Reproducible Research: Peer Assessment 1

Didier DUMET

Reproducible Research: Peer Assessment 1

This research studies the activity (number of steps) patterns of an individual over a period of two months (October and November 2012). Steps are counted on a 5 minutes interval.
Original data are available here (https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip)

Loading and preprocessing the data

It is assumed that individuals willing to reproduce this research, will clone this GIT repository, and therefore the zipped raw data set is available. Loading and preprocessing the data is done using:

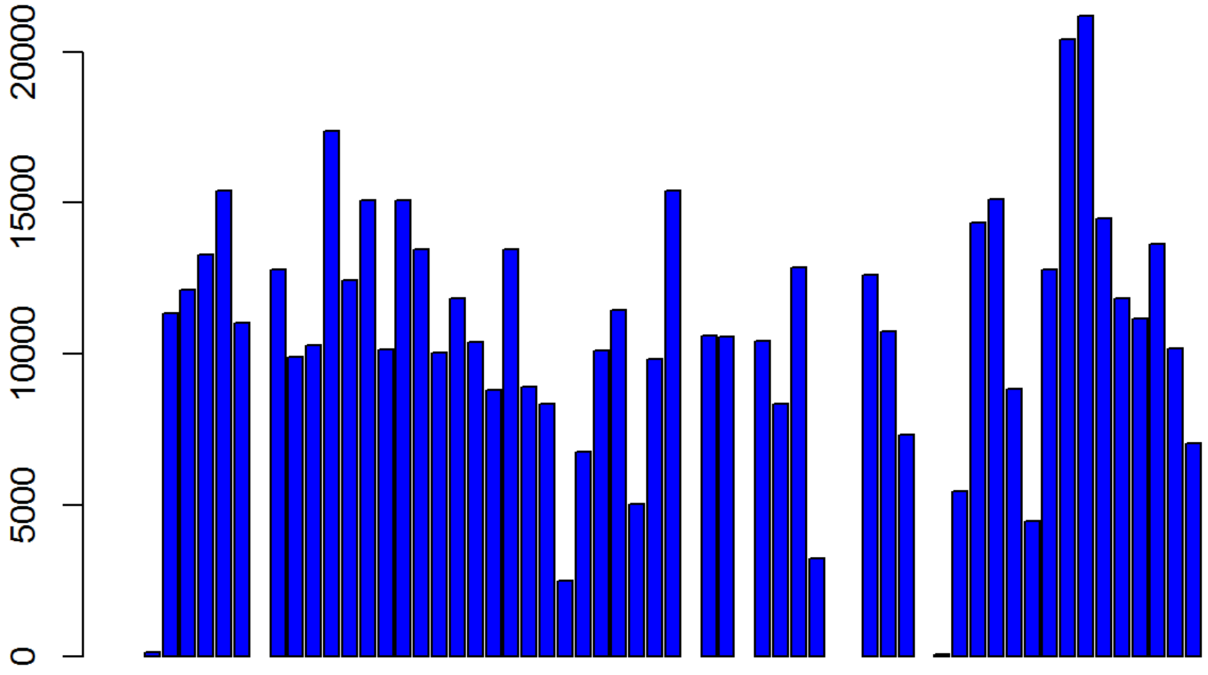
```
# unzipping and loading the data
unzip("activity.zip")
data <- read.csv("activity.csv", na.strings = c("NA"))

# Setting the format for date
data$date <- as.Date(data$date, format="%Y-%m-%d")
```

What is mean total number of steps taken per day?

Let’s explore what is the total number of steps taken each day.

```
# Plot
stepsPerDay<-aggregate(data$steps,list(date=data$date),sum)
barplot(height=stepsPerDay$x,col="blue")
```



And let’s calculate the mean and median for the total number of steps taken per day. This is done using:

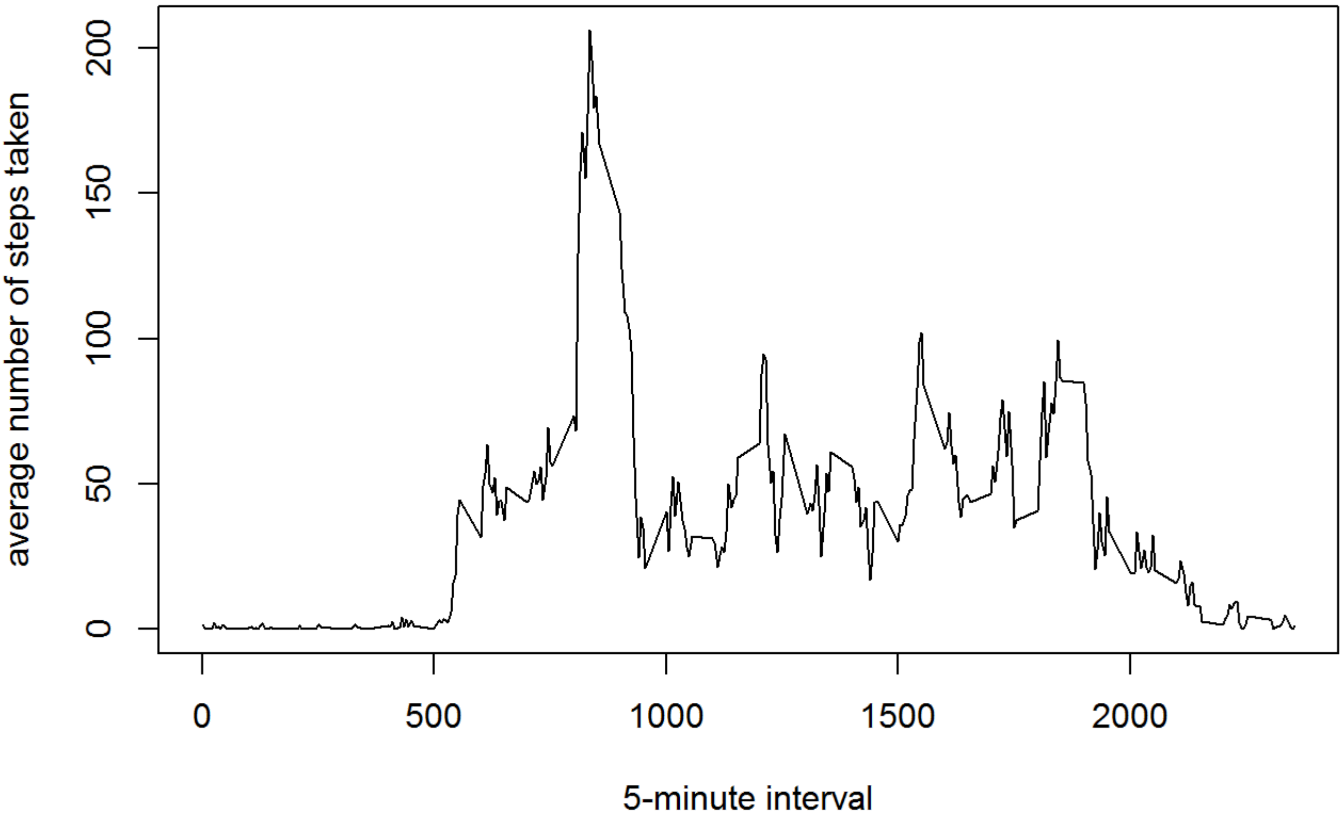
```
meanOfStepsPerDay<-mean(stepsPerDay$x,na.rm=TRUE)
medianOfStepsPerDay<-median(stepsPerDay$x,na.rm=TRUE)
```

The mean is 1.076618910^{4} and the median is 10765.

What is the average daily activity pattern?

To look for a daily pattern, we’re showing the average number of steps across all days for each 5 minute interval. That is done calculating the mean for each 5 minute interval across all days of the two month period. This is done by:

```
meanByInterval<-aggregate(data$steps~data$interval,data=data,FUN=mean,na.action=na.omit)
names(meanByInterval)<-c("interval","maxSteps")
plot(meanByInterval,type="l",xlab="5-minute interval",ylab="average number of steps taken")
```



The interval containing the maximum number of steps is interval 835 with a number of steps of 206.1698113.

