PA1_template

Siddharth Sachin Muthe

12/16/2019

Downloading the zip file, creating datasets and loading required libraries:

```
fileURL = "https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip"
activity = "step_data.zip"
download.file(fileURL, activity, method="curl")
unzip(activity)
activity <- read.csv("/Users/home/Downloads/activity.csv", sep = ",")
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date
```

```
library(ggplot2)
```

1. What is mean total number of steps taken per day?:
    i. Total number of steps per day

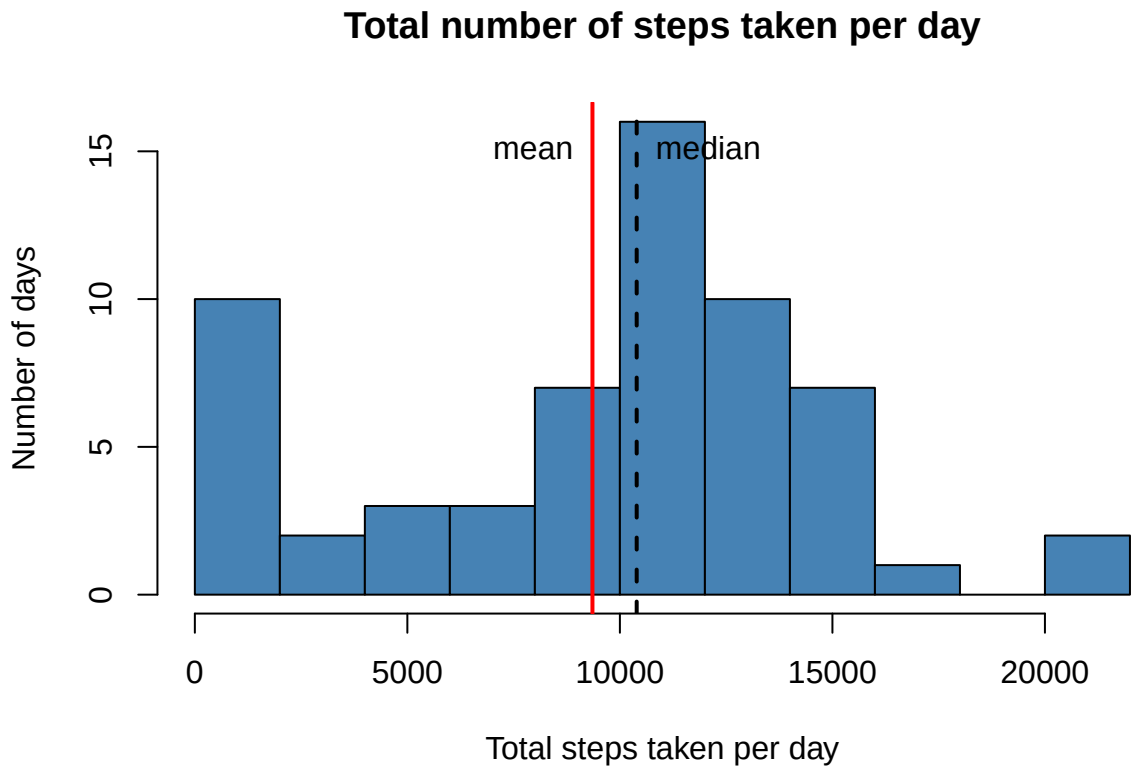
```
total_steps_per_day <- activity %>%
                       group_by(date) %>%
                       summarise(total_steps=sum(steps, na.rm = TRUE))
head(total_steps_per_day)
```

```
## # A tibble: 6 x 2
##   date       total_steps
##   <fct>            <int>
## 1 2012-10-01           0
```

```
## 2 2012-10-02          126
## 3 2012-10-03        11352
## 4 2012-10-04        12116
## 5 2012-10-05        13294
## 6 2012-10-06        15420
```

ii. Histogram of the total number of steps taken each day

```
hist(x=total_steps_per_day$total_steps, main = "Total number of steps taken per day",
     xlab = "Total steps taken per day", ylab = "Number of days",
     breaks = 10, col = "steel blue")
abline(v = mean(total_steps_per_day$total_steps, na.rm = TRUE), lty = 1, lwd = 2, col = "red")
abline(v = median(total_steps_per_day$total_steps, na.rm = TRUE), lty = 2, lwd = 2, col = "black")
text(y = 15, x = mean(total_steps_per_day$total_steps, na.rm = TRUE), pos=2, labels = "mean")
text(y = 15, x = median(total_steps_per_day$total_steps, na.rm = TRUE), pos=4, labels = "median")
```



iii. Mean and median of the total number of steps taken per day

```
mean_of_steps_per_day = mean(total_steps_per_day$total_steps, na.rm = TRUE)
mean_of_steps_per_day
```

```
## [1] 9354.23
```

