Project 1: Reproducible Research

Paul Ighofose

2/26/2021

```
knitr::opts_chunk$set(echo = TRUE)
```

Personal Activity Monitoring

This data set utilizes personal activity data obtained from an anonymous individual. Its observations (i.e. step count) were recorded via five minute intervals during the months of October and November of 2012. It is my intention to read and process the data to discover a daily mean and median step count, its distribution, and the wholesomeness of the data and its potential affects on any analytically output generated. All data and views represented within this study are not implications on the population as a whole, yet is only a fragmented representation of how personal activity data could, potentially, be utilized to further understand an individuals daily patterns and possibly its affects on health.

Methods

Utilizing R, the personal activity monitoring data set was read into the IDE and processed to provide a data frame consisting of three columns(i.e. steps,date, & interval) with 17,768 observations.

```
###Installing necessary R packages:
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyve
rse 1.3.0 ──
```

```
## ✓ ggplot2 3.3.2     ✓ purrr   0.3.4
## ✓ tibble  3.0.3     ✓ dplyr   1.0.2
## ✓ tidyr   1.1.2     ✓ stringr 1.4.0
## ✓ readr   1.4.0     ✓ forcats 0.5.0
```

```
## ── Conflicts ─────────────────────────────────────── tidyverse_co
nflicts() ──
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
###Reading Data into R:
activity <- read.csv("Activity Monitoring.csv")
summary(activity)
```

```
##      steps            date              interval
##  Min.   :  0.00   Length:17568       Min.   :   0.0
##  1st Qu.:  0.00   Class :character   1st Qu.: 588.8
##  Median :  0.00   Mode  :character   Median :1177.5
##  Mean   : 37.38                      Mean   :1177.5
##  3rd Qu.: 12.00                      3rd Qu.:1766.2
##  Max.   :806.00                      Max.   :2355.0
##  NA's   :2304
```

Of those observations, 2,304 or approximately 13.1% of the steps were noted as "Na's" and indicated a need for removal. An identifier (Id), was applied to the data set to ensure consistency and correlation between step,date, and interval observations.

```
Id <- 1:17568
activity.Id <- cbind(Id, activity)
head(activity.Id,3)
```

```
##   Id steps    date interval
## 1  1    NA 10/1/12        0
## 2  2    NA 10/1/12        5
## 3  3    NA 10/1/12       10
```

```
Steps <- activity.Id %>% select(Id, steps)
interval <- activity.Id %>% select(Id,interval)
dAte <- as.Date(activity$date, "%m/%d/%y")
id <- activity.Id$Id
Data <- data.frame(Id = id, steps = activity.Id$steps, date = dAte,
interval = activity.Id$interval)
Movement <- na.omit(Data)
tibble(Movement)
```

```
## # A tibble: 15,264 x 4
##       Id steps date       interval
##    <int> <int> <date>        <int>
##  1   289     0 2012-10-02        0
##  2   290     0 2012-10-02        5
##  3   291     0 2012-10-02       10
##  4   292     0 2012-10-02       15
##  5   293     0 2012-10-02       20
##  6   294     0 2012-10-02       25
##  7   295     0 2012-10-02       30
##  8   296     0 2012-10-02       35
##  9   297     0 2012-10-02       40
## 10   298     0 2012-10-02       45
## # … with 15,254 more rows
```

To further understand the total number of steps taken each day, a loop was applied to summarize the steps observations diurnally. From the matrix provided, the date column was subset and integrated with the sum total step count to provide a more robust data frame. More so, the same process was replicated to determine, both, the mean and median step counts per day.

