Reproducible Reports Assignment - Course Project 1

The data in this assignment, provided by course instructors, was from a personal activity monitoring device that collects data at 5-minute intervals. The data was downloaded from the course website. The variables were: 1) steps, number of steps taken at 5-min intervals; 2) date, date measurement was taken, and 3) interval, indentifier for the 5-min intervals. R codes to read and analyze the data, along with analysis results are presented below.

# read data  
setwd("C:/Users/user/DataScience\_Coursera/repdata-data-activity/")  
activity <- read.csv("activity.csv", stringsAsFactors=FALSE, header=TRUE)  
  
#format variables  
activity$date <- as.Date(as.character(activity$date))  
  
# exclude missing values  
activity\_complete <- na.exclude(activity)

# What is the mean total number of steps taken per day?

total\_steps\_per\_day <- aggregate(activity\_complete$steps, by=list(date=activity\_complete$date), sum)  
  
colnames(total\_steps\_per\_day) <- c("date", "total.steps")

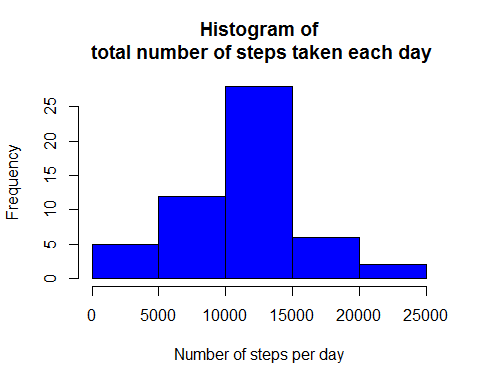
# 1. Total number of steps per day

total\_steps\_per\_day

## date total.steps  
## 1 2012-10-02 126  
## 2 2012-10-03 11352  
## 3 2012-10-04 12116  
## 4 2012-10-05 13294  
## 5 2012-10-06 15420  
## 6 2012-10-07 11015  
## 7 2012-10-09 12811  
## 8 2012-10-10 9900  
## 9 2012-10-11 10304  
## 10 2012-10-12 17382  
## 11 2012-10-13 12426  
## 12 2012-10-14 15098  
## 13 2012-10-15 10139  
## 14 2012-10-16 15084  
## 15 2012-10-17 13452  
## 16 2012-10-18 10056  
## 17 2012-10-19 11829  
## 18 2012-10-20 10395  
## 19 2012-10-21 8821  
## 20 2012-10-22 13460  
## 21 2012-10-23 8918  
## 22 2012-10-24 8355  
## 23 2012-10-25 2492  
## 24 2012-10-26 6778  
## 25 2012-10-27 10119  
## 26 2012-10-28 11458  
## 27 2012-10-29 5018  
## 28 2012-10-30 9819  
## 29 2012-10-31 15414  
## 30 2012-11-02 10600  
## 31 2012-11-03 10571  
## 32 2012-11-05 10439  
## 33 2012-11-06 8334  
## 34 2012-11-07 12883  
## 35 2012-11-08 3219  
## 36 2012-11-11 12608  
## 37 2012-11-12 10765  
## 38 2012-11-13 7336  
## 39 2012-11-15 41  
## 40 2012-11-16 5441  
## 41 2012-11-17 14339  
## 42 2012-11-18 15110  
## 43 2012-11-19 8841  
## 44 2012-11-20 4472  
## 45 2012-11-21 12787  
## 46 2012-11-22 20427  
## 47 2012-11-23 21194  
## 48 2012-11-24 14478  
## 49 2012-11-25 11834  
## 50 2012-11-26 11162  
## 51 2012-11-27 13646  
## 52 2012-11-28 10183  
## 53 2012-11-29 7047

# 2. Histogram of total number of steps per day

hist(total\_steps\_per\_day$total.steps, col="blue", xlab="Number of steps per day", main="Histogram of\n total number of steps taken each day")



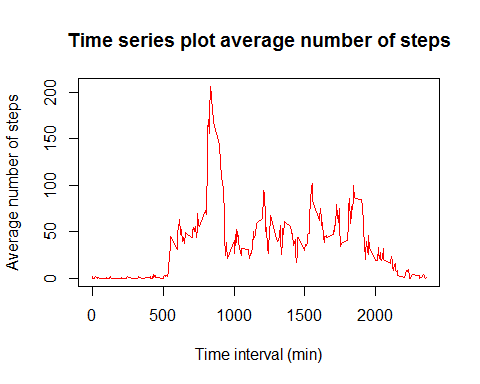
# 3.Mean and median total number of steps per day

summary(total\_steps\_per\_day$total.steps)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 41 8841 10760 10770 13290 21190

