Reproducible Research: Peer Assessment 1

Introducation

This assignment will be described in multiple parts. You will need to write a report that answers the questions detailed below. Ultimately, you will need to complete the entire assignment in a single R markdown document that can be processed by knitr and be transformed into an HTML file.

Throughout your report make sure you always include the code that you used to generate the output you present. When writing code chunks in the R markdown document, always use echo = TRUE so that someone else will be able to read the code. This assignment will be evaluated via peer assessment so it is essential that your peer evaluators be able to review the code for your analysis.

For the plotting aspects of this assignment, feel free to use any plotting system in R (i.e., base, lattice, ggplot2)

Fork/clone the GitHub repository created for this assignment. You will submit this assignment by pushing your completed files into your forked repository on GitHub. The assignment submission will consist of the URL to your GitHub repository and the SHA-1 commit ID for your repository state.

NOTE: The GitHub repository also contains the dataset for the assignment so you do not have to download the data separately.

Loading and preprocessing the data

Loading the activity data as data first.

```r
setwd("/Users/yxma/Documents/coursera/reproducible_research/RepData_PeerAssessment1")
data <- read.csv("activity.csv", header= TRUE, sep = ",")
```

Then convert the varibale *date* from factor to character.

```r
data$date <- as.Date(data$date, "%Y-%m-%d")
str(data)
```

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

Check the data dimensions

```r
dim(data)
```

```
## [1] 17568     3
```

```r
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
```

Expolaring the data

What is mean total number of steps taken per day?

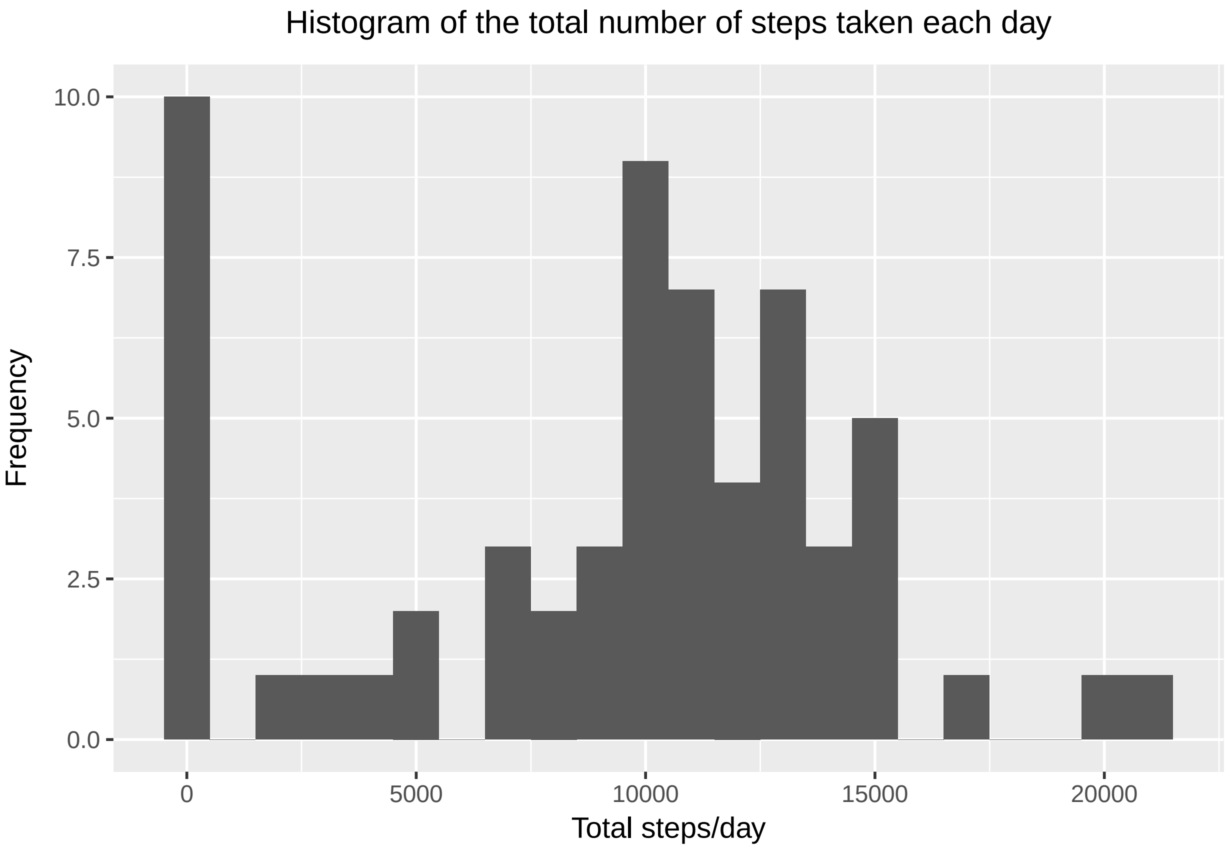
```
library(dplyr)
```

```
##
## 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
```

```
AvgNum <- data %>% group_by(date) %>%
    summarize(total.steps = sum(steps, na.rm = T),
              mean.steps = mean(steps, na.rm = T))
```

