PA1\_template

Yerzhan Kadyrsizov

8 Р В Р’В Р вЂ™Р’В Р В РІР‚в„ўР вЂ™Р’В°Р В Р’В Р вЂ™Р’В Р В Р Р‹Р Р†Р вЂљРІР‚СњР В Р’В Р В Р вЂ№Р В Р’В Р Р†Р вЂљРЎв„ўР В Р’В Р вЂ™Р’В Р В РІР‚в„ўР вЂ™Р’ВµР В Р’В Р вЂ™Р’В Р В РІР‚в„ўР вЂ™Р’В»Р В Р’В Р В Р вЂ№Р В Р’В Р В Р РЏ 2017 Р В Р’В Р вЂ™Р’В Р В Р Р‹Р Р†Р вЂљРІР‚Сљ

# Course: Reproducible research

# Set up the working directory

## Loading and preprocessing the data

setwd("C:/Users/Dana/Desktop/Projects/coursera/Reproducible Research/Week2")  
  
# make sure the sources data folder exists  
if (!file.exists('source data')) {  
 dir.create('source data')  
}  
  
# check to see if the existing tidy data set exists; if not, make it...  
if (!file.exists('source data/repdata\_2Fdata2Factivity.zip')) {  
   
 # download the zip file and unzip  
 unzip('source data/repdata\_2Fdata\_2Factivity.zip',exdir='source data',overwrite=TRUE)  
   
 # Read Activity monitoring data ('data') data set from the working directory  
 data <- read.csv("./source data/activity.csv", stringsAsFactors = FALSE)  
} else {  
   
 # Read Activity monitoring data ('data') data set from the working directory  
 data <- read.csv("./source data/activity.csv", stringsAsFactors = FALSE)  
   
}

# Looking at a summary for the dataset using “summary” and “str” methods:

summary(data)

## steps date interval   
## Min. : 0.00 Length:17568 Min. : 0.0   
## 1st Qu.: 0.00 Class :character 1st Qu.: 588.8   
## Median : 0.00 Mode :character Median :1177.5   
## Mean : 37.38 Mean :1177.5   
## 3rd Qu.: 12.00 3rd Qu.:1766.2   
## Max. :806.00 Max. :2355.0   
## NA's :2304

str(data)

## 'data.frame': 17568 obs. of 3 variables:  
## $ steps : int NA NA NA NA NA NA NA NA NA NA ...  
## $ date : chr "2012-10-01" "2012-10-01" "2012-10-01" "2012-10-01" ...  
## $ interval: int 0 5 10 15 20 25 30 35 40 45 ...

# Looking at the first 6 rows of the dataset:

head(data)

## 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

# Converting the “date” variable to a Date classe and the “interval” variable to a factor:

data$date <- as.Date(data$date, format = "%Y-%m-%d")  
data$interval <- factor(data$interval)

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

# Subsitting the dataset to ignore missing values

NA\_index <- is.na(as.character(data$steps))  
data\_no\_NA <- data[!NA\_index,]  
head(data\_no\_NA)

## 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

# Aggregating the number of steps taken each day:

# Creating a data frame with the steps taken for each day

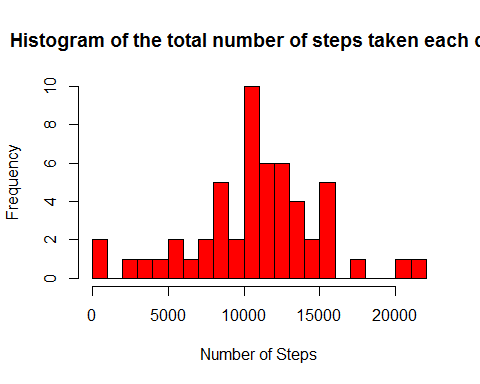
steps\_each\_day <- aggregate(steps ~ date, data = data\_no\_NA, sum)

# Adding column names to the created data frame

colnames(steps\_each\_day) <- c("date", "steps")

# Making a histogram of the total number of steps taken each day:

hist(as.numeric(steps\_each\_day$steps), breaks = 20, col = "red", xlab = "Number of Steps", main= "Histogram of the total number of steps taken each day")

 #number of steps taken per day: #Mean

mean(steps\_each\_day$steps)

## [1] 10766.19

# Median

median(steps\_each\_day$steps)

## [1] 10765

## What is the average daily activity pattern?

# Calculating the average

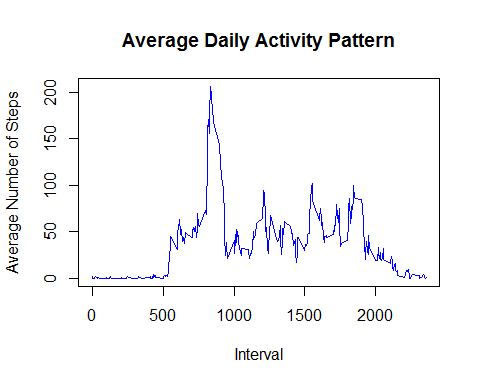
steps\_per\_interval <- aggregate(data\_no\_NA$steps, by=list(interval=data\_no\_NA$interval), FUN=mean)

