Data Science - Homework 2

Jake Rozran

2022-02-01

library(tidyverse)

# Task 1) Add your name and the appropriate date in the header above.

# Task 2) Enter the PollingReport.com data

PollingReport.com conducted a poll in 1999 in which they asked both men and women the following question: “All things considered, in our society today, do you think there are more advantages in being a man, more advantages, in being a woman, or are there no more advantages in being one than the other?” These results are labeled as man, woman, or none, respectively, in the data below. Those who did not know the answer to the question were labeled as “notknow”.

Of women, 57% said man, 6% said woman, 33% said none, and 4% said notknow. Of men, 41% said man, 14% said woman, 40% said none, and 5% said notknow.

Create three variables, men which contains the four percentages listed above for men, women containing the percentages for women, and response which is a vector of character strings that state what response was given (“man”, “woman”, “none”, and “notknow”). For the percentages, you are welcome to use either proportions or percentages, but do not include the “%” sign if you do the latter.

**For this task and all others, make sure to verify that data are read in**  **properly before moving forward**

women <- c(0.57, 0.06, 0.33, 0.04)  
women

## [1] 0.57 0.06 0.33 0.04

men <- c(0.41, 0.14, 0.4, 0.05)  
men

## [1] 0.41 0.14 0.40 0.05

response <- c("man", "woman", "none", "notknown")  
response

## [1] "man" "woman" "none" "notknown"

# Task 3) Explore the data and create new variables

1. Verify that the percentages in both men and women sum to 1

sum(women) == 1

## [1] TRUE

sum(men) == 1

## [1] TRUE

Does each one Sum to 1? Remove this line and answer the question.

1. Create a logical vector called men\_more of length 4, which is a function of both men and women, which equals TRUE if percentage of men is higher than the percentage of women and FALSE otherwise.

men\_more <- if\_else(men > women, TRUE, FALSE)  
men\_more

## [1] FALSE TRUE TRUE TRUE

1. Combine all four of the variables you created into a data frame called advantage. *(Hint: You could use either cbind() or data.frame())*

advantage <- cbind(response, women, men)  
advantage

## response women men   
## [1,] "man" "0.57" "0.41"  
## [2,] "woman" "0.06" "0.14"  
## [3,] "none" "0.33" "0.4"   
## [4,] "notknown" "0.04" "0.05"

advantage <- data.frame(response, women, men)  
advantage

## response women men  
## 1 man 0.57 0.41  
## 2 woman 0.06 0.14  
## 3 none 0.33 0.40  
## 4 notknown 0.04 0.05

1. Use ifelse (or if\_else) to create a new variable called who\_more that equals “men” if men\_more is TRUE and “women” if men\_more if FALSE. **This variable should be created directly within the advantage data frame.**

advantage$who\_more <- if\_else(men\_more == TRUE, "men", "women")  
advantage

## response women men who\_more  
## 1 man 0.57 0.41 women  
## 2 woman 0.06 0.14 men  
## 3 none 0.33 0.40 men  
## 4 notknown 0.04 0.05 men

# Task 4) Add a new chunk below this question

Explore the gapminder data to discover…

Reminder, to reference a variable within the gapminder dataset, use gapminder$varname where varname is the name of the variable you want to explore.

1. the earliest year (the variable is called year) in the dataset
2. the latest year in the dataset
3. the number of years between the latest and earliest (it’s better to use the functions here rather than just subtract the previous values)
4. the average population size (pop)
5. the average population size (pop) in 1000s (divide by 1000)
6. the median GDP per capita (gdpPercap)
7. whether there are any missing values in the dataset (any variable) *[hint: use the any() command]*
8. the midhinge [the average of the first and third quartile] of GDP per capita *[hint: use the quantile() command]*

library(gapminder)  
data(gapminder) # YOU CAN TECHNICALLY SKIP THIS STEP  
  
# A  
min(gapminder$year)

## [1] 1952

# B  
max(gapminder$year)

## [1] 2007

# C   
max(gapminder$year) - min(gapminder$year)

## [1] 55

# D   
mean(gapminder$pop)

## [1] 29601212

# E  
mean(gapminder$pop) / 1000

## [1] 29601.21

mean(gapminder$pop / 1000)

