Reproducible Research Assignment 1

Dave LeBaron

March 31, 2016

This is an R Markdown document produced to complete the first assignment in the Coursera Reproducible Research course

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.2.4
```

```
library(dplyr)
```

```
##
## 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
```

```
library(mice)
```

```
## Warning: package 'mice' was built under R version 3.2.4

## Loading required package: Rcpp
## mice 2.25 2015-11-09
```

```
library(gridExtra)
```

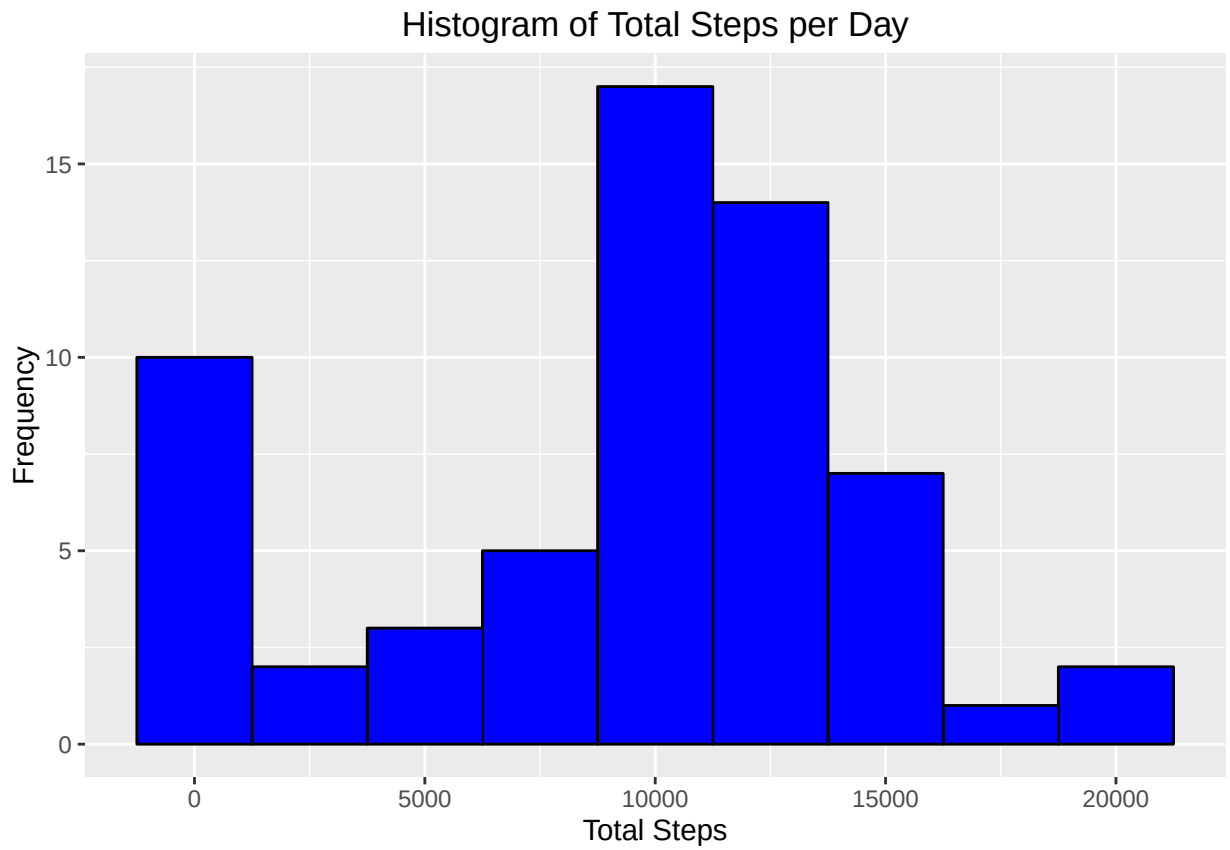
```
## Warning: package 'gridExtra' was built under R version 3.2.4

##
## Attaching package: 'gridExtra'
##
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
#read in data
activity <- read.csv("activity.csv", stringsAsFactors = FALSE)
```

Histogram of the total number of steps taken each day:

```
stepsbyday <- tapply(activity$steps, activity$date, FUN=sum, na.rm = TRUE)
qplot(stepsbyday, geom =  "histogram", binwidth=2500, main = "Histogram of Total Steps per Day", xlab =
```

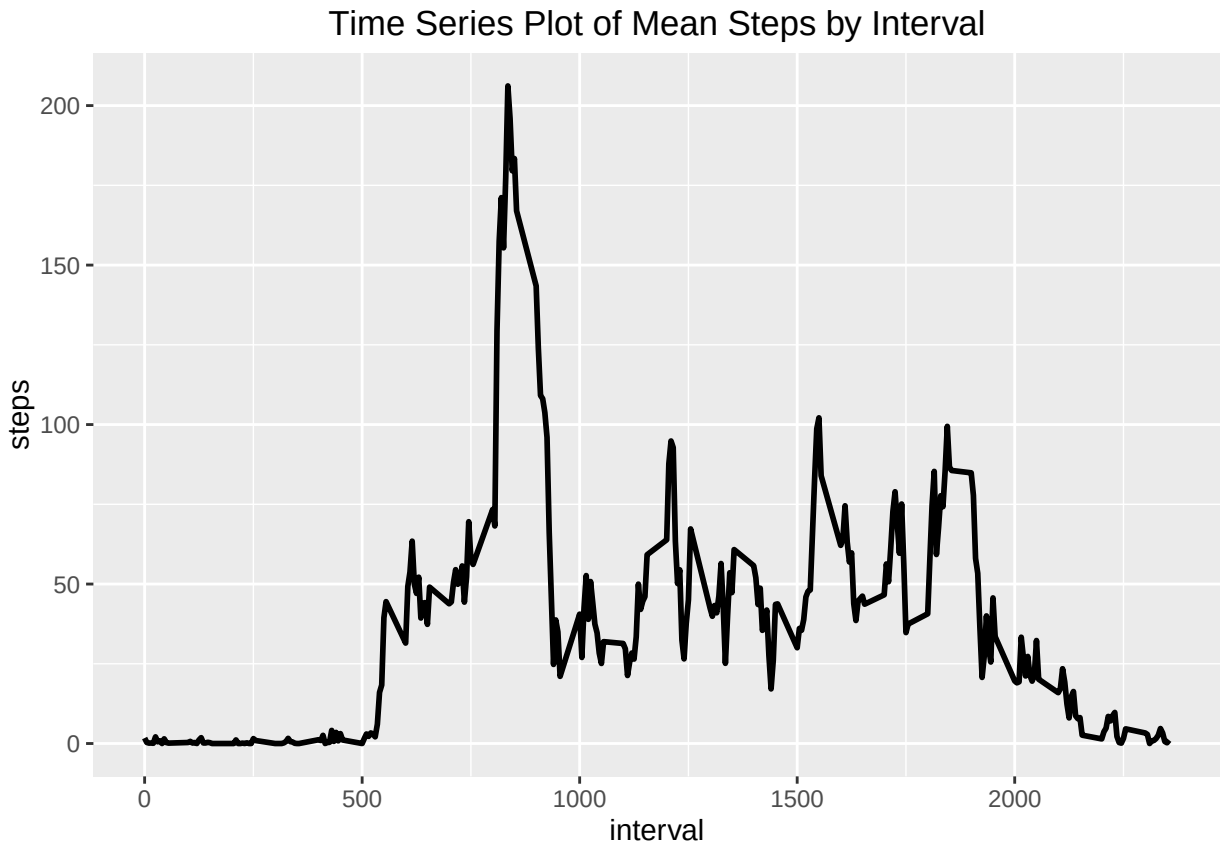


Mean and median of steps taken each day:

```
mean_steps <- mean(stepsbyday, na.rm = TRUE)
#mean steps = 9354.23
med_steps <- median(stepsbyday, na.rm = TRUE)
#median steps = 10395
```

Time series plot of the average number of steps taken:

```
activity_sub <- aggregate(steps ~ interval, activity, mean)
ggplot(data=activity_sub, aes(x=interval, y=steps), xlab = "Interval", ylab = "Mean Steps") + ggtitle("
```



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

```
activity_sub[which.max(activity_sub$steps),c("interval")]
```

```
## [1] 835
```

```
#The interval is 835, average number of steps = 206.1698
```

Code to describe and show a strategy for imputing missing data:

```
missingVal <- sum(is.na(activity$steps))
#There are 2304 missing values
#The mice package is used to impute missing values
set.seed(1234)
activity_sub2 <- subset(activity, select = c(steps, interval))
activity_imp <- complete(mice(activity_sub2))
```

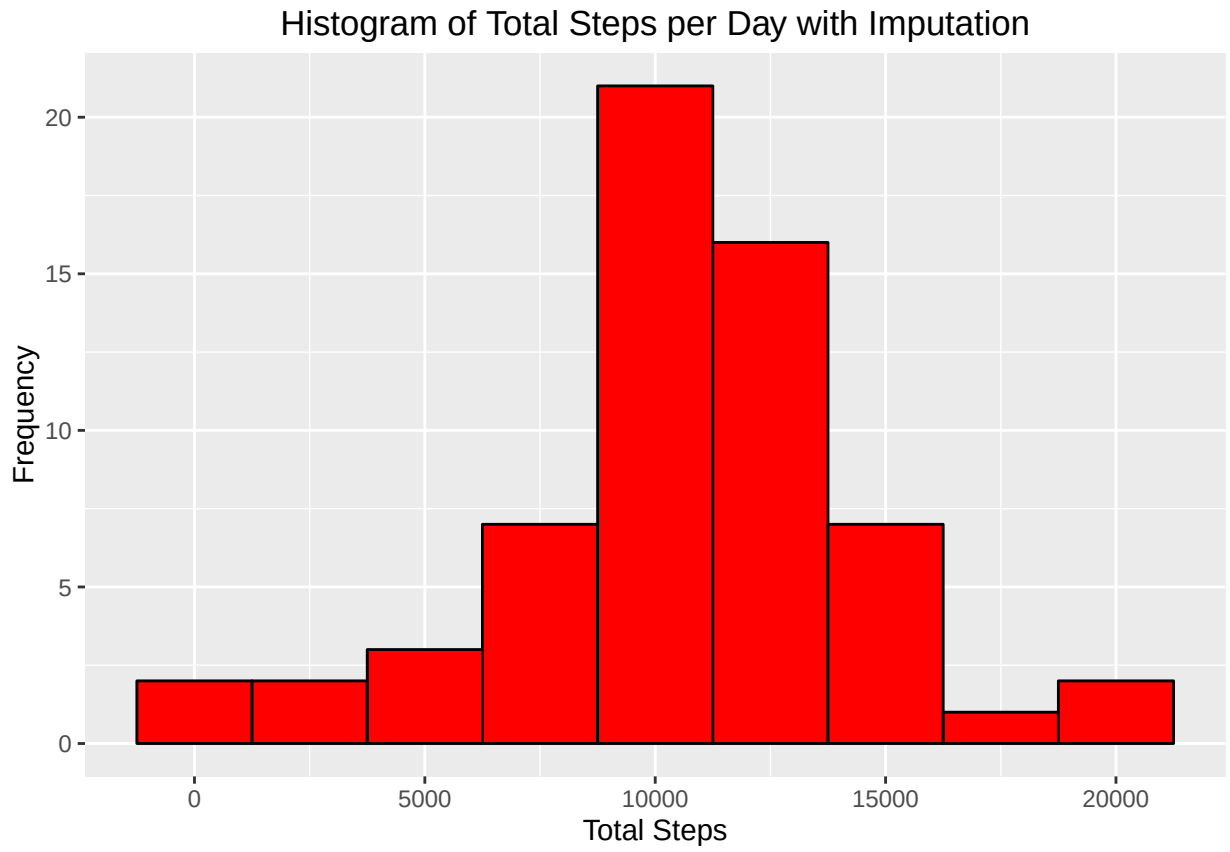
```
##
##  iter imp variable
##    1   1  steps
##    1   2  steps
##    1   3  steps
##    1   4  steps
##    1   5  steps
##    2   1  steps
```

```
##   2   2  steps
##   2   3  steps
##   2   4  steps
##   2   5  steps
##   3   1  steps
##   3   2  steps
##   3   3  steps
##   3   4  steps
##   3   5  steps
##   4   1  steps
##   4   2  steps
##   4   3  steps
##   4   4  steps
##   4   5  steps
##   5   1  steps
##   5   2  steps
##   5   3  steps
##   5   4  steps
##   5   5  steps
```

