Tutorial: R Markdown

Your name

2024-01-14

Learning objectives

- Become familiar with the R Markdown syntax and code chunk rules
- Understand how to include figures and tables in your Markdown reports
- Create R Markdown files and export them to HTML and Microsoft Word files

Why R Markdown?

All lab assignments will be completed within an R Markdown document.

Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. It presents your code alongside its output (graphs, tables, etc.) with conventional text to explain it, a bit like a notebook. This can make it easier for the instructors to present instructions, code, and other content for each lab. For students, R Markdown can makes it easier to integrate R code with their written explanations, such as answers to questions.

This file provides an overview of R Markdown. If you're reading a knitted version of this document (e.g., under the "Jupyter Hub, R, and RStudio - Resources" module in Canvas), the `.Rmd` file that was used to create it was provided to you in Week 2 of class (`Intro_to_RMarkdown.Rmd`).

Note: Some of the below content was borrowed from a tutorial about R Markdown at Coding Club.

The YAML header

At the top of an R Markdown document, there is a YAML header section enclosed by `---`. By default, it includes a title, author, date and the file type you want to output to. For this class, you will produce HTML (`.html`) and MS Word (`.docx`) documents. The instructors may return the Word document back to you with comments. Make sure to replace "Your name" with your actual name. The "date" will be automatically generated using the `Sys.Date()` function, which returns the current date.

Knitting a document

When you click the **Knit** button (ball of blue yarn with a needle through it), a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. By default, I set up R Markdown files to knit to an HTML file; however, you can knit other formats by clicking the drop-down menu next to the knit button. **Please remember to knit your assignments to MS Word files**.

Code chunks

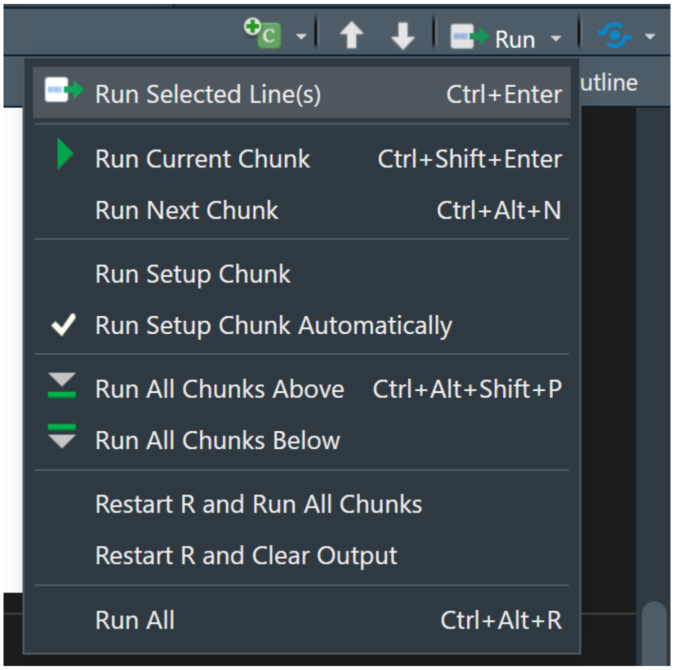
The first code chunk in R Markdown may contain R packages that are needed to run your code. For this file, we use the `here()` and `knitr()` packages.

You can embed an R code chunk like this:

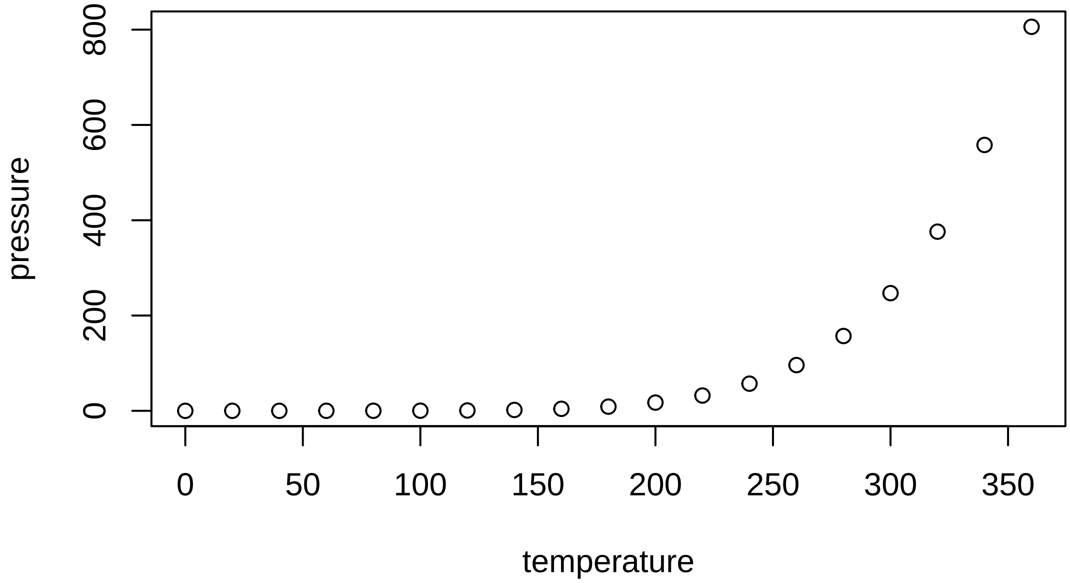
```r
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00  
```

When working through your lab assignment, you can run the code for an individual chunk by clicking the green "play button" triangle in the top-right corner of the chunk. To run multiple chunks, click on the `Run` button in the top-right corner of the "Source" pane in RStudio, and select your preference (e.g., restart R and run all chunks, or run all chunks above). You can run individual lines of code within chunks by selecting `Run selected lines` or using the keyboard shortcut (on Windows) `CTRL + SHIFT + ENTER`. See the screenshot below.


Run
utline
Run Selected Line(s) Ctrl+Enter
Run Current Chunk Ctrl+Shift+Enter
Run Next Chunk Ctrl+Alt+N
Run Setup Chunk
✓ Run Setup Chunk Automatically
Run All Chunks Above Ctrl+Alt+Shift+P
Run All Chunks Below
Restart R and Run All Chunks
Restart R and Clear Output
Run All Ctrl+Alt+R

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

You might want to create an object, but not include both the code and the output in the final `.html` file. To do this you can use, include = FALSE. Be aware though, when making reproducible research it's often not a good idea to completely hide some part of your analysis.

To insert an external figure, you can use the `include_graphics` function in the `knitr` R package, as shown above in the screenshot of the run options.

There are several options for inserting tables.

One is the `kable()` function from the `knitr` package. Also checkout the `gt()` function in the `gt` package and the `pander()` function in the `pander` package.

```
kable(pressure)
```

temperature	pressure
0	0.0002
20	0.0012
40	0.0060
60	0.0300
80	0.0900
100	0.2700
120	0.7500
140	1.8500
160	4.2000
180	8.8000
200	17.3000
220	32.1000

temperature	pressure
240	57.0000
260	96.0000
280	157.0000
300	247.0000
320	376.0000
340	558.0000
360	806.0000

You can manually create small tables using R Markdown syntax. For example:

Plant	Temp.	Growth
A	20	0.65
B	20	0.95
C	20	0.15

There are several other code chunk instructions that you could learn about (see **R Markdown Resources** below).

Formatting text

Once you knit your document, the output will display text formatted according to the following simple rules.

Italic

Italic

Bold

Bold

This is `code` in text

This is code in text

Header 1

Header 1 ## Header 2

Header 2 Note that when a # symbol is placed inside a code chunk it acts as a normal R comment, but when placed in text it controls the header size.

- Unordered list item

Unordered list item 1. Ordered list item

Ordered list item Link

Link

$A = \pi \times r^2$

Rendered equation example

The $ symbols tells R markdown to use LaTeX equation syntax.

Practice

To practice this, try writing some formatted text in your `.Rmd` document, add some additional code, and produce an `.html` and `.docx` file using the `Knit` button.

R Markdown Resources

Here are some tutorials and other potentially helpful documents. These same resources are also listed in the "R Markdown Resources" page on Canvas.

Official site
- Website for R Markdown

Books (online and free)
- R Markdown: The Definitive Guide
- R Markdown Cookbook
- R Markdown | R for Data Science

Tutorials
- Getting started with R Markdown (Coding Club)
- R Markdown quick tour (RStudio)
- Getting started with R Markdown - Guide and Cheatsheet (Dataquest)
- Getting started with R Markdown (Data Carpentry)