Dynamic documents and reproducible research with R Markdown

Sarah E. Bowden, Ph.D.
bowdens@caryinstitute.org

July 28, 2016

What is reproducible research?

All analyses and graphics from a project
can be reproduced by **anyone**

To test your projects for reproducibility (in R):

1. Clear your history/workspace.
2. Highlight all of your code.
3. Hit "Run."

Why should we strive for reproducible research?

- Transparency in our research

Why should we strive for reproducible research?

- Transparency in our research
- Promotes collaboration

Why should we strive for reproducible research?

- Transparency in our research
- Promotes collaboration
- Others can easily build off of previous research

Why should we strive for reproducible research?

- Transparency in our research
- Promotes collaboration
- Others can easily build off of previous research
- Helps out future you!

What is R Markdown?

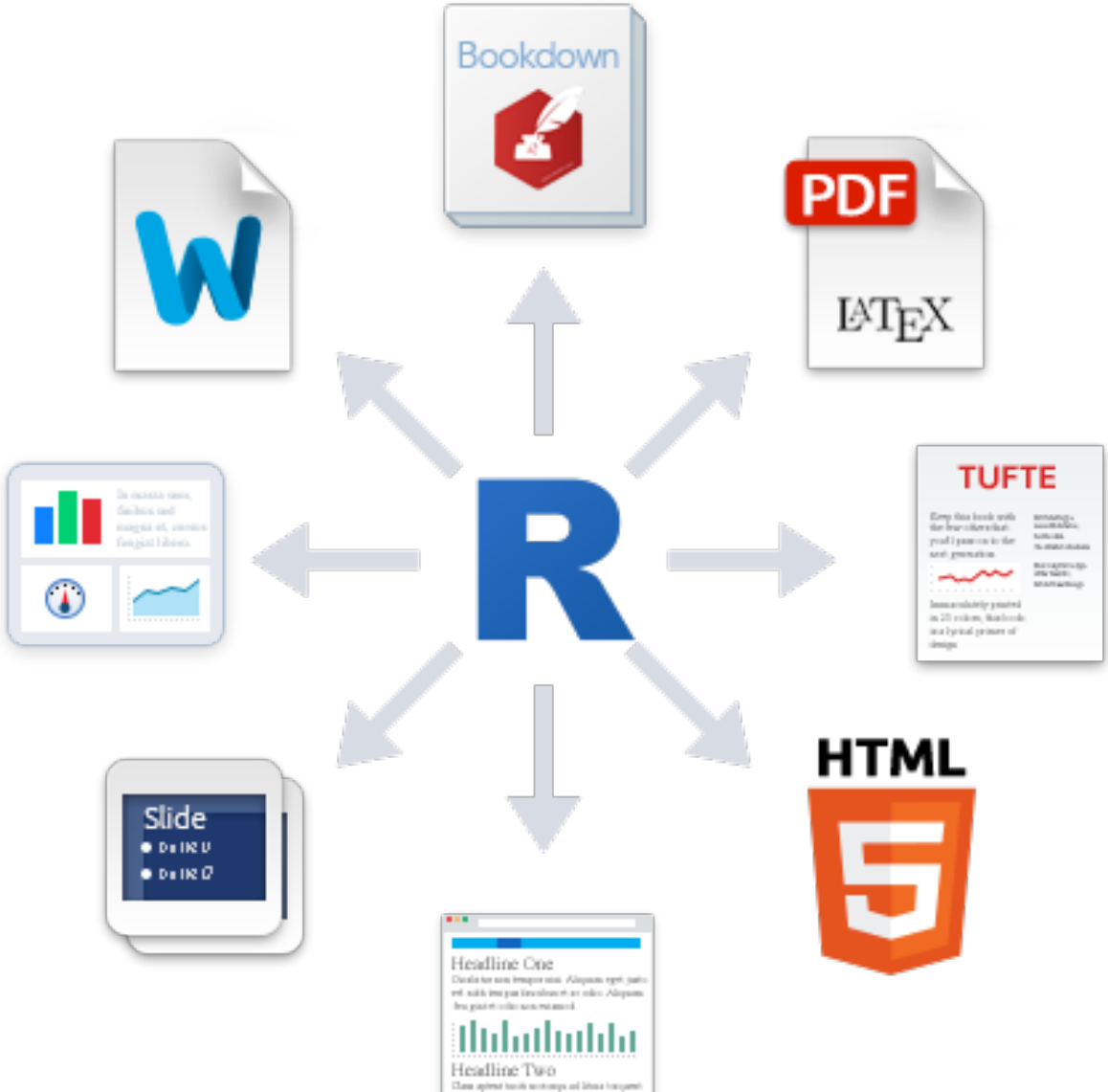
"R Markdown is an authoring format that enables easy creation of dynamic documents, presentations, and reports from R." - RStudio

- combines ("knits" together) markdown syntax (a simple, plain-text markup language) with *chunks* of R code

Figure 1: R Markdown flow for document rendering

What can R Markdown do?

- PDF, Word, or HTML output for documents/reports
- PDF and HTML (Slidy or ioslides) output for presentations



Creating and modifying a Rmd document script

To get started, install the `rmarkdown` package. You only need to do this once.

```
install.packages('rmarkdown')
library(rmarkdown)
```

Open a new R Markdown script: File > New file > R Markdown...
Select output format
- HTML or Word will work immediately
- PDF requires full TEX install

Insert title and author(s)

Integrating text and code chunks

- write plain-language description of what your code is doing
- write code chunk (offset by sets of 3 backticks)
- compile/render ("knit") often to check for errors

Text markup options for Rmd documents

- `**bold**` or `__bold__` -> **bold**
- `*italic*` or `_italic_` -> *italic*
- heading levels
- ordered (numbered) and unordered lists
- tables
- insert links
- insert images
- inline code or equations

- LaTeX markup can also be used within documents and presentations
- for example, `\vspace{0.2in}` adds 0.2 inches in between this line...

- and this line

Code chunk options

With the curly braces at the top of each code chunk, e.g., {r, tidy=TRUE}, there are a number of options you can specify.

`echo = FALSE` -> don't print code in the document/presentation
`eval = FALSE` -> don't evaluate code or print results
`tidy = TRUE` -> reformat code in a tidy way for display
`fig.width = 7` -> adjust width of figures (7 is default)
`fig.height = 7` -> adjust height of figures (7 is default)

. . . and many more.

Creating and modifying a Rmd presentation script

Identical to creating a new document script

- select output format
- insert title, author(s)

Text markup options for Rmd presentations

- does not support tables (yet)
- `#Header` starts a new slide, where `Header` is the slide title
- can also delineate between slides using `***` or `---`

- code chunk options are identical to Rmd documents

Document and presentation templates

- a few document templates available through RStudio
- some presentation templates available online

- play around with templates to edit some components (e.g., color scheme)
- design your own template for re-use

Continuing with R Markdown on your own

- There is a wealth of resources online
- Google is your friend
- Check out StackExchange and similar webpages
- Post questions in online forums
- New functionality constantly being developed

As with most computer programming, there will be a steep learning curve at the beginning. Once you get the hang of things, your work will go faster. It may seem time-consuming at first, but it will save you **SO. MUCH. TIME.** later.

TL;DR: R Markdown + LaTeX = Awesome Science