Regression Cube: A Technique for Multidimensional Visual Exploration and Interactive Pattern Finding

YU-HSUAN CHAN, University of California at Davis
CARLOS D. CORREA, Lawrence Livermore National Lab
KWAN-LIU MA, University of California at Davis

Scatterplots are commonly used to visualize multidimensional data; however, 2D projections of data offer limited understanding of the high-dimensional interactions between data points. We introduce an interactive 3D extension of scatterplots called the Regression Cube (RC), which augments a 3D scatterplot with three facets on which the correlations between the two variables are revealed by sensitivity lines and sensitivity streamlines. The sensitivity visualization of local regression on the 2D projections provides insights about the shape of the data through its orientation and continuity cues. We also introduce a series of visual operations such as clustering, brushing, and selection supported in RC. By iteratively refining the selection of data points of interest, RC is able to reveal salient local correlation patterns that may otherwise remain hidden with a global analysis. We have demonstrated our system with two examples and a user-oriented evaluation, and we show how RCs enable interactive visual exploration of multidimensional datasets via a variety of classification and information retrieval tasks. A video demo of RC is available.

Categories and Subject Descriptors: H.5.2 [**User Interfaces**]: Interaction Styles

General Terms: Visualization, scatterplot, sensitivity analysis, interactions, pattern discovery, data transformation, model fitting, multidimensional data visualization

ACM Reference Format:
Yu-Hsuan Chan, Carlos D. Correa, and Kwan-Liu Ma. 2014. Regression cube: A technique for multidimensional visual exploration and interactive pattern finding. ACM Trans. Interact. Intell. Syst. 4, 1, Article 7 (March 2014), 32 pages.
DOI: http://dx.doi.org/10.1145/2590349

1. INTRODUCTION

User interaction plays an important role in visual analytics. To gain insight from raw data, analysts go through an iterative process of discovery until their questions about the data are answered with sufficient confidence. From input to insight, data are subjected to a series of transformations at different levels: (1) the data level, often the subject of data mining, (2) the visual level, which is the focus of visualization, and (3) the view level, often the focus of human-computer interaction [Card et al. 1999]. During any step of these transformations, user interaction provides feedback by proposing different hypotheses, setting up algorithm parameters, changing visualization properties, or commenting on the visual evidences of the produced images. Spence [2007] pointed out that the importance of user interaction in visual analytics lies in the possibility for

The reviewing of this article was managed by special issue associate editors Remco Chang, David Ebert, and Daniel Keim.
Authors' addresses: Y.-H. Chan and K.-L. Ma, 2136 Kemper Hall, 1 Shields Avenue, Davis CA 95616; emails: chany@cs.ucdavis.edu; ma@cs.ucdavis.edu; C. D. Correa, Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, 7000 East Avenue, Livermore, CA 94550; email: cdcorrea@gmail.com.

DOI: http://dx.doi.org/10.1145/2590349

users to explore interactively the subsets of a corpus of data to find their way toward the view that triggers an "a-ha!" experience.

In this study, we examine the interplay between *filtering* and *insight*, as facilitated by interaction techniques such as zooming, panning, and view changes. We focus on *selection* as the main approach to filtering and *regression analysis* as the basic mechanism to obtain insight about the interactions between the different aspects of the data. Our objective with the proposed technique is to understand how interactive filtering benefits the study of correlation between variables in a dataset. Additionally, we seek to demonstrate that visual feedback is useful in revealing trends at multiple scales. By "selection," we refer to all interactive means that allows a user to select one or more elements from the data space. It includes *pointing and clicking*, rectangular *region picking* by clicking and dragging, and more sophisticated mechanisms such as *brushing* or *sketching*.

To further clarify our contribution, we now review the process of data discovery via regression. Linear regression, the focus of this work, is one of the most commonly used mechanisms for analysts to discover and quantify linear correlation between two variables. A positive correlation indicates a sense of dependence between two variables, whereas a negative correlation indicates an inverse dependence. No correlation is an indication of independence between two variables. In simple, low-dimensional datasets, it is often easy to see correlation directly from a scatterplot, and regression serves to quantify the extents of such correlation. However, in multidimensional datasets, the correlation may be the complex product of interactions between multiple variables at the same time. Therefore, it may not be easy to discover the correlations via simple visual inspection. Additionally, exhaustive exploration of all projective spaces becomes prohibitive since the number of possibilities expands with the factorial of the number of dimensions. In these situations, a more effective mechanism is the interactive visual exploration, where the users explore the data space via *selection* of data points and then reapply regression locally on specific selections. Although this process may be used currently in an ad hoc manner, this is never explicit to the user. In this article, we aim to make this iterative two-step process—(1) select the data and (2) update the regression analysis—explicit and explorative to the users.

To this end, we introduce the two mechanisms. First, we define the regression hierarchy, a tree-based representation of the process of regression. It consists of a root node representing all data elements and a set of descendant nodes linked from the root to represent the data partition by the selection. If a linear pattern is found in one of the leaf nodes, its node in the regression hierarchy shows a high correlation value. Second, we introduce the Regression Cube (RC), an interactive interface that "hosts" the subset of the dataset that has been explored by the user. In our case, we use a 3D rotatable interface for RC to handle at most three dimensions at the same time, shown as the three "facets" of 2D projections. On the facets, we show the sensitivity-augmented scatterplots. The regression hierarchy is then a tree-based summary of these RCs and the linkages between them. The analytic process can be then defined in terms of these special operations: (1) dimension selection or creation of the cube, which puts the input data into a 3D interface from the three variables the user chooses; (2) selection of data points, via brushing or other appropriate methods, that divides the dataset into two subsets: the *in-selection* set and its complement; (3) export, which creates two RCs: one for the selection and another for its complement. These cubes are connected to the parent cube where the selection takes place, and thereby the hierarchy of cubes is formed when the process is iterated. This top-down comprehension is explicitly visualized, and it helps analysts iteratively refine the regression analysis to narrow down to the proper subsets of data that exploit certain trends.

As mentioned earlier, the objective of selection is to discover interesting patterns that may be hidden from a particular projection. In addition to traditional filtering using the axis-aligned rectangular selections or drawing an arbitrary shape by a lasso tool to select, we provide users with visual cues of sensitivity about the local variation between variables on each data point. We employ sensitivity flows [Chan et al. 2010], which enhance traditional scatterplots with local regression analysis, so that the analysts can understand locally linear relationships at a glance. Although the idea of RCs is not exclusive to regression analysis, we have found that the regression-aligned selections are better for building regression hierarchies than axis-aligned selections, as they tend to group data elements that share similar trends together and thus are more resilient to small changes in the selection.

Note that RC uses an underlying augmentation of scatterplots based on our previous work on flow-based scatterplots [Chan et al. 2010] and the generalized sensitivity scatterplot [Chan et al. 2013a], which deals more with the theoretical aspects of sensitivity analysis in visual analytics. The former introduces the analogy between regression and flow to help the analysts understand correlations between a pair of variables, and the latter reveals the high-dimensional trends and generalizes the notion of sensitivities by inverting the order of data transformations to make the differentiation occur before the projection when computing regression lines. In this article, we focus on the pragmatic implications of regression in the interactive visual analytics that has led to the creation of the regression hierarchy, obtained by progressively fitting regression lines and then filtering by the conventional or the regression-aligned selections. It refers the sensitivity information in building such hierarchies and thus incorporates the regression analytics that are not available with traditional plots, and it visualizes the process of the iterative regression analysis as the regression hierarchy graph explicitly. Moreover, the regression hierarchies are not unique to sensitivity-augmented scatterplots but can be a general framework that applies to traditional or any other kind of augmented scatterplots as well. We believe that these are separate contributions within the larger field of regression and visual analytics.

RCs help users to perform interactive regression analysis and to develop hypotheses on different subsets of data at the same time. In this article, we describe our approach to generating these hierarchies, how this representation facilitates users' insight, a qualitative evaluation of two examples in Section 5, and a user-oriented quantitative evaluation in Section 6.

2. RELATED WORK

2.1. Multivariate Analysis

Multivariate analysis is at the core of visual analytics. Approaches can be categorized as automatic analysis, such as regression [Draper and Smith 1998], generalized additive models [Hastie and Tibshirani 1986] and response surface analysis [Box and Draper 1987], or visualization approaches. Since data is becoming increasingly large and complex, data-driven approaches often employ simplification techniques, which either reduce the number of observations, such as binning, sampling [Thompson et al. 1996] or clustering [Berkhin 2006], or reduce the number of dimensions in the data, such as projections [Shlens 2005] and multidimensional scaling. Visually centered approaches follow a different strategy, where correlations and trends emerge as salient structures in the human visual system. These approaches are often coupled with interactive manipulation. For example, Jeong et al. [2009] proposed to augment traditional data analysis tools such as Principal Component Analysis with interactive manipulation for a better understanding of the transformation and the data itself. Yang et al. [2007] integrated analysis tools with visual exploration of multivariate data and

incorporated user interest to guide the analysis. In this article, we present a combination of analysis and visualization tools that exploit sensitivity analysis for effective exploration and navigation of multidimensional data.

2.2. Sensitivity Analysis

Sensitivity analysis refers, in general, to the analysis of the variation of the outputs in a model to small perturbation of their inputs. Numerous approaches have been proposed to this end. A number of methods fall into the class of local analysis, such as adjoint analysis [Cacuci et al. 2005] and automated differentiation [Griewank et al. 2000], where the sensitivity parameters are found by simply taking the derivatives of the output with respect to the input, $s_{ij} = \partial Y_i / \partial X_j$. Because this is usually done in a small neighborhood of the data, they are usually called *local methods*. Others have proposed global estimates of sensitivity, which use sampling or statistical techniques. The most common statistical method is based on variance that provides an estimate of the sensitivity in terms of the probability distribution of the inputs [Chan et al. 1997; Jansen 1999; Sobol 2001; Helton et al. 2006; Arriola and Hyman 2007]. Other approaches directly introduce perturbation on the input data by manipulating certain parameters and compute the ensuing variation on the output. Since it is computationally expensive to try the entire parameter space, numerous approaches use sampling-based methods as extensively surveyed by Helton et al. [2006]. Different simulation strategies have been applied, including random, importance, and Latin hypercube sampling [Iman and Helton 1988].

Christopher Frey and Patil [2002] also reviewed a number of sensitivity analysis methods. Tanaka [1994] surveyed the sensitivity analysis in the scope of multivariate data analysis. Specific analyses for certain common data analysis tools have been proposed. Chan et al. [1997] presented a sensitivity analysis for variance-based methods in general. Cormode and McGregor [2008], Chau et al. [2006], and Ngai et al. [2006] proposed extensions to perform k-means clustering on uncertain data. Similar studies have been carried out to quantify the sensitivity and uncertainty of the principal components of multivariate data [Šmídl and Quinn 2007; Yamanishi and Tanaka 2005]. Kurowicka and Cooke [2006] extended the issue of uncertainty analysis with high-dimensional dependence modeling, combining both analytical tools with graphic representations.

Barlowe et al. [2008] proposed the use of histograms and scatterplot matrices to visualize the partial derivatives of the dependent variable over the independent variables and to reveal the positive or the negative correlations between the output and the factors in a multivariate visual analysis. Correa et al. [2009] used sensitivity analysis to propagate the uncertainty in a series of data transformations and proposed a number of extensions to show this uncertainty in 2D scatterplots. In our work, we generalized the idea of sensitivity visualization as flow-based scatterplots [Chan et al. 2010]. Bachthaler and Weiskopf [2008] presented the continuous scatterplots, which generate a continuous density function for a scatterplot and alleviate the issues with missing data. Our idea of flow-based scatterplots has a similar concept, which attempts to find a continuous representation of the density that explains the 2D plot. However, we use a local analysis based on derivatives to find local trends in a scattered manner. Local analysis is popular in multivariate analysis because its computational efficiency and practicality, owing to the fact that the closed form of the data model is usually unknown. Berger et al. [2011] proposed an interactive exploration through the continuous parameter space as a local analysis using multivariate prediction. Similarly, Guo et al. [2011] proposed a pointwise local pattern exploration of the relationships between a focal point and its vicinity by a specific point-comparing layout, providing

the interactive exploration and highlighting anomaly with adjustable sensitivity coefficient to consider domain knowledge. And Brecheisen et al. [2009] has done previous work in parameter sensitivity visualization.

2.3. Multidimensional Visualization

Projection is a commonly used dimension reduction technique for multivariate datasets, useful when visualizing high-dimensional data in 2D or 3D spaces. Scatterplots are intuitive to understand when studying the relationship between two variables. However, projected points may result in clutter and overlap for large and high-dimensional datasets. To solve this problem, Keim et al. [2010] proposed generalized scatterplots to augment the degree of overlap and the distortion. Other augmentations have been proposed by Collins et al. [2009], which enhance the spatial layout of plots with clustering information, and Shneiderman and Aris [2006], which link multiple substrate to superimpose cross-substrate relationships. Another issue of scatterplots is that only a limited number of variables can be shown after projection. The scatterplot matrix is commonly used to enumerate all possible combinations of pairs of variables, but an effective navigation between these different projections is necessary. For example, ScatterDice [Elmqvist et al. 2008] is developed to maintain perception of data during transforming from one projection to another as a mean to reduce mental stress during exploration of the scatterplot matrix. Yet it is still a challenging task to mentally retain the knowledge of variable correlation that one may gain during high-dimensional exploration, especially when the dimensions are high and the correlations are not so strong to be explicitly shown.

