PA1_template

Xu Zhang

Sunday, January 18, 2015

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:

Dataset: Activity monitoring data [52K] 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.

Loading and preprocessing the data

I download the data firstly, I put it in my directory folder.

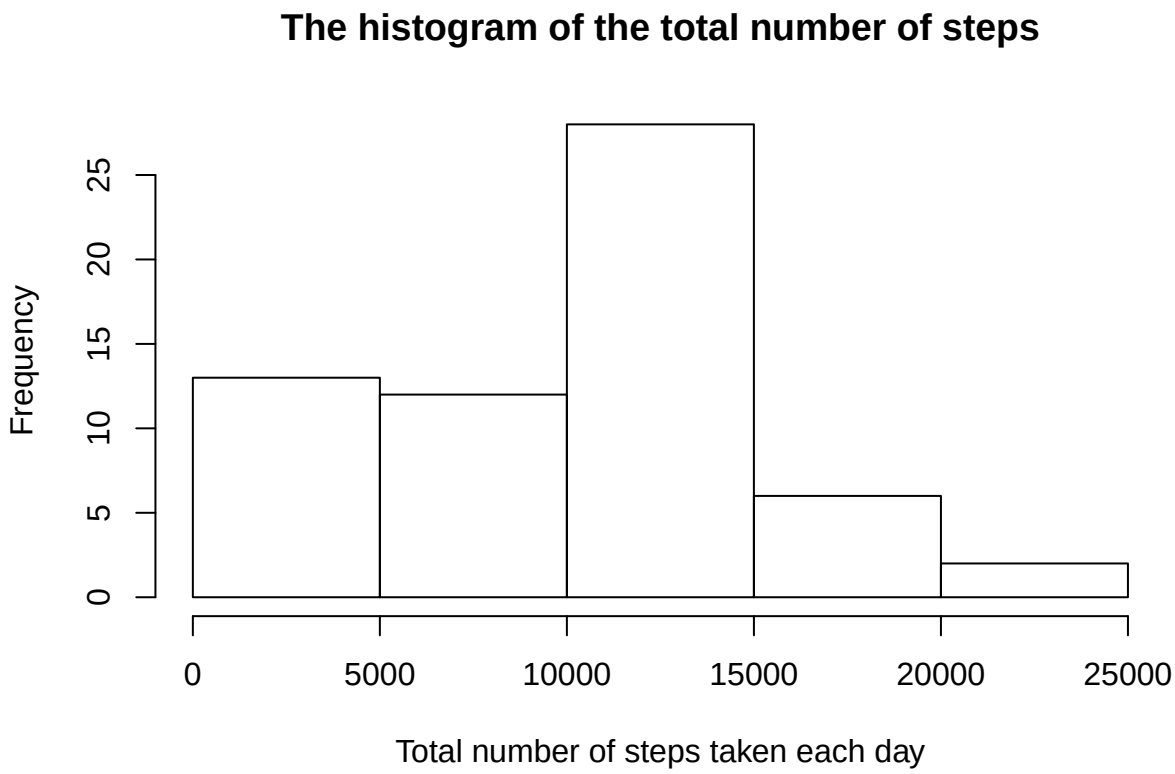
```
mydata<-read.csv(file.choose(),header=TRUE,sep=",")
```

What is mean total number of steps taken per day?

We ignore the missing values in the dataset.

- Make a histogram of the total number of steps taken each day

```
mydata1<-mydata
mydata1$steps[is.na(mydata1$steps)]<-0
data1<-mydata1
data2<-data1[,1:2]
library(data.table)
data2<-data.table(data2)
data3<-data2[,lapply(.SD,sum),by=data2$date]

#png(filename="plot1.png",width=480,height=480)
hist(data3$steps,xlab='Total number of steps taken each day',main='The histogram of the total number of
```




```
#dev.off()
```

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

The mean is:

```
mean(as.numeric(as.character(data3$steps)))
```

```
## [1] 9354.23
```

The median is:

