Neural Network

1) **What are the basic differences between the biological neuron of human brain and the machine?** 05, 03

Human brain	Machine
1. It is more intelligent.	1.It is less intelligent than brain
2. The processing speed is slower	2. The processing speed is faster
3. The capability of parallel processing is higher	3. The capability of parallel processing is lower.
4. Fault tolerance is higher.	4. Fault tolerance is lower.
5. It can learn things from its environment itself.	5. It can not learn things from its environment itself.
6. The capability of serial processi is lower.	6. The capability of serial processing is ~~lower~~ higher.
7. The knowledge required to solve any problem comes from many different sources in here.	7. There is no chance for gathering the knowledge required to solve any problem from many different sources i here.
8. Performance never be zero here.	8. Performance can be zero if one mistake in large program.
9. Distributed processing is occurre automatically.	9. It is not possible here to occur distributed process automatically.

2) **What are the advantages of human brain over machine?** 06,

1. Human brain is more intelligent than computer.
2. The brain with its parallel design is able to represent and store knowledge in an accessible.
3. The knowledge require to solve any problems comes from many different sources in human brain
4. Human brain can learn things from its environment itself.
5. In human brain distributed processing is occurred automatically.
6. Fault tolerance is higher than a machine.

3) **Define the following terms: a) Graceful degradation b) Fault tolerance c) Pattern recognition d) feature space e) feature vector.** 06, 05, 01, 02

Graceful degradation: Graceful degradation is a state where the performance of the system slowly falls from a high level to a reduced level but without dropping castrophically to zero.

Fault tolerance: Fault tolerance is a vital feature of the operation of the brain, since every day a few neuron die as a part of the natural part of the natural course of events. More are lost if the brain gets bumped about, but it continues working as if nothing has happened.

Pattern recognition: It can be defined as "the act of taking new data and taking an action based on category of data". A pattern recognition system can be considered as a two stage device, the first stage is feature extraction and the second is classification.

Feature vector: If we make n measurements on our input pattern, each of which is unique feature then we can use algebraic notation to create a set of these features and call it a feature vector.

Feature space: The dimensionality of the vector that is the number of elements in it creates n dimensional feature space.

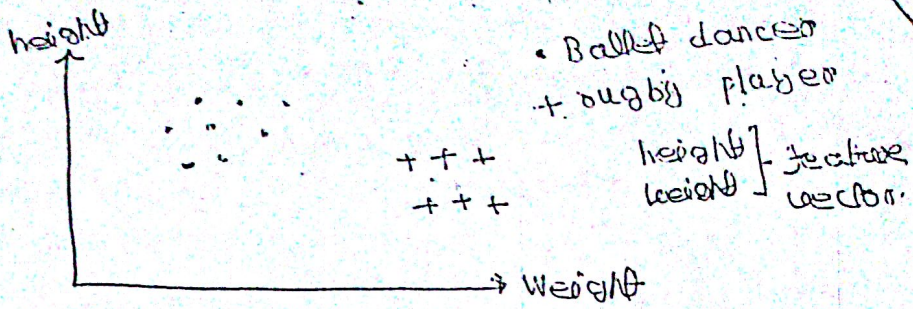


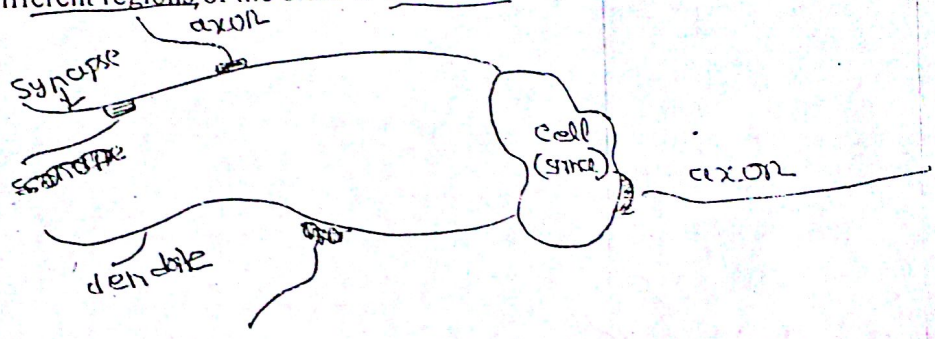
Fig: A two dimensional Euclidian feature space

4) Discuss briefly the structure of the brain. Or, briefly discuss about the Biological Neuron.

→ The brain contains approximately ten thousand million (10^{10}) basic units called neurons. Each of these neuron is connected to about ten thousand (10^4) others.

→ The neuron is the basic unit of the brain and it is a stand-alone analogue logical processing unit.

→ The neuron from two main types local processing interneuron cells that have their input and output connection over about 100μ and output cells that connect different regions of the brain to each other.



Soma: The soma is the body of the neuron.

Dendrite: Attached to the soma are long, irregularly shaped filaments called dendrites. The dendrite act as the connection through which all the inputs to the neuron arrive.

Axon: Another type of nerve process attached to the soma is called on axon. This is electrically active unlike the dendrite and serves as the output channel of the neuron. It is a nonlinear threshold device producing a voltage call the action potential.

Synapse: The axon terminates in a specialized contact called a synapse that couples the axon with the dendrite of another cell.

Neurotransmitter: The synapse releases chemicals called neurotransmitters when its potential is raised sufficient by the action potential.

5) Discuss the learning in Biological neuron

Learning is through to occur when modification are made to the effective coupling between one cell and another at the synaptijunction. Neurotransmitters released from the synaptic vesicles diffuse across the synaptic cleft and trigger the receivers on the dendrite, the release of more neurotransmitters. This has the effects of opening more gates on the dendrite on the post synaptic side of the junction and so increasing coupling effect of the two sides.

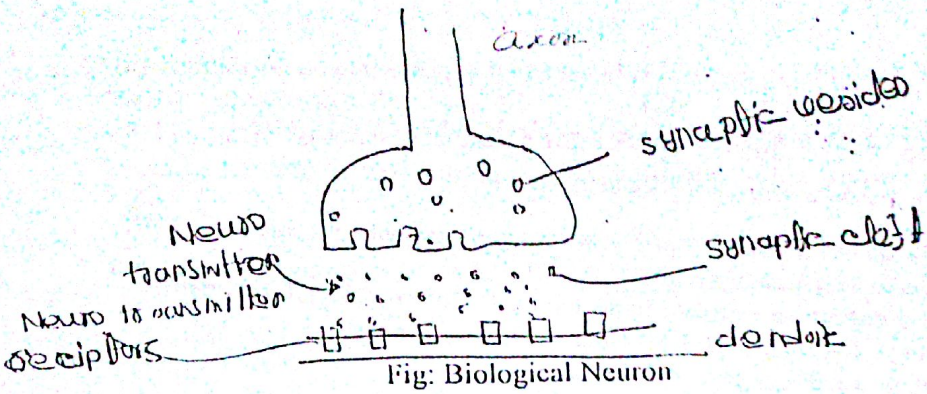


Fig: Biological Neuron

6) **What is the functionality of the Cerebral Cortex?** 09

The largest part of the brain is the cerebral which occupies most of the interior of the skull. They are layered structures. The most complex, being the outer layer known as the cerebral cortex in which the nerve cell are extremely densely packed to allow great interconnectivity. Its function is not fully understood. Thus the cerebral cortex seems to be the seat of the higher order functions of the brain and the core of intelligence.

Ch-6 10

7) **Write down the features of Hopfield Networks.** 05.

The major features of Hopfield networks are:

1. It is a fully connected network.
2. It is also symmetrically weighted network.
3. Each node has a single layer perception.
4. The node calculates the weighted sum of their inputs minus the threshold value.
5. The network takes only 2-state inputs there can be binary (0, 1) or bipolar (-1, 1).
6. It converges to local minima which may not give the optimal solution.
7. It acts as auto associative memory.

Ch-6

8) **For Hopfield net what are synchronous updating, asynchronous updating and metastable states?** (08)

Synchronous updating: An operation is known as synchronous updating when one update occurs across the entire network.

Asynchronous updating: An approach is called asynchronous updating when a node is chosen as random and update its output according to the output it receiving. The process is repeating. In asynchronous updating, the change in output of one is affected the state of the system and can therefore affects the next nodes change.

Metastable state: If there has been a sufficient inference between patterns to form intermediate local minima states that were not taught to the network, but which the network thinks are perfectly acceptable solution, the states are known as metastable states.

9) **Write down the Hopfield network algorithm.** 06.

06

1. Assign connection weights.

$$w_{ij} = \begin{cases} \sum_{s=0}^{M-1} x_i^s x_j^s & i \neq j \\ 0 & i = j \end{cases}, \quad 0 \le i, j \le N-1$$

2. Initialize with unknown patterns.

$M_i(0) = x_i$. $0 \le i \le N-1$ where, $M_i(t)$ = output of node i at time t.

3. Iterate until convergence.

$$M_i(t+1) = f_h\left[\sum_{i=0}^{N-1} w_{ij}\, M_j(t)\right] \quad 0 \le j \le N-1$$

Where f_h is the hard-limiting non-linearity function.

