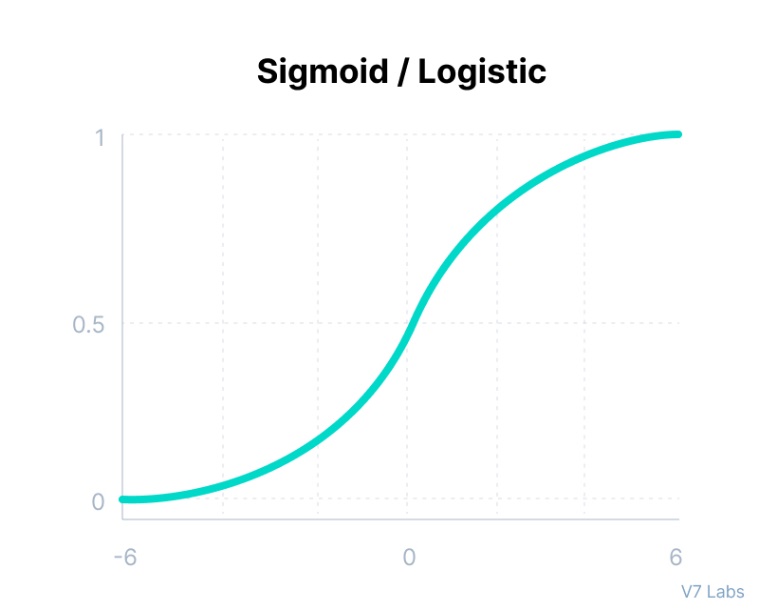
**Activation Function**

Purpose of activation function is to add non-linearity to the neural network.

* **Sigmoid/Logistic Activation Function**

This function takes any real value as input and outputs values in the range of 0 to 1.

The larger the input (more positive), the closer the output value will be to 1, whereas the smaller the input (more negative), the closer the output will be to 0.

Graph,

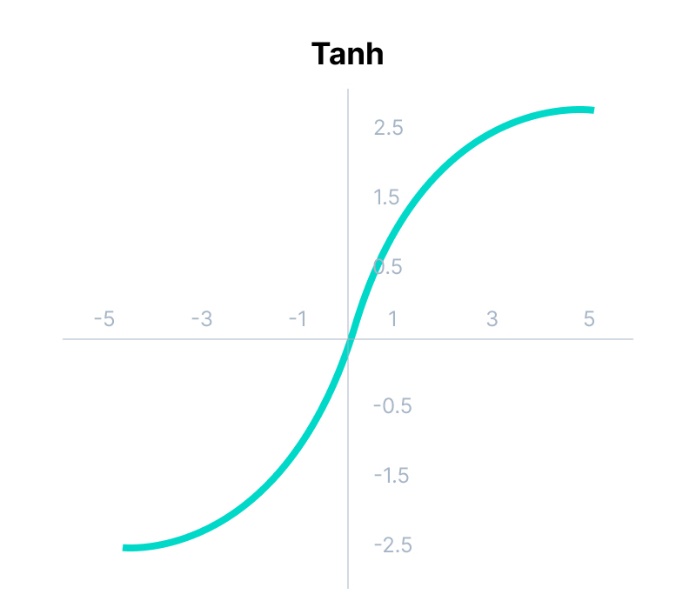
Mathematical representation,

It is used for model where we have to predict the probability as output.

* **Tanh Function (Hyperbolic Tangent)**

Tanh function is similar to sigmoid/logistic activation function with the difference in output range of -1 to 1.

The larger the input (more positive), the closer the output value will be to 1, whereas the smaller the input (more negative), the closer the output will be to -1.

