

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



7 Types of Neural Network Activation Functions

How to Choose?

Neural Network Concepts

7 Types of Neural Network Activation Functions: How to Choose?

Neural network activation functions are a crucial component of deep learning. Activation functions determine the output of a deep learning model, its accuracy, and also the computational efficiency of training a model—which can make or break a large scale neural network. Activation functions also have a major effect on the neural network’s ability to converge and the convergence speed, or in some cases, activation functions might prevent neural networks from converging in the first place.

This article is part of MissingLink’s [Neural Network Guide](#), which focuses on practical explanations of concepts and

[Features](#)

[Resources](#)

[Blog](#)

[Company](#)

[Login](#)

[deep learning platform](#) to speed up your experiments.

In this article you will learn

What is a Neural Network Activation Function?

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



Seven common nonlinear activation functions and how to choose an activation function for your model—sigmoid, TanH, ReLU and more

Derivatives or gradients of common activation functions

How neural network activation functions are used in real world projects

What is a Neural Network Activation Function?

Activation functions are mathematical equations that determine the output of a neural network. The function is attached to each neuron in the network, and determines whether it should be activated (“fired”) or not, based on whether each neuron’s input is relevant for the model’s prediction. Activation functions also help normalize the output of each neuron to a range between 1 and 0 or between -1 and 1.

An additional aspect of activation functions is that they must be computationally efficient because they are calculated across thousands or even millions of neurons for each data sample. Modern neural networks use a technique called backpropagation to train the model, which places an increased computational strain on the activation function, and its

[Features](#)

[Resources](#)

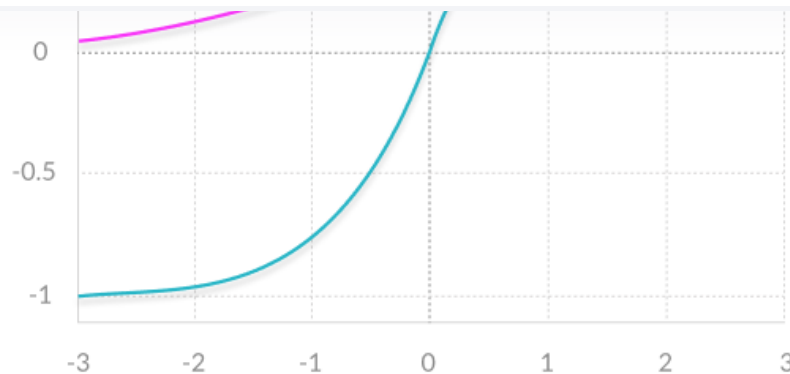
[Blog](#)

[Company](#)

[Login](#)

[activation functions](#) below).

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



Two common neural network activation functions - Sigmoid and Tanh

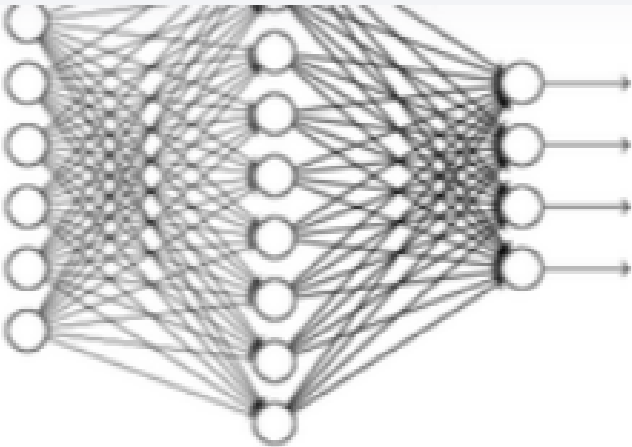
What are Artificial Neural Networks and Deep Neural Networks?

[Artificial Neural Networks](#) (ANN) are comprised of a large number of simple elements, called neurons, each of which makes simple decisions. Together, the neurons can provide accurate answers to some complex problems, such as natural language processing, computer vision, and AI.

