Practical Course on Pattern Recognition Version: 063006.10
Sommer-Semester 2006

Prof. Dr. Stefan Posch
Dipl.Bioinform. André Gohr
(andre.gohr@informatik.uni-halle.de)

Institute of Computer Science
University Halle

Series 10

Exercise 10.1 (4 points)
Define the back-prop learning rule for a multilayer perceptron that also allows connections (edges between neurons) between non-adjacent layers. But all connections stay feed-forward.

Solution 10.1
Notation: ${}^kN_i$ denotes the ith neuron in layer k. Layer k has M^k neurons at all. $w_{{}^kN_i}({}^qN_j)$ denotes the weight of the edge ingoing into neuron ${}^kN_i$ outgoing from neuron ${}^qN_j$ if any. The multilayer perceptron consists of L layers. Layer L is the output-layer. $\mathcal{P}_{{}^kN_i}$ denotes the set of "predecessor" neurons of ${}^kN_i$ that are neurons having an outgoing edge being directed towards neuron ${}^kN_i$. $\mathcal{D}_{{}^kN_i}$ denotes the set of "direct descendant" neurons of ${}^kN_i$.
Further on $\sigma(h_{{}^kN_i}) = y_{{}^kN_i}$ denotes the answer of neuron ${}^kN_i$ being in activation state $h_{{}^kN_i}$. $x_{{}^kN_i}({}^qN_j) = y_{{}^qN_j}$ denotes the input of neuron ${}^kN_i$ coming from neuron ${}^qN_j$. The activation state $h_{{}^kN_i}$ of neuron ${}^kN_i$ in layer k is determined by all weighted inputs (weighted answers of "predeseccor" neurons):
$h_{{}^kN_i} = \sum\limits_{{}^dN_z \in \mathcal{P}_{{}^kN_i}} x_{{}^kN_i}({}^dN_z)\, w_{{}^kN_i}({}^dN_z) = \sum\limits_{{}^dN_z \in \mathcal{P}_{{}^kN_i}} y_{{}^dN_z}\, w_{{}^kN_i}({}^dN_z)$.
$\vec{y}_L = (y_{{}^LN_1}, \ldots, y_{{}^LN_{M^L}})$ is the vector of outputs of the perceptron. The error-function is denoted by $E(\vec{y}_L, \vec{t}, \boldsymbol{w}) = \sum\limits_{i=1}^{M^L} E(y_{{}^LN_i}, t_i, \boldsymbol{w})$. $\vec{t} = (t_1, \ldots, t_{M^L})$ is the vector of target outputs the perceptron should give for a given data-set. $\boldsymbol{w}$ denotes the set of all weights of the perceptron.

Weights of edges ingoing into any neuron $^{L}N_i$ of layer L:

$$\begin{aligned}
\triangle w_{^L N_i}(^k N_j) &= -\varepsilon \frac{\partial E(\vec{y}, \vec{t_L}, \boldsymbol{w})}{\partial w_{^L N_i}(^k N_j)} \qquad \text{with: } ^k N_j \in \mathcal{P}_{^L N_i} \\
&= -\varepsilon \sum_{r=1}^{M^L} \frac{\partial E(y_{^L N_r}, t_r, \boldsymbol{w})}{\partial w_{^L N_i}(^k N_j)} \\
&= -\varepsilon \sum_{r=1}^{M^L} \frac{\partial E(y_{^L N_r}, t_r, \boldsymbol{w})}{\partial y_{^L N_r}} \frac{\partial y_{^L N_r}}{\partial h_{^L N_r}} \frac{\partial h_{^L N_r}}{\partial w_{^L N_i}(^k N_j)} \\
&= -\varepsilon \sum_{r=1}^{M^L} \frac{\partial E(y_{^L N_r}, t_r, \boldsymbol{w})}{\partial y_{^L N_r}} \sigma'(h_{^L N_r}) \frac{\partial \sum_{^d N_z \in \mathcal{P}_{^L N_r}} y_{^d N_z} \, w_{^L N_r}(^d N_z)}{\partial w_{^L N_i}(^k N_j)}
\end{aligned}$$

$\frac{\partial \sum_{^d N_z \in \mathcal{P}_{L_{N_r}}} y_{^d N_z} \, w_{^k N_i}(^d N_z)}{\partial w_{L_{N_i}}(^k N_j)}$ is always equal to zero unless $^d N_z = {}^k N_j$ and $^L N_r = {}^L N_i$. Hence we get:

$$\begin{aligned}
\triangle w_{^L N_i}(^k N_j) &= -\varepsilon \frac{\partial E(y_{^L N_i}, t_i, \boldsymbol{w})}{\partial y_{^L N_i}} \sigma'(h_{^L N_i}) \, y_{^k N_j} \\
&= -\varepsilon \, \delta_{^L N_i} \, y_{^k N_j} \qquad (1)
\end{aligned}$$

with: $\delta_{^L N_i} = \frac{\partial E(y_{L_{N_i}}, t_i, \boldsymbol{w})}{\partial y_{L_{N_i}}} \sigma'(h_{^L N_i})$.

