Report Reproducibly with

Slides CC BY-SARStudio

Navigate to the **06-Report** folder.

Open **06-Report-Exercises.Rmd**

R Markdown

R Markdown

Plain text file with 3 types of content:

A YAML header surrounded by ———

Text in markdown

Code chunks surrounded by ```

CC BY-SA RStudio

How it works

CC BY-SA RStudio

Logistics

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

CC BY-SA RStudio

Logistics

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

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

CC BY-SA RStudio

knitr is multi-lingual!

- ☑ SAS
- ☑ python
- ☑ more

CC BY-SA RStudio

engine

python

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

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

Some python code:

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

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

CC BY-SA RStudio

Reticulate

Python in R Markdown

(Optional) Build Python env to use.

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

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

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

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

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

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

Output displays below chunk, including matplotlib plots.

python.Rmd

Knit Insert Run

```{r setup, include = FALSE}
library(reticulate)
virtualenv_create("fmri-proj")
py_install("seaborn", envname = "fmri-proj")
use_virtualenv("fmri-proj")
```

```{python, echo = FALSE}
import seaborn as sns
fmri = sns.load_dataset("fmri")
```

```{r}
f1 <- subset(py$fmri, region == "parietal")
```

```{python}
import matplotlib as mpl
sns.lmplot("timepoint","signal", data=r.f1)
mpl.pyplot.show()
```

R Console

21:4 (Top Level) R Markdown

A package for using R and Python together.

bookdown.org/yihui/rmarkdown/

ONLINE, FREE

CC BY-SA RStudio

Markdown

R-Notebook.Rmd

Preview Insert

```
---
title: "R Notebook"
output: html_notebook
---

Text written in **markdown**

```{r}
# code written in R
(x <- rnorm(7))
```

[1] -1.2 1.0 -0.5 0.9 -0.6 -1.1 -1.5

Text written in _markdown_

```{r}
# code written in R
```
```

16:20 Chunk 2 R Markdown

Text in markdown

Headers

Use # to create headers.
Multiple #'s create lower level headers.

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

Header 1

Header 2

Header 3

Header 4

Header 5

Header 6

CC BY-SA RStudio

Text

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

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

```
Text
_italics_
**bold**
`code`
```

Text
italics
bold
`code`

CC BY-SA RStudio

Lists

Use asterisks to make bullet points.
Use numbers to make numbered lists.

```
Bullets

* bullet 1
* bullet 2

Numbered list

1. item 1
2. item 2
```

Bullets

- bullet 1
- bullet 2

Numbered list

1. item 1
2. item 2

CC BY-SARStudio

Hyperlinks

Use brackets to denote a link.
Place the URL in parentheses.

```
This is a
[link](www.git.com .
)
```

This is a link.

CC BY-SA RStudio

Images

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

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

```
![](images/1.png)
The RStudio logo.
```

CC BY-SA RStudio

Equations

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

```
According to
Einstein,
$E=mc^{2}$
```

According to Einstein, $E = mc^2$

CC BY-SA RStudio

Equation blocks

Use two $'s to make centered equation blocks.

```
According to
Einstein,


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

According to
Einstein,

$$E = mc^2$$

CC BY-SA RStudio

Markdown

Dictionary of formatting cues.

on back of RMarkdown cheat sheet

IDE Reference

Go to **Help > Markdown Quick Reference**

Code

R-Notebook.Rmd

Preview Insert

```
---
title: "R Notebook"
output: html_notebook
---

Text written in **markdown**

```{r}
# code written in R
(x <- rnorm(7))
```

[1] -1.2 1.0 -0.5 0.9 -0.6 -1.1 -1.5

Text written in _markdown_

```{r}
# code written in R
```
```

16:20 Chunk 2 R Markdown

Code chunks surrounded by ` ` `

Code chunks

Insert a chunk of R code with

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

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

CC BY-SA RStudio

Code chunks

Insert a chunk of R code with

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

⌘ + Opt + i (Mac)

Ctrl + Alt + i (PC)

CC BY-SA RStudio

chunk options

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

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

Here's some code

```
dim(iris)
```

```
## [1] 150   5
```

CC BY-SA RStudio

echo

Add options in the brackets after r.

echo = FALSE hides the code.

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

Here's some code

```
## [1] 150   5
```

Very useful for plots

CC BY-SARStudio

eval

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

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

Here's some code

```
dim(iris)
```


CC BY-SA RStudio

include

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

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

Here's some code

CC BY-SA RStudio

fig.height, fig.width

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

```
Here’s a plot
```{r echo=FALSE, fig.width=3, fig.height=5
hist(iris[[2]])
```
```

CC BY-SARStudio

Pop Quiz

Do you notice the **TODOs** in 06-RMarkdown-Exercises.Rmd?

Do you notice the long **setup** chunk**?**

CC BY-SA RStudio

Inline code

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

```
Today is
`r Sys.Date()`.
```

Today is 2015-04-16.

CC BY-SA RStudio

Inline code

Code whose results are inserted into the text.

