Reproducible Research: Peer Assessment 1

Loading and preprocessing the data

```
## Reading the activity file from a working directory.

setwd("~/Documents/Coursera/Data Science/05 - Reproducible Research/Project 1")
activity <- read.table("activity.csv", header = TRUE, sep = ",")
```

What is mean total number of steps taken per day?

```
## Making a histogram of the total number of steps per day.

StepsByDay <- aggregate(steps ~ date, activity, sum, na.rm = TRUE)
hist(StepsByDay$steps, main = paste("Total Steps per Day"),
     col = "white", xlab = "Number of Steps")
```

```
## Calculating and reporting mean and median of the total number of steps per day.

print(paste("The mean is", round(mean(StepsByDay$steps)),
            ", and the median is", round(median(StepsByDay$steps)),"."))
```

```
## [1] "The mean is 10766 , and the median is 10765 ."
```

What is the average daily activity pattern?

```
## Calculating the average number of steps taken.

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

## Plotting 5-minutes interval, considering the average across all days.

plot(StepsInterval$interval, StepsInterval$steps,
     type = "l", xlab = "5-minute Interval", ylab = "Average Steps",
     main = "Average Steps per Interval")
```

Imputing missing values

```
## Calculating and reporting dataset missing values.

print(length(which(is.na(activity$steps))))
```

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

```
## Making a histogram of the total number of steps per day for the new dataset.

stepsByDayImputed <- tapply(impute(activity$steps, fun=mean), activity$date, sum)
hist(stepsByDayImputed, main = paste("Total Steps per Day"),
     col = "white", xlab = "Number of Steps")
```

```
## Calculating and reporting mean and median of the total number of steps per day.

print(mean(stepsByDayImputed))
```

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

```
print(median(stepsByDayImputed))
```

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

Are there differences in activity patterns between weekdays and weekends?

```
## Creating a new variable in the dataset with two levels -- "weekday" and "weekend".

activity$date <- ifelse(as.POSIXlt(activity$date)$wday %in% c(0,6), 'weekend', 'weekday')
averagedActivity <- aggregate(steps ~ interval + date, data = activity, mean)

## Making a panel plot containing a time series of the 5-minute interval and
## the average number of steps taken.

ggplot(averagedActivity, aes(interval, steps)) + geom_line() + facet_grid(date ~ .) +
       xlab("5-minute Interval") + ylab("Number of Steps (Average)")
```