A few visual designs have been proposed to compare multiple variables. Eschenbach [1992] used spiderplots to show the relative change in the outcome for a unit change in multiple independent variables. Similarly, [Guo et al. 2011] used star glyphs to show multiple variables and colored glyphs by the significant test to highlight anomalies in the global space.

3. REGRESSION HIERARCHIES

Regression and sensitivity analysis is inherently multiscale. Depending on the degree of interest of the user in a particular region of a high-dimensional space, different regression lines or curves can be fitted.

In the most typical approach, analysts fit a line or curve to data in a 2D projection, a surface in 3D projections, or a hypersurface in higher dimensions. When dealing with data that exhibits multidimensional dependencies, it is often the case that a single line or surface might not explain the data and regression errors are large. Analysts then resort to filtering the data, based on either a priori knowledge about the data or in an exploratory manner throughout the domain, to obtain local regression fits. This process of filtering and refitting induces an implicit *regression hierarchy*. The goal of this article is to present a visual and interactive technique to expose these hierarchies and alleviate the cognitive load required to keep track of the filtering/regression steps. This becomes especially necessary when dealing with multiple dimensions and complex interactions between variables.

3.1. A Motivating Example

To understand the notion of regression hierarchies, consider the example in Figure 1. Clearly, the two variables do not exhibit correlation in a global sense. To analyze the data, we might want to filter that data by the vertical dimension into two intervals (Figure 1). One of the groups, when the variable in the vertical axis is low, shows a linear trend. The other group, in the higher values of the vertical axis, on the other hand, shows no monotonic relationship between the variables. In this case, the user

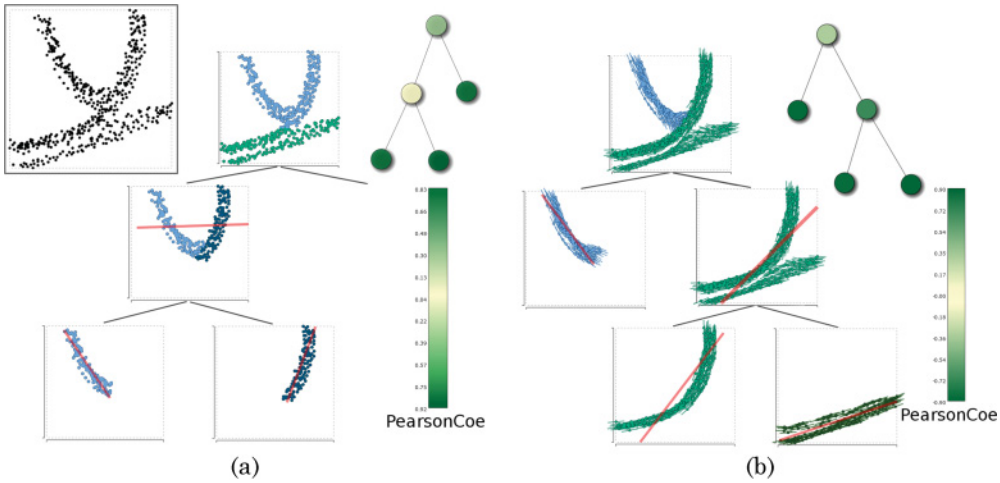


Fig. 1. Regression hierarchies of a synthetic dataset contain regions of the different patterns on the top left scatterplot. (a) A regression hierarchy built by rectangular picking of points. The first split results in a linear trend in green. The second selection is split into two smaller linear trends in blue and dark blue. A node-link graph called the *hierarchy graph* shows the structure of the hierarchy, in which nodes are colored in the green-yellow-green color map according their values of Pearson covariance coefficient. The darker the node, the more coherent the pattern found. Note that the intermediate cube on the second level has a quite low covariance because it exhibits a nonlinear relationship. (b) Another regression hierarchy by sensitivity-aligned selections. In this case, it is possible to select the nonrectangular groups that have similar (nonlinear) trend with ease. Now we see two splits along the different selections, revealing three types of trends at the end. The slope of the long straight line in red on a cube represents the simple linear regression of all points in the cube.

splits the domain along the horizontal axis, which shows two linear fits. This process of progressively refining the selection of data elements to achieve high accuracy is what we call the *regression hierarchy*. In the simplest cases, where data are linear, the regression hierarchy is shallow. In other cases, the regression hierarchy might be several levels deep.

As we have seen, the way to filter data affects the results of regression. To address this issue, we augment the process with sensitivity information, as suggested in Chan et al. [2010]. Consider the same example in Figure 1, where now points are augmented with sensitivity lines, indicating the local trend around a point, as shown in Figure 1(b). These lines reveal nonlinear trends that are difficult to extract using linear fits. In this particular case, the user detects two groups by the orientation of the sensitivity lines in blue and green. Whereas the group in blue shows a monotonic relationship, the group in green can be further split after a quick inspection. This split can be done easily by grouping trend lines in terms of their orientation. The group that has less steep trend lines can be explained by a linear relationship (dark green), whereas the group in light green is clearly represented by a monotonic nonlinear relationship. This process also results in a regression hierarchy, although different from the one obtained using linear regression. The exploration of these different levels of regression helps the user partition the data in a number of ways and efficiently explore complex relationships and groups that are not axis aligned.

Our objective is to enable general regression exploration with a unified structure. We call such a structure the *Regression Cube*, a visual representation of trivariate sensitivity between variables augmented with regression and sensitivity lines. We also present a series of interactive operations on the cube to support the visual analytics process.

4. REGRESSION CUBE

We define the concept of RCs to be the visual representations of trivariate sensitivity information via the simultaneous projection of pairwise sensitivity lines. Formally, the RC is a subspace formed by three orthogonal axes X, Y, and Z, such that each pair of combinations show the sensitivity along the respective plane while simultaneously showing a conventional 3D scatterplot. Such a 3D representation enables analysts to compare three sets of pairwise correlations at once with continuous visual cues of each dimension shared by two facets. Another reason for adopting 3D scatterplot in this work is that 3D projections decrease information loss by allowing better discrimination between data elements [Poco et al. 2011]. Although 3D visualization is susceptible to projective ambiguities, occlusion, and clutter, we mitigate these issues in the RCs with sensitivity lines and visual exploration techniques (Section 4.5). Note that the representation of 3D RC with projections to the three facets has been explored in Tory [2003], and the application of 3D scatterplots was covered by Sanftmann and Weiskopf [2012].

Before we discuss building such a cube, we introduce the notion of regression-oriented sensitivity analytics that can be used on the facets of the cube in exploration, as described in Chan et al. [2010, 2013a].

4.1. Regression-Related Analytics

4.1.1. Estimating Local Sensitivity on a Facet. Sensitivities represent the rate of change of one variable as a function of variations of another. For a pair of 2D points, (x, y), we can estimate sensitivity as the slope of the locally fitted linear regression around a given point, using the linear Taylor expansion of a given variable y with respect to another variable x.

For a point (x_0, y_0),

$$y_i - y_0 \approx \frac{\partial y}{\partial x}|_{(x_0,y_0)}(x_i - x_0). \tag{1}$$

Because we generally only have scattered points (as opposed to say a regular grid or mesh), we must approximate these derivatives using sparse fitting. Let us define a neighborhood around each point consisting of points (x_i, y_i). Then we can estimate the local rate using orthogonal distances solving the quadratic problem:

$$\begin{aligned}
\beta^2 + C\beta - 1 &= 0 \\
C &= \frac{S_{yy} - S_{xx} + \left(S_x^2 - S_y^2\right)/S_w}{S_x S_y / S_w - S_{xy}} \\
S_w &= \sum_i w_i \\
S_x &= \sum_i w_i (x_i - x_0), \;\; S_y = \sum_i w_i (y_i - y_0), \\
S_{xx} &= \sum_i w_i (x_i - x_0)^2, \;\; S_{yy} = \sum_i w_i (y_i - y_0)^2, \\
S_{xy} &= \sum_i w_i (x_i - x_0)(y_i - y_0),
\end{aligned}$$

where w_i is a weight associated with each point in the neighborhood of (x_0, y_0), usually a smoothly varying function that decreases with distance. In our case, we choose a

Gaussian weighting scheme:

$$w_i = e^{-d(\mathbf{x}_0, \mathbf{x}_i)^2}, \tag{2}$$

where $d(\mathbf{x}_0, \mathbf{x}_i)$ is the Euclidean distance between two points in $\mathbb{R}^2$.

We prefer orthogonal regression, where the offset from the point in the neighborhood (x_i, y_i) to the given point (x_0, y_0) is perpendicular to the fitting trend [Weisstein 2009b], as opposed to vertical regression, where the offset is approximated by the distance along the axes directions, such as the ones used in Chan et al. [2010] and Weisstein [2009a], in order to produce regression lines that are less sensitive to the inherent orientation of points along the different axes, and thus are capable of extracting more salient and arbitrarily aligned sensitivity structures. The similar local neighborhood of points in scatterplots was analyzed previously in Sanftmann and Weiskopf [2009], where the quality of the linear fitting was discussed as well.

4.1.2. Computing Global Regression on a Facet. We also approximate a global linear regression line for the facet considering all data points present on the facet and apply the sensitivity estimation shown previously, but with a uniform weight scheme in Equation (2). Some examples of the global regression augmented facet are shown in Figure 1 (red lines) and later in Figure 6 (cyan lines).

4.1.3. Approximating Sensitivity Streamlines on a Facet. We use sensitivities to build three 2D flow plots, one for each facet of the cube, which summarize the most likely flow of the gradient along that dimension, given the derivatives at each point. The representation follows an analogy with fluid dynamics so that if the location of a data record in a plot represents position, then the partial derivatives can be understood as the *velocity* of a particle that has been dropped on that location.

To create a 2D flow plot, we use streamlines that integrate the velocities to simulate the path that a particle would take if placed in this flow field. Since we have a scattered collection of points, we use a *scattered integration* scheme, which computes new directions based on the local velocity (or derivative) sampled from the neighboring data elements on the projection. To integrate the derivatives along the streamline, we used second-order Runge-Kutta. A streamline spanned from a data point p_0 in the 2D domain is a series of connected points found using the following recursive method. For a point p_k, the next point in the streamline p_{k+1} is found as

$$p'_k = p_k \pm 0.5 \times h \times v(p_k) \tag{3}$$

$$p_{k+1} = p_k \pm h \times v(p'_k) \tag{4}$$

$$v(p_k) = [1, \partial y/\partial x], \tag{5}$$

where h is the discretization distance between consecutive points in the streamline and $v(p)$ is the derivative evaluated at point p. We apply this mechanism forward and backward in time (by the positive and negative sign before h) and stop the streamline at the boundaries of the scatterplot. Note that we compute streamlines only for the scattered data elements on the projection—that is, we do not synthesize streamlines for nonexisting grid points.

4.2. Building a Cube

A cube is built as a combination of 2D sensitivity plots and a 3D scatterplot, an example of which is shown in Figure 2. Let us define three variables, X, Y, and Z, between which we want to understand the nature of their interaction. First, for each data point (x, y, z), we uniquely find its position by placing it in its respective coordinates after normalization in the unit cube. Other normalizations, such as the ones based on the standard deviation, are applicable as well. For the simplicity, we assume that each

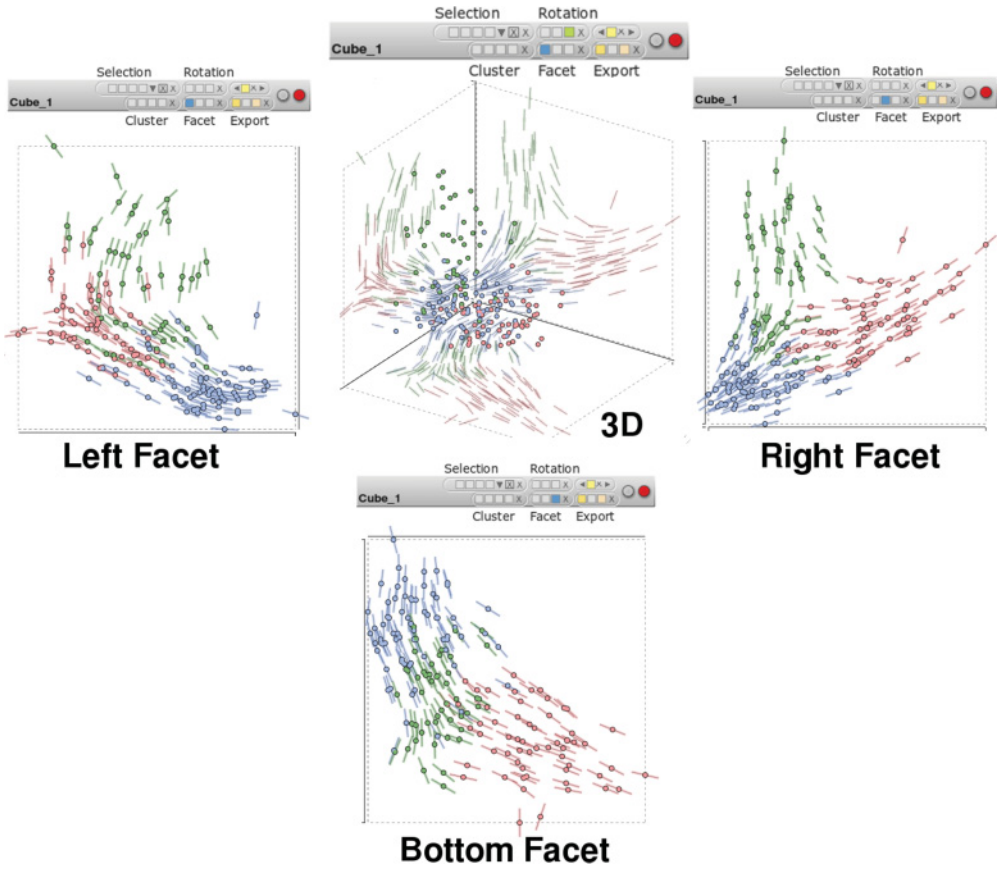


Fig. 2. An RC example with different examining modes: 3D, the left facet, the right facet, and the bottom facet, respectively.

point is defined in the interval $[0, 1]^3$. Note that to avoid blocking the information shown on the facets, most of the 3D scatterplots in cubes are hidden in the following figures.

