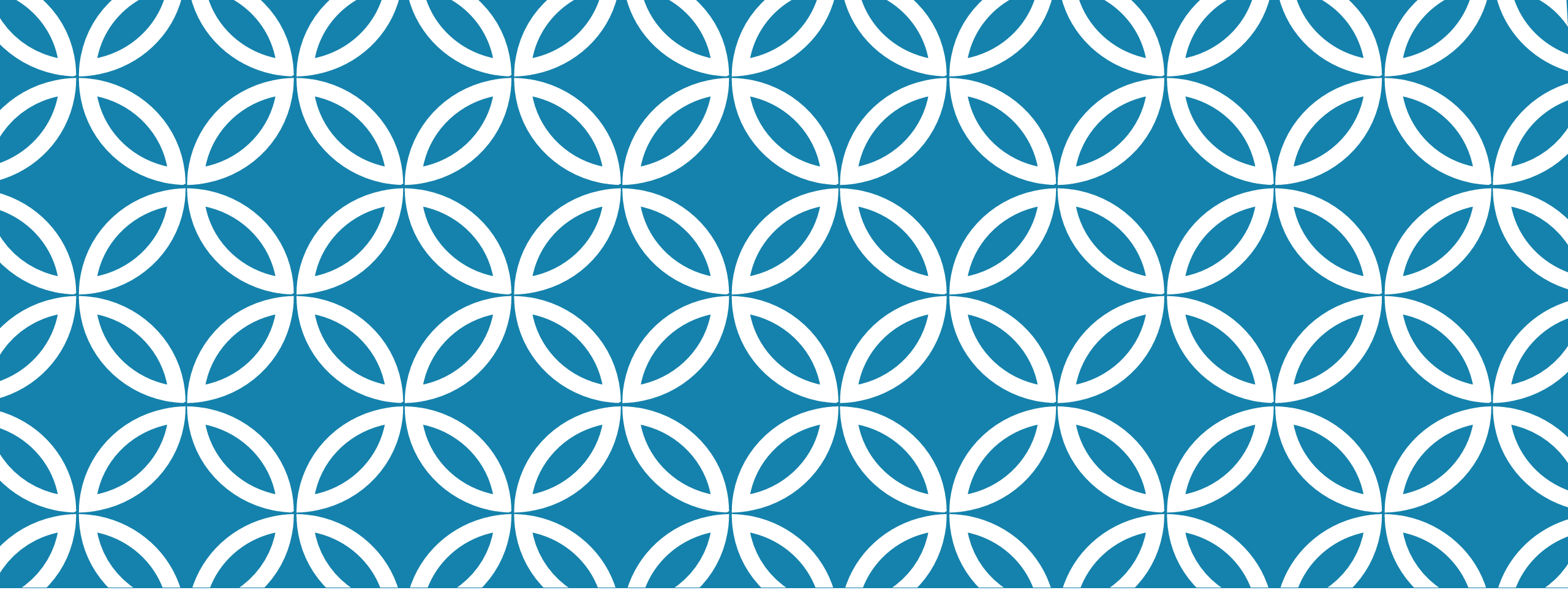


Data Analysis Basics

Patrick Mathias

Lesson 0

DLMP R Lessons

Lesson Goals

1. Understand good practices for performing data analysis safely
2. Learn the basic “tidy data” model

Lesson Objectives

1. Organize data and projects in a safe way
2. Use basic strategies for data inspection when working with a new data set
3. Describe the 3 principles of tidy data

GOOD DATA ANALYSIS PRACTICES

PRACTICE #1: BUILD A PROJECT FOLDER STRUCTURE TO ORGANIZE YOUR WORK

- You will find it much easier to recycle previous work if you have a clear structure for organizing your projects
- Make a folder for each project
- Include folders within the project folder for at least the following:
  - Data (raw data)
  - Output (figures and/or intermediate tables or files) – can choose to put intermediate files into a separate folder
  - Analysis files – can choose to keep this in main folder

PRACTICE #2: DATA ANALYSIS SHOULD BE SEPARATED FROM RAW DATA

- No matter how you choose to analyze your data you should always create a new file for your analysis
- Why?
  - Traceability: it is more difficult to identify mistakes in an analysis if you can't look back at what you originally received
  - Mistakes: if you overwrite an important cell or field you may not be able to identify the mistake and may not be able to fix it without re-requesting the data

HOW DO YOU SEPARATE YOUR ANALYSIS FROM RAW DATA?

- In Excel: use “Save As…” immediately after you receive your data
- In R or other programming languages, reading a file in generally does not change the original file
  - This is very helpful and can help enforce good practices
  - If you need to create an output file, you can use a write to file function
- Name your output/analysis file with a name that you’ll be able to recognize a year from now

SHOULD YOU SAVE INTERMEDIATE VERSIONS OF YOUR ANALYSIS?

