ReproducibleResearchAssignment1

N

August 19, 2018

Setup and Data

Set up and get the data from the Coursera Website.

```
setwd("C:/Nick/07 R/6JohnHopkins/5 Reproducible Research/Assignment1")
download.file("https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip","C:/Nick/07 R/6JohnH
zipF<- "C:\\Nick\\07 R\\6JohnHopkins\\5 Reproducible Research/Assignment1/repdata_Fdata_Factivity.zip"
unzip(zipF)
activity<-read.csv("activity.csv")
head(activity)
```

```
##   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
```

```
tail(activity)
```

```
##       steps       date interval
## 17563    NA 2012-11-30     2330
## 17564    NA 2012-11-30     2335
## 17565    NA 2012-11-30     2340
## 17566    NA 2012-11-30     2345
## 17567    NA 2012-11-30     2350
## 17568    NA 2012-11-30     2355
```

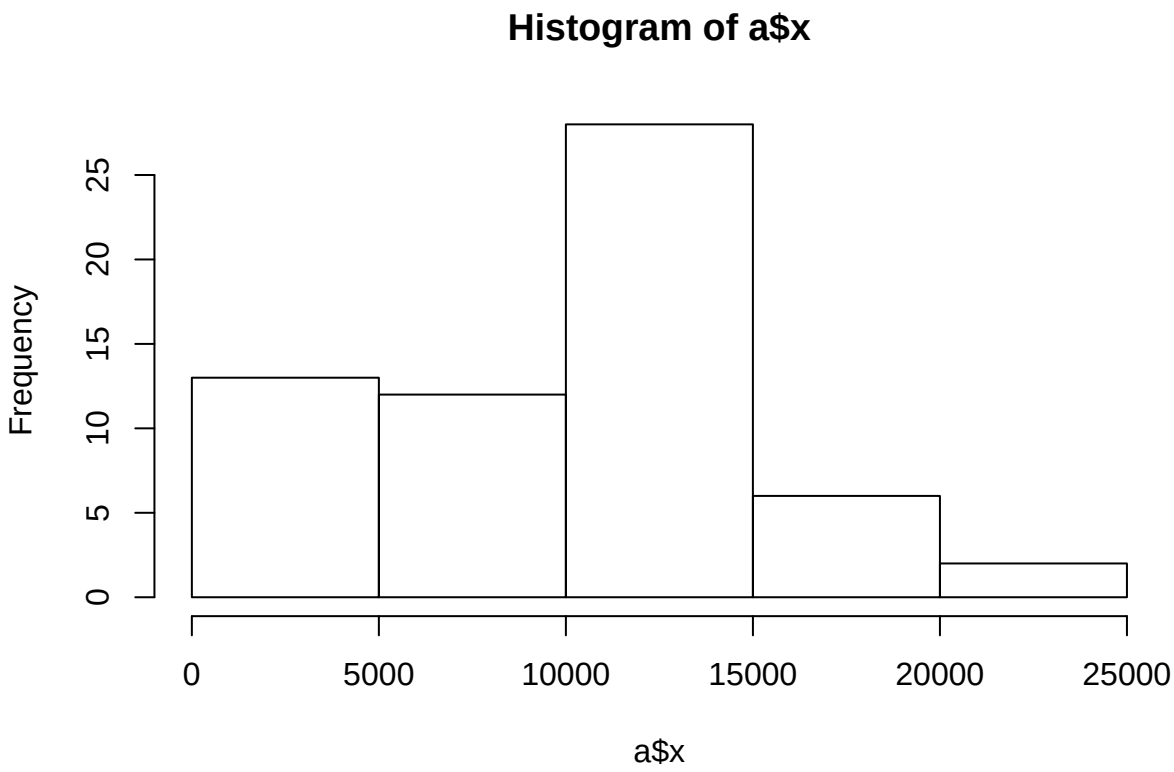
Lots of NA and zero data in the activity set:
- steps {NA, 0, integer/count data}
- date {yyyy-mm-dd format}
- interval {increments of time in 5 minutes}

NAs are likely to be days the battery was run down or the wearable was not used.

These should be excluded from the analysis.

