# Project № 1

Mikhailichenko Anastasia Sergeevna

2022-11-22

1) Since the observations are divided into several files, I create a function to combine the observations into a single table. The function combines all files of a certain extension from a given folder. The path to the folder and extensions of data files are passed as arguments.

```
making_table_from_files = function (path_files, files_type) {
pat <- stringi::stri_join("*.", files_type)
filenames <- list.files(path_files, pattern=pat, full.names=TRUE)
ldf <- lapply(filenames, read.csv)

total <- do.call(rbind, ldf)
return(total)
}
```

Using the function, we get a table with data about the participants in the Olympic Games

```
total <- making_table_from_files("data", "csv")
str(total)
```

```
## 'data.frame':    271115 obs. of  15 variables:
##  $ ID    : int  1 2 3 4 5 5 5 5 5 5 ...
##  $ Name  : chr  "A Dijiang" "A Lamusi" "Gunnar Nielsen Aaby" "Edgar Lindenau Aabye" ...
##  $ Sex   : chr  "M" "M" "M" "M" ...
##  $ Age   : int  24 23 24 34 21 21 25 25 27 27 ...
##  $ Height: int  180 170 NA NA 185 185 185 185 185 185 ...
##  $ Weight: num  80 60 NA NA 82 82 82 82 82 82 ...
##  $ Team  : chr  "China" "China" "Denmark" "Denmark/Sweden" ...
##  $ NOC   : chr  "CHN" "CHN" "DEN" "DEN" ...
##  $ Games : chr  "1992 Summer" "2012 Summer" "1920 Summer" "1900 Summer" ...
##  $ Year  : int  1992 2012 1920 1900 1988 1988 1992 1992 1994 1994 ...
##  $ Season: chr  "Summer" "Summer" "Summer" "Summer" ...
##  $ City  : chr  "Barcelona" "London" "Antwerpen" "Paris" ...
##  $ Sport : chr  "Basketball" "Judo" "Football" "Tug-Of-War" ...
##  $ Event : chr  "Basketball Men's Basketball" "Judo Men's Extra-Lightweight" "Football Men's Footbal
##  $ Medal : chr  NA NA NA "Gold" ...
```

2) Some cells are empty. Let's put NA in them

```
colSums(is.na(total))
```

```
##     ID   Name    Sex    Age Height Weight   Team    NOC  Games   Year Season
##      0      0      0   9476  60172  62876      0      0      0      7      0
##   City  Sport  Event  Medal
##      0      0      0 231324
```

```
total <- total %>% mutate_all(na_if,"")
colSums(is.na(total))
```

```
##     ID   Name    Sex    Age Height Weight   Team    NOC  Games   Year Season
##      0      1      2   9476  60172  62876      3      4      5      7      7
##   City  Sport  Event  Medal
##      7      7      8 231333
```

Let's look at the values in those columns in which we can guess what values should be.

For example, two "G" values appear in the gender column, although it is expected that there should be only "F" or "M" values. Let's replace strange values with "NA"

```
data.frame(table(total$Sex))
```

```
##   Var1   Freq
## 1    F  74523
## 2    G      2
## 3    M 196588
```

```
total$Sex[total$Sex=="G"] <- NA
data.frame(table(total$Sex))
```

```
##   Var1   Freq
## 1    F  74523
## 2    M 196588
```

There were no surprises in the "Medal" column

```
data.frame(table(total$Medal))
```

```
##     Var1  Freq
## 1 Bronze 13295
## 2   Gold 13372
## 3 Silver 13115
```

Let's analyze the sports column. I guess it's suspicious to see sports that involve too few people. Let's find those sports in which less than five people performed. Among these suspicious sports, one is clearly erroneously tabulated. Let's replace "Footba" with "Football"

```
sport <- data.frame(table(total$Sport))
sport[sport[2]<5][]
```

```
## [1] "Aeronautics"   "Basque Pelota" "Footba"        "Roque"
## [5] "    1"         "    2"         "    1"         "    4"
```

```
total$Sport[total$Sport=="Footba"] <- "Football"
```

