Analyzing FitBit Data

Prabhu Thaipulley

Nov, 2016

About

This was the first project for the **Reproducible Research** course in Coursera's Data Science specialization track. The purpose of the project was to answer a series of questions using data collected from a FitBit.

Synopsis

The purpose of this project was to practice:

- loading and preprocessing data
- imputing missing values
- interpreting data to answer research questions

The dataset is stored in a comma-separated-value (CSV) file and there are a total of 17,568 observations in this dataset.

Loading and preprocessing the data

Download, unzip and load data into data frame `data`.

Loading the data

```
## setwd("/Users/tsprabhu/Downloads/RepData_PeerAssessment1-master")
data <- read.csv("activity.csv", header=TRUE)
```

Histogram of Steps per Day

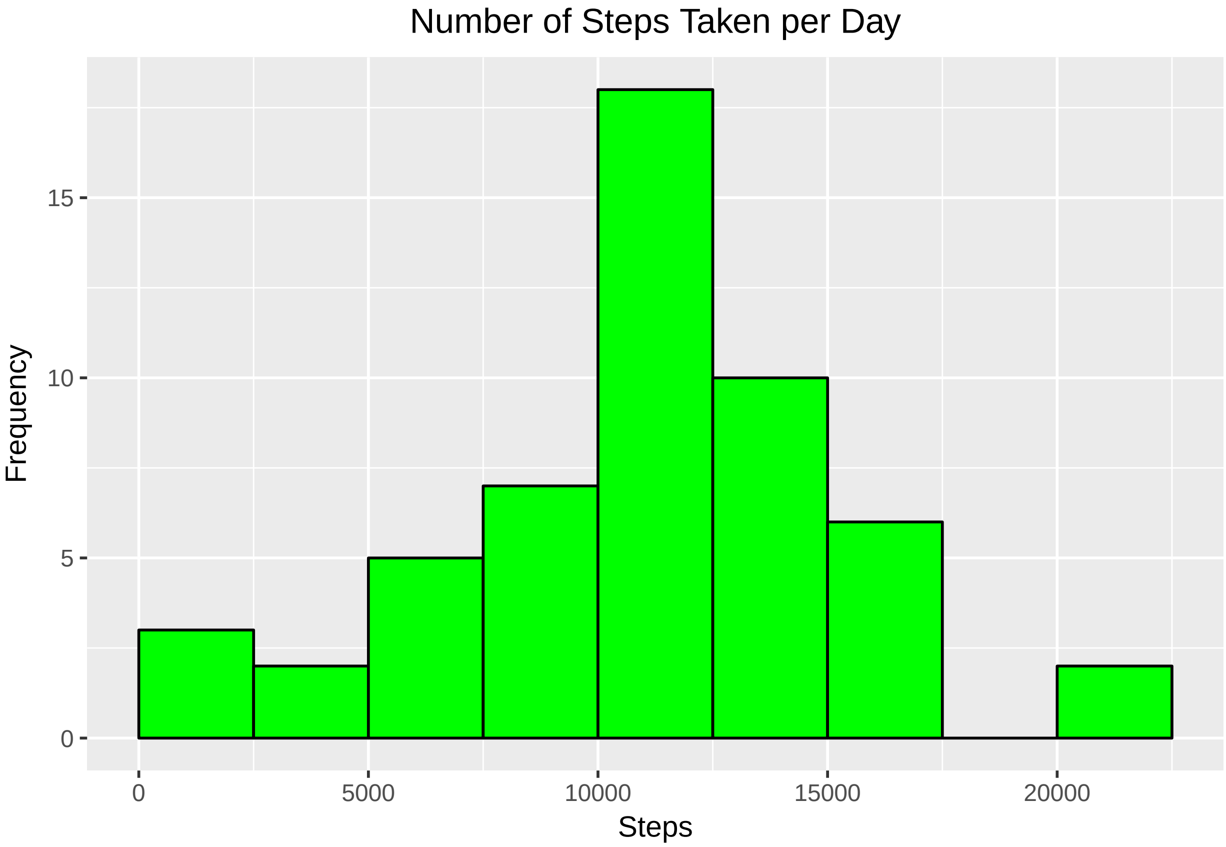
Aggregate the number of steps taken per day and create a histogram of the aggregated data.

```
stepsPerDay <- aggregate(list(Steps=data$step), by=list(Date=data$date), FUN=sum)
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.3.2
```

```
ggplot(stepsPerDay, aes(x=Steps, col=Date)) +
    geom_histogram(breaks=seq(0, 22500, by=2500), fill="green", col="black") +
    ylab("Frequency") +
    ggtitle("Number of Steps Taken per Day") +
     theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Removed 8 rows containing non-finite values (stat_bin).
```



