Graph-based normalization for non-linear data analysis (II) : application to SOM optimization

Catherine Aaron
SAMOS-MATISSE
Paris France
catherine_aaron@hotmail.com

Abstract - *We have seen in the "graph-based normalization" paper that the proposed method of normalization is efficient to compute geodesic distance and to "help" the Kohonen algorithm to find an adapted structure. We are now going to apply these properties to find an index of SOM performance that will lead us to find "good" input structure (i.e. an index that quantifies the topology preservation between data set and computed lattice)*

Key words - normalization, geodesic distance, graph, SOM

1 Introduction

In the whole paper we use the following notations : we have a data set X of N points in $\Re^D$ $X_i \in \Re^D$ is the i^{the} individual $X_i = (X_i^1, ..., X_i^D)$.
The main assumption is that the points are drawn from uniform distribution on a $d-$ dimensional variety (with potential noise) which is homeomorph to an hyper-rectangle $\prod_{i=1}^{d}[a_i, b_i]$.

By running the SOM algorithm on a $d-$dimensional map (with rectangular neighborhoods) we expect to observe the organization of points on the hyper-rectangle $\prod_{i=1}^{d}[a_i, b_i]$.
The main purpose of this paper is to find d and a "good" number of cells in each direction such that the Kohonen map topology is pertinent. In order to do that, we are first going to define a "topology preservation" measure.

2 Measure of topology preservation

2.1 Previous works

Some previous works give measure of the topology preservation in a Kohonen map. Goodhill and Sejnowski inventoried them in [10], they class topology quantification index into two groups : based on C measure indexes and the others ones. Let's detail what they call C measure :

Let F be a similarity (or dissimilarity) matrix on the data set.

Let $cl(i)$ represent the cell in which the point X_i is projected by the SOM algorithm
Let G be a similarity (if F is a similarity) or a dissimilarity (if F is a dissimilarity) matrix on the map structure (depends on the topology asked as input).

$$C = \sum_{i=1}^{N} \sum_{j<i} F(i,j)G(cl(i),cl(j))$$

Maximization of C approximates the maximization of the correlation coefficient between similarity (resp dissimilarity) of the data set and their projection after running SOM.
As usual, to see if the correlation coefficient is a pertinent index observation of scatter plots of similarities can be computed.

finally authors listed some couple of F and G they seen in papers :

- A- minimal writing (see [10] for exact references) :
  - $F(i,j) = 1$ if X_i and X_j neighbors, 0 otherwise
  - $G(cl(i),cl(j)) = ||cl(i),cl(j)||$
- B- minimal path length (see [10] for exact references):
  - $F(i,j) = ||X_i - X_j||$
  - $G(cl(i),cl(j)) = 1$ if X_i and X_j neighbors, 0 otherwise
- C- Jones et al (see [10] for exact references):
  - $F(i,j) = 1$ if X_i and X_j neighbors, 0 otherwise
  - $G(cl(i),cl(j)) = 1$ if X_i and X_j neighbors, 0 otherwise

Other approaches :
In [3] (D) Demartines and Heraut unfold data space according to the SOM results and compare euclidian distance between data points after unfolding operation and euclidian distance between cells in the map. contrary of Goodhill, we think it can be considered as a C measure with :

- $F(i,j) = ||unfold(X_i), unfold(X_j)||$
- $G(cl(i),cl(j)) = ||cl(i),cl(j)||$

Another approach, radically different, is the topographic function (E), exposed by Vilman et al in [8]. They observe neighborhood conservation by comparing neighborhood in SOM structure and Voronoi cells of weight vectors.
Let notice that the list we made here is not an exhaustive one, other indicators exist such as the minimal distortion, the STRESS measure...

2.2 Another approach

2.2.1 Why another approach ?

If methods A,B,C, seem quite intuitive they are note based on the true topology preservation as defined in mathematics. Even if they look interesting, similarity matrices based only on the neighborhood function coded by 0 or 1 will only give indication on "local" topology preservation and may not be helpful to determine the total topology preservation.
Method E is based on mathematics considerations but its biggest default is the computation of Voronoi's cells which will make the running time too long since the dimension of the data set d is such as $d \geq 3$.
If it is not clearly explained for the method D, the idea is really similar to ours : unfolding the data space and considering the euclidian distance on the unfolded space can be seen as computing the geodesic distance. The main critic that we can make is that the unfolding operation is done on the results of SOM algorithm, and it is supposed that result is "good" to check if it is or not.
Our approach tries to be based on mathematical considerations (as E) and it is very similar to D's results because of the underlying geodesic distance. As A,B and C, we will observe correlation between dissimilarity matrices.

