

# The History of ViSta: The Visual Statistics System

Pedro Valero-Mora,<sup>1\*</sup> Rubén D. Ledesma<sup>2</sup> and Michael Friendly<sup>3</sup>

ViSta is a project that focuses on dynamic and interactive graphics for statistics and was initiated by the late Forrest W. Young at the beginning of the 1990s. For over approximately 20 years, Forrest and other collaborators, including the authors of this article, have used ViSta for experimenting with these kinds of graphics in different settings, applying them to different scenarios of data and statistical analysis, searching to develop the right combination of features most appropriate in each case. In this time, ViSta evolved quite considerably, going through what we reckon were three different stages, namely: the initial one setting forth the foundations of ViSta; the second period where versions 5 and 6 of ViSta were released; and the consolidation period when a book summarizing the lessons learnt in the project was published. This book was titled 'Visual Statistics: Seeing your data with interactive and dynamic graphics' and was completed in the last days of life of Forrest, who continued to work enthusiastically in the project even though his health was seriously deteriorating during that time. This article is a tribute to this work, but also describes the innovative features of ViSta, many of which are still relevant today. © 2012 Wiley Periodicals, Inc.

#### How to cite this article:

*WIREs Comput Stat* 2012, 4:295–306. doi: 10.1002/wics.1203

**Keywords:** statistical graphics; dynamic; interactive; statistical package; computational statistics; data visualization

## INTRODUCTION

ViSta, the Visual Statistics System, is a statistical package commenced by Forrest W. Young in the 1990s which he continued developing for 16 years until his death in 2006. The ViSta project focuses on dynamic interactive graphics developed almost entirely in Lisp-Stat.<sup>1</sup> ViSta statistical procedures range from very basic procedures, such as those usually taught in introductory courses on Statistics, up to advanced multivariate methods. But more importantly it also serves as a testbed for the development of novel methods for interacting with visualizations and statistical analysis. The long history of ViSta makes it a rich source of ideas for

those interested in developing innovative, dynamic, interactive graphics for statistics.

This article will describe the history of ViSta by distinguishing among three periods of time, namely, the origins of ViSta, the ViSta 5.6 and 6 period, and the Visual Statistics book period. The authors were involved in the developing of ViSta mainly during the second and third periods; hence, our description of the beginnings of ViSta is possibly incomplete. Still, we hope to provide a reasonable account of the progress of ViSta along these years that can appeal to others interested in computational statistics in general and interactive dynamic graphics in particular.

## THE ORIGINS OF ViSta

The origin of ViSta is clearly related to the language in which it was developed: Lisp-Stat.<sup>1,2</sup> By the time that Lisp-Stat was released, Forrest had already been working on computational statistics. In particular, he had worked extensively on the topic of Multidimensional

\*Correspondence to: valerop@uv.es

<sup>1</sup>Departamento de las Ciencias del Comportamiento, University of Valencia (UVEG), Valencia, Spain

<sup>2</sup>CONICET/Faculty of Psychology, Universidad Nacional de Mar del Plata, Buenos Aires, Argentina

<sup>3</sup>Department of Psychology, York University, Toronto, Canada

Scaling,<sup>3</sup> developing software for this purpose that became part of two large commercial statistical packages. However, by the end of the 1980s, he had progressively become more interested in pure data visualization problems, and got involved in a project with this aim.<sup>4-6</sup> Besides, the emergence of the Macintosh computer had a very strong impact on him; he was quite impressed with its advanced graphical capabilities that he suddenly regarded the type of software he had been developing as inadequate. Hence, for a few years, his programming efforts stalled while he searched for an environment that made it possible for him to develop software that took advantage of these graphical capabilities. It was not until the beginning of the 1990s, when the first implementations of Lisp-Stat were released that Forrest found the right tool for restarting a computational and visual statistics project. This project was ViSta and it would be the focus of his work for the following 16 years, literally until the end of his life.

Lisp-Stat is 'an extensible statistical computing environment based on the Common Lisp language' (Ref 2, p. 626). Among other goals, Luke Tierney developed Lisp-Stat as '(...) a framework for experimenting with dynamical graphical methods' (p. 629). This involved the implementation of a number of features which environments focused on purely statistical computing would consider as relatively secondary or unimportant: object-oriented programming, native support of graphical user interfaces (GUIs), interactive plots, and so forth. Also the choice of an interactive interpreted language like Lisp as the basis for the system provided the opportunity of quickly prototyping new ideas in order to evaluate their potential. Despite the fact that Lisp-Stat undoubtedly had decreased in its popularity along the years for a number of reasons that were discussed in a special issue of the *Journal of Statistical Software*,<sup>7</sup> its combination of a simple approach for building GUIs, powerful capabilities for statistical graphics in general and dynamic graphics in particular, a solid statistical toolbox, and the value of a rich and mature standard programming language such as Common Lisp had a lasting impact on Forrest and on other experts in statistics.<sup>8,9</sup> Consequently several higher level analysis systems based on Lisp-Stat were started, the two most important examples perhaps being the regression analysis system Arc<sup>10,11</sup> and ViSta itself.<sup>12</sup>

With regard to ViSta, its first version was released ca 1992<sup>13,14</sup> and included a number of innovative features, namely: workmaps, guidemaps, linked plots and spreadplots, tourplots, command tools, data type recognition, datasheet, and implementations of a

variety of analytic statistical methods. These features will be discussed in the following subsections.

