

# Report Reproducibly with



Navigate to the **05-Report** folder.  
Open **05-Report-Exercises.Rmd**

# R Markdown

# R Markdown

Plain text file with 3 types of content:

The screenshot shows an RStudio interface with an R Markdown file open. The file contains the following content:

```
1 ---  
2 title: "R Notebook"  
3 output: html_notebook  
4 ---  
5  
6 Text written in **markdown**  
7  
8 ```{r}  
9 # code written in R  
10 (x <- rnorm(7))  
11 ...  
12  
13 Text written in _markdown_  
14  
15 ```{r}  
16 # code written in R  
17 hist(x)  
18 ...  
18:4 (Top Level) ⇩  
Console
```

Annotations explain the three types of content:

- A green callout points to the YAML header: "A YAML header surrounded by ---".
- A grey callout points to the text in markdown: "Text in markdown".
- A blue callout points to the code chunks: "Code chunks surrounded by ```".

# How it works

R

# knitr



# pandoc



**HTML**



ioslides  
slidy, Beamer



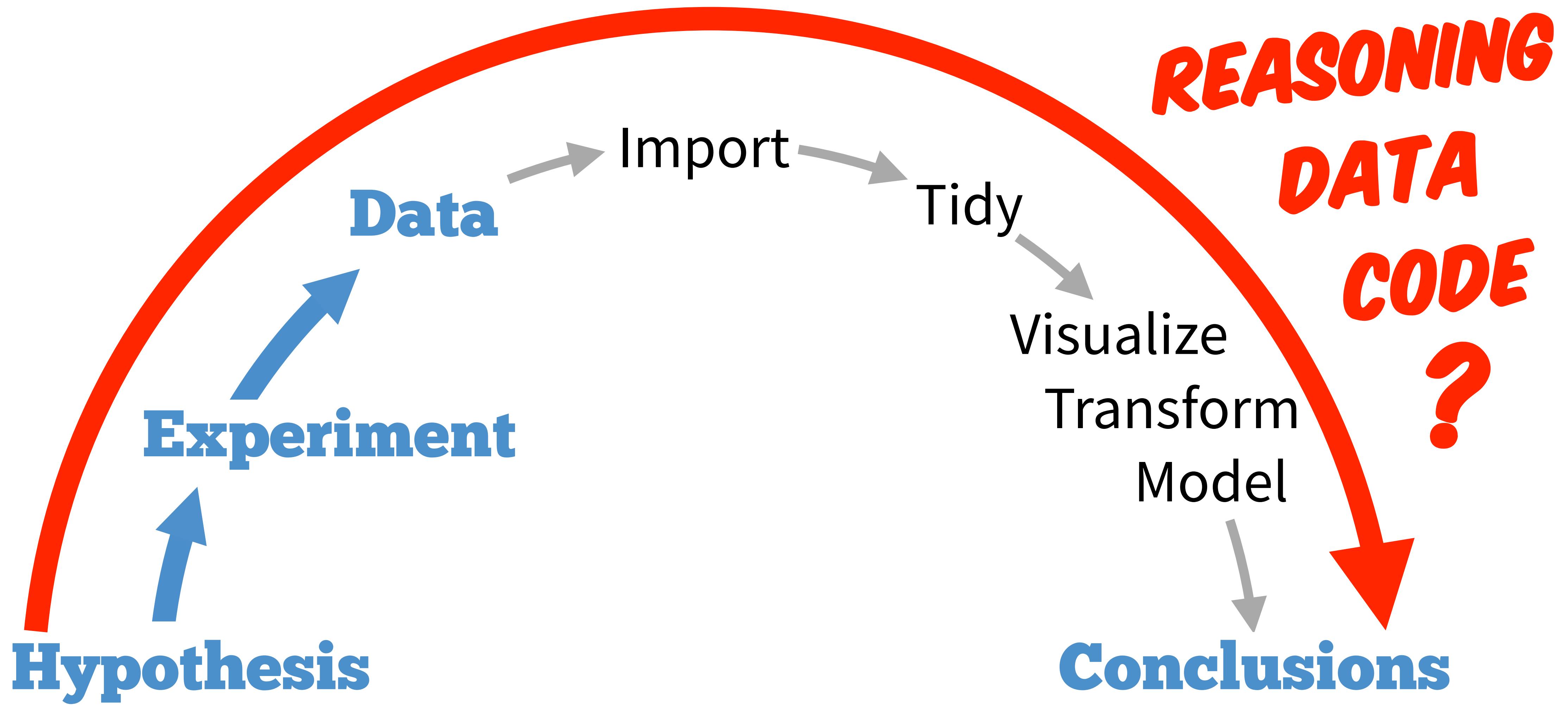
Powerpoint



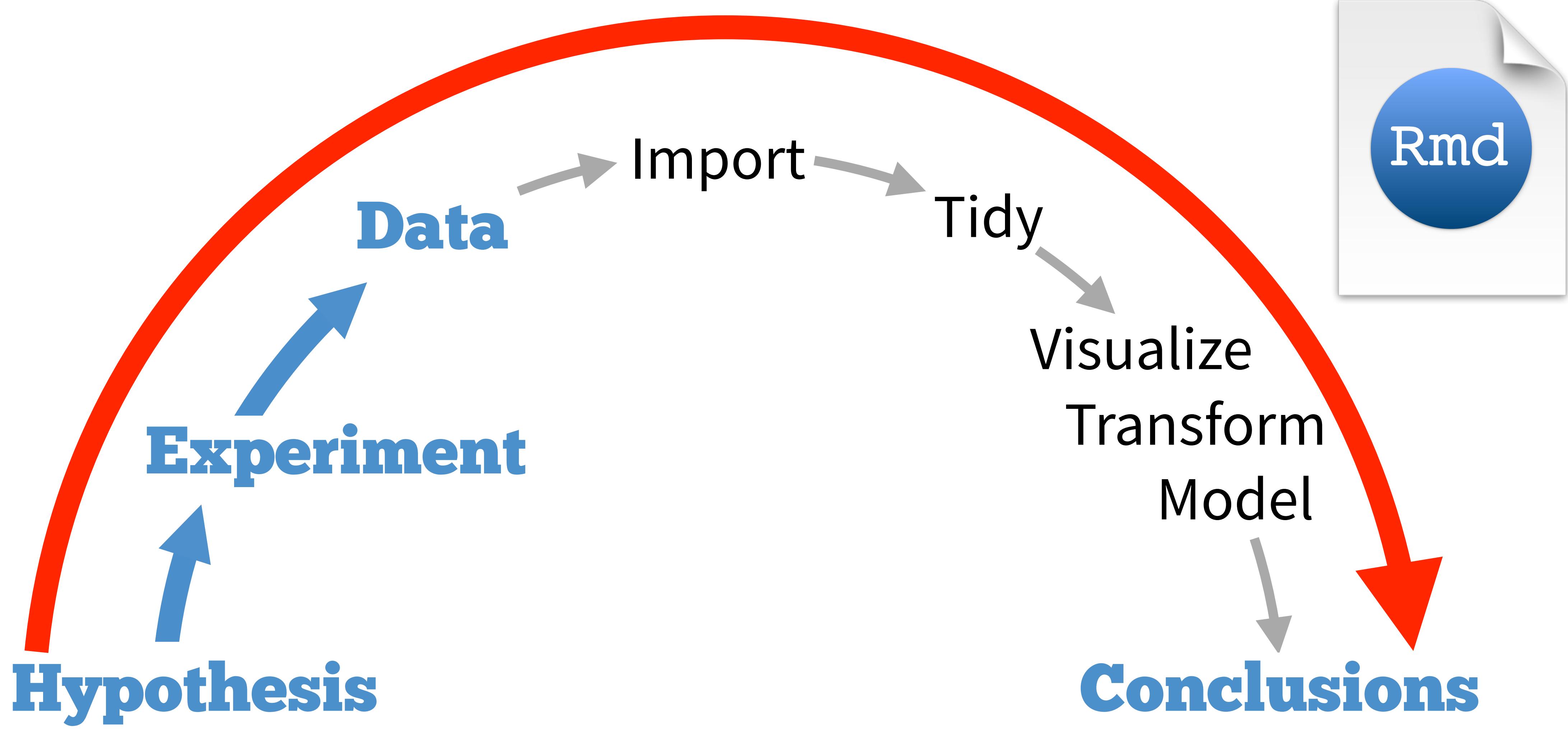
Microsoft Word



# Data Science



# Data Science



# 1

# Logistics

Knitr runs the document in a fresh R session, which means you need to load the libraries that the document uses *in the document*



# Logistics

1

