考虑单隐层 (p 个神经元) 的 NN. 设训练数据为

$$(x_{k,1}, x_{k,2}, \cdots, x_{k,n}; d_{k,1}, d_{k,2}, \cdots, d_{k,m}),$$

$$k = 1, 2, \cdots, N.$$

权重调整

$$w_{ij}^{0} \leftarrow w_{ij}^{0} + \Delta w_{ij}^{0}$$

$$w_{ij}^{1} \leftarrow w_{ij}^{1} + \Delta w_{ij}^{1}$$

随机初始化权重向量， 置

$$\Delta w_{ij}^{0} = 0, \quad i = 1, 2, \cdots, p; j = 0, 1, 2, \cdots, n$$

$$\Delta w_{ij}^{1} = 0, \quad i = 1, 2, \cdots, m; j = 0, 1, 2, \cdots, p$$

第 k 个样本被训练后， 隐层神经元的输出为

$$x_{k,i}^{1} = \sigma\left(\sum_{j=1}^{n} w_{ij}^{0} x_{k,j} + w_{i,0}^{0}\right), \quad i = 1, \cdots, p$$

NN 的输出为

$$y_{k,i} = \sum_{j=1}^{p} w_{ij}^{1} x_{ij}^{1} + w_{i0}^{1}, \quad i = 1, \cdots, m$$

改进误差函数为

$$E_k = \frac{1}{2}\sum_{i=1}^{m}[\lambda(d_{k,i} - y_{k,i})^2 + (1-\lambda)\Phi(d_{k,i} - y_{k,i})]$$

其中 $\Phi(x) = \ln(\cosh(\beta x))/\mu$, 而 β, μ 是常数.

权重调整公式:

$$\Delta w_{ij}^{1} \leftarrow -\alpha \frac{\partial E_k}{\partial w_{ij}^{1}} + \eta \Delta w_{ij}^{1} = \alpha c_i^{1} x_{k,j}^{1} + \eta \Delta w_{ij}^{1}$$

其中

$$c_i^{1} = \lambda(d_{k,i} - y_{k,i}) + (1-\lambda)\tanh(\beta(d_{k,i} - y_{k,i})), \quad x_{k,0}^{1} = 1$$

$$\Delta w_{ij}^{0} \leftarrow -\alpha \frac{\partial E_k}{\partial w_{ij}^{0}} + \eta \Delta w_{ij}^{0} = \alpha c_i^{0} x_{k,j} + \eta \Delta w_{ij}^{0}$$

其中 α, $\eta \in (0,1)$ 为常数, 而

$$c_i^{0} = \left(1 - (x_{k,i}^{1})^2\right) \sum_{s=1}^{m} c_s^{1} w_{si}^{1}, \quad x_{k,0} = 1$$

反向传播算法

算法 (反向传播算法)

Step 1. 初始化权向量 w, 选置 β, α, η, E_0, $\lambda = 1$ and $k = 0$.

Step 2. $k \leftarrow k + 1$.

Step 3. 调整权重 w.

Step 4. 计算误差 E_k.

Step 5. 如果 $k < N$, 到第二步.

Step 6. 置 $E = E_1 + E_2 + \cdots + E_N$.

Step 7. 如果 $E > E_0$, 那么 $k = 0$, $\lambda = \exp(-1/E^2)$ 到第二步.

Step 8. 结束.