10) With suitable example, explain how weight vector is calculated from stored pattern in Hopfield network. 06,05,02
Or, How can u calculate weights from the patterns $p_1=\{1,-1,1,-1\}$, $p_2=\{-1,-1,-1,-1\}$, $p_3=\{1,1,-1,1\}$, $p_4=\{1,1,1,1\}$ using Hopfield network.

Our weight matrix is,

$$w_{ij} = \begin{bmatrix} w_{00} & w_{01} & w_{02} & w_{03} \\ w_{10} & w_{11} & w_{12} & w_{13} \\ w_{20} & w_{21} & w_{22} & w_{23} \\ w_{30} & w_{31} & w_{32} & w_{33} \end{bmatrix}$$

So here for every pattern we have the inputs X_0, X_1, X_2, X_3.
When i=j then the weight is 0. So we can set these positions as zero. This is a symmetric weighted figure. So here $w_{10}=w_{10}$ and so on. So here,

$$w_{ij} = \begin{cases} \sum_{s=0}^{m-1} x_i^s x_j^s & \text{for } i \neq j \\ 0 & \text{for } i = j \end{cases}$$

Finally the matrix is,

$$\begin{bmatrix} 0 & 2 & 2 & 2 \\ 2 & 0 & 0 & 2 \\ 2 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 \end{bmatrix}$$

11) With appropriate notations briefly describe the "Kohonen Network Algorithm". 05, 03,02

1. Initialize network
   Define , $w_{ij}(0)$ $(0 \le i \le n-1)$
   Set the initial radius of the neighborhood around node j, $N_j(0)$ to be large.
2. Present input
   input : $x_0(0), x_1(0), \ldots, x_{n-1}(0)$
3. Calculate distance
   $$d_j = \sum_{i=0}^{N-1} \left(x_i(t) - w_{ij}(t)\right)^2$$
4. Select minimum distance
5. Update weights. New weights are,
   $$w_{ij}(t+1) = w_{ij}(t) + \eta(t)\left(x_i(t) - w_{ij}(t)\right)$$
   Also update the radius. New radius is,
   $$N_j^*(t+1) = N_j(t) - \eta N_j(t)$$
6. Repeat by going to step 2.

12) What is the necessity of the reduction of neighborhood size in KSON? 04

→ The effect of shrinking the neighborhood is to localize areas of similar activities.
→ This ensures that clusters from accurate internal representations of the training data as well as causing the network to converge to a solution within a predefined time limit.

→ The neighborhood size

13) Briefly discuss the basic features of a Biological Neuron? What is an Artificial Neuron? 04

→ The basic features of a biological neuron are to add up its inputs and to produce an output if its sum is greater than some value, known as threshold value.

→ The inputs to the neuron arrive along the dendrite, which are connected to the outputs from other neuron by specialized junctions called synapse.

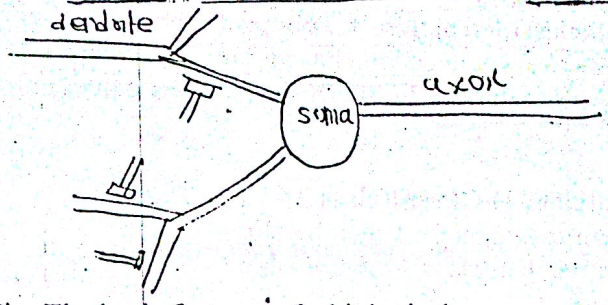


Fig: The basic features of a biological neuron

The features are summaries as follows:

1. ... The inputs from a neuron is either on or off
2. ... The output depends only on the inputs
3. ... A certain number must be on at any one time in order to make the neuron fire.

Artificial neuron: One model of the neuron with capturing the above with important features is called artificial neuron.

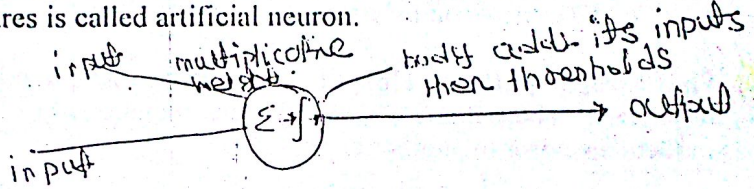


Fig: Outline of the basic model

14) What is discriminate function?

If we could define a dividing boundary for our data classification would become a process of deciding on which side of the boundary any new input falls. The mathematical definition of such a decision boundary is called a "Discriminating function". It is a function that maps our input features onto a classification space, by defining a plane that would separate the clusters.

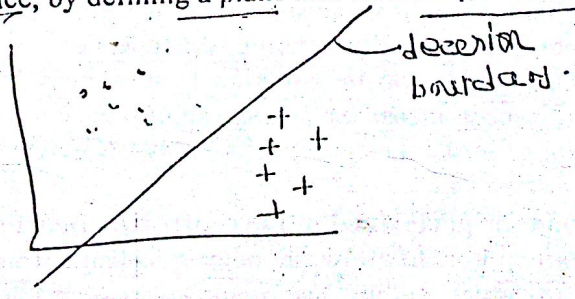


Fig: A linear classification decision boundary

15) Describe the classification techniques? Briefly discuss about the nearest neighbor classification.

There are two classification techniques:

1. **Numeric:** The technique which includes deterministic and statistical measures which can be considered as measures made on the geometric pattern space.
2. **Non-Numeric:** These techniques are those which take us into the domain of symbolic processing that is dealt with by such methods as fuzzy sets.

Nearest Neighbour Classification: we wish to decide to which of the two classes, the unclassified pattern X belongs to. Nearest neighbor classification techniques, Make a decision based on the shortest distance of the neighboring class samples. Formally that defines a discriminate function f(x) by,

f(x)= Closest(class 1)-Closest(class 2)
if f(x) is negative then class 1 membership
if f(x) is positive then class 2 membership

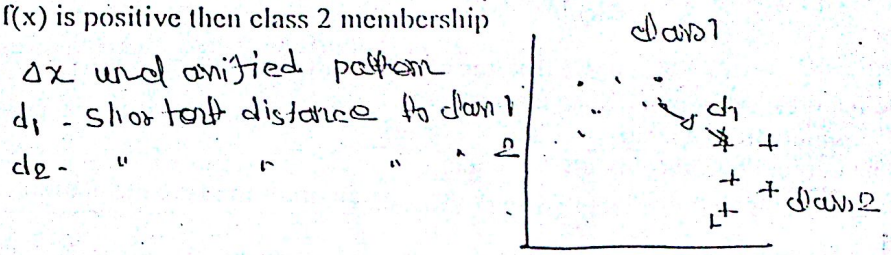


Fig: Classification by comparison to the nearest neighbor

(03) **16) What is rogue pattern? How the problem can be solved?** The problem of nearest neighbour.

Rogue pattern: The pattern which has class membership of one class but does in fact lies closer to another class.

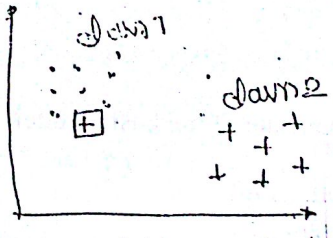


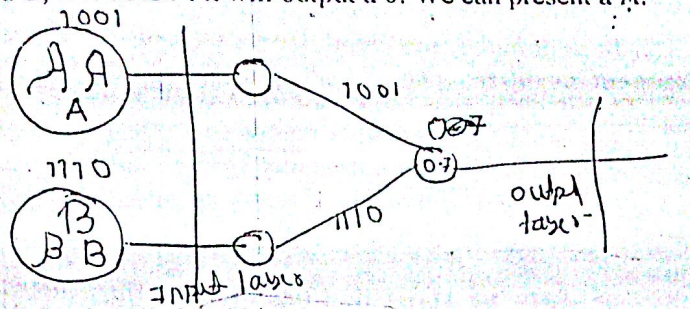
Fig: Measuring to the nearest neighbor can produce errors in classification if a rogue sample is selected.

Solution: If our unclassified input is measured against the rouge sample, it will invariably result in misclassification. The solution to this fairly basic problem is to take several distances for the measurement and then finding out their average and finally making the decision on the basis of the averaged distance value. So the solution to this fairly basic problem is to take several distances measures against many class samples such that the effect of any range measurement made is likely to be averaged out.

17) How learning is performed in Perceptron or neuron? 04,12

The guiding principle is to allow the neuron to learn from its mistakes. If it produces an incorrect output, we want to reduce the changes of that happening again. If it comes up with correct output, than we need to do nothing. This means that for the network to learn, we want to increase the weights on the active inputs when we want the outputs to be active and to decrease them when we want to output to be inactive. If we set up the neuron with random weights on its inputs lines, corresponding to a starting state in which it knows nothing, we can present an A.

If the weighted sum of the inputs exceeds the threshold, it will output a 1, we can present a B, if it doesn't it will output a 0. We can present a A.



This learning rule is called Hebbian learning.

