DIWAS ADHIKARI (076BEI014) - 14

(Adaline)

Adaptive Linear Learning to design AND Neuron:-

for two-input AND gate; considering that inputs and output are bipolar, following is the truth table for AND Gate:-

x_1	x_2	$F = x_1 \cdot x_2 = t$
-1	-1	-1
-1	1	-1
1	-1	-1
1	1	1

Step I: Set up weights and bias to a small random value.

$\therefore$ Bias(b) = 0.1

$\therefore$ Weights: $\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} 0.2 \\ 0.3 \end{bmatrix}$

$\therefore$ Learning rate (α) = 0.1

Step II: Provide inputs with the targets for training.

Step III: Weight adjustment for first cycle/epoch;

$$\left\{ \begin{aligned} &\therefore y_{in} = \sum_{i=1}^{n} x_i w_i + b \\ &\therefore w_i^{(new)} = w_i(old) + \alpha \cdot x_i (t - y_{in}) \end{aligned} \right\}$$

(i) With $(x_1, x_2) = (-1, -1)$ & $t = -1$:-

$\therefore y_{in} = b + x_1 w_1 + x_2 w_2 = 0.1 + (-1) \cdot 0.2 + (-1) \cdot 0.3 = (-0.4)$

$\therefore w_1(new) = w_1(old) + \alpha \cdot x_1 (t - y_{in}) = 0.2 + 0.1(-1) \cdot [-1 + 0.4] = 0.26$

$\therefore w_2^{(new)} = w_2^{(old)} + \alpha \cdot x_2 \cdot (t - y_{in}) = 0.3 + 0.1(-1) \cdot [-1 + 0.4] = 0.36$

$\therefore b^{(new)} = b^{(old)} + \alpha \cdot (t - y_{in}) = 0.1 + 0.1[-1 + 0.4] = 0.04$

(ii) With $(x_1, x_2) = (-1, 1)$ & $t = -1$:-

$\therefore y_{in} = +0.04 + (-1) \cdot 0.26 + (1) 0.36 = 0.14$

$\therefore w_1^{(n)} = w_1^{(0)} + \alpha \cdot x_1 (t - y_{in}) = 0.26 + 0.114 = 0.374$

$\therefore w_2^{(n)} = w_2^{(0)} + \alpha x_2 (t - y_{in}) = 0.36 - 0.114 = 0.246$

$\therefore b^{(n)} = b^{(0)} + \alpha \cdot (t - y_{in}) = 0.04 - 0.114 = (-0.074)$

(iii) With $(x_1, x_2) = (1, -1)$ & $t = -1$:-

$\therefore y_{in} = (-0.074) + (1) \cdot 0.374 + (-1) \cdot 0.246 = 0.054$

$\therefore w_1^{(n)} = 0.374 + 0.1 \cdot (1) \cdot [-1 - 0.054] = 0.2686$

$\therefore w_2^{(n)} = 0.246 + 0.1 \cdot (-1) \cdot [-1 - 0.054] = 0.3514$

$\therefore b^{(n)} = (-0.074) + 0.1 [-1 - 0.054] = (-0.1794)$

(iv) With $(x_1, x_2) = (1, 1)$ & $t = 1$

$\therefore y_{in} = (-0.1794) + 0.2686 + 0.3514 = 0.4406$

$\therefore w_1(n) = 0.2686 + 0.1 \cdot (1) \cdot [(1 - 0.4406)] = 0.32454$

$\therefore w_2^{(n)} = 0.3514 + 0.1 (1) [1 - 0.4406] = 0.40734$

$\therefore b^{(n)} = b^{(0)} + \alpha \cdot (t - y_{in}) = (-0.1794) + 0.1 \cdot [1 - 0.4406]$
$= (-0.12346)$

Step IV:-

Continue weight adjustments till the maximum weight change $\Delta w_i = \alpha x_i(t - y_{in})$ is ~~not~~ less than the specified tolerance value (ε) i.e. stop the training if $\Delta w < \varepsilon$.

Till our first cycle; $\therefore \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} 0.32454 \\ 0.40734 \end{bmatrix}$

$\therefore b = -0.12346.$

Thus, if we keep on training the weights, we will approximate the weights to values that can distinctly predict the targets.

Step V:- Let's try to predict the target just by using weights from first cycle/epoch training. [for bipolar inputs; consider threshold = 0]

for $(x_1, x_2) = (-1, -1)$;

$$\therefore y_p = \sum_{i=1}^{2} x_i w_i + b = x_1 w_1 + x_2 w_2 + b$$
$$= (-1).0.32454 + (-1).0.40734 + (-0.12346)$$
$$= -0.85534 \ (<0)$$
$$\Rightarrow y_p = -1 \text{ (True)}$$

for $(x_1, x_2) = (-1, 1)$;

$$\therefore y_p = (-1).0.32454 + (1)0.40734 + (-0.12346)$$
$$= -0.04066 \ (<0)$$
$$\Rightarrow y_p = -1 \text{ (True)}$$

for $(x_1, x_2) = (1, -1)$;

$$\therefore y_p = (1.)0.32454 + (-1).0.40734 + (-0.12346)$$
$$= -0.20626 \ (<0)$$
$$\Rightarrow y_p = -1 \text{ (True)}$$

for $(x_1, x_2) = (1, 1)$;

$$\therefore y_p = (1).0.32454 + (1).0.40734 + (-0.12346)$$
$$= 0.60842 \ (>0)$$
$$\Rightarrow y_p = 1 \text{ (True)}$$

Hence, further training would make better distinction on predicted results for inputs like (-1, 1) and (1, -1) whose predicted target is too closer to zero but, still the weights are good enough to confirm the predictions and AND Neuron successfully works; based on just the first cycle as well.