Day2

Sigurd Sørensen

2022-08-31

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.1.2

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## Warning: package 'ggplot2' was built under R version 4.1.2

## Warning: package 'tibble' was built under R version 4.1.2

## Warning: package 'tidyr' was built under R version 4.1.2

## Warning: package 'readr' was built under R version 4.1.2

## Warning: package 'purrr' was built under R version 4.1.2

## Warning: package 'dplyr' was built under R version 4.1.2

## Warning: package 'stringr' was built under R version 4.1.2

## Warning: package 'forcats' was built under R version 4.1.2

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Instructions: For this assignment, you need to answer a couple questions with code and then take a screenshot of your working environment. Submit the solutions including the URL to the screenshot in a doc/pdf to Brightspace.

1) Use R to figure out how many elements in the vector below are greater than 2 and then tell me what their sum (of the larger than 2 elements) is. rooms <- c(1, 2, 4, 5, 1, 3, 1, NA, 3, 1, 3, 2, 1, NA, 1, 8, 3, 1, 4, NA, 1, 3, 1, 2, 1, 7, 1, 9, 3, NA)

```
rooms <- c(1, 2, 4, 5, 1, 3, 1, NA, 3, 1, 3, 2, 1, NA, 1, 8, 3, 1, 4, NA, 1, 3, 1, 2, 1, 7, 1, 9, 3, NA]

rooms[rooms >2 & !is.na(rooms)] %>%
  sum()
```

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

2) What type of data is in the 'rooms' vector?

```
class(rooms)
```
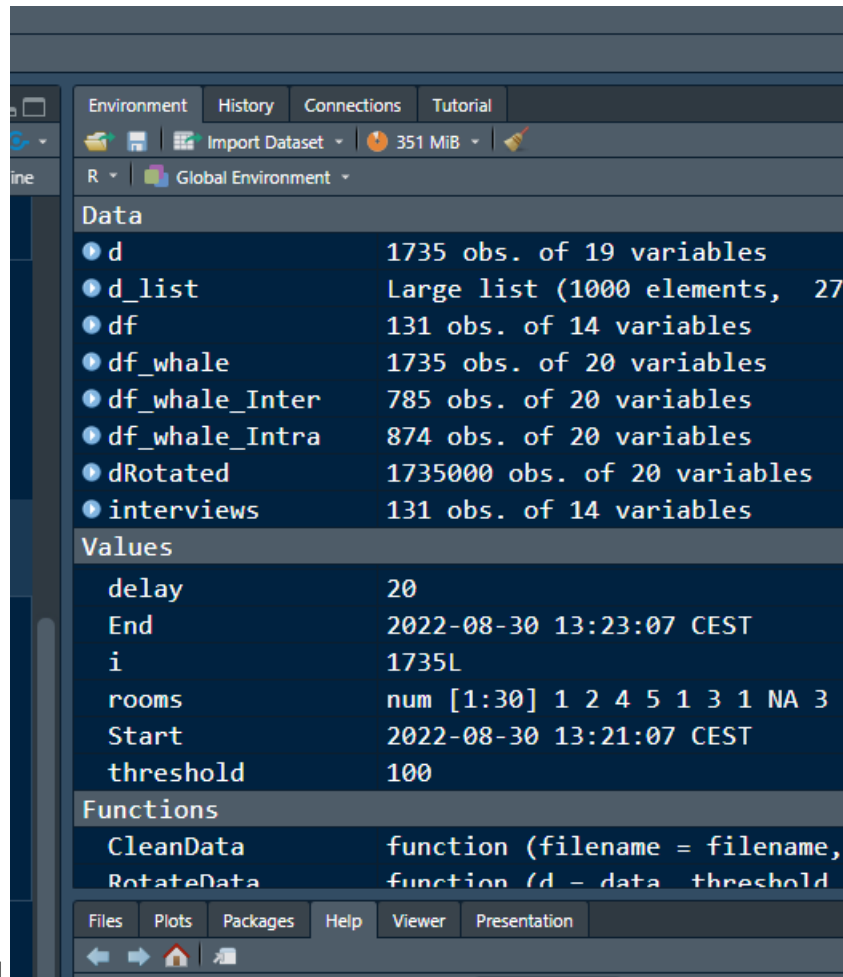
```
## [1] "numeric"
```

More specifically integers.

