Activity Summary

Reproducible Research Module 5 - Project 1

Introduction

Nowadays, it is possible to collect data of personal movement using activity monitoring devices such as a Fitbit or Nike Fuelband. These devices are helps enthusiasts to 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.

Assignment

This assignment makes use of data from a personal activity monitoring device. This device collects data at 5 minute intervals through out the day over 2-months period (i.e. October and November 2012).

Set up global option to turn warnings off

```
knitr::opts_chunk$set(warning=FALSE)
```

Read Data from Folder

```
activitydata <- read.csv('/Users/user/Coursera/Mod 5 - Project 1/activity.csv')
```

Information about the data frame

```
head(activitydata)
```

```
##   steps       date interval
## 1    NA 2012-10-01        0
## 2    NA 2012-10-01        5
## 3    NA 2012-10-01       10
## 4    NA 2012-10-01       15
## 5    NA 2012-10-01       20
## 6    NA 2012-10-01       25
```

```
str(activitydata)
```

```
## 'data.frame':    17568 obs. of  3 variables:
##  $ steps   : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ date    : chr  "2012-10-01" "2012-10-01" "2012-10-01" "2012-10-01" ...
##  $ interval: int  0 5 10 15 20 25 30 35 40 45 ...
```

The variables in this dataset are:

1. **steps**: Number of steps taking in a 5-minute interval (missing values are coded as NA)
2. **date**: The date on which the measurement was taken in YYYY-MM-DD format
3. **interval**: Identifier for the 5-minute interval in which measurement was taken

Processing Data

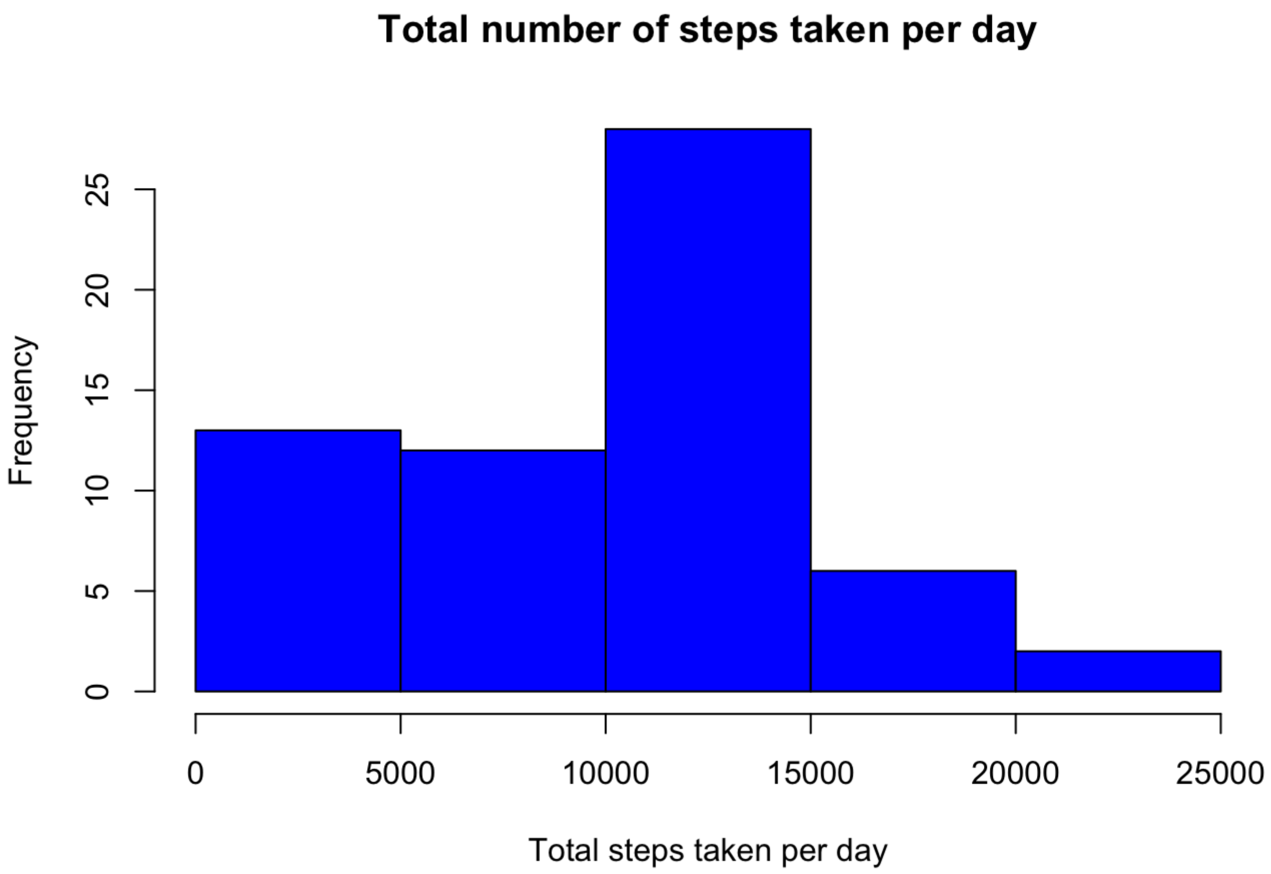
```
activitydata$date <- as.POSIXct(activitydata$date, "%Y-%m-%d")
weekday <- weekdays(activitydata$date)
activity <- cbind(activitydata,weekday)
summary(activity)
```

```
##      steps             date                interval        weekday
##  Min.   :  0.00   Min.   :2012-10-01   Min.   :   0.0   Length:17568
##  1st Qu.:  0.00   1st Qu.:2012-10-16   1st Qu.: 588.8   Class :character
##  Median :  0.00   Median :2012-10-31   Median :1177.5   Mode  :character
##  Mean   : 37.38   Mean   :2012-10-31   Mean   :1177.5
##  3rd Qu.: 12.00   3rd Qu.:2012-11-15   3rd Qu.:1766.2
##  Max.   :806.00   Max.   :2012-11-30   Max.   :2355.0
##  NA's   :2304
```

```
activity_total_steps <- with(activity, aggregate(steps, by = list(date), FUN = sum, n
a.rm = TRUE))
names(activity_total_steps) <- c("date", "steps")
```

