



# **INTRODUCTION TO PUBLIC HEALTH DATA SCIENCE**

## **USING THE R PROGRAMMING LANGUAGE**

Layla Bouzoubaa  
University of Miami

March 06, 2021

# INTRODUCTIONS

- Who am I?
- Who are you?
- Familiarity with R & RStudio
- [Optional Workspace](#)

# WHERE TO GET THE MATERIAL



# WHAT IS DS?

Data **Science** is the practice of using data to try to understand and solve real-world problems.

Coined in 2008 as technologies evolved and data became bigger.\*

A *broad* field. People get into it from all backgrounds and there is an abundance of resources available to get you started or advance.

Data is **everywhere**. Data scientists will always be in demand.

\*Build a Career in Data Science, 2019

# HOW THE UNTRAINED DO SCIENCE

The workflow:

- Collect data in Excel
- Do summary statistics
- Use the import menu to import into an analysis package
- Do analysis with menus
- Fix problems in Excel
- Use the import menu again
- Point and click to more and more analyses
- Copy and paste numbers into Word
- Copy and paste into PowerPoint

*This is antithetical to reproducible research*

# WHY R?

Elegant functional programming language that dominates health research

Base R:

- `c()`
- `df[]`
- `<-`
- `order()`
- `which()`

# BASE R VS. TIDYVERSE

## WHAT IS THE TIDYVERSE?

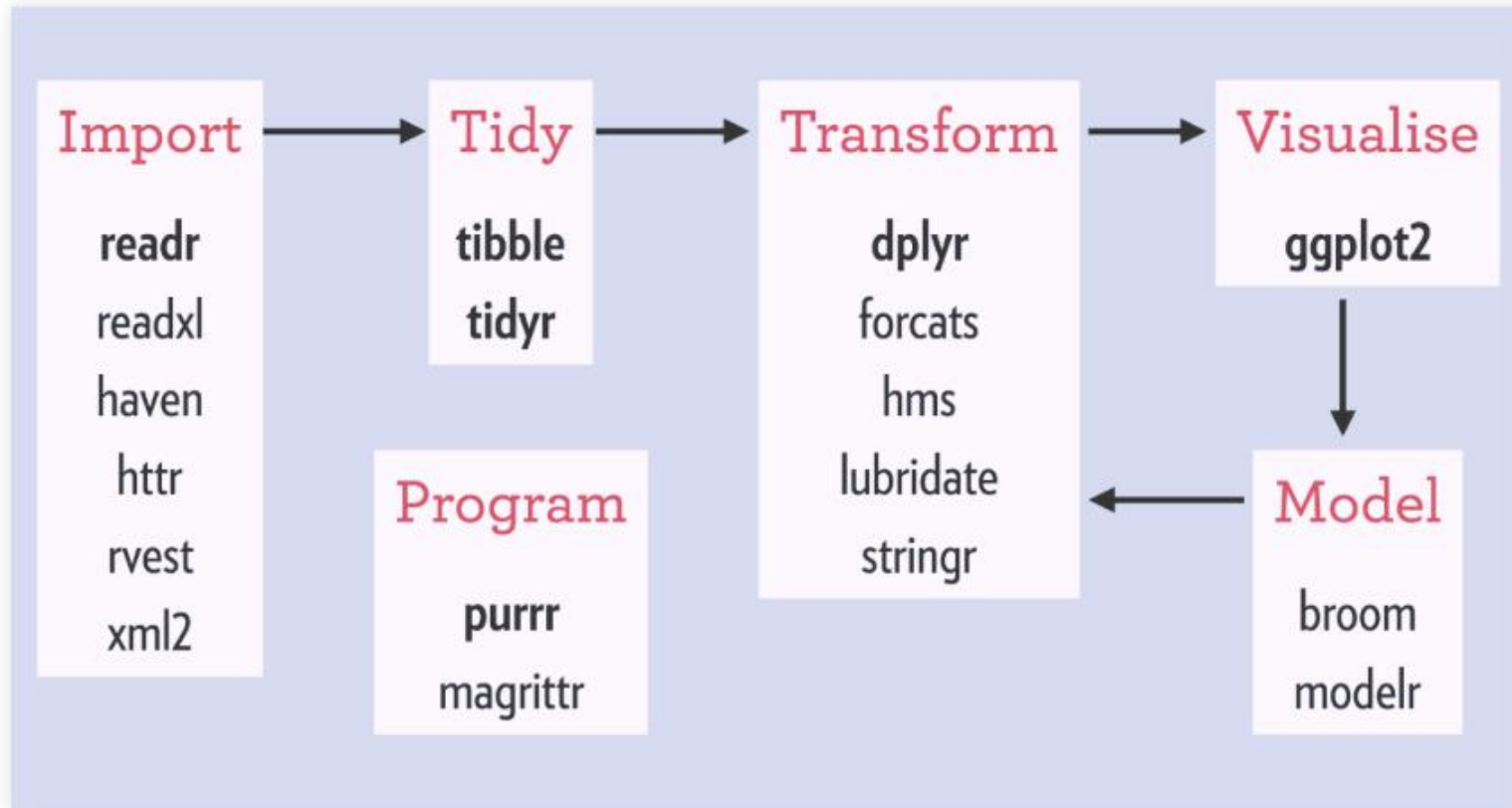
"At a high level, the tidyverse is a language for solving data science challenges with R code. Its primary goal is to facilitate a conversation between a human and a computer about data. Less abstractly, the tidyverse is a collection of R packages that share a high-level design philosophy and low-level grammar and data structures, so that learning one package makes it easier to learn the next."