Visulazing the results using histogram

```
library(ggplot2)
p <- ggplot(AvgNum, aes(x=total.steps))
p + geom_histogram(binwidth = 1000)+
    theme(plot.title = element_text(size = 12,hjust = 0.5),
          axis.title.x = element_text(size = 10.5),
          axis.title.y = element_text(size = 10.5)) +
    labs( x = "Total steps/day",  y = expression("Frequency"),
          title = expression("Histogram of the total number of steps taken each day"))
```



Let's get a summary of the data, which will give the mean and median of the total steps and average steps.

```
summary(AvgNum$total.steps)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##       0    6778   10400    9354   12810   21190
```

```
summary(AvgNum$mean.steps)
```

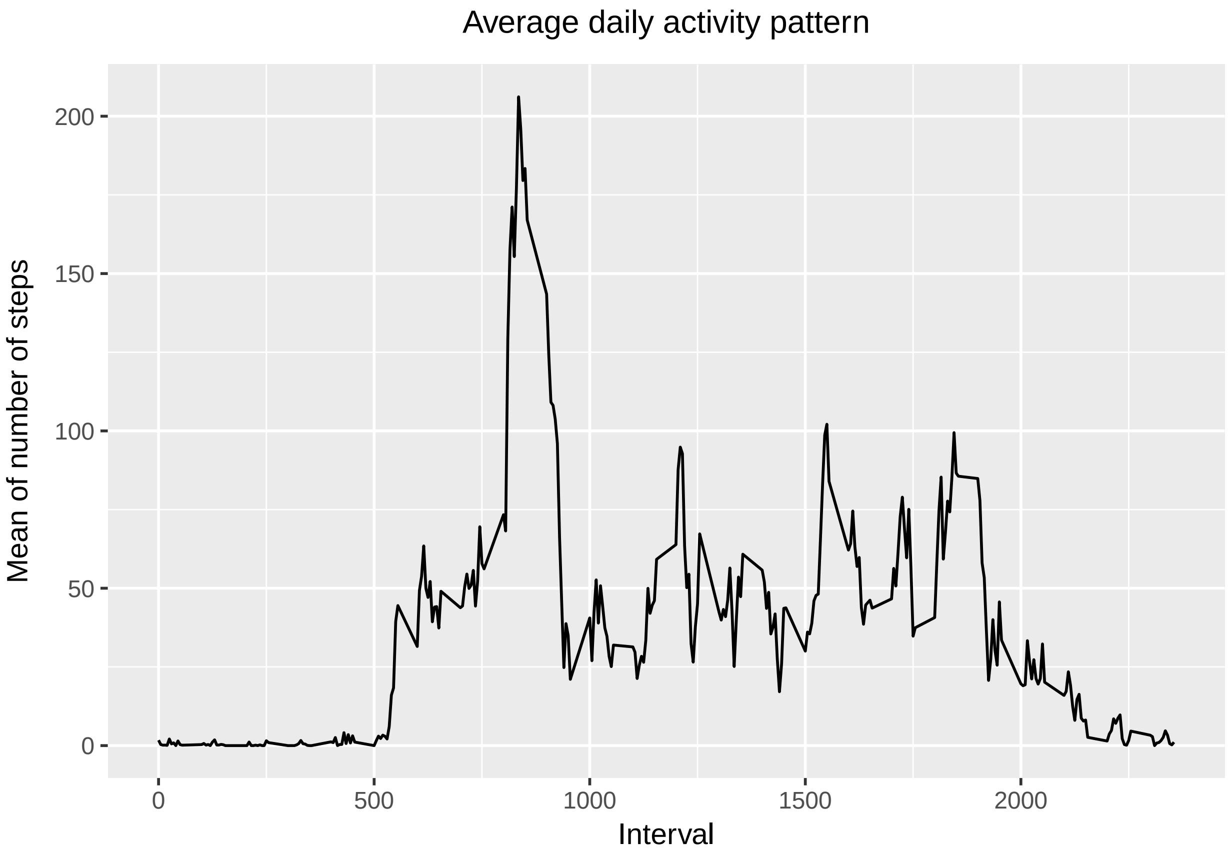
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##  0.1424 30.7000 37.3800 37.3800 46.1600 73.5900       8
```

As we can see the mean and median of total number of steps are 9354 and 10400 steps.

What is the average daily activity pattern?

First, I will make a time series plot of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis)

