Assignment Instructions

1.Code for reading in the dataset and/or processing the data

2.Histogram of the total number of steps taken each day

3.Mean and median number of steps taken each day

4.Time series plot of the average number of steps taken

5.The 5-minute interval that, on average, contains the maximum number of steps

6.Code to describe and show a strategy for imputing missing data

7.Histogram of the total number of steps taken each day after missing values are imputed

8.Panel plot comparing the average number of steps taken per 5-minute interval across weekdays and weekends

9.All of the R code needed to reproduce the results (numbers, plots, etc.) in the report

knitr::opts_chunk$set(warning=FALSE)

Loading and Data preparation

```
library(data.table)
```

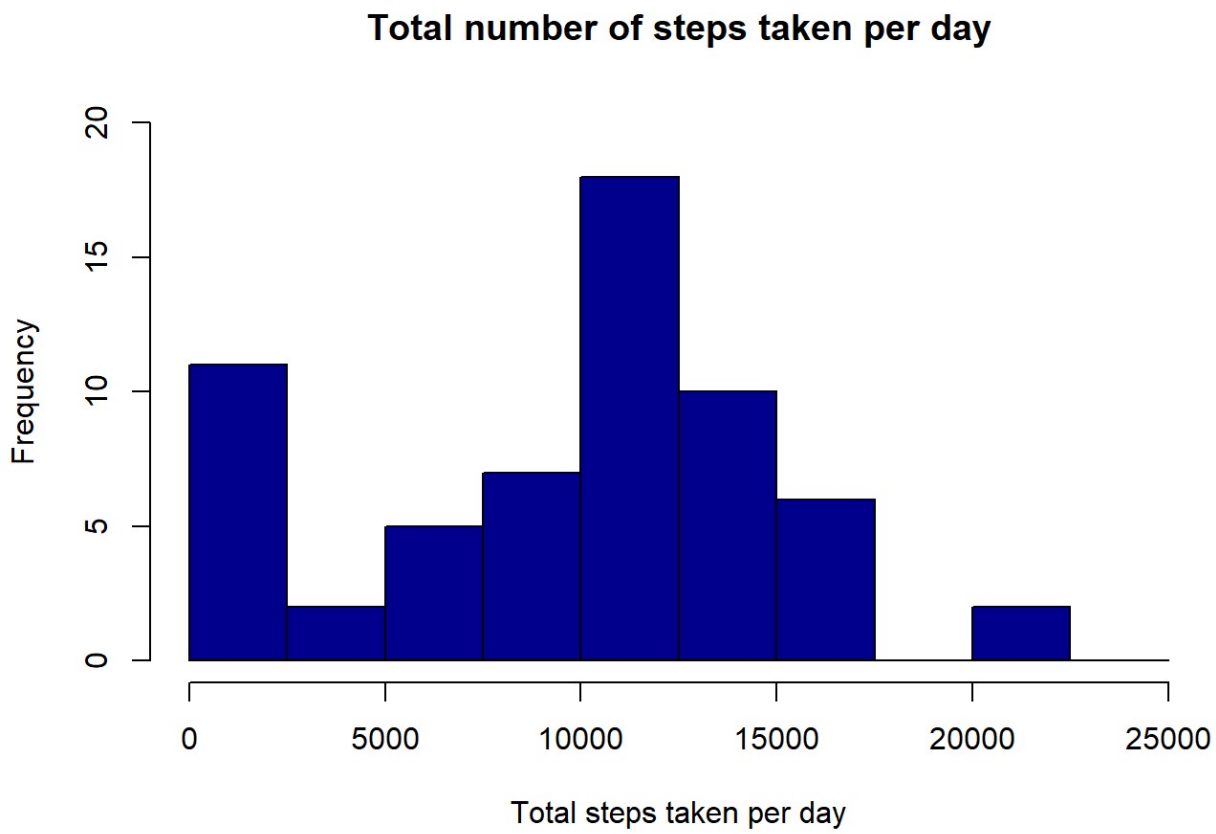
```
## Warning: package 'data.table' was built under R version 3.5.3
```

```
library(ggplot2)

activity <- read.csv("C:/Users/snamin/Documents/GitHub/RepData_PeerAssessment1/activit
y.csv")

activity$date <- as.POSIXct(activity$date, format ="%Y-%m-%d")
weekday <- weekdays(activity$date)
activity <- cbind(activity,weekday)

summary(activity)
```

```
##      steps             date                       interval     
##  Min.   :  0.00   Min.   :2012-10-01 00:00:00   Min.   :   0.0  
##  1st Qu.:  0.00   1st Qu.:2012-10-16 00:00:00   1st Qu.: 588.8  
##  Median :  0.00   Median :2012-10-31 00:00:00   Median :1177.5  
##  Mean   : 37.38   Mean   :2012-10-31 00:25:34   Mean   :1177.5  
##  3rd Qu.: 12.00   3rd Qu.:2012-11-15 00:00:00   3rd Qu.:1766.2  
##  Max.   :806.00   Max.   :2012-11-30 00:00:00   Max.   :2355.0  
##  NA's   :2304                                                   
##       weekday     
##  Friday   :2592  
##  Monday   :2592  
##  Saturday :2304  
##  Sunday   :2304  
##  Thursday :2592  
##  Tuesday  :2592  
##  Wednesday:2592
```

