GPLOM: The Generalized Plot Matrix for Visualizing Multidimensional Multivariate Data

Jean-François Im, Michael J. McGuffin, and Rock Leung

Abstract—Scatterplot matrices (SPLOMs), parallel coordinates, and glyphs can all be used to visualize the multiple continuous variables (i.e., dependent variables or measures) in multidimensional multivariate data. However, these techniques are not well suited to visualizing many categorical variables (i.e., independent variables or dimensions). To visualize multiple categorical variables, "hierarchical axes" that "stack dimensions" have been used in systems like Polaris and Tableau. However, this approach does not scale well beyond a small number of categorical variables. Emerson et al. [8] extend the matrix paradigm of the SPLOM to simultaneously visualize several categorical and continuous variables, displaying many kinds of charts in the matrix depending on the kinds of variables involved. We propose a variant of their technique, called the Generalized Plot Matrix (GPLOM). The GPLOM restricts Emerson et al.'s technique to only three kinds of charts (scatterplots for pairs of continuous variables, heatmaps for pairs of categorical variables, and barcharts for pairings of categorical and continuous variable), in an effort to make it easier to understand. At the same time, the GPLOM extends Emerson et al.'s work by demonstrating interactive techniques suited to the matrix of charts. We discuss the visual design and interactive features of our GPLOM prototype, including a textual search feature allowing users to quickly locate values or variables by name. We also present a user study that compared performance with Tableau and our GPLOM prototype, that found that GPLOM is significantly faster in certain cases, and not significantly slower in other cases.

Index Terms—Multidimensional data, tabular data, relational data, mdmv, high-dimensional data, database visualization, database overview, parallel coordinates, scatterplot matrix, user interfaces, business intelligence

1 Introduction

Many datasets are stored in tabular form, with one row for each tuple, and one column for each attribute. If the attributes are *dependent variables* (e.g., dependent variables of a key or row id), we speak of *multivariate* data, for which many techniques exist for visualizing several variables at once, such as scatterplot matrices (SPLOMs) [11], parallel coordinates [14], and glyphs [1, 3, 16, 22]. Some of the columns, however, may be best thought of as *independent variables*, in which case we speak of multidimensional multivariate (mdmv) data [38]. Stolte et al. [29] use the term *dimension* for a (categorical or ordinal) independent variable, and *measure* for a dependent variable. We will refer to dimensions as categorical variables, and measures as continuous variables.

The aforementioned techniques, of SPLOMs, parallel coordinates, and glyphs, all suffer from problems when naively applied to datasets with many categorical variables. An alternative approach involves "stacking" multiple categorical variables along axes, used in trellis charts and [17, 19, 29] and more recently in the commercially successful product Tableau [18]. However, dimensional stacking suffers from a combinatorial explosion if too many categorical variables are displayed at once.

Recent work [8] offers a new solution for visualizing mdmv data, based on the observation that SPLOMs need not display scatterplots for all pairs of variables. A plot matrix could instead display different charts for different pairs of variables, which Emerson et al. [8] demonstrated with a wide variety of charts. We adapted this idea with our own technique called the Generalized Plot Matrix (GPLOM). In our approach, the visualization is simpler than in [8], as we use only three kinds of charts, chosen with rules similar to those in [18]: scatterplots for pairs of continuous variables, barcharts to show a continuous variable as a function of a categorical variable, and heatmaps to show some selected continuous variable as a function of a pair of categorical variables. These three charts are the minimum number necessary to cover the three possible pairings of variable types. This makes the matrix easier to understand, which could be beneficial to casual business users and other non-expert users. At the same time, we extend part of Emerson et al. [8]'s work by presenting interactive features for highlighting, selecting, searching, and filtering the data.

Both Emerson et al. [8]'s technique, and our own GPLOM, can comfortably display several categorical and continuous variables at once, avoiding the combinatorial explosion of dimensional stacking because the data can be aggregated within each chart. This makes these approaches appropriate for data with multiple categorical variables, as is common in business intelligence and other domains. These approaches can also provide the initial overview of a database shown to a user, serving as a visual launching point for further investigation. This is in contrast to the approach in Polaris [29] or Tableau [18], where the user must first select one or several variables of interest to explicitly construct a visualization. Finally, for non-expert users, the GPLOM approach has the advantage of only using three kinds of charts, avoiding the more complicated charts such as mosaic plots or box plots that may be difficult for non-expert users to understand and that don't scale as well to high cardinality variables.

Our contributions are (1) the GPLOM technique for visualizing multidimensional multivariate data using only three kinds of charts, making it as easy to understand as possible while still showing charts that are adapted to the kinds of variables involved; (2) a description of the visual design choices and features of our prototype implementation, including bendy highlights, associative highlighting, and a text search feature that highlights data, allowing users to quickly find charts of interest; and (3) an experimental comparison of GPLOM and Tableau that found GPLOM to be significantly faster in certain cases.

2 Related Work

Surveys of techniques for visualizing mdmv data can be found in [38, 9, 15]. We will consider the most relevant of these techniques, and consider a fictitious "nuts-and-bolts" dataset to illustrate some differences between previous work. The nuts-and-bolts data is stored as a table, and involves 3 (independent) categorical variables: Region (North, Central, or South), Month (January, February, ...), and Product

- *Jean-François Im is with École de technologie supérieure, Montreal, Canada. E-mail: jfim@jean-francois.im.*
- *Michael J. McGuffin is with École de technologie supérieure, Montreal, Canada. E-mail: michael.mcguffin@etsmtl.ca.*
- *Rock Leung is with SAP, Vancouver, Canada. E-mail: rock.leung@sap.com.*

Manuscript received 31 March 2013; accepted 1 August 2013; posted online 13 October 2013; mailed on 4 October 2013.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

(Nuts or Bolts). It also involves 3 (dependent) continuous variables: Sales, Equipment costs, and Labor costs. There are $3 \times 12 \times 2 = 72$ combinations of categorical values, each corresponding to a row in a table, and each mapping to values of the continuous variables:

Region	Month	Product	Sales	Equipment costs	Labor costs
North	Jan	Nuts	2.76	0.92	4.30
North	Jan	Bolts	4.92	1.64	4.30
North	Feb	Nuts	4.20	1.00	4.30
North	Feb	Bolts	8.40	2.00	4.30
North	Mar	Nuts	5.28	9.60	4.30
⋮	⋮	⋮	⋮	⋮	⋮
South	Dec	Bolts	9.50	2.44	5.20

TableLens [24] and FOCUS [26] (later renamed InfoZoom) provide ways to aggregate the tuples in a list such as the one above, while still presenting an essentially tabular view to the user. Both systems allow the user to sort tuples by any variable, but have limited ability to ease the understanding of multiple categorical variables.

Scatterplot matrices (SPLOMs) were proposed in [11], and display a scatterplot for every possible pair of variables. Notable more recent work includes Scagnostics [36], which enable SPLOMs to scale up to many continuous variables, and Scatterdice [6], which demonstrates how they can be made highly interactive. SPLOMs nevertheless have shortcomings when used to visualize categorical variables. In Figure 1, the top three scatterplots (e.g., Month vs Region) each show a crossing of two categorical variables, resulting in an uninformative grid of points. Scatterplots showing a continuous vs categorical variable suffer from overplotting: in the Sales vs Product scatterplot, it is not obvious which of the products resulted in higher overall sales.

