Soft Computing

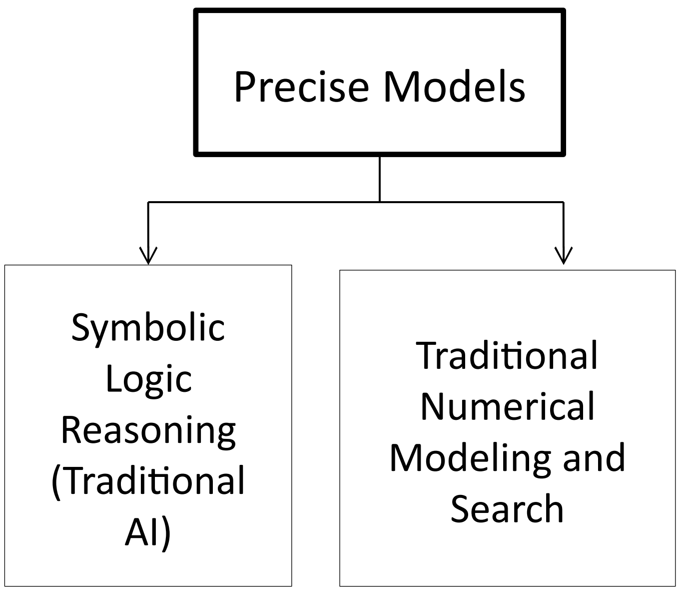
Introduction

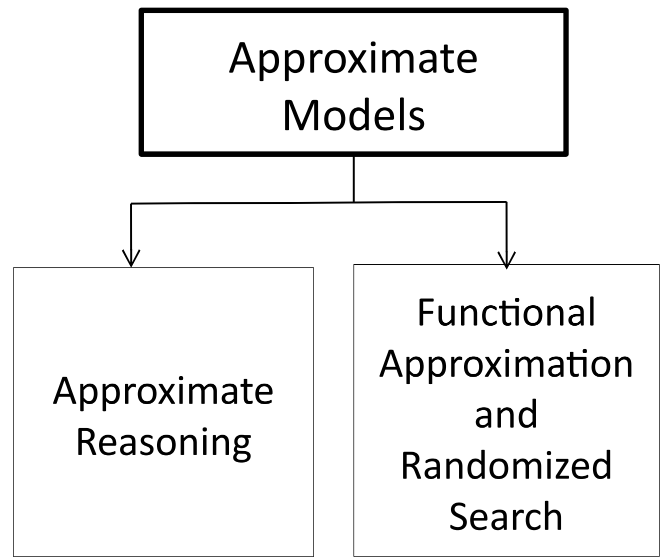
- Computer systems

Good at	Not Good at
Rule-based systems: doing what the programmer wants them to do	Dealing with noisy data
	Dealing with unknown environment data
	Massive parallelism
	Fault tolerance
	Adapting to circumstances

Problem Solving Techniques

- Hard Computing Soft Computing





Hard Computing

- Conventional computing
- Requires a precisely stated analytical model and often a lot of computation time.
- Many analytical models are valid for ideal cases.
- Usually real world problems exist in a non-ideal environment

Soft Computing

- Soft computing
  - was developed in the year 1981 by Lotfi A. Zadeh
- **Main goal**: develop intelligent machines to provide solutions to real world problems, which are not modeled, or too difficult to model mathematically
- **Aim:** Exploit the tolerance for Approximation, Uncertainty , Imprecision, and Partial Truth in order to achieve close resemblance with human like decision making
  - Approximation : here the model features are similar to the real ones, but not the same
  - Uncertainty : here we are not sure that the features of the model are the same as that of the entity (belief)
  - Imprecision : here the model features (quantities) are not the same as that of the real ones, but close to them.

Soft Computing

- Soft computing techniques
  - Fuzzy Logic
  - Neural Network
  - Genetic Algorithm
  - Probabilistic Computing
- It is the composition of methodologies designed to model and enable solution to real world problems.

Current Applications using Soft Computing

- Handwriting recognition
- Automotive systems and manufacturing
- Image processing and data compression
- Architecture
- Decision-support systems
- Power systems
- Neurofuzzy systems
- Fuzzy logic control

Artificial Neural Networks

- A neural net is an artificial representation of the human brain that tries to simulate its learning process.
- Information processing system inspired by the way the brain (biological nervous system) process information.
- ANN is composed of large number of interconnected processing elements working in unison to solve specific problems.
- Artificial Neural Networks (ANNs), like people, learn by example
- An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process.

Fuzzy Logic

- An organized method for dealing with imprecise data
- Proposed by Prof. Zadeh
- Superset of conventional (Boolean) logic that has been extended to handle the concept of partial truth
- Fuzzy knowledge: knowledge which is vague, imprecise, uncertain, ambiguous, inexact, or probabilistic in nature
- Human thinking and reasoning frequently involve fuzzy information
- Deals with real world vagueness
- Allows partial set membership
- Central notion of fuzzy systems is that truth values (in fuzzy logic) are indicated by a value on the range [0.0, 1.0], with 0.0 representing absolute Falseness and 1.0 representing absolute Truth.

