PA1_template.Rmd

B. Kavalar

September 3, 2017

First Project for Reproducable Resesarch for JHU Data Science Course

This first chuck of code will answer the first question in the assignment:

What is mean total number of steps taken per day?

```r
#set working directory
setwd("C:/Backup/2017 IRAD/R Programming/JHU Data Science Course/Reproducible Research/Week 2/Project 1
knitr::opts_chunk$set(fig.path = './figure/')

library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.4.1
```

```
##
## 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
```

```r
library(lattice)
library(knitr)
```

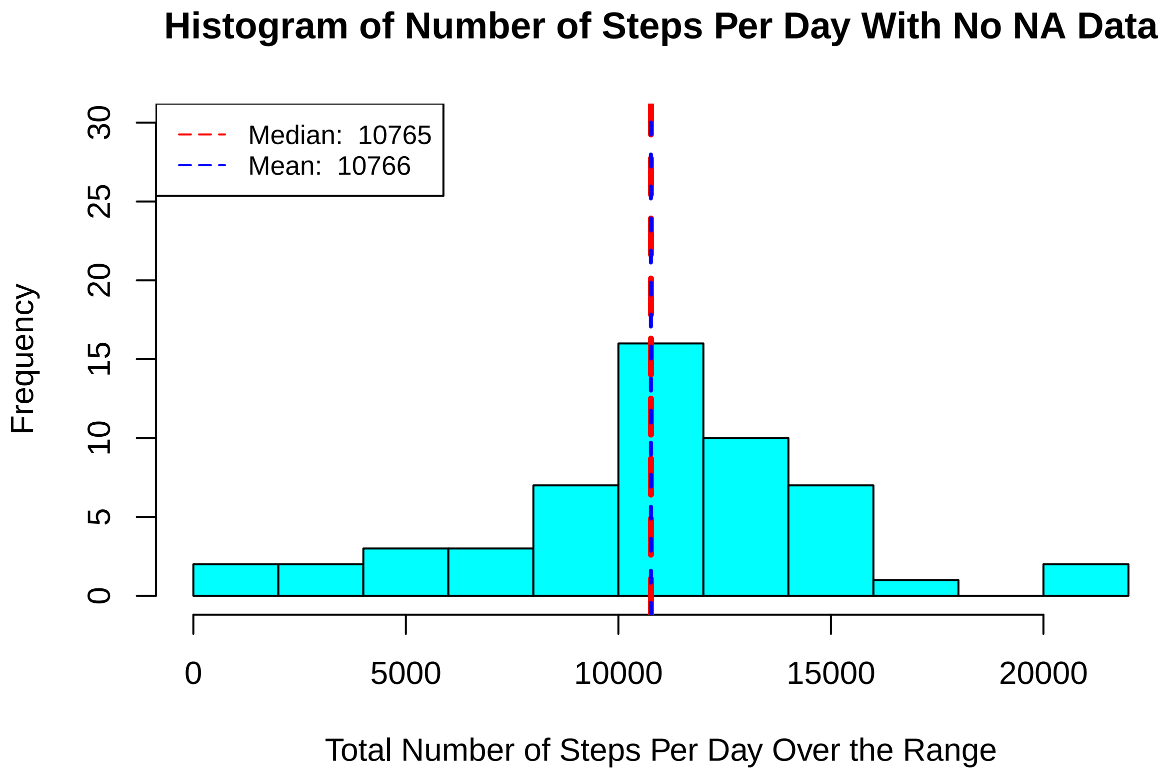
```
## Warning: package 'knitr' was built under R version 3.4.1
```

```r
#read in data set
fitData <- read.csv("activity.csv")

#remove rows with NA data
fitData <- fitData[complete.cases(fitData), ]

#put date in as.Date format and use aggregate function to sum total steps per day
fitData$date <- as.Date(fitData$date)
sumData <- aggregate(fitData$steps, by=list(fitData$date), FUN=sum)

#add meaningful names to the columns
names(sumData)[1] <- "Date"
names(sumData)[2] <- "Total_Steps"

#convert steps to numeric from integer
sumData$Total_Steps <- as.numeric(as.integer(sumData$Total_Steps))
```

What is mean total number of steps taken per day?

