ARTIFICIAL NEURAL NETWORK

TOPIC 3

Artificial Neural Network

- An artificial neural network is an abstract computational model of the human brain. The human brain has an estimated 1011 tiny units called neurons. These neurons are interconnected with an estimated 1015 links. Similar to the brain, an ANN is composed of artificial neurons (or processing units) and interconnections.

- When we view such a network as a graph, neurons can be represented as nodes (or vertices) and interconnections as edges. Although the term artificial neural network is most commonly used, other names include "neural network", parallel distributed-processing system (PDP), connectionist model, and distributed adaptive system. ANNs are also referred to in the literature as neurocomputers

Artificial Neural Network

An artificial neural network is a **massive parallel distributed processor** made up of **simple processing units**. It has the ability to **learn from experiential knowledge** expressed through interunit connection strengths, and can make such knowledge available for use.

Properties and Capabilities and ANN

1. **Nonlinearity** - An artificial neuron as a basic unit can be a linear- or nonlinear-processing element, but the entire ANN is highly nonlinear. It is a special kind of nonlinearity in the sense that it is distributed throughout the network.

This characteristic is especially important, for ANN models the inherently nonlinear real-world mechanisms responsible for generating data for learning.

Properties and Capabilities and ANN

2. Learning from examples - An ANN modifies its interconnection weights by applying a set of training or learning samples.

The final effects of a learning process are tuned parameters of a network (the parameters are distributed through the main components of the established model), and they represent implicitly stored knowledge for the problem at hand.

Properties and Capabilities and ANN

3. **Adaptivity**: An ANN has a built-in capability to adapt its interconnection weights to changes in the surrounding environment. In particular, an ANN trained to operate in a specific environment can be easily retrained to deal with changes in its environmental conditions.

Properties and Capabilities and ANN

4. Evidential Response: In the context of data classification, an ANN can be designed to provide information not only about which particular class to select for a given sample, but also about confidence in the decision made.

Properties and Capabilities and ANN

5. Fault Tolerance: An ANN has the potential to be inherently fault-tolerant, or capable of robust computation. Its performances do not degrade significantly under adverse operating conditions such as disconnection of neurons, and noisy or missing data. There is some empirical evidence for robust computation, but usually it is uncontrolled.

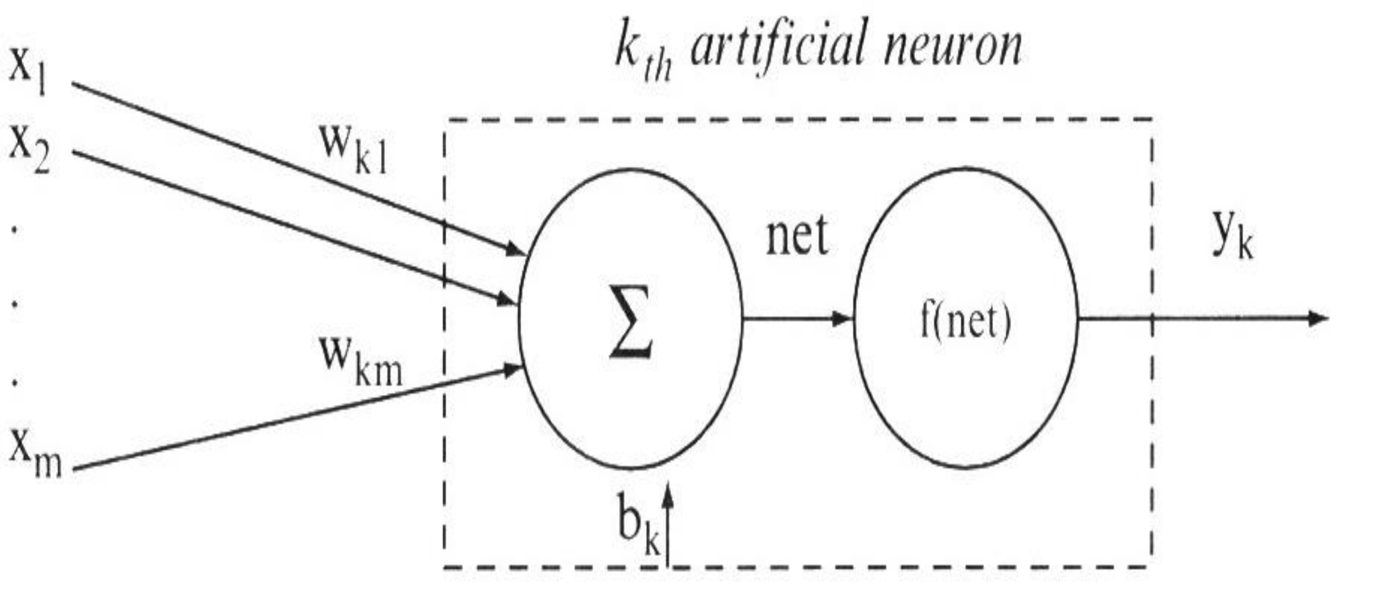
Properties and Capabilities and ANN

6. Uniformity of Analysis and Design: Basically, artificial neural networks enjoy universality as information processors. The same principles, notation, and the same steps in methodology are used in all domains)nvolving application of artificial neural networks.

Model of an Artificial Neuron

- An artificial neuron is an information-processing unit that is fundamental to the operation of an ANN. The block diagram which is a model of an artificial neuron shows that it consists of three basic elements:

1. A set of connecting links from different inputs x_i (or synapses), each of which is characterized by a weight or strength w_{ki}.
2. An adder for summing the input signals xi weighted by the respective synaptic strengths w_{ki}. The operation described here constitutes a linear combiner.
3. .An activation function f for limiting the amplitude of the output yk of a neuron.



