Critiquing Multivariate Visualization Systems
A Task-based Review of Four Software Programs

1. Introduction

One of the primary purposes of information visualization is to allow viewers the opportunity to gain insight into represented data. However, properties of visualizations themselves, such as the clarity information depicted, as well as the efficacy and ease with which users may go about interacting with dynamic visualizations, factor heavily into the nature, amount, and quality of insight viewers may derive from these displays. Assessments of how completely individual visualizations promote viewer insight can therefore be difficult to establish. One way to address this issue involves the examination of common data in a variety of different visualization tools. This technique can help surface for analysis some common influential properties related to visualization display composition, and thus make assessments of how successfully different visualization techniques promote viewer insight easier to state with confidence.

In the following report I explore some potentially influential properties of information visualization displays by examining the output of four commercially produced tools (reviewed briefly in section 2). I evaluate the tools in section 4 on the basis of how effectively their output allows me to confirm and/or deny several hypotheses related to some preexisting data sets. (Descriptions of both data sets and hypotheses appear in section 3.) I also examine each tool in terms of its ability to reveal interesting aspects of the data sets that I might otherwise have overlooked. Finally, I express critiques of each tool in terms of visualization strengths and weaknesses they exhibit, as well as positive and/or negative qualities associated with each of their user-interfaces.

2. Overview of Visualization Tools

This report reflects my examination of four software programs, specifically InfoZoom, Eureka, Spotfire, and SeeIt. Each program functions as a tool for generating interactive visual depictions of preexisting data that users may modify, adjust, and explore at their whim. All tools differ in terms of supported visualization formats (e.g. scatter plot, table, bar chart, etc.), display and control appearances, and support for user-data interaction. InfoZoom arranges data into a table format and allows users to see different amounts of data in varying degrees of detail by executing zooms via mouse clicks. Eureka also collects data into a table, but emphasizes depicting all cases simultaneously so that the
resulting display resembles a cross between a spreadsheet and a bar chart. Users of Eureka may pick out individual cases of interest with a mouse and enlarge (i.e. "focus") them for easier viewing. Seelt primarily presents data in the form of a 3-D bar chart (2-D charts are also available for summary purposes). Users may move around, explore details, and manipulate the appearance of the visualization display using controls in the program interface. Finally, Spotfire visualizes data in a variety of formats: scatter plot, 3-D scatter plot, bar chart, and pie chart. Users may mark and view information pertaining to depicted data in Spotfire directly by clicking on graphs using a mouse, or they may utilize a variety of query tools for sorting and filtering purposes embedded in the program interface.

3. Data Sets and Hypotheses

With each tool I focused on examining two preexisting data sets: "Cars" and "Grocery store survey." The former, "Cars," contains 407 cases of 10 variables, and the latter, "Grocery store survey," contains 5164 cases of 8 variables. Of the "Cars" data set I specified the following hypotheses:

1. Americans make heavier cars than do the Japanese.
2. Considering all cars produced between model years 1977 and 1979, the car with the highest MPG rating possessed the least amount of horsepower.

Of the "Grocery store survey" data set I specified the following hypotheses:

1. The average purchase amount spent by females was higher than the average purchase amount spent by males.
2. Women who earned incomes between eighty and ninety thousand dollars per year on average spent more money than did women who earned incomes between ten and twenty thousand dollars per year.

