TECHNICAL RESEARCH REPORT

A Rank-by-Feature Framework for Unsupervised Multidimensional Data Exploration Using Low Dimensional Projections (2004)

by Jinwook Seo, Ben Shneiderman

TR 2005-54

A Rank-by-Feature Framework for Unsupervised Multidimensional Data Exploration Using Low Dimensional Projections

Jinwook Seo* and Ben Shneiderman†
Department of Computer Science &
Human-Computer Interaction Laboratory, Institute for Advanced Computer Studies
University of Maryland, College Park, MD 20742 USA

ABSTRACT

Exploratory analysis of multidimensional data sets is challenging because of the difficulty in comprehending more than three dimensions. Two fundamental statistical principles for the exploratory analysis are (1) to examine each dimension first and then find relationships among dimensions, and (2) to try graphical displays first and then find numerical summaries [1]. We implement these principles in a novel conceptual framework called the rank-by-feature framework. In the framework, users can choose a ranking criterion interesting to them and sort 1D or 2D axis-parallel projections according to the criterion. We introduce the *rank-by-feature prism* that is a color-coded lower-triangular matrix that guides users to desired features. Statistical graphs (histogram, boxplot, and scatterplot) and information visualization techniques (overview, coordination, and dynamic query) are combined to help users effectively traverse 1D and 2D axis-parallel projections, and finally to help them interactively find interesting features.

CR Categories: I.6.9.c Information visualization, H.5.2 User Interfaces, H.5.2.f Graphical user interfaces, I.5.2.b Feature evaluation and selection, H.2.8.d Data mining, H.1.2.a Human factors, H.2.8.c Data and knowledge visualization, H.2.8.h Interactive data exploration and discovery

Keywords: information visualization, exploratory data analysis, dynamic query, feature detection/selection, statistical graphics.

1 INTRODUCTION

Multidimensional data sets are common in various data analysis applications; e.g., microarray data analysis, census data analysis, and market basket analysis. A data set that can be represented in a spreadsheet where there are more than three columns can be thought of as multidimensional. Without losing generality, we can assume that each column is a dimension (or a variable), and each row is a data item. Dealing with multidimensionality has been challenging for researchers in many disciplines due to the difficulty in comprehending more than three dimensions and the computational overhead.

One of the commonly used methods to handle multidimensionality is to use low dimensional projections. Since human eyes and minds are effective in understanding at most two-dimensional (2D) or three-dimensional (3D) objects, and computer displays are intrinsically 2D, 2D projections have been widely used as useful representations of the original multidimensional data.

There are three categories of low dimensional projection techniques according to the way axes of the projection are composed: (1) General projection methods use a linear/nonlinear combination of dimensions for an axis of the projection plane. For example, principal component analysis (PCA) and multidimensional scaling (MDS) are famous techniques in this category, (2) Axis-parallel projection methods use a dimension for an axis of the projection plane. For example, one of the original dimensions is selected as the horizontal axis, and another as the vertical axis. Sometimes, other dimensions can be mapped as color, size, length, angle, etc., (3) Other methods use axes that are not directly derived from any dimension. For example, the horizontal axis of the parallel coordinate view is a completely new concept where dimensions are aligned sequentially.

Although, theoretically, techniques in the first category can generate all possible 2D projections of a multidimensional data set, it suffers from users' difficulty in interpreting 2D projections whose axes are linear/nonlinear combination of two or more dimensions. For example, even though a user finds an interesting linear correlation on a projection where the horizontal axis is 3*(*body weight*)-2*(*height*) and the vertical axis is 2*(*body weight*)+3*(*height*), the finding is not so useful, because users can hardly understand the meaning of the projection.

Techniques in the second category have a limitation that features can be detected in only the two selected dimensions. However, since it is and easy for users to interpret the meaning of the projection, these techniques have been widely used and implemented in visualization tools. A problem with the techniques belonging to this category (and to the first category) is how to deal with the large number of possible low dimensional projections. If we have an *N*-dimensional data set, we can generate *N**(*N*-1)/2 2D projections using the technique in the second category.

Research has been conducted in several fields to address this problem. Most work focuses on the detection of dimensions that are most useful for a certain application. For example, in data mining, researchers want to find dimensions with which they can build a better classifier. In the field of pattern recognition, researchers want to find a subset of dimensions with which they can better detect specific patterns in a data set. In subspace-based clustering analysis, people want to find projections where it is easy to naturally partition the data set. Unlike previous applications, in the exploratory analysis, users don't know in advance what they are looking for or what kinds of projections are interesting.

