Introduction to R

1. What is R? Why use R?
2. Installing R and RStudio
3. Working with R
4. Tidyverse
5. R Commander (Optional)

Examples:

1. R Example #1
2. *R Help Hints*

1. What is R? Why use R?

- R is a command line programming language for statistical computing.
- R is free!
- It makes great graphics.
- It provides a wide range of packages and functions for statistical analysis.
- Command line programming helps achieve “reproducible research”.
- It is open source software.

- R comes with a standard set of packages, but many more (thousands!) are available for download.
- I have heard it said that “The best thing and worst thing about R is the large number of packages that are available”.
- This is a “good” thing because it greatly extends R’s functionality.
- This is a “bad” thing because the packages are written by many different people and the syntax is not consistent.

- Every time you open R, you will see the following warning:

  “R is free software and comes with ABSOLUTELY NO WARRANTY.”

2. Installing R and RStudio

- To install R go to: www.r-project.org
- Click on “download R”.
- Select a CRAN mirror.
- Download R (binaries for base distribution) for your operating system.
- Be aware that R is updated frequently, with a few major releases each year.

- RStudio is an “integrated development environment” or IDE.
- It is a program that makes it more convenient to use R.
- Go to the website www.Rstudio.com and download RStudio. We will use the free Desktop Open Source License.

3. Working with R

- NOTE: It is assumed that students have already reviewed the R Basics Bootcamp modules.
- There will be considerable overlap in this intro but at a faster pace.

Starting and Saving

- From RStudio, File > New File.
- Choose R Markdown > Document.
- Alternatively can use R Script.
- With either of these approaches we save the code and can easily recreate the results.
- Save file as you go and certainly save before exiting R.

Importing Data

- To import CSV files (comma separated values), we will use `read.csv()`.
- Ex: `cows <- read.csv("C:/hess/STAT511/CH2_Cows.csv")`
- Alternatively, the file.choose() function allows files to be chosen interactively (allowing the user to “browse” to find a file.)
- Ex: `cows <- read.csv(file.choose())`

Objects vs Functions

- R objects store information.
  - Every object in R has a type (class).
- R functions "do things".
  - The syntax of functions is the function name followed by open parentheses, a comma separated list of arguments (possibly named) and then a closing parentheses:
    `functionname(name1= arg1, name2 = arg2, …)`

  Hint: In RStudio, use the tab key to see function arguments!

$ Operator

- $ can be used to reference a specific column from a data.frame.
- In other words, $ can be used to extract a vector from a data.frame.
- Ex: `mean(cows$gain)`

data = Option

- Many (but not all) R functions have a data = option. This allows us to avoid $.
- When available, use the data = option!
- Ex: `aggregate()` or `boxplot()`
- If the data = option is not available, can use the `with()` function.
- Ex: `with(mean(gain), data = cows)`
- Another option is `attach()`. Not shown.

Finding Help

- Finding help with R:
  - To find help on a specific topic (for example *mean*) just type `?mean` or go to Help and search for “mean”.
  - Rseek.org and Quick-R are helpful sites.
- Always look at the data! The `View()` and `str()` functions are very helpful.

Import Options (Optional)

- Full list of options (skipping rows, comment characters, etc) are given in the help page: `?read.csv`.
- `read.table()` is a (related) alternative to `read.csv()`. Note: The defaults are different for `read.csv()` and `read.table()`.
- Excel files can be imported (and exported) to R. This can be done using the XLConnect or readxl packages.

Working Directory (Optional)

- The “working directory” is the default directory for R (used for saving).
- Use `getwd()` to find the location and `setwd()` to set the location.
- Specifying the working directory is not necessary but can be helpful if you are working on several different projects.

Saving R Objects (Optional)

- It is possible to save an individual R object using the `save()` function.
- Saved objects can be “restored” (in a new R session) using `load()`.
- I sometimes use this approach after “cleaning” a large dataset or for saving a results in a data.frame.

R Packages

- R packages contain additional functions (and data).
- Before using a package for the first time, you may need to install it. This step only needs to be done once! If a package is already installed, you will still need to load it during the R session when you want to use it.
- From RStudio, use the “Packages” tab to install and load packages. Packages can also be loaded using the `library()` or `require()` functions.

4. Tidyverse

- Tidyverse refers to a popular collection of R packages for data manipulation and graphing.
- These packages are not critical for most STAT511 topics.
- But because tidyverse is (1) extremely popular (2) extremely helpful for more advanced analyses, I will use it for some examples.

5. R Commander (Rcmdr)

- Rcmdr (pronounced R commander) is a more user-friendly interface for running some basic statistics in R.
- Students are welcome to use Rcmdr when possible, but will still be responsible for interpreting R code on exams.
- Rcmdr is optional for STAT511!
- Start by installing the Rcmdr package.

Working with Rcmdr

- Type the following command at the command line prompt (>)

  ```
  library(Rcmdr)
  ```

  and hit return.
- If successful, the GUI should open in a new window.
- You may find that additional components need to be installed the first time you “load” Rcmdr.
- Problems installing? See help at: https://socialsciences.mcmaster.ca/jfox/Misc/Rcmdr/

Rcmdr: Importing Data

- Go to Data → Import Data → from text, clipboard or URL.
- A dialogue box will appear.
  - Choose a name for the data set and the appropriate “field separator”.
  - If you are using a CSV file, be sure to change the “field separator” to “,”.
- After hitting OK, navigate to where the data is saved.
- The imported data will now be the active data set.

Rcmdr: Summary Statistics

- Go to Statistics → Summaries → Numerical summaries.
- From the dialogue box
  - From the Data tab, choose the variable.
  - If you have both a numerical and categorical variable and you may want to “Summarize by groups…”
  - From the Statistics tab, select the quantities you want returned (defaults are good).
- The mean, sd, “5 number summary” and sample size will appear in the Output Window.

Rcmdr: Summary Graphics

- Go to Graphs → Histogram…
  - From the Data tab, choose the Variable.
  - Can also “Plot by groups..”
- Go to Graphs → Boxplot…
  - From the Data tab, choose the variable.
  - Can also “Plot by groups…”
- Note that graphs appear in a new window (R Graphics) in the R window.