A neural network is a machine learning ([ML](https://www.techtarget.com/searchenterpriseai/definition/machine-learning-ML)) model designed to mimic the function and structure of the human brain. Neural networks are intricate networks of interconnected nodes, or neurons, that collaborate to tackle complicated problems.

Also referred to as artificial neural networks (ANNs) or deep neural networks, neural networks represent a type of deep learning technology that's classified under the broader field of artificial intelligence ([AI](https://www.techtarget.com/searchenterpriseai/definition/AI-Artificial-Intelligence)).

### Artificial Neural Networks

Artificial Neural Networks contain artificial neurons which are called **units**.

These units are arranged in a series of layers that together constitute the whole Artificial Neural Network in a system.

A layer can have only a dozen units or millions of units as this depends on how the complex neural networks will be required to learn the hidden patterns in the dataset.

Commonly, Artificial Neural Network has an input layer, an output layer as well as hidden layers. The input layer receives data from the outside world which the neural network needs to analyze or learn about.

Then this data passes through one or multiple hidden layers that transform the input into data that is valuable for the output layer. Finally, the output layer provides an output in the form of a response of the Artificial Neural Networks to input data provided.

In the majority of neural networks, units are interconnected from one layer to another.

Each of these connections has weights that determine the influence of one unit on another unit.

As the data transfers from one unit to another, the neural network learns more and more about the data which eventually results in an output from the output layer.



*Neural Networks Architecture*

### ****Artificial neurons vs Biological neurons****

The concept of artificial neural networks comes from biological neurons found in animal brains So they share a lot of similarities in structure and function wise.

* **Structure**: The structure of artificial neural networks is inspired by biological neurons. A biological neuron has a cell body or soma to process the impulses, dendrites to receive them, and an axon that transfers them to other neurons.  The input nodes of artificial neural networks receive input signals, the hidden layer nodes compute these input signals, and the output layer nodes compute the final output by processing the hidden layer’s results using activation functions.

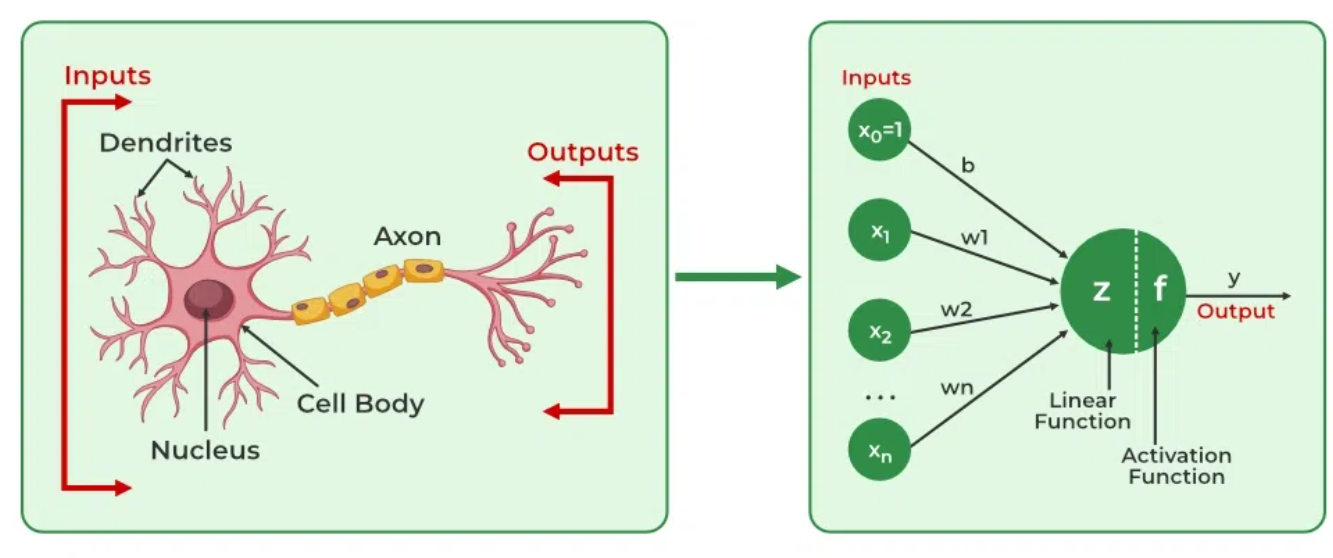
| **Biological Neuron** | **Artificial Neuron** |
| --- | --- |
| Dendrite | Inputs |
| Cell nucleus or Soma | Nodes |
| Synapses | Weights |
| Axon | Output |

* **Synapses**: Synapses are the links between biological neurons that enable the transmission of impulses from dendrites to the cell body. Synapses are the weights that join the one-layer nodes to the next-layer nodes in artificial neurons. The strength of the links is determined by the weight value.
* **Learning**: In biological neurons, learning happens in the cell body nucleus or soma, which has a nucleus that helps to process the impulses. An action potential is produced and travels through the axons if the impulses are powerful enough to reach the threshold.

