Reproducible research - assignment 1

Adrian Marcia

July 9, 2017

Loading raw data and library packages for the assignment

Data downloaded from Coursera Data Science Program - Reproducible Research

Reading the .csv data from the link above into variable activity_data

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

Loading the plyr, ggplot2 and lattice packages for data manipulation and plotting

```
library(plyr)
library(ggplot2)
library(lattice)
```

Cleaning activity_data

Removing the 'NA' data from the activity_data and storing the clean data in clean_data

```
clean_data <- activity_data[!is.na(activity_data$steps),]
```

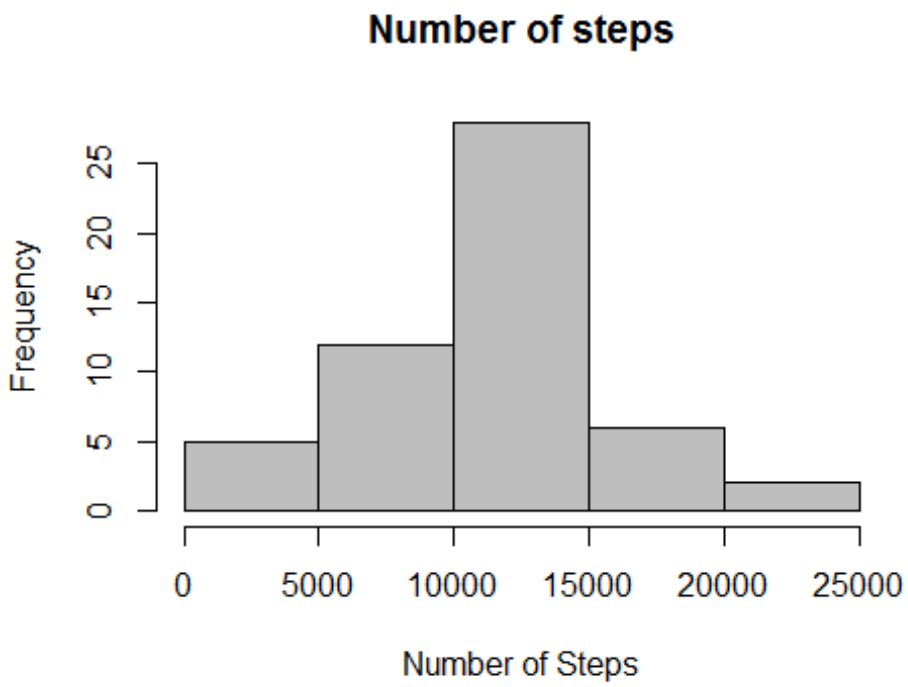
Question 1.1: *What is the total number of steps taken per day?*

Calculates the total number of steps per day

```
total_steps_per_day <- aggregate(clean_data$steps ~ clean_data$date, FUN=sum,)
```

Histogram of the total number of steps per day

```
hist(total_steps_per_day$`clean_data$steps`, breaks = 7, xlab = "Number of Steps", main =
"Number of steps", col = "grey")
```



Mean and median number of steps

The mean of is calculated below

```
mean <- as.integer(mean(total_steps_per_day$`clean_data$steps`))
```

Producing a resulf of 10766

The median is calculated below

```
median <- as.integer(median(total_steps_per_day$`clean_data$steps`))
```

Producing a resulf of 10765

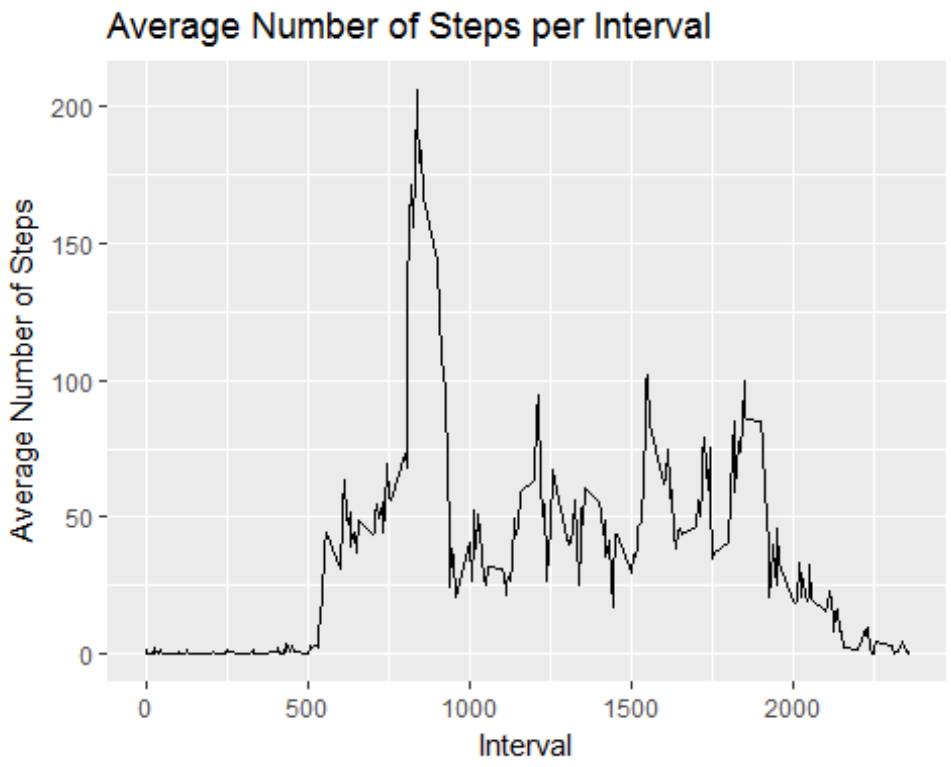
Question 1.2: *What is the average daily activity pattern?*

