Interactive Exploration of Multivariate Categorical Data: Exploiting Ranking Criteria to Reveal Patterns and Outliers

Darya Filippova

Abstract— Analyzing multivariate datasets requires users to understand distributions of single variables and at least the two-way relationships between the variables. Lower-dimension projection techniques may assist users in finding interesting combinations. To explore the 2D relationships in a systematic way, we suggest ranking such relationships according to some measure of interestingness. This approach has been valuable for continuous data; however, metrics for categorical data are a novel contribution. We propose CateRank a tool for analyzing categorical datasets which visualizes one-dimensional relationships as histograms and uses re-orderable matrix for two-dimensional relationships. CateRank implements several metrics based on the histogram and matrix properties that enable users to discover relationships between the two categorical variables. User controls support data filtering to remove extreme or uninteresting values.

Index Terms—Categorical data, systematic exploration, ranking, reorderable matrix.

1 Introduction

Large multivariate datasets are quite common today. Often, the dataset is a mix of continuous and categorical variables and the number of variables can be significant. US Census and credit approval records, road accidents, protein interaction data are a few examples of such datasets. To go through data similar to US Census data in a systematic way, users would first study single variables distribution, then it would be logical for them to study the two-dimensional relationships and then move up to the higher dimensions. However, if there are just 10 variables, there are $C(2,10) = 10!/8!2! = 45$ two-dimensional relationships, $C(3,10) = 10!/7!3! = 120$ three-dimensional relationships, etc. Users can become overwhelmed with the amount of visualizations they need to look at rather quickly.

Seo and Shneiderman [23] solve this problem by assigning a score to each such relationship thus making the relationships comparable. The users can choose more interesting relationships by looking at how the relationship ranks against the others. Seo and Shneiderman applied this approach in the Hierarchical Clustering Explorer (HCE) application and HCE proved a valuable tool for a variety of data including the Census and gene ontology data. However, the original version of HCE operated on continuous variables only; a later version [22] deals with categorical variables only in a limited way. In general, metrics for continuous data are not applicable to categorical data: categorical variables have no mean nor range, comparisons like *greater than* and *less than* are not defined for categorical data, and the mapping from categorical to continuous domain does not always work (nominal variables can not be converted into numbers without a loss of information).

In our work, we focused on ranking one- and two-dimensional categorical relationships. We visualize single variables through histograms and two-dimensional relationships are displayed as contingency tables. We apply a barycentric heuristic to each table to reorder rows and columns which results in a clustered view (see figure 1) and makes the table a reorderable matrix as developed by Siirtola and Makinen [24]. Our contributions are the ranking criteria for histograms and reorderable matrices that allow for systematic exploration of the categorical datasets.

• *Darya Filippova is with University of Maryland, E-mail: dfilippo@umd.edu.*

Manuscript received 31 March 2008; accepted 1 August 2008; posted online 19 October 2008; mailed on 13 October 2008.
For information on obtaining reprints of this article, please send e-mailto:tvcg@computer.org.

2 Related work

2.1 Multivariate datasets exploration

It is natural to browse through the data before attempting rigorous statistical analysis. In his book [29], Tukey says that "'it is important to understand what you CAN DO before you learn to measure how WELL you seem to have done it'". The techniques and methods summarized in the book make the data exploration easy and allow to look at the data from all sides. Earlier, Friedman and Tukey introduced a "'projection pursuit"' [9] concept: they suggested that the projections of the multivariate data be ranked to allow quantitative comparison among them. Assigning a score to each projection helps identify the most revealing combintations.

There is a plethora of commercial and open-source applications for multivariate data analysis, including: SAS [21], Spotfire [26], Tableau [20], ManyEyes [31], GeoVista [19]. Users have access to a variety of visualizations that accommodate both continuous and categorical variables. With the use of color, object size and shape, scatterplots, trellises, and small multiples views the users can study several variables at once. However, the space of all possible visualizations is large and the interesting structures in it tend to be sparse [15]. With nothing guiding them, the users are likely to stop upon finding a single interesting view and overlook other significant insights.

To guide the users towards more interesting combination of variables, Spotfire [26] provides a ranking mechanism that finds correlations between variables. Spotfire can handle many kinds of data relationships – numerical to numerical, numerical to categorical, and categorical to numerical, but for the categorical variables there is only a χ^2-test. χ^2-test only tests the presence of correlation, but tells nothing about its nature and strength.

Fekete et al. [6] take a different direction in multidimensional dataset exploration: they construct a scatterplot matrix and order its rows and columns according to a dimension similarity measure. Noting that row and column orderings are independent, they place similar columns next to each other and bring dissimilar rows closer together. Users then navigate through the scatterplots either stepping left and right to the next similar view or up and down to the most dissimilar view (only rectilinear movements are allowed). Interactive exploration allows users to become more familiar with a dataset; however, tasks such as identifying the two most dissimilar rows or finding a group of columns that are all similar to each other are hard.

Parallel coordinates [16] are another powerful visual display for analyzing multidimensional data. There are several issues to consider with the parallel coordinates: the data clutter, occlusion, and the axis arrangement. The data occlusion becomes especially prominent when plotting categorical variables. Occlusion can be mitigated by using transparent lines for each data record; overlapping lines will accumu-

late color intensity and become more prominent indicating that multiple data items are overlaid on top of each other. The clutter may be reduced by permuting the axes algorithmically, using random sampling or filtering [32], [5]. Finding a optimal permutation of axes that minimizes the clutter is hard (NP-hard); filtering and random sampling techniques that reduce the original dataset may actually obscure some patterns. Besides, it is hard to reason about the relationship between any non-consecutive axes.

Parallel Sets [18], a parallel coordinates adaptation for categorical data, uses frequency bands instead of a single line per data record to visualize the dataset. The frequency bands solve the occlusion problem, but the clutter and the axis arrangement are still a problem.

HCE [23], upon which the current work is built, stands out by providing rankings for every one- and two-dimensional relationship for continuous variables. A later version of HCE [22] was adapted to work with categorical variables and included histograms ranking, but, similar to Spotfire, did not provide ranking for two- or higher-dimensional categorical relationships other than the χ^2-test. Some work on ranking the categorical relationships was also done in Fervor [8].

