Task Description & Findings: XOR with Perceptron vs. Multi-Layer Perceptron (MLP)

Objective:

To implement a neural network that learns the XOR logic function, which outputs 1 when the inputs differ and 0 when they are the same.

Why a Single Perceptron Fails

* A single-layer perceptron computes a linear decision boundary.
* The XOR function is not linearly separable—there’s no straight line that can separate the 1s and 0s in its input-output mapping.
* As a result, a simple perceptron cannot learn XOR, regardless of training.

How It Was Solved: Using a Multi-Layer Perceptron (MLP)

* We implemented a 2-layer feedforward neural network:
  + 2 input neurons
  + 2 hidden neurons
  + 1 output neuron
* We used sigmoid activation and trained the network using backpropagation.
* After training on the XOR truth table, the network successfully learned to output correct values.

Findings:

* Non-linear problems like XOR require multiple layers to be solved.
* The hidden layer enables the network to learn non-linear transformations of the input space.
* This demonstrates the power of MLPs and lays the foundation for deeper neural networks used in complex tasks like image and speech recognition.