Let's do the same as we did with the "Sports" column, but with the "Games" column. Let's find 2 rare variants with a typo (someone didn't finish the word summer) and replace them with the correct one.

```
Games <- data.frame(table(total$Games))
Games[Games[2]<5]
```

```
## [1] "2000 Su"    "2004 Summe" "    1"      "    1"
```

```
total$Games[total$Games=="2000 Su"] <- "2000 Summer"
total$Games[total$Games=="2004 Summe"] <- "2004 Summer"
```

Let's see if there are suspicious values in the age variable. I believe that if the age of the athlete is more than 100, the value can be considered unrealistic. In this dataset, there is just one such value, replace it with NA

```
outA <- total %>% filter(!is.na(Age)) %>% .$Age > 100
sum(outA)
```

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

```
total$Age[total$Age==max(total$Age, na.rm = T)] <- NA
outA <- total %>% filter(!is.na(Age)) %>% .$Age > 100
sum(outA)
```

```
## [1] 0
```

We will do the same with height values greater than 250. Again, one suspicious case was found and we will replace it with NA

```
outH <- total %>% filter(!is.na(Height)) %>% .$Height > 250
sum(outH)
```

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

```
total$Height[total$Height==max(total$Height, na.rm = T)] <- NA
outH <- total %>% filter(!is.na(Height)) %>% .$Height > 250
sum(outH)
```

```
## [1] 0
```

I would consider weight values more than 250 suspicious, but there were no such athletes

```
outW <- total %>% filter(!is.na(Weight)) %>% .$Weight > 250
sum(outW)
```

```
## [1] 0
```

Alternatively, you can see if there are outlayers by plotting graphs (for example, a boxplot), but since values can be judged from everyday experience are presented here, there are few outliers and they are obvious; it was faster to set a threshold and cut off on it

3) Find the age of the youngest athletes of both sexes at the 1992 Olympics.

```
total %>% group_by(Sex) %>% filter(Year == 1992) %>% summarise(min_age = min(Age, na.rm = T))
```

```
## # A tibble: 2 x 2
##   Sex   min_age
##   <chr>   <int>
## 1 F          12
## 2 M          11
```

4) Calculate the mean and standard deviation of the Height variable for athletes of each gender.

```
total %>% group_by(Sex) %>% filter(!is.na(Sex)) %>%
  summarise(mean_hight = mean(Height, na.rm = TRUE), hight_sd = sd(Height, na.rm = TRUE))
```

```
## # A tibble: 2 x 3
##   Sex   mean_hight hight_sd
##   <chr>      <dbl>    <dbl>
## 1 F           168.     8.78
## 2 M           179.     9.36
```

5) Calculate the mean and standard deviation of the Height variable for female tennis players at the 2000 Olympics. Answer rounded to tenths.

```
total %>% filter(!is.na(Sex)&Sex=="F"&Year==2000&Sport=="Tennis") %>%
  summarise(mean_hight = round(mean(Height, na.rm = TRUE), 1),
            hight_sd = round(sd(Height, na.rm = TRUE),1))
```

```
##   mean_hight hight_sd
## 1      171.8      6.5
```

6) The heaviest athlete at the 2006 Olympics competed in the Skeleton

```
total %>% filter(Year==2006) %>%
  select(Weight, Sport) %>%
  arrange(-Weight) %>% .[1, 2]
```

```
## [1] "Skeleton"
```

7) 2249 gold medals were won by women from 1980 to 2010

```
total %>% filter(!is.na(Medal)&!is.na(Sex)&Sex=="F"&Year>=1980&Year<=2010) %>%
  summarise(Gold_medals = sum(Medal=="Gold"))
```

```
##   Gold_medals
## 1        2249
```

