Introduction

This assignment makes use of data from a personal activity monitoring device. This device collects data at 5 minute intervals through out the day. The data consists of two months of data from an anonymous individual collected during the months of October and November, 2012 and include the number of steps taken in 5 minute intervals each day.

Data

The data for this assignment can be downloaded from the course web site: https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip

The variables included in this dataset are:

- steps: Number of steps taking in a 5-minute interval (missing values are coded as NA)
- date: The date on which the measurement was taken in YYYY-MM-DD format
- interval: Identifier for the 5-minute interval in which measurement was taken

The dataset is stored in a comma-separated-value (CSV) file and there are a total of 17,568 observations in this dataset.

Data analysis

Loading and preprocessing the data

The data loading is with: read.csv() function. Then changing to the data frame tbl form for further analysis:

```
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
```

```
setwd("C:/my home/coursera/Data Science Specialization/Reproducible Research/assignments/assignment 1/report")



dat<-read.csv("../activity.csv")

tbl_dat<-tbl_df(dat)
```

```
rm(dat)

tbl_dat
```

```
## Source: local data frame [17,568 x 3]

##    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

## 7     NA 2012-10-01       30

## 8     NA 2012-10-01       35

## 9     NA 2012-10-01       40

## 10    NA 2012-10-01       45

## ..   ...        ...      ...
```

What is mean total number of steps taken per day?

- removing the NA steps rows from the data frame

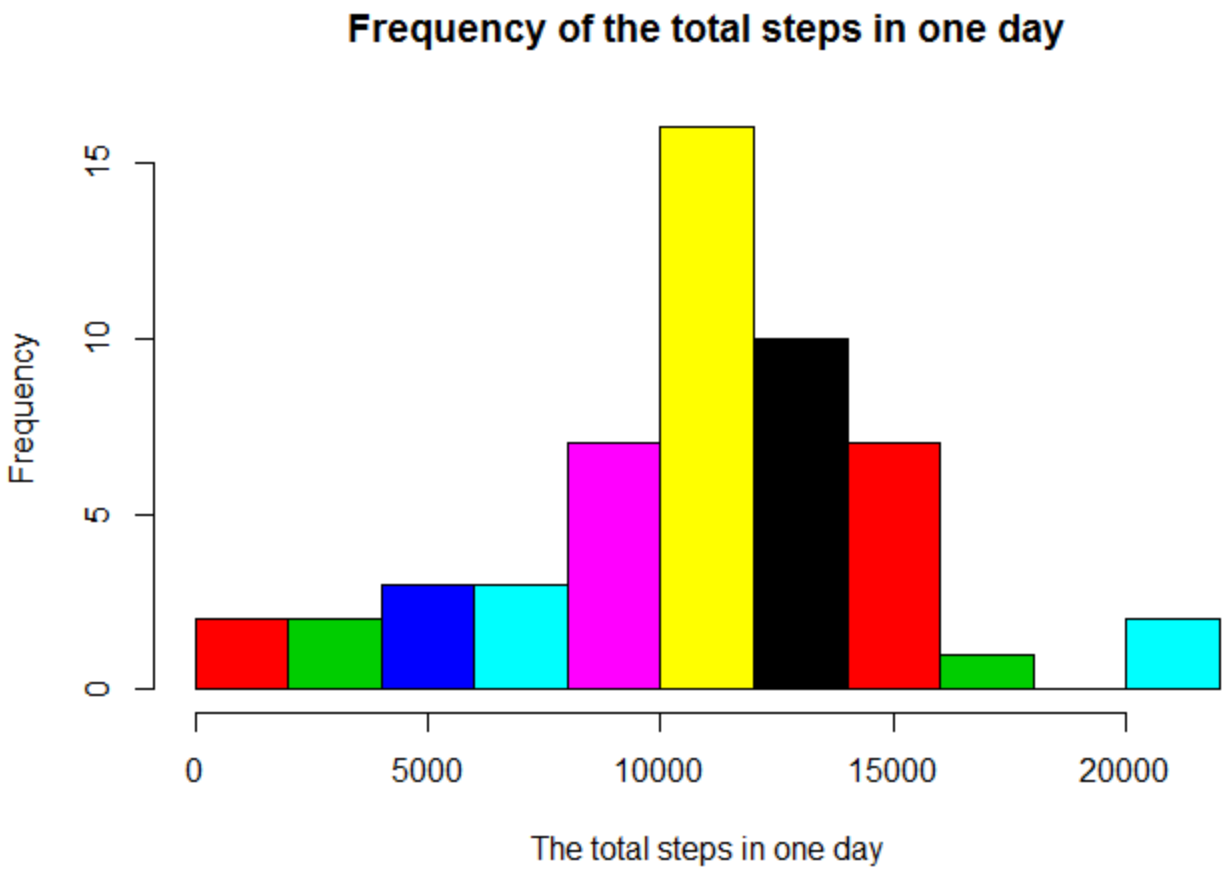
```
rm_na<-filter(tbl_dat,!is.na(steps))
```