4. Examination of Visualization Tools

4.1 InfoZoom

I began my detailed examination of InfoZoom by loading the "Cars" data set. The program seemed to encounter some trouble interpreting the file, due to the fact that two attributes (AKA variables; displayed in the program as rows) consisting primarily of numerical information also contained some non-numerical data. I was unsure how such inconsistencies might impact the overall visualization of the "Cars" data set, but quickly found that I could correct all formatting confusion by right-clicking my mouse on problematic attributes and selecting the "Format..." option. Once I managed to configure data inside the program correctly, I looked over the relatively large-scale table that first
appeared and became a bit disoriented. The size of cells in the table did not seem to correspond with the size of the data set listed in the upper-left corner of the main program window ("406 of 406 Objects; 11 Attributes). I then noticed scroll bars bordering the far left and bottom portions of the main display and became horrified by the prospect of having to manually move through nearly forty five hundred cells of data! Fortunately, InfoZoom's "Overview" feature serves as a quick and enjoyable way to generate a large-scale, relatively low-resolution image of an entire data set. When viewed in "Overview" mode, the Cars data set began to make more sense. I could read all attributes listed clearly in the leftmost column of the main display and quickly associate them with their corresponding values on the right. InfoZoom’s "Overview" display of attribute values allowed me approximate rough locations of particular data objects while simultaneously depicting large scale trends of numerical data (e.g. "Weight").

In evaluating the first hypothesis related to cars, "Americans make heavier cars than do the Japanese," I found InfoZoom’s "Overview" feature useful, but not immediately as I initially expected. A quick glance at the overview revealed that both attribute rows corresponding to "Weight" and "Origin" depicted values in increasing order from left to right across the display. "Weight" values increased in number, while "Origin" values increased according to the alphabet (i.e. A-E-J). To compare weights of American and Japanese cars, I decided first to eliminate unnecessary data to avoid confusion. Hiding all attributes except "Weight" and "Origin" resulted in a clean, two-row stacked chart well suited to reveal a simple comparison. I then eliminated European cars from the "Origin" row using the "Exclude" command in the appropriate spot, and pushed the "Compress" button to see a different view of the two-row overview chart I ended up with. The resulting display revealed a comparison of histogram lines indicating that my hypothesis was correct (see fig. 1)! I later eliminated the one data case with which no weight value was associated using a mouse-initiated delete command.
I confirmed the second hypothesis related to car MPG rating and horsepower using a method quite similar to the one described above. In general, the process of obtaining an overview of a data set, excluding extraneous data, hiding unnecessary data, and compressing the resulting chart into a visualization that could be sorted according to whichever attribute I used as a basis of comparison proved effective in evaluating hypotheses associated with the "Cars" data set. After sorting a reduced, focused data set, I found I could add context and make unexpected associations by using the "Show Hidden" command to display formerly hidden attributes. For example, replacing formerly hidden attributes while evaluating the second car hypothesis and sorting displayed data by "Origin" revealed that the three cars with the highest MPG ratings were made by European manufacturers—not Japanese firms as I formerly expected.

Unfortunately, my hypotheses pertaining to the Grocery Store Survey proved much more difficult to evaluate, and I could derive no clear conclusions based on output from InfoZoom. A visualization generated to analyze the first, which called for a comparison of female and male spending habits, depicted two curves of nearly indistinguishable shape. The shape of the curves indicated that spending habits of males and females were strongly similar, but provided no clear indication of who may have spend more or less on average. An attempt to produce a visualization addressing the second hypothesis, which examined the total amount of purchased by females in two different income ranges, resulted in two adjacent, different sized scatter plots of extremely high variability that failed to explain which income group averaged a higher rate of spending. Overall, three main attributes of the "Grocery Store Survey" data seemed to complicate my analysis of it using InfoZoom: the larger size of the grocery store data (almost 12 times as large as the data pertaining to cars), the absence of well-defined trends to facilitate analysis, and my search a fine-grained piece of statistical information (i.e. an average value) in the midst of highly variable data. As a result I concluded that InfoZoom works better for analyzing large trends in both large and small data sets. It also seemed particularly useful for locating specific pieces of information embedded within smaller sized data sets, due to its "Overview" mode (in which it does a good job of indexing), as well as its ability to highly magnify and show large amounts of detail regarding individual data cases.

