ESANN'2009 proceedings, European Symposium on Artificial Neural Networks - Advances in Computational Intelligence and Learning. Bruges (Belgium), 22-24 April 2009, d-side publi., ISBN 2-930307-09-9.

A Computational Framework for Exploratory Data Analysis

Axel Wismüller

Depts. of Radiology and Biomedical Engineering, University of Rochester, New York
601 Elmwood Avenue, Rochester, NY 14642-8648, U.S.A.

Abstract. We introduce the Exploration Machine (Exploratory Observation Machine, XOM) as a novel versatile method for the analysis of multidimensional data. XOM systematically inverts structural and functional components of so-called topology-preserving mappings. It provides a surprising flexibility to simultaneously contribute to complementary domains of unsupervised learning for exploratory pattern analysis, namely *both* structure-preserving dimensionality reduction *and* data clustering. We demonstrate XOM's applicability to synthetic and real-world data.

1 Introduction

To extract useful knowledge from high-dimensional observations, such as provided by multidimensional imaging data, scientists need to concisely visualize hidden regularities in such data by methods of intelligent *data reduction.* A classical machine learning approach to this problem has been contributed by so-called 'topology-preserving mappings' which have been pioneered by Kohonen's discovery of the Self-Organizing Map (SOM) algorithm almost three decades ago. This approach has found wide-spread use throughout science and technology for both visualization and partitioning of high-dimensional data. However, a significant drawback of the SOM is that it cannot accurately preserve an eventual cluster structure prevalent in the data as pointed out in [1].

In this contribution, we propose to systematically reverse the data-processing workflow in topology-preserving mappings in order to alleviate this problem. By simply exchanging functional and structural components of topology-preserving mappings, we obtain the Exploration Machine (Exploratory Observation Machine, XOM) as a novel computational framework for *both* structure-preserving dimensionality reduction *and* data clustering. After introducing the XOM algorithm, we analyze its applicability to both synthetic and real-world data.

2 The Exploration Machine Algorithm

The Exploration Machine (XOM) algorithm can be resolved into three steps. For simplicity, let us first consider N real-valued input vectors $\mathbf{r}_i$ in the 'observation space' O, each of dimensionality D.

Step 1: Define the topology of the input data in the observation space O by computing distances $d(\mathbf{r}_i, \mathbf{r}_j)$ between the data vectors $\mathbf{r}_i, i \in \{1, \ldots, N\}$. This step is omitted, if the input data is already given as a set of distances between

input data items.

Step 2: Define a 'hypothesis' on the structure of the data in the embedding space E, represented by 'sampling' vectors $\mathbf{x}_k \in E$, $k \in \{1, \ldots, K\}, K \in \mathbb{N}$, and randomly initialize an 'image' vector $\mathbf{w}_i \in E$, $i \in \{1, \ldots, N\}$ for each input vector $\mathbf{r}_i$. Typical choices for sampling distributions are: for structure-preserving visualization, use a uniform distribution (e.g. in a 2D square, as used in figs. 2 and 3A,B); for data clustering use a mixture of several (e.g. Gaussian) distributions with different centers, e.g. located on the vertices of a regular simplex.

Step 3: Reconstruct the topology induced by the input data in O by moving the image vectors in the embedding space E using the computational scheme of a topology-preserving mapping T. The final positions of the image vectors $\mathbf{w}_i$ represent the output of the algorithm.

In the third step of the XOM algorithm, the topology-preserving mapping T can be considered as a free variable. Besides topographic vector quantizers, e.g. [2], a simple choice for T is Kohonen's self-organizing map algorithm [1], e.g. in its basic incremental version. Here, the image vectors $\mathbf{w}_i$ are incrementally updated by a sequential learning procedure. For this purpose, the neighborhood couplings between the input data items are represented by a so-called cooperativity function ψ. A typical choice for ψ is a Gaussian

$$\psi(\mathbf{r}, \mathbf{r}'(\mathbf{x}(t)), \sigma(t)) := \exp\left(-\frac{(\mathbf{r} - \mathbf{r}'(\mathbf{x}(t)))^2}{2\sigma(t)^2}\right). \tag{1}$$