This becomes possible by synaptic plasticity, which represents the ability of synapses to become stronger or weaker over time in reaction to changes in their activity. In artificial neural networks, backpropagation is a technique used for learning, which adjusts the weights between nodes according to the error or differences between predicted and actual outcomes.

| **Biological Neuron** | **Artificial Neuron** |
| --- | --- |
| Synaptic plasticity | Backpropagations |

* **Activation**: In biological neurons, activation is the firing rate of the neuron which happens when the impulses are strong enough to reach the threshold. In artificial neural networks, A mathematical function known as an activation function maps the input to the output, and executes activations.

**

### How do Artificial Neural Networks learn?

Artificial neural networks are trained using a training set.

For example, suppose you want to teach an ANN to recognize a cat. Then it is shown thousands of different images of cats so that the network can learn to identify a cat.

Once the neural network has been trained enough using images of cats, then you need to check if it can identify cat images correctly.

This is done by making the ANN classify the images it is provided by deciding whether they are cat images or not.

The output obtained by the ANN is corroborated by a human-provided description of whether the image is a cat image or not.

If the ANN identifies incorrectly then [back-propagation](https://www.geeksforgeeks.org/deep-neural-net-with-forward-and-back-propagation-from-scratch-python/) is used to adjust whatever it has learned during training.

[Backpropagation](https://www.geeksforgeeks.org/backpropagation-in-data-mining/) is done by fine-tuning the weights of the connections in ANN units based on the error rate obtained.

This process continues until the artificial neural network can correctly recognize a cat in an image with minimal possible error rates.

### Applications of Artificial Neural Networks

1. **Social Media:**Artificial Neural Networks are used heavily in Social Media. For example, let’s take the **‘People you may know’** feature on Facebook that suggests people that you might know in real life so that you can send them friend requests.
2. Another common application of [Machine Learning](https://www.geeksforgeeks.org/machine-learning/) in social media is **facial recognition**. This is done by finding around 100 reference points on the person’s face and then matching them with those already available in the database using convolutional neural networks.
3. **Marketing and Sales:**When you log onto E-commerce sites like Amazon and Flipkart, they will recommend your products to buy based on your previous browsing history. Similarly, suppose you love Pasta, then Zomato, Swiggy, etc. will show you restaurant recommendations based on your tastes and previous order history. This is true across all new-age marketing segments like Book sites, Movie services, Hospitality sites, etc. and it is done by implementing **personalized marketing**. This uses Artificial Neural Networks to identify the customer likes, dislikes, previous shopping history, etc., and then tailor the marketing campaigns accordingly.
4. **Healthcare**: Artificial Neural Networks are used in Oncology to train algorithms that can identify cancerous tissue at the microscopic level at the same accuracy as trained physicians. Various rare diseases may manifest in physical characteristics and can be identified in their premature stages by using **Facial Analysis** on the patient photos. So the full-scale implementation of Artificial Neural Networks in the healthcare environment can only enhance the diagnostic abilities of medical experts and ultimately lead to the overall improvement in the quality of medical care all over the world.
5. **Personal Assistants:**I am sure you all have heard of Siri, Alexa, Cortana, etc., and also heard them based on the phones you have!!! These are personal assistants and an example of speech recognition that uses **Natural Language Processing** to interact with the users and formulate a response accordingly. Natural Language Processing uses artificial neural networks that are made to handle many tasks of these personal assistants such as managing the language syntax, semantics, correct speech, the conversation that is going on, etc.

* [Chatbots](https://www.techtarget.com/searchcustomerexperience/definition/chatbot).
* Stock market predictions.
* Delivery driver route planning and optimization.
* Drug discovery and development.

