Reproducible Research - Assignment 1

Maria Fernandez Carcedo

10 January 2016

Introduction

This assignment makes 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.

The assigment takes the form of a report that will analyse the data set described above and aswer the following questions:

1. What is mean total number of steps taken per day?
2. What is the average daily activity pattern?
3. Are there differences in activity patterns between weekdays and weekends?

Data

Dataset The data for this assignment can be downloaded from the course web site:

- **Dataset**: activity.csv

The variables included in this dataset are:

- **steps**: Number of steps taking in a 5-minute interval (missing values are coded as NA)
- **date**: The date on which the measurement was taken in YYYY-MM-DD format
- **interval**: Identifier for the 5-minute interval in which measurement was taken

Loading and preprocessing the data We will dowload the data from the working directory:

```r
setwd("~/Desktop/WD")
x <-read.csv("./activity.csv", header=TRUE, sep= ",", stringsAsFactors=FALSE)
summary(x)
```

```
 steps              date                interval
```

```
Min. : 0.00 Length:17568 Min. : 0.0
1st Qu.: 0.00 Class :character 1st Qu.: 588.8
Median : 0.00 Mode :character Median :1177.5
Mean : 37.38 Mean :1177.5
3rd Qu.: 12.00 3rd Qu.:1766.2
Max. :806.00 Max. :2355.0
NA's :2304
```

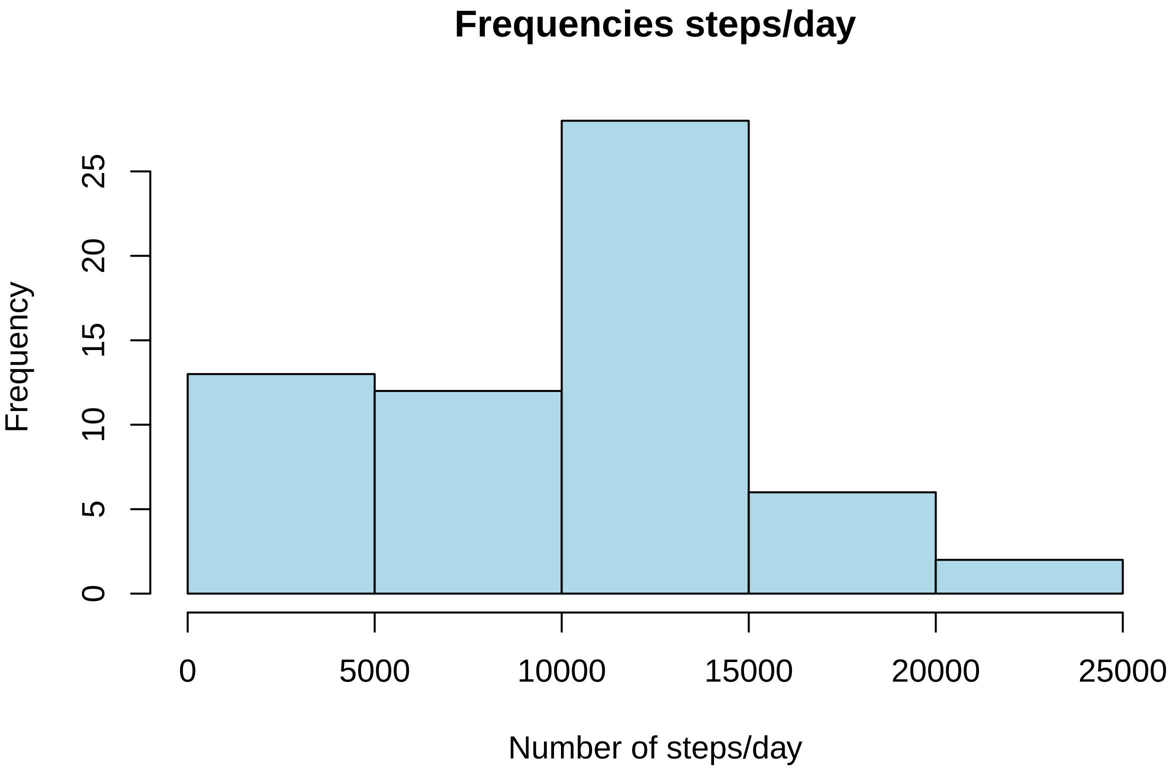
1. What is the mean total number of steps taken per day?

