An Sweave tutorial

Mixing R and LaTeX

Nicola Sartori

Università "Ca' Foscari" Venezia

http://www.dst.unive.it/~sartori

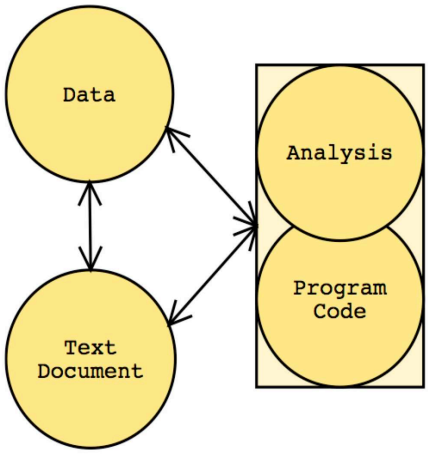
Writing statistical reports

When doing data analysis and writing reports, usually we tend to separate the two stages:

1. Data and analysis using some statistical software (files for the data, files for the code).
2. The results from 1. are used as a basis for a written report (file(s) for the report).

- After several modifications of one of the files involved things tend to get out of sync: which version *exactly* correspond to the final results in the report?!
- Just hope you don't need to go back to 1. after some months!
- More "philosophically", data and code could be seen as the "proof" for the results.

Writing statistical reports



Writing statistical reports: an alternative

- Embed the analysis into the report!
- End up with only the report (and data) file(s).
- The purpose is to create
  1. reproducible reports,
  2. dynamic reports.

After some months you need to do some changes in your analysis (the data and/or the code)?
Just do it in your report file and the report gets automatically updated!

What is Sweave?

- Sweave is a tool that allows to embed R code in (sort of) LaTeX documents.
- The document will contain both documentation parts (written in LaTeX) and code parts (written in R).
- The code is evaluated in R.
- The resulting console output, figures and tables are automatically inserted into the final document.
- This produces a `.tex` file on which it is possible to run LaTeX.

Padova - PhD Program slide 5

What is Sweave?

- A set of S (R) functions, written by Friedrich Leisch (http://www.statistik.lmu.de/~leisch), working under one command in `utils` package.
  - Processes R code within a LaTeX document
  - Returns output from such code (if so desired).
  - Creates plots and automatically creates the LaTeX code for their inclusion (if so desired).
- A LaTeX package and style (`Sweave.sty`).
- Homepage: http://www.statistik.lmu.de/~leisch/Sweave

Padova - PhD Program slide 6

How to install Sweave

- Assuming LaTeX and R are installed, there is no need for installation!
- Sweave is distributed with R (since version 1.5.0).
- In the latest versions of R it is included in the `utils` package (no need to load it).
- No need to learn new languages:
  - in the documentation part, do LaTeX,
  - in the code part, do R.

Padova - PhD Program slide 7

How does it work?

- Write the LaTeX file, but with extension `.Rnw` (or `.Snw`) instead of `.tex`: `myfile.Rnw`.
- The file will also contain code segments, *suitably separated* from the LaTeX segments.
- Within R, execute `Sweave("myfile.Rnw")`, assuming `myfile.Rnw` is in the working directory of R.
- This executes the code segments and will produce the file `myfile.tex`.
- Run LaTeX on `myfile.tex` and obtain your report.

Padova - PhD Program slide 8

The Noweb syntax

- To separate code and documentation chunks the Noweb syntax is used.
- Noweb is a simple literate programming tool which allows to combine program source code and the corresponding documentation into a single file.
- Different segments are called *chunks*:
  - `<< options >>=` denotes the start of code chunk,
  - `@` denotes the start of a documentation chunk.
- Two kind of operations:
  - *weave*: typeset documentation together with code,
  - *tangle*: extract code chunks.

Padova - PhD Program slide 9

Basic options for code chunks

- `label` is an optional name for the chunk. If it is the first option in the chunk then `label=` can be omitted.
- `echo` if `TRUE` it echoes the commands, if `FALSE` it does not. Default is `TRUE`.
- `fig` if `TRUE` it includes the plot created in the code. Default is `FALSE`
- More options in a moment...

