Ludwig-Maximilians-Universität München 17.05.2013
Institut für Informatik
Prof. Dr. Volker Tresp
Tobias Emrich

Maschinelles Lernen und Data Mining

Sommersemester 2013

Übungsblatt 3

Besprechung des Übungsblattes am 23.05.2013

Aufgabe 3-1 Wdh. Vektor Calculus

Eine Funktion $g(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}$ über einen Vektor $\mathbf{x} \in \mathbb{R}^n$, die nach $\mathbf{x}$ abgeleitet wird ist:

$$\frac{\partial g(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}_1} \\ \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}_2} \\ \vdots \\ \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}_n} \end{bmatrix}.$$

Berechnen Sie $\frac{\partial \langle g(\mathbf{x})}{\partial \mathbf{x}}$ für:

a) $g(\mathbf{x}) = \sum_{i=1}^{n} \mathbf{x}_i$,

b) $g(\mathbf{x}) = \langle \mathbf{x}, \mathbf{x} \rangle$, das Standard-Skalarprodukt von $\mathbf{x}$ mit sich selbst,

c) $g(\mathbf{x}) = (\mathbf{x} - \mu)^2$ für $\mu \in \mathbb{R}^n$.

Aufgabe 3-2 Basisfunktionen von Neuronalen Netzen

Die Ausgabe eines neuronalen Netzes für einen Testvektor $\mathbf{x}_i$ ist definiert durch

$$f(\mathbf{x}_i) = \sum_{h=0}^{M_\phi - 1} w_h \phi_h(\mathbf{x}_i, \mathbf{v}_h).$$

Die Gewichte der einzelnen Neuronen können über die Backpropagation-Regel mit musterbasiertem Gradientenabstieg gelernt werden. In der Vorlesung wurden die neuronalen Netze mit sigmoiden Neuronen vorgestellt. Natürlich können auch andere Basisfunktionen verwendet werden.

a) Welche Eigenschaften müssen diese Basisfunktionen erfüllen?

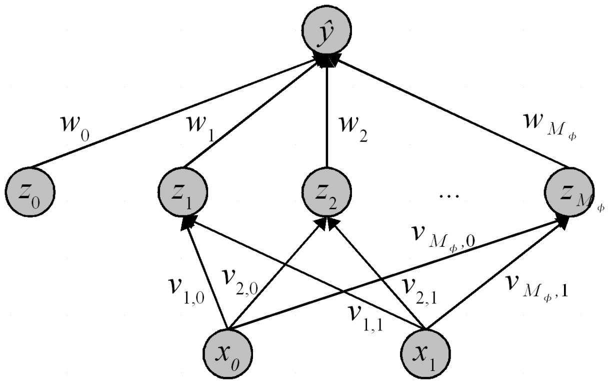
b) Ist eine Linearkombination $\phi(\mathbf{x}_i, \mathbf{v}_h) = z_h = \sum_{j=0}^{M} v_{h,j} x_{i,j}$ hierfür geeignet? Begründen Sie.

c) Ist die Anzahl der Parameter für $\phi(\mathbf{x}_i, \mathbf{v}_h)$ beschränkt? Können mehrere verschiedene Basisfunktionen für ein neuronales Netz verwendet werden?

b.w.

Aufgabe 3-3 Ein einfaches neuronales Netz

Unten abgebildet sehen Sie ein zweischichtiges neuronales Netz mit einem Eingabeneuron $x \in \mathbb{R}$ und je einem Biasneuron $x_0 = z_0 = 1$ (d.h. $\mathbf{x}_i = (1, x_{i,1})^T$) in der Eingabeschicht und der versteckten Schicht.



Als Aktivierungsfunktion der versteckten Neuronen verwenden wir einen Sigmoiden, also

$$z_h = \phi(\mathbf{x}_i, \mathbf{v}_h) = \frac{1}{1 + \exp\left(-\sum_{j=0}^{M} v_{h,j} x_{i,j}\right)} ,$$

das Ausgangsneuron $\hat{y}$ wird wie üblich über eine Linearkombination gebildet.

a) Zeigen Sie, dass gilt: $\frac{\partial z_h}{\partial v_{h,j}} = x_{i,j} \cdot z_h \cdot (1 - z_h)$

b) Drücken Sie den maximalen Wert von $\hat{y}$ in Abhängigkeit von $\mathbf{w}$ aus, wenn alle Ausgangsgewichte w_h ($h \in \{0, \ldots, M_\phi\}$) positiv sind. Was ist der minimale Wert?

c) Wie sieht $\hat{y}$ aus, wenn $v_{h,j} = 0$ für alle $j \in \{0, \ldots, M\}, h \in \{1, \ldots, M_\phi\}$? Welche Funktion erhalten Sie für $\hat{y}$, wenn alle $v_{h,j} = c, c \neq 0$?