2.2 Views for categorical data

Bar and pie charts are the common visualizations for single categorical variables in the same way the histograms are for the continouos data; their use dating back to at least 1700s [17]. Bar charts are easier to read than the pie charts (comparing items by height is visually easier than comparing by area) and are the most widely used way of visualizing a single variable distribution. In CateRank, we visualize single categorical variables as bar charts.

There is no single visualization that is commonly used to display a two-dimensional categorical relationship. In statistics, a pair of binary variables is usually given as a two-way table. The two-way table can be easily extended for the general case of the multivariate variables (multi-way, or contingency, table).

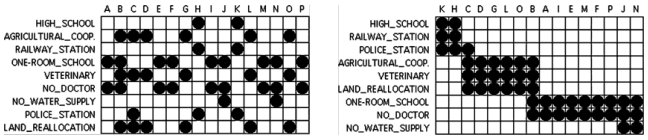

Fig. 1. Bertin's vote matrix.

Friendly [11] has introduced the fourfold displays for analysis of the 2x2 distributions; however, unlike the two-way table, the fourfold display can not be extended to work with multivariate variables. Mosaic plot, an adaptation of a contingency table concept to data visualization, was introduced by Hartigan and Kleiner [13] and further developed by Friendly [10]. Like the work by Fekete et al., mosaic plot lacks a quantifiable comparison of the relationships between variables and is hard to read when all rectangular areas are approximately the same size.

Upton [30] has suggested a cobweb diagram to visualize the relationship between the two categorical variables: the variable's values are placed as nodes on a circumference and are connected by lines across the circle. The thickness of the lines is proportional to the square of the adjusted differences between the actual frequencies in the multiway table and the frequencies in independent model. Since the values are laid out in a circle, the lines joining them result in a cluttered view which only increases as the number of values that the categorical variable takes grows.

A matrix in figure 1 is a simpler view for two-dimensional categorical relationship. The matrix was pioneered by Jacques Bertin [2] and further developed by Siirtola and Makinen [24], [25]. The matrix is a multiway table where the cells at the intersection of each row and column contain a count for the number of occurrences. The labels may be replaced by a circle or rectangle proportional in size to the original count; these nodes can also vary in color or transparency as in [24]. In a sense, the matrix is analogous to the scatterplot for continuous data. However, unlike the values on the axes in scaterplots, the rows and columns are independent, so one could permute them to bring the nonempty cells together. Figure 1, right, is an example of such permutation given by Bertin: before the matrix was sorted, it was impossible to identify the clusters within the data; once the rows and columns were permuted, the three clusters became prominent explaining the responses. Identifying clusters like these is an important part of the analysis and is similar to finding clusters on a scatterplot for two particular variables. Siirtola et al. suggested two algorithms for permuting the matrix: 2D Sort and a barycenter heuristic. 2D Sort produces different arrangements depending on the row that the matrix is first sorted by; in this sense, the barycenter heuristic is a more stable algorithm. The paper by Sugiyama et al. [28] contains a detailed description of the barycenter heuristic.

3 Interface

CateRank, a tool developed for the categorical data analysis, exploits the idea of guiding the users through the 1D and 2D categorical relationships by assigning each relationship a score (CateRank is available from `http://www.cs.umd.edu/hcil/caterank`). CateRank's major innovations are that it ranks categorical relationships based on geometrical and statistical properties and that it provides an interactive visualization that supports rapid exploration (fig. 2). As with the rank-by-feature framework [23], users start with exploration of single variables and later move on to visualizations for the pairs of variables. Switching between ranking criteria (area 1, Fig. 2) allows looking at one-dimensional variable distributions from different perspectives. In the same manner, the users explore two-dimensional relationships by going through their corresponding matrices and using ranking criteria to compare the relationships to each other (area 2, figure 3).

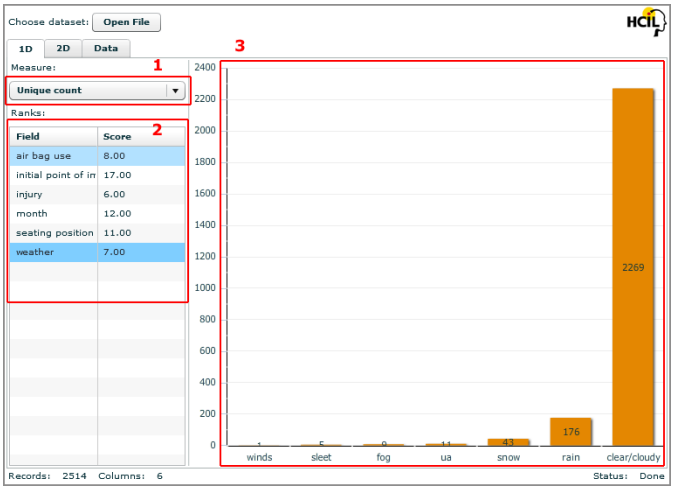

Fig. 2. One-dimensional tab with a list of rankings on the left (1) and the corresponding bar charts on the right (3). The weather variables had 7 unique values which corresponds to 7 bars on the chart sorted in an ascending order. The chart makes it clear that the majority of the accidents happened during dry weather. Users can sort the rankings by score or by the variable name (2).

To visualize one-dimensional relationships, CateRank uses bar charts. Two-dimensional relationships are visualized as reorderable matrices. In CateRank, the individual cells in the matrices are color-coded according to their count; such representation resembles a heatmap and highlights cells with the high counts. CateRank builds a matrix for each pair of variables in the data set. To reorder the matrix, we treat all non-zero values as 1's and run the barycenter heuristic on each table to find a permutation of rows and columns that yields a clustered view (see fig. 10, right). When the barycenter heuristic has run on all the tables, we rank the resulting reorderable matrices according to the default criteria (variance).