Knitr runs the document in a fresh R session, which means you need to load the libraries that the document uses *in the document*

2

Objects made in one code chunk will be available to code in later code chunks.



# **KNITR IS MULTILINGUAL!**

 **SAS**

 **PYTHON**

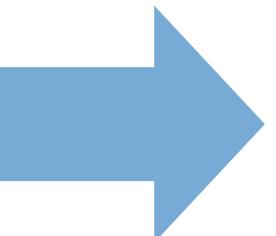
 **MORE**

# engine

python

```
Some python code,  
```{python}  
x = 'hello, python  
world!'  
print(x)  
print(x.split(' '))  
```
```

To embed non R code, change the chunk label from r to the language to use.



Some python code:

```
x = 'hello, python world!'  
print(x)  
print(x.split(' '))
```

```
## hello, python world!  
## ['hello,', 'python', 'world!']
```



# Reticulate

## Python in R Markdown

(Optional) Build Python env to use.

Add `knitr::knit_engines$set(python = reticulate::eng_python)` to the setup chunk to set up the reticulate Python engine (not required for `knitr >= 1.18`).

Suggest the Python environment to use, in your setup chunk.

Begin Python chunks with ````{python}`. Chunk options like `echo`, `include`, etc. all work as expected.

Use the `py` object to access objects created in Python chunks from R chunks.

Python chunks all execute within a **single** Python session so you have access to all objects created in previous chunks.

Use the `r` object to access objects created in R chunks from Python chunks.

