Reproducible_Research_Project_1

K. Maxwell

28/10/2020

This document contains the first course project for Reproducible Research.

This assignment makes use of data from a personal activity monitoring device. The device collects data at 5-minute intervals throughout the data. 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.

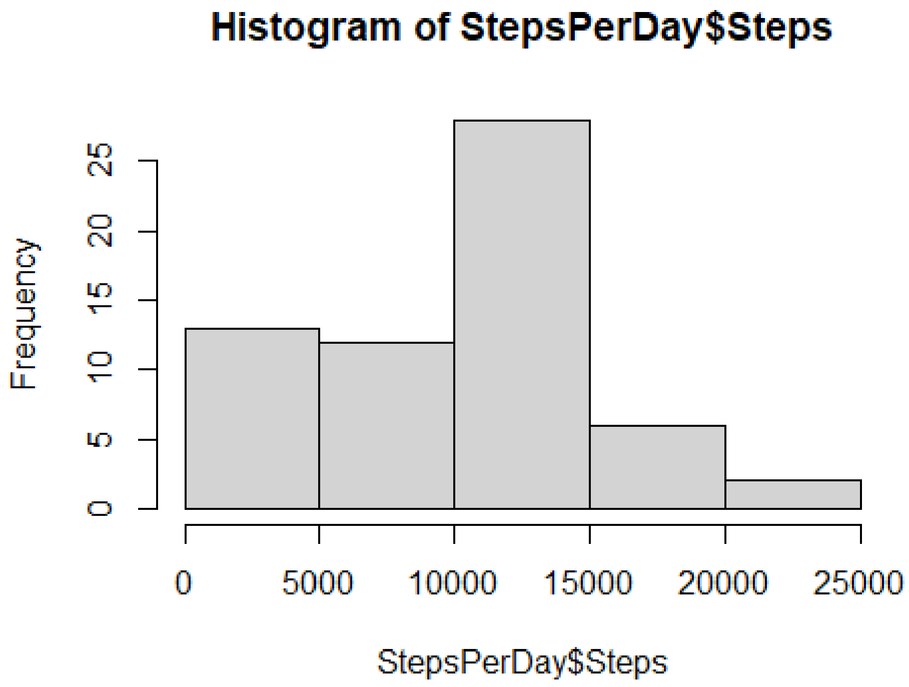
The purpose of the assignment is to load and preprocess the data, Determine the mean total number of steps taken per day, determine the average daily activity pattern, impute missing values, and finally understand if there are differences between weekday and weekend activity.

Loading the Data

```
activityData <- read.csv("U:/activity.csv")
```

The Mean and Median Total Number of Steps Taken per Day

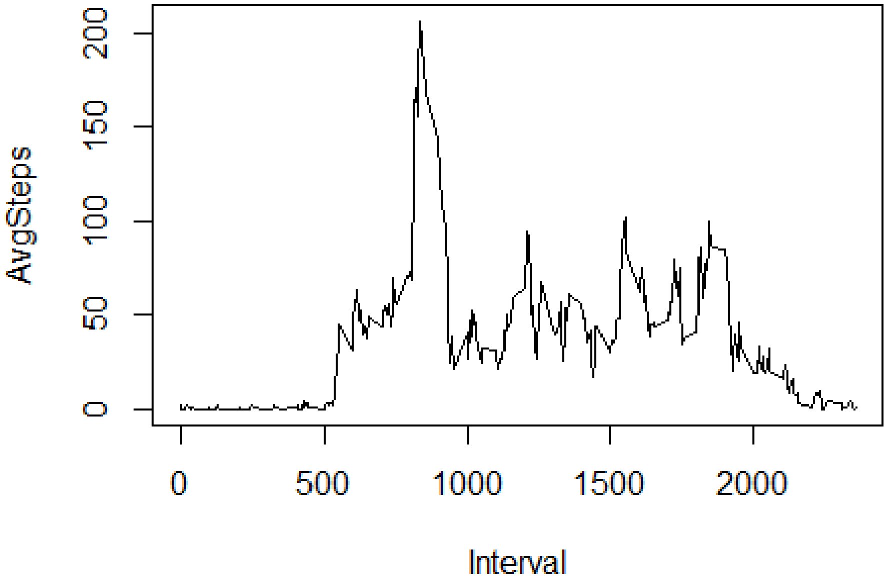
```
StepsPerDay <- aggregate(activityData$steps, list(activityData$date), sum, na
.rm=TRUE)

colnames(StepsPerDay) <- c("Date","Steps")

#Make a Histogram of the Total Number of Steps Taken per Day
hist(StepsPerDay$Steps)
```



```
#Calculate the Mean Steps per Day
MeanStepsPerDay <-mean(StepsPerDay$Steps)
MeanStepsPerDay

## [1] 9354.23

#Calculate the Median Steps per Day
MedianStepsPerDay <- median(StepsPerDay$Steps)
MedianStepsPerDay

## [1] 10395
```