- There is always a risk you will make a mistake in an analysis and want to backtrack
- For analysis in Excel, consider a couple strategies:
  - Save intermediate analysis files with dates to track
  - Use OneDrive – you can save the same file and revert to a previous version
- Analysis with a programming language emphasizes saving a script rather than a data file – you can re-execute the script if needed
  - May still want to use OneDrive or a version control system (e.g. git) for your script

PRACTICE #3: ALWAYS INSPECT YOUR RAW DATA

- Before diving into a data analysis, lay eyes on the data set
  - Can open the file in Excel
  - Programming language development environment may also allow you to look at the raw data – e.g. Rstudio
- Understand the meaning of rows and columns – will cover tidy data later

PRACTICE #3: ALWAYS INSPECT YOUR RAW DATA

- Inspect for any missing data
  - How is missing data represented? Blanks, NULLs (SQL), NAs (R), -1?
  - Programming languages provide functions that perform a summary of the data set
- If applicable, sort data column by column
  - Most helpful if each column = specific field or data element
  - Helpful to see the "lowest" and "highest" values by column
  - *Can identify unexpected data types*

COMPUTERS THINK ABOUT DATA DIFFERENTLY

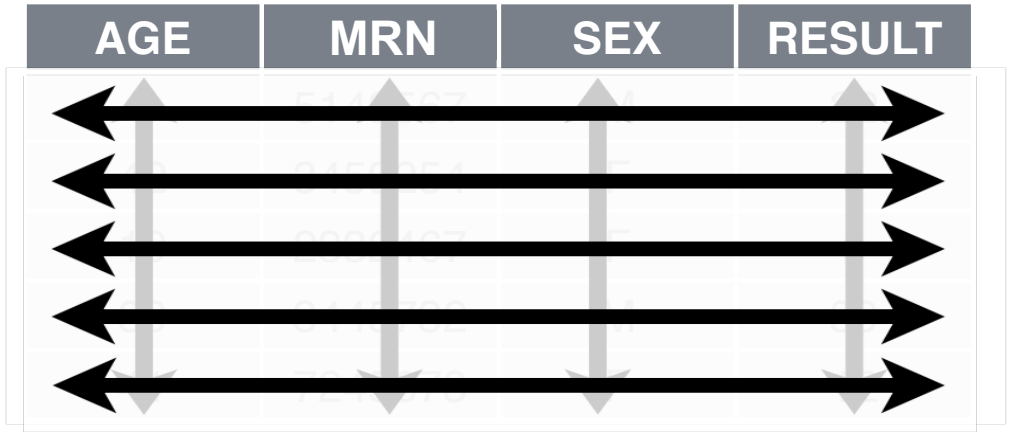
- Regardless of the tool you use to analyze data (e.g. Excel vs. R), the tool will categorize your data into data types
- Data types dictate the computer's "rules" for acting on your data
- Common data types:
  - Character
  - Number (could be integer vs. numeric with higher precision)
  - Logical (TRUE vs. FALSE)

SOME SPECIAL DATA TYPES ARE LINKED TO SPECIFIC BEHAVIOR

- Dates, times, or datetimes are common in lab data sets
  - Special rules required to sort these types: AM vs. PM, parsing hours, minutes, seconds
- Categorical data, or factors in R, may have a special representation to enable quicker summaries/calculations
  - Normal, STAT, and Timed may be more appropriate to represent as a categorical variable than a character
  - Categorial representation may be more efficient to count and display as different variables on a plot

PRINCIPLES OF TIDY DATA

TIDY DATA SUMMARIZED



A data set is **tidy** if:

1. Each **variable** is in its own **column**
2. Each **observation** is in its own **row**
3. Each **value** is in its own **cell**

Wickham, H. . (2014). Tidy Data. *Journal of Statistical Software*, *59*(10), 1–23.

CONSIDER A SINGLE DATA SET WITH 4 VARIABLES

- country
- year
- population
- cases

Public health data set intended to represent the cases of a disease in a population by country and year

REPRESENTATION 1

```
## # A tibble: 12 x 4
##    country      year type            count
##    <chr>       <int> <chr>           <int>
##  1 Afghanistan  1999 cases             745
##  2 Afghanistan  1999 population   19987071
##  3 Afghanistan  2000 cases            2666
##  4 Afghanistan  2000 population   20595360
##  5 Brazil       1999 cases            37737
##  6 Brazil       1999 population  172006362
##  7 Brazil       2000 cases            80488
##  8 Brazil       2000 population  174504898
##  9 China        1999 cases           212258
## 10 China        1999 population 1272915272
## 11 China        2000 cases           213766
## 12 China        2000 population 1280428583
```