Genetic Algorithms

- Genetic Algorithms are a way of solving problems by mimicking processes the nature uses to generate offsprings
- A genetic algorithm maintains a population of candidate solutions for the problem at hand
- Based on their characteristics, they may be more fit or less fit
- The more fit members are selected and they evolve by iteratively applying a set of operators

Hybrid Systems

- Hybrid systems employ more than one technology to solve a problem.
- Neural Networks, Fuzzy Logic, and Genetic Algorithms are three distinct technologies.
- Each of these technologies has advantages and disadvantages.
- It is therefore appropriate that hybridization of these three technologies are done so as to over come the weakness of one with the strength of other.

Neuro-Fuzzy Hybrid Systems

- Combines learning algorithm derived from or inspired by neural network theory to determine its parameters (fuzzy sets and fuzzy rules) by processing data samples
  - It can handle any kind of information (numeric, linguistic, logical, etc.).
  - It can manage imprecise, partial, vague or imperfect information.
  - It can resolve conflicts by collaboration and aggregation.
  - It has self-learning, self-organizing and self-tuning capabilities
  - It doesn‘t need prior knowledge of relationships of data.
  - It can mimic human decision-making process.
  - It makes computation fast by using fuzzy number operations.

Neuro Genetic Hybrid Systems

- Genetic algorithms (GAs) are applied in ANN design
  - **Topology optimization**, GA is used to *select a topology* (number of hidden layers, number of hidden nodes, interconnection pattern) for the ANN which in turn is trained using some training scheme, most commonly back propagation.
  - **Genetic training algorithms**, the learning of an ANN is formulated as a *weight optimization problem*, usually using the inverse mean squared error as a fitness measure.
  - **Control parameters optimization-**learning rate, momentum rate, tolerance level, etc., can also be optimized using GAs.

Fuzzy Genetic Hybrid Systems

- The optimization abilities of GAs are used *to develop the best set of rules* to be used by a fuzzy inference engine, and to *optimize the choice of membership functions*.
- A particular use of GAs is in *fuzzy classification systems*, where an object is classified on the basis of the linguistic values of the object attributes.
- The most difficult part of building a system like this is to find *the* appropriate set of fuzzy rules.
  - First approach: obtain knowledge from experts and translate into fuzzy rules.
    - Time consuming
  - Second approach: obtain the fuzzy rules through machine learning, whereby the knowledge is automatically extracted or deduced from sample cases.

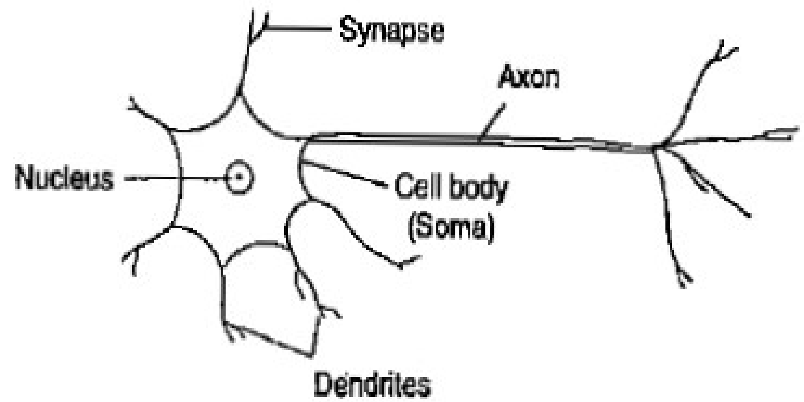
Artificial Neural Networks

Artificial Neural Networks

- **Neural network:** information processing paradigm inspired by biological nervous systems, such as our brain
- Structure: large number of highly interconnected processing elements (*neurons*) working together
- Like people, they *learn from experience* (by example)
- Objective: Develop a computational device for modeling the brain to perform various computational tasks at a faster rate than traditional systems

Biological Neuron

- Dendrites: Input
- Cell body: Processor
- Synapse: Link
- Axon: Output



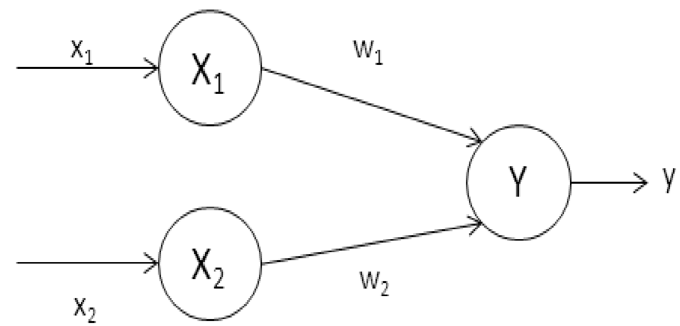
- Basically, a biological neuron receives inputs from other sources, combines them in some way, performs a generally nonlinear operation on the result, and then output the final result.