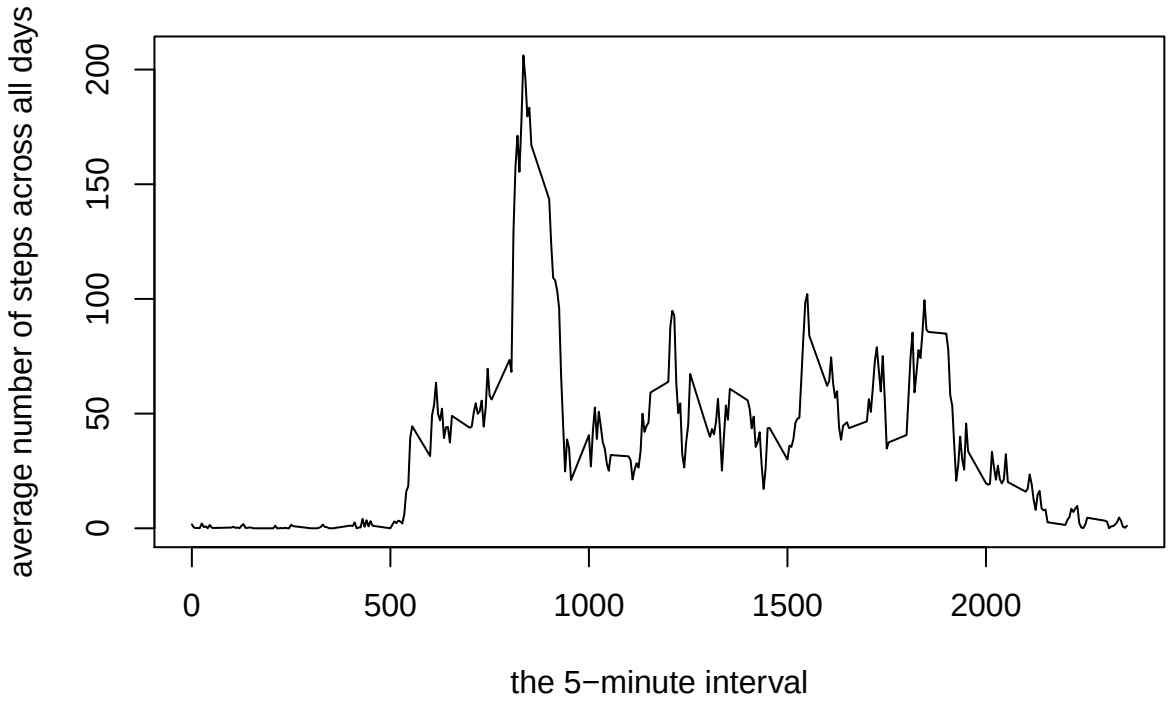
```
median(as.numeric(as.character(data3$steps)))
```

```
## [1] 10395
```

What is the average daily activity pattern?

- Make a time series plot (i.e. type = "l") of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis)

```
dataB1<-mydata[,c(1,3)]
library(data.table)
dataB1<-data.table(dataB1)
dataB2<-dataB1[,lapply(.SD,mean,na.rm = TRUE),by=dataB1$interval]

#png(filename="plot2.png",width=480,height=480)
plot(dataB2$dataB1,dataB2$steps,xlab="the 5-minute interval", ylab="average number of steps across all d
```




```
#dev.off()
```

- Which 5-minute interval, on average across all the days in the dataset, contains the maximum number of steps?

This is the maximum value:

```
max(dataB2$steps)
```

```
## [1] 206.1698
```

This gives the number of the row of the maximu value

```
which.max(dataB2$steps)
```

```
## [1] 104
```

This gives the value of the 5-minute interval:

```
dataB2[which.max(dataB2$steps)]
```

```
##    dataB1    steps
## 1:    835 206.1698
```

The maximual value of the 5-minute interval is 835.

Imputing missing values

Note that there are a number of days/intervals where there are missing values (coded as NA). The presence of missing days may introduce bias into some calculations or summaries of the data.

- Calculate and report the total number of missing values in the dataset (i.e. the total number of rows with NAs)

```
summary(mydata$steps)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00    0.00   37.38   12.00  806.00    2304
```

The total number of missing value is 2304.

- Devise a strategy for filling in all of the missing values in the dataset.

We take the mean of the 5-minute interval to replace the missing value "NA".

- Create a new dataset that is equal to the original dataset but with the missing data filled in.

