Course Project 1

Jie Tao

8/27/2020

Loading and preprocessing the data

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.1
## v tidyr   1.1.1     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts ------------------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
download.file(url = "https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip",
              destfile = "E:/Data Science Specialization/Reproducible Research/Course Project 1/activity

activity <- read_csv(file = unz(description = "E:/Data Science Specialization/Reproducible Research/Cou
                                filename = "activity.csv"), col_names = TRUE )
```

```
## Parsed with column specification:
## cols(
##   steps = col_double(),
##   date = col_date(format = ""),
##   interval = col_double()
## )
```

```
str(activity)
```

```
## tibble [17,568 x 3] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ steps   : num [1:17568] NA NA NA NA NA NA NA NA NA NA ...
##  $ date    : Date[1:17568], format: "2012-10-01" "2012-10-01" ...
##  $ interval: num [1:17568] 0 5 10 15 20 25 30 35 40 45 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   steps = col_double(),
##   ..   date = col_date(format = ""),
##   ..   interval = col_double()
##   .. )
```

```
head(activity)
```

```
## # A tibble: 6 x 3
##   steps date       interval
##   <dbl> <date>        <dbl>
## 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
```

```
tail(activity)
```

```
## # A tibble: 6 x 3
##   steps date       interval
##   <dbl> <date>        <dbl>
## 1    NA 2012-11-30     2330
## 2    NA 2012-11-30     2335
## 3    NA 2012-11-30     2340
## 4    NA 2012-11-30     2345
## 5    NA 2012-11-30     2350
## 6    NA 2012-11-30     2355
```

What is mean total number of steps taken per day?

```
# total number of steps taken per day
library(lubridate)
```

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

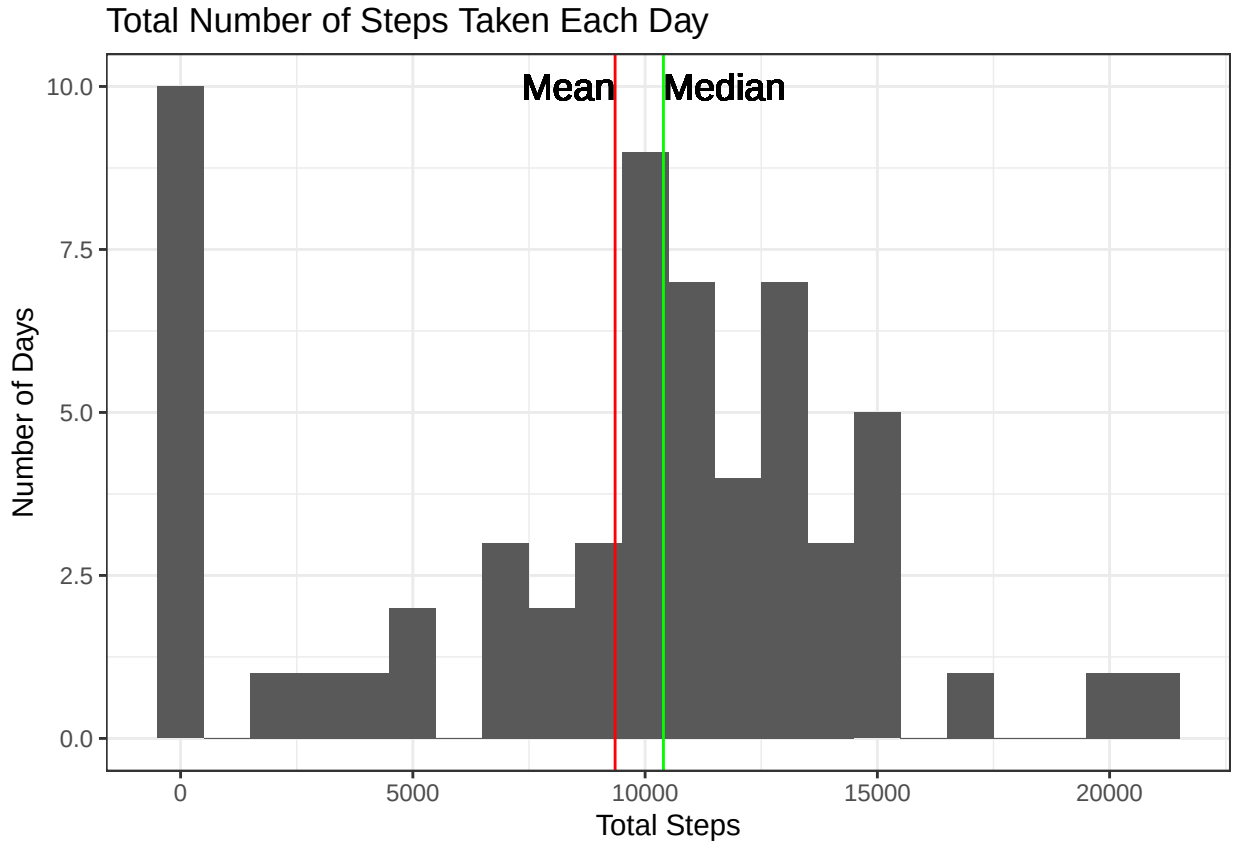
```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
total_steps_pd <- activity %>%
    mutate(year = year(date), month = month(date), mday = mday(date)) %>%
    group_by(year, month, mday) %>%
    summarize(total_steps_pd = sum(steps, na.rm = TRUE))
```