REPRESENTATION 2

```
## # A tibble: 6 x 3
##   country      year rate
## * <chr>       <int> <chr>
## 1 Afghanistan  1999 745/19987071
## 2 Afghanistan  2000 2666/20595360
## 3 Brazil       1999 37737/172006362
## 4 Brazil       2000 80488/174504898
## 5 China        1999 212258/1272915272
## 6 China        2000 213766/1280428583
```

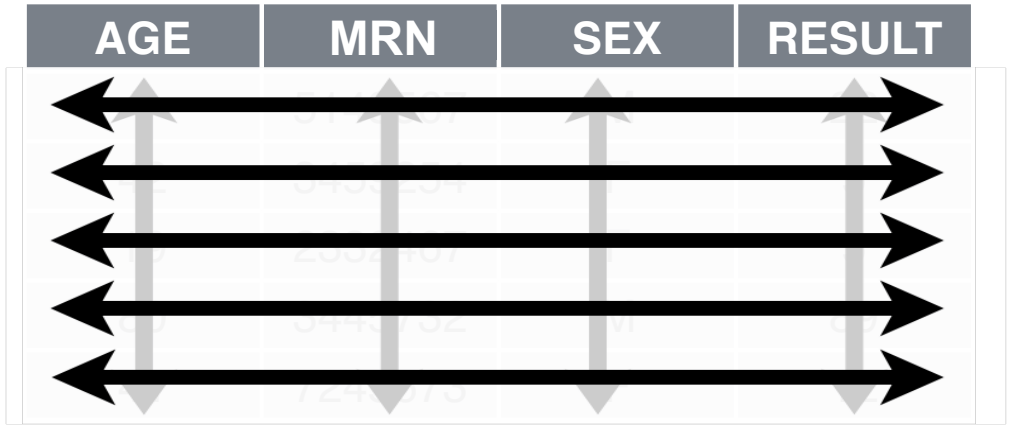
REPRESENTATION 3 IS TIDY

```
## # A tibble: 6 x 4
##       country      year  cases population
##         <chr>     <int>  <int>      <int>
## 1 Afghanistan  1999    745   19987071
## 2 Afghanistan  2000   2666   20595360
## 3 Brazil       1999  37737  172006362
## 4 Brazil       2000  80488  174504898
## 5 China        1999 212258 1272915272
## 6 China        2000 213766 1280428583
```

WHY KEEP THINGS TIDY?

- Consistent mental model for data = common data manipulation becomes easier
- Many tools in R (and Excel + other languages) make it very easy to do sophisticated analysis and visualization with tidy data
- Databases that store data from clinical information systems often already represent data in a tidy form

TIDY DATA SUMMARIZED



A data set is **tidy** if:

1. Each **variable** is in its own **column**
2. Each **observation** is in its own **row**
3. Each **value** is in its own **cell**

Wickham, H. . (2014). Tidy Data. *Journal of Statistical Software*, *59*(10), 1–23.

TIPS FOR GENERATING YOUR OWN DATA

BUILDING YOUR OWN DATA SET

- Data analysis projects beyond a certain complexity require creating your own variables or data frames (tables/spreadsheets)
- In addition to building tidy data sets, consider steps to make future analysis easier

NAMING VARIABLES

- Use variable (column) names that are easy for someone not performing your analysis to understand
  - “result_value” as opposed to “x”
  - “collection_time” instead of “time”
- Stick to naming that will make it easier for a computer to read the data
  - Avoid spaces in names: “result_value” instead of “result value” (space has meaning in languages like R)
- All lowercase (or uppercase) will be easier for you to type, even if it is not easier for the computer to parse

RESPECT DATA TYPES WHEN CREATING VARIABLES

Variable inspection and summarization is much easier (and less error prone) if known data types are used