The Average Daily Activity Pattern

```
#Calculate the Steps per Interval
StepsPerInterval <- aggregate(activityData$steps, list(activityData$interval)
, mean, na.rm=TRUE)
colnames(StepsPerInterval)<- c("Interval","AvgSteps")
TS_AvgStepsPerInterval <- plot(AvgSteps~Interval, data=StepsPerInterval, type
="l")
```



The 5-minute interval that contains the maximum number of steps is interval 835

```
MaxInterval2 <- StepsPerInterval [which.max(StepsPerInterval$AvgSteps),]$Inte
rval
MaxInterval2

## [1] 835
```

Impute the Missing Values

There are 2304 missing values in the dataset.

```
library(tidyverse)

## -- Attaching packages -------------------------------------------------------
---------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.2     v dplyr   1.0.0
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## -- Conflicts -------------------------------------------------------------
---------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
map(activityData, ~sum(is.na(.)))
```

```
## $steps
## [1] 2304
##
## $date
## [1] 0
##
## $interval
## [1] 0
```

Replace the NAs with Zeros and Create a New Dataset

```
activityDataNoNA <- read.csv("U:/activity.csv")
activityDataNoNA[is.na(activityData)] <- 0
```

Create a Historgram of the Total Number of Steps Taken Each Day After the Missing Values are Imputed

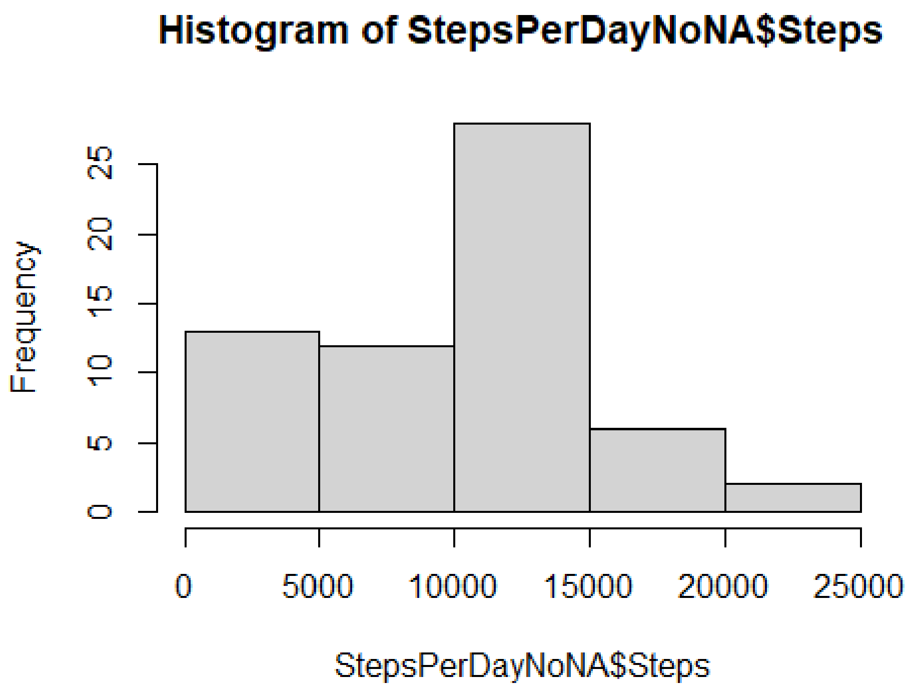
```
StepsPerDayNoNA <- aggregate(activityDataNoNA$steps, list(activityDataNoNA$da
te), sum)
colnames(StepsPerDayNoNA) <- c("Date","Steps")
StepsPerDayNoNA
```

```
##          Date Steps
## 1   10/1/2012     0
## 2  10/10/2012  9900
## 3  10/11/2012 10304
## 4  10/12/2012 17382
## 5  10/13/2012 12426
## 6  10/14/2012 15098
## 7  10/15/2012 10139
## 8  10/16/2012 15084
## 9  10/17/2012 13452
## 10 10/18/2012 10056
## 11 10/19/2012 11829
## 12  10/2/2012   126
## 13 10/20/2012 10395
## 14 10/21/2012  8821
## 15 10/22/2012 13460
## 16 10/23/2012  8918
## 17 10/24/2012  8355
## 18 10/25/2012  2492
## 19 10/26/2012  6778
## 20 10/27/2012 10119
## 21 10/28/2012 11458
## 22 10/29/2012  5018
## 23  10/3/2012 11352
## 24 10/30/2012  9819
## 25 10/31/2012 15414
## 26  10/4/2012 12116
## 27  10/5/2012 13294
## 28  10/6/2012 15420
## 29  10/7/2012 11015
## 30  10/8/2012     0
## 31  10/9/2012 12811
## 32  11/1/2012     0
## 33 11/10/2012     0
## 34 11/11/2012 12608
## 35 11/12/2012 10765
## 36 11/13/2012  7336
## 37 11/14/2012     0
## 38 11/15/2012    41
## 39 11/16/2012  5441
## 40 11/17/2012 14339
## 41 11/18/2012 15110
## 42 11/19/2012  8841
## 43  11/2/2012 10600
## 44 11/20/2012  4472
## 45 11/21/2012 12787
## 46 11/22/2012 20427
## 47 11/23/2012 21194
## 48 11/24/2012 14478
## 49 11/25/2012 11834
## 50 11/26/2012 11162
## 51 11/27/2012 13646
## 52 11/28/2012 10183
## 53 11/29/2012  7047
## 54  11/3/2012 10571
## 55 11/30/2012     0
## 56  11/4/2012     0
## 57  11/5/2012 10439
## 58  11/6/2012  8334
## 59  11/7/2012 12883
## 60  11/8/2012  3219
## 61  11/9/2012     0
```

```r
str(StepsPerDayNoNA$Steps)
```

```
##  num [1:61] 0 9900 10304 17382 12426 ...
```

```
hist(StepsPerDayNoNA$Steps)
```