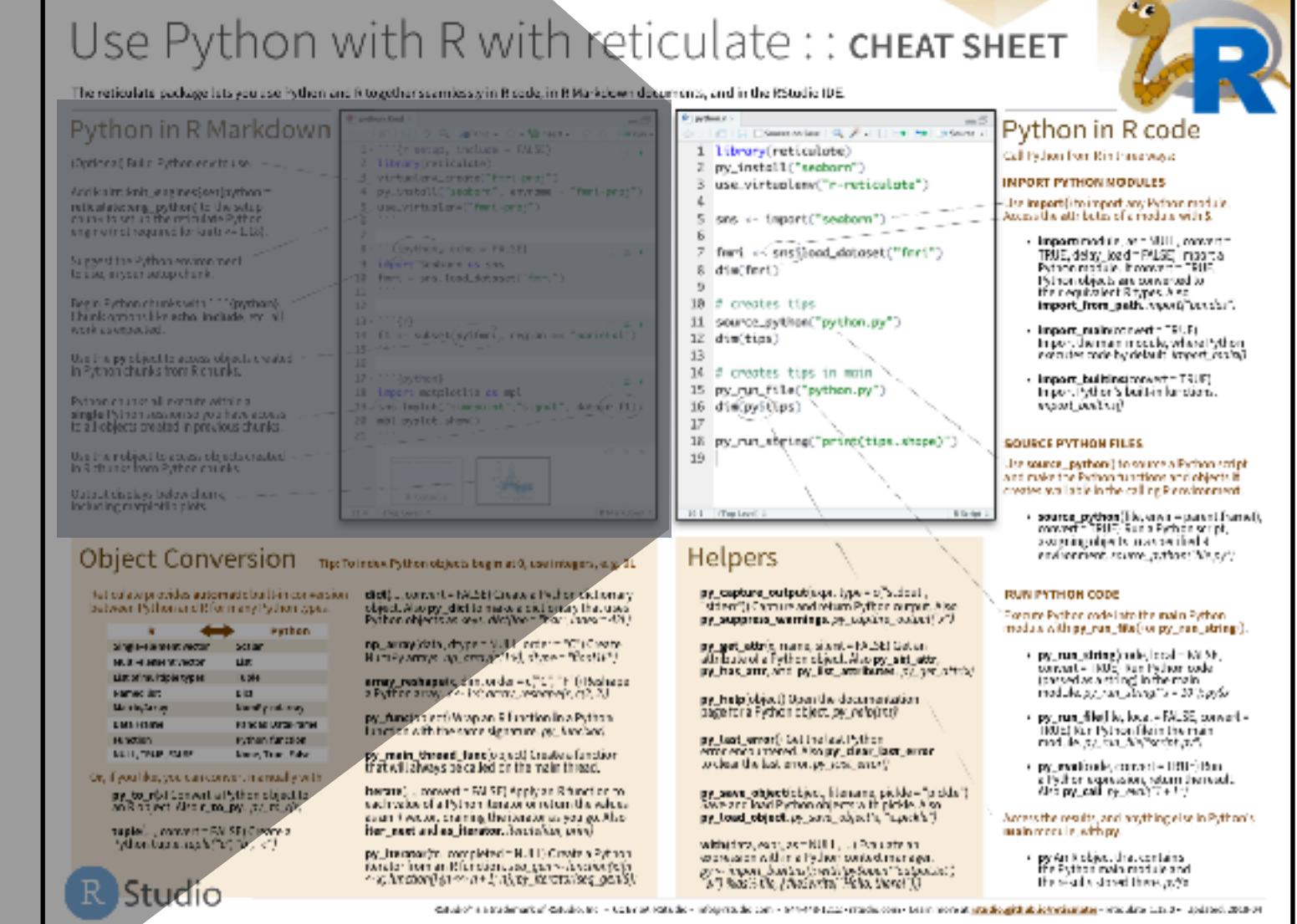
Output displays below chunk, including matplotlib plots.

