

# Using R and RStudio with Lertap 5

Larry R Nelson<sup>1</sup>  
Curtin University, Western Australia  
Burapha University, Thailand

Document date: 31 October 2020

To see if there might be a newer version of this document, [click here](#).

R is both a programming language and a potent software environment for use in data analysis. The main website is [here](#). [RStudio](#) is a companion to R, providing a special IDE<sup>2</sup> used to write and run scripts written in "R Markdown". Such scripts have file extensions of "Rmd".

Lertap 5 is an Excel-based data analysis package used for processing responses to test and survey questions. Use [this link](#) for an introduction.

Lertap 5 supports an extensive battery of practical item and test analysis tools, but there are indeed times when it comes up a bit short in some areas of teaching and research. So it is that, over the years, Lertap 5 has been trained to "talk" to other data analysis programs.

Among these are programs that cater to users of "IRT", item response theory. These include, for example, [Bilog-MG](#), [Xcalibre](#), [FIRT](#), [SAS](#), and an R package called [TAM](#). Lertap 5 can also communicate with the R [psych](#) package for those users looking for estimates of [coefficient omega](#), a reliability index for tests and surveys.

The purposes of this document are (1) to show how selected R packages are installed, and (2), to then discuss and describe how Lertap 5's special R and RStudio scripts are applied.

**Note:** the installation of R and RStudio is discussed in [another document](#). The presentation below assumes that the R and RStudio apps have already been installed; the discussion to follow will cover the installation of two selected R packages, and the use of special "scripts" produced by Lertap 5.

## Installing R packages

While there is truly a world of things we can do with R, in this document I will limit the discussion to the use of two R "[packages](#)"

The work I've done thus far with Lertap 5 and R is based largely on the "[psych](#)" and "[TAM](#)" packages. I will go through the steps needed to install them.

I elected to install "psych" first. I started R and typed the text in red in the Console window (see Figure 1):

---

<sup>1</sup> Comments / questions may be sent to [l.nelson@curtin.edu.au](mailto:l.nelson@curtin.edu.au)

<sup>2</sup> Integrated Development Environment

```
'citation()' on how to cite R or R packages in publica
Type 'demo()' for some demos, 'help()' for on-line hel
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> install.packages("psych")|
```

Figure 1

I was asked to pick a "mirror". I took one of the Australian mirrors as for most of the year I live in Western Australia.

```
> install.packages("psych")
--- Please select a CRAN mirror for use in this session ---
also installing the dependency 'mnormt'

trying URL 'https://cran.curtin.edu.au/bin/windows/contrib/3.4$
Content type 'application/zip' length 102302 bytes (99 KB)
downloaded 99 KB

trying URL 'https://cran.curtin.edu.au/bin/windows/contrib/3.4$
Content type 'application/zip' length 5342279 bytes (5.1 MB)
downloaded 5.1 MB

package 'mnormt' successfully unpacked and MD5 sums checked
package 'psych' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:\Users\LP\AppData\Local\Temp\Rtmp63dpIw\downloaded_p$
> install.packages("TAM")|
```

Figure 2

I then installed the TAM package (as seen above in red letters, Figure 2). The following messages appeared in the Console:

```
R Console
trying URL 'https://cran.curtin.edu.au/bin/windows/contrib/3.4$
Content type 'application/zip' length 2192930 bytes (2.1 MB)
downloaded 2.1 MB

trying URL 'https://cran.curtin.edu.au/bin/windows/contrib/3.4$
Content type 'application/zip' length 1804878 bytes (1.7 MB)
downloaded 1.7 MB

package 'polycor' successfully unpacked and MD5 sums checked
package 'CDM' successfully unpacked and MD5 sums checked
package 'coda' successfully unpacked and MD5 sums checked
package 'mvtnorm' successfully unpacked and MD5 sums checked
package 'Rcpp' successfully unpacked and MD5 sums checked
package 'sfsmisc' successfully unpacked and MD5 sums checked
package 'RcppArmadillo' successfully unpacked and MD5 sums che$
package 'TAM' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:\Users\LP\AppData\Local\Temp\Rtmp63dpIw\downloaded_p$
> |
```

Figure 3

This completed the installation of the two packages I wanted to install, and I thought I'd be ready to give things a go. I closed R before moving on.

### A test drive using RStudio

Lertap 5 readily computes a reliability estimate known as "alpha", also frequently called "Cronbach's alpha". Alpha is by far the most commonly-encountered reliability estimate, but it is not the only one. Coefficient "omega" is another.

