

# Reproducible Research Peer Assessment 1

## Reproducible Research Peer Assessment 1

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

## Load the libraries needed to complete assignment

### 1. Load and process the data

```
options(tinytex.verbose = TRUE)

if(!file.exists('activity.csv')){
  unzip('activity.zip')
}

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

### 2. transform the data to fit our analysis

```
options(tinytex.verbose = TRUE)

#activityData$interval <- strptime(gsub("([0-9]{1,2}) ([0-9]{2})", "\\1:\\2",
activityData$interval), format='%H:%M')
```

---

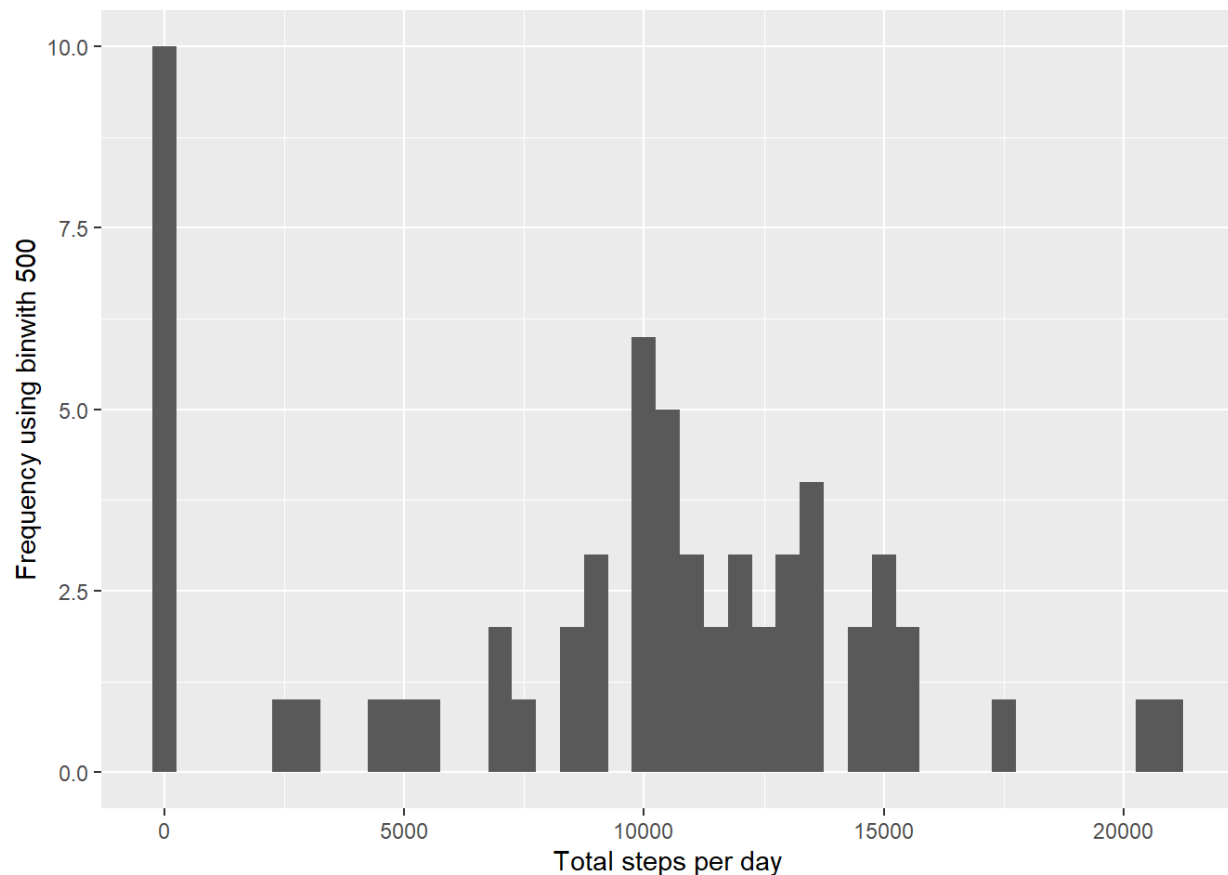
## Obtain the mean total number of steps taken per day.

```
options(tinytex.verbose = TRUE)

stepsByDay <- tapply(activityData$steps, activityData$date, sum, na.rm=TRUE)
```

### 3. Generate the Histogram of the total number of steps taken each day

```
options(tinytex.verbose = TRUE)
qqplot(stepsByDay, xlab='Total steps per day', ylab='Frequency using binwidth 500', binwidth=500)
```



### 4. Generate the Mean and Median total number of steps taken per day

