Exploratory Data Analysis with Categorical Variables: An Improved Rank-by-Feature Framework and a Case Study

Jinwook Seo and Heather Gordish-Dressman

{jseo, hgordish}@cnmcresearch.org

Research Center for Genetic Medicine

Children's Research Institute

111 Michigan Ave NW, Washington, DC 20010

RUNNING HEAD: RANK-BY-FEATURE FRAMEWORK FOR CATEGORICAL DATA

Acknowlegement: This work was supported by NIH 5R24HD050846-02 Integrated molecular core for rehabilitation medicine, and NIH 1P30HD40677-01 (MRDDRC Genetics Core). We also thank the FMS study group, especially Joseph Devaney and Eric Hoffman, for providing genotype data.

Corresponding Author's Contact Information:

Jinwook Seo

Research Center for Genetic Medicine

Children's Research Institute

111 Michigan Ave NW, Washington, DC 20010

Tel: +1 202-884-4942 Fax: +1 202-884-6014

ABSTRACT

Multidimensional datasets often include categorical information. When most dimensions have categorical information, clustering the dataset as a whole can reveal interesting patterns in the dataset. However, the categorical information is often more useful as a way to partition the dataset: gene expression data for healthy vs. diseased samples or stock performance for common, preferred, or convertible shares. We present novel ways to utilize categorical information in exploratory data analysis by enhancing the rank-by-feature framework. First, we present ranking criteria for categorical variables and ways to improve the score overview. Second, we present a novel way to utilize the categorical information together with clustering algorithms. Users can partition the dataset according to categorical information vertically or horizontally, and the clustering result for each partition can serve as new categorical information. We report the results of a longitudinal case study with a biomedical research team, including insights gained and potential future work.

Color figures are available at www.cs.umd.edu/hcil/ben60

1 INTRODUCTION

In many analytic domains, multidimensional datasets frequently include categorical information that plays an important role in the data analysis. In our work in bioinformatics, many of the biologists we collaborate with have datasets that include categorical information, such as labels for healthy vs. diseased samples. In that case, the goal is to compare gene expression levels (quantitative measurements of gene activity) to determine which genes might have higher or lower expression levels in the diseased samples as compared to the healthy samples. Other biology researchers compare male and female patients because some genes are differentially expressed in one gender but not in the other. We have received similar requirements from stock market analysts, meteorologists, and others.

The inclusion of such categorical information in a multidimensional dataset imposes a different challenge to the way researchers analyze the dataset. First, different test statistics are necessary for the dataset. For example, a chi-square test is typically used for testing an association between categorical variables while a linear correlation coefficient is typically used for testing an association between real (continuous) variables. Secondly, stratification by categorical information is crucial to delve into such datasets, without which features hidden in the stratified subgroups cannot be identified during exploratory data analysis. Ignoring or mistreating such information could result in a flawed conclusion costing days or even months of effort.

Most statistical packages support functionalities for categorical data, but biologists and biostatisticians are in need of exploratory visualization tools with which they can interactively examine their large multidimensional datasets containing categorical information. To address

these issues and requests, we developed new features in our order-based data exploration framework, or *rank-by-feature* framework (Seo & Shneiderman, 2005b) in the Hierarchical Clustering Explorer (HCE, www.cs.umd.edu/hcil/hce/, also see Section 2), which enabled users to effectively explore multidimensional datasets containing categorical entities or variables.

First, we added to the rank-by-feature framework new ranking criteria for categorical or categorized variables in multidimensional datasets. The score overview (see Section 2) that gives users a brief overview of the ranking result was also improved by introducing size-coding by strength in addition to the color coding. The inclusion of strength information from a ranking criterion for categorical data can make the score overview of the rank-by-feature framework more informative at a glance.

Second, we enabled users to stratify their datasets according to the categorical information in the datasets. We support two different partitioning mechanisms according to the direction they split the datasets: *vertical* and *horizontal* partitioning. In this paper, we assume that the input dataset is organized in a tabular way such that the rows represent items or entities and the columns represent dimensions or attributes. Users can stratify their datasets vertically by separating columns according to the categorical information conveyed by a special row and then conduct comparisons among items in different partitions. We call this *vertical partitioning*. For example, biologists are often interested in partitioning samples according to their phenotypes (normal vs. diseased) in case-control microarray projects. Clustering of the rows is then performed in each partition to generate two clustering results of the rows, each of which is homogeneous (i.e. only includes the same value for the special categorical row). By comparing the partitioned clustering results, users can get meaningful insights into finding an

interesting group of genes that are differentially or similarly expressed in the normal and the diseased groups.

Users can also stratify their datasets horizontally by separating rows according to a column of a categorical attribute such as gender and ethnicity. We call this *horizontal partitioning*, where the rows of the dataset are partitioned and each partition has the same set of columns. For example, in genome-wide association studies where the study subjects are in the rows and the genotype and the phenotype information are in the columns, it is often inevitable to partition the subjects (or the rows) according to the gender or the ethnicity column before performing any statistical tests to the dataset. Without such a partitioning, associations hidden in a subgroup cannot be revealed. Similar to vertical partitioning, clustering of columns can also be done in each partition to compare the clustering results.

We evaluated these new features added to the rank-by-feature framework through a longitudinal case study with a biomedical research team. While comparing clustering results of partitions can still provide insights into the dataset, it turns out from our longitudinal case study that researchers are also interested in using categorical information in more dynamic ways. First, they want to stratify datasets both vertically and horizontally in evaluating ranking criteria in the rank-by-feature framework. Second, they also want to compare multiple score overviews for the levels of a given categorical variable. In response to users' interests, we allow flexibility in users' stratification tasks. Users can assign variables to two different groups to study relationships between the variables in the two groups. In the score overview, variables in one group are arranged along the x-axis and variables in the other groups are arranged along the y-axis. Users can also have one score overview for each level of the categorical variable, for example, one for male and one for female if partitioned by the gender

variable. Interactive coordination between score overviews for the levels will highlight the difference between levels. These improvements in the rank-by-feature framework enable users to identify important patterns hidden in the data, which may have been missed without the partitioning.

In the following section, we first summarize our previous work on HCE and the rank-by-feature framework upon which this paper is based. After presenting related work in the next section, we explain the ranking criteria for categorical information and the improved score overview where cells are appropriately size-coded. We then explain two partitioning methods and a new ranking criterion ("Cluster Similarity") to compare clustering results with partitions. Next, a longitudinal case study with a biomedical research team is summarized and discussed in detail. Lastly, we present the conclusion and future work.

2 HISTORY OF HCE AND RANK-BY-FEATURE FRAMEWORK

This section summarizes HCE and the rank-by-feature framework to give readers some background for the work presented in this paper. HCE was, at the beginning, designed to help a group of researchers at the NIH with examining and understanding hierarchical agglomerative clustering results, which are presented in a form of a binary tree, or *dendrogram*. We later discovered that this problem was prevalent among microarray researchers around the globe. Two dynamic query controls were conceived in HCE to help users better understand hierarchical clustering results (Figure 1). The first control is the "Minimum Similarity Bar" which users can drag to change the similarity threshold so that only the subtrees satisfying the threshold show. Users can interactively determine the most reasonable separation of clusters. The second control is the "Detail Cutoff Bar" which users can drag to manage the level of detail. The subtrees (or clusters) below the detail cutoff bar

are rendered using their averages. It helps users see the global pattern of the clustering results at a glance.

<<insert Figure 1 about here>>

Then we added scatterplots to HCE. Users can select two variables, one for the x-axis and the other for the y-axis, to see their relationships in a scatterplot which was coordinated with the clustering result view. Users can select a group of items in any view and they are highlighted in other views. As scatterplots were integrated into HCE, it became clear that scatterplots with two variables could be one of the simplest and most comprehensive means to visually examine large multidimensional datasets. The problem, however, is that a large multidimensional dataset has too many potential scatterplots.. A dataset with n variables has $n(n-1)/2$ possible 2D scatterplots. The rank-by-feature framework was a solution to the problem. Instead of looking at scatterplots in a random order, the rank-by-feature framework encouraged users to set a goal first and then look at the scatterplots in an orderly way.

The statistician John Tukey envisioned a similar approach called “Scagnostics” (Tukey & Tukey, 1985) about 20 years ago, in which he suggested ranking scatterplots according to some statistical criteria. However, the rank-by-feature framework (Figure 2) was the first to bring the idea into reality with an open-ended framework, where user interface designs and information visualization techniques played an important role. The framework was also extended to cover histograms (1D) as well as scatterplots (2D). It consists of four GUI components: a combo box for ranking criterion selection, a score overview for 1D/2D ranking results overview, an ordered list control for the same ranking results display, and a histogram and scatterplot browser (the projection browser).

Using this framework, users can explore multidimensional datasets by ranking 1D and 2D projections (or histograms and scatterplots) according to user-selected criteria, such as correlation coefficient, normality, uniformity, outlierness, least squares errors, etc. Novel statistical ranking criteria had to be created, such as “quadracity” to find positive or negative quadratic relationships. The results of these rankings are shown not only in a list control (score list) but also in a color-coded lower triangular matrix (score overview), which provides a natural space for user-controlled exploration. Users simply move their cursors over the bright yellow or black squares and the scatterplots or histograms appear within 100msec in the adjoining panel (projection browser). Many users find this a compelling and rapid way to explore their data. After a session reviewing data with collaborators, one commented that “what you did in an hour would have taken us a month.”

<<insert Figure 2 about here>>

The rank-by-feature framework was later generalized to become a set of principles called the GRID (Graphics, Ranking, and Insight for Discovery) principles for interactive exploration of multidimensional datasets (Seo & Shneiderman, 2005a). GRID principles extend Moore and McCabe’s principles (Moore & McCabe, 1999) for statistical exploratory data analysis to take in aspects of user interaction and information visualization. These principles can help users clarify their goals and examine multidimensional datasets using order-based exploration strategies instead of opportunistic ways. GRID principles are summarized as follows: “1) study 1D, study 2D, then find features and 2) ranking guides insight, statistics confirm.” A survey and three case studies with users in biology, statistics, and meteorology showed the efficacy of the GRID principles and the rank-by-feature

framework (Seo & Shneiderman, 2005a). Note that the original rank-by-feature framework provided ranking criteria only for real variables.

