

Week 13: Communicating Information

EMSE 6574 | John Paul Helveston | November 18, 2019

Quiz 5

Go here: <https://rstudio.cloud/spaces/39111/>

Link also on Slack [announcements](#) channel

Rules:

- Up to [20 minutes](#)
- You may use a single cheat sheet.
- You may *only* open the RStudio Cloud site to take the quiz.
- Type code to find answers to the questions.
- Type your final answer for each question as a comment.

Getting started

- 1) Download the `week13notes.zip` file for class today (link in `slack/classroom`).
- 2) Make sure you have the "tidyverse" installed and loaded:

```
library(tidyverse)
```

- 3) Load two data frames:

```
birds  
bears
```

Literate programming

Treat programs as a "literature" understandable to human beings

Core tenant: *reproducibility*

[Donald E. Knuth](#)



The horrors of a non-reproducible workflow

a reproducible workflow



RMarkdown to the rescue

01:12



RMarkdown to the rescue

Integrates two languages:

- Markdown: Documentantion language
- R: Programming language

Enables *reproducible* integration of text, code, output



Elements of an RMarkdown file ([.Rmd](#))

1) YAML metadata

```
---
```

```
title: "This is a demo"
author: "John Helveston"
date: "11/18/2019"
output: html_document
---
```

2) Markdown text

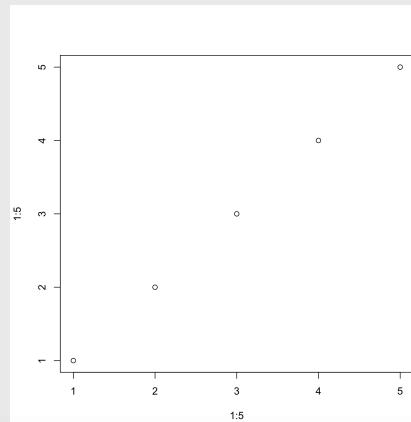
```
# Section 1
```

```
This is a sentence...
```

3) R Code

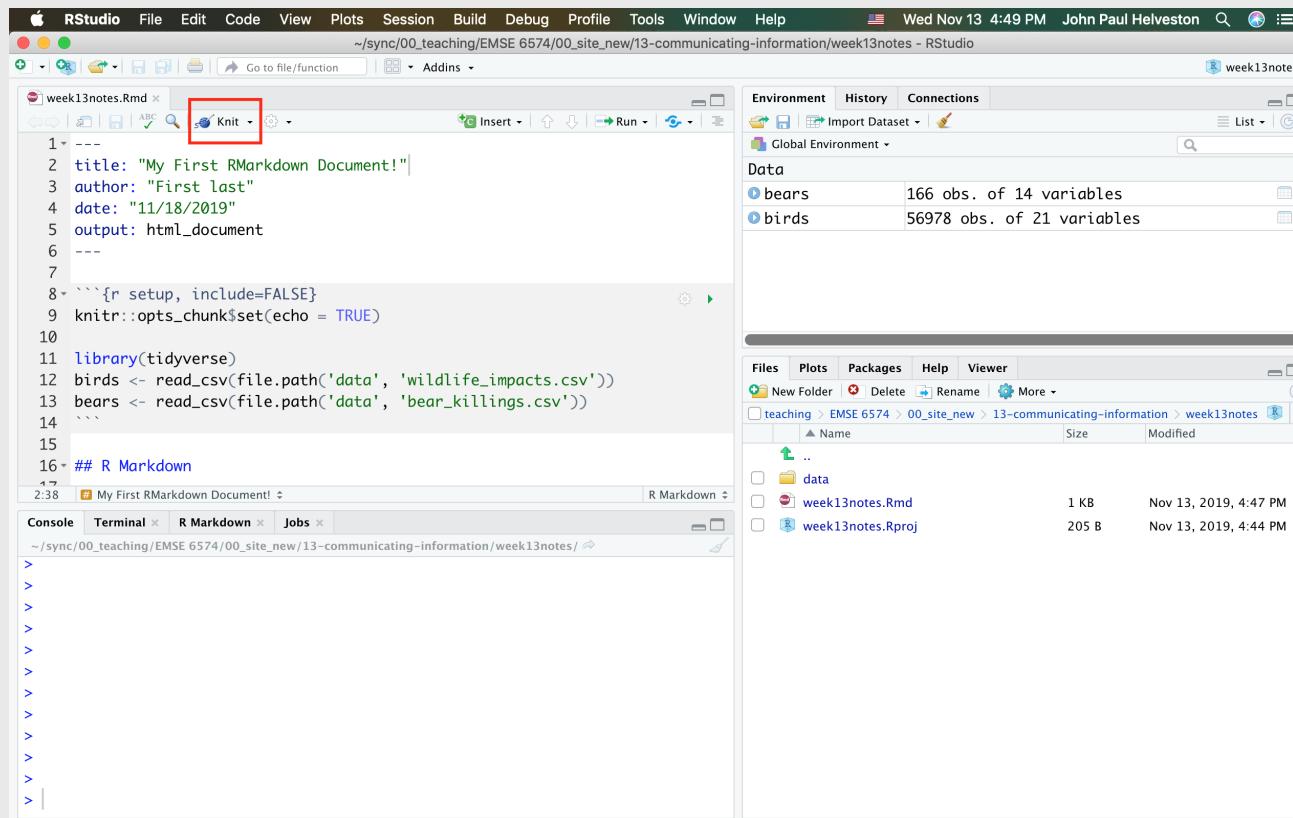
```
```{r}
plot(x = 1:5, y = 1:5)
````
```

