Assignment 1

Zhong Qihui

2018.05.11

# Assignment 1

## Reading and processing the data

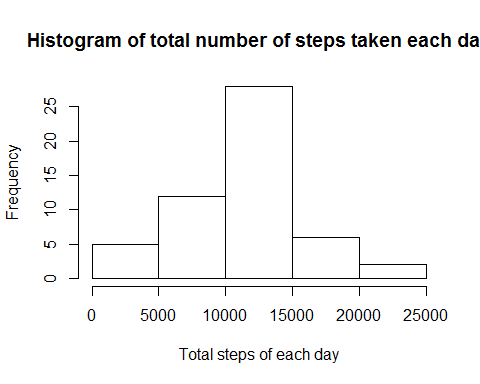
To analyze our data, first we should read the data.

activity <- read.csv('activity.csv')

## Plotting the histogram

So now we can draw the histogram with the processed dataset.

hist(with(activity,tapply(steps,date,sum)),xlab='Total steps of each day', main='Histogram of total number of steps taken each day')



We can conclude from the histogram that total numbers of total steps taken for more than 25 days fall between 10000 and 15000.

## Descriptive summary

Then let’s take a look at the mean and median number of steps taken each day.

summary(with(activity,tapply(steps,date,sum)))

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 41 8841 10765 10766 13294 21194 8

From the result, we can know that the average and median numbers of steps taken each day are 10,766 and 10,765 respectively.

## Time series plot

# Creating datetime variable

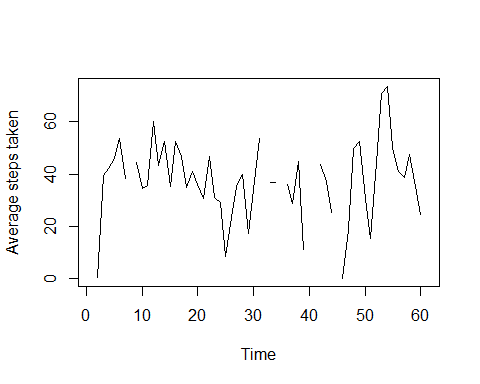
There are 2,400 intervals for each day, so 1 interval = 36 seconds. We should combine the date and interval variables to create a new datetime variable.

sec <- activity$interval \* 36  
day <- as.POSIXct(activity$date,tz='GMT')  
datetime <- day+sec

# Making the time series plot

With the new datetime variable, we can plot now.

plot(tapply(activity$steps,as.Date(datetime),mean),type='l',xlab='Time',ylab='Average steps taken')



From the above graph, we can easily find that there are missing values. They may affect our decisions. Therefore, we should exclude these missing values.

## Finding the specific interval

We need to find out the specific interval during which has the most steps taken.

datetime[grep(max(activity$steps,na.rm=TRUE),activity$steps)]

## [1] "2012-11-27 06:09:00 GMT"

The result shows that from 2012-11-27 06:06 to 2012-11-27 06:09, there are the most steps taken.

## Imputing missing data

As we can see from the raw data, there are missing values in the steps column. We should exclude the missing values so that we can continue with our research.

new\_activity <- na.exclude(activity)  
head(new\_activity)

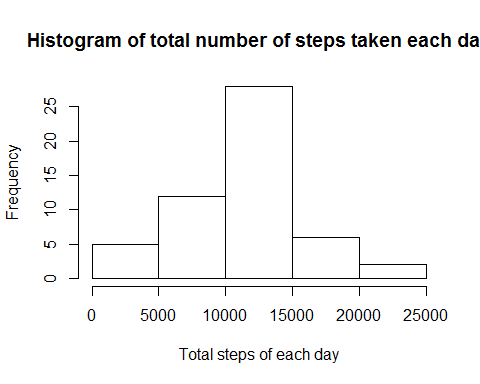
## steps date interval  
## 289 0 2012-10-02 0  
## 290 0 2012-10-02 5  
## 291 0 2012-10-02 10  
## 292 0 2012-10-02 15  
## 293 0 2012-10-02 20  
## 294 0 2012-10-02 25

Now we can see the data is clean now.

## Plotting the new histogram

With the new clean data, we can make a new histogram.

hist(with(new\_activity,tapply(steps,date,sum)),xlab='Total steps of each day', main='Histogram of total number of steps taken each day')



The new graph should look the same as the old one because missing values would not affect total number.

## Making the panel plot

### Grouping weekdays and weekends

Before plotting, we should figure out an important thing: which days are weekdays/weekends? It would be good if we define a new variable to group the dates. I come from China, so I need to set the locale first.

Sys.setlocale(category = "LC\_ALL", locale = "US")

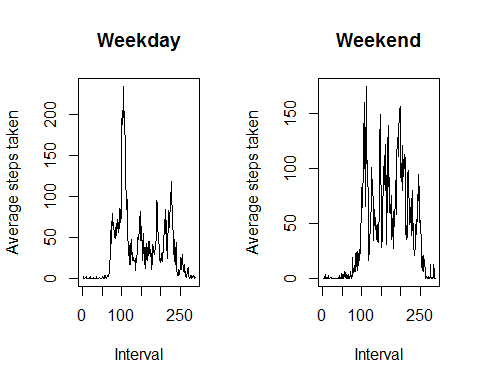
## [1] "LC\_COLLATE=English\_United States.1252;LC\_CTYPE=English\_United States.1252;LC\_MONETARY=English\_United States.1252;LC\_NUMERIC=C;LC\_TIME=English\_United States.1252"

sec1 <- new\_activity$interval \* 36  
day1 <- as.POSIXct(new\_activity$date,tz='GMT')  
datetime1 <- day1+sec1  
wkd <- weekdays(datetime1)  
wkd1 <- wkd  
wkd1[wkd1 == 'Monday'] <- 1  
wkd1[wkd1 == 'Tuesday'] <- 1  
wkd1[wkd1 == 'Wednesday'] <- 1  
wkd1[wkd1 == 'Thursday'] <- 1  
wkd1[wkd1 == 'Friday'] <- 1  
wkd1[wkd1 == 'Saturday'] <- 2  
wkd1[wkd1 == 'Sunday'] <- 2  
wkd2 <- factor(wkd1,c(1,2),labels=c('Weekday','Weekend'))  
new\_activity <- transform(new\_activity,wkd=wkd2)  
na1 <- subset(new\_activity,wkd=='Weekday')  
na2 <- subset(new\_activity,wkd=='Weekend')

### Panel plot

Now I have added the new variable to the data frame. I can plot now.

par(mfrow=c(1,2))  
with(na1,plot(tapply(steps,interval,mean),type='l',xlab='Interval',  
 ylab='Average steps taken'))  
title('Weekday')  
with(na2,plot(tapply(steps,interval,mean),type='l',xlab='Interval',  
 ylab='Average steps taken'))  
title('Weekend')



From the panel plot, we can see people take more steps on weekends.