Terminology relationships between biological and artificial neurons

Biological neuron	Artificial neuron
Cell	Neuron
Dendrites	Weights or interconnections
Soma	Net input
Axon	Output

Artificial Neural Networks

- Activation or activity level of neuron: internal state of neuron
- A neuron can send only one signal at a time
- The signal is broadcast to several other neurons

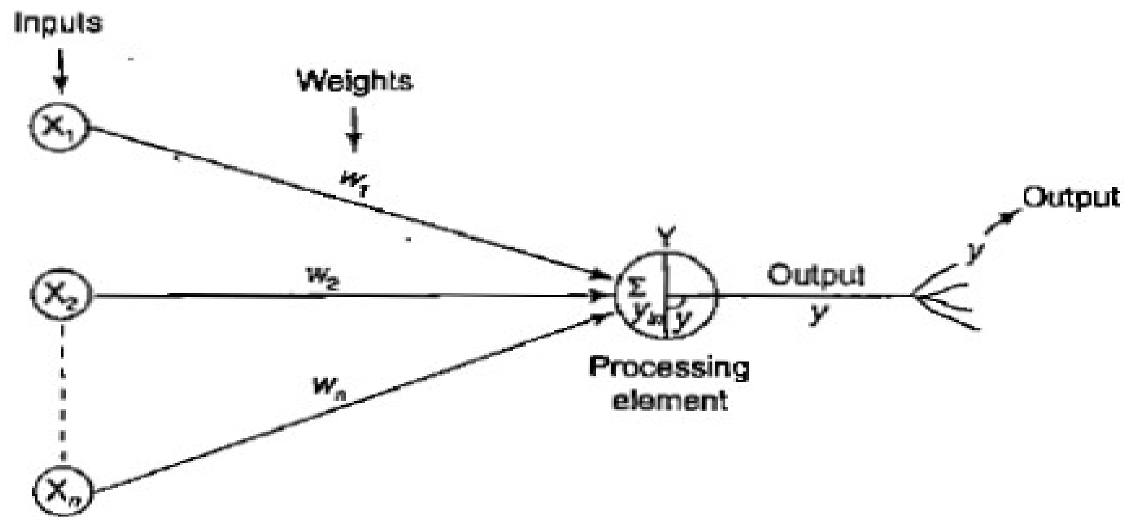


- $Y_{in}=x_1w_1+x_2w_2$
- $y=f(y_{in})$

Artificial Neural Networks

- ANNs are based on the following assumptions
  - Information processing occurs at neurons
  - Signals are passed between neurons over connection links
  - Each connection link has an associated weight that multiplies the signal transmitted
  - Each neuron applies an activation function (usually non linear) to its net input (sum of weighted input signals) to determine its output signal
- ANNs collective behaviour is characterized by their
  - ability to learn, recall and generalize training patterns or data
  - They model the network of original neurons found in brain
- Hence they are called neurons or artificial neurons

Mathematical representation



Net Input, $y_{in} = x_1 w_1 + x_2 w_2 + \cdots + x_n w_n = \sum_{i=1}^{n} x_i w_i$

Weight: Strength of synapse connecting the input and the output neurons
Positive weight: Excitatory synapse
Negative weight: Inhibitory synapse

Brain Vs. Computer

Criteria	Brain	ANN
Speed	milliseconds	nanoseconds
Processing	perform massive parallel operations simultaneously	perform several parallel operations simultaneously
Size and complexity	No. of neurons: 10^{11} No. of interconnections: 10^{15} Complexity higher	Size and complexity depends on the application and network designer
Storage capacity (memory)	-Interconnections or in synapse strength -New information added without destroying old data -Memory may fail to recollect stored information	-Contiguous memory locations -Overload memory location -Stored information can be retrieved -Adaptability more
Tolerance	Fault tolerant	No fault tolerance
Control mechanism	No control unit monitoring for brain	Control unit transfers and control precise scalar values from unit to unit

Characteristics of ANN

- It is a neutrally implemented mathematical model
- There exists a large number of highly interconnected processing elements called neurons in an ANN.
- The interconnections with their weighted linkages hold the informative knowledge.
- The input signals arrive at the processing elements through connections and connecting weights.
- The processing elements of the ANN have the ability to learn, recall and generalize from the given data by suitable assignment or adjustment of weights.
- The computational power can be demonstrated only by the collective behavior of neurons, and it should be noted that no single neuron carries specific information.

Artificial Neural Networks

- The models of ANN are specified by the three basic entities namely:
  - the model's synaptic interconnections
  - the training or learning rules adopted for updating and adjusting the connection weights
  - their activation functions.

Connections

- ANN consists of a set of highly interconnected processing elements (neurons)
- Each processing element is connected to other processing elements or to itself, through weights
- The arrangement of neurons to form layers and the connection pattern formed within and between layers is called the *network architecture*

