Roll No. [illegible] Total No. of Pages : 02

Total No. of Questions : 09

B.Tech. (CSE/ECE) (Sem.–7)

SOFT COMPUTING

Subject Code : BTEC-9080-18

M.Code : 90681

Date of Examination : 14-01-2023

Time : 3 Hrs. Max. Marks : 60

INSTRUCTIONS TO CANDIDATES :

1. SECTION-A is COMPULSORY consisting of TEN questions carrying TWO marks each.
2. SECTION-B contains FIVE questions carrying FIVE marks each and students have to attempt any FOUR questions.
3. SECTION-C contains THREE questions carrying TEN marks each and students have to attempt any TWO questions.

SECTION-A

1. *Write briefly :*

   a) Why McCulloch-Pitts neuron widely used in logic functions?

   b) What is the difference between soft and hard computing?

   c) What is the importance of sigmoid function?

   d) Differentiate between crisp and fuzzy set theory.

   e) Compare feed forward and feedback networks.

   f) How fuzzy sets are defined in Fuzzy Logic?

   g) Explain the concept of Widrow-Hoff rule.

   h) What is rule based learning?

   i) What do you mean by mutation?

   j) Define any two fuzzy set operations with example.

(S2) JJ

SECTION-B

2. Differentiate between uniform and three parent cross over.
3. Explain with an example any 2 operators involved in simple GA.
4. Explain different defuzzification techniques.
5. Explain how animal behaviour in bees is used for Swarm Intelligence.
6. Explain training algorithm used in adaptive linear neuron.

SECTION-C

7. Explain:

   a) McCulloh Pitts neuron model.

   b) Application of NN for Character Recognition.

8. Explain the characteristics and different classifications of a neuro-fuzzy hybrid system.
9. Implement OR function using perception training algorithm with binary inputs and bipolar targets.

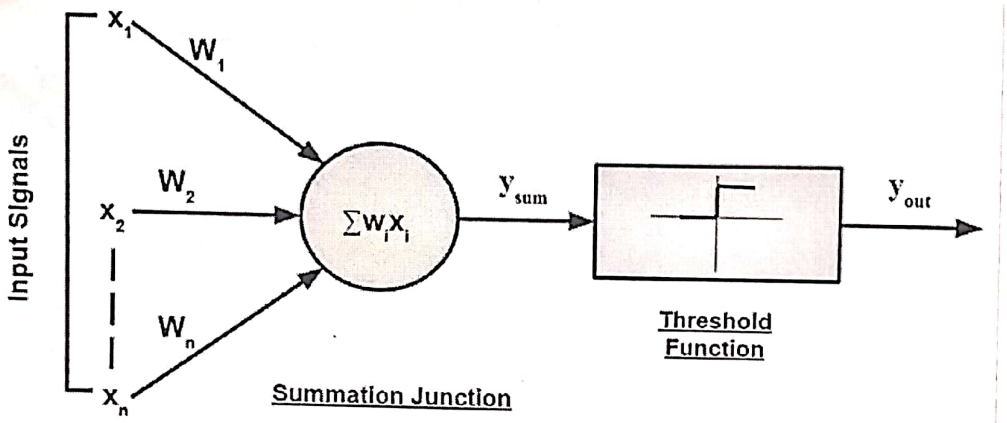
NOTE : Disclosure of Identity by writing Mobile No. or Making of passing request on any page of Answer sheet will lead to UMC against the Student.

(S2) JJ

IKGPTU Dec 2023 Regular Exam Solution
Section C

Q7 a. McCulloch-Pitts Model of Neuron

The McCulloch-Pitts neural model, which was the earliest ANN model, has only two types of inputs — **Excitatory and Inhibitory.** The excitatory inputs have weights of positive magnitude and the inhibitory weights have weights of negative magnitude. The inputs of the McCulloch-Pitts neuron could be either 0 or 1. It has a threshold function as an activation function. So, the output signal y_{out} is 1 if the input y_{sum} is greater than or equal to a given threshold value, else 0. The diagrammatic representation of the model is as follows:



McCulloch-Pitts Model

Simple McCulloch-Pitts neurons can be used to design logical operations. For that purpose, the connection weights need to be correctly decided along with the threshold function (rather than the threshold value of the activation function).

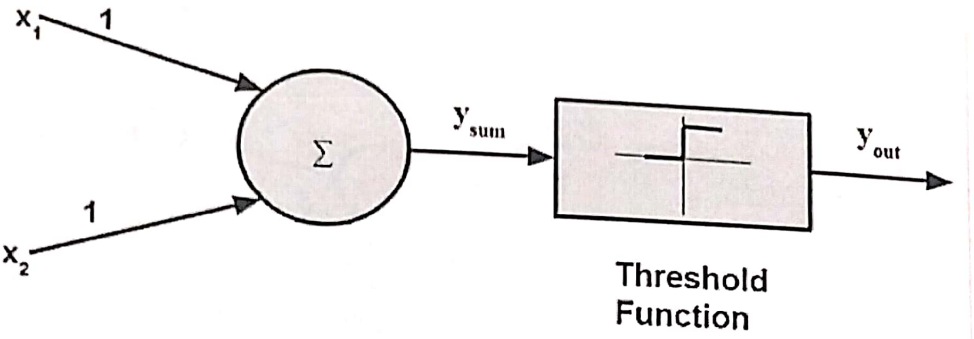
Example: John carries an umbrella if it is sunny or if it is raining. There are four given situations. The situations are as follows:

- First scenario: It is not raining, nor it is sunny
- Second scenario: It is not raining, but it is sunny
- Third scenario: It is raining, and it is not sunny
- Fourth scenario: It is raining as well as it is sunny

To analyse the situations using the McCulloch-Pitts neural model, consider the input signals as follows:

- X_1: Is it raining?
- X_2 : Is it sunny?