run_id	value
1	5
2	cancelled
3	11
4	< 2
5	4

run_id	value	comment
1	5	
2		cancelled
3	11	
4		< 2
5	4	

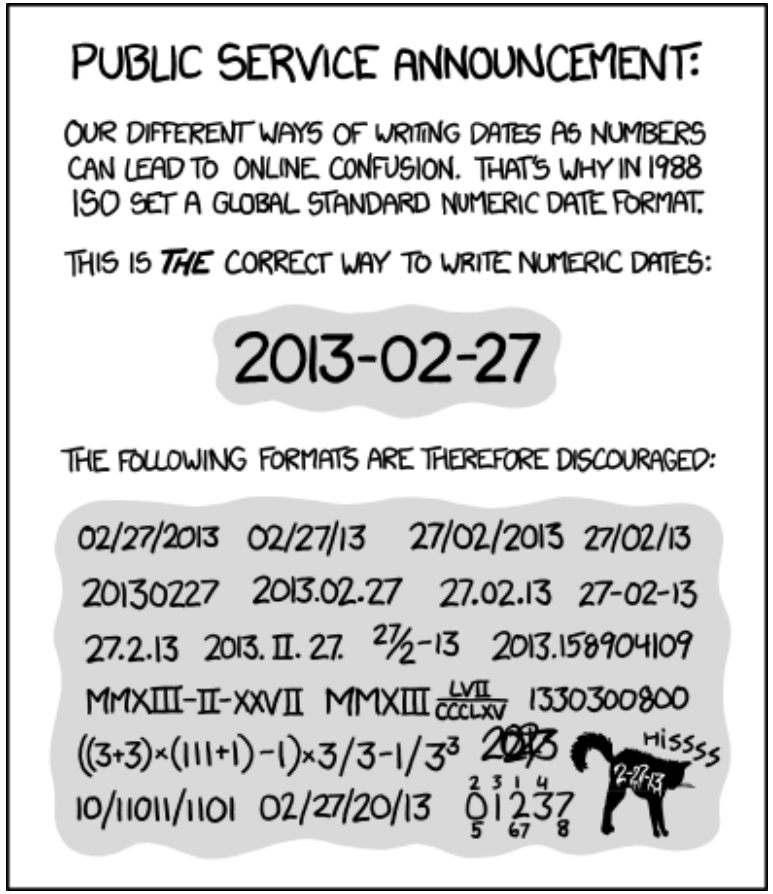
CONSISTENCY IS CRITICAL FOR ANALYZING CATEGORICAL VARIABLES

Summarization is seamless when consistent values are used to represent binary or categorical variables

subject	antibiotics	growth	antibiotic
1	Y	+	azith.
2	N	-	azithromycin
3	yes	+	penicillin
4	n	+	Penicillin

subject	antibiotics	growth	antibiotic
1	yes	yes	azithromycin
2	no	no	azithromycin
3	yes	yes	penicillin
4	no	yes	penicillin

USE STANDARD FORMATS FOR DATES AND TIMES



Obligatory xkcd (https://xkcd.com/1179/)

- Learn to know and love the ISO 8601 standard for representing dates and times
- Prevent ambiguity when interpreting and parsing
- Dates in filenames sort chronologically
- Excel unfortunately does not encourage the use of this format (but should parse it without a problem)

R COURSE INTRODUCTION

COURSE GOALS AND OBJECTIVES

- Establish good practices for working with data safely
- Teach features of the R programming language that improve reproducibility in clinical data analysis
- Demonstrate how R can be used to perform analyses of laboratory operational data
- Establish a basis of understanding in the 'tidy' approach to data analysis

SESSIONS

Loading Data to Create a Dataframe

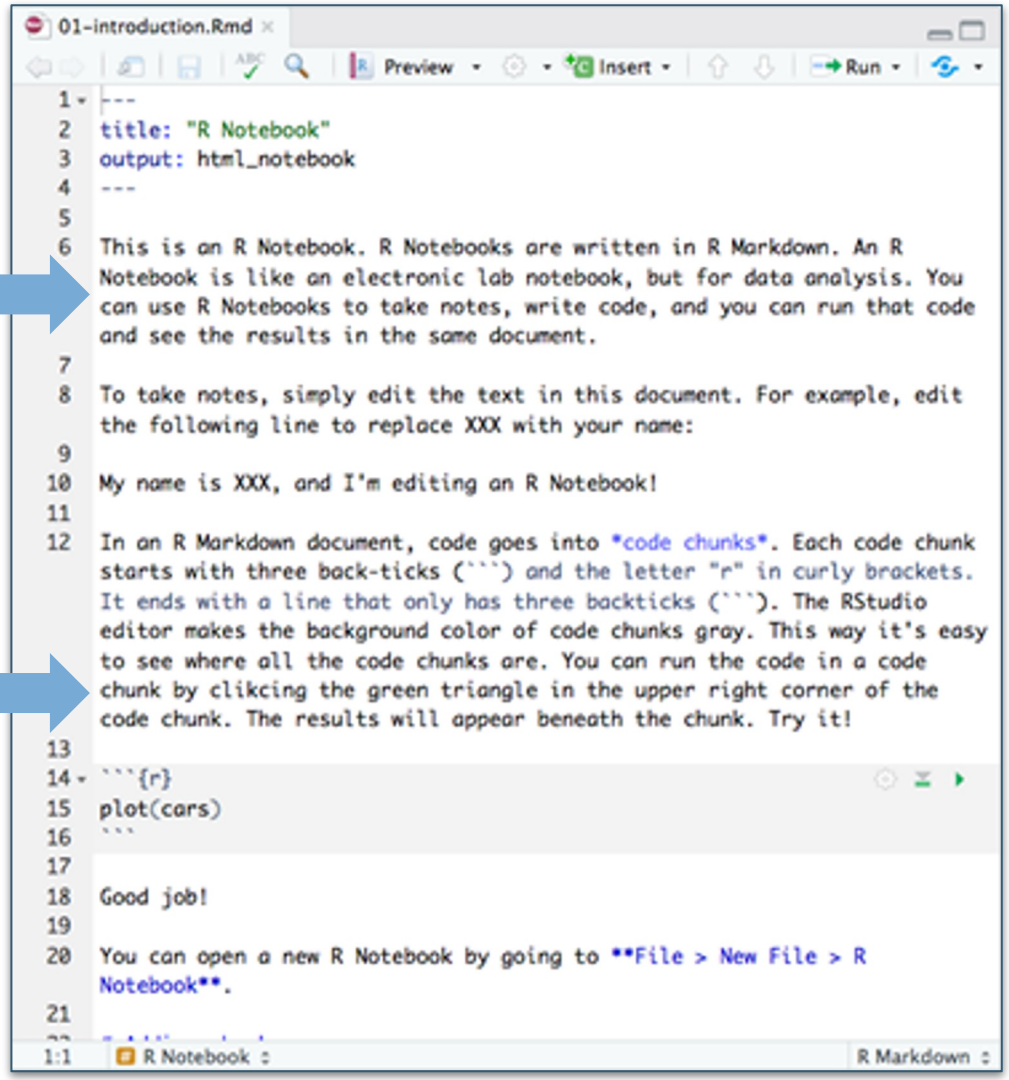
```
data_frame <- read_csv("file_name")
```

