Combining automated analysis and visualization techniques for effective exploration of high-dimensional data

Andrada Tatu[*]
University of Konstanz
Germany

Georgia Albuquerque[†]
TU Braunschweig
Germany

Martin Eisemann[‡]
TU Braunschweig
Germany

Jörn Schneidewind[§]
Telefonica o2 Business
Intelligence Center
Germany

Holger Theisel[¶]
University of Magdeburg
Germany

Marcus Magnor[‖]
TU Braunschweig
Germany

Daniel Keim[**]
University of Konstanz
Germany

Abstract

Visual exploration of multivariate data typically requires projection onto lower-dimensional representations. The number of possible representations grows rapidly with the number of dimensions, and manual exploration quickly becomes ineffective or even unfeasible. This paper proposes automatic analysis methods to extract potentially relevant visual structures from a set of candidate visualizations. Based on features, the visualizations are ranked in accordance with a specified user task. The user is provided with a manageable number of potentially useful candidate visualizations, which can be used as a starting point for interactive data analysis. This can effectively ease the task of finding truly useful visualizations and potentially speed up the data exploration task. In this paper, we present ranking measures for class-based as well as non class-based Scatterplots and Parallel Coordinates visualizations. The proposed analysis methods are evaluated on different datasets.

Index Terms: H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval I.3.3 [Computer Graphics]: Picture/Image Generation;

1 Introduction

Due to the technological progress over the last decades, today's scientific and commercial applications are capable of generating, storing, and processing massive amounts of data. Making use of these archives of data provides new challenges to analysis techniques. It is more difficult to filter and extract relevant information from the masses of data since the complexity and volume has increased. Effective visual exploration techniques are needed that incorporate automated analysis components to reduce complexity and to effectively guide the user during the interactive exploration process. The visualization of large complex information spaces typically involves mapping high-dimensional data to lower-dimensional visual representations. The challenge for the analyst is to find an insightful mapping, while the dimensionality of the data, and consequently the number of possible mappings increases. For an effective visual exploration of large data sources, it is therefore essential to support the analyst with Visual Analytics tools that helps the user in finding relevant mappings through automated analysis. One important goal of Visual Analytics, which is the focus of this paper, is to generate representations that best show phenomena contained in the high-dimensional data like clusters and global or local correlations.

Numerous expressive and effective low-dimensional visualizations for high-dimensional datasets have been proposed in the past, such as Scatterplots and Scatterplot matrices, Parallel Coordinates, Hyper-slices, dense pixel displays and geometrically transformed displays [12]. However, finding information-bearing and user-interpretable visual representations automatically remains a difficult task since there could be a large number of possible representations and it could be difficult to determine their relevance to the user. Instead, classical data exploration requires from the user to find interesting phenomena in the data interactively, starting from an initial visual representation. In large scale multivariate datasets, sole interactive exploration becomes ineffective or even unfeasible, since the number of possible representations grows rapidly with the number of dimensions. Methods are needed that help the user automatically find effective and expressive visualizations.

In this paper we present an automated approach that supports the user in the exploration process. The basic idea is to either generate or use a given set of potentially insightful candidate visualizations from the data and to identify potentially relevant visual structures from this set of candidate visualizations. These structures are used to determine the relevance of each visualization to common predefined analysis tasks. The user may then use the visualization with the highest relevance as the starting point of the interactive analysis. We present relevance measures for typical analysis tasks based on Scatterplots and Parallel Coordinates. The experiments based on class-based and non class-based datasets show that our relevance measures effectively assist the user in finding insightful visualizations and potentially speed up the exploration process.

2 Related Work

In the last years several approaches for selecting good views of high-dimensional projections and embeddings have been proposed. One of the first was the *Projection Pursuit* [6, 10]. Its main idea is to search for low-dimensional (one or two-dimensional) projections that expose interesting structures of the high-dimensional dataset, rejecting any irrelevant (noisy or information-poor) dimensions. To exhaustively analyze such a dataset using low-dimensional projections, Asimov presented the *Grand Tour* [3] that supplies the user with a complete overview of the data by generating sequences of orthogonal two-dimensional projections. The problem with this approach is that an extensive exploration of a high-dimensional dataset is effortful and time consuming. A combination of both approaches, Projection Pursuit and the Grand Tour, is proposed in [4] as a visual exploration system. Later on, different Projection Pursuit indices have been proposed [5, 10], but all these techniques do not consider possible class information of the data.

As an alternative to Projection Pursuit, the *Scagnostics* method [21] was proposed to analyze high-dimensional datasets. Wilkinson

[*] e-mail: tatu@inf.uni-konstanz.de
[†] e-mail:georgia@cg.cs.tu-bs.de
[‡] e-mail:eisemann@cg.cs.tu-bs.de
[§] e-mail:jschneidewind@googlemail.com
[¶] e-mail:theisel@isg.cs.uni-magdeburg.de
[‖] e-mail:magnor@cg.cs.tu-bs.de
[**] e-mail:keim@inf.uni-konstanz.de

presented more detailed graph-theoretic measures [23] for computing the Scagnostics indices to detect anomalies in density, shape and trend. These indices could be also used as a ranking for Scatterplot visualizations depending on the analysis task.

We present an image-based measure for non-classified Scatterplots in order to quantify the structures and correlations between the respective dimensions. Our measure can be used as an index in a Scagnostics matrix as an extension to evaluate such correlations.

Koren and Carmel propose a method of creating interesting projections from high-dimensional datasets using linear transformations [13]. Their method integrates the class decomposition of the data, resulting in projections with a clearer separation between the classes. Another interesting visualization method for multivariate datasets is *Parallel Coordinates*. Parallel Coordinates was first introduced by Inselberg [11] and is used in several tools, e.g. XmdvTool [22] and VIS-STAMP [7], for visualizing multivariate data. It is important for Parallel Coordinates to decide the order of the dimensions that are to be presented to the user. Aiming at dimension reordering, Ankerst et al. [1] presented a method based on similarity clustering of dimensions, placing similar dimensions close to each other. Yang [24] developed a method to generate interesting projections also based on similarity between the dimensions. Similar dimensions are clustered and used to create a lower-dimensional projection of the data.

