**Understanding neural networks**

An **artificial neural network** (**ANN**) models the relationship between a set of input

signals and an output signal using a model derived from our understanding of how a

biological brain responds to stimuli from sensory inputs. Just like a brain uses a network

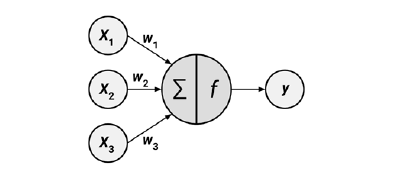
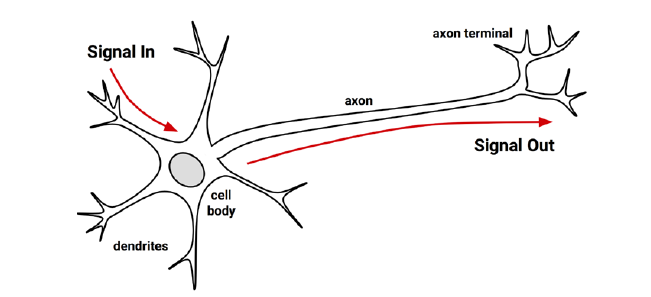
of interconnected cells called **neurons** to provide vast learning capability, the ANN uses

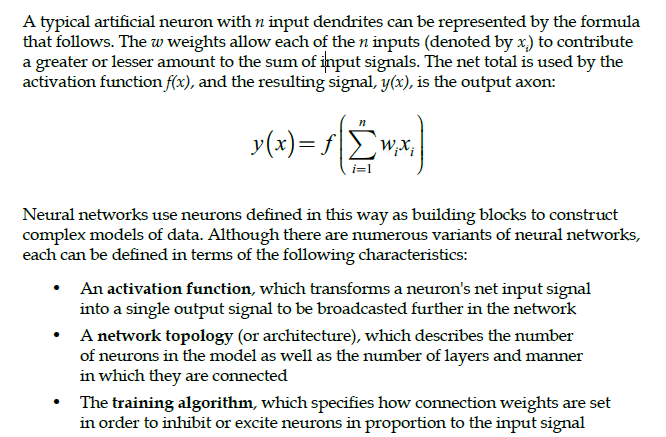
a network of artificial neurons or **nodes** to solve challenging learning problems.

**Turing test**, proposed in 1950 by the pioneering computer scientist Alan Turing,

which grades a machine as intelligent if a human being cannot distinguish its

behaviour from a living creature.

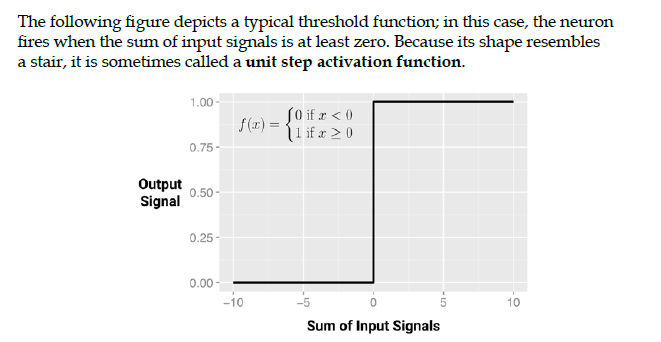




**Activation functions**

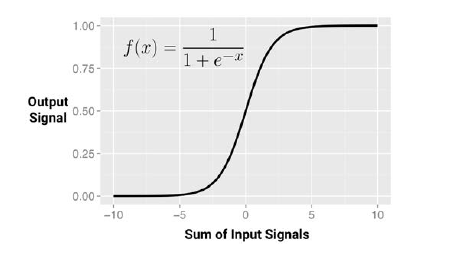
The activation function is the mechanism by which the artificial neuron processes

incoming information and passes it throughout the network. the activation function could be imagined as a process that involves summing the total input signal and determining whether it meets the firing threshold. If so, the neuron passes on the signal; otherwise, it does nothing. In ANN terms, this is known as a **threshold activation function**, as it results in an output signal only once a specified input threshold has been attained.

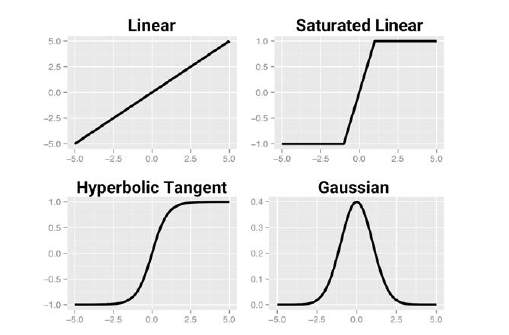


The threshold activation function is rarely used in ANNs.

Perhaps the most commonly used alternative is the **sigmoid activation function**



The choice of activation function biases the neural network such that it may fit certain types of data more appropriately, allowing the construction of specialized neural networks.



A linear activation function results in a neural network very similar to a linear regression model, while a Gaussian activation function is the basis of a **radial basis function** (**RBF**) **network**. Each of these has strengths better suited for certain learning tasks and not others.

Because this essentially squeezes the input values into a smaller range of outputs, activation

functions like the sigmoid are sometimes called **squashing functions**.

