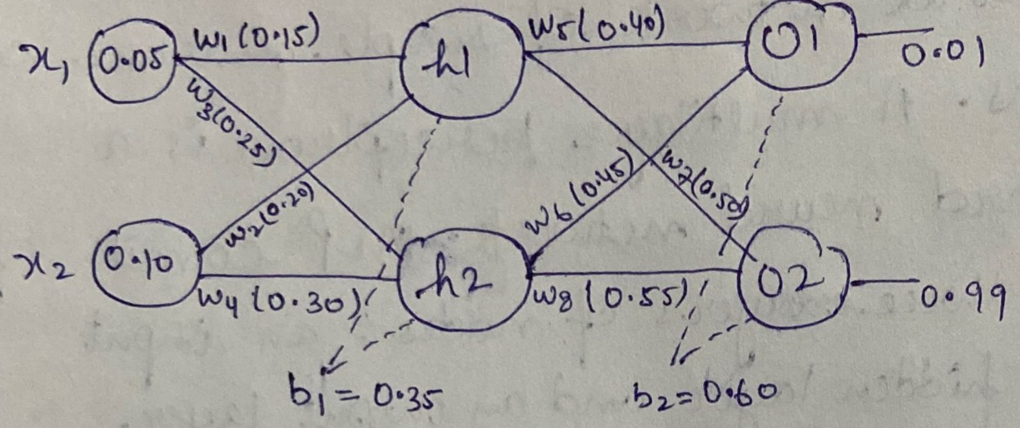
Backpropagation

Backpropagation is just a way of propagating the total loss back into the neural network to know how much of the loss every node is responsible for and subsequently updating the weights in such a way that minimizes the loss by giving the nodes with higher error rates lower weights & vice versa.



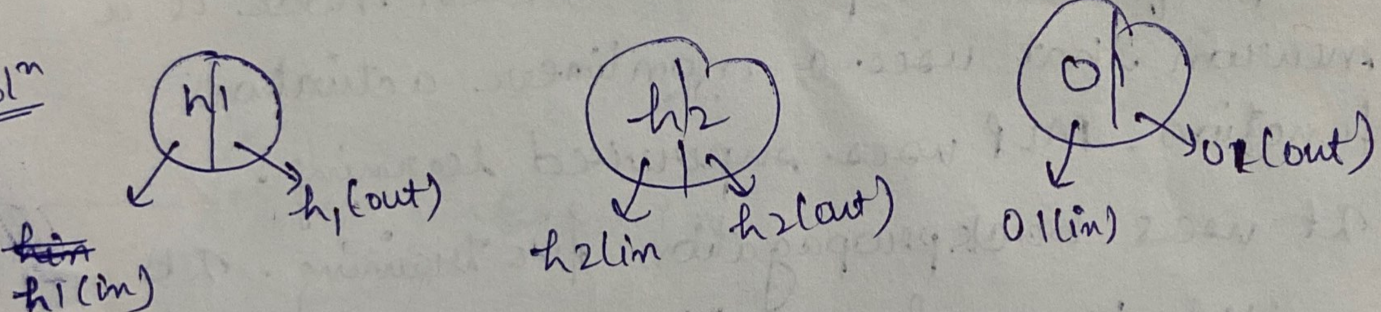
Backpropogation Numerical

Ques



and $\alpha = 0.6$

Sol^n



- Forward Pass / Forward Computing

$$h1(in) = w_1 x_1 + w_2 x_2 + b_1$$

$$= 0.15 \times 0.05 + 0.2 \times 0.1 + 0.35$$

$$= 0.377$$

$$h1(out) = \frac{1}{1 + e^{-h1(in)}} = 0.5932$$

Similarly $h_2(out) = 0.5968$

$O_1(in) = w_5 \times h_1(out) + w_6 \times h_2(out) + b_2$

$= 0.4 \times 0.593 + 0.45 \times 0.596 + 0.6$

$= 1.105$

$O_1(out) = \frac{1}{1+e^{-O_1(in)}} = \underline{\underline{0.7513}}$

$\boxed{O_2(out) = 0.7729}$

Find error

Check whether $O_1(out)$ = target value (i.e $O_1(out) = 0.7513$ & $t_1 = 0.01$) $O_1(out) \neq t_1$. It means there is some error.

~~Stm~~ $E_{Total} = \sum \frac{1}{2}(target - O)^2$ $O \rightarrow$ actual o/p

$= \frac{1}{2}(0.01 - 0.7513)^2$

$E_{O1} = 0.274$

Similarly check for $O_2(out)$ = target value

~~$E_{O2} = \sum \frac{1}{2}(target)$~~.

$0.7729 \neq 0.99$

$E_{O2} = \sum \frac{1}{2}(target - O)^2 = \frac{1}{2}(0.7729 - 0.99)^2$

$E_{O2} = 0.0235$

Total Errors = $E_{O_1} + E_{O_2}$

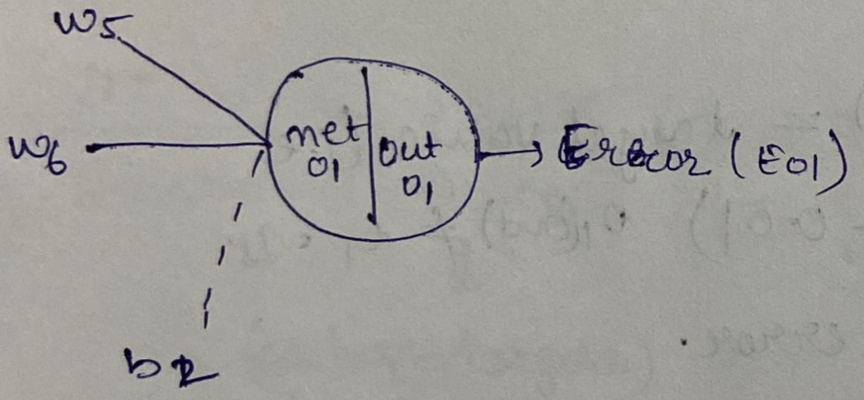
$= 0.274 + 0.0235$

$E_{total} = 0.2983$

Now to minimize this error we backpropogate the network & adjust the weights.

