Sweave: Dynamic Generation of Statistical Reports Using Literate Data Analysis

Friedrich Leisch

Report No. 69
March 2002

March 2002

SFB
‘Adaptive Information Systems and Modelling in Economics and Management Science’

Vienna University of Economics
and Business Administration
Augasse 2–6, 1090 Wien, Austria

in cooperation with
University of Vienna
Vienna University of Technology

http://www.wu-wien.ac.at/am

This paper has been accepted for publication in:
Proceedings of CompStat 2002, Springer Verlag, 2002.

This piece of research was supported by the Austrian Science Foundation (FWF) under grant SFB#010 (‘Adaptive Information Systems and Modelling in Economics and Management Science’).

Sweave: Dynamic Generation of Statistical Reports Using Literate Data Analysis

Friedrich Leisch

Institut für Statistik und Wahrscheinlichkeitstheorie, Technische Universität Wien, Wiedner Hauptstraße 8-10/1071, A-1040 Wien, Austria

Abstract. Sweave combines typesetting with LaTeX and data anlysis with S into integrated statistical documents. When run through R or Splus, all data analysis output (tables, graphs, ...) is created on the fly and inserted into a final LaTeX document. Options control which parts of the original S code are shown to or hidden from the reader, respectively. Many S users are also LaTeX users, hence no new software has to be learned. The report can be automatically updated if data or analysis change, which allows for truly reproducible research.

Keywords. R, S, literate statistical practice, integrated statistical documents, reproducible research

1 Introduction

The traditional way of writing a report as part of a statistical data analysis project uses two separate steps: First, the data are analyzed using one's favorite statistical software package, and afterwards the results of the analysis (numbers, graphs, ...) are used as the basis for a written report. In larger projects the two steps may be repeated alternately, but the basic procedure remains the same. Many statistical software packages try to support this process by generating pre-formatted tables and graphics that can easily be integrated into a final report using copy-and-paste from the data analysis system to the word processor. The basic paradigm is to write the report around the results of the analysis.

Another approach for integration of data analysis and document writing is to embed the analysis itself into the document, which reverses the traditional paradigm. Over the last decade a number of systems have been developed that integrate analysis and documentation and allow for *literate statistical practice*, see Rossini (2001) for a survey.

This new paradigm is probably most popular for creation of dynamic web pages and offers completely new possibilities for teaching statistics and delivering statistical methodology over the Internet. E.g., the ExploRe system (Härdle et al., 1999) provides means to embed statistical quantlets in web pages or electronic books to create interactive documents with direct access to a statistical data analysis package. Another example for a dynamic statistical analysis on a web page is given in Temple Lang (2001), by embedding R into netscape as a plugin. Report rendering is performed using XML and XSL.

We introduce a new system, called Sweave, which combines ideas from both worlds described above using literate programming tools. The purpose is to create dynamic reports, which can be updated automatically if data or analysis change, while using standard tools for both data analysis and word processing. Sweave is written in the S language, either the open source R (`http://www.R-project.org`) or the commercial Splus (`http://www.insightful.com`) can be used for statistical data analysis.

2 Sweave files

Sweave source files are regular noweb files (Ramsey, 1998) with some additional syntax that allows control over the final output. Noweb is a simple literate-programming tool which allows to combine program source code and the corresponding documentation into a single file. These consist of a sequence of code and documentation segments, called *chunks*. Different command line programs are used to extract the code (*"tangle"*) or typeset documentation together with the code (*"weave"*).

