



ROHINI

COLLEGE OF ENGINEERING & TECHNOLOGY

Approved by AICTE and Affiliated to Anna University (An ISO Certified Institution) | Accredited with A+ Grade by NAAC
Recognized under Section 2(f) of University Grants Commission, UGC ACT 1956
(AUTONOMOUS)

BASIC MODELS OF ARTIFICIAL NEURAL NETWORK

The models of ANN are specified by the three basic entities namely

1. The model's synaptic interconnection.
2. The training rules or learning rules adopted for updating and adjusting the connection weights.
3. Activation functions.

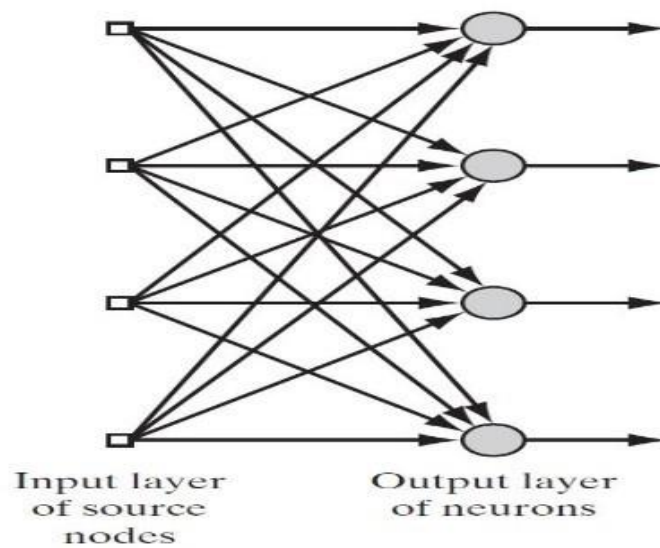
Network Architectures

The manner in which the neurons of a neural network are structured is intimately linked with the learning algorithm used to train the network.

There are three fundamentally different classes of network architectures:

1. Single-Layer Feedforward Networks
2. Multilayer Feedforward Networks
3. Recurrent Networks

Single-Layer Feedforward Networks

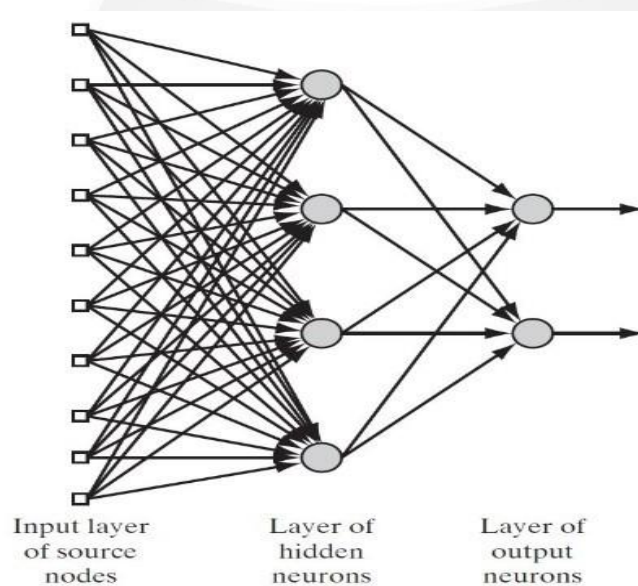


In this type of network, we have input layer and output layer but the input layer does not count because no computation is performed in this layer.

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 to compute the output signals.