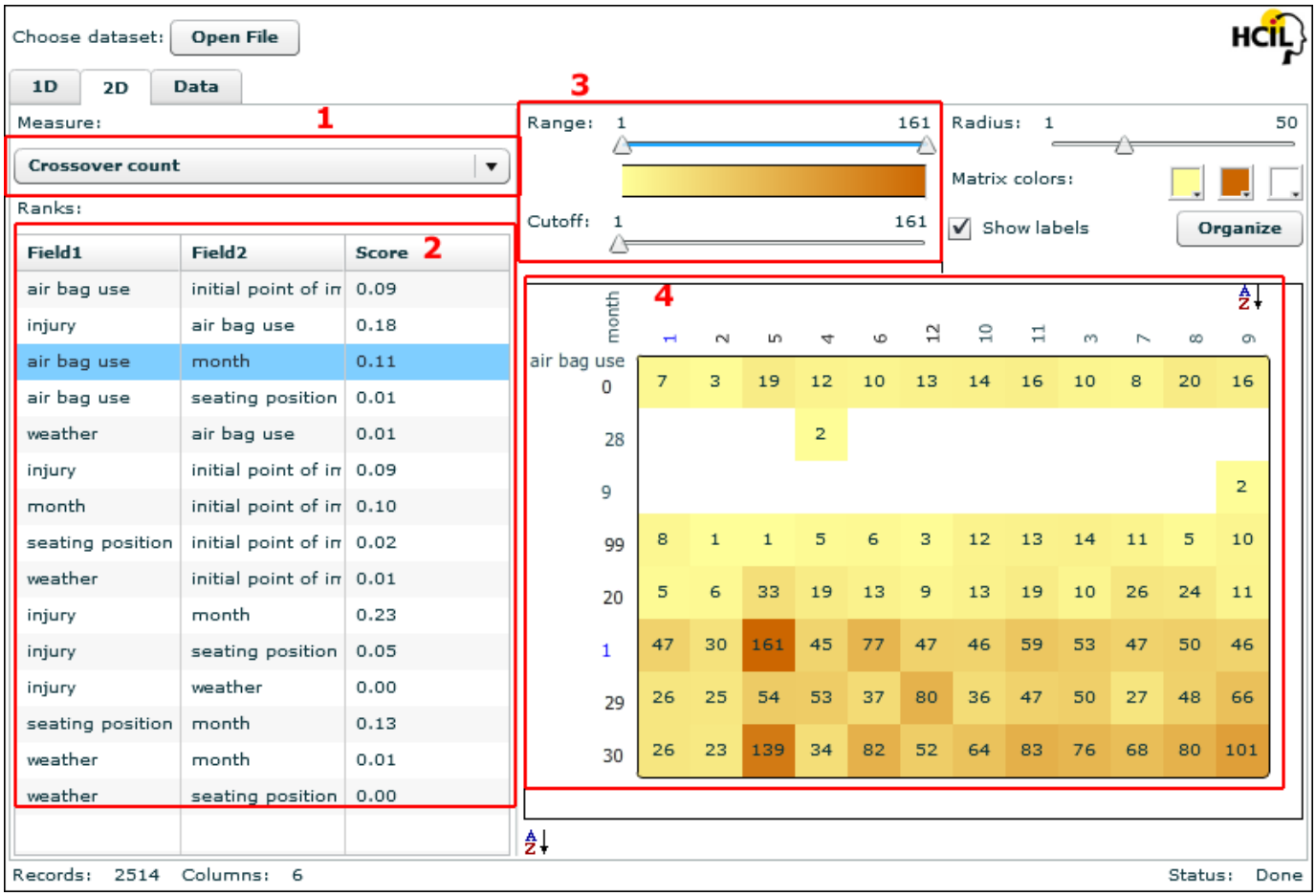

Fig. 3. Users can choose one of several two-dimensional rankings (1). The variables' relationships are listed on the left-hand side (2) together with their ranking scores. Each relationship is visualized as a reorderable matrix on the right (4). Users can look at all the counts in the range set by the range filter (3). The cutoff slider (3) allows users to set different cut-off values for the matrix that triggers a barycenter heuristic reordering on the matrix. Users have control over the color scheme and the size of the squares in the matrix.

3.1 Example walkthrough

3.1.1 Drug and health data

The National Survey on Drug Use and Health (NSDUH) collects data to estimate the level and patterns of drug abuse as well as to identify the groups at high risk for drug abuse. We have selected a recent 2008 survey [4] and filtered it to retain the respondents from 12 to 17 years old (school age) - this age group in particular is at high risk. Our dataset contains 23257 records randomly sampled from the original youth's responses. We have selected 10 variables to work with: age, race, health status, recency of use for the alcohol, cigarettes, marijuana, inhalants, hallucinogens, cocaine, and heroin. The dataset is available online [3].

Users start by exploring the individual variables. The age variable distribution tells the users that the majority (7860) of the respondents were 15 years old; the majority of the respondents had good to excellent health; an overwhelming 15327 identified themselves as white/Caucasian. The disturbingly large percentage of respondents (more than 50% - 13510 respondents) had had alcohol within the past 30 days of the survey date; while 6733 (29%) respondents have not ever tried smoking, 6897 (29.6%) of them had smoked in the past 30 days and another 9627 (41.4%) had smoked within the past 3 years (see fig. 4).

Now the question is how these variables relate to each other. The users switch to the "'2D"' tab (see fig 3) and start looking at the pairs of variables and their associated scores. The lowest variance score belongs to the (*age*, *health*) pair and various combinations of the recency of drug use variables (*rec_coc*, *rec_inh*, *rec_her*) score highest - a large

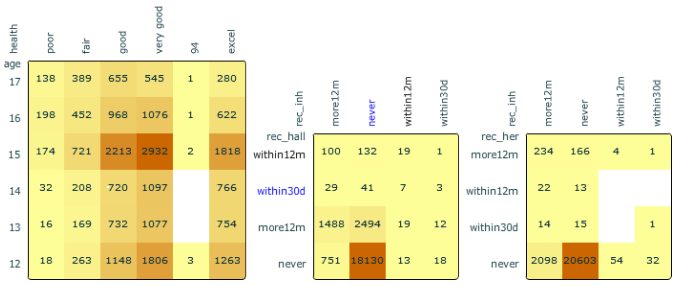

Fig. 5. Recency of alcohol use (left), recency of cigarette use (center), and recency of marijuana use (right)

number of the respondents have never tried either of those drugs (cocaine, inhalants, hallucinogens, heroin) resulting in a high count for "'Never tried"' (fig. 5).

