

Processing large datasets with R - exam: Exercise 2

Joris LIMONIER

```
library(dplyr)

##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##   filter, lag
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
library(ggplot2)
library(gridExtra)

##
## Attaching package: 'gridExtra'
## The following object is masked from 'package:dplyr':
##
##   combine
```

Part 1

Question 1a

```
winter <- read.csv("datasets_exam/winter_olympic.csv")
```

Question 1b

```
head(winter)
```

	Rank	NOC	Gold	Silver	Bronze	Total	Region
## 1	1	Russia (RUS)*	13	11	9	33	EURASIA
## 2	2	Norway (NOR)	11	5	10	26	EUROPE
## 3	3	Canada (CAN)	10	10	5	25	NORTH_A
## 4	4	United States (USA)	9	7	12	28	NORTH_A
## 5	5	Netherlands (NED)	8	7	9	24	EUROPE
## 6	6	Germany (GER)	8	6	5	19	EUROPE

Question 1c

```
colnames(winter)
```

```
## [1] "Rank" "NOC" "Gold" "Silver" "Bronze" "Total" "Region"
```

Question 1d

```
dim(winter)

## [1] 26 7

nrow(winter)

## [1] 26

ncol(winter)

## [1] 7
```

Part 2

```
sort_total <- winter %>% arrange(Total, NOC)
head(sort_total)
```

##	Rank		NOC	Gold	Silver	Bronze	Total	Region
## 1	25		Croatia (CRO)	0	1	0	1	EUROPE
## 2	26		Kazakhstan (KAZ)	0	0	1	1	EURASIA
## 3	21		Slovakia (SVK)	1	0	0	1	EUROPE
## 4	20		Ukraine (UKR)	1	0	1	2	EURASIA
## 5	24		Australia (AUS)	0	2	1	3	AUSTRALIA
## 6	19		Great Britain (GBR)	1	1	2	4	EUROPE

Part 3

```
print_stat <- function() {
  print(sum(is.na(sort_total)))
  print(summary(sort_total))
}
print_stat()
```

