ESOM-Maps: tools for clustering, visualization, and classification with Emergent SOM

Alfred Ultsch, Fabian Mörchen
Data Bionics Research Group, University of Marburg

D-35032 Marburg, Germany

March 17, 2005

Abstract

An overview on the usage of emergent self organizing maps is given. U-Maps visualize the distance structures of high dimensional data sets. P-Maps show their density structures and U*-Maps combine the advantages of the mentioned maps to a visualization suitable to detect non-trivial cluster structures. A concise summary on the usage of Emergent Self-organizing Maps (ESOM) for data mining is given. The tasks of visualization, clustering, and classification as they can be performed with the *Databionics ESOM Tools* are described.

1 Introduction

The power of self-organization that allows the emergence of structure in data is often neglected. We think this is in part due to a misusage of Self-organizing Maps that is widely spread in the scientific literature. The maps used by most authors are usually very small, consisting of somewhere between a few and some tens of neurons (e.g. [1, 2, 3]). Also, the concept of boundless maps (e.g. toroid maps [4]) to avoid border effects is rarely used. Using small SOM is almost identical to a k-Means clustering with k equal to the number of nodes in the map. The topology preservation of the SOM projection is of little use when using small maps. Emergent phenomena involve by definition a large number of individuums, where large means at least a few thousands. This is why we use large SOMs and called them Emergent Self-Organizing Maps (ESOM) to emphasize the distinction. It has been demonstrated, that using ESOM is a significantly different process from using k-Means [5].

The purpose of this article is to give a concise summary on the usage of Emergent Self-organizing Maps (ESOM) for data mining. We describe the tasks

We thank Mario Nöcker, Christian Stamm, Michael Thies, and Martin Kümmerer for programming most of the Databionics ESOM Tools.

of visualization, clustering, and classification as they can be performed with our new software - the *Databionics ESOM Tools*.

2 Map Architecture

If $M \subset \mathbb{R}^2$ the map has *planar* topology. In a *quadgrid* arrangement the number of immediate neighbors of a neuron is 4, for *hexgrids* there are 6 immediate neighbors.

Note, that the number of immediate neighbors is less at the borders of the grid. In these map spaces border effects occur, enlargening the probability of topology errors [6]. To avoid such border effects, grids can be embedded in a finite but *boundless* space e.g. a sphere or the toroid PAC-man space (PMS)(see [4]). In PMS the top row is connected to the bottom row and the left column to the right column within the lattice.

If the number of rows and columns in a 2D grid is equal, the map is called *square*, otherwise *rectangular*. The ratio of rows to columns is proposed to be chosen corresponding to the ratio of the first and second eigenvalues of the covariance matrix in [7]. An experimental analysis of topology errors showed, however, that the ratio of rows and columns should be chosen different from unity even when no dominant direction of variance exists [6].

We recommend the following ESOM architecture: boundless toroid grids with at least 4000 neurons and a ratio of rows and columns different from unity. This avoids border effects, topology errors, and enables an intuitive undistorted visualization.

3 Visualization

The result of ESOM training [7] is a low dimensional grid of high dimensional prototype vectors. The positions of the bestmatches for the data points alone does often not offer an intuitive visualization of the structures present in the high dimensional space. Additional methods are needed to visualize the structures, the most common being height values forming a 3D landscape on top of the grid.

The example to demonstrate the visualizations is taken from [8]. It consists of a 2 dimensional dataset typical for sonar applications. The dimensions are ENGY and TIME, the data set is a mixture of two Gaussians with 2048 points each (see Figure 1(a)).

Distance-based Visualization: The U-Matrix [9] is the canonical display of ESOM. The local distance structure is displayed at each neuron as a height value creating a 3D landscape of the high dimensional data space. The height is calculated as the sum of the distances to all immediate neighbors normalized by the largest occurring height. This value will be large in areas where no or few data points reside, creating mountain ranges for cluster boundaries. The sum will be small in areas of high densities, thus clusters are depicted as val-

leys. Figure 1(b) shows the U-Matrix of the EngyTime dataset, darker colors correspond to large distances.