Mean and median of the total number of steps taken each day

```
options(scipen = 999)
stepsMean <- mean(stepsPerDay$Steps, na.rm=T)
stepsMedian <- median(stepsPerDay$Steps, na.rm=T)
```

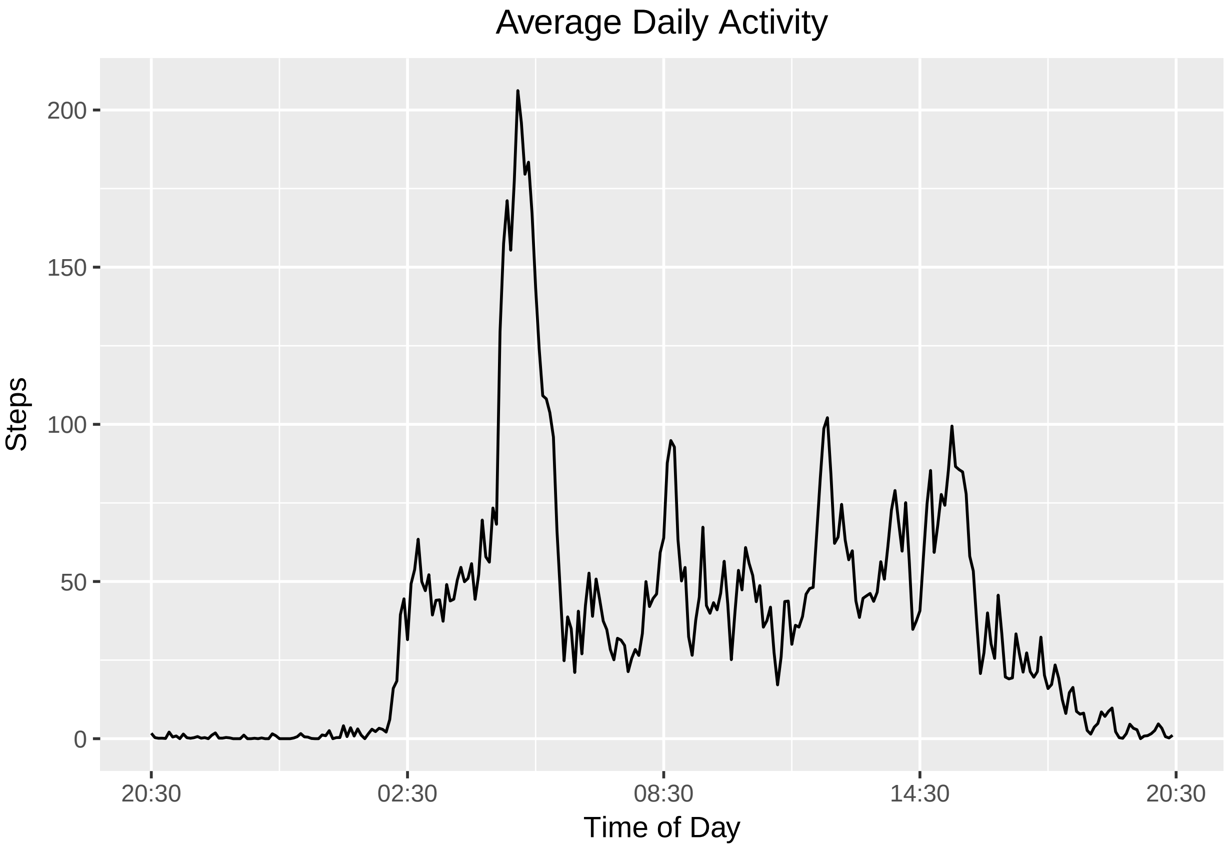
The mean of the steps taken per day is **10766.1886792** and the corresponding median is **10765**.

Average daily activity pattern and the interval with most activity

Take the mean of steps taken during each time interval and create a time series graph of the averaged data. Also find maximum activity interval

```
stepsInterval <- aggregate(list(Steps=data$step), by=list(Interval=data$interval), FUN=mean, na.rm=T)

# Add column in which time is in time format
stepsInterval <- transform(stepsInterval, timeOfDay=strptime( paste(formatC(Interval,width=4,flag="0"))
library(ggplot2)
library(scales)
ggplot(stepsInterval, aes(x=timeOfDay, y=Steps)) +
    geom_line() +
    xlab("Time of Day") +
     ggtitle("Average Daily Activity") +
     theme(plot.title = element_text(hjust = 0.5))  +
     scale_x_datetime(labels=date_format("%H:%M", tz="UTC-2"))
```



```
stepsInterval$timeOfDay[which.max(stepsInterval$Steps)]
```

```
## [1] "2017-01-20 08:35:00 IST"
```

The interval with most activity on average is **between 08:35 and 08:40 AM**.

Imputing missing values

```
missing <- sum(is.na.data.frame(data))
```

The number of rows with missing values is **2304**.

