Course Project 1

welkingliu

June 23, 2017

Introduction

It is now possible to collect a large amount of data about personal movement using activity monitoring devices such as a Fitbit, Nike Fuelband, or Jawbone Up. These type of devices are part of the "quantified self" movement - a group of enthusiasts who take measurements about themselves regularly to improve their health, to find patterns in their behavior, or because they are tech geeks. But these data remain under-utilized both because the raw data are hard to obtain and there is a lack of statistical methods and software for processing and interpreting the data.

This assignment makes use of data from a personal activity monitoring device. This device collects data at 5 minute intervals through out the day. The data consists of two months of data from an anonymous individual collected during the months of October and November, 2012 and include the number of steps taken in 5 minute intervals each day.

Loading and preprocessing the data

Show any code that is needed to

1. Load the data (i.e. read.csv())
2. Process/transform the data (if necessary) into a format suitable for your analysis

```
setwd("D:\\coursera\\proj6")
activity <- read.csv("activity.csv")

dim(activity)
```

```
## [1] 17568     3
```

```
str(activity)
```

```
## 'data.frame':    17568 obs. of  3 variables:
##  $ steps   : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ date    : Factor w/ 61 levels "2012-10-01","2012-10-02",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ interval: int  0 5 10 15 20 25 30 35 40 45 ...
```

```
head(activity)
```

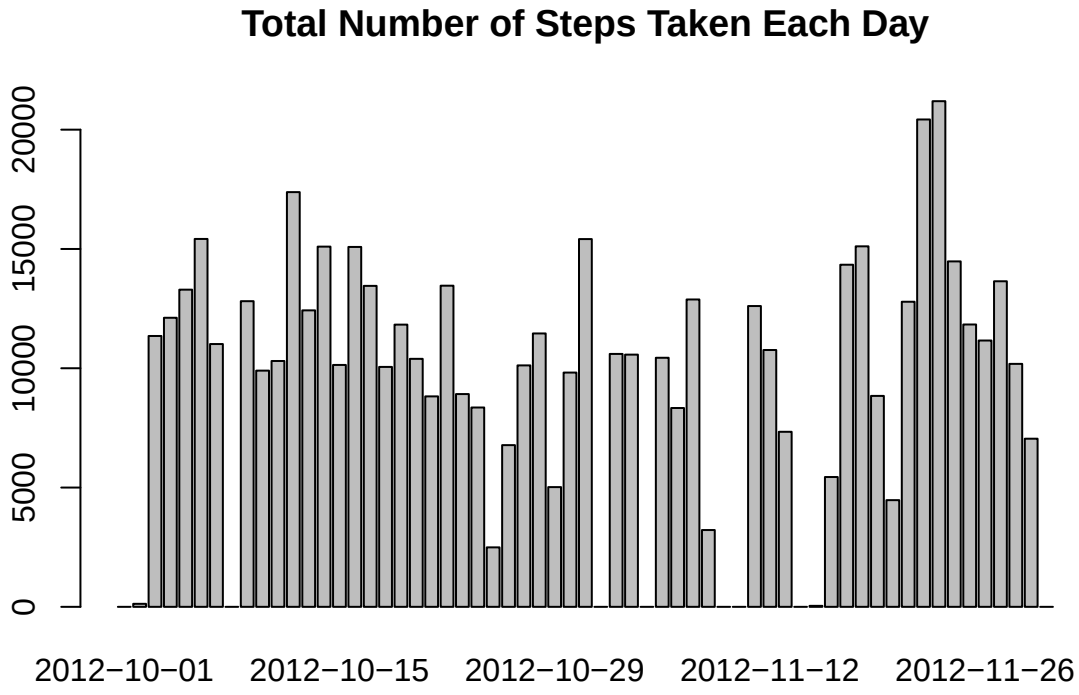
```
##   steps       date interval
## 1    NA 2012-10-01        0
## 2    NA 2012-10-01        5
## 3    NA 2012-10-01       10
## 4    NA 2012-10-01       15
## 5    NA 2012-10-01       20
## 6    NA 2012-10-01       25
```

What is mean total number of steps taken per day?