A few quirks in InfoZoom’s user interface somewhat complicated my ability to load, analyze, and understand the data sets I used it to explore. The program failed to register a number of keyboard commands and/or shortcuts that are commonly supported by GUI-based mouse-driven software. I quickly learned, for example, that I could not move through individual data cases by tapping the arrow keys upon selecting an attribute row using my mouse. This action instead relocated my presence in the interface to the text entry box located directly above the main display. At other times, properties of the InfoZoom UI that resemble those of a traditional Web browser contributed additional confusion, as well as a great deal of wasted screen space. Figure 1 shows a large row of buttons near the top portion of the application window that are completely unnecessary due to the availability of many of the functions they support in alternate areas of the UI, such as pull down menus. Almost half of the available screen space in figure 1 is taken up by the presence of such large unseemly buttons, all of which appear by default the first time the program is loaded. The Web browser paradigm also results in an unfortunate
(and frequently panic-inducing) coordination of an "Undo" command with a "Back" button. Users may use the "Back" button to undo a previous display change, but in doing so they must consciously think about how to execute an action that usually implicit—simply hitting Ctrl-Z or selecting the "Undo" command often results in either an unwanted or insignificant display change. Finally, after encountering a number of fatal errors and program crashes, I felt that overall InfoZoom seemed a bit unstable. I distinctly recall one such failure occurring when I attempted to select multiple data sets in a single attribute row by using my mouse in combination with the "Shift" key.

4.2 Eureka

Opening any reasonably sized data set in Eureka immediately exposes an almost stupefying display of graphics. The program’s quick and fluid response to virtually all user mouse action only serves to amplify this effect. At the same time, the dynamically updating fields located in the lower portion of the program window (known as the status bar) may also help users to obtain their bearings—especially when examining a data set which includes a large number of variables (displayed in the program as columns). My ability to establish relationships amongst all cases in a data set by simply moving a mouse over and around various columns adds an almost tactile dimension to the graphics in Eureka. I found this technique of "feeling out" where items are located particularly useful when examining the "Grocery Store Survey."

I began attempts to analyze more concretely hypotheses related to the grocery store data using a procedure similar to the one I employed while working with InfoZoom. Hiding unimportant information proved somewhat effective in evaluating the first hypothesis; reducing the number of columns increased my ability to view details of unfocused items using the mouse, but it made Eureka’s graphical display appear awkward and unbalanced compared to the initial eight-column view. Reducing the column number did lead me to notice a particular field in the lower display that provided the key to my problem obtaining average values, however. The middle-indentied field in the bottom right portion
of the status bar makes visible a "Median" value whenever the mouse is placed over a column of numerical information. By changing the contents of this field from "Median" to "Average" via the Options → Preferences… menu, I could analyze both grocery store hypotheses directly. All I needed was a way to sort columns and view their subsequent relationships to the "PurchaseAmount" column. I accomplished this using the "Filter" command to generate two separate views of data for each hypothesis. I then compared average values in both freshly filtered "PurchaseAmount" columns to either confirm or deny my hypotheses.

Eureka helped confirm my first and deny my second hypothesis about the grocery store data. My first hypothesis relating to male and female spending habits turned out to be correct: Women on average spent $127.32, whereas men spent an average of only $123.79. Figures denying my second hypothesis were similarly close; however, it turned out that women earning 20-30 thousand dollars annually actually spent more money on average ($123.72) than did women who earned between 80 and 90 thousand dollars per year. The close proximity of statistical figures associated with both hypotheses made visual representations of their corresponding data extremely difficult to distinguish from one another (see fig. 2). This led me to conclude that Eureka, with its delivery of precise statistical information in combination with elaborate graphics for the display of data trends, ideally supports differentiating between highly variable data sets from which no visual patterns may be derived.

Weaknesses in Eureka seemed to stem from the overwhelming density of its graphical display, as well some potentially limiting properties of its understated user interface. I encountered display problems when attempting to select a single item from a reduced, filtered version of the "Cars" data set. A line highlighting the position of my mouse in the data seemed to fall between two of the topmost data cases, and I had to "focus" (in other words enlarge) both of them in order to obtain details associated with the exact case I was looking for. The "Filter" feature proved invaluable for sorting and organizing data, but the tendency for each processed data set to appear in its own unique window could at times cause buildup leading to window management problems.

