Project1

Anna Toidze

10/24/2021

Analysis of olympic games

In this project it is assumed that the zipped raw data is located in a folder called *data_folder*, located in the directory above the current one. First we unzip our data saved as *athlete_events.zip*:

```
pwd
rm ../data_folder/*.csv
unzip ../data_folder/athlete_events.zip -d ../data_folder/
```

Let's see how our data looks like:

```
ls ../data_folder/
```

```
## athlete_events00.csv
## athlete_events01.csv
## athlete_events02.csv
## athlete_events03.csv
## athlete_events04.csv
## athlete_events05.csv
## athlete_events06.csv
## athlete_events07.csv
## athlete_events08.csv
## athlete_events09.csv
## athlete_events10.csv
## athlete_events11.csv
## athlete_events.zip
## features_description.txt
```

1. So we have 11 .csv files, that have to unite in one table. Let's load the needed libraries:

```
packages <- c("ggplot2", "dplyr", "ggExtra", "cowplot", "car", "broomExtra", "purrr", "stringr")
package.check <- lapply(
  packages,
  FUN = function(x) {
    if (!require(x, character.only = TRUE)) {
      install.packages(x, dependencies = TRUE)
      library(x, character.only = TRUE)
    }
  }
)
```

Uniting all data in one dataframe *df_ol*:

```
uniting_data <- function(path, extension = "csv"){
  pattern <- paste0("*.", extension)
```

```
  myfiles <-  paste0(path, list.files(path, pattern=pattern, all.files=FALSE,
           full.names=FALSE))
  myfiles %>% lapply(read.csv) %>% bind_rows()
}
path <- "../data_folder/"

df_ol <- uniting_data(path, "csv")
```

2. Checking whether our data is actually correct:

```
str(df_ol)
unique(df_ol$Sex)
unique(df_ol$Season)
unique(df_ol$Medal)
```

It's obvious, that instead of *NA* empty string " " has sometimes been used to showcase the lacking data. Let's reload our data with correct parameters:

```
uniting_data <- function(path, extension = "csv", na = "NA"){
  pattern <- paste0("*.", extension)
  myfiles <-  paste0(path, list.files(path, pattern=pattern, all.files=FALSE,
           full.names=FALSE))
  myfiles %>% lapply(read.csv, na.string = na) %>% bind_rows()
}

df_ol <- uniting_data("../data_folder/", "csv", na = c("", "NA"))
```

Let's factor our data, "G" has been changed to "M" based on the names of the participants and the events they participated in:

```
unique(df_ol$Sex)
```

```
## [1] "M" "F" "G" NA
```

```
df_ol <- df_ol %>% mutate(Sex=replace(Sex, Sex == "G", "M")) #df_ol[which(df_ol$Sex == "G"),] <- "M"

str(df_ol)
```

```
## '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" ...
```

```
df_ol$Sex <- factor(df_ol$Sex)
df_ol$Season <- factor(df_ol$Season)
```

```
df_ol$Medal <- factor(df_ol$Medal)
```

Let's check some other parameters starting with Sport column. There seems to be an typo with football - "Footba" instead of "Football".

```
unique(df_ol$Sport)
```

```
##  [1] "Basketball"                "Judo"
##  [3] "Football"                  "Tug-Of-War"
##  [5] "Speed Skating"             "Cross Country Skiing"
##  [7] "Athletics"                 "Ice Hockey"
##  [9] "Swimming"                  "Badminton"
## [11] "Sailing"                   "Biathlon"
## [13] "Gymnastics"                "Art Competitions"
## [15] "Alpine Skiing"             "Handball"
## [17] "Weightlifting"             "Wrestling"
## [19] "Luge"                      "Water Polo"
## [21] "Hockey"                    "Rowing"
## [23] "Bobsleigh"                 "Fencing"
## [25] "Equestrianism"             "Shooting"
## [27] "Boxing"                    "Taekwondo"
## [29] "Cycling"                   "Diving"
## [31] "Canoeing"                  "Tennis"
## [33] "Modern Pentathlon"         "Figure Skating"
## [35] "Golf"                      "Softball"
## [37] "Archery"                   "Volleyball"
## [39] "Synchronized Swimming"     "Table Tennis"
## [41] "Nordic Combined"           "Baseball"
## [43] "Rhythmic Gymnastics"       "Freestyle Skiing"
## [45] "Rugby Sevens"              "Trampolining"
## [47] "Beach Volleyball"          "Triathlon"
## [49] "Ski Jumping"               "Curling"
## [51] "Snowboarding"              "Rugby"
## [53] "Short Track Speed Skating" "Skeleton"
## [55] "Lacrosse"                  "Polo"
## [57] "Cricket"                   "Racquets"
## [59] "Motorboating"              "Military Ski Patrol"
## [61] "Croquet"                   "Jeu De Paume"
## [63] NA                          "Roque"
## [65] "Alpinism"                  "Basque Pelota"
## [67] "Footba"                    "Aeronautics"
```

```
df_ol <- df_ol %>% mutate(Sport=replace(Sport, Sport == "Footba", "Football"))
unique(df_ol$Sport)
```

```
##  [1] "Basketball"                "Judo"
##  [3] "Football"                  "Tug-Of-War"
##  [5] "Speed Skating"             "Cross Country Skiing"
##  [7] "Athletics"                 "Ice Hockey"
##  [9] "Swimming"                  "Badminton"
## [11] "Sailing"                   "Biathlon"
## [13] "Gymnastics"                "Art Competitions"
## [15] "Alpine Skiing"             "Handball"
## [17] "Weightlifting"             "Wrestling"
## [19] "Luge"                      "Water Polo"
```

```
## [21] "Hockey"                    "Rowing"
## [23] "Bobsleigh"                 "Fencing"
## [25] "Equestrianism"             "Shooting"
## [27] "Boxing"                    "Taekwondo"
## [29] "Cycling"                   "Diving"
## [31] "Canoeing"                  "Tennis"
## [33] "Modern Pentathlon"         "Figure Skating"
## [35] "Golf"                      "Softball"
## [37] "Archery"                   "Volleyball"
## [39] "Synchronized Swimming"     "Table Tennis"
## [41] "Nordic Combined"           "Baseball"
## [43] "Rhythmic Gymnastics"       "Freestyle Skiing"
## [45] "Rugby Sevens"              "Trampolining"
## [47] "Beach Volleyball"          "Triathlon"
## [49] "Ski Jumping"               "Curling"
## [51] "Snowboarding"              "Rugby"
## [53] "Short Track Speed Skating" "Skeleton"
## [55] "Lacrosse"                  "Polo"
## [57] "Cricket"                   "Racquets"
## [59] "Motorboating"              "Military Ski Patrol"
## [61] "Croquet"                   "Jeu De Paume"
## [63] NA                          "Roque"
## [65] "Alpinism"                  "Basque Pelota"
## [67] "Aeronautics"
```

Let's check our weight, age and height distributions.

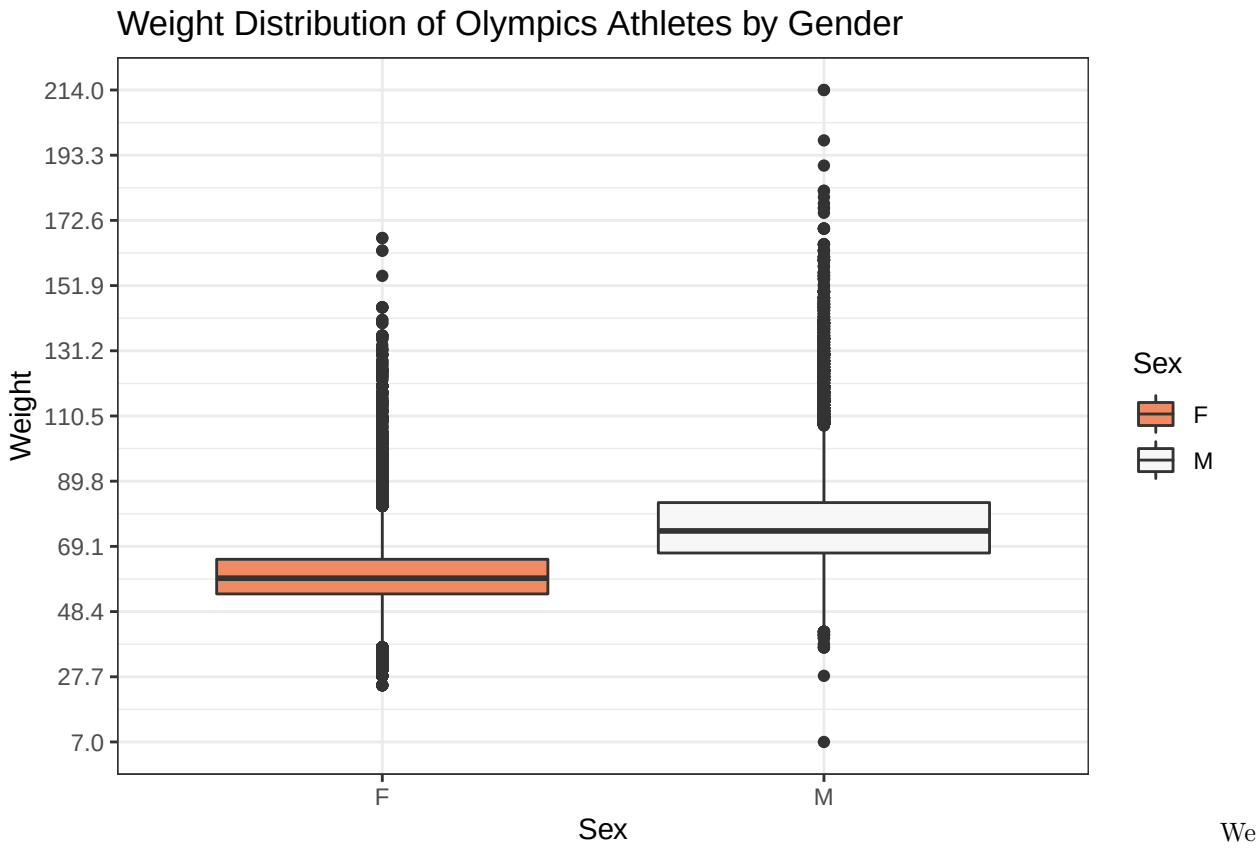
```
max(df_ol$Weight, na.rm = T)
```

```
## [1] 214
```

```
min(df_ol$Weight, na.rm = T)
```

```
## [1] 7
```

```
ggplot(subset(df_ol, complete.cases(Sex)), aes(x=Sex, y = Weight, fill = Sex))+
  geom_boxplot()+
  scale_fill_brewer(palette="RdBu") +
  scale_y_continuous(limits = c(min(df_ol$Weight, na.rm = T)
, max(df_ol$Weight, na.rm = T)), breaks = seq(min(df_ol$Weight, na.rm = T)
, max(df_ol$Weight, na.rm = T)
, (max(df_ol$Weight, na.rm = T)-min(df_ol$Weight, na.rm = T))/10
)) +

   labs(title= "Weight Distribution of Olympics Athletes by Gender")+
  theme_bw()
```



We have a weird outlier with the weight of less than 25. We have Helmut Lehmann, who doesn't really contribute anything except for his age and height - no other parameters are known. Let's just filter him off.

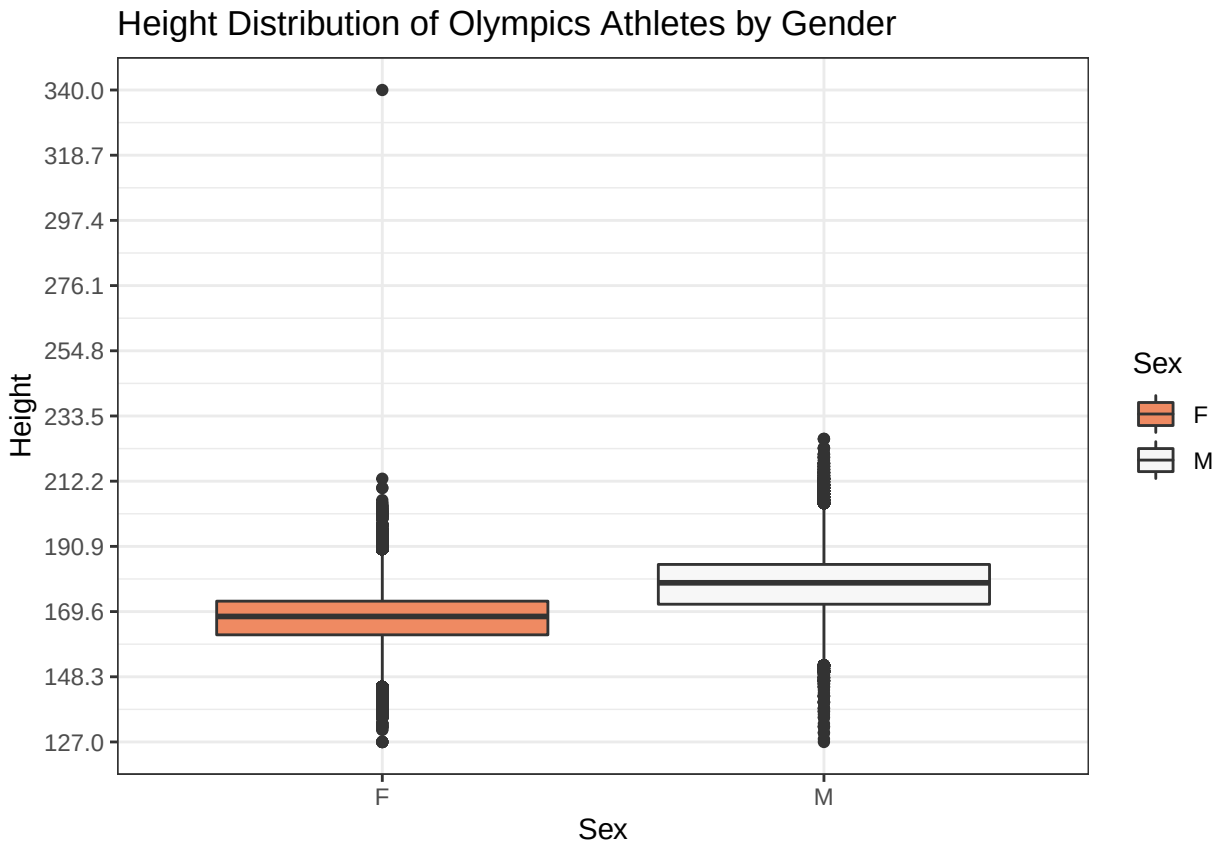
```
df_ol[which(df_ol$Weight < 25),]
```

```
##           ID           Name Sex Age Height Weight Team  NOC Games Year Season
## 135886 68370 Helmut Lehmann   M  25    178      7 <NA> <NA>  <NA>   NA   <NA>
##        City Sport Event Medal
## 135886 <NA>  <NA>  <NA>  <NA>
```