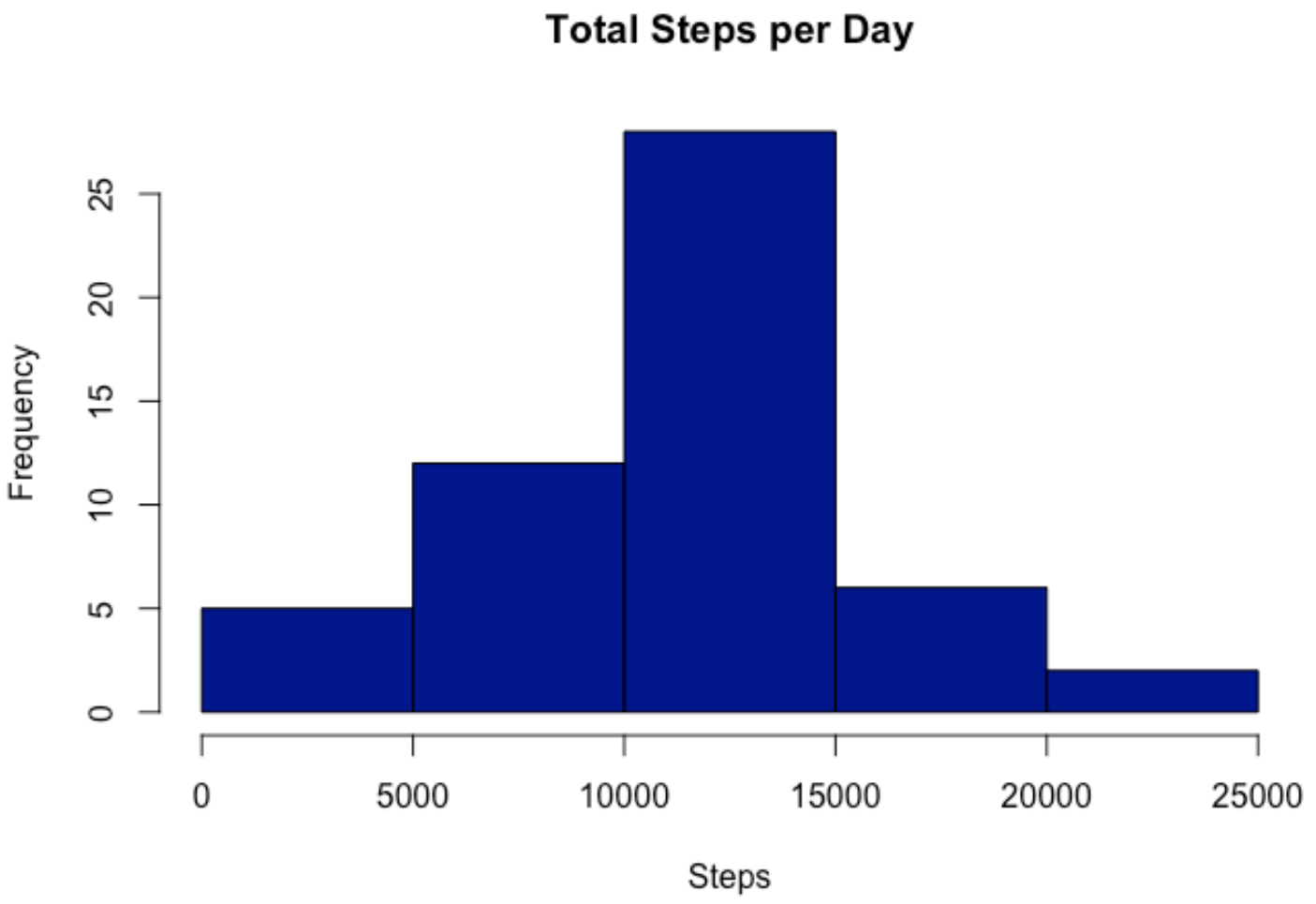
```
daily.sum <- tapply(Movement$steps,Movement$date,sum)
head(daily.sum,3)
```

```
## 2012-10-02 2012-10-03 2012-10-04
##        126      11352      12116
```

```
step.sum <- data.frame(sum = (daily.sum))
Dates <- data.frame(date = c("2012/10/02", "2012/10/03","2012/10/04"
,"2012/10/05","2012/10/06","2012/10/07","2012/10/09","2012/10/10","2
012/10/11","2012/10/12","2012/10/13","2012/10/14","2012/10/15","2012
/10/16","2012/10/17","2012/10/18","2012/10/19","2012/10/20","2012/10
/21","2012/10/22","2012/10/23","2012/10/24","2012/10/25","2012/10/26
","2012/10/27","2012/10/28","2012/10/29","2012/10/30","2012/10/31","
2012/11/02","2012/1103","2012/1105","2012/11/06","2012/11/07","2012/
11/08","2012/11/11","2012/11/12","2012/11/13","2012/11/15","2012/11/
16","2012/11/17","2012/11/18","2012/11/19","2012/11/20","2012/11/21"
,"2012/11/22","2012/11/23","2012/11/24","2012/11/25","2012/11/26","2
012/11/27","2012/11/28","2012/11/29"))
dates <- data.frame(date = as.Date(Dates$date, "%Y/%m/%d"))
movement.sum <- cbind(dates,step.sum)
head(movement.sum,3)
```

```
##                  date   sum
## 2012-10-02 2012-10-02   126
## 2012-10-03 2012-10-03 11352
## 2012-10-04 2012-10-04 12116
```

```
hist(movement.sum$sum, xlab = "Steps", main = "Total Steps per Day",
col = "dark blue")
```