We then add three facets that represent the projection of the 3D space onto the 2D planes. As a convention, in this article, we called these facets as such:

—The Right Facet (X-Y),
—The Left Facet (Y-Z), and
—The Bottom Facet (X-Z).

Some edges of the front faces of the cube are hidden to avoid distracting occlusions and ambiguities. This leaves us the three inner facets inside the cube as the walls on which to project information. We simply place the sensitivity scatterplots of X-Y, Y-Z, and X-Z variables to their respective facets. An example is shown in Figure 2. Such a representation has a number of advantages. (1) It uses the limited amount of screen space to show the interplay of the three variables, by distorting the three scatterplots according to the human's visual depth perception, and gluing the shared axes between two of the three scatterplots together. (2) Compared to well-known techniques for multivariate data space, such as the scatterplot matrix, RC is able to highlight tri-variate relationships while retaining the ease of use and understanding of 2D sensitivity plots.

(3) Compared to previous sensitivity plots, including our own flow-based scatterplot, the perception of 3D sensitivities is possible via the implicit integration of shadow-like structures in the 2D facets. Visually tracing along a shared axis (say Y) with the two facets of sensitivity lines on its both sides (X-Y and Z-Y), the user would perceive better understanding about the sensitivities of both adjacent variables (X or Z) when the variable at the axis (Y) changes. This eases the possible mental stress of the analysts when trying to retain mentally the insight attained from multiple pairwise correlations that they observed and to relate them in mind when information is presented as separated without linkage in between. (4) RC offers the reliable primitive in regression analysis, the sensitivity lines and streamlines, with which to interact. Coherent high-dimensional patterns can be visually integrated and mentally perceived from multiple sets of sensitivity scatterplots, which may be less ambiguous than the visual metaphor in the 3D space. The 2D facets also provide a solid and familiar interface for the general users to directly interact with the data elements.

4.3. Operations on Facets and Cubes

We provide a number of operations on the facets in an RC to exploit toward exploratory analysis.

4.3.1. Selection. Interacting with the data points in a 3D space in a 2D screen directly can be difficult and ambiguous; thus, we allow the user to brush and select points only on the 2D facets. RC provides the following four ways to select data points, as shown in Figure 3:

(1) Draw a rectangle to enclose the data points.
(2) Move the cursor to a target data point on a facet. It will automatically select all other points with similar slope of the short regression line to the target point.
(3) Move the cursor to a target data point so that it selects all other points that fall into the curvy band area around the sensitivity streamline of the target point
(4) Brush across the sensitivity streamlines to select the data points from which they originate.

Note that the brushing described in (4) is different from traditional brushing on scattered points but similar to the technique presented in Hurter et al. [2009]. Instead of painting on top of points scattered in 3D space, we check the stroke drawn by the cursor and see what streamlines it crosses. Then we select data points from which the crossed streamlines are computed. Such data brushing allows selecting data of complex shapes and various sizes. An example is shown in Figure 3(d), where we brush the streamlines by a stroke to select the data points of the positively increasing trend.

4.3.2. Classification. Classification provides a higher level of abstraction of the dataset, and it helps identify important patterns of the sensitivity streamlines on the facet. Moreover, classification partitions the dataset into groups that possibly reveal different patterns. This operation stems from the observation that points in the same high-dimensional feature are likely to have similar sensitivities. Thus, we expect sensitivity lines to be smooth but sufficiently different in terms of shape as they start to diverge in the high-dimensional space. Although the selecting technique like brushing streamlines allows the user to manually classify a set of streamlines, we can take advantage of the automatic classification on each facet that suggests grouping of data points to enhance efficient selections. Moreover, we use the interactive legends, which lead to faster perception of the mapping between data values and visual encoding [Riche et al.

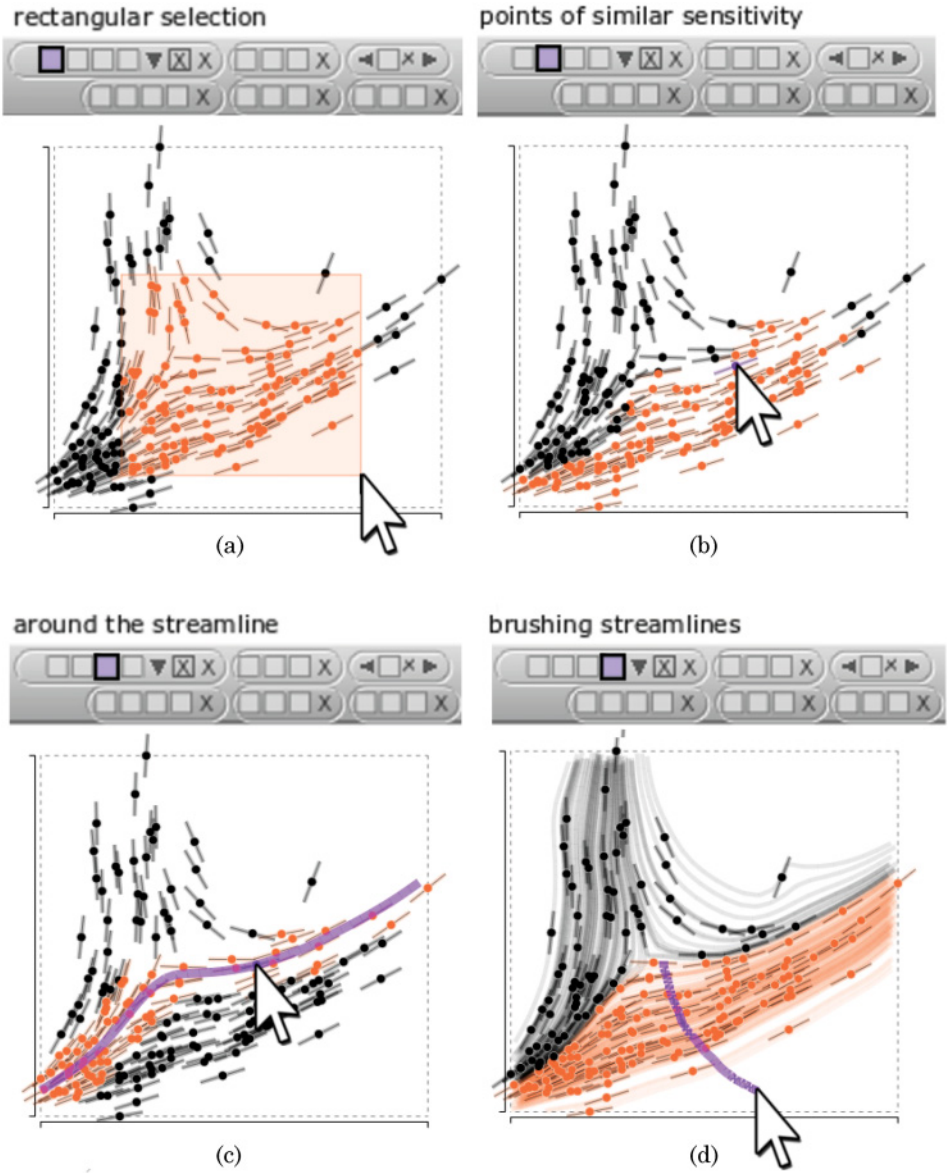


Fig. 3. Four types of selection provided in RC.

2010], to show the classification for the user to choose from. We provide four k-mean clustering techniques based on different metrics of data, as shown in Figure 4:

(1) The *position* of the data on the facet (x and y).
(2) The *slope* of the sensitivity line along two axes (dx and dy).
(3) The *position of the samples* along the sensitivity streamline that originates from the data point (x_i, y_i).
(4) The *curvature* on the samples along the sensitivity streamline computed from the data point (an angle ϑ_i). In this case, we apply the curve clustering algorithm

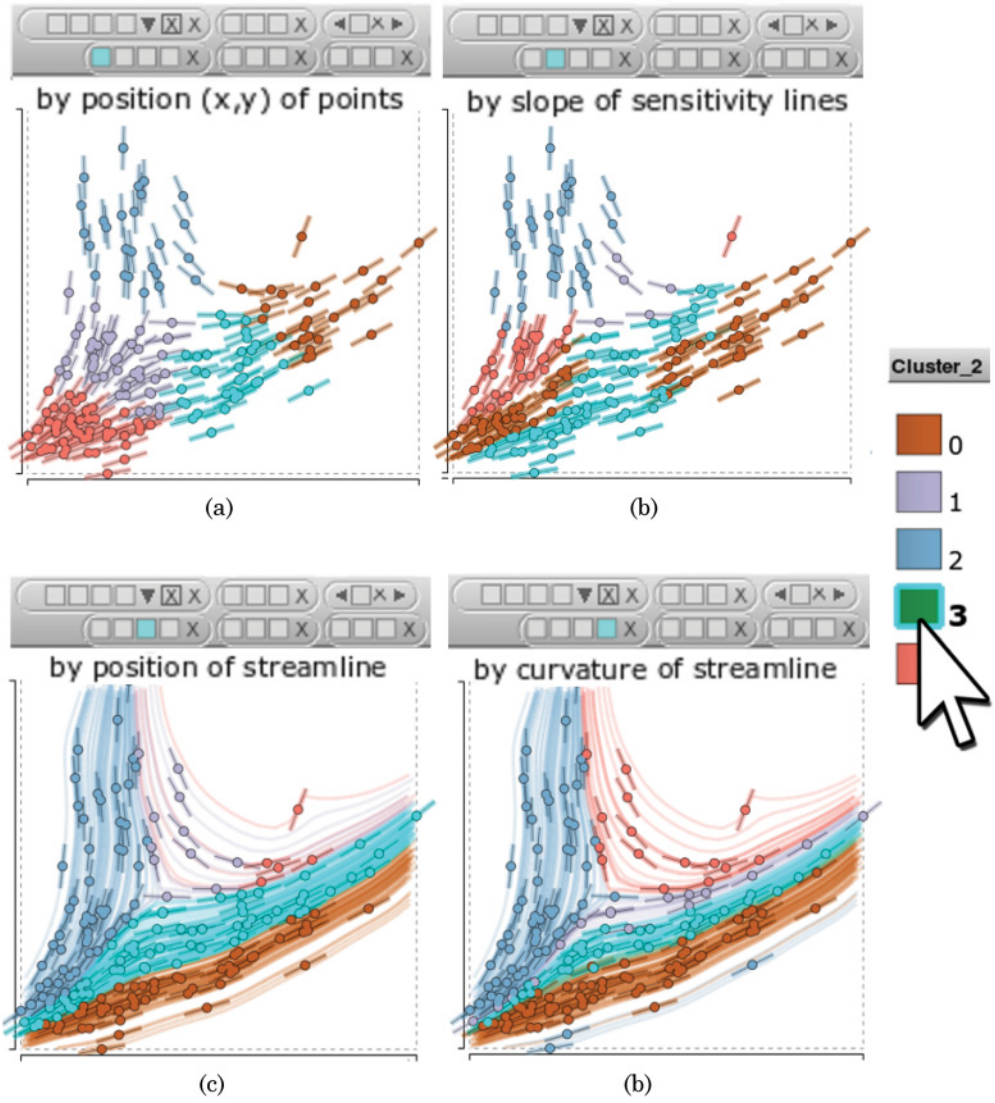


Fig. 4. The k-mean clustering from the four types of distance function in RC. It suggests the group of the data elements from which to select.

described in Wei et al. [2010] to differentiate sensitivity streamlines at a global scale. Since we use a rather synthetic notion of sensitivity streamlines and do not necessarily comply with physical requirements, we do not consider aspects such as repetition, scaling, and tolerances in their algorithms, which rarely occur in the streamlines that we construct with a large enough sampling radius. To compare the similarity between two sensitivity streamlines, we sample the curvature values along the streamline and compare the two streamlines by their summation of the total variance in curvature from all sampled points. Finally, we use single linkage clustering to obtain the last k clusters after merging clusters of high similarity. The algorithm works qualitatively well to provide classes with sufficiently different behaviors, as shown in Figure 4(d).

4.3.3. Rotation and Reprojection. We can rotate the 3D interface in RC to examine the point of interest and disambiguate projection artifacts, and we can switch to different 2D facets iteratively to inspect the trivariate relationship in the multidimensional dataset.