The HCE software has also contributed to muscular dystrophy research, identifying novel genes that are likely to participate in the progression of the disease. This work and other HCE developments have been done in close collaboration with researchers at the Research Center for Genetic Medicine, Children's National Medical Center. Biomedical researchers have found HCE to be a valuable tool in their research, and has led to four papers in refereed bioinformatics journals (Seo et al., 2004; Seo, Gordish-Dressman, & Hoffman, 2006; Seo & Hoffman, 2006; Zhao et al., 2003).

3 RELATED WORK

Most visualization tools use categorical variables to label displays distinctively according to the categorical information by using different sizes, colors, or shapes. Friendly suggested several visualization techniques and graphical displays for categorical data (Friendly, 2000), which include Fourfold display, Mosaic displays, and Association plots. There is a book on the visualization of categorical multivariate data especially in the social sciences (Blasius & Greenacre, 1998), which presents models for categorical data, visual representations and modeling processes. Rosario et al. proposed a pre-processing approach to “assign order and spacing” among categorical variables for displaying them in general purpose visualization tools (Rosario, Rundensteiner, Brown, Ward, & Huang, 2004).

While there has been a large number of clustering algorithms for numerical data, there has been much less work on clustering categorical data. Similarity between two items with categorical attributes should be calculated differently from that between items with only real

attributes. Typically, co-occurrence measures and link analyses are used to build a graph structure from a categorical dataset, and then a graph partitioning algorithm or a traditional clustering algorithm generates clusters (Gibson, Kleinberg, & Raghavan, 2000; Guha, Rastogi, & Shim, 1999).

Dimension selection, partitioning and management for multidimensional/multivariate datasets are related to our work. Selection and partitioning of the attributes (or variables) in a multidimensional/multivariate dataset (Elomaa & Rousu, 1999; Guo, Gahegan, Peuquet, & MacEachren, 2003; Liu & Motoda, 1998) have been studied in the KDD and machine learning research fields to improve the algorithm performance by focusing on a smaller but more informative group or by using each partition for a different purpose during the learning/discovery process. Yang et al. proposed and implemented an interactive hierarchical approach for ordering, spacing and filtering dimensions (or attributes) by hierarchically or manually clustering them in order to reduce clutter and support multi-resolution display (Yang, Peng, Ward, & Rundensteiner, 2003).

There is some related work that seeks to compare hierarchical structures, such as the Tree Juxtaposer (Munzner, Guimbretiere, Tasiran, Zhang, & Zhou, 2003) that highlights differing items and subtrees between two versions of a tree. The goal of showing relationships between two different hierarchies such as a geographical hierarchy and a jobs hierarchy was supported by coordinated views in PairTrees (Kules, Shneiderman, & Plaisant, 2003). Users could select a node in the geographic hierarchy such as a state in the U.S., and that would produce highlights in the jobs hierarchy to identify which jobs were held by residents of the selected state. Similarly, if a job node were selected, that would produce a highlight in the geographic hierarchy to identify where those jobs were most frequent.

Adjacency matrix representations, such as the MatrixExplorer (Henry & Fekete, 2006) and the Matrix Browser (Ziegler, Kunz, & Botsch, 2002), show relationships between items, typically link relationships between nodes in a graph. These adjacency matrices are of order *n* x *n* for an *n* node graph. Adjacency matrices for bi-partite graphs with *n* nodes in one partition and *m* nodes in the second partition are close to what we are using in this work. Selections from two hierarchies were also shown in Matrix Zoom (Abello & van Ham, 2004) which has similarities to our work. However, our emphasis is to enable users to compare clustering results to identify items that are noticeably different in performance across partitions.

Another source of related work is on reorderable or permutation matrices (Siirtola & Makinen, 2005), which are often referred to as heatmaps in commercial systems such as Spotfire (Spotfire). Kincaid suggested an extended permutation matrix for microarray studies (Kincaid, 2004), where cells were size-coded as well as color-coded by fold change and users could sort the rows according to similarity measures and/or supplemental annotations. Our use of heatmaps is tied to the clusters in two dendrograms, and the similarity index we use represents features that are of interest to users seeking to identify items that are noticeably different in performance across partitions. Moreover, our extended heatmaps encode two different quantities (significance & strength) into size and color instead of using one value for both.

A further distinction in our work in terms of clustering results comparison is the two levels of analysis. We start by trying to match clusters, and then drill down to identify the items that account for the similarity. The capacity to see the clusters and select individual items rapidly enables exploration of datasets with thousands of items. Also the capacity to see where items from a cluster in one partition fall in the other partition reveals differences across

partitions. For example, by selecting a cluster with high gene expression levels for healthy patients, users can determine if some of these genes have lower expression levels in diseased patients.

In terms of evaluation methods, one of the most popular methods in the HCI community is a controlled user study, where subjects are asked to perform small specific tasks in a controlled laboratory environment within a relatively short time. However, the uncontrollable nature of human subjects and other independent variables makes controlled user studies subject to unexpected bias and variance. As an alternative way to evaluate advanced information visualization tools, longitudinal case studies or field studies have been performed with subjects in their real work environments (Gonzales & Kobsa, 2003; Saraiya et al., 2006; Seo & Shneiderman, 2006). Although a case study result might not be generalizable to other users in different situations, success stories and problems reported in a case study could benefit other designers and potential users by providing valuable insights gained during the case study.

4 RANKING WITH CATEGORICAL VARIABLES

In this section, we present new ranking criteria to evaluate relationships between variables when at least one variable is categorical. In addition, we present a way to improve the score overview for those criteria by size-coding as well as by color-coding cells in the score overview. Variables in multidimensional datasets are usually distinguished into two categories: *categorical* and *quantitative* (or *real*):

- Categorical variables can be further defined as *nominal* or *ordinal* variables. Their values are elements of a bounded discrete set. For example, ‘type of songs’ is a

nominal categorical variable since all possible values can be drawn from an enumerated set, {rock, jazz, pop, hip-hop, R&B, classical, others}, but that set has no valued order, i.e. rock does not have a greater value than jazz. An *ordinal* categorical variable also has values that can be drawn from an enumerated set, but the values in that set are ordered. If the set has only two elements, those variables are called *binary*.

- Quantitative variables are continuous as they can take on any of a number of values, i.e. age can take on any values from a large range, and there are more specific distinctions in the continuous variables. They also support mathematical operations such as addition.

Previously, HCE only dealt with quantitative variables. Categorical variables require different treatments. If we can encode categorical values as integer values and treat them as quantitative values (for example, rock=1, jazz=2 and so on), then we can calculate for example the Pearson correlation coefficient between a continuous variable and a categorical variable. But the result is meaningless because the Pearson correlation coefficient measure is applicable only to quantitative variable pairs.

When we examine relationships between a pair of variables (categorical or quantitative), it is better to start with more general relationships than correlation coefficients. The correlation coefficient is only one of many possible associations between variables. One of the most common statistical methods to measure associations between two categorical variables is the chi-square test. Any non-categorical variable can be transformed to a categorical variable by binning or quantizing the values. Thus, it is possible to measure associations between

categorical and quantitative variables. Since the term "association" means dependency in statistics, the chi-square statistic is a measure of dependency between two variables.

Let's assume that we measure an association between two variables, x and y. x has n bins (or categories) $\{xb_i | i=1..n\}$ and y has m bins (or categories) $\{yb_j | j=1..m\}$. Then the chi-square statistic is calculated as follows:

$\chi^2 = \sum_{i=1}^{n}\sum_{j=1}^{m}\frac{(O_{ij} - E_{ij})^2}{E_{ij}}$, where O_{ij} is the observed frequency of a value belonging to both xb_i and yb_j, and E_{ij} is the expected frequency of a value belonging to both xb_i and yb_j.

Like other statistics, the chi-square statistic also returns a p-value which represents the significance of an association. Smaller p-values mean greater significance. While the chi-square statistic and p-value confirms that there is some association, the nature or the strength of association is not revealed by the test statistic. The nature of association can be identified by investigating visual displays or through other ranking criteria such as "number of items in ROI." There are several methods to evaluate the strength of associations. The mutual information measure from information theory is a good candidate, or other statistics like *Cramer's V* and *Contingency coefficient C* can be used (Press, Flannery, Teukolsky, & Vetterling, 1992).

The score overview can be improved by visualizing more than one measure at the same time. For example, each cell can be color-coded by the significance of association, and size-coded by the strength of association, or vice versa. Figure 3 shows an improved score overview for the ranking by association. The strength of association is coded by color in (a), (b), and (c). The original score overview (a) is improved by introducing the significance

measure for size-coding. In Figure 3 with a breakfast cereal dataset, all variables (for nutritional information) except three categorical variables (SHELF, TYPE, MFR-manufacturer) were transformed to be categorical variables by binning before applying the association measures.

<<insert Figure 3 about here>>

According to Mackinlay (Mackinlay, 1986), color saturation is more perceptually accurate to represent ordinal data types than area is. Thus, the ranking measure (strength of association for Figure 3) would be better coded by color saturation, and confidence measure (significance of association for Figure 3) would be better coded by size. Users can better identify more meaningful (or significant) and interesting scatterplots in (b) or (c) than in (a). Many associations look strong in (a), but not all of them are statistically significant. The significance information is not available in (a), but after size-coding the significance information, less significant associations get less attention, so significant strong associations can be more clearly recognized.

Similar coding schemes can be applied to other ranking criteria for quantitative variables. The quadracity ranking criterion uses the coefficient of x^2 of the regression curve as the score for each pair of variables. We can use, for example, least square error measures as a significance measure for the quadracity ranking criterion (Seo & Shneiderman, 2005b). Figure 4b more clearly reveals interesting and important scatterplots than the one without size-coding (Figure 4a). Many dark-shaded cells attract attention in Figure 4a, but not all of them retain their visibility after introducing the size-coding scheme in Figure 4b.

<<insert Figure 4 about here>>

5 RANKING BY CLUSTER SIMILARITY FOR PARTITIONS

Users often have to partition a multidimensional dataset according to categorical information. Then they have to compare the partitions to identify the differences between them. Such comparisons can be facilitated by clustering each of them and comparing the clustering results instead of comparing them directly. Considering that the clustering result for each partition can serve as new categorical information, we present in this section a novel way to utilize the categorical information together with clustering algorithms.

We first define two partitioning methods, vertical and horizontal partitioning, according to the direction the original dataset is split. Once partitions are generated, users can apply clustering algorithms to each partition and compare the clustering results to find groups of items that are noticeably different in performance across partitions. Since a clustering algorithm can be thought of as a way to generate a new categorical attribute representing cluster identifiers, we can facilitate this process by introducing a new ranking criterion for categorical variables, or ranking by cluster similarity measure.

