Reproducible Research: Peer Assessment 1

Loading and preprocessing the data

```
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(chron)
```

```
## Warning: package 'chron' was built under R version 4.1.3
```

```
library(knitr)
unzip("activity.zip")
data <- read.csv("activity.csv", colClasses = c("numeric","Date","numeric"))
```

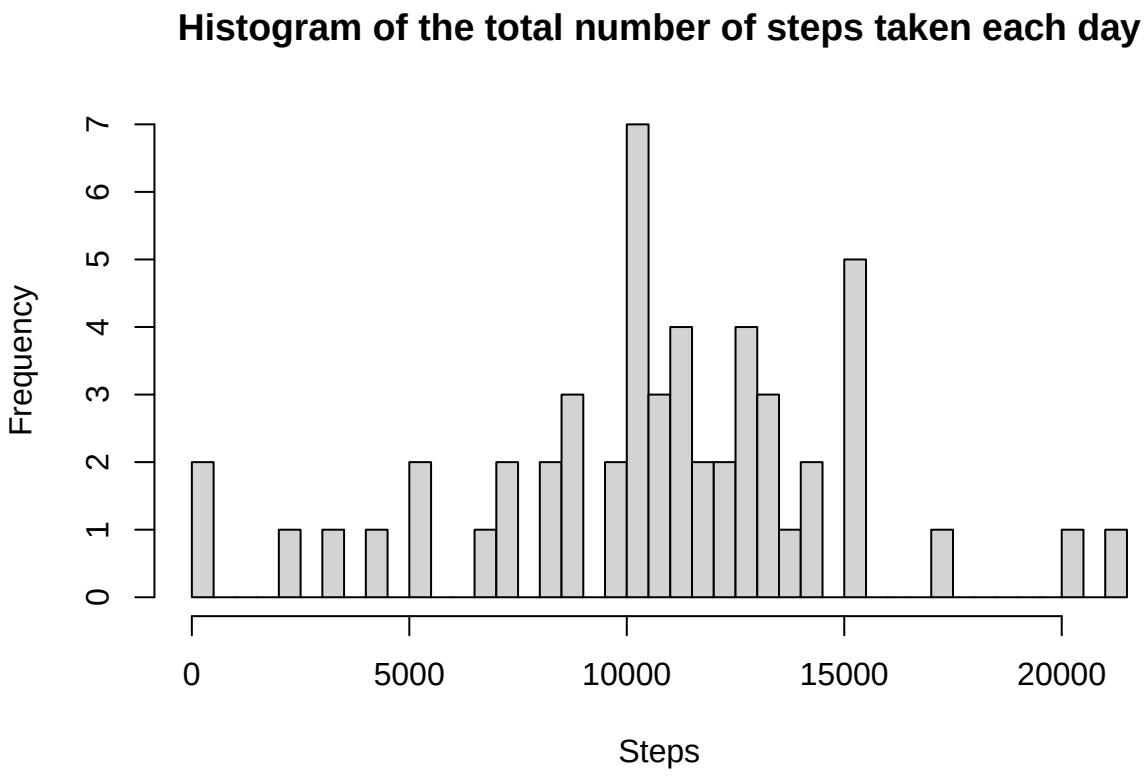
What is mean total number of steps taken per day?

For this part of the assignment, we will ignore the missing values in the dataset.

First, we are going to calculate the total number of steps taken for each day.

Then we make a histogram of the total number of steps taken each day.

```
data_day <- aggregate(steps ~ date, data, sum)
hist(data_day$steps, breaks = 53, xlab = "Steps",
     main = "Histogram of the total number of steps taken each day")
```



Mean and median of the total number of steps taken per day:

Next, we need to calculate and report the mean and median total number of steps taken per day.

```
mean(data_day$steps)
```

