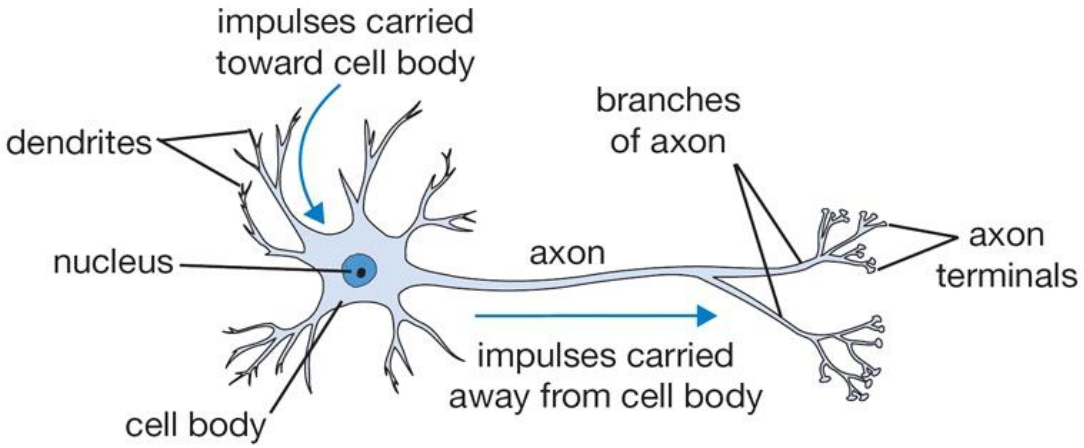
Artificial Neural network

Neural networks reflect the behavior of the human brain, allowing computer programs to recognize patterns and solve common problems in the fields of AI, machine learning, and deep learning. Their name and structure are inspired by the human brain, mimicking the way that biological neurons signal to one another.

Components of a neuron

The components of a neuron .In doing so follow the way of electrical information takes within neuron. The dendrites of a neuron received information by special connections called synapses.



Synapses weight the individual parts of information

1. The majority of ***neurons*** encode their outputs or activations as a series of brief electrical pulses (i.e. spikes or action potentials).
2. ***Dendrites*** are the receptive zones that receive activation from other neurons. Branch like trees from the cell nucleus of the neuron (which is called *soma*) and receive electrical signals from many different sources, which are then transferred into the nucleus of the cell. The amount of branching dendrites is also called ***dendrite tree***
3. The ***cell body (soma)*** of the neuron's processes the incoming activations and converts them into output activations.
4. ***Axons:*** are transmission lines that send activation to other neurons.
5. ***Synapses:*** allow weighted transmission of signals (using ***neurotransmitters***) between axons and dendrites to build up large neural networks In spite of the more complex functioning; the chemical synapse has compared with the electrical synapse utmost advantages:

After the cell nucleus (***soma***) has received a plenty of activating (=stimulating) and inhibiting (=diminishing) signals by synapses or dendrites, the soma accumulates these signals. As soon as the

accumulated signal exceeds a certain value (called threshold value), the cell nucleus of the neuron activates an electrical pulse which then is transmitted to the neurons connected to the current one .

Transition to technical neurons: neural networks are a caricature of biology

How change from biological neural networks to the technical ones? Through radical simplification. To briefly summarize the conclusions relevant for the technical part: they take over into technical approximation:

Vector input: The input of technical neurons consists of many components, therefore it is a *vector*. In nature a neuron receives pulses of 10^3 t0 10^4 other neurons on average.

Scalar output: The output of a neuron is a scalar, which means that the neuron only consists of one component. Several scalar outputs in turn form the vector input of another neuron. This particularly means that somewhere in the neuron the various input components have to be summarized in such a way that only one component remains.

Synapses change input: In technical neural networks the inputs are preprocessed, too. They are multiplied by a number they are *weighted*. The set of such weights represents the information storage of a neural network in both biological original and technical adaptation.

Accumulating the inputs: In biology, the inputs are summarized to a pulse according to the chemical change, i.e., they are accumulated – on the technical side this is often realized by the weighted sum, which we will get to know later on. This means that after accumulation we continue with only *one* value, a scalar, instead of a vector.

Non-linear characteristic: The input of our technical neurons is also not proportional to the output.

