PVP 23

Code: 23EE6601

III B.Tech - II Semester - Honors Examinations – APRIL 2026

AI APPLICATIONS IN POWER SYSTEMS
(HONORS in ELECTRICAL & ELECTRONICS ENGINEERING)

Duration: 3 hours Max. Marks: 70

Note: 1. This question paper contains two Parts A and B.
2. Part-A contains 10 short answer questions. Each Question carries 2 Marks.
3. Part-B contains 5 essay questions with an internal choice from each unit. Each Question carries 10 marks.
4. All parts of Question paper must be answered in one place.

BL – Blooms Level CO – Course Outcome

PART – A

		BL	CO
1.a)	Explain supervised learning.	L2	CO1
1.b)	Mention the limitations of perceptron training algorithm.	L2	CO1
1.c)	What is a Radial Basis Function (RBF) network?	L3	CO2
1.d)	State the delta rule for neural network learning.	L1	CO2
1.e)	Explain any two properties of fuzzy sets.	L2	CO3
1.f)	What is uncertainty in fuzzy systems?	L2	CO3
1.g)	Explain the role of the rule base in a fuzzy logic system.	L2	CO4
1.h)	Explain the output of the fuzzification process.	L3	CO4
1.i)	Why is Back Propagation used in load flow analysis?	L3	CO4
1.j)	What is meant by a single area power system?	L2	CO4

PART – B

			BL	CO	Max. Marks
UNIT-I					
2	a)	Explain the model of an artificial neuron with a neat diagram.	L2	CO1	5 M
	b)	Explain the learning rules used in neural networks with examples.	L3	CO1	5 M
OR					
3	a)	Explain classification using the Discrete Perceptron algorithm.	L2	CO1	5 M
	b)	Discuss the applications of perceptron networks in real-world problems.	L3	CO1	5 M
UNIT-II					
4	a)	Explain the architecture and learning process of Kohonen's Self-Organizing Feature Map (SOFM).	L3	CO2	5 M
	b)	Describe the Back Propagation algorithm and explain its working with steps.	L2	CO2	5 M
OR					
5	a)	Discuss the advantages of Functional Link Networks compared to multilayer networks.	L3	CO2	5 M
	b)	Explain the structure and working of the Hopfield Neural Network.	L2	CO2	5 M

UNIT-III					
6	a)	Differentiate between Crisp set and Fuzzy set. Also explain Fuzzy operators like Union, Intersection and complement with suitable example.	L2	CO3	5 M
	b)	Describe Fuzzy IF-THEN rule along with fuzzy inference system.	L3	CO3	5 M
OR					
7	a)	Explain various types of membership functions used in Fuzzy Systems.	L2	CO3	5 M
	b)	Explain Fuzzy relationship with suitable example.	L3	CO3	5 M
UNIT-IV					
8	a)	Define following terms: (1) Support (2)α-cut (3) Linguistic variable (4) Membership Function (5) Fuzzy singleton.	L2	CO4	5 M
	b)	What is de-fuzzification? Why it is required in Fuzzy Systems? Explain various methods of de-fuzzification.	L3	CO4	5 M
OR					
9	a)	For the application of Fuzzy logic system in voltage control, discuss the following (1) Selection of input variable and fuzzification (2) Development of Fuzzy rules for the same application.	L2	CO4	5 M
	b)	Find the crisp output 'y_1' and 'y_2' for the following membership functions using	L3	CO4	5 M

		COS method, assume x_1=4, x_2=4. Also comment on results. $$\mu(x_1) = \left\{\frac{0.3}{0}, \frac{0.6}{2}, \frac{1}{4}, \frac{1}{6}, \frac{0.6}{8}, \frac{0.3}{10}\right\},$$ $$\mu(x_2) = \left\{\frac{0.3}{0}, \frac{0.6}{2}, \frac{0.6}{4}, \frac{0.6}{6}, \frac{0.6}{8}, \frac{0.3}{10}\right\}$$			
UNIT-V					
10	a)	Explain the training procedure of load forecasting using back propagation algorithm.	L2	CO4	5 M
	b)	Explain single area load frequency control using fuzzy logic controller.	L3	CO4	5 M
OR					
11	a)	Discuss the role of fuzzy controllers in improving frequency stability in two-area systems.	L3	CO4	5 M
	b)	Discuss role of intelligent systems in Load forecasting.	L2	CO4	5 M

CODE: 23EE6601 **PVP23**

PVP SIDDHARTHA INSTITUTE OF TECHNOLOGY
III BTECH II SEM – HONORS EXAMINATION - A. Y: 2025-26
ELECTRICAL AND ELECTRONICS ENGINEERING
AI APPLICATIONS IN POWER SYSTEMS
SCHEME OF VALUATION

PART- A

1a) Supervised Learning Explanation ---------------------------**2M**

1b) Two limitations ---2M

1c) Radial base function description ----------------------------2M

1d) Delta rule ---2M

1e) Any two properties---2M

1f) Uncertainty in fuzzy system-----------------------------------2M

1g) Role of rule base in fuzzy in fuzzy logic -------------------2M

1h) Output of fuzzification--2M

1i) Back propagation used in load flow analysis----------------2M

1j) Single are power system--------------------------------------2M

PART- B

2a) Model Figure --3M
Explanation ---2M

2b) Any three rules with figures---------------------------------3M
Example ---2M

3a) Classification with algorithm --------------------------------5M

3b) Any 5 applications ---5M

4a) Architecture --3M
Learning process--2M

4b) Back propagation algorithm steps ----------------------------5M

5a) Any 5 advantages ---5M

5b) Structure --3M
Explanation--2M

6a) Difference between crisp and fuzzy set -------------------------2M
Properties--3M

6b) IF – THEN rule ---2M
Fuzzy inference system---3M

7a) Member ship function explanation-------------------------------5M

7b) Any one Fuzzy relationship explanation ------------------------5M

8a) Five definitions ---5M

8b) De-fuzzification --1M
Requirement --1M
Methods ---3M

9a) Selection of input variables --------------------------------------2.5M
Fuzzy rules application --2.5M

9 b) Formula --1M
Solution --3M
Comment --1M

10a) Training procedure--5M

10b) Explanation ---5M

11a) Explanation ---5M

11b) Explanation--5M

PVP SIDDHARTHA INSTITUTE OF TECHNOLOGY
III BTECH II SEM – HONORS EXAMINATION - A. Y: 2025-26
ELECTRICAL AND ELECTRONICS ENGINEERING
AI APPLICATIONS IN POWER SYSTEMS
KEY