4.4. The Process of Visual Exploration and Pattern Finding

The value of the RCs rests in the interactive process that users can perform on them in order to navigate through a dataset and iteratively filter out data to discover trend patterns. With the various selecting and clustering mechanisms that we have provided, of which some are regression-aligned interactive filtering techniques, users are guided by the visual cues to take advantage of their visual perception. Going through this process, the user gradually drills down to the meaningful patterns that would otherwise be hidden at once. Yet it is still the user who decides on the data content of the filtering, so different users may find different regression hierarchies on the same dataset. The guiding visual cues of RC in the process include:

—The local linear regression indicated by the short lines on data points
—The integrated local linear regression shown by the long sensitivity streamlines extended from data points
—The global regression visualized by a long straight line on a set of selected points
—Pearson covariance coefficient, which is shown by the label and color of the nodes on the regression hierarchy that represents the predictive relationship of the two variables on a facet of the cube

The pattern discovery process with RC is an iterative three-step procedure as follows:

(1) *Locate the projection*. Choose from the left, right, or bottom facet of the RC that you are working on. A good strategy is to pick a facet for which the working cube's Pearson covariance is smaller than that of its parent, if it has a parent, so that the hierarchy will have increasing covariance from high-level nodes to low-level leaf nodes, and thus the patterns found in the lower-level nodes will exhibit a coherent trend on the facet.
(2) *Perform selecting and exporting*. The users can apply one of the four data selection techniques described in Section 4.3.1, or they can select data by a higher abstraction from the groups computed from classification as described in Section 4.3.2.

    The selected data points are highlighted in all linked cubes, so the user has visual feedback about the selection. The quality of the selection can be verified by visual cues such as (1) the aggregated directions of short regression lines (i.e., sensitivity lines), (2) the coherent sensitivity streamlines, or (3) whether the direction of the most of sensitivity lines on the points is consistent with the direction of the global regression line of this selection.
(3) *Compare cubes by visual cues*. Note that the sensitivity analytics is approximated locally around the neighborhood of the data point, so its visual cues highly depend on the constitution of the data points in the cube and need to be updated accordingly when the neighborhood changes. Thus, all of the regression-related visual cues are recomputed in the new cubes, including the short sensitivity lines, sensitivity streamlines, and the global linear regression line of the plot. Then we can evaluate whether the selection performed in the previous step exhibits a meaningful correlation pattern or not by comparing the regression visual cues between the new cube and the other cubes in the hierarchy. Another intuitive way to evaluate the selection is Pearson covariance mapped as the colors of nodes in the hierarchy, as it summarizes how strongly the two variables on the facet are linearly dependent

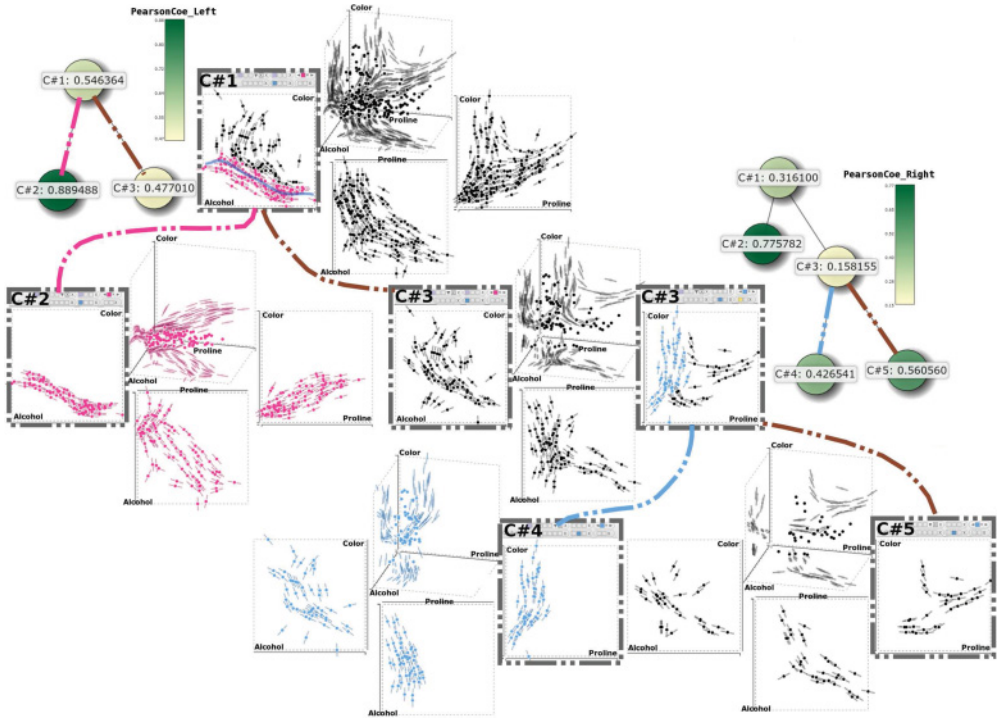


Fig. 5. Visual exploration by RCs. Two regression-aligned partitions are shown in this example. (1) The negatively correlated trend (pink) found on the left facet on the root node. (2) The positively correlated trend (blue) selected on the right facet. The regression hierarchy is shown by the node-link diagram in green and yellow. The greener the node, the higher the covariance between the two variables on the facet. The pink trend has quite strong correlation ($p = 0.8894$) between Alcohol and Color, whereas the blue trend has medium strong correlation ($p = 0.7757$).

for the subset data in the cube. Then users can go back to the first step, where they may switch to other facets and keep building the hierarchy until interesting trend patterns are found in cubes on the leaf cubes in the hierarchy.

4.5. An Example of Visual Exploration and Pattern Finding

In this section, we show how to utilize RC to explore the multidimensional data and extract sensitivity patterns that may be hidden in subsets of the dataset, with an example in Figure 5.

As described in Section 4.4, we propose a hierarchical process of pruning a dataset, refitting flow lines, and refining the queries based on newly acquired insight. Typical setup for the hierarchical exploration on RC consists of the following: (1) set up a root RC of three variables and plot sensitivity lines and then (2) select and export data by regression-aligned operations to reveal trends on the subset of data.

We can think of this process as the *linking* of several cubes, where each link represents a "select and export" that progressively refines the data to smaller regions, and during the process it creates a *hierarchy* of subsets of the data. Note that since the sensitivity lines are computed from the remained data points in the cube, they are locally performed analytics independent from other cubes, and thus the different correlation patterns can be revealed from a different subset of data.

In the figure, we show all three facets and the 3D view of a RC on an illustrative hierarchy with the dotted binary branches. We visualize this illustrative hierarchy

by the node-link diagram in the green-yellow color mapping from Pearson coefficient values of the cube. This diagram is laid out automatically when any new RC is added to the hierarchy. For each cube, we show its different views (from top in the clockwise order): 3D, Right facet, Bottom facet, and Left facet.

4.5.1. Generation of the Root RC. We start the process by showing the root RC with all data points in the dataset. To this end, we rely on the user to decide which three variables in the dataset are used to initialize the root RC. (In practice, we can suggest to users some possible choices of the three projection variables by analyzing the distances between points.) Because we start with nonsegmented data, the sensitivity lines shown on the facets of the root RC are approximated from the whole dataset, as shown on the top root node in Figure 5.

4.5.2. Selection and Exporting/Linking. In the example, two regression-aligned selections are performed (see the two binary splits drawn in the dashed lines). The first one is selecting a negative correlated trend (pink) around a streamline on the Left facet, and the other one is done among the remaining data points in Cube 3 from selecting a strongly positive correlated trend (blue) on the Right facet.

In each split, when the selection is finalized, it creates a new child cube to contain the selection and another child cube for the unselected data; this process is called *export*. We implicitly link the parent and child cubes to build the hierarchy to keep track of the user workflow in the pattern discovery process. In child RC, the sensitivity estimates are recomputed for this subset, since now the constitution of the data in the cube is different from its parent cube. Two types of data export can occur from a selection:

(1) *Exclusive export*. An RC is created to contain the selected points. When the selection coincides with features, this process effectively reveals local correlations. In the example, the exclusive export produces the left child RC.
(2) *Negative export*. An RC is created to hold data points that are not in the selection. This type of data manipulation is handy when we want to remove the anomaly or outliers. In the example, the negative export created the right child RC.

Note that the binary split constraint in the pattern discovery process can be relaxed so that we have more flexible way in partition the dataset, resulting in the regression hierarchy as a multiple branches graph instead of a binary tree, as shown in Section 5.

4.5.3. Comparison between RCs and Refinement. When the new RCs exhibit high covariance, we may infer that the data points in the cubes can be better described as a single feature, such as Cube 2 (the pink pattern) and Cube 4 (the blue pattern) in Figure 5. The covariance is reported by the regression hierarchy graph in the green-yellow color mapping. The darker the node, the higher covariance of two projection variables for the data points in the cube. Note that the covariance is calculated based on the projected scatterplot on the facet. Thus, after the first partition occurs, two new nodes representing the two new RCs from exclusive and negative export are added to the hierarchy and their covariance values are updated. Then, when the second "select and export" takes place, another two children nodes are added.

This evolving regression hierarchy graph reveals good patterns with strong covariance after two times of regression-aligned filtering, since both of the left leaf nodes have high covariance comparing to its predecessor. This indicates the progressively improvement of the coherent pattern finding.

After "select and export," the user can refine the exported selections by removing part of the original selection, or incrementally add more node to the new RC. This is useful for the large-scale datasets or for the regions with a high density of points where it is difficult to perform the accurate selection in a single selection. For example, in the

right children RC of the second split, Cube 5, the data points close to the origin could be removed from Cube 5 and be re-exported to the pink pattern found in Cube 2, to be part of the positive correlation on the right facet of Cube 2.

From the example presented earlier, one can see that the interactive visual exploration in RCs is an iterative and repeatable process in which analysts can move data points from one to another and see how different selections result in different patterns. For instance, if the noise or outliers are removed from an RC, then this RC is updated to show the trend of the remained data points. If we repeatedly export selected data to new cubes and refine the selection, we can analyze the behavior of a particular group of points with carefully selected data. Another possibility that can be achieved by such an interactive regression analysis is that we can use trial and error to add or remove data from the RC of a discovered pattern and see how "sensitive" it is to the data points from other patterns or to the noise.

5. EXAMPLES

We have applied RC to the analysis of several multidimensional datasets. Our aim is to show that sensitivity-augmented 3D RC helps us uncover trends and groups in 2D plots with little need of interaction.

5.1. Automobile MPG

The Automobile MPG dataset concerns the city-cycle fuel consumption in Miles Per Gallon (MPG) [Quinlan 2011]. It contains 398 records of the cars made between 1970 and 1982. Five of the eight attributes are continuous: MPG, Weight, Acceleration, Horsepower, and Displacement. We use the predicted variable MPG as the output variable and investigate its relationship with other dependent variables, as well as the correlation between pairs of variables.

First, we construct an initial RC with MPG in the vertical axis, the output variable of the dataset, and pick Weight and Horsepower as the other two variables to understand their correlation to MPG as well as their own interdependency. The 3D view of the root RC on the top right in Figure 6 reveals the strong negative correlation between Weight and MPG and between Horsepower and MPG. The sensitivity of these two variables under investigation is on the bottom facet X-Z, which shows the strongly positive correlation. The RC cube depicts the correlations of the three pairs of variables simultaneously.

Now we show that the RC enables more detailed local analysis and visual validation of the perceived information. We have already seen the strongly positive correlation on the bottom facet. To further understand the correlation between Weight and Horsepower, we switch the projection to the bottom facet. In Figure 6, data points are colored by the number of cylinders of the automobile, varying from three to eight. Groups consisting of four (yellow), six (green), and eight (purple) cylinders are dominant in the dataset. Since all data show a global pattern of positive correlation between Horsepower and Weight, we create a hierarchy for these groups by the exclusive export to see their within-group correlation. Besides the sensitivity flows, we also show the simple linear regression as a straight blue line on the fact in each cube to show the summarized correlation that is laid on the centroid of data points. Purple and yellow groups exhibit a similar regression pattern as in the parent cube, shown by both the blue regression line on the facet and the sensitivity short lines on the data points. The green group, however, reveals a slightly different trend than the simple positive correlation, which indicates that Horsepower is irrelevant to Weight sampled from the data points in this group. This would not have been revealed without filtering out data points in other cylinder groups.

