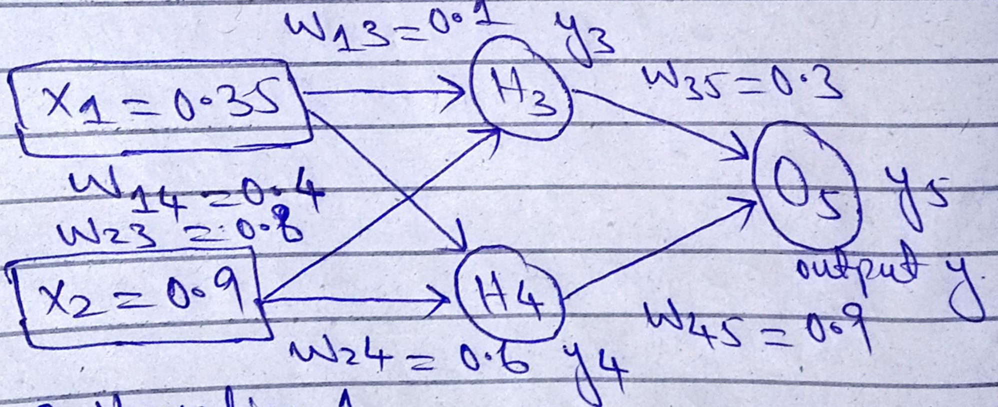
1

BACK PROPAGATION ALGORITHM

MULTI LAYER PERCEPTRON NETWORK

Example

Assume that the neurons have a sigmoid activation function, perform a forward pass and a backward pass on the network. Assume that the actual output of y is 0.5. and learning rate is 1. Perform another forward pass



Iteration 1

Forward Pass :- Compute output for y_3, y_4 and y_5.

$$a_j = \sum_j (w_{i,j} * x_i)$$

$$y_j = F(a_j) = \frac{1}{1+e^{-a_j}}$$

$a_1 = (w_{13} * x_1) + (w_{23} * x_2)$

$= (0.1 * 0.35) + (0.8 * 0.9) = 0.755$

$y_3 = f(a_1) = 1/(1 + e^{-0.755}) = 0.68$

$\boxed{y_3 = 0.68}$

$a_2 = (w_{14} * x_1) + (w_{24} * x_2)$

$= (0.4 * 0.35) + (0.6 * 0.9) = 0.68$

$y_4 = f(a_2) = 1/(1 + e^{-0.68}) = 0.6637$

$\boxed{y_4 = 0.6637}$

$a_3 = (w_{35} * y_3) + (w_{45} * y_4)$

$= (0.3 * 0.68) + (0.9 * 0.6637)$

$= 0.801$

$y_5 = f(a_3) = 1/(1 + e^{-0.801})$

$y_5 = 0.69$ (Network output)

$\boxed{y_5 = 0.69}$

$Y_{target} = 0.5$ (given in question)

$y_5 = 0.69$

Error $= y_{target} - y_5 = 0.5 - 0.69$

Error $= -0.19$

- Each weight changed by:

$$\Delta w_{ji} = \eta \delta_j o_i$$

$$\delta_j = o_j(1 - o_j)(t_j - o_j)$$ if j is an output unit

$$\delta_j = o_j(1 - o_j) \sum_k \delta_k w_{kj}$$ if j is a hidden unit

- where η is a constant called the learning rate
- t_j is the correct ~~[illegible]~~ target output for unit j
- δ_j is the error measure for unit j.

Backward Pass: Compute $\delta_3, \delta_4, \delta_5$

$$\delta_5 = y(1-y)(y_{target} - y)$$
$$= (0.69)(1-0.69)(0.5-0.69) = -0.0406$$

$$\delta_3 = y_3(1-y_3)(\delta_5 * w_{35})$$
$$= (0.68)(1-0.68)(-0.0406 \times 0.3)$$

$$\boxed{\delta_3 = -0.00265}$$

$$\boxed{\delta_5 = -0.0406}$$

$$\delta_4 = y_4(1-y_4)(\delta_5 \times w_{45})$$

$$= (0.6637)(1-0.6637)(-0.0406 \times 0.9)$$

$$\boxed{\delta_4 = -0.0082}$$

Compute new weights

$$\Delta w_{ji} = \eta \delta_j o_i$$

$$\Delta w_{45} = \eta \delta_5 y_4$$

$$\Delta w_{45} = (1)(-0.0406)(0.6637)$$

$$\Delta w_{45} = -0.0269$$

$$w_{45}(\text{new}) = \Delta w_{45} + w_{45}(\text{old})$$

$$= (-0.0269) + (0.9)$$

$$\boxed{w_{45}(\text{new}) = 0.8731}$$

$$\Delta w_{35} = \eta \delta_5 y_3$$

$$= (1)(-0.0406)(0.68)$$

$$\Delta w_{35} = -0.027608$$

$$w_{35}(\text{new}) = \Delta w_{35} + \Delta_{35}(\text{old})$$

$$= (-0.027608) + 0.3$$

$$\boxed{w_{35}(new) = 0.2724}$$

$$\Delta w_{24} = \eta \delta_4 x_2$$
$$= (1)(-0.0082)(0.9)$$
$$\Delta w_{24} = -0.00738$$

