# #Activation Function

An activation function decides how much a neuron should fire.

A neuron does this: $z = w \cdot x + b$

The activation function transforms this raw value z into an output:

$$a = f(z)$$

without activation functions, neural networks would only learn straight lines - no matter how deep they are.

---

1. Sigmoid

-Formula: $$\sigma(z) = \frac{1}{1 + e^{-z}}$$

-Range (0, 1)

-Intuition
- squashes input into a probability
- used historically for binary classification

-Problems
- Vanishing gradients
- Not zero-centered

- where used
- Output layer for binary classification

## 2. Tanh (Hyperbolic Tangent)

Formula

$$\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$$

Range

$$(-1, 1)$$

Intuition
- Centered around ~~zero~~.
- Stronger gradients than sigmoid.

Problems
- Still suffers from vanishing gradients

Where used
- Older neural networks
- RNNs (historically)

---

## 3. ReLU (Rectified Linear Unit)

Formula

$$\text{ReLU}(z) = \max(0, z)$$

Range

$$[0, \infty)$$

Intuition
- Simple and fast
- Keeps positive values, kills negatives

Advantages
- Solves vanishing gradient problem
- Sparse activations

Problems
- "Dying ReLU" (neurons stuck at 0)

Where used
- Hidden layers (default choice)

## 4. GELU (Gaussian Error Linear Unit)

- Formula:

$$GELU(z) = z \cdot \Phi(z)$$

where $\Phi(z)$ is the Gaussian CDF.

- Intuition
  - Smooth version of ReLU
  - Weighs inputs by probability
- Advantages
  - Better gradient flow
  - Smooth non-linearity
- Where used
  - Transformers (BERT, GPT)

---

## 5. Softmax

- Formula

$$Softmax(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}$$

- Range

$$(0,1), \quad \sum p_i = 1$$

- Intuition
  - Converts raw scores into probabilities
  - Emphasizes the largest values
- Where used
  - Output layer for multi-class classification.

---

# Mental Model

Weights learn patterns, Activation functions decide behavior.