

# Authoring Documents in RStudio

khailper

February 16, 2017

# Basics

- ▶ R Markdown
  - ▶ R version of Markdown, a markup language
- ▶ YAML
  - ▶ Header (RStudio builds this for you)
  - ▶ Tells computer what kind of doc your building
- ▶ knitr
  - ▶ Package that integrates the above (along with other components like LaTeX) to actually build the document
- ▶ RStudio has a couple of documents that flesh out most of the material in this presentation
  - ▶ cheatsheet
  - ▶ reference document
  - ▶ Available in RStudio (Help -> Cheatsheets)
  - ▶ Even more resources at <http://rmarkdown.rstudio.com/>
  - ▶ Good blog post on integrating R Markdown into your workflow

# Requirements

- ▶ Rstudio
- ▶ Packages
  - ▶ markdown
  - ▶ rmarkdown
  - ▶ yaml
- ▶ Additonal Programs
  - ▶ TeX for PDF documents
  - ▶ MS Word or Open/Libre Office for Word documents

# Creating a Document

- ▶ File -> New File -> R Markdown
- ▶ Select document type you want to build
  - ▶ this presentation was knit as a Slidy presentation
- ▶ RStudio automatically creates the appropriate YAML header and some sample code
- ▶ When you want to create the document, click “Knit”

# R Markdown Basics

- ▶ Starting line with #s followed by space for headers (more #s = sub-headers)
- ▶ Format is done by surrounding text with symbols (\* or \_ for italics, \*\* or \_\_ for bold, for ^ for superscripts)
  - ▶ Equations are similar, surrounded by \$
- ▶ These bullets were created with \*
  - ▶ The + symbol was used for this line (preceded by four spaces)
  - ▶ For ordered lists, use [number].
- ▶ For PDF/DOC files page breaks can be created by \newpage or \pagebreak (on their own line)
- ▶ for slides, breaks by either (sub)headers or \*\*\* on one line

# Math Syntax Basics

- ▶ R Markdown uses LaTeX for handling equations
- ▶ Superscripts are done with `^`; subscripts with `_`
  - ▶ for both, `{ }` helps with longer strings
- ▶ Most other commands start with `\`
- ▶ for fractions, use `\frac{a}{b}` gives  $\frac{a}{b}$
- ▶ Sigma notation is done with `\sum_{n=1}^N` to get  $\sum_{n=1}^N$
- ▶ `\sqrt{27}` yields  $\sqrt{27}$
- ▶ More basics:
  - ▶ <http://www.calvin.edu/~rpruim/courses/m343/F12/RStudio/LatexExamples.html>
  - ▶ [https://rstudio-pubs-static.s3.amazonaws.com/18858\\_0c289c260a574ea08c0f10b944abc883.html](https://rstudio-pubs-static.s3.amazonaws.com/18858_0c289c260a574ea08c0f10b944abc883.html)
- ▶ For something more comprehensive:  
<https://www.giss.nasa.gov/tools/latex/ltx-2.html>

## R Code in documents

- ▶ For inline code, start code with `'r` and end with `'`.
  - ▶ e.g. to get 1.7320508, use `'r sqrt(3)'`.
- ▶ For code chunks, first line is `"'{r nameofchunk}`
  - ▶ code chunks don't need names, but can be useful
- ▶ Last line of chunk is `"'`.
- ▶ Both should be their own line
- ▶ Shortcut for creating chunks: `Ctrl/Cmd + Alt + I`

# Code Chunk Parameters

- ▶ A variety of parameters can be included within `{}`
- ▶ Some common ones (and their default values)
  - ▶ `eval (TRUE)`: Should the code be run
  - ▶ `echo (TRUE)`: should the code be displayed in doc
- ▶ Can use `knitr::opts_chunk$set(Parameter = TRUE/FALSE)` to change defaults



# Caching

- ▶ By default, data from chunks not stored
- ▶ Can set `cache = TRUE` to store output
- ▶ This lasts until you change the chunk (however, doesn't check if earlier chunk changed)

## Quick Demo

# R Notebooks

- ▶ Introduced in RStudio v1.0
- ▶ Allows chunks to be run as you work, instead of having to re-knit the whole documents
- ▶ Also allows chunks written in other languages
- ▶ More resources at  
[http://rmarkdown.rstudio.com/r\\_notebooks.html](http://rmarkdown.rstudio.com/r_notebooks.html) and  
<https://blog.rstudio.org/2016/10/05/r-notebooks/>

# bookdown

- ▶ r package for authoring books w/R Markdown
- ▶ In addition to HTML and PDF, can have EPUB outputs.
- ▶ More resources at <https://bookdown.org/> and <https://bookdown.org/yihui/bookdown/>

# flexdashboards

- ▶ Package to allow you to build a dashboard of visualizations using R Markdown
- ▶ Incorporates htmlwidgets
- ▶ Designed to work well on mobile devices as well
- ▶ Can integrate with shiny for reactivity
- ▶ More resources at <http://rmarkdown.rstudio.com/flexdashboard/>
- ▶ Useful blog post on building flexdashboards

# ReporteRs

- ▶ Package for producing Word and PowerPoint documents
- ▶ Also contains functions for inserting code output into existing documents
- ▶ Package homepage:  
<https://davidgohel.github.io/ReporteRs/index.html>