Course Project 1

Evan Valdes

6/2/2020

Loading and preprocessing the data

```
getdata <- function(){
        if(!file.exists("data")) {
                message("Creating Data Folder in working directory")
                dir.create("data")
        }

        if(!file.exists("data/repdata-data-activity")) {
                fileURL <- "http://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip"
                download.file(fileURL, destfile = "./data/repdata-data-activity.zip")
                filename <- "./data/repdata-data-activity.zip"
                unzip(filename, exdir = "data")
                unlink(filename)
        }
        else message("data already exists")
}
getdata()
activity <- read.csv("./data/activity.csv", colClasses = c("numeric", "Date", "numeric"))
activity$day <- weekdays(activity$date)
```

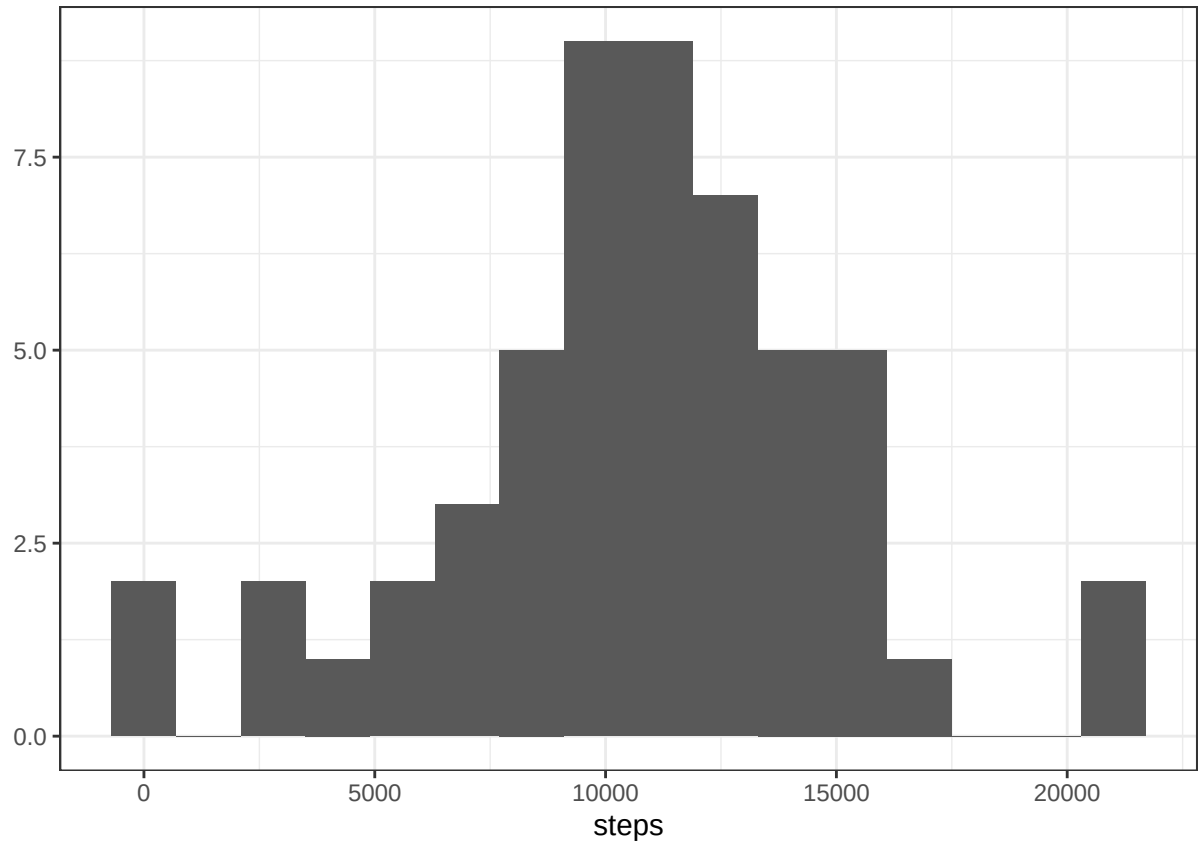
Steps Taken Per Day

Histogram of the total number of steps per day

```
library(ggplot2)

actaggregate <- aggregate(steps ~ date, activity, sum, na.rm = TRUE)

qplot(steps, data = actaggregate, binwidth = 1400) + theme_bw()
```