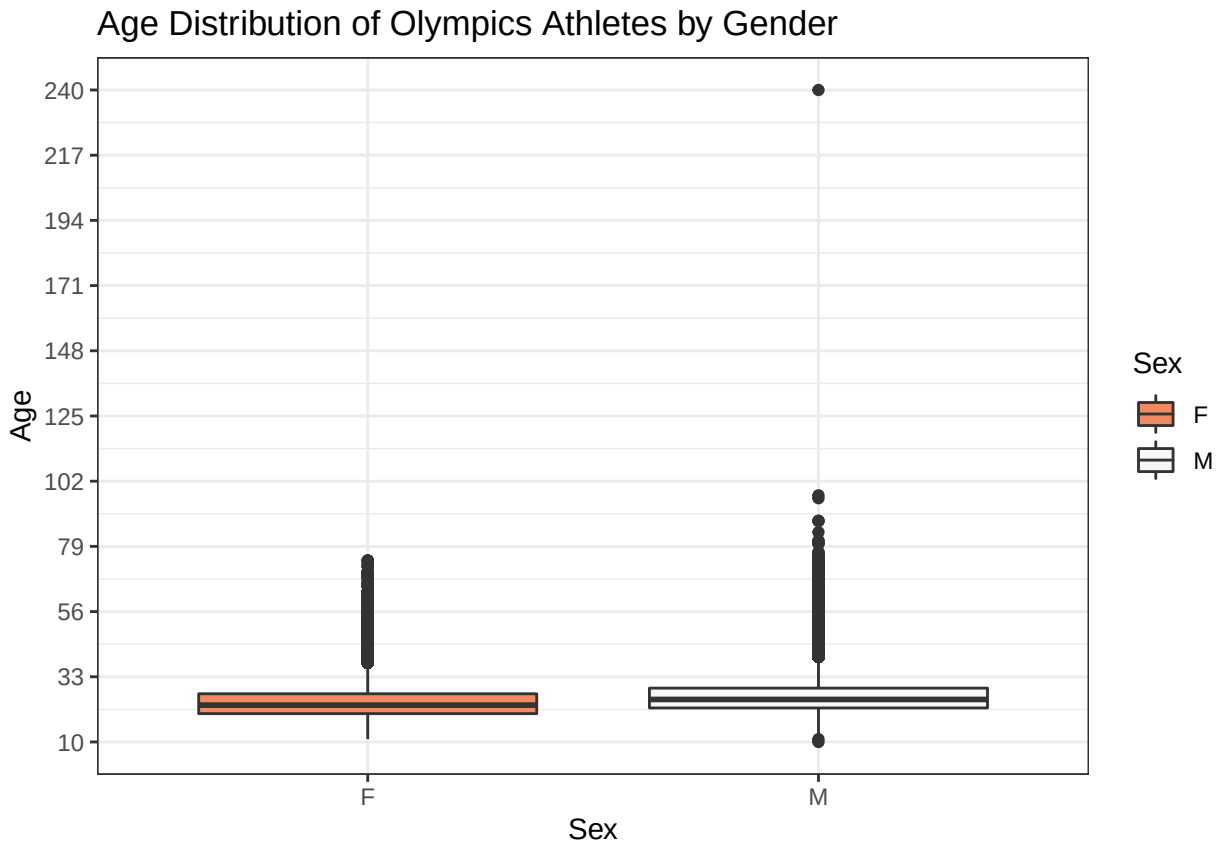
```
df_ol <- df_ol[-c(which(df_ol$Weight < 25)),]
```

Let's do the same with age and height. We have a very tall female, with height of 340 which is impossible, and a male of age 240. Let's check whether that woman might have participated in other events as well and substitute her height accordingly. Same goes for the age.

```
ggplot(subset(df_ol, complete.cases(Sex)), aes(x=Sex, y = Height, fill = Sex))+
  geom_boxplot()+
  scale_fill_brewer(palette="RdBu") +
  scale_y_continuous(limits = c(min(df_ol$Height, na.rm = T)
, max(df_ol$Height, na.rm = T)), breaks = seq(min(df_ol$Height, na.rm = T)
, max(df_ol$Height, na.rm = T)
, (max(df_ol$Height, na.rm = T)-min(df_ol$Height, na.rm = T))/10
)) +

   labs(title= "Height Distribution of Olympics Athletes by Gender")+
  theme_bw()
```




```
ggplot(subset(df_ol, complete.cases(Sex)), aes(x=Sex, y = Age, fill = Sex))+
  geom_boxplot()+
  scale_fill_brewer(palette="RdBu") +
  scale_y_continuous(limits = c(min(df_ol$Age, na.rm = T)
, max(df_ol$Age, na.rm = T)), breaks = seq(min(df_ol$Age, na.rm = T)
, max(df_ol$Age, na.rm = T)
, (max(df_ol$Age, na.rm = T)-min(df_ol$Age, na.rm = T))/10
)) +

   labs(title= "Age Distribution of Olympics Athletes by Gender")+
  theme_bw()
```




```
#Who is 340 cm tall?
df_ol[which(df_ol$Height == 340),]
```

```
##          ID                            Name Sex Age Height Weight     Team NOC
## 45651 23549 Kirsty Leigh Coventry (-Seward)   F  28    340     64 Zimbabwe ZIM
##             Games Year Season   City    Sport
## 45651 2012 Summer 2012 Summer London Swimming
##                                              Event Medal
## 45651 Swimming Women's 200 metres Individual Medley  <NA>
```

```
#Let's check her height:
df_ol[which(df_ol$Name == "Kirsty Leigh Coventry (-Seward)"),]
```

```
##          ID                            Name Sex Age Height Weight     Team NOC
## 45638 23549 Kirsty Leigh Coventry (-Seward)   F  16    176     64 Zimbabwe ZIM
## 45639 23549 Kirsty Leigh Coventry (-Seward)   F  16    176     64 Zimbabwe ZIM
## 45640 23549 Kirsty Leigh Coventry (-Seward)   F  16    176     64 Zimbabwe ZIM
## 45641 23549 Kirsty Leigh Coventry (-Seward)   F  16    176     64 Zimbabwe ZIM
## 45642 23549 Kirsty Leigh Coventry (-Seward)   F  20    176     64 Zimbabwe ZIM
## 45643 23549 Kirsty Leigh Coventry (-Seward)   F  20    176     64 Zimbabwe ZIM
## 45644 23549 Kirsty Leigh Coventry (-Seward)   F  20    176     64 Zimbabwe ZIM
## 45645 23549 Kirsty Leigh Coventry (-Seward)   F  24    176     64 Zimbabwe ZIM
## 45646 23549 Kirsty Leigh Coventry (-Seward)   F  24    176     64 Zimbabwe ZIM
## 45647 23549 Kirsty Leigh Coventry (-Seward)   F  24    176     64 Zimbabwe ZIM
## 45648 23549 Kirsty Leigh Coventry (-Seward)   F  24    176     64 Zimbabwe ZIM
## 45649 23549 Kirsty Leigh Coventry (-Seward)   F  28    176     64 Zimbabwe ZIM
## 45650 23549 Kirsty Leigh Coventry (-Seward)   F  28    176     64 Zimbabwe ZIM
## 45651 23549 Kirsty Leigh Coventry (-Seward)   F  28    340     64 Zimbabwe ZIM
```

```
## 45652 23549 Kirsty Leigh Coventry (-Seward)   F  32    176     64 Zimbabwe ZIM
## 45653 23549 Kirsty Leigh Coventry (-Seward)   F  32    176     64 Zimbabwe ZIM
##             Games Year Season           City    Sport
## 45638 2000 Summer 2000 Summer         Sydney Swimming
## 45639 2000 Summer 2000 Summer         Sydney Swimming
## 45640 2000 Summer 2000 Summer         Sydney Swimming
## 45641 2000 Summer 2000 Summer         Sydney Swimming
## 45642 2004 Summer 2004 Summer         Athina Swimming
## 45643 2004 Summer 2004 Summer         Athina Swimming
## 45644 2004 Summer 2004 Summer         Athina Swimming
## 45645 2008 Summer 2008 Summer        Beijing Swimming
## 45646 2008 Summer 2008 Summer        Beijing Swimming
## 45647 2008 Summer 2008 Summer        Beijing Swimming
## 45648 2008 Summer 2008 Summer        Beijing Swimming
## 45649 2012 Summer 2012 Summer         London Swimming
## 45650 2012 Summer 2012 Summer         London Swimming
## 45651 2012 Summer 2012 Summer         London Swimming
## 45652 2016 Summer 2016 Summer Rio de Janeiro Swimming
## 45653 2016 Summer 2016 Summer Rio de Janeiro Swimming
##                                              Event  Medal
## 45638          Swimming Women's 50 metres Freestyle   <NA>
## 45639         Swimming Women's 100 metres Freestyle   <NA>
## 45640        Swimming Women's 100 metres Backstroke   <NA>
## 45641 Swimming Women's 200 metres Individual Medley   <NA>
## 45642        Swimming Women's 100 metres Backstroke Silver
## 45643        Swimming Women's 200 metres Backstroke   Gold
## 45644 Swimming Women's 200 metres Individual Medley Bronze
## 45645        Swimming Women's 100 metres Backstroke Silver
## 45646        Swimming Women's 200 metres Backstroke   Gold
## 45647 Swimming Women's 200 metres Individual Medley Silver
## 45648 Swimming Women's 400 metres Individual Medley Silver
## 45649        Swimming Women's 100 metres Backstroke   <NA>
## 45650        Swimming Women's 200 metres Backstroke   <NA>
## 45651 Swimming Women's 200 metres Individual Medley   <NA>
## 45652        Swimming Women's 100 metres Backstroke   <NA>
## 45653        Swimming Women's 200 metres Backstroke   <NA>
```

```
#It seems to be 176 cm, let's correct the false data:
df_ol <- df_ol %>% mutate(Height=replace(Height, Height == 340, 176))
#Now height distribution looks good

#Who's 240 years old?
df_ol[which(df_ol$Age == 240), 10]
```

```
## [1] 1912
```

```
#Let's check whether he participated in other events as well:
df_ol %>% filter(Name == "Flicien Jules mile Courbet", Year == df_ol[which(df_ol$Age == 240), 10])
```

```
##      ID                      Name Sex Age Height Weight    Team NOC
## 1 23459 Flicien Jules mile Courbet   M  24     NA     NA Belgium BEL
## 2 23459 Flicien Jules mile Courbet   M 240     NA     NA Belgium BEL
## 3 23459 Flicien Jules mile Courbet   M  24     NA     NA Belgium BEL
##         Games Year Season      City      Sport
## 1 1912 Summer 1912 Summer Stockholm Water Polo
```

```
## 2 1912 Summer 1912 Summer Stockholm   Swimming
## 3 1912 Summer 1912 Summer Stockholm   Swimming
##                                    Event  Medal
## 1            Water Polo Men's Water Polo Bronze
## 2 Swimming Men's 200 metres Breaststroke   <NA>
## 3 Swimming Men's 400 metres Breaststroke   <NA>
```

```
#We see he's 24 that year:
df_ol <- df_ol %>%  mutate(Age=replace(Age, Age == 240, 24))
#Now age distribution is fixed
```

3. The age of two youngest athletes of each sex on Olympics in year 1992:

```
df_ol %>% filter(Year == 1992) %>% arrange(Age) %>% group_by(Sex) %>% slice(c(1)) %>% select(Age)
```

```
## Adding missing grouping variables: `Sex`
```

```
## # A tibble: 2 x 2
## # Groups:   Sex [2]
##   Sex     Age
##   <fct> <dbl>
## 1 F        12
## 2 M        11
```

4. Calculate mean and standard deviation for **height**, each sex:

```
aggregate(df_ol$Height, by = list(df_ol$Sex) , FUN = function(x) cbind(mean(x, na.rm = T), sd(x, na.rm =
```

```
##   Group.1        x.1        x.2
## 1       F 167.839728   8.778463
## 2       M 178.858620   9.360377
```

```
df_ol[complete.cases(df_ol[,3]),] %>%  group_by(Sex) %>%
  summarize(
    mean = mean(Height, na.rm = TRUE),  sd = sd(Height, na.rm = T)
  )
```

```
## # A tibble: 2 x 3
##   Sex    mean    sd
##   <fct> <dbl> <dbl>
## 1 F      168.  8.78
## 2 M      179.  9.36
```

```
df_ol %>% filter(is.na(Sex) == F) %>% group_by(Sex) %>%
  summarize(
    mean = mean(Height, na.rm = TRUE),  sd = sd(Height, na.rm = T)
  )
```

```
## # A tibble: 2 x 3
##   Sex    mean    sd
##   <fct> <dbl> <dbl>
## 1 F      168.  8.78
## 2 M      179.  9.36
```

5. Mean and standard deviation of **height** for female tennis players on 2000 Olympics. Round to 1 digit after decimal.

```
df_ol %>% filter(Sex %in% "F", Year %in% 2000) %>%
  summarize(
```

```
    mean = round(mean(Height, na.rm = TRUE), 1),  sd = round(sd(Height, na.rm = T), 1)
  )
```

```
##   mean  sd
## 1  169 9.5
```

6. In which category of sport did the heaviest athlete of 2006 Olympics participate?

```
df_ol %>% filter(Year %in% 2006) %>%
  arrange(-Weight) %>%
  slice(1) %>%
  select(Sport)
```

```
##      Sport
## 1 Skeleton
```

7. How many gold medals obtained by females from 1980 till 2010 (both inclusive)?

```
df_ol %>% filter(Sex %in% "F", Year %in% (1980:2010), Medal %in% "Gold") %>% summarize( n = n())
```

```
##      n
## 1 2249
```

8. How many times did John Aalberg participate in Olympic games over the years in total?

```
df_ol %>% filter(Name == "John Aalberg") %>%
  summarize(n = length(unique(Year)))
```

```
##   n
## 1 2
```

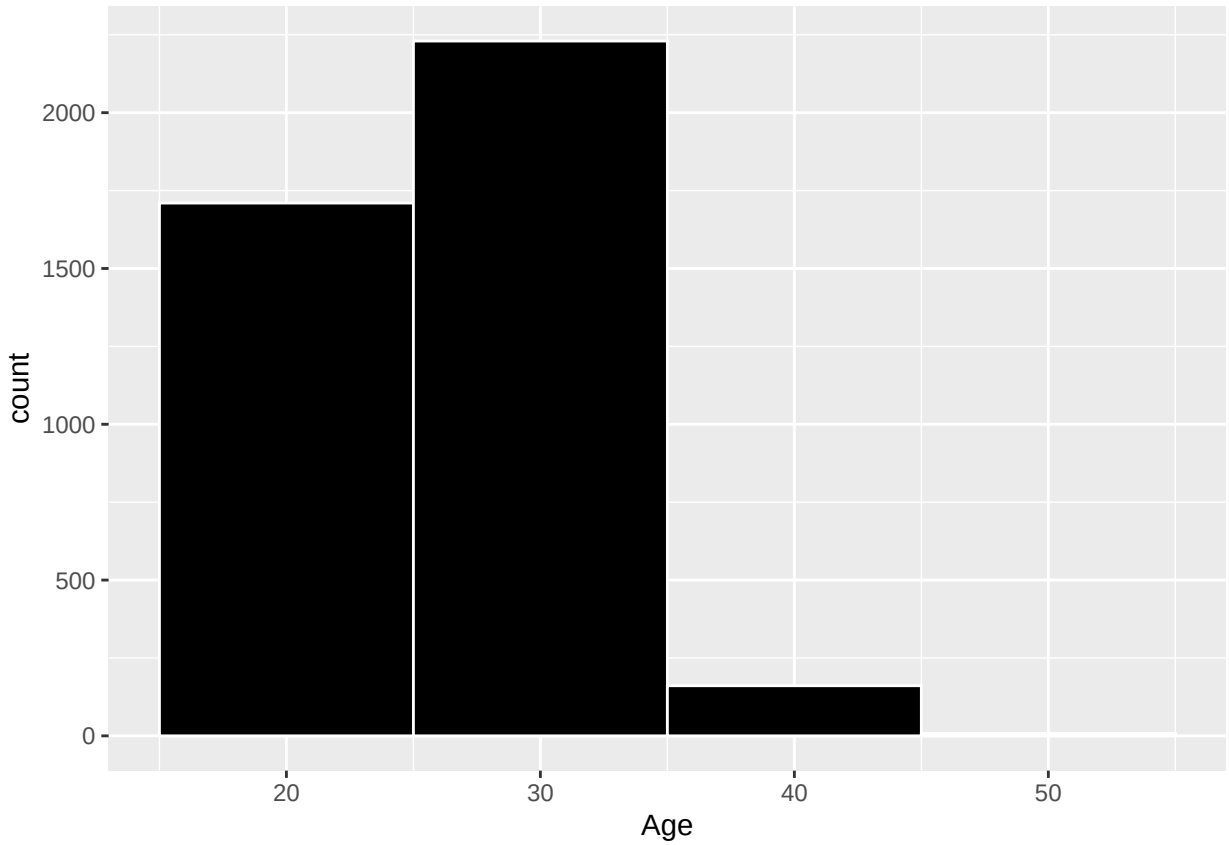
He participated twice in Olympic games.

