

Reproducible Data Analysis With R

bmRn CSM: Writing dynamic reports
with R Markdown

Martin Frigaard

2020-12-13

Rmarkdown

TEXT. CODE. OUTPUT.
(GET IT TOGETHER, PEOPLE.)



Artwork by @allison_horst

Materials



Link to slides:

<https://mjfrigaard.github.io/rep-res-rmarkdown/Index.html>

Link to exercises:

<https://mjfrigaard.github.io/rep-re-rmd-exercises/Index.html>

Getting set up



Make sure you've created a Github account

If you haven't, head over to Github and create a free account.

Follow this link:

<https://github.com/join>

The screenshot shows the GitHub sign-up page. At the top, there's a navigation bar with links for 'Why GitHub?', 'Team', 'Enterprise', 'Explore', 'Marketplace', 'Pricing', and a search bar. Below the navigation is a large button labeled 'Join GitHub'. The main section is titled 'Create your account'. It contains three input fields: 'Username *', 'Email address *', and 'Password *'. Below the password field is a note: 'Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more](#)'. There are two checkboxes: 'Email preferences' (unchecked) and 'Send me occasional product updates, announcements, and offers.' (unchecked). A 'Verify your account' section includes a CAPTCHA challenge: 'Please solve this puzzle so we know you are a real person' with a 'Verify' button below it.

Getting RStudio.Cloud Set Up



Head over to [RStudio.Cloud](#) and sign in with your Github account

The landing page for RStudio Cloud. It features a large blue header with the text "Welcome to RStudio Cloud beta". Below the header, there is a subtext "Do, share, teach and learn data science with R." and a "Get Started" button. In the top right corner, there is a "Log In" link.

The login page for RStudio Cloud. It has two input fields for "Email" and "Password", followed by a "Log in" button. Below the button are links for "Forgot your password?" and "Sign up". There are also two social login buttons: "Log in with Google" (with a G logo) and "Log in with GitHub" (with a GitHub logo).

By clicking log in, you agree to the RStudio.cloud terms of use.

Your RStudio.Cloud Workspace



This is where your new projects will show up!

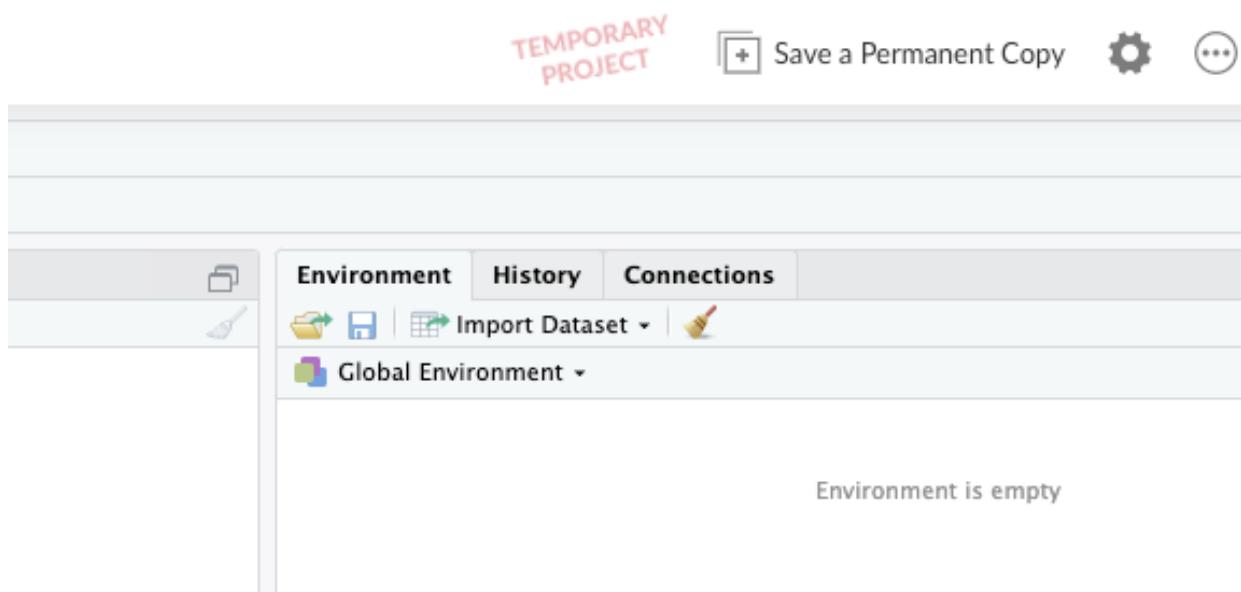
A screenshot of the RStudio Cloud web interface. The top navigation bar shows the title "RStudio Cloud" and a URL "rstudio.cloud/projects". The main content area is titled "Your Workspace" and "Projects". A sidebar on the left contains links for "Spaces" (selected), "New Space", "Learn" (Guide, What's New, Primers, Cheat Sheets, Feedback and Questions), and "Info" (Terms and Conditions, System Status). The central "Your Projects" section displays a message "no projects". A "New Project" button is available. To the right, there are sections for "Options" (Search Projects, Sort Projects: By name selected, By date created), "Capacity" (described as a personal workspace for unlimited projects), and a note about learning more in the Guide.

Click on the link below:



<https://rstudio.cloud/project/1859537>

Click *Save a Permanent Copy* and create a personal copy in your workspace



What you should see



Now you have a local copy of this RStudio project in your RStudio.Cloud workspace.

A screenshot of the RStudio Cloud interface. The top navigation bar shows "Your Workspace / bmrn-04-rmd-intro". The top right corner shows the user's name, Martin J Frigaard, and profile picture. The interface includes several panes: a "Console" pane showing R version 4.0.3 startup messages; a "Terminal" pane showing the command "/cloud/project/"; an "Environment" pane which is currently empty; and a "Files" pane showing a single item named "project" in the "Cloud" folder. The bottom left has a small status bar with the number "10".

Installing the packages



Run the `install.R` file

Copy the code below and run it in the *Console* pane and click enter/return

```
source("install.R")
```

A screenshot of the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The status bar indicates R 4.0.3. The left sidebar has tabs for Console, Terminal, and Jobs, with the Console tab active and showing the path /cloud/project/. The main area has tabs for Environment, History, Connections, and Tutorial, with the Environment tab active. The Environment pane displays the message "Environment is empty". The bottom right pane shows a file browser with tabs for Files, Plots, Packages, Help, and Viewer. It lists a "Cloud" folder containing a "project" folder, which contains an "index.html" file and a "project.Rproj" file. The "project.Rproj" file is selected, showing its size (205 B) and last modified date (Nov 4, 2020, 9:20 PM).

```
R version 4.0.3 (2020-10-10) -- "Bunny-Wunnies Freak Out"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (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.

> source("install.R")
```

What is RMarkdown?



Three technologies:

- 1) Markdown is a plain text markup language for capturing *human-readable* prose
- 2) Data manipulation/graphing/statistical language engines for computing *machine-readable* code
- 3) Multiple *output options* for creating PDFs, Word docs, PowerPoints, HTML, etc.

How R Markdown works

rmarkdown works directly with knitr

`rmarkdown` combines YAML, markdown, and R code into a markdown document and passes it to `knitr`



`knitr` uses `pandoc` (a universal document conversion tool) to generate the specified document format





Exercises

We will create an example HTML report using the R Markdown template provided by RStudio

Exercise 1: create a new RMarkdown file

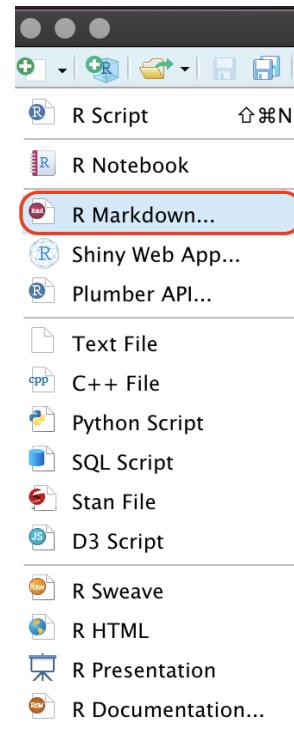


File >>>

New File >>>

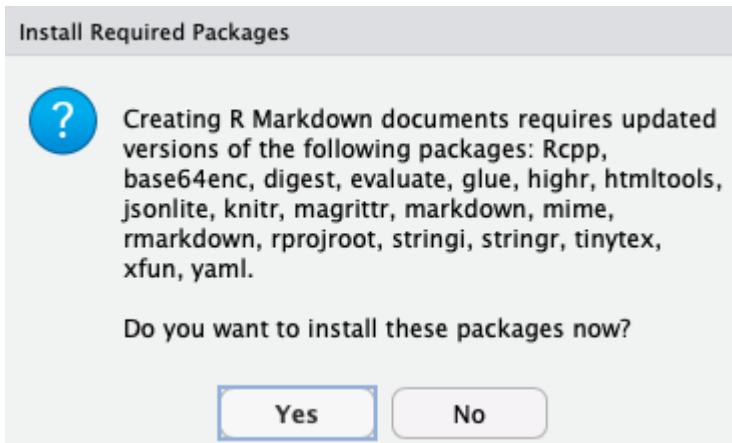
R Markdown... >>>

Or use the drop-down menu



Install required packages

RStudio.Cloud *might* ask to install the required packages for R Markdown, Click **Yes**



You will see RStudio installing the packages in the **Jobs** pane

The screenshot shows the RStudio interface with the "Jobs" tab selected in the top navigation bar. A progress bar indicates the status of "R Markdown Dependencies". The terminal output shows the installation of the "highr" package:

```
Console Terminal × Jobs × 0:07
R Markdown Dependencies
* DONE (glue)

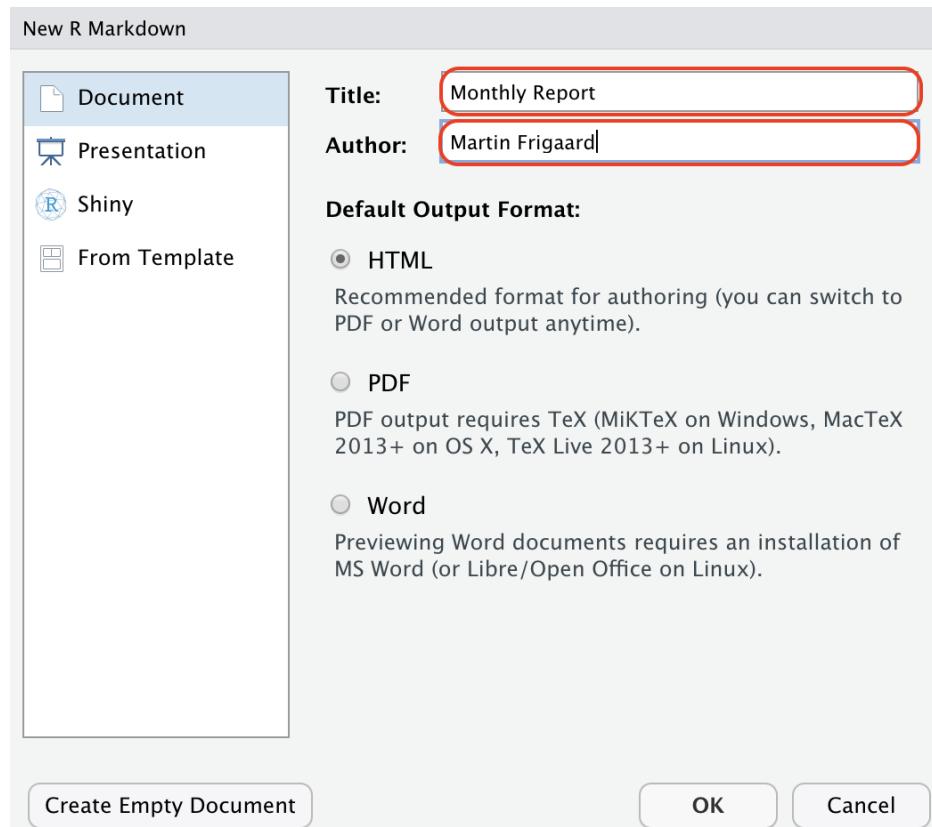
The downloaded source packages are in
  '/tmp/RtmpzyUUTc/downloaded_packages'
Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-library/4.0'
(as 'lib' is unspecified)
trying URL 'http://package-proxy/src/contrib/highr_0.8.tar.gz'
Content type 'application/x-tar' length 40478 bytes (39 KB)
downloaded 39 KB

[7/21] Installing rlang...
* installing *binary* package 'highr' ...
* DONE (highr)

The downloaded source packages are in
  '/tmp/RtmpzyUUTc/downloaded_packages'
Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-library/4.0'
(as 'lib' is unspecified)
trying URL 'http://package-proxy/src/contrib/rlang_0.4.8.tar.gz'
```

New R Markdown

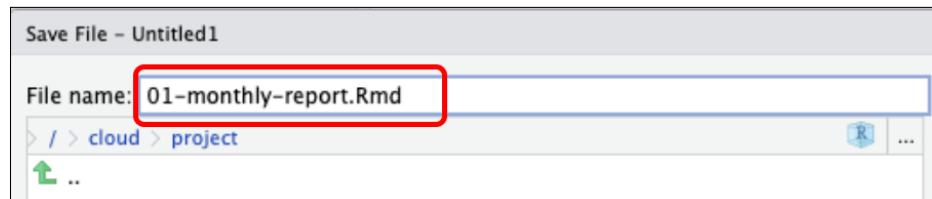
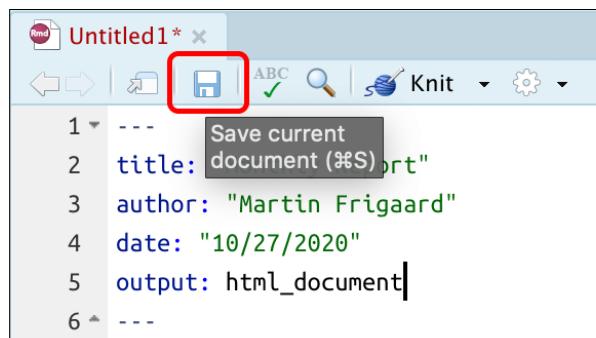
Enter *Title* and *Author* of your report and click **OK**



Save your .Rmd file

Click on the small floppy disk to save your .Rmd file

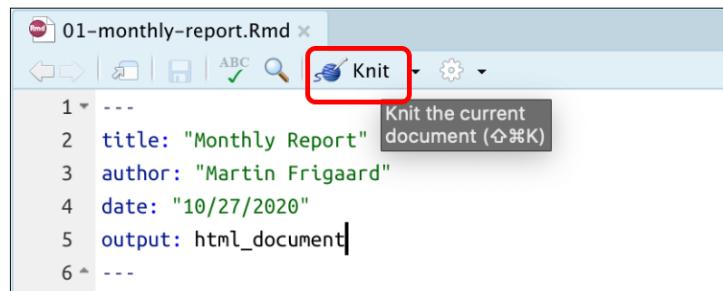
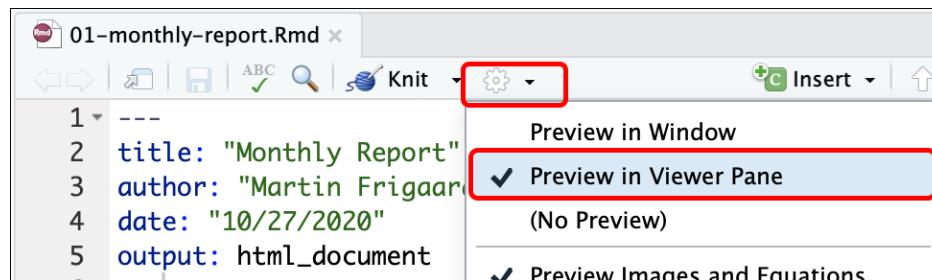
Enter a name (**01-monthly-report.Rmd**)



Knit your .Rmd file

Click on the small gear, select *Preview in Viewer Pane*

Click on the knit icon (ball of yarn)



Our First R Markdown Report!



Your Workspace / bmrn-04-rmd-intro

File Edit Code View Plots Session Build Debug Profile Tools Help

01-monthly-report.Rmd x

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "10/27/2020"  
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. For more details on using R Markdown see http://rmarkdown.rstudio.com.  
15  
16 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:  
17  
18-```{r cars}  
19 summary(cars)  
20-```  
21  
22-## Including Plots  
23  
24 You can also embed plots, for example:  
25  
26-```{r pressure, echo=FALSE}  
27 plot(pressure)  
28-```  
29  
30 Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.  
31
```

Environment History Connections Tutorial

Files Plots Packages Help Viewer

R 4.0.3

Monthly Report

Martin Frigaard
10/27/2020

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>.

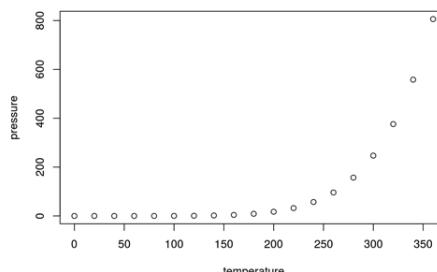
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
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:



A scatter plot showing the relationship between temperature (x-axis) and pressure (y-axis). The x-axis ranges from 0 to 350 with increments of 50. The y-axis ranges from 0 to 800 with increments of 200. The data points show a positive correlation, with most points clustered below 250 and above 300, and a few outliers at higher temperatures (around 330-340) with higher pressures (around 400-800).

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.



How R Markdown Works (under the hood)

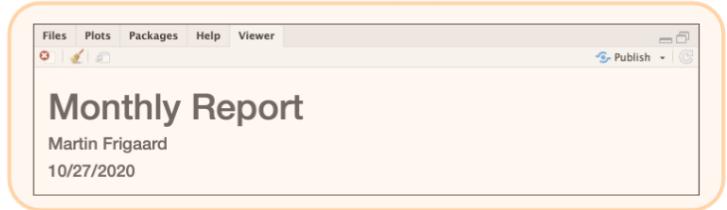
R Markdown is made up of three elements

YAML header = metadata

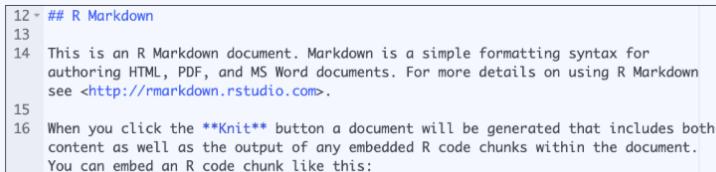


A screenshot of the RStudio interface showing the YAML header of an R Markdown file named "01-monthly-report.Rmd". The code is:

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "10/27/2020"  
5 output: html_document  
6 ---  
7
```

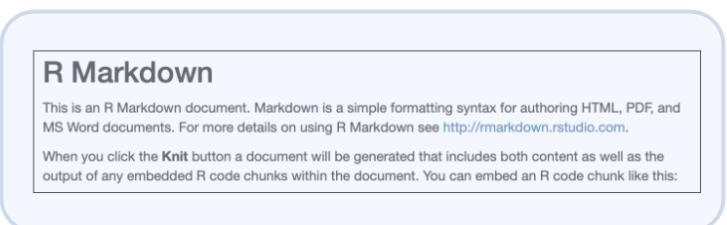


Markdown = prose



An R Markdown document containing prose and an R code chunk. The prose explains what R Markdown is and how it works. The R code chunk is used to embed an R code chunk.

```
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. For more details on using R Markdown  
see <http://rmarkdown.rstudio.com>.  
15  
16 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:  
17
```



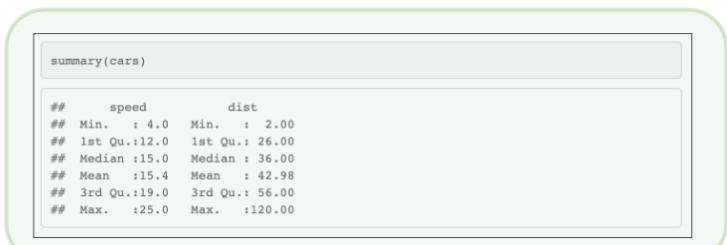
The generated HTML output shows the prose and the R code chunk. The R code chunk is rendered as a block of text within the HTML page.

Code chunks = R code



An R Markdown code chunk starting with `summary(cars)`.

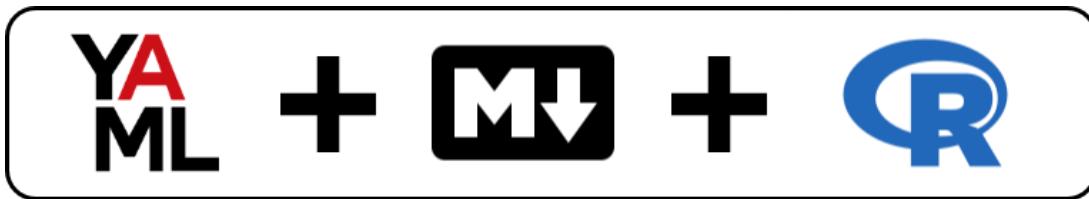
```
18 ````{r cars}  
19 summary(cars)  
20 ````
```



The generated HTML output shows the output of the `summary(cars)` R code chunk, which is a table of statistics for the "cars" dataset.

	speed	dist
## Min. :	4.0	2.00
## 1st Qu.:	12.0	18.00
## Median :	15.0	36.00
## Mean :	15.4	42.98
## 3rd Qu.:	19.0	56.00
## Max. :	25.0	120.00

Rmarkdown combines metadata, markdown, and R code



The result is a file framework for creating reproducible reports using YAML, Markdown, and computer code

- .yaml = Metadata
- .md = Prose
- .R = Code

R Markdown: YAML



.yaml = Metadata

~~.md = Prose~~

~~.R = Code~~

YAML is a human friendly data serialization standard for all programming languages.

YAML stands for '*YAML Ain't Markup Language*'
(funny, huh?)

R Markdown: YAML

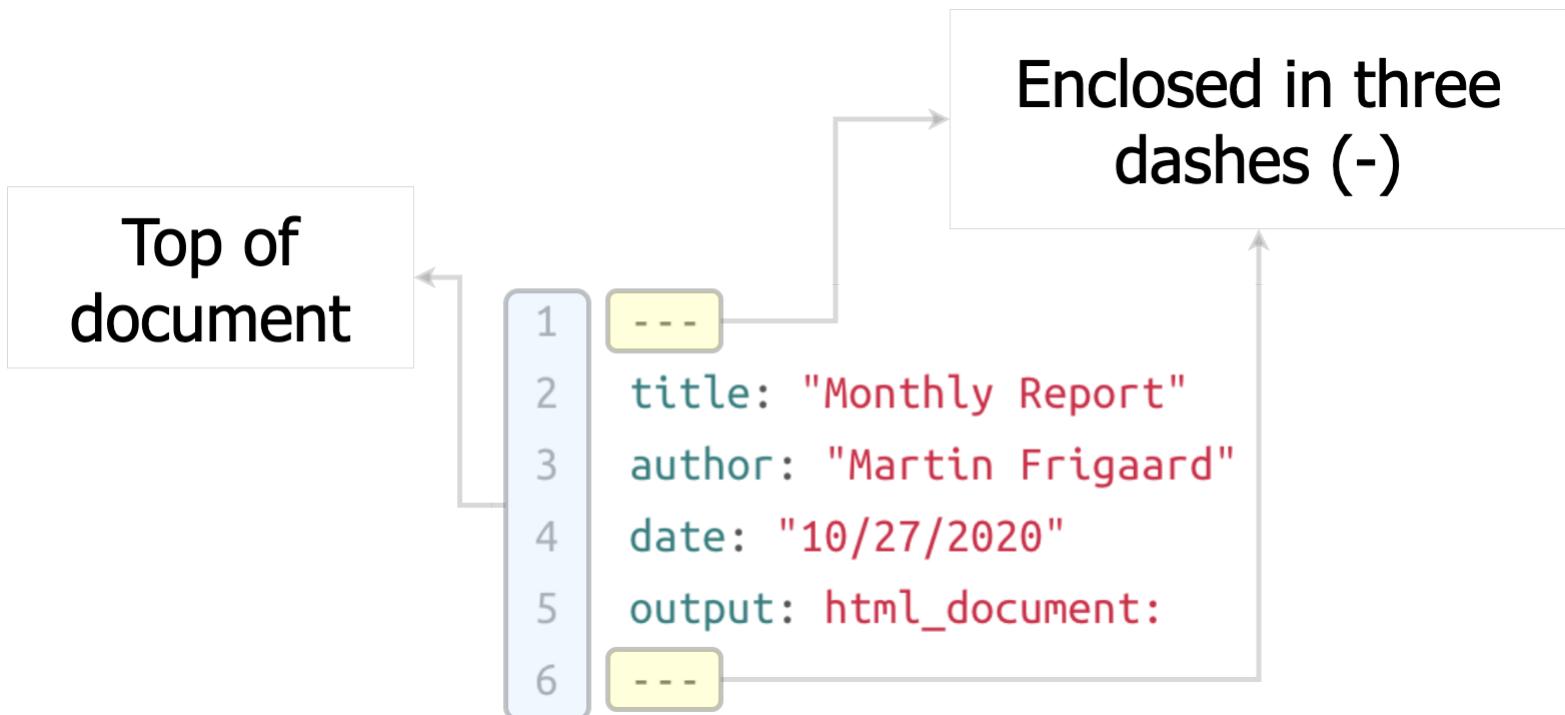


YAML contains the information about the document we're going to create

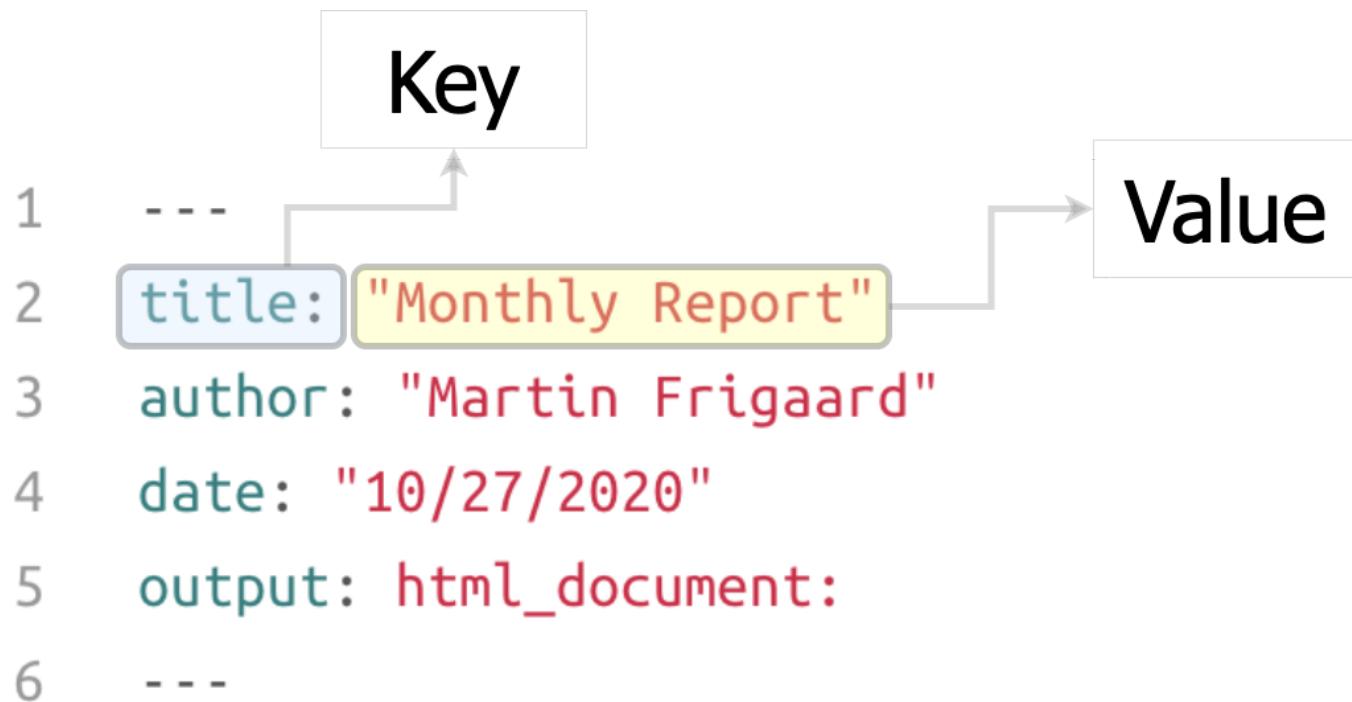
```
---
```

```
title: "Monthly Report"
author: "Martin Frigaard"
date: "10/27/2020"
output: html_document
---
```

R Markdown: YAML format



R Markdown: YAML format



R Markdown: YAML



There are many YAML arguments and options

Indentation matters in YAML!!

| Check out the [YAML Fieldguide](#) for a comprehensive list

Example YAML output options

Table of contents:

toc: logical (true or false)

toc_float: logical (true or false)

toc_depth: set numerically 0 - 6



Exercise 2: create a floating table of contents

Change the **output** in the YAML header to the following:

```
output:  
  html_document:  
    toc: yes  
    toc_float: true
```

Knit the document again



YAML output options: table of contents

Floating table of contents (rendered)

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "10/27/2020"  
5 output:  
6   html_document:  
7     toc: yes  
8     toc_float: true
```

R MarkdownMonthly Report

Including Plots

Monthly Report

Martin Frigaard
10/27/2020

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:



Exercise 3: text highlighting and themes

Add the following two options to your YAML header

```
output:  
  html_document:  
    toc: yes  
    toc_float: yes  
    highlight: zenburn  
    theme: united
```

Knit the document again



YAML: text highlighting and theme options

Text highlighting and theme options (rendered)

The diagram illustrates the relationship between YAML configuration and its rendered output in R Markdown. On the left, a box contains the YAML code:

```
1 - ---  
2   title: "Monthly Report"  
3   author: "Martin Frigaard"  
4   date: "10/27/2020"  
5   output:  
6     html_document:  
7       toc: yes  
8       toc_float: yes  
9       highlight: zenburn  
10      theme: united  
11 - ---
```

Two arrows point from the 'highlight: zenburn' and 'theme: united' lines in the YAML to the corresponding sections in the rendered output. The rendered output shows a screenshot of the RStudio interface with the following content:

Monthly Report
Martin Frigaard
10/27/2020

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:

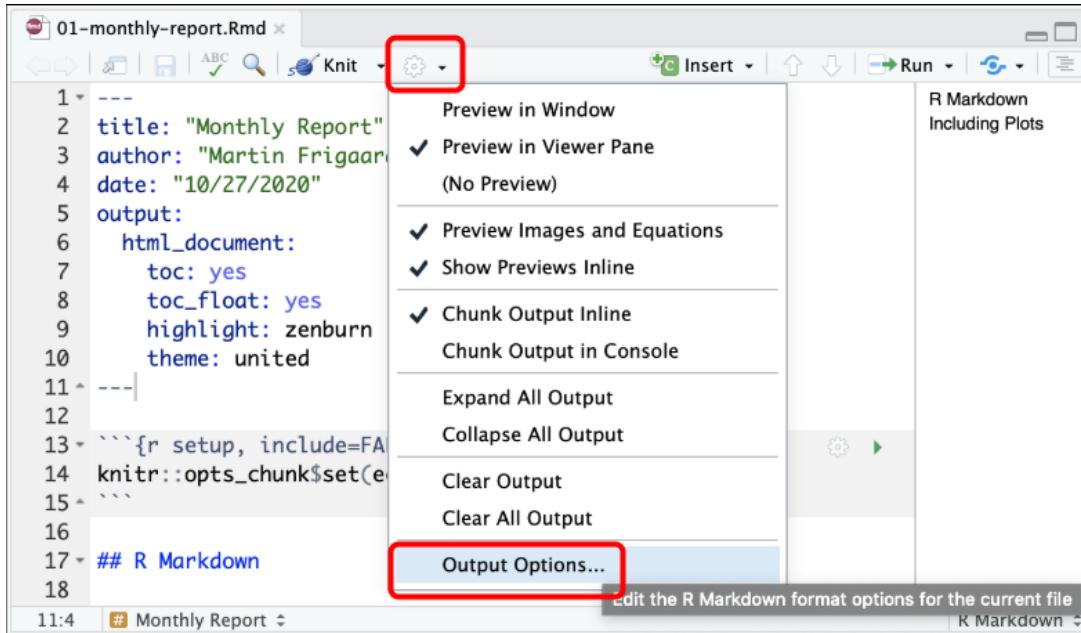
```
summary(cars)
```

	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



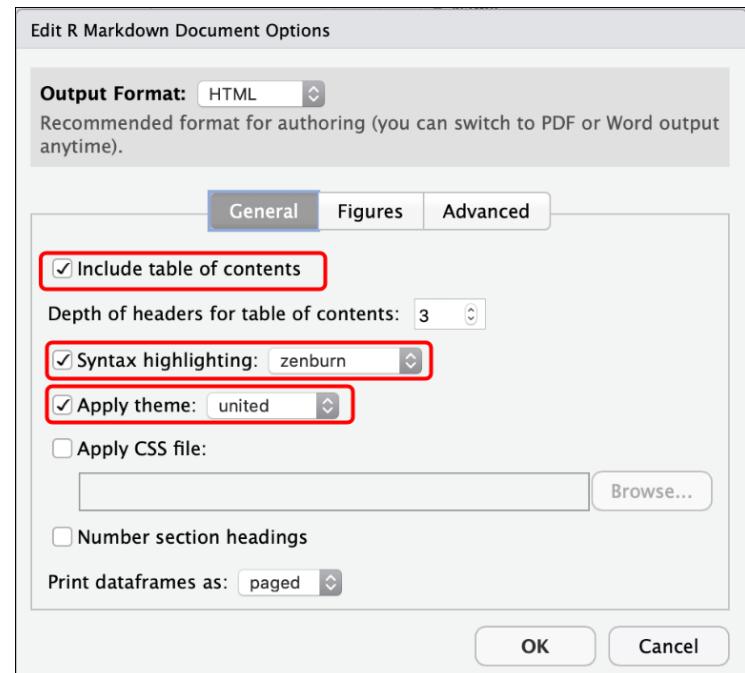
More YAML options

You can change the YAML contents using the settings (small gear)



Edit R Markdown Document Options

This window gives us the ability to manually change some of the YAML settings (but not all of them!)



YAML Parameters

YAML parameters can be referred to throughout the document

Create params in YAML header

```
params:  
  param1: x  
  param2: y  
  data: df
```

Refer to params in .Rmd document

```
params$param1  
params$param2  
params$data
```



Exercise 4: Using YAML parameters

Add the following `params` option in the YAML header

```
params:  
  small_pressure: !r head(pressure)
```

Add this code to the end of the document

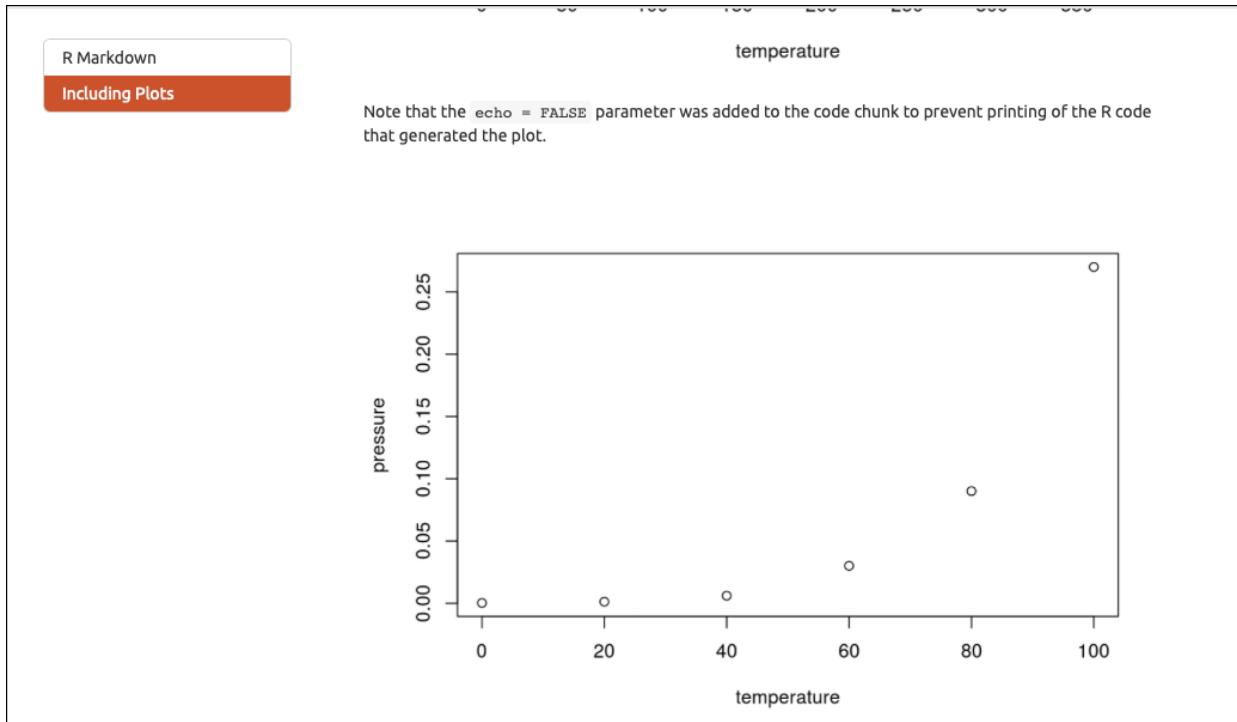
```
```{r small_pressure, echo=FALSE}  
plot(params$small_pressure)
```
```

Knit the document again



See our new plot with the params

We can see the new plot with the reduced sample size



YAML output formats



| Function | Outputs |
|--|---------------------|
| <code>html_document()</code> | HTML document |
| <code>pdf_document()</code> | PDF document |
| <code>word_document()</code> | Word document |
| <code>odt_document()</code> | ODT document |
| <code>rtf_document()</code> | RTF document |
| <code>md_document()</code> | Markdown document |
| <code>slidy_presentation()</code> | Slidy Slides (HTML) |
| <code>beamer_presentation()</code> | Beamer Slides (PDF) |
| <code>ioslides_presentation()</code> | ioslides (HTML) |
| <code>powerpoint_presentation()</code> | PowerPoint (pptx) |

R Markdown

~~.yaml = Metadata~~



.md = Prose

~~.R = Code~~

Basic Markdown Syntax



Italics & Bold

```
*italic*    **bold**  
_italic_    __bold__
```

italic **bold**
italic **bold**

Basic Markdown Syntax



Headers

```
# Header 1  
## Header 2  
### Header 3
```

Header 1

Header 2

Header 3

Basic Markdown Syntax



Bullets & Numeric Lists

```
* Item 1
* Item 2
    + Item 2a
    + Item 2b

1. Item 1
2. Item 2
```

- Item 1
- Item 2
 - Item 2a
 - Item 2b

1. Item 1
2. Item 2

Basic Markdown Syntax



Hyperlinks

```
https://www.biomerin.com/
```

```
[linked phrase] (https://www.biomerin.com/)
```

becomes...

linked phrase

Basic Markdown Syntax



Images

```
![] (https://www.r-project.org/logo/Rlogo.png)
```

```
![optional caption] (https://www.r-project.org/logo/Rlogo.png)
```

becomes...



Basic Markdown Syntax



Math Equations

```
$equation$
```

```
$$ equation $$
```

equation

equation

Basic Markdown Syntax



Super scripts & Strike-through

```
superscript^2^  
~~strikethrough~~
```

superscript^2^

~~strikethrough~~

Exercise 5: Markdown Formatting

Delete the top portion of the markdown in `01-monthly-report.Rmd`.

Delete

```
1 --
2 title: "Monthly Report"
3 author: "Martin Frigaard"
4 date: "10/27/2020"
5 output:
6   html_document:
7     toc: yes
8     toc_float: yes
9     highlight: zenburn
10    theme: united
11 params:
12   small_pressure: !r head(pressure)
13 ---
14
15 ``{r setup, include=FALSE}
16 knitr::opts_chunk$set(echo = TRUE)
17 ```
18
19 ## R Markdown
20
21 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>.
22
23 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:
```



Exercise 5: Markdown Formatting

Add the text below to your report

This is a monthly report generated with RMarkdown, a literate programming tool for combining text and code.

Include the following formatting:

1. make `monthly report` italic
2. include this hyperlink for Rmarkdown: <https://rmarkdown.rstudio.com/>
3. format `code` as code.

Knit the document when you're finished



Exercise 5: Markdown Formatting (rendered)

The screenshot shows the RStudio interface with an R Markdown file named "01-monthly-report.Rmd" on the left and its rendered output on the right.

Code Editor (Left):

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "10/27/2020"  
5 output:  
6   html_document:  
7     toc: yes  
8     toc_float: yes  
9     highlight: zenburn  
10    theme: united  
11 params:  
12   small_pressure: !r head(pressure)  
13 ---  
14  
15 ````{r setup, include=FALSE}  
16 knitr::opts_chunk$set(echo = TRUE)  
17 ````  
18  
19 ## R Markdown  
20  
21 This is a *monthly report* generated with  
[RMarkdown](https://rmarkdown.rstudio.com/), a literate programming tool  
for combining text and `code`.  
22  
23 ````{r cars}  
24 summary(cars)  
25 ````
```

RStudio Environment (Top Bar):

Environment History Connections Tutorial

Viewer (Right):

Monthly Report

Martin Frigaard
10/27/2020

R Markdown

This is a *monthly report* generated with **RMarkdown**, a literate programming tool for combining text and `code`.

```
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 |



Exercise 6: Tabssets



Remove the `toc` and `toc_float` options from your YAML header

```
output:  
  html_document:  
    highlight: zenburn  
    theme: united  
params:  
  small_pressure: !r head(pressure)
```

Exercise 6: Tabssets



Make the following changes to the R Markdown header sections

```
18
19 - ## R Markdown {.tabset}
20
21 This is a **monthly report** generated with
[RMarkdown](https://rmarkdown.rstudio.com/), a literate programming tool
for combining text and `code`.
22
23 - ### Summary
24
25 - ```{r cars}
26 summary(cars)
27 ```
28
29 - ### Including Plots
30
31 You can also embed plots, for example:
32
33 - ```{r pressure, echo=FALSE}
34 plot(pressure)
35 ```
36
37 Note that the `echo = FALSE` parameter was added to the code chunk to
prevent printing of the R code that generated the plot.
38
39 - ```{r small_pressure, echo=FALSE}
40 plot(params$small_pressure)
41 ```

Include tabset in curly brackets
Add 'Summary' level 3 header
Change 'Including Plots' to level 3 header
```

Knit the document again

Exercise 6: Tabs (rendered)



Tab 1

A screenshot of the RStudio interface. On the left, the code editor shows an R Markdown file named "01-monthly-report.Rmd". The code includes YAML front matter, a code chunk for "cars", and sections for "Summary" and "Including Plots". It also contains a note about using `echo = FALSE` in plots. On the right, the "Viewer" panel displays the rendered "Monthly Report". The title is "Monthly Report" by "Martin Frigaard" on "11/5/2020". Below the title is the section "R Markdown" with a brief description. There are two tabs: "Summary" (selected) and "Including Plots". Under "Summary", the `summary(cars)` command is shown, followed by its output:

| | speed | dist |
|---------|-------|----------------|
| Min. | 4.0 | 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 |

Exercise 6: Tabs (rendered)



Tab 2

A screenshot of the RStudio interface. On the left, the code editor shows an R Markdown file named "01-monthly-report.Rmd". The code includes YAML front matter, a title section, and several code chunks. One chunk uses the `knitr::opts_chunk\$set(echo = TRUE)` option. Another chunk shows a note about using `echo = FALSE` to prevent printing of the R code that generates a plot. A third chunk plots pressure against temperature. On the right, the "Viewer" panel displays the rendered "Monthly Report". The report has a title "Monthly Report", author "Martin Frigaard", and date "11/5/2020". It includes a section titled "R Markdown" with a note about being a monthly report generated with RMarkdown. Below this, there are two tabs: "Summary" (which is selected) and "Including Plots". Under "Summary", there is a note about embedding plots. The "Including Plots" tab shows a scatter plot of pressure versus temperature with data points following a positive linear trend.

01-monthly-report.Rmd

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "11/5/2020"  
5 output:  
6   html_document:  
7     highlight: zenburn  
8     theme: united  
9 params:  
10    small_pressure: !r head(pressure)  
11 ---  
12  
13 ```{r setup, include=FALSE}  
14 knitr::opts_chunk$set(echo = TRUE)  
15 ```  
16  
17 ## R Markdown {.tabset}  
18  
19 This is a **monthly report** generated with [RMarkdown](https://rmarkdown.rstudio.com/), a literate programming tool for combining text and code.  
20  
21 ### Summary  
22  
23 ```{r cars}  
24 summary(cars)  
25 ```  
26  
27 ## Including Plots  
28  
29 You can also embed plots, for example:  
30  
31 ```{r pressure, echo=FALSE}  
32 plot(pressure)  
33 ```  
34  
35 Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.  
36  
37 ```{r small_pressure, echo=FALSE}  
38 plot(params$small_pressure)  
39 ```
```

Environment History Connections Tutorial

Files Plots Packages Help Viewer

Monthly Report

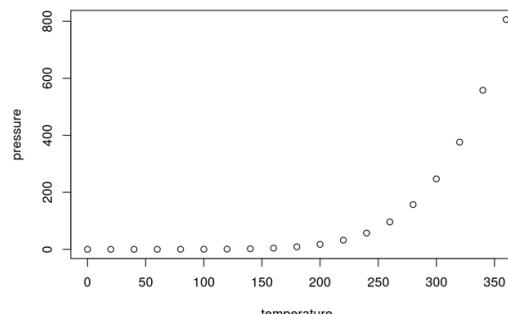
Martin Frigaard
11/5/2020

R Markdown

This is a **monthly report** generated with [RMarkdown](#), a literate programming tool for combining text and `code`.

[Summary](#) [Including Plots](#)

You can also embed plots, for example:



A scatter plot showing the relationship between temperature (x-axis, ranging from 0 to 350) and pressure (y-axis, ranging from 0 to 800). The data points show a strong positive correlation, starting near (0, 0) and ending near (350, 800).

| temperature | pressure |
|-------------|----------|
| 0 | 0 |
| 25 | 25 |
| 50 | 50 |
| 75 | 75 |
| 100 | 100 |
| 125 | 125 |
| 150 | 150 |
| 175 | 175 |
| 200 | 200 |
| 225 | 225 |
| 250 | 250 |
| 275 | 275 |
| 300 | 300 |
| 325 | 325 |
| 350 | 800 |

R Markdown



~~.yaml = Metadata~~

~~.md = Prose~~

.R = Code

Code chunks (**setup**)



The first bit of R code in our RMarkdown file is the **setup** chunk

```
```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
```



Chunks named '**setup**' are special because they can set global options

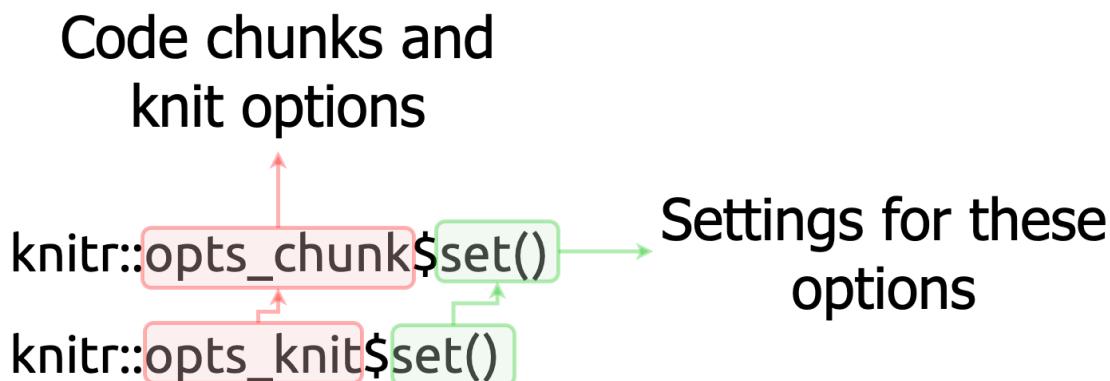
'include=FALSE' means this code is run, but not displayed

Code chunks (setup)



R Markdown document options come from the `knitr` package

We can access both the with syntax below:



Code chunks (`setup`)



The `echo=TRUE` option controls whether we want to display the code in the code chunk

Other common options regarding code are `eval`, `tidy`, `error`, `message`, and `warning`

Advanced options can control language engines (`engine`), caching (`cache`, `dependson`), and plot animations (`fig.show`)

Code chunks (setup)



Many options for code chunks

| Option | Document Effect |
|---------|---|
| include | run code, but do/don't print code and results |
| eval | do/don't evaluate the code |
| echo | run code, but don't print code |
| message | do/don't print messages (e.g. from functions) |
| warning | do/don't print warnings |

Code chunks



```
1   ```{r pressure, echo=FALSE}
2   plot(pressure)
3   ````
```

Code chunk fences



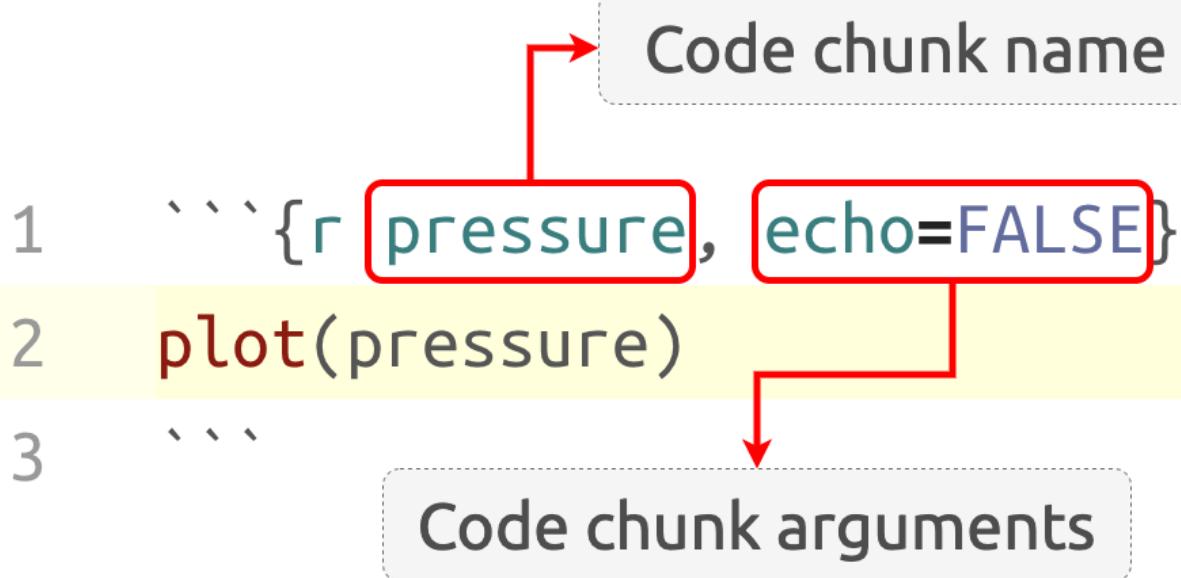
Code chunk fences

```
1   ````{r pressure, echo=FALSE}
```

```
2   plot(pressure)
```

```
3   ````
```

Code chunk names and arguments

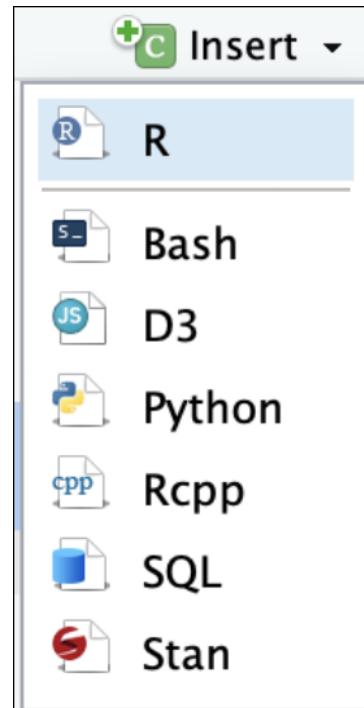


See the knitr web page for complete list of options

Inserting code chunks



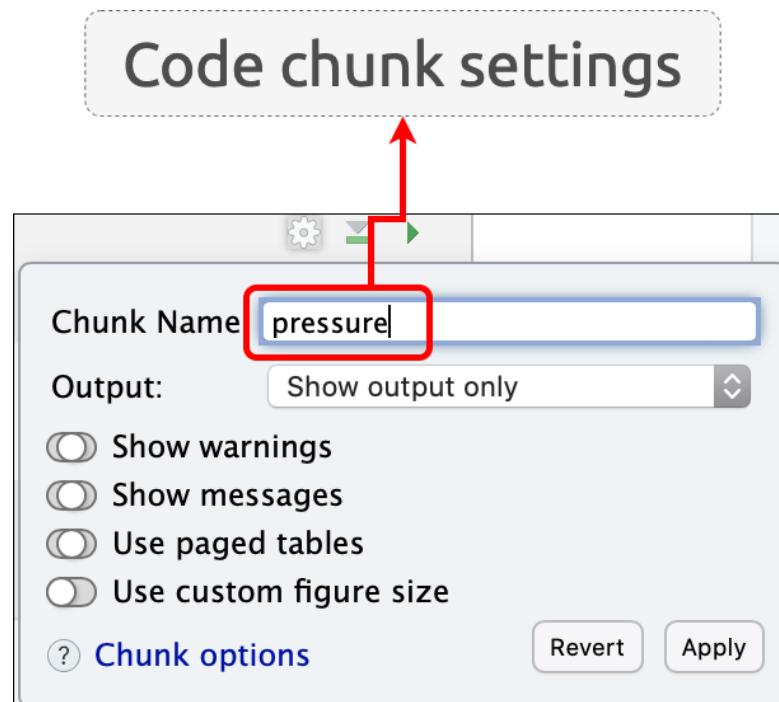
Use keyboard shortcuts **CMD/CTRL + I** or
ALT/OPTION + I



Edit code chunk options



You can edit code chunk options using the icon (small gear)



Code Chunk Engines



More and more code engines all the time

```
names(knitr::knit_engines$get())
```

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

Even SAS!

Exercise 7: code chunks (kable)

Create a new heading and code chunk

Create a new **Tables** level three header under the **Summary** heading,

```
### Tables
```

Insert the following code chunk under **Tables** (*manually, with the keyboard short-cut, or use the "Insert" button*)

```
```{r kable}
knitr::kable(params$small_pressure)
```

```

Knit the document



Exercise 7: code chunks (kable rendered)



We can see the `small_pressure` parameter from the YAML has been rendered in the new **Tables** tab. `kable` tables are great for presenting small, summary tables.

The screenshot shows the R Markdown interface. On the left, a code editor window displays the following R code:## Tables
```{r kable}
knitr::kable(params\$small\_pressure)
```
A red box highlights the first two lines of code. A red arrow points from this box to the 'Tables' tab in the navigation bar on the right. The 'Tables' tab is selected, and the rendered output is displayed below it. The output shows the following table:| | temperature | pressure |
| --- | --- | --- |
| 0 | 0.0002 |
| 20 | 0.0012 |
| 40 | 0.0060 |
| 60 | 0.0300 |
| 80 | 0.0900 |
| 100 | 0.2700 |

Read more about `kable` table options [here](#)

Exercise 8: code chunks (paged)

We are going to repeat the process above, but with a larger table (`mtcars`)

Insert the following code chunk above the `knitr::kable()` output:

```
### Tables
```

```
```{r paged}
rmarkdown::paged_table(mtcars)
```
```



```
```{r kable}
knitr::kable(params$small_pressure)
```
```



Knit the document

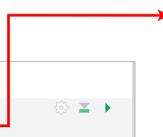


Exercise 8: code chunk (paged rendered)

Paged tables are great for larger datasets

```
### Tables
```{r paged}
rmarkdown::paged_table(mtcars)
```
```{r kable}
knitr::kable(params$small_pressure)
```

```



rmarkdown::paged_table(mtcars)

R Markdown

This is a monthly report generated with **RMarkdown**, a literate programming tool for combining text and code .

[Summary](#) [Tables](#) [Including Plots](#)

```
rmarkdown::paged_table(mtcars)
```

| | mpg | cyl | disp | hp | drat | wt | qsec | vs | am |
|-------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| | <dbl> |
| Mazda RX4 | 21.0 | 6 | 160.0 | 110 | 3.90 | 2.620 | 16.46 | 0 | 1 |
| Mazda RX4 Wag | 21.0 | 6 | 160.0 | 110 | 3.90 | 2.875 | 17.02 | 0 | 1 |
| Datsun 710 | 22.8 | 4 | 108.0 | 93 | 3.85 | 2.320 | 18.61 | 1 | 1 |
| Hornet 4 Drive | 21.4 | 6 | 258.0 | 110 | 3.08 | 3.215 | 19.44 | 1 | 0 |
| Hornet Sportabout | 18.7 | 8 | 360.0 | 175 | 3.15 | 3.440 | 17.02 | 0 | 0 |
| Valiant | 18.1 | 6 | 225.0 | 105 | 2.76 | 3.460 | 20.22 | 1 | 0 |
| Duster 360 | 14.3 | 8 | 360.0 | 245 | 3.21 | 3.570 | 15.84 | 0 | 0 |
| Merc 240D | 24.4 | 4 | 146.7 | 62 | 3.69 | 3.190 | 20.00 | 1 | 0 |
| Merc 230 | 22.8 | 4 | 140.8 | 95 | 3.92 | 3.150 | 22.90 | 1 | 0 |
| Merc 280 | 19.2 | 6 | 167.6 | 123 | 3.92 | 3.440 | 18.30 | 1 | 0 |

1-10 of 32 rows | 1-10 of 12 columns

Previous [1](#) [2](#) [3](#) [4](#) Next



Exercise 8: paged tables



R Markdown

This is a **monthly report** generated with **RMarkdown**, a literate programming tool for combining text and `code`.

Summary

Tables

Including Plots

```
rmarkdown::paged_table(mtcars)
```

| | mpg
<dbl> | cyl
<dbl> | disp
<dbl> | hp
<dbl> | drat
<dbl> | wt
<dbl> | qsec
<dbl> | vs
<dbl> | am
<dbl> | ► |
|-------------------|--------------|--------------|---------------|-------------|---------------|-------------|---------------|-------------|-------------|---|
| Mazda RX4 | 21.0 | 6 | 160.0 | 110 | 3.90 | 2.620 | 16.46 | 0 | 1 | |
| Mazda RX4 Wag | 21.0 | 6 | 160.0 | 110 | 3.90 | 2.875 | 17.02 | 0 | 1 | |
| Datsun 710 | 22.8 | 4 | 108.0 | 93 | 3.85 | 2.320 | 18.61 | 1 | 1 | |
| Hornet 4 Drive | 21.4 | 6 | 258.0 | 110 | 3.08 | 3.215 | 19.44 | 1 | 0 | |
| Hornet Sportabout | 18.7 | 8 | 360.0 | 175 | 3.15 | 3.440 | 17.02 | 0 | 0 | |
| Valiant | 18.1 | 6 | 225.0 | 105 | 2.76 | 3.460 | 20.22 | 1 | 0 | |
| Duster 360 | 14.3 | 8 | 360.0 | 245 | 3.21 | 3.570 | 15.84 | 0 | 0 | |
| Merc 240D | 24.4 | 4 | 146.7 | 62 | 3.69 | 3.190 | 20.00 | 1 | 0 | |
| Merc 230 | 22.8 | 4 | 140.8 | 95 | 3.92 | 3.150 | 22.90 | 1 | 0 | |
| Merc 280 | 19.2 | 6 | 167.6 | 123 | 3.92 | 3.440 | 18.30 | 1 | 0 | |

1-10 of 32 rows | 1-10 of 12 columns

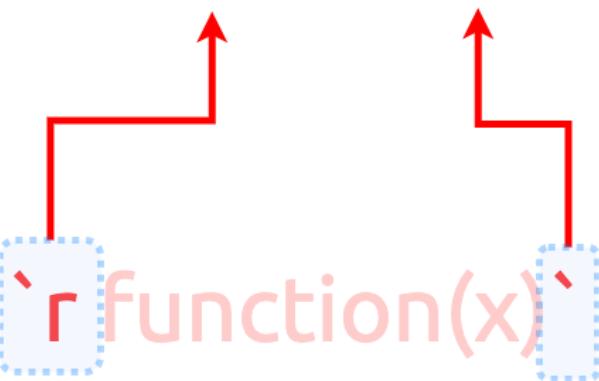
Previous **1** **2** **3** **4** Next

Inline R Code



R Markdown also supports inline R code

Grave accent or back-tick
around code



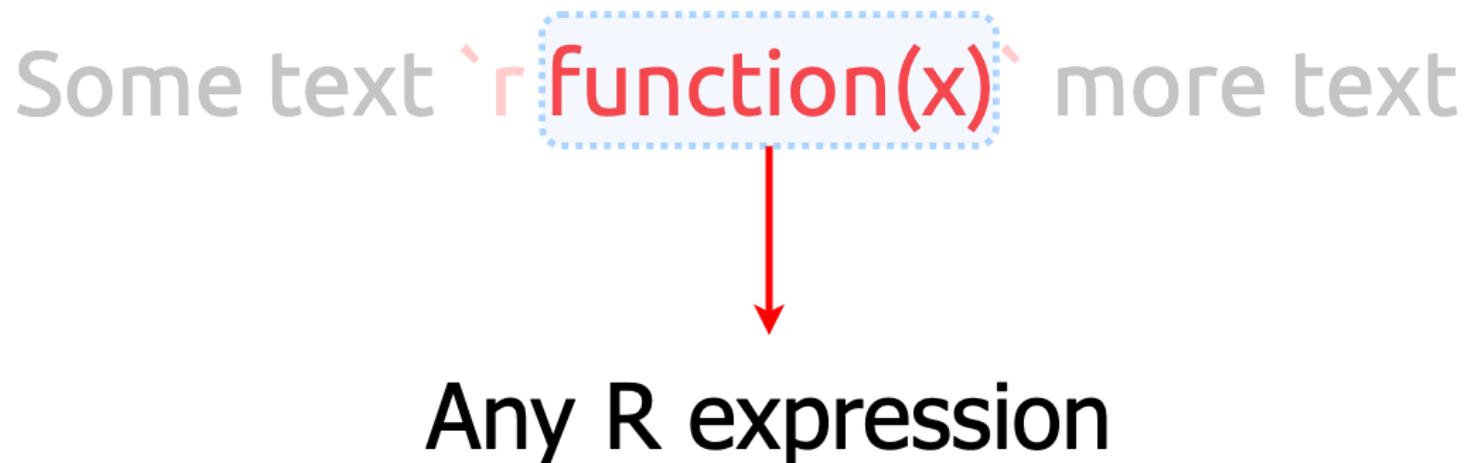
The diagram illustrates the use of inline R code within a sentence. The text "Some text" is in gray, followed by a blue dashed box containing a red grave accent (`) and the letter "r". To the right of the "r" is the text "function(x)" in red, followed by another blue dashed box containing a red grave accent (`). The text "more text" is in gray. Two red arrows point from the grave accents in the boxes up towards the "r" and "x" characters, indicating they are part of the same inline R code block.

Some text `r function(x)` more text

Inline R Code



Inline R code allows us to include summaries of our analysis in the report



Exercise 9: Add Inline Code



We're going to add a Pearson correlation between speed and stopping distance to the **01-monthy-report.Rmd**

Include the following code under the **Summary** level three header

```
The correlation between speed and stopping distance  
is `r cor(x = cars$speed, y = cars$dist)`
```

Knit the document again

Exercise 9: Add Inline Code (rendered)



Monthly Report

Martin Frigaard
10/27/2020

R Markdown

This is a monthly report generated with RMarkdown, a literate programming tool for combining text and code .

Summary Tables Including Plots

The correlation between speed and stopping distance is 0.8068949

```
## Summary
```

The correlation between speed and stopping distance is `r cor(x = cars\$speed, y = cars\$dist)`

```
```{r cars}
summary(cars)
````
```

```
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
```

Make cool stuff in R Markdown!

bookdown

blogdown

these slides!



Resources



- **YAML**: check out the [yamlthis package](#) for tools and documentation for working with YAML
- **Markdown**: [Commonmark](#) has a quick ten-twenty minute tutorial on markdown.
- **R Markdown**: A comprehensive but friendly introduction to R Markdown and friends. Free online!
- **R for Data Science**: A comprehensive but friendly introduction to the tidyverse. Free online.
- **R Markdown for Scientists**: R Markdown for Scientists workshop material