Reproducible Research: Peer Assessment 1

Fernando Martínez Plumed

23th of September,2015

Loading and preprocessing the data

Load libraries

```
 # List of packages for session
.packages <- c("dplyr","reshape2","lubridate","ggplot2", "xtable")

installed <- function(pack){pack %in% installed.packages()[,1]}

load.pack <- function(packs){
  for (p in packs){
    if (!installed(p)){
      install.packages(p)
    }
    require(p, character.only=TRUE)

  }
}

load.pack(.packages)
```

Load the data (i.e. read.csv())

```
Activity <- tbl_df(read.csv("activity.csv"))
Activity
```

```
## Source: local data frame [17,568 x 3]
##
##    steps       date interval
## 1     NA 2012-10-01        0
## 2     NA 2012-10-01        5
## 3     NA 2012-10-01       10
## 4     NA 2012-10-01       15
## 5     NA 2012-10-01       20
## 6     NA 2012-10-01       25
## 7     NA 2012-10-01       30
## 8     NA 2012-10-01       35
## 9     NA 2012-10-01       40
## 10    NA 2012-10-01       45
## ..   ...        ...      ...
```

Process/transform the data (if necessary) into a format suitable for your analysis: Date from factor to Date

```
Activity$date <- ymd(Activity$date)
#Activity <- Activity[complete.cases(Activity)]
```

What is mean total number of steps taken per day?

For this part of the assignment, you can ignore the missing values in the dataset.

Make a histogram of the total number of steps taken each day

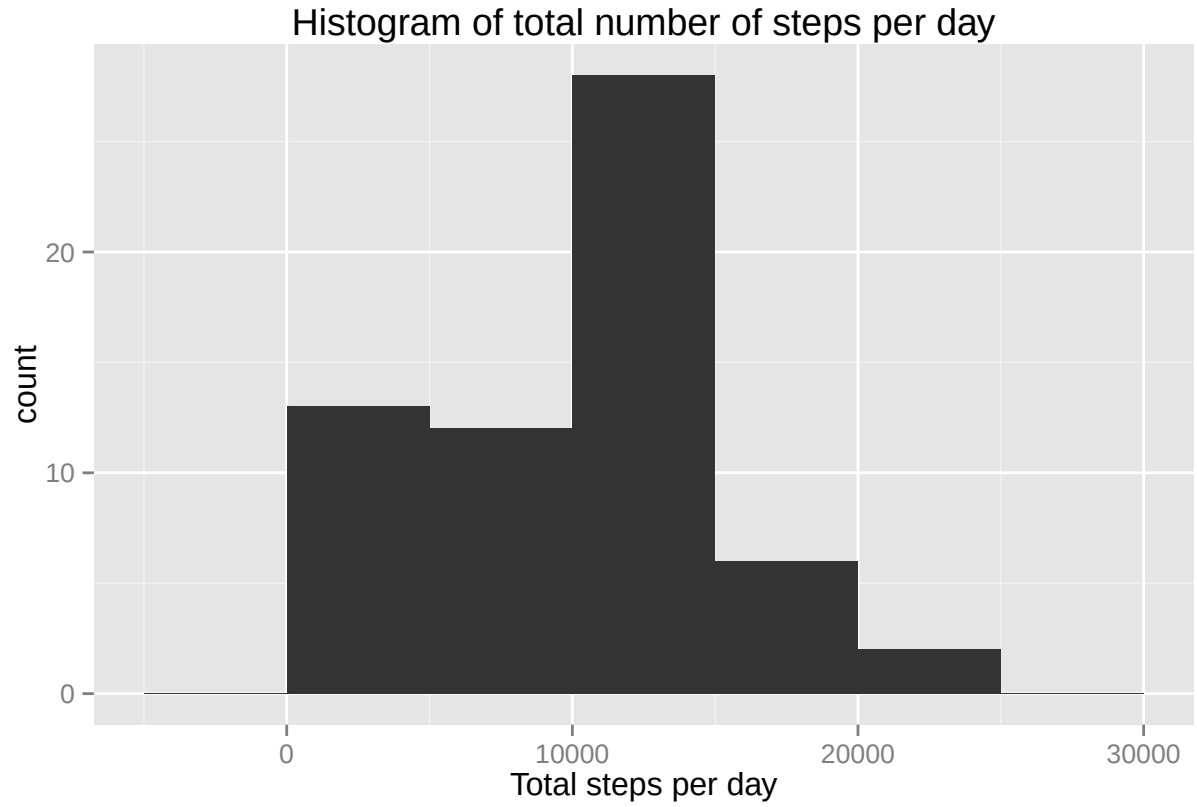
```
Activity.day <- group_by(Activity, date)
Activity.day
```

```
## Source: local data frame [17,568 x 3]
## Groups: date
##
##    steps       date interval
## 1     NA 2012-10-01        0
## 2     NA 2012-10-01        5
## 3     NA 2012-10-01       10
## 4     NA 2012-10-01       15
## 5     NA 2012-10-01       20
## 6     NA 2012-10-01       25
## 7     NA 2012-10-01       30
## 8     NA 2012-10-01       35
## 9     NA 2012-10-01       40
## 10    NA 2012-10-01       45
## ..   ...        ...      ...
```

```
Activity.day.sum <- summarise(Activity.day, total= sum(steps,na.rm = T))



g <- ggplot(select(Activity.day.sum, date, total), aes(x=total))
g + geom_histogram(binwidth=5000) + ggtitle("Histogram of total number of steps per day")+ xlab("Total s
```



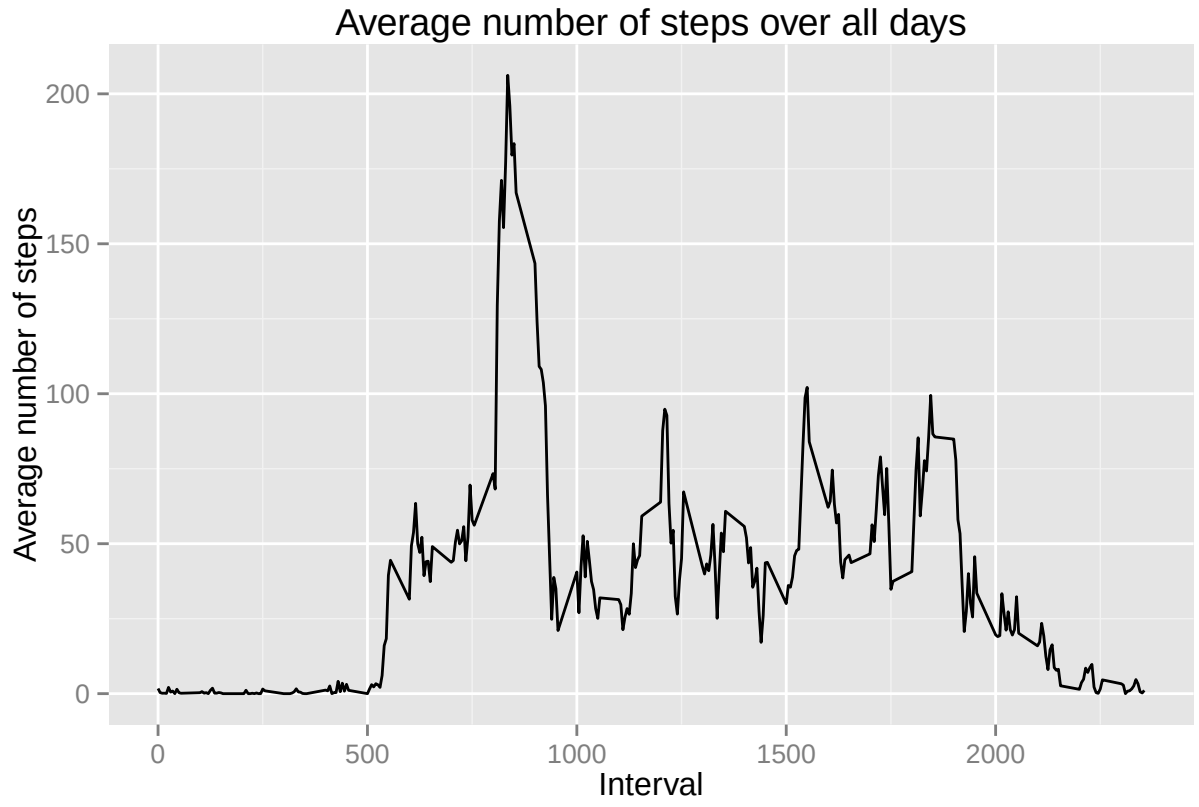

```
mean <- trunc(mean(Activity.day.sum$total, na.rm = T))
median <- trunc(median(Activity.day.sum$total, na.rm = T))
```

What is the average daily activity pattern?

Make a time series plot (i.e. type = "l") of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis)

```
Activity.interval <- group_by(Activity, interval)
Activity.interval.avg <- summarise(Activity.interval, avg = mean(steps, na.rm=T))


g <- ggplot(Activity.interval.avg, aes(interval, avg))
g + geom_line() + ggtitle("Average number of steps over all days") + xlab("Interval") + ylab("Average nu
```




```
max <- Activity.interval.avg[which.max(Activity.interval.avg$avg),]$interval
```

Which 5-minute interval, on average across all the days in the dataset, contains the maximum number of steps? 835

Imputing missing values

Note that there are a number of days/intervals where there are missing values (coded as NA). The presence of missing days may introduce bias into some calculations or summaries of the data.

```
rowsNA <- sum(is.na(Activity))
```

The total number of missing values in the dataset (i.e. the total number of rows with NAs) : 2304

Replacing NA's with the mean for that 5-minute interval. Creating a new dataset that is equal to the original dataset but with the missing data filled in.

```
Activity.clean <- Activity
for(i in 1:nrow(Activity.clean)){
  if (is.na(Activity.clean$steps[i])){
    thisInterval <- Activity.clean$interval[i]
    AvgValue <- Activity.interval.avg[Activity.interval.avg$interval == thisInterval,]$avg
    Activity.clean$steps[i]<- AvgValue
  }
}
```

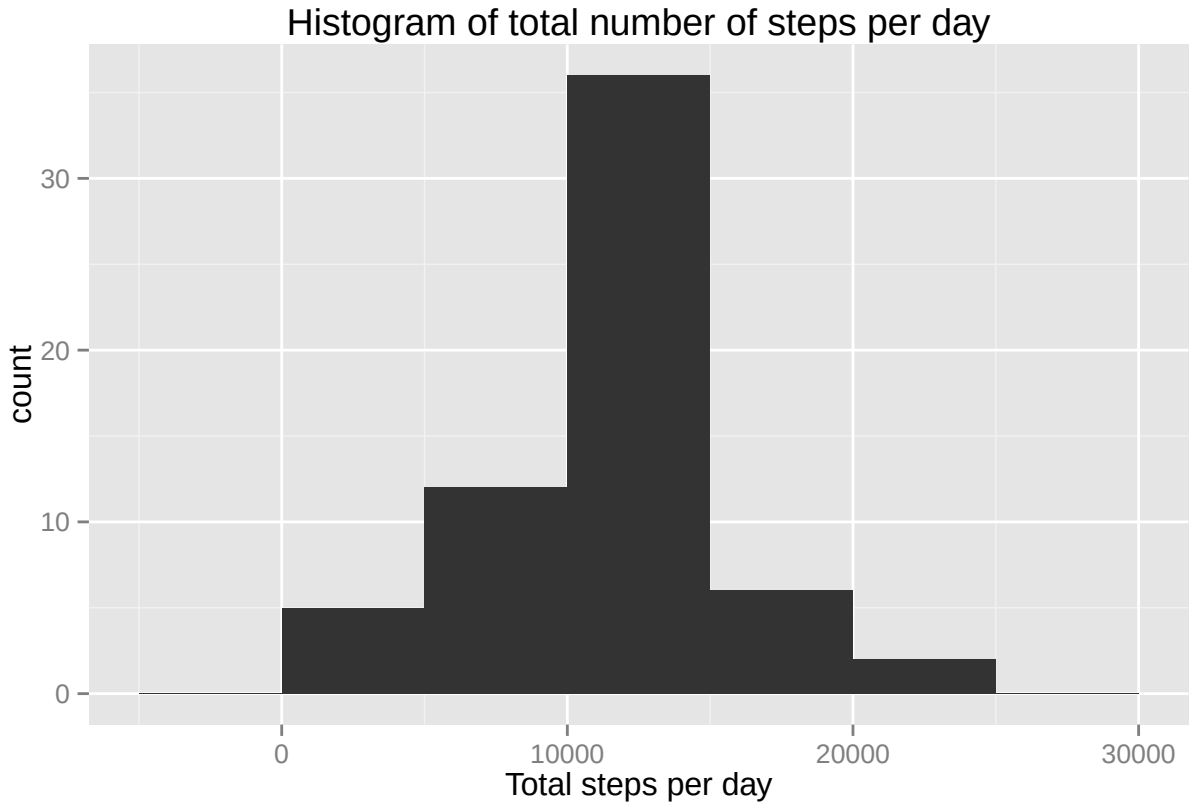
Histogram of the total number of steps taken each day and the mean and median total number of steps taken per day. Do these values differ from the estimates from the first part of the assignment? What is the impact of imputing missing data on the estimates of the total daily number of steps?

```
Activity.clean.day <- group_by(Activity.clean, date)
Activity.clean.day
```

```
## Source: local data frame [17,568 x 3]
## Groups: date
##
##        steps       date interval
## 1  1.7169811 2012-10-01        0
## 2  0.3396226 2012-10-01        5
## 3  0.1320755 2012-10-01       10
## 4  0.1509434 2012-10-01       15
## 5  0.0754717 2012-10-01       20
## 6  2.0943396 2012-10-01       25
## 7  0.5283019 2012-10-01       30
## 8  0.8679245 2012-10-01       35
## 9  0.0000000 2012-10-01       40
## 10 1.4716981 2012-10-01       45
## ..       ...        ...      ...
```

```
Activity.clean.day.sum <- summarise(Activity.clean.day, total= sum(steps))


g <- ggplot(Activity.clean.day.sum, aes(total))
g + geom_histogram(binwidth = 5000) + ggtitle("Histogram of total number of steps per day") + xlab("Tota
```




```
mean <- trunc(mean(Activity.clean.day.sum$total, na.rm = T))
median <- trunc(median(Activity.clean.day.sum$total, na.rm = T))
```

Mean and median show slight differences between both datasets.

Are there differences in activity patterns between weekdays and weekends?

For this part the weekdays() function may be of some help here. Use the dataset with the filled-in missing values for this part.

Create a new factor variable in the dataset with two levels - "weekday" and "weekend" indicating whether a given date is a weekday or weekend day.

Make a panel plot containing a time series plot (i.e. type = "l") of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all weekday days or weekend days (y-axis). See the README file in the GitHub repository to see an example of what this plot should look like using simulated data.

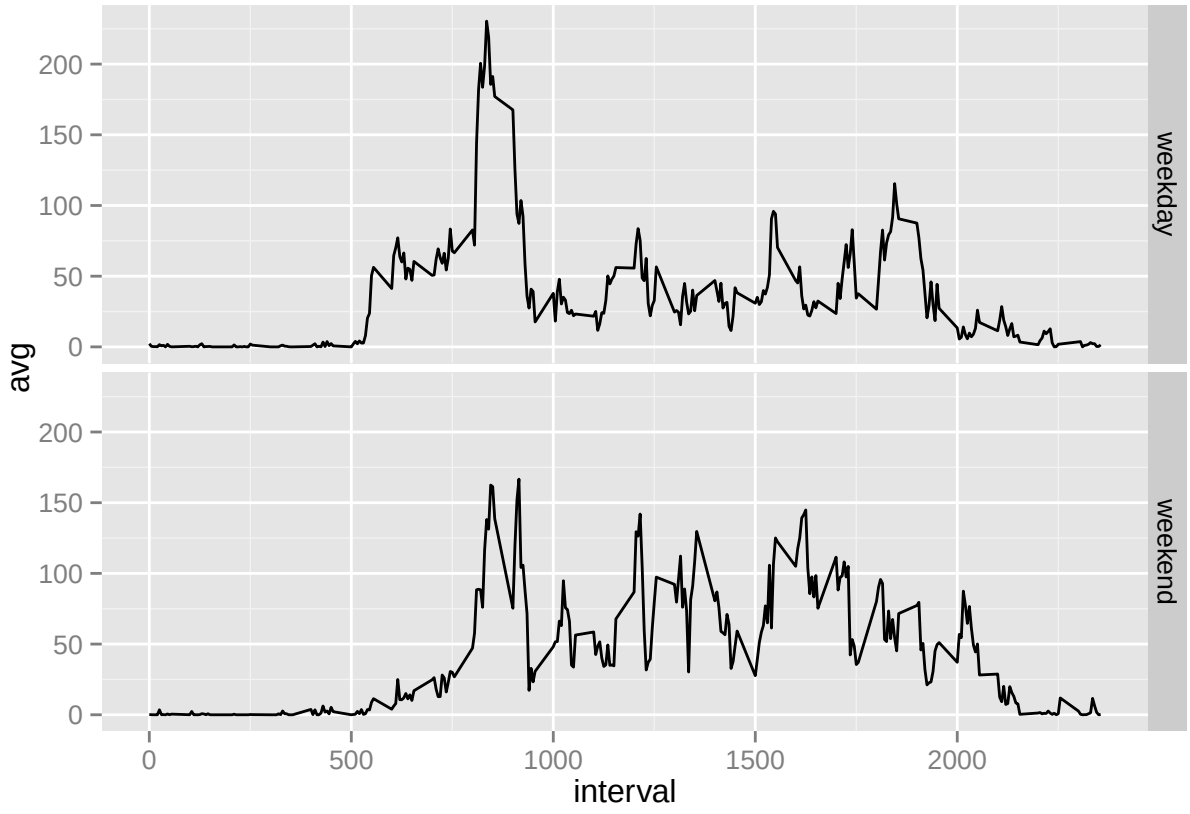
```
#New factor variable
Activity.clean$wd <- ifelse(weekdays(Activity.clean$date) %in% c("sábado", "domingo"), "weekend", "week
#Activity.clean$wd <- ifelse((wday(Activity.clean$date) == 1 || wday(Activity.clean$date) == 7),"weeken

Activity.clean$wd<- as.factor(Activity.clean$wd)
```

```
Activity.clean.wd <- group_by(Activity.clean, interval, wd)
Activity.clean.wd.avg <- summarise(Activity.clean.wd, avg = mean(steps))



g <- ggplot(Activity.clean.wd.avg, aes(interval, avg))
g + geom_line() + facet_grid(wd ~ .)
```



Yes, it seems there are a lot of differences between weekdays and weekends. People tend to wake up later. During weekdays the activity peak is at 8:35 am whereas in the weekend the peaks are around 10:00 am and 4:00 pm