# Dynamic Reports

## With RMarkdown

The R Bootcamp
Twitter: @therbootcamp

September 2017

# What is a dynamic report?

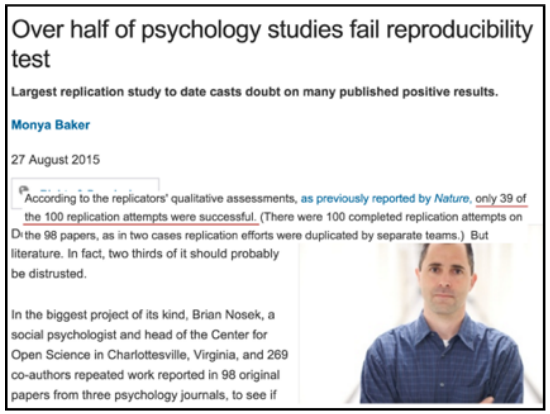
A dynamic report is a document that contains key elements which are automatically accessed and/or created by code:

- Access Data
- Wrangling / Statistics
- Plots
- Regular Text
- Formatting


http://rpubs.com/fabiorocha5150/decisiontreemodel

# Why are dynamic reports important?

1. Efficiency
2. Reproducibility
3. Scaleability



Over half of psychology studies fail reproducibility test

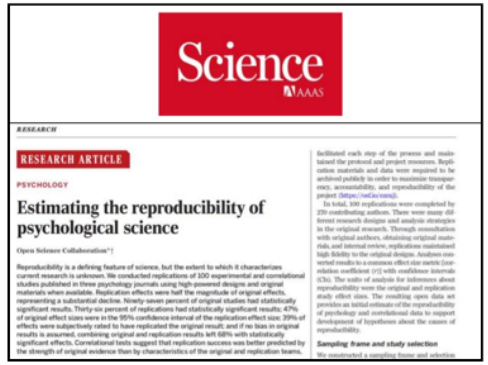
**Largest replication study to date casts doubt on many published positive results.**

Monya Baker

27 August 2015

According to the replicators' qualitative assessments, as previously reported by *Nature*, only 39 of the 100 replication attempts were successful. (There were 100 completed replication attempts on the 98 papers, as in two cases replication efforts were duplicated by separate teams.) But literature. In fact, two thirds of it should probably be distrusted.

In the biggest project of its kind, Brian Nosek, a social psychologist and head of the Center for Open Science in Charlottesville, Virginia, and 269 co-authors repeated work reported in 98 original papers from three psychology journals, to see if



Science AAAS

RESEARCH

RESEARCH ARTICLE

PSYCHOLOGY

**Estimating the reproducibility of psychological science**

Open Science Collaboration*†

Reproducibility is a defining feature of science, but the extent to which it characterizes current research is unknown. We conducted replications of 100 experimental and correlational studies published in three psychology journals using high-powered designs and original materials when available. Replication effects were half the magnitude of original effects, representing a substantial decline. Ninety-seven percent of original studies had statistically significant results. Thirty-six percent of replications had statistically significant results; 47% of original effect sizes were in the 95% confidence interval of the replication effect size; 39% of effects were subjectively rated to have replicated the original result; and if no bias in original results is assumed, combining original and replication results left 68% with statistically significant effects. Correlational tests suggest that replication success was better predicted by the strength of original evidence than by characteristics of the original and replication teams.

facilitated each step of the process and maintained the protocol and project resources. Replication materials and data were required to be archived publicly in order to maximize transparency, accountability, and reproducibility of the project (https://osf.io/ezum).

In total, 100 replications were completed by 270 contributing authors. There were many different research designs and analysis strategies in the original research. Through consultation with original authors, obtaining original materials, and internal review, replications maintained high fidelity to the original designs. Analyses converted results to a common effect size metric (correlation coefficient (r)) with confidence intervals (CIs). The suite of analysis for inferences about reproducibility were the original and replication study effect sizes. The resulting open data set provides an initial estimate of the reproducibility of psychology and correlational data to support development of hypotheses about the causes of reproducibility.

**Sampling frame and study selection**

We constructed a sampling frame and selection

NIH National Institutes of Health

FNSNF

FONDS NATIONAL SUISSE
SCHWEIZERISCHER NATIONALFONDS
FONDO NAZIONALE SVIZZERO
SWISS NATIONAL SCIENCE FOUNDATION

DEPARTMENT OF HEALTH & HUMAN SERVICES • USA

FDA

# What are the benefits of dynamic reports?

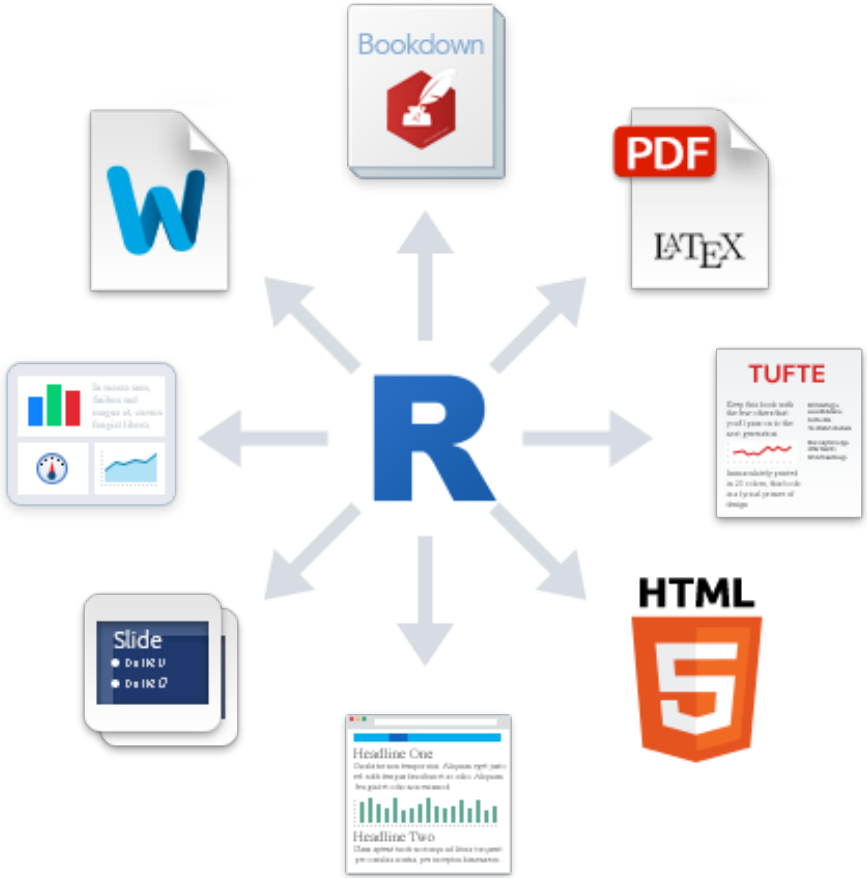
- Data, code, analyses, and plots are all interconnected.
- The code *must* be correct because it is being run as the report is created.
- Anyone, even your future self, can replicate your analyses.
- If something changes in the data, the changes are automatically reflected in the results.
- If you get new data, you can easily create new analyses.