Create an interval table with average steps for each interval

```
interval_table_data <- ddply(clean_data, .(interval), summarize, Avg = mean(steps))
```

Create a line plot with the intervals and their average number of steps

```
plot <- ggplot(interval_table_data, aes(x = interval, y = Avg), xlab = "Interval",
ylab="Average Number of Steps")
plot + geom_line() + xlab("Interval") + ylab("Average Number of Steps") + ggtitle("Average
Number of Steps per Interval")
```



Returns the maximum number of steps per interval

```
max_steps <- max(interval_table_data$Avg)
```

The maximum number of steps per interval was 206.1698113

Returns the maximum number of steps per interval

```
max_interval <- interval_table_data[interval_table_data$Avg == max_steps, 1]
```

The interval where the highest number of steps occured is 835

Question 1.3: *Imputing missing values*

Calculate the number of missing rows using the nrow function on the steps column in the activity_data table

```
missing <- nrow(activity_data[is.na(activity_data$steps),])
```

The number of missing values in the steps data is 2304

Subsetting all missing data from activity_data, assigning the day of the week as a new column.

```
missing_data <- activity_data[is.na(activity_data),]
missing_data$day <- weekdays(as.Date(missing_data$date))
```

Calculating the average for a day from clean data

```
clean_data$day <- weekdays(as.Date(clean_data$date))
average_day <- ddply(clean_data, .(day), summarize, Avg = mean(steps))
```

Replace all missing variables with the averages from the clean data set and reorder and rename the data into the format per the clean data set

```
replaced_data <- merge(missing_data, average_day, by=c("day"))
replaced_data <- replaced_data[,c(5,3,4,1)]
colnames(replaced_data)<- c("steps", "date", "interval", "day")
```

Merge the clean data set and the replaced data sets

```
combined_data <- rbind(clean_data, replaced_data)
```

Calculate the mean number of steps

```
mean_imputed <- as.integer(mean(combined_data$steps))
```

The mean number of steps is 37. Creating a change of -10729 steps.

Calculate the median number of steps

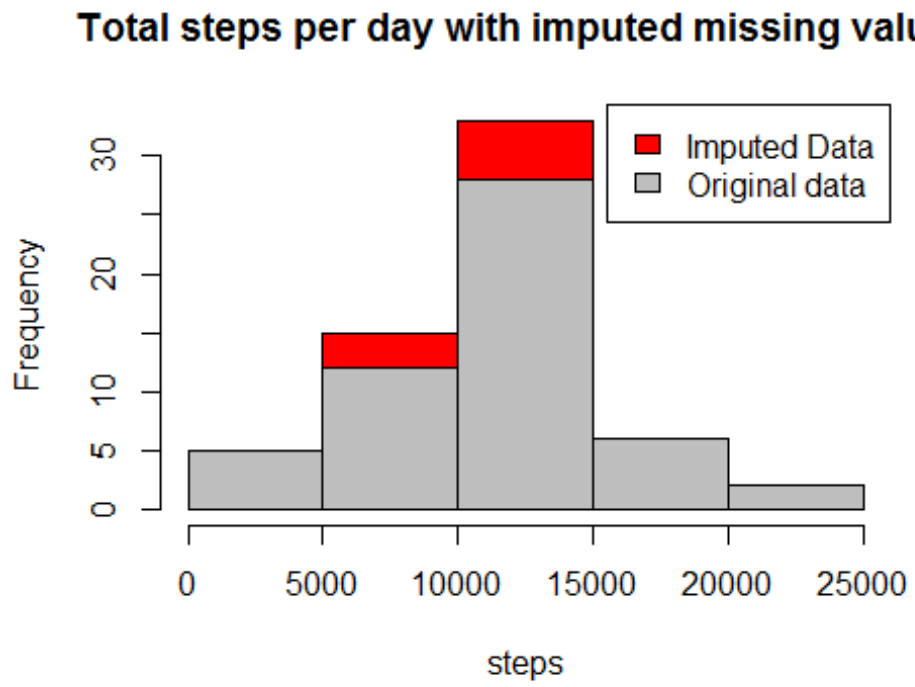
```
median_imputed <- as.integer(median(combined_data$steps))
```

The mean number of steps is 0. Creating a change of -10765 steps.

Graph the new histogram and show the differences in the imputed data

```
imputed_steps_per_day <- aggregate(combined_data$steps ~ combined_data$date, FUN=sum,)

hist(imputed_steps_per_day$`combined_data$steps`, breaks = 7, xlab = "steps", main = "Total
steps per day with imputed missing values", col = "red")
hist(total_steps_per_day$`clean_data$steps`, breaks = 7, xlab = "steps", main = "Total steps
per day with imputed missing values", col = "grey", add = TRUE)
legend("topright", c("Imputed Data", "Original data"), fill=c("red", "grey") )
```



Question 1.4: *Differences in activity patterns between weekdays and weekends*

Assign combined_data into weekdays and weekends

```
combined_data$catagorical_day <- ifelse(combined_data$day %in% c("Saturday", "Sunday"),
"Weekend", "Weekday")
```

Create an intervalof the weekend and weekday data

```
catagorical_week_intervals <- ddply(combined_data, .(interval, catagorical_day), summarize,
Avg = mean(steps))
```

Plot graph of catagorical intervals comparing week and weekend data

```
xyplot(Avg ~ interval | catagorical_day, data = catagorical_week_intervals, type="l",  layout
= c(1,2), main = "Average Steps per Interval Based on Type of Day", ylab="Average Number of
Steps", xlab="Interval")
```