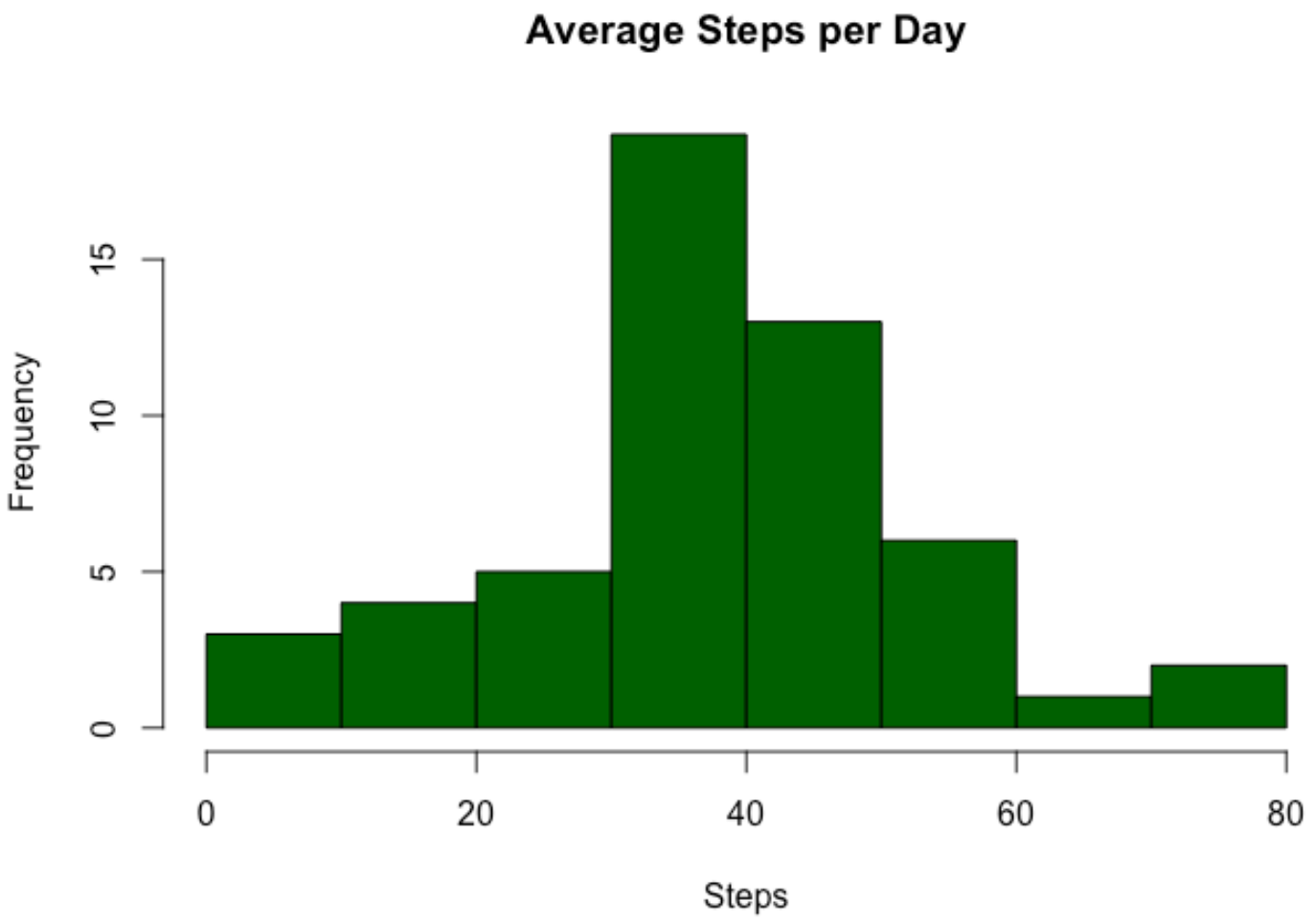

```
daily.mean <- tapply(Movement$steps,Movement$date,mean)
head(daily.mean,)
```

```
## 2012-10-02 2012-10-03 2012-10-04 2012-10-05 2012-10-06 2012-10-07
##    0.43750   39.41667   42.06944   46.15972   53.54167   38.24653
```

```
step.mean <- data.frame(mean = (daily.mean))
Dates <- data.frame(date = c("2012/10/02", "2012/10/03","2012/10/04"
,"2012/10/05","2012/10/06","2012/10/07","2012/10/09","2012/10/10",
"2012/10/11", "2012/10/12", "2012/10/13", "2012/10/14", "2012/10/15"
, "2012/10/16", "2012/10/17", "2012/10/18", "2012/10/19", "2012/10/2
0","2012/10/21","2012/10/22","2012/10/23","2012/10/24","2012/10/25",
"2012/10/26","2012/10/27","2012/10/28","2012/10/29","2012/10/30","20
12/10/31","2012/11/02","2012/11/03","2012/11/05","2012/11/06","2012/
11/07","2012/11/08","2012/11/11","2012/11/12","2012/11/13","2012/11/
15","2012/11/16","2012/11/17","2012/11/18","2012/11/19","2012/11/20"
,"2012/11/21","2012/11/22","2012/11/23","2012/11/24","2012/11/25","2
012/11/26","2012/11/27","2012/11/28","2012/11/29"))
dates <- data.frame(date = as.Date(Dates$date, "%Y/%m/%d"))
movement.mean <- cbind(dates,step.mean)
head(movement.mean,3)
```

```
##                  date     mean
## 2012-10-02 2012-10-02  0.43750
## 2012-10-03 2012-10-03 39.41667
## 2012-10-04 2012-10-04 42.06944
```

```
hist(movement.mean$mean, xlab = "Steps", main = "Average Steps per D
ay", col = "dark green")
```




```
###Step Median:
daily.median <- tapply(Movement$steps,Movement$date,median)
head(daily.median,3)
```

```
## 2012-10-02 2012-10-03 2012-10-04
##          0          0          0
```

```
step.median <- data.frame(median = (daily.median))
Dates <- data.frame(date = c("2012/10/02", "2012/10/03","2012/10/04"
,"2012/10/05","2012/10/06","2012/10/07","2012/10/09","2012/10/10",
"2012/10/11", "2012/10/12", "2012/10/13", "2012/10/14", "2012/10/15"
, "2012/10/16", "2012/10/17", "2012/10/18", "2012/10/19", "2012/10/2
0","2012/10/21","2012/10/22","2012/10/23","2012/10/24","2012/10/25",
"2012/10/26","2012/10/27","2012/10/28","2012/10/29","2012/10/30","20
12/10/31","2012/11/02","2012/11/03","2012/11/05","2012/11/06","2012/
11/07","2012/11/08","2012/11/11","2012/11/12","2012/11/13","2012/11/
15","2012/11/16","2012/11/17","2012/11/18","2012/11/19","2012/11/20"
,"2012/11/21","2012/11/22","2012/11/23","2012/11/24","2012/11/25","2
012/11/26","2012/11/27","2012/11/28","2012/11/29"))
dates <- data.frame(date = as.Date(Dates$date, "%Y/%m/%d"))
movement.median <- cbind(dates,step.median)
head(movement.median,3)
```

```
##                  date median
## 2012-10-02 2012-10-02      0
## 2012-10-03 2012-10-03      0
## 2012-10-04 2012-10-04      0
```

```
setwd("/users/paulighofose/Desktop/Reproducible Research")
png(filename = "Median Steps per Day.png", width = 480, height = 480
)
hist(movement.median$median,col = "dark red", xlab = "Steps", main =
"Median Steps per Day")
```

