Generating beautiful statistical reports quickly and faithfully

Yihui Xie

Department of Statistics, Iowa State University

$Y = \beta_0 + \beta_1 x + \epsilon$

2012-02-03

Introduction to knitr

Yihui Xie
Automate, Automate, Automate!

gently pollute professors’ data with Excel
all-mighty R gives us everything
copying and pasting, we submit the homework

– Yihui’s haiku on homework
Reproducible Research

• with the same tools and data, I can reproduce what you have got

• manual work is easy to be polluted
  – copied a wrong row
  – pasted into a wrong worksheet
  – typo
  – coffee spilled on keyboard
  – tired
• computers are good at doing tedious jobs as long as you tell them the right commands

• humans focus on commands

• but commands are often ugly and do not make a report by themselves
Literate Programming

• Donald Knuth

• basic idea: mix code and prose, compile code to its results, and get a mixture of texts

• A haiku example

today is \{r Sys.Date() r\}
I feel very happy
so I wrote a haiku
• then you can write a different haiku for 365 days by compiling the document again and again

today is 2012–02–14
I feel very happy
so I wrote a haiku

• we are talking about statistical reports, so there are much more details to take care of
Previous Work

• 10 years of Sweave (S + weave)

• several R packages derived from there (cacheSweave, pgfSweave, odfWeave, R2HTML, highlight, …)

• other statistical software: SASweave, Statweave, …

• other languages: Org-mode (Emacs), Javadoc, noweb, cweb, …
Why Reinvent the Wheel

• you love it more, you hate it more

• open source – how “open”?
  – request for bitmap devices support (png, jpeg)
  – 700 lines of source code: want to add 300 lines? copy 700 lines;
    want to add 3 lines? copy 700 lines
  – want more flexible output? hackish, hackish, hackish (e.g. \setkeys{Gin})
The Name

• knitr = knit + R (compare to S + weave); neater?

• tangle: extract code from the document
  – Stangle() = S + tangle; knitr has purl() (suggestion from a German wife; why?)

• new name: stitch() (suggested by an Indian)
  – insert an external R script into an existing template, and compile
Features

- code formatting
- fine control of graphics
- cache
- code externalization
- conditional evaluation
- customization
Syntax

• start code: `<<meta-information>>=
  – `<<my-label, eval=TRUE, fig.width=5, cache=FALSE>>=
  – `<<foobar, echo=FALSE, fig.show=animate>>=

• start prose: @

• other types of syntax are also possible, e.g. in HTML
  – `!!--begin.rcode my-label, fig.width=5, dev=png
  – `end.rcode-->

Code Formatting

- **formatR** (reformat code) and **highlight** (highlight code)

```r
## option tidy=FALSE for this chunk; TRUE for next
for(k in 1:10){j=cos(sin(k)*k^2)+3;print(j-5)}
```

```r
for (k in 1:10) {
  j <- cos(sin(k) * k^2) + 3
  print(j - 5)
}
```
• graphical devices: PDF, PNG, JPEG, Cairo-series, tikz

```r
library(ggplot2)  # show off the tikz device
qplot(1:10, 1:10) + opts(title = 
sprintf("$\mathbf{d}x_t = \alpha(\theta - \mathbf{x}_t) \mathbf{d}t + \%dB_t$",
4))
```
\[ dx_t = \alpha[(\theta - x_t)dt + 4]dB_t \]
two plots side by side

```r
fit <- lm(dist ~ speed, data = cars)
par(mar = c(4, 4, 1, 0.1), mgp = c(2, 1, 0))
with(cars, {
  plot(speed, dist, panel.last = abline(fit))
})
text(10, 100, "$Y = \beta_0 + \beta_1 x + \epsilon$"
).library(ggplot2)
qplot(speed, dist, data = cars) + geom_smooth()
```
\[ \text{dist} = \beta_0 + \beta_1 \text{speed} + \epsilon \]

- animation
demo("Mandelbrot", "animation", echo = FALSE)
Cache

- may not need to re-compute a chunk if it is not changed
Code Externalization

• do not have to put code in the document

• it can be externalized

• you may not be well-prepared for a report and need to play with R before you put the code in

• inconvenient to run code in a literate document
Customization

- you have control over almost everything

- hook functions can be used to customize output
  - wrap output in \LaTeX{} or HTML?
  - need to capture a screenshot for a chunk?
Documentation

- website: http://yihui.name/knitr
Boasting Time


• wide attention with a few surprises (WV, BB, MF, FH, JA)

• donation (German again)

• RStudio support

• LyX support (2.0.3)
• applications
  – UCLA online learning resources
  – ggplot2 transition guide to 0.9.0

• design of website
  – Jekyll + GitHub
  – say goodbye to old clunky R help pages
  – more than 3 hours to pick an icon
  – font

• 118 issues so far; most from users
Convinced?

- professors do not have to suffer from ugly Word documents
- students may start using \texttt{stitch()} (beginners) and \texttt{knit()} (advanced users)
Future Development

• endless bug fixing, of course

• use it to update vignettes of my other packages (e.g. cranvas)

• automatic web building for R packages? (think ggplot2’s website)

• foster a community (Google Groups, R-help, StackOverflow)

• publication