Course Project

It is now possible to collect large amount of data about personal movement using activity monitoring devices such as 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.

Loading Packages and the Data

The first step is to load all the required packages and then load the 'activity.csv' file into R Studio.

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.0.3

##
## 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(ggplot2)
library(data.table)
```

```
##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
library(timeDate)
library(mlr)
```

```
## Warning: package 'mlr' was built under R version 4.0.3

## Loading required package: ParamHelpers

## Warning: package 'ParamHelpers' was built under R version 4.0.3

## 'mlr' is in maintenance mode since July 2019. Future development
## efforts will go into its successor 'mlr3' (<https://mlr3.mlr-org.com>).
```

```
my_data <- read.csv('activity.csv')
```

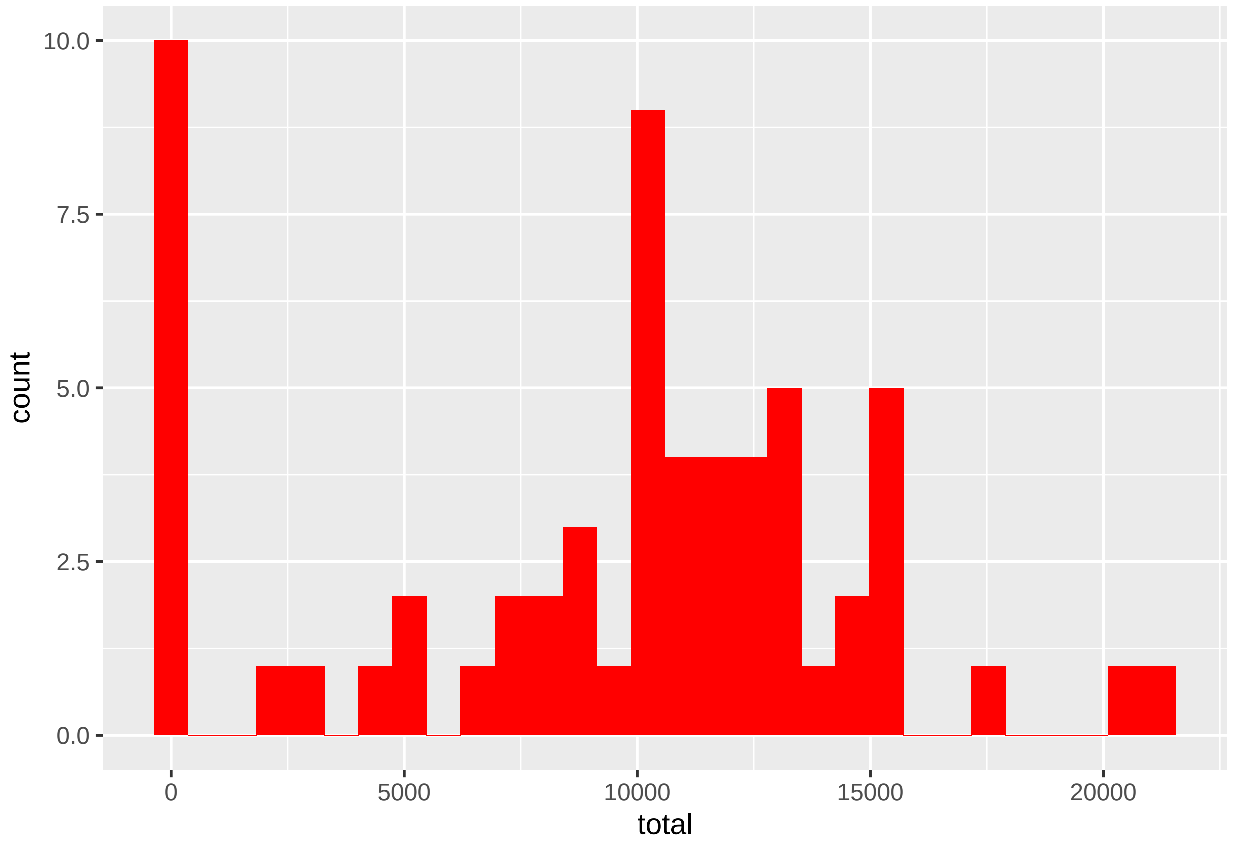
Total Steps taken each day

The first step is to convert the 'date' column from character to actual date type so that inferences could be made.

