Project 1: Reproducible Research

Antonio Rubio Calzado

2 de junio de 2017

Introduction

We take the following sentences from the coursera webpage course:

"In the project we make use of data from a personal activity monitoring device. This device collects data at 5 minute intervals through out the day. The data consists of two months of data from an anonymous individual collected during the months of October and November, 2012 and include the number of steps taken in 5 minute intervals each day."

Loading and preprocessing the data

We start changing the working directory where the file `activity.csv` has been downloaded. After that, we read the .csv file.

```
setwd("C:/Users/arubioca/Desktop/CourseraDataScience")
data <- read.csv("activity.csv")
```

Now we make some exploratory data analysis to check how are the distict variables:

```
str(data)
```

```
## 'data.frame':    17568 obs. of  3 variables:
##  $ steps   : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ date    : Factor w/ 61 levels "2012-10-01","2012-10-02",..: 1 1 1 1
1 1 1 1 1 1 ...
##  $ interval: int  0 5 10 15 20 25 30 35 40 45 ...
```

```
summary(data)
```

```
##      steps                date          interval
##  Min.   :  0.00   2012-10-01:  288   Min.   :   0.0
##  1st Qu.:  0.00   2012-10-02:  288   1st Qu.: 588.8
##  Median :  0.00   2012-10-03:  288   Median :1177.5
##  Mean   : 37.38   2012-10-04:  288   Mean   :1177.5
##  3rd Qu.: 12.00   2012-10-05:  288   3rd Qu.:1766.2
##  Max.   :806.00   2012-10-06:  288   Max.   :2355.0
##  NA's   :2304     (Other)   :15840
```

As we check that the value `NA` is taken in many step variable's registers, we can filter our original data to omit that values:

```
data_no_NA <- data[!is.na(data$steps),]
```

Now the data is suitable for our analysis.

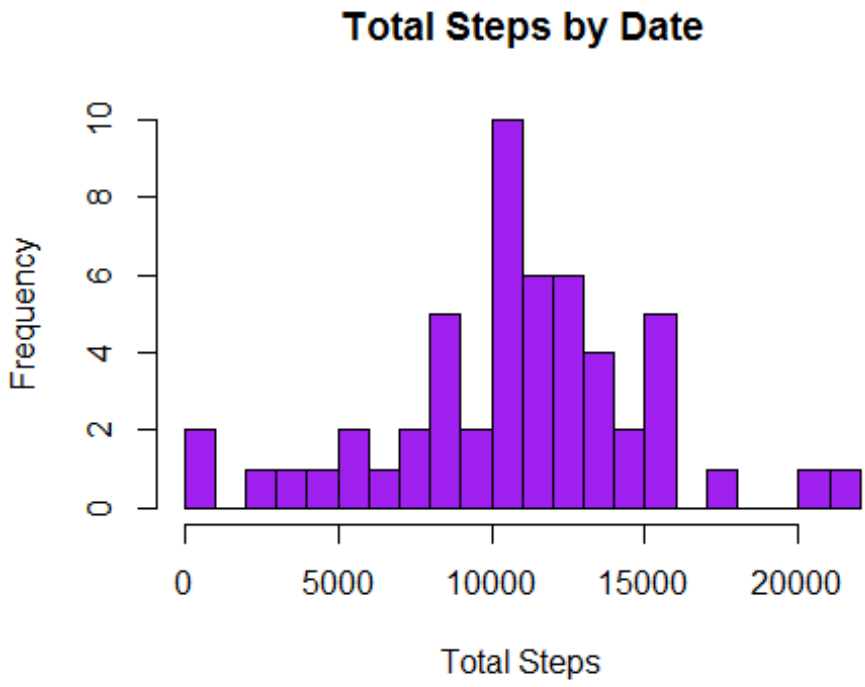
Is mean total number of steps taken per day?