# PROS:

- designed for functional programming?
  - Functional Programming is an approach to replace iterative (i.e. for) loops. `purrr` package
- consistent functions
- workflow coverage
- a path to data science education
- a parsimonious approach to the development of data science tools
- and the possibility of greater productivity



# DS WITH THE VERSE



(Rickert, 2017)

# THE PIPE

%>%

```
leave_house(get_dressed(get_out_of_bed(wake_up(me, time =  
"8:00"), side = "correct"), pants = TRUE, shirt = TRUE), car  
= TRUE, bike = FALSE)
```

me %>%

```
wake_up(time = "8:00") %>%  
get_out_of_bed(side = "correct") %>%  
get_dressed(pants = TRUE, shirt = TRUE) %>%  
leave_house(car = TRUE, bike = FALSE)
```



Andrew Heiss  
@andrewheiss

Replying to @jocue

I've been teaching tidyverse first for years and students catch on to pipes pretty quick. This slide helps with the intuition (from [evalsp21.classes.andrewheiss.com/projects/01\\_la...](https://evalsp21.classes.andrewheiss.com/projects/01_la...))

2:22 PM · Feb 10, 2021 from Georgia, USA · Twitter for iPhone

205 Retweets · 47 Quote Tweets

1,194 Likes

💬 ↕️ ❤️ 📎 ⚠️ Tip



Sylvain Lapointe @SylL... · Feb 10  
Replying to @andrewheiss and @jocue

For a second, I thought I read lyrics from "A day in the life" from the Beatles.

# TIDYVERSE {dplyr}

Design for humans!! 🦊

Main dplyr verbs:

- filter
- arrange
- select
- mutate
- summarise

*Code Time!* 🦸

# THE UNIX PHILOSOPHY

Rule 4. Fancy algorithms are buggier than simple ones, and they're much harder to implement. Use simple algorithms as well as simple data structures.

Basically...

- Write simple parts
- Being clear is better than being clever
- Design programs to be connected to other programs (modularity)
- When you must fail, fail noisily

# THE UNIX PHILOSOPHY (CONT..)

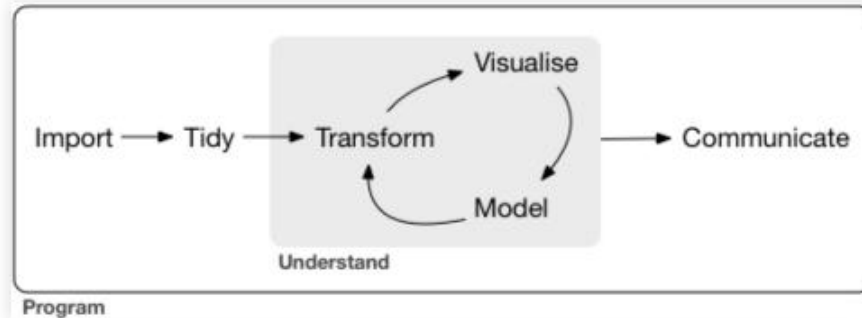
- Tidyverse Specific:
  - File names should be meaningful and end in .R. Avoid using special characters in file names
  - If files should be run in a particular order, prefix them with numbers.
  - Use commented lines of `-` and `=` to break up your file into easily readable chunks.
  - Variable and function names should use only lowercase letters, numbers, and `_`.
    - snake\_case
  - Generally variable names = nouns, function names = verbs

# TIDYVERSE STYLE

An example:

```
1 # Name Vars -----  
2 my_var  
3 var_1  
4 make_names()
```

# TYPICAL DS PIPELINE



Wickham et al, 2018

# PLOTTING (CAKE FIRST)

📦 : ggplot2 📊

📖 : <https://ggplot2.tidyverse.org/>

- Based on [Grammar of Graphics](#)
- Components of the ggplot are combined with the + operator





# PLOTTING (CAKE FIRST, CONT..)

Some Terminology:

- *Geoms* are the geometric objects that are drawn to represent the data, such as bars, lines, and points
- Aesthetic attributes, or *aesthetics*, are visual properties of geoms, such as x and y position, line color, point shapes, etc
- There are *mappings* from data values to aesthetics
- *Scales* control the mapping from the values in the data space to values in the aesthetic space. A continuous y scale maps larger numerical values to vertically higher positions in space

# 1. IMPORT DATA

Take data stored in a file, database, or web application programming interface (API), and load it into a data frame in R.

