

R COURSE

Markdown and Quarto

Daniel Vaultot

2025-01-17



R - Session 04

- What is Markdown ?
- Rmarkdown syntax
- R chunks
- Some applications
- Quarto - the new Rmarkdown
- Cooperative writing

What is markdown

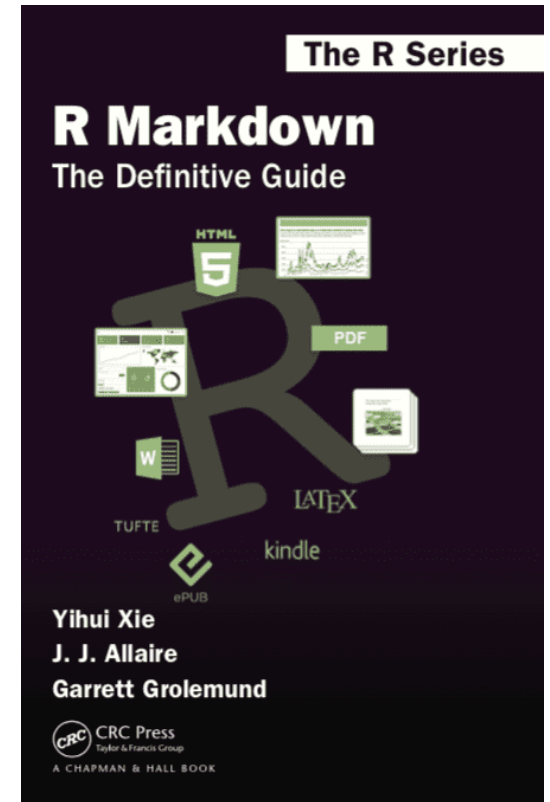
Installation and Resources

Packages

- rmarkdown (will install also knitr)
- tinytex (Latex)

Resources

- [On-line Book](#)
- [Cheat sheet](#)



What is markdown ?

- Created in 2004 by [John Gruber](#) and [Aaron Swartz](#)
- Goal : “to write using an easy-to-read and easy-to-write plain text format, optionally convert it to structurally valid HTML”.

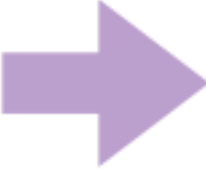
Many flavors...

- MultiMarkdown
- GitHub Flavored Markdown (GFM)
- Pandoc
- CommonMark

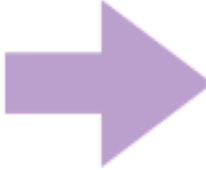


Rmarkdown

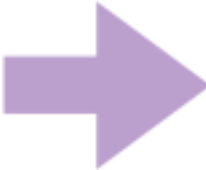
i. Open - Open a file that uses the .Rmd extension.



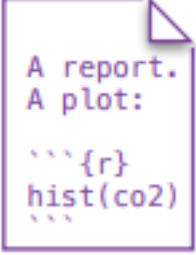
ii. Write - Write content with the easy to use R Markdown syntax



iii. Embed - Embed R code that creates output to include in the report



iv. Render - Replace R code with its output and transform the report into a slideshow, pdf, html or ms Word file.



=



=



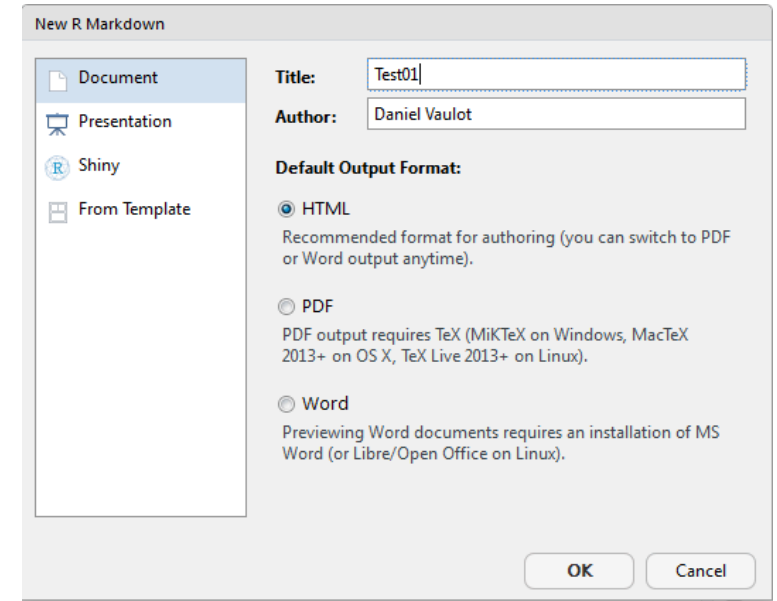
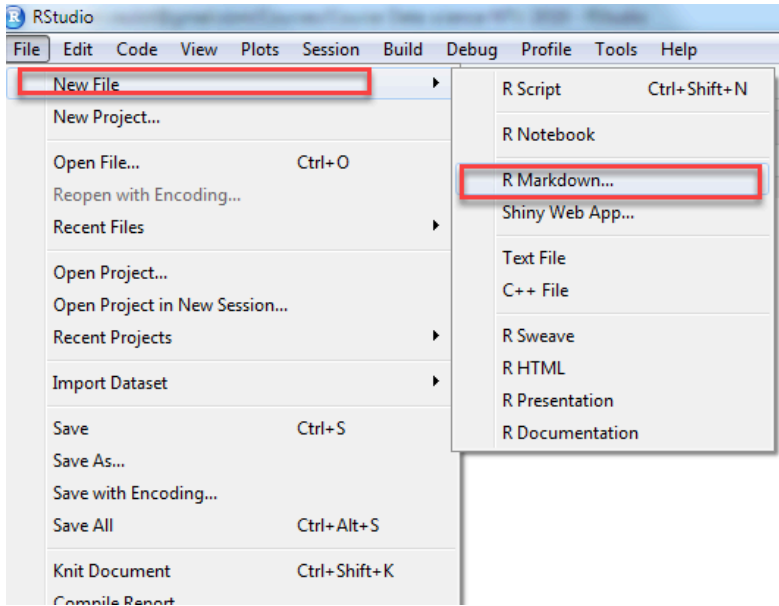
Mix

- Markdown
 - paragraph structure
 - comments
 - links
- R code (“chunks”)
- Output of R code

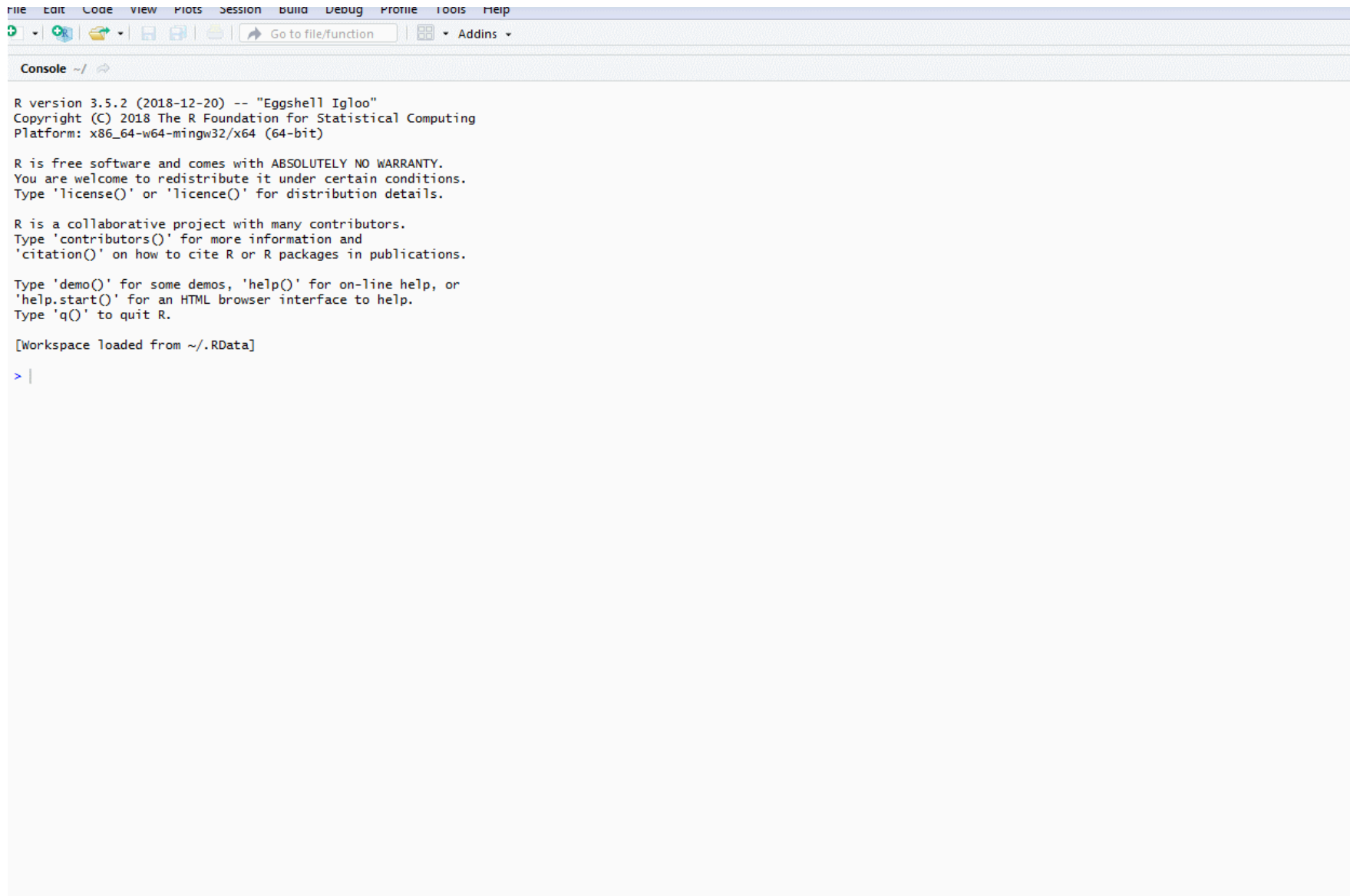
Your first Rmarkdown file

Your first Rmarkdown file

- Who has not been able to install Rmarkdown and Latex ?



Your first Rmarkdown file



```
file Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins
Console ~/
R version 3.5.2 (2018-12-20) -- "Eggshell Igloo"
Copyright (C) 2018 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

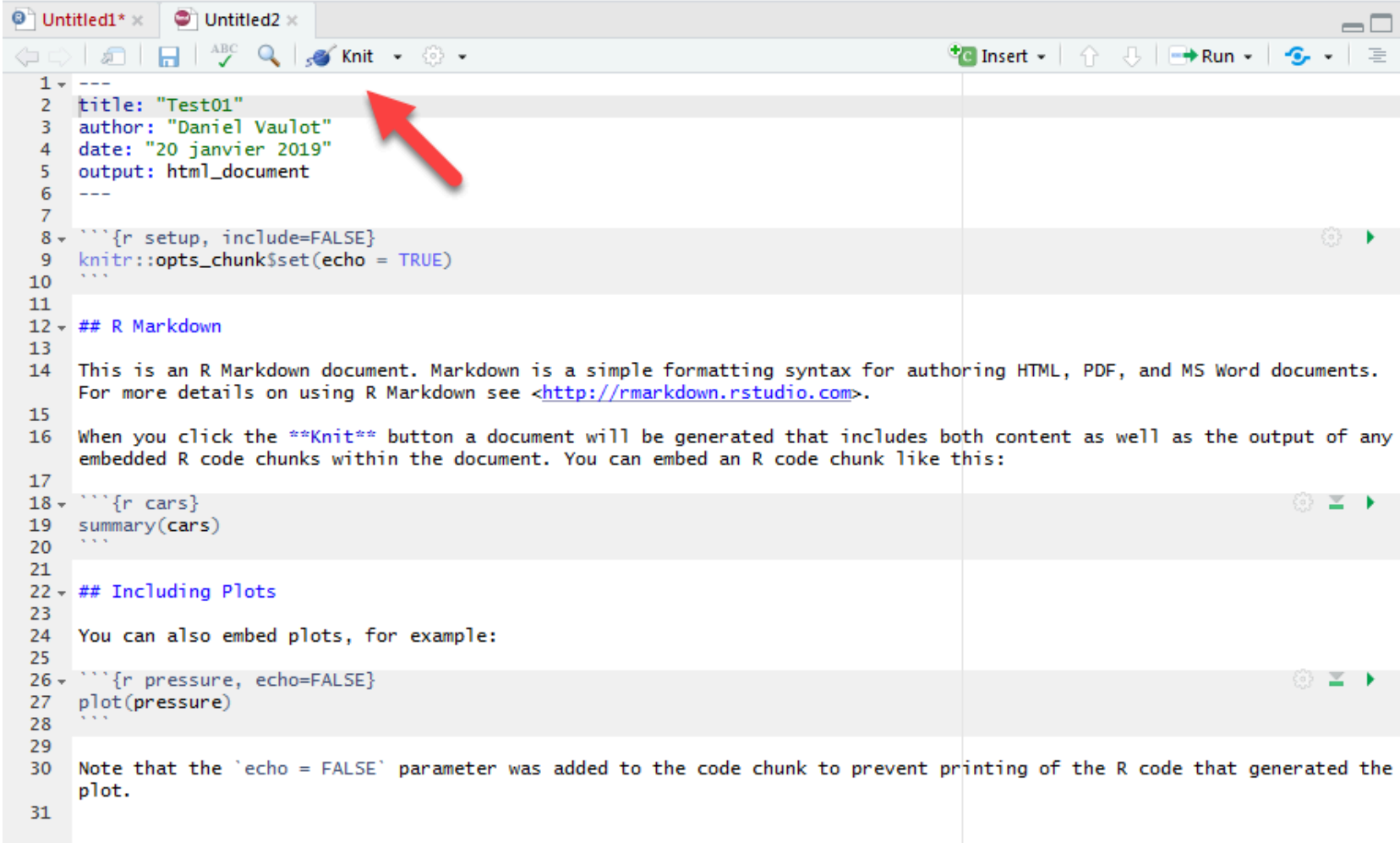
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Workspace loaded from ~/.RData]

> |
```

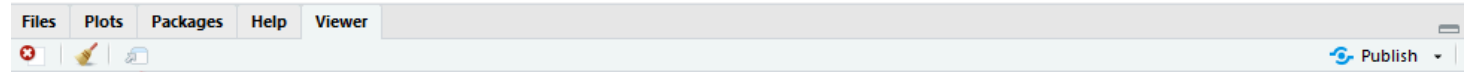
Knit to HTML

- Save to “xxx.Rmd”



```
1 ---
2 title: "Test01"
3 author: "Daniel Vaulot"
4 date: "20 janvier 2019"
5 output: html_document
6 ---
7
8 ```{r setup, include=FALSE}
9 knitr::opts_chunk$set(echo = TRUE)
10 ```
11
12 ## R Markdown
13
14 This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents.
15 For more details on using R Markdown see <http://rmarkdown.rstudio.com>.
16
17 When you click the Knit button a document will be generated that includes both content as well as the output of any
18 embedded R code chunks within the document. You can embed an R code chunk like this:
19
20 ```{r cars}
21 summary(cars)
22 ```
23
24 ## Including Plots
25
26 You can also embed plots, for example:
27
28 ```{r pressure, echo=FALSE}
29 plot(pressure)
30 ```
31
32 Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the
33 plot.
```

Knit to HTML



The screenshot shows the RStudio interface with the 'Knit' button (represented by a document icon) highlighted by a red arrow. The main content area displays the following text:

Test01

Daniel Vaulet
20 janvier 2019

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

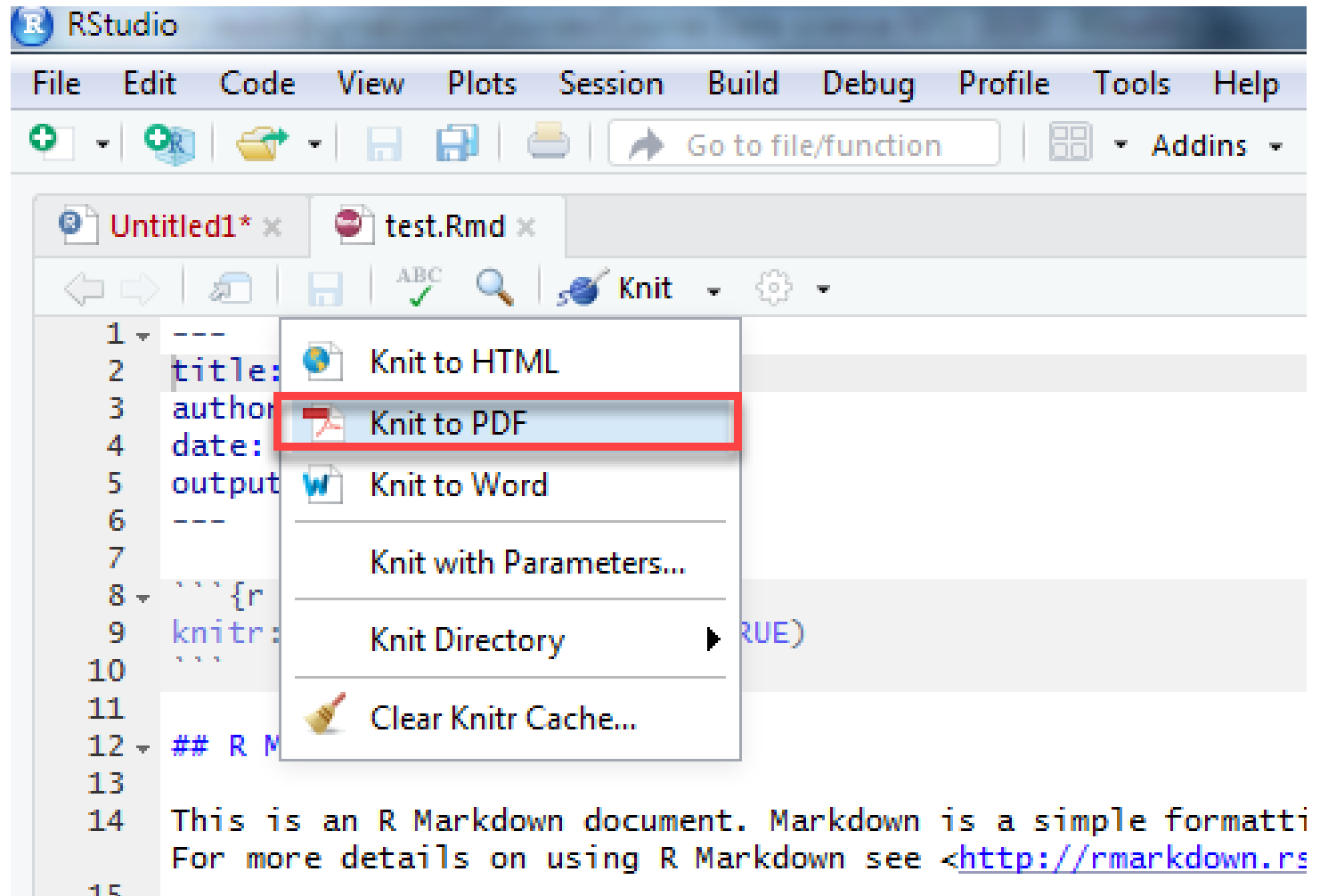
```
summary(cars)
```

##	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

Including Plots

You can also embed plots, for example:

Knit of pdf



Test01

Daniel Vaultot

20 janvier 2019

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
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
```

Including Plots

You can also embed plots, for example:

Markdown syntax

Quick guide

Structure

Headings

```
# Heading - level 1  
  
## Heading - level 2
```

Heading - level 1

Heading - level 2

Paragraphs

```
Paragraphs are separated  
by a blank line.  
  
Two spaces at the end of a line  
produces a line break.
```

Paragraphs are separated by a blank line.

Two spaces at the end of a line
produces a line break.

Formatting

Characters

```
_italic_, *italic*, **bold**, `monospace`.
```

italic, *italic*, **bold**, monospace.

- Do not mix straight and backward quotes

Formatting

Bullet lists

Bullet list:

```
* apples  
* oranges  
* pears  
  * passe crassane (4 spaces to indent)
```

- apples
- oranges
- pears
 - passe crassane

Formatting

Numbered lists

```
Numbered list:
```

```
1. wash  
1. rinse  
1. repeat
```

1. wash
2. rinse
3. repeat

Formatting

Hyperlinks

```
[Text of the link] (URL of the link)
```



```
# Example
```

```
[Markdown syntax] (https://www.markdownguide.org/basic-syntax/)
```

[Markdown syntax](#)

Formatting

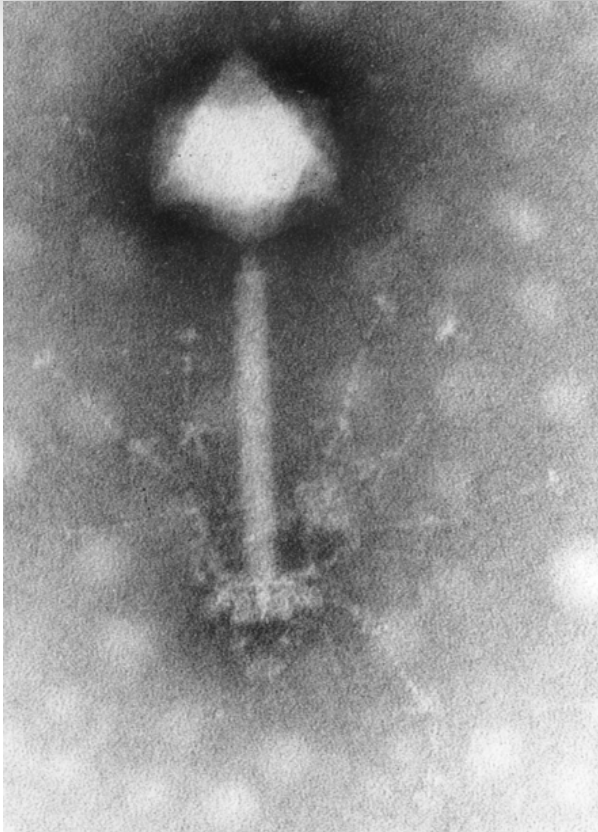
Images

```
![Image name](URL of the link - can also be a local file on your computer)
```



```
# Example
```

```
![ ](../../../../../Images/R/Synechococcus_phage.png)
```



Formatting

Tables

```
ID | First | Last
--|_--|_--
1 | Joe | Biden
```

ID	First	Last
1	Joe	Biden

Alignement

```
| Default | Left | Right | Center | |
|_-----|_:_-----|_-----|_:_-----|_:_-----|
|_ 12     | 12   | 12    | 12     |
|_ 123    | 123  | 123   | 123    |
|_ 1      | 1    | 1     | 1      |
```

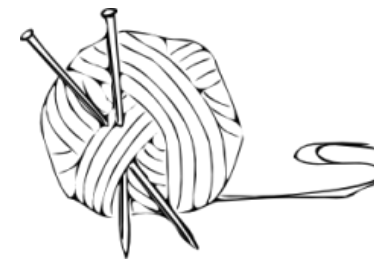
Default	Left	Right	Center
12	12	12	12
123	123	123	123
1	1	1	1

Rmarkdown

Rmarkdown conversion process



- [knitr](#) : R library
- [Pandoc](#) : command line tool
 - Converts from **md** to **pdf**, **html**, **docx**...



```
# HTML
> pandoc test1.md -f markdown -t html -s -o

# pdf
> pandoc test1.md -s -o test1.pdf
```

Pandoc a universal document converter

Donate FlatTr

- About
- Installing
- Getting started
- Demos ▾
- Documentation ▾
- Help
- Extras
- Releases

About pandoc

If you need to convert files from one markup format into another, pandoc is your swiss-army knife. Pandoc can convert documents in (several dialects of) [Markdown](#), [reStructuredText](#), [textile](#), [HTML](#), [DocBook](#), [LaTeX](#), [MediaWiki markup](#), [TWiki markup](#), [TikiWiki markup](#), [Creole 1.0](#), [Vimwiki markup](#), [roff man](#), [OPML](#), [Emacs Org-Mode](#), [Emacs Muse](#), [txt2tags](#), [Microsoft Word docx](#), [LibreOffice ODT](#), [EPUB](#), or [Haddock markup](#) to

HTML formats

XHTML, HTML5, and HTML slide shows using [Slidy](#), [reveal.js](#), [Slideous](#), [S5](#), or [DZSlides](#)

Word processor formats

[Microsoft Word docx](#), [OpenOffice/LibreOffice ODT](#), [OpenDocument XML](#), [Microsoft PowerPoint](#).

Ebooks

[EPUB version 2 or 3](#), [FictionBook2](#)

Documentation formats

[DocBook version 4 or 5](#), [TEI Simple](#), [GNU TexInfo](#), [roff man](#), [roff ms](#), [Haddock markup](#)

Archival formats

[JATS](#)

Page layout formats

[InDesign ICML](#)

Outline formats

[OPML](#)

TeX formats

[LaTeX](#), [ConTeXt](#), [LaTeX Beamer slides](#)

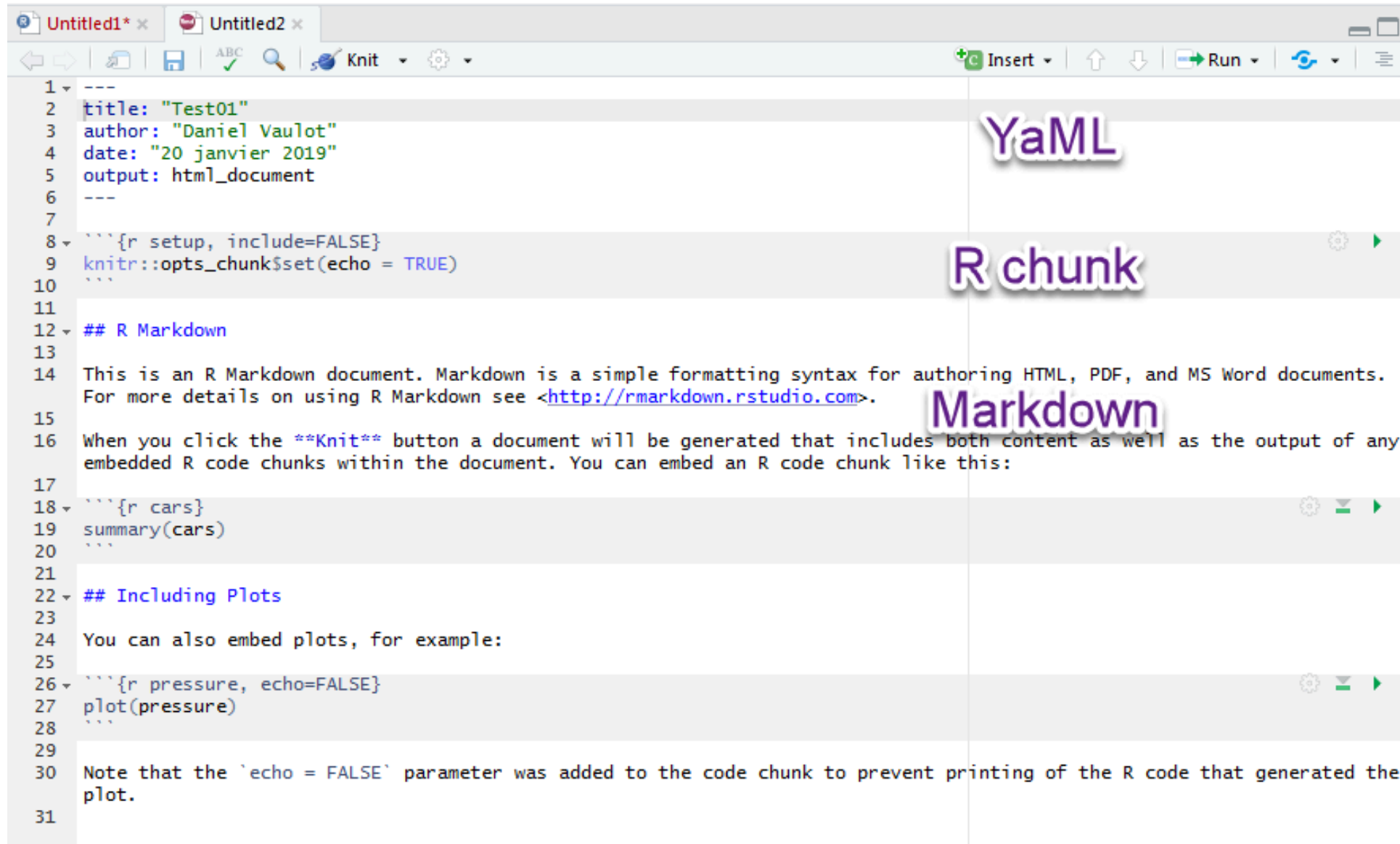
PDF

via [pdflatex](#), [xelatex](#), [lualatex](#), [pdfroff](#), [wkhtml2pdf](#), [prince](#), or [weasyprint](#).

Lightweight markup formats

[Markdown](#) (including [CommonMark](#) and [GitHub-flavored Markdown](#)), [reStructuredText](#), [AsciiDoc](#), [Emacs Org-Mode](#), [Emacs Muse](#), [Textile](#), [txt2tags](#), [MediaWiki markup](#), [DocuWiki markup](#), [TikiWiki](#)

The Rmarkdown file structure



The image shows a screenshot of the RStudio editor interface with an R Markdown file open. The file content is as follows:

```
1 ---
2 title: "Test01"
3 author: "Daniel Vaulot"
4 date: "20 janvier 2019"
5 output: html_document
6 ---
7
8 ```{r setup, include=FALSE}
9 knitr::opts_chunk$set(echo = TRUE)
10 ```
11
12 ## R Markdown
13
14 This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents.
15 For more details on using R Markdown see <http://rmarkdown.rstudio.com>.
16
17 When you click the Knit button a document will be generated that includes both content as well as the output of any
18 embedded R code chunks within the document. You can embed an R code chunk like this:
19
20 ```{r cars}
21 summary(cars)
22 ```
23
24 ## Including Plots
25
26 You can also embed plots, for example:
27
28 ```{r pressure, echo=FALSE}
29 plot(pressure)
30 ```
31
32 Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the
33 plot.
```

Annotations in the image:

- YaML**: A purple, rounded text label pointing to the YAML front-matter (lines 2-5).
- R chunk**: A purple, rounded text label pointing to the first R code chunk (lines 8-10).
- Markdown**: A purple, rounded text label pointing to the main text of the document (lines 14-16).

Knit process

```
Console | R Markdown x
C:/Users/vaulot/Desktop/test.Rmd

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: cars
|.....| 71%
ordinary text without R code

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

|.....| 100%
ordinary text without R code

"C:/Program Files/RStudio/bin/pandoc/pandoc" +RTS -K512m -RTS test.utf8.md --to latex --from markdown+autolink_bare_uris+asci
i_identifiers+tex_math_single_backslash --output test.tex --template "C:\PROGRA~1\R\R-35~1.2\library\RMARKD~1\rmd\latex\DEFAU
L~3.TEX" --highlight-style tango --pdf-engine pdflatex --variable graphics=yes --variable "geometry:margin=1in" --variable "c
ompact-title:yes"
output file: test.knit.md

Output created: test.pdf
```

Output

Test01

Daniel Vaultot

20 janvier 2019

Markdown

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the Knit button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

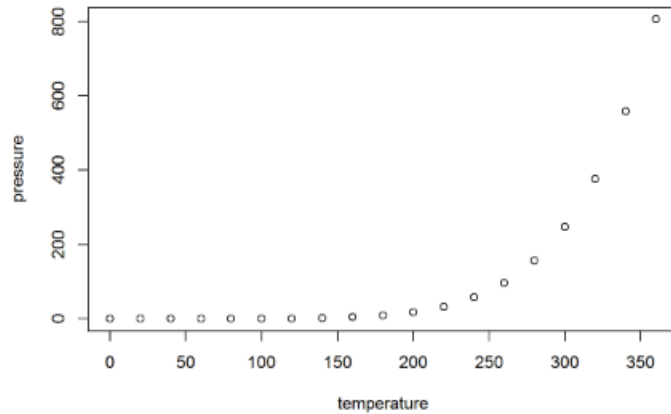
```
##      speed      dist
## Min.   : 4.0   Min.   : 2.00
## 1st Qu.:12.0   1st Qu.: 26.00
## Median :15.0   Median : 36.00
## Mean   :15.4   Mean    : 42.98
## 3rd Qu.:19.0   3rd Qu.: 56.00
## Max.   :25.0   Max.   :120.00
```

R code

R output

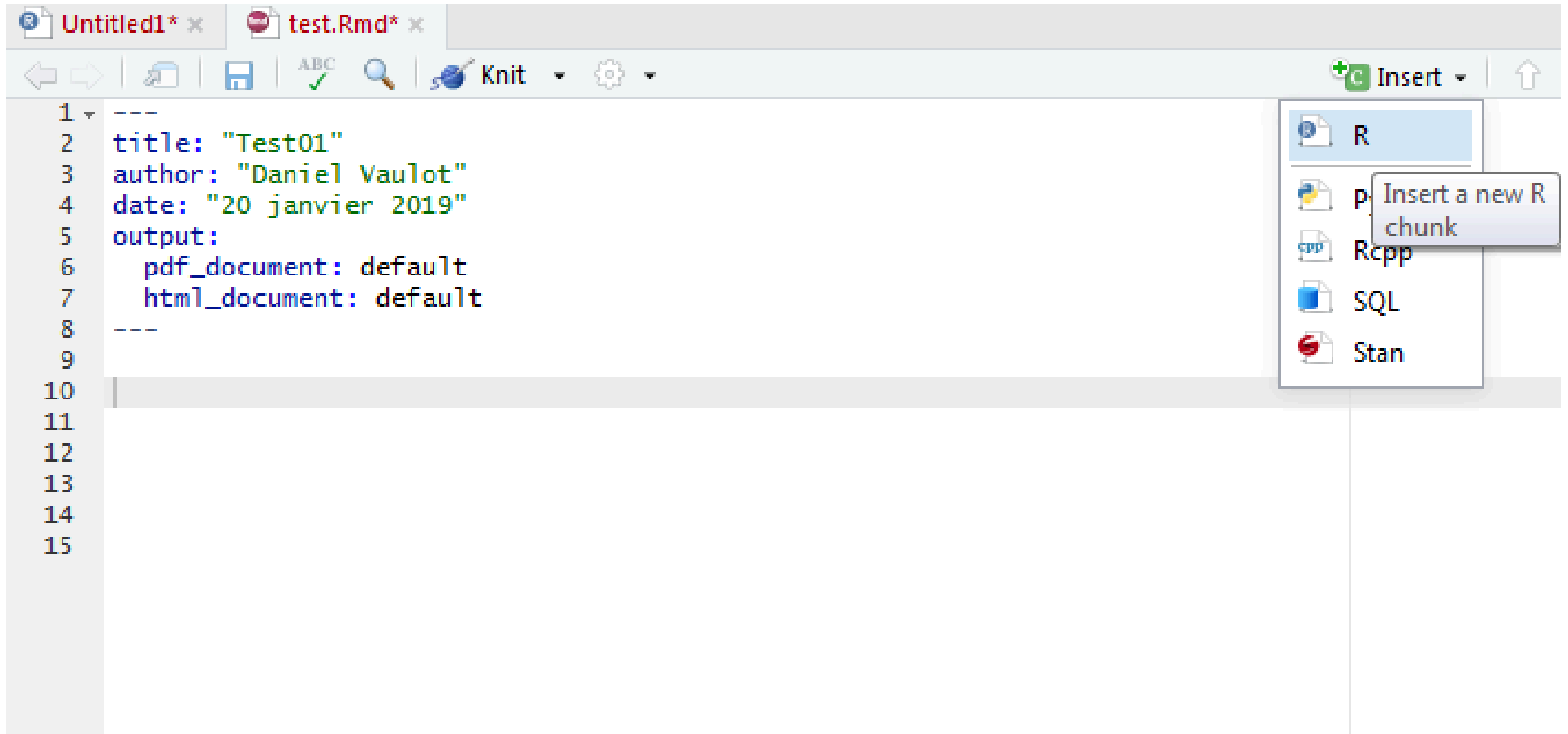
Including Plots

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.

Insert R chunk



The screenshot shows the Quarto editor interface. The top toolbar includes navigation arrows, a save icon, a search icon, and a 'Knit' button. The main editor area displays a code chunk with the following content:

```
1 ---  
2 title: "Test01"  
3 author: "Daniel Vaulet"  
4 date: "20 janvier 2019"  
5 output:  
6   pdf_document: default  
7   html_document: default  
8 ---  
9  
10  
11  
12  
13  
14  
15
```

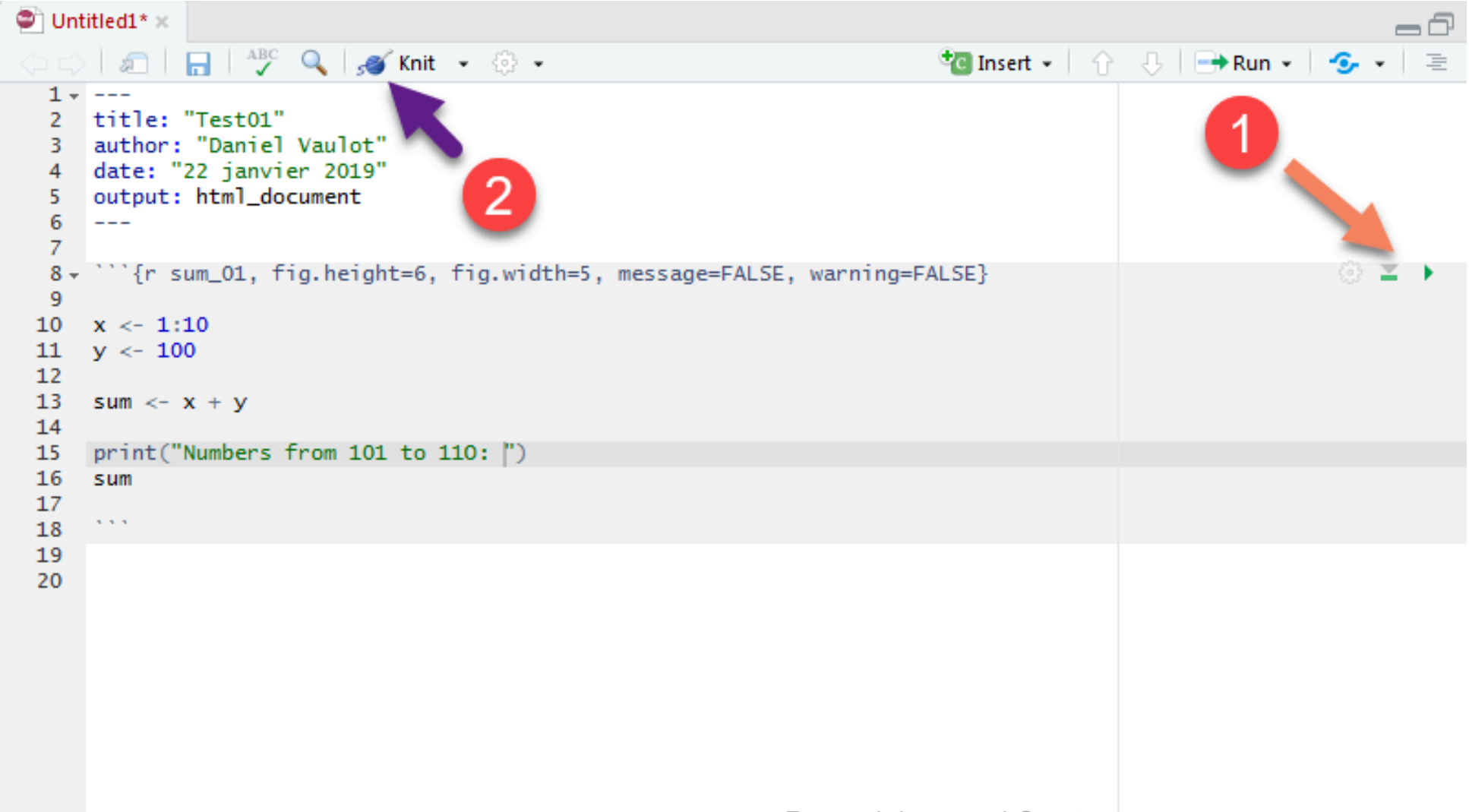
The 'Insert' menu is open, showing options for inserting a new chunk of code:

- R
- p (Python)
- Rcpp
- SQL
- Stan

A tooltip is visible over the 'R' option, stating: "Insert a new R chunk".

Run R chunk

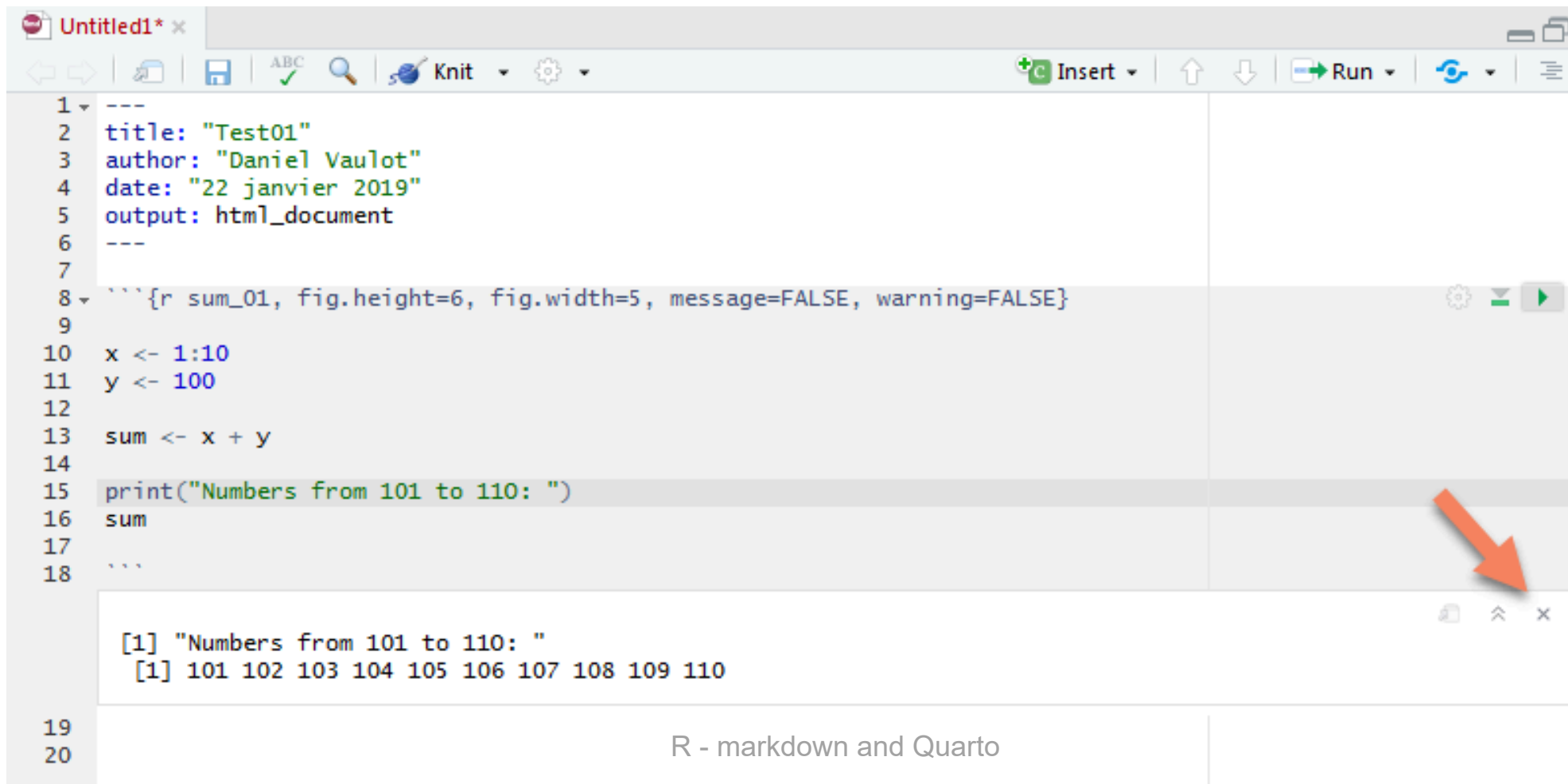
Two options



Run R chunk

Option 1: Run R chunk inside Rmd file

- Use when building and debugging an Rmd file



The screenshot shows the RStudio interface with an R chunk being executed. The code in the chunk is as follows:

```
1 ---  
2 title: "Test01"  
3 author: "Daniel Vaillot"  
4 date: "22 janvier 2019"  
5 output: html_document  
6 ---  
7  
8 ```{r sum_01, fig.height=6, fig.width=5, message=FALSE, warning=FALSE}  
9  
10 x <- 1:10  
11 y <- 100  
12  
13 sum <- x + y  
14  
15 print("Numbers from 101 to 110: ")  
16 sum  
17  
18 ```
```

The output of the chunk is displayed in the console window below the code:

```
[1] "Numbers from 101 to 110: "  
[1] 101 102 103 104 105 106 107 108 109 110
```

An orange arrow points to the Run button (a green play icon) in the top right corner of the chunk editor.

Run R chunk

Option 2: Knit R chunk to HTML

- Use for final production

Test01

Daniel Vaultot

22 janvier 2019

```
x <- 1:10
y <- 100

sum <- x + y

print("Numbers from 101 to 110: ")
```

```
## [1] "Numbers from 101 to 110: "
```

```
sum
```

```
## [1] 101 102 103 104 105 106 107 108 109 110
```

Options for R chunks

The screenshot shows the RStudio interface with a code editor on the left and a chunk options dialog on the right. The code editor contains the following R Markdown content:

```
1 ---  
2 title: "Test01"  
3 author: "Daniel Vaulot"  
4 date: "20 janvier 2019"  
5 output:  
6   pdf_document: default  
7   html_document: default  
8 ---  
9  
10 {r sum_01, echo=TRUE, fig.height=6, fig.width=5, message=FALSE, warning=FALSE}  
11  
12  
13  
14  
15  
16  
17  
18
```

The chunk options dialog is open for the chunk named "sum_01". It shows the following settings:

- Name:
- Output:
- Show warnings
- Show messages
- Use paged tables
- Use custom figure size
- Width (inches):
- Height (inches):
- [? Chunk options](#)
-

A red arrow points to the gear icon in the top right corner of the code editor, which is used to open the chunk options dialog.

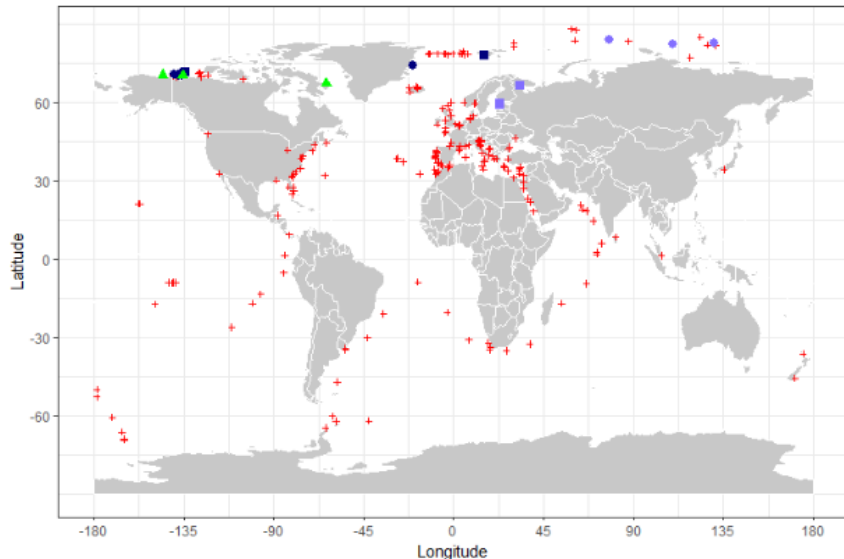
Useful options

Options	Default value	Aim
echo	TRUE	Print code (= FALSE in a report for example)
eval	TRUE	Evaluate code (= FALSE if want to show code only)
warning	TRUE	Warning message (= FALSE to remove long warnings)
message	TRUE	Messages (= FALSE to remove long messages)
cache	FALSE	If TRUE only modified chunks will be evaluated very useful for computing heavy codes
fig.height		inches
fig.width		inches

What can you do with Rmarkdown ?

Document your data analyses

- If the data changes, you can re-run analysis in a matter of minutes
- More and more journal request analyses scripts



Draw maps

Mercator projection

```
species_map <- ggplot() + geom_polygon(data = world.df, aes(x = long, y = lat,
  group = group, fill = ""), colour = "white", size = 0.1) + scale_fill_manual(values = color_cont
  guide = FALSE) + scale_x_continuous(breaks = (-4:4) * 45) + scale_y_continuous(breaks = (-2:2) *
  30) + xlab("Longitude") + ylab("Latitude") + coord_fixed(1.3) + theme_bw()
# species_map <- species_map + coord_map() # Mercator projection
# species_map <- species_map + coord_map('gilbert') # Nice for the poles

# Add points where not detected
df_map <- dplyr::filter(metabarcodes_species, is.na(reads_total_species))
species_map <- species_map + geom_point(data = df_map, aes(x = longitude, y = latitude),
  color = color_not_present, size = size_cross, shape = 3)

# Add the ice metabarcodes
df_map <- dplyr::filter(metabarcodes_species, !is.na(reads_total_species) &
  substrate == "ice")
species_map <- species_map + geom_point(data = df_map, aes(x = longitude, y = latitude),
  color = color_ice, size = size_points)

# Add the water metabarcodes
df_map <- dplyr::filter(metabarcodes_species, !is.na(reads_total_species) &
  substrate == "water")
species_map <- species_map + geom_point(data = df_map, aes(x = longitude, y = latitude),
  color = color_water, size = size_points)

# Add the water sequence
df_map <- dplyr::filter(genbank_species, substrate == "water")
species_map <- species_map + geom_point(data = df_map, aes(x = longitude, y = latitude),
  color = color_water, size = size_points, shape = 15)

# Add the ice sequence
df_map <- dplyr::filter(genbank_species, substrate == "ice")
species_map <- species_map + geom_point(data = df_map, aes(x = longitude, y = latitude),
  color = color_ice, size = size_points, shape = 15)

# Add the culture
df_map <- dplyr::filter(cultures)
species_map <- species_map + geom_point(data = df_map, aes(x = longitude, y = latitude),
  color = color_cultures, size = size_points, shape = 17)

# Display and save
species_map
```

Presentation

<https://github.com/yihui/xaringan>



Presentation Ninja

✕

with xaringan

Yihui Xie

2016/12/12

Hello World

Install the **xaringan** package from **GitHub**:

```
devtools::install_github("yihui/xaringan")
```

You are recommended to use the **RStudio IDE**, but you do not have to.

- Create a new R Markdown document from the menu **File -> New File -> R Markdown -> From Template -> Ninja Presentation**^[1]
- Click the **Knit** button to compile it;
- or use the **RStudio Addin**^[2] "Infinite Moon Reader" to live preview the slides (every time you update and save the Rmd document, the slides will be automatically reloaded in RStudio Viewer).

[1] 中文用户请看这份教程

[2] See #2 if you do not see the template or addin in RStudio.

3 /

Presentation Ninja

✕

with xaringan

Yihui Xie

2016/12/12

Hello World

Install the **xaringan** package from **GitHub**:

```
devtools::install_github("yihui/xaringan")
```

You are recommended to use the **RStudio IDE**, but you do not have to.

- Create a new R Markdown document from the menu **File -> New File -> R Markdown -> From Template -> Ninja Presentation**^[1]
- Click the **Knit** button to compile it;
- or use the **RStudio Addin**^[2] "Infinite Moon Reader" to live preview the slides (every time you update and save the Rmd document, the slides will be automatically reloaded in RStudio Viewer).

[1] 中文用户请看这份教程

[2] See #2 if you do not see the template or addin in RStudio.

R - markdown and Quarto

Posters

<https://github.com/brentthorne/posterdown>



Using posterdown to generate reproducible conference posters via RMarkdown > Knitr > Markdown > Pandoc > Latex > PDF workflow

Author One¹ Author Two²

¹Department of Poster Layouts, University of Markdown, ²Department of Another Institution, Institution University

Introduction

Welcome to posterdown! This is my attempt to provide a semi-smooth workflow for those who wish to take their RMarkdown skills to the conference world. Many creature comforts from RMarkdown are available in this package such as Markdown section notation, figure captioning, and even citations like this one [1]. The rest of this example poster will show how you can insert typical conference poster features into your own document.

Study Site

Here is a map made to show the study site using ggplot2, ggspatial, and sf. Lorem ipsum dolor sit amet, [1] consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Phasellus vestibulum lorem sed risus ultricies tristique nulla. Mauris vitae ultricies leo integer malesuada nunc vel risus commodo. Suspendisse potenti nullam ac tortor vitae. Enim nunc faucibus a pellentesque sit amet porttitor eget.

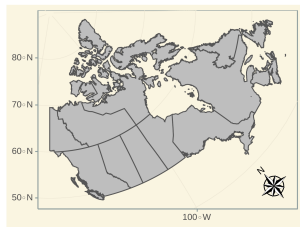


Figure 1: This is a map of Canada, the ggspatial package is great for GIS folks in R!

Objectives

1. Easy to use reproducible poster design.
2. Integration with RMarkdown.
3. Easy transition from posterdown to thesistdown or rtitles

Methods

This package uses the same workflow approach as the RMarkdown you know and love. Basically it goes from RMarkdown > Knitr > Markdown > Pandoc > Latex > PDF

Results

Usually you want to have a nice table displaying some important results that you have calculated. In posterdown this is as easy as using the kable table formatting you are probably use to as per typ-

cal RMarkdown formatting. I suggest checking out the kableExtra package and its in depth documentation on customizing these tables found [here](#).

Table 1: Tables are a breeze with Kable and Kable extra package!

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa

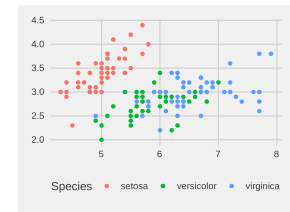


Figure 2: A typical plot using ggplot using the classic iris dataset.

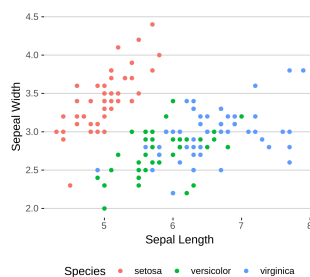


Figure 3: Another typical plot using ggplot, this time with a different theme and r code chunk options for fig.width and fig.height.

```
# Here is some code for people  
# so look at and be in awe of!!!!  
library(ggplot2)  
library(ggthemes)  
  
ggplot(data=iris,  
       aes(x = Sepal.Width,  
           y = Sepal.Length,  
           colour = Species)) +  
  geom_point() +  
  theme_stata() +  
  NULL
```

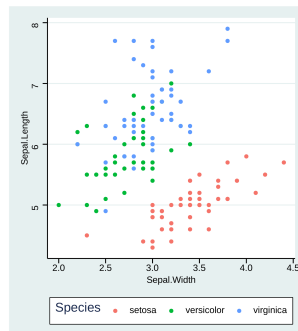


Figure 4: Another figure showing how base R plots might look on this poster!

Next Steps

There is still A LOT of work to do on this package which include (but are note limited to):

- Better softcoding for front end user options in YAML
- Images in the title section for logo placement which is a common attribut to posters as far as I have come to know
- Figure out compatibility with natbib which wasn't working during the initial set up
- MUCH BETTER PACKAGE DOCUMENTATION. For example, there is nothing in the README.
- Include References section only if initiated by the user like in RMarkdown.

References

[1] Erin-Lou Hilder et al. "Identifying structural complexity in environmental data: An image analysis approach to generalised gull reproduction". In: *On Ecology Review* 46 (Aug. 2020), pp. 45-56. doi:10.1080/00141801.2020.1812302. <https://doi.org/10.1080/00141801.2020.1812302>

[2] Isaac Robinson, Yu Shih, and Peter Szepesvári. "OPTICAL CHARACTER RECOGNITION USING THE OBJECT-BASED IMAGE ANALYSIS APPROACH: RESULTS OF SEMI-AUTOMATED ANNOTATING VISUAL INTERPRETATION". In: *ICDAR*

Curriculum vitae

<https://cloud.r-project.org/web/packages/vitae/index.html>

Eric Scott

PhD Candidate

Ecometabolomics, multivariate statistics, R



Education

2014–2019 **PhD**, *Tufts University*, Medford.

2007–2010 **MS**, *University of Illinois at Urbana-Champaign*, Urbana.

2002–2006 **B.A.**, *Whitman College*, Walla Walla.

Research Experience

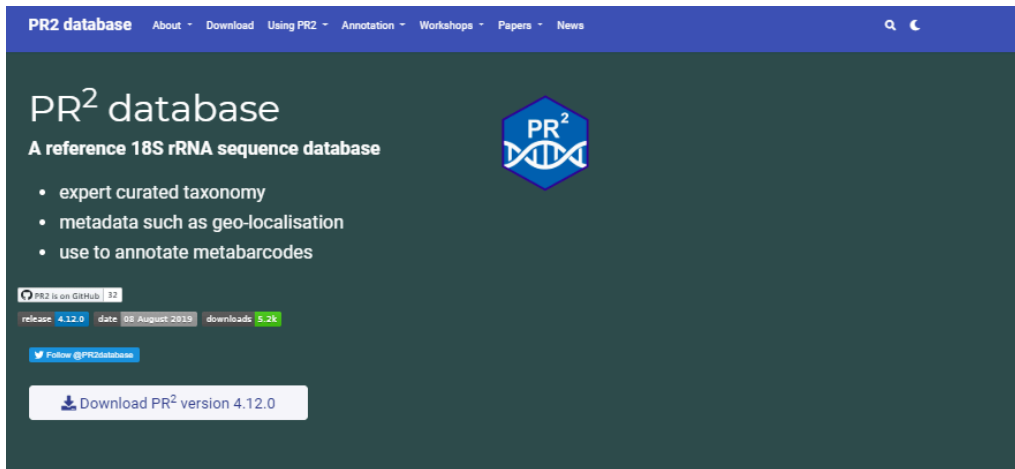
2017–2018 **NSF grant coordinator**, *Tufts University*, Medford, MA.

- Schedule and implement conference calls
- Coordinate in-person meetings

R - markdown and Quarto

Website

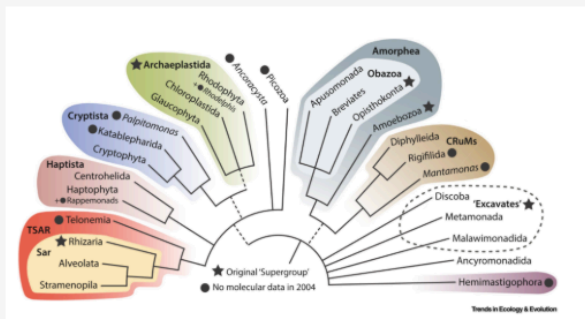
<https://bookdown.org/yihui/blogdown/>



The screenshot shows the PR2 database website. The header includes navigation links: About, Download, Using PR2, Annotation, Workshops, Papers, News, and a search icon. The main content area features the title "PR² database" and the subtitle "A reference 18S rRNA sequence database". A blue hexagonal logo with "PR²" and a DNA double helix is positioned to the right. Below the title, there is a list of features: expert curated taxonomy, metadata such as geo-localisation, and use to annotate metabarcodes. A GitHub repository link is shown with 32 stars. Release information indicates version 4.12.0, dated 08 August 2019, with 5.2k downloads. A "Follow @PR2database" button and a "Download PR² version 4.12.0" button are also present.



About PR2



The eukaryotic tree. From Burki et al. 2019. The New Tree of Eukaryotes. Trends in Ecology & Evolution. DOI: 10.1016/j.tree.2019.08.008.

Current version : 4.12.0
Last update : 8 August 2019
DOI : 10.6084/m9.figshare.5913181

R - markdown and Quarto

Quarto

Quarto

- New flavor of markdown
- Independent of R
- Can include also Python, Julia chunks
- Will evolve while R markdown will not be updated
- Many new powerful features

Welcome to Quarto

An open-source scientific and technical publishing system

- Author using [Jupyter](#) notebooks or with plain text markdown in your favorite editor.
- Create dynamic content with [Python](#), [R](#), [Julia](#), and [Observable](#).
- Publish reproducible, production quality articles, presentations, dashboards, websites, blogs, and books in HTML, PDF, MS Word, ePub, and more.
- Share knowledge and insights organization-wide by publishing to [Posit Connect](#), [Confluence](#), or other publishing systems.
- Write using [Pandoc](#) markdown, including equations, citations, crossrefs, figure panels, callouts, advanced layout, and more.

Quarto vs Rmarkdown

1. Different formatting of yaml options
2. Different formatting of chunk options

Hello, Quarto

Python R Julia Observable

Quarto is a multi-language, next generation version of R Markdown from RStudio, with many new new features and capabilities. Like R Markdown, Quarto uses [Knitr](#) to execute R code, and is therefore able to render most existing Rmd files without modification.

```
---
title: "ggplot2 demo"
author: "Norah Jones"
date: "5/22/2021"
format:
  html:
    fig-width: 8
    fig-height: 4
    code-fold: true
---

## Air Quality

@fig-airquality further explores the impact of temperature
on ozone level.

```{r}
#| Label: fig-airquality
#| fig-cap: Temperature and ozone level.
#| warning: false

library(ggplot2)

ggplot(airquality, aes(Temp, Ozone)) +
 geom_point() +
 geom_smooth(method = "loess"
)
```
```

ggplot2 demo
Norah Jones
May 22nd, 2021

Air Quality

[Figure 1](#) further explores the impact of temperature on ozone level.

► Code

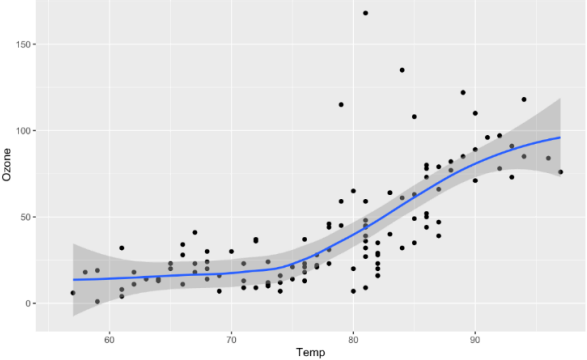


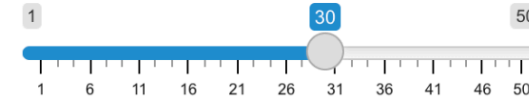
Figure 1: Temperature and ozone level.

Interactive documents (Shiny)

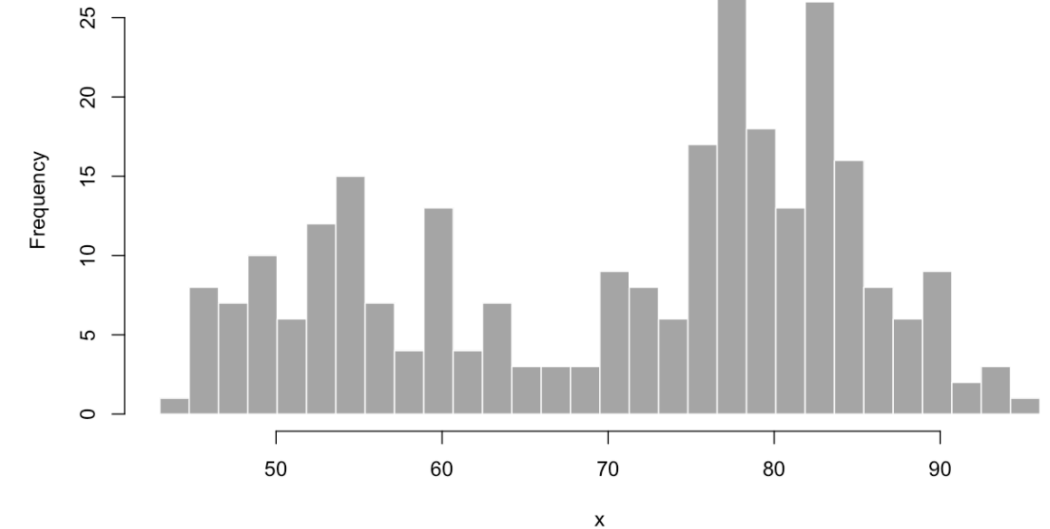
```
---  
title: "Old Faithful"  
format: html  
server: shiny  
---  
  
``{r}  
sliderInput("bins", "Number of bins:",  
           min = 1, max = 50, value = 30)  
plotOutput("distPlot")  
---  
  
``{r}  
#| context: server  
output$distPlot <- renderPlot({  
  x <- faithful[, 2] # Old Faithful Geyser data  
  bins <- seq(min(x), max(x), length.out = input$bins + 1)  
  hist(x, breaks = bins, col = 'darkgray', border = 'white')  
})  
---
```

Old Faithful

Number of bins:



Histogram of x



Writing a report with quarto

- report-quarto folder in data.zip file
- Files
 - img
 - Tara_Ocean.png
 - bibliography.bib
 - report.pdf
 - report.qmd

Report.qmd

Yaml header

```
title: "Report template"
format: pdf
author:
  - name: Name1 Surname
  - name: Name2 Surname
  - name: Name3 Surname
  - name: Name4 Surname
  - name: Name5 Surname
abstract: |
  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur eget porta erat. Morbi consectetur
  est vel gravida pretium. Suspendisse ut dui eu ante cursus gravida non sed sem. Nullam sapien tellus
  commodo id velit id, eleifend volutpat quam. Phasellus mauris velit, dapibus finibus elementum ve
bibliography: bibliography.bib
```

Report.qmd

Text



```
# Introduction
```

```
Lorem ipsum dolor sit [bib1] amet, consectetur adipiscing elit.  
Curabitur eget porta erat. Morbi consectetur est vel gravida pretium.
```

```
# Materials and methods
```

```
## Etiam eget sapien nibh
```

```
Nulla mi mi, venenatis sed ipsum varius, volutpat euismod diam.
```

```
# Results
```

```
Nulla mi mi, venenatis sed ipsum varius, volutpat euismod diam.
```

```
## 2 Level
```

```
### 3rd level heading
```

```
Maecenas convallis mauris sit amet sem ultrices gravida. Etiam eget
```

Report.qmd

References

Call a reference

```
Lorem ipsum dolor sit [@bib1] amet, consectetur adipiscing elit.  
Curabitur eget porta erat. Morbi consectetur est vel gravida pretium.
```

Create list of references

- Use file bibliography.bib defined in yaml header

```
\newpage  
  
# References  
  
::: {#refs}  
:::
```


bibliography.bib

- bibtex format for references
- Created from Zotero

```
@article{bib1,  
  language = {eng},  
  number = {12},  
  pages = {938-950},  
  title = {Turning a hobby into a job: How duplicated genes find new functions},  
  volume = {9},  
  year = {2008},  
  author = {Wolfe, Kenneth H and Conant, Gavin C},  
  address = {England},  
  copyright = {COPYRIGHT 2008 Nature Publishing Group},  
  issn = {1471-0056},  
  journal = {Nature reviews. Genetics},  
}
```

Report.qmd

Tables

Call a table

```
# Results
```

```
Aliquam in enim semper, aliquam massa id, cursus neque. Praesent  
faucibus semper libero (@tbl-summary).
```

Display a table

```
# Tables
```

```
Col1	Col2	Col3
A	B	C
E	F	G
A	G	G
```

```
: My Caption {#tbl-summary}
```

Report.qmd

Figures

Call a figure

```
# Results  
  
massa. In vitae diam ac augue semper tincidunt eu ut eros (@fig-tara). Fusce
```



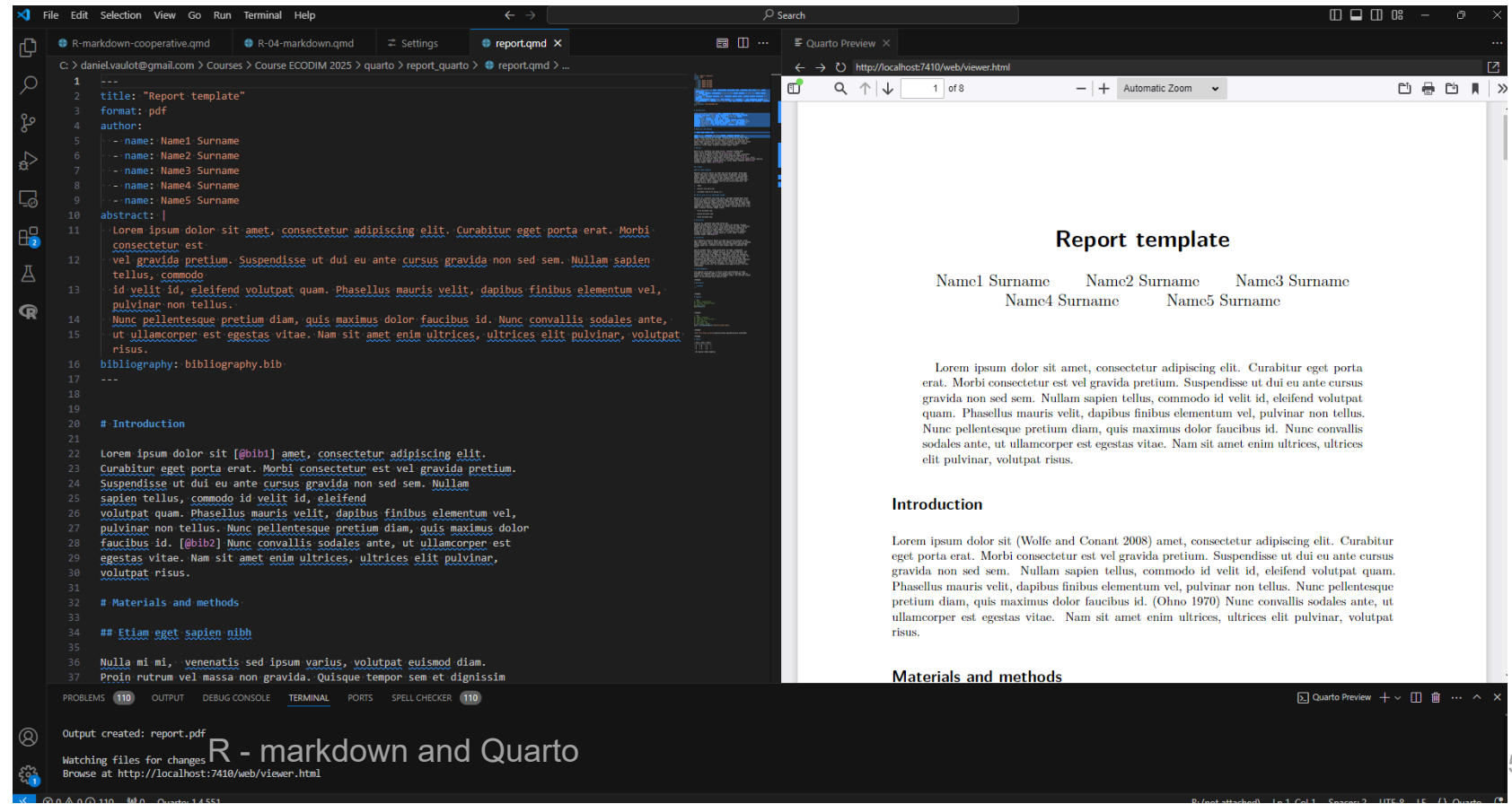
Display a figure

```
![The Tara Ocean project] (img/Tara_Ocean.png) {#fig-tara}
```



Compile to pdf or html

- R studio
- Visual Studio Code
 - Need to install [Quarto extension](#)



Cooperative writing

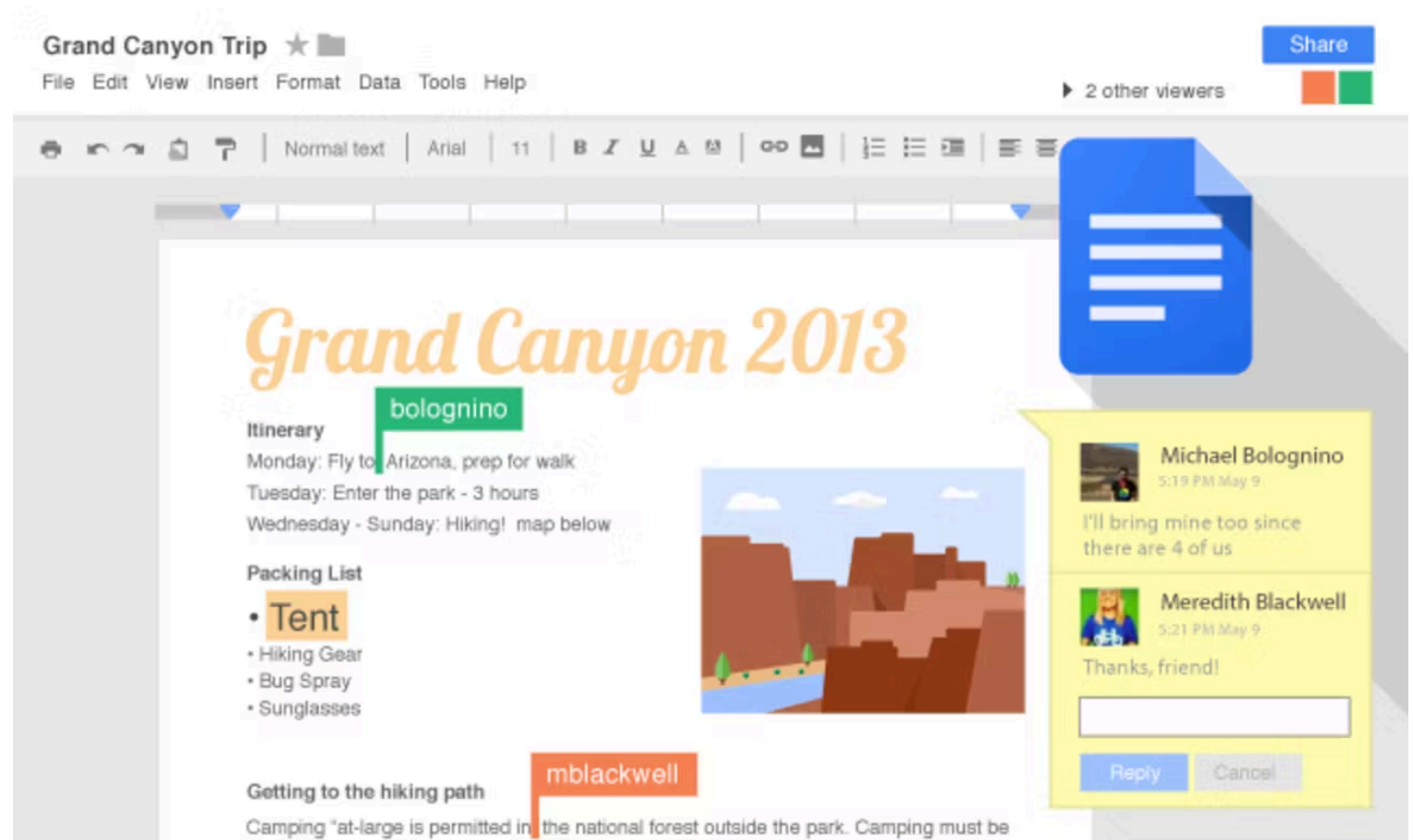
Word / Google doc

Advantages

- Easy to use
- “Free”
- Reference management easy

Disadvantages

- Can become very messy
- Output is often ugly
- Figure management awful
- Hard to reformat



Visual Studio

Live Share extension

The image shows the Visual Studio Live Share extension landing page and a code editor interface. The landing page has a dark background with the title "Visual Studio Live Share" in white. Below the title is the subtitle "Développement collaboratif en temps réel". There are three main buttons: "Télécharger Visual Studio" (with a dropdown arrow), "Visual Studio Code" (in a blue button), and "Visual Studio Code pour le Web".

The code editor interface shows a sidebar on the left with "SESSION DETAILS" and a list of participants: Jon W Chu (Header.js:12), Amanda Silver (GuestbookGrid.js:13), and PJ Meyer (GuestbookGrid.js:9). The main editor area displays JavaScript code for a React component. The code includes imports for GridArrow, GridLegend, and GuestbookGridCell. The GuestbookGrid class extends Component and has a constructor and a render method. The render method uses a map function to render GuestbookGridCell components based on the state signatures. The code is as follows:

```
1 import GridArrow from "./GridArrow";
2 import GridLegend from "./GridLegend";
3 import GuestbookGridCell from "./GuestbookGridCell";
4
5 export default class GuestbookGrid extends Component {
6   constructor(props) {
7     super(props)
8     this.state = PJ Meyer
9     signatures: signatures
10  }
11 }
12
13 Amanda Silver
14 render() {
15   const cells = this.state.signatures.map((signature, index) => (
16     <GuestbookGridCell key={index} {...signature} />
17   ));
18 }
```

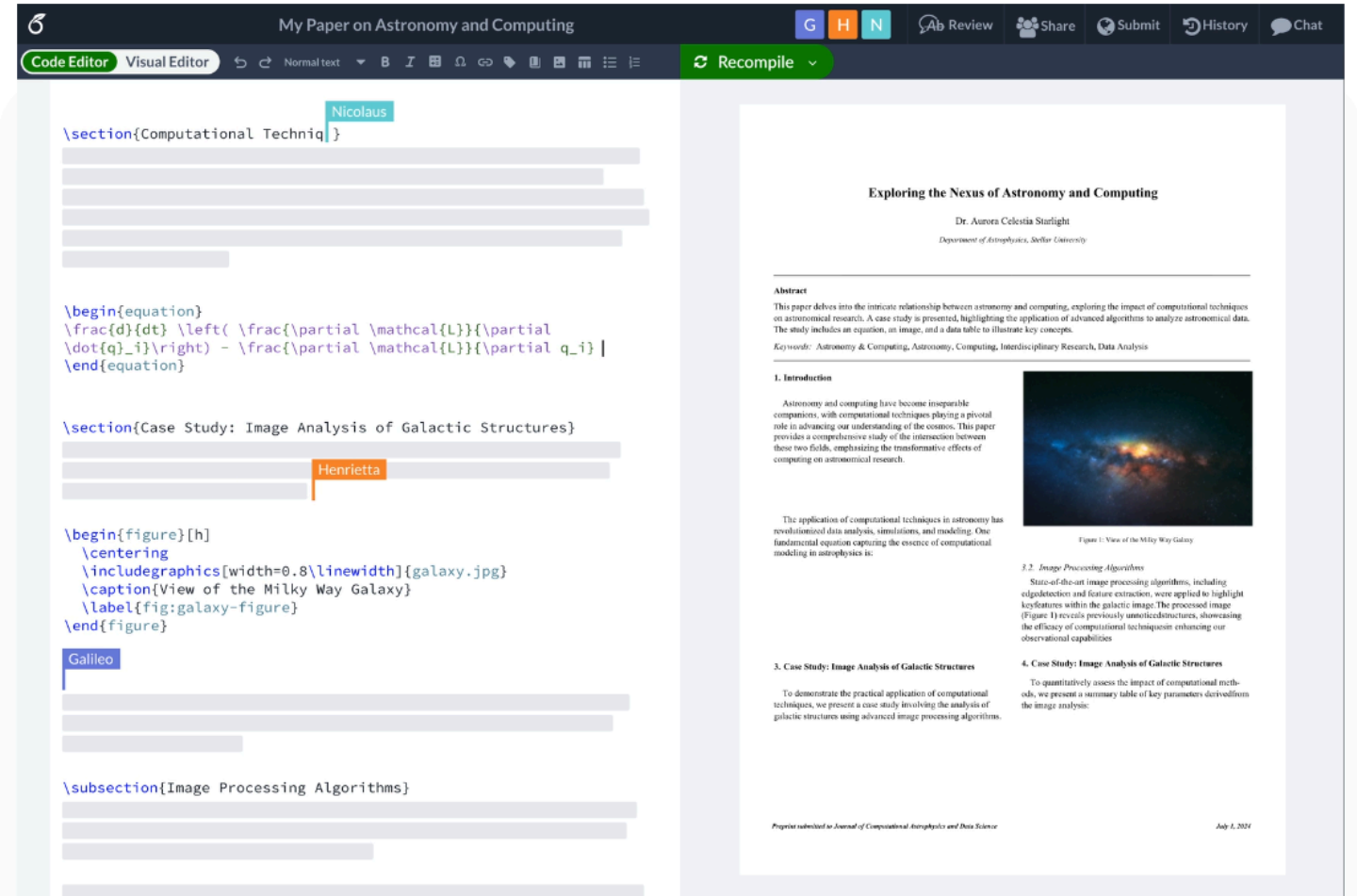
Overleaf

- <https://overleaf.com/>
- Based on Latex

Advantages

- Very powerful
- Can be used to write papers and thesis
- Very professional look (pdf)
- Reformatting quite easy
- No jumping of figures
- Sharing and comments
- Handle equations

Disadvantages



The screenshot displays the Overleaf online LaTeX editor interface. The top navigation bar includes the Overleaf logo, the document title "My Paper on Astronomy and Computing", and various utility icons for Google, Home, and Next, along with "Review", "Share", "Submit", "History", and "Chat" buttons. Below the navigation bar, there are tabs for "Code Editor" and "Visual Editor", and a "Recompile" button. The main workspace is split into two panes. The left pane shows the LaTeX source code with several sections and a mathematical equation. The right pane shows a preview of the rendered PDF document. The document title is "Exploring the Nexus of Astronomy and Computing" by Dr. Aurora Celestia Starlight, Department of Astrophysics, Stellar University. The document includes an abstract, keywords, an introduction, and two case studies. A figure of the Milky Way Galaxy is included in the second case study.

```
\section{Computational Techniq}

\begin{equation}
\frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{q}_i} \right) - \frac{\partial \mathcal{L}}{\partial q_i} |
\end{equation}

\section{Case Study: Image Analysis of Galactic Structures}

\begin{figure}[h]
\centering
\includegraphics[width=0.8\linewidth]{galaxy.jpg}
\caption{View of the Milky Way Galaxy}
\label{fig:galaxy-figure}
\end{figure}

Galileo

\subsection{Image Processing Algorithms}
```

Exploring the Nexus of Astronomy and Computing
Dr. Aurora Celestia Starlight
Department of Astrophysics, Stellar University

Abstract
This paper delves into the intricate relationship between astronomy and computing, exploring the impact of computational techniques on astronomical research. A case study is presented, highlighting the application of advanced algorithms to analyze astronomical data. The study includes an equation, an image, and a data table to illustrate key concepts.

Keywords: Astronomy & Computing, Astronomy, Computing, Interdisciplinary Research, Data Analysis

1. Introduction
Astronomy and computing have become inseparable companions, with computational techniques playing a pivotal role in advancing our understanding of the cosmos. This paper provides a comprehensive study of the intersection between these two fields, emphasizing the transformative effects of computing on astronomical research.

The application of computational techniques in astronomy has revolutionized data analysis, simulations, and modeling. One fundamental equation capturing the essence of computational modeling in astrophysics is:

2.2. Image Processing Algorithms
State-of-the-art image processing algorithms, including edgedetection and feature extraction, were applied to highlight keyfeatures within the galactic image. The processed image (Figure 1) reveals previously unnoticeablestructures, showcasing the efficacy of computational techniquesin enhancing our observational capabilities.

3. Case Study: Image Analysis of Galactic Structures
To demonstrate the practical application of computational techniques, we present a case study involving the analysis of galactic structures using advanced image processing algorithms.

4. Case Study: Image Analysis of Galactic Structures
To quantitatively assess the impact of computational methods, we present a summary table of key parameters derivedfrom the image analysis.

Progress submitted to Journal of Computational Astrophysics and Data Science July 6, 2024

Overleaf

- Really good for papers

The screenshot displays the Overleaf web editor interface. The top navigation bar includes options for Review, Share, Submit, History, Layout, and Chat. The main workspace is divided into three panels: a file explorer on the left, a code editor in the center, and a preview window on the right.

File Explorer: Shows a project structure with folders like 'fig', 'img', 'tables', and 'version 1.0', 'version 2.0'. Files include 'main_no_supplementary.tex', 'naturemag.doi.sty', and 'wscirep.cls'.

Code Editor: Displays LaTeX source code for a document class. Key sections include:

- Line 1-11: Document class and package loading (utf8, fontenc, setspace, lineno, gensymb, toctof, lscap, soul, longtable).
- Line 12-14: Hyperref setup for color links.
- Line 15: \usepackage{ur}.
- Line 18-24: \newcommand for figure labels in the supplementary.
- Line 25-29: Comments on visualizing references.
- Line 30-42: \usepackage for biber and biblatex options.
- Line 43-44: \title{Annual phytoplankton dynamics in coastal waters from Fildes Bay, Western Antarctic Peninsula.}
- Line 45-50: \author list for Nicole Trefault, Rodrigo De la Iglesia, Mario Moreno-Pino, Adriana Lopes dos Santos, and Catherine G erikas Ribeiro.
- Line 51-59: \affil list for affiliations and corresponding authors.

Preview Window: Shows the rendered document with the title "Annual phytoplankton dynamics in coastal waters from Fildes Bay, Western Antarctic Peninsula." and the authors: "Nicole Trefault^{1,*}, Rodrigo De la Iglesia^{2,*}, Mario Moreno-Pino¹, Adriana Lopes dos Santos³, Catherine G erikas Ribeiro¹, G enesis Parada-Pozo¹, Antonia Cristi¹, Dominique Marie⁴, and Daniel Vaultot^{4,3*}".

Footnote: *Corresponding authors: nicole.trefault@umayor.cl, vaultot@gmail.com

ORCID Numbers:

- Adriana Lopes dos Santos: 0000-0002-0736-4937
- Daniel Vaultot: 0000-0002-0717-5685
- Catherine G erikas Ribeiro: 0000-0003-0531-2313
- Antonia Cristi: 0000-0003-1381-8170
- Rodrigo De la Iglesia: 0000-0002-2000-8697

Typst

- <https://typst.app/>
- Own language close to Markdown
- Quarto can export to typst format

The screenshot displays the Typst editor interface. The left pane shows the source code in a typst dialect, and the right pane shows the rendered output. The rendered document is titled "Towards Swifter Interstellar Mail Delivery" and includes author information, a date, an abstract, and a diagram of Earth-to-Mars communication.

```
1 #import "template.typ": *
2 #show: paper.with(
3   title: [Towards Swifter Interstellar Mail Delivery],
4   date: [May 17th, 2022],
5   // ...
6 )
7
8 = Introduction To A-Mail Delivery
9 Our concept suggests three ways that A-Mail can be best utilized.
10
11 - First is to reduce the probability of the failure of a space mission. This
12   problem is known as the Mars problem and suggests problems with human
13   communication.
14
15 - High round-trip times Nicole Johanna on long-distance communication between Mars and Earth
16   inhibits successful human developments on the planet. In contrast, the delivery
17   speed of an A-Mail can be determined through this simple formula:
18   $ v(t) = \lim_{t \rightarrow \infty} \int_1^{\infty} c \cdot \sqrt{t^2} \, dt $
19
20 #figure(
21   image("a-mail.svg"),
22   caption: [FTL Earth-to-Mars communication enabled by Typst. Egon]
23 )
```

Towards Swifter Interstellar Mail Delivery

Johanna Swift
Delivery Institute

Egon Stellaris
Space Institute

Oliver Liam
Mail Institute

May 17th, 2022

Until there is a definitive answer to the mystery of the dead star,
please use the old postal system to submit your question and report
the location of missing letters to the P.I.

ABSTRACT

Recent advances in space-based document processing have enabled faster mail delivery between different planets of a solar system. Given the time it takes for a message to be transmitted from one planet to the next, its estimated that even a one-way trip to a distant destination could take up to one year. During these periods of interplanetary mail delivery there is a slight possibility of mail being lost in transit. This issue is considered so serious that space management employs P.I. agents to track down and retrieve lost mail. We propose A-Mail, a new anti-matter based approach that can ensure that mail loss occurring during interplanetary transit is unobservable and therefore potentially undetectable. Going even further, we extend A-Mail to predict problems and apply existing and new best practices to ensure the mail is delivered without any issues. We call this extension AI-Mail.

References: Johanna Swift, Egon Stellaris, Oliver Liam. Towards Swifter Interstellar Mail Delivery. <https://doi.org/10.7801/1209485:0>

Figure 1: FTL Earth-to-Mars communication enabled by Typst.

Building on the strong foundations of A-Mail, we

Recap

- Rmarkdown: mix text, R chunk, R output
- Compile to HTML or to PDF
- Can be used for many different purposes
- Use to document your analysis process (for papers...)
- Use Quarto rather than R markdown
- Cooperative writing is very useful

Other R topics

- Make interactive maps
- Git and GitHub - Cooperate
- Create your own package
- Create interactive applications (Shiny)
- Interact with database (MySQL, SQLite)
- Google/Amazon cloud