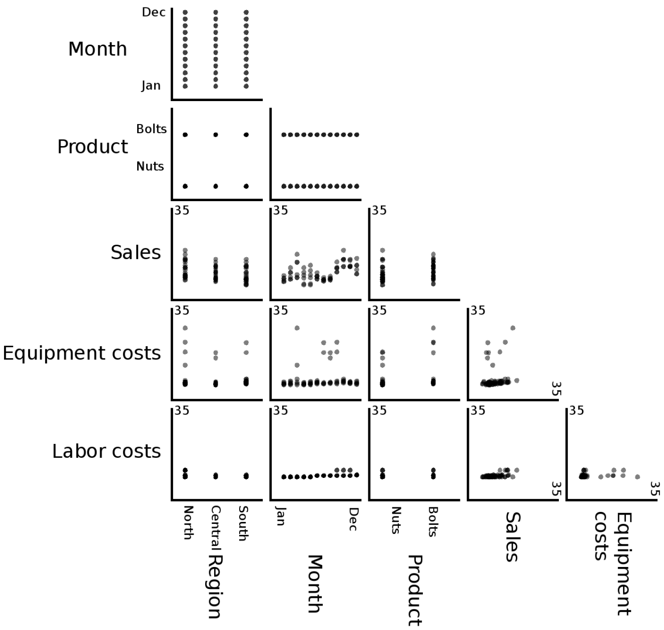


Fig. 1. A SPLOM of the nuts-and-bolts dataset.

HyperSlice [30] displays a matrix of slices of a scalar function of many dimensions, but cannot display several (dependent) continuous variables at once. The heatmaps of GPLOM, explained in the next section, are similar to HyperSlice, though GPLOM's heatmaps display aggregations of data rather than slices.

Parallel coordinates [14, 35] show each tuple as a polygonal line intersecting an axis once for each of the variables. Figure 2 shows an example. The 3 right-most axes show continuous variables, allowing us to see the distribution of values along them (the range and central tendency of values, and outliers). However, the 3 left-most axes show categorical variables, where every possible combination of values is covered, resembling complete bipartite graphs. This creates ambiguities that prevent us from visually tracing a tuple across all axes (although interactive brushing could alleviate this).

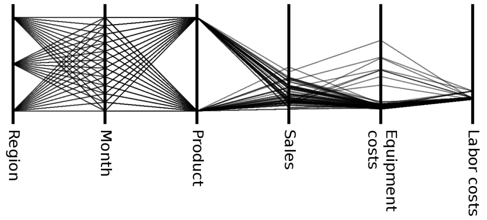


Fig. 2. A parallel coordinates plot of the nuts-and-bolts dataset.

Various combinations of scatterplots and parallel coordinates have been proposed, displaying them side-by-side [23, 27] or more tightly integrated [39, 12, 32, 4], but none of these approaches facilitate the visualization of categorical variables.

Arrays of glyphs can be used to visualize mdmv data, where each glyph shows one tuple [1, 3, 16, 22, 34]. This works well when there are at most 2 (independent) categorical variables. For example, an arrow plot [37] can display an arrow-shaped glyph at each of the points on a 2D grid, showing wind speed and wind direction over a geographic map. Extending this to 3 spatial dimensions results in occlusion, and beyond 3 dimensions it becomes very difficult to understand the ordering of glyphs along each dimension.

Dimensional stacking [17, 19] allows more than one categorical variable to be mapped to the same spatial axis, and has been used in database visualization [29, 18]. Figures 3 and 4 show examples, each of which shows a total of 4 variables. The two innermost variables of the stacking determine the type of chart shown: if the innermost vertical variable is a continuous variable (e.g., Sales), and the innermost horizontal variable is a categorical variable (e.g., Month), then barcharts are used. On the other hand, scatterplots are used if the two innermost variables are continuous variables (e.g., Equipment costs vs Sales).

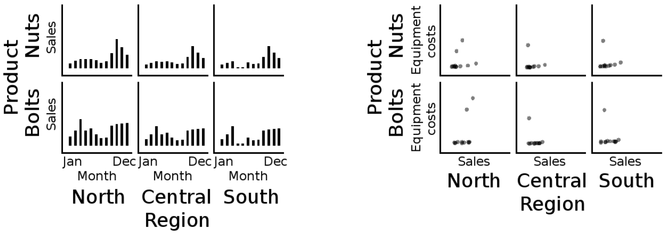


Fig. 3. Examples of dimensional stacking with the nuts-and-bolts data. Left: Product and Sales are mapped to the vertical axis, Region and Month are mapped to the horizontal. Right: Product and Equipment costs mapped to the vertical, Region and Sales to the horizontal.

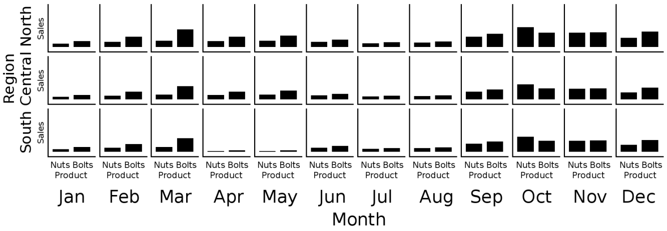


Fig. 4. Another example of dimensional stacking applied to the nuts-and-bolts data. Region and Sales are mapped to the vertical axis, Month and Product to the horizontal.

Each of the charts in Figures 3 and 4 show a *slice* of the data, allowing the user to see more detail. For example, Figure 4 reveals that sales were very low in the South in April and May. By comparison, in Figure 1, the Sales vs Month scatterplot also shows low sales in April and May, without revealing the Region.

The added detail visible in Figures 3 and 4, however, comes at the cost of exponential growth in space requirements as categorical variables are added. For example, if the dataset had an additional categorical variable Year with values 2001, 2002, ..., 2010, adding this as an outer variable to Figure 4 would increase the number of charts by a factor of 10. Partly for this reason, software like Tableau [18] does not show the user an initial visualization of the data. Instead, the user selects variables from a menu to construct the desired visualization.

The most closely related work to ours is the Generalized Pairs Plot [8], which extends the matrix in a SPLOM to allow a mix of chart types to be displayed, including mosaic plots, box plots, histograms, and density contours. As demonstrated in the next section, this is scalable to a larger number of continuous *and* categorical variables than previous techniques, because the space requirements scale linearly with the number of variables, rather than exponentially as with the previous example of dimensional stacking. Our GPLOM work further explores Emerson et al.'s ideas by (1) only using 3 kinds of charts, to make the visualization easier to understand by non-expert users who may simply want a visual overview of a business database as a first step in asking analytic questions; and by (2) extending the static plots in [8] through interactive techniques. We also (3) empirically compared GPLOM to a commercial product and found significant advantages with GPLOM in certain cases.

3 The GPLOM Visualization Technique

Figure 5 shows an example GPLOM of 6 variables. In the Sales vs Product chart, we clearly see that Bolts outsold Nuts, thanks to the use of aggregation (via a sum operator) that generated the bar heights. The overplotting seen in Figure 1's Sales vs Product scatterplot is thus avoided.

