Neural Network

Brijendra Singh

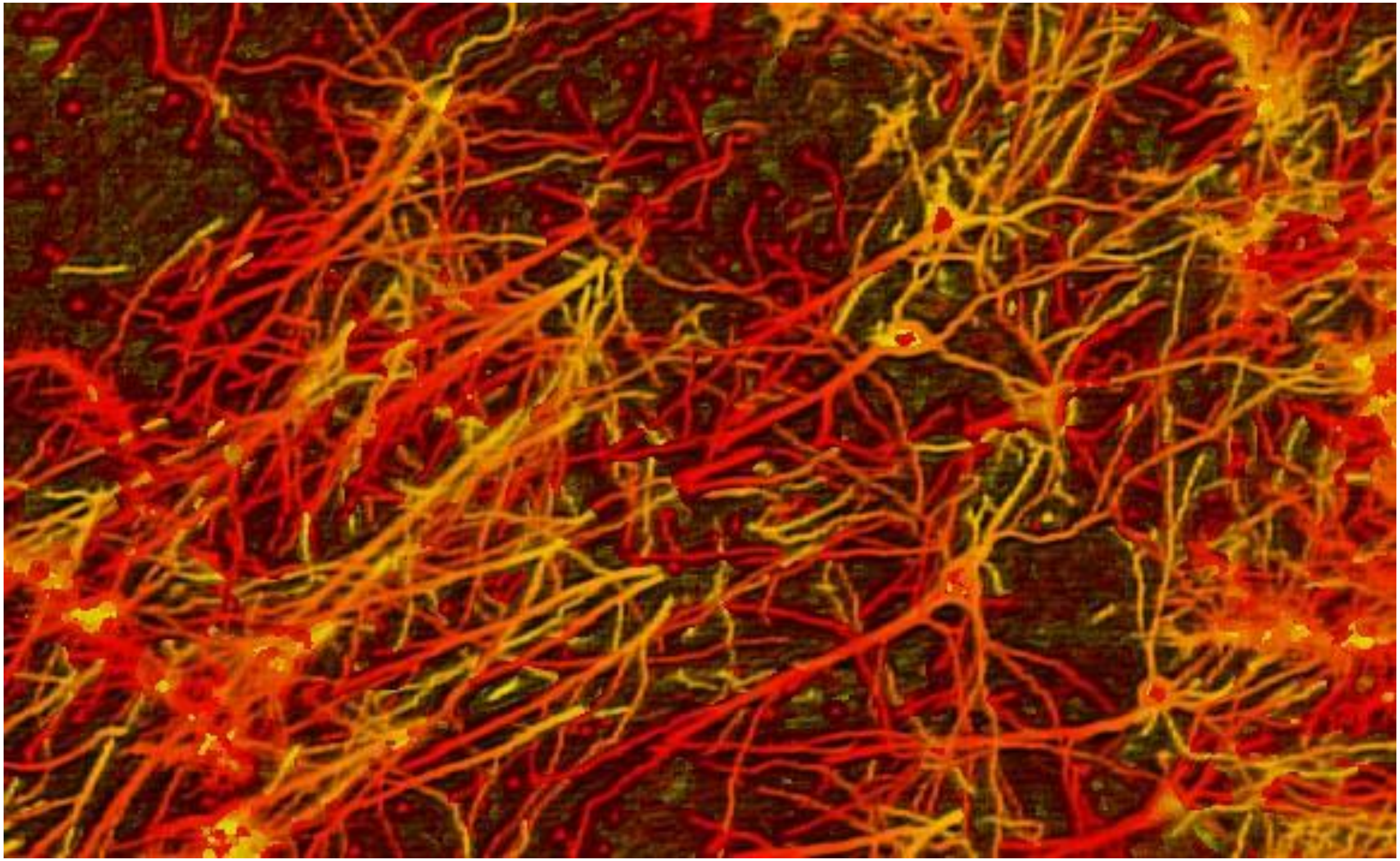
Neural Network

- It is a processing device.
- It can be a algorithm or an actual hardware.
- It is inspired by the function of animal brain.
- It has ability to learn by example which makes them very flexible and powerful.
- It is well suited for real world applications because of its parallel architecture.
- It works on the concept of human brain.
- Human brain is the amazing processor.
- Working of brain is still a mystery.

Neural Network

- The most basic element of the human brain is a specific type of cell, known as neuron, which doesn't regenerate.
- The human brain comprises about 100 billion neurons.
- Each neuron can connect with up to 200,000 other neurons, although 1,000-10,000 interconnections are typical.
- The power of the human mind comes from the sheer numbers of neurons and their multiple interconnections.
- It also comes from generic programming and learning.

