ReproResearch_Assignment1

C Staples

3 January 2018

Assignment 1 Introduction

It is now possible to collect a large amount of data about personal movement using activity monitoring devices such as a Fitbit, Nike Fuelband, or Jawbone Up. These type of devices are part of the “quantified self” movement - a group of enthusiasts who take measurements about themselves regularly to improve their health, to find patterns in their behavior, or because they are tech geeks. But these data remain under-utilized both because the raw data are hard to obtain and there is a lack of statistical methods and software for processing and interpreting the data.

This assignment makes use of data from a personal activity monitoring device. This device collects data at 5 minute intervals through out the day. The data consists of two months of data from an anonymous individual collected during the months of October and November, 2012 and include the number of steps taken in 5 minute intervals each day.

Data

steps: Number of steps taking in a 5-minute interval (missing values are coded as NA) date: The date on which the measurement was taken in YYYY-MM-DD format interval: Identifier for the 5-minute interval in which measurement was taken The dataset is stored in a comma-separated-value (CSV) file and there are a total of 17,568 observations in this dataset.

```
#download and unzip data

if(!file.exists("activity.csv")) {
        tempfile <- tempfile()
        download.file("http://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip",de
stfile = tempfile)
        unzip(tempfile)
        unlink(tempfile)
}
#load data

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

str(activity)
```

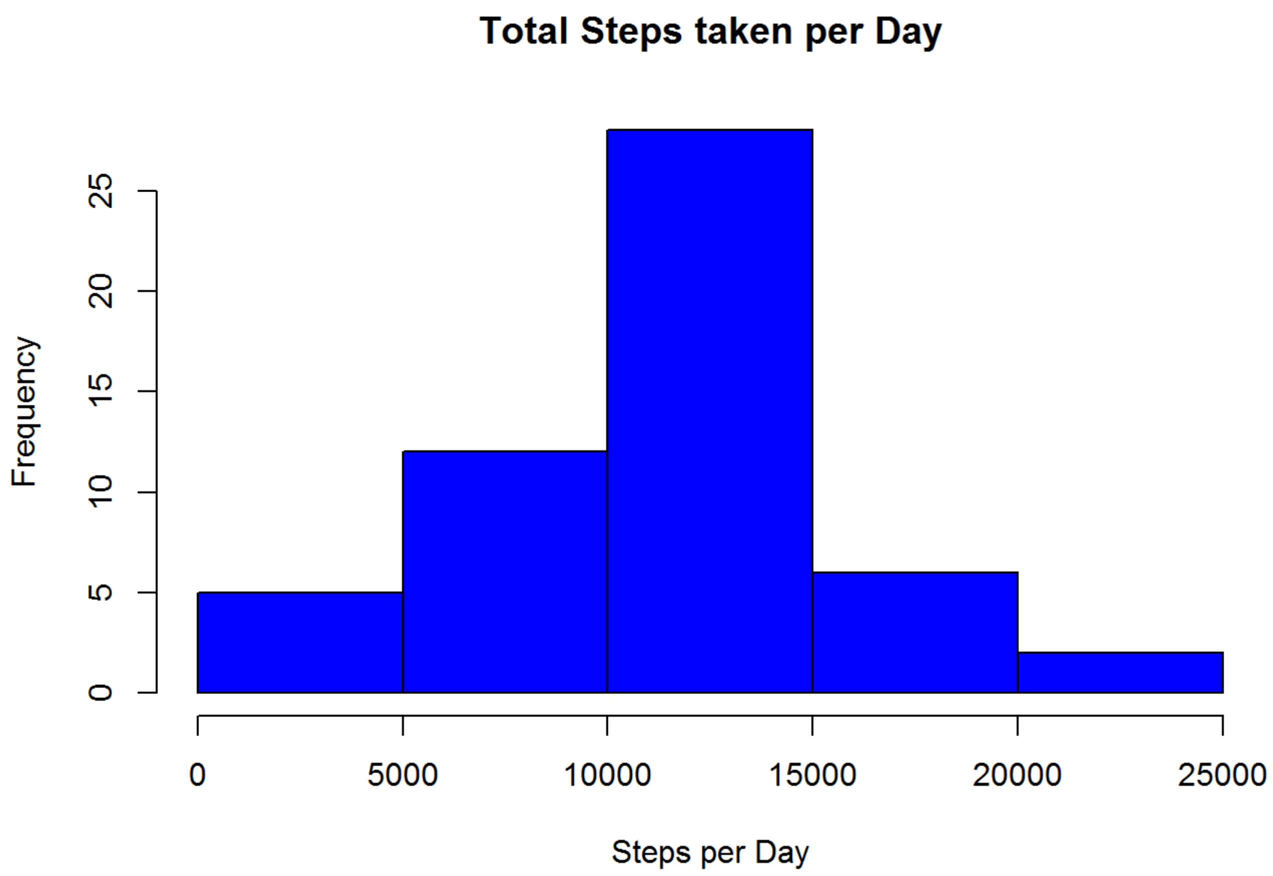
```
## 'data.frame':    17568 obs. of  3 variables:
##  $ steps   : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ date    : Factor w/ 61 levels "2012-10-01","2012-10-02",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ interval: int  0 5 10 15 20 25 30 35 40 45 ...
```