18) What is supervised and unsupervised learning? What are the differences between them? 06,05,02

09 Supervised learning is a machine learning technique for creating a function from training data. But unsupervised learning a type of machine learning where manual labels of inputs are not used. There are some differences between them which are now given below:

Supervised	Unsupervised
1. The learning is guided by the knowledge what we want to achieve.	1. There are no previous knowledge for learning.
2. It relies on an external training response.	2. It doesn't rely on an external training response.
3. It performs a task using a set of human prepared examples.	3. There are no prepared examples here for performing the task.
4. The output of the function can be continuous value.	4. The output function can be clustering value.

08 **19) Write down the perceptron learning algorithm with necessary documentation? Also write the Widrow Hoff Delta Rule? 06,05, 02**

1. Initialize weights and thresholds.
   $w_i(t)$= small random values.
2. Present input and desired output
   Input: x0,x1,x2,……,xn
   Output: d(t)
3. Calculate the actual output

$$y(t) = f_h\left[\sum_{i=0}^{n} (w_i(t)\, x_i(t))\right]$$

4. Adapt weights
   If correct $w_i(t+1) = w_i(t)$

If output 0, should be 1(class A) $w_i(t+1) = w_i(t) + x_i(t)$
If output 1, should be 0(class B) $w_i(t+1) = w_i(t) - x_i(t)$

3. Adapt weights(modified)
   If correct $w_i(t+1) = w_i(t)$

If output 0, should be 1(class A) $w_i(t+1) = w_i(t) + \eta x_i(t)$
If output 1, should be 0(class B) $w_i(t+1) = w_i(t) - \eta x_i(t)$

Where $0 \le \eta \le 1$.

For the Widrow-Huff Delta Rule the adaption of the weight procedure is

Error term $\Delta = d(t) - y(t)$

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

$d(t) = +1$, if inputs from class A

0, if inputs from class B

20) What is Credit assignment problem? How can u solve this problem? 05,09

Credit assignment problem: This a type of problem happens when the network is unable to determine which of the input weights should be increased and which should not and so is unable to work out what changes should be made to produce a better solution next time.

We can solve this problem by using hard limiting function and sigmoidal function.

21) What do you mean by Hard limiting thresholding function and Sigmoid function? What are the advantages of sigmoid function over hard limiting thresholding function? 06, 05,

Hard limiting thresholding function: It makes the inputs from the outputs and adjusting the model so that difficulty by the tracing the root of the problem can be solved. It is related with single layer perceptron.

Sigmoidal function: It is a mathematical function that produces a sigmoid curve. It is a nonlinear thresholding function, since layers of the perceptron units using linear function are no more powerful than a suitably chosen single layer. It is related with multi-layer perceptron.

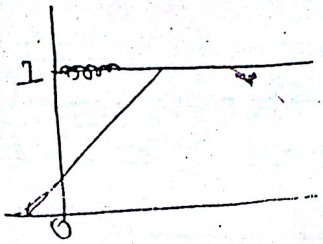


Fig: Hard limiting thresholding function

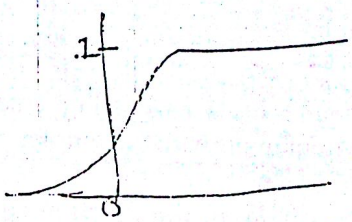


Fig: Sigmoidal Function

Advantages:

1. The use of the sigmoidal function means that enough information's about the output is available to units in earlier layer. So that for these enough information's these units can have their weights adjusted so as to decrease the error next time.
2. It is a nonlinear thresholding function. So it is quite like the step function and so should demonstrate behavior of similar nature.

22) Define gradientdesent and basins of attraction? 06, 05, 04

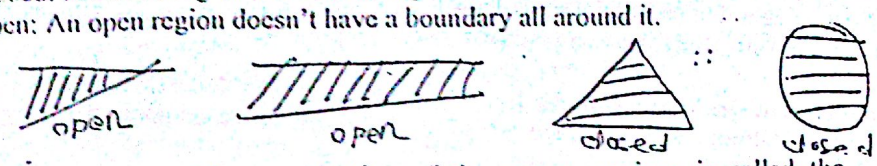
Gradientdesent: In the gradientdesent the energy function is calculated and changes are made in the steepest downward direction. This is guaranteed to find a solution where the energy landscape is simple.

Basins of attraction: It represents the solutions to the values of the weights that produce the correct output from a given inputs.

23) Define convex hull and arbitrary shapes. Define the Kolmogrov theorem. 06, 03, 02

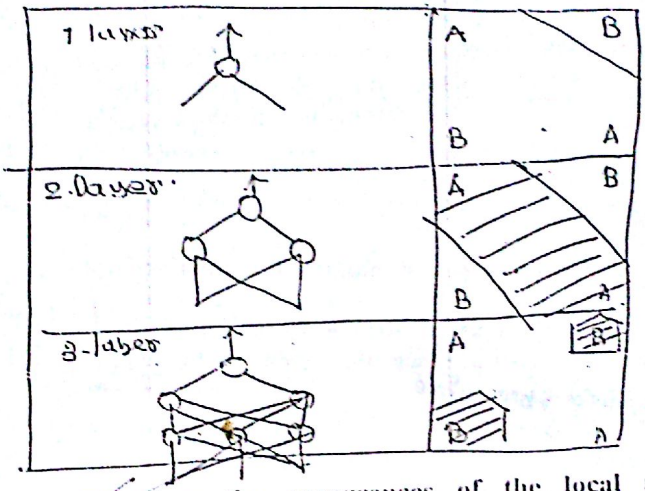
Convex hull: A convex hull is a region in which any point can be connected to any other by a straight line that doesn't cross the boundary of the region. Regions can be closed or open.

Closed: A closed region has a boundary all around it.
Open: An open region doesn't have a boundary all around it.



Arbitrary shapes: The combination of the convex regions is called the arbitrary shapes. The convex regions can be intersect, overlap or be separated from each other producing these arbitrary shapes.

Kolmogrov theorem: The arbitrary complexity of shapes that we can create, means that we never need more4 than three layers in a network, a statement that is referred to as the kolmogrov theorem.



24) How the occurrences of the local minima in a Multilayer Backpropogation Neural Network can be avoided?

We can avoid the occurrences of the local minima by the following ways:

Lowering the gaining term: If the rate at which the weights are altered is progressively decreased, then the gradient descent algorithm is able to achieve a better solution. So we can take the value of gain term from 0.5 to 0.7.

Addition of internal nodes: Local minima can be considered to occur when two or more disjoint classes are categorized as the same. So adding more units to this layer will allow a better recoding of the inputs losses the occurrence of these minima.

Momentum term: The addition of the momentum term is fairly successful to reduce the occurrences of the minima.

Addition of noise: If random noise is added, this perturbs the gradient descent algorithm from the line of steepest descent and often this noise is enough to knock the system out local minima.

25) Discuss the applications of the Multilayer Perceptron.

Nettalk: Nettalk is a multilayer perceptron that learns to pronounce English text.

Airline Marketing Tactician: Airline marketing tactician is a two stage procedure. The first stage consists of a multilayer perceptron that produces forecasts of seat demand and the second stage allocates airline resources.
ECG Noise Filtering: An Electro-cardiograh (ECG) shows the heartbeat of a patient.
Financial Application: Networks have been developed that have discovered significant patterns in the movement of the markets. Another area of the financial application is loan scoring.

Pattern Recognition :-

26) What are the learning difficulties in the BPL?

The learning difficulties are:

1. Equal weights:

2. Local minima:

3. Time complexity:

4. Landscape complexity:

5. Computational complexity:

27) In multilayer perceptron how recovery from damage can be achived quickly? (03)

If a node or its weight are lost or damaged the network therefore demonstrates graceful degradation in performance that catastrophically failure.

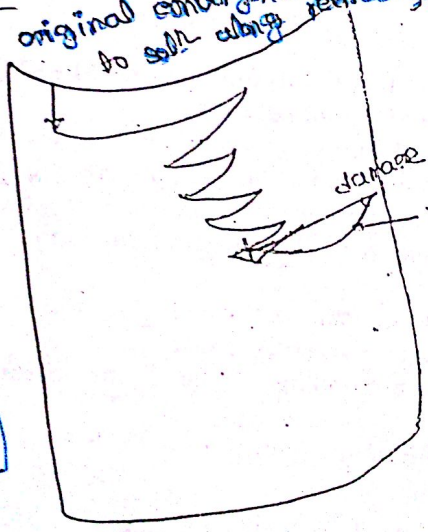


Fig:4.18

Fig: Diagram shows how recovery from damage can be achieved quickly
They are also tolerant to noise due to their intrinsic ability to generalize from taught examples to corrupted versions of the original pattern. Damage to a network is recovered by relearning and in this cases the recovery of the network is often very quick.

28) Now Hamming distance and Euclidean techniques can be used for measuring deviation from pattern of different classes? Or, Discuss different types of distance matrix techniques. 04

Distance Matrix: The matrix which allows us to measure the similarity of pattern samples in the geometric pattern space.