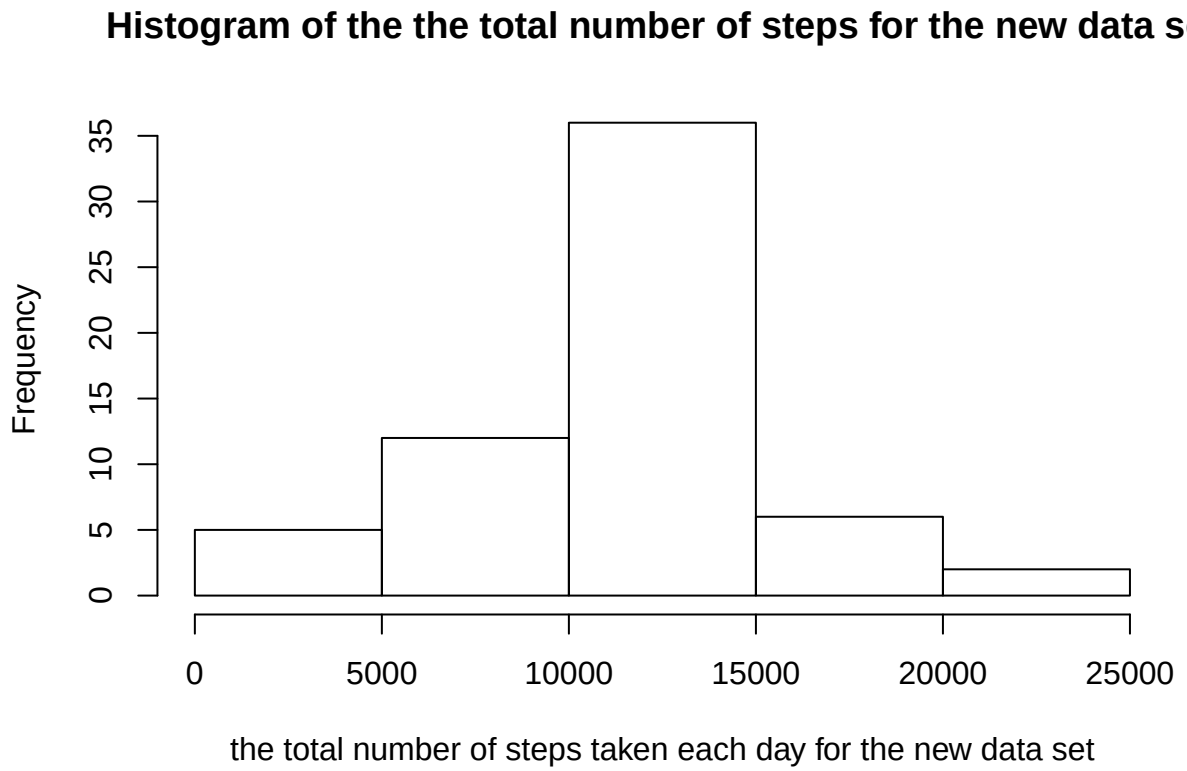
The data set dataC1 is a new dataset that is equal to the original dataset but with the missing data filled in. And, the first ten rows of this data set is listed as follows:

```
for(i in 1:length(dataC1$steps)){
        if(is.na(dataC1$steps[i])){
                index1<-match(dataC1$interval[i],dataC2$dataB1)
                dataC1$steps[i]<-dataC2$steps[index1]
        }
}
head(dataC1,n=10)
```

```
##        steps       date interval
## 1  1.7169811 2012-10-01        0
## 2  0.3396226 2012-10-01        5
## 3  0.1320755 2012-10-01       10
## 4  0.1509434 2012-10-01       15
## 5  0.0754717 2012-10-01       20
## 6  2.0943396 2012-10-01       25
## 7  0.5283019 2012-10-01       30
## 8  0.8679245 2012-10-01       35
## 9  0.0000000 2012-10-01       40
## 10 1.4716981 2012-10-01       45
```

- Make a histogram of the total number of steps taken each day and Calculate and report the mean and median total number of steps taken per day. Do these values differ from the estimates from the first part of the assignment? What is the impact of imputing missing data on the estimates of the total daily number of steps?

```
dataC3<-dataC1[,1:2]
library(data.table)
dataC3<-data.table(dataC3)
dataC4<-dataC3[,lapply(.SD,sum),by=dataC3$date]
#png(filename="plot3.png",width=480,height=480)
hist(dataC4$steps, xlab="the total number of steps taken each day for the new data set",
main="Histogram of the the total number of steps for the new data set")
```



```
#dev.off()
```

The mean is:

```
mean(as.numeric(as.character(dataC4$steps)))
```

```
## [1] 10766.19
```

The median is:

```
median(as.numeric(as.character(dataC4$steps)))
```

```
## [1] 10766.19
```

The mean and median are different from the first part of this assignment.

By observing these two histograms, we notice that the frequency of the average steps less than 5000 is larger in the new data set,

Are there differences in activity patterns between weekdays and weekends?

Use the dataset with the filled-in missing values for this part.