A minimal Sweave file is shown in Figure 1, which contains two code chunks embedded in a simple LaTeX document. '`<<...>>=`' at the beginning of a line marks the start of a code chunk, while a '`@`' at the

```
\documentclass[a4paper]{article}

\begin{document}

In this example we embed parts of the examples from the
\texttt{kruskal.test} help page into a \LaTeX{} document:

<<>>=
data(airquality)
kruskal.test(Ozone ˜ Month, data = airquality)
@
which shows that the location parameter of the Ozone
distribution varies significantly from month to month.
Finally we include a boxplot of the data:

\begin{center}
<<fig=TRUE,echo=FALSE>>=
boxplot(Ozone ˜ Month, data = airquality)
@
\end{center}

\end{document}
```

Figure 1: A minimal Sweave file: `example-1.Snw`.

beginning of a line marks the start of a documentation chunk. Sweave translates this into the LaTeX document shown in Figures 2 and 3. The first difference between `example-1.Snw` and `example-1.tex` is that the LaTeX style file `Sweave.sty` is automatically loaded, which provides environments for typesetting S input and output (the LaTeX environments `Sinput` and `Soutput`). Otherwise, the documentation chunks are copied without any modification from `example-1.Snw` to `example-1.tex`.

The real work of Sweave is done on the code chunks: The first code chunk has no name, hence the default behavior of Sweave is used, which transfers both the S commands and their respective output to the LaTeX file, embedded in `Sinput` and `Soutput` environments, respectively. This emulates an R terminal session.

The second code chunk shows one of the Sweave extension to the noweb syntax: Code chunk names (the text within the double angle brackets) can be used to pass options to Sweave which control the final output: The chunk is marked as a figure chunk (`fig=TRUE`) such that Sweave creates EPS and PDF files corresponding to the plot created by the commands in the chunk. Furthermore, a `\includegraphics{example-1-002}` statement is inserted into the LaTeX file ('`002`' is the default for figures from the second code chunk). Option `echo=FALSE` indicates that the S input should not be included in the final document (no `Sinput` environment).

3 Design principles and features

"One of the hardest tasks known is to convince a competent computer user to switch from a preferred computing environment to a new one" (Rossini, 2001). Hence, one of the major design principles of Sweave was to provide a system for literate statistical practice which S users will find familiar and convenient. This naturally leads to embedding the statistical analysis in LaTeX documents, because many S users already use LaTeX for typesetting.

The emacs text editor (Stallman, 1999) offers a perfect authoring environment for Sweave, especially for people who already use emacs for writing LaTeX documents and interacting with an S engine. We have chosen to use noweb as basis for the Sweave system because

1. the syntax is extremely simple and hence easy to learn

```
\documentclass[a4paper]{article}

\usepackage{Sweave}
\begin{document}

In this example we embed parts of the examples from the
\texttt{kruskal.test} help page into a \LaTeX{} document:

\begin{Sinput}
R> data(airquality)
R> kruskal.test(Ozone ˜ Month, data = airquality)
\end{Sinput}
\begin{Soutput}
      Kruskal-Wallis rank sum test

data: Ozone by Month
Kruskal-Wallis chi-squared = 29.2666, df = 4, p-value = 6.901e-06
\end{Soutput}
which shows that the location parameter of the Ozone
distribution varies significantly from month to month.
Finally we include a boxplot of the data:

\begin{center}
\includegraphics{example-1-002}
\end{center}

\end{document}
```

Figure 2: The output of `Sweave("example-1.Snw")` is the file `example-1.tex`.