9. Determine the most and least represented age group ([15-25), [25-35), [35-45), [45-55]) on 2008 Olympics. Least represented is the one between 45 and 55, most represented between 25 and 35.

```
df_ol %>% filter(Year %in% 2002) %>%
  mutate(Agegroup = cut(Age, breaks = c(15, 25, 35, 45, 56), right = F)) %>%
  group_by(Agegroup) %>%
  summarize(n = n()) %>%
  slice(which.max(n), which.min(n))
```

```
## # A tibble: 2 x 2
##   Agegroup     n
##   <fct>    <int>
## 1 [25,35)   2230
## 2 [45,56)      8
```

```
# df_ol_agegr[c(which.max(df_ol_agegr$n), which.min(df_ol_agegr$n)),]

df_ol_2002 <- df_ol[df_ol$Year %in% 2002, ]
ggplot(df_ol_2002, aes(Age)) +
  geom_histogram(boundary = 15, binwidth = 10, bins = 5, color = "white", fill = "black", right = F)
```



10. How did the number of different categories of sports change from 1994 to 2002 Olympics?

```
df_ol_num_sports_02_94 <- df_ol %>% filter(Year %in% c(1994, 2002)) %>%
  group_by(Year) %>%
  summarize(n = length(unique(Sport)))


#Increase by:
df_ol_num_sports_02_94$n[2] - df_ol_num_sports_02_94$n[1]
```

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

11. What are the top 3 countries for each Olympics season and medal?

```
df_ol %>% filter(is.na(Medal) == F) %>%
  group_by(Season, Medal, NOC) %>%
  summarize(n = n()) %>%
  summarize("Num. of Medals" = max(n), Country = paste(NOC[which(n == max(n))], collapse = ", "))
```

```
## `summarise()` has grouped output by 'Season', 'Medal'. You can override using the `.groups` argument
```

```
## `summarise()` has grouped output by 'Season'. You can override using the `.groups` argument.
```

```
## # A tibble: 6 x 4
## # Groups:   Season [2]
##   Season Medal  `Num. of Medals` Country
##   <fct>  <fct>             <int> <chr>
## 1 Summer Bronze             1197 USA
## 2 Summer Gold               2472 USA
## 3 Summer Silver             1333 USA
```

```
## 4 Winter Bronze              215 FIN
## 5 Winter Gold                305 CAN
## 6 Winter Silver              308 USA
```

```
df_ol %>% filter(is.na(Medal) == F) %>%
  group_by(Season, Medal, NOC) %>%
  summarize(n = n()) %>%
  arrange(-n) %>%
  slice(1:3) %>%
 summarise_all(~paste(., collapse = '; '))
```

```
## `summarise()` has grouped output by 'Season', 'Medal'. You can override using the `.groups` argument
```

```
## # A tibble: 6 x 4
## # Groups:   Season [2]
##   Season Medal  NOC           n
##   <fct>  <fct>  <chr>         <chr>
## 1 Summer Bronze USA; GER; GBR 1197; 649; 620
## 2 Summer Gold   USA; URS; GBR 2472; 832; 636
## 3 Summer Silver USA; GBR; URS 1333; 729; 635
## 4 Winter Bronze FIN; SWE; USA 215; 177; 161
## 5 Winter Gold   CAN; URS; USA 305; 250; 166
## 6 Winter Silver USA; CAN; NOR 308; 199; 165
```

12. Put standardized height values into a new column *Height_z_scores*:

```
df_ol_z <- df_ol %>%
  mutate(Height_z_scores = (Height - mean(Height, na.rm =T))/sd(Height, na.rm = T))

head(scale(df_ol$Height))
```

```
##             [,1]
## [1,]  0.4431254
## [2,] -0.5075774
## [3,]         NA
## [4,]         NA
## [5,]  0.9184768
## [6,]  0.9184768
```

```
df_ol_scale <- df_ol %>% mutate(Height_scaled = scale(Height))
```

13. Put standardized height values into a new column *Height_min_max_scores* (min-max normalization, every value in [0,1]):

```
df_ol_z_minmax <- df_ol_z %>%
  mutate(Height_min_max_scaled = (Height - min(Height, na.rm =T))/(max(Height, na.rm = T)-min(Height, na
```

14. Height, weight and age of male and female athletes, that participated in winter Olympics. As these variables are continuous, we could theoriticaly compare them by t-test. For this purpose, we first have to test for normal distribution.

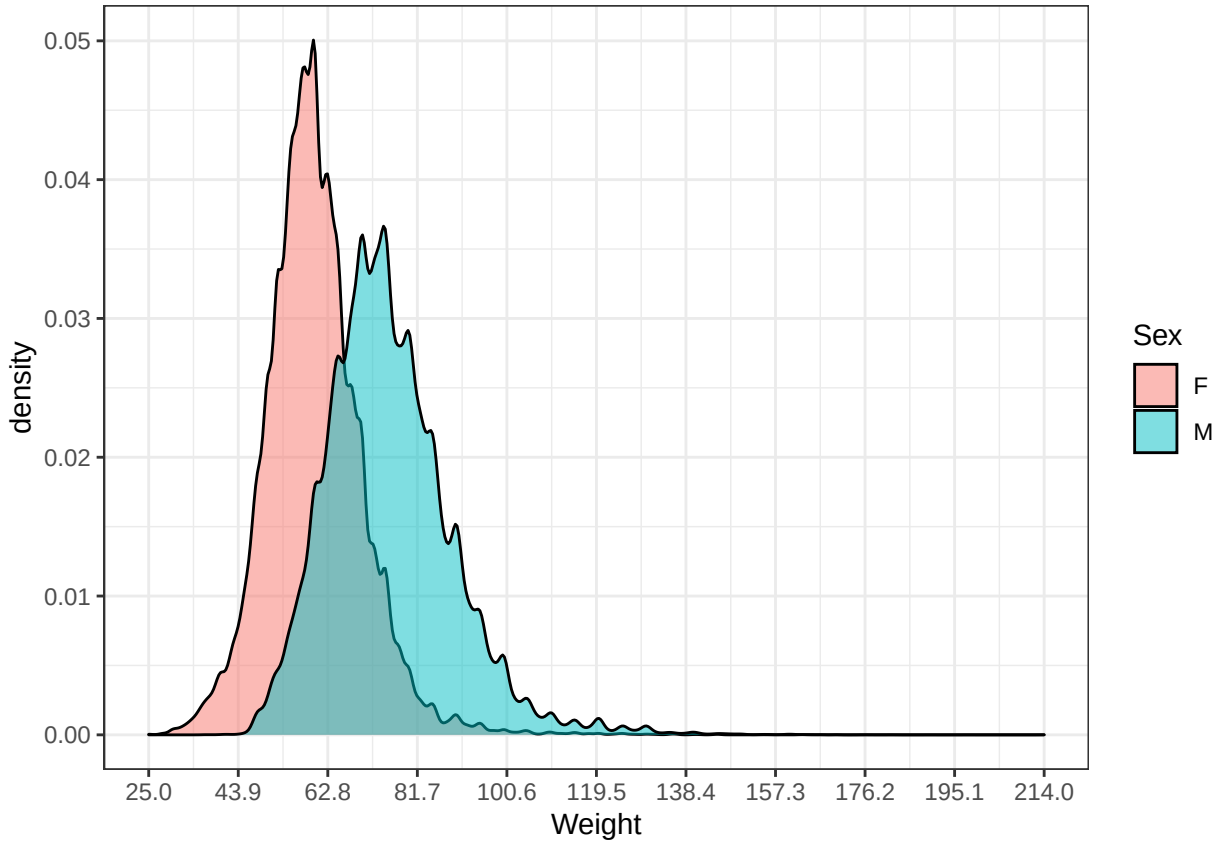
```
#General statistics:
df_ol %>% filter(Season %in% "Winter") %>%
  group_by(Sex) %>%
  summarize(across(.cols = c(Height, Weight, Age), list(mean = ~ round(mean(x = ., na.rm = T), 2), sd =
```

```
## # A tibble: 2 x 7
##   Sex   Height_mean Height_sd Weight_mean Weight_sd Age_mean Age_sd
##   <fct>       <dbl>     <dbl>       <dbl>     <dbl>    <dbl>  <dbl>
```

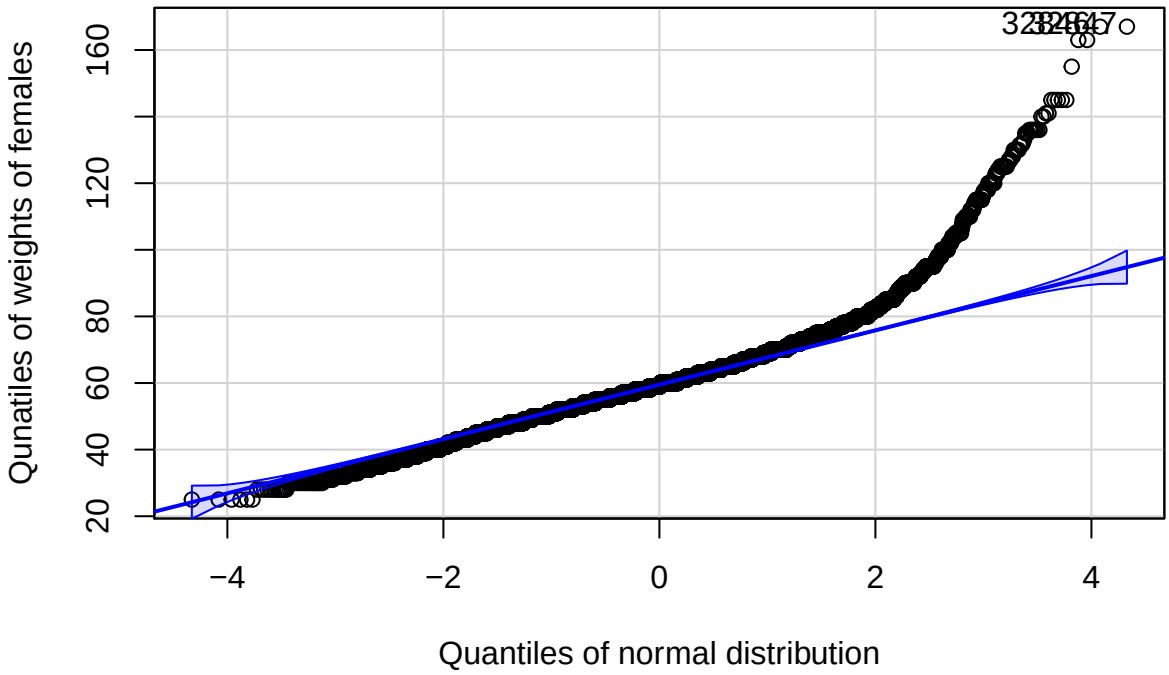
```
## 1 F            167.      6.03        59.8      7.06     24.0   4.69
## 2 M            179.      6.59        76.4     10.3      25.5   4.75
```

Let's take a look at weight first. The data is normally distributed as seen on QQplot and the density distribution, so we can use the t-test.

```
#Testing for normal distribution
ggplot(subset(df_ol, complete.cases(Sex)), aes(Weight, fill = Sex))+
  geom_density(alpha = 0.5)+
  scale_x_continuous(name = "Weight",
                     limits = c(min(df_ol$Weight, na.rm = T), max(df_ol$Weight, na.rm = T)),
                     breaks = seq(min(df_ol$Weight, na.rm = T), max(df_ol$Weight, na.rm = T),
                                  (max(df_ol$Weight, na.rm=T)-min(df_ol$Weight, na.rm = T))/10)) +
  scale_fill_discrete(name = "Sex")+
  theme_bw()
```

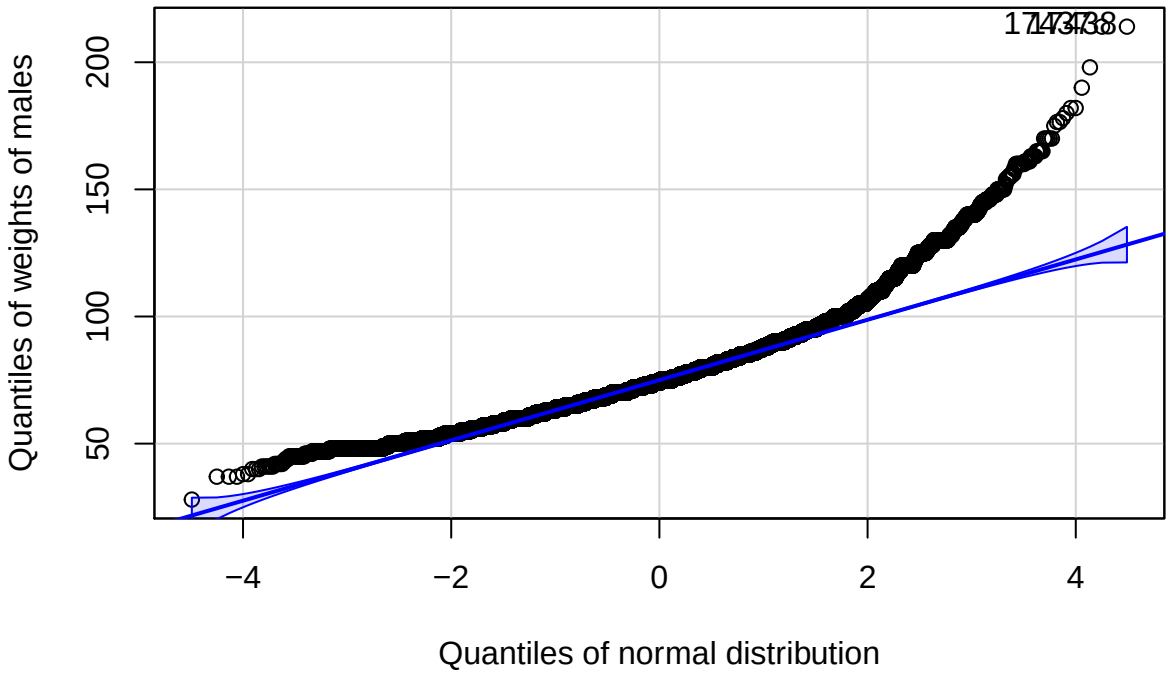



```
#QQplot
qqPlot(filter(df_ol, Sex == "F")$Weight, xlab = "Quantiles of normal distribution", ylab = "Qunatiles o
```




```
## [1] 32846 32847
```

```
qqPlot(filter(df_ol, Sex == "M")$Weight, xlab = "Quantiles of normal distribution", ylab = "Quantiles o
```




```
## [1] 17437 17438
```

```
#t-test
t_weight <- t.test(filter(df_ol, Sex == "F")$Weight, filter(df_ol, Sex == "M")$Weight, alternative = "t
t_weight
```

```
##
##  Welch Two Sample t-test
```

```
##
## data:  filter(df_ol, Sex == "F")$Weight and filter(df_ol, Sex == "M")$Weight
## t = -297.36, df = 165260, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -15.82638 -15.61911
## sample estimates:
## mean of x mean of y
##  60.02133  75.74407
```