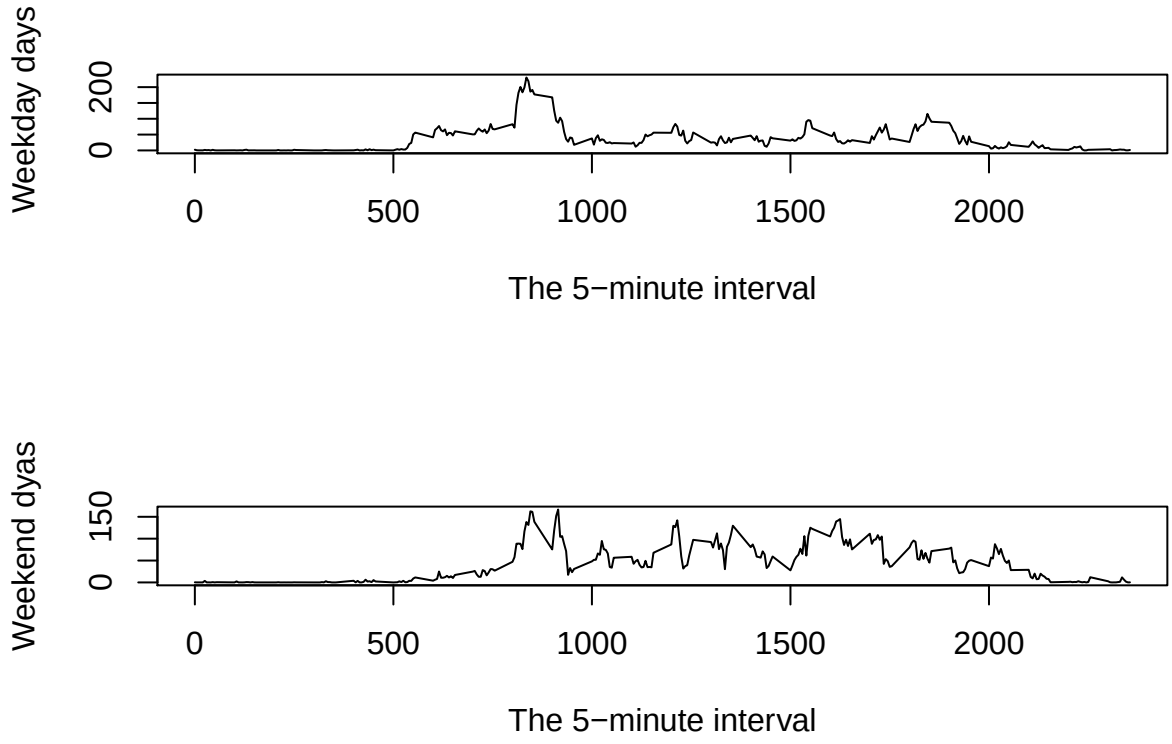
- Create a new factor variable in the dataset with two levels - "weekday" and "weekend" indicating whether a given date is a weekday or weekend day.

The factor variable "Work" indicates whether a given date is a weekday or weekend day.

```
dataC1$date<-as.Date(dataC1$date,"%Y-%m-%d")

dataC5<-weekdays(dataC1$date)

# Put "Monday"--"Friday" to "weekday", "Saturday"-"Sunday" to "weekend"
for(i in 1:length(dataC5)){if(dataC5[i]=="Saturday" | dataC5[i]=="Sunday"){dataC5[i]<-"weekend"}  else

#Add dataC5 to the dataC1, which the missing value is adjusted
Work<-dataC5
dataC6<-cbind(dataC1,Work)

#split the table according to "Work": weekend and weekday
# dataC8$weekday dataC8$weekend
dataC8<-split(dataC7,dataC7$Work)
#weekday
dataC9<-dataC8$weekday[,lapply(.SD,mean),by=dataC8$weekday$interval]
#weekend
dataC10<-dataC8$weekend[,lapply(.SD,mean),by=dataC8$weekend$interval]
```

- Make a panel plot containing a time series plot (i.e. type = "l") of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all weekday days or weekend days (y-axis).

```
#png(filename="plot4.png",width=480,height=480)
par(mfrow=c(2,1))
#weekday
plot(dataC9$dataC8,dataC9$steps,xlab="The 5-minute interval", ylab=" Weekday days", type="l")
#weekend
plot(dataC10$dataC8,dataC10$steps,xlab="The 5-minute interval", ylab="Weekend dyas",type="l")
```




```
#dev.off()
```

From these two graphs, there is more exercise on weekend days than that on weekday days.