# Reproducible Research: Peer Assessment 1

## Loading and preprocessing the data

```
activity <- read.csv('activity.csv', na.strings = 'NA')
activity$date <- as.Date(activity$date, '%Y-%m-%d')
str(activity)
```

```
## 'data.frame':    17568 obs. of  3 variables:
##  $ steps   : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ date    : Date, format: "2012-10-01" "2012-10-01" ...
##  $ interval: int  0 5 10 15 20 25 30 35 40 45 ...
```

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

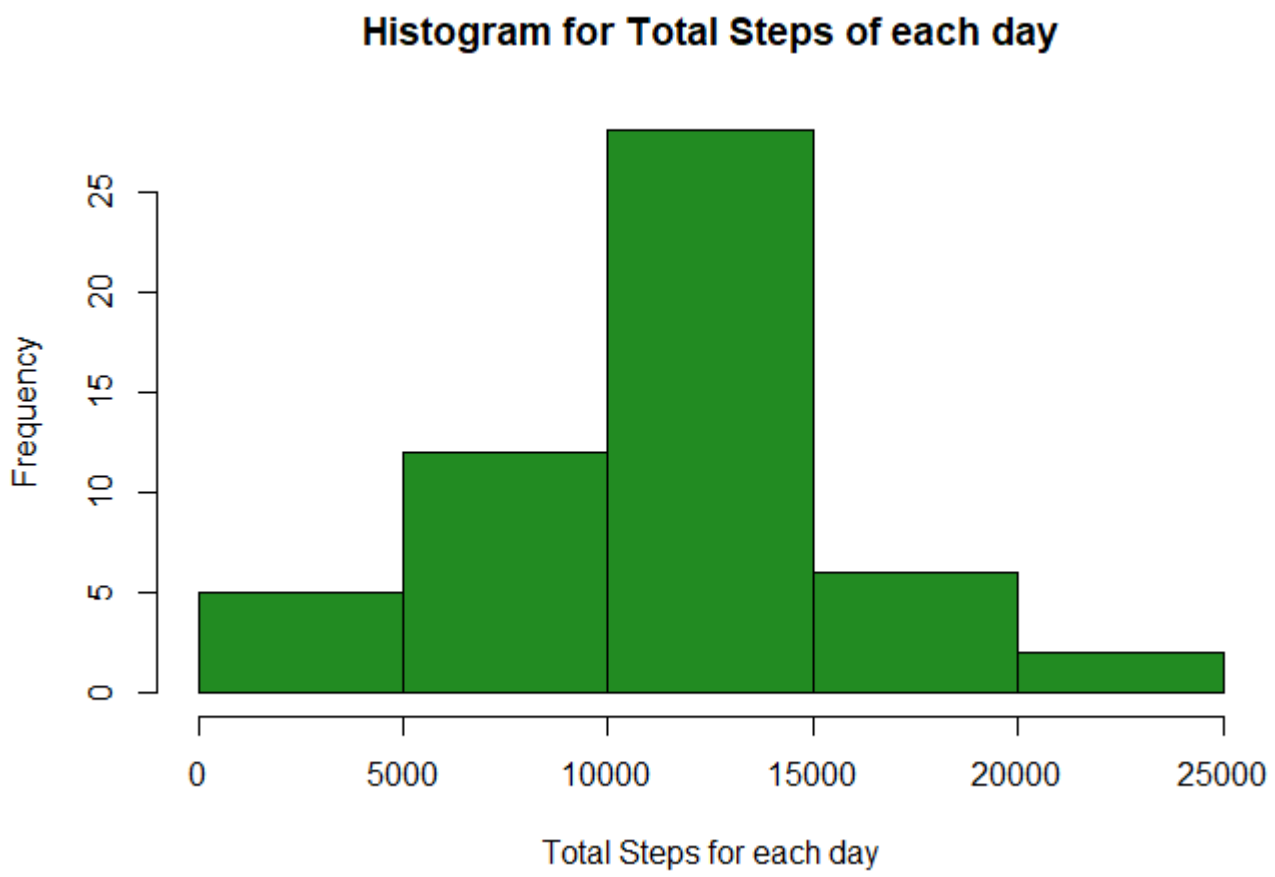
```
library(data.table)
activity <- data.table(activity)
activity.clean <- activity[complete.cases(activity),]
dailytotal <- activity.clean[, .(TotalSteps = sum(steps, na.rm = TRUE)), by=date]
cat('mean total number of steps taken per day:', mean(dailytotal$TotalSteps), '\n')
```

```
## mean total number of steps taken per day: 10766.19
```

```
cat('median total number of steps taken per day:', median(dailytotal$TotalSteps))
```

```
## median total number of steps taken per day: 10765
```