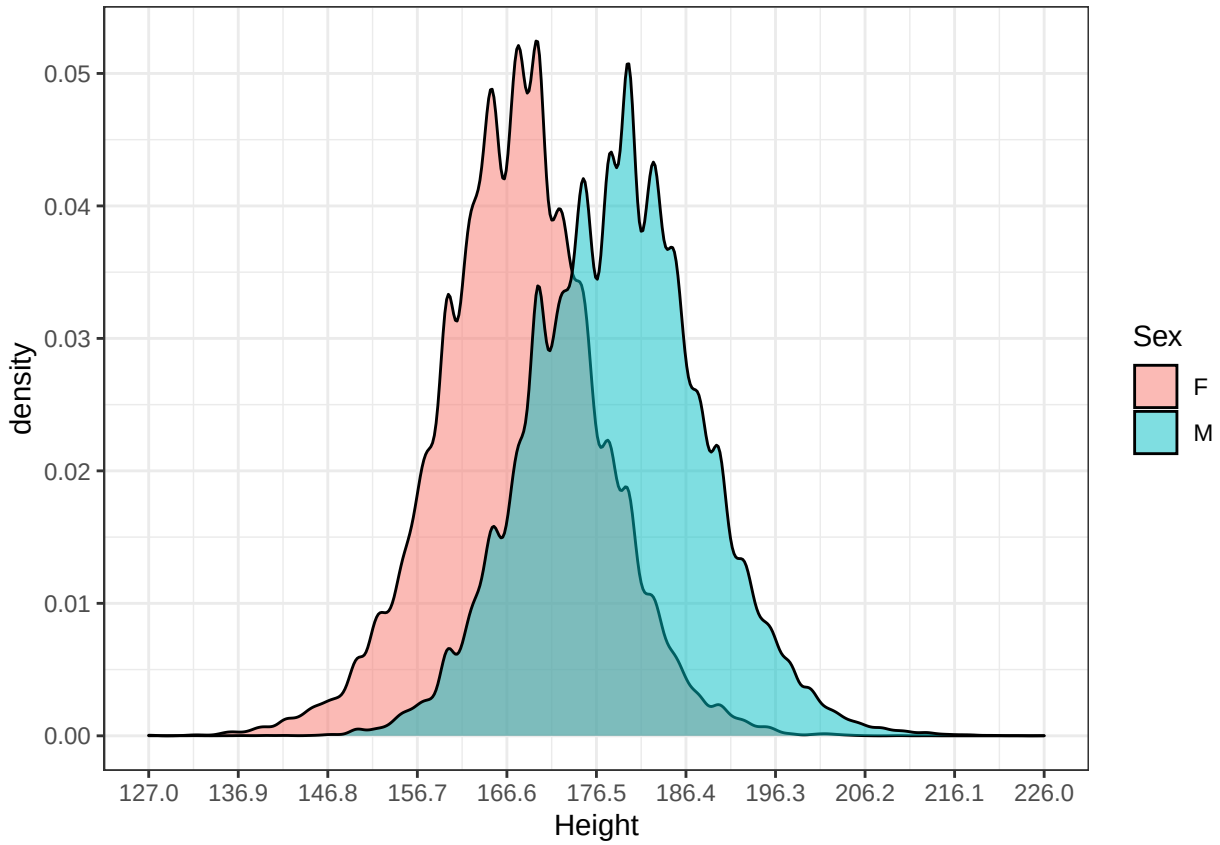
```
t_weight <- tidy(t_weight)[c("statistic", "p.value", "parameter", "method", "alternative")]
names(t_weight) <- c("Statistic", "P Value", "Degrees of Freedom", "Method", "Alternative Hypothesis")
t_weight
```

```
## # A tibble: 1 x 5
##   Statistic `P Value` `Degrees of Freedom` Method                  `Alternative Hy~
##       <dbl>     <dbl>                <dbl> <chr>                   <chr>
## 1     -297.         0              165260. Welch Two Sample t-test two.sided
```

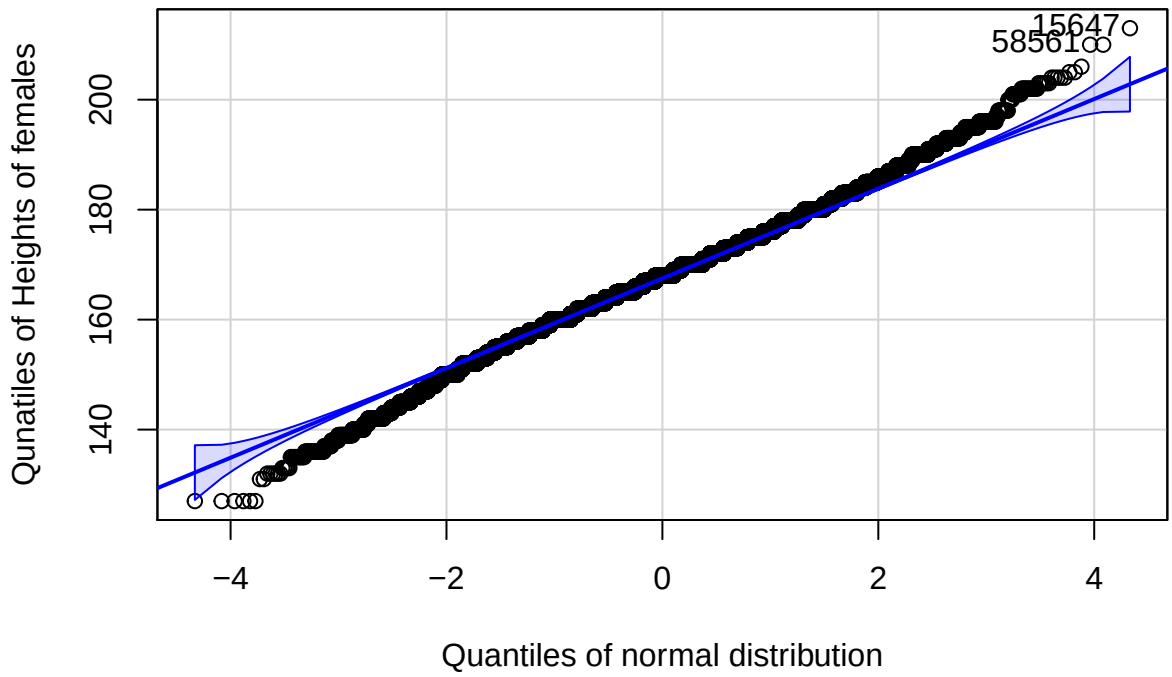
Description as in article: The weight distribution of females and males participating in Olympic games were compared. As the weight was normally distributed, two-sided Welch Two Sample t-test was used and significant difference was discovered, with test statistic equal to -297.3577556 and p-Value of 0.

Now let's do the same for height:

```
#Testing for normal distribution
ggplot(subset(df_ol, complete.cases(Sex)), aes(Height, fill = Sex))+
  geom_density(alpha = 0.5)+
  scale_x_continuous(name = "Height",
                     limits = c(min(df_ol$Height, na.rm = T), max(df_ol$Height, na.rm = T)),
                     breaks = seq(min(df_ol$Height, na.rm = T), max(df_ol$Height, na.rm = T),
                                  (max(df_ol$Height, na.rm=T)-min(df_ol$Height, na.rm = T))/10)) +
  scale_fill_discrete(name = "Sex")+
  theme_bw()
```

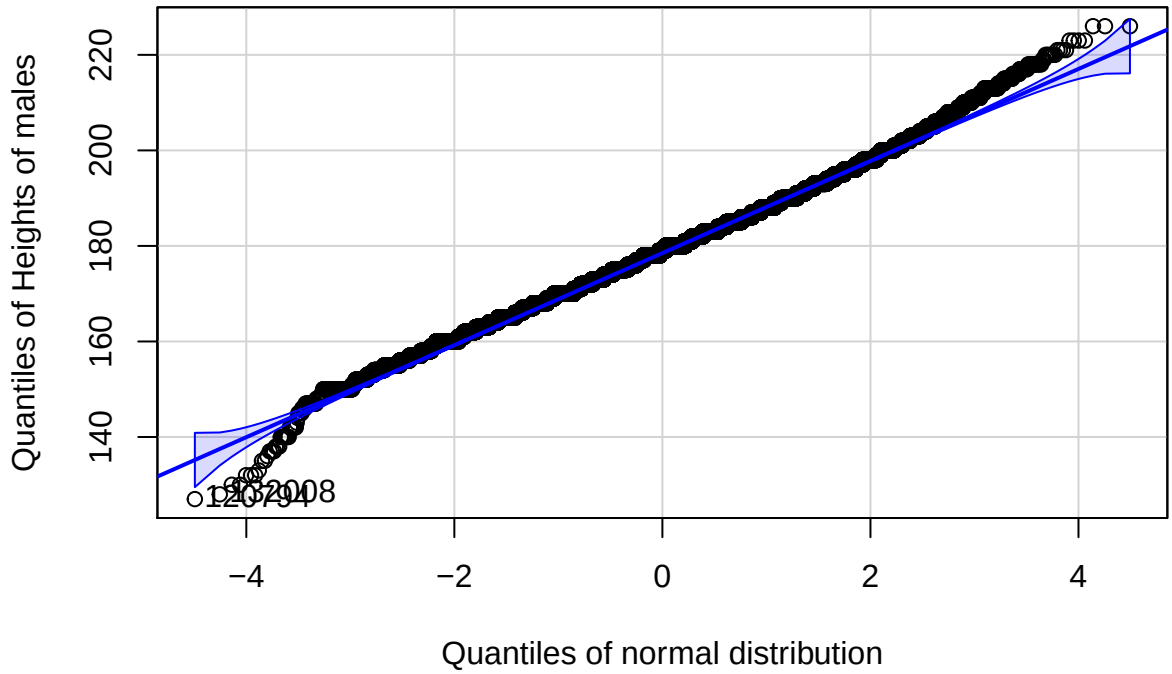



```
#QQplot
qqPlot(filter(df_ol, Sex == "F")$Height, xlab = "Quantiles of normal distribution", ylab = "Qunatiles o
```




```
## [1] 15647 58561
```

```
qqPlot(filter(df_ol, Sex == "M")$Height, xlab = "Quantiles of normal distribution", ylab = "Quantiles o
```




```
## [1] 120794 132008
```

```
#Everything is normally distributed.

#t-test
t_height <- t.test(filter(df_ol, Sex == "F")$Height, filter(df_ol, Sex == "M")$Height, alternative = "t
t_height
```

```
##
##  Welch Two Sample t-test
##
## data:  filter(df_ol, Sex == "F")$Height and filter(df_ol, Sex == "M")$Height
## t = -263.1, df = 139790, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -11.10098 -10.93681
## sample estimates:
## mean of x mean of y
##  167.8397  178.8586
```

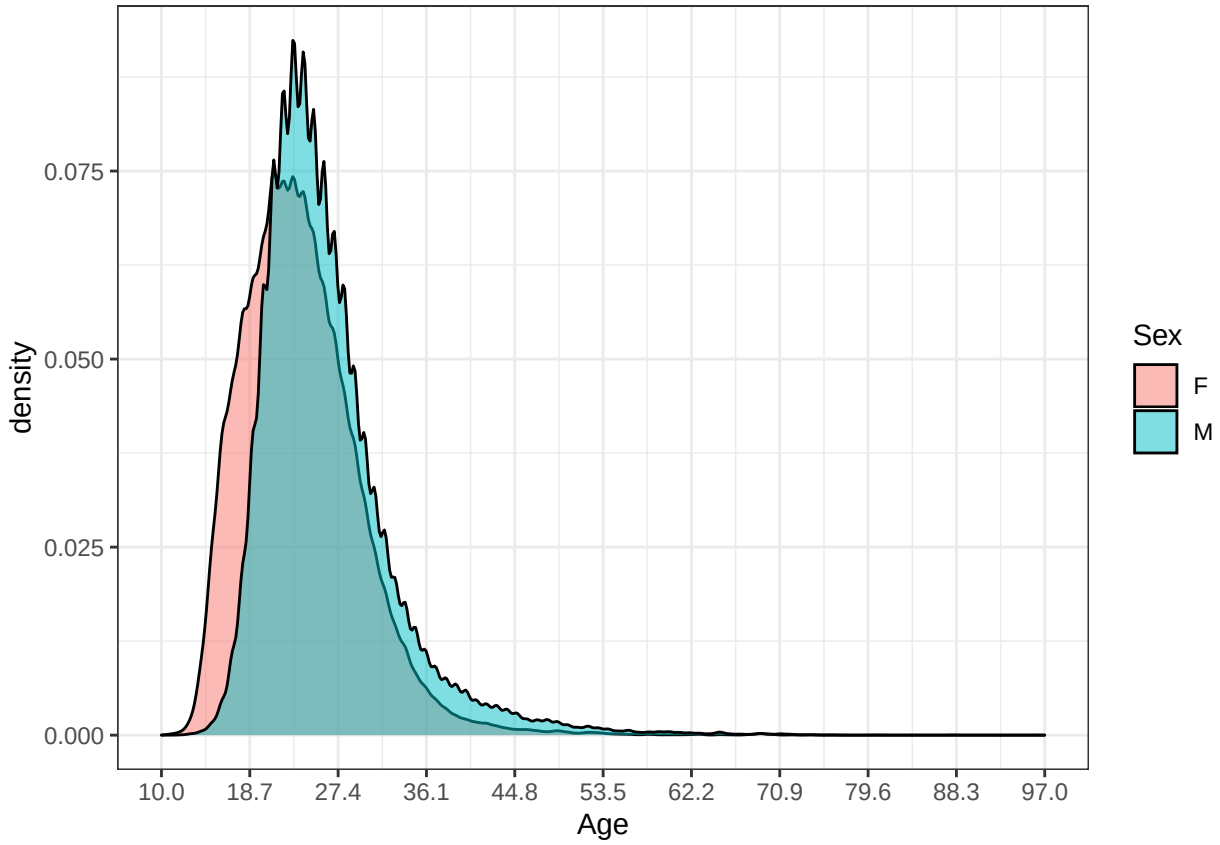
```
t_height <- tidy(t_height)[c("statistic", "p.value", "parameter", "method", "alternative")]
names(t_height) <- c("Statistic", "P Value", "Degrees of Freedom", "Method", "Alternative Hypothesis")
t_height
```

```
## # A tibble: 1 x 5
##   Statistic `P Value` `Degrees of Freedom` Method                  `Alternative Hy~
##       <dbl>     <dbl>                <dbl> <chr>                   <chr>
## 1     -263.         0              139790. Welch Two Sample t-test two.sided
```

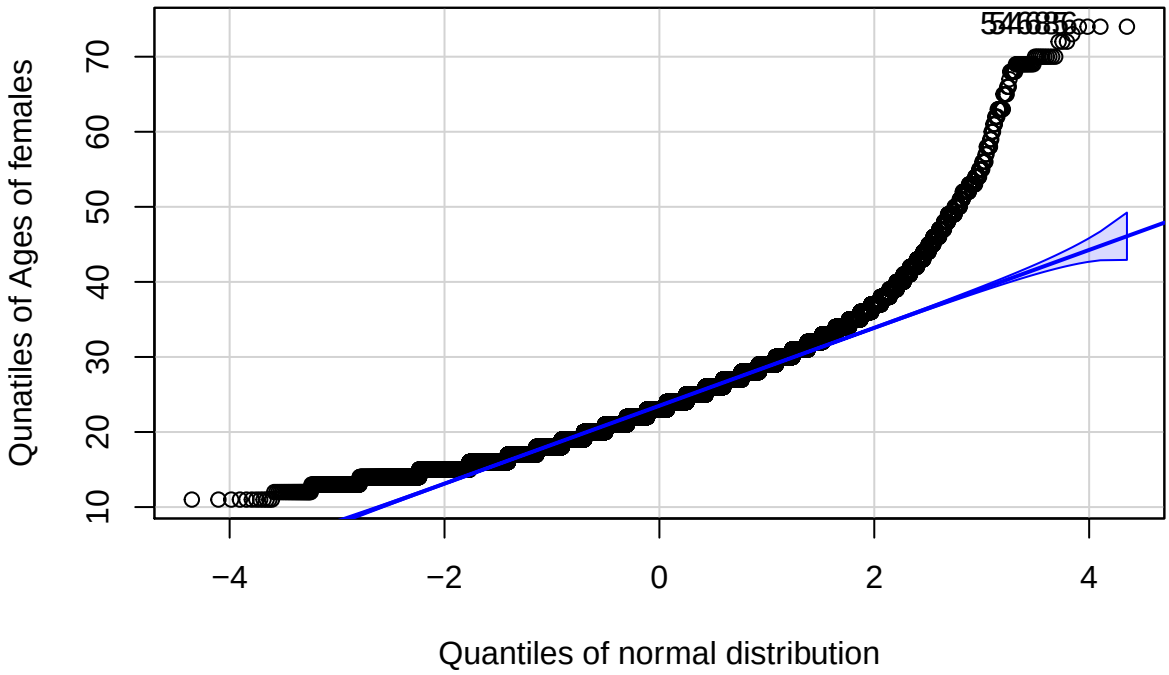
Description as in article: The height distribution of females and males participating in Olympic games were compared. As the height was normally distributed, two-sided Welch Two Sample t-test was used and significant difference was discovered, with test statistic equal to -263.101237 and p-Value of 0.

