Data Science - Homework 2

Raymond Ogunjimi

1 February 2022

```
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

```
men = c(41, 14, 40, 5)/100
men
```

```
## [1] 0.41 0.14 0.40 0.05
```

```
women = c(57, 6, 33, 4)/100
women
```

```
## [1] 0.57 0.06 0.33 0.04
```

```
response = c("man", "woman", "none", "notknow")
response
```

```
## [1] "man"     "woman"   "none"    "notknow"
```

Task 3) Explore the data and create new variables

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

```
sum(men)
```

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

```
sum(women)
```

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

Does each one Sum to 1? Yes, they both sum to 1.

b. 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 = men > women
men_more
```

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

c. 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(men, women, response, men_more)
advantage
```

```
##      men    women  response  men_more
## [1,] "0.41" "0.57" "man"     "FALSE"
## [2,] "0.14" "0.06" "woman"   "TRUE"
## [3,] "0.4"  "0.33" "none"    "TRUE"
## [4,] "0.05" "0.04" "notknow" "TRUE"
```

d. 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.**

```
library(tidyverse)
who_more = if_else(men_more == TRUE, "men", "women")
who_more
```

```
## [1] "women" "men"   "men"   "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.

a. the earliest year (the variable is called `year`) in the dataset
b. the latest year in the dataset
c. the number of years between the latest and earliest (it`s better to use the functions here rather than just subtract the previous values)
d. the average population size (`pop`)
e. the average population size (`pop`) in 1000s (divide by 1000)
f. the median GDP per capita (`gdpPercap`)
g. whether there are any missing values in the dataset (any variable) *[hint: use the any() command]*
h. the `midhinge` [the average of the first and third quartile] of GDP per capita *[hint: use the quantile() command]*

```
library(gapminder)
min(gapminder$year)
```

```
## [1] 1952
```

```
max(gapminder$year)
```

```
## [1] 2007
```

```
diff(range(gapminder$year))
```

```
## [1] 55
```

```
mean(gapminder$pop)
```

```
## [1] 29601212
```

```
mean(gapminder$pop)/1000
```

```
## [1] 29601.21
```

```
median(gapminder$gdpPercap)
```

```
## [1] 3531.847
```

```
any(is.na(gapminder))
```

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

```
mean(quantile(gapminder$gdpPercap, c(0.25,0.75)))
```

```
## [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.

```
phl_chld_bld_lvls = read.csv(url("https://phl.carto.com/api/v2/sql?q=SELECT+*+FROM+child
_blood_lead_levels_by_zip&filename=child_blood_lead_levels_by_zip&format=csv&skipfields=
cartodb_id,the_geom,the_geom_webmercator"))
phl_chld_bld_lvls
```

```
##    zip_code data_redacted num_bll_5plus num_screen perc_5plus
## 1     19102          true            NA         51         NA
## 2     19103          true            NA        224         NA
## 3     19107          true            NA        139         NA
## 4     19104         false            28        805        3.5
## 5     19106          true            NA        118         NA
## 6     19111         false            33       1071        3.1
## 7     19114          true            NA        294         NA
## 8     19123         false             8        374        2.1
## 9     19115          true            NA        397         NA
## 10    19125         false            20        577        3.5
## 11    19116          true            NA        330         NA
## 12    19126         false            21        302        7.0
## 13    19118          true            NA        121         NA
## 14    19134         false           131       2235        5.9
## 15    19119         false            27        534        5.1
## 16    19135         false            14        698        2.0
## 17    19120         false            95       1940        4.9
## 18    19121         false            68       1181        5.8
## 19    19122         false            11        475        2.3
## 20    19124         false           104       2124        4.9
## 21    19127         false             0         54        0.0
## 22    19128         false            18        631        2.9
## 23    19136         false             6        575        1.0
## 24    19129         false             8        226        3.5
## 25    19130         false            14        503        2.8
## 26    19131         false            36        886        4.1
## 27    19132         false           104       1256        8.3
## 28    19133         false            54       1109        4.9
## 29    19137          true            NA        120         NA
## 30    19138         false            34        739        4.6
## 31    19139         false            43       1272        3.4
## 32    19140         false           173       2026        8.5
## 33    19141         false            62        770        8.1
## 34    19146         false            24       1094        2.2
## 35    19142         false            35       1054        3.3
## 36    19143         false           109       1884        5.8
## 37    19147         false             7        897        0.8
## 38    19144         false           100       1088        9.2
## 39    19145         false            23       1045        2.2
## 40    19150         false             6        308        1.9
## 41    19151         false            27        651        4.1
## 42    19148         false            34       1349        2.5
## 43    19149         false            37       1348        2.7
## 44    19152         false             7        468        1.5
## 45    19153          true            NA        276         NA
## 46    19154         false             0        298        0.0
```

```
str(phl_chld_bld_lvls)
```

```
## 'data.frame':    46 obs. of  5 variables:
##  $ zip_code     : int  19102 19103 19107 19104 19106 19111 19114 19123 19115 19125
...
##  $ data_redacted: chr  "true" "true" "true" "false" ...
##  $ num_bll_5plus: int  NA NA NA 28 NA 33 NA 8 NA 20 ...
##  $ num_screen   : int  51 224 139 805 118 1071 294 374 397 577 ...
##  $ perc_5plus   : num  NA NA NA 3.5 NA 3.1 NA 2.1 NA 3.5 ...
```

Task 6) Explore the Lead Level data

a. Verify the following. Unless otherwise stated, feel free to use whatever functions you wish.

    i. There are 10 values missing for `num_bll_5plus` and for `perc_5plus`.
    ii. These 10 missing values (see above) are the ones that have `data_redacted` equal to `TRUE`.