Hamming distance measure: For two vectors, $X=(x_1,x_2,.....)$
$Y=(y_1,y_2,.....)$

The measure is defined by,

$$H=\Sigma(|x_i-y_i|)$$

The hamming distance is often used to compare binary vectors. In actual facts the hamming distance measure for binary data can be performed simply by the exclusive-OR function since $|x_i-y_i|$ is equivalent to x_i XOR y_i

Euclidian distance measure: This can be defined by,

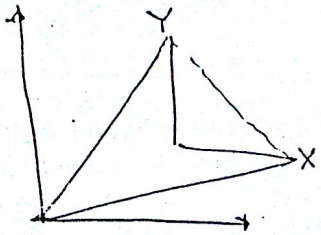
$$d(X,Y)_{euc}=\sqrt{(\sum_{i=1}^{n}(x_i-y_i)^2)}$$

Where n is the dimensionality of the vector.

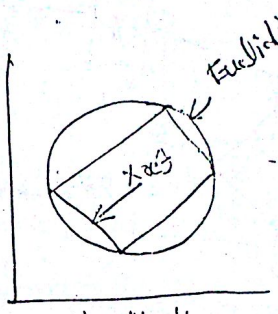
So for two dimensions,

$$d(X,Y)_{euc}=\sqrt{(x_1-y_1)^2+(x_2-y_2)^2}$$

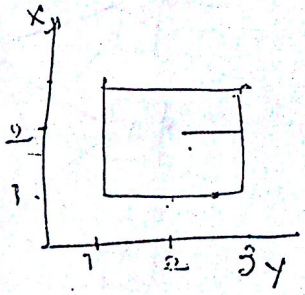
For binary input vectors the metric reduces to a special case which is mathematically equivalent to the square root of the hamming distance. It is simply Pythagoras theorem.



Euclidian



city block



City-Block distance measure: The method is defined by,

$$D_{cd}=\Sigma|x_i-y_i|$$

Without calculating the square or the square root function and the effect of this apart from the obvious one that it is much faster to compute than the Euclidian. The Euclidian circle shown in the Euclidian boundary for equidistance points about the vector. For the city block distance anything falling on the square boundary will yield the same distance value. The error is tolerated as a compromise between accuracy and speed of calculation.

Square distance: With this measure the distance between the vector is defined as the maximum of the differences of the between each element of the two.

$$D_{sq}=MAX\,|x_i-y_i|$$

It is large square than that of the city block and is consequently a coarser measure. The error is tolerated as a compromise between speed and accuracy.

29) Summarize the learning paradigms in a simple neuron. 03

.set the weights and thresholds randomly.

.present an input.

.calculate the actual output by taking the threshold value of the weighted sum of the inputs.

.alter the weights to reinforce correct decisions and discourage incorrect decisions, i.e. reduce the error.
.present the next input.

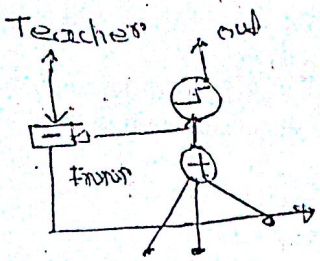
30) Define ADALINE and MADALINE. 06

09 **ADALINE:** Adaline is a single layer neural network with multiple nodes where each node accepts multiple inputs and generates one output.

$Y=\Sigma x_i w_i + 0$

If $x_{n+1}=1$ and $w_{n+1} = 0$

Then $y= x_j w_j$



Madaline: Madaline is a two layer neural network with a set of ADALINE in parallel as its input layer and a single processing unit in its output layer. Another alternative proposed is to use inputs that are not 0 or 1 (binary) but are instead -1 or +1 (bipolar)

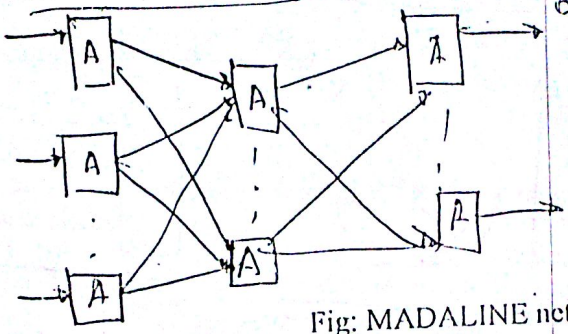


Fig: MADALINE net

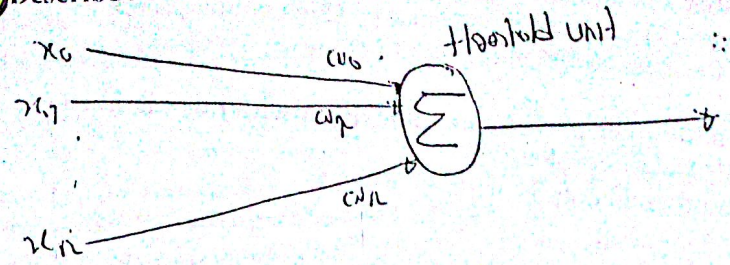
31) With an example of four class problem show that linear classifier can separate them. 05,

09 Linear classifier can be used to separate more than two classes by arranging many decision boundaries and performing several tests to satisfy the conditions for each class. In a four class problem (A, B, C, D), the decision boundaries can be selected to test between A or BCD, if the result is not A then test for B or CD, If not B then test for C or D. Classes that cannot be separated by a single linear decision boundary are termed as non-linearly separable. In nonlinearly separable problem it is also possible to introduce the required non-linearity into the decision surface by applying a non-linear transformation.

classification	Sign of decision line d1	d2
Class 1	+	+
Class 2	+	-
Class 3	-	+
Class 4	-	-

Fig: Piecewise linear classification for a non-linearly separable problem

(32) **Describe Mccullah and Pitt's model of Artificial Neural Network.** 02

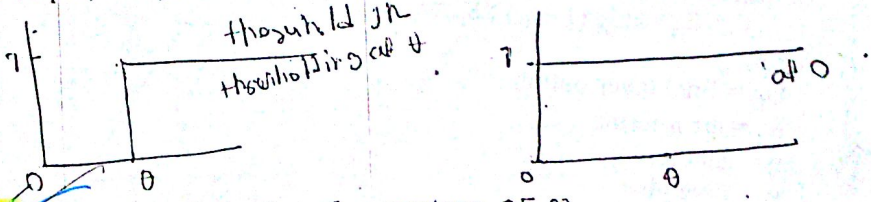


Total input= $w_0x_0 + w_1x_1 + w_2x_2 + \quad + w_nx_n$

$= \Sigma w_ix_i$

This sum then has to be compared to a certain value in the neuron, the threshold value. This thresholding process is accomplished by comparison. If the sum is greater than the threshold value, then output an 1, if less, output a 0.

Feed Forward: The inputs are passed through a model of neuron to produce the output, the system is known as a feed forward.

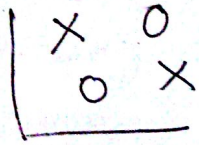


(33) **Mention the limitation of perceptron.** 05, 02

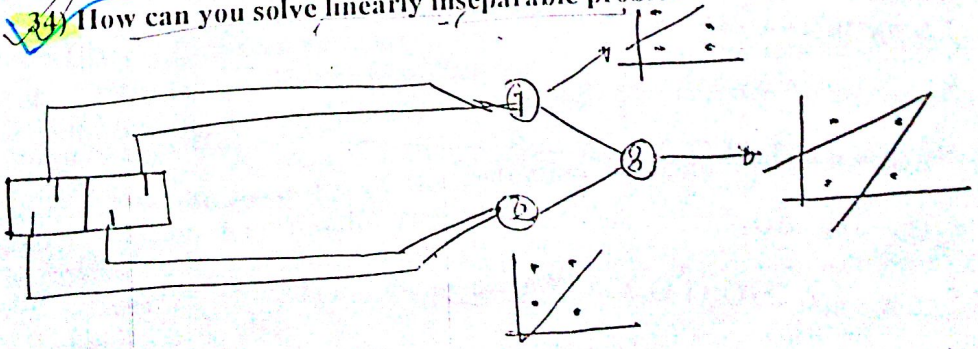
There are limitations to the capabilities of perceptron is that linearly inseparable since no straight line can divide them up successfully. In fact a single layer perceptron cannot solve any problem that is linearly inseparable. Suppose the XOR problem which is linearly inseparable. So it is not solved by single layer perceptron.