A neural network can be “shallow”, meaning it has an input layer of neurons, only one “hidden layer” that processes the inputs, and an output layer that provides the final output of the model. A Deep Neural Network (DNN) commonly has between 2-8 additional layers of neurons. Research from [Goodfellow, Bengio and Courville](#) and other experts suggests that neural networks increase in accuracy with the number of hidden layers.

[Features](#)[Resources](#)[Blog](#)[Company](#)[Login](#)

Important announcement: Missinglink has shut down. [Click here to learn more.](#)

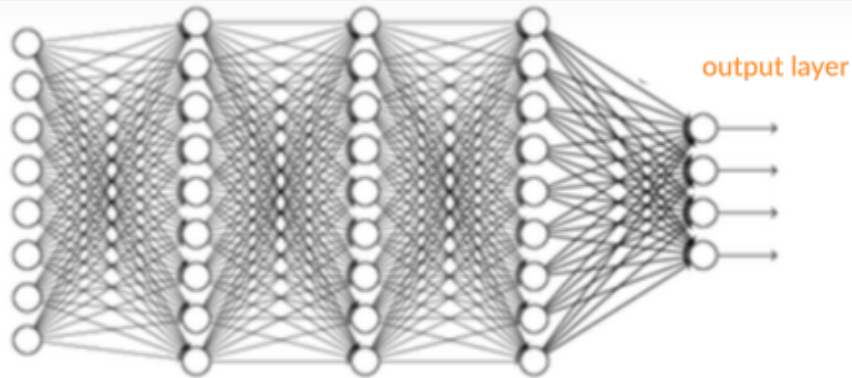


"Many-deep" feedforward neural network

Features Resources Blog Company

Login

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



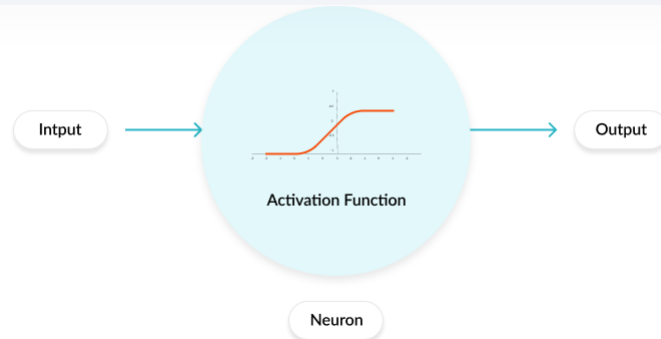
Deep neural network

— Features — Resources — Blog — Company —
Login

into the neurons in the input layer. Each neuron has a weight, and multiplying the input number with the weight gives the output of the neuron, which is transferred to the next layer.

The activation function is a mathematical “gate” in between the input feeding the current neuron and its output going to the next layer. It can be as simple as a step function that turns the

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



Increasingly, neural networks use non-linear activation functions, which can help the network learn complex data, compute and learn almost any function representing a question, and provide accurate predictions.

The basic process carried out by a neuron in a neural network is:



* This is just the number 1, making it possible to represent activation functions that do not cross the origin. [Biases](#) are

[Features](#)

[Resources](#)

[Blog](#)

[Company](#)

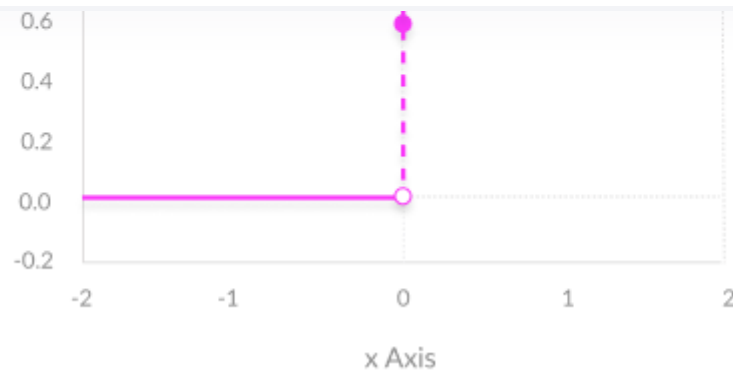
[Login](#)

5 Types of Activation Functions

Binary Step Function

A binary step function is a threshold-based activation function. If the input value is above or below a certain threshold, the neuron is activated and sends exactly the same signal to the next layer.

