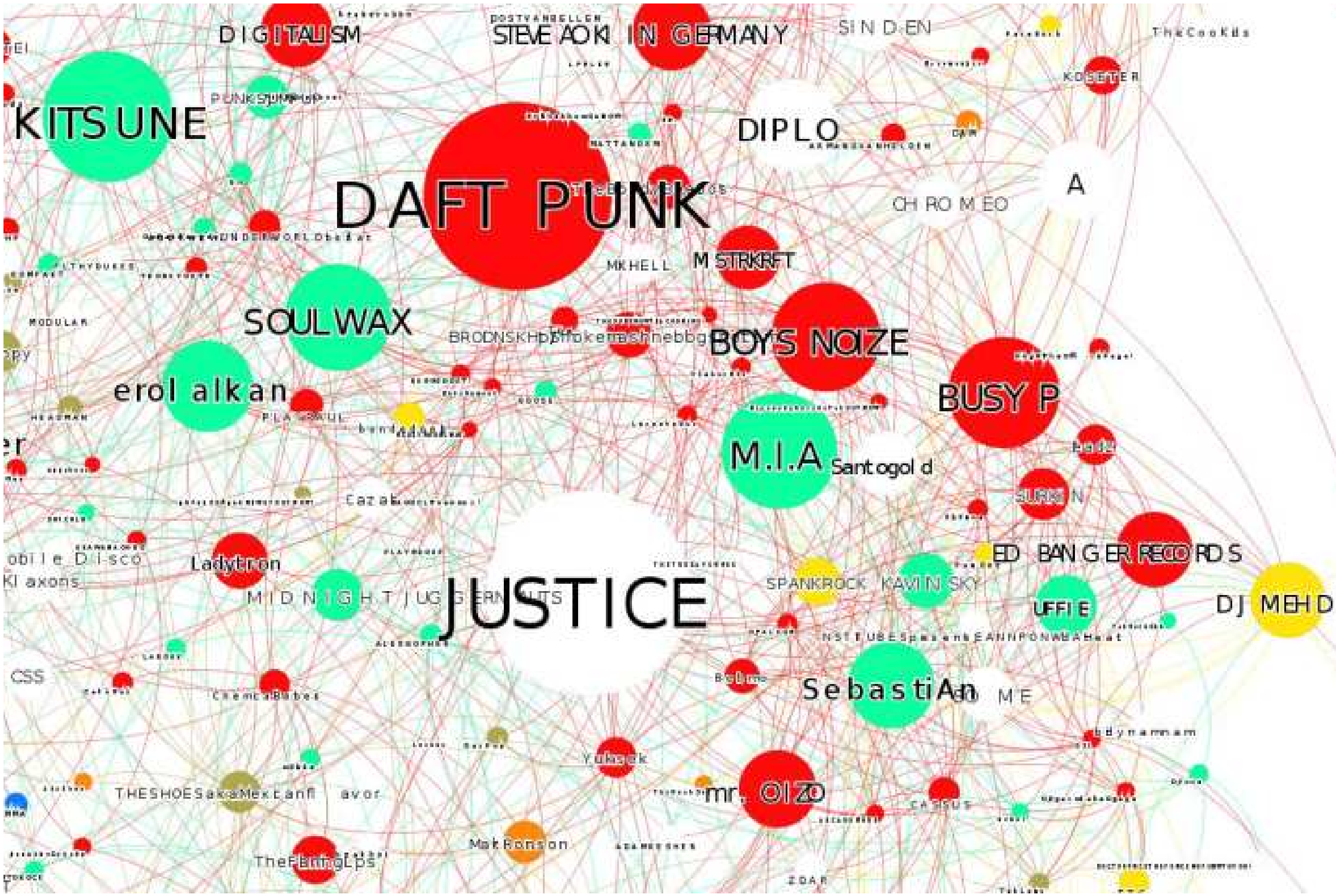
Clustering of the Self Organizing Map

Roland Halbig
14. Januar 2014
Pattern Recognition Lab (CS 5)

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT


DIGITALISM
STEVE AOKI
IN GERMANY
SINDEN
KITSUNE
DAFT PUNK
DIPLO
A
CHROMEO
MSTRKRFT
SOULWAX
BOYS NOIZE
BUSY P
erol alkan
M.I.A
Santogold
ED BANGER RECORDS
DJ MEHDI
JUSTICE
Ladytron
SPANKROCK
KAVINSKY
UFFIE
SebastiAn
mr. OIZO
Mark Ronson
CSS

