Reproducible Research: Peer Assessment 1

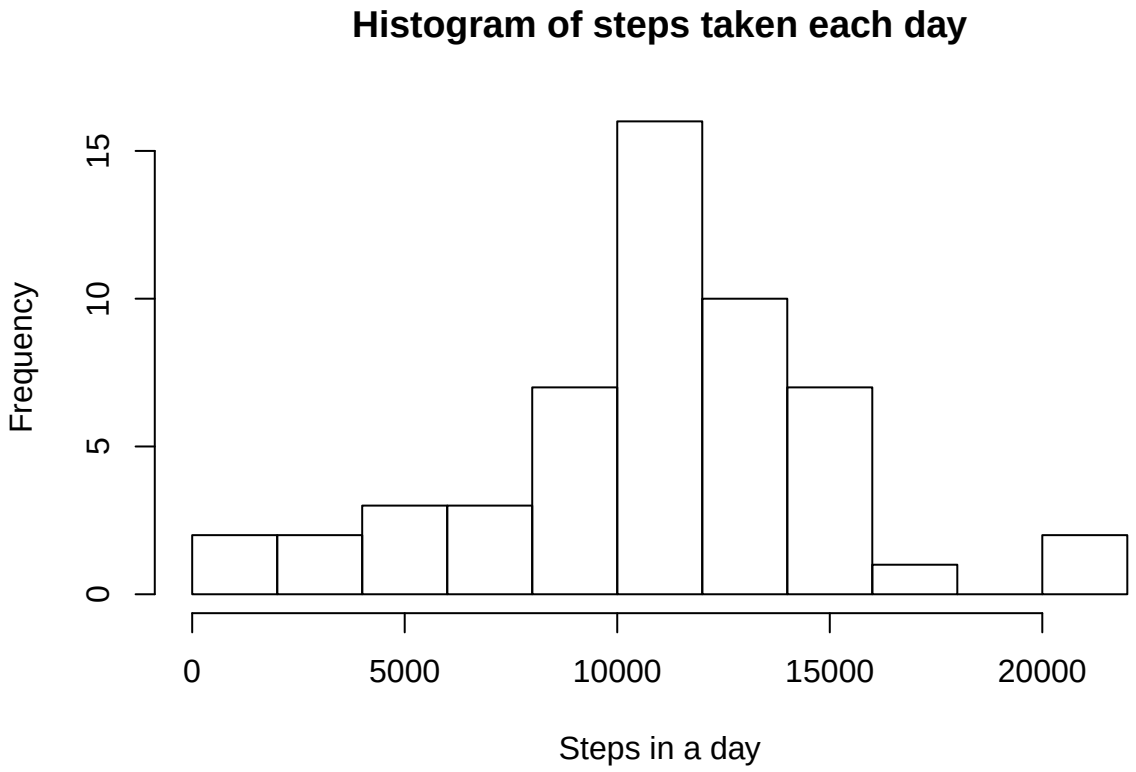
Loading and preprocessing the data

To manipulate data easier I will be using tidyr and dplyr libraries. First I load activity data from a zip file into a data.frame date. In the next step I create two new variables that I'll use to answer later questions, one holds aggregate of number of steps by day (byDay) and the second one average number of steps by the interval of the day.

```
#Using the tidyr library to manipulate data
library(tidyr)
library(dplyr)

#One line unzip, read csv and make a data frame
data <- data.frame(read.csv(unzip('activity.zip', "activity.csv")))

#Aggregate number of steps by date (used for histogram 1)
byDay <- aggregate(steps ~ date, data, sum)

#A table with Average number of steps by interval
byInterval <- aggregate(steps ~ interval, data, sum)
byInterval$steps <- byInterval$steps/length(unique(data$date))

#calendar, num of weekdays, num weekends, to use for the last plot
calendar <- unique(data$date)
weekends <- 0
weekdays <- 0

for(i in 0:length(calendar)){
  dow <- format(strptime(calendar[i], "%Y-%M-%d"), "%u")
  ifelse(dow>5, weekends <- weekends+1, weekdays <- weekdays+1)
}
```

What is mean total number of steps taken per day?

```
hist(byDay$steps, breaks=10, main="Histogram of steps taken each day", xlab="Steps in a day")
```

