Reproducible Research project1

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.

2. Data Processing

2.1 data

The data for this assignment can be downloaded from the given web site:

Dataset: Activity monitoring data [52K]

The variables included in this dataset are:

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. #2.2 Loading the Data

```
library(ggplot2)


activity<- read.csv("D:/profile/documents/GitHub/activity.csv")





activity$date<- as.Date.factor(activity$date,  "%m/%d/%Y" )


weekday<- weekdays(activity$date)
activity <-cbind(activity, weekday)


summary(activity)
```

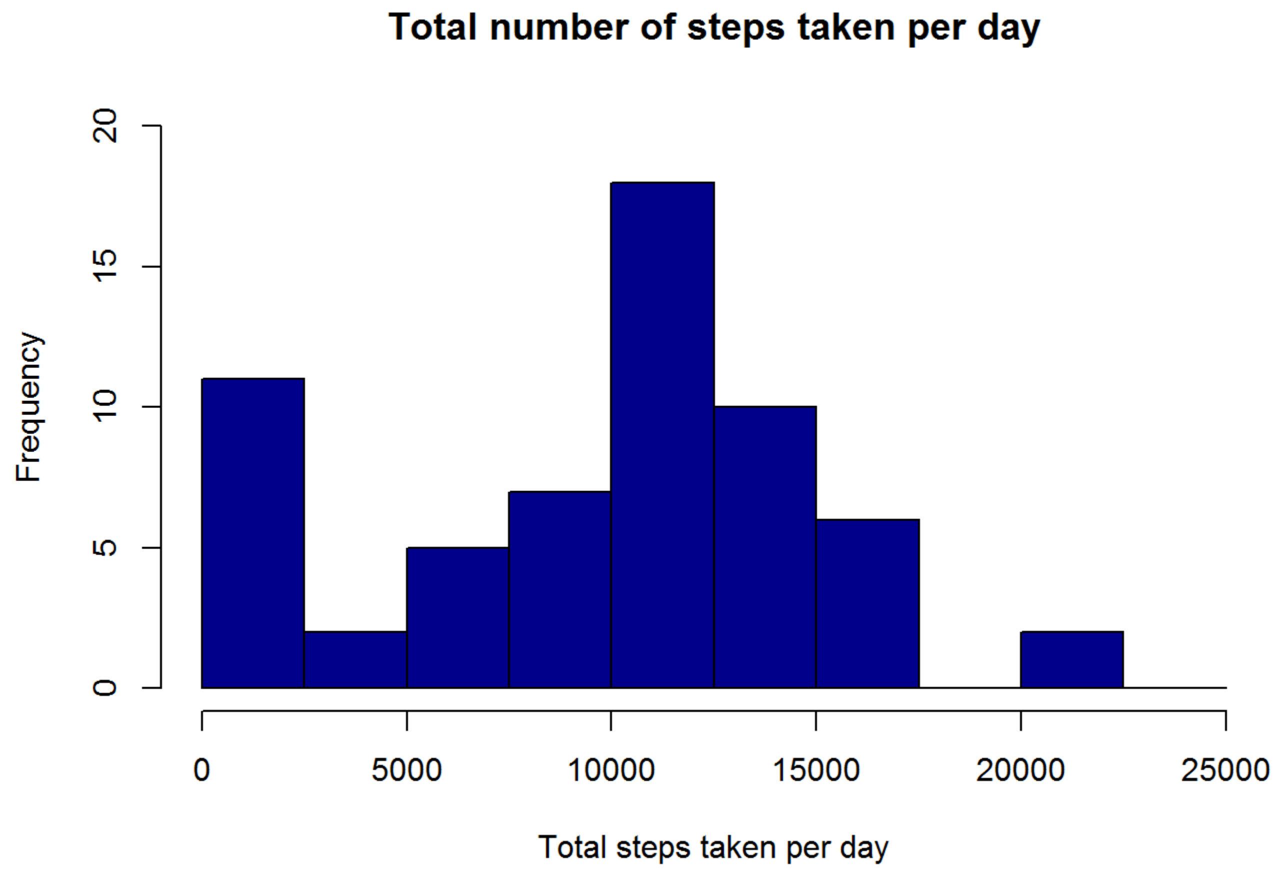
```
##      steps              date                interval            weekday
##  Min.   :  0.00   Min.   :2012-10-01   Min.   :   0.0   Friday   :2592
##  1st Qu.:  0.00   1st Qu.:2012-10-16   1st Qu.: 588.8   Monday   :2592
##  Median :  0.00   Median :2012-10-31   Median :1177.5   Saturday :2304
##  Mean   : 37.38   Mean   :2012-10-31   Mean   :1177.5   Sunday   :2304
##  3rd Qu.: 12.00   3rd Qu.:2012-11-15   3rd Qu.:1766.2   Thursday :2592
##  Max.   :806.00   Max.   :2012-11-30   Max.   :2355.0   Tuesday  :2592
##  NA's   :2304                                           Wednesday:2592
```

