Peer assesment 1_reproducible research

Marijan Mihaldinec

2nd November 2017

Table of Contents

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.

The data for this assignment can be downloaded from the course web site:

Dataset: Activity monitoring data [52K] The variables included in this dataset are:

steps: Number of steps taking in a 5-minute interval (missing values are coded as NA) date: The date on which the measurement was taken in YYYY-MM-DD format interval: Identifier for the 5-minute interval in which measurement was taken The dataset is stored in a comma-separated-value (CSV) file and there are a total of 17,568 observations in this dataset.

Assignment

When writing code chunks in the R markdown document, always use echo = TRUE

set working directory

```
setwd("C:/Users/mihaldma/Documents/coursera/REPRODUCIBLE_RESEARCH/REPRODUCIBL
E_RESEARCH_PEER ASSESSMENT1")
```

Included packages

```
library(knitr)
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)
library(lattice)
library(reshape2)
library(lubridate)

##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date
```

Loading and preprocessing the data

```
data <- read.csv("activity.csv")
head(data)

##   steps       date interval
## 1    NA 2012-10-01        0
## 2    NA 2012-10-01        5
## 3    NA 2012-10-01       10
## 4    NA 2012-10-01       15
## 5    NA 2012-10-01       20
## 6    NA 2012-10-01       25

tail(data)

##       steps       date interval
## 17563    NA 2012-11-30     2330
## 17564    NA 2012-11-30     2335
## 17565    NA 2012-11-30     2340
## 17566    NA 2012-11-30     2345
## 17567    NA 2012-11-30     2350
## 17568    NA 2012-11-30     2355

summary(data)

##      steps                date          interval
##  Min.   :  0.00   2012-10-01:  288   Min.   :   0.0
```

```
##  1st Qu.:  0.00   2012-10-02:  288   1st Qu.: 588.8
##  Median :  0.00   2012-10-03:  288   Median :1177.5
##  Mean   : 37.38   2012-10-04:  288   Mean   :1177.5
##  3rd Qu.: 12.00   2012-10-05:  288   3rd Qu.:1766.2
##  Max.   :806.00   2012-10-06:  288   Max.   :2355.0
##  NA's   :2304     (Other)   :15840
```

```
str(data)
```

```
## '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 ...
```

Process/transform the data => format suitable for your analysis

```
dt = Sys.time()

date <- format(dt,"%d-%b-%Y")

time <- format(dt,"%H:%M:%S")

str(data)
```

```
## '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 ...
```

```
head(data)
```

```
##   steps       date interval
## 1    NA 2012-10-01        0
## 2    NA 2012-10-01        5
## 3    NA 2012-10-01       10
## 4    NA 2012-10-01       15
## 5    NA 2012-10-01       20
## 6    NA 2012-10-01       25
```

