Activity Monitoring Data Analysis

Xiang Li

May 17, 2017

Loading and preprocessing the data

```
activity = read.csv("Q:/Desktop/Coursera/5. Reproducible Research/Course Project 1/activity.csv")
```

```
# Explore the dataset
head(activity)
```

```
##   steps       date interval
## 1    NA 2012-10-01        0
## 2    NA 2012-10-01        5
## 3    NA 2012-10-01       10
## 4    NA 2012-10-01       15
## 5    NA 2012-10-01       20
## 6    NA 2012-10-01       25
```

```
summary(activity)
```

```
##      steps                date          interval
##  Min.   :  0.00   2012-10-01:  288   Min.   :   0.0
##  1st Qu.:  0.00   2012-10-02:  288   1st Qu.: 588.8
##  Median :  0.00   2012-10-03:  288   Median :1177.5
##  Mean   : 37.38   2012-10-04:  288   Mean   :1177.5
##  3rd Qu.: 12.00   2012-10-05:  288   3rd Qu.:1766.2
##  Max.   :806.00   2012-10-06:  288   Max.   :2355.0
##  NA's   :2304     (Other)   :15840
```

```
str(activity)
```

```
## 'data.frame':    17568 obs. of  3 variables:
##  $ steps   : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ date    : Factor w/ 61 levels "2012-10-01","2012-10-02",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ interval: int  0 5 10 15 20 25 30 35 40 45 ...
```

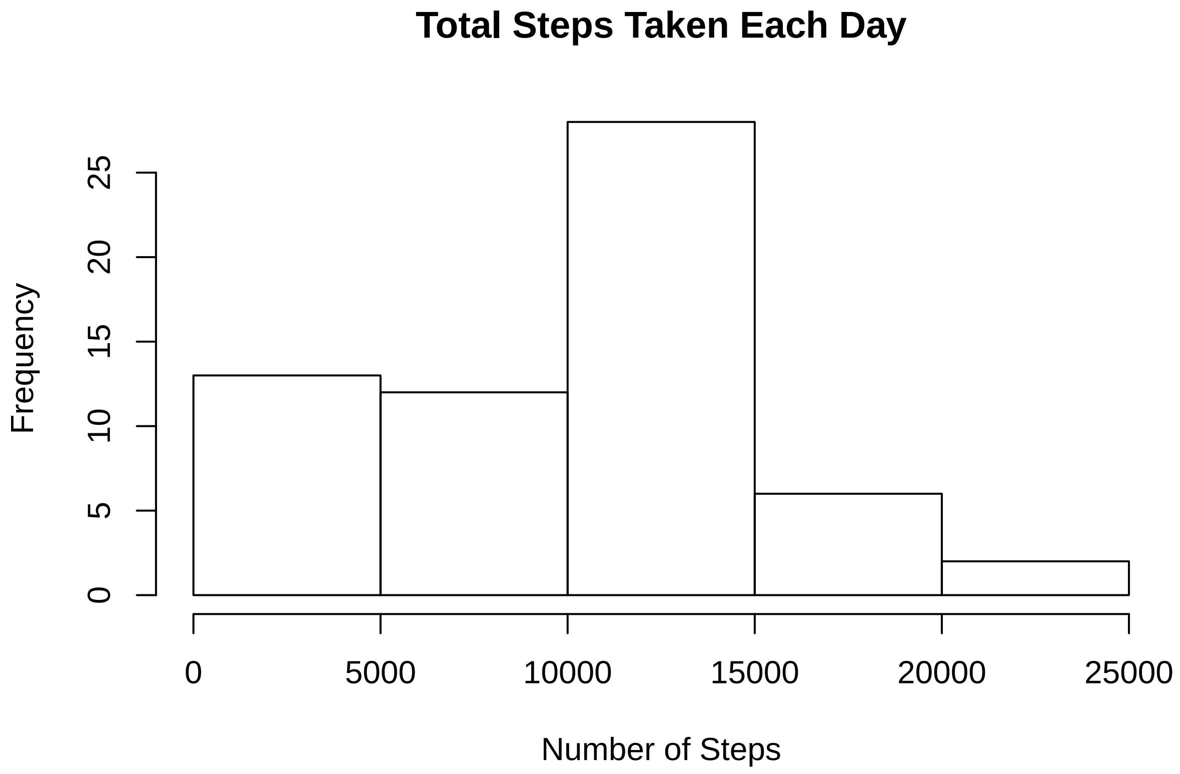
What is mean total number of steps taken per day?

```
# Calculate the steps by date
library(data.table)
```

```
## Warning: package 'data.table' was built under R version 3.3.2
```

```
activity = data.table(activity)
totalsteps = activity[, list(total_steps = sum(steps, na.rm = TRUE)), by = date]

# Plot the histogram
hist(totalsteps$total_steps, main = "Total Steps Taken Each Day", xlab = "Number of Steps")
```



```r
# calculate the mean number of steps taken each day
mean_step = round(mean(totalsteps$total_steps, na.rm = TRUE),0)
mean_step
```

```
## [1] 9354
```

```r
# calculate the median number of steps taken each day
median_step = round(median(totalsteps$total_steps, na.rm = TRUE),0)
median_step
```