http://rpubs.com/fabiorocha5150/decisiontreemodel

# What dynamic reports can I make in RStudio?

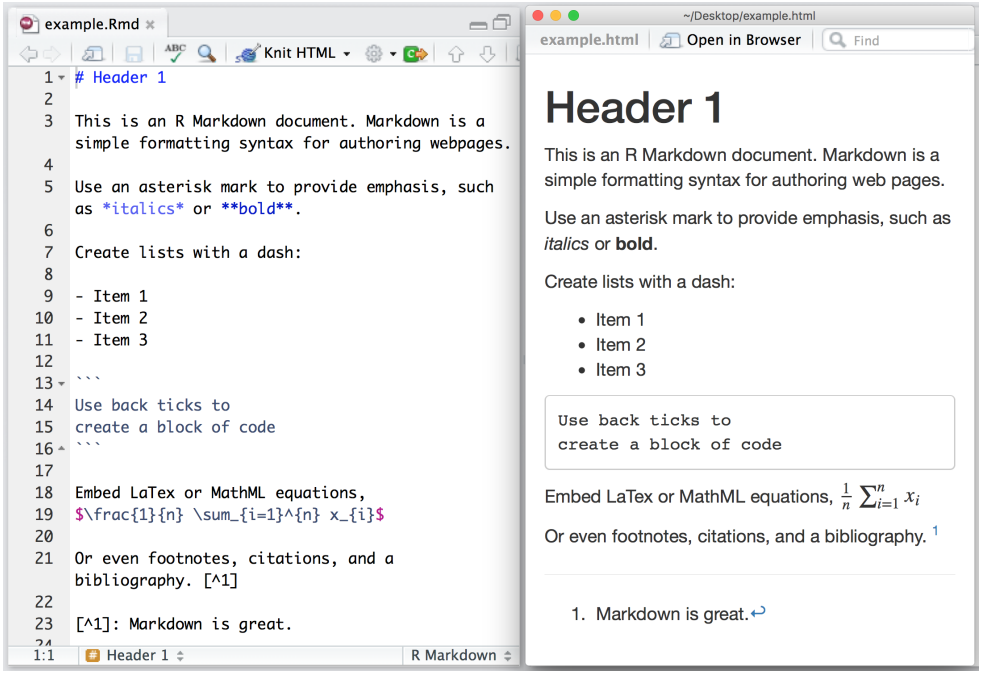
- Web (HTML) pages
  - Websites, blog posts, single documents to share
- PDF documents
- Slideshows (like this one)
- Shiny Apps
- Books
- R Packages

## How? Answer: R Markdown

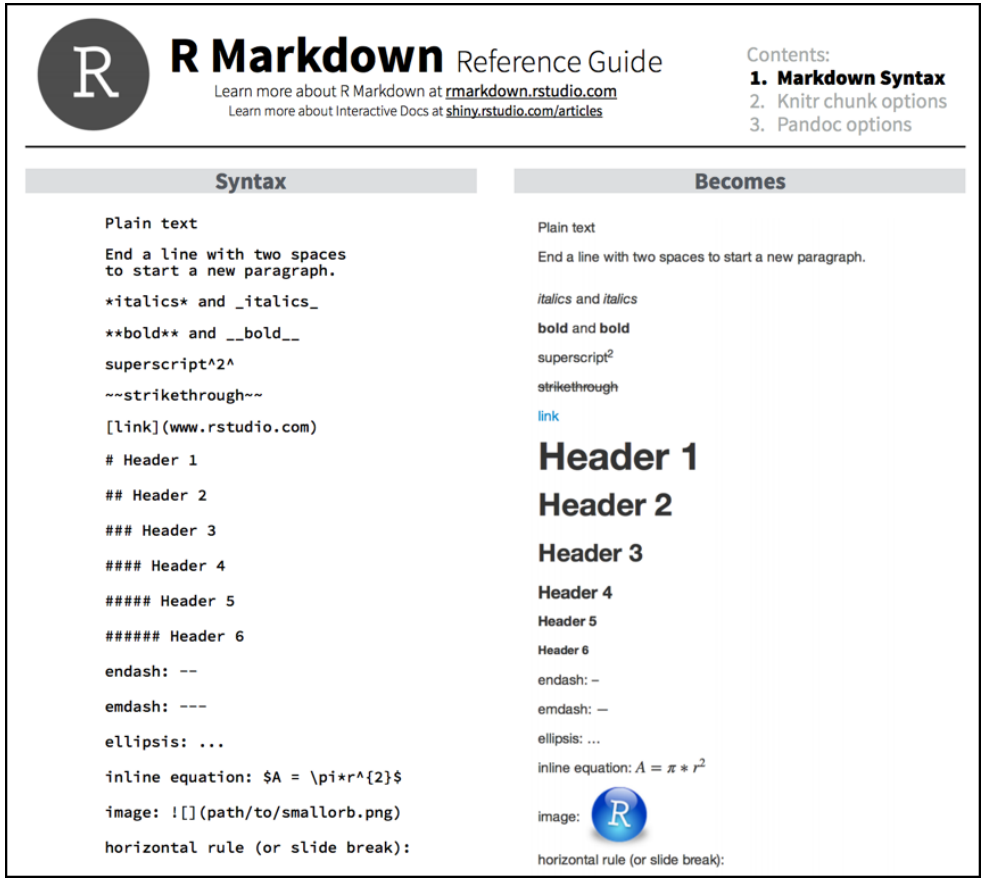


# What is Markdown?

- Markdown is a simple, readible **markup language** that allows you to easily write text with special formatting **tags**, which are then converted to formatted outputs.
- **R Markdown** is a combination of Markdown and R code.