```
activity_imp$date <- activity$date
```

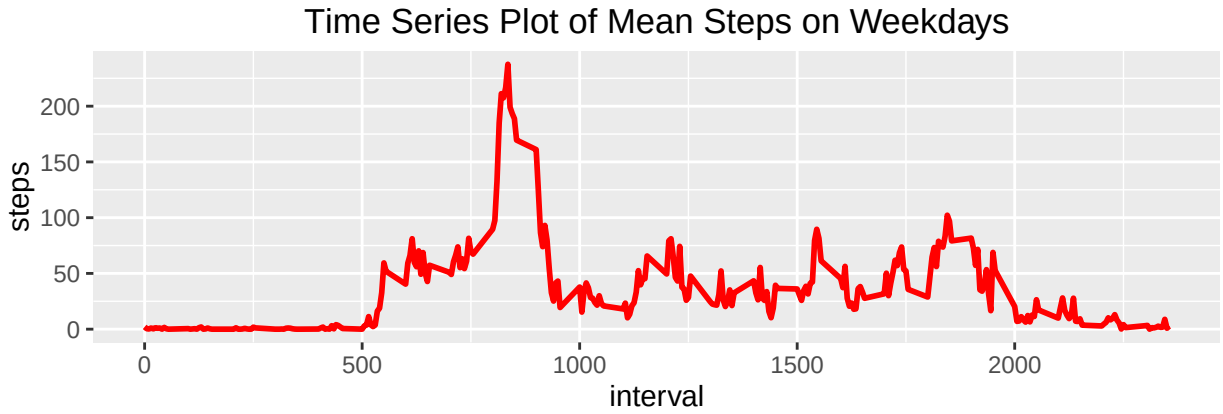
Histogram of the total number of steps taken each day after missing values are imputed:

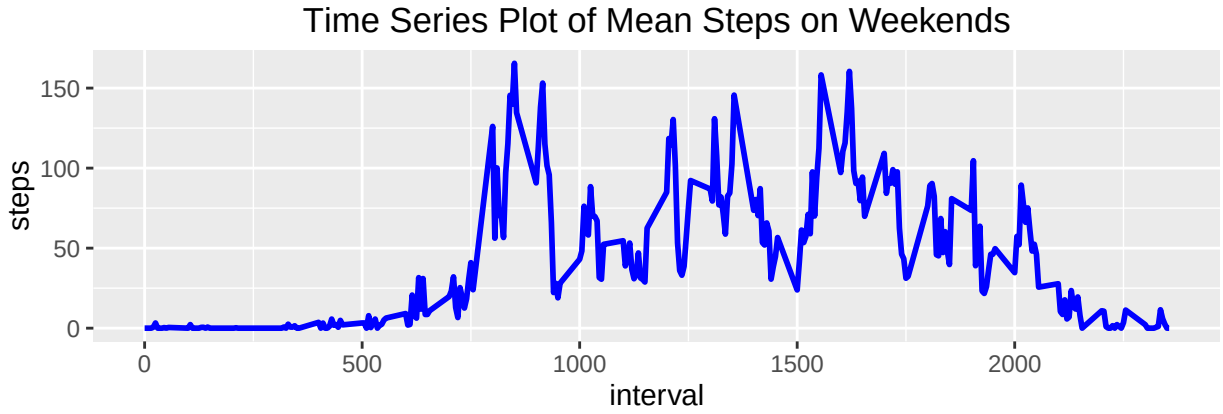
```
stepsbyday_imp <- tapply(activity_imp$steps, activity_imp$date, sum, na.rm = TRUE)
mean_steps_imp <- mean(stepsbyday_imp)
#mean steps = 10659.21
med_steps_imp <- median(stepsbyday_imp)
#median_steps = 10600
qplot(stepsbyday_imp, geom =  "histogram", binwidth=2500, main = "Histogram of Total Steps per Day with
```



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

```
activity_imp$date <- as.Date(activity_imp$date, "%Y-%m-%d")
activity_imp$day <- weekdays(activity_imp$date)
activity_imp$weekday <- ifelse(activity_imp$day %in% c("Monday", "Tuesday", "Wednesday", "Thursday", "Fr
imp_wkday <- filter(activity_imp, weekday == 1)
imp_wkend <- filter(activity_imp, weekday == 0)
imp_wkday_steps <- aggregate(steps ~ interval, imp_wkday, mean)
imp_wkend_steps <- aggregate(steps ~ interval, imp_wkend, mean)
p1 <- ggplot(data=imp_wkday_steps, aes(x=interval, y=steps), xlab = "Interval", ylab = "Mean Steps") + g
p2 <- ggplot(data=imp_wkend_steps, aes(x=interval, y=steps), xlab = "Interval", ylab = "Mean Steps") + g
grid.arrange(p1, p2)
```


Time Series Plot of Mean Steps on Weekdays
steps
200
150
100
50
0
0
500
1000
1500
2000
interval


Time Series Plot of Mean Steps on Weekends
steps
150
100
50
0
0
500
1000
1500
2000
interval