Yet, when considering the data along a continuum such as a time series, the information is limited. Thus, to measure the activity (i.e. step count) across an acculmination of 5 minute intervals, a time series plot was utilized.

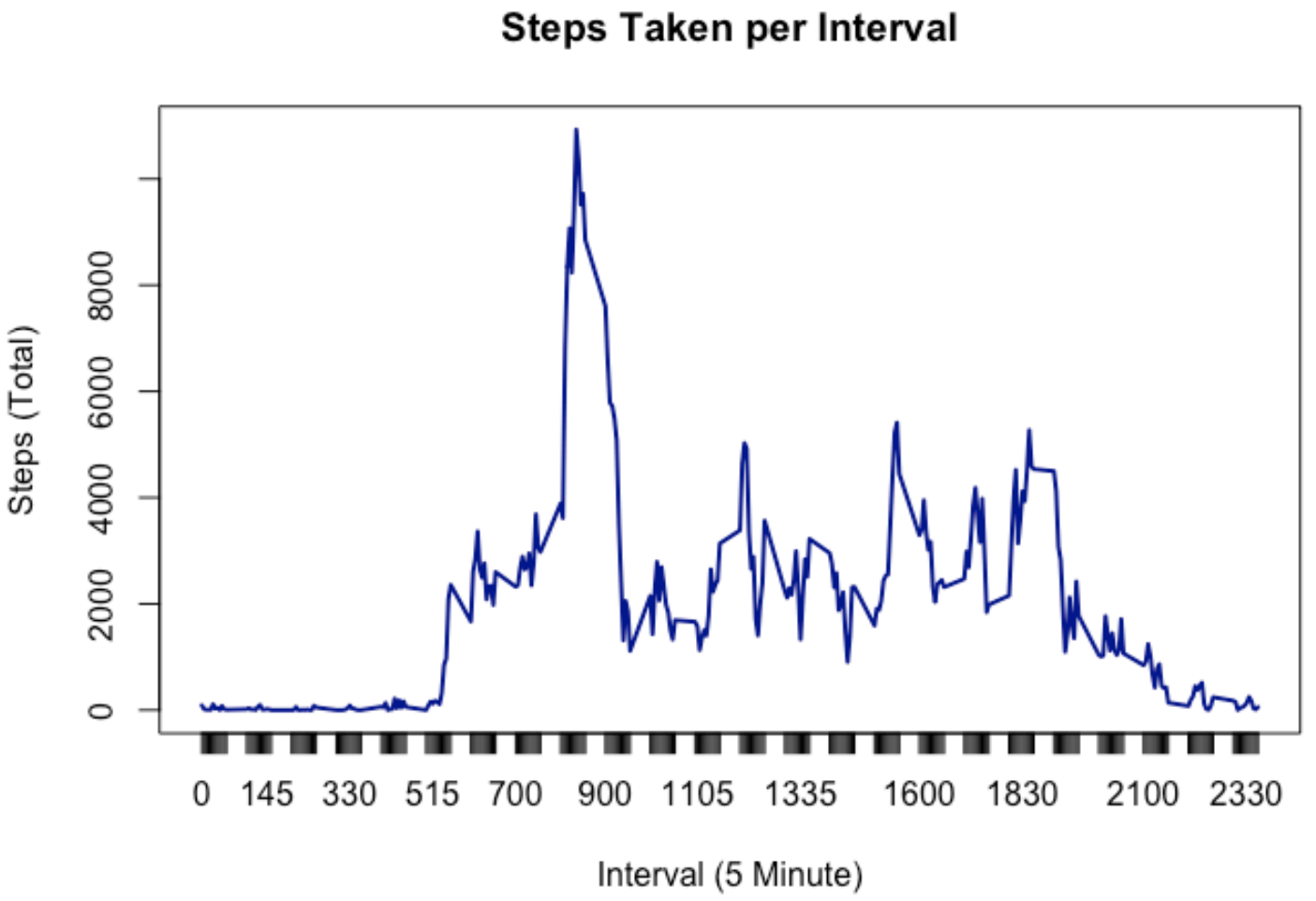
```
steps.total <- as.table(tapply(Movement$steps, Movement$interval, su
m ))
head(steps.total,3)
```

```
##  0  5 10
## 91 18  7
```

```
max(steps.total)
```

```
## [1] 10927
```

```
plot(steps.total,type  = "l", col = "dark blue", xlab = "Interval (5
Minute)", ylab = "Steps (Total)", main = "Steps Taken per Interval")
```



