analyzing personal movement using activity monitoring devices

mirzarashid abbasov | reproducible research course - project 1 | coursera

18.OCT.2017

## Synopsis

There are many ways and possible to collect a large amount of data about personal movement using activity monitoring devices. 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.

This assignment makes use of data from a personal activity monitoring device. The data for this assignment can be downloaded from: [Dataset: Activity monitoring data](https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip)

The dataset variables included:

\* 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

## Library loading

suppressMessages(library(R.utils))

## Warning: package 'R.utils' was built under R version 3.4.2

## Warning: package 'R.oo' was built under R version 3.4.1

## Warning: package 'R.methodsS3' was built under R version 3.4.1

suppressMessages(library(dplyr))

## Warning: package 'dplyr' was built under R version 3.4.1

suppressMessages(library(ggplot2))

## Warning: package 'ggplot2' was built under R version 3.4.1

suppressMessages(library(gridExtra))

## Warning: package 'gridExtra' was built under R version 3.4.2

## Loading and preprocessing the data

1. Code for reading in the dataset and/or processing the data

knitr::opts\_chunk$set(echo = TRUE)  
  
# set working directory  
setwd("/Users/mirzarashid.abbasov/repos/Reproducible\_research/week2")  
  
# clean up workspance  
rm(list = ls())  
  
# set source url link  
url <- "https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip"  
  
# downloand data from url  
download.file(url, "activity.zip")  
  
# convert to the csv format  
if(!file.exists('activity.csv')){  
 unzip('activity.zip')  
}  
  
# read data from source files to the temp variable  
temp <- read.csv("activity.csv", header=T, sep=',')  
  
head(temp)

## 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

## Data processing & results

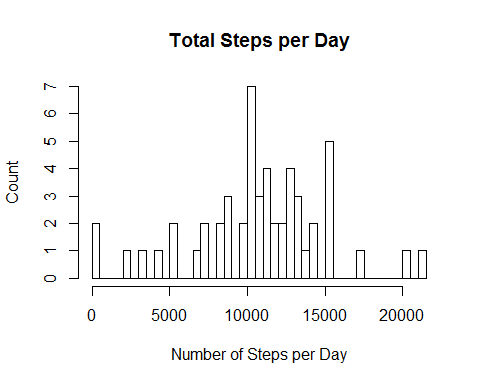
### What is mean total number of steps taken per day?

#### 2. Histogram of the total number of steps taken each day

knitr::opts\_chunk$set(echo = TRUE)  
  
##select steps & date from data.frame  
result.steps <- temp %>% select(steps, date) %>%   
 ## exclude all NA values from steps column  
 filter(!is.na(steps)) %>%  
 group\_by(date) %>%  
 ##calculate sum & mean & median grouped by date  
 summarise(total.steps = sum(steps))

## Warning: package 'bindrcpp' was built under R version 3.4.1

##histogram of count per day  
hist(result.steps$total.steps,   
 main="Total Steps per Day",   
 xlab="Number of Steps per Day",   
 ylab = "Count",  
 breaks=50)



## What is mean total number of steps taken per day?

### 3. Mean and median number of steps taken each day

## calculate mean of steps taken each day  
mean <- mean(result.steps$total.steps)  
mean

## [1] 10766.19

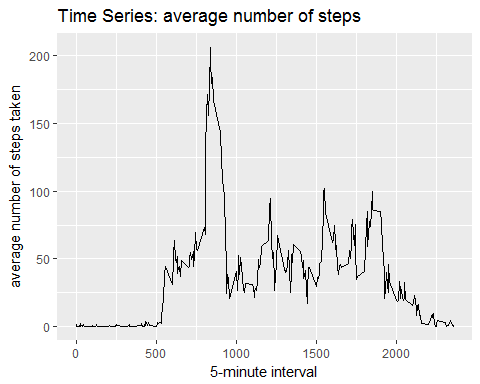
## calculate median of steps taken each day  
median <- median(result.steps$total.steps)  
median

## [1] 10765

## What is the average daily activity pattern?

### 4. Time series plot of the average number of steps taken

##select steps & date from data.frame  
result.mean <- temp %>% select(steps, interval) %>%   
 ## exclude all NA values from steps column  
 filter(!is.na(steps)) %>%  
 group\_by(interval) %>%  
 ##calculate sum & mean & median grouped by date  
 summarise(total.steps = mean(steps))  
  
##create a plot  
ggplot(data=result.mean,   
 aes(x = interval, y = total.steps)) +  
 geom\_line() +  
 ggtitle("Time Series: average number of steps") +  
 xlab("5-minute interval") +  
 ylab("average number of steps taken")