*email: jinwook@cs.umd.edu
†email: ben@cs.umd.edu

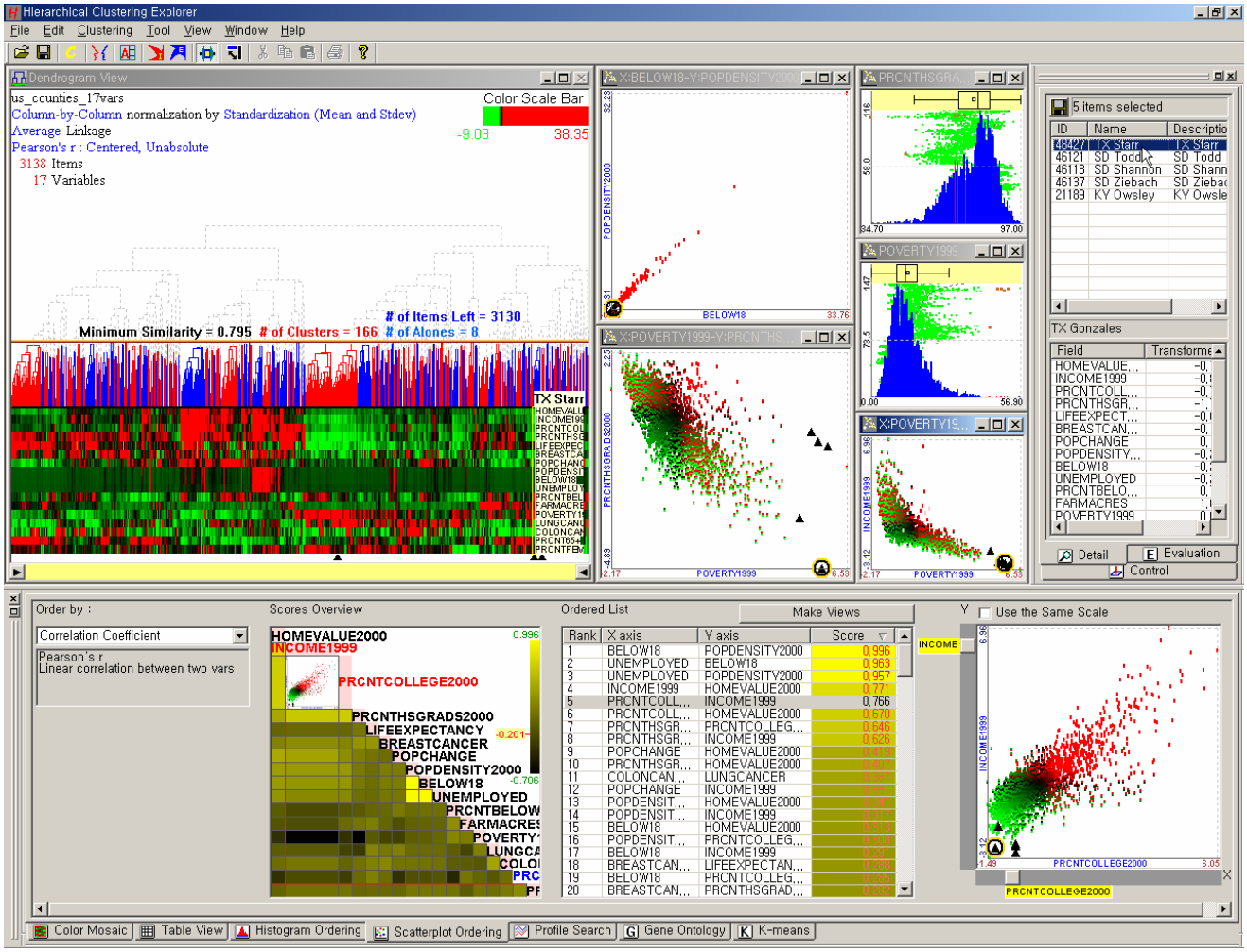


Figure 1. The Hierarchical Clustering Explorer (HCE) The rank-by-feature framework is implemented as two new tab windows in HCE 3.0. The main view is the dendrogram view where users can interactively explore hierarchical clustering results [16]. Whenever users identify an interesting projection in the rank-by-feature framework, they can generate a separate child window that will interactively coordinate with all other views in HCE 3.0. A 2D scatterplot ordering result (section 3.2) by correlation coefficient is shown with the U.S. counties data set (section 4). The high-rank 2D projections (or scatterplots) and low-rank 2D projections are shown in separate views together with the clustering result view and histogram views. Five counties that have a high poverty level and low income level are selected in a scatter plot view and they are all highlighted in other views (with black triangles).

In such a situation, it is necessary to allow users not only to interactively specify what kinds of projections are interesting, but also to easily identify interesting projections. There have been many research projects that utilize low dimensional projections for exploratory data analysis, but little work has been done to help users systematically examine low dimensional projections. In this paper, we present a conceptual framework for interactive feature detection named ***rank-by-feature framework*** to address these issues. In the rank-by-feature framework, users can select an interesting ranking criterion from the carefully chosen available ranking criteria, and then all possible axis-parallel projections are ranked by the selected ranking criterion. The ranking result is visually presented in a tabular display where each cell represents a projection and is color-coded by the ranking score, so that users can not only easily perceive the most (or the least) interesting projection, but also grasp the overall ranking score distribution. It is also possible to manually browse projections by rapidly changing the dimension for an axis using the item slider attached to the corresponding axis of the projection view (histogram and boxplot for 1D, and scatterplot for 2D).

We implement the rank-by-feature framework in our interactive exploration tool for multidimensional data, the Hierarchical Clustering Explorer (HCE) [16] (Figure 1) as two new tab windows ("Histogram Ordering" for 1D projections, and "Scatterplot Ordering" for 2D projections). By using the rank-by-feature framework, users can easily find interesting histograms and scatterplots, and generate separate windows to visualize those plots. All these plots are interactively coordinated with other views (e.g. dendrogram and color mosaic view, tabular view, parallel coordinate view) in HCE. If users select a group of items in any view, users can see the selected items highlighted in all other views. Thus, it is possible to comprehend the data from various perspectives to get more meaningful insights.

The next section introduces related work, and then we present the rank-by-feature framework for axis-parallel 1D and 2D projections with examples in section 3. An application example is presented in section 4. Discussion and possible future work is in section 5. We conclude the paper in section 6.

2 Related Work

Two-dimensional projections have been utilized in many visualization tools and graphical statistics tools for multidimensional data analysis. Geometric projection techniques are used to find interesting projections of multidimensional data sets. PCA, MDS, and parallel coordinates [15] are also projection techniques. Self-organizing map (SOM) [2] can be thought of as a kind of projection technique. Taking a look at only a single projection for a multidimensional data is not enough to capture interesting features in the original data since one projection may obscure the features [3]. Thus it is inevitable to scrutinize a series of projections to reveal the features of the data set.