```
my_data$date <- strptime(my_data$date, '%Y-%m-%d')
my_group <- group_by(my_data, date)
my_summary <- summarise(my_group, total = sum(steps, na.rm = TRUE))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
g <- ggplot(my_summary, aes(total))
g + geom_histogram(fill = 'red', bins =30)
```



Mean and Median Number of Steps taken each day

```
mean_steps <- summarise(my_summary, myMean =mean(total, na.rm = TRUE))
median_steps <- summarise(my_summary, myMedian = median(total, na.rm = TRUE))
mean_steps$myMean
```

```
## [1] 9354.23
```

```
median_steps$myMedian
```

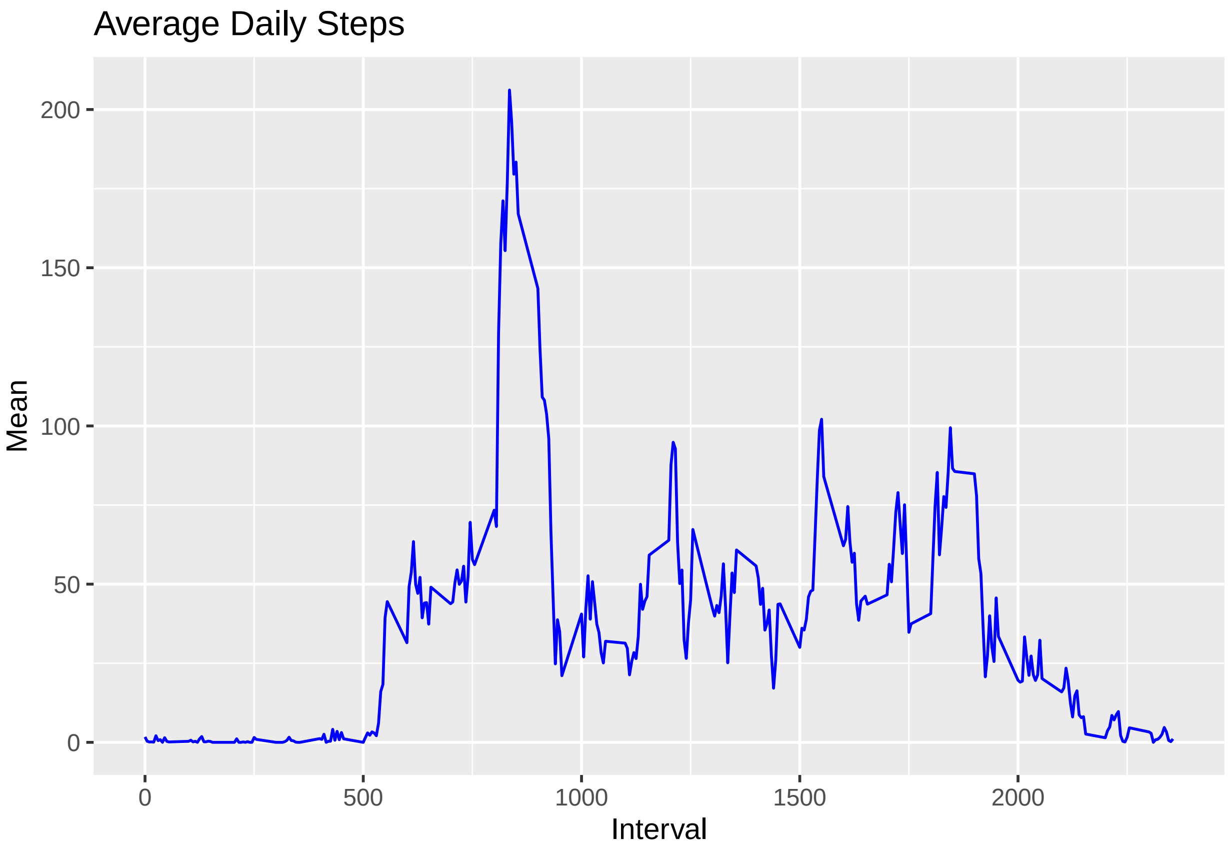
```
## [1] 10395
```

Time series plot of the average number of steps taken

```
my_group_2 <- group_by(my_data, interval)
my_summary_2 <- summarise(my_group_2, mean = mean(steps, na.rm =TRUE))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
g <- ggplot(my_summary_2, aes(interval, mean))
g + geom_line(color = 'blue') + labs(x = 'Interval', y = 'Mean',
                                     title = 'Average Daily Steps')
```



The 5 Minute Interval that contains the maximum number of steps