- Grouping the data set by date

```
by_date<-group_by(rm_na,date)   ##%>%
```

- getting and plotting the total steps taken per day

```
total<-summarize(by_date,total=sum(steps))

with(total,hist(total,10,col=date,

main="Frequency of the total steps in one day",

xlab="The total steps in one day"))
```



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

```
mean<-summarize(total,mean(total))

median<-summarize(total,quantile(total,probs=0.5))
```

So the mean total steps taken per is: 1.0766×10^4.

The median is: 1.0765×10^4.

What is the average daily activity pattern?

- group data by interval

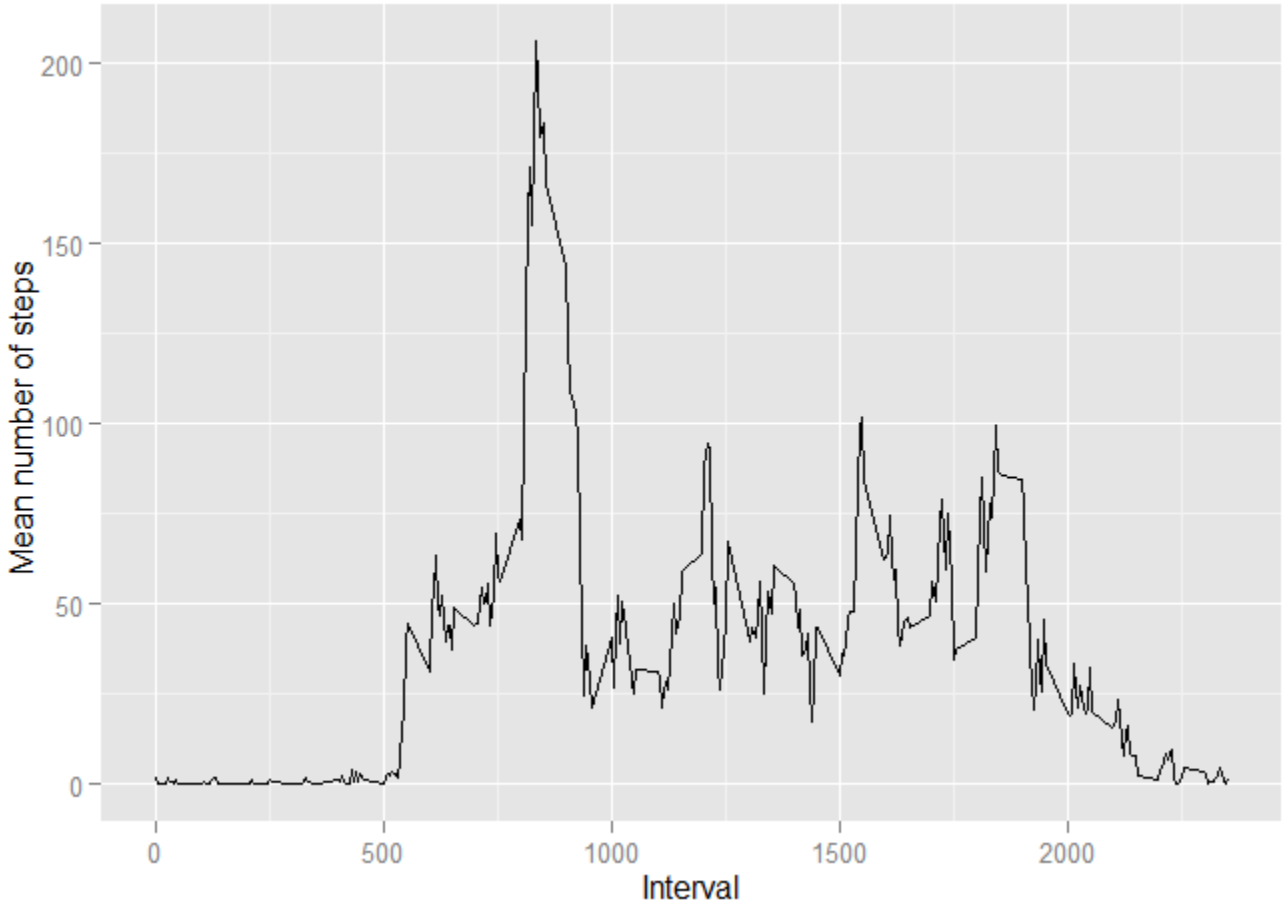
```
by_int<-group_by(rm_na,interval)
```

