Untitled

Darrr

July 30, 2016

Basic settings

```
echo = TRUE  # Always make code visible
options(scipen = 1)  # Turn off scientific notations for numbers
```

1. Code for reading in the dataset and/or processing the data

```
data<-read.csv("./activity.csv", colClasses = c("integer", "Date", "factor"))

noNA<-na.omit(data)

head(noNA)
```

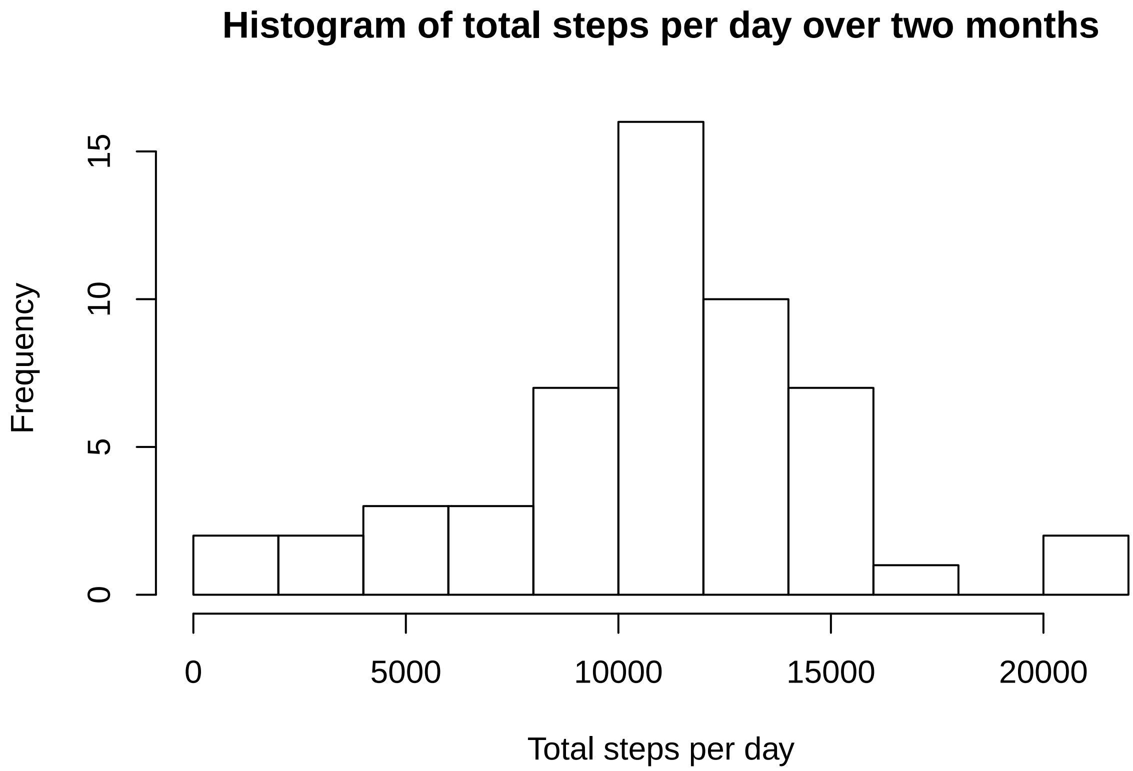
```
##     steps       date interval
## 289     0 2012-10-02        0
## 290     0 2012-10-02        5
## 291     0 2012-10-02       10
## 292     0 2012-10-02       15
## 293     0 2012-10-02       20
## 294     0 2012-10-02       25
```

```
dim(noNA)
```

```
## [1] 15264     3
```

2. Histogram of the total number of steps taken each day

```
ttlnum <- sapply(split(noNA$steps, noNA$date), sum)

hist(ttlnum, breaks=10,
     main="Histogram of total steps per day over two months",
     xlab="Total steps per day")
```

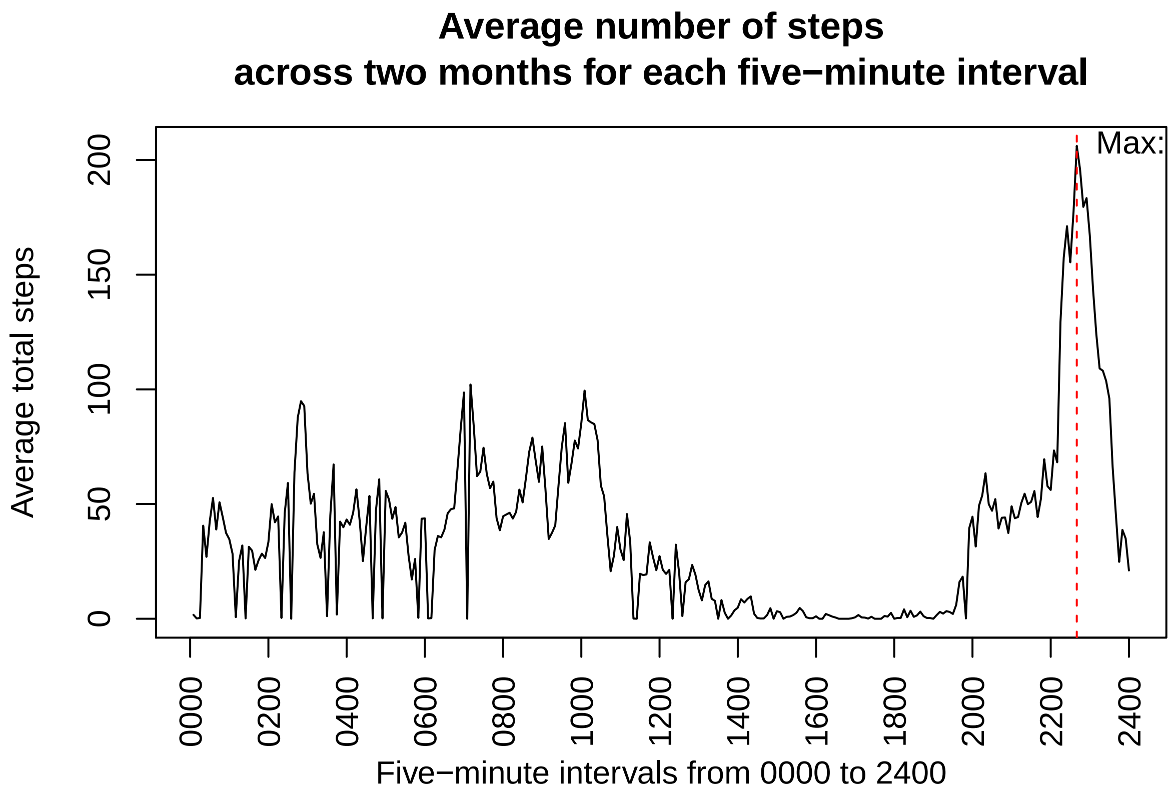


3. Mean and median number of steps taken each day

```
mean_ttlnum<-mean(ttlnum) #10766.19
median_ttlnum<-median(ttlnum) #10765
```

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

```
tspav<- sapply(split(noNA$steps, noNA$interval), mean)
plot(tspav, type = "l", xaxt="n",
     main="Average number of steps\nacross two months for each five-minute interval",
     ylab="Average total steps",
     xlab="Five-minute intervals from 0000 to 2400")


axis(1, at=seq(0, length(tspav), by=24), labels=formatC(seq(0,2400,by=200), width=4, flag="0"), las=2)
max.index = which.max(tspav)
max.xtick = names(max.index)
abline(v=max.index, col="red", lty=2)
text(max.index, tspav[max.index], paste0("Max: ", formatC(tspav[max.index], digits=5), " steps"), pos=4
```

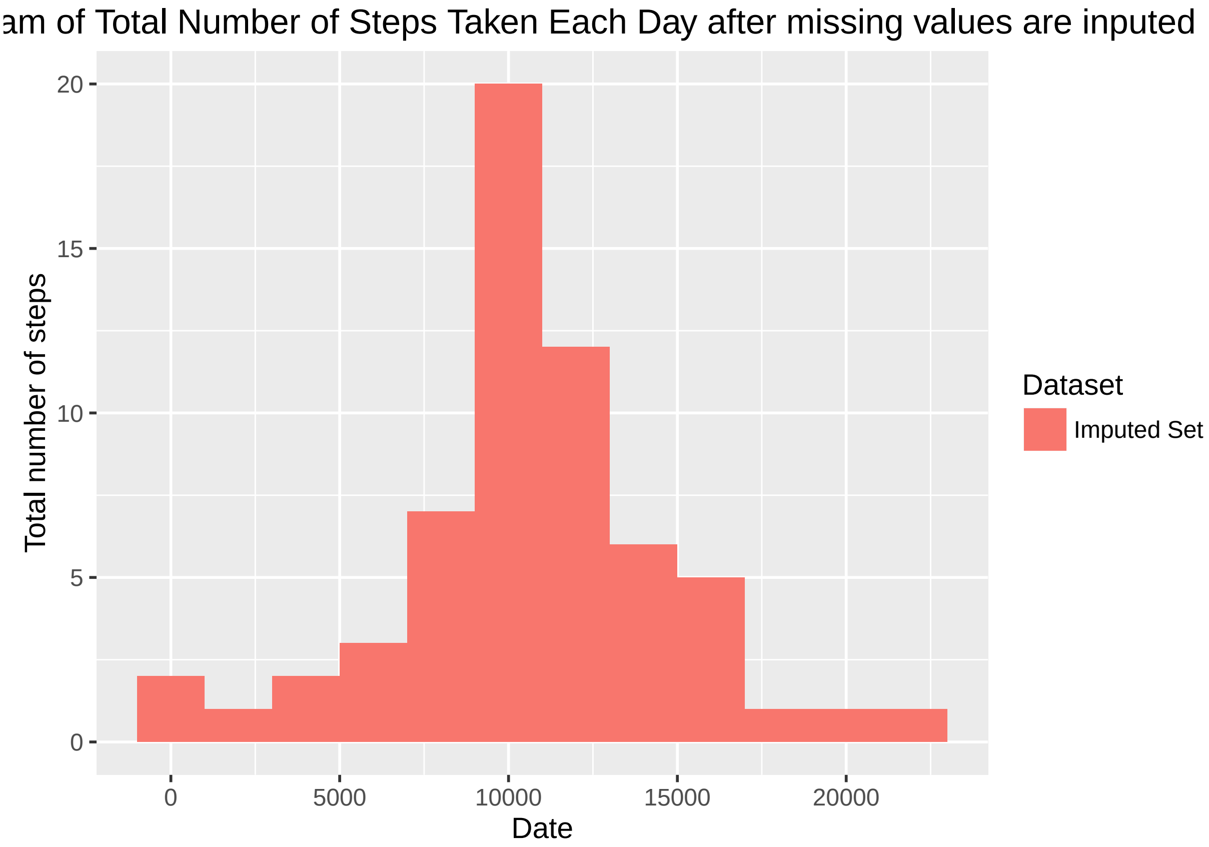


## 5. Code to describe and show a strategy for imputing missing data

```
nas<-is.na(data$steps)
sum_nas<-sum(nas) #2304

# Copy data frame to create imputed dataset.
data_imp<-data
# Replace NA values with the interval average from the previous section.
rep.times = length(data$steps) / length(tspav)  #17568 / 288 = 61
data_imp[nas,"steps"] <- rep(tspav, times=rep.times)[nas]
```

6. Histogram of the total number of steps taken each day after missing values are imputed

```
library(ggplot2)

impmv <- sapply(split(data_imp$steps, data_imp$date), sum)
df_imp <- data.frame(ave_steps=impmv, Dataset="Imputed Set", row.names=NULL)

ggplot(df_imp, aes(x=ave_steps, fill=Dataset)) + geom_histogram(binwidth=2000, position="dodge") + labs
```



```
data_imp_mean = mean(impmv)#10766.19
data_imp_median = median(impmv)#10766.19
```

The mean and median total number of steps taken per day remains the same after imputing the dataset. They are 10766.19 and 10765 respectively.

