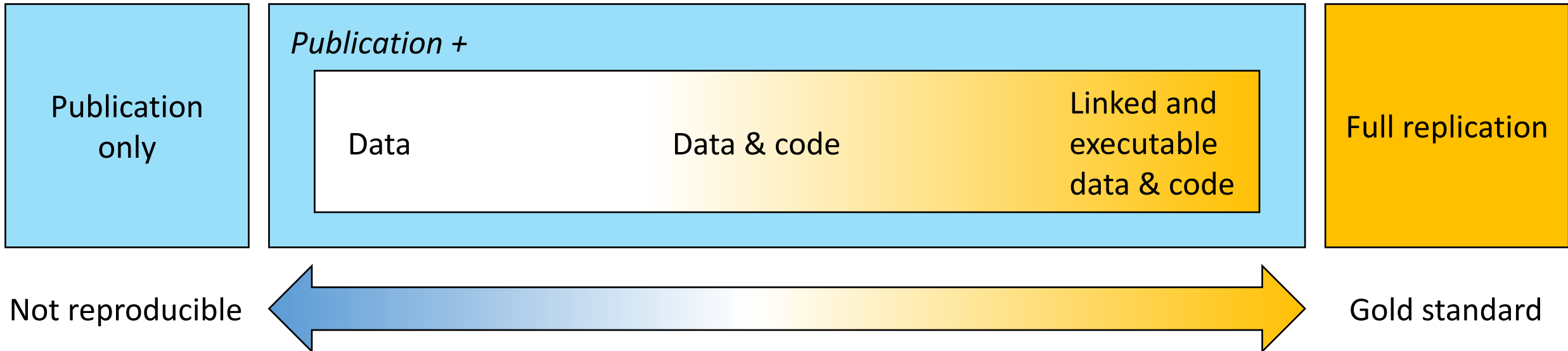


EEB 603 – Reproducible Science

Chapter 1 – Part C: Advanced R Markdown settings



**BOISE STATE
UNIVERSITY**

STRUCTURE OF CHAPTER 1

The chapter will be subdivided into four parts as follows:

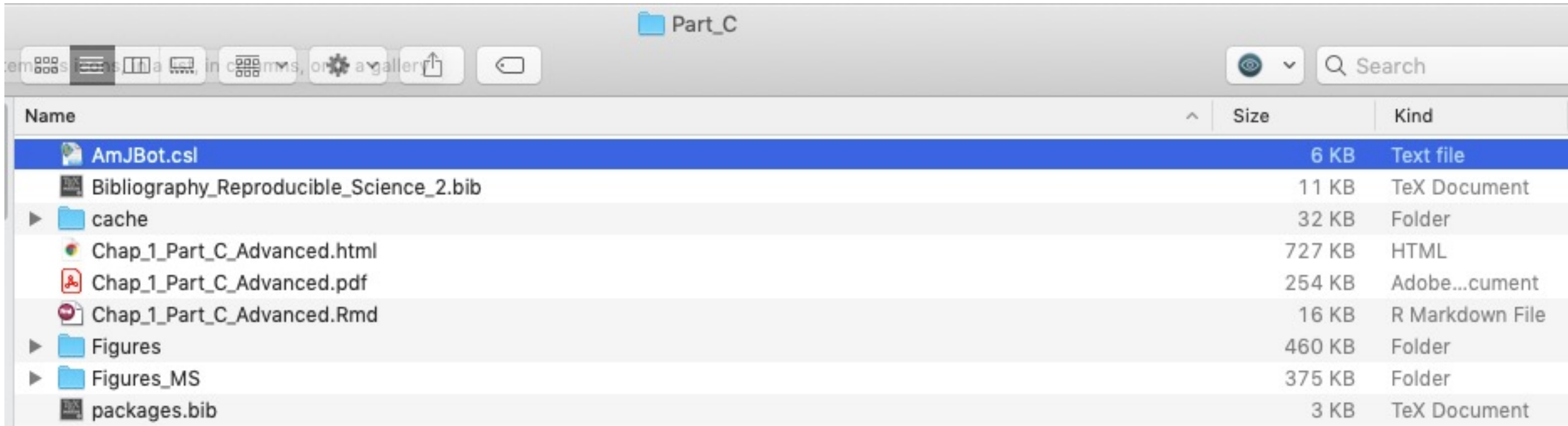
- **Part A:** Basics of R Markdown.
- **Part B:** Tables, Figures and References.
- **Part C:** Advanced R Markdown settings.
- **Part D:** User Defined Functions in R.