2. the ESS noweb mode for emacs already provides a perfect authoring environment

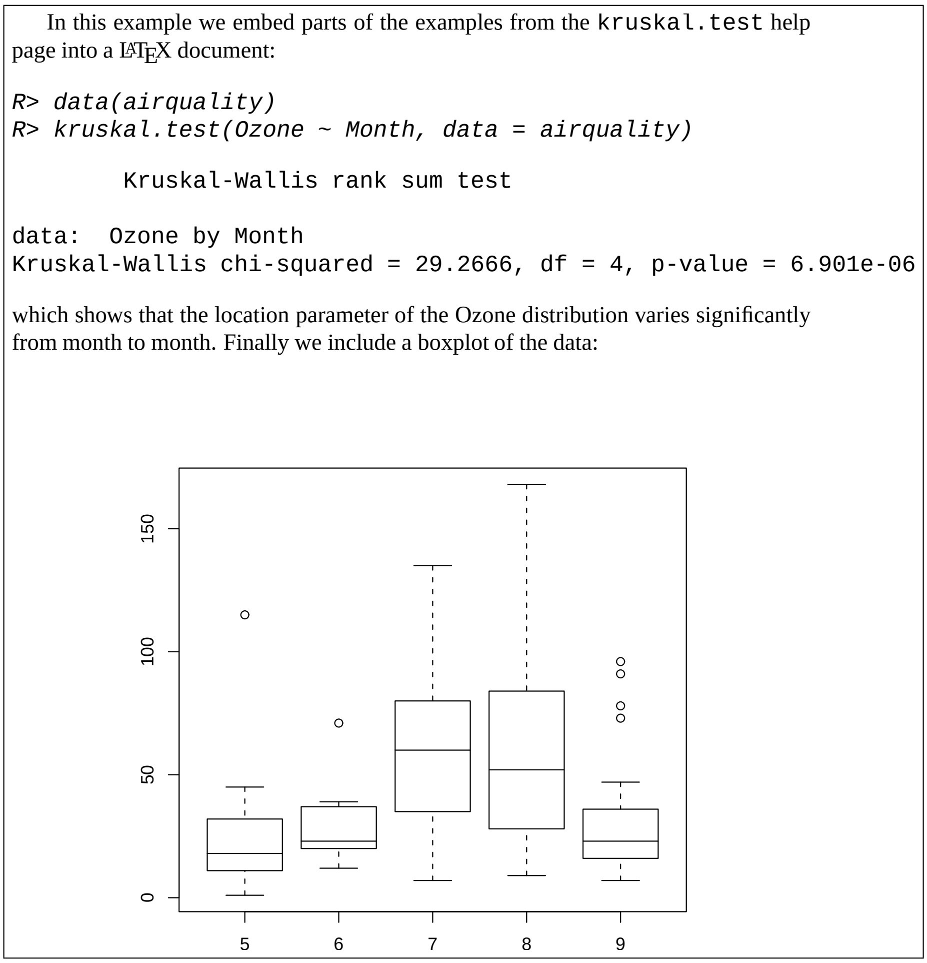
The importance of 2 should not be underestimated, a document format without convenient tools for authors will almost certainly be ignored by prospective users. However, it is not necessary to use emacs, Sweave is a standalone system, the noweb source files for Sweave can be written using any text editor.

ESS (*Emacs speaks statistics*, Rossini et al., 2001) allows to connect an Sweave file to a running R or Splus process while writing the document. Code chunks can be sent to the S engine and evaluated using simple keyboard shortcuts or popup menus. Syntax highlighting, automatic indentation and keyboard shortcuts depend on the location of the pointer: in documentation chunks emacs behaves as if editing a standard LaTeX file, when the pointer moves to a code chunk the mode switches automatically to S programming.

Sweave is not the first attempt to combine S and LaTeX using noweb: the RevWeb system (Lang and Wolf, 1997) uses the same principle. The major difference is that Sweave has been implemented in the S language itself, hence it works on all platforms where S is available (Linux, Mac, Unix, Windows). RevWeb uses system calls to the noweb command line utilities and therefore has not the same portability.

Another advantage of using S itself for all tangling and weaving steps is much finer control over the S output. Options that can be set either globally to modify the default behavior or separately for each code chunk control how the the output of the code chunks is inserted into the LaTeX file, e.g.:

- Hide or show S input code lines; include text output verbatim, as LaTeX code or hide it completely.
- Emulate an interactive R session: values of assignments are not printed, values of single objects are printed.
- Suppress evaluation of single code chunks.
- Text output can be written to separate files for each code chunk, EPS and PDF files of graphics can be created, and corresponding `\input{}` or `\includegraphics{}` statements inserted.



In this example we embed parts of the examples from the `kruskal.test` help page into a LaTeX document:

```
R> data(airquality)
R> kruskal.test(Ozone ~ Month, data = airquality)

        Kruskal-Wallis rank sum test

data:  Ozone by Month
Kruskal-Wallis chi-squared = 29.2666, df = 4, p-value = 6.901e-06
```

which shows that the location parameter of the Ozone distribution varies significantly from month to month. Finally we include a boxplot of the data:

Figure 3: The final document is created by running `latex` on `example-1.tex`.

