Reproducible Research Project

Question 1: Import, load, read, summarize, and identify header names of the activity data set.

```
activity_dataset <- read.csv("C:/Users/rashe/Documents/Education/Data_Science_Specialization/Foundations
View(activity_dataset)
summary(activity_dataset)
```

```
##      steps            date                cc               interval
##  Min.   :  0.00   Length:17568       Length:17568       Min.   :   0.0
##  1st Qu.:  0.00   Class :character   Class :character   1st Qu.: 588.8
##  Median :  0.00   Mode  :character   Mode  :character   Median :1177.5
##  Mean   : 37.38                                         Mean   :1177.5
##  3rd Qu.: 12.00                                         3rd Qu.:1766.2
##  Max.   :806.00                                         Max.   :2355.0
##  NA's   :2304
```

```
names(activity_dataset)
```

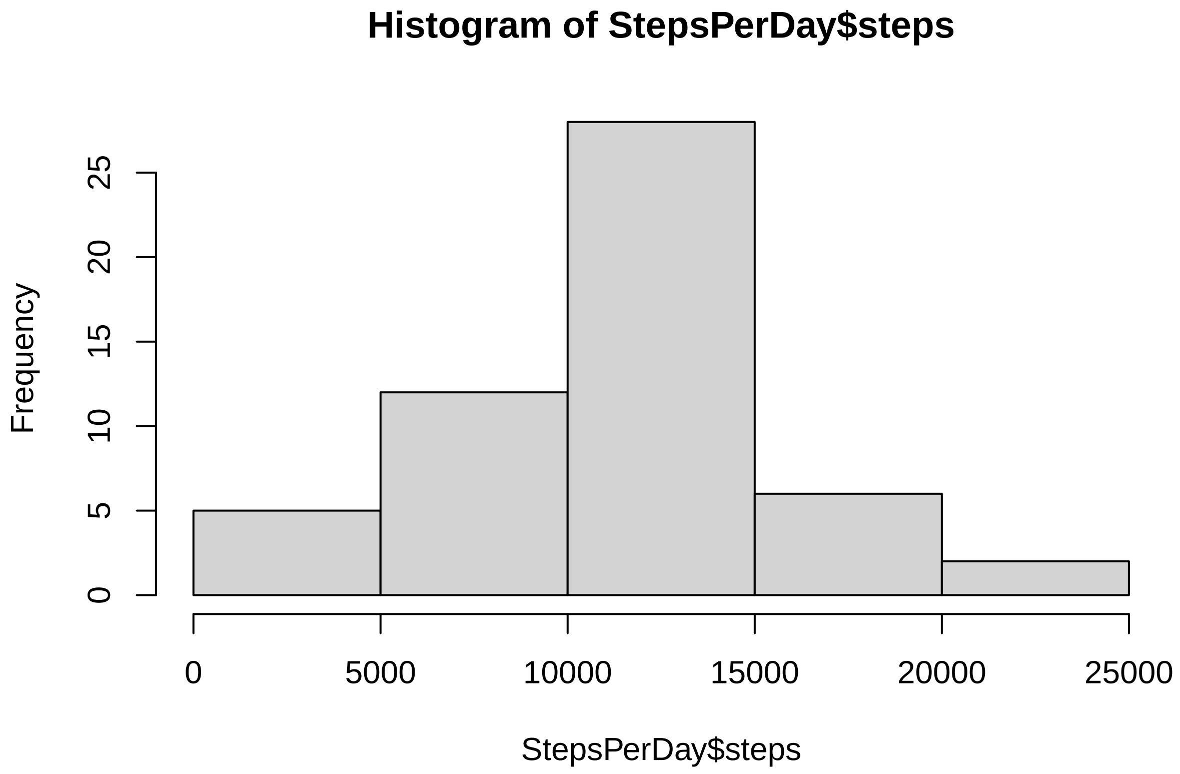
```
## [1] "steps"    "date"     "cc"       "interval"
```

Question 2: Calculate the total number of steps per day.

```
StepsPerDay <- aggregate(steps ~ date, activity_dataset, sum, na.rm = TRUE)
```

Question 3: Plot a histogram of the total number of steps per day.

```
hist(StepsPerDay$steps)
```



Question 4a: Calculate the mean of total steps per day.

```
avg_StepsPerDay <- mean(StepsPerDay$steps)
avg_StepsPerDay
```

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

Question 4b: Calculate the median of total steps per day.