```
AvgInterval <- data %>% group_by(interval) %>%
    summarize(mean.steps = mean(steps, na.rm = T))
p <- ggplot(AvgInterval, aes(x= interval, y = mean.steps))
p + geom_line() + theme(plot.title = element_text(size = 12,hjust = 0.5),
          axis.title.x = element_text(size = 10.5),
          axis.title.y = element_text(size = 10.5)) +
    labs( x = "Interval",  y = expression("Mean of number of steps"),
          title = expression("Average daily activity pattern"))
```



As it is shown in the above, the largest amount of steps occures between time interval of 750 and 1000. The maximum average number of steps is: 206 and occurs in time interval 835.

Imputing missing values

```
mean(is.na(data$steps))
```

```
## [1] 0.1311475
```

```
sum(is.na(data$steps))
```

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

First I will create a new data set replcaing NAs with estimated values.

```
newdata <- data
```

For estimated values, I will use the average steps pre interval to fill the NAs.

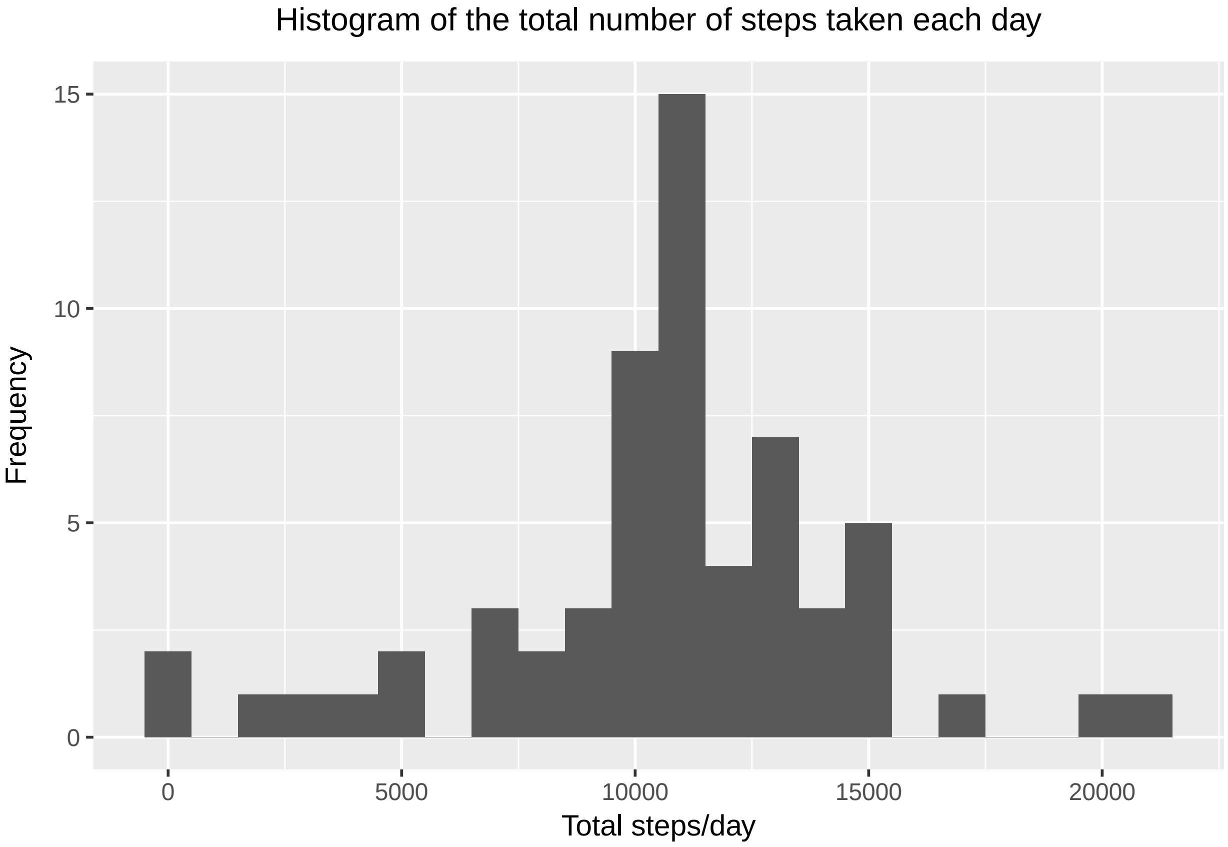
```
for (i in 1:nrow(newdata)) {
      if (is.na(newdata$steps[i])) {
            index <- newdata$interval[i]
            value <- subset(AvgInterval, interval==index)
            newdata$steps[i] <- value$mean.steps
      }
}
```

Then calculate the daily total steps.

