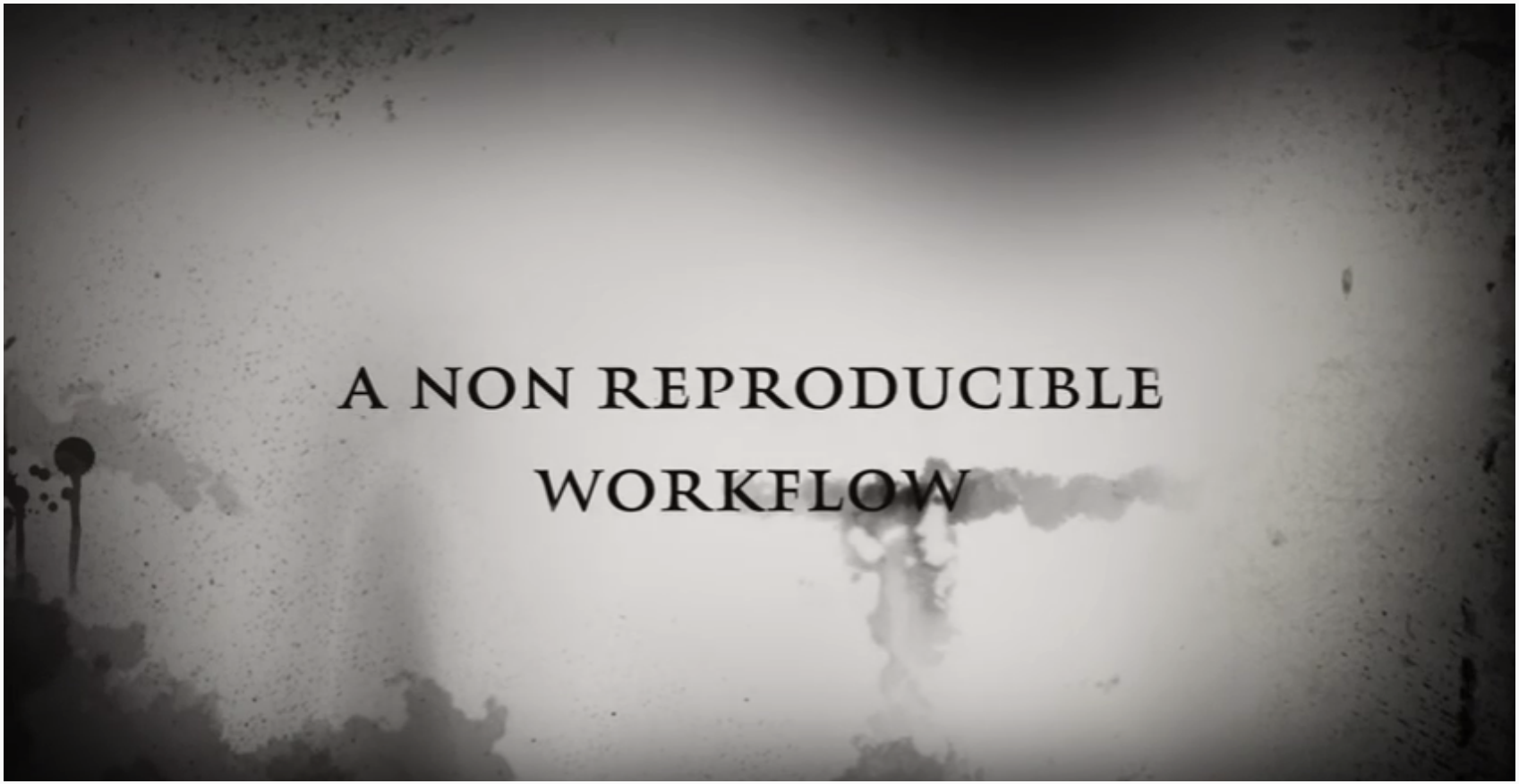
Reproducible dynamic documents with Rmarkdown

Francisco Rodríguez-Sánchez

@frod_san

https://frodriguezsanchez.net

A scary movie... with happy ending

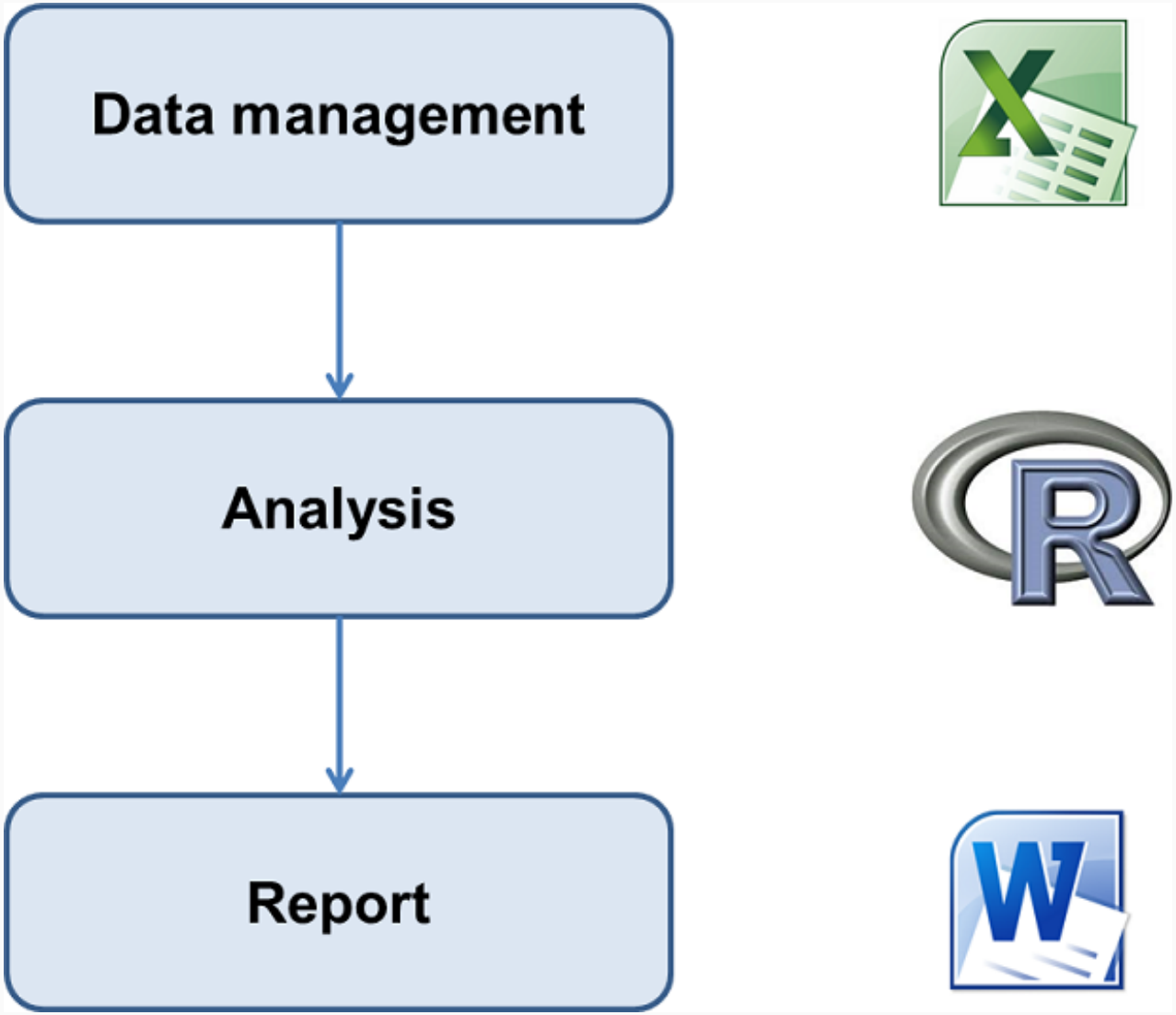


https://youtu.be/s3JldKoA0zw

A typical research workflow

1. Prepare data (**spreadsheet**)
2. Analyse data (**R**)
3. Write report/paper (**Word**)
4. Start the email attachments nightmare...

This workflow is broken



Problems of a broken workflow

- How did you do this? What analysis is behind this figure? Did you account for ...?

Problems of a broken workflow

- How did you do this? What analysis is behind this figure? Did you account for ...?
- What dataset was used? Which individuals were left out? Where is the clean dataset?

Problems of a broken workflow

- How did you do this? What analysis is behind this figure? Did you account for ...?
- What dataset was used? Which individuals were left out? Where is the clean dataset?
- Oops, there is an error in the data. Can you repeat the analysis? And update figures/tables in Word!