Lertap 5 has a "special macro" designed to download little R and RStudio files for use with the psych and TAM packages just mentioned above. Two of them are used to get coefficient omega.

The special macro is [described here](#).

I will use the "FIMS" dataset in order to demonstrate how the macro is used, the files it downloads, and the use of those files in RStudio.

Figure 4 shows how I started. I had the FIMS.xlsx file in a working folder. Figure 5 displays the contents of the folder after I had used the macro.

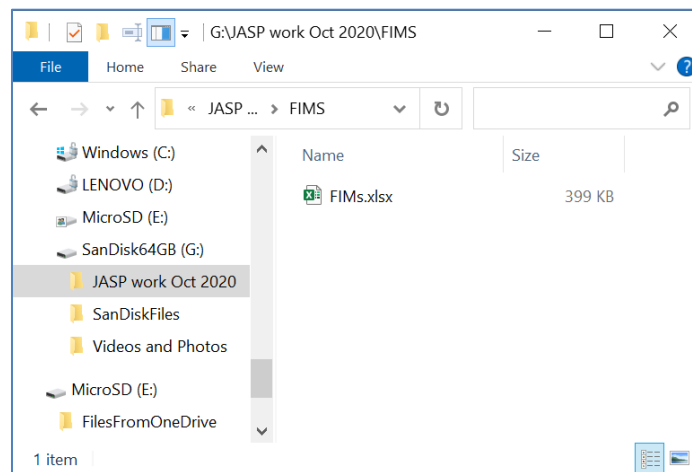


Figure 4

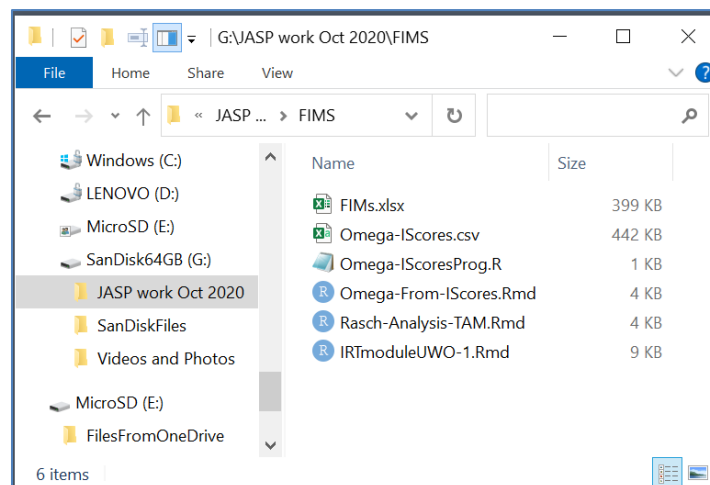


Figure 5

The macro added five files to the folder. Three of the new files have an extension of "Rmd". Each of these contains a script ready to use with RStudio.

When I right-click on the "Omega-From-IScores.Rmd" file, Windows File Explorer indicates that my computer would open the file with three apps – Google Chrome, Notepad, and RStudio.

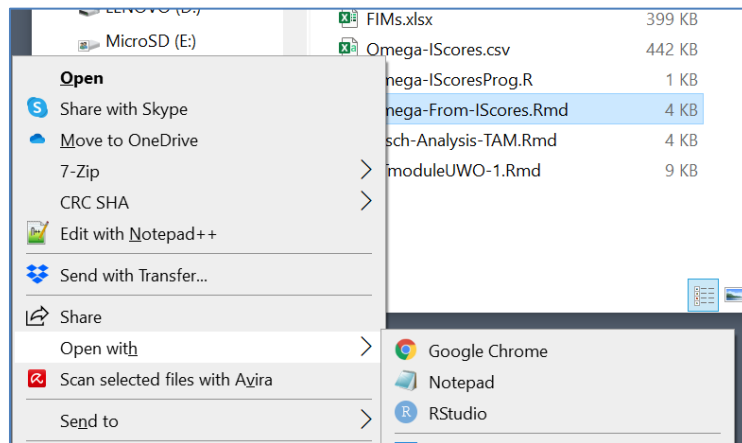


Figure 6

I elect to open it with RStudio. See Figure 7. I have snipped the "Knit" option in Figure 7 and displayed a closeup of it in Figure 8.

