

# Workshop: Basic R and More, Part 4 of 4

Wake Forest School of Medicine  
Winston-Salem, NC

Philip Turk, PhD<sup>1</sup>

Director, Biostatistics and Data Science  
Center for Outcomes Research and Evaluation (CORE)  
Atrium Health; Charlotte, NC

07/09/2021



---

<sup>1</sup><[Philip.Turk@atriumhealth.org](mailto:Philip.Turk@atriumhealth.org)>

- I always work with script (.R) files instead of typing directly into the console. If you have a data analysis of any substance, it is virtually impossible to write all your R code correctly the first time **and** remember all that you did later on if the need arises.
- Writing a script file fully documents how you did your analysis. Hence, having a script makes it easy to re-run an analysis after a change in the data (additional data values, transformed data, or removal of outliers) or if a journal referee wants revisions.
- Let's see an example. Open your browser, go to [<https://github.com/philturk/R\\_Wshop>](https://github.com/philturk/R_Wshop) and click on Ex01.R. Keep this window open. (Note that I have also sent this to you via email.)

- It often makes your script more readable if you break a single command up into multiple lines. R will disregard all whitespace (including line breaks) so you can safely spread your command over multiple lines.
- It's also useful to leave comments in the script for things such as explaining a tricky step, who wrote the code and when, or why you chose a particular name for a variable. So 'comment' your R code and be kind to your future self!

# R Scripts

- One way to create a new .R script in RStudio is to go to File, New File, R Script. This opens a blank script in the Source window where you can type commands and functions.
- Copy-and-paste or type the R code from Ex01.R in the Source window and then you can execute the code as follows:
  - Click the line of code you want to run, and then press Cmd + Return or click the Run button (Windows users, try Ctrl + Enter).
  - Highlight the block of code you want to run, and then again press Cmd + Return or click the Run button.
  - Run the entire script by clicking anywhere in the source editor and then pressing Cmd + Shift + Return (Windows users, try Ctrl + Alt + R)



- Working with R scripts is an acceptable way of documenting what you did, but the script file itself doesn't contain the actual results of commands that were run, nor does it show you the plots.
- Also, anytime I want to comment on some output, it needs to be offset with the commenting character #.
- It would be nice to have all the commands, the results, and comments merged into one document. This is what the R Markdown file does for us.

# R Markdown

- It is very inefficient to copy output and save figures from RStudio and paste them into an MS Word document.
- An *R Markdown* document (or .Rmd file) is a way of making, or “knitting”, a completely self-contained reproducible research document. It has the following:
  - your executable R code in “code chunks”
  - your R output where you want it
  - your text where you want it
- Thus, your entire data analysis and report is forever etched in stone. If you make a mistake, simply go back to the .Rmd file, make the correction, re-run the file, and the entire report is reproduced and updated. (I have actually created the slides for this workshop using RMarkdown.)
- R Markdown is also great for ‘automating’ the same report with different values for various important inputs.

- As a bit of history, R Markdown is an implementation of so-called *Markdown* syntax that cooperates with R. Markdown started as an easy way to write web pages and give instructions for how to do typesetting with a “markup language” called LaTeX.
  - <<https://rmarkdown.rstudio.com>>
  - <<https://bookdown.org/yihui/rmarkdown/>>
- You need the `rmarkdown` package, but you don't need to explicitly install it or load it, as RStudio automatically does both when needed.
- There is another way to do reproducible research (Sweave), but R Markdown is easier to learn and use, and much more popular.

- One way to create a new .Rmd file in RStudio is to go to File, New File, R Markdown. A menu will appear asking you for a title, author, and output format (HTML, PDF, or Word). In order to create a PDF, you'll need to have LaTeX installed, but the default HTML output should work fine.
  - The `tinytex` package in R provides a lightweight and easily maintained version of LaTeX. This might be enough to get you by if you want PDF output. We will not cover this here.
- In my own experience, MS Word output works well but not as good as the other two formats. Be prepared to have a few more challenges along the way.



# R Markdown





Let's take a look at the basic structure of an R Markdown file by carefully inspecting the .Rmd file you just created. It has the following components





- 1 A YAML header (everything between and including the two `---` lines)
  - The one given to you by default is very basic. There are many options available. We will add a few things to this momentarily.
- 2 R code chunks
  - Each chunk forms a sandwich of R code wrapped by a pair of triple backticks. At a minimum, the first group of backticks must be followed by `{r}`.





```
```{r}
your lines of R code go here
```
```

- 3 Text, equations, etc., with simple formatting
  - R Markdown is very good at weaving these in and out of your R code and output.



- To compile the .Rmd file into an HTML file, near the upper left-hand corner of the Source window in which you are doing your editing, there is a button entitled Knit (it has a blue ball of yarn on it). Click on it to Save File and generate the HTML file  
- Go to Help, Markdown Quick Reference. This opens a short guide in the Help window that shows you how to do various forms of text formatting  
  - For example, if you cursor down, you will see a brief explanation on how to place images into your .Rmd file (on the web or local files in the same directory).