Examination of the time series indicates an increased number of steps taken between intervals 800 and 1000, and more specifically a maximum of 10,927 steps were taken within that period.

Although simplistic in its' representation, this maximum value also indicates an extreme within the step count observation. For if, one were to include the complete data set, thereby replacing all "NA" values with that of the population's mean, the average step count figure would have become skewed and inconsiderable as a accurate

representation of that data. Thus, when re-examining the data and replacing those observations with a median value of "0.00", the data more accurately depicts the distribution of a complete data set.

```
head(activity,3)
```

```
##   steps    date interval
## 1    NA 10/1/12        0
## 2    NA 10/1/12        5
## 3    NA 10/1/12       10
```

```
summary(activity)
```

```
##      steps             date              interval     
##  Min.   :  0.00   Length:17568       Min.   :   0.0  
##  1st Qu.:  0.00   Class :character   1st Qu.: 588.8  
##  Median :  0.00   Mode  :character   Median :1177.5  
##  Mean   : 37.38                      Mean   :1177.5  
##  3rd Qu.: 12.00                      3rd Qu.:1766.2  
##  Max.   :806.00                      Max.   :2355.0  
##  NA's   :2304
```

```
median <- 0
activity[is.na(activity)] = median

###Creating a new dataset of data with input values:
activity2 <- activity
tibble(activity2)
```

```
## # A tibble: 17,568 x 3
##    steps date    interval
##    <dbl> <chr>      <int>
##  1     0 10/1/12        0
##  2     0 10/1/12        5
##  3     0 10/1/12       10
##  4     0 10/1/12       15
##  5     0 10/1/12       20
##  6     0 10/1/12       25
##  7     0 10/1/12       30
##  8     0 10/1/12       35
##  9     0 10/1/12       40
## 10     0 10/1/12       45
## # … with 17,558 more rows
```

And just as before, the new inclusive data set was looped and a total step count per day was derived.

```
stepsperday <- tapply(activity2$steps,activity2$date, sum)

###Renaming column:
stepsperday.with <- data.frame(steps = (stepsperday))
head(stepsperday.with,3)
```

```
##          steps
## 10/1/12      0
## 10/10/12  9900
## 10/11/12 10304
```

```
as.array(stepsperday)
```

```
##  10/1/12 10/10/12 10/11/12 10/12/12 10/13/12 10/14/12 10/15/12 10
/16/12
##        0     9900    10304    17382    12426    15098    10139
15084
## 10/17/12 10/18/12 10/19/12  10/2/12 10/20/12 10/21/12 10/22/12 10
/23/12
##    13452    10056    11829      126    10395     8821    13460
8918
## 10/24/12 10/25/12 10/26/12 10/27/12 10/28/12 10/29/12  10/3/12 10
/30/12
##     8355     2492     6778    10119    11458     5018    11352
9819
## 10/31/12  10/4/12  10/5/12  10/6/12  10/7/12  10/8/12  10/9/12  1
1/1/12
##    15414    12116    13294    15420    11015        0    12811
0
## 11/10/12 11/11/12 11/12/12 11/13/12 11/14/12 11/15/12 11/16/12 11
/17/12
##        0    12608    10765     7336        0       41     5441
14339
## 11/18/12 11/19/12  11/2/12 11/20/12 11/21/12 11/22/12 11/23/12 11
/24/12
##    15110     8841    10600     4472    12787    20427    21194
14478
## 11/25/12 11/26/12 11/27/12 11/28/12 11/29/12  11/3/12 11/30/12  1
1/4/12
##    11834    11162    13646    10183     7047    10571        0
0
##  11/5/12  11/6/12  11/7/12  11/8/12  11/9/12
##    10439     8334    12883     3219        0
```