# Cheatsheet



**R Markdown** Reference Guide

Learn more about R Markdown at rmarkdown.rstudio.com
Learn more about Interactive Docs at shiny.rstudio.com/articles

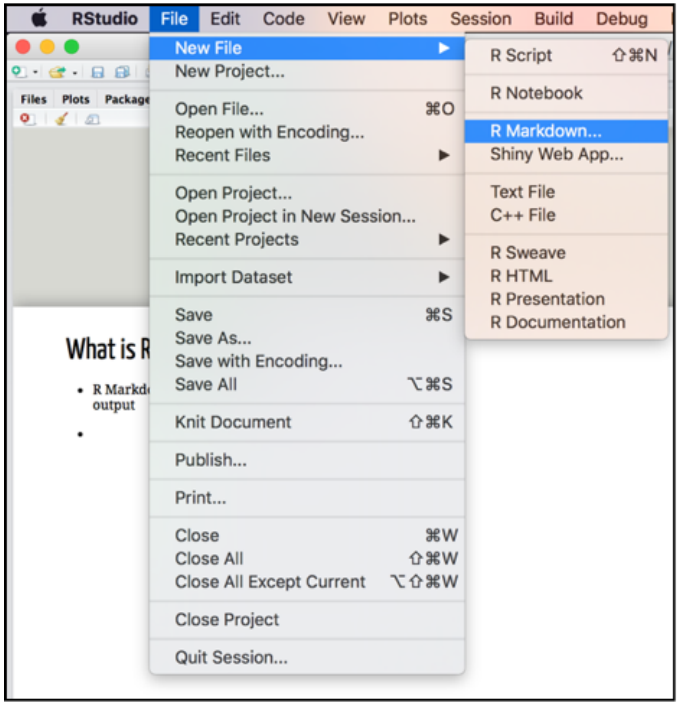
Contents:
1. **Markdown Syntax**
2. Knitr chunk options
3. Pandoc options

| Syntax | Becomes |
|---|---|
| `Plain text` | Plain text |
| `End a line with two spaces`<br>`to start a new paragraph.` | End a line with two spaces to start a new paragraph. |
| `*italics* and _italics_` | *italics* and *italics* |
| `**bold** and __bold__` | **bold** and **bold** |
| `superscript^2^` | superscript$^2$ |
| `~~strikethrough~~` | ~~strikethrough~~ |
| `[link](www.rstudio.com)` | link |
| `# Header 1` | **Header 1** |
| `## Header 2` | **Header 2** |
| `### Header 3` | **Header 3** |
| `#### Header 4` | **Header 4** |
| `##### Header 5` | **Header 5** |
| `###### Header 6` | **Header 6** |
| `endash: --` | endash: – |
| `emdash: ---` | emdash: — |
| `ellipsis: ...` | ellipsis: … |
| `inline equation: $A = \pi*r^{2}$` | inline equation: $A = \pi * r^2$ |
| `image: ![](path/to/smallorb.png)` | image: |
| `horizontal rule (or slide break):` | horizontal rule (or slide break): |

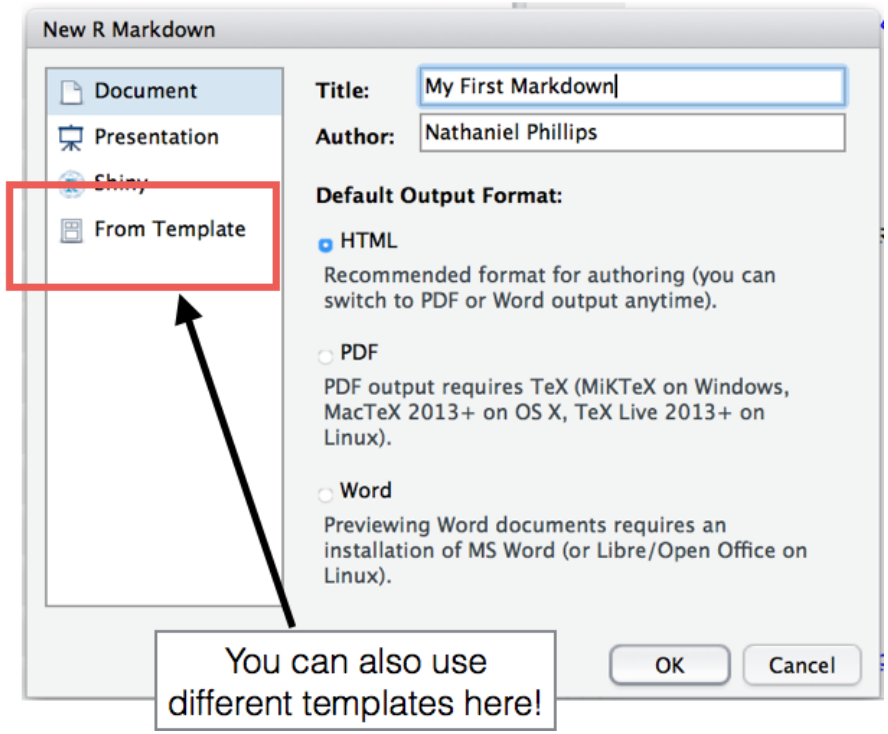
https://www.rstudio.com/wp-content/uploads/2015/03/rmarkdown-reference.pdf