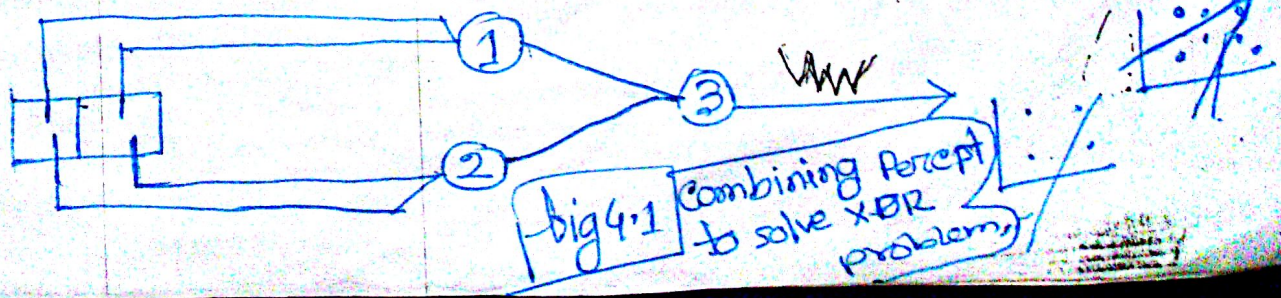
XOR table

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	0



(34) **How can you solve linearly inseparable problem?** 05, 02





To solve linearly inseparable problem we use more than one perceptrons, each set up to identify small, linearly separable sections of the inputs, then combining their outputs into another perceptron, which would produce a final indication of the class to which the input belongs.

35) Describe the multilayer perceptron algorithm with documentations. 05

1. Initialize weights and thresholds.
2. Present input and desired output
   Input: $Xp= X_0, X_1, X_2, \ldots\ldots, X_{n-1}$
   Output: $Tp= t_0, t_1, t_2, \ldots\ldots, t_{n-1}$
3. Calculate the actual output

$$Y_{pj}=f_h\left[\sum_{i=0}^{N-1} w_i x_i\right]$$

4. Adapt weights
   Start from the output layer and work backward

$$w_{ij}(t+1)= w_{ij}(t) + \eta\delta_{pj}O_{pj}$$

   For output units

$$\delta_{pj}= kO_{pj}(1-O_{pj})(t_{pj}-O_{pj})$$

   For hidden units

$$\delta_{pj}= kO_{pj}(1-O_{pj})\Sigma\delta_{pk}W_{pk}$$

where,
- O_{pj} = final layer output
- δ_{pj} = error terms
- η = gain term
- w_{pj} = weights

Documentation of the backward propogation algorithm are:

$net_{aj}=\Sigma w_{ij}\, O_{ai}$ ..... (1)

$active_j = net_{aj} + Uh_j$ ..... (2)

$O_{aj}=(1/(1+e^{-K1 * activej}))$ ..... (3)

$net_{ak}=\Sigma w_{jk} O_{aj}$ ..... (4)

$active_k = net_{ak} + UO_k$ ..... (5)

$O_{ak}=(1/(1+e^{-K2 * activek}))$ ..... (6)

$\delta_{ak} = t_{ak}-O_{ak}$ ..... (7)

$\Delta w_{jk}=\eta_2 k_2 \delta_{ak} O_{aj} O_{ak}(1-O_{ak})$ ..... (8)

$w_{jk}= w_{jk} + \Delta w_{jk}$ ..... (9)

$\Delta U_{ok}= \eta_2 k_2 \delta_{ak}(1-O_{ak})$ ..... (10)

$U_{ok}= U_{ok} + \Delta U_{ok}$ ..... (11)

$\Delta w_{ij}=\eta_1 k_1 O_{ai} O_{aj}(1-O_{aj})\Sigma\delta_{ak}W_{jk}$ ..... (12)

$W_{ij}= w_{ij} + \Delta w_{ij}$ ..... (13)

Omar Farug
Al Mehedi Hasan
Md. Omar Faruk

Fuzzy Logic

$\Delta U_{hj} = \eta_1 k_1 (1 - O_{aj}) \Sigma \delta_{ak} W_{jk}$ .....(14)

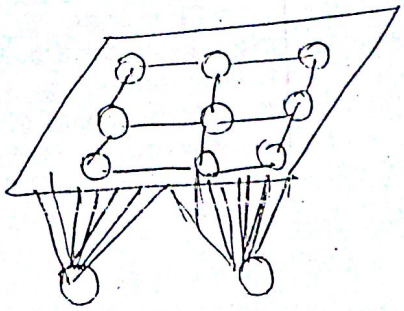
$U_{hj} = U_{hj} + \Delta U_{hj}$ .....(15)

36) Describe the architecture of the Kohonen's self-organizing Network. Why this network model is called so? 08

There are two basic assumptions of the KSON on which the network works upon. They are:

1. The first is that class membership is broadly defined as input patterns that share common features.
2. The other is that the network will be able to identify common features across the range of input patterns.

So the KSON map is one such network that works upon these assumptions and uses unsupervised learning to modify the internal state of the network to model the features found in the training data. KSON is called self-organizing because it uses the unsupervised learning.



37) What is vector quantization? Write down the operational steps of vector quantization. Also discuss the necessity of it? 06, 04, 09

vector quantization: To perform data compression on the vectors to be stored in the network using a technique known as vector quantization.

operational steps of vector quantization:

1. All inputs connect to every node in the network.
2. Feedback is restricted to lateral interconnections to immediate neighbouring nodes.
3. Each of the nodes in the grid is itself an output node.

necessity of it:

1. To represent multidimensional data in a much lower dimensional space. (Data compression)
2. To store data in such a way that spatial or topological relationship in the training data are maintained and represented in meaningful way

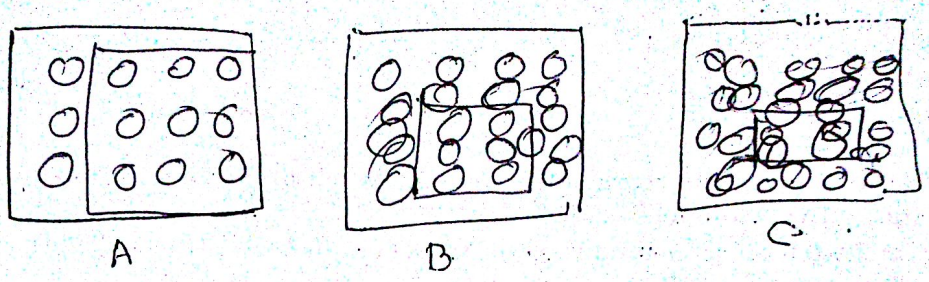
38) What is the necessity of the reduction of neighbourhood size in KSON? Discuss the reduction procedure. 04

Necessity:

1. The effect of shrinking the neighbourhood is to localize the areas of similar activities.
2. This ensures that clusters from accurate internal representations of the training data as well as causing the network to coverage to a solution within a predefined time limit.

Procedure: In A, the network is shown in its initialize state with random weight vectors. In B we can see that the network after many passes through

the training set. The fully trained network is shown in C. As time passes the neighbourhood is reduced and only the nodes in the neighbourhood of the



Winning nodes are altered. These assign themselves more and more, until the area around the winner consists of similar weight vectors. Training is effected by adaption rate and shape of the neighbourhood boundary.

39) **Define fuzzy logic. What are the differences between fuzzy logic and crisp logic?** 01, 03

Fuzzy logic: The logic which is multi-valued and that can be extended to allow for fuzzy premises from which graded conclusions may be drawn.

Differences between fuzzy logic and crisp logic

Fuzzy logic	Crisp logic
1. Fuzzy logic are multivalued logic.	1. Crisp logic is Boolean or classical valued.
2. The values of fuzzy logic are between 0 to 1.	2. The values of crisp logic can be only 0 or 1.
3. It supports a flexible sense of membership of elements to a set.	3. It doesn't support this.

40) **What is fuzzy set? With suitable example explain the following operations on fuzzy sets.(Also described another operations)** 05, 01, 02

1) Equality of two fuzzy sets
2) Power of fuzzy sets
3) Difference of two fuzzy sets
4) Disjunctive sum of two fuzzy sets

Fuzzy Set: Fuzzy sets are those sets whose elements have the degree of membership. A fuzzy set is pair (A,m) where $A \to [0,1]$. For each $x \in A$, $m(x)$ is the grade of membership of x.

1) **Equality:** Two fuzzy sets A and B are said to be equal $(A=B)$ if $\mu_A=\mu_B$.
Example: $A=\{(x_1,0.4),(x_2,0.2)\}$ and $B=\{(x_1,0.4),(x_2,0.2)\}$ then A=B.

