Perceptron

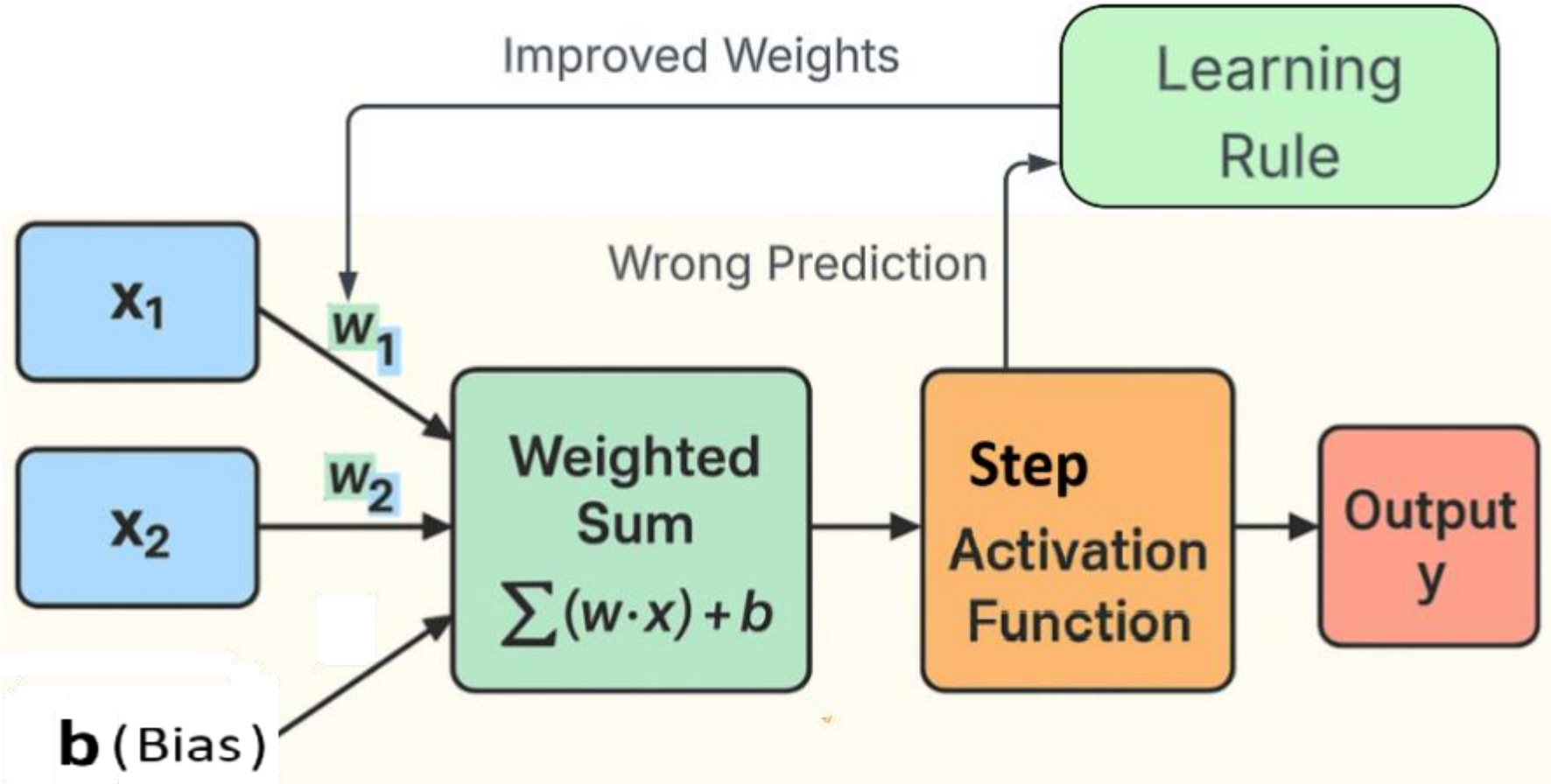
Peceptron

- A **perceptron** is the simplest type of artificial neural network model.

- It is a **binary classifier** that determines whether an input belongs to one of two classes (**Yes/No, 1/0, True/False**) by computing a **weighted sum of the inputs plus a bias**, and then passing the result through an **activation function**.