The idea of projection pursuit [3] is to find the most interesting low dimensional projections that are important to identify interesting features in the multidimensional data set. An automatic projection pursuit method is the grand tour. The grand tour [4] is a method for viewing multidimensional data via orthogonal projection onto a sequence of two-dimensional subspaces. It changes the viewing direction, generating a movie-like animation that makes a complete search of the original space. However, it would take up to three hours to complete a reasonably complete visual search in four dimensions [5]. It is evident that an exhaustive visual search is out of the question if the number of dimensions exceeds 4.

Friedman and Tukey [3] devised a method to automate the task of projection pursuit. They defined interesting projections as ones deviating from the normal distribution, and provide a numerical index to indicate the interestingness of the projection. When an interesting projection is found, the features on the projection are extracted and projection pursuit is continued until there is no remaining feature found. XGobi [6] is a widely-used graphical tool that implemented both the grand tour and the projection pursuit, but not ranking.

These automatic projection pursuit methods made impressive gains in the problem of multidimensional data analysis, but they have limitations. One of the most important problems is the difficulty in interpreting the solutions from the automatic projection pursuit. Since the axes are the linear combination of the variables (or dimensions) of the original data, it is hard to determine what the projection actually means to users. Conversely, this is one of the reasons that axis-parallel projections are used in many multidimensional analysis tools [7][8][9].

Feature selection is a process that chooses an optimal subset of features according to a certain criterion [10]. It has been studied in machine learning and data mining areas mostly for supervised classification problems. In these areas, feature means dimension. Basically, the goal is to find a good feature, or a good subset of features that contribute to the construction of a good classifier. Unsupervised feature selection methods are also studied in close relation with unsupervised clustering algorithms. In this case, the goal is to find an optimal subset of features with which clusters are well identified [8][11][12].

In the information visualization field, there are researchers who tried to optimally arrange dimensions in a way that similar or correlated dimensions are put close to each other so that users can find interesting patterns using a better visualization of a multidimensional data [13][14][17]. Most closely related to our work is Yang et al. [17], who proposed innovative dimension ordering methods for better visualizations. They rearrange dimensions within a single display according to similarities between dimensions or relative importance defined by users. Our work is to rank all dimensions or all pairs of dimensions whose visualization contains desired features.

There are also some research tools and commercial products for helping users to find more informative visualizations. Spotfire [9] has a guidance tool called "ViewTip" for rapid assessment of potentially interesting scatterplots, which shows the ordered list of all possible scatterplots from the one with highest correlation to the one with lowest correlation. Guo et al. [15] also evaluated all possible axis-parallel 2D projections according to the maximum conditional entropy to identify ones that are most useful to find clusters. They visualized the entropy values in a matrix display called the entropy matrix. Our work takes these two nascent ideas with the goal of developing a potent framework for discovery.

The contributions of the rank-by-feature framework include the following:

- a general framework where users can incorporate their interests into interactive exploratory analysis process by selecting a ranking criterion among available ranking criteria.
- a ranking not only for dimensions but also for pairs of dimensions according to users' interest.
- an implementation in the Hierarchical Clustering Explorer (HCE), that is well coordinated with other views such as the dendrogram view, parallel coordinates view, and so on.

3 Rank-by-Feature Framework

This section presents a conceptual framework named the rank-by-feature framework for interactive feature detection in unsupervised multidimensional data. We use the term, "features" in a broader sense. What we mean by a "feature" is not only a dimension (or a variable) but also any interesting characteristic (patterns or items) of the data set. Since, as we discussed in the introduction, general geometric projections could be useful but most people have difficulty in interpreting these projections where an axis can be combination of dimensions, we concentrate on axis-parallel projections. However, the rank-by-feature framework can be used with general geometric projections if the number of possible projections is constrained. 3D projections are sometimes useful to reveal hidden features, but they suffer from occlusion and the disorientation brought on by the cognitive burden of navigation. On the other hand, 2D projections are limited to two dimensions (or variables) at a time, but most users easily understand them, and they can concentrate on the data itself rather than being distracted by navigation controls.

Detecting interesting features in low dimensions (1D or 2D) by maximally utilizing powerful human perceptual abilities is crucial to understand the original multidimensional data set. Familiar graphical displays such as histograms, scatterplots, and other well-known 2D plots are effective to reveal features including basic summary statistics, and even unexpected features in the data set. There are also many algorithmic or statistical techniques that are especially effective in low dimensional spaces. While there have been many approaches utilizing such visual displays and low dimensional techniques, most of them lack a systematic framework that organizes such functionalities to help users' feature detection tasks.

There are two basic principles for a better exploration of an unsupervised multidimensional data: (1) examine individual dimensions first and then the relationships among the dimensions (or variables), (2) use graphical displays first and then numerical summaries of specific aspects of the data [1]. Abiding by these principles, we will present the rank-by-feature framework and its interface for 1D projections first and then those for 2D projections. Users can begin their exploration with the main graphical display in each interface, for example, histograms for 1D and scatterplots for 2D, and they can also find the numerical summaries for more detail.

The rank-by-feature framework helps users systematically traverse low dimensional (1D or 2D) projections to maximize the benefit of exploratory tools. In this framework, users can select an interesting ranking criterion from the available criteria. Users can rank low dimensional projections (1D or 2D) of the multidimensional data set according to the strength of the selected feature in the projection. When there are many dimensions, the number of possible projections is too large to investigate them one by one until users find interesting ones. The rank-by-feature framework relieves users from such burden by recommending projections interesting to users in an ordered manner defined by a ranking criterion that users selected. This framework has been implemented in our interactive visualization tool, HCE 3.0 (www.cs.umd.edu/hcil/hce/) [16].