4.3 SeeIt

I freely admit that my first few uses of SeeIt were colored by a great deal of skepticism. I doubted the program’s ability to visualize in three dimensions what InfoZoom and Eureka imperfectly displayed in two. Nevertheless, the possibility of obtaining clear and stunning 3-D graphics from SeeIt intrigued me as I attempted to load the "Cars" data set and analyze my second hypothesis, which required comparing MPG rating and horsepower for all cars produced between 1977 and 1979. Then I met the Data Import Wizard. At first this obtrusive device seemed overly simplistic and almost opposed to visualization goals of encouraging a deeper understanding of data through interactivity and clever organization of graphic elements. Upon closer and retrospective examination, however, I realized that the Wizard actually plays a key role in the overall ability for users to comprehend visualizations they use employ SeeIt to construct. By forcing users to think a data set before seeing any picture of it whatsoever, SeeIt’s Data Import Wizard
sets the stage for more immediate comprehension of visual displays, and helps avoid bowling users over with elaborate pictures and immersive scenery. It also helps introduce users to essential program terminology like "bucket" (items that to me bared much closer resemblance to cells in a spreadsheet).

For my analysis of the second hypothesis, I used the Data Import Wizard to set the height of each bucket to equal the maximum vehicle MPG rating, and the width of each bucket to reflect the minimum value for vehicle horsepower. I also configured the X and Y-axes according to MPG and horsepower to facilitate the direct comparison of each attribute. After the Data Import Wizard finished, SeeIt returned some interpretation errors due to the presence of non-numerical values associated with one of the attributes; these errors closely resembled the ones I encountered upon loading the "Cars" data set into InfoZoom. SeeIt appeared to correct the problem on its own without direct intervention on my part. The program then presented its graphical interpretation of the data set—images that to me looked to me like green pylons of various heights and widths erupting out of a platform matrix. I quickly moved my mouse to the tallest thinnest pillar, and was tempted to jot down the "brushed" data that appeared as verification of my hypothesis. Something about the data didn't look right, though—the number of pylons on the screen seemed to exceed the number of cases I needed to examine. This caused me to remember a need to somehow limit the represented data to years between the range of 1977 and 1979. SeeIt’s "Filter" pull-down menu and well-designed slider widgets made doing this a snap. Applying the filter resulted in the clear, easy-to-read chart depicted in Fig. 3, whose tallest and thinnest pillar revealed my second hypothesis to be correct—the car with the highest MPG rating (a 1978 VW Rabbit Custom Diesel) did in fact have the possess the least amount of horsepower.

Further use of SeeIt allowed me to successfully evaluate other hypotheses in a quick and effective manner. The program seemed to make up for lag imposed by the Data Import Wizard by allowing me to gain an immediate understanding of visualization structures shortly after they appeared on screen. Unfortunately, I could attribute no such benefit to SeeIt’s poor implementation of 3-D navigation techniques. Using flat buttons within the application window to navigate inside a three-dimensional space struck me as inherently
counterintuitive, and keyboard shortcuts for panning, rotating, and zooming provided little relief from this activity. I did find the "brushing" feature extremely useful as a way to reveal primary attributes of individual data cases without having to search through the adjoining table. The "Edit Axis Connections" and "Edit Shape Connections" features also allowed me to use existing visualizations to quickly explore alternate hypotheses I found interesting. For example, in examining my third hypothesis (spending habits of males vs. females), simply switching the X-axis connection to reflect "Age" instead of "Gender" led me to discover that on average 80-90 year olds purchased more highly priced orders than any other age group.

Despite previously mentioned navigational shortcomings, overall I found SeeIt to be the most effective visualization program I examined due to the speed with which it allowed me to evaluate my hypotheses. However, the structure of the data import process definitely seems oriented to support users who have a good idea of what to look for in their data before viewing it in visual form. Users wishing only to explore their data might encounter difficulty assigning values to unseen visualizations in the Data Import Wizard, and therefore could find SeeIt less valuable.