```
## `summarise()` regrouping output by 'year', 'month' (override with `.groups` argument)
```

```
# histogram of the total number of steps taken per day
total_steps_pd %>%
    ggplot(aes(total_steps_pd)) +
    geom_histogram(binwidth = 1000) +
    geom_vline(aes(xintercept = mean(total_steps_pd)), color = "red") +
```

```
    geom_vline(aes(xintercept = median(total_steps_pd)), color = "green") +
    geom_text(aes(x = mean(total_steps_pd),
                  y = 10), label = "Mean", hjust = 1, size = 5) +
    geom_text(aes(x = median(total_steps_pd),
                  y = 10), label = "Median", hjust = 0, size = 5) +
    labs(x = "Total Steps",
         y = "Number of Days",
         title = "Total Number of Steps Taken Each Day") +
    theme_bw()
```




```
# mean and median of the total number of steps taken per day
mean(total_steps_pd$total_steps_pd)
```

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

```
median(total_steps_pd$total_steps_pd)
```

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

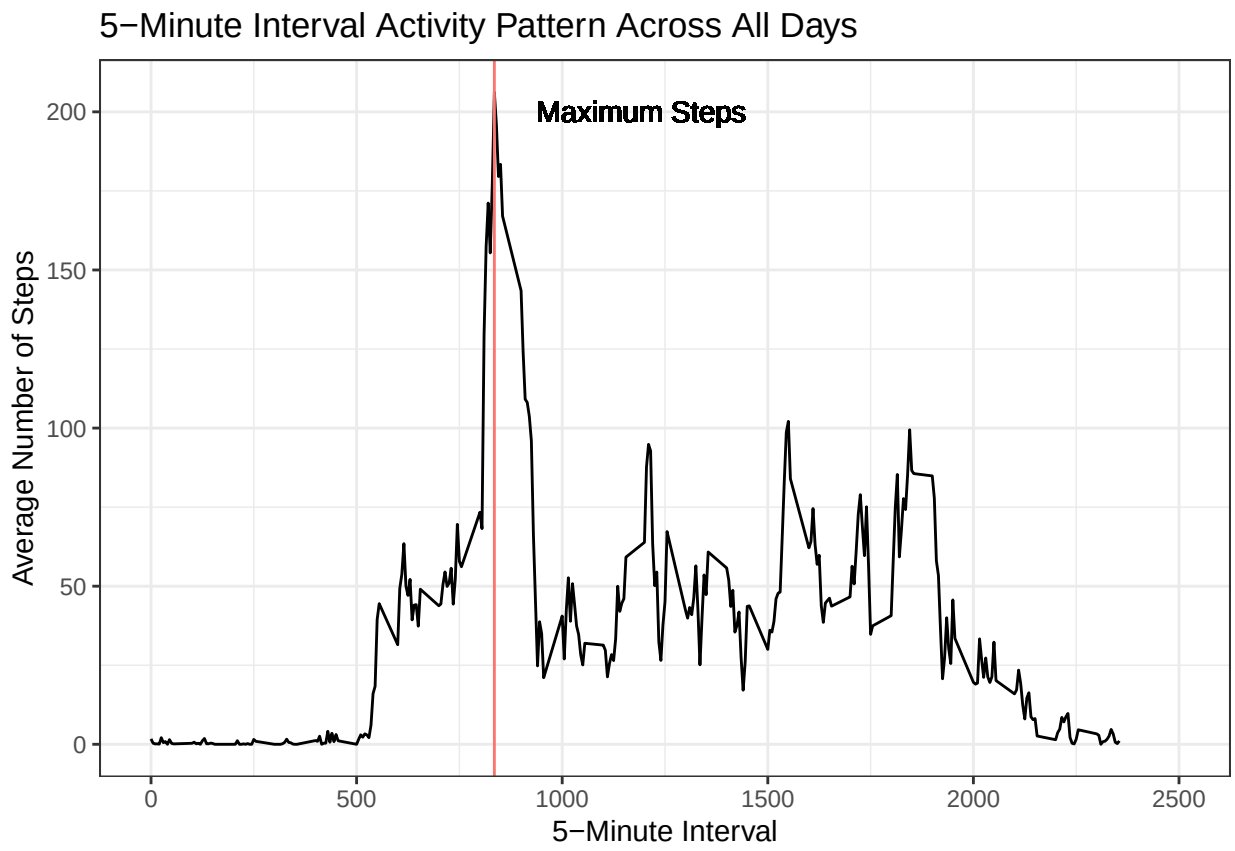
What is the average daily activity pattern?

