Intro to R

Introduction

Welcome to class!

1. Introductions
2. Class overview
3. Getting R up and running

[Photo by Belinda Fewings on Unsplash]

Before we start ..

Poll: How are you feeling right now?

About Us

Carrie Wright (she/her)

Senior Staff Scientist, Fred Hutchinson Cancer Center

Associate, Department of Biostatistics, JHSPH

PhD in Biomedical Sciences

Email: cwrigh60@jhu.edu Web: https://carriewright11.github.io

About Us

Ava Hoffman (she/her)

Senior Staff Scientist, Fred Hutchinson Cancer Center

Associate, Department of Biostatistics, JHSPH

PhD in Ecology

Email: ava.hoffman@jhu.edu Web: https://avahoffman.com

About Us

Candace Savonen (she/her)

Research Associate, Department of Biostatistics, JHSPH

Masters in Neuroscience

Former Data Analyst for Childhood Cancer Data Lab

Email: csavone1@jhu.edu

Website: https://www.cansavvy.com/

About Us - TAs

Padmashri Saravanan (she/they)

1st Year MHS Student, Department of Epidemiology, BSPH

MSc in Mathematics, Birla Institute of Technology and Science, Pilani

Email: psarava1@jhu.edu

About you!

Please introduce yourself!

Find the “introductions” channel on Slack: `https://intro-to-r-jan2023.slack.com/`

What is R?

- R is a language and environment for statistical computing and graphics developed in 1991
- R is the open source implementation of the S language, which was developed by Bell laboratories in the 70s.
- The aim of the S language, as expressed by John Chambers, is “to turn ideas into software, quickly and faithfully”

[source: http://www.r-project.org/, https://en.wikipedia.org/wiki/S_(programming_language), https://en.wikipedia.org/wiki/Bell_Labs)]

What is R?

- **R**oss Ihaka and **R**obert Gentleman at the University of Auckland, New Zealand developed R
- R is both open source and open development

[source: http://www.r-project.org/, https://en.wikipedia.org/wiki/R_(programming_language)]

Why R?

- Free (open source)
- High level language designed for statistical computing
- Powerful and flexible - especially for data wrangling and visualization
- Extensive add-on software (packages)
- Strong community

[source: https://rladies-baltimore.github.io/]

Why not R?

- Little centralized support, relies on online community and package developers
- Annoying to update
- Slower, and more memory intensive, than the more traditional programming languages (C, Perl, Python)

[source -School vector created by nizovatina - www.freepik.com]

Introductions

What do you hope to get out of the class?

Why do you want to use R?

[Photo by Nick Fewings on Unsplash]

Course Website

http://jhudatascience.org/intro_to_r

Materials will be uploaded the night before class. We are constantly trying to improve content! Please refresh/download materials before class.

Learning Objectives

- Understanding basic programming syntax
- Reading data into R
- Recoding and manipulating data
- Using add-on packages (more on what this is soon!)
- Making exploratory plots
- Performing basic statistical tests
- Writing R functions

Course Format

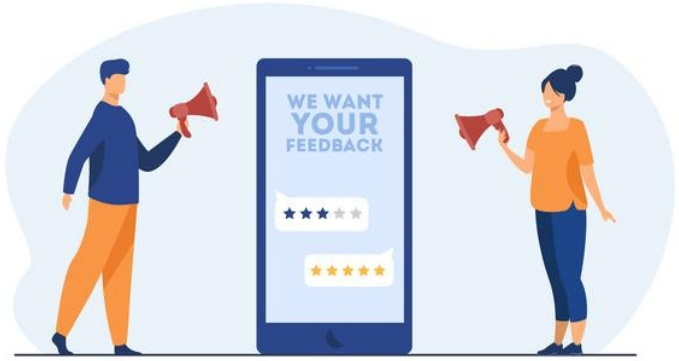
- Lecture with slides (possibly “Interactive”)
- Lab/Practical experience
- Two 10 min breaks each day - timing may vary
- Jan 9-20, 2022, 1:30PM-5:00PM on Zoom
- No class on Jan 16th for Martin Luther King, Jr. Day
- Last two classes will focus on final project

CoursePlus

https://courseplus.jhu.edu/core/index.cfm/go/syl:syl.public.view/coid/17889/

- Surveys throughout the class for the instructors
- Upload homework

End of class Survey - link in email.



