Weihs, Claus; Luebke, Karsten; Raabe, Nils

Working Paper

KMC/EDAM : A new approach for the visualization of K-Means Clustering results

Technical Report, No. 2004,65

Provided in Cooperation with:
Collaborative Research Center 'Reduction of Complexity in Multivariate Data Structures' (SFB 475), University of Dortmund

Suggested Citation: Weihs, Claus; Luebke, Karsten; Raabe, Nils (2004) : KMC/EDAM : A new approach for the visualization of K-Means Clustering results, Technical Report, No. 2004,65, Universität Dortmund, Sonderforschungsbereich 475 - Komplexitätsreduktion in Multivariaten Datenstrukturen, Dortmund

This Version is available at:
https://hdl.handle.net/10419/22578

KMC/EDAM: A new approach for the visualization of K-Means Clustering results

Nils Raabe, Karsten Luebke, and Claus Weihs

University of Dortmund*
Department of Statistics
44221 Dortmund, Germany

Abstract. In this work we introduce a method for classification and visualization. In contrast to simultaneous methods like e.g. Kohonen SOM this new approach, called KMC/EDAM, runs through two stages. In the first stage the data is clustered by classical methods like K-means clustering. In the second stage the centroids of the obtained clusters are visualized in a fixed target space which is directly comparable to that of SOM.

1 Introduction

In many applications a classification of the examined objects in both inter-heterogeneous and intra-homogeneous groups (clusters) is desired. Many methods have been developed to solve this problem and are subsumed under the term classification-methods as well as clustering-methods.

In the context of clustered objects another problem often occurs. This problem consists of the graphical representation - called visualization - of the objects resp. classes which are often represented by high-dimensional data vectors in a space of lower dimension. The requirement for such representations is topology preservation, i.e. objects which are comparatively close in the original space should also be close together in the representation space and, corresponding by, pairs of distant objects should have high distances in the visualization.

One method, which can be interpreted both as a visualization and a classification method, is the so called Kohonen Self-Organizing-Map (SOM) (Kohonen, 1990). SOM performs classification and visualization simultaneously. Many alternatives to SOM have been proposed in the past. One example is another simultaneous method suggested by Bock (1997). Bezdek and Pal (1995) compare the methods principal component analysis (PCA) and the Sammon algorithm to SOM concerning topology preservation. They try to avoid the problem of different solution spaces - with SOM in contrast to the latter methods only a subset of the objects is visualized - by assigning to each object an image in the neighborhood of the nearest visualized object.

* This work has been supported by the Deutsche Forschungsgemeinschaft, Sonderforschungsbereich 475.

Since this is done by randomly jittering it is questionable if the corresponding results can still be seen as the results generated by SOM. Hence the results of Bezdek and Pal – PCA and Sammon are superior to SOM – are to be interpreted cautiously.

Being aware of the aforementioned comparability-problems we introduce a new approach of carrying out classification and visualization one after the other. This approach consists of a combination of classical classification methods (mainly K-Means-Clustering, KMC) and a new approach for the visualization of the corresponding centroids. This approach is called Eight-Directions-Arranged-Map (EDAM) and has a fixed representation space. This solution space can be chosen in SOM as well. Under these conditions criteria for classification and topology preservation can be defined and compared between the two methods.

This paper starts with a description of the methods in section 2. Then section 3 gives a view on a few examples. The paper concludes in a summary given in section 4.

2 Methods

2.1 Preliminaries

All following methods refer to a data matrix $X \in I\!R^{n \times k}$. Its rows $x_{1\cdot}, ..., x_{n\cdot} \in I\!R^k$ represent the data vectors of n corresponding objects and its columns $x_{\cdot 1}, ..., x_{\cdot k} \in I\!R^n$ represent the measurement vectors of k corresponding variables. Distances between two data vectors $x_{i\cdot}$ and $x_{j\cdot}$ are denoted by $d(x_{i\cdot}, x_{j\cdot})$. We use the ordinary euclidean distance in this paper.