```
Today is `r Sys.Date()`.
```

Surround with `r `

Code to run. Only the result will be included.

CC BY-SA RStudio

Your Turn 1

In **06-RMarkdown-Exercises.Rmd**:

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

CC BY-SA RStudio

YAML

R-Notebook.Rmd

Preview Insert

```
---
title: "R Notebook"
output: html_notebook
---

Text written in **markdown**

```{r}
# code written in R
(x <- rnorm(7))
```

[1] -1.2 1.0 -0.5 0.9 -0.6 -1.1 -1.5

Text written in _markdown_

```{r}
# code written in R
```
```

16:20 Chunk 2 R Markdown

A YAML header surrounded by ———

YAML

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

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

CC BY-SA RStudio

output

The output: field sets the format of the final report

on back

output value	creates
html_document	html
pdf_document	pdf (requires Tex)
word_document	Microsoft Word (.docx)
odt_document	OpenDocument Text
rtf_document	Rich Text Format
md_document	Markdown
github_document	Github compatible markdown
ioslides_presentation	ioslides HTML slides
slidy_presentation	slidy HTML slides
beamer_presentation	Beamer pdf slides (requires Tex)

More at
rmarkdown.rstudio.com/formats.html

CC BY-SA RStudio

Parameters

Your Turn 2

Open **05-RMarkdown-Parameters.Rmd**.

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

What happens if you type in a different name?

CC BY-SA RStudio

Parameters

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

params list

elements and values

```
---
title: "Untitled"
output: html_document
params:
  filename: "data.csv"
  symbol: "FB"
---
```

colon

New line. Indented two spaces

CC BY-SA RStudio

Using Parameters

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

```
---
params:
  num: 42
---
The value of the
parameter is
`r params$num`, e.g.

```{r}
params$num
```
```

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

```
params$num
```

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

CC BY-SA RStudio

CountyStat PDF Template

Mon July 19, 2021 11:12 AM

Contents

Getting Started 2

Common Standards 2

Loading Data . . . 2

Database Connections . . . 2

Reading Flat Files . . . 2

Spatial Data . . . 2

Census Data . . . 3

Charts and Graphs . . . 4

Colors . . . 4

Tables . . . 5

Mapping . . . 6

ggplot2 map . . . 6

ggmap . . . 7

Landscaped Pages . . . 9

CountyStat Templates

github.com/Allegheny-CountyStats/CountyStat-Templates

rmarkdown

Getting Started

Common Standards

HTML Template

Analyst Name

Ran on: Fri December 17, 2021 10:34 AM

Getting Started

For many of the basics in RMarkdown the RStudio introduction is very worth while. Almost every R visualization output is compatible with it.

Table of Contents:

A floating table of contents for HTML documents is preferred for longer analysis. This allows the reader to go between sections easily without having to scroll back to the top of the document. The template also includes some slight changes to the markdown output format to make the most use of the reader's screen, but recognize that floating TOC is neither Mobile friendly (nobody should really be viewing on a phone anyway) nor very good on screens smaller than 1000 pixels in width.

A `toc_depth` of two is preferred, this means that any section that starts with either one or two `#`'s will appear. Other markdown headers will function, they are just omitted from the table of contents.

Common Standards

Loading Data

Database Connections

```
require(DBI)

wh_host <- Sys.getenv('WH_HOST')
wh_db <- Sys.getenv('WH_DB')
wh_user <- Sys.getenv('WH_USER')
wh_pass <- Sys.getenv('WH_PASS')

# Connect to the CountyStat DataWarehouse
wh_con <- dbConnect(odbc::odbc(), driver = "{ODBC Driver 17 for SQL Server}", server = wh_host, dat
abase = wh_db, UID = wh_user, pwd = wh_pass, Trusted_Connection= "yes")
```

Always keep your credentials for the DataWarehouse in an `.env` file. An example is in this repo, but in normal circumstances this file should be in the `.gitignore` file.

To use DBI check out their documentation, but for the most part you will be using either the `dbGetQuery()` or `dbReadTable()` function.

CountyStat Templates

github.com/Allegheny-CountyStats/CountyStat-Templates

rmarkdown

Report Reproducibly with

Slides CC BY-SARStudio

Thank you

CC BY-SA RStudio