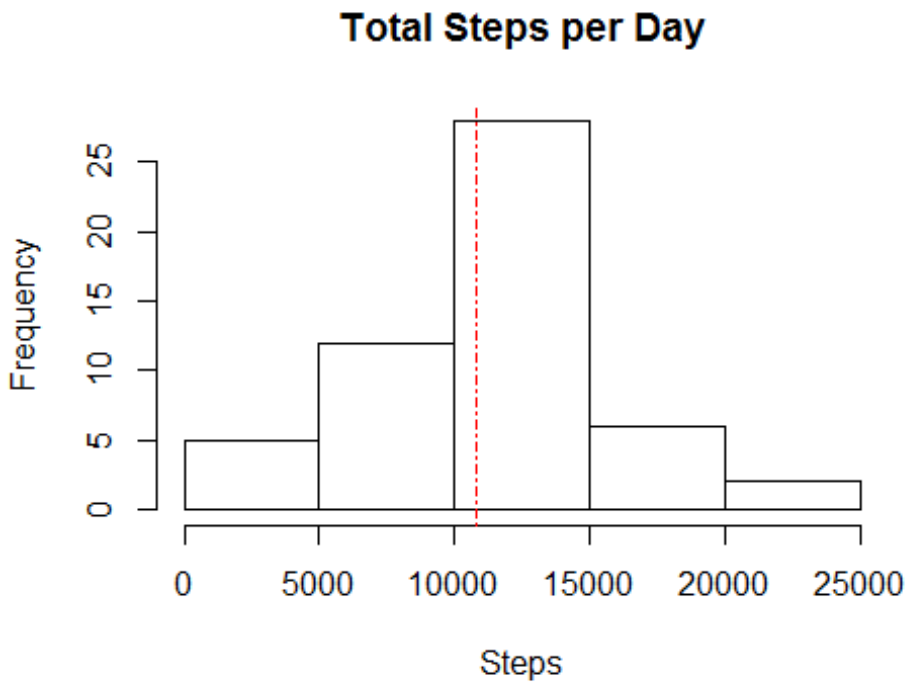
```
summary(data)
```

```
##      steps               date          interval
##  Min.   :  0.00   2012-10-01:  288   Min.   :   0.0
##  1st Qu.:  0.00   2012-10-02:  288   1st Qu.: 588.8
##  Median :  0.00   2012-10-03:  288   Median :1177.5
##  Mean   : 37.38   2012-10-04:  288   Mean   :1177.5
##  3rd Qu.: 12.00   2012-10-05:  288   3rd Qu.:1766.2
##  Max.   :806.00   2012-10-06:  288   Max.   :2355.0
##  NA's   :2304     (Other)   :15840
```

```
datatransform <- na.omit(data)

head(datatransform)

##     steps       date interval
## 289     0 2012-10-02        0
## 290     0 2012-10-02        5
## 291     0 2012-10-02       10
## 292     0 2012-10-02       15
## 293     0 2012-10-02       20
## 294     0 2012-10-02       25

summary(datatransform)

##      steps                date          interval
##  Min.   :  0.00   2012-10-02:  288   Min.   :   0.0
##  1st Qu.:  0.00   2012-10-03:  288   1st Qu.: 588.8
##  Median :  0.00   2012-10-04:  288   Median :1177.5
##  Mean   : 37.38   2012-10-05:  288   Mean   :1177.5
##  3rd Qu.: 12.00   2012-10-06:  288   3rd Qu.:1766.2
##  Max.   :806.00   2012-10-07:  288   Max.   :2355.0
##                   (Other)   :13536
```

What is mean total number of steps taken per day?

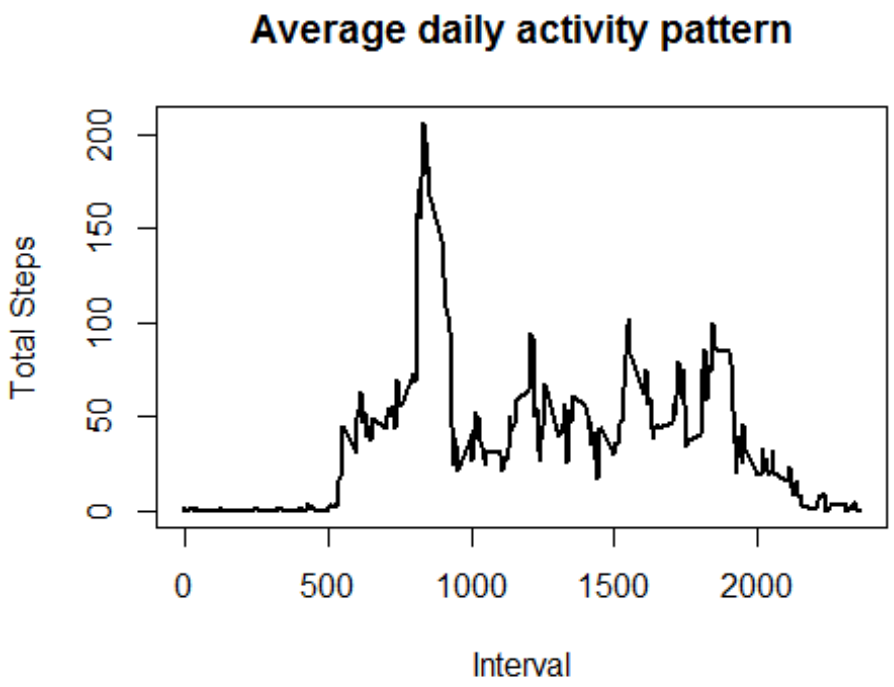
```
sumsteps<- aggregate(steps ~ date, datatransform, FUN=sum)

hist(sumsteps$steps, main= "Total Steps per Day", xlab="Steps")

abline(v=median(sumsteps$steps), lty=4, col="red")
```




```
summary(sumsteps$steps)

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

What is the average daily activity pattern?

