Reproducible Research: Peer Assessment 1

Nisheeth Joshi

5 January 2017

Loading and preprocessing the data

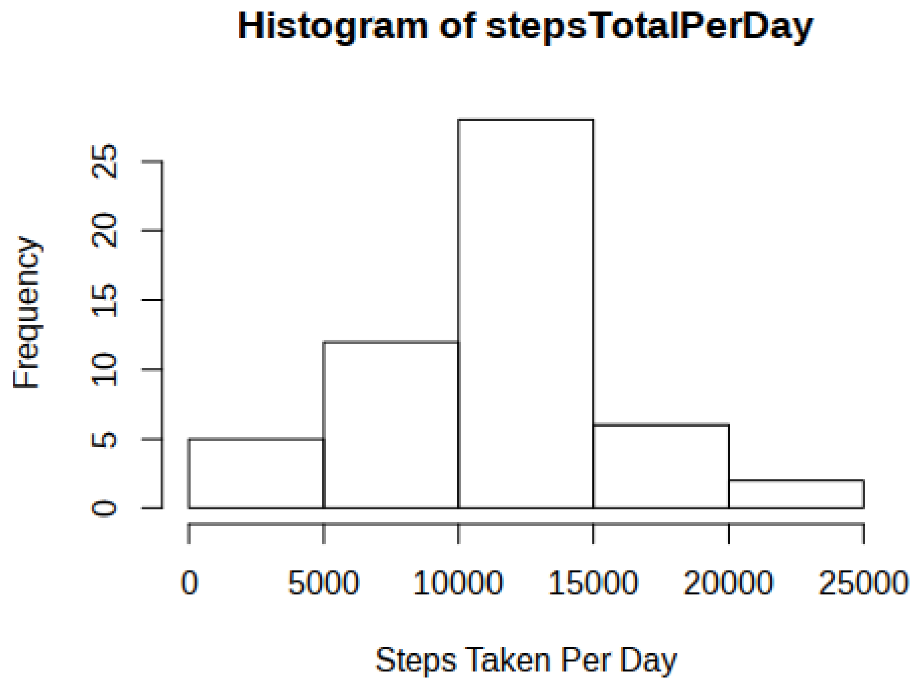
This section show how dataset is read.

```
activity <- read.csv("activity.csv")
View(activity)
```

What is mean total number of steps taken per day?

This section plots a Histogram of the total number of steps taken each day. Moreover it also calculates mean and median number of steps taken each day.

```
stepsTotalPerDay <- tapply(activity$steps, activity$date, sum)
hist(stepsTotalPerDay, xlab = "Steps Taken Per Day", ylab = "Frequency")
```



```
mean(stepsTotalPerDay, na.rm = TRUE)

## [1] 10766.19
```

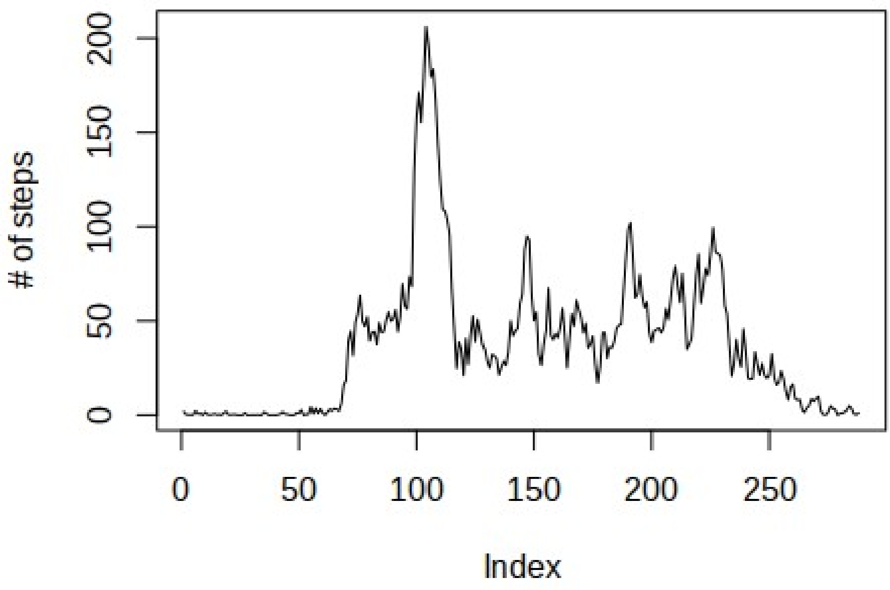
```
median(stepsTotalPerDay, na.rm = TRUE)

## [1] 10765
```

What is the average daily activity pattern?