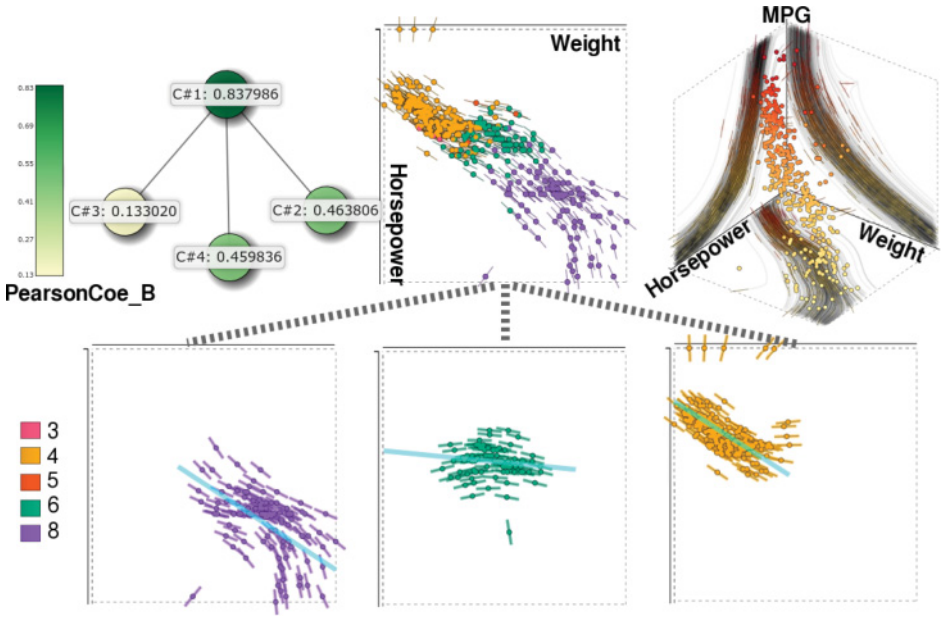
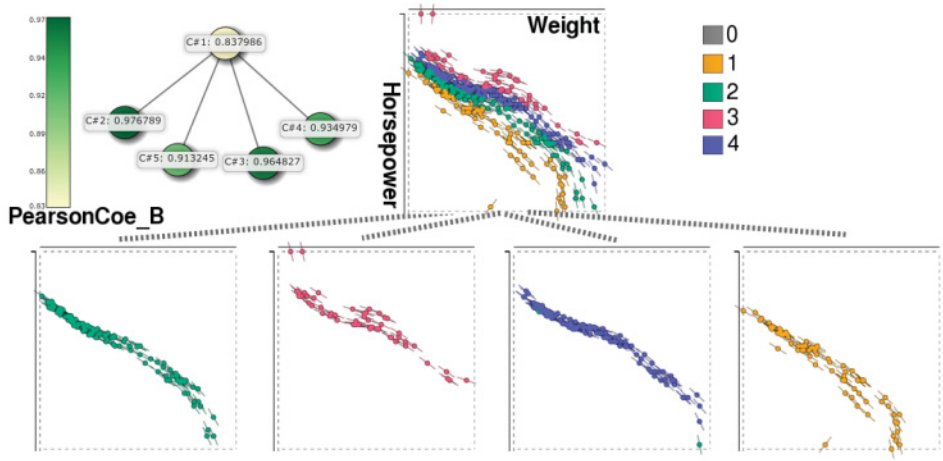


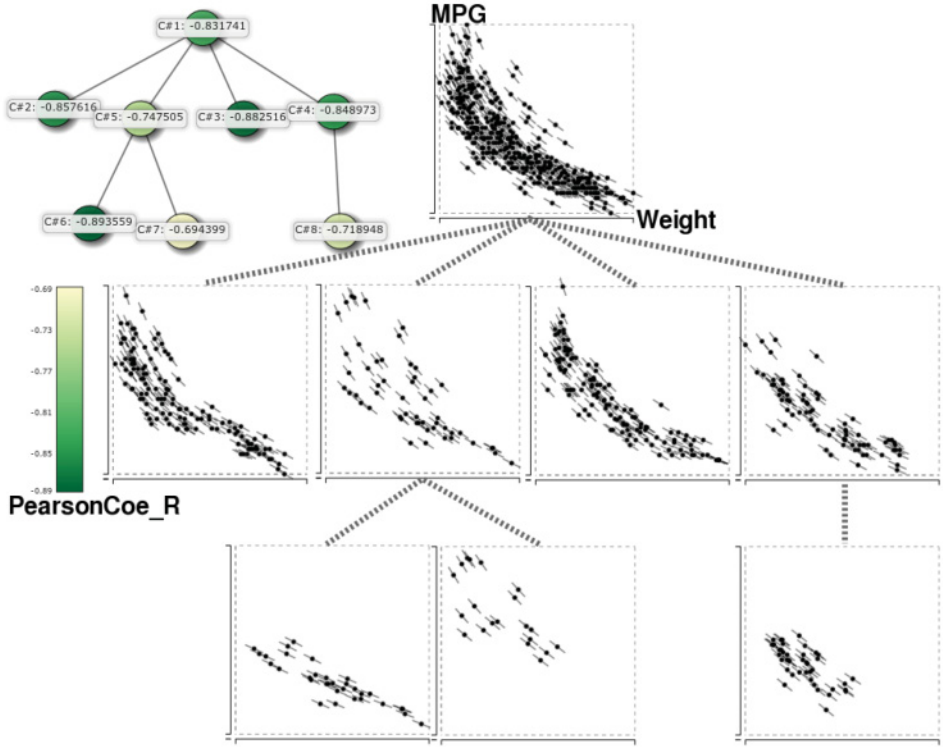
Fig. 6. Automobile MPG. A regression hierarchy built from selecting by the categorical variable (the number of the cylinders) on the bottom facet. The five categorical values are also shown in the legend at bottom left (see text).

Then we revisit the same root RC with other selecting mechanisms to continue our exploration in the 3D space spanned by the three variables. As shown in Figure 7(a), we cluster the dataset by their sensitivity streamlines and export all dominant groups to individual RCs. All of the clusters exhibit a positive correlation between Weight and Horsepower. Note that all four groups end in a trend that is almost a perfectly linear correlation, which in facet we can visually cross validate by the very large magnitude of the covariance on the regression hierarchy graph. The reason these children RCs are more coherent is that, although the root RC explicitly shows a positively correlated trend, there is still a certain degree of divergence of the sensitivity lines between clusters. After validating the positive correlation between Weight and Horsepower on the bottom facet of the root RC, we switch to the right facet to see if these groups of data points exhibit any trend between Weight and MPG, as shown in Figure 7(b). The root cube shows a clear negatively correlation between them, and the four cubes from clustering demonstrate the same trend as well. Even the two sparsest cubes (the second and the fourth cubes from the left on the second level) show the same trend, which means the dependency of the two variables is a solid relationship regardless of the data partition. Then we try to split the two sparse trends to see if we can find more evidence about the negative trend. For the second cube, we find a weaker negative linear correlation on its left child. For the fourth cube, however, the further split does not show any different pattern.

As shown in both Figure 6 and Figure 7(a), we see that with the ability to export a cluster of data records exclusively or negatively in the linked RCs, one can easily conduct a comparative regression analysis between clusters and expand the hierarchy with different data partition depending on which groups are included. To summarize, this case study shows that RCs are effective in showing multiple pairs of trend simultaneously in a global sense, and they enable hierarchical exploration when relationships are not evidently linear or functional. By filtering and refitting data, RCs enable *visual*



(a) Starting from the right facet. Four new cubes are exported from streamline clustering results.



(b) Switch to the left facet and improve the covariance to C6; C7 and C8 are the residuals noises.

Fig. 7. Regression hierarchy built from (Weight, MPG, Horsepower). (a) The hierarchy is built from the clustering results of the sensitivity streamlines. All four main clusters exhibit a positive correlation on the facet, and they also have higher covariance comparing to the root RC. (b) We switch to the right facet to see if these clustering groups exhibit any trend between the other pair of variables: Weight and MPG. The root cube shows a negative correlation between them, and the four cubes below the root RC from clustering demonstrate the same trend as well. We try to build another level in the regression hierarchy from two of the cubes, but only the left-most one among the three cubes shows an obvious trend that can be numerically verified by the darker nodes in the regression hierarchy graph.

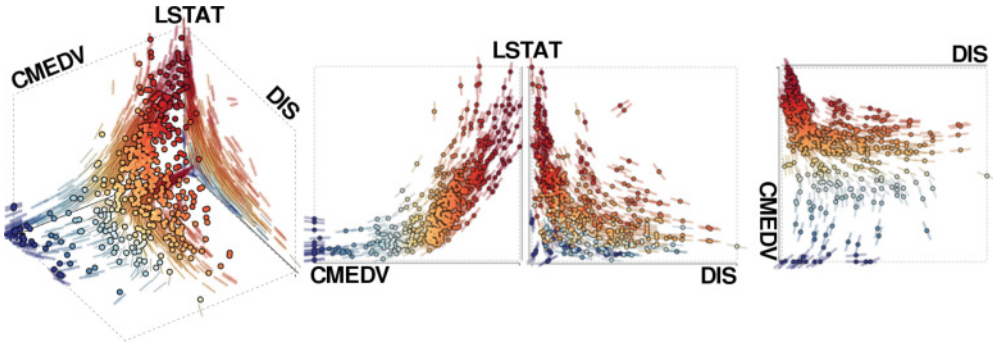


Fig. 8. Boston housing price. RC shows three sets of pairwise correlation, among which some data records in navy blue on the bottom facet do not follow the monotonically increasing trend.

validation of regression models and classification results as an ideal way to increase the robustness of the visual analytic process.

5.2. Boston Housing Price

The Boston housing price dataset is a collection of environmental, geographic, economic, and social variables to predict the median value of housing price in the Boston metropolitan area [Harrison and Rubinfeld 2011], which contains 506 records and 15 continuous variables. Some variables include geographic information, such as DIS, the weighted distances to five Boston employment centers; LSTAT, the percentage of the lower status of the population; and CMEDV, the price of the house as the output variable.

We set up an RC to compare the three variables as shown in Figure 8, where the Z axis is assigned as the output variable CMEDV. The data points are also colored by their prices, from expensive (navy blue) to inexpensive (maroon). At first glance, one can quickly point out the two negatively correlated relationships among the three pairwise relationships: LSTAT versus CMEDV, and LSTAT versus DIS. Both of them are reasonable compared to what we usually would expect. On this cube, we also notice that the bottom facet of DIS and CMEDV exhibits an interesting correlation, indicating that for the houses that are inexpensive, the housing price and DIS are positively correlated. For the more expensive units (points in light blue and navy blue), however, the correlation of DIS and the housing price is more complex. The units in navy blue, for example, have small DIS but are extremely expensive. From these observations, we would like to start the exploration on the bottom facet.

After we switch the bottom facet, we apply clustering by the locations of the samples on the sensitivity streamlines and then export the four clusters exclusively to the new cubes. In Figure 9, the first level under the root cube consists of these the clusters. The color of the nodes representing these clusters in the node-link graph on the right suggests that this partition enhances the coherency of the trend in each individual cube, as all children nodes have a larger covariance value than the root node. The color of the nodes suggests that most of the cubes show quite strong positive correlations between the two variables, except Cube 2 on the left which has a slightly smaller covariance value, and therefore we may not need to go beyond this level in the hierarchy.

The second and the third rows in Figure 9 are the screenshot of the other two facets, the left and right facets, and their corresponding covariance coefficients shown on the hierarchy graph on their right. Note that the relationship between the two variables is negatively correlated for these two facets, which is also shown in the left and right facets in Figure 8 before the clustering partition. The color mapping in the legend is

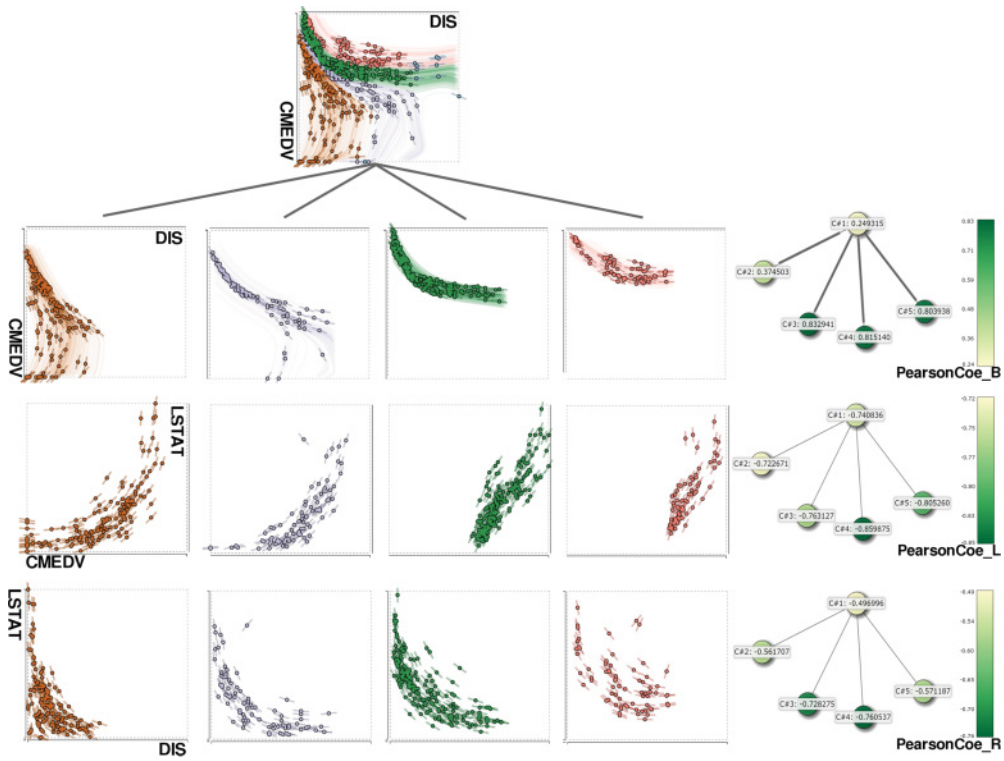


Fig. 9. Boston housing price. Regression hierarchy is built from the four clusters computed on the bottom facet in Figure 8. The three rows under the root node show the bottom, left, and right facets of the children cubes accompanied by their hierarchy graphs that show the covariance coefficients. The four patterns in the children RCs have rather high covariance values in all three facets.

upside down for the negative covariance coefficient, so the dark green indicates the large magnitude of the negative covariance, which still means the strong correlation. The viewing modes of the views of the linked RCs are synced, so when the users switch the facets, they can easily compare between multiple RCs.

