

Lewis E Tomalin, PhD
Assistant Prof
Department of Population Health Science and Policy
Icahn School of Medicine at Mount Sinai



- If you have never learned programming before, DO NOT WORRY!!!
- The lecture will introduce you to the very basics of R programming and is thus perfect for begginers.
- Knowing these basics will allow you to apply many of the genomics tools and techniques covered later in the course.
- Once you understand the basics of R programming you will be able to teach yourself to any techniques you may need for your future research.

Lecture Overview

- This lecture will be split into 3 separate videos
- Video l (this video)
 - Will provide a very brief overview of R, including installation instructions.

■ Video 2

■ Live demo, showing you how to read, write and manipulate data using R.

■Video 3

Live demo, showing you how to use functions in R, as well as some basic plots.

R Language

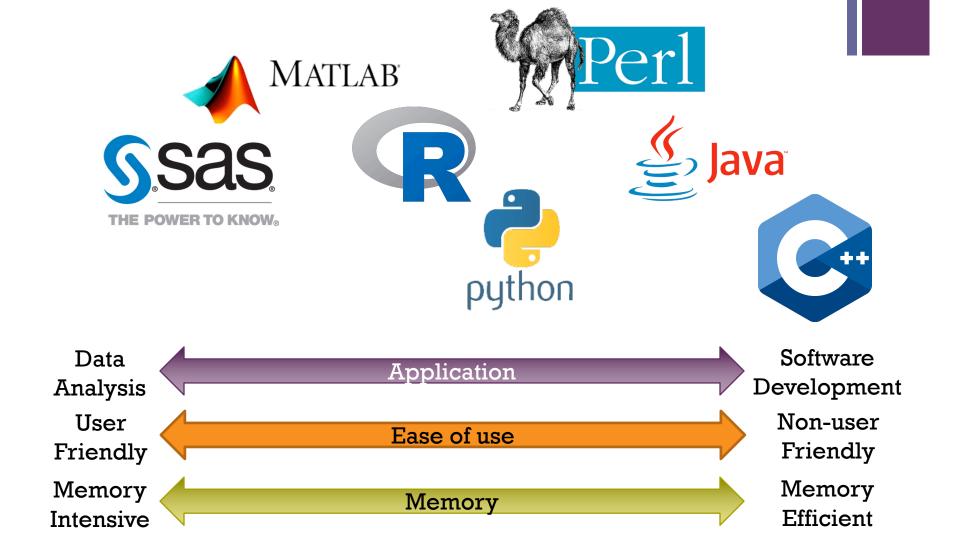
- R is a powerful language and environment for statistical computing and graphics that uses command-line scripting.
- R was initially written by Robert Gentleman and Ross Ihaka 1993 and version 1.0.0 came out in 2000.
- R is free and open-source (researchers like you and me can write out own R functions and make them available online)
- Thus, R includes virtually every type of data manipulation, statistical tool and plot type.



Facebook's Social Network Graph Created using R!!!



Other Languages



+

Installing R and RStudio

+

Install R

- https://www.r-project.org/
- 2. Click download CRAN in the left bar
- 3. Choose a download site (pick closest location)
- 4. Choose your operation system
- 5. Select R version
- 6. Choose Download R v. (latest version)
- 7. Complete installation process
- 8. If you don't have admin rights for your computer, it is also possible to run R and RStudio from a USB stick instead of installing them

RStudio



- RStudio is a powerful and productive user interface for R
- It's free and open-source
- Works great on Windows, Mac, and Linux



Install RStudio

- 1. https://www.rstudio.com/products/rstudio-desktop.
- 2. Download RStudio Desktop
- 3. Click on the Installer link for your platform (Mac/Windows/Linux)
- 4. Complete installation process
- 5. Open RStudio (no need to separately launch R)

Project



Rstudio Layout

~/Desktop/Projects/Sue/Light Experiments/Sue_Light_R - RStudio Sue_Light_R Environment History Analysis.Light.R * Punctions.Light.R * ☐ Import Dataset ▼ Run 🕪 🕞 Source 🔻 List
 ▼ 1 - # 1. import ----Global Environment ▼ 2 - import.me<-function(file.names){ Data # saving date of the measurnment and temperatures O Experiment... 2016 obs. of 25 variables db<-read.csv(file.names, header=F, stringsAsFactors=F) %>% 2016 obs. of 8 variables mutate (Measure.Date=as.Date(substr(V4[3],1,10),format="%m/%d/%Y"), X Inside.Temp.Start=as.numeric(substr(V5[3],1,4)), History/Envi Inside.Temp.End=as.numeric(substr(V6[3],1.4)) Ambient.Temp.Start=as.numeric(substr(V7) ① Y **Editor** 9 Ambient.Temp.End=as.numeric(substr(V8[3 O z ronment 10 batch <- regmatches (file.names, gregexpr("[0-9]+" ply('[',2) %>% unlist() 11 colnames(db)[1:20]<-gsub(":",".",subset(db, V1== 12 # removing first 5 rows that dont have means data chr [1:4] "794" "797" db<-subset(db, (!Well%in%c("Plate information ","Plate", "No background information available.", 13 chr [1:2] "Batch" "Well" 14 "Well", "Plate information ", "1")&(!MeasD.Result==""&!MeasG.Result=="")), "MeasC.Result" 15 select=(!colnames(db)=="NA")) o.alpha 0.05 16 db<-mutate(db,Batch=batch) outlier.ro... chr [1:284] "6" "8" "19" "28... 17 return(db) 18 Files Plots Packages Help Viewer 19 20 - # 2. Modified Thompson tau technique ----R Script \$ Console ~/Desktop/Projects/Sue/Light Experiments/Sue_Light_R/ A Type 'license()' or 'licence()' for distribution details. Files/Plots/ Natural language support but running in an English locale Packages/ R is a collaborative project with many contributors. Console Type 'contributors()' for more information and 'citation()' on how to cite R or R packages in public Help Type 'demo()' for some demos, 'help()' for on-line help, or 'help.start()' for an HTML browser interface to help. Type 'q()' to quit R. [Workspace loaded from ~/Desktop/Projects/Sue/Light Experiments/Sue Light R/.RData] Microsoft PowerPoint