Multilayer Feedforward Networks



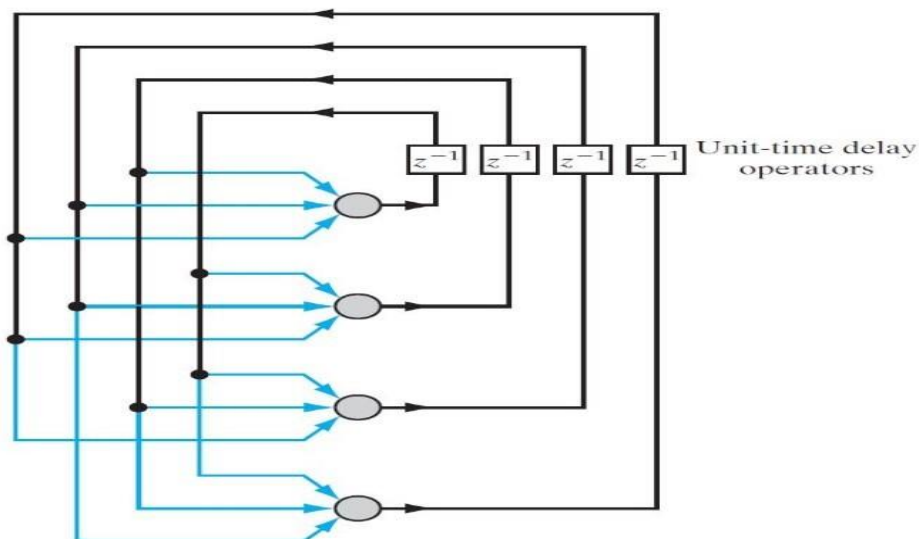
This network has one or more hidden layers, the term "hidden" refers to the fact that this part of the neural network is not seen directly from either the input or output of the network. The function of hidden neurons is to intervene between the external input and the network output in some useful manner.

The existence of one or more hidden layers enables the network to be computationally stronger.

Recurrent Networks

Are current neural network distinguishes itself from a feedforward neural network in that it has at least one feedback loop

Single Layer Recurrent Network



This network is a single-layer network with a feedback connection in which the processing element's output can be directed back to itself or to other processing elements or both.

A recurrent neural network is a class of artificial neural network where the connection between nodes forms a directed graph along a sequence.

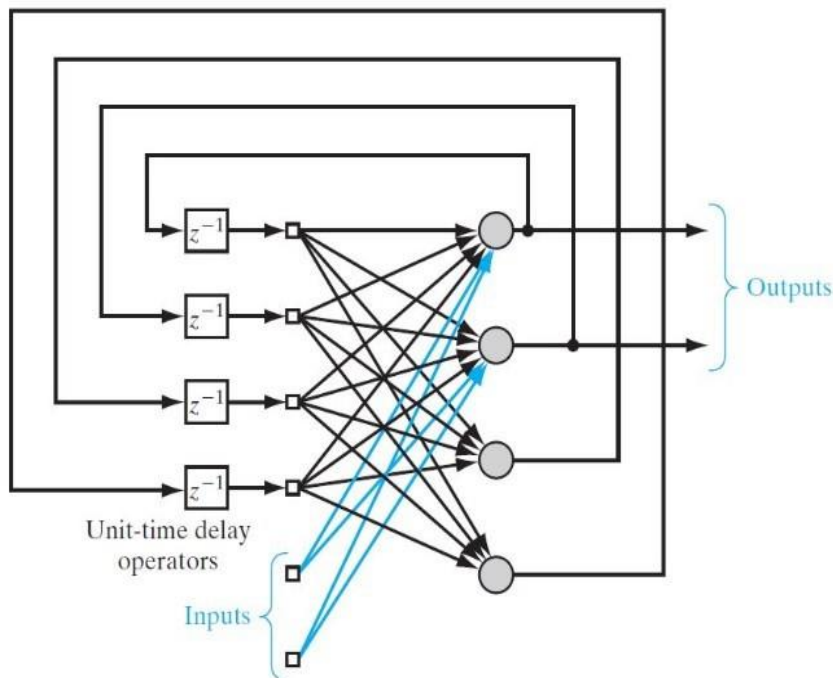
This allows it to exhibit dynamic temporal behavior for a time sequence. Unlike feedforward neural networks, RNNs can use their internal state (memory) to process sequences of inputs.

