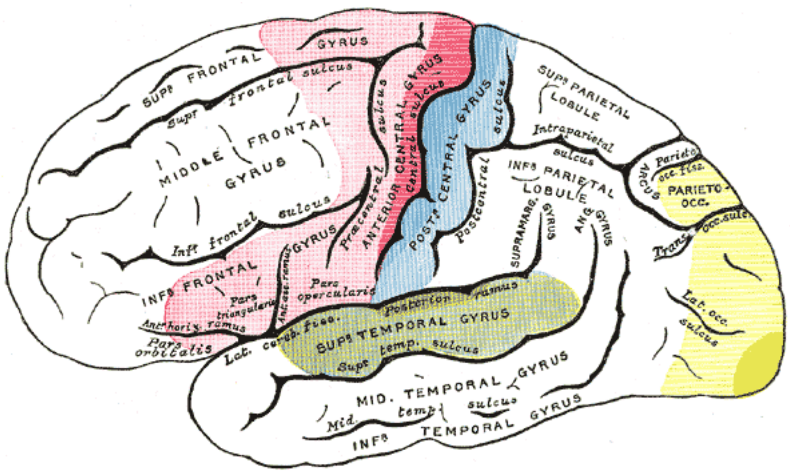
Kapitel DM:III

III. Nearest Neighbor Strategies

- Self Organizing Maps

Self Organizing Maps

Motivation: Räumliche Organisation von Aktivitäten im Gehirn



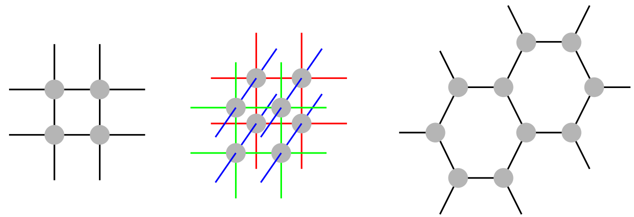
- primary motoric area
- pre/supplemental motoric areas
- primary sensible areas
- sensible association areas
- auditory sensation areas
- visually sensation areas

- Strukturen im Gehirn oft linear oder planar, Signale multidimensional.
- Ähnliche Reize werden durch räumlich nahe Nervenzellen verarbeitet.
- Stärker genutzte Bereiche sind besser ausgebildet.

Self Organizing Maps

Idee [Kohonen 1982]

- Neuronen bilden die Knoten einer Gitterstruktur (typisch 2D, 3D), Kanten beschreiben Nachbarschaften.



- Abbildung von Eingangssignalen auf Erregungszustände von Neuronen. (Alle Neuronen im Gitter sind mit allen Inputknoten verbunden.)

- Ähnliche Eingangssignale erregen „räumlich benachbarte“ Neuronen.

- Unüberwachte Anpassung an Eingangssignale nach dem Prinzip der lateralen Inhibition (seitliche Hemmung):
  Erregungszustände von stark erregten Neuronen und deren unmittelbarer Nachbarschaft werden verstärkt, Erregungszustände entfernterer Neuronen werden gedämpft.

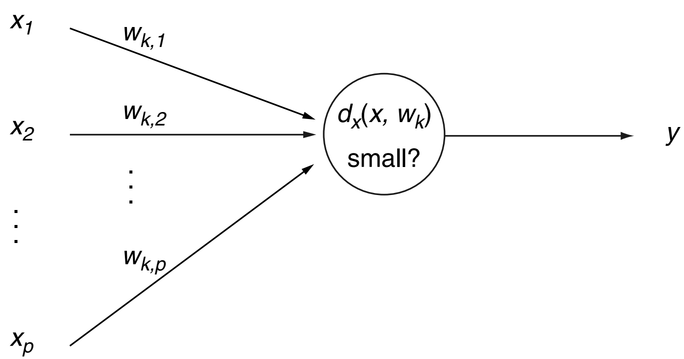
➜ Kartierung des Merkmalsraumes: Selbstorganisierende Karten (Self-Organizing Feature Map, SOM)

Self Organizing Maps

Formales Modell

- $D := \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subseteq X = \mathbf{R}^p$ endliche Menge von Eingabestimuli (Merkmalsvektoren).
- $d_X : \mathbf{R}^p \times \mathbf{R}^p \to \mathbf{R}^+$ Metrik auf $\mathbf{R}^p$.

  Beispiel: Euklidischer Abstand $d(x, y) = ||x - y|| = \sqrt{\sum_{i=1}^{p}(x_i - y_i)^2}$
- $N = \{N_1, \ldots, N_K\}$ endliche Menge von Neuronen, N_k definiert durch Gewichtsvektoren $\mathbf{w}_k \in \mathbf{R}^p$.



- $d_N : N \times N \to \mathbf{R}^+$ Metrik auf N (Nachbarschaft).

  Beispiel: Neuronen aus N liegen auf zweidimensionalen Gitter, d_N bestimmt den Euklidischen Abstand der Gitterpositionen oder die Manhattan-Distanz.