Functions & Packages

- Functions are are at the core of programming
- They take in 'arguments' (e.g data), and then process these inputs to return results (e.g data, plots etc).
- Example: The 'mean()' function, takes in a series of numbers and outputs their mean.



Functions & Packages

- Functions are organized in so-called packages or libraries
- CRAN Packages can be installed:
 - By point-and-click method: Rstudio->Packages->Install
 - Manually by running the code:

```
install.packages("package name")
```

- GitHub Packages:
 - require(devtools)
 - install_github("project/package name")
- Bioconductor Packages:
 - source("https://bioconductor.org/biocLite.R")
 - 2. biocLite("package name")
- After installing make package available by running either command:

```
library("package name")
    require("package name")
```

+ CRAN

■ The Comprehensive R Archive Network https://cran.r-project.org/

- CRAN is a network of ftp and web servers around the world that store identical, up-to-date, versions of code and documentation for R
- Use CRAN to download versions of R and its packages

Bioconductor

■ Bioconductor provides tools for the analysis and comprehension of high-throughput genomic data

■ Goals:

- Provide widespread access to a broad range of powerful statistical and graphical methods for the analysis of genomic data
- Facilitate the inclusion of biological metadata in the analysis of genomic data, e.g. literature data from PubMed, annotation data from LocusLink/Entrez
- Further scientific understanding by producing highquality documentation and reproducible research
- Train researchers on computational and statistical methods for the analysis of genomic data



+

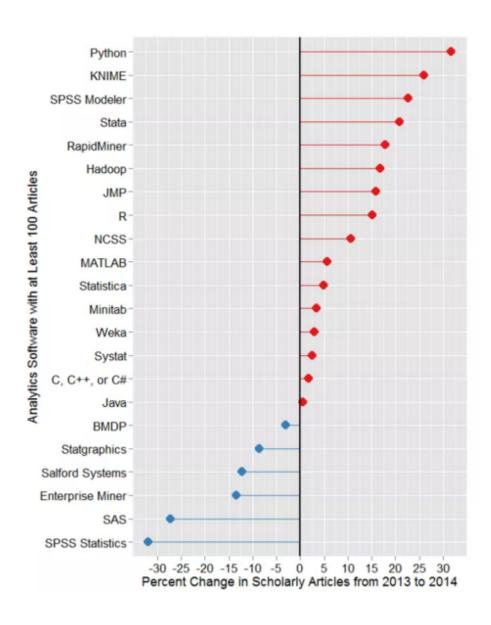
Why R?



IEEE 2017 Top 10 Programming Languages (48 in total)

Language Rank	Types	Spectrum Ranking
1. Python	⊕ 🖵	100.0
2. C		99.7
3. Java	\oplus \Box \Box	99.5
4. C++		97.1
5. C#	\oplus \Box \Box	87.7
6. R	_	87.7
7. JavaScript		85.6
8. PHP	(81.2
9. Go	₩ 〒	75.1
10. Swift		73.7

+ Fast acceptance among researchers



Great Community

- As a thriving open-source project, R is supported by a community of more than 2 million users and thousands of developers worldwide
- LinkedIn Group: "The R Project for Statistical Computing"
 - 67,340 members
- Meetups: "R User Group"
 - 233 groups with > 100k members
- Meetup in NYC: "New York Open Statistical Programming Meetup"
 - 6,348 members

R Help / Learning

R Help

A lot of ways to get help with R:

- Info about function ?function or help(function)
- Google: "R function"
- Most of the packages have extensive support pages and google groups
- Can post questions to get quick responses on:
 - LinkedIn Group "The R Project for Statistical Computing"
 - StackOverflow http://stackoverflow.com/
 - CrossValidated http://stats.stackexchange.com/

Learning R

Many <u>free</u> online resources:

- Coursera John Hopkins Data Science track https://www.coursera.org/specializations/jhu-data-science
- UCLA http://www.ats.ucla.edu/stat/r/
- CodeSchool http://tryr.codeschool.com/
- Programiz http://www.programiz.com/r-programming/