Self Organizing Maps

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

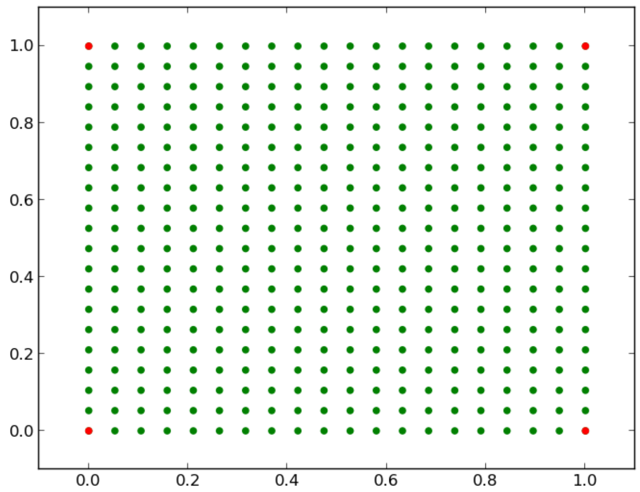
Definition: SOM

Let $\mathcal{I} \subset \mathbb{N}$ be a finite index set. A **Self Organizing Map** (SOM) is defined as a set

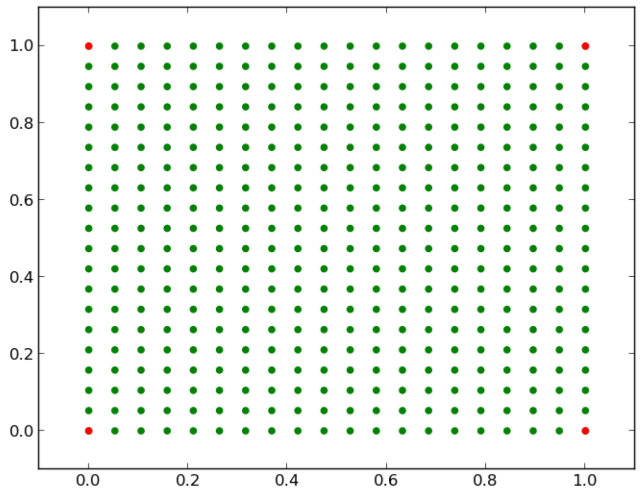
$$SOM_{\mathcal{I}} = \{(N_\lambda, r_\lambda) \in R^m \times \mathbb{R}^d \,|\, \lambda \in \mathcal{I}\}$$

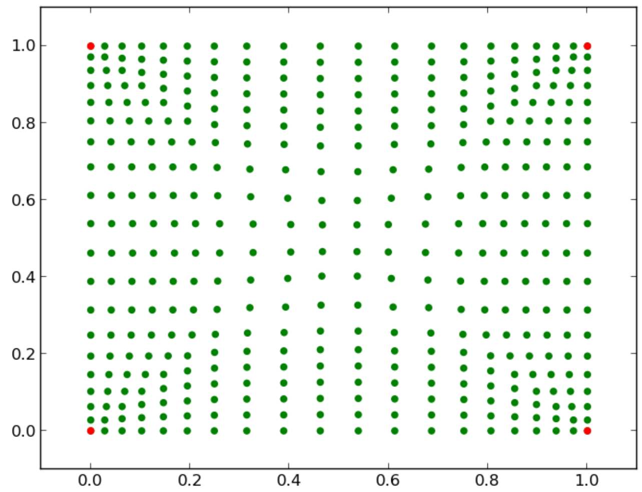
of prototype vectors $N_\lambda \in \mathbb{R}^m$ and their position $r_\lambda \in \mathbb{R}^d$ in a grid.

Example: XOR



Example: XOR





The SOM-Algorithm

1. Initialize the prototype vectors N_λ of the SOM, set the counter $t = 0$

The SOM-Algorithm

1. Initialize the prototype vectors N_λ of the SOM, set the counter $t = 0$
2. Iterate steps 2-6 while $t < tmax$

The SOM-Algorithm

1. Initialize the prototype vectors N_λ of the SOM, set the counter $t = 0$
2. Iterate steps 2-6 while $t < tmax$
3. Choose a sample $X_s \in \mathbb{R}^m$

The SOM-Algorithm

1. Initialize the prototype vectors N_λ of the SOM, set the counter $t = 0$
2. Iterate steps 2-6 while $t < tmax$
3. Choose a sample $X_s \in \mathbb{R}^m$
4. Calculate the **Best Matching Unit** N_b with

$$b = \arg\min_{\lambda \in \mathcal{I}} \left\{ \|X_s - N_\lambda\|_{L^2(\mathbb{R}^m)} \right\}$$

