MULTILAYER PERCEPTRON (MLP) / FFNN :-

→ Perceptron is limited to the classification of linearly separable patterns. Also, it was based on single linear neuron with adjustable weights, which limits the computing power of the algorithm.

→ ~~Thus, to overcome~~ Thus, to overcome the practical limitations of perceptron multilayer perceptrons were proposed.

⋆ Basic features of MLP :-

① Model of each neuron in the n/w includes a nonlinear activation function that is differentiable

② The n/w contains one or more layers that are hidden from both the i/p & o/p nodes.

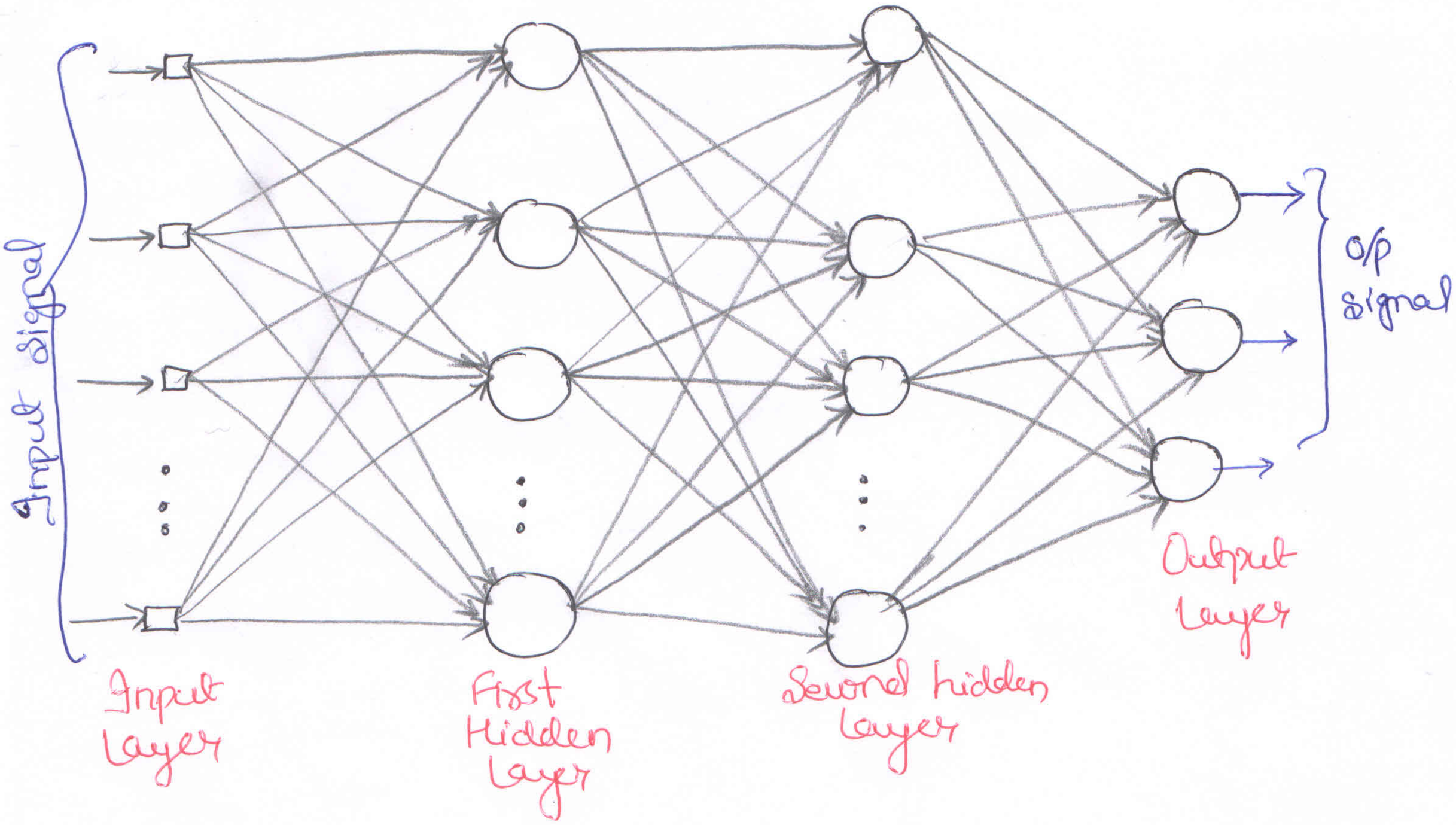
③ The n/w exhibits a high degree of connectivity, the extent of which is determined by synaptic weights of the n/w.