2.2.2 Characterization of topology preservation

Let (E_1, d_1) and (E_2, d_2) be two metric spaces, they are said to be C_0 homeomorph if it exists $g : E_1 \to E_2$ a C_0 function such that $\forall (x, y) \in E_1^2, d_1(x, y) = d_2(g(x), g(y))$
If (E_2) is a $d-$dimensional variety homeomorph with an euclidian set the natural distance on E_2 is the geodesic one.

2.2.3 Application to Kohonen maps

In our case the first metric set E_1 will be the lattice and the second one the data-set space in which we will particularly focus on the weight vectors.
To a cell $(i_1, ..., i_d)$ in the $d-$dimensional lattice computed by the Kohonen algorithm corresponds the weight vector $w_{i_1,...,i_d} \in \Re^D$. This gives us the easiest definition of g : $g((i_1, ..., i_d)) = w_{i_1,...,i_d}$.
The main problem is to find "good" distances that correspond to the topology preservation. As noticed previously, a natural way to choose d_1 and d_2 is to consider:

$$d_1((i_1, ..., i_d), (j_1, ..., j_d)) = \sqrt{\sum (i_k - j_k)^2}$$

and $d_2(w_{i_1,...,i_d}, w_{j_1,...,j_d})$ the geodesic distance between the two code vectors which correspond to the euclidian distance on the unfolded space.
We can consider three way to compute d_2 :

- Consider $k-$nearest neighbors graph on the data set + weight vectors set and run Dijkstra's algorithm. As the computation of the geodesic distance is long we have to explore other computation ways

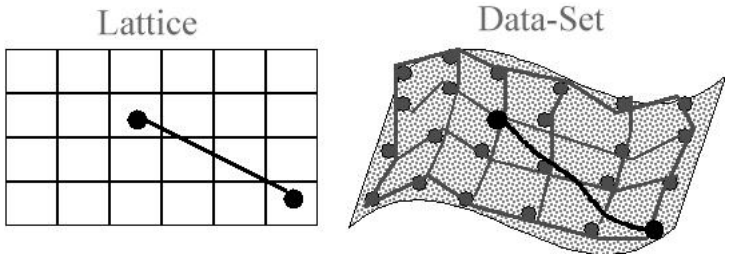

Figure 1: Distance on the lattice and corresponding distance on the data-set

- Another method could be to allow paths to go through results of a vector quantization of data set
- Compute the geodesic distance only on weight vectors set
- The last idea, which we unfortunately not computed yet, should be to run growing neural gas on the "total set" i.e. data-set and weight vectors results lead us to a graph structure (no need to give the number of neighbors as input) on which we can compute geodesic distance

Just notice that, the computation of geodesic distance for more than 500 points is long.

3 Results For Structure Recognition

We tested the proposed method, i.e. examination of link between geodesic distance in the data-space and the euclidian distance in the map-structure set for 4 different set :

- A : 500 points drawn on $[0,1]^2$ (variety dimension : 2 ; space dimension : 3)
- B : 500 points on horseshoe surface (variety dimension : 2 ; space dimension : 3)
- C : 500 points on a sinusoïde drawn (variety dimension : 1 ; space dimension :3) $X \rightarrow \mathcal{U}[0,1], Y = sin(40X)$ without normalization
- C' : C example graph-based normalized
- D : 500 points on a circle (doesn't not match the hypothesis)

Results are presented as follows :

- First scatter plots between the two distances (horizontal : map distance, vertical geodesic distance) for 10 Kohonen maps from "string" $(1;100)$ to "square" one $(10;10)$
- Correlation coefficient between the two distances for all the Kohonen maps
- points scatter-plot and Kohonen map subplot for the maximum of correlation

