

9. Übungsblatt

Aufgabe 29 Selbstorganisierende Karten

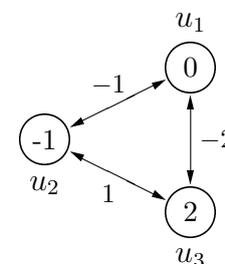
Das unter der URL <http://fuzzy.cs.uni-magdeburg.de/~borgelt/doc/somd> verfügbare Programm visualisiert die Entwicklung einer selbstorganisierenden Karte, die mit zufällig erzeugten Datenpunkten aus einer zweidimensionalen Figur trainiert wird. Führen Sie mit diesem Programm einige Trainingsläufe durch! 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). (Tipp: Stellen Sie das *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? (Tipp: 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. (Tipp: Die Ansicht lässt sich mit der Maus drehen, so dass sich die Anpassung besser einschätzen lässt.)

Aufgabe 30 Hopfield-Netze

Die nebenstehende Abbildung zeigt ein einfaches Hopfield-Netz. Bestimmen Sie für dieses Netz ausgehend vom Anfangszustand $(act_{u_1}, act_{u_2}, act_{u_3}) = (1, -1, 1)$ den bzw. die Endzustände. (Hinweis: Verwenden Sie einen Zustandsübergangsgraphen. Markieren Sie in diesem Graphen Start- und Endzustand/Endzustände. Beachten Sie, daß es kein explizites Startneuron gibt und somit alle möglichen Folgezustände zu berücksichtigen sind.)



Aufgabe 31 Hopfield-Netze: Mustererkennung

In einem Hopfield-Netz mit vier Neuronen sollen die beiden Muster

$$(-1, -1, +1, +1) \quad \text{und} \quad (+1, -1, -1, +1)$$

gespeichert werden, d.h., diese beiden Muster sollen stabile Zustände des Netzes sein, die durch eine Aktualisierung beliebiger Neuronen nicht verlassen werden.

- Berechnen Sie die Verbindungsgewichte und die Schwellenwerte der Neuronen eines Hopfield-Netzes, das die obigen Muster speichert!
- Wie viele weitere Muster können in diesem Netz noch gespeichert werden?
- Finden Sie zwei weitere Muster, die man zusätzlich in dem von Ihnen konstruierten Netz speichern könnte, ohne daß die alten Muster vergessen werden!
(Um diese Muster tatsächlich zu speichern, müssen natürlich eventuell die Verbindungsgewichte geändert werden.)

Aufgabe 32 Hopfield-Netze: Mustererkennung

Für eine praktische Anwendung soll ein automatisches System zur Erkennung von Ziffern und Großbuchstaben konzipiert werden. Die 36 zu unterscheidenden Muster werden dazu durch ein 10×10 Raster mit Schwarz/Weiß-Pixeln repräsentiert.

Die Eingaben sind ebenfalls durch Pixelmuster in diesem Raster dargestellt, können jedoch gestört sein. Ziel der Mustererkennung ist es, eine gestörte Eingabe einem Referenzmuster zuzuweisen. Betrachten Sie die folgenden zwei Vorgehensweisen:

- Verwendung eines Hopfield-Netzes als Assoziativspeicher mit einem Neuron pro Pixel
- Direkter Vergleich der Eingabe mit den gespeicherten Referenzmustern. Ausgegeben wird das Referenzmuster mit der geringsten Hammingdistanz zur Eingabe (d.h. der niedrigsten Zahl unterschiedlich gefärbter Pixel).

Vergleichen Sie beide Ansätze hinsichtlich Speicheraufwand, Laufzeitkomplexität und praktischer Einsetzbarkeit. Was zeigt sich dabei?

Freiwillig: Nennen Sie weitere Verfahrensweisen, die sich zur die Lösung des genannten Problems eignen würden.