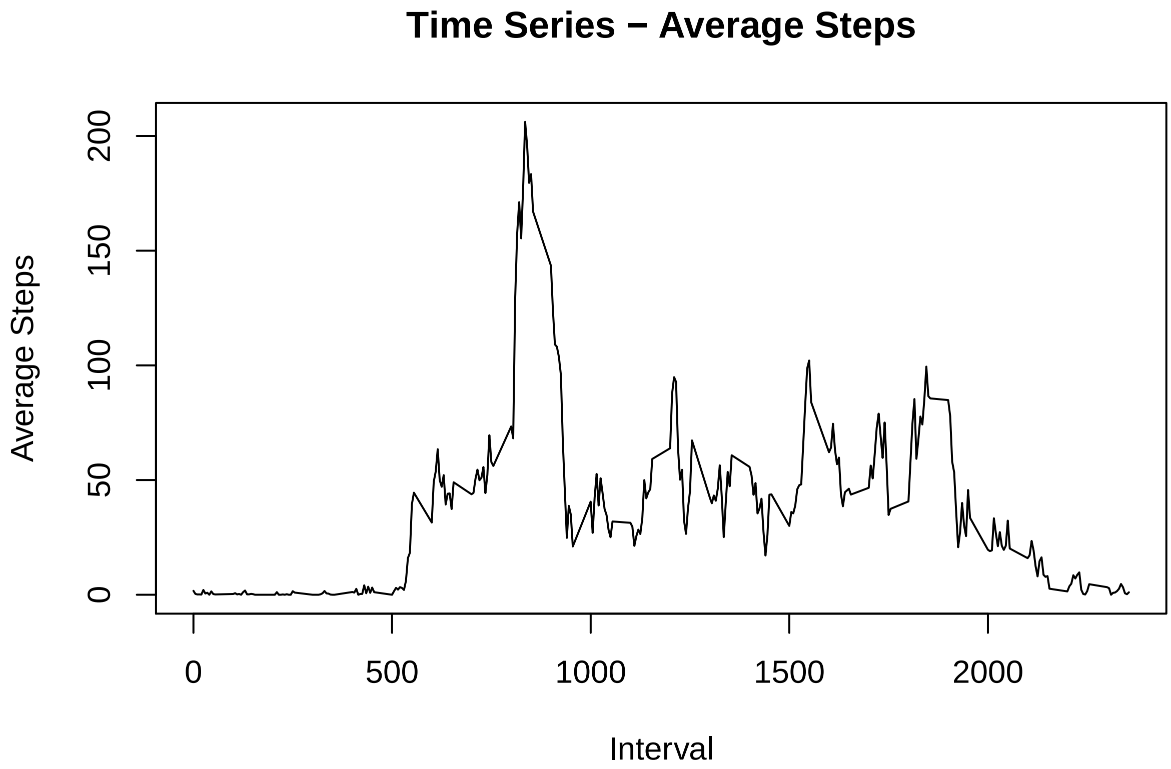
```
## [1] 10395
```

The mean of steps taken each day is 9354 steps and the median is 10395 steps.

What is the average daily activity pattern?

```r
# Calculate the average steps by interval
library(data.table)
activity = data.table(activity)
avgsteps = activity[, list(avg_steps = mean(steps, na.rm = TRUE)), by = interval]

# Plot the time series chart
with(avgsteps, plot(interval, avg_steps, type = "l", main = "Time Series - Average Steps",
                    xlab = "Interval", ylab = "Average Steps"))
```



```r
#The interval that contains the maximum number of steps
avgsteps[which.max(avgsteps$avg_steps),]$interval
```