3.1 1D Histogram Ordering

Users begin the exploratory analysis of a multidimensional data by scrutinizing each dimension (or variable) one by one. Just looking at the distribution of values of a dimension gives us useful insight into the dimension. The most familiar graphical display tools for 1D data are *histograms* and *boxplots*. Histograms graphically reveal the scale and skewness of the data, the number of modes, gaps, and outliers in the data. Boxplots are also excellent tools for detecting and illustrating location and variation changes of a dimension. They graphically show the five-number summary (the minimum, the first quartile, the median, the third quartile, and the maximum). These numbers provide users with an informative summary of a dimension's center and spread, and they are the foundation of multidimensional data analysis for deriving a model for the data or for selecting dimensions for effective visualization.

We will combine histogram and boxplot to use them as the main display for the rank-by-feature framework for 1D projections. Figure 2 shows the interactive interface design for the rank-by-feature framework for 1D projections. The interface consists of four coordinated parts: the *control panel*, the *score overview*, the *ordered list*, and the *histogram browser*. Users can select a ranking criterion from a combo box in the control panel, and then they see the overview of scores for all dimensions in the score overview according to the selected ranking criterion. All dimensions are aligned from top to bottom in the original order, and each dimension is color-coded by the score value. The greater the score is, the brighter the color is (The color scale and mapping are shown at the top right corner of the overview). Thus, users can easily see the overall pattern of the score distribution, and more importantly they can *preattentively* identify the dimension of the highest/lowest score in this overview. Once they identify an interesting row on the score overview, they can just mouse over the row to view the numerical score value and the name of the dimension are shown in a tooltip window (Figure 2).

The mouseover event is also instantaneously relayed to the ordered list and the histogram browser, so that the corresponding list item is highlighted in the ordered list and the corresponding histogram and boxplot are shown in the histogram browser. The score overview, the ordered list, and the histogram browser are interactively coordinated according to the change of the dimension in focus. In other words, a change of dimension in focus in one of the three components leads to the instantaneous change of dimension in focus in the other two components.

In the ordered list, users can see the numerical detail about the distribution of each dimension in an orderly manner. The numerical detail includes the five-number summary of each dimension and the mean and the standard deviation. The numerical score values are also shown at the third column whose background is color-coded using the same color-mapping as in the score overview. While numerical summaries of distributions are very useful, sometimes they are misleading. For example, when there are two peaks in a distribution, neither the median nor the mean explains the center of the distribution. This is one of the cases for which a graphical representation of a distribution (e.g., a

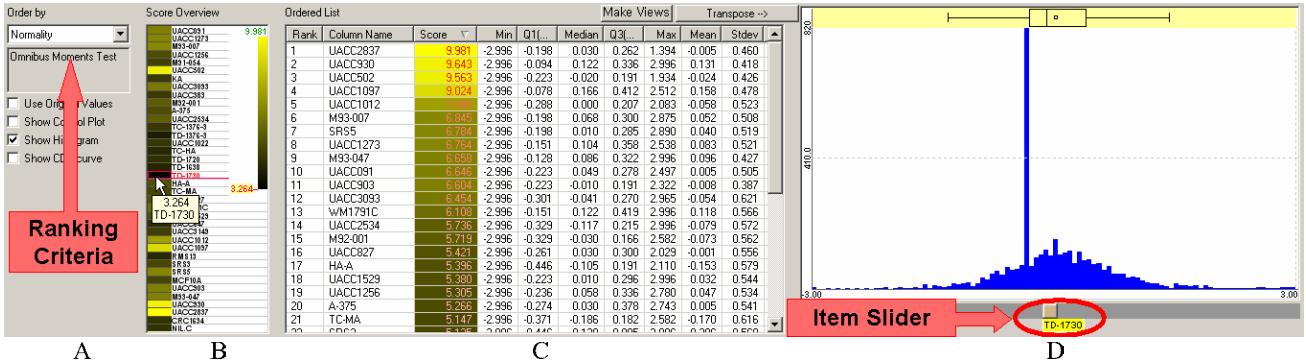


Figure 2. Rank-by-feature framework interface for histograms (1D) All 1D histograms are ordered according to the current order criterion (A) in the ordered list (C). The score overview (B) shows an overview of scores of all histograms. Mouseover event activates a cell in the score overview, highlights the corresponding item in the ordered list (C) and shows the corresponding histogram in the histogram browser (D) simultaneously. A double click on a cell enlarges the cell and activates it until another double click event occurs. A selected histogram is shown in the histogram browser (D), where users can easily traverse histogram space by changing the dimension for the histogram using item slider. Boxplot is also shown above the histogram to show the graphical summary of the distribution of the dimension. (A gene expression data set for melanoma study (3614 genes x 38 samples)).

histogram) works better. In the histogram browser, users can see the visual representation of the distribution of a dimension at a time. A boxplot is a good graphical representation of the five-number summary, which together with a histogram provides an informative visual description of a dimension's distribution. It is possible to interactively change the dimension in focus just by dragging the item slider attached to the bottom of the histogram.