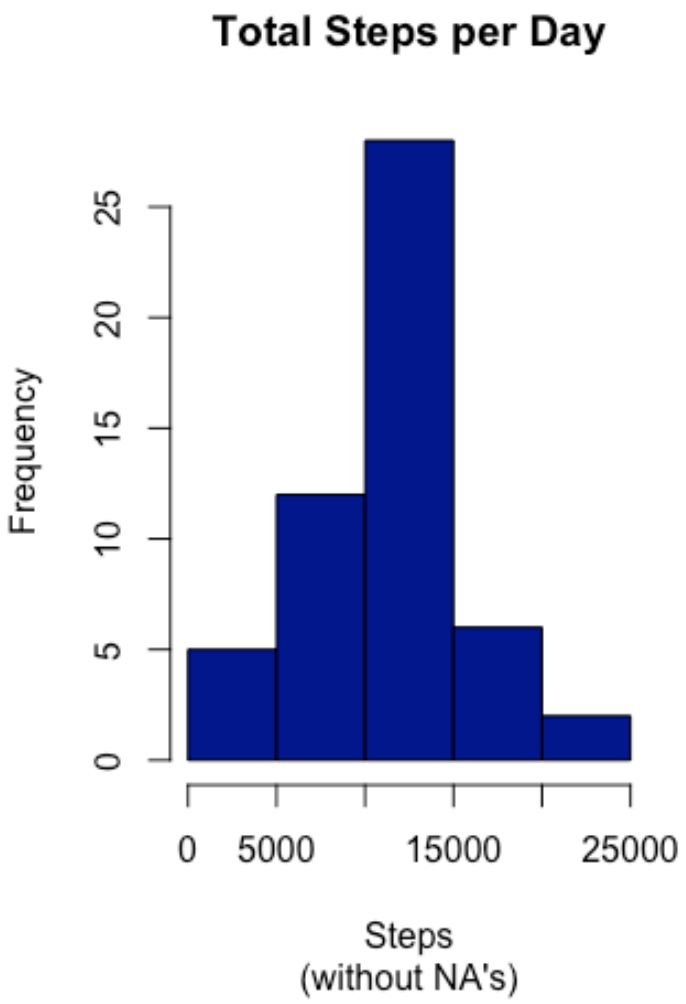
```
###Dates are copied and a new data.frame is created with Dates and s
teps
Dates <- data.frame(dates = c("10/1/12", "10/10/12", "10/11/12", "10
/12/12", "10/13/12", "10/14/12", "10/15/12", "10/16/12", "10/17/12",
"10/18/12", "10/19/12", "10/2/12", "10/20/12", "10/21/12", "10/22/12
", "10/23/12", "10/24/12", "10/25/12", "10/26/12", "10/27/12", "10/2
8/12", "10/29/12","10/3/12", "10/30/12", "10/31/12",  "10/4/12",  "1
0/5/12",  "10/6/12",  "10/7/12",  "10/8/12",  "10/9/12",  "11/1/12",
"11/10/12", "11/11/12", "11/12/12", "11/13/12", "11/14/12", "11/15/1
2", "11/16/12", "11/17/12", "11/18/12", "11/19/12",  "11/2/12", "11/
20/12","11/21/12", "11/22/12", "11/23/12", "11/24/12", "11/25/12", "
11/26/12", "11/27/12", "11/28/12", "11/29/12",  "11/3/12", "11/30/12
", "11/4/12",  "11/5/12",  "11/6/12",  "11/7/12",  "11/8/12",  "11/9
/12"))
Date <- data.frame(date = as.Date(Dates$dates, "%m/%d/%y"))
head(Date,3)
```