- Average data by interval across all dates

```
mn_int<-summarize(by_int,steps=mean(steps))
```

- PLotting the time series sequence of average interval steps

```
library(ggplot2)

ggplot(data=mn_int,aes(x=interval,y=steps))+geom_line()+ xlab("Interval")+

ylab("Mean number of steps ")
```



- Finding the maximum average steps interval

```
int_maxsteps<-mn_int[which(mn_int$steps==max(mn_int$steps)),]$interval
```

The maximum average steps taken interval is 835 to 840 minutes interval, which taken the 206.1698 stpes in that 5 minute interval.

Imputing missing values

The approach of fiiling the missing value is filling the missed steps interval as the average steps values of that interval.

The steps as:

- Finding the missing steps

```
na_steps<-is.na(tbl_dat$steps)
```

The total number of missing values in the dataset is `sum(na_steps)`

- Join mean interval steps across all days to the original dat set tbl_dat

```
join_tbl<-left_join(tbl_dat,mn_int,by="interval")
```

- Assigning the missing stpes field with the mean steps of that interval

```
join_tbl[na_steps,]$steps.x<-join_tbl[na_steps,]$steps.y
```

- Selecting the needed data set from the joined data set

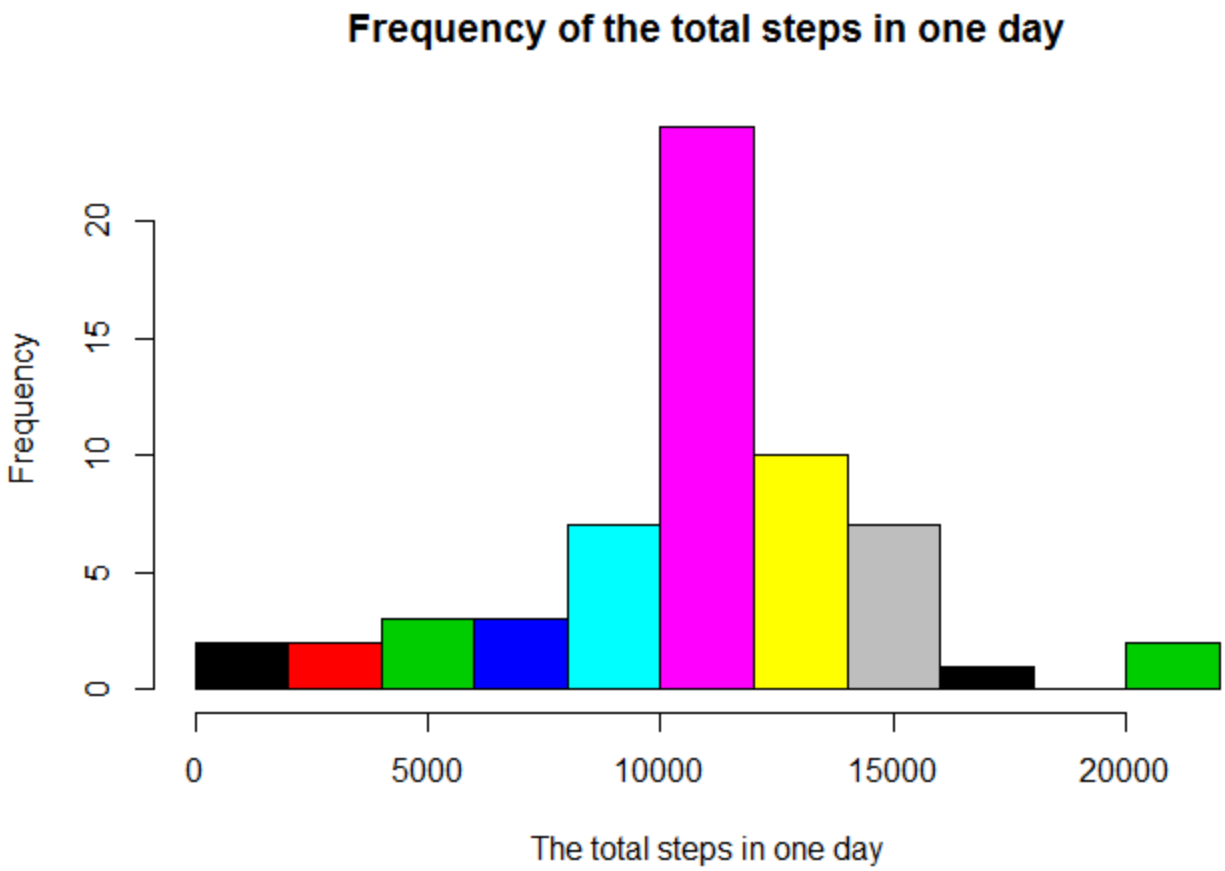
```
filled_df<-select(join_tbl,steps=steps.x,date,interval)
```

- grouping the selected data set by date

```
filled_by_date<-group_by(filled_df,date)
```

- New total stpes taken per day

```
filled_total<-summarize(filled_by_date,total=sum(steps))

with(filled_total,hist(total,10,col=date,

main="Frequency of the total steps in one day",

xlab="The total steps in one day"))
```



- New average and median total steps taken per day

```
filled_mean<-summarize(filled_total,mean(total))

filled_median<-summarize(filled_total,quantile(total,probs=0.5))
```

After filling the NA records, the new mean total steps taken per day become to: 1.0766×10^4.

The new median becomes to: 1.0766×10^4.

Are there differences in activity patterns between weekdays and weekends

- Loading the “lubridate” library

```
library(lubridate)
```

- Adding a new coloumn to filling data frame to indicate whether the date is the weekend

```
filled_df<-mutate(filled_df, weekend=factor(weekdays(ymd(date)) %in% c('Saturday','Sunday')))
```

- Group the new filled data frame by weekend and interval

```
filled_by_int_wk<-group_by(filled_df,weekend,interval)
```

- Calcating the average interval mean based on groups

```
filled_mn_int<-summarize(filled_by_int_wk,steps=mean(steps))
```

- Plotting the averge interval steps taken per day for the weekday and weekend individually