Interconnections in brain



Application Scope of Neural Networks

- *Air traffic control* could be automated with the location, altitude, direction and speed of each radar blip taken as input to the network.
- *Betting* on horse races, stock markets, sporting events, etc. could be based on neural network predictions.
- *Fraud detection* regarding credit cards, insurance or taxes could be automated using a neural network analysis of past incidents.
- *Medical diagnosis* is an ideal application for neural networks.
- *Voice recognition* could be obtained by analysing the audio oscilloscope pattern, much like a stock market graph.
- *River water levels* could be predicted based on upstream reports, and rime and location of each report.

Artificial Neural Network

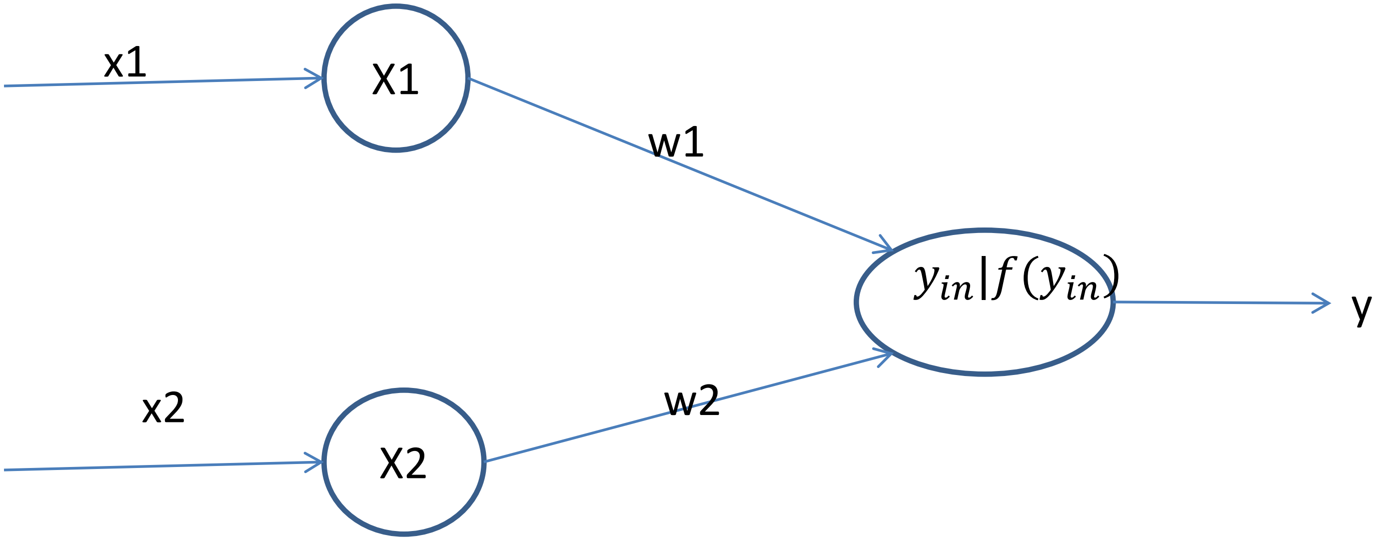
- An artificial neural network (ANN) is an efficient information processing system which resembles in characteristics with a biological neural network.
- ANNs possess large number of highly interconnected processing elements called *nodes* or *units* or *neuron,* which usually operate in parallel and are configured in regular architectures.
- Each neuron is connected with the other by a connection link.
- Each connection link is associated with weights which contain information about the input signal.
- This information is used by the neuron net to solve a particular problem.

Architecture of simple artificial neuron net

- Fig. $y_{in} = x_1 w_1 + x_2 w_2 \quad y = f(y_{in}) \quad f$: Activation function