Manual copy-paste is tedious & problematic

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.0651657  0.4264970  -0.153    0.879
sunshine     0.0100228  0.0004232  23.683   <2e-16
```

'Transcribing numbers **from stats software by hand** was the largest source of errors'

(Eubank 2016)

Trevor A. Branch
@TrevorABranch

Follow

My rule of thumb: every analysis you do on a dataset will have to be redone 10–15 times before publication. Plan accordingly. #Rstats

Your **closest collaborator** is you 6 months ago,

and you don’t respond to emails.

(P. Wilson)

Even **you** will struggle to reproduce

your own results from a few weeks/months ago.

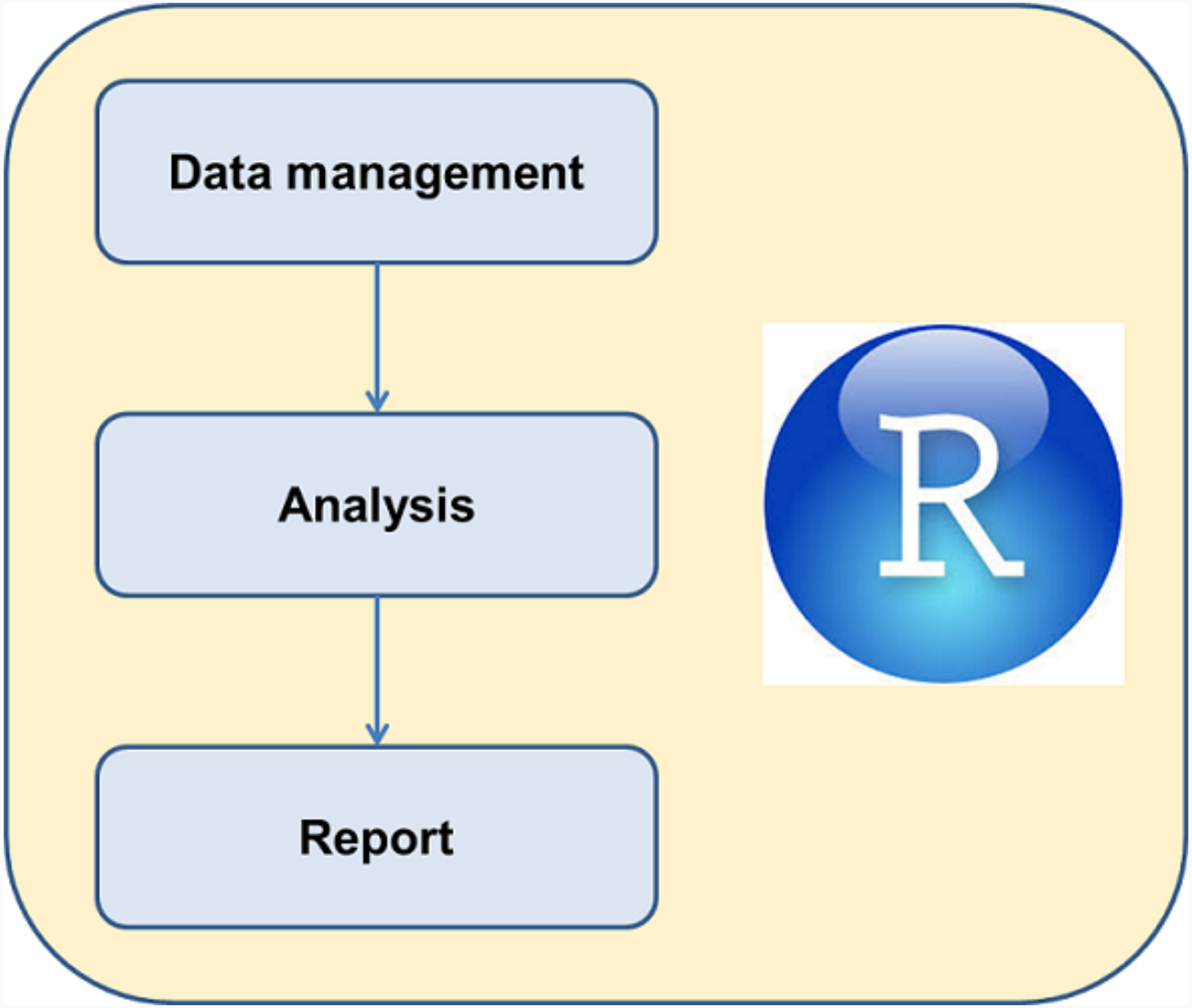
Writing reproducible manuscripts is hard

Revising non-reproducible manuscripts is even harder

.

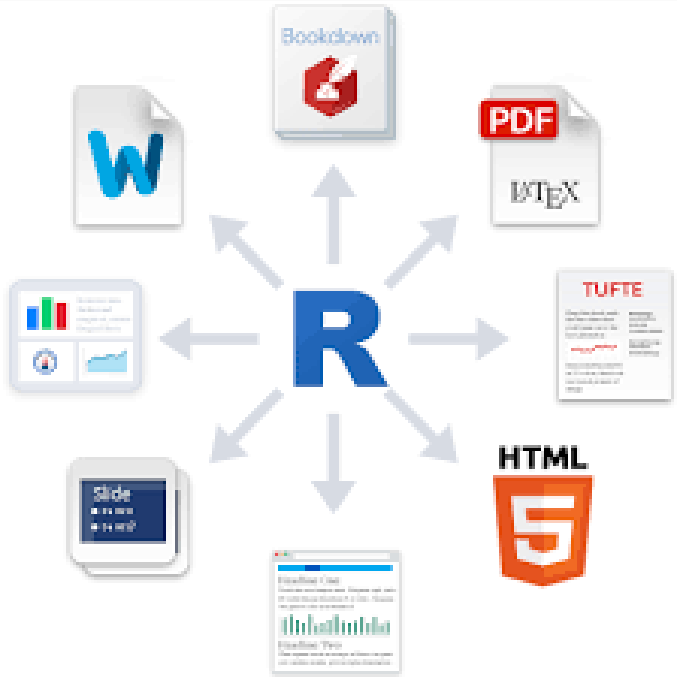
Also, please note that because rev#1 asked to re-calculate effect sizes (...) we need to change every single number in the main text.

Dynamic reports

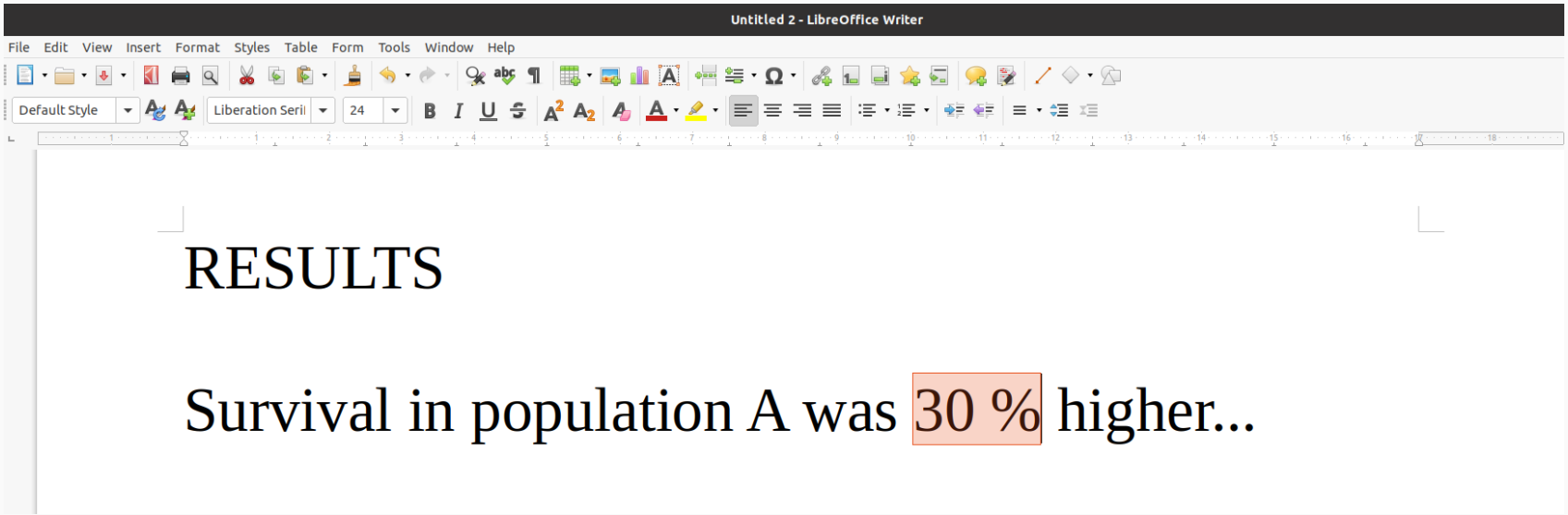


Rmarkdown documents

- **Fully reproducible** (trace all results inc. tables and plots)
- **Dynamic** (regenerate with 1 click)
- **Multiple outputs**:
  - documents (HTML, Word, PDF)
  - presentations (HTML, PDF, PowerPoint)
  - books
  - websites...



Where does this value come from?



Dynamic documents with Rmarkdown

Rmarkdown:

> Survival in population A was `` `r surv.diff` `` % higher

Output:

> Survival in population A was **30** % higher

Dynamic documents with Rmarkdown

```
mydata <- read.csv("data.txt")
```

Rmarkdown:

> We measured `` `r nrow(mydata)` `` individuals

Output:

> We measured **100** individuals

> *Much better than copy-paste!*

Rmarkdown: code (R, Python, etc) + text (Markdown)

````
---
title: "Does sunshine make people happy?"
author: "FRS"
output: word_document
---

## Introduction

It is well known that individual well-being can be influenced by climatic conditions.


