Selbstorganisierende Karten

Marten Jäger

6. August 2007

Zusammenfassung

Selbstorganisierte Karten (*SOM*) sind ein unüberwachtes Lernverfahren mit dem Ziel eine topologische Darstellung des Eingaberaums zu erreichen. Die bekanntesten *SOM*s sind die topologieerhaltenden[1] Kohonen Karten von Teuvo Kohonen. Sie Basieren auf der Idee, daß man mit Hilfe einer Trainingsmenge $M = \{m_j = (x_j) | x_j \in X \subseteq \Re^n, j = 1 \dots \mu_M\}$ eine Menge von Neuronen $N = \{n_i = (w_i, k_i) | w_i \in X \subseteq \Re^n, k_i \in K^l, i = 1 \dots \mu_N\}$ derart aufspannt, daß die Dichte und Verteilung der Neuronen dem Wahrscheinlichkeitsmodel der Trainingsmenge entsprechen[2]. Das Bemerkenswerte an diesem Model ist die Möglichkeit, die Dimension des Problems zu reduzieren, so daß einzig ein Neuron des trainierten Netzes letztendlich für einen bestimmten Raumabschnitt des Ursprungsraumes zuständig ist.

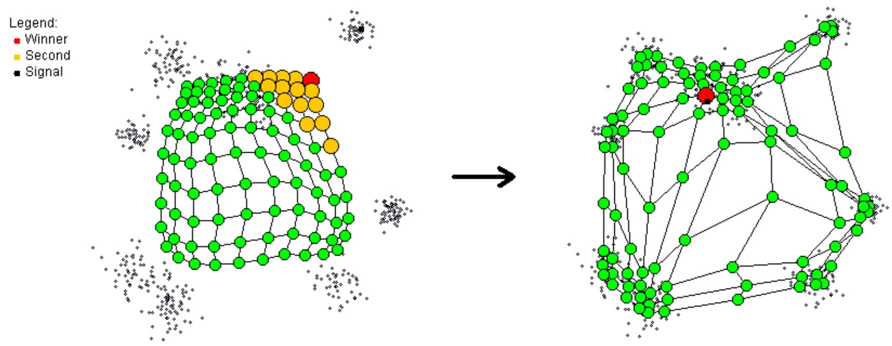

Abbildung 1: Training eines Kohonennetzes ohne Dimensionsreduktion. Auf der linken Seite sieht man das Netz und die aktivierten Neuronen am Beginn des Trainings. Auf der rechten Seite sieht man ein fertig trainiertes Netz.[1]

1 Einleitung

Der Kohonen Algorithmus berechnet eine Funktion f, welche von einem Eingaberaum A nach einen Ausgaberaum B definiert ist. Wobei die Dimension

[1]topologierhaltend: die zugrundeliegende Funktion f ist eine stetige Funktion. Dies wird später näher erläutert.

[2]siehe auch das Beispiel in Abbildung 1

des Eingaberaums A größer oder gleich der Dimension des Ausgaberaums B ist. Hierfür wird der Eingaberaum A in Subregionen aufgeteilt, welche man *chart*(engl.) oder auch *map*(engl.)[3] nennt. Die Region A, in welcher f definiert ist, läßt sich durch eine Kohonen Karte derart abdecken, daß wenn man einen Eingabevektor aus der Region a_1 auswählt, nur ein Neuron im Netzwerk B feuert.

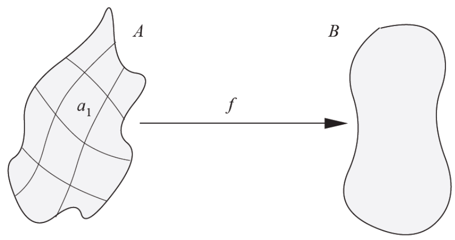

Abbildung 2: Eine Funktion $f : A \to B$

Das Model von Kohonen hat neben dem eben genannten mathematischen Hintergrund auch einen biologischen. Zahlreiche Strukturen im Gehirn haben eine lineare oder planare Topologie. Das bedeutet, sie sind ein- oder zweidimensional. Auf der anderen Seite sind sensorische Erlebnisse mehrdimensional. Hierfür möchte ich zwei kurze Beispiele geben.