Median steps per day

```
median(actaggregate$steps)
```

```
## [1] 10765
```

Mean steps per day

```
mean(actaggregate$steps)
```

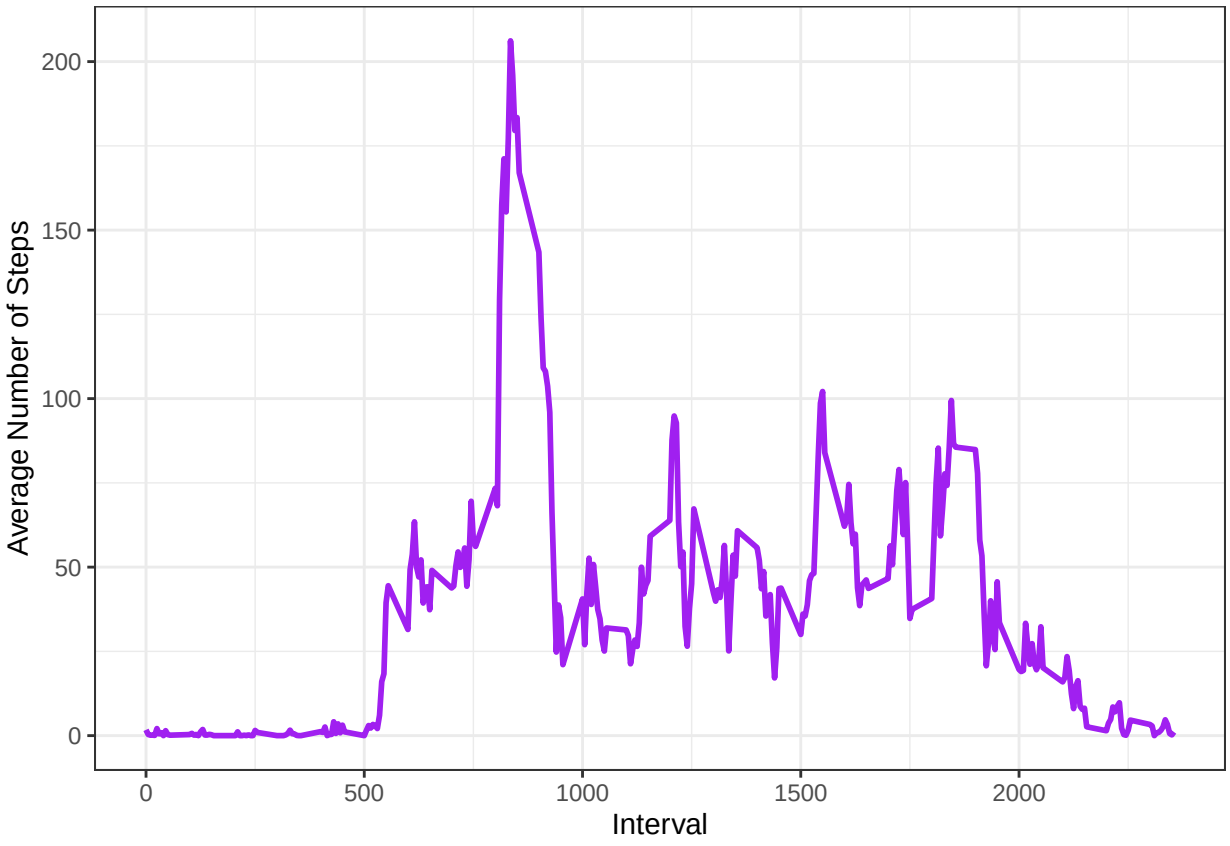
```
## [1] 10766.19
```

Average Daily Activity Pattern

Time series plot of the 5-minute interval and the avg number of steps taken, averaged across all days

```
avginterval <- aggregate(steps ~ interval, activity, mean, na.rm = TRUE)

ggplot(avginterval, aes(x = interval, y = steps)) + geom_line(color = "purple", size = 1) + labs(x = "I
```



5-minute interval containing the maximum number of steps (averaged)

```
avginterval[which.max(avginterval$steps),]$interval
```