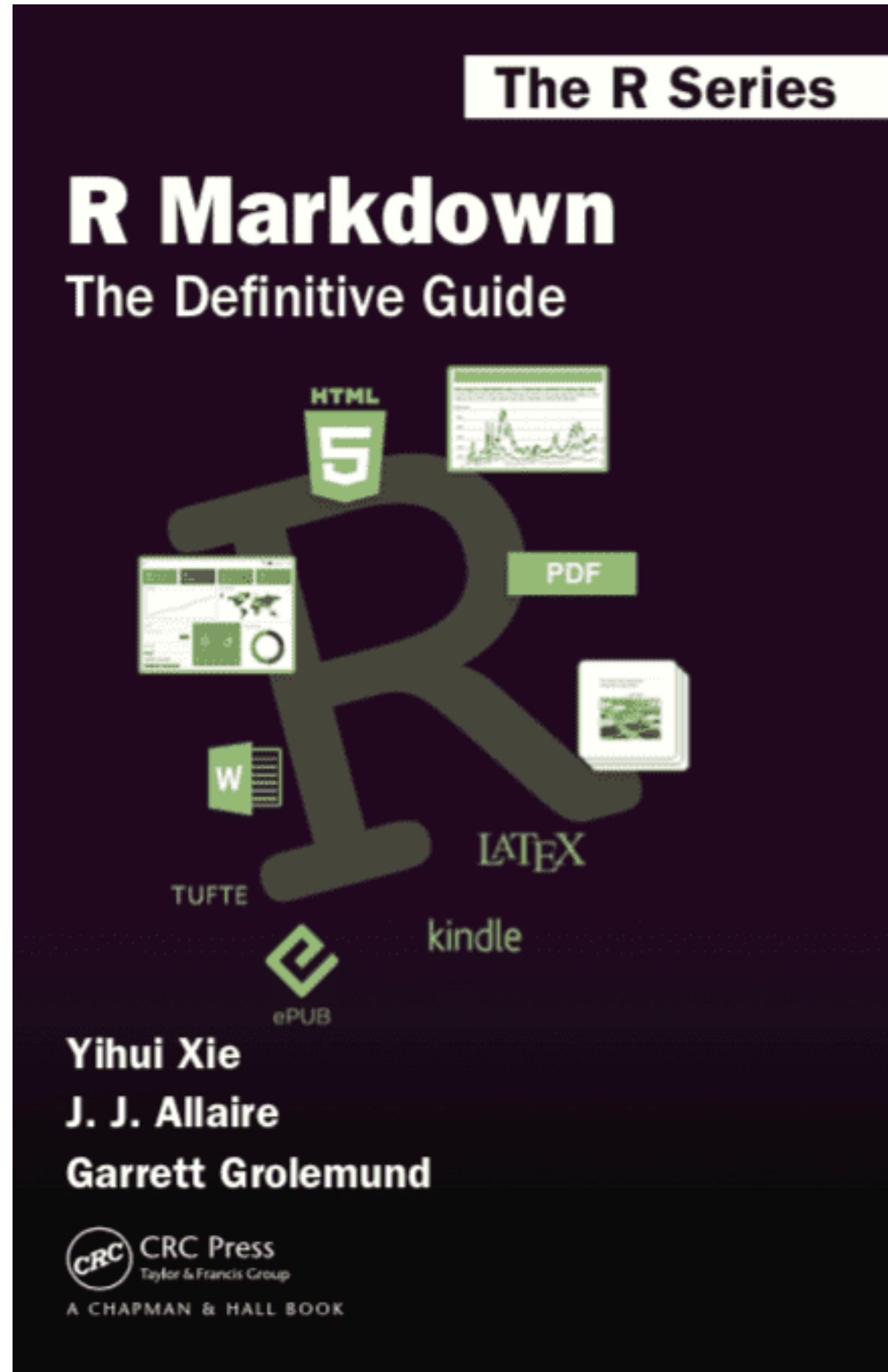
The screenshot shows an RStudio interface with an R Markdown file open. The code includes a setup chunk to build a Python environment, followed by several Python chunks demonstrating various features:

```
1  ```{r setup, include = FALSE}
2  library(reticulate)
3  virtualenv_create("fmri-proj")
4  py_install("seaborn", envname = "fmri-proj")
5  use_virtualenv("fmri-proj")
6
7
8  ```{python, echo = FALSE}
9  import seaborn as sns
10 fmri = sns.load_dataset("fmri")
11
12
13  ```{r}
14  f1 <- subset(py$fmri, region == "parietal")
15
16
17  ```{python}
18  import matplotlib as mpl
19  sns.lmplot("timepoint", "signal", data=r.f1)
20  plt.show()
21
```

The RStudio interface shows the R Console and R Markdown panes. Below the code, two matplotlib plots are displayed: a scatter plot of signal vs timepoint and a corresponding heatmap.

## A package for using R and Python together.





[bookdown.org/yihui/rmarkdown/](http://bookdown.org/yihui/rmarkdown/)

ONLINE, FREE



# Markdown

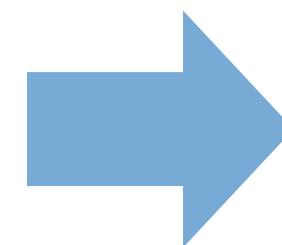
A faint watermark of the R logo is visible in the bottom right corner, consisting of two overlapping circles with the letters 'R' inside.

# Headers

Use # to create headers.

Multiple #'s create lower level headers.

```
# Header 1  
## Header 2  
### Header 3  
#### Header 4  
##### Header 5  
##### Header 6
```



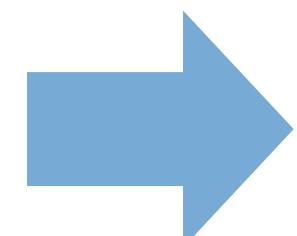
**Header 1**  
**Header 2**  
**Header 3**  
**Header 4**  
**Header 5**  
**Header 6**

# Text

Add two spaces at  
the end of a line to  
start a new line

Text is rendered as plain text. Surround  
text with \_, **\*\***, or ``` to format it.

Text  
italics  
**bold**  
`code`



**Text**  
*italics*  
**bold**  
`code`

# Lists

Use asterisks to make bullet points.

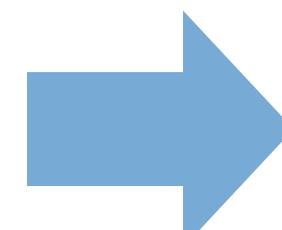
Use numbers to make numbered lists.

## Bullets

- \* bullet 1
- \* bullet 2

## Numbered list

1. item 1
2. item 2



## Bullets

- bullet 1
- bullet 2

## Numbered list

1. item 1
2. item 2

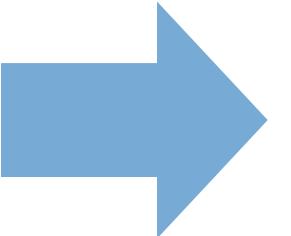
# Hyperlinks

Use brackets to denote a link.

Place the URL in parentheses.

This is a  
[link]([www.git.com](http://www.git.com)).

**This is a link.**



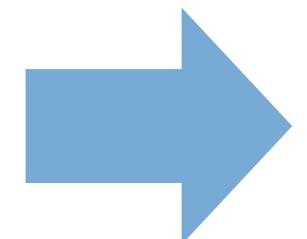
# Images

Use a link preceded by an ! to insert an image.

*The link text should be a URL (if the image is hosted online), or a file path (if the image is saved as a file)*



The RStudio logo.

