PA1_template

Joseph Givens

5/27/2020

The data for this assignment contains the following variables: - steps: Number of steps taken in five-minute intervals(missing values are NA) - date: The date on which the measurement was taken in YYYY-MM-DD format - interval: Identifier for the 5-minute interval in which measurement was taken

Loading and preprocessing the data

```r
stepData <- read.csv("activity.csv")
```

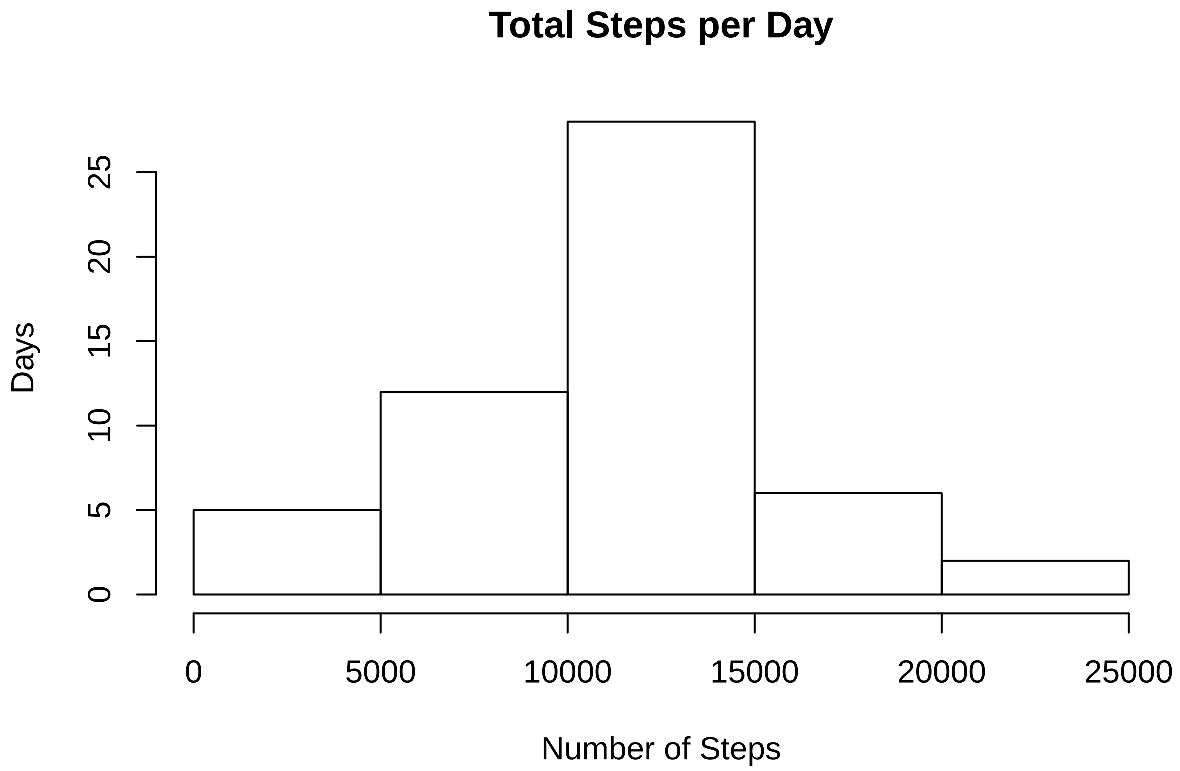
What is mean total number of steps taken?

Total number of steps taken per day

```r
totalSteps <- aggregate(steps ~ date, stepData, FUN = sum, na.rm = TRUE)
```

Histogram of the total number of steps taken each day

```r
hist(totalSteps$steps,
     xlab = "Number of Steps",
     ylab = "Days",
     main = "Total Steps per Day")
```



Mean number of steps taken each day

```
meanSteps <- mean(totalSteps$steps)
meanSteps
```