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

result.max <- temp %>% select(steps, interval) %>%   
 ## exclude all NA values from steps column  
 filter(!is.na(steps)) %>%  
 group\_by(interval) %>%  
 ##calculate sum & mean & median grouped by date  
 summarise(total.steps = sum(steps)) %>%  
 arrange(desc(total.steps)) %>% ##descending results  
 head(result.max, n=1L) ##only TOP 20  
  
## maximum interval number  
result.max$interval

## [1] 835

## maximum steps per interval  
result.max$total.steps

## [1] 10927

## Imputing missing values

### 6. Code to describe and show a strategy for imputing missing data

Use approximation method to fill missing data for example to use median() or mean()

## number of NA recods before  
before <- sum(is.na(temp$steps))  
before

## [1] 2304

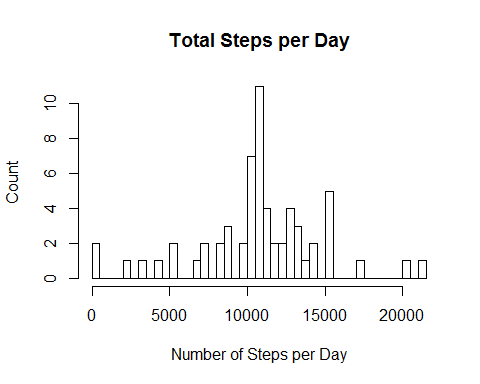
## to fill missing values with mean  
temp$steps[is.na(temp$steps)] <- mean(temp$steps[!is.na(temp$steps)])  
  
## number of NA recods after  
after <- sum(is.na(temp$steps))  
after

## [1] 0

## Are there differences in activity patterns between weekdays and weekends

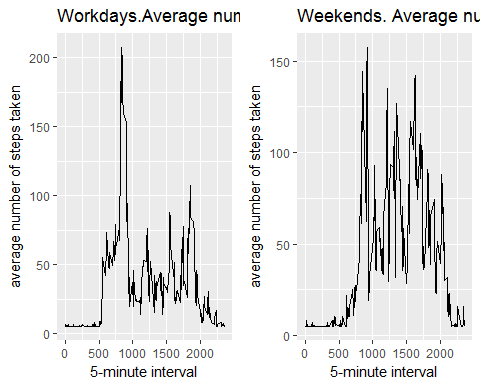
### 7. Histogram of the total number of steps taken each day after missing values are imputed

##select steps & date from data.frame  
result.steps <- temp %>% select(steps, date) %>%   
 ## exclude all NA values from steps column  
 filter(!is.na(steps)) %>%  
 group\_by(date) %>%  
 ##calculate sum & mean & median grouped by date  
 summarise(total.steps = sum(steps))  
   
##histogram of count per day  
hist(result.steps$total.steps,   
 main="Total Steps per Day",   
 xlab="Number of Steps per Day",   
 ylab = "Count",  
 breaks=50)



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

##weekdays as a decimal number(1-7, Monday is 1)  
  
##create a data.frame with weekends days  
temp.weekends <- temp[strftime(temp$date, format = "%u") > 5, ]  
  
##create a data.frame with workdays   
temp.workdays <- temp[strftime(temp$date, format = "%u") < 6, ]  
  
##select steps & date from data.frame  
result.weekends <- temp.weekends %>% select(steps, interval) %>%   
 ## exclude all NA values from steps column  
 filter(!is.na(steps)) %>%  
 group\_by(interval) %>%  
 ##calculate sum & mean & median grouped by date  
 summarise(total.steps = mean(steps))  
  
##select steps & date from data.frame  
result.workdays <- temp.workdays %>% select(steps, interval) %>%   
 ## exclude all NA values from steps column  
 filter(!is.na(steps)) %>%  
 group\_by(interval) %>%  
 ##calculate sum & mean & median grouped by date  
 summarise(total.steps = mean(steps))  
  
  
##create a plot for weekends  
g.weekends <- ggplot(data=result.weekends,   
 aes(x = interval, y = total.steps)) +  
 geom\_line() +  
 ggtitle("Weekends. Average number of steps") +  
 xlab("5-minute interval") +  
 ylab("average number of steps taken")  
  
##create a plot for workdays  
g.workdays <- ggplot(data=result.workdays,   
 aes(x = interval, y = total.steps)) +  
 geom\_line() +  
 ggtitle("Workdays.Average number of steps") +  
 xlab("5-minute interval") +  
 ylab("average number of steps taken")  
  
##create a final plot  
grid.arrange(g.workdays, g.weekends, ncol=2)



###### Mirzarashid Abbasov, almaty, 2017