So, the value of both scenarios can be either 0 or 1. We can use the value of both weights X_1 and X_2 as 1 and a threshold function as 1. So, the neural network model will look like:



Truth Table for this case will be:

Situation	x_1	x_2	y_{sum}	y_{out}
1	0	0	0	0
2	0	1	1	1
3	1	0	1	1
4	1	1	2	1

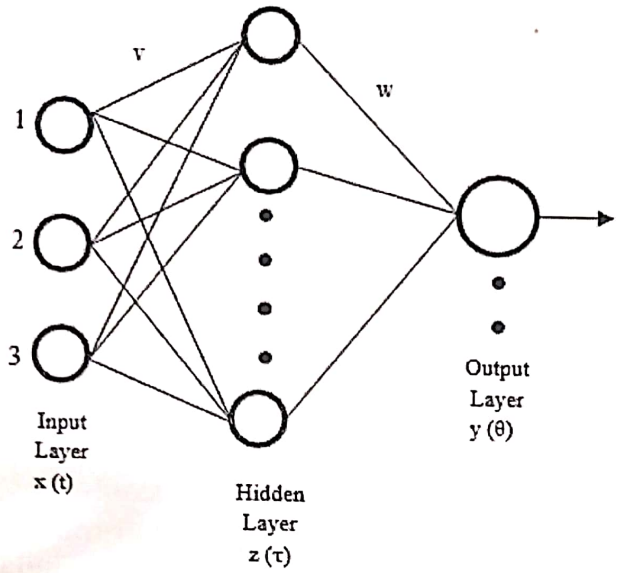
$$y_{sum} = \sum_{i=1}^{2} w_i x_i$$

$$y_{out} = f(y_{sum}) = \begin{cases} 1, x \geq 1 \\ 0, x < 1 \end{cases}$$

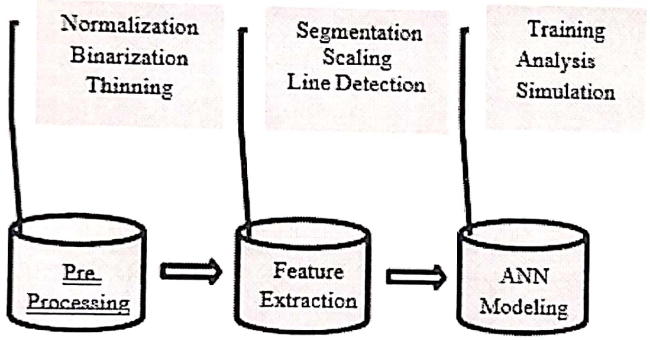
The truth table built with respect to the problem is depicted above. From the truth table, it is concluded that in the situations where the value of y_{out} is 1, John needs to carry an umbrella i.e., in scenarios 2, 3 and 4.

Q7b. Applications of NN for character recognition

Figure 1 shows the basic structure of an ANN model. For each training values, a series of steps are done. These steps can be broken down mainly into Forward Pass and Backward Pass.At the beginning of training,the weights are randomly initialized. The nomenclatures used in the algorithm are given below:



The entire process can be broken down into Pre-processing,Feature Extraction,and then passing into ANN for training and simulation.These steps are visualized in fig2.



Pre-processing

Prior to ANN modeling, it is important that the images are in good quality. This enhances the image and decreases noise and distortion. This helps in achieving higher accurate results. It is essential in any OCR system that a preprocessing stage exists.

Normalization

This helps to decrease the noise in the image by performing a local averaging operation on a 5 X 5 neighborhood. In order to achieve this, a median or mean filter can be used. To achieve minimal blurring, median filter is used.

Binarization

Local binarization is then carried out in the resulting image. Since the color information is irrelevant, this gives us uniformity between the samples. This also reduces computational power as it has to deal with only 2 colors.

Thinning

Thinningreducesthewidthofsimilarpixelsto1pixel.Thinning is done with the help of edge detection by sobel'smethod.Thinning reduces redundancy and makes the characters uniform.

Q8. Classification of Neuro fuzzy hybrid systems

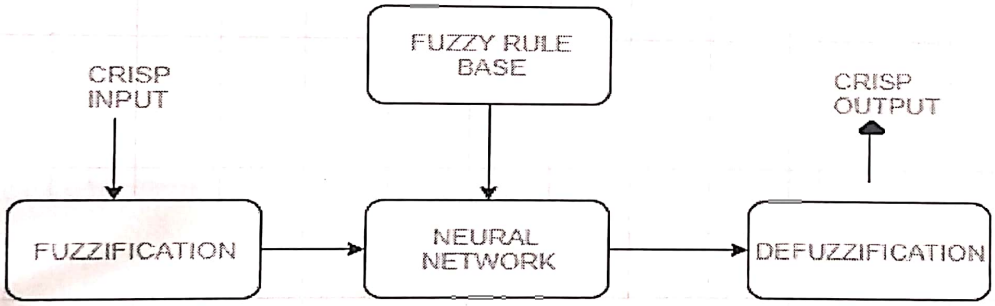
Hybrid systems: A Hybrid system is an intelligent system that is framed by combining at least two intelligent technologies like Fuzzy Logic, Neural networks, Genetic algorithms, reinforcement learning, etc. The combination of different techniques in one computational model makes these systems possess an extended range of capabilities. These systems are capable of reasoning and learning in an uncertain and imprecise environment. These systems can provide human-like expertise like domain knowledge, adaptation in noisy environments, etc.

Types of Hybrid Systems:

- Neuro-Fuzzy Hybrid systems
- Neuro Genetic Hybrid systems
- Fuzzy Genetic Hybrid systems

(A)Neuro-Fuzzy Hybrid systems

The Neuro-fuzzy system is based on fuzzy system which is trained on the basis of the working of neural network theory. The learning process operates only on the local information and causes only local changes in the underlying fuzzy system. A neuro-fuzzy system can be seen as a 3-layer feed forward neural network. The first layer represents input variables, the middle (hidden) layer represents fuzzy rules and the third layer represents output variables. Fuzzy sets are encoded as connection weights within the layers of the network, which provides functionality in processing and training the model.



Working flow:

- In the input layer, each neuron transmits external crisp signals directly to the next layer.
- Each fuzzification neuron receives a crisp input and determines the degree to which the input belongs to the input fuzzy set.
- The fuzzy rule layer receives neurons that represent fuzzy sets.
- An output neuron combines all inputs using fuzzy operation UNION.
- Each defuzzification neuron represents the single output of the neuro-fuzzy system.

Advantages:

- It can handle numeric, linguistic, logic, etc kind of information.
- 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 can mimic the human decision-making process.