Your Turn

Introduce yourself to your neighbors

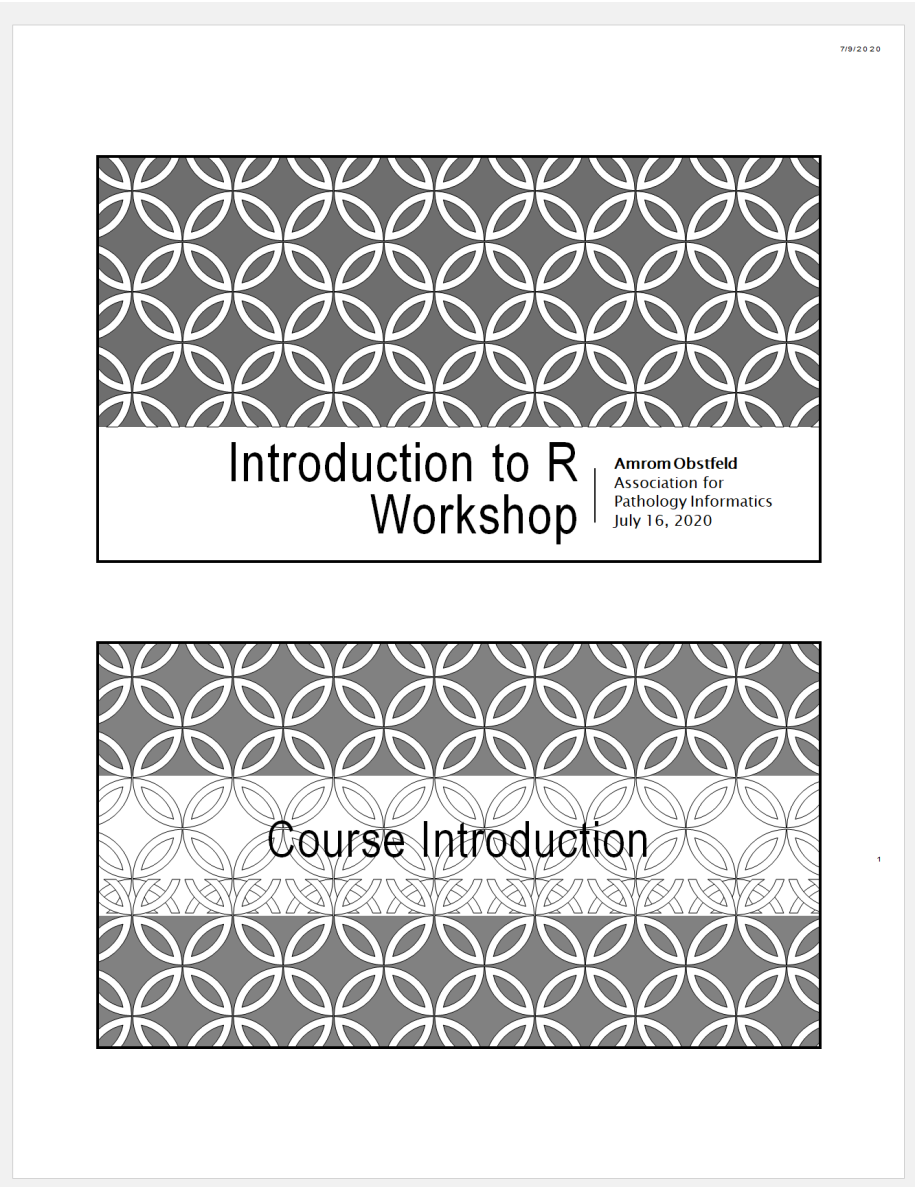
- Who are you?
- Where are you from?
- What do you do with data?
- Have you ever used R?

