Efficient Visualization of High-Dimensional Data with Polar Coordinates

Frank Rehm*
German Aerospace Center
Lilienthalplatz 7
38108 Braunschweig
Germany
voice: +49 531-295-2577
fax: +49 531-295-2899
email: frank.rehm@dlr.de

Frank Klawonn
Department of Computer Science
University of Applied Sciences Braunschweig/Wolfenbuettel
Salzdahlumer Str. 46/48
38302 Wolfenbuettel
Germany
voice: +49 5331-939-6111
fax: +49 5331-939-6002
email: f.klawonn@fh-wolfenbuettel.de

Rudolf Kruse
University of Magdeburg
39106 Magdeburg
Germany
voice: +49 391 67 18706
fax: +49 391 67 12018
email: kruse@iws.cs.uni-magdeburg.de

(* **Corresponding author**)

Efficient Visualization of High-Dimensional Data with Polar Coordinates

Frank Rehm, German Aerospace Center, Germany

Frank Klawonn, University of Applied Sciences Braunschweig/Wolfenbuettel, Germany

Rudolf Kruse, University of Magdeburg, Germany

INTRODUCTION

Many applications in science and business such as signal analysis or costumer segmentation deal with large amounts of data which are usually high dimensional in the feature space. As a part of preprocessing and exploratory data analysis, visualization of the data helps to decide which kind of method probably leads to good results. Since the visual assessment of a feature space that has more than three dimensions is not possible, it becomes necessary to find an appropriate visualization scheme for such datasets.

In this work we will present two methods that visualize high-dimensional data on the plane. An algorithm will be presented that allows applying our method on larger data sets.

BACKGROUND

Mulidimensional scaling (MDS) provides low-dimensional visualization of high-dimensional feature vectors (Kruskal & Wish, 1978; Borg & Groenen, 1997). MDS is a method that estimates the coordinates of a set of objects in a feature space of specified (low) dimensionality that come from data trying to preserve the distances between pairs of objects. In the recent years much research has been done (Chalmers, 1996; Faloutsos & Lin, 1995;

Morrison, Ross, & Chalmers, 2003; Williams & Munzner, 2004). Different ways of computing distances and various functions relating the distances to the actual data are commonly used. These distances are usually stored in a distance matrix. The estimation of the coordinates will be carried out under the constraint, that the error between the distance matrix of the dataset and the distance matrix of the corresponding transformed dataset will be minimized. Thus, different error measures to be minimized were proposed, i.e. the absolute error, the relative error or a combination of both. A commonly used error measure is the so-called Sammon's mapping. To determine the transformed dataset by means of minimizing the error a gradient descent method is used.

Many modifications of MDS are published so far, but high computational costs prevent their application to large datasets (Tenenbaum, de Silva, V., & Langford, 2000). Besides the quadratic need of memory, MDS, as described above is solved by an iterative method, expensive with respect to computation time. Furthermore, a completely new solution must be calculated, if a new object is added to the dataset.

MAIN FOCUS

With MDS_{polar} and POLARMAP we present two approaches to find a two-dimensional projection of a p-dimensional dataset X. Both methods try to find a representation in polar coordinates $Y = \{(l_1, \varphi_1), K, (l_n, \varphi_n)\}$, where the length l_k of the original vector x_k is preserved and only the angle φ_k has to be optimized. Thus, our solution is defined to be optimal if all angles between pairs of data objects in the projected dataset Y coincide as good as possible with the angles in the original feature space X. As we will show later, it is possible to transform new data objects without extra costs.

MDS_{polar}

A straight forward definition of an objective function to be minimized for this problem would be

$$E = \sum_{k=2}^{n} \sum_{i=1}^{k-1} \left(\left| \varphi_i - \varphi_k \right| - \psi_{ik} \right)^2 \qquad (1)$$

where φ_k is the angle of y_k, ψ_{ik} is the positive angle between x_i and x_k. The absolute value is chosen in equation (1) because the order of the minuends can have an influence on the sign of the resulting angle. The problem with this notation is that the functional E is not differentiable, exactly in those points we are interested in, namely, where the difference between angles φ_i and φ_k becomes zero.