```
my_summary_2[which.max(my_summary_2$mean),]$interval
```

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

Imputing missing data and Histogram of steps taken each day with imputed data included

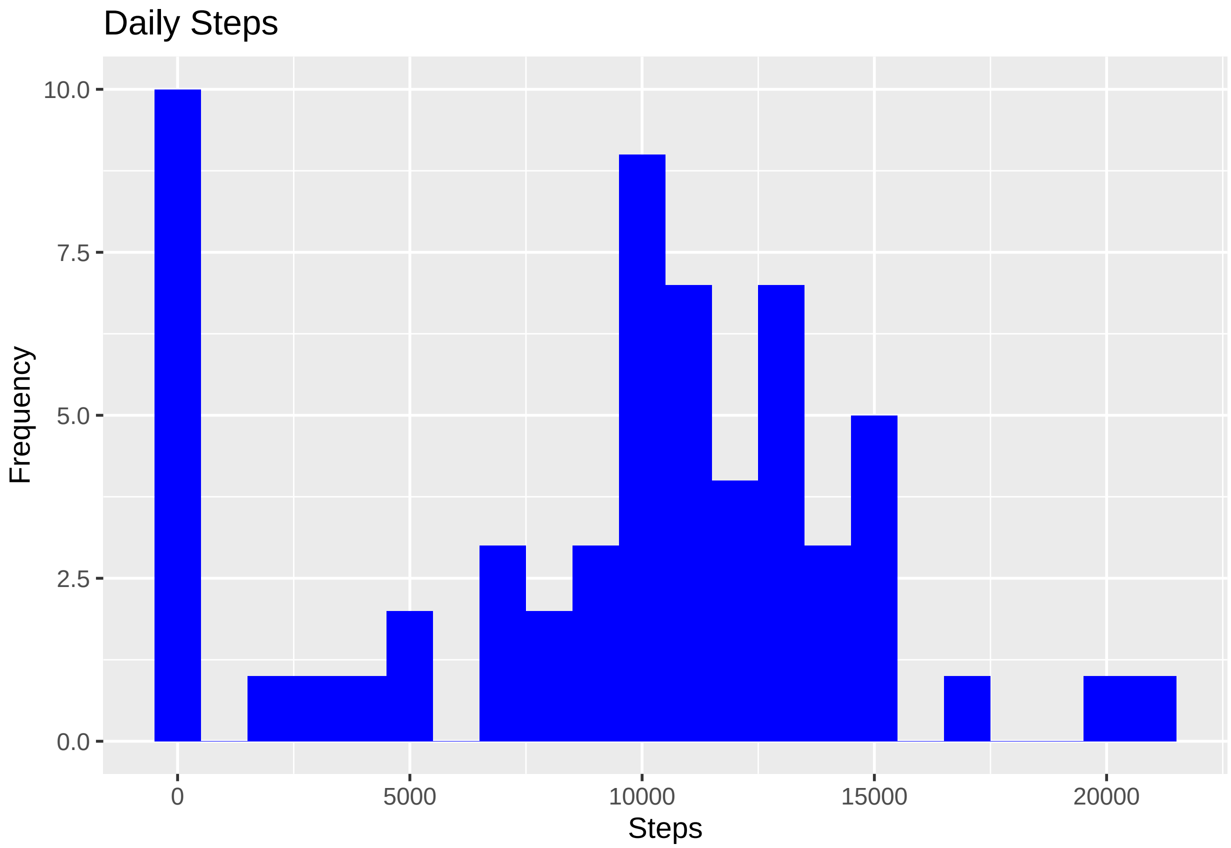
```
activityDT <- data.table::fread(input = "activity.csv")
nrow(activityDT[is.na(steps),])
```

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

```
activityDT[is.na(my_data$steps), "steps"] <- activityDT[, c(lapply(.SD, median, na.rm = TRUE)), .SDcols
data.table::fwrite(x = activityDT, file = "tidyData.csv", quote = FALSE)
Total_Steps <- activityDT[, c(lapply(.SD, sum)), .SDcols = c("steps"), by = .(date)]
Total_Steps[, .(Mean_Steps = mean(steps), Median_Steps = median(steps))]
```