```
newAvg <- newdata %>% group_by(date) %>%
      summarize(total.steps = sum(steps, na.rm = T))
```

And draw the histogram of the new data

```
p <- ggplot(newAvg, aes(x=total.steps))
p + geom_histogram(binwidth = 1000)+
    theme(plot.title = element_text(size = 12,hjust = 0.5),
          axis.title.x = element_text(size = 10.5),
          axis.title.y = element_text(size = 10.5)) +
    labs( x = "Total steps/day",  y = expression("Frequency"),
          title = expression("Histogram of the total number of steps taken each day"))
```



Compare the mean, median and standard deviations of the data and the new data

```
summary (AvgNum$total.steps)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##       0    6778   10400    9354   12810   21190
```

```
sd(AvgNum$total.steps, na.rm=T)
```

```
## [1] 5405.895
```

```
summary (newAvg$total.steps)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##      41    9819   10770   10770   12810   21190
```

```
sd(newAvg$total.steps, na.rm=T)
```

```
## [1] 3974.391
```

The standerd deviation of new data is smaller than the orginal data. And the mean and median after filling missing data are larger than orginal data.

Are there differences in activity patterns between weekdays and weekends?

Create two subsets, one containing the weekdays and one containing the weekends data:

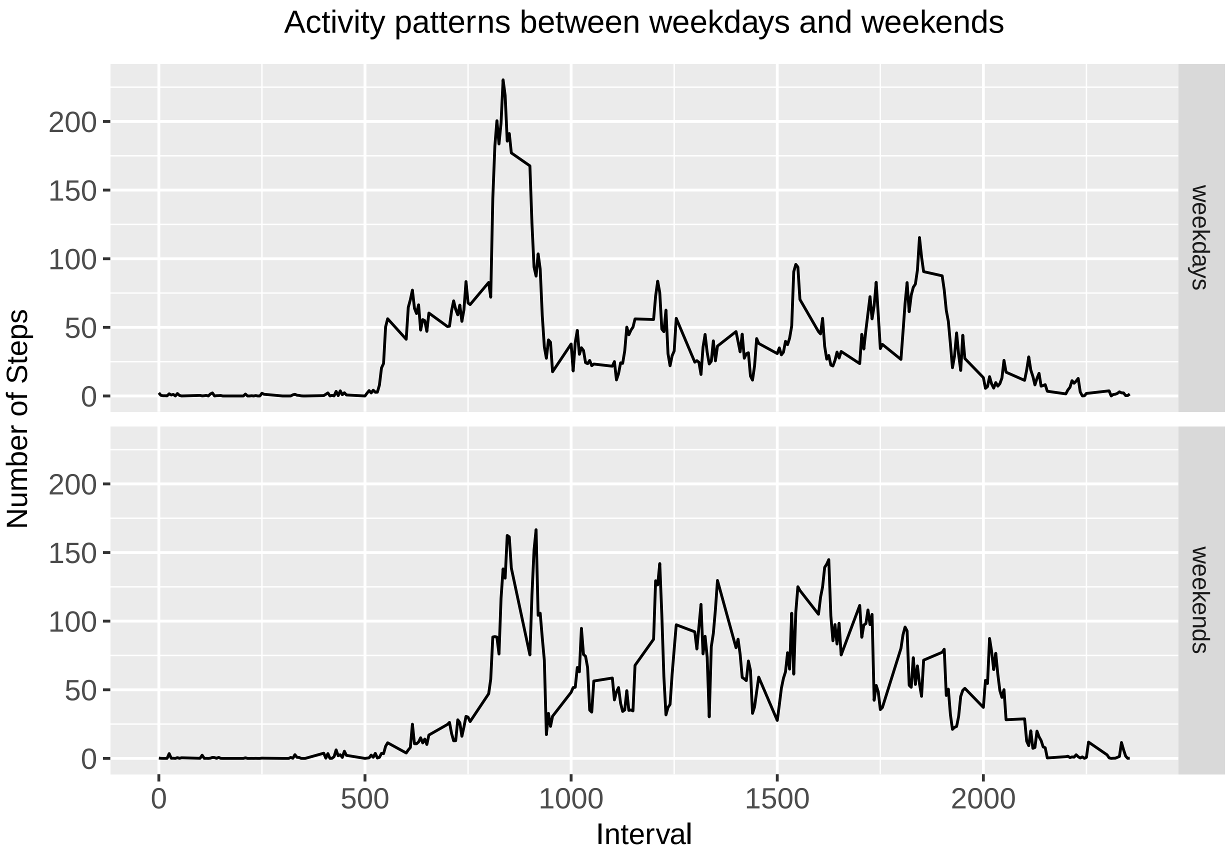
```
newdata$day <- ifelse(weekdays(newdata$date) %in% c("Saturday", "Sunday"), "weekend", "weekday")
weekends <- filter(newdata, day == "weekend")
weekdays <- filter(newdata, day == "weekday")
```

Then, we group by the intervals and calculate the mean number of steps for each time interval in two subsets

```
weekdays <- weekdays%>%
      group_by(interval) %>%
      summarize(mean.steps = mean(steps))
weekdays$day <- "weekdays"

weekends <- weekends %>%
      group_by(interval) %>%
      summarize(mean.steps = mean(steps))
weekends$day <- "weekends"

newInterval <- rbind(weekdays, weekends)
```

Plot the weekdays and weekends interval data.

```
p <- ggplot (newInterval, aes (interval, mean.steps))
p + geom_line() + facet_grid (day~.) + theme(plot.title = element_text(size = 12,hjust = 0.5),