In this example, we see that the interactive RCs make it easy for us to compare different projections of the different partition of the dataset at once. Users can switch the viewing modes back and forth to visually validate the differences between small multiples. Meanwhile, the up-to-date visual cues, such as the color showing the covariance, or the slope of the sensitivity lines indicating the magnitude of the correlation, provide instant feedback during the interactive visual exploration. We demonstrate in this example how the interactive exploration augmented with the regression analytics of low computational cost can help the users to efficiently compare visual analytics of different partitions of the dataset.

6. EVALUATION

In this evaluation, we assessed how RCs may influence the process whereby users drill down to discover possible patterns in the dataset. Particularly, we wanted to examine two main designs of RCs—3D interface and sensitivity functionality—to see whether a general user would benefit from them in the pattern discovery process.

The 3D view control is useful and helps to mitigate occlusion via depth cues. However, visualizing multivariate datasets with a 3D interface has been shown to be undesirable

Table I. The 10 Seed Functions Being Interpolated in Preparing 300 3D trends in Section 6.2.2

LIN	$y = x$	LIN1	$y = 1.5x$	MLIN	$y = 1 - x$	LOG	$y = \frac{\log(ax^2+1)}{\log(a+1)}$	SIN	$y = \sin 2\pi x$
QUAD	$y = x^2$	QUAD2	$y = 0.3x^2$	MQUAD	$y = 1 - x^2$	CUB	$y = x^3$	EXP	$y = e^{-x^2} - 1$

due to the limitation of the 2D displays and the perception bias between individuals. Therefore, to validate the 3D design of RCs, we compared the performance of pattern discovery with and without 3D interface in a controlled study.

The sensitivity-aligned interactions, such as selection and clustering by sensitivity lines or streamlines, are provided in RC. When the user moves the cursor to change the point of interest on a 2D facet using these interaction techniques, the sensitivity analysis is computed on-the-fly and shown to the users during the pattern discovery process. With the sensitivity functionality, RCs suggest the portion of the data that exploits a local pattern according to the user's query by the cursor. However, even with the suggestion by the sensitivity analysis, sometimes it takes some effort to combine several selections on different 2D facets to finally achieve a desirable data selection, especially when the embedded high-dimensional features are complicated. We thus also like to assess the usability of RC's sensitivity functionality in the user study.

6.1. Hypotheses

Our initial hypotheses with regard to the two main designs of RCs were as follows:

—We expected users to perform fairly well in RCs with the sensitivity functionality such as sensitivity-aligned selection and clustering.
—We speculated that users would benefit from the 3D interface in RCs because one can examine the shape of the 3D trend better by rotating the cube rather than merely looking at the three projected facets.

6.2. Experiment Setup

6.2.1. Subjects. We recruited seven participants, one female and six males, among which five were graduate students and two were postdoctoral researchers. All of the participants were computer science majors. The ages of the subjects ranged from 27 to 31 years of age with an average of 28.7 years. None of the participants had prior experience using RC.

We showed the interactive RC to the participants using the same laptop with a 15.6- inch display and an optical mouse as the input device, and we recorded the performance of each participant in each condition on the same computer.

6.2.2. Dataset. We used a subset from the synthetic dataset that we created previously for another user study on how people interpret 3D trends with 2D plots [Chan et al. 2013b]. This synthetic data collection consists of 300 3D trends from interpolating any pair of the *two seed functions* Y_1 and Y_2, with an interpolating function w and a noise level N_α. Each synthetic 3D trend was created as follows:

$$Y(x, z_w) = z_w Y_1(x) + (1 - z_w)Y_2(x) + N_\alpha(x), \tag{6}$$

where Y_1 and Y_2 are the two seed functions, w denotes the interpolating function (linear or sigmoid), z_w and $(1 - z_w)$ are the interpolating weights of the seed functions, and N_α is the noise level (low, medium, or high). Table I summarizes the seed functions (LIN, LIN1, MLIN, QUAD, QUAD2, MQUAD, LOG, SIN, CUB, EXP), which are also shown in Figure 3 in our previous paper [Chan et al. 2013a]; four examples of the 3D trends can also be found in Figure 4 of that paper.

When a 3D trend is randomly chosen to be used in the test, the data points in the 3D trend are imported to the root RC and visualized in the 3D space according to the

Table II. Four Conditions of the Two Factors

		Sensitivity Functions	
		Disabled	Enabled
3D Interface	Disabled	**A**	**C**
	Enabled	**B**	**D**

(x, y, z_w) values. The values of the interpolating weights of the two seed functions, z_w and $(1 - z_w)$, range from 0.0 to 1.0. We can determine which seed function the data point is closer to from its interpolating weight z_w value on the Z axis: when $z_w > 0.5$, it is closer to the seed function Y_1; when $z_w < 0.5$, it is closer to the seed function Y_2. Note that we assign each data point to the seed function that it is closer to as the "ground-truth" of which trend it belongs to. This information is used to evaluate the quality of the patterns in Section 6.3.1.

According to the results of our previous user study [Chan et al. 2013a], we ranked the synthetic 3D trends by the average error of interpreting them on 2D plots. This ranking corresponds to the level of the difficulty to recognize the two seed functions perceptually, from the easiest to the most difficult ones. We picked the datasets ranked the top one-third as the easiest to interpret for this evaluation.

6.2.3. Design. We designed the user study to evaluate the ability of RC to help users discover high-dimensional patterns. The two independent variables the we set to test in the experiment are *3D interface* and *sensitivity-related functions*. Each variable has two levels: enabled and disabled. The 2×2 design creates four experimental conditions, as shown in Table II. In each condition, the participant used RCs to find patterns by building the regression hierarchy from a 3D dataset. Certain interactions were enabled or disabled according to the experimental conditions:

—*Rows: 3D interface*. This variable indicates whether the rotation navigation in RCs is enabled or not. We put this factor under consideration to determine whether and by how much 3D navigation had impact on the users' ability to understand the hidden patterns in the data.
  —*Disabled*. Users can only look at the dataset from the orthogonal projections—that is, the three 2D facets (Left, Right, and Bottom) of RCs.
  —*Enabled*. Besides the three facets of RCs, users can also freely rotate the cube to any viewing angle to see how the data points scattered in the 3D space.
—*Columns: Sensitivity-related functions*. RC provides two visual clues for the sensitivity analysis of the two projection variables: the sensitivity short lines on the data points, and the sensitivity streamlines that follow the direction of the local sensitivity to travel through the projection space on the 2D facet. Based on these two visual clues, the sensitivity-aligned selection and clustering are provided in RC for users to interact with the data more directly and intuitively. We consider this factor in this evaluation to examine whether sensitivity-related interactions enhance the pattern discovery in RC:
  —*Disabled*. Users can select data points by the union of multiple rectangular selections (as shown in Figure 3(a)); they can cluster data points by the distance function estimated by their (x, y) locations (as shown in Figure 4(a)).
  —*Enabled*. The sensitivity lines and flows are shown in RCs. The three ways of selecting data (as shown in Figure 3(b–d)) and the three ways of clustering data (as shown in Figure 4(b–d)) are both enabled in RCs.

6.2.4. Procedure. Before starting the tests of the four conditions, the participants practiced the prototype software in a tutorial session where we walked through all of the navigation and interaction functions provided by RC. Following the tutorial session,

the participants went through all four conditions in a randomly decided order. In this way, we wished to counterbalance our study design to avoid the learning effect. In each condition, the dataset of 3D trend used was randomly chosen, and the root node cube contained all of the data points in the dataset. From the root node, we asked the participants to build a regression hierarchy in the interactive RCs. In the process of building the hierarchy to discover patterns, the participants decided when to finish the condition at which they were working, and they proceeded to the next condition.

All participants were informed that each dataset in the conditions contained "two patterns" with noise. Their objective was to recognize these patterns and separate them out in RC. We revealed this information to the participants because we desired that our subjects have a certain understanding about the element of our 3D dataset and what to find. During the tests, participants were allowed to ask us any question if they needed any help in finding a particular function that they wanted to use.

6.3. Performance Metrics

In order to collect the *dependent variables* of the two factors that we tested, we recorded the following information on each participant for each of the four tests:

—The information about the dataset that we randomly chose for this test, as mentioned in Section 6.2.2, including two seed functions from which the dataset was interpolated, the interpolant, the noise level in interpolation, and the difficulty level of the 3D trend (estimated by the average error of the trend in our previous user study).
—Number of interactions. We recorded the number of times the following interactions occurred in a test and reported them in Section 6.4.2:
  —Selecting data with one of the four selection methods
  —Selecting data with one of the four clustering methods
  —Switching to either Left, Right, or Bottom facets
  —Rotating a cube
  —Exporting the selection to the new cubes
—The time the participants spent to create a split on the hierarchy when they decided to export the selection to the new cube. We discuss this performance metric in Section 6.4.3.
—The graph of the regression hierarchy and Pearson coefficient values of each node in the graph for all three facets at the end of the test. The latter is one of the ways to measure the quality of the patterns in Section 6.4.4.
—The data points that each leaf RC contains. We took this information to evaluate the quality of the patterns found in these cubes in Section 6.4.4.

6.3.1. Quality of the Patterns in Cubes. Besides the dependent variables, we applied three performance metrics to measure the quality of the patterns found by our participants in the leaf cubes: Pearson Coefficient, Pattern Fit, and Number of Correct Points. The first metric measures the linearity of the data in the leaf cubes, while the latter two metrics measure the similarity between the patterns in the leaf cubes and the two seed functions that generated the 3D trend in the dataset:

—*Pearson Coefficient* measures the linearity of the data points that were exported to the cube. We calculated this metric on the right facet of each leaf cube.
—*Pattern Fit (cubewise score)* measures whether all of the leaf cubes in a regression hierarchy do a good job of differentiating the two seed functions from each other. For each leaf RC, we calculated two *fitness values* f_1 and f_2 to the two seed functions Y_1 and Y_2. Each fitness value indicates how much the data in the cube fits to that seed functions. It is calculated by a weighted summation of all data points, where the closer a given data point is to the seed function i, the more weight it would contribute

to the f_i value of that seed function. The larger fitness value of f_1 and f_2 is the *fitness score F* of this cube. We assigned the cube to the seed function Y_i with the larger f_i value, believing that the participants perceived these data points exported to this cube as the pattern of the seed function Y_i. If an RC contains only the data points closer to the same seed patterns ($z_w > 0.5$ or $z_w < 0.5$ for all the points in the cube), this RC would have a high fitness score F. Then we calculated the average of the fitness scores of all leaf cubes in a regression hierarchy as the *Pattern Fit*.

When a participant assigned the data points of the same seed function into more than one cube, as long as these cubes were "clean" without any other data points from the other seed pattern, the participant would score from the clean separation of trends, instead of being punished for partitioning the data points in the same pattern into more than one cube.

—*Number of Correct Points (pointwise score)* measures the quality of the data decomposition. We calculated *the number of the correctly classified data points* (n_c) for each leaf cube by comparing the following two groupings of data points:

(1) As mentioned in Section 6.2.2, the *ground truth* of which of the two seed trends the data point belongs to is determined by the z_w value of the data point ($Y_1 : z_w > 0.5$; $Y_2 : z_w < 0.5$).
(2) In calculating Pattern Fit, each leaf cube was assigned to one of the two seed functions according to the fitness values (f_1 and f_2) of the cube.

We calculated the pointwise score by the average of n_c of all leaf cubes in a regression hierarchy as the *Number of Correct Points*.

6.4. Results

We structure the results of the user study in this section as follows. First, we report ANOVA analysis of all of the variables we collected in Section 6.4.1 to have an overview of the impact of the two factors—3D interface and sensitivity functions—on the variables that we collected. Then we report the occurrence of the different types of interaction of each participant using a heatmap for each condition in Section 6.4.2 so that we can visually compare between the participants, the interaction types, and the conditions at once. For this particular variable, we are more interested in gaining insights about user behavior in the pattern discovery process in different conditioned RCs than in comparing how the participants performed in tests. The reason for choosing such a nonstandard way to report the interaction variable is that since the numbers of the interaction highly depends on the personal preference of each of the participants, the standard descriptive statistics would require the advanced weighted scheme or normalization on this variable to compare between the subjects. Finally, we report the dependent variable *Time* in Section 6.4.3, and the three metrics that we propose to measure the quality of the patterns found by the participants in the RCs in Section 6.4.4: Pearson Coefficient, Pattern Fit, and Number of Correct Points. For each of these four performance metrics, we use the standard descriptive statistics: mean and standard deviation.