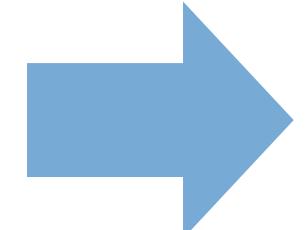


The RStudio logo.

# Equations

Write equations with latex math commands and surround them in \$'s.

According to Einstein,  
 $E=mc^2$



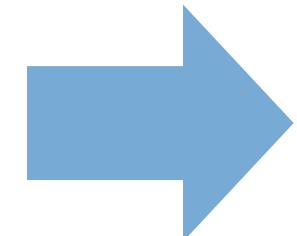
According to Einstein,  $E = mc^2$

# Equation blocks

Use two \$'s to make  
centered equation blocks.

According to  
Einstein,

`$$E=mc^{\{2\}}$$`



According to  
Einstein,

$$E = mc^2$$

# Markdown

**Pandoc's Markdown**  
Write with syntax on the left to create effect on right (after render)

```

Plain text
End a line with two spaces
to start a new paragraph.
*italics* and **bold**
`verbatim code`
sub/superscript22
~~strikethrough~~
escaped: `\\` 
endash: -, emdash: ---
equation: $A = \pi r^2$ 
equation block:

$SE = mc^2$$

> block quote

# Header1 [#anchor]

## Header 2 {css_id}

### Header 3 {css_class}

#### Header 4

##### Header 5

##### Header 6

<--Text comment-->

\textbf{Text ignored in HTML}
<em>HTML ignored in pdfs</em>

<a href="http://www.rstudio.com">
  [link]([www.rstudio.com])
</a> Jump to [Header 1](#anchor)
image: ![Caption](smallorb.png)

* unordered list
  + sub-item 1
  + sub-item 2
    - sub-sub-item 1
* item 2
  • item 2
    Continued (indent 4 spaces)
Continued (indent 4 spaces)
1. ordered list
2. item 2
  i. sub-item 1
    A. sub-sub-item 1
  1. A list whose numbering continues after
  2. an interruption
  (@) an interruption
Term 1
Definition 1
Right Left Default Center
12 12 12 12
123 123 123 123
1 1 1 1
- slide bullet 1
- slide bullet 2
(>- to have bullets appear on click)
horizontal rule/slide break:
*** 
A footnote [^1]
[^1]: Here is the footnote.
1. Here is the footnote.2

```

**Pandoc's Markdown**  
Write with syntax on the left to create effect on right (after render)

```

Plain text
End a line with two spaces
to start a new paragraph.
*italics* and **bold**
`verbatim code`
sub/superscript22
~~strikethrough~~
escaped: `\\` 
endash: -, emdash: ---
equation: $A = \pi r^2$ 
equation block:

$SE = mc^2$$

> block quote

# Header1 [#anchor]

## Header 2 {css_id}

### Header 3 {css_class}

#### Header 4

##### Header 5

##### Header 6

<--Text comment-->

\textbf{Text ignored in HTML}
<em>HTML ignored in pdfs</em>

<a href="http://www.rstudio.com">
  [link]([www.rstudio.com])
</a> Jump to [Header 1](#anchor)
image: ![Caption](smallorb.png)

* unordered list
  + sub-item 1
  + sub-item 2
    - sub-sub-item 1
* item 2
  • item 2
    Continued (indent 4 spaces)
Continued (indent 4 spaces)
1. ordered list
2. item 2
  i. sub-item 1
    A. sub-sub-item 1
  1. A list whose numbering continues after
  2. an interruption
  (@) an interruption
Term 1
Definition 1
Right Left Default Center
12 12 12 12
123 123 123 123
1 1 1 1
- slide bullet 1
- slide bullet 2
(>- to have bullets appear on click)
horizontal rule/slide break:
*** 
A footnote [^1]
[^1]: Here is the footnote.
1. Here is the footnote.2

```

## Dictionary of formatting cues.

**Set render options with YAML**

When you render R Markdown

1. runs the R code, embeds results and text into .md file with knitr
2. then converts the .md file into the finished format with pandoc

Set a document's default output format in the YAML header:

```

```yaml
output: html_document
```

```

**sub-option** **description**

| sub-option            | description   |
|-----------------------|---|
| citation_package      | The LaTeX package to process citations, natbib, biblatex or none                |
| code_folding          | Let readers to toggle the display of R code, "none", "hide", or "show"          |
| colortheme            | Beamer color theme to use   |
| css                   | CSS file to use to style document   |
| dev                   | Graphics device to use for figure output (e.g. "png")                           |
| duration              | Add a countdown timer (in minutes) to footer of slides                          |
| fig_caption           | Should figures be rendered with captions?                                       |
| fig_height, fig_width | Default figure height and width (in inches) for document                        |
| highlight             | Syntax highlighting: "tango", "pygments", "kate", "zenburn", "textmate"         |
| includes              | File of content to place in document (in _header, before_body, after_body)      |
| incremental           | Should bullets appear one at a time (on presenter mouse clicks)?                |
| keep_md               | Save a copy of .md file that contains knitr output                              |
| keep_tex              | Save a copy of .tex file that contains knitr output                             |
| latex_engine          | Engine to render latex, "pdflatex", "xelatex", or "luatex"                      |
| lib_dir               | Directory of dependency files to use (Bootstrap, MathJax, etc.)                 |
| mathjax               | Set to local or a URL to use a local/URL version of MathJax to render equations |
| md_extensions         | Markdown extensions to add to default definition of R Markdown                  |
| number_sections       | Add section numbering to headers  |
| pandoc_args           | Additional arguments to pass to Pandoc  |
| preserve_yaml         | Preserve YAML front matter in final document?                                   |
| reference_docx        | docx file whose styles should be copied when producing docx output              |
| self_contained        | Embed dependencies into the doc   |
| slide_level           | The lowest heading level that defines individual slides                         |
| smaller               | Use the smaller font size in the presentation?                                  |
| smart                 | Convert straight quotes to curly, dashes to em-dashes, ... to ellipses, etc.    |
| template              | Pandoc template to use when rendering file quarterly_report.html).              |
| theme                 | Beamer or Bootstrap theme to use for page                                       |
| toc                   | Add a table of contents at start of document                                    |
| toc_depth             | The lowest level of headings to add to table of contents                        |
| toc_float             | Float the table of contents to the left of the main content                     |