## [1] 29601.21

# F  
median(gapminder$gdpPercap)

## [1] 3531.847

# G  
any(is.na(gapminder))

## [1] FALSE

# H  
midhinge <- mean(c(quantile(gapminder$gdpPercap, 0.25),  
 quantile(gapminder$gdpPercap, 0.75)))  
midhinge

## [1] 5263.761

# Task 5) Read data from external file

Many cities are publicizing their data as part of an “Open Data” initiative. Philadelphia’s is located at [Open Data Philly](https://www.opendataphilly.org/). Let’s take a look at the cleanliness of neighborhoods around Philadelphia. I downloaded a csv file on Child Blood Lead Levels in Philadelphia from [here](https://www.opendataphilly.org/dataset/philadelphia-child-blood-lead-levels). It can be found in the data section of the website. The “metadata” (information about the variables) can be found [here](http://metadata.phila.gov/#home/datasetdetails/594d26988d68a4593a61bcf0/).

Read the data file into R. Run a str() command to make sure it was read in properly. Verify that there are 46 observations and 5 variables.

data\_url <- paste0("https://phl.carto.com/api/v2/sql?q=SELECT+\*+FROM+child\_bl",  
 "ood\_lead\_levels\_by\_zip&filename=child\_blood\_lead\_levels\_b",  
 "y\_zip&format=csv&skipfields=cartodb\_id,the\_geom,the\_geom\_",  
 "webmercator")  
  
odp <- read\_csv(data\_url)

## Rows: 46 Columns: 5  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## dbl (4): zip\_code, num\_bll\_5plus, num\_screen, perc\_5plus  
## lgl (1): data\_redacted  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

str(odp)

## spec\_tbl\_df [46 × 5] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ zip\_code : num [1:46] 19102 19103 19107 19104 19106 ...  
## $ data\_redacted: logi [1:46] TRUE TRUE TRUE FALSE TRUE FALSE ...  
## $ num\_bll\_5plus: num [1:46] NA NA NA 28 NA 33 NA 8 NA 20 ...  
## $ num\_screen : num [1:46] 51 224 139 805 118 ...  
## $ perc\_5plus : num [1:46] NA NA NA 3.5 NA 3.1 NA 2.1 NA 3.5 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. zip\_code = col\_double(),  
## .. data\_redacted = col\_logical(),  
## .. num\_bll\_5plus = col\_double(),  
## .. num\_screen = col\_double(),  
## .. perc\_5plus = col\_double()  
## .. )  
## - attr(\*, "problems")=<externalptr>

# Task 6) Explore the Lead Level data

1. Verify the following. Unless otherwise stated, feel free to use whatever functions you wish.
2. There are 10 values missing for num\_bll\_5plus and for perc\_5plus.
3. These 10 missing values (see above) are the ones that have data\_redacted equal to TRUE.

sum(is.na(odp$num\_bll\_5plus))

## [1] 10

which(is.na(odp$num\_bll\_5plus))

## [1] 1 2 3 5 7 9 11 13 29 45

sum(is.na(odp$perc\_5plus))

## [1] 10

which(is.na(odp$perc\_5plus))

## [1] 1 2 3 5 7 9 11 13 29 45

which(is.na(odp$num\_bll\_5plus)) == which(is.na(odp$num\_bll\_5plus))

## [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE

odp[odp$data\_redacted == TRUE, ]

