moresun.org

Name:

Matrikelnummer: StKz:

Beschriften Sie bitte alle abgegebenen Blätter mit Name und Matrikelnr.!

Anzahl der Angabeblätter: **6**

Anzahl zusätzlich abgegebener Blätter:

**Vorlesungsprüfung Neural Computation**
30.1.2008

1. Perzeptron (*perceptron*): Beschreiben Sie den *online*-Perzeptron-Trainingsalgorithmus. Welche Voraussetzungen müssen gelten, damit das Verfahren konvergiert (terminiert)? (7Pkte)

2. *Multilayer-Perceptron*: Beschreiben Sie Aufbau und Funktionsweise des *Multilayer-Perceptron* (MLP). Erklären Sie anhand der Abbildung 1 wie das MLP mit dem Backpropagation-Lernalgorithmus trainiert wird. Verwenden Sie dabei die Summe der quadratischen Abweichungen (Sum of Squared Errors - SSE) als Kostenfunktion und die gleiche Aktivierungsfunktion

$$g(a) = \frac{1}{1 + e^{-a}}$$

für alle 3 Neuronen. Dabei gilt: $g(a)' = g(a)(1 - g(a))$. (12Pkte)

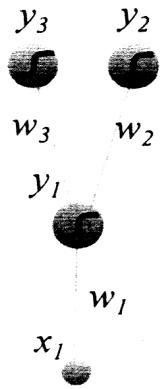


Abbildung 1: Ein einfaches MLP mit 2 Ausgabeeinheiten.

3. lineare Regression/lineare Neuronen (insgesamt 25 Punkte)

Gegeben sind Eingabedaten (skalare Inputs) $x_i$ mit dazugehörigen Ausgabedaten (skalare Targets) $y_i$.

$$x_1 = -2, x_2 = -1, x_3 = 0, x_4 = 1, x_5 = 2 \tag{1}$$
$$y_1 = -1, y_2 = -1, y_3 = 0, y_4 = 2, y_5 = 1 \tag{2}$$

(a) (5Pkte) Zeichnen Sie die Trainingsdaten als 5 Datenpunkte in einem Koordinatensystem ein, bei dem Eingabewerte auf der Abszisse und Ausgabewerte auf der Ordinate dargestellt sind. Zeichnen Sie (durch Schätzen) die Ausgleichsgerade ein (Regressionsgerade).

(b) (8Pkte) Rechnen Sie die aus den Trainingsdaten geschätzten Gewichte $w_0$ (*bias*) und $w$ (=Anstieg der Regressionsgeraden) einer linearen Einheit aus. *Tipp*: Ergänzen Sie die Eingabewerte zunächst um eine weitere Komponente mit dem Wert 1 (homogene Koordinaten). Sie erhalten 5 Eingabevektoren bestehend aus je 2 Komponenten. Rechnen Sie danach (in geschlossener Form) mit Hilfe der Pseudo-Inversen (ist in diesem Fall leicht zu berechnen) der $2 \times 5$ Eingabedatenmatrix und der 5 Ausgabewerte die 2 Gewichte (Regressionsparameter) aus.

(c) (7Pkte) Führen Sie rechnerisch mindestens zwei Schritte der LMS Lernregel durch. Initialisieren Sie zuvor $w_0$ und $w$ jeweils mit 0 und wählen Sie die Lernrate 0.1. Wählen Sie dabei dazu zufällig 2 Datenpunkte aus.

(d) (5Pkte) Erklären Sie das Prinzip der (*online*) LMS Lernregel und den Zusammenhang mit dem Gradientenabstiegsverfahren. Welchen Einfluss hat die Lernrate?

4. Optimierungsverfahren (insgesamt 8 Punkte)

   (a) (5Pkte) Welche iterativen (nicht-linearen) Optimierungsverfahren kennen Sie, welche Vor- und Nachteile haben diese und wo werden sie im Bereich der Neuronalen Netze eingesetzt (nennen Sie mindestens 2 Verfahren).

   (b) (3Pkte) Bei welchen Arten von Fehlerfunktionen gibt es eine Lösung in geschlossener Form?

