Reproducible Research - Project 1 - PA1.RMD

Michael Guril

2023-07-26

#Libraries

```
library(lattice)
library(dplyr)
```

```
## Warning: Paket 'dplyr' wurde unter R Version 4.3.1 erstellt
```

```
##
## Attache Paket: 'dplyr'
```

```
## Die folgenden Objekte sind maskiert von 'package:stats':
##
##     filter, lag
```

```
## Die folgenden Objekte sind maskiert von 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
```

```
## Warning: Paket 'ggplot2' wurde unter R Version 4.3.1 erstellt
```

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

```
activity <- read.csv("D:/Data Science Foundations using R/5 Reproducible Research/Woche 2/Course Projec
```

#Datas

```
str(activity)
```

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

```
summary(activity)
```

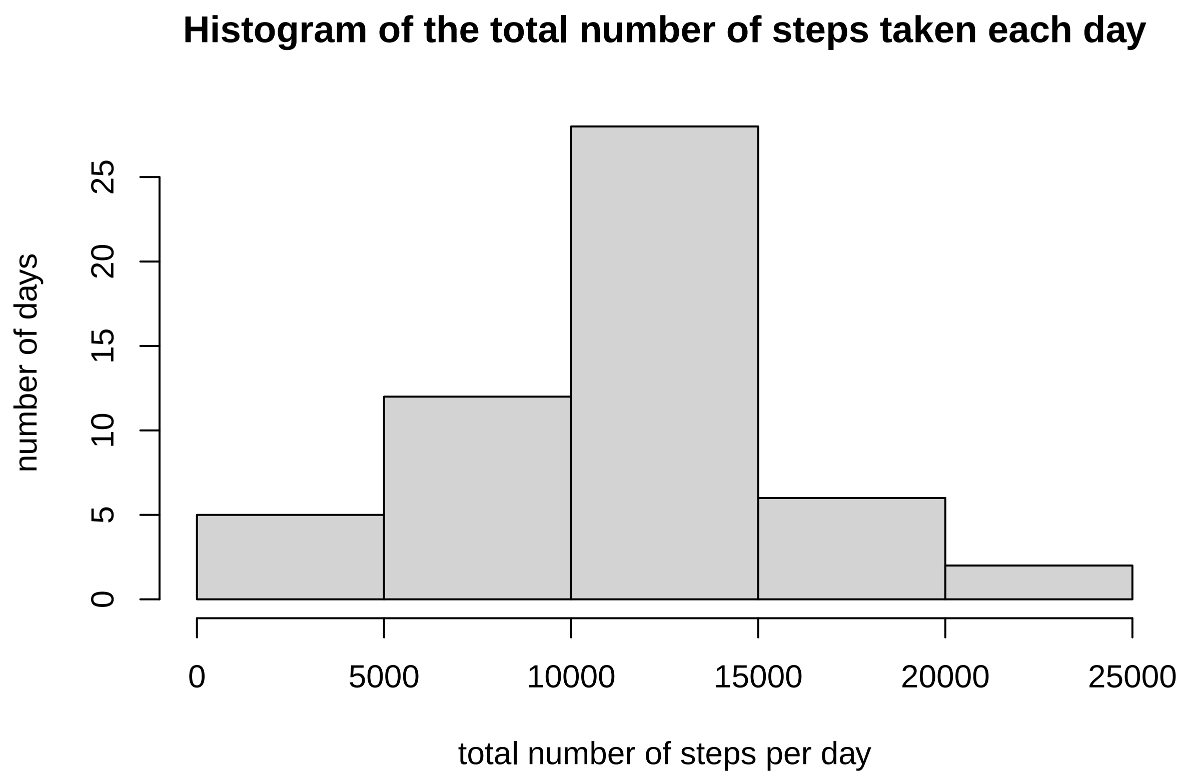
```
##      steps            date              interval
##  Min.   :  0.00   Length:17568       Min.   :   0.0
##  1st Qu.:  0.00   Class :character   1st Qu.: 588.8
##  Median :  0.00   Mode  :character   Median :1177.5
##  Mean   : 37.38                      Mean   :1177.5
##  3rd Qu.: 12.00                      3rd Qu.:1766.2
##  Max.   :806.00                      Max.   :2355.0
##  NA's   :2304
```

```
head(activity)
```

```
##   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
```