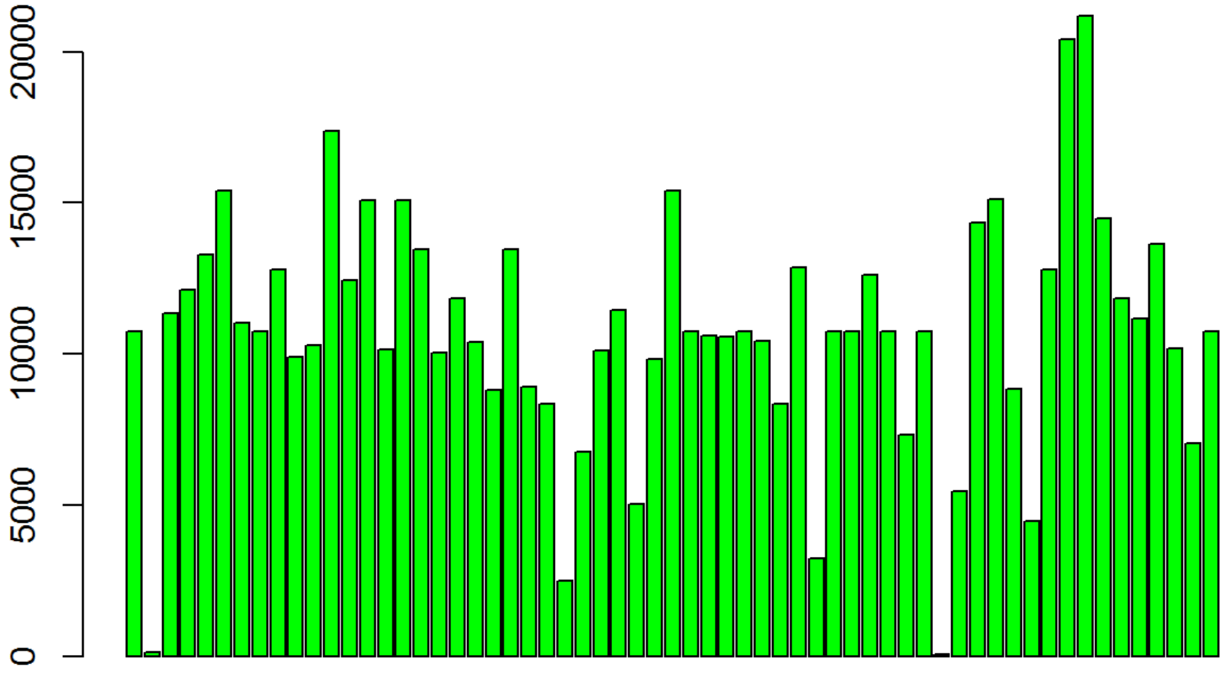
Inputing missing values

We will input missing value by using the mean for each interval. This is done as follow:

```
dataInt<-merge(data,meanByInterval,by.x='interval',by.y='interval')
for(i in 1:dim(dataInt)[1]) {
  if(is.na(dataInt[i,2])) dataInt[i,2]<-dataInt[i,4]
  i<i+1
}
dataNoNA<-dataInt[,2:3]
dataNoNA<-cbind(dataNoNA,dataInt[,1])
names(dataNoNA)[3]<-c("interval")
```

Let’s plot the new activity dataset (with no missing value):

```
stepsPerDayNoNA<-aggregate(dataNoNA$steps,list(date=dataNoNA$date),sum)
barplot(height=stepsPerDayNoNA$x,col="green")
```



The diagram does not look that different from the initial diagram we produced (with missing values). This is confirmed by calculating the mean and median for the total number of steps taken per day, the mean being 1.076618910^{4} and the median being 1.076618910^{4}. These values differ slightly from the initial value, with the mean being similar and the median moving closer to the mean. The impact of inputing missing data with this strategy is limited, one can argue if the strategy chosen is a correct one.