[source - Banner vector created by pch.vector - www.freepik.com]

Grading

1. Attendance/Participation: 20% - this can be asynchronous - just some sort of interaction with the instructors/TAs (turning in assignments, emailing etc.)
2. Homework: 3 x 15%
3. Final “Project”: 35%

Homework and Final Project due by **Wednesday, Jan 25, 2022 at 11:59pm EST**.

If you turn homework in earlier this can allow us to potentially give you feedback earlier.

Note: Only people taking the course for credit must turn in the assignments. However, we will evaluate all submitted assignments in case others would like feedback on their work.

Your Setup

If you can, we suggest working virtually with a **large monitor or two screens**. This setup allows you to follow along on Zoom while also doing the hands-on coding.

Installing R

- Install the latest version from: http://cran.r-project.org/
- Install RStudio

RStudio is an **integrated development environment** (IDE) that makes it easier to work with R.

More on that soon!

Getting files from downloads

This course will involve moving files around on your computer and downloading files.

If you are new to this - check out these videos.

If you have a PC: https://youtu.be/we6vwB7DsNU

If you have a Mac: https://www.youtube.com/watch?v=Ao9e0cDzMrE

Basic terms

R jargon: https://link.springer.com/content/pdf/bbm%3A978-1-4419-1318-0%2F1.pdf

Package - a package in R is a bundle or "package" of code (and or possibly data) that can be loaded together for easy repeated use or for **sharing** with others.

Packages are sort of analogous to a software application like Microsoft Word on your computer. Your operating system allows you to use it, just like having R installed (and other required packages) allows you to use packages.

Basic terms

Function - a function is a particular piece of code that allows you to do something in R. You can write your own, use functions that come directly from installing R, or use functions from additional packages.

A function might help you add numbers together, create a plot, or organize your data. More on that soon!

```
sum(1, 20234)
```

```
[1] 20235
```

Basic terms

Argument - what you pass to a function

- can be data like the number 1 or 20234

```
sum(1, 20234)
```

```
[1] 20235
```

- can be options about how you want the function to work such as `digits`

```
round(0.627, digits = 2)
```

```
[1] 0.63
```

```
round(0.627, digits = 1)
```

```
[1] 0.6
```

Basic terms

Object - an object is something that can be worked with in R - can be lots of different things!

- a matrix of numbers
- a plot
- a function

... many more

Variable and Sample

- **Variable**: something measured or counted that is a characteristic about a sample

examples: temperature, length, count, color, category

- **Sample**: individuals that you have data about -

examples: people, houses, viruses etc.

```
head(iris)
```

```
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
```

Columns and Rows

[source]

Sample = Row
Variable = Column

Data objects that looks like this is often called a **data frame**.

Fancier versions from the tidyverse are called **tibbles** (more on that soon!).

More on Functions and Packages

- When you download R, it has a “base” set of functions/packages (**base R**)
  - You can install additional packages for your uses from CRAN or GitHub
  - These additional packages are written by RStudio or R users/developers (like us)

Using Packages

- Not all packages available on CRAN or GitHub are trustworthy
- RStudio (the company) makes a lot of great packages
- Who wrote it? **Hadley Wickham** is a major authority on R (Employee and Developer at RStudio)
- How to trust an R package

(source: https://fosstodon.org/@hadleywickham)

Tidyverse and Base R

We will mostly show you how to use tidyverse packages and functions.

This is a newer set of packages designed for data science that can make your code more **intuitive** as compared to the original older Base R.

Tidyverse advantages:
- **consistent structure** - making it easier to learn how to use different packages
- particularly good for **wrangling** (manipulating, cleaning, joining) data
- more flexible for **visualizing** data

Packages for the tidyverse are managed by a team of respected data scientists at RStudio.

See this article for more info.

Collection of R packages

We have an R package called jhur that will make sure all the packages are installed.

You can just copy and paste the code below into your RStudio console - we'll explain what it all means in the next day or two.