## Workmaps

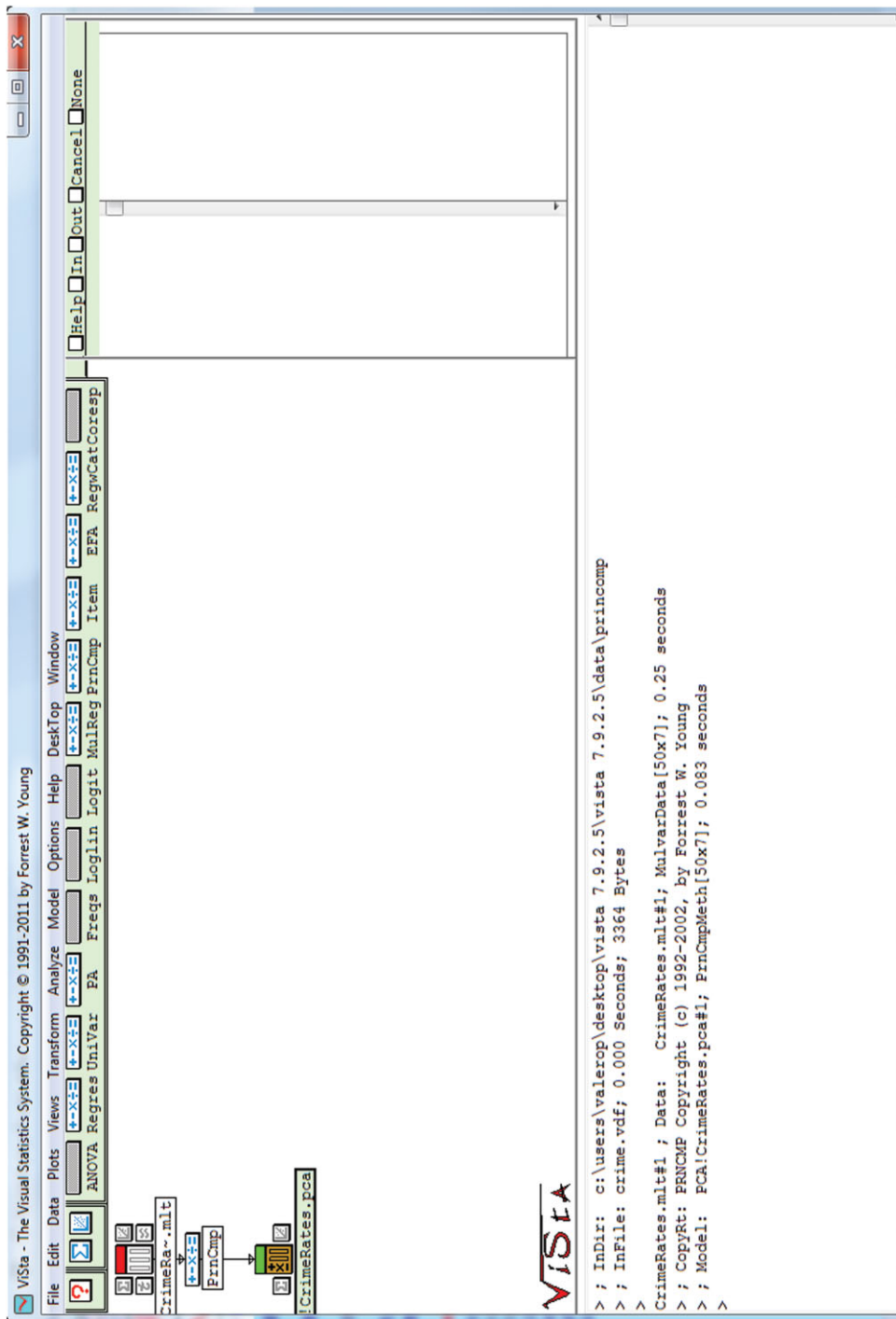
One of the first innovations in ViSta relative to other statistical packages consisted in representing data and the *process* of analysis with icons in its main window, as shown in Figure 1. In this case, a dataset of crime rates for the different US states has been opened by the user and a principal components analysis (PCA) has been applied to it. One result of the analysis is an icon that stands for the PCA model. The icons for data and analysis are linked via small arrows that help to follow the different steps taken by the user. This way, workmaps become a graphical representation of the process of data analysis and the steps taken in a specific session are built automatically as the analysis proceeds.

The workmap may contain several data objects opened at the same time, allowing a seamless transition from one dataset to another. Moreover, each workmap icon contains small buttons indicating actions that can be performed directly on the data or analysis icons, such as visualizing the plots associated with a specific icon. Finally, many data manipulations such as selecting specific variables/observations and creating new subsets of data, editing and transforming the data can be performed using the workmap.

The workmap concept is quite unique among statistical software environments, which simply use window datasheets for showing the open file in most cases. DataDesk<sup>15</sup> is another program with a special approach, using icons for representing datasets and also for individual variables, giving an additional layer of flexibility to the user interaction with the program. ViSta's approach can be used to answer one of the most frequent criticisms for GUIs for statistical analysis, which is they do not provide a *history* of the commands performed for arriving at a result. At this moment, ViSta does not save the icons created in a specific session but we think that this is an idea that would probably be worth further exploration.

## Guidemaps

Forrest was involved in data analysis teaching and consulting with students and researchers in different areas of knowledge. These users of statistics are often proficient in their own branch of knowledge, but are lost in the details of statistical methods and techniques applicable to their data. Therefore, Forrest came up with the idea of creating guidemaps as a way to guide novice users throughout the steps of data analysis. This concept of a computer guiding the users has been



**FIGURE 1** | A workmap with icons for data and analysis. Each workmap icon contains small buttons that link plots, summaries and other data displays.

discussed previously in some sources<sup>16–18</sup> but has only a limited appearance in other software. The idea of guidemaps was to provide guidance to the user via a visual diagram that indicated the steps that should be chosen next. The user could then select among the options offered and then new steps would be suggested. A more complete description of guidemaps can be found in Ref. 14.

