Class Setup

Organization + Intro to R

Material in this section is covered by Chapters 1 and 2 on the notes website.

Download section 2 Rmd.

Set up course directory

Create a folder named `STAT240`.

Add subfolders `data`, `discussion`, `hw`, `lecture` (copy Github)

Files must be set up consistently!

Example code to read in file:

```
df <- read_csv("../../data/example.csv")
```

There are four main windows in Rstudio:

- Console
- Environment
- Viewer
- Editor

If you do not see any, use the View tab.

R code is run in the **console**.

- Run 1 line at once
- Good for simple actions

Let's use it as a calculator.

We keep track of variables in the **Environment** tab.

- Can look up variable names
- Clearing environment resets variables

Want to use meaningful variable names!

The **Viewer** tab has many uses.

- R help
- File explorer

Can use ? with any R function to open the help file and ?? to search.

The **Editor** pane is where we edit documents.

- Code here can be saved for later
- Need to run code separately

We will be working with the .Rmd format.

Try stuff out!

- Perform a mathematical operation with variables (not just numbers)
- Clear your console output - can you still get the old code back?
- Find the Apperance Options menu and change the color scheme
- Try to figure out how to find your R version (it's a pre-defined object)

Let's focus on the editor, where we work with R Markdown documents. Three parts:

- YAML (preamble)
- Markdown language (plain text)
- R code chunks

We write in a plaintext editor, then *knit* to a nicely formatted file.

The YAML begins and ends with --- at the start of the document.

- Information on author, date, output type
- Most can be left the same

There are different output formats, but we will stick with .html.

The .Rmd is primarily **Markdown**.

- Used to write text (as opposed to code)
- Lightweight

See the R Markdown Cheatsheet for details.

In the **code chunks**, we write R code.

- Separated by three backticks ‘‘‘ and {r}

- Insert with **Code** → **Insert Chunk**
- Or on Windows, **Ctrl+Alt+I**
- Or on Mac, **Command+Option+I**

Leave a space before/after chunks!

To run a single line of R code:

- **Ctrl/Command + Enter**

To run the whole chunk:

- Green "play" button
- Or **Ctrl/Command + Shift + Enter**

What is happening in Rstudio when you do this?

Knit a document with the blue ball of yarn button. You may need to follow the prompts to install a package.

Knit the file and check that you have a new .html in the same directory as the .Rmd.

Now, let's break it. Make a new code chunk in the document with **Ctrl+Alt+I** or **Command+Option+I**

- Add the line "`months <- age * 12`"
- Try to knit. What error message do you see?
- How can we fix this?

Be careful with your Environment tab.

R is FOSS - anyone can contribute to it.

- Pre-made collections of commands and data are **packages**
- Our class will make use of the `tidyverse` package

  `tidyverse` helps with data manipulation and visualization

A package needs to be downloaded from the internet *once* - like buying a toolbox from the store.

When you open RStudio, the "toolbox" is closed, and you don't have access to the commands in the package.

Open the toolbox by **loading** a package.

Run `install.packages("tidyverse")` in your R console.

- "use a personal library": pick yes
- "install from source": pick no, then yes if it fails
- "compile": pick no

We have to load the package before we can use it.

One of the functions in tidyverse is called `mutate`.

- Run `?mutate` in the console - you should get an error.
- Now run `library(tidyverse)` first.

Packages must be loaded *each time you open RStudio*.

We've defined several variables with names we decided on. However, R has several special 'reserved' keywords.

- Booleans: `TRUE` and `FALSE`
- Numbers: `NaN`, `Inf/-Inf`

We'll learn others later on.