# Adding columns names

colnames(steps\_per\_interval) <- c("interval", "average\_steps")

# ploting the average daily activity pattern

plot(as.integer(levels(steps\_per\_interval$interval)), steps\_per\_interval$average\_steps, type="l",  
 xlab = "Interval", ylab = "Average Number of Steps", main = "Average Daily Activity Pattern", col ="blue")

 #The maximum number of average steps

max\_steps <- max(steps\_per\_interval$average\_steps)  
max\_steps

## [1] 206.1698

## Imputing missing values

sum(is.na(as.character(data$steps)))

## [1] 2304

sum(is.na(as.character(data$date)))

## [1] 0

# finding the indices of missing values (NAs)

NA\_index <- which(is.na(as.character(data$steps)))  
complete\_data <- data

# Imputing missing values using the mean for that 5-minute interval

complete\_data[NA\_index, ]$steps<-unlist(lapply(NA\_index, FUN=function(NA\_index){  
 steps\_per\_interval[data[NA\_index,]$interval==steps\_per\_interval$interval,]$average\_steps  
 }))  
summary(complete\_data)

## steps date interval   
## Min. : 0.00 Min. :2012-10-01 0 : 61   
## 1st Qu.: 0.00 1st Qu.:2012-10-16 5 : 61   
## Median : 0.00 Median :2012-10-31 10 : 61   
## Mean : 37.38 Mean :2012-10-31 15 : 61   
## 3rd Qu.: 27.00 3rd Qu.:2012-11-15 20 : 61   
## Max. :806.00 Max. :2012-11-30 25 : 61   
## (Other):17202

str(complete\_data)

## 'data.frame': 17568 obs. of 3 variables:  
## $ steps : num 1.717 0.3396 0.1321 0.1509 0.0755 ...  
## $ date : Date, format: "2012-10-01" "2012-10-01" ...  
## $ interval: Factor w/ 288 levels "0","5","10","15",..: 1 2 3 4 5 6 7 8 9 10 ...

# 4 - Making a histogram of the total number of steps taken each day for the complete dataset:

# Creating a data frame with the steps taken for each day

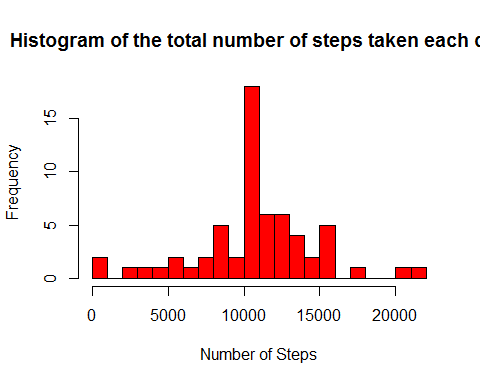
steps\_each\_day\_complete <- aggregate(steps ~ date, data = complete\_data, sum)

# Adding column names to the created data frame

colnames(steps\_each\_day\_complete) <- c("date", "steps")

# Making the histogram

hist(as.numeric(steps\_each\_day\_complete$steps), breaks = 20, col = "red", xlab = "Number of Steps", main= "Histogram of the total number of steps taken each day")



mean(steps\_each\_day\_complete$steps)

## [1] 10766.19

median(steps\_each\_day\_complete$steps)

## [1] 10766.19

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

# Creating a factor variable "day "to store the day of the week:

complete\_data$day <- as.factor(weekdays(complete\_data$date))

# Creating a logical variable "is\_weekday" (weekday=TRUE, weekend = FALE) :

complete\_data$is\_weekday <- ifelse(!(complete\_data$day %in% c("Saturday","Sunday")), TRUE, FALSE)

# Calculating the average number of steps for weekdays

weekdays\_data <- complete\_data[complete\_data$is\_weekday,]  
steps\_per\_interval\_weekdays <- aggregate(weekdays\_data$steps, by=list(interval=weekdays\_data$interval), FUN=mean)

# Calculating the average number of steps for weekends

weekends\_data <- complete\_data[!complete\_data$is\_weekday,]

# steps\_per\_interval\_weekends <- aggregate(weekends\_datainterval), FUN=mean)

# Adding columns names

colnames(steps\_per\_interval\_weekdays) <- c("interval", "average\_steps")

# colnames(steps\_per\_interval\_weekends) <- c("interval", "average\_steps")

# Adding a column to indecate the day

steps\_per\_interval\_weekdays$day <- "Weekday"

# steps\_per\_interval\_weekends$day <- "Weekend"

# Merging the two togather

week\_data <- steps\_per\_interval\_weekdays

# Converting the day variabke to a factor

week\_data$day <- as.factor(week\_data$day)

# Making the plot

library(lattice)  
xyplot(average\_steps ~ interval | day, data = week\_data, layout = c(1,2), type ="l", ylab="Number of Steps")

