Reproducible Research: Peer Assessment 1

Loading and preprocessing the data

```
file.name <- "activity.csv"
df <-  read.csv(file.name, stringsAsFactor = F)
df$date <- as.Date(df$date)
#Preparation of set without missing values
index <- which(is.na(df$steps))
df1 <- df[-index,]
#Data preparation for ploting
day.steps <- list(mean = lapply(split(df1$steps,df1$date), mean)
                  , median = lapply(split(df1$steps,df1$date), median)
                  , total = lapply(split(df1$steps,df1$date), sum))
number.of.steps <-data.frame(date = as.Date(names(day.steps$total))
                             , total = as.numeric(day.steps$total)
                             , mean = as.numeric(day.steps$mean)
                             , median = as.numeric(day.steps$mean))
```

What is mean total number of steps taken per day?

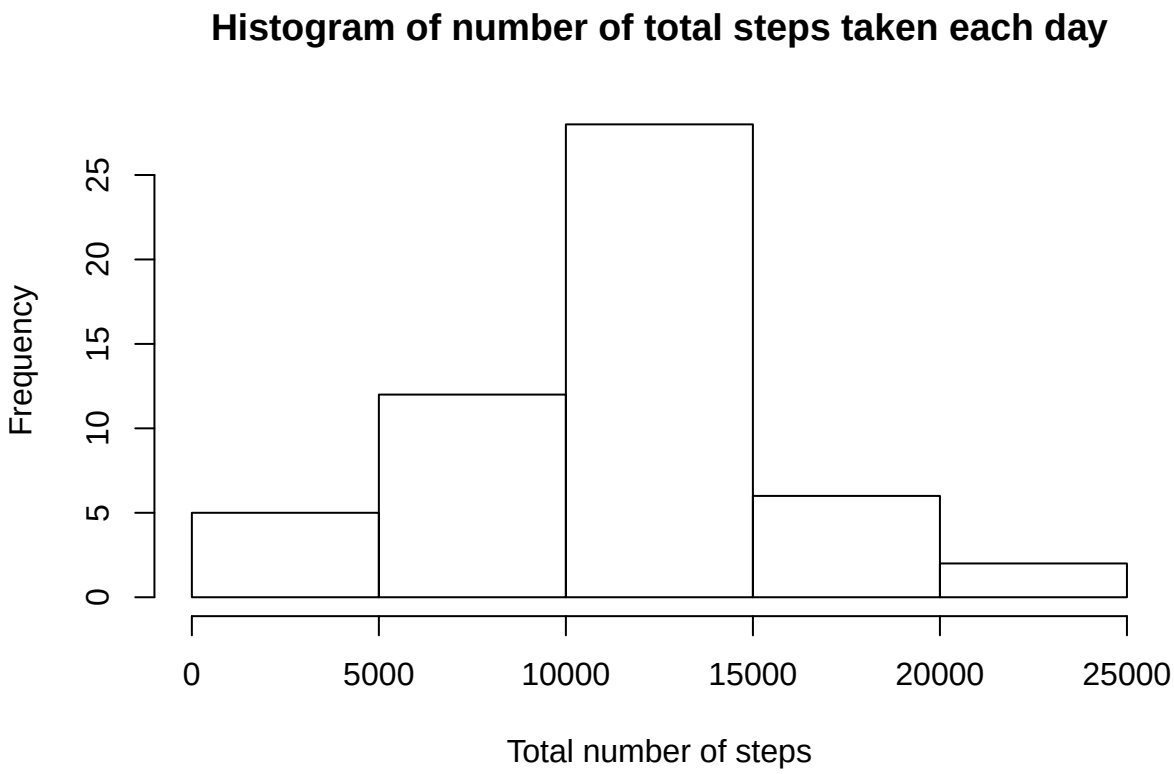
```
#Calculation mean and median
median(number.of.steps$total)
```

```
## [1] 10765
```

```
mean(number.of.steps$total)
```

