Reproducible Research: Peer Assessment 1

Ong Kwee Hian

Sunday, May 17, 2015

==========

Loading and preprocess/transform the data

```
unzip("activity.zip")
dataActivity <- read.csv("activity.csv")
```

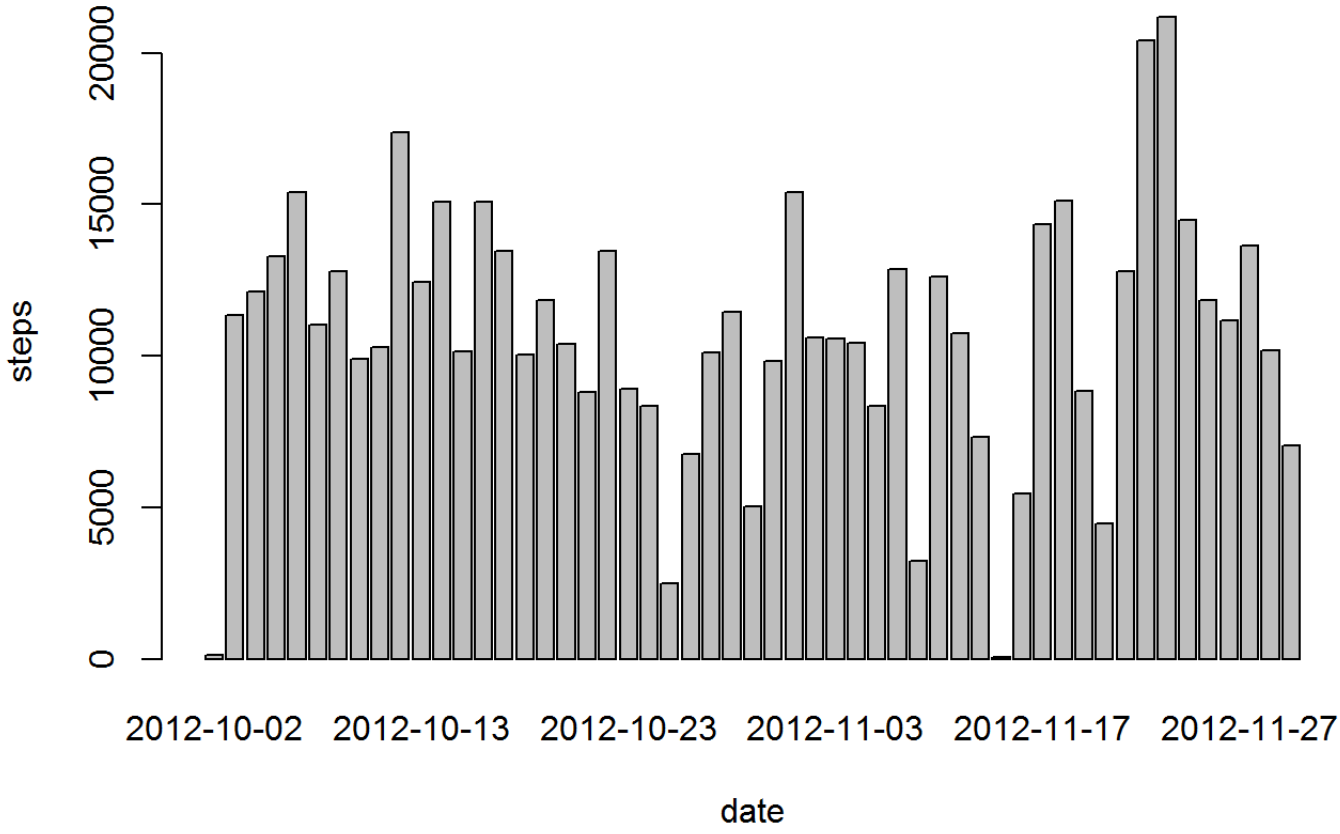
What is the mean total number of steps taken per day?

