Analysis of Personal Movement Using Activity Monitoring Devices

Sergei Abramov

1/24/2021

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 project 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. The goal of this project is to write a report that answers the questions detailed below Loading and preprocessing the data.

```
#require Packages
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.6.3
```

```
##
## 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)
```

```
## Warning: package 'ggplot2' was built under R version 3.6.3
```

```
library(gridExtra)
```

```
## Warning: package 'gridExtra' was built under R version 3.6.3
```

```
## 
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
## 
##     combine
```

```
library(imputeTS)
```

```
## Warning: package 'imputeTS' was built under R version 3.6.3
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
setwd("~/Coursera/Reproducible Research")
dest <-"activity.csv"
activ_3<-read.csv(dest,stringsAsFactors = F,na.strings = "NA")
activ_3$date <- as.Date(activ_3$date)
```

Subsetting the dataset to ignore missing values

```
activ_3 <- na.omit(activ_3)
```

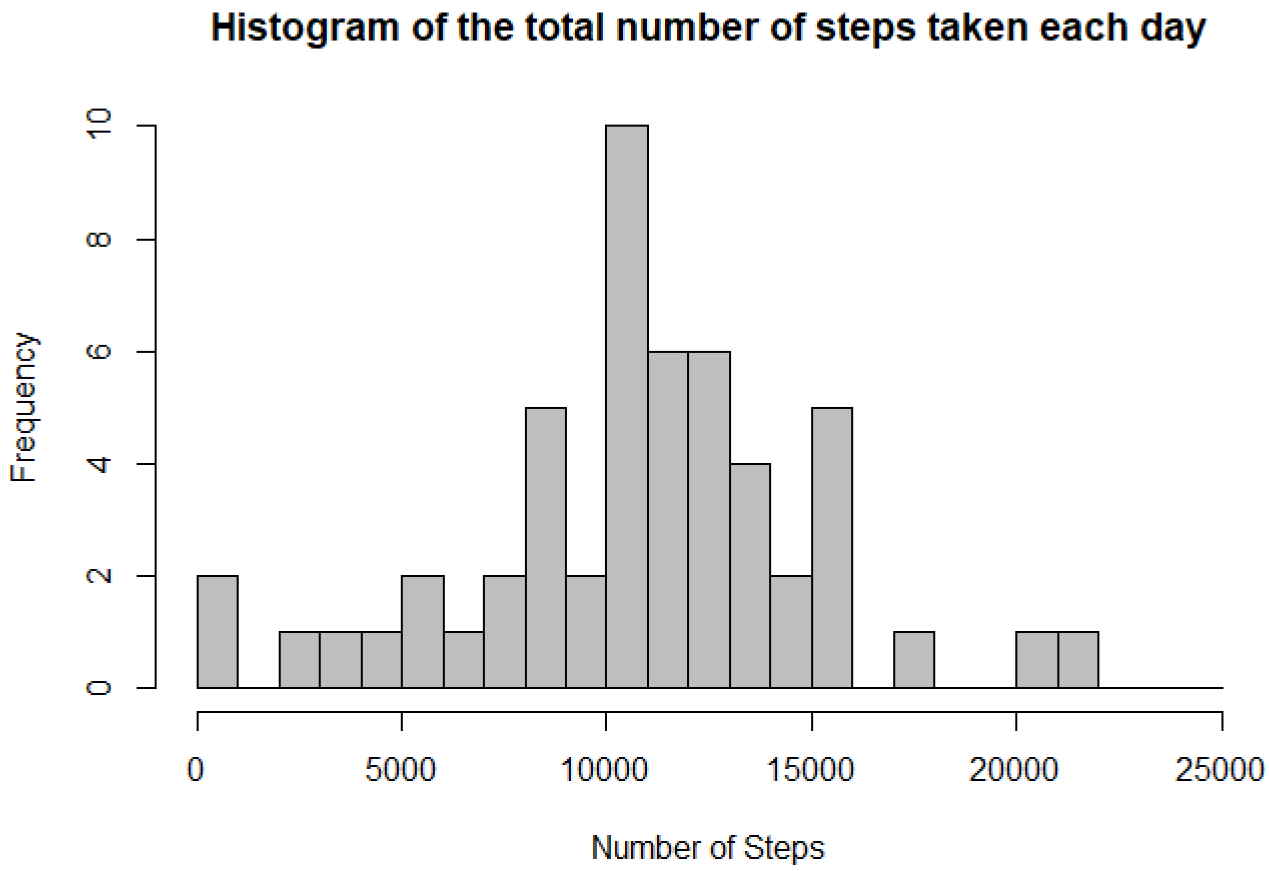
Aggregating the number of steps taken each day in dplyr packadge

```
res <- activ_3 %>%
   group_by(date) %>%
   #summarise_each(funs(mean))
   summarise(sum = sum(steps), n = n())
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

Making a histogram of the total number of steps taken each day:

```
hist(res$sum,breaks = seq(0,25000, by=1000),xlab = "Number of Steps", col = "gray", main= "Histogram of the total number of steps taken each day")
```



Calculate mean and median numbers of steps taken per day:

```
mean_Value<- mean(res$sum)
mean_Value
```

```
## [1] 10766.19
```

```
median_Value<- median(res$sum)
median_Value
```

```
## [1] 10765
```

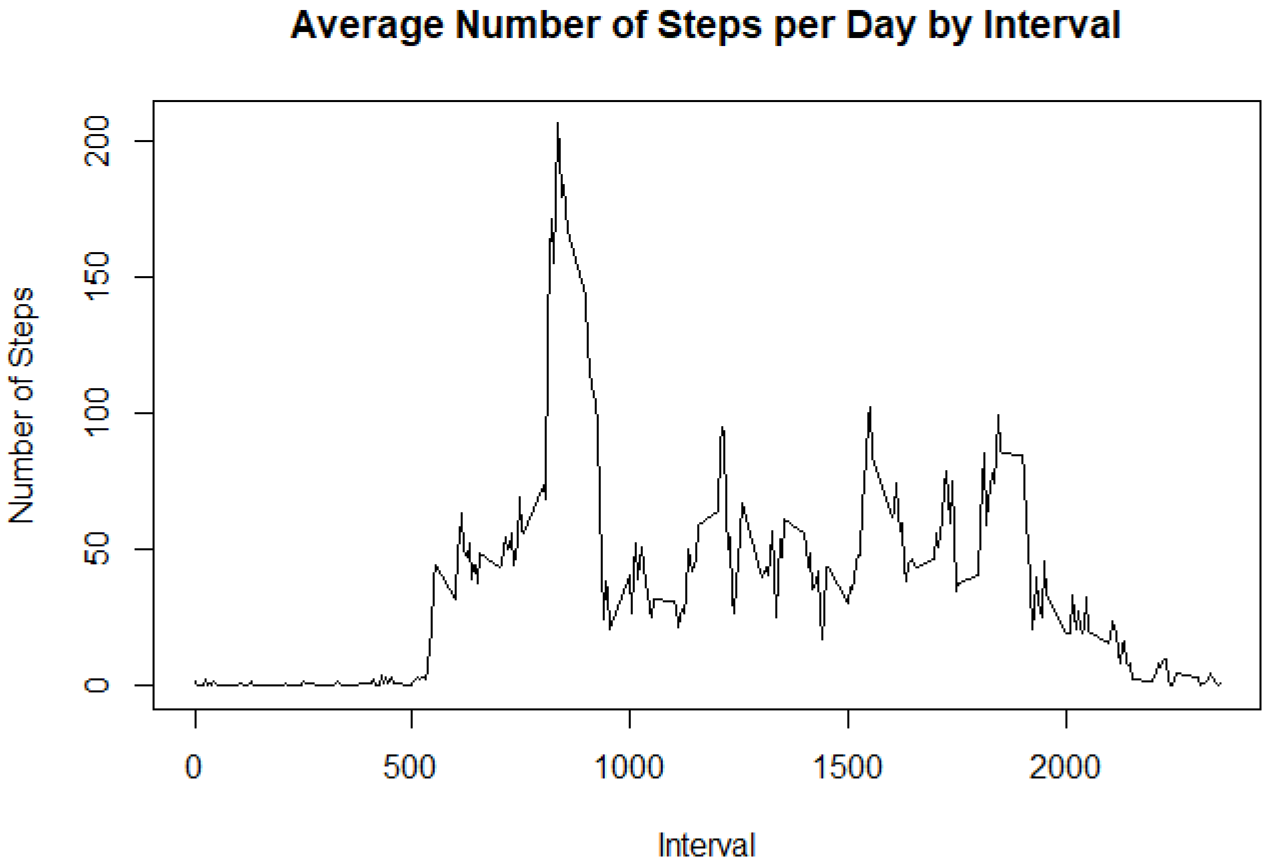
What is the average daily activity pattern? Calculating the average number of steps taken, averaged across all days:

```
res1 <- activ_3 %>%
    group_by(interval) %>%
    #summarise_each(funs(mean))
    summarise(mean = mean(steps), n = n())
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

Plot the Average Number Steps per Day by Interval

```
plot(res1$interval,res1$mean,type="l", xlab="Interval",ylab ="Number of Steps",main= "Average Number of Steps per Day by Interval")
```



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

```
 max_interval <-res1[which.max(res1$mean),]
 max_interval
```

```
## # A tibble: 1 x 3
##   interval  mean     n
##      <int> <dbl> <int>
## 1      835  206.    53
```

# A tibble: 1 x 3 # interval mean n
# int dbl int
# 835 206. 53

Imputing missing values

Load Raw data again without cleaning from “NA” value

```
activ_4<-read.csv(dest,stringsAsFactors = F)
activ_4$date <- as.Date(activ_4$date)
```

The total number of missing values in the dataset (for each variable): For the “steps” variable:

```
test <-
   activ_4 %>%
   filter(is.na(steps))
  nrow(test)
```

```
## [1] 2304
```

For the “date” variable:

```
 test <-
    activ_4 %>%
    filter(is.na(date))
nrow(test)
```

```
## [1] 0
```

For the “interval” variable:

```
test <-
   activ_4 %>%
   filter(is.na(interval))
  nrow(test)
```

```
## [1] 0
```

Now, Using Mean for the day compute missing values. Create a new dataset including the imputed missing values. Load library(imputeTS) and replace missing values with column mean.

