Week 2 Project

Introduction

It is now possible to collect a large amount of data about personal movement using activity monitoring devices such as a Fitbit, Nike Fuelband, or Jawbone Up. These type of devices are part of the "quantified self" movement - a group of enthusiasts who take measurements about themselves regularly to improve their health, to find patterns in their behavior, or because they are tech geeks.

This assignment makes use of data from a personal activity monitoring device to answer questions about the subject's activity levels. Specificly, it uses data collected from an anonymous individual during the months of October and November, 2012 at 5 minute intervals through out the day to offer:

1. A histogram of the total number of steps taken each day
2. The mean and median number of steps taken each day
3. A time series plot of the average number of steps taken
4. The 5-minute interval that, on average, contains the maximum number of steps
5. Calculate and report the number of missing values
6. Code to describe and show a strategy for imputing missing data
7. A histogram of the total number of steps taken each day after missing values are imputed; and
8. A panel plot comparing the average number of steps taken per 5-minute interval across weekdays and weekends ## Code for Reading and Processing The Data

```
activity <- read.csv("C:/Users/wk469/OneDrive/Documents/activity.csv")
library(lubridate)
```

```
## Warning: package 'lubridate' was built under R version 3.3.3
```

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

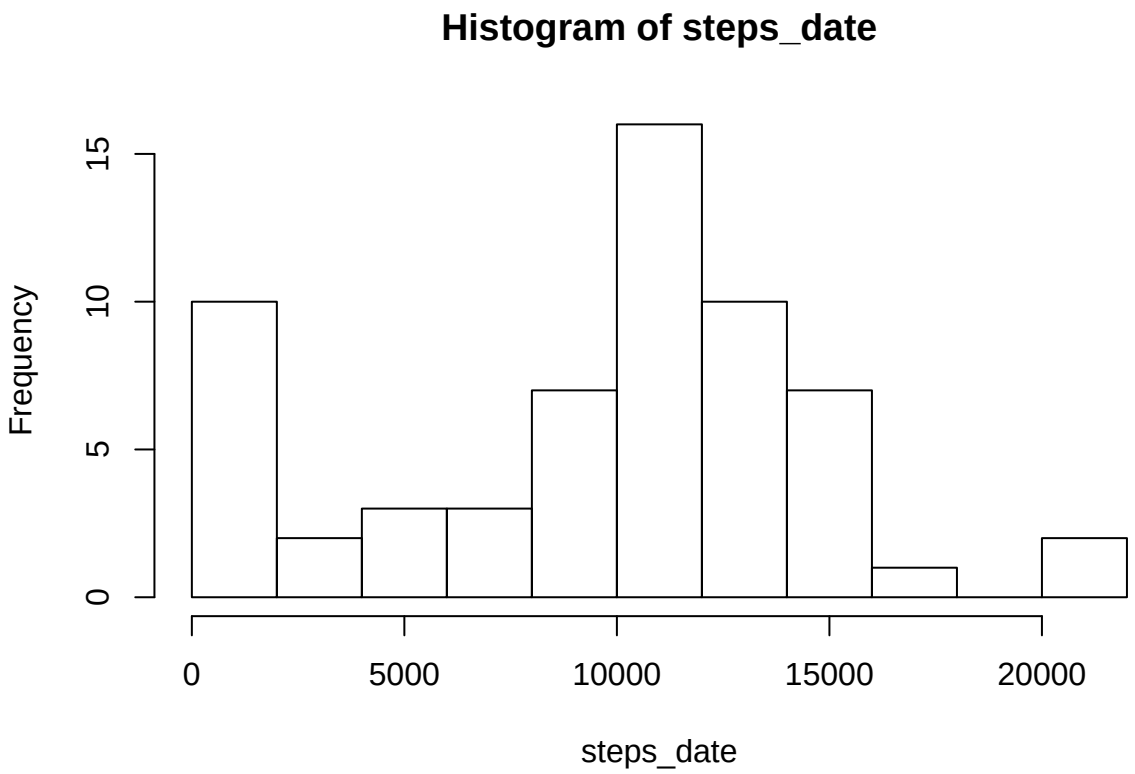
```
## The following object is masked from 'package:base':
##
##     date
```

```
activity$date <- ymd(activity$date)
summary(activity)
```

```
##      steps             date               interval
##  Min.   :  0.00   Min.   :2012-10-01   Min.   :   0.0
##  1st Qu.:  0.00   1st Qu.:2012-10-16   1st Qu.: 588.8
##  Median :  0.00   Median :2012-10-31   Median :1177.5
##  Mean   : 37.38   Mean   :2012-10-31   Mean   :1177.5
##  3rd Qu.: 12.00   3rd Qu.:2012-11-15   3rd Qu.:1766.2
##  Max.   :806.00   Max.   :2012-11-30   Max.   :2355.0
##  NA's   :2304
```

A Histogram of the Total Number of Steps Taken Each Day

```
steps_date <- tapply(activity$steps,activity$date, FUN = sum, na.rm = TRUE)
hist(steps_date, breaks = 10)
```



The Mean and Median Number of Steps Taken Each Day

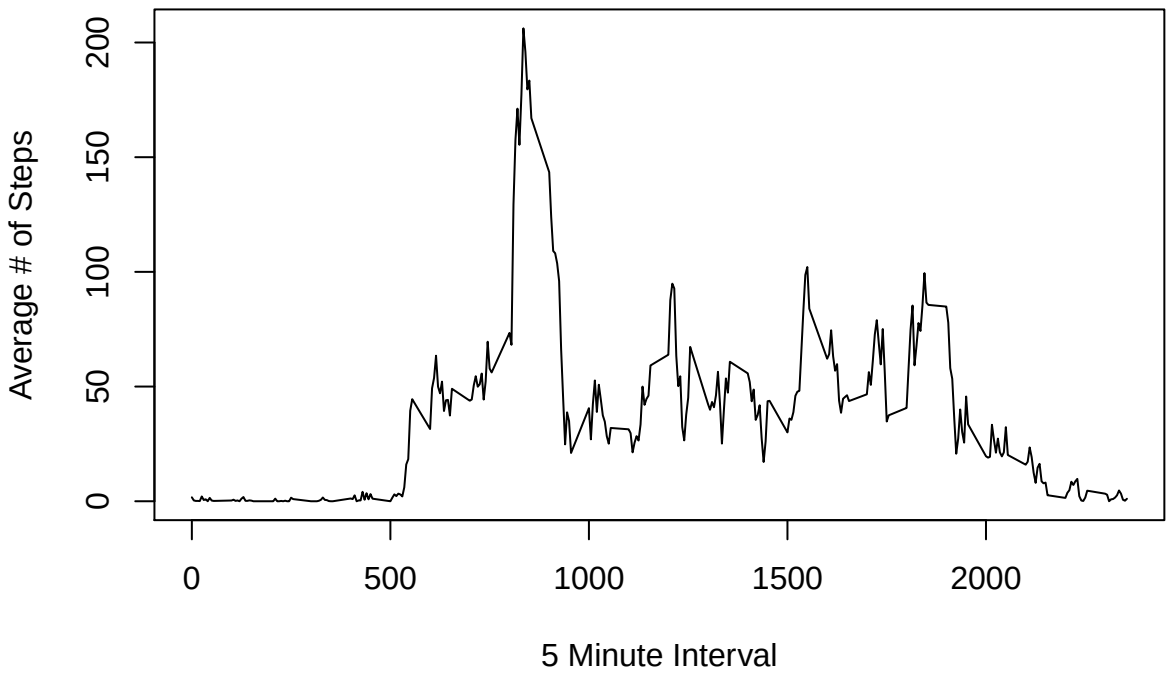
```
mean <- mean(steps_date, na.rm = TRUE)
median <-median(steps_date, na.rm = TRUE)
vals <- c(mean,median)
title <- c("mean","median")
mmtable <- rbind(title,vals)
print(mmtable)
```

```
##       [,1]               [,2]
## title "mean"             "median"
## vals  "9354.22950819672" "10395"
```

A time series plot of the average number of steps taken

I use Taply to create a data frame which describes the average steps at each 5 minute time interval.

```
steps_time <- tapply(activity$steps,activity$interval, FUN = mean, na.rm = TRUE)
ints <-  activity$interval[1:288]
steps_time_f <- cbind(steps_time,ints)
steps_time_f<- as.data.frame(steps_time_f)
plot(steps_time_f$ints,steps_time_f$steps_time, type = "l", xlab = "5 Minute Interval", ylab = "Average
```



The 5-minute interval that, on average, contains the maximum number of steps

```
maxinter <- max(steps_time_f$steps_time)
steptimesmax <- subset(steps_time_f, steps_time_f$steps_time== maxinter)
print(steptimesmax)
```

```
##     steps_time ints
## 835   206.1698  835
```

Code to estimate the # of missing values

```
missingv <- sum(is.na(activity$steps))
print(missingv)
```

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

Code to describe and show a strategy for imputing missing data

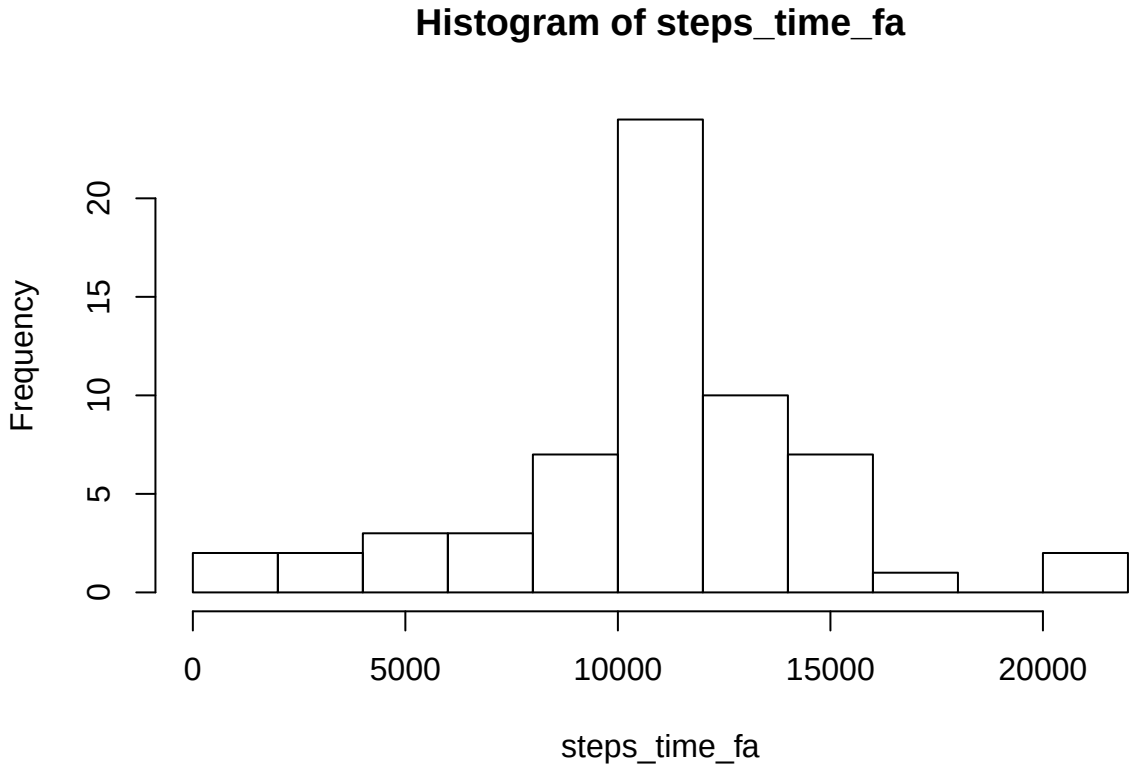
I use Transform to replace all N.A values in the Steps aray of the Activity data frame with the average steps over the relevant time interval.

```
activity_nar <- transform(activity, steps = ifelse(is.na(activity$steps), yes = steps_time_f$steps_time
summary(steps_time_f)
```

```
##    steps_time           ints
##  Min.   :  0.000   Min.   :   0.0
##  1st Qu.:  2.486   1st Qu.: 588.8
##  Median : 34.113   Median :1177.5
##  Mean   : 37.383   Mean   :1177.5
##  3rd Qu.: 52.835   3rd Qu.:1766.2
##  Max.   :206.170   Max.   :2355.0
```

A histogram of the total number of steps taken each day after missing values are imputed

```
steps_time_fa <- tapply(activity_nar$steps,activity_nar$date, FUN = sum)
hist(steps_time_fa, breaks =  10)
```



A panel plot comparing the average number of steps taken per 5-minute interval across weekdays and weekends

I used the wday function from the lubridate library to add a weekday array to my transformed activity data frame. I then used that frame to create two new objects; an activity data frame for weekdays, and an activity data frame for weekends. I then used the Tapply function to create two new arrays describing the average steps for each time interval within the weekend and weekeday arrayS, and plotted both of those series.

```
activity_nar$wday <- wday(activity_nar$date)
activity_nar_wend <- subset(activity_nar, wday == 1 | wday == 7)
```

```
activity_nar_wday <- subset(activity_nar, wday != 1 & wday != 7)
wday_steps_time <- tapply(activity_nar_wday$steps,activity_nar_wday$interval, FUN = mean, na.rm = TRUE)
wend_steps_time <- tapply(activity_nar_wend$steps,activity_nar_wend$interval, FUN = mean, na.rm = TRUE)
day_frame <- as.data.frame( cbind(wday_steps_time,wend_steps_time,ints))
plot(day_frame$ints,day_frame$wday_steps_time, type = "l", col = "red")
lines(day_frame$ints,day_frame$wend_steps_time, col = "blue")
title(main = "Average Steps at Time Intervals")
legend("topleft",c("weekdays","weekends"),fill =c("red","blue"))
```