What is mean total number of steps taken per day?

In this section we aggregate the total number of steps per day, create a histogram to visualise the distribution and calculate the mean and median steps taken per day.

```
steps_day <- aggregate(steps ~ date, data = activity, FUN = sum, na.rm = TRUE)

hist(steps_day$steps, xlab = "Steps per Day", main = "Total Steps taken per Day", col = "blu
e")
```



```
mean(steps_day$steps)
```

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

```
median(steps_day$steps)
```

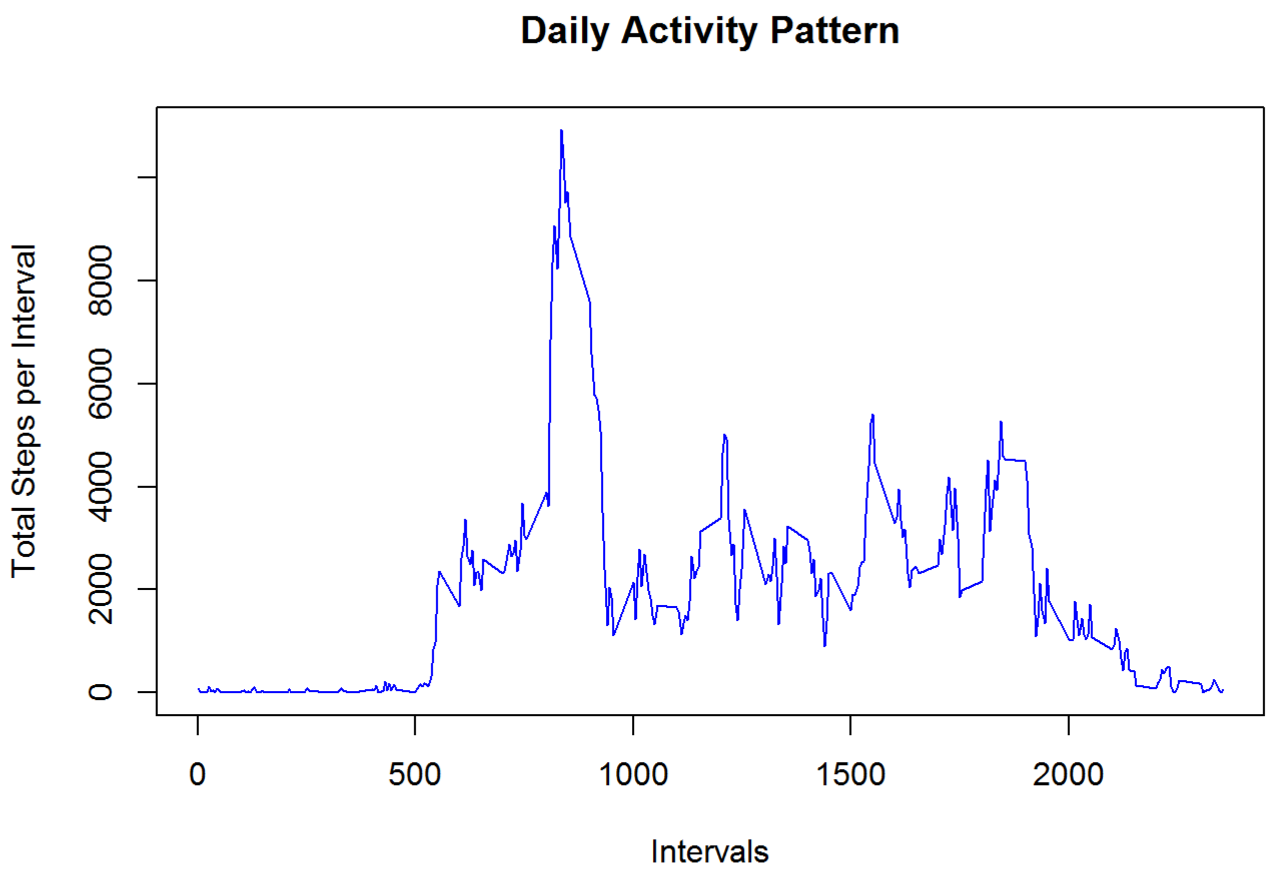
```
## [1] 10765
```

What is the average daily activity pattern?

In this section we aggregate the steps into 5 minute intervals, a time series plot will then visualise the average number of steps in the interval and finally find the interval with the max average number of steps.

```
steps_mean_int <- aggregate(steps ~ interval, data = activity, FUN = sum, na.rm = TRUE)

plot(steps_mean_int$interval, steps_mean_int$steps, type = "l", col = "blue", xlab = "Interva
ls", ylab = "Total Steps per Interval", main = "Daily Activity Pattern")
```



```
maxavsteps <- max(steps_mean_int$steps)

maxint <- steps_mean_int$interval[which(steps_mean_int$steps == maxavsteps)]
```

Imputing Missing Values

In this section we will find the number of missing values and then impute for missing vlaues to remove any potential bias that may result from missing values. To do this the average per interval will replace the missing values and this will then be visualised for the dataset with missing values and dataset with imputed values.