Padova - PhD Program slide 10

A simple example: example-1.Rnw

```
\documentclass[a4paper]{article}

\title{Sweave Example 1}
\author{Friedrich Leisch}

\begin{document}

\maketitle

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

<<>>=
data(airquality)
library(ctest)
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}
```

First code chunk

Second code chunk

A simple example: chunks

- No options were set on the first code chunk.
  - Defaults to `echo=TRUE,fig=FALSE`.
  - Consequence: command and output are printed.
- On the second code chunk we set `echo=FALSE,fig=TRUE`.
  - Consequence 1: no echo of commands.
  - Consequence 2: Plot will be included. Both eps and pdf files will be created.
  - The name of the plot files will be `filename-chunk number` (`example-1-002.eps/.pdf`). If a chunk `label` was given, then it will substitute the chunk number in the file name.
- To produce the file `example-1.tex` we only need to run `Sweave("example-1.Rnw")` in R.
- WARNING: we only make changes in the `.Rnw` file.

A simple example: example-1.tex

```
\documentclass[a4paper]{article}

\title{Sweave Example 1}
\author{Friedrich Leisch}

\usepackage{/Library/Frameworks/R.framework/Resources/share/texmf/Sweave}
\begin{document}

\maketitle

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

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

data:  Ozone by Month
Kruskal-Wallis chi-squared = 29.267, df = 4, p-value =
6.901e-06
\end{Soutput}
\end{Schunk}
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}
```

First code chunk

Second code chunk

`(pdf)latex example-1.tex`

Sweave Example 1

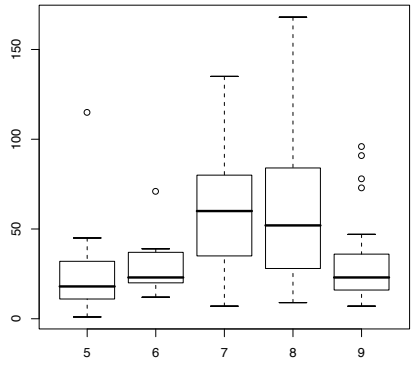
Friedrich Leisch

November 19, 2006

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

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

        Kruskal-Wallis rank sum test

data:  Ozone by Month
Kruskal-Wallis chi-squared = 29.267, 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:



1

Padova - PhD Program slide 14

Options slide 15

A few more options

- `eval`: logical (`TRUE`). If `FALSE`, the code chunk is not evaluated, and hence no text or graphical output produced.
- `results`: character string (`verbatim`). If `verbatim`, the output of S commands is included in the verbatim-like Soutput environment. If `tex`, the output is taken to be already proper latex markup and included as is. If `hide` then all output is completely suppressed (but the code executed during the weave).
- `prefix`: logical (`TRUE`). If `TRUE` generated filenames of figures and output have a common prefix.
- `prefix.string`: a character string, default is the name of the "`.Rnw`" source file.

Padova - PhD Program slide 15

A few more options

- `include`: logical (`TRUE`), indicating whether input statements for text output and includegraphics statements for figures should be auto-generated. Use `include = FALSE` if the output should appear in a different place than the code chunk (by placing the input line manually).
- `fig`: logical (`FALSE`), indicating whether the code chunk produces graphical output. Note that only one figure per code chunk can be processed this way.
- `eps`: logical (`TRUE`), indicating whether EPS figures shall be generated. Ignored if `fig = FALSE`.
- `pdf`: logical (`TRUE`), indicating whether PDF figures shall be generated. Ignored if `fig = FALSE`.
- `width`: numeric (6), width of figures in inch.
- `height`: numeric (6), height of figures in inch.