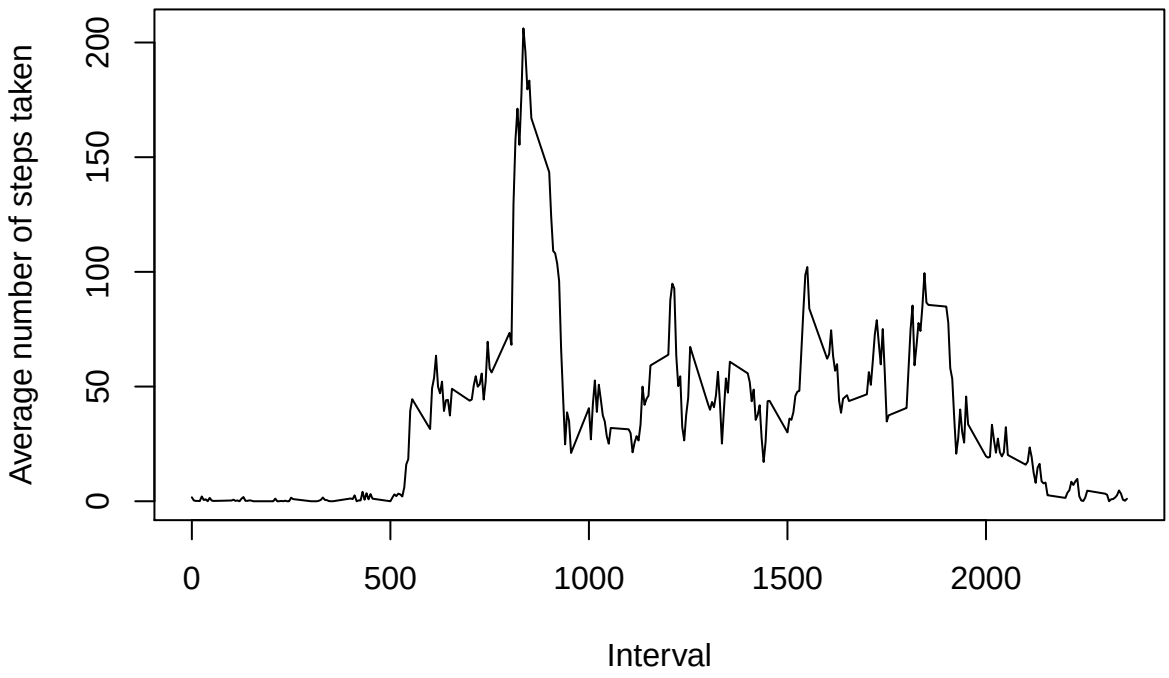
```
## [1] 10766.19
```

```
#Plot of appropriate histogram
hist(number.of.steps$total
     , xlab = "Total number of steps"
     , main = "Histogram of number of total steps taken each day")
```



What is the average daily activity pattern?

```
mean.steps<-lapply(split(df1$steps,df1$interval),mean)
#Plot of mean number of steps per specific interval
plot(names(mean.steps), mean.steps, type="l"
     , xlab = "Interval", ylab = "Average number of steps taken")
```




```
#Calculated maximum value and interval for which it occur
max(as.numeric(mean.steps))
```

```
## [1] 206.1698
```

```
max.steps.index <- which(mean.steps == max(as.numeric(mean.steps)))
```

Imputing missing values

```
#Function to feel in missing values with appropriate input
#It would put: mean value for whole day if there was any data
#              mean value of whole data set if there is no value for particular day
insert.mean <- function(x, y, response = number.of.steps){
   if(is.na(x)){
      if(!y %in% response$date)
         return(mean(response$mean))
      return(response$mean[which(y == response$date)])
   }
   return(x)
}
new.df <- df
#Filling missing values
new.df$steps<- mapply(insert.mean, df$steps,df$date)
```

