  
Richard Layton  
[graphdoctor@gmail.com](mailto:graphdoctor@gmail.com)

July 14, 2014

To: My R friends.  
Re: Comparing methods of creating docx files.

The usual report I create for my collaborators includes data tables, graphs, and short discussions. In this test document, I've included some of each, formatted as closely as possible to my usual standards for document design.

# Creating a docx file using R Markdown v2 in RStudio

I first created an R file to manipulate data and create a graph. With the analysis complete, I opened a new .Rmd file and copied all the R code into a code chunk. (I usually avoid copy and paste but I wanted this test document to be self-contained.)

Then I added a markdown header and added text. With the document fairly complete, I saved the output docx file as a template, then edited the styles in the template to obtain the document design elements I wanted.

Some thoughts:

* Spell-check works fine.
* I like how *Knit Word* updates the docx file even when it's open. Eliminates many mouse clicks.
* Lists in the .Rmd file are easy to make and easy to read.
* Paragraphs in the .Rmd file require no markup syntax.
* I couldn't figure out how to place a logo or page numbers in the docx header.
* Emphasizing individual words in *italics* or **bold** is much simpler in **rmarkdown** than it is in **ReporteRs**.
* Because rmarkdown parses LaTeX math expressions, it was easy to create an em-dash in a paragraph.
* Formatting a table was easy in **rmarkdown**, but limited compared to **ReporteRs**.
* On the other hand, figuring out table captions in **rmarkdown** took me quite some time, unlike figure captions. (Figure captions are built-in to the knitr markdown code chunk header.) My final approach was to use inline R markup and an R function that creates consecutive table numbers.
* Control commands such as *Insert page break* are unavailable. I used HTML "br" tags to manually create space. Clunky, but works.

# Data source

The *VADeaths* data are furnished in the base R install.

Learning to format tables was easyknitr's *kable()* function has reasonable defaults and only a few arguments. The downside is that I could not figure out how to add horizontal rules.

Table 1. Death rate data (per 1000), Virginia 1940

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Age group | Rural male | Rural female | Urban male | Urban female |
| 50-54 | 12 | 9 | 15 | 8 |
| 55-59 | 18 | 12 | 24 | 14 |
| 60-64 | 27 | 20 | 37 | 19 |
| 65-69 | 41 | 31 | 55 | 35 |
| 70-74 | 66 | 54 | 71 | 50 |

# Data display

The figure is drawn using the lattice package. The size of the figure is controlled using options in the knitr code chunk.

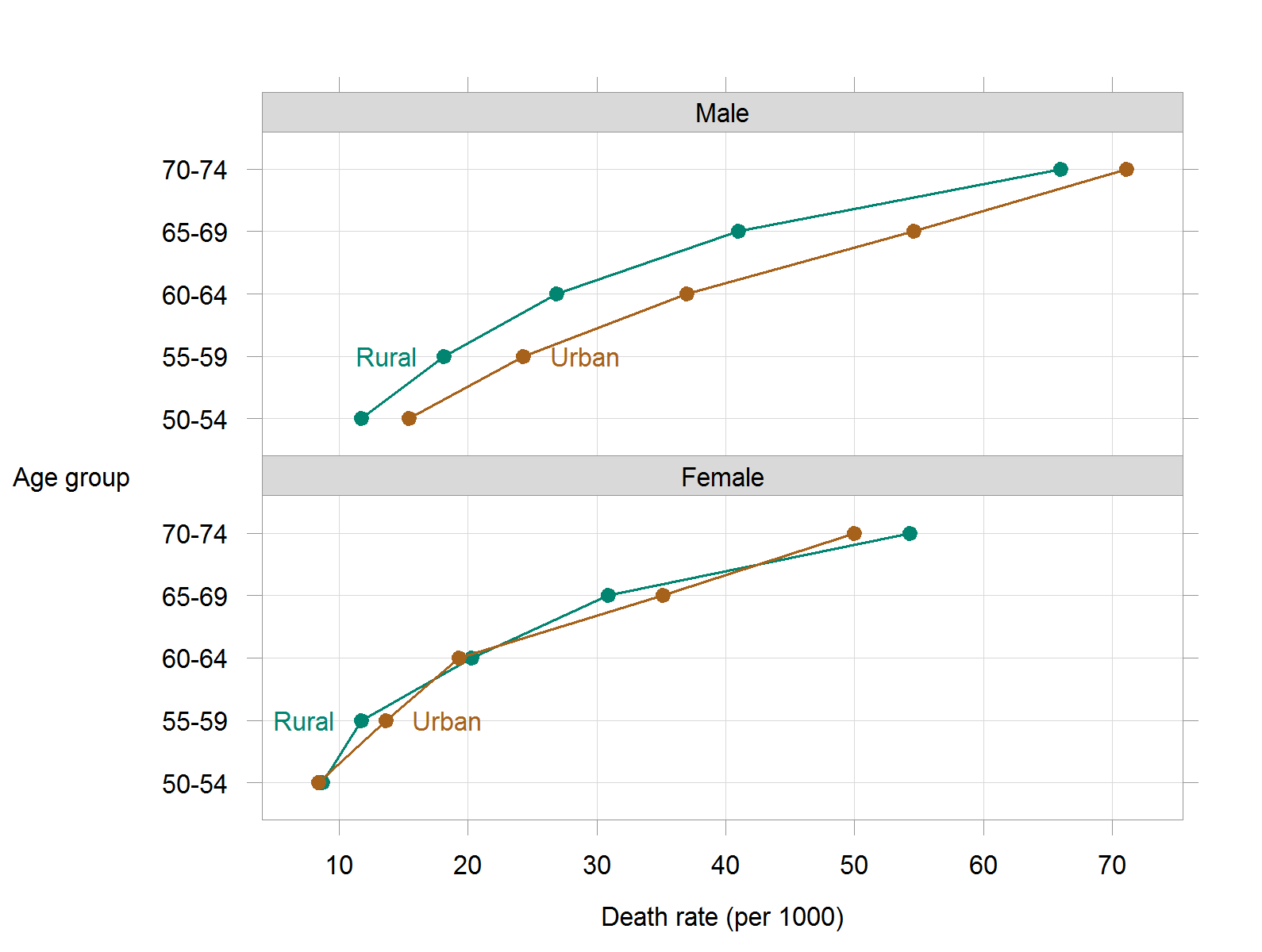


Figure 1. Comparing male and female death rates in rural and urban Virginia in 1940.

Rates are nearly identical for rural and urban females, with a systematic increase among rural males and a further increase for urban males.