Plot histgram of the total number of steps taken each day

```
hist(activity_total_steps$steps, main = "Total number of steps taken per day", xlab =
"Total steps taken per day", col = "blue")
```



Calculate the mean and median number of steps taken each day

```
mean(activity_total_steps$steps)
```

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

```
median(activity_total_steps$steps)
```

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

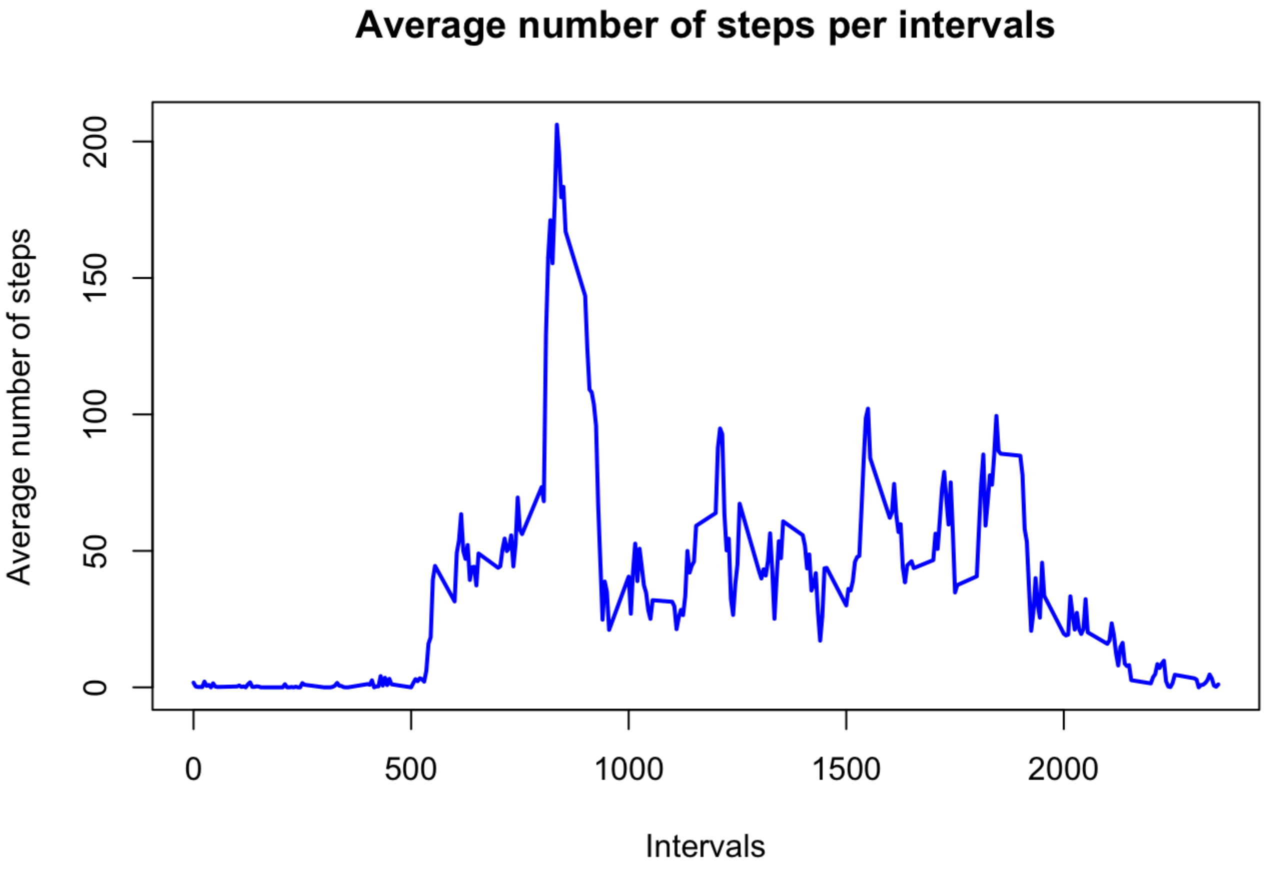
Time series plot of the average number of steps taken

```
# Calculating the average number of steps taken, averaged across the days by 5-min in
tervals.
average_daily_activity <- aggregate(activity$steps, by = list(activity$interval), FUN
= mean, na.rm = TRUE)

# Changing col names
names(average_daily_activity) <- c("Interval", "mean")

# Converting the data set into a dataframe
average_daily_activity_df <- data.frame(average_daily_activity)

# Plot the chart
da <- plot(average_daily_activity_df, col="blue", type = "l", lwd = 2, xlab="Interval
s", ylab="Average number of steps", main="Average number of steps per intervals")
```



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

```
average_daily_activity[which.max(average_daily_activity$mean),]
```

```
##     Interval     mean
## 104      835 206.1698
```

Describe and show a strategy for imputing missing data

```
# Check the number of NA entries
sum(is.na(activity$steps))
```

