MatrikelNr:

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Prof. Dr. R. Brause

Musterlösung zur

Klausur zur Vorlesung Adaptive Systeme Wintersemester 2009/2010

Datum: 23.02.2010

Vorname:
Name:
Matrikelnummer:
Geburtsdatum:
Studiengang:

Als BSc bearbeiten Sie bitte den Teil der Aufgaben, der mit „AS-1“ gekennzeichnet ist. Als MSc bearbeiten Sie den „AS-1“ und/oder den „AS-2“-Teil. Im AS-1-Teil können 37 Punkte in insgesamt vier Aufgaben erreicht werden. Im AS-2-Teil aus insgesamt fünf Aufgaben sind es 45 Punkte.

Durch die Übungspunkte können maximal 10% der Klausurleistung erbracht werden. Als Hilfsmittel ist ein Taschenrechner erlaubt.

Viel Erfolg!

Wird vom Prüfer ausgefüllt:

1	2	3	4	5	6	7	8	9	Σ
/10	/12	/5	/10	/10	/10	/6	/6	/13	

Punkte Klausur:
Punkte Übungen:
Punkte Gesamt:
Note:

MatrikelNr:

AS-1.1 Wichtige Definitionen 10P

Definieren Sie folgende Begriffe mit kurzen Stichworten

a) formales Neuron
b) binäres Neuron
c) Online-Lernverfahren
d) Offline-Lernverfahren
e) überwachtes Lernverfahren
f) unüberwachtes Lernverfahren
g) Overfitting
h) Fluch der Dimensionen
i) Klassenprototyp
j) Multi-Layer-Perzeptron

a) ein Tupel aus Eingabemenge, Ausgabemenge, Gewichtemenge, Transformationsabbildung (Aktivierung und Ausgabefunktion) und einer Lernfunktion.

b) formales Neuron mit binärer Ausgabefunktion, etwa S(z) aus {0,1}

c) Beim Online-Lernen ist die Trainingsmenge unbekannt; die Trainingsmuster erscheinen sukzessive und werden nicht gespeichert.

d) Beim Offline-Lernen ist die Trainingsmenge endlich und bekannt und kann mehrmals verwendet werden.

e) Beim überwachten Lernen liegt für jedes Eingabemuster auch eine Bewertung eines Lehrers vor. Die Verarbeitung (Lernen) erfolgt gemäß einer Zielfunktion.

f) Beim unüberwachten Lernen ist die Bewertung der Eingabemuster unbekannt. Die Befolgung einer Lernregel führt zu einem Ziel, was nicht vorgegeben ist.

g) Overfitting (Überanpassung) tritt auf, wenn die zufälligen Abweichungen der Trainingsmenge vom gewünschten Ergebnis genau gelernt werden und mit der dann fehlenden Generalisierung zukünftige Testmuster falsch behandelt werden.

h) Gibt es bei nur wenigen Beispielen zu viele Merkmale pro Beispiel, so ist eine Konvergenz der Lernverfahren schwer möglich, da die Anzahl der Beispiel für die Bestimmung der benötigten Parameter nicht ausreicht. Dies nennt man auch den „Fluch der Dimensionen“.

i) Ein Klassenprototyp ist entweder ein einzelnes, ausgezeichnetes Muster einer Klasse (Mustermenge), was als „typisch“ für die Klasse betrachtet wird, oder eine rechnerische Größe, etwa der Mittelwert aller Muster einer Klasse.

j) Ein neuronales Netz, bestehend aus mehreren Schichten, wird ein Multi-layer-Perzeptron genannt, wenn die Ausgabefunktionen sigmoidale oder binäre Funktionen sind. Dabei wird ignoriert, dass das eigentliche Perzeptron nur binäre Ausgaben kennt.

MatrikelNr:

AS-1.2 Perzeptron-Lernen 12P

Folgend sind drei Mustervektoren X_i vorgegeben. Führen Sie auf diesen Mustern eine Trennung der Klassen A (Ausgabe = 0) und B (Ausgabe = 1) mit Hilfe einer Ihnen bekannten Perzeptronlernregel aus.

Die Anfangsgewichte seien **w**(0) = (-2, -2) mit dem Schwellwert s(0)=26 und γ=0,08. Die Reihenfolge, mit der die Muster gelernt werden, entspricht dem Index des Musters.

a) Schreiben Sie für jeden Lernschritt *t=1,2,3* die Werte der Ein-, Ausgabe und Gewichte hin.
b) Zeichnen Sie nach jedem Lernschritt im Diagramm die Lage der Geraden der aktuellen Klassentrennung des Perzeptrons ein.

