# BIOST 561: R Markdown Intro

David Whitney

November 2, 2017

# R Markdown: The Basics

The following information is **readily** available if you use Rstudio (e.g. by creating a blank *.Rmd file):

- 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.

# Supported Formats

R Markdown can generate output in several standard formats.

- **HTML** document, with presentations via **ioslides** or **Slidy**
- **PDF** document, with presentations via **beamer**
- **MS Word** document

**Note**: we will only scratch the surface today, check out other options at `http://rmarkdown.rstudio.com/lesson-9.html`.

# Minimalist Document

A RMarkdown file has extension `*.Rmd` and consists of a YAML metadata header used to specify rendering options and a body of formatted text and code chunks. A simple example is:

```
---
output: html_document
---

That's all, folks!
```

# Rendering Markdown

Though the RStudio GUI is well suited for working with Markdown, you may knit with more refined control using the `render` function:

```
rmarkdown::render("example.Rmd", output_format =
"html_document")
```

# Basic Markdown Syntax

Regardless of your chosen output format, some basic syntax will be useful:

- Section headers
- Text emphasis
- Lists
- `R` code

# Section Headers

To set up different sized header text in your document, use # for Header 1, ## for Header 2, and ### for Header 3.

- In a presentation, this creates a new slide.

# Text emphasis

- *Italicize* text via `*Italicize*` or `_Italicize_`.
- **Bold** text via `**Bold**` or `__Bold__`.

# Unordered Lists

This code

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

Renders these bullets (sub-lists need 1 tab or 4 spaces!)

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

# Ordered Lists

This code

```
1. Item 1
2. Item 2
    + Item 2a
    + Item 2b
```

Renders this list (be advised - the bullets may not look great in all templates)

1. Item 1
2. Item 2

    - Item 2a
    - Item 2b

# Inline R Code

- To use R within a line, use the syntax `` `r foo` ``.
- e.g. `` `r round(pi, 5)` `` renders as 3.14159.
- This can be useful to refer to estimates, confidence intervals, p-values, etc. in the body of an article/homework without worrying about copy errors.

# R Code Chunks

- R code chunks let you run/render code and results similar to Sweave or knitr.
- To start a code chunk, use the syntax “‘{r chunkName, options}.
- To end the chunk, type “‘.

# Example Chunk Output 1

With no options specified, a typical code chunk might look like:

````
```{r pressure}
summary(pressure)
```
````

```
##    temperature      pressure
##  Min.   :  0   Min.   :  0.0002
##  1st Qu.: 90   1st Qu.:  0.1800
##  Median :180   Median :  8.8000
##  Mean   :180   Mean   :124.3367
##  3rd Qu.:270   3rd Qu.:126.5000
##  Max.   :360   Max.   :806.0000
```

# Example Chunk Output 2

- Want to display the output of a code chunk and the underlying `R` code?
- Specify the `echo = TRUE` option.