# How do I write an R Markdown Document

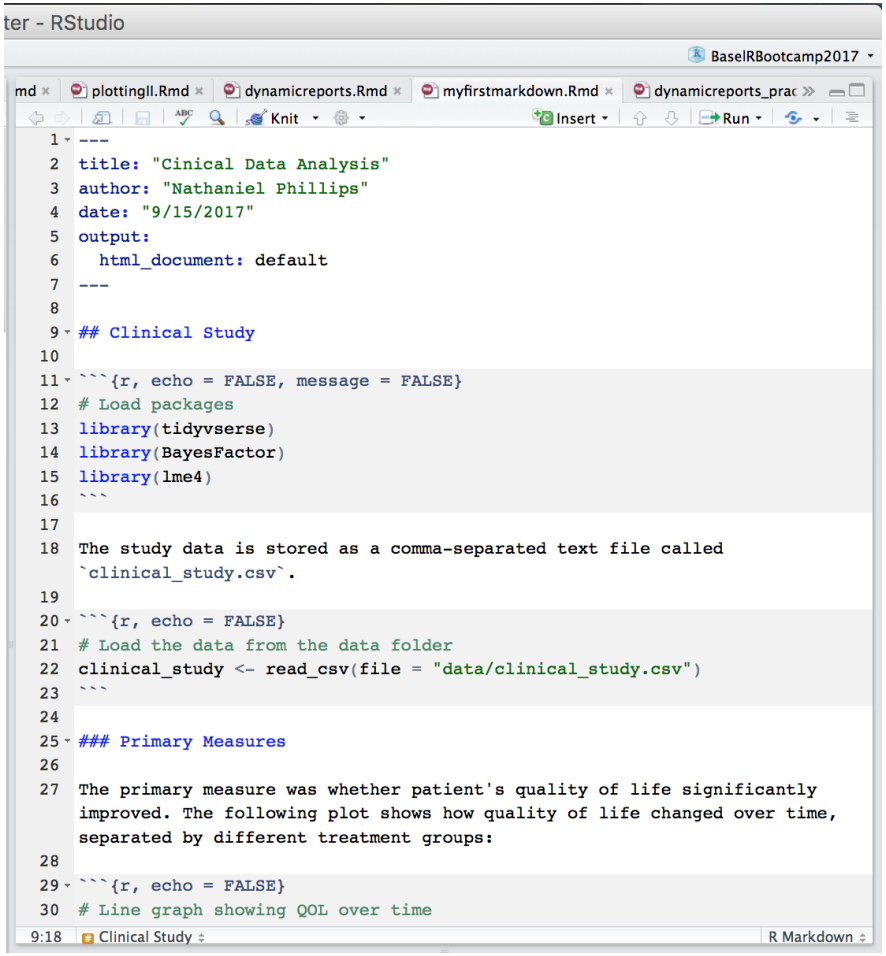
Open New Markdown File in RStudio




Give it a name, and select an optional output format



# R Markdown = Markdown + R code

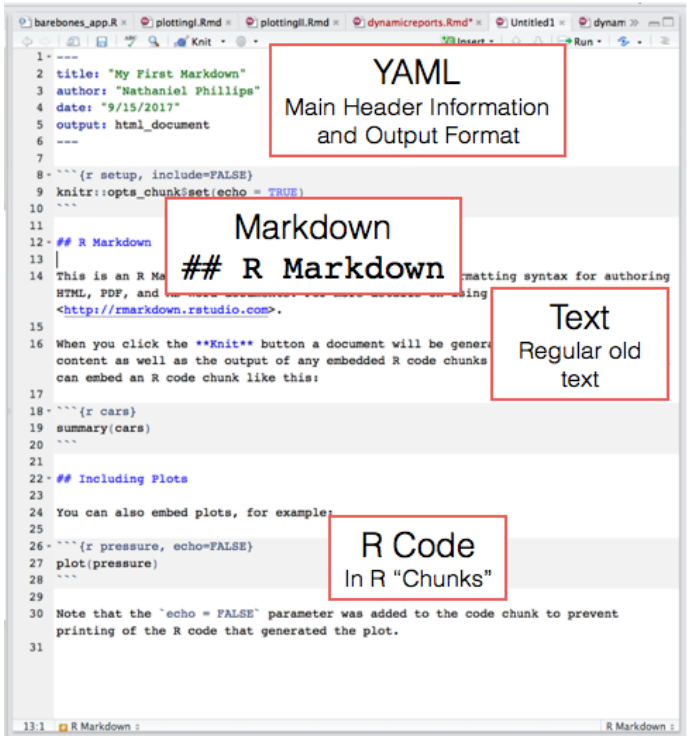


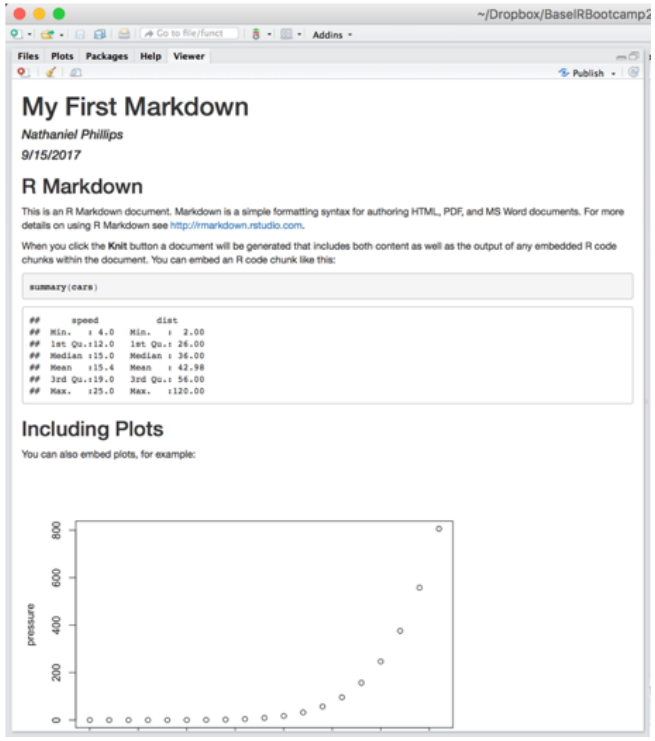
# R Markdown = Markdown + R code

R Markdown
.Rmd File

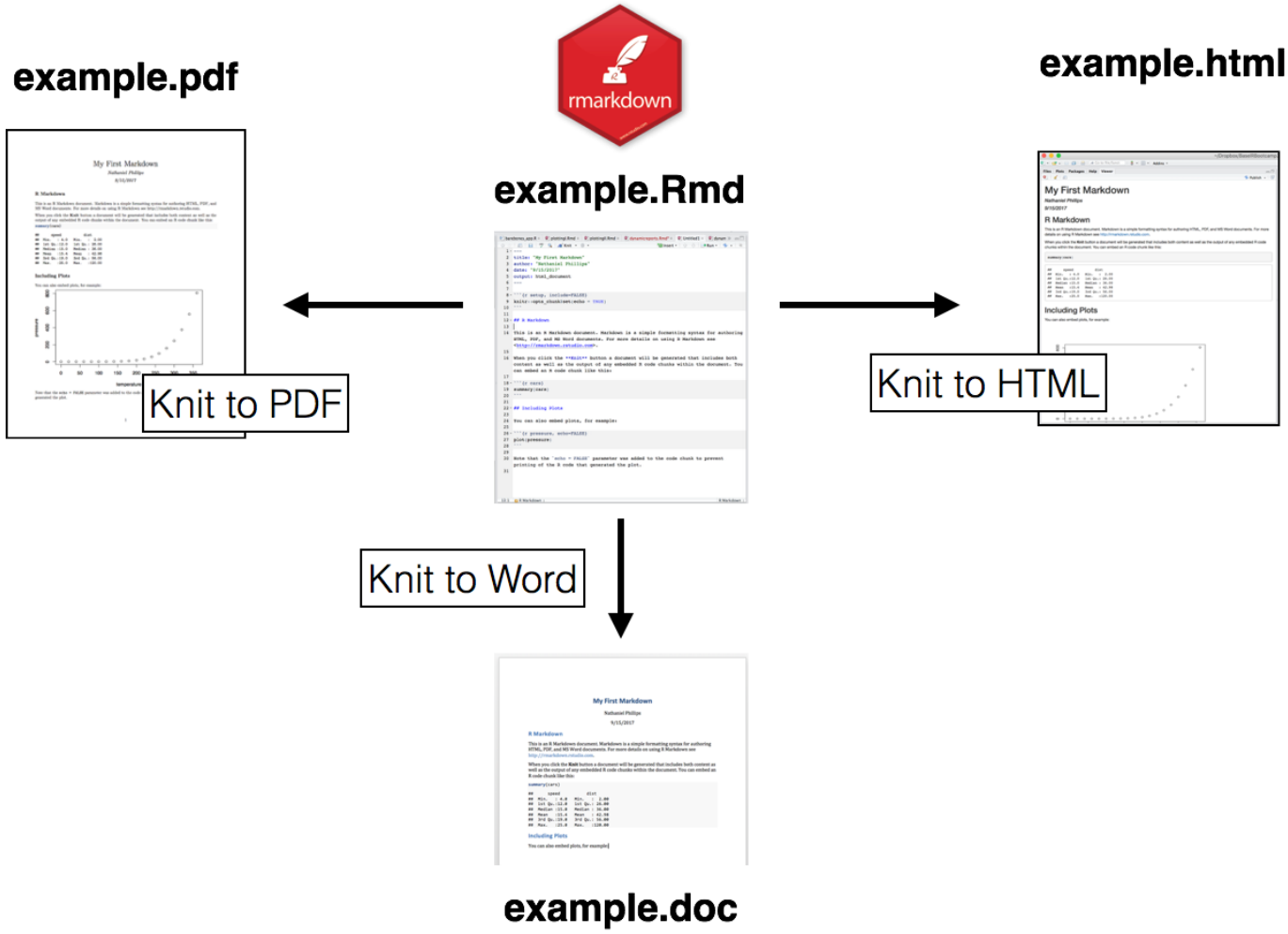
**"Knit"** →

Output
.HTML File



# R Markdown = Markdown + R code



# R Markdown Gallery

http://rmarkdown.rstudio.com/gallery.html

R Markdown (http://rmarkdown.rstudio.com/index.html)

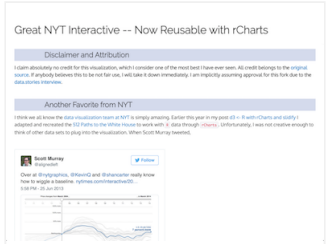
from (https://www.rstudio.com/) Get Started (http://rm

## Gallery

Check out the range of outputs and formats you can create using R Markdown.

### Documents

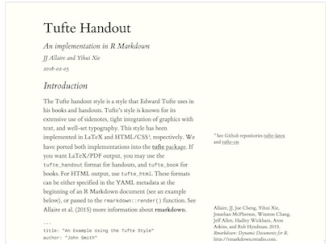
With R Markdown, you write a single .Rmd file and then use it to render finished output in a variety of formats.




(http://timelyportfolio.github.io/rCharts_nyt_vs_gs/)


(https://github.com/svmiller/svm-r-markdown-templates)


Microsoft Word




(https://rstudio.github.io/tufte/)

