Interactive Data Mining with 3D-Parallel-Coordinate-Trees

Elke Achtert; Hans-Peter Kriegel; Erich Schubert; Arthur Zimek
Institut für Informatik
Ludwig-Maximilians-Universität München
Oettingenstr. 67, 80538 München, Germany
{achtert,schube,kriegel,zimek}@dbs.ifi.lmu.de

ABSTRACT

Parallel coordinates are an established technique to visualize high-dimensional data, in particular for data mining purposes. A major challenge is the ordering of axes, as any axis can have at most two neighbors when placed in parallel on a 2D plane. By extending this concept to a 3D visualization space we can place several axes next to each other. However, finding a good arrangement often does not necessarily become easier, as still not all axes can be arranged pairwise adjacently to each other. Here, we provide a tool to explore complex data sets using 3D-parallel-coordinate-trees, along with a number of approaches to arrange the axes.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—*Data Visualization Methods*

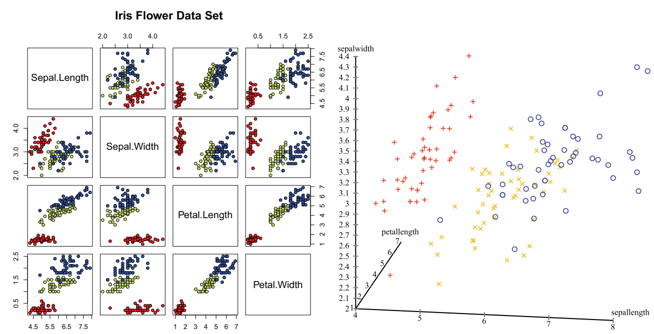
Keywords

Parallel Coordinates; Visualization; High-Dimensional Data

1. INTRODUCTION

Automated data mining methods for mining high-dimensional data, such as subspace and projected clustering [5, 6, 11] or outlier detection [7, 22, 26], found much attention in database research. Yet all methods in these fields are still immature and all have deficiencies and shortcomings (see the discussion in surveys on subspace clustering [24, 25, 27] or outlier detection [32]). Visual, interactive analysis and supporting tools for the human eye are therefore an interesting alternative but are susceptible to the "curse of dimensionality" themselves.

Even without considering interactive features, visualizing high-dimensional data is a non-trivial challenge. Traditional scatter plots work fine for 2D and 3D projections, but for high-dimensional data, one has to resort to selecting a subset of features. Technically, a 3D scatter plot also is a 2D visualization. In order to get a proper 3D impression, animation or stereo imaging is needed. In Figure 1(a), each pair of dimensions is visualized with a scatter plot. Figure 1(b) visualizes 3 dimensions using a scatter plot.



(a) Pairwise scatterplots (b) 3D scatterplot

Figure 1: Visualization examples for Iris data set

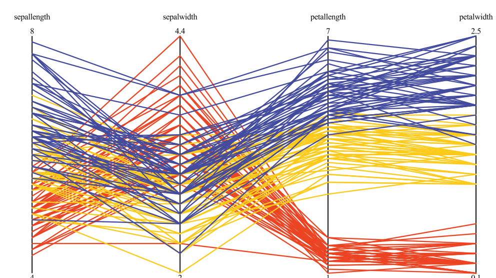


Figure 2: Parallel coordinates plot for Iris data set

Parallel coordinates were popularized for data mining by Alfred Inselberg [18, 19]. By representing each instance as a line path, we can actually visualize more than 2 dimensions on a 2 dimensional plane. For this, axes are placed in parallel (or alternatively, in a star pattern), and each object is represented by a line connecting the coordinates on each axis. Figure 2 is the same data set as above, with the four dimensions parallel to each other. Each colored line is one observation of the data set. Some patterns become very well visible in this projection. For example one of the classes is clearly separable in attributes 3 and 4, and there seems to be an inverse relationship between axes 1-2 as well as 2-3: one of the three Iris species has shorter, but at the same time wider sepal leaves. Of course in this particular, low-dimensional data set, these observation can also be made on the 2D scatter plots in Figure 1(a).

SIGMOD'13, June 22–27, 2013, New York, New York, USA.