1. Calculate the total number of steps taken per day

```
stepsByDay <- aggregate(steps ~ date, data=dataActivity, FUN=sum)
```

2. Make a histogram of the total number of steps taken each day

```
barplot(stepsByDay$steps, names.arg=stepsByDay$date, xlab="date", ylab="steps")
```



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

```
mean(stepsByDay$steps)
```

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

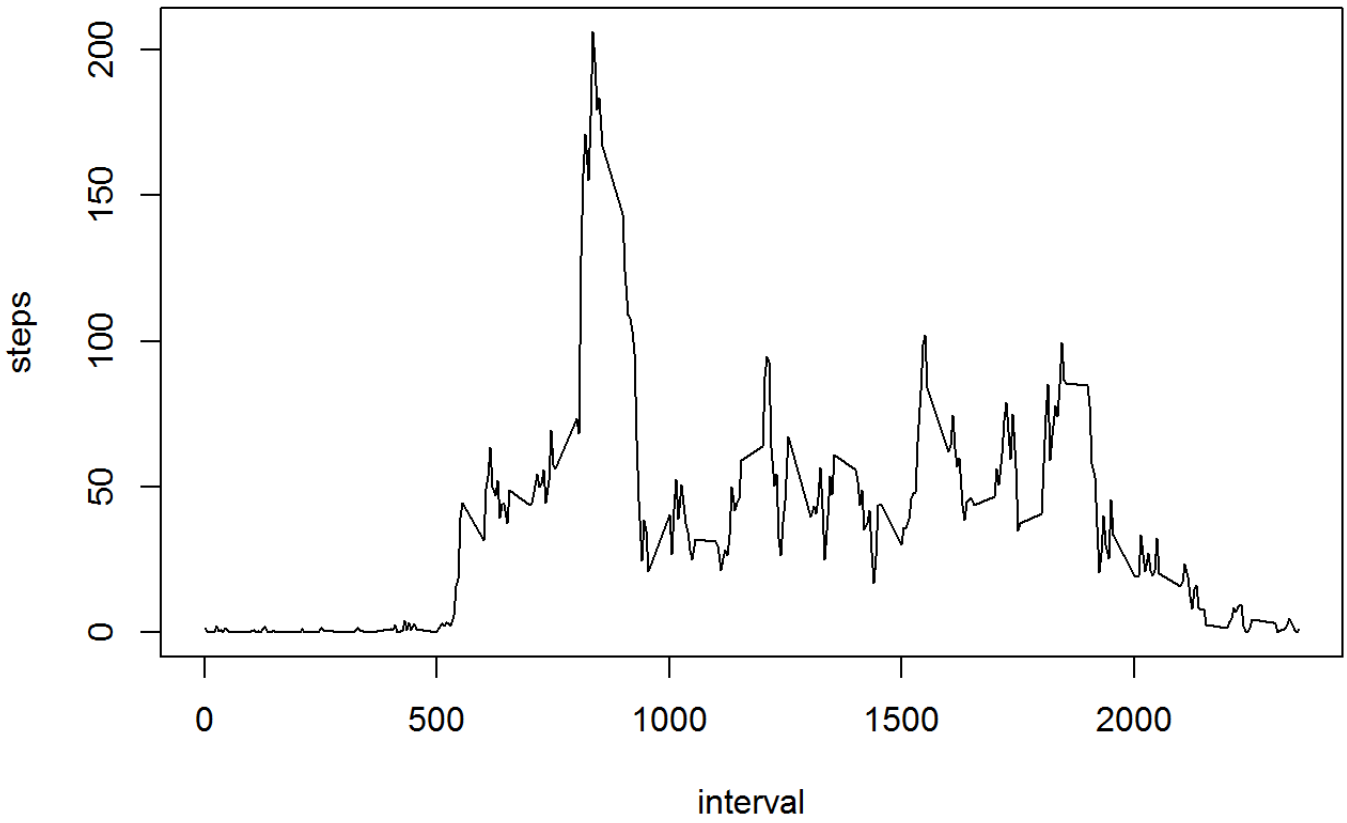
```
median(stepsByDay$steps)
```

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

What is the average daily activity pattern?

1. 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)

```
stepsByInterval <- aggregate(steps ~ interval, data=dataActivity, FUN=mean)
plot(stepsByInterval, type="l")
```



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

```
stepsByInterval$interval[which.max(stepsByInterval$steps)]
```

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

Imputing missing values

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

```
sum(is.na(dataActivity))
```

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

2. 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.