#Histogram of the total number of steps taken each day

```
totalStepsByDay<-aggregate(steps~date, activity, sum)
hist(totalStepsByDay$steps, xlab="total number of steps per day",
     ylab="number of days", main="Histogram of the total number of steps taken each day")
```



#Mean and median number of steps taken each day

```
mean_activity<-mean(totalStepsByDay$steps)
mean_activity
```

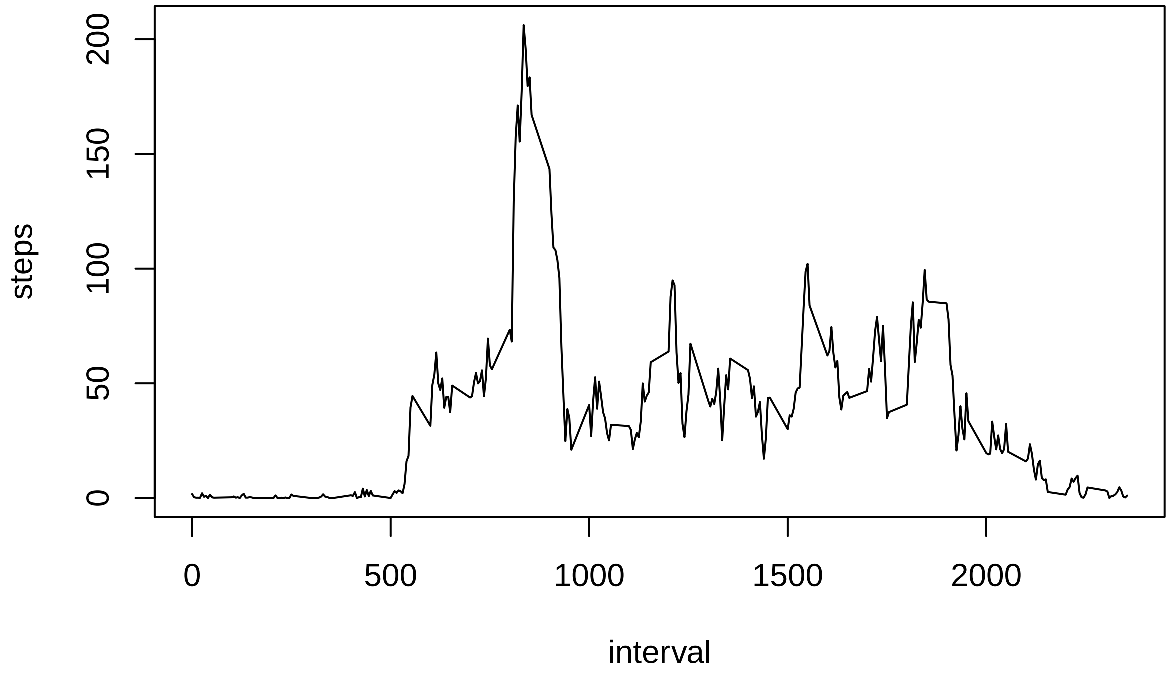
```
## [1] 10766.19
```

```
median_activity<-median(totalStepsByDay$steps)
median_activity
```

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

#Time series plot of the average number of steps taken

```
averageStepsbyInterval<-aggregate(steps~interval, activity, mean)
with(averageStepsbyInterval, plot(interval, steps, type = "l"))
```



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

```
averageStepsbyInterval[which.max(averageStepsbyInterval[,2]),1]
```