We propose an efficient method that enables us to compute an approximate solution for a minimum of the objective function (1) and related ones. In a first step we ignore the absolute value in (1) and consider

$$E = \sum_{k=2}^{n} \sum_{i=1}^{k-1} \left(\varphi_i - \varphi_k - \psi_{ik} \right)^2 . \qquad (2)$$

When we simply minimize (2), the results will not be acceptable. Although the angle between y_i and y_k might perfectly match the angle ψ_{ik}, $\varphi_i - \varphi_k$ can either be ψ_{ik} or $-\psi_{ik}$. Since we assume that $0 \leq \psi_{ik}$ holds, we always have $\left(\left| \varphi_i - \varphi_k \right| - \psi_{ik} \right)^2 \leq \left(\varphi_i - \varphi_k - \psi_{ik} \right)^2$. Therefore, finding a minimum of (2) means that this is an upper bound for the minimum of (1). Therefore, when we minimize (2) in order to actually minimize (1), we can take the freedom to choose whether we want the term $\varphi_i - \varphi_k$ or the term $\varphi_k - \varphi_i$ to appear in (2). Since

$$\left(\varphi_i - \varphi_k - \psi_{ik}\right)^2 = \left(-\left(\varphi_i - \varphi_k - \psi_{ik}\right)\right)^2 = \left(\varphi_k - \varphi_i + \psi_{ik}\right)^2$$

instead of exchanging the order of φ_i and φ_k, we can choose the sign of ψ_{ik}, leading to

$$E = \sum_{k=2}^{n}\sum_{i=1}^{k-1}\left(\varphi_i - \varphi_k - a_{ik}\psi_{ik}\right)^2 \qquad (3)$$

with $a_{ik} \in \{-1,1\}$.

In order to solve this optimization problem of equation (3) we take the partial derivatives of E, yielding

$$\frac{\partial E}{\varphi_k} = -2\sum_{i=1}^{k-1}\left(\varphi_i - \varphi_k - a_{ik}\psi_{ik}\right). \qquad (4)$$

Thus, on the one hand, neglecting that we still have to choose a_{ik}, our solution is described by a system of linear equations which means its solution can be calculated directly without the need of any iteration procedure. On the other hand, as described above, we have to handle the problem of determining the sign of the ψ_{ik} in the form of the a_{ik}-values. To fulfil the necessary condition for a minimum we set equation (4) equal to zero and solve for the φ_k-values, which leads to

$$\varphi_k = \frac{\sum_{i=1}^{k-1}\left(\varphi_i - a_{ik}\psi_{ik}\right)}{k-1}. \qquad (5)$$

Since we only want to preserve the angles between data vectors, it is obvious that any solution will be invariant with respect to rotation of the dataset. Due to the representation in polar coordinates it is necessary to apply a preprocessing step in form of a translation that makes all components of data vectors non-negative. Reasons for that and further details are given in (Rehm, Klawonn, & Kruse, 2005).

A Greedy Algorithm for the Approximation of MDS$_{\text{polar}}$

As mentioned above, this solution describes a system of linear equations. Since the desired transformation is rotation invariant φ_1 can be set to any value, i.e. $\varphi_1 = 0$. By means of a greedy algorithm we choose $a_{ik} \in \{-1,1\}$ such that for the resulting φ_k the error E of the objective function (3) is minimal. For φ_2 the exact solution can always be found, since a_{12} is the only parameter to optimize. For the remaining φ_k the greedy algorithm sets a_{ik} in turn either -1 or 1, verifying the validity of the result, setting a_{ik} the better value immediately and continuing with the next a_{ik} until all $k-1$ values for a_{ik} are set.

A Generalized MDS$_{\text{polar}}$

In certain cases the objective when transforming data is to preserve relations of feature vectors of the original feature space in the target feature space. Thus, feature vectors that form a cluster should be represented as exact as possible in the target feature space, too. The transformation of feature vectors with a large distance to the respective feature vector can have a lower accuracy. An approach to achieve this goal is the introduction of weights w_{ik} to our objective function

$$E = \sum_{k=2}^{n} \sum_{i=1}^{k-1} w_{ik} \left(\varphi_i - \varphi_k - a_{ik} \psi_{ik} \right)^2 . \tag{6}$$

The main benefit of weights, indeed, is the ability to decrease the computational complexity of the algorithm. This is the case if weights are chosen in such a way, that for feature vectors with a certain (large) distance the respecting weights become zero. A weighting function can control this behaviour automatically. For an efficient implementation it is useful to sort the feature vectors by means of their length. Note that sorting can be carried out in less than quadratic time.

Weighting functions should be decreasing and should lead to zero weights for proper feature vectors. Different weighting functions and further details can be seen in (Rehm, Klawonn, & Kruse, 2005). In this way, feature vectors can be grouped into suitable bins, reducing the complexity of our algorithm to $O(n \cdot \log n)$.