However, if the users look at the pairs like (*rec_alc*, *rec_her*), (*rec_cig*, *rec_inh*) they will notice that even though the respondents mostly stayed away from cocaine, heroin, inhalants, or hallucinogens, they did not abstain from cigarettes, alcohol, or marijuana - the more popular and accessible substances (see fig 6).

The users can switch between various rankings to explore different properties of the matrices. The ranking for the amount of empty white space immediately draws attention to the pairs of variables one of which is the recency of the heroin use: the matrices for these pairs

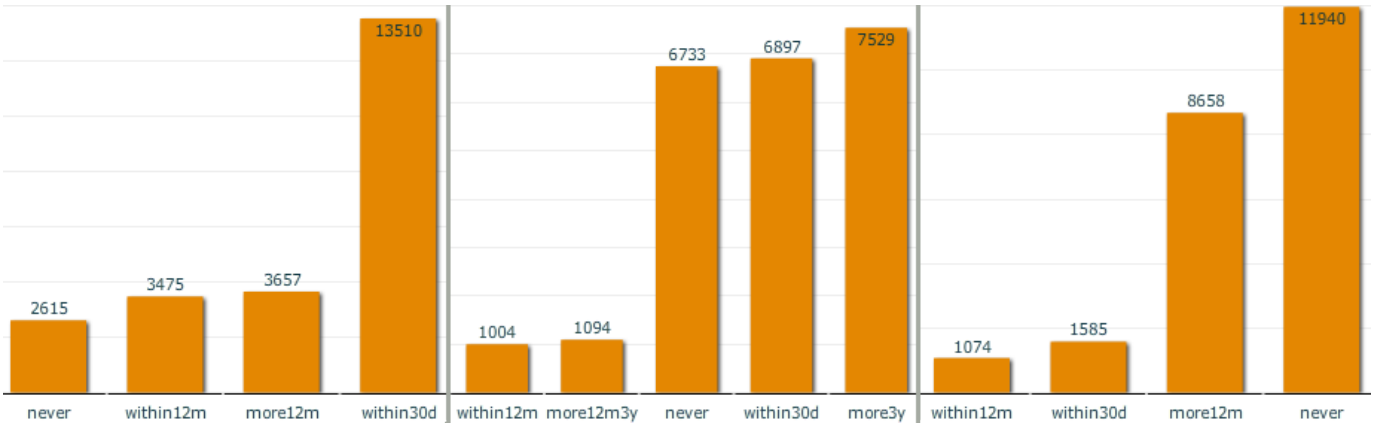

Fig. 4. Recency of alcohol use (left), recency of cigarette use (center), and recency of marijuana use (right)

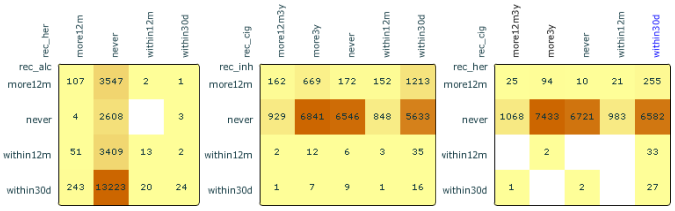

Fig. 6. The matrix for recency of alcohol use (rec_alc) and recency of the heroin use (rec_her) on the left shows that although the majority of respondents have not tried heroin ever, 86% of them have at some point tried alcohol. The matrices for recency of inhalants and heroin use as compared to the cigarette use show the same pattern: 60% and 68% of those that have never tried either subtance correspondingly have tried cigarettes.

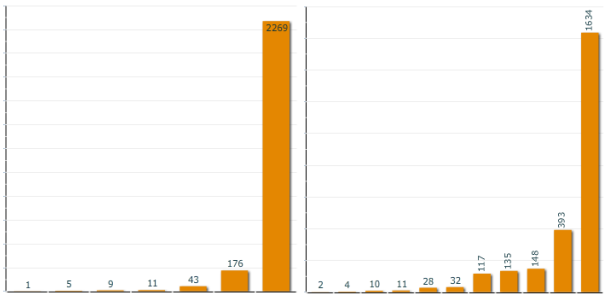

Fig. 7. *Weather* (left) and *seat position* variables received a high variance score that indicates that the data was aggregated in the few values.

are filled out only partially which, firstly, is a result of a low number of the respondents who tried heroin. Secondly, it may be indicative of a vicious circle with cigarettes and alcohol serving as the gateway drugs: from over 400 respondents who have tried heroin only 12 (less than 3%) have never smoked a cigarette.

3.1.2 Accident data

The accident data set is a set of fatal road crashes in the state of Maryland in the year 2007 as provided by the FARS [7] encyclopedia. The dataset contains a total of 2223 items. We have focused on the variables such as the data and time of the crash, weather conditions, car type, airbag deployment, seating position for the injured person, the point of impact direction. The dataset is available online [3].

As in the previous example, users start by exploring the one-dimensional relationships. A few high-variance variables stand out: *weather*, *seat position*, and *point of initial impact*. These bar charts indicate that the majority of the accidents happen on the clear days; people using the front seats are involved in fatal accidents more often than people sitting elsewhere in the car; the overwhelming majority of the fatal accidents (1433 out of 2223) were the head on collisions followed by the side impact collisions (see figure 7). These observations are partially confirmed by the uniformity score: the *weather* variable has the lowest score of (0.52) which indicates the the dataset is skewed. Most uniformly distributed variables are *month* and *hour* of the accident while the *day of week* variable's score is in between (2.80), therefore, *day of week* variable may display some bias.

Moving on to the two-dimensional relationships, users start by exploring the views by the variance score. The pairs having *hour* as one of the variables get low variance: accidents are more or less evenly distributed throughout the day. The users saw that head on collisions were the most common from the *point of impact* distribution. Now, looking at the (*weather*, *point of impact*) matrix, users see that the head on collisions happened the most when the weather was fine. So what explains such a large percentage of the head on incidents? Looking at the (*hour*, *point of impact*) matrix, users observe that the majority of the head on collisions happened around the evening rush hour time: 5 - 7 p.m. An interesting deviation form the pattern is 103 head on collisions at 2p.m.: if we look at the (*day of week*, *hour*) matrix, we notice that the majority of those 2p.m. collisions happened on Fridays and may be explained by the increased traffic when people start leaving jobs earlier on Friday. Accidents at the odd times, such as midnight - 3a.m. happen mostly on Saturday and Sunday - times when people go out to the bars and restaurants and tend to forget the driving safety rules.