2. RELATED WORK

The use of parallel coordinates for visualization has been extensively studied [18, 19]. The challenging question here is how to arrange the coordinates, as patterns are visible only between direct neighbors. Inselberg [18] discusses that $\mathcal{O}(^{N}/_{2})$ permutations suffice to visualize all pairwise relationships, but does not discuss approaches to choose good permutations automatically. The complexity of the arrangement problem has been studied by Ankerst et al. [8]. They discuss linear arrangements and matrix arrangements, but not tree-based layouts. While they show that the linear arrangement problem is NP-hard – the traveling salesman problem – this does not hold for hierarchical layouts. Guo [15] introduces a heuristic based on minimum spanning trees, that actually is more closely related to single-linkage clustering, to find a linear arrangement. Yang et al. [31] discuss integrated dimension reduction for parallel coordinates, which builds a bottom-up hierarchical clustering of dimensions, using a simple counting and threshold-based similarity measure. The main focus is on the interactions of hiding and expanding dimensions. Wegenkittl et al. [30] discuss parallel coordinates in 3D, however their use case is time series data and trajectories, where the axes have a natural order or even a known spatial position. As such, their parallel coordinates remain linear ordered. A 3D visualization based on parallel coordinates [12] uses the third dimension for separating the lines by revolution around the x axis to obtain so called star glyphs. A true 3D version of parallel coordinates [20] does not solve or even discuss the issue of how to obtain a good layout: one axis is placed in the center, the other axes are arranged in a circle around it and connected to the center. Tatu et al. [29] discuss interestingness measures to support visual exploration of large sets of subspaces.

3. ARRANGING DIMENSIONS

3.1 Similarity and Order of Axes

An important ingredient for a meaningful and intuitive arrangement of data axes is to learn about their relationship, similarity, and correlation. In this software, we provide different measures and building blocks to derive a meaningful order of the axes. A straightforward basic approach is to compute the covariance between axes and to derive the correlation coefficient. Since strong positive correlation and strong negative correlation are equally important and interesting for the visualization (and any data analysis on top of that), only the absolute value of the correlation coefficient is used to rank axis pairs. A second approach considers the amount of data objects that share a common slope between two axes. This is another way of assessing a positive correlation between the two axes but for a subset of points. The larger this subset is, the higher is the pair of axes ranked. Additionally to these two baseline approaches, we adapted measures from the literature: As an entropy based approach, we employ MCE [15]. It uses a nested means discretization in each dimension, then evaluates the mutual information of the two dimensions based on this grid. As fourth alternative, we use SURFING [9], an approach for selecting subspaces for clustering based on the distribution of k nearest neighbor distances in the subspace. In subspaces with a very uniform distribution of the kNN distances, the points themselves are expected to be uniformly distributed. Subspaces in which the kNN distances differ strongly from the mean are expected to be more useful and informative. HiCS [21] is a Monte Carlo approach that samples a slice of the data set in one dimension, and compares the distribution of this slice to the distribution of the full dataset in the other slices. This method was actually proposed for subspace outlier detection, but we found it valuable for arranging subspaces, too. Finally, a recent approach specifically designed to support visual exploration of high-dimensional data [28] is ordering dimensions according to their concentration after performing the Hough transformation [17] on the 2D parallel coordinates plot.

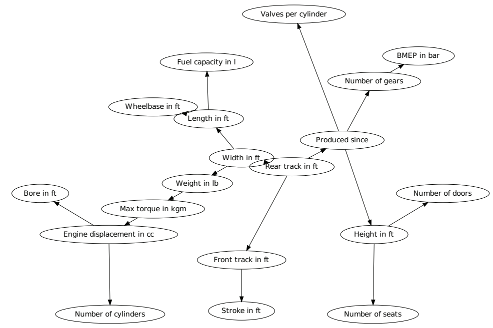


Figure 3: Axis layout for cars data set

3.2 Tree-Visualization

Based on these approaches for assessing the similarity of axes, we compute a pairwise similarity matrix of all dimensions. Then Prim's algorithm is used to compute a minimum spanning tree for this graph, and one of the most central nodes is chosen as root of the visualization tree. This is a new visualization concept which we call 3D-parallel-coordinate-tree (3DPC-tree). Note that both building the distance matrix and Prim's algorithm run in $\mathcal{O}(n^2)$ complexity, and yet the ordering can be considered optimal. So in contrast to the 2D arrangement, which by Ankerst et al. [8] was shown to be NP-hard, this problem actually is easier in 3 dimensions due to the extra degree of freedom. This approach is inspired by Guo [15], except that we directly use the minimum spanning tree, instead of extracting a linear arrangement from it. For the layout of the axis positions, the root of the 3DPC-tree is placed in the center, then the subtrees are layouted recursively, where each subtree gets an angular share relative to their count of leaf nodes, and a distance relative to their depth. The count of leaf nodes is more relevant than the total number of nodes: a chain of one node at each level obviously only needs a width of 1.