Graph,

Mathematical representation,

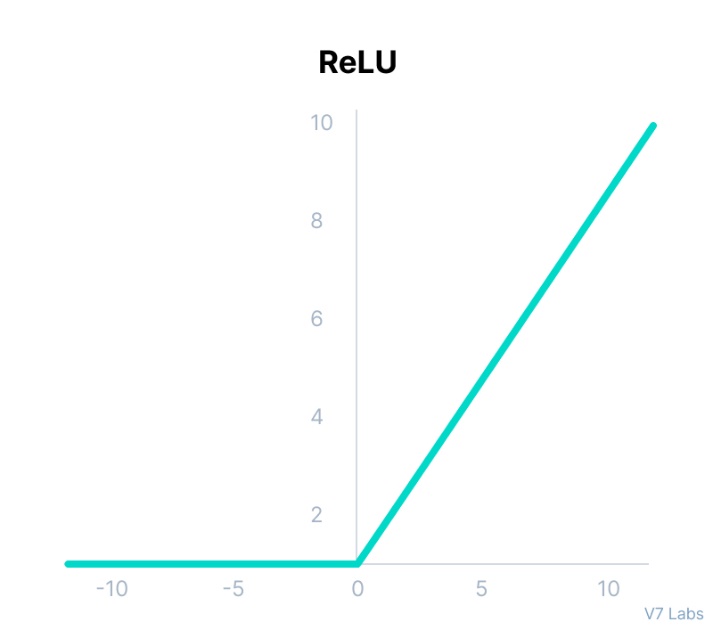
The output of the tanh activation function is zero cantered; hence we can easily map the output values as strongly negative, neutral, or strongly positive.

* **ReLU**

ReLU stands for Rectified Linear Unit.

ReLU function does not activate all the neurons at the same time.

The neurons will only be deactivated if the output of the linear transformation is less than 0.

Graph,

Mathematical representation,

Since only a certain number of neurons are activated, the ReLU function is far more computationally efficient as compared to the sigmoid and tanh functions.

* **SoftMax Function**

Sigmoid function faces certain problems.

Let’s suppose we have five output values of 0.8, 0.9, 0.7, 0.8, and 0.6, respectively. But we can’t move forward with it.

The above values don’t make sense as the sum of all the output probabilities should be equal to 1.

The SoftMax function is described as a combination of multiple sigmoids.

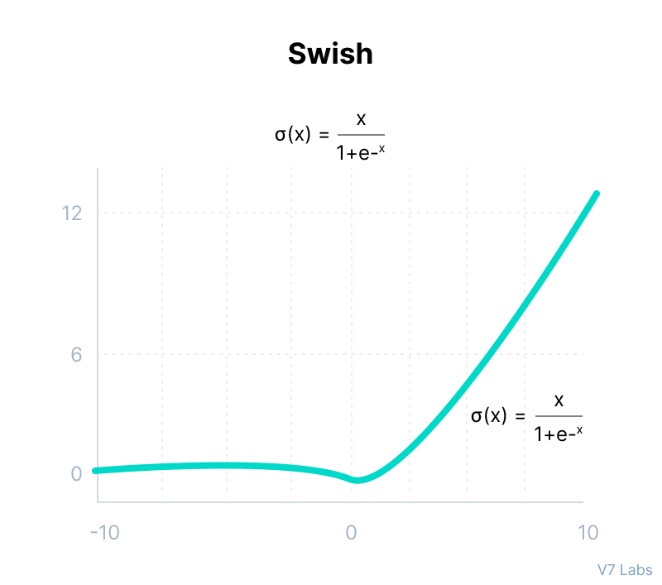
It calculates the relative probabilities. Similar to the sigmoid/logistic activation function, the SoftMax function **returns the probability of each class.**

It is most commonly used as an activation function for the last layer of the neural network in the case of multi-class classification.

Mathematical representation,

* **Swish**

This function is bounded below but unbounded above i.e. Y approaches to a constant value as X approaches negative infinity but Y approaches to infinity as X approaches infinity.

Graph,

Mathematical representation,

Swish is a smooth function that means that it does not abruptly change direction like ReLU does near x = 0. Rather, it smoothly bends from 0 towards values < 0 and then upwards again.

The swish function being non-monotonous enhances the expression of input data and weight to be learnt.