Customize output with sub-options (listed to the right):

```

```yaml
output: html_document
  code_folding: hide
  toc_float: TRUE
```

```

**html tabs**

Use tabset css class to place sub-headers into tabs

```

```yaml
# Tab 1
# Tab 2
# End tabset
```

```

**Create a Reusable Template**

1. Create a new package with a inst/markdown/templates directory
2. In the directory, Place a folder that contains: template.yaml (see below) Section 1 (contents of the template) any supporting files
3. Install the package
4. Access template in wizard at File ► New File ► R Markdown template.yaml

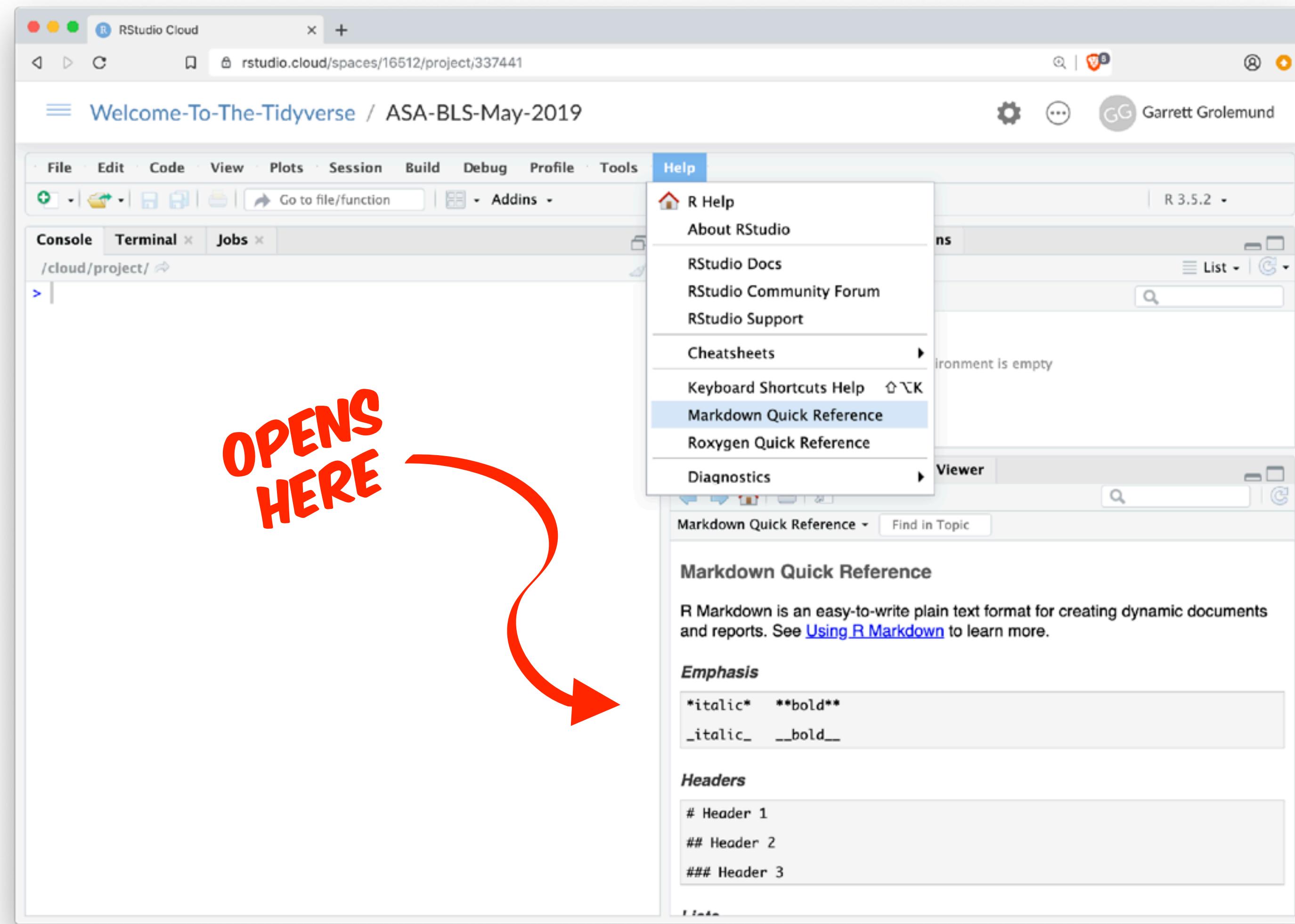
RStudio® is a trademark of RStudio, Inc. • CC BY SA RStudio • info@rstudio.com • 844-448-1212 • rstudio.com • Learn more at [rmarkdown.rstudio.com](#) • rmarkdown 1.6 • Updated: 2016-02

ON BACK OF  
RMARKDOWN  
CHEAT SHEET



# IDE Reference

Go to Help > Markdown Quick Reference



# Embed code

A large, semi-transparent watermark of the R logo is positioned in the bottom right corner. The logo consists of a bold, italicized capital letter 'R' enclosed within a circular arrow.