POLARMAP

As an extension of MDS$_\text{polar}$ we propose in this work a method that learns a function f that provides for any p-dimensional feature vector x_k the corresponding angle φ_k that is needed to map the feature vector to a 2-dimensional feature space. As for MDS$_\text{polar}$ the length of vector x_k is preserved. With the obtained function also angles for new feature vectors can be computed. A 2-dimensional scatter plot might not be suitable, when visualising mappings for large datasets. With the computed function it is simple to produce information murals, which allow more comprehensive visualizations (Jerding & Stasko, 1995).

Analogous to functional (1) we define our objective function E as follows:

$$E = \sum_{k=2}^{n} \sum_{i=1}^{k-1} \left(\left| f(x_i) - f(x_k) \right| - \psi_{ik} \right)^2 . \tag{6}$$

Since functional (6) is not differentiable, we propose analogous to the procedure for MDS$_\text{polar}$ to minimise the following differentiable objective function

$$E = \sum_{k=2}^{n} \sum_{i=1}^{k-1} \left(f(x_i) - f(x_k) - \psi_{ik} \right)^2 . \tag{7}$$

Albeit, f might be any function, we discuss in this work the following type of function

$$f(x) = a^T \cdot \tilde{x} \tag{8}$$

where a is vector whose components are the parameters to be optimised and $\tilde{x}$ is the feature

vector x itself or a modification of x. In the simplest case we use

$$
\begin{aligned}
\tilde{x} &= x \\
a &= (a_1, a_2, \mathrm{K}, a_p)
\end{aligned}
\tag{9}
$$

where f describes in fact the linear combination of x. Other functions f are discussed in

(Rehm, Klawonn, & Kruse, 2006).

Replacing term f by the respective function we obtain

$$
E = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(a^T \tilde{x}_i - a^T \tilde{x}_j - \psi_{ik} \right)^2 = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(a^T \left(\tilde{x}_i - \tilde{x}_j \right) - \psi_{ik} \right)^2 .
\tag{10}
$$

For a better readability we replace $\tilde{x}_i - \tilde{x}_j$ by $\tilde{x}_{ij}$ and obtain

$$
E = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(a^T \tilde{x}_{ij} - \psi_{ij} \right)^2 .
\tag{11}
$$

The derivative of E w.r.t. a can by easily obtained

$$
E = 2 \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(a^T \tilde{x}_{ij} - \psi_{ij} \right) \tilde{x}_{ij}
\tag{12}
$$

which results in a system of linear equations in $a = \left(a_1, a_2, \mathrm{K}, a_p \right)^T$. As mentioned already,

angles computed by $f(x_i) - f(x_j)$, might be positive or negative, while ψ_{ik} is always positive

by definition. Thus, in the case where $a^T \tilde{x}_{ij} < 0$ holds, E might be minimal, but our original

objective function E might not be minimal. Hence, replacing $\tilde{x}_{ij}$ by $-\tilde{x}_{ij}$ in this case might

lower the error. Consequently, finding the appropriate sign for $\tilde{x}_{ij}$ is a crucial step when

minimizing E.

Determining the sign for each $\tilde{x}_{ij}$ requires exponential need of computation time in the

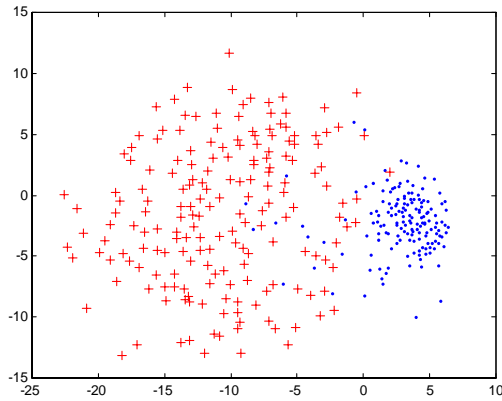number of feature vectors. For real-world datasets this is unacceptable. When relaxing the

Figure 1: Sammon's Mapping of the Wisconsin Breast Cancer Dataset
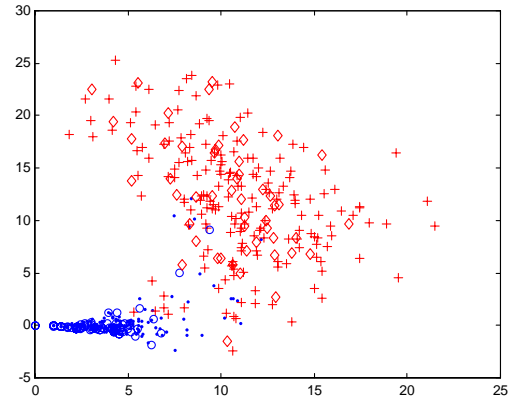

Figure 2: POLARMAP on the Wisconsin Breast Cancer Dataset

problem in favour to an approximation of the exact solution one can reduce the time complexity down to $O(n \cdot \log n)$. As for $\text{MDS}_{\text{polar}}$ this can be achieved by means of introducing weights which is discussed in detail in (Rehm, Klawonn, & Kruse, 2006).