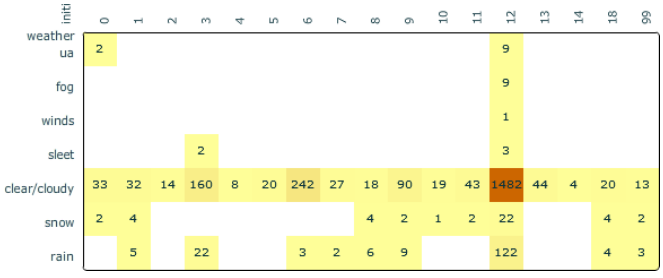

Fig. 8. Rows: weather, columns: point of impact. (*weather*, *point of impact*) matrix displays a large cluster of accidents that happened on the clear days. The majority of those were head on collisions (point of impact value of 12).

Users then switch to the outlier ranking and study the matrices with high number of outliers. One of them, (*weather*, *body type*), reveals that sedans are involved in fatal crashes during the clear weather conditions more often than other vehicles (766 crashes). This fact may be

explained by the number of sedans driven - it is the most common type of a car on the roads and, therefore, if a car is involved in an accident, it is more likely that the car is a sedan. Filtering out the sedans with the range slider, users observe that pickups, utility vehicles, and motorcycles follow the sedans in the number of fatal crashes (229, 208, 181).

As the number of views that users have to analyze grows, it becomes increasingly important to provide users with some guidance. Ranking the views is one such alternative. Below we discuss the one- and two-dimensional categorical relationships in more detail.

3.2 Other controls

Below we describe the additional tools available to users allowing them to control the visual displays.

3.2.1 Range slider

The range slider allows users to set the range of values that they want to display in the matrix. The slider allows them to cut-off extreme values bringing the focus to the mid range, or, conversely, focus on the outliers by filtering the mid range values out.

When users start to drag the slider tick controlling the maximum displayed value, CateRank adjusts the distribution of color in the matrix. This way users can observe the emergence of the secondary high values that were otherwise obscured by the maximum counts.

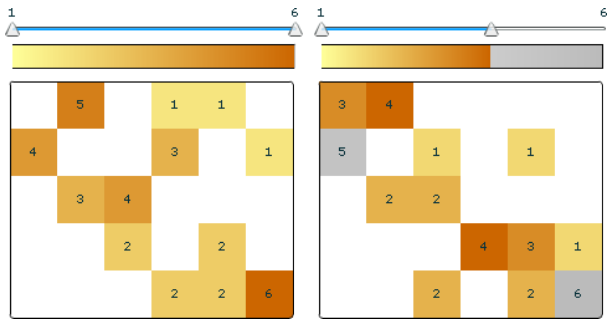

Fig. 9. Different views of the same matrix: on the left, the range slider thumbs are set at 0 and 6 which includes all counts. On the right, the thumbs are set at 0 and 4: all values above 4 are grayed out helping the user concentrate on the filtered values.

3.2.2 Cutoff slider

By default, the algorithm for reordering the matrix considers all non-zero counts as 1. This way, when calculating the weighted sums for rows and columns (see [28] for the details of algorithm implementation), all non-zero values make the same contribution to the sum. The cutoff slider allows users to change the cutoff value: if the cutoff is set to x, then only the values that are greater than x would contribute to the weighted sums. Moving the cutoff slider provides a new clustering for each cutoff value (see Figure 10).

3.2.3 Other controls

Among other controls available to users is the color picker that gives users an instrument to adjust the color scheme for the matrix. The radius slider changes the radius of individual counts in the matrix view: moving the slider, users can zoom in on a matrix or zoom out to get an overview of a particularly large matrix.

3.3 Implementation details

CateRank was developed in Flex/ActionScript3 and the sample datasets we ran on the 2-core AMD Turion 64 1.90 GHz.machine with 3Gb of RAM.

We ran CateRank on several datasets to get a range for run times; table 1 contains the datasets details. *Cars* dataset came from the CMU's StatLib [27] and describes car models produced by major car manufacturers in USA, Europe, and Japan from 1970 to 1980. The columns

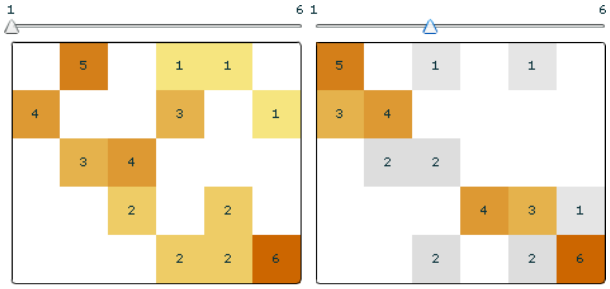

Fig. 10. When the cutoff value is set to 1 (left), all non-zero counts are taken into account when reordering the matrix. Toggling the cutoff slider triggers the reordering with the counts less than or equal to the slider's value being ignored. The matrix to the right is the same matrix but reordered with the counts below 4 ignored.

are miles per gallon (MPG), number of cylinders, horsepower, weight, time to accelerate to 100 MPH, manufacturer, origin and model year. *Accidents1* dataset contains records on the accidents that happened during 2004-2006 period on one of the busy streets in a city in the State of Maryland as reported by State Police. The columns include collision type, harmful event, lighting conditions and weather, road surface type (paved, gravel, etc.), road condition (ice, mud, snow, etc.), road division type (undivided, double yellow line division, has a median, etc.). Credit approval dataset comes from the UCI Machine Learning Repository [1]. We removed real- and integer- valued columns which left us with 9 categorical fields. *Accidents2*and *accidents3* datasets are collections of all fatal accident records in 2007 for the State of Maryland as reported by FARS [7]. *Accidents3* contains the same data records as *accidents2* but includes different columns with a high unique value count. The size of the largest matrix for *accidents3* explains the difference in running times between the two datasets. The columns include first and most harmful events, injury, accident month, weather, initial point of impact, air bag use, and the seating position of the fatally injured occupant.

