Reproducible Reasearch: Project 1

sessionInfo()

## R version 3.4.1 (2017-06-30)  
## Platform: x86\_64-w64-mingw32/x64 (64-bit)  
## Running under: Windows 7 x64 (build 7601) Service Pack 1  
##   
## Matrix products: default  
##   
## locale:  
## [1] LC\_COLLATE=English\_United States.1252   
## [2] LC\_CTYPE=English\_United States.1252   
## [3] LC\_MONETARY=English\_United States.1252  
## [4] LC\_NUMERIC=C   
## [5] LC\_TIME=English\_United States.1252   
##   
## attached base packages:  
## [1] stats graphics grDevices utils datasets methods base   
##   
## loaded via a namespace (and not attached):  
## [1] compiler\_3.4.1 backports\_1.1.2 magrittr\_1.5 rprojroot\_1.3-2  
## [5] tools\_3.4.1 htmltools\_0.3.6 yaml\_2.1.18 Rcpp\_0.12.16   
## [9] stringi\_1.1.7 rmarkdown\_1.9 knitr\_1.20 stringr\_1.3.0   
## [13] digest\_0.6.15 evaluate\_0.10.1

## 1. Loading and preprocessing the data

1. Load the data (i.e. read.csv())
2. Process/transform the data (if necessary) into a format suitable for your analysis

knitr::opts\_chunk$set(echo = TRUE)  
file\_dir <- "C:/Google Drive/Coursera - Data Science/04 Reproducible Research/proj1"  
file\_name <- "activity.csv"  
   
setwd(file\_dir)  
library(data.table)

## Warning: package 'data.table' was built under R version 3.4.3

DT <- fread(input = file\_name, sep = ",")

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

1. Calculate the total number of steps taken per day
2. If you do not understand the difference between a histogram and a barplot, research the difference between them. Make a histogram of the total number of steps taken each day
3. Calculate and report the mean and median of the total number of steps taken per day

dim(DT)

## [1] 17568 3

names(DT)

## [1] "steps" "date" "interval"

str(DT)

## Classes 'data.table' and '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 ...  
## - attr(\*, ".internal.selfref")=<externalptr>

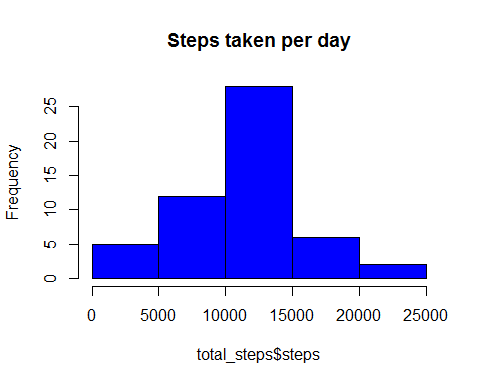
summary(DT)

## 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

total\_steps <- DT[, c(lapply(.SD, sum)),   
 .SDcols = c("steps"),   
 by = date]  
head(total\_steps) #1. Steps taken per day

## date steps  
## 1: 2012-10-01 NA  
## 2: 2012-10-02 126  
## 3: 2012-10-03 11352  
## 4: 2012-10-04 12116  
## 5: 2012-10-05 13294  
## 6: 2012-10-06 15420

hist(total\_steps$steps,   
 col = "blue",  
 main = "Steps taken per day") #2. Histogram



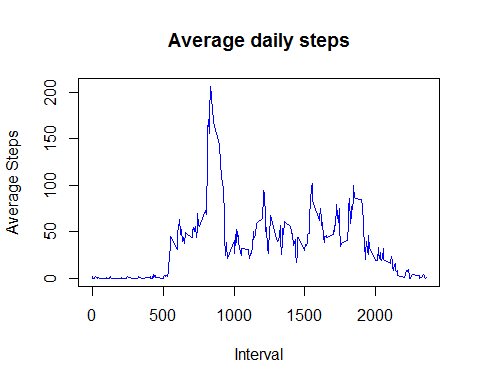
total\_steps[, .(mean\_steps = mean(steps, na.rm = TRUE),   
 median\_steps = median(steps, na.rm = TRUE))] #3. Mean and median steps per day

## mean\_steps median\_steps  
## 1: 10766.19 10765

## 3. What is the average daily activity pattern?

1. Make a time series plot (i.e. type=“1”) of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis)
2. Which 5-minute interval, on average across all the days in the dataset, contains the maximum number of steps?

ts\_DT <- DT[, c(lapply(.SD, mean, na.rm = TRUE)),   
 .SDcols = c("steps"),   
 by = interval]  
plot(ts\_DT,   
 type="l",   
 main="Average daily steps",   
 xlab="Interval",  
 ylab='Average Steps',   
 col = "blue") #1. Time series



ts\_DT[steps == max(steps), .(max\_interval = interval)] #2. Interval with max steps

## max\_interval  
## 1: 835

## 4. Imputing missing values

1. Calculate and report the total number of missing values in the dataset (i.e. the total number of rows with NAs)
2. Devise a strategy for filling in all of the missing values in the dataset. The strategy does not need to be sophisticated. For example, you could use the mean/median for that day, or the mean for that 5-minute interval, etc.
3. Create a new dataset that is equal to the original dataset but with the missing data filled in.
4. Make a histogram of the total number of steps taken each day and calculate and report the mean and median total number of steps taken per day. Do these values differ from the estimates from the first part of the assignment? What is the impact of imputing missing data on the estimates of the total daily number of steps?

