

## EXPERIMENT NO. 5

**Note:** use this link: <https://scte-iitkgp.vlabs.ac.in/exp/multilayer-perceptron/theory.html> to perform this experiment.

**AIM:** Multilayer Perceptron and Application.

### LABORATORY OUTCOMES:

- Students will be able to classify inputs using MLP network.
- Students will be able to observe effect of changing number of hidden layers and total no. of iterations by keeping constant momentum and learning rate.

### THEORY:

The Multi-Layer-Perceptron was first introduced by M. Minsky and S. Papert in 1969. It is an extended Perceptron and has one or more hidden neuron layers between its input and output layers. Due to its extended structure, a Multi-Layer-Perceptron is able to solve every logical operation, including the XOR problem.

Multilayer perceptrons (MLPs) are feedforward neural networks trained with the standard backpropagation algorithm. They are supervised networks so they require a desired response to be trained. They learn how to transform input data into a desired response, so they are widely used for pattern classification. With one or two hidden layers, they can approximate virtually any input-output map. They have been shown to approximate the performance of optimal statistical classifiers in difficult problems. Most neural network applications involve MLPs.

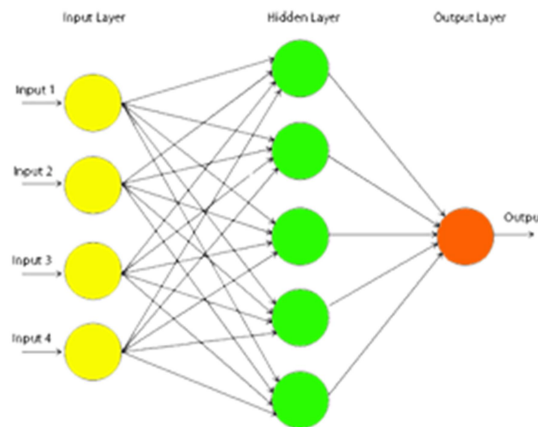


Fig. 1: Multi-layer Perceptron

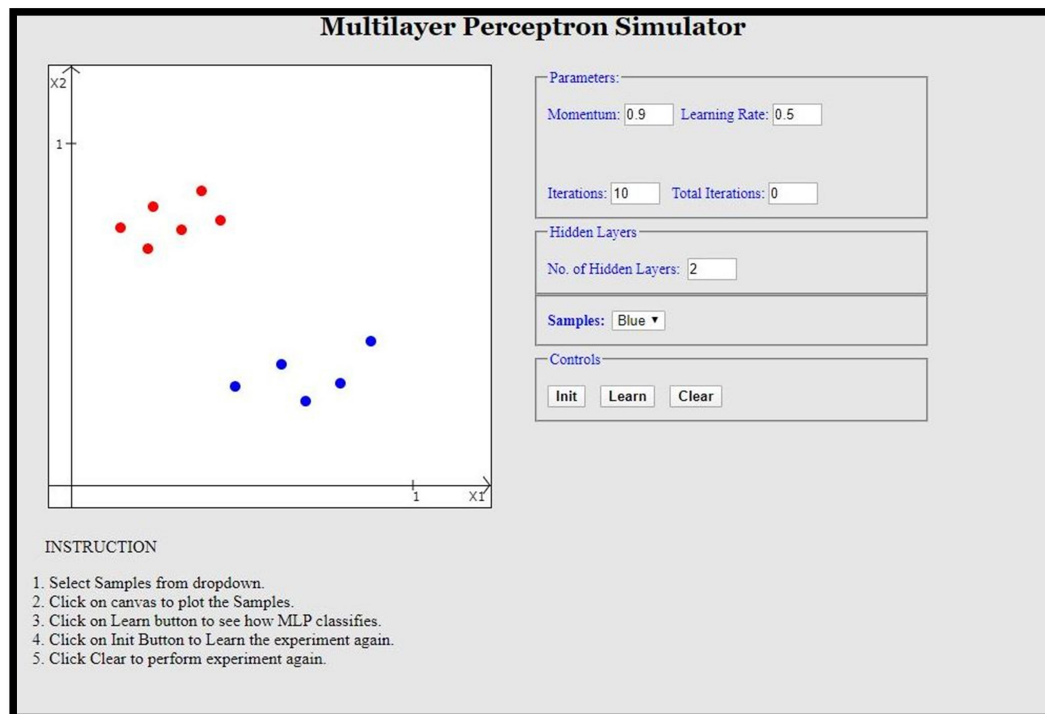
### Multilayer Perceptron Network:

- This network was introduced around 1986 with the advent of the back-propagation algorithm. Until then there was no rule via which we could train neural networks with more than one layer. As the name implies, a Multi-layer Perceptron is just that, a network that is comprised of many neurons, divided in layers. These layers are divided as follows: The input layer, where the input of the network goes. The number of neurons here depends on the number of inputs we want our network to get.
- One or more hidden layers. These layers come between the input and the output and their number can vary. The function that the hidden layer serves is to encode the input and map it to the output. It has been proven that a multi-layer perceptron with only one hidden layer can approximate any function that connects its input with its outputs if such a function exists.

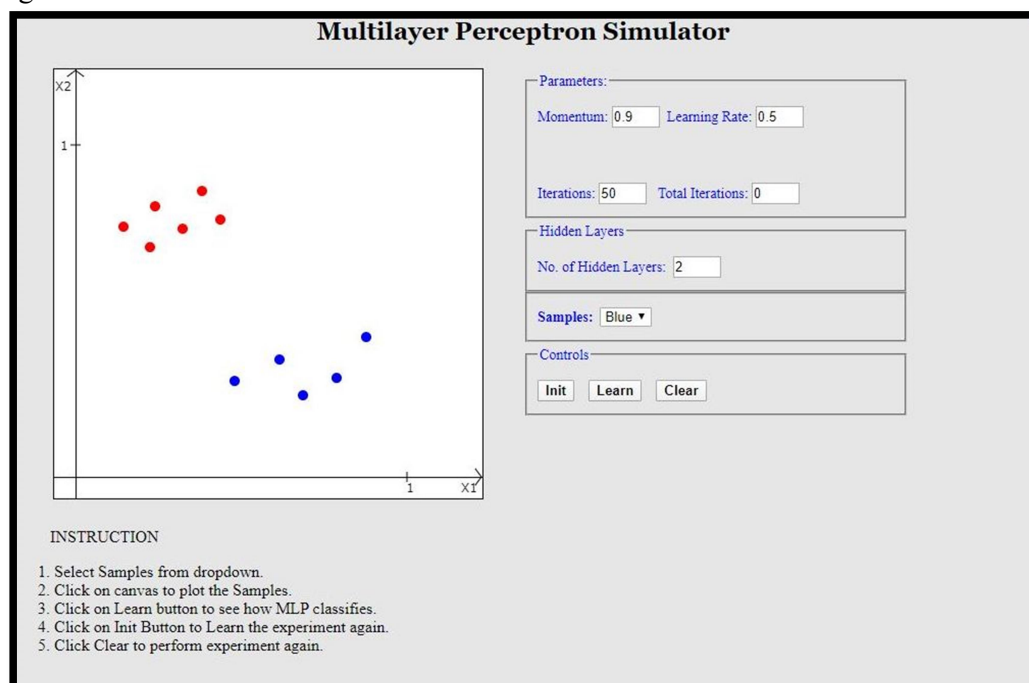
- The output layer, where the outcome of the network can be seen. The number of neurons here depends on the problem we want the neural net to learn.
- The Multi-layer perceptron differs from the simple perceptron in many ways. The same part is that of weight randomization. All weights are given random values between a certain range, usually  $[-0.5, 0.5]$ . Having that aside though, for each pattern that is fed to the network three passes over the net are made

### PROCEDURE:

1. Select Samples from dropdown and click on the board to plot samples.

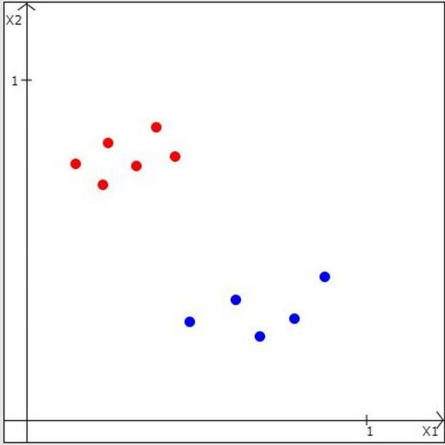


2. Change values in the Parameters section.



3. Input no. of Hidden Layers in the Hidden Layers section.

### Multilayer Perceptron Simulator



Parameters:

Momentum:  Learning Rate:

Iterations:  Total Iterations:

Hidden Layers

No. of Hidden Layers:

Samples:

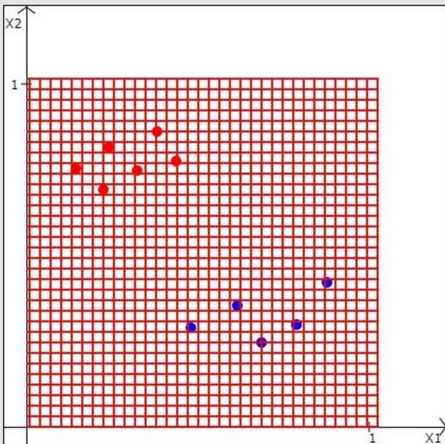
Controls

INSTRUCTION

1. Select Samples from dropdown.
2. Click on canvas to plot the Samples.
3. Click on Learn button to see how MLP classifies.
4. Click on Init Button to Learn the experiment again.
5. Click Clear to perform experiment again.

4. Click on Learn to see how MLP classifies the inputs you supplied.

### Multilayer Perceptron Simulator



Parameters:

Momentum:  Learning Rate:

Iterations:  Total Iterations:

Hidden Layers

No. of Hidden Layers:

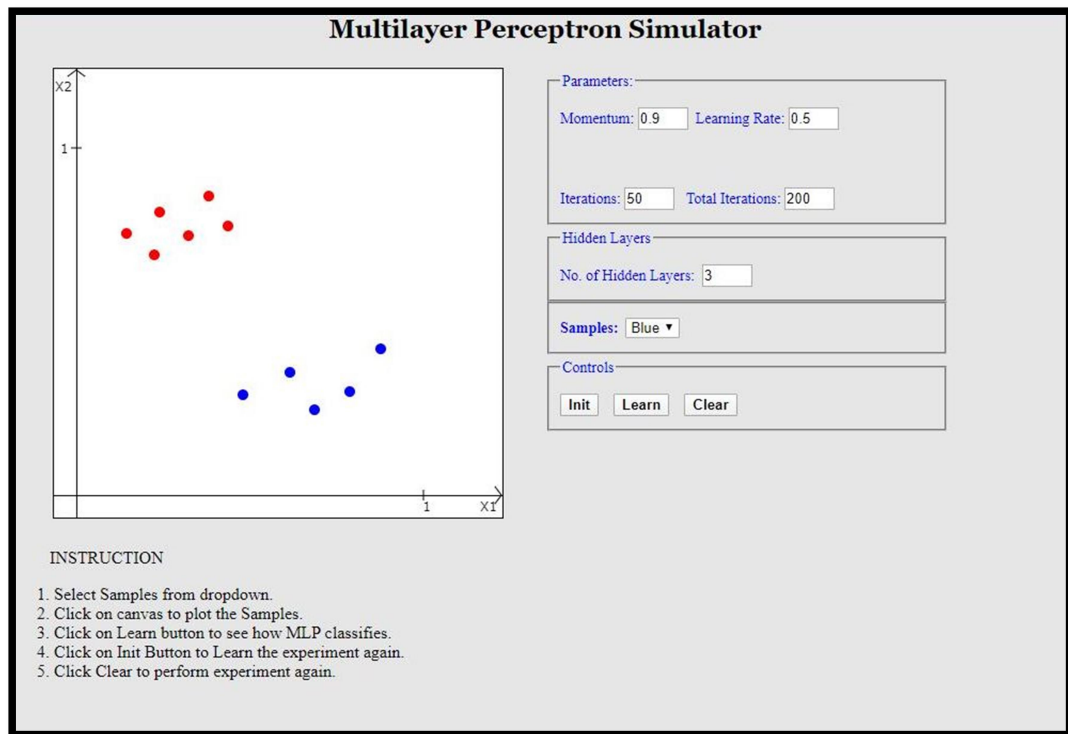
Samples:

Controls

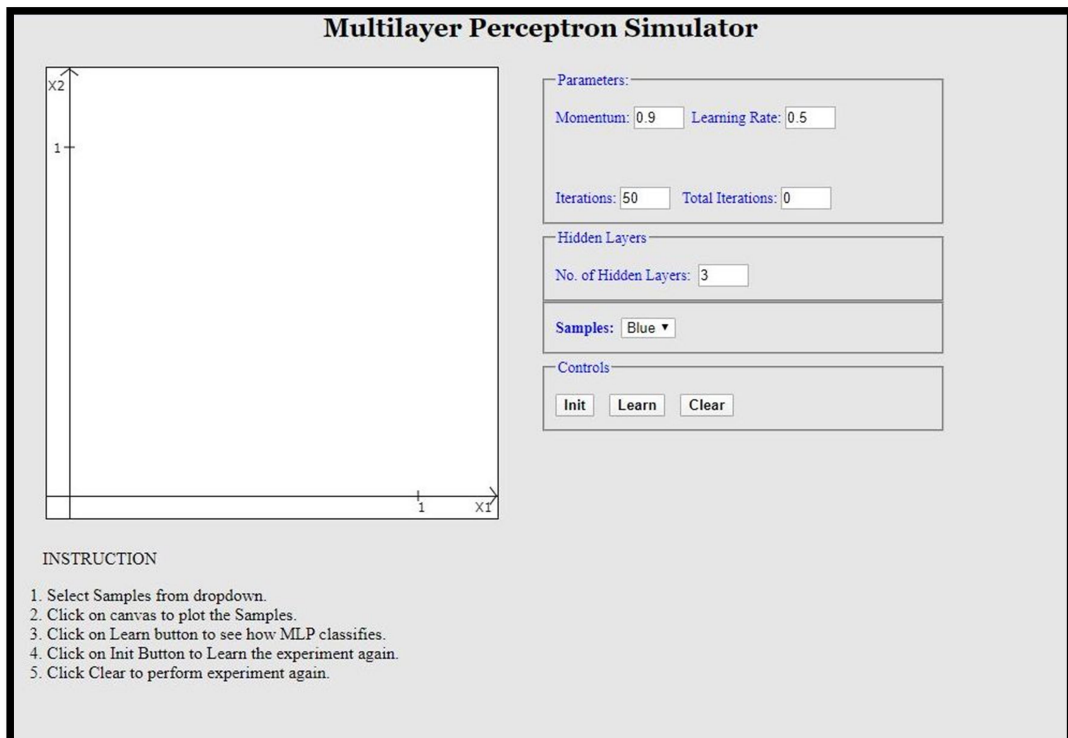
INSTRUCTION

1. Select Samples from dropdown.
2. Click on canvas to plot the Samples.
3. Click on Learn button to see how MLP classifies.
4. Click on Init Button to Learn the experiment again.
5. Click Clear to perform experiment again.

5. Click on Init Button to Restart the experiment from 1st Iteration.



6. Click on Clear Button to perform the experiment again.



## CONCLUSION:

MLP simulator used for classification of input data points. We observed the effect of changing number of hidden layers and total no. of iterations by keeping constant momentum and learning rate in MLP network.

## REFERENCES:

1. Rosenblatt, Frank. x. Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Spartan Books, Washington DC, 1961.
2. Rumelhart, David E., Geoffrey E. Hinton, and R. J. Williams. "Learning Internal Representations by Error Propagation". David E. Rumelhart, James L. McClelland, and the PDP research group. (editors), Parallel distributed processing: Explorations in the microstructure of cognition, Volume 1: Foundations. MIT Press, 1986.
3. Cybenko, G. 1989. Approximation by superpositions of a sigmoidal function Mathematics of Control, Signals, and Systems, 2(4), 303-314.
4. Haykin, Simon (1998). Neural Networks: A Comprehensive Foundation (2 ed.). Prentice Hall.ISBN 0-13-273350-1.
5. Neural networks. II. What are they and why is everybody so interested in them now?; Wasserman, P.D.; Schwartz, T.; Page(s): 10-15; IEEE Expert, 1988, Volume 3, Issue 1.
6. R. Collobert and S. Bengio (2004). Links between Perceptrons, MLPs and SVMs. Proc. Int'l Conf. on Machine Learning (ICML).

-----X-----X-----