Disadvantages:

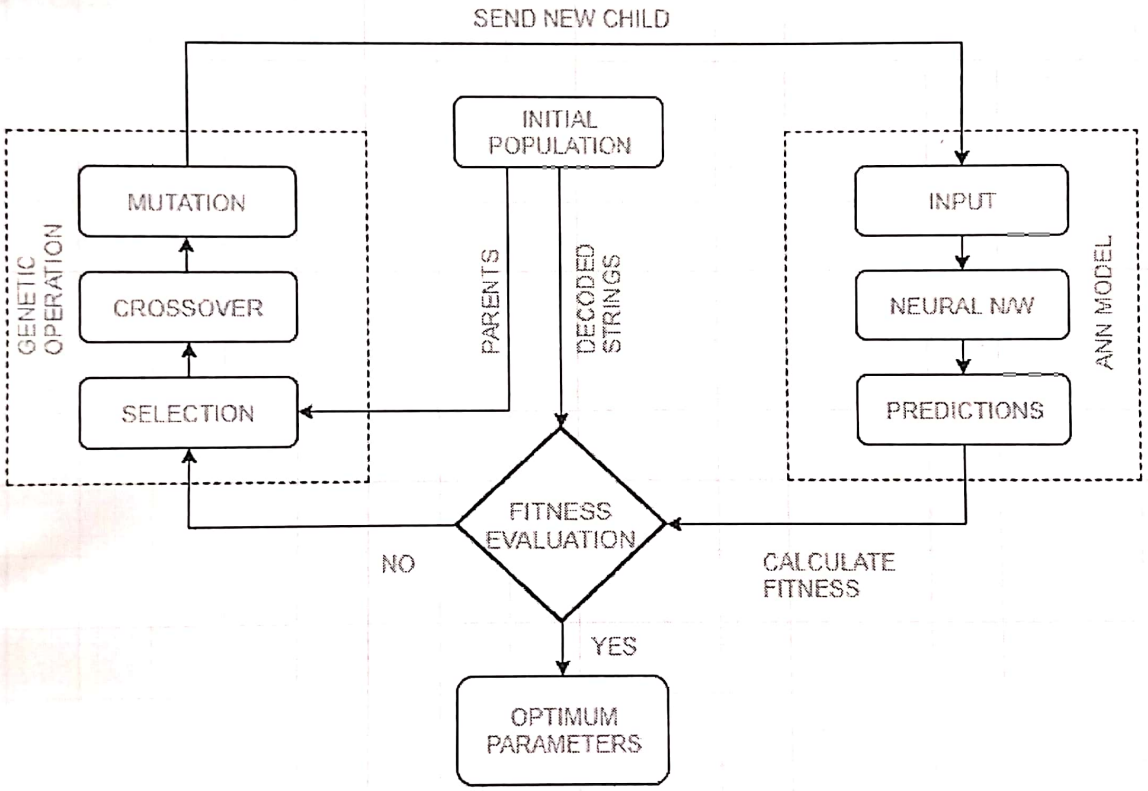
- Hard to develop a model from a fuzzy system
- Problems of finding suitable membership values for fuzzy systems
- Neural networks cannot be used if training data is not available.

Applications:

- Student Modelling
- Medical systems
- Traffic control systems
- Forecasting and predictions

(B) Neuro Genetic Hybrid systems:

A Neuro Genetic hybrid system is a system that combines **Neural networks**: which are capable to learn various tasks from examples, classify objects and establish relations between them, and a **Genetic algorithm**: which serves important search and optimization techniques. Genetic algorithms can be used to improve the performance of Neural Networks and they can be used to decide the connection weights of the inputs. These algorithms can also be used for topology selection and training networks.



Working Flow:

- GA repeatedly modifies a population of individual solutions. GA uses three main types of rules at each step to create the next generation from the current population:
    1. **Selection** to select the individuals, called parents, that contribute to the population at the next generation
    2. **Crossover** to combine two parents to form children for the next generation
    3. **Mutation** to apply random changes to individual parents in order to form children
- GA then sends the new child generation to ANN model as a new input parameter.

- Finally, calculating the fitness by the developed ANN model is performed.

Advantages:

- GA is used for topology optimization i.e to select the number of hidden layers, number of hidden nodes, and interconnection pattern for ANN.
- In GAs, the learning of ANN is formulated as a weight optimization problem, usually using the inverse mean squared error as a fitness measure.
- Control parameters such as learning rate, momentum rate, tolerance level, etc are also optimized using GA.
- It can mimic the human decision-making process.

Disadvantages:

- Highly complex system.
- The accuracy of the system is dependent on the initial population.
- Maintenance costs are very high.

Applications:

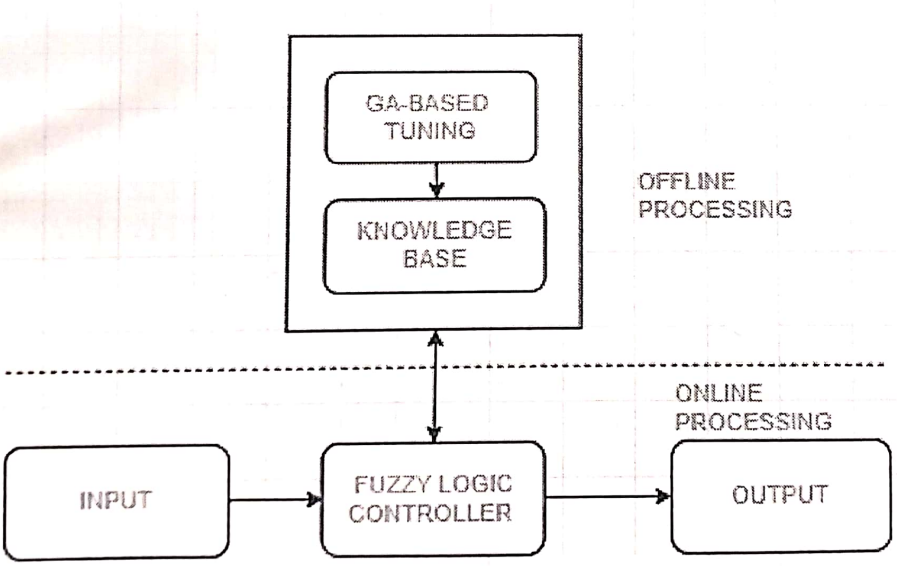
- Face recognition
- DNA matching
- Animal and human research
- Behavioral system

(C) Fuzzy Genetic Hybrid systems:

A Fuzzy Genetic Hybrid System is developed to use fuzzy logic-based techniques for improving and modeling Genetic algorithms and vice-versa. Genetic algorithm has proved to be a robust and efficient tool to perform tasks like generation of the fuzzy rule base, generation of membership function, etc.

Three approaches that can be used to develop such a system are:

- Michigan Approach
- Pittsburgh Approach
- IRL Approach



Working Flow:

- Start with an initial population of solutions that represent the first generation.
- Feed each chromosome from the population into the Fuzzy logic controller and compute performance index.
- Create a new generation using evolution operators till some condition is met.

Advantages:

- GAs are used to develop the best set of rules to be used by a fuzzy inference engine
- GAs are used to optimize the choice of membership functions.
- A Fuzzy GA is a directed random search over all discrete fuzzy subsets.
- It can mimic the human decision-making process.