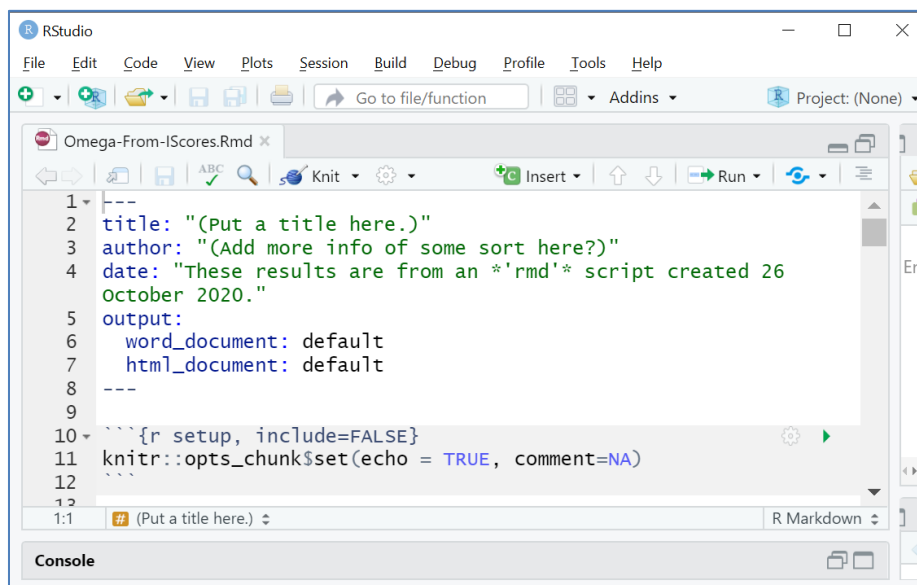


Figure 7

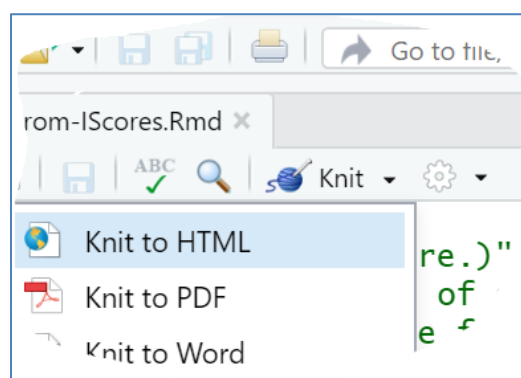


Figure 8

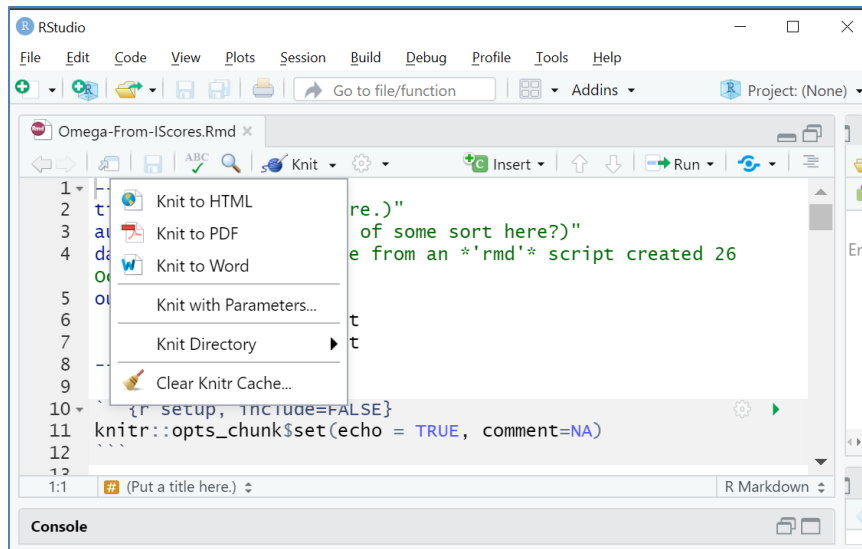


Figure 9

Next, I click on the "Knit to Word" option. RStudio will then silently do "its thing". There is no apparent activity for about 20 seconds, but then, to my delight, Microsoft Word opens and displays the report generated by Knit.

The report has been saved in my working folder as "Omega-From-IScores.docx" (Figure 10).