In the XOM context, $\mathbf{r}'(\mathbf{x}(t))$ represents the 'best-match' input data vector. For a randomly selected sampling vector $\mathbf{x}(t) \in E$, this best-match input data vector is identified by $\|\mathbf{x} - \mathbf{w}_{\mathbf{r}'}\| = \min_{\mathbf{r}} \|\mathbf{x} - \mathbf{w}_{\mathbf{r}}\|$. Once the best-match input data vector $\mathbf{r}'(\mathbf{x}(t))$ has been identified, the image vectors $\mathbf{w}_{\mathbf{r}}$ are updated in a sequential adaptation step according to the learning rule $\mathbf{w}_{\mathbf{r}}(t+1) = \mathbf{w}_{\mathbf{r}}(t) + \epsilon(t)\,\psi(\mathbf{r}, \mathbf{r}'(\mathbf{x}(t)), \sigma(t))\,(\mathbf{x}(t) - \mathbf{w}_{\mathbf{r}}(t))$, where t represents the iteration step, $\epsilon(t)$ a learning parameter, and $\sigma(t)$ a measure for the width of the neighborhood taken into account by the cooperativity function ψ. In general, $\sigma(t)$ as well as $\epsilon(t)$ are changed in a systematic manner depending on the number of iterations t by some appropriate annealing scheme, e.g. an exponential decay with t, as used in the examples of this paper. The algorithm is terminated, once a problem-specific cost criterion is satisified, or a maximum number of iterations has been completed. – Although the above computational scheme formally resembles Kohonen's self-organizing map algorithm, both approaches actually describe fundamentally different concepts. Specifically, the meaning of the variables $\mathbf{r}$, $\mathbf{w}$, and $\mathbf{x}$ *completely differs* in the Exploration Machine: Whereas in Kohonen's algorithm the sampling vectors $\mathbf{x}$ represent the input data, this role is attributed to the vectors $\mathbf{r}$ in XOM, i.e. XOM completely inverts the role of input data and structure hypotheses, given the conventions of topology-preserving mappings as known from the literature. For detailed analyses, extensions, and variants of the Exploration Machine, such as related to computational complexity, convergence properties, free parameter selection, supervised learning, analysis of non-metric data, geodesic coordinates, out-of-sample extension, growing XOM variants, and constrained incremental learning, we refer to [3].

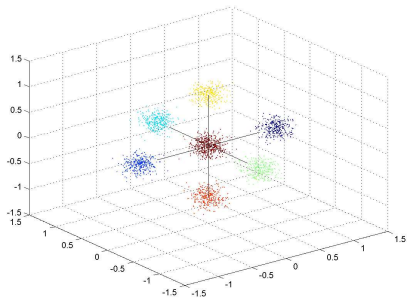


Fig. 1: 'Hepta' Data Set. For explanation, see text.

(A) (B)

Fig. 2: XOM embedding of the 'Hepta' data sets. The figure depicts the best (A) and the worst (B) embedding result obtained by XOM for the 40 data sets constructed according to the specifications explained in Fig. 1.

3 Experiments

Hepta Data Sets. In order to quantitatively evaluate the quality of dimensionality reduction by XOM and to relate its results to other methods known from the literature, we investigated the degree of structure preservation which can be achieved by classical and advanced recent nonlinear embedding methods, namely Principal Component Analysis (PCA), Locally Linear Embedding [4], and Isomap [5]. For this purpose, we used 40 data sets similar to a synthetic benchmark data set called 'Hepta' proposed in [6] for the evaluation of structure preservation. — A single realization of a 'Hepta' data set is depicted in Fig. 1. It consists of 2300 points randomly sampled from seven Gaussian distributions, thus forming 'clusters' in $\mathbb{R}^3$. The centroids of the six non-central Gaussian distributions span the coordinate axes of the $\mathbb{R}^3$, the respective clusters consist of 300 data points each. The central Gaussian distribution consists of 500 data points. We created 40 'Hepta' data sets according to these specifications.

To quantitatively evaluate structure preservation, we used Sammon's error function [7]. Our results are summarized in Tab. 1. On average, XOM outper-

Table 1: Comparative evaluation of computation times and structure preservation for nonlinear embedding of 40 'Hepta' data sets as specified in Fig. 1. The table lists average computation times using an ordinary PC (Intel Pentium 4 CPU, 1.6 GHz, 512 MB RAM). Average and minimum values as well as the standard deviation of Sammon's error E' were computed as a measure of structure preservation. The free parameters of all the methods examined in the comparison (except PCA) were optimized to obtain the best results, i.e. to minimize E'. Note that XOM yields competitive structure preservation at acceptable computation times.