## Methods

```{r echo=FALSE}
## Read data
data <- read.table("data.txt", header = TRUE)

# Fit linear model
model <- lm(happiness ~ sunshine, data = data)
```

We collected data on `r nrow(data)` individuals and fitted a linear model.
````

Metadata
(YAML)

Text
(Markdown)

Code
(R, Python...)

Code chunk options

````
```{r echo=FALSE, eval=TRUE, cache=TRUE, fig.height=3}
plot(iris)
```
````

https://yihui.org/knitr/options/

Code chunk options

````
```{r}
#| echo = FALSE
#| eval = TRUE
#| fig.cap = "My figure caption"
plot(iris)
```
````

Naming chunks helps debugging

```
processing file: test.Rmd
  |.........                                                       |  14%
  ordinary text without R code

  |.....................                                           |  29%
label: setup (with options)
List of 1
 $ include: logi FALSE

  |.............................                                   |  43%
  ordinary text without R code

  |.....................................                           |  57%
label: read.data
  |.............................................                   |  71%
  ordinary text without R code

  |.................................................................|  86%
label: plot (with options)
List of 1
 $ echo: logi FALSE


Quitting from lines 28-29 (test.Rmd)
Error in eval(predvars, data, env) : object 'specie' not found
Calls: <Anonymous> ... plot.formula -> eval -> eval -> <Anonymous> -> eval -> eval
Execution halted
```

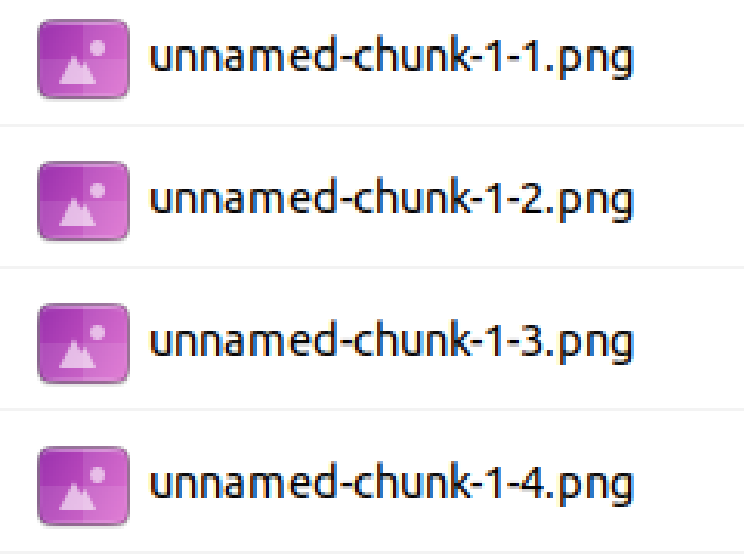
Naming chunks helps navigating long docs

```
1  ---
2  title: "My Analysis"
3  author: "FRS"
4  output: html_document
5  ---
6
7  ```{r setup, include=FALSE}
8  knitr::opts_chunk$set(echo = TRUE)
9  ```
10
11 This is an R Markdown document. Markdown is a simple
                  for authoring HTML, PDF, and MS Word
                re details on using R Markdown see
                .rstudio.com>.
12
```

My Analysis
Chunk 1: setup
Chunk 2: read.data
Chunk 3: plot

11:60 (Top Level) R Markdown

Naming chunks: figure files take chunk name



¡Not only R! Python, Julia, C++, SQL, Stan, etc

`knitr` engines:

```
 [1] "asis"       "asy"        "awk"        "bash"       "block"      "block2"
 [7] "bslib"      "c"          "cat"        "cc"         "coffee"     "comment"
[13] "css"        "ditaa"      "dot"        "embed"      "eviews"     "exec"
[19] "fortran"    "fortran95"  "gawk"       "go"         "groovy"     "haskell"
[25] "highlight"  "js"         "julia"      "lein"       "mysql"      "node"
[31] "octave"     "perl"       "php"        "psql"       "python"     "R"
[37] "Rcpp"       "Rscript"    "ruby"       "sas"        "sass"       "scala"
[43] "scss"       "sed"        "sh"         "sql"        "stan"       "stata"
[49] "targets"    "tikz"       "verbatim"   "zsh"
```

Markdown: easy text formatting

```
# Header

## Subheader

*italic*

**bold**

[a link](https://example.com)
```

.

Handy: https://thinkr-open.github.io/remedy/

Or use Visual Markdown Editor

Regenerate Word/PDF/HTML with one click

a

```
---
title: "Does sunshine make people happy?"
output: pdf_document
bibliography: refs.bib
---

# Introduction

Climate influences individual well-being [@Rehdanz_2005].
However, ...

# Methods

```{r echo=FALSE}
# read data
data <- read.table("data.txt", header=T)
data[10,1] <- 11   # correct error

# fit linear model
model <- lm(happiness ~ sunshine, data=data)
```

We collected data on `r nrow(data)` individuals and fitted a
linear model.

Results

We found that...

```{r echo=FALSE, results='asis'}
# make table with model output
print(xtable::xtable(model), comment = FALSE)
```

```{r echo=FALSE, fig.height=3, fig.width=3, fig.align='center'}
visreg::visreg(model) # plot
```

Discussion

Our results confirm that happiness is related to
sunshine (slope = `r coef(model)[2]`).

References
```

b

Does sunshine make people happy?

Introduction

Climate influences individual well-being (Rehdanz and Maddison 2005). However, ...

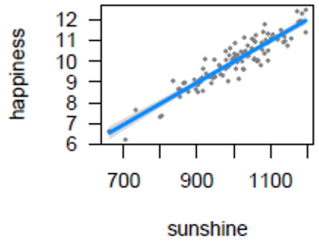
Methods

We collected data on 100 individuals and fitted a linear model.

Results

We found that...

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-0.0986	0.4271	-0.23	0.8180
sunshine	0.0101	0.0004	23.75	0.0000



Discussion

Our results confirm that happiness is related to sunshine (slope = 0.0100652).

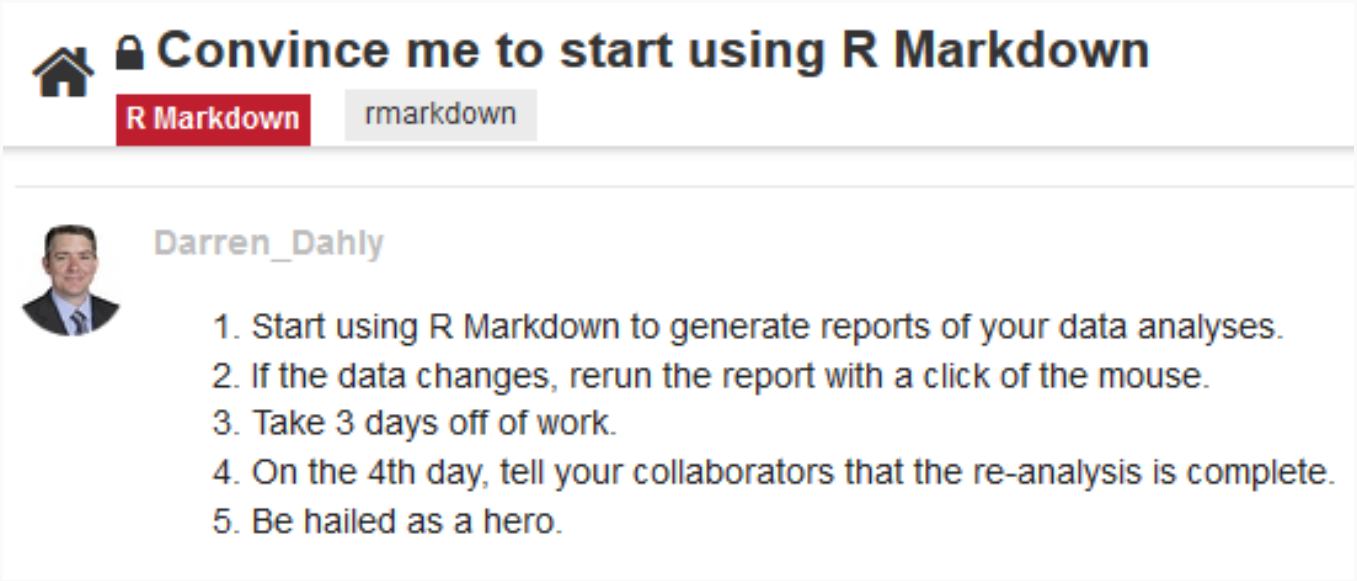
References

Rehdanz, Katrin, and David Maddison. 2005. "Climate and Happiness." *Ecological Economics* 52 (1). Elsevier BV: 111–25. doi:10.1016/j.ecolecon.2004.06.015.

Spotted error in the data? No problem!

- Make changes in Rmarkdown document
- Click `Knit` in Rstudio
- Report will **update automatically!**

Why Rmarkdown?



https://community.rstudio.com/t/convince-me-to-start-using-r-markdown/1636/12

Your turn

Create, edit and share Rmarkdown document

File > New File > Rmarkdown

Write text

Insert code chunks

Change chunk options (echo, eval, etc)

HTML/Word/PDF output

PDF generation requires LaTeX

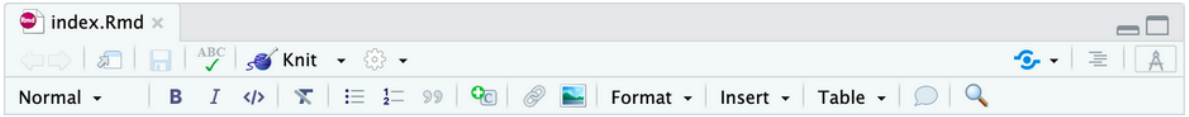
```
library('tinytex')