LEARNING OUTCOMES

This tutorial is devoted to part C and provides students with opportunities to learn procedures to:

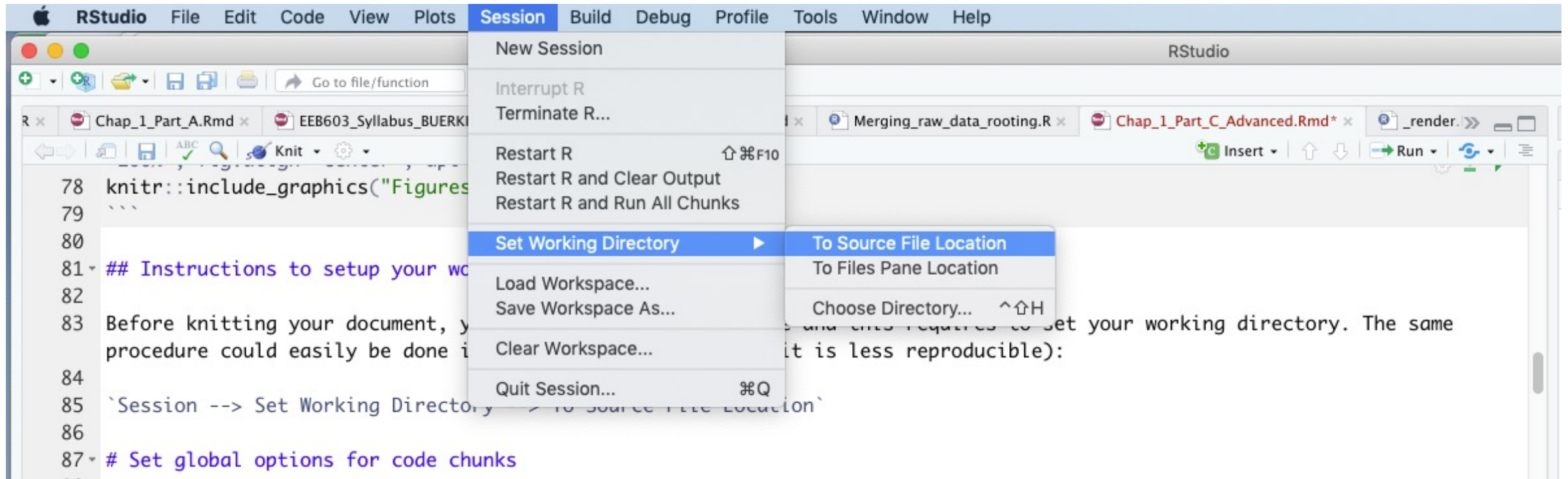
- Set your working directory.
- Set global options for code chunks related to:
 - text,
 - code decoration,
 - caching,
 - plots (output format and resolution),
 - positioning figures (close to associated code chunks).

SETUP YOUR WORKING DIRECTORY



Name	Size	Kind
AmJBot.csl	6 KB	Text file
Bibliography_Reproducible_Science_2.bib	11 KB	TeX Document
▶ cache	32 KB	Folder
Chap_1_Part_C_Advanced.html	727 KB	HTML
Chap_1_Part_C_Advanced.pdf	254 KB	Adobe...cument
Chap_1_Part_C_Advanced.Rmd	16 KB	R Markdown File
▶ Figures	460 KB	Folder
▶ Figures_MS	375 KB	Folder
packages.bib	3 KB	TeX Document