1. What is mean total number of steps taken per day?

```
activity.tsteps<- with(activity, aggregate(steps, by = list(date), FUN = sum, na.rm = TRUE))


names(activity.tsteps)<- c("dates", "steps")


hist(activity.tsteps$steps, main = "Total number of steps taken per day", xlab = "Total steps taken per day", col = "darkblue", ylim = c(0,20), breaks = seq(0,25000, by=2500))
```



Mean number of steps taken per day

```
mean(activity.tsteps$steps)
```

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

Median number of steps taken per day

```
median(activity.tsteps$steps)
```

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

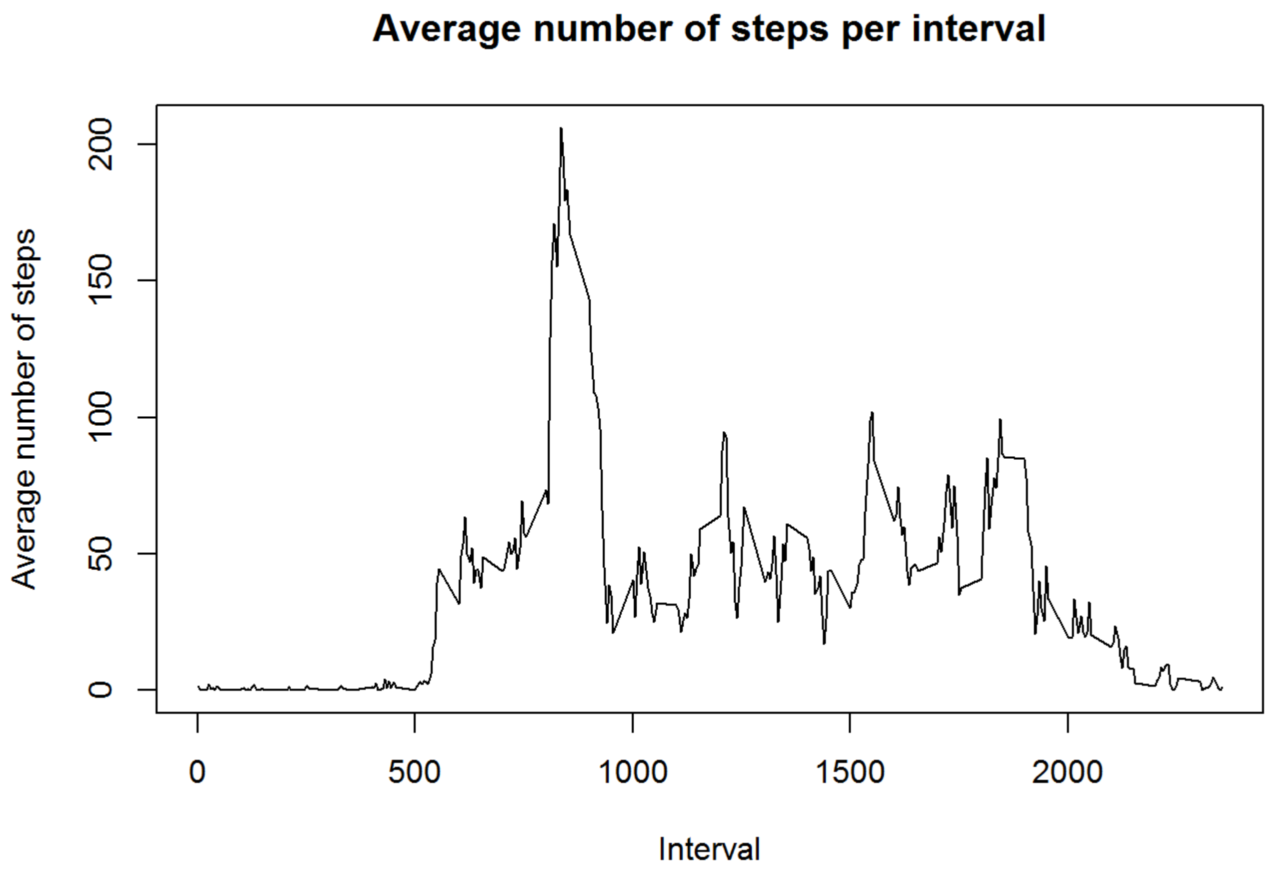
2. What is the average daily activity pattern?

