Reproducible Research assignment - I.

A. D.

2017 february 24

Summary of the assignment

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.

The data for this assignment 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 datase

### Reading data from the working directory and creating the "activity" dataset:

activity <- read.csv("activity.csv", header = TRUE, sep = ",")

Checking the class of the variables

str(activity)

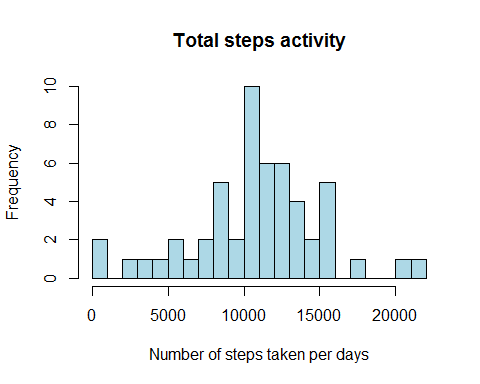
## 'data.frame': 17568 obs. of 3 variables:  
## $ steps : int NA NA NA NA NA NA NA NA NA NA ...  
## $ date : Factor w/ 61 levels "2012-10-01","2012-10-02",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ interval: int 0 5 10 15 20 25 30 35 40 45 ...

Transforming "date" variable to Date format

activity$date <- as.Date(activity$date)

### Histogram on the mean of the total number of steps taken per day.

steps\_day <- aggregate(activity$steps ~ activity$date, FUN = sum, na.rm = TRUE)  
  
hist(steps\_day$`activity$steps`, breaks = 20, col = "lightblue", main = "Total steps activity",   
 xlab = "Number of steps taken per days")



Mean and median of the total number of steps taken per day. As one can see, the date follows the structure of normal distribution

mean(steps\_day$`activity$steps`)

## [1] 10766.19

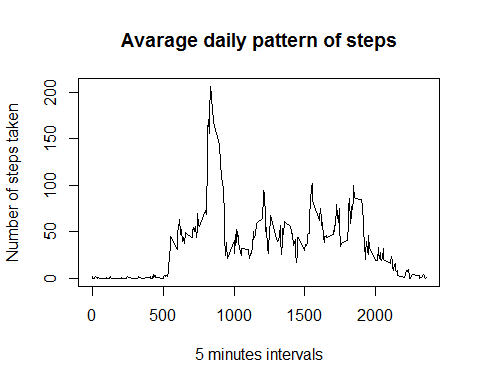
median(steps\_day$`activity$steps`)

## [1] 10765

### Average daily activity pattern

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)

steps\_int <- aggregate(activity$steps ~ activity$interval, FUN = mean, na.rm = TRUE)  
  
plot(steps\_int$`activity$interval`, steps\_int$`activity$steps`, type = "l",   
 main = "Avarage daily pattern of steps", xlab = "5 minutes intervals",   
 ylab = "Number of steps taken")



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

max\_steps <- subset(steps\_int, steps\_int$`activity$steps` == max(steps\_int$`activity$steps`))  
  
max\_steps$`activity$interval`

## [1] 835

### Total number of missing values in the dataset (i.e. the total number of rows with NAs):

sum(is.na(activity))

## [1] 2304

### Filling in all of the missing values in the dataset.

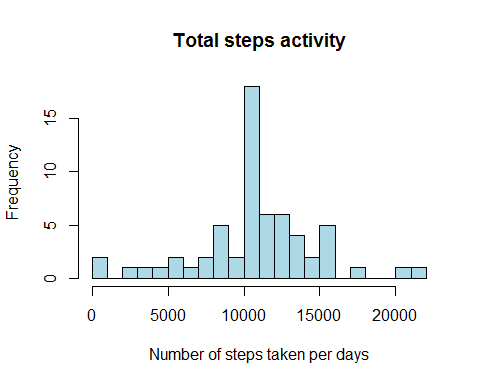
In the process of imputation I used the mean values from the 5 minutes intervals.

data\_2\_new <- data.frame()  
  
 for(j in 1:61){  
   
 data\_2\_new <- rbind(data\_2\_new, steps\_int)  
   
 }  
   
 for (i in 1:17568) {  
   
 if (is.na(activity$steps[i]) == TRUE) {  
   
 activity$steps[i] <- data\_2\_new[i ,2]  
   
 }  
 }  
   
 return(sum(is.na(activity)))

## [1] 0

### Histogram of the total number of steps taken each day and the mean and median total number of steps taken per day.

steps\_day\_clean <- aggregate(activity$steps ~ activity$date, FUN = sum, na.rm = TRUE)  
  
hist(steps\_day\_clean$`activity$steps`, breaks = 20, col = "lightblue",   
 main = "Total steps activity",   
 xlab = "Number of steps taken per days")



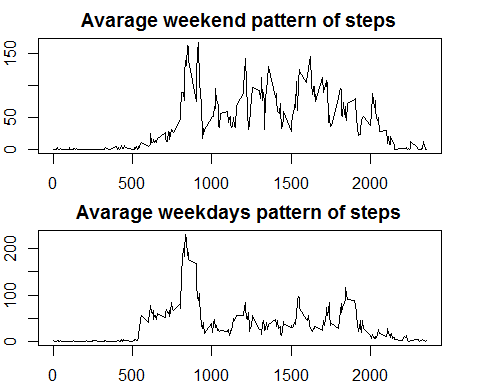
The new histogram shows the same pattern, but as one can see, the scale of the y axis changed (due to higher number of usable observations)

### Differences in activity patterns between weekdays and weekends

activity$date <- as.Date(activity$date)  
  
  
library(chron)

## Warning: package 'chron' was built under R version 3.3.2

activity$weekend <- is.weekend(activity$date)  
  
activity\_weekend <- subset(activity, activity$weekend == TRUE)  
activity\_weekdays <- subset(activity, activity$weekend == FALSE)  
  
  
steps\_int\_weekend <- aggregate(activity\_weekend$steps ~ activity\_weekend$interval, FUN = mean, na.rm = TRUE)  
  
steps\_int\_weekdays <- aggregate(activity\_weekdays$steps ~ activity\_weekdays$interval, FUN = mean, na.rm = TRUE)  
  
par(mfrow = c(2,1), mar = c(2, 2, 2, 2))  
  
plot(steps\_int\_weekend$`activity\_weekend$interval`, steps\_int\_weekend$`activity\_weekend$steps`, type = "l",   
 main = "Avarage weekend pattern of steps", xlab = "5 minutes intervals",   
 ylab = "Number of steps taken")  
  
  
  
  
plot(steps\_int\_weekdays$`activity\_weekdays$interval`, steps\_int\_weekdays$`activity\_weekdays$steps`,   
 type = "l",   
 main = "Avarage weekdays pattern of steps", xlab = "5 minutes intervals",   
 ylab = "Number of steps taken")



According to the graphs regarding the weekdays - weekend step numbers, we can observe higher activity during the weekend.