# REPRODUCIBLE RESEARCH COURSE PROJECT 1

Loading and preprocessing the data

setwd("C:/Users/mahe/Documents/data")  
dataset <- read.csv("activity.csv")  
head(dataset)

## 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

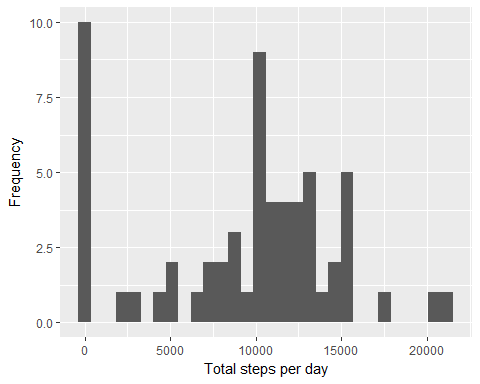
What is mean total number of steps taken per day?

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.4.4

stepsbyday <- tapply(dataset$steps, dataset$date, sum, na.rm=TRUE)  
plot1<- qplot(stepsbyday, xlab='Total steps per day', ylab='Frequency')  
print(plot1)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



Mean and median of the total number of steps taken per day

stepsByDayMean <- mean(stepsbyday)  
stepsByDayMedian <- median(stepsbyday)  
stepsByDayMean

## [1] 9354.23

stepsByDayMedian

## [1] 10395

What is the average daily activity pattern?

averageStepsPerTime <- aggregate(x=list(meanSteps=dataset$steps), by=list(interval=dataset$interval), FUN=mean, na.rm=TRUE)  
ggplot(data=averageStepsPerTime, aes(x=interval, y=meanSteps)) +  
 geom\_line() +  
 xlab("5-minute interval") +  
 ylab("average number of steps taken")

# 

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

mostSteps <- which.max(averageStepsPerTime$meanSteps)  
timeMostSteps <- gsub("([0-9]{1,2})([0-9]{2})", "\\1:\\2", averageStepsPerTime[mostSteps,'interval'])  
timeMostSteps

## [1] "8:35"

Imputing missing values Total number of missing values in the dataset

MissingValues <- length(which(is.na(dataset$steps)))  
MissingValues

## [1] 2304

A new dataset that is equal to the original dataset but with the missing data filled in

library(dplyr)

## Warning: package 'dplyr' was built under R version 3.4.3

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

replace\_with\_mean <- function(num) replace(num, is.na(num), mean(num, na.rm = TRUE))  
meanday <- (dataset %>% group\_by(interval) %>% mutate(steps = replace\_with\_mean(steps)))

## Warning: package 'bindrcpp' was built under R version 3.4.3

head(meanday)

## # A tibble: 6 x 3  
## # Groups: interval [6]  
## steps date interval  
## <dbl> <fctr> <int>  
## 1 1.7169811 2012-10-01 0  
## 2 0.3396226 2012-10-01 5  
## 3 0.1320755 2012-10-01 10  
## 4 0.1509434 2012-10-01 15  
## 5 0.0754717 2012-10-01 20  
## 6 2.0943396 2012-10-01 25

sum(is.na(meanday))

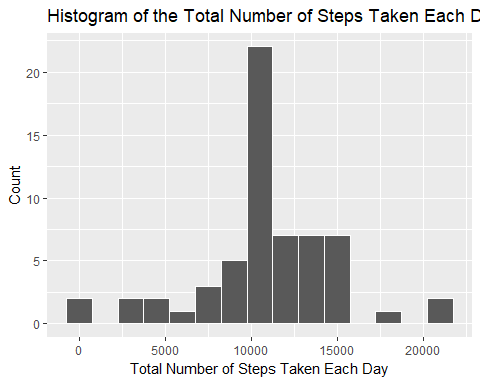
## [1] 0

new\_dataset <- as.data.frame(meanday)  
head(new\_dataset)

## steps date interval  
## 1 1.7169811 2012-10-01 0  
## 2 0.3396226 2012-10-01 5  
## 3 0.1320755 2012-10-01 10  
## 4 0.1509434 2012-10-01 15  
## 5 0.0754717 2012-10-01 20  
## 6 2.0943396 2012-10-01 25

Total number of steps taken each day without the missing values

new\_steps <- aggregate(new\_dataset$steps, by = list(new\_dataset$date), FUN = sum)  
names(new\_steps)[names(new\_steps) == "x"] <- "Total"  
names(new\_steps)[names(new\_steps) == "Group.1"] <- "Date"  
hist2 <- ggplot(data = new\_steps, aes(Total)) +   
 geom\_histogram(binwidth = 1500, colour = "white") +  
 xlab("Total Number of Steps Taken Each Day") +  
 ylab("Count") +  
 ggtitle("Histogram of the Total Number of Steps Taken Each Day with New Version Dataset")  
print(hist2)



library(grid)  
library(gridExtra)

## Warning: package 'gridExtra' was built under R version 3.4.4

##   
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':  
##   
## combine

grid.arrange(plot1, hist2, ncol = 2)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

# 

Mean and Median of the steps taken

mean(new\_steps$Total)

## [1] 10766.19

median(new\_steps$Total)

## [1] 10766.19

Are there differences in activity patterns between weekdays and weekends?

new\_dataset$WeekendOrWeekday <- ifelse(weekdays(as.Date(new\_dataset$date)) %in% c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday"), "Weekday", "Weekend")  
head(new\_dataset)

## steps date interval WeekendOrWeekday  
## 1 1.7169811 2012-10-01 0 Weekday  
## 2 0.3396226 2012-10-01 5 Weekday  
## 3 0.1320755 2012-10-01 10 Weekday  
## 4 0.1509434 2012-10-01 15 Weekday  
## 5 0.0754717 2012-10-01 20 Weekday  
## 6 2.0943396 2012-10-01 25 Weekday

new\_dataset <- (new\_dataset %>% group\_by(interval, WeekendOrWeekday) %>% summarise(Mean = mean(steps)))  
ggplot(new\_dataset, mapping = aes(x = interval, y = Mean)) + geom\_line() +  
 facet\_grid(WeekendOrWeekday ~.) + xlab("Interval") + ylab("Mean of Steps") +  
 ggtitle("Comparison of Average Number of Steps in Each Interval")