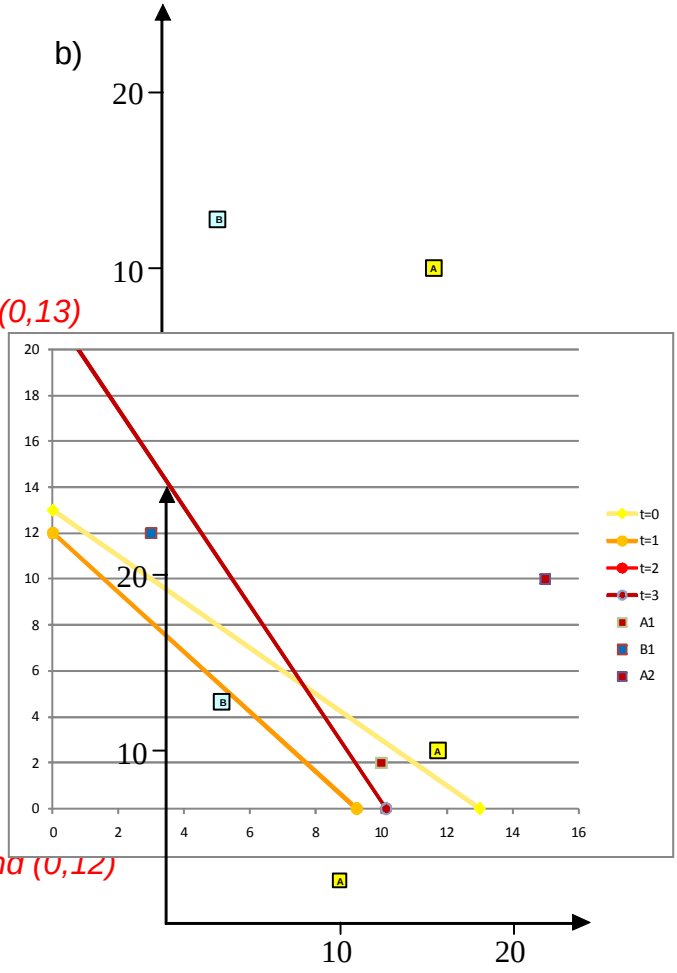
$X_1(A) = (10, 2)^T$; $X_2(B) = (3, 12)^T$; $X_3(A) = (15, 10)^T$

Schritt1: a) b)

x1 = 10, x2 = 2, x3 = 1
z= w1x1+w2x2+w3x3 = 2, also y = 1, L(A)=0

w**(1) = **w**(0) + g(L-y)**x
w1(1) = -2,8, w2(1) = -2,16, w3(1) = 25,92

Achsenschnittpunkte der Geraden bei (13,0) und (0,13)



Schritt2: a)

x1 = 3, x2 = 12, x3 = 1
z= w1x1+w2x2+w3x3 = -8,4, also y = 0, L(B)=1

w**(2) = **w**(1) + g(L-y)**x
w1(2) = -2,56, w2(2) = -1,2, w3(2) = 26

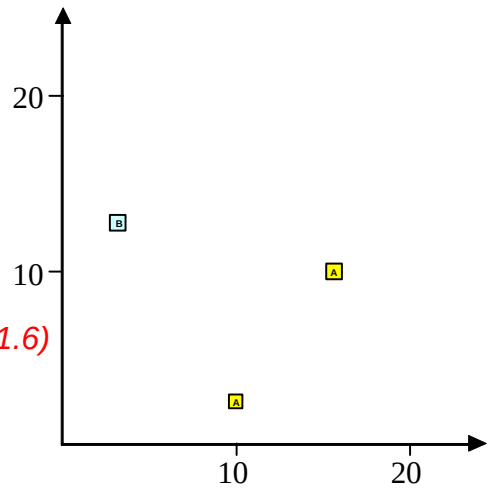
Achsenschnittpunkte der Geraden bei (9.26, 0) und (0,12)

Schritt3: a) b)

x1 = 15, x2 = 10, x3 = 1
z= w1x1+w2x2+w3x3 = -24,4, also y = 0, L(A)=0

***w**(3) = **w**(2)*
w1(3) = -2,56, w2(3) = -1,2, w3(3) = 26

Achsenschnittpunkte der Geraden bei (10.1, 0) und (0, 21.6)