## Linked Plots and Spreadplots

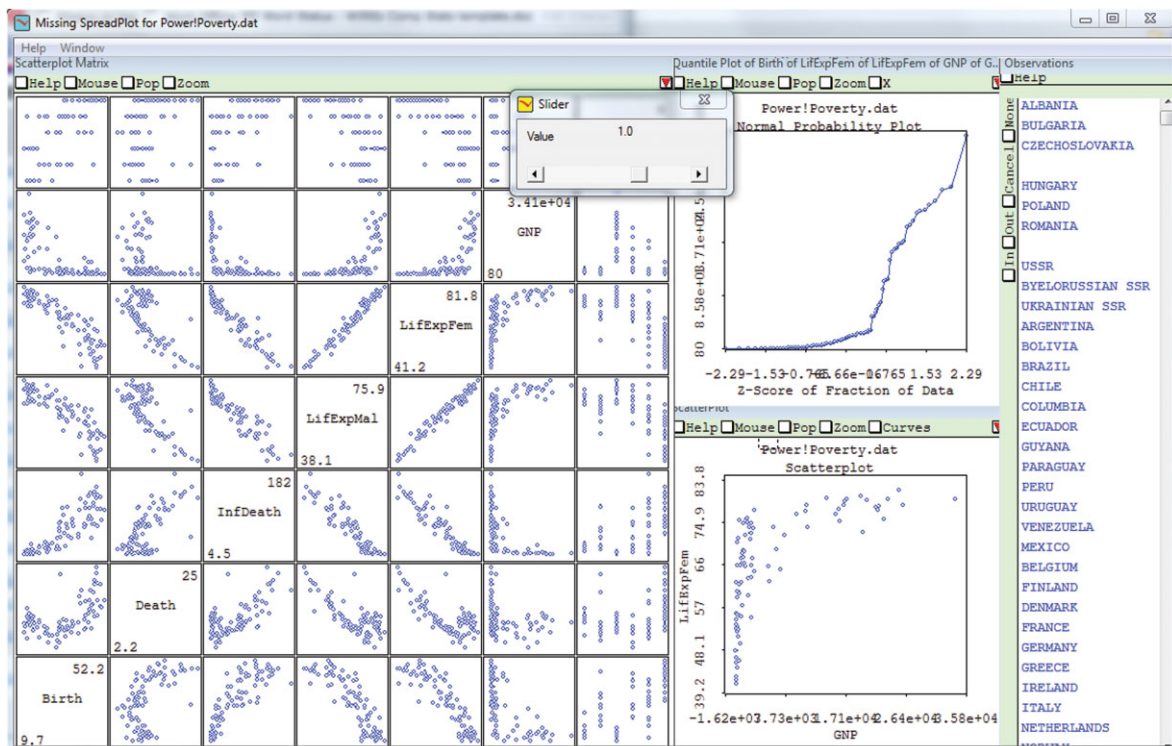
Linking is a dynamic interactive technique that coordinates different plots such that actions carried out on one of them will propagate to all the other plots.<sup>12</sup> Linking is a technique that was first described by Stuetzle<sup>19</sup> and is implemented in Lisp-Stat,<sup>1</sup> Datadesk,<sup>1</sup> ggobi,<sup>20</sup> and Manet<sup>21</sup> among others. The simplest version of linking is probably brushing several linked scatterplots.<sup>22</sup> In this scenario, the mouse is moved or dragged on the points of a scatterplot. The selected points are then activated, which is indicated by a visual change of color or shape, and the points in the linked scatterplot(s) are activated also. This allows the user to see the data directly in multiple views, but perhaps more importantly to manipulate the graphic display or analysis using what is seen.

The first version of ViSta extended the idea of simple linking to sets of plots called spreadplots.<sup>23</sup>

In spreadplots, an arbitrary collection of plots and data views appears in different windows but are all worked together as in a spreadsheet in several ways. For example, linking via selection of points or data variables was referred to as *empirically linked plots* but could encompass a wider range of interactive manipulation than available in other statistical graphics systems. A more advanced form of linking was called *algebraically linked plots*, meaning that the plots were linked via equations or computation, and changes in one plot would first apply equations to the data before yielding changes in the linked plots. This idea still has powerful implications and allows advanced explorations of high-dimensional data space.

The idea of working with groups of plots rather than individual plots resulted in the development of many spreadplots tuned to specific circumstances: data visualization, visual transformations, or visualization of statistical models. Consequently, a variety of spreadplots were developed by Forrest and by others<sup>24,25</sup> for numerical data, classification data, categorical data, analysis of variance (ANOVA), regression analysis, principal components analysis, multidimensional scaling, item analysis, and so forth.

Figure 2 displays a spreadplot for applying the Box–Cox transformation.<sup>26</sup> Examining the effect of this transformation is one of the earliest applications



**FIGURE 2** | Spreadplot for Box–Cox transformations. The slider applies a power transformation to a given variable, updating all plots.

of dynamic graphics according to Tierney (Ref 1, p. 297). In this case, the scatterplot matrix on the left displays several numerical variables portraying relationships that in some cases look nonlinear and/or heteroskedastic. Clicking on a plot cell of the scatterplot matrix changes the scatterplot and the normal probability plot to display the variables in the plot cell. Then, clicking on the slider changes the value of the parameter  $\lambda$  of the Box–Cox transformation<sup>26</sup> and applies it to the variable in focus (the one shown in the normal probability plot). As a result, the three plots change, displaying the transformed variables. The user can monitor the changes in the different plots in order to choose the transformation that better fits the goals and data.

