

Dynamic Reporting

An Interactive Workshop for Statistical Reproducible Research

Kirsten Bell Burdett, MB

Department of Preventive Medicine
Feinberg School of Medicine
Northwestern University

February 18, 2020

Intro

Required Materials

Materials for this presentation can be found on [GitHub](https://github.com/kburdett5/Dynamic_Reporting_NUpres) :
https://github.com/kburdett5/Dynamic_Reporting_NUpres

- R
 - RStudio: <https://rstudio.com/products/rstudio/download/>
 - R version 3.5.2 or later
 - R packages: knitr, survminer, kableExtra, tableone, plotly, broom, tidyverse, survival (`install.packages("knitr")`)
- LaTeX
 - <https://www.latex-project.org/get/>

What is R Markdown

Popular data science tool for efficient and dynamic reporting. It is used to record your reproducible research (save and execute code) in order to create beautiful and easily sharable reports.

- Flexible
 - Create dynamic output and inline code
- Reproducible Research
 - Code and report combined
- Fun endless possibilities!

No Copy-Paste Results

Imagine you finish a project and generate the results.

THEN... the data changes, or they want to use $\alpha = 0.1$ reaction if results were copy-pasted.



If using R Markdown, simply render the file and VOILA you have your beautiful report!

How it works

- Markdown is a formatting syntax
- Dynamic reporting allows you to export finished report in several formats (Word, html, PDF, LaTeX Beamer slides, and more).



When you run (render), feeds to knitr which executes code chunks. Then pandoc processes it and creates the finished format.

<https://rmarkdown.rstudio.com/lesson-2.html>

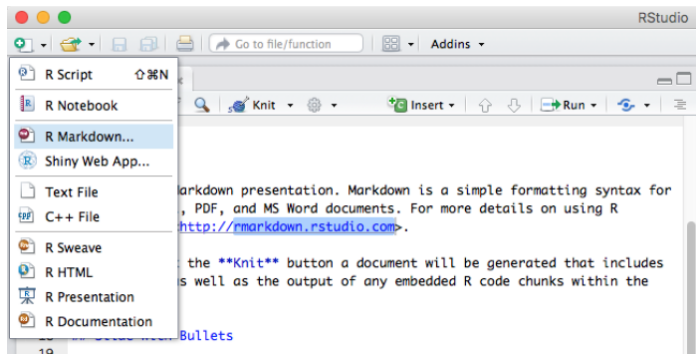
Getting Started

Getting Started

- R package “knitr” (`install.packages(“knitr”)`)
 - .Rmd (R + Markdown)
 - .Rnw (R + LaTeX)
 - Allows use of various languages

We will be using R Markdown (.Rmd). If you are more comfortable with LaTeX, you may want to try using R Sweave (.Rnw).

R Markdown File (File -> New File -> R Markdown)





<https://rmarkdown.rstudio.com>


<https://rstudio.com/wp-content/uploads/2015/03/rmarkdown-reference.pdf>


Select Report Format

New R Markdown

 Document

 Presentation

 Shiny

 From Template

Title:

Author:

Default Output Format:

☐ HTML
Recommended format for authoring (you can switch to PDF or Word output anytime).

☒ PDF
PDF output requires TeX (MiKTeX on Windows, MacTeX 2013+ on OS X, TeX Live 2013+ on Linux).

☐ Word
Previewing Word documents requires an installation of MS Word (or Libre/Open Office on Linux).

OK Cancel

YAML Header

Basic Header:

```
---  
title: "Untitled"  
author: "Kirsten Burdett"  
date: "1/27/2020"  
output: pdf_document  
---
```

Complex Header:

```
---  
title: "Example Report"  
author: "Kirsten Burdett"  
institute: "Northwestern University: Feinberg School of Medicine"  
date: "\r format(Sys.Date()), '%d %b %Y')"  
geometry: margin=2cm  
output:  
  pdf_document:  
    toc: yes  
    toc_depth: 2  
header-includes: \usepackage{float, graphicx, subfig, amsmath,  
amssymb, booktabs, tikz, caption}  
\usetikzlibrary{shapes,arrows,positioning,backgrounds}  
---
```

- Applied to entire document
- Output changes report format

For more details: [R Markdown Cheatsheet](#)