Let's create a dataframe with two columns: The first one has the diferents dates appearing on `data_no_NA` and the second one has the sum of all the steps, grouped by this date.

```
steps_by_day <- aggregate(data_no_NA$steps, by = list(data_no_NA$date) ,F
UN = sum)
colnames(steps_by_day) <- c("date","steps_sum")
```

Now we can compute the mean, the median and plot a histogram of the variable `steps_sum`:

```
mean(steps_by_day$steps_sum)

## [1] 10766.19

median(steps_by_day$steps_sum)

## [1] 10765

hist(steps_by_day$steps_sum, main = "Total Steps by Date" , breaks = 20,
col ="purple" ,xlab = "Total Steps")
```



What is the average daily activity pattern?

Let's create a dataframe with two columns: The first one has the diferents intervals appearing on `data_no_NA` and the second one has the mean of all the steps, grouped by interval.

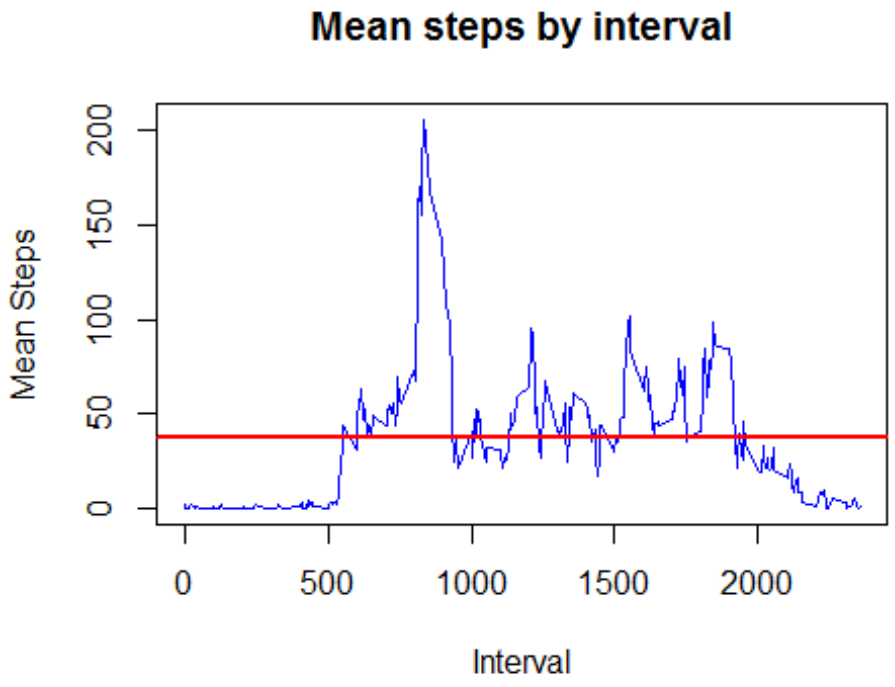
```
meansteps_by_interval <- aggregate(data_no_NA$steps, by = list(data_no_NA
$interval) ,FUN = mean)
colnames(meansteps_by_interval) <- c("interval","steps_mean")
```

We transform thte variable `steps_mean` to be integer:

```
meansteps_by_interval$steps_mean <- round(meansteps_by_interval$steps_mea
n)
```

Now we can plot a time series of the variables `interval` versus `steps_mean`:

```
plot(meansteps_by_interval$interval,meansteps_by_interval$steps_mean, typ
e = "l", col = "blue",lwd = 1.5, xlab = "Interval", ylab="Mean Steps")
abline(h = mean(meansteps_by_interval$steps_mean), col = "red", lwd = 2)
title(main = "Mean steps by interval")
```



With the next command, we get the `interval` in which the time series reach its maximum `steps_mean` value:

```
meansteps_by_interval[which.max(meansteps_by_interval$steps_mean),]
```

```
##     interval steps_mean
## 104      835        206
```

