



Journal of Statistical Software

February 2005, Volume 13, Issue 10.

<http://www.jstatsoft.org/>

The Health of Lisp-Stat

Pedro Valero-Mora
Universitat de València

Frederic Udina
Universitat Pompeu Fabra

Abstract

Lisp-Stat seems to be quite inert nowadays. However, there is no widespread agreement about what actually killed it, or even whether it has really passed away or not. Here we present the papers included in this special issue of the Journal of Statistical Software about it. Some of the included papers are about what Lisp-Stat was able to do in the past, other give testimony of the current state of health of Lisp-Stat, and other analyze whether there is any chance of its eventual recovery.

We believe that the diagnosis performed here will appeal not only to people interested in Lisp-Stat but to all those involved in the development of statistical languages, as it provides hints as to what elements could make them succeed or fail.

Keywords: Lisp-Stat, statistical software, statistical computing.

1. Introduction

Lisp-Stat does not seem to be in good health lately. By way of example, activity in the stat-lisp-news mailing list showed a gradual decline in number of messages up to July 2003. Then, somebody sent a message asking if there was anybody still reading it. The question was followed by a number of replies asserting that there were people reading the list, and that some of them were still using, or considering using, Lisp-Stat. However, despite these demonstrations of interest, the impression remaining was that Lisp-Stat was indeed running short of stamina.

But what is the problem? Is it something truly fundamental, or is it just that the spectacular growth of R over the last decade has made it redundant or unnecessary? As it happened, some of the people who replied to the lonely message referred to above seemed to believe that Lisp-Stat's illness was not terminal, and that therefore it was not necessary to put the language aside for good. On the contrary, some claimed that there were still powerful reasons to carry on using it. They mentioned aspects of Lisp-Stat that R had not yet been able to provide, such as its extensible dynamic graphics system, its facilities for building user interfaces, and

the advantage of using a general language that can be used for a large variety of problems, not merely those of a statistical type. Apparently, even though it was admitted that something had gone wrong with **Lisp-Stat**, the general consensus seemed to be that there was enough blood in its veins to keep it going for years.

However, as we all know, discussions in e-mail lists are not exactly the best arena for certain types of debates, as opinions manifested in them are often a product of one-minute feelings that may well evaporate after some more thoughtful reflection. People trying to make up their mind about a subject may feel that these views rely more on the discussants' hearts than on their brains, and consequently reject them. Therefore, we conceived this special issue of the JSS as a way of promoting and publishing grounded arguments that those evaluating the prospects of **Lisp-Stat** might trust.

We requested two types of contributions for this issue. On the one hand, we wanted examples of projects still going on that used **Lisp-Stat** as its main statistical environment, as we felt that it was important to evaluate in which areas no good replacement for **Lisp-Stat** had yet been found. This would provide evidence of the current strengths of **Lisp-Stat**. On the other hand, we requested accounts by people who had used it extensively in the past and had given it up for some other language. We hoped that these contributions would help us to gain a better understanding of the more serious drawbacks of **Lisp-Stat**. These stories of failure would inform us of the actual weaknesses of **Lisp-Stat** and its possibilities of recovery (if any).

We will discuss the papers in the following section. A short conclusion will close this editorial.

2. The papers

We classified the papers that we received in three categories. Two of them correspond to the categories of submissions we requested: descriptions of ongoing projects, and accounts of researchers' experiences of switching to other languages. Finally, reading the submissions, we decided to add a new group of papers for those providing general arguments for keeping **Lisp-Stat** active in the future. We will discuss the papers below, following these categories.

2.1. From **Lisp-Stat** to other languages

This first group of papers describes the experience of their respective authors with **Lisp-Stat**: how they started with it, what they accomplished using it, and when and why they decided to move to other environments for statistical programming (and whether they liked it).

The paper by Balasubramanian Narasimhan is a personal description of his experiences in moving from **Lisp-Stat** to **Java** to **R**. He illustrates the first part of his paper with examples of projects he developed in **Lisp-Stat** that relied heavily upon the interactive features of the system. The results were satisfactory, but the rise of **Java** led him to move some of his projects to it. As he realized, **Java** was an appropriate environment for statistical programming, and the interactive capabilities of **Lisp-Stat** could be replicated in it, but for large projects it proved to be inadequate. Hence he started programming with **R**, which he qualifies as high quality software.

In his contribution, Jan de Leeuw explains how, despite his personal tastes and previous investment in **Lisp-Stat**, his teaching and organizational responsibilities forced him to abandon **Lisp-Stat** when it was clear that it was losing ground versus **R**. This was a hard decision, not

only because he and his department were responsible for a large amount of code and services to `Lisp-Stat`, but because he still found many reasons for preferring it to R. Just to mention one that we found specially appealing, `Lisp-Stat` had a number of programming utilities inherited from Lisp that R will have to reinvent sooner or later. Nevertheless, as he puts it, departments like his had no choice but to promote R once the scientific community had selected it.

2.2. Projects currently using `Lisp-Stat`

This second group of papers is doubly interesting. Firstly, the papers are noteworthy in their own right, as they describe new statistical techniques, applications or programs that could have been published elsewhere as regular papers. And secondly, they contribute to the discussion on `Lisp-Stat` by providing examples of what can actually be done using this environment and that would not be possible in other environments.