# R Chunks in R Markdown

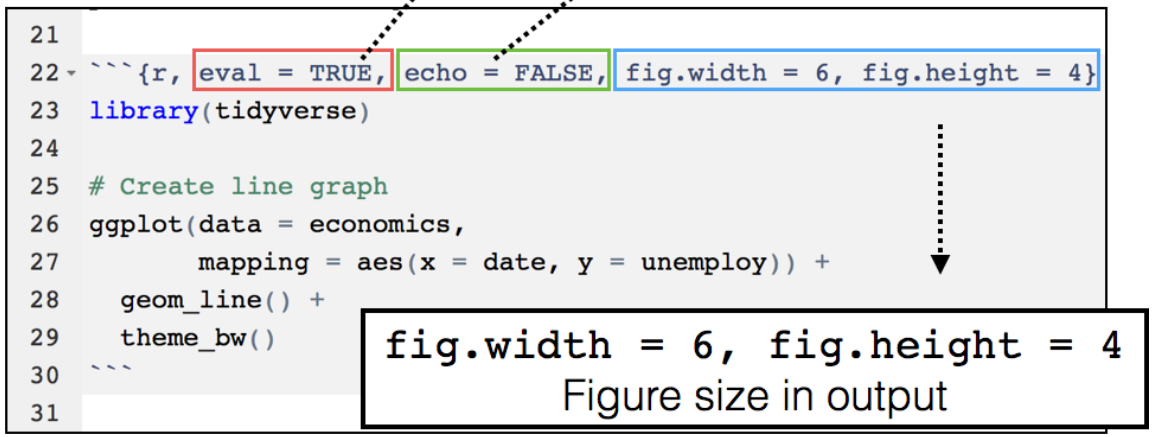
R Chunks look like this:

````
```{r, ARGUMENTS}
# Code
1 + 1
```
````

You can include many arguments in a chunk

**`eval = TRUE`**
"**Do** evaluate this code"

**`echo = FALSE`**
"Do **NOT** print the code"




**`fig.width = 6, fig.height = 4`**
Figure size in output

# R Chunks in R Markdown

- You can set default chunk options with `ops_chunk$set()`. When you do, all future chunks will have these settings (unless you specify otherwise)

```
# Example: Set default values for ALL future chunks
#   with knitr::ops_chunk$set


knitr::opts_chunk$set(fig.width = 6,         # Figure width (in)
                      fig.height = 6,        # Figure height (in)
                      echo = TRUE,           # Repeat code
                      eval = TRUE,           # Evaluate chunks
                      message = FALSE,       # Don't print messages
                      warning = FALSE,       # Don't print warnings
                      fig.align = 'center') # Center figures
```

- We recommend setting chunk options at the beginning of each document!

