🔹 2. Models of Neurons

a. McCulloch & Pitts Model

- **Binary neurons**.
- No learning—just a fixed rule.
- Example: Logic Gate –
    - Inputs: x1 = 1, x2 = 1
    - Weights: w1 = 1, w2 = 1
    - Threshold = 2
    - Output = 1 if (x1 + x2 ≥ 2), else 0
      ➤ Works like an **AND gate**.

b. Perceptron

- Can learn from labeled data.
- Works for problems like **linearly separable classification** (e.g., separating red and blue dots).

c. Adaline (Adaptive Linear Neuron)

- Similar to perceptron but:
    - Uses **Mean Squared Error (MSE)** loss
    - Output is **continuous**, not binary
    - Learning happens by **gradient descent**

🔹 3. Basic Learning Laws

These define **how weights are updated** in neurons.

Common Laws:

- **Hebbian Learning**:
  "Neurons that fire together, wire together."
    - Example: If x and y activate together, strengthen their connection.
- **Delta Rule (Widrow-Hoff Rule)**:
    - Used in Adaline
    - Update = learning rate × (target - output) × input

4. Perceptron Learning Algorithm

A Perceptron is the simplest type of artificial neural network, used for binary classification.

It simulates the behavior of a biological neuron. It takes several inputs, applies weights, sums them, and passes the result through an activation function (usually a step function) to give a binary output (0 or 1).

5. Supervised Learning

Supervised Learning is a type of machine learning where the model is trained on a labeled dataset – meaning each input has a corresponding correct output.

👉 The algorithm "learns by example" — it maps inputs to desired outputs by minimizing errors.

🔹 **6. Backpropagation**

Backpropagation (short for *"backward propagation of errors"*) is a **supervised learning algorithm** used to **train multi-layer neural networks** by minimizing the error between predicted and actual output.

It works by adjusting the weights in the network **using gradient descent**.

🔹 **7. Multilayered Network**

A **Multilayered Network**, also called a **Multilayer Perceptron (MLP)**, is a type of **feedforward artificial neural network** with **one or more hidden layers** between the input and output layers.

Unlike a single-layer perceptron, it can **solve non-linear and complex problems**.

8. Feedforward Neural Networks (FNN)

A **Feedforward Neural Network** is the **simplest type** of artificial neural network where the connections between the nodes **do not form cycles**.

9. Feedback Neural Networks (Recurrent Neural Networks - RNNs)

A **Feedback Neural Network** (also known as a **Recurrent Neural Network**) allows connections that **form cycles** – meaning the output of neurons can be **fed back as input** to the same or previous layers.

1. Associative Learning

Associative Learning is a type of learning where an **association is formed between input and output patterns**.

💡 Think of it like **human memory** – when you see a face, you recall the person's name.

2. Hopfield Network

A **Hopfield Network** is a type of **recurrent neural network** used for **auto-associative memory**. It stores patterns and retrieves them when given partial or noisy inputs.

🔷 **Situation:**

Imagine you're looking at a **blurry or incomplete photo of your friend's face**, but your brain can still **recognize the person**.

🔷 **1. Boltzmann Machine (BM)**

A **Boltzmann Machine** is a type of **stochastic recurrent neural network** that can learn and store complex probability distributions over binary patterns.

🔷 **2. Boltzmann Learning**

This is the **learning algorithm** used to train Boltzmann Machines. It aims to **adjust weights** so that the network's distribution matches the distribution of the training data.

🔹 **1. Pattern Association**

📌 **Definition**: Pattern association is the process where a neural network **learns to recall** a specific output **pattern** when a **related input pattern** is presented.

🔁 **Think of:**

- **Memory recall**: You see a half-visible image of a car, and your brain fills in the rest.

Example:

- Input: Noisy A
- Output: Clean A

2. Pattern Classification

📌 **Definition**: Pattern classification is the process of assigning an input pattern to **one of several predefined categories** or **classes**.

🔁 **Think of:**

- Email spam filter: Is this email **spam** or **not spam**?

🧠 **Example:**

- Input: [height=180, weight=80]
- Output class: Male

3. Pattern Mapping

📌 **Definition**: Pattern mapping is the process of transforming an **input pattern** into a **different output pattern**, possibly of different size/dimensionality.