Example of classes:

Email Classification
Class 1 → Spam
Class 0 → Not Spam
Student Result Prediction
Class 1 → Pass
Class 0 → Fail
Image Recognition (very simple)
Class 1 → Cat
Class 0 → Not Cat


Improved Weights
Learning Rule
Wrong Prediction
x_1
x_2
w_1
w_2
Weighted Sum
$\sum(w \cdot x) + b$
Step Activation Function
Output y
b (Bias)

Perceptron

Step-by-Step Working of a Perceptron

Weighted Sum + Bias

The perceptron first calculates the **net input**:

$$z = \sum_{i=1}^{n} w_i x_i + b$$

- Here, inputs (x_i) are multiplied by their weights (w_i)
- Then the bias b is added

- The **bias shifts the activation function** to the left or right.
- Without bias, the decision boundary (line, plane, or hyperplane) would **always pass through the origin**.
- Bias gives the model **flexibility** to fit data better.

Activation Function

The result z is passed into an **activation function** $f(z)$, which decides the perceptron's **final output**.

For the *classical perceptron*, the **step function** is used:

$$f(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{if } z < 0 \end{cases}$$

- If the weighted sum + bias is **positive or zero**, the perceptron outputs **1**
- If it is **negative**, the perceptron outputs **0**

Perceptron Learning Rule

- In the **Perceptron**, the output is based on the **step function** (0 or 1).
- If the prediction is **wrong**, the weights are updated as:

$$w_i^{new} = w_i^{old} + \Delta w_i$$

where

$$\Delta w_i = \eta \cdot (y_{true} - y_{pred}) \cdot x_i$$

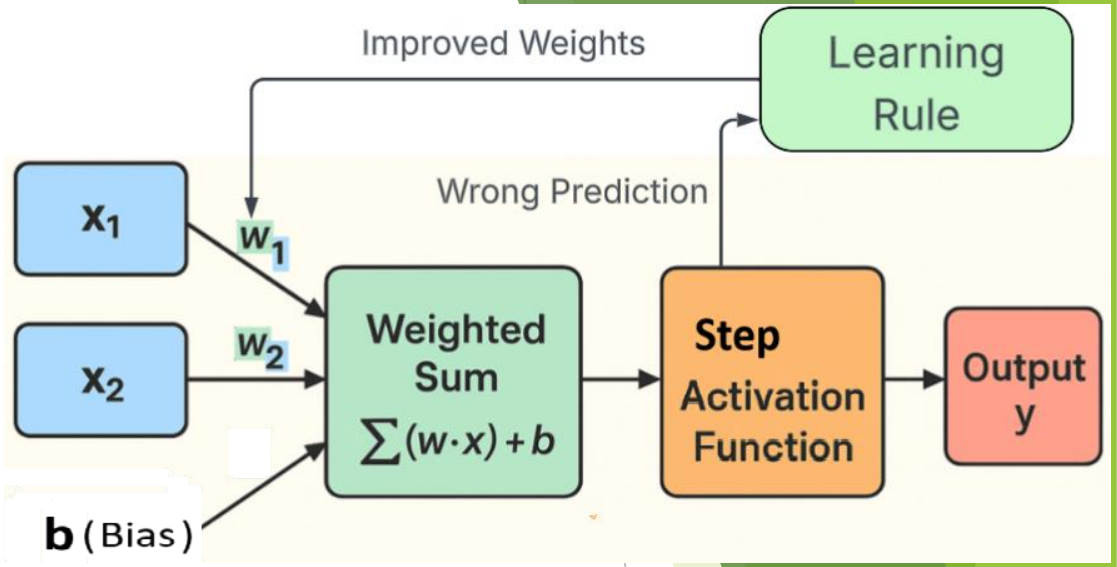
- Here:
  - η = learning rate
  - y_{true} = actual label (0 or 1)
  - y_{pred} = perceptron's predicted output
  - x_i = input feature

Final Output

Thus, the perceptron's **output** is:

$$y = f\left(\sum_{i=1}^{n} w_i x_i + b\right)$$

Conclusion



- Inputs are multiplied and added together with bias
- The activation function acts like a **decision rule**
- The perceptron finally gives an output: **Yes/No, 1/0, True/False**
- Classes are the possible labels or outcomes (e.g., Spam vs Not Spam, Pass vs Fail)

In the **perceptron model**:

- The **inputs** (x_i) are **features** of the data, not classes.
  - Example: If we are classifying emails as spam/not spam →
    - x_1 = number of links,
    - x_2 = number of capital letters,
    - x_3 = presence of "FREE" word, etc.
- The **output** (y) is what represents the **class**.
  - With a step activation function:
    - $y = 1$ → belongs to **Class 1**
    - $y = 0$ → belongs to **Class 0**

So:

- x_i = **inputs/features** (measurements, values, observations)
- y = **class label** (predicted by the perceptron)