Disadvantages:

- Interpretation of results is difficult.
- Difficult to build membership values and rules.
- Takes lots of time to converge.

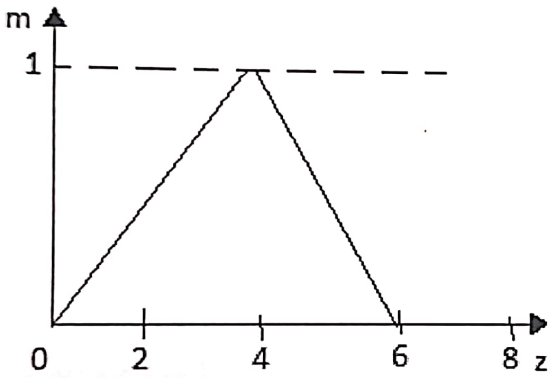
Applications:

- Mechanical Engineering
- Electrical Engine
- Artificial Intelligence
- Economics

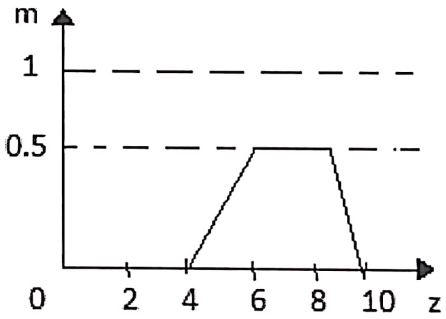
Section B

Q4.Defuzzification:

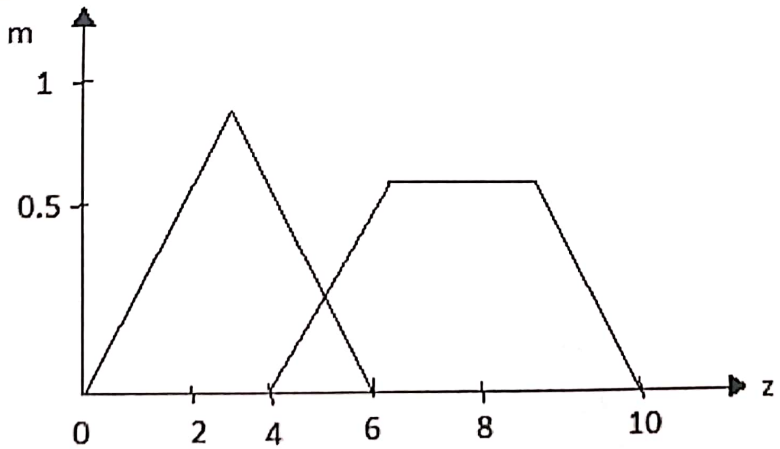
Defuzzification is the process of conversion of fuzzy quantity into a precise quantity.



[A]



[B]



[C]

[A] first part of fuzzy output (C1)

[B] Second part of fuzzy output (C2)

[C] Union of part [A] and [B].

The union of two membership function in values the max operator, which is going to be the outer envelope of the two or more shapes.

Defuzzification methods include:

[1] max membership principle.

[2] centroid method.

[3] weighted average method.

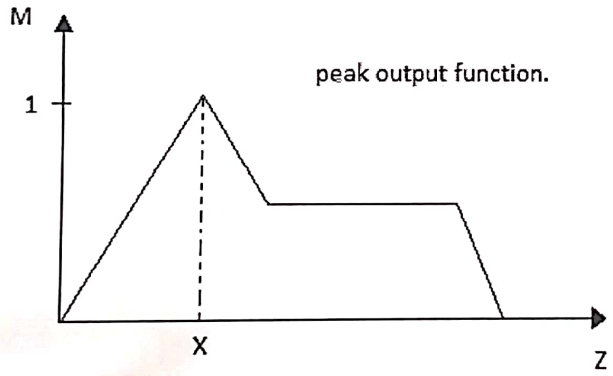
[4] mean max membership.

[5] center of sums.

[6] centre of largest area.

[7] first of maxima, last of maxima.

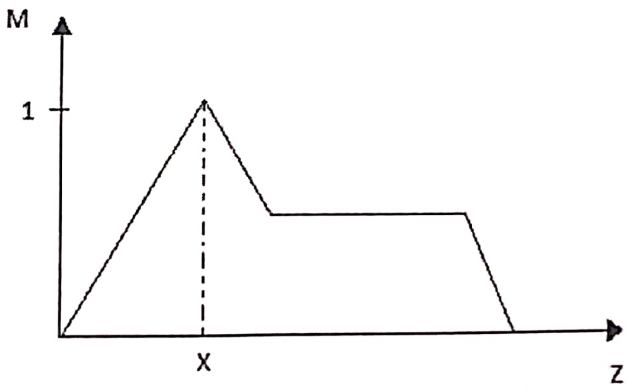
[1] Max – membership principle:



$M\,c\,(x^*) > M\,c\,(x)$ for all x ∈ X

[2] Centroid method: centre of mall, centre of gravity or area.

$$X^A = \frac{\int Ms(x).xdx}{\int Mc(x).dx}$$



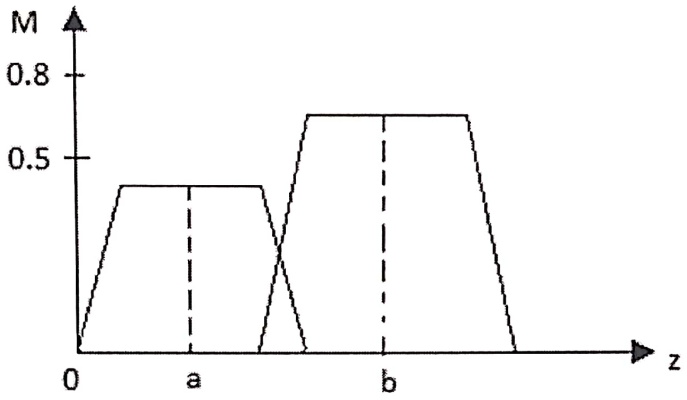
[3] Weighted average method: Valid for symmetrical output membership function.

Each membership function is weighted by its max membership value.

$$X^* = \frac{\sum Mc(\bar{x}i).\bar{x}p}{\sum MC(\bar{x}i)}$$

$\bar{X}i$ = maximum of with member function.

$\sum$ = algebraic sum.