$$w_{24}(new) = \Delta w_{24} + w_{24}(old)$$
$$= (-0.00738) + (0.6)$$
$$\boxed{w_{24}(new) = 0.5926}$$

$$\Delta w_{13} = \eta \delta_3 x_1$$
$$= (1)(-0.00265)(0.35)$$
$$\Delta w_{13} = -0.0009275$$

$$w_{13}(new) = \Delta w_{13} + w_{13}(old)$$
$$= (-0.0009275) + (0.1)$$
$$\boxed{w_{13}(new) = 0.0991}$$

$$\Delta w_{23} = \eta \delta_3 x_2$$
$$= (1)(-0.00265)(0.9)$$
$$\Delta w_{23} = -0.002385$$

$w_{23}(new) = \Delta w_{23} + w_{23}(old)$

$= (-0.002385) + (0.8)$

$\boxed{w_{23}(new) = 0.7976}$

$\Delta w_{14} = \eta \delta_4 x_1$

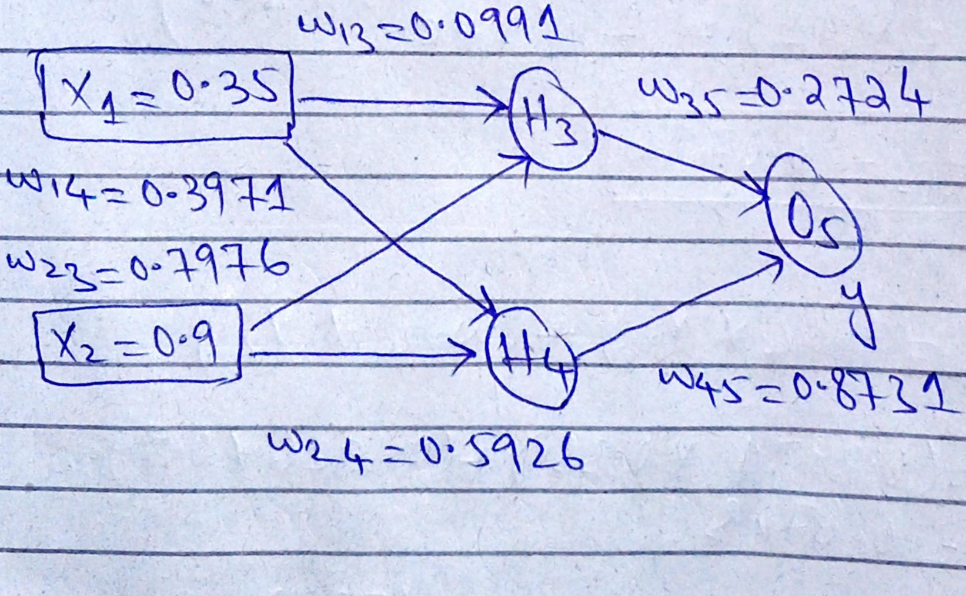
$= (1)(-0.0082)(0.35)$

$\Delta w_{14} = -0.00287$

$w_{14}(new) = \Delta w_{14} + w_{14}(old)$

$= -0.00287 + 0.4$

$\boxed{w_{14}(new) = 0.3971}$



Iteration 2

Forward Pass: Compute output for y_3, y_4 and y_5

$$a_j = \sum_j (w_{ij} * x_i)$$

$$y_j = F(a_j) = \frac{1}{1 + e^{-a_j}}$$

$$a_1 = (w_{13} * x_1) + (w_{23} * x_2)$$
$$= (0.0991 * 0.35) + (0.7976 * 0.9)$$
$$a_1 = 0.7525$$

$$y_3 = f(a_1) = 1 / (1 + e^{-0.7525})$$

$$\boxed{y_3 = 0.6796}$$

$$a_2 = (w_{14} * x_1) + (w_{24} * x_2)$$

$$= (0.3971 * 0.35) + (0.5926 * 0.9)$$
$$= 0.6723$$

$$y_4 = f(a_2) = 1 / (1 + e^{-0.6723})$$

$$\boxed{y_4 = 0.6620}$$

$a_3 = (w_{35} \times y_3) + (w_{45} \times y_4)$

$= (0.2724 \times 0.6797) +$
(0.8731×0.6620)

$= 0.7631$

$y_5 = f(a_3) = 1/(1 + e^{-0.7631})$

$\boxed{y_5 = 0.6820}$ Network Output

Error = $y_{target} - y_{output}$

Error = $y_{target} - y_5$
$= 0.5 - 0.6820$

Error = -0.182

→ update weights → find network output → find error → update weights → find network output → find error → cycles repeat.

—x——x—