The approach most similar to ours is probably *Pixnostics*, proposed by Schneidewind *et al.* [19]. They also use image-analysis techniques to rank the different lower-dimensional views of the dataset and present only the best to the user. The method provides to the user not only valuable lower-dimensional projections, but also optimized parameter settings for pixel-level visualizations. But while this approach concentrates on pixel-level visualizations as Jigsaw Maps and Pixel Bar Charts, we focus on Scatterplots and Parallel Coordinates.

Additional to the measure for classified and non-classified Scatterplots, we also propose two measures for classified Scatterplots as an alternative to [13]. Our measures first select the best projections of the dataset and therefore have the advantage, over embeddings generated by linear combination of the the original variables, that the orthogonal projection axes can be more easily interpreted by the user. As an alternative to the methods for dimension reordering for Parallel Coordinates we propose a method based on the structure presented on the low-dimensional embeddings of the dataset. Three different kind of measures to rank these embeddings are presented in this paper for class and non-class based visualizations.

3 Overview and Problem Description

Increasing dimensionality and growing volumes of data lead to the necessity of effective exploration techniques to present the hidden information and structures of high-dimensional datasets. For supporting visual exploration, the high-dimensional data is commonly mapped to low-dimensional views. Depending on the technique, exponentially many different low-dimensional views exist, which can't be analyzed manually.

A commonly used visualization technique to deal with multivariate datasets is Scatterplots. This low-dimensional embedding of the high-dimensional data in a 2D view can be interpreted easily, especially in the most common case of orthogonal linear projections. Since there are $\frac{n^2-n}{2}$ different plots for an n-dimensional dataset in a Scatterplot matrix, an automatic analysis technique to preselect the important dimensions is useful and necessary.

Another well known and widely used visualization method for multivariate datasets is Parallel Coordinates. One problem of this kind of visualization is the large number of possible arrangements of the dimension axes. For an n-dimensional dataset it has been shown, that $\frac{n+1}{2}$ permutations are needed to visualize all adjacencies, but there are $n!$ possible arrangements. An automated analysis of the visualizations can help finding the best visualizations out of all possible arrangements. We attempt to analyze the pairwise combinations of dimensions which are later assembled to find the best visualizations, reducing the visual analysis to n^2 visualizations.

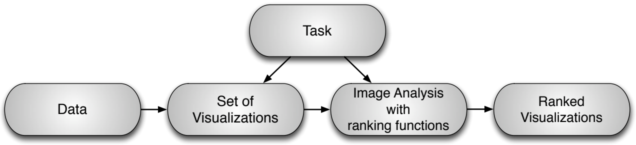


Figure 1: Working steps to get a ranked set of good visualizations of high-dimensional data.

Some applications involve classified data. We have to take this property into account when proposing our ranking functions. When dealing with unclassified data, we search for patterns or correlations between the data points. This might reveal important characteristics that can be of interest to the user. In order to see the structure of classified data, it is necessary for the visualizations to separate the clusters or at least to have a minimal overlap. The greater the number of classes, the more difficult the separation.

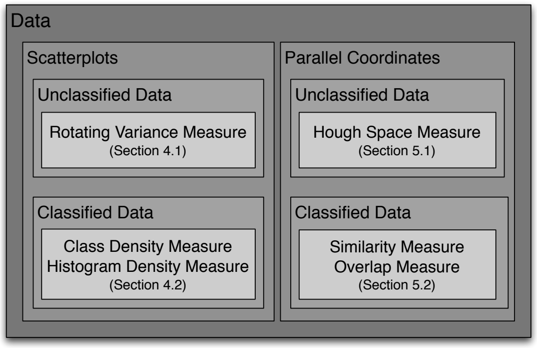


Figure 2: Overview and classification of our methods. We present measures for Scatterplots and Parallel Coordinates using classified and unclassified data.

In our paper we describe ranking functions that deal with visualizations of classified and unclassified data. An overview of our approach is presented in Figure 1. We start from a given multivariate dataset and create the low-dimensional embeddings (visualizations). According to the given task, there are different visualization methods and different ranking functions, that can be applied to these visualizations. The functions can measure the quality of the views and provide a set of useful visualizations. An overview of these techniques is shown in Figure 2. For Scatterplots on unclassified data, we developed the *Rotating Variance Measure* which highly ranks *xy*-plots with a high correlation between the two dimensions. For classified data, we propose measures that consider the class information while computing the ranking value of the images. For Scatterplots we developed two methods, a *Class Density Measure* and a *Histogram Density Measure*. Both have the goal to find the best Scatterplots showing the separating classes. For Parallel Coordinates on unclassified data, we propose a *Hough Space Measure* which searches for interesting patterns such as clustered lines in the views. For classified data, we propose two measures, one, the *Overlap Measure*, focusing on finding views with as little overlap as possible between the classes, so that the classes separate well. The other one, *Similarity Measure*, looks for correlations between the lines.

As analysis tasks, we exemplarily chose correlation search in Scatterplots (Section 4.1) and cluster search (i.e. similar lines) in

Parallel Coordinates (Section 5.1) for unclassified datasets. If class information is given, the tasks are to find views, where distinct clusters in the dataset are also well separated in the visualization (Section 4.2) or show a high level of inter- and intraclass similarity (Section 5.2).

4 Quality Measures for Scatterplots

Our approaches aim at two main tasks of visual analytics of Scatterplots: finding views which show a large extend of correlation and separating the data into well defined clusters. In Section 4.1 we propose analysis functions for task one, ranking functions for task two are then proposed in Section 4.2. In the case of unclassified, but well separable data, class labels can be automatically assigned using clustering algorithms [16, 17, 18].

4.1 Scatterplots Measures for unclassified data

4.1.1 Rotating Variance Measure

Good correlations are represented as long, skinny structures in the visualization. Due to outliers even almost perfect correlations can lead to skewed distributions in the plot and attention needs to be paid to this fact. The *Rotating Variance Measure* (RVM) is aimed at finding linear and nonlinear correlations between the pairwise dimensions of a given dataset.

