# Reproducible Research

Sri Vamsi

6/26/2020

## Downloading the data

```
setwd("C:/Users/HEMANT/Documents/GIT/RepData_PeerAssessment1")
list.files()
```

```
##  [1] "activity.csv"                  "activity.zip"
##  [3] "course5week2"                   "doc"
##  [5] "instructions_fig"              "markdown.Rmd"
##  [7] "PA_1.pdf"                      "PA_1.Rmd"
##  [9] "PA_1_files"                    "PA1_template.html"
## [11] "PA1_template.md"               "PA1_template.Rmd"
## [13] "PA1_template_files"            "plot1"
## [15] "plot1.png"                     "plot2.png"
## [17] "plot3.png"                     "plot4.png"
## [19] "README.md"                     "rep_1"
## [21] "RepData_PeerAssessment1.Rproj" "Rplots.pdf"
## [23] "week_2"
```

```
activity <- read.csv("activity.csv")
head(activity)
```

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

```
activity_final <- na.omit(activity)
head(activity_final)
```

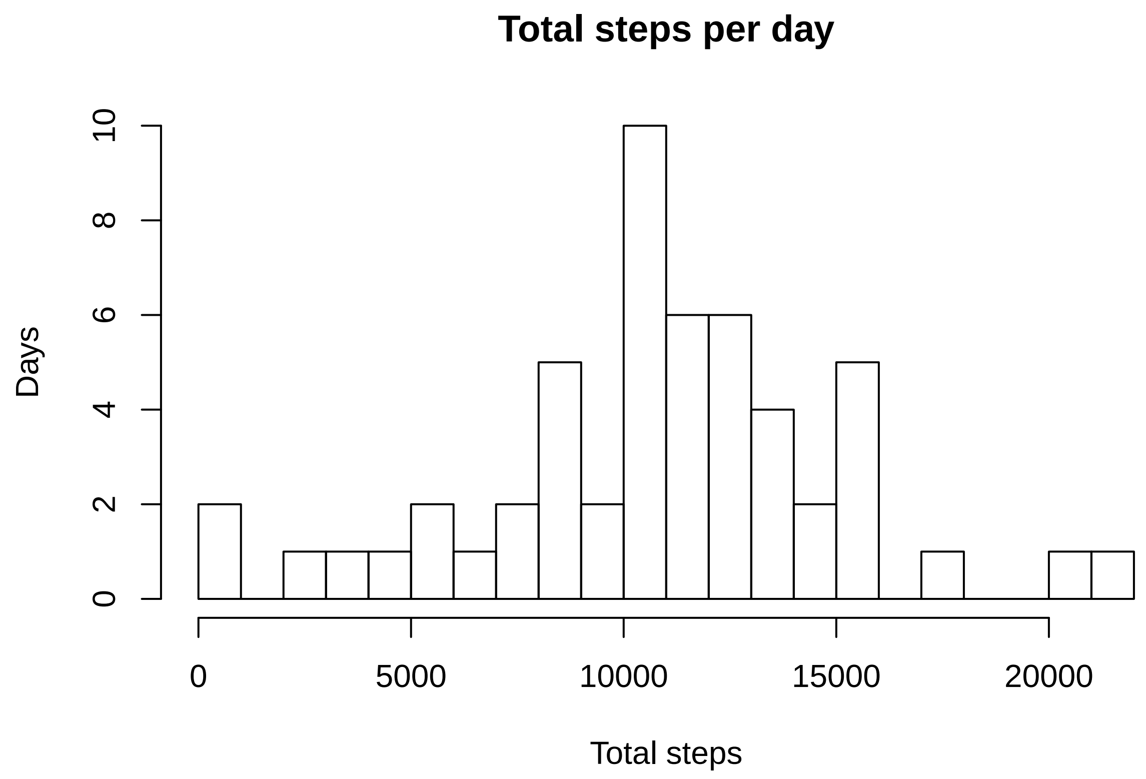
```
##     steps       date interval
## 289     0 2012-10-02        0
## 290     0 2012-10-02        5
## 291     0 2012-10-02       10
## 292     0 2012-10-02       15
## 293     0 2012-10-02       20
## 294     0 2012-10-02       25
```

