

8. Übungsblatt

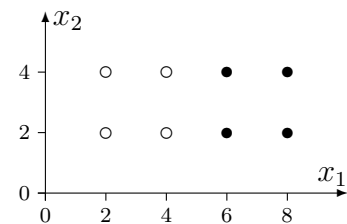
Aufgabe 28 Voronoi-Diagramm

Bestimmen Sie auf graphischem Wege die Voronoi-Zerlegung des Eingaberaums eines neuronalen Netzes mit den in der untenstehenden Tabelle gegebenen Punkten P_i der x_1 - x_2 -Ebene. (Hinweis: Benutzen Sie die Delaunay-Triangulation.)

i	x_1	x_2
0	1	2
1	2	6
2	3	9
3	4	7
4	5	1
5	5	4
6	7	6

Aufgabe 29 Wettbewerbslernen / Lernende Vektorquantisierung

Gegeben seien die rechts gezeigten acht Trainingsmuster, die zwei Klassen A (leere Kreise) und B (ausgefüllte Kreise) angehören. Dieser Mustersatz soll mit Hilfe von zwei Referenzvektoren unter Verwendung des euklidischen Abstands quantisiert werden. Welche Endposition werden die Referenzvektoren im Idealfall einnehmen, wenn



- nur die „Anziehungsregel“ (Muster gleicher Klasse ziehen Referenzvektoren an),
- sowohl die „Anziehungsregel“ als auch die „Abstoßungsregel“ (Muster anderer Klasse stoßen Referenzvektoren ab)

zur Änderung der Positionen der Referenzvektoren verwendet werden? (Hinweis: Sie brauchen das Verfahren nicht im Detail durchzurechnen. Die Lösung kann direkt aus der Struktur der Trainingsmuster abgelesen werden.)

Aufgabe 30 Wettbewerbslernen / Lernende Vektorquantisierung

Das unter der URL <http://www.borgelt.net/lvqd.html> verfügbare Programm visualisiert die lernende Vektorquantisierung für zweidimensionale Daten. (Es können höherdimensionale Daten geladen werden. Das Programm bietet dann die Möglichkeit, zwei Eingabevariablen auszuwählen, die verwendet werden sollen — siehe Menüpunkt `Settings > Attributes`.)

a) Wenden Sie dieses Programm auf die Irisdaten an (die Eingabedatei `iris.tab` ist im Quelltextpaket `lvqd.zip` enthalten). Verwenden Sie die Eingabegrößen Blütenblattlänge (`petal.length`) und Blütenblattbreite (`petal.width`). Wählen Sie als Klasse

“<none>” (d.h. Lernen nur mit Anziehungsregel) und lassen Sie drei Referenzvektoren bestimmen! Was passiert bei 6 Referenzvektoren? Welchen Einfluß hat die Lernrate? Verwendet das Programm *Batch-* oder *Online-Lernen*?

b) Lassen Sie jetzt die Klasseninformation berücksichtigen. Wählen Sie ein oder zwei Referenzvektoren je Klasse. Vergleichen Sie die Ergebnisse!

Aufgabe 31 Selbstorganisierende Karten

Das unter der URL <http://www.borgelt.net/somd.html> verfügbare Programm visualisiert die Entwicklung einer selbstorganisierenden Karte, die mit zufällig erzeugten Datenpunkten aus einer zweidimensionalen Figur trainiert wird. Führen Sie mit diesem Programm einige Trainingsläufe durch! Variieren Sie dabei die Parameter, speziell die Form der Fläche, aus der die Datenpunkte gewählt werden, die Lernrate und den Radius der Nachbarschaft. Was geschieht, wenn Sie die Lernrate oder den Nachbarschaftsradius zu klein wählen oder zu schnell abnehmen lassen? (Die Lernrate nimmt nach der Formel $\eta(t) = \eta_0 t^{-\kappa}$ ab. Die Werte für η_0 (initial learning rate) und κ (learning rate decay exponent) können in der Parameterdialogbox eingegeben werden. Der Nachbarschaftsradius nimmt nach einem gleichartigen Gesetz ab.)

Betrachten Sie insbesondere folgende Initialisierungen:

1. Wählen Sie zunächst als Kartengröße 100×1 , d.h. eine eindimensionale Karte. Wie passt sich diese Karte den zweidimensionalen Figuren in der Ebene (square, triangle, circle) an? Wählen Sie verschiedene Nachbarschaftsradien (z.B. 2.5, 5, 10, 1). (Hinweis: Stellen Sie den *Delay* auf 0 und *Redraw* auf 100. Sie können neu initialisieren, ohne den laufenden Lernvorgang zu stoppen.)
2. Wie paßt sich eine zweidimensionale Karte denselben Figuren an? Nehmen Sie z.B. 15×15 Neuronen (und entsprechend kleinere Radien). Wann kommt es zu “Verknotungen”? Was passiert bei längerem Lernen, wenn der Radius zu klein wird? (Hinweis: Sie können den Radius auch während des Lernens manuell verkleinern.)
3. Lernen Sie nun mit den zweidimensionalen Figuren im dreidimensionalen Eingaberaum (Kugeloberfläche, bzw. zweistellige Funktionen). Verwenden Sie ggf. größere Karten. (Hinweis: Die Ansicht läßt sich mit der Maus drehen, so daß sich die Anpassung besser einschätzen läßt.)