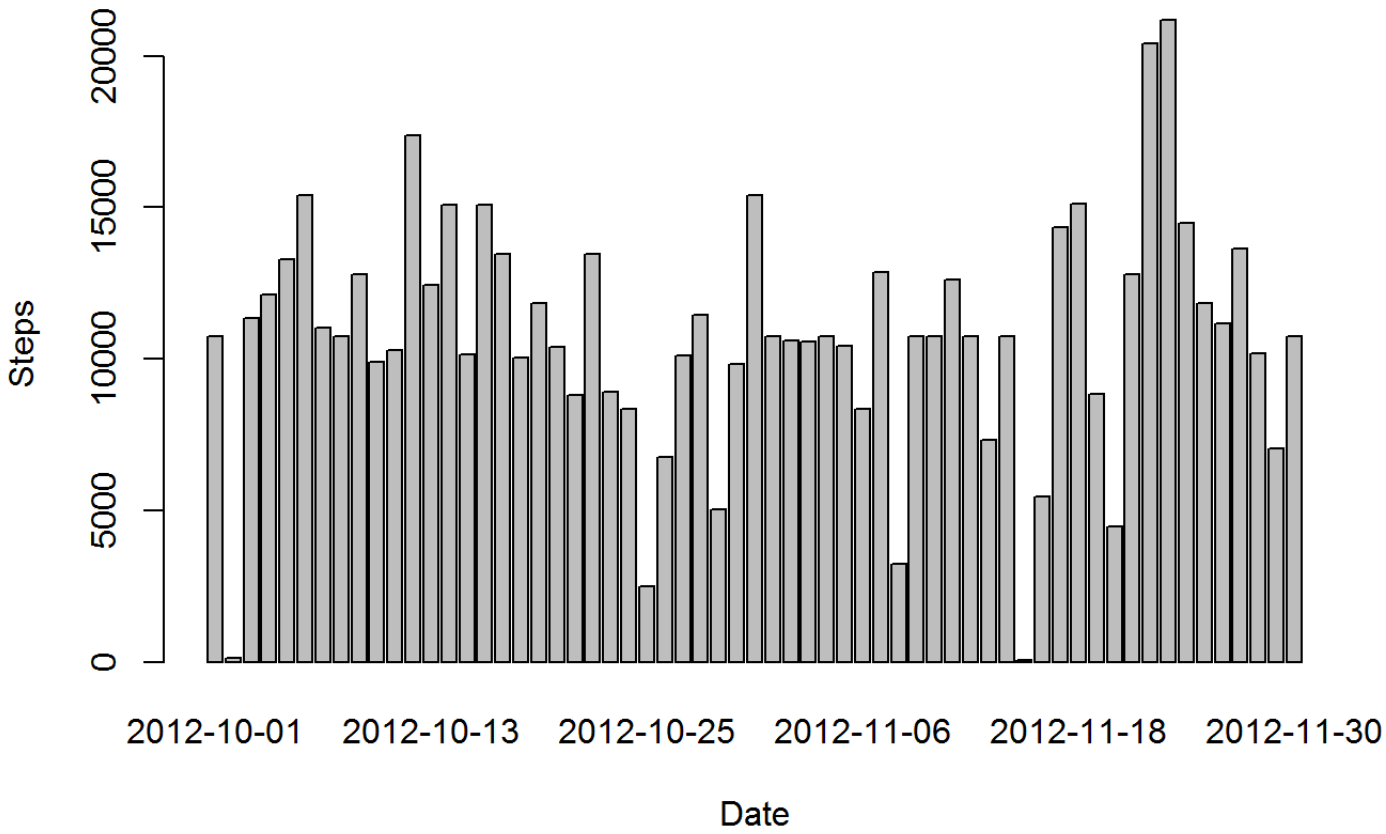
The missing values will be replaced by the means for the 5-minute intervals.

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

```
dataActivity <- merge(dataActivity, stepsByInterval, by="interval", suffixes=c("",".tmp"))
naList <- is.na(dataActivity$steps)
dataActivity$steps[naList] <- dataActivity$steps.tmp[naList]
dataActivity <- dataActivity[,c(1:3)]
```

4. 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?

```
stepsByDay <- aggregate(steps ~ date, data=dataActivity, FUN=sum)
barplot(stepsByDay$steps, names.arg=stepsByDay$date, xlab="Date", ylab="Steps")
```




```
mean(stepsByDay$steps)
```

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

```
median(stepsByDay$steps)
```

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

Yes, these values differ from the estimates from the first part of the assignment due to the replacement of NA values, however the impact of the missing data to estimate the total number of steps per day is low.

