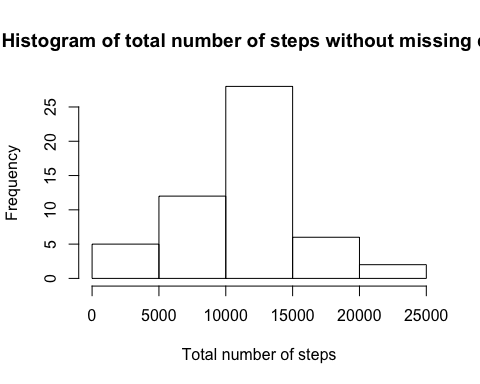
Reproducible Research: Peer Assessment 1

## Loading and preprocessing the data

#Load the data  
data<-read.csv("activity.csv",header=T,sep=",",na.strings="NA")  
#get subset without missing values  
dataNoNA<-na.omit(data) #remove rows with NA data

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

#total number of steps taken per day  
totalSteps<-aggregate(steps~date,FUN=sum,data=dataNoNA)   
#histogram of the total number of steps taken each day  
hist(totalSteps$steps,xlab="Total number of steps",  
 main="Histogram of total number of steps without missing data")



#mean of the total number of steps taken per day  
meanSteps <- mean(totalSteps[,"steps"]) #store to use for imputing  
meanSteps

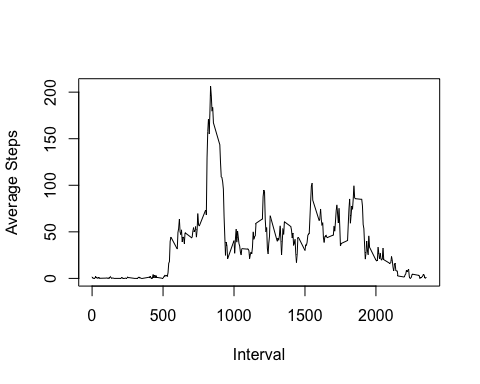
## [1] 10766.19

#median of the total number of steps taken per day  
median(totalSteps[,"steps"])

## [1] 10765

## What is the average daily activity pattern?

#plot of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis)  
#get total number of days with data  
numDays <-nrow(totalSteps)  
#average number of steps per day taken per interval  
avgStepsPerInt<-aggregate(steps~interval,FUN=sum,data=dataNoNA)   
avgStepsPerInt$steps <- avgStepsPerInt$steps/numDays  
#plot interval (x-axis) vs average number of steps (y-axis)  
plot(avgStepsPerInt$interval,avgStepsPerInt$steps,type="l",  
 xlab="Interval",ylab="Average Steps")



#5-minute interval, on average across all the days in the dataset,   
#contains the maximum number of steps  
avgStepsPerInt[which.max(avgStepsPerInt[,"steps"]),"interval"]

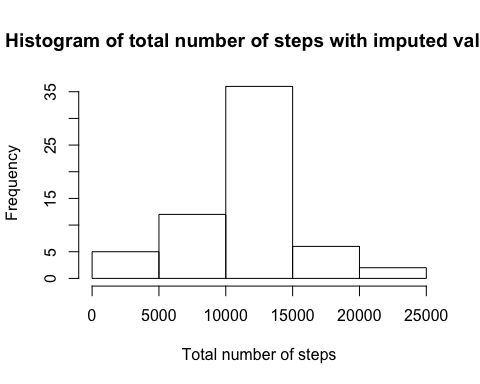
## [1] 835

## Imputing missing values

#total number of missing values in the dataset  
sum(is.na(data))

## [1] 2304

#create new dataset for imputed data  
imputed <- data  
#fill in all of the missing values in the dataset with mean for each interval in a day  
imputed$steps[is.na(imputed$steps)] <- meanSteps/288  
#total number of steps taken per day with imputed data  
totalSteps<-aggregate(steps~date,FUN=sum,data=imputed)  
#histogram of total number of steps taken each day with imputed data  
hist(totalSteps$steps,xlab="Total number of steps",  
 main="Histogram of total number of steps with imputed values")



#calculate mean and median with dataset with imputed values  
imputedTotal<-aggregate(steps~date,FUN=sum,data=imputed) #total steps taken per day with imputed data  
mean(imputedTotal[,"steps"])

## [1] 10766.19

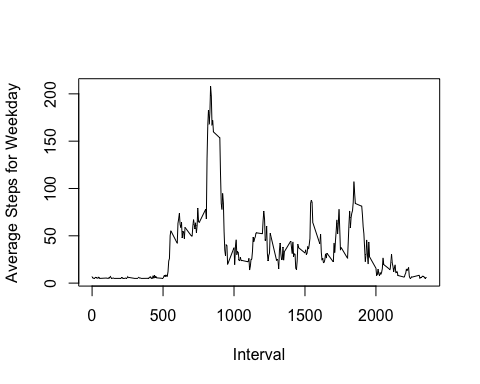
median(imputedTotal[,"steps"])

## [1] 10766.19

Imputing missing values with the mean keeps the mean same. If there are enough missing values, the median changes to reflect that. In our case since the number of missing values was large, the median became equal to the mean after imputation.

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

#new column in the dataset with two levels – “weekday” and “weekend”  
imputed$date <- as.Date(imputed$date)  
weekend <- c('Saturday','Sunday')  
imputed$dayType <- ifelse(weekdays(imputed$date) %in% weekend,"weekend","weekday")  
  
#panel plot containing a time series plot  
#of the 5-minute interval (x-axis) and the average number of steps taken,   
#averaged across all weekday days or weekend days (y-axis)  
#par(mfrow=c(2,1))   
weekdayData <- imputed[imputed$dayType=="weekday",]  
numWeekdayDays <- nrow(weekdayData)/288  
avgStepsPerInt<-aggregate(steps~interval,FUN=sum,data=weekdayData)   
avgStepsPerInt$steps <- avgStepsPerInt$steps/numWeekdayDays  
#plot interval (x-axis) vs average number of steps (y-axis) for weekdays  
plot(avgStepsPerInt$interval,avgStepsPerInt$steps,type="l",  
 xlab="Interval",ylab="Average Steps for Weekday")



weekendData <- imputed[imputed$dayType=="weekend",]  
numWeekendDays <- nrow(weekendData)/288  
avgStepsPerInt<-aggregate(steps~interval,FUN=sum,data=weekendData)  
avgStepsPerInt$steps <- avgStepsPerInt$steps/numWeekendDays  
#plot interval (x-axis) vs average number of steps (y-axis) for weekends  
plot(avgStepsPerInt$interval,avgStepsPerInt$steps,type="l",  
 xlab="Interval",ylab="Average Steps for Weekend")