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

Median number of steps taken each day

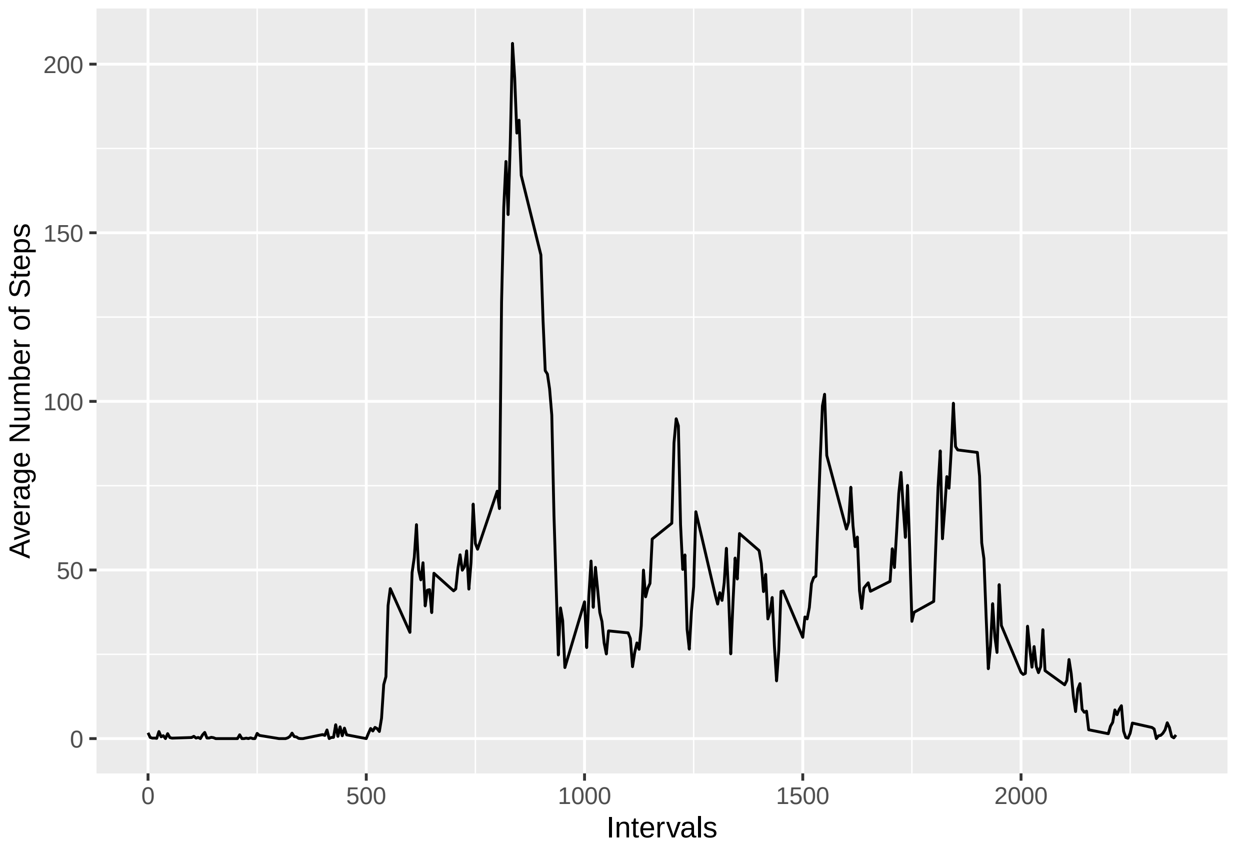
```
medianSteps <- median(totalSteps$steps)
medianSteps
```

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

What is the average daily activity pattern?

Time series plot of the 5-minute interval and the average number of steps taken

```
library(ggplot2)
avgSteps <- aggregate(steps ~ interval, stepData, mean, na.rm = TRUE)
g <- ggplot(data = avgSteps, aes(x = interval, y = steps))
g + geom_line() +
    xlab("Intervals") +
    ylab("Average Number of Steps")
```



Five-minute interval with maximum number of steps

```
avgSteps[which.max(avgSteps$steps), ]
```

```
##     interval    steps
## 104      835 206.1698
```

Imputing Missing Values

Calculate and report the total number of missing values in the dataset

```
missingValues <- sum(is.na(stepData$steps))
missingValues
```