PART- A

		BL	CO	MARKS
1.a)	**Explain supervised learning**	L2	CO1	2
	In supervised learning, we give the neural network training data that has correct answers already given. The network looks at the input and output examples and learns to match them. It keeps adjusting its weights to reduce the difference between what it predicts and what the actual answer is. This way it learns the pattern.			
1.b)	**Mention the limitations of perceptron training algorithm.**	L2	CO1	2
	1. Perceptron can only solve problems where data can be separated by a straight line. It cannot solve complex problems like XOR gate problem. 2. The perceptron has only one layer, so it cannot learn complicated patterns. For complex problems, we need multiple layers which perceptron doesn't have.			
1.c)	**What is a Radial Basis Function (RBF) network?**	L3	CO2	2
	RBF network is a type of neural network with 3 layers. The hidden layer uses special functions called radial basis functions (usually Gaussian shaped). These functions measure how far the input is from a center point. RBF networks learn faster than other networks and are good for pattern recognition and function approximation problems.			
1.d)	**State the delta rule for neural network learning**	L1	CO2	2
	The delta rule is a formula for updating weights in the neural network: Weight change = learning rate × error × derivative × input $\Delta w = \eta \times (\text{desired - actual}) \times f'(\text{net}) \times x$ Basically, it changes the weights based on how much the network made a mistake. Bigger mistakes mean bigger weight changes. This is the foundation of back propagation algorithm.			
1.e)	**Explain any two properties of fuzzy sets.**	L2	CO3	2
	1. Opposite/Complement: If something belongs to a fuzzy set with value 0.8, then it belongs to the opposite set with value 0.2 (= 1 - 0.8). For example, if "hot" is 0.8, then "not hot" is 0.2. 2. AND/Intersection: When we combine two fuzzy sets with AND, we take the smaller value. For example, if "hot" is 0.8 and "humid" is 0.6, then "hot AND humid" is 0.6 (the minimum).			

1.f)	**What is uncertainty in fuzzy systems?**	L2	CO3	2
	Uncertainty means things are not clear or exact in real world. Fuzzy systems handle this by using values between 0 and 1 instead of just yes/no. For example, instead of saying temperature is "hot" or "not hot", we can say it's "0.7 hot and 0.3 warm". This matches how humans think about things that are not perfectly clear or exact.			
1.g)	**Explain the role of the rule base in a fuzzy logic system.**	L2	CO4	2
	The rule base is like a set of instructions or rules that tell the system what to do. For example: "IF temperature is high AND humidity is high THEN turn on fan to high speed." These rules are written in simple language and come from expert knowledge. The system uses these rules to make decisions. If the rules are good, the whole system works well.			
1.h)	**Explain the output of the fuzzification process**	L3	CO4	2
	Fuzzification takes normal numbers and converts them into fuzzy values. For example, if temperature is 35°C (a number), fuzzification converts it to: "0.8 hot and 0.3 warm" (fuzzy values). So the output is membership values between 0 and 1 for different categories. These fuzzy values are then sent to the next step (rule checking). It basically translates real world numbers into fuzzy language that the system understands.			
1.i)	**Why is Back Propagation used in load flow analysis**	L3	CO4	2
	1. Power equations are complex: The equations for power systems are very complicated and non-linear. Back propagation network can learn these complicated relationships. 2. Fast calculation: Once the network is trained, it can calculate power flow answers very quickly, faster than traditional methods that take many steps. 3. Handles real variations: Real power systems have unexpected changes. The trained network can handle these variations and uncertainties well.			
1.j)	**What is meant by a single area power system?**	L2	CO4	2
	A single area power system is a simplified model of a power system. We treat the whole system as one area. All power plants are combined into one generator, and all loads are combined into one load. The whole area has the same frequency (50 Hz or 60 Hz). This simple model is used to study how the system behaves overall, without worrying about all the detailed regional differences. It's easier for neural networks to learn from.			

NOTE: Please award marks if the students have considered other relevant points related to the question.

PART B

PART B

	UNIT -I	BL	CO	MARKS
2a)	**Explain the model of an artificial neuron with a neat diagram**	L2	CO1	5M

An **artificial neuron** is the basic building block of neural networks. It simulates how biological neurons work by taking multiple inputs, processing them, and producing an output.

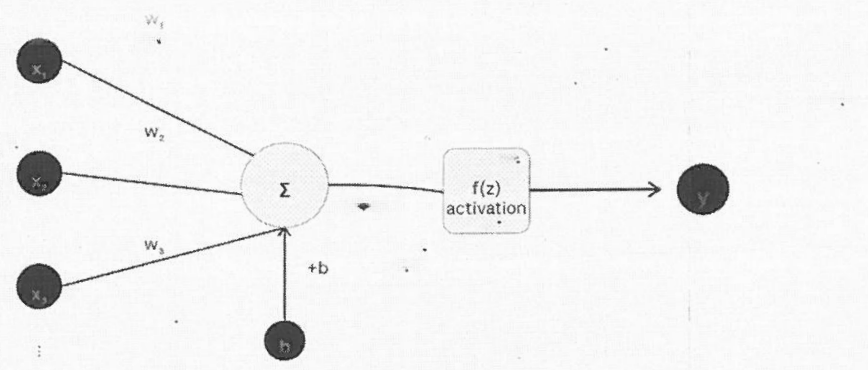
Key Components:

1. Inputs (x_1, x_2, x_3, ...) These are the feature values fed into the neuron. For example, in image recognition, inputs could be pixel values. In a medical diagnosis system, inputs might be patient attributes like age, blood pressure, etc.

2. Weights (w_1, w_2, w_3, ...) Each input has an associated weight that determines how much that input contributes to the final output. During training, these weights are adjusted to improve the neuron's performance. Larger weights mean that input has more influence.

3. Bias (b) The bias is an additional parameter that allows the neuron to shift the activation function. It provides flexibility in learning patterns and helps the neuron make better predictions even when all inputs are zero.