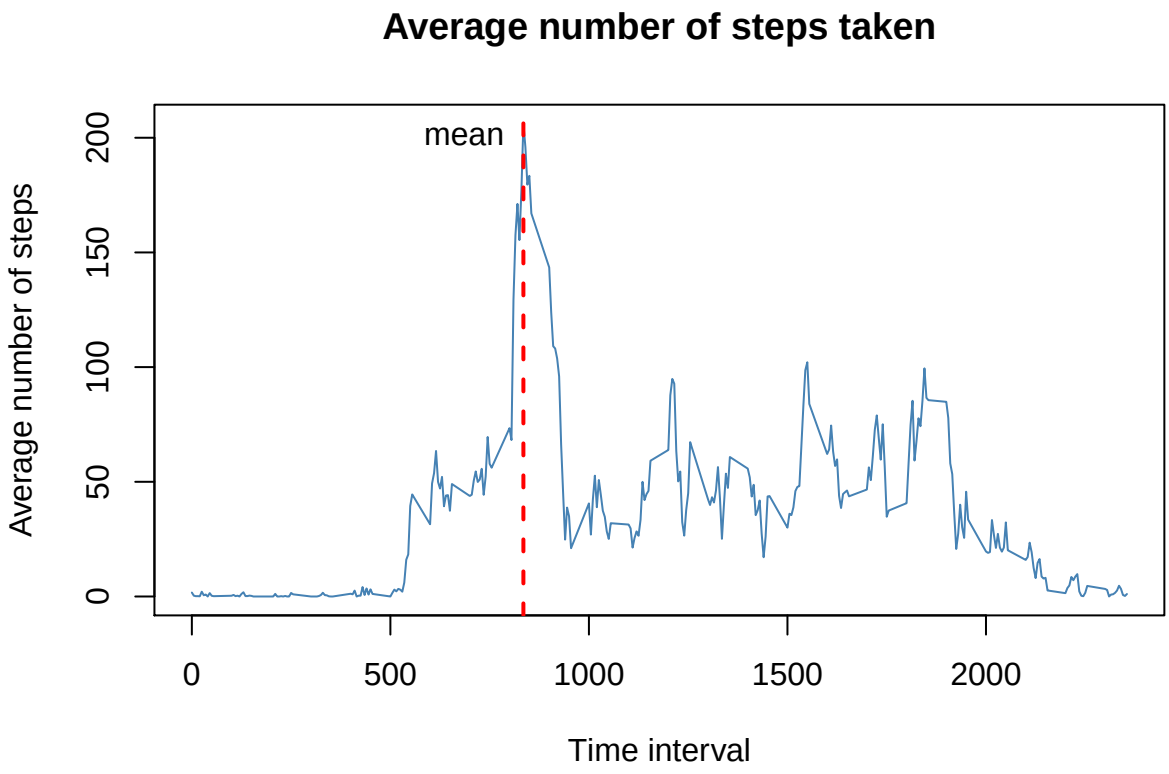
```
median_of_steps_per_day = median(total_steps_per_day$total_steps, na.rm = TRUE)
median_of_steps_per_day
```

```
## [1] 10395
```

2. What is the average daily activity pattern?:

i. Average daily activity pattern plot

```
mean_steps_over_interval <- activity %>%
                            select(interval, steps) %>%
                            group_by(interval) %>%
                            summarise(mean_steps = mean(steps, na.rm = TRUE))

plot(mean_steps_over_interval$interval, mean_steps_over_interval$mean_steps, ty = "l", col = "steel blu
     xlab = "Time interval", ylab = "Average number of steps",
     main = "Average number of steps taken")
abline(v = mean_steps_over_interval$interval[which.max(mean_steps_over_interval$mean_steps)], lty = 2,
text(y = 200, x = mean_steps_over_interval$interval[which.max(mean_steps_over_interval$mean_steps)], pos
```



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

```
mean_steps_over_interval$interval[which.max(mean_steps_over_interval$mean_steps)]
```