First we transform the discrete Scatterplot visualization into a continuous density field. For each pixel **p** and its position $\mathbf{x} = (x,y)$ the distance to its k-th nearest sample points N_p in the visualization is computed. To obtain an estimate of the local density ρ at a pixel **p**, we define $\rho = 1/r$, where r is the radius of the enclosing sphere of the k-nearest neighbors of **p** given by

$$r = max_{i \in N_p} ||\mathbf{x} - \mathbf{x}^i||. \tag{1}$$

Choosing the k-th neighbor instead of the nearest eliminates the influence of outliers. k is chosen to be between 2 and $n-1$, so that the minimum value of r is mapped to 1. We used 4 throughout the paper. Other density estimations could of course be used as well.

Visualizations containing good correlations should, in general, have corresponding density fields with a small band of larger values, while views with less correlation have a density field consisting of many local maxima spread in the image. We can estimate this amount of spread for every pixel by computing the normalized mass distribution by taking s samples along different lines l_θ centered at the corresponding pixel positions $\mathbf{x}_{l_\theta}$ and with length equal to the image width, see Figure 3. For these sampled lines we compute the weighted distribution for each pixel position $\mathbf{x}^i$.

$$v_\theta^i = \frac{\sum_{j=1}^{s} \mathbf{p}_{l_\theta}^{s_j} ||\mathbf{x}^i - \mathbf{x}^{s_j}||}{\sum_{j=1}^{s} \mathbf{p}_{l_\theta}^{s_j}} \tag{2}$$

$$v^i = \min_{\theta \in [0,2\pi]} v_\theta^i \tag{3}$$

where $\mathbf{p}_{l_\theta}^{s_j}$ is the j-th sample along line l_θ and $\mathbf{x}^{s_j}$ is its corresponding position in the image. For pixels positioned at a maximum of a density image conveying a real correlation the distribution value will be very small, if the line is orthogonal to the local main direction of the correlation at the current position, in comparison to other positions in the image. Note that such a line can be found even in non-linear correlation. On the other hand, pixels in density images conveying no or few correlation will always have only large v values.

For each column in the image we compute the minimum value and sum up the result. The final RVM value is therefore defined as:

$$RVM = \frac{1}{\sum_x \min_y v(x,y)}, \tag{4}$$

where $v(x,y)$ is the mass distribution value at pixel position (x,y).

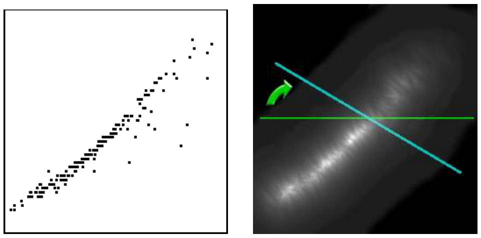

(a) (b)

Figure 3: Scatterplot example and its respective density image. For each pixel we compute the mass distribution along different directions and save the smallest value, here depicted by the blue line.

4.2 Scatterplot Measures for classified data

Most of the known techniques calculate the quality of a projection, without taking the class distribution into account. In classified data plots we can search for the class distribution in the projection, where good views should show good class separation, i.e. minimal overlap of classes.

In this section we propose two approaches to rank the scatterplots of multivariate classified datasets, in order to determine the best views of the high-dimensional structures.

4.2.1 Class Density Measure

The *Class Density Measure* (CDM) evaluates orthogonal projections, i.e. Scatterplots, according to their separation properties. The goal is to identify those plots that show minimal overlap between the classes. Therefore, CDM computes a score for each candidate plot that reflects the separation properties of the classes. The candidate plots are then ranked according to their score, so that the user can start investigating highly ranked plots in the exploration process.

In order to compute the overlap between the classes, a continuous representation for each class is necessary. In the case we are given only the visualization without the data, we assume that every color used in the visualization represents one class. We therefore first separate the classes into distinct images, so that each image contains only the information of one of the classes. For every class we estimate a continuous, smooth density function based on local neighborhoods. For each pixel **p** the distance to its k-th nearest neighbors N_p of the same class is computed and the local density is derived as described earlier in Section 4.1.

Having these continuous density functions available for each class we estimate the mutual overlap by computing the sum of the absolute difference between each pair and sum up the result:

$$CDM = \sum_{k=1}^{M-1} \sum_{l=k+1}^{M} \sum_{i=1}^{P} ||\mathbf{p}_k^i - \mathbf{p}_l^i|| \quad , \tag{5}$$

with M being the number of density images, i.e. classes respectively, $\mathbf{p}_k^i$ is the i-th pixel in the k-th density image and P is the number of pixels. If the range of the pixel values is normalized to $[0,1]$ the range for the CDM is between 0 and P. This value is large, if the densities at each pixel differ as much as possible, i.e. if one class has a high density value compared to all others. Therefore, the visualization with the fewest overlap of the classes will be given the highest value. Another property of this measure is not only in assessing well separated but also dense clusters, which eases the interpretability of the data in the visualization.

4.2.2 Histogram Density Measure

The *Histogram Density Measure* (HDM) is a density measure for Scatterplots. It considers the class distribution of the data points

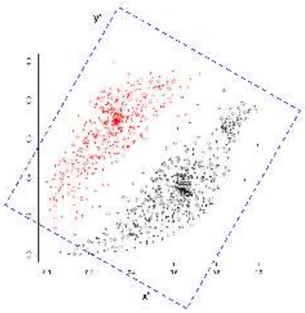


Figure 4: 2D view and rotated projection axes. The projection on the rotated plane has less overlap, and the structures of the data can be seen even in the projection. This is not possible for a projection on the original axes.

using histograms. Since we are interested in plots that show good class separations, HDM looks for corresponding histograms that show significant separation properties. To determine the best low-dimensional embedding of the high-dimensional data using HDM, a two step computation is conducted.