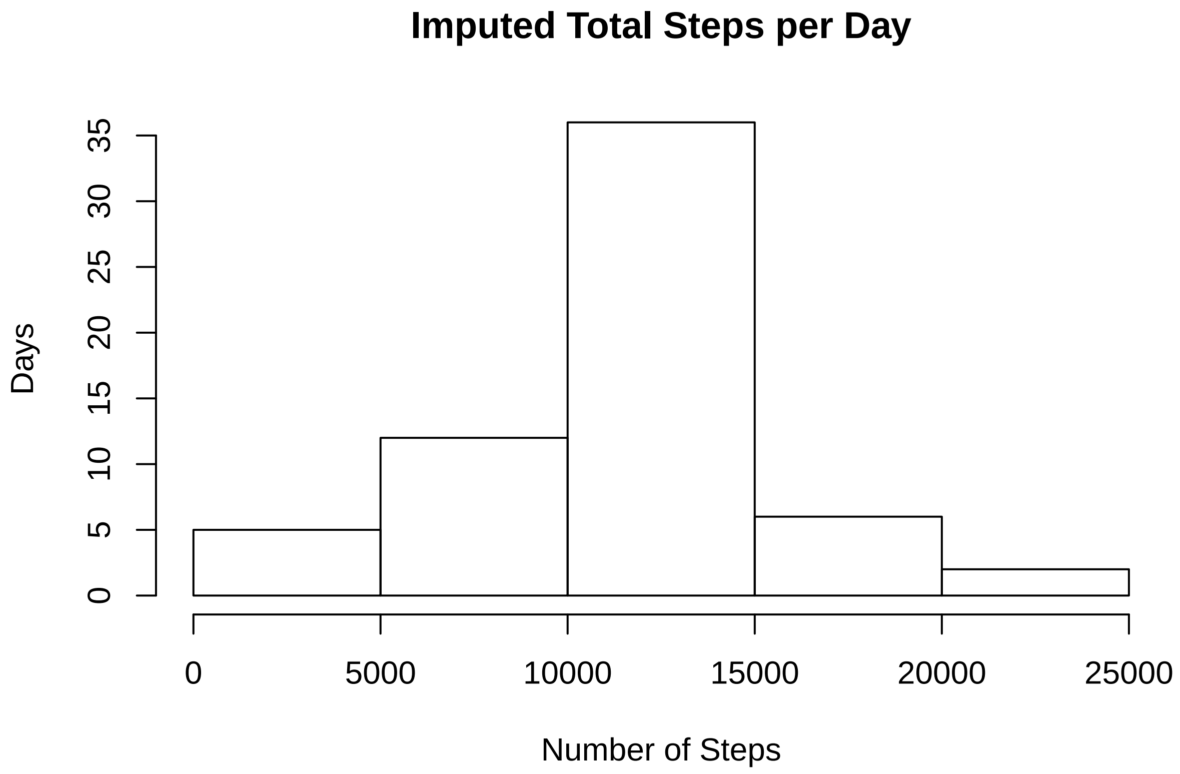
```
## [1] 2304
```

Fill in all of the missing values in the dataset

```
impStepData <- transform(stepData,
                         steps = ifelse(is.na(stepData$steps),
                                        avgSteps$steps[match(stepData$interval,
                                                             avgSteps$interval)],
                                        stepData$steps))
```

Histogram of the total number of steps taken each day

```
impTotalSteps <- aggregate(steps ~ date, impStepData, FUN = sum, na.rm = TRUE)
hist(impTotalSteps$steps,
     xlab = "Number of Steps",
     ylab = "Days",
     main = "Imputed Total Steps per Day")
```



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

```
impMeanSteps <- mean(impTotalSteps$steps, na.rm = TRUE)
impMeanSteps
```

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

```
impMedianSteps <- median(impTotalSteps$steps, na.rm = TRUE)
impMedianSteps
```

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

Impact if imputing missing data on the estimates of the total daily number of steps

```
diffMean = impMeanSteps - meanSteps
diffMean
```

```
## [1] 0
```

```
diffMedian = impMedianSteps - medianSteps
diffMedian
```

```
## [1] 1.188679
```

```
diffTotal = sum(impTotalSteps$steps) - sum(totalSteps$steps)
diffTotal
```

```
## [1] 86129.51
```