Der visuelle Cortex Der visuelle Cortex ist eine sehr gut studierte Region, welche visuelle Eindrücke decodiert und verarbeitet. Die visuelle Informationen wird als zweidimensionale Projektion auf dem Cortex abgebildet[4]. In Abbildung 3 kann man zwei wichtige Merkmale dieser Projektion sehen.

1. Benachbarte Regionen des visuellen Feldes werden von benachbarten Regionen im Cortex verarbeitet.
2. Die Oberfläche des visuellen Cortex, welcher für die Verarbeitung der Informationen von der Fovea[5] zuständig ist, ist überproportional hoch. Dies impliziert, daß die Informationen vom Zentrum des visuellen Feldes detaillierter und in höherer Auflösung verarbeitet werden. Letztendlich ist dies ein sehr schönes Beispiel für Topologieerhaltung.

Der somatosensorische und motorischer Cortex Der Cortex enthält nicht nur eine topologisch geordnete Repräsentation des visuellen Feldes, sondern auch von Sinneseindrücken, welche von anderen Organen kommen. In Abbildung 4 kann man zwei Schnitte von Regionen des Gehirns sehen, in welchem die Cortices mit den dazugehörigen Körperteilen abgebildet sind. Auf der linken Seite sieht man sen somatosensorischen Cortx, welcher für die Verarbeitung von mechanischen Signalen zuständig ist. Auf der rechten Seite sieht man den motorischen Cortex, welcher die Bewegung der

[3] *chart* als auch *map* lassen sich als Karten übersetzen
[4] siehe Abbildung 3
[5] Fovea centralis, eine im Zentrum des sogenannten Gelben Flecks (Macula lutea) gelegene Einsenkung, den Bereich des schärfsten Sehens der Netzhaut bei Säugetieren.[2]

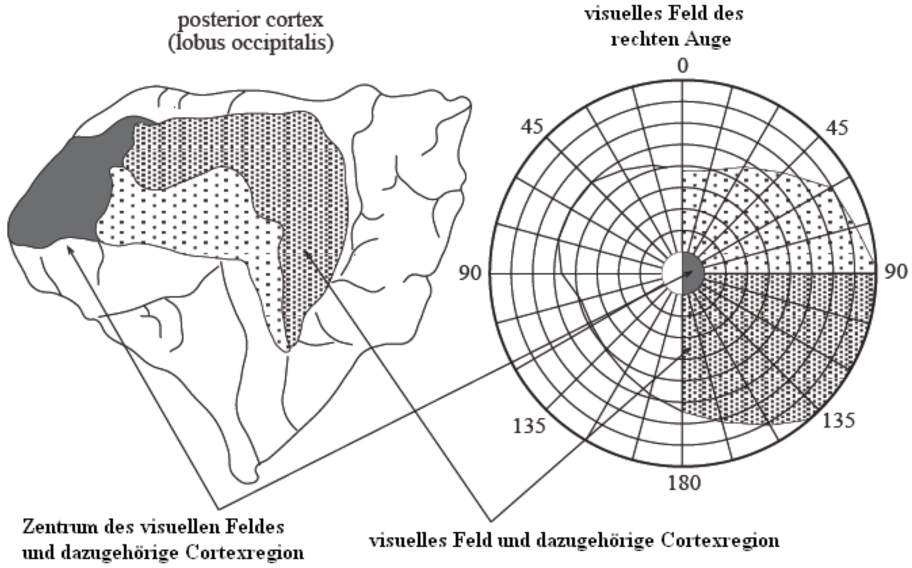

Abbildung 3: Abbildung des visuellen Feldes auf den Cortex.[3]

einzelnen Körperteile steuert. Beide Cortices sind für jede Körperhälfte in der jeweiligen Hemisphären vorhanden und fortlaufend über beide Hemisphären angeordnet. Anhand der Abbildung 4 kann man erkennen, daß die Hirnregionen für die Signalverarbeitung ebenfalls topologieerhaltend angeordnet sind. Besonders gut sieht man das, wenn man sich die Region von den Zehen zu den Fingerspitzen anschaut.

