# Reproducible Research: Peer Assessment 1

## Loading and preprocessing the data

### 1. Load the data (i.e. read.csv())

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

### Process/transform the data (if necessary) into a format suitable for your analysis

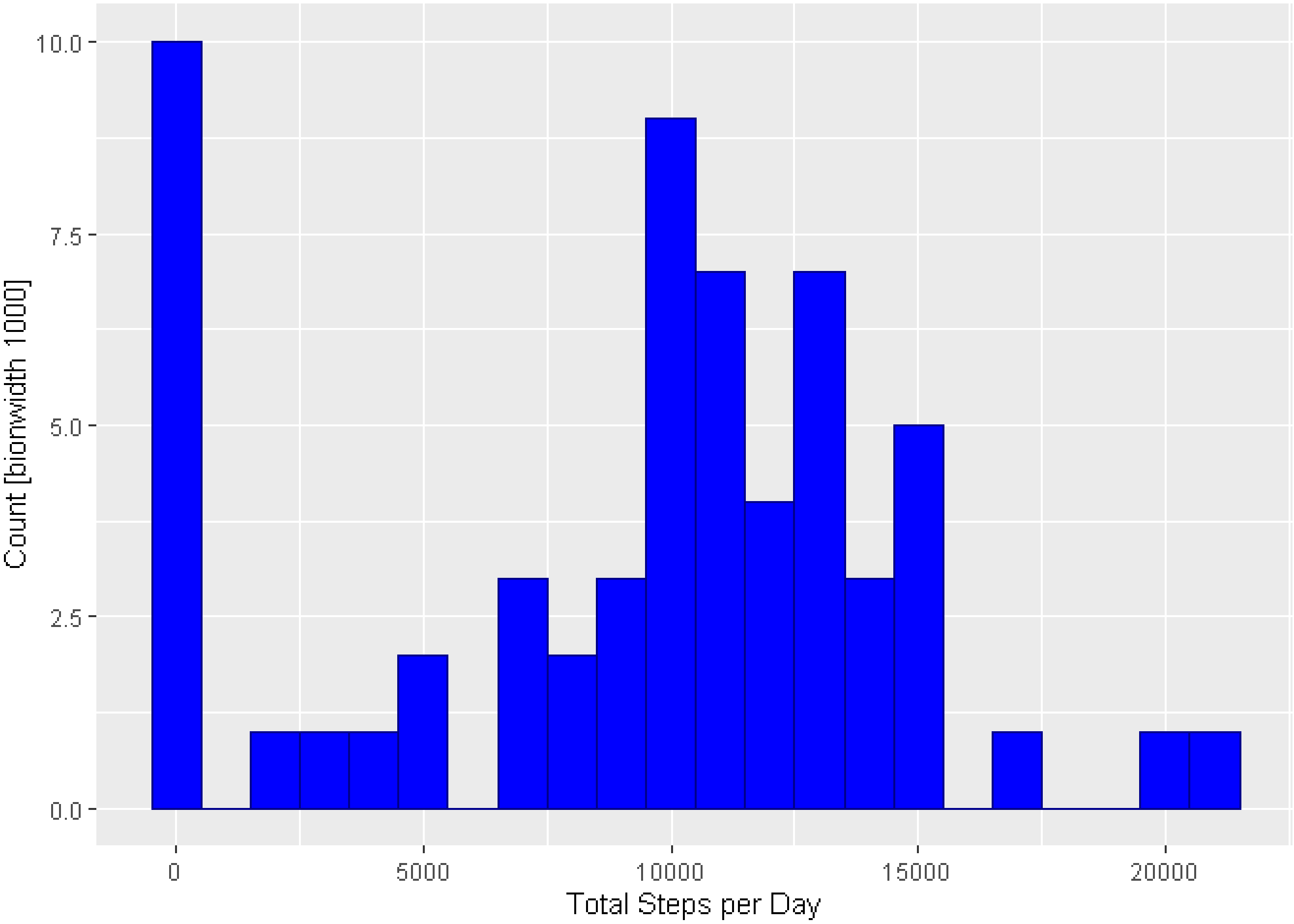
## What is mean total number of steps taken per day?