Name	# Records	# Columns	Matrixx	Time, ms
cars	406	9	13x30	11906
accidents1	504	10	10x17	4519
credit approval	690	9	8x14	1309
accidents2	2223	10	24x29	27314
accidents3	2223	8	34x35	34318

Table 1. Datasets details. The size of the largest matrix, number of columns, and number of records all affect the time required for the initial CateRank setup.

4 Rankings

The rankings guide the users through the individual variables and their combinations. The rankings allow users to choose views that display more "'interesting"' features - be it the variance, outliers, or number of unique values.

4.1 1D ranking

Variance and uniformity rankings are the discrete equivalents of those used in HCE [23], [22] for continuous data and the rest are the novel rankings.

4.1.1 Number of unique values

Number of unique values is a natural metric to consider when working with categorical data. CateRank show the number of unique values for each variable in a sortable table (figure 11).

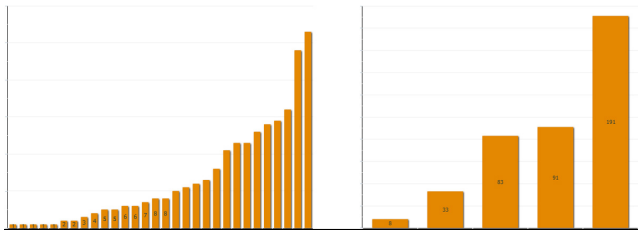

Fig. 11. Number of unique values: variable on the left has 30 values and the variable on the right has only 5 distinct values.

4.1.2 Variance

Variance provides users with a measure of how spread the distribution is for some variable x. We assume that the variable x takes a particular value i with equal probability $p = \frac{1}{n}$ where n is the number of unique values x can take. Then the expected value for x is

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} = \frac{N}{n}$$

where x_i is the number of times the variable takes the value i and N is the total number of records in the dataset. For each variable x, the variance is calculated by the formula:

$$Var(x) = \sum_{x_i} p \cdot (x_i - \bar{x})^2$$

Figure 12 shows an example of high and low variance histograms.

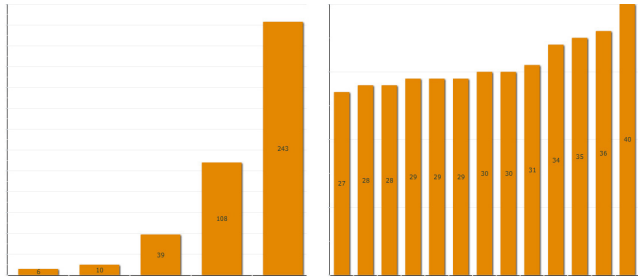

Fig. 12. A variable on the left has a higher variance than the variable on the right.

4.1.3 Uniformity of the distribution

Uniformity of distribution ranking is the entropy measure as in [23]. The distribution with the higher entropy is closer to a uniform distribution and the variable's histogram looks like figure 13, left. The low entropy value indicates that the distribution is non-uniform (skewed); figure 13, right, is an example of such a histogram. If the variable x takes n different values with each value appearing in x_i records in the dataset, then

$$H(x) = \sum_{i=1}^{n} \frac{x_i}{N} \log \frac{x_i}{N}$$

is the x's uniformity score.

4.1.4 Number of outliers

We use the same principle for calculating the outliers as HCE [23] does; however, to calculate the interquartile range we had to use the counts per variable's value. First, we obtain the counts for the variable's unique values, sort the counts in an ascending order, and calculate first (Q1) and third (Q3) quartiles. From that, we derive the interquartile range (IQR) and search for the values that have counts above $Q3 + 1.5 * IQR$ or below $Q1 - 1.5 * IQR$. Such values are considered outliers. Figure 14 demonstrates the concept.

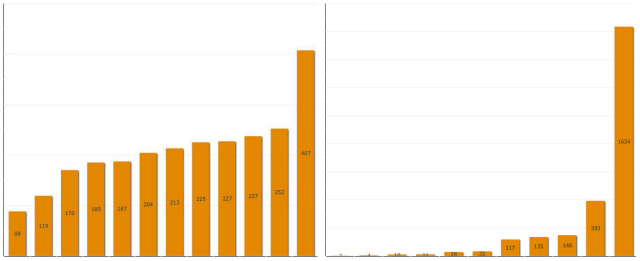

Fig. 13. Accident dataset: month of accident bar chart got the highest 3.50 score in entropy ranking (left) and low entropy of 1.74 (right) for the seat position.

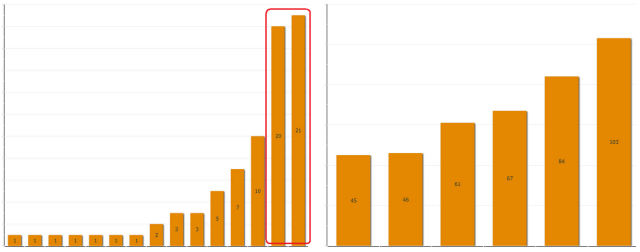

Fig. 14. Number of outliers: 2 outliers on the left and 0 on the right.

4.2 2D ranking

Two-dimensional rankings describe the geometrical properties of the matrices corresponding to the pairs of variables as well as the matrices' quantifiable properties.

4.2.1 Variance

We calculate the variance for the pairs of variables x, y in a manner similar to the one-dimensional case. We treat the pair's counts as a one-dimensional variable with nm values (assuming x takes n values and y - m values) and naively assume that the variable has a uniform distribution. Then the variance for x,y is

$$Var(M) = \sum_{i,j} (a_{ij} - \frac{N}{nm})^2$$

Figure 15 contains an example of high and low variance matrices.

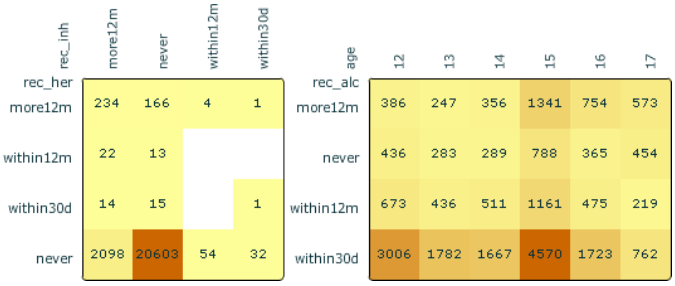

Fig. 15. A high variance matrix on the left on the left has a single count of 20603 aggregating the majority of items in the dataset. The counts in the lower variance matrix on the right are distributed more evenly.