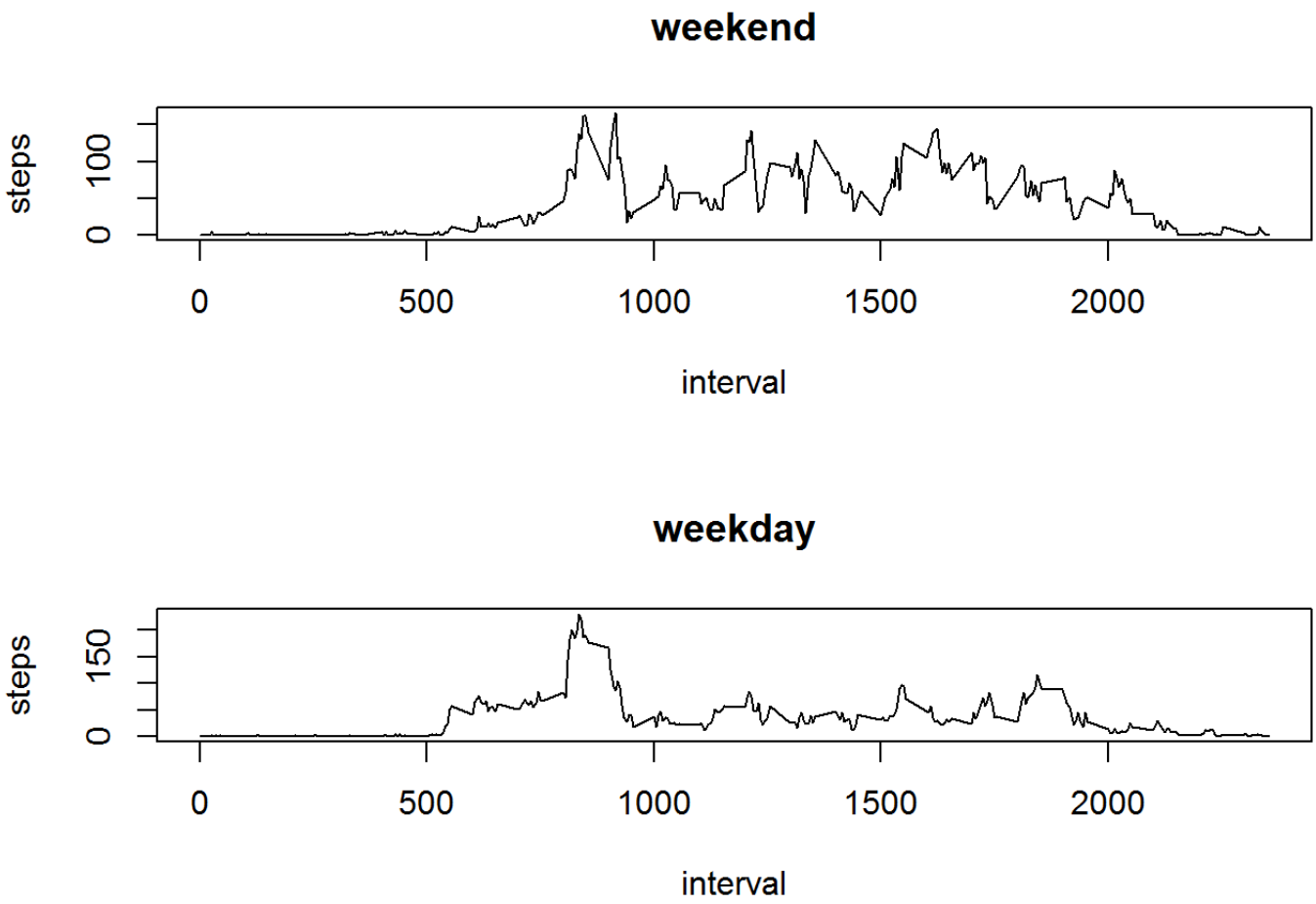
Are there differences in activity patterns between weekdays and weekends?

1. 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.

```
Evalday <- function(date) {
    if (weekdays(as.Date(date)) %in% c("Saturday", "Sunday")) {
        "weekend" }
    else {"weekday"}}
dataActivity$Evalday <- as.factor(sapply(dataActivity$date, Evalday))
```

2. 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).

```
par(mfrow=c(2,1))
for (DayType in c("weekend", "weekday")) {
    stepsByDayType <- aggregate(steps ~ interval, data=dataActivity, subset=dataActivity$Evalday==Day
Type, FUN=mean)
    plot(stepsByDayType, type="l", main=DayType)
}
```


weekend
steps
0
100
0
500
1000
1500
2000
interval
weekday
steps
0
150
0
500
1000
1500
2000
interval