For this part of the assignment, you can ignore the missing values in the dataset.

1. Calculate the total number of steps taken per day
2. If you do not understand the difference between a histogram and a barplot, research the difference between them. Make a histogram of the total number of steps taken each day
3. Calculate and report the mean and median of the total number of steps taken per day

```
steps_day <- with(activity, tapply(steps, date, sum, na.rm=TRUE))
names(steps_day) <- as.Date(names(steps_day))
barplot(steps_day, main="Total Number of Steps Taken Each Day")
```




```
steps_day_avg <- with(activity, tapply(steps, date, mean, na.rm=TRUE))
steps_day_med <- with(activity, tapply(steps, date, median, na.rm=TRUE))
steps_day_avg
```

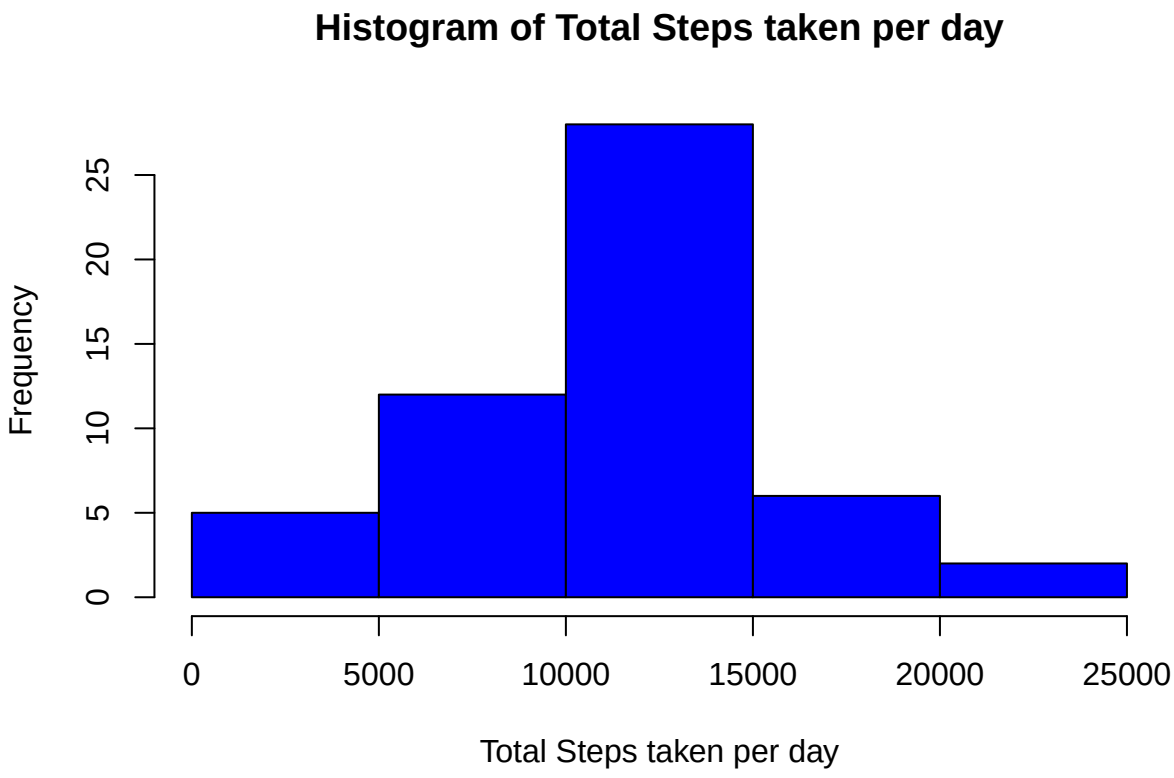
```
## 2012-10-01 2012-10-02 2012-10-03 2012-10-04 2012-10-05 2012-10-06
##        NaN  0.4375000 39.4166667 42.0694444 46.1597222 53.5416667
## 2012-10-07 2012-10-08 2012-10-09 2012-10-10 2012-10-11 2012-10-12
## 38.2465278        NaN 44.4826389 34.3750000 35.7777778 60.3541667
## 2012-10-13 2012-10-14 2012-10-15 2012-10-16 2012-10-17 2012-10-18
## 43.1458333 52.4236111 35.2048611 52.3750000 46.7083333 34.9166667
## 2012-10-19 2012-10-20 2012-10-21 2012-10-22 2012-10-23 2012-10-24
## 41.0729167 36.0937500 30.6284722 46.7361111 30.9652778 29.0104167
## 2012-10-25 2012-10-26 2012-10-27 2012-10-28 2012-10-29 2012-10-30
##  8.6527778 23.5347222 35.1354167 39.7847222 17.4236111 34.0937500
```

```
## 2012-10-31 2012-11-01 2012-11-02 2012-11-03 2012-11-04 2012-11-05
## 53.5208333        NaN 36.8055556 36.7048611        NaN 36.2465278
## 2012-11-06 2012-11-07 2012-11-08 2012-11-09 2012-11-10 2012-11-11
## 28.9375000 44.7326389 11.1770833        NaN        NaN 43.7777778
## 2012-11-12 2012-11-13 2012-11-14 2012-11-15 2012-11-16 2012-11-17
## 37.3784722 25.4722222        NaN  0.1423611 18.8923611 49.7881944
## 2012-11-18 2012-11-19 2012-11-20 2012-11-21 2012-11-22 2012-11-23
## 52.4652778 30.6979167 15.5277778 44.3993056 70.9270833 73.5902778
## 2012-11-24 2012-11-25 2012-11-26 2012-11-27 2012-11-28 2012-11-29
## 50.2708333 41.0902778 38.7569444 47.3819444 35.3576389 24.4687500
## 2012-11-30
##        NaN
```