3:00



WORKSHOP COURSEBOOK

- Coursepack folder on website contains:
  - PDFs for slides
  - Cheatsheets



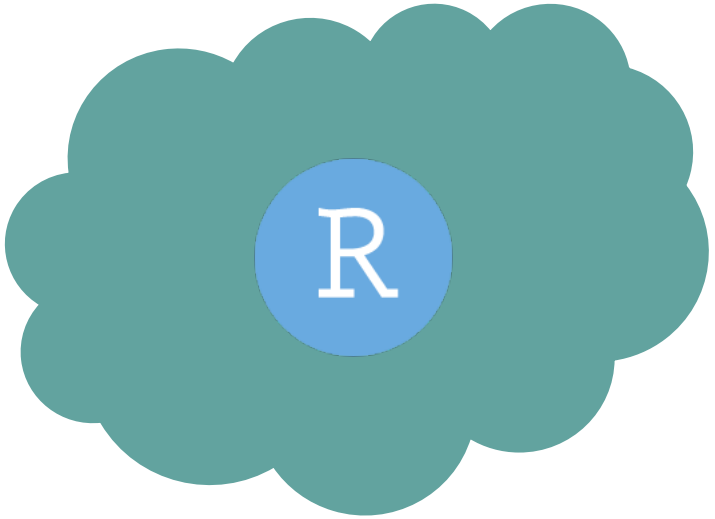
USING ZOOM IN A HYBRID SETTING

- Raise hand in room or virtually for assistance
- Remote participants muted
- Chat window
- Non-verbal feedback

TIPS FOR LEARNING

- Cheatsheets show how to do common things – orient yourself with them early
- The best way to learn to code is by doing
- Practice is key!
- Programming is hard, even for those with a lot of experience. Find resources and ask for help!

RSTUDIO CLOUD WILL BE THE TOOL FOR OUR R LESSONS



RStudio Cloud
Hosted on a server (in the cloud)

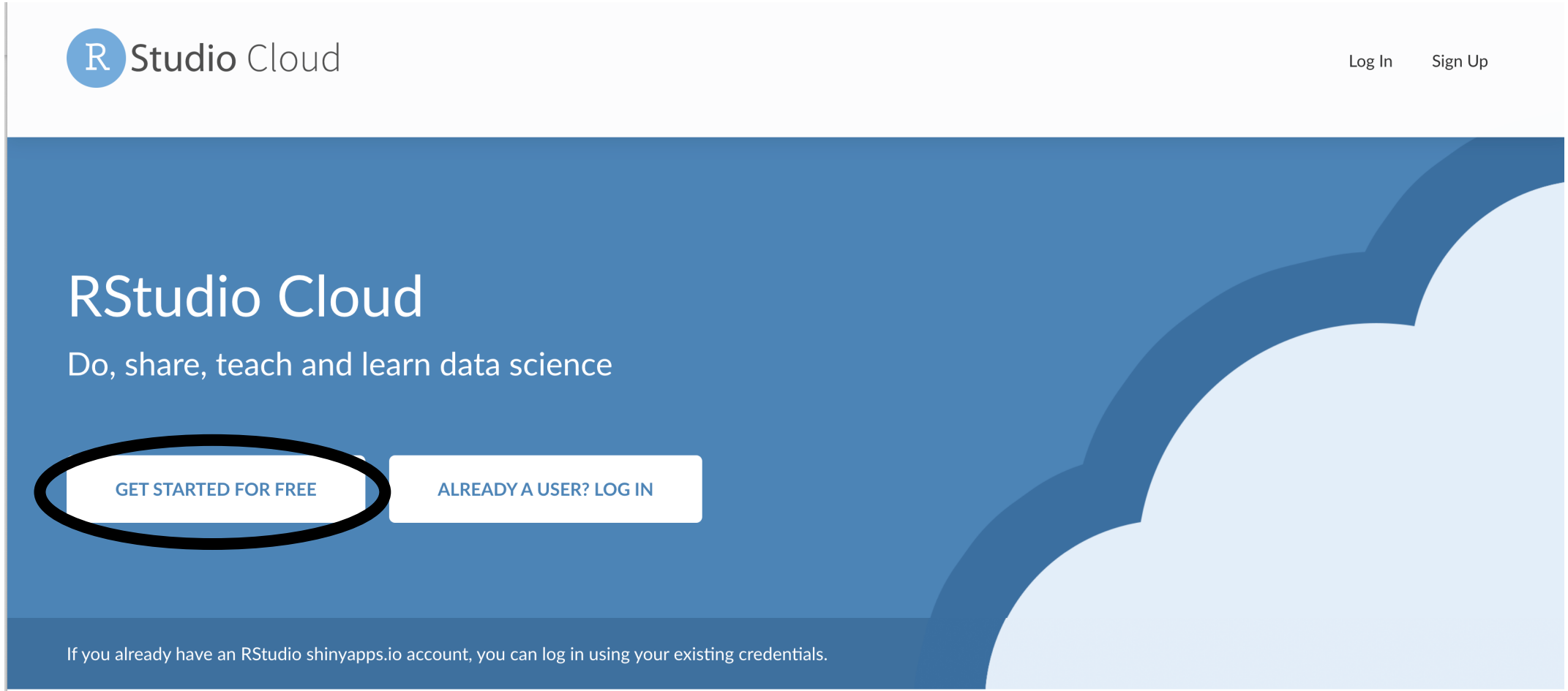
RStudio Server
Hosted on a server (DLMP Server)