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



The problem with a step function is that it does not allow multi-value outputs—for example, it cannot support classifying the inputs into one of several categories.

Linear Activation Function

A linear activation function takes the form:

$$A = cx$$



It takes the inputs, multiplied by the weights for each neuron, and creates an output signal proportional to the input. In one

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



and has no relation to the input, X . So it's not possible to go back and understand which weights in the input neurons can provide a better prediction.

[Go in-depth: See our guide on backpropagation](#)

2. All layers of the neural network collapse into one—with linear activation functions, no matter how many layers in the neural network, the last layer will be a linear function of the first layer (because a linear combination of linear functions is still a linear function). So a linear activation function turns the neural network into just one layer.

A neural network with a linear activation function is simply a linear regression model. It has limited power and ability to handle complexity varying parameters of input data.

Non-Linear Activation Functions

Modern neural network models use non-linear activation functions. They allow the model to create complex mappings between the network's inputs and outputs, which are essential for learning and modeling complex data, such as images, video, audio, and data sets which are non-linear or have high dimensionality.

Almost any process imaginable can be represented as a functional computation in a neural network, provided that the activation function is non-linear.

Features

Resources

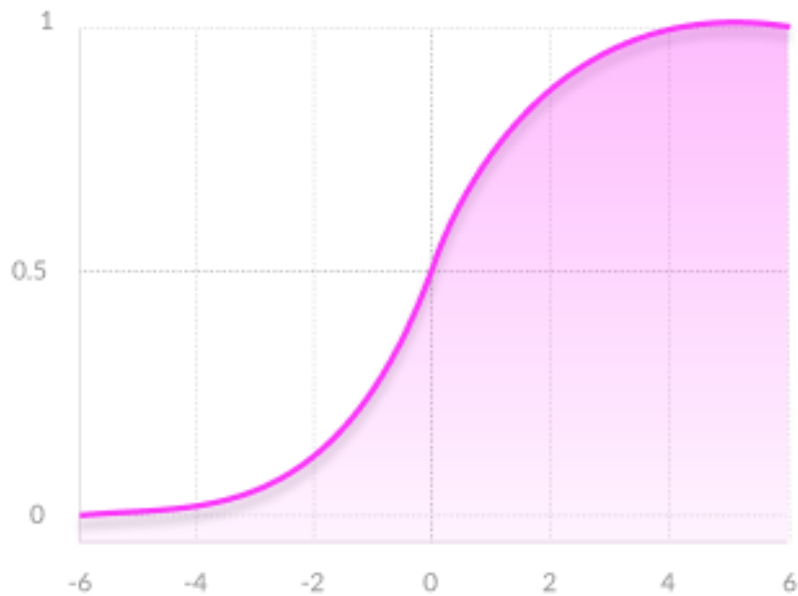
Blog

Company

Login

1. They allow backpropagation because they have a derivative function which is related to the inputs.
2. They allow “stacking” of multiple layers of neurons to create a deep neural network. Multiple hidden layers of neurons are needed to learn complex data sets with high levels of accuracy.