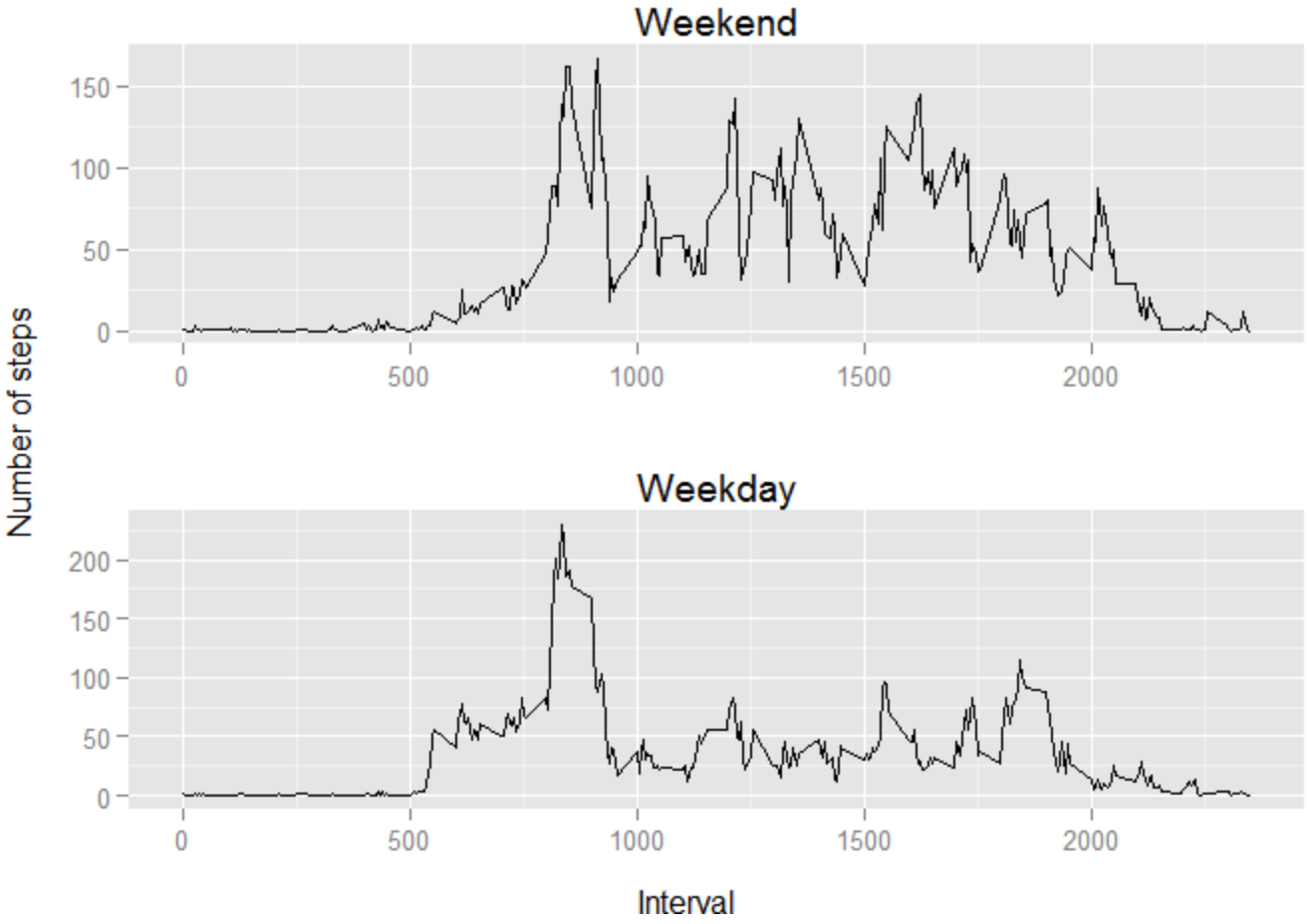
```
p_wday<-ggplot(data=subset(filled_mn_int,weekend==FALSE),aes(x=interval,y=steps))+geom_line()+

ggtitle("Weekday")+theme(axis.title = element_blank())

p_wend<-ggplot(data=subset(filled_mn_int,weekend==TRUE),aes(x=interval,y=steps))+geom_line()+

ggtitle("Weekend")+theme(axis.title = element_blank())
```

- Plotting two plot in one graph

```
library(gridExtra)
```

```
## Loading required package: grid
```

```
grid.arrange(p_wend,p_wday, ncol=1,sub="Interval",left="Number of steps")
```


Weekend
Weekday
Number of steps
Interval
0
50
100
150
200
0
500
1000
1500
2000