The Perceptron

Shreeyesh Menon

1 What is a Perceptron?

The **perceptron** is one of the simplest and earliest models in Machine Learning. It is a type of *linear classifier* that tries to separate data into two classes using a straight line (in 2D), a plane (in 3D), or a hyperplane (in higher dimensions).

It was introduced by Frank Rosenblatt in 1957 and remains important as a building block for more complex neural networks.

2 The Perceptron Model

Given an input vector

$$\mathbf{x} = (x_1, x_2, \ldots, x_d),$$

the perceptron computes a weighted sum of the components:

$$z = w_1 x_1 + w_2 x_2 + \cdots + w_d x_d + b,$$

where

- w_i are the *weights*,
- b is the *bias*.

This value is then passed through a **step function** (also called the *Heaviside* function):

$$\hat{y} = \begin{cases} +1 & \text{if } z \geq 0, \\ -1 & \text{otherwise.} \end{cases}$$

So the perceptron outputs either $+1$ or -1, making it a binary classifier.

3 Geometric Interpretation

The perceptron learns a *decision boundary*. For input vector $\mathbf{x}$, the condition

$$\mathbf{w} \cdot \mathbf{x} + b = 0$$

defines a hyperplane. Points on one side of the hyperplane are classified as $+1$, and points on the other side as -1.

4 The Perceptron Learning Rule

The perceptron learns by updating the weights whenever it misclassifies a training example.

Given a training pair $(\mathbf{x}, y)$ where $y \in \{-1, +1\}$, the update rule is:

$$\mathbf{w} \leftarrow \mathbf{w} + \eta\, y\, \mathbf{x}$$

$$b \leftarrow b + \eta\, y$$

Here, $\eta > 0$ is the *learning rate*.

4.1 Why does this update make sense?

If the perceptron misclassifies a point, it means the dot product $\mathbf{w} \cdot \mathbf{x} + b$ had the wrong sign. Multiplying by y fixes that:

- If $y = +1$, the algorithm adjusts weights to push $\mathbf{w} \cdot \mathbf{x}$ *up*.
- If $y = -1$, it pushes $\mathbf{w} \cdot \mathbf{x}$ *down*.

This moves the decision boundary toward correctly classifying the example.

5 Perceptron Algorithm

1. Initialize weights and bias (for example, to zeros).
2. For each training example $(\mathbf{x}_i, y_i)$:
   (a) Compute prediction $\hat{y}_i$.
   (b) If $\hat{y}_i \neq y_i$, update:
   $$\mathbf{w} \leftarrow \mathbf{w} + \eta\, y_i\, \mathbf{x}_i, \qquad b \leftarrow b + \eta\, y_i.$$
3. Repeat until all points are classified correctly or a maximum number of iterations is reached.

6 When Does the Perceptron Work?

The perceptron algorithm **converges** if the data is *linearly separable*. That is, if a perfect straight-line separator exists. If the data is not linearly separable, the perceptron will never settle and will keep adjusting its weights forever.

7 Limitations

The perceptron:

- can only learn linear decision boundaries,
- cannot solve problems like XOR,
- is the foundation of modern neural networks.

Despite its simplicity, it represents the idea of combining weighted inputs and applying a nonlinear activation. The simple perceptron is the precursor to deep neural networks.

8 Summary

- The perceptron is a simple binary classifier.
- It computes a weighted sum of inputs and applies a step function.
- Learning is done via the perceptron update rule.
- Works only if the data is linearly separable.