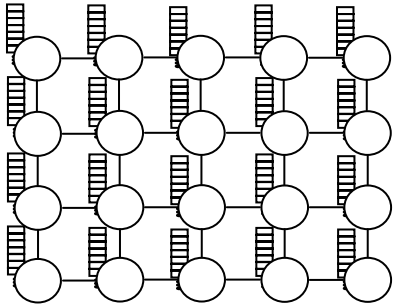
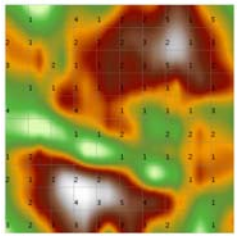
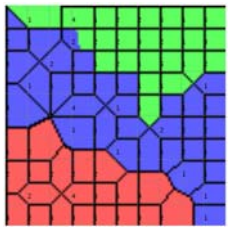
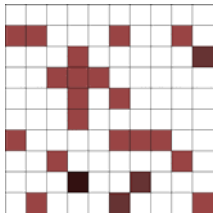
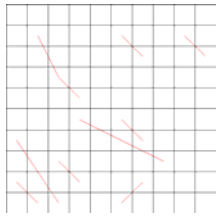
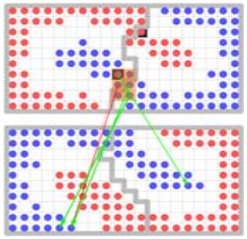
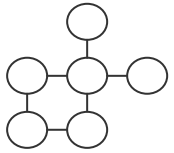
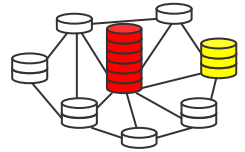
SOM Basics

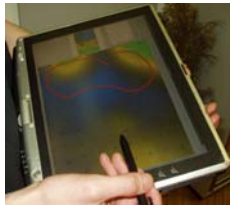
VU Selbst-Organisierende Systeme
Andreas Rauber
http://www.ifs.tuwien.ac.at/~andi

Outline

- SOM Basics
- Visualisierungen
- SOM Qualitätsmaße
- SOM Comparison
- Verwandte Verfahren & Architekturen
- Applikationen

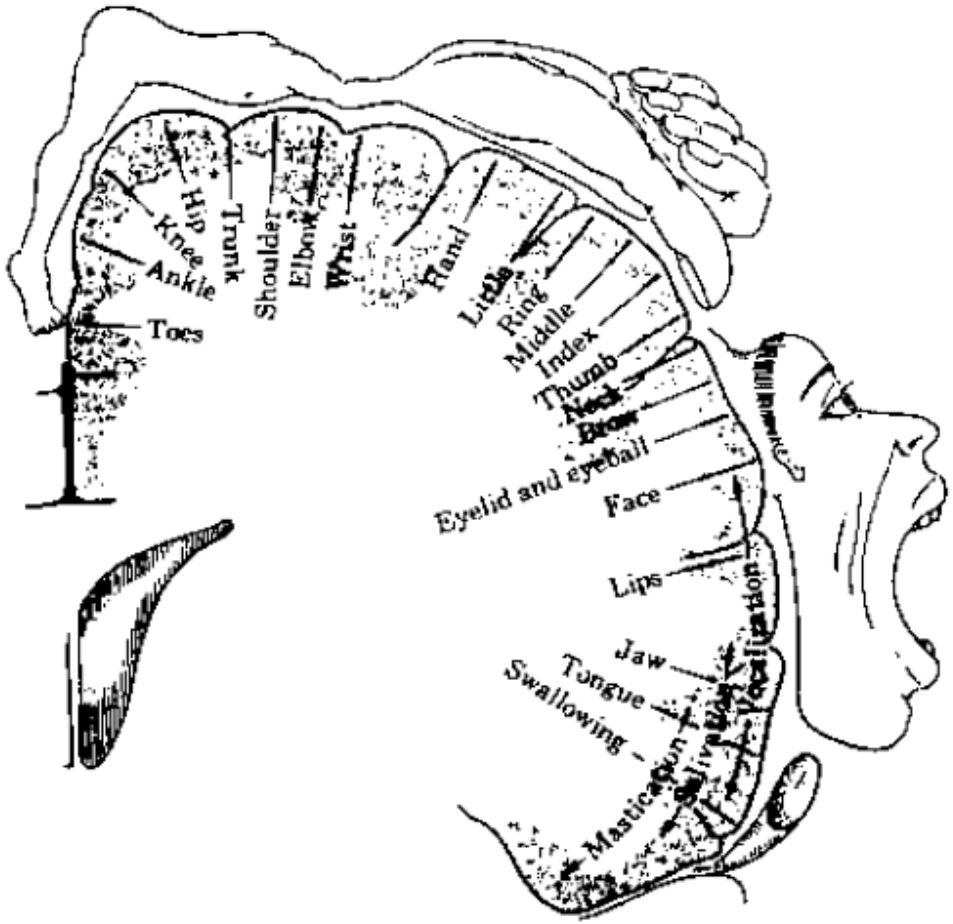



Outline

- Self-Organizing Map (SOM)
  - Physiologische Grundlagen
  - Architekturen
  - Trainingsprozess
  - Beispiele
- Verwandte Verfahren
- Visualisierungen der SOM
- Applikationen