A classification of X is a set of c clusters, where each object belongs to exactly one cluster. A classification is denoted by a vector $\kappa \in \{1, ..., c\}^n$, where the ith element κ_i of κ gives the cluster-number of the ith object. A common representative of cluster i is the so called centroid $\mu_i \in I\!R^k$, which is defined as:

$$\begin{gathered} \mu_i = (\mu_{i1}, ..., \mu_{ik})' \quad \text{with} \quad \mu_{ih} = \frac{1}{n_i} \sum_{j:\kappa_j = i} x_{jh}, \quad h = 1, ..., k, \\ n_i = \#\{j : \kappa_j = i\}, \quad i = 1, ..., c. \end{gathered} \tag{1}$$

All centroids are compiled in the centroid matrix $M = (\mu_{ij})_{\substack{1 \le i \le c \\ 1 \le j \le k}}$.

A visualization of X is a function $f : \{x_{1\cdot}, ...x_{n\cdot}\} \to \mathcal{Z} \subseteq I\!R^{n \times m}, m < k$, which assigns an image $z^i = (z^i_1, .., z^i_m)' = f(x_{i\cdot})$ to each row of X. $\mathcal{Z}$ is called the image-space.

With $Z = (z^i_j)_{\substack{1 \le i \le n \\ 1 \le j \le m}}$ the visualization f may be written as $f(X) = Z$. In the following we only consider the case of $m = 2$.

2.2 Basic Idea

Our approach to visualize high-dimensional data in a plane is based on the idea of considering the plane as a topographical map. When the images are visualized as the vertices of a rectangular grid, each object has eight direct neighbors, one in each direction of the compass (by taking NE, SE, SW and NW into account, compare figure 1). We try to obtain topology preservation by re-ordering the objects on each of these eight directions corresponding to the distances of their data vectors in the original space $\mathbb{R}^k$. Considering the example of the vector pointing from z^{20} to west in figure 1 this means, that with $x_{i\cdot} = f^{-1}(z^i)$ after re-ordering, i.e. interchanging the values of $x_{21\cdot}$ to $x_{24\cdot}$, the relation $d(x_{20\cdot}, x_{21\cdot}) \leq d(x_{20\cdot}, x_{22\cdot}) \leq d(x_{20\cdot}, x_{23\cdot}) \leq d(x_{20\cdot}, x_{24\cdot})$ holds.

The method EDAM visualizes by repeating this "star-shaped" re-ordering step successively for all objects up to either convergence or to another stopping criterion. The following subsection gives a formal definition of the method.

2.3 KMC/EDAM

The classification of X into a set of $c < n$ clusters, c given, by the method KMC/EDAM is performed by a combination of a K-Means-algorithm and a hierarchical method. First $g > c$ clusters are constructed by applying the K-Means-algorithm suggested by Forgy (see Anderberg 1973). Then the agglomerative hierarchical Centroid-method (see Kaufmann and Pape 1996) is applied to these clusters. After $(g - c)$ steps of this method the final classification κ of the n objects into c clusters is obtained.

In the next stage of KMC/EDAM the centroids $\{\mu_1, ..., \mu_c\}$ of κ are visualized. Therefore first the image space is fixed to the points of intersections of b_1 vertical and b_2 horizontal lines of a two-dimensional, equally spaced grid, with $c = b_1 \cdot b_2$. By labelling the images by their integer Euclidean coordinates and enumerating them from the lower left corner by rows the image-space can be written as:

$$\mathcal{Z} = \left\{ z^1, \cdots, z^c \right\} \quad \text{with} \quad z^i = \begin{pmatrix} z_1^i \\ z_2^i \end{pmatrix} = \begin{pmatrix} i - \lfloor \frac{i-1}{b_1} \rfloor \cdot b_1 \\ \lceil \frac{i}{b_1} \rceil \end{pmatrix}. \qquad (2)$$

The problem of visualizing the centroids in $\mathcal{Z}$ by a visualization f is to find a permutation π of $\{1, ..., c\}$, such that $f(\mu_{\pi(i)}) = z^i, i = 1, ..., c$, preserves topology as well as possible (concerning to a predefined criterion).

The main idea of our method is to consider[1] each centroid $\mu_{\pi_{t-1}(i)}$ as a "reference point" for the centroids whose images are lying on the vectors pointing from z^i to each direction $D \in \{N, NE, ..., NW\}$, where π_0 is a

[1] The consideration of one centroid defines one step denoted by index t; the index i defining the actual centroid computes to $i = t - \lfloor \frac{t-1}{c} \rfloor \cdot c$, i.e. each time t exceeds a multiple of c, i is switched back to 1.