⋆ The training proceeds in two phases :-

① In forward phase, the synaptic weights of the n/w are fixed and the i/p signal is propagated through the n/w, layer by layer, until it reaches the o/p.
→ Thus, in this phase changes are confined to the activation potentials and o/ps of the neurons in the n/w

② In backward phase, an error signal is produced by comparing the o/p of the n/w with desired response. The resulting error signal is propagated through the n/w, again layer by layer, but in backward dir?

→ In the second phase, successive adjustments are made to the synaptic weights of the n/w. Calculation of the adjustments for the o/p layer is straightforward, but it is much more challenging for hidden layers.



⊛ Function signals:- A function signal is an i/p signal (stimulus) that comes in at the i/p end of the n/w, propagates forward (neuron by neuron) through the n/w & emerges at the o/p end of the n/w as an o/p signal.

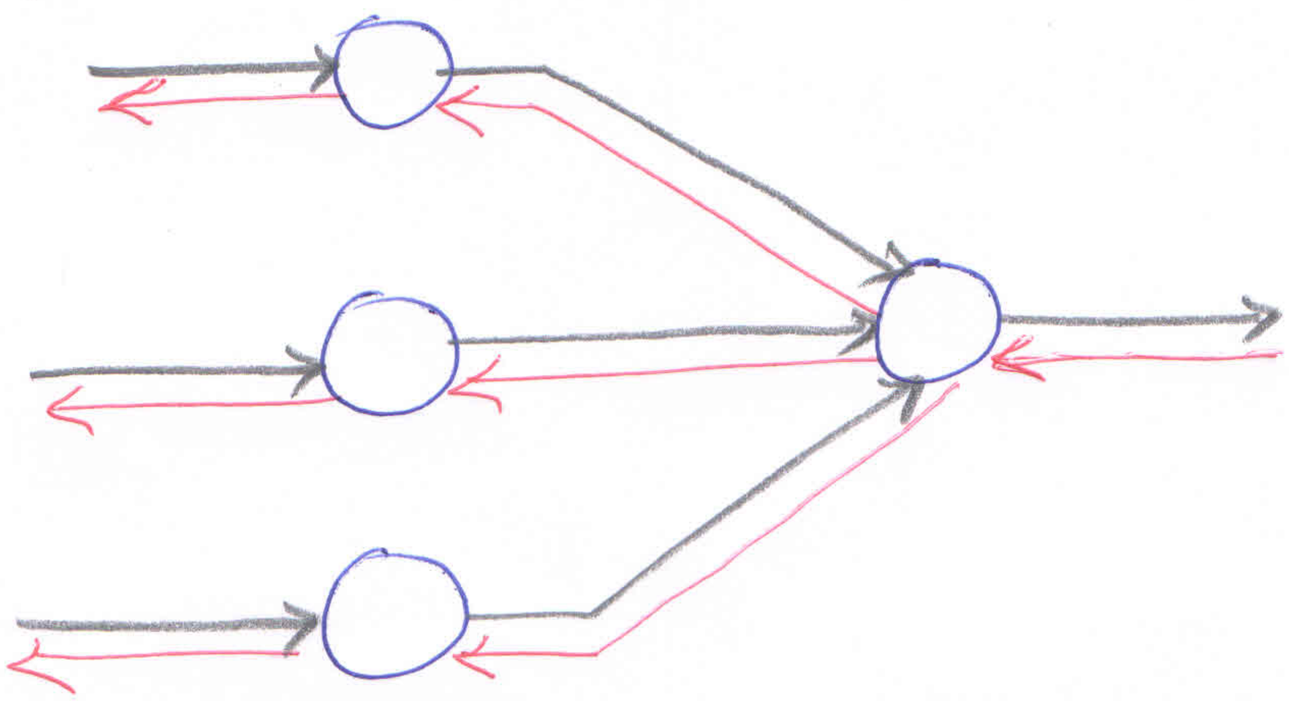
⊛ Error Signals:- An error signal originates at an o/p neuron of the n/w and propagates backward (layer by layer) through the n/w.