Some comments on `options`

- Options can be set globally at the beginning of the file (and changed everywhere else) with `\SweaveOpts{option1=value1,option2=value2,...}`.
- `width` and `height` are fed to R.
  - These determine the size of the plot that is produced in R.
  - This is NOT the size that will appear in the LaTeX document.
  - LaTeX defaults to `textwidth`: use `\setkeys{Gin}{width=0.8\textwidth}` to change the size of the figure in LaTeX (change 0.8 with something else).
- Only one figure for each chunk is produced.

Figures

- I prefer to define a `label` with `fig=TRUE` and `include=FALSE` in the chunk and then place manually the figure wherever I prefer (even way before the chunk that actually generated it).

Example:

```
\SweaveOpts{prefix.string=EPFL}
⋮
<<label=histx,fig=TRUE,include=FALSE>>=
hist(x)
@
⋮
\begin{figure}
\includegraphics[width=5in]{EPFL-histx}
\caption{Histogram of x.} \label{histogram-x}
\end{figure}
```

Another example: example-2.Rnw

```
\documentclass[a4paper]{article}

\SweaveOpts{echo=true}

\begin{document}


First we define a figure hook:
<<results=hide>>=
options(SweaveHooks = list(fig = function() par(mfrow=c(2,2))))
@

Then we setup variable definitions without actually evaluating them
<<xydef,eval=false>>=
x <- 1:10
y <- rnorm(x)
@


Then we put the pieces together:
\begin{center}
<<fig=T>>=
<<xydef>>
lm1 <- lm(y~x)
summary(lm1)
plot(lm1)
@
\end{center}

\end{document}
```

. . . which produces

First we define a figure hook:

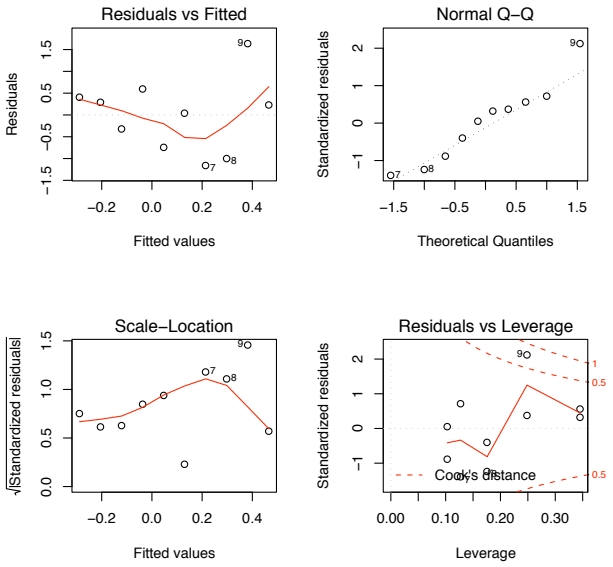
```
> options(SweaveHooks = list(fig = function() par(mfrow = c(2,
+     2))))
```

Then we setup variable definitions without actually evaluating them

```
> x <- 1:10
> y <- rnorm(x)
```

Then we put the pieces together:

```
> x <- 1:10
> y <- rnorm(x)
> lm1 <- lm(y ~ x)
> summary(lm1)

Call:
lm(formula = y ~ x)

Residuals:
   Min     1Q Median     3Q    Max
-0.922 -0.318 -0.120  0.386  1.204

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.7493     0.4471   -1.68     0.13
x             0.0580     0.0721    0.80     0.44

Residual standard error: 0.654 on 8 degrees of freedom
Multiple R-Squared: 0.0748,        Adjusted R-squared: -0.0408
F-statistic: 0.647 on 1 and 8 DF,  p-value: 0.444

> plot(lm1)
```