Create histogram of the number of steps per day and the frequency

```
hist(sumData$Total_Steps, col = "cyan", xlab = "Total Number of Steps Per Day Over the Range",
     ylim = range(0, 30), main="Histogram of Number of Steps Per Day With No NA Data",
     breaks = 10)

#mean and median info for total steps
medianSteps <- median(sumData$Total_Steps)
meanSteps <- mean(sumData$Total_Steps)
firstQuant <- quantile(sumData$Total_Steps, probs = c(.25))
thirdQuant <- quantile(sumData$Total_Steps, probs = c(.75))

abline(v=meanSteps, lwd = 3, lty = 5, col = 'red')
abline(v=medianSteps, lwd = 2, lty = 2, col = 'blue')

legend('topleft', lty = 5, lwd = 1, col = c("red", "blue"), cex = .8,
       legend = c(paste('Median: ', medianSteps),
               paste('Mean: ', format(meanSteps, scientific=FALSE, digits = 5))))
```



```
dev.copy(png, file = "./figure/hist1.png")
```

```
## png
##   3
```

```r
dev.off()
```

```
## pdf
##   2
```

The mean number of steps is 10766.

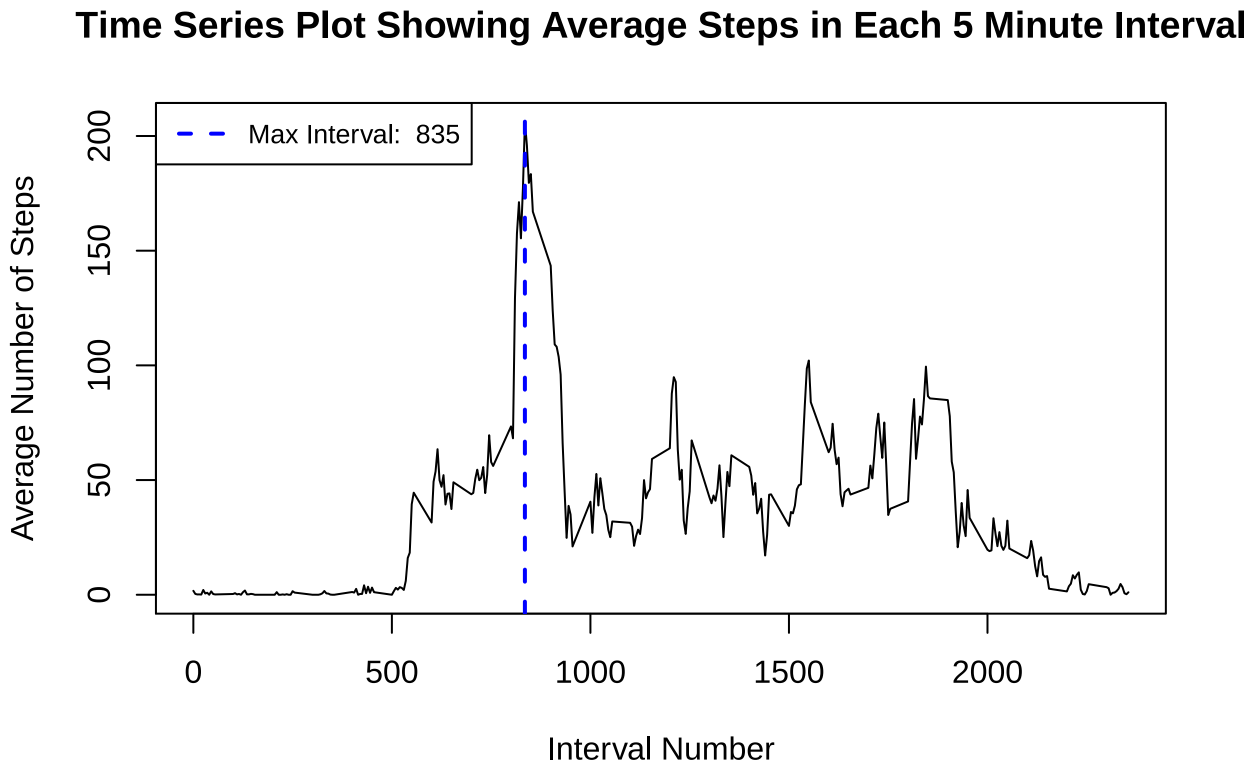
The median number of steps is 10765.

The next chuck of code will answer the second question:

What is the average daily activity pattern?

Make a time series plot (i.e. type = “l”) of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis)

```r
#find mean number of steps per interval across all dates
meanData <- aggregate(fitData$steps, by=list(fitData$interval), mean)
names(meanData)[1] <- "Interval"
names(meanData)[2] <- "Average_Steps"

#plot the average daily activity pattern
plot(meanData$Average_Steps ~ meanData$Interval, type="l",
     xlab = "Interval Number", ylab = "Average Number of Steps",
     main = "Time Series Plot Showing Average Steps in Each 5 Minute Interval")

#find interval value at maximum peak of steps
InterValMax <- meanData$Interval[meanData$Average_Steps==max(meanData$Average_Steps)]

abline(v=InterValMax, lwd = 2, lty = 2, col = 'blue')

legend('topleft', lty = 2, lwd = 2, col = c("blue"),
               cex = .8,
               legend = c(paste('Max Interval: ', InterValMax)))
```



```r
dev.copy(png, file = "./figure/plot1.png")
```

```
## png
##   3
```

```r
dev.off()
```

```
## pdf
##   2
```

Next we find the 5-minute interval, on average across all the days in the dataset,

contains the maximum number of steps.

The interval number with maximum number of steps is 835.

Next we discuss the question: Imputing missing values.

Calculate and report the total number of missing values in the dataset (the total number of rows with NAs).

```r
#read in data set
fitData <- read.csv("activity.csv")
```

```r
#find the number rows with incomplete data
IncompData <- fitData[!complete.cases(fitData), ]

#find number of incomplete cases - rows with NAs using "not" complete.cases
IncompRows <- nrow(fitData[!complete.cases(fitData), ])
```

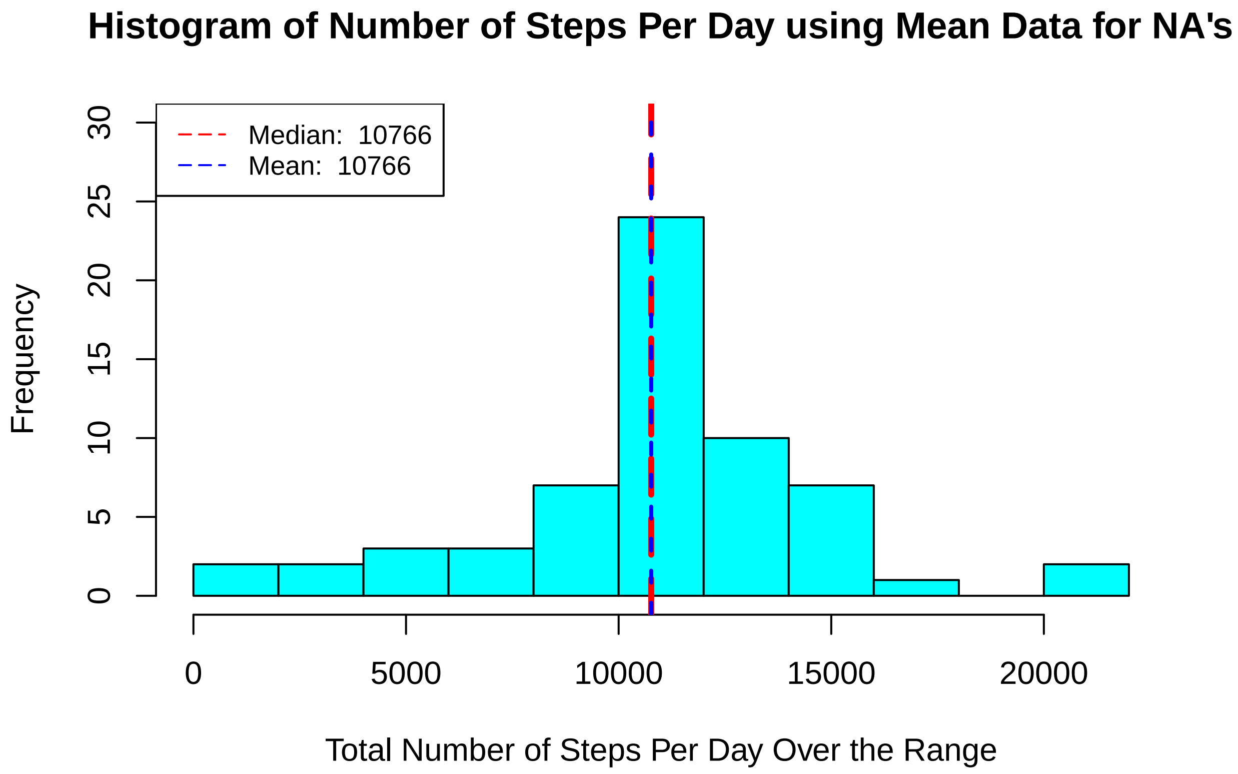
The total number of rows with missing data is 2304.

Next we devise a strategy for filling in all of the missing values in the dataset. The strategy chosen was to use the mean step value for the data set since many dates have NA only.

Create a new dataset that is equal to the original dataset but with the missing data filled in.

```r
#in this section we read in the data set and don't clear out NA rows
fitData <- read.csv("activity.csv")

#find mean number of steps to use in replacement of NA's using previous calculations
meanStepValue <- mean(meanData$Average_Steps)

#Create new fitData set with NA step data filled in with the mean number of
#steps dervied from the data above
fitData[, 1][is.na(fitData[, 1])] <- meanStepValue

#put date in as.Date format and use aggregate function to sum total steps per day
fitData$date <- as.Date(fitData$date)
sumData <- aggregate(fitData$steps, by=list(fitData$date), FUN=sum)

#add meaningful names to the columns
names(sumData)[1] <- "Date"
names(sumData)[2] <- "Total_Steps"

#convert steps to numeric from integer
sumData$Total_Steps <- as.numeric(as.integer(sumData$Total_Steps))

#What is mean total number of steps taken per day?
#create histogram of the number of steps per day and the frequency
hist(sumData$Total_Steps, col = "cyan", xlab = "Total Number of Steps Per Day Over the Range",
     ylim = range(0, 30), main="Histogram of Number of Steps Per Day using Mean Data for NA's",
     breaks = 10)

#mean and median info for total steps
medianSteps2 <- median(sumData$Total_Steps)
meanSteps2 <- mean(sumData$Total_Steps)
firstQuant2 <- quantile(sumData$Total_Steps, probs = c(.25))
thirdQuant2 <- quantile(sumData$Total_Steps, probs = c(.75))

abline(v=meanSteps2, lwd = 3, lty = 5, col = 'red')
abline(v=medianSteps2, lwd = 2, lty = 2, col = 'blue')
```

```
legend('topleft', lty = 5, lwd = 1, col = c("red", "blue"),
               cex = .8,
               legend = c(paste('Median: ', medianSteps2),
                          paste('Mean: ', format(meanSteps, scientific=FALSE, digits = 5))))
```



```
dev.copy(png, file = "./figure/hist2.png")
```

```
## png
##   3
```

```
dev.off()
```

```
## pdf
##   2
```

The mean number of steps using imputed data is 10766.

The median number of steps using imputed data is 10766.

How do these values differ from the estimates from the first part of the assignment? What is the impact of imputing missing data on the estimates of the total daily number of steps?

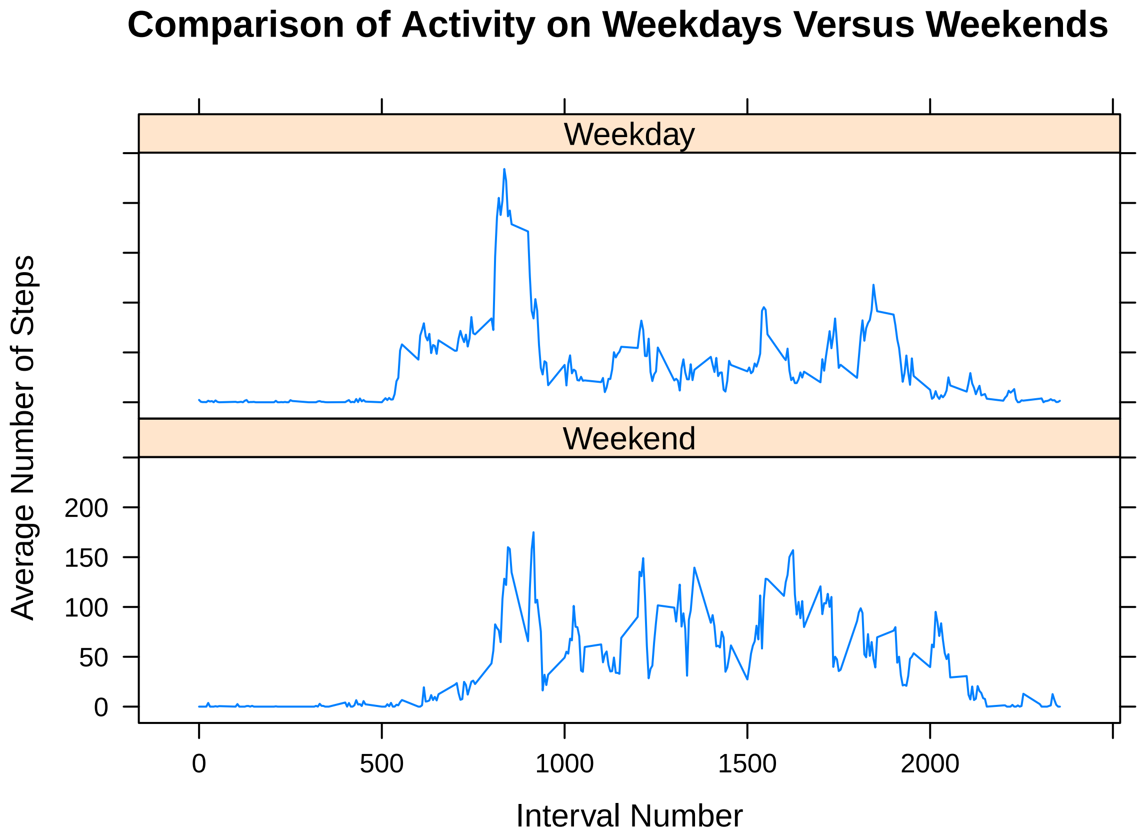
Since we used the mean value to replace the NA's, the mean and median are pretty much the same as before. But as expected the 1st and 3rd quantiles are not the same since the data is now more biased towards the center with more mean data values used as shown in the table below:

Quantiles Using Raw Data	Quantitles Using Imputed Data
1st Quant is: 8841	**1st Quant is: 9819**
3rd Quant is: 13294	**3rd Quant is: 12811**

Are there differences in activity patterns between weekdays and weekends?

Make a panel plot containing a time series plot (i.e. type = "l") of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all weekday days or weekend days (y-axis).

```r
#in this section we read in the data set and don't clear out NA rows
fitData <- read.csv("activity.csv")

#create a data frame for weekdays
weekdayData <- c('Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday')

#Use `%in%` and weekdays and use factor function and add levels and labels
fitData$workDay <- factor((weekdays(as.Date(fitData$date)) %in% weekdayData),
                          levels = c(FALSE, TRUE), labels = c('Weekend', 'Weekday'))

#aggreate data related to mean number of steps using both interval and workday
newMeanData <- aggregate(steps ~ interval + workDay , fitData , mean )

#use lattice function to create dual plot showing both weekend and weekday values
print( xyplot((newMeanData$steps ~ newMeanData$interval|newMeanData$workDay),
              type='l', layout=c(1,2),
              xlab='Interval Number', ylab='Average Number of Steps',
              main="Comparison of Activity on Weekdays Versus Weekends"))
```



```r
dev.copy(png, file = "./figure/plot2.png")
```

```
## png 
##   3
```

```r
dev.off()
```

```
## pdf 
##   2
```