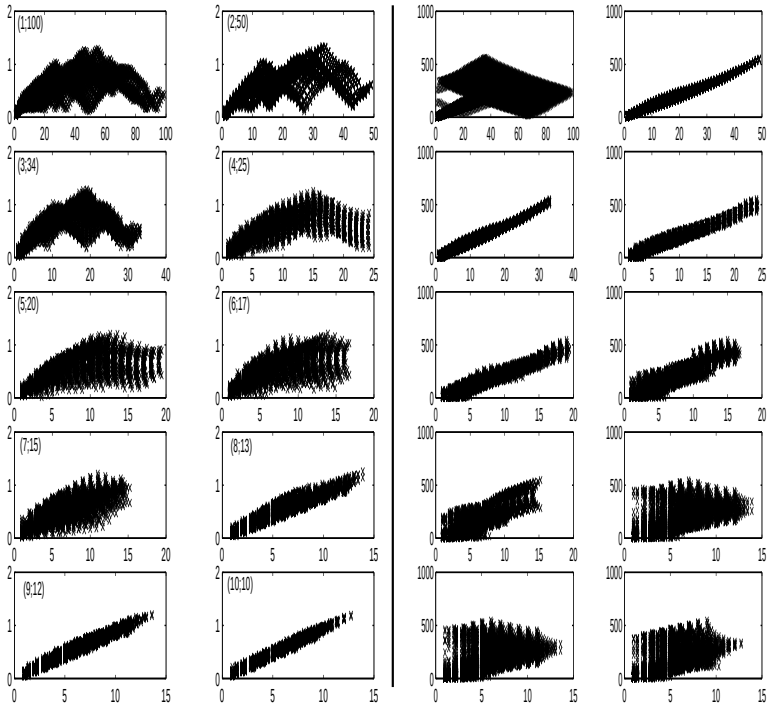

Figure 2: Scatter plots of the two distances for examples A and B)

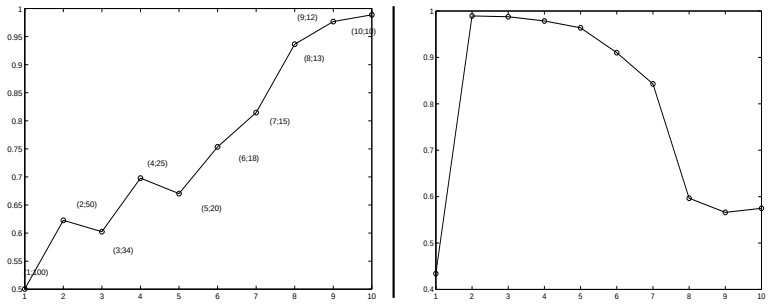

Figure 3: Correlation coefficient between the two distances for examples A and B

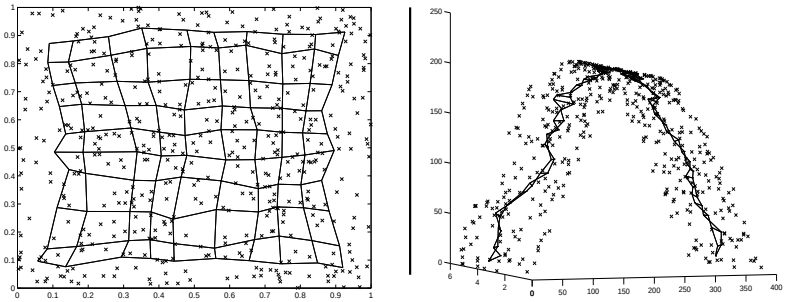

Figure 4: Kohonen map and data-set for maximum of correlation for examples A and B