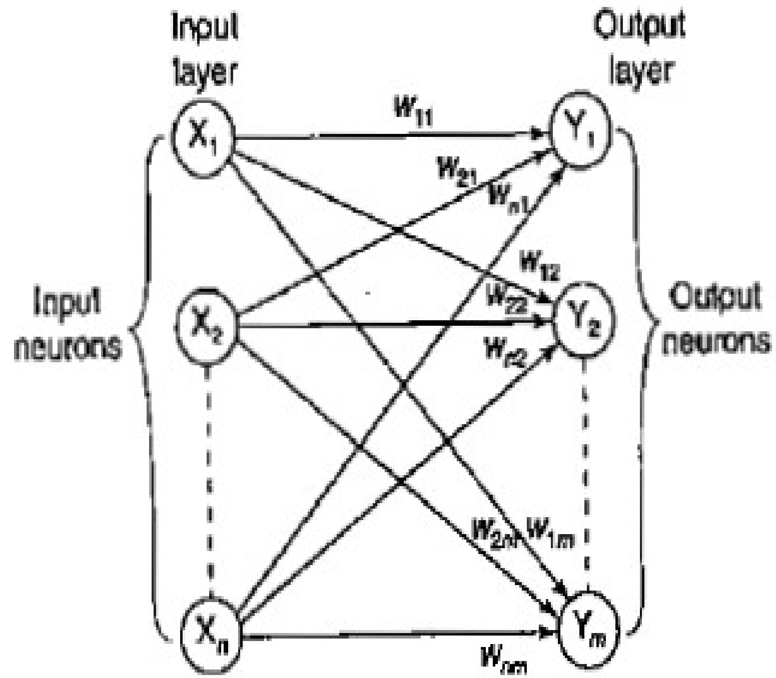
Basic types of neuron connection architectures

- Basically, neural nets are classified into single layer or multilayer neural nets.
- Layer is formed by taking a processing element and combining it with other processing elements

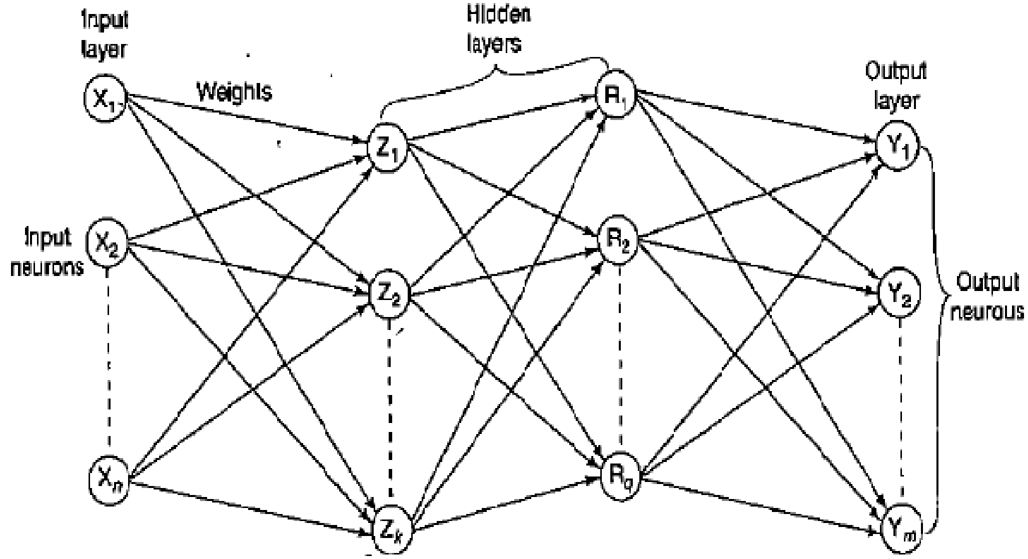
1. Single-layer feed-forward network
2. Multilayer feed-forward network
3. Single node with its own feedback
4. Single-layer recurrent network
5. Multilayer recurrent network

Single Layer Feed-Forward Network

- First and simplest type of ANN
- Information moves in only one direction, forward, from the input nodes, to the output nodes
- No cycles or loops in the network
- Series of outputs, one per node is produced



Multilayer Feed-Forward Network

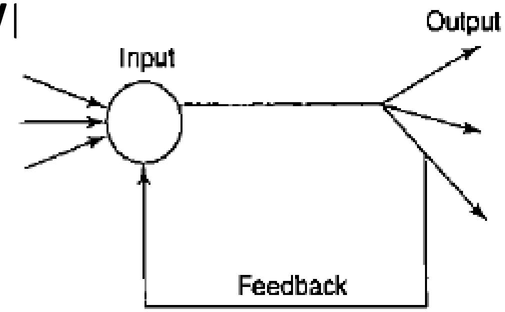


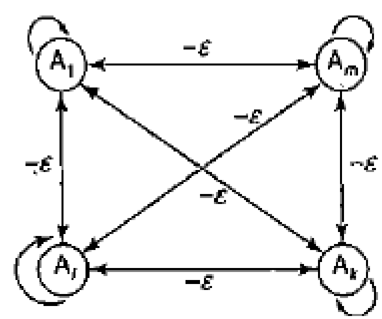
- Formed *by* the interconnection of several layers
- Input layer receives the input and buffers the input signal
- Output layer generates the output of the network
- Any layer that is formed between the input and output layers is called *hidden layer*
- Zero to several hidden layers in an ANN
- Fully connected network: Every output from one layer is connected to every node in the next layer.