SETUP YOUR WORKING DIRECTORY: HOW TO



SETUP YOUR WORKING DIRECTORY: R SCRIPT

```
# Where is your script?  
# file.choose() prompts a window allowing selecting file location  
scriptLoc <- file.choose()  
  
# Infer folder path from scriptLoc  
dirPath <- dirname(scriptLoc)  
  
# Set working directory to script location  
setwd(dirPath)  
  
# Confirm that working directory is correct  
getwd()
```

SET GLOBAL OPTIONS FOR CODE CHUNKS

- To improve code reproducibility and efficiency and to follow publication requirements, it is customary to include a "code chunk" at the beginning of your .Rmd file to set global options applying to the whole document. Those settings are related to the following elements of your code:
 1. Text results.
 2. Code decoration.
 3. Caching code.
 4. Plots.
 5. Positioning figures.
- These general settings will be set using the **opts_chunk()** function implemented in *knitr*.

THE `OPTS_CHUNK$SET()` FUNCTION

- The *knitr* function `opts_chunk$set()` is used to change the default global options in a .Rmd document.
- Before starting, a few special notes should be known on the options:
 1. Chunk options must be written in one line; no line breaks are allowed inside chunk options.
 2. Avoid spaces and periods (.) in chunk labels and directory names.
 3. All option values must be valid R expressions just like how we write function arguments.

THE OPTS_CHUNK\$SET() FUNCTION: TEXT RESULTS

```
# Setup options for text results
```

```
opts_chunk$set(echo = TRUE, warning = TRUE, message = TRUE, include = TRUE)
```

- **echo = TRUE:** Include all R source codes in the output file.
- **warning = TRUE:** Preserve warnings (produced by `warning()`) in the output like we run R code in a terminal.
- **message = TRUE:** Preserve messages emitted by `message()` (similar to `warning`)
- **include = TRUE:** Include all chunk outputs in the final output document.
- If you want some of the text results to have different options, please adjust those in their specific code chunks. This comment is valid for all the other general settings.

THE OPTS_CHUNK\$SET() FUNCTION: CODE DECO

```
# Setup options for code decoration
```

```
opts_chunk$set(tidy = TRUE, tidy.opts = list(blank = FALSE, width.cutoff = 60), highlight = TRUE)
```

- **tidy = TRUE:** Use `formatR::tidy_source()` to reformat the code. Please see `tidy.opts` below.
- **tidy.opts = list(blank = FALSE, width.cutoff = 60):** This provides a list of options to be passed to the function determined by the `tidy` option. Here we format the code to avoid blank lines and with a width cutoff of 60 characters.
- **highlight = TRUE:** This highlights the source code.

THE OPTS_CHUNK\$SET() FUNCTION: CACHING CODE

```
# Setup options for code cache  
opts_chunk$set(cache = 2, cache.path = "cache/")
```

- **To compile your .Rmd document faster** (especially if you have computing intensive tasks), **you can cache the output of your code into files associated to each of your code chunks.**
- This process allows **compute intensive chunks to be saved** and the **output used later without being re-run.**
- The *knitr* package has options to only evaluate cached chunks when necessary, but this has to be set by users. Such procedure creates a unique MD5 digest (= a data storage technique) of each chunk to track when changes are present.
- When the option **cache=TRUE** (there are more granular settings; see below), the chunk will only be evaluated in the following scenarios:
 1. There are no cached results (either this is the first time running or the results were moved/deleted).
 2. The code chunk has been modified.

THE OPTS_CHUNK\$SET() FUNCTION: PLOT

```
# Setup options for plots  
opts_chunk$set(fig.path = "figures/", dev=c('pdf', 'png'), dpi = 300)
```

- Plots are at the core of your figures. We can take advantage of options implemented in the *knitr* package to output plots meeting publication requirements.
- **fig.path = "figures/"**: Set directory to save figures. As above, this folder doesn't need to exist prior to executing the code chunks. Files will be save based on code chunk title and assigned figure number.
- **dev = c('pdf', 'png')**: Save figures in both pdf and png formats.
- **dpi = 300**: The DPI (dots per inch) for bitmap devices (dpi * inches = pixels). Please look at publishing requirements to set this parameter appropriately.