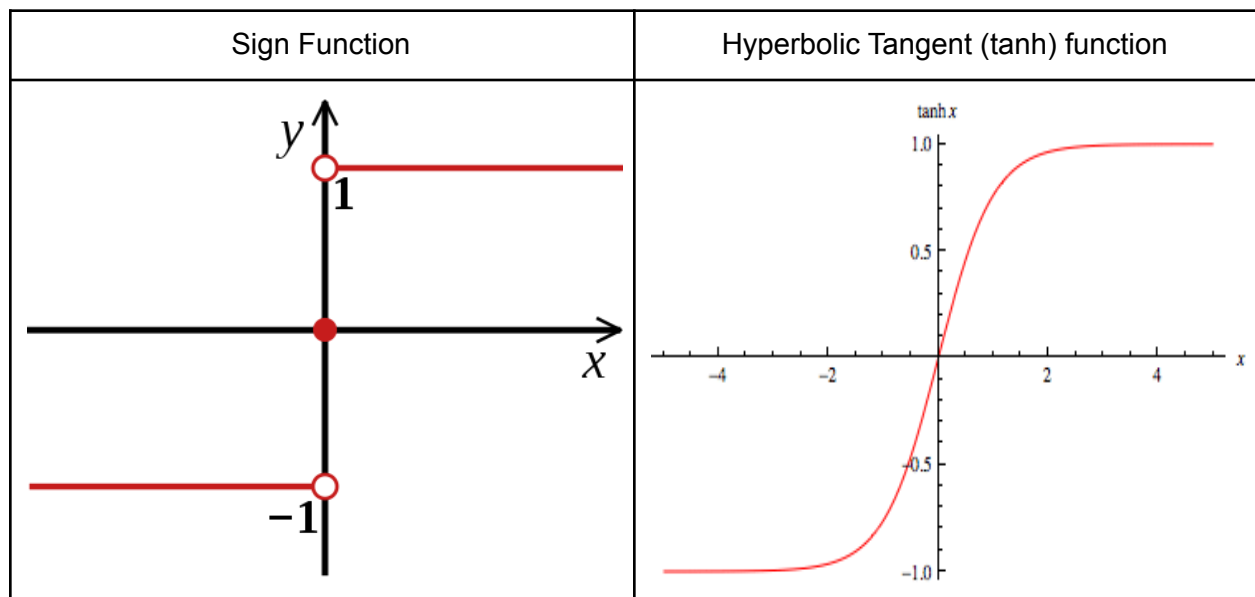


Neural network learning algorithm is based on a technique called backpropagation. Although it's hard to fully understand the backpropagation from some frameworks, backpropagation simply consists of three steps. First step is to present training examples to the neural network. Second step is to compare the output value to the desired value (ground truth). These 2 steps are called the **forward pass**. Third step is to adjust the weight on the neural network to produce a better output. This step is called the **backward pass**. The weight is adjusted using the gradient descent method. These 3 steps will be done many times until we produce a good value of the weight due to the constantly changing gradient from each step or in other words until the neural network model can produce good results.

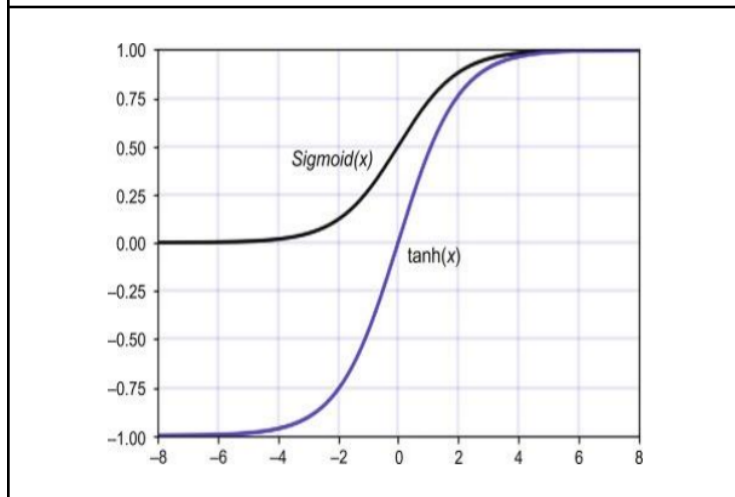
For multilevel networks, the method is different because gradient descent doesn't work the same way while being applied to the perceptron. More precisely because on a multilevel network, output from one layer must be forwarded to the next layer. To solve this, we have to use a different function that is differentiable in order to calculate the gradient. Based on a previous research done by Rumelhart, Hinton, and Williams in 1986, it's known that the sign function has discontinuity at zero which is not suitable for this type of network. Sign function will give a result between -1 or 1, similarly to 0 or 1 which is only suitable for a binary type of result. These researchers then used an S-shaped function to replace the old sign function due to its similar shape but different behavior. One of function that is S-shaped is Hyperbolic Tangent function.



As Hyperbolic Tangent is continuous and can be used to define the output value no matter how big or small the value of the input variable, it's used widely to calculate a certain value as the output ranges from -1 to 1 but it's centered on 0. It works well for the perceptron because the output value then can be used for the next layer and its differentiability is useful for the learning process.

There are some other continuous functions that are also used as an activation on a neural network process. One that is pretty similar in terms of shape to Hyperbolic Tangent is known as Sigmoid Function.

Sigmoid Function and Hyperbolic Tangent Comparison



Sigmoid Function is also widely used on many Deep Learning cases for multilevel networks. Just like Hyperbolic Tangent, it's also symmetrical around 0 on the x-axis. And around 0.5 on the y-axis. The difference between these functions is located on its y-axis scale. Sigmoid function scaled from 0 to 1 and centered in 0.5 and is suitable for probability type of result since the value of probability cannot go below 0. Practically, Hyperbolic Tangent is normally used for hidden layers because the output number is centered around 0, and sigmoid is normally used for the output layer to produce probability types of output.