## # A tibble: 10 × 5  
## zip\_code data\_redacted num\_bll\_5plus num\_screen perc\_5plus  
## <dbl> <lgl> <dbl> <dbl> <dbl>  
## 1 19102 TRUE NA 51 NA  
## 2 19103 TRUE NA 224 NA  
## 3 19107 TRUE NA 139 NA  
## 4 19106 TRUE NA 118 NA  
## 5 19114 TRUE NA 294 NA  
## 6 19115 TRUE NA 397 NA  
## 7 19116 TRUE NA 330 NA  
## 8 19118 TRUE NA 121 NA  
## 9 19137 TRUE NA 120 NA  
## 10 19153 TRUE NA 276 NA

odp[odp$data\_redacted == "true", ]

## # A tibble: 0 × 5  
## # … with 5 variables: zip\_code <dbl>, data\_redacted <lgl>, num\_bll\_5plus <dbl>,  
## # num\_screen <dbl>, perc\_5plus <dbl>

odp %>%  
 filter(data\_redacted == TRUE)

## # A tibble: 10 × 5  
## zip\_code data\_redacted num\_bll\_5plus num\_screen perc\_5plus  
## <dbl> <lgl> <dbl> <dbl> <dbl>  
## 1 19102 TRUE NA 51 NA  
## 2 19103 TRUE NA 224 NA  
## 3 19107 TRUE NA 139 NA  
## 4 19106 TRUE NA 118 NA  
## 5 19114 TRUE NA 294 NA  
## 6 19115 TRUE NA 397 NA  
## 7 19116 TRUE NA 330 NA  
## 8 19118 TRUE NA 121 NA  
## 9 19137 TRUE NA 120 NA  
## 10 19153 TRUE NA 276 NA

odp %>%   
 filter(data\_redacted == "true")

## # A tibble: 0 × 5  
## # … with 5 variables: zip\_code <dbl>, data\_redacted <lgl>, num\_bll\_5plus <dbl>,  
## # num\_screen <dbl>, perc\_5plus <dbl>

1. Which zip code has the highest percent of kids with a high lead level? Which zip code has the lowest? Use the perc\_5plus variable to determine these.

# Highest  
max(odp$perc\_5plus, na.rm = TRUE)

## [1] 9.2

odp$zip\_code[!is.na(odp$perc\_5plus) &   
 odp$perc\_5plus == max(odp$perc\_5plus, na.rm = TRUE)]

## [1] 19144

odp %>%  
 filter(!is.na(perc\_5plus) &   
 perc\_5plus == max(perc\_5plus, na.rm = TRUE)) %>%   
 select(zip\_code)

## # A tibble: 1 × 1  
## zip\_code  
## <dbl>  
## 1 19144

# LOWEST  
min(odp$perc\_5plus, na.rm = TRUE)

## [1] 0

odp$zip\_code[!is.na(odp$perc\_5plus) &   
 odp$perc\_5plus == min(odp$perc\_5plus, na.rm = TRUE)]

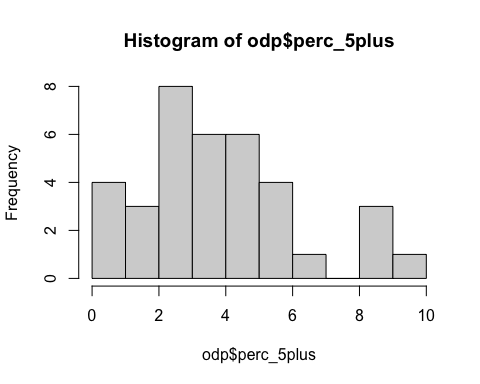
## [1] 19127 19154

odp %>%   
 filter(!is.na(perc\_5plus) &   
 perc\_5plus == min(perc\_5plus, na.rm = TRUE)) %>%   
 select(zip\_code)

## # A tibble: 2 × 1  
## zip\_code  
## <dbl>  
## 1 19127  
## 2 19154

1. Use the hist() function to show the distribution of perc\_5plus. Comment on what you see.

hist(odp$perc\_5plus)



Unimodal, skewed right, possible outliers > 8.