Activation Function – Easy Explanation Notes

1. Definition

An **activation function** is a mathematical rule used in a neural network to decide the final output of a neuron. It takes the total input received by a neuron and decides whether the neuron should be active (send signal) or inactive. In simple words, it helps a neuron decide how much influence it should have in the next layer.

2. Purpose

- Adds **non-linearity** to the network so it can learn complex data patterns.
- Helps the network make correct decisions and adjust outputs smoothly.
- Allows the network to learn real-world data like images, speech, and numbers.

3. Why We Need Activation Functions

If we do not use activation functions, the neural network will behave like a simple linear equation (only performing addition and multiplication). Such a model can only draw straight lines and cannot understand curved or complex relationships. By using activation functions, the network becomes more powerful and can solve complex problems.

4. Common Types of Activation Functions

- **Step Function:**

$$f(x) = \begin{cases} 1, & x > 0 \\ 0, & x \leq 0 \end{cases}$$

  Output is either 0 or 1 (yes or no). It gives a hard decision.

- **Sigmoid Function:**

$$f(x) = \frac{1}{1 + e^{-x}}$$

  Converts any number into a value between 0 and 1. It is smooth and continuous.

- **Tanh (Hyperbolic Tangent) Function:**

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

  Output lies between -1 and $+1$. It is similar to Sigmoid but centered around zero.

- **ReLU (Rectified Linear Unit):**

$$f(x) = \max(0, x)$$

  Keeps only positive values and ignores negatives. It is simple and fast.

- **Leaky ReLU:**

$$f(x) = \begin{cases} x, & x > 0 \\ 0.01x, & x \leq 0 \end{cases}$$

It allows a small negative value instead of making it zero to prevent "dead neurons".

- **ELU (Exponential Linear Unit):**

$$f(x) = \begin{cases} x, & x > 0 \\ \alpha(e^x - 1), & x \leq 0 \end{cases}$$

ELU gives smooth negative values instead of sharp cuts, helping training stability.

- **Softplus Function:**

$$f(x) = \ln(1 + e^x)$$

This is a smooth version of ReLU. It gradually increases instead of cutting off sharply.

- **Softmax Function:**

$$f(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

Converts multiple outputs into probabilities that add up to 1. Used for multi-class classification.

5. Real-Life Example: "Wake Up or Sleep More" Decision

Every morning your brain decides whether to wake up or sleep more. This decision depends on how energetic or tired you feel. We can represent this process with a simple neuron formula:

$$z = w \times x + b$$

where:

- x = your **energy level** (input)
- w = **weight** that shows how important your energy is
- b = **bias** that represents your natural tendency (habit)
- z = total score that decides if you wake up or not

6. Where Do These Values Come From?

In this example, we choose:

$$w = 1.5, \quad b = -0.3, \quad x = 0.8 \text{ or } -1.0$$

Each value has a meaning:

- **Input (x):** This represents your energy level.

  - When $x = 0.8$, you are feeling fresh and have good energy.
  - When $x = -1.0$, you are very tired and want to sleep more.

- **Weight (w):** This controls how strongly your energy level affects the decision.
  - $w = 1.5$ means energy has a big effect — more energy makes waking up more likely.
  - If we used a smaller weight like 0.5, energy would affect the decision less.

- **Bias (b):** This represents your natural habit or tendency.
  - $b = -0.3$ means you are a little lazy — you need higher energy to wake up.
  - If b were positive, you would be more likely to wake up even with low energy.

Example Outputs for Each Activation Function (using z = 0.9):

- **Step Function:** Since $z > 0$, $f(z) = 1$. Output = 1 (active).

- **Sigmoid Function:**
$$f(z) = \frac{1}{1 + e^{-0.9}} = 0.711$$
Output = 0.711 (71% active).

- **Tanh Function:**
$$f(z) = \frac{e^{0.9} - e^{-0.9}}{e^{0.9} + e^{-0.9}} = 0.716$$
Output = 0.716 (balanced activation).

- **ReLU Function:**
$$f(z) = \max(0, 0.9) = 0.9$$
Output = 0.9 (positive signal kept).

- **Leaky ReLU:**
$$f(z) = 0.9 \text{ since } z > 0; \text{ if } z = -1, f(-1) = 0.01(-1) = -0.01$$
Small negative values pass through.

- **ELU Function (with $\alpha = 1$):**
$$f(0.9) = 0.9, \quad f(-1) = e^{-1} - 1 = -0.63$$
Smooth for negative inputs.

- **Softplus Function:**
$$f(z) = \ln(1 + e^{0.9}) = \ln(1 + 2.46) = 1.24$$
Smooth version of ReLU.

- **Softmax Function:** For inputs $[0.9, 0.1, -0.5]$:
$$f(0.9) = \frac{e^{0.9}}{e^{0.9} + e^{0.1} + e^{-0.5}} = \frac{2.46}{2.46 + 1.10 + 0.61} = 0.55$$
Output probability for this class = 55%.

We choose these numbers because they are small and simple, so calculations are easy and we can test both situations (awake and asleep).

7. Step-by-Step Calculation

Case 1: You feel fresh ($x = 0.8$)

$$z = (1.5)(0.8) + (-0.3) = 1.2 - 0.3 = 0.9$$

Now apply activation functions:

- **Step Function:** Since $z > 0$, $f(z) = 1$ (you wake up).
- **Sigmoid Function:**

$$f(z) = \frac{1}{1 + e^{-0.9}} = \frac{1}{1 + 0.4066} = 0.711$$

  There is a 71% chance that you will wake up.

- **ReLU Function:**

$$f(z) = \max(0, 0.9) = 0.9$$

  You are active and ready to start the day.

Case 2: You are tired ($x = -1.0$)

$$z = (1.5)(-1.0) + (-0.3) = -1.5 - 0.3 = -1.8$$

Apply activation functions again:

- **Step Function:** Since $z < 0$, $f(z) = 0$ (you sleep more).
- **Sigmoid Function:**

$$f(z) = \frac{1}{1 + e^{-(-1.8)}} = \frac{1}{1 + e^{1.8}} = \frac{1}{1 + 6.0496} = 0.142$$

  There is only a 14% chance you will wake up — you keep sleeping.

- **ReLU Function:**

$$f(z) = \max(0, -1.8) = 0$$

  You are inactive (still asleep).

8. Understanding the Results

- When z is **positive**, the neuron activates — this means "wake up".
- When z is **negative**, the neuron does not activate — this means "sleep more".

The values we chose (x, w, b) let us test both sides — one where you have energy (positive z) and one where you are tired (negative z). This helps us see how activation functions behave in different situations.

9. Key Points

- Step function gives a clear yes/no decision.
- Sigmoid converts the result into a probability between 0 and 1.
- ReLU keeps positive outputs and removes negative ones.
- Tanh works well for data that can be both positive and negative.
- Softmax is used when there are multiple output classes.
- ELU and Softplus are smooth versions that improve learning in some models.
- The weight and bias decide how sensitive the decision is.

10. Summary

Activation functions make neurons intelligent. They help the network decide when to activate and how strongly to react, just like your brain decides whether to wake up or sleep based on your energy level.