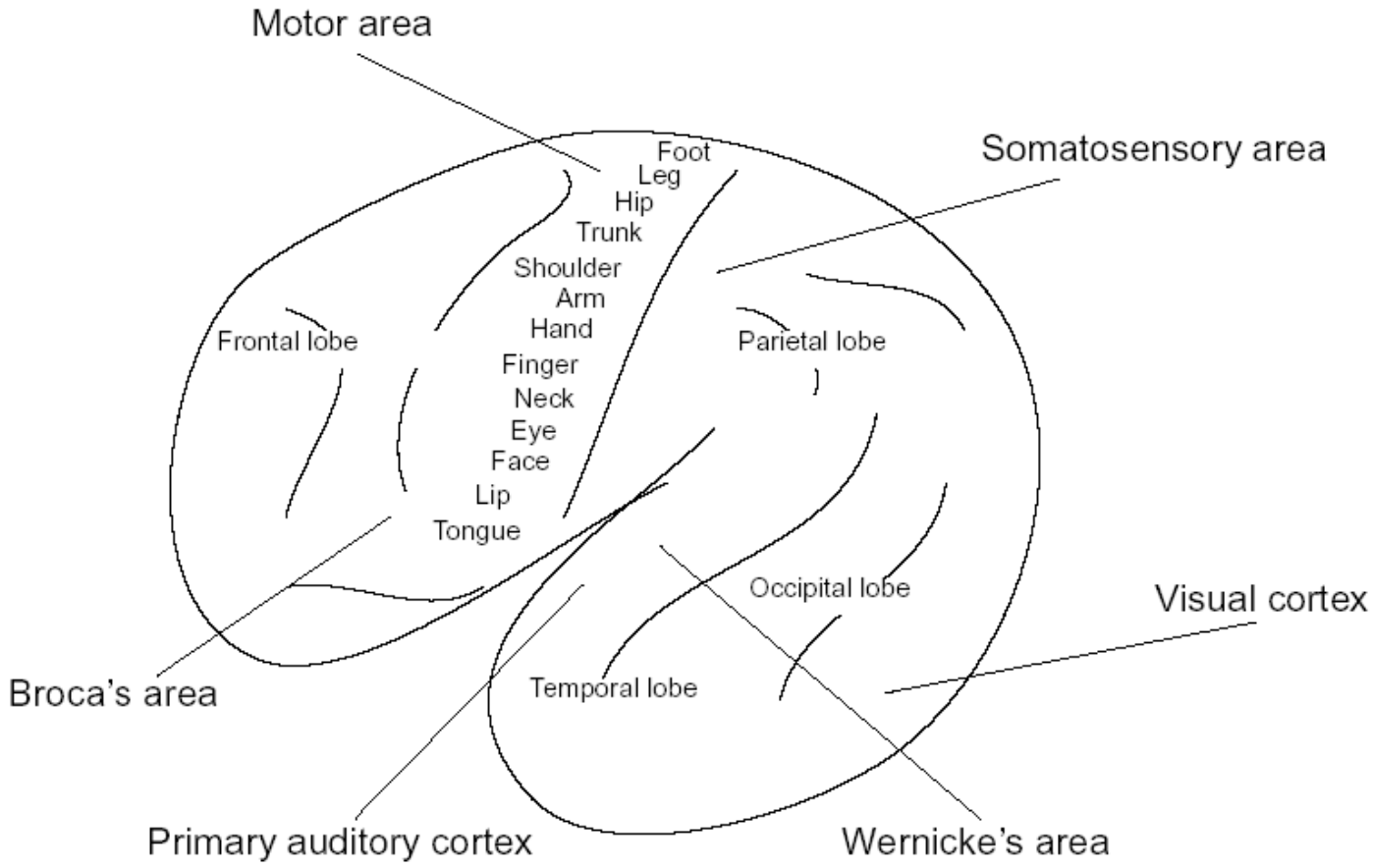
Self-Organizing Map

Physiologische Parallelen



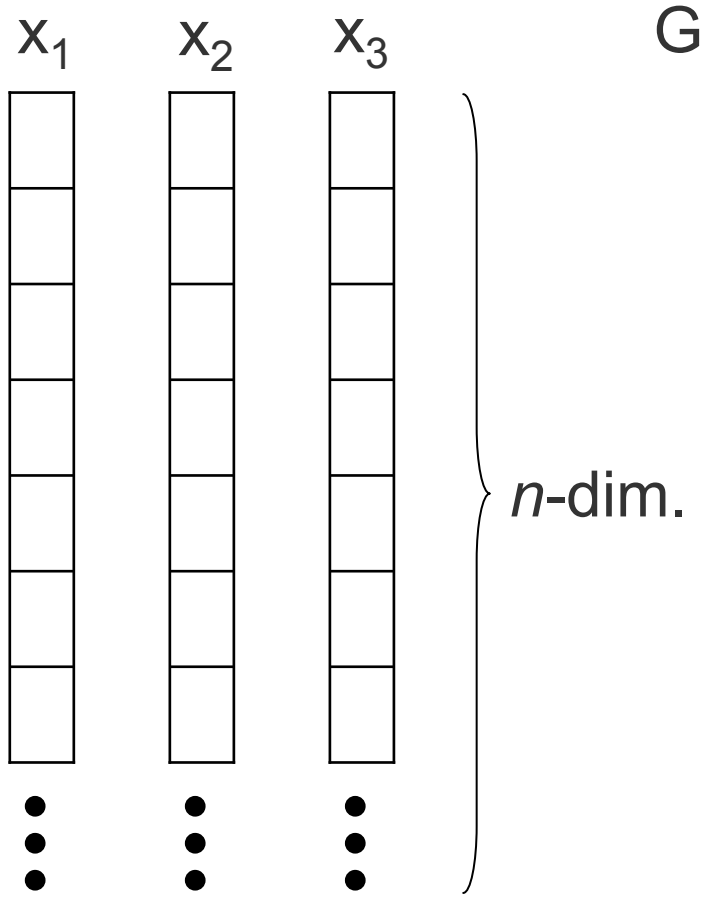
Self-Organizing Map

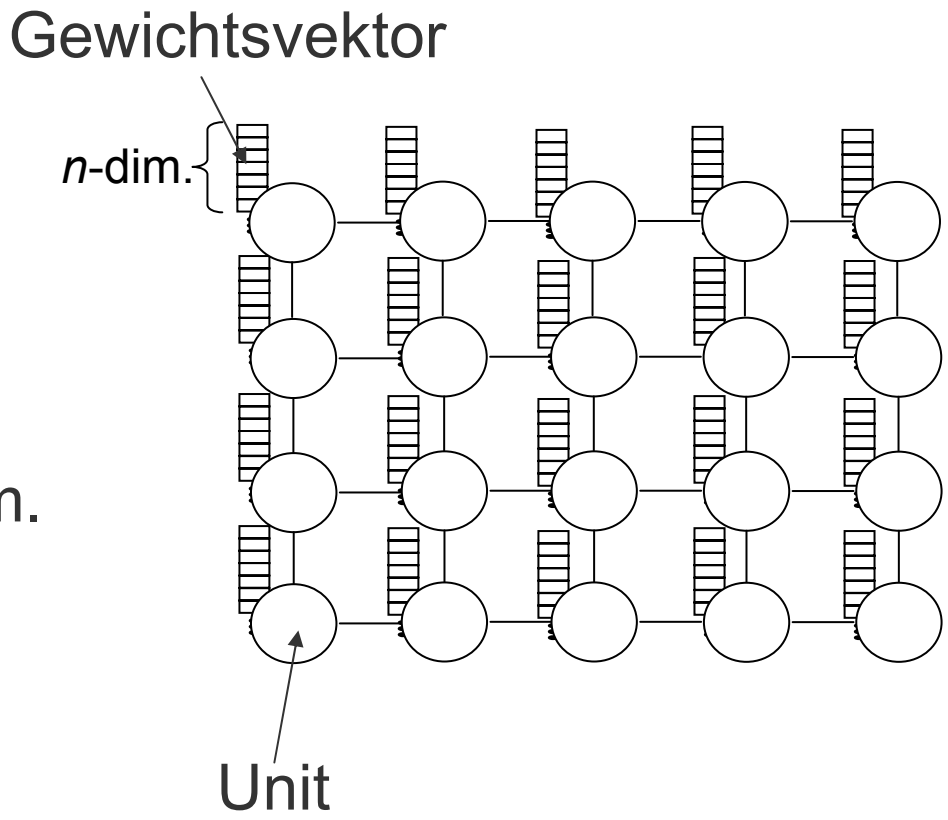
Physiologische Parallelen:



Self-Organizing Map

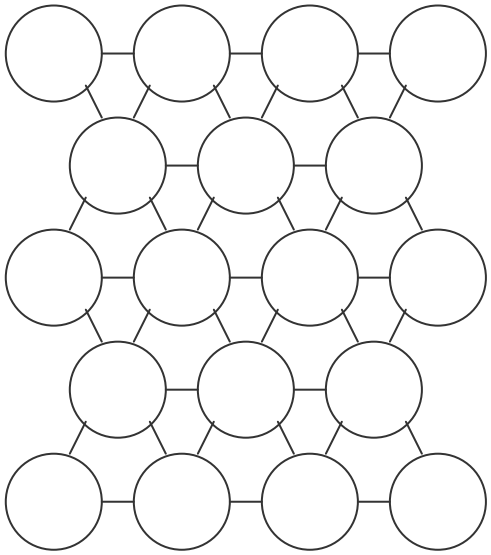
Architektur:





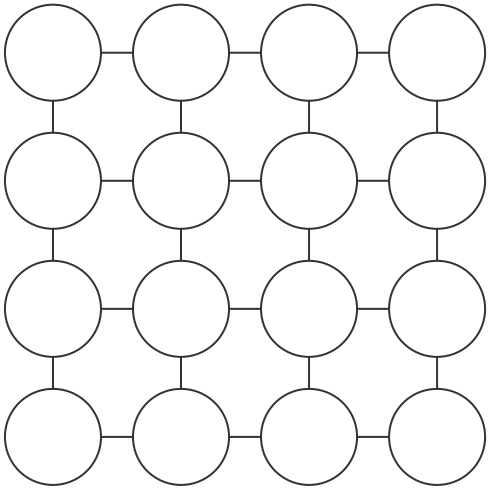
Self-Organizing Map

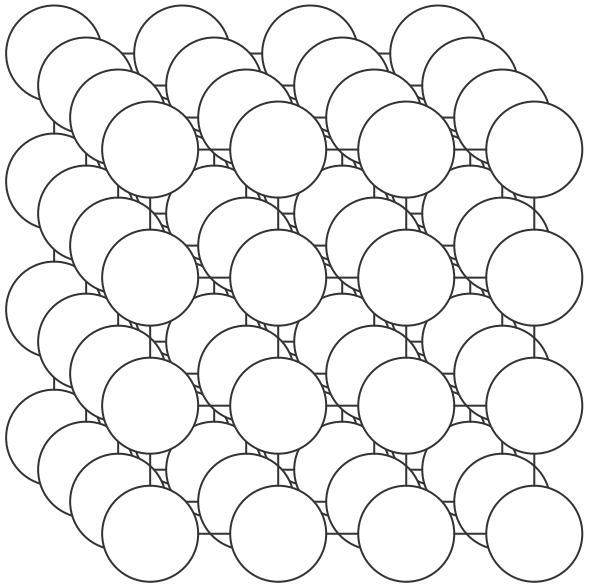
Architekturen:



hexagonal

rechteckig

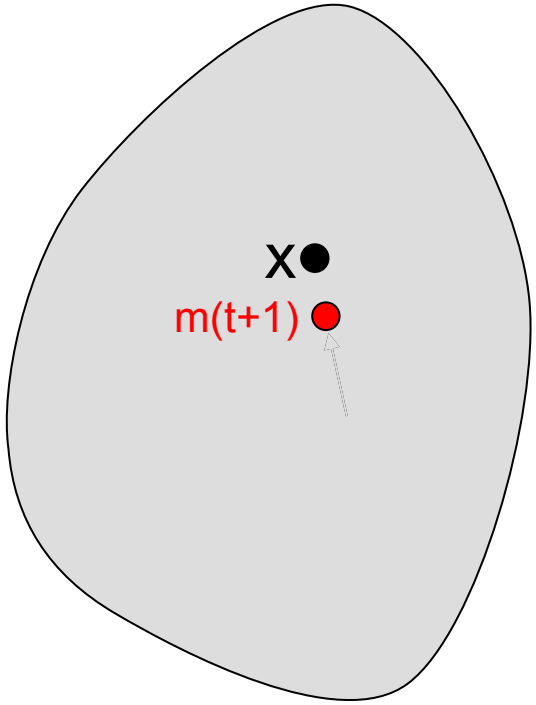


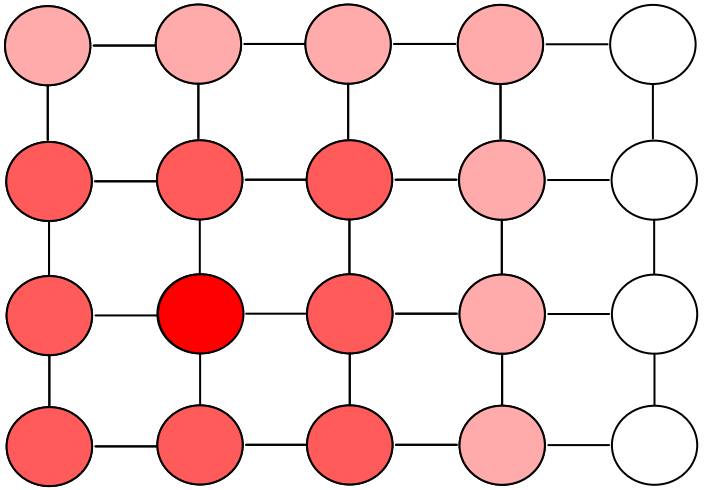


3-dimensional

Self-Organizing Map

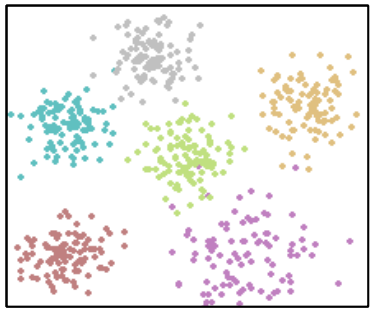
Lernverfahren:





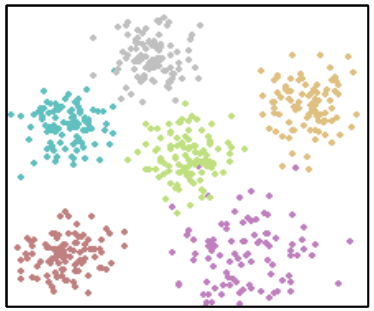
Self-Organizing Map

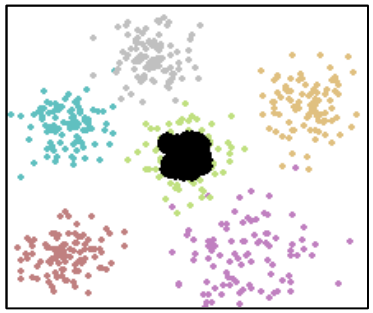
- Self-Organizing Map (SOM)



Self-Organizing Map

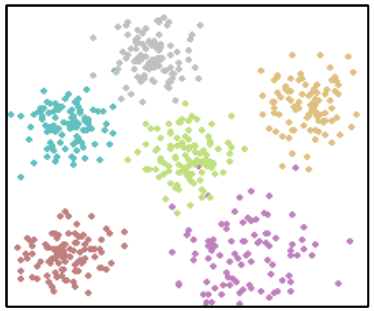
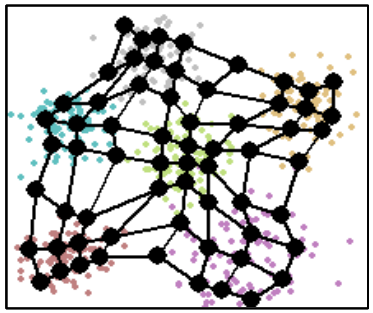
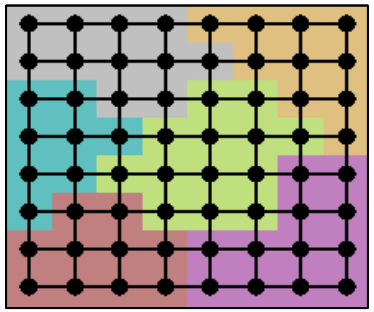
- Self-Organizing Map (SOM)





Self-Organizing Map

- Self-Organizing Map (SOM)

Self-Organizing Map

Lernverfahren:

[1] Zufällige Auswahl eines Inputvektors

[2] Berechnung der Aktivierung der Output Units. Aktivierung korrelliert mit der Distanz zwischen Input und Gewichtsvektor

[3] Auswahl der best-matching Unit (“Winner”) – Unit mit größter Aktivierung, d.h. jene Unit mit der gerinsten Distanz zwischen Input- und Gewichtsvektor

[4] Adaption der Gewichtsvektoren des “Winners” und der Units in seiner Nachbarschaft

[5] Wiederholung der Schritte 1-4 bis ein definiertes Kriterium für das Ende des Lernens erreicht ist

Self-Organizing Map

Initialisierung der Gewichtsvektoren:

- zufällige Initialisierung:
  - längerer Trainingsprozess (mehr Iterationen)
  - in jedem Trainingsprozess andere (rotierte?) Karte
- Initialisierung mit Beispieldaten
  - Vordefinierte Orientierung der Karte
  - Risiko, Topologieverletzungen „vorzuprogrammieren“
  - evtl. basierend auf existierender SOM
- komplexere Methoden: PCA
  - Initialisierung entlang der ersten beiden Hauptkomponenten
  - aufwändig bei großen Datenmengen
  - beschleunigt Trainingsprozess -> direkt zu Finetuning-Phase

Self-Organizing Map

- Lernregel

  m_i : Gewichtsvektor der Unit *i* (Codebook)

  x(t): Trainingsvektor zum Zeitpunkt *t*

  α: lernrate

  h_{ci}*:* Nachbarschaftsfunktion zum Zeitpunkt *t* zwischen Winner *c* und aktueller Unit *i*

$$m_i(t+1) = m_i(t) + \alpha(t) \cdot h_{ci}(t) \cdot [x(t) - m_i(t)]$$

Self-Organizing Map

Lernrate α :

- zu Beginn des Trainings hohe Lernrate, die sich im Laufe des Trainingsprozesses verringert
- konvergiert gegen 0 --> Konvergenz des Netzes

Self-Organizing Map

Nachbarschaftsfunktion h_{ci}:

- zu Beginn des Trainings hoch, im Laufe des Traininsprozesses verringernd
- Gauss-Funktion, Mexican Hat, Linear, Box, ...
- Vorteile v. boxed neighborhood: performance

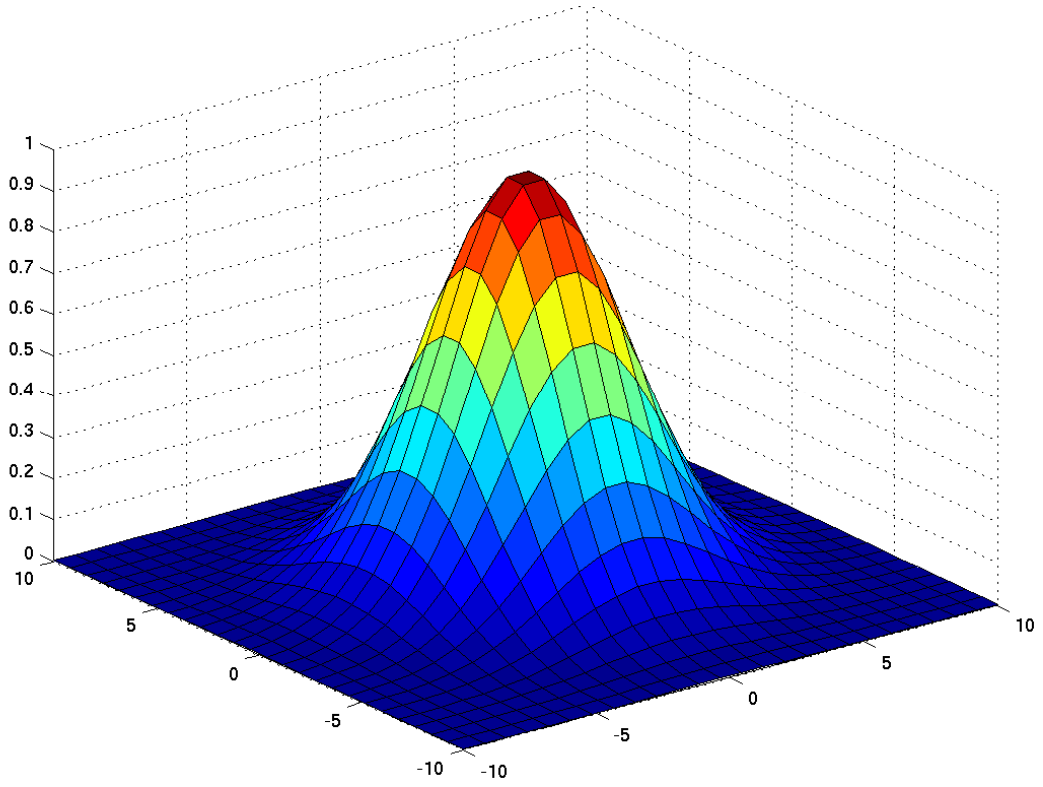
$$\lim_{t \to \infty} = N_c(t) = \{c\} \qquad \lim_{t \to \infty} = h_{ci}(t) = 0 \quad (c \neq i)$$

z.B.: $$h_{ci}(t) = \exp\left(-\frac{||r_c - r_i||^2}{2 \cdot \delta(t)^2}\right)$$

Self-Organizing Map

Nachbarschaftsfunktion

:



Self-Organizing Map

Distanzmetriken:

- L_{inf} Metrik $$d(x_1, x_2) = \max_n (|x_{1_n} - x_{2_n}|)$$
- Minkowski Metrik (L_k norm) $$d(x_1, x_2) = \sqrt[k]{\sum_n (x_{1_n} - x_{2_n})^k}$$
- Euklidische Metrik (L_2 norm) $$d(x_1, x_2) = \sqrt[2]{\sum_n (x_{1_n} - x_{2_n})^2}$$
- City block Metrik (L_1 norm) $$d(x_1, x_2) = \sum_n |x_{1_n} - x_{2_n}|$$

Self-Organizing Map

Trainingsende:

- fixe Anzahl von Iterationen
- keine weitere Änderung des Abbildungsfehlers (“quantization error”)
- Schwellwert für den Abbildungsfehler wird erreicht
- manueller Abbruch

Self-Organizing Map

Eigenschaften

- Vektor-Quantsierung
  - Clustering auf Units ähnlich k-means
  - je mehr Units, desto bessere Quantisierung
  - Beziehung zwischen “Clustern”
  - Achtung: 1 Knoten entspricht nicht einem Cluster!
  - Anzahl der Knoten nicht unmittelbar kritisch
- Topologie-erhaltende Abblidung
  - Projektion aus hoch-dim Raum in 2D
  - ähnlich PCA, Sammons Mapping
  - nicht-linear

Self-Organizing Map

Abbildungsfehler (Quantisierungsfehler):

- eine Maßzahl für die Abbildungsqualität der Daten auf die Karte
- mittlere Distanz zwischen jedem Inputvektor und seiner „best-matching Unit“ (Winner)

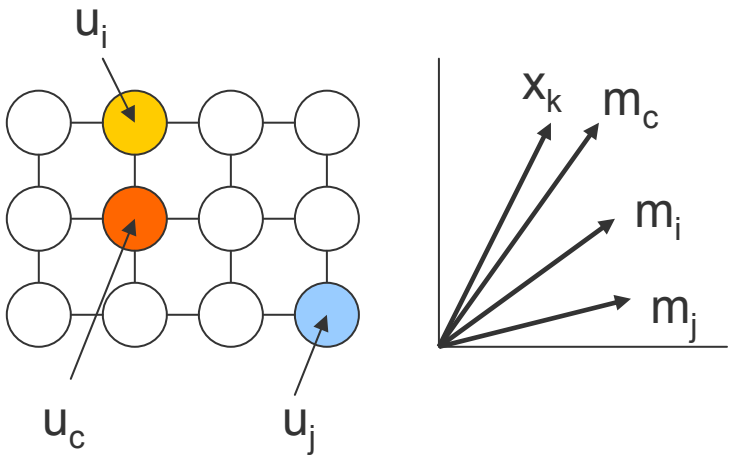
$$\mathrm{qe} = \frac{1}{|I|} \cdot \sum_{i \in I} ||x_i - m_{c(x_i)}||, \quad c(x_i) = \arg\min_j (||x_i - m_j||)$$

$I \ldots$ menge der units

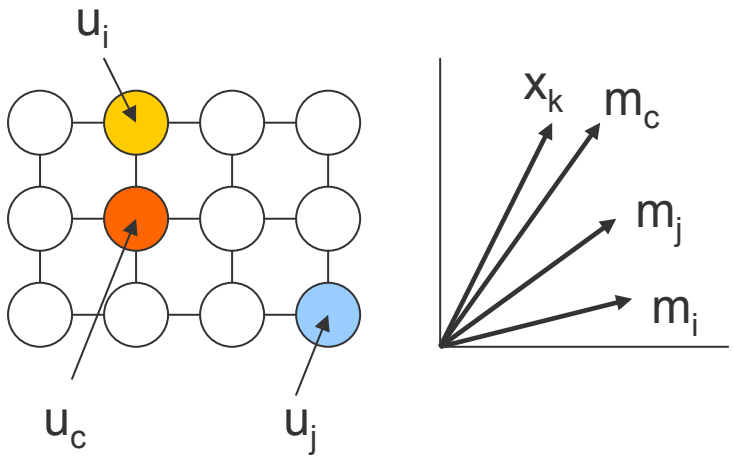
Self-Organizing Map

Eigenschaften:

- topologieerhaltende Abbildung



(a) ordered mapping



(a) disordered mapping

Self-Organizing Map

Topographischer Fehler:

- eine Maßzahl für die Qualität der Topologietreue/-Erhaltung der Daten auf der Karte
- Prozentsatz jener Inputvektoren, deren „best-matching Unit“ und „second best-matching Unit" nicht benachbart sind

