Class One Getting Started

Kim Johnson

January 18, 2018

Lecture Outline

- Class Introductions
- Review course outline (open from Github website: https://github.com/kijohnson/Advanced-Data-Analysis)
- Getting started with data analysis

Getting help with R and R errors

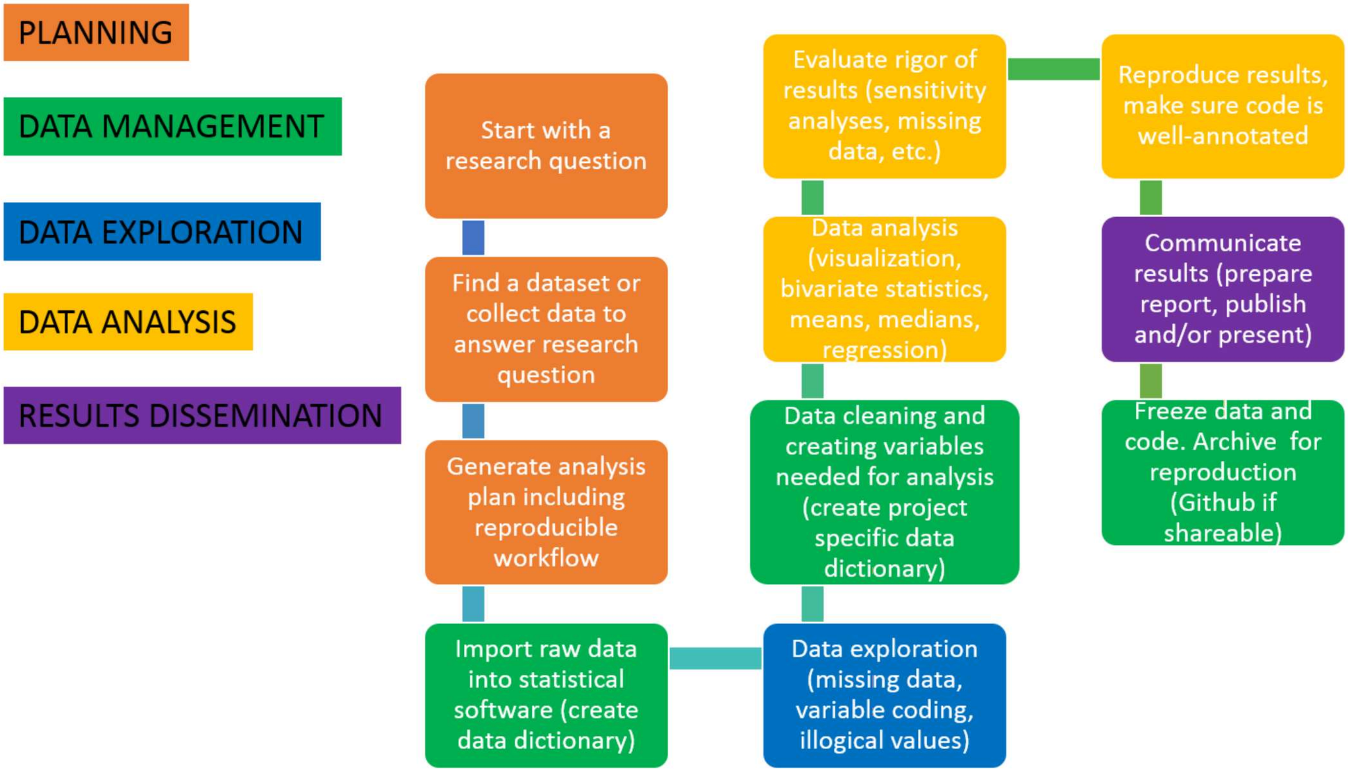
- My advice: Google the problem or error message to try to find a solution.
- Coders are a tribe of people from all over the world who help each other
- From talking to people who code for a living and from my own personal experience, this is a standard problem solving (and learning) approach
- Consult R help often by typing *help(topic)* in the console or by searching for the topic in the help window!

Laptop or Keyboard
Frustration (and weird neck thing?)
Small bits of progress + new determination

Learning objectives

- Understand typical project workflow
- Be able to import data
- Be able to characterize the dataset
- Clean up variables
- Derive one variable
- Calculate simple statistics

Example project workflow



PLANNING

DATA MANAGEMENT

DATA EXPLORATION

DATA ANALYSIS

RESULTS DISSEMINATION

Start with a research question

Find a dataset or collect data to answer research question

Generate analysis plan including reproducible workflow

Import raw data into statistical software (create data dictionary)

Data exploration (missing data, variable coding, illogical values)

Data cleaning and creating variables needed for analysis (create project specific data dictionary)

Data analysis (visualization, bivariate statistics, means, medians, regression)

Evaluate rigor of results (sensitivity analyses, missing data, etc.)