Part1: Calculate the total number of steps taken per day, ploy the histogram and calculate the mean and median (Excludes NA records)

```
a<-aggregate(activity$steps, by=list(Category=activity$date), FUN=sum,na.rm=TRUE)
hist(a$x)
```




```
b<-round(mean(a$x, na.rm=TRUE),0); c<-round(median(a$x, na.rm=TRUE),0)
b<-noquote(format(b, digits=9, big.mark=",")); c<-noquote(format(c, digits=9, big.mark=","))
```

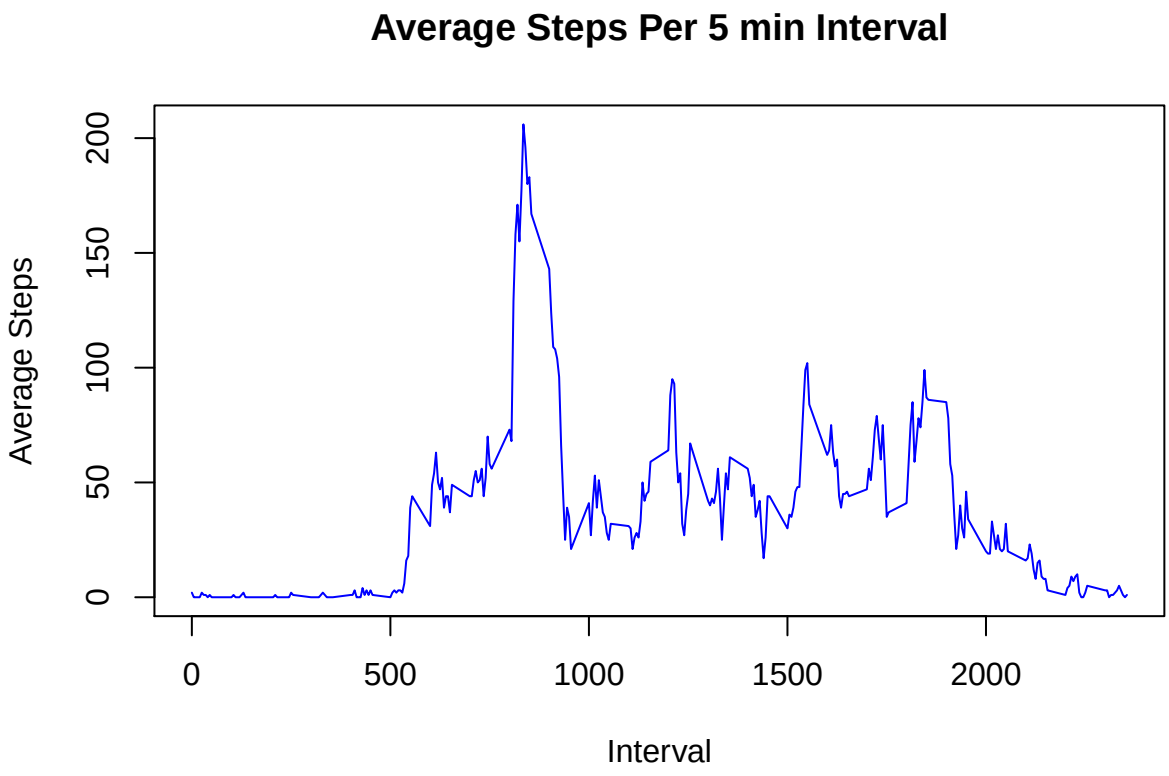
The mean is 9,354 and the median is 10,395.

Part2: Find the Average Daily Pattern of All Time Intervals Excluding NAs

```
#sum and count by day, then create h to give the average daily steps
d<-aggregate(activity$steps, by=list(Category=activity$interval), FUN=sum,na.rm=TRUE)
e<-is.na(activity$steps)  #find the NAs
f<-activity[!e,] #get the number of rows
g<-data.frame(table(f$interval))
h<-data.frame(d,g, round(d$x/g$Freq))
names(h) <- c("interval", "totalSteps","interval_","obs","averageSteps")
h<-h[,-3]
```

Having calculated the data, now create the plot and calculate the mean:

```
#now plot the time series
plot(h$interval,h$averageSteps, type="l", col="blue",xlab="", ylab="")
title(main="Average Steps Per 5 min Interval", xlab="Interval", ylab="Average Steps")
```




```
i<-as.vector(h[h$averageSteps==max(h$averageSteps),1])
```

The interval with the highest mean number of steps is 835.