Since different users might be interested in different features in the data sets, it is desirable to allow users to customize the available set of ranking criteria. However, we have chosen the following four ranking criteria that are fundamental and common for histograms as a starting point, and we have implemented them in HCE:

(1) Normality of the distribution (0 to *inf*)
(2) Uniformity of the distribution (0 to *inf*)
(3) The number of potential outliers (0 to n)
(4) The number of unique values (0 to n)

There are at least ten statistical tests for normality since a distribution can be nonnormal due to many different reasons. We used the *omnibus moments test* for normality in the current implementation. For the uniformity test, we used an information-based measure called *entropy*. Given k bins in a histogram, the entropy of a histogram H is $entropy(H) = -\sum_{i=1}^{k} p_i \log_2^{p_i}$, where p_i is the probability that an item belongs to the i-th bin. High entropy means that values of the dimension are from a uniform distribution and the histogram for the dimension tends to be flat. To count outliers in a distribution, we used the 1.5**IQR* (Interquartile range: the difference between the first quartile (*Q1*) and the third quartile (*Q3*)) criterion that is the basis of a rule of thumb in statistics for identifying suspected outliers [1]. An item of value d is considered as a suspected outlier if $d >$ (*Q3*+1.5**IQR*) or $d <$ (*Q1*-1.5**IQR*). It is also possible to use an outlier detection algorithm developed in the data mining or database area for counting outliers.

For some of the ranking criteria for histogram ordering such as normality, there are many available statistical tests to choose from. We envision that many researchers could contribute statistical tests that could be easily incorporated into the rank-by-feature framework. For example, since outlier detection is a rich research area, novel statistical tests are likely to be proposed in the coming years, and they could be made available as plug-ins.

3.2 2D Scatterplot Ordering

According to the basic principles for a better exploration of an unsupervised multidimensional data, after scrutinizing 1D projections, it is natural to move on to 2D projections where pair-wise relationships will be identified. Relationships between two dimensions (or variables) are best visualized in a scatterplot. The values of one dimension are aligned on the horizontal axis, and the values of the other dimension are aligned on the vertical axis. Each data item in the data set is shown as a point in the scatterplot whose position is determined by the values at the two dimensions. A scatterplot graphically reveals the form (e.g., linear or curved), direction (e.g., positive or negative), and strength (e.g., weak or strong) of relationships between two dimensions. It is also easy to identify outlying items in a scatterplot.

We used scatterplots as the main display for the rank-by-feature framework for 2D projections. Figure 3 shows the interactive interface design for the rank-by-feature framework for 2D projections. Analogous to the interface for 1D projections, the interface consists of four coordinated components: the *control panel*, the *score overview*, the *ordered list*, and the *scatterplot browser*. Users select an ordering criterion in the control panel on the left, and then they see the complete ordering of all possible 2D projections according to the selected ordering criterion (Figure 3). The ordered list shows the result of ordering sorted by the ranking (or scores) with scores color-coded on the background. Users can click on any column to sort the list by the column. Users can easily find the most or least interesting scatterplots by changing the sort order to ascending or descending order of score (or rank). It is also easy to examine the scores of all scatterplots with a certain variable for horizontal or vertical axis after sorting the list according to X or Y column by clicking the corresponding column header.

However, users cannot see the overview of entire relationships between variables at a glance in the ordered list. Overviews are important because they can show the whole distributions and reveal interesting parts of data. We have implemented a new

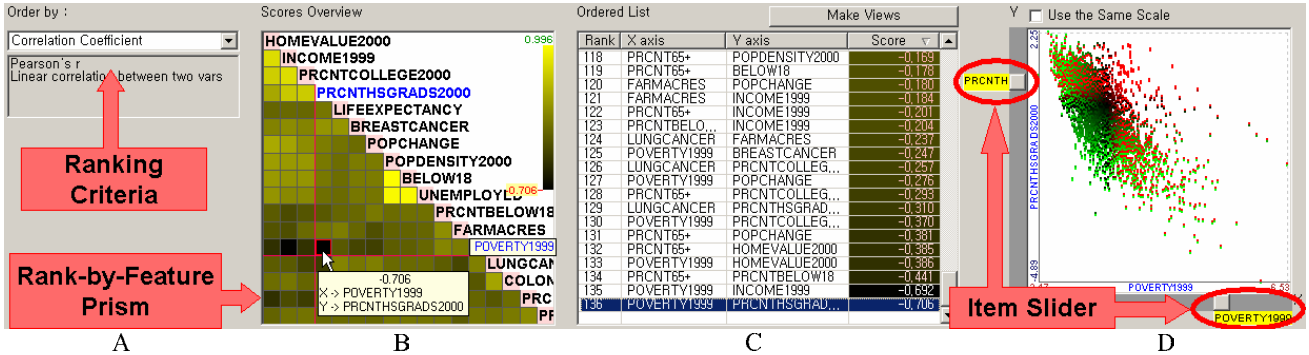


Figure 3. Rank-by-feature framework interface for scatterplots (2D) All 2D scatterplots are ordered according to the current ordering criterion (A) in the ordered list (C). Users can select multiple scatterplots at the same time and generate separate scatterplot windows for them to compare them in a screen. The score overview, known as the *rank-by-feature prism* (B) shows an overview of scores of all scatterplots. Mouseover event activates a cell in the score overview, highlights the corresponding item in the ordered list (C) and shows the corresponding scatterplot in the scatterplot browser (D) simultaneously. A double click on a cell enlarges the cell and activates it until another double click event occurs. A selected scatterplot is shown in the scatterplot browser (D), where it is also easy to traverse scatterplot space by changing X or Y axis using item sliders on the horizontal or vertical axis. (A demographic and health related statistics for 3138 U.S. counties with 17 attributes.)

version of the score overview for 2D projections that we call the *rank-by-feature prism*. This name was inspired by the use of color to show differences as well as the triangular shape. The rank-by-feature prism is an *m*-by-*m* tabular view where all dimensions are aligned in the rows and columns. Each cell of the rank-by-feature prism represents a scatterplot whose horizontal and vertical axes are dimensions at the corresponding column and row respectively. Since this table is symmetric, we used only the lower-triangular part for showing scores and the diagonal cells for showing the dimension names as shown in Figure 3. Each cell is color-coded by its score. The color mapping is shown at the top right corner of the table view. As users move the mouse over a cell, the scatterplot corresponding to the cell is shown in the scatterplot browser simultaneously, and the corresponding item is highlighted in the ordered list (Figure 3). The rank-by-feature prism, the ordered list, and the scatterplot browser are interactively coordinated according to the change of the dimension in focus. In other words, a change of dimension in focus in one of the three components leads to the instantaneous change of dimension in focus in the other two components.