Method	Comp. Time (s)	E'	$\min(E')$	$\sigma(E')$
Isomap	468	$1.851 \cdot 10^5$	$1.319 \cdot 10^5$	$0.384 \cdot 10^5$
LLE	11600	$3.681 \cdot 10^5$	$1.776 \cdot 10^5$	$1.267 \cdot 10^5$
PCA	0.3	$2.216 \cdot 10^5$	$1.279 \cdot 10^5$	$0.608 \cdot 10^5$
XOM	4.6	$1.732 \cdot 10^5$	$1.426 \cdot 10^5$	$0.247 \cdot 10^5$

formed LLE and Isomap with regard to structure preservation, although Isomap yielded better results in a few data sets. Interestingly, we frequently obtained poor results for PCA. This is caused by the spatial symmetry of the data set which makes the projection axis in PCA very sensitive to noise, i.e. to the random choice of data points sampled from the Gaussian distributions specified in the 'Hepta' data set construction. Thus, different clusters are frequently projected onto each other in the embedding result, i.e. cannot be separated, which leads to impaired structure preservation. For illustration, the best and the worst of the 40 embedding results obtained by XOM are shown in Fig. 2. — We emphasize that the results depicted in Tab. 1 depend on the structure of the data set, and do not allow to draw final conclusions on the overall performance of the nonlinear embedding algorithms with regard to the general degree of structure preservation or their computational expense. In addition, the choice of other measures for structure preservation may also result in different ranking scenarios. For example, we conjecture that PCA will be superior in situations where the data is approximately located in a linear subspace of the observation space. LLE and Isomap will perform better in situations where the data is not distributed inhomogeneously in the observation space, i.e. does not exhibit an underlying distinct cluster structure of almost isolated data patches, but rather consists of a 'connected' single cluster. In such data sets, both LLE and Isomap can accurately reconstruct the data with a smaller number of nearest neighbors, which will also reduce their computational expense considerably.

However, even taking all these limitations into account, our investigation at least shows that there exist classes of data sets where XOM yields competitive results in comparison to the methods known from the literature.

Microarray Gene Expression Data. Fig. 3A shows the visualization result obtained by XOM for structure-preserving dimensionality reduction of gene expression profiles related to ribosomal metabolism, as a detailed visualization of a gene subset included in the genome-wide expression data taken from Eisen et

ESANN'2009 proceedings, European Symposium on Artificial Neural Networks - Advances in Computational Intelligence and Learning. Bruges (Belgium), 22-24 April 2009, d-side publi., ISBN 2-930307-09-9.

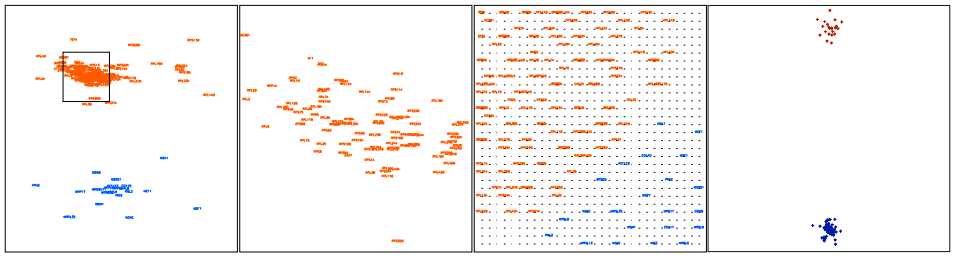
al. [8]. The figure illustrates the exploratory analysis of the 147 genes labeled as '5' (22 genes) and '8' (125 genes) according to the cluster assignment by Eisen et al. [8]. Besides several genes involved in respiration, cluster '5' (blue) contains genes related to mitochondrial ribosomal metabolism, whereas cluster '8' (orange) is dominated by genes encoding ribosomal proteins.

In the XOM genome map of Fig. 3A, it is clearly visible at first glance that the data consists of two distinct clusters. Comparison with the functional annotation known for these genes reveals that the map overtly separates expression profiles related to mitochondrial and to extramitochondrial ribosomal metabolism. Fig. 3B shows an enlarged section of the map indicated by the small frame in Fig. 3A. Fig. 3C shows a data representation obtained by a SOM on the same data, using a regular grid of 30×30 'neurons'. As can be clearly seen in the figure, the SOM *cannot* achieve a structure-preserving mapping result as provided by the Exploration Machine in Fig. 3A: Although the genes related to mitochondrial and to extramitochondrial ribosomal metabolism are collocated on the map, the distinct cluster structure underlying the data remains invisible, if the color coding is omitted. In other words, visualization by the Exploration Machine outperforms the SOM w.r.t. structure preservation in this example.