X_1, X_2 : Input neurons $\quad Y$: Output neuron

X_1, X_2 : transmit signals, Y : receives signal



x_1, x_2 : activations of the input neurons:output of input signals

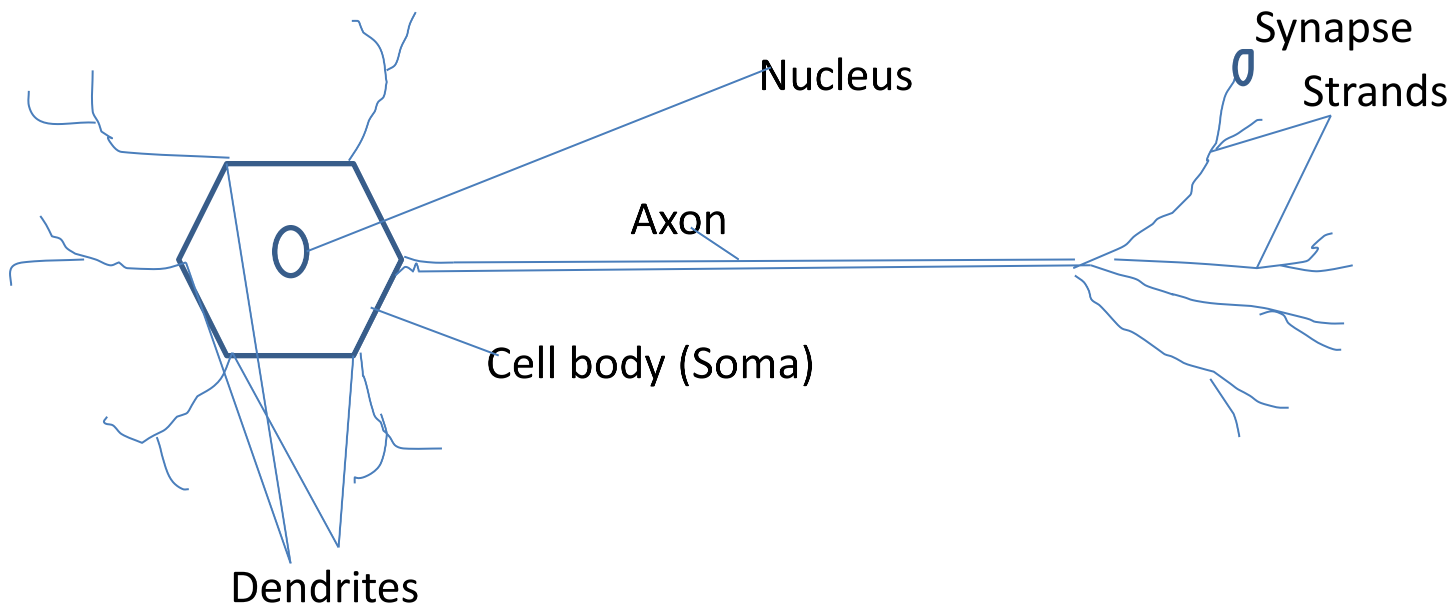
w_1, w_2 : associated weights,which containinformation about the input signals

Diagram of a biological neuron

- Diag.

Synapse : bulb like organ at the end of strands

Strands : The splits at the end of axons



Dendrites : Where the nerve is connected to the cell body

Axon : Which carries the impulses of the neuron

Biological Neural Network Components

- Cell body/Soma: Cell nucleus is found inside the cell body.
- Dendrites: Where the nerve is connected to the cell body.
- Axon: It carries the impulses of the neuron. It is a single long connection extending from the cell body and carrying signals from the neuron

Biological Neural Network Components

- Strands :The components into which an axon splits into at its end.
- Synapse: A small bulb-like organ into which each strand terminates.
- There are approximately 10^4 synapses per neuron in the human brain.
- Neuron introduces its signals to nearby neurons using synapses.

Mathematical representation of above process in artificial neurons

- Suppose we have n inputs from n neurons

  $X_1, X_2, \ldots X_n$ and with activations $x_1, x_2, \ldots x_n$

respectively .

- Let the weights of the interconnections between $X_1, X_2, \ldots X_n$ and the connecting neuron Y be $w_1, w_2, \ldots w_n$ respectively.

Mathematical representation of above process in artificial neurons

- The net input to the neuron Y is given by the formula:

$$y_{in} = x_1 w_1 + x_2 w_2 + \ldots x_n w_n$$

- Further, activation function is applied on y_{in} to compute output.
- Weights can be positive or negative.

- **+ve** weight means the synapse is **excitatory**
- **-ve** weight means the synapse is **inhibitory**

Biological Vs Artificial Neuron

Factors	Biological Neuron	Artificial Neuron
Speed	Few milliseconds	Few nanoseconds
Processing	Massive parallel operations simultaneously.	Several parallel operations simultaneously. Processing is faster than biological neuron.
Size and Complexity	The total no. of neurons is 10^{11} The total no. of interconnections is 10^{15} The complexity of a biological neuron is high The size of a biological neuron is high	Size and complexity depends on the chosen application and network designer.
Storage capacity	Stores the information in its interconnection or in synapse. Fails to recollect the lost information.	Stored in its contiguous memory locations. Artificial Neuron can recollect the lost information.
Tolerance	It has fault tolerance capability. Fault tolerance is the property that enables a system to continue operating properly in the event of the failure (or one or more faults within) of some of its components.	It has no fault tolerant property.
Control mechanism	There is no control unit in the brain	In AN there is a control unit in CPU, which can transfer and control precise scalar values from unit to unit

Basic models of artificial neural network

The models are specified by the three basic entities

- The model's synaptic (through synapses) interconnections
- The training or learning rules adopted for updating and adjusting the connection weights
- Their activation functions

Connections

- A neurons should be visualized for their arrangements in layers.
- A ANN consists of a set of highly interconnected processing element (neurons) such that each processing element output is found to be connected through weights to the other processing elements or to itself.
- The arrangement of neurons to form layers and the connection pattern formed within and between layers is called the **network architecture**. Five basic types of neuron connection architectures:
  – Single-layer feed-forward network
  – Multilayer feed-forward network
  – Single-node with its own feedback
  – Single-layer recurrent network
  – Multilayer recurrent network

Various Layers in ANN

1. Input Layer:

- It receives the input.
- Each input unit may be designated by an attribute value possessed by the instance.

