Institut für Informatik 4
Priv.-Doz. Dr. V. Steinhage
Friedrich-Ebert-Allee 144
53113 Bonn
Email: steinhage@cs.uni-bonn.de
WWW: http://net.cs.uni-bonn.de/ivs/

Blatt 10 (9 Punkte)

Abgabe durch Hochladen auf der eCampus-Seite bis Sonntag, 26.06.2016, 23:59 Uhr, in Gruppen von 2-3 Personen.

Aufgabe 10.1: Lineare SVM: Trennebene $(\mathbf{0.5 + 0.5 + 1.5 + 1.5 = 4})$

Gegeben seien folgende Punkte: $\mathbf{x}_1 = (2,1)$, $\mathbf{x}_2 = (3,2)$, $\mathbf{x}_3 = (1,3)$, und $\mathbf{x}_4 = (1,4)$, sowie deren Klassifikationsvektor $\mathbf{y} = (1,1,-1,-1)$. Es soll mit Hilfe des SVM-Algorithmus (Vorlesung 16) die Maximum-Margin-Trennlinie $\langle w, x\rangle - b = 0$ bestimmt werden.

a) Berechnen und tragen Sie als Vorbereitung zunächst die Produkte $\mathrm{p}(i,j) = y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j\rangle$ für $i, j \in \{1,2,3,4\}$ in die folg. Tabelle ein. *Tipp: Aus Symmetriegründen sind es nur zehn verschiedene Werte.*

p(i,j)	1	2	3	4
1				
2				
3				
4				

b) Bestimmen Sie nun unter Nutzung der in Teil a) ermittelten Produkte die vier Lagrange-Multiplikatoren α_i, indem Sie die Zielfunktion (s. Folie 14, Vorlesung 16)

$$W(\boldsymbol{\alpha}) = \sum_i \alpha_i - 0.5 \sum_{ij} [\alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j\rangle] = \sum_i \alpha_i - 0.5 \sum_{ij} [\alpha_i \alpha_j \, \mathrm{p}(i,j)]$$

bezüglich der α_i unter den Nebenbedingungen $\alpha_i \geq 0$ und $\sum_i \alpha_i y_i = 0$ maximieren. **Wichtig für die Bewertung**: Schreiben Sie zunächst die bis auf die α_i, α_j vollständig instanziierte Zielfunktion $W(\boldsymbol{\alpha})$ und die ebenso instanziierten Nebenbedingungen auf! Zur Lösung der instanziierten Zielfunktion $W(\boldsymbol{\alpha})$ unter den instanziierten Nebenbedingungen verwenden Sie einen LP-Solver Ihrer Wahl. Hinweis: I. A. müssen Sie dort die $\alpha_1, ..., \alpha_4$ als Variable a, b, c, d oder x, y, z, t o. ä. eingeben. Ein Beispiel-LP-Solver: http://calculator.tutorvista.com/math/594/linear-programming-calculator.html. Reichen Sie einen Screenshot der Lösung Ihres gewählten LP-Solvers ein! Zur Kontrolle und zum Weiterarbeiten in c) und d) das korrekte Ergebnis: $\alpha_1 = \frac{2}{9}, \alpha_2 = \frac{2}{9}, \alpha_3 = \frac{4}{9}, \alpha_4 = 0$.

c) Berechnen Sie nun den Normalenvektor der Trennlinie

$$\mathbf{w} = \sum_i (\alpha_i y_i \mathbf{x_i})$$

und bestimmen Sie die Indexmenge der Stützvektoren $N = \{i | \mathbf{x}_i \text{ ist Stützvektor}\}$. Hiermit errechnen Sie den Abstand b der Trennlinie zum Ursprung gemäß

$$b = \frac{1}{|N|} \sum_{i \in N} [\langle \mathbf{w}, \mathbf{x}_i\rangle - y_i]$$

d) Klassifizieren Sie $\mathbf{x}_{\text{neu}} = (4,4)$ mittels $y_{\text{neu}} = \text{sign}(\langle \mathbf{w}, \mathbf{x}_{\text{neu}}\rangle - b)$ und zeichnen Sie die Trainingsmenge, sowie die Trennebene und den Punkt $\mathbf{x}_{\text{neu}}$.