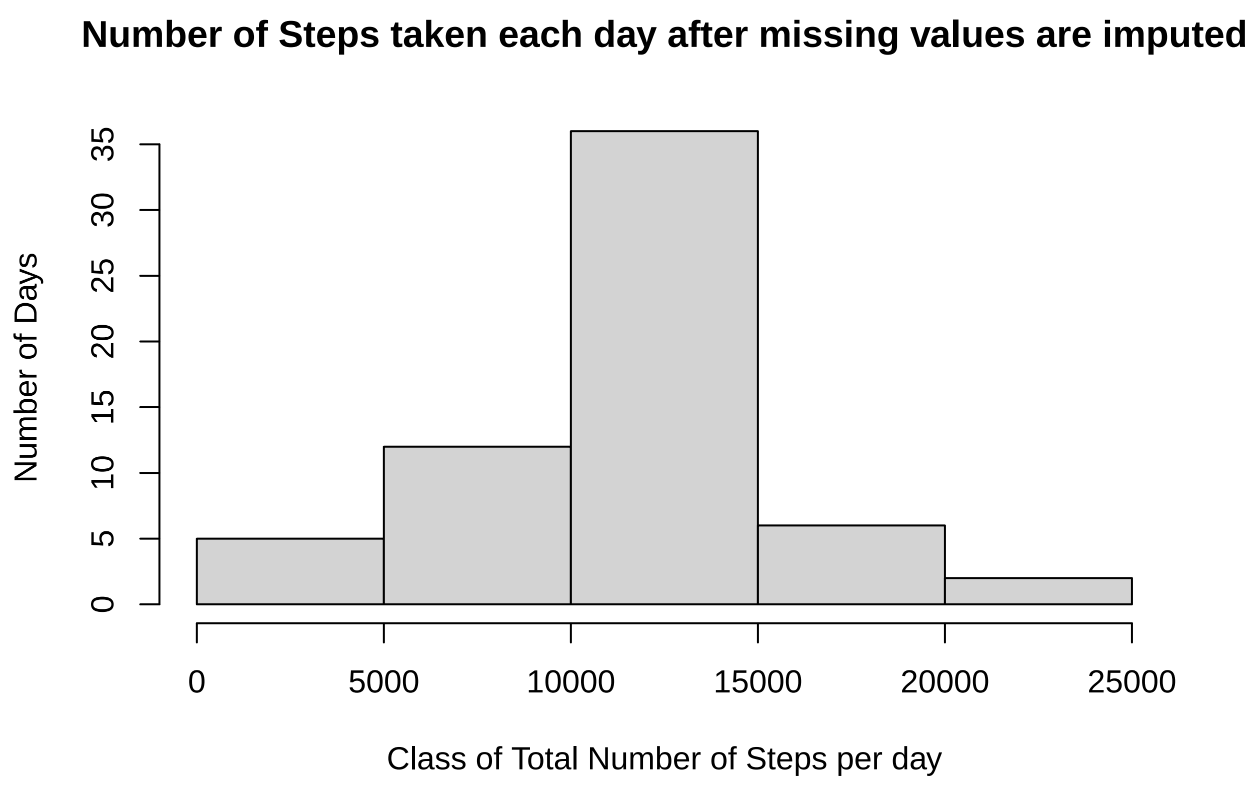
```
## [1] 835
```

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

```
missingIndex<-is.na(activity[,1])
activitym<-mean(averageStepsbyInterval$steps)
activityNEW<-activity
activityNEW[missingIndex,1]<-activitym
```

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

```
totalStepsByDayNEW<-aggregate(steps~date, activityNEW, sum)
hist(totalStepsByDayNEW$steps, xlab="Class of Total Number of Steps per day",
     ylab="Number of Days", main="Number of Steps taken each day after missing values are imputed")
```



#Mean and median number of steps taken each day (NEW)

```
totalStepsByDayNEW<-aggregate(steps~date, activityNEW, sum)
mean_activity_afterImput<-mean(totalStepsByDayNEW$steps)
mean_activity_afterImput
```

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

```
median_activity_afterImput<-median(totalStepsByDayNEW$steps)
median_activity_afterImput
```

