

R Lab 4: Introduction to Data Analysis with R

Ikuma Ogura

August 23, 2019

Today

- Introduction to
 - ▶ loading data into R
 - ▶ data preprocessing with R
 - ▶ data summarization with R

Datasets We Use Today

- Ideology score of U.S. legislators for the 115th Congress
 - ▶ `HS115_members.csv.csv`
 - ▶ <https://voteview.com/data>
- Ideology score of countries using United Nations General Assembly votes
 - ▶ `IdealpointsPublished.dta`
 - ▶ <https://dataverse.harvard.edu/dataset.xhtml?persistentId=hdl:1902.1/12379>
- Sushi preference survey data
 - ▶ `sushi_preference.csv`
 - ▶ Codebook: `sushi_preference_codebook.pdf`
 - ▶ <http://www.kamishima.net/sushi/>

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.

Package: Example

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

Loading Dataset in R: Working Directory

- It is recommended that you store all the data use 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.
- Example

```
setwd("C:/Users/ikuma/Documents/math-camp-2019/R")
```

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

```
# Read .csv file  
voteview <- read_csv("HS115_members.csv")  
  
# Read .dta file  
UNideal <- read_dta("IdealpointsPublished.dta")
```


How the Data Look Like

- Rows: observations
- Columns: variables

voteview

Filter

	congress	chamber	icpsr	state_icpsr	district_code	state_abbrev	party_code	occupancy	last_means	bioname	bioguide_id	born	died	nominate_dir
1	115	President	99912	99	0	USA	200	0	0	TRUMP, Donald John	NA	1946	NA	NA
2	115	House	20301	41	3	AL	200	NA	NA	ROGERS, Mike Dennis	R000575	1958	NA	0.352
3	115	House	21102	41	7	AL	100	NA	NA	SEWELL, Terri	S001185	1965	NA	-0.390
4	115	House	21192	41	2	AL	200	NA	NA	ROBY, Martha	R000591	1976	NA	0.362
5	115	House	21193	41	5	AL	200	NA	NA	BROOKS, Mo	B001274	1954	NA	0.643
6	115	House	21376	41	1	AL	200	NA	NA	BYRNE, Bradley	B001289	1955	NA	0.605
7	115	House	21500	41	6	AL	200	NA	NA	PALMER, Gary James	P000609	1954	NA	0.718
8	115	House	29701	41	4	AL	200	NA	NA	ADERHOLT, Robert	A000055	1965	NA	0.365
9	115	House	14066	81	1	AK	200	NA	NA	YOUNG, Donald Edwin	Y000033	1933	NA	0.283
10	115	House	20304	61	8	AZ	200	NA	NA	FRANKS, Trent	F000448	1957	NA	0.749
11	115	House	20305	61	3	AZ	100	NA	NA	GRUALVA, Raúl M.	G000551	1948	NA	-0.599
12	115	House	21103	61	4	AZ	200	NA	NA	GOSAR, Paul	G000565	1958	NA	0.668
13	115	House	21105	61	6	AZ	200	NA	NA	SCHWEIKERT, David	S001183	1962	NA	0.595
14	115	House	21300	61	9	AZ	100	NA	NA	SINEMA, Kyrsten	S001191	1976	NA	-0.105
15	115	House	21501	61	2	AZ	200	NA	NA	McSALLY, Martha	M001197	1966	NA	0.346
16	115	House	21502	61	7	AZ	100	NA	NA	GALLEGO, Ruben	G000574	1979	NA	-0.451
17	115	House	21705	61	5	AZ	200	NA	NA	BIGGS, Andrew S.	B001302	1958	NA	0.877
18	115	House	21739	61	1	AZ	100	NA	NA	O'HALLERAN, Thomas C.	O000171	1946	NA	-0.179
19	115	House	21757	61	8	AZ	200	NA	NA	LESKO, Debbie	L000589	1958	NA	0.601
20	115	House	21106	42	1	AR	200	NA	NA	CRAWFORD, Rick	C001067	1966	NA	0.400
21	115	House	21108	42	3	AR	200	NA	NA	WOMACK, Steve	W000809	1957	NA	0.347
22	115	House	21503	42	2	AR	200	NA	NA	HILL, French	H001072	1956	NA	0.455
23	115	House	21563	42	4	AR	200	NA	NA	WESTERMAN, Bruce Eugene	W000821	1967	NA	0.547