First, we search in the 1D linear projections which dimension is separating the data. For this purpose, we calculate the projections and rank them by the entropy value of the 1D projections separated in small equidistant parts, called histogram bins. p_c is the number of points of class c in one bin. The entropy, average information content of that bin, is calculated as:

$$H(p) = -\sum_c \frac{p_c}{\sum_c p_c} log_2 \frac{p_c}{\sum_c p_c} \tag{6}$$

$H(p)$ is 0, if a bin has only points of one class, and $log_2 M$, if it contains equivalent points of all M classes. This projection is ranked with the *1D-HDM*:

$$\begin{aligned} HDM_{1D} &= 100 - \frac{1}{Z}\sum_x (\sum_c p_c H(p)) & (7) \\ &= 100 - \frac{1}{Z}\sum_x \sum_c p_c (-\sum_c \frac{p_c}{\sum_c p_c} log_2 \frac{p_c}{\sum_c p_c}). & (8) \end{aligned}$$

where $\frac{1}{Z}$ is a normalization factor, to obtain ranking values between 0 and 100, having 100 as best value:

$$\frac{1}{Z} = \frac{100}{log_2 M \sum_x \sum_c p_c}. \tag{9}$$

In some datasets, paraxial projections are not able to show the structure of high-dimensional data. In these cases, simple rotation of the projection axes can improve the quality of the measure. In Figure 4. we show an example, where a rotation is improving the projection quality. While the paraxial projection of these classes cannot show this structures on the axes, the rotated (dotted projection) axes have less overlay for a projection on the x' axes. Therefore we rotate the projection plane and compute the *1D-HDM* for different angles θ. For each plot we choose the best 1D-HDM value. We experimentally found $\theta = 9m$ degree, with ($m \in [0, 20)$) to be working well for all our datasets.

Second, a subset of the best ranked dimensions are chosen to be further investigated in higher dimensions. All the combinations of the selected dimensions enter a PCA computation. The first two components of the PCA are plotted to be ranked by the *2D-HDM*. The *2D-HDM* is an extended version of the *1D-HDM*, for which a 2-dimensional histogram on the Scatterplot is computed. The quality is measured, exactly as for the *1D-HDM*, by summing up a weighted sum of the entropy of one bin. The measure is normalized between 0 and 100, having 100 for the best data points visualization, where each bin contains points of only one class. Also the bin neighborhood is taken into account, as for each bin p_c we sum the information of the bin itself and the direct neighborhood, labeled as u_c. Consequently the *2D-HDM* is:

$$HDM_{2D} = 100 - \frac{1}{Z}\sum_{x,y}\sum_c u_c (-\sum_c \frac{u_c}{\sum_c u_c} log_2 \frac{u_c}{\sum_c u_c}) \tag{10}$$

with the adapted normalization factor

$$\frac{1}{Z} = \frac{100}{log_2 M \sum_{x,y}(\sum_c u_c)}. \tag{11}$$

5 Quality Measures for Parallel Coordinates

When analyzing Parallel Coordinate plots, we focus on the detection of plots that show good clustering properties in certain attribute ranges. There exist a number of analytical dimension ordering approaches for Parallel Coordinates to generate dimension orderings that try to fulfill these tasks [1, 24]. However, they often do not generate an optimal parallel plot for correlation and clustering properties, because of local effects which are not taken into account by most analytical functions. We therefore present analysis functions that do not only take the properties of the data into account, but also the properties of the resulting plot.

5.1 Parallel Coordinate Measures for unclassified data

5.1.1 Hough Space Measure

Our analysis is based on the assumption that interesting patterns are usually clustered lines with similar positions and directions. Our algorithm for detecting these clusters is based on the Hough transform [9].

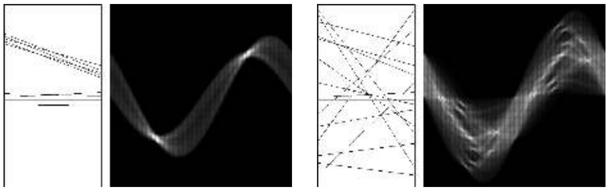
Straight lines in the image space can be described as $y = ax + b$.The main idea of the Hough transform is to define a straight line according to its parameters, i.e. the slope a and the interception b. Due to a practical difficulty (the slope of vertical lines is infinite) the normal representation of a line is:

$$\rho = xcos\theta + ysin\theta \tag{12}$$

Using this representation, for each non-background pixel in the visualization, we have a distinct sinusoidal curve in the $\rho\theta$-plane, also called Hough or accumulator space. An intersection of these curves indicates that the corresponding pixels belong to the line defined by the parameters (ρ_i, θ_i) in the original space. Figure 5 shows two synthetic examples of Parallel Coordinates and their respective Hough spaces: Figure 5(a) presents two well defined line clusters and is more interesting for the cluster identification task than Figure 5(b), where no line cluster can be identified. Note that the bright areas in the $\rho\theta$-plane represent the clusters of lines with similar ρ and θ.

To reduce the bias towards long lines, e.g. diagonal lines, we scale the pairwise visualization images to an $n \times n$ resolution, usually 512×512. The accumulator space is quantized into a $w \times h$ cell grid, where w and h control the similarity sensibility of the lines. We use 50×50 grids in our examples. A lower value for w and h reduces the sensibility of the algorithm because lines with a slightly different ρ and θ are mapped to the same accumulator cells.

Based on our definition, good visualizations must contain fewer well defined clusters, which are represented by accumulator cells with high values. To identify these cells, we compute the median value m as an adaptive threshold that divides the accumulator function $h(x)$ into two identical parts:



(a) (b)

Figure 5: Synthetic examples of Parallel Coordinates and their respective Hough spaces: (a) presents two well defined line clusters and is more interesting for the cluster identification task than (b), where no line cluster can be identified. Note that the bright areas in the $\rho\theta$-plane represent the clusters of lines with similar ρ and θ.

$$\begin{aligned} \frac{\sum h(x)}{2} &= \sum g(x) \quad \text{, where} \qquad (13) \\ g(x) &= \begin{cases} x & \text{if } x \leq m; \\ m & \text{else.} \end{cases} \end{aligned}$$

Using the median value, only a few clusters are selected in an accumulator space with high contrast between the cells (See Fig 5(a)), while in a uniform accumulator space many clusters are selected (See Fig 5(b)). This adaptive threshold is not only necessary to select possible line clusters in the accumulator space, but also to avoid the influence of outliers and occlusion between the lines. In the occlusion case, a point that belongs to two or more lines is computed just once in the accumulator space.

