Coursera:- Reproducible Research Peer assessment 1

Author: Mehul Patel.

Loading and preprocessing the data

Loading the preprocessing the data from the zip file.

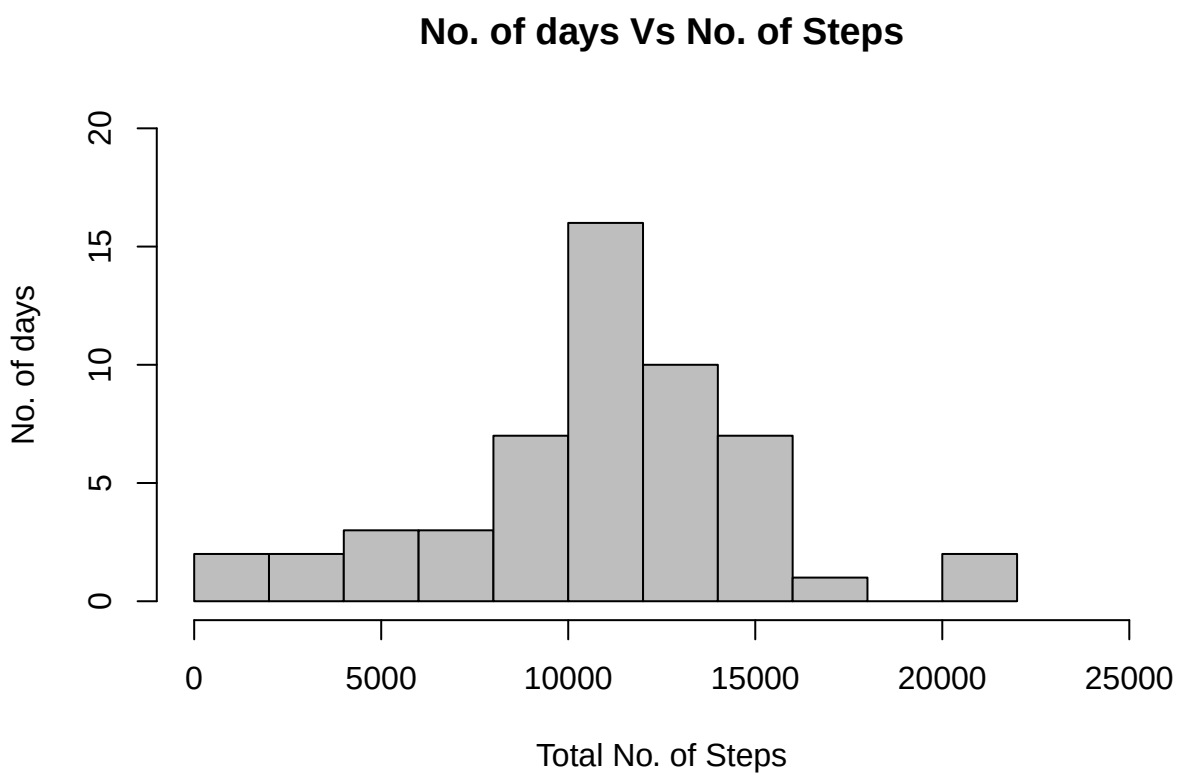
```
activity <- read.table(unz("activity.zip", "activity.csv"), header=T, quote="\"", sep=",")
activity$date <- as.Date(activity$date)
```

What is mean total number of steps taken per day?

Steps to find the mean, median and plotting the histogram

- Create a data frame from the CSV file removing NA values
- Plot the Histogram of Total number of steps taken per day
- Find mean and median

```
activity.no.na <- na.omit(activity)
dailySteps <- rowsum(activity.no.na$steps, format(activity.no.na$date,'%Y-%m-%d'))
dailySteps <- data.frame(dailySteps)
names(dailySteps) <- "Steps"
hist(dailySteps$Steps, main = "No. of days Vs No. of Steps", xlab = "Total No. of Steps", ylab = "No. o
breaks = 10, col = "gray",xlim = c(0, 25000), ylim = c(0, 20))
```



```
mean(dailySteps$Steps)
```

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

```
median(dailySteps$Steps)
```