Imputing missing values

From the original data, we can compute how many NA values does the steps variable takes:

```
sum(is.na(data$steps))

## [1] 2304
```

Our strategy now is filling the NA values with the mean step value that we have computed for the intervals, so we can start merging the original dataframe data with meansteps_by_interval by the field interval:

```
new_data <- merge(data,meansteps_by_interval,by ="interval", all.x = TRUE
)
```

With the following R sentence, we find the NA values in the variable steps in our merged dataset and change it by its corresponding mean by interval:

```
for (i in 1:length(new_data$steps)){
  if (is.na(new_data$steps[i])){
    new_data$steps[i] <- new_data$steps_mean[i]
  }
}
```

Let's calculate a dataframe with the days and the total number of steps taken each day.

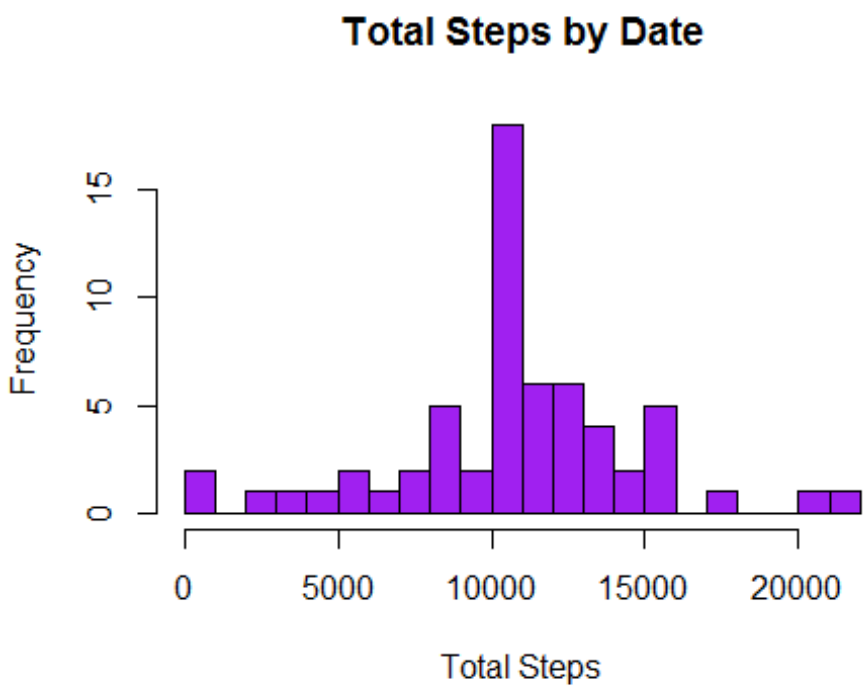
```
new_steps_by_day <- aggregate(new_data$steps, by = list(new_data$date) ,F
UN = sum)
colnames(new_steps_by_day) <-c("date","steps_sum")
```

Now we compute the new mean, the new median of steps_sum:

```
mean(new_steps_by_day$steps_sum)

## [1] 10765.64

median(new_steps_by_day$steps_sum)

## [1] 10762
```

And a histogram of this variable is:

```
hist(new_steps_by_day$steps_sum, main = "Total Steps by Date" , breaks =
20, col ="purple" ,xlab = "Total Steps")
```



We can see that both the mean and the median of `steps_sum` was higher when dropping the `NA` values.

Are there differences in activity patterns between weekdays and weekends?