Embed Code

Syntax	Becomes
<p>Make a code chunk with three back ticks followed by an <code>r</code> in braces. End the chunk with three back ticks:</p> <pre>```{r} paste("Hello", "World!") ```</pre>	<p>Make a code chunk with three back ticks followed by an <code>r</code> in braces. End the chunk with three back ticks:</p> <pre>paste("Hello", "World!") ## [1] "Hello World!"</pre>
<p>Place code inline with a single back ticks. The first back tick must be followed by an <code>R</code>, like this <code>`r paste("Hello", "World!")`</code>.</p>	<p>Place code inline with a single back ticks. The first back tick must be followed by an <code>R</code>, like this <code>Hello World!</code>.</p>
<p>Add chunk options within braces. For example, <code>`echo=FALSE`</code> will prevent source code from being displayed:</p> <pre>```{r eval=TRUE, echo=FALSE} paste("Hello", "World!") ```</pre>	<p>Add chunk options within braces. For example, <code>echo=FALSE</code> will prevent source code from being displayed:</p> <pre>## [1] "Hello World!"</pre>

Learn more about chunk options at <http://yihui.name/knitr/options>

Formatting

Some helpful Markdown syntax. You can also utilize LaTeX or HTML code.

- ****BOLD**** —> **BOLD** text
- **italics** —> *italics* text
- $\frac{\alpha}{5}$ —> $\frac{\alpha}{5}$
- Graphics
 - ````{r, fig.width=5, fig.height=4}`

For more details: [R Markdown Cheatsheet](#)

Instructions Inside Chunks

- **echo**: Display code. `echo = FALSE` - hides code and displays results
- **include**: `include = FALSE` - Run code without displaying anything in document
- **message**: Display messages. `message = FALSE` - suppresses messages from being displayed in document. Same idea for `error = FALSE` and `warning = FALSE`
- **results**: Set to 'asis' to let text output be written as is. Write raw Markdown text from R code. Ex: `cat('**Markdown** is cool. \n')`
- **tidy**: Tidy code for display

For more details: [R Markdown Cheatsheet](https://bookdown.org/yihui/rmarkdown/r-code.html)

<https://bookdown.org/yihui/rmarkdown/r-code.html>

Loop through Slides

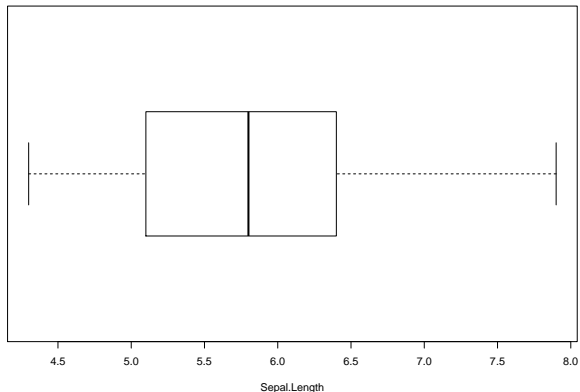
Loop Slides

- Useful when repetition is needed
- Create Box Plots for continuous variables in iris data
 - `results='asis'`: Write raw Markdown text from R code
 - Slide title: `cat("##Box Plot for", i, "\n")`
 - Text on slides: `cat("Assess the distribution of", i, "\n")`
 - Box Plots: `boxplot(iris[,i], horizontal = TRUE, xlab = i)`