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



Sigmoid / Logistic

Advantages

Features

Resources

Blog

Company

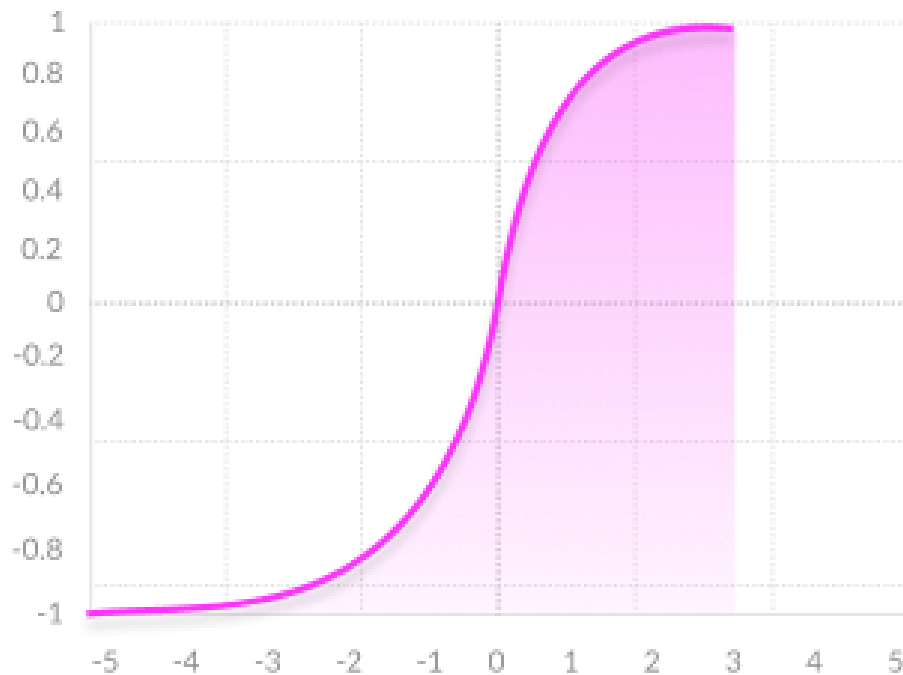
Login

- **Clear predictions**—For X above 2 or below -2, tends to bring the Y value (the prediction) to the edge of the curve, very close to 1 or 0. This enables clear predictions.

Disadvantages

- **Vanishing gradient**—for very high or very low values of

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



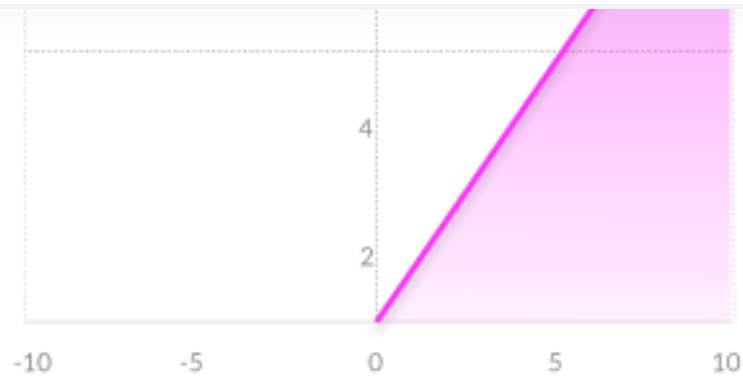
TanH / Hyperbolic Tangent

Advantages

- **Zero centered**—making it easier to model inputs that have strongly negative, neutral, and strongly positive values.
- Otherwise like the Sigmoid function
- Like the Sigmoid function

Features Resources Blog Company
Login

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



ReLU (Rectified Linear Unit)

Advantages

- **Computationally efficient**—allows the network to converge very quickly
- **Non-linear**—although it looks like a linear function, ReLU has a derivative function and allows for backpropagation

Disadvantages

- **The Dying ReLU problem**—when inputs approach zero, or are negative, the gradient of the function becomes zero, the network cannot perform backpropagation and