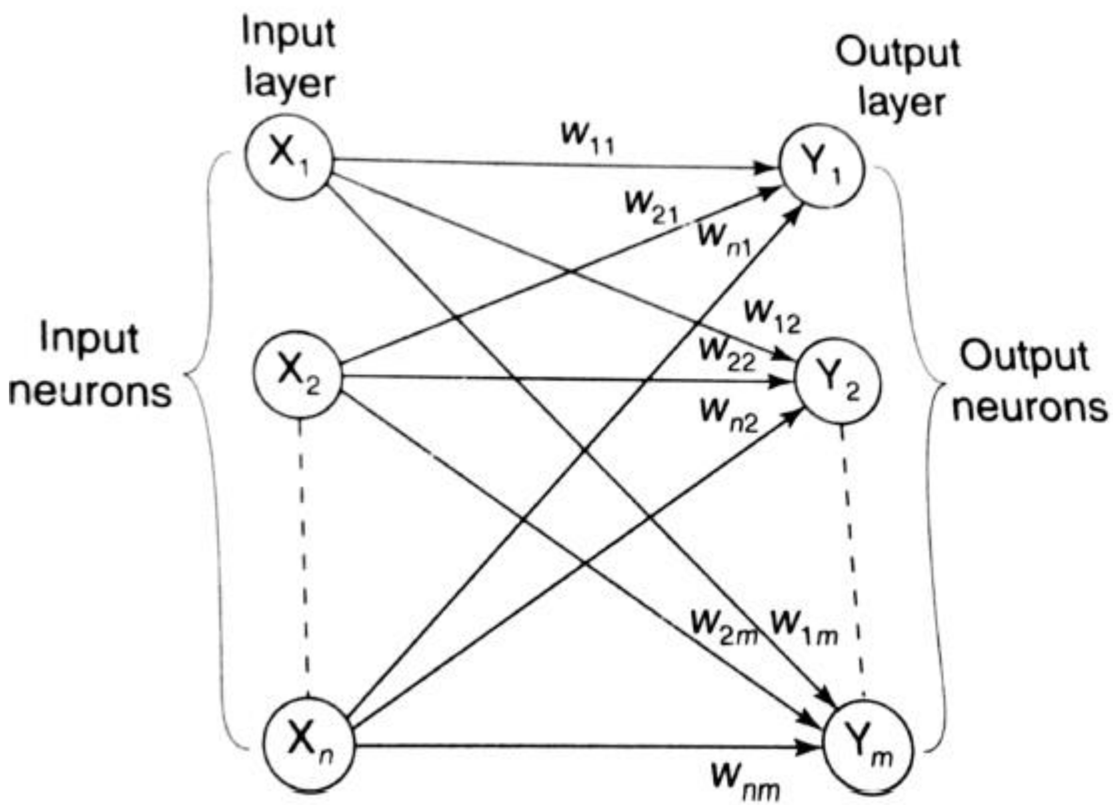
2. Hidden Layer:

- Any layer which is formed between the input and output layer is called as hidden layer.
- This layer is internal to the network and has no direct contact with the external environment.
- Complexity of the network increases with the increased number of hidden layer.

3. Output Layer:

- It generates the output of the network.