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

3. Imputing missing values:

    i. Total number of missing values

```
sum(is.na(activity))
```

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

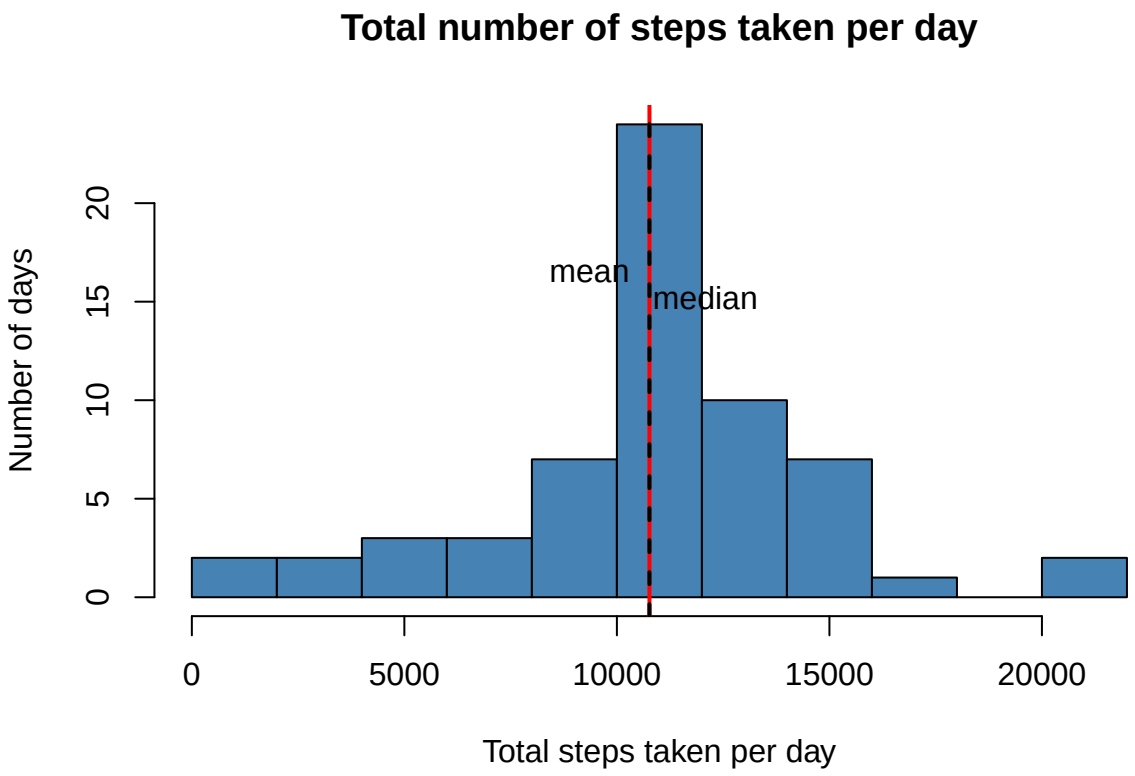
ii. Replacing missing values and creating a new dataset

```
replace_na <- split(activity, activity$interval)
replace_na <- lapply(replace_na, function(x) {
  x$steps[which(is.na(x$steps))] <- mean(x$steps, na.rm = TRUE)
  return(x)
})
replace_na <- do.call("rbind", replace_na)
row.names(replace_na) <- NULL

replace_na <- split(replace_na, replace_na$date)
df <- lapply(replace_na, function(x) {
  x$steps[which(is.na(x$steps))] <- mean(x$steps, na.rm = TRUE)
  return(x)
})
replace_na <- do.call("rbind", replace_na)
row.names(replace_na) <- NULL
head(replace_na)
```

```
##       steps       date interval
## 1 1.7169811 2012-10-01        0
## 2 0.3396226 2012-10-01        5
## 3 0.1320755 2012-10-01       10
## 4 0.1509434 2012-10-01       15
## 5 0.0754717 2012-10-01       20
## 6 2.0943396 2012-10-01       25
```

iii. Histogram of total number of steps taken after replacing missing values

```
replace_na_plot <- replace_na %>%
  group_by(date) %>%
  summarise(total_steps=sum(steps, na.rm = TRUE))

hist(x=replace_na_plot$total_steps, main = "Total number of steps taken per day",
     xlab = "Total steps taken per day", ylab = "Number of days",
     breaks = 10, col = "steel blue")
abline(v = mean(replace_na_plot$total_steps, na.rm = TRUE), lty = 1, lwd = 2, col = "red")
abline(v = median(replace_na_plot$total_steps, na.rm = TRUE), lty = 2, lwd = 2, col = "black")
text(y = 15, x = mean(total_steps_per_day$total_steps, na.rm = TRUE), pos=3, labels = "mean")
text(y = 15, x = median(total_steps_per_day$total_steps, na.rm = TRUE), pos=4, labels = "median")
```




```
mean(replace_na_plot$total_steps, na.rm = TRUE)
```

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

```
median(replace_na_plot$total_steps, na.rm = TRUE)
```

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

iv. Difference in mean and median before and after replacing missing values

```
difference_in_means = mean(replace_na_plot$total_steps, na.rm = TRUE) - mean(total_steps_per_day$total_s
difference_in_means
```

```
## [1] 1411.959
```

```
difference_in_median = median(replace_na_plot$total_steps, na.rm = TRUE) - median(total_steps_per_day$to
difference_in_median
```

```
## [1] 371.1887
```

4. Are there differences in activity patterns between weekdays and weekends?:

```
replace_na$day <- weekdays(as.Date(replace_na$date))
replace_na$dayofweek <- factor(replace_na$day, levels = c('Sunday', 'Monday', 'Tuesday', 'Wednesday', '
                       labels = c('Weekends', 'Weekdays', 'Weekdays', 'Weekdays', 'Weekdays', 'Weekdays
dayofweek_plot <- replace_na %>%
                  group_by(dayofweek, interval) %>%
                  summarise(mean_steps = mean(steps, na.rm = TRUE))

ggplot(dayofweek_plot, aes(interval, mean_steps)) +
  geom_line() +
  facet_grid(dayofweek ~ .) +
  xlab("Time interval") +
  ylab("Number of steps taken") +
  ggtitle("Average number of steps taken") +
  theme(plot.title = element_text(face="bold", color="black",size=22, hjust=0.5))
```