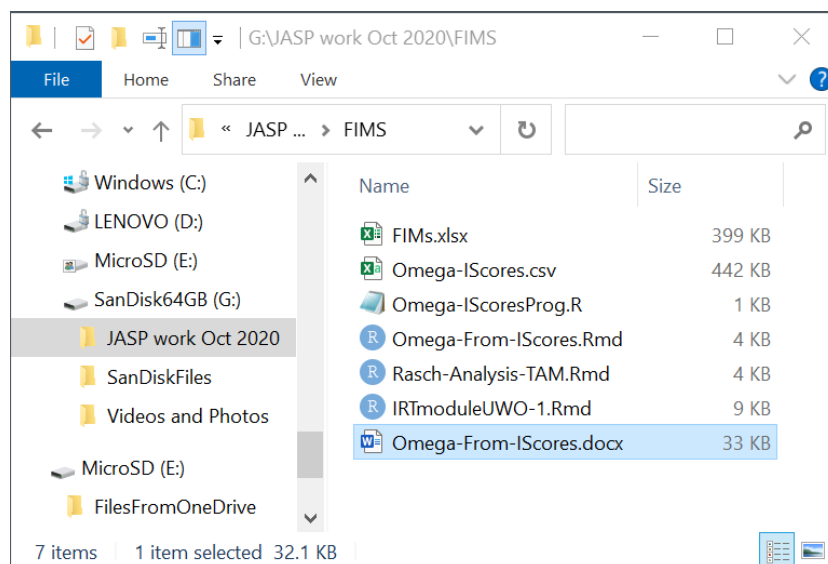


Figure 10

Parts of the report are seen below in Figures 11 and 12.

### Omega results from the R 'psych' package

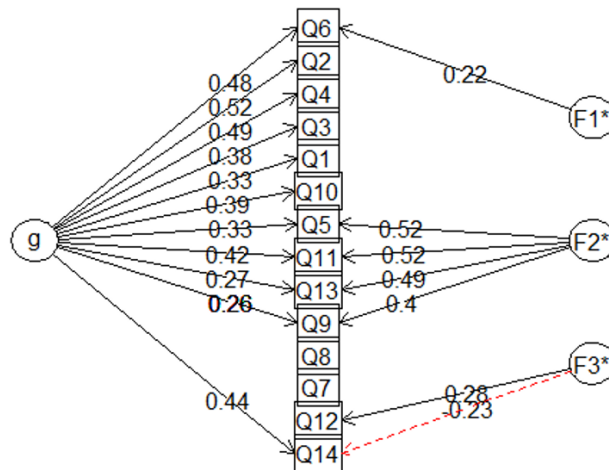


Figure 11

```
Alpha: 0.697
G.6: 0.7
Omega Hierarchical: 0.533
Omega H asymptotic: 0.732
Omega Total 0.728
```

```
With eigenvalues of:
  g  F1*  F2*  F3*
1.80 0.19 1.01 0.19
```

Figure 12

### A test drive using R

The test drive above was based on the use of an RStudio script found in the "Omega-From-IScores.Rmd" file.

Another of the files seen in Figure 5 is "Omega-IScoresProg.R". This file contains a script to use with R.

To demonstrate its use I will begin by opening the R "app" on my computer. See Figure 13 below.

I use the File menu to open "Omega-IScoresProg.R". See Figure 14 please.

My next step is to go to the Edit menu where I select the "Run all" option. It should read and run all of steps in the script.

Results are indicated in Figure 15 – running the script has produced two new files in my working folder. Figures 16 and 17 display results.