Single node with its own feedback

- **Feed-forward Networks:** No neuron in the output layer is input to the same or preceding layers
- **Feedback Networks:** Outputs directed back as inputs to same or preceding layer nodes
- **Lateral Feedback:** Feedback of output of the processing elements is directed back as input to the processing elements in the same layer
- **Recurrent networks:** feedback networks with a closed loop
- **Competitive Net:** competitive interconnections having fixed weights of *-ε.* This net is called *Maxnet*

- Single node with its ow

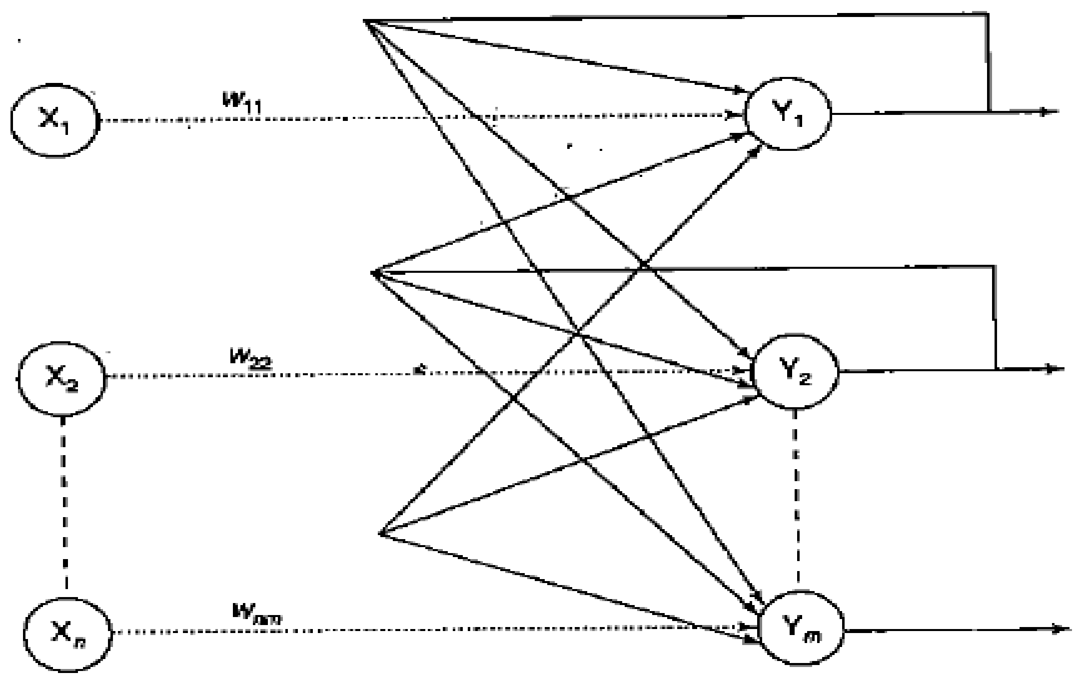




- Cc ts

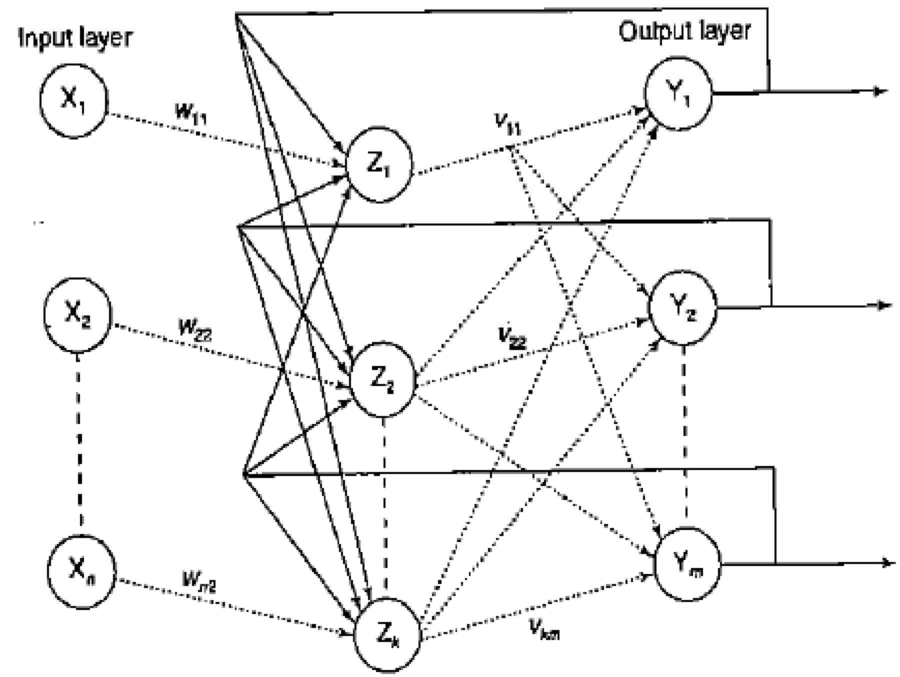
Single Layer Recurrent Network

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

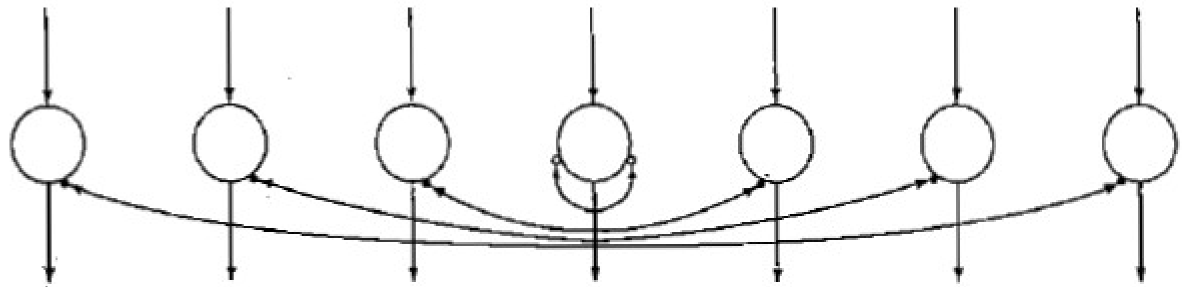


Multilayer Recurrent Network

- A processing element's output can be directed back to the processing element itself or to other processing elements or to both



Lateral Inhibition Structure



- Network with lateral feed back.
- Called on-center-off-surround or lateral inhibition structure
- Each neuron receives two inputs
  - Excitatory inputs from nearby processing elements
  - Inhibitory inputs from more distantly located processing elements
- Connection with open circle: excitatory connections
- Links with solid connective circles: inhibitory connections

Learning

- Learning or training
  - a neural network adapts itself to a stimulus by making proper parameter adjustments