Remarks:

- Die Beispiele können auch zufällig aus dem Merkmalsraum gezogen werden nach einer festen Wahrscheinlichkeitsverteilung.

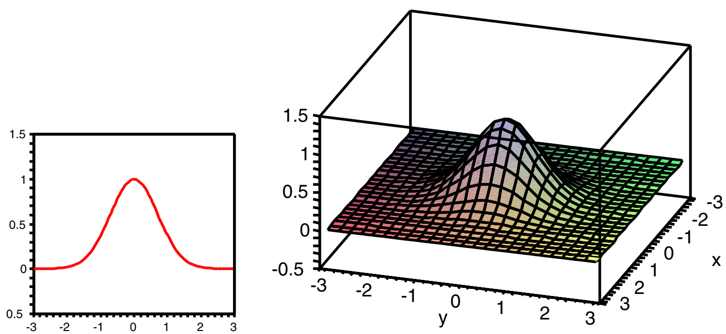
Self Organizing Maps

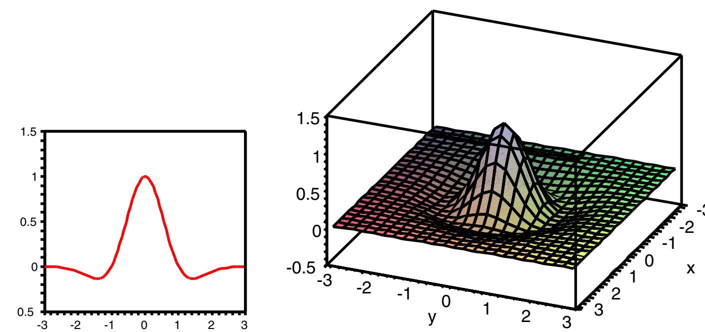
Formales Modell (Fortsetzung)

- $h(d, \sigma)$ Nachbarschaftsfunktion zur Realisierung der lateralen Inhibition: maximmal für $d = 0$, monoton fallend.

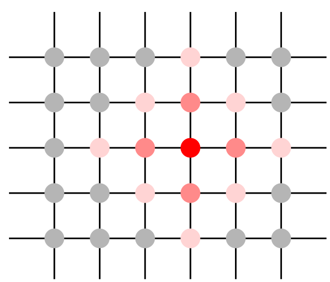
  Beispiel: $h(d, \sigma) = \exp(-\frac{d^2}{2\sigma^2})$ (Gaußglocke mit Radius σ um 0) oder

  $$h(d, \sigma) = \frac{1}{\sqrt{2\pi}\sigma^3}(1 - \frac{d^2}{\sigma^2})\exp(-\frac{d^2}{2\sigma^2})$$ (Mexican Hat Function).





  Anwendung auf Gitter:



Self Organizing Maps

Formales Modell (Fortsetzung)

- $\sigma > 0$ bestimmt die „enge“ Nachbarschaft.

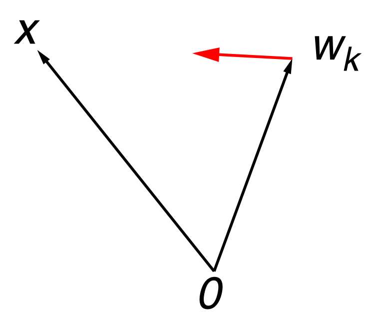
  Beispiel: σ kann abhängig von der Runde t gewählt werden als $\sigma(t) = \sigma_a \left(\frac{\sigma_e}{\sigma_a}\right)^{\frac{t}{t_{max}}}$ mit Anfangswert $\sigma_a > 0$ und Endwert σ_e mit $0 < \sigma_e \leq \sigma_a$.

- $\eta > 0$ bestimmt die Lernrate (typisch $\eta \in [0, 1]$).

  Beispiel: η kann abhängig von der Runde t gewählt werden als $\eta(t) = \eta_a \left(\frac{\eta_e}{\eta_a}\right)^{\frac{t}{t_{max}}}$ mit Anfangswert $\eta_a > 0$ und Endwert η_e mit $0 < \eta_e \leq \eta_a$.

- $k_0(t) = \text{argmin}_{k=1,\ldots,K}\, d_X(\mathbf{x}, \mathbf{w}_k(t))$ bestimmt Index des Neuron mit dem geringsten „Abstand“ zum Eingangsstimulus $\mathbf{x}$.

- $\Delta\mathbf{w}_k = \eta(t) \cdot h(d_N(N_k, N_{k_0}), \sigma(t)) \cdot (\mathbf{x} - \mathbf{w}_k)$ Anpassung der Gewichte aller Neuronen.



Remarks:

- Gebräuchlich ist auch die sogenannte Bubble-Neigborhood, die Neuronen innerhalb eines gegebenen Radius σ mit 1 bewertet und alle anderen mit 0.
- Eine Anpassung der Nachbarschaftsfunktion h an die Rundenzahl t erfolgt indirekt durch den Parameter $\sigma(t)$ in $h(d, \sigma(t))$.