Features

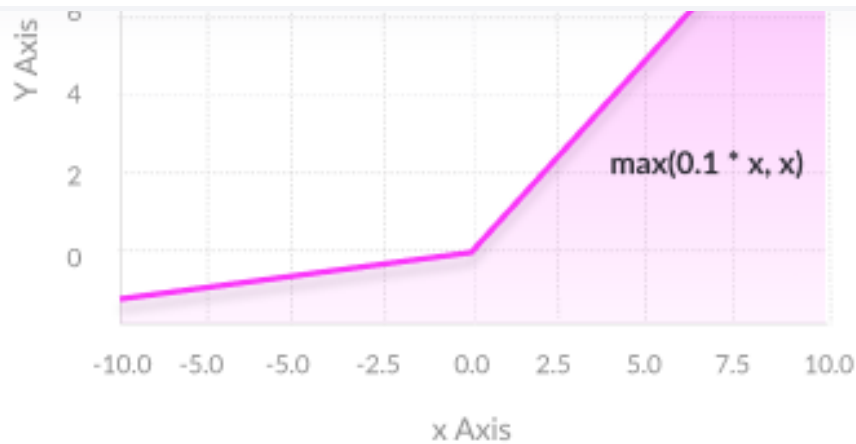
Resources

Blog

Company

Login

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



Leaky ReLU

Advantages

- **Prevents dying ReLU problem**—this variation of ReLU has a small positive slope in the negative area, so it does enable backpropagation, even for negative input values
- Otherwise like ReLU

Disadvantages

- **Results not consistent**—leaky ReLU does not provide

Features

Resources

Blog

Company

Login

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



Parametric ReLU

Advantages

- **Allows the negative slope to be learned**—unlike leaky ReLU, this function provides the slope of the negative part of the function as an argument. It is, therefore, possible to perform backpropagation and learn the most appropriate value of α .
- Otherwise like ReLU

Disadvantages

- **May perform differently** for different problems.

Features

Resources

Blog

Company

Login

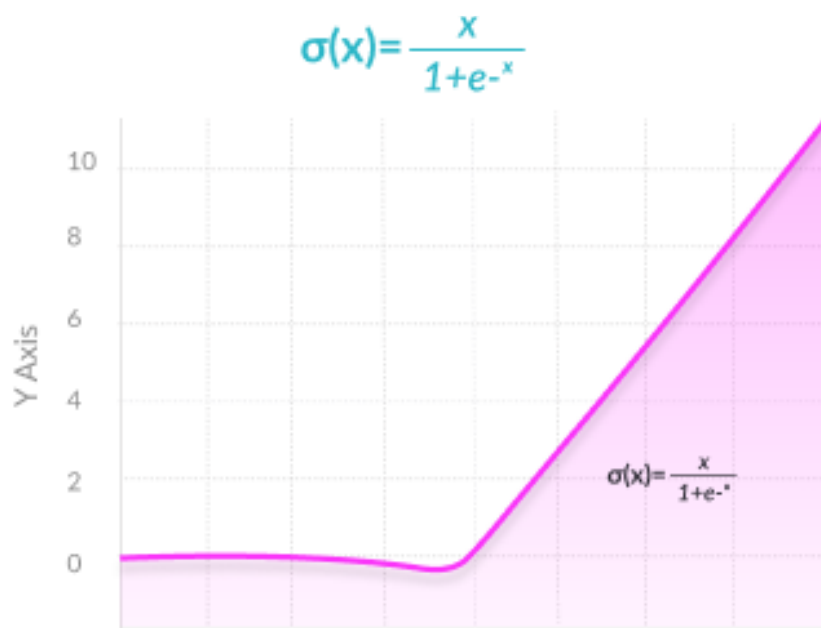
Softmax

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



class between 0 and 1, and divides by their sum, giving the probability of the input value being in a specific class.

- **Useful for output neurons**—typically Softmax is used only for the output layer, for neural networks that need to classify inputs into multiple categories.



Swish

Swish is a new, self-audited activation function discovered by Google. It is a smooth, non-monotonic function that is self-normalizing. It is a combination of the ReLU and Sigmoid functions. It is a smooth, non-monotonic function that is self-normalizing. It is a combination of the ReLU and Sigmoid functions.

During 2018 and 2019, the new function achieved top 1 classification accuracy 0.6-0.9% higher.