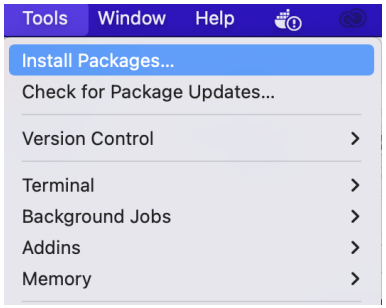
```
install.packages("remotes")
library(remotes)
install_github("jhudsl/jhur")
```

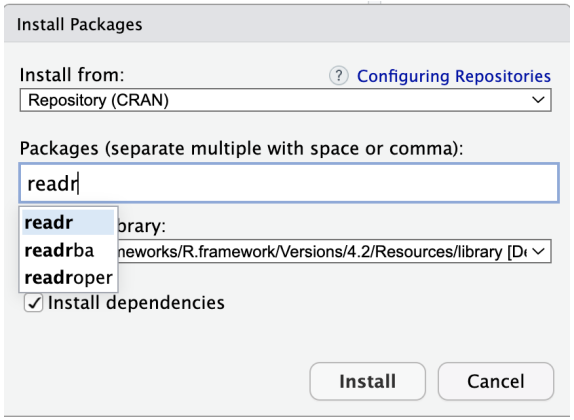
Note it may take ~5-10 minutes to run.

Generally Installing Packages

You can typically install packages (which only needs to be done once) with each version installation of R using:

tools > Install Packages





Loading packages

After installing packages, each time you use them you will need to “load” them into memory - so that you can actively use them.

This is typically done using a function called `library` to load the package.

Here we “load” the `dplyr` package:

```
library(dplyr)
```

We will cover this many times so just worry about the term “load” and the function `library` for now

Useful (+ mostly Free) Resources

Found on our website under the `Resources` tab:
https://jhudatascience.org/intro_to_r/resources.html

- videos from previous offerings of the class
- cheatsheets from the class

Useful (+ mostly Free) Resources

Want more?

- Tidyverse Skills for Data Science Book: https://jhudatascience.org/tidyversecourse/ (more about the tidyverse, some modeling, and machine learning)
- Tidyverse Skills for Data Science Course: https://www.coursera.org/specializations/tidyverse-data-science-r (same content with quizzes, can get certificate with $)
- R for Data Science: http://r4ds.had.co.nz/ (great general information)
- R basics by Rafael A. Irizarry: https://rafalab.github.io/dsbook/r-basics.html (great general information)
- Open Case Studies: https://www.opencasestudies.org/ (resource for specific public health cases with statistical implementation and interpretation)
- Dataquest: https://www.dataquest.io/ (general interactive resource)

Useful (+ mostly Free) Resources

Need help?

- Various “Cheat Sheets”: https://www.rstudio.com/resources/cheatsheets/
- R reference card: http://cran.r-project.org/doc/contrib/Short-refcard.pdf
- R jargon: https://link.springer.com/content/pdf/bbm%3A978-1-4419-1318-0%2F1.pdf
- R vs Stata: https://link.springer.com/content/pdf/bbm%3A978-1-4419-1318-0%2F1.pdf
- R terminology: https://cran.r-project.org/doc/manuals/r-release/R-lang.pdf

Useful (+ mostly Free) Resources

Interested in Reproducibility?

Check out Candace's courses:

- Introduction: https://jhudatascience.org/Reproducibility_in_Cancer_Informatics/
- Advanced: https://jhudatascience.org/Adv_Reproducibility_in_Cancer_Informatics/

Summary

- R is a powerful data visualization and analysis software language
- We will focus on **packages** (code shared among people) of the `tidyverse`, which helps make R more intuitive.
- We will also talk a bit about **base R** because some resources online and R users will use this.
- **Functions** perform specific tasks in R and are found within packages.
- **Arguments** within functions specify how a function is to be performed.
- Materials will be updated frequently as we improve it.
- Class **surveys** are available on CoursePlus so you can provide feedback!
- Lots of **resources** can be found on the website.

Class Website

Website tour!

Class Website

Image by Gerd Altmann from Pixabay