The final goodness value for a 2D visualization is due to the number of accumulator cells n_{cells} that have a higher value than m normalized by the total number of cells (wh) to the interval $[0,1]$:

$$s_{i,j} = 1 - \frac{n_{cells}}{wh}, \qquad (14)$$

where i, j are the indices of the respective dimensions, and the computed measure $s_{i,j}$ presents higher values for images containing well defined line clusters (similar lines) and lower values for images containing lines in many different directions and positions.

Having combined the pairwise visualizations, we can now compute the overall quality measure by summing up the respective pairwise measurements. This overall quality measure of a parallel visualization containing n dimensions is:

$$HSM = \sum_{a_i \in I} s_{a_i, a_{i+1}}, \qquad (15)$$

where I is a vector containing any possible combination of the n dimensions indices. In this way we can measure the quality of any given visualization, using Parallel Coordinates.

Exhaustively computing all n-dimensional combinations in order to choose the best/worst ones, requires a very long computation time and becomes unfeasible for a large n. In these cases, in order to search for the best n-dimensional combinations in a feasible time, an algorithm to solve a Traveling Salesman Problem is used, e.g. the A*-Search algorithm [8] or others [2]. Instead of exhaustively combining all possible pairwise visualizations, these kind of algorithms would compose only the best overall visualization.

5.2 Parallel Coordinates Measures for classified data

While analyzing Parallel Coordinates visualizations with class information, we consider two main issues. First, in good Parallel Coordinates visualizations, the lines that belong inside a determined class must be quite similar (inclination and position similarity). Second, visualizations where the classes can be separately observed and that contain less overlapping are also considered to be good. We developed two measures for classified Parallel Coordinates that take these matters into account: the *Similarity Measure* that encourages inner class similarities, and the *Overlap Measure* that analyzes the overlap between classes. Both are based on the measure for unclassified data presented in section 5.1.

5.2.1 Similarity Measure

The similarity measure is a direct extension of the measure presented in section 5.1. For visualizations containing class information, the different classes are usually represented by different colors. We separate the classes into distinct images, containing only the pixels in the respective class color, and compute a quality measure s_k for each class, using equation (14). Thereafter, an overall quality value s is computed as the sum of all class quality measures:

$$SM = \sum_k s_k. \qquad (16)$$

Using this measure, we encourage visualizations with strong inner class similarities and slightly penalize overlapped classes. Note that due to the classes overlap, some classes have many missing pixels, which results in a lower s_k value compared to other visualizations where less or no overlap between the classes exists.

5.2.2 Overlap Measure

In order to penalize overlap between classes, we analyze the difference between the classes in the Hough space (see section 5.1). As in the similarity measure, we separate the classes to different images and compute the Hough transform over each image. Once we have a Hough space h for each class, we compute the quality measure as the sum of the absolute difference between the classes:

$$OM = \sum_{k=1}^{M-1} \sum_{l=k+1}^{M} \sum_{i=1}^{P} ||\mathbf{h}_k^i - \mathbf{h}_l^i|| \qquad (17)$$

Here M is the number of Hough space images, i.e. classes respectively and P is the number of pixels. This value is high if the Hough spaces are disjoint, i.e. if there is no large overlap between the classes. Therefore, the visualization with the smallest overlap between the classes receives the highest values.

Another interesting use of this measure is to encourage or search for similarities between different classes. In this case, the overlap between the classes is desired, and the previously computed measure can be inverted to compute suitable quality values:

$$OM\_INV = 1/OM. \qquad (18)$$

6 Application and Evaluation

We tested our measures on a variety of real datasets. We applied our *Class Density Measure (CDM), Histogram Density Measure (HDM), Similarity Measure (SM)* and *Overlap Measure (OM)* on classified data, to find views on the data which try to either separate the data or show similarities between the classes. For unclassified data, we applied our *Rotating Variance Measure (RVM)* and *Hough Space Measure (HSM)* in order to find linear or non-linear correlations and clusters in the datasets, respectively. Except for the HDM, we chose to present only relative measures, i.e. all calculated measures are scaled so that the best visualization is assigned 100 and the worst 0. For the HDM, we chose to present the unchanged measure values, as the HDM allows an easy direct interpretation, with a value of 100 being the best and 0 being the worst possible constellation. If not stated otherwise our examples are proof-of-concepts, and interpretations of some of the results should be provided by domain experts.

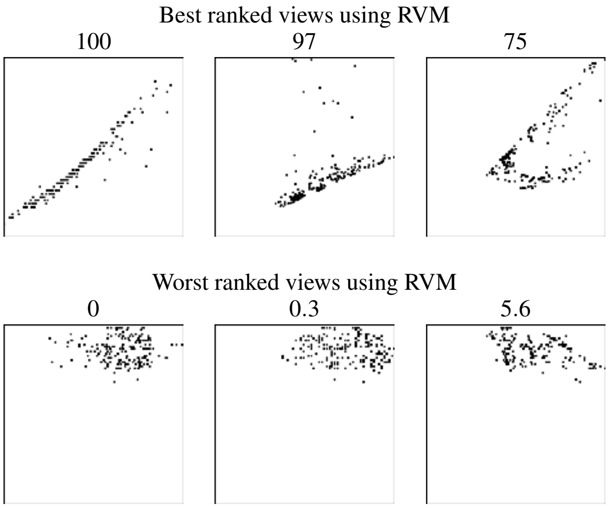


Figure 6: Results for the Parkinson's Disease dataset using our RVM measure (Section 4.1). While clumpy non-correlation bearing views are punished (bottom row), views containing more correlation are preferred (top row).

