# GOVT 701 Lab Section 1: Math Camp Review and Catch-up

Theodore Landsman

August 26, 2021

# Today

- Math Camp Review:
  - objects in R
  - dataframes in R
- Today's Exercises
  - loading data into R
  - data preprocessing with R
  - data summarization with R

# Math Camp Review: Creating Vectors and Matrices

- As we covered in Math Camp, in R we create vectors using `c()` command.

```
# Creating vectors
a <- c(1, 4, 5, 3, 7)
b <- c(3, 2, 4, 7, 1)
c <- c(8, -2, -4)
program.lang <- c("R", "Python", "C", "Java", "HTML")
# Logical vector
comparison <- (a >= 5)
comparison
```

```
## [1] FALSE FALSE  TRUE FALSE  TRUE
```

# Creating Matrices

- We can create matrices with `matrix()` command

## Usage

```
matrix(data, nrow, ncol, byrow)
```

where

- `data`: vector of matrix elements
- `nrow`, `ncol`: number of rows/columns
- `byrow`: if `TRUE`, the matrix is filled by rows; if `FALSE`, it is filled with columns

# Creating Vectors and Matrices: Example

```r
# Creating matrices
A <- matrix(data = c(1, 4, 3, 5), nrow = 2, byrow = TRUE)
B <- matrix(data = c(1, 4, 3, 5), nrow = 2, byrow = FALSE)
C <- matrix(data = c(9, 7, 6, 2, 1, 3), nrow = 2,
            byrow = TRUE)
D <- matrix(data = c(2, 4, 5, 7, 1, 2), nrow = 3,
            byrow = TRUE)
# Print
A
```

```
##      [,1] [,2]
## [1,]    1    4
## [2,]    3    5
```

```r
B
```

```
##      [,1] [,2]
## [1,]    1    3
## [2,]    4    5
```

# Creating a dataset from mixed numeric and character data

- We can join vectors into a dataframe using the `data.frame()` command.
- Matrices can also be converted into dataframes using the `as.data.frame()` command.

```
fake_dataset <- data.frame(program.lang,a,b, comparison)
fake_dataset
```

```
##   program.lang a b comparison
## 1            R 1 3      FALSE
## 2       Python 4 2      FALSE
## 3            C 5 4       TRUE
## 4         Java 3 7      FALSE
## 5         HTML 7 1       TRUE
```

```
as.data.frame(D)
```

```
##   V1 V2
## 1  2  4
## 2  5  7
## 3  1  2
```

# Datasets We Are Using Today

- Ideology score of U.S. legislators for the 117th Congress
  - `HS117_members.csv`
  - https://voteview.com/data
- Ideology score of countries using United Nations General Assembly votes
  - `IdealpointestimatesAll_Mar2021.tab`
  - https://dataverse.harvard.edu/dataset.xhtml?persistentId=hdl:1902.1/12379

# Package

- A collection of functions, data, and documentations which is publicly shared to enhance the functionality of R.
- Install packages if your R environment does not have them with `install.packages()` command.
  - Your computer must be connected to the Internet
- Call packages you want to use with `library()` or `require()` commands.
`library()`

# Package: Example

```
# Install packages
# install.packages("haven")
# install.packages("readr")

# Load packages
require(haven)
require(readr)
```

# Loading Dataset in R: Working Directory or Project

- It is recommended that you store all the data you use in the **working directory**
- Working directory: the directory (folder) that R refers to in reading and storing information
- To check where the current working directory is, type `getwd()` in the console. To change the working directory, use `setwd()` command.
- Creating a new project or opening an existing project will set your working directory to whatever folder the .Rproj file is in, it will also load any files that were open when you last saved the project into your RStudio console.
- Note that you cannot change your working directory if you are using the project workflow.
- Example

```
setwd("Documents/GOVT_701_Lab/")
```

# Loading Dataset in R

- How to load datasets into R's workspace depends on the file type of the data.
- Examples
  - .csv (comma-separated) files: use `read.csv()` function or `read_csv()` function in `readr` package
  - .dta files (file format for data created with Stata): use `read.dta()` function in `foreign` package or `read_dta()` command in `haven` package
  - .por/.sav files (file format for data created with SPSS): use `read.spss()` function in `foreign` package or `read_spss()` command in `haven` package
  - Excel (.xlsx/.xls) files: use `read_excel()` command in `readxl` package

# Loading Dataset in R: Example

```r
# Read .csv file
voteview <- read_csv("HS117_members.csv")

# Read tab file
UNideal <- read_delim("IdealpointestimatesAll_Mar2021.tab")
```

