Reproducible Result Assignemnt1

Abhinav

19 May 2017

## Set up the environment

# 1. Read data

DF<- read.csv("activity.csv")  
 str(DF)

## 'data.frame': 17568 obs. of 3 variables:  
## $ steps : int NA NA NA NA NA NA NA NA NA NA ...  
## $ date : Factor w/ 61 levels "2012-10-01","2012-10-02",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ interval: int 0 5 10 15 20 25 30 35 40 45 ...

head(DF)

## 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

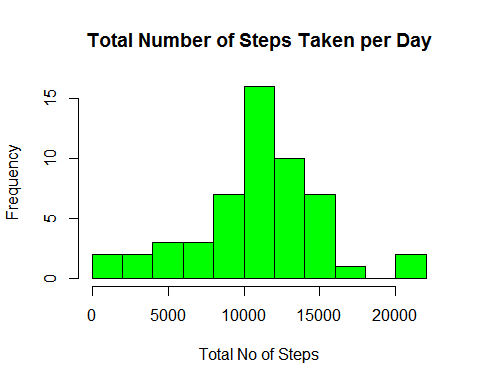
## Analyze the data

Steps\_Data <- summarise(group\_by(DF, date), Total\_Daily\_Steps=sum(steps) ,n())  
 head(Steps\_Data)

## # A tibble: 6 x 3  
## date Total\_Daily\_Steps `n()`  
## <fctr> <int> <int>  
## 1 2012-10-01 NA 288  
## 2 2012-10-02 126 288  
## 3 2012-10-03 11352 288  
## 4 2012-10-04 12116 288  
## 5 2012-10-05 13294 288  
## 6 2012-10-06 15420 288

# 2. Histogram- Number of steps per day

x <-as.numeric(na.omit(Steps\_Data$Total\_Daily\_Steps))  
 hist(x, breaks=10, main= "Total Number of Steps Taken per Day",col="green", xlab= "Total No of Steps",axes= TRUE)



# 3. Mean, Median No of Steps

mean(Steps\_Data$Total\_Daily\_Steps,na.rm= TRUE)

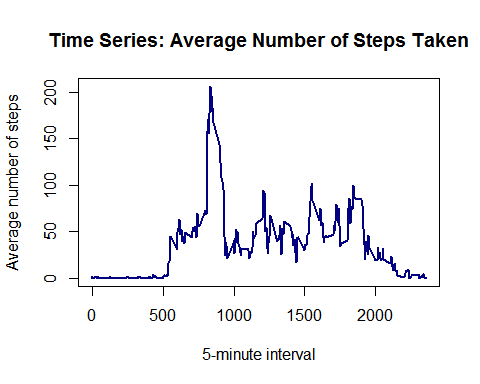
## [1] 10766.19

median(Steps\_Data$Total\_Daily\_Steps,na.rm= TRUE)

## [1] 10765

# 4. Average Daily Activity Pattern- Average number of steps taken

avg\_step <- aggregate(steps ~ interval, data = DF, mean, na.rm = TRUE)  
plot(avg\_step$interval, avg\_step$steps, type = "l", lwd = 2, col = "navy",  
 main = "Time Series: Average Number of Steps Taken", axes = TRUE,  
 xlab = "5-minute interval", ylab = "Average number of steps")



# 5. Which 5 Min interval contains maximum number of steps?

avg\_step$interval[which.max(avg\_step$steps)]

## [1] 835

# Imputing missing values

## Total number of missing values

sum(is.na(DF))

## [1] 2304

# 6. Strategy for filling in null values -

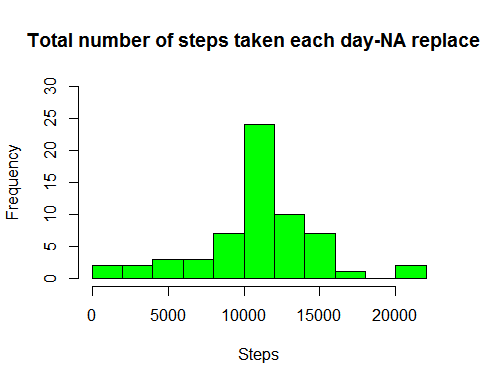
## Replace missing data with mean of that column, use mean of 5 min interval corresponding to missing value. Create a new dataset with missing values filled in

New\_DF <- DF  
  
for(i in 1:nrow(DF))   
 {  
 if (is.na(New\_DF[i,1]))  
 {  
 int\_5= New\_DF[i,3]  
 New\_DF[i,1] <- mean(New\_DF$steps[New\_DF$interval== int\_5],na.rm=TRUE)  
 }  
 }  
 head(New\_DF)

## 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

# 7. Histogram of the total number of steps taken each day and report the mean and median total number of steps taken per day.

New\_x <- summarise (group\_by(New\_DF, date), Total\_Daily\_Steps=sum(steps) ,n())  
   
 a <- as.numeric(New\_x$Total\_Daily\_Steps)  
 hist(a,breaks=10, main='Total number of steps taken each day-NA replaced', xlab='Steps', ylim=c(0,30),axes= TRUE, col= "green")



## Mean, Median No of Steps

mean(New\_x$Total\_Daily\_Steps,na.rm= TRUE)

## [1] 10766.19

median(New\_x$Total\_Daily\_Steps,na.rm= TRUE)

## [1] 10766.19

## New factor variable in the dataset with two levels - "weekdays" and "weekend" indicating whether a given date is a weekday or weekend day.

New\_DF$Day\_Status <- ifelse(weekdays(as.Date(New\_DF$date)) %in% c("Saturday", "Sunday"), "weekend", "weekday")

# 8.Time series plot (i.e. type = "l") of the 5- minute interval (x-axis) and the average number of steps taken, averaged across all weekday days or weekend days (y-axis).

avg\_step <- aggregate(steps ~ interval + Day\_Status, data = New\_DF, mean, na.rm = TRUE)  
  
xyplot(steps ~ interval| Day\_Status, avg\_step,   
 type="l",   
 lwd=2,   
 xlab="Interval",   
 ylab="Number of steps",   
 layout=c(1,2))