→ First adjust weight (w_5).

$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial out_{o1}} \times \frac{\partial out_{o1}}{\partial net_{o1}} \times \frac{\partial net_{o1}}{\partial w_5}$$



$$\frac{\partial E_{total}}{\partial out_{o1}} = out_{o1} - Target_{o1}$$

$= 0.751365 - 0.01$

$= 0.741365$

$$\frac{\partial out_{o1}}{\partial net_{o1}} = out_{o1}(1 - out_{o1})$$

$= 0.751365(1 - 0.751365)$

$= 0.18.6815602$

$$\frac{\partial net_{o1}}{\partial w_5} = out_{h1} = 0.59326992$$

$$\frac{\partial E_{total}}{\partial w_5} = 0.7413 \times 0.186 \times 0.593$$

$$= 0.08216$$

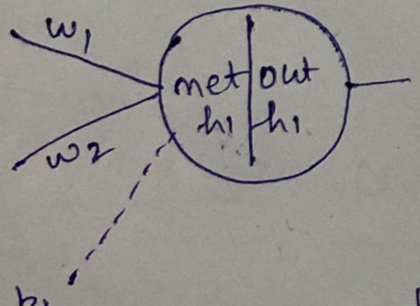
$$\text{update weight } (w_5^*) = w_5(\text{old}) - \alpha \times \frac{\partial E_{total}}{\partial w_5}$$

$$= 0.4 - 0.6 \times 0.08216$$

$$= 0.3506$$

Similarly find w_6^*, w_7^*, w_8^*.

→ update ~~erro~~ weights at hidden layer.



$$\frac{\partial E_{total}}{\partial w_1} = \underbrace{\frac{\partial E_{total}}{\partial out_{h1}}} \times \frac{\partial out_{h1}}{\partial net_{h1}} \times \frac{\partial net_{h1}}{\partial w_1}$$

$$\frac{\partial E_{total}}{\partial out_{h1}} = \underbrace{\frac{\partial E_{o1}}{\partial out_{h1}}} + \underbrace{\frac{\partial E_{o2}}{\partial out_{h1}}}$$

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial E_{o1}}{\partial net_{o1}} \times \frac{\partial net_{o1}}{\partial out_{h1}} \qquad \frac{\partial E_{o2}}{\partial net_{o2}} \times \frac{\partial net_{o2}}{\partial out_{h1}}$$

$$\frac{\partial E_{o1}}{\partial net_{o1}} = \frac{\partial E_{o1}}{\partial out_{o1}} \times \frac{\partial out_{o1}}{\partial net_{o1}}$$

$$\frac{\partial E_{o1}}{\partial out_{o1}} = (out_{o1} - target_{o1})$$

$$\frac{\partial out_{o1}}{\partial net_{o1}} = out_{o1}(1 - out_{o1})$$

$$= 0.13849$$

$$\frac{\partial net_{o1}}{\partial out_{h1}} = w_5 = 0.4$$

$$\boxed{\frac{\partial E_{o1}}{\partial out_{h1}} = 0.13849 \times 0.4 = 0.05539}$$

$$\frac{\partial E_{o2}}{\partial net_{o2}} = \frac{\partial E_{o2}}{\partial out_{o2}} \times \frac{\partial out_{o2}}{\partial net_{o2}}$$

$\frac{\partial E_{o2}}{\partial out_{o2}}$: $(out_{o2} - target_{o2})$

$\frac{\partial out_{o2}}{\partial net_{o2}}$: w_7

0.50

$$\cancel{= -0.0380992}$$

$$\cancel{= -0.01904}$$

$$\frac{\partial E_{o2}}{\partial out_{o2}} = \frac{\partial E_{o2}}{\partial net_{o2}} \times \frac{\partial out_{o2}}{\partial net_{o2}}$$

$out_{o2} - target_{o2}$ $\qquad$ $out_{o2}(1 - out_{o2})$

$= -0.2170$ $\qquad$ 0.1755

$$= -0.0380$$

$$\frac{\partial E_{o2}}{\partial net_{o2}} = -0.380 \times 0.50$$

$$= 0.01949$$

$$\frac{\partial E_{total}}{\partial out_{h1}} = \frac{\partial E_{o1}}{\partial out_{h1}} + \frac{\partial E_{o2}}{\partial out_{h1}}$$

$$= 0.05539 + (-0.019049)$$

$$\boxed{= 0.036350306}$$

$$\frac{\partial out_{h1}}{\partial net_{h1}} = out_{h1}(1 - out_{h1})$$

$$\boxed{= 0.241300}$$

$$\frac{\partial net_{h1}}{\partial w_1} = x_1 = \boxed{0.05}$$

$$\frac{\partial E_{total}}{\partial w_1} = 0.0363 \times 0.241 \times 0.05$$

$$= 0.0004385$$

$$\text{update weight } (w_1^*) = w_1 - \alpha \times \frac{\partial E_{total}}{\partial w_1}$$

$$= 0.15 - 0.6 \times 0.0004385$$

$$= 0.1497$$

Similarly find w_2^*, w_3^*, w_4^*

Scanned with CamScanner