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

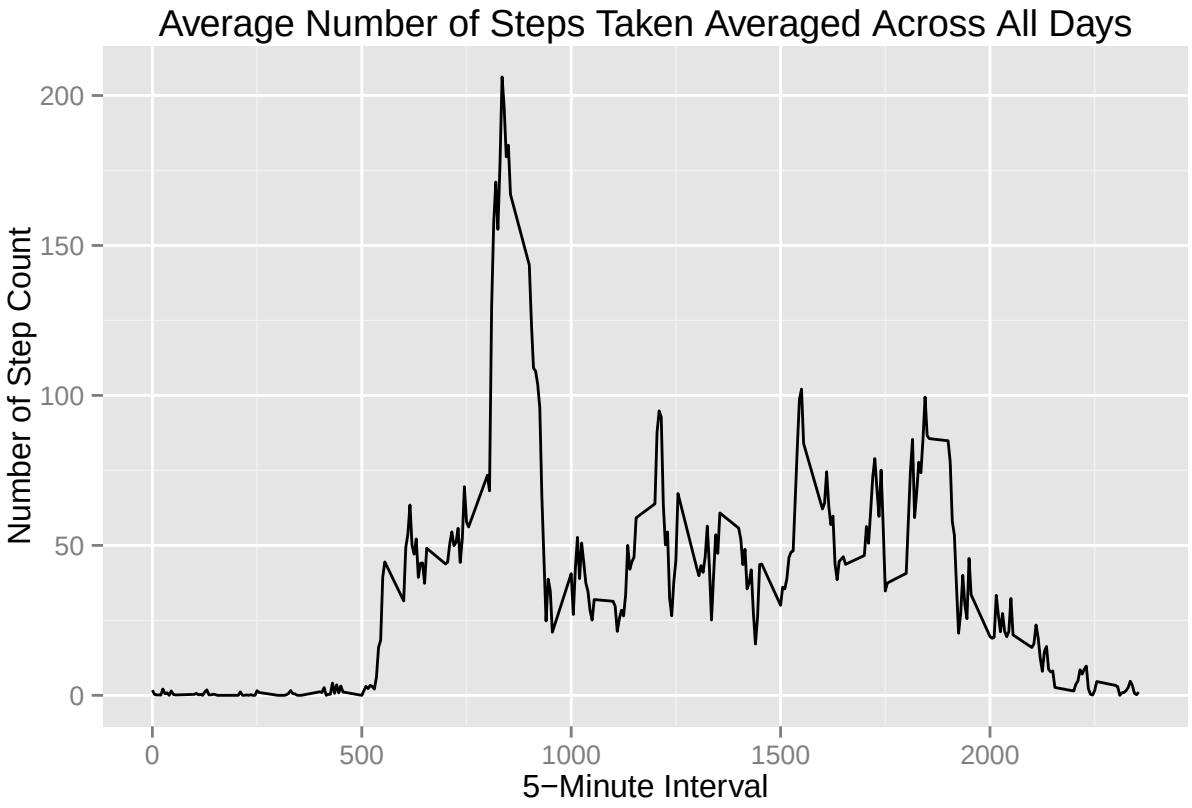
What is the average daily activity pattern?

- Calculate average steps for each of 5-minute interval during a 24-hour period.
- Make a time series plot (i.e. type = "l") of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis)
- Report which 5-minute interval, on average across all the days in the dataset, contains the maximum number of steps?
- Observe and comment the average daily activity pattern

```
library(plyr)
interval.steps <- ddply(activity.no.na, ~interval, summarize, mean = mean(steps))
```

Plot time series of the 5-minute interval and the average number of steps taken, averaged across all days.

```
library(ggplot2)
qplot(x=interval, y=mean, data = interval.steps,  geom = "line",
      xlab="5-Minute Interval",
      ylab="Number of Step Count",
      main="Average Number of Steps Taken Averaged Across All Days"
      )
```



Report the 5-min interval, on average across all the days in the dataset, contains the maximum number of steps.

```
interval.steps[which.max(interval.steps$mean), ]
```

```
##     interval     mean
## 104      835 206.1698
```

Observation: The person's daily activity peaks around 8:35 am.

Imputing missing values

Note that there are a number of days/intervals where there are missing values (coded as NA). The presence of missing days may introduce bias into some calculations or summaries of the data. In this section:

- Calculate and report the total number of missing values in the dataset (i.e. the total number of rows with NAs).
- Implement a strategy for filling in all of the missing values in the dataset. For this assignment the strategy is to use the mean for that 5-minute interval to replace missing valuse. Create a new dataset that is equal to the original dataset but with the missing data filled in.
- Make a histogram of the total number of steps taken each day.
- Calculate and report the mean and median total number of steps taken per day.

- Make following comments: Do these values differ from the estimates from the first part of the assignment? What is the impact of imputing missing data on the estimates of the total daily number of steps?