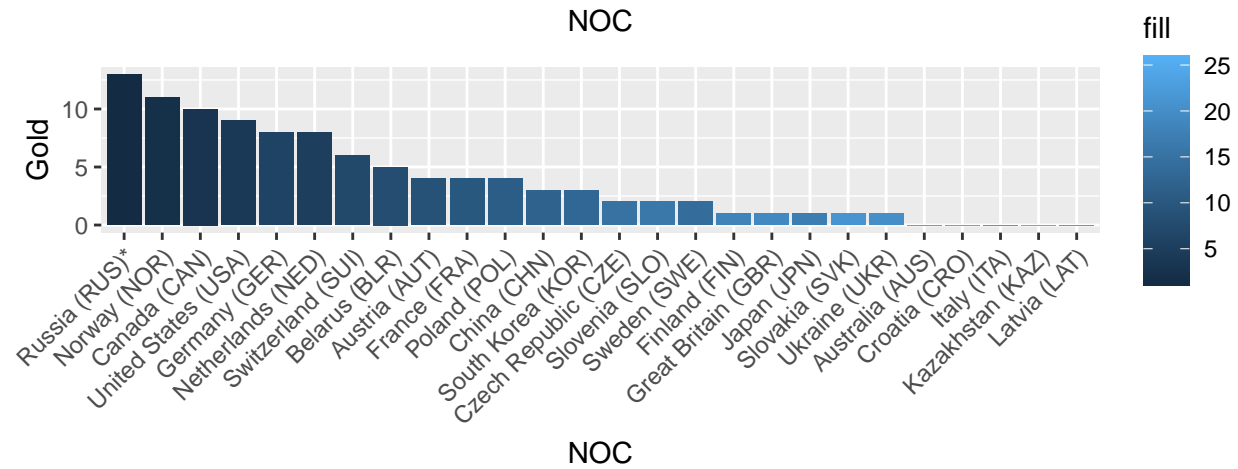
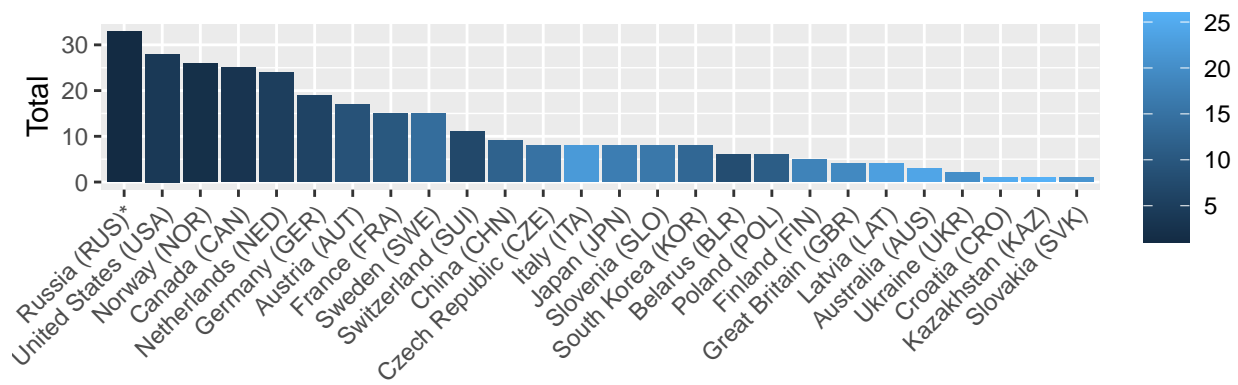
```
## [1] 0
##      Rank      NOC      Gold      Silver
## Min.   : 1.00   Length:26   Min.    : 0.000   Min.    : 0.000
## 1st Qu.: 7.25   Class :character   1st Qu.: 1.000   1st Qu.: 1.250
## Median :13.50   Mode  :character   Median : 2.500   Median : 3.000
## Mean   :13.50                   Mean   : 3.808   Mean   : 3.731
## 3rd Qu.:19.75                   3rd Qu.: 5.750   3rd Qu.: 5.750
## Max.   :26.00                   Max.    :13.000   Max.    :11.000
##      Bronze      Total      Region
## Min.   : 0.000   Min.    : 1.00   Length:26
## 1st Qu.: 1.000   1st Qu.: 4.25   Class :character
## Median : 2.000   Median : 8.00   Mode  :character
## Mean   : 3.808   Mean    :11.35
## 3rd Qu.: 5.750   3rd Qu.:16.50
## Max.   :12.000   Max.    :33.00
```

```
plot_desc <- function(
  x, y, fill=sort_total$Rank,
  x_label="NOC", y_label
){
  ggplot(sort_total, aes(reorder(x, -y, sum), y, fill=fill)) +
```

