Project1_TMW

Tianming

1/16/2021

```
if(!file.exists("./data")){dir.create("./data")}
fileUrL <- "https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip"
download.file(fileUrL, destfile = "./data/Activity_monitoring_data.zip")
unzip(zipfile = "./data/Activity_monitoring_data.zip", exdir = "./data/")
```

Loading and preprocessing the data

```
activity <- read.csv("./data/activity.csv", header = T)

## check which column(s) contains NA value
list_na <- colnames(activity)[apply(activity, 2, anyNA)]
list_na
```

```
## [1] "steps"
```

```
## remove NA from data frame
activityClean <- activity[complete.cases(activity),]

## check data summary
summary(activityClean)
```

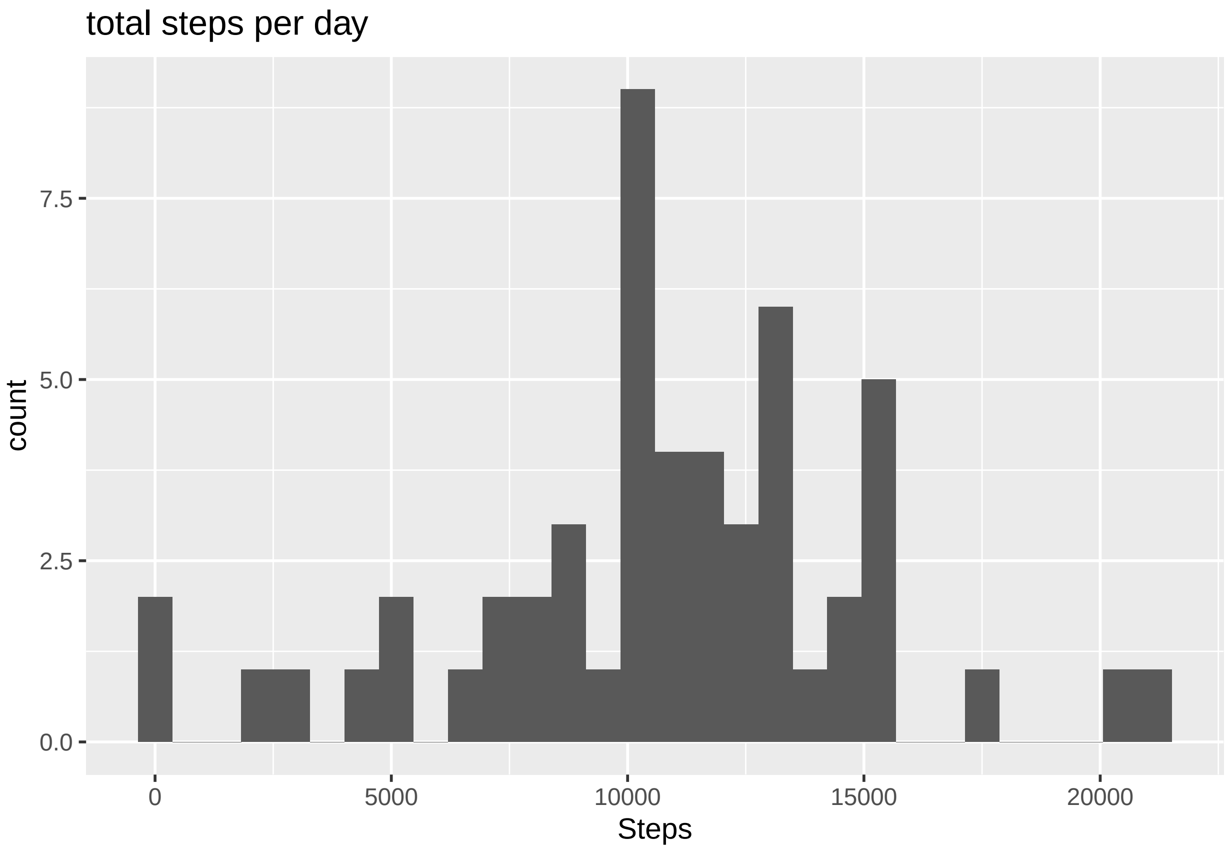
```
##      steps                date          interval
##  Min.   :  0.00   2012-10-02:  288   Min.   :   0.0
##  1st Qu.:  0.00   2012-10-03:  288   1st Qu.: 588.8
##  Median :  0.00   2012-10-04:  288   Median :1177.5
##  Mean   : 37.38   2012-10-05:  288   Mean   :1177.5
##  3rd Qu.: 12.00   2012-10-06:  288   3rd Qu.:1766.2
##  Max.   :806.00   2012-10-07:  288   Max.   :2355.0
##                   (Other)   :13536
```

What is mean total number of steps taken per day?