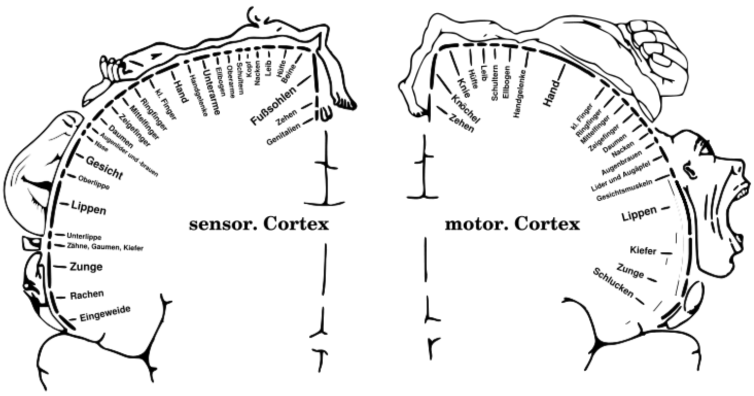

Abbildung 4: Somatosensorischer und motorischer Cortex mit dazugehörigen Körperteilen.[4]

1.1 Stetigkeit (Topologie)

Intuitiv ist eine Funktion f stetig, wenn eine Menge von Punkten in der Nähe von $f(x)$ immer das Abbild von einer Menge von Punkten in der Nähe von x enthalten. Für den verallgemeinerten topologischen Raum bedeutet das, in der Nachbarschaft von $f(x)$ sind immer die Nachbarn von x enthalten.

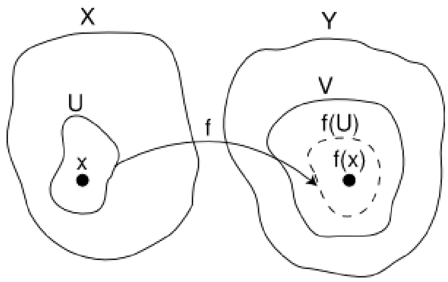

Abbildung 5: Ein graphisches Beispiel für eine stetige bzw. topologieerhaltende Funktion f.[5]

Angenommen es gibt eine Funktion f, welche von X nach Y abbildet und X und Y sind topologische Räume. f ist stetig in x für $x \in X$, wenn für jede Nachbarschaft V von $f(x)$ eine Nachbarschaft U von x existiert, so daß $f(u) \subseteq V$. In Abbildung 5 kann man eine graphische Veranschaulichung des Ganzen sehen. Obwohl diese Definition sehr kompliziert klingt, kann man sehen, daß man egal wie klein V ist, immer ein U mit einem x finden kann, daß nach V abgebildet wird.[6]

2 Kohonens Algorithmus

2.1 Algorithmus

Start Die n-dimensionalen Gewichtsvektoren $w_1 \ldots w_{\mu_N}$ werden zufällig ausgewählt. Ein Startradius δ, eine Lernkonstante ε und die Nachbarschaftsfunktionen d_A und d_x werden bestimmt.

Bei der Wahl der Nachbarschaftsfunktion muß man beachten, daß der Einfluß mit zunehmenden Abstand abnehmen soll. In den meisten Fällen wählt man eine der Gaußverteilung ähnliche Funktion. In Abbildung 6 sieht man diese und zwei ungeeignete Verteilungsfunktionen.

Schritt 1 Anhand der Trainingsmenge oder der zugrundeliegenden Wahrscheinlichkeitsverteilung wird ein Eingabevektor x ausgewählt.

Schritt 2 Das Neuron n_s^t mit der größten Erregung[6], als auch die Menge der Neuronen $N^{+t} \subset N^t$, die innerhalb des Radius δ um n_s^t liegen, werden bestimmt.

Für diesen Schritt werden zwei Metriken (Abstandsfunktionen) benötigt.

$$d_X(x_j^t, w_s^t) = \min\{d_X(x_j^t, w_i^t) | i = 1 \ldots \mu_N\}$$

[6] Das ist das Neuron, für welches der Abstand zwischen w_i und x minimal ist.

