



Neuronale Netze, Sommersemester 2012

Übungsblatt 7

Abgabe: 28.06.2012, Besprechung: 05.07.2012

Aufgabe 15 Self-Organizing-Maps (SOMs) [10 Punkte]

Bei Selbstorganisierenden Karten handelt es sich um ein unüberwachtes Lernverfahren für Daten ohne Klassen-Labels. Deswegen ist es nicht möglich, anhand von Labels einen Trainings- oder Testfehler zu berechnen.

- (a) Wie kann man trotzdem messen, wie präzise eine Selbstorganisierende Karte die Eingangsdaten repräsentiert? Geben Sie eine mögliche Formel an und erklären Sie diese kurz! [2 Punkte]
- (b) Für die Qualität einer SOM ist es nicht nur wichtig, die Eingangsdaten möglichst gut zu repräsentieren. Welcher weitere Fehler sollte ebenfalls betrachtet werden und warum? (Tipp: Bedenken Sie die Nachbarschaftsbeziehungen der Codebookvektoren.) [2 Punkte]
- (c) SOMs werden häufig zur explorativen Datenanalyse eingesetzt, d.h. man trainiert die SOM auf einer Menge von ungelabelten Daten $D = \{\mathbf{x}_i\}_{i=1}^N$, um Wissen über die Struktur und die Verteilung $p(\mathbf{x})$ der Daten zu erhalten. Angenommen, man hätte nun für die Hälfte der Daten auch zugehörige Klassen-Labels. Erläutern Sie, wie man eine SOM benutzen könnte, um den restlichen Mustern \mathbf{x}_i ohne Klassen-Labels Klassenzugehörigkeiten zuzuweisen. [2 Punkte]
- (d) Geben seien vier Codebookvektoren: $A = (0, 0)$, $B = (1, 0)$, $C = (1, 1)$ und $D = (0, 1)$, die eine A-B-C-D Kette einer eindimensionalen SOM bilden. Als Ähnlichkeitsmaß wird der euklidische Abstand verwendet. Die Nachbarschaft sei definiert durch

$$h_{cj}(t) = \begin{cases} 1 & \text{falls } j = c \\ 0.5 & \text{falls } |j - c| = 1 \\ 0 & \text{sonst} \end{cases}$$

Die Lernrate sei 0.5. Wie lauten die Gewichte der vier Neuronen nach einem Lernschritt bei Eingabe von $X_1 = (0.4, 0.6)$? Welche Gewichte hätten sich bei der Eingabe von $X_2 = (0, 0)$ ergeben? Geben Sie jeweils das Gewinnerneuron, die Gewichtsänderung und die neuen Gewichte an. [4 Punkte]

Aufgabe 16 Implementierung einer SOM (10 Punkte)

Laden Sie sich die Datei `som.zip` von der Übungsseite herunter und entpacken Sie sie in ein separates Verzeichnis.

- (a) Vervollständigen sie die mit TODO gekennzeichneten Codeabschnitte in den Methoden `getGaussHcj` und `getEuclideanDistance`. [3 Punkte]

- (b) Vervollständigen sie die Methode *decayEta*, welche den Lernfaktor abfallen lässt, in den mit TODO gekennzeichneten Abschnitten. [2 Punkte]
- (c) Implementieren Sie die Methoden *findNearestNeuron* und *processPattern* in den mit TODO gekennzeichneten Abschnitten. [3 Punkte]
- (d) Führen Sie das vollständige Programm mit der Eingabedatei *SOMinput.dat* aus, welche Eingabemuster aus einem zweidimensionalen Raum enthält. Verwenden sie hierfür eine Karte mit 5x4 Neuronen. Fügen sie der Abgabe die Ausgabedatei *output-5-4.txt* bei. [2 Punkte]