install_tinytex()
```

Rmarkdown bells and whistles

‘Visual Rmarkdown’: Rmd as in word processor

The editor toolbar includes buttons for the most commonly used formatting commands:



Additional commands are available on the **Format**, **Insert**, and **Table** menus:

Format

- Bold ⌘B
- Italic ⌘I
- Code ⌘D
- Text ▸
- Bullets & Numbering ▸
- Blockquote
- Line Block
- Div Block...
- Code Block...
- Raw ▸
- Clear Formatting ⌘\
- Edit Attributes... F4

Insert

- Rmd Chunk ⌥⌘I
- Image... ⇧⌘I
- Link... ⌘K
- Horizontal Rule ⌘_
- Definition ▸
- Inline Math
- Display Math
- Footnote ⇧⌘F7
- Citation...
- Div Block...
- Code Block...
- YAML Block
- Comment ⇧⌘C

Table

- Insert Table... ⌥⌘T
- ✓ Table Header
- Table Caption
- Align Column ▸
- Insert Row Above
- Insert Row Below
- Insert Column Left
- Insert Column Right
- Delete Row
- Delete Column
- Delete Table

https://rstudio.github.io/visual-markdown-editing

Automatic table generation

```
model <- lm(happiness ~ sunshine, data = mydata)
xtable(model)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-0.0652	0.4265	-0.15	0.8789
sunshine	0.0100	0.0004	23.68	0.0000

Many alternatives: `gtsummary`, `modelsummary`, `huxtable`, etc

`equatiomatic` describes model structure

We fitted a linear model:

```
library('equatiomatic')
model <- lm(happiness ~ sunshine, data = mydata)
extract_eq(model)
```

$$\text{happiness} = \alpha + \beta_1(\text{sunshine}) + \epsilon \tag{1}$$

Models that describe themselves!

```
library("report")
model <- lm(happiness ~ sunshine, data = mydata)
report(model)
```

We fitted a linear model (estimated using OLS) to predict happiness with sunshine (formula: happiness ~ sunshine). The model explains a statistically significant and substantial proportion of variance (R2 = 0.85, F(1, 98) = 560.90, p < .001, adj. R2 = 0.85). The model's intercept, corresponding to sunshine = 0, is at -0.07 (95% CI [-0.91, 0.78], t(98) = -0.15, p = 0.879). Within this model:

- The effect of sunshine is statistically significant and positive (beta = 0.01, 95% CI [9.18e-03, 0.01], t(98) = 23.68, p < .001; Std. beta = 0.92, 95% CI [0.85, 1.00])

Standardized parameters were obtained by fitting the model on a standardized version of the dataset. 95%

Confidence Intervals (CIs) and p-values were computed using a Wald t-distribution approximation.

Insert equations with LaTeX

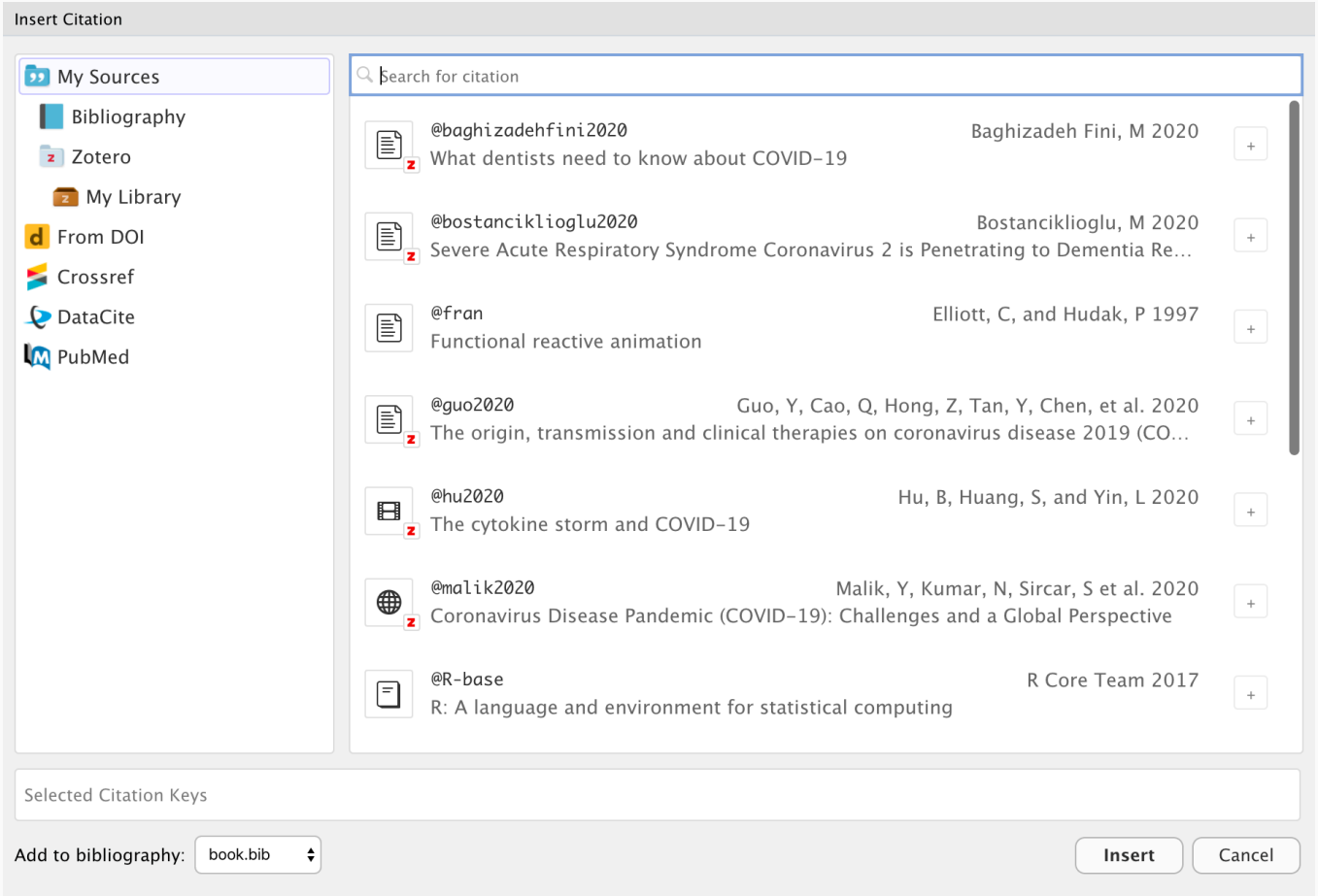
Using LaTeX:

```
$$
y \sim N(\mu, \sigma^2)
$$
```

$$y \sim N(\mu, \sigma^2)$$

- Mathpix: https://github.com/jonocarroll/mathpix

Citing bibliography



https://rstudio.github.io/visual-markdown-editing/#/citations

Using BibTeX file with references

```
---
title: "My awesome Rmd"
output: html_document
bibliography: references.bib
---
```

Format bibliography for any journal

```
---
title: "Does sunshine make people happy?"
author: "FRS"
output: word_document
bibliography: myrefs.bib
csl: ecology-letters.csl
---
```

Thousands of Citation Styles:

https://www.zotero.org/styles

https://github.com/citation-style-language/styles

Rmarkdown templates

- rticles
- papaja
- rrtools
- pinp
- rmdTemplates
- pagedreport
- GitHub!