## Tourplots

ViSta implemented dynamic 3D plots (spinplots) that could be viewed in various rotations via plot controls. But what about higher dimensional data? A tourplot is a rotating plot that spins in more than three dimensions (3D),<sup>4,6,27,28</sup> via some computation on the data to reduce it to a 3D space in a given view. As mentioned in Ref 14, p. 10, ‘A guided tourplot spins as directed by the user, the user creating a guided tour of the data. An unguided tourplot spins as it wishes, taking the viewer on a “grand” tour of the data. Just as a spinplot is designed to help the user visualize structure in three-dimensional data, a tourplot is designed to help the user visualize structure in high-dimensional data.’

ViSta included unguided and guided tourplots from the beginning of its history. The guided tourplots used principal components as the method for computing the spaces that the spinplot would use as targets. The implementation used real-time dynamic graphics that could be controlled by the user with high interaction, immediate feedback, and point-and-click actions. Hence, a button labeled ‘Go/Stop’ would allow the user to start the tourplot spinning toward the second target. Clicking and dragging on the plot would stop the automatic rotation and would allow the user to manipulate the spinplot manually. Then, automatic rotation could be initiated later continuing with the search for structure or interesting values.

## Command Tools

Point-and-click interfaces are not universally accepted as the best way of interacting with a data analysis system. In brief, expert users carrying out complicated steps often feel more comfortable writing commands than using a GUI. The advantages of typing commands are several: exact recording of a complicated

process, the possibility of repeating them with small modifications, transfer of scripts to other users, etc. ViSta has access to several command-based languages that work at different levels:

- **Command lines:** In Lisp, the listener is the window that evaluates the Lisp expressions. In ViSta, commands can be entered from the keyboard to the listener. All the ViSta menus and graphical operations have commands that can be typed using standard Lisp syntax. So, for example, writing (principal components) at the prompt cursor will perform a principal component analysis on all the variables in the current dataset and an icon representing the principal components will be added to the workmap. A sequence of commands can be used for recording complex sessions of data analysis that can be reproduced as a script when needed.
- **Lisp-Stat:** In ViSta, Lisp-Stat is hidden for the novice user but is still available to the advanced user to carry out computations and analysis that are not available in ViSta using the listener or menus. These analysis results can be piped to ViSta objects and then analyzed using the standard features in it. In this way, Lisp-Stat and ViSta could be used complementarily to each other, providing an environment able to accommodate advanced users. Later versions of ViSta made it easier to add analysis and commands to the system so it could be expanded with new capabilities by these advanced users.

## Data Theory and Automatic Selection of Statistical Techniques

Johan Sebastian Bach said on one occasion that ‘playing the keyboard was not so remarkable as... all one has to do is hit the right keys at the right time and the instrument plays itself’. Some might want to apply this statement to statistical analysis done with a computer: after all, if you knew what is the right key to press at any time the computer might carry out any statistical analysis for you. Of course, there would still remain the issue of making sense of the outputs you would obtain, but that is a different story.

Although the fantasy of getting software that carries out certain tasks auto-magically is out of reach in many cases, there are features that are customary in some software applications but often not in statistical environments: at the least, they should prevent one from doing something stupid. In particular, many software applications gray out or hide commands that are not applicable given the current state of the

system. Thus, for example, in a word processor, the command for pasting will not be available if there is nothing in the clipboard to be pasted. This way, the probability of hitting the wrong key is reduced and there are lesser options to be considered at any time.

ViSta was among the first statistical analysis systems to implement a feature designed to remove options that are not applicable to the current dataset. Other modern GUI systems for statistics<sup>29</sup> (e.g., JMP,<sup>30</sup> SAS/Insight, R Commander<sup>31</sup>) now provide some near equivalents. In ViSta, the variables selected in a dataset are evaluated after each user action and the active data type is determined. This determines the available options at any stage.

Table 1 shows a list of data types and its characteristics (Ref 12, p. 33). Routines for statistical analysis, visualization, transformation, and so on are implemented such that they know what data types they may accept. If the current data type is not appropriate, then the corresponding menu item is grayed out and is unavailable to the user.

For instance, a univariate dataset (only a numeric variable is activated) is not amenable to regression models, principal components, ANOVA, and so forth. Also, scatterplots, scatter matrix plots, or spin plots are graphics that cannot be used in this case. All these options then would be deselected in ViSta so the user will not be able to click on them, whereas histograms, numeric transformations, and univariate *t*-tests would be ready for being used as they are reasonable ways of working with data of this type.

## Datasheet

Certainly the most common interface to a data object is the datasheet. A datasheet is similar to a spreadsheet,

having cells organized in a rectangular array of rows and columns. Data can be entered into the cells, and standard editing tasks can be performed.

Building a datasheet often requires much more programming effort than it is needed for many statistical routines. In consequence, many freeware statistical computing environments do not include a datasheet and rely on external programs for data entry or editing instead. In the case of ViSta, a datasheet that permits examining the variables and carry out basic editing of the data is provided even though it is not as complete as spreadsheets available elsewhere.

## Statistical Methods

In ViSta, any statistical method produces a model icon that can be reported, visualized, or used to create new data derived from the analysis. The first version of ViSta featured a limited number of statistical modules that increased along the years. Table 2 lists the statistical modules in this first version of ViSta together with a short description of the characteristics of their visualization, report, and options to create new data variables. This table also lists new modules added in later versions.

