# Neural Network Parameters: **Weights, Biases, and Total Parameters**

## 1. **Basic Terminology**

Before diving into calculations, let's understand the fundamental concepts:

- **Neuron**: The basic unit in a neural network that receives inputs, applies a weighted sum, adds a bias, and passes the result through an activation function.
  - Formula: $z = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n + b$
  - $w_i$: Weight for input $x_i$
  - $b$: Bias for the neuron
- **Layer**: A collection of neurons operating together. Layers are categorized as:
  - **Input Layer**: Where the data enters the network (no weights/biases here).
  - **Hidden Layers**: Intermediate layers between input and output layers.
  - **Output Layer**: Produces the final prediction/output.
- **Parameters**: Trainable variables in the network.
  - **Weights**: One for each connection between neurons in consecutive layers.
  - **Biases**: One bias for each neuron (except input neurons).

## 2. **Calculating Parameters**

To calculate the **total number of parameters** in a neural network, we need to compute weights and biases layer by layer.

---

## Weight Calculation

The number of weights is determined by the connections between neurons in two consecutive layers:

$$\text{Number of weights} = (\text{Number of neurons in previous layer}) \times (\text{Number of neurons in current layer})$$

---

## Bias Calculation

Each neuron (except input neurons) has one bias associated with it:

$$\text{Number of biases} = (\text{Number of neurons in current layer})$$

---

## Example Neural Network

Suppose we have the following architecture:

1. **Input Layer**: 4 neurons ($x_1, x_2, x_3, x_4$)
2. **Hidden Layer 1**: 5 neurons ($h_1, h_2, h_3, h_4, h_5$)
3. **Hidden Layer 2**: 3 neurons ($h_6, h_7, h_8$)
4. **Output Layer**: 2 neurons ($y_1, y_2$)

---

## Step-by-Step Calculation

### Layer 1 (Input → Hidden Layer 1)

- **Weights**:
  - Connections between 4 input neurons and 5 hidden neurons:
$$4 \times 5 = 20$$
- **Biases**:
  - 1 bias for each of the 5 neurons in Hidden Layer 1:
$$5$$

### Layer 2 (Hidden Layer 1 → Hidden Layer 2)

- **Weights**:
  - Connections between 5 neurons in Hidden Layer 1 and 3 neurons in Hidden Layer 2:
$$5 \times 3 = 15$$
- **Biases**:
  - 1 bias for each of the 3 neurons in Hidden Layer 2:
$$3$$

### Layer 3 (Hidden Layer 2 → Output Layer)

- **Weights**:
  - Connections between 3 neurons in Hidden Layer 2 and 2 neurons in the Output Layer:
$$3 \times 2 = 6$$
- **Biases**:
  - 1 bias for each of the 2 neurons in the Output Layer:

---

## Total Parameters

Add all the weights and biases across layers:

$$\text{Total Weights} = 20 + 15 + 6 = 41$$

$$\text{Total Biases} = 5 + 3 + 2 = 10$$

$$\text{Total Parameters} = 41 + 10 = 51$$

---

## General Formula for Total Parameters

For a neural network with $L$ layers:

1. **Weights**:

$$\text{Total Weights} = \sum_{i=1}^{L} (\text{neurons in layer } i) \times (\text{neurons in layer } (i+1))$$

2. **Biases**:

$$\text{Total Biases} = \sum_{i=1}^{L} (\text{neurons in layer } (i+1))$$

3. **Total Parameters**:

$$\text{Total Parameters} = \text{Total Weights} + \text{Total Biases}$$

---

## Conceptual Notes

- **Increasing Parameters**: Adding more neurons or layers increases the parameters, which makes the model more expressive but risks overfitting.
- **Bias Role**: Bias shifts the activation function, enabling the network to model data better.
- **Weight Updates**: During training, weights and biases are adjusted using optimization algorithms (e.g., Gradient Descent).