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

```
## histogram of total number of steps taken each day
StepTable <- aggregate(activityClean$steps, by = list(activityClean$date), sum)
colnames(StepTable) <- c("Date", "TotalStep")

library(ggplot2)
ggplot(StepTable, aes(TotalStep)) + geom_histogram() + labs(x = "Steps", title = "total steps per day")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



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

```
mean(activityClean$steps)
```

```
## [1] 37.3826
```

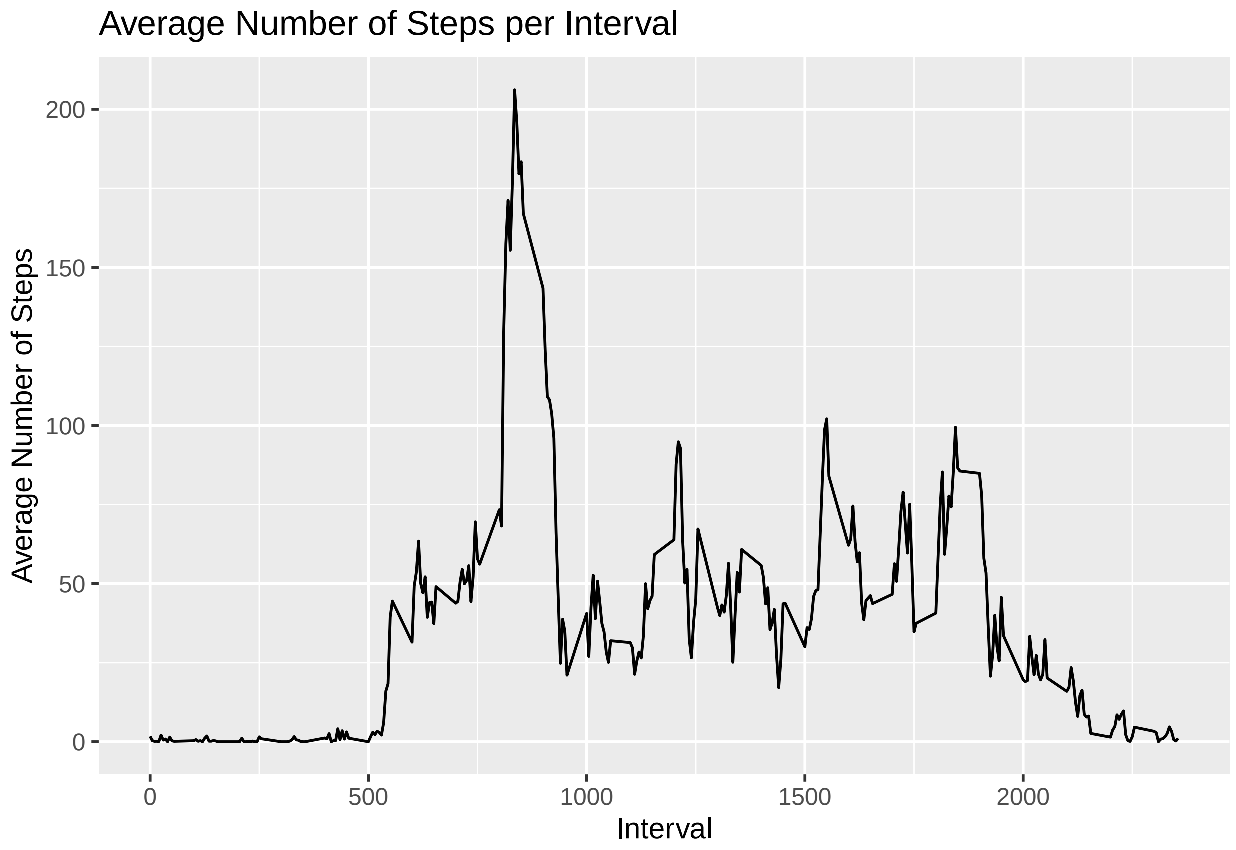
```
median(activityClean$steps)
```

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

What is the average daily activity pattern?

1. line plot of avaerage number of steps pper interval across all days