We used the following datasets: *Parkinson's Disease* is a dataset composed of 195 voice measures from 31 people, 23 with Parkinson's disease [15, 14]. Each of the 12 dimensions is a particular voice measure. *Olives* is a classified dataset with 572 olive oil samples from nine different regions in Italy [25]. For each sample the normalized concentrations of eight fatty acids are given. The large number of classes (regions) poses a challenging task to the algorithms trying to find views in which all classes are well separated. *Cars* is a previously unpublished dataset of used cars automatically collected from a national second hand car selling website. It contains 7404 cars listed with 24 different attributes, including price, power, fuel consumption, width, height and others. We chose to divide the dataset into two classes, benzine and diesel to find the similarities and differences between these. *Wisconsin Diagnostic Breast Cancer* (WDBC) dataset consists of 569 samples with 30 real-valued dimensions each [20]. The data is classified into malign and benign cells. The task is to find the best separating dimensions. *Wine* is a classified dataset with 178 instances and 13 attributes describing chemical properties of Italian wines derived from three different cultivars.

First we show our results for RVM on the *Parkinson's Disease* dataset [15, 14]. The three best and the three worst results are shown in Figure 6. Interesting correlations have been found between the dimensions Dim 9(DFA) and Dim 12(PPE), Dim 2(MDVP:Fo(Hz)) and Dim 3(MDVP:Fhi(Hz)), as well as Dim 2(MDVP:Fo(Hz)) and Dim 4(MDVP:Flo(Hz)) (Fig. 6). On the other hand visualizations containing few or no correlation information at all received a low value.

In Figure 7 the results for the *Olives* dataset using our CDM measure are shown. Even though a view separating all different olive classes does not exist, the CDM reliably choses three views which separate the data well in the dimensions Dim 4(oleic) and Dim 5(linoleic), Dim 1(palmitic) and Dim 5(linoleic) as well as Dim 1(palmitic) and Dim 4(oleic).

We also applied our HDM technique to this dataset. First the *1D-HDM* tries to identify the best separating dimensions, as presented in Section 4.2.2. The dimensions Dim 1(palmitic), Dim 2(palmitoleic), Dim 4(oleic), Dim 5(linoleic) and Dim 8(eicosenoic) were ranked as the best separating dimensions. We computed all subsets of these dimensions and ranked their PCA views with the *2D-HDM*. In the best ranked views presented in Figure 8 the different classes are well separated. Compared to the upper row in Figure 7, the visualization uses the screen space better, which is due to the PCA transformation.

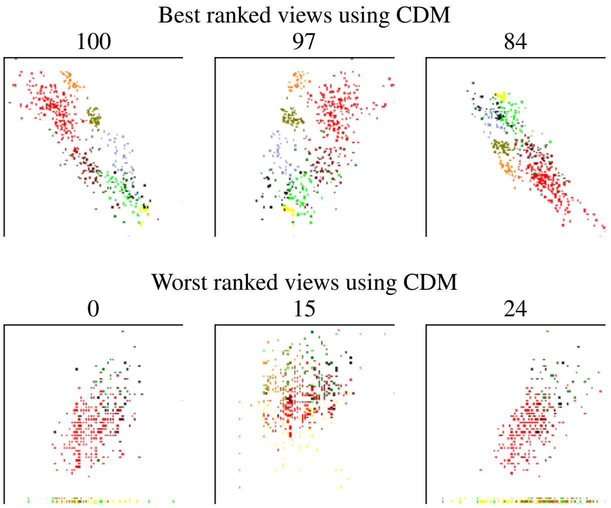


Figure 7: Results for the olive dataset using our CDM measure (Section 4.2.1). The different colors depict the different classes (regions) of the dataset. While it is impossible for this dataset to find views completely separating all classes, our CDM measure still found views where most of the classes are mutually separated (top row). In the worst ranked views the classes clearly overlap with each other (bottom row).

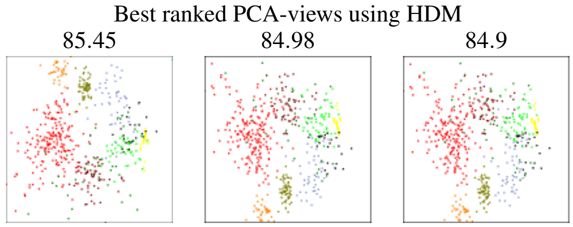


Figure 8: Results for the Olives dataset using our HDM measure (Section 4.2.2). The best ranked plot is the PCA of Dim(4,5,8) were the classes are good visible, the second best is the PCA of Dim(1,2,4) and the third is the PCA on all 8 dimensions. The differences between the last two are small, because the variance in that additional dimensions for the 3rd relative to the 2nd is not big. The difference between these and the first is good visible.

To measure the value of our approaches for Parallel Coordinates we estimated the best and worst ranked visualizations of different datasets. The corresponding visualizations are shown in Figure 9, 10 and 11. For a better comparability the visualizations have been cropped after the display of the 4th dimension. We used a size of 50×50 for the Hough accumulator in all experiments. The algorithms are quite robust with respect to the size and using more cells generally only increases computation time but has little influence on the result. Figure 9 shows the ranked results for the Parkinsons Disease dataset using our *Hough Space Measure*.

The HSM algorithm prefers views with more similarity in the distance and inclination of the different lines, resulting in the prominent small band in the visualization of the *Parkinsons Disease* dataset, which is similar to clusters in the projected views of these dimension, here between Dim 3(MDVP:Fhi(Hz)) and Dim 12(PPE) as well as Dim 6(HNR) and Dim 11(spread2).

Applying our *Hough Similarity Measure* to the *Cars* dataset we can see that there seem to be barely any good clusters in the dataset (see Figure 10). We verified these by exhaustively looking at all pairwise projections. However, the only dimension where the classes can be separated and at least some form of cluster can be reliably found is (Dim 6(RPM)), in which cars using diesel generally have a lower value compared to benzine (Fig. 10 top row). Also the similarity of the majority in Dim 15(Height), Dim 18(Trunk) and Dim 3(Price) can be detected. Obviously cars using diesel are cheaper, this might be due to the age of the diesel cars, but age was unfortunately not included in the data base. On the other hand the worst ranked views using the HSM (Fig. 10, bottom row) are barely interpretable, at least we weren't able to extract any useful information.