- Broadly, there are two kinds learning in ANNs:
  - Parameter learning: the connecting weights are updated
  - Structural learning: It focuses on the change in network structure (which includes the number of processing elements as well as their connection types)
- The above two types of learning can be performed simultaneously or separately

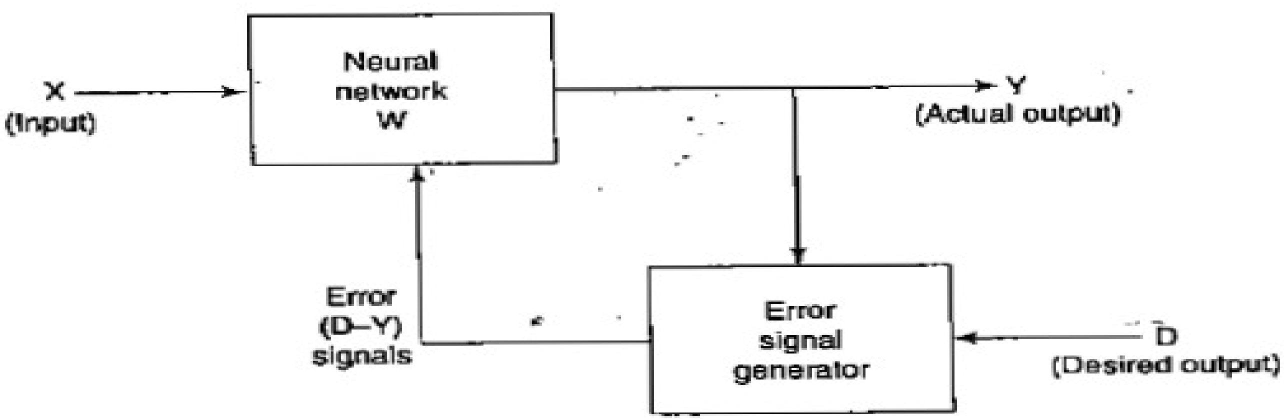
Learning

- Learning in an ANN can be generally classified into three categories as:
  - supervised learning
  - unsupervised learning
  - reinforcement learning

Supervised Learning

- Learning here is performed with the help of a teacher.
  - A network is fed with a set of training samples (inputs and corresponding output),
  - It uses these samples to learn the general relationship between the inputs and the outputs.
- The input vector along with the target vector is called *training pair.*
- The network is informed precisely about what should be emitted as output.

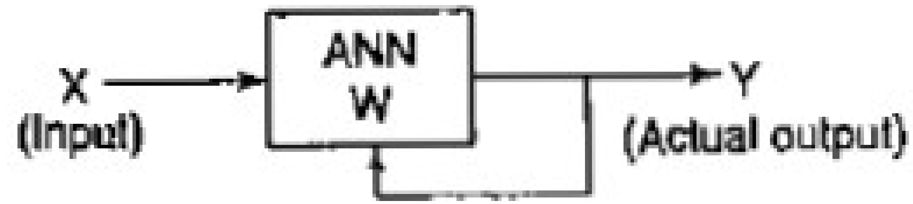
Supervised Learning



Unsupervised Learning

- Sequence of input vectors is provided
- No target vectors specified
- The input vectors of similar type are grouped without the use of training
- Organizes the input patterns to clusters
- When a new input pattern is applied, the neural network gives an output response indicating the cluster to which the input pattern belongs.
- New class is generated if the input does not belong to any pattern class