3) Submit the following image to Github: Inside your R Project (.Rproj), install the 'tidyverse' package and use the download.file() and read_csv() function to read the SAFI_clean.csv dataset into your R project as 'interviews' digital object (see instructions in https://datacarpentry.org/r-socialsci/setup.html and 'Starting with Data' section). Take a screenshot of your RStudio interface showing

a) the line of code you used to create the object,

```
interviews <- read_csv("SAFI_clean.csv")
```

```
## Rows: 131 Columns: 14
## -- Column specification ---------------------------------------------------
## Delimiter: ","
## chr  (7): village, respondent_wall_type, memb_assoc, affect_conflicts, items...
## dbl  (6): key_ID, no_membrs, years_liv, rooms, liv_count, no_meals
## dttm (1): interview_date
##
## 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.
```

Environment | History | Connections | Tutorial

Import Dataset ▾ | 351 MiB ▾

R ▾ | Global Environment ▾

Data

d	1735 obs. of 19 variables
d_list	Large list (1000 elements, 27_
df	131 obs. of 14 variables
df_whale	1735 obs. of 20 variables
df_whale_Inter	785 obs. of 20 variables
df_whale_Intra	874 obs. of 20 variables
dRotated	1735000 obs. of 20 variables
interviews	131 obs. of 14 variables

Values

delay	20
End	2022-08-30 13:23:07 CEST
i	1735L
rooms	num [1:30] 1 2 4 5 1 3 1 NA 3
Start	2022-08-30 13:21:07 CEST
threshold	100

Functions

| CleanData | function (filename = filename, |
| RotateData | function (d = data, threshold |

Files | Plots | Packages | Help | Viewer | Presentation

b) the 'interviews' object in the Environment, and

c) the file structure of your R project in the bottom right "Files" pane. Save the screenshot as an image and put it in your AUID_lastname_firstname repository inside our Github organisation (github.com/Digital-Methods-HASS) or equivalent. Place here the URL leading to the screenshot in

3
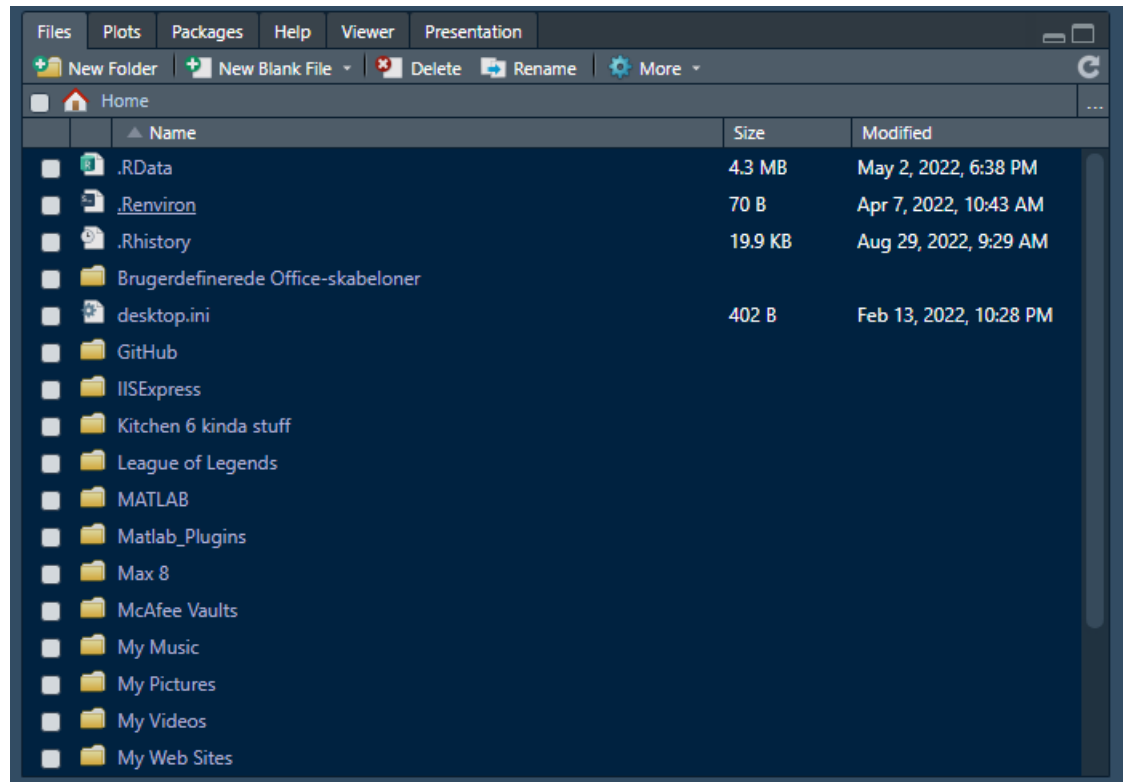
your repository.

