Assignment of the Reproducible Research course

Fabio

25 febbraio 2017

Setting the environment and reading the data

Needed packages are dplyr and lattice

```
library("dplyr")
library("lattice")
```

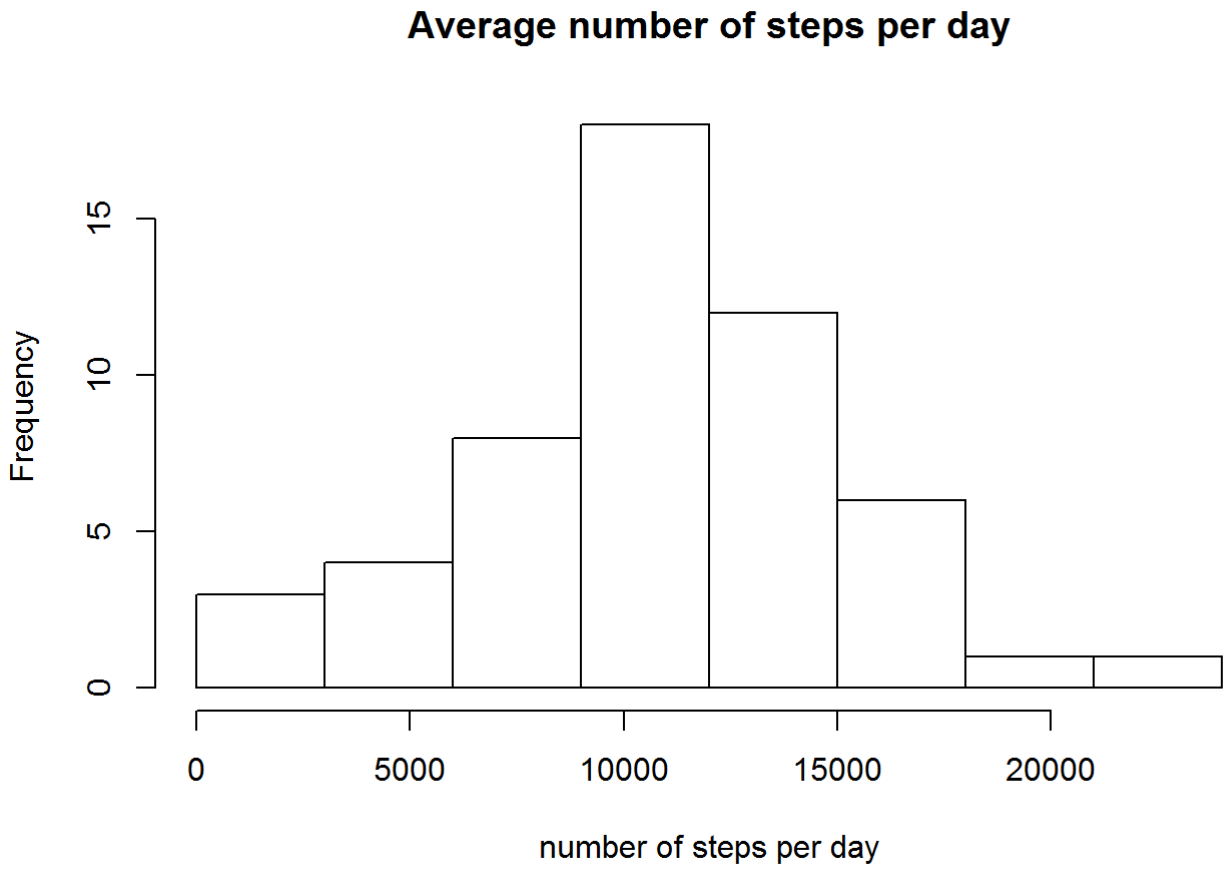
Code for reading the data. This code assumes that in the working directory is accessible a file named "activity.csv". Downloaded and unzipped from https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip (https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip) last February 24, 2017. activityWONA filters out NA's

```
activity <- read.csv("activity.csv", header=TRUE, sep=",")
activityWONA <- filter(activity, steps>=0)
```

Histogram of the number of steps taken each day

Steps per day are calculated and the frequency histogram is plotted

```
stepsPerDay <- aggregate(steps~ date, data=activityWONA, FUN = "sum")
hist(stepsPerDay$steps, breaks=seq(0,24000, by=3000), main= "Average number of steps per
day", xlab="number of steps per day")
```



Mean and median steps per day

```
meanToReport<-as.integer(mean(stepsPerDay$steps))
```