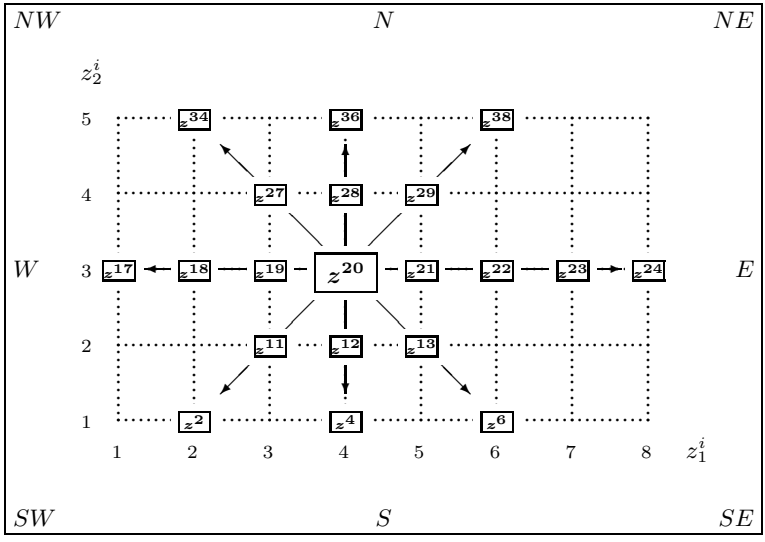

Fig. 1. $\mathcal{Z}$ as topographical map

randomly chosen initial permutation. First, for each direction D, the indices $j_q^D, q = 1, ..., n_D$ of these images are determined. Table 1 gives an overview of how these indices are calculated for all directions.

Table 1. Calculation of indices

D	j_q^D	n_D	D	j_q^D	n_D
N	$i + qb_1$	$b_2 - z_2^i$	NE	$i + qb_1 + q$	$min(n_N, n_E)$
E	$i + q$	$b_1 - z_1^i$	SE	$i - qb_1 + q$	$min(n_S, n_E)$
S	$i - qb_1$	$z_2^i - 1$	SW	$i - qb_1 - q$	$min(n_S, n_W)$
W	$i - q$	$z_1^i - 1$	NW	$i + qb_1 - q$	$min(n_N, n_W)$

Let now φ_D be the permutation of $\{\pi_{t-1}(j_1^D), ..., \pi_{t-1}(j_{n_D}^D)\}$ so that

$$d(\mu_{\pi_{t-1}(i)}, \mu_{\varphi_D[\pi_{t-1}(j_1^D)]}) \leq d(\mu_{\pi_{t-1}(i)}, \mu_{\varphi_D[\pi_{t-1}(j_2^D)]}) \leq ... \leq d(\mu_{\pi_{t-1}(i)}, \mu_{\varphi_D[\pi_{t-1}(j_{n_D}^D)]})$$

for each direction D. Now, set $\pi_t := \pi_{t-1}$. Next, the following substeps are repeated for all directions D:

1. $\pi_t^D := \pi_t$

2. $\{\pi_t^D(j_1^D), ..., \pi_t^D(j_{n_D}^D)\} := \{\varphi_D([\pi_t^D(j_1^D)], ..., \varphi_D([\pi_t^D(j_{n_D}^D)]\}$
3. $\pi_t := \begin{cases} \pi_t^D & , \quad if \quad S(\pi_t^D) < S(\pi_t) \\ \pi_t & , \quad else \end{cases}$.

The function S is a predefined criterion for visualizations with lower values indicating better visualizations. Repeating the described procedure for all centroids – i.e. a set of c steps – builds one iteration. In our investigations we choose S as the STRESS known from MDS (see Hamerle and Pape, 1996, p. 769).

Each time, when no more improvement can be obtained after a complete iteration (or alternatively if a given maximum number of iterations is reached), the area, in which re-ordering is possible is decreased by changing the values of n_D in table 1 to $min(n_D, max[b_1, b_2] - r)$, where r runs successively from 1 to $max[b_1, b_2] - 2$. A set of iterations with the same value of r is called iteration cycle.

The final visualization result $f(\mu_{\pi(i)}) = z^i, i = 1, ..., c$, of KMC/EDAM is obtained by setting $\pi := \pi_t$ where t is the number of the last step.