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

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

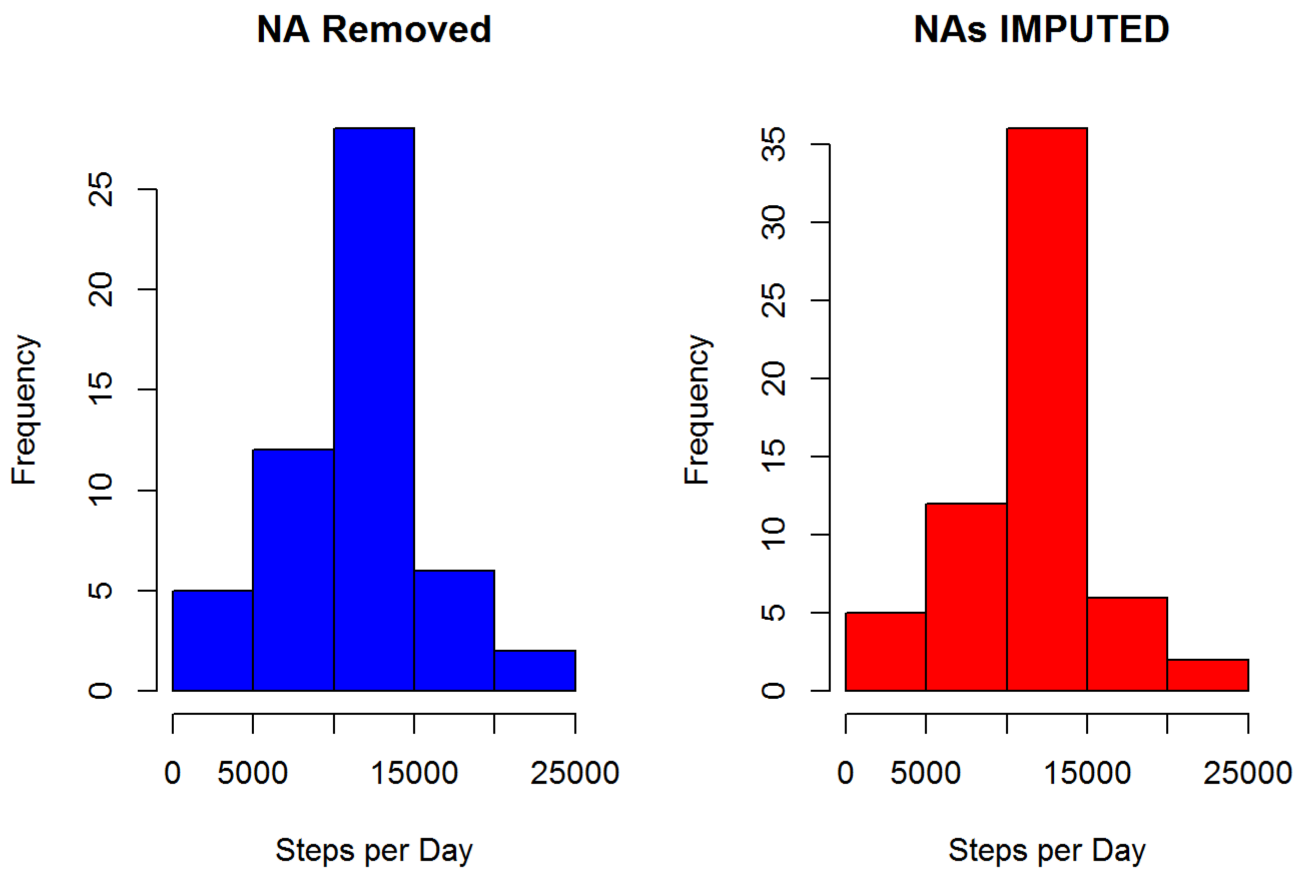
```
act_imputed <- activity
act_remove <- activity[which(!is.na(activity$steps)),]
act_imputed_mean <- tapply(act_remove$steps, act_remove$interval, mean)

act_imputed[which(is.na(act_imputed$steps)),1]  <-act_imputed_mean[as.character(act_imputed[which(is.na(act_imputed$steps)),3])]

par(mfrow = c(1,2))

steps_day <- aggregate(steps ~ date, data = activity, FUN = sum, na.rm = TRUE)
hist(steps_day$steps, xlab = "Steps per Day", main = "NA Removed", col = "blue")

steps_day_imp <- aggregate(steps ~ date, data = act_imputed, FUN = sum, na.rm = TRUE)