My cool paper written in Rmarkdown

F. Rodriguez-Sanchez[a,1,2] and And Friends[a,b]

[a]Some Institute of Technology, Department, Street, City, State, Zip; [b]Another University Department, Street, City, State, Zip

This manuscript was compiled on September 18, 2018

Please provide an abstract of no more than 250 words in a single paragraph. Abstracts should explain to the general reader the major contributions of the article. References in the abstract must be cited in full within the abstract itself and cited in the text.

one | two | optional | optional | optional

This PNAS journal template is provided to help you write your work in the correct journal format. Instructions for use are provided below.

Note: please start your introduction without including the word "Introduction" as a section heading (except for math articles in the Physical Sciences section); this heading is implied in the first paragraphs.

Guide to using this template

Please note that whilst this template provides a preview of the typeset manuscript for submission, to help in this preparation, it will not necessarily be the final publication layout. For more detailed information please see the PNAS Information for Authors.

Author Affiliations. Include department, institution, and complete address, with the ZIP/postal code, for each author. Use lower case letters to match authors with institutions, as shown in the example. Authors with an ORCID ID may supply this information at submission.

Submitting Manuscripts. All authors must submit their articles at PNAScentral. If you are using Overleaf to write your article, you can use the "Submit to PNAS" option in the top bar of the editor window.

Format. Many authors find it useful to organize their manuscripts with the following order of sections; Title, Author Affiliation, Keywords, Abstract, Significance Statement, Results, Discussion, Materials and methods, Acknowledgments, and References. Other orders and headings are permitted.

Manuscript Length. PNAS generally uses a two-column format averaging 67 characters, including spaces, per line. The maximum length of a Direct Submission research article is six pages and a PNAS PLUS research article is ten pages including all text, spaces, and the number of characters displaced by figures, tables, and equations. When submitting tables, figures, and/or equations in addition to text, keep the text for your manuscript under 39,000 characters (including spaces) for Direct Submissions and 72,000 characters (including spaces) for PNAS PLUS.

References. References should be cited in numerical order as they appear in text; this will be done automatically via bibtex, e.g. (1) and (2, 3). All references, including for the SI, should be included in the main manuscript file. References appearing in both sections should not be duplicated. SI references

Fig. 1. Placeholder image of a frog with a longer example caption to show justification setting.

included in tables should be included with the main reference section.

Data Archival. PNAS must be able to archive the data essential to a published article. Where such archiving is not possible, deposition of data in public databases, such as GenBank, ArrayExpress, Protein Data Bank, Unidata, and others outlined in the Information for Authors, is acceptable.

Language-Editing Services. Prior to submission, authors who believe their manuscripts would benefit from professional editing are encouraged to use a language-editing service (see list at www.pnas.org/site/authors/language-editing.xhtml). PNAS does not take responsibility for or endorse these services, and their use has no bearing on acceptance of a manuscript for publication.

Significance Statement

Authors must submit a 120-word maximum statement about the significance of their research paper written at a level understandable to an undergraduate educated scientist outside their field of speciality. The primary goal of the Significance Statement is to explain the relevance of the work in broad context to a broad readership. The Significance Statement appears in the paper itself and is required for all research papers.

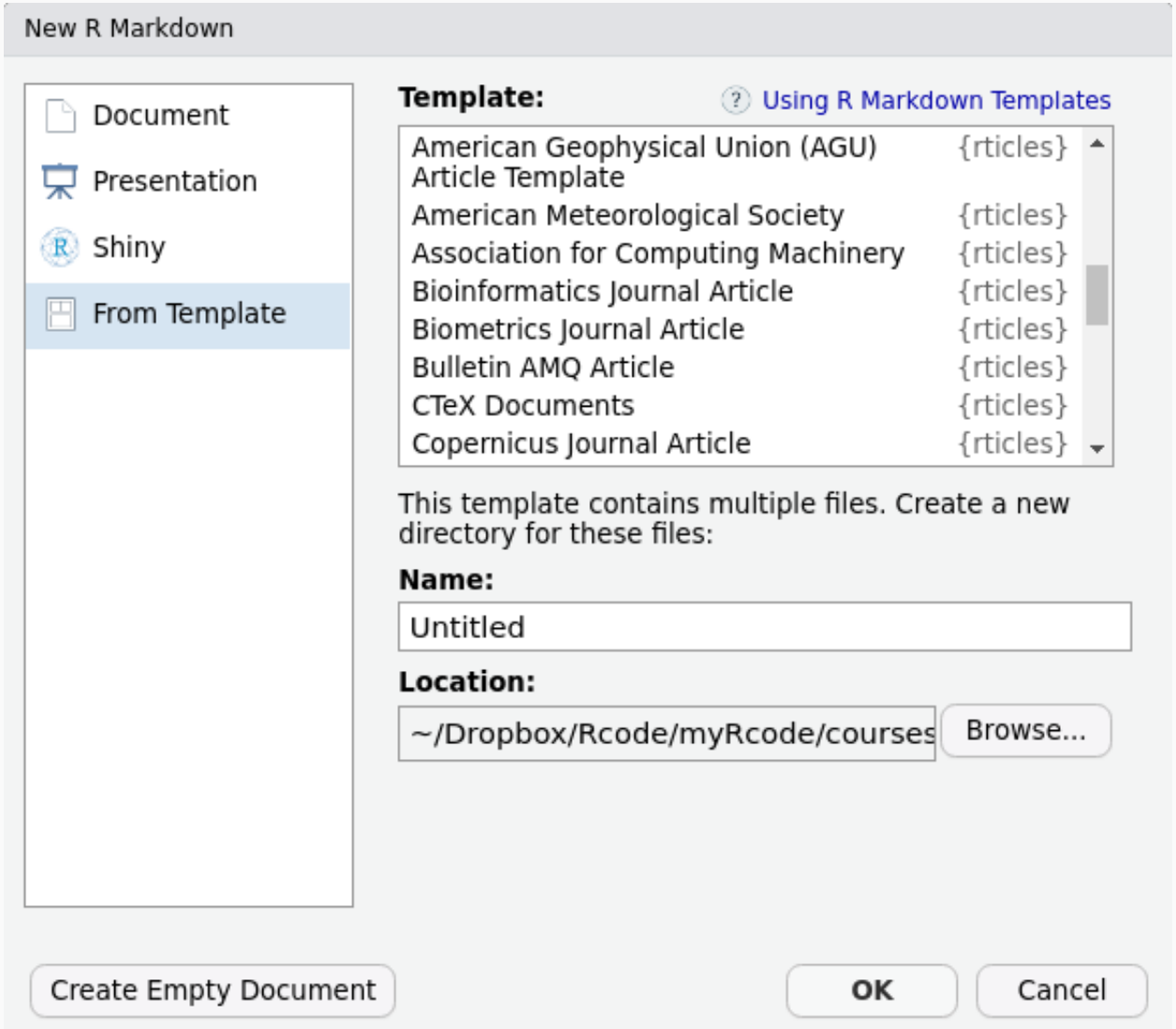
Please provide details of author contributions here.

Please declare any conflict of interest here.

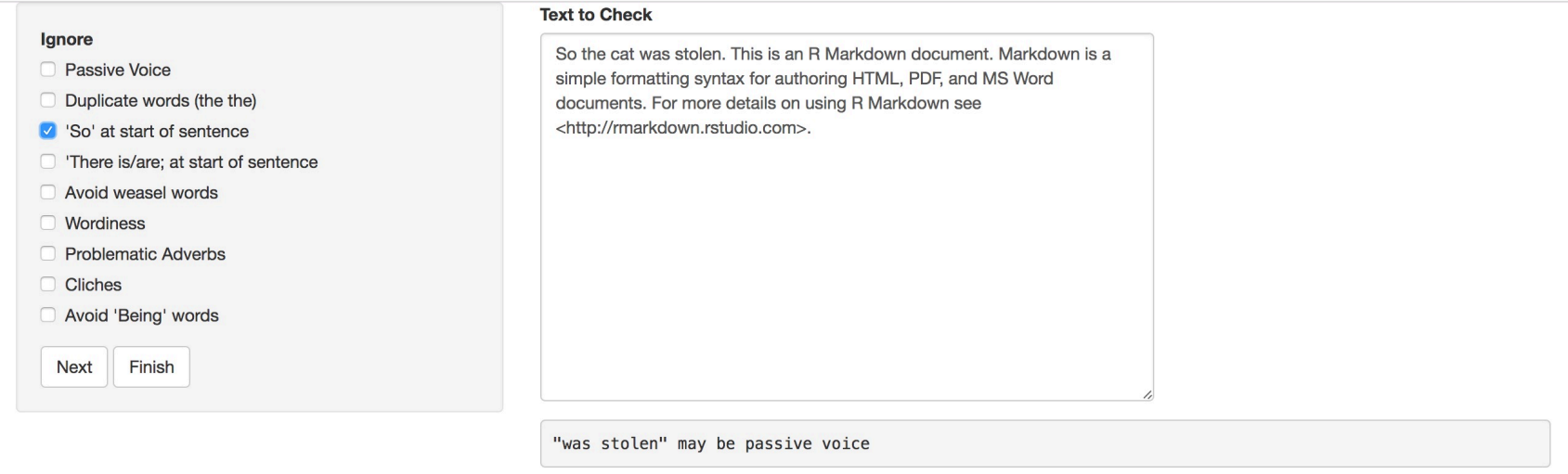
www.pnas.org/cgi/doi/10.1073/pnas.XXXXXXXXXX