First of all we are going to calculate the total number of steps taken per day

```
sum <-tapply(x$steps, x$date, sum, na.rm=TRUE)
```

Then we are going to produce an histogram of the total number of steps per day

```
hist(sum, freq=TRUE, main="Frequencies steps/day", xlab= "Number of steps/day", col="lightblue")
```



The mean and the median of the total number of steps taken per day are provided in the summary below:

```
summary (sum)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##       0    6778   10400    9354   12810   21190
```

Thus, the mean number of steps taken per day is 9354 steps, and the median number is 10400.

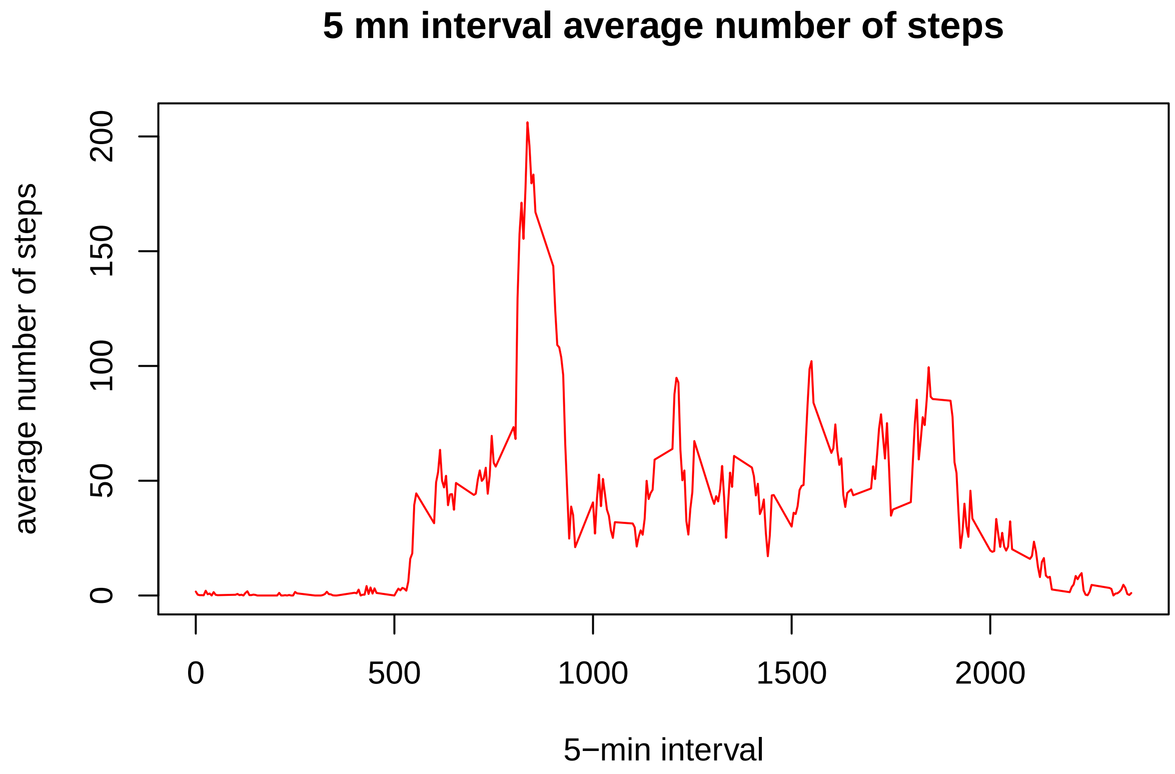
2. What is the average daily activity pattern?

In this case we will first calculate the mean number of steps taken by interval across all days:

```
mean <- tapply(x$steps, x$interval, mean, na.rm=TRUE)
```

And then we will produce below a 5 minute interval series plot and the average number of steps taken across all days:

```
plot(mean ~ unique(x$interval), type="l", main="5 mn interval average number of steps", xlab = "5-min i
```



The 5 mn interval that contains the highest average number of steps is:

```
max(mean)
```

```
## [1] 206.1698
```

Imputing Missing Values Since there are a number of days/intervals where there are missing values (coded as NA), which may introduce bias into some calculations or summaries of the data, we are going to device a strategy to imput the missing values.