8)John Aalberg participated in the Olympic Games 8 times

```
total %>% filter(Name == "John Aalberg") %>% nrow(.)
```

```
## [1] 8
```

He participated in the Olympic Games in 1992 and 1994, that is, 2 times in different years

```
temp <- total %>% filter(Name == "John Aalberg")
unique(temp$Year)
```

```
## [1] 1992 1994
```

```
length(unique(temp$Year))
```

```
## [1] 2
```

9) Age groups and the number of athletes in them

```
age_groups <- total %>% filter(Year==2008&!is.na(Age)) %>%  summarise("[12-25)"= sum(Age<25), "[25-35)"=
age_groups
```

```
##   [12-25) [25-35) [35-45) [45-55) [55, 67]
## 1    6311    6367     790     116       16
```

Age group with the minimum number of athletes

```
colnames(age_groups)[which.min(age_groups[][])]
```

```
## [1] "[55, 67]"
```

Age group with the maximum number of athletes

```
colnames(age_groups)[which.max(age_groups[][])]
```

```
## [1] "[25-35)"
```

10) The number of sports at the 2002 Olympics increased by 3 compared to the 1994 Olympics

```
olimp2002 <- total %>% filter(Year==2002)
olimp1994 <- total %>% filter(Year==1994)
nrow(data.frame(table(olimp2002$Sport))) - nrow(data.frame(table(olimp1994$Sport)))
```

```
## [1] 3
```

11) For the winter and summer Olympiads separately, the top 3 countries for each type of medals are displayed

```
df_for_medals <- total %>% filter(!is.na(Medal)&!is.na(Season))
df_for_medals_freq <- data.frame(table(df_for_medals$Medal, df_for_medals$Season, df_for_medals$NOC))
stats <- df_for_medals_freq %>% group_by(Var1,Var2) %>%
  arrange(Freq)  %>%
  top_n(n = 3) %>% arrange(Var1)

colnames(stats) <- c("Medal_type", "Season", "Country", "Medal_count")
stats %>% filter(Season=="Summer")
```

```
## # A tibble: 9 x 4
## # Groups:   Medal_type, Season [3]
##   Medal_type Season Country Medal_count
##   <fct>      <fct>  <fct>         <int>
## 1 Bronze     Summer GBR             620
## 2 Bronze     Summer GER             649
## 3 Bronze     Summer USA            1197
## 4 Gold       Summer GBR             636
## 5 Gold       Summer URS             832
## 6 Gold       Summer USA            2472
## 7 Silver     Summer URS             635
## 8 Silver     Summer GBR             729
## 9 Silver     Summer USA            1333
```

```
stats %>% filter(Season=="Winter")
```

```
## # A tibble: 9 x 4
## # Groups:   Medal_type, Season [3]
##   Medal_type Season Country Medal_count
##   <fct>      <fct>  <fct>         <int>
## 1 Bronze     Winter USA             161
## 2 Bronze     Winter SWE             177
## 3 Bronze     Winter FIN             215
## 4 Gold       Winter USA             166
## 5 Gold       Winter URS             250
## 6 Gold       Winter CAN             305
## 7 Silver     Winter NOR             165
## 8 Silver     Winter CAN             199
## 9 Silver     Winter USA             308
```

12) Height variable standardization is written in Height_z_scores

```
Height_z_scores <- rep(NA, nrow(total))
Height_z_scores[!is.na(total$Height)] <-
  scale(total$Height[!is.na(total$Height)], center=TRUE, scale = TRUE)
```

14) Let's check if there are differences between men and women in height, weight and age with a two-sample t-test

```
t.test(formula =  Height~Sex , data = total)
```

```
##
##  Welch Two Sample t-test
##
## data:  Height by Sex
## t = -263.1, df = 139788, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group F and group M is not equal to 0
## 95 percent confidence interval:
##  -11.10107 -10.93690
## sample estimates:
## mean in group F mean in group M
##        167.8396        178.8586
```

```
t.test(formula =  Weight~Sex , data = total)
```