Reproduce results, make sure code is well-annotated

Communicate results (prepare report, publish and/or present)

Freeze data and code. Archive for reproduction (Github if shareable)

A refresher on basic stats tasks in R

- Installing packages and libraries
- Reading in files of different types
- Characterizing the dataset
- Renaming, cleaning, and creating variables
- Simple stats (mean, median, etc.)

Open R studio and let’s install some packages and libraries

```
# install.packages('knitr')#for creating nicer tables
# install.packages('foreign') #for accessing foreign library of functions
# install.packages('haven') #for accessing haven library of functions
# install.packages('readr') #for accessing readr libary of functions

library(foreign)  #for reading spss file
library(haven)  #for reading stata and xpt file
library(readr)  #for reading csv file
library(knitr)  #for creating nicer tables
```

Let’s read in some different file types

- **NOTE about copying links to datsets housed on Github:** On Github click on the file you want to import and if it is readable as is (.csv, .txt), copy and paste the link into your R code for reading the file. If not readable as is (e.g. .sav, .xpt, .dta), in the gray ‘view Raw box’, right click and select ‘open link in new window’ and copy and paste the link address, which should include the following text in the first part: https://raw.githubusercontent.com…

```
starbucks_csv <- read_csv("https://raw.githubusercontent.com/kijohnson/Advanced-Data-Analysis/master/Class%201%20Getting%20Started/Class%20one/s
```

```
## Parsed with column specification:
## cols(
##   Drink = col_character(),
##   Category = col_character(),
##   `_Calories` = col_character(),
##   `_Fat__g_` = col_double(),
##   `_Carb___g_` = col_integer(),
##   `_Fiber__g_` = col_integer(),
##   `_Protein__g_` = col_integer()
## )
```

```
starbucks_stata <- read_dta("https://github.com/kijohnson/Advanced-Data-Analysis/blob/master/Class%201%20Getting%20Started/Class%20one/starbucks

starbucks_tab <- read.delim("https://raw.githubusercontent.com/kijohnson/Advanced-Data-Analysis/master/Class%201%20Getting%20Started/Class%20one
```

read in xpt and spss files

```
starbucks_xpt <- read_xpt("https://github.com/kijohnson/Advanced-Data-Analysis/blob/master/Class%201%20Getting%20Started/Class%20one/starbucks_c
#'The SAS transport format is a open format, as is required for submission of the data to the FDA.'
# (from help page when *??read_xpt* is typed into the console)


starbucks_spss <- read.spss("https://github.com/kijohnson/Advanced-Data-Analysis/blob/master/Class%201%20Getting%20Started/Class%20one/starbucks
    to.data.frame = TRUE)  #read in SPSS file
```

Characterize the datasets (no. of obs, variables, basic summary stats, missing data)

```r
dim(starbucks_xpt)
```

```
[1] 298 7
```

```r
kable(summary(starbucks_xpt))  #creates nice looking table of summary stats for each variable
```

DRINK	CATEGORY	CALORIES	_FAT__G_	CARBS_G	FIBER_G	PROT_G
Length:298	Length:298	Length:298	Min. : 0.000	Min. : 0.00	Min. :0.0000	Min. : 0.000
Class :character	Class :character	Class :character	1st Qu.: 0.000	1st Qu.:15.00	1st Qu.:0.0000	1st Qu.: 0.000
Mode :character	Mode :character	Mode :character	Median : 2.500	Median :32.00	Median :0.0000	Median : 5.000
NA	NA	NA	Mean : 3.369	Mean :30.75	Mean :0.6276	Mean : 5.566
NA	NA	NA	3rd Qu.: 6.000	3rd Qu.:45.00	3rd Qu.:0.0000	3rd Qu.:10.000
NA	NA	NA	Max. :20.000	Max. :71.00	Max. :8.0000	Max. :20.000
NA	NA	NA	NA's :153	NA's :153	NA's :153	NA's :153