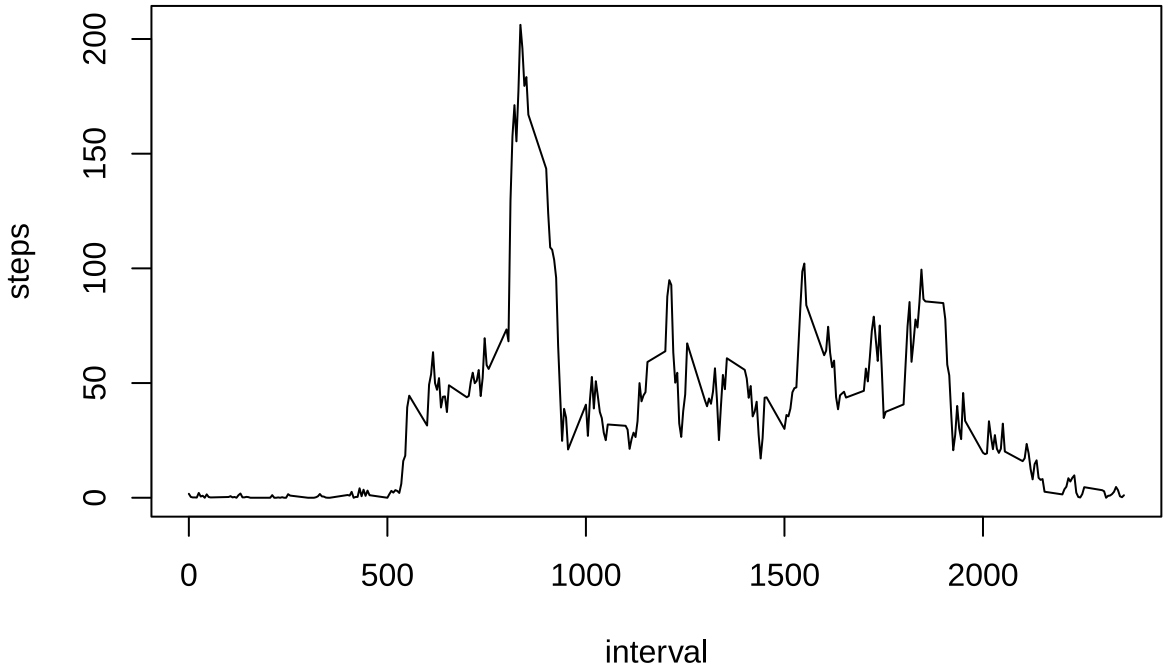
```
med_StepsPerDay <- median(StepsPerDay$steps)
med_StepsPerDay
```

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

Question 5a: Create a time series plot of 5 minute intervals and average total steps per day.

```
avg_StepsPerInterval <- aggregate(steps ~ interval, activity_dataset, mean, na.rm = TRUE)
```

```
plot(steps ~ interval, data = avg_StepsPerInterval, type="l")
```



Question 5b: Calculate interval with maximum number of steps.

```
max_StepsPer_Interval <- avg_StepsPerInterval[which.max(avg_StepsPerInterval$steps),]$interval
max_StepsPer_Interval
```

```
## [1] 835
```

Question 6a: Count total number of missing values (N/A).

```
missing_count <- sum(is.na(activity_dataset$steps))
missing_count
```