PNAS | September 18, 2018 | vol. XXX | no. XX | 1–2

Accessing Rmd templates



Revise writing style: gramr

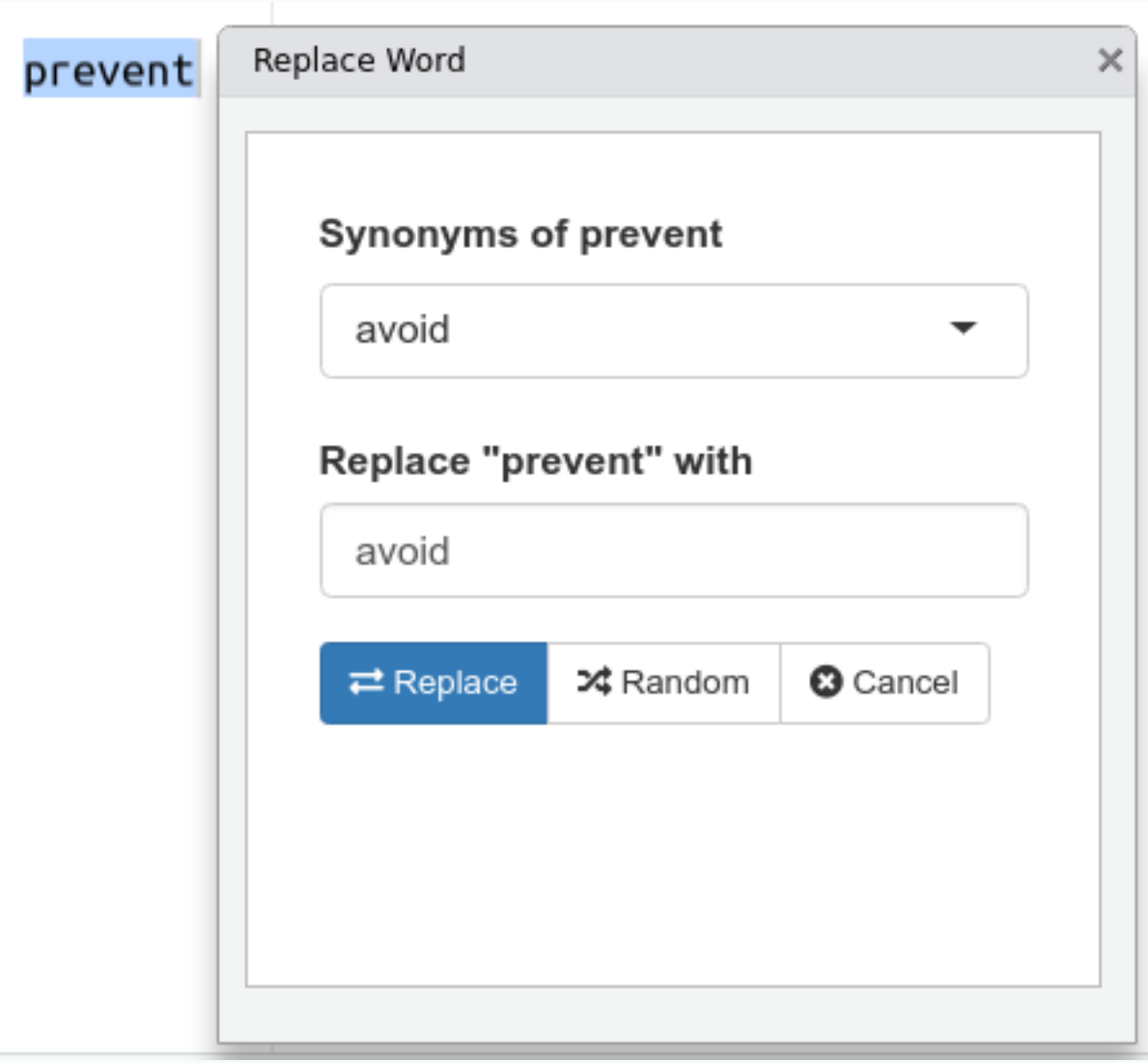


https://github.com/ropenscilabs/gramr

Spell and grammar checking

https://github.com/nevrome/wellspell.addin

Find synonyms



https://github.com/gadenbuie/synamyn

Word count and readability

Method	koRpus	stringi
Word count	107	104
Character count	604	603
Sentence count	10	Not available
Reading time	0.5 minutes	0.5 minutes

https://github.com/benmarwick/wordcountaddin

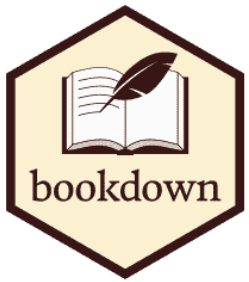
Write books, theses, with **bookdown**

BOOKDOWN

Write HTML, PDF, ePub, and Kindle books with R Markdown

The **bookdown** package is an open-source R package that facilitates writing books and long-form articles/reports with R Markdown. Features include:

- Generate printer-ready books and ebooks from R Markdown documents.
- A markup language easier to learn than LaTeX, and to write elements such as section headers, lists, quotes, figures, tables, and citations.
- Multiple choices of output formats: PDF, LaTeX, HTML, EPUB, and Word.
- Possibility of including dynamic graphics and interactive applications (HTML widgets and Shiny apps).
- Support a wide range of languages: R, C/C++, Python, Fortran, Julia, Shell scripts, and SQL, etc.
- LaTeX equations, theorems, and proofs work for all output formats.
- Can be published to GitHub, bookdown.org, and any web servers.
- Integrated with the RStudio IDE.
- One-click publishing to https://bookdown.org.



https://bookdown.org/

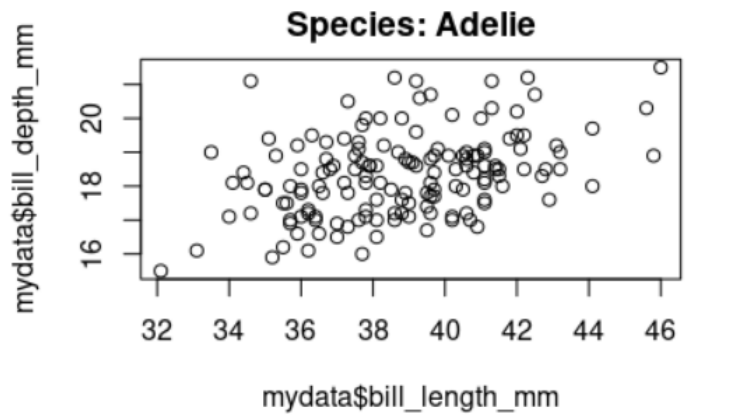
Slide presentations with `xaringan`

https://slides.yihui.org/xaringan/

Parameterised reports

```
---
title: "My template report"
output: html_document
params:
  sp: Adelie
---
```

```{r}
library(palmerpenguins)
data("penguins")

mydata <- subset(penguins, species == params$sp)

plot(mydata$bill_length_mm, mydata$bill_depth_mm,
     main = paste0("Species: ", params$sp))
```