### Calculate the total number of steps taken per day

```
vecTotalStepsPerDay <- tapply(dfActivityData$steps, dfActivityData$date, sum, na.r
m=TRUE)
mean(vecTotalStepsPerDay)
```

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

### If you do not understand the difference between a histogram and a barplot, research the difference between them. Make a histogram of the total number of steps taken each day



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

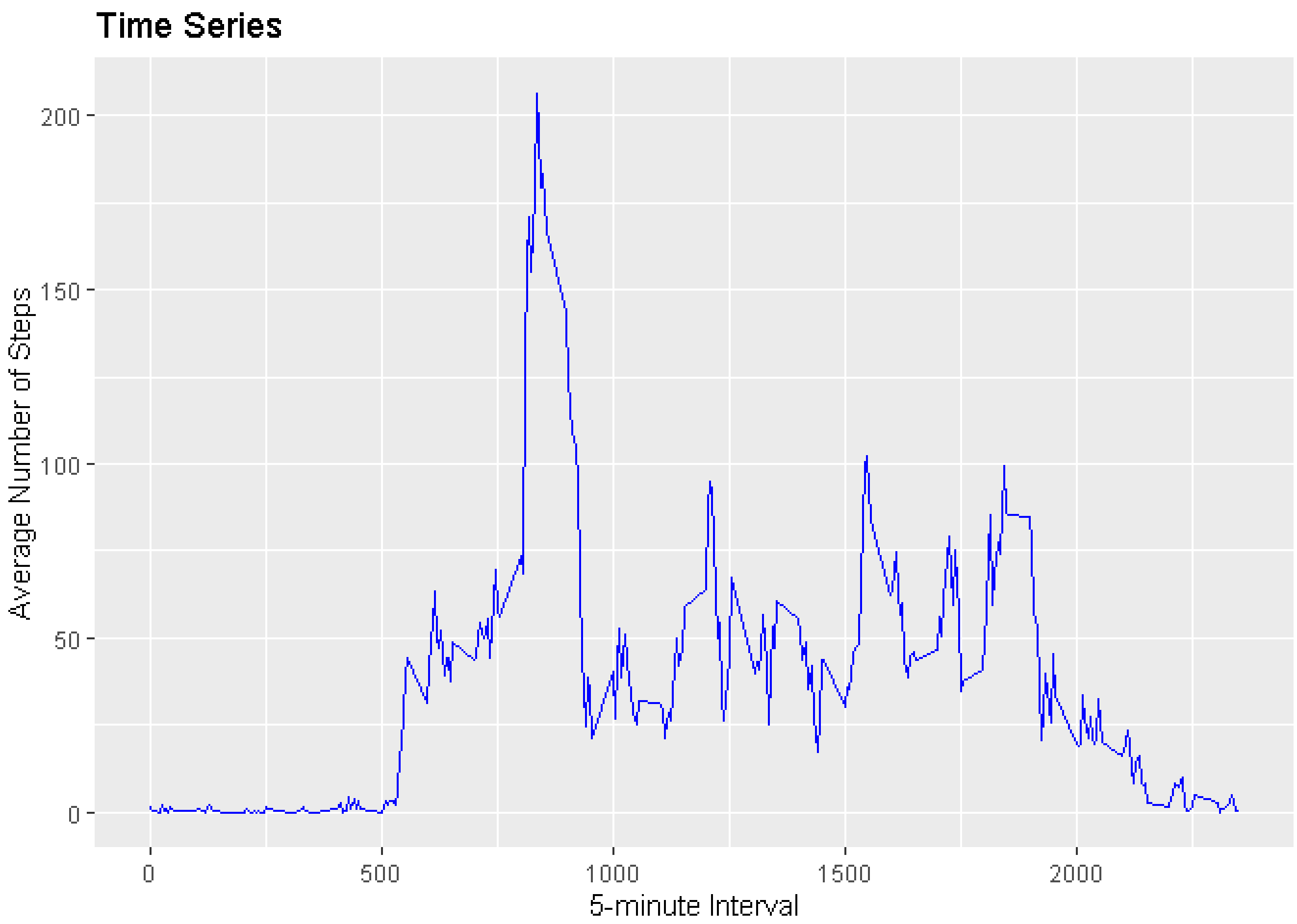
```
summary(vecTotalStepsPerDay)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##       0    6778   10395    9354   12811   21194
```

## What is the average daily activity pattern?

```
#dtActivityData <- as.data.table(dfActivityData)
dtInterval <- as.data.table(dfActivityData)[, c(lapply(.SD, mean, na.rm = TRUE)), .
SDcols = c("steps"), by = .(interval)]
```

### Make a time series plot (i.e. type = “l”) of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis)



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

```
dtInterval[steps == max(steps), .(max_interval = interval)]
```

```
##    max_interval
## 1:          835
```