The paper by Arturo Medrano-Soto *et al.* presents a method for classifying biological genes using variables of a heterogeneous nature. The implementation of the method is object oriented and it is programmed entirely in `Lisp-Stat`. However, as the volume of data to analyze grew to very large sizes, the authors opted to move the system to a commercial Common Lisp interpreter that provided higher speed. This paper offers an interesting example of one of the stronger aspects of `Lisp-Stat`: it is a computer language with a variety of available resources that are not specifically related to statistics. These resources, such as fast commercial versions in this case or a variety of free code for carrying out all kinds of tasks, can be used to enhance the `XLISP-STAT` interpreter.

The paper by Philip Iversen and Mervyn Marasinghe describes a program for visualizing experimental designs. The user can interact with the diagrams so that adding, modifying or deleting factors affects the proposed analysis. The authors see applications of these interactive tools both for teaching and for consulting, as they facilitate the understanding of the models under consideration. Extensible interactive graphics is perhaps the most important and still distinctive feature of `Lisp-Stat` and we found this paper a fine example of such capabilities.

The paper by Frederic Udina also focuses on the application of interactive graphics to data analysis. However, unlike Iversen's paper, it considers a display that has been implemented previously as a static display in other packages. This represents a twofold challenge: First, to add interaction to the display, and second, to demonstrate that his implementation is better than previous ones. The author of this editorial who was not involved in the paper believes that it succeeds amply in this enterprise, as the software described includes an overwhelming array of both interactive and non-interactive options that should delight anyone interested in biplot displays.

Didier Josselin's contribution introduces a topic that seems to us to be specially exciting: Geographical Information Systems. While the idea of interactivity in this area is not rare (as maps seem to suggest zooming in, zooming out, selecting, etc., quite automatically), the programs involved are usually based on low level programming routines that cannot be expanded easily to accommodate new ideas. In contrast, the philosophy of exploratory programming supported by `Lisp-Stat` facilitates enormously the task of pursuing new ideas for exploring data. These new concepts can be programmed, in this case by the geographer, and used to explore the data in previously unsuspected ways. The concepts need not be only statistically based, but can also use data base elements, interactivity and graphics. We feel that the results attained in this paper are both conceptually and aesthetically very appealing.

The paper by J. Gabriel Molina *et al.* is somewhat different from the rest, as it uses animations to show the main features of **ViSta**, a general purpose program for computation and visualization of statistics written entirely in **Lisp-Stat**. **ViSta** is one of the longest projects in **Lisp-Stat** (the other being **Arc**) with a history of about 15 years of development. This long history has enabled it to grow to the point where it is a viable alternative to other statistical programs. Readers can get a quick overview of its many characteristics by watching the movies on their computer screen.

2.3. Using **Lisp-Stat** in the future

The final two papers provide general arguments to support **Lisp-Stat**. Their authors coincide in emphasizing that, as some of the most distinctive features of **Lisp-Stat** have not yet been replicated by **R**, it would be a bad decision to sacrifice **Lisp-Stat** and renounce the possibility of using them.

Sanford Weisberg's contribution compares statistical languages to handtools. Tools, he explains, are usually more appropriate for some tasks than for others. This means that the user who uses a specific tool is restricted to producing certain outputs rather than others. Therefore, the current situation, in which one highly successful environment is becoming the only option available for statistical computing, means that the range of possible tasks that can be developed and explored is limited to what this environment is able to do. As this environment happens not to incorporate the tools for dynamic graphics that are available in **Lisp-Stat**, possible developments in this direction are severely limited. This is very unfortunate, as the author has the conviction that dynamic graphics are useful in many important statistical scenarios. Therefore, the author makes a plea for a variety of statistical languages from which the developers can choose depending on the ideas to be explored.

Finally, the paper by Luke Tierney, the original creator and main developer of **Lisp-Stat**, falls in its own category. Thus, while the rest of the papers offer a view of the strengths and weaknesses from an external point of view, as users of the language, his paper provides an outline of these issues from an internal point of view. The paper mentions several aspects in which **Lisp-Stat** can be considered still a good choice, and also gives some pointers about other aspects that could be improved in the future. His paper also draws from his ample experience with other statistical languages, particularly **R**, providing comparisons that highlight the technical qualities of the different systems.

3. Conclusion

We believe that this special issue has fulfilled our goal of providing an in-depth view of **Lisp-Stat**'s state of health. The papers included in it give examples and counter-examples of reasons for using or not using **Lisp-Stat**. They also give indications for improving **Lisp-Stat** with features not available in it, and for improving other languages with features already available in **Lisp-Stat**. We trust that the discussion will be useful for those making up their minds about **Lisp-Stat** or statistical languages in general.

Acknowledgements

This work was supported by grant from the spanish Ministry of Science and Technology BSO2002-02220 (first author) and BMF-2003-03324 (second author).

Affiliation:

Dr. Pedro M. Valero Mora
Departamento de Metodología de las Ciencias del Comportamiento
Facultad de Psicología
Universitat de València
C/Blasco Ibáñez, 21
46010 Valencia, Spain
Telephone: +34/96364507
E-mail: valerop@uv.es

Frederic Udina
Departament d'Economia i Empresa
Universitat Pompeu Fabra
Ramon Trias Fargas 25
08005 Barcelona, Spain
E-mail: udina@upf.es
URL: <http://tukey.upf.es/>