# What is the average daily activity pattern?

average\_steps\_per\_interval <- aggregate(activity\_complete$steps, by=list(interval=activity\_complete$interval), mean)  
  
colnames(average\_steps\_per\_interval) <- c("interval", "average.steps")  
  
plot(x=average\_steps\_per\_interval$interval, y=average\_steps\_per\_interval$average.steps, type='l', col=2, xlim=c(0,max(average\_steps\_per\_interval$interval)),   
 xlab="Time interval (min)", ylab="Average number of steps",  
 main="Time series plot average number of steps")



# The 5-minute interval that displays the max average number of steps:

average\_steps\_per\_interval[which.max(average\_steps\_per\_interval$average.steps),]

## interval average.steps  
## 104 835 206.1698

# Imputing missing values

# 1. Number of missing values

sum(is.na(activity))

## [1] 2304

# 2. creating a new data set with mean imputation

require(Hmisc)

## Loading required package: Hmisc

## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

## Loading required package: ggplot2

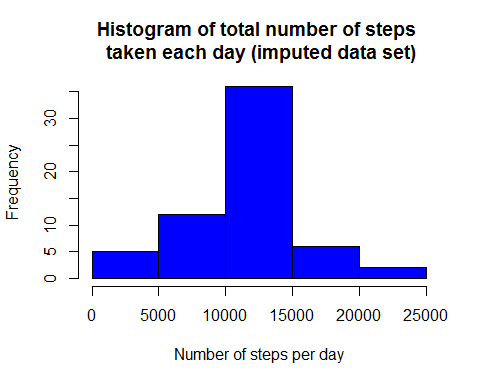
##   
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:base':  
##   
## format.pval, round.POSIXt, trunc.POSIXt, units

activity$imputed.steps <- with(activity, impute(steps, mean))

# 3. Histogram of total number of steps per day in imputed data set

total\_number\_steps\_a\_day <- aggregate(activity$imputed.steps, by=list(date=activity$date), sum)  
  
colnames(total\_number\_steps\_a\_day) <- c("date", "total.steps")  
  
hist(total\_number\_steps\_a\_day$total.steps, col="blue", xlab="Number of steps per day", main="Histogram of total number of steps \n taken each day (imputed data set)")



# 3.Mean and median total number of steps per day in imputed data set

summary(total\_number\_steps\_a\_day$total.steps)

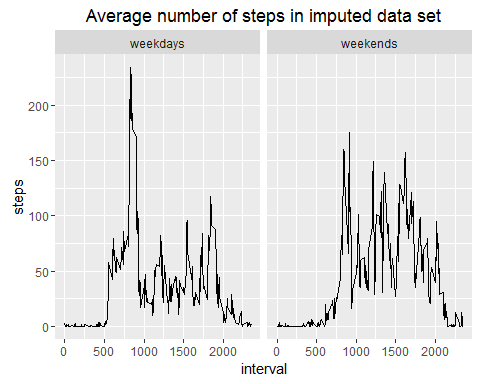
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 41 9819 10770 10770 12810 21190

Yes, imputation produced different values compared to the data set earlier. The imputed data set contained additional data points between 5,000 & 10,000 steps. As seen in the histogram, the imputed data set has additional data points between the first quartile and the median.

## Activity patterns between weekdays and weekends

weekdays.list <- c("Monday", "Tuesday", "Wednesday", "Thursday","Friday")  
  
activity$date.group <- c("weekends", "weekdays")[(weekdays(activity$date) %in% weekdays.list)+1L]  
  
activity$date.group <- as.factor(activity$date.group)  
  
ggplot(activity, aes(interval, steps)) +   
 geom\_line(stat="summary", fun.y="mean") +   
 facet\_grid(. ~ date.group) +  
 ylab("steps") +  
 xlab("interval") +  
 ggtitle("Average number of steps in imputed data set")

## Warning: Removed 2304 rows containing non-finite values (stat\_summary).



yes, activity patterns between weekdays and weekends are different. In general, weekday acttivities data are skewed to the left where as weekend activities are somewhat normally distributed.