## Advantages of artificial neural networks

Artificial neural networks offer the following benefits:

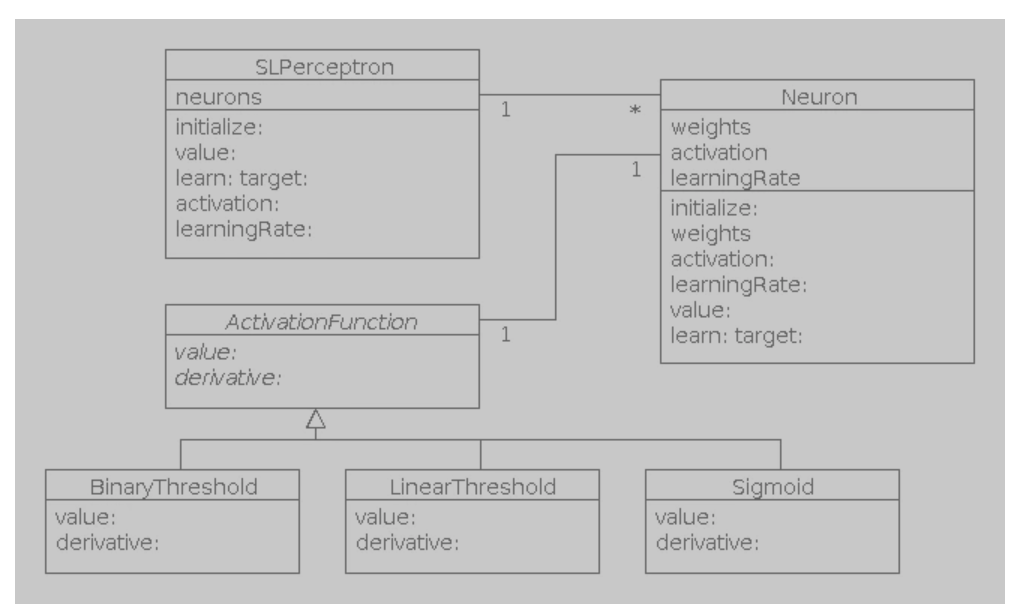
* **Parallel processing abilities.** ANNs have parallel processing abilities, which means the network can perform more than one job at a time.
* **Information storage.** ANNs store information on the entire network, not just in a database. This ensures that even if a small amount of data disappears from one location, the entire network continues to operate.
* **Non-linearity.** The ability to learn and model nonlinear, complex relationships helps model the real-world relationships between input and output.
* **Fault tolerance.** ANNs come with [fault tolerance](https://www.techtarget.com/searchdisasterrecovery/definition/fault-tolerant), which means the corruption or fault of one or more cells of the ANN won't stop the generation of output.
* **Gradual corruption.** This means the network slowly degrades over time instead of degrading instantly when a problem occurs.
* **Unrestricted input variables.** No restrictions are placed on the input variables, such as how they should be distributed.
* **Obsevation-based decisions.** Machine learning means the ANN can learn from events and make decisions based on the observations.
* **Unorganized data processing.** Artificial neural networks are exceptionally good at organizing large amounts of data by processing, sorting and categorizing it.
* **Ability to learn hidden relationships.** ANNs can learn the hidden relationships in data without commanding any fixed relationship. This means ANNs can better model highly volatile data and non-constant variance.
* **Ability to generalize data.** The ability to generalize and infer unseen relationships on unseen data means ANNs can predict the output of unseen data.

## Disadvantages of artificial neural networks

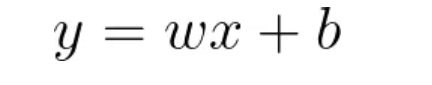
Along with their numerous benefits, neural networks also have some drawbacks, including the following:

* **Lack of rules.** The lack of rules for determining the proper network structure means the appropriate artificial neural network architecture can only be found through trial, error and experience.
* **Hardware dependency.** The requirement of processors with parallel processing abilities makes neural networks dependent on hardware.
* **Numerical translation.** The network works with numerical information, meaning all problems must be translated into numerical values before they can be presented to the ANN.
* **Lack of trust.** The lack of explanation behind probing solutions is one of the biggest disadvantages of ANNs. The inability to explain the why or how behind the solution generates a lack of trust in the network.
* **Inaccurate results.** If not trained properly, ANNs can often produce incomplete or inaccurate results.
* **Black box nature.** Because of their [black box AI](https://www.techtarget.com/whatis/definition/black-box-AI) model, it can be challenging to grasp how neural networks make their predictions or categorize data.
* A single layer perceptron (SLP) is a feed-forward network based on a threshold transfer function. SLP is the simplest type of artificial neural networks and can only classify linearly separable cases with a binary target. 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 or not, based on whether each neuron’s input is relevant for the model’s prediction.

