Assignment Solution: AND Gate Learning

Using a Single Layer Perceptron

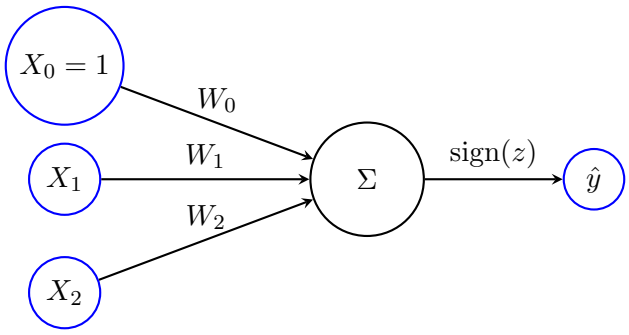
1 Perceptron Architecture

1.1 Structure Description

The perceptron consists of the following components:

- **Single Neuron**
- **Three Inputs:** $X_0 = 1$ (bias), X_1, X_2
- **Three Weights:** W_0, W_1, W_2
- **Activation Function:** Step function (sign)

1.2 Diagram



1.3 Formulas

$$z = W_0 \cdot X_0 + W_1 \cdot X_1 + W_2 \cdot X_2 = \boldsymbol{W}^T \boldsymbol{X} \tag{1}$$

$$\hat{y} = \text{step}(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{if } z < 0 \end{cases} \tag{2}$$

$$\text{error} = Y - \hat{y} \tag{3}$$

$$W_{\text{new}} = W_{\text{old}} + \eta \cdot \text{error} \cdot X \quad (\eta = 1) \tag{4}$$

2 Data Table - AND Gate

X_0	X_1	X_2	Y (Target)
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

3 Initial Weights

- **Initial Weights:** $W_0 = 0.5$, $W_1 = -0.3$, $W_2 = 0.2$
- **Learning Rate:** $\eta = 1$

4 Iteration Process

4.1 Legend

- Green - Correct classification (error = 0, no weight update)
- Red - Classification error (error $\neq$ 0, weight update triggered)
- Yellow - Epoch separator

Iter	Epoch	W_0	W_1	W_2	X_0	X_1	X_2	z	$\hat{y}$	Y	Error	W_0'	W_1'	W_2'
EPOCH 1														
1	1	0.5	-0.3	0.2	1	0	0	0.5	1	0	-1	-0.5	-0.3	0.2
2	1	-0.5	-0.3	0.2	1	0	1	-0.3	0	0	0	-0.5	-0.3	0.2
3	1	-0.5	-0.3	0.2	1	1	0	-0.8	0	0	0	-0.5	-0.3	0.2
4	1	-0.5	-0.3	0.2	1	1	1	-0.6	0	1	1	0.5	0.7	1.2
EPOCH 2														
5	2	0.5	0.7	1.2	1	0	0	0.5	1	0	-1	-0.5	0.7	1.2
6	2	-0.5	0.7	1.2	1	0	1	0.7	1	0	-1	-1.5	0.7	0.2
7	2	-1.5	0.7	0.2	1	1	0	-0.8	0	0	0	-1.5	0.7	0.2
8	2	-1.5	0.7	0.2	1	1	1	-0.6	0	1	1	-0.5	1.7	1.2
EPOCH 3														
9	3	-0.5	1.7	1.2	1	0	0	-0.5	0	0	0	-0.5	1.7	1.2
10	3	-0.5	1.7	1.2	1	0	1	0.7	1	0	-1	-1.5	1.7	0.2
11	3	-1.5	1.7	0.2	1	1	0	0.2	1	0	-1	-2.5	0.7	0.2
12	3	-2.5	0.7	0.2	1	1	1	-1.6	0	1	1	-1.5	1.7	1.2
EPOCH 4														
13	4	-1.5	1.7	1.2	1	0	0	-1.5	0	0	0	-1.5	1.7	1.2
14	4	-1.5	1.7	1.2	1	0	1	-0.3	0	0	0	-1.5	1.7	1.2
15	4	-1.5	1.7	1.2	1	1	0	0.2	1	0	-1	-2.5	0.7	1.2
16	4	-2.5	0.7	1.2	1	1	1	-0.6	0	1	1	-1.5	1.7	2.2
EPOCH 5														
17	5	-1.5	1.7	2.2	1	0	0	-1.5	0	0	0	-1.5	1.7	2.2
18	5	-1.5	1.7	2.2	1	0	1	0.7	1	0	-1	-2.5	1.7	1.2
19	5	-2.5	1.7	1.2	1	1	0	-0.8	0	0	0	-2.5	1.7	1.2
20	5	-2.5	1.7	1.2	1	1	1	0.4	1	1	0	-2.5	1.7	1.2
EPOCH 6 - Final Verification														
21	6	-2.5	1.7	1.2	1	0	0	-2.5	0	0	0	-2.5	1.7	1.2
22	6	-2.5	1.7	1.2	1	0	1	-1.3	0	0	0	-2.5	1.7	1.2
23	6	-2.5	1.7	1.2	1	1	0	-0.8	0	0	0	-2.5	1.7	1.2
24	6	-2.5	1.7	1.2	1	1	1	0.4	1	1	0	-2.5	1.7	1.2

5 Solution Summary

5.1 Final Weights

$$\boxed{W_0 = -2.5,\ W_1 = 1.7,\ W_2 = 1.2}$$

5.2 Statistics

- **Total Iterations:** 24 (20 learning iterations + 4 verification iterations)
- **Number of Epochs:** 6 (5 learning epochs + 1 verification epoch)
- **Final Accuracy:** 100% - all four examples classified correctly!

5.3 Verification of Results

Checking that the final weights correctly classify all examples:

$$\begin{aligned}
(X_1, X_2) = (0,0) &: z = -2.5 + 1.7(0) + 1.2(0) = -2.5 \to \hat{y} = 0 \quad \checkmark \\
(X_1, X_2) = (0,1) &: z = -2.5 + 1.7(0) + 1.2(1) = -1.3 \to \hat{y} = 0 \quad \checkmark \\
(X_1, X_2) = (1,0) &: z = -2.5 + 1.7(1) + 1.2(0) = -0.8 \to \hat{y} = 0 \quad \checkmark \\
(X_1, X_2) = (1,1) &: z = -2.5 + 1.7(1) + 1.2(1) = 0.4 \to \hat{y} = 1 \quad \checkmark
\end{aligned}$$

5.4 Decision Boundary Equation

The linear separation line generated by the perceptron:

$$-2.5 + 1.7X_1 + 1.2X_2 = 0 \tag{5}$$

Or:

$$X_2 = \frac{2.5 - 1.7X_1}{1.2} \tag{6}$$

5.5 Notes

- The AND gate is **linearly separable**, so the perceptron algorithm is guaranteed to converge to a solution.
- The final weights create a hyperplane that successfully separates the classes.
- The point $(1, 1)$ is the only one that receives a positive value in z, thus classified as 1 (True).
- All other three points receive negative values in z, thus classified as 0 (False).

6 Conclusions

1. The perceptron successfully learned to represent the logical AND gate.
2. The process converged after 5 epochs of active learning.
3. Epoch 6 served as verification that the model correctly classifies all samples.
4. The final weights reflect AND logic: both inputs must be 1 to overcome the negative bias.