5.1 Vertical and Horizontal Partitioning

Numerous microarray projects are case-control studies. In other words, microarray projects usually include more than two different phenotypes of samples (e.g. normal samples vs. cancer samples), and each sample is usually represented as a column in a dataset. Then the phenotype information can be thought of as a category to stratify columns. The stratification partitions the original set of columns into two or more distinct subsets, each of which has only the columns with the same phenotype. We define this stratification as *vertical partitioning* (Figure 5), where each partition has the same set of rows (or genes). Users can start a vertical partitioning by assigning columns into two or more groups on the clustering dialog box in

HCE. Each partition can then be fed into the clustering algorithm to generate separate clustering results. By comparing those clustering results, a biologist could find an interesting group of genes that are similarly or differentially expressed in different groups of homogeneous samples.

<<insert Figure 5 about here>>

Unlike the case-control microarray projects, there are many datasets where a different partitioning is more meaningful. For example, genome-wide association studies often generate datasets where each row represents a subject and each column represents a property of the subjects (e.g. gender, ethnicity, weight, height, and genotype and phenotype information). For these projects, researchers are more interested in stratification by a categorical column such as gender or ethnicity. We define this stratification as *horizontal partitioning* (Figure 6), where each partition has the same set of columns. In doing so, researchers can identify important patterns that could not be found otherwise because they exist only in a part (or partition) of the dataset. In this case, clustering can then be performed in each partition of subjects to generate interesting groups of quantitative columns. Users can start a horizontal partitioning by designating a categorical column as a stratification variable on the clustering dialog box in HCE.

<<insert Figure 6 about here>>

As an example, we show how a partitioning is done in HCE using a real biological dataset on spinal cord injuries (Di Giovanni et al., 2005). A group of biologists studied the molecular mechanisms of spinal cord degeneration and repair. They analyzed the spinal cord above the injury site (at the thoracic vertebrae T9) at various time points up to 28 days post injury. Mild,

moderate and severe injuries were examined. We only use two categories of injury for this example; 10 control samples and 12 severe injury samples.

In Figure 7, the spinal cord injury dataset was vertically partitioned into two groups according to the injury type (no injury vs. severe injury). In the input data file, the genes are in the rows and the samples are in the columns. The first ten columns represent the control samples (without injury), and the next 12 columns represent the severe injury samples. Each partitioned group has different samples but the two groups have the same genes. Each window in Figure 7 shows a clustering result of genes within each partition with Euclidean distance measure.

Once clustering is completed for each partition separately in HCE, users can define clusters for each partition by moving the minimum similarity bar (Figure 7). A typical user would create tens of clusters in each dendrogram view and then look for similarities and differences in items; a very tedious process. The user can then click on a cluster on a dendrogram to examine how elements in the cluster are distributed on other dendrograms. For example, when the user clicks on a cluster in Figure 7 (see the label A), all genes in the cluster are highlighted with orange triangles under the dendrogram display. In Figure 7, the selected cluster shares most of the genes with the cluster labeled B, but there are several dissenting genes shown outside the cluster B. To accelerate this could-be tedious work, we took inspiration from the rank-by-feature framework that ranks 1D and 2D projections according to some criteria such as correlation coefficient, entropy, or outlierness. The next section presents a new ranking criterion to facilitate this analysis process.

<<insert Figure 7 about here>>

5.2 Ranking for Partitioned Datasets using Cluster Similarity

Researchers can often gain deeper insights into multidimensional datasets with categorical information by comparing partitions according to categorical information. Clustering each partition makes such comparisons systematic and efficient. We consider the problem of comparing partitions a problem of comparing clustering results. Then, we address the problem by introducing a new ranking criterion in the rank-by-feature framework for cluster comparison. We first present a new ranking criterion for cluster similarity, and then we revise the score overview and the scatterplot accordingly. We then show two application examples of the enhanced rank-by-feature framework with the two partitioning methods defined in Section 5.1.

Clustering can be thought of as generating a new categorical variable for cluster labels (e.g. cluster1, cluster2, and so on). The new variable could take part in the ranking by association to identify other variables that have strong dependence with the new variable, but we can also define a new ranking criterion for cluster comparison. Suppose two clustering results ($CR1$ and $CR2$) have been produced with two separate groups. We suggest a heuristic similarity measure to compare two clusters each of which is from $CR1=\{CR1_i|i=0..n_1\}$ and $CR2=\{CR2_j|j=1..n_2\}$:

$$ClusterSimilarity(CR1_i, CR2_j) = \frac{\frac{|CR1_i \cap CR2_j|}{|CR1_i|} + \frac{|CR1_i \cap CR2_j|}{|CR2_j|}}{2} = |CR1_i \cap CR2_j| \cdot \frac{|CR1_i| + |CR2_j|}{2 \cdot |CR1_i| \cdot |CR2_j|} \quad (5.1).$$

Other measures such as the correlation between average patterns of two clusters or the F-measure (Rijsbergen, 1979) can be possible choices for measuring similarity between two clusters. The rank-by-feature framework is easily extended to include such cluster similarity measures. For the comparison of the two clustering results, the goal was to rank all clusters in one partition with clusters in the other partition by a similarity measure. If a user defines *n* clusters for the first partition and *m* clusters for the second partition, the score overview matrix of the rank-by-feature framework would have *m* x *n* cells, which could be color coded to show the similarity of clusters. The coordination between clustering result displays and the score overview enables users to identify interesting clusters, and a simple grid display clearly reveals correspondence between two clusters.

In this case, the score overview changes to show measures between clusters instead of those between variables. The scatterplot browser also changes to display relationships between clusters instead of those between variables. Figure 8 shows two rank-by-feature framework user interface components for ranking by cluster similarity. In the score overview (a), each row or column represents a cluster in a clustering result. Each cell is color-coded by a cluster similarity measure like the one in equation 5.1. Similar cluster pairs can be preattentively identified in this display. For example, in Figure 8a, the darker cells indicate similar pairs of clusters. Intuitively, equation 5.1 means that the more similar the two clusters are in size and the more items they share in common, the bigger the *ClusterSimilarity* is.

In the scatterplot (Figure 8b) for comparison of two clusters, each vertical or horizontal line represents an item in the clusters. An intersection point has a blue square if the vertical item and the horizontal items are the same. The fraction of vertical or horizontal lines with a blue dot visualizes the similarity between two clusters. In other words, if users see less empty

lines in the scatterplot, the two clusters are more similar. The linear alignment of the blue dots on the scatterplot view indicates approximately how similar the orders of items are in the two selected clusters.

<<insert Figure 8 about here>>

As an application example, this new ranking criterion (see equation 5.1) is applied to the same spinal cord injury dataset shown in Figure 7. Once two clustering results are generated in the same way as in Figure 7 after a vertical partitioning (one with the control samples and the other with the severe injury samples), two dendrogram views are shown side by side. Since the two dendrogram views are coordinated with each other and other views, users can click on a cluster in a dendrogram view and then the items in the cluster are highlighted with orange triangles in all other views including the other dendrogram view. Just by looking at where the orange triangles appear in the other dendrogram view, users can notice how items in a cluster are grouped in the other clustering result.

However, this manual process becomes tedious as the number of clusters to compare increases. The ranking by cluster similarity facilitates this task by providing an overview of similarity measures for all possible pairs of clusters in the two clustering results. When users select the “Cluster Similarity” ranking criterion from the scatterplot ordering tab in HCE, a modeless dialog box pops up (Figure 9) and users can drag and drop the target-shaped icon on dendrogram views to choose two dendrograms to compare.

<<insert Figure 9 about here>>

Once users select two clusters to compare, the ranking result by the cluster similarity measure for the spinal cord injury dataset looks like Figure 10. Each cell of the score overview represents the similarity of a pair of clusters. A mouse-over event on the overview highlights the corresponding clusters with a yellow bounding rectangle in the selected dendrograms. The revised scatterplot view shows the overview of the mapping of items between the two clustering results. Any change in the number of clusters by dragging the minimum similarity bar updates the current score overview instantaneously to reflect the change.

<<insert Figure 10 about here>>

The ranking by cluster similarity can also be applied after a *horizontal* partitioning. As an example, we use the same ranking criterion to a small multidimensional dataset ("Sleep in Mammals," http://lib.stat.cmu.edu/). This dataset has 62 mammals in the rows and 7 variables (body weight, brain weight, non-dreaming sleep hours, dreaming sleep hours, total sleep hours, maximum life span, gestation time, and overall danger index) in the columns. Overall danger index is a categorical variable which has two levels, "high" and "low." Users can partition the rows by this categorical variable and generate two partitions. By clustering each partition, users can generate two clustering results of the columns.

<< insert Figure 11 about here>>

In Figure 11, users generate two clusters using the minimum similarity bar in each dendrogram view. The score overview (at the bottom left corner) shows the similarity values for all four possible pairs of clusters. Two black cells indicate that there are two perfectly matching pairs of clusters, but the revised scatterplot view (at the bottom right corner) shows

that the arrangement of the items in the two clusters are not consistent. The second cluster in the left dendrogram shows that the nondreaming sleep hours dominates total sleeping hours for the mammals with the high overall danger index. This might mean that those mammals are too cautious to have a long dreaming sleep.

6 CASE STUDY

We teamed up with biomedical researchers at a large biology research laboratory to evaluate and improve the new rank-by-feature framework for multidimensional datasets with categorical information. The biomedical research team consisted of four members: (1) a male geneticist (Ph.D.) with a strong research record in muscular dystrophy research and genome wide expression profiling studies, (2) a female biomedical engineer (Ph.D.) with expertise in high dimensional data clustering and machine learning, (3) a female biostatistician (Ph.D) with expertise in statistics and epidemiology working on genome-wide SNP (Single Nucleotide Polymorphism) association studies, and (4) a female information systems architect (MS in Computer Science) with many years of professional experience in developing infrastructure for various research projects in the laboratory. They were experienced HCE users and the biostatistician had used the rank-by-feature framework for her projects. They were all involved in designing and implementing a research framework for genome wide SNP association studies. They were enthusiastic about interactive visualization systems such as HCE.

This case study was in continuation of our previous case study (Seo & Shneiderman, 2006) with a larger dataset from the same SNP association study. In addition to investigating gene functions, the biomedical field has begun to focus on SNP studies, which will eventually detect important single nucleotide changes in gene sequences and may have a huge impact on

individual long-term and short-term healthcare. As more and more SNPs are identified and as high-intensity SNP chips are developed, it is becoming challenging to study and analyze such huge SNP association datasets. Since our case study participants have already started to face this problem, our case study included SNPs which were all categorical, some administrative categorical variables, and some quantitative traits.