Adjustable weights: The weights weighting the inputs are variable, similar to the chemical processes at the synaptic cleft. This adds a great dynamic to the network because a large part of the "knowledge" of a neural network is saved in the weights and in the form and power of the chemical processes in a synaptic cleft So our current, only casually formulated and very simple neuron model receives a vector input bits of information

Artificial neural networks (ANNs)

An artificial network consists of a pool of simple processing units which communicate by sending signals to each other over a large number of weighted connections. A set of major aspects of a parallel distributed model can be distinguished.

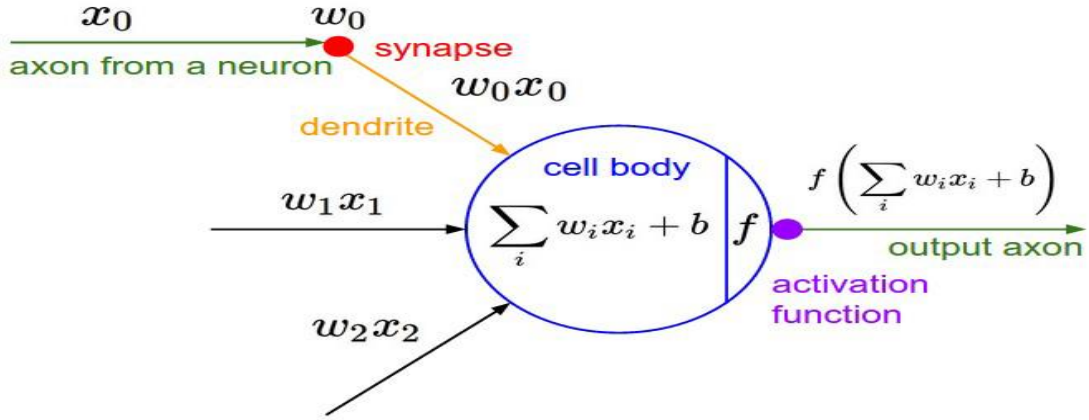
- A set of processing units (neurons, cells).
- A state of activation y_k for every unit_ which equivalent to the output of the unit.

- Connections between the units. Generally each connection is defined by a weight w_{jk} which determines the effect which the signal of unit j has on unit k.

- A propagation rule, which determines the effective input s_k of a unit from its external inputs.

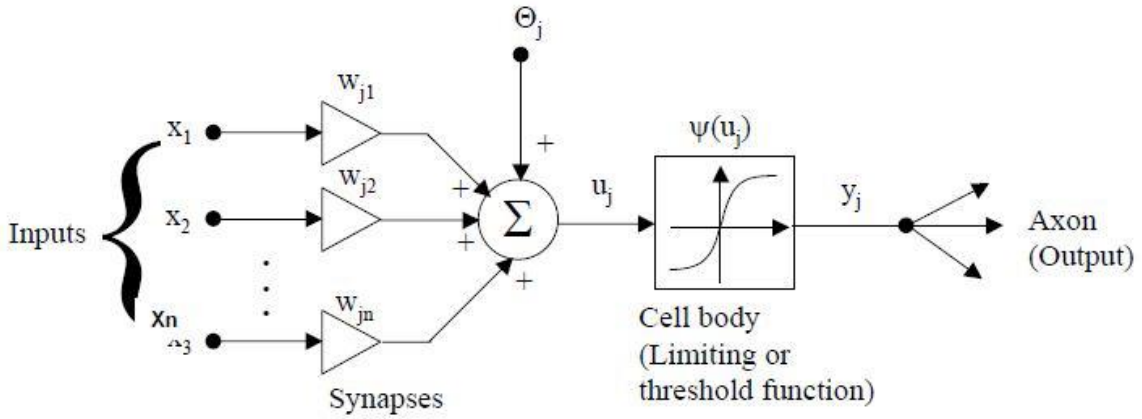
  An activation function F_k, which determines the new level of activation based on the

- Effective input $s_k(t)$ and the current activation $y_k(t)$ (i.e… the update).
- An external input (aka ,bias, offset) k for each unit.
- A method for information gathering (the learning rule).
- An environment within which the system must operate_ providing input signals and(if necessary)error signals.



Artificial Neural Network Compound

Artificial neural networks (ANNs) are comprised of node layers, containing an input layer, one or more hidden layers, and an output layer. Each node or artificial neuron connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value(Activate function) , that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network.



Neural networks rely on training data to learn and improve their accuracy over time. However, once these learning algorithms are fine-tuned for accuracy allowing us to classify and cluster data at a high velocity.