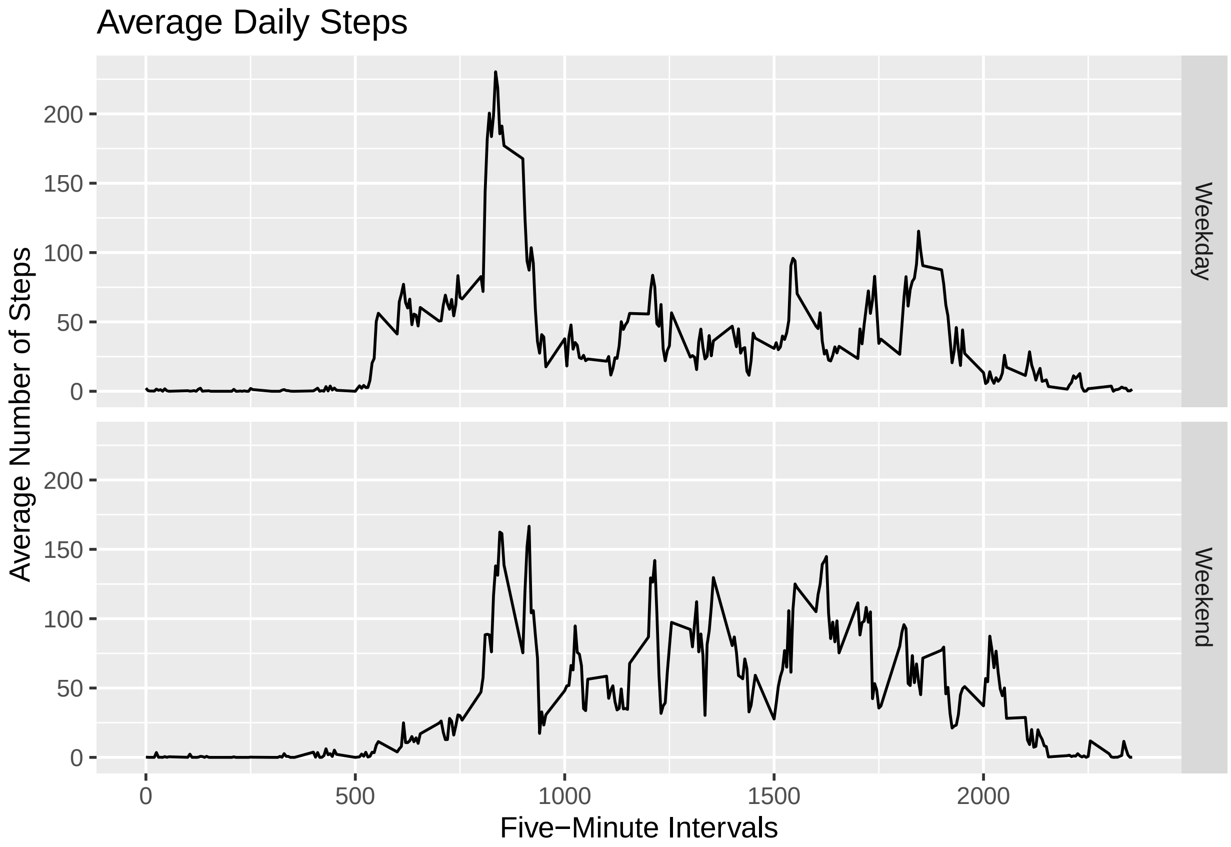
Differences in activity patterns between weekdays and weekends

New factor variable indicating weekday or weekend

```
weekPart <- function(date) {
    day <- weekdays(date)
    if (day %in% c('Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday'))
        return ("Weekday")
    else if (day %in% c('Saturday', 'Sunday'))
        return ("Weekend")
}
impStepData$date <- as.Date(impStepData$date)
impStepData$day <- sapply(impStepData$date, FUN = weekPart)
```

Panel plot of average number of steps, averaged across all weekdays and weekends

```
meanWeekPart <- aggregate(steps ~ interval + day, impStepData, mean)
g <- ggplot(data = meanWeekPart, aes(x = interval, y = steps))
g + geom_line() +
    facet_grid(rows = vars(day)) +
    ggtitle("Average Daily Steps") +
    xlab("Five-Minute Intervals") +
    ylab("Average Number of Steps")
```


Average Daily Steps
Average Number of Steps
Five−Minute Intervals
Weekday
Weekend
0
50
100
150
200
0
500
1000
1500
2000