## Interactive Graphical Modeling

Interactive graphical modeling is ‘a statistical visualization technique for visually exploring the nature of alternative parameterizations of statistical models’ (Ref 14, p. 18). This concept is based on using graphical tools for modifying the parameters of the model so that the new alternatives display the fit and the residuals automatically and the analyst may understand the consequences of these changes and find better alternatives. As well, the individual observations (or points) can also be modified, with the consequence that the fit and the residual are altered. This is again a technique for graphically exploring the contributions of individual observations to the model.

A first application of this idea was to multidimensional scaling (MDS). In MDS, coordinates for objects in the space are found as function of the distances among these objects. This can be applied for metric problems such as the distances among cities, but it becomes especially interesting when the distances are actually dissimilarities among objects as a function of subjective perceptions. The algorithm used for finding the solution is iterative and there is not any guarantee that the result conforms to a ‘global’ best solution but it can be merely a ‘local’ best solution. Computational algorithms used for this purpose sometimes apply an

**TABLE 1** | Organization and Metric of Various Datatypes

Datatype	Organization		Metric
	Type	Number	
Frequency			
Table	<i>n</i> -way table	<i>n</i>	Frequency and category
Category	Variable	<i>n</i>	Category
Magnitude			
Univariate	Variable	<i>n</i> = 1	Magnitude
Bivariate	Variable	<i>n</i> = 2	Magnitude
Multivariate	Variable	<i>n</i> > 2	Magnitude
Grouped	Variable	<i>n</i> + <i>c</i>	Grouped and magnitude
Missing	Variable	<i>n</i>	Magnitude
Association			
Association	Matrix	<i>n</i>	Associations

**TABLE 2** | Modules in ViSta

Statistical Module	Spreadplot Visualization	Statistical Report	Create-Data Option	Version of ViSta
Univariate analysis	Histograms, normal probability plot, and box and diamond plot	One sample $z$ or $t$ -tests, two independent samples $t$ -test, dependent samples $t$ -test, nonparametric tests	Empty	First version
Simple regression analysis	Scatterplot, residuals, influence, leverage, demonstration of influence, demonstration for restricted range	Parameter estimates, fit, ANOVA, collinearity measures, autocorrelation	Fit values, raw residuals, studentized residuals, externally studentized residuals, Cook distances	First version
ANOVA	Box and diamond plots, frequency distribution, ANOVA (partial) regression plot, residuals plot, profile plot	ANOVA, parameter estimates and model fit	Fit values, raw residuals, studentized residuals, externally studentized residuals, Cook distances	First version
Metric averaged multidimensional scaling	Stimulus plane and spin-plot, Scree plot, Stress plot	Eigenvalues, stimulus coordinates, stress	Coordinate, averaged data	First version
Multivariate regression/redundancy analysis	Scatterplot matrix for predicted variables, predictor biplot, Cook's distances, response fit plot, response residuals plot	Correlation matrix, Multivariate Test, parameter estimates and model fit Univariate regression analysis	Predicted values, regression coefficients	First version
Principal components analysis	Scree plot, scatterplot matrix, spinning biplot, biplot, box and diamond plots	Correlation matrix. Fit measures (eigenvalue, and explained variance) Eigenvectors and component scores Coordinates, absolute and relative contribution of variables and observations	Component scores, component coefficients, coordinates, relative contributions, absolute contributions	First version
Correspondence analysis	Spinplot and biplot (several versions), scree plot, residual and fit plot	Inertia, singular values Contingency table, row and column profiles, $\chi^2$ statistics, the row and column coordinates and contributions	Row and column coordinates, absolute and relative contributions	First version
Parallel analysis	Parallel scree plot, parallel boxplot for simulation	Observed eigenvalues mean and 95% CI	Eigenvalues of the simulation	ViSta 6
Frequency analysis	Mosaic plots, stacked bar graphs plots, interactive tables of frequencies	Percentages, expected values, $\chi^2$ , $\Phi$ coefficient, contingency coefficient	Observed data, expected frequency and residuals	ViSta 6
Loglinear analysis	Deviance and $\chi^2$ (can be recomputed for different models), mosaic plot for predicted data, mosaic plot for observed, plot of parameters in the model, history of fitted models	Deviance, list of parameters in the model	Design matrix, predicted, $\chi$ residuals, adjusted $\chi$ residuals, leverages, Cook distances	ViSta 6

TABLE 2 | Continued

Statistical Module	Spreadplot Visualization	Statistical Report	Create-Data Option	Version of ViSta
Logit analysis	Same as in loglinear	Same as in loglinear	Same as in loglinear	ViSta 6
Item-test analysis	There are visualizations for the following situations: Item descriptive analysis, test score descriptive analysis, Item-test analysis, Cronbrach's $\alpha$ model, split-half model, $\theta$ model, binary item analysis, exploratory factor analysis	There are reports for the following situations: Item descriptive analysis, test score descriptive analysis, Item-test analysis, Cronbrach's $\alpha$ model, split-half model, $\theta$ model, binary item analysis, exploratory factor analysis	Total scores, normalized scores, mean scores, deviation scores, estimated score, confidence interval for observed and estimated score, median and quartiles scores	ViSta 6
Factor analysis	Biplot, scree plot, scatterplot of communalities, factor loadings plot	Correlation matrix (Pearson or Tetrachoric), determinant, and Bartlett's test. Initial and extracted eigenvalues, variables loading and communalities. Reproduced and residual correlation matrix	Factor scores	ViSta 6
Homogeneity analysis	Joint display of categories and objects, scree plot, discrimination measures plot	Object scores, category quantifications, discrimination measures and average discrimination measures	Object scores, category quantifications	ViSta 6
Missing data imputation	Parallel boxplot for correlations, parallel boxplot, enhanced scatterplot, imputed data leverages, patterns plot, pattern $n$ -size versus contribution to Little's test scatterplot	Descriptives for missing values, correlations (pairwise, listwise, EM), Little's MCAR means test	Imputed data, multiple imputed data, correlations, listwise data, missingness matrix	ViSta 6
Bootstrap CI for univariate statistics	Empirical bootstrap distribution plots (scatterplot, histogram, Np-plot)	Bootstrap confidence interval (percentile and normal method)	Empirical bootstrap distribution	ViSta 6
Regression with categorical variables	Predicted/residual plot, influence/leverage plot, history of R-square, parameter plot	Parameter estimates, model fit, ANOVA		ViSta 7
Cluster analysis	Parallel boxplot, scatterplot matrix, dendrogram	Distance matrix	Groups according to color, sorted data	ViSta 7