```
library(plyr)

intervalTable <- ddply(activityClean, .(interval), summarize, averaged_step = mean(steps))
ggplot(intervalTable, aes(interval, averaged_step)) +
  geom_line() +
  labs(x = "Interval", y = "Average Number of Steps", title = "Average Number of Steps per Interval")
```



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

```
maxSteps <- max(intervalTable$averaged_step)

intervalTable[intervalTable$averaged_step == maxSteps, 1]
```

```
## [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)

```
nrow(activity[is.na(activity$steps), ])
```

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

2&3. Strategy for filling in NAs

```
## replace NA by mean total steps taken per day
library(dplyr)
```

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

```
## The following objects are masked from 'package:plyr':
##
```

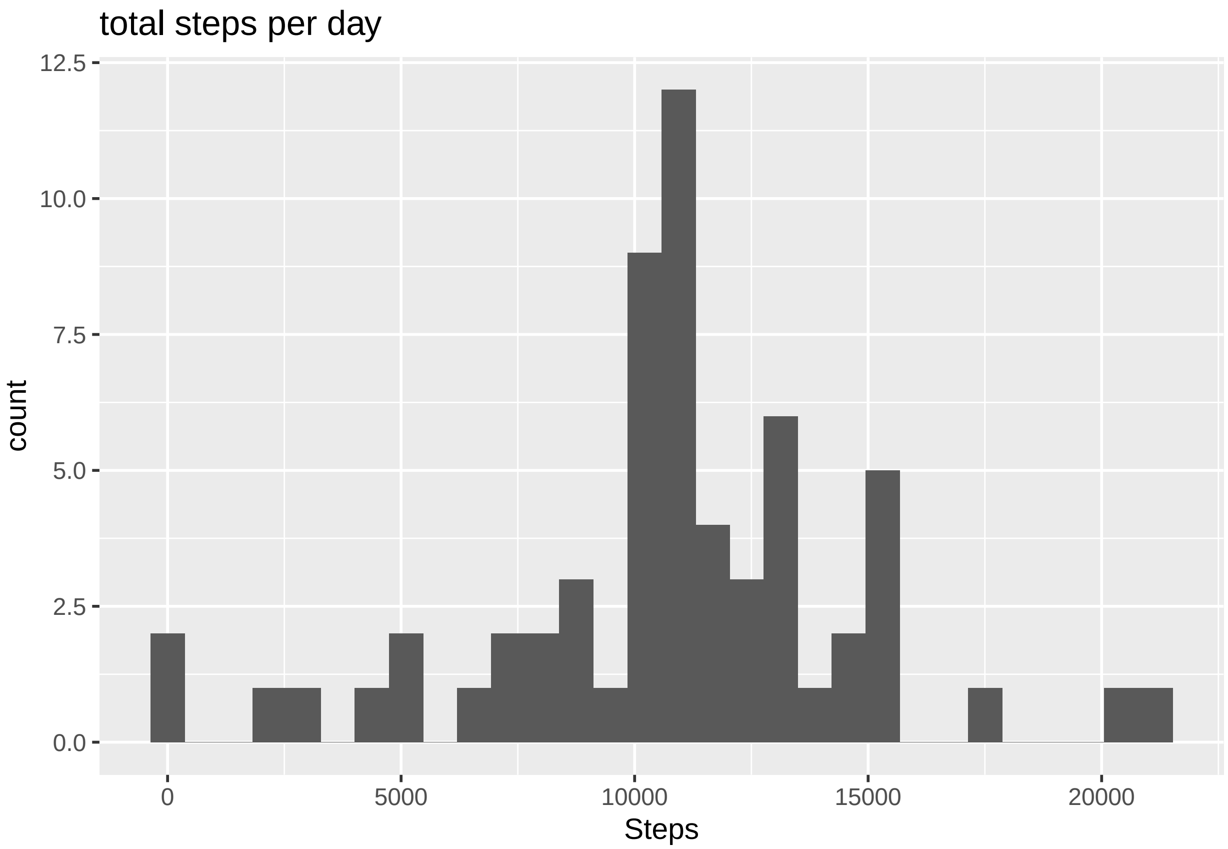
```
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize

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

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

```
activity_replace <- mutate(activity, step_replace  = ifelse(is.na(steps), 37.3826, steps))
```

4. Make a histogram of the total number of steps per day with imputeted data set.

```
StepTable_replaced <- aggregate(activity_replace$step_replace,
                                by = list(activity_replace$date), sum)
colnames(StepTable_replaced) <- c("date", "steps")
ggplot(StepTable_replaced, aes(steps)) + geom_histogram() + labs(x = "Steps", title = "total steps per
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



The re-made histogram is the same as previous.

