

Reproducible Research: Peer Assessment 1

Qinqing Liu

10/1/2016

Loading and preprocessing the data

```
unzip(zipfile="activity.zip")
data <- read.csv("activity.csv", header = TRUE, sep = ',',
                colClasses = c("numeric", "character",
                              "integer"))
```

What is mean total number of steps taken per day?

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.2.4
```

```
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)
```

```
## Warning: package 'lubridate' was built under R version 3.2.5
```

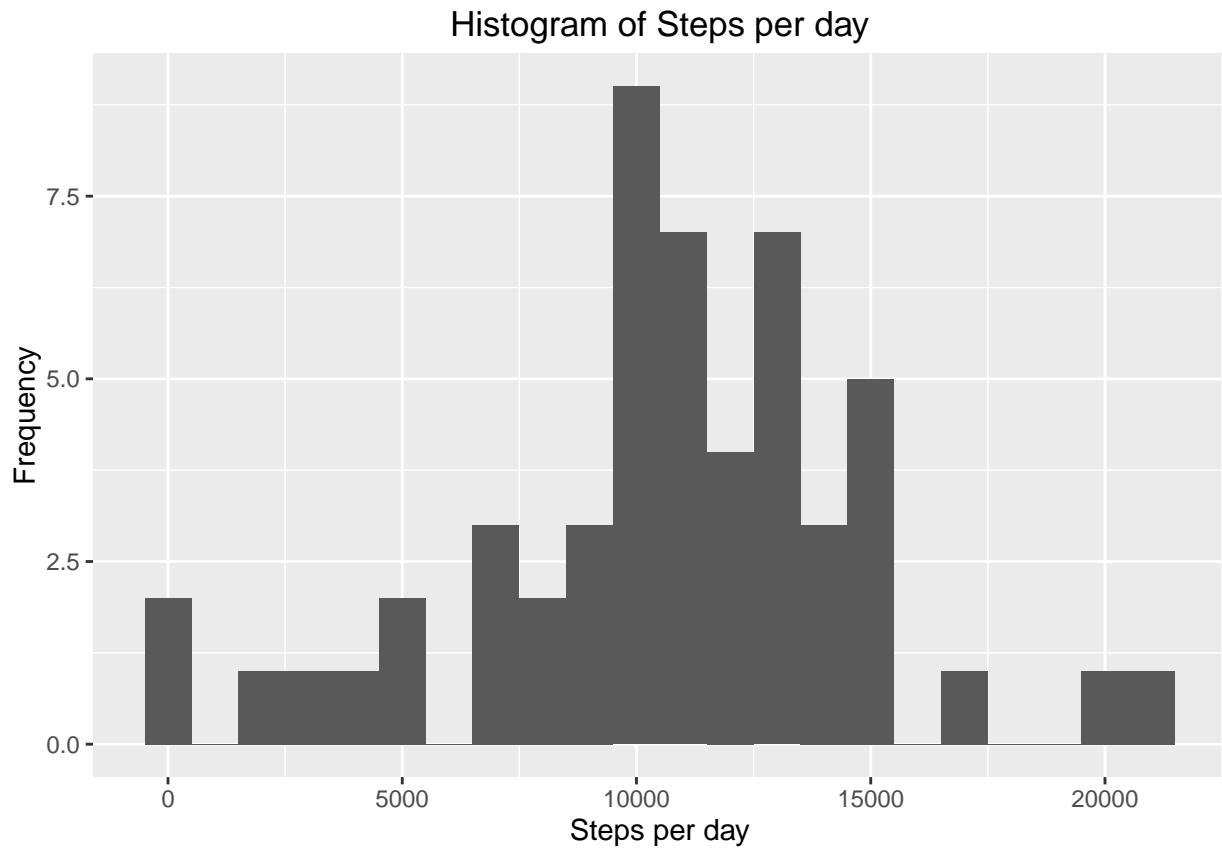
```
##
## Attaching package: 'lubridate'
##
## The following object is masked from 'package:base':
##
##   date
```

```
data$date <- ymd(data$date)
```

```
steps <- data %>%
  filter(!is.na(steps)) %>%
```

```
group_by(date) %>%
  summarize(steps = sum(steps))

ggplot(steps, aes(x = steps)) +
  geom_histogram(binwidth = 1000) +
  labs(title = "Histogram of Steps per day", x = "Steps per day", y = "Frequency")
```



```
mean_steps <- mean(steps$steps, na.rm = TRUE)
median_steps <- median(steps$steps, na.rm = TRUE)
```