```r
# average steps taken by 5-minute intervals
avg_steps_by_int_day <- activity %>%
    mutate(year = year(date), month = month(date), mday = mday(date)) %>%
    group_by(interval) %>%
    summarize(avg_sbid = mean(steps, na.rm = TRUE))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
# time series plot
avg_steps_by_int_day %>%
    ggplot(aes(interval, avg_sbid)) +
    geom_line() +
    geom_vline(aes(xintercept = 835, color = "blue")) +
    geom_text(aes(x = 835, y = 200), label = "Maximum Steps", hjust = -0.2, size = 4) +
    labs(x = "5-Minute Interval",
         y = "Average Number of Steps",
         title = "5-Minute Interval Activity Pattern Across All Days") +
    scale_x_continuous(breaks = c(0, 500, 1000, 1500, 2000, 2500)) +
    xlim(0, 2500) +
    theme_bw() +
    theme(legend.position = "none")
```

```
## Scale for 'x' is already present. Adding another scale for 'x', which will
## replace the existing scale.
```



```r
# which 5-minute interval has the maximum number of steps across all days
avg_steps_by_int_day_ranked <- avg_steps_by_int_day %>%
    arrange(desc(avg_sbid))
avg_steps_by_int_day_ranked[1, ]
```

```
## # A tibble: 1 x 2
##   interval avg_sbid
##      <dbl>    <dbl>
## 1      835     206.
```

Imputing missing value

```r
# total number of missing values in the dataset
sum(is.na(activity))
```

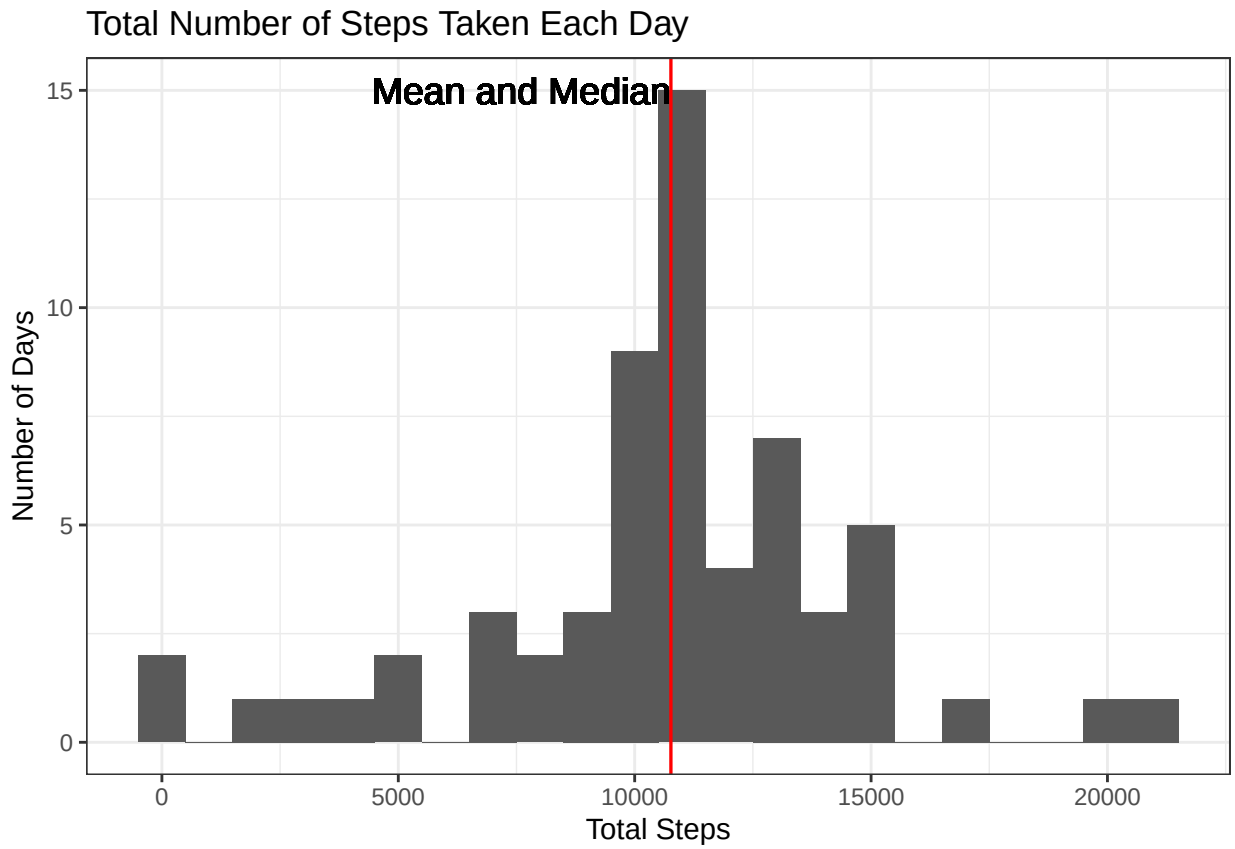
```
## [1] 2304
```