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

Imputing Missing Values

Number of NA step values in dataset

```
sum(is.na(activity))
```

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

Method for imputing missing step values

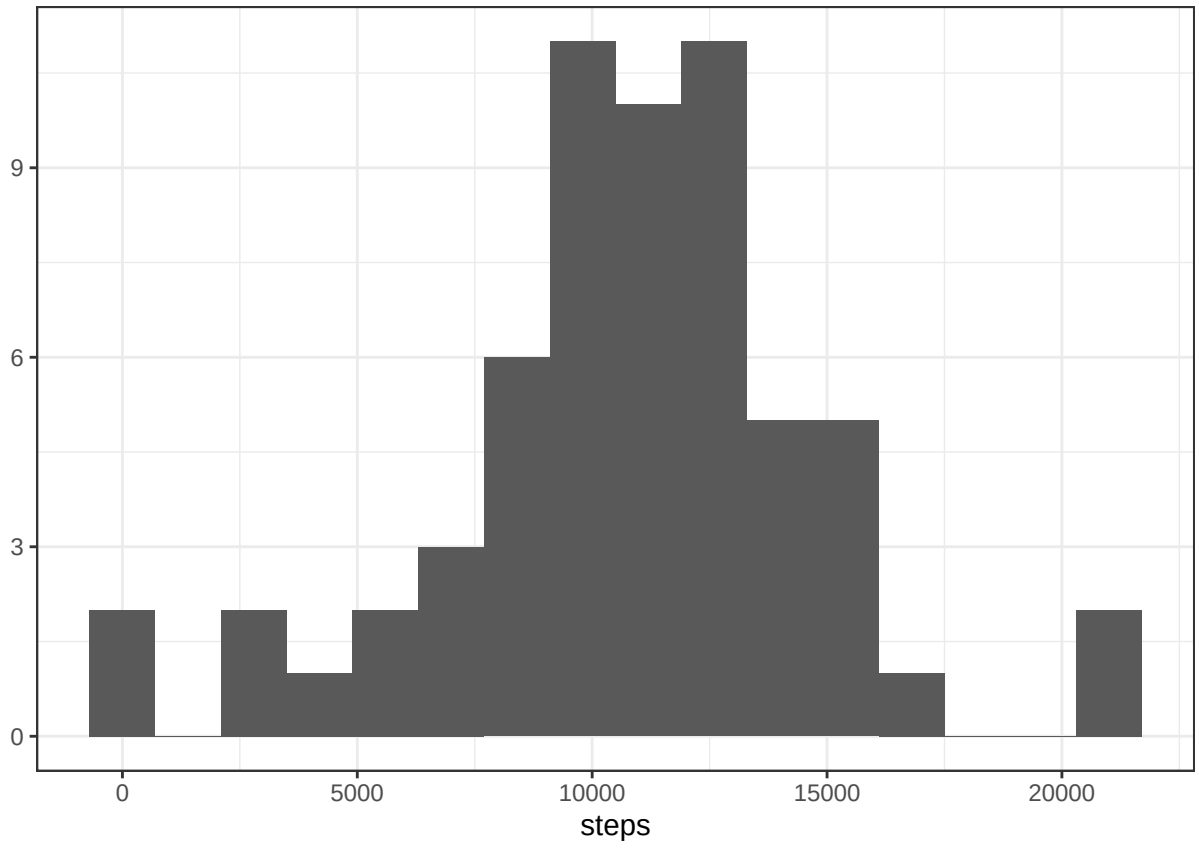
Missing step values (NA) were replaced by the mean number of steps taken for the correspondig time interval and weekday. For example, if the mean number steps taken in interval 5 on Mondays was 20 across the entire dataset, all NA entries for interval 5 on Mondays would be replaced with 20.

A new dataset is created using the imputed values

```
intdayAve <- aggregate(steps ~ interval + day, activity, mean, na.rm = TRUE)
activityImpute <- merge(activity, intdayAve, by = c("interval", "day"))
activityImpute <- transform(activityImpute, steps.x = ifelse(is.na(steps.x), steps.y, steps.x))
activityImpute <- data.frame(activityImpute[, 1:4])
names(activityImpute) <- c("interval", "day", "steps", "date")
activityImpute$steps <- round(activityImpute$steps, digits = 0)
activityImpute <- activityImpute[order(activityImpute$date, activityImpute$interval),]
```

Histogram of the total steps taken per day with imputed values

```
activityImputeAgg <- aggregate(steps ~ date, activityImpute, sum, na.rm = TRUE)
qplot(steps, data = activityImputeAgg, binwidth = 1400) + theme_bw()
```



Median steps per day with imputed values

```
median(activityImputeAgg$steps)
```

```
## [1] 11015
```

