Getting started with R

Matt Lee (mlee8@g.harvard.edu), PHS Launch 2022

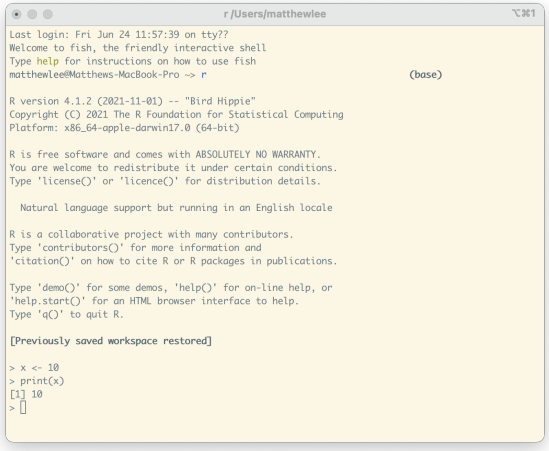
What is R?

- R is an open-source **interpreted** programming language: when you install R, you install an *interpreter* that translates your R code into computer code (sometimes called "machine" code), which is what actually gets run
- This is in contrast to **compiled** languages (e.g. C or C++), where the programmer writes code that is directly converted into machine code
- Several advantages of interpreted languages: much more user friendly, easily read, consistent across operating systems
- Some disadvantages: often slower, and less control over system hardware*

Action step: Install R (https://cloud.r-project.org/)

R Studio: Integrated Development Environments (IDEs)

- We can interact with R directly via a command line (e.g. Terminal)



- But this is not very pretty or reproducible! A population alternative is to use an IDE, such as R Studio, which is a program that adds a whole lot of convenience to writing and running R code
- IDEs are not the language themselves, they provide a way to interact with the language installed on your computer in a friendly way

Action step: Install R Studio
(https://www.rstudio.com/products/rstudio/download/)

Working in R Studio

Key R Studio panes:

- Console: Runs R code, either interactively or via an R script
- Terminal: Convenient terminal application (primarily useful for version control programs like git/GitHub)
- Environment: Objects you've saved to your **working R environment**
- Files: File navigator, useful if you need to figure out where data/R scripts are located
- Plots: Plots generated will populate in this pane – you can also export plots you create using the "Export" button

R Scripting

Use of an **R script** helps keep your code neat and reproducible

- In R Studio: File → New File → R Script
- This is simply a text file with the extension ".R" that will hold all of the R commands we want to run
- Similar to other languages, "#" is reserved for comments

There are 5 main **data types** in R:

- Character (e.g. `"hello world"`)
- Numeric (e.g. `3.14159265`)
- Integer (e.g. `5L`)
- Logical (`TRUE`, `FALSE`)
- Complex (`1i`)

R Scripting

R also has various `data structures`, but the main ones are:

- Vectors: a collection of elements of one data type
- Lists: a collection of objects of arbitrary types (e.g. the first element could be a vector, the second element could be a matrix, the third element could be a data frame)
- Matrices: a vector (so much be one data type only), with dimensions defined
- Data frames: structure that most resembles a data set, each variable is a single data type
- Factors: a numeric vector that has a label attribute

Troubleshooting/Q&A