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

#Panel plot average number of steps taken per 5-minute interval weekdays

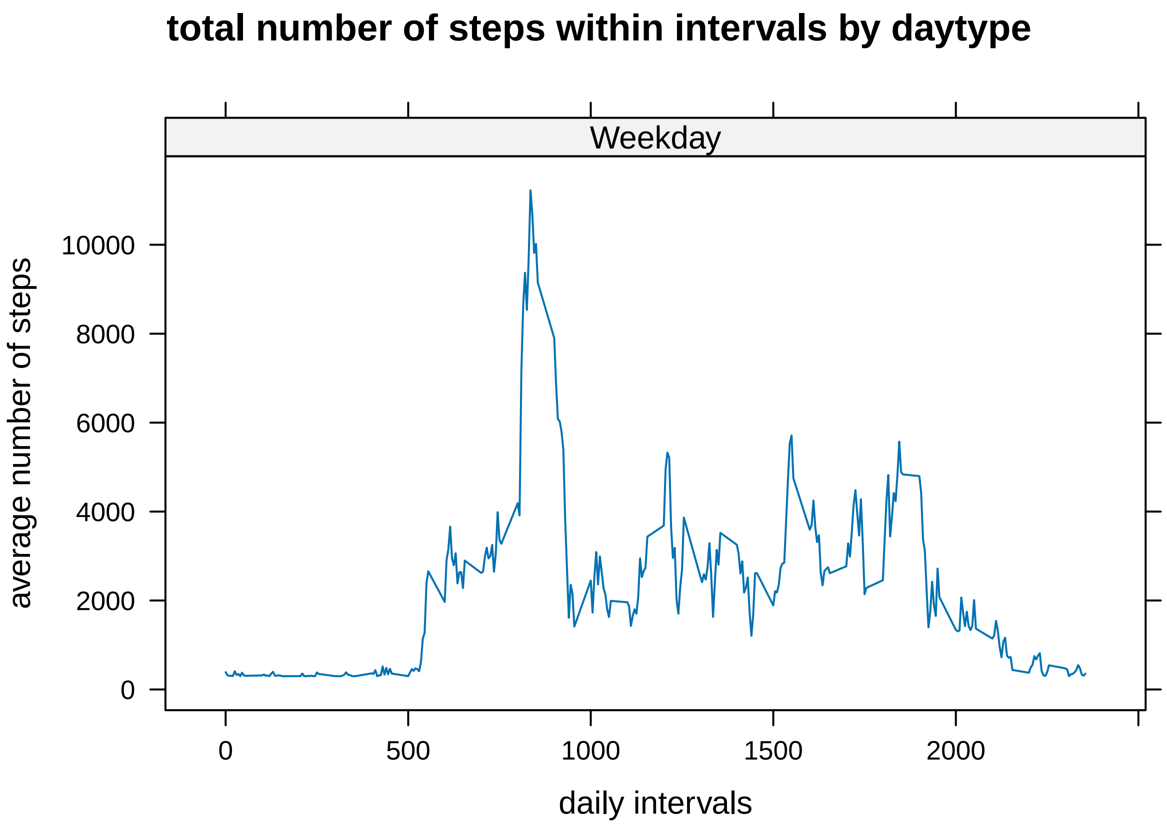
```
activityNEW$date<-as.Date(activityNEW$date)

activityfinal<-activityNEW %>%
    mutate(dayType= ifelse(weekdays(activityNEW$date)=="Saturday" | weekdays(activityNEW$date)=="Sunday

averageStepByDayTypeAndInterval<-activityfinal %>%
    group_by(dayType, interval) %>%
    summarize(averageStepByDay=sum(steps))
```

```
## `summarise()` has grouped output by 'dayType'. You can override using the
## `.groups` argument.
```

```
with(averageStepByDayTypeAndInterval,
    xyplot(averageStepByDay ~ interval | dayType, type = "l",
       main = "total number of steps within intervals by daytype",
       xlab = "daily intervals",
       ylab = "average number of steps"))
```


total number of steps within intervals by daytype
Weekday
average number of steps
daily intervals