Reproducible Research Project 1

06/11/2017

1 Introduction

It is now possible to collect a large amount of data about personal movement using activity monitoring devices such as a Fitbit, Nike Fuelband, or Jawbone Up. These type of devices are part of the “quantified self” movement - a group of enthusiasts who take measurements about themselves regularly to improve their health, to find patterns in their behavior, or because they are tech geeks. But these data remain under-utilized both because the raw data are hard to obtain and there is a lack of statistical methods and software for processing and interpreting the data.

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.

2 Data For the Analysis

The data can be downloaded from the course web site:

Dataset: Activity Monitoring Data [52K]

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.

3 Prepare the environment and clean data

```r
list.packages<-c("ggplot2","lattice","plyr")
new.packages<-list.packages[!(list.packages %in% installed.packages()[,"Package"])]
if(length(new.packages)) install.packages(new.packages)

library(ggplot2)
library(lattice)
library(plyr)

# load the data from local
data<-read.csv("activity.csv")

# clean the data and get the new without NAs

data_cleaned<-data[complete.cases(data),]
```

4 Mean and median total number of steps taken per day

```
# Aggregate the data by date

data_by_date<-aggregate(steps~date,data=data_cleaned,sum)
steps_avg<-mean(data_by_date$steps)
as.integer(steps_avg)
```

```
## [1] 10766
```

```
steps_median<-median(data_by_date$steps)
as.integer(steps_median)
```

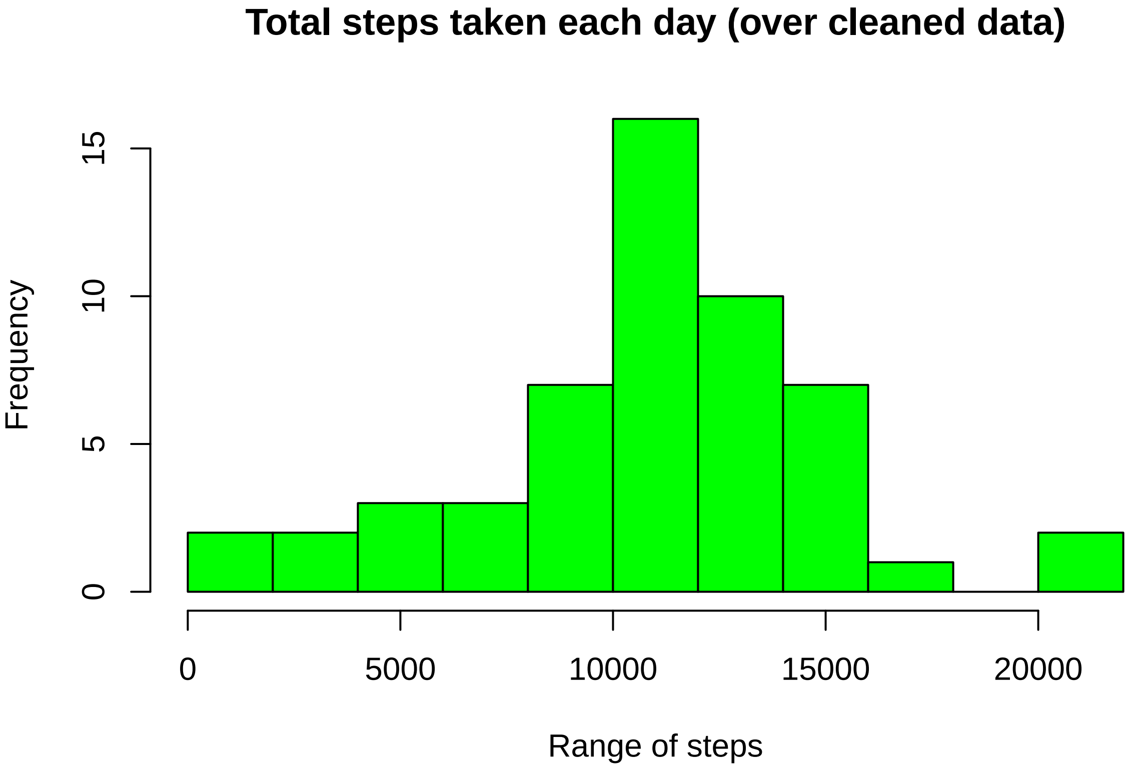
```
## [1] 10765
```

The average number of steps taken each day was 10766 steps.