3 Examples

First the introduced method is applied to the synthetic Chainlink data, which consist of two three-dimensional interlocking ring-shaped classes as seen in figure 2. In our example each class contains 1000 data points.

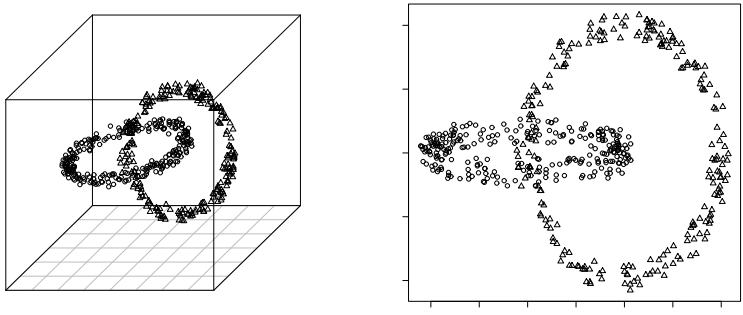

Fig. 2. The Chainlink data and its MDS visualization

On the right side of figure 2 the two-dimensional result of the method MDS for this example is depicted. The STRESS of this result is 0.246. Note that in the original space the two classes have exactly the same relation to each other, i.e. they have the same shape, the rings have the same radius

and the center of each ring lies on the other one. But looking at the MDS visualization one gets the impression that there are differences between the shapes of the classes.

For the computation of KMC/EDAM for the Chainlink data the following settings were used: g=750, c=500,$b_1 = 20$, $b_2 = 25$, maximum number of iterations per cycle: 10. The result is shown in U-Matrix-representation on the left side of figure 3. The U-matrix is a well-known tool developed for the representation of Self-Organizing Maps (compare Ultsch, 2003). Since the image space of KMC/EDAM is restricted to a rectangular grid the U-Matrix can easily be applied to the results of this method as well. For comparison purposes the right side of figure 3 shows the U-matrix of a SOM of the same size applied to the same data. For the computation of the SOM the package `som` available for the statistical software `R` (R Development Core Team, 2004) with its default settings was used. The size of the symbols in both pictures corresponds to the number of objects assigned to each cluster.

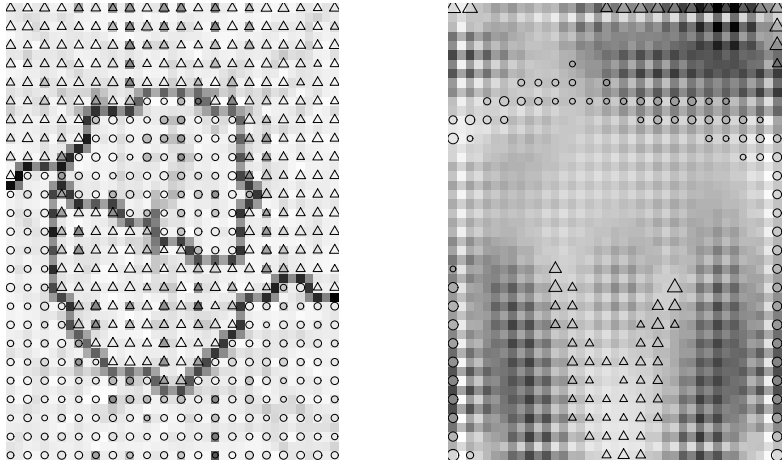

Fig. 3. KMC/EDAM and SOM visualization of the Chainlink data

The STRESS of the KMC/EDAM solution is 0.209, that of SOM is 0.252, so KMC/EDAM seems to be better. Beyond this superiority of KMC/EDAM to the MDS and SOM the result of KMC/EDAM gives an evidently better mirror of the fact, that the Chainlink classes are equally placed relatively to each other. This is not the case for SOM, since the class depicted by circles seems to surround parts of the classes depicted by triangles. At first glance

the separation of the classes seems better for MDS resp. SOM, since the latter leave gaps between the classes. But in the U-Matrix of the KMC/EDAM result a dark line is visible which corresponds to relatively high distances between objects along the line. This line runs between the two classes like a boundary. The brightness of the rest of the map is well-adjusted, which suggests that the topology within the classes is well preserved. Another advantage of the KMC/EDAM result compared to that of SOM is, that it maintains the connection of the classes, i.e. there are now exclaves. In the SOM result there are apparently a few objects of the "triangle class" separated from the rest by the "circle class".