In Figure 11 the results for our *Hough Overlap Measure* applied to the *WDBC* dataset are shown. This result is very promising. In the top row, showing the best plots, the malign and benign are pretty well separated. It seems that the dimensions Dim 22(radius (worst)), Dim 9(concave points (mean)), Dim 24 (perimeter (worst)), Dim 29(concave points (mean)) and Dim 25(Area (worst)) separate the two classes pretty well. We showed these results to a medical scientist who confirmed our findings, that these measures are some of the most reliable to discern cancer cells, as cancer cells tend to either divide themselves more often, which results in larger nuclei due to the mitosis, or do not completely divide resulting in deformed, concave nuclei.

7 Conclusion

In this paper we presented several methods to aid and potentially speed up the visual exploration process for different visualization techniques. In particular, we automized the ranking of Scatterplot and Parallel Coordinates visualizations for classified and unclassified data for the purpose of correlation and cluster separation. In the future a *ground truth* could be generated, by letting users choose the most relevant visualizations from a manageable test set and compare them to the automatically generated ranking in order to prove our methods. Some limitations are recognized as it is not always possible to find good separating views, due to a growing number of classes and due to some multivariate relations, which is a general problem and not related to our techniques. As future work, we plan to apply α-transparency and clutter reduction to overcome overplotting.

Furthermore, we will aim at finding measures for other, maybe more complex tasks, and we would like to generalize our techniques so that they can be applied and adapted to further visualization techniques.

Acknowledgements

The authors would like to acknowledge the contributions of the Institute for Information Systems at the Technische Universität Braunschweig (Germany). This work was supported in part by a grant from the German Science Foundation (DFG) within the strategic research initiative on Scalable Visual Analytics.

References

[1] M. Ankerst, S. Berchtold, and D. A. Keim. Similarity clustering of dimensions for an enhanced visualization of multidimensional data. *Information Visualization, IEEE Symposium on*, 0, 1998.

[2] D. L. Applegate, R. E. Bixby, V. Chvatal, and W. J. Cook. *The Traveling Salesman Problem: A Computational Study (Princeton Series in Applied Mathematics)*. Princeton University Press, January 2007.

[3] D. Asimov. The grand tour: a tool for viewing multidimensional data. *Journal on Scientific and Statistical Computing*, 6(1):128–143, 1985.

[4] D. Cook, A. Buja, J. Cabreta, and C. Hurley. Grand tour and projection pursuit. *Journal of Computational and Statistical Computing*, 4(3):155–172, 1995.

[5] M. A. Fisherkeller, J. H. Friedman, and J. W. Tukey. *Prim-9: An interactive multi-dimensional data display and analysis system*, volume In W. S. Sleveland, editor. Chapman and Hall, 1987.

[6] J. Friedman and J. Tukey. A projection pursuit algorithm for exploratory data analysis. *Computers, IEEE Transactions on*, C-23(9):881–890, Sept. 1974.

[7] D. Guo, J. Chen, A. M. MacEachren, and K. Liao. A visualization system for space-time and multivariate patterns (vis-stamp). *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1461–1474, 2006.

[8] P. N. Hart, N. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans, Sys. Sci. Cybernetics*, S.S.C.-4(2):100–107, 7 1968.

[9] P. V. C. Hough. Method and means for recognizing complex patterns. *US Patent*, 3069654, December 1962.

[10] P. J. Huber. Projection pursuit. *The Annals of Statistics*, 13(2):435–475, 1985.

[11] A. Inselberg. The plane with parallel coordinates. *The Visual Computer*, 1(4):69–91, December 1985.

[12] D. A. Keim, M. Ankerst, and M. Sips. *Visual Data-Mining Techniques*, pages 813–825. Kolam Publishing, 2004.

[13] Y. Koren and L. Carmel. Visualization of labeled data using linear transformations. *Information Visualization, IEEE Symposium on*, 0:16, 2003.

[14] M. A. Little, P. E. McSharry, E. J. Hunter, and L. O. Ramig. Suitability of dysphonia measurements for telemonitoring of parkinson's disease. In *IEEE Transactions on Biomedical Engineering*.

[15] M. A. Little, P. E. Mcsharry, S. J. Roberts, D. A. E. Costello, and I. M. Moroz. Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection. *BioMedical Engineering OnLine*, 6:23+, June 2007.

[16] S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

[17] J. B. Macqueen. Some methods for classification and analysis of multivariate observations. In *Procedings of the Fifth Berkeley Symposium on Math, Statistics, and Probability*, volume 1, pages 281–297. University of California Press, 1967.

[18] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems 14*, pages 849–856. MIT Press, 2001.

[19] J. Schneidewind, M. Sips, and D. Keim. Pixnostics: Towards measuring the value of visualization. *Symposium On Visual Analytics Science And Technology*, 0:199–206, 2006.

[20] W. Street, W. Wolberg, and O. Mangasarian. Nuclear feature extraction for breast tumor diagnosis. *IS&T / SPIE International Symposium on Electronic Imaging: Science and Technology*, 1905:861–870, 1993.

[21] J. Tukey and P. Tukey. Computing graphics and exploratory data analysis: An introduction. In *Proceedings of the Sixth Annual Conference and Exposition: Computer Graphics 85. Nat. Computer Graphics Assoc.*, 1985.

[22] M. O. Ward. Xmdvtool: Integrating multiple methods for visualizing multivariate data. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 326–333, 1994.

[23] L. Wilkinson, A. Anand, and R. Grossman. Graph-theoretic scagnostics. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 157–164, 2005.

[24] J. Yang, M. Ward, E. Rundensteiner, and S. Huang. Visual hierarchical dimension reduction for exploration of high dimensional datasets, 2003.