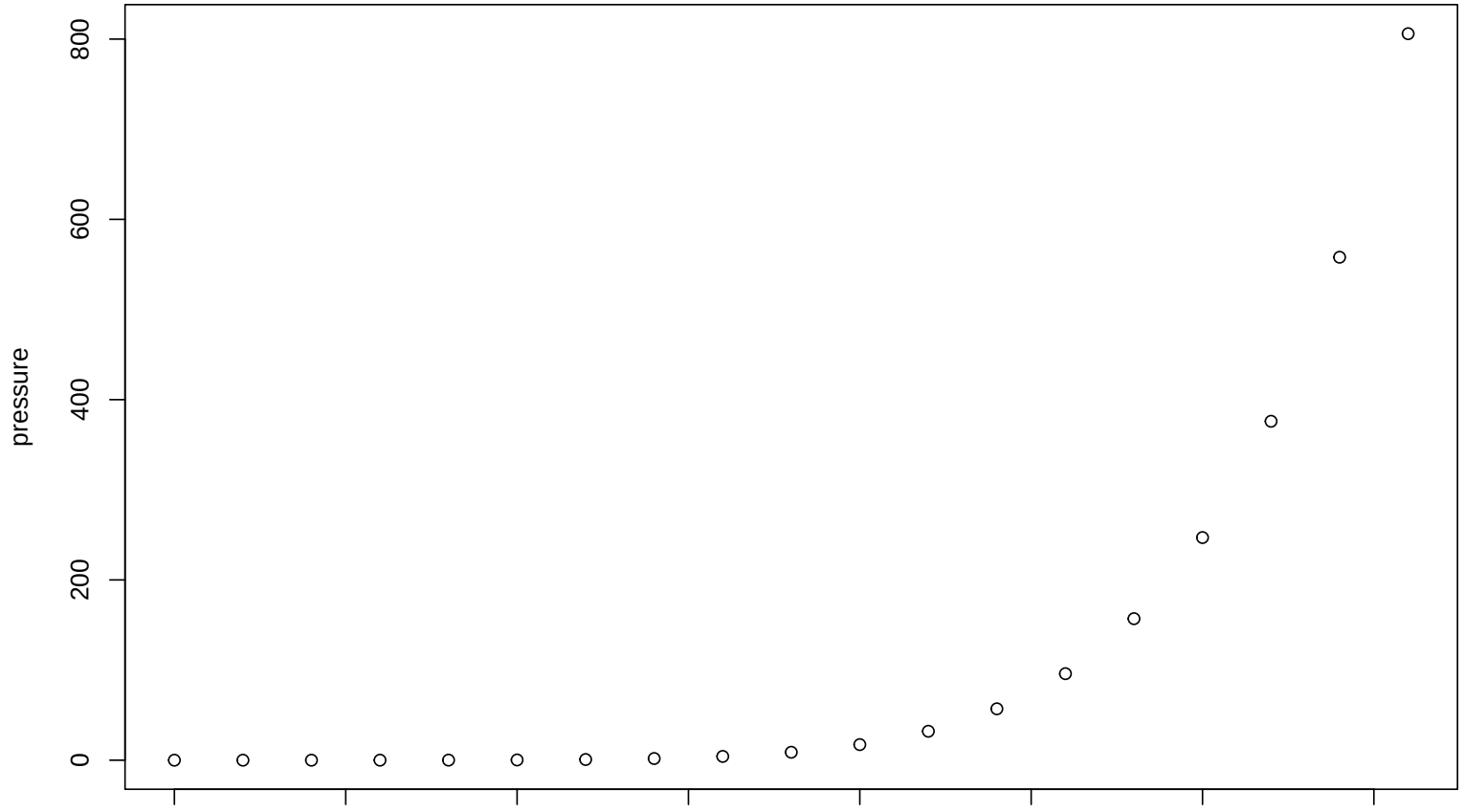
```
summary(pressure)
```

```
##   temperature      pressure
##  Min.   :  0   Min.   :  0.0002
##  1st Qu.: 90   1st Qu.:  0.1800
##  Median :180   Median :  8.8000
##  Mean   :180   Mean   :124.3367
##  3rd Qu.:270   3rd Qu.:126.5000
##  Max.   :360   Max.   :806.0000
```

# Example Chunk Output 3

- R code chunks can also be used to render plots.

```
plot(pressure)
```



# Example Chunk Output 4

- Want to display `R` code without evaluating it?
- Specify `eval = FALSE, echo = TRUE` in the chunk options:

```
summary(pressure)
```

# Example Chunk Output 5

- For chunks which contain intensive/long computations, you may want to cache the results. (Note: this may not be a *great* of this!)

````
```{r pressure, cache = TRUE}
summary(pressure)
```
````

# More on Chunk Options

For additional information, check out
`https://yihui.name/knitr/options/`

- Note that you can use `knitr::opts_chunk$set(echo = TRUE)` to change the default chunk options.

# Mathematical Symbols/Equations in Markdown

- LaTeX's inline (e.g. `$foo$`) and display (e.g. `$$foo$$`) math modes are supported in Markdown for output to HTML, Word or PDF.