### Mean steps taken per day

```
library(magrittr)
library(dplyr)
```

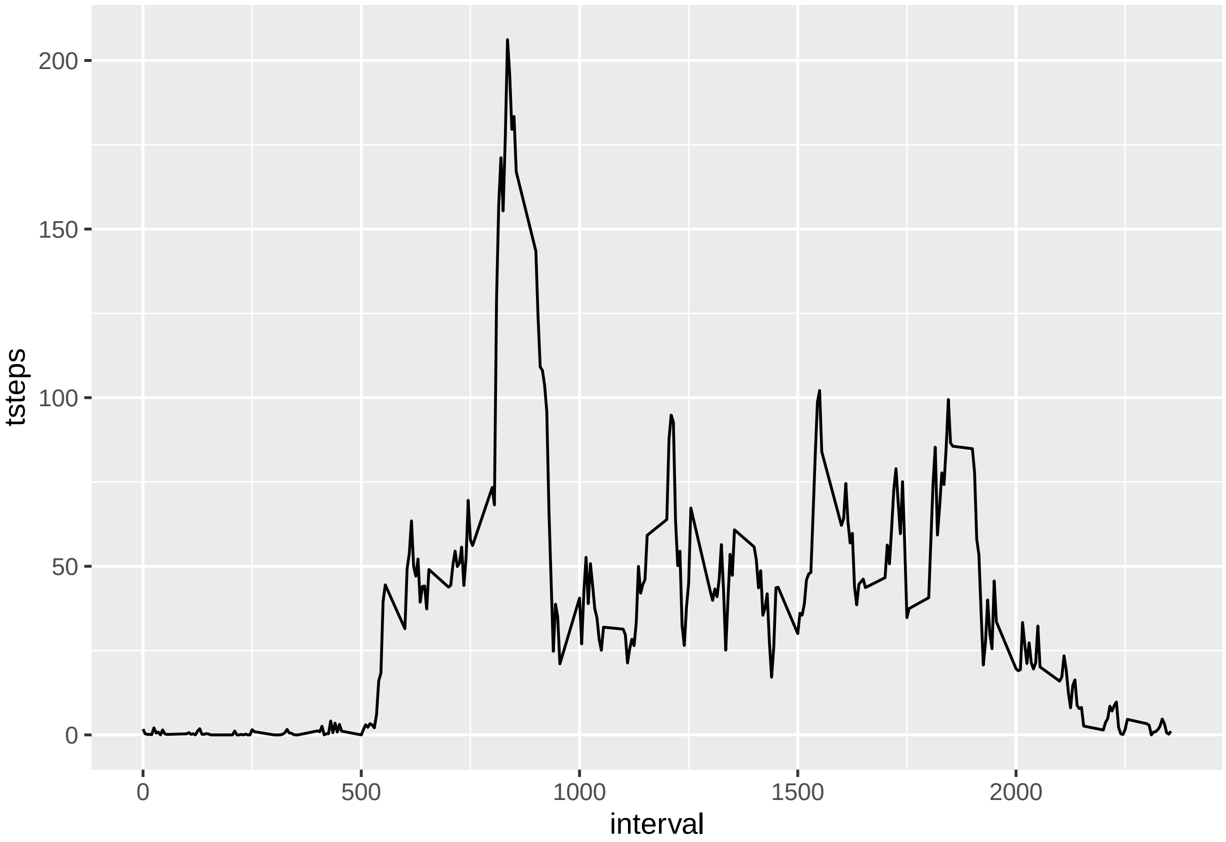
```
##
## Attaching package: 'dplyr'

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

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

```
library(ggplot2)
activity_date <- activity_final %>% group_by(date) %>% summarise(totalsteps = sum(steps))
mean_steps <- mean(activity_date$totalsteps)
median_steps <- median(activity_date$totalsteps)
hist(activity_date$totalsteps,xlab = "Total steps",ylab = "Days",main = "Total steps per day",breaks = 2
```



## Daily activity