In the rank-by-feature prism, users can *preattentively* detect the most/least interesting combination of dimensions thanks to the linear color-coding scheme and the intuitive tabular display. Sometimes, users can also easily find a dimension that is the least or most correlated to most of dimensions by just locating a whole row or column where all cells are the mostly dark or bright. It is also possible to find an outlying scatterplot whose cell has distinctive color intensity compared to the rest of the same row or column. After locating an interesting cell, users can double click on the cell to select, and enlarge it, and then they can scrutinize it on the scatterplot browser and on other tightly coordinated views in HCE.

While the ordered list shows the numerical score values of relationships between two dimensions, the interactive scatterplot browser best displays the relationship graphically. In the scatterplot browser, users can quickly take a look at scatterplots by using item sliders attached to the scatterplot view. Simply by dragging the vertical or horizontal item slider bar, users can change the dimension for the horizontal or vertical axis. With the current version implemented in HCE, users can investigate multiple scatterplots at the same time. They select more than one scatterplots in the ordered list by clicking them with the control key pressed. Then, click "Make Views" button on the top of the ordered list, and each selected scatterplot will be shown in a separate child window. Users can select a group of items by dragging a rubber rectangle over a scatterplot, and the items within the rubber rectangle will be highlighted in all other scatterplots. On some scatterplots they might gather tightly together, while on other scatterplots they scatter around.

Again interesting ranking criteria might be different from user to user, or from application to application. Initially, we have chosen the following five ranking criteria that are fundamental and common for scatterplots, and we have implemented them in HCE:

(1) Correlation coefficient (-1 to +1)
(2) Least square error for simple linear regression (0 to 1)
(3) Least square error for curvilinear regression (0 to 1)
(4) The number of items in the region of interest (0 to *n*)
(5) Uniformity of scatterplots (0 to *inf*)

The first three criteria are useful to reveal statistical (linear or curved) relationships between two dimensions (or variables), and the next two are useful to find scatterplots of interesting distributions. For the first criterion, we use Pearson's correlation coefficient (r) for a scatterplot (S) with n points defined as

$$r(S) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

Pearson's r is a number between -1 and 1. The sign tells us direction of the relationship and the magnitude tells us the strength of the linear relationship. The magnitude of r increases as the points lie closer to the straight line. The second criterion is closely connected to the first criterion. r^2 is a measure of how successful the regression was in explaining the relationship between two dimensions. These linear relationships are very common and practically important for describing 2D data. The third criterion is to sort scatterplots in terms of least-square errors from the optimal quadratic curve fit so that users can easily isolate ones where all points are closely/loosely arranged along a quadratic curve. The fourth criterion is the most interactive since it requires users to specify a (rectangular, elliptical, or free-formed) region of interest, and then uses the number of items in the region to order scatterplots so that users can easily find ones with most/least number of items in the given region. For the last ordering criterion, we calculate the entropy in the same way as we did for histograms, but this time we divide a two dimensional space into regular grid cells and then use each cell as a bin. For example, if we have generated *k*-by-*k* grid, the entropy of a scatterplot S is

$$entropy(S) = -\sum_{i=1}^{k}\sum_{j=1}^{k} p_{ij} \log_2^{p_{ij}}$$, where p_{ij} is the probability that an item belongs to the cell at (i, j) of the grid.

4 Application Example

We show an application example of the rank-by-feature framework with a collection of county information data set. The data set has 3139 rows (U.S. counties) and 17 columns (attributes). 17 attributes are the following:

1. population, 2000
2. population percent change, 4/1/2000-7/1/2001
3. person under 18 years old, 2000
4. percent under 18 years old, 2000
5. population 65 years old and over, 2000
6. percent of female persons, 2000
7. percent of college graduates or higher age 25+, 2000
8. percent of high school graduates age 25+, 2000
9. median value of owner-occupied housing value, 2000
10. per capita money income, 1999
11. percent below poverty level, 1999
12. person unemployed, 1999
13. farm land (acres), 1997
14. breast cancer per 100,000 white female, 19970-1994
15. colon cancer rate per 100,000, 1997
16. lung cancer mortality rate per 100,000, 1997
17. life expectancy, 1997

We first select the "Uniformity" for 1D ranking, and we can preattentively identify the three dimensions such as "population," "percent under 18 years old," and "person unemployed" that have very low scores in the score overview as shown in Figure 4. This means the distribution of values of these dimensions is very

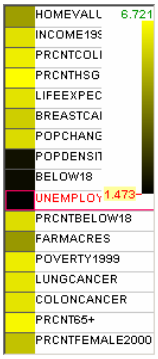


Figure 4. The score overview for U.S. county data

biased to a certain small range as shown in Figure 5d. The county of the extreme value (highlighted in red at the right most bin of the histogram) on all three low-scored dimensions is "Los Angeles, CA." In the histogram for "percent of high school graduates" that has a high score (Figure 5a), LA is mapped to a bin below the first quartile on the histogram (also highlighted in red), which means there are relatively lower percentage of high school graduates in LA.