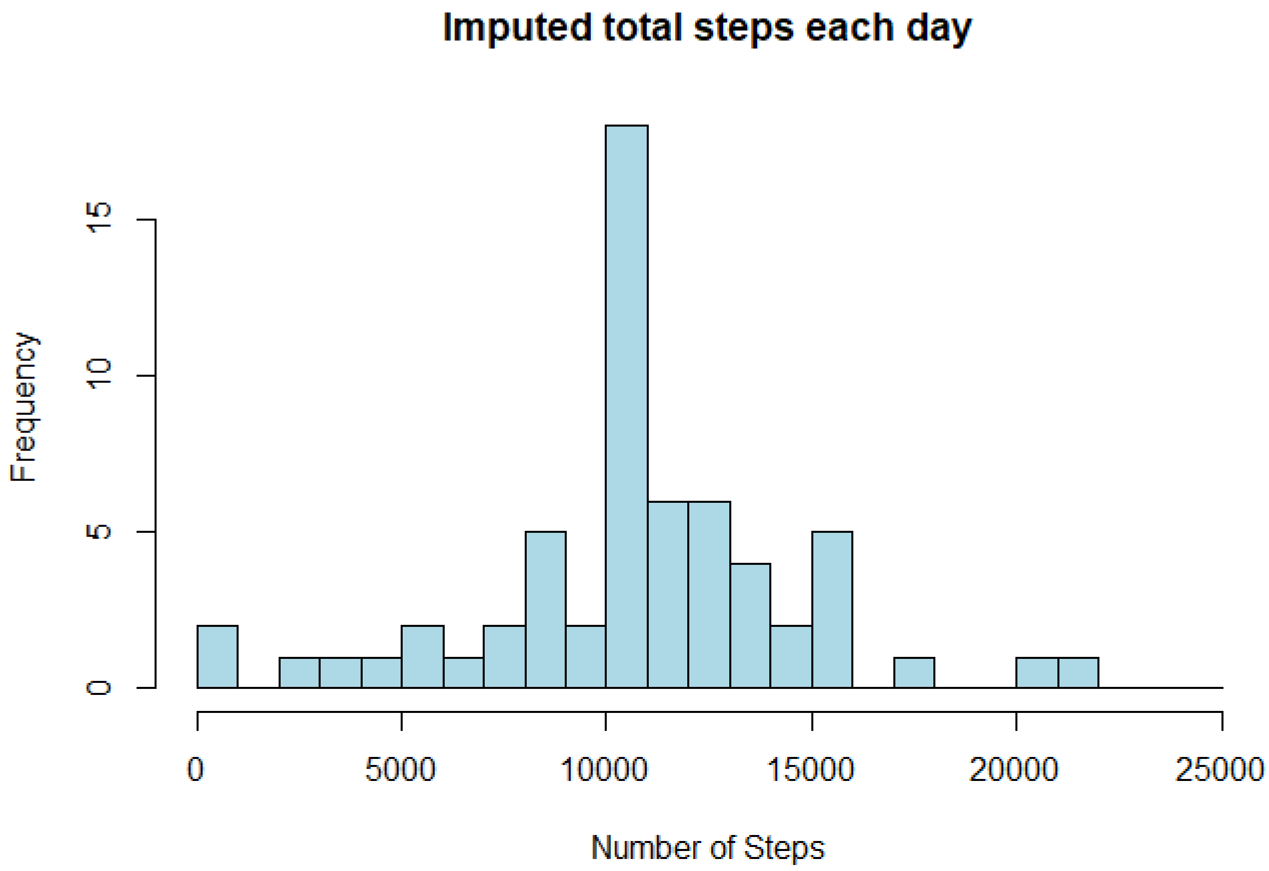
```
activ_5<- na_mean(activ_4)
```

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

```
  res2 <- activ_5 %>%
  group_by(date) %>%
  summarise(sum = sum(steps), n = n())
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
 hist(res2$sum,breaks = seq(0,25000, by=1000),xlab = "Number of Steps", col = "light blue", m
ain= "Imputed total steps each day")
```



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

Mean number of steps taken per day with impute mean data.

```
mean_Value_Impute<- mean(res2$sum)
mean_Value_Impute
```

```
## [1] 10766.19
```

Median number of steps taken per day with impute mean data.

```
median_Value_Impute<- median(res2$sum)
median_Value_Impute
```

```
## [1] 10766.19
```

Are there differences in activity patterns between weekdays and weekends?

Creating a variable “weekday”to store the day of the week:

```
  activ_5$weekday<-weekdays(activ_5$date)
```

Creating a logical variable “is_weekday” (weekday=TRUE, weekend = FALSE) and subsetting data by this logical value.

```
  activ_5$is_weekday <- ifelse(!(activ_5$weekday %in% c("Saturday","Sunday")), TRUE, FALSE)
    weekdays_data <- subset(activ_5,activ_5$is_weekday==T,)
    weekend_data <- subset(activ_5,activ_5$is_weekday==F,)
```