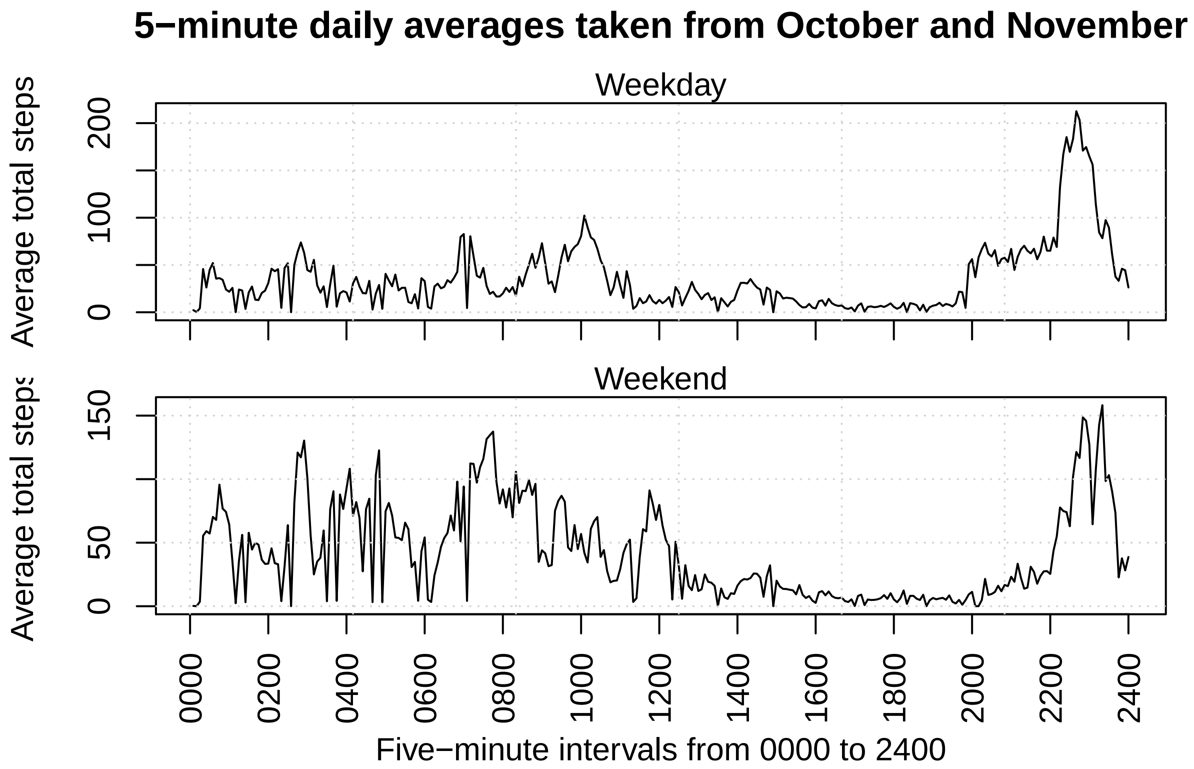
On the otherhand, the count of the daily totals is different. Because the NA data was filled with the daily means, the count of the mean total is higher for the imputed dataset as seen in the graph.

7. Panel plot comparing the average number of steps taken per 5-minute interval across weekdays and weekends

```
# Add a weekday/weekend factor column to dataset.
days = weekdays(data_imp$date)
weekend = c("Saturday", "Sunday")
data_imp$week___ = factor(days %in% weekend, labels=c("weekday", "weekend"))
str(data_imp)
```

```
## 'data.frame':    17568 obs. of  4 variables:
##  $ steps   : num  1.717 0.132 0.321 40.566 26.981 ...
##  $ date    : Date, format: "2012-10-01" "2012-10-01" ...
##  $ interval: Factor w/ 288 levels "0","10","100",..: 1 226 2 73 136 195 198 209 212 223 ...
##  $ week___ : Factor w/ 2 levels "weekday","weekend": 1 1 1 1 1 1 1 1 1 1 ...
```

```r
# Split the data into weekday and weekend groups and find 5-minute averages for each.
split.act = split(data_imp, data_imp$week___)
df = data.frame(sapply(split.act, function(x) sapply(split(x$steps, x$interval), mean)))

# Plot the results side-by-side.
# Plot 'weekday' data
par(mfrow=c(2,1), mar=c(1.5,4.1,4.1,2.1))
plot(df$weekday, type="l", xaxt="n",
     main="5-minute daily averages taken from October and November",
     ylab="Average total steps")
mtext("Weekday")
axis(1, at=seq(0, length(tspav), by=24), labels=rep("", times=13), las=2)
grid()
# Plot 'weekend' data
par(mar=c(5.1,4.1,0.5,2.1))
plot(df$weekend, type="l", xaxt="n",
     ylab="Average total steps",
     xlab="Five-minute intervals from 0000 to 2400")
mtext("Weekend")
axis(1, at=seq(0, length(tspav), by=24), labels=formatC(seq(0,2400,by=200), width=4, flag="0"), las=2)
grid()
```


5−minute daily averages taken from October and November
Weekday
Average total steps
Weekend
Average total steps
Five−minute intervals from 0000 to 2400