6.1 Problems, Tasks and the Dataset

The genetic analysis of quantitative traits (or phenotypes) is fast becoming problematic with the current environment of whole-genome analyses with very large datasets. Historically these analyses were limited to a few genotypes and a few phenotypes and were easily accomplished using a standard statistical package where each genotype-phenotype association was evaluated independently. With the rise of data comprising genotypes from the entire genome, evaluating each genotype-phenotype association individually is no longer an option for the average researcher. This type of analysis requires advanced statistical knowledge and the ability to use a statistical package, such as SAS, where an understanding of the programming language is required. The average researcher wanting to detect genotype-phenotype associations in a large dataset can no longer rely on many of the statistical programs they are familiar with.

Researchers often have to deal with datasets containing more than 50 phenotypes and more than 200 genotypes. The time required to analyze each of these genotype-phenotype associations is prohibitive. Once they consider that each genotype-phenotype comparison has several facets to the analysis, different genetic models, covariates and stratifying factors, the required time increases dramatically. To date, they have been dealing with the data volume simply by analyzing a limited number of genotype-phenotype comparisons at a time. Each set

of genotypes and phenotypes are analyzed using a bioinformatics tool such as Stata (StataCorp). The initial analysis consists of bivariate analyses between each phenotype and potential covariate and stratifying factors. Those which are significantly related are then used in the analysis of variance models for each genotype-phenotype pair.

By using ANCOVA (Analysis of Covariance), researchers can calculate adjusted means for each phenotype (by stratifying by a genotype) and they can use those means to determine which genetic model is correct. Once the correct model is determined, covariates and stratifying factors and genetic models are chosen, and then the analysis proceeds with pairwise comparisons of the phenotype among each genotype. Finally, for each ANCOVA model, a measure of the variability attributable to the genotype is calculated using a likelihood ratio test comparing the full model with one constrained to exclude the genotype. The output of this analysis consists of several pages of results which then have to be organized into an easily understandable table or figure. Many of the results can be directly output to data files or tables, but any required figures usually need to be made individually.

The dataset used in this case study was phenotype and genotype data from a study of 1242 young adults (average age 24 years) participating in an exercise intervention for genetic association studies (FAMuSS study) (Thompson et al., 2004). This dataset included 23 SNPs (single nucleotide polymorphisms), weight and height, BMI, and semi-automated volumetric magnetic resonance imaging data of fat, muscle, and bone, both before and after a 12-week intervention. The subjects performed unilateral supervised resistance training of *non-dominant* arms. 48 quantitative traits (phenotypes) have been selected as the test data for this case study. In the input data file, the subjects are in the rows, and all genotype and phenotype information are in the columns.

6.2 Methods and Goals

We focused this case study on the usefulness of the new features of the rank-by-feature framework for multidimensional datasets with categorical information. We conducted one-hour biweekly team meetings for two months with the biomedical research team. After the team meeting to introduce our case study, each member of our research team had a tutorial session for about 30 minutes. Follow-up questions and comments made the tutorial session extend up to an hour.

After each biweekly meeting, team members were encouraged to use the new features in HCE for their projects until next meeting. While all of them tried to do so, the biostatistician was the most enthusiastic participant since her main job was to analyze the dataset with statistical tools. All participants except the biostatistician seemed to be using the new features less as time went by while they all shared comments and discussions at biweekly meetings. The biostatistician had used the new features with some other SNP association datasets, but other participants had stayed only with the given test dataset over the two month period. We think this is because the new features became more beneficial to the biostatistician as we improved the features throughout the study. Whereas, others did not expect direct benefits from this case study, but rather appeared as though they thought they were just helping us evaluate the system.

We communicated with participants in person or by email. Whenever asked, we answered questions using live demonstrations if possible. Important suggestions or implementation requests were implemented as much as possible, so that participants could use an improved version before the next meeting. Since those improvements were all incremental, there were no serious usability issues regarding the change of interfaces and interactions.

We concentrated on the following aspects in conducting this case study:

- How are categorical variables incorporated into the analysis process?
- How do revised score overviews improve users' analysis tasks?
- What improvements are further anticipated?

The next subsections describe case study results with the four researchers, but the most active contributor was the biostatistician who found the new features useful in her daily analysis tasks.

6.3 Findings and Suggestions

The biostatistician first checked the correlations between the measurements of dominant arms before and after the exercise intervention, which should not change much because the resistant training was done only with their non-dominant arms. She found a few measurement errors using the "Correlation Coefficient" ranking. Then, the biostatistician tried to partition the dataset vertically into the before and after exercise measurements since it was important to find meaningful changes after the exercise in the case study dataset. She assigned variables into two groups (before and after the exercise intervention) in the clustering dialog box to cluster them. Once the clustering was done, she adjusted the minimum similarity bar at the two dendrogram views until visibly meaningful separations were made at each view. After using coordination between the two views several times, she eventually tried the new ranking criterion, or "Cluster Similarity."

She examined the revised score overview to find a very similar pair of clusters (Figure 12). One of them was a group of subjects who were very high in six measurements after the exercise intervention. However, in the revised scatterplot a few of the subjects in the group

were not present in the matching cluster of the other clustering result with the measurements before the intervention. The deviating subjects (see B in Figure 12) were easily identified when she clicked on the cluster (see A in Figure 12) to see them highlighted apart from other subjects who maintained the similar measurement pattern to the selected group. Those who showed a great change and deviated from the group in the pre-exercise measurement clustering result could be worth more attention afterwards because it could reveal an unexpected biologically significant association between a SNP and some phenotypes.

<< insert Figure 12 about here>>

While the cluster-similarity ranking interested participants to a certain degree, they were also intrigued by the fact that more dynamic stratifications were possible in the enhanced rank-by-feature framework. For example, they could stratify variables according to the types of information (genotype data or phenotype data) for the association score evaluation (i.e. vertical partitioning), and at the same time they could partition subjects by the gender column to generate multiple score overviews (i.e. horizontal partitioning). Multiple score overviews with a dynamic stratification enabled participants to gain in-depth insight into the case study dataset (Figure 13). The case study dataset consisted of measurements of muscle, fat and bone size in both genders and in several different ethnic groups. The enhanced rank-by-feature framework quickly showed the vast differences in muscle sizes between men and women. It also showed just how strongly related body weight was to these size measurements, confirming the hypothesis that body weight is very important as a covariate in any analysis. A surprising insight was also gained in that body weight, while extremely important for the analyses using single size measurements, is not as important when exploring changes in size before and after the exercise.

Similar insights were seen with other covariates and differences between men and women continued to be clearly shown through the visualization. In Figure 13, a participant selected “ANCOVA” as a ranking criterion and “gender” as the partitioning variable. Then HCE calculated ANCOVA models for all combinations in each partition to generate one score overview for each partition and one for the entire population without partitioning. They interactively examined the three coordinated score overviews (one for female, one for male, and one for the entire population). Since the dark-shaded cells indicated significant associations, the biostatistician easily identified such cells, for example, the one for AKT1_G738A (SNP) and EX_PRE_WA_VOL (baseline whole arm volume of the exercised non-dominant arm) at the score overview for female. Then she immediately noticed that the association was neither significant with the male population nor with the whole population by seeing the highlighted corresponding cells in the two other score overviews.

<< insert Figure 13 about here>>

6.4 Discussion

The benefits of the new rank-by-feature framework are two-fold from the biological standpoint. First, the ease with which HCE can simultaneously analyze large numbers of genotype-phenotype associations is an enormous benefit. Rather than taking substantial time to analyze only a limited number of associations, the new rank-by-feature framework allows users to easily scan all of the data for interesting and significant results. Those few statistically significant results can then be explored more fully. The biostatistician team member who joined a demonstration session after a biweekly team meeting said this tool could keep researchers from missing important associations, and could free them from tedious time-consuming manual tasks that use lengthy text outputs and static diagrams.

The second major benefit is its visual interface. Users can easily visualize those associations which are significant, rather than scanning many rows of numbers to pick out those which are most interesting. Simultaneously showing results between different groups on the same screen reduces the effort needed to compare several lists of numbers. Showing figures of each ANCOVA model, complete with the adjusted means, not only aids in the interpretation, but reduces the need to use other graphing software. A participant said,

"HCE is a wonderful tool for comparing large numbers of genotypes and phenotypes in a time-friendly manner. Its visualizations make interpretation of the results easy and will allow ones' time to be devoted to further exploring significant results rather than scanning large numbers of insignificant genotype and phenotype comparisons."

A participant also pointed out a potential limitation of the new tool, which is not the program itself, but the potential for misuse. By allowing the easy analysis of data using any covariate or genetic model (recessive, dominant, etc.), the possibility of fishing the data for any significant association exists. If the user does not have a strong background in genetics or statistics, one could easily find associations, which while being statistically significantly, may not be biologically plausible.

We made a series of incremental improvements to the HCE visualization tool through team meetings and the following demonstration sessions. After the first meeting where datasets and tasks were explained, we decided to add ANCOVA to the rank-by-feature framework for the association between a categorical variable and a quantitative variable to make our case study more pragmatic. More importantly, since a certain association could exist only in a certain sub-category, researchers had to incorporate the partitioning methods into the

ANCOVA ranking criterion to detect biologically and statistically meaningful associations in the SNP association studies. Thus, we enabled users to dynamically stratify the dataset and examine multiple score overviews (one for each partition by a stratification variable) at once using interactive coordination between those score overviews. Users first choose the new ranking criterion, or ANCOVA, and then they can assign variables to appropriate categories including covariate and stratification variables. When users select the ANCOVA ranking, HCE replaces the scatterplot view with a bar chart view where each bar represents a level of a genotype.

In a demonstration session with the case study dataset, we examined score overviews to find an association whose cell is dark brown, meaning the association was significant. However, when we saw the bar chart for the association, it did not look significant (Figure 14a). After examining the dependent variable in the histogram ordering view, it turned out that an outlier drew the scale of the bar chart larger than necessary. While outlier detection and removal could also solve this problem, we decided to adjust the maximum value so that the bar with the maximum average value can reach the top (Figure 14b). After the adjustment, the bar chart looked more consistent with the color coding at the score overview.

<<insert Figure 14 about here>>

In a different occasion, a participant raised an important point regarding error bars displayed within a bar chart. Such error bars can assist researchers by visually confirming the statistical significance. We improved the bar chart by showing error bars using the adjusted standard error measure. We adjusted the standard error measure using the same formula used

to adjust means in ANCOVA. We still need to confirm that this adjustment of the standard error measure is statistically correct.