```
options(tinytex.verbose = TRUE)
stepsByDayMean <- mean(stepsByDay)
stepsByDayMedian <- median(stepsByDay)
```

- Mean: 9354.2295082
- Median: 10395

---

Determine if there is an average daily activity pattern

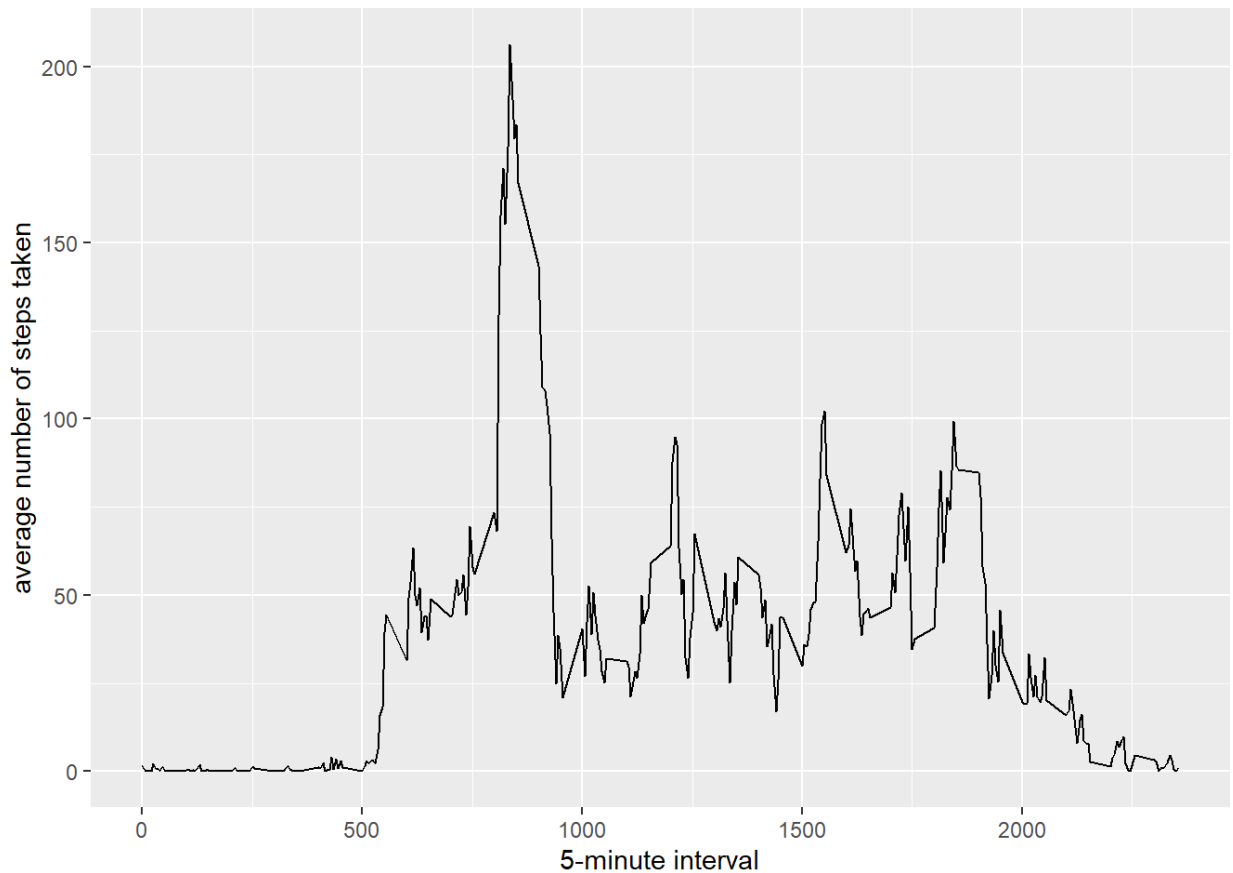
```
options(tinytex.verbose = TRUE)

averageStepsPerTimeBlock <- aggregate(x=list(meanSteps=activityData$steps), b
y=list(interval=activityData$interval), FUN=mean, na.rm=TRUE)
```

## 5. Generate the Time series plot for the data

```
options(tinytex.verbose = TRUE)

ggplot(data=averageStepsPerTimeBlock, aes(x=interval, y=meanSteps)) +
  geom_line() +
  xlab("5-minute interval") +
  ylab("average number of steps taken")
```



## 6. Calculate the 5-minute interval on average across all the days in the dataset that contains the maximum number of steps

```
options(tinytex.verbose = TRUE)

mostSteps <- which.max(averageStepsPerTimeBlock$meanSteps)
```

```
timeMostSteps <- gsub("([0-9]{1,2}) ([0-9]{2})", "\\1:\\2", averageStepsPerTimeBlock[mostSteps, 'interval'])
```

- Determined that the Most Steps occur at: 8:35

---

## Determine the Number of Missing values

7. Calculate the total/length number of missing values in the dataset