Single-layer feed-forward network



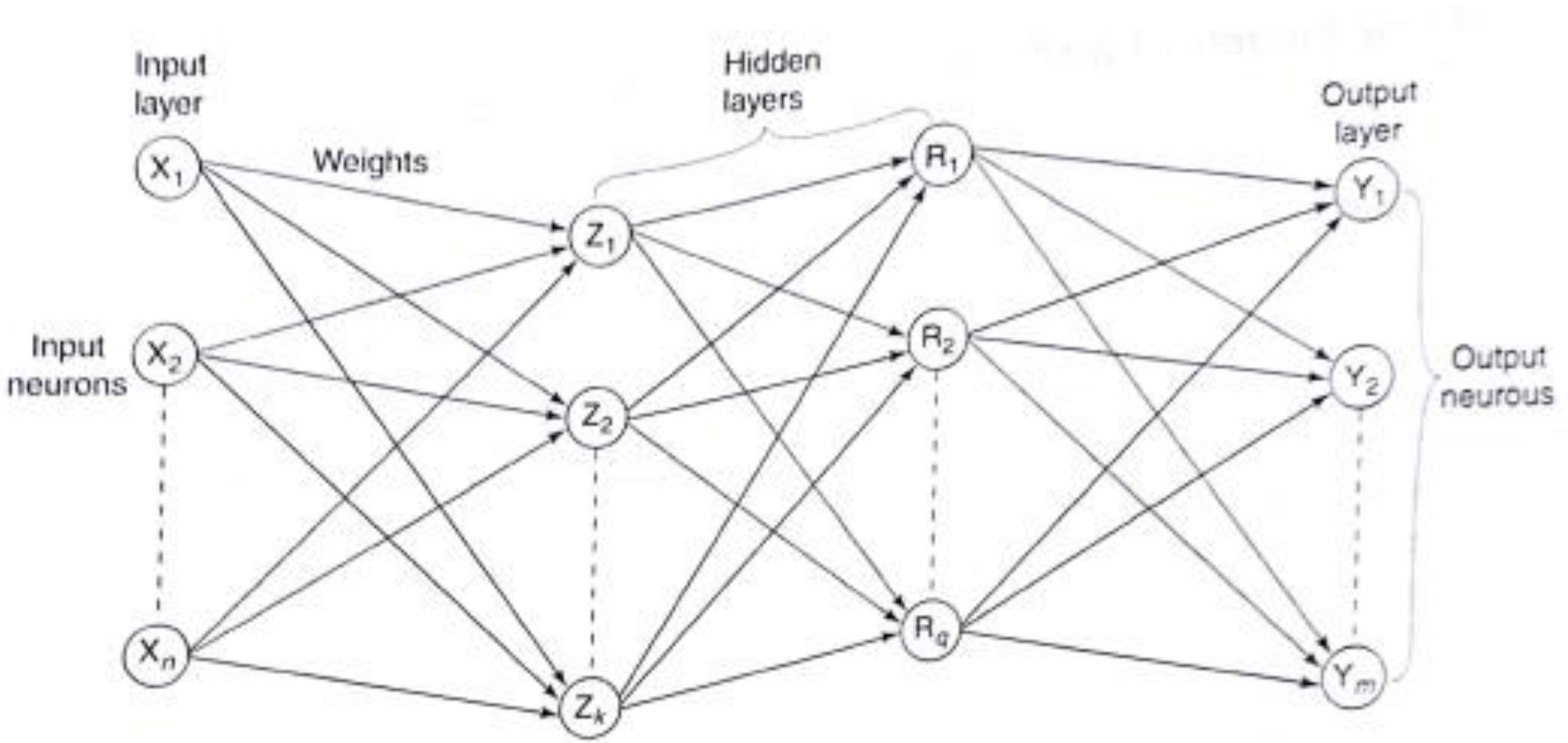
Description

- Basically. Neural nets are classified into single-layer/multi-layer neural nets.
- A layer is formed by taking a processing element and combining it with other processing elements.
- Practically, a layer implies a stage, going stage by stage.
- Input and output stages are linked with each other.

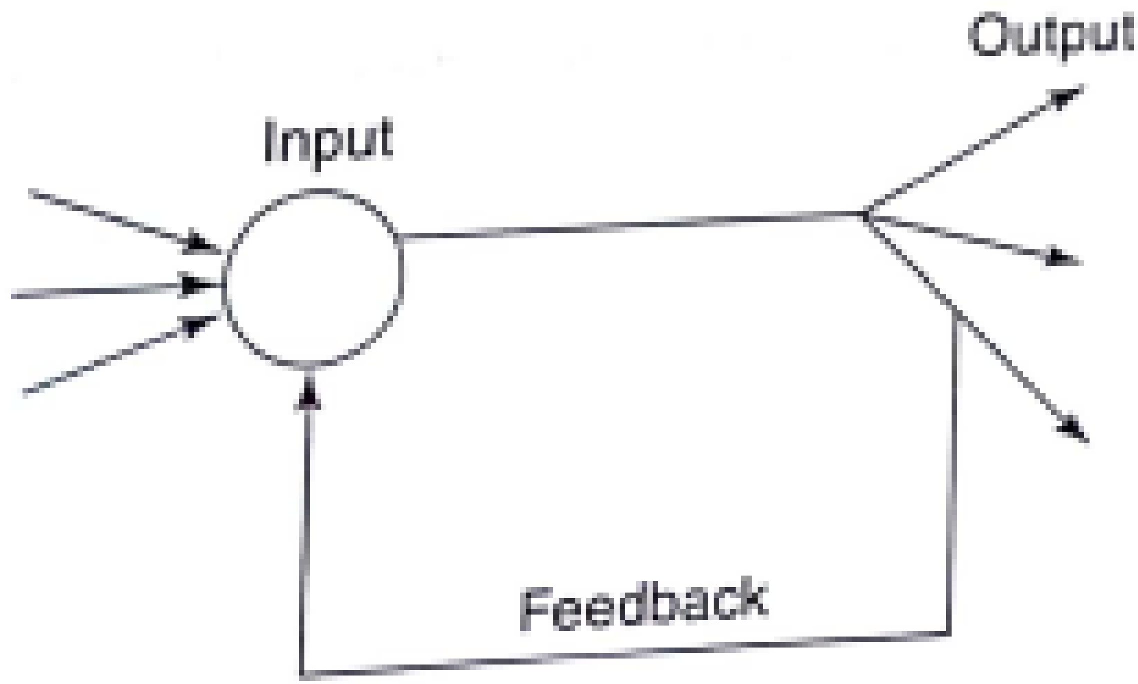
Description

- A network is said to be feed forward network if no neuron in the output later is an input to a node in the same layer or preceding layer.
- If the output can be directed back as input to same or preceding layers nodes then it is called as feedback network.
- When the feedback is directed as input to the nodes in the same layer then it is called as lateral feedback.
- Recurrent networks are feedback networks with closed loop.