Renaming variables

- Point to remember about renaming: always try to use decriptive names rather than x, y, a, b, c.

```
names(starbucks_xpt) <- c("drink", "category", "calories", "fat (g)", "carb. (g)",
    "fiber (g)", "protein (g)")  #renames variables in order of appearance
kable(summary(starbucks_xpt))  #creates 'nice' looking table of summary stats for each variable
```

drink	category	calories	fat (g)	carb. (g)	fiber (g)	protein (g)
Length:298	Length:298	Length:298	Min. : 0.000	Min. : 0.00	Min. :0.0000	Min. : 0.000
Class :character	Class :character	Class :character	1st Qu.: 0.000	1st Qu.:15.00	1st Qu.:0.0000	1st Qu.: 0.000
Mode :character	Mode :character	Mode :character	Median : 2.500	Median :32.00	Median :0.0000	Median : 5.000
NA	NA	NA	Mean : 3.369	Mean :30.75	Mean :0.6276	Mean : 5.566
NA	NA	NA	3rd Qu.: 6.000	3rd Qu.:45.00	3rd Qu.:0.0000	3rd Qu.:10.000
NA	NA	NA	Max. :20.000	Max. :71.00	Max. :8.0000	Max. :20.000
NA	NA	NA	NA's :153	NA's :153	NA's :153	NA's :153

Clean up calories variable/convert to numeric/find mean and median

```
starbucks_xpt$calories_n <- as.numeric(as.character(starbucks_xpt$calories))  #convert calories variable to numeric so
summary(starbucks_xpt$calories_n)  #get summary stats
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##     0.0    70.0   140.0   174.3   280.0   470.0     153
```

```
mean(starbucks_xpt$calories_n, na.rm = TRUE)  #get mean
```

```
## [1] 174.3448
```

```
median(starbucks_xpt$calories_n, na.rm = TRUE)  #get median
```

```
## [1] 140
```

```
sd(starbucks_xpt$calories_n, na.rm = TRUE)  #get sd
```

```
## [1] 118.618
```

```
var(starbucks_xpt$calories_n, na.rm = TRUE)  #get variance
```

```
## [1] 14070.23
```

```
quantile(starbucks_xpt$calories_n, na.rm = TRUE)  #get quantile
```

```
##   0%  25%  50%  75% 100% 
##    0   70  140  280  470
```

Categorize calories as above and below the median, label level values

```
starbucks_xpt$calories_med[starbucks_xpt$calories_n > 140] <- 1  #above median
starbucks_xpt$calories_med[starbucks_xpt$calories_n <= 140] <- 0  #below median
str(starbucks_xpt$calories_med)  #check the type of variable
```

```
##  num [1:298] 0 NA 1 NA 1 NA NA 1 NA 1 ...
```

```
starbucks_xpt$calories_med.f <- factor(starbucks_xpt$calories_med, labels = c("Below the median",
    "Above the median"))  #change to factor variable and label levels
table(starbucks_xpt$calories_med.f)  #determine how many observations are in each level
```

```
## 
## Below the median Above the median 
##               73               72
```

Find mean number of calories for ‘Starbucks Espresso Beverages’

```
espresso <- starbucks_xpt[which(starbucks_xpt$category == "Starbucks Espresso Beverages"),
    ]  #subset espresso data (I am calling this the child dataframe)
table(starbucks_xpt$category)  #check that subsetting worked by checking number of espresso drinks in parent dataframe
```

```
##
##                   Bottled Drinks              Chocolate Beverages
##                               56                               11
##        Cold Brew and Iced Coffee       Fizzio"! Handcrafted Sodas
##                               14                                3
## Frappuccino<U+00AE> Blended Beverages            Freshly Brewed Coffee
##                               58                               15
##                         Iced Tea        Kids\031  Drinks & Others
##                               27                                9
##                  Lattes and Teas                        Smoothies
##                               43                                2
##     Starbucks Espresso Beverages Starbucks Refreshers"! Beverages
##                               48                               12
```

```
dim(espresso)  #check that subsetting worked by checking number of espresso drinks in child dataframe
```

```
## [1] 48 10
```

```
mean(espresso$calories_n, na.rm = TRUE)  #calculate mean number of calories in espresso drinks, removing 'NAs' first
```

```
## [1] 268.125
```

```
summary(espresso$calories_n)  #another way to see the mean number of calories in espresso drinks
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    45.0   207.5   270.0   268.1   312.5   470.0       8
```

Class activity and HW2

- Go to our Github website to download and open the class activity/HW2
- Follow the instructions on the HW2 pdf and let's start exploring the *class1survey* data!

If you receive error messages while installing packages/libraries, see here:

https://stackoverflow.com/questions/32932354/how-to-install-the-libraryreadr