What is mean total number of steps taken per day?

```
activity_total_steps <- with(activity, aggregate(steps, by = list(date), FUN = sum, n
a.rm = TRUE))
names(activity_total_steps) <- c("date", "steps")
hist(activity_total_steps$steps, main = "Total number of steps taken per day", xlab =
"Total steps taken per day", col = "darkblue", ylim = c(0,20), breaks = seq(0,25000, b
y=2500))
```



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

```
mean(activity_total_steps$steps)
```

```
## [1] 9354.23
```

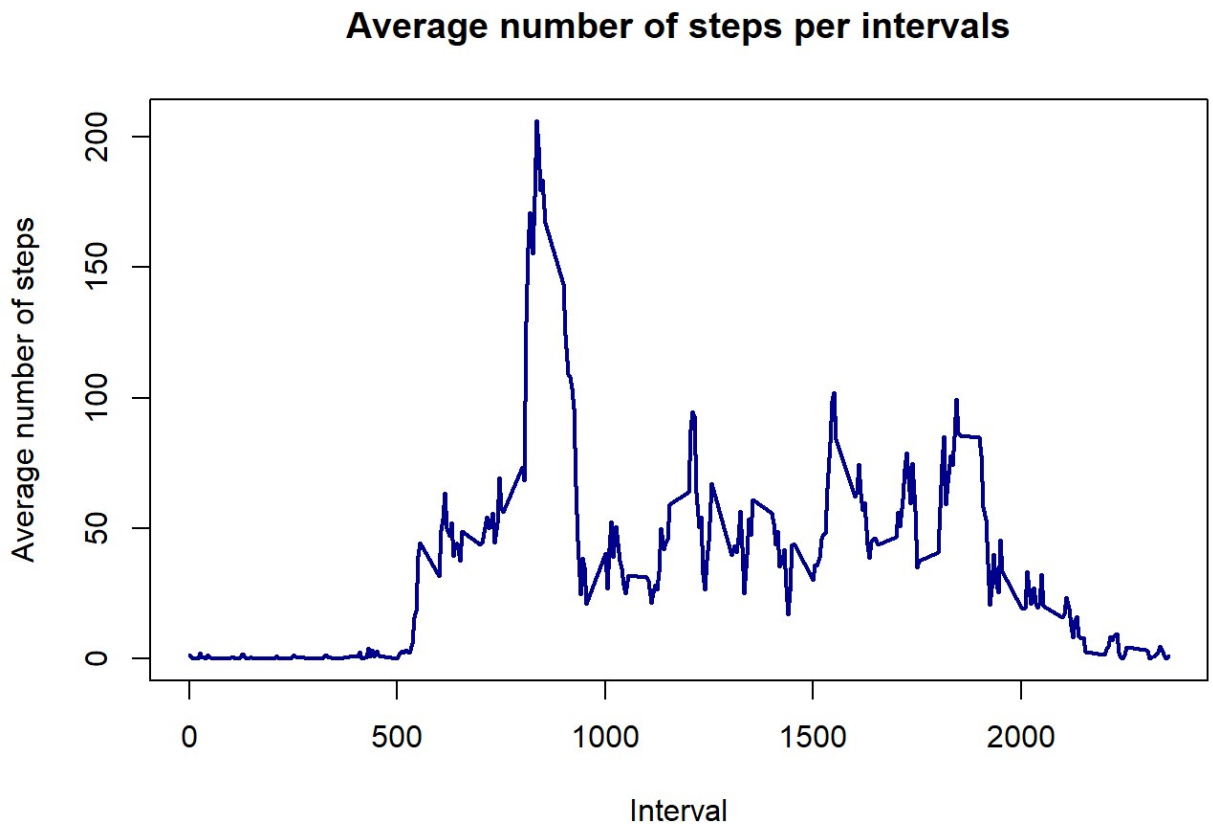
What is the median of the total number of steps taken per day?

```
median(activity_total_steps$steps)
```

```
## [1] 10395
```

What is the average daily activity pattern?

```
average_daily_activity <- aggregate(activity$steps, by=list(activity$interval), FUN=me
an, na.rm=TRUE)
names(average_daily_activity) <- c("interval", "mean")
plot(average_daily_activity$interval, average_daily_activity$mean, type = "l", col="da
rkblue", lwd = 2, xlab="Interval", ylab="Average number of steps", main="Average numbe
r of steps per intervals")
```