hist(steps_day_imp$steps, xlab = "Steps per Day", main = "NAs IMPUTED ", col = "red")
```



```
mean(steps_day_imp$steps)
```

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

```
median(steps_day_imp$steps)
```

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

```
mean(steps_day_imp$steps) - mean(steps_day$steps)
```

```
## [1] 0
```

```
median(steps_day_imp$steps) - median(steps_day$steps)
```

```
## [1] 1.188679
```

```
### Imputing has not changed the overall mean and median has only changed by approx 1 step ##
#
```

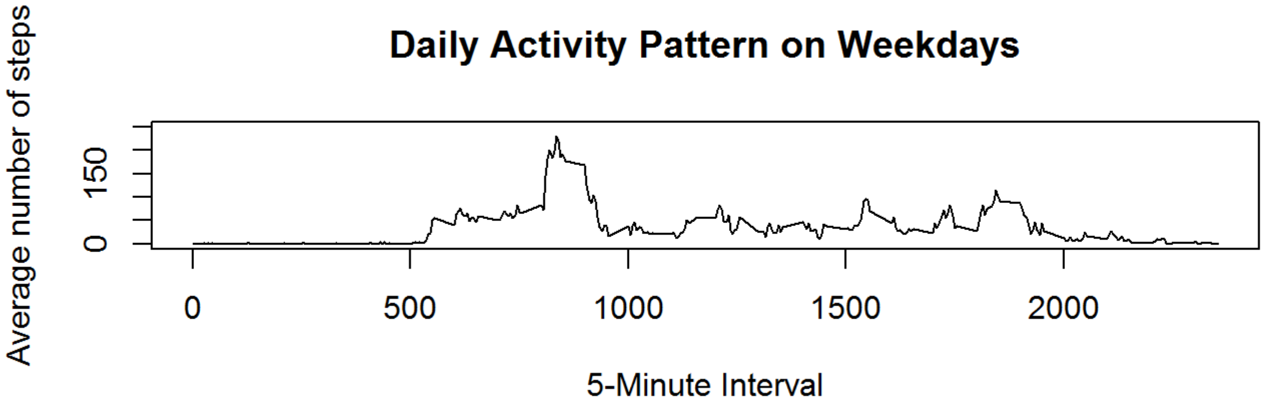
Are there differences in activity patterns between weekdays and weekends?