🔁 **Think of:**

- **Regression tasks** or **sequence-to-sequence models**

🧠 **Example:**

- Input: English sentence → Output: French sentence

🔹 **1. CL Network (Competitive Learning Network)**

📌 **Definition:**

A **Competitive Learning Network (CL Network)** is an unsupervised learning algorithm where neurons **compete** to respond to a given input, and the **winning neuron** is responsible for the output. It is designed to **cluster similar patterns** together.

🔧 **Example:**

Consider a network learning different **flower species** based on **petal and sepal length**:

- Input: [4.5, 3.2]
- Output: Neuron 2 wins, which corresponds to a particular species cluster.

🔹 **2. Pattern Clustering**

📌 **Definition:**

Pattern clustering is the process of grouping input patterns into **clusters** where **patterns within the same cluster are similar**, and **patterns from different clusters are dissimilar**.

🔧 **Example:**

Given a dataset of **animal features** like size, weight, and diet:

- Cluster 1: Small, herbivorous animals (e.g., rabbits)
- Cluster 2: Large, carnivorous animals (e.g., lions)

3. Feature Mapping Network

📌 **Definition:**

A **Feature Mapping Network** is a type of network that **transforms the input space** into a **feature space** to make it easier for the model to find patterns or relationships between features.

🔹 **Convolutional Neural Networks (CNNs)**

Convolutional Neural Networks (CNNs) are a class of deep learning models primarily used for **image recognition**, **video analysis**, **object detection**, and other tasks that involve grid-like data (such as images).

CNNs are particularly effective because they can **automatically learn spatial hierarchies** of features, which means they can recognize patterns like edges, textures, shapes, and more complex patterns as the network deepens.

🔹 **Large Language Models (LL LLM)**

Large Language Models (LLMs) are a type of neural network specifically designed for processing and generating human language. These models are called **"large"** because they have **billions** (or even trillions) of parameters, which enable them to learn intricate language patterns and generate highly sophisticated language outputs.

Self-Organizing Map (SOM)

A **Self-Organizing Map (SOM)** is an unsupervised artificial neural network that is used for clustering and dimensionality reduction. It maps high-dimensional data onto a lower-dimensional grid (typically 2D) while preserving the **topological structure** of the data. It's particularly useful in visualizing high-dimensional data and performing clustering.

🧳 **Example: Customer Segmentation for Marketing**

Scenario:
Imagine a retail company that wants to **segment its customer base** to tailor personalized marketing strategies. The company has data on various customer attributes such as:

- **Age**

- **Annual income**
- **Shopping habits** (e.g., frequency of purchase, product categories, etc.)

These customer attributes form a high-dimensional dataset. The company wants to **group similar customers together** to create targeted marketing campaigns. This is where **Self-Organizing Maps (SOM)** can be very useful.

Adaptive Pattern Classification

Adaptive Pattern Classification refers to the process of classifying patterns (such as data points, objects, or signals) into predefined categories or classes, with the system being capable of adapting and improving its performance as it encounters new data. It is a dynamic approach, where the classifier adjusts over time based on the input data to improve accuracy, flexibility, and robustness.

---

1. Padding in Convolutional Layers

Padding refers to the practice of adding extra pixels (typically zeros) around the input image or feature map before performing convolution. Padding is used to control the spatial dimensions of the output feature map.

Deep Learning Frameworks

Deep Learning frameworks provide tools, libraries, and pre-built modules to efficiently design, train, and deploy deep learning models. These frameworks abstract the complexity of low-level mathematical operations and provide high-level APIs that make it easier to work with neural networks.

Multitask Learning

Multitask Learning (MTL) is a type of learning paradigm where multiple related tasks are learned simultaneously using shared representations. In MTL, the model learns to perform multiple tasks by leveraging common features and shared knowledge across the tasks.

End-to-End Deep Learning

End-to-End Deep Learning refers to the concept of training a model directly from raw input to the final output in one unified process, without requiring manual feature extraction or intermediate steps.

1. Texture Classification

Texture Classification is the process of identifying the type of texture in a given image region. This is useful in many applications such as medical imaging, satellite imagery, and material inspection, where distinguishing between different textures can be critical.

2. Texture Segmentation

Texture Segmentation refers to the process of partitioning an image into regions based on their texture. In other words, segmentation involves dividing an image into areas that have similar textures, which is useful in scenarios where objects with different textures need to be identified or isolated.