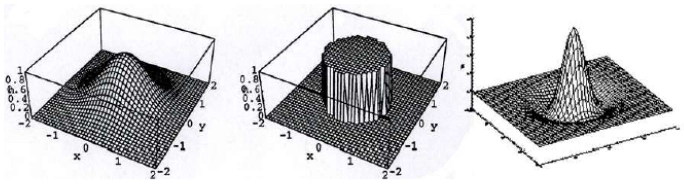

Abbildung 6: Mögliche Verteilungsfunktionen, die man für die Berechnung der Nachbarschaft verwenden könnte. v.l.n.r. Gaußsche Glockenfunktion, Zylinderfunktion und Mexican-Hat .[7]

für die Bestimmung des Gewinnerneurons n_s^t und $d_A(k_s, k_i)$ für die Bestimmung der Nachbarschaft

$$N^{+t} = \{n_i = (w_i, k_i) | d_A(k_s, k_i) \leq \delta^t\}$$

von n_s^t.

Schritt 3 Die Gewichtsvektoren in N^{+t} werden mit Hilfe der Nachbarschaftsfunktion und der Aktualisierungsregel:

$$w_i^{t+1} = w_i^t + \varepsilon^t \cdot h_{si}^t \cdot d_X(x_j^t, w_i^t)$$

mit

$$h_{si}^t = \varepsilon \frac{-d_A(k_s, k_i)^2}{2\delta^{t^2}}$$

aktualisiert.

Schritt 4 Höre auf, wenn der Algorithmus eine bestimmte Anzahl von Iterationen durchlaufen hat, ansonsten bestimme den Radius δ und die Lernkonstante ε neu und beginne erneut bei **Schritt 1**.

δ^t und ε^t werden im t-ten Iterationsschritt folgendermaßen angepaßt:

$$\delta^t = \delta_{start} \cdot \left(\frac{\delta_{end}}{\delta_{start}}\right)^{\frac{t}{t_{max}}} \quad \text{bzw. } \varepsilon^t = \varepsilon_{start} \cdot \left(\frac{\varepsilon_{end}}{\varepsilon_{start}}\right)^{\frac{t}{t_{max}}}$$

3 Anwendungsbeispiele

3.1 Kohonenkarten ohne Reduktion der Dimension

In den Abbildungen 7 und 8 sieht man verschiedene Iterationsschritte des zweidimensionalen Kohonennetzes beim Aufspannen der Silhouetten von einem Kaktus bzw. einer Tomcat.

3.2 Kohonenkarten mit Reduktion der Dimension

In den Abbildungen 9 und 10 sind Beispiele abgebildet, wie man mit Hilfe von Kohonenkarten mit einem niederdimensionalen Netz auch höherdimensionale Objekte aufspannen kann.

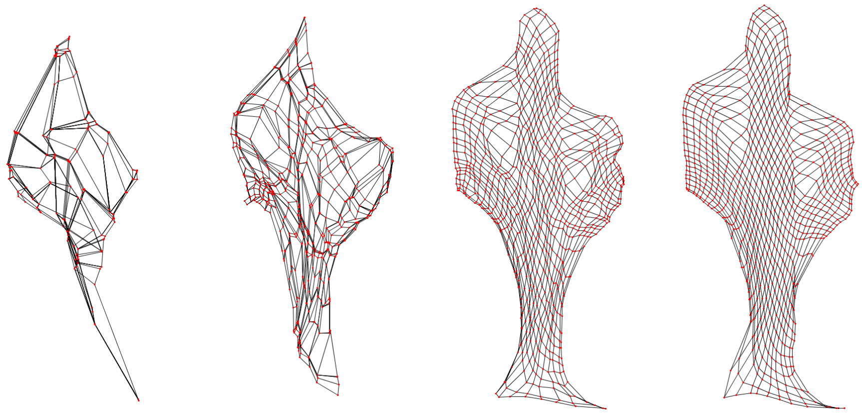

Abbildung 7: Abbildung auf die Silhouette eines Kaktus.[8]

