Lab 1 Solved: Introduction to R

Harris Coding Camp

Summer 2023

Open an R script so you can save your code for later!

1. If you have not yet, run the following code to install the `tidyverse`.

```r
# download from the internet -- you only need to do this once.
install.packages("tidyverse")
```

2. This code returns an error. Why? Fix it and install the packages.

   Answer: need to put package names in quotes here. `install.packages("haven")`

```r
install.packages(haven)
install.packages(readxl)
```

3. In order to have access to `tidyverse` functions, you need to load the library. Run the following code.

```r
library(tidyverse)
```

4. Run the following code to make sure you have installed successfully. Then, use the function `View()` to see the full dataset.

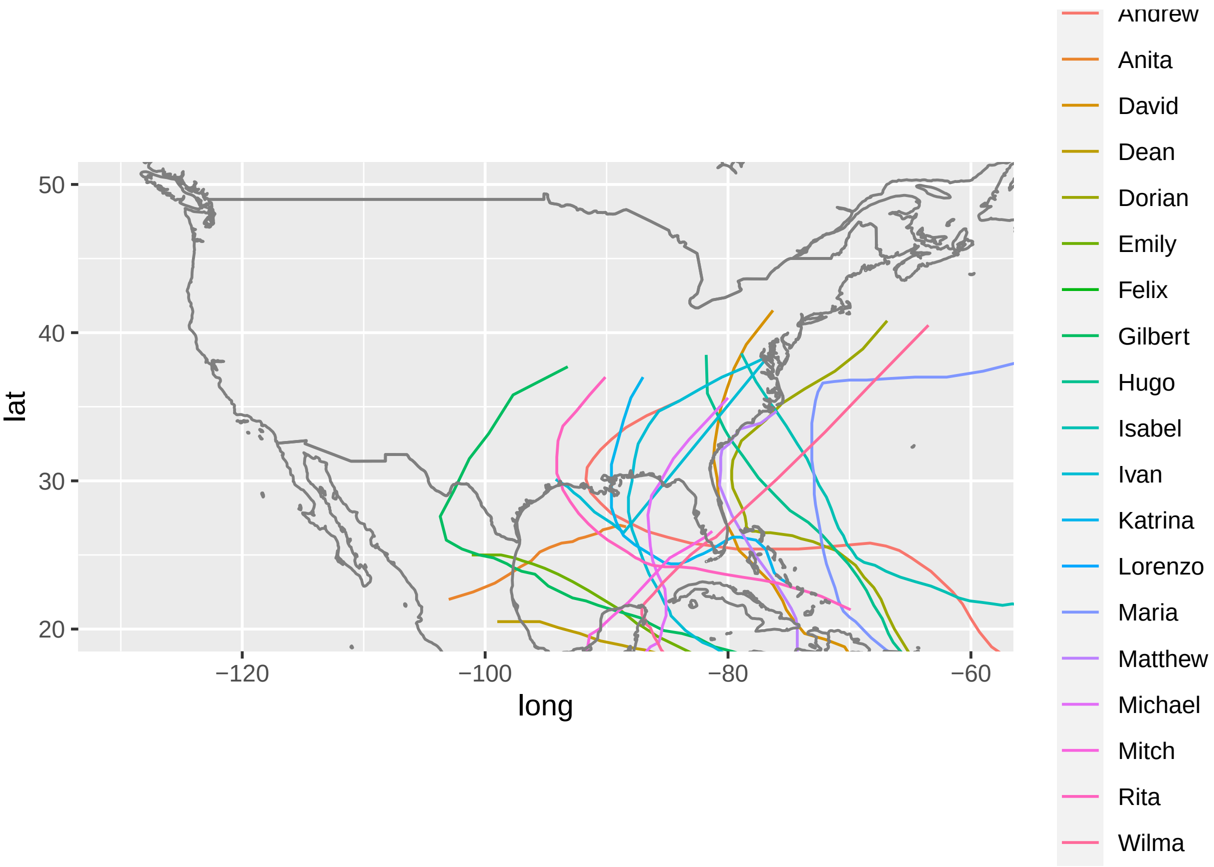
```r
# storms is a dataset that comes with the tidyverse
# use View(storms) to see the full dataset
big_storms <- storms %>%
  group_by(name, year) %>%
  filter(max(category, na.rm = TRUE) == 5)
```

5. Assign the code above to the name `big_storms`. What is the difference between `storms` and `big_storms`?

   big_storms has fewer storms, because we remove storms that never made it category 5

6. This code makes the map seen below. If you run it you will get a message that says your code is missing a required package, R 4.1 will ask you to install the package. Choose yes. [This package is not essential – if the package fails to install, make a note of what error messages you get and then move on. You may consult with TAs after finishing the tasks below.]

```r
# Heads up: copy and paste might mess up the quotes or other formatting
# look carefully if the code doesn't work.
ggplot(aes(x = long, y = lat, color = name), data = big_storms) +
      geom_path() +
      borders("world") +
      coord_quickmap(xlim = c(-130, -60), ylim = c(20, 50))
```


50
40
30
20
lat
−120
−100
−80
−60
long
Andrew
Anita
David
Dean
Dorian
Emily
Felix
Gilbert
Hugo
Isabel
Ivan
Katrina
Lorenzo
Maria
Matthew
Michael
Mitch
Rita
Wilma

Warm-up

1. Which of these allow you to pull up the documentation for a command in R?

a. `*` *b.* `?`
b. `help()`
c. `documentation()`

2. In the code block below, run code that will pull up documentation for the function `paste0()`.

```
?paste0()
```

What does this function do?

paste0() concatenates vectors and makes them characters

3. The second example in ?paste0 is

```
## If you pass several vectors to paste0, they are concatenated in a
## vectorized way.
nth <- paste0(1:12, c("st", "nd", "rd", rep("th", 9)))
```

This example uses a bunch of code/concepts we haven't covered yet!

a. Try to make sense of how the code works by running each input seperately. i.e. what does the code `1:12`, `c("st", "nd", "rd", rep("th", 9))` and `rep("th", 9)` do? And what does `paste0` do with it's inputs?
b. The function `paste0` is given two inputs here. What are they? *1:12 is a vector of numbers and c("st", "nd", "rd", rep("th", 9)) is a vector of charatcters*
c. What does `paste0` do with two vector inputs? *concatenates them in a vectorized way*

4. *Assigning variables.* `score` is tracking your score in a game. The first round you got 3 points, the next round you got 2 points.

Your partner is keeping track of your score and wrote the following code.

```
# Solution
score <- 0
# round 1
score <- score + 3
# round 2
score <- score + 2
```

a. What is `score` now?

   score was still 0

b. Fix the code so that `score` is accurately tracking your score.

Comment: A common error for new programmers is to not assign output to a name!

The rest of the problems are answered in the lab.