Opening up the directory on my computer for this file:

setwd("C:/Users/Manuel/Desktop/Coursera")  
datafile <- read.csv("./repdata/activity.csv")  
  
head(datafile)

## 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

summary(datafile)

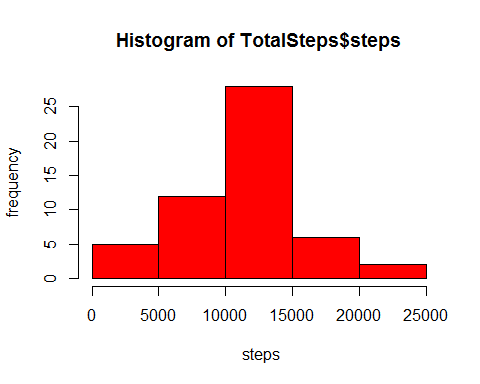
## steps date interval   
## Min. : 0.00 2012-10-01: 288 Min. : 0.0   
## 1st Qu.: 0.00 2012-10-02: 288 1st Qu.: 588.8   
## Median : 0.00 2012-10-03: 288 Median :1177.5   
## Mean : 37.38 2012-10-04: 288 Mean :1177.5   
## 3rd Qu.: 12.00 2012-10-05: 288 3rd Qu.:1766.2   
## Max. :806.00 2012-10-06: 288 Max. :2355.0   
## NA's :2304 (Other) :15840

What is the mean of steps taken per day?

#Getting the mean  
#Fist: Add the number of steps per day  
TotalSteps <- aggregate(steps ~ date, datafile, FUN = sum)  
summary(TotalSteps)

## date steps   
## 2012-10-02: 1 Min. : 41   
## 2012-10-03: 1 1st Qu.: 8841   
## 2012-10-04: 1 Median :10765   
## 2012-10-05: 1 Mean :10766   
## 2012-10-06: 1 3rd Qu.:13294   
## 2012-10-07: 1 Max. :21194   
## (Other) :47

#Make the histogram:  
hist(TotalSteps$steps, xlab = " steps", ylab = "frequency", col = "red")



#Getting the mean:  
  
StepsMean<- mean(TotalSteps$steps, na.rm = "TRUE")  
StepsMedian <- median(TotalSteps$steps, na.rm = "TRUE")  
  
#Printing the values for mean and median:  
StepsMean

## [1] 10766.19

StepsMedian

## [1] 10765

Next: Making a time series plot

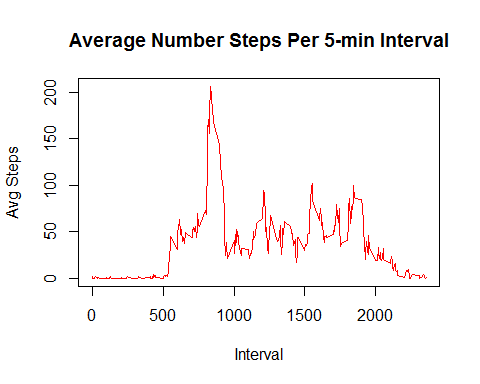
#Getting the average number of steps in a 5min interval:  
AvgSteps <- aggregate(steps ~ interval , datafile , FUN = mean)  
summary(AvgSteps)

## interval steps   
## Min. : 0.0 Min. : 0.000   
## 1st Qu.: 588.8 1st Qu.: 2.486   
## Median :1177.5 Median : 34.113   
## Mean :1177.5 Mean : 37.383   
## 3rd Qu.:1766.2 3rd Qu.: 52.835   
## Max. :2355.0 Max. :206.170

head(AvgSteps)

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

#Making the plot:  
plot(AvgSteps$interval, AvgSteps$steps, type = "l", col = "red", xlab = "Interval" , ylab = "Avg Steps", main = "Average Number Steps Per 5-min Interval")



#Finding which 5 min interval has the most number of steps taken:  
MostSteps <- which.max(AvgSteps$steps)  
MostSteps #This returns the number of the list, which is interval 835 , which has ~209 steps

## [1] 104

#A more refined approach to getting the interval with the most steps is below:  
maxInt <- AvgSteps[which.max(AvgSteps$steps),]  
maxInt

## interval steps  
## 104 835 206.1698

Finding and Replacing the Missing Data

#Total number of missing values is:   
sum(is.na(datafile[,1]))

## [1] 2304

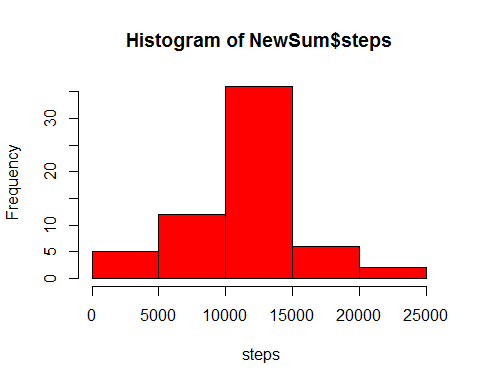
#Replacing the missing data with Average in 5 min interval  
missing <- !complete.cases(datafile)  
REMNA <- datafile[missing == TRUE,]  
REMNA[,1] <- AvgSteps$steps  
  
summary(REMNA) #Shows no NA values

## steps date interval   
## Min. : 0.000 2012-10-01:288 Min. : 0.0   
## 1st Qu.: 2.486 2012-10-08:288 1st Qu.: 588.8   
## Median : 34.113 2012-11-01:288 Median :1177.5   
## Mean : 37.383 2012-11-04:288 Mean :1177.5   
## 3rd Qu.: 52.835 2012-11-09:288 3rd Qu.:1766.2   
## Max. :206.170 2012-11-10:288 Max. :2355.0   
## (Other) :576

#Making a newFile with replaced data  
datafile1 <- rbind(datafile[complete.cases(datafile),], REMNA)  
  
#Getting the New Average with NA values filled in  
NewSum <- aggregate(steps ~ date, datafile1, FUN = sum, na.rm = TRUE)  
  
summary(NewSum)

## date steps   
## 2012-10-01: 1 Min. : 41   
## 2012-10-02: 1 1st Qu.: 9819   
## 2012-10-03: 1 Median :10766   
## 2012-10-04: 1 Mean :10766   
## 2012-10-05: 1 3rd Qu.:12811   
## 2012-10-06: 1 Max. :21194   
## (Other) :55

#Making a histogram of total number of steps in new file  
hist(NewSum$steps, xlab = " steps", col = "red")



Capturing the Diferrence between Weekend and Weekday activity

#labeling Weekdays and weekends appropropiately  
DayLabel <- function(date) {  
 day <- weekdays(date)  
 if (day %in% c('Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday'))  
 return ("weekeday")  
 else if (day %in% c('Saturday', 'Sunday'))  
 return ("weekend")  
 else  
 stop ("Invalid Date Format.")  
}  
  
datafile1$date <- as.Date(datafile1$date)  
datafile1$day <- sapply(datafile1$date, FUN = DayLabel)

Plotting the Data

LabelMean <- aggregate(steps ~ interval + day, datafile1, mean)  
  
head(LabelMean)

## interval day steps  
## 1 0 weekeday 2.25115304  
## 2 5 weekeday 0.44528302  
## 3 10 weekeday 0.17316562  
## 4 15 weekeday 0.19790356  
## 5 20 weekeday 0.09895178  
## 6 25 weekeday 1.59035639

#Plotting the Data  
library(lattice)  
xyplot(steps ~ interval | day, data = LabelMean, type = "l", xlab = "Interval",   
 ylab = "Number of steps", layout = c(1, 2))