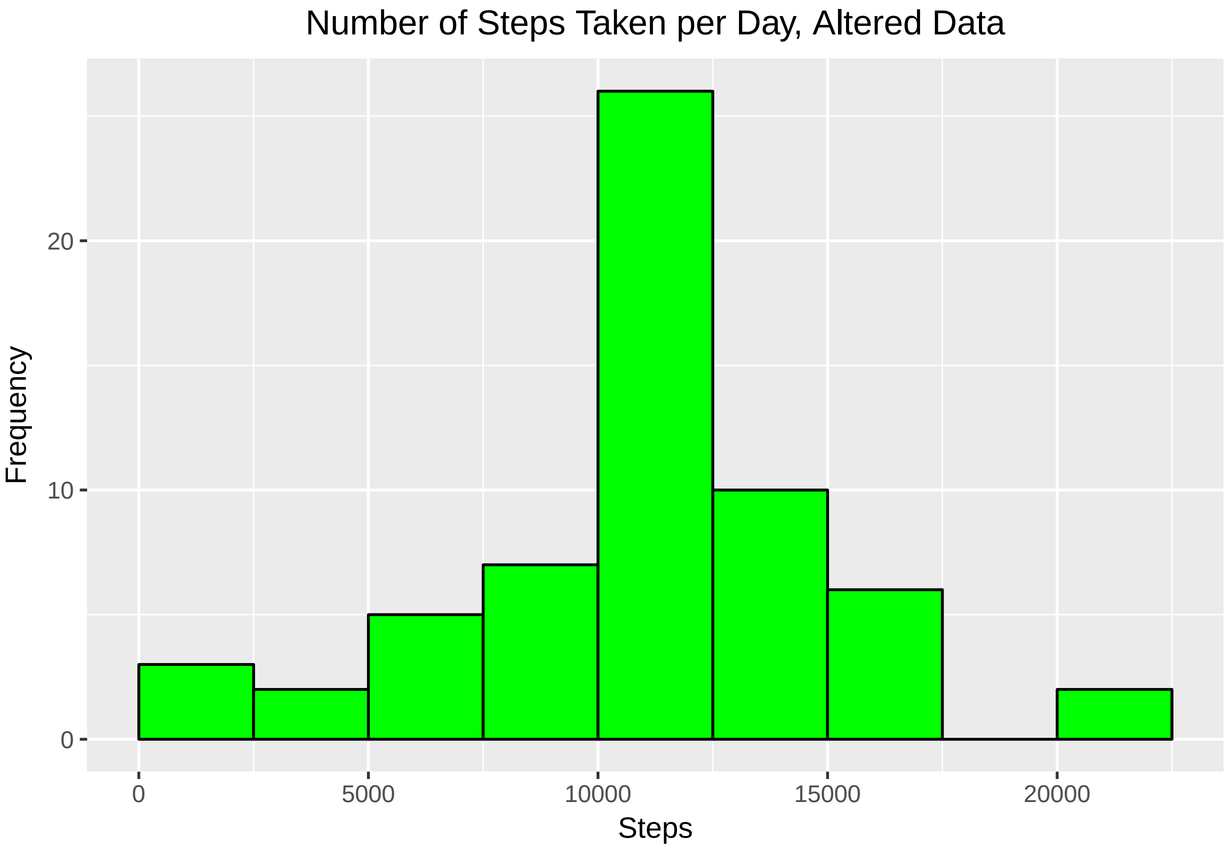
Replace the number of steps missing by interval mean.

```
x1<-NULL
moddata <- data
for (i in 1:length(data$steps)) {
    if(is.na(data$steps[i])==TRUE) {
        x1 <- subset(stepsInterval, Interval == data$interval[i])
        moddata$steps[i] <- x1$Steps
    }

}
```

Aggregate the number of steps taken per day in the modified data and create a histogram of the aggregated data.

```
stepsPerDayMod <- aggregate(list(Steps=moddata$step), by=list(Date=moddata$date), FUN=sum)
library(ggplot2)
ggplot(stepsPerDayMod, aes(x=Steps, col=Date)) +
    geom_histogram(breaks=seq(0, 22500, by=2500), fill="green", col="black") +
    ylab("Frequency") +
    ggtitle("Number of Steps Taken per Day, Altered Data") +
     theme(plot.title = element_text(hjust = 0.5))
```



```
options(scipen = 999)
stepsMeanMod <- mean(stepsPerDayMod$Steps)
stepsMedianMod <- median(stepsPerDayMod$Steps)
```

The mean of the steps taken per day is now **10766.1886792** and the corresponding median is **10766.1886792**.Replacing the missing values witht he interval mean does not affect mean steps taken per day, for obvious reasons. The median was very close to the mean to begin with, and after replacing NAs the mean and median are equal.