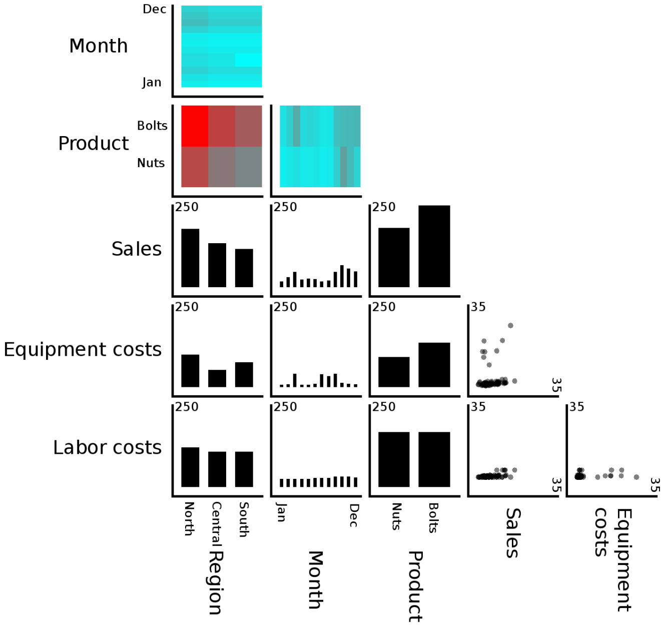


Fig. 5. A GPLOM of the nuts-and-bolts dataset. Barcharts and heatmaps show data aggregated by sum. The vertical axes on the barcharts extend to 250, to accommodate the larger values than in the scatterplots. The heatmaps are colored to show "Sales" as a function of categorical variables, and use a color scale varying from cyan for low values, through grey for mid values, to red for the highest values.

Figure 6 shows the layout of a GPLOM for M categorical variables $x_1,\ldots,x_M$ and N continuous variables $y_1,\ldots,y_N$. A full matrix would have $(M+N)\times(M+N)$ cells, however we only display the lower triangular half, without the diagonal, as is often done with SPLOMs (e.g., [36]). Thus, our GPLOM saves space compared to the full matrices in [8], leaving room for interactive elements such as the infobox (discussed shortly).

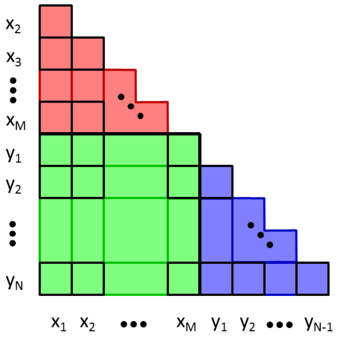


Fig. 6. Structure of a GPLOM.

The red region in Figure 6 contains pairs of categorical variables, and GPLOM visualizes these with heatmaps. The green region contains pairings of a continuous vs categorical variable, shown as barcharts. The purple region contains pairs of continuous variables, shown as scatterplots. (This grouping of variable types is comparable to Peng et al.'s [21] ordering of variables in a SPLOM according to their cardinality.) Note that the scatterplots show individual tuples, whereas the barcharts and heatmaps show aggregated data.

Other charts in these regions are possible, as demonstrated by Emerson et al. [8], such as boxplots or linecharts. However, their example plots show categorical variables with at most 4 distinct values. Complex charts, such as box plots and mosaic plots, become difficult to read with categorical variables with high-cardinality (Figure 7). Figure 8 shows a GPLOM for a large real-world dataset, where the categorical variables of Year, Day of month, and Carrier have 26, 31, and 32 distinct values, respectively. Restricting the GPLOM to only show 3 kinds of simple charts, namely heatmaps, barcharts, and scatterplots, helps keep the charts readable at these higher cardinality values.

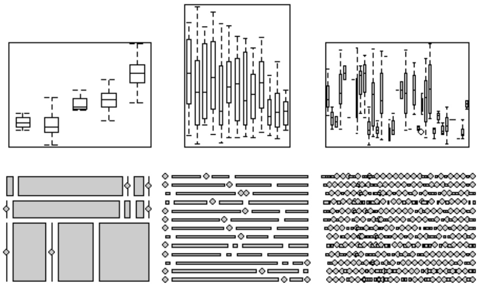
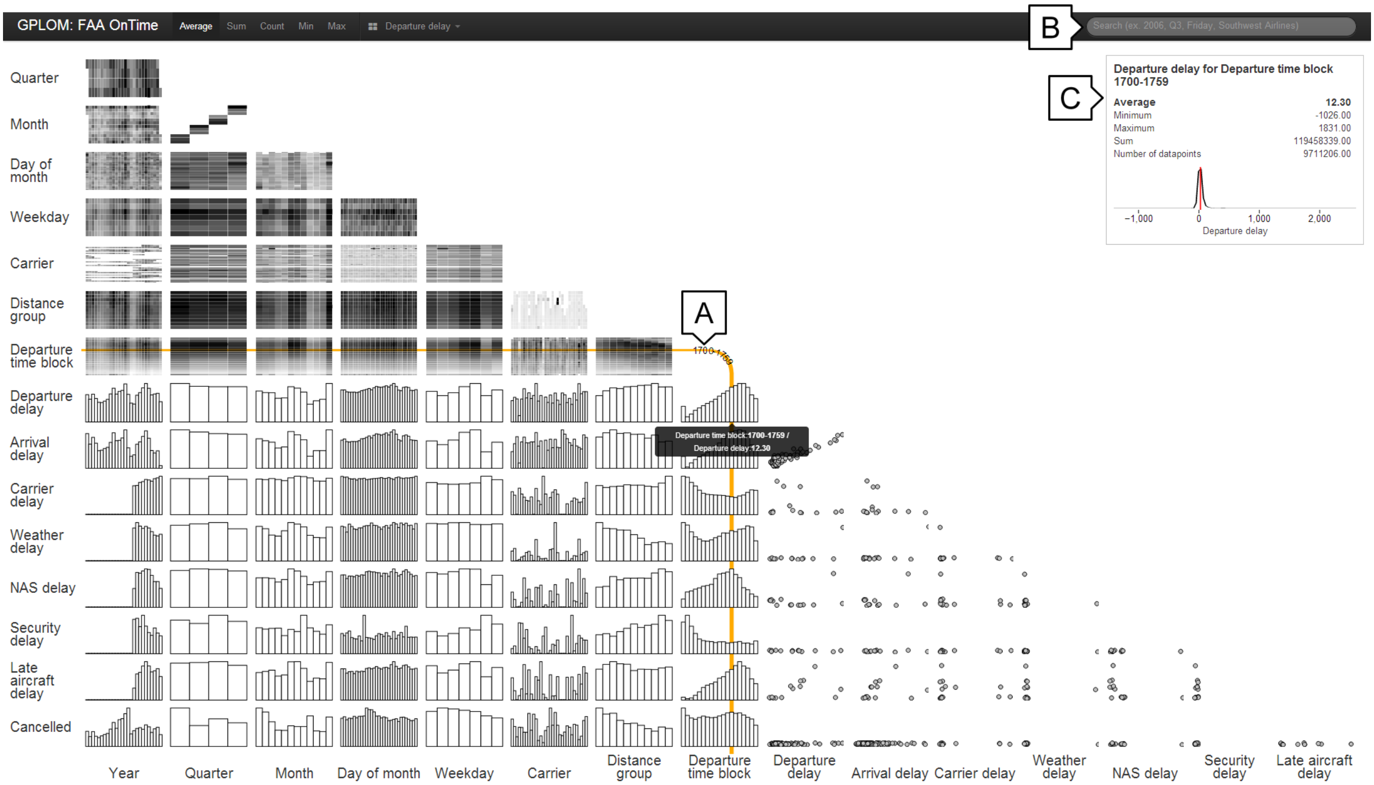


Fig. 7. Example plots extracted from a matrix generated with the `gpairs` package in R [7]. Top row: boxplots over variables of cardinality 5, 13, and 35, respectively. Bottom row: mosaic plots with cardinality 3×5, 5×13, and 13×35, respectively.