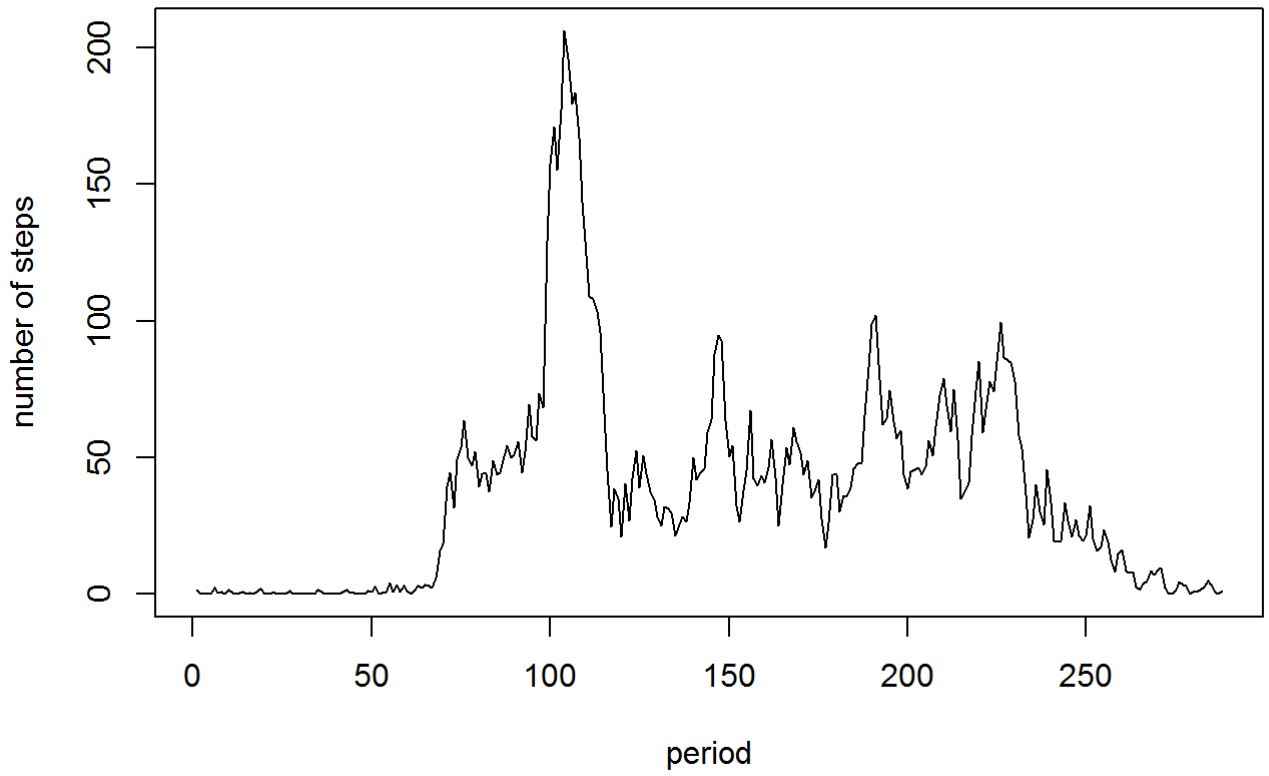
Mean steps per day is 10766.

```
medianToReport<-median(stepsPerDay$steps)
```

its median is 10765

Time series of the steps per day

```
steps5Min <- aggregate(steps~ interval, data=activityWONA, FUN = "mean")
plot(steps5Min$steps, type="l", xlab="period", ylab="number of steps")
```



Calculation of the interval with the highest number of steps on average

The highest average number of steps in one interval and the interval are computed as follows

```
temp<-filter(steps5Min,steps==max(steps5Min$steps))
```

The interval of interest is 835. The average number of steps is 206.1698113.

Replacing missing values with credible values

How many missing values ?

There are 2304 entries with missing values

Now let's create a new dataset with NA's replaced by the corresponding mean for that interval. We take the floor (i.e. the integer part) of that mean