```
options(tinytex.verbose = TRUE)
numMissingValues <- length(which(is.na(activityData$steps)))
```

- Determined that Number of missing values is: 2304

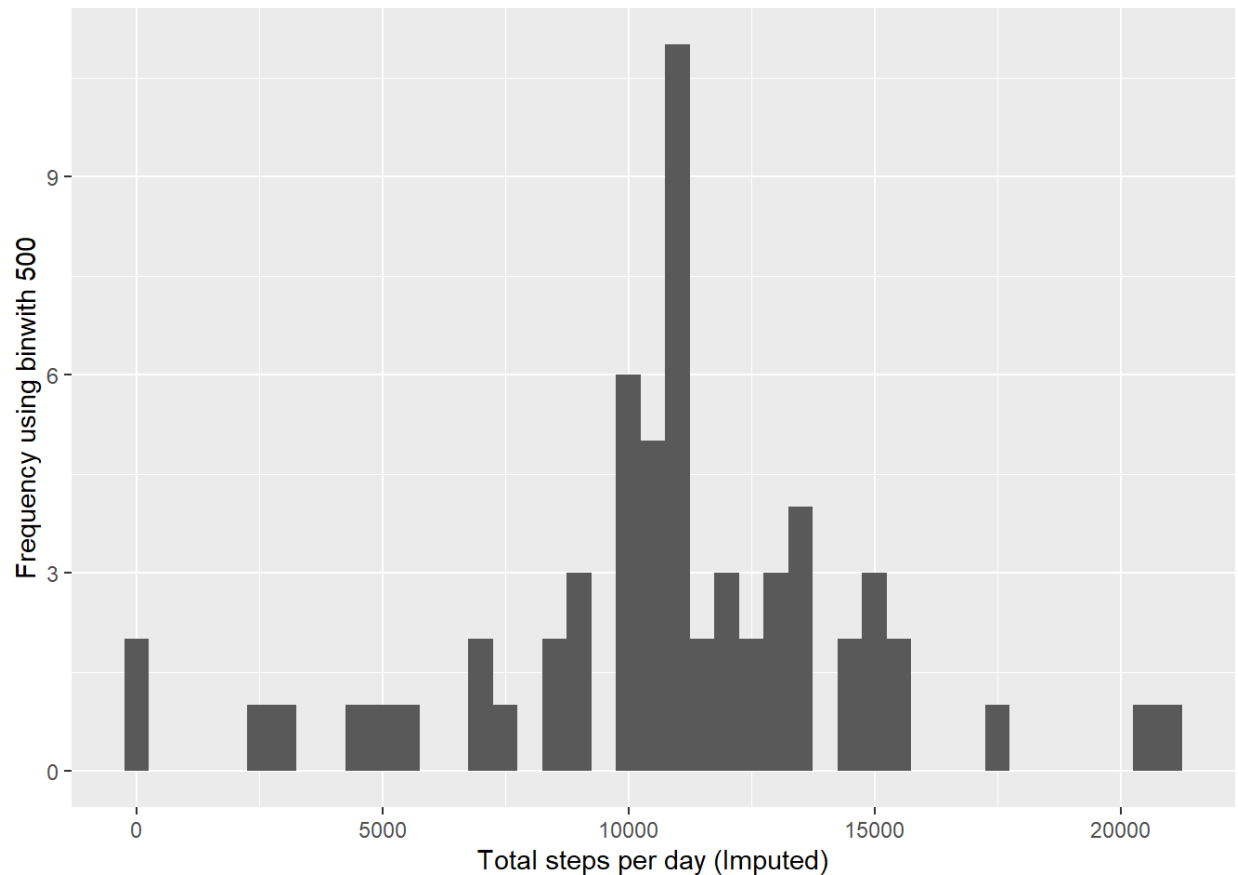
8. Devise a strategy for filling in all of the missing values in the dataset.

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

```
options(tinytex.verbose = TRUE)
activityDataImputed <- activityData
activityDataImputed$steps <- impute(activityData$steps, fun=mean)
```

10. Create a Histogram of the total number of steps taken each day

```
options(tinytex.verbose = TRUE)
stepsByDayImputed <- tapply(activityDataImputed$steps, activityDataImputed$date, sum)
qqplot(stepsByDayImputed, xlab='Total steps per day (Imputed)', ylab='Frequency using binwidth 500', binwidth=500)
```



11. Calculate and report the mean and median total number of steps taken per day.

```
options(tinytex.verbose = TRUE)
stepsByDayMeanImputed <- mean(stepsByDayImputed)
stepsByDayMedianImputed <- median(stepsByDayImputed)
```

- Mean (Imputed): 1.076618910<sup>4</sup>
- Median (Imputed): 1.076618910<sup>4</sup>

## Are there differences in activity patterns between weekdays and weekends?

12. 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.

```
options(tinytex.verbose = TRUE)
```

```
activityDataImputed$dateType <- ifelse(as.POSIXlt(activityDataImputed$date)$  
wday %in% c(0,6), 'weekend', 'weekday')
```

### 13. Generate a Panel plot containing a time series plot

```
options(tinytex.verbose = TRUE)  
  
averagedActivityDataImputed <- aggregate(steps ~ interval + dateType, data=ac  
tivityDataImputed, mean)  
  
ggplot(averagedActivityDataImputed, aes(interval, steps)) +  
  geom_line() +  
  facet_grid(dateType ~ .) +  
  xlab("5-minute interval") +  
  ylab("average number of steps")
```