- Let's try out some modifications. Find some whitespace in your .Rmd file and type the following.
  - Type `*Hello*` (will give  $\rightarrow$  *Hello*)
  - Type `**Hello**` (will give  $\rightarrow$  **Hello**)


When you are done, knit the document again  

- For code chunks, you can insert them manually *or* hit the Insert button icon in the editor toolbar above the Source window  
- To test out the code chunk(s), hit the green arrow on the right-hand side of the code chunk *or* use the Run button icon in the editor toolbar above the Source window  

# R Markdown

- The chunk header structure is ````${r optional chunk name or number, comma separated options}`
- One good reason to give your code chunks numbers is that it makes it easier to debug your program.
- There are about 60 *options* that you can use to customize your code chunks. In addition to showing the code and output, you can do other things. For example, as we will soon see, using `echo = FALSE` shows only the output in the finished file and not the R code.
- To see/modify the code chunk options, hit the gear icon on the right-hand side of the code chunk.
  - Click on the Output drop-down menu and notice the choices.
  - Click on **Chunk options** and notice what happens.
  - For the code chunk entitled `pressure`, erase `echo = FALSE`, knit the document again, and notice what happens  



# R Markdown



- One nice feature of R Markdown is that it gives you the ability to insert complex mathematical expressions using the powerful typesetting system LaTeX. Since this is specific and complex, I only touch upon it here.
- In your R Markdown document, you can include LaTeX code by enclosing it with dollar signs. LaTeX is not based on “what you see is what you get”; it’s a markup language. So you might type `$\alpha = 0.05$` in your text, but after you knit the document, you’ll see  $\alpha = 0.05$  
- If you want a mathematical equation to be centered on its own line, then enclose it with double dollar signs. So you might type `$$\bar{X} = \frac{\sum X_i}{n}$$` in your text, but after you knit the document, you’ll see:

$$\bar{X} = \frac{\sum X_i}{n}$$

- A good LaTeX resource is at [<https://en.wikibooks.org/wiki/LaTeX>](https://en.wikibooks.org/wiki/LaTeX)

# R Markdown

- Rather than accepting R's default output from a code chunk, suppose we want the output to be in the form of a nicer table. Here are your options:
  - Make a table by hand using LaTeX.
  - R Markdown has three basic options: simple, grid, and pipe tables. Of the three, the pipe tables seem to be the easiest to set up. Truth be told, tables are not R Markdown's strong suit. The best documentation seems to be here: <<https://pandoc.org/MANUAL.html#tables>>.
  - There are a couple of different packages that convert a data frame to a beautified table. The most basic approach uses the `kable()` function from the `knitr` package. More complicated approaches use the `pander` or `kableExtra` packages. The later is my favorite.
  - For example, install the `pander` package. For the code chunk entitled `cars`, add the line `library(pander)`, and then `pander(summary(cars), style = "rmarkdown")`, knit the document again, and notice what happens  

- You can control many other document settings by tweaking the parameters of the YAML header.
- Go to <[https://github.com/philturk/R\\_Wshop](https://github.com/philturk/R_Wshop)> and click on the big green button that says Clone or download, then click on Download ZIP. On my Mac, this saves the entire repository (“repo”) as a R\_Wshop-master folder to my Downloads. Click on Ex01.Rmd; it should open up in RStudio. This file is almost identical to your .Rmd file with a few exceptions. (Note that I have also sent this to you via email.)  
- Examine my YAML header. Modify the author, date, and subtitle fields to your own personal taste.
- Knit the document again, and notice what happens.





# R Markdown





- A huge timesaver is the ability to easily create a bibliography using R Markdown. I will show you one way that imports references from your own master repository.
- To use this feature, specify a bibliography file using the `bibliography` field in your YAML header. The field should contain a path from the directory that contains your `.Rmd` file to the file that contains the bibliography file.
- For example, in your `R_Wshop-master` folder, I have created a file called `citations.bib`. For now, determine its path (e.g., `/Users/pturk/R/Workshop/citations.bib`). Copy this path. Go back to the `Ex01.Rmd` file and overwrite the path currently in the `bibliography` field. Finally, uncomment the `bibliography` field, i.e., remove the `#`. Knit the document again, and notice what happens





# R Markdown

- You can use many common bibliography formats but my favorite is BibTeX. If you open up `citations.bib` in a text editor, you will see the references all have a similar construction  
- Almost every reference has a BibTeX reference. University libraries, Google Scholar, literature databases, etc., have them as 'ready-to-go' downloadable files that you then just copy-and-paste into your `.bib` file.
- Advantages:
  - You need to copy-and-paste or type each reference only once.
  - The style of all your citations in any given document will be consistent.
  - Your bibliography is automatically generated.
  - Your bibliography uses only what you cite.
  - Your bibliography can be customized to journal style (e.g., APA).
- My `.bib` file has hundreds and hundreds of references and lives in *one location* on my computer. Zotero is a great open-source reference management software *and* has a very nice search tool at <https://zbib.org>  

- To create a bibliography, notice how I typed `## References` at the bottom of `Ex01.Rmd`. You could also use `## Bibliography`. They are two keywords  
- Observe how I created the citations in `Ex01.Rmd`. Other options exist but we do not discuss them here  
  - Citations go inside square brackets. If there are multiple citations, then they are separated by semicolons. Each citation must be denoted by '@' + the citation identifier from the BibTeX reference in your `.bib` file. Use the syntax `[@name of BibTeX reference]`.
  - For an in-text citation, remove the square brackets and use `@name of BibTeX reference`.



Thank you for your attention and time!