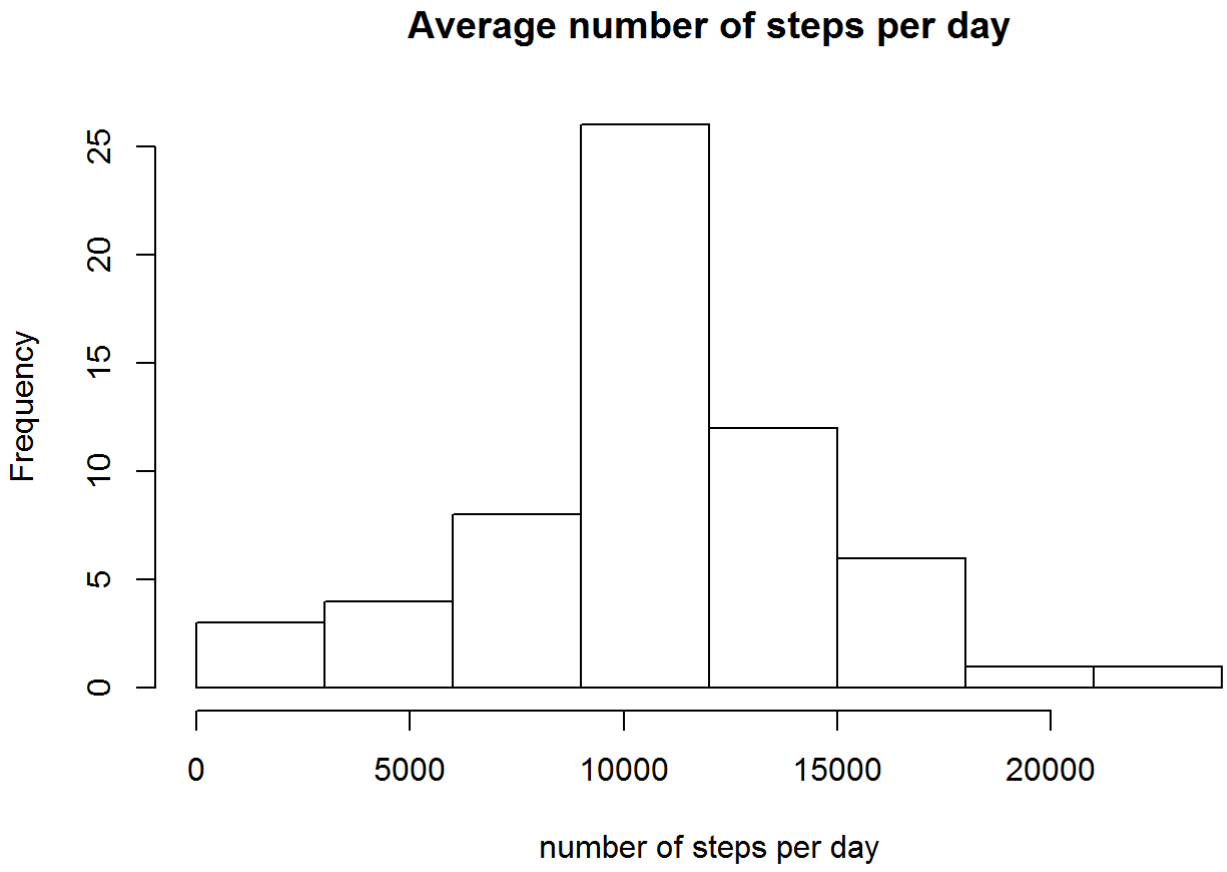
```
unDates<-!duplicated(activity$date)
NumOfDays<-nrow(activity[unDates,])
```

Data refer to 61 different days.

```
cleanedActivity <- activity
compare <- rep(floor(steps5Min$steps), NumOfDays)
for (i in (1:nrow(cleanedActivity))) {if (is.na(cleanedActivity$steps[i])){
  cleanedActivity$steps[i]<-compare[i]
   }
}
```

Histogram of the total number of steps taken each day after missing values are imputed

```
CleanedstepsPerDay <- aggregate(steps~ date, data=cleanedActivity, FUN = "sum")
hist(CleanedstepsPerDay$steps, breaks=seq(0,24000,by=3000), main= "Average number of steps pe
r day", xlab="number of steps per day")
```



Mean and median steps taken each day computed on the cleaned dataset

```
mean(CleanedstepsPerDay$steps)
```

```
## [1] 10749.77
```

```
median(CleanedstepsPerDay$steps)
```

```
## [1] 10641
```

Panel plot comparing the average number of steps taken per 5-minute interval across weekdays and weekends

```
cleanedActivity$date<-as.Date(cleanedActivity$date, "%Y-%m-%d")
weekdays(cleanedActivity$date[1])
```

```
## [1] "lunedì"
```

so the first day considered is Monday and we have 288 intervals per day.

Now we will create a new column named weekend.It will be a factor with two levels: weekday and weekend.

```
cleanedActivity<-mutate(cleanedActivity, weekend=as.factor(rep(c(0,0,0,0,0,1,1), each=288, le
ngth.out=288*NumOfDays)))

levels(cleanedActivity$weekend)<- c("weekday","weekend")
```

Plot of the time series on weekend and weekdays

The results are shown into two different panels side by side

```
groupbyIntAndWE<-group_by(cleanedActivity, interval,weekend)
final<-summarize(groupbyIntAndWE,n=mean(steps))
xyplot(n~interval | weekend, data=final, type="l")
```