HCE had been fairly stable during the case study, but there had been some usability problems raised by the case study participants. First, the inclusion of multiple data types in the input file increased the chance of malfunctions because of the input file formatting errors. A more thorough format checking routine and more specific error messages can help solve this problem in HCE (and other similar information visualization tools). Second, two participants had some difficulties selecting a dendrogram view by dragging and dropping an icon for the cluster similarity ranking (see Figure 9). A more distinctive selection marker around a dendrogram might help together with a noticeable message to show the current selection in the selection dialog box. Third, a couple of participants had difficulty understanding the concept of ANCOVA and the stratification by a categorical variable. These concepts could be difficult for some potential users to grasp, so effective training methods and more case studies would be helpful for new users and designers.

Overall, our team members seem to have enjoyed the case study and the live demonstration sessions. Mentionable suggestions that deserve implementation in HCE include the adjusted error bars display on the bar chart and the partitioning by more than one categorical variable. Some participants' quick comments and suggestions made good sense to us from the information visualization perspective. For example, one of the participants suggested that the bar chart view could be embedded in the score overview. It might be more effective to use a fisheye view for the score overview, where users can click on a cell to enlarge the cell and see its bar chart while shrinking surrounding cells. It is also necessary to

make the new rank-by-feature framework scaleable and generalizable to deal with the increasing volume of genome-wide association datasets.

When it is hard to confine users' tasks to a very specific set or when there are no alternative tools to compare with, a case study with real users and real datasets can be a good alternative evaluation method. Even though the result from one case study cannot be directly applied to different situations with different users and different tasks, other designers in a similar situation could gain valuable insight into the design and implementation process. The case study and interdisciplinary design process presented in this paper will eventually lead to a public domain software package able to quickly and intuitively interrogate large datasets of genome-wide genotype data for multiple phenotypes, and to discover relationships between phenotypes that result in an ensemble risk for common disease (metabolic syndrome). We hope that this case study can provide useful guidance to other designers and programmers with similar problems and datasets.

7 CONCLUSION

Stimulated by users' requests for the capability to deal with multidimensional datasets containing categorical information, we designed and implemented extensions to the rank-by-feature framework in the Hierarchical Clustering Explorer:

- First, we enabled users to partition the data vertically or horizontally by categorical information in the dataset. To compare resulting partitions, users can either create clusters within each partition or evaluate each of them separately with various ranking criteria for categorical information. They can then look for similar or

differing clusters, or compare coordinated overviews for partitions. They can drill down to find the specific items that account for differences.

- Second, we added a new ranking criterion, or Cluster Similarity to the rank-by-feature framework to facilitate the comparison task. The score overview and the scatterplot were enhanced accordingly. The improved score overview provided by a heatmap display combined with rapid coordination among windows provides support for this challenging task.

- Third, we enabled users to interactively examine multiple score overviews after dynamic stratifications. This extension made it possible for researchers to identify patterns or associations hidden in the stratified subgroups, which cannot be revealed without stratification.

We conducted a case study with a biomedical research team to evaluate and improve the new extensions to the rank-by-feature framework. The case study presented evidence that the rank-by-feature framework extensions for categorical multidimensional datasets were useful for genome-wide SNP association studies. At the same time, it also suggested potential future work such as embedding the bar char view in the score overview using a fisheye view. Due to screen space limitations, the current implementation can handle approximately 100 clusters each containing approximately 100 items on an average PC. The current implementation is already useful, but scaling up using aggregation and distortion techniques is a natural next step.

REFERENCES

Abello, J., & van Ham, F. (2004). Matrix Zoom: a visual interface to semi-external graphs. In *Proceedings of IEEE Symposium on Information Visualization* (pp. 183-190). Austin, TX, USA: IEEE Computer Society Press.

Blasius, J., & Greenacre, M. L. (1998). *Visualization of Categorical Data*. San Diego: Academic Press.

Di Giovanni, S., Faden, A. I., Yakovlev, A., Duke-Cohan, J. S., Finn, T., Thouin, M., et al. (2005). Neuronal plasticity after spinal cord injury: identification of a gene cluster driving neurite outgrowth. *The FASEB Journal, 19*(1), 153-154.

Elomaa, T., & Rousu, J. (1999). General and efficient multisplitting of numerical attributes. *Machine Learning, 36*(3), 201-244.

Friendly, M. (2000). *Visualizing Categorical Data.*: SAS Publishing.

Gibson, D., Kleinberg, J. M., & Raghavan, P. (2000). Clustering categorical data: an approach based on dynamical systems. *VLDB Journal, 8*(3-4), 222-236.

Gonzales, V., & Kobsa, A. (2003). A workplace study of the adoption of information visualization systems. In *Proc. I-KNOW'03: Third International Conference on Knowledge Management* (pp. 92-102).

Guha, S., Rastogi, R., & Shim, K. (1999). ROCK: a robust clustering algorithm for categorical attributes. In *Proceedings of the 15th International Conference on Data Engineering* (pp. 512-521). Sydney, Australia.

Guo, D., Gahegan, M., Peuquet, D., & MacEachren, A. (2003). Breaking down dimensionality: an effective feature selection method for high-dimensional clustering.

In *Proceedings of the third SIAM International Conference on Data Mining, Workshop on Clustering High Dimensional Data and its Applications* (pp. 29-42). San Francisco, CA, USA.

Henry, N., & Fekete, J.-D. (2006). MatrixExplorer: a dual-representation system to explore social networks. *IEEE Transactions on Visualization and Computer Graphics, 12*(5), 677-684.

Kincaid, R. (2004). VistaClara: an interactive visualization for exploratory analysis of DNA microarrays. In *Proceedings of the ACM Symposium on Applied Computing* (pp. 167-174). Nicosia, Cyprus: ACM Press.

Kules, B., Shneiderman, B., & Plaisant, C. (2003). Data exploration with paired hierarchical visualizations: initial designs of PairTrees. In *Proceedings of the National Conference on Digital Government Research* (pp. 255-260). Boston, MA, USA.

Liu, H., & Motoda, H. (1998). *Feature selection for knowledge discovery and data mining.* Boston: Kluwer Academic.

Mackinlay, J. (1986). Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics, 5*(2), 110-141.

Moore, D. S., & McCabe, G. P. (1999). *Introduction to the Practice of Statistics* (3rd ed.). New York: W.H. Freeman.

Munzner, T., Guimbretiere, F., Tasiran, S., Zhang, L., & Zhou, Y. (2003). TreeJuxtaposer: scalable tree comparison using Focus+Context with guaranteed visibility. *ACM Transactions on Graphics, 22*(3), 453-462.

Press, W. H., Flannery, B. P., Teukolsky, S. A., & Vetterling, W. T. (1992). *Numerical Recipes in C : The Art of Scientific Computing*: Cambridge University Press

Rijsbergen, C. J. V. (1979). *Information Retrieval* (2 ed.). London: Butterworth.

Rosario, G. E., Rundensteiner, E. A., Brown, D. C., Ward, M. O., & Huang, S. (2004). Mapping nominal values to numbers for effective visualization. *Information Visualization, 3*(2), 80-95.

Saraiya, P., North, C., Lam, V., & Duca, K. (2006). An insight-based longitudinal study of visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 12(6), 1511-1522.

Seo, J., Bakay, M., Chen, Y.-W., Hilmer, S., Shneiderman, B., & Hoffman, E. P. (2004). Interactively optimizing signal-to-noise ratios in expression profiling: project-specific algorithm selection and detection p-value weighting in Affymetrix microarrays. *Bioinformatics, 20*(16), 2534-2544.

Seo, J., Gordish-Dressman, H., & Hoffman, E. P. (2006). An interactive power analysis tool for microarray hypothesis testing and generation. *Bioinformatics, 22*(7), 808-814.

Seo, J., & Hoffman, E. (2006). Probe set algorithms: is there a rational best bet? *BMC Bioinformatics, 7*(1), 395.

Seo, J., & Shneiderman, B. (2005a). A knowledge integration framework for information visualization. *Lecture Notes in Computer Science, 3379*, 207-220.

Seo, J., & Shneiderman, B. (2005b). A rank-by-feature framework for interactive exploration of multidimensional data. *Information Visualization, 4*(2), 99-113.

Seo, J., & Shneiderman, B. (2006). Knowledge discovery in high-dimensional data: case studies and a user survey for the rank-by-feature framework. *IEEE Transactions on Visualization and Computer Graphics, 12*(3), 311-322.

Siirtola, H., & Makinen, E. (2005). Constructing and reconstructing the reorderable matrix. *Information Visualization, 4*(1), 32-48.

Spotfire. Spotfire DecisionSite. from http://www.spotfire.com/

StataCorp. http://www.stata.com/.

Thompson, P. D., Moyna, N., Seip, R., Price, T., Clarkson, P., Angelopoulos, T., et al. (2004). Functional polymorphisms associated with human muscle size and strength *Medicine & Science in Sports & Exercise, 36*(7), 1132-1139.

Tukey, J. W., & Tukey, P. A. (1985). Computer graphics and exploratory data analysis: An introduction

In *Proceedings of Annual Conference and Exposition: Computer Graphics* (Vol. 3, pp. 773-785). Fairfax, VA, USA: National Micrograhics Association: Silver Spring.

Yang, J., Peng, W., Ward, M. O., & Rundensteiner, E. A. (2003). Interactive hierarchical dimension ordering, spacing and filtering for exploration of high dimensional datasets. In *Proceedings of IEEE Symposium on Information Visualization* (pp. 105-112). Seattle, WA, USA: IEEE Computer Society Press.

Zhao, P., Seo, J., Wang, Z., Wang, Y., Shneiderman, B., & Hoffman, E. P. (2003). In vivo filtering of in vitro expression data reveals MyoD targets. *Comptes Rendus Biologies, 326*(10-11), 1049-1065.

Ziegler, J., Kunz, C., & Botsch, V. (2002). Matrix browser: visualizing and exploring large networked information spaces. In *Proceedings of CHI '02 Extended Abstracts on Human Factors in Computing Systems* (pp. 602-603). Minneapolis, MN, USA: ACM Press.