In addition the user can specify hook functions which are automatically executed before code chunks of certain types, e.g., to set graphical parameters. A detailed description of all current features can be found in Leisch (2002).

4 Summary and outlook

Sweave is already used for a wide variety of applications: Reports for medical statistical consulting that can be updated automatically when new data arrive or data change; lecture notes for statistics classes with executable examples; and manuals with embedded examples for bioinformatics software, especially DNA microarray analysis. Sweave is part of every R installation (version 1.5.0 or higher) and can be used to automatically create and check manuals in R packages. All code can be executed to verify consistency of code and documentation and is part of the quality assurance tests of the software.

A disadvantage of TeX-based file formats is that they are harder to parse than other formats, hence computation on documentation objects is not as easy as, e.g., with XML-based formats. Hence for the next generation of the R help page format an XML-based solution may be preferable. An obvious disadvantage of XML is that these formats tend to be very verbose, i.e., the markup occupies a larger percentage of the file making it harder to read for humans (simply compare an HTML file to a LaTeX file, especially more complicated markup like formulas). Hence, the ideal authoring environment is not obvious. These issues are currently under investigation, see also `http://www.omegahat.org`.

Sweave files are easy to write and offer the full power of LaTeX for high-quality typesetting. This makes them ideal for documents where quality is essential and which are mainly read in printed form, e.g., manuals, lecture notes or consulting reports. Using PDF even hyperlinked online documents can be created easily from LaTeX. Storing code and documentation in a single source file makes research completely reproducible, all results can easily be verified and regenerated.

Acknowledgments

This research was supported by the Austrian Science Foundation (FWF) under grant SFB#010 ('Adaptive Information Systems and Modeling in Economics and Management Science'). The author thanks Kurt Hornik for comments and ideas.

References

W. Härdle, S. Klinke, and M. Müller. *XploRe Learning Guide*. Springer Verlag, 1999.

L. Lang and H. P. Wolf. The REVWEB manual for Splus in Windows. URL `http://wiwi.uni-bielefeld.de/StatCompSci/software`. Universität Bielefeld, Germany, 1997.

F. Leisch. Sweave user manual. URL `http://www.ci.tuwien.ac.at/~leisch/Sweave`. Institut für Statistik und Wahrscheinlichkeitstheorie, Technische Universität Wien, Vienna, Austria, 2002.

N. Ramsey. *Noweb man page*. University of Virginia, USA, 1998. URL `http://www.cs.virginia.edu/~nr/noweb`. version 2.9a.

A. Rossini. Literate statistical analysis. In K. Hornik and F. Leisch, editors, *Proceedings of the 2nd International Workshop on Distributed Statistical Computing, March 15-17, 2001, Technische Universität Wien, Vienna, Austria*, 2001. URL `http://www.ci.tuwien.ac.at/Conferences/DSC-2001/Proceedings/`. ISSN 1609-395X.

A. J. Rossini, M. Mächler, K. Hornik, R. M. Heiberger, and R. Sparapani. Emacs speaks statistics: A universal interface for statistical analysis. Report 164, Department of Biostatistics, University of Washington, 2001. URL `http://software.biostat.washington.edu/statsoft/ess/ess-techrep.pdf`.

R. M. Stallman. *The Emacs Editor*. Free Software Foundation, Boston, MA, USA, 1999. URL `http://www.gnu.org`. version 20.7.

D. Temple Lang. Embedding S in other languages and environments. In K. Hornik and F. Leisch, editors, *Proceedings of the 2nd International Workshop on Distributed Statistical Computing, March 15-17, 2001, Technische Universität Wien, Vienna, Austria*, 2001. URL `http://www.ci.tuwien.ac.at/Conferences/DSC-2001/Proceedings/`. ISSN 1609-395X.