```r
# imputation strategy (I use mean for the 5-minute interval to impute)
activity1 <- activity
activity1$all_steps <-
    ifelse(is.na(activity$steps), avg_steps_by_int_day$avg_sbid[match(activity$interval, avg_steps_by_i

total_steps_pd1 <- activity1 %>%
    mutate(year = year(date), month = month(date), mday = mday(date)) %>%
    group_by(year, month, mday) %>%
    summarize(total_steps_pd = sum(all_steps, na.rm = TRUE))
```

```
## `summarise()` regrouping output by 'year', 'month' (override with `.groups` argument)
```

```r
# histogram of the total number of steps taken per day
total_steps_pd1 %>%
    ggplot(aes(total_steps_pd)) +
    geom_histogram(binwidth = 1000) +
    geom_vline(aes(xintercept = mean(total_steps_pd1$total_steps_pd)), color = "red") +
    geom_vline(aes(xintercept = median(total_steps_pd1$total_steps_pd)), color = "red") +
    geom_text(aes(x = mean(total_steps_pd1$total_steps_pd),
                  y = 15), label = "Mean and Median", hjust = 1, size = 5) +
    labs(x = "Total Steps",
         y = "Number of Days",
         title = "Total Number of Steps Taken Each Day") +
    theme_bw()
```




```
# meand and median
mean(total_steps_pd1$total_steps_pd)
```

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

```
median(total_steps_pd1$total_steps_pd)
```

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

Are there differences in activity patterns between weekdays and weekends?

```
#create factor variable
activity1 <- activity1 %>%
    mutate(weekday = weekdays(date), wd_we = as_factor(case_when(weekday == "Monday" |
                                                                 weekday == "Tuesday" |
                                                                 weekday == "Wednesday" |
                                                                 weekday == "Thursday" |
                                                                 weekday == "Friday" ~ "weekday",
                                                                 weekday == "Saturday" |
                                                                 weekday == "Sunday" ~ "weekend")))

# group by 5-minute interval and factor variable
```

```r
avg_steps_wdwe <- activity1 %>%
    group_by(interval, wd_we) %>%
    summarize(avg_steps_by_int_wdwe = mean(all_steps, na.rm = TRUE))
```

```
## `summarise()` regrouping output by 'interval' (override with `.groups` argument)
```

```r
# making a panel plot
avg_steps_wdwe %>%
    ggplot(aes(interval, avg_steps_by_int_wdwe, color = wd_we)) +
    geom_line() +
    facet_wrap(vars(wd_we), nrow = 2) +
    labs(x = "5-Minute Interval",
         y = "Average Number of Steps",
         title = "5-Minute Interval Activity Pattern Across All Weekdays or Weekends") +
    scale_x_continuous(breaks = c(0, 500, 1000, 1500, 2000, 2500)) +
    scale_color_discrete(name = "Day in the Week") +
    xlim(0, 2500) +
    theme_bw() +
    theme(legend.position = "none")
```

```
## Scale for 'x' is already present. Adding another scale for 'x', which will
## replace the existing scale.
```