Self Organizing Maps

Algorithmus zur Gewichtsanpassung

Sei D eine Menge von Trainingsbeispielen, η eine positive kleine Konstante, die Lernrate, σ eine positive Konstante, der Nachbarschaftsradius, und p die Dimension der Merkmalsvektoren.

som_training(D, η, σ)

initialize_random_weights$(\mathbf{w}_1, \ldots, \mathbf{w}_K)$, $t = 0$;
REPEAT
    $t = t + 1$, $\mathbf{x} =$ *random_select*(D);
    $k_0 = \text{argmin}_{k=1,\ldots,K}\, d_X(\mathbf{x}, \mathbf{w}_k)$;
    FOR $k = 1$ **TO** K **DO**
        FOR $i = 0$ **TO** p **DO**
            $\Delta w_{k,i} = \eta \cdot h(d_N(N_k, N_{k_0}), \sigma) \cdot (x_i - w_{k,i})$;
            $w_{k,i} = w_{k,i} + \Delta w_{k,i}$;
        ENDDO
    ENDDO
UNTIL$(t > t_{\max})$;

Remarks:

- Die Initialisierung der Gewichte der Neuronen kann mit kleinen Zufallswerten erfolgen.
- Die Initialisierung der Gewichte der Neuronen kann mit zufällig gezogenen Elementen der Trainingsmenge erfolgen.
- Die Initialisierung der Gewichte der Neuronen kann durch Belegung mit linear geordneten Werten des durch die beiden größten Eigenwerte der Matrix der Trainingsmenge aufgespannten Teilraumes erfolgen.

Self Organizing Maps

Algorithmus zur Gewichtsanpassung

Natürlichsprachliche Formulierung:

1. Initialisiere die Gewichte der Neuronen in der SOM.
2. Wähle zufällig einen Eingabestimulus $\mathbf{x}$ aus D.
3. Bestimme das aktuelle Erregungszentrum N_{k_0} (Neuron mit ähnlichstem Gewichtsvektor).
4. Passe den Gewichtsvektor des Erregungszentrums und seiner Nachbarschaft mit nach außen abnehmender Stärke an den Eingabestimulus an.

$$\mathbf{w}_k = \mathbf{w}_k + \eta \cdot h(d_N(N_k, N_{k_0}), \sigma) \cdot (\mathbf{x} - \mathbf{w}_k)$$

5. (Reduziere Lernrate η und Nachbarschaftsgröße σ.)
6. Falls Trainingsphase nicht zu Ende, gehe zu Schritt 2.

Self Organizing Maps

Beispiel: Dimensionsreduktion mit SOMs

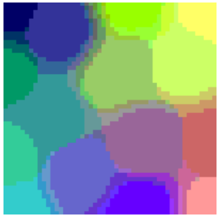
- 3-dimensionaler Integer-Merkmalsraum: $[0,5] \times [0,5] \times [0,5]$ dargestellt durch Farben ($6^3 = 216$ Elemente).
- 15 Eingabevektoren werden gleichverteilt zufällig gezogen.
- SOM mit zweidimensionalem Gitter mit 50×50 Neuronen.
- Initialisierung der Neuronen mit
  - Zufallszahlen oder
  - Farbverlauf mit Farben rot, gelb, grün, schwarz in Ecken oder
  - Farbverlauf mit drei Zentren rot, gelb grün.
- Abbildung Gewichte auf Farben durch Rundung (Farben entsprechen Größenordnung der Gewichte).

Self Organizing Maps

Beispiel: Dimensionsreduktion mit SOMs

- 3-dimensionaler Integer-Merkmalsraum: $[0,5] \times [0,5] \times [0,5]$ dargestellt durch Farben ($6^3 = 216$ Elemente).
- 15 Eingabevektoren werden gleichverteilt zufällig gezogen.
- SOM mit zweidimensionalem Gitter mit 50×50 Neuronen.
- Initialisierung der Neuronen mit
  - Zufallszahlen oder
  - Farbverlauf mit Farben rot, gelb, grün, schwarz in Ecken oder
  - Farbverlauf mit drei Zentren rot, gelb grün.
- Abbildung Gewichte auf Farben durch Rundung (Farben entsprechen Größenordnung der Gewichte).

➜ Ausbildung von Regionen in der SOM für die einzelnen Beispiele.