AS-1.3 Korrelation **5P**

a) Nennen Sie die Oja-Lernregel. (2P)

$w(t+1) = w(t) + \gamma(t)\, y(t)\, (x(t)-y(t)w(t))$

b) Benennen Sie das Konvergenzziel der Oja-Lernregel. (1P)

Das Konvergenzziel ist der Eigenvektor der Autokorrelationsmatrix mit dem größten Eigenwert.

c) Berechnen Sie für die in Aufgabe AS-1.2 angegebenen drei Muster die Autokorrelationsmatrix. (4P)

$A_{ij} = \langle x_i x_j \rangle,\ i,j = 1,2$ *$A12 = A21 = (20+36+150)/3 = 69$*
$A11 = (100+9+225)/3 = 111,$ *$A22 = (4+144+100)/3 = 83$*

AS-1.4 Self-Organizing-Maps (SOM) **10P**

a) Wie lautet die Auswahlregel für das Gewinnerneuron? (2P)

Wähle das Neuron mit dem kleinsten Abstand seines Gewichtsvektors zum Eingabevektor

b) Geben Sie die Lernregel für SOM an. (2P)

$w_k(t+1) = w_k(t) + \gamma(t)\, h(t,c,k)\, [\, (x(t)-w_k(t)]$
mit $h(t,c,k)=1$ für alle Nachbarneuronen k von c, sonst null.
Die Nachbarschaft in $h(t)$ nimmt mit der Zeit ab.

c) Was sind die typischen Unterschiede zwischen Eingabe- und Ausgaberaum? (3P)

Die Anzahl der Dimensionen des Eingaberaums entspricht der Diensionszahl der Eingabevektoren, während im Ausgaberaum die Anzahl der direkten Nachbarn entscheidend ist. Typischerweise ist der Eingaberaum höherdimensional als der Ausgaberaum. Beide Dimensionen sind unabhängig von einander.

d) Begründen Sie, warum das Netz besser konvergiert, wenn beim Training Nachbarschaftsverhältnisse mit berücksichtigt werden. (3P)

Beim Training ist die Anzahl der Iterationen für einen Parameter (ein Gewicht) ausschlaggebend. Wird die Nachbarschaft mittrainiert, so erhöht sich die Anzahl der Iterationen pro Gewicht bei gleichbleibender Anzahl von Trainingsbeispielen.

Ende des Teils AS-1

Beginn Teil AS-2

AS-2.5 XOR-Problem **10P**

Geben Sie die Werte für die Gewichte für ein zweischichtiges neuronales Netz an, welches die boolsche Funktion XOR implementiert. Diese ist für zwei Eingaben x_1 und x_2 folgendermaßen definiert: XOR(x_1,x_2) = $x_1\bar{x}_2 + \bar{x}_1 x_2$

Im *hidden Layer* sollen zwei binäre Neuronen mit S(z) = $\begin{cases} 1 & z > 0{,}5 \\ 0 & z \leq 0{,}5 \end{cases}$ zum Einsatz kommen.

Die Ausgabe wird von einem linearen Neuron erzeugt.

Die Aktivität der ersten Schicht ist
z1 = x1w1 + x2w2 + w3 und y1 = S(z1)
z2 = x1w4 + x2w5 + w6 und y2 = S(z2)

und der zweiten Schicht
y = y1w7 + y2w8 + w9

Wählen wir w1=-1, w2=+1, w3 = 0, so ist y1=1 nur bei x1=0,x2=1, sonst null.
Wählen wir w4=+1, w5=-1, w6 = 0, so ist y2=1 nur bei x1=1,x2=0, sonst null.
Wählen wir w7=+1, w8=+1, w9 = 0, so ist y=1 nur bei y1=1 oder y2=1, also bei (x1=0,x2=1) oder bei (x1=1,x2=0), sonst null. Dies ist die XOR-Funktion.