Model of an Artificial Neuron

- The model of the neuron given in the figure also includes an externally applied bias, denoted by b_k. The bias has the effect of increasing or lowering the net input of the activation function, depending on whether it is positive or negative.

Model of an Artificial Neuron

- In mathematical terms, an artificial neuron is an abstract model of a natural neuron, and its processing capabilities are formalized using the following notation.
- First, there are several inputs x_i, i = 1, …, m. Each input xi is multiplied by the corresponding weight w_{ki} where k is the index of a given neuron in an ANN.
- The weights simulate the biological synaptic strengths in a natural neuron.

Model of an Artificial Neuron

- The weighted sum of products $x_i w_{ki}$, for i = 1, ..., m is usually denoted as net in the ANN literature:

$$net_k = x_1 w_{k1} + x_2 w_{k2} + \ldots + x_m w_{km} + b_k$$

- Using adopted notation for $w_{k0} = b_k$ and default input $x_0 = 1$, a new uniform version of net summation will be

$$net_k = x_0 w_{k0} + x_1 w_{k1} + x_2 w_{k2} + \ldots + x_m w_{km} = \sum_{i=1}^{m} x_i w_{ki}$$

Model of an Artificial Neuron

The same sum can be expressed in vector notation as a scalar product of two m-dimensional vectors:

$$net_k = X \cdot W$$

where

$$X = \{x_0, x_1, x_2, \ldots, x_m\}$$
$$W = \{w_{k0}, w_{k1}, w_{k2}, \ldots, w_{km}\}$$

Model of an Artificial Neuron

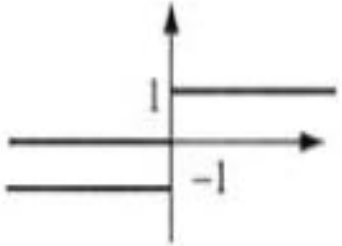
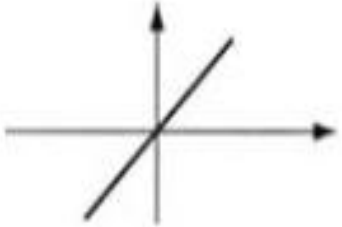
Finally, an artificial neuron computes the output yk as a certain function of net_k value:

$$y_k = f(net_k)$$

The function f is called the activation function. Various forms of activation functions can be defined.

Activation Functions

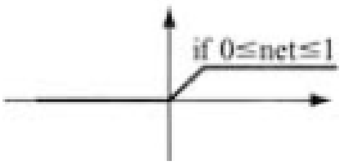
Some commonly used activation functions are:

Hard Limit	$y = \begin{cases} 1 & \text{if } net \geq 0 \\ 0 & \text{if } net < 0 \end{cases}$	
Symmetrical Hard Limit	$y = \begin{cases} 1 & \text{if } net \geq 0 \\ -1 & \text{if } net < 0 \end{cases}$	
Linear	Y= net	

Activation Functions

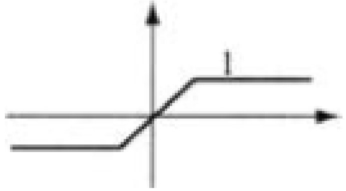
Sataurating Linear

$$y = \begin{cases} 1 & \text{if } net > 1 \\ \text{net} & \\ 0 & \text{if } net < 0 \end{cases}$$



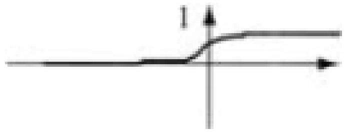
Symmetric Saturating Linear

$$y = \begin{cases} 1 & \text{if } net > 1 \\ \text{net} & \text{if } -1 \le net \le 1 \\ -1 & \text{if } net < -1 \end{cases}$$