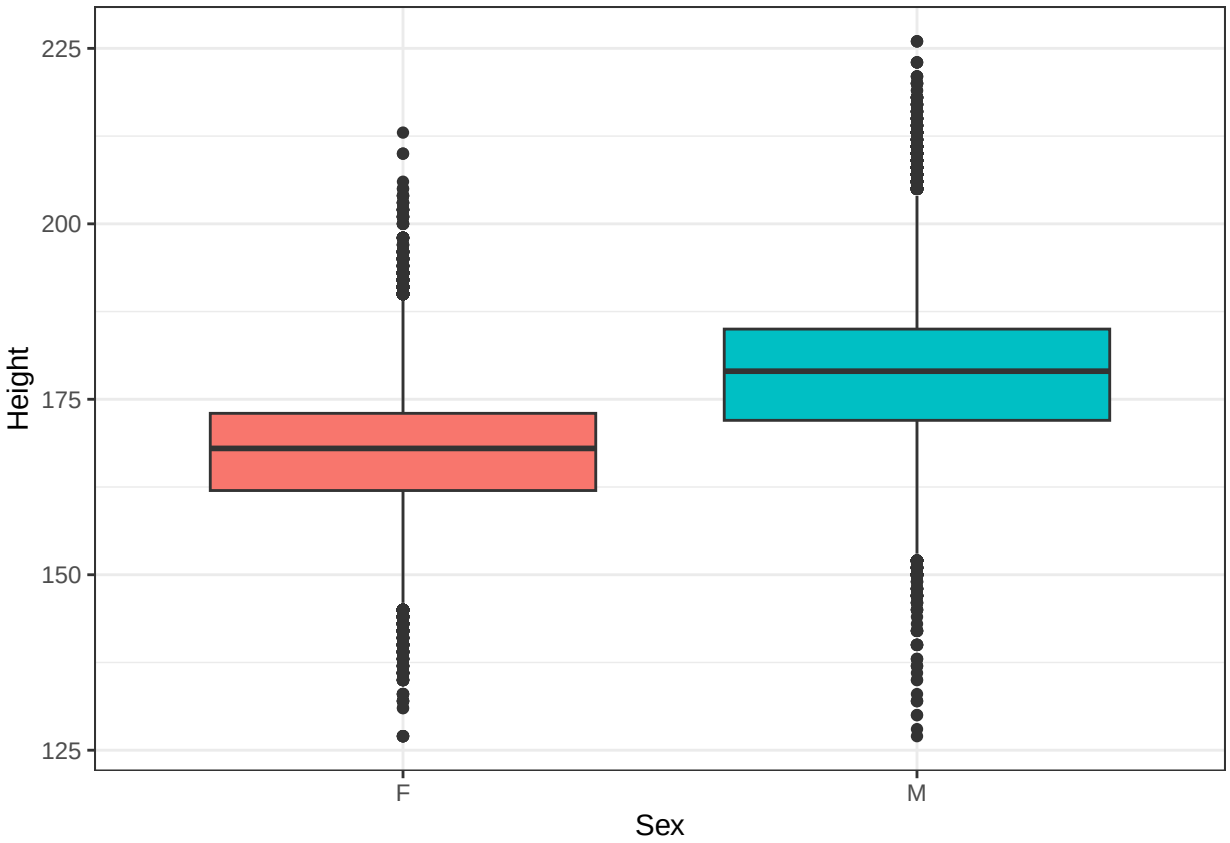
```
##
##  Welch Two Sample t-test
##
## data:  Weight by Sex
## t = -297.34, df = 165274, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group F and group M is not equal to 0
## 95 percent confidence interval:
##  -15.82588 -15.61860
## sample estimates:
## mean in group F mean in group M
##        60.02133        75.74356
```

```
t.test(formula =  Age~Sex , data = total)
```