→ Its computation by every neuron of the n/w involves an error-dependent function in one form or another.

Each hidden or op neuron of MLP is designed to perform 2 computations:

① The computation of the function signal appearing at the o/p of each neuron, which is expressed as a continous non-linear function of the i/p signal & synaptic weights associated with that neuron.

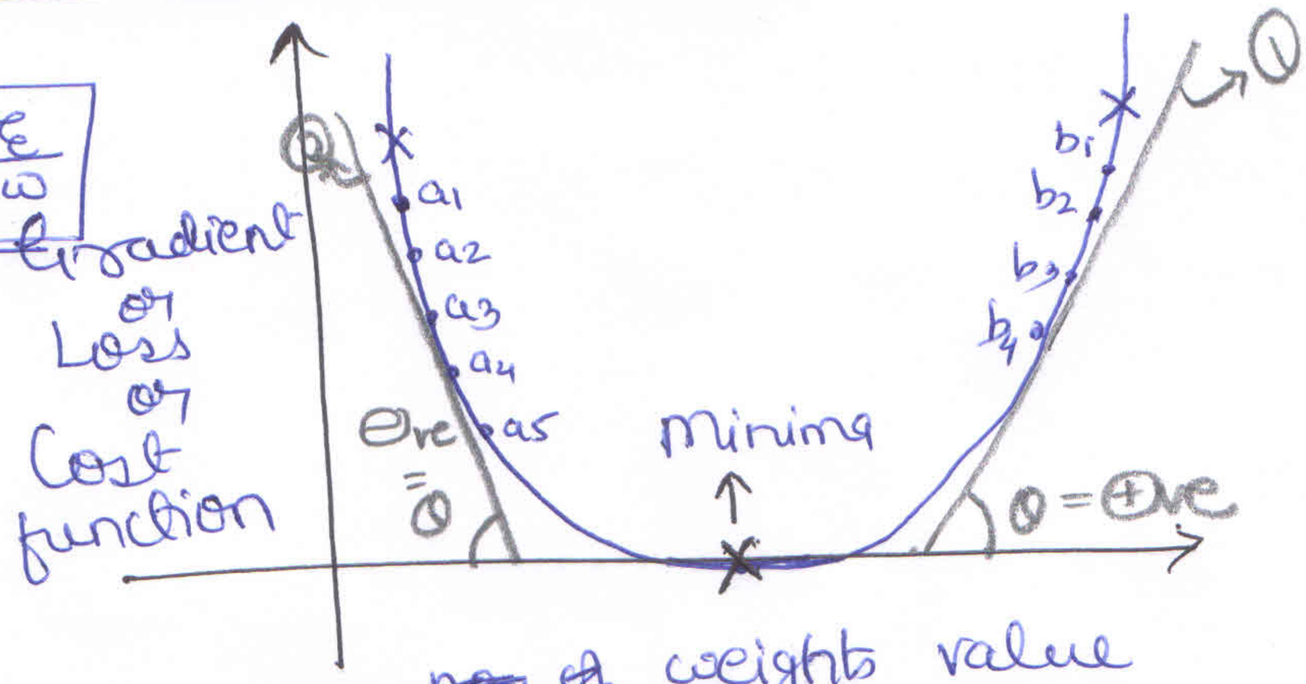
② The computation of an estimate of the gradient vector (ie. gradients of the error surface w.r.t. the weights connected to the i/ps of a neuron), which is needed for the backward pass through the n/w.



→ Function signal
← Error signal

Gradient Descent phenomenon:-

$$w_{new} = w_{old} - \eta \frac{\partial E}{\partial w}$$



* If θ = +ve, weights are to be subtracted from the previous value or are to be reduced.
* If θ = -ve etc, are to be incremented.
* Ultimate goal is to reach at minima point.
* η → is kept small for accurate convergence