For quantitative comparison of results we computed Sammon's error E' [7] as a quantitative measure of structure preservation for several other embedding algorithms known from the literature. We obtained $2.21 \cdot 10^3$ (1.00) for XOM, $2.45 \cdot 10^3$ (1.11) for Sammon's mapping [7], $2.77 \cdot 10^3$ (1.25) for Locally Linear Embedding (LLE) [4], $2.82 \cdot 10^3$ (1.28) for PCA, $3.36 \cdot 10^3$ (1.52) for Isomap [5], and $10.19 \cdot 10^3$ (4.61) for SOM, where numbers in brackets denote relative values compared to XOM. Computation times in seconds were 0.72, 8.52, 1.36, 0.03, 0.27, 988.21 for XOM, Sammon, LLE, PCA, Isomap, SOM, respectively. These results show that XOM yields competitive structure preservation results at acceptable computation time. – Fig. 3D shows how XOM can be used for *clustering* of this data as well. To this end, the structure hypothesis in step 2 of the XOM algorithm is changed from a uniform distribution in a unit square as used in Fig. 3A to a set of two Gaussians centered at different locations of the exploration space, e.g. on top and bottom of the embedding space in Fig. 3D. As can be seen, the two clusters are separated completely. Note that the decision to use *two* clusters – instead of any different number of clusters – can be conveniently based on the results of structure-preserving XOM visualization in Fig. 3A, which can, thus, serve as a useful preprocessing step to clustering. The visualization obtained by SOM in Fig. 3C, in contrast, is not clearly indicative for the presence of exactly two clusters, if the color coding is omitted.

4 Conclusion and Outlook

We have introduced the Exploration Machine as a novel versatile learning method for the analysis of multidimensional data. As can be concluded from our work, XOM yields competitive results when compared to established methods known from the literature. In addition, it is evident that XOM can *directly* be applied to both nonlinear embedding and clustering of *non-metric* data, as shown



(A) (B) (C) (D)

Fig. 3: Visualization of genome expression profiles related to ribosomal metabolism using the Exploration Machine. (A) Genome map obtained by structure-preserving dimensionality reduction using the Exploration Machine. (B) Enlarged section of (A). (C) Data representation obtained by SOM. (D) XOM clustering result. For explanation, see text.

in [3]. As an outlook, it is important that XOM can be used to systematically reverse other concepts related to topology-preserving learning. Obvious examples are batch and growing XOM variants [3] and the Fuzzy-Labeled XOM with Relevance Learning which is easy and straightforward to derive as a systematic XOM inversion of [9], thus linking the XOM framework to LVQ. Although future systematic scientific investigation has to further elucidate the properties of our method, we conjecture that XOM can constructively contribute to the field of learning with neural maps.

References

[1] Kohonen, T., [*Self-Organizing Maps*], Springer, Berlin, Heidelberg, New York, 3rd ed. (2001).

[2] Graepel, T., Burger, M., and Obermayer, K., "Self-organizing maps: Generalizations and new optimization techniques," *Neurocomputing* **21**, 173–190 (1998).

[3] Wismüller, A., *Exploratory Morphogenesis (XOM): A Novel Computational Framework for Self-Organization*, Ph.D. thesis, Technical University of Munich, Department of Electrical and Computer Engineering (2006).

[4] Roweis, S. and Saul, L., "Nonlinear dimensionality reduction by locally linear embedding," *Science* **290**(5500), 2323–2326 (2000).

[5] Tenenbaum, J., de Silva, V., and Langford, C., "A global geometric framework for nonlinear dimensionality reduction," *Science* **290**(5500), 2319–2323 (2000).

[6] Ultsch, A., "Maps for the visualization of high-dimensional data spaces," in [*Proc. of the Workshop on Self-Organizing Maps 2003 (WSOM03)*], 225–230 (2003).

[7] Sammon, J., "A nonlinear mapping for data structure analysis," *IEEE Transactions on Computers* **C 18**, 401–409 (1969).

[8] Eisen, M., "Cluster analysis and display of genome-wide expression patterns," *Proc. Natl. Acad. Sci. USA* **95**, 14863–14868 (1998).

[9] Villmann, T., Seiffert, U., Schleif, F., Brüß, C., Geweniger, T., and Hammer, B., "Fuzzy labeled self-organizing map with label-adjusted prototypes," in [*ANNPR 2006, LNAI 4087*], 46–56, Springer-Verlag, Berlin (2006).