4.2.2 Crossover

The barycenter heuristic tries to re-order rows and columns in the bipartite graph corresponding to the (x,y) matrix to minimize the crossovers (see [24]). The crossover ranking indicates how well the barycenter heuristic was able to reorganize the bipartite graph. A crossover score closer to 0 means that there are much fewer edges

between the vertices of the bipartite graph that is theoretically possible, in other words, it is more likely that the bipartite graph, and the matrix, has segmented into separate clusters.

The crossover score is a ratio between the actual number of crossovers and the maximum number possible for a given graph:

$$\frac{e}{e_{max}}$$

where e is the number of edges in the bipartite graph corresponding to the matrix and

$$e_{max} = \frac{mn(m-1)(m-1)}{4}$$

is the number of edges in the full bipartite graph (two sets of vertices are connected by every possible edge), m, n as above.

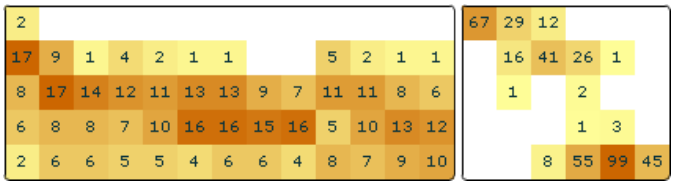

Fig. 16. High crossover score matrix (left) is dense compared to the low crossover score matrix (right).

4.2.3 Uniformity of distribution

Uniformity score is calculated similarly to the variance score: we treat the matrix as a single variable with $n \cdot m$ counts and compute its uniformity as follows:

$$H(X) = \sum_i p_i \log p_i$$

where $p_i = \frac{a_{ij}}{\sum_{ij} a_{ij}} = \frac{a_{ij}}{N}$.

4.2.4 Non-null values

A non-null measure tells the user the amount of the white space in matrix display. This measure indicates how values are spread in the table. The low score for this ranking would indicate that all items are concentrated around the few pairs of values. The matrix on the left in figure 16 gets a higher non-null values score than its neighbor instantly tells the users that the left matrix is almost all filled in.

4.2.5 Null values

Null values measure is an inverse of the non-null values measures; in other words, it is the a ratio of the number of null counts to the total number of counts (nm) in the matrix.

4.2.6 Number of outliers

CateRank applies the outlier detection method in 1D to 2D relationships. To calculate the number of outliers in 2D, CateRank merges the counts from all matrix rows, sorts them in the ascending order and then applies the outlier measure to that array.

These measures do not describe every possible aspect of the variable or relationship distribution, but serve as a good starting point for the further analysis.

5 Discussion

While variables with a large number of values produce complex and interesting matrices, the matrices produced for the binary variables are simple 2x2 tables with little to explore visually. Sometimes the binary values can be merged into a single variable as it was done for the voting example in figure 1. The matrix in 1 reflects only the "yes" votes; however, one may look at the votes against as well.

Continuous variables are often converted into categorical through binning. For example, time of day can be easily aggregated by the hour

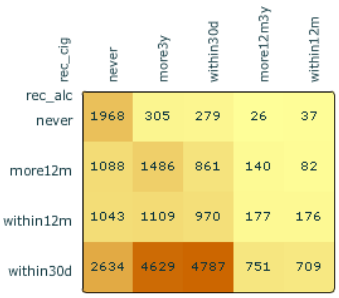

Fig. 17. Matrix with two outliers: from all respondents, 4629 (19.7%) have consumed alcohol in the past 30 days while additional 4787 (20.4%) not only drank, but also smoked within 30 days of the survey.

or into four generic categories morning, afternoon, evening, night; age can be broken down into the age groups, etc.

For the smaller datasets, CateRank aggregates data and generates matrices instantly with a hardly noticeable delay. However, as the number of variables grows, the number of pairs increases exponentially, therefore, increasing the runtime. For larger datasets, the barycenter heuristic becomes a bottleneck since it has to converge for every matrix. In turn, the barycenter heuristic is dependent on the matrix size (note the difference in runtime for *accidents2* and *accidents3* datasets). We are investigating ways to speed up the barycenter heuristic to accommodate for larger matrices and larger datasets.

We asked four users to complete a series of tasks using CateRank during an informal study. We described the main UI parts, their purpose, and the dataset to the users. After the introduction, the users were given unlimited time to complete the tasks. Two users were Masters students (computer science, education); two had Bachelor degrees (computer science, engineering). All participants were experienced computer users. The users provided correct answers for most questions: on average, users answered correctly 8 out of 10 questions. However, the cutoff and range sliders were the source of confusion for two users leaving them wondering why the rows and columns got reordered when they moved the cutoff slider. Two other users used the range slider successfully ("it hides unnecessary information", "it was very useful to find four largest counts"). To complete the tasks, users had to use rankings for 1D and 2D relationships and the users reported that rankings helped them to gain some interesting insights (for example, using the outlier ranking, the user found that "Ford" produced more car models than any other manufacturer).

Based on these observations, it is clear that users needed a more thorough introduction that would explain the concept of the reorderable matrix in more detail. The sliders were not the main focus of our attention, but the responses we got make us think that a more extensive introduction would be beneficial to the users as well. Overall, the users found the whole experience pleasant and completed the tasks "with ease". Users also appreciated the fact that the matrices (especially large ones) were initially reordered allowing the participants to focus on the analysis.

6 Conclusions and Future Work

CateRank is a useful tool for an initial analysis of the categorical data and can provide insight about the datasets. However, the informal user observation made it obvious that CateRank needs additional work on rankings to provide the users with a powerful set of tools.