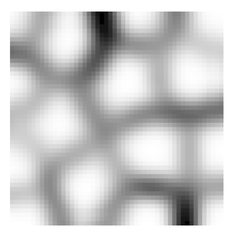


Self Organizing Maps

Beispiel: Dimensionsreduktion mit SOMs (Fortsetzung)

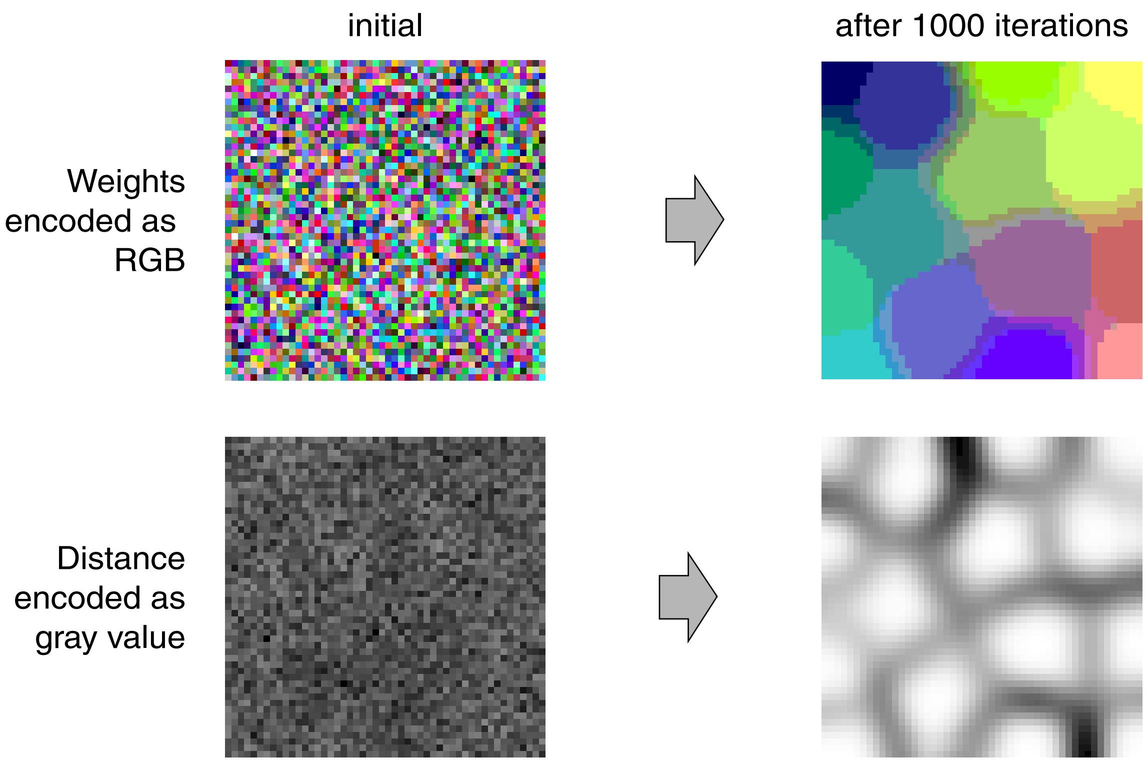
- Cluster werden gebildet von Vektoren, die nahe beieinander liegen im Vergleich zu allen übrigen Vektoren.
- Jedes Neuron im Gitter der SOM repräsentiert einen Bereich des Eingaberaumes.
  Das Gewicht des Neuron kann als Repräsentant dieses Bereiches aufgefasst werden.
- Ähnlichkeit von Gewichten zwischen benachbarten Neuronen kann visualisiert werden:
  Grauwert steht für durchschnittliche Distanz zu unmittelbaren Nachbarn.

➜ Visualisierung zeigt Schärfe der Trennung von benachbarten Clustern: Uniform Distance Matrix (U-Matrix).



Self Organizing Maps

Beispiel: Dimensionsreduktion mit SOMs (Fortsetzung)



[Matthew Ward, WPI]

http://davis.wpi.edu/∼matt/courses/soms//applet.html

Self Organizing Maps

Beispiel: Spezialfall 2D-SOMs

- 2-dimensionale Merkmalsvektoren, also $D \subset \mathbf{R}^2$:
  Eingabestimuli bezeichenen Positionen in der Ebene.
- 2-dimensionale Gewichtsvektoren können ebenfalls als Positionen in der Ebene aufgefasst werden:
  durch Neuronen repräsentierter Eingaberaum unmittelbar erkennbar.
- Unmittelbare Nachbarschaftsbeziehungen zwischen Neuronen werden durch Linien gekennzeichnet:
  SOM liegt als Gitternetz auf der Ebene des Eingaberaumes.