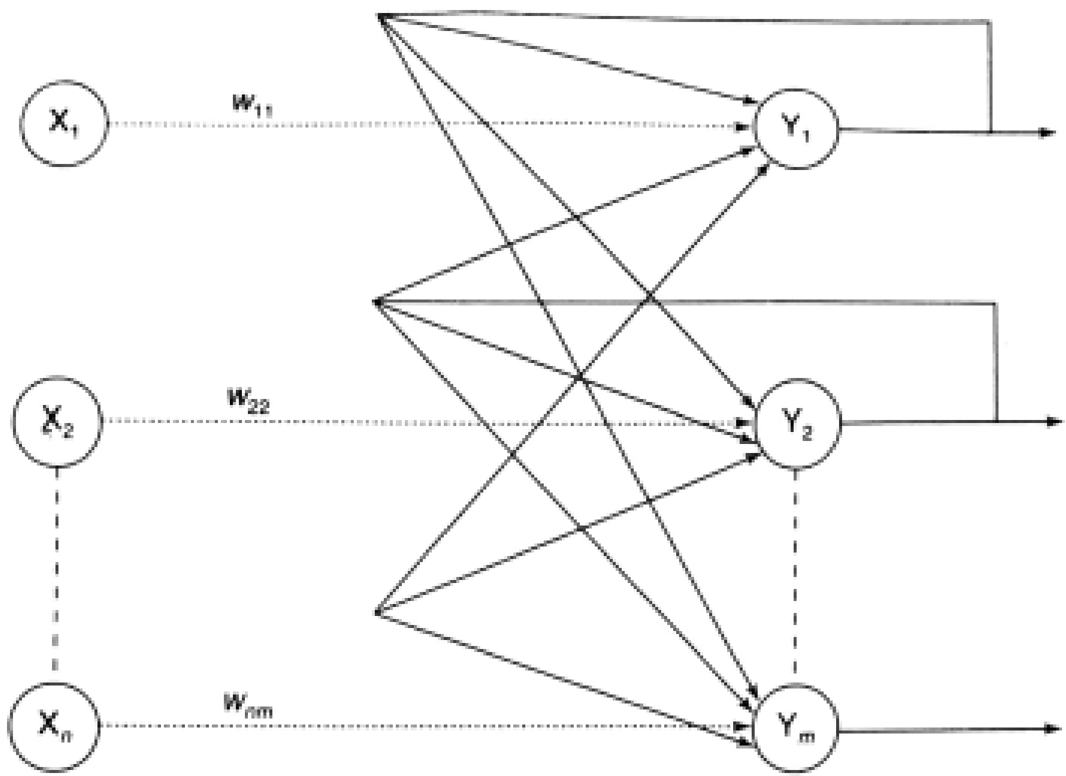
Multilayer feed-forward network



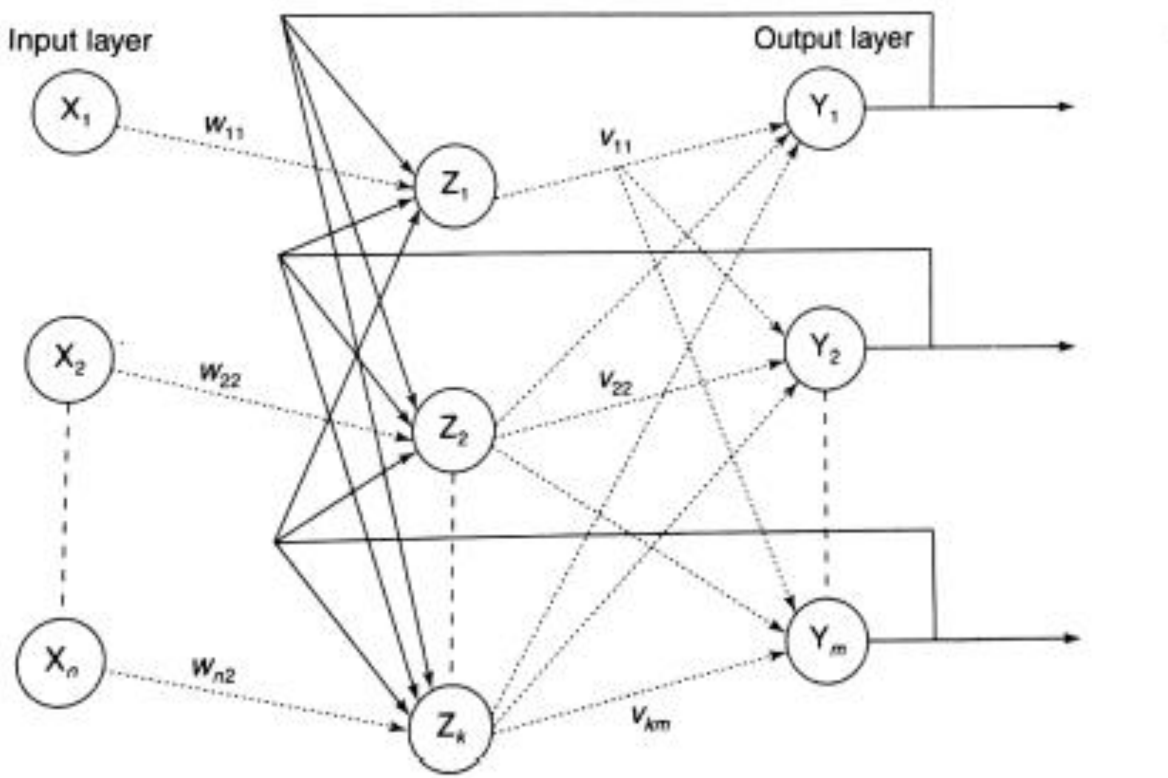
Single-node with its own feedback



Single-layer recurrent network



Multilayer recurrent network



Setting the Weights (Training or Learning)

- The main purpose of ANN is its capacity to learn.
- Learning or training is a process by means of which a NN adapts itself to a stimulus by making proper parameter adjustments, resulting in the production of desired response.
- There are two kinds of learning in ANNs:
  - Parameter Learning: updates the connecting weights in a NN
  - Structure Learning: It focuses on the change in network structure (which includes the number of processing elements as well as their connection types)
- Other common way to categorized ANN in 3 ways:
  - Supervised
  - Unsupervised
  - Reinforcement

Learning categories

- **Supervised learning:**
  Learning through a teacher/ supervisor
- **Unsupervised learning:**
  Learning without a teacher /supervisor
- **Reinforcement learning:**
  Learning basing upon a **critic information**
  It is similar to supervised learning

Supervised Learning

- The learning here *is* performed with the help of a teacher.
- Let us take the example of the learning process of a small child. The child doesn't know how to read/write.
- He/she is being taught by the parents at home and by the teacher in school.
- In ANN context, each input vector requires a corresponding target vector, which represents the desired output.
- The input vector along with the target vector is called ***training pair.***
- The network here is informed precisely about what should be emitted as output.
- During training the input vector is presented to a network which result in an output vector (actual).

Supervised Learning

- Actual output vector is compared with the desired (target).
- If there exists a difference between the two output vectors then an error signal is generated by the network.
- This error signal is used for adjustment of weights until the actual output matches the desired (target) output.
- In this learning, it is assumed that the correct “target” output values are known for each input pattern.

Unsupervised learning

- The learning here is performed without the help of a teacher.
- Consider the learning process of a tadpole, it learns by itself, that is, a child fish learns to swim by itself, it is not taught by its mother.
- The input vectors of similar type are grouped without the use of training data to specify how a member of each group looks or to which group a member belongs.
- In the training process the network receives the input patterns and organizes these patterns into clusters.
- When a new pattern arrives the ANN gives an output response indicating the class to which the pattern belongs.

Reinforcement learning

- This learning process is similar to supervised learning.
- In the case of supervised learning, the correct target output values are known for each input pattern.
- But, in some cases, less information might be available.
- For example, the network might be told chat its actual output *is* only "50% correct" or so.
- Thus, here only critic information is available, not the exact information.
- The learning is based on this critic information is called as reinforcement learning.