The median number of steps taken each day was 10765 steps.

5 Histogram for each day steps

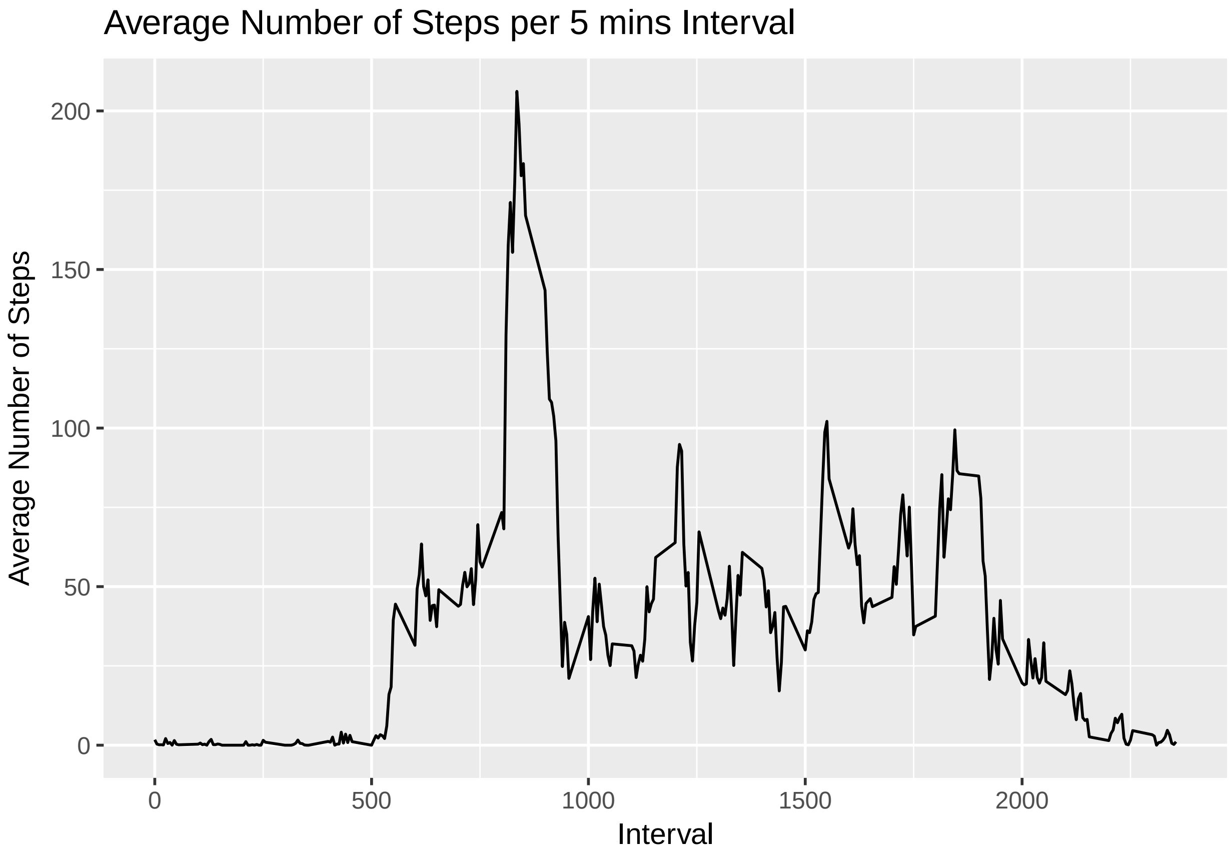
```
hist(data_by_date$steps, main = "Total steps taken each day (over cleaned data)", xlab = "Range of steps
```



6 Time series plot of the average number of steps taken

```
#create average number of steps per interval
intervalTable <- ddply(data_cleaned, .(interval), summarize, Avg = mean(steps))
# Create line plot of average number of steps per interval
```

```
g <- ggplot(intervalTable, aes(x=interval, y=Avg), xlab = "Interval", ylab="Average Number of Steps")
g + geom_line()+xlab("Interval")+ylab("Average Number of Steps")+ggtitle("Average Number of Steps per 5
```



7 The 5-minute interval that, on average, contains the maximum number of steps

```
# Maximum steps by interval
maxSteps <- max(intervalTable$Avg)

# Which interval contains the maximum average number of steps
intervalTable[intervalTable$Avg==maxSteps,1]
```