# Code chunks

Insert a chunk of R code with

```
```{r}
# some code
```
```

When you render the report, R Markdown will run the code and include its results. R Markdown will also remove the ```{r} and ```.

# Code chunks

Insert a chunk of R code with

```
```{r}  
# some code  
```
```

 + Opt + i (Mac)

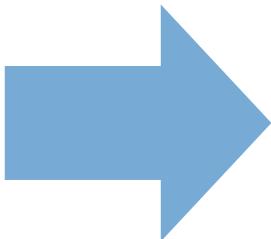
Ctrl + Alt + i (PC)



# chunk options

By default, R Markdown includes both the code and its results

```
Here's some code  
```{r}  
dim(iris)  
```
```

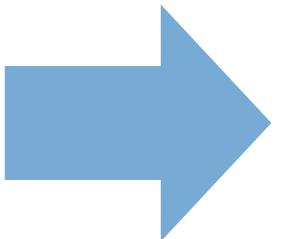


```
Here's some code  
  
dim(iris)  
  
## [1] 150      5
```

# echo

Add options in the brackets after r.  
**echo = FALSE** hides the code.

```
Here's some code  
```{r echo=FALSE}  
dim(iris)  
```
```



```
Here's some code  
## [1] 150 5
```

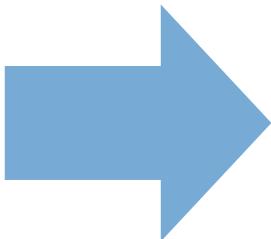
Very useful  
for plots



# eval

**eval = FALSE** prevents the code from being run. As a result, no results will be displayed with the code.

```
Here's some code  
```{r eval=FALSE}  
dim(iris)  
```
```



```
Here's some code  
dim(iris)
```

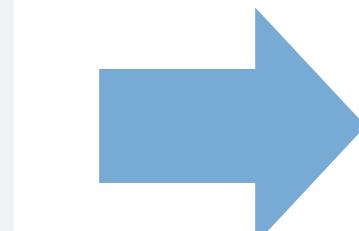
# include

**include = FALSE** runs the code, but prevents both the code and the results from appearing (e.g. to setup).

Here's some code

```
```{r include=FALSE}  
dim(iris)  
```
```

Here's some code

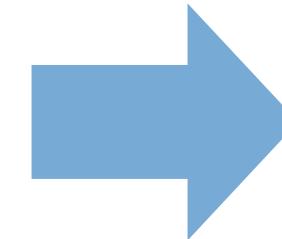


# fig.height, fig.width

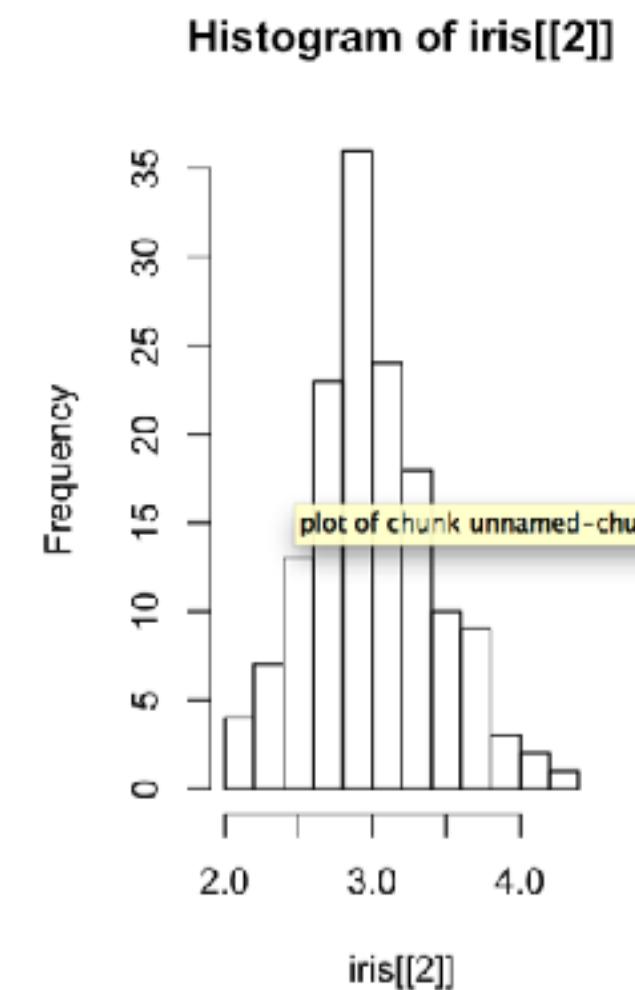
Specify the dimension of plots (in inches) with `fig.width` and `fig.height`. Separate multiple arguments with commas.