Weights of edges ingoing into any neuron $^{L-1}N_i$ of layer $L-1$:

$$\begin{aligned}
\triangle w_{^{L-1} N_i}(^k N_j) &= -\varepsilon \frac{\partial E(\vec{y}, \vec{t_L}, \boldsymbol{w})}{\partial w_{^{L-1} N_i}(^k N_j)} \qquad \text{with: } ^k N_j \in \mathcal{P}_{^{L-1} N_i} \\
&= -\varepsilon \sum_{r=1}^{M^L} \frac{\partial E(y_{^L N_r}, t_r, \boldsymbol{w})}{\partial y_{^L N_r}} \frac{\partial y_{^L N_r}}{\partial h_{^L N_r}} \frac{\partial h_{^L N_r}}{\partial w_{^{L-1} N_i}(^k N_j)} \\
&= -\varepsilon \sum_{r=1}^{M^L} \frac{\partial E(y_{^L N_r}, t_r, \boldsymbol{w})}{\partial y_{^L N_r}} \sigma'(h_{^L N_r}) \sum_{^d N_z \in \mathcal{P}_{L_{N_r}}} \frac{\partial y_{^d N_z} w_{^L N_r}(^d N_z)}{\partial y_{^d N_z}} \frac{\partial y_{^d N_z}}{\partial w_{^{L-1} N_i}(^k N_j)} \\
&= -\varepsilon \sum_{r=1}^{M^L} \delta_{^L N_r} \sum_{^d N_z \in \mathcal{P}_{L_{N_r}}} \frac{\partial y_{^d N_z} w_{^L N_r}(^d N_z)}{\partial y_{^d N_z}} \frac{\partial y_{^d N_z}}{\partial h_{^d N_z}} \frac{\partial h_{^d N_z}}{w_{^{L-1} N_i}(^k N_j)} \\
&= -\varepsilon \sum_{r=1}^{M^L} \delta_{^L N_r} \sum_{^d N_z \in \mathcal{P}_{L_{N_r}}} w_{^L N_r}(^d N_z) \, \sigma'(h_{^d N_z}) \sum_{^s N_t \in \mathcal{P}_{^d N_z}} \frac{\partial y_{^s N_t} w_{^d N_z}(^s N_t)}{w_{^{L-1} N_i}(^k N_j)}
\end{aligned}$$

$\frac{\partial y_{^s N_t} w_{d_{N_z}}(^s N_t)}{w_{L-1_{N_i}}(^k N_j)}$ is always equal to zero unless $^s N_t = {}^k N_j$ and $^d N_z = {}^{L-1} N_i$. Hence we get:

$$
\begin{aligned}
\triangle w_{^{L-1}N_i}(^kN_j) &= -\varepsilon \sum_{r=1}^{M^L} \delta_{^LN_r}\, w_{^LN_r}(^{L-1}N_i)\, \sigma'(h_{^{L-1}N_i})\, y_{^kN_j} \\
\triangle w_{^{L-1}N_i}(^kN_j) &= -\varepsilon\, y_{^kN_j}\, \sigma'(h_{^{L-1}N_i}) \sum_{r=1}^{M^L} \delta_{^LN_r}\, w_{^LN_r}(^{L-1}N_i)
\end{aligned}
$$

All outgoing edges of neuron $^{L-1}N_i$ are ingoing into neurons of layer L since layer L is the last layer. One may state this fact more general: All outgoing edges of neuron $^{L-1}N_i$ are ingoing into one neuron of $\mathcal{D}_{^{L-1}N_i}$.

$$
\boxed{
\begin{aligned}
\triangle w_{^{L-1}N_i}(^kN_j) &= -\varepsilon\, y_{^kN_j}\, \sigma'(h_{^{L-1}N_i}) \sum_{^zN_v \in \mathcal{D}_{^{L-1}N_i}} \delta_{^zN_v}\, w_{^zN_v}(^{L-1}N_i) \\
&= -\varepsilon\, y_{^kN_j}\, \sigma'(h_{^{L-1}N_i})\, \delta_{^{L-1}N_i}
\end{aligned}
} \tag{2}
$$

with: $\delta_{^{L-1}N_i} = \sum_{^zN_v \in \mathcal{D}_{^{L-1}N_i}} \delta_{^zN_v}\, w_{^zN_v}(^{L-1}N_i)$.

Weights of edges ingoing into any neuron ^mN_i of layer m in analogy to the previous derivation (especially equation (2)):

$$
\boxed{
\triangle w_{^mN_i}(^kN_j) = -\varepsilon\, y_{^kN_j}\, \sigma'(h_{^mN_i})\, \delta_{^mN_i} \qquad \text{with: } {^kN_j} \in \mathcal{P}_{^mN_i}
} \tag{3}
$$

with: $\delta_{^mN_i} = \sum_{^zN_v \in \mathcal{D}_{^mN_i}} \delta_{^zN_v}\, w_{^zN_v}(^mN_i)$.