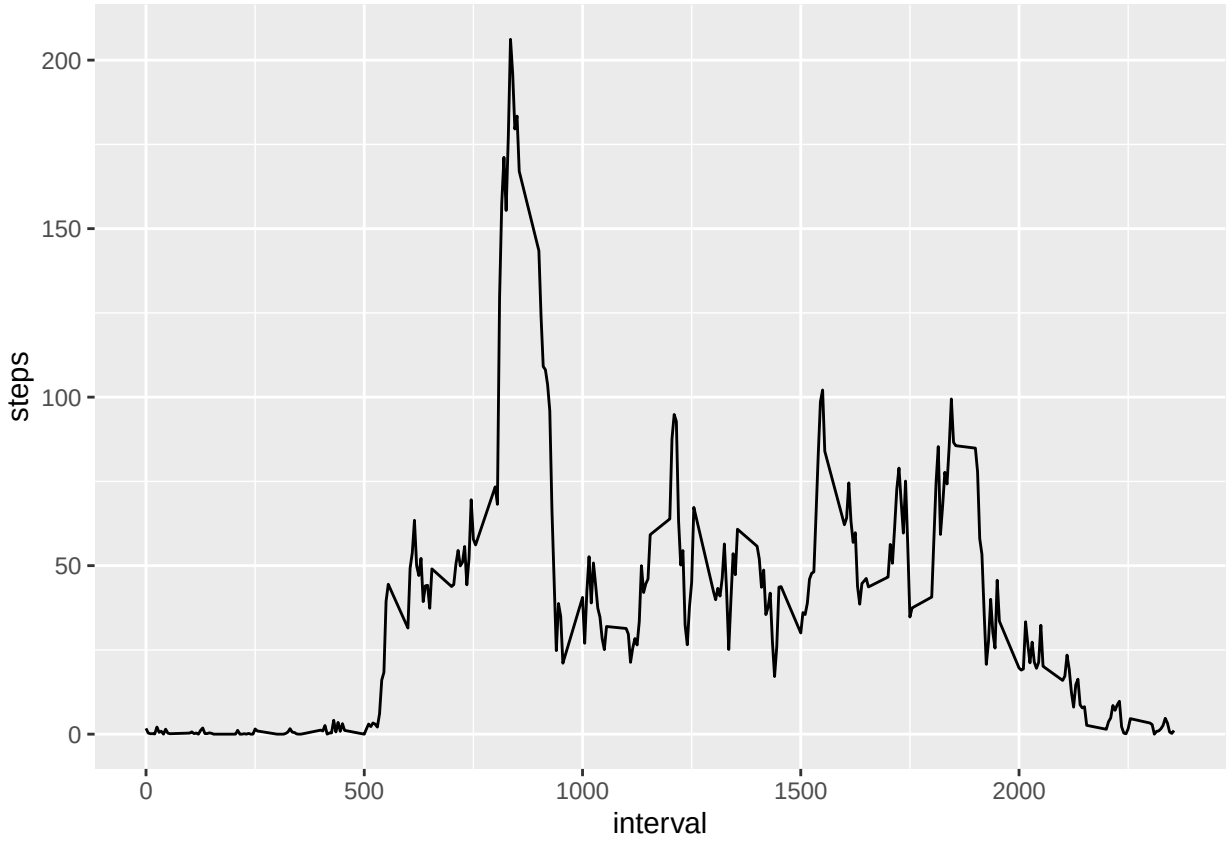
```
## [1] 10766.19
```

```
median(data_day$steps)
```

```
## [1] 10765
```

What is the average daily activity pattern?

```
data_interval <- aggregate(steps ~ interval, data, mean)
ggplot(data_interval, aes(x=interval, y=steps)) +
        geom_line()
```



Therefore, across all the days in the dataset on average, the 5-minute interval contains the maximum number of steps is 8:35 am. ### This interval contains the maximum number of steps:

```
ind <- match(max(data_interval$steps), data_interval$steps)
data_interval$interval[ind]
```

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

Imputing missing values

Note that 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.

First, we calculate the total number of missing values in the dataset:

```
sum(!complete.cases(data_day))
```

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

Thus, the total number of missing values in the dataset is 2304.