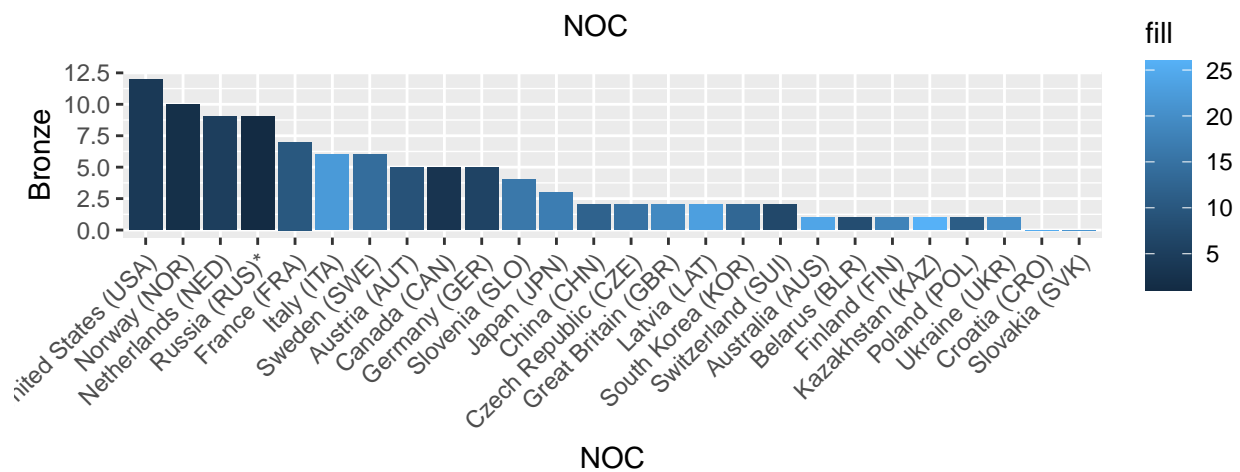
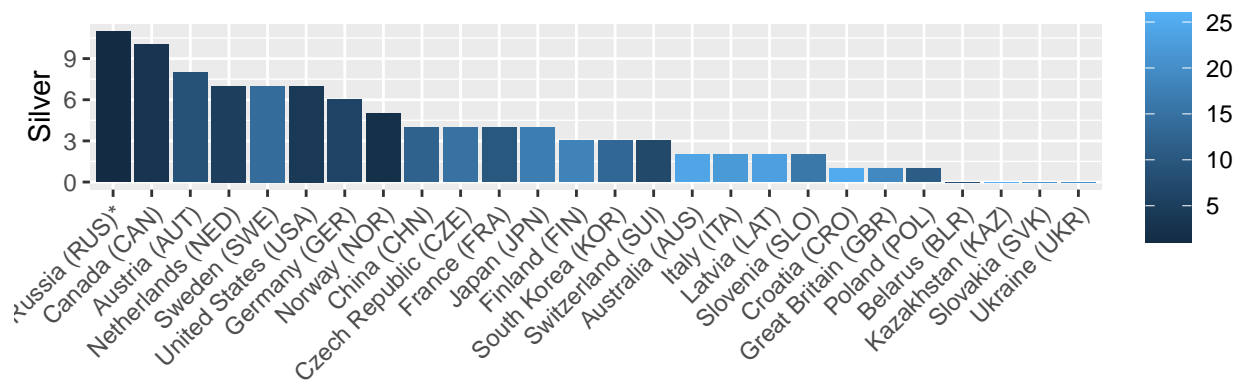
```

    geom_col() +
    scale_x_discrete(guide=guide_axis(angle=45)) +
    xlab(x_label) +
    ylab(y_label)
}
p_total <- plot_desc(
  sort_total$NOC,
  sort_total$Total,
  y_label="Total"
)
p_gold <- plot_desc(
  sort_total$NOC,
  sort_total$Gold,
  y_label="Gold"
)
p_silver <- plot_desc(
  sort_total$NOC,
  sort_total$Silver,
  y_label="Silver"
)
p_bronze <- plot_desc(
  sort_total$NOC,
  sort_total$Bronze,
  y_label="Bronze"
)
grid.arrange(p_total, p_gold, nrow=2)

```



```
grid.arrange(p_silver, p_bronze, nrow=2)
```



Part 4

Question 4a

```
for (column in c("Gold", "Silver", "Bronze", "Total")) {
  print(
    paste(
      column,
      "-> median:",
      median(sort_total[[column]])
    )
  )
}
```

```
## [1] "Gold -> median: 2.5"
## [1] "Silver -> median: 3"
## [1] "Bronze -> median: 2"
## [1] "Total -> median: 8"
```

Question 4b

```
for (column in c("Gold", "Silver", "Bronze", "Total")) {
  print(
    paste(
      column,
      "-> mean:",

```

```

        mean(sort_total[[column]])
    )
}

## [1] "Gold -> mean: 3.80769230769231"
## [1] "Silver -> mean: 3.73076923076923"
## [1] "Bronze -> mean: 3.80769230769231"
## [1] "Total -> mean: 11.3461538461538"

for (column in c("Gold", "Silver", "Bronze", "Total")) {
  print(
    paste(
      column,
      "-> total:",
      sum(sort_total[[column]])
    )
  )
}

## [1] "Gold -> total: 99"
## [1] "Silver -> total: 97"
## [1] "Bronze -> total: 99"
## [1] "Total -> total: 295"

```

Part 6

```

winter_group_region <- winter %>%
  group_by(Region)

```

Question 6a

```

winter_group_region <- winter %>%
  group_by(Region)

print("median:")

```

```
## [1] "median:"
```

```

winter_group_region %>%
  summarise(
    median(Gold),
    median(Silver),
    median(Bronze),
    median(Total)
  )