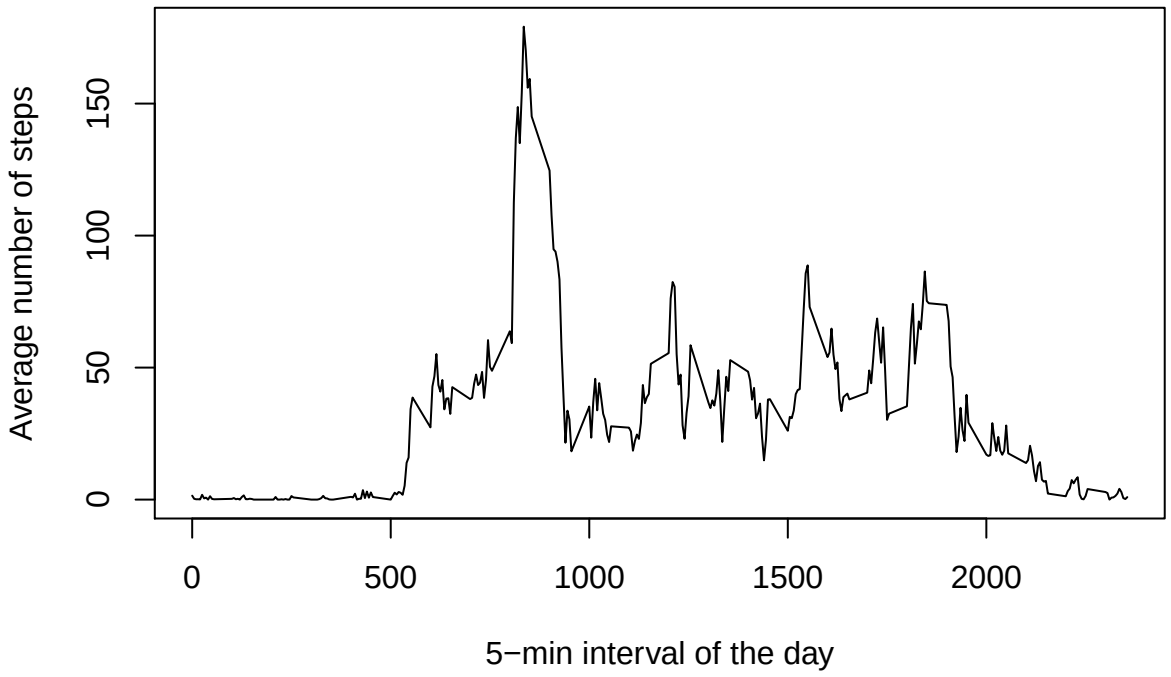



```
meanByDaySteps <- mean(byDay$steps)
medianByDaySteps <- median(byDay$steps)
```

First I draw the total number of steps taken each day and then calculate the mean (10766.19) and median (10765).

What is the average daily activity pattern?

```
plot(byInterval$interval, byInterval$steps, type='l', ylab="Average number of steps", xlab="5-min interv
```




```
maxObservation <- which.max( byInterval$steps )
maxInterval <- byInterval$interval[maxObservation]
maxInterval
```

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

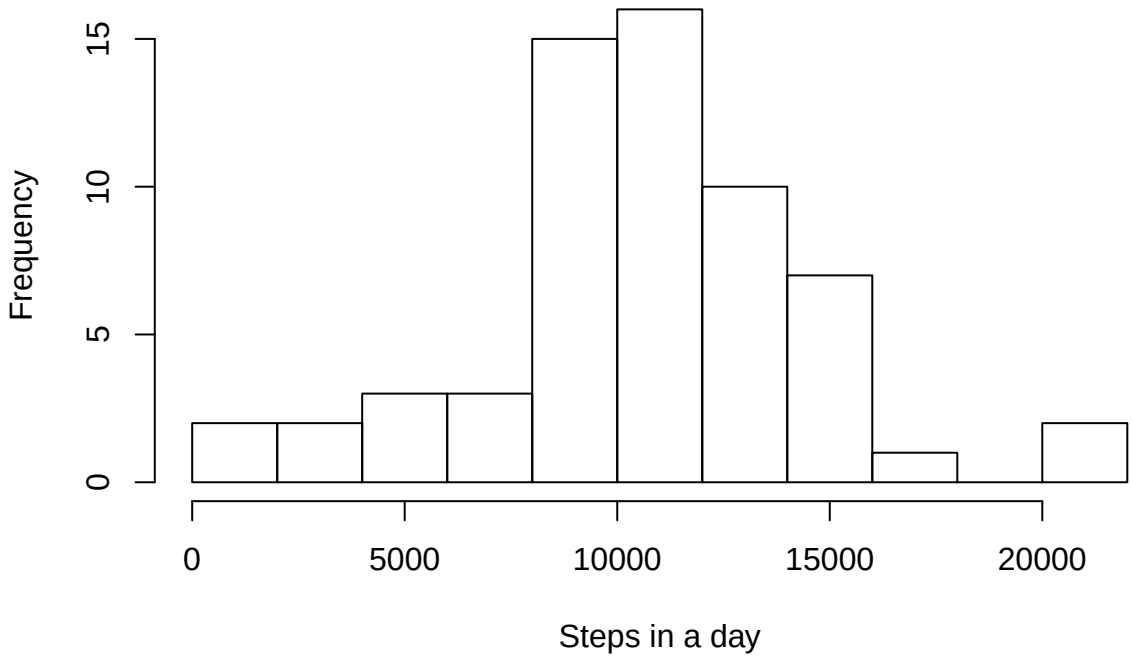
To answer this question I first draw a time series plot, of the 5-minute interval and then calculate which interval contains the maximum number of steps (interval 835, observation 104).

