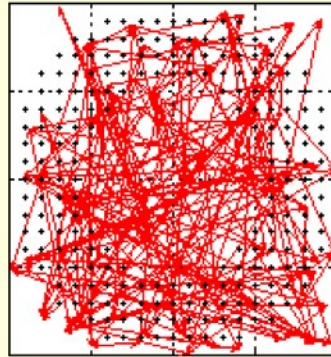


~~Growing self-organizing maps~~

SOM salabim und andere Datenmodelle



Marc Strickert

II. Phys. Institut, Arbeitsgruppe Düren, FB 07, JLU-Gießen

Mini Workshop: **Neural Networks for Anomaly Detection in Physics**

marc.strickert@exp2.physik.uni-giessen.de



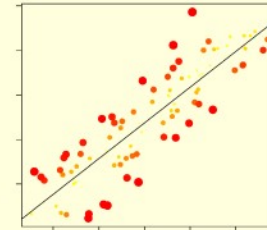
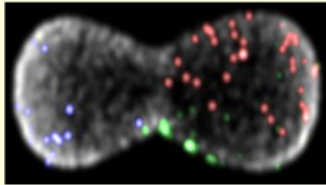
21 Feb 2020



Data Analysis Goals

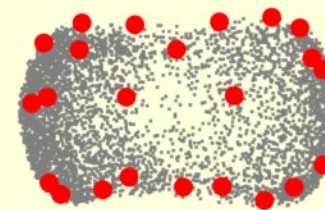
1. Daten-Inspektion & -Vorverarbeitung
2. Clustering (SOM extensions)
3. Classification
4. Visualization of high-dimensional data

Typische Aufgaben der Datenanalyse



1. Attribute rating

(supervised)

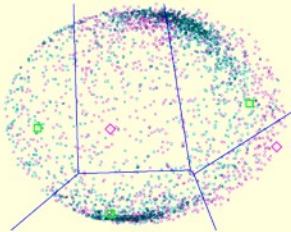


2. Clustering

(Neural Gas, LBG, k-means)

4. Visualisierung

(SOM / MDS)

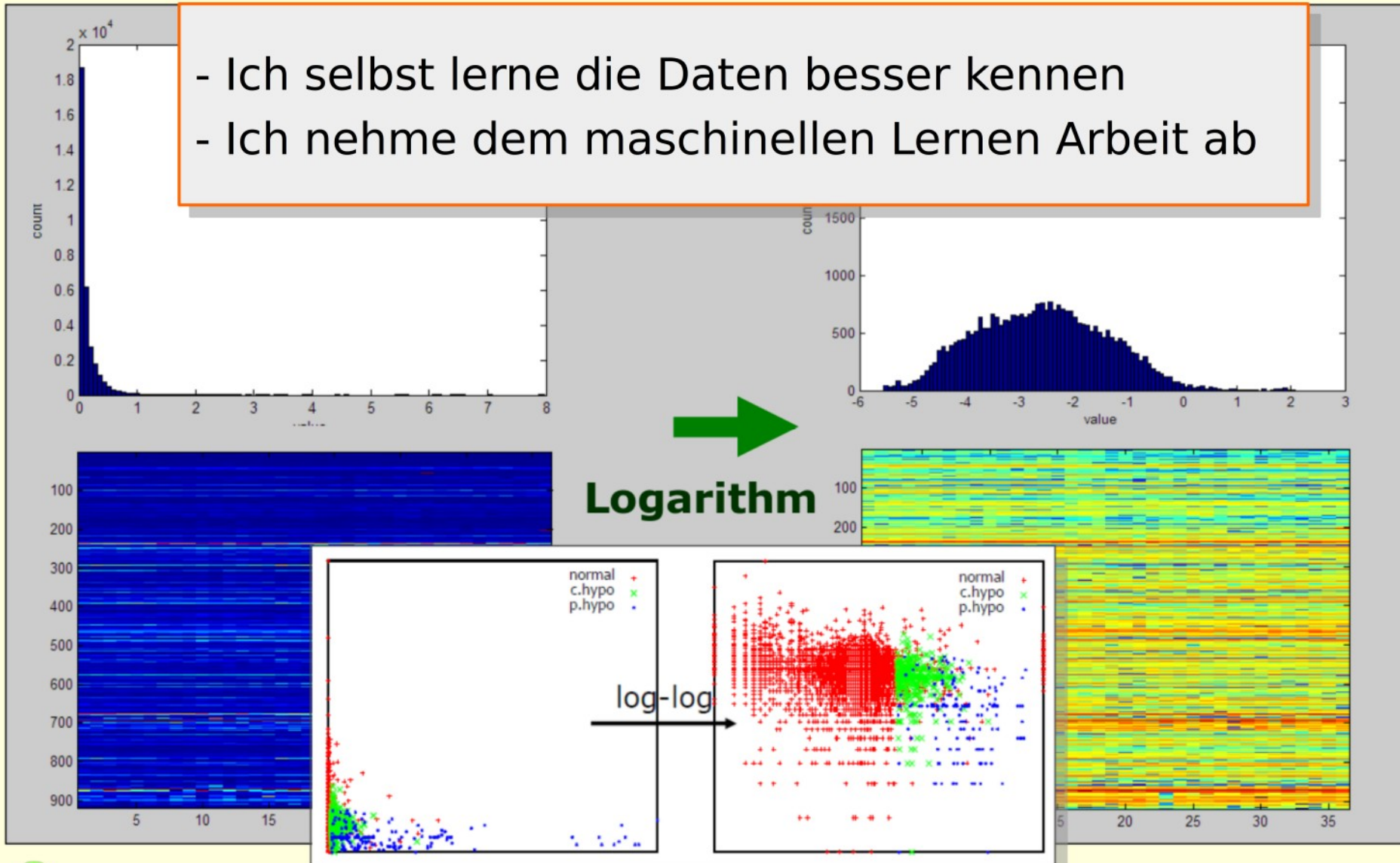


3. Klassifikation

(Lernende Vektorquantisierung LVQ)

Dateninspektion und -Vorverarbeitung

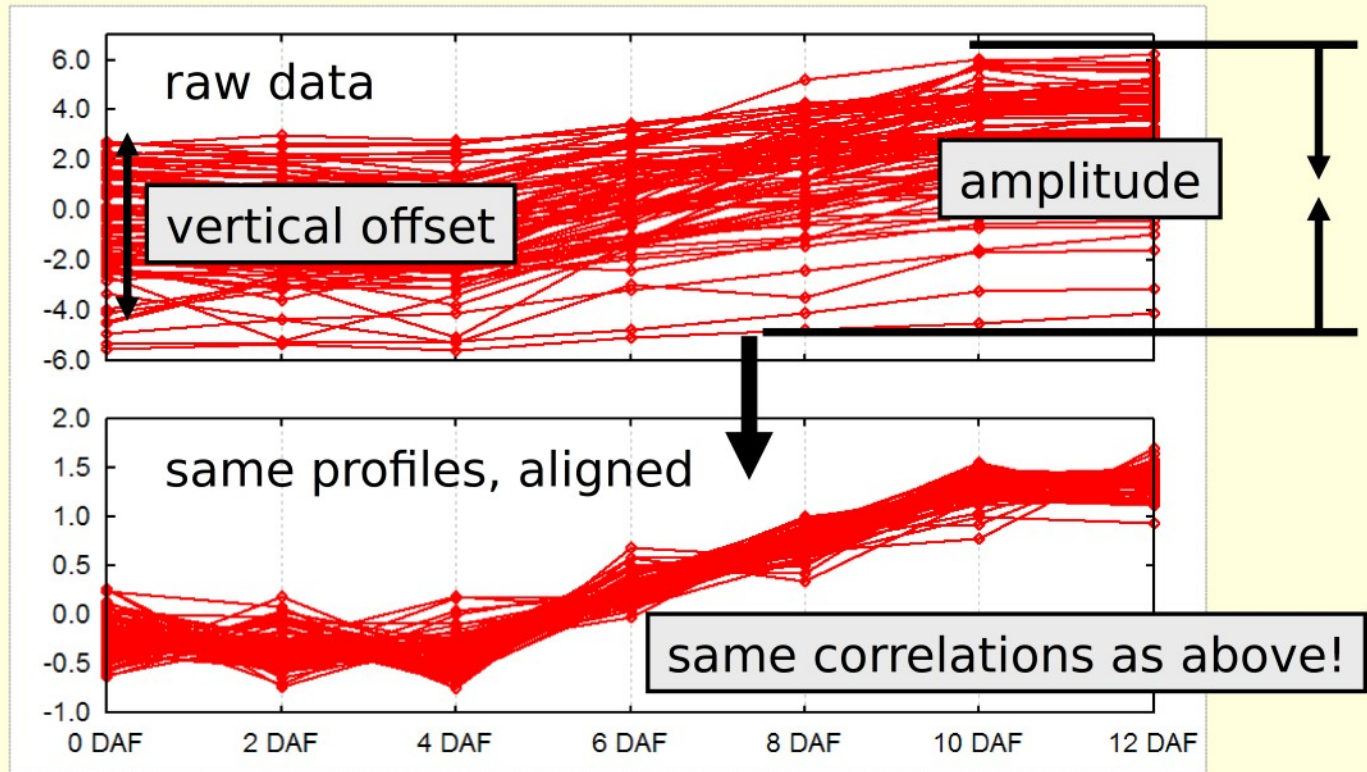
- Ich selbst lerne die Daten besser kennen
- Ich nehme dem maschinellen Lernen Arbeit ab



Datenvorverarbeitung je nach Datenmetrik

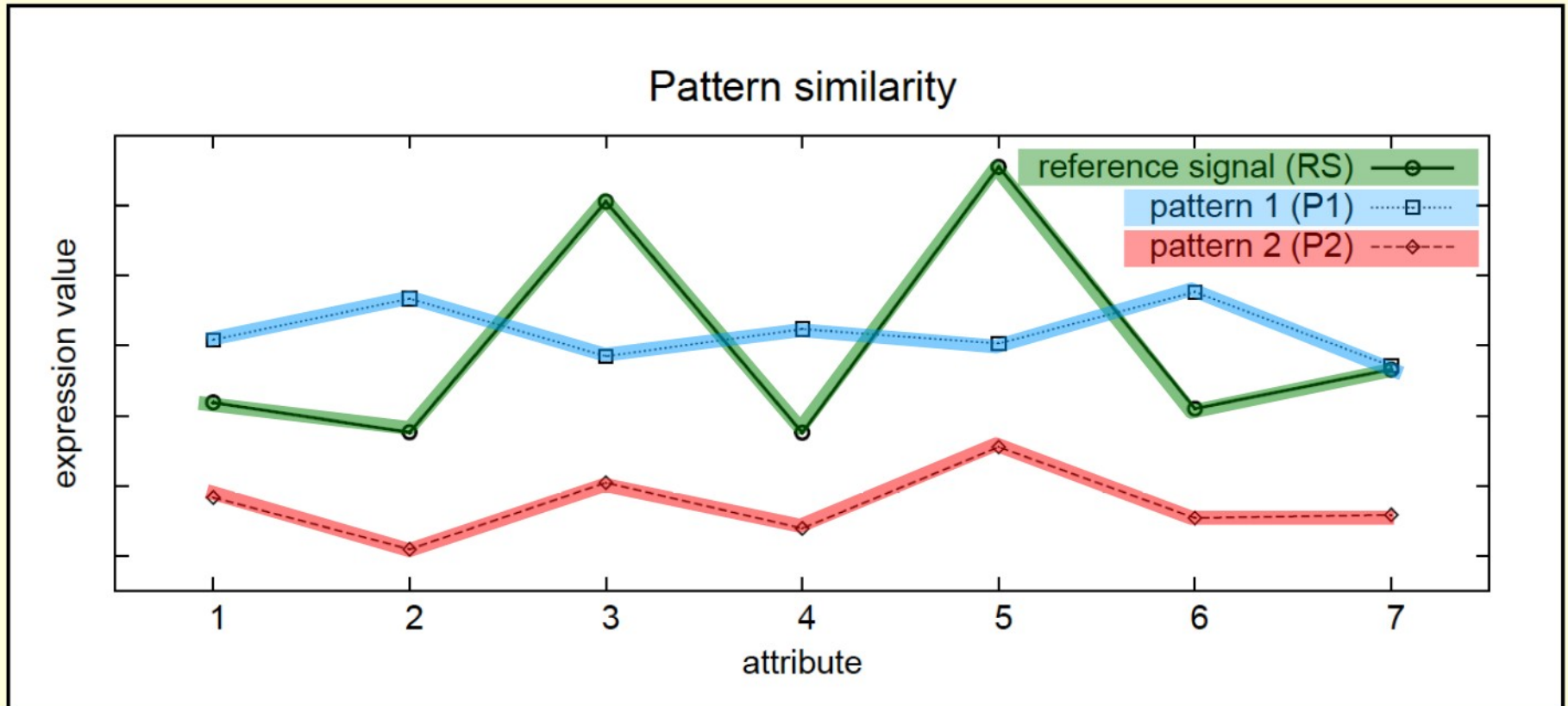
Up-regulated gene profiles

Euclidean
view



'Pearson'
view

Metrik vs. Ähnlichkeit

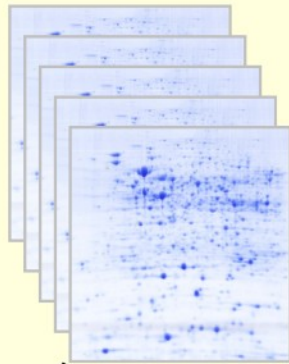


Euclidean distance: **P1** closer to RS.

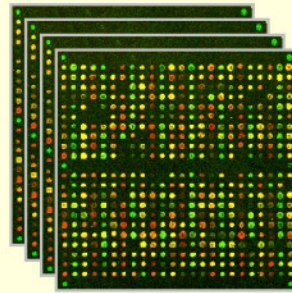
Pearson correlation: **P2** closer to RS.

Fluch der hohen Dimensionalität (curse of dimension)

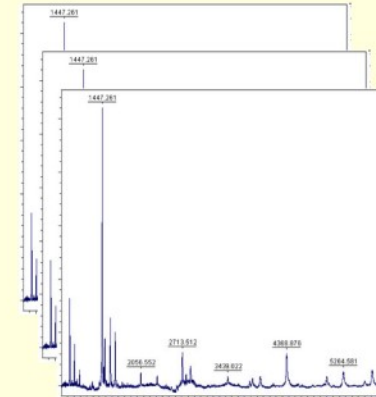
high-dimensional
experiments



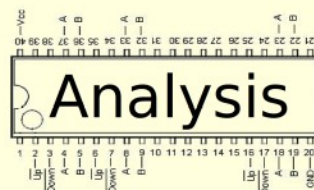
images



microarrays



spectra



few entities (~100)

many attributes (~10,000)

low-dimensional
data items



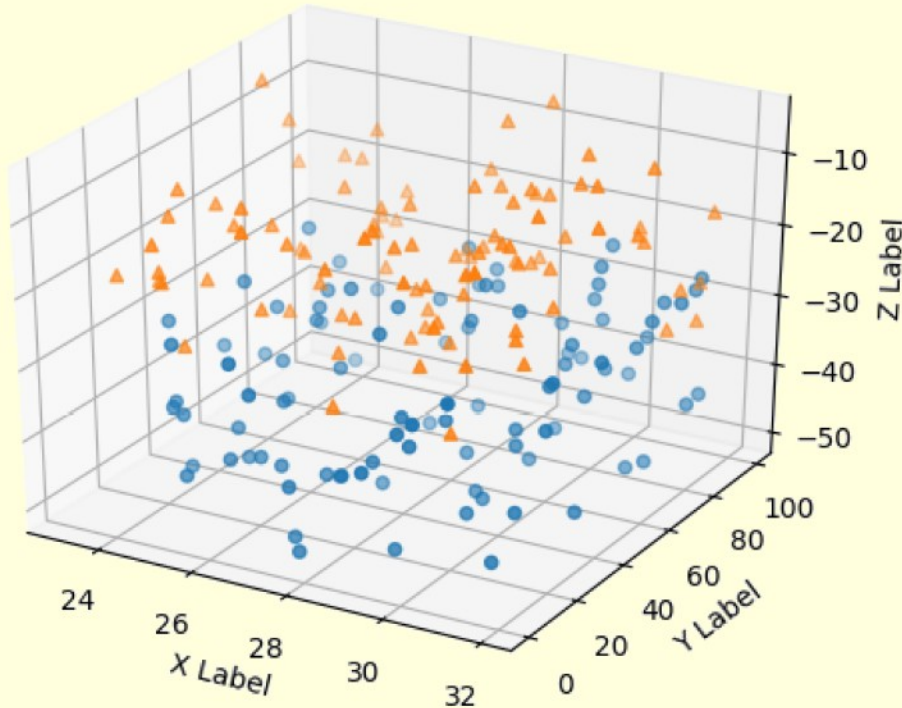
spreadsheet data

→ Curse of dimension!

← Grouping of data?
Candidate attributes?

2/3-D Daten führen in die Irre

Was ist das Innere einer Datenwolke?



Hochdimensionale Daten leben auf dem Rand und sind ,immer' Ausreißer in irgendwelchen Dimensionen!
→ Datenwolken ,sehen aus' wie ein Stern.

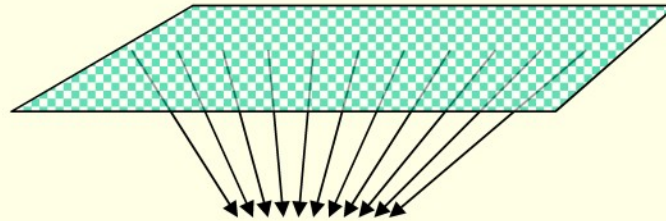
1. SOM (unüberwacht)

„What fires together
that wires together.“

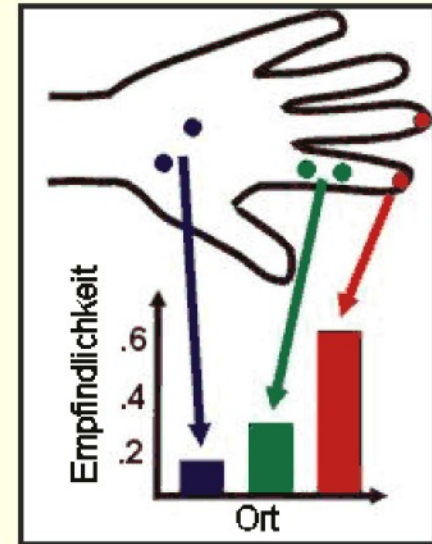
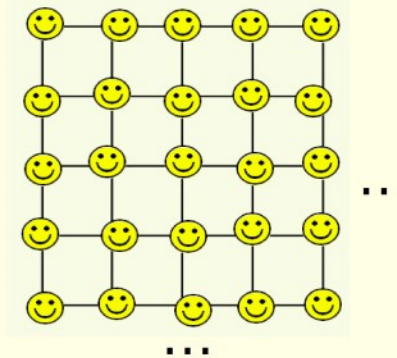
D. Hebb (1949)

SOM: Kortikale topologie-erhaltende Abbildung

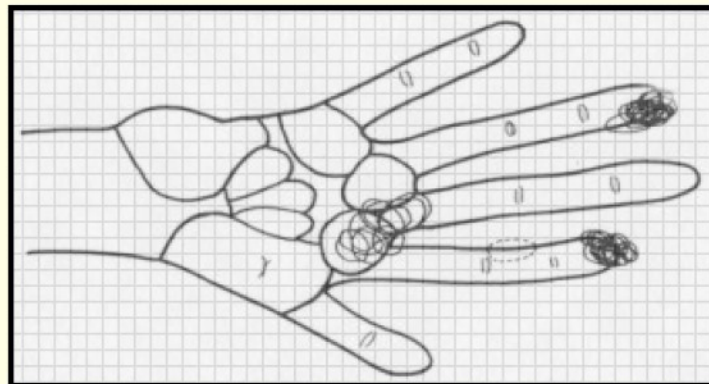
Tastzellen einer Hand.



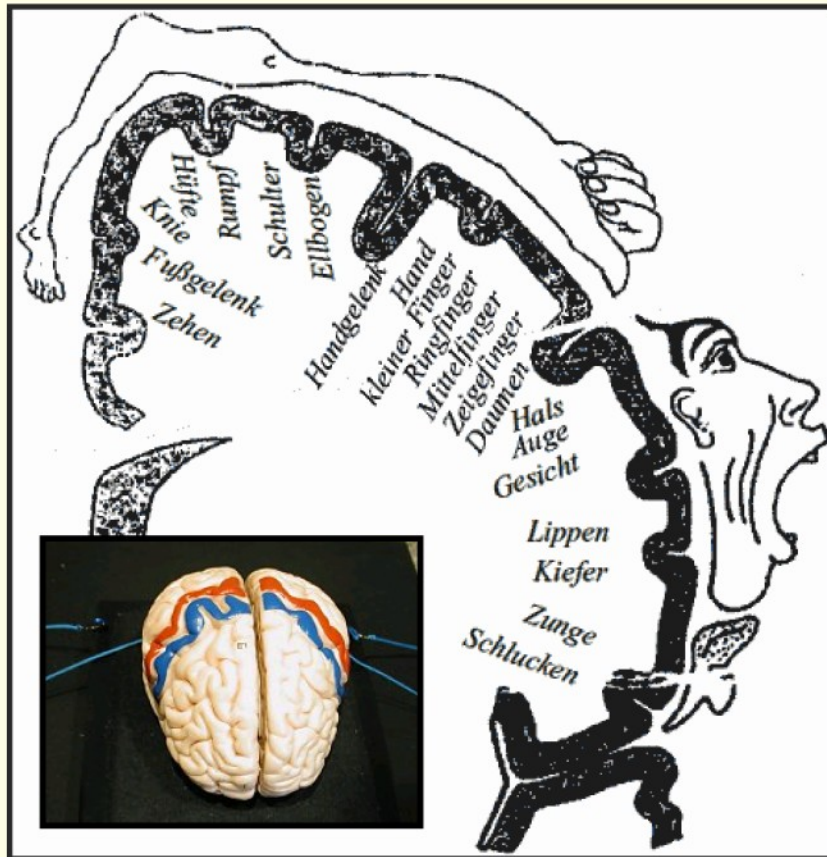
Verarbeitung in stark schematischem Kortex.



Spezialisierte kortikale Karte der Reize.



Wir sind Homunkulus



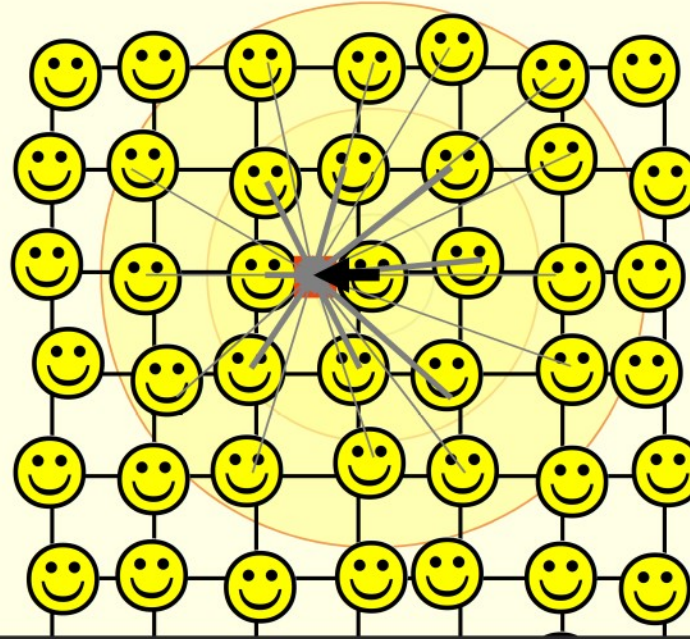
Somato-sensomotrische Karte
nach Wilder Penfield und Theodore Rasmussen

Unüberwacht: Self-Organizing Map (**SOM**)

Map: Abbildung **Eingabereize** → **Neuronen-Ensembles**

Ein Neuron hat:

1. Position im Gitter und Nachbarn.
2. Inneren Zustand w (**Prototyp**).



w s und \blacksquare im
(hochdim.)
Eingaberaum,
„Reiz“.

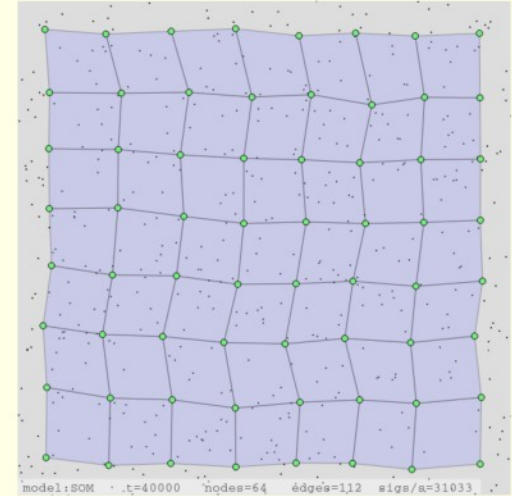
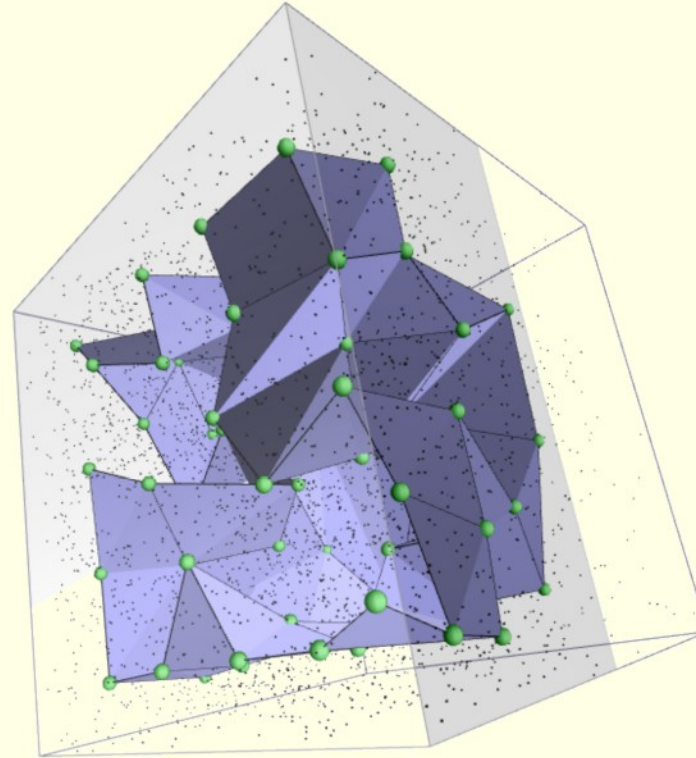
Minimierung der mittleren Prototypen-Unähnlichkeit

→ **Flächig spezialisierte Neuronenverbände.**

→ **Dimensionsreduktion.**

[Kohonen, 1983]

SOM-Beispiel

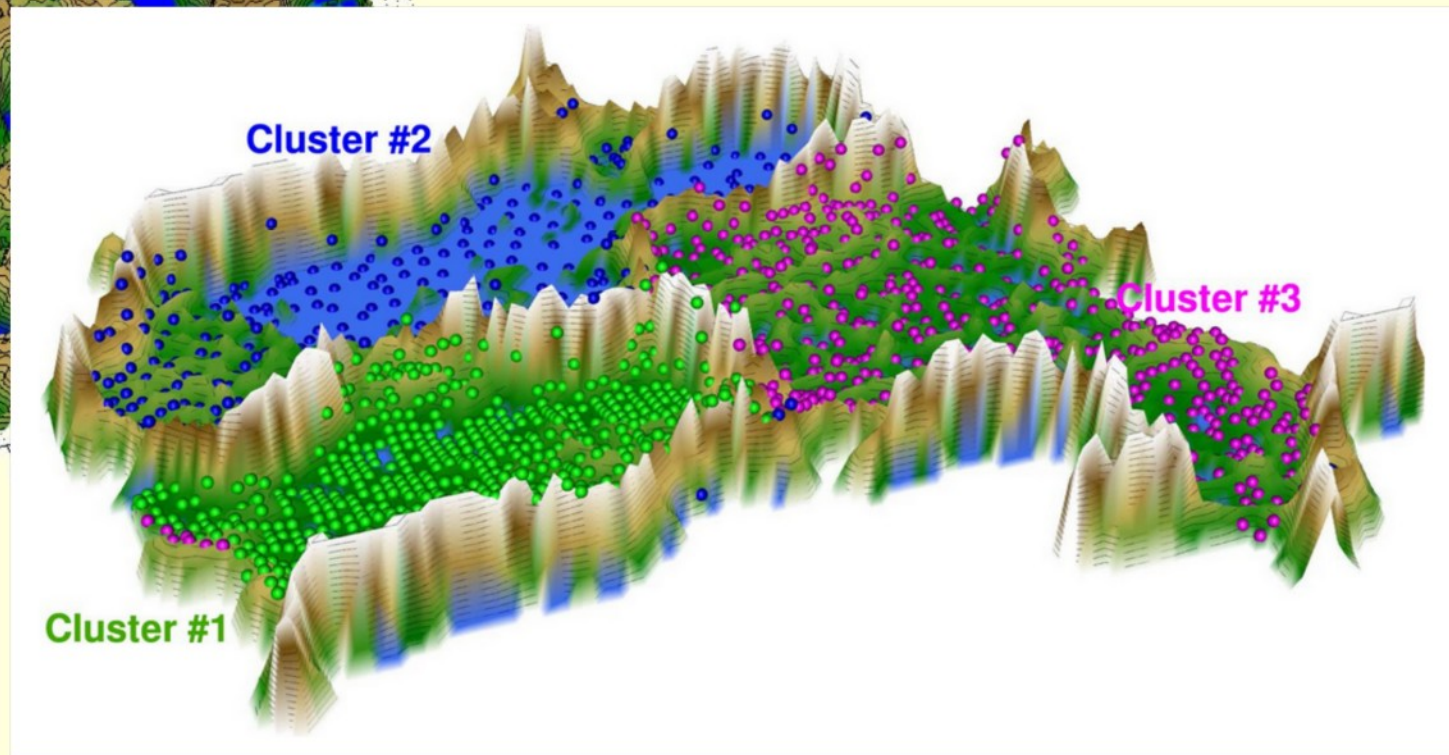
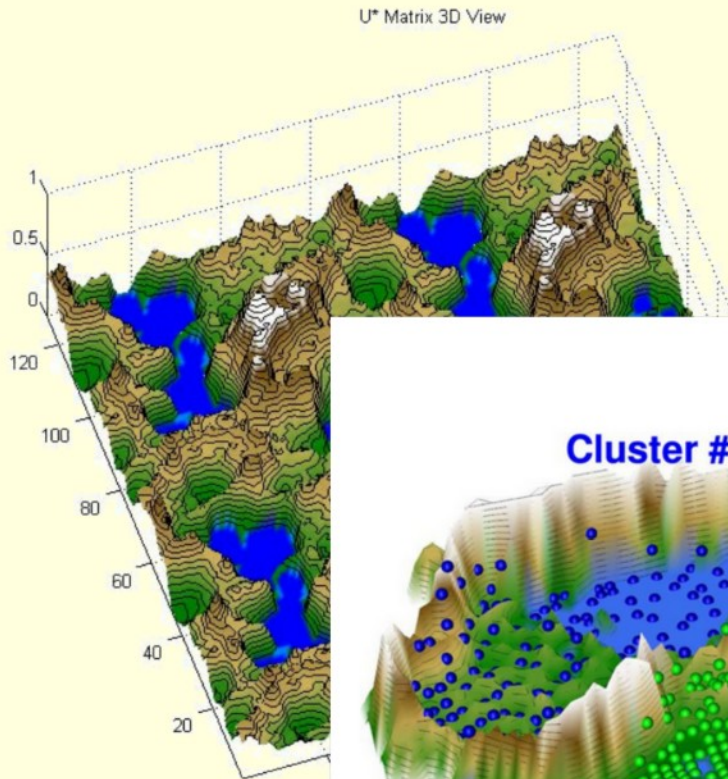


<https://demogng.de>

Bewertung der trainierten SOM-Karten

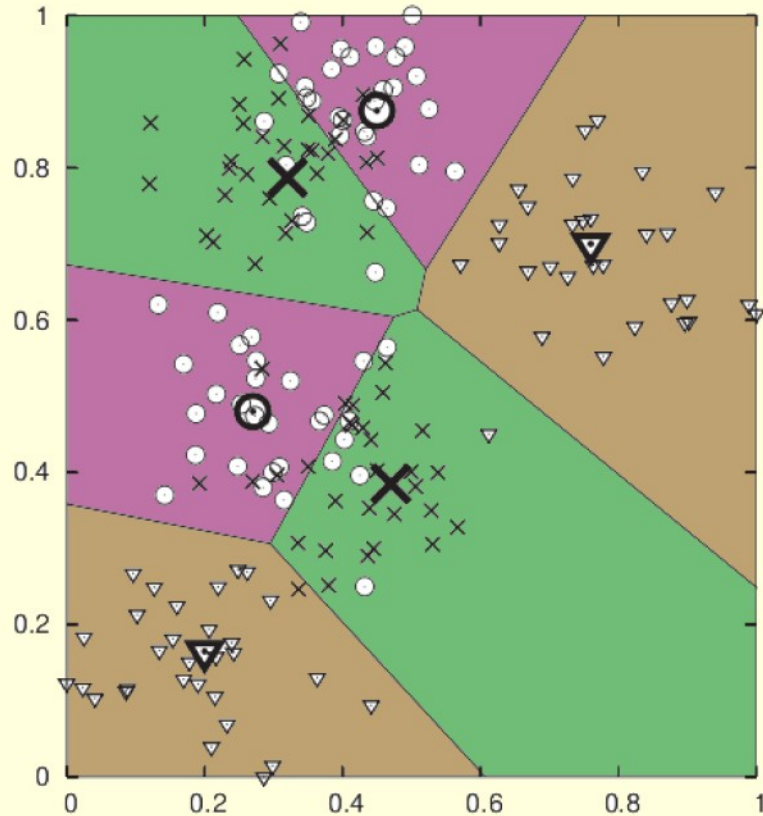
U-Matrix

[Alfred Ultsch, Databionics Marburg]

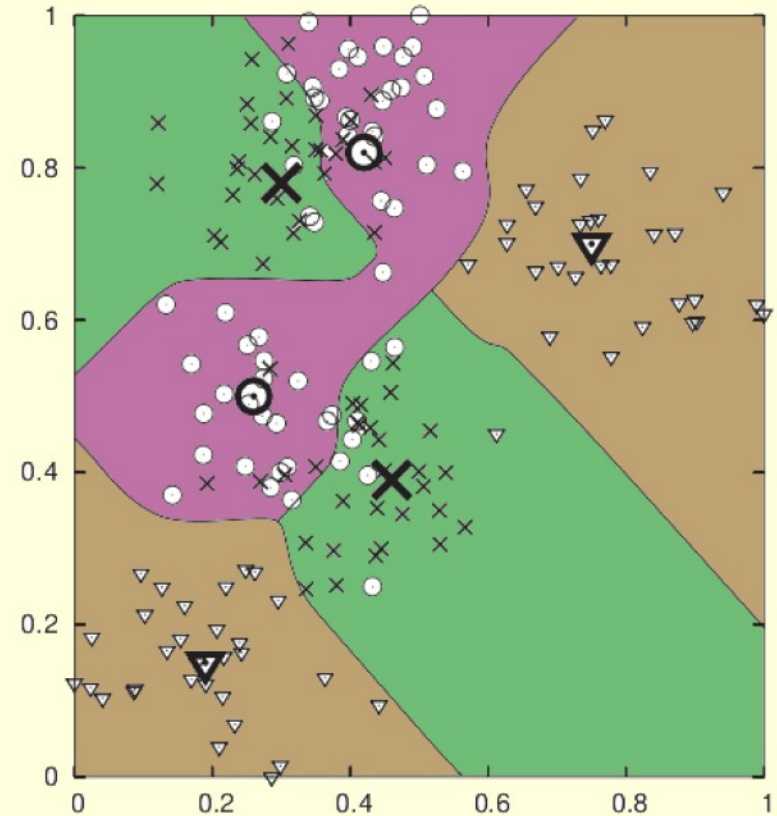


Vektorquantisierung mit nicht-standard Metriken

Rezeptive Felder für bimodale Daten mit drei Klassen



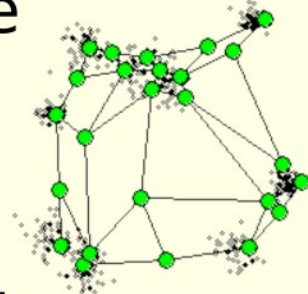
$$d(\mathbf{x}, \mathbf{y}) = \sum_i w_i \cdot (x_i - y_i)^2 \quad (\text{euklidisch})$$



$$d(\mathbf{x}, \mathbf{y}) = \sum_i w_i \cdot (x_i - y_i)^{10}$$

SOM: Was haben wir erreicht?

1. Komplexe hoch-dimensionale Eingabe-Reize können auf flächigen Strukturen nach Ähnlichkeiten **gruppiert** werden.



→ Vielschichtige Assoziationen und Bindungen.

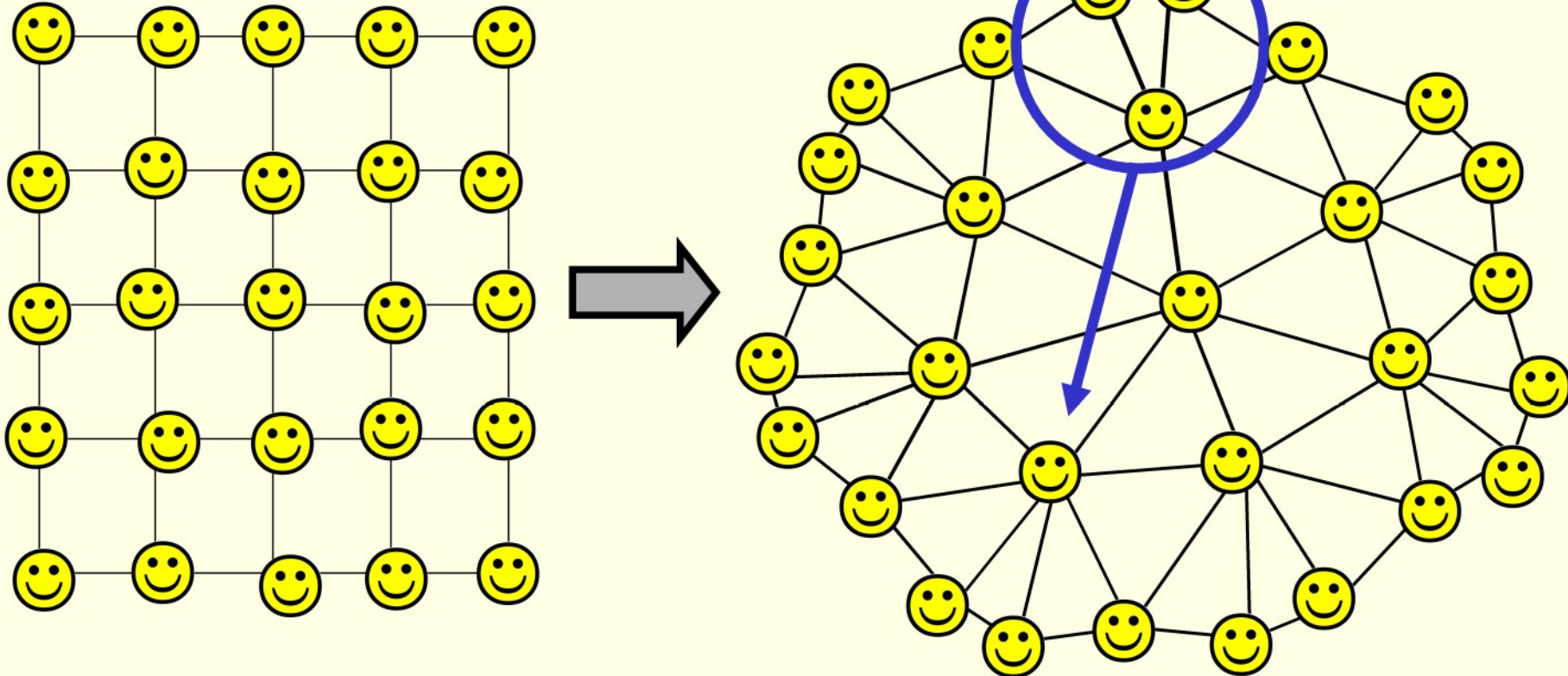
2. Prototypen-Modell erlaubt Daten-Abstraktion:

→ Viele einzelne Daten werden durch wenige Repräsentanten dargestellt.

3. Biologisch **plausible** Form der Organisation.

SOM: Erweiterungen

1. Höherer Vernetzungs-Grad („mehr Synapsen“)

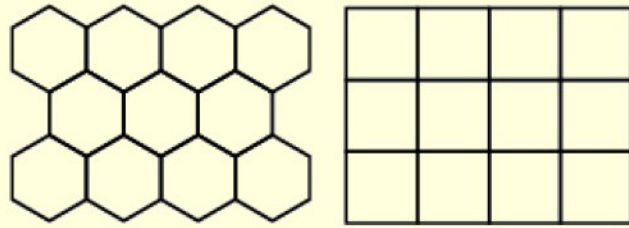


2. Neuronen, die Gewinner d. vorherigen Zeitschritts berücksichtigen.

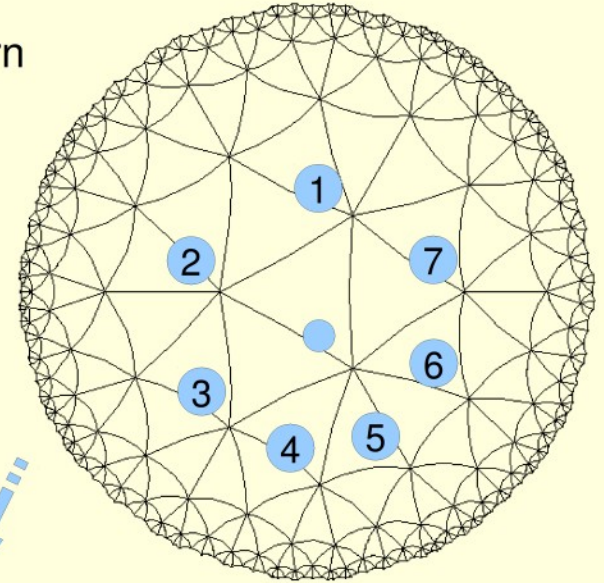
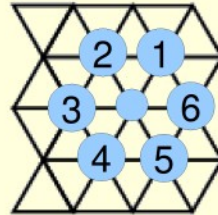
3. Neuronen, die in der Lage sind, sich selbst zu verdrahten

→ Neuronen-Gas [Martinetz, Schulden, Ritter, 1993].

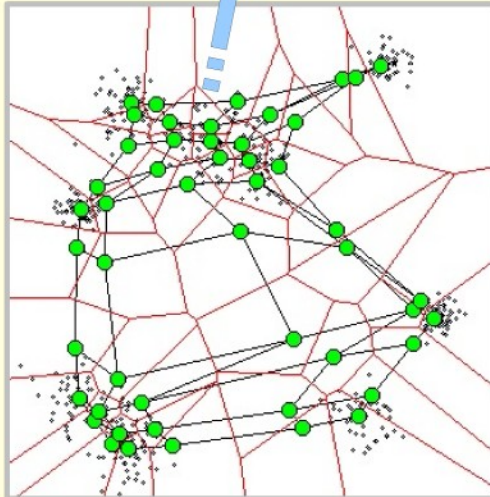
Hyperbolische Strukturen für exponentielle Kontexte



Zahl der Nachbarn



Entfalte Dich mit 2D-
Struktur im hoch-
dim. Raum!
„Filtere“ Daten

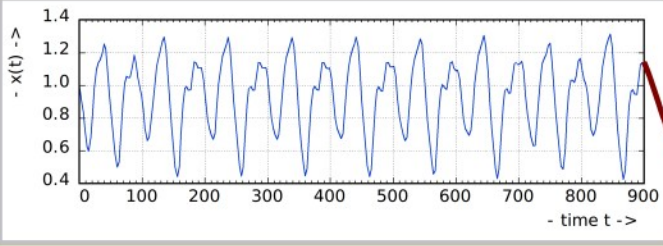


Dr. Diana Taimina
crochetcoralreef.org

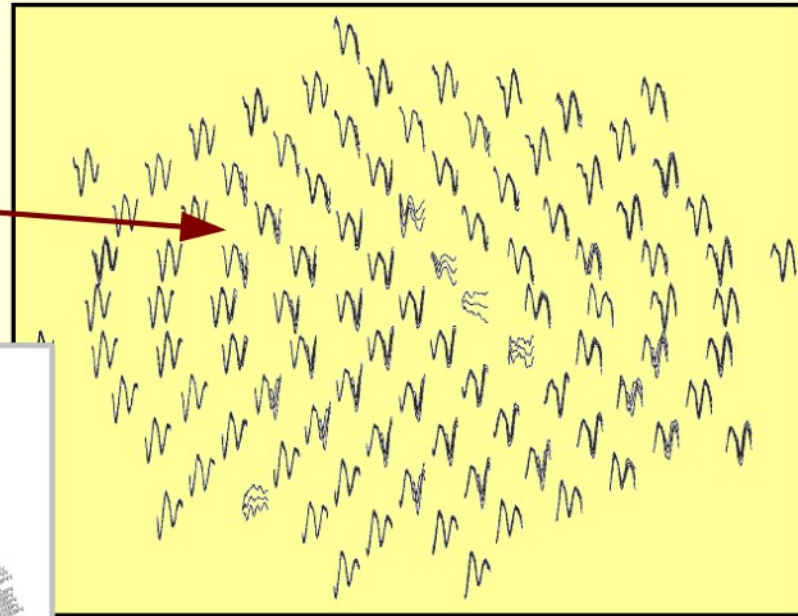
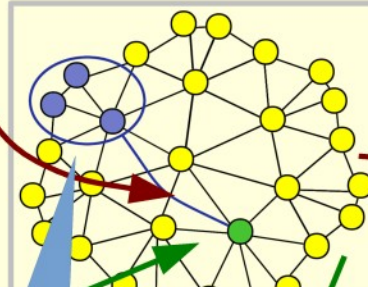


„Open Brain“ Koralle

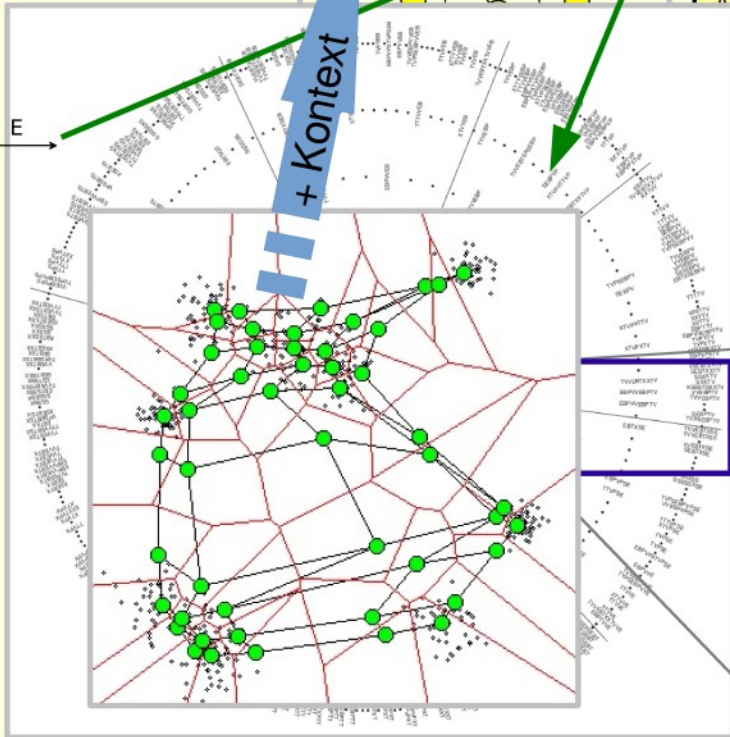
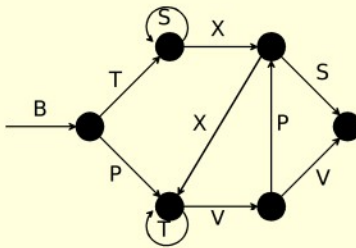
Hyperbolische SOM für Sequenzen (H-SOM-S)



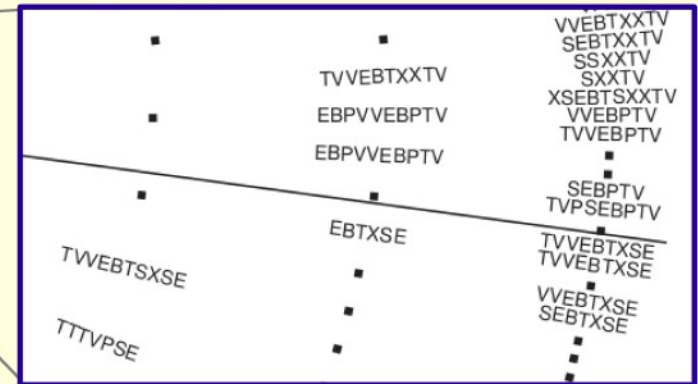
(H-)SOM-S Modell



Sequentielle Prozessdaten

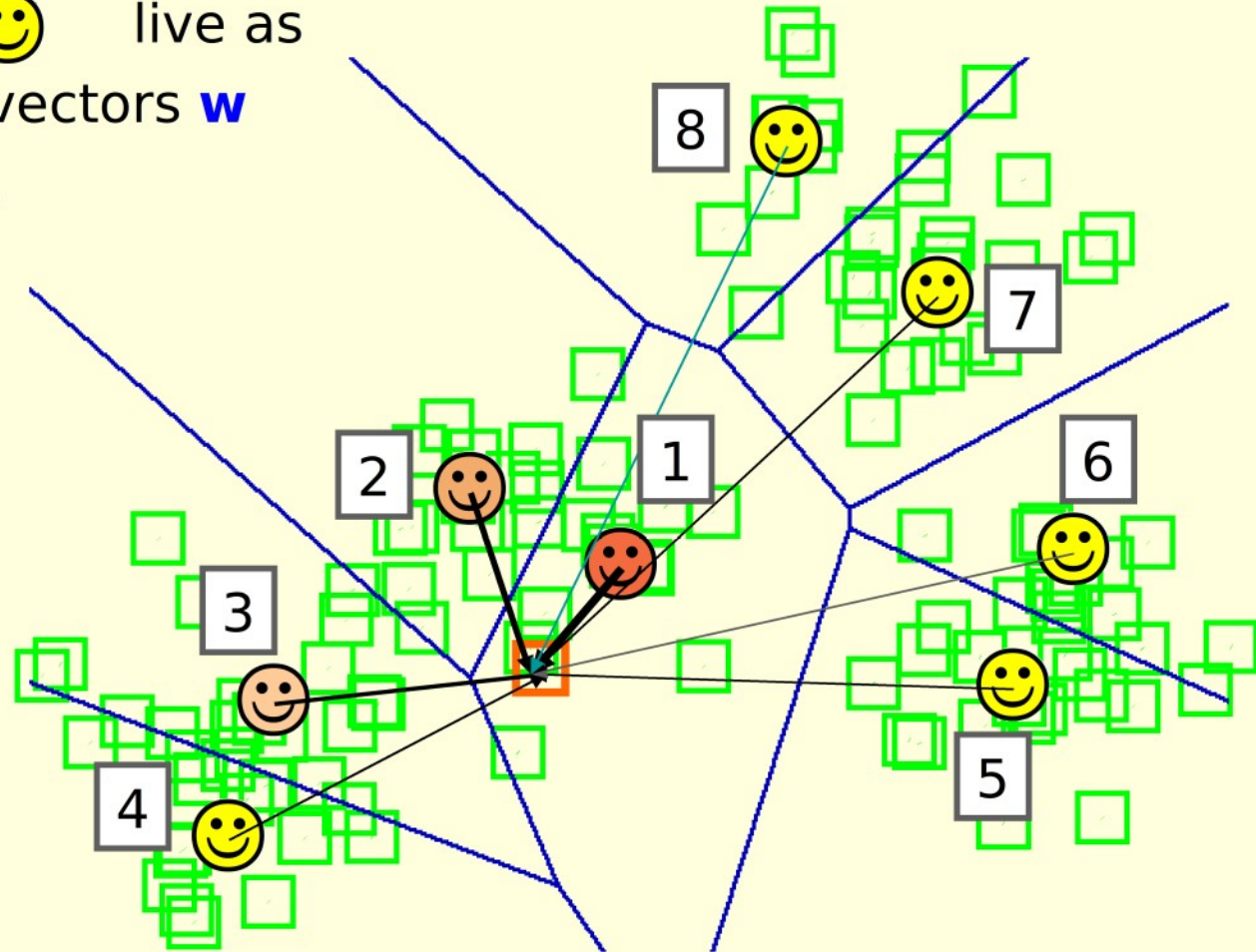


Topologie-erhaltende Abbildung



Dynamische Gittertopologie: Neural-Gas (NG)

Prototypes (😊) live as representing vectors w in data space.



[Martinetz and Schulten, 1991]

NG: Gradientenabstieg auf Kostenfunktion

$$E(\mathbf{W}, \sigma) = \frac{1}{C(\sigma)} \cdot \sum_{j=1}^m \sum_{i=1}^n h_{\sigma}(\text{rnk}(\mathbf{x}^i, \mathbf{w}^j)) \cdot d(\mathbf{x}^i, \mathbf{w}^j) \rightarrow \min!$$

Algorithm SOM / NG centroid update

repeat

 chose randomly a data vector \mathbf{x}

$k \leftarrow \arg \min_i \{ d(\mathbf{w}^i, \mathbf{x}) \}$

$\{ \mathbf{w}^k \text{ is closest centroid to data vector } \mathbf{x} \}$

for all m centroids j **do**

$\mathbf{w}^j \leftarrow \mathbf{w}^j + \gamma \cdot h_{\sigma}(D(\mathbf{w}^k, \mathbf{w}^j)) \cdot \frac{\partial d(\mathbf{x}, \mathbf{w}^j)}{\partial \mathbf{w}^j}$

end for

until no more major changes

2. Supervised LVQ

(points with class label)

Learning vector quantization (**LVQ**)

[Kohonen, 1986]

Neurons (☺) live as
prototypes w_j
in data space.

Labels for active class separation

Adaptation of “weights” w_j
as **prototypes** of
classes {red, green, blue}.

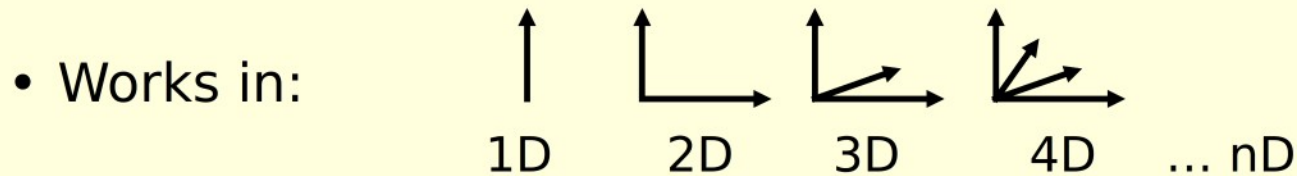
LVQ - Algorithm

repeat: *choose* $(x^i, y^i) \in X$ class label
 compute w^j *such that* $x^i \in R^j$
 $w^j := \begin{cases} w^j + \epsilon(x^i - w^j) & \text{if } y^i = c^j \\ w^j - \epsilon(x^i - w^j) & \text{otherwise} \end{cases}$

Hebbian learning term

LVQ Features (and a Desire)

- Reduction: many data items are represented by few prototypes.



- Crucial: notion of 'similarity':

→ Data metric important!

Extension

Automatic attribute weighting.



Adaptive Metric in Relevance LVQ (**RLVQ**)

[Bojer et al. 2001]

Euclidean Metric

$$|x - w|^2 = \sum_{i=1}^n (x_i - w_i)^2$$

attribute weight = relevance factor

λ -Adaptation

repeat: choose $(x^i, y^i) \in X$

compute w^j such that $x^i \in R_\lambda^j$

for all l :

$$\lambda_l := \begin{cases} \max\{\lambda_l - \alpha|x_l^i - w_l^j|, 0\} & \text{if } y^i = c^j \\ \lambda_l + \alpha|x_l^i - w_l^j| & \text{otherwise} \end{cases}$$

for all l :

$$\lambda_l := \lambda_l / |\lambda|$$

Cost Function Approach to **LVQ**

1. Formulate misclassification as differentiable function.
2. Do function optimization.

LVQ with cost function → **GLVQ**

Generalized LVQ (**GLVQ**) [Sato & Yamada, 1995]:

Minimize cost function

$$K = \sum_{\text{all inputs } \mathbf{x}} \text{error}(\mathbf{x}, \mathbf{w}_{\text{correct}}, \mathbf{w}_{\text{wrong}})$$

...using gradient descent $\partial K / \partial \mathbf{w} \rightarrow 0$!

Iteratively move prototype weights \mathbf{w}_s in small steps into the direction of error minimization!

GLVQ with relevance terms \rightarrow GRLVQ

Quantization $error_{\lambda}(\mathbf{x}, \mathbf{w}^{\text{current}}) \approx \lambda_j \cdot (x_j - w_j^{\text{current}})^2$

global relevance of data column j 

$$\partial error_{\lambda} / \partial \lambda \rightarrow 0$$

Generalized relevance LVQ (**GRLVQ**)

[Hammer & Villmann 2002]

Is large margin optimizer * (like SVM)

* for diagonal metrics

[Hammer, Strickert, Villmann 2005]

GRLVQ: cost function minimization framework

Misclassification costs:

$$E_{\text{GRLVQ}} := \sum_{i=1}^n g(q_{\lambda}(\mathbf{x}^i)) \quad \text{where}$$

data distance / similarity measure

$$q_{\lambda}(\mathbf{x}^i) = \frac{d_{\lambda}^{+}(\mathbf{x}^i) - d_{\lambda}^{-}(\mathbf{x}^i)}{d_{\lambda}^{+}(\mathbf{x}^i) + d_{\lambda}^{-}(\mathbf{x}^i)}$$

↑ ↑

closest correct wrong

Generic update formulas:

$$\Delta \mathbf{w}^{i+} = -\gamma^{+} \cdot \frac{\partial E_{\text{GRLVQ}}}{\partial \mathbf{w}^{i+}}$$

$$\Delta \mathbf{w}^{i-} = \gamma^{-} \cdot \frac{\partial E_{\text{GRLVQ}}}{\partial \mathbf{w}^{i-}}$$

$$\Delta \lambda = -\gamma^{\lambda} \cdot \frac{\partial E_{\text{GRLVQ}}}{\partial \lambda}$$

GRLVQ: Derivatives for two measures

Weighted Euclidean:

Adaptive metric parameters

$$d_{\lambda}^{\text{EUC}}(\mathbf{x}, \mathbf{w}^i) = \sum_{j=1}^d \lambda_j^{b_{\lambda}} \cdot (x_j - w_j^i)^{b_{\mathbf{w}}},$$

$$\Rightarrow \frac{\partial d_{\lambda}^{\text{EUC}}(\mathbf{x}, \mathbf{w}^i)}{\partial w_j^i} = -b_{\mathbf{w}} \cdot \lambda_j^{b_{\lambda}} \cdot (x_j - w_j^i)^{b_{\mathbf{w}}-1},$$

$$\frac{\partial d_{\lambda}^{\text{EUC}}(\mathbf{x}, \mathbf{w}^i)}{\partial \lambda_j} = b_{\lambda} \cdot \lambda_j^{b_{\lambda}-1} \cdot (x_j - w_j^i)^{b_{\mathbf{w}}}.$$

Pearson correlation:

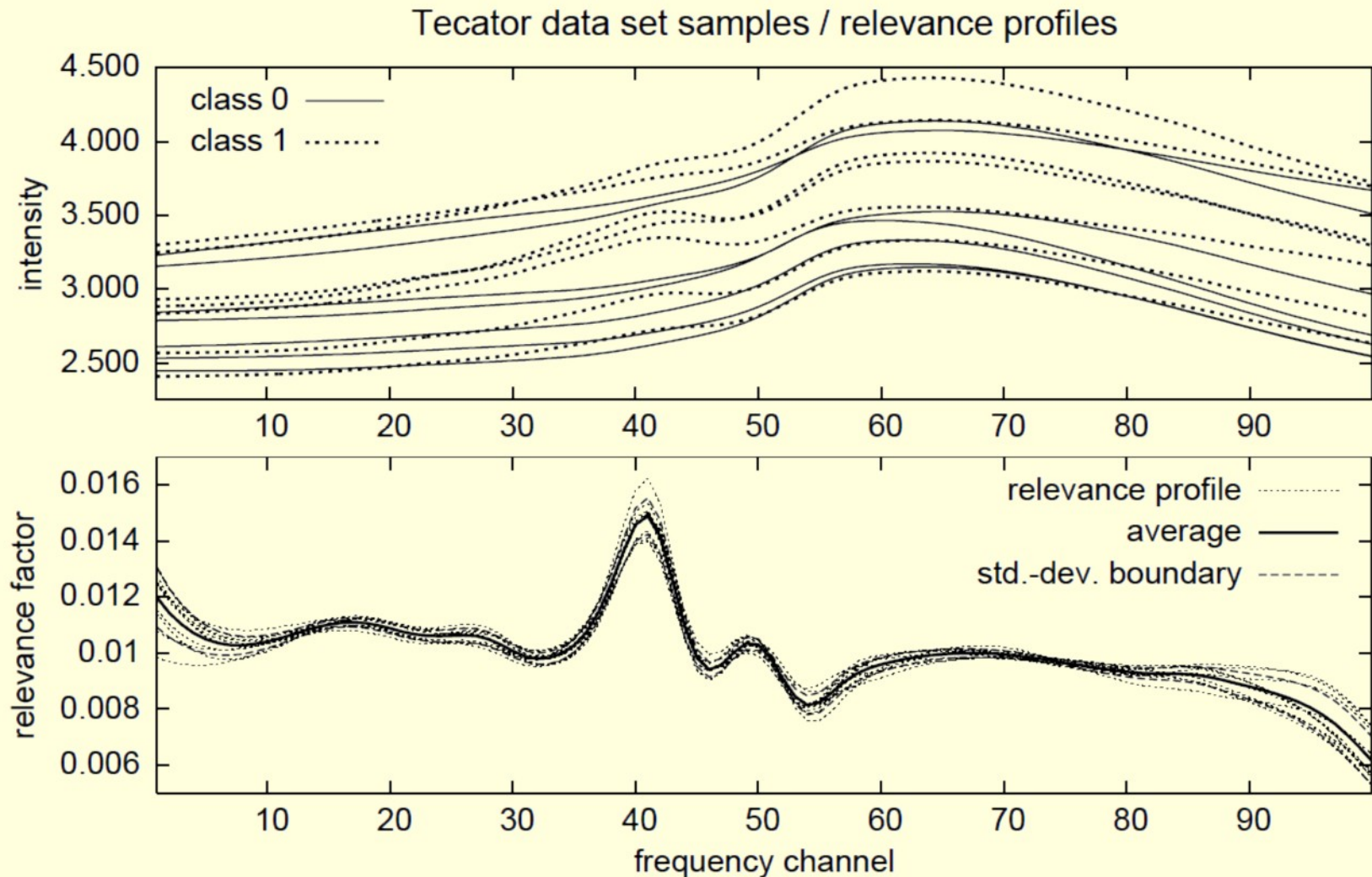
$$r = d^r(\mathbf{x}, \mathbf{w}^i) = \frac{\sum_{l=1}^d (w_l^i - \mu_{\mathbf{w}^i}) \cdot (x_l - \mu_{\mathbf{x}})}{\sqrt{\sum_{l=1}^d (w_l^i - \mu_{\mathbf{w}^i})^2} \cdot \sqrt{\sum_{l=1}^d (x_l - \mu_{\mathbf{x}})^2}}$$

$$R = \left(\frac{1}{C + r_{\lambda}(\mathbf{x}, \mathbf{w})} \right)^k - R_{\min}$$

$$\frac{\partial R}{\partial w_j} , \frac{\partial R}{\partial \lambda_j}$$

[Strickert et al., 2006]

App: **GRLVQ** for Tecator spectral data set

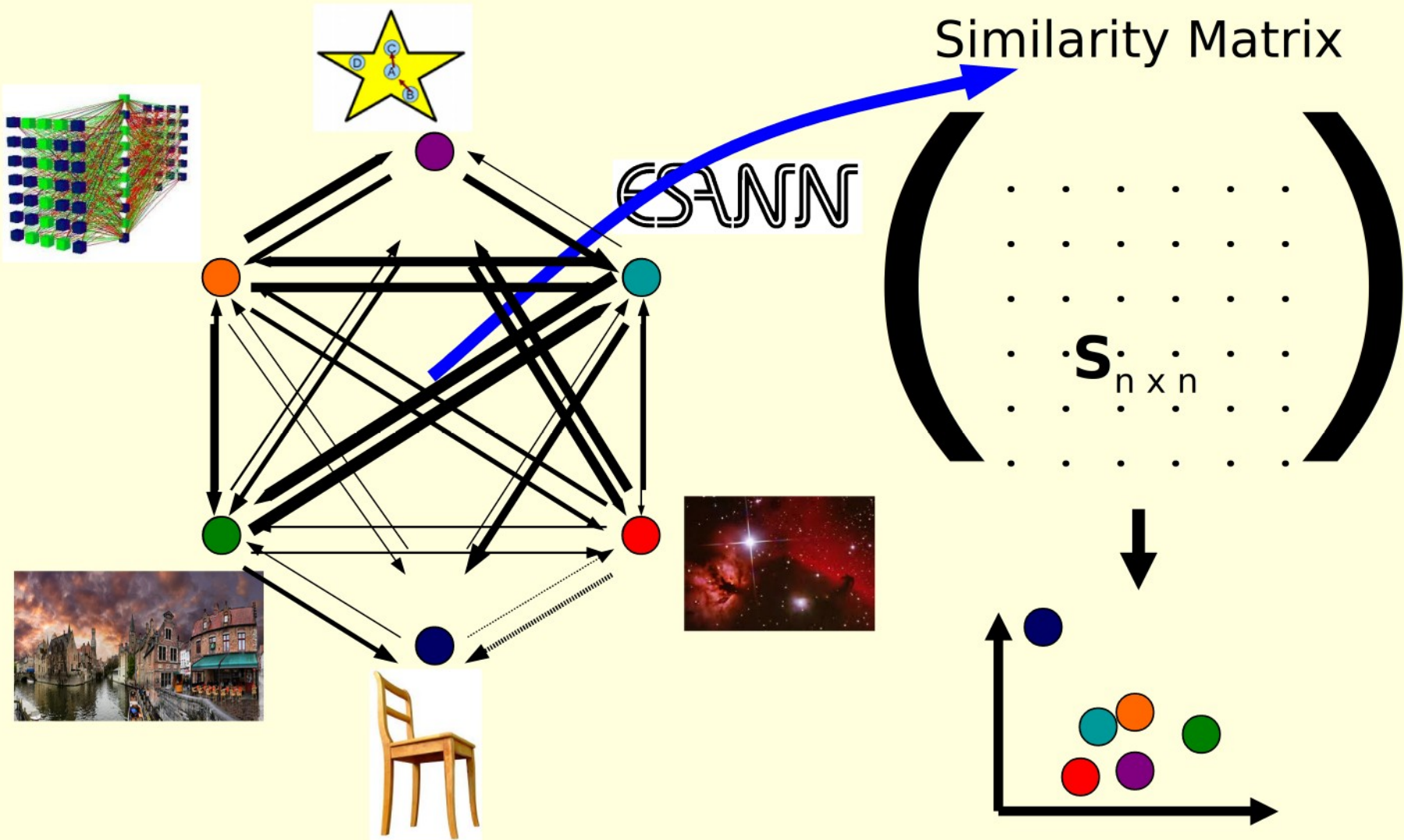


GRLVQ Performance for Tecator Data

Classification accuracies (test set):

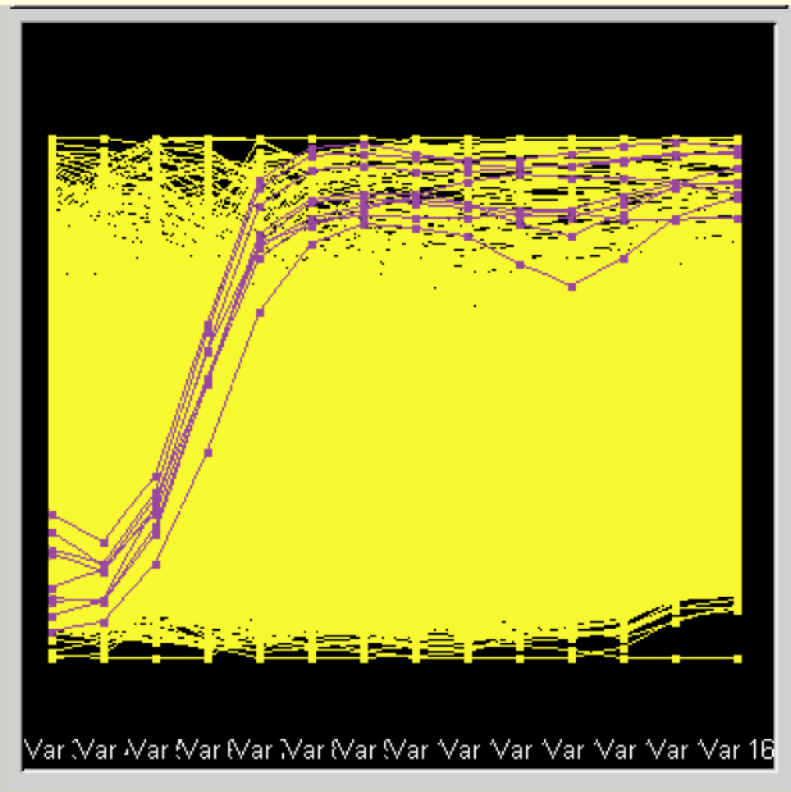
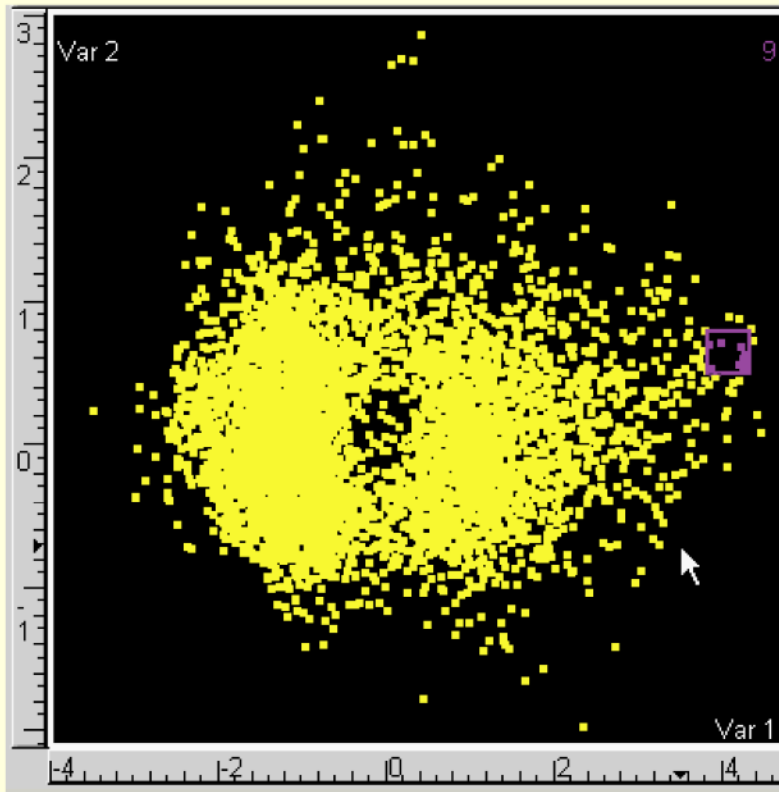
Without relevance terms:	96.3% (Corr)	84.2% (Euc)
With relevance terms:	97.7% (Corr)	88.4% (Euc)

Embedding Relational Data*



* going beyond dimension reduction: multidimensional scaling MDS

What's the use?

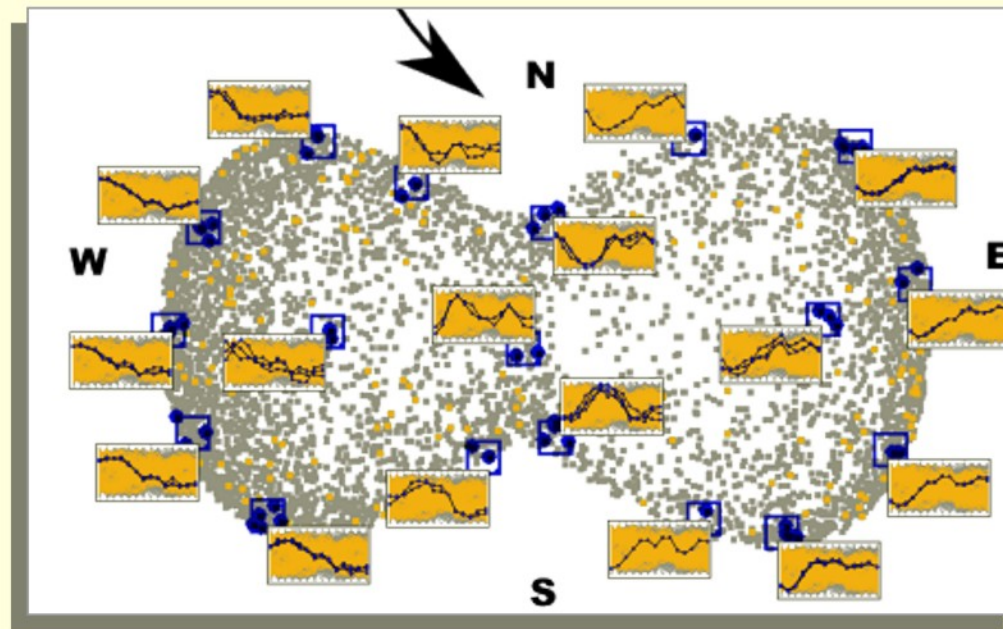


Left cluster: down-regulation,
Right cluster: up-regulation.

GGobi data visualization software

Fin

Vielen Dank für die Aufmerksamkeit!



Metrics vs. similarity measures

1. **Metric:** strict pattern matching

$$d(x,y) = 0 \text{ only if } x = y.$$

Euclidean distance, Minkowski distances.

2. **Measure:** relaxed comparison

$$d(x,y) = \text{max. for maximum similarity.}$$

- Pearson correlation (real-valued),
 - Spearman correlation (rank-based).
- For intensity fields.
-

3. **Adaptivity:** Data-driven, adaptive parameters.

(Let the measure evolve according to requirements.)