# Tables and Figures in Markdown

- The default `R` output suffices for teaching, but requires cleaning up for assignments, theses, or papers.
- This is when knitting to *.pdf becomes an attractive option.

# Knitting


Figure 1: Image courtesy of http://rmarkdown.rstudio.com/lesson-2.html

# LaTeX and Markdown

- Rendering Markdown as a pdf requires a LaTeX installation.
- You will additionally need to install Pandoc from `http://pandoc.org/`
- With LaTeX, many customizations are possible.

# LaTeX Customization, 1

- You can include additional LaTeX commands and content.
- Use the `includes` option as follows to add your favorite style files for the preamble, title/abstract, bibliography, etc. . .

```
---
title: 'A More Organized Person's Document'
output:
  beamer_presentation:
    includes:
      in_header: header.tex
      before_body: doc_prefix.tex
      after_body: doc_suffix.tex
---
```

# LaTeX Customization, 2

- If you prefer a self-contained document, you may opt for the `header-includes` option over the modular approach:

```
---
title: 'BIOST 561: R Markdown Intro'
author: "David Whitney"
date: "November 2, 2017"
header-includes:
   - \usepackage{graphicx}
output:
  beamer_presentation:
    theme: "Frankfurt"
---
```

# Note: LaTeXin Text

- In Markdown, "`\LaTeX rocks`" renders as "LaTeXrocks" (no space!).
- Use "`\LaTeX\ rocks`" to render "LaTeX rocks", instead.
- This can be especially important when using new commands.

# Tables

Customization for tables can be carried out using functions in the `knitr` or `xtable` packages from `R`.

````
```{r table}
knitr::kable(summary(pressure))
```
````

````
```{r table, results='asis'}
xtable::xtable(summary(pressure))
```
````

# Example Output: `knitr`

| temperature | pressure |
|---|---|
| Min. : 0 | Min. : 0.0002 |
| 1st Qu.: 90 | 1st Qu.: 0.1800 |
| Median :180 | Median : 8.8000 |
| Mean :180 | Mean :124.3367 |
| 3rd Qu.:270 | 3rd Qu.:126.5000 |
| Max. :360 | Max. :806.0000 |

# Example Output: xtable

```
library(xtable)
print(xtable(summary(pressure)),
      include.rownames = FALSE, comment = FALSE)
```

| temperature | pressure |
|---|---|
| Min. : 0 | Min. : 0.0002 |
| 1st Qu.: 90 | 1st Qu.: 0.1800 |
| Median :180 | Median : 8.8000 |
| Mean :180 | Mean :124.3367 |
| 3rd Qu.:270 | 3rd Qu.:126.5000 |
| Max. :360 | Max. :806.0000 |

# Resources for customizing tables

- `xtable`: `https://cran.r-project.org/web/packages/xtable/vignettes/xtableGallery.pdf`
- `kable`: `https://cran.r-project.org/web/packages/kableExtra/vignettes/awesome_table_in_html.html`

# Customizing Figures: Captions

The `fig.cap` option allows you to specify the caption for the figure generated by a given chunk:

````
```{r caption, fig.cap="I am the caption"}
plot(pressure)
```
````