```
steps_day_med
```

```
## 2012-10-01 2012-10-02 2012-10-03 2012-10-04 2012-10-05 2012-10-06
##         NA          0          0          0          0          0
## 2012-10-07 2012-10-08 2012-10-09 2012-10-10 2012-10-11 2012-10-12
##          0         NA          0          0          0          0
## 2012-10-13 2012-10-14 2012-10-15 2012-10-16 2012-10-17 2012-10-18
##          0          0          0          0          0          0
## 2012-10-19 2012-10-20 2012-10-21 2012-10-22 2012-10-23 2012-10-24
##          0          0          0          0          0          0
## 2012-10-25 2012-10-26 2012-10-27 2012-10-28 2012-10-29 2012-10-30
##          0          0          0          0          0          0
## 2012-10-31 2012-11-01 2012-11-02 2012-11-03 2012-11-04 2012-11-05
##          0         NA          0          0         NA          0
## 2012-11-06 2012-11-07 2012-11-08 2012-11-09 2012-11-10 2012-11-11
##          0          0          0         NA         NA          0
## 2012-11-12 2012-11-13 2012-11-14 2012-11-15 2012-11-16 2012-11-17
##          0          0         NA          0          0          0
## 2012-11-18 2012-11-19 2012-11-20 2012-11-21 2012-11-22 2012-11-23
##          0          0          0          0          0          0
## 2012-11-24 2012-11-25 2012-11-26 2012-11-27 2012-11-28 2012-11-29
##          0          0          0          0          0          0
## 2012-11-30
##         NA
```

```
totalSteps <- aggregate(steps ~ date, data = activity, sum, na.rm = TRUE)
hist(totalSteps$steps,col="blue",main="Histogram of Total Steps taken per day",
     xlab="Total Steps taken per day",cex.axis=1,cex.lab = 1)
```