AS-2.6 Lernregeln **10P**

a) Leiten Sie eine Gradienten-Lernregel für die Gewichte eines linearen Neurons für die Minimierung des LMSE (Least Mean Squared Error) für überwachtes Lernen her. (7P)

Mit der Zielfunktion $R = \langle (y(\boldsymbol{x}) - L(\boldsymbol{x}))^2 \rangle$ und der Ableitung

$$d\langle (y(\boldsymbol{x}) - L(\boldsymbol{x}))^2 \rangle / d\boldsymbol{w} = \langle 2(y(\boldsymbol{x}) - L(x))\, y'(\boldsymbol{w}) \rangle = \langle 2(y(\boldsymbol{x}) - L(\boldsymbol{x}))\, \boldsymbol{x} \rangle$$

ist die Gradienten-Lernregel

$$\boldsymbol{w}(t+1) = \boldsymbol{w}(t) - \gamma\, dR/d\boldsymbol{w} = \boldsymbol{w}(t) - \gamma \langle (y(\boldsymbol{x}) - L(\boldsymbol{x}))\, \boldsymbol{x} \rangle$$

b) Vergleichen Sie diese Regel mit derjenigen von Widrow-Hoff. Was ist an der Widrow-Hoff-Regel verschieden von der Hergeleiteten und warum? (3P)

Die Widrow-Hoff-Lernregel ist

$\boldsymbol{w}(t+1) = \boldsymbol{w}(t) - \gamma(y(\boldsymbol{x}) - L(\boldsymbol{x}))\, \boldsymbol{x}/\boldsymbol{x}^2$

und unterscheidet sich in zwei Aspekten: Zum einen nutzt sie den stochastischen Gradienten und nicht den Erwartungswert, und zum anderen normiert sie die Eingabe mit dem Längenquadrat des Eingabevektors, um eine Korrektur unabhängig von der Eingabe zu erreichen.

AS-2.7 ROC-Kurven 6P

1. Definieren Sie die folgenden wichtigen Begriffe: (3P)
   a) Sensitivität

Die Sensitivität ist die Wahrscheinlichkeit, bei vorliegender Klasse auch darauf zu entscheiden.

   b) Spezifität

Die Spezifität ist die Wahrscheinlichkeit, bei nicht vorliegender Klasse auch darauf zu entscheiden, dass sie nicht vorliegt.

   c) Fehlalarm und Ignoranz

Die „Ignoranz“ oder „Fehler 1.Art“ ist, wenn beim Auftreten einer Klasse mit einer Wahrscheinlichkeit der Auftritt nicht erkannt wird; erkennt man die Klasse zu Unrecht, obwohl sie nicht auftritt, wird diese Wahrscheinlichkeit „Fehlalarm“ oder „Fehler 2. Art“ genannt.

2. Erklären Sie mit kurzen Stichworten, wie man die ROC-Kurve für ein gegebenes Diagnosesystem D(x) ermittelt. (1P)

Seinen die auftretenden Ereignisse x mit der Diagnose D(x) in ihrer Art bestimmt, so erhält man für alle Ereignisse X = {x} je einen Wert für das Tupel S = (Sensitivität, Spezifität). Da D(x) = D(p,x) mindestens von einem Parameter p abhängig ist, kann man durch Veränderung von p für jeden Wert von p auch ein davon abhängenden Wert S für die Diagnose D(X) erhalten. Die Menge aller Ergebnispunkte {S} von unterschiedlichen Parameterwerten bildet approximiert eine Kurve, die ROC-Kurve.

3. Wie sieht die ROC-Kurve eines von der wahren Diagnose völlig unabhängigen Diagnosesystems aus? (1P)

Wenn die Diagnosewahrscheinlichkeit unabhängig von dem Klassenauftritt w ist, so gilt P((D(x)=w | w) = P((D(x)=w | ¬w) oder Sensitivität = 1 – Spezifität. Die ROC-Kurve bildet eine Gerade von S = (0,1) zu (1,0).

4. Wozu werden ROC-Kurven benötigt? Welche anderen Gütekriterien kennen Sie? (1P)

Aus ROC-Kurven wird die Fläche unter der Kurve AUC gemessen. Dies dient zum Vergleich der Diagnosegüten unterschiedlicher Diagnosesysteme. Man kann dazu auch die mittlere Diagnosewahrscheinlichkeit (Spezifität+Sensitivität)/2 oder den Schnittpunkt der Diagonalen von S = (0,0) zu (1,1) mit der ROC-Kurve, dem Punkt gleicher Fehler EER (equal error rate) nehmen.

