

# Neuro- und Fuzzy-Methoden Übungsblatt 3

Jonas Jacobi, Felix Oppermann, Jan Geyken

9. November 2004

## Aufgabe 3.2: Perceptron

a)

Ein Perceptron kann eine Lernaufgabe genau dann lernen, wenn es die Aufgabe mit optimal gesetzten Gewichten berechnen kann. Dies folgt aus dem Konvergenztheorem (Minsky und Papert, „Perceptrons“). Beim einfachen Perceptron muss die Aufgabe linear separierbar sein um vom Perceptron gelöst und damit auch gelernt werden zu können. Bei zweidimensionalen Eingabevektoren heißt dies, dass der Eingaberaum durch eine Gerade zu separieren sein muss. In höheren Dimensionen muss es möglich sein den Eingaberaum durch eine Hyperebene zu separieren.

b)

Im folgenden soll der Lernvorgang für ein Perceptron, das die NOR-Funktion lernt angegeben werden. Nach dem Lernvorgang soll das Perceptron folgende Klassifizierung gelernt haben:

$$x^T \in \{(0, 0)\} \rightarrow \text{Klasse } A : y_d = 1 \quad x^T \in \{(0, 1), (1, 0), (1, 1)\} \rightarrow \text{Klasse } B : y_d = 0$$

Die Ausgabefunktion sei definiert durch:

$$f(\alpha) = \begin{cases} 1 & \text{falls } \alpha \geq 0 \\ 0 & \text{falls } \alpha < 0 \end{cases} \quad \text{mit } \alpha = \sum_{i=1}^N w_i x_i - \Theta$$

Wir setzen zunächst die Gewichte und den Schwellenwert auf null:

$$w_1 = 0 \text{ und } w_2 = 0$$

$$\Theta = 0$$

Die Lernrate soll eins betragen:

$$\eta = 1$$

Im Folgenden die Schritte des Lernalgorithmus:

**t = 0: Eingabe des Musters (0, 0)**

$$x^T = (0, 0)$$

$$y_d = 1$$

$$\rightarrow y = f(w_1 x_1 + w_2 x_2 - \Theta) = f(0 * 0 + 0 * 0 - 0) = f(0) = 1$$

Der Vektor wurde korrekt klassifiziert.

**t = 1: Eingabe des Musters (0, 1)**

$$x^T = (0, 1)$$

$$y_d = 0$$

$$\rightarrow y = f(w_1 x_1 + w_2 x_2 - \Theta) = f(0 * 0 + 0 * 1 - 0) = f(0) = 1$$

Der Vektor wurde falsch klassifiziert, es erfolgt eine Gewichtskorrektur.

$$w_1(1) = w_1(0) + \eta * (y_d - y) * x_1 = 0 + 1 * (0 - 1) * 0 = 0$$

$$w_2(1) = w_2(0) + \eta * (y_d - y) * x_2 = 0 + 1 * (0 - 1) * 1 = -1$$

**t = 2: Eingabe des Musters (1, 0)**

$$x^T = (1, 0)$$

$$y_d = 0$$

$$\rightarrow y = f(w_1 x_1 + w_2 x_2 - \Theta) = f(0 * 1 + -1 * 0 - 0) = f(0) = 1$$

Der Vektor wurde falsch klassifiziert, es erfolgt eine Gewichtskorrektur.

$$w_1(2) = w_1(1) + \eta * (y_d - y) * x_1 = 0 + 1 * (0 - 1) * 1 = -1$$

$$w_2(2) = w_2(1) + \eta * (y_d - y) * x_2 = -1 + 1 * (0 - 1) * 0 = -1$$

**t = 3: Eingabe des Musters (1, 1)**

$$x^T = (1, 1)$$

$$y_d = 0$$

$$\rightarrow y = f(w_1x_1 + w_2x_2 - \Theta) = f(-1 * 1 + -1 * 1 - 0) = f(-2) = 0$$

Der Vektor wurde korrekt klassifiziert.

(An diese Stelle ist der eigentliche Lernvorgang abgeschlossen. Ein weiterer Durchgang wird nur noch durchgeführt um festzustellen ob noch Gewichtsveränderungen auftreten.)

**t = 4: Erneute Eingabe des Musters (0, 0)**

$$x^T = (0, 0)$$

$$y_d = 1$$

$$\rightarrow y = f(w_1x_1 + w_2x_2 - \Theta) = f(-1 * 0 + -1 * 0 - 0) = f(0) = 1$$

Der Vektor wurde korrekt klassifiziert.

**t = 5: Erneute Eingabe des Musters (0, 1)**

$$x^T = (0, 1)$$

$$y_d = 0$$

$$\rightarrow y = f(w_1x_1 + w_2x_2 - \Theta) = f(-1 * 0 + -1 * 1 - 0) = f(-1) = 0$$

Der Vektor wurde korrekt klassifiziert.

**t = 6: Eingabe des Musters (1, 0)**

$$x^T = (1, 0)$$

$$y_d = 0$$

$$\rightarrow y = f(w_1x_1 + w_2x_2 - \Theta) = f(-1 * 1 + -1 * 0 - 0) = f(-1) = 0$$

Der Vektor wurde korrekt klassifiziert.

**t = 7: Eingabe des Musters (1, 1)**

$$x^T = (1, 1)$$

$$y_d = 0$$

$$\rightarrow y = f(w_1x_1 + w_2x_2 - \Theta) = f(-1 * 1 + -1 * 1 - 0) = f(-2) = 0$$

Der Vektor wurde korrekt klassifiziert.

Es traten in diesem Durchgang keine Veränderung der Gewichte mehr auf und der Algorithmus kann terminieren. Das Perzeptron hat die NOR-Funktion gelernt.