A QUICK GUIDE ON SWEAVE*

CHRISTOPHER F. PARMETER

> "The only people who have anything to fear from free software are those whose products are worth even less." - David Emery

1. Sweave

1.1. **What is Sweave?** Most R users also use LaTeX. Sweave is a package that 'weaves' these two softwares together to allow you to produce fully reproducible research (Meredith & Racine 2009, Koenker & Zeileis 2009). Reproducible research can be easily defined using *Claerbout's Principle* (Buckheit & Donoho 1995, de Leeuw 2001):

> An article about computational science in a scientific publication is not the scholarship itself, it is merely advertising of the scholarship. The actual scholarship is the complete software development environment and the complete set of instructions which generated the figures.

It is not enough to have published a paper in a reputable economics/econometric journal. The ability for outside readers to fully reproduce your research is what truly matters in scientific endeavors.

Consider why the ability to reproduce another author's research is so important.When you conduct research currently it probably proceeds in the following steps. First, you construct a .tex file that contains your thoughts on the topic of interest as well as your econometric model and a data description. You then run your results in a statistical software environment (such as R, STATA or Matlab). You then construct tables and figures and import them back into your .tex file. You then clean up your paper and jockey your friends at other universities to invite you for a seminar so that you can increase your awards miles and get a few free meals. Then you submit your paper to a reputable journal and wait (sometimes for quite a while). Finally, after 4-5 resubmissions you are granted a revision (editors don't like using the word accepted until after your paper is published). This revision inevitably entails reanalyzing your data which means adding columns to tables and creating new figures. This places an onus on the researcher to keep track of where the original files were, which code produced which columns of the tables and the figures. In general you will have numerous versions of your code along with figures and tables that did not make it into the final product. So, tracking down where the correct code is may not be straightforward.

University of Miami
Date: January 12, 2012.
Christopher F. Parmeter, Department of Economics, University of Miami; e-mail: cparmeter@bus.miami.edu.
*These notes have been prepared for Camp Resources XVIII. The generous hospitality of the conference organizers is warmly recognized. I would also like to thank Jeff Racine whose work in this area is unparalleled. Numerous discussions with him have led to a greater understanding of the topics here.

Once your research is published you will bask in the glow of your name appearing in print and graduate students from around the world asking for your data and codes so that they can attempt to replicate your work. Now it is even longer since you have thought about your code and where your data is. You may send the wrong code over or simply cannot track it down.

What is the moral of this story? Sweave can help alleviate these burdens and ensure that your research is fully reproducible (down to the coloring of the figures). Sweave is no harder to learn than learning LaTeXor R. In fact, if you currently use LaTeXto typeset your documents and you program in R there is no reason not to use Sweave.

1.2. **How Can I get Sweave?** Sweave is a set of commands within the `utils` package in R (Leisch 2002). This package is automatically available upon start up of R so there is no need to call a `library` command. Don't think of using Sweave as coding though. At the end of writing your document almost everything will still appear like a normal LaTeXfile would. When you create a document to weave R and LaTeX, instead of having a .tex extension, it will have a .Rnw extension (which stands for R no weave). When you Sweave the .Rnw file it will then produce a .tex document which you can then compile as you normally would.

1.3. **So What Do I Do With Sweave?** When you construct your .Rnw file it will have the look of a .tex file. The only difference will be that you will have a special code that lets Sweave know whether you are writing (LaTeX) or running code (R). Sweave uses what is known as `NOWEB` syntax (Ramsey 2008). The `NOWEB` syntax for beginning code in the .Rnw file is `<<>>=` while the `NOWEB` syntax for ending code is `@`. Your R code would go in between these two syntactical instructions. Here is an example:

```
<<example,eval=FALSE,echo=TRUE,results=verbatim,keep.source=TRUE>>=
#YOUR R CODE GOES HERE!
@
```

Now, this may seem a little awkward. Inside the `<<>>=` call you see things like `example`, `eval`, `echo`, `results`, and `keep.source`. These are simply commands to Sweave so that it knows what to do with the code. The first term, `example` is your name for the chunk of R code you have. This is not necessary, but I like to have it so that when I strip my code off of the .Rnw file (more on that later) each piece has a title. The `eval` portion is very simple. It can either be set to `TRUE` or `FALSE`. If it is set to `TRUE`, this means that the code chunk is evaluated, otherwise it is not. The default for `eval` is `TRUE`.[1] The `echo` command tells Sweave whether or not you would like the R code written inside the LaTeXdocument. Thus, you can put code into your .Rnw file that will not appear in your .tex file whatsoever. In general most of the documents you create for research will not have any code echoed to the .tex file. The `results` command has three options, `verbatim`, `hide` and `tex`. `verbatim` means that the results of the code that would be printed in your R console are also printed inside the .tex document. `hide` means that results are not printed in the .tex document, while `tex` will produce results that look like you had created them inside a .tex

[1]Sometimes if you want to quickly compile your .Rnw file, you may turn the evaluation of R chunks off.

document (i.e., properly coded LaTeX). Finally, `keep.source` tells Sweave if you want comments and things of that nature which appear in your R code to also be placed in the .tex file. There are several additional commands that go within the `<<>>=` call, but in principle those listed above will be the main commands to focus on.