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.

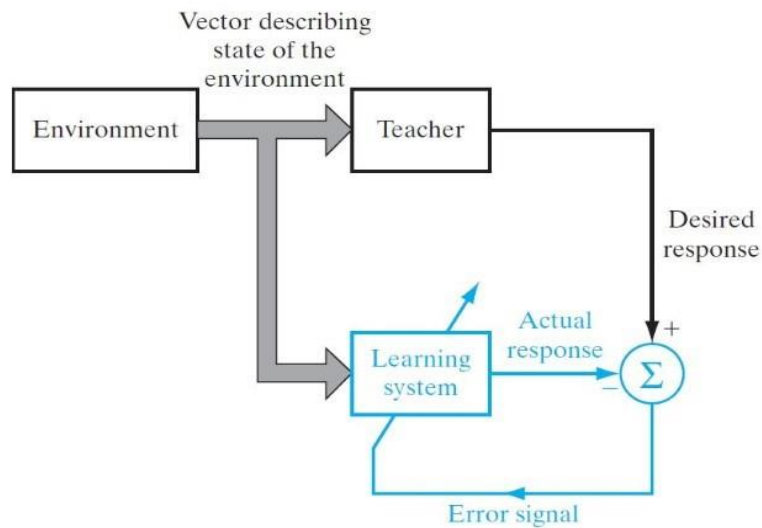
They perform the same task for every element of the sequence, with the output being dependent on the previous computations. Inputs are not needed at each time step.

The main feature of a multilayer recurrent network is its hidden state, which captures information about a sequence.



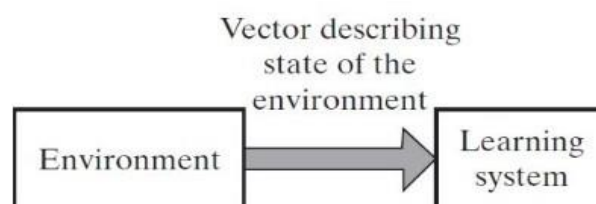
Learning rules

Just as there are different ways in which we ourselves learn from our own surrounding environments, neural networks learning processes as follows: learning with a teacher (**supervised learning**) and learning without a teacher (**unsupervised learning and reinforcement learning**). **Supervised Learning**



Learning with a teacher is also referred to as supervised learning. In conceptual terms, we may think of the teacher as having knowledge of the environment and unknown to the neural network. The teacher is able to provide the neural network with a desired response for that training vector. Indeed, the desired response represents the "optimum" action to be performed by the neural network. The network parameters are adjusted under the combined influence of the training vector and the error signal. The error signal is defined as the difference between the desired response and the actual response of the network. This adjustment is carried out iteratively in a step-by-step fashion with the aim of eventually making the neural network emulate the teacher; the emulation is presumed to be optimum in some statistical sense. In this way, knowledge of the environment available to the teacher is transferred to the neural network through training and stored in the form of "fixed" synaptic weights, representing long-term memory. When this condition is reached, we may then dispense with the teacher and let the neural network deal with the environment completely by itself.

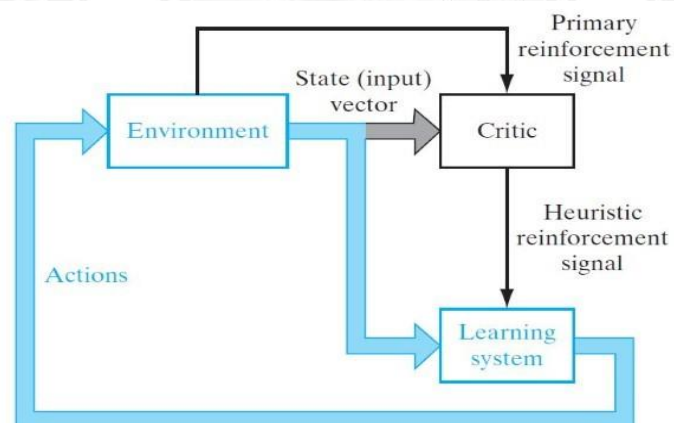
Unsupervised Learning



In unsupervised, or self-organized, learning is done without the supervision of a teacher. The goal of unsupervised learning is to find the underlying structure of dataset, group that data according to similarities, and represent that dataset in a compressed format.