2) **Power of fuzzy sets:** the α power of a fuzzy set A is a new fuzzy set A^α whose membership function is given by $\mu_{A^\alpha}(x)=(\mu_A(x))^\alpha$
Example: $A=\{(x_1,0.4),(x_2,0.2)\}$ and $\alpha=2$ $\mu_{A^2}(x)=(\mu_A(x))^2$
Hence $(A)^2=\{(x_1,0.16),(x_2,0.04)\}$

3) **Difference:** The difference of fuzzy sets A and B is a new fuzzy set A-B defined as $A-B=(A \cap B^c)$
Example: $A=\{(x_1,0.4),(x_2,0.2)\}$ and $B=\{(x_1,0.3),(x_2,0.5)\}$ and $B^c=\{(x_1,0.7),(x_2,0.5)\}$ hence $A-B=A \cap B^c=\{(x_1,0.4),(x_2,0.2)\}$

4) **Disjunctive sum**: The disjunctive sum of two fuzzy sets A and B is a new fuzzy set $A \oplus B$ as $A \oplus B=(A^c \cap B) \cup (A \cap B^c)$
Example: $A=\{(x_1,0.4),(x_2,0.8),(x_3,0.6)\}$ $B=\{(x_1,0.2),(x_2,0.6),(x_3,0.9)\}$
$A^c=\{(x_1,0.6),(x_2,0.2),(x_3,0.4)\}$ $B^c=\{(x_1,0.8),(x_2,0.4),(x_3,0.1)\}$

$(A^c \cap B) = \{(x_1,0.2),(x_2,0.2),(x_3,0.4)\}$,
$(A \cap B^c) = \{(x_1,0.4),(x_2,0.4),(x_3,0.1)\}$
$A \oplus B = \{(x_1,0.4),(x_2,0.4),(x_3,0.4)\}$

5) **Union:** The union of two fuzzy sets A and B is a new fuzzy set AUB with membership function defined as, $\mu_{A\cup B}=\max(\mu_A(x),\mu_B(x))$

6) **Intersection:** The intersection of two fuzzy sets A and B is a new fuzzy set $A\cap B$ with membership function defined as, $\mu_{A\cap B}=\min(\mu_A(x),\mu_B(x))$

7) **Complement:** The complement of a fuzzy set A is a new fuzzy set with membership function defined as, $\mu_A{}^c(x)=1-\mu_A(x)$

8) **Product of two fuzzy set:** The product of two fuzzy sets A and B is a new fuzzy set A.B with membership function defined as, $\mu_{A.B}=\mu_A(x).\mu_B(x)$

9) **Product of two fuzzy set with a crisp member:** The product of two fuzzy sets A by a crisp member a is a new fuzzy set a.A with membership function defined as, $\mu_{a.A}=a.\mu_A(x)$

41) **What is membership? Explain why the membership function is necessary for working with Fuzzy Logic?** 09

Membership: An element is said to be a membership of a set A if X belongs to the set A. A fuzzy set accommodates membership values which are not only 0 or 1 but also anything between 0 and 1.

Explain: In fuzzy set theory many degrees of membership (between 0 and 1) are allowed. Thus a membership function $\mu_A(x)$ is associated with a fuzzy set A such that the function maps every element of the universe of discourse X to the interval [0,1]

$\mu_A(x) = X \longrightarrow [0,1]$

If X is a universe of discourse and x is a particular element of X then a fuzzy set A defined on X may be written as a collection of ordered pairs.

$A = \{(x, \mu_A(x)): x \in X\}$ $(x, \mu_A(x))$= singletor

If crisp sets , $\mu_A(x)$ is droped then,

$A=\{x: x \in X\}$

So, membership function is necessary for working with fuzzy logic.

42) **Suppose that A and B are two different fuzzy sets. Prove that** 09

$$(AUB)^c = A^c \cap B^c$$

$(AUB)^c \rightarrow (\max(\mu_A(x),\mu_B(x)))^c$
$\rightarrow \min(\mu_A(x),\mu_B(x)) \rightarrow \min(\mu_{A^c}(x), \mu_{B^c}(x))$
$\rightarrow A^c \cap B^c$

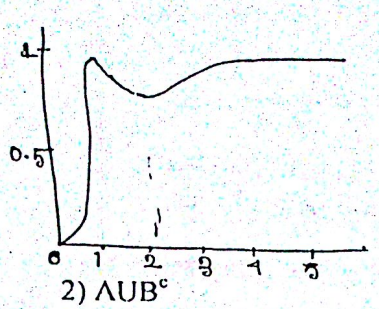
again, $A^c \cap B^c \rightarrow \min(1-\mu_A(x),1-\mu_B(x))$
$\rightarrow (\max(\mu_A(x),\mu_B(x)))^c$
$\rightarrow (AUB)^c$

Hence, $(AUB)^c = A^c \cap B^c$ (proved)

43) **Consider the fuzzy set A and B defined on the internal X= [0,10] of real numbers, by the member grade functions,** 08
$\mu_A(x) = (1/(x^2+1))$, $\mu_B(x) = x/2\pi$. **Determine the mathematical formulae and graphs of the membership grade functions of each of the following sets:** 06, 04

1. $A^c \cap B^c$ 2. $(AUB)^c$ 3. $A \cap B^c$

$A^c \rightarrow \mu_A{}^c(x) = 1 - (1/(x^2+1)) = (x^2+1-1)/(x^2+1) = (x^2/(x^2+1))$
$B^c \rightarrow \mu_B{}^c(x) = 1 - x/2\pi = (2\pi - x)/2\pi$
$\mu_A{}^c \cap \mu_B{}^c(x) = \min(\mu_A{}^c(x), \mu_B{}^c(x)) = \min((x^2/(x^2+1)), (2\pi-x)/2\pi)$



2) $A \cup B^c$

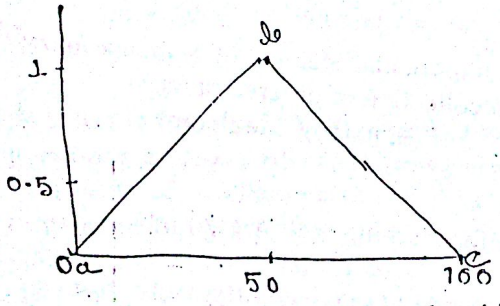
$\mu_{A \cup B^c}(x) = \max(\mu_A(x), \mu_{B^c}(x)) = \max((1/(x^2+1)), x/2\pi)$

3) $(A \cup B)^c = A^c \cap B^c$

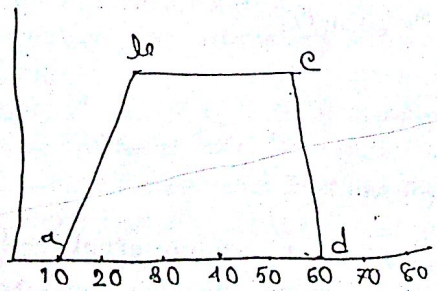
44) Define the followings with necessary figures.

1. Triangular membership function. 2. Trapezoidal 3. S membership function 4. Sigmoidal membership 5. Π_1 membership function 05, 02

Triangular: A triangular membership function is specified by three parameters {a,b,c} as follows: $Triangle(x:a,b,c) = \begin{cases} 0 & x<a \\ (x-a)/(b-a) & ; a \le x \le b \\ (c-x)/(c-b) & ; b \le x \le c \\ 0 & ; x>c \end{cases}$

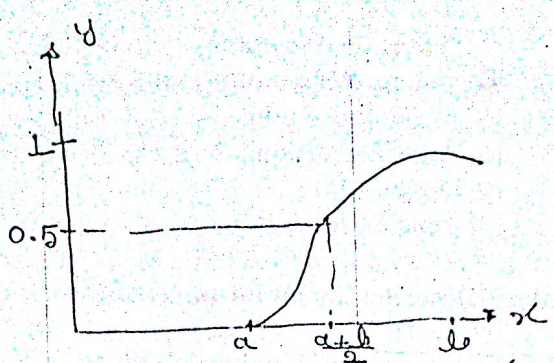


Trapezoidal: A trapezoidal membership function is specified by four parameters {a,b,c,d} as follows: $trapezoid(x:a,b,c,d) = \begin{cases} 0 & ; x<a \\ (x-a)/(b-a) & ; a \le x \le b \\ 1 & ; b \le x \le c \\ (d-x)/(d-c) & ; c \le x \le d \\ 0 & ; x>d \end{cases}$

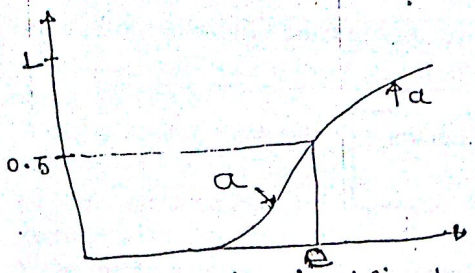


S membership function: The s membership function is a smooth membership function with two parameters: a and b

$$S(x:a,b) = \begin{cases} 0 & ; x<a \\ 2\left(\frac{x-a}{b-a}\right)^2 & ; a \le x \le \frac{a+b}{2} \\ 1-2\left(\frac{x-a}{b-a}\right)^2 & ; \frac{a+b}{2} \le x < b \\ 1 & ; x \ge b \end{cases}$$

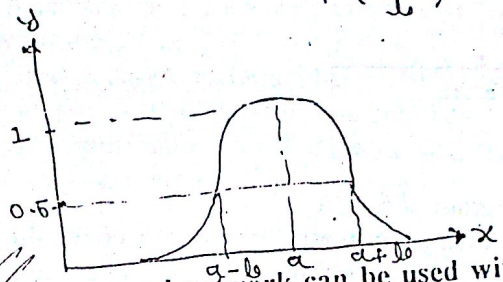


Sigmoidal membership: Defined as $Sg(x; a, c) = \frac{1}{1 + e^{-a(x-c)}}$



Π membership function: is defined with two parameters: a and b.

$$\pi_1(x; a, b) = \frac{1}{1 + \left(\frac{x-a}{b}\right)^2}$$



45) How neural network can be used with Fuzzy based system? Write down the applications of the Fuzzy Logic. 05, 02

The simplest way is to use a fuzzifier function to preprocessors or post process data for a neural network. The preprocessing fuzzifier converts data into fuzzy data for application to a neural network.