# How the Data Look Like

- Rows: observations
- Columns: variables

Filter

| | congress | chamber | icpsr | state_icpsr | district_code | state_abbrev | party_code | occupancy | last_means | bioname |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 117 | President | 99913 | 99 | 0 | USA | 100 | 0 | 0 | BIDEN, Joseph Robinette, Jr. |
| 2 | 117 | House | 20301 | 41 | 3 | AL | 200 | NA | NA | ROGERS, Mike Dennis |
| 3 | 117 | House | 21102 | 41 | 7 | AL | 100 | NA | NA | SEWELL, Terri |
| 4 | 117 | House | 21193 | 41 | 5 | AL | 200 | NA | NA | BROOKS, Mo |
| 5 | 117 | House | 21500 | 41 | 6 | AL | 200 | NA | NA | PALMER, Gary James |
| 6 | 117 | House | 22108 | 41 | 1 | AL | 200 | NA | NA | CARL, Jerry L. |
| 7 | 117 | House | 22140 | 41 | 2 | AL | 200 | NA | NA | MOORE, Barry |
| 8 | 117 | House | 29701 | 41 | 4 | AL | 200 | NA | NA | ADERHOLT, Robert |
| 9 | 117 | House | 14066 | 81 | 1 | AK | 200 | NA | NA | YOUNG, Donald Edwin |
| 10 | 117 | House | 20305 | 61 | 3 | AZ | 100 | NA | NA | GRIJALVA, Raúl M. |
| 11 | 117 | House | 20902 | 61 | 2 | AZ | 100 | NA | NA | KIRKPATRICK, Ann |
| 12 | 117 | House | 21103 | 61 | 4 | AZ | 200 | NA | NA | GOSAR, Paul |
| 13 | 117 | House | 21105 | 61 | 6 | AZ | 200 | NA | NA | SCHWEIKERT, David |
| 14 | 117 | House | 21502 | 61 | 7 | AZ | 100 | NA | NA | GALLEGO, Ruben |
| 15 | 117 | House | 21705 | 61 | 5 | AZ | 200 | NA | NA | BIGGS, Andrew S. |
| 16 | 117 | House | 21739 | 61 | 1 | AZ | 100 | NA | NA | O'HALLERAN, Thomas C. |
| 17 | 117 | House | 21757 | 61 | 8 | AZ | 200 | NA | NA | LESKO, Debbie |
| 18 | 117 | House | 21968 | 61 | 9 | AZ | 100 | NA | NA | STANTON, Greg |

Showing 1 to 19 of 540 entries, 22 total columns

Figure 1: Voteview dataset

# data.frame Object

- If we load datasets using commands like `read_csv()`, the corresponding objects will be of the `data.frame` class.

```
# Let's check
class(voteview)
```

```
## [1] "spec_tbl_df" "tbl_df"      "tbl"         "data.frame"
```

- `data.frame` objects are two-dimensional arrays in which column vectors (= variables) are bound together, often of different types.

# Accessing Variables in the Dataset

- How to access variables in a `data.frame` object?
- To call variables within a `data.frame`, we use `$` to write `dfname$varname`.
- Since each variable is a vector, we can access its elements using `[]`
- Example

```
# 2nd - 5th observations of nominate_dim1 variable
voteview$nominate_dim1[c(2:5)]
```

```
## [1]  0.359 -0.392  0.654  0.703
```

# Accessing Variables in the Dataset (cont.)

- To access elements of a variable, we can also specify logical expressions
- Example

```
# Name of House Democrats in Arizona
voteview$bioname[voteview$chamber == "House"
                 & voteview$state_abbrev == "AZ"
                 & voteview$party_code == 100]
```

```
## [1] "GRIJALVA, Raúl M."     "KIRKPATRICK, Ann"      "GALLEGO, Rub
## [4] "O'HALLERAN, Thomas C." "STANTON, Greg"
```

```
# UN ideal points of US 1990 & 2007
UNideal$IdealPointAll[UNideal$ccode == 2
                      & (UNideal$session == 45 | UNideal$session == 62)
```

```
## [1] 3.096880 2.756276
```

# Summarizing Variables

- Examining how the variables are distributed
  - `summary()` for continuous variables
  - `table()` for discrete variables
  - `prop.table()` for tables entries in proportions
- Obtaining summary statistics
  - `mean()`, `median()`, `sd()`, `quantile()`…

# Summarizing Variables: Example

```
# Distribution of UN General Assembly ideal point
summary(UNideal$IdealPointAll)
# Number of countries per each region in 2008
table(UNideal$unsc_region[UNideal$session == 63])
# Crosstab of chamber and party
table(voteview$chamber, voteview$party_code)
```

# Summarizing Variables: Example (cont.)

```r
# Proportion of countries by region in 2008
prop.table(table(UNideal$unsc_region[UNideal$year == 2008]))
```

```
## Warning: Unknown or uninitialised column: `unsc_region`.
```

```
## Warning: Unknown or uninitialised column: `year`.
```

```
## numeric(0)
```

```r
# Party composition by chamber
prop.table(table(voteview$chamber, voteview$party_code),
           margin = 1)
```

```
##
##                    100       200       328
##   House      0.51258581 0.48741419 0.00000000
##   President  1.00000000 0.00000000 0.00000000
##   Senate     0.48039216 0.50000000 0.01960784
```

# Missing Values in R

- In R, we represent missing values with `NA`
- Many functions (e.g., `mean()`) cannot conduct their operations if there are missing values
  - To circumvent the problem, we set the `na.rm` argument to `TRUE`
- Example

```
mean(voteview$nominate_dim1)
mean(voteview$nominate_dim1, na.rm = TRUE)
```

# Exercises!

- Create a small dataset for something in your life, think `family` where each row is a person, their relationship to you, and their age, or `food` where each item is a food item you need to pickup, its price, and what meal you plan to eat it for.
- Compute the mean and the standard deviation of `nominate_dim2` variable to the mean and standard deviation for `nominate_dim2`
- Find the mean and standard deviation of `nominate_dim2` for only House members and for only Senate members.