```
library(ggplot2)
library(dplyr)
databyinterval <-activity %>% select(interval, steps) %>% na.omit() %>% group_by(interval) %>% summariz
ggplot(databyinterval,aes(x=interval,y=tsteps))+geom_line()
```

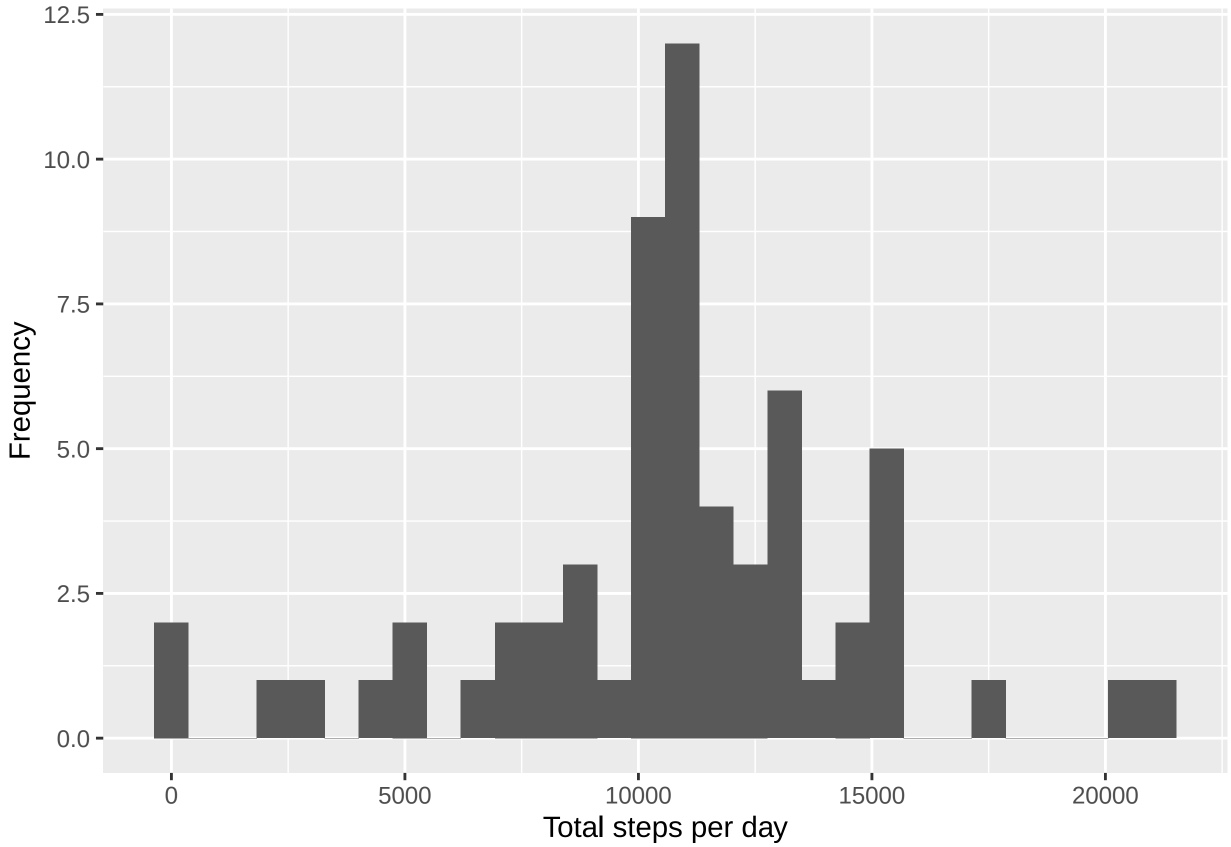


## Frequency of steps and Input of missing values

Our strategy to input the missing values is quite simple. We first subset those rows from the original data that have missing values and create a dataframe in which the value of steps for all rows is NA. We then fill in these with value of averages according to the corresponding interval.

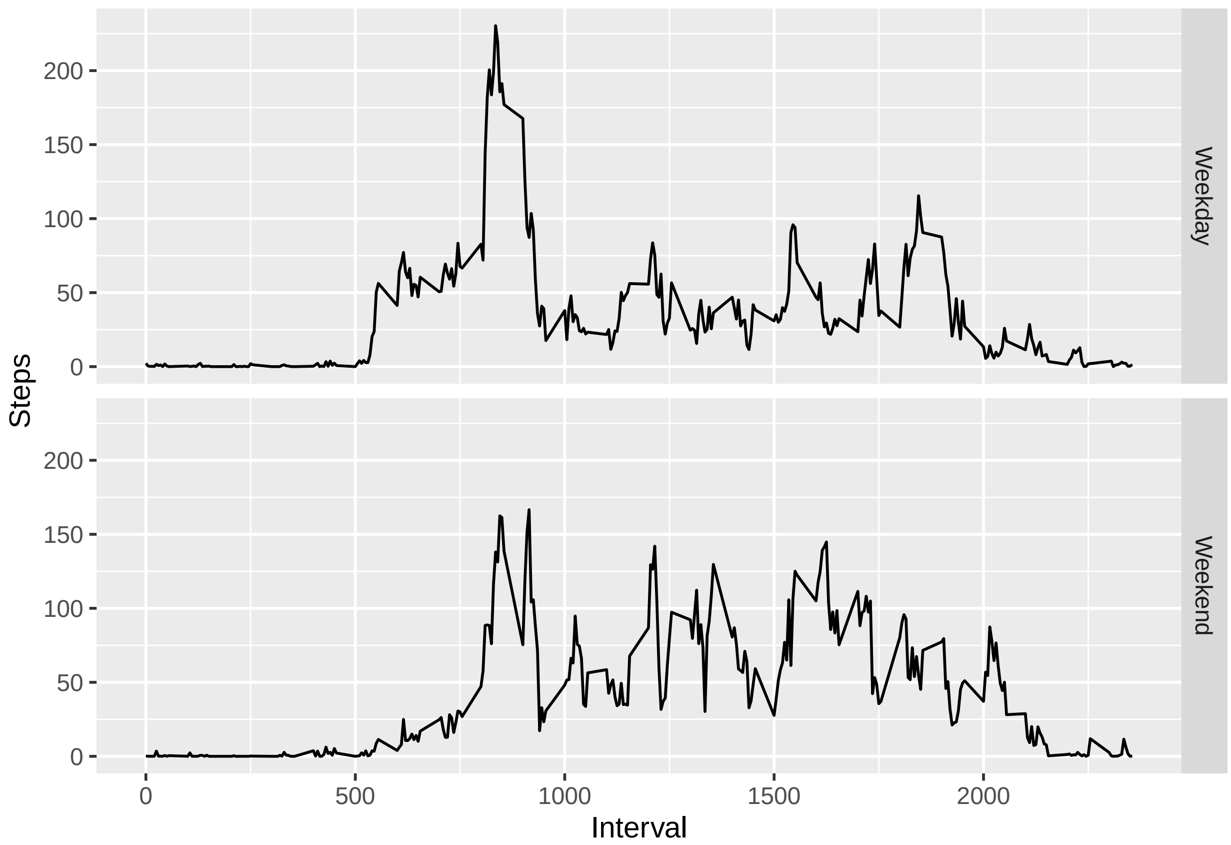
```
library(dplyr)
library(ggplot2)
activity_NA <- activity[which(is.na(activity$steps)),]
activity_NA$steps <- ifelse(activity_NA$interval == databyinterval$interval , databyinterval$tsteps)
activity_noNA <- rbind( activity_NA , activity_final)
View(activity_noNA)
Daily_steps <- activity_noNA %>% group_by(activity_noNA$date) %>% summarise(daily_steps = sum(steps))
qplot(daily_steps , data = Daily_steps , xlab = "Total steps per day" , ylab = "Frequency" )
```

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



## Difference in activity patterns between weekdays and weekends

```
activity_noNA$date <- as.Date(activity_noNA$date)
weekday <- weekdays(activity_noNA$date)
activity_noNA$weekday <- weekdays(activity_noNA$date)
View(activity_noNA)
activity_noNA$weekend <- ifelse(activity_noNA$weekday == "Saturday" |
                                  activity_noNA$weekday == "Sunday","Weekend","Weekday")
View(activity_noNA)
library(ggplot2)
y <- aggregate(activity_noNA$steps , by = list(activity_noNA$weekend , activity_noNA$interval) , na.omi
names(y) <- c("Weekend" , "Interval" , "Steps")
ggplot(data = y , aes(x = Interval , y = Steps)) + geom_line() + facet_grid(Weekend ~ .)
```


Steps
200
150
100
50
0
Weekday
200
150
100
50
0
Weekend
0
500
1000
1500
2000
Interval