## Some useful packages:


 : readr

 : <https://readr.tidyverse.org/>

 : Rectangular data (.csv, .tsv, etc)

 : readxl

 : <https://readxl.tidyverse.org/>

 Excel files (.xls, .xlsx, etc)

 : haven

 : <https://haven.tidyverse.org/>

 Files from other statistical software (SAS, SPSS, STATA etc)

# 1.1 OUR DATA

## ***HYPOTHESIS***

Places within Miami-Dade County with higher income have  
lower percentages of food stamp recipients

## ***OUR DATA***

ACS Supplemental Nutrition Assistance Program (SNAP)  
benefits 2019 5-year estimates

## ***SOURCE***

American Community Survey (ACS) data from the U.S. Census  
Bureau - TableID: S2201

## ***GEOGRAPHY***

Census tracts in Miami-Dade County

# 1.2 {tidycensus}

What is {tidycensus}?

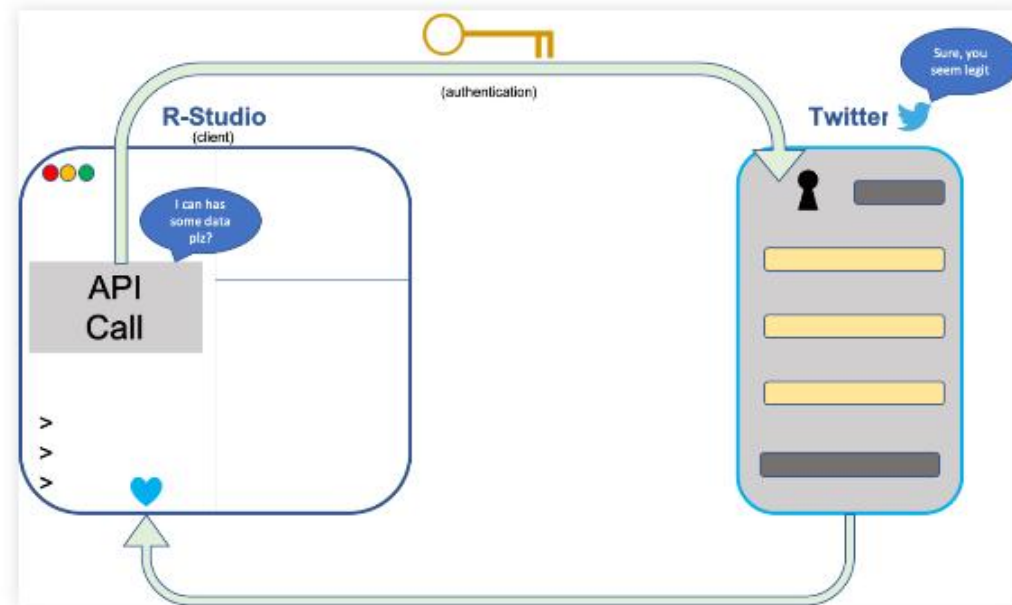
*an R package that allows users to interface with the US Census Bureau's decennial Census and five-year American Community APIs and return tidyverse-ready data frame.*

## **A**pplication **P**rogram **I**nterface

allows a user to programmatically pull data from a source given  
that source provides one  
*ex. The NYT, Twitter, Facebook, Google, US Census*

## 1.2.1 HOW DO APIS WORK?

Think of it like this, just like a Graphical User Interface (GUI) allows you to interact with your code, an API lets your code interact with other code



**DO NOT SHARE/PUBLISH YOUR API KEY!!**

## 2. TIDY

Tidy data:

1. Each variable forms a column.
2. Each observation forms a row.
3. Each value must have its own cell.

country	year	cases	population
Afghanistan	1999	18215	15467071
Afghanistan	2000	18666	20035360
Brazil	1999	31737	172006362
Brazil	2000	80488	174004898
China	1999	211258	1272015272
China	2000	211766	1280428583

variables

country	year	cases	population
Afghanistan	1999	18215	15467071
Afghanistan	2000	18666	20035360
Brazil	1999	31737	172006362
Brazil	2000	80488	174004898
China	1999	211258	1272015272
China	2000	211766	1280428583

observations

country	year	cases	population
Afghanistan	1999	18215	15467071
Afghanistan	2000	18666	20035360
Brazil	1999	31737	172006362
Brazil	2000	80488	174004898
China	1999	211258	1272015272
China	2000	211766	1280428583

values



# 3. TRANSFORM/EDA

What is EDA?