The SOM-Algorithm

1. Initialize the prototype vectors N_λ of the SOM, set the counter $t = 0$
2. Iterate steps 2-6 while $t < tmax$
3. Choose a sample $X_s \in \mathbb{R}^m$
4. Calculate the **Best Matching Unit** N_b with

$$b = \arg\min_{\lambda \in \mathcal{I}} \left\{ \|X_s - N_\lambda\|_{L^2(\mathbb{R}^m)} \right\}$$

5. Adjust all nodes N_λ in the proximity of N_b:

$$N_\lambda^{t+1} = N_\lambda^t + h_{b,\lambda}(t) \left(X_s - N_\lambda^t \right)$$

The SOM-Algorithm

1. Initialize the prototype vectors N_λ of the SOM, set the counter $t = 0$
2. Iterate steps 2-6 while $t < tmax$
3. Choose a sample $X_s \in \mathbb{R}^m$
4. Calculate the **Best Matching Unit** N_b with

$$b = \arg\min_{\lambda \in \mathcal{I}} \left\{ \|X_s - N_\lambda\|_{L^2(\mathbb{R}^m)} \right\}$$

5. Adjust all nodes N_λ in the proximity of N_b:

$$N_\lambda^{t+1} = N_\lambda^t + h_{b,\lambda}(t) \left(X_s - N_\lambda^t \right)$$

6. Set $t = t + 1$ and adjust the **kernel radius** σ

The Neighbourhood Kernel Function

Kernel Function:

$$h_{b,\lambda}(t) = \exp\left[-\frac{\|r_b - r_\lambda\|_{L^2(\mathbb{R}^m)}}{2\sigma^2(t)}\right]$$

The Neighbourhood Kernel Function

Kernel Function:

$$h_{b,\lambda}(t) = \exp\left[-\frac{\|r_b - r_\lambda\|_{L^2(\mathbb{R}^m)}}{2\sigma^2(t)}\right]$$

Adaption of kernel radius σ:

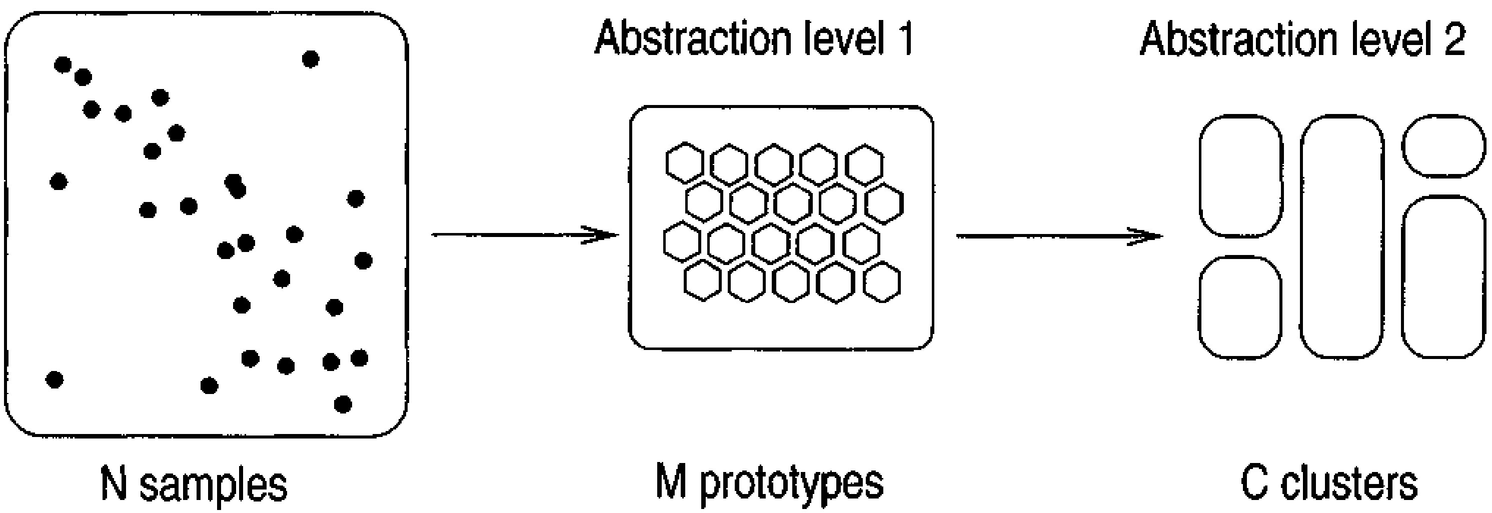
$$\sigma(t+1) = \sigma_0 \exp\left[-\frac{t}{t_{max}} \log\left(\sigma(t) + 1\right)\right]$$