Figure 3 visualizes the layout result on the 2D base plane for an example data set containing various car properties such as torque, chassis size and engine properties. Some interesting relationships can already be derived from this plot alone, such that the fuel capacity of a car is primarily connected to the length of the car (longer cars in particular do have more space for a tank), or the number of doors being related to the height of the car (sports cars tend to have fewer doors and are shallow, while when you fit more people in a car, they need to sit more upright).

3.3 Outlier- or Cluster-based Color Coding

An optional additional function for the visualization is to use color coding of the objects according to a clustering

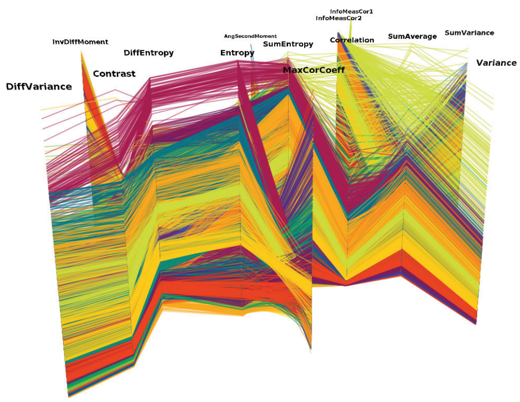


Figure 4: 3DPC-tree plot of Haralick features for 10692 images from ALOI, ordered by the HiCS measure.

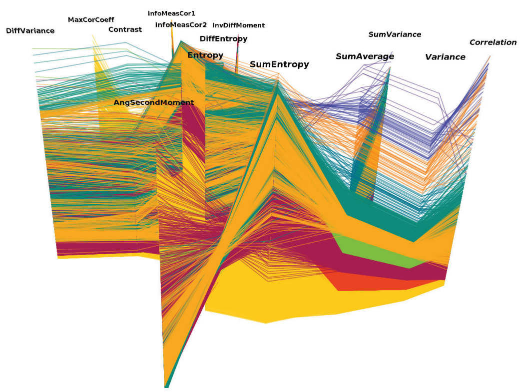


Figure 5: Degenerate k-means result on Haralick vectors

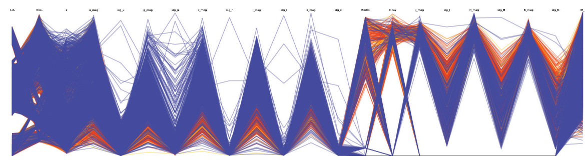
or outlier detection result. As our 3DPC-tree interactive visualization is implemented using the ELKI framework [3, 4], a wide variety of such algorithms comes with it, such as specialized algorithms for high-dimensional data (e.g., SOD [22], COP [23], or subspace clustering algorithms [1, 2, 5, 6, 10, 11]) but also many standard, not specialized, algorithms.

Using color-codes of some algorithm result in the visualization is usefull for example to facilitate a convenient analysis of the behavior of the algorithm.

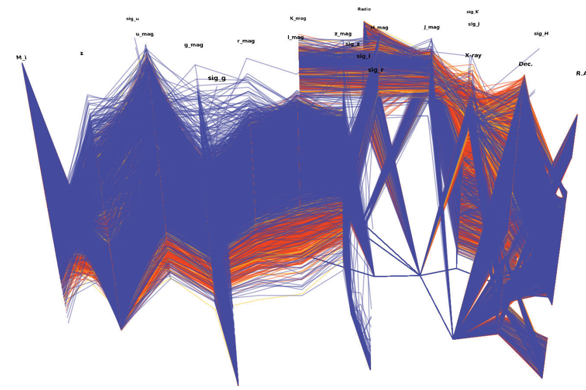
4. DEMONSTRATION SCENARIO

In this demonstration, we present software to interactively explore and mine large, high-dimensional data sets. The view can be customized by selecting different arrangement measures as dicussed above, and can be rotated and zoomed using the mouse. By using OpenGL accelerated graphics, we obtain a reasonable visualization speed even for large data sets (for even larger data sets, sampling may be necessary, but will also be sensible to get a usable visualization).

