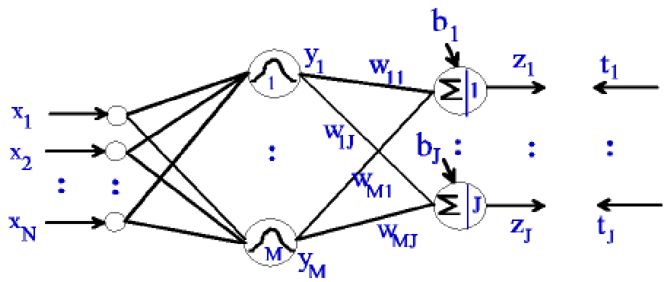
Derivation of the Steepest Descent Algorithm for Radial Basis Function Neural Networks



$$y_m^{(q)} = \exp[-\|x^{(q)} - c^{(m)}\|^2/(2\sigma_m^2)] \qquad (1)$$

$$E = \sum_{q=1}^{Q} \sum_{j=1}^{J} (z_j^{(q)} - t_j^{(q)})^2 \qquad (2)$$

$$z_j = \sum_{m=1}^{M} w_{mj} y_m + b_j \qquad (3)$$

The method of *steepest descent* from differential calculus is used for the derivation. To solve respectively for the weights $\{w_{mj}\}$, we use the standard formulation

$$w_{mj} \Leftarrow w_{mj} - \eta[\partial E/\partial w_{mj}]$$

where the factor η is the step size, or *learning rate*.

Upon using the equations below the diagram above, we derive the partial derivative. To keep the values small for less error build-up, we divide E by the factor QJ.

We take the partial derivative of E as given in Equation (2) above, except that we divided E by QJ to keep the values small.

$$\partial E/\partial w_{mj} = [\partial E/\partial z_j][\partial z_j^{(q)}/\partial w_{mj}] = [\partial/\partial z_j](1/QJ)\sum_{(q=1,Q)}(z_j^{(q)} - t_j^{(q)})^2 [\partial z_j^{(q)}/\partial w_{mj}] =$$

$$(2/QJ)\sum_{(q=1,Q)}(z_j^{(q)} - t_j^{(q)})[\partial z_j^{(q)}/\partial w_{mj}] = (2/QJ)\sum_{(q=1,Q)}(z_j^{(q)} - t_j^{(q)})[y_m]$$

Thus the steepest algorithm becomes

$$w_{mj} \Leftarrow w_{mj} - \eta\,(2/QJ)\sum_{(q=1,Q)}(z_j^{(q)} - t_j^{(q)})[y_m]$$