U-maps: topograpic visualization techniques for projections of high dimensional data

Alfred Ultsch[1] and Fabian Mörchen[2]

[1] Databionic Research Group,
Philipps-University Marburg, D-35032, Germany
[2] Siemens Corporate Research,
Princeton, NJ, 08540, USA***

Abstract. The visualization of distance structures in high dimensional data as topographic maps (U-matrix) is a standard method for Emergent Self Organizing Maps (ESOM). This work describes the extension of this visualization to other projections like principal component analysis (PCA), independent component analysis (ICA), multidimensional scaling (MDS), Sammon's mapping, or Isomap. Each of the methods optimize different criteria in the projection to a low dimensional space that are desirable in certain applications. The results are commonly displayed as two-dimensional scatterplots. This visualization does not indicate if and where the mapping is folded, i.e. where the projected distances are displayed compressed compared to the original distances. The U-map technique presented here visualizes the original high-dimensional distances in the low-dimensional projection space using 3-dimensional landscapes. These landscapes can be explored using the paradigm of topograpic maps. High ridges indicate foldings of the projections. The added value of the U-map visualization technique for low dimensional projections is demonstrated using various synthetic and real data sets.

Key words: visualization, scatter plot, projection, distances, U-map

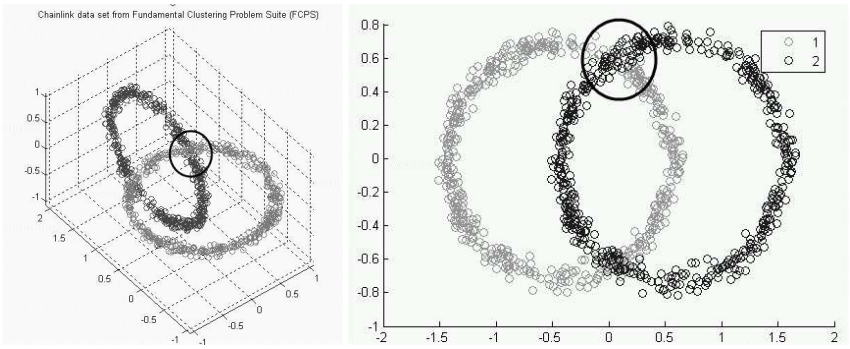
1 Introduction

In order to get an impression on the structural features of high dimensional data, in particular for data mining, it is a good idea to project the data in a low dimensional space where the data can be visualized. If the projection is able to conserve the distance relationships of the data in low dimensions, such a visualization gives a hint on structural properties of the data. Projections that reduce the dimensionality cannot, however, conserve all distance relationships proportionally. The distance relations distorted in the image of the data. These projection errors are usually called foldings of the projection [8]. In this paper we propose a method to visualize the foldings. The method is similar to the U-matrix [3] technique for Emergent Self-Organizing Maps (ESOM) [7,9]. The rest of the paper is organizes as follows. Section 2 repeats the basic properties of projections. Section 3 defines the algorithm for the construction of U-maps. Section 4 and Section 5 show examples of U-maps for some popular projections.

*** This research was performed at the Databionic research group in Marburg

2 Projections

There are many methods to project data from high dimensional spaces to low dimensions. Among these methods are linear projections such as principal components analysis (PCA) [2] and independent component analysis (ICA) [5]. Popular nonlinear projections are multidimensional scaling (MDS) [4], Sammon's mapping [1] and Isomap [6]. ESOM create a nonlinear and discontinuous mapping which is able to preserve topographic structures such as clusters [3,9]. In order to get a correct visual impression about the data in the high dimensional space the distance relations in the projected space should be correlated to the distances in the original space. It is impossible, however, to proportionally conserve all the distance relationships for a projection from a higher to a lower dimensionality. Some of the original distances are well represented, while others are more or less distorted. For MDS and Sammon's mapping the projections are constructed minimizing some formulation of the global projection error. This global minimization can have large local errors. Consider for example the Chainlink data set, first published in [11]. It consists of two torus shaped clusters intertwined like links in a chain and well separated in 3D. Figure 1(a) shows the 3D data set, Figure **??** shows a 2D MDS projection mapping of the data [1].