Yet another example: example-3.Rnw

```
\documentclass[a4paper]{article}

\begin{document}

<<echo=false,results=hide>>=
library(lattice)
library(xtable)
data(cats, package="MASS")
@

\section*{The Cats Data}

Consider the \texttt{cats} regression example from Venables \& Ripley
(1997). The data frame contains measurements of heart and body weight
of \Sexpr{nrow(cats)} cats (\Sexpr{sum(cats$Sex=="F")} female,
\Sexpr{sum(cats$Sex=="M")} male).

A linear regression model of heart weight by sex and gender can be
fitted in R using the command
<<>>=
lm1 = lm(Hwt~Bwt*Sex, data=cats)
lm1
@
Tests for significance of the coefficients are shown in
Table~\ref{tab:coef}, a scatter plot including the regression lines is
shown in Figure~\ref{fig:cats}.

\SweaveOpts{echo=false}

<<results=tex>>=
xtable(lm1, caption="Linear regression model for cats data.",
label="tab:coef")
@

\begin{figure}
  \centering
<<fig=TRUE,width=12,height=6>>=
trellis.par.set(col.whitebg())
print(xyplot(Hwt~Bwt|Sex, data=cats, type=c("p", "r")))
@
  \caption{The cats data from package MASS.}
  \label{fig:cats}
\end{figure}
\begin{center}
\end{center}

\end{document}
```

. . . which produces

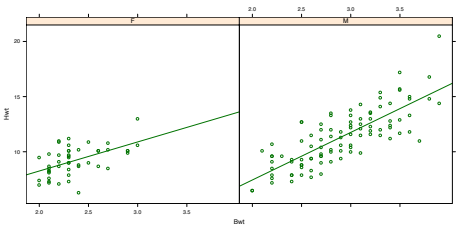


Figure 1: The cats data from package MASS.

The Cats Data

Consider the `cats` regression example from Venables & Ripley (1997). The data frame contains measurements of heart and body weight of 144 cats (47 female, 97 male).

A linear regression model of heart weight by sex and gender can be fitted in R using the command

```
> lm1 = lm(Hwt ~ Bwt * Sex, data = cats)
> lm1

Call:
lm(formula = Hwt ~ Bwt * Sex, data = cats)

Coefficients:
(Intercept)          Bwt          SexM      Bwt:SexM
       2.98         2.64        -4.17         1.68
```

Tests for significance of the coefficients are shown in Table 1, a scatter plot including the regression lines is shown in Figure 1.

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	2.9813	1.8428	1.62	0.1080
Bwt	2.6364	0.7759	3.40	0.0009
SexM	−4.1654	2.0618	−2.02	0.0453
Bwt:SexM	1.6763	0.8373	2.00	0.0472

Table 1: Linear regression model for cats data.

A few more comments

- The use of `label` allows chunk reuse (as in `example-2`).
- It is clear from `example-2` that when the data change, the final document changes accordingly.
- `\Sexpr{}` (as in `example-3`) allows the evaluation of R objects within documentation chunks (only `character` or something that can be coerced to `character`).
- Sweave combined with the package `xtable` (as in `example-3`) produces nice LaTeX tables from R objects.

Stangle slide 24

Stangle

- Run the `Stangle("myfile.Rnw")` command on R.
- This will ignore all LaTeX code and gather all R code.
- `\Sexpr{}` expressions in the text are ignored.
- It will create a text file named `myfile.R` with all the chunks of R code (again, the use of `label` is very useful).
- Chunks with `eval=FALSE` will be included but commented out.
- The file created can be sourced into R.

Padova - PhD Program slide 24

LaTeX syntax slide 25

A different syntax: LaTeX

If the source code has extension `.Rtex` (or `.Stex`) then an alternative syntax (to Noweb) is used:

- `<< options >>=` is replaced by a more usual `\begin{Scode}{options}`
- `@` is replaced by `\end{Scode}`.
- `\Scoderef{chunkname}` is used for chunks reuse.
- Everything else is exactly the same.
- The choice of the syntax can be set with an option in the `Sweave` command (regardless of the extension).