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

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

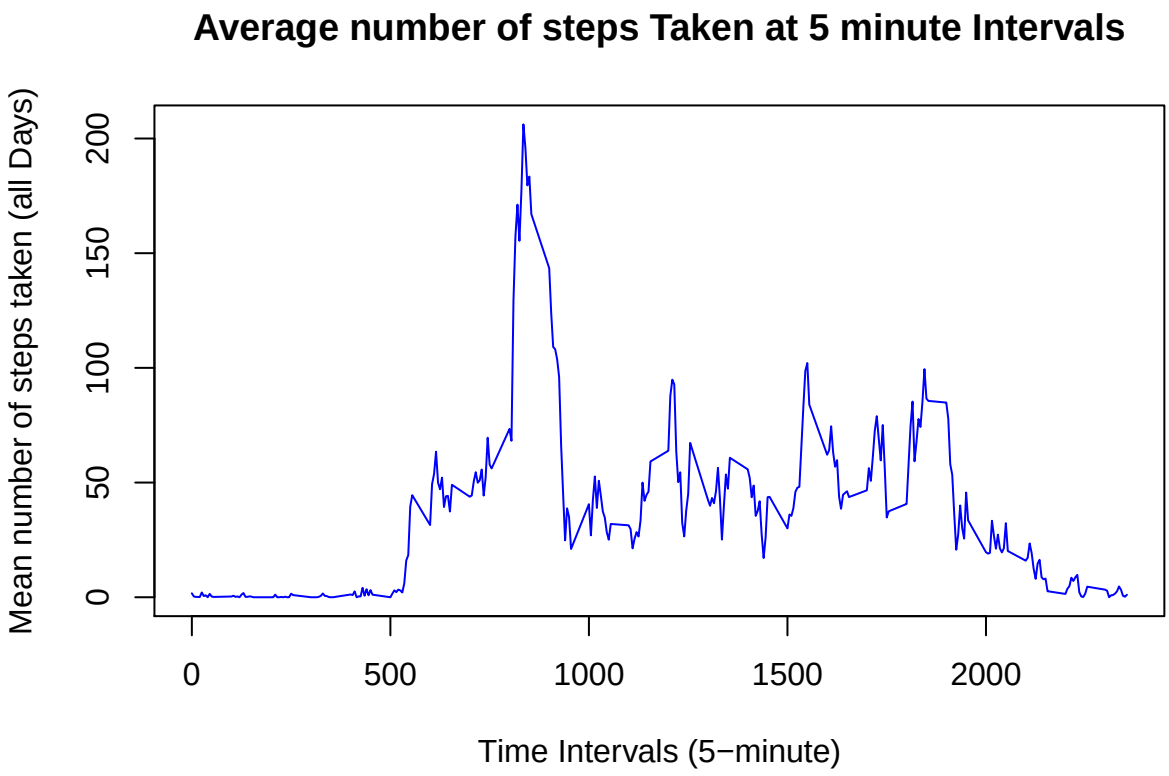
```
median_steps <- median(totalSteps$steps)
median_steps
```

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

What is the average daily activity pattern?

1. 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)
2. Which 5-minute interval, on average across all the days in the dataset, contains the maximum number of steps?

```
steps_interval <- aggregate(steps ~ interval, data = activity, mean, na.rm = TRUE)
plot(steps ~ interval, data = steps_interval, type = "l",
     xlab = "Time Intervals (5-minute)", ylab = "Mean number of steps taken (all Days)",
     main = "Average number of steps Taken at 5 minute Intervals",  col = "blue")
```




```
maxStepInterval <- steps_interval[which.max(steps_interval$steps),"interval"]
maxStepInterval
```

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

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.

1. Calculate and report the total number of missing values in the dataset (i.e. the total number of rows with NAs)
2. Devise a strategy for filling in all of the missing values in the dataset. The strategy does not need to be sophisticated. For example, you could use the mean/median for that day, or the mean for that 5-minute interval, etc.
3. Create a new dataset that is equal to the original dataset but with the missing data filled in.
4. Make a histogram of the total number of steps taken each day and Calculate and report the mean and median total number of steps taken per day. 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?