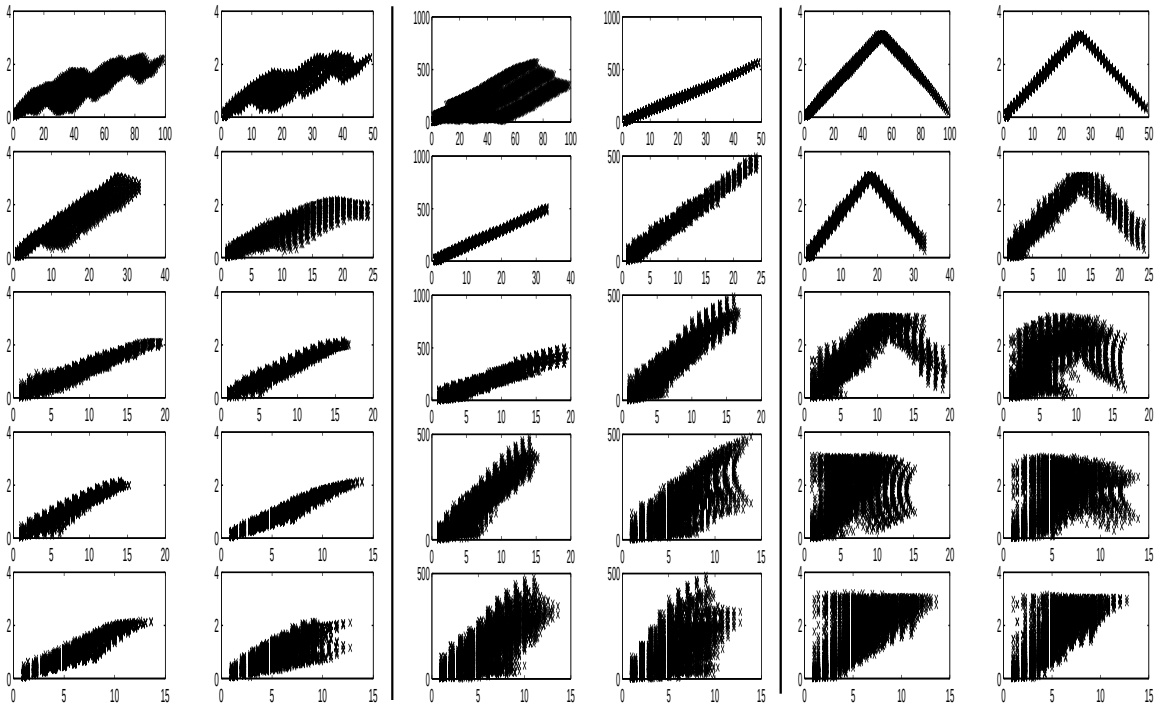

Figure 5: Scatter plot of the two distances for examples C,C' and D

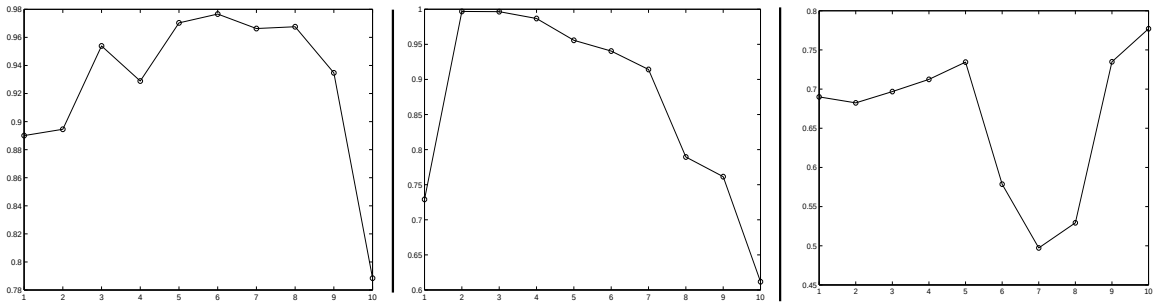

Figure 6: Correlation coefficient between the two distances for examples C,C' and D

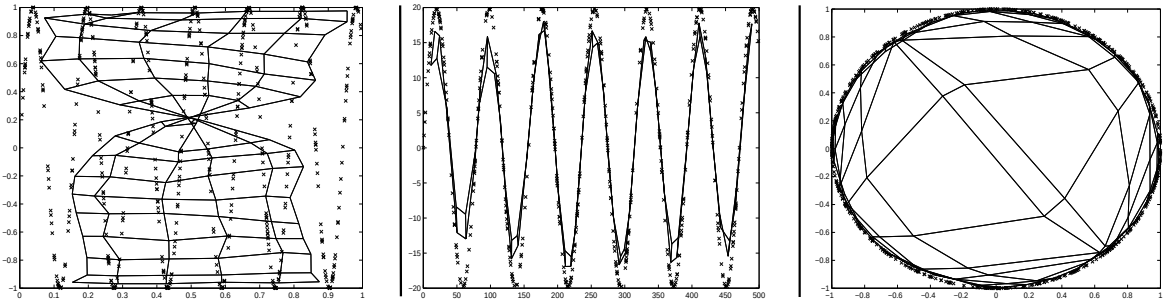

Figure 7: Kohonen map and data-set for maximum of correlation for examples C,C' and D

We can observe that, for examples A (i.e. we find what we expect to, knowing the "true" structure of data), for C' it is not so bad even if we'd prefer a string structure but the problem is due to over-fitting (see next section).
The fact that C fails and C' works underlines once again the importance of graph-based normalization to be able to observe data structure. Results are not so good for the horseshoe example (graph-based normalized), all Kohonen maps failed here and we should have tested more... Example D doesn't satisfy the initial hypothesis : it is not homeomorph to an hyper-rectangle but illustrates what happens when a direction is "circular" : quite fine scatter plot but low correlation coefficient.