Calculate average steps for each interval for all week days:

```
    res4 <- weekdays_data %>%
   group_by(interval) %>%
   summarise(mean = mean(steps), n = n())
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

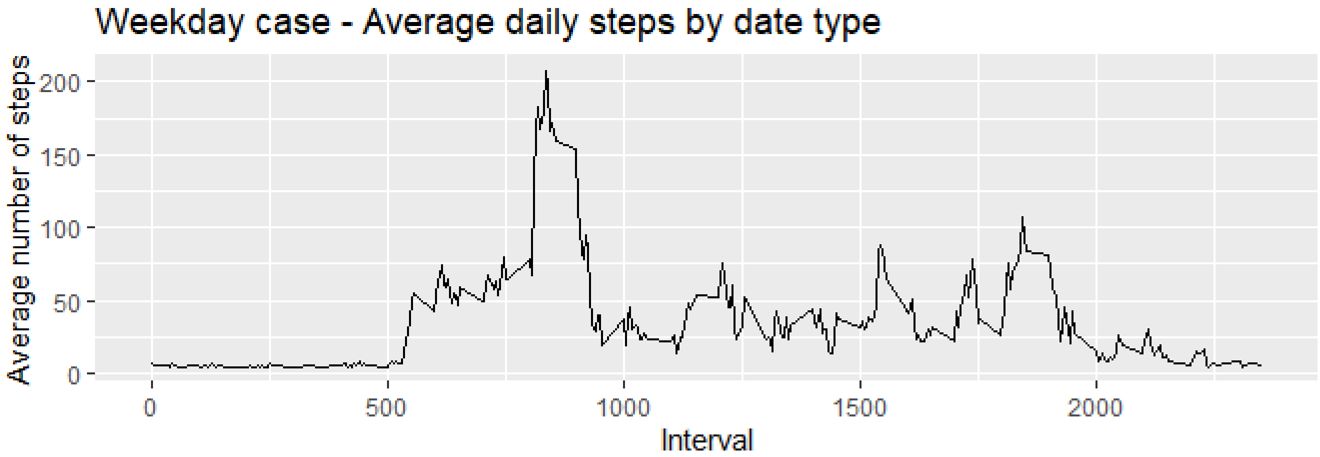
Calculate average steps for each interval for weekend days:

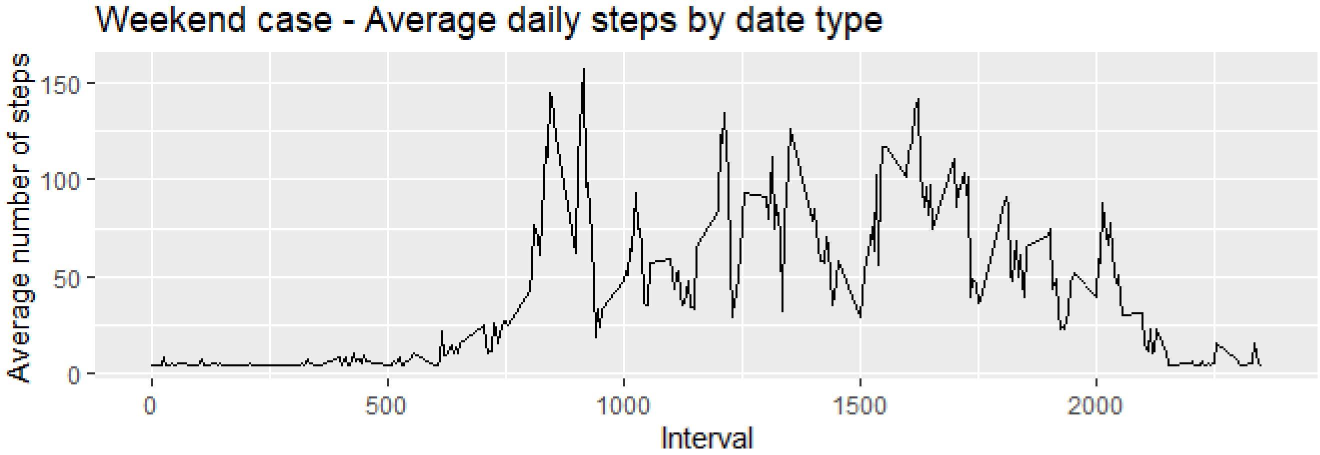
```
    res5 <- weekend_data %>%
   group_by(interval) %>%
   summarise(mean = mean(steps), n = n())
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

Created a plot to compare and contrast number of steps between the week and weekend.

```
     p1<-ggplot(res4, aes(x = interval, y = mean))+ geom_line() + labs(title = "Weekday case
- Average daily steps by date type", x = "Interval", y = "Average number of steps")
    p2<-ggplot(res5, aes(x = interval, y = mean))+ geom_line() + labs(title = "Weekend case -
Average daily steps by date type", x = "Interval", y = "Average number of steps")
    grid.arrange(p1, p2, nrow = 2)
```





There is a higher peakearlier on weekdays, and more overall activity on weekends.
The plot shows that that activity on the weekends tends to be more spread out over the day compared to the weekdays.