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

8 Code to describe and show a strategy for imputing missing data

Imupting by substituting the missing steps with the average 5-minute interval based on the day of the week.

```
data_cleaned$day<-weekdays(as.Date(data_cleaned$date))

avgTable <- ddply(data_cleaned, .(interval, day), summarize, Avg = mean(steps))

# Create dataset with all NAs for substitution

data$day <- weekdays(as.Date(data$date))
nadata<- data[is.na(data$steps),]
```

```
## Merge NA data with average weekday interval for substitution
newdata<-merge(nadata, avgTable, by=c("interval", "day"))

## Reorder the new substituded data in the same format as clean data set
newdata2<- newdata[,c(5,4,1,2)]
colnames(newdata2)<- c("steps", "date", "interval", "day")

## Merge the NA averages and non NA data together
mergeData <- rbind(data_cleaned, newdata2)


# #Create sum of steps per date to compare with step 1
sumTable2 <- aggregate(mergeData$steps ~ mergeData$date, FUN=sum, )
colnames(sumTable2)<- c("Date", "Steps")
# Mean of Steps with NA data taken care of
as.integer(mean(sumTable2$Steps))
```

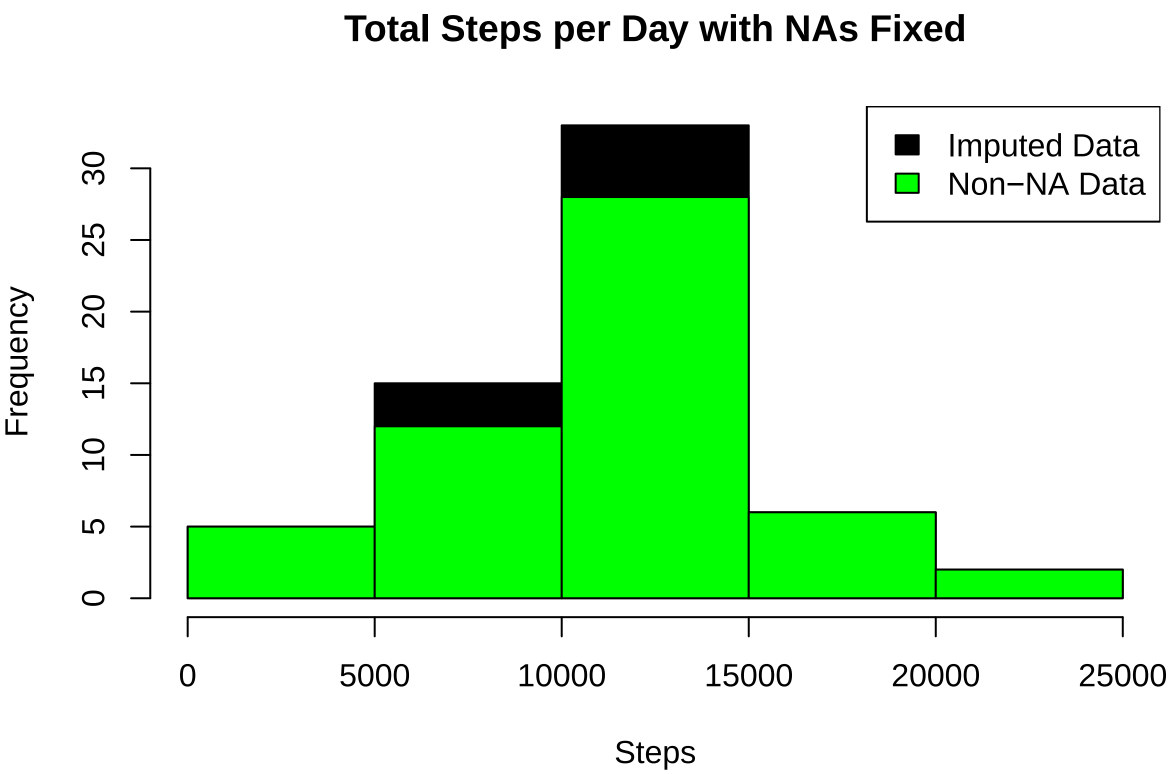
```
## [1] 10821
```

```
# Median of Steps with NA data taken care of
as.integer(median(sumTable2$Steps))
```

```
## [1] 11015
```