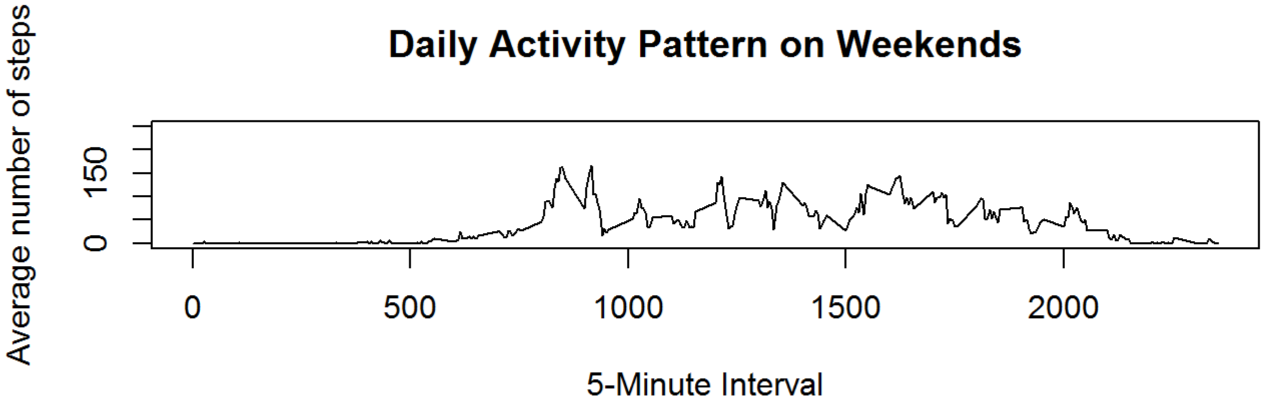
In this section two new variables will be created as factors for weekdya and weekend. Using this we can then look at differences between the days for the 5 minute intervals for average number of steps.

```
act_imputed$wd <-weekdays(as.Date(act_imputed$date))
act_imputed$fwd <- as.factor(c("weekend", "weekday"))
act_imputed[act_imputed$wd == "Sunday" | act_imputed$wd == "Saturday" ,5]<- factor("weekend")
act_imputed[!(act_imputed$wd == "Sunday" | act_imputed$wd == "Saturday"),5 ]<- factor("weekda
y")


act_imputed_we <- subset(act_imputed, fwd == "weekend")
act_imputed_wd <- subset(act_imputed, fwd == "weekday")
dailyact_we<-tapply(act_imputed_we$steps, act_imputed_we$interval, mean)
dailyact_wd<-tapply(act_imputed_wd$steps, act_imputed_wd$interval, mean)

par(mfrow=c(2,1))
plot(y = dailyact_wd, x = names(dailyact_wd), type = "l", xlab = "5-Minute Interval",
     main = "Daily Activity Pattern on Weekdays", ylab = "Average number of steps",
     ylim =c(0, 250))
plot(y = dailyact_we, x = names(dailyact_we), type = "l", xlab = "5-Minute Interval",
     main = "Daily Activity Pattern on Weekends", ylab = "Average number of steps",
     ylim =c(0, 250))
```





The two charts show different patterns to steps in the 5 minute intervals. Weekdays have much greater activity around 8-9am then much lower activity throughout the middle of the day (10am-5pm) compared to Weekdends.