PA1.R

Ravi

Sun Jul 12 16:29:41 2015

```
## Project 1
## Loading and preprocessing the data
## setwd("E:/Cousera/Reproducible Research/Projects/Project1")
ActivityData <- read.csv("./Data/activity.csv", colClasses = c("numeric", "character",
                                                               "numeric"))
head(ActivityData)
```

```
##   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
```

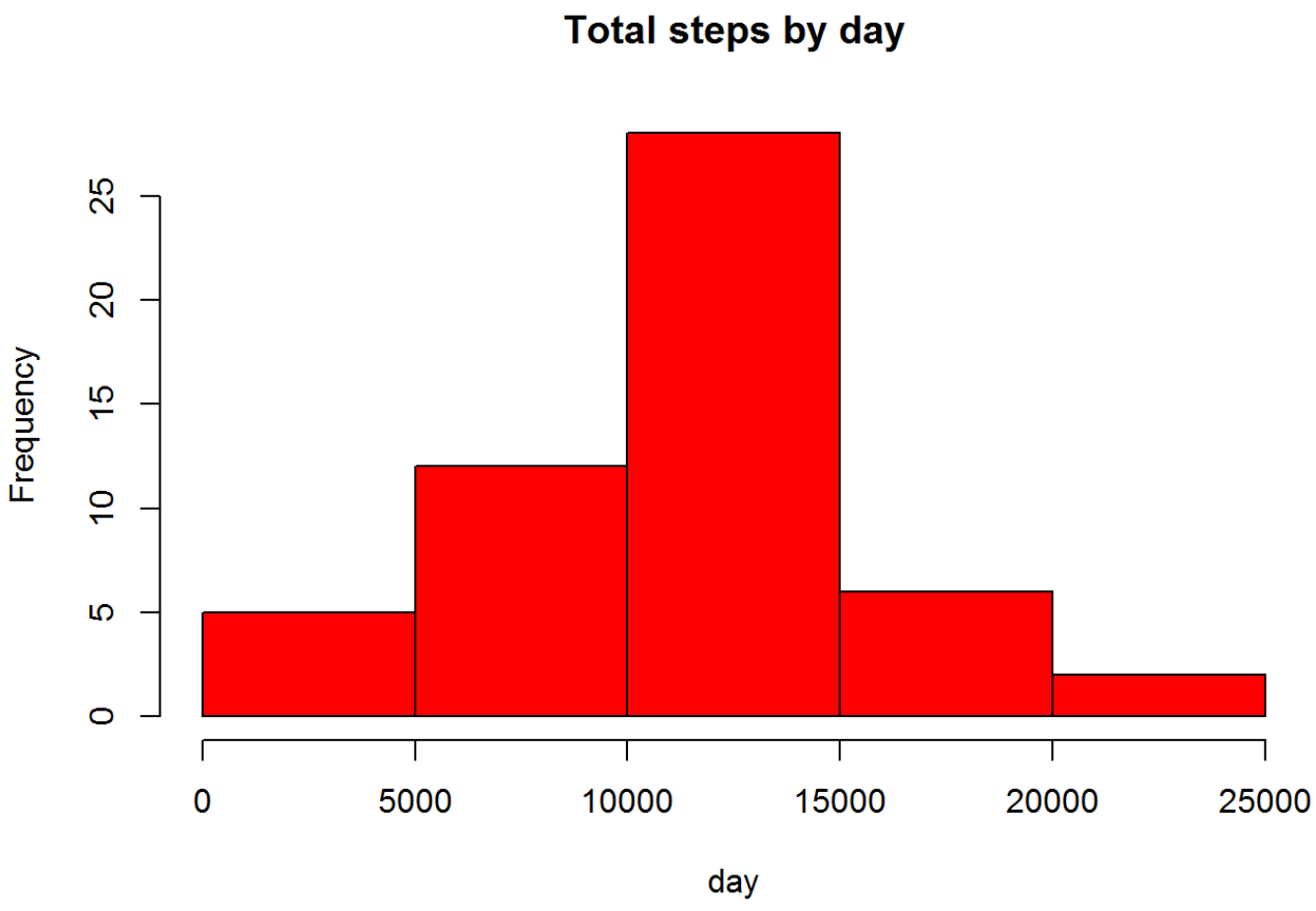
```
##   steps       date interval
## 1    NA 2012-10-01        0
## 2    NA 2012-10-01        5
## 3    NA 2012-10-01       10
## 4    NA 2012-10-01       15
## 5    NA 2012-10-01       20
## 6    NA 2012-10-01       25

names(ActivityData)
```

