|  |  |
| --- | --- |
| Subject: Machine Learning Lab | Course ID: CSL-604 |
| Semester: VI | Course: AI & DS |
| Laboratory: 406-B | Name of teacher: Prof. Seema Pawar |
| Name of Student: Ayush Gupta | Roll No: VU2S2324001 |

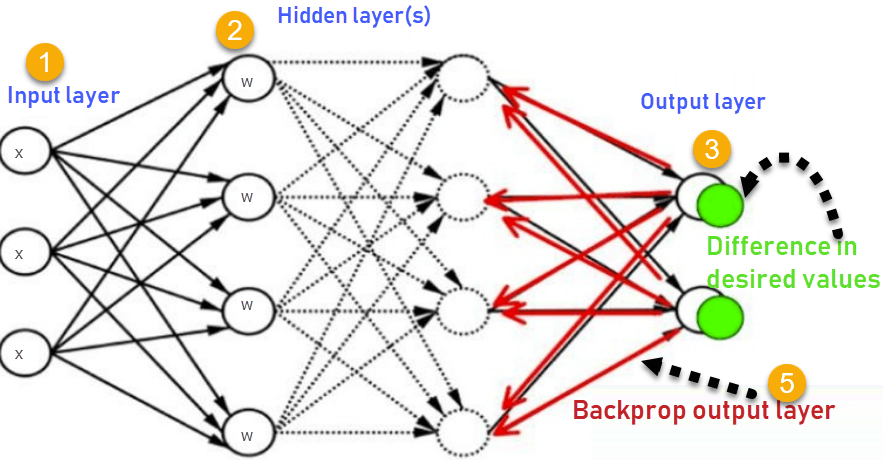
**EXPERIMENT NO. 8**

**Aim:**

To implement the Error Back Propagation Perceptron Training Algorithm

**Theory:**

Backpropagation is the fundamental technique used for training Artificial Neural Networks (ANNs). It is an optimization method that fine-tunes the weights of the network based on the error rate from the previous epoch (iteration). By adjusting these weights iteratively, the model minimizes errors and improves its generalization capability. The term "backpropagation" refers to the backward propagation of errors, which helps compute the gradient of a loss function with respect to the network’s weights.



**How Backpropagation Works:**

1. **Forward Propagation**: The input is passed through the network to generate an output.
2. **Error Calculation**: The difference between the actual output and the desired output is computed:

Error = Actual Output − Desired Output

1. **Backward Propagation**: The error is propagated backward through the network to adjust the weights.
2. **Weight Update**: Using gradient descent, the weights are updated to reduce the error.
3. **Repeat** until the error is minimized and the model converges.

**Advantages of Backpropagation:**

* **Fast and efficient** learning algorithm.
* **Automatically fine-tunes weights** to reduce errors.
* **Works well for complex functions** without needing prior knowledge.
* **Standard method** used in deep learning and neural networks.

**Disadvantages of Backpropagation:**

* **Prone to Overfitting**: If the network is too complex, it may memorize training data instead of generalizing well.
* **Slow Convergence**: Training deep networks requires many iterations and can be computationally expensive.
* **Vanishing Gradient Problem**: In deep networks, gradients can become very small, making weight updates ineffective.
* **Requires Large Datasets**: Backpropagation performs best with a significant amount of labelled data.

**Learning Objectives:**

* To understand and implement the Error Back Propagation Perceptron Training Algorithm.
* To apply gradient descent for optimizing neural networks.
* To develop a neural network that can correctly classify the XOR function.

**Conclusion / Learning Outcome:**

The Error Back Propagation Perceptron Training Algorithm was successfully implemented for learning an XOR Gate. The model trained by adjusting weights and biases using gradient descent, reducing the error iteratively over 10,000 epochs. After training, the network correctly classified the XOR function, which is not linearly separable and cannot be solved using a simple perceptron. This experiment demonstrated the effectiveness of multi-layer perceptrons (MLP) with backpropagation for solving complex problems, paving the way for deeper understanding of neural networks in machine learning.



**Program and Output:**

****