Density-based Visualization: While distance-based methods usually work well for clearly separated clusters, problems can occur with slowly changing densities and overlapping clusters. Density-based methods more directly measure the density in the data space sampled at the prototype vectors. The P-Matrix [4] displays the local density measures with the Pareto Density Estimation (PDE) [10], an information optimal kernel density estimation. Figure 1(c) shows the P-Matrix of the EngyTime dataset, darker colors correspond to larger densities.

Distance- and Density-based Visualization: In dense regions of the data space the local distances depicted in an U-Matrix are presumably distances measured inside a cluster. Such distances may be disregarded for the purpose of clustering. In thin populated regions of the data space, however, the distances do matter. In this case the U-Matrix heights correspond to cluster boundaries. This leads to the definition of an U*-Matrix [11] which combines the distance based U-Matrix and the density based P-Matrix. The values of the U-Matrix are dampened in highly dense regions, unchanged in regions of average density, and emphasized in sparse regions.

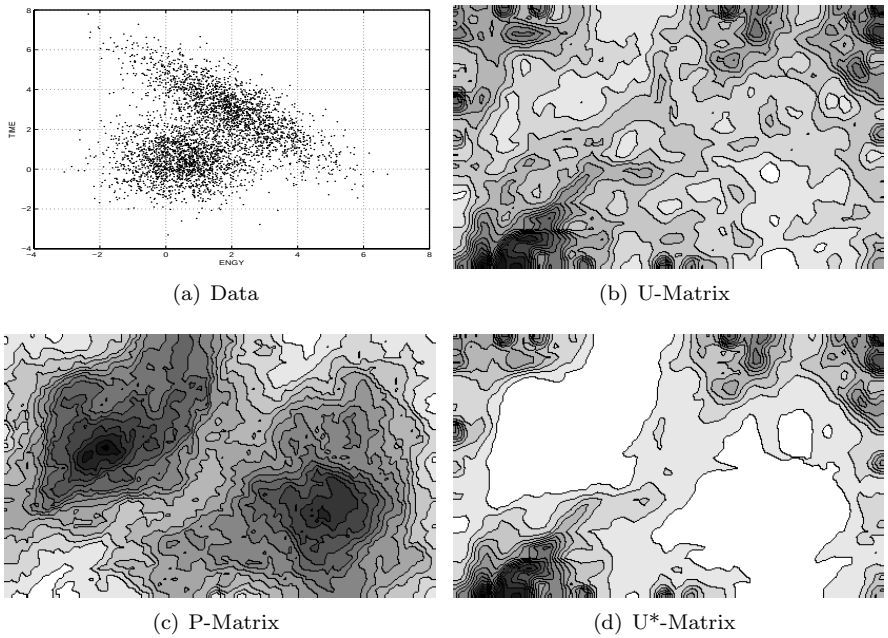
The advantage of the U*-Matrix over the U-Matrix in datasets with clusters that are not clearly separated in the high dimensional space can be see from the example. The U*-Matrix in Figure 1(d) shows clearly the two Gaussians of the EngyTime dataset. The U-Matrix in Figure 1(b), however, may mislead a clustering.

Topology-based Visualizations: There are two kinds of topographical errors that have to be considered. The forward projection error (FPE) occurs when a pair of similar data points is assigned to a distant pair of positions on the map. The backward projection error (BPE) corresponds to a pair of close grid positions being the image of two distant data points. The latter can be visualized using Zrehen's Measure [12], for the former the Minimal-U-Path [6] can be used.

Bestmatch Visualizations: A display of the bestmatches can be placed on top of the images created by the above mentioned methods. Each neuron that is a bestmatch for at least one data point can be marked with a point. The point can be colored indicating a known or chosen cluster membership.