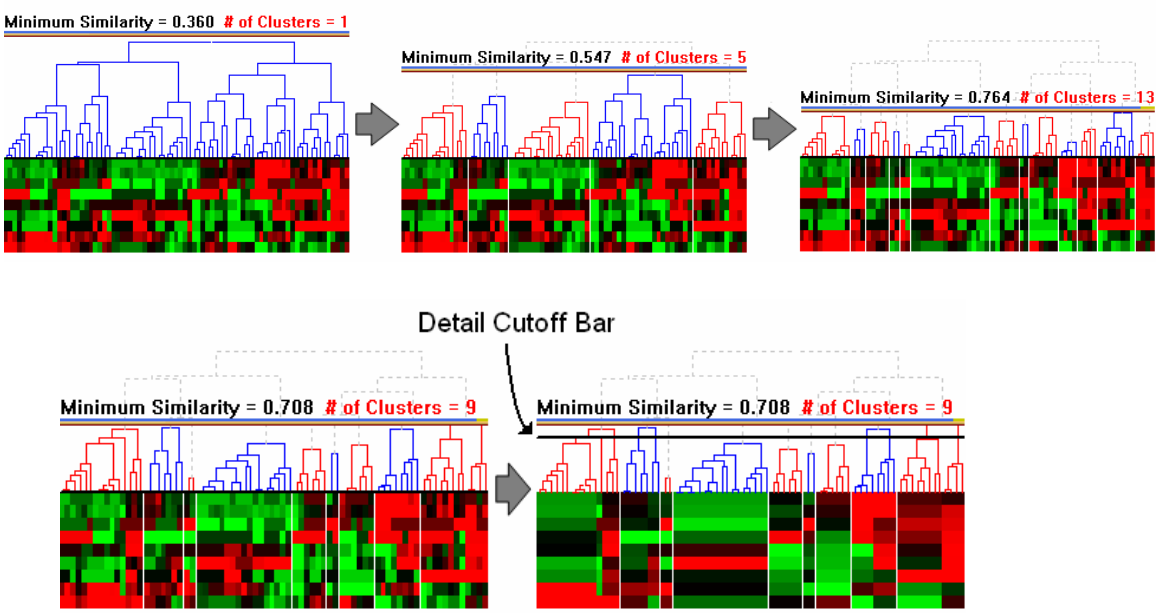


Figure 1. Two dynamic query controls in the dendrogram view of HCE. The minimum similarity bar (top) to help users find the right number of clusters by separating the subtrees below the bar, and the detail cutoff bar (bottom) to help users control the level of detail by rendering the subtrees below the bar with their averages. (see www.cs.umd.edu/hcil/ben60 for color figures)

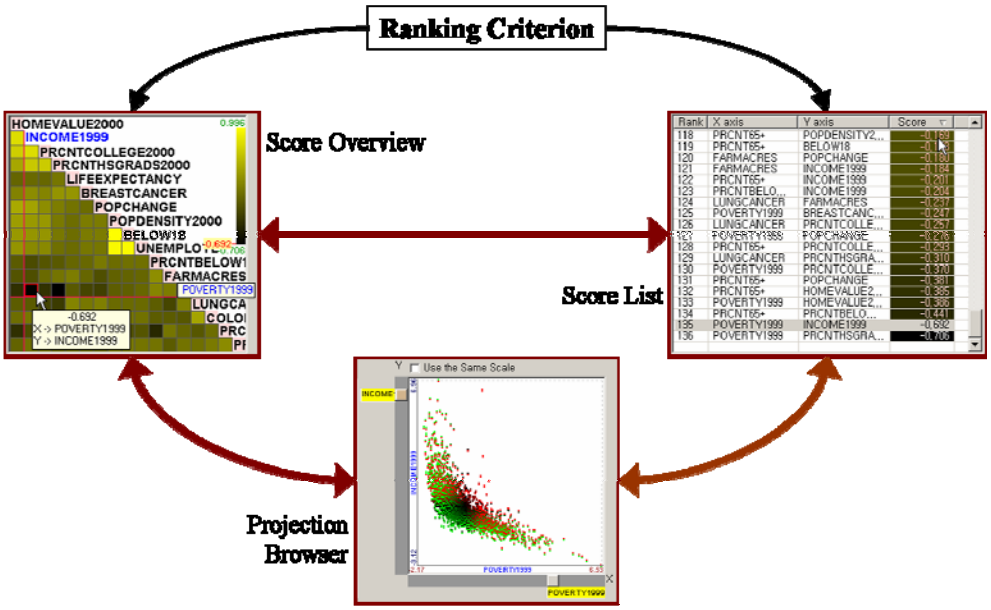
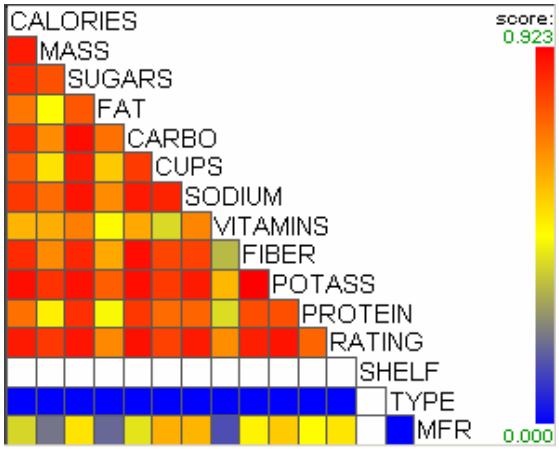
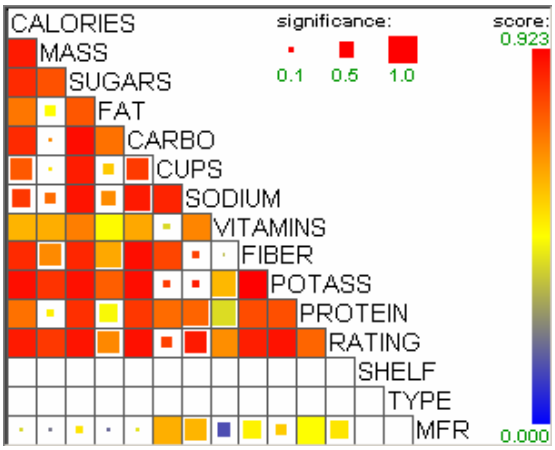


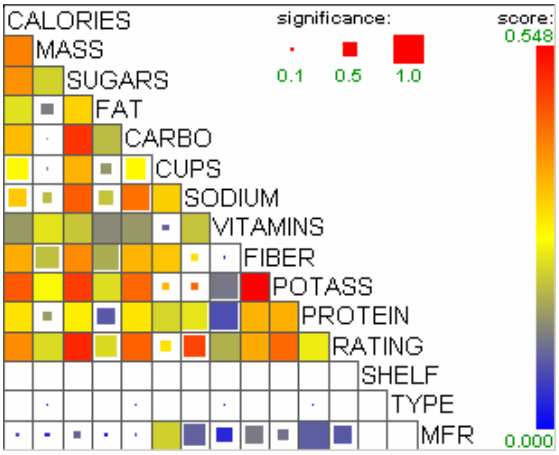
Figure 2. Rank-by-Feature Framework. Selecting a ranking criterion allows the user to see the ranking result in the score overview as well as in the score list. Moving the cursor over the overview or selecting a row in the list produces the corresponding scatterplot in the projection browser where one can also navigate the projections interactively.



(a) color only: Contingency coefficient C

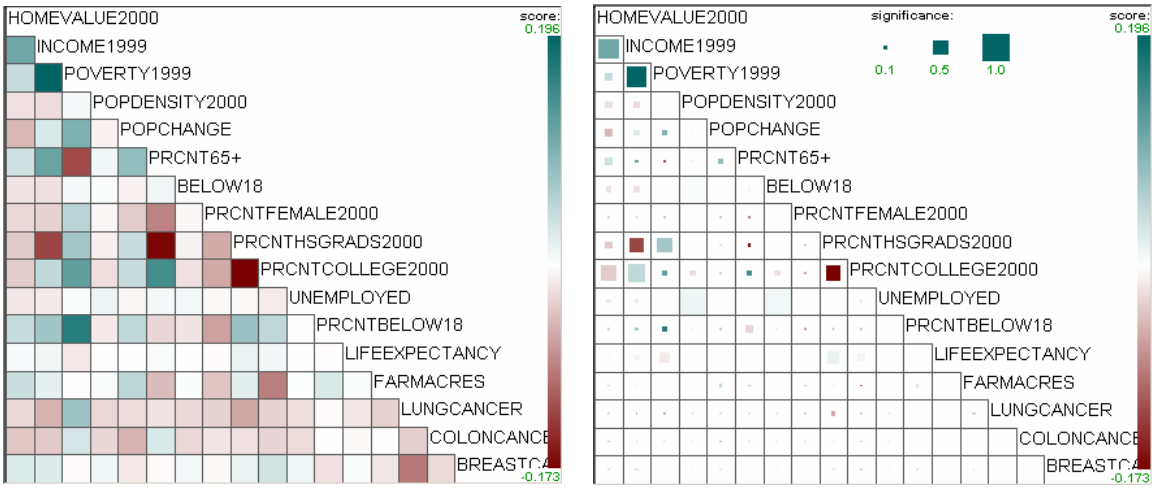


(b) color : Contingency coefficient C
size : Chi-square p-value



(c) color : Mutual information
size : Chi-square p-value

Figure 3. The score overview was enhanced for ranking by association (77 cereals dataset) in (b) and (c) from the original score overview shown in (a). The larger the rectangle, the more significant the association is in (b) and (c).



(a) without size-coding (b) with size-coding

Figure 4. Score overviews for quadracity ranking criterion (a multidimensional data set of demographic and health related statistics for 3,138 U.S. counties with 17 attributes). On the left (a) is the original score overview, and on the right (b) is the score overview with least-square error as size-coding and quadratic coefficient as color-coding. Only a few pairs retain their visibility after the size-coding.

	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8
type	A	A	A	A	A	B	B	B
$gene_1$								
$gene_2$								
……								
$gene_n$								

Figure 5: Vertical partitioning of the columns by the sample types A or B.

	A_1	a_2	a_3	a_4	a_5	a_6	a_7	gender
p_1								male
p_2								male
p_3								male
…..								
p_{n-1}								female
p_{n-2}								female

Figure 6: Horizontal partitioning of the rows by a categorical attribute 'gender'.