Imputing missing values

```
#NA values
NAValues <- sum(is.na(data))
NAValues
```

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

```
dataNoNA <- data
for(i in 1:nrow(data)) {
  if(is.na(data[i,]$steps)){
    avgSteps = subset(byInterval, interval==data[i,]$interval)$steps
    dataNoNA[i,]$steps = avgSteps
  }
}

byDayNoNA <- aggregate(steps ~ date, dataNoNA, sum)

hist(byDayNoNA$steps, breaks=10, main="Histogram of steps taken each day (missing values replaced)", xla
```




```
meanByDayNoNA <- mean(byDayNoNA$steps)
medianByDayNoNA <- median(byDayNoNA$steps)

meanByDayNoNA
```

```
## [1] 10581.01
```

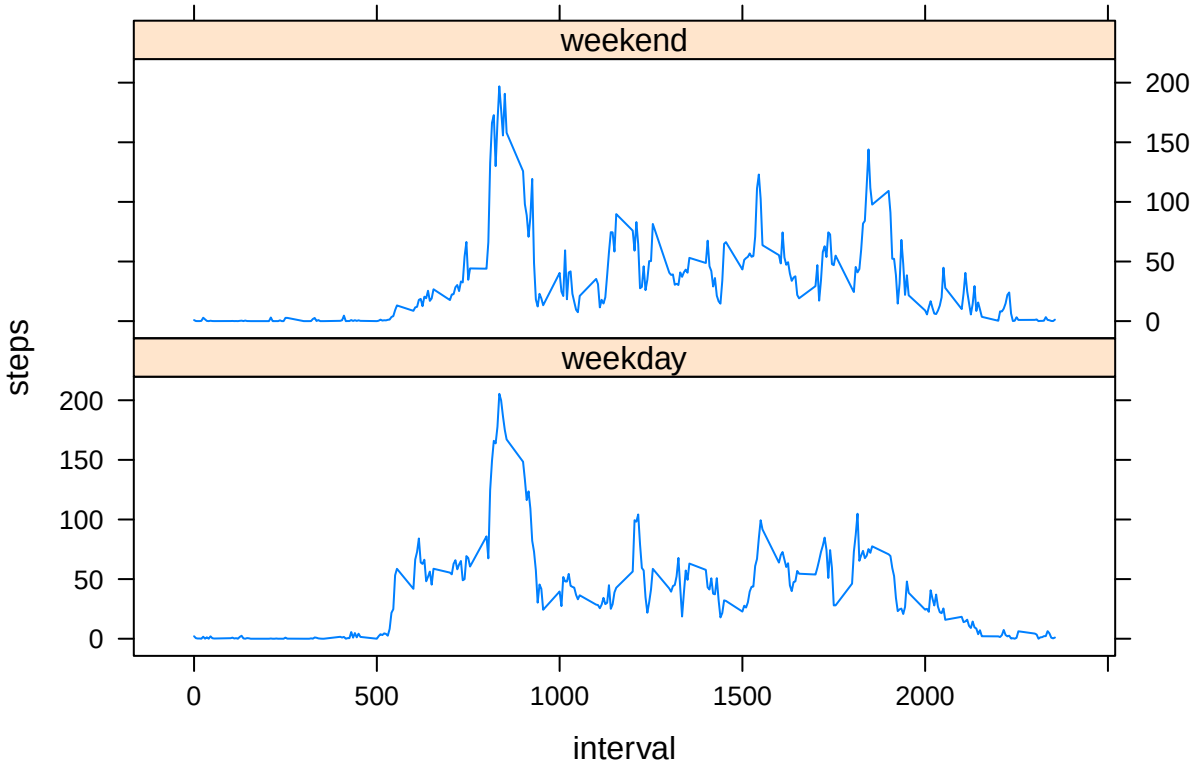
```
medianByDayNoNA
```

```
## [1] 10395
```

In this step I replaced the missing step values with average step value for that interval. The average step values are stored in byInterval variable. There were 2304 missing values that were replaced. The new mean was 10581.01 and median 10395.

Are there differences in activity patterns between weekdays and weekends?

```
dataNoNA <- mutate(dataNoNA, day = format(strptime(date, "%Y-%M-%d"), "%u"))
dataNoNA <- mutate(dataNoNA, day = ifelse(day>5, "weekend", "weekday"))
dataNoNA <- aggregate(steps ~ interval + day, dataNoNA, sum)


dataNoNA <- mutate(dataNoNA, steps = ifelse(day=="weekend", steps/weekends, steps/weekdays))

library(lattice)
xyplot(steps ~ interval| factor(day), data=dataNoNA, type='l', layout=c(1, 2))
```



In this last part I separated the observations to weekends and weekdays and ploted them out for comparison.