Reproducible Research Project 1

Tonya MacDonald

2/15/2021

Load Data

Download and unzip data, then load into data table

```
library("data.table")
library(ggplot2)

fileUrl <- "https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip"

download.file(fileUrl, destfile = paste0(getwd(), "/repdata%2Fdata%2Factivity.zip"), method = "curl")

unzip("repdata%2Fdata%2Factivity.zip",exdir = "data")

activitydata <- data.table::fread(input = "data/activity.csv")
```

What is mean total number of steps taken per day?

1. Calculate the total number of steps taken per day

```
# sum the total steps
totalsteps <- as.data.table(setNames(aggregate(activitydata$steps, by=list(activitydata$date), FUN=sum), c("date"
,"steps")))
```

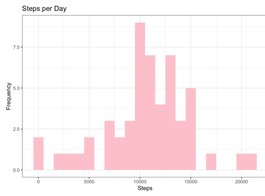
```
# take a look at the first 10 rows
head(totalsteps, 10)
```

```
##          date steps
##  1: 2012-10-01    NA
##  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
##  7: 2012-10-07 11015
##  8: 2012-10-08    NA
##  9: 2012-10-09 12811
## 10: 2012-10-10  9900
```

2. Make a histogram of the total number of steps per day

```
# histogram
ggplot(totalsteps, aes(x = steps)) +
    geom_histogram(fill = "pink", binwidth = 1000) +
    labs(title = "Steps per Day", x = "Steps", y = "Frequency") +
    theme_bw()
```

```
## Warning: Removed 8 rows containing non-finite values (stat_bin).
```



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

```
# average steps
meansteps <- mean(totalsteps$steps, na.rm=TRUE)
meansteps
```

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

```
# median step value
mediansteps <- median(totalsteps$steps, na.rm=TRUE)
mediansteps
```

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

What is the average daily activity pattern?

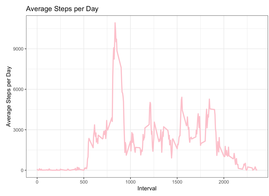
1. Make a time series plot (i.e. `type = "l"`) of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis)

```
# find all the intervals
intervals <- as.data.table(setNames(aggregate(activitydata$steps, by=list(activitydata$interval), FUN=sum, na.rm=
TRUE), c("interval","steps")))

# show the data
head(intervals,10)
```

```
##    interval steps
##  1:        0    91
##  2:        5    18
##  3:       10     7
##  4:       15     8
##  5:       20     4
##  6:       25   111
##  7:       30    28
##  8:       35    46
##  9:       40     0
## 10:       45    78
```

```
# line chart
ggplot(intervals, aes(x = interval , y = steps)) +
  geom_line(color="pink", size=1) +
  labs(title = "Average Steps per Day", x = "Interval", y = "Average Steps per Day") +
  theme_bw()
```



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

```
# find the interval with the max steps
intervals$steps == max(steps)$interval
```

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

Imputing missing values

1. Calculate and report the total number of missing values in the data

```
#number of rows with NAs
nrow(activitydata[is.na(steps),])
```

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

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

```
# replace NAs with the median
activitydata[is.na(steps), "steps"] <- activitydata[, c(lapply(.SD, median, na.rm = TRUE)), .SDcols = c("steps")]

# verify no more NAs
nrow(activitydata[is.na(steps),])
```

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

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?

```
# sum total steps again, this time with the NAs replaced with the median
totalsteps2 <- as.data.table(setNames(aggregate(activitydata$steps, by=list(activitydata$date), FUN=sum), c("date
","steps")))

# view data
head(totalsteps2)
```

```
##          date steps
## 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
```

```
# average the total steps with NAs removed
meansteps2 <- mean(totalsteps2$steps)
meansteps2
```

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

```
# median should still be the same
mediansteps2 <- median(totalsteps2$steps)
mediansteps2
```

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

```
# histogram
ggplot(totalsteps, aes(x = steps)) +
    geom_histogram(fill = "pink", binwidth = 1000) +
    labs(title = "Steps per Day", x = "Steps", y = "Frequency") +
    theme_bw()
```

```
## Warning: Removed 8 rows containing non-finite values (stat_bin).
```

Compare the mean and median before and after NA removal

```
meansteps
```

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

```
meansteps2
```

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

```
mediansteps
```

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

```
mediansteps2
```

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

Are there differences in activity patterns between weekdays and weekends?

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.

```
#classify dates as weekdays or weekends
activitydata[, `Day of Week`:= weekdays(x = date)]
activitydata[grepl(pattern = "Monday|Tuesday|Wednesday|Thursday|Friday", x = `Day of Week`), "weekday or weekend"
] <- "weekday"
activitydata[grepl(pattern = "Saturday|Sunday", x = `Day of Week`), "weekday or weekend"] <- "weekend"
activitydata[, `weekday or weekend` := as.factor(`weekday or weekend`)]
head(activitydata, 10)
```

```
##     steps       date interval Day of Week weekday or weekend
##  1:     0 2012-10-01        0      Monday             weekday
##  2:     0 2012-10-01        5      Monday             weekday
##  3:     0 2012-10-01       10      Monday             weekday
##  4:     0 2012-10-01       15      Monday             weekday
##  5:     0 2012-10-01       20      Monday             weekday
##  6:     0 2012-10-01       25      Monday             weekday
##  7:     0 2012-10-01       30      Monday             weekday
##  8:     0 2012-10-01       35      Monday             weekday
##  9:     0 2012-10-01       40      Monday             weekday
## 10:     0 2012-10-01       45      Monday             weekday
```

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.

```
activitydata[is.na(steps), "steps"] <- activitydata[, c(lapply(.SD, median, na.rm = TRUE)), .SDcols = c("steps")]
IntervalDT <- activitydata[, c(lapply(.SD, mean, na.rm = TRUE)), .SDcols = c("steps"), by = .(interval, `weekday
or weekend`)]

ggplot(IntervalDT , aes(x = interval , y = steps, color=`weekday or weekend`)) + geom_line() + labs(title = "Aver
age Steps per Day", x = "Interval", y = "Steps") + facet_wrap(~`weekday or weekend` , ncol = 1, nrow=2)
```