Reproducible Research: Peer Assessment 1

RINIE

16 Oktober 2015

## Introduction

It is now possible to collect a large amount of data about personal movement using activity monitoring devices such as a Fitbit, Nike Fuelband, or Jawbone Up. These type of devices are part of the "quantified self" movement - a group of enthusiasts who take measurements about themselves regularly to improve their health, to find patterns in their behavior, or because they are tech geeks. But these data remain under-utilized both because the raw data are hard to obtain and there is a lack of statistical methods and software for processing and interpreting the data.

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.

## Loading and Preprocessing the Data

First, read the date coercing the date column to character rather than factor

xtvtData <- read.csv("activity.csv", header = T, sep = ",", stringsAsFactors = F)

Convert the date column to the appropriate format:

xtvtData$date <- as.Date(xtvtData$date, "%Y-%m-%d")  
str(xtvtData)

## 'data.frame': 17568 obs. of 3 variables:  
## $ steps : int NA NA NA NA NA NA NA NA NA NA ...  
## $ date : Date, format: "2012-10-01" "2012-10-01" ...  
## $ interval: int 0 5 10 15 20 25 30 35 40 45 ...

## What is Mean Total Number of Steps Taken per Day?

Group and summarize the data and store it in the variable avgDay

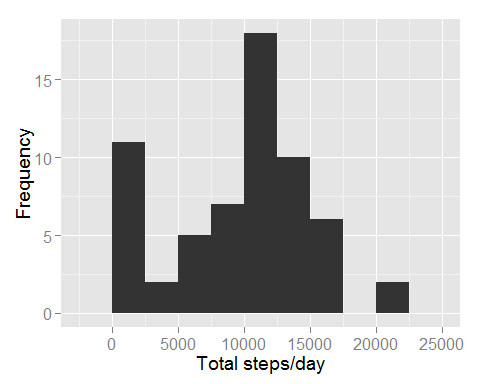
library(dplyr)

##   
## 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

avgDay <- xtvtData %>% group\_by(date) %>%  
 summarize(total.steps = sum(steps, na.rm = T),   
 mean.steps = mean(steps, na.rm = T))

Construct the histogram

library(ggplot2)  
g <- ggplot(avgDay, aes(x = total.steps))  
g + geom\_histogram(binwidth = 2500) + theme(axis.text = element\_text(size = 12),   
 axis.title = element\_text(size = 14)) + labs(y = "Frequency") + labs(x = "Total steps/day")



summary(avgDay$total.steps)

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

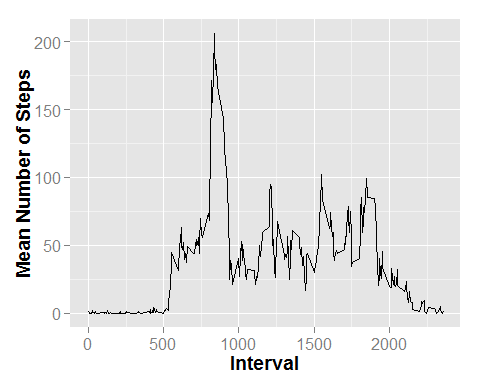
summary (avgDay$mean.steps)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 0.1424 30.7000 37.3800 37.3800 46.1600 73.5900 8

## What is the Daily Activity Pattern?

Group the data by intervaland then calculate the mean of each interval group:

avgInterval <- xtvtData %>% group\_by(interval) %>%  
 summarize(mean.steps = mean(steps, na.rm = T))  
  
g <- ggplot(avgInterval, aes(x = interval, y = mean.steps))  
  
g + geom\_line() + theme(axis.text = element\_text(size = 12),   
 axis.title = element\_text(size = 14, face = "bold")) +   
 labs(y = "Mean Number of Steps") + labs(x = "Interval")



## Imputing Missing Values

mean(is.na(xtvtData$steps))

## [1] 0.1311475

sum(is.na(xtvtData$steps))

## [1] 2304

sum(is.na(avgInterval$mean.steps))

## [1] 0

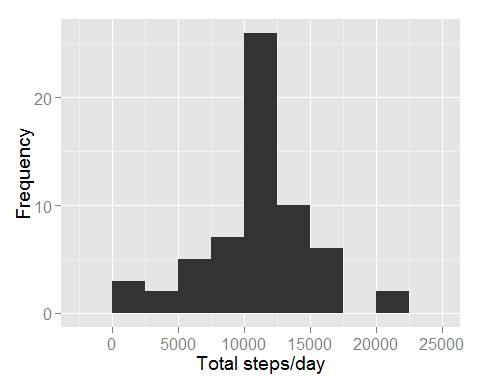
xtvtData2 <- xtvtData  
  
for (i in 1:nrow(xtvtData2)) {  
 if (is.na(xtvtData2$steps[i])) {  
 index <- xtvtData2$interval[i]  
 value <- subset(avgInterval, interval==index)  
 xtvtData2$steps[i] <- value$mean.steps  
 }  
}  
head(xtvtData2)

## 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

newAvg <- xtvtData2 %>% group\_by(date) %>%  
 summarize(total.steps = sum(steps, na.rm = T))

Construct the histogram

g <- ggplot(newAvg, aes(x=total.steps))  
g + geom\_histogram(binwidth = 2500) + theme(axis.text = element\_text(size = 12),  
 axis.title = element\_text(size = 14)) + labs(y = "Frequency") + labs(x = "Total steps/day")



summary (avgDay$total.steps)

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

sd(avgDay$total.steps, na.rm = T)

## [1] 5405.895

summary (newAvg$total.steps)

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

sd(newAvg$total.steps, na.rm = T)

## [1] 3974.391

## Are There Differences in Activity Patterns Between Weekdays and Weekends?

daytype <- function(date) {  
 if (weekdays(as.Date(date)) %in% c("Saturday", "Sunday")) {  
 "Weekend"  
 } else {  
 "Weekday"  
 }  
}  
xtvtData2$date <- as.Date(xtvtData2$date)  
xtvtData2$day <- sapply(xtvtData2$date, FUN = daytype)  
  
averages <- aggregate(steps ~ interval + day, data = xtvtData2, mean)  
ggplot(averages, aes(interval, steps)) + geom\_line() + facet\_grid(day ~ .) +   
 xlab("Interval") + ylab("Number of Steps")