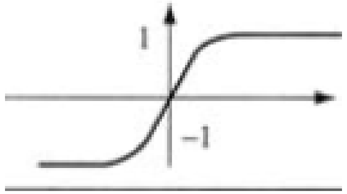
Log-Sigmoid

$y = 1/(1 + e^{-ner})$



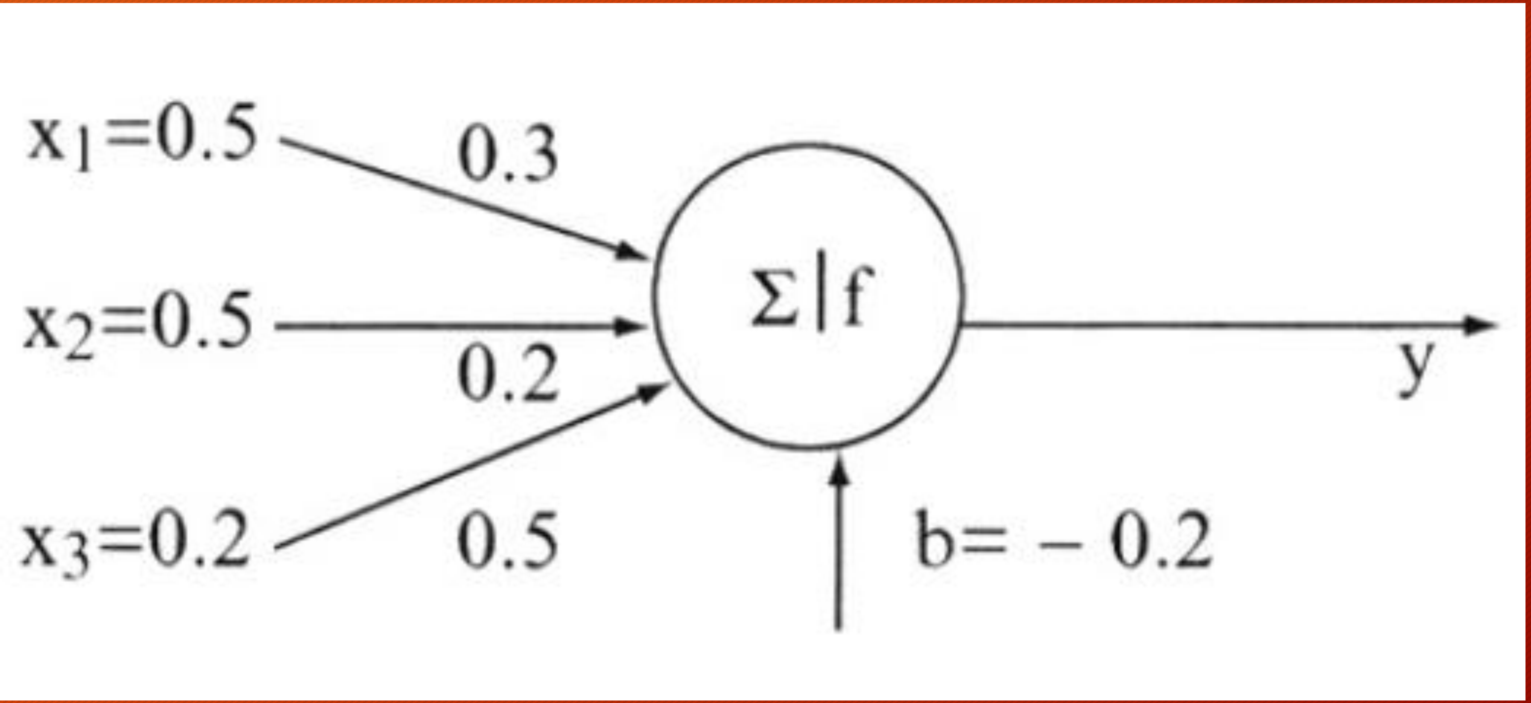
Hyperbolic Tangent Sigmoid

$y = (e^{net} - e^{-net})/(e^{net} + e^{-net})$



Examples

For example, for the neuron With three inputs and one output, the corresponding input values, weight factors, and bias are given in the figure. It is necessary to find the output y for different activation functions such as Symmetrical Hard Limit, Saturating Linear, and Log-Sigmoid.