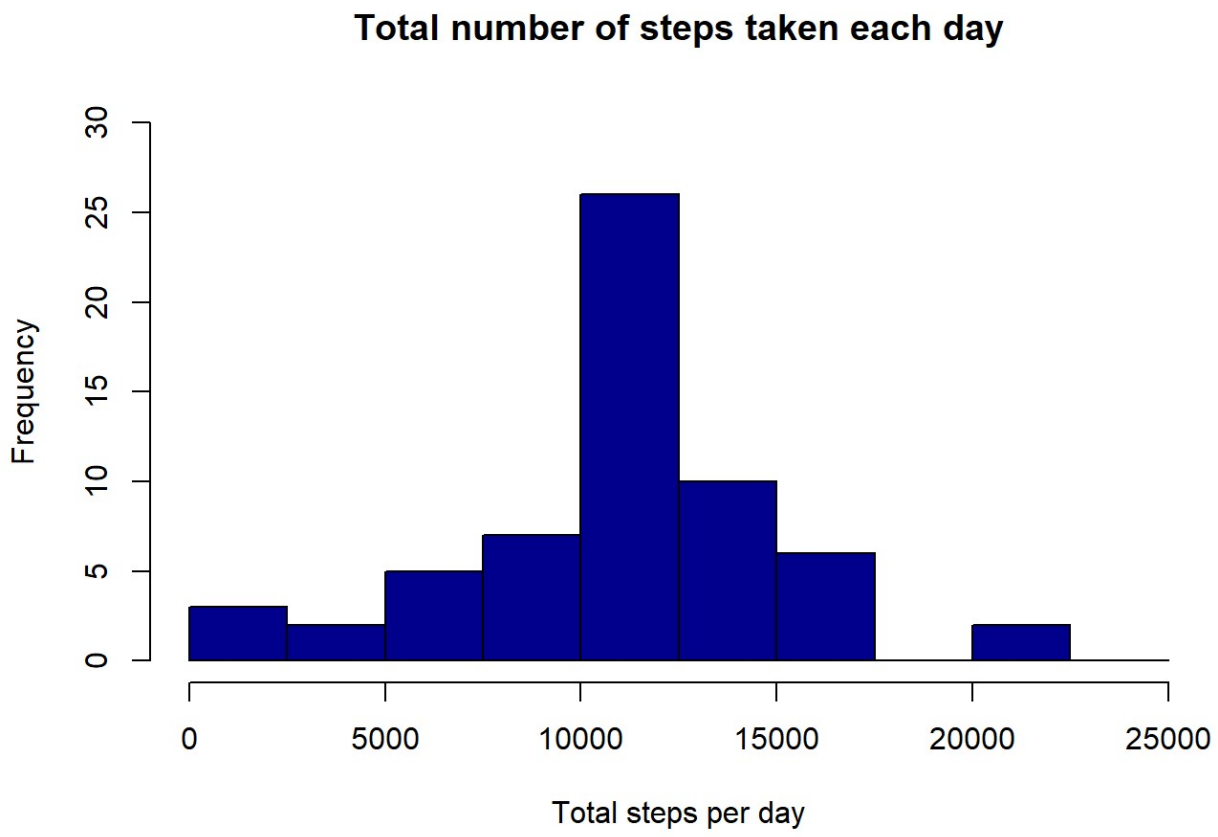


Imputing missing values

```
sum(is.na(activity$steps))
```

```
## [1] 2304
```

```
imputed_steps <- average_daily_activity$mean[match(activity$interval, average_daily_ac
tivity$interval)]

# Create a new dataset with the missing data filled in
activity_imputed <- transform(activity, steps = ifelse(is.na(activity$steps), yes = im
puted_steps, no = activity$steps))
total_steps_imputed <- aggregate(steps ~ date, activity_imputed, sum)
names(total_steps_imputed) <- c("date", "daily_steps")

# Create its histogram

hist(total_steps_imputed$daily_steps, col = "darkblue", xlab = "Total steps per day",
ylim = c(0,30), main = "Total number of steps taken each day", breaks = seq(0,25000,by
=2500))
```




```
#Calculate mean and median again

mean(total_steps_imputed$daily_steps)
```

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

```
median(total_steps_imputed$daily_steps)
```

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

Are there differences in activity patterns between weekdays and weekends?

```
#Create a new factor variable in the dataset with two levels - "weekday" and "weekend"

activity$date <- as.Date(strptime(activity$date, format="%Y-%m-%d"))
activity$datetype <- sapply(activity$date, function(x) {
        if (weekdays(x) == "Saturday" | weekdays(x) =="Sunday")
                {y <- "Weekend"} else
                {y <- "Weekday"}
                y
        })
```

```
#Make plots

activity_by_date <- aggregate(steps~interval + datetype, activity, mean, na.rm = TRUE)
plot<- ggplot(activity_by_date, aes(x = interval , y = steps, color = datetype)) +
       geom_line() +
       labs(title = "Average daily steps by type of date", x = "Interval", y = "Averag
e number of steps") +
       facet_wrap(~datetype, ncol = 1, nrow=2)
print(plot)
```