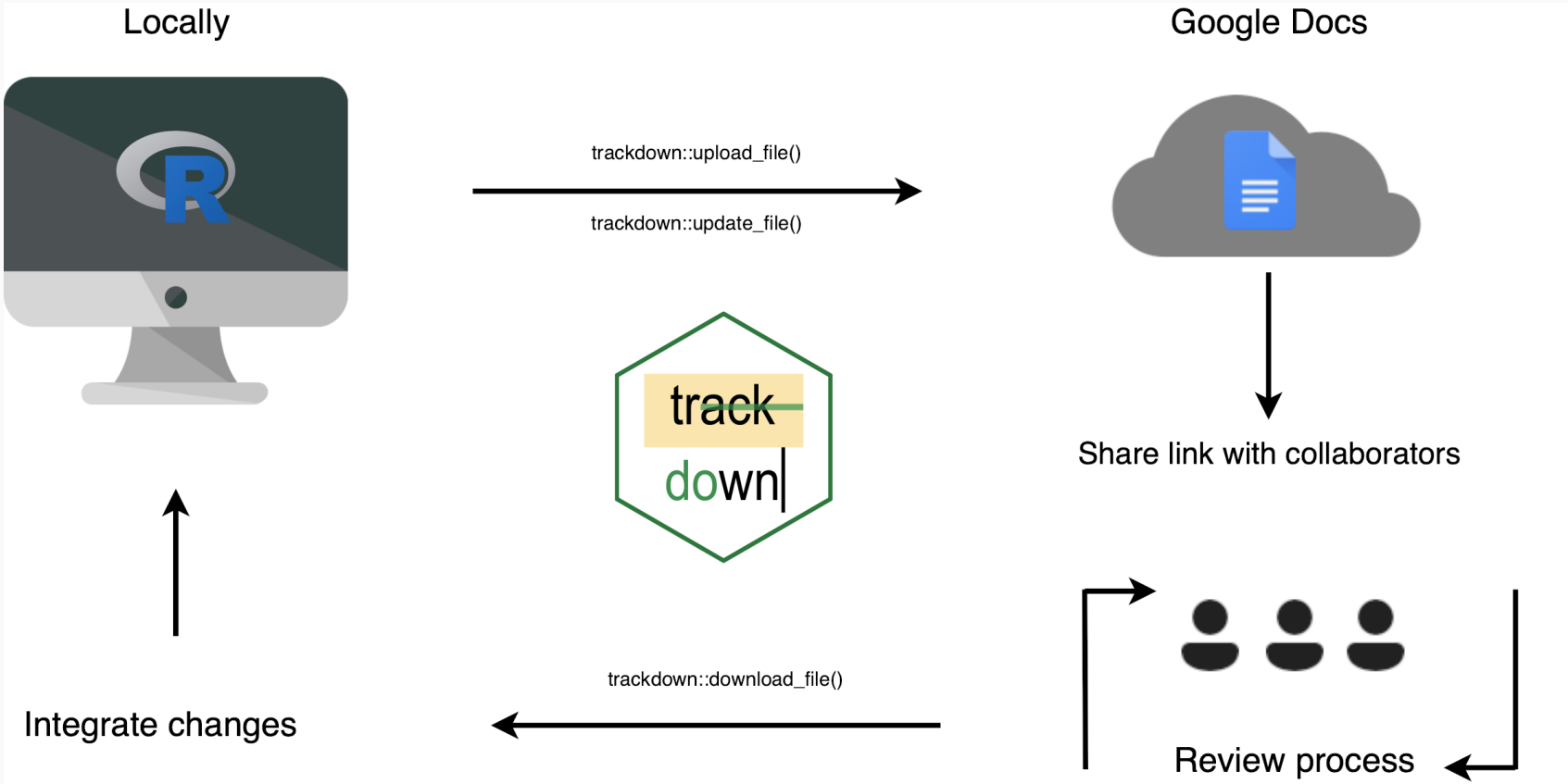
https://bookdown.org/yihui/rmarkdown/parameterized-reports.html

Render thousands of individual reports from Rmd template

```
library('rmarkdown')

for (i in unique(penguins$species)) {

  render("template_report.Rmd",
         params = list(sp = i))

}
```

Collaborative writing

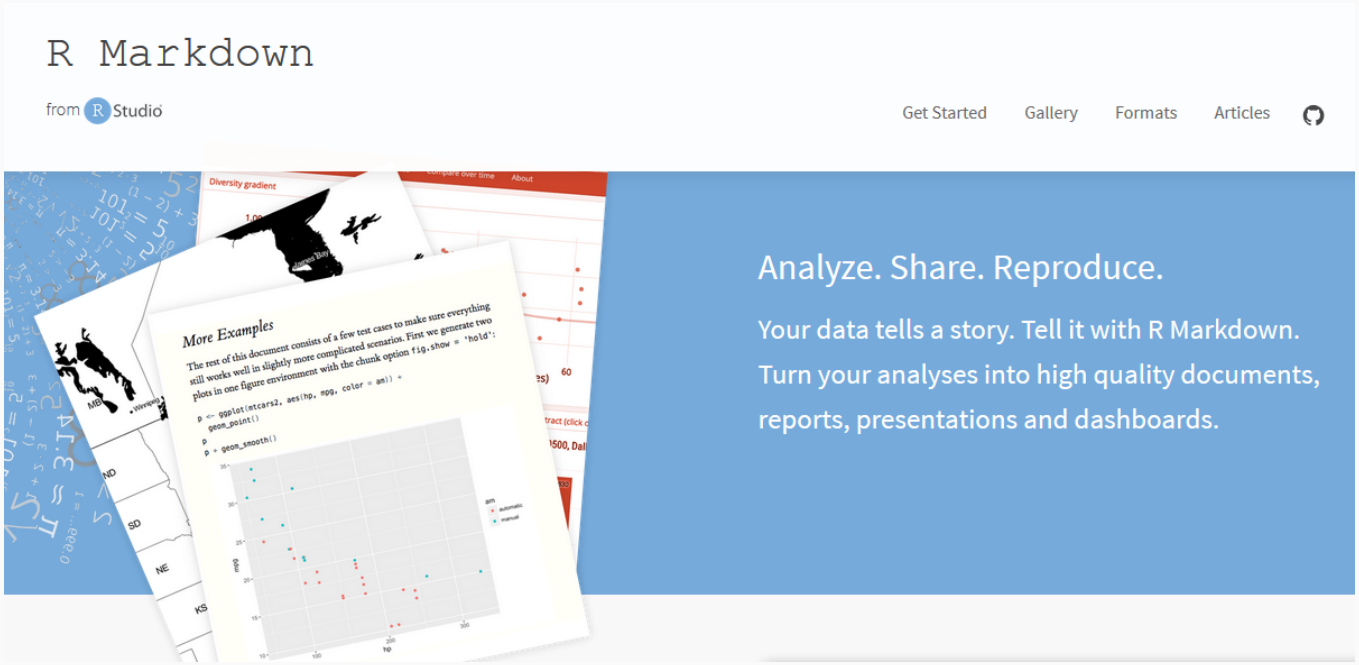
- GitHub, GitLab, etc
- Google Docs (trackdown)
- redoc