Self-Organizing Map

Beispiel: Tiere

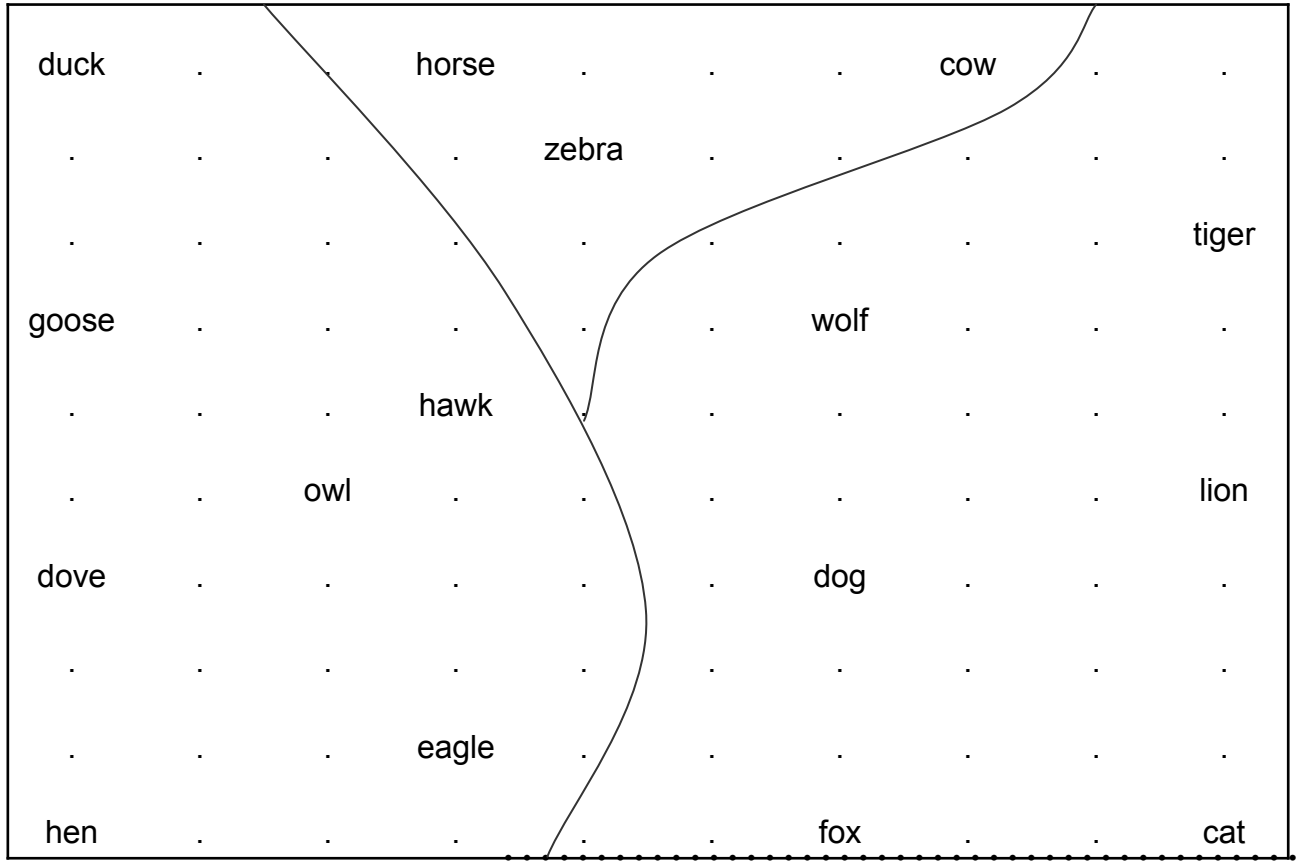
	dove	hen	duck	goose	owl	hawk	eagle	fox	dog	wolf	cat	tiger	lion	horse	zebra	cow
small	1	1	1	1	1	1	0	0	0	0	1	0	0	0	0	0
medium	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
big	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
2 legs	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
4 legs	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
hair	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
hooves	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
mane	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1	0
feathers	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
hunt	0	0	0	0	1	1	1	1	0	1	1	1	1	0	0	0
run	0	0	0	0	0	0	0	0	1	1	0	1	1	1	1	0
fly	1	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0
swim	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0

(ritter&kohonen, 1989)