Self Organizing Maps

Beispiel: Spezialfall 2D-SOMs

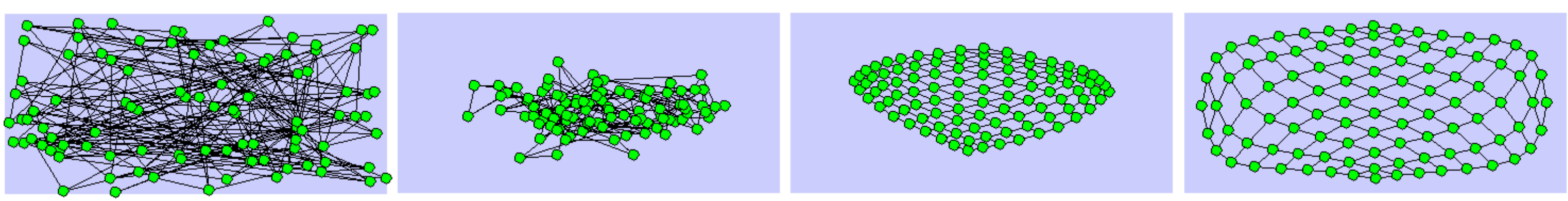
- 2-dimensionale Merkmalsvektoren, also $D \subset \mathbf{R}^2$:
  Eingabestimuli bezeichenen Positionen in der Ebene.

- 2-dimensionale Gewichtsvektoren können ebenfalls als Positionen in der Ebene aufgefasst werden:
  durch Neuronen repräsentierter Eingaberaum unmittelbar erkennbar.

- Unmittelbare Nachbarschaftsbeziehungen zwischen Neuronen werden durch Linien gekennzeichnet:
  SOM liegt als Gitternetz auf der Ebene des Eingaberaumes.

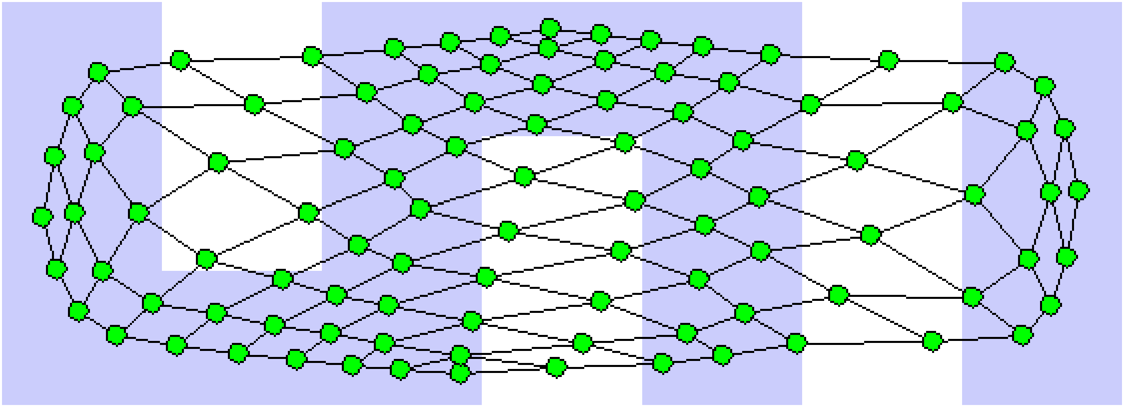
➜ Erregungszentrum der SOM leicht erkennbar. Gewichtsanpassung erscheint als Ziehen am Netzknoten in Richtung Eingabestimulus.

➜ Netz entfaltet sich und passt sich an Eingaberaum an. (vgl. Spring Embedder)



Self Organizing Maps

Beispiel: Spezialfall 2D-SOMs (Fortsetzung)



[Bernd Fritzke] https://www.demogng.de/

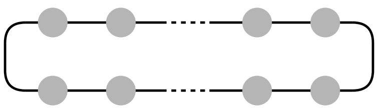
Self Organizing Maps

Anwendungsbeispiel: Traveling Salesman Problem (TSP)

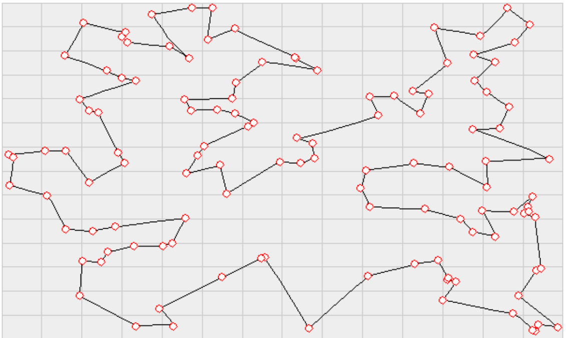
- Aufgabe: Verbinde n Städte mit einer kürzestmöglichen Rundtour.

  Positionen der Städte auf Landkarte fest, alle Luftlinien zwischen zwei Städten als Wege möglich.

- SOM-Lösungsansatz:
  Verwendung einer geschlossenen Kette von $n * k$ Neuronen als SOM.



- Positionen der Städte werden gleichverteilt zufällig als Eingabestimuli gezogen.