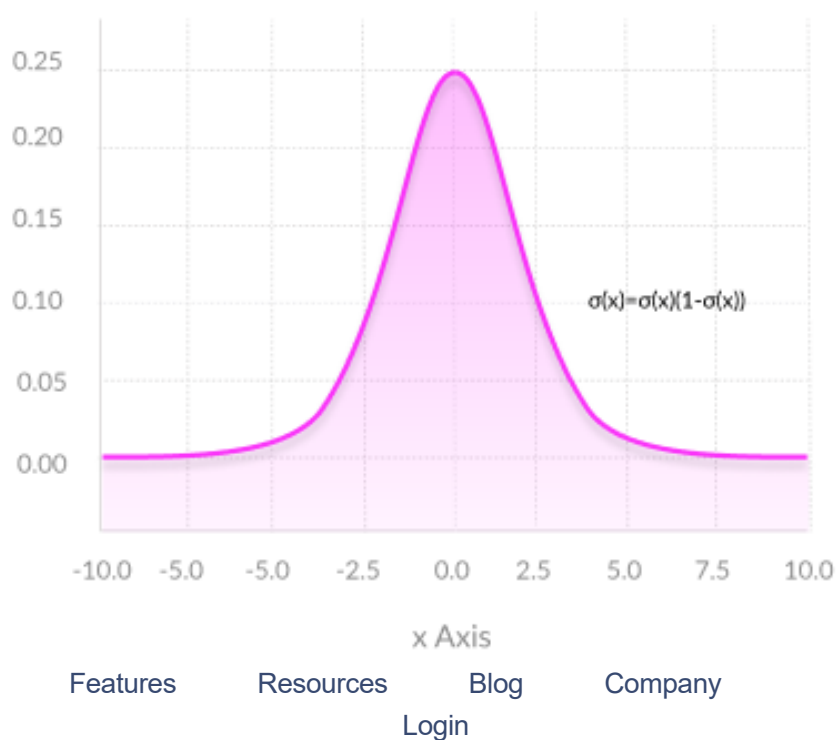
Important announcement: Missinglink has shut down. [Click here to learn more.](#)



the output of the model, through the different neurons which were involved in generating that output, back to the original weight applied to each neuron. Backpropagation suggests an optimal weight for each neuron which results in the most accurate prediction.

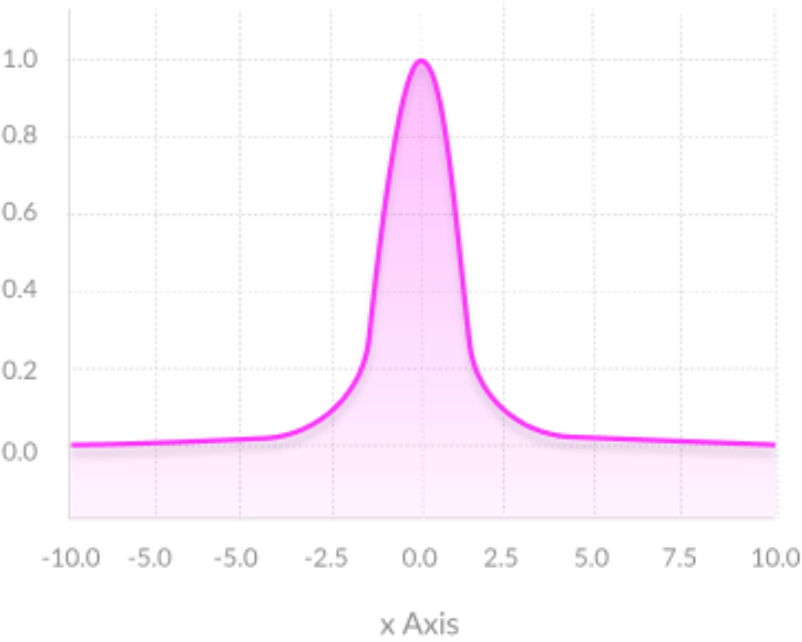
[Go in-depth: See our guide on backpropagation](#)

Below are the derivatives for the most common activation functions.