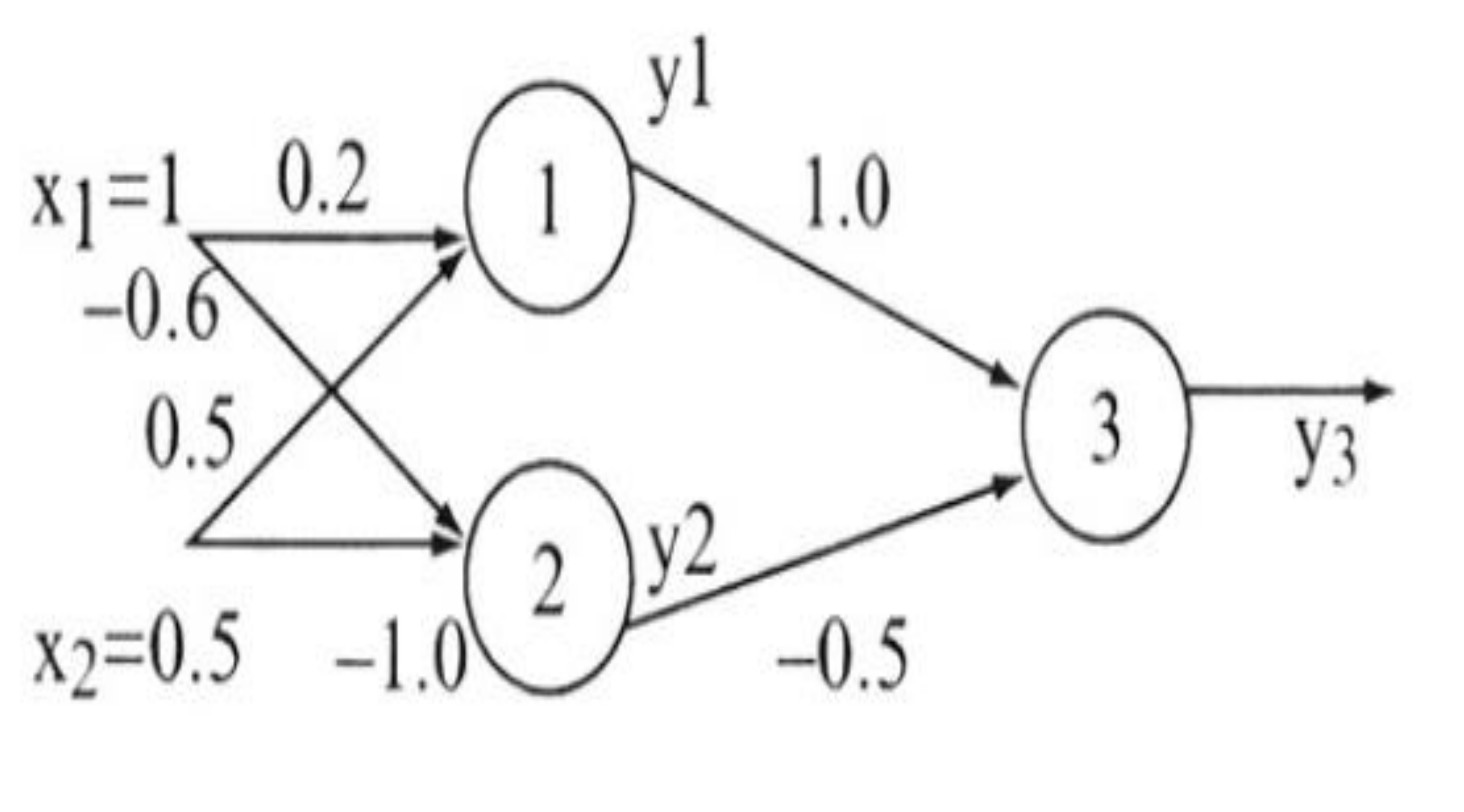


Assignment

Using the Saturating linear activation function, consider the following values to find the value of y: $x_1 = 0.5$, $x_2 = 0.35$, $x_3 = 0.1$, $w_1 = -0.3$, $w_2 = 0.1$, $w_3 = 0.2$, and $b = 0.1$

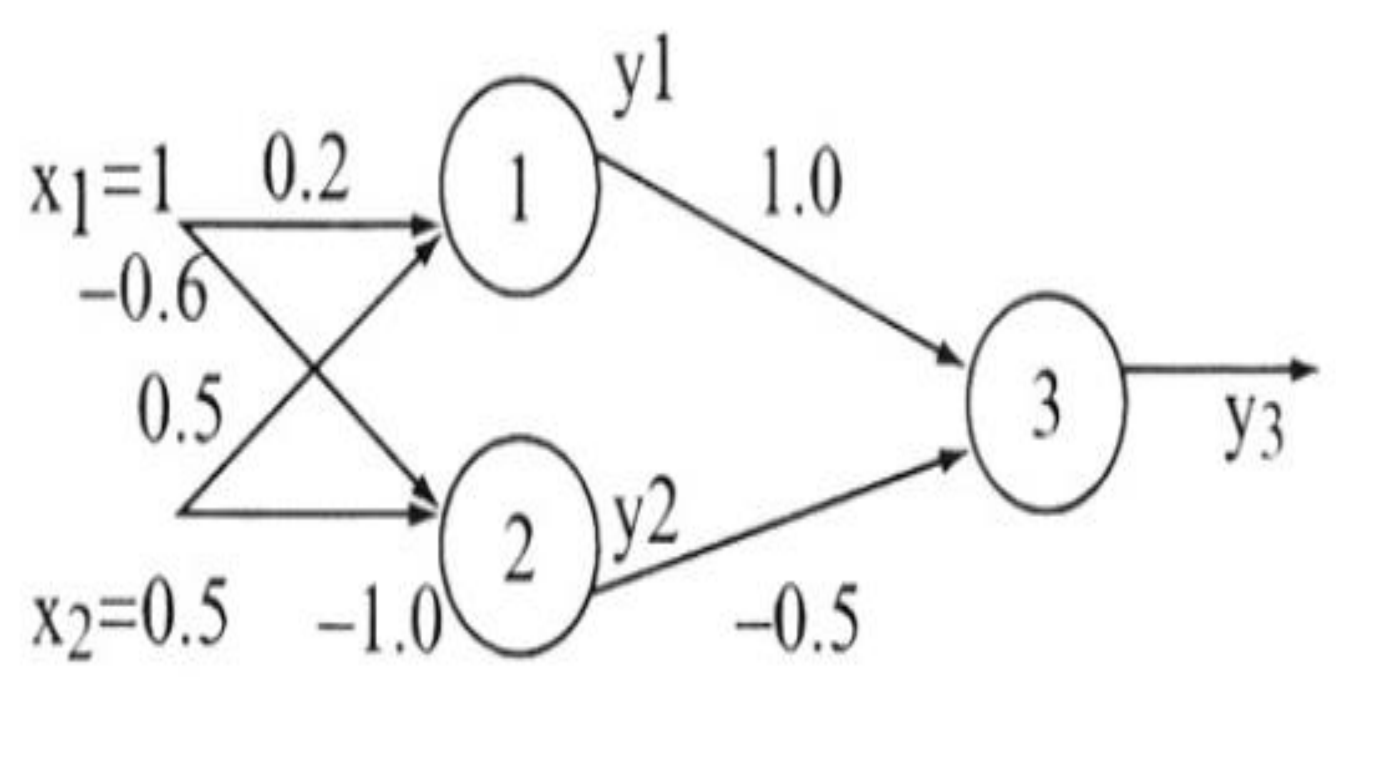
Examples

The basic principles of computation for one node may be extended for an artificial neural network with several nodes even if they are in different layers, as given in Figure. Suppose that for the given configuration of three nodes all bias values are equal to 0 and activation functions for all nodes are symmetric saturating linear. What is the final output Y3 from the node 3?



Assignment

Assume everything remains the same except for the value of x_1 = 0,8 and x_2 = 0.3. Use the linear activation function. What is the final output Y3 from the node 3?