[Sven Börner]

Self Organizing Maps

Beispiel: Spezialfall Vektorquantisierung

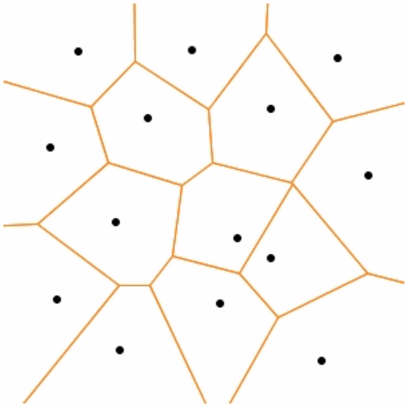
- Leere Nachbarschaftsrelation: Einzelne Neuronen ohne Verbindungen.

$$d_N(N_i, N_j) = \begin{cases} 1 & \text{für } i = j \\ 0 & \text{sonst} \end{cases}$$

- Anziehung für N_{k_0}, Abstoßung für alle andere Neuronen.

$$h(1, \sigma) = 1 \quad \text{und} \quad h(0, \sigma) = -1$$

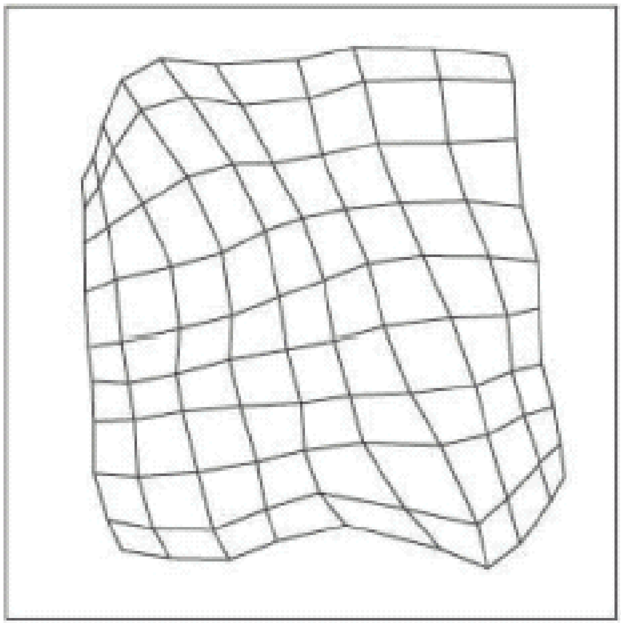
- Die Neuronen repräsentieren *Voronoi-Regionen* im Eingaberaum: N_k repräsentiert $\{\mathbf{x} \mid d_X(\mathbf{x}, \mathbf{w}_k) < d_X(\mathbf{x}, \mathbf{w}_{k'}), k' \neq k\}$ (Winner Takes It All). Vektoren des Eingaberaumes, die von N_k einen geringeren Abstand haben als von allen anderen Neuronen.

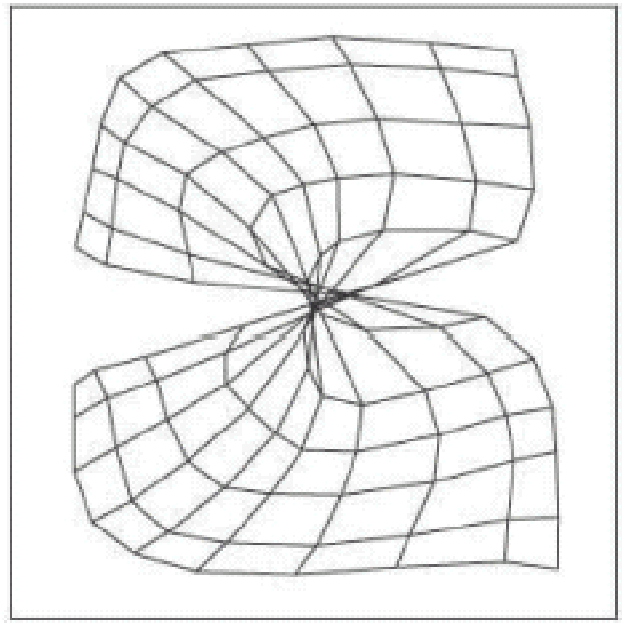


Neural Gas

Motivation

- SOMs realisieren eine räumliche Organisation von hochdimensionalen Eingangsdaten.
- Topologische Defekte





  - Topologie SOM passt nicht zur Topologie des Eingaberaumes.
  - Nachbarschaft wird nicht adäquat angepasst über $\sigma(t)$.
- Gehirn kann Strukturen selbständig ausbilden und anpassen.

Neural Gas

Idee [Martinetz/Schulten 1991]

- ❑ Abbildung von Eingangssignalen auf Erregungszustände von Neuronen wie beim SOM.
- ❑ Aber: Neuronen weisen keine vorgegebene Nachbarschaft auf.
- ❑ Durch ein Eingangssignal stark erregte Neuronen verbinden sich zu Nachbarschaften.
- ❑ Festigkeit einer Verbindung zwischen Neuronen nimmt mit der Zeit ab, Verbindung müssen immer wieder erneuert werden.
  (Siehe auch Hebbsches Lernen.)

➜ Struktur des Netzes bildet sich während des Lernvorgangs aus.
Bessere Anpassung an die Topologie des Merkmalsraumes.

- ❑ Erweiterung des Ansatzes: Growing Neural Gas
  Nicht nur Kanten, sondern auch Knoten können erzeugt und gelöscht werden.

Neural Gas

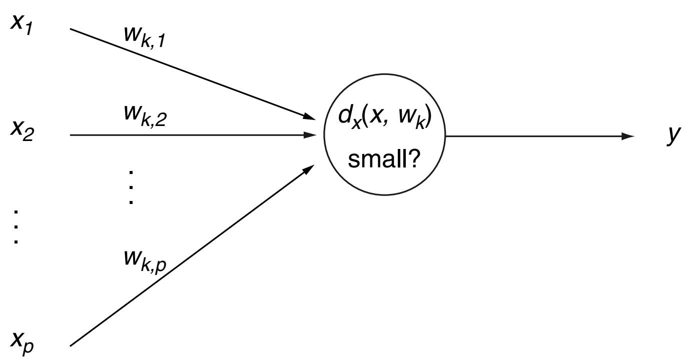
Formales Modell

- $D := \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subseteq X = \mathbf{R}^p$ endliche Menge von Eingabestimuli (Merkmalsvektoren).

- $d_X : \mathbf{R}^p \times \mathbf{R}^p \to \mathbf{R}^+$ Metrik auf $\mathbf{R}^p$.

  Beispiel: Euklidischer Abstand $d(x, y) = ||x - y|| = \sqrt{\sum_{i=1}^{p}(x_i - y_i)^2}$

- $N = \{N_1, \ldots, N_K\}$ endliche Menge von Neuronen, N_k definiert durch Gewichtsvektoren $\mathbf{w}_k \in \mathbf{R}^p$.



Neural Gas

Formales Modell (Fortsetzung)

- sort-index $: \{1, \ldots, K\} \times \mathbf{R}^K \to \{1, \ldots, K\}$

  Position in der sortierten Reihenfolge der K Werte für die angegebene Komponente im Ausgangstupel.

  Beispiel: Für $K = 7$ liefert sort-index$(2, (7.3, 6.1, 5.34, 1.02, 3.8, 4.21, 2.6)) = 6$.

  6.1 ist der 6.te Wert in dem aufsteigend sortierten Eingabetupel.

- $s_k :=$ sort-index$(k, (d_X(\mathbf{x}, \mathbf{w}_1), \ldots, d_X(\mathbf{x}, \mathbf{w}_K))$

  Reihenfolge der Indizes der Neuronen bezüglich des Abstands des Gewichtsvektors zu $\mathbf{x}$, beginnend beim kleinsten Abstand, d.h. s_k ist Position von N_i in Sortierung von N bzgl. Abstand von $\mathbf{x}$.
  (Ersatz für die Nachbarschaftsrelation der Neuronen)

Remarks:

- Alternative Definition der s_k:

$$s_k := |\{j \in \{1, \ldots, K\} : d_X(\mathbf{x}, \mathbf{w}_j) < d_X(\mathbf{x}, \mathbf{w}_k)\}|$$

(verschiedene Abstände von $\mathbf{x}$ für alle Neuronen vorausgesetzt).

Neural Gas

Formales Modell (Fortsetzung)

- $h(d, \sigma)$ Nachbarschaftsfunktion zur Realisierung der lateralen Inhibition: maximmal für $d = 0$, monoton fallend.

  Beispiel: Übliche Wahl $h(d, \sigma) = \exp(-\frac{d}{\sigma})$.

- $\sigma > 0$ bestimmt die „enge“ Nachbarschaft.

  Beispiel: σ kann abhängig von der Runde t gewählt werden als $\sigma(t) = \sigma_a \left(\frac{\sigma_e}{\sigma_a}\right)^{\frac{t}{t_{max}}}$ mit Anfangswert $\sigma_a > 0$ und Endwert σ_e mit $0 < \sigma_e \leq \sigma_a$.

- $\eta > 0$ bestimmt die Lernrate (typisch $\eta \in [0, 1]$).

  Beispiel: η kann abhängig von der Runde t gewählt werden als $\eta(t) = \eta_a \left(\frac{\eta_e}{\eta_a}\right)^{\frac{t}{t_{max}}}$ mit Anfangswert $\eta_a > 0$ und Endwert η_e mit $0 < \eta_e \leq \eta_a$.

- $\Delta \mathbf{w}_k = \eta(t) \cdot h(s_k, \sigma(t)) \cdot (\mathbf{x} - \mathbf{w}_k)$ Anpassung der Gewichte aller Neuronen.

Neural Gas

Formales Modell (Fortsetzung)

- *Connect* Verbindungsmatrix für N: quadratische $K \times K$ Matrix.
  Connect$(k_1, k_2) > 0$ gibt an, dass eine Verbindung zwischen N_{k_1} und N_{k_2} besteht und seit wieviel Runden sie besteht.
  Connect$(k_1, k_2) = 0$ gibt an, dass keine Verbindung zwischen N_{k_1} und N_{k_2} existiert.