Sigmoid

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



Features

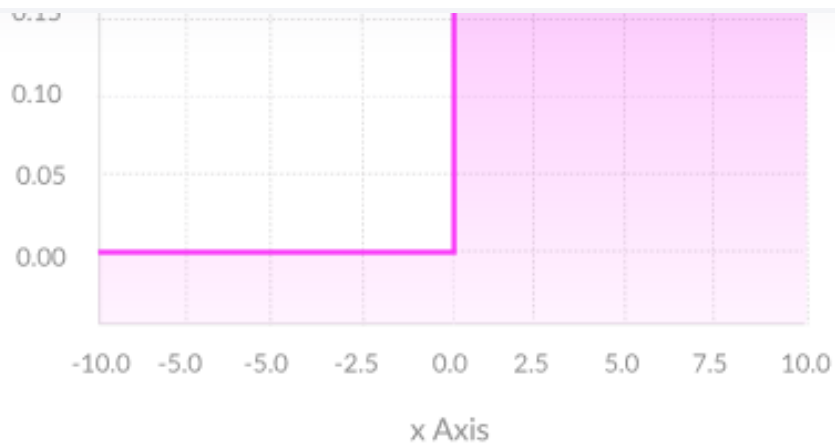
Resources

Blog

Company

Login

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



ReLU

Features

Resources

Blog

Company

Login

[Recent research](#) by Franco Manessi and Alessandro Rozza attempted to find ways to automatically learn which is the optimal activation function for a certain neural network and to even automatically combine activation functions to achieve the highest accuracy. This is a very promising field of research because it attempts to discover an optimal activation function

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



When building a model and training a neural network, the selection of activation functions is critical. Experimenting with different activation functions for different problems will allow you to achieve much better results.

In a real-world neural network project, you will switch between activation functions using the deep learning framework of your choice.

For example, here is how to use the ReLU activation function via the Keras library (see all [supported activations](#)):

```
keras.activations.relu(x, alpha=0.0,  
max_value=None)
```

While selecting and switching activation functions in deep learning frameworks is easy, you will find that managing multiple experiments and trying different activation functions on large test data sets can be challenging.

It can be difficult to:

[Features](#)[Resources](#)[Blog](#)[Company](#)[Login](#)

Track experiment progress source code, metrics and hyperparameters across different experiments trying different activation functions for a model, or variations of the same model.

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



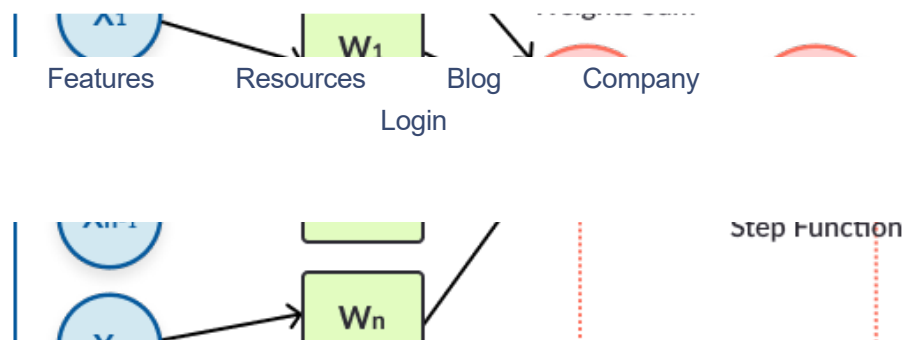
on several machines; you'll need to provision and maintain these machines.



Manage training data to achieve good results, you'll need to experiment with different sets of test data across multiple model variations on different machines. Moving the training data each time you need to run an experiment is difficult, especially if you are processing heavy inputs like images or video.

MissingLink can help you manage all this, as you experiment to find the best activation function for your model. [Learn more](#) to see how easy it is to run deep learning experiments at scale.

Learn More About Neural Network Concepts



The Complete Guide to Artificial Neural Networks: Concepts and Models

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



Backpropagation in Neural Networks: Process, Example & Code

Hyperparameters: Optimization Methods and Real World Model Management

Features

Resources

Blog

Company

Login

Convolutional Neural Network: How to Build One in Keras & PyTorch

Important announcement: Missinglink has shut down. [Click here to learn more.](#)



Resources

[Blog](#)

[Product Features](#)

[Docs](#)

Company

[About us](#)

[Join the team](#)

[Contact us](#)

[Privacy Policy](#)

[Terms of Service](#)

Solutions

[Deep Learning on AWS](#)

[Deep Learning on Google Cloud](#)

Important announcement: Missinglink has shut down. [Click here to learn more.](#)