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

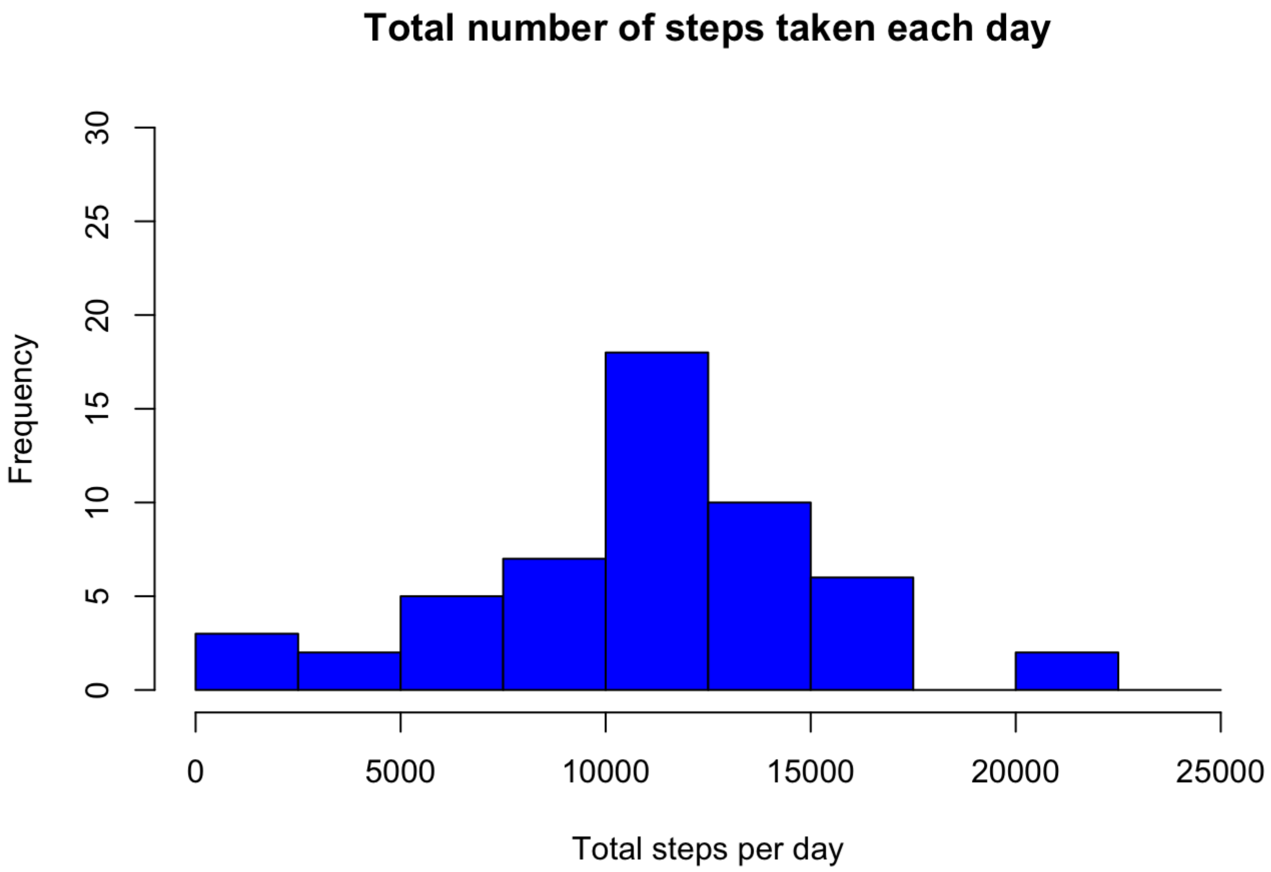
```
# Filling in the missing values
imputed_steps <- average_daily_activity$mean[match(activity$interval, average_daily_activity$interval)]
activity_imputed <- transform(activity, steps = ifelse(is.na(activity$steps), yes = imputed_steps, no = activity$steps))
total_steps_imputed <- aggregate(steps ~ date, activity_imputed, sum)
names(total_steps_imputed) <- c("date", "daily_steps")

# Check any missing NA entries with the new dataset
sum(is.na(total_steps_imputed$daily_steps))
```

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

Plot histogram of the total number of steps taken each day after missing values are imputed (new dataset)