6.4.1. ANOVA. Using the information that we collected from the tests of four conditions, we performed a *within-subjects repeated-measures two-way ANOVA*. We used the two factors on pattern finding tasks—3D interface and sensitivity-related functions—to understand how these two factors may affect the dependent variables that we collected. We also assessed if any of the two factors caused a statistically significant difference in the results. We summarized the results from the ANOVAs as follows:

(1) For some indexes, such as completion time and number of interactions, the sensitivity-related function was the main effect, meaning that sensitivity-related functions had a significant impact on how long it took to find patterns, and on how

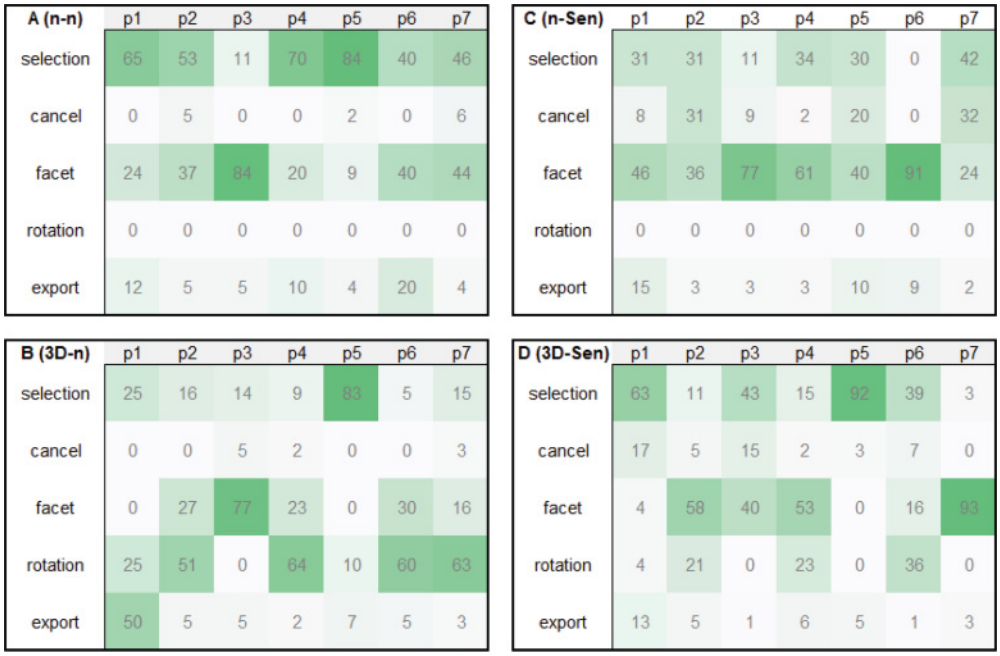


Fig. 10. The four heatmap tables of the interactions that the participants performed in each condition. In each table, we compare the five types of interaction in the rows and the seven participants in the columns.

much interaction took place in the study. More specifically, when sensitivity-related functions were enabled, these following indexes changed significantly:

—The total time spent to finish the task was significantly longer (from 110.15msec to 236.41msec, where $F(1,6) = 5.2$, $p = 0.0318$).

—The average time spent to export a selection to a new cube was significantly longer (from 76.97msec to 142.30msec, where $F(1,6) = 5.63$, $p = 0.026$).

—The number of times that the participant canceled a selection was significantly larger (from 0.64 to 7.00, where $F(1,6) = 4.51$, $p = 0.0442$).

—The number of times that the participant switched to one of the 2D facets was significantly larger (from 8.42 to 20.42, where $F(1,6) = 5.75$, $p = 0.02$).

(2) We did not find any variable on which the 3D interface had a main effect.

Additionally, we did not find any index showing significant interaction between the two factors, suggesting that the interaction between the two RC design features, the 3D interface and sensitivity-related functions, was minimal or insignificant.

6.4.2. Interactions. In Figure 10, we show the four heatmap tables of the interactions that each participant performed in each test: Condition A (3D disabled, Sensitivity disabled), B (3D enabled, Sensitivity disabled), C (3D disabled, Sensitivity enabled), and D (3D enabled, Sensitivity enabled). In each table of the figure, each cell corresponds to the interaction of a participant (column) and the type of interaction (row). Note that the number in a given cell indicates the percentage of the selected type of interaction compared to the total number of the interactions that the selected participant performed in the given test, not the actual number of times that the interaction occurred. This method of display gives us an overview on the frequencies of the different types of interaction so that we can visually compare the participants, interaction types, and conditions all at once. In Conditions A and C where rotating was disabled,

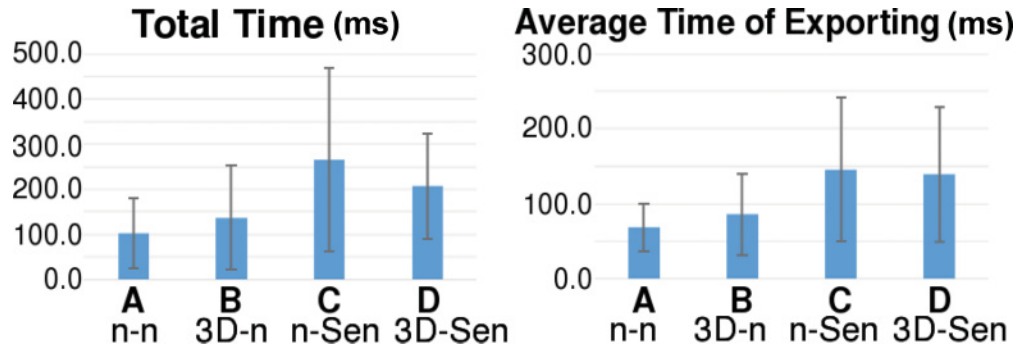


Fig. 11. The average performance in time under four conditions.

users relied more heavily on the rectangular selections and facet switching in their attempt to understand the data. In Condition B, the participants rotated the cubes quite often, presumably in order to identify the high-dimensional features in the data. If sensitivity functions were enabled, disabling rotation (Condition D vs. C) caused some users to switch between facets more often in order to try out different sensitivity selections (p1, p3, p4, p5, and p6). If rotation was enabled, as in Conditions B and D, some participants were stimulated by the three methods of sensitivity selection and performed more selection. This is identified via the noticeable increase in selections (p1, p3, p4, p5, and p6) and cancellations (p1, p2, p3, p5, and p6), indicating more attempts to select different sets of data points.

When only one of the factors, 3D or Sensitivity, was enabled (Conditions B and D), the behavior of users changed, and they preferred more navigation (i.e., switching the projection and rotating the view, shown in the third and fourth rows of the tables) than data manipulation (i.e., selecting data points shown in the first row). These results suggest that the sensitivity-related functions and 3D interface stimulated the users' interest in exploring the data space. One might expect this trend to continue when both factors were enabled, but this was not necessarily the case. In particular (Condition B vs. D), if rotation was already enabled, enabling sensitivity resulted in stronger user preference for data manipulation (the first row) over navigation (the third and the fourth rows combined) (p1, p3, p4, p5, and p6). Similarly (Condition C vs. D), if sensitivity functions were already enabled, enabling rotation also resulted in stronger user preference for data manipulation over navigation (p1, p3, p5, and p6). By visually comparing multiple interaction heatmaps, we gained insights about the users' behavior in different conditioned RCs.

6.4.3. Time. For the performance in time, we had information about how long it took the participants to create a split in the hierarchy and the total time to finish the test. Figure 11 shows two charts of the average performance measurements in time: the total time to finish the test on the left, and the time to create a split in the hierarchy and export the selection to the new cubes on the right. The standard deviation of each condition is marked in the chart as well. In both charts, we see that when sensitivity functions were enabled (Conditions C and D), participants spent more time exploring the data and testing different selection techniques provided than in the test where these functions were disabled (Conditions A and B). Since we did not ask participants to "finish the test as quickly as possible" but rather instructed participants to "take as long as they needed until they were satisfied with their patterns," our results indicated that users spent more time when the sensitivity functions were enabled. This indicated to us that RC engages more active exploration of the data by users.

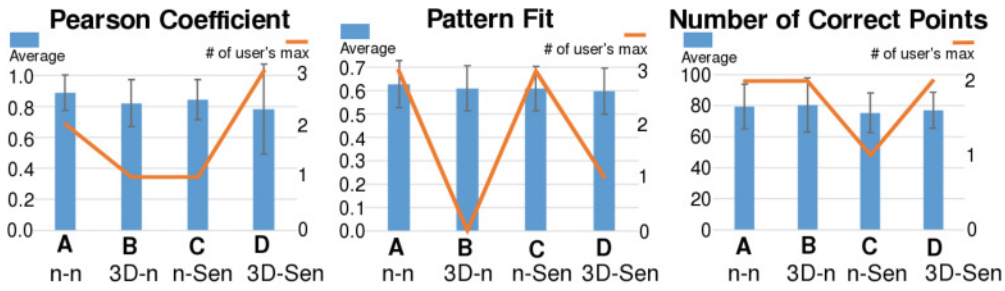


Fig. 12. Performance in the quality of the patterns found in cubes.

6.4.4. Quality of the Patterns in Cubes. For each regression hierarchy that a participant created in a test, we measured the leaf cubes by the following three metrics concerning to what degree the decomposition of the data points in the cube matched one of the seed functions that generated the dataset. The three metrics were Pearson Coefficient, Pattern Fit, and Number of Correct Points.

Figure 12 presents the three measurements of the data decomposition in RC. The charts show the average measurements from all seven participants (blue bars), the standard deviation (the error range on the top of the bar), and the number of participants who had their best performance among all four conditions (the orange lines). From the bar charts, we see that all three measurements indicate a decreasing trend from Condition A to Condition D, suggesting that using RC does not guarantee the best pattern discovery results. One limitation of this analysis is that, owing to the small number of participants, ANOVA in Section 6.4.1 found no significant difference between the four conditions for any of the three metrics of performance. Although we are unable to draw a solid conclusion, we can say that the error bars show some degree of similarity between the performances of the four conditions across all three metrics. We observed that there is a large variation in scores between users (especially for the Pearson Coefficient metric in Condition D), and we considered the possibility that a clearer pattern might be visible after removing the influence of the lowest scores among users. Thus, we examined individual log files and identified four extremely low performance scores from four different participants on four different conditions. Specifically, we plotted the raw number of individuals who had their best score on a give condition (shown by the orange lines in Figure 12.

Taking both analyses into account, we drew the following conclusions:

—The four conditions were similar in terms of all three performance metrics.
—The participants did not necessarily perform their best with all of the functions enabled (Condition D).
—The average performance was the best for Condition A (both 3D and sensitivity interactions disabled) for all three metrics.

6.5. Discussion

Here are a few observations that we have made about our participants' use of RC during this evaluation study:

(1) Most of the participants tended to rotate the initial RC if the 3D interface was enabled. On the tests where the 3D interface was enabled, participants rotated more than 10 times on average. This shows that although the direct manipulation of the data points, such as selection or clustering, was only limited to be performed

on the 2D facets, the participants still resorted to the 3D interaction in order to get a general impression of the distribution of the points in the dataset.

(2) Some participants used the clustering function for the purpose of labeling the data points with color so that the data points would be easier for them to identify when they rotated the cube or switched between facets. This reminds us that, besides the direction shown by the sensitivity lines and flow, the users may prefer another visual metaphor to help them locate the data points, such as the point color.

(3) Some participants pointed out that the synchronized RCs with the same viewing angle and the view size helped them quickly compare the patterns spanned by the data points in the cubes. Thus, we believe that the small multiples with interactive data manipulation help users visually compare the groups that they just created from the whole dataset.

(4) Some participants preferred the traditional rectangular selection over the sensitivity-related selections not only because they were more familiar with such selection but also because of the confidence they had when they knew what part of the dataset would be selected when they drew the rectangle to enclose the data. On the contrary, when using the automated selections by similar sensitivity or around the flow line, they needed to move around data points by trial and error to see whether the autoselected points were what they wanted to select. Brushing data by flow lines sometimes had a similar problem where they brushed the flow lines and some undesired data points were selected. Therefore, some of the participants focused on the rectangular selection, even though it required much more manipulation than the automated selections that we provided. Specifically, the users often struggled to drag multiple rectangles to enclose all of the data points desired.

We also received some valuable suggestions from our participants about the design of RC. Three participants acknowledged that RC provides an easy way to select data points in the 3D space. Some participants wished there was an "undo" function when building the regression hierarchy. Most of them wanted more flexible ways to create the selection, such as by a combination of different selections, or partial deselection.

The decreasing quality of the patterns found (Pearson Coefficient, Pattern Fit, and Number of Correct Points) when sensitivity functions were enabled (comparing Conditions A-C or B-D) appears to have more to do with how much the users are familiar with RC. As described in the previous sections, RC can aid in interactive visual analytics. However, for users to fully benefit from the interactive data partitioning process, they must gain background knowledge about the sensitivity analysis method and have sufficient training to become familiar with the interactions provided by our system. That is, it became clear to us that more extensive tutorials and examples would be required for the users to fully understand the regression analysis and what interaction to use under each circumstance. On the other hand, the significantly longer data exploration time when users were provided with the interactive visual analytic tools indicates that RC encourages them to explore the high-dimensional data in multiple ways for gaining insights.