Unsupervised Learning

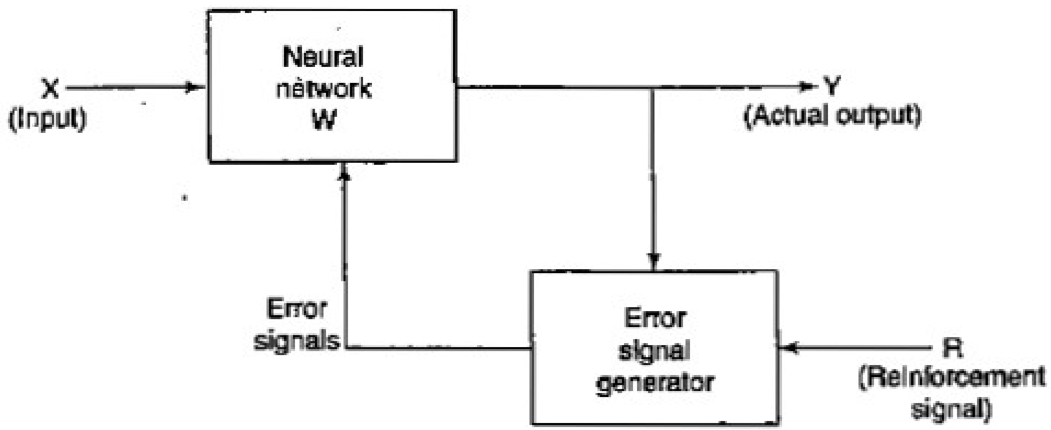


- No feedback
- Network must itself discover patterns, similarities, features or categories from the input data and relations for the input data over the output
- While discovering all these features, the network undergoes changes in its parameters.
- This process is called **self organizing**: exact clusters will be formed by discovering similarities and dissimilarities among the objects

Reinforcement Learning

- Similar to supervised learning
- Correct target value not known
- Here only less information or critic information is available, not the exact information.
  - Eg: Output is only 50% correct
- The learning based on this critic information is called *reinforcement learning* and the feedback sent is called *reinforcement signal*

Reinforcement Learning



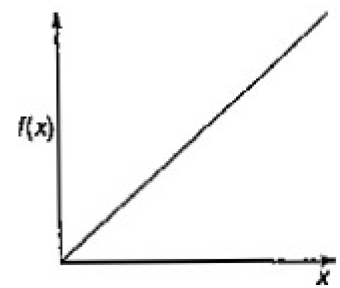
- Feedback obtained is only evaluative and not instructive
- External reinforcement signals are processed in the critic signal generator
- The obtained critic signals are sent to ANN for adjustment of weights
- The reinforcement learning is also called learning with a critic

Activation Functions

- A function that transforms the values or states the conditions for the decision of the output neuron is known as an activation function
- There are several activation functions

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

$f(x) = x$ for all x



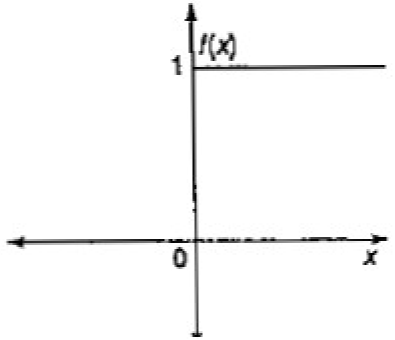
The output here remains the same as input.

The input layer uses the identity activation function

Activation Functions

2. Binary step function

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

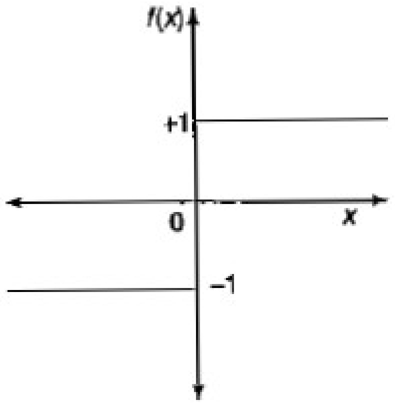


- θ represents the threshold value
- This function is most widely used in single-layer

nets to convert the net input to an output that is a binary (1 or 0).

3. Bipolar step function

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



- θ represents the threshold value

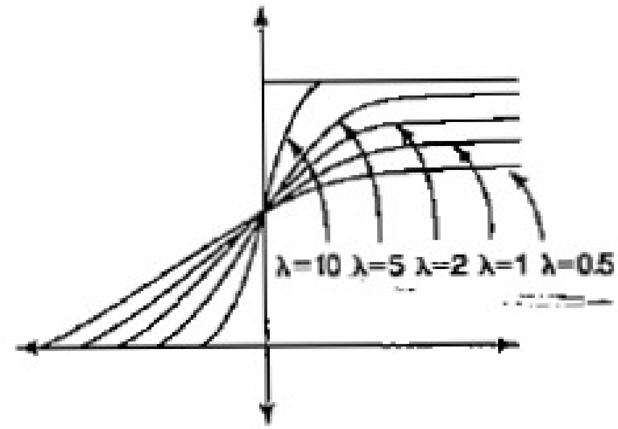
- This function is also used in single-layer nets to convert the net input to an output that is bipolar(+ 1 or -1).