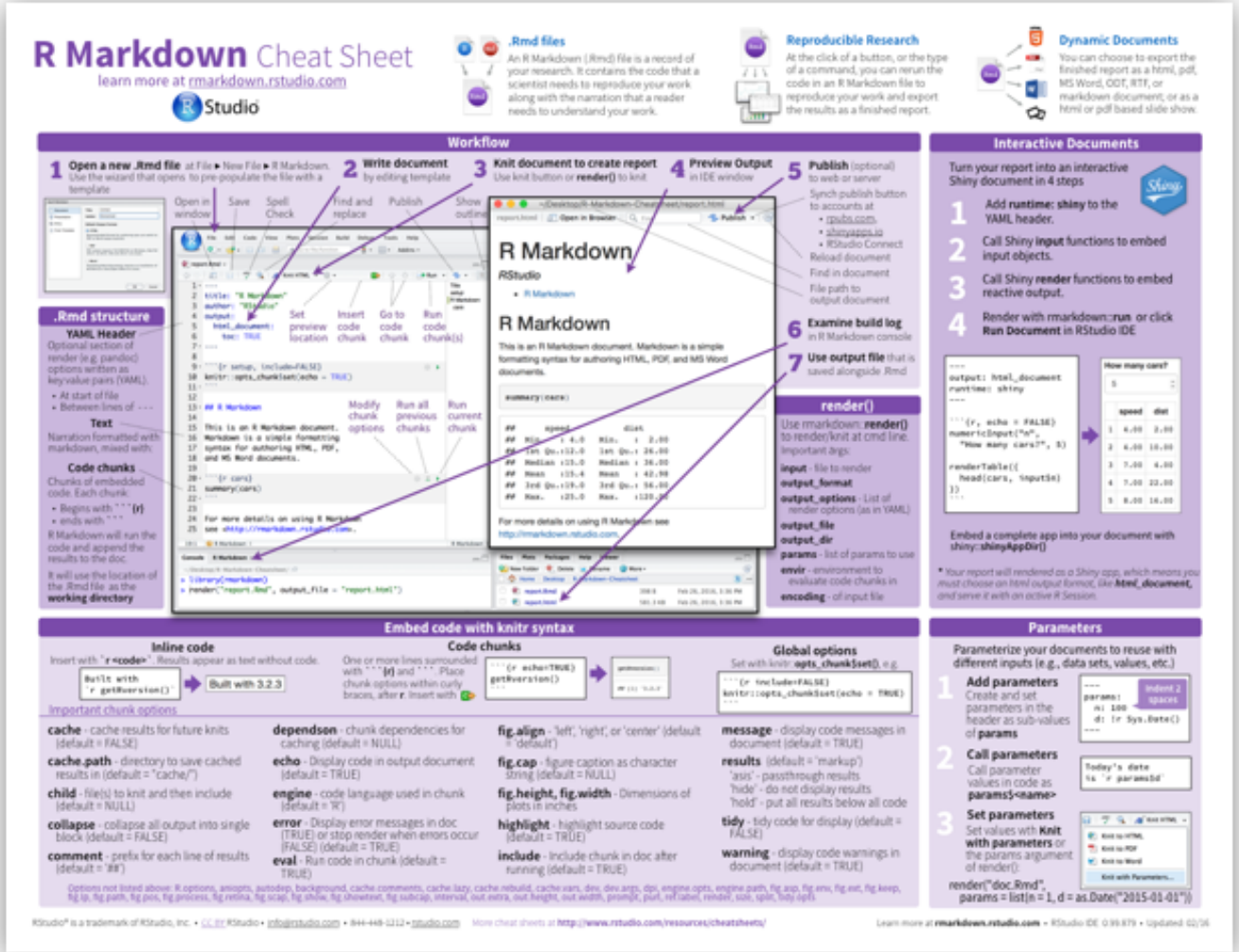
Rmarkdown resources

Rmarkdown website

http://rmarkdown.rstudio.com/



Rmarkdown cheat sheet



https://www.rstudio.org/links/r_markdown_cheat_sheet

Rmarkdown reference guide

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
Make a code chunk with three back ticks followed by an r in braces. End the chunk with three back ticks: ```{r} paste("Hello", "World!") ```	Make a code chunk with three back ticks followed by an r in braces. End the chunk with three back ticks: `paste("Hello", "World!")` `## [1] "Hello World!"`
Place code inline with a single back ticks. The first back tick must be followed by an R, like this `r paste("Hello", "World!")`.	Place code inline with a single back ticks. The first back tick must be followed by an R, like this Hello World!.
Add chunk options within braces. For example, `echo=FALSE` will prevent source code from being displayed: ```{r eval=TRUE, echo=FALSE} paste("Hello", "World!") ```	Add chunk options within braces. For example, `echo=FALSE` will prevent source code from being displayed: `## [1] "Hello World!"`

Learn more about chunk options at http://yihui.name/knitr/options

Chunk options

option	default value	description
Code evaluation		
child	NULL	A character vector of filenames. Knitr will knit the files and place them into the main document.
code	NULL	Set to R code. Knitr will replace the code in the chunk with the code in the code option.
engine	'R'	Knitr will evaluate the chunk in the named language, e.g. **engine = 'python'**. Run **names(knitr::knit_engines$get())** to see supported languages.
eval	TRUE	If **FALSE**, knitr will not run the code in the code chunk.
include	TRUE	If **FALSE**, knitr will run the chunk but not include the chunk in the final document.
purl	TRUE	If **FALSE**, knitr will not include the chunk when running **purl()** to extract the source code.
Results		
collapse	FALSE	If **TRUE**, knitr will collapse all the source and output blocks created by the chunk into a single block.
echo	TRUE	If **FALSE**, knitr will not display the code in the code chunk above it's results in the final document.
results	'markup'	If **'hide'**, knitr will not display the code's results in the final document. If **'hold'**, knitr will delay displaying all output pieces until the end of the chunk. If **'asis'**, knitr will pass through results without reformatting them (useful if results return raw HTML, etc.)
error	TRUE	If **FALSE**, knitr will not display any error messages generated by the code.
message	TRUE	If **FALSE**, knitr will not display any messages generated by the code.
warning	TRUE	If **FALSE**, knitr will not display any warning messages generated by the code.
Code Decoration		
background	'#F7F7F7'	A background color for chunks in LaTex output.
comment	'##'	A character string. Knitr will append the string to the start of each line of results in the final document.
highlight	TRUE	If **TRUE**, knitr will highlight the source code in the final output.
prompt	FALSE	If **TRUE**, knitr will add > to the start of each line of code displayed in the final document.
size	'normalsize'	Fontsize for LaTex output.
strip.white	TRUE	If **TRUE**, knitr will remove white spaces that appear at the beginning or end of a code chunk.
tidy	FALSE	If **TRUE**, knitr will tidy code chunks for display with the **tidy_source()** function in the **formatR** package.

RStudio Updated 10/30/2014

2

https://www.rstudio.org/links/r_markdown_reference_guide

Rmarkdown books

R Markdown: The Definitive Guide

by Yihui Xie, J. J. Allaire, Garrett Grolemund

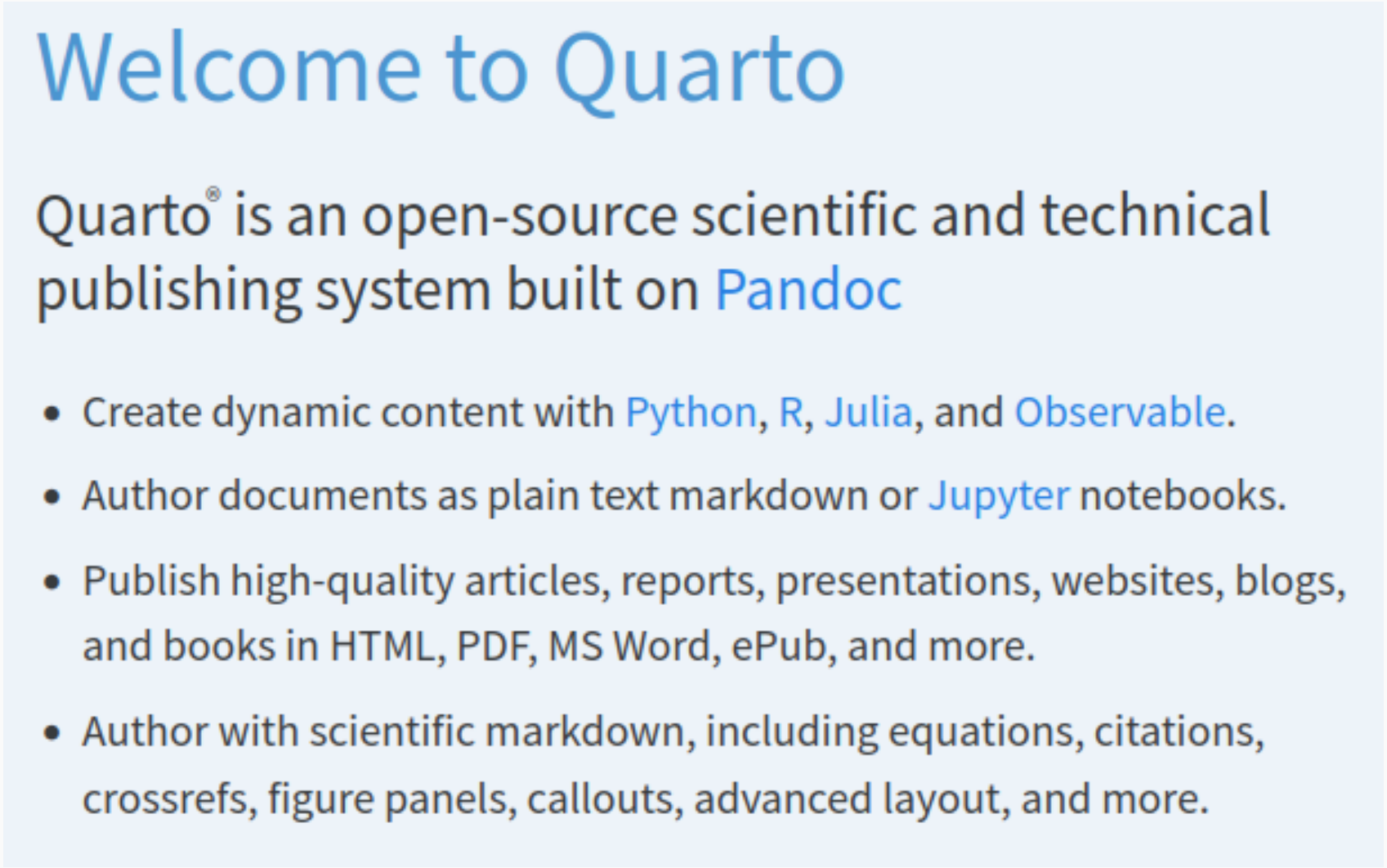
2018-09-11

Star 239

The first official book authored by the core R Markdown developers that provides a comprehensive and accurate reference to the R Markdown ecosystem. With R Markdown, you can easily create reproducible data analysis reports, presentations, dashboards, interactive applications, books, dissertations, websites, and journal articles, while enjoying the simplicity of Markdown and the great power of R and other languages. *Read more* →

https://bookdown.org/yihui/rmarkdown/

https://bookdown.org/yihui/rmarkdown-cookbook/



https://quarto.org/

Your turn

Your turn

- Try visual markdown editor
- Add bibliography
- Try templates (rticles, rmdTemplates)
- Parameterised reports (e.g. different iris or penguin species)