```
```${r, results='asis', out.width= "70%"}
nameval <- names(iris)[!names(iris) == "Species"]
for(i in nameval){
 cat("\n")
 cat("##Box Plot for", i, "\n")
 cat("Assess the distribution of", i, "\n")
 boxplot(iris[,i], horizontal = TRUE, xlab = i)
 cat("\n")
}
```
```

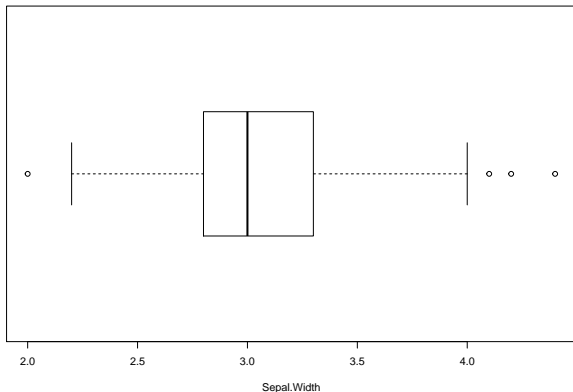

Box Plot for Sepal.Length

Assess the distribution of Sepal.Length



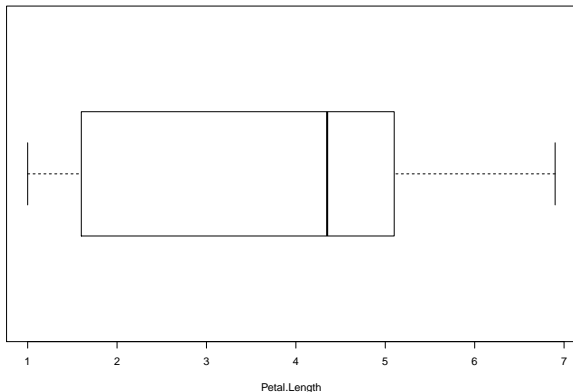
Box Plot for Sepal.Width

Assess the distribution of Sepal.Width



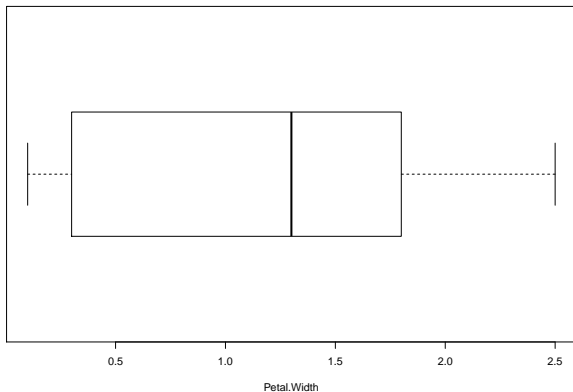
Box Plot for Petal.Length

Assess the distribution of Petal.Length



Box Plot for Petal.Width

Assess the distribution of Petal.Width

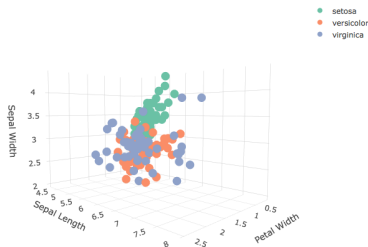


Hands-on

Highlights

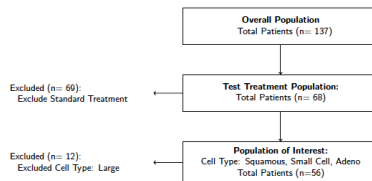
3D Plots:

3D Scatter Plot by Species: Iris Data



CONSORT:

CONSORT Flow Diagram



HTML Example: 3D Plots

“HTML_example.html”

HTML Example: 3D Plots

Kirsten Bell Burdett

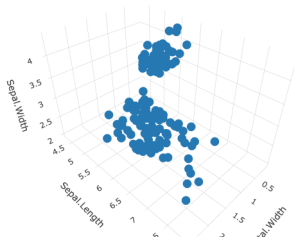
February 18, 2020

Load Libraries

```
library(plotly) # for 3D plotting
```

3D Plot

```
plot_ly(iris, x=~Petal.Width, y=~Sepal.Length, z=~Sepal.Width) %>%  
  add_markers()
```



Word Example: “Word_example.docx”

Megan McCabe, MS
 Department of Preventive Medicine
 Feinberg School of Medicine
 Northwestern University

Word Example

Megan McCabe|

14 Feb 2020

Linear Regression Model

```
fit <- lm(dist ~ speed, data = cars)
fit_res <- tidy(fit); fit_res
```

```
## # A tibble: 2 x 5
##   term      estimate std.error statistic  p.value
##   <chr>      <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept) -17.6      6.76    -2.60  1.23e- 2
## 2 speed        3.93     0.416     9.46  1.49e-12
```

Easily pull out results with tidy() function. Then we can use inline code like `format.pval(fit_res$p.value[fit_res$term == "speed"], digit = 4)` to show the p-value 1.49e-12 corresponding to speed.

Baseline Patient Demographic and Clinical Characteristics

Demographics by Prior Therapy

| | level | No | Yes |
|---------------|----------|-----------|-----------|
| n | | 97 | 40 |
| Cell Type (%) | squamous | 21 (21.6) | 14 (35.0) |

PDF Example: Interactive

“PDF_example.pdf”

Example Report

Kirsten Bell Burdett

14 Feb 2020

Contents

| | |
|------------------------------------------------------------------|----------|
| Background | 2 |
| Objectives | 2 |
| CONSORT Flow Diagram | 2 |
| Baseline Patient Demographic and Clinical Characteristics | 3 |
| Primary Objective | 4 |
| Adverse Events | 4 |
| Survival Outcomes | 4 |
| Supplementary | 5 |

Data:

```
readfile <- file.path(dir, "FakeExampleAEs.csv")  
fakeAEdat <- read_csv(readfile)
```

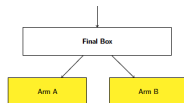
```
/FakeExampleAEs.csv  
cb2247c25d3ed473e0e531a8f97a04a7
```

Veteran Data:

We will be using the veteran data from the survival package. The original data has 137 subjects, but we will be subsetting to only those subjects that have test treatment and cell type of squamous, smallcell, or adeno which will result in 56 subjects.

Exercises

- ❶ Make “statistical analysis plan (SAP)” bold on page 2
- ❷ Add section “Secondary Objective” before “Supplementary” section with description underneath similar to primary objective
- ❸ Add inline code to replace the hardcoded 56 at the end of page 1 (“will result in 56 subjects”)
- ❹ Show the code that generates the Kaplan-Meier Plots
- ❺ CONSORT
 - Add bordered box underneath “Population of Interest” titled “Final Box”
 - Add an arrow from “Population of Interest” to the box you just created
 - Extra: Create “Arm A” and “Arm B” boxes below “Final Box” in yellow with arrows



Exercise 1

- 1 Make “statistical analysis plan (SAP)” bold on page 2

Exercise 1: Answer

- 1 Make “statistical analysis plan (SAP)” bold on page 2

```
**statistical analysis plan (SAP)**
```

Exercise 2

- 2 Add section “Secondary Objective” before “Supplementary” section with description underneath similar to primary objective

Exercise 2: Answer

- 2 Add section “Secondary Objective” before “Supplementary” section with description underneath similar to primary objective

```
#Secondary Objective
```

```
\underline{Secondary Objective:} Define HERE
```

Exercise 3

- 3 Add inline code to replace the hardcoded 56 at the end of page 1 (“will result in 56 subjects”)

Exercise 3: Answer

- 3 Add inline code to replace the hardcoded 56 at the end of page 1 (“will result in 56 subjects”)

will result in `'r nrow(analysisdat)'` subjects

Exercise 4

- 4 Show the code that generates the Kaplan-Meier Plots

Exercise 4: Answer

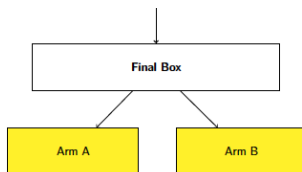
- 4 Show the code that generates the Kaplan-Meier Plots

Remove `echo=FALSE`

Exercise 5

5 Edit CONSORT

- Add bordered box underneath “Population of Interest” titled “Final Box”
- Add an arrow from “Population of Interest” to the box you just created
- Extra: Create “Arm A” and “Arm B” boxes below “Final Box” in yellow with arrows



Exercise 5: Answer

- Add bordered box underneath “Population of Interest” titled “Final Box”

```
\node [block_full, below=of finalpop] (finalbox) {  
  \textbf{Final Box}};
```

- Add an arrow from “Population of Interest” to the box you just created

```
\draw[->, line width=0.5pt] (finalpop) -- (finalbox);
```

Exercise 5: Extra Answer

5 Edit CONSORT

- Extra: Create “Arm A” and “Arm B” boxes below “Final Box” in yellow with arrows

```
\tikzset{
  block_full/.style={rectangle,draw=black,thick,fill=white,
    text width=20em,text centered,minimum height=4em,inner sep=6pt},
  block_noborder/.style={rectangle,draw=none,thick,fill=none,
    text width=15em,text ragged,minimum height=1em,inner sep=6pt},
  block_arm/.style={rectangle,draw=black,thick,fill=yellow,
    text width=10em,text centered,minimum height=4em,inner sep=6pt},
  line/.style={draw,very thin,-latex',shorten >=0pt}
}
```

```
\node [block_arm, below=of finalbox, xshift = -2.5cm] (arma) {Arm A};
\node [block_arm, right=of arma] (armb) {Arm B};

%% Arrows %%
\draw[>, line width=0.5pt] (start) -- (testpop);
\draw[>, line width=0.5pt] (testpop) -- (finalpop);
\draw[>, line width=0.5pt] (testpop) -- (lefttest);
\draw[>, line width=0.5pt] (finalpop) -- (excluded_large);
\draw[>, line width=0.5pt] (finalpop) -- (finalbox);
\draw[>, line width=0.5pt] (finalbox) -- (arma);
\draw[>, line width=0.5pt] (finalbox) -- (armb);

\end{tikzpicture}
```

Thank You!

Thank You!