4 From Matrix to Map

For boundless grids the visualizations should be viewed in tiled mode [4], displaying four adjacent copies of the grid. Clusters stretching over the edge of the grid image can then be seen connected. Unfortunately every data point and cluster is shown in 4 different places. This can be compensated by extracting a map from the tiled view removing the redundancies [4]. The obvious resemblance with geographical landscapes led us to call these displays *maps*. For an U-Matrix we get an U-Map and similar P-Maps and U*-Maps. The set of all these maps is called ESOM-Maps. The resemblance of ESOM-Maps to geo-



(a) Data (b) U-Matrix

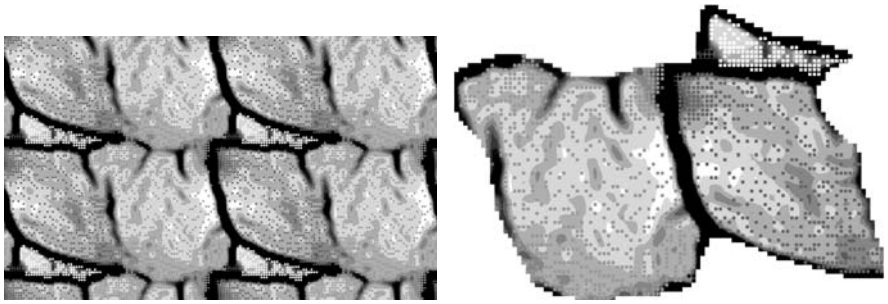
(c) P-Matrix (d) U*-Matrix

Figure 1: ESOM visualizations for 2 clusters in the EngyTime dataset

graphical maps or landscapes may be enhanced by computer graphical means such as texturing, coloring and lightening. The Figure 2(a) shows a tiled view of an U*-Matrix with 3 clusters. Figure 2(b) shows the non-redundant U*-Map display extracted from the center of the tiled view.

5 Clustering

The clustering of the ESOM can be performed at two different levels. First, the bestmatches and thus the corresponding data points can be manually grouped into several clusters. Not all points need to be labeled, outliers are usually easily detected and can be removed. The cluster membership can be visualized by a coloring of the bestmatches. Secondly, the neurons can be clustered. This way regions on the map representing a cluster can be identified and used for classification of new data (see Section 6). The regions can be visualized by semi-transparent colored regions in order not to overlay the background visualizations completely. The visualization can show whether there actually is a cluster structure. Outlying data points will also be easily detectable. In contrast, the popular k-Means algorithm will always converge to the given number of clusters no matter whether the data supports this structure. Outliers will be handled like any other data point.



(a) Tiled U*-Matrix (b) Extracted U*-Map

Figure 2: ESOM visualizations for 3 clusters in the Skating data

6 Classification

Labeling all (or most) neurons instead of only the bestmatches creates a subsymbolic classificator similar to a k-nearest neighbor (KNN) classifier with $k = 1$ that can be applied to new data automatically. The main difference to KNN is, that the user can use the visualization of the ESOM to create the labeling whereas KNN does not offer this convenience. Further KNN classification always classifies a point, no matter how near (or rather far) the neighbors are. In contrast, ESOM classification offers a *don't know* class by leaving neurons unlabeled, e.g. for sparsely populated regions separating clusters.

We have successfully used ESOM classification to detect movement phases in a multivariate time series from Inline-Speed Skating [13, 14]. Based on the results of a single skating speed we tried to extend the analysis to six different speeds. Each dataset contained 30K 6-dimensional vectors. We used a pooled sample from all speeds for training. Three clusters were detected, the corresponding regions on the map were labeled. The U*-Matrix and the U*-Map are shown in Figure 2(a) and Figure 2(b), respectively. The six full datasets were then classified into the three movement phases by bestmatch projection on the map. The resulting segmentation of the time series was labeled by an expert into the movement phases *glide*, *push*, and *swing*.