Activation Function

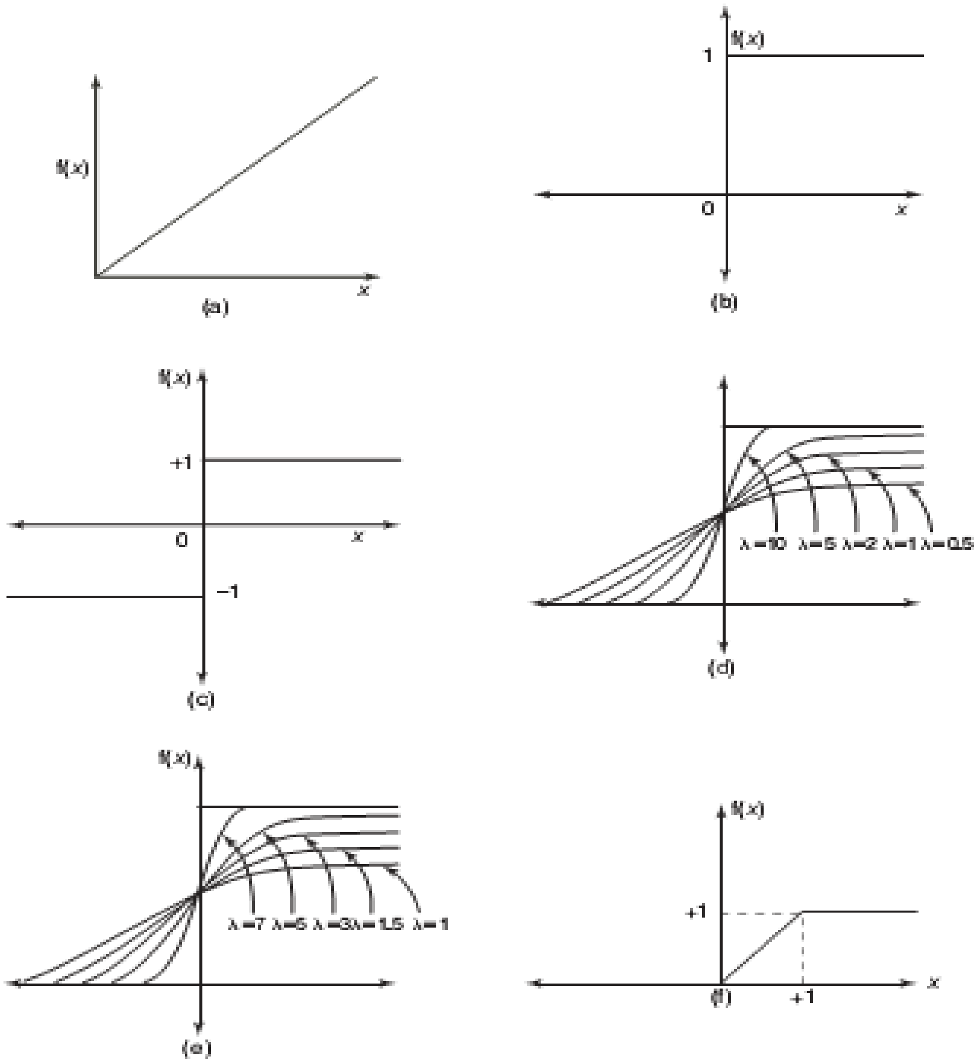
- To better understand the role of the activation function, let us assume a person is performing some work.
- To make the work more efficient and to obtain exact output, some force or activation may be given.
- This activation helps in achieving the exact output.
- In a similar way, the activation function is applied over the net input to calculate the output of an ANN.

Activation Function

Most of the time you will be concerned about the following points when choosing the activation function:

- continuity of the function
- computational power to process all neurons of the network
- type of the desired output (logistic/continuous variables or classification/categorical data)

ACTIVATION FUNCTION



Activation functions:

(A) Identity

(B) Binary step

(C) Bipolar step

(D) Binary sigmoidal

(E) Bipolar sigmoidal

(F) Ramp

Activation Functions

1. *Identity function:* It is a linear function and can be defined as

$$f(x) = x \quad \text{for all } x$$

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

2. *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).

Activation Functions

3. *Bipolar step function*: This function can be defined as

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

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).

Activation Functions

4. *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:

   - *Binary sigmoid function*: It is also termed as logistic sigmoid function or unipolar sigmoid function. It can be defined as

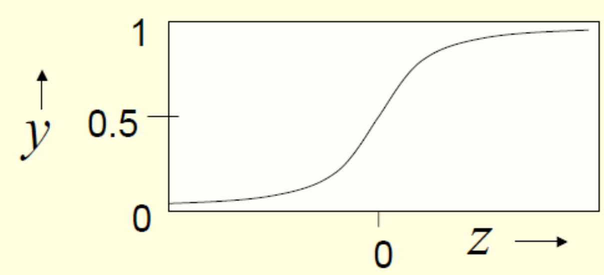
$$f(x) = \frac{1}{1 + e^{-\lambda x}}$$

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

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

Here the range of the sigmoid function is from 0 to 1.

$$z = b + \sum_i x_i w_i \qquad y = \frac{1}{1 + e^{-z}}$$



Activation Functions

- *Bipolar sigmoid function*: This function is 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 and the sigmoid function range is between -1 and $+1$. The derivative of this function can be

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

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

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

The derivative of the hyperbolic tangent function is

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

Activation Functions

If the network uses a binary data, it is better to convert it to bipolar form and use the bipolar sigmoidal activation function or hyperbolic tangent function.

5. *Ramp function*: The ramp function is defined as

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

The graphical representations of all the activation functions are shown in Figure 2-15(A)–(F).