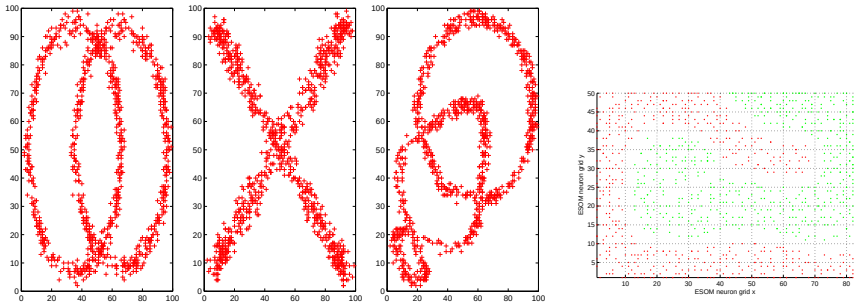
(a) Original 3D data (b) 2D MDS projection

Fig. 1. Chainlink dataset with two clearly separated but entangled classes. The points in the circled region are well separated in 3D but very close in the projection.

MDS minimizes the sum of the squared differences between the original distances and the distances in the projected space. Consider the location marked with a circle in Figure 1(a) and Figure 1(b), respectively. In the original space the distances between the points from the two different classes are very large but in the projection the distances are vanishing. Looking at the projected points one gets the impression that the data are from one common cluster instead of two different groups. This

[1] Note to the reviewers: all figures will be replaced with black and white images using different glyphs to the classes in the final version

means in the region marked with a circle in Figure 1(b) the globally optimized projection produces large local errors. The is observed for other projection methods applied to this data. Figure 2 shows the Chainlink dataset projected with PCA, ICA, Sammon's mapping, and ESOM. It is interesting to see that ESOM is the only projection which disentangles the two clusters. Sammon's mapping even entangles one of the clusters with itself (see Figure 2(c)).



(a) PCA (b) ICA (c) Sammon (d) ESOM

Fig. 2. Chainlink projections with different methods

3 U-Maps

Given a high dimensional dataset $X = \{x_i = \langle x_{i,1}, ..., x_{i,d}\rangle | i = 1, ..., N\} \subset \mathbb{R}^d$ $d > 2$ and an arbitrary two-dimensional projection $P = \{p_i = \langle p_{i,1}, p_{i,2}\rangle | i = 1, ..., N\} \subset \mathbb{R}^2$ we calculate the U-Map of this projection in five steps.

Step 1: A $r \times r$ grid of uniformly spaced points is defined within the range of the projected values, i.e.

$$G = \{g_j\} = \{\ (min_i\{p_{i,1}\} + k(max_i\{p_{i,1}\} - min_i\{p_{i,1}\}), \quad (1)$$

$$min_i\{p_{i,2}\} + l(max_i\{p_{i,2}\} - min_i\{p_{i,2}\}))\ \ |k, l = 1, ..., r\} \quad (2)$$

Step 2: Each projected point p_i is assigned to the closest grid point, the so-called bestmatch, with the function

$$bm(p_i) = argmin_j\{D(p_i, g_j) | j = 1, ..., r^2\} \quad (3)$$

where D is the Euclidean distance in 2D.

Step 3: Let $P_j \subseteq P$ be the set of points assigned to the grid point g_j. For each g_j with $|P_j| > 0$ create a prototype vector $\hat{x}_j = \frac{1}{|P_j|} \sum_{p_i \in P_j} p_i$ (where Σ is the vector sum in $\mathbb{R}^d$).

Step 4: For all g_j with $|P_j| = 0$ calculate $\hat{x}_j$ with a high dimensional interpolation method using the prototype vectors already determined in step 3. This can be performed by applying Self-Organizing Map training without changing the bestmatch

assignments from step 2. In the t-th training epoch, the data points are shuffled and for each point p_j the prototypes are adjusted via

$$\hat{x}_k(t) = \hat{x}_k(t-1) - \lambda(t) h\left(D\left(g_k, g_{bm(p_j)}\right), t\right)\left(p_j - \hat{x}_k(t-1)\right) \qquad (4)$$

where $\lambda(t)$ and $h(\cdot, t)$ are the learning rate and the neighborhood kernel at time t, respectively. A typical choice for h is a Gaussian kernel cut off at 2 standard deviations. The prototype $\hat{x}_j$ and more importantly the prototypes neighboring it on the grid are thus drawn slightly towards p_j during each training step. Performing many training epochs with decreasing learning rate and kernel radius creates a high dimensional interpolation of the data set X preserving the topology of the projection P sufficiently well.