7 Databionics ESOM Tools

Most of the described methods for visualization, clustering, and classification have been implemented in a software tool called *Databionics ESOM Tools*[1]. The main part is a graphical front-end to perform tasks like visualization, outlier removal, clustering of bestmatches and neurons, creation of non-redundant map views. The visualization of the maps is designed in a modular way. The different background displays of the map can be applied to subsets of the data features and can be combined with foreground displays of the bestmatches. Several color

[1] `http://databionic-esom.sf.net`

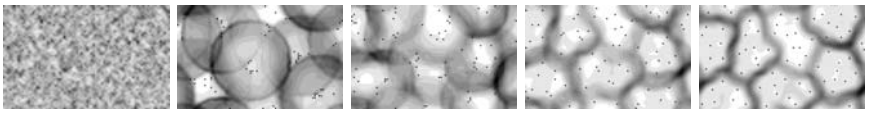


Figure 3: U-Matrix during ESOM training for 7 clusters (Epochs $0 + k * 5$)

gradients with an adjustable number of colors are available as well as contour rendering. Even the training process can be visualized by creating a slide show of the map unfolding in the data space (see Figure 3). The training, rendering, projection, and classification tasks can also be run from the command line to allow an automatic analysis of large data sets.

The programs are written in Java and the source code is available under the GPL[15]. We welcome other scientists to get involved in the further development of the tool.

8 Summary

Emergent SOMs are a powerful tool for clustering and classification. To exploit nontrivial emergent phenomena, however, large maps must be used. Only large maps can adequately project complicated structures onto the two dimensional map space. Distance-based and density-based visualization of ESOM can be used to analyze the data for possible clusters. A (partly) labeled ESOM serves as a sub-symbolic classificator in the spirit of KNN classification. Many ESOM related tasks can be performed with the freely available *Databionics ESOM Tools*.

References

[1] C. Li, P. S. Yu, and V. Castelli. MALM: a framework for mining sequence database at multiple abstraction levels. In *Proc. of the 7th ACM CIKM*, pages 267–272, Bethesda, MD, 1998.

[2] T. C. Fu, F. L. Chung, V. Ng, and R. Luk. Pattern Discovery from Stock Time Series Using Self-Organizing Maps. *Workshop Notes of KDD2001 Workshop on Temporal Data Mining, San Francisco*, pages 27–37, 2001.

[3] E. Pampalk, A. Rauber, and D. Merkl. Using Smoothed Data Histograms for Cluster Visualization in Self-Organizing Maps. In *Proc. ICANN'02*, Madrid, Spain, August 27-30 2002. Springer.

[4] A. Ultsch. Maps for the Visualization of high dimensional Data Spaces. In *Proc. WSOM'03, Japan*, 2003.

[5] A. Ultsch. Self Organizing Neural Networks perform different from statistical k-means clustering. In *Proc. GfKl 1995, Basel, Swiss.*

[6] A. Ultsch and L. Herrmann. Architecture of emergent self-organizing maps to reduce projection errors. In *Proc. ESANN, Bruges, Belgium*, 2005.

[7] T. Kohonen. *Self-Organizing Maps.* Springer, 1995.

[8] P. M. Baggenstoss. Statistical Modeling Using Gaussian Mixtures and HMMs with Matlab, `http://www.npt.nuwc.navy.mil/Csf/htmldoc/pdf/pdf.html`, 2002.

[9] A. Ultsch. Self-Organizing Neural Networks for Visualization and Classification. In *Proc. GfKl 1992, Dortmund, Germany.*

[10] A. Ultsch. Pareto Density Estimation: Probability Density Estimation for Knowledge Discovery. In *Proc. GfKl 2003, Cottbus, Germany*, 2003.

[11] A. Ultsch. U*-Matrix: a Tool to visualize Clusters in high dimensional Data. Technical Report 36, CS Department, Philipps-University Marburg, Germany, 2004.

[12] S. Zrehen. Analyzing Kohonen Maps With Geometry. In *Proc. of the International Conference on Artificial Neural Networks (ICANN'93).* Springer, 1993.

[13] F. Mörchen, A. Ultsch, and O. Hoos. Discovering interpretable muscle activation patterns with the Temporal Data Mining Method. In *Proc. PKDD 2004, Pisa, Italy*, 2004.

[14] F. Mörchen, A. Ultsch, and O. Hoos. Extracting interpretable muscle activation patterns with time series knowledge mining. *International Journal of Knowledge-Based & Intelligent Engineering Systems*, 2005.

[15] GNU General Public License, `http://www.gnu.org/licenses`.