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

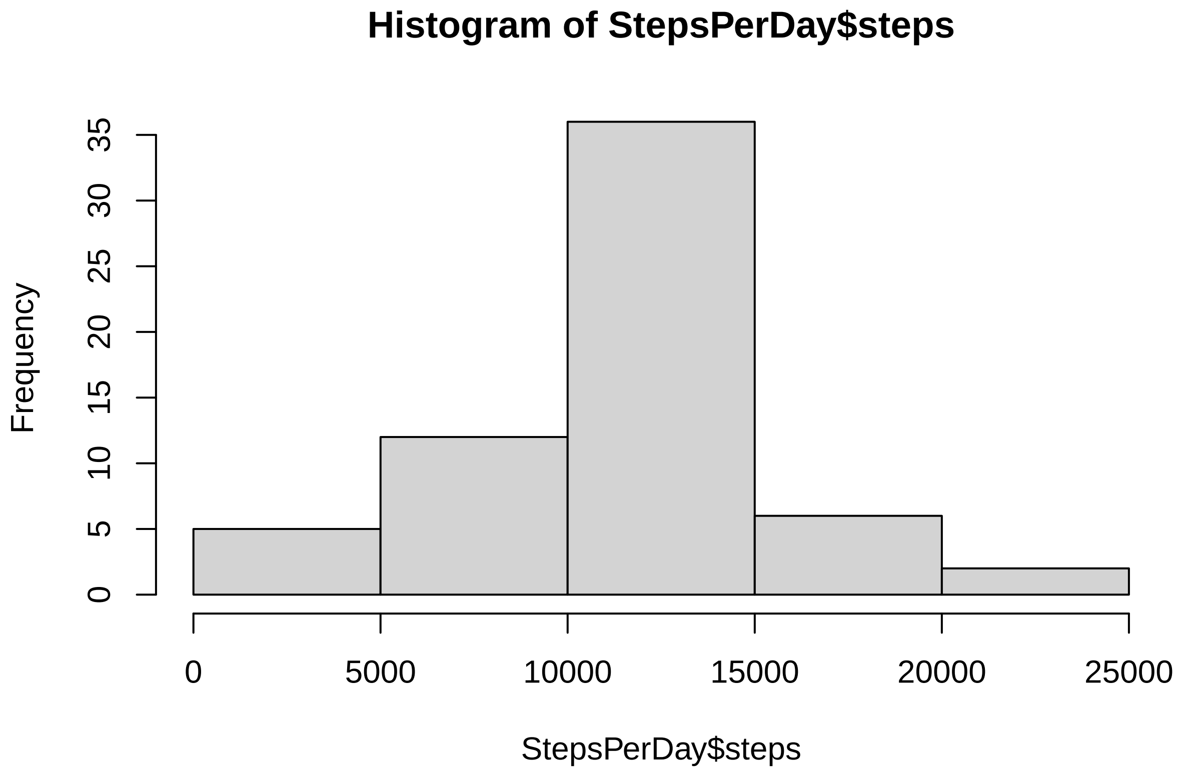
Question 6b: Replace missing values (N/A) with average steps per interval.

```
Mean_StepsPerInterval <- function(interval){
    avg_StepsPerInterval[avg_StepsPerInterval$interval==interval,]$steps}
```

Question 6c: Update data set, histogram, and recalculate mean and median.

```
new_estimates <- activity_dataset
for(i in 1:nrow(new_estimates)){
    if(is.na(new_estimates[i,]$steps)){
        new_estimates[i,]$steps <- Mean_StepsPerInterval(new_estimates[i,]$interval)}}
```

```
StepsPerDay <- aggregate(steps ~ date, data = new_estimates, sum)
hist(StepsPerDay$steps)
```



Mean after filling missing values.

```
mean_new <- mean(StepsPerDay$steps)
mean_new
```

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

Median after filling missing values.

```
median_new <- median(StepsPerDay$steps)
median_new
```