Here's a plot

```
```{r echo=FALSE, fig.width=3, fig.height=5}
hist(iris[[2]])
````
```



Here's a plot



# Pop Quiz

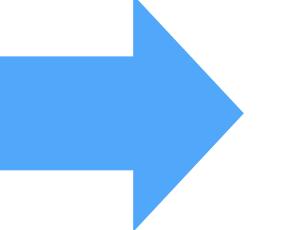
Do you notice the TODOs in  
05-RMarkdown-Exercises.Rmd?

Do you notice the long setup chunk?

# Inline code

Place code in a sentence with `r <code>`. R Markdown will replace the code with its results.

Today is  
`r Sys.Date()`.



Today is 2015-04-16.

# Inline code

Code whose results are inserted into the text.

Today is `r Sys.Date()`.

Surround  
with `r`

Code to run. Only the  
result will be included.

# Your Turn 1

In 05-RMarkdown-Exercises.Rmd:

1. Replace every Garrett with your name
2. Replace every TODO with inline R code
3. Check that the setup chunk is not included with the output
4. Ensure that only the output of the plot chunk is shown (not the code)
5. Knit the document



# Parameters

R

# Your Turn 2

Open 05-RMarkdown-Parameters.Rmd.

Click the dropdown menu next to Knit and use **Knit with Parameters** to render the document.

What happens if you type in a different name?

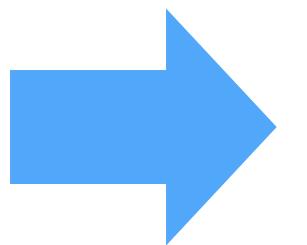


# YAML

A section of key:value pairs  
separated by dashed lines ----

```
---  
title: "Untitled"  
author: "RStudio"  
date: "February 4, 2015"  
output: html_document  
---
```

Text of document



**Untitled**

*RStudio*

***February 4, 2015***

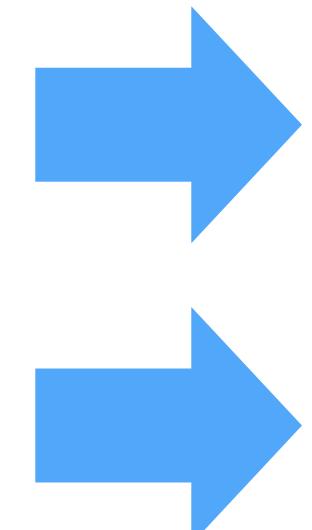
Text of document



# Parameters

A list of values that you can call in R code chunks

**params list**  
**elements and  
values**



```
---
```

```
title: "Untitled"
```

```
output: html_document
```

```
params:
```

```
  filename: "data.csv"
```

```
  symbol: "FB"
```

```
---
```

colon

New line.  
Indented two  
spaces

# Using Parameters

Call parameter values as elements of the params list, **params\$num**

```
---
```

```
params:
```

```
  num: 42
```

```
---
```

```
The value of the parameter is `r params$num`, e.g.
```

```
```{r}
```

```
params$num
```

```
```
```

The value of the parameter is 42, e.g.

```
params$num
```

```
## [1] 42
```

# One final thing

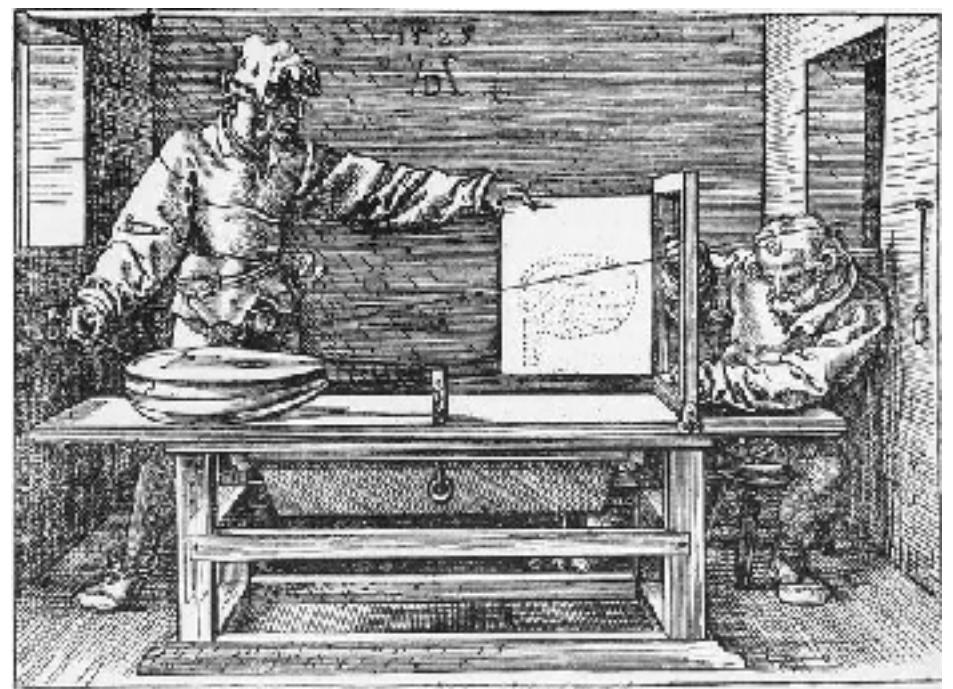


Code is an  
opportunity

# Report Reproducibly with



# Thank you



Please take the class survey  
[rstd.io/class-survey](https://rstd.io/class-survey)