Architecture of ANN

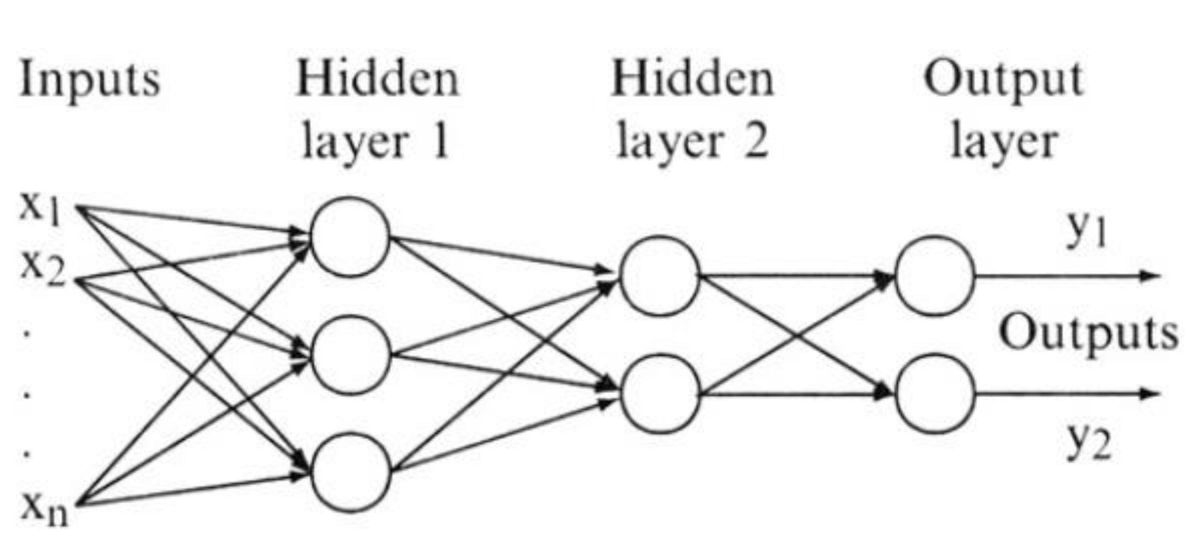
- The architecture of an artificial neural network is defined by the characteristics of a node and the characteristics of the node's connectivity in the network.

- Network architecture is specified by the **number of inputs to the network**, **the number of outputs**, **the total number of elementary nodes** that are usually equal processing elements for the entire network, and their organization and interconnections.

- Neural networks are generally classified into two categories on the basis of the type of interconnections: **feedforward** and **recurrent**.
- :

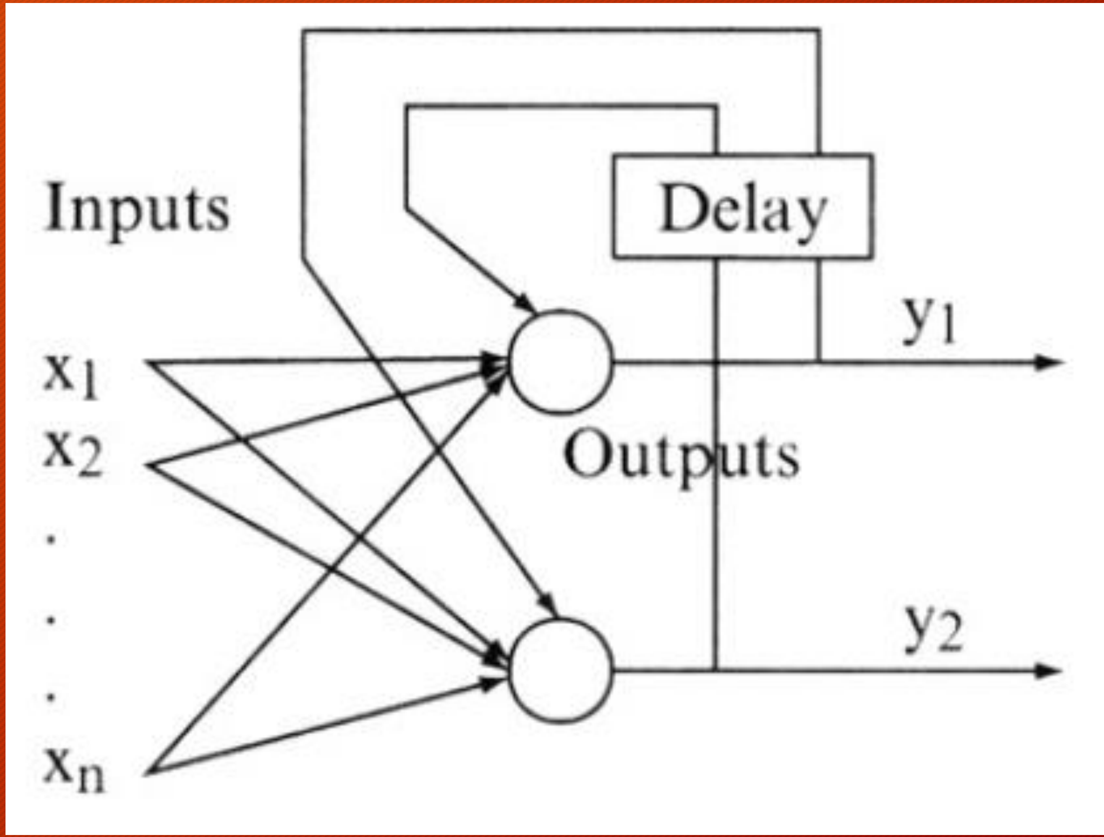
Feedforward ANN

- The network is feedforward if the processing propagates from the input side to the output side unanimously, without any loops or feedbacks.
- In a layered representation of the feedforward neural network, there are no links between nodes in the same layer; outputs of nodes in a specific layer are always connected as inputs to nodes in succeeding layers.



Recurrent ANN

- If there is a feedback link that forms a circular path in a network (usually with a delay element as a synchronization component), then the network is recurrent.



Learning Process

- Learning is a process by which the free parameters of a neural network are adapted through a process of stimulation by the environment in which the network is embedded. The type of learning is determined by the manner in which the parameters change.