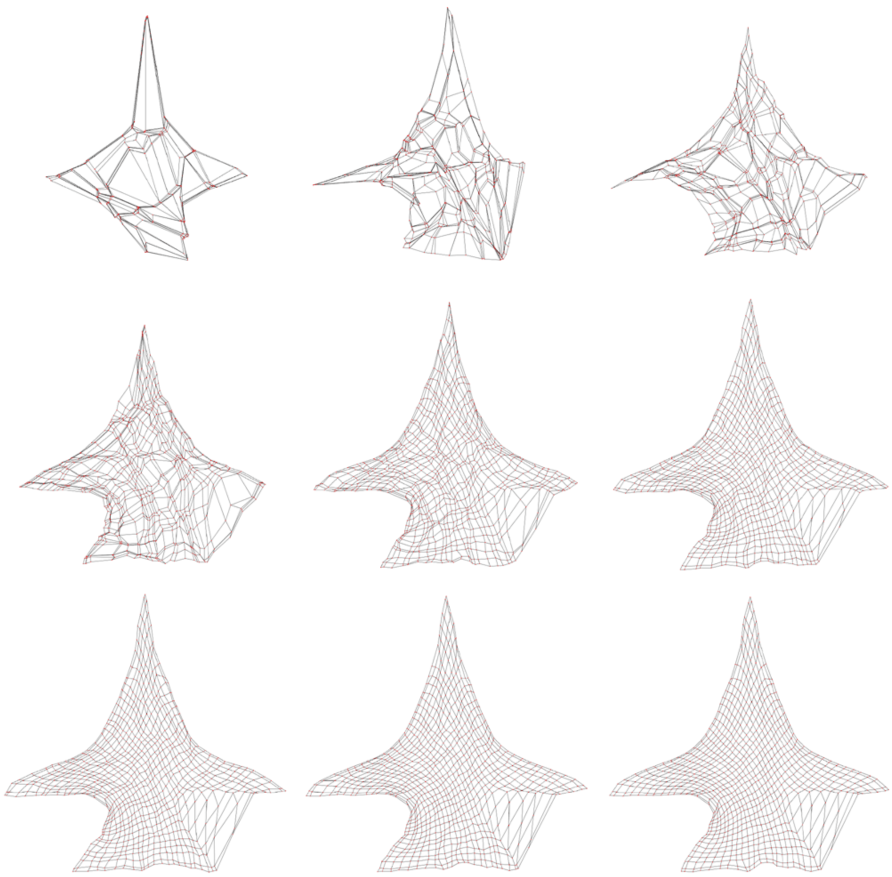

Abbildung 8: Abbildung auf die Silhouette einer F-14 Tomcat.[9]

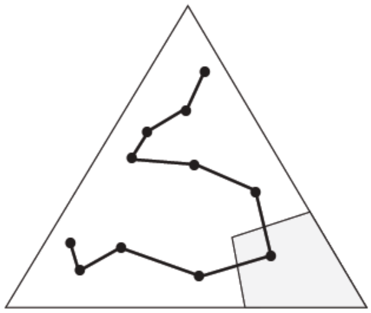

Abbildung 9: Eindimensionale Kohonenkarte, welche ein zweidimensionales Dreieck aufspannt.[3]

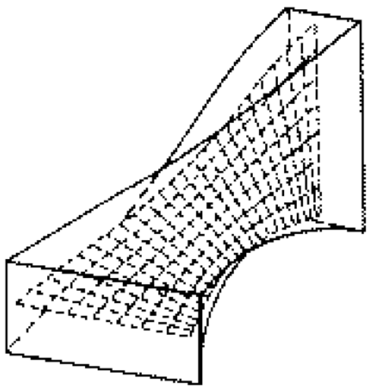

Abbildung 10: Zweidimensionale Kohonenkarten, welche ein dreidimensionales Objekt aufspannt.[10]

Literatur

[1] Hartmut S. Loos und Bernd Fritzke. Demogng.

[2] wikipedia. Fovea centralis.

[3] Raul Rojas. *Neural Networks. A Systematic Introduction.* Springer, Berlin, 1996.

[4] ralf@ark.in berlin.de. somatosensorischer und motorischer cortex.

[5] Derrick Coetzee. Continuity topology.png.

[6] wikipedia. Continuous function (topology).

[7] Yacin Bessas. Selbstorganisierende karten.

[8] James MAtthews. Mapping across a silhouette of a cactus.

[9] James MAtthews. Mapping across a silhouette of an f-14 tomcat.

[10] *Neural Nets by Kevin Gurney*, chapter 7: Competition and self-organisation: Kohonen nets. UCL Press, 1 Gunpowder Square, London EC4A 3DE, UK, 1996.