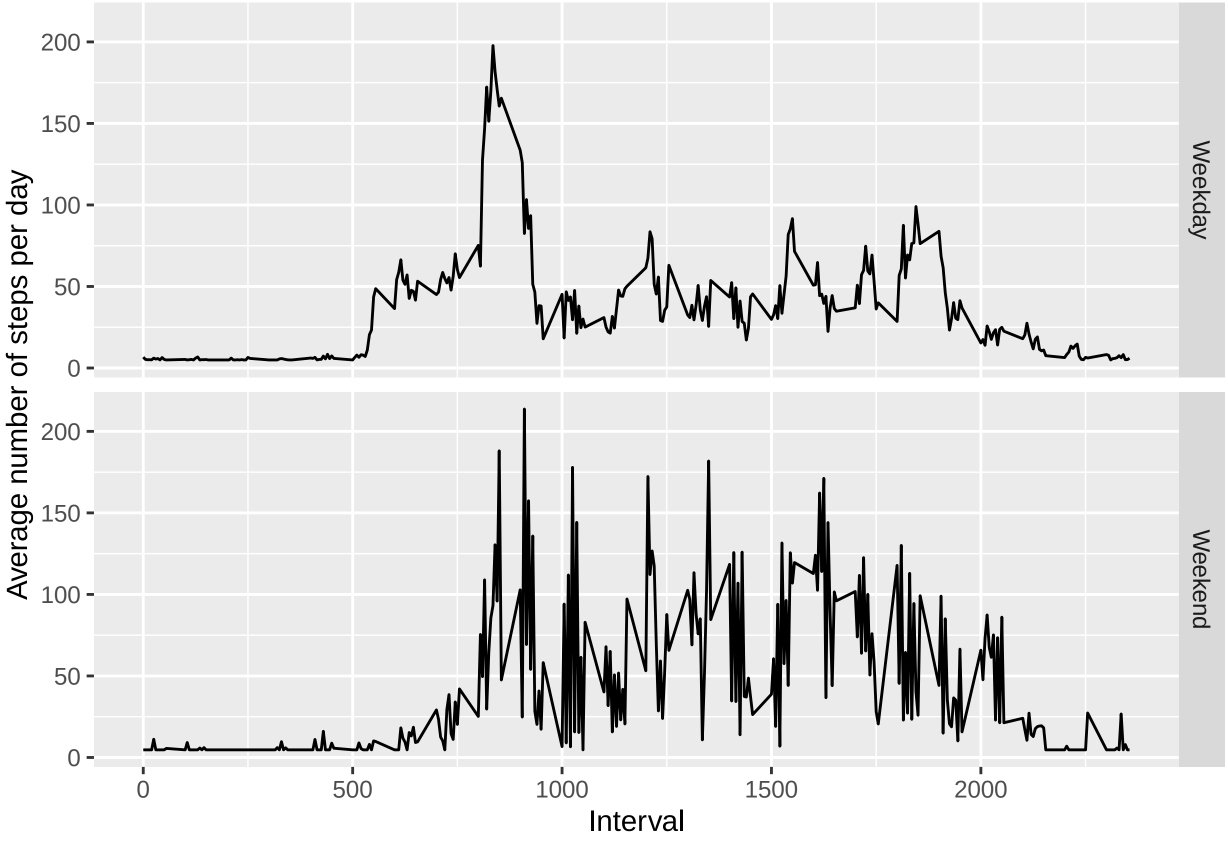
Are there difference in activity patterns between weekdays and weekends?

```
activity_replace$date <- as.Date(activity_replace$date, format = "%Y-%m-%d")
activity_replace$days <- weekdays(activity_replace$date)
```

```r
table(activity_replace$days)
```

```
##
##    Friday    Monday  Saturday    Sunday  Thursday   Tuesday Wednesday
##      2592      2592      2304      2304      2592      2592      2592
```

```r
activity_replace$dayType <- ifelse(activity_replace$days == c("Saturday", "Sunday"), "Weekend", "Weekday

activity_replace$dayType <- as.factor(activity_replace$dayType)

intervalTable_dayType <- aggregate(step_replace~interval + dayType, data = activity_replace, mean)

ggplot(intervalTable_dayType, aes(interval, step_replace)) +
  geom_line() + facet_grid(dayType~.) +
  xlab("Interval") + ylab("Average number of steps per day")
```


Average number of steps per day
Weekday
Weekend
Interval