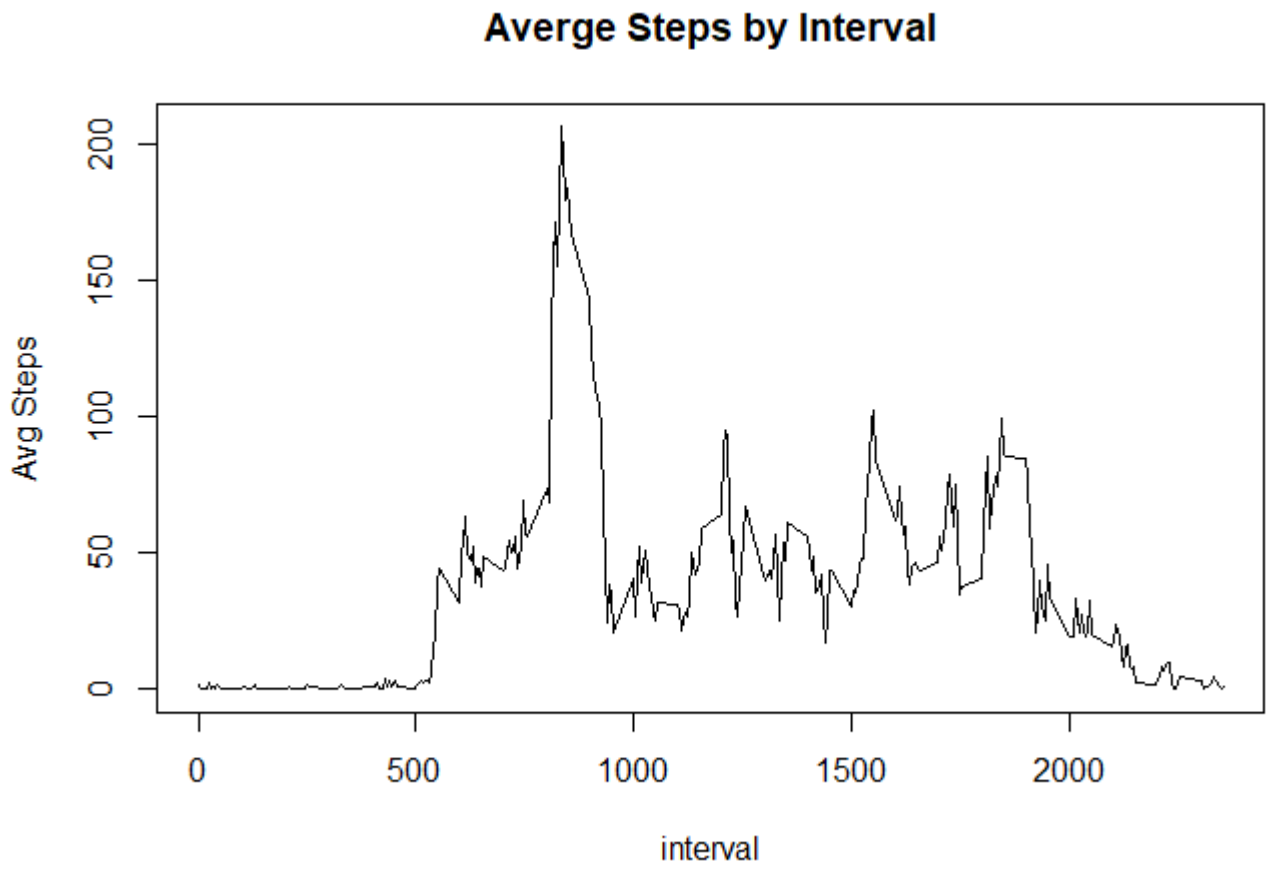
```
hist(dailytotal$TotalSteps, main='Histogram for Total Steps of each day',
     xlab = 'Total Steps for each day', col='forestgreen')
```




# What is the average daily activity pattern?

```
avg.interval <- activity.clean[, .(Avg = mean(steps)), by=interval]

plot(avg.interval$interval, avg.interval$Avg, type='l', main='Averge Steps by Interval',
     xlab = 'interval', ylab = 'Avg Steps')
```




Find the interval which contains the maxium steps:

```
avg.interval[which(avg.interval$Avg == max(avg.interval$Avg)),]
```

```
##    interval      Avg
## 1:      835 206.1698
```

# Imputing missing values

```
cat('Total number of rows that have missing values: ', sum(is.na(activity)), '\n')
```

```
## Total number of rows that have missing values:  2304
```

```
cat('Number of missing dates:', sum(is.na(activity$date)),'\n' )
```

```
## Number of missing dates: 0
```

```
cat('Number of missing steps:', sum(is.na(activity$steps)), '\n' )
```

```
## Number of missing steps: 2304
```

```
cat('Number of missing intervals"', sum(is.na(activity$interval)), '\n')
```

```
## Number of missing intervals" 0
```

Impute the missing values by filling in the interval averages using data.table packages.

```
activity[, avg:=mean(steps, na.rm = TRUE), by=interval][is.na(steps), steps:=avg][, avg:=NULL]
```

Replot the daily average steps and recaculate the mean and median.