Experimental Results

Since a function is learned by POLARMAP it becomes possible to map new vectors in the target space. To demonstrate the power of POLARMAP, we applied it on the well known Wisconsin breast cancer dataset[1] (Mangasarian & Wolberg, 1990). Each patient in the database had a fine needle aspirate taken from her breast. Resultant, nine attributes where determined and analysed to discriminate benign from malignant breast lumps. Figure 1 shows the Sammon's mapping of the dataset. The transformation of the Wisconsin breast cancer dataset with POLARMAP is shown in figure 2. The different classes are represented by different symbols. Both transformations are similar regarding the scattering of the different classes. Patients with benign lumps and those with malignant lumps can be almost separated linearly in both

---

[1] The breast cancer database was obtained from the University of Wisconsin Hospitals, Madison from Dr. William H. Wolberg.

transformations. Only few points can be found in regions where the opposite class mainly represented.

For the transformation with POLARMAP, the dataset is split into a training dataset and a test dataset. The training dataset consists of 80% of each class. This part of the data is used to learn the desired coefficients. The test dataset, that contains the remaining 20% of the data, is mapped to the target space by means of the learned function. The mapping of the training dataset is plotted with the different symbols again, each for the corresponding class. The mapped feature vectors of the test dataset are marked with a small circle or a diamond, respectively. As the figure shows, the learned function maps the new feature vectors in an appropriate way.

FUTURE TRENDS

CONCLUSION

In this paper we have described a powerful data visualisation method. Under the constraint to preserve the length of feature vectors, it was our aim to find a mapping that projects feature vectors from a high-dimensional space to the plane in such a way that we minimise the errors in the angles between the mapped feature vectors. The solution is described by a system of linear equations. To overcome the problem in high-dimensional feature spaces, that no differentiation between positive and negative angles can be made as for a 2-dimensional feature space, an algorithm is provided to obtain the desired signs for the angles. With the bin-algorithm, we presented an algorithm, that lowers the computation complexity down to $O(n \cdot \log n)$.

REFERENCES

Borg, I., Groenen, P. (1997). Modern Multidimensional Scaling : Theory and Applications. Springer, Berlin.

Chalmers, M. (1996). A Linear Iteration Time Layout Algorithm for Visualising High-Dimensional Data. Proceedings of IEEE Visualization, San Francisco, CA, 127-132.

Faloutsos, C., & Lin, K. (1995). Fastmap: A Fast Algorithm for Indexing, Data-Mining and Visualization of Traditional and Multimedia Datasets. Proceedings of ACM SIGMOD International Conference on Management of Data, San Jose, CA, 163-174.

Kruskal, J. B., & Wish, M. (1978). Multidimensional Scaling. SAGE Publications, Beverly Hills.

Jerding, D. F., & Stasko, J. T. (1995). The information mural: a technique for displaying and navigating large information spaces. Proceedings of the 1995 IEEE Symposium on Information Visualization, 43–50.

Mangasarian, O.L., & Wolberg, W.H. (1990). Cancer diagnosis via linear programming. SIAM News, 23(5), 17-18.

Morrison, A., Ross, G., & Chalmers, M. (2003). Fast Multidimensional Scaling through Sampling, Springs and Interpolation. Information Visualization, 2, 68-77.

Rehm, F., Klawonn, F., & Kruse, R. (2005). MDS_{polar} - A New Approach for Dimension Reduction to Visualize High Dimensional Data. In: Advances in Intelligent Data Analysis VI: 6th International Symposium on Intelligent Data Analysis, IDA 2005, Springer, 316-327.

Rehm, F., Klawonn, F., & Kruse, R. (2006). POLARMAP - Efficient Visualization of High Dimensional Data. (Submitted for publication).

Tenenbaum, J. B., de Silva, V., & Langford, J. C. (2000). A Global Geometric Framework for Nonlinear Dimensionality Reduction. Science, 290, 2319–2323.

Williams, M., & Munzner, T. (2004). Steerable, Progressive Multidimensional Scaling. 10th IEEE Symposium on Information Visualization, Austin, TX, 57-64.

KEY TERMS AND THEIR DEFINITIONS

Data Visualization: Presentation of data in human understandable graphics, images, or animation.

Visual Data Mining: Data mining process through data visualization. The fundamental concept of visual data mining is the interaction between data visual presentation, human graphics cognition, and problem solving.

Multidimensional scaling: Multidimensional scaling provides low-dimensional visualization of high-dimensional feature vectors.