RStudio Desktop
Installed locally on your computer

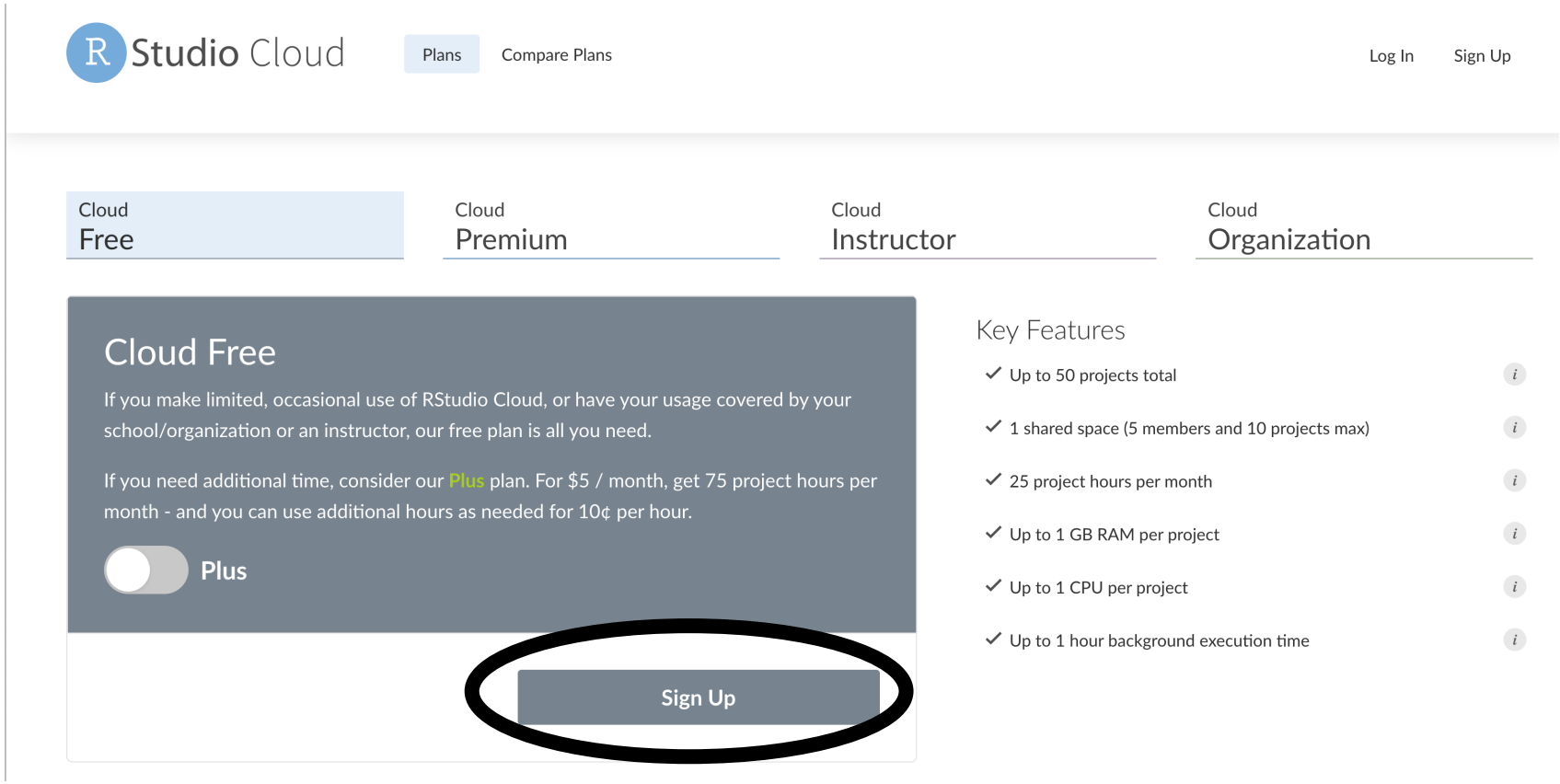
Note: Use RStudio Cloud only for this course. Do not upload protected health information to the cloud!