In fact, as shown below, there is a significant number of 2304 missing data in the database:

```
sum (is.na(x)==TRUE)
```

```
## [1] 2304
```

And it is to be noted that all the NA values are in the *steps* column:

```
summary (is.na(x)==TRUE)
```

```
##    steps              date            interval
##  Mode :logical   Mode :logical   Mode :logical
##  FALSE:15264     FALSE:17568     FALSE:17568
##  TRUE :2304      NA's :0         NA's :0
##  NA's :0
```

To replace the NA values in the dataset we are going to use the mean value for the corresponding 5 mn interval instead of the missing data, and we will create a new dataset with the replacements:

```
newx <- x
for (i in 1:nrow(x)){
   if(is.na(x$steps[i])==TRUE){
           newx$steps[i] <- mean[[as.character(x[i, "interval"])]]
   }
}
```

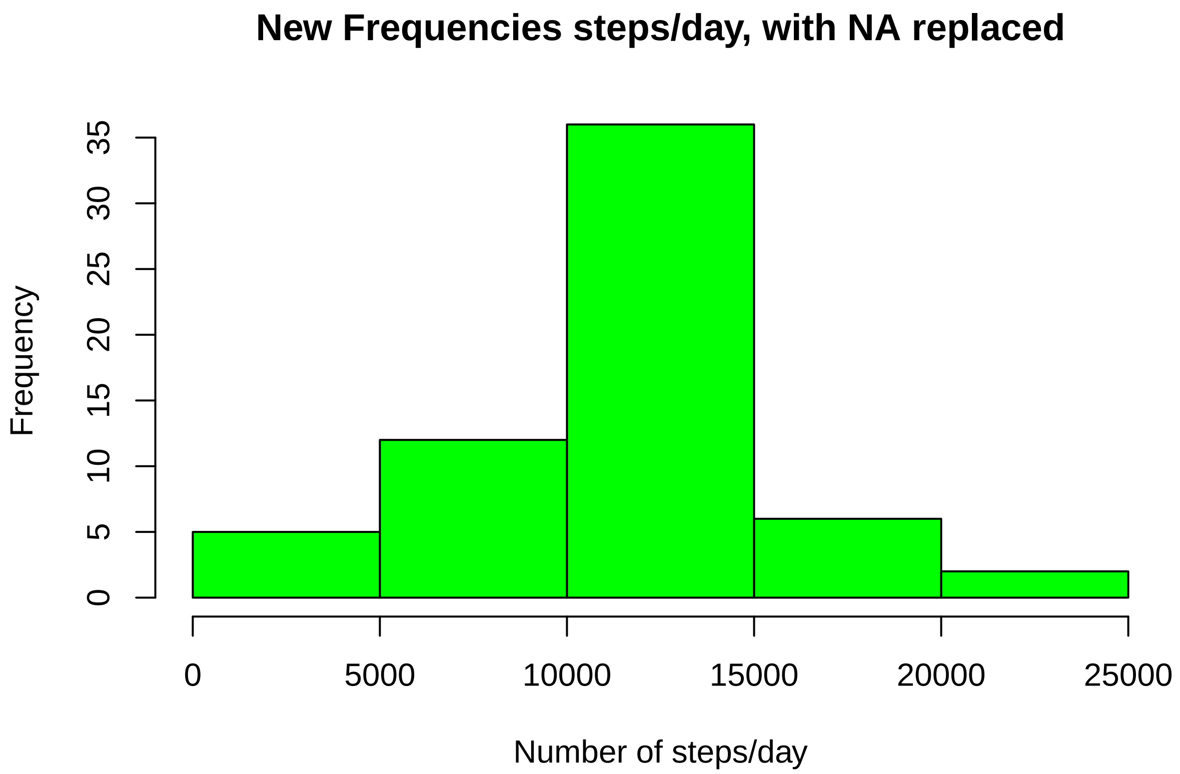
where the NA values are:

```
sum (is.na(newx)==TRUE)
```

```
## [1] 0
```

And that could be represented in the following histogram of frequencies of steps per day:

```
sum2 <- tapply(newx$steps, newx$date, sum)
hist(sum2, freq=TRUE, main="New Frequencies steps/day, with NA replaced", xlab= "Number of steps/day",
```



With the NA values replaced the mean and median number of steps are summarized below:

```
summary (sum2)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      41    9819   10770   10770   12810   21190
```

Both figures overlap at 10770 steps,

thus increasing the figures that were obtained with NA values, 9354.2295082 mean of steps, and 10395 median of steps.