## Imputing missing values

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

```
numNASteps <- length(which(is.na(dfActivityData$steps)))
print(numNASteps)
```

```
## [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.

Taking mean

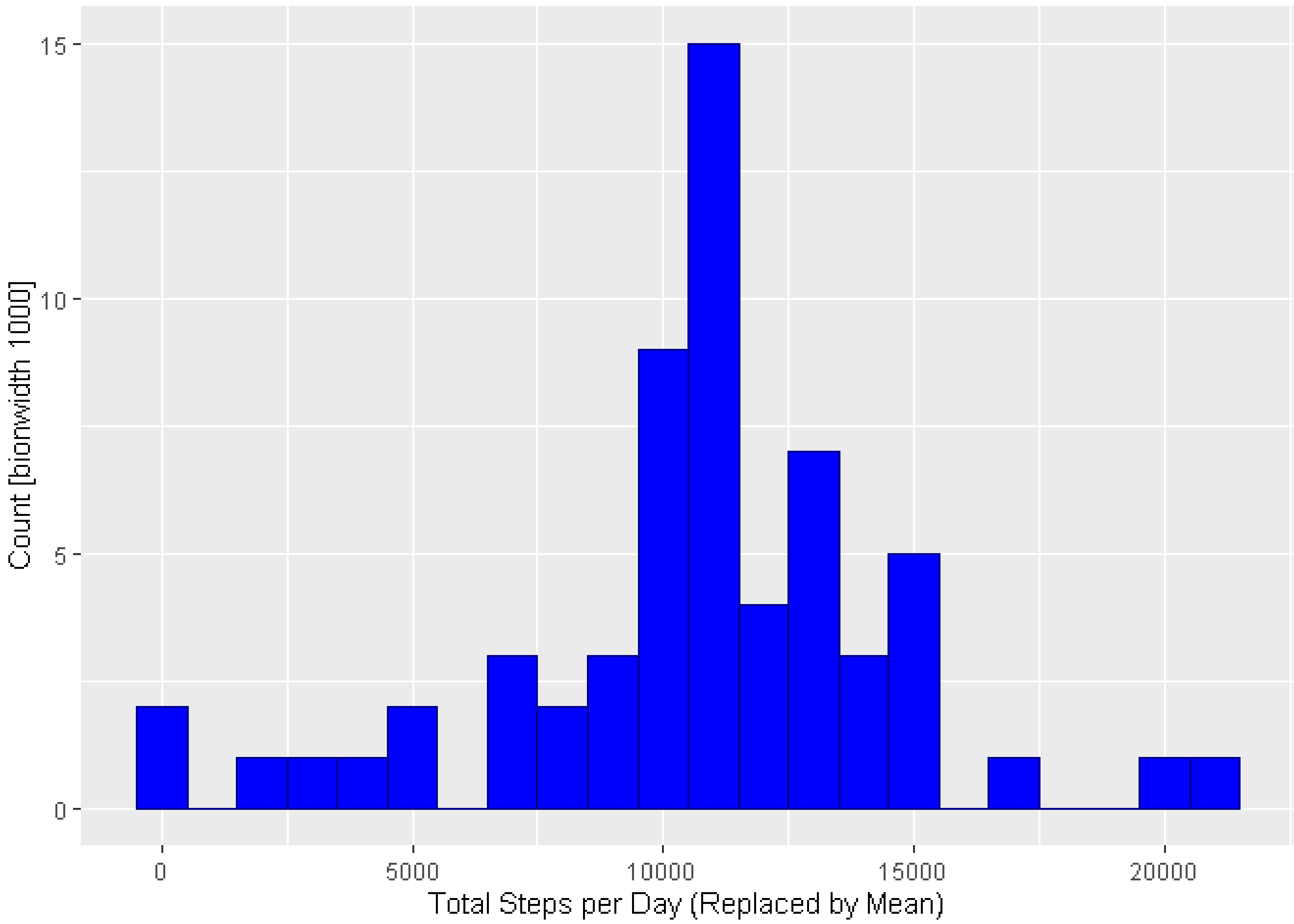
### Create a new dataset that is equal to the original dataset but with the missing data filled in.

```
dfActivityDataRep <- dfActivityData
dfActivityDataRep[is.na(dfActivityDataRep)] <- as.integer(mean(dfActivityData$step
s, na.rm = TRUE))
```

### 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?

