Chapter 4 : Activation Functions

We use different activation functions for different cases.

The activation function is applied to the output of a neuron (or layer of neurons), which modifies outputs.

Neural network will have two types of activation functions:

Activation function used in hidden layers(will be the same for all of them, but it doesn’t have to.)

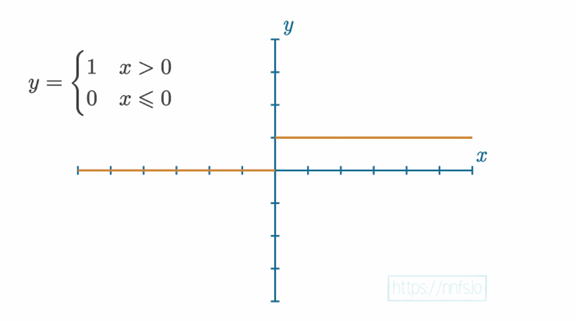
Activation function used in the output layer.

**The Step Activation Function**

this activation function serves is to mimic a neuron “firing” or “not firing” based on input information.

if the weights · inputs + bias​ results in a value greater than 0, the neuron will fire and output a 1; otherwise, it will output a 0.

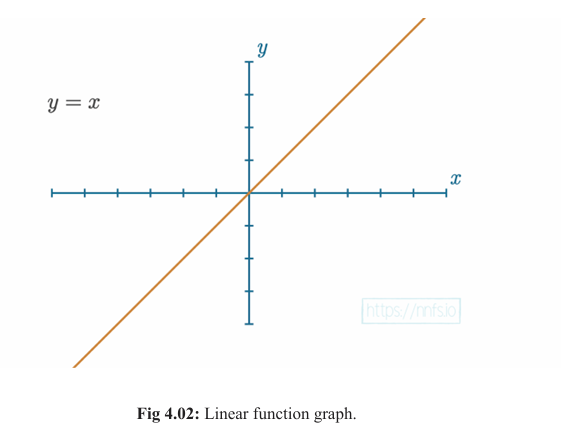
has been used historically in hidden layers, but nowadays, it is rarely a choice.



**The Linear Activation Function**

simply the equation of a line where y=x and the output value equals the input.

usually applied to the last layer’s output in the case of a regression model — a model that outputs a scalar value instead of a classification.



**The Sigmoid Activation Function**

The step function isn't very helpful for training because it only gives outputs of 0 or 1, offering no info on how close it was to switching. This makes it hard for optimizers to adjust weights and biases effectively. More detailed activation functions work better for learning.

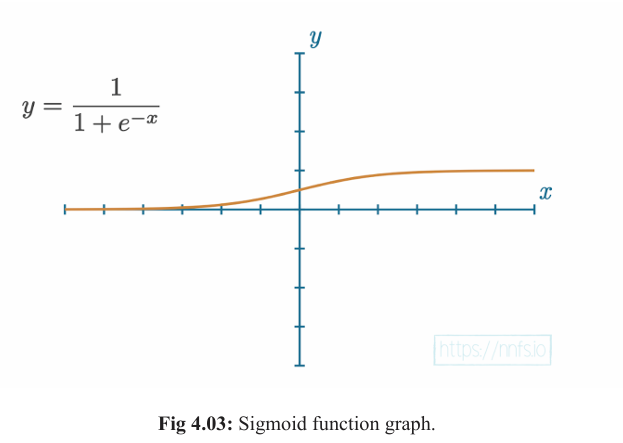
original, more granular, activation function used for neural networks.

Returns a value:

in the range of 0 for negative infinity,

0.5 for the input of 0,

1 for positive infinity.



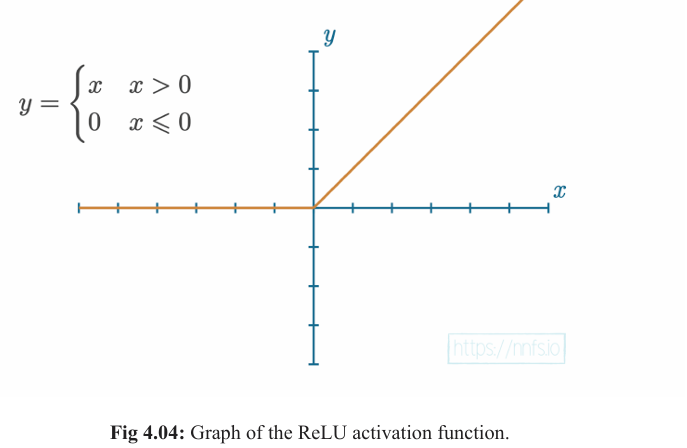
**The Rectified Linear(ReLU) Activation Function**

y=x​, clipped at 0 from the negative side.

If x​ is less than or equal to 0​, then y​ is 0​ — otherwise, y​ is equal to x​.

Simple, powerful, most widely use – mainly speed and efficiency.

extremely close to being a linear activation but non linear due to bend after 0



Why is non-linear function essential to the neural network?

To introduce non-linearity into the model, which helps to learn complex patterns and approximate any functions unlike the linear function. Thus, helps to learn complex, real-world data with any flexibility.

So, for a neural network to fit a nonlinear function, it needs two or more hidden layers, each with nonlinear functions.

**The Softmax Activation Function**

why are we bothering with another activation function? It just depends on what our overall goals are.