Inputing Missing Values

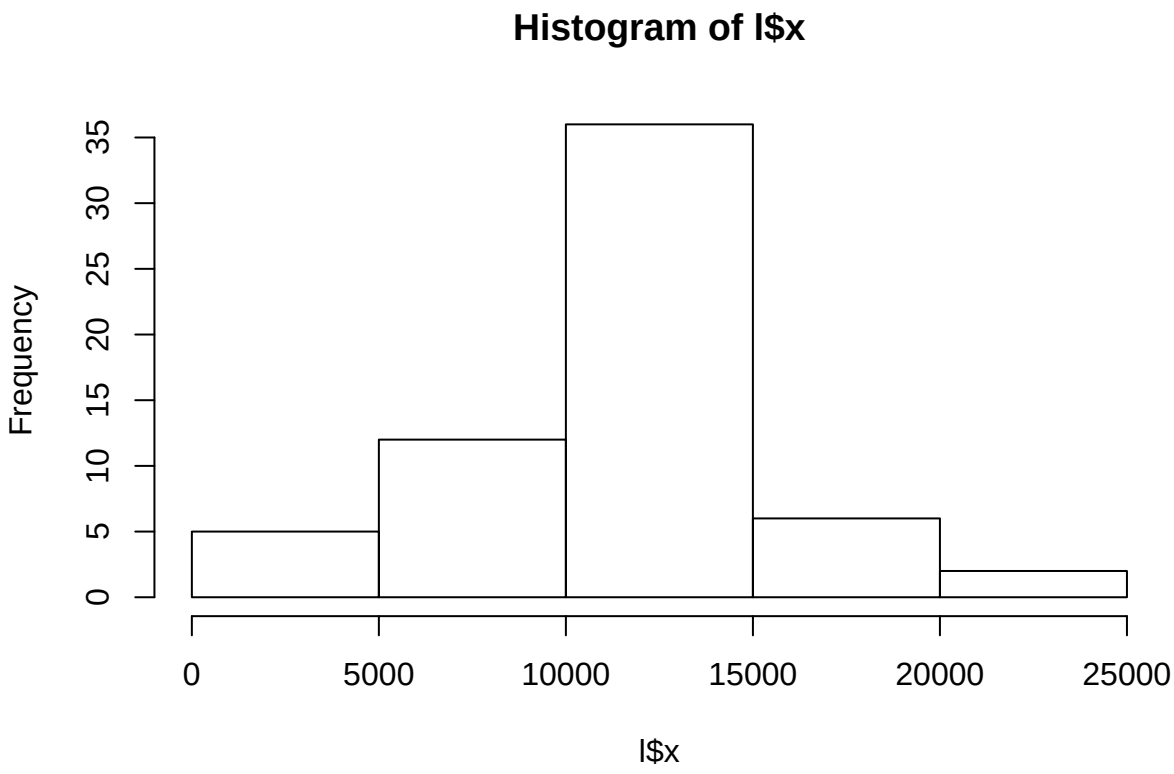
Quantify, Identify and Backfill the NA values

```
#count the number of NA records
j<-sum(e); j<-noquote(format(j, digits=9, big.mark=","))
#use the mean to fill in the dataset (merge on "interval")
k<-merge(x=activity, y=h[,c(1,4)], by.x=c("interval"),by.y=c("interval"))
k <- k[order(k[,3],k[,1]),]   #re-order
steps<-ifelse(is.na(k$steps),round(k$averageSteps),k$steps)  #get the steps vector filled with mean ave
steps <- as.data.frame(steps)          #now fill in the NAs in the k dataset
k<-data.frame(k[,c(1,3)],steps)
```

There are 2,304 day-interval combinations with a NA or missing reference that must be filled.

Plot the historgam of filled total steps per day. Get the mean and median number of steps each day from the adjusted data

```
l<-aggregate(k$steps, by=list(Category=k$date), FUN=sum)
hist(l$x)
```




```
m<-round(mean(l$x),0); n<-round(median(l$x),0)
m<-noquote(format(m, digits=9, big.mark=",")); n<-noquote(format(n, digits=9, big.mark=","))
```

The mean is 10,766 and the median is 10,762. The mean in particular is very difference from the un-filled values and the median is also higher. The impact is that we have substantially changed some of the inferences and outcomes that we might get around the daily steps.

However, this may be useful in computing daily average by day-of-the-week. The bias for these features may have reduced by increasing the sample.

Are there Differences in Patterns by Weekends and Weekdays?