```

```
## # A tibble: 5 x 5
##   Region      `median(Gold)` `median(Silver)` `median(Bronze)` `median(Total)`
##   <chr>          <dbl>          <dbl>          <dbl>          <dbl>
## 1 ASIA              3              4              2              8
## 2 AUSTRALIA         0              2              1              3
## 3 EURASIA           1              0              1              4
## 4 EUROPE            2              3              4              8
## 5 NORTH_A          9.5            8.5            8.5            26.5

```

```
print("mean:")
```

```
## [1] "mean:"
```

```
winter_group_region %>%  
  summarise(  
    mean(Gold),  
    mean(Silver),  
    mean(Bronze),  
    mean(Total)  
  )
```

```
## # A tibble: 5 x 5  
##   Region      `mean(Gold)` `mean(Silver)` `mean(Bronze)` `mean(Total)`  
##   <chr>          <dbl>         <dbl>         <dbl>         <dbl>  
## 1 ASIA           2.33          3.67          2.33          8.33  
## 2 AUSTRALIA       0            2            1            3  
## 3 EURASIA         3.8          2.6          2.8          9.2  
## 4 EUROPE          3.6          3.6          4            11.2  
## 5 NORTH_A        9.5          8.5          8.5          26.5
```

```
print("total:")
```

```
## [1] "total:"
```

```
winter_group_region %>%  
  summarise(  
    sum(Gold),  
    sum(Silver),  
    sum(Bronze),  
    sum(Total)  
  )
```

```
## # A tibble: 5 x 5  
##   Region      `sum(Gold)` `sum(Silver)` `sum(Bronze)` `sum(Total)`  
##   <chr>          <int>         <int>         <int>         <int>  
## 1 ASIA           7            11            7            25  
## 2 AUSTRALIA       0            2            1            3  
## 3 EURASIA        19           13           14           46  
## 4 EUROPE         54           54           60          168  
## 5 NORTH_A        19           17           17           53
```

Question 6b

```
winter_group_region <- winter %>%  
  group_by(Region)
```

```
print("total:")
```

```
## [1] "total:"
```

```
winter_group_region %>%  
  summarise(  
    sum(Gold),  
    sum(Silver),  
    sum(Bronze),
```

```

    sum(Total)
  )

## # A tibble: 5 x 5
##   Region    `sum(Gold)` `sum(Silver)` `sum(Bronze)` `sum(Total)`
##   <chr>         <int>         <int>         <int>         <int>
## 1 ASIA              7             11              7             25
## 2 AUSTRALIA          0              2              1              3
## 3 EURASIA           19             13             14             46
## 4 EUROPE            54             54             60            168
## 5 NORTH_A           19             17             17             53

```

Question 6b

```

max_total_mean <- winter_group_region %>%
  summarise(mean_total = mean(Total)) %>%
  arrange(desc(mean_total)) %>%
  filter(row_number() == 1)
max_total_mean

```

```

## # A tibble: 1 x 2
##   Region mean_total
##   <chr>     <dbl>
## 1 NORTH_A    26.5

```

```

region_max_total_mean <- max_total_mean$Region
print(
  paste(
    "Region with maximum mean total medals:",
    region_max_total_mean
  )
)

```

```
## [1] "Region with maximum mean total medals: NORTH_A"
```

Question 6c

```

nb_countries_north_am <- nrow(
  winter %>%
    filter(Region == region_max_total_mean)
)

print(
  paste(
    "Number of countries in region",
    region_max_total_mean,
    ": ",
    nb_countries_north_am
  )
)

```

```
## [1] "Number of countries in region NORTH_A : 2"
```


Question 6d

```
nb_countries_eur <- nrow(
  winter %>%
    filter(Region == "EUROPE")
)

print(
  paste(
    "Number of countries in region EUROPE: ",
    nb_countries_eur
  )
)

## [1] "Number of countries in region EUROPE: 15"
```

Question 6e

```
max_nb_total <- winter %>%
  arrange(desc(Total)) %>%
  filter(row_number() == 1)

print(
  paste(
    "The maximum number of medals won is",
    max_nb_total$Total,
    "medals won by",
    max_nb_total$NOC
  )
)

## [1] "The maximum number of medals won is 33 medals won by Russia (RUS)*"
```