Calculate the Mean and Median Steps per Day with New Dataset

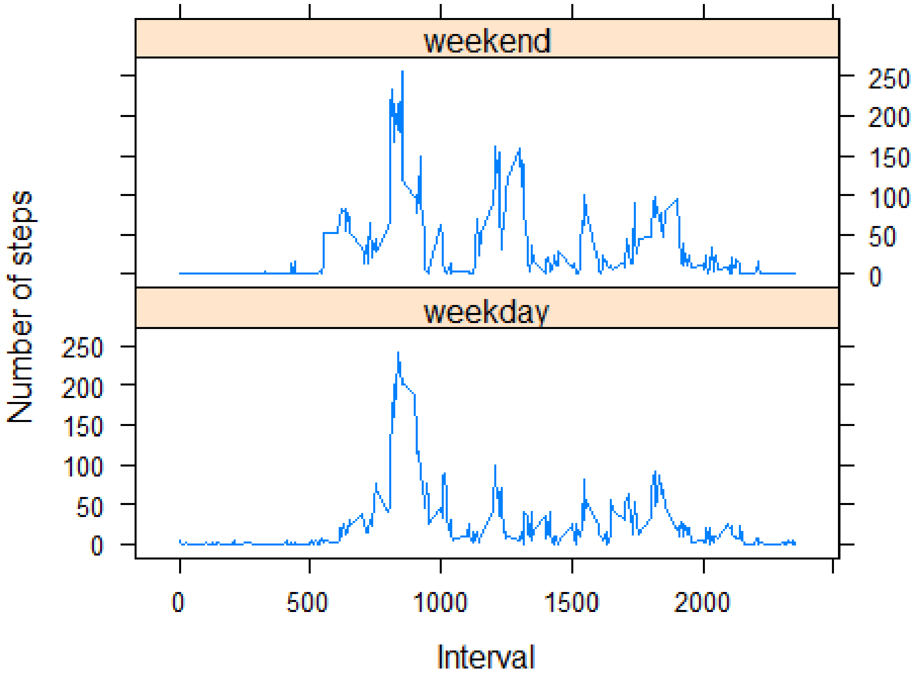
The mean and median did not change after using the new dataset.

```
MeanStepsPerDayNoNA <-mean(StepsPerDayNoNA$Steps)
MeanStepsPerDayNoNA

## [1] 9354.23

MedianStepsPerDayNoNA <- median(StepsPerDayNoNA$Steps)
MedianStepsPerDayNoNA

## [1] 10395
```

Panel Plot Comparing the Average Number of Steps Taken Each Day per 5-Minute Interval Across Weekdays and Weekends

It appears from the plot that the average steps taken increases at an earlier interval on the weekend than it does on the weekdays. It also appears that there is an increase in the average stteps taken mid day on the weekend where this same increase doesn’t appear to occur on the weekdays.

```
activityDataNoNA$date <- factor(activityDataNoNA$date)
activityDataNoNA$date <- ifelse(weekdays(as.Date(activityDataNoNA$date)) == "
Saturday" | weekdays(as.Date(activityDataNoNA$date)) == "Sunday", "weekend",
"weekday")

StepsbyDayType <- aggregate(steps~date + interval, data=activityDataNoNA, FUN
=mean)



library(lattice)
xyplot(steps ~ interval | factor(date),
       layout = c(1, 2),
       xlab="Interval",
       ylab="Number of steps",
       type="l",
       lty=1,
       data=StepsbyDayType)
```


weekend
weekday
Number of steps
250
200
150
100
50
0
250
200
150
100
50
0
0
500
1000
1500
2000
Interval