AS-2.8 Lagrange-Optimierung 6P

Ein Blumenkasten hat einen rechteckigen Boden der Abmessungen *a* und *b* und eine bestimmte Höhe *h*. Das Volumen *V* ist vorgegeben. Sie haben nur wenig Farbe, um ihn von außen anzustreichen. Bei welchen Abmessungen benötigen Sie am wenigsten Farbe?

a) Benennen Sie das Optimierungsziel sowie die Nebenbedingungen und stellen Sie dazu die Langrange-Funktion auf. (3P)

Das Ziel besteht darin, die Oberfläche F=ab + 2ah + 2bh zu minimieren über alle Blumenkästen (Quader) mit dem festen Volumen V = abh. Damit erhalten wir die Nebenbedingung (V-abh) = 0 und die Lagrangefunktion ist

$$L(a,b,h,\mu) = F(a,b,h) + \mu(V-abh)$$

b) Berechnen Sie aus der Funktion das Optimum (3P)

Die Ableitungen der Lagrangefunktion nach den Parametern a, b, h ergeben die Gleichungen

1) $b+2h - \mu bh = 0$
2) $a+2h - \mu ah = 0$
3) $2a+2b+\mu ab = 0$

Aus 1) und 2) folgt $(b+2h)/bh = \mu = (a+2h)/ah$ *oder* $ab+2ah = ab+2bh$ *oder* $a = b$.
Aus 1) und 3) folgt $(b+2h)/bh = \mu = (2a+2b)/ab$ *oder* $ab+2ah = 2ah+2bh$ *oder* $a =2h$.

Also wird das Optimum bei $a= b = 2h$ *erreicht.*

AS-2.9 ICA-Verfahren 13P

Die *Independent Component Analysis* ICA ist ein lineares Trennungsverfahren.

a) Erläutern Sie mit Stichworten die Problemstellung sowie mögliche Verfahren zur ICA. (2P)

Wenn mehrere unabhängige Signale linear gemischt werden, entstehen lineare Mischungen. Die ICA bietet Verfahren an, die inverse Matrix dieser linearen Mischung zu finden, also die Originalsignale aus den Mischungen wieder herzustellen. Mögliche Verfahren dazu beruhen auf der minimalen Transinformation zwischen den gewonnen Signalen oder aber auf dem unterschiedlichen statistischen Moment (Kurtosis) der Quellen.

b) Welche 4 Einschränkungen gelten für die ICA? (4P)

a. *Nicht mehr als eine Gauß'sche Quelle*
b. *Reihenfolge der Quellen unbestimmt*
c. *Varianz (Skalierung) der Quellen unbekannt*
d. *Für jede Quelle wird ein Mischsignal benötigt. Sonst wäre die Mischmatrix nicht regulär und damit nicht invertierbar.*

c) Geben Sie die notwendigen Berechnungsschritte an und beschreiben Sie mit kurzen Stichworten, was der jeweilige Schritt für Aufgaben hat. (4P)

a. *Zentrieren: Die Mischsignale jeweils auf Erwartungswert = 0 bringen.*
b. *Weißen: Zuerst PCA durchführen auf den Signalen, beispielsweise durch Finden der Eigenvektoren der Autokorrelationsmatrix. Dann die Eigenvektoren (=Gewichtsvektoren) der Länge 1 normieren mit der Quadratwurzel der Eigenwerte => Dekorrelation und Normalisierung der Daten*
c. *Entmischen: z.B. über minimale Transinformation oder extremale Kurtosis.*

d) Wie erhält man mehrere unabhängige Komponenten, wenn die Methode immer nur eine liefert? (1P)

*Bei der Methode der extremalen Kurtosis erhält man nur die Komponente mit maximaler Kurtosis. Man muss dann von der Eingabe **v** den Anteil der letzten Komponente **w**y abziehen und von der so korrigierten Eingabe erneut die maximale Kurtosis finden. Man erhält so die zweite Komponente mit der zweitgrößten Kurtosis. Dies führe mit iterativ durch und erhält so alle Komponenten.*
Grundlage dafür ist die Tatsache, dass die gewünschte Gewichtsmatrix orthogonal ist.

e) Wie lautet die Fixpunktgleichung für eine Komponente? (2P)

*Für eine Komponente bzw. einen Gewichtsvektor **w** der Matrix **W** lautet die Fixpunktgleichung von Hyvarinen für die geweisste Eingabe **v***

$$\mathbf{w}(t+1) = \langle(\mathbf{w}^T\mathbf{v})^3\mathbf{v}\rangle - 3\mathbf{w} \quad \text{mit } |\mathbf{w}(t+1)| = 1 \text{ normiert}$$

MatrikelNr:

MatrikelNr: