# Reproducible Research, Coursera. Peer Assessment 1.

Víctor M. Medina

Nov. 14 2015

## 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.

## Data

The data used for this assignment was downloaded from the course web site: Activity monitoring data.

## Loading and preprocessing data

Open the data set assuming data are already downloaded to your working directory with the name "activicty.csv".

In order to be able to process and transform data, and create some of the plots showed here, open ggplot2 and dplyr libraries.

library(dplyr)

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

library(ggplot2)

Open the data set assuming data are already downloaded to your working directory with the name "activity.csv" and convert it to a tbl class object.

datos <- read.csv("activity.csv")  
datbl <- tbl\_df(datos)

## 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.

1. Calculate the total number of steps taken per day.

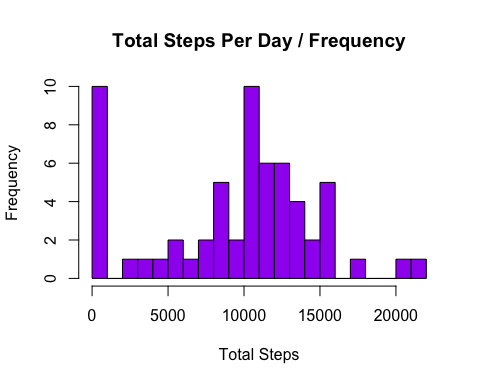
Create an object "datact" by using group\_by function and use summarise function to aggregate steps by day. Create object "datsum" to select and sum the desired columns.

datact <- datbl %>% group\_by(date) %>% summarise(total.steps = sum(steps))  
datsum<- tapply(datbl$steps, datbl$date, sum, na.rm=TRUE)

1. Make a histogram of the total number of steps taken each day. Calculate and report the mean and median of the total number of steps taken per day.

Use "hist" function to create the requested plot. With breaks set to 30.

hist(datsum, breaks=30, xlab="Total Steps",   
 main="Total Steps Per Day / Frequency", col="purple")



1. Calculate and report the mean and median of the total number of steps taken per day.

mean((datact$total.steps), na.rm=TRUE)

## [1] 10766.19

median((datact$total.steps), na.rm=TRUE)

## [1] 10765

The mean of the total number of steps taken per day is 10766.19

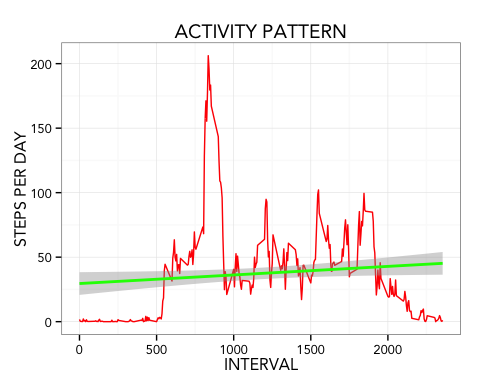
The median of total number of steps taken per day is 10765

## What is the average daily activity pattern?

1. 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)

First aggregate the total steps per interval and obtain the mean using "mean" function. Then plot the results with ggplot2 using geom\_l to obtain a linear plot, add smooth to better understand the pattern by interval.

steps.inter <- aggregate(steps ~ interval, data = datos, FUN = mean)  
  
ggplot(steps.inter, aes(x=interval, y=steps)) +   
 labs(title="ACTIVITY PATTERN", x="INTERVAL", y="STEPS PER DAY") +  
 geom\_line(color="red") +  
 geom\_smooth(method = "lm",   
 col= "green", size = 1) +  
 theme\_bw(base\_family = "Avenir")



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

Obtain the interval with the highest mean of steps across the days.

steps.inter$interval[which.max(steps.inter$steps)]

## [1] 835

The interval with the highest mean of steps across the days is interval 835.

## Imputing missing values

1. Calculate and report the total number of missing values in the dataset (i.e. the total number of rows with NAs)

sum(is.na(datos$steps))

## [1] 2304

The total number of NA values is 2304.

1. Devise a strategy for filling in all of the missing values in the dataset.

Use the median number of steps to fill the missing values in data set, creating a function. (Here i obtained help from https// rpubs.com/c0d3\_k1ra/RRPA1 to create the relNA function with data and pervalue arguments)

relNA <- function(data, pervalue) {  
 cualNA <- which(is.na(data$steps))  
 corrNA <- unlist(lapply(cualNA, FUN=function(idx){  
 interval = data[idx,]$interval  
 pervalue[pervalue$interval == interval,]$steps  
 }))  
 pasosnuev <- data$steps  
 pasosnuev[cualNA] <- corrNA  
 pasosnuev}

1. Create a new dataset that is equal to the original dataset but with the missing data filled in.

Create a new data set from the relNA function.

NAllenos <- data.frame(   
 steps = relNA(datos, steps.inter),   
 date = datos$date,   
 interval = datos$interval)

A quick look of the new data set.

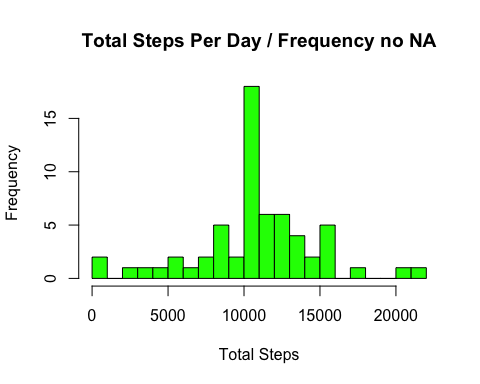
head(NAllenos)

## 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

1. Make a histogram of the total number of steps taken each day and...

First, use tapply to obtain the sum of each column by interval from the new data set, then use "hist" function to build the requested histogram.

datsum<- tapply(NAllenos$steps, NAllenos$date, sum, na.rm=TRUE)  
  
hist(datsum, breaks=30, xlab="Total Steps",   
 main="Total Steps Per Day / Frequency no NA", col="green")



...calculate and report the mean and median total number of steps taken per day.

Here, to be able to obtain the mean and median values, convert "datsum" data set to data frame and again to tbl\_df object, then re-calculate the sum of steps per interval.

datsum\_1 <- as.data.frame(datsum)  
datsum2 <- tbl\_df(datsum\_1)  
datsum3<- tapply(NAllenos$steps, NAllenos$date, sum, na.rm=TRUE)  
mean(datsum3)

## [1] 10766.19

median(datsum3)

## [1] 10766.19

4.1. 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?

The mean and median values has no significant difference from those obtained in the first part, with no impact in the results.

## Are there differences in activity patterns between weekdays and weekends?

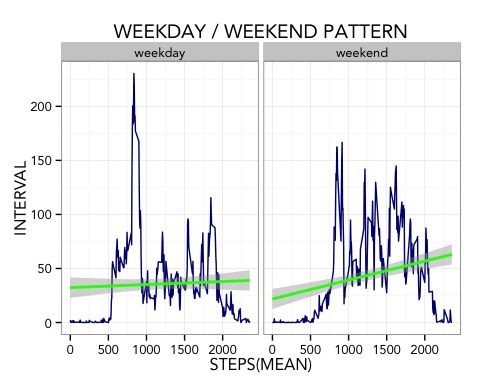
For this part the weekdays() function may be of some help here. Use the dataset with the filled-in missing values for this part.

1. 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.

NAllenos$date <- as.Date(NAllenos$date, "%Y-%m-%d")  
NAllenos$day <- weekdays(NAllenos$date)  
NAllenos$wdwe <- c("weekday")  
for (i in 1:nrow(NAllenos)){  
 if (NAllenos$day[i] == "Saturday" || NAllenos$day[i] == "Sunday"){  
 NAllenos$wdwe[i] <- "weekend"  
 }  
}  
NAllenos$wdwe <- as.factor(NAllenos$wdwe)  
wd\_we <- aggregate(steps ~ interval+wdwe, NAllenos, mean)

1. 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).

obj <- ggplot(wd\_we, aes(interval, steps))  
obj + geom\_line(color = "navyblue") +  
 geom\_smooth(method = "lm",   
 col= "green", size = 1) +  
 facet\_grid(. ~ wdwe) +  
 labs (x = "STEPS(MEAN)") +  
 labs (y = "INTERVAL") +  
 labs(title = "WEEKDAY / WEEKEND PATTERN") +  
 theme\_bw(base\_family = "Avenir")



There is a clear difference on pattern between weekdays and weekends. Although weekdays shows more activity on mornings, weekends shows more activity with a constant pattern through the day.