Calculate and report the total number of missing values in the dataset (i.e. the total number of rows with NAs).

```
library(sqldf)
```

```
## Loading required package: gsubfn
## Loading required package: proto
## Could not load tcltk.  Will use slower R code instead.
## Loading required package: RSQLite
## Loading required package: DBI
```

```
totalNAs <- sqldf('
    SELECT ac.* FROM "activity" as ac
    WHERE ac.steps IS NULL
    ORDER BY ac.date, ac.interval ')
NROW(totalNAs)
```

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

Implement a strategy for filling in all of the missing values in the dataset. For this assignment the strategy is to use the mean for that 5-minute interval to replace missing valuse. Create a new dataset (newActivity) that is equal to the original dataset but with the missing data filled in. The dataset is ordered by date and interval. The following SQL statement combines the original "activity" dataset set and the "interval.steps" dataset that contains mean values of each 5-min interval ageraged across all days.

```
newActivity <- sqldf('
    SELECT d.*, i.mean
    FROM "interval.steps" as i
    JOIN "activity" as d
    ON d.interval = i.interval
    ORDER BY d.date, d.interval ')

newActivity$steps[is.na(newActivity$steps)] <- newActivity$mean[is.na(newActivity$steps)]
```

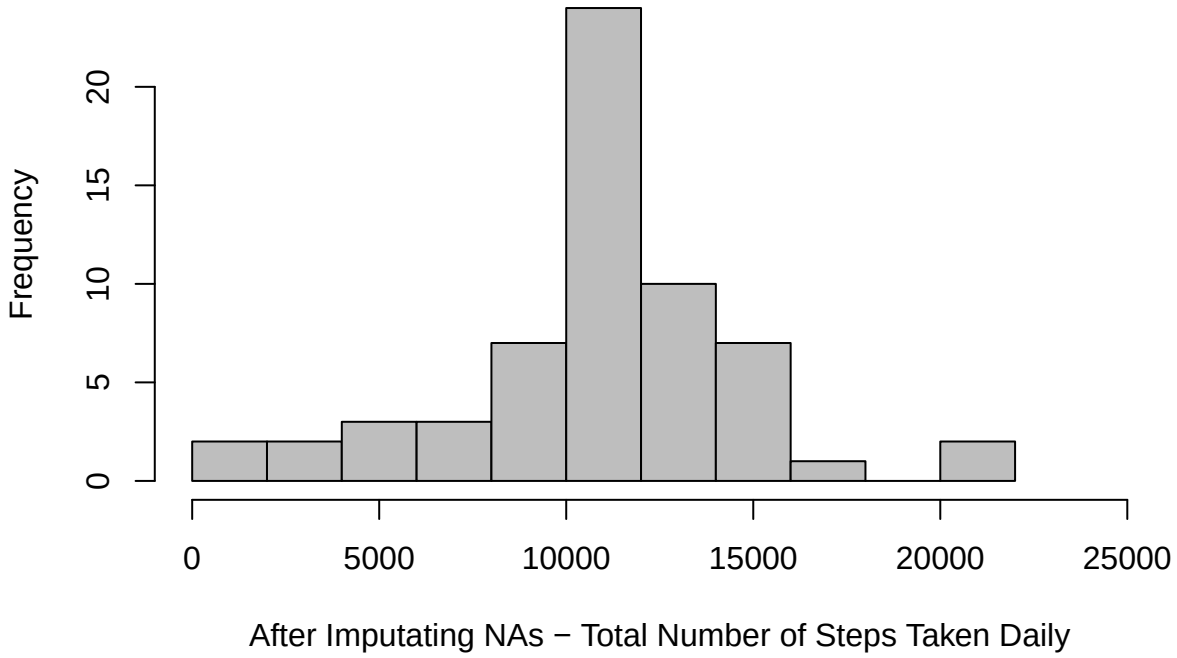
Prepare data to plot histogram calculate mean and median.

```
total.steps <- as.integer(sqldf('SELECT sum(steps) FROM newActivity'))

total.steps.by.date <- sqldf('
    SELECT date, sum(steps) as "total.steps.by.date"
    FROM newActivity GROUP BY date
    ORDER BY date')
```

Make a histogram of the total number of steps taken each day.

```
hist(as.numeric(total.steps.by.date$total.steps.by.date), main=" ", breaks=10,
    xlab="After Imputating NAs - Total Number of Steps Taken Daily", col = "gray", xlim = c(0,25000))
```



Calculate and report the mean and median total number of steps taken per day.

```
mean.steps.per.day <- as.integer(total.steps / NROW(total.steps.by.date) )
print(mean.steps.per.day)
```

```
## [1] 10766
```

```
median.steps.per.day <- median(total.steps.by.date$total.steps.by.date)
print(median.steps.per.day)
```

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

Observations:

- Do these values (mean and median) differ from the estimates from the first part of the assignment? No.
- What is the impact of imputing missing data on the estimates of the total daily number of steps? The shape of the histogram remains the same as the histogram from removed missing values. However, the frequency counts increased as expected. In this case, it seems that the data imputation strategy should work for the downstream data analysis and modeling.