3. What is the average daily activity pattern?

We are going first to convert the Date column in the data base with NA substituted into a Date and Time vector, and then into a weekdays column, obtaining a new data frame. We will use the weekdays fuction in this operation:

```
data <- newx
{for (i in 1:17568){
        newx$date[i]<- weekdays(as.Date((newx$date[i]), origin="1970-01-01", tz="GMT"), abbreviate=FALS
        }
}
```

Then we will convert the days of the week in the date colum into weekday or weekend:

```
data$date <- ifelse(data$date %in% c("Saturday", "Sunday"), "weekend", "weekday")
```

Next we create two subsets based on the two factors created: “weekend” and “weekday”

```
library (dplyr)
```

```
##
## Attaching package: 'dplyr'
##
## The following object is masked from 'package:stats':
##
##     filter
##
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
sub_weekday <- filter (data, date=="weekday")
sub_weekend <- filter (data, date=="weekend")
```

We will also have to calculate the mean number of steps for each interval in both subsets, produce two data sets with three variables (interval, mean of steps and day) and combine them into one final dataset:

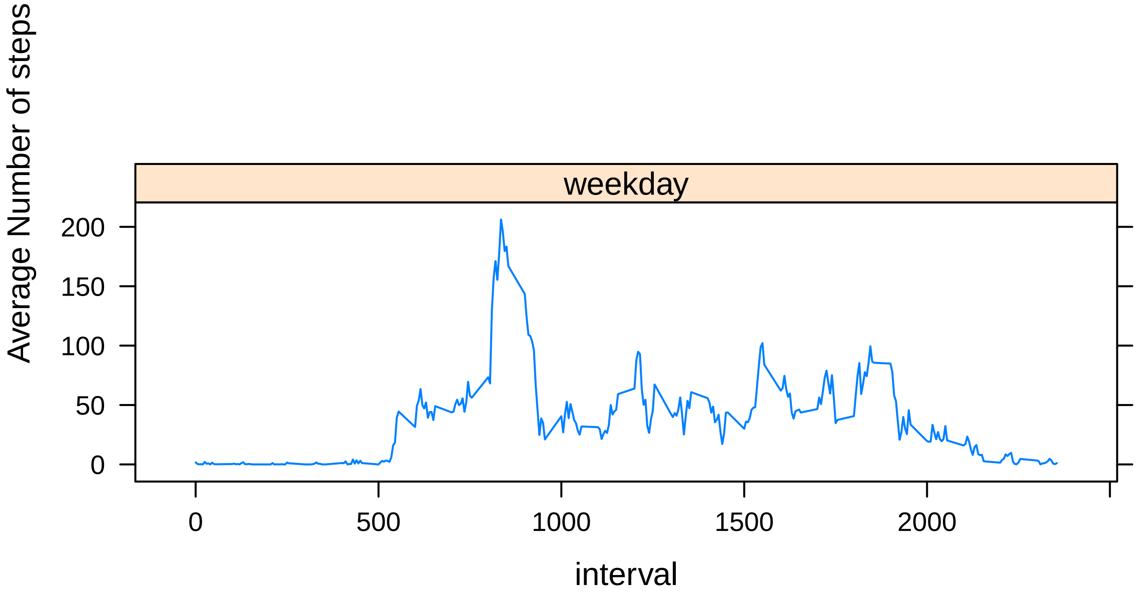
```
mean_sub_weekday <- tapply(sub_weekday$steps, sub_weekday$interval, mean, na.rm=TRUE)

mean_sub_weekend <- (tapply(sub_weekend$steps, sub_weekend$interval, mean, na.rm=TRUE))

df_wday <- data.frame(interval = unique(sub_weekday$interval), avg = as.numeric(mean_sub_weekday), day

df_wend <- data.frame(interval = unique(sub_weekend$interval), avg = as.numeric(mean_sub_weekend), day

df_final <- rbind(df_wday, df_wend)
```

Finally, if we still have any neurones left after all… we can dedicate it to a last effort and plot, before the remains of our brain take a more revolutionary path and burst into a decided “basta ya!”

```
library (lattice)
```

```
## Warning: package 'lattice' was built under R version 3.1.3
```

```
xyplot(avg ~ interval | day, data = df_final, layout = c(1, 2),
       type = "l", ylab = "Average Number of steps")
```



Based on the graphics above, the average number of steps seems to be greater durinig the 900 first 5mn inteval of the weekday series than the weekends, and invert the trend after this point in time, where the average number of steps is greater during the weekends than during the weekdays.

Note: due to reasons that are beyond my understanding the first panel of the last graphic is not printed out in the output, although if works well when I run the code to the console!