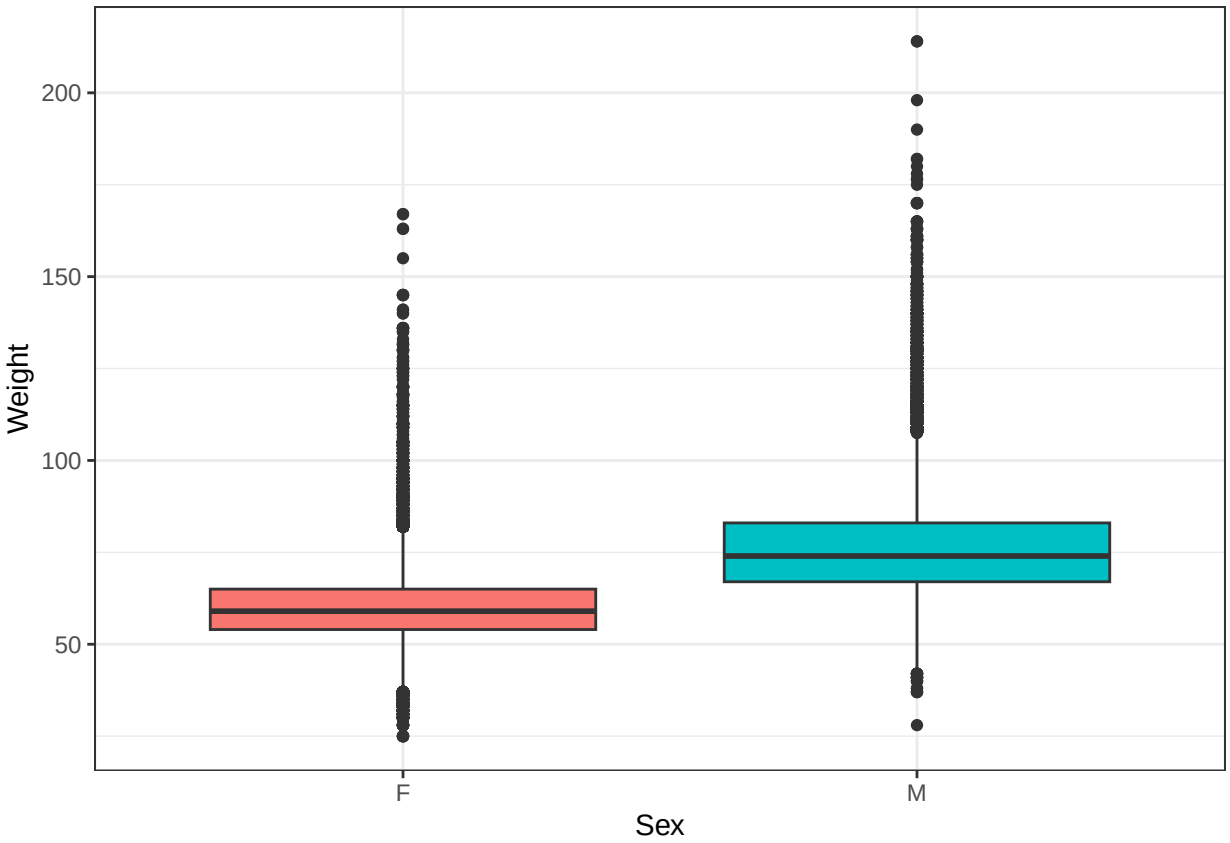
```
##
##  Welch Two Sample t-test
##
## data:  Age by Sex
## t = -97.814, df = 150733, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group F and group M is not equal to 0
## 95 percent confidence interval:
##  -2.595648 -2.493669
## sample estimates:
## mean in group F mean in group M
##        23.73284        26.27750
```

p-value in all three cases is greater than 0.05, which means that the differences in the values are statistically significant. Let's plot graphs for all three cases.