Step 5: The distance relations of the high dimensional prototypes $\hat{x}_j$ can be visualized as a 3D landscape on top of the two dimensional projection P. Let $N_j \subset \{1, ..., r^2\}$ be the indices of all grid point immediately neighboring g_j. The U-height of g_j is defined as

$$U(g_j) = \frac{1}{|N_j|} \sum_{k \in N_j} D'(\hat{x}_j, \hat{x}_k)$$

where D' is a distance function in $\mathbb{R}^n$ of choice. The U-heights for all grid points g_j form the U-Map.

The parameter r determines the resolution of the U-Map, we used $r = 100$ in our experiments. The results are not very sensitive to the choice of parameters for the training. We simply used the default values of the Databionic ESOM Tools [9][2], namely 20 training epochs, linear cooling of learning rate from 0.5 to 0.1, linear cooling of a Gaussian neighborhood kernel from 24 to 1.

4 Demonstrations

Figure 3 shows the U-map as gray scales on top of a PCA projection of the Chainlink data. Bright areas indicate that the distances of prototypes neighboring on the grid are small. Dark areas indicate regions where the projection onto the 2D plane does not represent distances in the original space adequately. This can occur for two reasons. At the regions marked with a circle it can bee seen that PCA projects points from the two separate classes in the same region. In this case the dark colors of the U-map indicate large local projection errors, as the original distances of these points are not well represented. Other dark regions of outside the circles do not contain any projected data points. They are caused by the Self-Organizing map training that compresses empty regions of the high dimensional space to only few grid points. These dark regions represent cluster boundaries, thus valid and desired foldings of the projection.

Figure 5 shows the U-maps for other projections. ICA and Sammon also have regions with projected data points and high folding errors. For ESOM the U-map

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

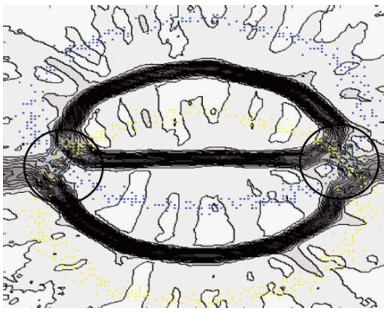
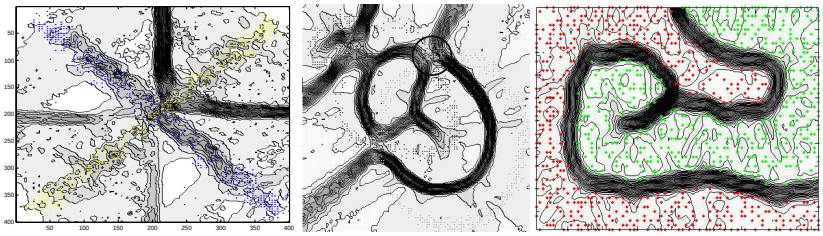


Fig. 3. U-map on top of a PCA projection of the Chainlink data. There are large local projection errors in the circled regions.

is the same as the well known U-matrix [3], all dark regions exclusively indicate cluster boundaries.



(a) ICA (b) Sammon (c) ESOM

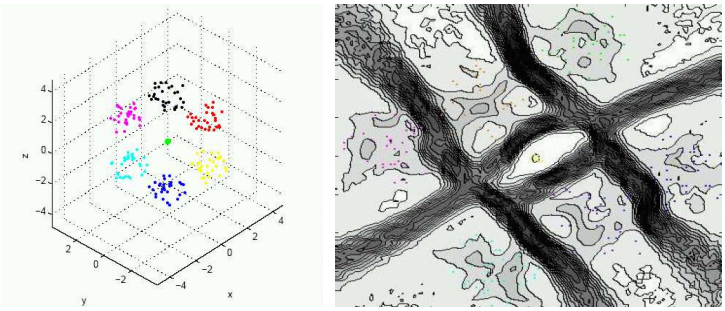
Fig. 4. U-maps of different Chainlink projections

The U-Map not only helps to identify local projection errors, it also supports the detection of cluster boundaries. As an example consider the Hepta data set from the Fundamental Clustering Problem Suite (FCPS) [3] shown in Figure 5(a) consisting of 7 well seperated clusters in 3D. Figure shows the U-map of a PCA projection. The display clearly enhances the visualization of the cluster structure.

5 Application to real world data

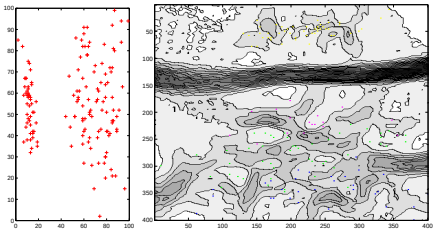
Figure 6(a) shows Fisher's Iris data (see FCPS) projected with an Independent Component Analysis (ICA). From this figure one could conclude that there are two well separated but coherent clusters in the data. The corresponding U-map shown in Figure 6(b) reveals, however, that one of the seemingly coherent clusters is presumably more complex. This cluster may then be further subdivided until finally the different types of Iris flowers are identified.