Activation Functions

4. *Sigmoidal functions*

- Binary Sigmoid Function

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

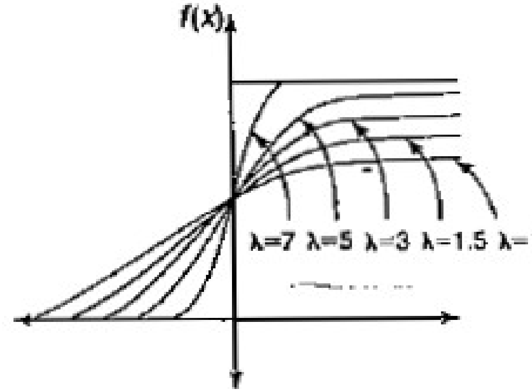


- where λ is the steepness parameter
- also referred as logistic sigmoid function or unipolar sigmoid function.
- It is a function which is plotted as 'S' shaped graph.
- Derivative: $f'(x)=\lambda f(x)[1-f(x)]$
- Value Range : 0 to 1

Activation Functions

Bipolar sigmoid function

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



λ is the steepness parameter and range of sigmoid function is between -1 and +1

- Derivative: $f'(x) = \frac{\lambda}{2}[1 + f(x)][1 - f(x)]$
- **Hyperbolic tangent function**

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

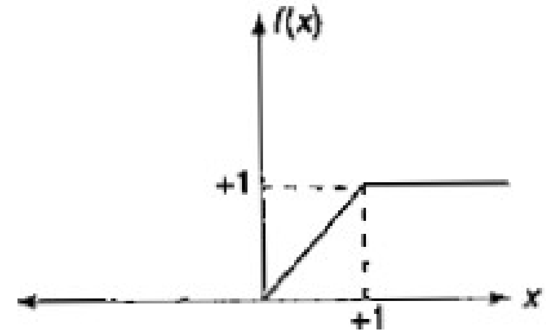
- Derivative: $h'(x) = [1 + h(x)][1 - h(x)]$

Activation Functions

- 5. Ramp function

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

- Graphical Representation



Important Terminologies

- **Weights**
  - Each neuron is connected to other neuron with a weighted communication link
  - Weights contain information about the input signal
  - This information is used to solve a problem
  - Weights represented as a weight matrix (connection matrix)
  - $w_i=[w_{i1}, w_{i2}, \ldots.., w_{in}]^T$, i=1,2,…n
  - The ANN can be realized by finding an appropriate weight matrix

$$W = \begin{bmatrix} w_1^T \\ w_2^T \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ w_n^T \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & \cdot & \cdot & \cdot & \cdot & w_{1m} \\ w_{21} & w_{22} & \cdot & \cdot & \cdot & \cdot & w_{2m} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ w_{n1} & w_{n2} & \cdot & \cdot & \cdot & \cdot & w_{nm} \end{bmatrix}$$

Bias

- Bias has impact in calculating the net input
- A component x_0=1 is added to the input vector

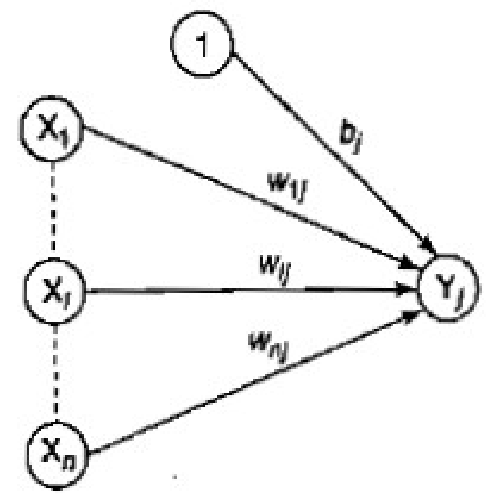
  $X=(1, X_1, X_2, \ldots.X_n)$
- Bias is considered like weight, $w_{0j}=b_j$
- Two types: Positive bias-increases net input

  Negative bias: decreases net input

$$y_{inj} = \sum_{i=0}^{n} x_i w_{ij} = x_0 w_{0j} + x_1 w_{1j} + x_2 w_{2j} + \cdots + x_n w_{nj}$$

$$= w_{0j} + \sum_{i=1}^{n} x_i w_{ij}$$

$$y_{inj} = b_j + \sum_{i=1}^{n} x_i w_{ij}$$



Threshold

- Value based on which the final output of the network is calculated
- Used in activation function
- Comparison between the calculated net input and threshold is made.
- Activation function using threshold can be defined as

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

θ is the fixed threshold value

Important Terminologies

- **Learning Rate**
  - Denoted by α
  - Ranges from 0 to 1
  - Controls the weight adjustment at each step of training
  - Determines the rate of learning at each step
- **Momentum Factor**
  - Convergence made faster by adding momentum to the weight updation process.
  - Done in back propagation network
  - The weights from one or more previous training patterns must be saved
  - Helps in reasonably large weight adjustments

Important Terminologies

- **Vigilance Parameter**
  - Denoted by ρ
  - Used in adaptive resonance theory network
  - Controls the degree of similarity required for patterns to be assigned to the same cluster unit
  - Range: 0.7 to 1