Aufgabe 10.2: SVM: Kernel Trick **(1)**

In dieser Aufgabe soll eine Support-Vektormaschine konstruiert werden, die aus zwei Eingaben x_1 und x_2 die XOR-Funktion berechnet. Sie wissen, dass XOR nicht linear separierbar ist. Also brauchen wir den *Kernel Trick*. Es ist praktisch, als Eingaben und als Ausgaben Werte von 1 und -1 statt 1 und 0 zu verwenden:

x_1	x_2	XOR
-1	-1	-1
-1	1	1
1	-1	1
1	1	-1

Es ist typisch, eine Eingabe (einen Datenvektor) x in einen Merkmalsraum aus fünf Dimensionen abzubilden, der aus den beiden ursprünglichen Dimensionen x_1 und x_2 und den drei Kombinationen x_1^2, x_2^2 und $x_1 \cdot x_2$ aufgespannt wird. Für diese Aufgabe betrachten wir jedoch nur die beiden Dimensionen $f_1 = x_1$ und $f_2 = x_1 \cdot x_2$. Zeichnen Sie die 4 möglichen Eingabebeispiele in diesen Merkmalsraum. Zeichnen Sie außerdem die Trennlinie ein, die den Abstand zu den Trainingsbeispielen maximiert, und den Rand (bze. die *Margin*).

Aufgabe 10.3: Verstärkendes Lernen: Naives Aktualisieren **(2)**

Betrachten Sie eine Modellwelt mit vier Zuständen (A, B, C, D), in der sich ein Agent bewegen kann. D sei Terminalzustände mit den *Reward* von +1 Utility-Einheiten. Jeder Zustandsübergang kostet den Agenten -0.1 Utility-Einheiten.
Beim passiven Lernen gibt es nur eine einzige von der (durch die Umwelt erzeugten) Strategie π festgelegte Aktion $\pi(s)$ in jedem Zustand s. Diese ist hier der Einhachheit halber $a = \rightarrow$ (“Right”) für jeden nichtterminalen Zustand. Folgende Trainingssequenz wurde beobachtet:

$$A \rightarrow B \rightarrow A \rightarrow A \rightarrow C \rightarrow D\ (+1).$$

Berechnen Sie die Nutzen U^π der einzelnen Zustände nach der Methode des *Naiven Aktualisierens* gemäß den Algorithmen *Passive-RL-Agent* und *Naive Update* (Vorl. 17, Folien 12, 17). Geben Sie die Werte nach jedem Besuch eines Zustandes in einer Tabelle nach der folg. Form an. Zu Ihrer eigenen Benotungssicherheit geben Sie auch die Berechnungen dazu an.

Besuch	$U^\pi(A)$	$U^\pi(B)$	$U^\pi(C)$	$U^\pi(D)$
#1				
#2				
#3				

Aufgabe 10.4: Verstärkendes Lernen: ADP **(2)**

Betrachten Sie dieselbe Modellwelt und die dieselbe Trainingssequenz wie in der letzten Aufgabe:

$$A \to B \to A \to A \to C \to D\ (+1).$$

Berechnen Sie folgende Werte nach der Methode des *Adaptiven Dynamischen Programmierens* (s. Algm. *Passive-ADT-Agent*):

1. U(A) und R(A) nach dem 1. Besuch von Zustand A;
2. nach dem 1. Besuch von Zustand B:
   - U(B) und R(B),
   - $N_{sa}(A, \to)$, $N_{sas'}(A, \to, B)$ und $M(A, \to, B)$,
   - $U(A)$.