4. Summation (Σ) All weighted inputs are added together along with the bias. This is the linear combination: $z = w_1x_1 + w_2x_2 + w_3x_3 + ... + b$



Mathematical representation:

```
z = w₁x₁ + w₂x₂ + w₃x₃ + ... + b
y = f(z) where f is activation function (ReLU, sigmoid, tanh, etc.)
```

		BL	CO	MARKS
2 b)	**Explain the learning rules used in neural networks with examples.**	L3	CO1	5M

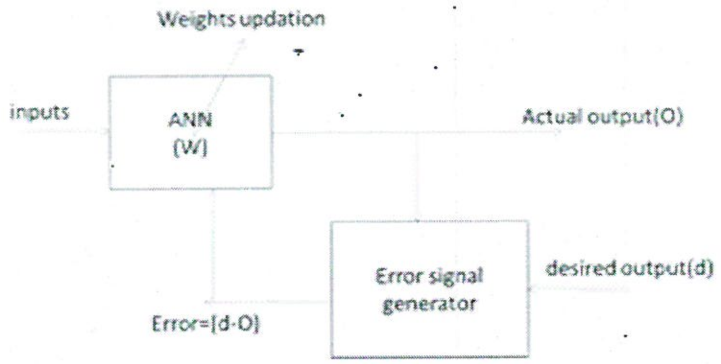
- Supervised learning rule
- Unsupervised learning rule
- Reinforced learning rule

Supervised Learning: When labeled data is available for prediction tasks like spam filtering, stock price forecasting.

Unsupervised Learning: When exploring data structures without predefined labels like customer segmentation, anomaly detection.

Reinforcement Learning: When decision-making is required in a dynamic environment like game AI, robotics, self-driving cars.

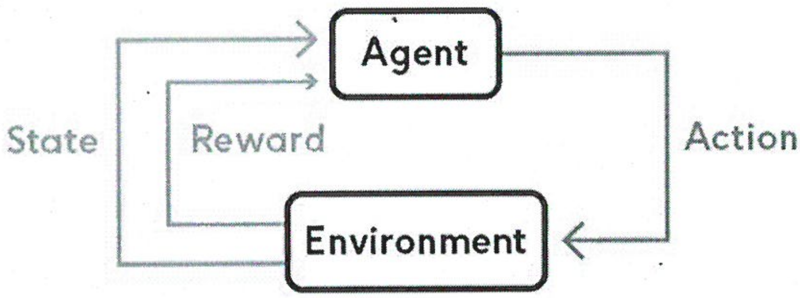
Supervised learning rule:



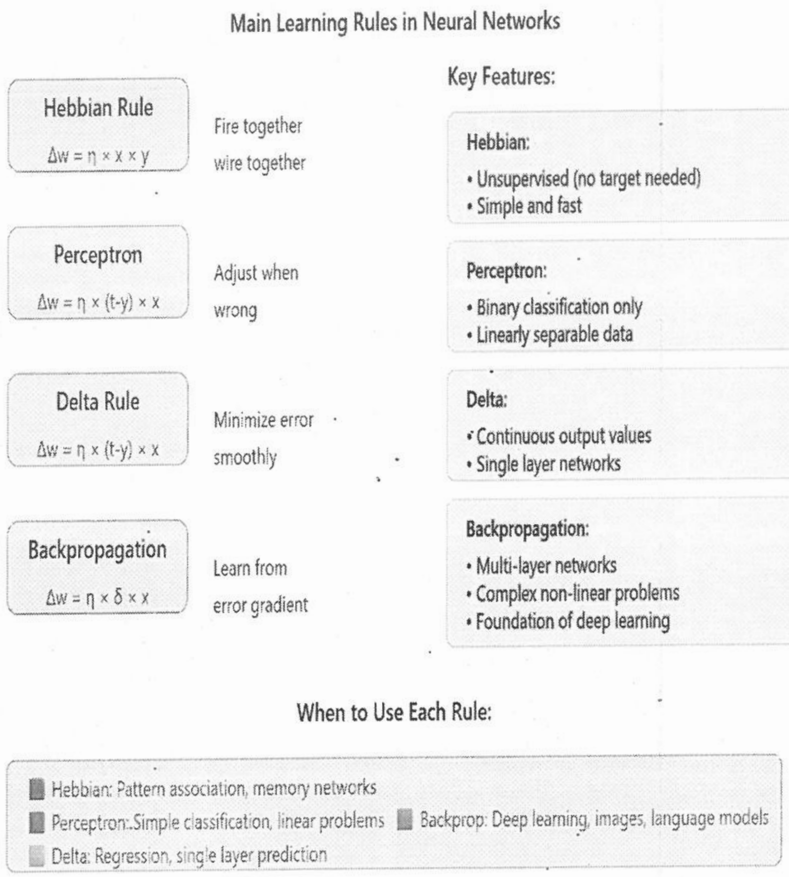
Unsupervised Learning:

- In this learning process we did not have desired output
- Network learns/ trains on its own
- Inputs corrects the weights

Reinforcement Learning:



Learning rules used in neural networks with examples:



(OR)

3a)	Explain Classification Using Discrete Perceptron Algorithm	L2	CO1	5M

The **Discrete Perceptron** is a fundamental algorithm for binary classification:

Algorithm Steps:

1. **Initialize weights and bias** to small random values (w, b)
2. **For each training sample (x, d)**:
   - Calculate the net input: $\mathbf{z = w \cdot x + b}$
   - Apply activation function: $\mathbf{y = f(z)}$ (threshold function)
     - If $z \geq 0$, then $y = 1$
     - If $z < 0$, then $y = 0$
   - Calculate error: $\mathbf{e = d - y}$ (desired - actual)
   - Update weights: $\mathbf{w = w + \eta \cdot e \cdot x}$
   - Update bias: $\mathbf{b = b + \eta \cdot e}$
3. **Repeat** for multiple epochs until convergence (error = 0)

Decision Boundary:

The perceptron creates a linear decision boundary: $\mathbf{w_1 \cdot x_1 + w_2 \cdot x_2 + b = 0}$

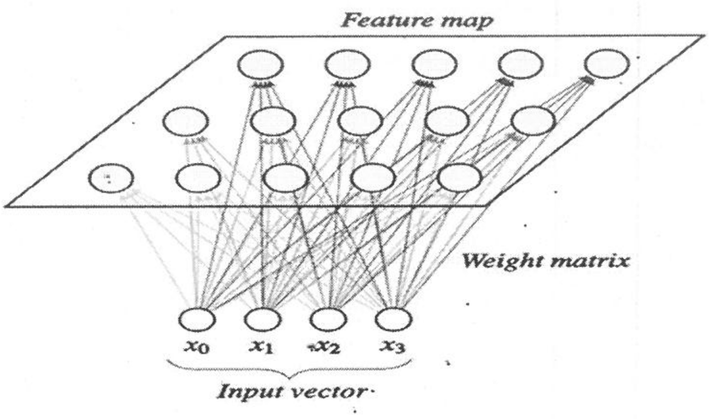
Limitations:

- Can only solve **linearly separable** problems
- Cannot solve XOR problem
- Requires discrete outputs

Example:

Binary classification of flowers as Type-A or Type-B based on petal length and width.