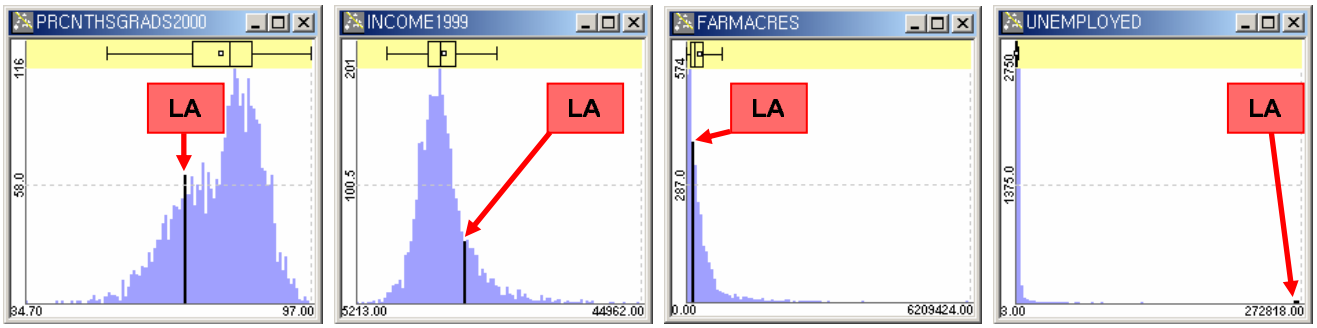
Then, we move on the rank-by-feature framework for 2D projections, and choose "Correlation coefficient" as the current ranking criterion. And again we preattentively identify three very bright yellow cells and two very dark cells in the rank-by-feature prism ((B) in Figure 3). The scatterplot for one of the high-scored cells is shown in Figure 6a, where LA is highlighted with orange triangle in a circle at the top right corner. Interestingly, the three bright cells are composed by the three dimensions that have very low scores in 1D ranking by "Uniformity." LA is also a distinctive outlier in all three high scored scatterplots and the scatterplot in Figure 6b. The scatterplot for one of the two dark cells is shown in Figure 6d. From the scatterplot, we can confirm a trivial relationship between poverty and income, i.e. poor counties have less income.

The rank-by-feature framework is to HCE users what maps are to the explorer of unknown areas. It helps users get some useful idea about better direction for the next step of their exploratory analysis of a multidimensional data set. The rank-by-feature framework in HCE 3.0 can handle much larger data sets with many more dimensions than this application example. More columns with environment statistics and hospital statistics data will be added to this example data set in the future to understand interesting relationships among various attributes across many different kinds of knowledge domains.

5 Discussion

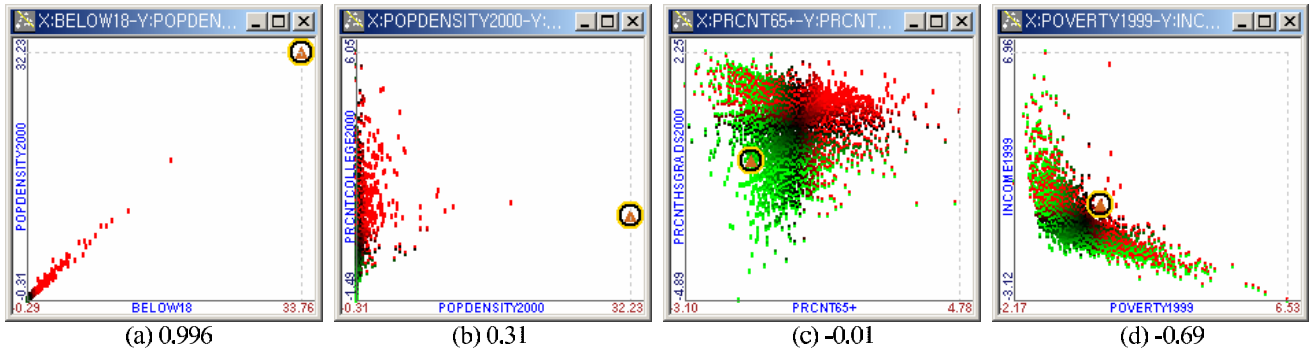
In spite of their limitations, low-dimensional projections are useful tools for users to understand multidimensional data sets. Since 3D projections have the problem of the cognitive burden brought in by the navigation controls, we decided to concentrate on 1D and 2D projections. Since the axis parallel projections are much more easily interpreted by users compared to the arbitrary 1D or 2D projection, users usually begin their exploratory analysis with analyzing axis-parallel 1D and 2D projections.

We introduced a new framework called the rank-by-feature framework for effective exploration of these axis-parallel projections. Interactive interfaces for the rank-by-feature framework were designed for 1D and 2D projections. There are four coordinated components in each interface: the control panel, the score overview, the ordered list, and the histogram/scatterplot



(a) 6.7 (b) 6.1 (c) 4.5 (d) 1.5

Figure 5. Four selected histograms ranging from high uniformity (5a) to low uniformity (5d). The bar for LA is highlighted in black in each figure. In 5d the distribution is concentrated on the far left and LA appears as an outlier at the far right.



(a) 0.996 (b) 0.31 (c) -0.01 (d) -0.69

Figure 6. Four selected scatterplots ordered by correlation coefficient (The point for LA is highlighted with a circle.)

browser. Users can choose a ranking criterion at the control panel, and they can examine the ranked result using the remaining three components that interactively coordinate with each other. Among the three components, the score overview enables users to preattentively detect the projection with the highest/lowest score due to the intuitive layout and linear color-mapping, and it also helps users to grasp the overall pattern of the score distribution and some distinctive outliers. While the ordered list provides users with the numerical detail of each projection, the browser enables users to interactively examine the graphical representation of a projection (the combination of histogram and boxplot for a 1D projection, and scatterplot for a 2D projection). The item slider attached to histogram/scatterplot display facilitates the exploration by allowing the interactive change of the dimension in focus.