This section plots a time series of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis).

```
stepsMeanPerInterval <- tapply(activity$steps, activity$interval, mean, na.rm = T)
plot(stepsMeanPerInterval, type = "l", ylab = "# of steps")
```



```
stepsMeanPerInterval <- tapply(activity$steps, activity$interval, mean, na.rm = T)
seq(along = stepsMeanPerInterval)[stepsMeanPerInterval ==
max(stepsMeanPerInterval)]

## [1] 104
```

Imputing missing values

This section calcuates the total number of NAs and plots a Histogram for Frequecy against number of steps taken per data. While doing this NAs have been ignored.

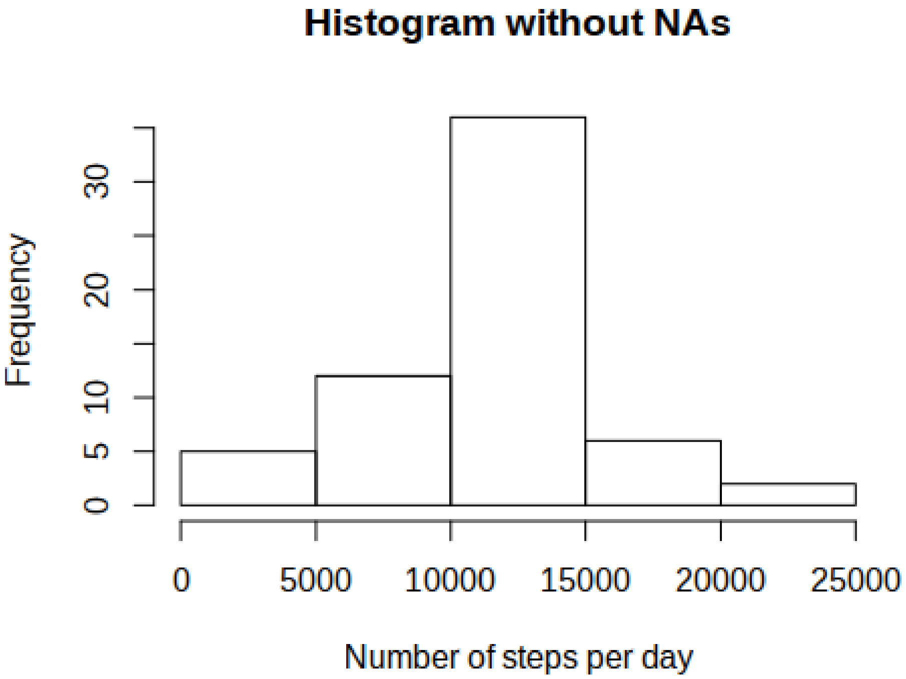
```
sum(as.numeric(is.na(activity$steps)))

## [1] 2304
```

```
temp_stepsMeanPerInterval <- as.vector(stepsMeanPerInterval)
temp_stepsMeanPerInterval <- rep(temp_stepsMeanPerInterval, 61)
temp_stepsMeanPerInterval[!is.na(activity$steps)] = 1
temp_dataTest <- as.vector(activity$steps)
temp_dataTest[is.na(temp_dataTest)] = 1

data_NoNA <- activity

data_NoNA$steps <- temp_stepsMeanPerInterval * temp_dataTest
stepsTotalPerDay_NoNA <- tapply(data_NoNA$steps, data_NoNA$date, sum)
hist(stepsTotalPerDay_NoNA,  main = "Histogram without NAs", xlab = "Number of steps
per day", ylab = "Frequency")
```