Mean steps per day with imputed values

```
mean(activityImputeAgg$steps)
```

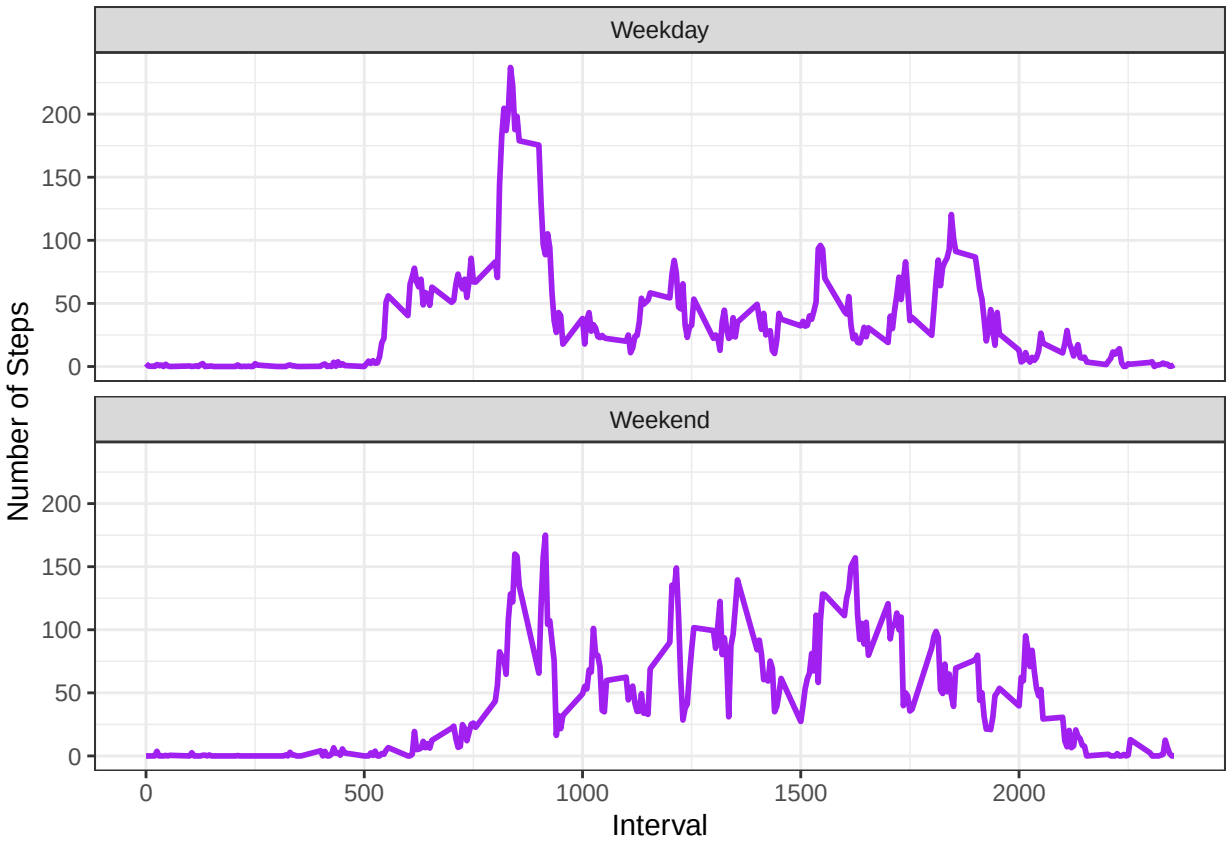
```
## [1] 10821.1
```

By including the imputed values in the dataset, both the median and the mean total number of steps taken per day increase, as expected. A comparison of histograms for the non-imputed and imputed datasets demonstrates that the imputation had the greatest impact on the 10,000 - 15,000 steps per day range and that the distribution of the data with the imputed data appears to be more normal

Investigating Differences in Activity Patterns

Between Weekdays and Weekends

```
activityImpute$daytype <- ifelse(activityImpute$day %in% c("Saturday", "Sunday"), "Weekend", "Weekday")
activityImputedDayAgg <- aggregate(steps ~ interval + daytype, activityImpute, mean)
ggplot(activityImputedDayAgg, aes(x = interval, y = steps)) + geom_line(color = "purple", size = 1) + fa
```


Weekday
Weekend
Number of Steps
0
50
100
150
200
0
500
1000
1500
2000
Interval