Clustering Of The SOM

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

TECHNISCHE FAKULTÄT

Two-Level-Approach

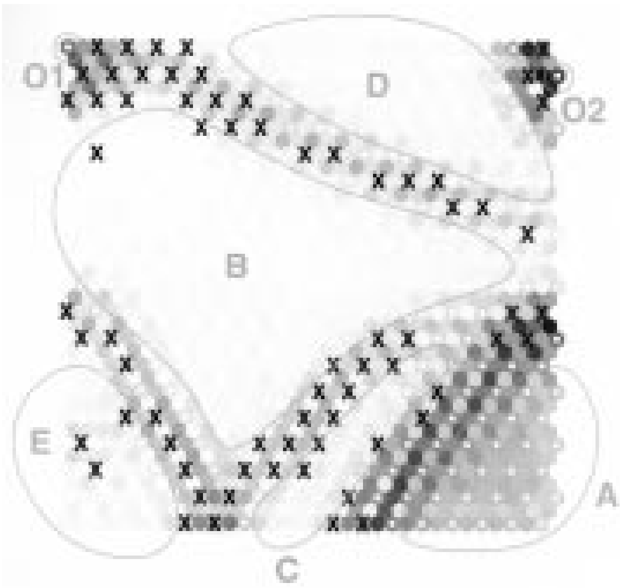


Visualization - Unified Similarity Matrix

$$U_{i,j} = d(N_{i,j}, N_{i+1,j}) + d(N_{i,j}, N_{i-1,j}) + d(N_{i,j}, N_{i,j+1}) + d(N_{i,j}, N_{i,j-1})$$

Visualization - Unified Similarity Matrix

$$U_{i,j} = d(N_{i,j}, N_{i+1,j}) + d(N_{i,j}, N_{i-1,j}) + d(N_{i,j}, N_{i,j+1}) + d(N_{i,j}, N_{i,j-1})$$



Agglomerative and Partitive Clustering

Agglomerative

1. Initialize
2. Compute inter-cluster distance
3. Merge two clusters that are closest
4. Return to step 2 until one cluster is left

Agglomerative and Partitive Clustering

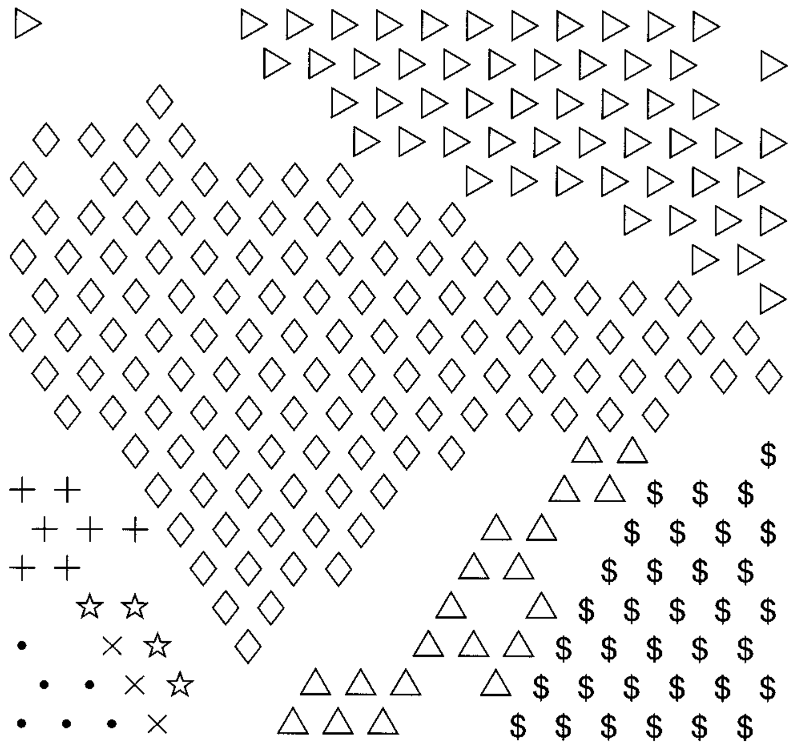
Agglomerative

1. Initialize
2. Compute inter-cluster distance
3. Merge two clusters that are closest
4. Return to step 2 until one cluster is left

Partitive

1. Determine number of clusters and initialize cluster centers
2. Compute partitioning of the data
3. Update the cluster centers
4. Return to step 2 until partitioning is unchanged

Visualizing the Clustering



Thank You For Your Attention!

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

Sources

Clustering of the Self Organizing Map by Juha Vesanto and Esa Alhoniemi
Self-Organizing Map and social networks: Unfolding online social popularity by Couronne Thomas, Beuscard Jean-Samuel and Chamayou Cedric

code: https://github.com/wedgeCountry