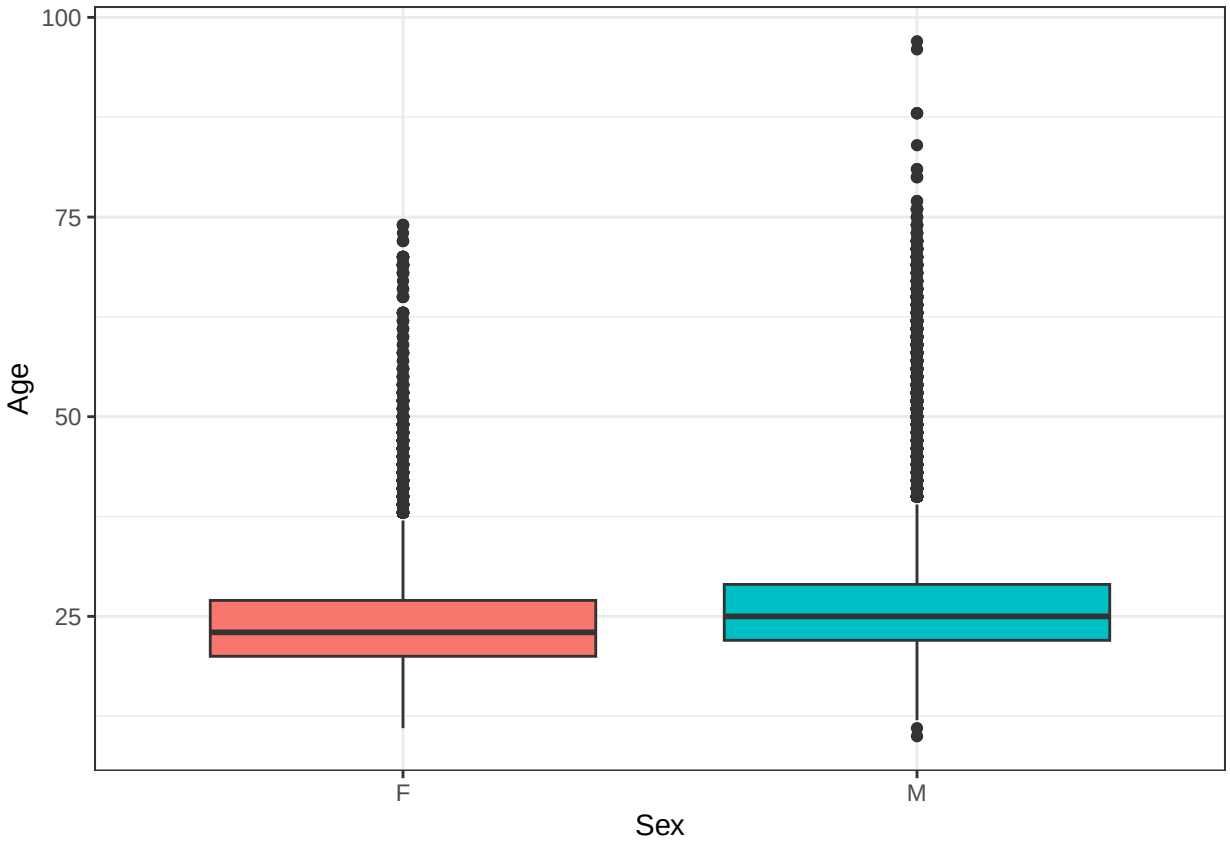
```
plot <- total %>% filter(!is.na(Sex)&!is.na(Season)) %>% ggplot(., aes(x = Sex, fill = Sex)) +
  theme(legend.position =" none ")
plot + geom_boxplot(aes(y = Height))
```




```
plot + geom_boxplot(aes(y = Weight))
```




```
plot + geom_boxplot(aes(y = Age))
```




It would be possible to draw p-values on the graphs, but when I tried to install the package for this, everything just broke and P stopped seeing the installed packages. Reinstalling R, I realized that well, for the article, you can then add p-values via Photoshop