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

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

library(lattice)
ActivityData$date <- as.Date(ActivityData$date, "%Y-%m-%d")
head(ActivityData)
```

```
##   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?

## 1. Calculate the total number of steps taken per day: Use Aggregate function

TotalStepsPerDay <- aggregate(steps ~ date, data = ActivityData, sum, na.rm = TRUE)

## 2. Make a histogram of the total number of steps taken each day.
hist(TotalStepsPerDay$steps, main = "Total steps by day", xlab = "day", col = "red")
```



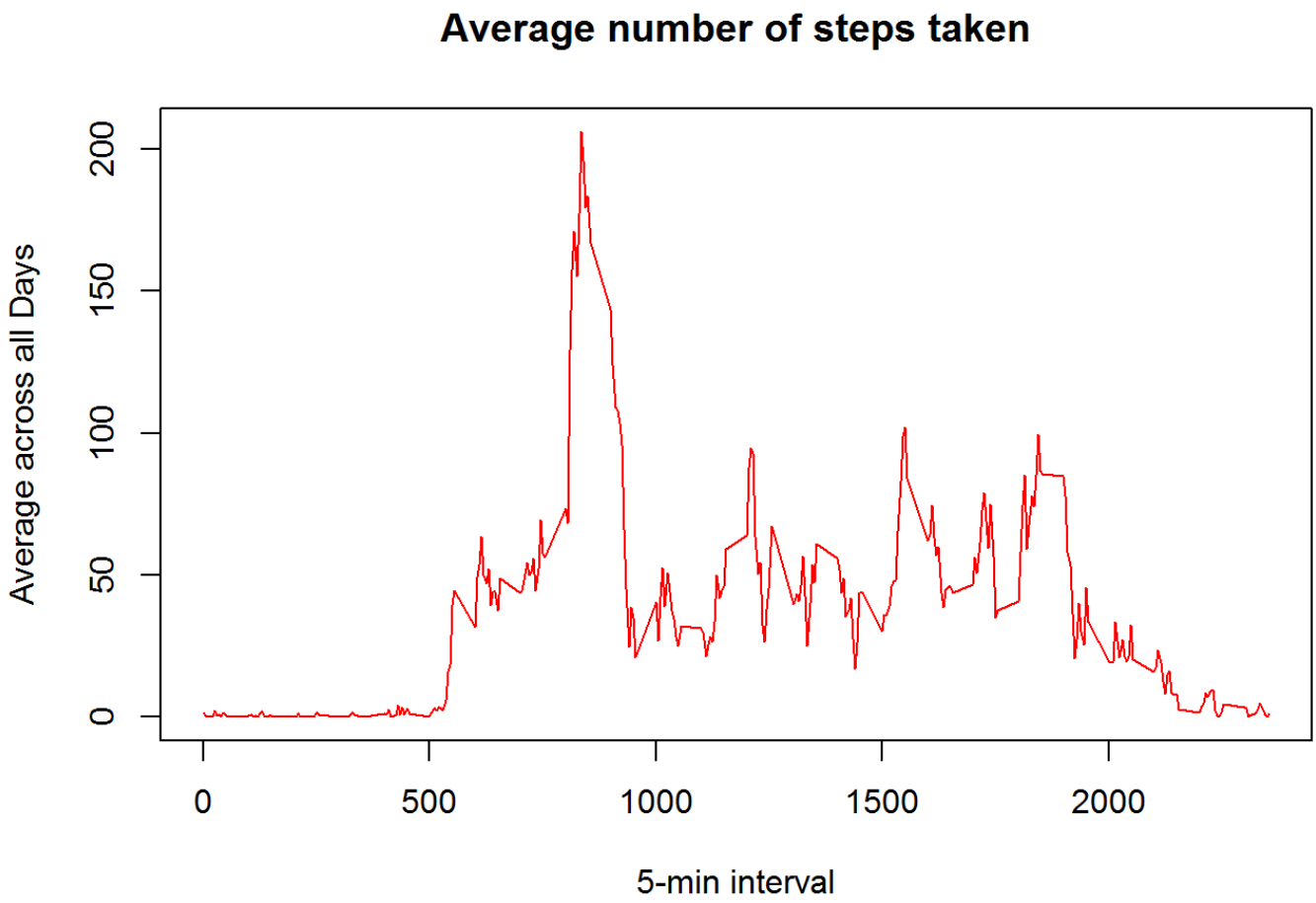