One final point I would like to make (which can be vexing). Your `<<>>=` and `@` calls must be fully left aligned in your .Rnw file. If they are tabbed over or appear in a list environment Sweave will not recognize them as proper `NOWEB` syntax. Just be aware of this when you compile and you are wondering why no R code is being run. This is also the case if you wish to create a beamer document using Sweave.

2. An example

To demonstrate the flexibility and ease of use of Sweave we will now conduct a basic regression analysis. Regression analysis makes for a useful tool since a majority of empirical research will make use of these methods. First we will load in the `Ecdat` package (Croissant 2011) and gain access to the `Hedonic` dataset of Harrision & Rubinfeld (1978). The following Sweave commands would do this for us:

```
<<Boston,echo=FALSE,results=hide>>=
## Load Ecdat package
library(Ecdat)
data("Hedonic")
@
```

Having loaded the dataset we can now run a simple log-linear hedonic regression. The following Sweave commands would do this for us:

```
<<Boston_regression,echo=FALSE,results=hide>>=
Hedonic$lmv <- log(Hedonic$mv)
attach(Hedonic)
model <- lm(lmv~nox+crim+chas+lstat+dis,data=Hedonic)
@
```

Having run a regression (or 20) we can now print our results. There are two main options. The first would be to print the R results verbatim by

```
<<Boston_summary,echo=FALSE>>=
summary(model)
@
```

```
Call:
lm(formula = lmv ~ nox + crim + chas + lstat + dis, data = Hedonic)

Residuals:
      Min        1Q    Median        3Q       Max
```

```
-0.087226 -0.011513 -0.000627  0.011532  0.074198

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.2267606  0.0082635 269.470  < 2e-16 ***
nox         -0.0005278  0.0001136  -4.645 4.35e-06 ***
crim        -0.0013849  0.0001246 -11.117  < 2e-16 ***
chasyes      0.0138113  0.0037276   3.705 0.000235 ***
lstat       -0.0497620  0.0019333 -25.739  < 2e-16 ***
dis         -0.0176785  0.0029400  -6.013 3.51e-09 ***
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.1 âĂŸ âĂŹ 1

Residual standard error: 0.02082 on 500 degrees of freedom
Multiple R-squared: 0.751,        Adjusted R-squared: 0.7485
F-statistic: 301.6 on 5 and 500 DF,  p-value: < 2.2e-16
```

but as you can see this is not that appealing and detracts from the flow of your .tex document. Alternatively, you can instruct Sweave to print out the results in tabular form. Using the `xtable`, `memmisc` or `Hmisc` packages will help you with this endeavor. Some generic code for producing a LaTeXtable is

```
 <<Boston_table,echo=FALSE,results=tex>>=
library(xtable)
print(xtable(model, caption="Regression Output for Boston Dataset",
              label="tab:sam",digits=3),
      latex.environment="center",
      caption.placement="top",table.placement="h!")
```

Table 1. Regression Output for Boston Dataset

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	2.227	0.008	269.470	0.000
nox	-0.001	0.000	-4.645	0.000
crim	-0.001	0.000	-11.117	0.000
chasyes	0.014	0.004	3.705	0.000
lstat	-0.050	0.002	-25.739	0.000
dis	-0.018	0.003	-6.013	0.000

The elegance of Sweave can be appreciated with this example. While this table may not look exactly how you want your table to appear, if you wanted to change features of the regression, nothing in your `xtable` call would need to be changed. You would simply have to change your call to `lm()`

and upon recompiling your table would automatically be updated, no having to go between excel or something of that ilk and your .tex document.

Aside from tables another desirable output from your research is figures to help readers visualize what is happening. Figures can also be included in a .Rnw document by using the option `fig=TRUE` inside `<<>>=`. For example, the following chunk will plot the fitted OLS line for the generic regression of `lmv` on `nox` along with a scatterplot of the data.

```
 <<Boston_figure,echo=FALSE,fig=TRUE>>=
fit <- fitted(lm(lmv~nox))
ord <- order(nox)
plot(nox[ord],fit[ord],type="l",
        col="blue",lwd=2,lty=2)
points(nox,lmv,col="grey",cex=0.9)
 @
```

The current setup of Sweave will only allow one figure to be produced per code chunk. This can easily be remedied by specifying a hook function prior to the code chunk being evaluated. The following Sweave command would do this:

```
 <<multiple_figure,echo=FALSE,results=hide>>=
 options(SweaveHooks=list(multiple_fig=function() par(mfrow=c(2,2))))
```