One tradeoff in designing a GPLOM is deciding if axes of the same variable should be scaled to the same range (facilitating comparisons of adjacent charts) or scaled to the maximum of the data in the chart. In Figure 1, all axes are scaled to 35. However, in Figure 5, the barcharts contain (aggregated) sums, and are therefore scaled to a larger range. The scatterplots in Figure 5, however, are still scaled to 35, to avoid having all the points clustered in a corner of the scatterplots. Furthermore, the heatmaps in Figure 5 share the same color scale, and we notice that only one of the heatmaps has a value close to the maximal red, because the other heatmaps are subdivided into months, reducing



Departure delay for Departure time block 1700-1759

Average	**12.30**
Minimum	-1026.00
Maximum	1831.00
Sum	119458339.00
Number of datapoints	9711206.00

Fig. 8. A GPLOM of 8 categorical variables, 8 continuous variables, and 144 million flights from the OnTime dataset. The menubar at the top displays the possible aggregation operators for barcharts and heatmaps: Average (currently selected), Sum, Count, Min, and Max. The menubar also shows that "Departure delay" is the currently selected (dependent) continuous variable for heatmaps. Other interface elements: A: bendy highlight; B: textual search box; C: infobox. Note that heatmaps and barcharts are computed over the whole dataset, but scatterplots only show a random sample of 200 data points each.

the values in them. Figure 8 instead scales each chart independently, according to the maximal value within it. This makes better use of spatial (and color) resolution, but makes it more difficult to compare charts.

3.1 Interaction

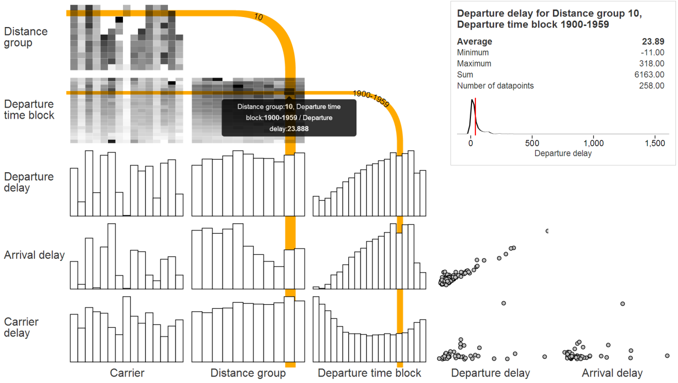
The user may interact with the GPLOM in several ways. A GPLOM contains bars and rectangles that afford easier pointing and clicking than the small points or dots in a normal SPLOM. In our GPLOM prototype, rolling the mouse cursor over a barchart bar or heatmap cell causes it to highlight. Clicking on a bar or cell selects it.

3.1.1 Linking

Linking (or coordination [25, 33, 20]) between charts is shown in two ways: *bendy highlights*, and *associative highlighting*.

Bendy highlights are specialized links that connect different charts, comparable to previous work that also draw links between views [5, 28, 4, 31]. Bendy highlights are curved links that show the value of a categorical variable during rollover or selection. A text string is displayed at the curved corner of the link to show the category (for example, the 5-6pm departure time block is displayed as "1700-1759" on the corner of the bendy highlight in Figure 8, A). Bendy highlights can also help understand the relationship between a heatmap cell and other charts (Figure 9).

Associative highlighting shows the relationship between charts when a categorical value is selected. There are three types of such highlighting. If the aggregation used in barcharts and heatmaps is the Sum or Count operator, then associative highlighting is achieved by highlighting the fraction of bars in other barcharts that is associated with the selected value (Figure 10). This is similar to the proportional highlighting of bars in [40]. If, instead, the aggregation used is Average, Min, or Max, then associative highlighting is achieved by displaying dots to show the average, min, or max value of data for the selected value (Figure 11). Finally, regardless of the aggregation operator, the corresponding dots in the scatterplots are highlighted.



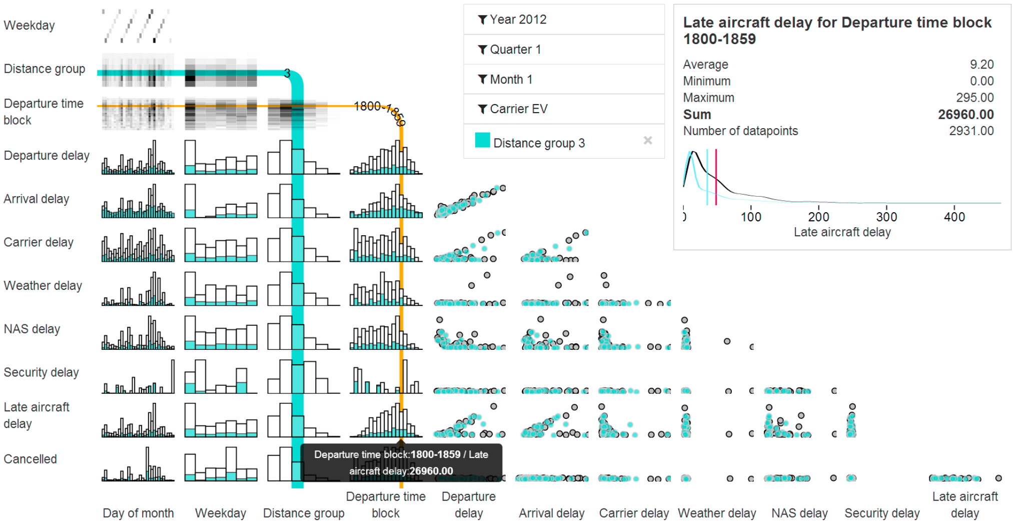
Departure delay for Distance group 10, Departure time block 1900-1959

Average	**23.89**
Minimum	-11.00
Maximum	318.00
Sum	6163.00
Number of datapoints	258.00

Fig. 9. Bendy highlights and a tooltip.

3.1.2 Filtering

To drill down, the user can double click on a bar (such as a bar for "Year" = 2012), causing a filter to be created that restricts the displayed data to that value. This "sheds" the corresponding categorical variable, removing a row and column of charts from the GPLOM, and creates a filter box that the user can later click on to roll back up. Figure 10 shows the result of applying 4 successive filters: "Year" = 2012, "Quarter" = 1, etc. We call this feature "dimensional shedding".