```
## 3. Calculate and report the mean and median of the total number of steps taken per day

mean(TotalStepsPerDay$steps)
```

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

```
## [1] 10766
median(TotalStepsPerDay$steps)
```

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

```
## [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)

time_series <- tapply(ActivityData$steps, ActivityData$interval, mean, na.rm = TRUE)

plot(row.names(time_series), time_series, type = "l", xlab = "5-min interval",
       ylab = "Average across all Days", main = "Average number of steps taken",
       col = "red")
```



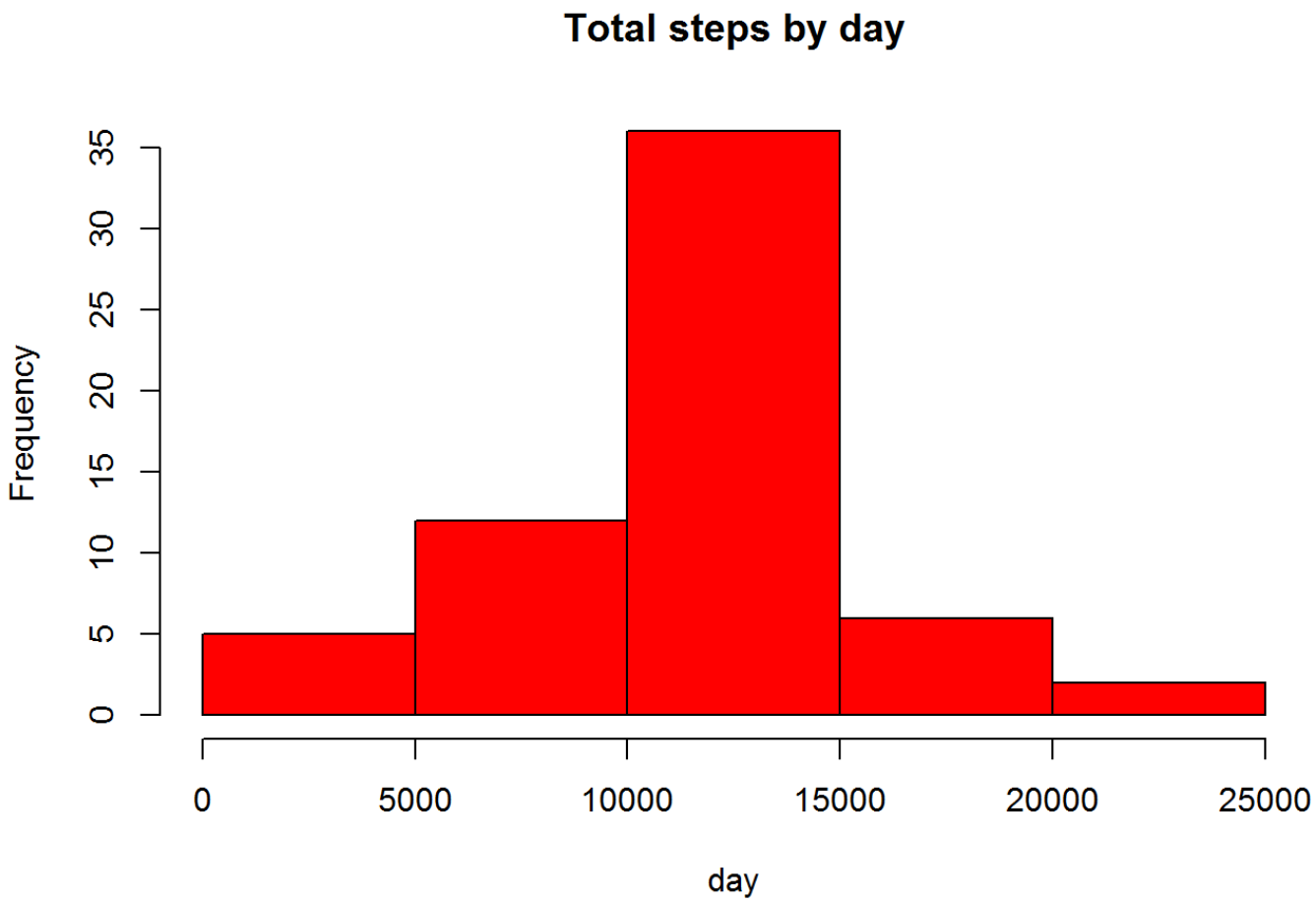
```
## 2. Which 5-minute interval, on average across all the days in the dataset,
##    contains the maximum number of steps?