One solution to the squashing problem is to transform all neural network inputs

such that the feature values fall within a small range around zero. This may involve

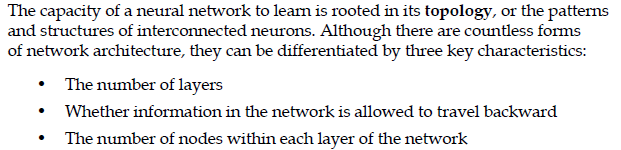
standardizing or normalizing the features. By restricting the range of input values,

the activation function will have action across the entire range. A side benefit is that

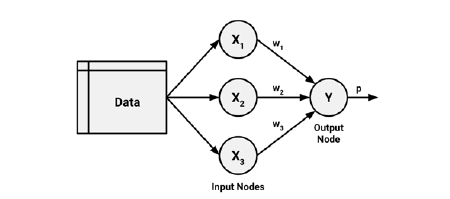
the model may also be faster to train, since the algorithm can iterate more quickly

through the actionable range of input values.

**Network topology**

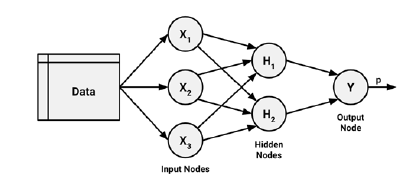


**The number of layers**

**Single-layer networks** can be used for basic pattern classification, particularly for patterns that are linearly separable, but more sophisticated networks are required for most learning tasks.  


Most **multilayer networks** are **fully connected**, which means that every node

in one layer is connected to every node in the next layer, but this is not required.

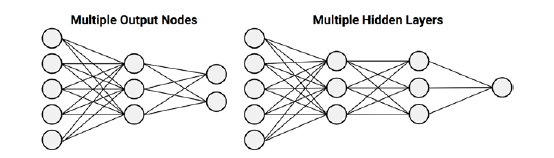




**The direction of information travel**

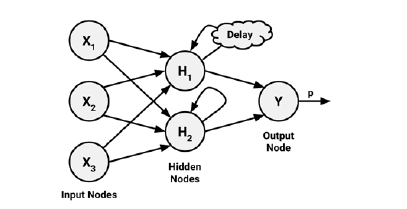
Networks in which the input signal is fed continuously in one direction from the input layer to the output layer are called **feedforward networks**.

A neural network with multiple hidden layers is called a **deep neural network** (**DNN**), and the practice of training such networks is referred to as **deep learning**. Deep neural networks trained on large datasets are capable of human-like performance on complex tasks like image recognition and text processing.



In contrast to feedforward networks, a **recurrent network** (or **feedback network**) allows signals to travel backward using loops.

The addition of a short-term memory, or **delay**, increases the power of recurrent networks immensely.



the multilayer feedforward network, also known as the **multilayer perceptron** (**MLP**), is the de facto standard ANN topology.

**The number of nodes in each layer**

Neural networks can also vary in complexity by the number of nodes in each

layer. The number of input nodes is predetermined by the number of features in the

input data. Similarly, the number of output nodes is predetermined by the number

of outcomes to be modelled or the number of class levels in the outcome. However,

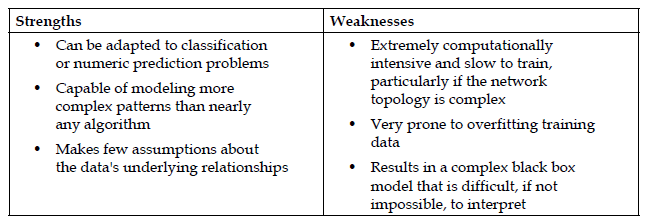
the number of hidden nodes is left to the user to decide prior to training the model.

**Training neural networks with backpropagation**

The algorithm, which used a strategy of back-propagating errors, is now known

simply as **backpropagation**.

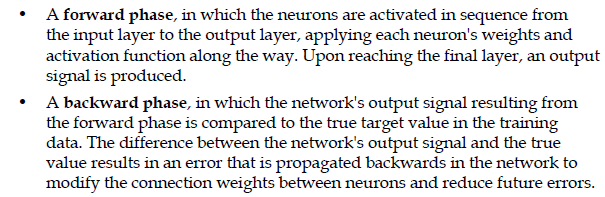
Multilayer feedforward networks that use the backpropagation algorithm are now common in the field of data mining. Such models offer the following strengths and weaknesses:



In its most general form, the backpropagation algorithm iterates through many

cycles of two processes. Each cycle is known as an **epoch**.

Each epoch in the backpropagation algorithm includes:



Because the relationship between each neuron's inputs and outputs is complex, the algorithm determines how much a weight should be changed by a technique called **gradient descent**.

In a similar process, the backpropagation algorithm uses the derivative of each

neuron's activation function to identify the gradient in the direction of each of the

incoming weights—hence the importance of having a differentiable activation

function. The gradient suggests how steeply the error will be reduced or increased

for a change in the weight. The algorithm will attempt to change the weights that

result in the greatest reduction in error by an amount known as the **learning rate**.

The greater the learning rate, the faster the algorithm will attempt to descend down

the gradients, which could reduce training time at the risk of overshooting the valley.