Automating reports with Sweave

Greg Snow, SDC

Sweave is a system for automatically adding statistical output to a document. Currently sweave only works with the statistical package R (that I know of, others may be in the works) and with LaTeXand HTML as the documentation systems. LaTeXis a system for preparing documents, LaTeXcan then create the document as a postscript or PDF file, or the document can be easily converted to HTML or RTF (rtf files can be loaded into popular word processors).

R is a statistical programming language and analysis package (see `http://www.r-project.org/`). R is based on the S programming language by Chambers et. al. and can be downloaded for free. Many additional packages exist for R to expand the capabilities of the base program (the "qcc" package adds functions to plot SPC charts etc.).

Sweave is implemented as the function `Sweave` in R. The basic idea of Sweave is to write your report in the same manner that you regularly would, with the exception that instead of pasting in output and graphs from a statistical analysis, you insert the R code that generates the output. The file is then processed by the `Sweave` function in R and the pieces of code are replaced with the output and graphics. One big advantage of this is that if something changes (new data, correcting a data error, . . .) you can just reprocess the file and have the updated final report quickly rather than having to rerun and recut and repaste the updated output.

The file you create should follow the guidelines for the document language you are using (LaTeXor HTML). Pieces of code start with the line `<<>>=` (with the first `<` flush left and nothing after the `=`). You indicate the end of the R code with a single at sign `@` in the furthest left column with nothing else on that line. Some options can be included between the angle brackets. A base HTML file can be created using MS Word, but the code pieces need to be inserted using a text editor (Word modifies the start of code indicator).

Look at the following input file using HTML as the document language (named sample.Rnw):

```
1  <HTML>
2  <Head>
3  <Title>
4  Demonstration of Sweave
5  </Title>
6  </Head>
7  <Body>
8
9  <H1>
10 This is an example of using Sweave.
11 </H1>
12
13 <<echo=F>>=
14 library(qcc)
15 mydata <- read.table('c:/projects/samp.dat')
16 @
17
18 Dataset has been loaded with <Sexpr nrow(mydata)> groups observed.
19
20 <<fig=T>>=
21 qcc(mydata,type='xbar')
22 @
23
24 </Body>
25 </HTML>
```

Lines 1–11: This is standard HTML code to begin the document.

Line 13: This indicates that the following lines are R code rather than HTML markup, the option `echo=F` means that the input code on the following lines will not be included in the final documents (and since neither of the following lines produces output the corresponding section of the HTML file will be empty).

Line 14: This command to R loads the “qcc” library which will make functions for spc charts available.

Line 15: This loads the data from the file "samp.dat" in the "projects" directory and puts the data into a variable named "mydata".

Line 16: The single `@` sign indicates the end of this block of R code and return to HTML markup.

Line 18: This is an HTML paragraph with the markup command `<Sexpr nrow(mydata)>`. This will be replaced with the result of the R command `nrow(mydata)` (the number of rows of data we just read in). This is the way to insert simple values into a paragraph.

Line 20: This indicates the start of another R code block. The `fig=T` option indicates that the following code will generate a graph and the graph should be included in the final report.

Line 21: The `qcc` command creates spc charts ($\bar{x}$ in this case) as well as printing some other summary information. If you only want the graph and not the printed output then assign the result to a variable (`tmp <- qcc(mydata,type='xbar')`).

Line 22: End the code block.

Lines 24–25: HTML markup finishing the file.

Save the above in a file (this one was "sample.Rnw", some of the syntax comes from a system called "NoWeb", hence the nw in the name). Inside of R run the following commands (the `>` is the R prompt):

```
> library(R2HTML)
> Sweave("sample.Rnw", driver=RweaveHTML)
```

This will create the file "sample.html" with the R code replaced with the output.

Another possibility is to create a Makefile with the following command:

```
echo "library(R2HTML); Sweave('sample.Rnw', driver=RweaveHTML)" | R --no-save --no-restore
```

Then you can use the make utility to update the final reports anytime something changes.