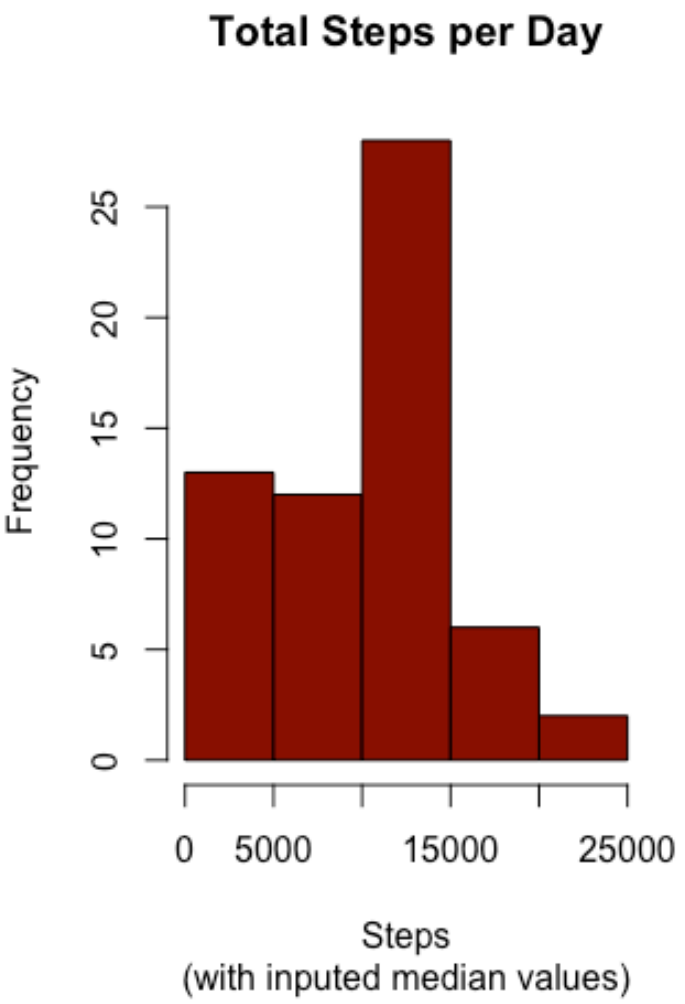
```
##         date
## 1 2012-10-01
## 2 2012-10-10
## 3 2012-10-11
```

```
activity4 <- cbind(Date,steps = stepsperday.with$steps)
head(activity4,)
```

```
##         date steps
## 1 2012-10-01     0
## 2 2012-10-10  9900
## 3 2012-10-11 10304
## 4 2012-10-12 17382
## 5 2012-10-13 12426
## 6 2012-10-14 15098
```

```
par(mfrow = c(1,2))
hist(movement.sum$sum, xlab = "Steps", main = "Total Steps per Day",
sub = "(without NA's)", col = "dark blue")
hist(activity4$steps,xlab = "Steps",col = "dark red", main = "Total
Steps per Day", sub = "(with inputed median values)")
```





The distribution of the data is slightly different, with the imputed data set skewing slightly left. This indicates an increase in steps below 15,000 in comparison to the data set without "NA" observations. Thus, one must consider if the data had been complete (i.e without any missing values) then the individual's activity monitor would have indicated an increase in total steps per day. The only question to now consider is whether the individuals weekday and weekend activities are similar or not.

To do so, the character vector "date" was converted to a date vector and filtered by day. Weekdays were combined to provide one collective, as weekend observations were combined to provide another. And, just as before the two subsets were individually looped and a mean step count calculated.

```
date.2 <- activity2$date
date.3 <- data.frame(date = as.Date(date.2, "%m/%d/%y"))
activity.data <- cbind(steps = activity2$steps,date = date.3,interva
l = activity2$interval)
head(activity.data)
```

```
##   steps       date interval
## 1     0 2012-10-01        0
## 2     0 2012-10-01        5
## 3     0 2012-10-01       10
## 4     0 2012-10-01       15
## 5     0 2012-10-01       20
## 6     0 2012-10-01       25
```

```
activity.data.1 <- mutate(activity.data,weekday = weekdays(activity.
data$date))
tibble(activity.data.1)
```

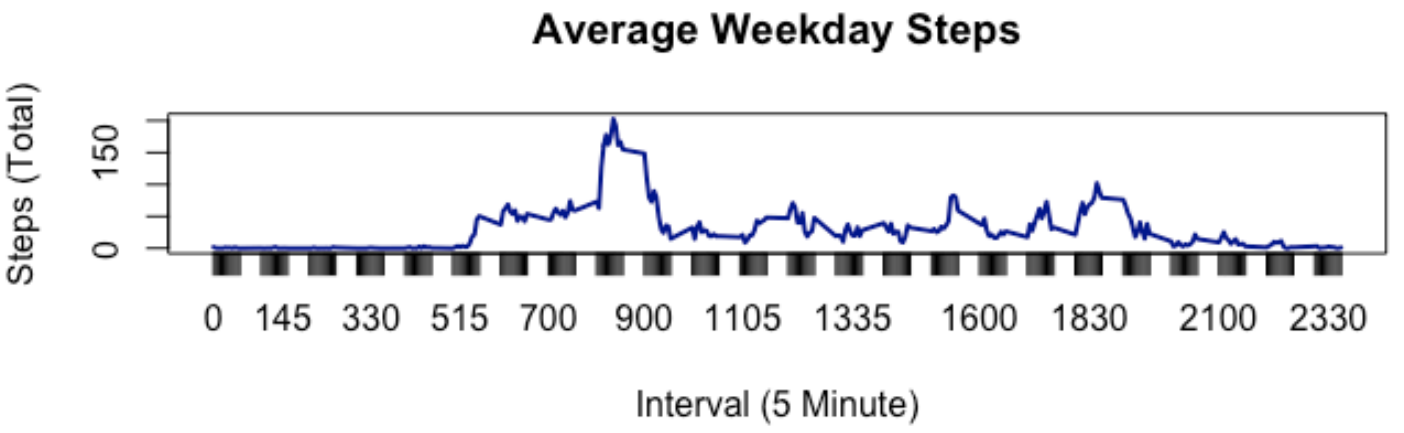
```
## # A tibble: 17,568 x 4
##    steps date       interval weekday
##    <dbl> <date>        <int> <chr>
##  1     0 2012-10-01        0 Monday
##  2     0 2012-10-01        5 Monday
##  3     0 2012-10-01       10 Monday
##  4     0 2012-10-01       15 Monday
##  5     0 2012-10-01       20 Monday
##  6     0 2012-10-01       25 Monday
##  7     0 2012-10-01       30 Monday
##  8     0 2012-10-01       35 Monday
##  9     0 2012-10-01       40 Monday
## 10     0 2012-10-01       45 Monday
## # … with 17,558 more rows
```

