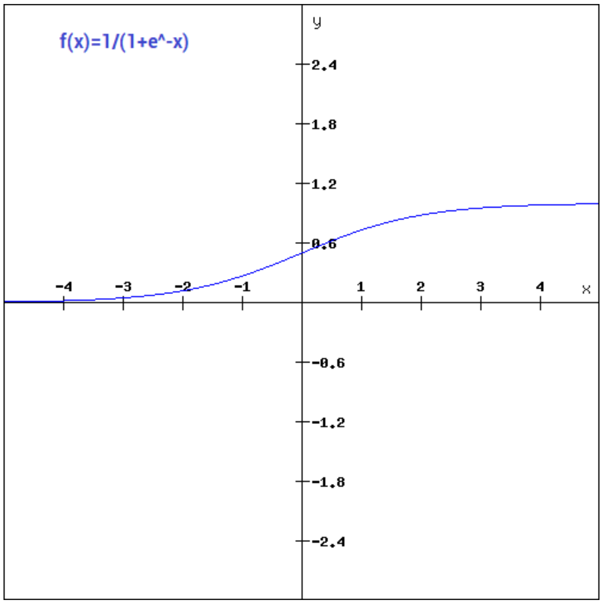
Activation Functions and Their Uses

Activation are very important for artificial neural networks (ANN) to understand and learn the complex patterns present in a given data. Most of the datasets used is not linear and is very complex. It makes it easy for the model to generalize or adapt with variety of data and to differentiate between the output.

Popularly used Activation Functions

Sigmoid function: This is a smooth function and is continuously differentiable. Its gradient is very high between the values of -3 and 3 but gets much flatter in other regions. This means that in this range small changes in x would also bring about large changes in the value of Y. So the function essentially tries to push the Y values towards the extremes. This is a very desirable quality when we're trying to classify the values to a particular class. The main reason why we use sigmoid function is because its values lies between 0 and 1. It is used in models where probability is given as an output as probability lies between 0 and 1.

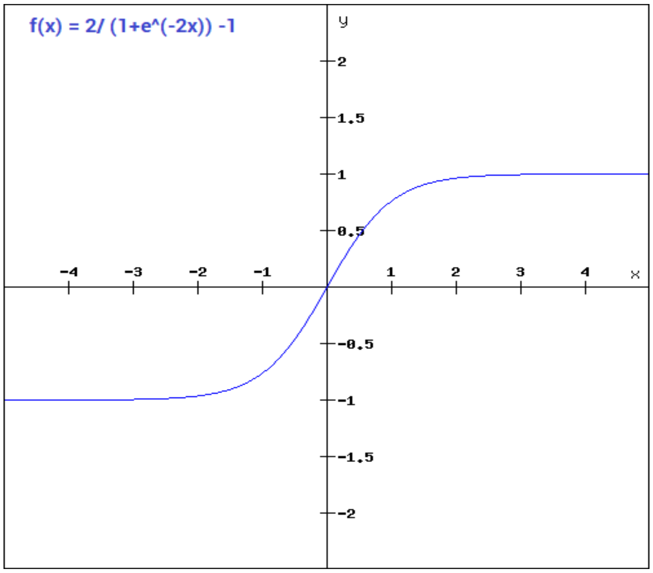
$$f(x) = \frac{1}{1 + e^{-x}}$$



Although it also has some disadvantages. As its value only range from 0 to 1, it is not symmetric around the origin and the values received are all positive. Thus, the values going to the next neuron will be of the same sign. The function is pretty flat beyond the +3 and -3 region. This means that once the function falls in that region the gradients become very small. This means that the gradient is approaching to zero and the network is not really learning. This can be addressed by scaling the sigmoid function.