Self-Organizing Map

Beispiel: Tiere

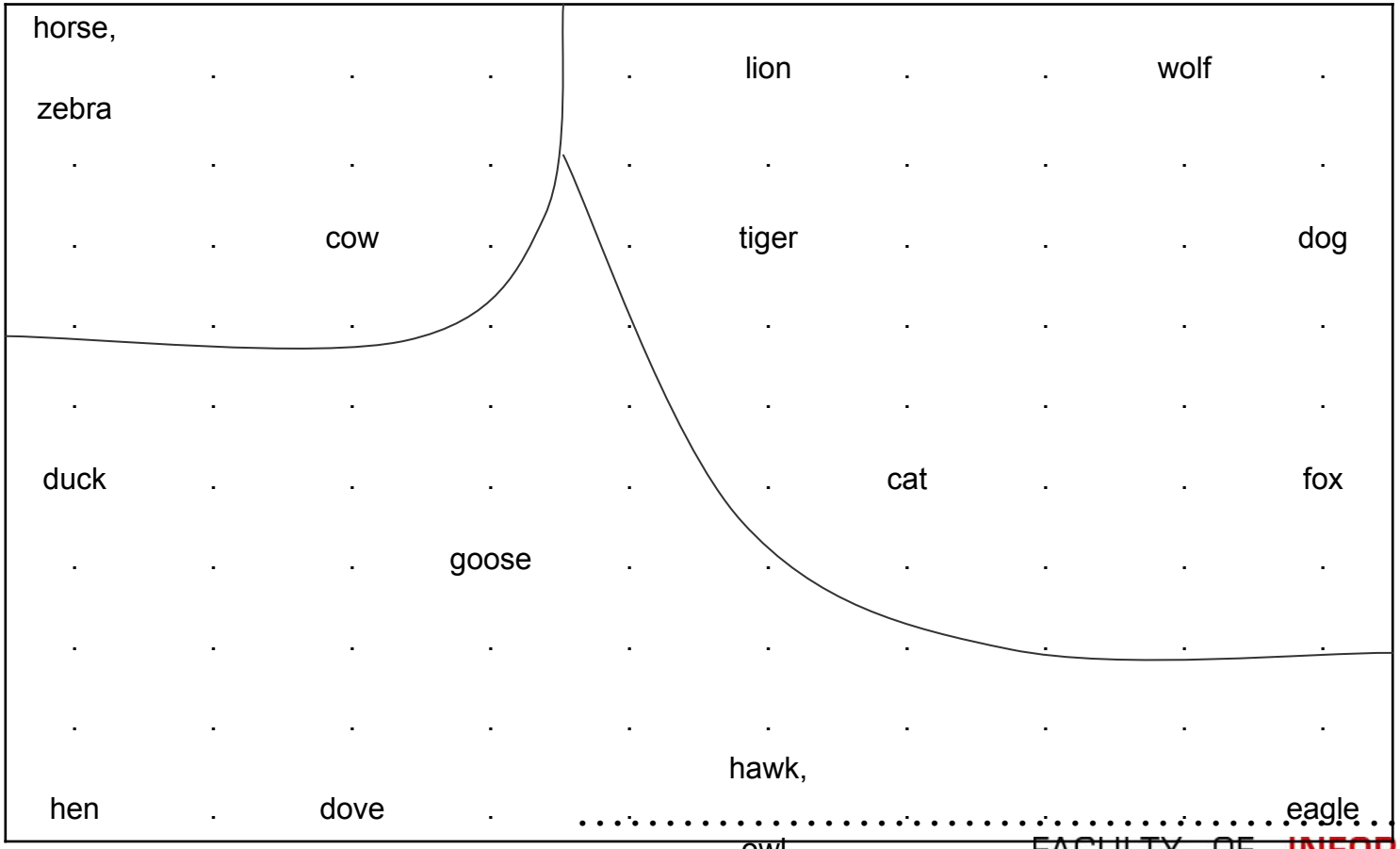
- 10x10 SOM



Self-Organizing Map

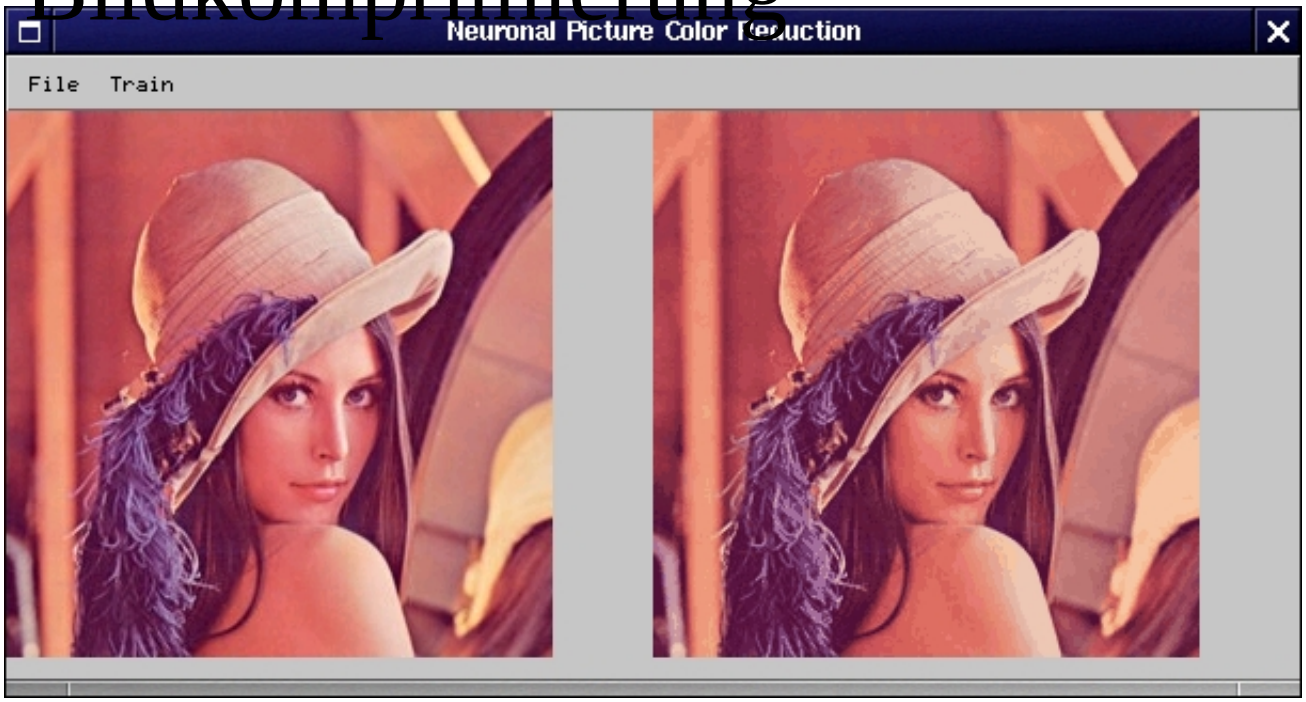
Beispiel: Tiere

- 10x10 SOM



Self-Organizing Map

Beispiel: Bildkomprimierung



2000 cycles

Self-Organizing Map

Beispiel: Bildkomprimierung

2700 cycles

Self Organizing Map

- Vorteile der Standard SOM
  - extrem stabil
  - verarbeitet sehr hochdimensionale Datenräume
  - intuitive Interpretation (Visualisierungen)
- Nachteile der Standard SOM
  - fixe Größe
  - keine hierarchischen Strukturen
  - statische Rechteckform
  - Clusterstruktur nur aus Visualisierungen ableitbar

Outline

- SOM
- Verwandte Verfahren
  - Incremental Grid Growing
  - Growing Grid
  - Growing Cell Structures
  - Hierarchical Feature Maps
  - Growing Hierarchical SOM (GHSOM)
  - Mnemonic SOM