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

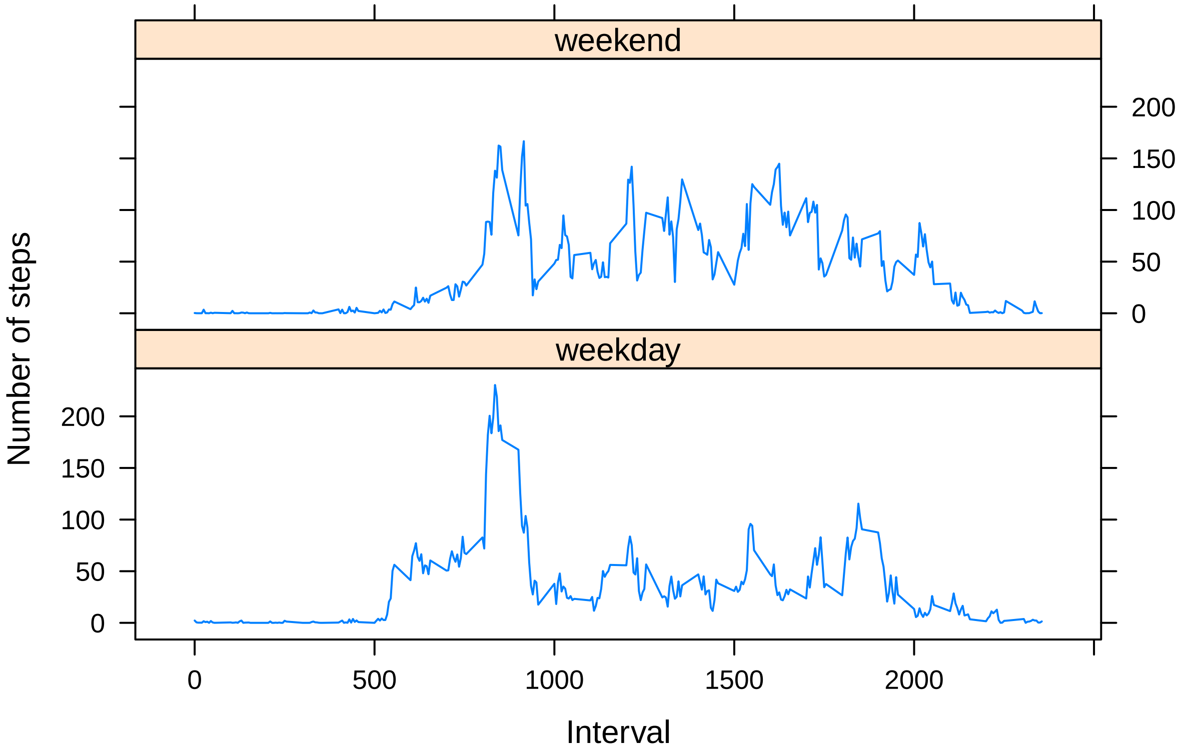
Question 7a: Create new factor variable.

```
new_estimates$date <- as.Date(strptime(new_estimates$date, format="%Y-%m-%d"))
new_estimates$day <- weekdays(new_estimates$date)
for (i in 1:nrow(new_estimates)) {
    if (new_estimates[i,]$day %in% c("Saturday","Sunday")) {
        new_estimates[i,]$day<-"weekend"}
    else{new_estimates[i,]$day<-"weekday"}}
```

```
StepsByDay <- aggregate(new_estimates$steps ~ new_estimates$interval + new_estimates$day, new_estimates
```

Question 7b: Create time series plot of weekdays versus weekends.

```
names(StepsByDay) <- c("interval", "day", "steps")
library(lattice)
xyplot(steps ~ interval | day, StepsByDay, type = "l", layout = c(1, 2),
    xlab = "Interval", ylab = "Number of steps")
```


weekend
weekday
Number of steps
Interval