```
mean_steps
```

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

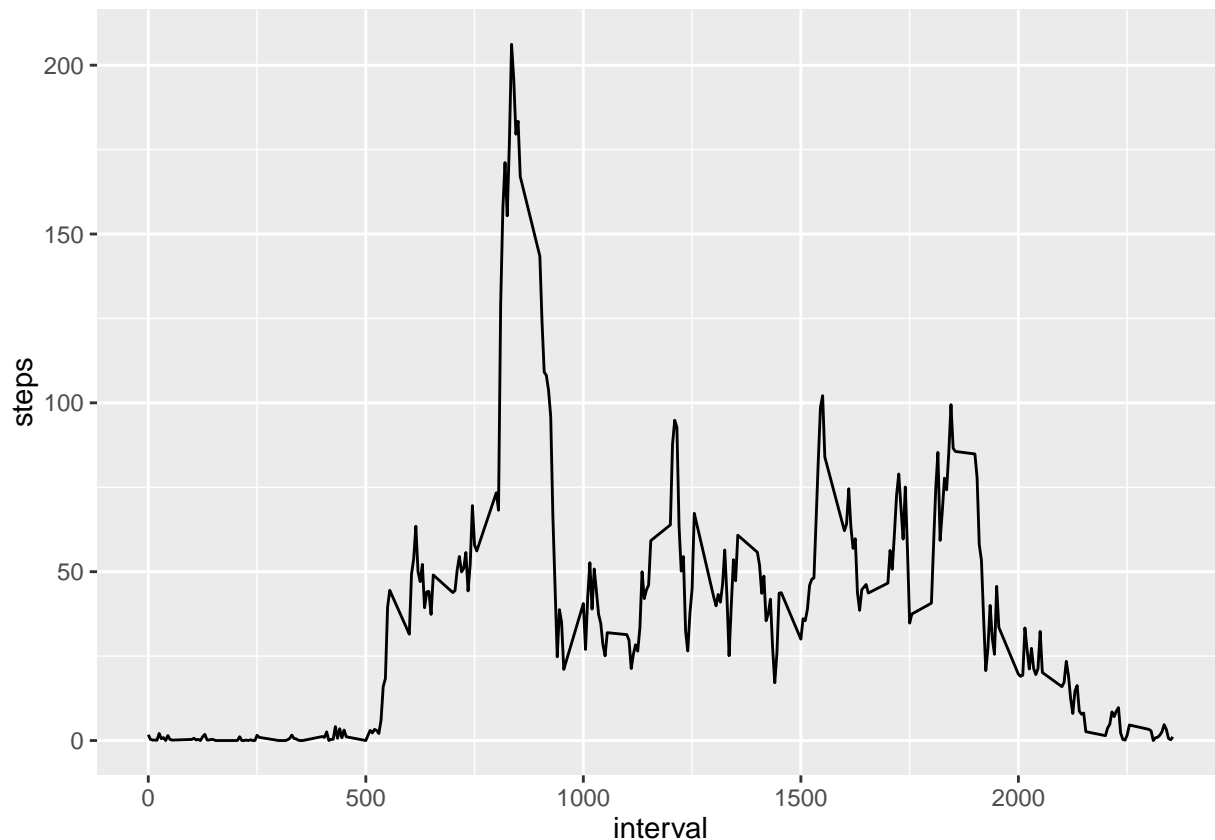
```
median_steps
```

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

```
# total_steps <- tapply(data$steps, data$date, FUN=sum, na.rm=TRUE)
# step_rec <- as.data.frame(rowsum(data$steps, data$date, reorder = TRUE, na.rm = TRUE))
# names(step_rec) <- c("Total_Steps")
# ggplot(step_rec, aes(x = Total_Steps)) +
#   geom_histogram()
# mean(step_rec[,1])
# median(step_rec[,1])
```

What is the average daily activity pattern?

```
interval <- data %>%  
  filter(!is.na(steps)) %>%  
  group_by(interval) %>%  
  summarize(steps = mean(steps))  
ggplot(interval, aes(x=interval, y=steps)) +  
  geom_line()
```



```
interval[which.max(interval$steps),]
```

```
## Source: local data frame [1 x 2]  
##  
##   interval    steps  
##   (int)    (dbl)  
## 1      835 206.1698
```

The interval 835 has the highest count of steps on average with 206 steps.

Imputing missing values

```
sum(is.na(data$steps))
```

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

Missing values are 2304.

Replace NA by mean value

```
data_full <- data
nas <- is.na(data_full$steps)
avg_interval <- tapply(data_full$steps, data_full$interval, mean, na.rm=TRUE, simplify=TRUE)
data_full$steps[nas] <- avg_interval[as.character(data_full$interval[nas])]
```