To perform unsupervised learning, we may use a competitive-learning rule. For example, we may use a neural network that consists of two layers, an input layer and a competitive layer. The input layer receives the available data. The competitive layer consists of neurons that compete with each other (in accordance with a learning rule) for the "opportunity" to respond to features contained in the input data. In its simplest form, the network operates in accordance with a "winner-takes-all" strategy. In such a strategy, the neuron with the greatest total input "wins" the competition and turns on; all the other neurons in the network then switch off.

Reinforcement Learning



Reinforcement Learning is a feedback-based Network technique in which an agent learns to behave in an environment by performing the actions and seeing the results of actions. For each good action, the agent gets positive feedback, and for each bad action, the agent gets negative feedback or penalty.

Since there is no labeled data, so the agent is bound to learn by its experience only.

The agent interacts with the environment and explores it by itself. The primary goal of an agent in reinforcement learning is to improve the performance by getting the maximum positive rewards.

The goal of reinforcement learning is to minimize a cost-to-go function, defined as the expectation of the cumulative cost of actions taken over a sequence of steps instead of simply the immediate cost.

Activation functions

The activation function is applied over the net input to calculate the output of an ANN. An integration function (say f) is associated with the input of a processing element. This function serves to combine activation, information or evidence from an external source or other processing elements into a net input to the processing element.

When a signal is fed through a multilayer network with linear activation functions, the output obtained remains same as that could be obtained using a single-layer network. Due to this reason, nonlinear functions are widely used in multilayer networks compared to linear functions.

There are many activation functions available. In this part, we'll look at a few:

Identity function:

It is a linear function and can be defined as $f(x) = x$ for all x

The output here remains the same as input. The input layer uses the identity activation function.

Binary step function:

This function can be defined as

$$f(x) = \begin{cases} 1 & \text{if } x \geq \Theta \\ 0 & \text{if } x < \Theta \end{cases}$$

where Θ represents the threshold value. This function is most widely used in single-layer nets to convert the net input to an output that is a binary (1 or 0).

Bipolar step function:

This function can be defined as

$$f(x) = \begin{cases} 1 & \text{if } x \geq \Theta \\ -1 & \text{if } x < \Theta \end{cases}$$

where Θ represents the threshold value. This function is also used in single-layer nets to convert the net input to an output that is bipolar (+1 or -1).

Sigmoidal functions:

The sigmoidal functions are widely used in back-propagation nets because of the relationship

between the value of the functions at a point and the value of the derivative at that point which reduces the computational burden during training.

Sigmoidal functions are of two types:

1. Binary sigmoid function:

It is also termed as logistic sigmoid function or unipolar sigmoid function. The range of the sigmoid function is from 0 to 1. It can be defined as

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

where λ is the steepness parameter. The derivative of this function is

$$f'(x) = \lambda f(x) [1 - f(x)]$$

2. Bipolar sigmoid function:

The range of the Bipolar sigmoid function is from -1 to 1. It can be defined as

$$f(x) = \frac{2}{1 + e^{-\lambda x}} - 1 = \frac{1 - e^{-\lambda x}}{1 + e^{-\lambda x}}$$

where λ is the steepness parameter. The derivative of this function is

$$f'(x) = \frac{\lambda}{2} [1 + f(x)] [1 - f(x)]$$

The bipolar sigmoidal function is closely related to hyperbolic tangent function, which is written as

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

The derivative of the hyperbolic tangent function is

$$h'(x) = [1 + h(x)][1 - h(x)]$$