# Inline chunks

- You can also include *inline chunks* where R code is included in a sentence. This allows you to include R output in your text!

```
39
40  Text text `r CODE` text text. Text text
    text `r CODE` text text text.
41
```

# Inline chunks

R Markdown with inline-chunks → *Knit!* → HTML Output!

```
31 ## Key Results
32
33 There were `r nrow(study_df)` patients in the
   clinical trial. Of these, `r sum(study_df$sex ==
   "f")` were female. The median age of the
   patients was `r median(study_df$age)` and the
   standard deviation was `r
   round(sd(study_df$age), 2)`.
34
35 The primary measure of treatment success was
   defined as patients who indicated that their
   quality of life had significantly improved as a
   result of treatment. Across all conditions, `r
   sum(study_df$success)` patients were successes.
```

## Key Results

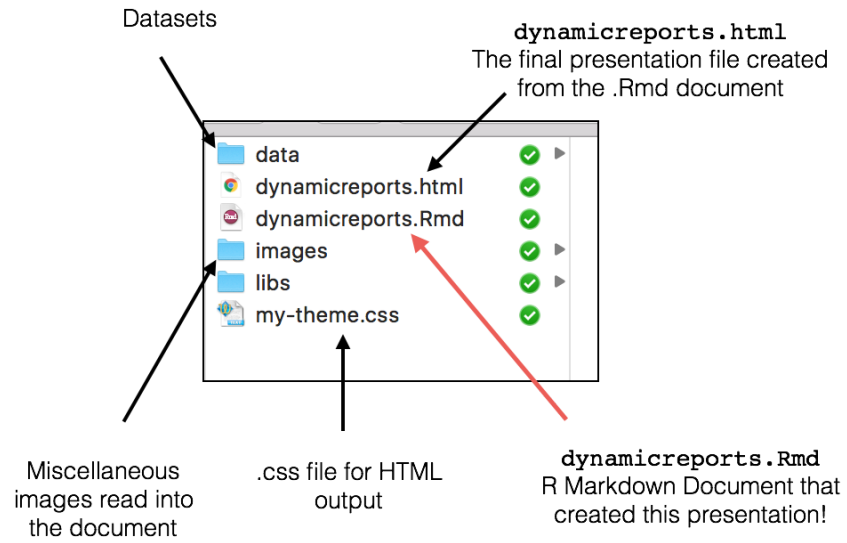
There were 100 patients in the clinical trial. Of these, 52 were female. The median age of the patients was 33 and the standard deviation was 9.77.

The primary measure of treatment success was defined as patients who indicated that their quality of life had significantly improved as a result of treatment. Across all conditions, 20 patients were successes.

# Key points in R Markdown

1. You can easily read in external files like images, R code, or datasets.
2. All external files should be in a folder in the "Root" Directory containing the Markdown (.Rmd) file
3. All data necessary for your report *must* be explicitly read into the document. So keep them close by!

**Here is the "Root" directory of the R Markdown file `dynamicreports.Rmd` that created *this* presentation**



# Rendering output with `knitr`


- To render output like images and tables into R Markdown, use the `knitr` package.
- The two most useful `knitr` functions are `include_graphics()` and `kable()`

| Function | Output |
|---|---|
| `include_graphics(path)` | Include an external image (e.g.; .png, .jpg) |
| `kable(df, format)` | Include a dataframe as a table |

## Print a dataframe as a table with `kable()`

```
kable(economics[1:3, c("date", "pop")],
      format = 'markdown')
```

| date | pop |
|---|---|
| 1967-07-01 | 198712 |
| 1967-08-01 | 198911 |
| 1967-09-01 | 199113 |

