Project 1 - knitr in data analysis

Reut Timor

October 30, 2017

The data for this assignment can be downloaded from the course web site:

Dataset: [Activity monitoring data] (<https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip>) [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 dataset.

turn warnings off

knitr::opts\_chunk$set(warning=FALSE)

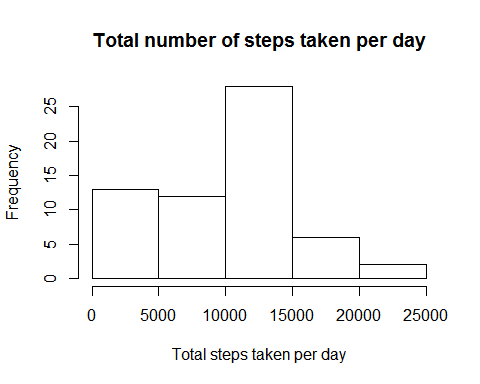
## 1. Code for reading in the dataset and preprocessing the data

library(ggplot2)  
library(data.table)  
url<-"https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip"  
download.file(url,"activity.zip")  
unzip("activity.zip")  
data\_activ <- read.csv("activity.csv")  
Sys.setenv(TZ="Europe/Berlin")  
data\_activ$date <- as.POSIXct(data\_activ$date, "%Y-%m-%d")  
weekday <- weekdays(data\_activ$date)  
data\_activ <- cbind(data\_activ,weekday)  
  
summary(data\_activ)

## steps date interval weekday   
## Min. : 0.00 Min. :2012-10-01 Min. : 0.0 Friday :2592   
## 1st Qu.: 0.00 1st Qu.:2012-10-16 1st Qu.: 588.8 Monday :2592   
## Median : 0.00 Median :2012-10-31 Median :1177.5 Saturday :2304   
## Mean : 37.38 Mean :2012-10-31 Mean :1177.5 Sunday :2304   
## 3rd Qu.: 12.00 3rd Qu.:2012-11-15 3rd Qu.:1766.2 Thursday :2592   
## Max. :806.00 Max. :2012-11-30 Max. :2355.0 Tuesday :2592   
## NA's :2304 Wednesday:2592

## 2. Code for creating a plot presenting the mean steps taken each day

sumsteps <- with(data\_activ, aggregate(steps,list(date),sum, na.rm = TRUE))  
names(sumsteps) <- c("date", "steps")  
hist(sumsteps$steps, main = "Total number of steps taken per day", xlab = "Total steps taken per day")

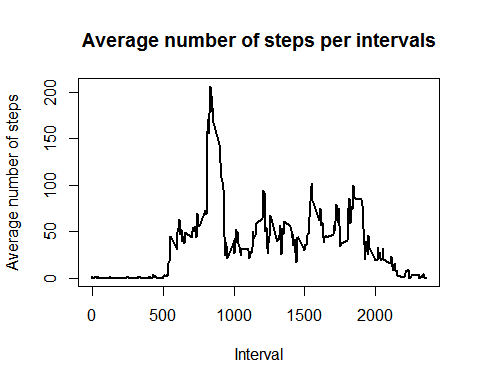


stepsmean<-mean(sumsteps$steps,na.rm = TRUE)  
stepsmedian<-median(sumsteps$steps)

1. The mean of all steps is 9354.2295082 The median is 10395

## 4. Code for tracking the time series plot of the average number of steps taken

average\_data\_activ <- with(data\_activ,aggregate(steps, list(interval), mean, na.rm=TRUE))  
names(average\_data\_activ) <- c("interval", "mean")  
plot(average\_data\_activ$interval, average\_data\_activ$mean, type = "l", lwd = 2, xlab="Interval", ylab="Average number of steps", main="Average number of steps per intervals")



## 5. The 5-minute interval, on average across all the days in the dataset, which contains the maximum number of steps is:

average\_data\_activ[which.max(average\_data\_activ$mean), ]$interval

## [1] 835

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

Since there are missing values in the data, filling those gaps should ease calculations on the data.

Na\_data<-is.na(data\_activ$steps)

## The number of missing values is 17568

One approch to fill missing values instead of ignoring them is to use the mean value of the 5 min interval

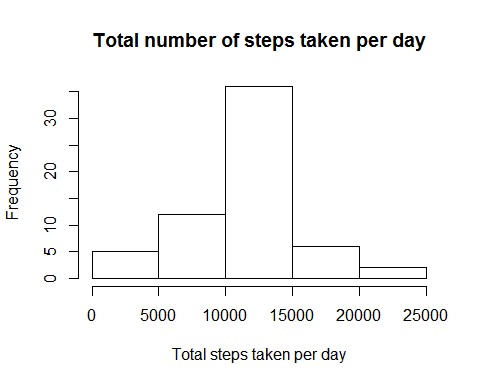
imputed\_steps <- average\_data\_activ$mean[match(data\_activ$interval, average\_data\_activ$interval)]

Now, lets create a new dataset with imputed values instead of the Na's

data\_activ\_imputed <- transform(data\_activ, steps = ifelse(is.na(data\_activ$steps), yes = imputed\_steps, no = data\_activ$steps))  
sumsteps\_imputed <- aggregate(steps ~ date, data\_activ\_imputed, sum)  
names(sumsteps\_imputed) <- c("date", "steps")  
means<-mean(sumsteps\_imputed$steps)  
medians<-median(sumsteps\_imputed$steps)

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

hist(sumsteps\_imputed$steps, main = "Total number of steps taken per day", xlab = "Total steps taken per day")



The mean of all steps after imputed values is 1.076618910^{4} The median of all steps after imputed values is 1.076618910^{4}

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

First, a new factor variable in the dataset with two levels - "weekday" and "weekend" will indicate if a given date is a weekday or weekend day.

data\_activ$date <- as.Date(strptime(data\_activ$date, format="%Y-%m-%d"))  
data\_activ$datetype <- sapply(data\_activ$date, function(x) {  
 if (weekdays(x) == "Saturday" | weekdays(x) =="Sunday")   
 {y <- "Weekend"} else   
 {y <- "Weekday"}  
 y  
 })

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).

data\_activ\_date <- with(data\_activ,aggregate(steps~interval + datetype, data\_activ, mean, na.rm = TRUE))  
plot<- ggplot(data\_activ\_date, aes(x = interval , y = steps, color = datetype)) +  
 geom\_line() +  
 labs(title = "Average daily steps by type of date", x = "Interval", y = "Average number of steps") +  
 facet\_wrap(~datetype, ncol = 1, nrow=2)  
print(plot)