Padova - PhD Program slide 25

Sweatex slide 26

Sweatex: do it all at once

- I wrote a simple R function, called `Sweatex`, that runs Sweave on the source file (default `extension="Rnw"`) and then pdflatex (default), or latex, on the resulting `.tex` file.
- In any case, it produces a PDF file as an output.
- Possibility to launch a PDF preview directly after the compilation process (option `preview=TRUE`, default is `FALSE`).
- Usage:

```
> Sweatex("myfile", command = "latex",
+     preview = TRUE)
```

- It SHOULD work in any system: I'm happy to share it...

Padova - PhD Program slide 26

```
> Sweatex

function (filename, extension = "Rnw", command = "pdflatex",
    silent = FALSE, preview = FALSE, bibtex = FALSE)
{
    latex.path <- dirname(options("latexcmd")[[1]])
    path <- as.character(Sys.getenv("PATH"))
    if (regexpr(latex.path, path) == -1) {
        Sys.putenv(PATH = paste(path, ":", latex.path, sep = ""))
    }
    if (command == "latex")
        command = "simpdftex latex --maxpfb"
    extension <- paste(".", extension, sep = "")
    Sweave(paste(filename, extension, sep = ""))
    if (bibtex) {
        system(paste(command, " ", filename, sep = ""), intern = silent)
        system(paste("bibtex", " ", filename, sep = ""), intern = silent)
        system(paste(command, " ", filename, sep = ""), intern = silent)
    }
    else system(paste(command, " ", filename, sep = ""), intern = silent)
    if (preview) {
        system(paste(options("pdfviewer")[[1]], " ", filename,
            ".pdf", sep = ""))
    }
}
```


A few tips

Time consuming computations...

If time consuming chunks are present (such as simulations), it may be worthwhile to execute them only once, save the results and load them in future compilation processes, thus avoiding the need for evaluation.
This can be achieved, for instance,

- by modular typesetting in LaTeX (using `\include`),
- by caching results of long computations: see packages `cacheSweave` and `weaver`.

Something useful...

I find useful to include something like the following text somewhere in the text (as footnote in the first or last page):

"These slides have been generated on March 10, 2008 with R version 2.6.2 (2008-02-08) on a i386-apple-darwin8.10.1 platform."

This is simply obtained with:

- `\today`,
- `\Sexpr{print(version$version.string)}`,
- `\Sexpr{print(version$platform)}`.

Sometimes default options of functions may change in different version of R (an example is `plot(lm)` since version 2.2.0): you may want to know which version of R generated your report.

Concluding remarks

slide 30

Concluding remarks

- Your analysis is reproducible. Even after many months, when you've completely forgotten what you did...
- Your analysis actually works...at least in this particular instance. The code you show actually executes without error.
- Toward the end of your work, with the write-up almost done you discover an error. Months of rework to do? No! Just fix the error and rerun `Sweave` and `latex`.
- This methodology provides discipline. There's nothing that will make you clean up your code like the prospect of actually revealing it to the world.
- Whether we're talking about classnotes, a consulting report, a textbook, or a research paper, this should be the way to do it (perhaps at different levels of usage).

Padova - PhD Program slide 30

References

slide 31

References

These slides are (strongly) based on the following material:

1. Geyer C.J. (2005). An Sweave Demo.
2. Leisch, F. (2002). Sweave: Dynamic generation of statistical reports using literate data analysis. In Compstat 2002 - Proceedings in Computational Statistics, pages 575-580. Physika Verlag, Heidelberg, Germany. (Use it to cite Sweave!).
3. Leisch, F. (2002). Sweave user manual. Institut fur Statistik und Wahrscheinlichkeitstheorie, Technische Universitat Wien, Vienna, Austria, 2002.
4. Leisch, F. (2003). Sweave and beyond: Computations on text documents. In Proceedings of the 3rd International Workshop on Distributed Statistical Computing, Vienna, Austria, 2003.

All references are available on line. In particular:

`http://www.stat.umn.edu/~charlie/Sweave` (1.)

`http://www.ci.tuwien.ac.at/~leisch/Sweave` (2.-4.)

`http://www.statistik.lmu.de/~leisch/Sweave` (new homepage)

Padova - PhD Program slide 31