# Caption Example

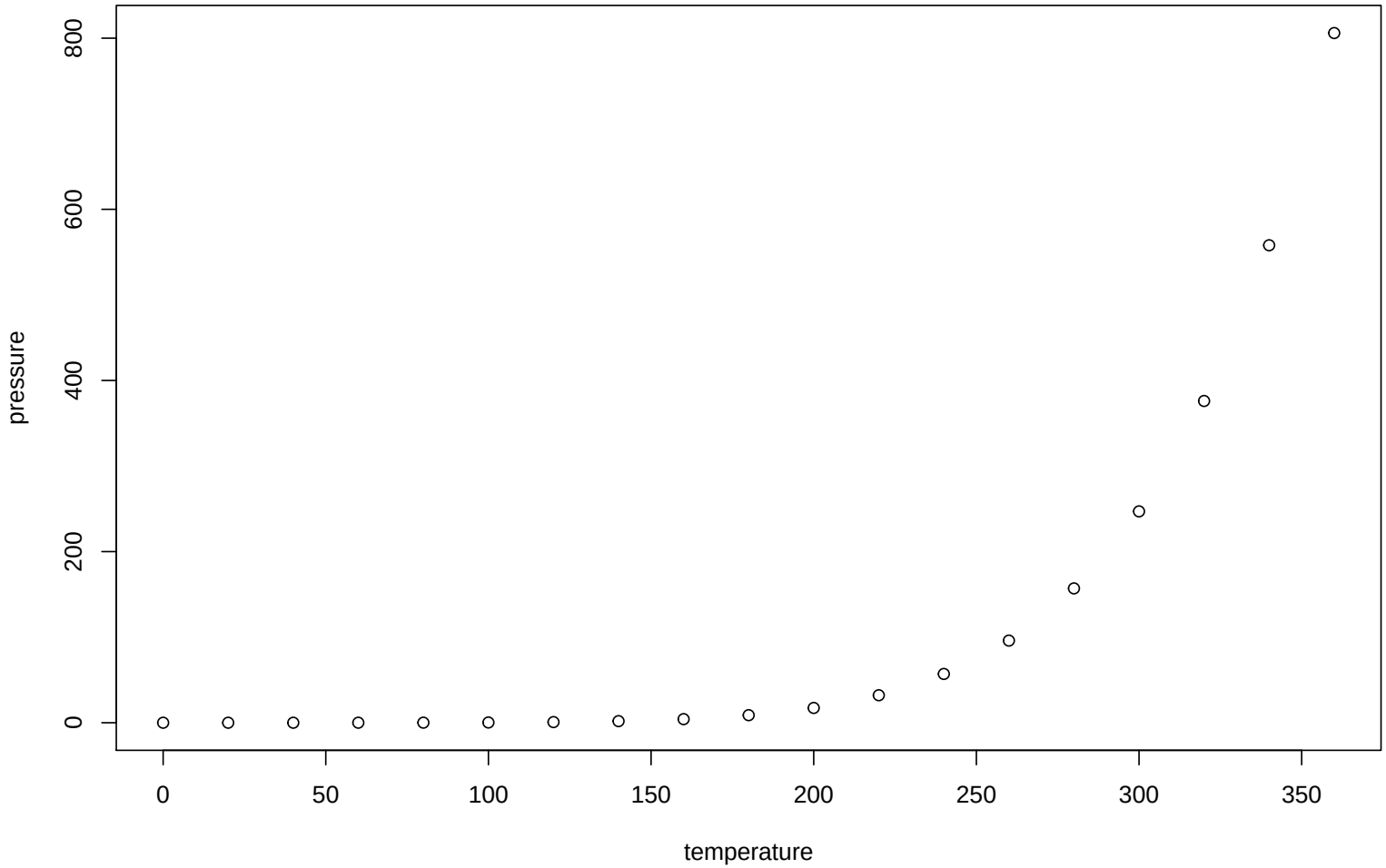


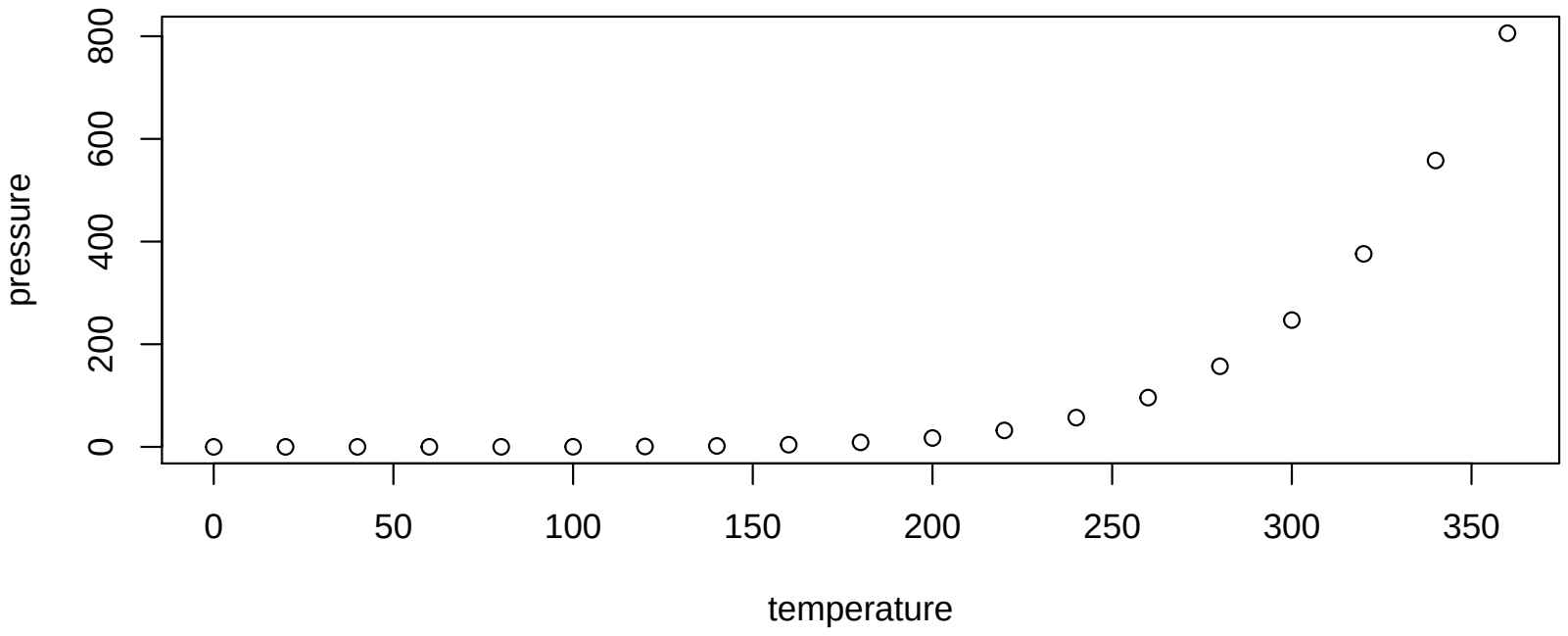

Figure 2: I am the walrus

# Customizing Figures: Size

The `fig.height` and `fig.width` options let you specify the dimensions of your plots:

````
```{r caption, fig.height = 4, fig.width = 8}
plot(pressure)
```
````

# Figure Size Example



# Making it on your own

Altering the default Rmarkdown file each time you write a homework, report, or article would be a pain.

- Fortunately, you don't have to!

# Templates

You can create your own templates which set-up packages, fonts, default chunk options, etc.

- `http://rmarkdown.rstudio.com/developer_document_templates.html`
- Some packages (e.g `rticles`) provide templates that meet journal requirements or provide other.

# Parameters

You may also set parameters in your document's YAML header

```
---
output: html_document
params:
  date: "2017-11-02"
---
```

or pass new values with the `render` function.

- This creates a read-only list `params` containing the values declared.
- e.g. `params$date` returns `2017-11-02`.

# Child code chunks

To manage large documents, it can be useful to write sections/chapters in separate *.Rmd files and include them as children in the main (parent) file:

```
```{r ch1, child = chap1.Rmd}
```
```

- Examples can be found at `https://yihui.name/knitr/demo/child/`

# Congratulations!

You now have all the tools to start knitting your own documents using RMarkdown.

# Note

We did not talk much about HTML (or Word, I guess). You can create interactive HTML documents from Rmarkdown with packages such as:

- `shiny`
- `flexdashboard`

# Additional Resources

- RStudio: `http://rmarkdown.rstudio.com/`
- `xtable` and `knitr` documentation
- Pandoc: `http://pandoc.org/`
- Google

# Wrapping up

Any Questions?