

- **Supervised Learning**

A teacher is present during learning process and presents expected output.

Every input pattern is used to train the network.

Learning process is based on comparison, between network's computed output and the correct expected output, generating "error". The "error" generated is used to change network parameters that result improved performance.

- **Unsupervised Learning**

No teacher is present.

The expected or desired output is not presented to the network.

The system learns of it own by discovering and adapting to the structural features in the input patterns.

- **Reinforced learning**

A teacher is present but does not present the expected or desired output but only indicated if the computed output is correct or incorrect.

The information provided helps the network in its learning process.

A reward is given for correct answer computed and a penalty for a wrong answer.

Note : The Supervised and Unsupervised learning methods are most popular forms of learning compared to Reinforced learning.

- **Hebbian Learning**

Hebb proposed a rule based on correlative weight adjustment. In this rule, the input-output pattern pairs  $(X_i, Y_i)$  are associated by the weight matrix  $W$ , known as correlation matrix computed as

$$W = \sum_{i=1}^n X_i Y_i^T$$

where  $Y_i^T$  is the transpose of the associated output vector  $Y_i$

There are many variations of this rule proposed by the other researchers

(Kosko, Anderson, Lippman)

- **Gradient descent Learning**

This is based on the minimization of errors  $E$  defined in terms of weights and the activation function of the network.

- Here, the activation function differentiable, because the of the updates network is of weight required to is dependent be on the gradient of the error  $E$ .

Note : The Hoffs Delta rule and Back-propagation learning rule are the examples of Gradient descent learning.

- **Competitive Learning**

In this method, those neurons which respond strongly to the input stimuli have their weights updated. When an input pattern is presented, all neurons in the layer compete, and the winning neuron undergoes weight adjustment . This strategy is called "winner-takes-all".

- **Stochastic Learning**

In this method the weights are adjusted in a probabilistic fashion.  
- Example : Simulated annealing which is a learning mechanism employed by Boltzmann and Cauchy machines.