Now let's do the same for age. Age is not a continuous variable and is not normally distributed, however the sample size is very large which is why t-test might still be a choice. Mann Whitney U Test was also run.

```
#Testing for normal distribution
ggplot(subset(df_ol, complete.cases(Sex)), aes(Age, fill = Sex))+
  geom_density(alpha = 0.5)+
  scale_x_continuous(name = "Age",
                     limits = c(min(df_ol$Age, na.rm = T), max(df_ol$Age, na.rm = T)),
                     breaks = seq(min(df_ol$Age, na.rm = T), max(df_ol$Age, na.rm = T),
                                  (max(df_ol$Age, na.rm=T)-min(df_ol$Age, na.rm = T))/10)) +
  scale_fill_discrete(name = "Sex")+
  theme_bw()
```

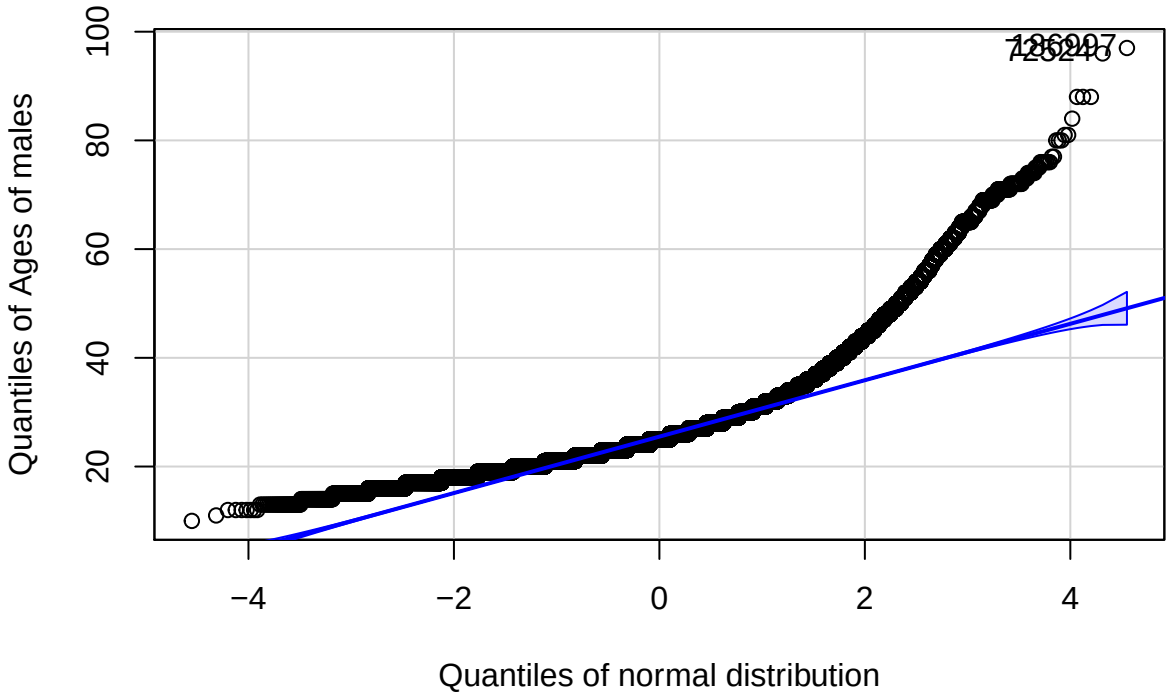



```
#QQplot
qqPlot(filter(df_ol, Sex == "F")$Age, xlab = "Quantiles of normal distribution", ylab = "Qunatiles of Ag
```




```
## [1] 54685 54686
```

```
qqPlot(filter(df_ol, Sex == "M")$Age, xlab = "Quantiles of normal distribution", ylab = "Quantiles of Ag
```




```
## [1] 186997  72524
```

```
#Everything is not really normally distributed but we have very large dataset, which still enables us t

#t-test
t_age <- t.test(filter(df_ol, Sex == "F")$Age, filter(df_ol, Sex == "M")$Age, alternative = "two.sided"
t_age
```

Table 1: Comparing the height, weight, and age differences for females and males participating in the Olympics.

Variable	Statistic	P Value	Deg. of Freed.	Method	Alt. Hypoth.
Weight	−297.3578	0	165260.5	Welch Two Sample t-test	Two Sided
Height	−263.1012	0	139790.5	Welch Two Sample t-test	Two Sided
Age	−97.81495	0	150732.7	Welch Two Sample t-test	Two Sided

```
##
##  Welch Two Sample t-test
##
## data:  filter(df_ol, Sex == "F")$Age and filter(df_ol, Sex == "M")$Age
## t = -97.815, df = 150733, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.595677 -2.493698
## sample estimates:
## mean of x mean of y
##  23.73284  26.27753
```

```
t_age <- tidy(t_age)[c("statistic", "p.value", "parameter", "method", "alternative")]
names(t_age) <- c("Statistic", "P Value", "Degrees of Freedom", "Method", "Alternative Hypothesis")
t_age
```

```
## # A tibble: 1 x 5
##   Statistic `P Value` `Degrees of Freedom` Method                  `Alternative Hy~
##       <dbl>     <dbl>                <dbl> <chr>                   <chr>
## 1     -97.8         0              150733. Welch Two Sample t-test two.sided
```

```
#Mann-Whitney U test
wlic <- wilcox.test(filter(df_ol, Sex == "F")$Age, filter(df_ol, Sex == "M")$Age)
wlic <- tidy(wlic)
names(wlic) <- c("Statistic", "P Value", "Method", "Alternative Hypothesis")
wlic
```

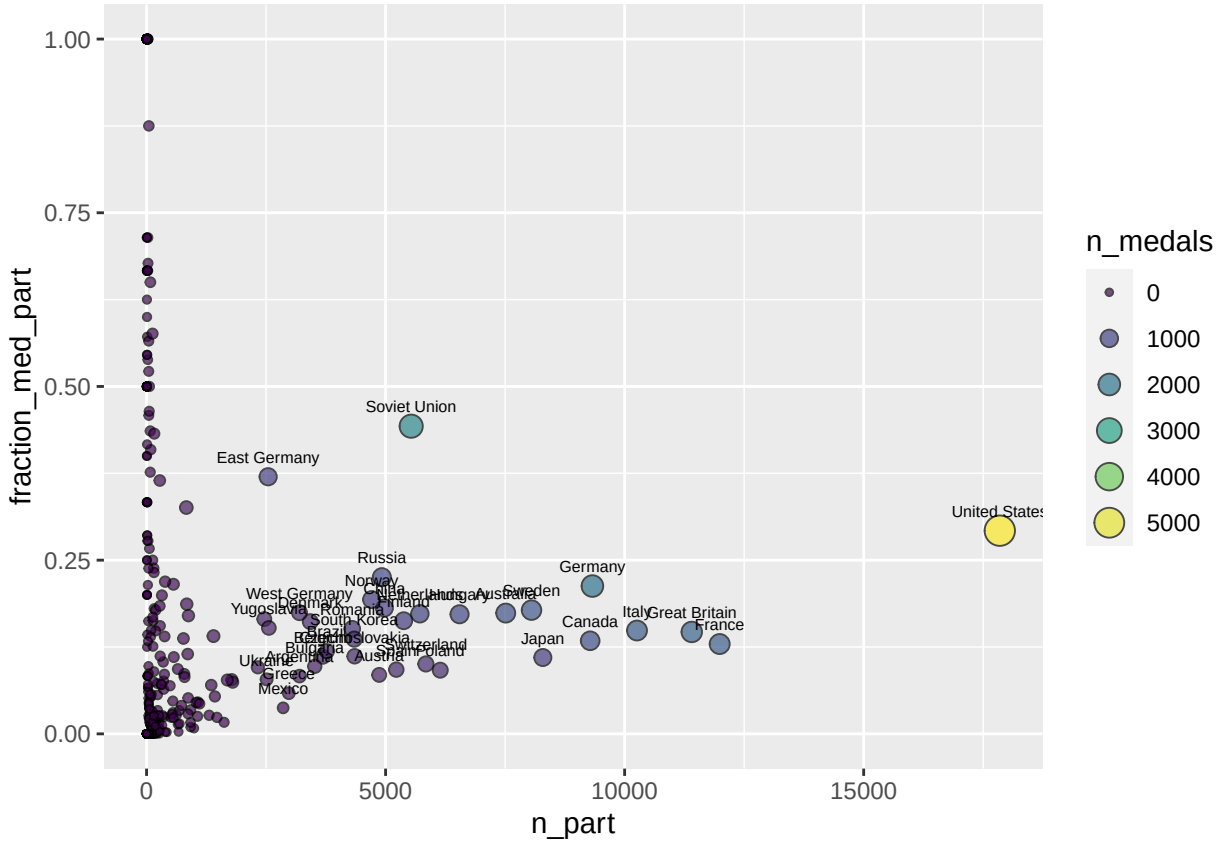
```
## # A tibble: 1 x 4
##     Statistic `P Value` Method                                          `Alternative Hy~
##         <dbl>     <dbl> <chr>                                           <chr>
## 1 5295764730.         0 Wilcoxon rank sum test with continuity correction two.sided
```

Description as in article: The age distribution of females and males participating in Olympic games were compared. As the sample size was very large, two-sided Welch Two Sample t-test was used despite the fact that some of the criteria were not fulfilled, and significant difference was discovered, with test statistic equal to -97.8149499 and p-Value of 0. Also Mann-Whitney test was run with test statistic equal to 5.2957647×10^9 and p-Value of 0.

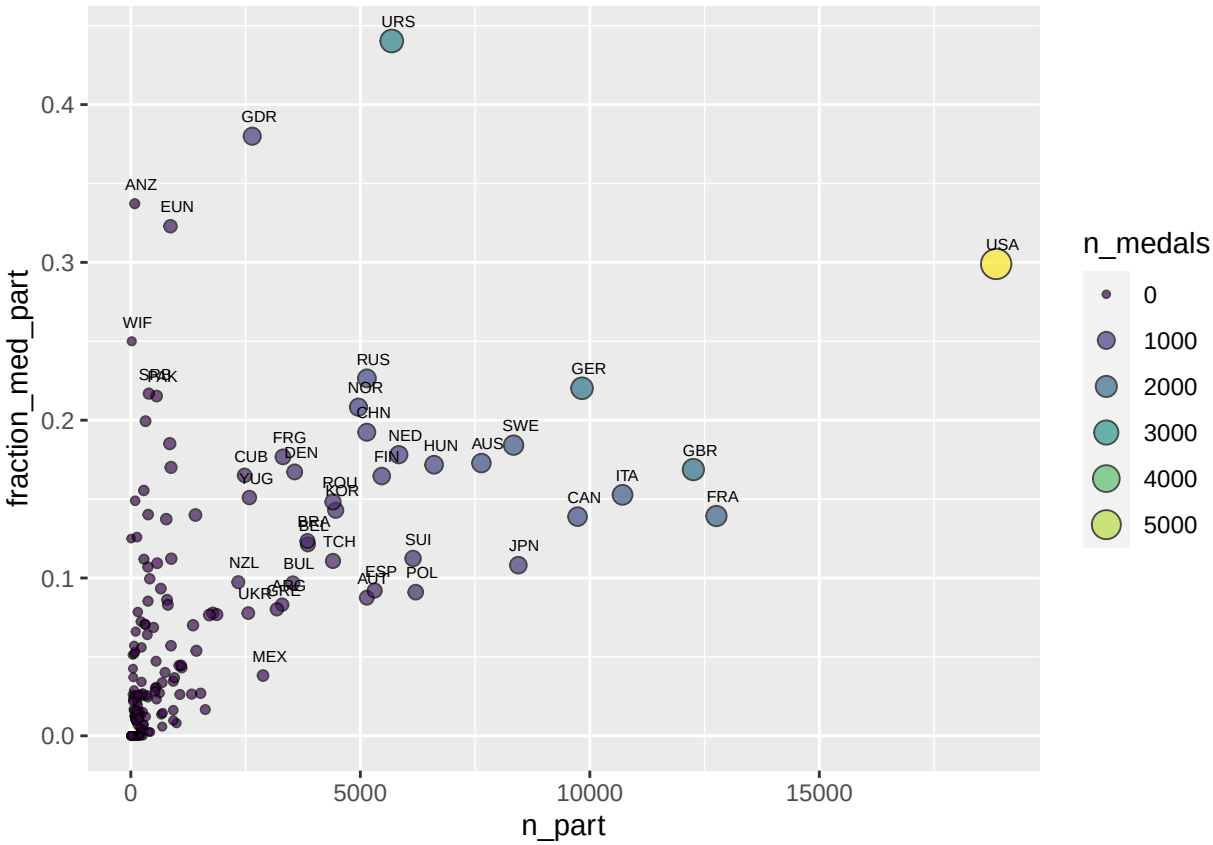
15. Relationship between Team and Medal.

```
df_fract_Team <- df_ol %>% mutate(ex_medal = ifelse(!is.na(Medal), 1, 0)) %>%
  group_by(Team) %>%
  select(ex_medal, Team) %>%
  summarize(n_medals = sum(ex_medal),
            n_part = n(),
            fraction_med_part = sum(ex_medal)/n())
```

```
ggplot(df_fract_Team, aes(x = n_part, y = fraction_med_part, group =n_medals))+
  geom_point(aes(fill=n_medals, size=n_medals), shape = 21, alpha = 0.7)+
  geom_text(aes(label=ifelse(n_part > 2500, as.character(Team),'')),hjust=0.5,vjust=-1.2, size = 2)+
  scale_fill_viridis_c(guide = "legend") +
  scale_size_continuous(range = c(1, 5))
```



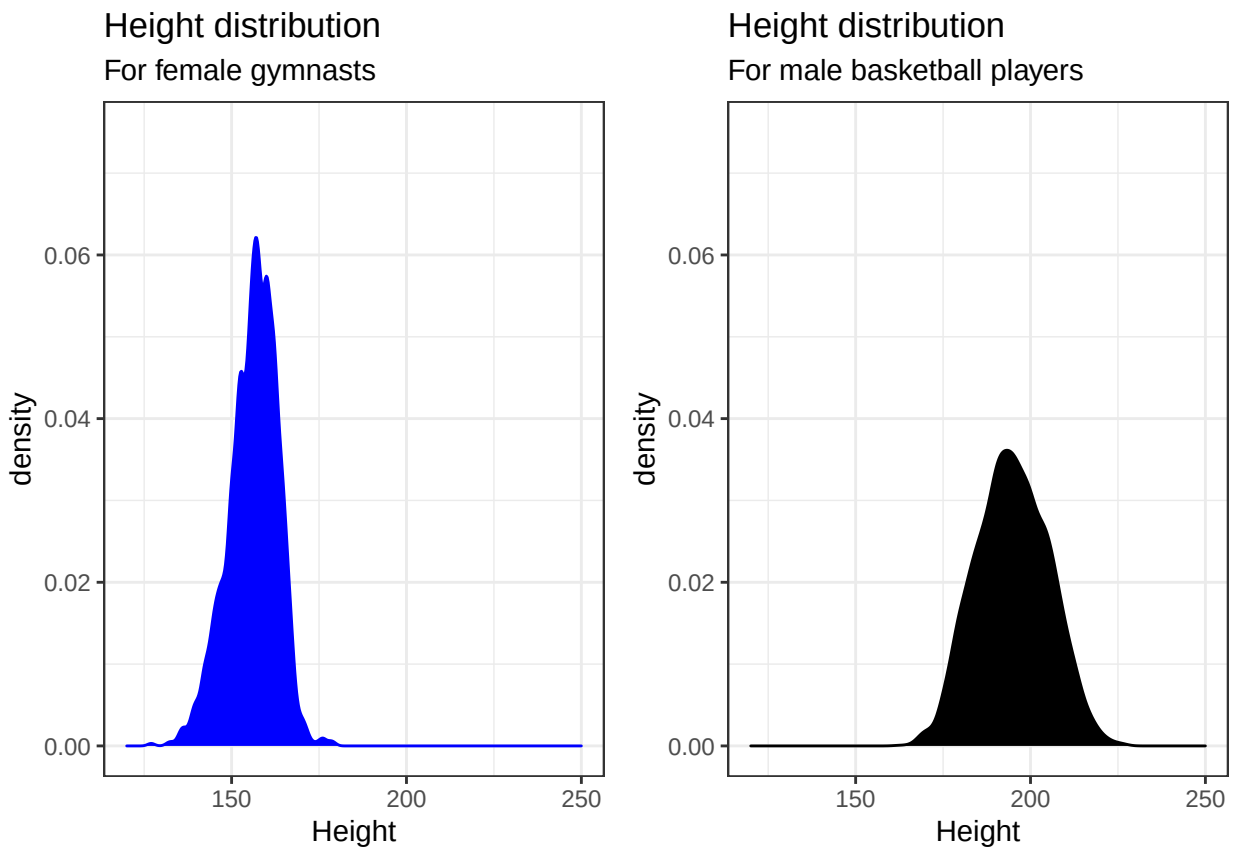

```
df_fract_NOC <- df_ol %>% mutate(ex_medal = ifelse(!is.na(Medal), 1, 0)) %>%
  group_by(NOC) %>%
  select(ex_medal, NOC) %>%
  summarize(n_medals = sum(ex_medal),
            n_part = n(),
            fraction_med_part = sum(ex_medal)/n())

#Due to the fact that there are a lot of teams and countries, I'm just going to highlight some of them

ggplot(df_fract_NOC, aes(x = n_part, y = fraction_med_part, group =n_medals))+
  geom_point(aes(fill=n_medals, size=n_medals), shape = 21, alpha = 0.7)+
  geom_text(aes(label=ifelse(fraction_med_part>0.2 | n_part > 2000, as.character(NOC),'')),hjust=0.3,vj
  scale_fill_viridis_c(guide = "legend") +
  scale_size_continuous(range = c(1, 5))
```