```
vecTotalStepsPerDayRep <- tapply(dfActivityDataRep$steps, dfActivityDataRep$date, s
um, na.rm=TRUE)
summary(vecTotalStepsPerDayRep)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##      41    9819   10656   10752   12811   21194
```

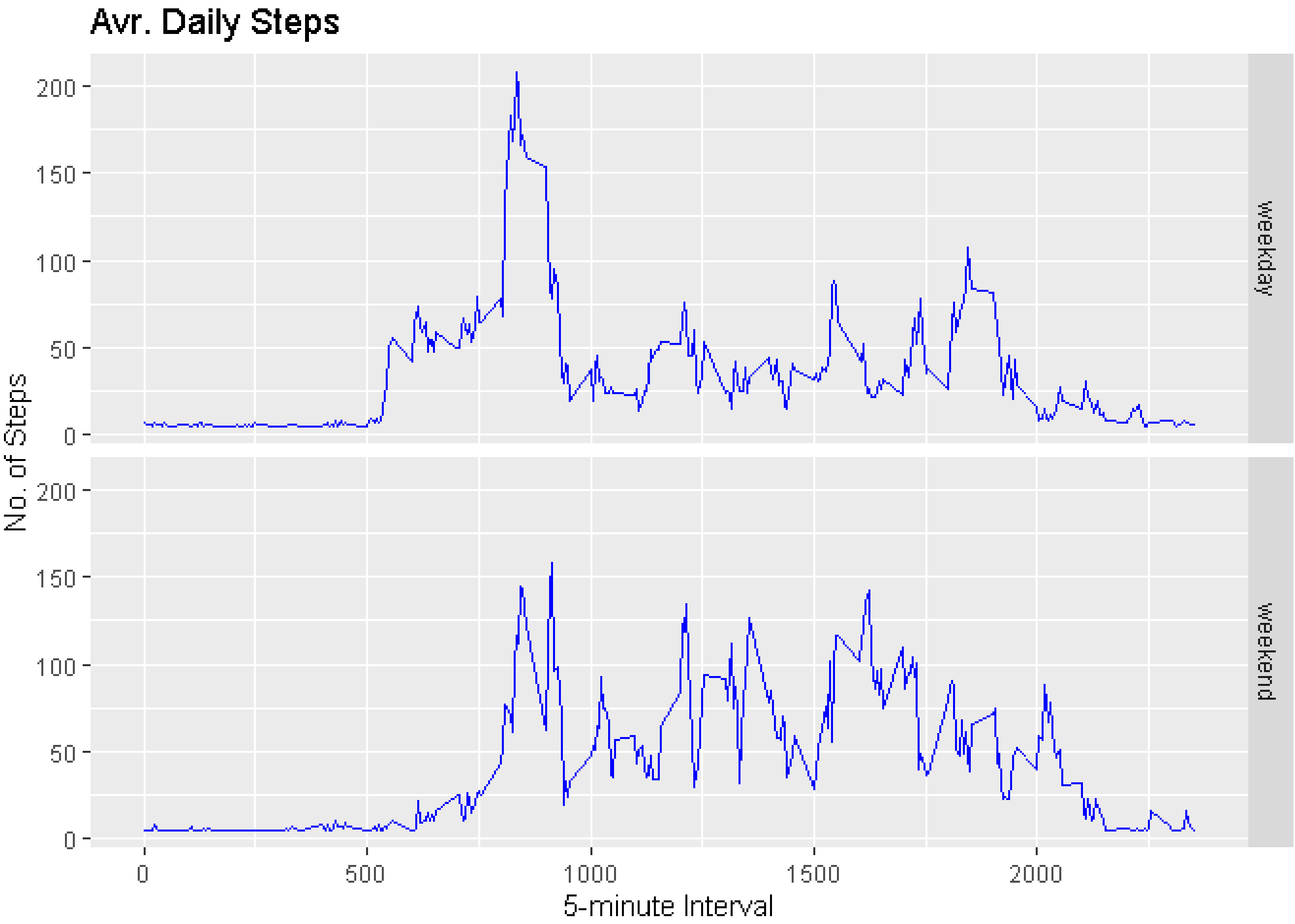


## Are there differences in activity patterns between weekdays and weekends?

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.

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

Make a panel plot containing a time series plot (i.e. 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). See the README file in the GitHub repository to see an example of what this plot should look like using simulated data.


Avr. Daily Steps
No. of Steps
weekday
weekend
5-minute Interval