## Include an image

```
include_graphics(path = ".../rlogo.png")
```

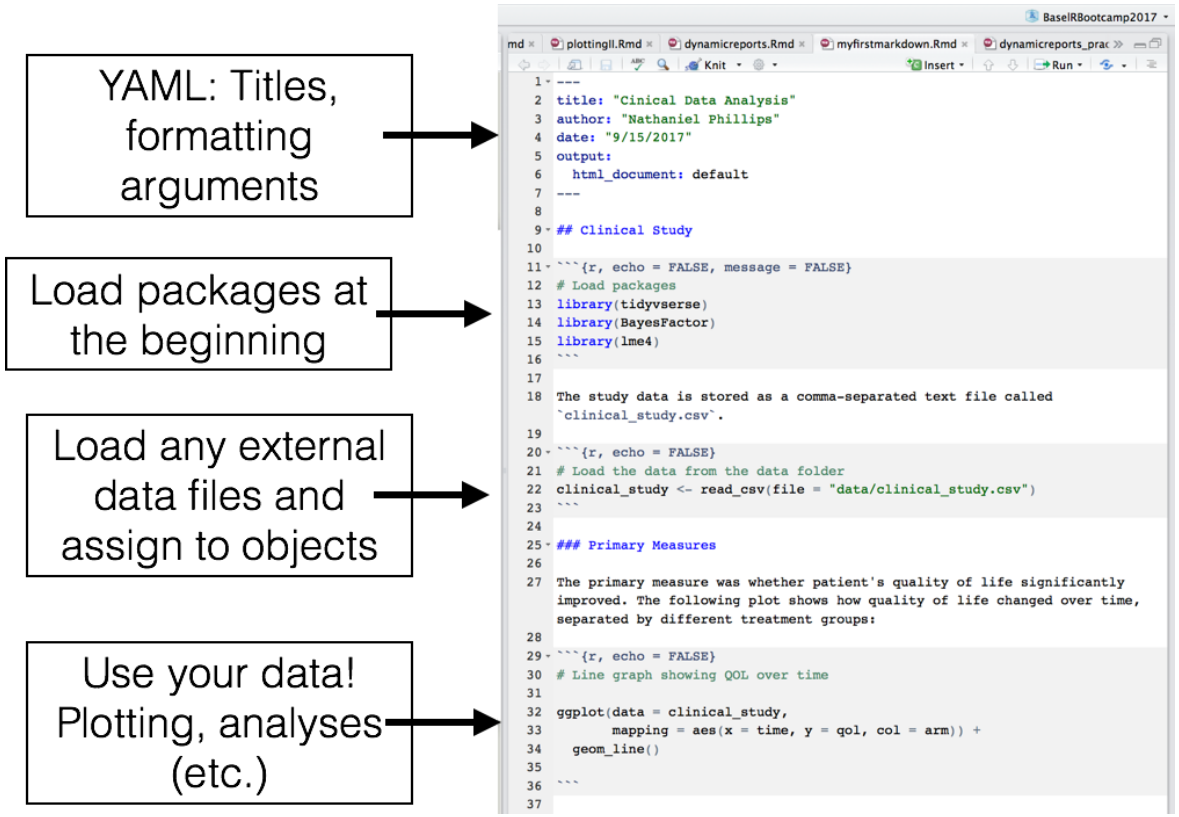
# Key points in R Markdown

1. When you "Knit" an R markdown document, it will start with an *empty* workspace (ie. it will forget everything!)

2. You must explicitly load all packages with `library()` and load in external datasets (e.g.; `clinical_data <- read_csv("data/data.csv")`)

3. If you have any typos, errors, or missing code, the document will *not* knit (this is a *good* thing!)

**Well formatted R Markdown document**



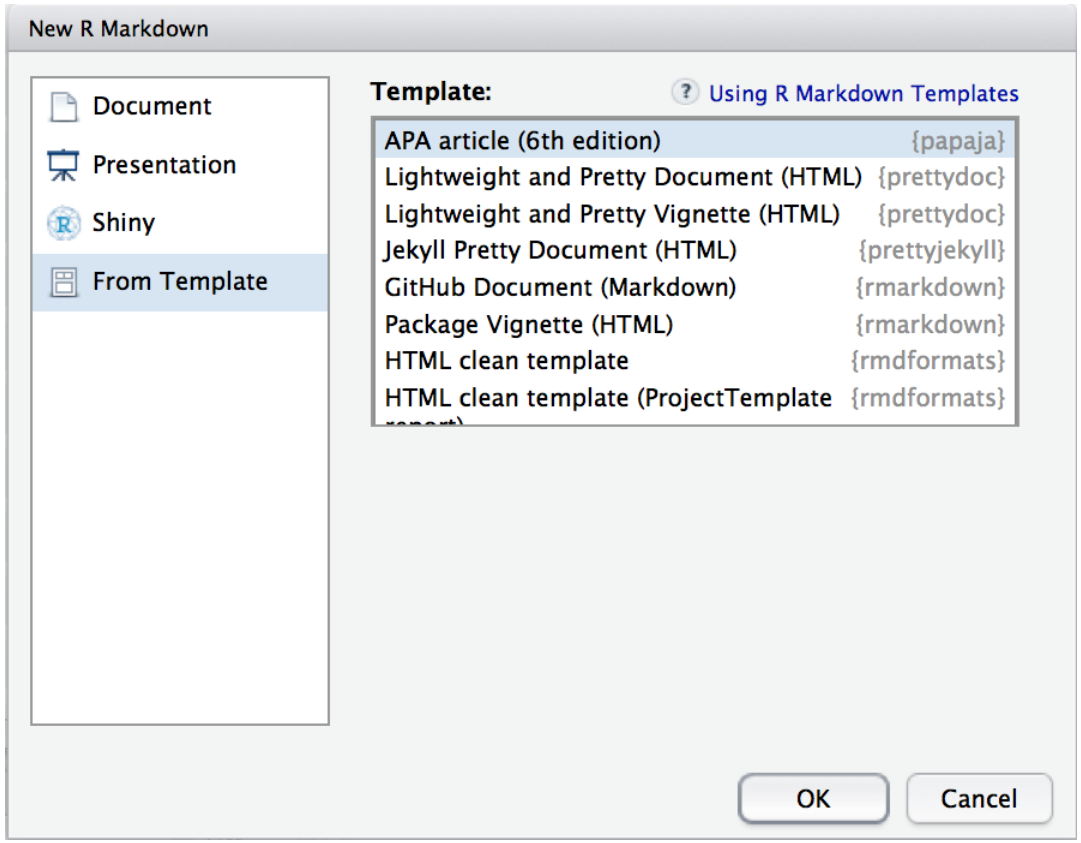
# Output types

- There are *many* different output formats you can create from an R Markdown document
- Many come with RStudio, many are distributed in packages:

| Package | Description |
|---|---|
| `xaringan`* | Slideshows (like this one!) |
| `papaja`* | APA Manuscripts |
| `rmdformats` | Many templates |
| `prettydoc` | Many templates |

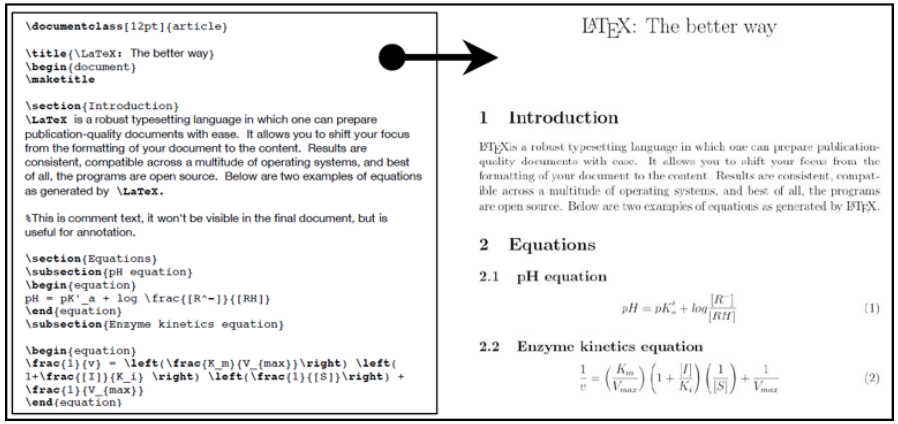
- = On GitHub

## R Markdown templates in R Studio



# To write to PDF, you need TeX

In order to knit your document to a PDF, you'll need a TeX installation

```
\documentclass[12pt]{article}

\title{\LaTeX: The better way}
\begin{document}
\maketitle

\section{Introduction}
\LaTeX is a robust typesetting language in which one can prepare
publication-quality documents with ease. It allows you to shift your focus
from the formatting of your document to the content. Results are
consistent, compatible across a multitude of operating systems, and best
of all, the programs are open source. Below are two examples of equations
as generated by \LaTeX.

%This is comment text, it won't be visible in the final document, but is
useful for annotation.

\section{Equations}
\subsection{pH equation}
\begin{equation}
pH = pK'_a + log \frac{[R^-]}{[RH]}
\end{equation}
\subsection{Enzyme kinetics equation}

\begin{equation}
\frac{1}{v} = \left(\frac{K_m}{V_{max}}\right) \left(
1+\frac{[I]}{K_i} \right) \left(\frac{1}{[S]}\right) +
\frac{1}{V_{max}}
\end{equation}
```