Further investigation into the ranking measures can continue in the direction of estimating the strength of variables' correlation. It would be interesting to devise a measure that calculates the number of geometrical clusters in 2D matrices (as the three clusters in figure 1, right); however, this ranking highly depends on the definition of clusters. As we have observed with many datasets, it may be difficult to draw the actual borders for a cluster unless some counts are ignored. It is also necessary to add the standard measures used in statistics for estimating the correlation between the categorical variables such as χ^2 test,

Cramer's V. It may also be useful to incorporate the information on the expected counts for each cell in the matrix and to color the cells according to the amount of deviation from the cell's expected value.

During the trial runs with the users, we observed that often the users wanted to see the percentages instead of the counts. Percentages will help the users estimate the contributions of a particular value in the variable's distribution.

It will be helpful to equip the matrices with counts along the rows and columns so that the users could see the counts for the individual variables making up the matrix. Making the rows and columns of the matrix sortable by these aggregate counts would give the users more control over the matrix.

The real-world data rarely comes as purely categorical or purely numerical. It would be an interesting and challenging task to analyze numerical-to-categorical relationships and come up with the meaningful rankings for them.

Following the example of HCE, it may prove useful to combine ranking criteria with categorical clustering algorithms. Several such algorithms have been used successfully in the area of machine learning, for example, categorical k-means [14] and categorical subspace clustering [12].

Acknowledgements

The author wishes to thank Ben Shneiderman for encouraging the project's development and supervising its progress as well as reading through the editions of this paper. The author would also like to extend gratitude to Michael Pack for providing early feedback on CateRank and to Krist Wongsuphasawat for his thoughtful comments on the tool and the paper.

References

[1] A. Asuncion and D. J. Newman. UCI machine learning repository, 2007.

[2] J. Bertin. Matrix theory of graphics. *Information Design Journal*, 10(1):5–19, 2001.

[3] Caterank - interactive exploration of multivariate categorical data, 2009.

[4] National survey on drug use and health, 2008, November 2009.

[5] G. Ellis and A. Dix. Enabling automatic clutter reduction in parallel coordinate plots. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):717–724, 2006.

[6] N. Elmqvist, P. Dragicevic, and J.-D. Fekete. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *Visualization and Computer Graphics, IEEE Transactions on*, 14(6):1539–1148, Nov.-Dec. 2008.

[7] FARS Encyclopedia. `http://www-fars.nhtsa.dot.gov/Main/index.aspx`, January 2009.

[8] D. Filippova, A. Olea, K. Wongsuphasawat, M. VanDaniker, and M. Pack. Visual analytics for transportation incident datasets. In *2009 TRB 88th Annual Meeting: Compendium of Papers*, 2009.

[9] J. H. Friedman and J. Tukey. A projection pursuit algorithm for exploratory data analysis. *IEEE Trans. Computers*, C-23, 1974.

[10] M. Friendly. Mosaic displays for multi-way contingency tables. *Journal of the American Statistical Association*, 89:190–200, 1994.

[11] M. Friendly. A fourfold display for 2 by 2 by k tables. Technical Report 217, Psychology Department, York Univesity, Toronto, Canada, 1995.

[12] G. Gan and J. Wu. Subspace clustering for high dimensional categorical data. *SIGKDD Explor. Newsl.*, 6(2):87–94, 2004.

[13] J. A. Hartigan and B. Kleiner. Mosaics for contingency tables. *Computer Science and Statistics: Proceedings of the 13th Symposium on the Interface*, 1981.

[14] Z. Huang. Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Min. Knowl. Discov.*, 2(3):283–304, 1998.

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

[16] I. Inselberg. The plane with parallel coordinates. *The Visual Computer*, 1(2):69–91, 1985.

[17] Y. Ioannidis. The history of histograms (abridged). In *VLDB '2003: Proceedings of the 29th international conference on Very large data bases*, pages 19–30. VLDB Endowment, 2003.

[18] R. Kosara, F. Bendix, and H. Hauser. Parallel Sets: interactive exploration and visual analysis of categorical data. *Visualization and Computer Graphics, IEEE Transactions on*, 12(4):558–568, July-Aug. 2006.

[19] A. MacEachren, D. Xiping, F. Hardisty, D. Guo, and G. Lengerich. Exploring high-d spaces with multiform matrices and small multiples. In *Proceedings of the International Symposium on Information Visualization*, pages 31–38, Oct. 2003.

[20] J. Mackinlay, P. Hanrahan, and C. Stolte. Show Me: Automatic presentation for visual analysis. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1137–1144, 2007.

[21] SAS. `http://www.sas.com/`, January 2009.

[22] J. Seo and H. Gordish-Dressman. Exploratory data analysis with categorical variables: An improved rank-by-feature framework and a case study. *International Journal of Human-Computer Interaction*, 23(3):287–314, 2007.

[23] J. Seo and B. Shneiderman. A rank-by-feature framework for interactive exploration of multidimensional data. *Information Visualization*, 4(2):96–113, 2005.

[24] H. Siirtola and E. Mäkinen. Reordering the reorderable matrix as an algorithmic problem. In *Diagrams '00: Proceedings of the First International Conference on Theory and Application of Diagrams*, pages 453–467, London, UK, 2000. Springer-Verlag.

[25] H. Siirtola and E. Mäkinen. Constructing and reconstructing the reorderable matrix. *Information Visualization*, 4(1):32–48, 2005.

[26] Spotfire. http://spotfire.tibco.com, January 2009.

[27] StatLib. `http://lib.stat.cmu.edu/`, January 2009.

[28] K. Sugiyama, S. Tagawa, and M. Toda. Methods for visual understanding of hierarchical system structures. *Systems, Man and Cybernetics, IEEE Transactions on*, 11(2):109–125, Feb. 1981.

[29] J. W. Tukey. *Exploratory Data Analsys*. Addison-Wesley, 1977.

[30] G. J. G. Upton. Cobweb diagram for multiway contingency tables. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 49(1):79–85, 2000.

[31] M. Wattenberg, J. Kriss, M. McKeon, F. B. Viegas, and F. Van Ham. ManyEyes: a site for visualization at internet scale. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1121–1128, 2007.

[32] K. Zhao, B. Liu, T. M. Tirpak, and A. Schaller. V-Miner: using enhanced parallel coordinates to mine product design and test data. In *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 494–502, New York, NY, USA, 2004. ACM.