max_interval <- which.max(time_series)
names(max_interval)
```

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

```
## [1] "835"

## Imputing missing values

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

ActivityData_NA <- sum(is.na(ActivityData))
ActivityData_NA
```

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

```
## [1] 2304

## 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

## NA replaced by mean in 5 min interval

StepsAverage <- aggregate(steps ~ interval, data = ActivityData, FUN = mean)
fillNA <- numeric()
for (i in 1:nrow(ActivityData)) {
  obs <- ActivityData[i, ]
  if (is.na(obs$steps)) {
    steps <- subset(StepsAverage, interval == obs$interval)$steps
  } else {
    steps <- obs$steps
  }
  fillNA <- c(fillNA, steps)
}

## 3.Create a new dataset that is equal to the original dataset
## but with the missing data filled in.

NewActivityData <- ActivityData
NewActivityData$steps <- fillNA

## 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?

StepsTotal2 <- aggregate(steps ~ date, data = NewActivityData, sum, na.rm = TRUE)

hist(StepsTotal2$steps, main = "Total steps by day", xlab = "day", col = "red")
```



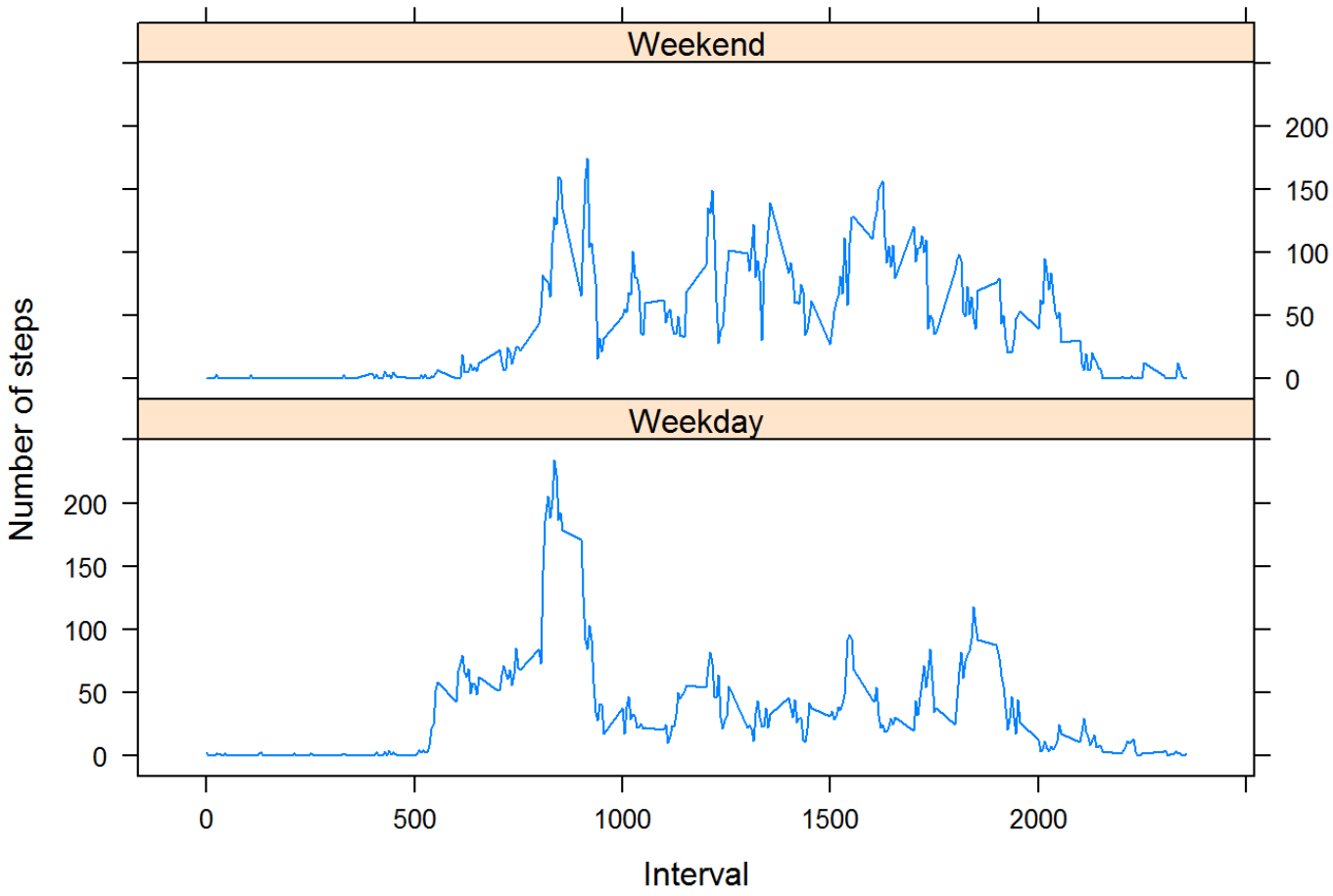

```
## mean and median

mean(StepsTotal2$steps)
```

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

```
## [1] 10766
median(StepsTotal2$steps)
```

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

```
## [1] 10766

## After replacing the mean is the same but the median is a little bit different

## 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.

day <- weekdays(ActivityData$date)
daylevel <- vector()
for (i in 1:nrow(ActivityData)) {
  if (day[i] == "Saturday") {
    daylevel[i] <- "Weekend"
  } else if (day[i] == "Sunday") {
    daylevel[i] <- "Weekend"
  } else {
    daylevel[i] <- "Weekday"
  }
}
ActivityData$daylevel <- daylevel
ActivityData$daylevel <- factor(ActivityData$daylevel)

stepsByDay <- aggregate(steps ~ interval + daylevel, data = ActivityData, mean)
names(stepsByDay) <- c("interval", "daylevel", "steps")

## 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).
## The plot should look something like the following, which was creating using simulated
data:

xyplot(steps ~ interval | daylevel, stepsByDay, type = "l", layout = c(1, 2),
         xlab = "Interval", ylab = "Number of steps")
```


Weekend
Weekday
Number of steps
Interval
0
50
100
150
200
0
500
1000
1500
2000