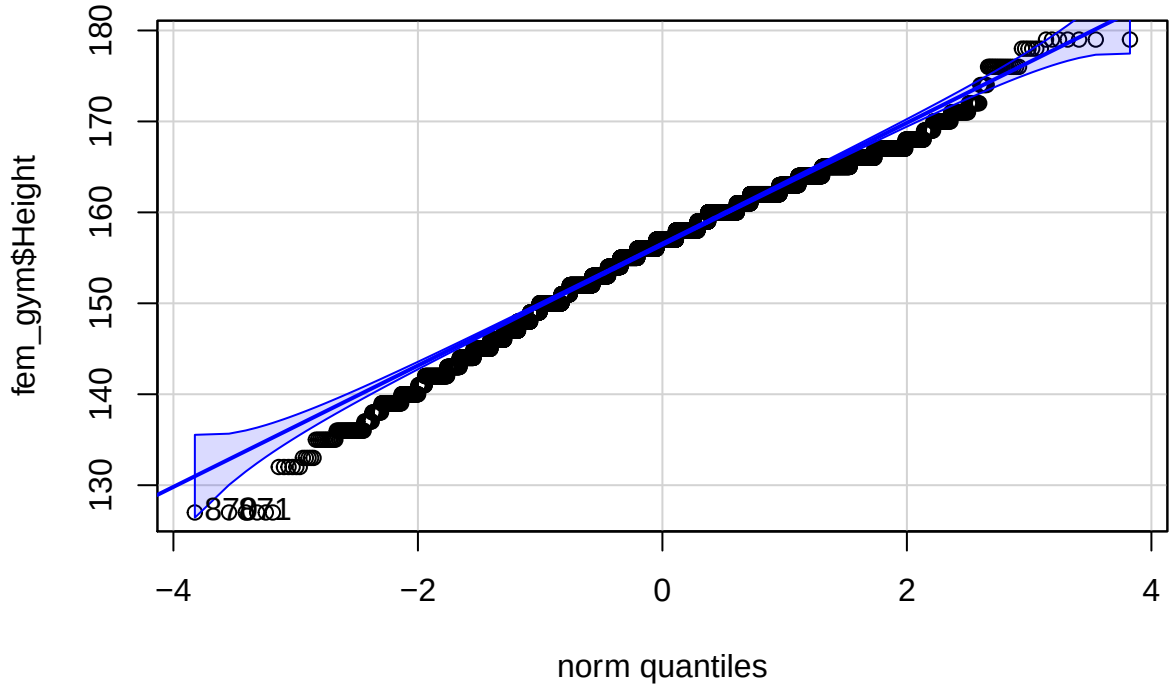
16. Compare some statistics - be creative!

Is the mean height of sportswomen competing in gymnastics, different from the height of male competing in basketball? Hmmmmm...

```
#Testing for normal distribution
fem_gym <- df_ol %>%
  filter(Sex %in% c("F"), Sport %in% c("Gymnastics"))

m_bask<- df_ol %>%
  filter(Sex %in% c("M"), Sport %in% c("Basketball"))

pfem_gym <- ggplot(fem_gym, aes(Height))+
  geom_density(color = "Blue", fill = "Blue")+
  labs(title = "Height distribution", subtitle = "For female gymnasts")+
  scale_x_continuous(name = "Height",
                     limits = c(120, 250)) +
  scale_y_continuous(limits = c(0, 0.075)) +
  theme_bw()
pm_bask <- ggplot(m_bask, aes(Height))+
  geom_density(fill = "Black")+
  labs(title = "Height distribution", subtitle = "For male basketball players")+
  scale_x_continuous(name = "Height",
                     limits = c(120, 250)) +
  scale_y_continuous(limits = c(0, 0.075)) +
  theme_bw()
plot_grid(pfem_gym, pm_bask)
```

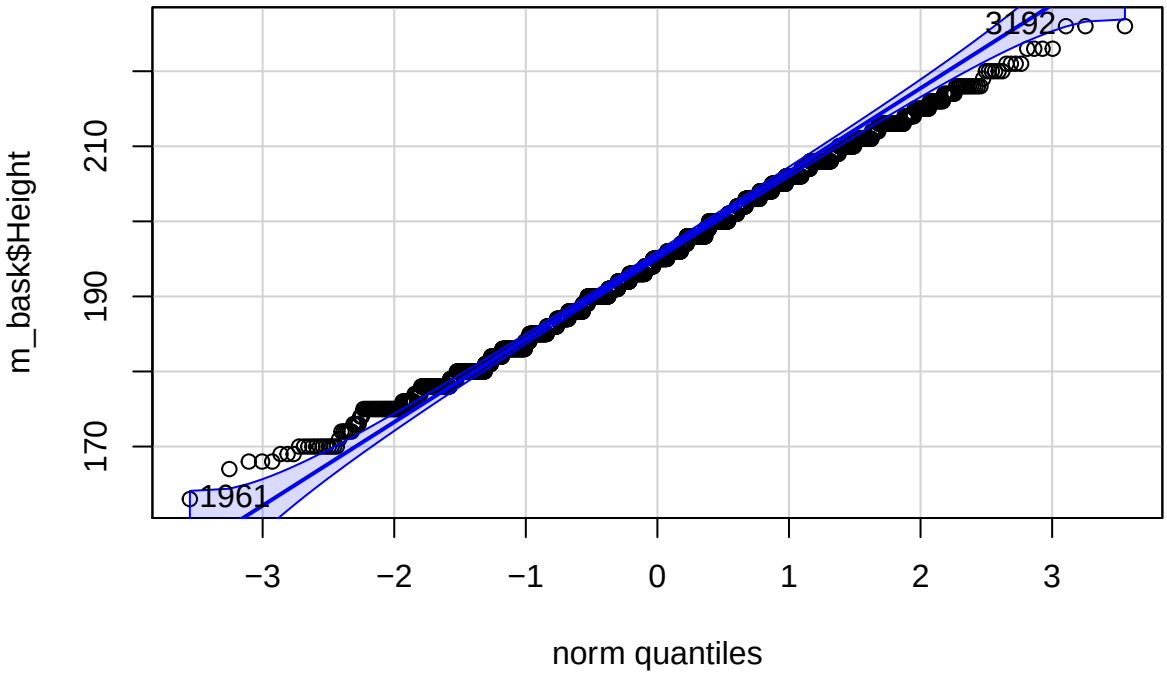



```
#QQplot
qqPlot(fem_gym$Height)
```




```
## [1] 870 871
```

```
qqPlot(m_bask$Height)
```




```
## [1] 1961 3192
```

```
#Seems to be normally distributed


#t-test
t_gym_bask <- t.test(fem_gym$Height, m_bask$Height, alternative = "two.sided")
t_gym_bask <- tidy(t_gym_bask)[c("statistic", "p.value", "parameter", "method", "alternative")]
names(t_gym_bask) <- c("Statistic", "P Value", "Degrees of Freedom", "Method", "Alternative Hypothesis")
t_gym_bask
```

```
## # A tibble: 1 x 5
##   Statistic `P Value` `Degrees of Freedom` Method                  `Alternative Hy~
##       <dbl>     <dbl>                <dbl> <chr>                   <chr>
## 1     -179.         0                3482. Welch Two Sample t-test two.sided
```

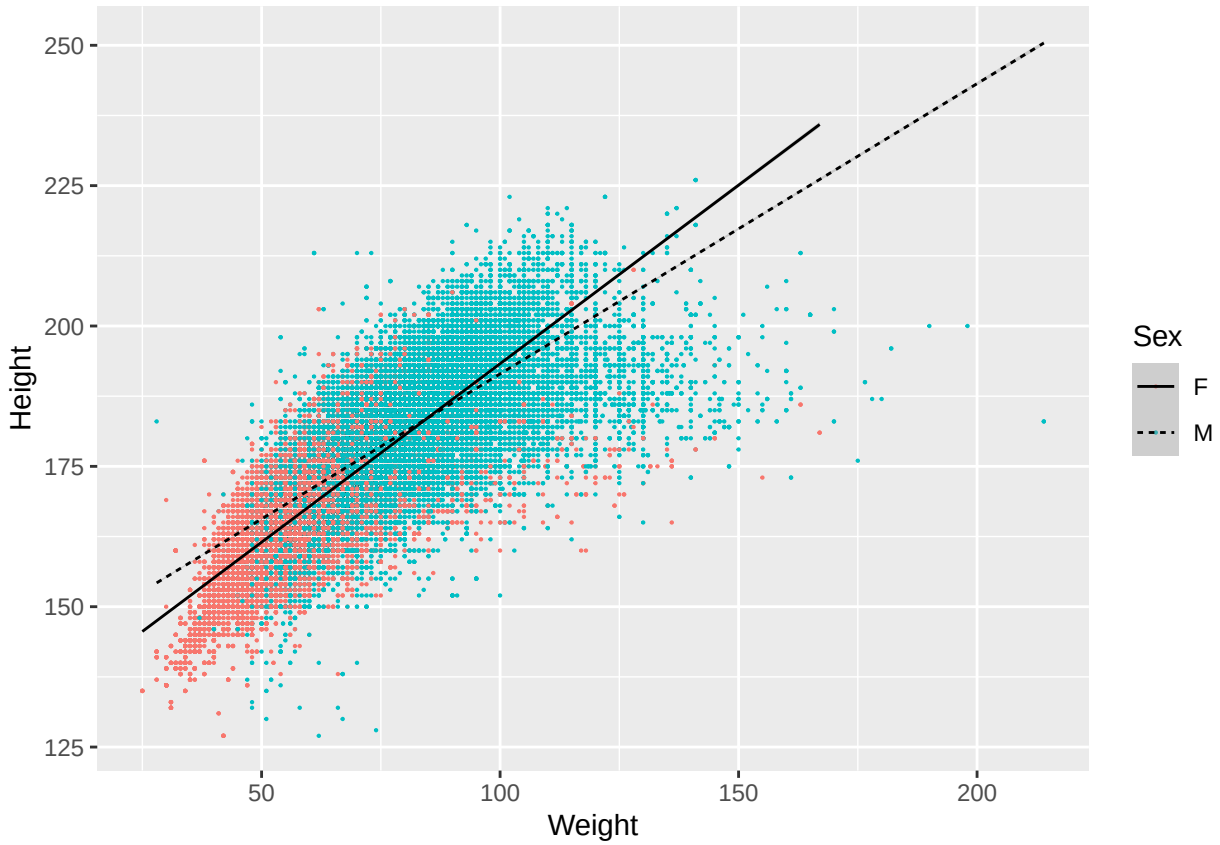
You're not going to believe this!!!

The height distribution of female gymnasts and males basketball players participating in Olympic games differ! As the height was normally distributed, two-sided Welch Two Sample t-test was used and significant difference was discovered, with test statistic equal to -178.9551871 and p-Value of 0.

Is weight and height correlated? How does the correlation differ between Sexes? Hmmmm...

```
#Let's see... We know that Height and Weight are both normally distributed.
ggplot(subset(df_ol, complete.cases(Height, Weight)), aes(Weight, Height))+
  geom_point(aes(color = Sex), size = 0.1)+
  geom_smooth(aes(linetype=Sex, color=Sex),
              color = "black",
              size=0.5,
              method = 'lm')
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
fem <- filter(df_ol, df_ol$Sex %in% c("F"))
m <- filter(df_ol, df_ol$Sex %in% c("M"))


c_m <- cor.test(m$Weight, m$Height)
c_fem <- cor.test(fem$Weight, fem$Height)

c_m <- tidy(c_m)[c("estimate", "statistic", "p.value",
                   "conf.low", "conf.high", "method", "alternative")]
names(c_m) <- c("Correlation coefficient estimate", "Statistic", "P value",
                "Lower conf.int.", "Upper conf.int.", "Method", "Alternative Hypothesis")
c_m
```

```
## # A tibble: 1 x 7
##   `Correlation coe~ Statistic `P value` `Lower conf.int~ `Upper conf.int~ Method
##               <dbl>     <dbl>     <dbl>            <dbl>            <dbl> <chr>
## 1             0.727      396.         0            0.724            0.729 Pears~
## # ... with 1 more variable: Alternative Hypothesis <chr>
```

```
c_fem <- tidy(c_fem)[c("estimate", "statistic", "p.value",
                   "conf.low", "conf.high", "method", "alternative")]
names(c_fem) <- c("Correlation coefficient estimate", "Statistic", "P value",
                "Lower conf.int.", "Upper conf.int.", "Method", "Alternative Hypothesis")
c_fem
```

```
## # A tibble: 1 x 7
##   `Correlation coe~ Statistic `P value` `Lower conf.int~ `Upper conf.int~ Method
##               <dbl>     <dbl>     <dbl>            <dbl>            <dbl> <chr>
```

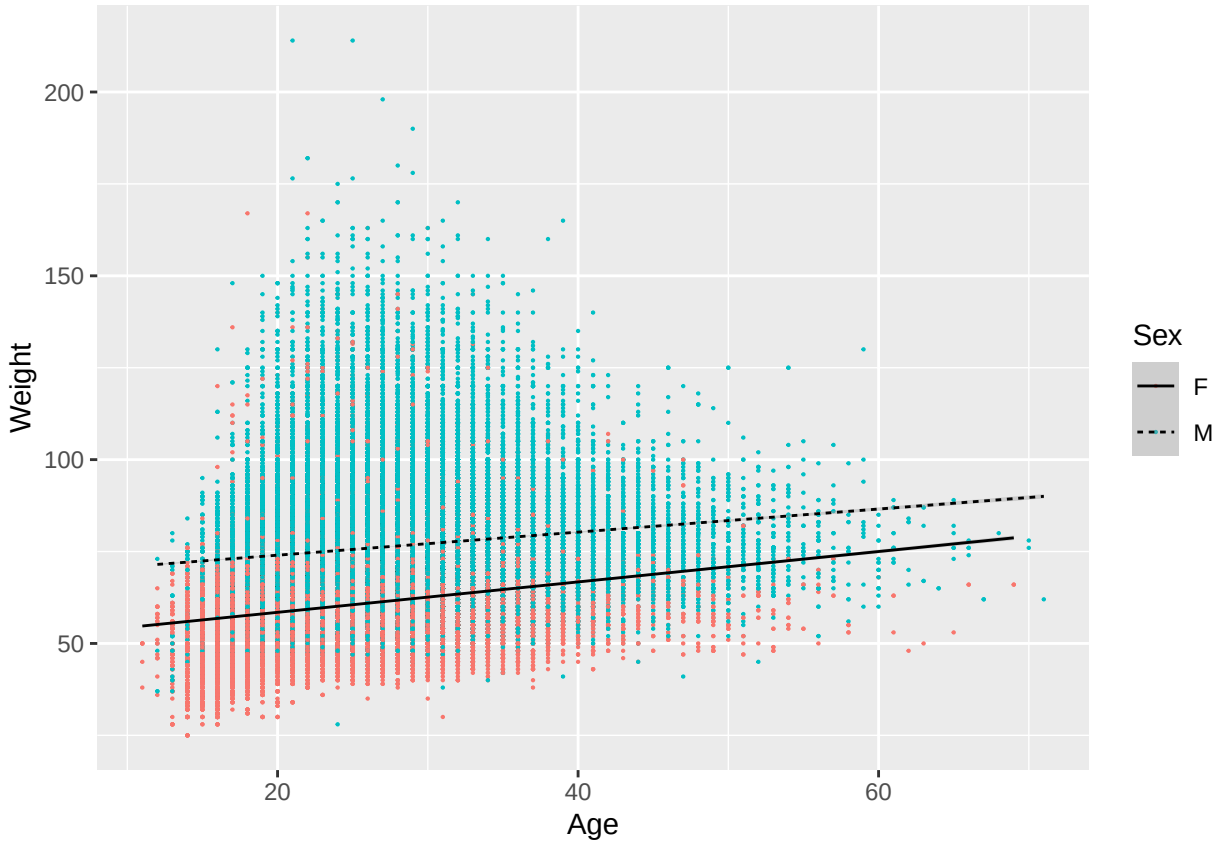
```
## 1             0.740      284.         0            0.737            0.744 Pears~
## # ... with 1 more variable: Alternative Hypothesis <chr>
```

Correlation coefficient estimate for males is 0.7269577 with lower confidence interval of 0.7244793 and upper confidence interval of 0.7294172. For females coefficient estimate for males is 0.7401129 with lower confidence interval of 0.7366623 and upper confidence interval of 0.743525. For both sexes there is positive correlation between height and weight, however for women it seems to be a little bit more pronounced.