How do neural networks work?

The individual node as its own linear regression model, composed of input data, weights, a bias (or threshold), and an output. The formula would look something like this:

$$S_k(t) = \sum_{i=1}^{m} w_i x_i + bais = w_1 x_1 + w_2 x_2 + w_3 x_3 + bais$$

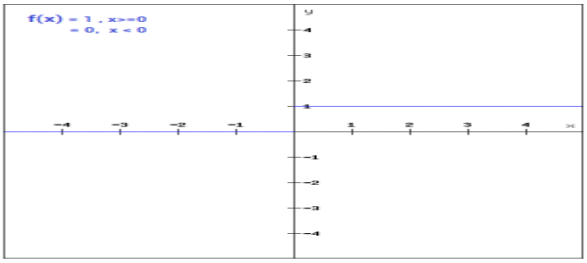
$$output = \begin{cases} 1 & if \sum w_i x_i + bais \geq 0 \\ 0 & if \sum w_i x_i + bais < 0 \end{cases}$$

Once an input layer is determined, weights are assigned. These weights help determine the importance of any given variable. All inputs are then multiplied by their respective weights and then summed.

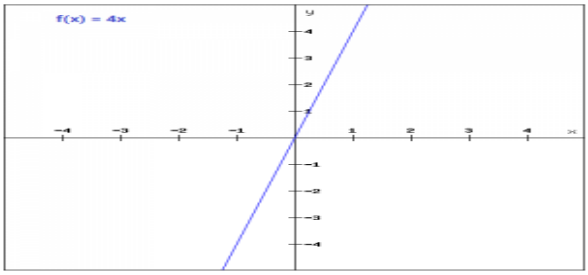
Activation and output rules

It is used to determine the output of neural network like yes or no. It maps the resulting values in between 0 to 1 or -1 to 1 etc. (depending upon the function). The basically Activation Functions are the following –

• ***Binary Step Function*** *:* if the input to the activation function is greater than a threshold, then the neuron is activated, else it is deactivated, i.e. its output is not considered for the next hidden layer. Let us look at it mathematically

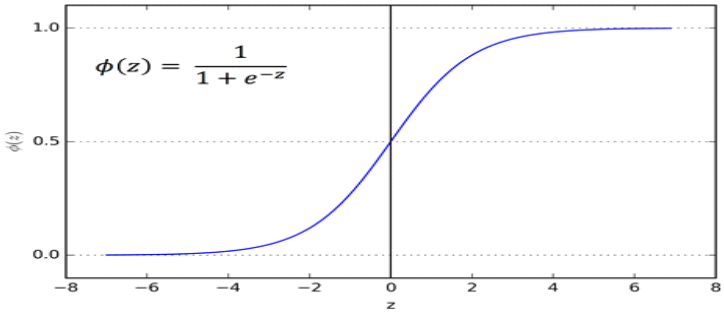


• ***Linear Function****:* the gradient of the function became zero. This is because there is no component of x in the binary step function. Instead of a binary function, we can use a linear function. We can define the function as: *f(x)=ax*



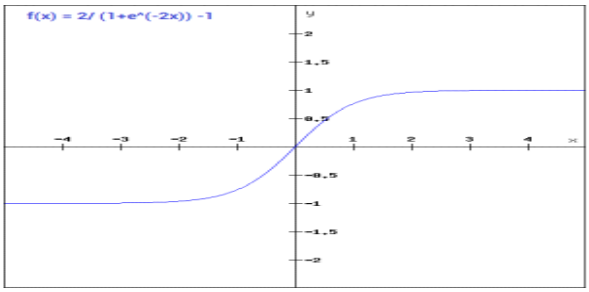
• ***Sigmoid****:* It is one of the most widely used non-linear activation function. Sigmoid transforms the values between the range 0 and 1. Here is the mathematical expression for sigmoid:

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

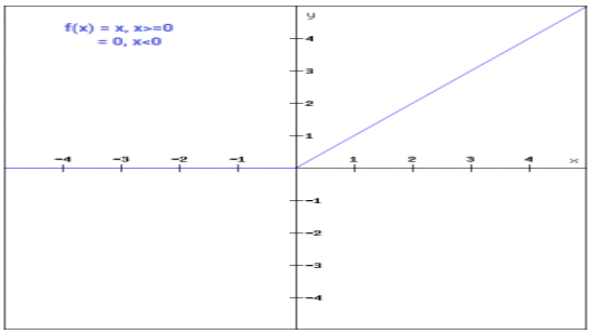


4.Tanh *:* is very similar to the sigmoid function. The only difference is that it is symmetric around the origin. The range of values in this case is from -1 to 1. Thus the inputs to the next layers will not always be of the same sign. The tanh function is defined as:

$$\tanh(x) = \frac{2}{sigmoid\ (2x) - 1}$$



5. ReLU : The ReLU function is another non-linear activation function . This means that the neurons will only be deactivated if the output of the linear transformation is less than 0. The plot below will help you understand this better: *f(x)=max(0,x)*



Learning Methods

The learning process tries to teach the network how to produce the output when the corresponding input is presented. When learning is complete: the trained neural network, with the updated optimal weights, should be able to produce the output within desired corresponding to an input pattern.

Learning methods

- Supervised learning: Supervised learning means guided learning by "teacher"; requires a training set which consists of input vectors and a target vector associated with each input vector.
- Unsupervised learning: The objective of unsupervised learning is to discover patterns or features in the input data with no help from a teacher, basically performing a clustering of input space.
- Reinforced learning: The ANN makes a decision by observing its environment. If the observation is negative, the network adjusts its weights to be able to make a different required decision the next time.

Error-correction:

Assume that binary-valued functions are used e.g the step function. The weights are updated using

$$w_i(t+1) = w_i(t) + \Delta w_i(t)$$

$$with \quad \Delta w_i(t) = \eta(-\frac{\partial E}{\partial w_i})$$

$$where \quad \frac{\partial E}{\partial w_i} = -2(t_p - f_p)x_{i,p}$$

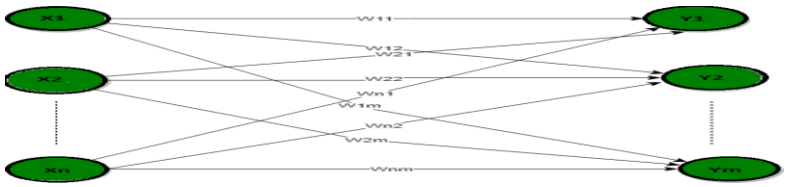
$$\boxed{w_i(t+1) = w_i(t) + 2\eta(t_p - f_p)x_{i,p}}$$

Different Network Topologies

There exist five basic types of neuron connection architecture:

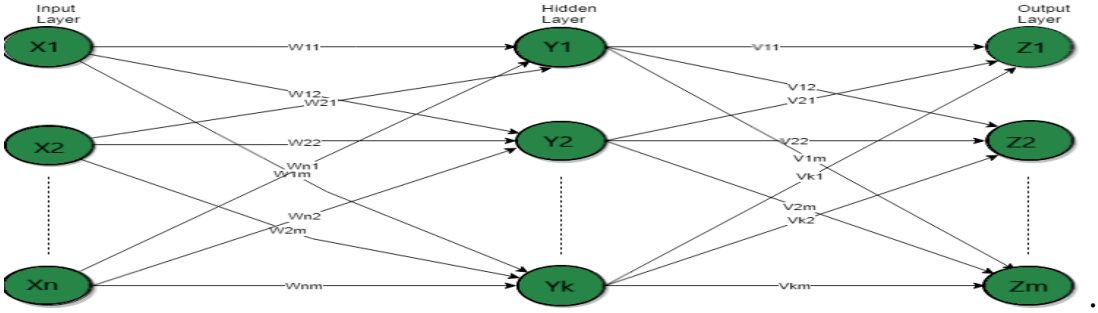
1. Single-layer feed forward network
2. Multilayer feed forward network
3. Single node with its own feedback
4. Single-layer recurrent network
5. Multilayer recurrent network.