                                             axis.text = element_text(size = 10.5))  +
    labs(x = "Interval", y = "Number of Steps" , title = expression("Activity patterns between weekdays
```



What is interesting about this plot is that it very much sums up the activity that any normal person would undergo depending on whether it is a weekday or weekend. For both days, the intervals between 0 and about 525 are uniform. This most likely represents when the subject was sleeping. The differences start at between 525 and roughly 800. On the weekdays, movement is most likely attributed to the subject getting ready to go to work or starting their day. On the weekends, movement is less frequent. This could be attributed to the subject sleeping in and perhaps making breakfast to start their day.

There is a huge jump at roughly 830 for the weekdays. This could be attributed to the subject walking or making their way to work. On the weekends this behaviour is mimicked as well, perhaps due to making their way to begin engaging in some extracurricular activities that the subject is engaged in. Now where it really gets different is at roughly the 1000 mark. On the weekdays, this could be reflected with the subject being at work and there is not much movement. A good indication of this behaviour could be attributed to the fact that the subject has a low stress job that requires minimal movement. On the weekends, the behaviour varies wildly. This could be attributed to the extra amount of movement required to engage in such extracurricular activities.

Where it starts to become similar again is at roughly the 1700 mark. This could mean that during the weekdays, the subject is getting ready to head home for the day which is why there is an increased amount of moment in comparison to the interval between 1000 to 1700. At the same time with the weekends, there is a relative amount of increased moment after the 1700 mark but it could reflect that at this time of day, the subject may be socializing.

All in all, this seems to reflect the behaviour of an individual going through a normal work day during the weekdays, and having a relaxing time on the weekends.