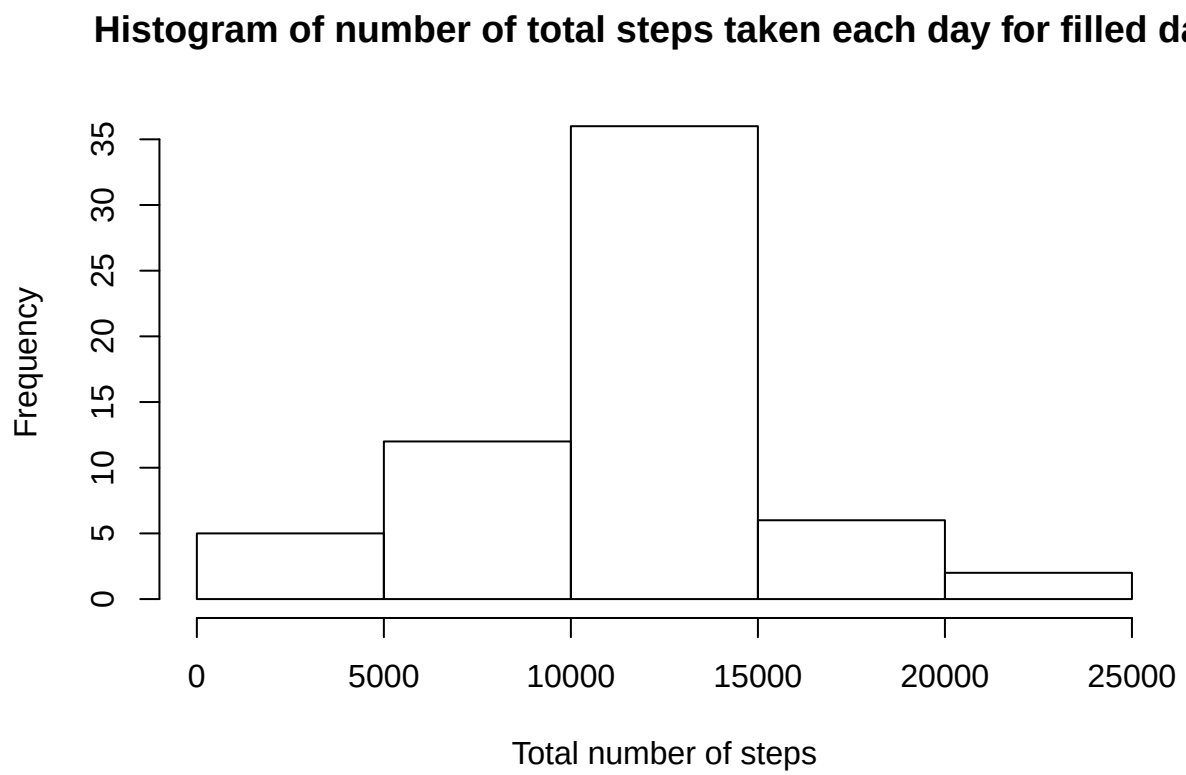
```
day.steps.new <- list(mean = lapply(split(new.df$steps, new.df$date), mean)
                  , median = lapply(split(new.df$steps, new.df$date), median)
                  , total = lapply(split(new.df$steps, new.df$date), sum))

number.of.steps.new <- data.frame(date = as.Date(names(day.steps.new$total))
                              , total = as.numeric(day.steps.new$total)
                              , mean = as.numeric(day.steps.new$mean)
                              , median = as.numeric(day.steps.new$median))
#Calculated median mean and plot of histogram
median(number.of.steps.new$total)
```

```
## [1] 10766.19
```

```
mean(number.of.steps.new$total)
```