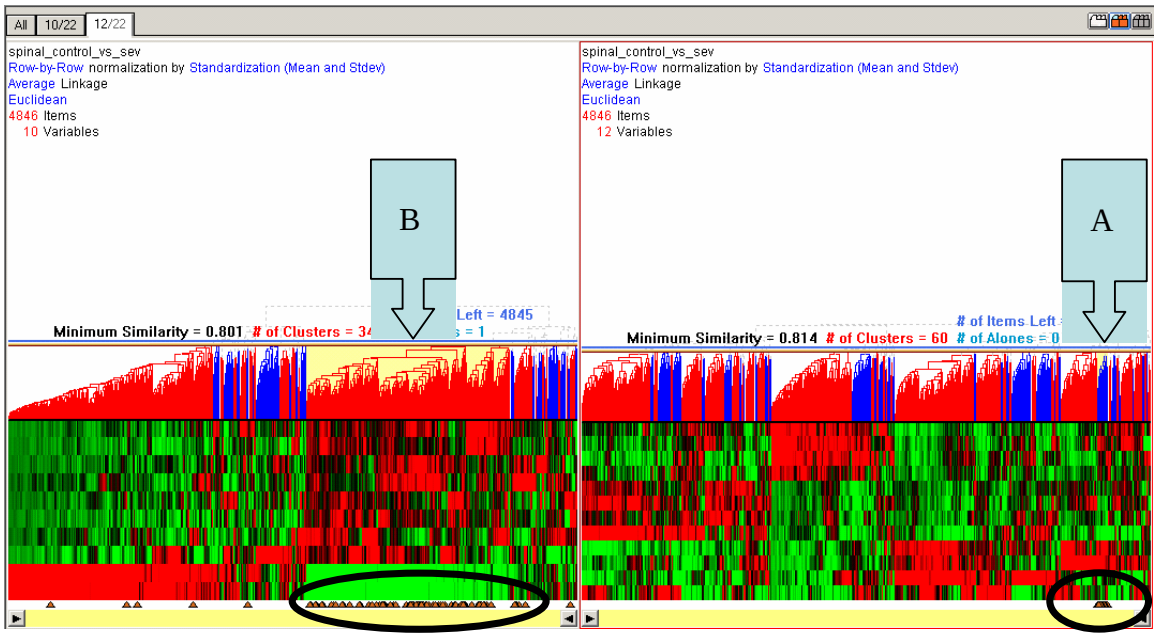


Figure 7. Comparison of clustering results for two partitions in a spinal cord injury dataset (Di Giovanni et al., 2005): one with 10 control samples (left) and the other with 12 sever injury samples (right). A click on a cluster on a dendrogram highlights items in the cluster on both dendrograms with orange triangles below the heatmaps. The user has clicked on a cluster on the right side of the rightmost dendrogram (A). The triangles on the left side show where the corresponding items appear in the other clustering result. They are mostly within cluster (B) but five appear to the far left and four are to the right.

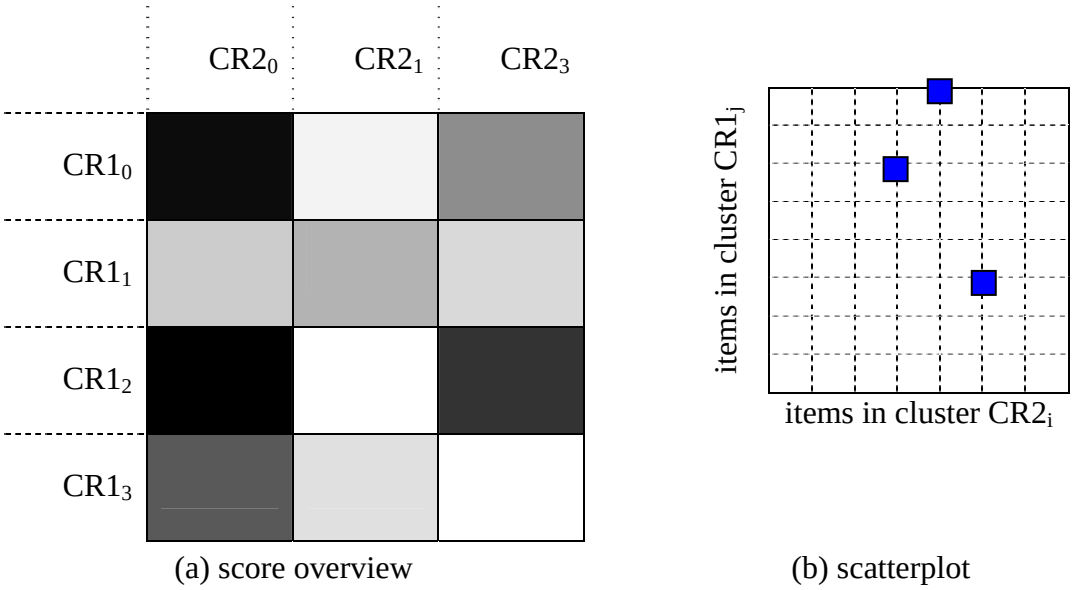


Figure 8. Score overview and scatterplot display for ranking by the cluster similarity measure as defined in equation 5.1. Each cell of the score overview encodes the *ClusterSimilarity* values for the corresponding pair of clusters. Each line represents an item and the lines are arranged in the order they appear in the clustering result. A blue square dot appears on the revised scatterplot if items from two clusters are matched.

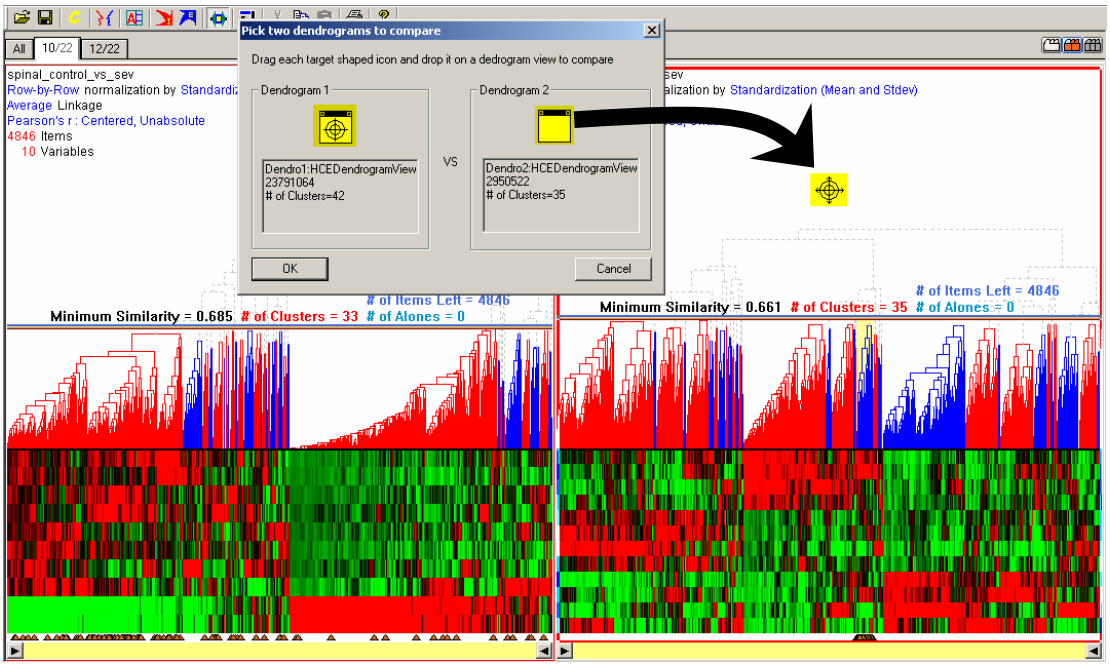


Figure 9. Interactive selection of two clustering results (or dendrograms). When a user selects the "Cluster Similarity" ranking criterion in the scatterplot ordering, a modeless dialog box opens, and the user can drag the target-shaped icon over the dendrogram views to pick the two clustering results to compare.

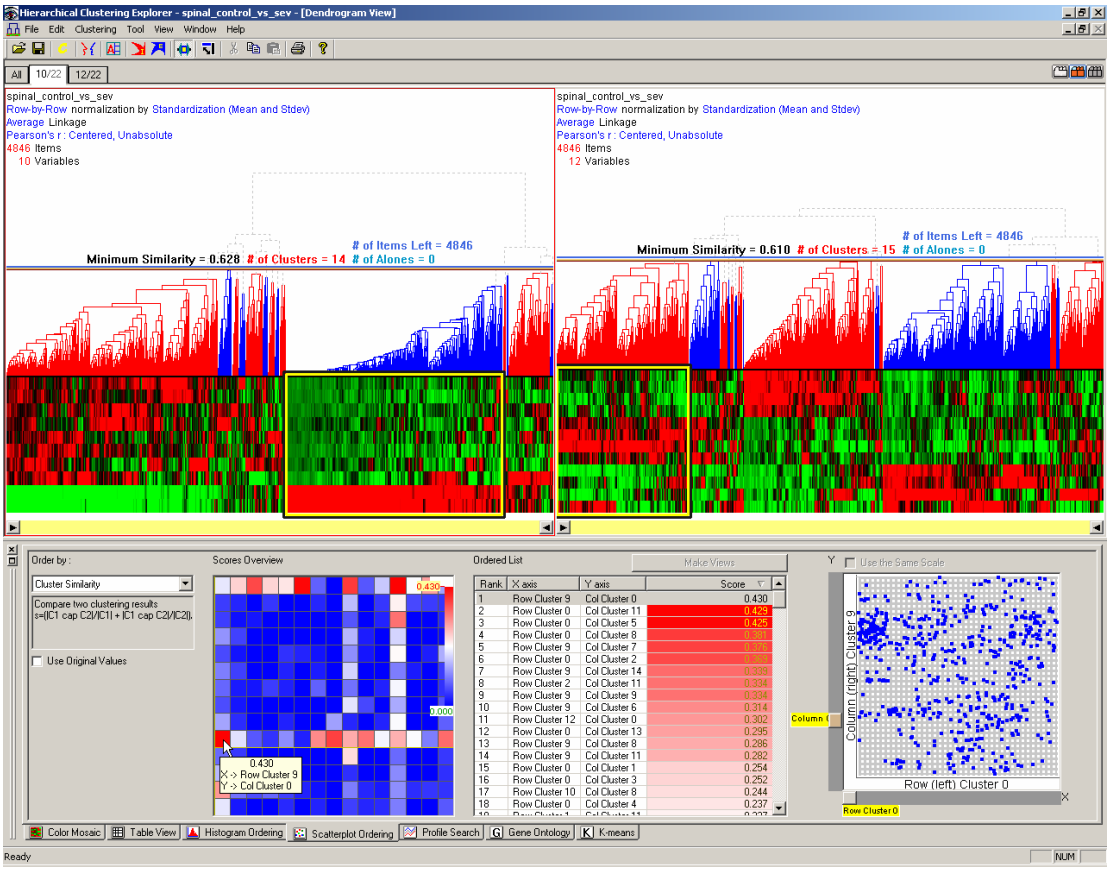


Figure 10. An example of ranking by cluster similarity with a spinal cord injury dataset (Di Giovanni et al., 2005) where there are two levels of the severity of injuries (vertical partitioning). The left dendrogram shows the clustering result with the control samples, and the right dendrogram shows the clustering result with the severe injuries samples. When users select a pair of clusters on the score overview by moving the cursor over a cell, the selected clusters are highlighted with a yellow rectangle in the dendrogram view and the scatterplot is updated with the two clusters on x- and y-axes.