Using this command along with the following chunk

```
 <<Boston_multiple_figure,fig=TRUE,echo=FALSE,multiple_fig=TRUE>>=
 plot(model)
```

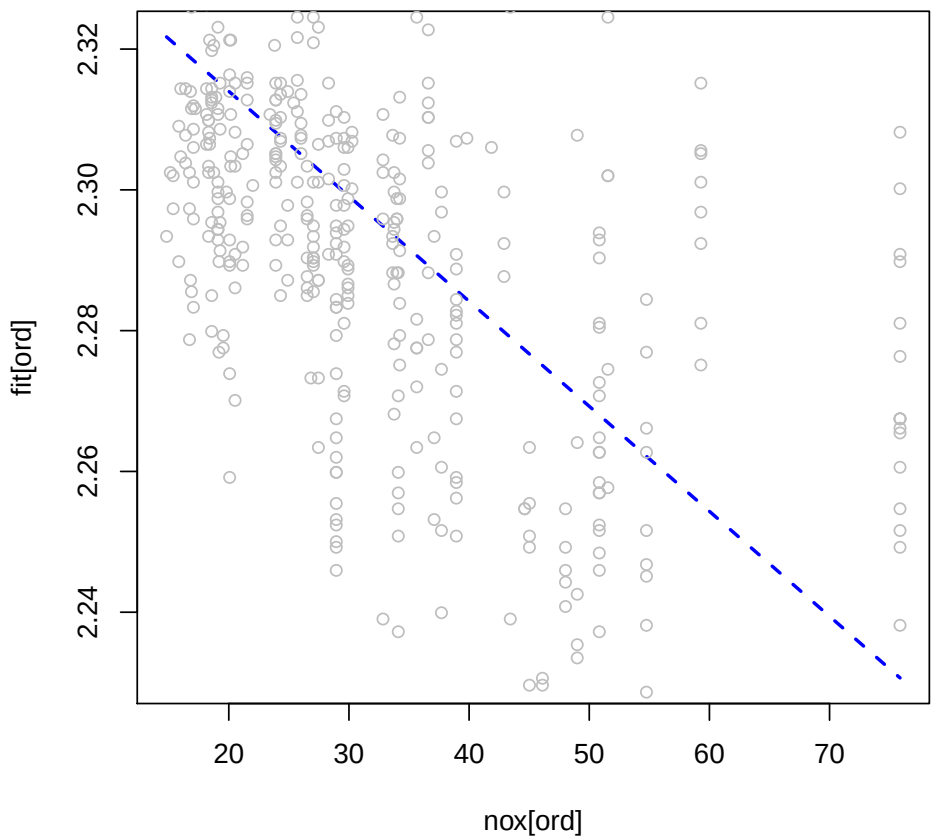
will produce Figure 2.

3. Additional Features

Two additional features of Sweave are cacheing and calling R withing the body of text. Invariably, structural econometric research goes well beyond simple linear regression and can be computationally demanding. Generically using Sweave means you cannot have the finished .tex document until all the code has run. This would seem to make Sweave an unattractive alternative. However, when code chunks exist that require a large run time one can cache the code while envoking Sweave. There are several options to cache code. In this setting, code chunks that require prolonged run times will be placed in the background enabling the Sweave process to run quicker and the .tex document can be quickly edited so that the code does not need to be reevaluted for each compilation of the .Rnw file. Two readily available packages to cache are `weaver` (Falcon n.d.) and `pgfSweave` (Bracken & Sharpsteen 2011).

To include the results from simple command line calls within your code the `\Sexpr{}` command is invaluable. To use this command simply type `\Sexpr` followed by the relevant code you wish to evaluate. Suppose you wanted to compare the standard deviations of `nox` and `lstat` from the

FIGURE 1. Fitted regression line of `lmv` on `nox`.