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] 0.352 -0.390 0.362 0.643
```

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, Raul M."      "SINEMA, Kyrsten"        "GALLEGRO, Ruben"  
## [4] "O'HALLERAN, Thomas C."
```

```
# UN ideal points of US 1990 & 2007
```

```
UNideal$Idealpoint[UNideal$ccode == 2  
                   & (UNideal$year == 1990 | UNideal$year == 2007)]
```

```
## [1] 2.892100 2.844069
```

Creating New Variables

- It is often the case that the dataset does not contain the exact variables we want to use for analysis and it only includes variables that are closely related.
- In these cases, we need to create a new variable based on the information we have.
- Example
 - ▶ We would like to know the age of US legislators as of January 1, 2015.
 - ▶ We want to create a string variable representing the party affiliation of US legislators

Creating New Variables (cont.)

- We can apply the knowledge learned so far so to do variable recodings!
- Example

```
# Age of US legislators
voteview$age <- 2015 - voteview$born
# String variable on the party affiliation
voteview$party_name <- "Democrat"
voteview$party_name[voteview$party_code == 200] <- "Republican"
voteview$party_name[voteview$party_code == 328] <- "Independent"
```

ifelse() Function

- We can also use `ifelse()` function to do the same thing (and in a somewhat simpler way).
- `ifelse()` command;

Usage

```
ifelse(test, yes, no)
```

where

- ▶ `test` is the logical expression
- ▶ `yes` is the return value for elements in which `test` is `TRUE`
- ▶ `no` is the return value for elements in which `test` is `FALSE`
- ▶ (Therefore `ifelse()` returns a vector of the same length as `test`)

- Example

```
voteview$party_name <-  
  ifelse(voteview$party_code == 100, "Democract",  
         ifelse(voteview$party_code == 200,  
                 "Republican", "Independent"))
```

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$Idealpoint)
```

```
##           Min.      1st Qu.      Median      Mean      3rd Qu.      Max.
## -2.562354 -0.661084 -0.175478 -0.000279  0.808866  3.004224
```

```
# Number of countries per each region in 2008
```

```
table(UNideal$unsc_region[UNideal$year == 2008])
```

```
##
##  1  2  3  4  5
## 53 58 20 28 33
```

```
# Crosstab of chamber and party
```

```
table(voteview$chamber, voteview$party_name)
```

```
##
##           Democrat Independent Republican
## House           200             0           250
## President           0             0             1
## Senate           48             2           55
```

Summarizing Variables: Example (cont.)

```
# Proportion of countries by region in 2008
```

```
prop.table(table(UNideal$unsc_region[UNideal$year == 2008]))
```

```
##
```

```
##           1           2           3           4           5
## 0.2760417 0.3020833 0.1041667 0.1458333 0.1718750
```

```
# Party composition by chamber
```

```
prop.table(table(voteview$chamber, voteview$party_name),
              margin = 1)
```

```
##
```

```
##           Democract Independent Republican
## House      0.44444444  0.00000000 0.55555556
## President  0.00000000  0.00000000 1.00000000
## Senate     0.45714286  0.01904762 0.52380952
```

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)
```

```
## [1] NA
```

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

```
## [1] 0.09835676
```

Excercises!

- Compute the mean and the standard deviation of `nominate_dim1` variable for Democratic Senators and Republican House members, respectively.
- Calculate the differences in ideal points between US and Russia 1946-2015 and summarize the result
- For the sushi preference survey dataset,
 - ▶ read the dataset in R and name the object as `sushi.dat`.
 - ▶ create a variable `pref_same` which takes 1 when respondents live in the same prefectures as they were in 15 years old and 0 otherwise.
 - ▶ compare the distribution of most preferred sushi item (`itemID_1`) among men and women. Do you see any meaningful differences?