4) Code output



Build your first RMarkdown document

Open the `week13notes.Rmd` file, and click the "knit" button



The YAML header

YAML stands for:

- Y: Yet
- A: Another
- M: Markdown
- L: Language

At the top of the file:

```
---  
title: "This is a demo"  
author: "John Helveston"  
date: "11/18/2019"  
output: html_document  
---
```

Markdown

Basic formatting

| Type this... | ...to get this |
|------------------------|--------------------------------|
| normal text | normal text |
| *italic text* | <i>italic text</i> |
| **bold text** | bold text |
| ***bold italic text*** | <i>bold italic text</i> |
| ~~strikethrough~~ | strikethrough |
| `code text` | code text |

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

Markdown

Bullet list:

- first item
- second item
- third item

- first item
- second item
- third item

Numbered list:

1. first item
2. second item
3. third item

1. first item
2. second item
3. third item

Markdown

Simple **url link** to another site:

[Download R](http://www.r-project.org/)

Download R

Markdown

Basic tables:

| Table Header | Second Header |
|--------------|---------------|
| Cell 1, 1 | Cell 2, 1 |
| Cell 1, 2 | Cell 2, 2 |

| Table Header | Second Header |
|--------------|---------------|
| Cell 1, 1 | Cell 2, 1 |
| Cell 1, 2 | Cell 2, 2 |

Markdown

Useful tools:

- Quick markdown reference guide: <https://commonmark.org/help/>
- Quick demo guide: <https://markdown-it.github.io/>.
- Online table converter: <http://www.tablesgenerator.com>

Practice: Markdown basics

- Open `week13notes.Rmd`
 - At the bottom, write markdown text to create a new section with the following content (with the correct formatting - use `#` for the header):
-

Markdown practice

This is a markdown sentence. Some words are in *italics*, others in **bold**, and some are **both!** Finally, some are written in `code formatting`. Here is a markdown table:

| Animal | Sound |
|--------|-------|
| Dog | Woof |
| Cat | Meow |
| Cow | Moo |

R Code

Two types:

- Inline code
- Code chunks

Inline R code

To embed R code directly in a markdown sentence, use

```
`r <insert code>`
```

For example, this:

```
The value of pi is `r pi`
```

Produces this:

The value of pi is 3.141593

R Code chunks

Basic syntax for code chunk:

```
```{r}  
<insert code>
````
```

For example, this:

```
```{r}  
cat('hello world!')
````
```

Produces this:

```
## hello world!
```

Outputs

Control what chunks output using options inside `{r}`:

Example: `{r, echo=FALSE, message=FALSE}`

| option | default | effect |
|------------|----------|---|
| eval | TRUE | Whether to evaluate the code and include its results |
| echo | TRUE | Whether to display code along with its results |
| warning | TRUE | Whether to display warnings |
| error | FALSE | Whether to display errors |
| message | TRUE | Whether to display messages |
| tidy | FALSE | Whether to reformat code in a tidy way when displaying it |
| results | "markup" | "markup", "asis", "hold", or "hide" |
| cache | FALSE | Whether to cache results for future renders |
| comment | "##" | Comment character to preface results with |
| fig.width | 7 | Width in inches for plots created in chunk |
| fig.height | 7 | Height in inches for plots created in chunk |

Outputs

By default, code chunks print **code + output**:

```
cat('hello world!')
```

```
## hello world!
```

{r, eval=FALSE}

Prints only **code** (doesn't run the code)

```
cat('hello world!')
```

{r, echo=FALSE}

Prints only **output**

```
## hello world!
```

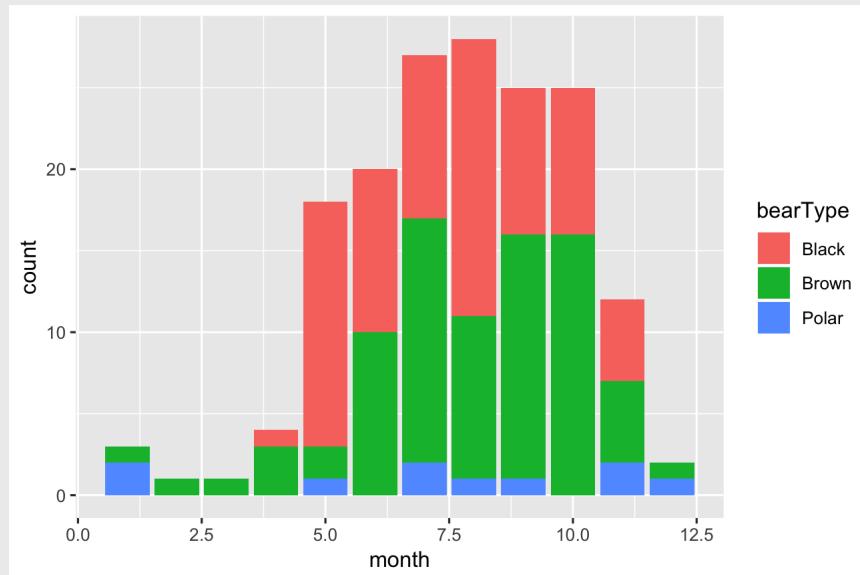
{r, include=FALSE}

Runs, but doesn't print anything

Outputs

Adjust the dimensions of plots using `fig.height` & `fig.width`
`{r, fig.height=4, fig.width=6}`

```
ggplot(data = bears) +  
  geom_bar(aes(x = month, fill = bearType))
```



Practice: Building a `.Rmd` from scratch!

1. Create a new R Markdown file (`.Rmd`) in RStudio.
2. Insert a YAML Header with the title "Fuel Economy Analysis". Add your name and today's date.
3. Create a chunk and load the `tidyverse` library in that chunk.
4. Create another chunk; in that chunk, use the `mpg` data frame to make and display a scatter plot of the highway fuel economy versus the engine displacement.
5. Click the `knit` button and preview your document.
6. In the chunk where you loaded the `tidyverse` library, change the chunk settings to hide the message that appeared after loading the library (HINT: Use `message`).
7. Change the plot chunk settings to hide the code that made the plot - only show the plot (HINT: Use `echo` setting).
8. Set the `fig.width` and `fig.height` of your plot to `5`.
9. Click the `knit` button and preview your document.

5 minute break!