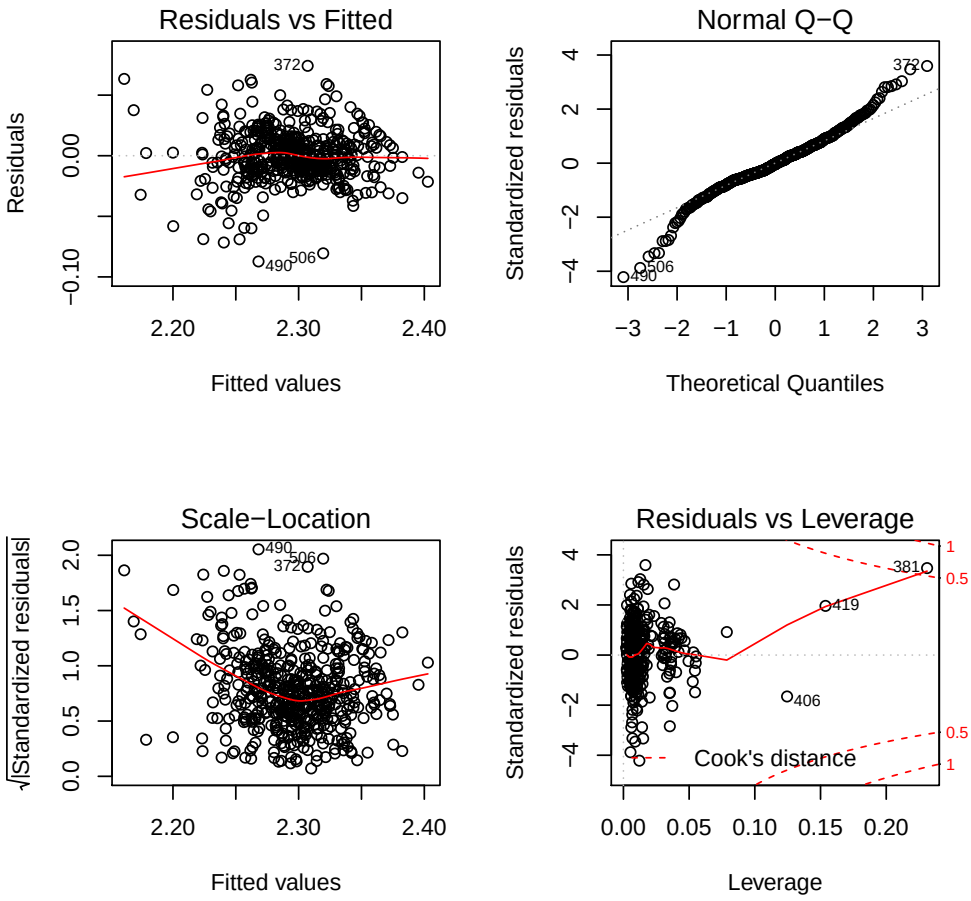


`Hedonic` dataset. Then `\Sexpr{sd(nox)}` and `\Sexprsd{sd(lstat)}` would provide 13.921 and 0.601. Be aware that calls to `\Sexpr{}` cannot have a line break within them.

Another option available to Sweave users is `Stangle()`. `Stangle()` works in the opposite manner that `Sweave` does. `Stangle()` will extract all of the R code from a .Rnw file. This means that you can easily construct a .R file fron your .Rnw file to use for debugging, giving your code away (in case you have to deal with people who don't use LaTeX) and providing code so that others don't have to copy/paste to run just your code and make edits to it. An invocation of `Stangle()` will produce a .R file that has the same name as the .Rnw file. To `Stangle` simply type

```
> Stangle(filename.Rnw)
```

FIGURE 2. Regression Diagnostic Plots for Linear Model.



References

Bracken, C. & Sharpsteen, C. (2011), *pgfSweave: Quality speedy graphics compilation and caching with Sweave*. R package version 1.2.1.
URL: *http://CRAN.R-project.org/package=pgfSweave*

Buckheit, J. B. & Donoho, D. L. (1995), Wavelab and reproducible research, *in* A. Antoniadis & G. Oppenheim, eds, 'Wavelets in Statistics: Lecture Notes in Statistics', Springer: New York, pp. 55–82.

Croissant, Y. (2011), *Ecdat: Data sets for econometrics*. R package version 0.1-6.1.
URL: *http://CRAN.R-project.org/package=Ecdat*

de Leeuw, J. (2001), *Reproducible Research: The bottom line*. Technical Report 2001031101, Department of Statistics Papers, University of California, Los Angeles.
URL: *http://repositories.cdlib.org/uclastat/papers/2001031101*

Falcon, S. (n.d.), *weaver: Tools and extensions for processing Sweave documents*. R package version 1.18.0.

Harrision, D. & Rubinfeld, D. L. (1978), 'Hedonic housing prices and the demand for clean air', *Journal of Environmental Economics and Management* **5**, 81–102.

Koenker, R. & Zeileis, A. (2009), 'On reproducible econometric research', *Journal of Applied Econometrics* **24**, 833–847.

Leisch, F. (2002), Sweave: dynamic generation of statistical reports using literate data analysis, *in* W. Hardle & B. Ronz, eds, 'Compstat 2002: Proceedings in Computational Statistics', pp. 575–580.

Meredith, E. & Racine, J. S. (2009), 'Towards reproducible econometric research: The Sweave framework', *Journal of Applied Econometrics* **24**, 366–374.

Ramsey, N. (2008), *Noweb: a simple, extensible tool for literate programming*.
URL: *http://www.eecs.harvard.edu/nr/noweb/*