Adaptive batch SOM for multiple dissimilarity data tables

Anderson Dantas
absd@cin.ufpe.br

Batch Self-Organizing Map Algorithm for dissimilarity data

Iterative two step algorithm:
Affectation and representation steps

Cost function

Extension of the k-means cost function

$$J = \sum_{i=1}^{n} \sum_{r=1}^{m} K^T(\delta(f(e_i), r)) d(e_i, g_r)$$

$$\lim_{|x| \to \infty} K(x) = 0$$

Generalised distance

$$d^T(e_i, g_{f(e_i)}) = \sum_{r=1}^{m} K^T(\delta(f^T(e_i), r))d(e_i, g_r)$$

Affectation step

Function f associates e_i to the "closest"neuron

$$c = f^T(e_i) = arg \min_{1 \leq r \leq m} d^T(e_i, g_r)$$

Representation step

New prototypes are selected

$$g_r^* = arg \min_{e \in E} \sum_{i=1}^{n} K^T(\delta(f^T(e_i), r)) d^T(e_i, e_r)$$

Adaptive batch SOM for data based on multiple dissimilarity matrices

Also iterative, but three-step algorithm
Representation, weighting and affectation

The B-SOM Algorithm

1. Initialization
   - ✓ Fix: m, δ, K^T, N_{iter}, T_{min}, T_{max}, $T \leftarrow T_{max}$, $t \leftarrow 0$
   - ✓ Randomly select m distinct prototypes
   - ✓ Set the map $L(m, G^0)$
   - ✓ Assign each object e_i to the closest neuron (cluster)

The B-SOM Algorithm