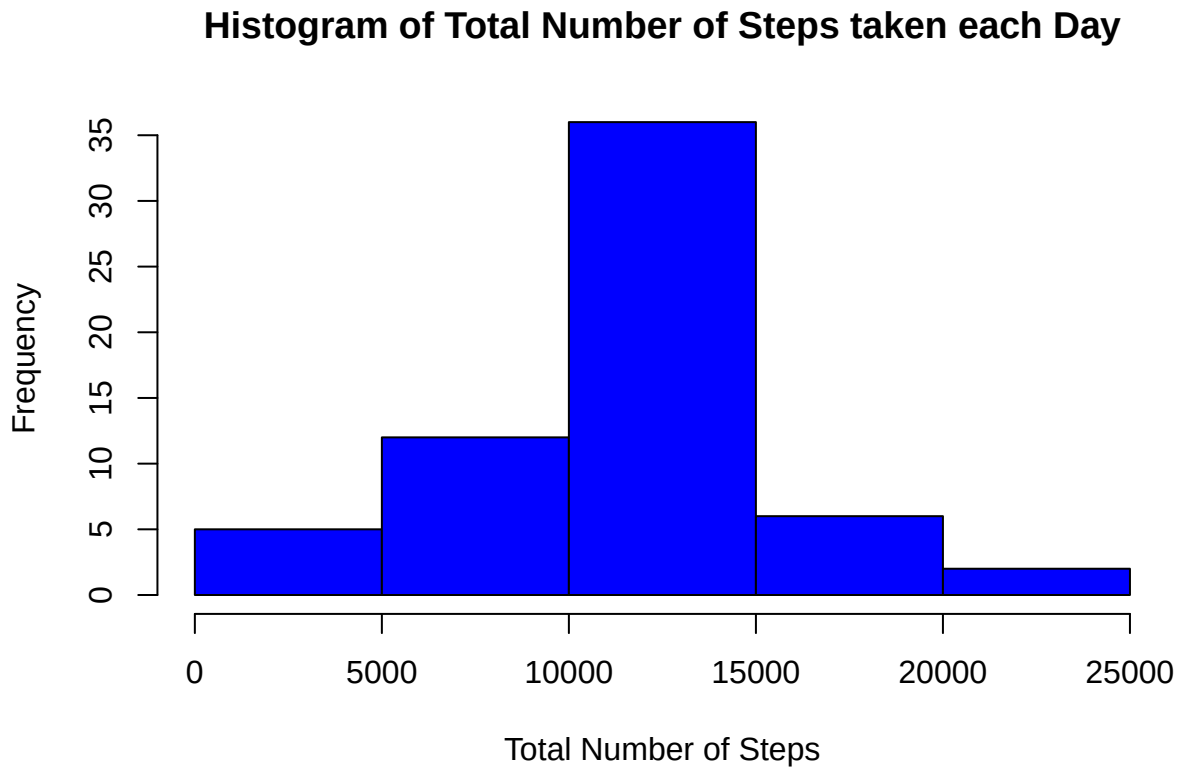
```
missing_rows <- sum(!complete.cases(activity))

getMeanStepsPerInterval <- function(interval){
    steps_interval[steps_interval$interval==interval,"steps"]
}

complete.activity <- activity
```

```
flag = 0
for (i in 1:nrow(complete.activity)) {
    if (is.na(complete.activity[i,"steps"])) {
        complete.activity[i,"steps"] <- getMeanStepsPerInterval(complete.activity[i,"interval"])
        flag = flag + 1
    }
}

total.steps.per.days <- aggregate(steps ~ date, data = complete.activity, sum)
hist(total.steps.per.days$steps, col = "blue", xlab = "Total Number of Steps",
     ylab = "Frequency", main = "Histogram of Total Number of Steps taken each Day")
```




```
showMean <- mean(total.steps.per.days$steps)
showMedian <- median(total.steps.per.days$steps)

showMean
```

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

```
showMedian
```

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

Are there differences in activity patterns between weekdays and weekends?

For this part the weekdays() function may be of some help here. Use the dataset with the filled-in missing values for this part.

1. 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.
2. 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). See the README file in the GitHub repository to see an example of what this plot should look like using simulated data.

```
complete.activity$day <- ifelse(as.POSIXlt(as.Date(complete.activity$date))$wday%%6 ==
                                  0, "weekend", "weekday")
complete.activity$day <- factor(complete.activity$day, levels = c("weekday", "weekend"))

steps.interval= aggregate(steps ~ interval + day, complete.activity, mean)
library(lattice)
xyplot(steps ~ interval | factor(day), data = steps.interval, aspect = 1/2,
       type = "l")
```