Multiple dissimilarity SOM for clustering and visualizing graphs with node and edge attributes

Nathalie Villa-Vialaneix, INRA, UR 0875 MIAT & Madalina Olteanu, Université Paris 1, SAMM, FRANCE

nathalie.villa@toulouse.inra.fr - madalina.olteanu@univ-paris1.fr

References


Standard SOM for multidimensional data [4]

Cluster data \( (x_i)_{i=1,...,n} \in \mathbb{R}^d \) on a grid made of \( U \) units and equipped with a distance between units, \( d(u, u') \). Units have representers called prototypes \( (p_u)_{u \in U} \). Clustering \( f : \mathbb{R}^d \to \{1, \ldots, U\} \) and prototypes are updated iteratively in order to preserve the topology of the input space:

1. **affection step**: pick a data \( x_i \) at random and find the best matching unit: \( f(x_i) := \arg\min_{u=1,...,U} \|x_i - p_u\|^2 \)
2. **representation step**: update the BMU and its neighbors’ prototypes with a stochastic gradient descent like scheme: \( p_u \leftarrow p_u + \mu H(d(f(x_i), u))(x_i - p_u) \)

**Extension of SOM to data described by a kernel / a dissimilarity**

Data: \( (x_i)_{i=1,...,n} \in G \) described by pairwise relations with a kernel \( K \in M_{n \times n} \) or a dissimilarity \( \Delta \in M_{n \times n} \Rightarrow \) stochastic kernel SOM [2] and stochastic relational SOM [6] implemented in SOMbrero (R package)

Prototypes: linear convex combination of the data \( p_u = \sum_{i=1}^n \beta_{ui} \phi(x_i) \) (only \( \beta_{ui} \) are trained). \( \phi \) is implicitly defined by the kernel/dissimilarity

1. **affection step** writes \( f(x_i) = \arg\min_u \beta_{ui}^T K \beta_{ui} - 2 \beta_{ui}^T K_i \) (kernel SOM) or \( f(x_i) = \arg\min_u \Delta_i \beta_{ui} - \beta_{ui}^T \Delta_u \) (relational SOM)
2. **representation step** writes \( \beta_{ui} \leftarrow \beta_{ui} + \mu H(d(f(x_i), u))(1_i - \beta_{ui}) \)

**Applications to graphs**

Type of data that can be handled:

- graphs with node attributes (a kernel for the graph structure - e.g., Laplacian based kernels; kernels for each of the attributes)
- graphs with different types of edge (a kernel for each subgraph defined by an edge type)
- both... and can also be used to combine different kernels with different parameters

Useful for:

- uncover communities...
- ... and visualize the relations between communities
- as shown in [7], the result of the SOM can be combined with clustering of the prototypes to obtain a simplified representation of a graph

**Mixing multiple kernels**

Data are described by several pairwise relations (kernels/dissimilarities) \( K_1, \ldots, K_D \Rightarrow \) Multiple kernel: \( K = \sum_{k=1}^D \alpha_k K^k \) with \( \alpha_k \geq 0 \) and \( \sum_k \alpha_k = 1 \)

How to choose \( (\alpha_k)_k \)?

Similarly to [8], add a stochastic gradient descent step in SOM training:

3. **multiple kernel tuning step** \( \alpha_k \leftarrow \alpha_k + \nu D_{k,i} \) with \( D_{k,i} = \sum_u H(d(f(x_i), u)) (K^k(x_i, x_i) - 2 \beta_{ui}^T K_i^k) + \beta_{ui}^T K^k \beta_u \) (+ reduction & projection to ensure the \( \alpha_k \) remain positive and sum to 1)

see [5] (multiple kernels) or [6] (multiple dissimilarities)

An example (on simulated data)

Simulation of 8 groups of observations made from:

- unweighted graph (planted 3-partition graph; see [1]) with two dense groups of nodes: commute time kernel (\( L^+ \) with \( L \) the Laplacian; see [3])
- nodes are labelled with numeric data from a 2D Gaussian mixture: Gaussian kernel;
- ... and nodes are labelled with a factor (2-levels): Gaussian kernel on 0/1 recoding.

Resulting Map

Comparison

(100 datasets - NMI with true groups)

References