Are there differences in activity patterns between weekdays and weekends?

- Use the dataset with the filled-in missing values for this part. Create a new factor variable in the dataset with two levels – "weekday" and "weekend" indicating whether a given date is a weekday or weekend day.

- Make a panel plot containing a 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).

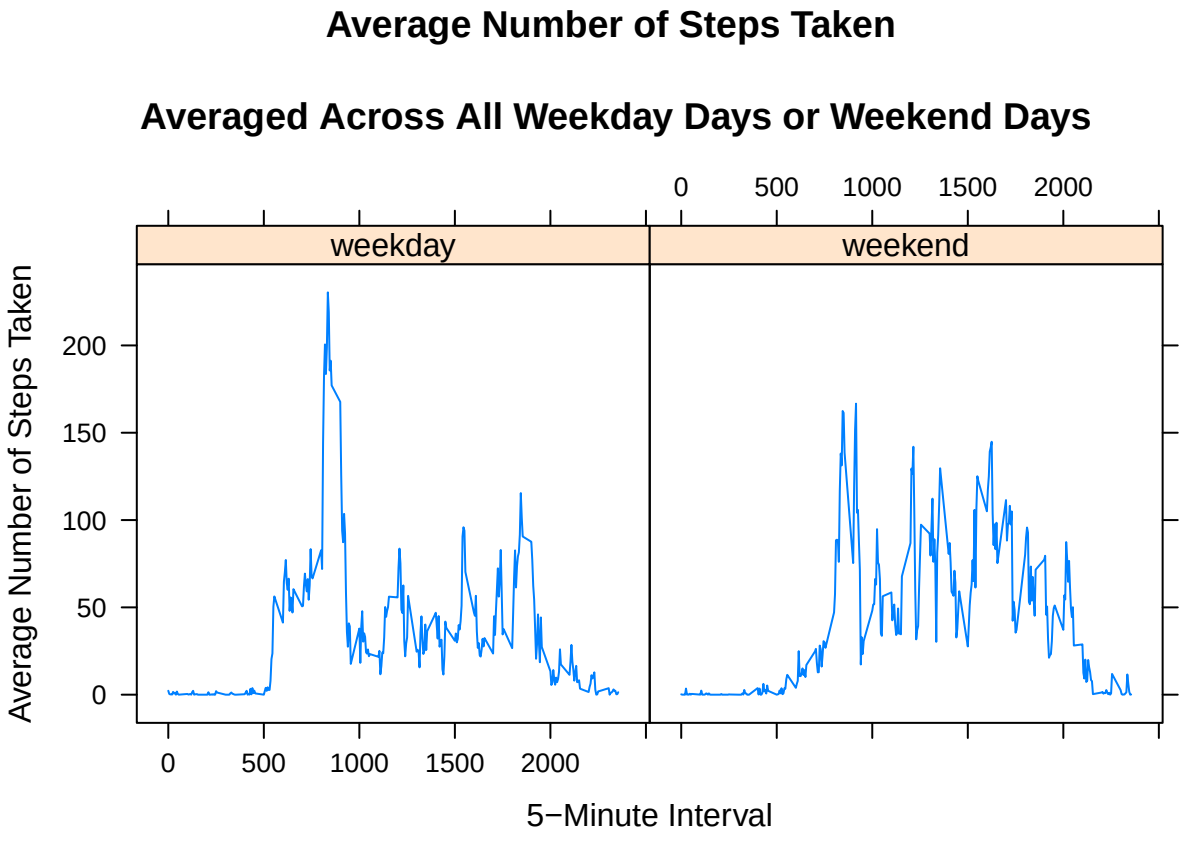
Create a factor variable weektime with two levels (weekday, weekend). The folowing dataset ww.newActivity contains data: average number of steps taken averaged across all weekday days and weekend days, 5-min intervals, and a facter variable weektime with two levels (weekday, weekend).

```
newActivity$weektime <- as.factor(ifelse(weekdays(newActivity$date) %in%
                c("Saturday","Sunday"),"weekend", "weekday"))

ww.newActivity <- sqldf('
    SELECT interval, avg(steps) as "mean.steps", weektime
    FROM newActivity
    GROUP BY weektime, interval
    ORDER BY interval ')
```

Make a panel plot containing a 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).

```
library("lattice")
p <- xyplot(mean.steps ~ interval | factor(weektime), data=ww.newActivity,
       type = 'l',
       main="Average Number of Steps Taken
       \nAveraged Across All Weekday Days or Weekend Days",
       xlab="5-Minute Interval",
       ylab="Average Number of Steps Taken")
print (p)
```



Observation:

- Are there differences in activity patterns between weekdays and weekends?
  Yes. The plot indicates that the person is more active during the weekend.