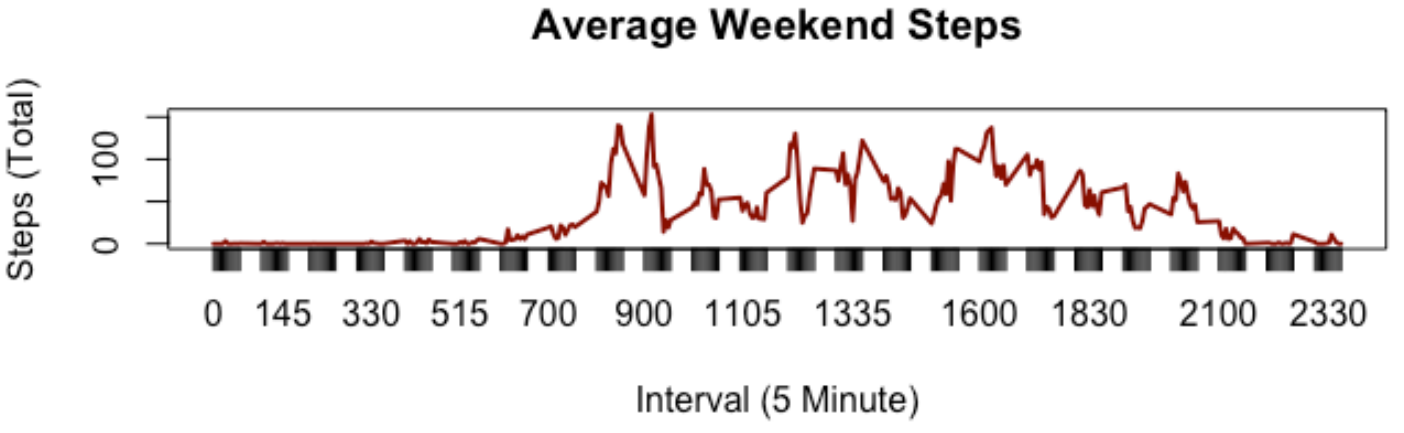
```
###Sorting Days:
activity6m <- activity.data.1 %>% filter(weekday == "Monday")
activity6t <- activity.data.1 %>% filter(weekday == "Tuesday")
activity6w <- activity.data.1 %>% filter(weekday == "Wednesday")
activity6th <- activity.data.1 %>% filter(weekday == "Thursday")
activity6f <- activity.data.1 %>% filter(weekday == "Friday")
activity6sat <- activity.data.1 %>% filter(weekday == "Saturday")
activity6sun <- activity.data.1 %>% filter(weekday == "Sunday")

###Combining weekdays and weekend days:
weekday <- rbind(activity6m, activity6t, activity6w, activity6th, ac
tivity6f)
weekend <- rbind(activity6sat, activity6sun)


###Creating time series of weekday and weekend activity:
steps.weekday <- as.table(tapply(weekday$steps, weekday$interval, me
an ))
steps.weekend <- as.table(tapply(weekend$steps, weekend$interval,mea
n ))
```

```
par(mfrow = c(2,1))
plot(steps.weekday,type  = "l", col = "dark blue", xlab = "Interval
(5 Minute)", ylab = "Steps (Total)", main = "Average Weekday Steps")
plot(steps.weekend,type  = "l", col = "dark red", xlab = "Interval (
5 Minute)", ylab = "Steps (Total)", main = "Average Weekend Steps")
```





In comparison the average weekday step count appears greater, however if one were to remove the weekday maximum, one would notice that the weekend step count maximum and occurrence are greater and more frequent. Indicating greater mobility within that time period.