[25] J. Zupan, M. Novic, X. Li, and J. Gasteiger. Classification of multicomponent analytical data of olive oils using different neural networks. In *Analytica Chimica Acta*, volume 292, pages 219–234, 1994.

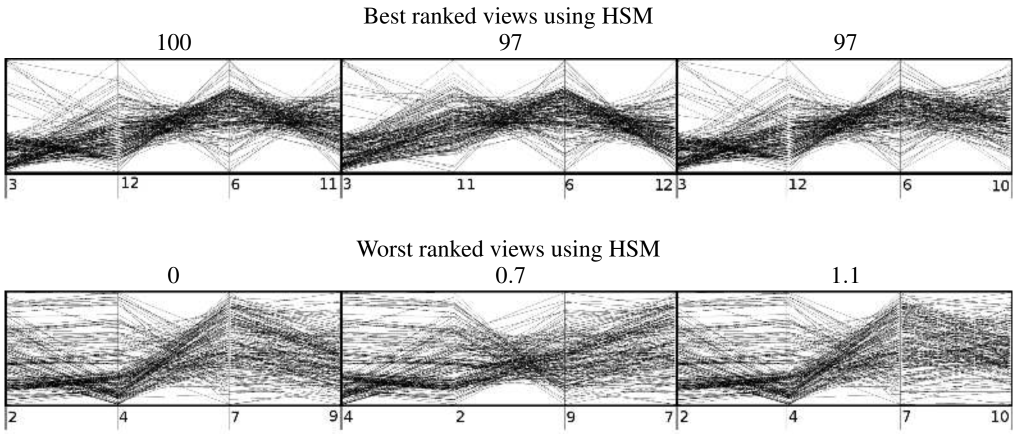


Figure 9: Results for the non-classified version of the *Parkinsons Disease* dataset. Best and worst ranked visualizations using our HSM measure for non-classified data (ref. Section 5.1.1). (a) Top row: The three best ranked visualizations and their respective normalized measures. Well defined clusters in the dataset are favored. Bottom row: The three worst ranked visualizations. The large amount of spread exacerbates interpretation. Note that the user task related to this measure is not to find possible correlation between the dimensions but to detect good separated clusters.

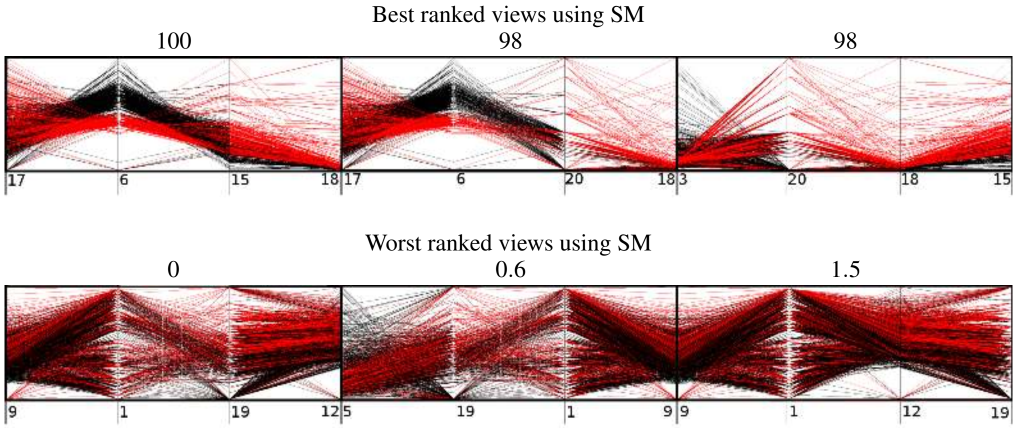


Figure 10: Results for the *Cars* dataset. Cars using benzine are shown in black, diesel in red. Best and worst ranked visualizations using our Hough similarity measure (Section 5.2.1) for Parallel Coordinates. (a) Top row: The three best ranked visualizations and their respective normalized measures. Bottom row: The three worst ranked visualizations.

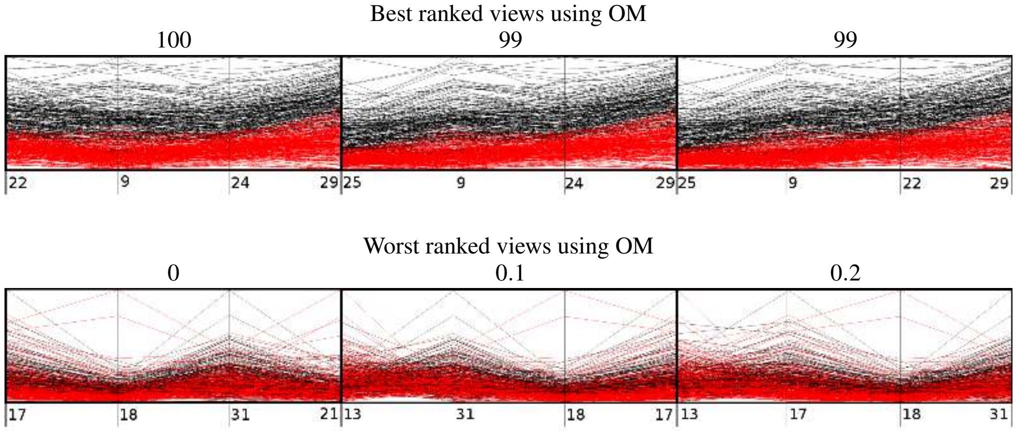


Figure 11: Results for the *WDBC* dataset. Malign nuclei are colored black while healthy nuclei are red. Best and worst ranked visualizations using our overlap measure (Section 5.2.1) for Parallel Coordinates. (a) Top row: The three best ranked visualizations. Despite good similarity, which are similar to clusters, visualizations are favored that minimize the overlap between the classes, so the difference between malign and benign cells becomes more clear. Bottom row: The three worst ranked visualizations. The overlap of the data complicates the analysis, the information is useless for the task of discriminating malign and benign cells.