Time series plot (type=“l”) of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis)

```
average.daily.activity<- aggregate(activity$steps, by= list(activity$interval), FUN = mea
n , na.rm = TRUE)


names(average.daily.activity)<-c("interval", "mean")


plot(average.daily.activity$interval, average.daily.activity$mean, type = "l", xlab = "In
terval", ylab = "Average number of steps", main = "Average number of steps per interval")
```



5-minute interval, on average across all the days in the dataset, contains the maximum number of steps

```
average.daily.activity[which.max(average.daily.activity$mean),]$interval
```

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

3. Imputing missing values

There are a number of days/intervals where there are missing values (coded as NA). The presence of missing days may introduce bias into some calculations or summaries of the data.

Calculate and report the total number of missing values in the dataset (i.e. the total number of rows with NAs)

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

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

Devise a strategy for filling in all of the missing values in the dataset. The strategy does not need to be sophisticated. For example, you could use the mean/median for that day, or the mean for that 5-minute interval, etc.

```
clean.steps<- average.daily.activity$mean[match(activity$interval,average.daily.activity$
interval)]
```

Create a new dataset that is equal to the original dataset but with the missing data filled in.

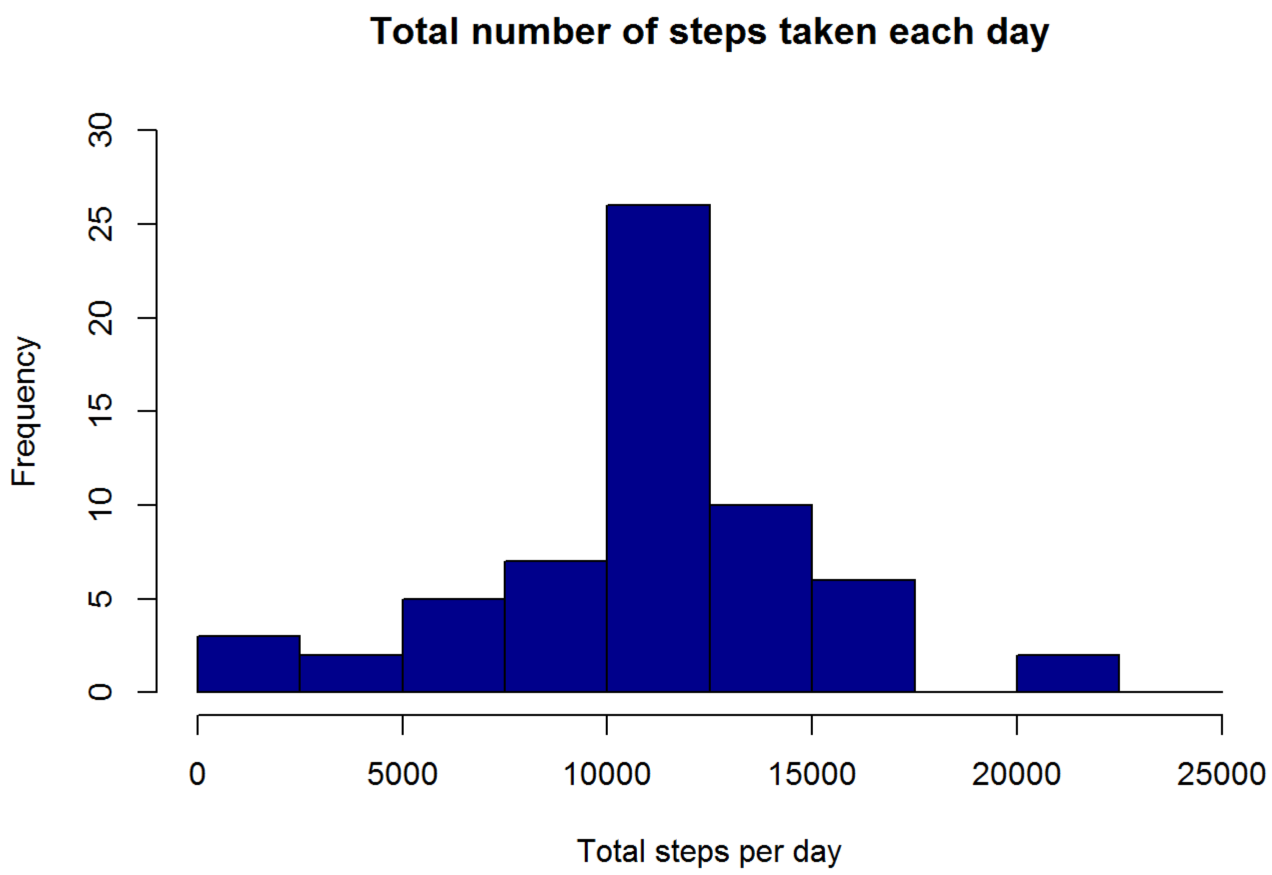
```
activity.clean <- transform(activity, steps = ifelse(is.na(activity$steps), yes = clean.s
teps, no = activity$steps))


total.clean.steps<- aggregate(steps ~ date, activity.clean, sum)


names(total.clean.steps)<- c("date", "daily.steps")
```

Make a histogram of the total number of steps taken each day and Calculate and report the mean and median total number of steps taken per day. Do these values differ from the estimates from the first part of the assignment? What is the impact of imputing missing data on the estimates of the total daily number of steps?

```
hist(total.clean.steps$daily.steps, col = "darkblue", xlab = "Total steps per day", ylim
= c(0,30), main = "Total number of steps taken each day", breaks = seq(0,25000,by=2500))
```



Mean of the total number of steps taken per day

```
mean(total.clean.steps$daily.steps)
```

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

Median of the total number of steps taken per day

```
median(total.clean.steps$daily.steps)
```