Although the added interactivity and complexity of the selection contributed to the increase in completion time for 3D- and sensitivity-augmented RCs, we found an interesting behavior regarding the quality metrics. The Pattern Fit and the Number of Correct Points seem to be similar for all cases, but the Pearson Coefficient is only highest for the straightforward 2D scatterplot (Condition A). This seems to indicate that the linear relationships are best uncovered with a 2D scatterplot, whereas nonlinear relationships are harder to discover. Although it is known from our previous

work that sensitivity lines and 3D can help to identify these nonlinear or linear relationships, this benefit comes at the cost of extra complexity to the users. Our user study suggests the presence of a steep learning curve that could hamper some applications of RC and, potentially, other visual analytic tools for pattern identification; nevertheless, in situations where patterns become higher dimensional and more complex, this cost may be amortized by the benefit of discovering nonlinear, functional relationships in the data. Although the scatterplot remains a familiar, easy-to-use visual tool, the augmentation with RCs, regression hierarchy, and sensitivity lines can potentially be a useful complement in the case when linear relationships are hidden from simple projections, or rectangular selection fails to separate data in higher dimensions.

7. CONCLUSIONS AND FUTURE WORK

In this article, we have introduced an interactive regression-aligned visual analytics technique for multidimensional data. Our work is based on constructing RCs from three sensitivity augmented scatterplots. These RCs show correlation between pairs of variables by three visual cues: (1) the short sensitivity lines for the local linear regression, (2) the sensitivity streamlines for the integrated trend, and (3) the long straight line for the simple global linear regression on the presented data points in the cube.

Additionally, we present the notion of regression hierarchy. Regression hierarchy is obtained by progressively fitting regression lines and then filtering data by either the conventional selections or the regression-aligned selections. The users can refer to the sensitivity information in building such hierarchies. This process produces regression analytics that is not available with traditional plots, and the process of the iterative regression is visualized explicitly.

RCs unify three pairwise correlations at once so that analysts can study how multiple variables react to the change of the other variable by visually tracing shared axes in the cube. The three facets of RC display the correlation patterns between any two variables. Subsequently, we can formulate hypotheses of the relationship between three variables and their rates of the change. On each facet, users can not only perform the various regression-aligned or simple rectangular selections, but they can also compute the different regression-aligned clusters. These regression-aligned interactions provide higher-level abstractions of the whole dataset. Then these selections and the clusters of data points can be exported to other RCs for further investigations. The hierarchy visualization is another insightful visual cue during the regression-aligned filtering process. It provides the comparative covariance by the color of the nodes and the values of the Pearson correlation coefficient on the labels of the nodes. During the iterative process of exporting and linking RCs, users can quickly refer to the covariance to evaluate the quality of the trends in the new cubes so that they can make the proper decisions as to whether the regression-aligned filtering should proceed or not. We evaluated RCs via two examples in Section 5 and an empirical user study on the 3D visual representation and sensitivity functionality of RCs in Section 6.

As datasets increase in size and the correlations between variables expand in complexity, RCs with visual partitioning and coupling of data offer a promising solution for interactive visual exploration and insight. In our future work, we intend to develop an intelligent system that recommends to users which interactions to execute. Additionally, we plan to develop automated mechanisms to select the proper regression sampling size for the local linear regression. We also anticipate that GPU acceleration will facilitate the interactive visual exploration of large data. To more extensively evaluate the usability of RCs in pattern discovery, we plan to conduct a larger-scale online evaluation to collect the performance metrics from more participants. We also plan to

recruit more subjects with sufficient knowledge on statistical analysis to see how RCs could help the experts in regression analysis.

ACKNOWLEDGMENTS

This work was sponsored in part by the U.S. National Science Foundation through grants CCF-1025269, CCF-0811422, and IIS-1255237, and also by the U.S. Department of Energy through grants DE-CS0005334 and DE-FC02-12ER26072.

REFERENCES

Leon M. Arriola and James M. Hyman. 2007. Being sensitive to uncertainty. *Computing in Science and Engineering* 9, 2, 10–20.

Sven Bachthaler and Daniel Weiskopf. 2008. Continuous scatterplots. *IEEE Transactions on Visualization and Computer Graphics* 14, 6, 1428–1435.

Scott Barlowe, Tianyi Zhang, Yujie Liu, Jing Yang, and Donald Jacobs. 2008. Multivariate visual explanation for high dimensional datasets. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*. 147–154.

Wolfgang Berger, Harald Piringer, Peter Filzmoser, and Eduard Gröller. 2011. Uncertainty-aware exploration of continuous parameter spaces using multivariate prediction. *Computer Graphics Forum* 30, 3 (June 2011), 911–920.

Pavel Berkhin. 2006. A survey of clustering data mining techniques. In *Grouping Multidimensional Data*. Springer, 25–71.

George E. P. Box and Norman R. Draper. 1987. *Empirical Model-Building and Response Surfaces*. Wiley.

Ralph Brecheisen, Anna Vilanova, Bram Platel, and Bart ter Haar Romeney. 2009. Parameter sensitivity visualization for DTI fiber tracking. *IEEE Transactions on Visualization and Computer Graphics* 15, 6, 1441–1448.

Dan G. Cacuci, Mihaela Ionescu-Bujor, and Ionel Michael Navon. 2005. *Sensitivity and Uncertainty Analysis, Volume II: Applications to Large-Scale Systems*. CRC Press.

Stuart K. Card, Jock D. Mackinlay, and Ben Schneiderman. 1999. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann.

Karen Chan, Andrea Saltelli, and Stefano Tarantola. 1997. Sensitivity analysis of model output: Variance-based methods make the difference. In *Proceedings of the 29th Conference on Winter Simulation*. IEEE Computer Society, Washington, DC, 261–268.

Yu-Hsuan Chan, Carlos D. Correa, and Kwan-Liu Ma. 2010. Flow-based scatterplots for sensitivity analysis. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*. 43–50.

Yu-Hsuan Chan, Carlos D. Correa, and Kwan-Liu Ma. 2013a. The generalized sensitivity scatterplot. *IEEE Transactions on Visualization and Computer Graphics* 19, 10, 1768–1781.

Yu-Hsuan Chan, Carlos D. Correa, and Kwan-Liu Ma. 2013b. User Study Dataset of GSS. http://vidi.cs.ucdavis.edu/projects/RegressionStudy/.

Yu-Hsuan Chan, Carlos D. Correa, and Kwan-Liu Ma. 2013c. Video Demo of Regression Cubes. Retrieved March 18, 2014, from http://vidi.cs.ucdavis.edu/projects/RegressionStudy/RegressionCubes2013.wmv.

Michael Chau, Reynold Cheng, Ben Kao, and Jackey Ng. 2006. Uncertain data mining: An example in clustering location data. In *Advances in Knowledge Discovery and Data Mining*. Springer, 199–204.

H. Christopher Frey and Sumeet R. Patil. 2002. Identification and review of sensitivity analysis methods. *Risk Analysis* 22, 3, 553–578.

Christopher Collins, Gerald Penn, and Sheelagh Carpendale. 2009. Bubble sets: Revealing set relations with isocontours over existing visualizations. *IEEE Transactions on Visualization and Computer Graphics* 15, 6, 1009–1016.

Graham Cormode and Andrew McGregor. 2008. Approximation algorithms for clustering uncertain data. In *Proceedings of the 27th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. ACM Press, New York, NY, 191–200.

Carlos D. Correa, Yu-Hsuan Chan, and Kwan-Liu Ma. 2009. A framework for uncertainty-aware visual analytics. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*. IEEE Computer Society, 51–58.

Norman R. Draper and Harry Smith. 1998. *Applied Regression Analysis (Wiley Series in Probability and Statistics)*. Wiley.

Niklas Elmqvist, Pierre Dragicevic, and J.-D. Fekete. 2008. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE Transactions on Visualization and Computer Graphics* 14, 6, 1539–1148.

Ted G. Eschenbach. 1992. Spiderplots versus tornado diagrams for sensitivity analysis. *Interfaces* 22, 6, 40–46.

Andreas Griewank, Jean Utke, and Andrea Walther. 2000. Evaluating higher derivative tensors by forward propagation of univariate Taylor series. *Mathematics of Computation of the American Mathematical Society* 69, 231, 1117–1130.

Zhenyu Guo, Matthew O. Ward, Elke A. Rundensteiner, and Carolina Ruiz. 2011. Pointwise local pattern exploration for sensitivity analysis. In *Proceedings of the IEEE Conference on Visual Analytics Science and Technology*. 131–140.

D. Harrison and D. L. Rubinfeld. 2011. Boston Neighborhood Housing Price Dataset. Retrieved March 18, 2014, from http://lib.stat.cmu.edu/S/Harrell/data/descriptions/boston.html.

Trevor Hastie and Robert Tibshirani. 1986. Generalized additive models. *Statistical Science* 1, 3, 297–318.

Jon C. Helton, Jay Dean Johnson, Cedric J. Sallaberry, and Curt B. Storlie. 2006. Survey of sampling-based methods for uncertainty and sensitivity analysis. *Reliability Engineering and System Safety* 91, 10, 1175–1209.

Christophe Hurter, Benjamin Tissoires, and Stephane Conversy. 2009. FromDaDy: Spreading aircraft trajectories across views to support iterative queries. *IEEE Transactions on Visualization and Computer Graphics* 15, 6, 1017–1024.

Ronald L. Iman and Jon C. Helton. 1988. An investigation of uncertainty and sensitivity analysis techniques for computer models. *Risk Analysis* 8, 1, 71–90.

Michiel J. W. Jansen. 1999. Analysis of variance designs for model output. *Computer Physics Communications* 117, 1, 35–43.

Dong Hyun Jeong, Caroline Ziemkiewicz, Brian Fisher, William Ribarsky, and Remco Chang. 2009. iPCA: An interactive system for PCA-based visual analytics. *Computer Graphics Forum* 28, 3 (June 2009), 767–774.

Daniel A. Keim, Ming C. Hao, Umeshwar Dayal, Halldor Janetzko, and Peter Bak. 2010. Generalized scatter plots. *Information Visualization* 9, 4, 301–311.

Dorota Kurowicka and Roger M. Cooke. 2006. *Uncertainty Analysis with High Dimensional Dependence Modelling*. Wiley.

Wang Kay Ngai, Ben Kao, Chun Kit Chui, Reynold Cheng, Michael Chau, and Kevin Y. Yip. 2006. Efficient clustering of uncertain data. In *Proceedings of the 6th International Conference on Data Mining*. IEEE Computer Society, Washington, DC, 436–445.

Jorge Poco, Ronak Etemadpour, Fernando Vieira Paulovich, T. V. Long, Paul Rosenthal, M. C. F. Oliveira, Lars Linsen, and Rosane Minghim. 2011. A framework for exploring multidimensional data with 3D projections. In *Computer Graphics Forum* (Vol. 30). Wiley Online Library, 1111–1120.

Ross Quinlan. 2011. Automobile MPG Data Set. Retrieved March 18, 2014, from http://archive.ics.uci.edu/ml/datasets/Auto+MPG.

Nathalie Henry Riche, Bongshin Lee, and Catherine Plaisant. 2010. Understanding interactive legends: A comparative evaluation with standard widgets. *Computer Graphics Forum* 29, 3 (August 2010), 1193–1202.

Harald Sanftmann and Daniel Weiskopf. 2009. Illuminated 3D scatterplots. *Computer Graphics Forum* 28, 3, 751–758.

Harald Sanftmann and Daniel Weiskopf. 2012. 3D scatterplot navigation. *IEEE Transactions on Visualization and Computer Graphics* 18, 11 (November 2012), 1969–1978.

Jonathon Shlens. 2005. *A Tutorial on Principal Component Analysis*. Systems Neurobiology Laboratory, University of California at San Diego.

Ben Shneiderman and Aleks Aris. 2006. Network visualization by semantic substrates. *IEEE Transactions on Visualization and Computer Graphics* 12, 5, 733–740.

Václav Šmídl and Anthony Quinn. 2007. On bayesian principal component analysis. *Computational Statistics and Data Analysis* 51, 9, 4101–4123.

Ilya M. Sobol. 2001. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and Computers in Simulation* 55, 1–3, 271–280.

Robert Spence. 2007. *Information Visualization: Design for Interaction* (2nd ed.). Prentice Hall.

Yutaka Tanaka. 1994. Recent advance in sensitivity analysis in multivariate statistical methods. *Journal of the Japanese Society of Computational Statistics* 7, 1, 1–25.

Steven K. Thompson and George Arthur Frederick Seber. 1996. *Adaptive Sampling*. Wiley, New York. NY.

Melanie Tory. 2003. Mental registration of 2D and 3D visualizations (an empirical study). In *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*. IEEE Computer Society, Washington, DC, 49.

Jishang Wei, Chaoli Wang, Hongfeng Yu, and Kwan-Liu Ma. 2010. A sketch-based interface for classifying and visualizing vector fields. In *Proceedings of the IEEE Pacific Visualization Symposium*. 129–136.

Eric W. Weisstein. 2009a. Least Squares Fitting. Retrieved March 18, 2014, from http://mathworld.wolfram.com/LeastSquaresFitting.html.

Eric W. Weisstein. 2009b. Least Squares Fitting. Retrieved March 18, 2014, from http://mathworld.wolfram.com/LeastSquaresFittingPerpendicularOffsets.html.

Yoshihiro Yamanishi and Yutaka Tanaka. 2005. Sensitivity analysis in functional principal component analysis. *Computational Statistics* 20, 2, 311–326.

Di Yang, Elke A. Rundensteiner, and Matthew O. Ward. 2007. Analysis guided visual exploration of multivariate data. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*. 83–90.

Received June 2012; revised December 2013; accepted January 2014