Do people get fatter when they age? Let's compare age and weight for corresponding ages. There are some people that participated in the Olympics many times, and might skew the data a little bit - but due to the fact that the sample size is very large and the number of these people is not as high, we can not filter them off.

```
#Let's see... We know that Age is (nearly) normally distributed and weight is normally distributed.
ggplot(subset(df_ol, complete.cases(Age, Weight)), aes(Age, Weight))+
  geom_point(aes(color = Sex), size = 0.1)+
  geom_smooth(aes(linetype=Sex, color=Sex),
              color = "black",
              size=0.5,
              method = 'lm')
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
fem <- filter(df_ol, df_ol$Sex %in% c("F"))
m <- filter(df_ol, df_ol$Sex %in% c("M"))

c_m_age <- cor.test(m$Weight, m$Age)
c_fem_age <- cor.test(fem$Weight, fem$Age)
```

```
c_m_age <- tidy(c_m_age)[c("estimate", "statistic", "p.value",
                  "conf.low", "conf.high", "method", "alternative")]
names(c_m_age) <- c("Correlation coefficient estimate", "Statistic", "P value",
                "Lower conf.int.", "Upper conf.int.", "Method", "Alternative Hypothesis")
c_m_age
```

```
## # A tibble: 1 x 7
##   `Correlation coe~ Statistic `P value` `Lower conf.int~ `Upper conf.int~ Method
##               <dbl>     <dbl>     <dbl>            <dbl>            <dbl> <chr>
## 1             0.127      48.1         0            0.122            0.132 Pears~
## # ... with 1 more variable: Alternative Hypothesis <chr>
```

```
c_fem_age <- tidy(c_fem_age)[c("estimate", "statistic", "p.value",
                  "conf.low", "conf.high", "method", "alternative")]
names(c_fem_age) <- c("Correlation coefficient estimate", "Statistic", "P value",
                "Lower conf.int.", "Upper conf.int.", "Method", "Alternative Hypothesis")
c_fem_age
```

```
## # A tibble: 1 x 7
##   `Correlation coe~ Statistic `P value` `Lower conf.int~ `Upper conf.int~ Method
##               <dbl>     <dbl>     <dbl>            <dbl>            <dbl> <chr>
## 1             0.225      59.5         0            0.217            0.232 Pears~
## # ... with 1 more variable: Alternative Hypothesis <chr>
```

There is positive correlation between age and weight for both females and males, however surprisingly this correlation is much higher for females. Correlation coefficient estimate for males is 0.1272387 with lower confidence interval of 0.1220933 and upper confidence interval of 0.1323773. For females coefficient estimate for males is 0.2245396 with lower confidence interval of 0.2173239 and upper confidence interval of 0.2317309. For both sexes there is positive correlation between height and weight, however for women it seems to be a little bit more pronounced. This might be explained with some female-popular sports, like gymnastics, that require lower weight and height and is primarily represented by younger sportswomen.

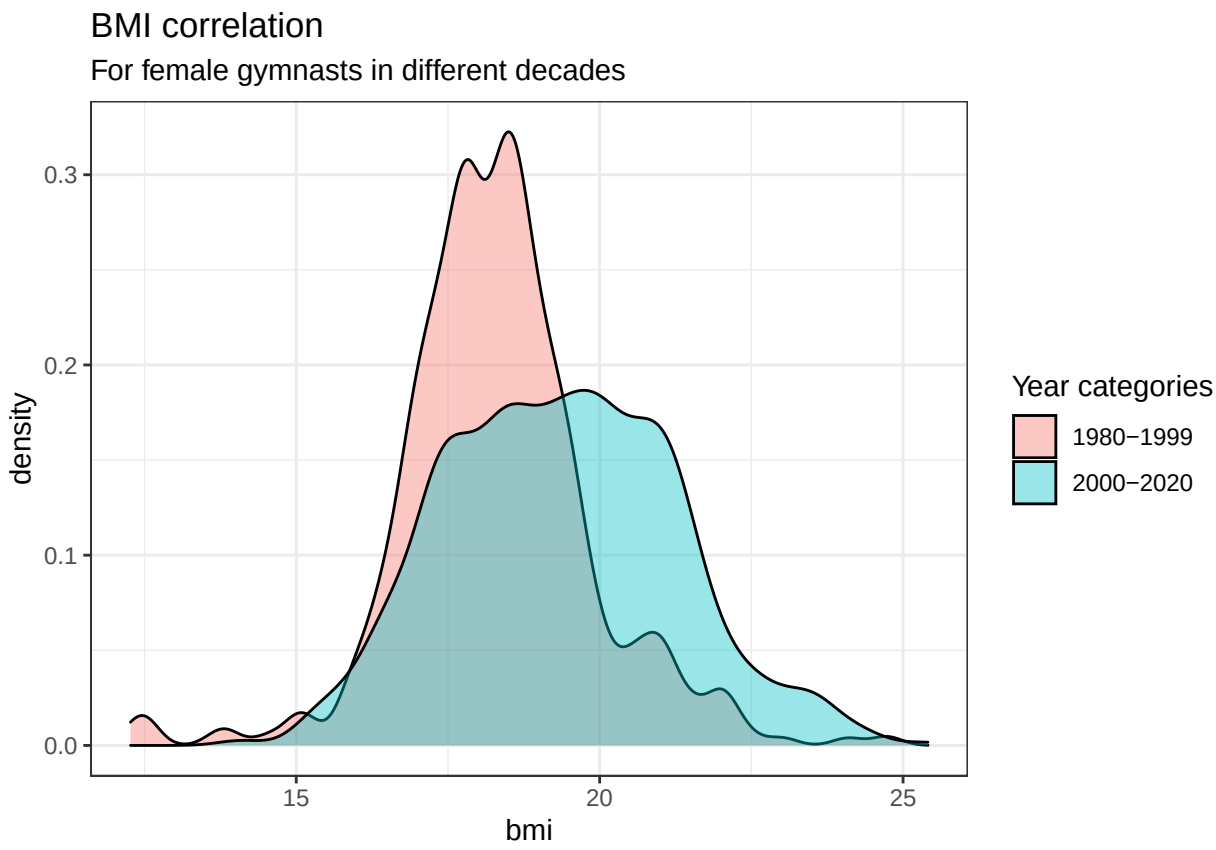
Did gymnasts get bulkier in this century? Let's group the gymnasts from last century (80s and 90s) and this one, and take a look at their BMI (formula from https://www.cdc.gov/nccdphp/dnpao/growthcharts/training/bmiage/page5_1.html).

```
df_ol_bmi <- df_ol %>%
  mutate(Year_category = case_when(Year >= 1980  & Year < 2000 ~ '1980-1999',
                           Year >= 2000  & Year <= 2020 ~ '2000-2020',)) %>%
  filter(Sex %in% c("F"), Sport %in% c("Gymnastics"), complete.cases(Year_category)) %>%
  mutate(bmi = Weight/Height^2*10000)

df_ol_bmi$Year_category <- factor(df_ol_bmi$Year_category)
str(df_ol_bmi)
```

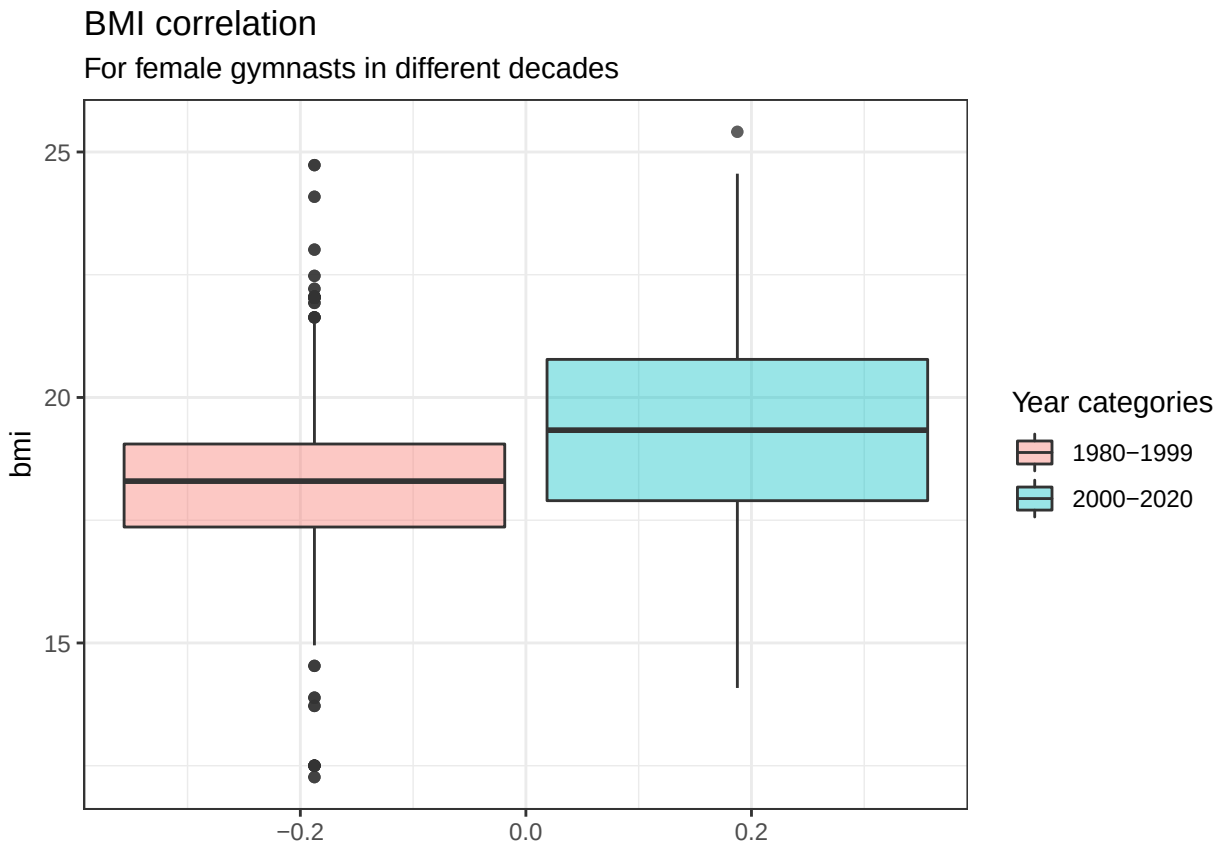
```
## 'data.frame':    4575 obs. of  17 variables:
##  $ ID           : int  396 396 396 396 396 627 627 627 627 976 ...
##  $ Name         : chr  "Katja Abel" "Katja Abel" "Katja Abel" "Katja Abel" ...
##  $ Sex          : Factor w/ 2 levels "F","M": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Age          : num  25 25 25 25 25 16 16 16 16 17 ...
##  $ Height       : num  165 165 165 165 165 150 150 150 150 153 ...
##  $ Weight       : num  55 55 55 55 55 40 40 40 40 38 ...
##  $ Team         : chr  "Germany" "Germany" "Germany" "Germany" ...
##  $ NOC          : chr  "GER" "GER" "GER" "GER" ...
##  $ Games        : chr  "2008 Summer" "2008 Summer" "2008 Summer" "2008 Summer" ...
##  $ Year         : int  2008 2008 2008 2008 2008 2008 2008 2008 2008 2012 ...
```

```
##  $ Season       : Factor w/ 2 levels "Summer","Winter": 1 1 1 1 1 1 1 1 1 1 ...
##  $ City         : chr  "Beijing" "Beijing" "Beijing" "Beijing" ...
##  $ Sport        : chr  "Gymnastics" "Gymnastics" "Gymnastics" "Gymnastics" ...
##  $ Event        : chr  "Gymnastics Women's Individual All-Around" "Gymnastics Women's Team All-Aroun
##  $ Medal        : Factor w/ 3 levels "Bronze","Gold",..: NA NA NA NA NA NA 1 NA NA NA ...
##  $ Year_category: Factor w/ 2 levels "1980-1999","2000-2020": 2 2 2 2 2 2 2 2 2 2 ...
##  $ bmi          : num  20.2 20.2 20.2 20.2 20.2 ...
```

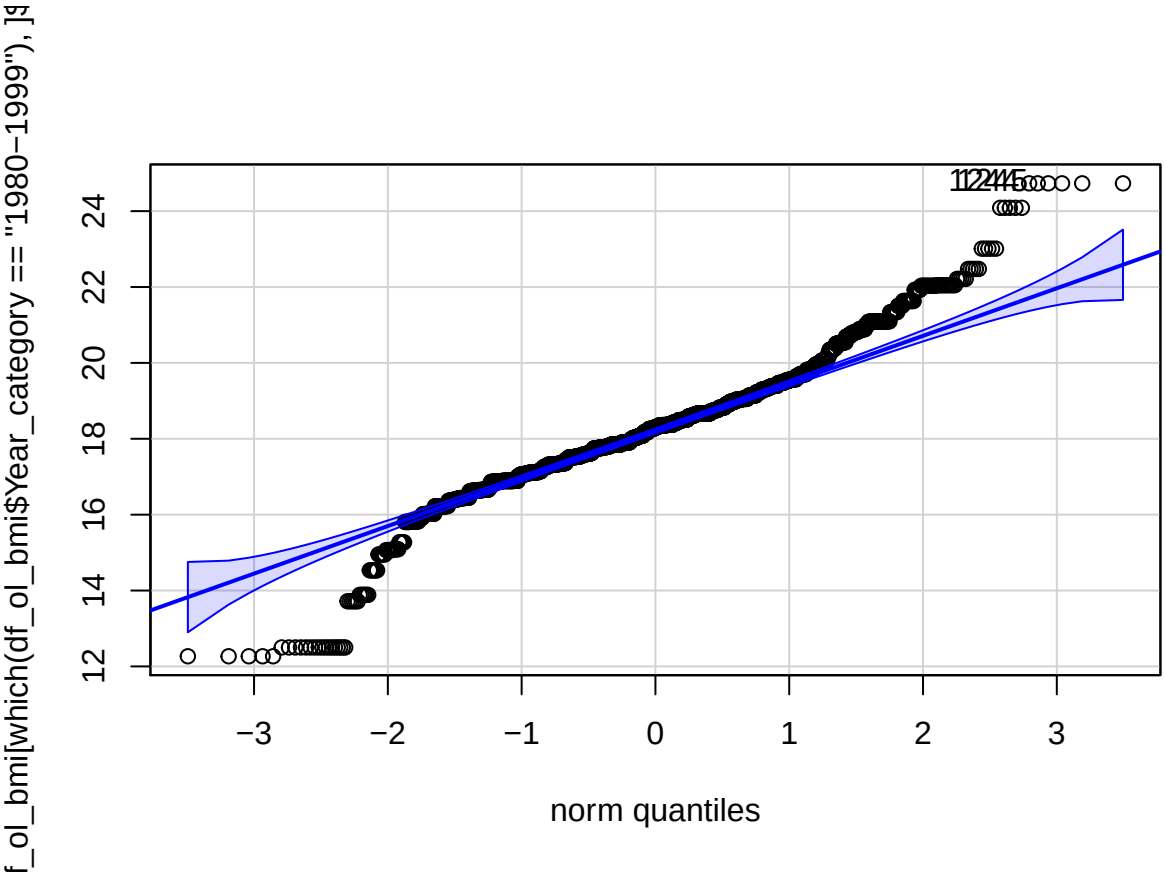
```
#Density distribution of bmi for the year categories
ggplot(df_ol_bmi, aes(x = bmi, fill = Year_category))+
  geom_density(alpha = 0.4) +
  labs(title = "BMI correlation", subtitle = "For female gymnasts in different decades") +
  theme_bw()+
  scale_fill_discrete(name = "Year categories")
```




```
#It is quite evident that the range of BMI has gotten larger and on higher end:
ggplot(df_ol_bmi, aes(y = bmi, fill = Year_category))+
  geom_boxplot(alpha = 0.4) +
  labs(title = "BMI correlation", subtitle = "For female gymnasts in different decades") +
  theme_bw()+
  scale_fill_discrete(name = "Year categories")
```