```
sum(is.na(data_full$steps))
```

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

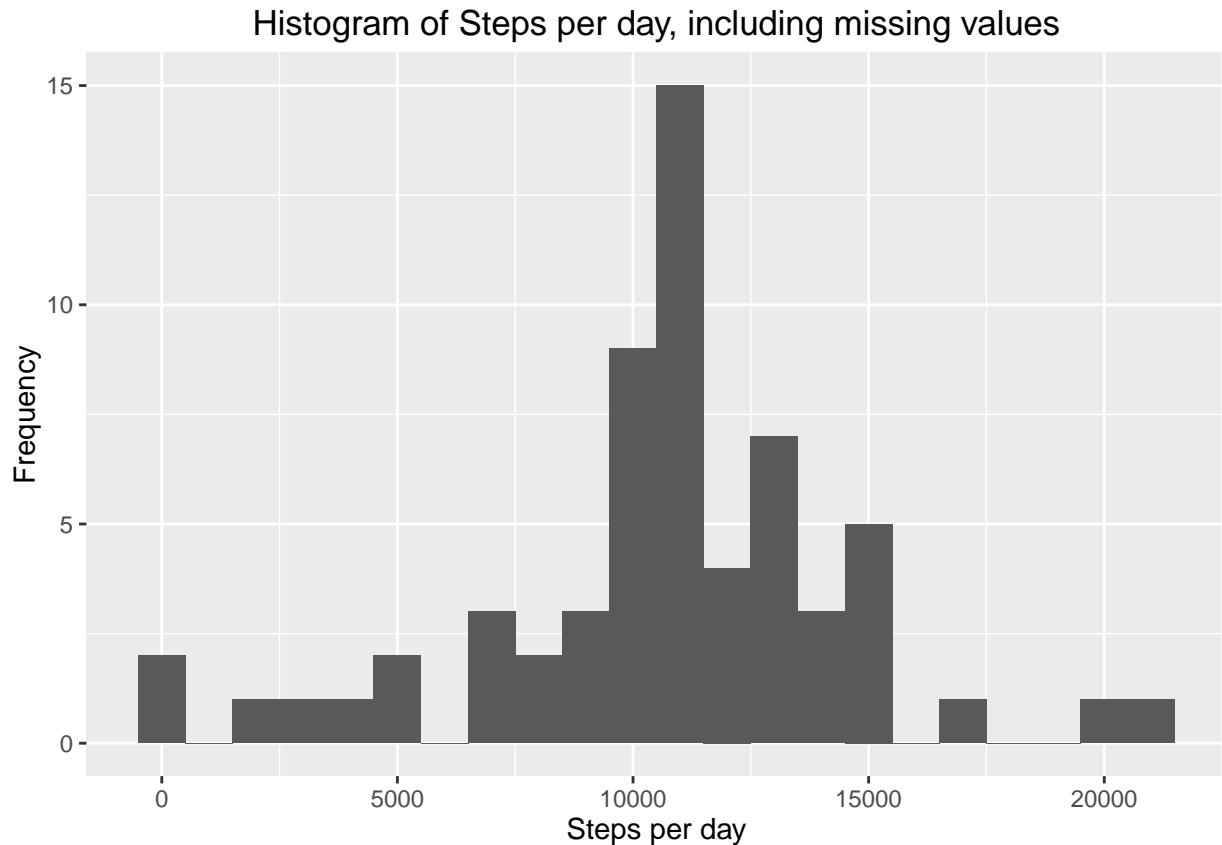
No more missing values now.

```
steps_full <- data_full %>%
  filter(!is.na(steps)) %>%
  group_by(date) %>%
  summarize(steps = sum(steps)) %>%
  print
```

```
## Source: local data frame [61 x 2]
```

```
##
##      date      steps
##      (date)    (dbl)
## 1  2012-10-01 10766.19
## 2  2012-10-02  126.00
## 3  2012-10-03 11352.00
## 4  2012-10-04 12116.00
## 5  2012-10-05 13294.00
## 6  2012-10-06 15420.00
## 7  2012-10-07 11015.00
## 8  2012-10-08 10766.19
## 9  2012-10-09 12811.00
## 10 2012-10-10  9900.00
## ..      ...      ...
```

```
ggplot(steps_full, aes(x = steps)) +
  geom_histogram(binwidth = 1000) +
  labs(title = "Histogram of Steps per day, including missing values", x = "Steps per day", y = "Frequency")
```



```
mean_steps_full <- mean(steps_full$steps, na.rm = TRUE)
median_steps_full <- median(steps_full$steps, na.rm = TRUE)
```

```
mean_steps_full
```

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

```
median_steps_full
```

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

Are there differences in activity patterns between weekdays and weekends?

```
data_full <- mutate(data_full, weektype = ifelse(weekdays(data_full$date) == "Saturday" | weekdays(data_full$date) == "Sunday", "weekend", "weekday"))
data_full$weektype <- as.factor(data_full$weektype)
head(data_full)
```

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

```
interval_full <- data_full %>%
  group_by(interval, weektype) %>%
  summarise(steps = mean(steps))
ggplot(interval_full, aes(x=interval, y=steps, color = weektype)) +
  geom_line() +
  facet_wrap(~weektype, ncol = 1, nrow=2)
```