As an example dataset analysis, Figure 4 visualizes Haralick [16] texture features for 10692 images from the ALOI image collection [14]. The color coding in this image corresponds to the object labels. Clearly there is some redundancy in these features, that can be intuitively seen in this visualization. Dimensions in this image were aligned using the HiCS [21] measure. For a full 3D impression, rotation of course is required.



(a) Default linear arrangement



(b) 3DPC-tree plot

Figure 6: Sloan SDSS quasar dataset.

Visualization is an important control technique. For example, naively running k-means [13] on this data set will yield a result that at first might seem to have worked. However, when visualized as in Figure 5, it becomes visible that the result is strict in both the attributes "Variance" and "SumAverage" – and in fact a one dimensional partitioning of the data set. This of course is caused by the different scales of the axes. Yet, k-means itself does not offer such a control fuctionality.

Figure 6 visualizes the Sloan Digital Sky Survey quasar data set[1]. The first plot visualizes the classic parallel coordinates view, the second plot the 3DPC-tree using covariance similarity. Colors are obtained by running COP outlier detection [23] with expected outlier rate 0.0001, and the colorization thresholds 90% (red) and 99% (yellow) outlier probability. The 3DPC-tree visualization both shows the important correlations in the data set centered around the near-infrared J-band and X-ray attributes, and the complex overall structure of the data set. The peaks visible in the traditional parallel plot come from many attributes in pairs of magnitude and error. In the 3DPC-tree plot, the error attributes are on the margin and often connected only to the corresponding band attribute. With a similarity threshold, they could be pruned from the visualization altogether.

While the demonstration will focus on the visualization technique, we hope to inspire both new development with respect to measuring the similarity of dimensions, layouting methods of axes in the visualization space, and novel ideas for feature reduction and visual data mining in general. By integrating the visualization into the leading toolkit for subspace outlier detection and clustering, the results of various algorithms can visually be explored. Furthermore, we want to encourage the integration of unsupervised and manual (in particular visual) data mining approaches.

[1] `http://astrostatistics.psu.edu/datasets/SDSS_quasar.html`

5. CONCLUSIONS

We provide an open source software for interactive data mining in high-dimensional data, supporting the researcher with optimized visualization tools. This software is based on ELKI [3, 4] and, thus, all outlier detection or clustering algorithms available in ELKI can be used in preprocessing to visualize the data with different colors for different clusters or outlier degrees. This software is available with the release 0.6 of ELKI at `http://elki.dbs.ifi.lmu.de/`.

6. REFERENCES

[1] E. Achtert, C. Böhm, J. David, P. Kröger, and A. Zimek. Global correlation clustering based on the Hough transform. *Stat. Anal. Data Min.*, 1(3):111–127, 2008.

[2] E. Achtert, C. Böhm, H.-P. Kriegel, P. Kröger, I. Müller-Gorman, and A. Zimek. Finding hierarchies of subspace clusters. In *Proc. PKDD*, pages 446–453, 2006.

[3] E. Achtert, S. Goldhofer, H.-P. Kriegel, E. Schubert, and A. Zimek. Evaluation of clusterings – metrics and visual support. In *Proc. ICDE*, pages 1285–1288, 2012.

[4] E. Achtert, A. Hettab, H.-P. Kriegel, E. Schubert, and A. Zimek. Spatial outlier detection: Data, algorithms, visualizations. In *Proc. SSTD*, pages 512–516, 2011.

[5] C. C. Aggarwal, C. M. Procopiuc, J. L. Wolf, P. S. Yu, and J. S. Park. Fast algorithms for projected clustering. In *Proc. SIGMOD*, pages 61–72, 1999.

[6] C. C. Aggarwal and P. S. Yu. Finding generalized projected clusters in high dimensional space. In *Proc. SIGMOD*, pages 70–81, 2000.

[7] C. C. Aggarwal and P. S. Yu. Outlier detection for high dimensional data. In *Proc. SIGMOD*, pages 37–46, 2001.

[8] M. Ankerst, S. Berchtold, and D. A. Keim. Similarity clustering of dimensions for an enhanced visualization of multidimensional data. In *Proc. INFOVIS*, pages 52–60, 1998.