Stand up

Move around

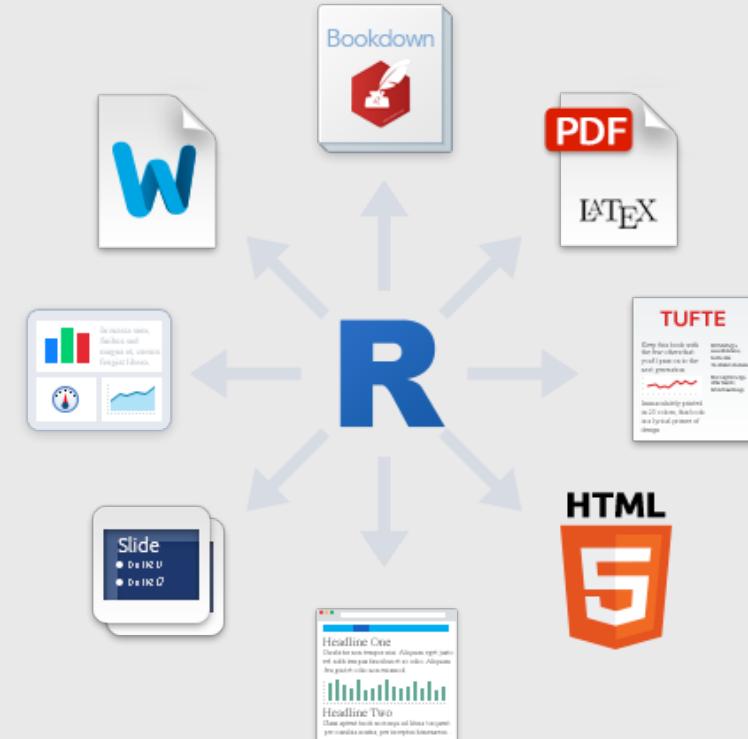
Stretch!

Knitting to other output formats

Change the YAML [output](#):

```
---
```

```
title: "This is a demo"
author: "John Helveston"
date: "11/18/2019"
output: pdf_document
---
```



Convert a data frame to a table

Use the `kable()` function:

```
bears %>%  
  count(bearType, wildOrCaptive) %>%  
  kable()
```

| bearType | wildOrCaptive | n |
|----------|---------------|----|
| Black | Captive | 16 |
| Black | Wild | 60 |
| Brown | Captive | 8 |
| Brown | Wild | 72 |
| Polar | Captive | 4 |
| Polar | Wild | 6 |

Inserting a Python code chunk

Change `{r}` to `{python}` in the code chunk.

```
'In Python, you can concatenate strings' + ' like this!'
```

```
## 'In Python, you can concatenate strings like this!'
```

Publishing to the web

You can publish your report to the web for free using RPubs: <http://rpubs.com/>

1. Click the `knit` button to compile the `.html` file
2. Click the `publish` button in the top-right

Practice analysis 1: Birds & Bears

- 1) Create a new R Markdown file ([.Rmd](#)) in RStudio - title it "*Birds and Bears Analysis*"
- 2) Create a code chunk to load the `tidyverse` library and the `birds.csv` and `bears.csv` files.
- 3) Answer each of the following questions with text and code chunks with supportive plots of the data:
 - Which months have the highest and lowest number of bird impacts with aircraft?
 - Does the annual number of bird impacts appear to be changing over time?
 - At what heights do most bird impacts occur?
 - Which months have the highest frequency of bear killings?
 - How do the the number of bear attacks on men vs women compare, and does there appear to be any trend over time?
- 4) Publish your results to the web using RPubs.

Practice analysis 2: College Majors

- 1) Create a new R Markdown file ([.Rmd](#)) in RStudio - title it "*College Majors Analysis*"
- 2) Create a code chunk to load the `tidyverse` library and the `recent_grads.csv` file.
- 3) Answer each of the following questions with text and code chunks with supportive plots of the data:
 - What categories of majors make more money than others?
 - What categories of majors have the best employment rate?
 - What are the highest earning majors?
 - Within the engineering majors, which ones have a better gender balance?
- 4) Publish your results to the web using RPubs.