[3] `http://www.mathematik.uni-marburg.de/~databionics/`



(a) Hepta data (b) U-map on top of PCA projection of Hepta

Fig. 5. The Hepta data set from FCPS



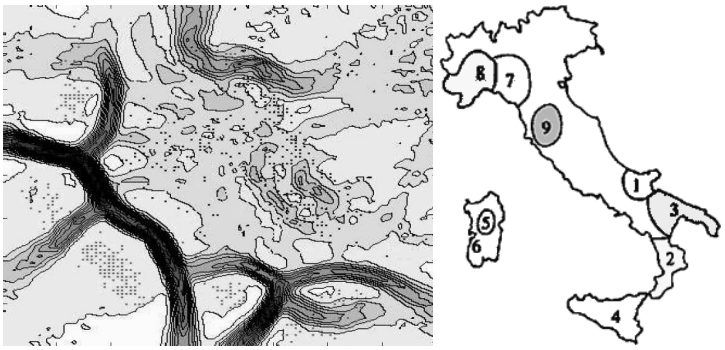
(a) ICA projection (b) U-map on top of ICA

Fig. 6. Iris data

A more realistic data set is taken from [12]. The dataset consists of analytical data from 572 Italian olive oils produced in nine different regions of Italy. For each olive oil the percentual contents of 8 different fatty acids was measured. Figure 7(a) shows the U-map of a multidimensional scaling projection (MDS) projection of this data set. The colors of the points corresponded to the region where the olive oil originated. Two types of oils are projection in the same are. These are the olive oils from Sardinia, where one production region is located within the other (Regions numbers 5 and 6 in Figure 7(b)).

6 Conclusion

In this work a method to visualize the local errors for an arbitrary projection from high dimensional data spaces onto a two dimensions is proposed. The result is a three dimensional landscape, called U-map, in which low regions represent small errors and high mountains indicate severe projection errors. The method is demonstrated on instructive examples. Even if global projection errors are minimized, such as in MDS or Sammon's mapping, there might be locations where the projections are severely misleading. U-maps are a tool to point out such locations in the projected



(a) U-map on top of MDS projection (b) Production regions

Fig. 7. Olive oil data

data. They can further used as a valuable tool tor the identification of structures, such as clusters in a data set.

References

1.Sammon JW (1969) A Nonlinear Mapping for Data Structure Analysis. IEEE Transactions on Computation C-18(5):401–409.
2.Joliffe IT (1986) Principal Component Analysis, Springer, New-York.
3.Ultsch A (1992): Self-organizing neural networks for visualization and classification. In: Opitz O, Lausen B, Klar R (eds) Proceedings of the 16th Annual Conference of the German Classification Society. Springer, Berlin, 307-313.
4.Borg I, Groenen P (1996) Modern Multidimensional Scaling : Theory and Applications, Springer, New-York.
5.Hyvärinen A (1999) Survey on independent component analysis. Neural Computing Surveys 2:94–128.
6.Tenenbaum JB, De Silva V, Langford JC (2000) A global geometric framework for nonlinear dimensionality reduction. Science 290:2319–2323.
7.Ultsch A (2003) Maps for the Visualization of high dimensional Data Spaces. In: Yamakawa T (eds) Proceedings of the 4th Workshop on Self-Organizing Maps, 225-230.
8.Ultsch A, Herrmann L (2005) The architecture of emergent self-organizing maps to reduce projection errors. In: Verleysen M (eds) Proceedings of the European Symposium on Artificial Neural Networks, 1–6.
9.Ultsch A, Mörchen F (2005): ESOM-Maps: tools for clustering, visualization, and classification with Emergent SOM. Technichal Report No. 46, Department of Mathematics and Computer Science, Philipps-University Marburg.
10.Ultsch A (2005): Clustering with SOM: U*C. Proceedings Workshop on Self-Organizing Maps, 75-82.
11.Ultsch A (1996): Self Organizing Neural Networks perform different from statistical k-means clustering. In: Bock HH, Polasek W (eds) Proceedings of the 19th Annual Conference of the German Classification Society. Springer, Berlin.
12.Zupan, J. , Gasteiger, J.(1993): Neural Networks for Chemists VCH, Weinheim New York, 1993.