- **NB: Study Section 9.3 and practice the example of the three iterations**

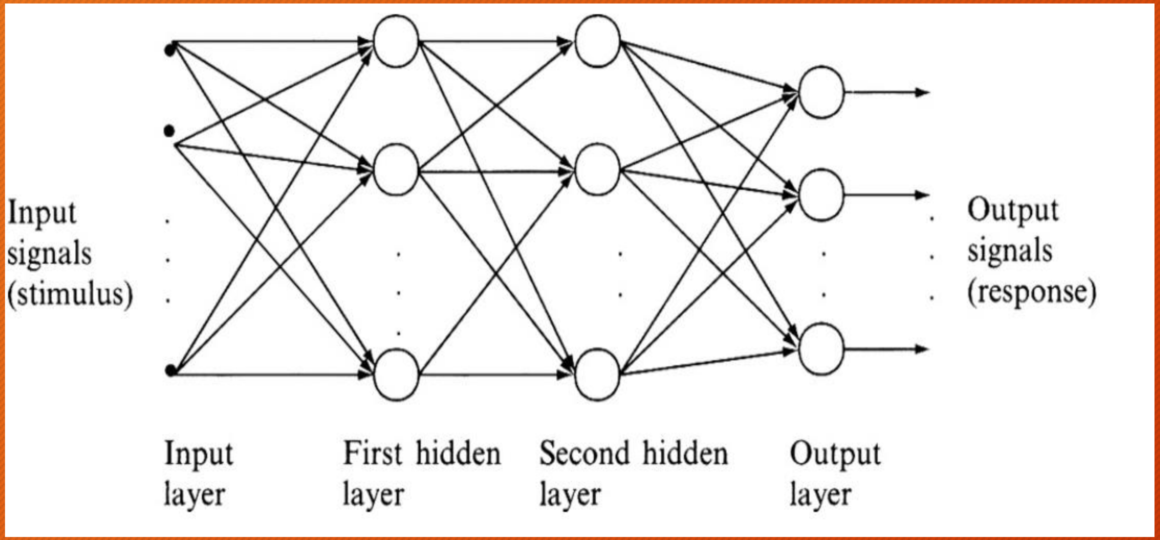
Learning Tasks of ANN

- Pattern association
- Pattern recognition
- Function Approximation
- Control
- Filtering

Assignment

Write short note on each as they apply to ANN

Multilayer Perceptrons



A graph of a multilayered perceptron architecture with two hidden layers

A multiplayer perceptron has three distinctive characteristics:

1. The model of each neuron in the network includes usually a nonlinear activation function, sigmoidal or hyperbolic.
2. The network contains one or more layers of hidden neurons that are not a part of the input or output of the network. These hidden nodes enable the network to learn complex and highly nonlinear tasks by extracting progressively more meaningful features from the input patterns.
3. The network exhibits a high degree of connectivity from one layer to the next one.

Competitive Networks and Competitive Learning

- Competitive neural networks belong to a class of recurrent networks, and-they are based on algorithms of unsupervised learning, such as the competitive algorithm.

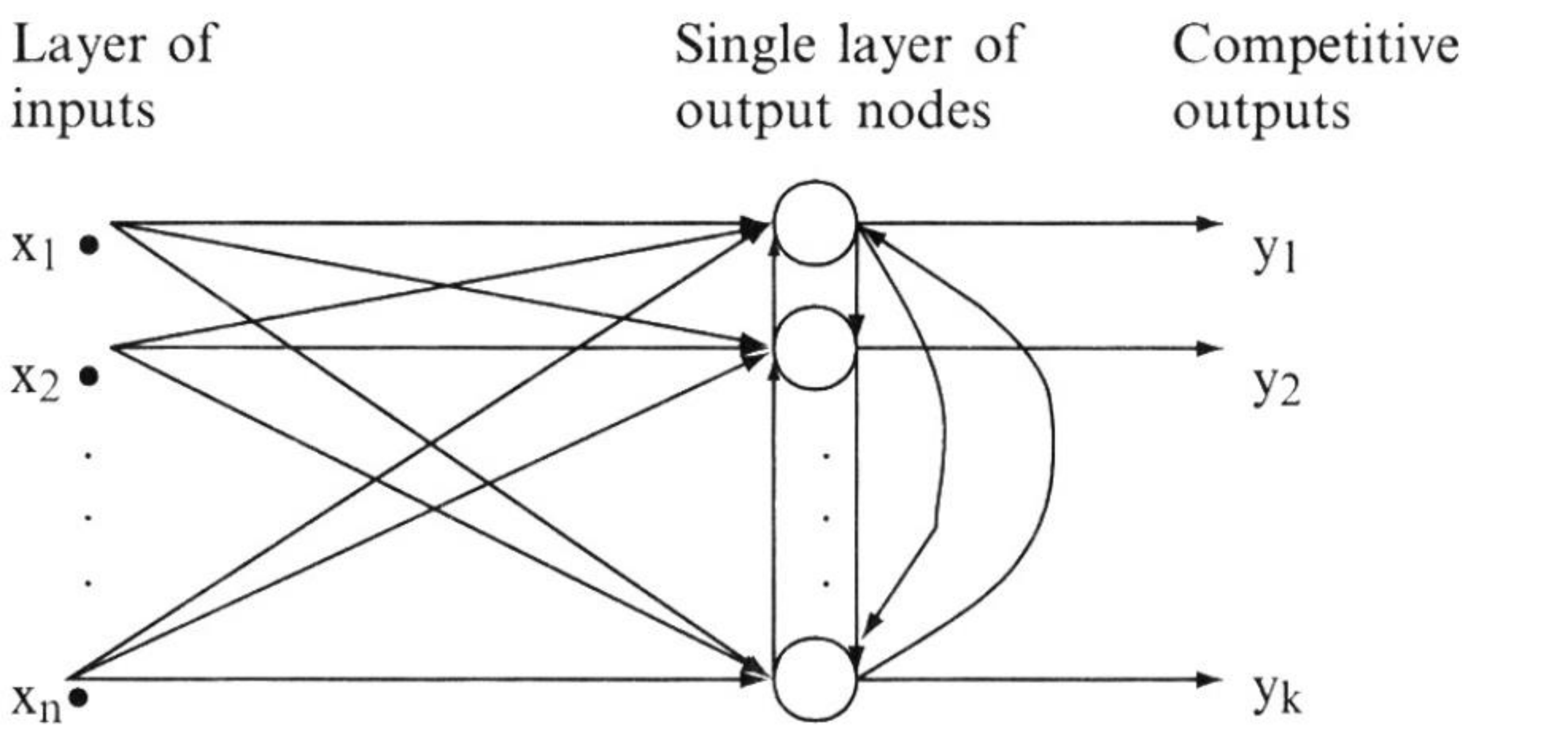
- Whereas in multilayer perceptrons several output neurons may be active simultaneously, in competitive learning only a single output neuron is active at anyone time.