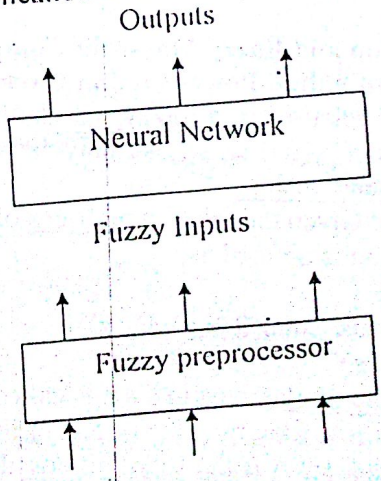


Fig: Neural network with fuzzy preprocessor

Applications of Fuzzy Logic:

1. Artificial intelligence
2. Engineering

3. Computer Science
4. Operational research
5. Robotics
6. Pattern Recognition
7. Filtering
8. Image Processing
9. Intelligent Control
10. Machine Vision
11. Motor Analysis
12. Optimization
13. Prediction
14. Time Series Analysis

46) Define Fit Vector and Fit Value? Describe the major parts of the closed loop FCS. 05,02

Fit Vector: if a, b, c, d are such that their degrees of membership in the fuzzy set A are 0.9, 0.4, 0.5 and 0, the fuzzy set A is given by the fit vector (0.9, 0.4, 0.5 ,0).

Fit Value: The components of this fit vector are called fit values a, b, c, d.

Closed loop FCS:

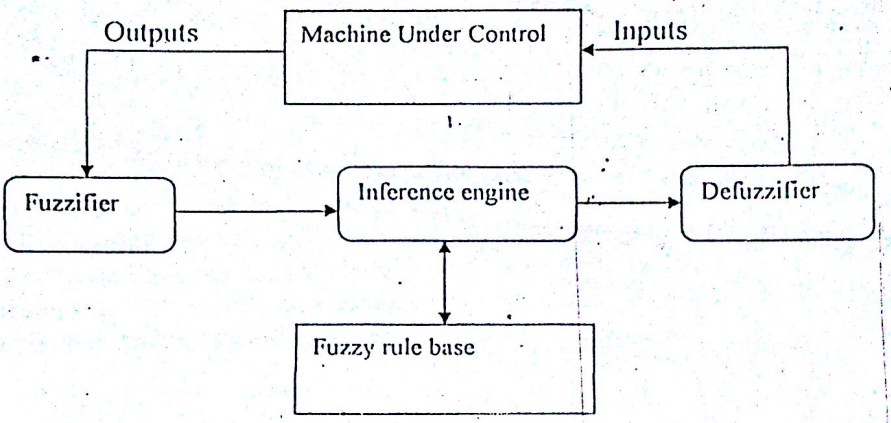


Fig: Diagram of a FCS

From Sheet

Machine under control is the machine or process that we are controlling. Outputs are the measured response. Fuzzy outputs are the same output passed through the fuzzifier. Inference engine is an inference engine converts outputs to actions to take by accessing fuzzy rule base. Fuzzy inputs are fuzzy actions to perform. Inputs are the crisp data converted from fuzzy input with a defuzzifier.

47) Define Fuzzy relation and Fuzzy Max-Min Composition. What do you understand by Cardinality, Power set and Cartesian product? 04,03

Fuzzy relation: Fuzzy relation is a fuzzy set defined on the Cartesian product of crisp set $x_1, x_2, x_3, \ldots\ldots, x_n$ may have varying degrees of membership within the relationship.

Max-Min Composition: Given the relation matrices of the relation R and S , the max-min composition is defined as,

$$T = R \circ S$$

$$T(x,z) = \max_{y \varepsilon Y} \left(\min(R(x,y), S(y,z)) \right)$$

Cardinality: The numbers of elements in a set is called its cardinality. It is denoted by n(A) or |A| or #A. Example: A={4,5,6,7} the |A| =4

Power Set: A power set of set A is the set of all possible subsets that are derived from A including null set. It is denoted by P(A) and its cardinality of $|P(A)| = 2^{|A|}$

Example: A={3,4,5} then P(A)= { {3},{4},{5},{3,4},{3,5},{4,5},{3,4,5},φ}

Here |A|=3 and $|P(A)| = 2^3 = 8$

Cartesian Product: Cartesian Product between the fuzzy sets A and B indicated as AXB resulting in the relation R , R = A X B and $\mu_R(x,y) = \mu_{A\times B}(x,y)=\min(\mu_A(x) ,\mu_B(y))$

48)Make a relative comparison among Expert Systems, Neural Networks and Fuzzy Systems? What are the advantages and disadvantages of the FLC? 02

→ Expert Systems are based on crisp values. But the neural networks are based on the fuzzy values.

→ Expert system have to consider an exhaustive set of possibilities. But NN need not to consider an exhaustive set of possibilities.

→ NN can manage with far less information then ES based.

→ NN provide more flexibility in its output. Es doesn't provide.

Advantages of FLC:

1. Relates input to output in linguistic terms, easily understood.
2. Allows for rapid prototyping.
3. Cheaper because they are easier to design.
4. Increased robustness.
5. Simplify knowledge acquisition and representation.
6. A few rules contain great complexity.
7. It can achieve less overshoot and oscillation.
8. It can achieve steady state in a shorter time period.

Disadvantages:

1. Hard to develop a model from a fuzzy system.
2. Require more fine tuning and simulation before operational.
3. Engineers and most other people are used to crispness and shy away from fuzziness.

49) What is Fuzzy proposition? Write down the application of the FLC (Fuzzy Logic Controller)? 09

Fuzzy Proposition: A fuzzy proposition is a statement which acquires a fuzzy truth value. Thus, given p to be a fuzzy proposition , T(P) represents the truth value , (0,-1) attached to p.i.e, $T(P) = \mu_A(x)$ where $0 \le \mu_A(x) \le 1$

Example: P: Ram is honest

T(P)= 0.8 , if P is partially true

T(P)= 1 , if P is absolutely true.

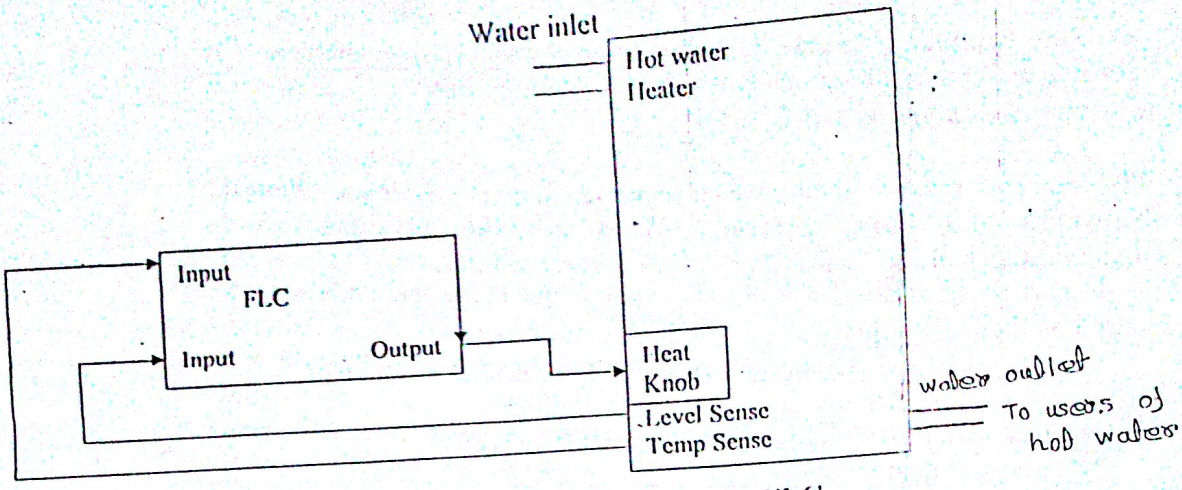
Application of FLC:

1. Video Camcorder
2. Washing Machine
3. Television
4. Motor Control
5. Subway Train
6. Vacuum Cleaner
7. Hot water heater
8. Helicopter Control

Answer from sheet

50) Suppose you have a water heater and you want to control this water heater automatically. Design a water heater in order to maintain a constant temperature 185 degree Celsius. The input to the FLC will be the level of the water and the temperature. The output will be the Crisp value for the setting the heat knob to a desired position and the Motor is on or off.

There are five stages in the designing of a FLC. The figure is,



Step One: Defining inputs and outputs for the FLC

We need to define the universe of discourse all of the inputs and outputs of the FLC, which all crisp values.

Name	Input/Output	Minimum Value	Maximum Value
Level Sense	I	0	10
Temp Sense	I	0	10
Heat Knob	O	0	125
Motor Knob	O	0	1

Step Two: Fuzzify the inputs

There are general guidelines to determine the range of fuzzy variables as related to the fuzzy inputs.