DT[is.na(steps), .N] #1. Number of missing values

## [1] 2304

DT[is.na(steps), "steps"] <- DT[, c(lapply(.SD, median, na.rm = TRUE)), .SDcols = c("steps")] #2. Filling in with the median  
  
data.table::fwrite(x = DT, file = "activity\_filled.csv", quote = FALSE) #3. New dataset  
  
file\_name <- "activity\_filled.csv"  
DT2 <- fread(input = file\_name, sep = ",")  
dim(DT2)

## [1] 17568 3

names(DT2)

## [1] "steps" "date" "interval"

str(DT2)

## Classes 'data.table' and 'data.frame': 17568 obs. of 3 variables:  
## $ steps : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ 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 ...  
## - attr(\*, ".internal.selfref")=<externalptr>

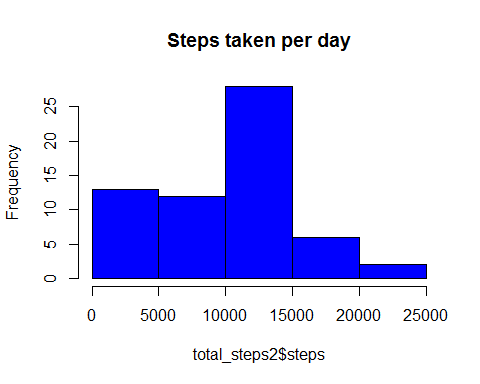
summary(DT2)

## 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 : 32.48 Mean :1177.5   
## 3rd Qu.: 0.00 3rd Qu.:1766.2   
## Max. :806.00 Max. :2355.0

total\_steps2 <- DT2[, c(lapply(.SD, sum)),   
 .SDcols = c("steps"),   
 by = date]  
head(total\_steps2) #4. Steps taken per day

## date steps  
## 1: 2012-10-01 0  
## 2: 2012-10-02 126  
## 3: 2012-10-03 11352  
## 4: 2012-10-04 12116  
## 5: 2012-10-05 13294  
## 6: 2012-10-06 15420

hist(total\_steps2$steps,   
 col = "blue",  
 main = "Steps taken per day") #4. Histogram



total\_steps2[, .(mean\_steps = mean(steps),   
 median\_steps = median(steps))] #4. Mean and median

## mean\_steps median\_steps  
## 1: 9354.23 10395

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

For this part the weekdays() function may be of some help here. Use the dataset with the filled-in missing values for this part. 1. Create a new factor variable in the dataset with two levels - “weekday” and “weekend” indicating whether a given date is a weekday or weekend day. 2. Make a panel plot containing a time series plot (i.e. type=“1”) of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all weekday days or weekend days (y-axis).

DT2[, date := as.POSIXct(date, format = "%Y-%m-%d")]  
DT2[, day := weekdays(x = date)]  
DT2[grepl(pattern = "Monday|Tuesday|Wednesday|Thursday|Friday", x = day), "type"] <- "weekday"  
DT2[grepl(pattern = "Saturday|Sunday", x = day), "type"] <- "weekend"   
head(DT2) #1. Create weekday/weekend factor

## steps date interval day type  
## 1: 0 2012-10-01 0 Monday weekday  
## 2: 0 2012-10-01 5 Monday weekday  
## 3: 0 2012-10-01 10 Monday weekday  
## 4: 0 2012-10-01 15 Monday weekday  
## 5: 0 2012-10-01 20 Monday weekday  
## 6: 0 2012-10-01 25 Monday weekday

wkday <- DT2[DT2$type == "weekday"]  
wkend <- DT2[DT2$type == "weekend"]  
  
ts\_wkday <- wkday[, c(lapply(.SD, mean, na.rm = TRUE)),   
 .SDcols = c("steps"),   
 by = interval]  
  
ts\_wkend <- wkend[, c(lapply(.SD, mean, na.rm = TRUE)),   
 .SDcols = c("steps"),   
 by = interval]  
  
par(mfrow = c(2, 1), mar = c(4, 4, 2, 1), oma = c(0, 0, 2, 0))  
  
plot(ts\_wkday,   
 type="l",   
 main="Average daily steps of weekdays",   
 xlab="Interval",  
 ylab='Average Steps',   
 col = "green") #1. Time series  
  
plot(ts\_wkend,   
 type="l",   
 main="Average daily steps of weekends",   
 xlab="Interval",  
 ylab='Average Steps',   
 col = "blue") #1. Time series

