Statistical Inference via Data Science with R

WNAR

Chester Ismay & Arturo Valdivia

Learning Objectives

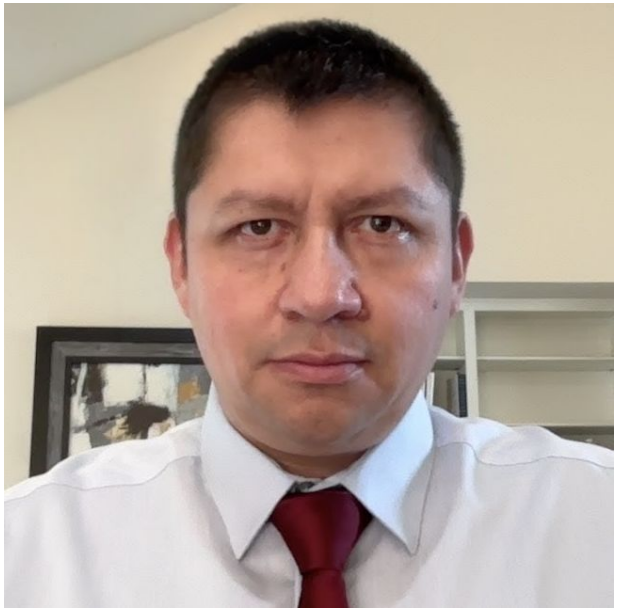
By the end of this workshop, you will be able to

- Perform data wrangling techniques in R via the `tidyverse`
- Develop skills in data visualization with `ggplot2`
- Apply fundamental concepts of statistical inference with `infer`
- Integrate Theory-Based and Simulation-Based Approaches

Instructors' Introduction

Chester Ismay

PORTLAND STATE UNIVERSITY



Arturo Valdivia

INDIANA UNIVERSITY

Agenda

Working with Data in R - Explore, Visualize, Wrangle, Import

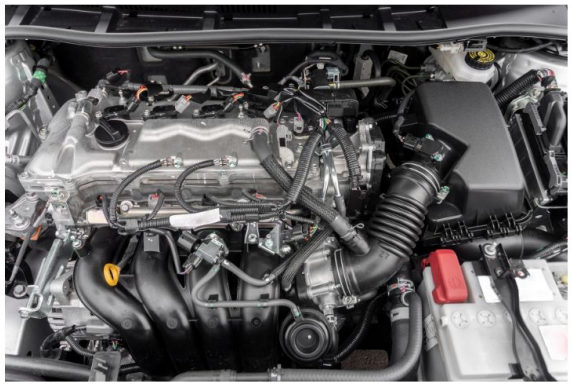
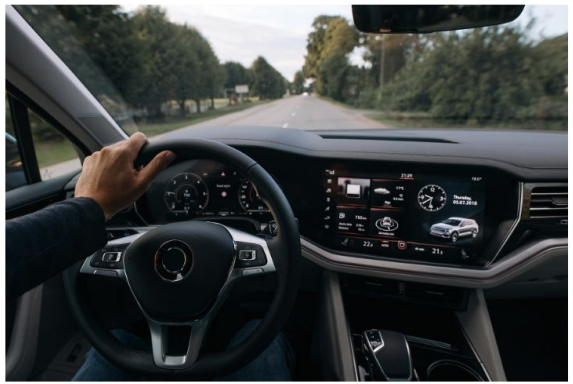
- Part 1: Introduction to R and RStudio
- Part 2: Data Visualization with `ggplot2`
- Part 3: Data Wrangling and Tidy Data
- Part 4: A Preview of Inference using ModernDive

Working with Data in R - Explore, Visualize, Wrangle, Import

Part 1: Introduction to R and RStudio

Introduction to R and RStudio

- R: programming language mainly for statistical computing and data analysis
- RStudio: IDE
- R vs RStudio

R: Engine	RStudio: Dashboard
	

Installing R and RStudio

- Download and install R: https://cloud.r-project.org/
    - Download the appropriate file for your operating system
- Download and install RStudio: https://posit.co/download/rstudio-desktop/
    - Download the appropriate file for your operating system
- Open RStudio

Raise your hand if you need help!

Exploring RStudio

Exploring the RStudio Interface

- In RStudio, you will see three panes: Console, Environment, and Files.
  - The Console is where you can type and run your R code.
  - The Environment pane shows all objects (like datasets) currently in memory.
  - The Files pane helps you navigate files in your project.

Working in RStudio

In RStudio you have the flexibility to work with different types of documents

- Files
    - R Scripts (.R files)
    - **Quarto documents (.qmd files)**
    - R Markdown (.Rmd files)
- Projects
- Shiny Apps
- Many more!

Working in RStudio

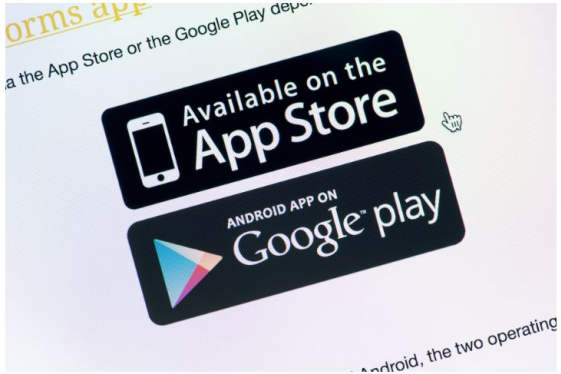
- Download and open in RStudio the following file:
    - https://bit.ly/md-wnar-code

Raise your hand if you need help!