4.4 Spotfire

In my view, a simple exploration of data seems about the only activity to which a user could apply Spotfire and achieve any degree of satisfaction. Repeated attempts to verify even the most straightforward hypothesis led to me to experience unmatched levels of frustration while using this program. Whenever I used it visualize distinctions between two variables, especially categorical ones, I encountered vague, confusing displays and no clear impressions about data I was evaluating.

![Fig. 4 – A 2-D scatterplot from Spotfire provides a vague affirmation of hypothesis #3](image-url)
My interrogation of hypothesis number three involving a comparison between male and female spending habits serves to illustrate many of the struggles alluded to above. Spotfire’s initial rendering of the "Grocery Store Survey" data set looked to me more like an abstract red and blue mosaic than a scatter plot. The legend told me that color was assigned to the variable of gender, meaning that data points associated with females appeared red, while those associated with males appeared in blue. After experimenting a bit, I decided to stick with the default setting to facilitate later analysis. I then sorted data points to depict a nice upward trend by assigning both X and Y-axes the variable "PurchaseAmount." This caused the data points to overlap and fall into a straight line, which made it nearly impossible to gather any large-scale understanding of how densely data along the trend was distributed without excessive use of the zoom function. Setting the "Jitter" value to "Max" resulted in a partial solution to this problem; formerly obscured data appeared like an explosion of confetti around the trend line. (More than likely, this use of jitter seriously compromised the accuracy of values on display. However, the fact that I was seeking to compare mean purchase amounts rather than exact values of individual data cases seemed to justify the loss in accuracy.) Finally, I managed to generate a makeshift male-female comparison by assigning the property of size to the "PurchaseAmount" variable.

The scatter plot in Fig. 4 is misleading because it does not clearly depict a relationship between male and female purchase amounts. Only by considering the density of red squares vs. blue squares in various portions of the chart, all while taking into account the overwhelmingly influential fact that females outnumber males throughout the represented data set, could a viewer hope to formulate a comparison of averages. Using Spotfire to collect this data into a 3-D scatter plot fails to clarify the relationship between gender and purchase amount; rather, the imbalance between males and females obscures it further. The fact that females made a drastically higher number of large-value purchases may lead a viewer to correctly conclude that the average purchase amount made by females was higher than the average purchase amount made by males. However, means by which a viewer arrived at such a conclusion would be extremely unsound. Naturally, my analysis of the third hypothesis in Spotfire left me with the overall impression that it does not do well in generating comparisons of average values!

I found my experiences in Spotfire unfortunate, because by programming, usability, and aesthetic standards it appears quite strong. Spotfire generates clean, colorful graphics inside a stable and highly flexible user interface. Users may pick in the Legend which widgets they feel most comfortable utilizing to probe data. Changes made to the widgets immediately appear on the main program display. Spotfire also supports highly fluid (and fun) mouse-driven interaction with 3-D scatter plots, as well as a well-animated and intuitively mapped Zoom feature.

5. Conclusion
In this report I examined four tools used for information visualization: InfoZoom, Eureka, SeeIt, and Spotfire. I investigated how each tool promoted insight into representations of data it depicted, specifically by using each one to confirm and/or deny hypotheses regarding preexisting data. The two sets of data I used in my evaluations pertained to cars and a survey of grocery store customers. Based on my evaluations, I found InfoZoom well suited to the analysis of large trends in both small and large data sets, as well as for locating specific pieces of information embedded within smaller sized data sets. I also found Eureka ideal for differentiating between highly variable data sets from which no visual patterns may be derived. I also determined that SeeIt ideally serves users seeking quick answers to specific data-related questions due to familiarizing properties associated with its data import process. Finally, I suggested that instead of visualizing average values and direct comparisons of two variables in context, Spotfire might better support tasks such as precise graphing of exact values and exploratory data analysis.