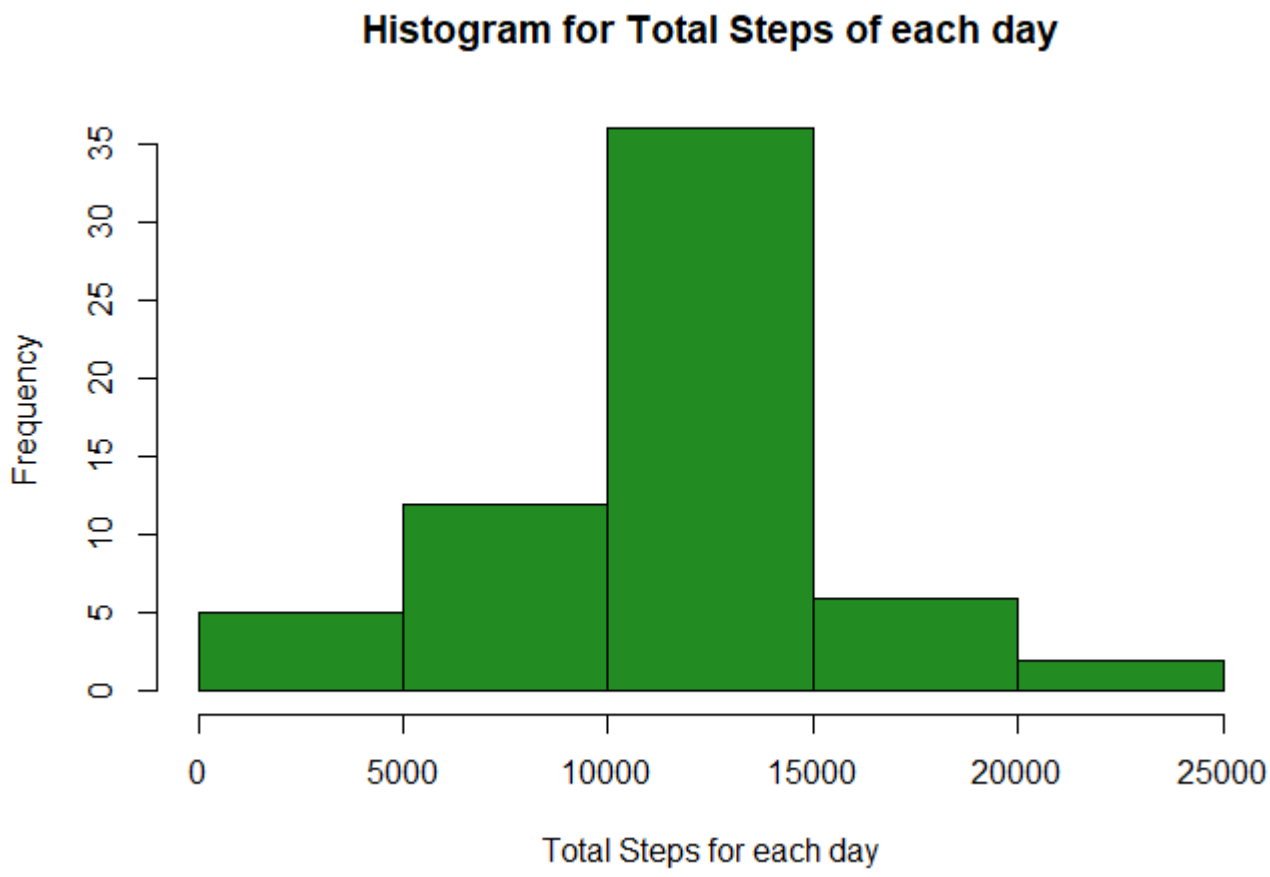
```
dailytotal2 <- activity[, .(TotalSteps = sum(steps, na.rm = TRUE)), by=date]
cat('mean total number of steps taken per day:', mean(dailytotal2$TotalSteps), '\n')
```

```
## mean total number of steps taken per day: 10749.77
```

```
cat('median total number of steps taken per day:', median(dailytotal2$TotalSteps))
```

```
## median total number of steps taken per day: 10641
```

```
hist(dailytotal2$TotalSteps, main='Histogram for Total Steps of each day',
     xlab = 'Total Steps for each day', col='forestgreen')
```



As we filled the missing values with interval averages, this does not have a big impact on the overall averages. The mean and median were only of slight difference and the distribution was almost identical.

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

```
activity[, weekday:=weekdays(date, abbreviate=TRUE)]
activity$weekday <- ifelse(activity$weekday %in% c('Sat', 'Sun'), "Weekend", "Weekday")
activity$weekday <- as.factor(activity$weekday)

avg2 <- activity[, .(Avg = mean(steps)), by=.(weekday, interval)]
library(lattice)

xyplot(Avg ~ interval | weekday, data=avg2, lay = c(1,2), type='l',
       ylab = 'Number of Steps', main='Avg Steps Comparison')
```