Year 2012

Quarter 1

Month 1

Carrier EV

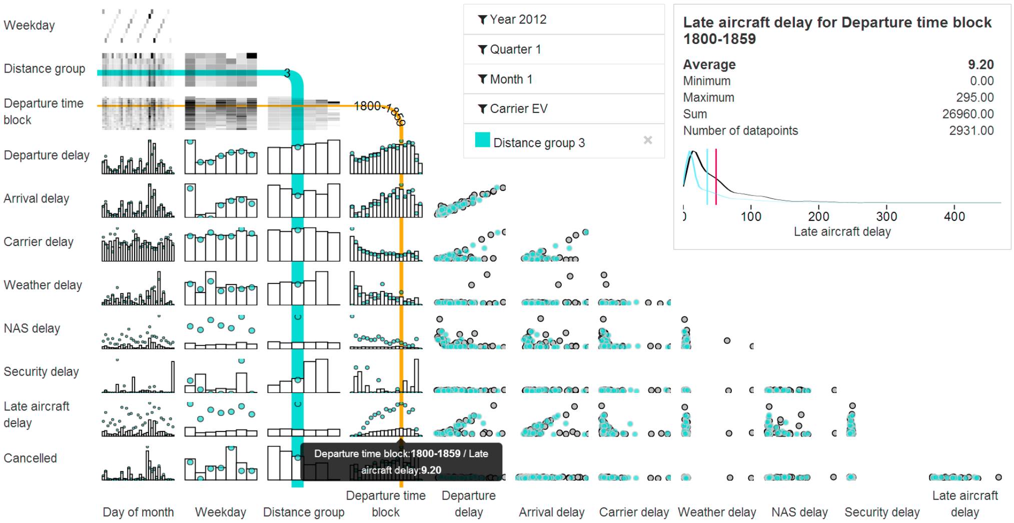
🟦 Distance group 3 ×

Late aircraft delay for Departure time block 1800-1859

Average	9.20
Minimum	0.00
Maximum	295.00
Sum	**26960.00**
Number of datapoints	2931.00

Departure time block:1800-1859 / Late aircraft delay:26960.00

Fig. 10. With the Sum aggregation operator, associative highlighting fills in the fraction of bars associated with the selected value “Departure time block” = “1800-1859”.



Year 2012

Quarter 1

Month 1

Carrier EV

🟦 Distance group 3 ×

Late aircraft delay for Departure time block 1800-1859

Average	**9.20**
Minimum	0.00
Maximum	295.00
Sum	26960.00
Number of datapoints	2931.00

Departure time block:1800-1859 / Late aircraft delay:9.20

Fig. 11. With the average aggregation operator, associative highlighting displays circles showing the average values for the selected value “Departure time block” = “1800-1859”.

3.1.3 Infobox

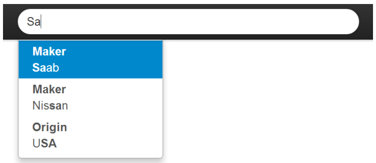
Additional information about the element under the cursor is displayed in the infobox (upper right corner of Figure 8), which contains the results of the various aggregation operators as well as a kernel density estimate plot, allowing the user to judge whether the underlying distribution is normal or not, its modality and its skewness.

3.1.4 Text Search

Because GPLOM displays a large number of charts, it may be time consuming for users to visually scan all variable names to find a desired chart. Thus, a textual search function (Figure 12) allows the user to enter a string, suggests autocompletions, and highlights the corresponding elements once the string is entered. Currently, our prototype only allows the user to enter values, however it would be easy to extend the prototype to also allow entering names of variables. This feature is similar to one proposed in section 6.1 of [10].

3.1.5 Labels

Due to the density of information displayed in a typical GPLOM, there is often insufficient room for labels showing the values of all categorical variables along their axes. Instead, GPLOM relies heavily on tooltips and bendy highlights that show the value under the cursor. In our first version of the prototype, we arranged categorical values on vertical axes sorted top-to-bottom, resulting in Figure 13, top. This resulted in many crossing bendy highlights. We therefore modified the prototype to sort values bottom-to-top, yielding Figure 13, middle, which is the order shown in other figures in this paper. A third possibility is shown in Figure 13, bottom, which avoids excessive crossed links while maintaining the usual top-to-bottom ordering of values.


Sa

Maker
Saab

Maker
Nissan

Origin
USA

Fig. 12. Entering the name of a value causes it to be highlighted in the GPLOM.

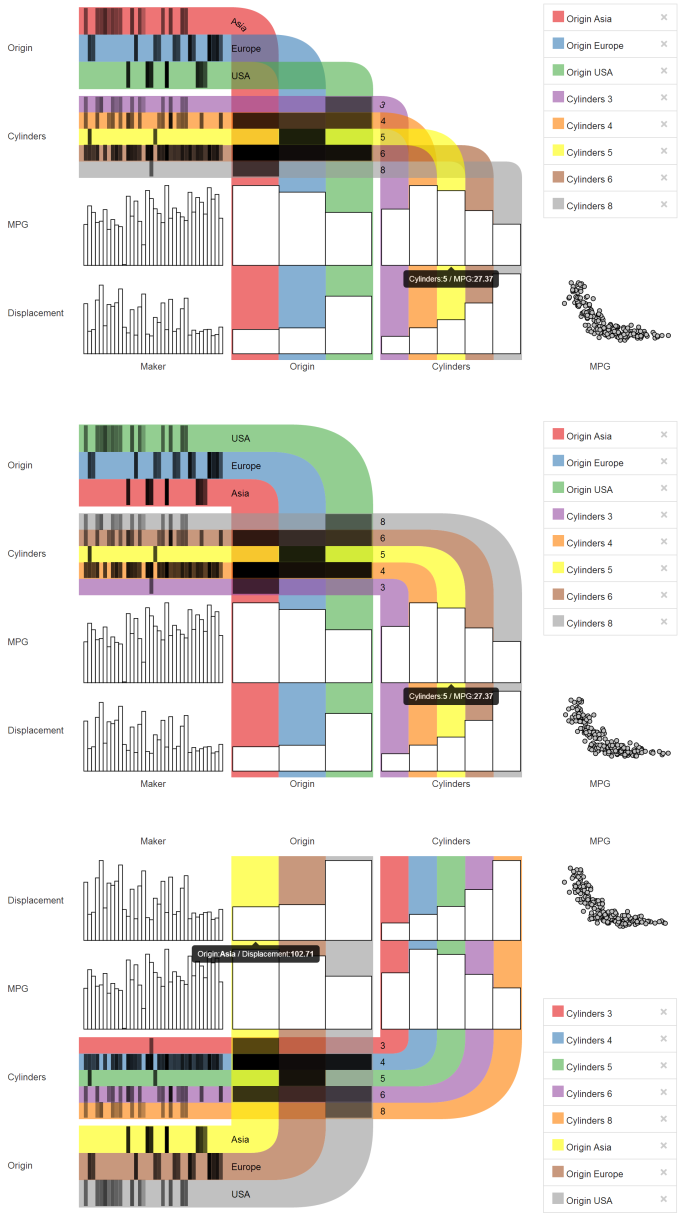


Fig. 13. Variants of bendy highlights. Top: alphabetical vertical sorting (e.g., Asia, Europe, USA). Middle: reverse alphabetical vertical sorting (e.g., USA, Europe, Asia). Bottom: a "reversed" GPLOM with alphabetical vertical sorting.

4 Experimental Evaluation

We suspect that one of the advantages of GPLOM is that users can answer questions by simply scanning for the appropriate chart, whereas in the commercial product Tableau they must explicitly construct a visualization. To investigate this idea, we performed an experimental comparison of user performance with both tools.

We chose three datasets for the experiment: a warm-up dataset (a converted sample SQL server database called Adventureworks) that was used to introduce users to both visualization tools, and two other datasets (Cars[1], and the OnTime[2] airline delay data for the month of December 2012). Cars and OnTime were each used with one of the tools, counterbalanced for dataset and ordering. One quarter of the users did (Tableau+Cars, GPLOM+OnTime), another quarter did (Tableau+OnTime, GPLOM+Cars), another quarter did (GPLOM+OnTime, Tableau+Cars) and the last quarter did (GPLOM+Cars, Tableau+OnTime).

Each trial required the user to answer a question about the dataset. There were 2 types of questions, and for each type of question, there could be zero or more criteria involved in the question. The *type of question* consisted of either questions that asked which type of trend or correlation (positive, negative or null) exists between two variables, if any, and questions that asked to find a particular data value, such as the year in which the average mileage per gallon for all cars was the highest. The *criteria count* ranged between zero to three criteria, so that a question "find the carrier with the highest average arrival delay" has zero criteria, while the question "find the day of the week when Hawaiian Airlines (HA) has the highest average delay for flights departing between 9:00-9:59" has two criteria (carrier=HA, departureTime=0900-0959).

In total, the experiment involved:
2 types of questions (trend or data)
× 4 criteria counts (0 through 3)
× 2 technique-dataset pairs (GPLOM and Tableau 7.0)
× 12 users
= 192 trials

The 12 students who participated (11 male, 1 female) were either from the software engineering or information technology engineering programs at ETS, at both the undergraduate and graduate levels. Each student was assigned to one of the four between-subjects groups and was asked to explore the warm-up dataset for five minutes with one of the two techniques (either Tableau or GPLOM, depending on the participant's group), as an exploration phase. Once the five minutes were over, each participant was shown how to use the software in order to answer the questions, then presented with eight questions for the warm-up dataset. After the questions on the warm-up dataset were answered, a second dataset (either Cars or OnTime, depending on the participant's group) was shown and the participant was asked to answer questions about the new dataset. Then, the participant explored the warm-up dataset again, using the other technique, answered the same eight questions using the other technique and, finally, answered a set of questions on a different dataset than the one explored with the first technique.

None of the participants indicated that they had any prior experience with Tableau or with the GPLOM prototype. During the exploration phases and warm up trials, users were free to ask questions, and were shown all the features of the user interfaces that were necessary to answer the questions in the experiment.

The participants used a single monitor workstation equipped with a 24 inch monitor, keyboard and mouse.

The GPLOM prototype consisted of a web application built using D3 [2] and JavaScript, running in the Chrome web browser (version 25), as well as a server-side backend. The server-side backend managed communication between the client and a MySQL server, computing agregates to be consumed by D3. It was built using the Play framework 2.0.4 and ran in production mode during user tests.

Tableau and GPLOM both connected to the same MySQL database over a wired gigabit Ethernet network.

As the GPLOM prototype was not optimized for performance, each time a participant added a filter by double clicking, a full page load by Chrome was executed, requiring Chrome to re-interpret and run JavaScript code (in theory, this could be eliminated with more careful coding) and also regenerating all charts (this is unavoidable with the GPLOM approach). We subsequently measured that the median time for all this to occur was 1.7 seconds for the Cars dataset, and 5.7 seconds for OnTime.

The questions were displayed to users on a second monitor controlled by the experimenter. When the user indicated they were ready to start a trial, the experimenter clicked a button to display the question and start a timer. The user then read the question and interacted with the visualization tool until they said they could answer the question, at which point the experimenter stopped the timer (triggering a simultaneous screen grab of the user's screen), and transcribed the user's

[1] http://lib.stat.cmu.edu/datasets/

[2] http://www.transtats.bts.gov/Fields.asp?Table_ID=236

verbal answer. The time elapsed was recorded. If the user decided they wished to check or change their answer by performing further interactions with the visualization tool, the time of their last answer determined the recorded duration. No feedback was given to indicate to the user if their final answer was correct or incorrect.

4.1 Results

Because each participant was only exposed to half of the four {GPLOM, Tableau} × {Cars, OnTime} combinations, the performance data were separated by dataset for analysis. Some of the main results are summarized in Figure 14 and below:

	GPLOM		Tableau	
	median time (s)	error rate	median time (s)	error rate
Cars	23.67	13%	41.14	10%
OnTime	48.68	17%	59.58	33%

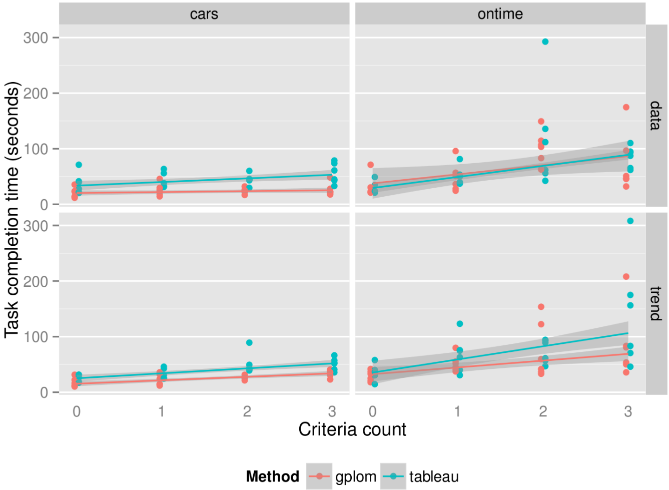


Fig. 14. Task completion time for each method, as a function of number of criteria, broken down by dataset and question type. A robust linear model was fitted to yield the straight lines.

The time taken by participants to answer was non-normally distributed (Shapiro-Wilk normality test, $p < 0.01$). The non-parametric ANOVA-type statistic (ATS) revealed that GPLOM was significantly faster than Tableau for the Cars dataset ($p < 0.01$), and that the criteria count had a significant effect on time in both the Cars dataset ($p < 0.01$) and the OnTime dataset ($p < 0.01$), with time increasing with criteria count. There was no significant difference in time between GPLOM and Tableau for the OnTime dataset, although GPLOM had a lower median time (48.68 seconds for GPLOM vs 59.58 seconds for Tableau).

Examining Figure 14, we note that the case where GPLOM seemed to have the least advantage with respect to Tableau was with "data" questions about the OnTime dataset when the criteria count was 2. This particular case corresponds to the only question that required the user to use the Count aggregation operator in GPLOM. In hindsight, we recall several users having difficulty with this question, and suspect that this question was relatively easier in Tableau because Tableau has a pre-defined variable "Number of records", obviating the need for users to select a Count aggregation operator in Tableau.

A logistic regression revealed that questions about the OnTime dataset had a significantly higher error rate than questions about Cars. GPLOM had a lower overall error rate than Tableau, but not significantly.

Post-questionnaires asked users to rate the two interfaces against nine criteria such as "intuitive", "easy to learn", etc. On average, users gave a higher (i.e., more positive) rating to GPLOM than Tableau for all of these criteria, but Wilcoxon signed rank tests showed that only two were significant: users judged GPLOM to be significantly more "fast" ($p < 0.01$) and significantly more "fluid" than Tableau ($p < 0.05$).

4.2 Discussion

We did not attempt to subtract the full page load time per filter incurred by the GPLOM prototype from the recorded time taken by the participant to answer, as there is no way to differentiate between the user waiting for Chrome to render the page and the user thinking about the next filter to enter while the page is loading. It is possible that, if the page load time were reduced with better coding, this would further differentiate GPLOM from Tableau, as Tableau did not incur such an overhead.

On the other hand, the error rate with Tableau was sometimes rather high, and this may be because the users were too inexperienced with it, despite the warm up trials. It is possible that a follow-up study with more experienced users would yield different results.

Nevertheless, in our study, GPLOM resulted in a lower median time in both datasets, and a significantly lower time in the Cars dataset. A possible explanation for this difference would be the difference between the process of building a filter in each visualization.

In Tableau, the process to build a filter comprises the following steps:

1. Pick the categorical variable to filter from the list of dimensions
2. Drag the selected categorical variable to the filter shelf
3. Select the desired value for the filter from the list of possible values for the categorical variable
4. Click OK to dismiss the filter dialog box

On the other hand, in GPLOM, the process to build a filter requires the following steps:

1. Locate one bar chart whose x axis corresponds to the categorical variable to filter
2. Locate the particular bar that corresponds to the desired value on which to filter by hovering over the bar and reading its associated tooltip
3. Double-click the bar

Alternatively, the user can build a filter in GPLOM in the following fashion:

1. Move the mouse cursor to the search box and click it
2. Enter the desired value to search for using the keyboard
3. Move the mouse cursor to one of the highlighted bars on a bar chart
4. Double-click the bar

Another explanation for the faster performance of GPLOM relative to Tableau would be the dimensional shedding feature of GPLOM. As participants drilled down in GPLOM by double clicking, the number of displayed charts was correspondingly reduced and the possible values on each chart's horizontal axis only contained the list of allowed values. In contrast, Tableau's design requires the list of dimensions (categorical variables) to stay static and building a filter often listed values incompatible with other filters. For example, even if a previous filter filtered out cars by Asian manufacturers, Honda and Toyota would still appear if the user attempted to add a filter for the manufacturer's name. Furthermore, when users built an invalid combination of filters, Tableau displayed no data at all, which stumped some participants and caused them to search (often for an extended period of time – see for example the outlier points in Figure 14) for a reason as to why the display was completely blank. As GPLOM always shows all available data and filtering is done by picking a particular subset of the displayed data, it is impossible for a user to build such a filter combination.

Another problem participants encountered with Tableau was their building of a chart that contained too many categorical variables or did not answer the question they were asking; on the other hand, some participants answered some questions using the wrong chart in GPLOM, so the problem could be one of user education or wanting to please the experimenter by answering something.

4.3 Improvements

Several improvements can be made to the GPLOM prototype. In its current iteration, the search box only contains the data contained in the database, without mapping it to more user friendly concepts (carrier name "WN" instead of Southwest or "1" as a day of the week, instead of Monday). This confused some of the users, who tried several times, unsuccessfully, to get the search box to find the values they were looking for. Ensuring that there is a rich data dictionary that has multiple synonyms for values would significantly improve the users' experience in that regard.

Another problem was that the search box's color contrast was insufficient (see top right of Figure 8) and eight of the twelve users missed it entirely during the five minute exploration period. Improving its contrast and adding a magnifying glass icon might make it easier for users to discover the feature. Even when they were told that the search box existed, most users did not use it, instead using the mouse to find and select values to filter on.

One significant problem that was repeatedly encountered during user testing of the GPLOM prototype is the lack of clear affordances for interaction. Users did not seem compelled to click, much less double click, on charts. During the exploration phase, out of twelve users, only one found that it was possible to filter data by double clicking on bars, although some tried right-clicking (which only brought Chrome's default right-click menu). This lack of clear affordances meant that users often tried to click and double click on the brightly colored bendy highlight, which did nothing; in retrospect, it seems like an obvious affordance for user interaction which could be used for highlighting and filtering.

The associative highlighting, while useful for part-to-whole comparisons, was often misunderstood by users; it was almost never used to answer questions on datasets, even though it displays the exact same data that double-click filtering on a particular value would.

Another misunderstood feature was the kernel density estimate plot, which confused users much more than it helped them. We postulate that histograms, density plots, Q-Q plots, rug plots and other statistical tools, while very important to evaluate distribution shape, are unlikely to be understood by average business users.

5 Conclusion and Future Directions

Despite a large variety of charts and visualizations, it remains unclear to many non-expert users how to visualize the contents of a typical database in a way that gives them an overview of variables they may not be familiar with. Many advanced techniques have been proposed [38, 9, 15], but most of these have seen limited real-world deployment, and almost none of them are designed for the simultaneous visualization of multiple categorical and continuous variables, with the exception of Emerson et al. [8]. Both Emerson et al.'s approach and the GPLOM can give users a visual overview of more than 10 variables, allowing them to visually scan for interesting relationships and allow for serendipitously discovering outliers or thinking of unplanned questions for further analysis.

Compared to Emerson et al., GPLOM (1) only uses 3 kinds of charts, the minimum number necessary to cover the three kinds of pairings of variables, which may make it easier to understand for non-expert users and scale better to high-cardinality categorical variables; (2) demonstrates two ways of interactively linking charts, through bendy highlights and associative highlighting; (3) demonstrates text search to quickly find values of interest; and (4) saves screen space by only displaying the lower triangular half of the matrix. We also presented experimental evidence that GPLOM is sometimes significantly faster than Tableau, a commercial product, for the kinds of questions we tested.

Future work might compare the performance of the reversed GPLOM (Figure 13, bottom) with the upright version. During the exploration phase of our study, most of the participants seemed to explore the software from top to bottom, left to right and spent most of their time trying to understand the heat maps. The reversed GPLOM would mean that the first visual elements encountered by users would be barcharts, which are easier to understand. This might better ease novice users into the GPLOM.

Future work could also compare GPLOM with other mdmv visualizations or database tools, such as xmdv, ggobi or Mondrian.

The GPLOM prototype could be modified to accommodate a wider range of user skills. Novice users could be shown only the matrix of barcharts, while more advanced types of plots (such as those in [8]) could be available for expert users.

There might also be hybrid ways to combine the matrix layout of GPLOM with the dimensional stacking of Polaris / Tableau, giving the user more control over the tradeoff of number of charts and level of detail.

Finally, we plan to explore ways of improving the performance of GPLOM with extremely large datasets. While performance of the GPLOM prototype on the complete OnTime dataset (≈144 million records) was still within acceptable bounds for interactive exploration, it required the usage of an in-memory columnar database running on a server with 16 dual-core processors equipped with 512 gigabytes of RAM. Incremental approaches for large data visualization, such as VisReduce [13], could also be applied to reduce the perceived system latency.

Acknowledgments

Thanks to the participants in our user study, and to Julian Gosper of SAP who gave feedback in early stages of this project. This research was supported by SAP and NSERC.

References

[1] J. Bertin. *Sémiologie graphique: Les diagrammes, Les réseaux, Les cartes*. Éditions Gauthier-Villars, Paris, 1967.

[2] M. Bostock, V. Ogievetsky, and J. Heer. D^3 data-driven documents. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 17(12):2301–2309, 2011.

[3] H. Chernoff. The use of faces to represent points in k-dimensional space graphically. *Journal of the American Statistical Association*, 68(342):361–368, June 1973.

[4] J. H. T. Claessen and J. J. van Wijk. Flexible linked axes for multivariate data visualization. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 17(12):2310–2316, 2011.

[5] C. Collins and S. Carpendale. VisLink: Revealing relationships amongst visualizations. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 13(6):1192–1199, 2007.

[6] N. Elmqvist, P. Dragicevic, and J.-D. Fekete. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 14(6):1141–1148, 2008.

[7] J. W. Emerson and W. A. Green. *gpairs: The Generalized Pairs Plot*, 2012. R package version 1.1.

[8] J. W. Emerson, W. A. Green, B. Schloerke, J. Crowley, D. Cook, H. Hofmann, and H. Wickham. The generalized pairs plot. *Journal of Computational and Graphical Statistics*, 22(1):79–91, 2013.

[9] U. C. Georges Grinstein, Marjan Trutschl. High-dimensional visualizations. In *Proc. International Workshop on Visual Data Mining*, pages 7–19, 2001.

[10] L. Grammel, M. K. Tory, and M.-A. Storey. How information visualization novices construct visualizations. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 16(6):943–952, 2010.

[11] J. A. Hartigan. Printer graphics for clustering. *Journal of Statistical Computation and Simulation*, 4(3):187–213, 1975.

[12] D. Holten and J. J. van Wijk. Evaluation of cluster identification performance for different PCP variants. In *Proceedings of Eurographics/IEEE-VGTC Symposium on Visualization (EuroVis)*, pages 793–802, 2010.

[13] J.-F. Im, F. G. Villegas, and M. J. McGuffin. VisReduce: Fast and responsive incremental information visualization of large datasets, 2013. Submitted for publication.

[14] A. Inselberg. The plane with parallel coordinates. *Visual Computer*, 1:69–91, 1985.

[15] D. A. Keim. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 8(1):1–8, 2002.

[16] B. Kleiner and J. A. Hartigan. Representing points in many dimensions by trees and castles. *Journal of the American Statistical Association*, 76(374):260–269, June 1981.

[17] J. LeBlanc, M. O. Ward, and N. Wittels. Exploring N-dimensional databases. In *Proceedings of IEEE Visualization (VIS)*, pages 230–237, 1990.

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

[19] T. Mihalisin, J. Timlin, and J. Schwegler. Visualization and analysis of multi-variate data: A technique for all fields. In *Proceedings of IEEE Visualization (VIS)*, pages 171–178, 1991.

[20] C. North and B. Shneiderman. Snap-together visualization: A user interface for coordinating visualizations via relational schemata. In *Proceedings of Advanced Visual Interfaces (AVI)*, pages 128–135, 2000.

[21] W. Peng, M. O. Ward, and E. A. Rundensteiner. Clutter reduction in multi-dimensional data visualization using dimension reordering. In *Proc. IEEE Symposium on Information Visualization (InfoVis)*, pages 89–96, 2004.

[22] R. M. Pickett and G. G. Grinstein. Iconographic displays for visualizing multidimensional data. In *Proceedings of IEEE Conference on Systems, Man, and Cybernetics*, pages 514–519, 1988.

[23] H. Qu, W.-Y. Chan, A. Xu, K.-L. Chung, K.-H. Lau, and P. Guo. Visual analysis of the air pollution problem in Hong Kong. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 13(6):1408–1415, 2007.

[24] R. Rao and S. K. Card. The table lens: merging graphical and symbolic representations in an interactive focus + context visualization for tabular information. In *Proceedings of ACM Conference on Human Factors in Computing Systems (CHI)*, pages 318–322, 1994.

[25] J. C. Roberts. State of the art: Coordinated & multiple views in exploratory visualization. In *Coordinated and Multiple Views in Exploratory Visualization (CMV)*, pages 61–71, 2007.

[26] M. Spenke, C. Beilken, and T. Berlage. FOCUS: the interactive table for product comparison and selection. In *Proceedings of ACM Symposium on User Interface Software and Technology*, pages 41–50, 1996.

[27] C. A. Steed, J. E. Swan II, T. J. Jankun-Kelly, and P. J. Fitzpatrick. Guided analysis of hurricane trends using statistical processes integrated with interactive parallel coordinates. In *Proc. IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 19–26, 2009.

[28] M. Steinberger, M. Waldner, M. Streit, A. Lex, and D. Schmalstieg. Context-preserving visual links. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 17(12):2249–2258, 2011.

[29] C. Stolte, D. Tang, and P. Hanrahan. Polaris: A system for query, analysis, and visualization of multidimensional relational databases. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 8(1):52–65, 2002.

[30] J. J. van Wijk and R. van Liere. HyperSlice: Visualization of scalar functions of many variables. In *Proceedings of IEEE Visualization (VIS)*, pages 119–125, 1993.

[31] C. Viau and M. J. McGuffin. ConnectedCharts: Explicit visualization of relationships between data graphics. *Computer Graphics Forum (Proceedings of EuroVis 2012)*, 31(3):1285–1294, 2012.

[32] C. Viau, M. J. McGuffin, Y. Chiricota, and I. Jurisica. The FlowVizMenu and parallel scatterplot matrix: Hybrid multidimensional visualizations for network exploration. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 16(6):1100–1108, 2010.

[33] M. Q. Wang Baldonado, A. Woodruff, and A. Kuchinsky. Guidelines for using multiple views in information visualization. In *Proceedings of Advanced Visual Interfaces (AVI)*, pages 110–119, 2000.

[34] M. O. Ward. A taxonomy of glyph placement strategies for multidimensional data visualization. *Information Visualization*, 1:194–210, 2002.

[35] E. J. Wegman. Hyperdimensional data analysis using parallel coordinates. *J. of the American Statistical Association*, 85(411):664–675, 1990.

[36] L. Wilkinson, A. Anand, and R. Grossman. Graph-theoretic scagnostics. In *Proc. IEEE Symposium on Information Visualization (InfoVis)*, pages 157–164, 2005.

[37] C. M. Wittenbrink, A. T. Pang, and S. K. Lodha. Glyphs for visualizing uncertainty in vector fields. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 2(3):266–279, September 1996.

[38] P. C. Wong and R. D. Bergeron. 30 years of multidimensional multivariate visualization, 1997. Chapter 1 (pp. 3–33) of Gregory M. Nielson, Hans Hagen, and Heinrich Müller, editors, Scientific Visualization: Overviews, Methodologies, and Techniques, IEEE Computer Society.

[39] X. Yuan, P. Guo, H. Xiao, H. Zhou, and H. Qu. Scattering points in parallel coordinates. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 15(6):1001–1008, 2009.

[40] J. Zhang and G. Marchionini. Coupling browse and search in highly interactive user interfaces: A study of the relation browser++. In *Proc. ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL)*, pages 384–384, 2004.