Reproducible Research: Activity Monitoring Data

Leonardo Simioni

2022-09-21

Loading and preprocessing the data

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

Loading the 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
```

```
act_nona <- na.omit(act)
head(act_nona)
```

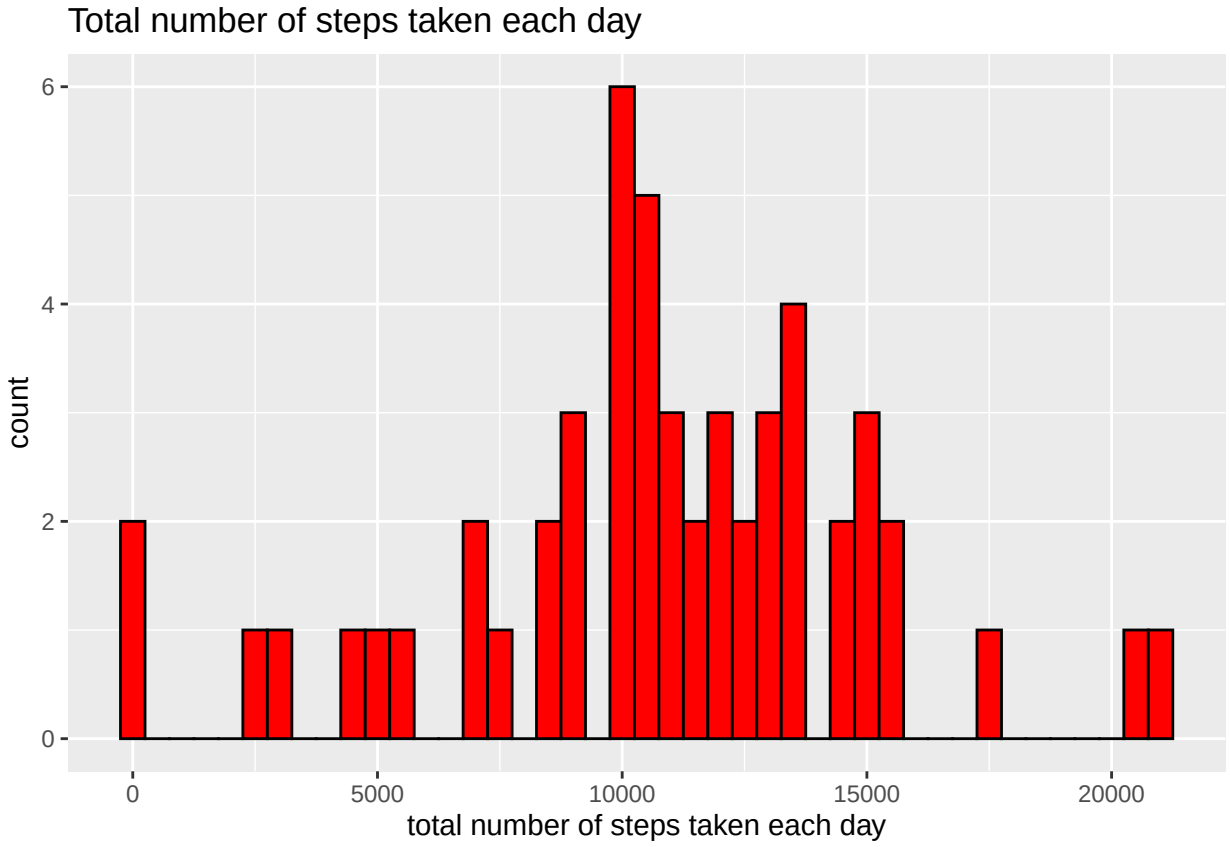
Processing the data into a suitable format for analysis

```
##     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
```

The mean total number of steps taken per day

```
act_date <- aggregate(steps~date, act_nona, sum)
library(ggplot2)
ggplot(data=act_date, aes(steps)) + geom_histogram(binwidth=500, fill="red", col="black") + labs(title=
```

Calculating and making a histogram of the total number of steps taken each day