Demo

Installing and Loading R packages

- Extend R's capabilities with additional functions and/or datasets
- First install the package with `install.packages()`
- Load the package using `library()`

R: A new phone	R Packages: Apps you can download
	

Working with data sets

- Loading and viewing a dataset
- Checking the data set structure and data types
- Accessing a single column of a data set
- Checking the first few rows

Loading and Viewing a Dataset

- There are different ways to access a data set
    - Data sets are often part of R packages

Demo & Exercises

Exploring Data in R with RStudio

- Data frames are like tables with rows and columns
- Use `View()` and `glimpse()` to inspect
- The `$` operator extracts columns from data frames
- Identification versus measurement variables/columns

Demo & Exercises

An Introduction to Coding in R

- Commands entered as code in the Console or via scripts.
- Key concepts include objects, vectors, and data types
- Conditional statements and functions help perform tasks
- Learning to code takes frequent practice, but it is one of the most rewarding things you can do!

An Introduction to Coding in R

- Basic Operations in R
- Using Functions in R

Demo & Exercises

Q & A

Working with Data in R - Explore, Visualize, Wrangle, Import

Part 2: Data Visualization using ggplot2

Introduction to Data Visualization

- Raw data typically does not provide much information about the variables in the data set
- Visualizations are very useful to gain most insights
  - They help to identify outliers, distributions, and relationships
- In R, visualizations can be obtained using different functions
- In this workshop, we present visualizations using the `ggplot2` package
  - Based on Grammar of Graphics by Leland Wilkinson

The Grammar of Graphics using ggplot2

- A statistical graphic maps data variables to aesthetic attributes
- Key components:
  - `data`: The dataset
  - `geom`: The geometric objects (points, lines, bars)
  - `aes`: Aesthetic attributes like position, color, shape, size
- Create visualizations by layering these components in `ggplot()`

The Five Named Graphs

- Essential tools for data visualization
- **Scatterplots**, linegraphs, **histograms**, **boxplots**, and barplots
- Each type works best for different data relationships and distributions
- Goal is to uncover trends, patterns, and outliers in data

Histograms

- Display the distribution of a single numerical variable
- Use `geom_histogram()`
- Visualize data spread, center, and frequency of values
- **Tip**: Adjust bin width or number of bins for better data representation

Boxplots

- Summarize numerical data using quartiles and medians
- Use `geom_boxplot()`
- Effective for identifying data spread and detecting outliers
- **Tip**: Use boxplots for comparing distributions across groups

Scatterplots

- Display relationships between two numerical variables
- Using `geom_point()`
- Customizing points (`color`, shape, `size`)
- **Tip**: Handling overplotting
  - `alpha` transparency
  - jittering with `geom_jitter()`

Demo & Exercises

Q & A

Working with Data in R - Explore, Visualize, Wrangle, Import

Part 3: Data Wrangling and Tidy Data

Data Wrangling

- Overview of the `tidyverse`
- Importance of Data Wrangling in Research
- Key Package: `dplyr`

Filter Rows

- Use `filter()` to select rows based on conditions
  - There is also `slice()` which selects rows by position, not condition
- Combine conditions with `&` (AND) and `|` (OR)
- **Tip**: Use `!=` to filter out specific values

Mutate Columns

- Use `mutate()` to create new columns based on existing ones
- Useful for transforming or calculating new values from existing data
- **Tip**: Can also be used to modify an existing column

Summarize Data

- Use `summarize()` to calculate summary statistics
- Reduces data to a single row or value; unlike `mutate()` which keeps original data format
- **Tip**: Can handle missing data with `na.rm = TRUE`

Group By and Summarize

- Use `group_by()` to split data into groups, then apply `summarize()`
- Organizes data into groups; unlike `arrange()`, which only sorts data
- Combine `group_by()` with `summarize()` to create grouped statistics
- **Tip**: `ungroup()` data after grouping if further processing is needed

Arrange Data

- Use `arrange()` to sort rows based on specific columns
- Sort data; unlike `filter()` which selects rows without changing order
- **Tip**: Sort in ascending order by default; use `desc()` for descending

Select Columns

- Use `select()` to choose specific columns
- Different from `mutate()`, which adds new columns
- Can deselect columns using `-` (e.g., `select(-year)`)
- **Tip**: Use helpers like `starts_with()` to select columns by pattern

Pipe Operator (|>)

- Use the pipe operator to chain multiple operations together
- Chains operations unlike using nested functions, which is harder to read
- Often improves workflows
- **Tip**: Think of |> as “then” to improve readability

Demo & Exercises

Q & A

Working with Data in R

Part 4: A Preview of Sampling and Confidence Intervals

Population Data

- Calculate the population mean
- Calculate the population standard deviation
- Visualize the distribution of the population data

Sampling

- Take many samples (1000) of size 50 from the population
- Calculate the sample means
  - Show that the mean of sample means is very close to the population mean
  - Show that the standard error (standard deviation of the sample means) is very close to the population standard deviation divided by the square root of the sample size.
- Create a histogram with the sample means
  - Show how this histogram tends to follow a normal distribution (bell-shaped curve):
- The Central Limit Theorem

Theory-Based Confidence Interval

- Take one sample of size 50 and calculate:
  - The sample mean
  - The sample standard deviation
  - The margin of error
  - The confidence interval
  - Interpret the confidence interval

Simulation-Based Confidence Interval

- The `infer` Framework
- Bootstrapping the Sample
  - Calculate the mean
- Bootstrapping 1000 Samples
  - Get the Mean of Each Bootstrap Sample

Simulation-Based Confidence Interval

- Visualizing the Bootstrap Distribution
  - Create a histogram of the bootstrap means
- Calculate the Bootstrap Confidence Interval
- Interpretation of the Bootstrap Confidence Interval

Demo & Exercises

Q & A