Using Sweave with TeXShop 2.47

Duncan Murdoch

October 27, 2013

1   Setup

These instructions apply to the current version of TeXShop on OS X 10.6.8, available from `http://www.tug.org/TeXShop`. Users of more recent OS X versions should use TeXShop 3.x. I don't know if these instructions apply to that version, but the same steps should be possible.

1.1   Telling TexShop to handle Sweave documents

Sweave documents are typically named *.Rnw. You want TeXShop to recognize these documents and treat them as if they are LaTeX.

First, you need to tell OS X to associate them with TeXShop: In the Finder, right click on a *.Rnw file, and choose

Open with...

and choose TeXShop from the offered list. Now, TeXShop will start whenever you double click on a Rnw file.

1.2   Telling TeXShop to use Sweave

Sweave needs to run before LaTeX runs. The simplest way to do this is to install a small package into R.

1.2.1   Installing the patchDVI package

This package is available on CRAN. Install it in the usual way, for example start R, and run the command

```
install.packages("patchDVI")
```

For more details about patchDVI, see the vignette included with the package by running

```
> library(patchDVI)
> vignette("patchDVI")
```

in R.

Development versions of the patchDVI package are available on R-forge.r-project.org.

1.2.2   Telling TeXShop to use patchDVI

You need to create a new Sweave engine. Your home directory should have a `Library/TeXShop/Engines` subdirectory. You create a new file there called `Sweave.engine`, containing the text

```
#!/bin/tcsh

# set path= ($path /usr/local/bin)
Rscript -e "patchDVI::SweavePDF(  '$1' )"
```

(The commented line may be needed if R is not already on your path.) This file needs to be given executable permission.

1.3   Telling the previewer to jump back to the editor

You can right-click on a line in the previewer, and go to that line in the source document by choosing `Sync`.

1.4   Jumping to the .Rnw file, not the .tex file

TeXShop uses information encoded in the .synctex file produced by pdflatex to know where to jump. LaTeXputs information about the .tex file that comes out of Sweave, not the .Rnw file that went in. The patchDVI package can convert the information.

To do so, put the lines

```
\usepackage{Sweave}
```

```
\SweaveOpts{concordance=TRUE}
```

early in your .Rnw file. Now sync'ing will switch to the right file.

2 Embedding R code in your document

2.1 Basics of Sweave

The Sweave manual gives a much more complete description; I'll just give a very short one here.

The idea of Sweave is that your document corresponds to a session of R. You embed text like

```
<<fig=TRUE>>=

set.seed(123)
x <- 1:10
y <- rnorm(10)
y
plot(x, y)

@
```
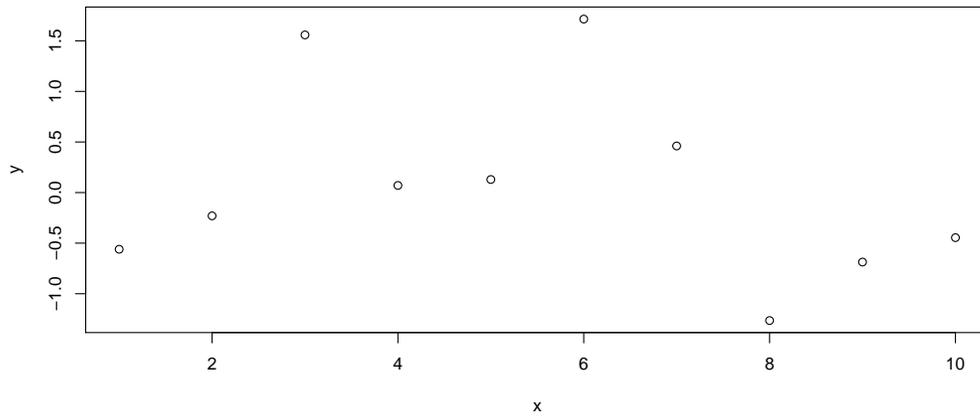
into your .Rnw file, and Sweave converts it into this:

```
> set.seed(123)
> x <- 1:10
> y <- rnorm(10)
> y

 [1] -0.56047565 -0.23017749  1.55870831  0.07050839
 [5]  0.12928774  1.71506499  0.46091621 -1.26506123
 [9] -0.68685285 -0.44566197

> plot(x, y)
```

You don't need to worry about the details of importing the code, the results or the figure; Sweave does that for you, by producing the figure in both EPS and PDF format, and generating LaTeX code like this:

```
\begin{Schunk}
\begin{Sinput}
> set.seed(123)
> x <- 1:10
> y <- rnorm(10)
> y
\end{Sinput}
\begin{Soutput}
 [1] -0.56047565 -0.23017749  1.55870831  0.07050839
 [5]  0.12928774  1.71506499  0.46091621 -1.26506123
 [9] -0.68685285 -0.44566197
\end{Soutput}
\begin{Sinput}
> plot(x, y)
\end{Sinput}
\end{Schunk}
\includegraphics{figs/-002}
```

The Schunk, Sinput and Soutput macros can be customized if you want fancy colours or other things in your document.

4

2.2   Recommended Sweave options

We've already seen the line `\SweaveOpts{concordance=TRUE}` above. There are other Sweave options I often use:

height=5, width=10 These allow you to specify the height and width of plots produced by R. I tend to choose values slightly larger than I want the plot to appear, because I like a slight reduction in the font sizes that results when LaTeX shrinks it.

keep.source=TRUE If you don't choose this, then R will reformat all of your carefully written source code.

prefix.string=figs/ This says to store all the figures (and the concordance) in a subdirectory called figs, so they don't mess up your main directory. You need to create that subdirectory or this will fail.

These can all be combined into one `\SweaveOpts{}` call, by separating them with commas.

I also set some options to the graphicx package, so my figures look consistent:

```
\setkeys{Gin}{width=\textwidth}
```

This says that figures should be resized so that they fit the full width of the text.

And finally, an option to R:

```
options(width=60)
```

will shorten the output lines that R produces, so they look nicer in a document. To get this to execute without showing up in your document, include it as

```
<<echo=FALSE>>=
```

```
options(width=60)
```

```
@
```

early in your document.