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

4. Are there differences in activity patterns between weekdays and weekends?

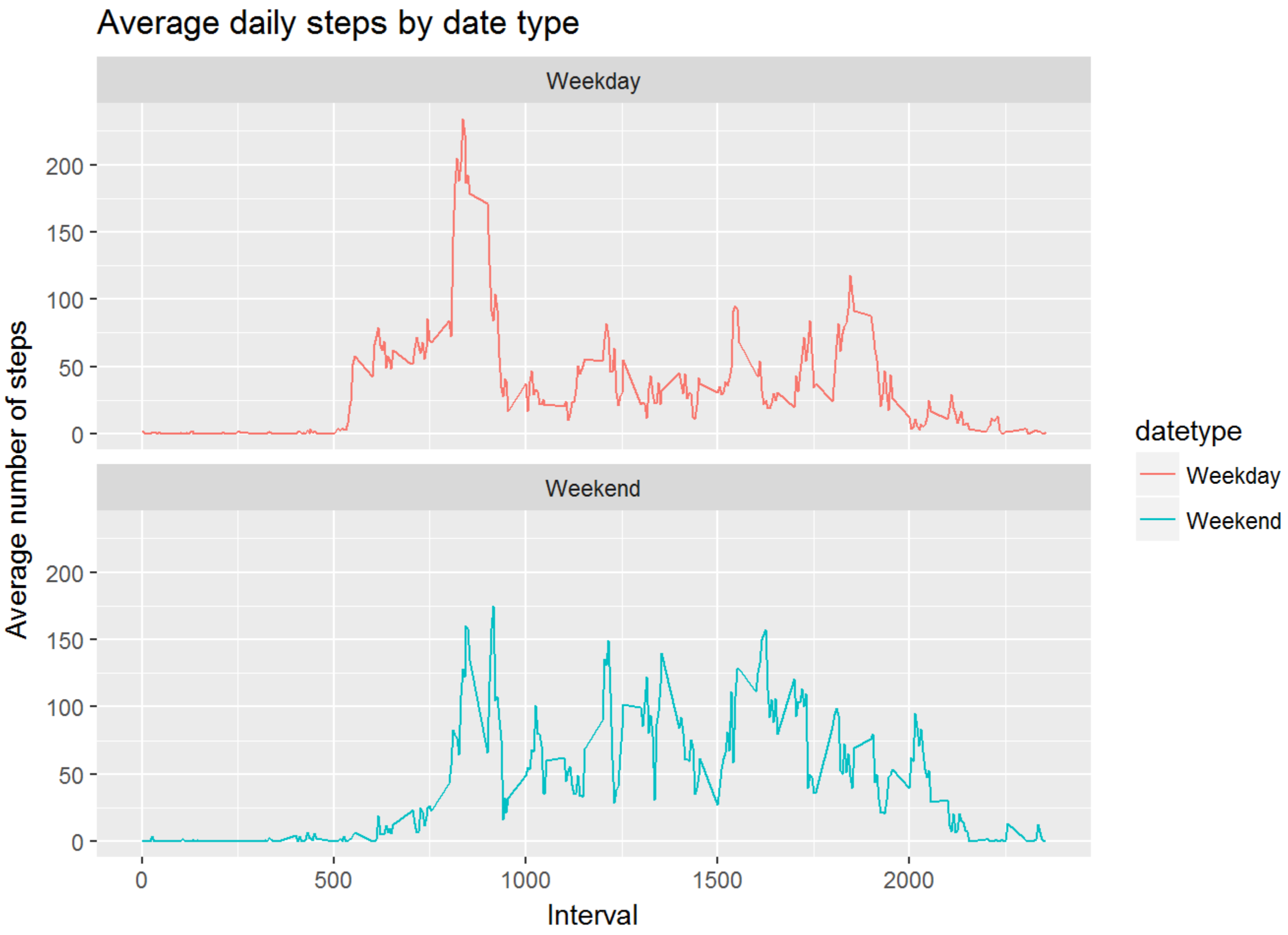
Create a new factor variable in the dataset with two levels - “weekday” and “weekend” indicating whether a given date is a weekday or weekend day.

```
activity$datetype <- sapply(activity$date, function(x) {
        if (weekdays(x) == "Saturday" | weekdays(x) =="Sunday")
                {y <- "Weekend"} else
                {y <- "Weekday"}
                y
        })
```

A panel plot containing a time series plot (type="l") of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all weekday days or weekend days (y-axis).

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

ggplot(activity.datetype, aes(x = interval, y = steps, color = datetype))+ geom_line() +
labs(title = "Average daily steps by date type", x = "Interval", y = "Average number of s
teps") + facet_wrap(~datetype, ncol = 1, nrow = 2)
```



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.