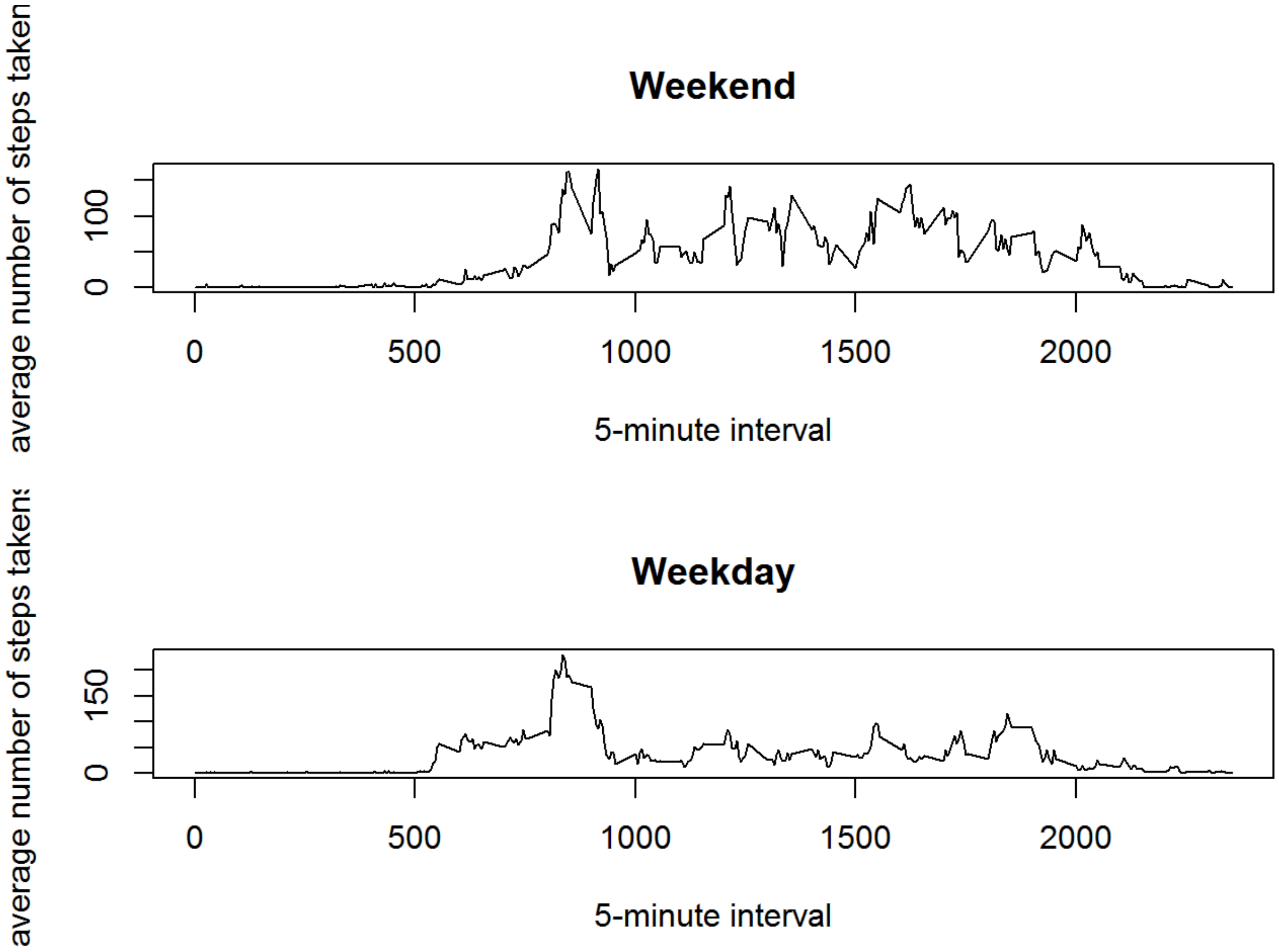
Are there differences in activity patterns between weekdays and weekends?

We add a factor variable to the dataset (the one with inputed missing values) for weekdays and weekend.

```
dayOfTheWeek<-as.factor(weekdays(as.POSIXlt(dataNoNA$date)))
weekend<-(dayOfTheWeek=='Saturday'|dayOfTheWeek=='Sunday')
factorWeek<-rep(c("weekday"),length(dayOfTheWeek))
factorWeek[weekend]<-"weekend"
factorWeek<-as.factor(factorWeek)
dataFinal<-cbind(dataNoNA,factorWeek)
```

And plot average steps across weekdays and weekends on a panel:

```
dataFinalWeekday<-dataFinal[dataFinal$factorWeek=='weekday',]
dataFinalWeekend<-dataFinal[dataFinal$factorWeek=='weekend',]
weekdayMeanByInterval<-aggregate(dataFinalWeekday$steps~dataFinalWeekday$interval,data=dataFinalWeekday,FUN=mean)
weekendMeanByInterval<-aggregate(dataFinalWeekend$steps~dataFinalWeekend$interval,data=dataFinalWeekend,FUN=mean)
names(weekdayMeanByInterval)<-c("interval","maxSteps")
names(weekendMeanByInterval)<-c("interval","maxSteps")
par(mfrow=c(2,1))
plot(weekendMeanByInterval,type="l",main="Weekend",xlab="5-minute interval",ylab="average number of steps taken")
plot(weekdayMeanByInterval,type="l",main="Weekday",xlab="5-minute interval",ylab="average number of steps takens")
```



Activity during weekends is more distributed across the day, whereas for weekdays one can observed two peaks of activity, one in the morning and another one (less important) in the evening.