```
summary(act_date$steps)
```

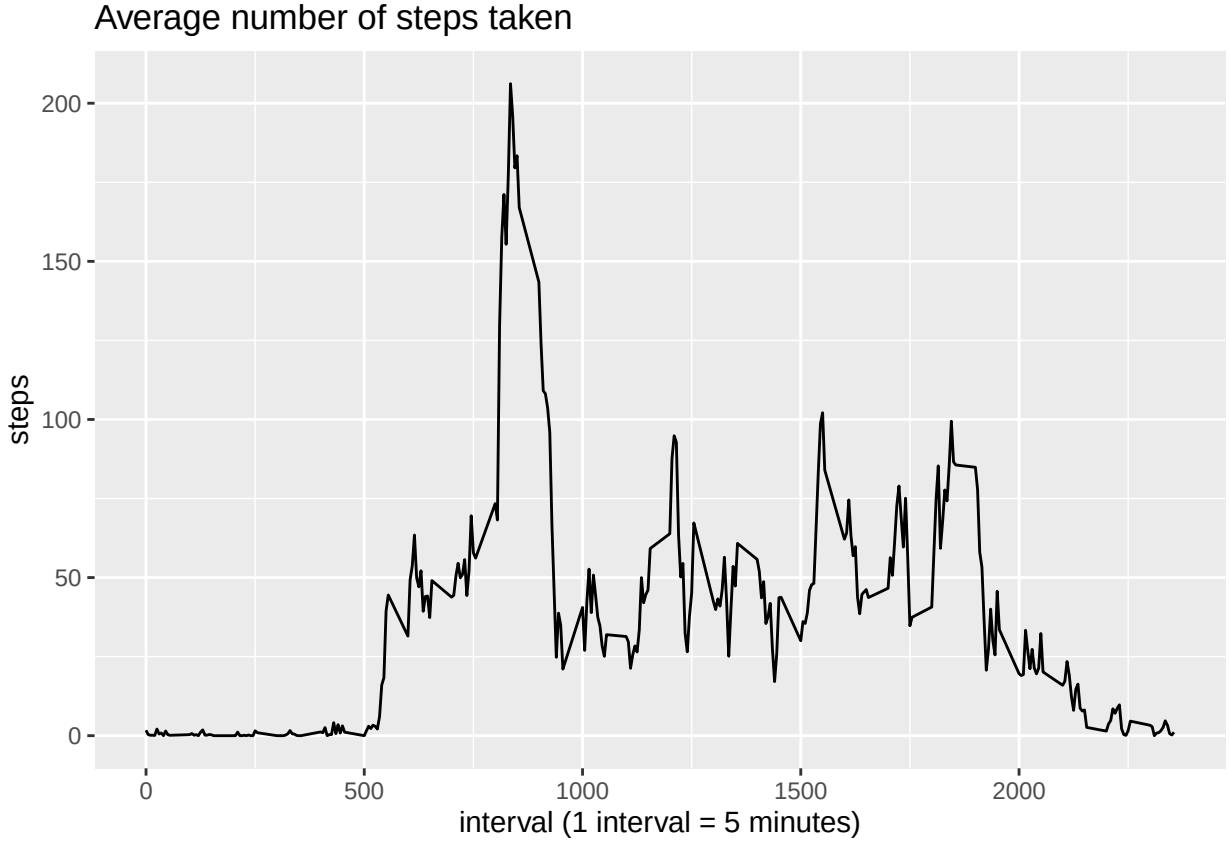
Calculating the mean and median of the total number of steps taken per day

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

The average daily activity pattern

```
act_int <- aggregate(steps~interval, act_nona, mean)
ggplot(data=act_int, aes(x=interval,y=steps)) + geom_line() + labs(title="Average number of steps taken
```

Time series plot of the 5-minute interval and the average number of steps taken




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

5-minute interval containing the maximum number of steps

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

Imputing missing values

```
colSums(is.na(act))
```

Calculating the total number of missing values in the dataset