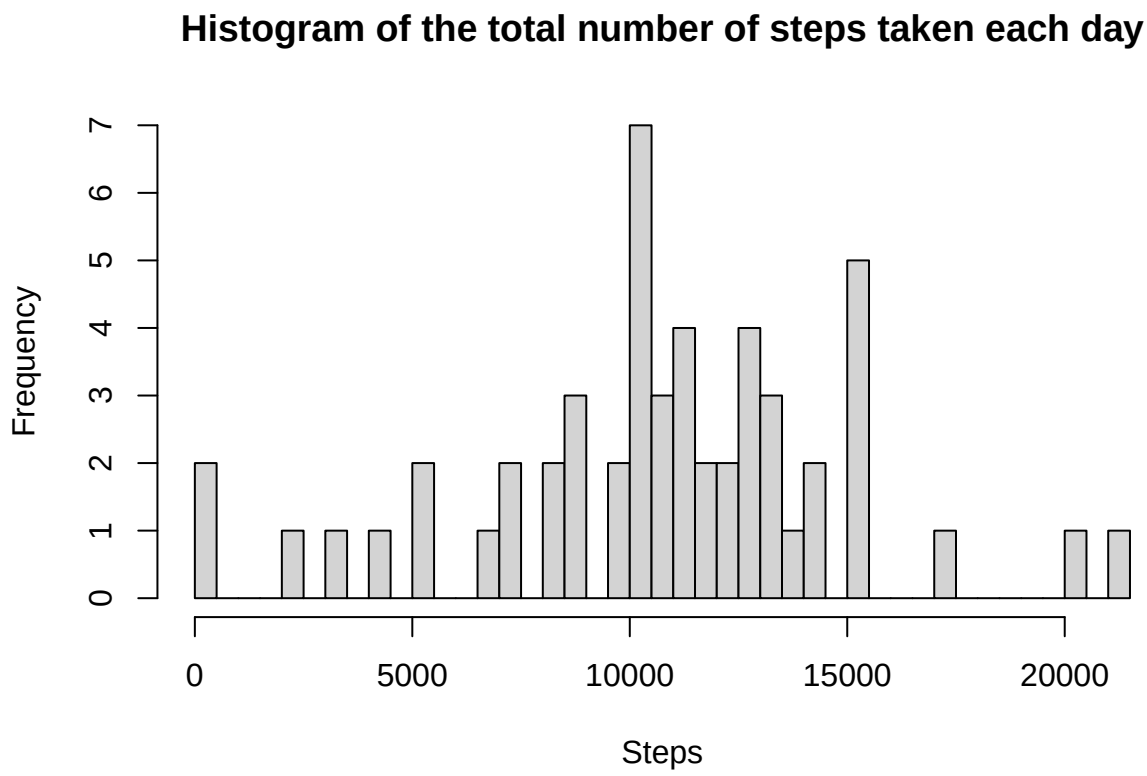
The strategy we are going implement for imputing missing values is to use the mean for the 5-minute interval.

We are to create a new dataset that is copied from the original data but with the missing values filled in by using the stragegy mentioned above. ### Total number of missing values:

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

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

```
data1 <- data
data1$steps[is.na(data1$steps)] <- mean(data1$steps, na.rm = T)
data_day1 <- aggregate(steps ~ date, data, sum)
hist(data_day1$steps, breaks = 53, xlab = "Steps",
     main = "Histogram of the total number of steps taken each day")
```



Mean and median of the total number of steps taken per day:

```
mean(data_day1$steps)
```

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

```
median(data_day1$steps)
```

```
## [1] 10765
```

Differences median and mean:

```
mean(data_day$steps) - mean(data_day1$steps)
```

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

```
median(data_day$steps) - median(data_day1$steps)
```

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

Are there differences in activity patterns between weekdays and weekends?

```
data1$day <- ifelse(is.weekend(data1$date), "weekend", "weekday")
data_intdays <- aggregate(steps ~ interval+day, data1, mean)
ggplot(data_intdays, aes(x=interval, y=steps)) +
        geom_line() +
        facet_wrap(~day, nrow=2) +
        labs(x="Interval", y="Steps")
```