

Using R Markdown in Emacs

2014-08-30

It is possible to compile R markdown files (.Rmd) to pdf and html in a similar fashion to L^AT_EX. The advantage is the ability to write with the simplicity and elegance of markdown with the power of adding in L^AT_EX expressions and running R code. The trick is the [rmarkdown](#) package, which plays a role similar to `pdflatex`.

But there's a problem: in order to compile your markdown file, you need to launch an R process and run:

```
library(rmarkdown); render("file_name.Rmd")
```

This is extremely tedious if your not actually interested in running any R code at the time. Instead, I was able to hack up a function that automatically compiles using rmarkdown, launching an R process if necessary. All of this made possible using [an informative discussion](#) found on the ess help mailing list. You'll also need [polymode](#) for emacs.

Here's the function (perhaps place it in your `.emacs`):

```
(defun ess-rmarkdown ()
  "Compile R markdown (.Rmd). Should work for any output type."
  (interactive)
  ; Check if attached R-session
  (condition-case nil
    (ess-get-process)
    (error
     (ess-switch-process)))
  (let* ((rmd-buf (current-buffer)))
    (save-excursion
      (let* ((sprocess (ess-get-process ess-current-process-name))
             (sbuffer (process-buffer sprocess))
             (buf-coding (symbol-name buffer-file-coding-system))
             (R-cmd
              (format "library(rmarkdown); rmarkdown::render(\"%s\")"
                      buffer-file-name)))
        (message "Running rmarkdown on %s" buffer-file-name)
        (ess-execute R-cmd 'buffer nil nil)
        (switch-to-buffer rmd-buf)
        (ess-show-buffer (buffer-name sbuffer) nil))))))
```

You might also want to add a keyboard shortcut. I use M-n s:

```
(define-key polymode-mode-map "\M-ns" 'ess-rmarkdown)
```

If the function doesn't work, you can always revert back to the more robust implementation in RStudio.