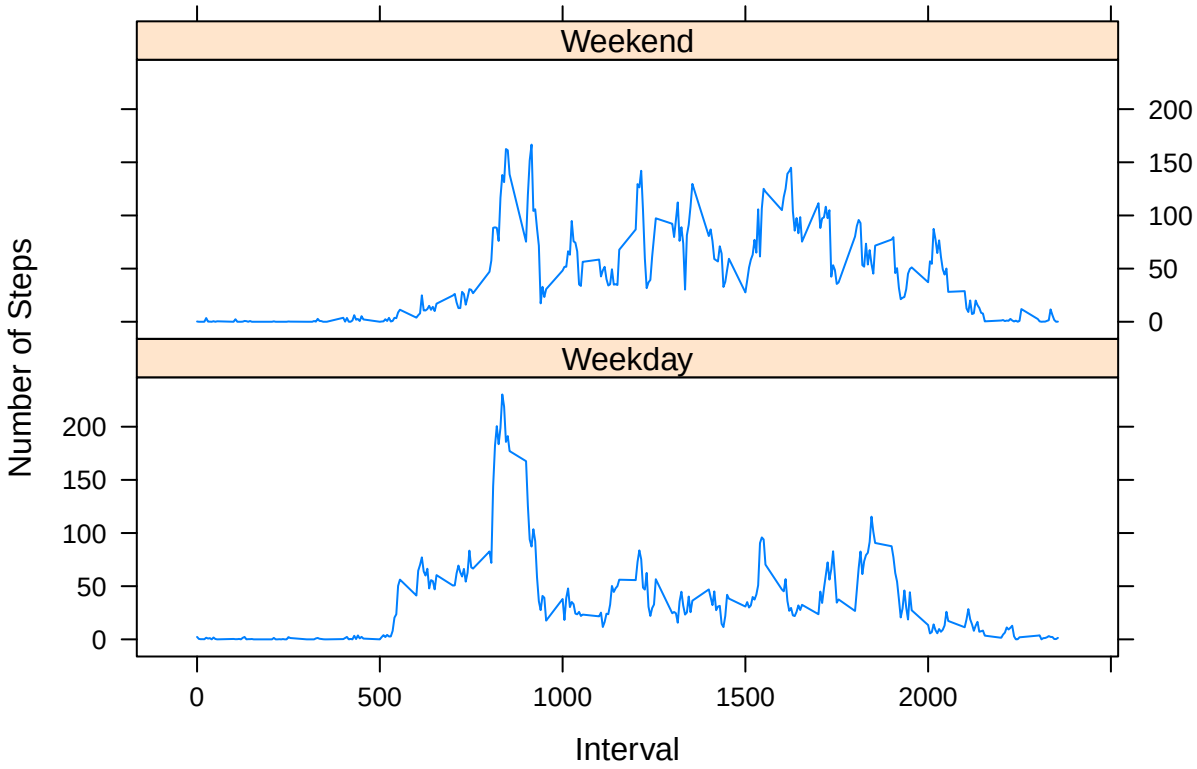
```
#add the day of the week to the set and the Weekday/Weekend indicator
k$wk <- weekdays(as.Date(k$date))
k$wk2[k$wk==c("Sunday")] <- "Weekend"
k$wk2[k$wk==c("Saturday")] <- "Weekend"
k$wk2[k$wk==c("Monday")] <- "Weekday"
k$wk2[k$wk==c("Tuesday")] <- "Weekday"
k$wk2[k$wk==c("Wednesday")] <- "Weekday"
k$wk2[k$wk==c("Thursday")] <- "Weekday"
k$wk2[k$wk==c("Friday")] <- "Weekday"

#get the average steps from filled data by weekend/weekday
p<-aggregate(k$steps, by=list(Category=k$wk2,k$interval), FUN=sum)
q<-data.frame(table(k$wk2,k$interval))
names(q)<-c("wk2","interval","days"); names(p)<-c("wk2","interval","steps")
```

```
r<-merge(x=p, y=q, by.x=c("wk2","interval") , by.y=c("wk2","interval"))
s<-data.frame(r[,c(1,2)],r$steps/r$days);names(s)<-c("wk2","interval","steps")
```

Now that we have the table, load a graphical package and plot the Weekday and Weekend average steps per interval

```
#now plot the two lines (Weekend vs Weekday) Timeseries
library(lattice)
s <- s[order(s[,1],s[,2]),]
xyplot(steps~interval| wk2, s,type="l",xlab="Interval",ylab="Number of Steps",layout=c(1,2),title="Avera
```



The observed data show a higher average footsteps on Weekday data earlier in the morning before 10am. Conversely, there are higher average footsteps in the afternoon (say from 12pm onwards) with the Weekend data.