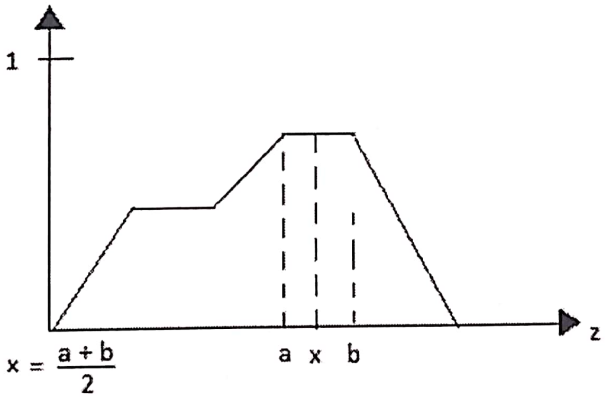


$$x^* = \frac{0.5a+0.8b}{0.5+0.8}$$

[4] Mean max membership method:

This is known as middle of the maxima.

$$X^* = \frac{\sum_{i=1}^{n} \bar{x}p}{n}$$

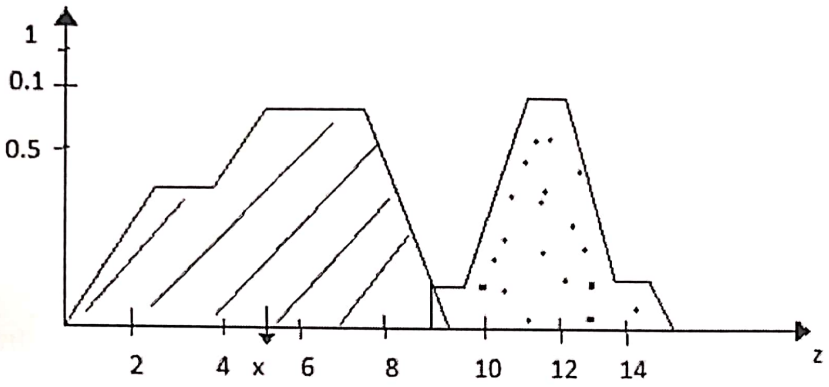


[5] Centre of sums: Algebraic sum of individual fuzzy the union, here, interesting areas are value twice, the defuzzified value X^+

$$X^* = \frac{\int_x X \sum_{i_2}^{n} MCI(x)dx}{\int_x \sum_{i_2}^{n} Mci(x)dx}$$

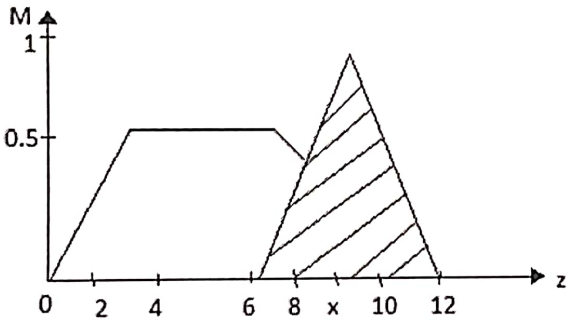
[6] Centre of largest area: When output consists of at least two converse fuzzy subsets which are not overlapping. When o/p fuzzy set has at least two converse regions, then the centre of gravity of converse fuzzy sub region having the largest area is used to obtain defuzzified value.

$$X^* = \frac{\int mci(x).xdx}{\int mci(x)dx}$$



[7] first of maxima (last of maxima)

This method uses the overall output or union of all individual output fuzzy sets ci for determining the smallest value of the domain maximized membership in ci.



Q5. Animal Behaviour of bees in swarm intelligence

Swarm intelligence refers to the intelligent behaviors of ants, birds, and other swarm animals that achieve their goals through the interaction between individuals or with the environment in the process of migration and foraging.

Bee swarms exhibit many intelligent behaviours in their tasks such as nest site building, marriage, foraging, navigation and task selection. There is an efficient task selection mechanism in a bee swarm that can be adaptively changed by the state of the hive and the environment. Foraging is another crucial task for bees. Forage selection depends on recruitment for and abandonment of food sources. There are three types of bees associated with the foraging task with respect to their selection mechanisms. Employed bees fly onto the sources which they are exploiting; onlooker bees choose the sources by watching the dances performed by employed bees, and scouts choose sources randomly by means of some internal motivation or possible external clue. The exchange of information among bees is the most important occurrence in the formation of the collective knowledge. The most important part of the hive in terms of exchanging information is the dancing area. Communication among bees related to the quality of food sources takes place in the dancing area. Various dances are

performed on the dancing area, such as waggle, round, tremble depending on the distance of the discovered source.

Particle swarm optimization (PSO) is accepted as the second population-based algorithm inspired from animals. Since James Kennedy (a social psychologist) and Russell C. Eberhart simulated the bird flocking and fish schooling foraging behaviors, they have used this simulation to the solution of an optimization problem and published their idea in a conference in 1995 [3] for the optimization of continuous nonlinear functions. There are two main concepts in the algorithm: velocity and coordinate for each particle. Each particle has a coordinate and an initial velocity in a solution space. As the algorithm progresses, the particles converge toward the best solution coordinates. Since PSO is quite simple to implement, it requires less memory and has no operator. Due to this simplicity, PSO is also a fast algorithm. Different versions of PSO have been developed, using some operators since the first version of PSO was published.

ABC optimization is developed by Karaboga in 2005. ABC is also a swarm intelligence algorithm based on the foraging behavior of honey bee swarms. The artificial bee colonies in the ABC algorithm consists of three groups: employed bees, onlookers and scouts. Employed bees search for food source and sharing this information to recruit onlooker bees. Onlooker bees select better food sources from those employed bees and further search around the selected food source. If a food source is not improved by some iteration, this employed bee will become a scout bee to search randomly for new food sources. In [45], a hybrid algorithm is developed combining PSO and ABC. Since PSO is fast convergent algorithm and ABC is slow convergent algorithm, hybrid algorithm uses the powerful sides of each algorithm.

Algorithm

Initialize the system parameters

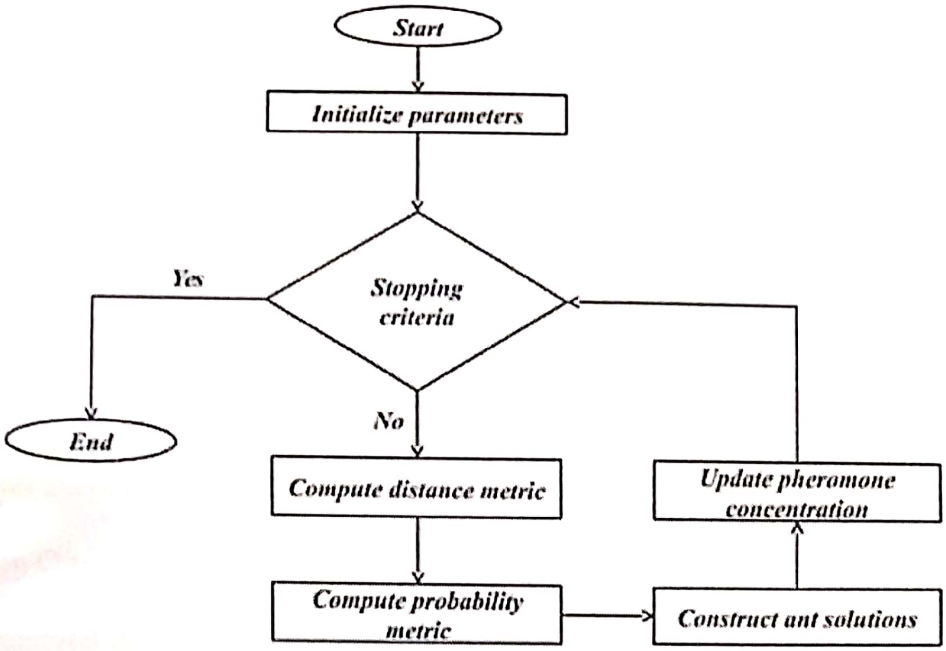
while** termination condition not met **do

Construct Solutions

Apply Path Search

Update Pheromones

End



Q3. Crossover methods

The crossover operator is analogous to reproduction and biological crossover. In this more than one parent is selected and one or more off-springs are produced using the genetic material of the parents. Crossover is usually applied in a GA with a high probability

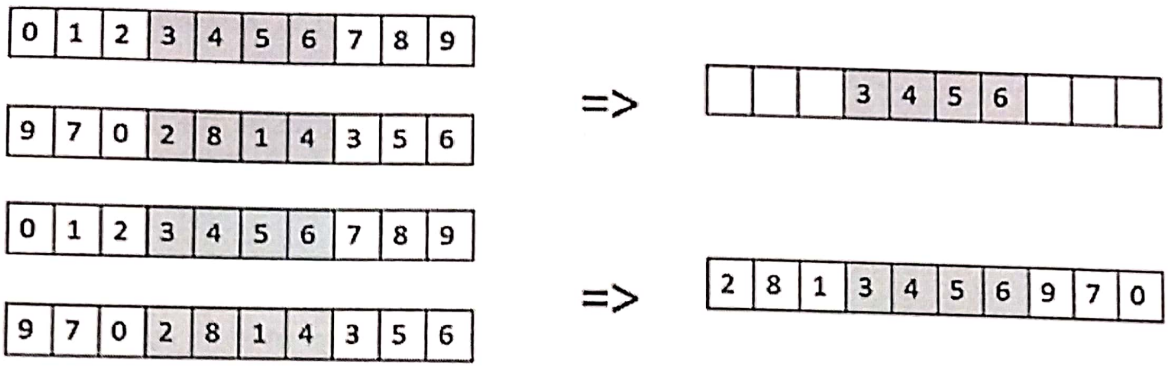
Uniform Crossover

In a uniform crossover, we don't divide the chromosome into segments, rather we treat each gene separately. In this, we essentially flip a coin for each chromosome to decide whether or not it'll be included in the off-spring. We can also bias the coin to one parent, to have more genetic material in the child from that parent.

Parent crossover

This is used for permutation based crossovers with the intention of transmitting information about relative ordering to the off-springs. It works as follows –

- Create two random crossover points in the parent and copy the segment between them from the first parent to the first offspring.
- Now, starting from the second crossover point in the second parent, copy the remaining unused numbers from the second parent to the first child, wrapping around the list.
- Repeat for the second child with the parent's role reversed.

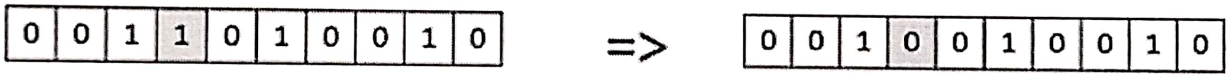


Q2. Operators used in GA

Mutation Operator
Mutation is the part of the GA which is related to the "exploration" of the search space. It has been observed that mutation is essential to the convergence of the GA while crossover is not.

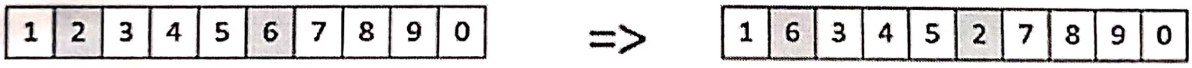
Bit Flip Mutation

In this bit flip mutation, we select one or more random bits and flip them. This is used for binary encoded GAs.



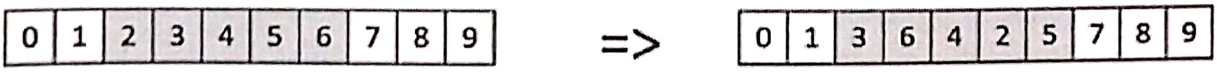
Swap Mutation

In swap mutation, we select two positions on the chromosome at random, and interchange the values. This is common in permutation based encodings.



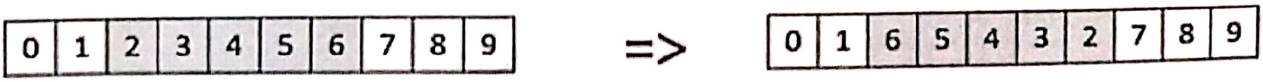
Scramble Mutation

Scramble mutation is also popular with permutation representations. In this, from the entire chromosome, a subset of genes is chosen and their values are scrambled or shuffled randomly.



Inversion Mutation

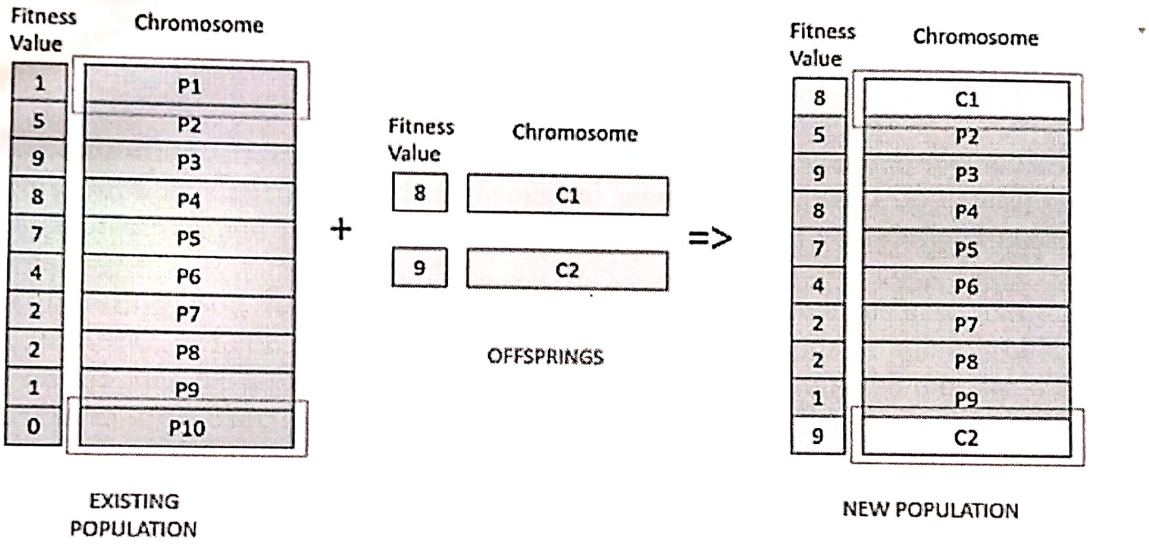
In inversion mutation, we select a subset of genes like in scramble mutation, but instead of shuffling the subset, we merely invert the entire string in the subset.



Fitness based selection in GA

In this fitness based selection, the children tend to replace the least fit individuals in the population. The selection of the least fit individuals may be done using a variation of any of the selection policies described before – tournament selection, fitness proportionate selection, etc.

For example, in the following image, the children replace the least fit individuals P1 and P10 of the population. It is to be noted that since P1 and P9 have the same fitness value, the decision to remove which individual from the population is arbitrary.



Section A

a.The Mc-Culloch-Pitts neuron was the earliest neural network discovered in 1943. It is usually called as M-P neuron. The M-P neurons are connected by directed weighted parts.

The activation function of an M-P neuron is binary, that is, any step the neuron may fire or may not fire. The weights associated with the communication links maybe excitatory w ($w>0$, weight is positive) or inhibitory p ($p<0$, weight is negative). All excitedly connected weights entering into a particular neuron will have the same weights. The threshold plays a major role in M-P neuron.

There is a fixed threshold for each neuron, and if the net input to the neuron is greater than the threshold then the neuron fires. Also, any non zero inhibitory input would prevent the neuron from firing. The M-P neurons are most widely used in the case of logic functions.

$\Rightarrow$ M-P neuron is excitatory with the weight ($w > 0$) and inhibitory with weight $-p$ ($p < 0$)

Input x_1 to x_n posses excitatory weighted connections and input from x_{n+1} to x_{n+m} posses inhibitory weighted connection.

Since the firing of the neuron is based upon the threshold, the activation function here is defined as,

$$f(y_{in}) = \begin{cases} 1, & if \quad y_{in} \geq \theta \\ 0, & if \quad y_{in} < \theta \end{cases}$$

$\Rightarrow$ The M-P neuron has no particular training algorithm.

$\Rightarrow$ An analysis has to be performed to determine the values of the weights and the threshold.

$\Rightarrow$ Here, the weights of the neurone are set along with the threshold to make neuron perform a single logic function.

$\Rightarrow$ M-P neuron can be used as a building block on which we can model any function or phenomena, which can be represented as a logic function.

b.

Hard Computing: Hard computing uses traditional mathematical methods to solve problems, such as algorithms and mathematical models. It is based on deterministic and precise calculations and is ideal for solving problems that have well-defined mathematical solutions.

Soft Computing: Soft computing, on the other hand, uses techniques such as fuzzy logic, neural networks, genetic algorithms, and other heuristic methods to solve problems. It is based on the idea of approximation and is ideal for solving problems that are difficult or impossible to solve exactly.

S.NO	Soft Computing	Hard Computing
1.	Soft Computing is liberal of inexactness, uncertainty, partial truth and approximation.	Hard computing needs a exactly state analytic model.
2.	Soft Computing relies on formal logic and probabilistic reasoning.	Hard computing relies on binary logic and crisp system.
3.	Soft computing has the features of approximation and dispositionality.	Hard computing has the features of exactitude(precision) and categoricity.
4.	Soft computing is stochastic in nature.	Hard computing is deterministic in nature.
5.	Soft computing works on ambiguous and noisy data.	Hard computing works on exact data.

S.NO	Soft Computing	Hard Computing
6.	Soft computing can perform parallel computations.	Hard computing performs sequential computations.
7.	Soft computing produces approximate results.	Hard computing produces precise results.
8.	Soft computing will emerge its own programs.	Hard computing requires programs to be written.
9.	Soft computing incorporates randomness .	Hard computing is settled.
10.	Soft computing will use multivalued logic.	Hard computing uses two-valued logic.

Ans c

Calculation of fitness value is done repeatedly in a GA and therefore it should be sufficiently fast. A slow computation of the fitness value can adversely affect a GA and make it exceptionally slow.

A fitness function should possess the following characteristics –

- The fitness function should be sufficiently fast to compute.
- It must quantitatively measure how fit a given solution is or how fit individuals can be produced from the given solution.

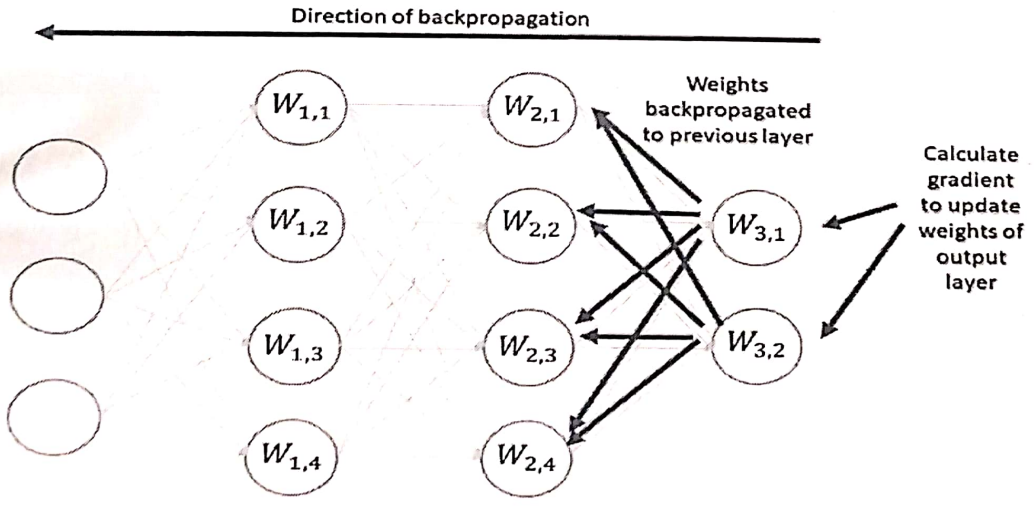
Ans d.