Weekdays vs weekends

Transform dates into appropriate format.

```
moddata <- transform(moddata,
                      date = strptime( paste(date,formatC(interval,width=4,flag="0")), "%Y-%m-%d"))
```

Create time series graph of mean activity during weekdays and weekends.

```
Sys.setlocale("LC_TIME", "C")
```

```
## [1] "C"
```

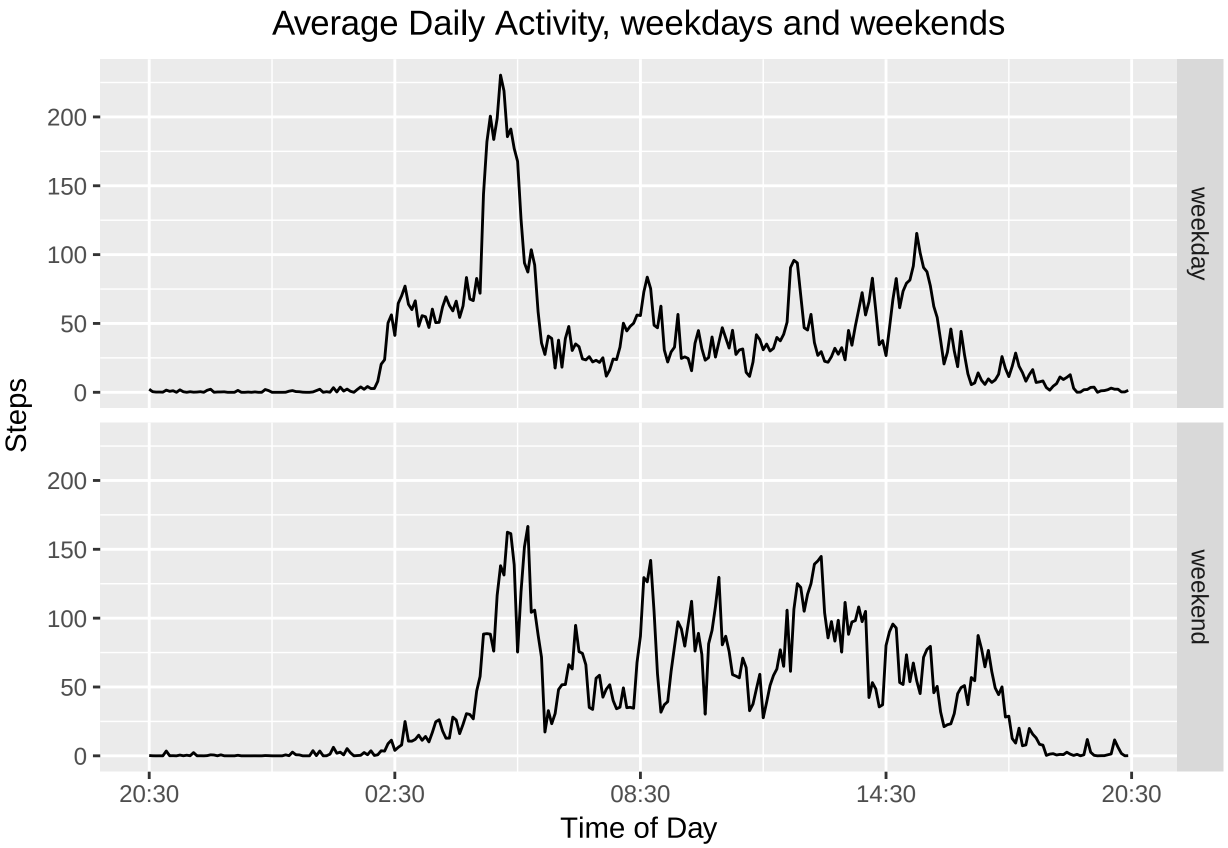
```
weekDay <- with(moddata,
                ifelse(weekdays(date) %in% c("Saturday","Sunday"),"weekend","weekday"))

moddata$weekDay <- as.factor(weekDay)

stepsInterval2 <- aggregate(list(Steps=moddata$step), by=list(Interval=moddata$interval, weekDay=moddata


# Add column in which time is in time format
stepsInterval2 <- transform(stepsInterval2, timeOfDay=strptime( paste(formatC(Interval,width=4,flag="0"


 library(ggplot2)
 library(scales)
 ggplot(stepsInterval2, aes(x=timeOfDay, y=Steps)) +
     facet_grid(weekDay~.) +
     geom_line() +
     xlab("Time of Day") +
      ggtitle("Average Daily Activity, weekdays and weekends") +
      theme(plot.title = element_text(hjust = 0.5)) +
      scale_x_datetime(labels=date_format("%H:%M", tz="UTF-2"))
```


Average Daily Activity, weekdays and weekends
Steps
200
150
100
50
0
weekday
weekend
20:30
02:30
08:30
14:30
20:30
Time of Day