SET UP YOUR RSTUDIO CLOUD ACCOUNT

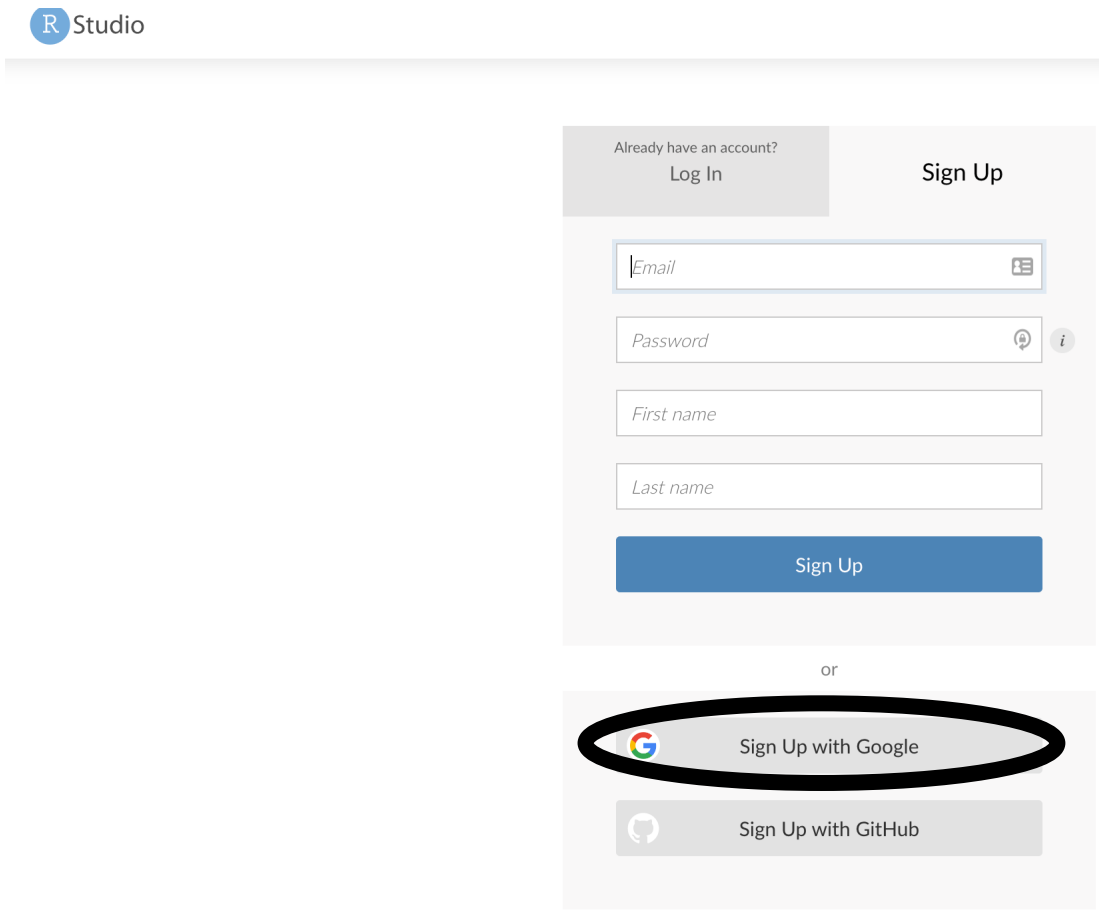
Go to rstudio.cloud on your web browser



SIGN UP FOR CLOUD FREE



SIGNING UP WITH GOOGLE USING YOUR UW EMAIL CAN SHORTCUT THE SETUP STEPS



Then select your Google account associated with your UW email (if you have Google set up already)

ONWARD!

Survey link

- Will pick up with orientation to RStudio tomorrow
- Please fill out survey: https://forms.gle/G6HhAxsHeBXKJtkZ7
- Course material available at: https://github.com/pcmathias/dlmp-data-analysis-with-r

- After first lessons, we will provide instructions for signing into our DLMP server to use RStudio (without having to install on your local desktop)