S.No	Crisp Set	Fuzzy Set
1	Crisp set defines the value is either 0 or 1.	Fuzzy set defines the value between 0 and 1 including both 0 and 1.
2	It is also called a classical set.	It specifies the degree to which something is true.
3	It shows full membership	It shows partial membership.
4	Eg1. She is 18 years old.	Eg1. She is about 18 years old.

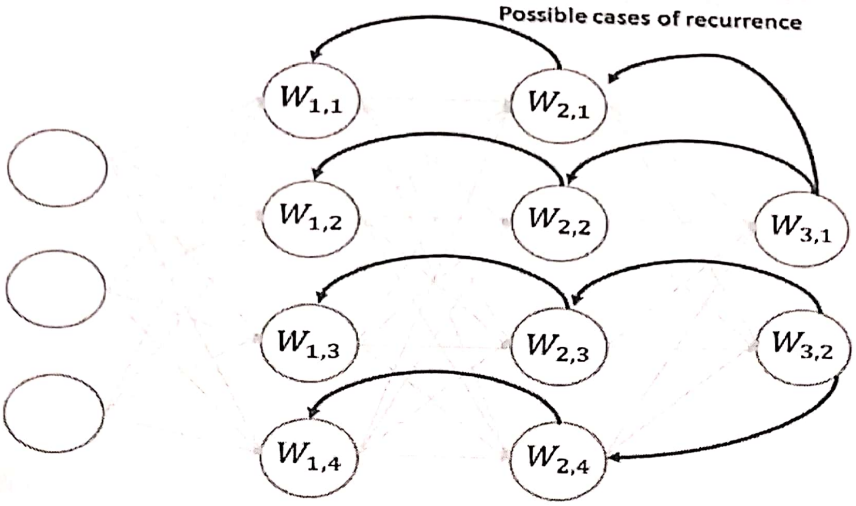
	Eg2. Rahul is 1.6m tall	Eg2. Rahul is about 1.6m tall.
5	Crisp set application used for digital design.	Fuzzy set used in the fuzzy controller.
6	It is bi-valued function logic.	It is infinite valued function logic
7	Full membership means totally true/false, yes/no, 0/1.	Partial membership means true to false, yes to no, 0 to 1.

Ans. e

In a feed-forward network, signals can only move in one direction. These networks are considered non-recurrent network with inputs, outputs, and hidden layers. A layer of processing units receives input data and executes calculations there. Based on a weighted total of its inputs, each processing element performs its computation. The newly derived values are subsequently used as the new input values for the subsequent layer. This process continues until the output has been determined after going through all the layers.



A feed-back network, such as a **recurrent neural network (RNN)**, features feed-back paths, which allow signals to use loops to travel in both directions. Neuronal connections can be made in any way. Since this kind of network contains loops, it transforms into a non-linear dynamic system that evolves during training continually until it achieves an equilibrium state.



Ans f. Fuzzy sets in FL

Fuzzy Logic uses the whole interval between 0 (false) and 1 (true) to describe human reasoning. • A Fuzzy Set is any set that allows its members to have different degree of membership, called membership function, having interval [0,1].

A fuzzy set A˜ in the universe of information U can be defined as a set of ordered pairs and it can be represented mathematically as –

$$\tilde{A} = \{(y, \mu_{\tilde{A}}(y)) \mid y \in U\}$$

Here $\mu_{\tilde{A}}(y)$ = degree of membership of y in \widetilde{A}, assumes values in the range from 0 to 1, i.e., $\mu_{\tilde{A}}(y) \in [0,1]$.

Ans g. Widrow Hoff Learning rule is same as Delta Learning Rule

It is widely used for finding the weights of an associative neural net. The training vector pairs here are denoted as s:t. The algorithm steps are given below:

Step0: set all the initial weights to 0

wij = 0

Step1: for each training target input out

output vector pairs s:t, perform steps 2-4

Step2: activate the input layer units to current training input.

xi = si (for i = 1 to n)

Step3: activate the output layer units to current target output,

yi = tj (for j = 1 to m)

Step4: start the weight adjustment

wij(new) = wij (old) + xiyi (for i =1 to n, j = 1 to m)

Ans. h Rule based learning

Rule-based machine learning (RBML) is a term in computer science intended to encompass any machine learning method that identifies, learns, or evolves 'rules' to store, manipulate or apply. The defining characteristic of a rule-based machine learner is the identification and utilization of a set of relational rules that collectively represent the knowledge captured by the system. Rules typically take the form of an *'{IF:THEN} expression'*, (e.g. {*IF 'condition' THEN 'result'*},

Ans i. Mutation: mutation may be defined as a small random tweak in the chromosome, to get a new solution. It is used to maintain and introduce diversity in the genetic population and is usually applied with a low probability – p_m. If the probability is very high, the GA gets reduced to a random search.

Mutation is the part of the GA which is related to the "exploration" of the search space.

Ans j. Fuzzy set operations with example

Given two Fuzzy sets A~ and B~

- **Union** : Fuzzy set C~ is union of Fuzzy sets A~ and B~ :

$$\mu_{\tilde{C}}(x) = \max(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)) \quad \tilde{C} = \tilde{A} \cup \tilde{B}$$

```
The First Fuzzy Set is : {'a': 0.2, 'b': 0.3, 'c': 0.6, 'd': 0.6}
The Second Fuzzy Set is : {'a': 0.9, 'b': 0.9, 'c': 0.4, 'd': 0.5}
Fuzzy Set Union is : {'a': 0.9, 'b': 0.9, 'c': 0.6, 'd': 0.6}
```

Intersection: Fuzzy set D~ is intersection of Fuzzy sets A~ and B~ :

$$\mu_{\tilde{D}}(x) = \min(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)) \quad \tilde{D} = \tilde{A} \cap \tilde{B}$$

```
The First Fuzzy Set is : {'a': 0.2, 'b': 0.3, 'c': 0.6, 'd': 0.6}
The Second Fuzzy Set is : {'a': 0.9, 'b': 0.9, 'c': 0.4, 'd': 0.5}
Fuzzy Set Intersection is : {'a': 0.2, 'b': 0.3, 'c': 0.4, 'd': 0.5}
```