```
##    steps     date interval
##     2304        0        0
```

```
act_fill <- act
act_fill$steps <- ifelse(is.na(act_fill$steps) == TRUE, act_int$steps[act_int$interval %in% act_fill$int
head(act_fill)
```

Filling all of the missing values in the dataset

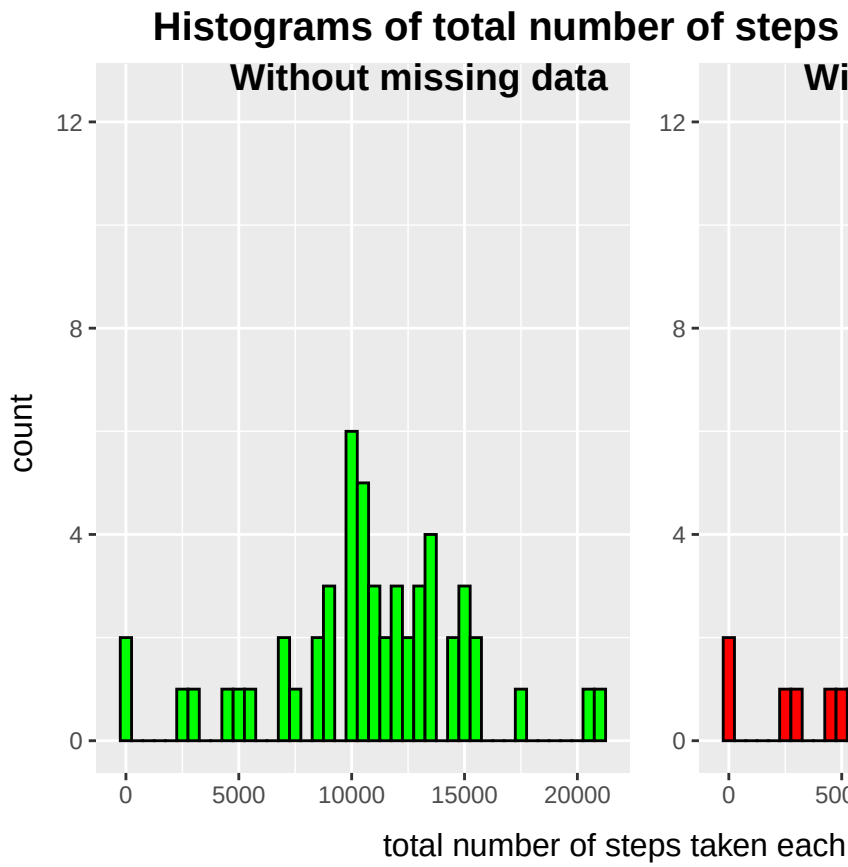
```
##       steps       date interval
## 1 1.7169811 2012-10-01        0
## 2 0.3396226 2012-10-01        5
## 3 0.1320755 2012-10-01       10
## 4 0.1509434 2012-10-01       15
## 5 0.0754717 2012-10-01       20
## 6 2.0943396 2012-10-01       25
```

```
act_fill_date <- aggregate(steps~date, act_fill, sum)

p1 <- ggplot(data=act_date, aes(steps)) + geom_histogram(binwidth=500, fill="green", col="black") + labs
p2 <- ggplot(data=act_fill_date, aes(steps)) + geom_histogram(binwidth=500, fill="red", col="black") + l

library(ggpubr)
p1_p2 <- ggarrange(p1, p2, labels = c("Without missing data", "With missing data"), ncol = 2, nrow = 1,
annotate_figure(p1_p2, top = text_grob("Histograms of total number of steps taken each day", face="bold
```

Making a histogram of the total number of steps taken each day and calculating the mean and



median total number of steps taken per day

Differences in activity patterns between weekdays and weekends

```
act_week <- act_fill
Sys.setlocale("LC_TIME", "English")
```

Creating a vector variable indicating whether a given date is a weekday or weekend day

```
## [1] "English_United States.1252"
```

```
act_week$day <- as.factor(weekdays(as.Date(act_week$date)))
act_week$WD_WE <- as.factor(ifelse(act_week$day == "Saturday" | act_week$day == "Sunday", "weekend", "w
head(act_week)
```

```
##       steps       date interval    day   WD_WE
## 1 1.7169811 2012-10-01        0 Monday weekday
## 2 0.3396226 2012-10-01        5 Monday weekday
## 3 0.1320755 2012-10-01       10 Monday weekday
## 4 0.1509434 2012-10-01       15 Monday weekday
## 5 0.0754717 2012-10-01       20 Monday weekday
## 6 2.0943396 2012-10-01       25 Monday weekday
```

```
act_week_date <- aggregate(steps ~ interval + WD_WE, act_week, mean)
head(act_week_date)
```

Making a plot containing a time series plot of the 5-minute interval and the average number of steps taken

```
##   interval   WD_WE      steps
## 1        0 weekday 2.25115304
## 2        5 weekday 0.44528302
## 3       10 weekday 0.17316562
## 4       15 weekday 0.19790356
## 5       20 weekday 0.09895178
## 6       25 weekday 1.59035639
```

```
ggplot(data=act_week_date, aes(colour=WD_WE, x=interval, y=steps)) + geom_line()+labs(x="interval (1 in
```