```
##    Mean_Steps Median_Steps
## 1:    9354.23        10395
```

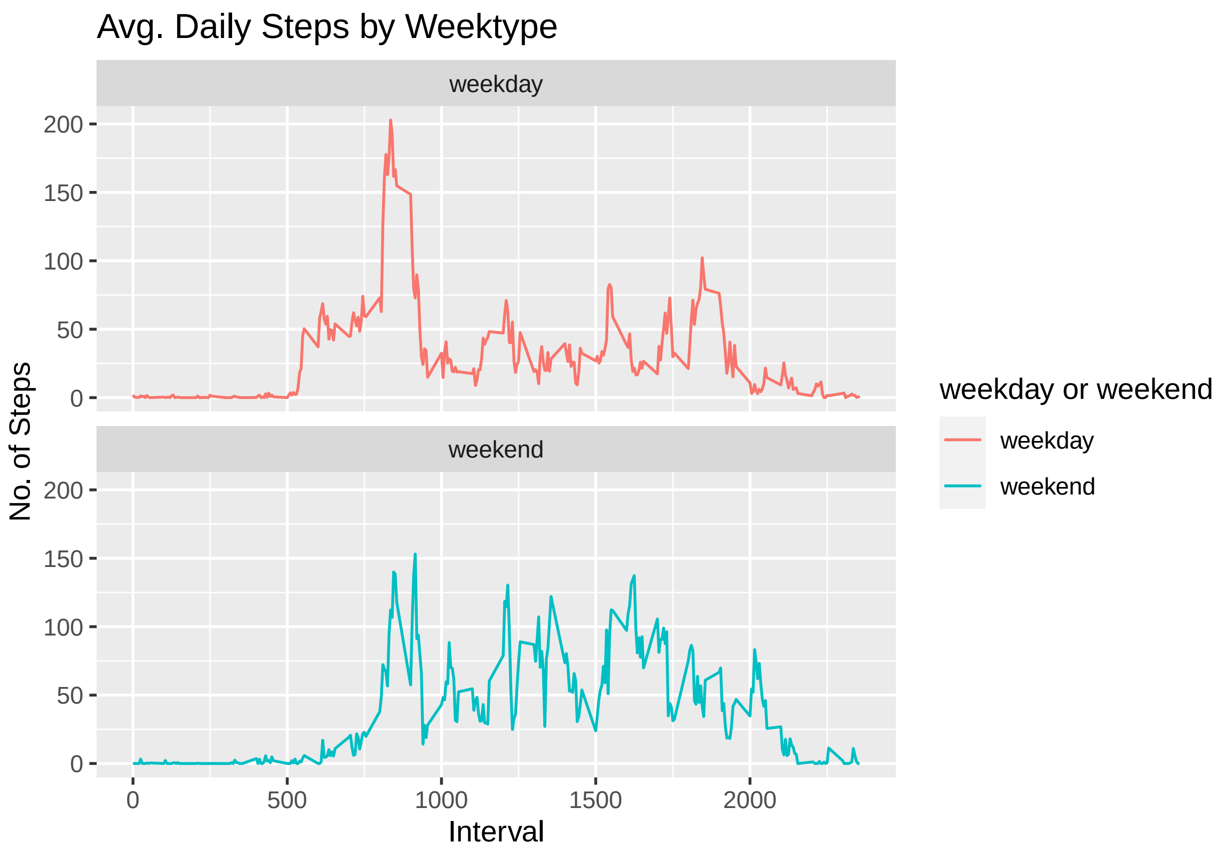
```
ggplot(Total_Steps, aes(x = steps)) + geom_histogram(fill = "blue", binwidth = 1000) + labs(title = "Da
```



Panel plot Comparing Steps/5 min interval across weekdays and weekends

```
ctivityDT <- data.table::fread(input = "activity.csv")
activityDT[, date := as.POSIXct(date, format = "%Y-%m-%d")]
activityDT[, `Day of Week`:= weekdays(x = date)]
activityDT[grepl(pattern = "Monday|Tuesday|Wednesday|Thursday|Friday", x = `Day of Week`), "weekday or w
activityDT[grepl(pattern = "Saturday|Sunday", x = `Day of Week`), "weekday or weekend"] <- "weekend"
activityDT[, `weekday or weekend` := as.factor(`weekday or weekend`)]
activityDT[is.na(steps), "steps"] <- activityDT[, c(lapply(.SD, median, na.rm = TRUE)), .SDcols = c("ste
IntervalDT <- activityDT[, c(lapply(.SD, mean, na.rm = TRUE)), .SDcols = c("steps"), by = .(interval, `w

ggplot(IntervalDT , aes(x = interval , y = steps, color=`weekday or weekend`)) + geom_line() + labs(tit
```


Avg. Daily Steps by Weektype
weekday
weekend
No. of Steps
200
150
100
50
0
0
500
1000
1500
2000
Interval
weekday or weekend
weekday
weekend