For doing this part of the project, we load the following library:

```
library(data.table)
```

We are creating two vectors with the days of the week:

```
week_days <- c("lunes","martes","miércoles","jueves","viernes")
weekend_days <- c("sábado","domingo")
```

Now we are going to change the datetype of `date` and inserting a new colums in our new dataset indicating the name of the day.

```
fech <- as.Date(new_data$date)
new_data$dia <- weekdays(fech)
```

The key idea now, is separating the days in two dataframes: The first one with the weekdays and a flag indicating that, and the second one with the weekend days and another flag.

```
week_df <- new_data[new_data$dia %in% week_days, ]
week_df$flag <- as.factor("Week")
```

```
weekend_df <- new_data[new_data$dia %in% weekend_days, ]
weekend_df$flag <- as.factor("Weekend")
```

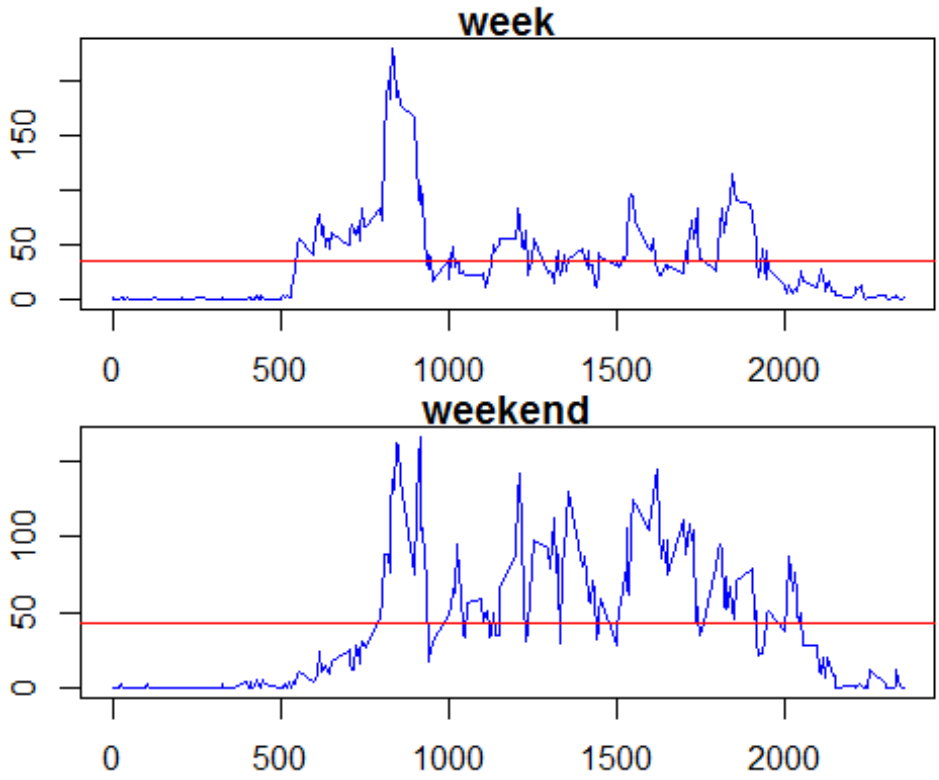
As requested, let's conclude making a panel plot containing a time series plot of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all weekday days or weekend days (y-axis).

But before, we need to create to dataframes studying the mean of steps by interval on week and weekend:

```
week_steps_by_minute <- aggregate(week_df$steps, by = list(week_df$interv
al) ,FUN = mean)
weekend_steps_by_minute <- aggregate(weekend_df$steps, by = list(weekend_
df$interval) ,FUN = mean)
```

And conclude with the panel plot time series:

```
par(mfcol=c(2, 1), mar=c(2,2,1,1))
plot(week_steps_by_minute$Group.1,week_steps_by_minute$x, type = "l", col
= "blue", xlab = "Interval" , ylab= "Mean of steps", main = "week")
abline(h=mean(week_steps_by_minute$x),lwd = 1.5,col="red")
plot(weekend_steps_by_minute$Group.1,weekend_steps_by_minute$x, type = "l
", col = "blue", xlab = "Interval" , ylab= "Mean of steps", main = "weeke
nd")
abline(h=mean(weekend_steps_by_minute$x),col="red")
```



The conclussion is that people change their habits respect activity between weekdays and weekend days and that could be due to jobs or similar.