Competitive Networks and Competitive Learning

- There are three basic elements necessary to build a network with a competitive learning rule, a standard technique for this type of artificial neural networks:

  - A set of neurons that have the same structure and that are connected with initially randomly selected weights. Therefore, the neurons. respond differently to a given set of input samples.
  - A limit value that is determined on the strength of each neuron.
  - A mechanism that permits the neurons to complete for the right to respond to a given subset of inputs, such that only one output neuron is active at a time. The neuron that wins the competition is called winner-takes-all neuron.

A graph of a simple competitive network architecture



Competitive Networks and Competitive Learning

- For a neuron k to be the winning neuron its net value net_k for a specified input sample X = {X_1, X_2, …, x_n} must be the largest among all the neurons in the network.
- The output signal Y_k of the winning neuron k is set equal to one; the outputs of all other neurons that lose the competition are set equal to zero. We thus write

$$y_k = \begin{cases} 1 \text{ if } net_k > net_j & \text{for all } j, j \neq k \\ 0 \text{ otherwise} \end{cases}$$

where the induced local value net_k represents the combined action of all the forward and feedback inputs to neuron k.

Competitive Networks and Competitive Learning

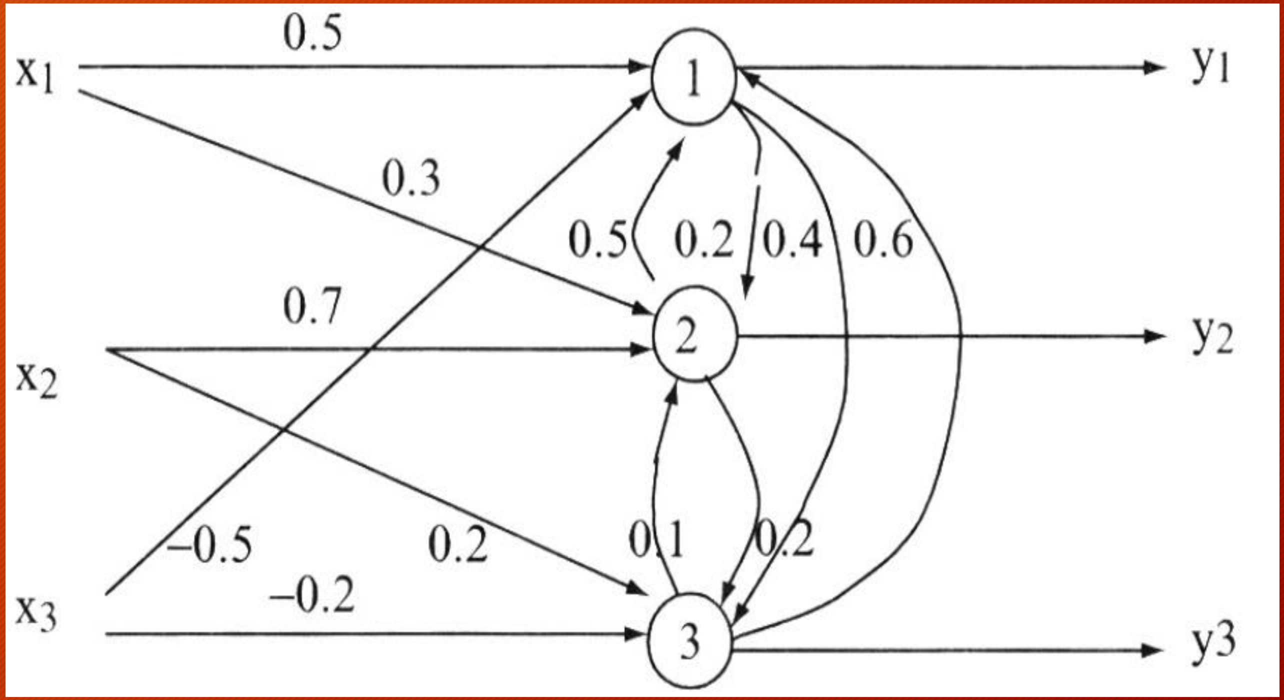
- Let W_{kj} denote the synaptic weights connecting input node j to neuron k. A neuron then learns by shifting synaptic weights from its inactive input nodes to its active input nodes.
- If a particular neuron wins the competition, each input node of that neuron relinquishes some proportion of its synaptic weight, and the weight relinquished is then distributed among the active input nodes.
- According to. the standard, competitive-learning rule, the change, Δw_{kj} applied to synaptic weight w_{kj} is defined. by

$$\Delta w_{kj} = \begin{cases} \eta(x_j - w_{kj}) & \text{if neuron } k \text{ wins the competition} \\ 0 & \text{if neuron } k \text{ loses the competition} \end{cases}$$

where η is the learning-rate parameter.