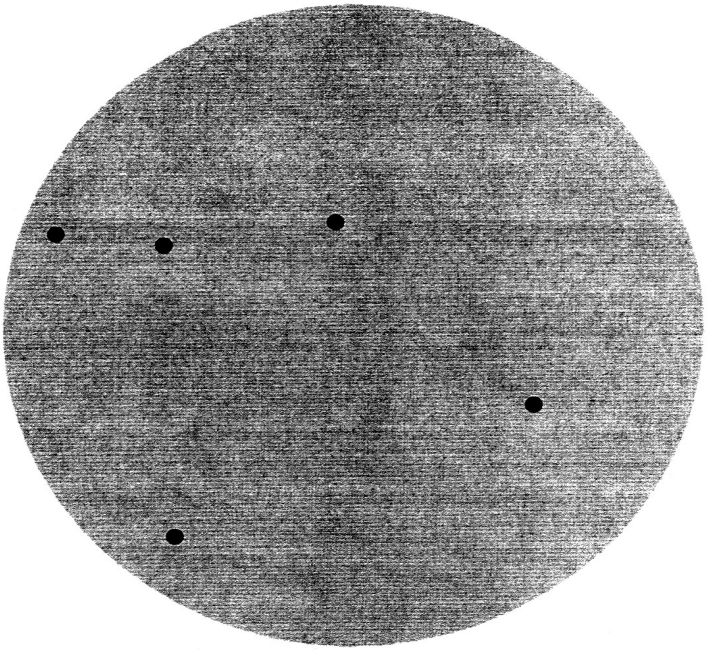

Abbildung 2: Referenzvektoren (*center*) eines Netzwerkes dargestellt als Punkte im Eingabesignalraum.

5. Wettbewerbslernen (insgesamt 13 Punkte)

   (a) (5Pkte) In Abbildung 2 ist eine Menge von Referenzvektoren (d.h. Stützpunkte der Einheiten eines unüberwachten Netzwerkes) im 2D Eingabesignalraum eingezeichnet. Eingabevektoren seien innerhalb der kreisförmigen Region gleichverteilt. Zeichnen Sie die Grenzen der Voronoi-Regionen der Referenzvektoren ein. Zeichnen Sie Positionen der Referenzvektoren nach einem Schritt des (*batch*) LBG-Verfahrens (Linde, Buzo, Gray). Gehen Sie dabei von einer großen Trainingsmenge aus.

   (b) (8Pkte) Erläutern Sie den Begriff "Wettbewerbslernen"(*competitive learning*). Beschreiben Sie ein *Online*-Vektor Quantisierungsverfahren, das auf Wettbewerbslernen basiert.

6. (15 Pkte) Beschreiben Sie das Verfahren der Hauptkomponentenanalyse (*Principal Component Analysis*). Wofür wird dieses Verfahren eingesetzt? Welche Lernregeln gibt es um die Hauptkomponenten in einem *Online*-Verfahren (Neuronalen Netz) zu bestimmen?

7. Support Vector Machines (SVM) (insgesamt 20 Punkte)

Gegeben sind folgende 4 linear separierbare Datenpunkte aus 2 Klassen:

$$\begin{array}{lll} \omega_1: & \mathbf{x}_1 = (1,1)^T & \mathbf{x}_2 = (0,0)^T \\ \omega_2: & \mathbf{x}_3 = (0,1)^T & \mathbf{x}_4 = (-1,1)^T \end{array} \tag{3}$$

mit Klassenlabels $t_1 = -1, t_2 = -1, t_3 = 1, t_4 = 1$. Gehen Sie bei Aufgabe (a) und (b) davon aus, dass KEINE *Kernel*-Funktionen benutzt werden, d.h. nur innere Produkte (die Elemente der Kernel Matrix $k_{ij} = \mathbf{x}_i^T \mathbf{x}_j$).

(a) Skizzieren Sie die 4 Datenpunkte. Welche sind die Supportvektoren? Ergänzen Sie zu der Punktemenge (durch Eintragen bei $\omega_1$ bzw. $\omega_2$) mindestens zwei weitere Datenpunkte, so dass die Menge der Support Vektoren unverändert bleibt (5Pkte)

(b) Zeichnen Sie (durch Schätzen) den Gewichtsvektor der optimalen "Hyperebene" (*optimal separating hyperplane*) und den optimalen *margin*. Erklären Sie den Zusammenhang zwischen *margin*, Gewichtsvektor und Generalisierung. (5Pkte)

(c) Was ist der Unterschied zwischen den Lernprinzipien "Empirical Risk Minimization" (ERM) und "Structural Risk Minimization" (SRM)? Welches Lernprinzip wird von einer SVM benutzt (5 Pkte).

(d) Was versteht man unter dem *dualen Optimierungsproblem* und welche Rolle spielen *Kernel*-Funktionen bei SVM (Wie "funktioniert" der *Kernel-trick*?). (10Pkte)