4) Challenge: If you managed to create your own Danish king dataset, use it. If not, you the one attached to this assignment (it might need to be cleaned up a bit). Load the dataset into R as a tibble. Calculate the mean() and median() duration of rule over time and find the three monarchs ruling the longest. How many days did they rule (accounting for transition year?)

```
#Easy Way
df_kings <- read_csv2("kings.csv", col_names = TRUE)
```

```
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
```

```
## Rows: 47 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ";"
## chr (2): Kings, Yearasruler
## dbl (2): Start_date, End_date
##
## 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.
```

```
#For fun way
df_kings <- read_csv("kings.csv", col_names = TRUE)
```

```
## Warning: One or more parsing issues, see `problems()` for details
```

```
## Rows: 47 Columns: 1
## -- Column specification --------------------------------------------------------
```

4

```
## Delimiter: ","
## chr (1): Kings;Start_date;End_date;Yearasruler
##
## 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.
```

```r
newnames <- strsplit(colnames(df_kings), split = ";")[[1]]

df_kings <- df_kings %>%
  separate(colnames(df_kings)[1], newnames, sep = ";") %>%
  filter(Yearasruler != "Unknown") %>%
  mutate_at(c("Start_date", "End_date", "Yearasruler"), as.numeric)
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: byte 2 top bits not 0x80'
##  for element 2
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: byte 2 top bits not 0x80'
##  for element 3
```

```
## Warning: Expected 4 pieces. Missing pieces filled with 'NA' in 2 rows [2, 3].
```

```r
summary(df_kings$Yearasruler)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    2.00    7.75   14.00   18.68   29.25   60.00
```

```r
df_kings %>%
  arrange(desc(Yearasruler))
```

```
## # A tibble: 44 x 4
##    Kings                 Start_date End_date Yearasruler
##    <chr>                      <dbl>    <dbl>       <dbl>
##  1 "Christian 4. "             1588     1648          60
##  2 "Erik 7. af Pommern"        1396     1439          43
##  3 "Christian 7. "             1766     1808          42
##  4 "Valdemar 2. Sejr "         1202     1241          39
##  5 "Erik 6. Menved"            1286     1319          35
##  6 "Valdemar 4. Atterdag "     1340     1375          35
##  7 "Chrstian 1."               1448     1481          33
##  8 "Hans "                     1482     1513          31
##  9 "Frederik 4. "              1699     1730          31
## 10 "Frederik 6. "              1808     1839          31
## # ... with 34 more rows
```

```r
sort(df_kings$Yearasruler, TRUE)[1:3]
```

```
## [1] 60 43 42
```