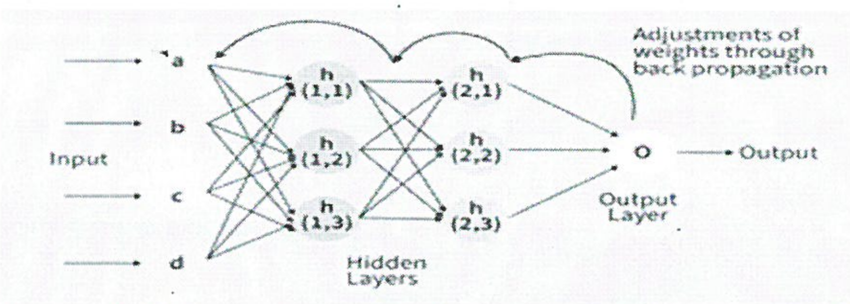
3 b)	Discuss the Applications of Perceptron in Real-World Problems	L3	CO1	5M
	1. Email Spam Classification • **Input**: Email features (word frequencies, sender reputation, domain) • **Output**: Spam (1) or Not-Spam (0) • **Application**: Filters unwanted emails **2. Medical Diagnosis** • **Input**: Patient test results (blood pressure, cholesterol, glucose levels) • **Output**: Disease present (1) or Absent (0) • **Application**: Early disease detection **3. Image Recognition (Simple)** • **Input**: Pixel values of small images • **Output**: Object class (cat vs dog) • **Application**: Basic image classification **4. Credit Approval** • **Input**: Credit history, income, debt ratio • **Output**: Approved (1) or Rejected (0) • **Application**: Automated loan decision systems **5. Pattern Recognition** • **Input**: Handwritten digit features • **Output**: Digit classification (0-9) • **Application**: Postal code reading, check processing **6. Fault Detection** • **Input**: Machine sensor readings (temperature, vibration, noise) • **Output**: Normal (0) or Faulty (1) • **Application**: Predictive maintenance in manufacturing **Note: Please award marks for any other correct and relevant applications/points written by the student related to the concept.**			
	UNIT-II			
4 a)	Explain the architecture and learning process of Kohonen's Self-Organizing Feature Map (SOFM)			
	The Self Organizing Map is one of the most popular neural models. It belongs to the category of the competitive learning network. The SOM is based on unsupervised learning, which means that is no human intervention is needed during the training and those little needs to be known about characterized by the input data. We could, for example, use the SOM for clustering membership of the input data. The SOM can be used to detect features inherent to the problem and thus has also been called SOFM the Self Origination Feature Map. Architecture:			



Architecture Components:

1. **Input Layer**: n-dimensional input vectors
2. **Output Layer**: 2D grid of neurons (typically rectangular or hexagonal)
3. **Weights**: Each neuron has a weight vector same dimension as input

learning process:
Algorithm:
Step:1
Initialize the weights wij. And initialize random weights
Step:2
Choose a random input vector x.
Step:3
Repeat steps 4 and 5 for all nodes on the map.
Step:4
Calculate the Euclidean distance between weight vector wij and the input vector x(t), and calculate the square of the distance
Step:5
Track the node that generates the smallest distance t and generate the winning weight using formula.

	Step:6 Calculate the overall Best Matching Unit (BMU). It means the node with the smallest distance from all calculated ones. **Step:7** Discover topological neighborhood of BMU in Kohonen Map. **Step:8** Repeat for all nodes in the BMU neighborhood: Update the winning weight of the first node in the neighborhood of the BMU by including a fraction of the difference between the input vector x(t) and the weight w(t) of the neuron. **Step:9** Repeat the complete iteration until reaching the iteration. Here, step 1 represents initialization phase, while step 2 to 9 represents the training phase.			
4 b)	Describe the back propagation algorithm and explain its working with steps	L2	CO2	5M
	 Step 0: Initialize weights and learning rate (take some small random values). Step 1: Perform Steps 2-9 when stopping condition is false. Step 2: Perform Steps 3-8 for each training pair.			

Feedforward Phase 1

Step 3: Each input unit receives input signal x_i and sends it to the hidden unit ($i = 1$ to n).

Step 4: Each hidden unit z_j ($j = 1$ to p) sums its weighted input signals to calculate net input:

$$z_{inj} = v_{0j} + \sum_{i=1}^{n} x_i v_{ij}$$

Calculate output of the hidden unit by applying its activation functions over z_{inj} (binary or bipolar sigmoidal activation function):

$$z_j = f(z_{inj})$$

and send the output signal from the hidden unit to the input of output layer units.

Step 5: For each output unit y_k ($k = 1$ to m), calculate the net input:

$$y_{ink} = w_{0k} + \sum_{j=1}^{p} z_j w_{jk}$$

and apply the activation function to compute output signal

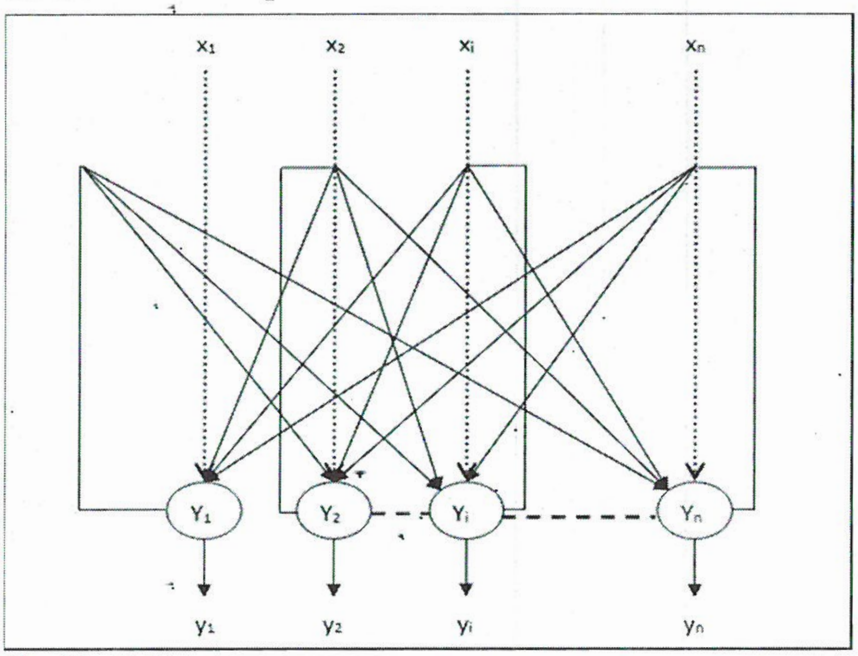
$$y_k = f(y_{ink})$$

	Back-propagation of error (Phase II) **Step 6**: Each output unit y_k ($k=1$ to m) receives a target pattern corresponding to the input training pattern and computes the error correction term: **Step 7**: Each hidden unit (z_j = 1 to p) sums its delta inputs from the output units: $$\delta_{inj} = \sum_{k=1}^{m} \delta_k w_{jk}$$ The term δ_{inj} gets multiplied with the derivative of $f(z_{inj})$ to calculate the error term: $$\delta_j = \delta_{inj} f'(z_{inj})$$ The derivative $f'(z_{inj})$ can be calculated as activation function section depending on whether binary or bipolar sigmoidal function is used. On the basis of the calculated δ_j, *Weight and bias updation (Phase III):* **Step 8**: Each output unit (y_k, k = 1 to m) updates the bias and weights: $$w_{jk}(new) = w_{jk}(old) + \Delta w_{jk}$$ $$w_{0k}(new) = w_{0k}(old) + \Delta w_{0k}$$ Each hidden unit (z_j; j = 1 to p) updates its bias and weights: $$v_{ij}(new) = v_{ij}(old) + \Delta v_{ij}$$ $$v_{0j}(new) = v_{0j}(old) + \Delta v_{0j}$$ **Step 9**: Check for the stopping condition. The stopping condition may be certain number of epochs reached or when the actual output equals the target output.			

	(OR)			
5 a)	Discuss the Advantages of Functional Link Networks (FLN) compared multilayer network.	L3	CO2	5M
	1. Simple Architecture • FLN has only **input and output layers**. • Hidden layers are not required. • Network structure is easy to design and implement. **Multilayer Network:** • Contains one or more hidden layers. • Architecture becomes complex. **2. Faster Training** • FLN requires fewer computations. • Weight adjustment is simple because there are no hidden layers. **Multilayer Network:** • Uses back propagation through many layers. • Training time is high. **3. Reduced Computational Complexity** • FLN performs direct mapping between expanded inputs and outputs. • Less memory and processing power are required. **Multilayer Network:** • Large number of neurons and connections increase complexity. **4. Avoids Local Minima Problems** • Since FLN has a simpler structure, optimization is easier. • Chances of getting trapped in local minima are reduced. **Multilayer Network:** • Back propagation may converge to local minima. **5. Better Convergence Speed** • FLN converges quickly during learning. • Suitable for real-time applications. **Multilayer Network:** • Convergence may be slow due to deep architecture. **6. Easy Hardware Implementation** • Simpler network requires fewer hardware resources. • Suitable for VLSI and embedded systems. **Multilayer Network:** • Hardware implementation is more difficult and costly. **7. Lower Storage Requirement** • Fewer weights and parameters are needed. • Memory usage is low. **Multilayer Network:** • Requires large storage for weights and biases. **8. Good Generalization for Simple Problems** • FLN performs effectively for linear and moderately nonlinear problems. • Functional expansion improves learning capability. **Multilayer Network:** • Often unnecessary for simple tasks. **Note: Please award marks for any other correct and relevant**			

	Advantages written by the student related to the concept.			
5 b)	**Explain the structure and working of the hope field network.**	L2	CO2	5M

Structure of the hope field network:



The Hopfield network is of two types

1. Discrete Hopfield Network
2. Continuous Hopfield Network

When this is operated in discrete line fashion it is called as discrete Hopfield network

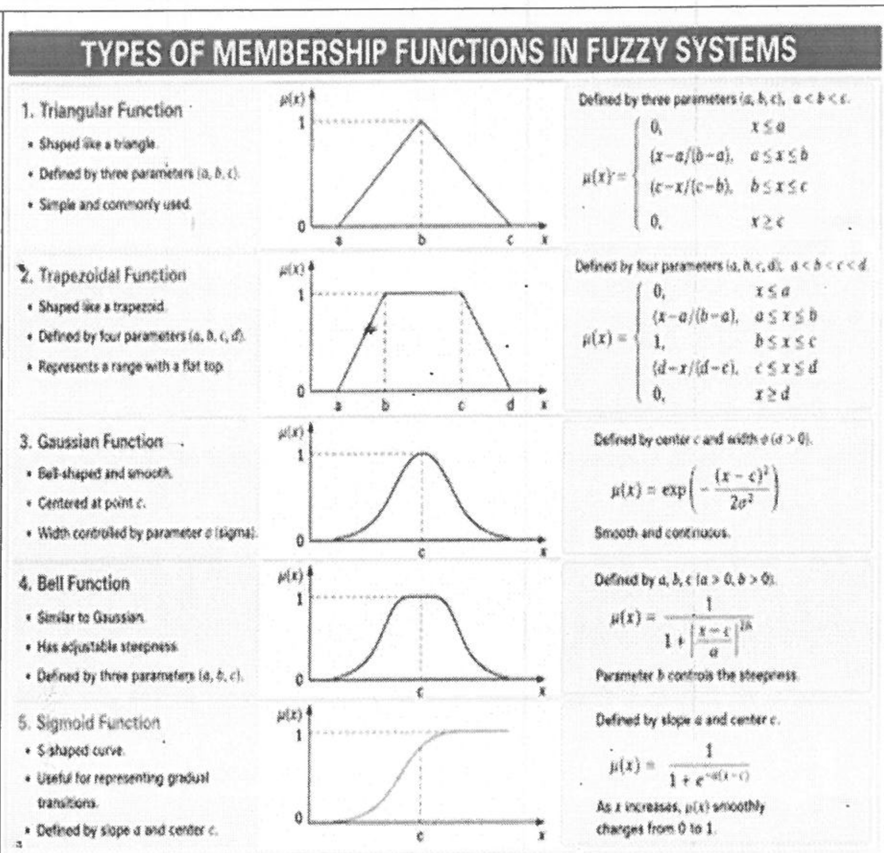
The network takes two-valued inputs: binary (0, 1) or bipolar (+1, -1); the use of bipolar inputs makes the analysis easier. The network has symmetrical weights with no self-connections, i.e.,

	$W_{ij} = W_{ji};$ **Training Algorithm of Discrete Hopfield Network** During training of discrete Hopfield network, weights will be updated. As we know that we can have the binary input vectors as well as bipolar input vectors. Let the input vectors be denoted by s(p), p = 1, ... , P. Then the weight matrix W to store a set of input vectors, where In case of input vectors being binary, the weight matrix W = $\{w_{ij}\}$ is given by $$w_{ij} = \sum_{p=1}^{P} [2s_i(p) - 1][2s_j(p) - 1] for i \neq j$$ When the input vectors are bipolar, the weight matrix W = $\{w_{ij}\}$ can be defined as $$w_{ij} = \sum_{p=1}^{P} [s_i(p)][s_j(p)] for i \neq j$$			
	UNIT-III			
6 a)	Differentiate between crisp set and fuzzy set. Also explain fuzzy operators like union, intersection, and complement with suitable example.	L2	CO3	5M

	Crisp Set: A crisp set has clear boundaries—an element either belongs to the set or it doesn't. For example, if we define a set of "tall people" with a crisp boundary at 6 feet, anyone 6 feet or taller is in the set, and anyone below is not. There's no in-between.
	Fuzzy Set: A fuzzy set allows partial membership. Elements can belong to a set with a degree of membership between 0 and 1. Using the same "tall people" example, a person who is 5'10" might have a membership degree of 0.7 (somewhat tall), while 6'2" might have 0.9 (very tall).
	
	Fuzzy Operators with Examples:
	1. **Union (OR):** Combines two fuzzy sets. The membership degree is the maximum of the two values.
	    o *Example:* If Set A (warm) = 0.6 and Set B (humid) = 0.8, then Union = max(0.6, 0.8) = 0.8
	2. **Intersection (AND):** The common part of two fuzzy sets. The membership degree is the minimum of the two values.
	    o *Example:* If Set A (warm) = 0.6 and Set B (humid) = 0.8, then Intersection = min(0.6, 0.8) = 0.6
	3. **Complement (NOT):** The opposite of a fuzzy set. The membership degree becomes 1 minus the original value.
	    o *Example:* If Set A (cold) = 0.3, then Complement = 1 - 0.3 = 0.7 (warm)			

6 b)	Describe fuzzy IF – THEN rule along with fuzzy inference system.	L3	CO3	5M

	Fuzzy IF-THEN Rule: A fuzzy IF-THEN rule is a conditional statement that relates input fuzzy sets to output fuzzy sets. • *Structure:* IF (condition) THEN (consequence) • *Example:* "IF temperature is high AND humidity is high THEN fan speed is very fast" **Fuzzy Inference System (FIS):** A Fuzzy Inference System (FIS) is a system that uses fuzzy logic and fuzzy IF–THEN rules to convert input values into output values. It is used for decision making and control applications where information is uncertain or imprecise. FIS works similarly to human reasoning. Instead of using exact values, it uses linguistic terms such as low, medium, high, slow, and fast. The main components of a Fuzzy Inference System are: **1. Fuzzification** Fuzzification converts crisp numerical inputs into fuzzy values using membership functions. **2. Knowledge Base** The knowledge base contains: • Database: defines membership functions. • Rule Base: contains fuzzy IF–THEN rules. Example: IF temperature is High THEN fan speed is Fast. **3. Inference Engine** The inference engine evaluates the fuzzy rules and determines the fuzzy output using logical operations such as AND and OR. **4. Defuzzification** Defuzzification converts the fuzzy output into a crisp numerical output.			
	(OR)			
7 a)	Explain the various types of membership functions used in fuzzy systems.	L2	CO3	5M
	• **Triangular Function:** Shaped like a triangle, defined by three parameters (a, b, c). Simple and commonly used. • **Trapezoidal Function:** Shaped like a trapezoid, defined by four parameters. Represents a range with a flat top. • **Gaussian Function:** Bell-shaped, smooth, centered at a point with a specified width. • **Bell Function:** Similar to Gaussian but with adjustable steepness. • **Sigmoid Function:** S-shaped curve, useful for representing gradual transitions.			



7 b)	Explain Fuzzy Relationship with suitable Example	L3	CO3	5M

FUZZY RELATIONS

Fuzzy relation is a fuzzy set defined on the Cartesian product of crisp sets X 1, X 2,..., Xn where the n-tuples (x 1, x 2,..., xn) may have varying degrees of membership within the relation. The membership values indicate the strength of the relation between the tuples.

Example

Let R be the fuzzy relation between two sets X 1 and X 2 where X 1 is the set of diseases and X 2 is the set of symptoms.

X 1 = {typhoid, viral fever, common cold}

X 2 = {running nose, high temperature, shivering}

	Suppose A is a fuzzy set on the universe of discourse X with $\mu_A(x) \mid x \in X$ B is a fuzzy set on the universe of discourse Y with $\mu_B(y) \mid y \in Y$ Then $R = A \times B \subset X \times Y$; where R has its membership function given by $\mu_R(x,y) = \mu_{A \times B}(x,y) = min\{\mu_A(x), \mu_B(y)\}$ Example : $A = \{(a_1, 0.2), (a_2, 0.7), (a_3, 0.4)\}$ and $B = \{(b_1, 0.5), (b_2, 0.6)\}$ $R = A \times B = \begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \end{array} \begin{array}{c} \begin{matrix} b_1 & b_2 \end{matrix} \\ \begin{bmatrix} 0.2 & 0.2 \\ 0.5 & 0.6 \\ 0.4 & 0.4 \end{bmatrix} \end{array}$			
	UNIT-IV			
8 a)	Define the following 1) support 2) α-cut 3)linguistic variables 4) member ship function 5) fuzzy singleton	L2	CO4	5M
	Definitions 1. **Support:** The set of elements that have a non-zero membership degree in a fuzzy set. Elements where membership > 0. 2. **α-cut (Alpha-cut):** A crisp set containing all elements with membership degrees greater than or equal to a specific threshold α (alpha). Used to convert fuzzy sets to crisp sets. 3. **Linguistic Variable:** A variable that takes on linguistic values rather than numerical values. o *Example:* Temperature can be "cold," "warm," or "hot" instead of specific degrees. 4. **Membership Function:** A mathematical function that defines how each element is mapped to a degree of membership (0 to 1) in a fuzzy set. 5. **Fuzzy Singleton:** A fuzzy set where only one element has			

	a membership degree of 1, and all other elements have 0.			
8b)	**What is De-fuzzification? Why is it Required in fuzzy system? Explain the various Methods of De-fuzzification**	L3	CO4	5M
	De-fuzzification is the process of converting fuzzy output values back into crisp (conventional) numerical values that can be used in real-world applications. • Controllers and mechanical systems need exact numerical commands, not fuzzy values • Makes fuzzy system outputs practical and usable in real applications • Bridges the gap between fuzzy logic processing and actual system control Centroid Method (Center of Gravity): Calculates the center point of the output fuzzy set. Most commonly used. The following are the known methods of defuzzification. • Center of Sums Method (COS) • Center of gravity (COG) / Centroid of Area (COA) Method • Center of Area / Bisector of Area Method (BOA) • Weighted Average Method			

Center of Sums (COS) Method

This is the most commonly used defuzzification technique. In this method, the overlapping area is counted twice.

The defuzzified value x^* is defined as :

$$x^* = \frac{\sum_{i=1}^{N} x_i \cdot \sum_{k=1}^{n} \mu_{A_k}(x_i)}{\sum_{i=1}^{N} \quad \sum_{k=1}^{n} \mu_{A_k}(x_i)},$$

Here, n is the number of fuzzy sets, N is the number of fuzzy variables, $\mu_{A_k}(x_i)$ is the membership function for the k-th fuzzy set.

Center of gravity (COG) / Centroid of Area (COA) Method

This method provides a crisp value based on the center of gravity of the fuzzy set. The total area of the membership function distribution used to represent the combined control action is divided into a number of sub-areas. The area and the center of gravity or centroid of each sub-area is calculated and then the summation of all these sub-areas is taken to find the defuzzified value for a discrete fuzzy set.

For discrete membership function, the defuzzified value denoted as x^* using COG is defined as:

$$x^* = \frac{\sum_{i=1}^{n} x_i \cdot \mu(x_i)}{\sum_{i=1}^{n} \mu(x_i)}$$

, Here x_i indicates the sample element, $\mu(x_i)$ is the membership function, and n represents the number of elements in the sample.

For continuous membership function, x^* is defined as :

$$x^* = \frac{\int x \,\mu_A(x)\, dx}{\int \mu_A(x)\, dx}$$

Center of Area / Bisector of Area Method (BOA)

This method calculates the position under the curve where the areas on both sides are equal. The BOA generates the action that partitions the area into two regions with the same area.

$\int_{\alpha}^{x^*} \mu_A(x)\; dx = \int_{x^*}^{\beta} \mu_A(x) dx$, where $\alpha = \min \{x \mid x \in X\}$ and $\beta = \max \{x \mid x \in X\}$

Weighted Average Method

This method is valid for fuzzy sets with symmetrical output membership functions and produces results very close to the COA method. This method is less computationally intensive. Each membership function is weighted by its maximum membership value. The defuzzified value is defined as :

	(OR)			
9 a)	For the application of Fuzzy logic system in voltage control, discus(1) Selection of input variable and fuzzification (2) Development of Fuzzy rules for the same application	L2	CO4	5M

1) Selection of Input Variables and Fuzzification

Selection of Input Variables

In voltage control, the controller must know: How much the voltage differs from the required value. Whether the error is increasing or decreasing. Therefore two input variables are selected.

(a) Voltage Error (E)

Voltage error is the difference between reference voltage and actual output voltage.

$$E = V_{ref} - V_{out}$$

Where:

- V_{ref} = Desired voltage
- V_{out} = Actual output voltage

Meaning

- If error is positive → output voltage is low.
- If error is negative → output voltage is high.
- If error is zero → voltage is correct.

(b) Change in Error (ΔE)

It shows how the error changes with time.

$$\Delta E = E(k) - E(k-1)$$

Where:

- $E(k)$ = Present error
- $E(k-1)$ = Previous error

Meaning

- Positive ΔE → error is increasing.
- Negative ΔE → error is decreasing.

2) Development of Fuzzy Rules for Voltage Control

After fuzzification, fuzzy rules are developed.

The rules are based on operator experience and system behavior.

The rules are written in IF–THEN form.

Rule 1

IF error is PB

AND change in error is PB

	THEN increase voltage greatly. Meaning: • Output voltage is very low and becoming worse. • Strong correction is needed. **Rule 2** IF error is ZE AND change in error is ZE THEN no control action. Meaning: • Voltage is already correct. **Rule 3** IF error is NB AND change in error is NB THEN decrease voltage greatly. Meaning: • Voltage is too high. • Controller must reduce it quickly.			
9 b)	Find the crisp output 'y₁' and 'y₂' for the following membership functions using COS method, assume $x_1=4$, $x_2=4$. Also comment on results. **Given Membership Functions** $\mu(x_1) = \left\{ \frac{0.3}{0}, \frac{0.6}{2}, \frac{1}{4}, \frac{1}{6}, \frac{0.6}{8}, \frac{0.3}{10} \right\}$ $\mu(x_2) = \left\{ \frac{0.3}{0}, \frac{0.6}{2}, \frac{0.6}{4}, \frac{0.6}{6}, \frac{0.6}{8}, \frac{0.3}{10} \right\}$ **CENTER OF SUM (COS) FORMULA:** The Center of Sum is a defuzzification method that calculates the weighted average of all output fuzzy sets: $y_{crisp} = \frac{\sum_{i=1}^{n} \mu(x_i) \times x_i}{\sum_{i=1}^{n} \mu(x_i)}$	L3	CO4	5M

For the given data, the exact value of y_1 using the standard **COS (Center of Sum / Weighted Average** formula is:

$$y_1 = \frac{\sum x_i \mu(x_i)}{\sum \mu(x_i)}$$

Substituting values:

$$y_1 = \frac{(0)(0.3) + (2)(0.6) + (4)(1) + (6)(1) + (8)(0.6) + (10)(0.3)}{0.3 + 0.6 + 1 + 1 + 0.6 + 0.3}$$

Numerator:

$$= 0 + 1.2 + 4 + 6 + 4.8 + 3$$
$$= 19$$

Denominator:

$$= 0.3 + 0.6 + 1 + 1 + 0.6 + 0.3$$
$$= 3.8$$

Therefore,

$$y_1 = \frac{19}{3.8} = 5$$

So the exact answer is:

$$\boxed{y_1 = 5}$$

Calculation of y_2

$$y_2 = \frac{(0)(0.3) + (2)(0.6) + (4)(0.6) + (6)(0.6) + (8)(0.6) + (10)(0.3)}{0.3 + 0.6 + 0.6 + 0.6 + 0.6 + 0.3}$$

Substitute fractions:

$$y_2 = \frac{0 + 2\left(\frac{3}{5}\right) + 4\left(\frac{3}{5}\right) + 6\left(\frac{3}{5}\right) + 8\left(\frac{3}{5}\right) + 10\left(\frac{3}{10}\right)}{\frac{3}{10} + \frac{3}{5} + \frac{3}{5} + \frac{3}{5} + \frac{3}{5} + \frac{3}{10}}$$

Numerator:

$$= \frac{6}{5} + \frac{12}{5} + \frac{18}{5} + \frac{24}{5} + 3$$
$$= \frac{6 + 12 + 18 + 24 + 15}{5}$$
$$= \frac{75}{5} = 15$$

Denominator:

$$= \frac{3}{10} + \frac{6}{10} + \frac{6}{10} + \frac{6}{10} + \frac{6}{10} + \frac{3}{10}$$
$$= \frac{30}{10} = 3$$

Therefore,

$$y_2 = \frac{15}{3}$$

$$\boxed{y_2 = 5}$$

Comment on the Results

	Both membership functions produce the same crisp value because they are symmetric about the center point x=5.			
	UNIT-V			
10 a)	Explain the training procedure of load forecasting using back propagation algorithm.	L2	CO4	5M
	Load forecasting is the prediction of future electrical load demand using historical data. Artificial Neural Networks (ANN) trained by the Back Propagation (BP) algorithm are widely used for accurate load forecasting. **Step 1: Collection of Input Data** Historical data related to load demand is collected. The input data may include: • Previous load values • Temperature • Humidity • Day of the week • Seasonal information • Time information The target output is the future load demand **Step 2: Preprocessing of Data** The collected data is normalized into a suitable range such as 0 to 1. Normalization improves: • Training speed • Accuracy • Stability of the neural network **Step 3: Design of Neural Network** A multilayer feed forward neural network is designed. The network contains: 1. Input layer 2. Hidden layer 3. Output layer • Input layer receives load data. • Hidden layer performs processing. • Output layer gives forecasted load. **Step 4: Initialization of Weights** Initially, small random values are assigned to: • Weights • Biases These values are adjusted during training. **Step 5: Forward Propagation** Input data is applied to the network. Each neuron calculates: The output is obtained using an activation function such as sigmoid function. The network produces the forecasted load value. $$Net = \sum w_i x_i + b$$			

Step6: Error Calculation
The output obtained from the network is compared with the target value.
The error is calculated as:

$$E = Target - Output$$

Mean Square Error (MSE) is generally used.

Step 7: Back Propagation of Error

The calculated error is propagated backward from output layer to hidden layer.

The weights are updated using:

$$W_{new} = W_{old} + \eta \delta x$$

10 b)	**Explain single area load frequency control using fuzzy logic controller.**	L3	CO4	5M

Load Frequency Control (LFC) is used to maintain system frequency at its nominal value when load changes occur. In a single area power system, sudden load increase causes a decrease in frequency. A Fuzzy Logic Controller (FLC) automatically controls the generator input to restore the frequency.

Single Area Load Frequency Control
A single area power system consists of:

- Governor
- Turbine
- Generator
- Load
- Fuzzy Logic Controller

The controller maintains frequency stability.

Selection of Input Variables
The inputs to the fuzzy controller are:

1. Frequency error
2. Change in frequency error

These inputs indicate:

- Magnitude of deviation
- Rate of change of deviation

Step 3: Fuzzification
The crisp inputs are converted into linguistic variables.
Typical fuzzy sets are:

- NB → Negative Big
- NS → Negative Small

	• ZE → Zero • PS → Positive Small • PB → Positive Big Membership functions are used for fuzzification. **Step 4: Rule Base Formation** The fuzzy controller uses IF–THEN rules. Example rules: • IF error is PB AND change in error is PS THEN decrease generation. • IF error is NB AND change in error is NB THEN increase generation. • IF error is ZE THEN maintain same control action. **Step 5: Fuzzy Inference** The inference engine evaluates all fuzzy rules and determines the control action. **Step 6: Defuzzification** The fuzzy output is converted into a crisp control signal. This signal controls: • Governor • Turbine input Thus system frequency is restored to normal value.			
	(OR)			
11a)	Discuss the role of fuzzy controllers in improving frequency stability in two-area systems.	L3	CO4	5M
	A two-area power system consists of two interconnected generating areas connected through a tie line. When load changes occur in one area, both system frequency and tie line power are disturbed. Fuzzy controllers are used to improve frequency stability and reduce oscillations. **Role of Fuzzy Controllers** A fuzzy controller provides intelligent control action based on system conditions. The controller uses: 1. Frequency deviation 2. Tie line power deviation 3. Change in error as input variables. **Working of Fuzzy Controller** **Step 1: Measurement of Deviations** The controller continuously measures: • Frequency error • Tie line power error **Step 2: Fuzzification** The measured values are converted into linguistic variables such as: <table><tr><th>Abbreviation</th><th>Meaning</th></tr><tr><td>NB</td><td>Negative Big</td></tr><tr><td>NS</td><td>Negative Small</td></tr><tr><td>ZE</td><td>Zero</td></tr><tr><td>PS</td><td>Positive Small</td></tr></table>			

	PB — Positive Big **Step 3: Rule Evaluation** The fuzzy controller applies IF–THEN rules. Example: • IF frequency deviation is NB AND tie line deviation is PB THEN increase generation greatly. • IF frequency deviation is ZE THEN no control action. **Step 4: Defuzzification** The fuzzy output is converted into a crisp control signal. The signal controls generator input through the governor.			
11 b)	Discuss role of intelligent systems in load forecasting.	L2	CO4	5M
	Load forecasting is the prediction of future electrical power demand. Intelligent systems are widely used because they provide accurate forecasting even for complex and nonlinear systems. Intelligent techniques include: • Artificial Neural Networks (ANN) • Fuzzy Logic Systems **Role of Intelligent Systems in Load Forecasting** **1. Handling Nonlinear Relationships** Electrical load depends on: • Weather conditions • Population • Industrial demand • Seasonal variations Intelligent systems can model these nonlinear relationships accurately. **2. Learning from Historical Data** Artificial Neural Networks learn from previous load patterns and improve forecasting accuracy automatically. **3. Fast and Accurate Prediction** Intelligent systems provide: • Faster computation • Accurate load prediction • Reduced forecasting error **4. Handling Uncertainty** Fuzzy logic systems handle uncertain and vague information effectively. Example: • High temperature • Moderate demand • Slight load increase **5. Adaptive Nature** Intelligent systems can adapt to changing load conditions without redesigning the entire model. **6. Optimization Capability** Genetic algorithms optimize: • Network parameters • Membership functions			

	• Forecasting accuracy				