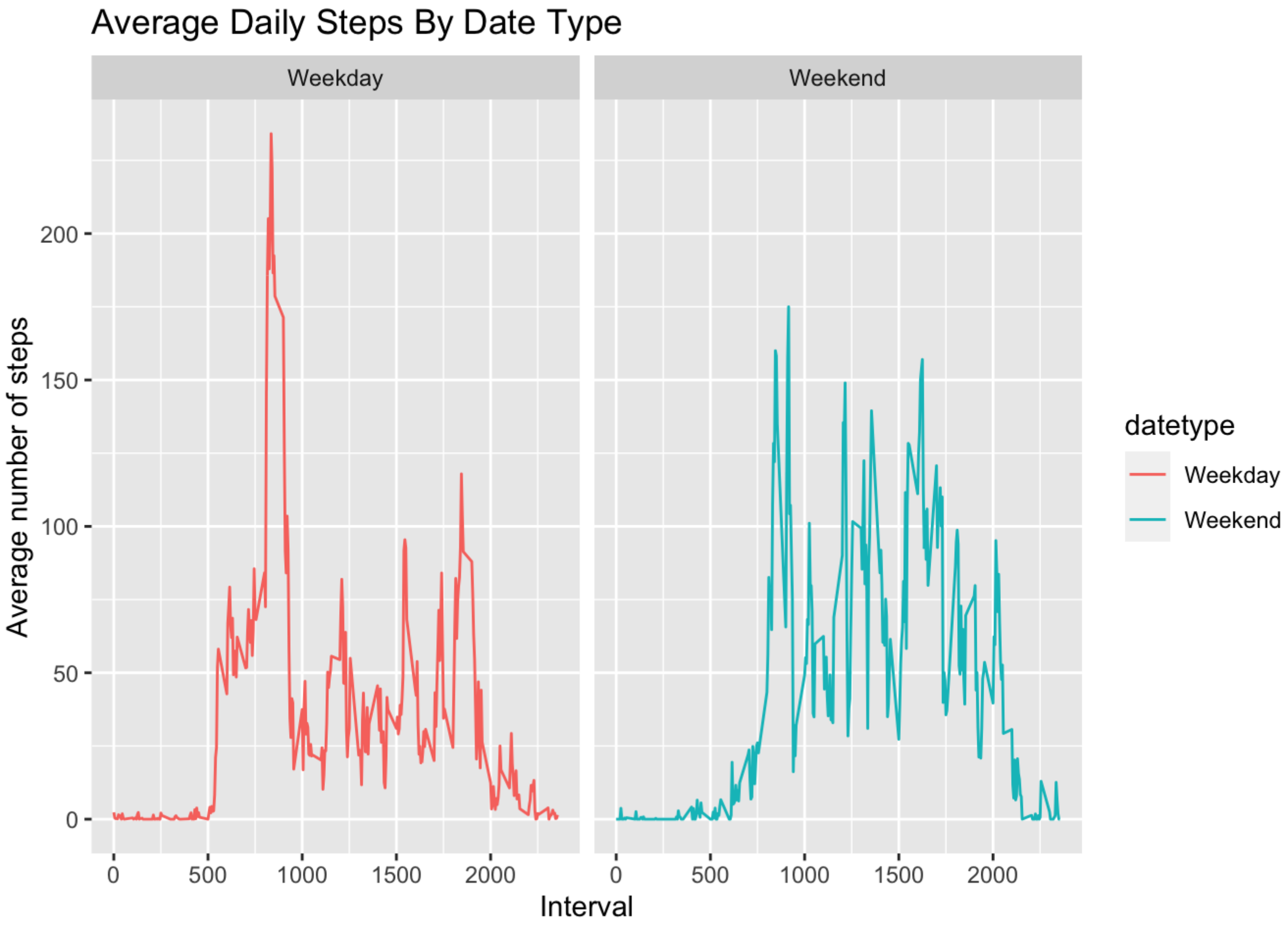
```
hist(total_steps_imputed$daily_steps, col = "blue", xlab = "Total steps per day", ylim = c(0,30), main = "Total number of steps taken each day", breaks = seq(0,25000,by=2500))
```



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

```
# Update date format and differentiate weekdays and weekends
activity$date <- as.Date(strptime(activity$date, format="%Y-%m-%d"))
activity$datetype <- sapply(activity$date, function(x) {
  if(weekdays(x) == "Saturday" | weekdays(x) == "Sunday")
  {y <- "Weekend"}
  else {y <- "Weekday"}
  y
})

# Plot the charts
activity_by_date <- aggregate(steps~interval + datetype, activity, mean, na.rm = TRUE
)
library(ggplot2)
plot <- ggplot(activity_by_date, aes(x = interval , y = steps, color = datetype)) + g
eom_line() +
labs(title = "Average Daily Steps By Date Type", x = "Interval", y = "Average number
 of steps") + facet_wrap(~datetype, ncol = 3, nrow=2)
print(plot)
```


Average Daily Steps By Date Type
Weekday
Weekend
Average number of steps
0
50
100
150
200
0
500
1000
1500
2000
Interval
datetype
Weekday
Weekend