- $\tau > 0$ bestimmt das Alter, bei dem eine Verbindung in *Connect* gelöscht wird.

  Beispiel: τ kann abhängig von der Runde t gewählt werden als $\tau(t) = \tau_a \left(\frac{\tau_e}{\tau_a}\right)^{\frac{t}{t_{max}}}$ mit Anfangswert $\tau_a > 0$ und Endwert τ_e mit $0 < \tau_e \leq \tau_a$.

- Neue Verbindungen werden zwischen den beiden Neuronen erzeugt, die dem Eingabewert am nächsten liegen.

- Verbindungen, die älter als τ sind, werden gelöscht.

Neural Gas

Algorithmus zur Gewichts- und Strukturanpassung

Sei D eine Menge von Trainingsbeispielen, η eine positive kleine Konstante, die Lernrate, σ eine positive Konstante, der Nachbarschaftsradius, und p die Dimension der Merkmalsvektoren.

neural_gas_training(D, η, σ)

initialize_random_weights$(\mathbf{w}_1, \ldots, \mathbf{w}_k)$, $t = 0$;
REPEAT
    $t = t + 1$, $\mathbf{x} =$ *random_select*(D);
    FOR $k = 1$ **TO** K **DO**
        FOR $i = 0$ **TO** p **DO**
            $\Delta w_{k,i} = \eta \cdot h(s_k, \sigma) \cdot (x_i - w_{k,i})$;
            $w_{k,i} = w_{k,i} + \Delta w_{k,i}$;
        ENDDO
    ENDDO
    Connect$(s_1, s_2) = 1$;
    FOREACH $(i, j) \in \{1, \ldots, K\}^2, i \neq j$ **DO**
        IF *Connect*$(i,j) > 0$ **THEN** *Connect*$(i,j) =$ *Connect*$(i,j) + 1$;
        IF *Connect*$(i,j) > \tau$ **THEN** *Connect*$(i,j) = 0$;
    ENDDO
UNTIL$(t > t_{\max})$;

Neural Gas

Algorithmus zur Gewichtsanpassung

Natürlichsprachliche Formulierung:

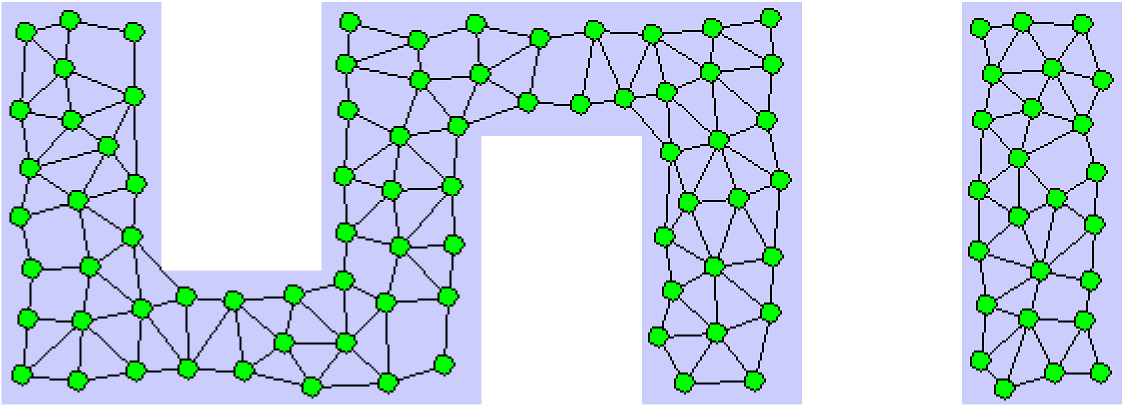
1. Initialisiere die Gewichte der Neuronen in der SOM.
2. Wähle zufällig einen Eingabestimulus $\mathbf{x}$ aus D.
3. Bestimme die Reihenfolge $s_1, \ldots, s_K$ der Neuronen nach den Abständen von $\mathbf{x}$.
4. Passe den Gewichtsvektor des Erregungszentrums und seiner Nachbarschaft mit nach außen abnehmender Stärke an den Eingabestimulus an.

   $$\mathbf{w}_k = \mathbf{w}_k + \eta \cdot h(s_k, \sigma) \cdot (x - w_k)$$

5. Bilde neue Nachbarschaft zwischen den beiden $\mathbf{x}$ am nächsten liegenden Neuronen.
6. Inkrementiere Alter der Kanten und lösche zu alte Kanten.
7. (Reduziere Lernrate η und Nachbarschaftsgröße σ, erhöhe τ.)
8. Falls Trainingsphase nicht zu Ende, gehe zu Schritt 2.

Neural Gas

Beispiel: Spezialfall Neuronales Gas in 2D



[Bernd Fritzke] https://www.demogng.de/