9 Histogram of the total number of steps taken each day after missing values are imputed

```
# Creating the histogram of total steps per day, categorized by data set to show impact
hist(sumTable2$Steps, breaks=5, xlab="Steps", main = "Total Steps per Day with NAs Fixed", col="Black")
hist(data_by_date$steps, breaks=5, xlab="Steps", main = "Total Steps per Day with NAs Fixed", col="Green
legend("topright", c("Imputed Data", "Non-NA Data"), fill=c("black", "green") )
```

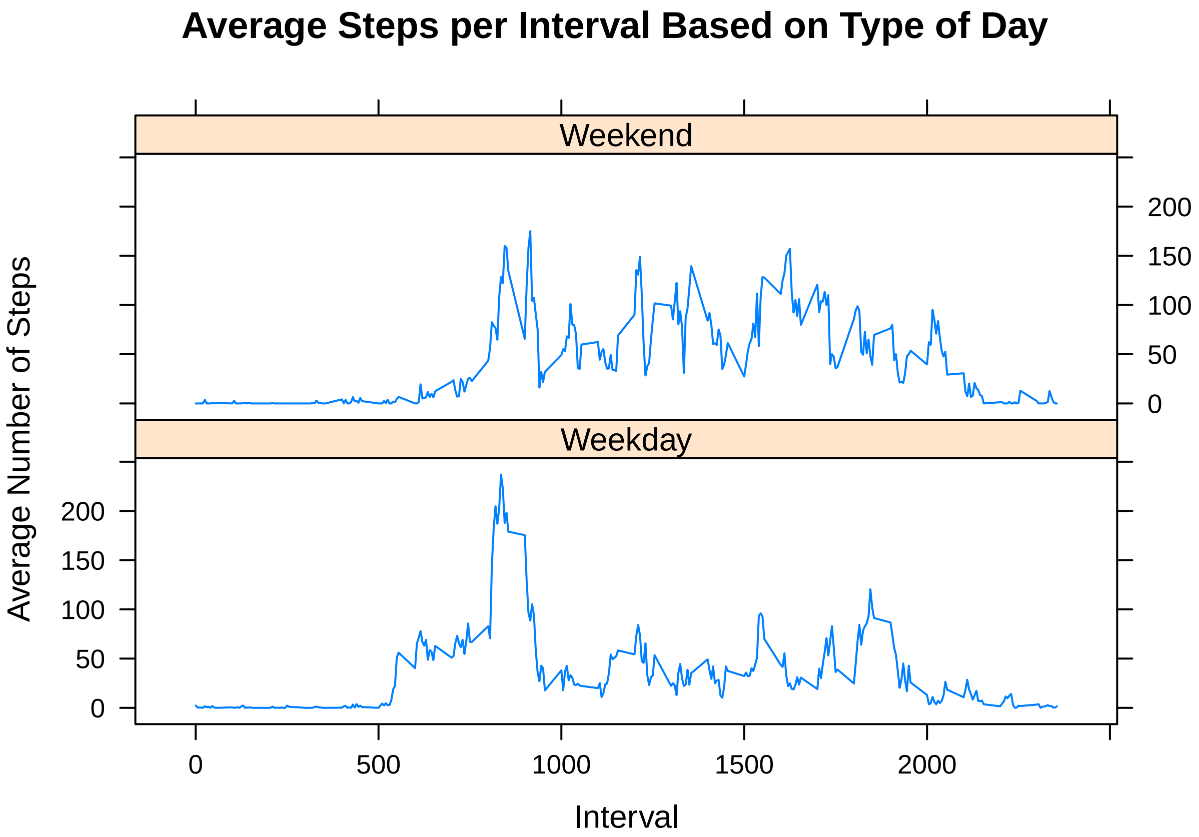


10 Panel plot comparing the average number of steps taken per 5-minute interval across weekdays and weekends

```
# Create new category based on the days of the week
mergeData$DayCategory <- ifelse(mergeData$day %in% c("Saturday", "Sunday"), "Weekend", "Weekday")

# Summarize data by interval and type of day
intervalTable2 <- ddply(mergeData, .(interval, DayCategory), summarize, Avg = mean(steps))

#Plot data in a panel plot
xyplot(Avg~interval|DayCategory, data=intervalTable2, type="l",  layout = c(1,2),
       main="Average Steps per Interval Based on Type of Day",
       ylab="Average Number of Steps", xlab="Interval")
```


Average Steps per Interval Based on Type of Day
Weekend
Weekday
Average Number of Steps
0
50
100
150
200
0
50
100
150
200
0
500
1000
1500
2000
Interval