Tanh/Hyperbolic Tangent: The Tanh function is an activation function which re-scales the values between -1 and 1 by applying a threshold just like a sigmoid function. The advantage is that the negative inputs will be mapped strongly negative and the positive inputs will be mapped positive. It is actually just a scaled version of the sigmoid function. An important feature is that it is symmetric over the origin. It can be directly written as:

$$\boldsymbol{tanh(x) = \frac{2}{1 + e^{-2x}} - 1}$$



Relu Function/ Rectified Linear Unit:

The ReLU is the most used activation function in the world right now. Instead of sigmoid, most recent deep learning networks use rectified linear units (ReLUs) for the hidden layers. It is given by:

$$f(x) = \begin{cases} x \; if \; x > 0 \\ 0 \; otherwise \end{cases}$$

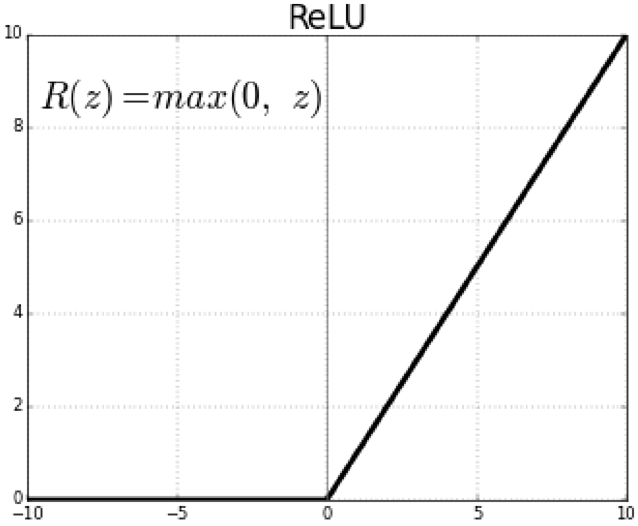
$$or$$

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

Research has shown that ReLUs result in much faster training for large networks. Most frameworks like Tensor Flow and TFLearn make it simple to use ReLUs on the hidden layers.

One of the disadvantages of Relu is that all the negative values become zero immediately which decreases the ability of the model to fit or train from the data properly. That means any negative input given to the ReLU activation function turns the value into zero immediately in the graph, which in turns affects the resulting graph by not mapping the negative values appropriately.

Dying Relu problem: Another problem of Relu is that a large gradient flowing through a ReLU neuron could cause the weights to update in such a way that the neuron will never activate on any data point again. If this happens, then the gradient flowing through the unit will forever be zero from that point on. That is, the ReLU units can irreversibly die during training since they can get knocked off the data manifold. For example, you may find that as much as 40% of your network can be "dead" (i.e. neurons that never activate across the entire training dataset) if the learning rate is set too high. With a proper setting of the learning rate this is less frequently an issue.



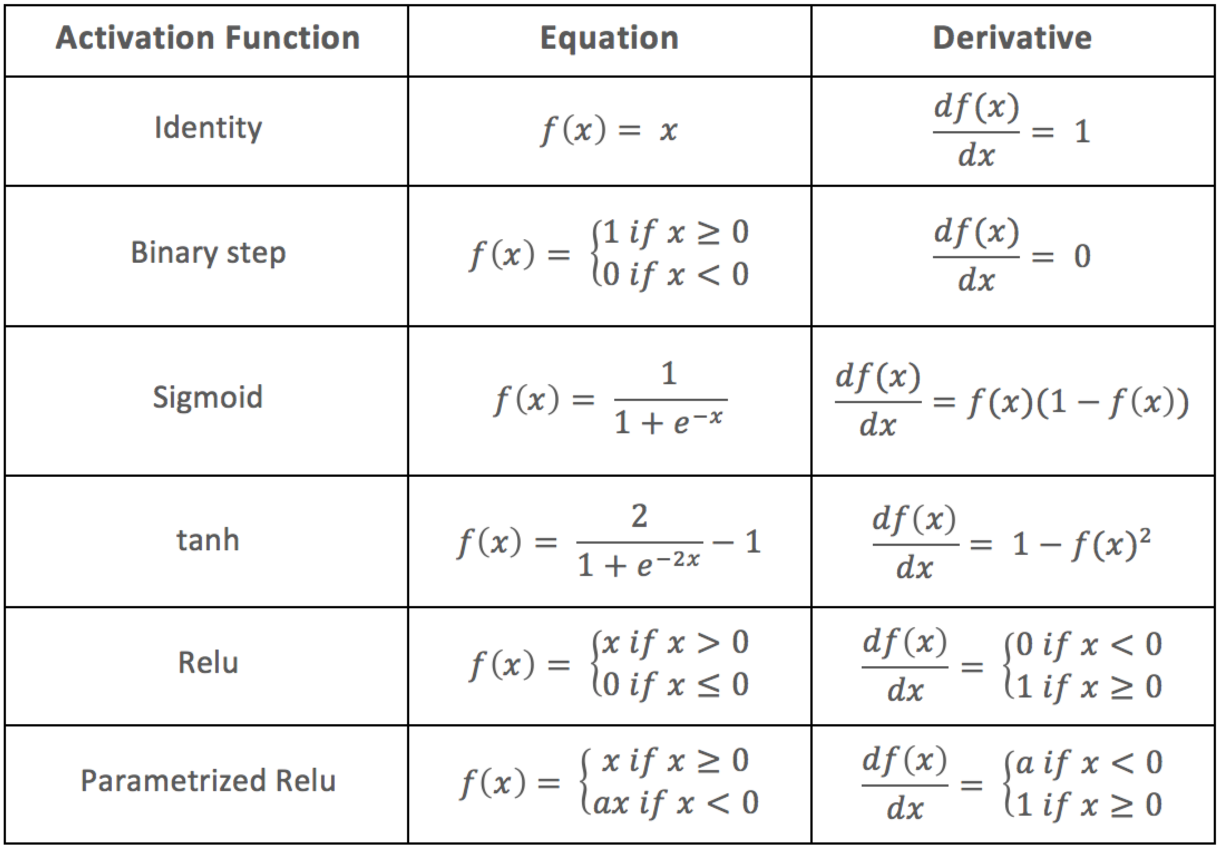
Leaky Relu:

To solve the dying Relu problem, leaky Relu is used. It is given by:

$$f(x) = \begin{cases} x \; if \; x > 0 \\ ax \; if \; x \leq 0 \end{cases}$$

For leaky rely, the value of a is usually 0.01. When a is not 0.01, it is called as parametrised relu.

To sum up, following are some of the activation functions along with their equations and their derivatives:



Activation Function	Equation	Derivative
Identity	$f(x) = x$	$\frac{df(x)}{dx} = 1$
Binary step	$f(x) = \begin{cases} 1\ if\ x \geq 0 \\ 0\ if\ x < 0 \end{cases}$	$\frac{df(x)}{dx} = 0$
Sigmoid	$f(x) = \frac{1}{1+e^{-x}}$	$\frac{df(x)}{dx} = f(x)(1-f(x))$
tanh	$f(x) = \frac{2}{1+e^{-2x}} - 1$	$\frac{df(x)}{dx} = 1 - f(x)^2$
Relu	$f(x) = \begin{cases} x\ if\ x > 0 \\ 0\ if\ x \leq 0 \end{cases}$	$\frac{df(x)}{dx} = \begin{cases} 0\ if\ x < 0 \\ 1\ if\ x \geq 0 \end{cases}$
Parametrized Relu	$f(x) = \begin{cases} x\ if\ x \geq 0 \\ ax\ if\ x < 0 \end{cases}$	$\frac{df(x)}{dx} = \begin{cases} a\ if\ x < 0 \\ 1\ if\ x \geq 0 \end{cases}$

Activation Function	**Equation**	**Derivative**
Identity	$f(x) = x$	$\frac{df(x)}{dx} = 1$
Binary step	$f(x) = \begin{cases} 1 \; if \; x \geq 0 \\ 0 \; if \; x < 0 \end{cases}$	$\frac{df(x)}{dx} = 0$
Sigmoid	$f(x) = \frac{1}{1 + e^{-x}}$	$\frac{df(x)}{dx} = f(x)(1 - f(x))$
tanh	$f(x) = \frac{2}{1 + e^{-2x}} - 1$	$\frac{df(x)}{dx} = 1 - f(x)^2$
Relu	$f(x) = \begin{cases} x \; if \; x > 0 \\ 0 \; if \; x \leq 0 \end{cases}$	$\frac{df(x)}{dx} = \begin{cases} 0 \; if \; x < 0 \\ 1 \; if \; x \geq 0 \end{cases}$
Parametrized Relu	$f(x) = \begin{cases} x \; if \; x \geq 0 \\ ax \; if \; x < 0 \end{cases}$	$\frac{df(x)}{dx} = \begin{cases} a \; if \; x < 0 \\ 1 \; if \; x \geq 0 \end{cases}$