LaTeX: The better way

## 1 Introduction

LaTeXis a robust typesetting language in which one can prepare publication-quality documents with ease. It allows you to shift your focus from the formatting of your document to the content. Results are consistent, compatible across a multitude of operating systems, and best of all, the programs are open source. Below are two examples of equations as generated by LaTeX.

## 2 Equations

### 2.1 pH equation

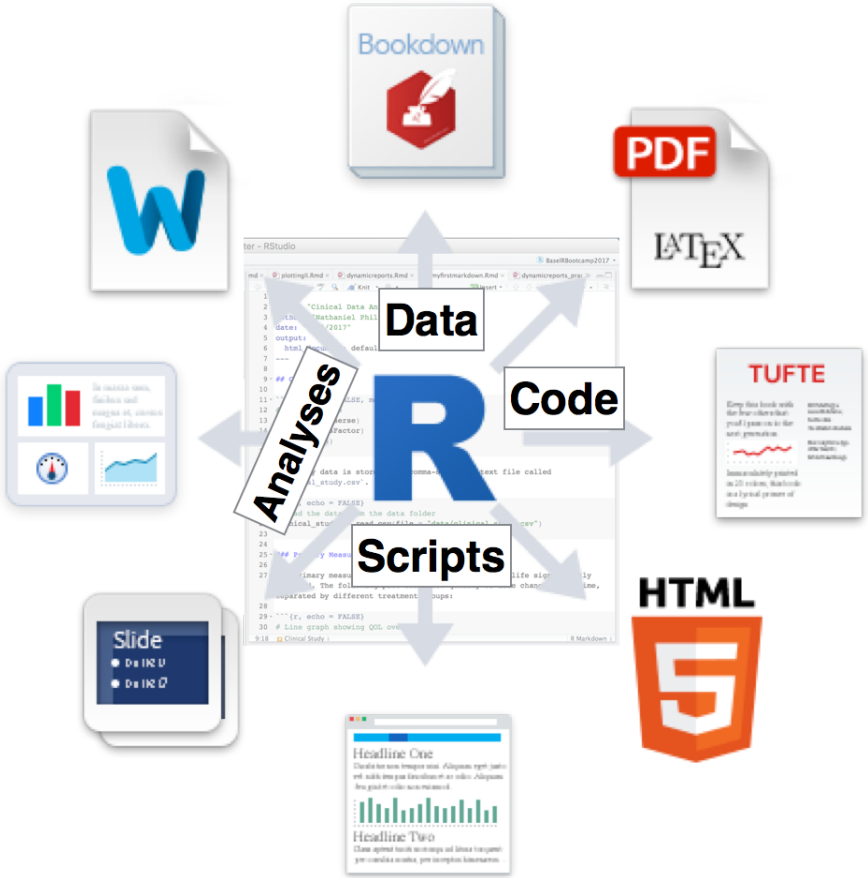
$$pH = pK'_a + log\frac{[R^-]}{[RH]} \quad (1)$$

### 2.2 Enzyme kinetics equation

$$\frac{1}{v} = \left(\frac{K_m}{V_{max}}\right)\left(1 + \frac{[I]}{K_i}\right)\left(\frac{1}{[S]}\right) + \frac{1}{V_{max}} \quad (2)$$

# Why are RMarkdown documents so great?

1. The data, code, and output are all in the same place.
2. Everything works and is replicable! (If it wasn't, the document wouldn't *Knit*!)
3. You can produce great looking documents, from simple PDFs, to webpages, to presentations (like this one), to books.



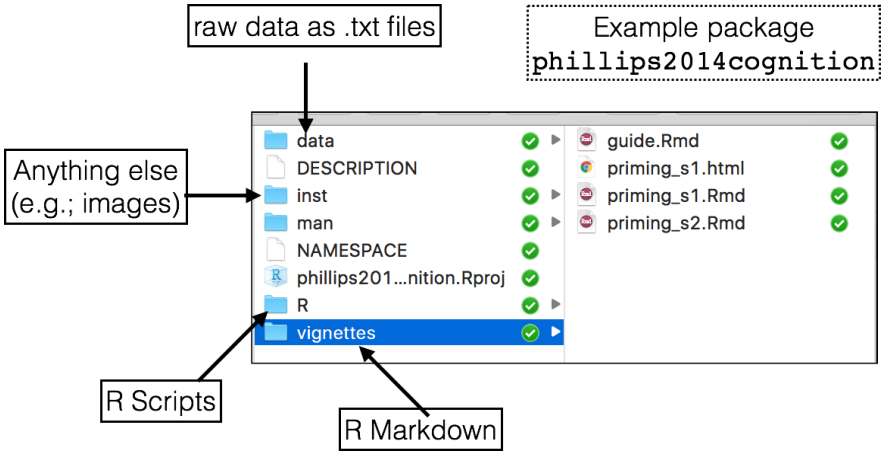
# R Packages - The ultimate dynamic report?

If you want a fully contained, dynamic report that contains data, code, statistics, plots, text, help files and documentation that you can easily share, your answer may be creating an **R Package**.

**Installing an example package from GitHub**

```
# Install cStudy2017 from github
install_github("bootcamp/cStudy2017")

# Load cStudy2017 package
library("cStudy2017")

# Open cStudy2017 package help
?cStudy2017

# Oh study1_A contains study 1...
ggplot(data = study1_A,
       ...)

FFTrees(success ~ .,
        data = study1_A)
```

## Folder structure for a simple package




Read Wickham's book **"R Packages"** (also available for free online) to learn how to write your own package

# Dynamic Reports Pratical

**Link to Dynamic Reports practical**