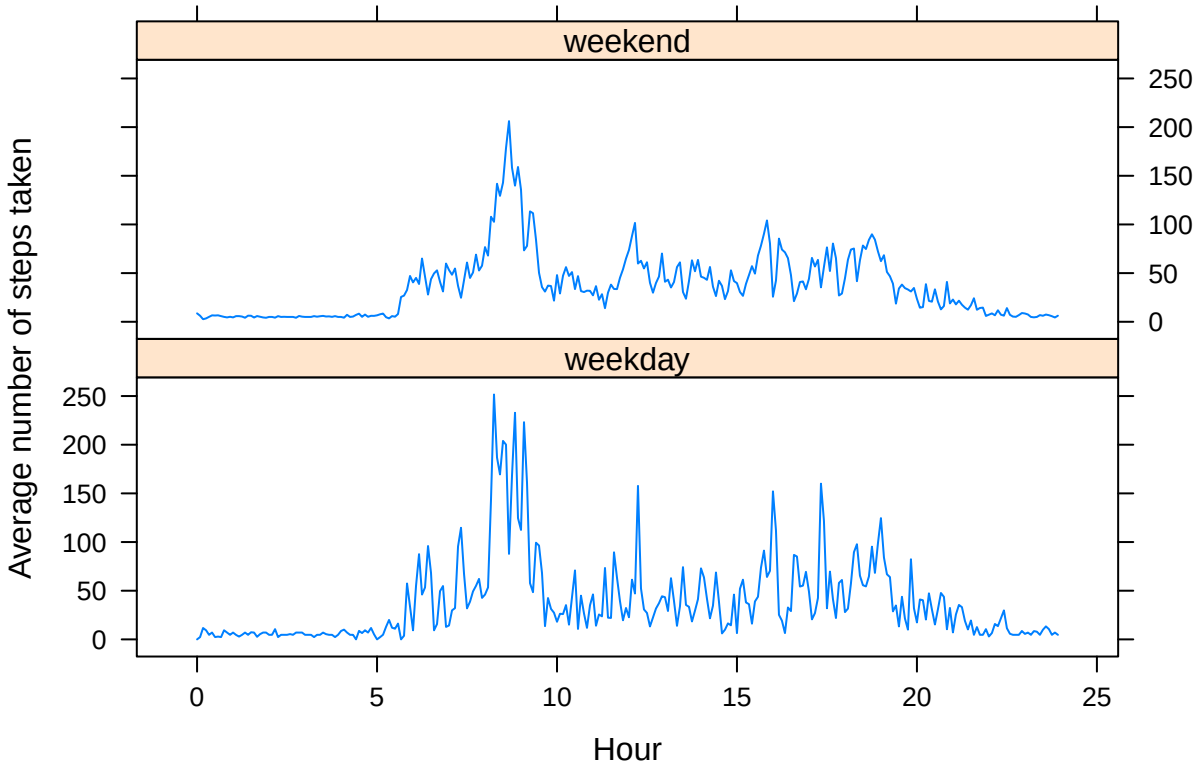
```
## [1] 10766.19
```

```
hist(number.of.steps.new$total
     , xlab = "Total number of steps"
     , main = "Histogram of number of total steps taken each day for filled data")
```



Are there differences in activity patterns between weekdays and weekends?

```
#Preparation data set to construct the plot of
#Number of steps taken for particular hour for weekdays and weekends
new.df$day_type <- ifelse(weekdays(number.of.steps.new$date) %in%
                            c("Saturday","Sunday"), "weekend", "weekday")
new.df$day_type <- as.factor(new.df$day_type)
new.df$day_type <- factor(new.df$day_type, labels = c("weekend", "weekday"))
df.weekends <- new.df[which(new.df$day_type == "weekend"), ]
df.weekdays <- new.df[which(new.df$day_type == "weekday"), ]

#Final data frame
data.paterns <- data.frame(means = c(as.numeric(lapply(split(df.weekends$steps
                                                    ,df.weekends$interval),mean))
                                , as.numeric(lapply(split(df.weekdays$steps
                                                      ,df.weekdays$interval),mean)))
                     , interval = c(names(lapply(split(df.weekends$steps
                                                    ,df.weekends$interval),mean))
                                , names(lapply(split(df.weekdays$steps
                                                   ,df.weekdays$interval),mean)))
                     , day_type = c(rep("weekend",288),rep("weekday",288)))


#Construction variable responsible for hour
data.paterns$time <- floor(as.numeric(as.character(data.paterns$interval))/100) +
                     ((as.numeric(as.character(data.paterns$interval))/100)%%1)/0.6

library(lattice)
#Plot of the Average number of steps taken for weekdays and weekends
xyplot(means ~ time | day_type, type = "l", data = data.paterns, layout = c(1, 2)
       , xlab = "Hour", ylab = "Average number of steps taken")
```


weekend
weekday
Average number of steps taken
Hour
0
50
100
150
200
250
0
5
10
15
20
25