1. Symmetrically distribute the fuzzyfied values across the universe of discourse.
2. Use an odd number of fuzzy sets for each variable so that some set is assured to be in the middle. The set of 5 to 7 sets is fairly typical.
3. Overlap adjacent sets (by 15% to 25% typically)

Fuzzy variable range for level sense

Crisp Input Range	Fuzzy Variable
0-2	Xsmall
1.5-4	Small
3-7	Medium
6-8.5	Large
7.5-10	Xlarge

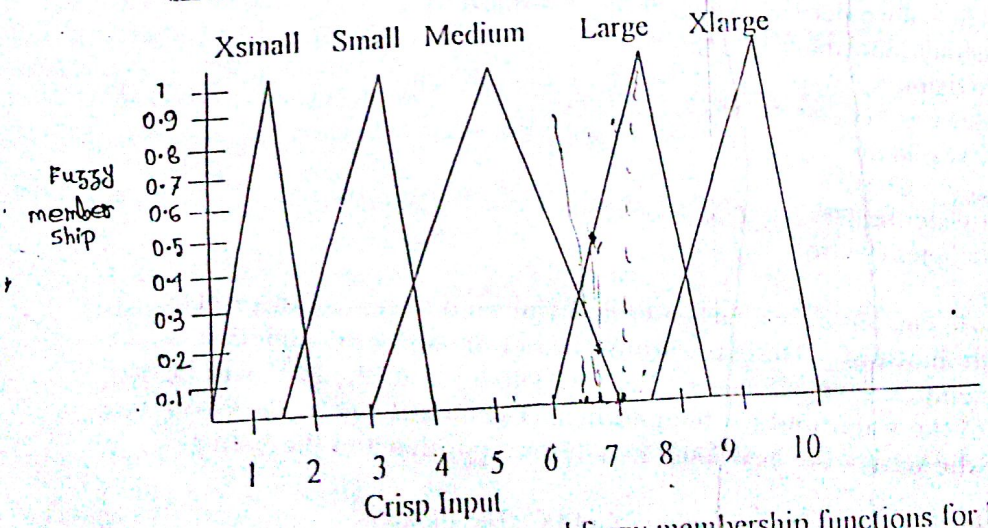


Fig: Fuzzy variable ranges and fuzzy membership functions for levels

Fuzzy variable range for temp sense

Crisp Input Range	Fuzzy Variable
0-20	Xsmall
10-35	Small
30-75	Medium
60-95	Large
85-125	Xlarge

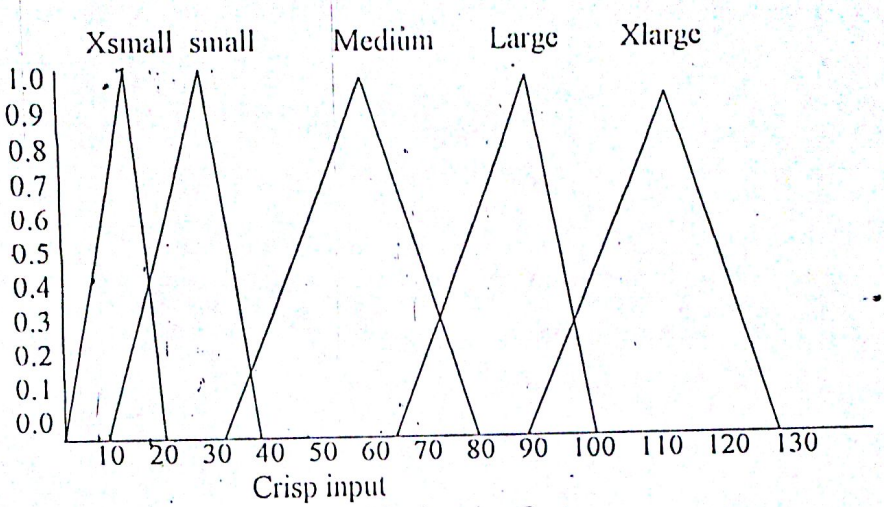


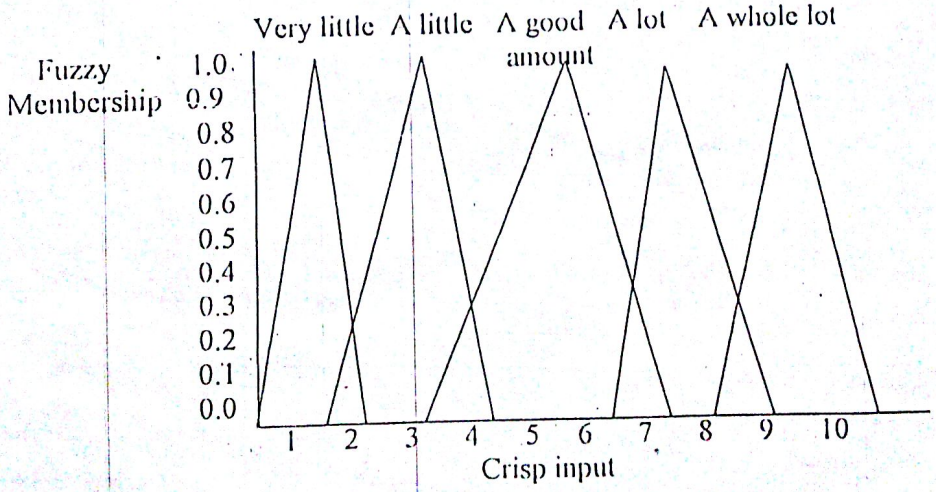
Fig: Fuzzy membership function for temp sense

Step 3: Set up fuzzy membership function for outputs

Here we have two outputs Heat knob and the motor knob. We need to assign fuzzy memberships to this variable just as we did for the inputs.

Fuzzy variable range for heat knob

Crisp Input range	Fuzzy variable
0-2	Very little
1.5-4	A little
3-7	A good amount
6-8.5	A lot
7.5-10	A whole lot



Fuzzy variable range for Motor

Crisp input range	Fuzzy variable
0.0-0.5	OFF
0.5-1.0	ON

Step 4: Create a fuzzy rule base

These rules usually take the form of IF-THEN rules and can be obtained from a human expert (heuristics) or can be supplied from a neural network that infers the rules from behavior of the system. Our heuristic guidelines in determining this matrix are the following statements and their converses.

1. When the temperature is low the heat knob should than when the temperature is high.
2. When the volume of water is low, the heat knob does not need to be as high as when the volume of water is high.

Fuzzy rule base for output knob

temp / level	Xs	S	M	L	Xl
Xs	Λ good amount	Λ little	Very little		
S	Λ lot	Λ good amount	Very little	Very little	
M	Λ whole lot	Λ lot	Λ good amount	Very little	
L	Λ whole lot	Λ lot	Λ lot	Λ little	
Xl	Λ whole lot	Λ lot	Λ lot	Λ good amount	

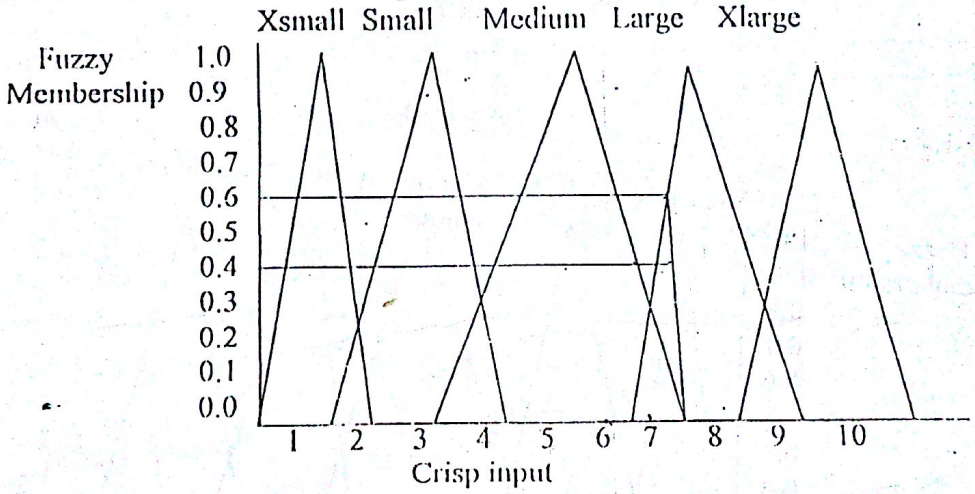
Fuzzy rule base for output motor

If the level sense is less than or medium then the motor is ON. Otherwise the motor is off.

Step 5: Defuzzify the outputs

Here we have to set values for the inputs and then find out the corresponding values in the membership graph. Suppose,

Level sense=7.0 then from the graph



Large: 0.6,
medium: 0.4
and all others: 0.0

Temp sense = 65 then from the graph,

Medium : 0.75
Large: 0.25
All other: 0.0

Now we have four combinations. Level large, level medium, temp large and temp small. So we have to 4 and prations and after that 1 or operations for finding the output