ANOVA, analysis of variance; EM, estimation-maximization; MCAR, missing completely at random.

approach that involves a sort of numerical 'shake' to the points: coordinates found in a first step are altered and the algorithm is applied again to see if the solution either converges to the previous solution or to a new solution with a significantly different measure of fit. In the second case, this would be taken as evidence that the first solution was only local and that the algorithm was trapped in a nonoptimal solution.

In ViSta, the numerical shake was augmented by the possibility of altering the positions of the objects by direct manipulation via the mouse. Afterward, the user could request that the algorithm started to iterate and then observe how the points traveled in the map until their new location. Most of the time, the points would relocate to where they were but sometimes they might wind up in new locations, indicating a previous local minimum.



## ViSta 5.5 AND ViSta 6.4

ViSta 5.5 was released in 1998 and incorporated a number of changes and improvements. Also, in this version the collaboration began with one of the authors of this article (Pedro Valero-Mora). For this reason, Forrest implemented a new architecture for ViSta that made it easier than before to write external modules that could be added to the ViSta core. Using this facility, a module for missing data imputation and visualization was added to ViSta<sup>12</sup> followed by dynamic power transformations and log-linear models.<sup>25</sup> However, the process of fitting these new modules into the general framework of ViSta was not as smooth as desired and it required considerable fine tuning to make them work properly. Consequently, Forrest improved the architecture by developing a plug-in system that allowed arbitrary new extensions to be included, which is currently in use.

Other additions to ViSta in ViSta 5.5/6 were new modules for frequency analysis that included mosaic plots.<sup>32</sup>

- New spreadplots for univariate, bivariate, frequency, classification, and categorical data, and for univariate and bivariate regression.
- New graphics such as frequency polygons, hollow histograms, side-by-side and stacked bar graphs, parallel-coordinate plots, and mosaic plots, and improvements in many graphs. Also, kernel density estimators for histograms and smoothed lines for scatterplots were added.
- System changes such as color added to many parts of the system, layout of windows, help system based on html, changes in icons, importing/exporting of data, etc.

ViSta 6.4 brought about an important change in spreadplots. Until then, spreadplots were basically plot windows that were interconnected and that opened and closed simultaneously. However, each window kept the individual elements of the user interface and did not have a common visual aspect. In ViSta 6.4, a new type of window, a container, was added to the system, so the individual plots were now inserted in this window, giving spreadplots a much more consistent view. In addition, the linking between the plots beyond the basic empirical linking was managed directly by the system, by sending messages among the plots in response to user interactions. These two issues were discussed in Ref 23 and a new architecture (Gossip) for the message passing was

implemented, which facilitated the programming of spreadplots considerably.

Finally, the second author of this article (Rubén Ledesma) also got involved in the project in this version of ViSta. As a first step, he adapted some routines for homogeneity analysis,<sup>24,33,34</sup> and then followed with developments psychometric item analysis,<sup>35</sup> parallel analysis,<sup>36</sup> and other topics.<sup>37</sup>

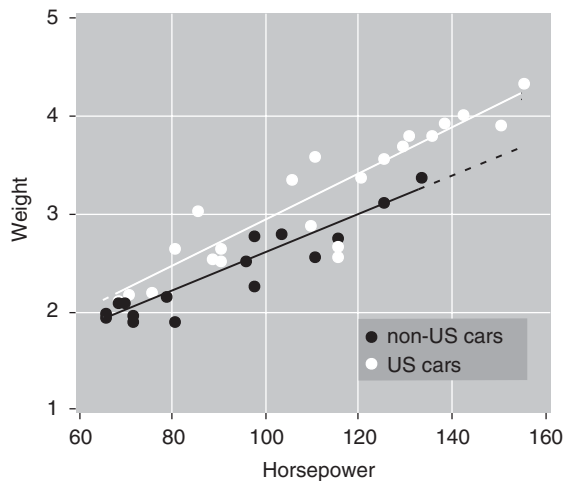
## THE VISUAL STATISTICS BOOK

Near the end of the 1990s, Forrest started to realize that it was time for wrapping up the ViSta project. After more than 10 years exploring dynamic graphics, innovative ways of interacting with statistical software, new plots or adaptations or older ones, and so forth, it was time to package all the knowledge gathered until then so that others could benefit in the future from the lessons learned by developing ViSta. This desire started to crystallize as a monograph describing the modules in ViSta that Valero and Forrest had already drafted by 2002. At that point, the possibility of publishing a book in the series on Probability and Statistics was explored. One of the authors (Michael Friendly) was enrolled in the project, and work on the book ‘Visual Statistics: Seeing Data with Dynamic Interactive Graphics’ began which was finished by early 2006.<sup>12</sup>

Unfortunately, Forrest’s health had deteriorated considerably during the last year of the writing. However, his passion for work filled him with enthusiasm and joy even in the face of chronic illness and he went on writing and programming even during his last months of life. During this time, Forrest continued adding improvements and new features to ViSta. Sadly, about the time that the final draft was being proofread, he became critically ill and passed away on April 9, 2006.

