# Reproducible Research - Peer Assessment 1

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.2.5

library(scales)

## Warning: package 'scales' was built under R version 3.2.5

library(Hmisc)

## Warning: package 'Hmisc' was built under R version 3.2.5

## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

##   
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:base':  
##   
## format.pval, round.POSIXt, trunc.POSIXt, units

### Load the data (i.e. read.csv())

activityData <- read.csv('activity.csv')

### Calculate the total number of steps taken per day

stepsByDay <- tapply(activityData$steps, activityData$date, sum, na.rm=TRUE)

### Make a histogram of the total number of steps taken everyday

qplot(stepsByDay, xlab='Total steps per day', ylab='Frequency using binwith 500', binwidth=500)



### Calculate and report the mean and median of the total number of steps taken per day

print(paste("Mean: ",mean(stepsByDay)))

## [1] "Mean: 9354.22950819672"

print(paste("Median: ",median(stepsByDay)))

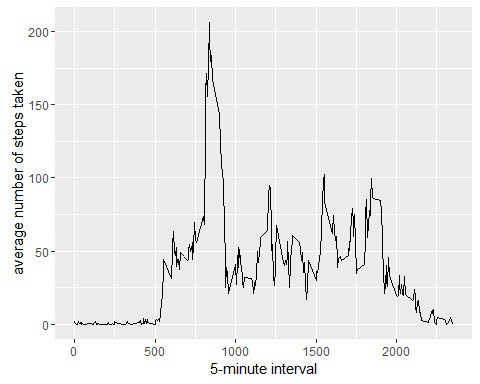
## [1] "Median: 10395"

#### Average Daily Activity Pattern

averageStepsPerTimeBlock <- aggregate(x=list(meanSteps=activityData$steps), by=list(interval=activityData$interval), FUN=mean, na.rm=TRUE)

### Time Series Plot

ggplot(data=averageStepsPerTimeBlock, aes(x=interval, y=meanSteps)) +  
 geom\_line() +  
 xlab("5-minute interval") +  
 ylab("average number of steps taken")



### Maximum number of steps

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

## [1] "Most Steps: 8:35"

### Calculate and report the total number of missing values in the dataset

numMissingValues <- length(which(is.na(activityData$steps)))

## \* Number of missing values: r numMissingValues

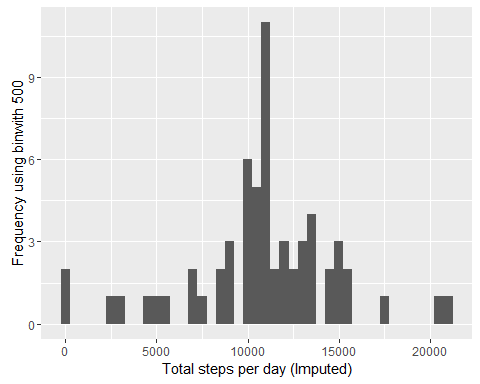
### Devise a strategy for filling in all of the missing values in the dataset.

### Create a new dataset that is equal to the original dataset but with the missing data filled in.

activityDataImputed <- activityData  
activityDataImputed$steps <- impute(activityData$steps, fun=mean)

### Make a histogram of the total number of steps taken each day

stepsByDayImputed <- tapply(activityDataImputed$steps, activityDataImputed$date, sum)  
qplot(stepsByDayImputed, xlab='Total steps per day (Imputed)', ylab='Frequency using binwith 500', binwidth=500)



### Calculate and report the mean and median total number of steps taken per day.

print(paste("Mean: ", mean(stepsByDayImputed)))

## [1] "Mean: 10766.1886792453"

print(paste("Median: ", median(stepsByDayImputed)))

## [1] "Median: 10766.1886792453"

### 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.

activityDataImputed$dateType <- ifelse(as.POSIXlt(activityDataImputed$date)$wday %in% c(0,6), 'weekend', 'weekday')

### Make a panel plot containing a time series plot

averagedActivityDataImputed <- aggregate(steps ~ interval + dateType, data=activityDataImputed, mean)  
ggplot(averagedActivityDataImputed, aes(interval, steps)) +   
 geom\_line() +   
 facet\_grid(dateType ~ .) +  
 xlab("5-minute interval") +   
 ylab("avarage number of steps")