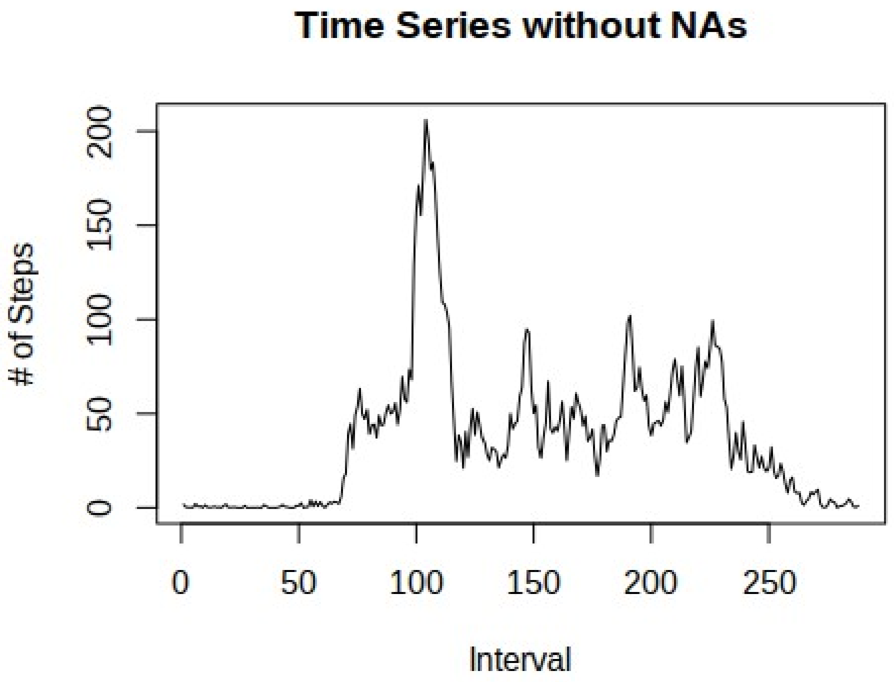
```
stepsMeanPerInterval_NoNA <- tapply(data_NoNA$steps, data_NoNA$interval, mean)
mean(stepsTotalPerDay_NoNA)

## [1] 10766.19

median(stepsTotalPerDay_NoNA)

## [1] 10766.19

plot(stepsMeanPerInterval_NoNA, type = "l", xlab = "Interval", ylab = "# of Steps", main =
"Time Series without NAs")
```



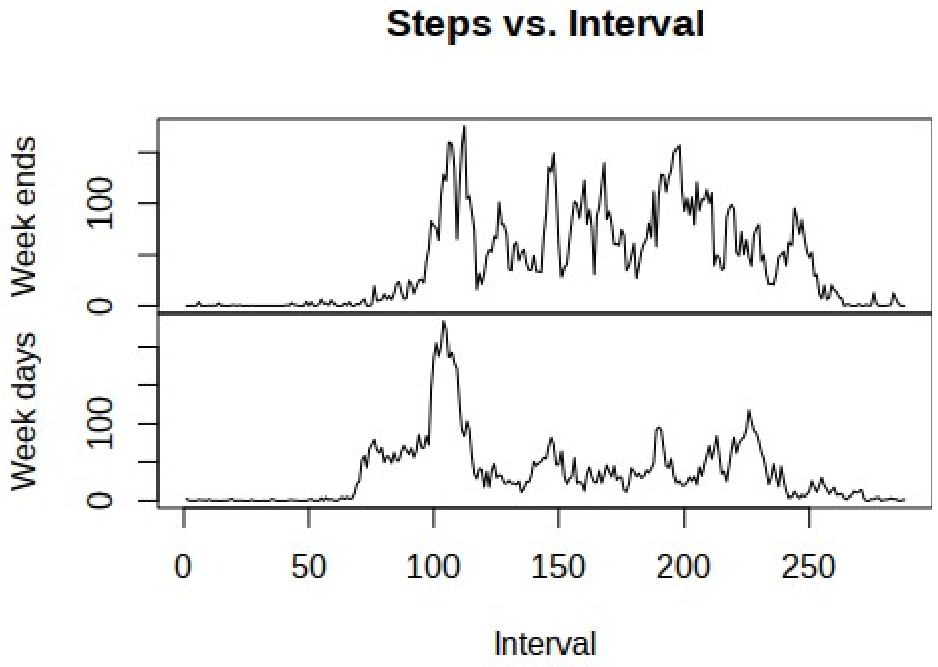
Are there differences in activity patterns between weekdays and weekends?

This section fators the data for two saperate type of days i.e. weekdays and weekends and plots the time series graph for both.

```
tmpLT <- as.POSIXlt(activity$date, format = "%Y-%m-%d")
tmpWeekDays <- tmpLT$wday
tmpWeekDays[tmpWeekDays == 0] = 0
tmpWeekDays[tmpWeekDays == 6] = 0
tmpWeekDays[tmpWeekDays != 0] = 1
tmpWeekDaysFactor <- factor(tmpWeekDays, levels = c(0, 1))
activity$WD <- tmpWeekDaysFactor
stepsMeanPerWeekday <- tapply(activity$steps, list(activity$interval, activity$WD),
mean, na.rm = TRUE)
par(mfrow = c(2, 1))

with(activity, {
  par(mai = c(0, 1, 1, 0))
  plot(stepsMeanPerWeekday[, 1], type = "l", main = ("Steps vs. Interval"),
       xaxt = "n", ylab = "Week ends")
  title = ("# of Steps v.s. Interval")
  par(mai = c(1, 1, 0, 0))
  plot(stepsMeanPerWeekday[, 2], type = "l", xlab = "Interval", ylab = "Week days")

})
```


Steps vs. Interval
Week ends
0
100
Week days
0
100
0
50
100
150
200
250
Interval