[9] C. Baumgartner, K. Kailing, H.-P. Kriegel, P. Kröger, and C. Plant. Subspace selection for clustering high-dimensional data. In *Proc. ICDM*, pages 11–18, 2004.

[10] C. Böhm, K. Kailing, H.-P. Kriegel, and P. Kröger. Density connected clustering with local subspace preferences. In *Proc. ICDM*, pages 27–34, 2004.

[11] C. Böhm, K. Kailing, P. Kröger, and A. Zimek. Computing clusters of correlation connected objects. In *Proc. SIGMOD*, pages 455–466, 2004.

[12] E. Fanea, S. Carpendale, and T. Isenberg. An interactive 3d integration of parallel coordinates and star glyphs. In *Proc. INFOVIS*, pages 149–156. IEEE, 2005.

[13] E. W. Forgy. Cluster analysis of multivariate data: efficiency versus interpretability of classifications. *Biometrics*, 21:768–769, 1965.

[14] J. M. Geusebroek, G. J. Burghouts, and A. Smeulders. The Amsterdam Library of Object Images. *Int. J. Computer Vision*, 61(1):103–112, 2005.

[15] D. Guo. Coordinating computational and visual approaches for interactive feature selection and multivariate clustering. *Information Visualization*, 2(4):232–246, 2003.

[16] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE TSAP*, 3(6):610–623, 1973.

[17] P. V. C. Hough. Methods and means for recognizing complex patterns. U.S. Patent 3069654, December 18 1962.

[18] A. Inselberg. *Parallel coordinates: visual multidimensional geometry and its applications.* Springer, 2009.

[19] A. Inselberg and B. Dimsdale. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In *Proc. VIS*, pages 361–378, 1990.

[20] J. Johansson, P. Ljung, M. Jern, and M. Cooper. Revealing structure in visualizations of dense 2d and 3d parallel coordinates. *Information Visualization*, 5(2):125–136, 2006.

[21] F. Keller, E. Müller, and K. Böhm. HiCS: high contrast subspaces for density-based outlier ranking. In *Proc. ICDE*, 2012.

[22] H.-P. Kriegel, P. Kröger, E. Schubert, and A. Zimek. Outlier detection in axis-parallel subspaces of high dimensional data. In *Proc. PAKDD*, pages 831–838, 2009.

[23] H.-P. Kriegel, P. Kröger, E. Schubert, and A. Zimek. Outlier detection in arbitrarily oriented subspaces. In *Proc. ICDM*, pages 379–388, 2012.

[24] H.-P. Kriegel, P. Kröger, and A. Zimek. Clustering high dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. *ACM TKDD*, 3(1):1–58, 2009.

[25] H.-P. Kriegel, P. Kröger, and A. Zimek. Subspace clustering. *WIREs DMKD*, 2(4):351–364, 2012.

[26] S. Ramaswamy, R. Rastogi, and K. Shim. Efficient algorithms for mining outliers from large data sets. In *Proc. SIGMOD*, pages 427–438, 2000.

[27] K. Sim, V. Gopalkrishnan, A. Zimek, and G. Cong. A survey on enhanced subspace clustering. *Data Min. Knowl. Disc.*, 26(2):332–397, 2013.

[28] A. Tatu, G. Albuquerque, M. Eisemann, P. Bak, H. Theisel, M. Magnor, and D. Keim. Automated analytical methods to support visual exploration of high-dimensional data. *IEEE TVCG*, 17(5):584–597, 2011.

[29] A. Tatu, F. Maaß, I. Färber, E. Bertini, T. Schreck, T. Seidl, and D. A. Keim. Subspace search and visualization to make sense of alternative clusterings in high-dimensional data. In *Proc. VAST*, pages 63–72, 2012.

[30] R. Wegenkittl, H. Löffelmann, and E. Gröller. Visualizing the behaviour of higher dimensional dynamical systems. In *Proc. VIS*, pages 119–125. IEEE, 1997.

[31] J. Yang, M. Ward, E. Rundensteiner, and S. Huang. Visual hierarchical dimension reduction for exploration of high dimensional datasets. In *Proc. Symp. Data Visualisation 2003*, pages 19–28, 2003.

[32] A. Zimek, E. Schubert, and H.-P. Kriegel. A survey on unsupervised outlier detection in high-dimensional numerical data. *Stat. Anal. Data Min.*, 5(5):363–387, 2012.