Besides, at the same time that we were working on the book, we did a thorough check of the functioning of its features and corrected a number of bugs and fixed many issues. The results were made available in ViSta 7, which was released simultaneously with the book. This version of ViSta brought the following new features:

- Improved start-up time: Previous versions of ViSta loaded several source files each time it started up. This version saved a binary image of the files with a notable increase in its start-up speed.
- Plots menu: Quite curiously, despite its heavy emphasis on statistical graphics, ViSta did not



**FIGURE 3** | Scatterplot of horsepower and weight of a set of cars.

have the possibility of creating plots individually. This was a consequence of the extensive use of spreadplots in ViSta, which were not always the best method for all the scenarios. Now the individual plots are also linked so the user can choose his own set of plots for visualizing his data and is not forced to use the plots included in a spreadplot.

- Improvements in dot-based plots: ViSta 7 added color and symbol palettes that can be used for changing the color and the symbol of the dots in scatterplots and similar plots.
- Interactive lines in scatterplots: In ViSta 7, different types of lines (regression, lowess, kernel density) can be drawn on the scatterplots. The lines can fit the selected points, or subsets of the points (Figure 3). Changing the selection or the color of the points changes the lines subsequently to reflect the new selection or coloring.
- Vectorized version of the plots: During the process of writing the book, we realized that screenshots were not able to produce publication quality graphics. Therefore, we added the capability of exporting/printing vector-drawn .pdf versions of the plots that could be easily enhanced using drawing programs. As an example, Figure 3 displays a scatterplot of the relation between 'horsepower' and 'weight' in a set of cars. The legend has been added manually using a program for vector drawing. The lines are least-squares regression lines fitting the dots of the same color (it can be seen that US cars usually weigh more than non-US cars for similar levels of horsepower).

- New analysis modules: This version has a number of new modules: Factor analysis, Cluster analysis (adapted from software developed by Huh and Lee), Logit analysis, and regression with categorical variables.

ViSta 7 was never distributed by Forrest before his demise but we had pre-release versions that worked partially. Once the book was published, one of the authors (Pedro Valero) created a version that ran satisfactorily. This version is distributed through a web site created especially for the Visual Statistics book (<http://www.uv.es/visualstats/Book/>). New versions and changes in the system can be found in this link, as well as documentation, a discussion list, and demonstrations.

## CONCLUSION

Dynamic graphical methods are research topics different from other topics in computational statistics because careful attention must be paid not only to numerical calculation but also to computational graphics and user interaction issues in order to yield satisfactory results. Thus, GUIs are not merely a matter of user convenience but they are a fundamental aspect of the research in this field, because the goal is to provide the analyst with tools for manipulating graphically the data to find answers to his questions. In this enterprise, good design is often a matter of trial and error, because we can only decide if a technique is effective by experiencing it in practice. Quick prototyping is therefore a must in this area because putting too much effort into programming a test makes it much more difficult to discard the code if the result is not satisfactory.

Luke Tierney originally developed Lisp-set as a 'framework for experimenting with dynamical graphical methods' (Ref 2, p. 629) and ViSta is one of the examples of the possibilities that this framework provided for those adopting it. So, although not perfect, Lisp-Stat allowed us to explore problems in a way that we think it was not possible to do in any other framework.

We currently see a renewed interest in interactive and dynamic graphics. Apart from the Visual Statistics book, two new books<sup>20,38</sup> have been released that are focused on this topic. Also, R<sup>39</sup> has been extended to include interactive dynamic graphics like, for example, rggobi<sup>20,40</sup> and iPlots.<sup>40</sup> As well, there are commercial programs—Spotfire, Tableau, etc.—that are very popular among certain users and Gapminder ([www.gapminder.org](http://www.gapminder.org)), now part of the Google Chart suite, offers the possibility of playing

with exciting interactive multimedia presentations based on data about the world. Among all these alternatives, ViSta stands as a software that, despite its age, implements solutions different from those offered in other systems—solutions shaped after almost 20 years of experimentation. Thus, although the process that gives birth to new things in computing is fast and unrelenting, and the substitution of the old for something newer is unavoidable, doing so without building on the experience previously gained would be simply foolish. Indeed, when we check current software we

sometimes find problems that Forrest and we had already struggled with a few years ago and that we solved satisfactorily, or so we would like to think. Developers of new software could probably learn much just by reviewing old software that has been around for a few years, and Forrest's ViSta project is one of the systems that deserves to be reviewed for the lessons it still provides. If ViSta turns out to be a source of inspiration for future advances in statistical software, it will have fulfilled its most important goal.