```
sum(is.na(phl_chld_bld_lvls$num_bll_5plus))
```

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

```
sum(is.na(phl_chld_bld_lvls$perc_5plus))
```

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

```
all(is.na(phl_chld_bld_lvls$num_bll_5plus) == if_else(phl_chld_bld_lvls$data_redacted ==
"true", TRUE, FALSE))
```

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

```
all(is.na(phl_chld_bld_lvls$perc_5plus) == if_else(phl_chld_bld_lvls$data_redacted == "t
rue", TRUE, FALSE))
```

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

b. 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.

```
phl_chld_bld_lvls$zip_code[which.max(phl_chld_bld_lvls$perc_5plus)]
```

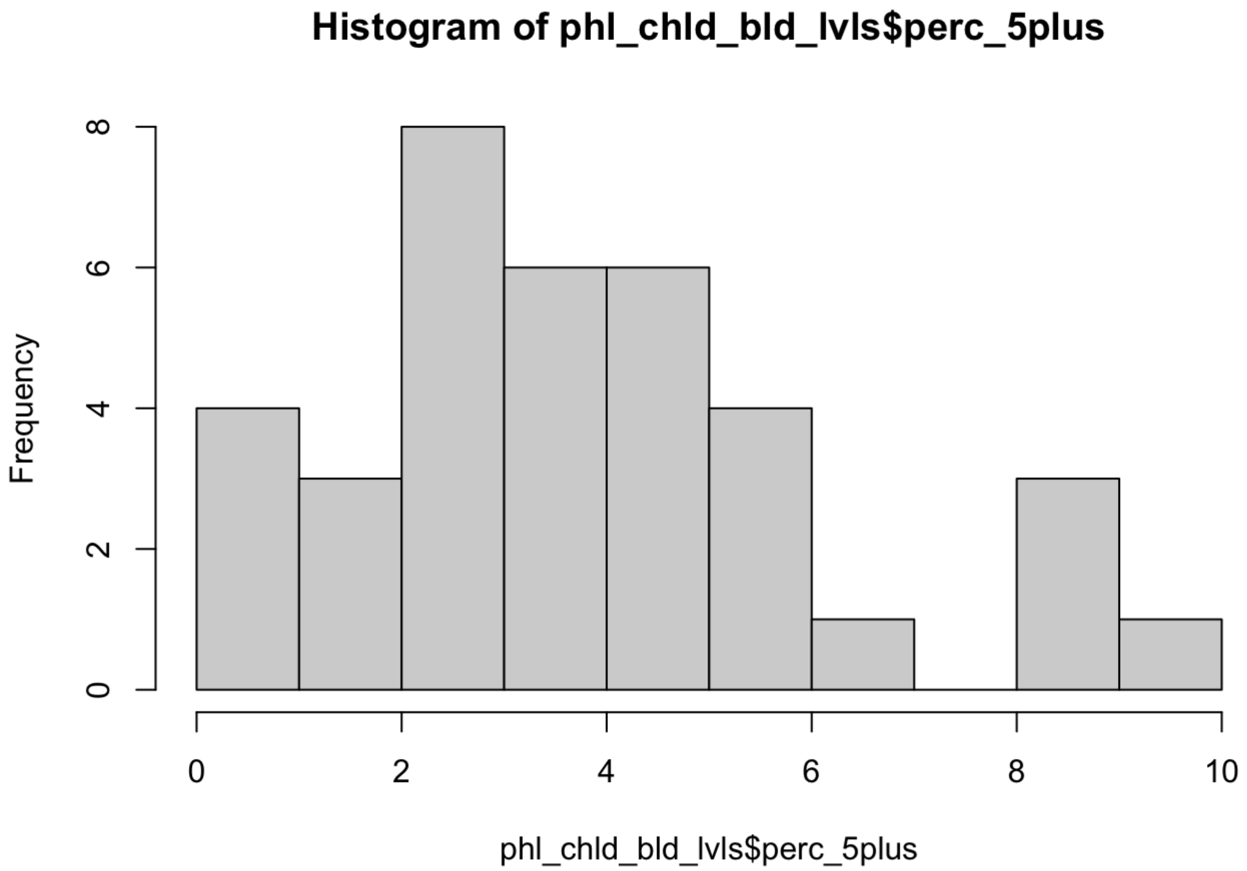
```
## [1] 19144
```

```
phl_chld_bld_lvls$zip_code[which.min(phl_chld_bld_lvls$perc_5plus)]
```

```
## [1] 19127
```

c. Use the `hist()` function to show the distribution of `perc_5plus`. Comment on what you see.

```
hist(phl_chld_bld_lvls$perc_5plus)
```



Based on the metadata and the histogram above it seems that most of the children have a blood lead level between 2 and 5 µg/dL.