Introduction to R

Task

1. Load the `HappinessAlcoholConsumption.csv` dataset that you can find in the `S2/data/` folder.

```
db <- read.csv("HappinessAlcoholConsumption.csv")
```

2. Look at the structure of the dataset

```
str(db)
```

```
## 'data.frame':    122 obs. of  9 variables:
##  $ Country         : chr  "Denmark" "Switzerland" "Iceland" "Norway" ...
##  $ Region          : chr  "Western Europe" "Western Europe" "Western Europe" "Western Europe" ...
##  $ Hemisphere      : chr  "north" "north" "north" "north" ...
##  $ HappinessScore  : num  7.53 7.51 7.5 7.5 7.41 ...
##  $ HDI             : int  928 943 933 951 918 922 928 915 938 932 ...
##  $ GDP_PerCapita   : num  53.6 79.9 60.5 70.9 43.4 ...
##  $ Beer_PerCapita  : int  224 185 233 169 263 240 251 203 261 152 ...
##  $ Spirit_PerCapita: int  81 100 61 71 133 122 88 79 72 60 ...
##  $ Wine_PerCapita  : int  278 280 78 129 97 100 190 175 212 186 ...
```

So, here we can see that we have four numeric variable, three characters one (that we must convert into factors) and one integer.

Let's convert the character variables as factors:

```
db$Region <- as.factor(db$Region)
db$Country <- as.factor(db$Country)
db$Hemisphere <- as.factor(db$Hemisphere)
```

Another useful function is `summary()`:

```
summary(db)
```

```
##       Country                                  Region   Hemisphere
##  Albania  :  1   Sub-Saharan Africa              :28   both : 5
##  Angola   :  1   Central and Eastern Europe      :27   north:92
##  Argentina:  1   Latin America and Caribbean     :23   noth : 4
##  Armenia  :  1   Western Europe                  :20   south:21
##  Australia:  1   Middle East and Northern Africa:11
##  Austria  :  1   Southeastern Asia               : 5
##  (Other)  :116   (Other)                         : 8
##  HappinessScore       HDI        GDP_PerCapita     Beer_PerCapita
##  Min.   :3.069   Min.   :351.0   Min.   :  1.029   Min.   :  1.00
##  1st Qu.:4.528   1st Qu.:663.8   1st Qu.:  4.134   1st Qu.: 38.25
##  Median :5.542   Median :757.5   Median : 12.016   Median :125.50
##  Mean   :5.525   Mean   :740.9   Mean   : 91.483   Mean   :137.57
##  3rd Qu.:6.477   3rd Qu.:861.5   3rd Qu.: 41.990   3rd Qu.:224.75
##  Max.   :7.526   Max.   :951.0   Max.   :953.000   Max.   :376.00
##
```

```
##  Spirit_PerCapita Wine_PerCapita
##  Min.   :  1.0    Min.   :  1.0
##  1st Qu.: 25.5    1st Qu.:  5.0
##  Median : 82.5    Median : 16.0
##  Mean   : 96.6    Mean   : 66.6
##  3rd Qu.:142.5    3rd Qu.:112.8
##  Max.   :373.0    Max.   :370.0
##
```

Here, we can see some position indices for the numerical variables, and the absolute frequency for each level for the factor (categorical) variables.

3. How many observations? How many variables?

There are many ways to understand the number of observations and variables:

```
dim(db) #dimension of the dataframe (rows: observations, columns: variables)
```

```
## [1] 122   9
```

```
nrow(db) #number of rows (i.e., observations)
```

```
## [1] 122
```

```
ncol(db) #number of columns (i.e., variables)
```

```
## [1] 9
```

4. Create another dataset containing the variables `Country` and `HappinessScore`

```
db_sel <- db[,c("Country", "HappinessScore")]
```

5. Compute the mean of the `HappinessScore` score for each region

One solution:

```
lev <- levels(db$Region)

mean(db$HappinessScore[db$Region == lev[1]])
```

```
## [1] 7.3235
```

```
mean(db$HappinessScore[db$Region == lev[2]])
```

```
## [1] 5.383444
```

```
mean(db$HappinessScore[db$Region == lev[3]])
```

```
## [1] 5.477
```

```
mean(db$HappinessScore[db$Region == lev[4]])
```

```
## [1] 6.061
```

```
mean(db$HappinessScore[db$Region == lev[5]])
```

```
## [1] 5.443727
```

```
mean(db$HappinessScore[db$Region == lev[6]])
```

```
## [1] 7.254
```

```
mean(db$HappinessScore[db$Region == lev[7]])
```

```
## [1] 5.492
```

```
mean(db$HappinessScore[db$Region == lev[8]])
```

```
## [1] 4.151464
```

```
mean(db$HappinessScore[db$Region == lev[9]])
```

```
## [1] 6.7314
```

another one:

```
lev <- levels(db$Region)

for(i in seq(length(lev))){

  mean(db$HappinessScore[db$Region == lev[i]])
}
```

another one:

```
library(tidyverse)
```

```
## Warning: il pacchetto 'tidyverse' è stato creato con R versione 4.3.2
```

```
## Warning: il pacchetto 'ggplot2' è stato creato con R versione 4.3.3
```

```
## Warning: il pacchetto 'stringr' è stato creato con R versione 4.3.3
```

```
## Warning: il pacchetto 'lubridate' è stato creato con R versione 4.3.2
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
db %>%
  group_by(Region) %>%
  summarise(mean_region = mean(HappinessScore))
```

```
## # A tibble: 9 x 2
##   Region                          mean_region
##   <fct>                                 <dbl>
## 1 Australia and New Zealand              7.32
## 2 Central and Eastern Europe             5.38
## 3 Eastern Asia                           5.48
## 4 Latin America and Caribbean            6.06
## 5 Middle East and Northern Africa        5.44
## 6 North America                          7.25
## 7 Southeastern Asia                      5.49
## 8 Sub-Saharan Africa                     4.15
## 9 Western Europe                         6.73
```

6. How many country has a mean below the global mean?

```r
global_mean <- mean(db$HappinessScore)

sum(db$HappinessScore< global_mean)
```

```
## [1] 59
```

7. Create a new variable as the sum of `BeerPerCapita`, `SpiritPerCapita` and `WinePerCapita`

```r
db$new_var <- db$Beer_PerCapita + db$Spirit_PerCapita + db$Wine_PerCapita
```

8. Compute the median of this new variable considering only the north hemisphere.

One solution:

```r
median(db$new_var[db$Hemisphere == "north"])
```

```
## [1] 320.5
```

another one:

```r
db %>%
  filter(Hemisphere == "north") %>%
  summarise(median_new_var = median(new_var))
```

```
##   median_new_var
## 1          320.5
```