## How It Works

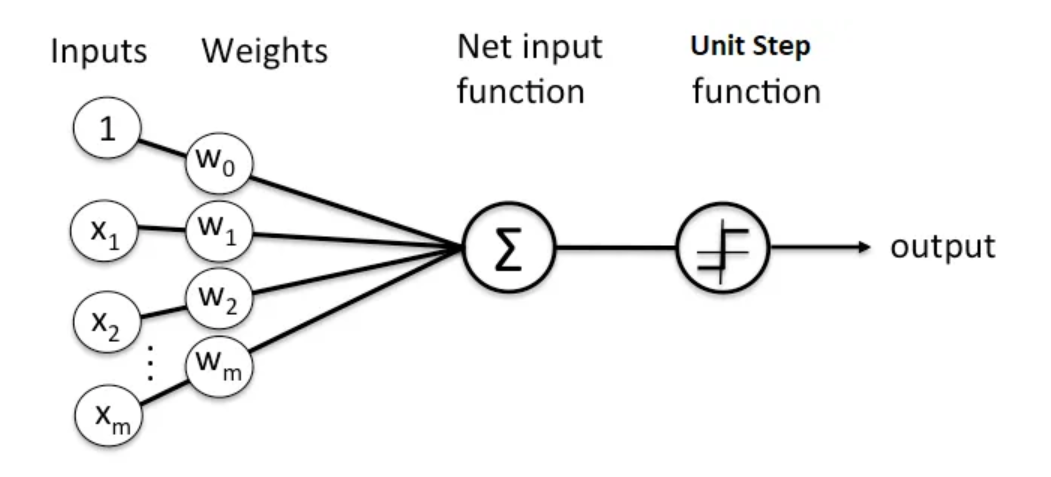


A perceptron is a linear classifier; that is, it is an algorithm that classifies input by separating two categories with a straight line. Input is typically a feature vector **x** multiplied by weights **w** and added to a bias **b** :

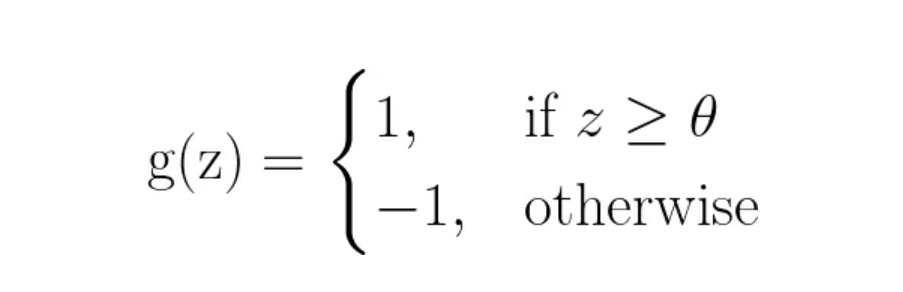


A single-layer perceptron does not include hidden layers, which allow neural networks to model a feature hierarchy. It is, therefore, a shallow neural network, which prevents it from performing non-linear classification.