**E**xploratory **D**ata **A**nalysis!

- Iterative cycle to develop questions about your data
- State of mind
- One of the most important steps of an analysis

Scroll down for some terminology and common types of plots  
for EDA :D

## 3.1 TERMINOLOGY

- *Variable* = quantity, quality, or property that you can measure
- *Value* = state of a variable when you measure it
- *Observation* = set of measurements made under similar conditions
- *Variaton* = describes the behavior within a variable
- *Covariation* = the tendency for the values of two or more variables to vary together in a related way

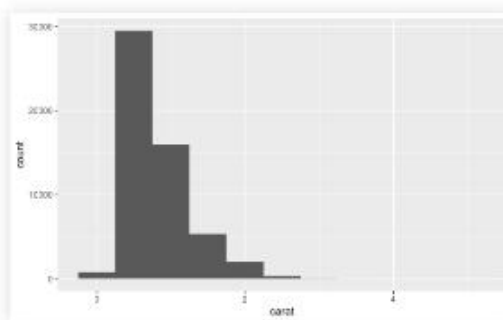


## 3.2 HISTOGRAMS

"A histogram divides the x-axis into equally spaced bins and then uses the height of a bar to display the number of observations that fall in each bin"

Great for examining the distribution of a continuous variable

```
ggplot(diamonds) %>%  
  geom_histogram(mapping = aes(x = carat), binwidth = 0.5)
```



# 3.3 BOX PLOTS

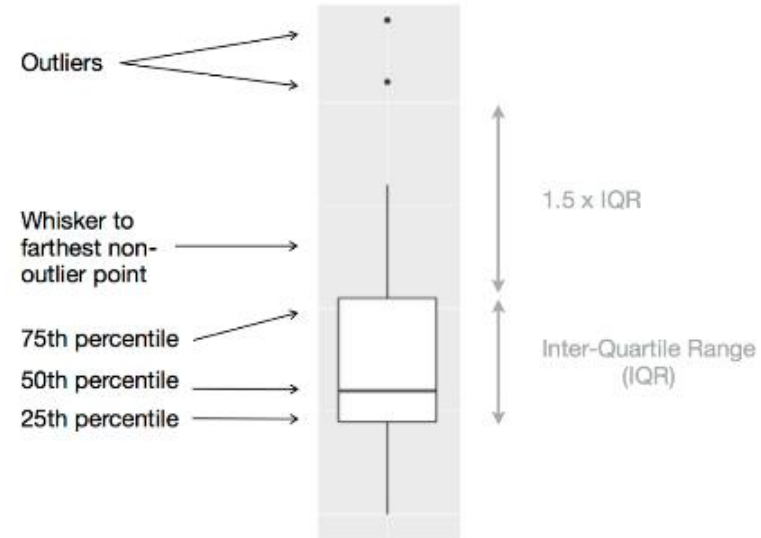
The actual values in a distribution



How a histogram would display the values (rotated)



How a boxplot would display the values



# 3.4 CORRELATION MATRIX

*What is correlation?*

- The correlation metric tells us how much one variable changes with a slight change in another variable.
- A high correlation value between a dependent variable and an independent variable indicates that the independent variable is of very high significance in determining the output



## 4. ANALYSIS/MODEL

Some useful terminology:

- *log transform*: replace each variable  $x$  with a  $\log(x)$ . Doing so usually helps skewed data become less skewed. It can also help make patterns more visible
- *linear regression*: finding the best-fitting straight line through the points. The best-fitting line is called a *regression line*.

```
lm(dependantVar ~ independentVar, data = df)
```

- *p-value*: evidence against a null hypothesis. The smaller the p-value, the stronger the evidence that you should reject the null hypothesis.

# 5. COMMUNICATE



# RESOURCES

[R For Data Science](#) : comprehensive guide to doing data science with R

[Tidyverse Style Guide](#) : how to make sure your code is elegant and redeable for optimal reproducibility

[RStudio Cheatsheets](#) : who doesn't love a cheatsheet?

[Unix Design Principle](#) : general programming best practices

[R Graphics Cookbook](#) : Up your `ggplot2` game with recipes for several types of plots





TWITTER

**@RLadiesMiami**



INSTAGRAM

**@rladiesmiami**



E-MAIL

**miami@r-ladies.org**



GITHUB

**github.com/rladiesmiami**



MEETUP

**meetup.com/rladies-miami**