4 Detection of over-fitting

Another interesting result is that over-fitting (too large number of cells as input) can be detected. In the passed section we observed the sinusoïde example for a string were not choose (the $(50; 2)$ map were prefer) it is due to over-fitting problem : $(100, 1)$ string fails and over-fit the data. To observe that effect we compute the algorithm for strings from size 10 to size 100. We can see that over-fitting effects happen when strings are longer than 70 for the same example as in previous section.

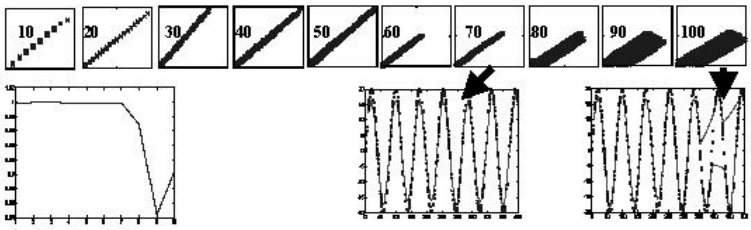

Figure 8: Detection of over-fitting for the sinusoïde example

5 Evolution of index through the Kohonen algorithm

When observing the correlation coefficient through steps of Kohonen algorithm we can observe, as in figure 9 a global increase of correlation.

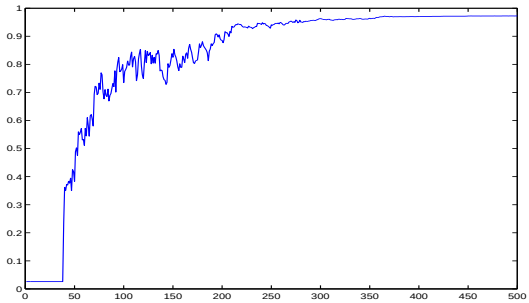

Figure 9: Evolution of correlation during SOM for the A exemple

6 Conclusion and Further work

The method gives encouraging results as for structure choice of Kohonen maps, as for overfitting detection. An hudge advantage of this method is that it is easily adaptable to higher map dimension than 2.
But there is still work to do : first we would like not to have to test a lot of maps and choose one of those tested but find a quicker way to find optimal parameters.
second, to get quicker, and may be also better results we would like to use growing neural gaz to compute the geodesic distance.
Finally, in a theoretical point of view, we would like to know if iteration of Kohonen algorithm does stochastically increase the correlation between the two distance, and if not, adapt the algorithm in this way.

References

[1] H-U Bauer and K. Pawelzic(1992), Quantifying the neigborhood preservation of self-organizing maps, *ieee transaction on Neural Network*, **vol. 3** p 570-579

p. 13-20.

[2] E. de Bodt, M. Cottrell and M. Verleysen (2002), Statistical tools to asses the reliability of self organizing maps *Neural Network*, **vol. 15** p. 967-978.

[3] P. Demartines (1994), Analyse des données par réseaux de neurones auto-organisés, *Thesis*

[4] E.W. Dijkstra (1951), A note on two problems in connection with graphs, *Numerische Mathematik*, **vol. 1** p. 269-271.

[5] J.A. Lee, A. Lendasse and M. Verleysen (2000), A global geometric framework for nonlinear dimensionality reduction, *proceedings of the* 8^{th} *European Symposium on Artificial Neural Networks*, **vol. 1** p. 13-20.

[6] J.B. Tenenbaum, V. de Silva (2000), A global geometric framework for non-linear dimensionality reduction, *Science*, **vol. 290** p. 2319-2323.

[7] T. Villmann, R. Der, M. Herrmann and T. Martinez (1997), Topology Preservation in Self-Organizing Feature Maps : Exact Definition and Measurement *ieee transaction on neural networks*, **vol. 8 N2** p. 256-266.

[8] S. Zrehen(1993), Analysing Kohonen maps with geometry *Proceedings of ICANN*, p 609-612

[9] J.A. Lee, A. Lendasse and M. Verleysen (2004), Nonlinear projection with curvilinear distances : Isomap versus curvilinear distance analysis *Neurocomputing*, **vol. 57** p. 49-76

[10] G.J. Goodhill, T. Sejnowski (1996), Quantifying neighbourhood preservation in topographic mapping *Proceeding of the 3rd joint Symposium on Neural Computation*, **vol. 1** p. 61-82