Figure 11. An example of a clustering results comparison with a small mammals sleep dataset containing 63 mammals with two partitions by overall danger index (horizontal partitioning). The score overview (at the bottom left) shows two pairs of matching clusters with the two dark cells. The selected cluster pair is seen in the scatterplot view (at the bottom right) where the lack of a similar structure is evident from the blue dots not being aligned along the diagonal line.

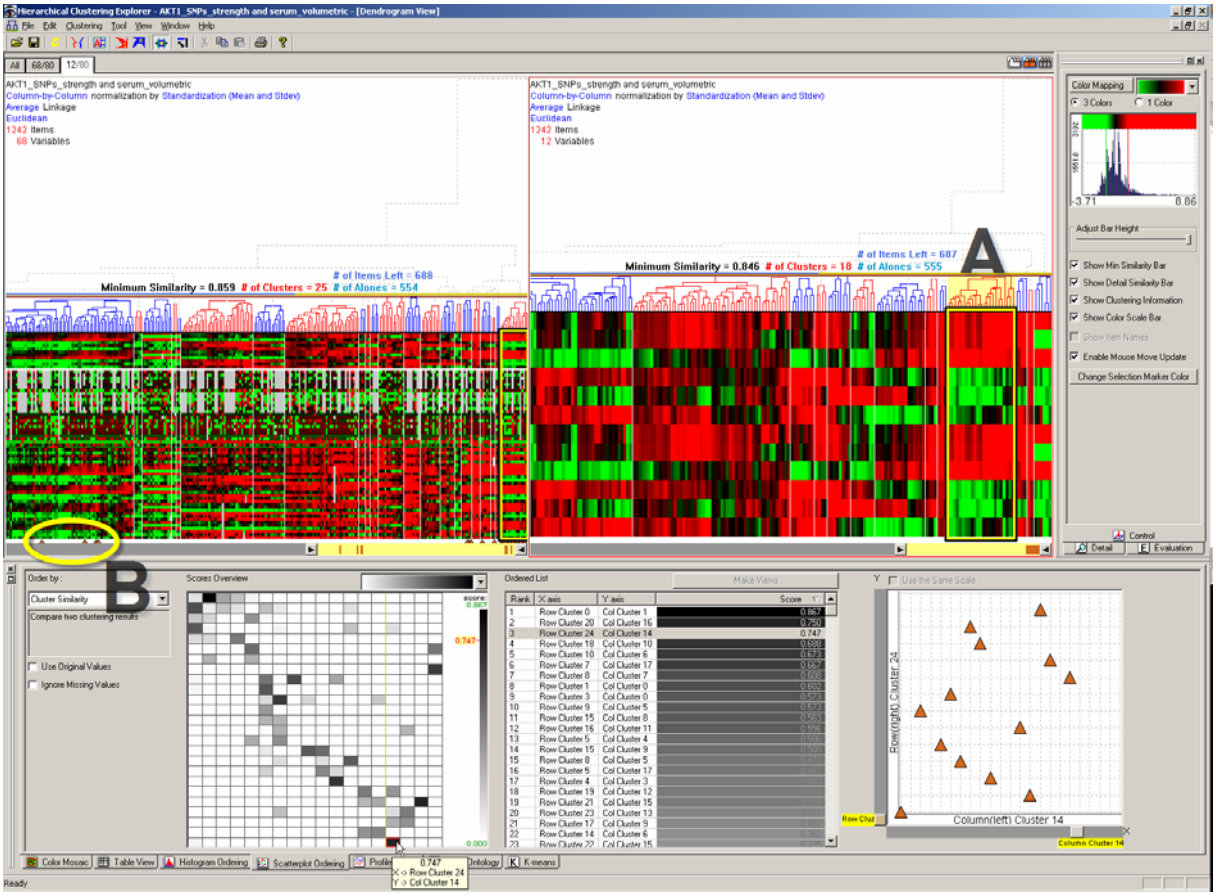


Figure 12. Cluster similarity ranking with a vertical partitioning (pre-exercise vs. post-exercise) of a SNP association dataset. On the top left is the clustering result of subjects with volumetric measurements of non-dominant arms before the exercise, and on the top right is the same after the exercise. A distinctively similar cluster pair is selected in the score overview. The user selects one of them with very high values in six measurements after exercise (A). Most of subjects in the cluster are high in the measurements before the exercise, but a few of them are not (B).

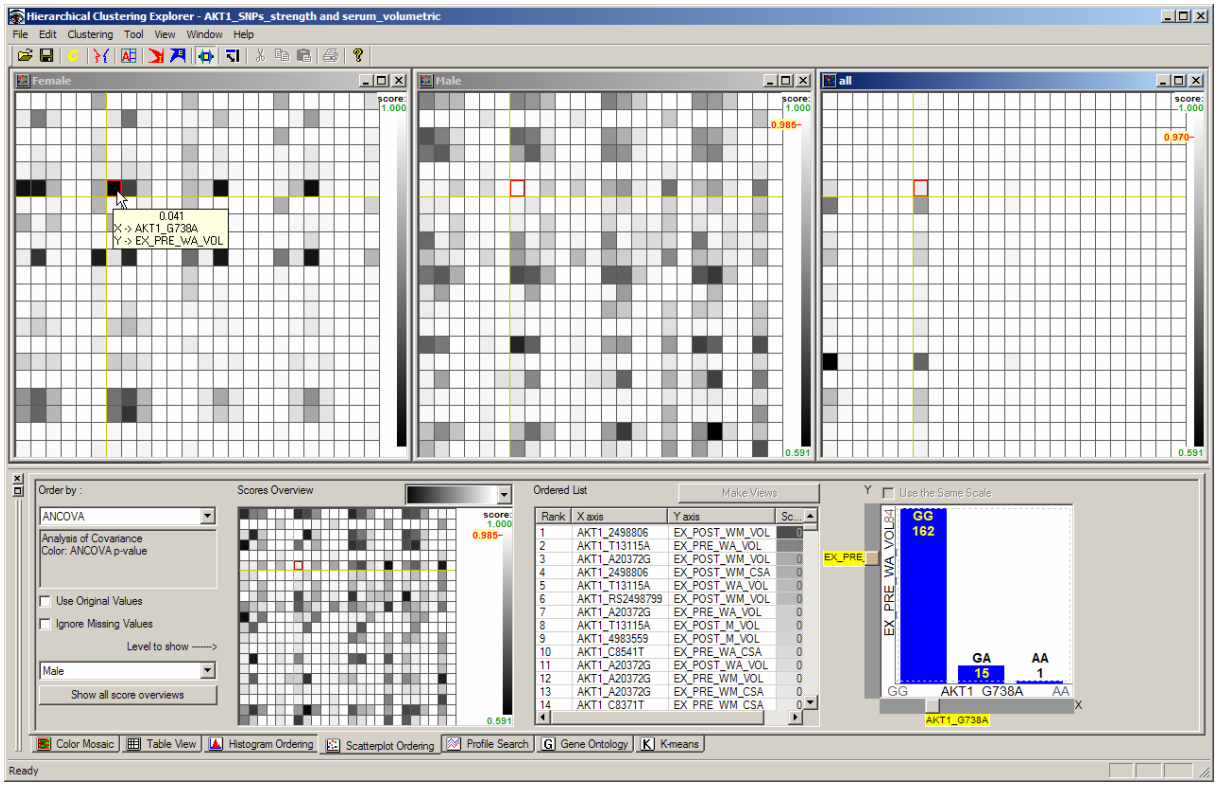
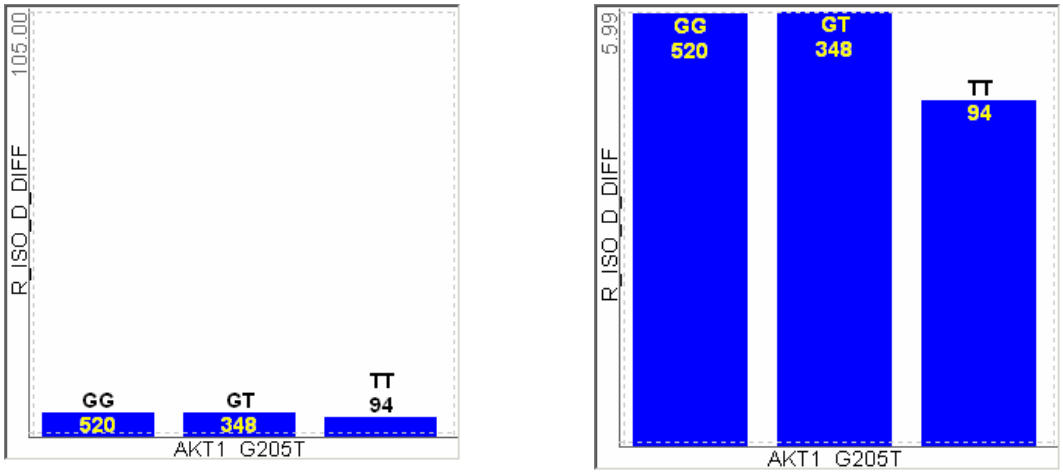


Figure 13. Ranking by ANCOVA (Analysis of Covariance) with stratification by gender. The three score overviews on the top row are for female, male, and the entire population, respectively. Each score overview shows significantly different score distributions. When users mouse over a cell on the top left overview, corresponding cells in other overviews are highlighted and the corresponding bar chart is shown at the bottom right corner.



(a) original scale (b) adjusted scale

Figure 14. Effect of scale in bar charts. The means in the left bar chart (a) do not look significantly different in the original scale, but the significance is apparent after the scale adjustment. GG, CT and TT represent genotypes and the number within the bar shows the sample size for each genotype.