Even though we have implemented only four or five ranking criteria in the current version of HCE, there are definitely many other useful ranking criteria: e.g., skewness, kurtosis, the number of clusters, the strength of the outlyingness of outliers, the size of gaps or holes, etc. When implementing a new ranking criterion in the rank-by-feature framework, we should take into account the time complexity of the score function of the ranking criterion. For example, if we have n data items in m-dimensional space, then we need to calculate the score of a 2D projection $m*(m-1)/2$ times. If the time complexity of the score function is $O(n)$, the total time complexity will be $O(nm^2)$. To achieve a reasonable response time, it is necessary to have an efficient routine for computing scores for a ranking criterion. The following table shows the amount of time (in seconds) to complete 2D rankings for four data sets of various sizes (# of items by # of dimensions) with our current implementation on a Intel Pentium 4.(2.53GHz, 1GB memory) PC.

criterion / size	correlation	linear regression	curvilinear regression	uniformity
3138 x 17	.05	.1	.2	.2
3614 x 38	.1	.7	.8	1.6
11704 x 105	2.6	14.2	17.4	38.6
22283 x 105	4.9	29.1	33.1	72.5

When there are so many dimensions, the rank-by-feature prism will be so crowded that it becomes difficult to know which cell is for which dimensions. As a future work, thus, a filtering or grouping mechanism would be necessary. We can attach a range slider to the right side of the score overview that will control the upper and lower bound of scores. If the score of a cell doesn't satisfy the thresholds, the cell will be grayed out. If an entire row or column is grayed out, the row or column can be filtered out so that remaining rows and columns will occupy more screen space. We can also utilize the dimension clustering result that is already available in HCE to rank clusters of dimensions instead of individual dimensions.

6 Conclusion

We introduced the rank-by-feature framework to help users systematically traverse the axis-parallel 1D and 2D projections of multidimensional data sets. By presenting projections in an ordered manner by a ranking criterion interactively chosen by users, users can effectively examine the projections of multidimensional data. The rank-by-feature prism provides a visual overview that helps users identify extreme values of criteria such as correlation coefficients or uniformity measures. Information visualization principles and techniques such as dynamic query by item sliders, combined with traditional graphical displays like histograms, boxplots, and scatterplots played a major role in the rank-by-feature framework. As future work, various statistical tools and data mining algorithms can be incorporated into our rank-by-feature framework as new ranking criteria, and more graphical statistics plots can also be incorporated into the framework to present useful visual representations of the data set in low-dimensional projections.

Acknowledgement: This work was supported by N01 NS-1-2339 from the NIH and by the National Science Foundation under Grant No. EIA 0129978.

References

[1] D. S. Moore and G. P. McCabe, *Introduction to the Practice of Statistics*, W.H. Freeman and Company, New York, NY, 3rd ed., 1999.
[2] T. Kohonen, *Self-Organizing Maps*, 3rd ed., Springer, New York, 2000.
[3] J. H. Friedman, "Exploratory Projection Pursuit," *J. Am. Statistical Assoc.*, Vol. 82, No. 397, pp. 249-266, 1987.
[4] D. Asimov, "The Grand Tour: a Tool for Viewing Multidimensional Data," *The SIAM Journal of Scientific and Statistical Computing*, Vol. 6, No. 1, pp. 128-143, 1985.
[5] P. J. Huber, "Projection Pursuit," *The Annals of Statistics*, Vol. 13, No. 2, pp. 435-475, 1985.
[6] D. R. Cook, A. Buja, J. Cabtea, and H. Hurley, "Grand Tour and Projection Pursuit", *Journal of Computational and Graphical Statistics*, Vol. 23, pp. 225-250, 1995.
[7] M. O. Ward, "XmdvTool: Integrating Multiple Methods for Visualizing Multivariate Data," *Proc. IEEE Visualization '94*, pp. 326-336, 1994.
[8] D. Guo, "Coordinating Computational and Visual Approaches for Interactive Feature Selection and Multivariate Clustering," *Information Visualization*, Vol. 2, pp. 232-246, 2003.
[9] Spotfire DecisionSite, Spotfire., http://www.spotfire.com/
[10] H. Liu, H. Motoda, *Feature Selection for Knowledge Discovery and Data Mining*, Kluwer Academic Publishers, Boston, 1998.
[11] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications," *Proc. SIGMOD'98*, pp. 94-105, 1998.
[12] C. C. Aggarwal, C. Procopiuc, J. Wolf, P. Yu, and J.-S. Park, "Fast Algorithms for Projected Clustering," *Proc. SIGMOD'99*, pp. 61-72, 1999.
[13] M. Ankerst, S. Berchtold, and D.A. Keim, "Similarity Clustering of Dimensions for an Enhanced Visualization of Multidimensional Data," *Proc. Int'l Symp. Information Visualization*, pp. 52-60, 1998.
[14] M. Friendly, Corrgrams, "Exploratory Displays for Correlation Matrices," *The American Statistician*, Vol. 19, pp. 316-325, 2002.
[15] A. Inselberg and B. Dimsdale, "Parallel Coordinates: A Tool for Visualizing Multidimensional Geometry," *Proc. Int'l Symp. Information Visualization*, pp. 361-375, 1990.
[16] J. Seo and B. Shneiderman, "Interactively Exploring Hierarchical Clustering Results," *IEEE Computer*, Vol. 35, No. 7, pp. 80-86, 2002.
[17] J. Yang, W. Peng, M. O. Ward, and E. A. Rundensteiner, "Interactive Hierarchical Dimension Ordering, Spacing and Filtering for Exploration of High Dimensional Datasets", *Proc. Int'l Symp. Information Visualization*, pp 105-112, October 2003.