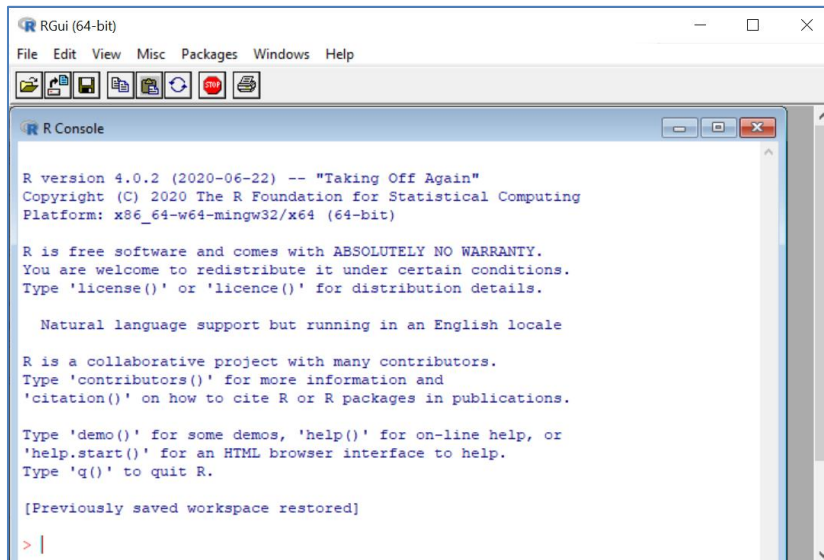


Figure 13

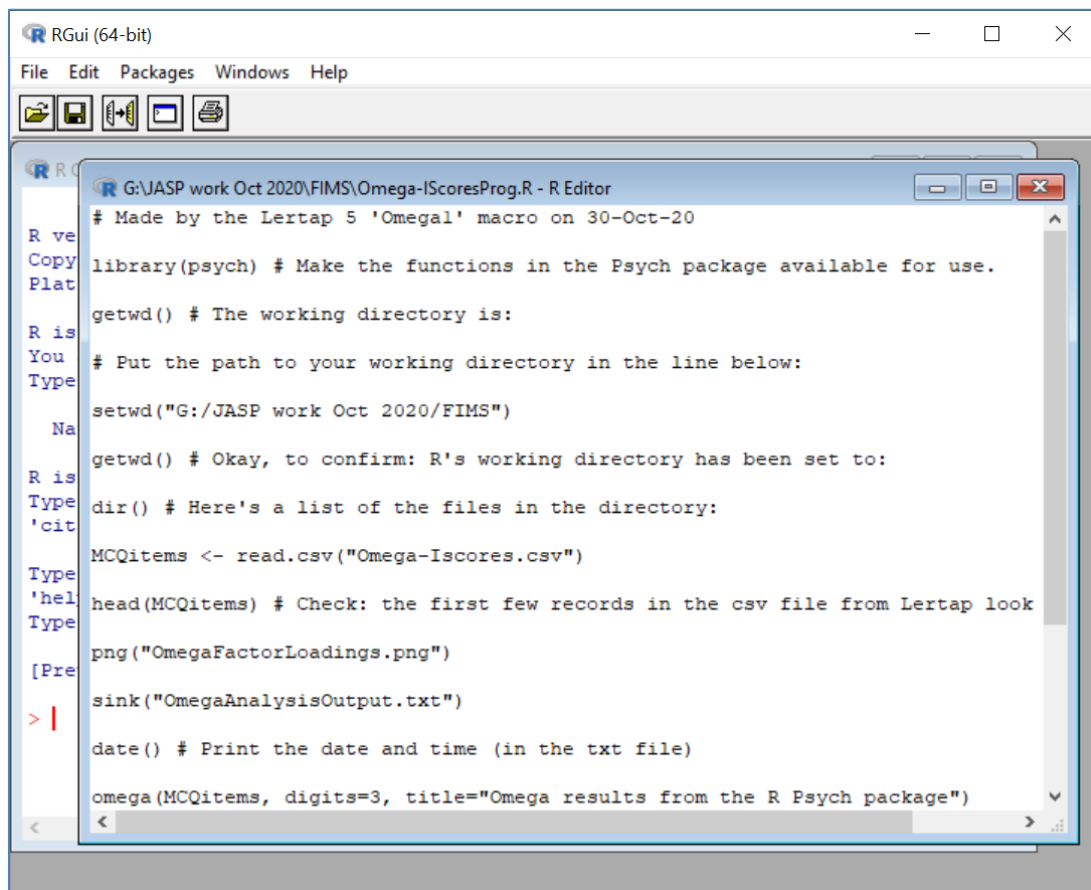


Figure 14

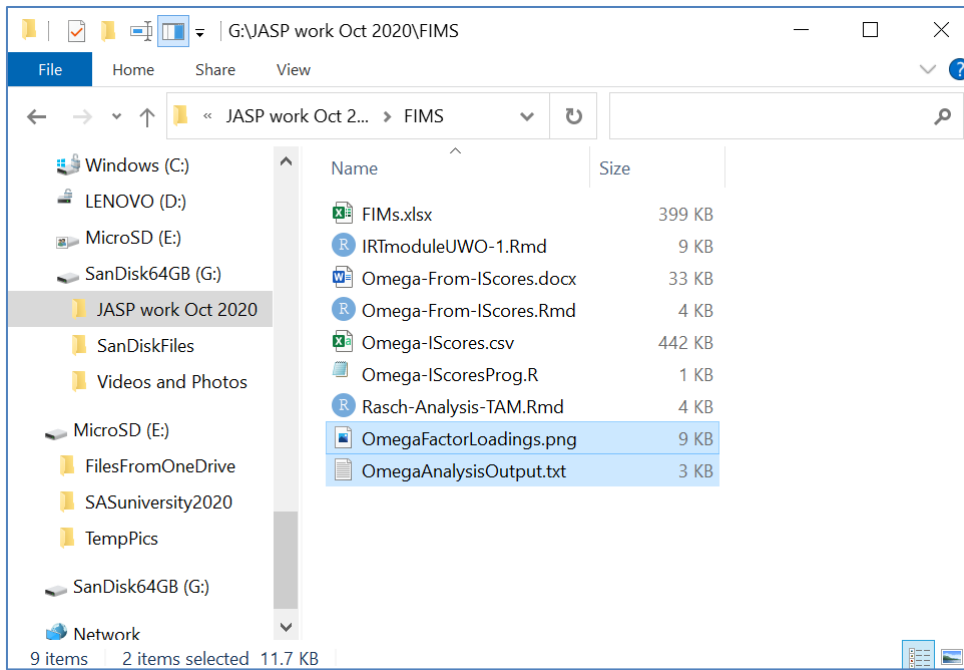


Figure 15

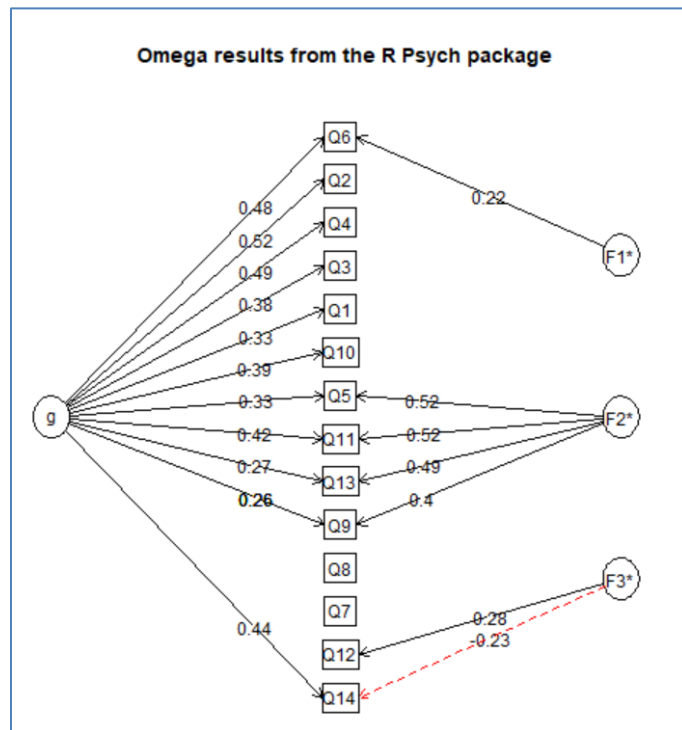


Figure 16 (png from R)

```

Alpha:                0.7
G.6:                  0.7
Omega Hierarchical:   0.53
Omega H asymptotic:   0.73 |
Omega Total           0.73

Schmid Leiman Factor loadings greater than 0.2
      g   F1*   F2*   F3*   h2   u2   p2
Q1  0.33                0.13 0.87 0.86
Q2  0.52                0.31 0.69 0.86
Q3  0.38                0.18 0.82 0.83
Q4  0.49                0.28 0.72 0.86
Q5  0.33                0.52   0.38 0.62 0.29
Q6  0.48  0.22                0.29 0.71 0.79
Q7                0.05 0.95 0.47
Q8                0.05 0.95 0.35
Q9  0.26                0.40   0.23 0.77 0.30
Q10 0.39                0.18 0.82 0.86
Q11 0.42                0.52   0.44 0.56 0.39
Q12                0.28 0.09 0.91 0.00
Q13 0.27                0.49   0.31 0.69 0.23
Q14 0.44                -0.23 0.27 0.73 0.71

With eigenvalues of:
      g   F1*   F2*   F3*
1.80 0.19 1.01 0.19

```

Figure 17 (txt from R)

## RStudio or just R?

RStudio is the method I myself prefer. I find it easier to use, and love having a single, well-formatted report, one I can modify as I would any Word document.

If it seems that R has produced more output that would be because I did not capture all of the output which resulted from using RStudio – it includes the factor loadings seen in Figure 17 (and much more – see [this example](#)).

The advantage to using just R, not RStudio, might have to do simply with not having to install RStudio. I cannot think of any other reason.

---

## Agradecimiento

*Thanks to Prof. Margaret Wu for much assistance, especially with regard to R Markdown. See some of her work [here](#); also see the [TAM tutorials](#) she has been working on.*