Example

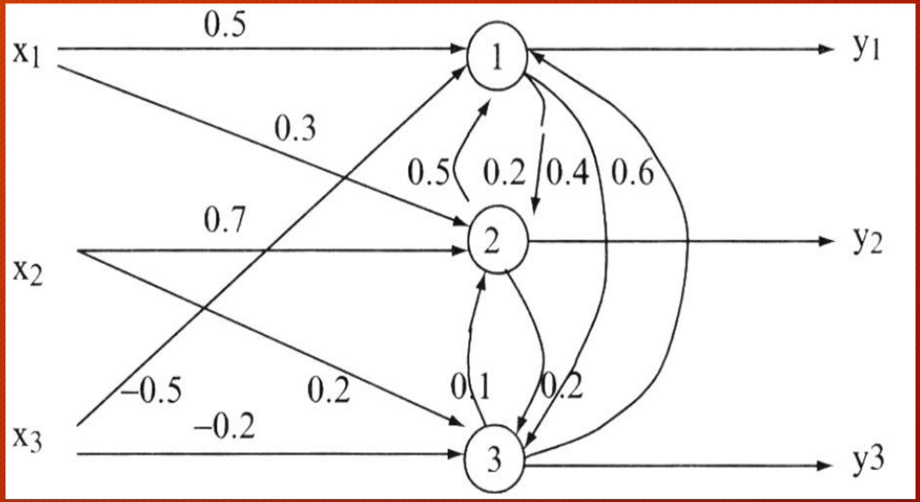
Suppose that there is a competitive network with three inputs and three outputs. The task is to group a set of 3-dimensional input samples into three clusters. The network is fully connected; there are connections between all inputs and outputs and there are also lateral connections between output nodes. Only local feedback weights are equal to zero, and these connections are not represented in the final architecture of the network. Output nodes are based on a linear-activation function with the bias value for all nodes equal to zero. The weight factors for all connections are given in the figure, and we assume that the network is already trained with some previous samples.



Example

Suppose that the new sample vector X has components

$$X = \{x_1, x_2, x_3\} = \{1, 0, 1\}$$



In the first, forward phase, the temporary outputs for competition are computed through their excitatory connections and their values are

$$net_1^* = 0.5 \cdot x_1 + (-0.5) \cdot x_3 = 0.5 \cdot 1 - 0.5 \cdot 1 = 0$$
$$net_2^* = 0.3 \cdot x_1 + 0.7 \cdot x_2 = 0.3 \cdot 1 + 0.7 \cdot 0 = 0.3$$
$$net_3^* = 0.2 \cdot x_2 + (-0.2) \cdot x_3 = 0.2 \cdot 0 - 0.2 \cdot 1 = -0.2$$

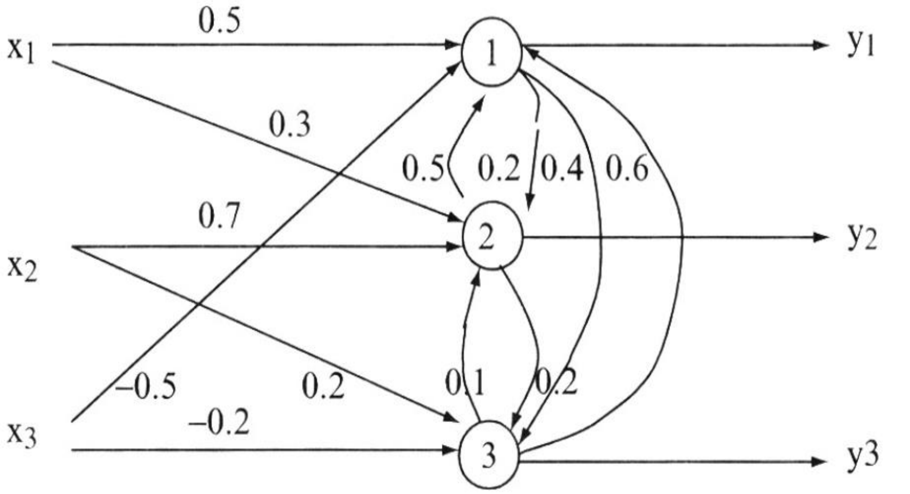
and after including lateral inhibitory connections

$$net_1 = net_1^* + 0.5 \cdot 0.3 + 0.6 \cdot (-0.2) = 0.03$$
$$net_2 = net_2^* + 0.2 \cdot 0 + 0.1 \cdot (-0.2) = 0.28 \text{ (maximum!!)}$$
$$net_3 = net_3^* + 0.4 \cdot 0 + 0.2 \cdot 0.3 = -0.14$$

Example

Competition between outputs shows that the highest output value is net2, and it is the winner. So the final outputs from the network for a given sample will be

$$Y = \{y_1, y_2, y_3\} = \{0, 1, 0\}$$



Based on the same sample, in the second phase of competitive learning, the procedure for a. weight factor's correction (only for the winning node y_2) starts. The results of the adaptation of the network, based on learning rate η = 0.2, are new weight factors:

$$\Delta w_{12} = 0.3 + 0.2(1 - 0.3) = 0.44$$
$$\Delta w_{22} = 0.7 + 0.2(0 - 0.7) = 0.56$$
$$\Delta w_{32} = 0.0 + 0.2(1 - 0.0) = 0.20$$