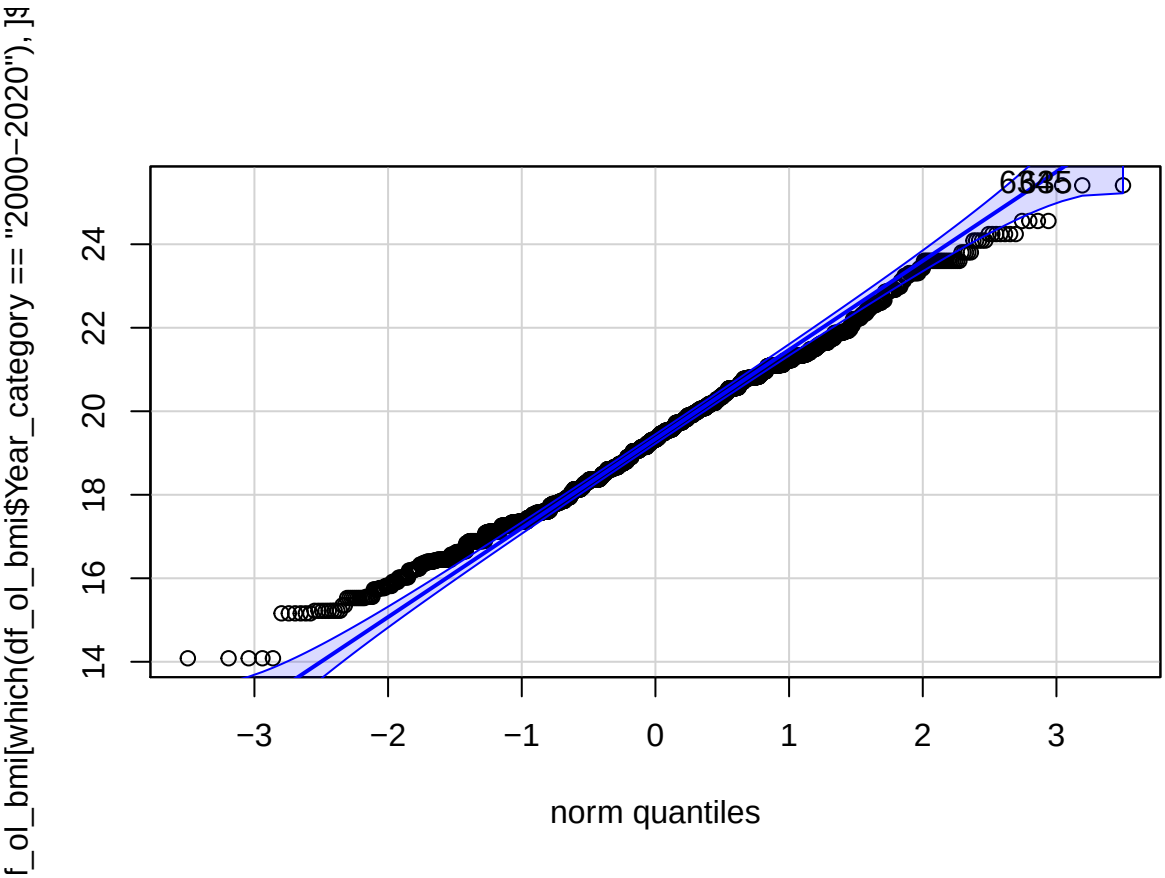



```
#QQPlot
qqPlot(df_ol_bmi[which(df_ol_bmi$Year_category == "1980-1999"), ]$bmi)
```




```
## [1] 1244 1245
```

```
qqPlot(df_ol_bmi[which(df_ol_bmi$Year_category == "2000-2020"), ]$bmi)
```




```
## [1] 634 635
```

```
#Checking for normal distribution:
shapiro.test(df_ol_bmi[which(df_ol_bmi$Year_category == "1980-1999"), ]$bmi)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df_ol_bmi[which(df_ol_bmi$Year_category == "1980-1999"), ]$bmi
## W = 0.95666, p-value < 2.2e-16
```

```
shapiro.test(df_ol_bmi[which(df_ol_bmi$Year_category == "2000-2020"), ]$bmi)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df_ol_bmi[which(df_ol_bmi$Year_category == "2000-2020"), ]$bmi
## W = 0.99457, p-value = 4.863e-07
```

```
#More or less normal distribution - let's do t-test
t_bmi <- t.test(df_ol_bmi[which(df_ol_bmi$Year_category == "1980-1999"), ]$bmi, df_ol_bmi[which(df_ol_bm
t_bmi <- tidy(t_bmi)[c("statistic", "p.value", "parameter", "method", "alternative")]
names(t_bmi) <- c("Statistic", "P Value", "Degrees of Freedom", "Method", "Alternative Hypothesis")
t_bmi
```

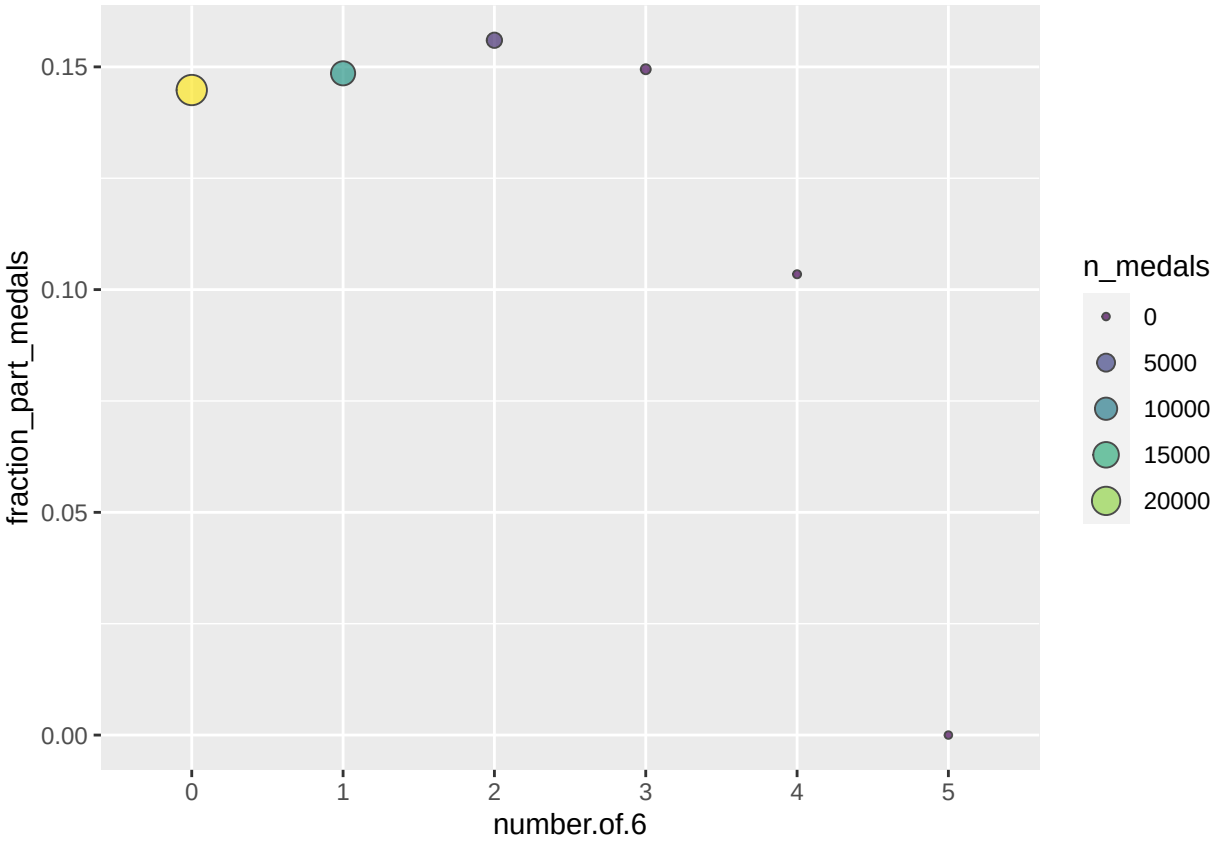
```
## # A tibble: 1 x 5
##   Statistic `P Value` `Degrees of Freedom` Method                  `Alternative Hy~
##       <dbl>     <dbl>                <dbl> <chr>                   <chr>
```

```
## 1     -19.7  8.20e-83                4127. Welch Two Sample t-test two.sided
```

As we can see there is significant difference between the BMIs of female gymnasts from 80s and 90s in comparison to the ones from this century. This confirms the alternative hypothesis that the gymnasts might have gotten "bulkier" on average - **DISCLAIMER** this enables sportswomen perform more complicated tricks and ensures more safety.

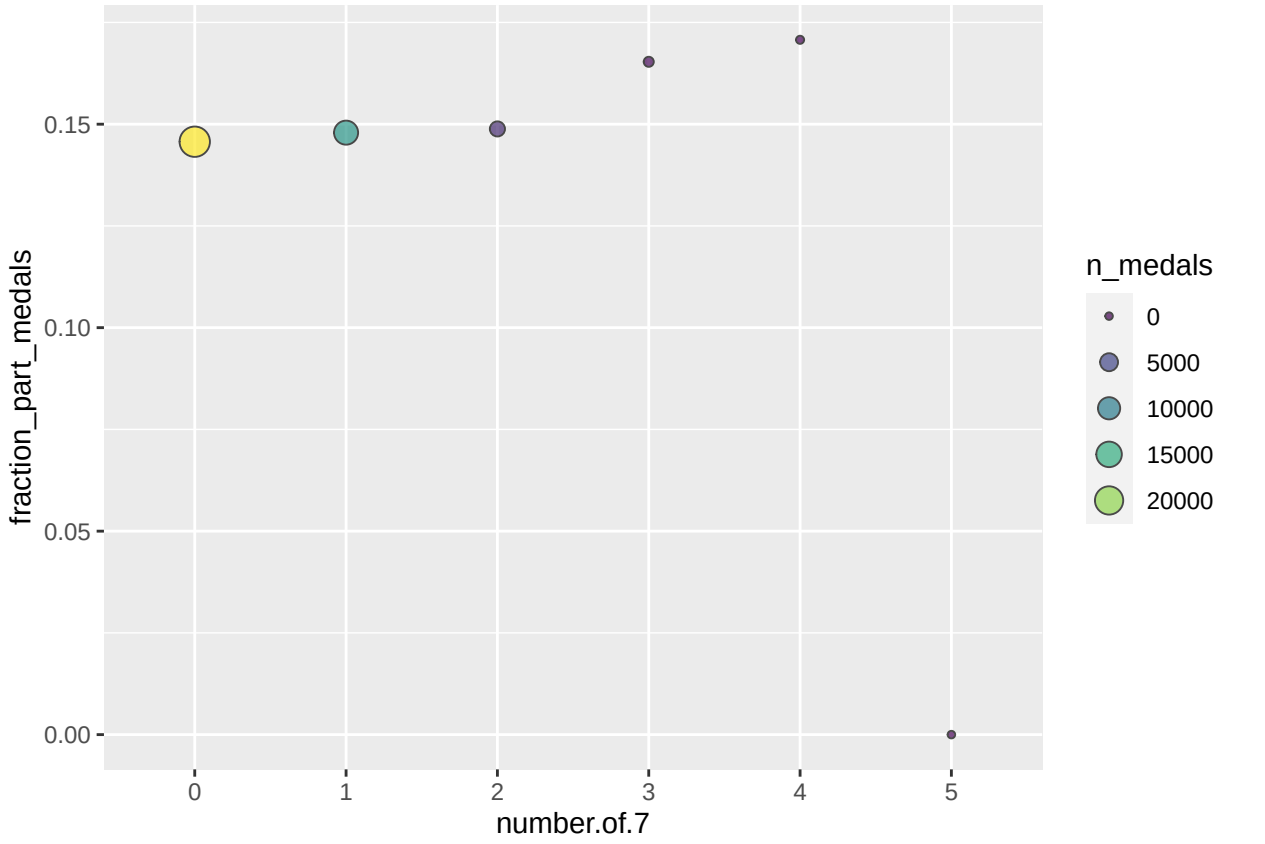
Lucky numbers - is the occurence of 6s or 7s in IDs maybe correlated with success/failure?

```
df_ol$number.of.6 <- str_count(df_ol$ID, "6")
df_ol$number.of.6 <- factor(df_ol$number.of.6)
df_ol_success6 <- df_ol %>% mutate(ex_medal = ifelse(!is.na(Medal), 1, 0)) %>%
  group_by(number.of.6) %>%
  select(number.of.6, ex_medal) %>%
  summarize(n_medals = sum(ex_medal),
            n_part = n(),
            fraction_part_medals = sum(ex_medal)/n())

ggplot(df_ol_success6, aes(x = number.of.6, y = fraction_part_medals, group = n_medals)) +
  geom_point(aes(fill=n_medals, size=n_medals), shape = 21, alpha = 0.7) +
  scale_fill_viridis_c(guide = "legend")+
  scale_size_continuous(range = c(1, 5))
```




```
df_ol$number.of.7 <- str_count(df_ol$ID, "7")
df_ol$number.of.7 <- factor(df_ol$number.of.7)
df_ol_success7 <- df_ol %>% mutate(ex_medal = ifelse(!is.na(Medal), 1, 0)) %>%
  group_by(number.of.7) %>%
  select(number.of.7, ex_medal) %>%
  summarize(n_medals = sum(ex_medal),
```

```
            n_part = n(),
            fraction_part_medals = sum(ex_medal)/n())

ggplot(df_ol_success7, aes(x = number.of.7, y = fraction_part_medals, group = n_medals)) +
  geom_point(aes(fill=n_medals, size=n_medals), shape = 21, alpha = 0.7) +
  scale_fill_viridis_c(guide = "legend")+
  scale_size_continuous(range = c(1, 5))
```




```
#We have 2 people with all 6 and 7s, which is why we have such hard outliers for the last points.
df_ol[which(df_ol$number.of.6 == 5 | df_ol$number.of.7 == 5),]
```

```
##           ID                      Name Sex Age Height Weight          Team NOC
## 132463 66666          Lupe Lara Quiala   M  24    181     82          Cuba CUB
## 132464 66666          Lupe Lara Quiala   M  27    181     82          Cuba CUB
## 154979 77777 Elaine Mary H. McLaughlin   F  24    167     58 Great Britain GBR
##              Games Year Season        City     Sport
## 132463 1968 Summer 1968 Summer Mexico City Wrestling
## 132464 1972 Summer 1972 Summer      Munich Wrestling
## 154979 1988 Summer 1988 Summer       Seoul Athletics
##                                        Event Medal number.of.6 number.of.7
## 132463 Wrestling Men's Middleweight, Freestyle  <NA>           5           0
## 132464 Wrestling Men's Middleweight, Freestyle  <NA>           5           0
## 154979    Athletics Women's 400 metres Hurdles  <NA>           0           5
```

The graph for 7a doesn't really look that bad. We can see that the ratio of 7s to the total participants having the medals to the total number of participants having the specific count of 7s increases with more 7s. Maybe it is lucky number after all..? Though making a pact with lucifer in order to get the medal might not be the best idea after all...