## REFERENCES

1. Tierney L. *LISP-STAT: An Object Oriented Environment for Statistical Computing and Dynamic Graphics*. New York: Wiley-Interscience; 1990, p 416.
2. Tierney L. Lisp-stat. *WIREs Comp Stat* 2010, 2:626–630.
3. Young FW, Hamer RM. *Multidimensional Scaling: History, Theory and Applications*. New York: Erlbaum; 1987, p 336.
4. Young FW, Kent DP, Kuhfeld WF. Dynamic graphics for exploring multivariate data. In: Cleveland WS, McGill M, eds. *Dynamic Graphics for Statistics*. Pacific Grove, CA: Wadsworth & Brooks/Cole; 1988, 391–424.
5. Young FW. Visualizing six-dimensional structure with dynamic statistical graphics. *Chance* 1989, 2: 22–30.
6. Young FW, Rheingans P. Visualizing structure in high-dimensional multivariate data. *IBM J Res Dev* 1991, 35:97–107.
7. Valero-Mora PM, Udina F. The health of lisp-stat. *J Stat Softw* 2005, 13:1–5.
8. de Leeuw J. On abandoning XLISP-STAT. *J Stat Softw* 2005, 13:1–5.
9. Weisberg S. Lost opportunities: why we need a variety of statistical languages. *J Stat Softw* 2005, 13:1–12.
10. Cook RD, Weisberg S. *Applied Regression Including Computing and Graphics*. New York: John Wiley & Sons; 1999.
11. Cook RD, Weisberg S. *An Introduction to Regression Graphics*. New York: John Wiley & Sons; 1994.
12. Young FW, Valero-Mora PM, Friendly M. *Visual Statistics: Seeing Data with Dynamic Interactive Graphics*. 1st ed. New York: Wiley-Interscience; 2006, p 363.
13. Young FW. *ViSta: The Visual Statistics System*. Report No.: 94-1(b). Chapel Hill, NC: L.L. Thurstone Psychometric Laboratory; 1996.
14. Young F, Faldowski R, McFarlane M. Multivariate statistical visualization. In: Rao CR, ed. *Computational Statistics. Handbook of Statistics*. Vol 9. North Holland: Amsterdam; 1993, 998.
15. Velleman PF. *DataDesk Version 6.0—Statistics Guide*. Ithaca, NY: Data Description Inc; 1997.
16. Chambers JM. Some thoughts on expert software. In: Eddy WF, ed. *Computer Science and Statistics: Proceedings of the 13th Symposium International of Computation and Statistics*. New York: Springer-Verlag; 1981, 36–40.
17. Gale WA. *Artificial Intelligence and Statistics*. Reading, MA: Addison Wesley; 1983.
18. Lubinsky DJ, Pregibon D. Data analysis as search. *J Econom* 1988, 38:247–268.
19. Stuetzle W. Plot Windows. In: Cleveland WS, McGill ME, eds. *Dynamic Graphics for Statistics*. Belmont, CA: Wadsworth; 1987, 225–246.
20. Cook D, Swayne DF. *Interactive and Dynamic Graphics for Data Analysis*. New York: Springer; 2007.
21. Unwin A, Hawkins G, Hoffman H, Siegl B. Interactive graphics for data sets with missing values: MANET. *J Comp Graphical Stat* 1996, 4:113–122.
22. Becker RA, Cleveland WS. Brushing scatterplots. *Technometrics* 1987, 29:127–142.
23. Young F, Valero-Mora P, Faldowski R, Bann CM. Gossip: The architecture of spreadplots. *J Comp Graphical Stat* 2003, 12: 80–100.
24. Ledesma RD, Valero-Mora P, Young F. Análisis de homogeneidad en ViSta “The visual statistics system”. *Metodología de las Ciencias del Comportamiento*. 2002, 4:139–149.
25. Valero-Mora PM, Rodrigo MF, Young FW. Visualizing parameters from log-linear models. *Comp Stat* 2004, 19: 113–133.
26. Box GEP, Cox DR. An analysis of transformations. *J R Stat Soc B* 1964, 26:296–311.
27. Asimov D. The grand tour: a tool for viewing multidimensional data. *SIAM J Sci Stat Comp* 1985, 6:128–143.

28. Buja A, Asimov D. Grand tour methods: an outline. *Comput Sci Stat* 1986, 17:63–67.
29. Martinez WL. Graphical user interfaces. *WIREs Comp Stat* 2011, 3:119–133.
30. SAS Institute Inc. *Using JMP 9*. Cary, NC: SAS Institute Inc.; 2010, p 540.
31. Fox J. The R commander: a basic-statistics graphical user interface to R. *J Stat Softw* 2005, 14:1–42.
32. Friendly M. Mosaic displays for multi-way contingency tables. *J Am Stat Assoc* 1994, 89:190–200.
33. Bond J, Michailides G. Homogeneity analysis in xlistat. *J Stat Softw* 1996, 1:1–16.
34. Valero-Mora PM, Young FW, Friendly M. Visualizing categorical data in ViSta. *Comp Stat Data Anal* 2003, 43:495–508.
35. Ledesma R, Molina J. Classical item and test analysis with graphics: the ViSta-CITA program. *Behav Res Methods* 2009, 41:1161–1168.
36. Ledesma RD, Valero-Mora PM. Determining the number of factors to retain in exploratory factor analysis: an easy-to-use computer program for carrying out parallel analysis. *Pract Assess Res Eval* 2007, 12: 1–11.
37. Ledesma R, Macbeth G, Cortada de Kohan N. Tamaño del efecto: revisión teórica y aplicaciones con el sistema estadístico ViSta. *Rev Latinoam Psicol* 2008, 40:425–439.
38. Unwin A, Theus M, Hofmann H. *Graphics of Large Datasets: Visualizing a Million (Statistics and Computing)*. Secaucus, NJ: Springer-Verlag New York, Inc; 2006.
39. Ihaka R, Gentleman R. R: A language for data analysis and graphics. *J Comp Graphical Stat* 1996, 5:299–314.
40. Theus M, Urbanek S. iPlots: interactive graphics for R. In: Hornik K, Leisch F, Zeileis A, eds. *Proceedings of the 3rd International Workshop on Distributed Statistical Computing*. Vienna: Technische Universität Wien; 2003, 1–11.

## WEB RESOURCE

<http://www.uv.es/visualstats/Book/>