```
stepsinterval<- aggregate(steps ~ interval, datatransform, mean, na.rm =
TRUE)

plot (stepsinterval$interval, stepsinterval$steps, type = "l", lwd = 2,
     xlab = "Interval",
     ylab = "Total Steps",
     main = "Average daily activity pattern")
```



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

```
summary(stepsinterval)
```

```
##     interval          steps
##  Min.   :   0.0   Min.   :  0.000
##  1st Qu.: 588.8   1st Qu.:  2.486
##  Median :1177.5   Median : 34.113
##  Mean   :1177.5   Mean   : 37.383
##  3rd Qu.:1766.2   3rd Qu.: 52.835
##  Max.   :2355.0   Max.   :206.170
```

```
stepsinterval[which.max(stepsinterval$steps),]
```

```
##     interval    steps
## 104      835 206.1698
```

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

```
missingvalues <- is.na(data)

totalmissingvalues <- sum(as.numeric(missingvalues))
```

Devise a strategy for filling in all of the missing values in the dataset. The strategy does not need to be sophisticated. For example, you could use the mean/median for that day, or the mean for that 5-minute interval, etc.

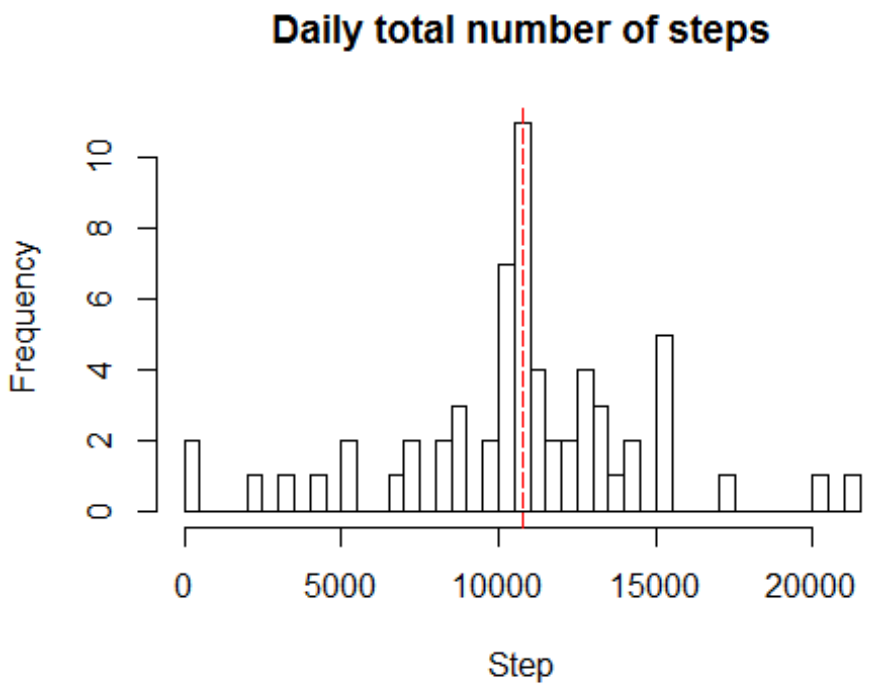
Median for 5-minute interval as the strategy to fill the missing values

```
nd <- data
for (i in stepsinterval) {
  nd[nd$interval == i & is.na(nd$steps), ]$steps <-
    stepsinterval$steps[stepsinterval$interval == i]
}

head(nd)

##       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

sum(is.na(nd))

## [1] 0
```

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

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?

```
daily_no_steps <- aggregate(steps ~ date, data = nd, sum, na.rm = TRUE)

hist(daily_no_steps$steps, main = "Daily total number of steps", xlab =
"Step", breaks = 60)
abline(v=median(daily_no_steps$steps), lty = 5, col = "red", main = "Median")
```




```
median(daily_no_steps$steps)
```

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

```
mean(daily_no_steps$steps)
```

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

```
median(sumsteps$steps)
```

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

```
mean(sumsteps$steps)
```

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

Are there differences in activity patterns between weekdays and weekends?

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.

```
nd <-nd %>%

  mutate(day=as.factor(ifelse(wday(date) %in% c(1,7),"weekend","weekday")))
```

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).

```
nd <- nd %>%

  group_by(day,interval) %>%

  summarise(meansteps=mean(steps))

ggplot(nd , aes(x = interval , y = meansteps)) + geom_line() + labs(title =
"weekday vs weekend", x = "Interval", y = "Number of Steps") +
facet_wrap(~`day` , ncol = 2, nrow=1)
```