Untitled

Carlos Diaz

8 de junio de 2018

knitr::opts\_chunk$set(echo = TRUE)

library(rmarkdown)  
library(knitr)

## Warning: package 'knitr' was built under R version 3.4.4

# REPRODUCIBLE RESEARCH COURSE - DATA SCIENCE SPECIALIZATION

## ASSIGNMENT: COURSE PROJECT 1

### 1. Code for reading in the dataset and/or processing the data:

## Reading csv datafile:   
DataActivity <- read.csv("activity.csv")  
str(DataActivity)

## 'data.frame': 17568 obs. of 3 variables:  
## $ steps : int NA NA NA NA NA NA NA NA NA NA ...  
## $ date : Factor w/ 61 levels "2012-10-01","2012-10-02",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ interval: int 0 5 10 15 20 25 30 35 40 45 ...

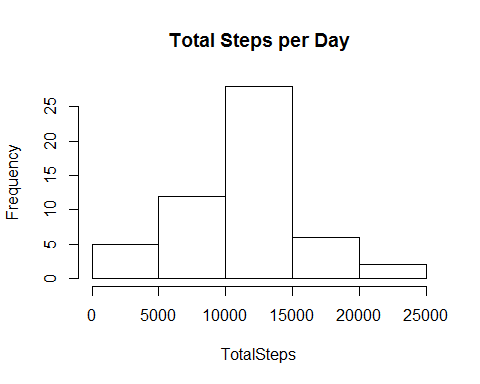
DataActivityNAs <- na.omit(DataActivity)  
str(DataActivityNAs)

## 'data.frame': 15264 obs. of 3 variables:  
## $ steps : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ date : Factor w/ 61 levels "2012-10-01","2012-10-02",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ interval: int 0 5 10 15 20 25 30 35 40 45 ...  
## - attr(\*, "na.action")=Class 'omit' Named int [1:2304] 1 2 3 4 5 6 7 8 9 10 ...  
## .. ..- attr(\*, "names")= chr [1:2304] "1" "2" "3" "4" ...

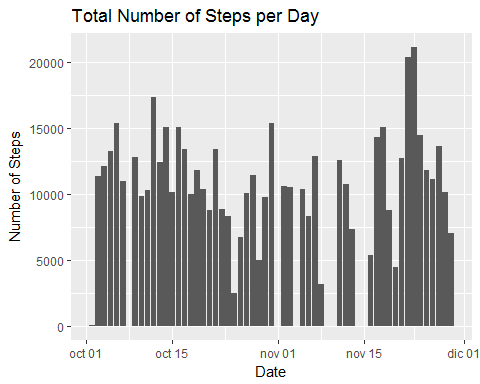
DataActivityNAs$date <- as.Date(DataActivityNAs$date, "%Y-%m-%d")

### 2. Histogram of the total number of steps taken each day:

library(ggplot2)  
TotalSteps <- tapply(DataActivityNAs$steps, DataActivityNAs$date, FUN=sum)  
hist(TotalSteps, main="Total Steps per Day")



g <- ggplot(DataActivityNAs, aes(date,steps))  
g <- g+geom\_bar(stat="identity")+labs(title="Total Number of Steps per Day", x= "Date", y="Number of Steps")  
print(g)



### 3. Mean and median number of steps taken each day:

summary(TotalSteps)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 41 8841 10765 10766 13294 21194

MeanSteps <- mean(TotalSteps)  
MeanSteps

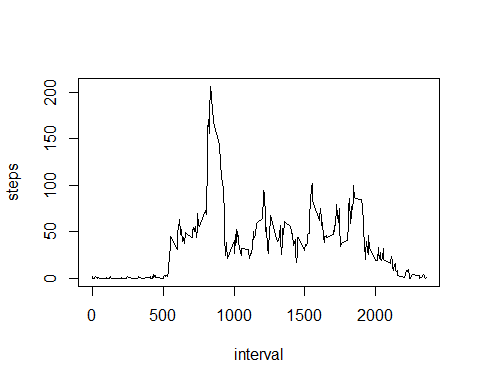
## [1] 10766.19

MedianSteps <- median(TotalSteps)  
MedianSteps

## [1] 10765

### 4. Time series plot of the average number of steps taken:

AvgDailySteps <- aggregate(steps~interval, data=DataActivityNAs, FUN=mean)  
plot1 <- plot(AvgDailySteps, type="l")



### 5. The 5-minute interval that, on average, contains the maximum number of steps:

IntMaxSteps <- AvgDailySteps$interval[which.max(AvgDailySteps$steps)]  
print(IntMaxSteps)

## [1] 835

### 6. Code to describe and show a strategy for imputing missing data:

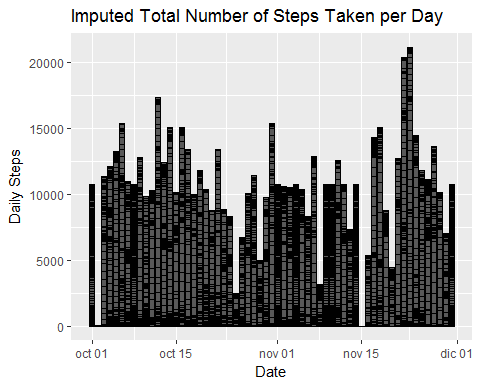
## Creating a new dataset that is equal to the original but with the missing data filled in.   
ImputedData <- merge(DataActivity, AvgDailySteps, by= "interval", suffixes = c("", ".y"))  
NAvalues <- is.na(ImputedData$steps)  
ImputedData$steps[NAvalues] <- ImputedData$steps.y[NAvalues]  
ImputedData <- ImputedData[, c(1:3)]  
sum(is.na(ImputedData$steps))

## [1] 0

ImputedData$date <- as.Date(ImputedData$date)

### 7. Histogram of the total number of steps taken each day after missing values are imputed:

g2 <- ggplot(ImputedData , aes(date, steps))  
plot2 <- g2+geom\_bar(stat="identity", color = "black")+labs(title="Imputed Total Number of Steps Taken per Day", x="Date", y="Daily Steps")  
print(plot2)



### 8. Panel plot comparing the average number of steps taken per 5-minute interval across weekdays and weekends:

ImputedDatadays <- ImputedData  
Weekenddays <- weekdays((ImputedDatadays$date)) %in% c("Saturday", "Sunday")  
ImputedDatadays$daytype <- "Weekday"  
ImputedDatadays$daytype[Weekenddays == TRUE] <- "Weekend"  
ImputedDatadays$daytype <- as.factor(ImputedDatadays$daytype)  
  
## Daily average steps by daytype  
AvgStepsByDayType <- aggregate(steps ~ interval + daytype, ImputedDatadays, mean)  
names(AvgStepsByDayType)[3] <- "Avg\_Steps"  
  
library(lattice)  
  
plot3 <- xyplot(Avg\_Steps ~ interval | daytype, AvgStepsByDayType, type="l", layout=c(1,2), main = "Weekend vs Weekday Average Number of Steps", xlab="5 min Interval", ylab="Avg Steps taken")  
print(plot3)