The next example we consider is the well-known iris data set introduced by Fisher (1936), which contain setal and petal lengths and widths of three species of iris for 150 flowers. Figure 4 shows a plot of the MDS result and the U-matrices of KMC/EDAM and SOM results for this example. The settings of KMC/EDAM were: g=50, c=35,$b_1 = 5$, $b_2 = 7$, maximum number of iterations per cycle: 10.

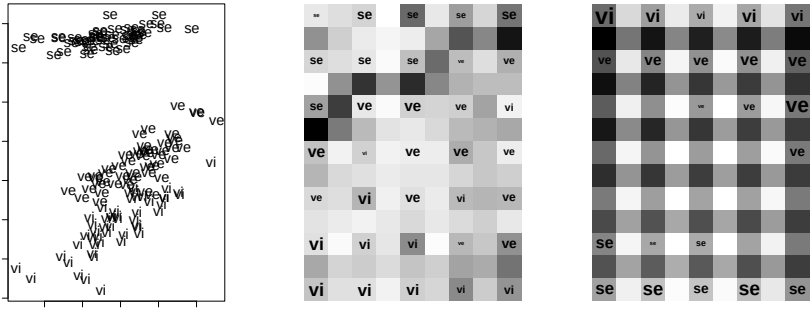

Fig. 4. MDS, KMC/EDAM and SOM visualizations of the iris data

The STRESS of KMC/EDAM is 0.351 in this case, while that of SOM is 0.252. MDS performs even better with a STRESS of 0.04. Similarly to the previous example SOM leaves a gap between the well-separated classes versicolor and setosa. Again this separation is visible as a dark line in the U-matrix of the KMC/EDAM result. The separation of the classes virginica and versicolor seems slightly better in the SOM result, since one can notice the darkest squares between these classes than at other regions of the map.

4 Conclusion

With KMC/EDAM a method is introduced which allows to visualize the results of classical clustering methods. Because of its specially chosen target space the results of this method are directly comparable to those of SOM. The method is applied to two popular examples, the artificial Chainlink data and Fisher's iris data. In the critical Chainlink example KMC/EDAM leads to better results than MDS and SOM.

In the iris example KMC/EDAM has the highest STRESS. But the relative positions of classes are the same with KMC/EDAM. Furthermore the lacking separation – which is probably the reason for the higher STRESS – becomes visible as well by representing the result in an U-matrix.

Modifications for the improvement of EDAM are conceivable. Such modifications may concern the optimization of the initial ordering of the centroids. On the other hand a method like Simulated Annealing could be integrated into the algorithm to avoid local optima. First attempts in this direction led to promising results.

References

ANDERBERG, M.R. (1973): *Cluster Analysis for Applications*. Academic Press Inc., New York.

BEZDEK, J.C., PAL, N.R. (1995): An index of topological preservation for feature extraction. *Pattern Recognition, 28/3, 381–391.*

BOCK, H.H. (1997): Simultaneous visualization and clustering methods as an alternative to Kohonen maps. In: DELLA RICCIA, G., KRUSE, R., LENZ, H.-J. (Eds.): *Learnings, networks and statistics*. CISM Courses and Lectures, 382, Springer, New York, 67–85.

FISHER, R.A. (1936): The use of multiple measurements in taxonomic problems. *Annals of Eugenics, 7/2, 179–188.*

HAMERLE, A., PAPE, H. (1996): Grundlagen der mehrdimensionalen Skalierung. In: FAHRMEIR, L., HAMERLE, A., TUTZ, G. (Eds.): *Multivariate statistische Verfahren*. De Gruyter, Berlin 765–792.

KAUFMANN, H., PAPE, H. (1996): Clusteranalyse. In: FAHRMEIR, L., HAMERLE, A., TUTZ, G. (Eds.): *Multivariate statistische Verfahren*. De Gruyter, Berlin 437–536.

KOHONEN, T. (1990): The Self-Organizing Map. *Proceedings of the IEEE, 78/9, 1464–1480.*

R DEVELOPMENT CORE TEAM (2004): *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna,Austria. `http://www.R-project.org`

ULTSCH, A. (2003): Maps for the visualization of high-dimensional data spaces. *Proc. Workshop on Self organizing Maps, 225-230.*