1. **Single-layer feed forward network**

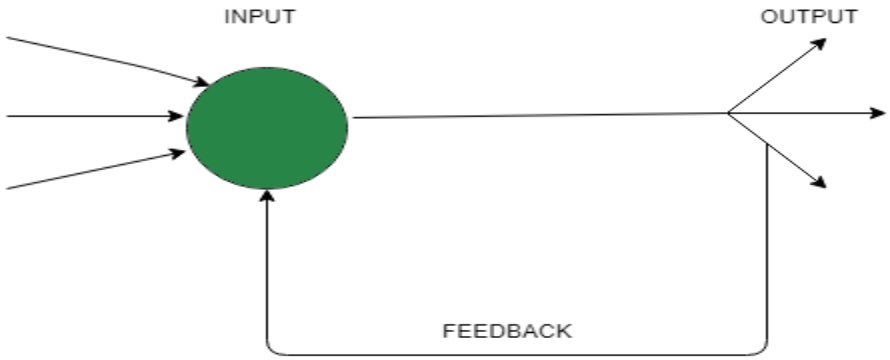


In this type of network, we have only two layers input layer and output layer but input layer does not count because no computation performed in this layer. Output layer is formed when different weights are applied on input nodes and the cumulative effect per node is taken. After this the neurons collectively give the output layer compute the output signals.

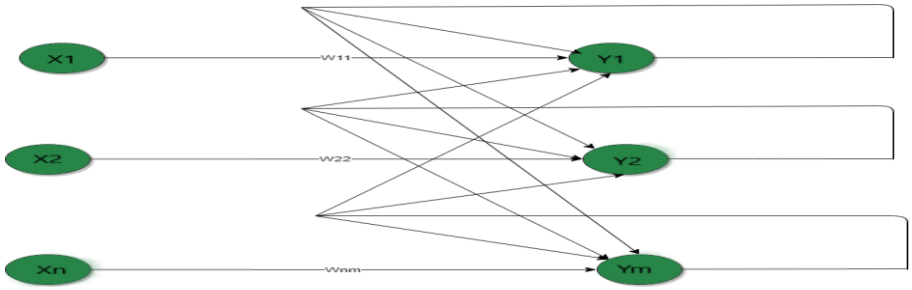
2. **Multilayer feed forward network:** This layer also has hidden layer which is internal to the network and has no direct contact with the external layer. Existence of one or more hidden layers enable the network to be computationally stronger, feed-forward network because information flows through the input function, and the intermediate computations used to define the output Z. There are no feedback connections in which outputs of the model are fed back into itself



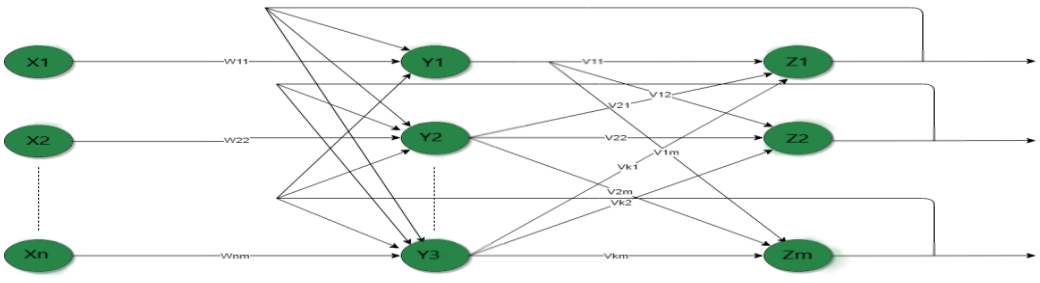
3. **Single node with its own feedback**: When outputs can be directed back as inputs to the same layer or proceeding layer nodes, then it results in feedback networks. Recurrent networks are feedback networks with closed loop. Above figure shows a single recurrent network having single neuron with feedback to itself.



4. **Single-layer recurrent network:** network is single layer network with feedback connection in which processing element's output can be directed back to itself or to other processing element or both. Recurrent neural network is a class of artificial neural network where connections between nodes form a directed graph along a sequence.



5. **Multilayer recurrent network:** In this type of network, processing element output can be directed to the processing element in the same layer and in the preceding layer forming a multilayer recurrent network. They perform the same task for every element of a sequence, with the output being depended on the previous computations. Inputs are not needed at each time step.



The ANN applications

• Classification, the aim is to predict the class of an input vector.

• Pattern matching, the aim is to produce a pattern best associated with a given input vector.

• Pattern completion, the aim is to complete the missing parts of a given input vector.

• Optimization, the aim is to find the optimal values of parameters in an optimization problem.

• Control, an appropriate action is suggested based on given an input vectors.

• Function approximation/times series modeling, the aim is to learn the functional relationships between input and desired output vectors;

• Data mining, with the aim of discovering hidden patterns from data artificial neural networks work through the optimized weight values. The method by which the optimized weight values are attained is called.