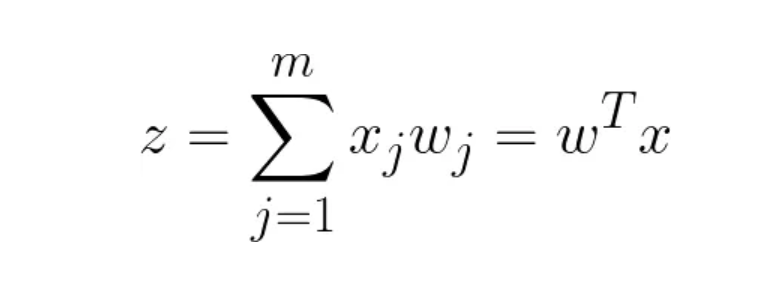
## Training using Unit Step Function



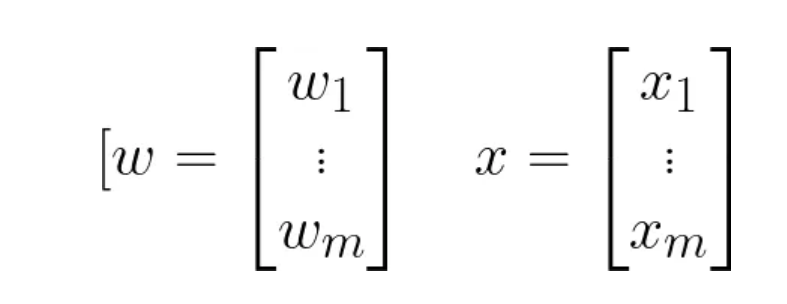
* label the positive and negative class in our binary classification setting as **1** and **-1**
* linear combination of the input values **x** and weights **w** as **(z=w₁x₁+⋯+wₘxₘ)**
* define an activation function **g(z)**, where if **g(z)** is greater than a defined threshold **θ** we predict **1** and **-1** otherwise; in this case, this activation function **g** is an alternative form of a simple **Unit step function**, which is sometimes also called **Heaviside step function**



Summarizing , we end up with :

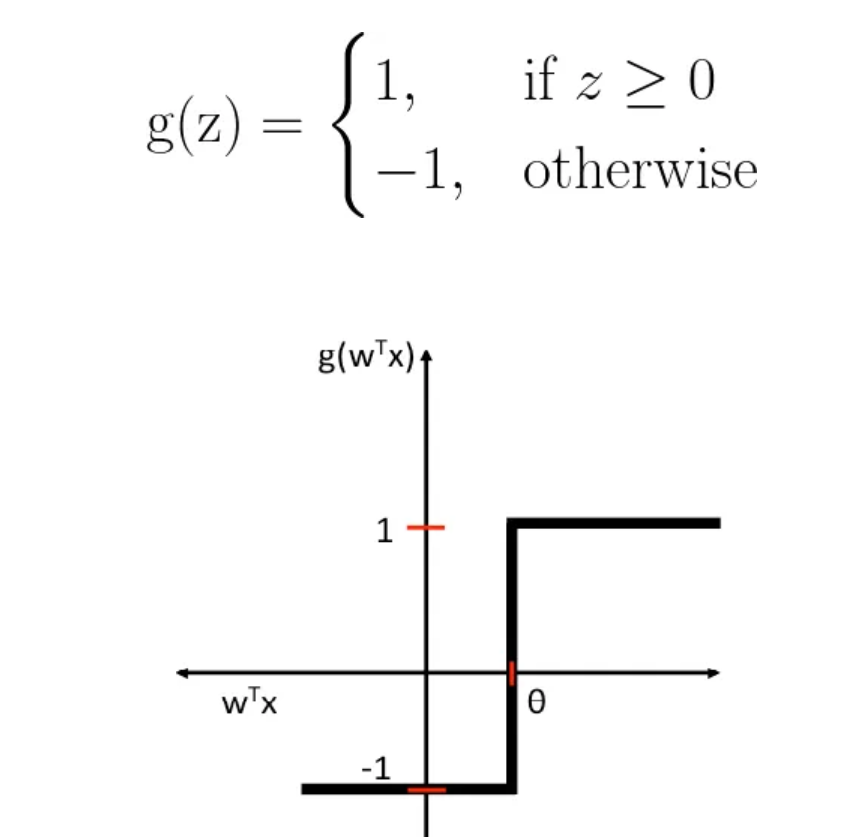


where,



* **w** is the feature (weight) vector,
* **x** is an **m**-dimensional sample from the training dataset:

In order to simplify the notation, we bring **θ** to the left side of the equation and define **w₀=−θ** and **x₀=1** (also known as **bias**)



## Learning rule

Initial perceptron rule is fairly simple and can be summarized by the following steps:

1. Initialize the weights to **0** or small random numbers.
2. For each training sample **xⁱ**: calculate the output value and update the weights.
3. The output value is the class label predicted by the unit step function that we defined earlier and the weight update can be written more formally as **wⱼ = wⱼ + Δwⱼ.**
4. The value for updating the weights at each increment is calculated by the learning rule: **Δwⱼ = η(targetⁱ − outputⁱ) xⱼⁱ**  
   where **η** is the learning rate (a constant between **0.0** and **1.0**), **target** is the true class label, and the **output** is the predicted class label.
5. All weights in the weight vector are being updated simultaneously

The convergence of the perceptron is only guaranteed if the two classes are linearly separable. If the two classes can’t be separated by a linear decision boundary, we can set a maximum number of passes over the training dataset **epochs** and /or a **threshold** for the number of tolerated misclassifications.