Calculation of Error

1) for ~~the~~ rth neuron & calculated o/p 'O' & target o/p 'T'

∴ Error norm for rth neuron :-

$$E_r' = \frac{1}{2} e_r^2 = \frac{1}{2} (T-O)^2$$

↓

$\frac{1}{2}$ second norm of error in rth neuron (e_r) for the given training pattern.

2) Euclidean norm of error E' for first training pattern is —

$$E' = \frac{1}{2} \sum_{r=1}^{n} (T_{o_r} - O_{o_r})^2$$

E' = Error function in one training pattern.

* Squaring - To take only magnitudes & not +/-ve signs.

3) Using same technique for all training patterns —

$$E(V,W) = \sum_{j=1}^{nset} E^j(V,W,I)$$

$I \rightarrow$ I/P set

V_{lxm} = Weights (I/P $\longrightarrow$ Hidden)

W_{mxn} = Weights (Hidden $\rightarrow$ O/P)

4) $E \Rightarrow$ Error func^n depending on $m(l+n)$ weights of [W] & [V].

* This is Classic optimization problem.

↓

∴ An objective func^n / cost func^n $\rightarrow$ needs to be maximized / minimized w.r.t set of parameters.

* N/w parameters that optimize error func^n E over 'nset' pattern sets $[I^{nset}, t^{nset}]$ are $\rightarrow$ Synaptic weight values $[V]_{lxm}$ & $[W]_{mxn}$

Training of Neural Network

1) The synaptic weighting and aggregation operations performed by synapses & soma provide a 'similarity measure' between input vector I & synaptic weights [V] & [W].

2) When a new I/P pattern different from previously learned pattern, similarity b/w input & existing base knowledge → small,

∴, By changing synaptic weights → dist. b/w i/f & accumulated knowledge decreases.

3) NNs undergo 'learning procedures' and use 'learning rules' to determine connection strengths.

4) Supervised Learning → Error based learning algo. → Generate Error Signal → Modify Synaptic connection

5) Unsupervised & Competitive learning → Adjustment of Synaptic weights according to correlation of response of two neurons that adjoin it.