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

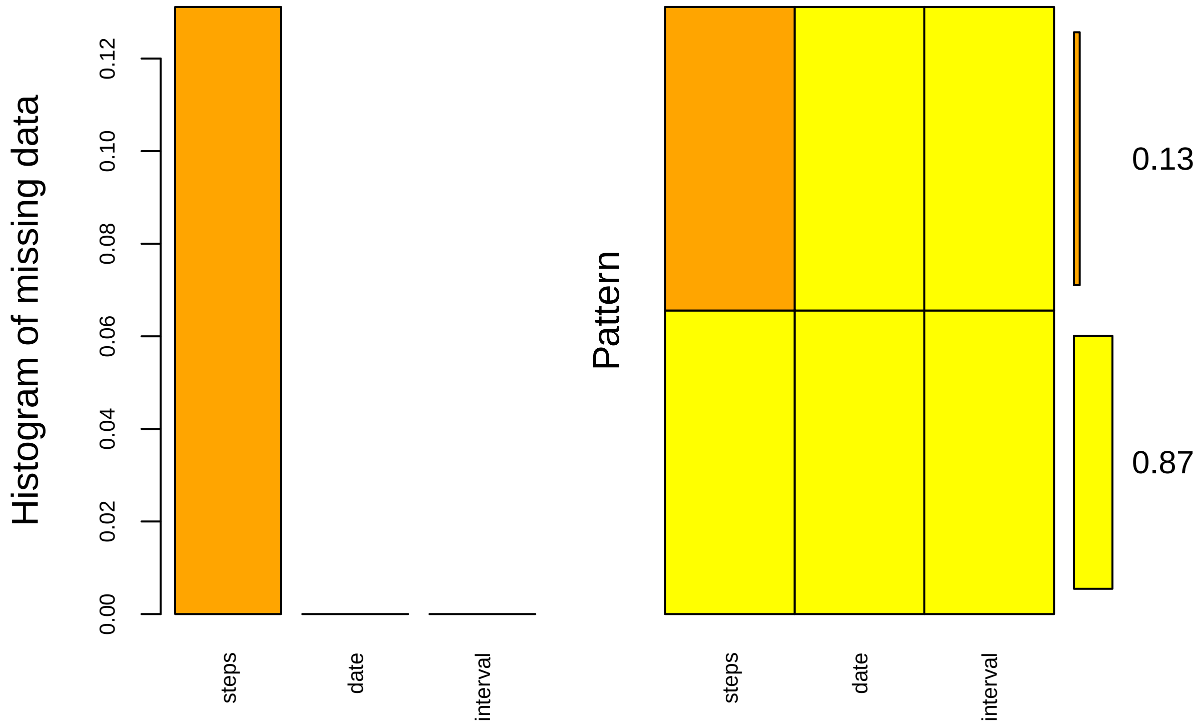
Imputing missing values

```r
# Look at missing data pattern
library(mice)
md.pattern(activity)
```

```
##       date interval steps
## 15264    1        1     1    0
##  2304    1        1     0    1
##          0        0  2304 2304
```

The output tells us that 15264 samples are complete, 2304 samples miss only the steps value.

```r
# Visualize missing data pattern
library(VIM)
aggr_plot = aggr(activity, col = c('yellow', 'orange'),
                 numbers = TRUE, sortVars = TRUE, labels = names(activity),
                 cex.axis = 0.7, gap = 3, ylab = c("Histogram of missing data", "Pattern"))
```



The plot helps us understanding that 87% of the samples are not missing any information, 13% are missing the Steps value.

```r
# Impute the missing data
tempData = mice(activity, m = 5, maxit = 50, meth = "pmm", seed = 500)
```

```r
summary(tempData)
```

```
## Multiply imputed data set
## Call:
## mice(data = activity, m = 5, method = "pmm", maxit = 50, seed = 500)
## Number of multiple imputations:  5
## Missing cells per column:
##    steps     date interval
##     2304        0        0
## Imputation methods:
##    steps     date interval
##    "pmm"    "pmm"    "pmm"
## VisitSequence:
## steps
##     1
## PredictorMatrix:
##          steps date interval
## steps        0    1        1
## date         0    0        0
## interval     0    0        0
## Random generator seed value:  500
```

```
# Check the imputed data
head(tempData$imp$steps, 10)
```

```
##     1   2  3   4   5
## 1   0  33  0  84 179
## 2   0   0 25   0   0
## 3   0   0  0   0   0
## 4   0  19  0   0   0
## 5   0   0 23 189   0
## 6   0  73 23   0   0
## 7   0   0  0   0 111
## 8   0   0  0   0   0
## 9   0 415  0   0   0
## 10  0 356  0   0   0
```

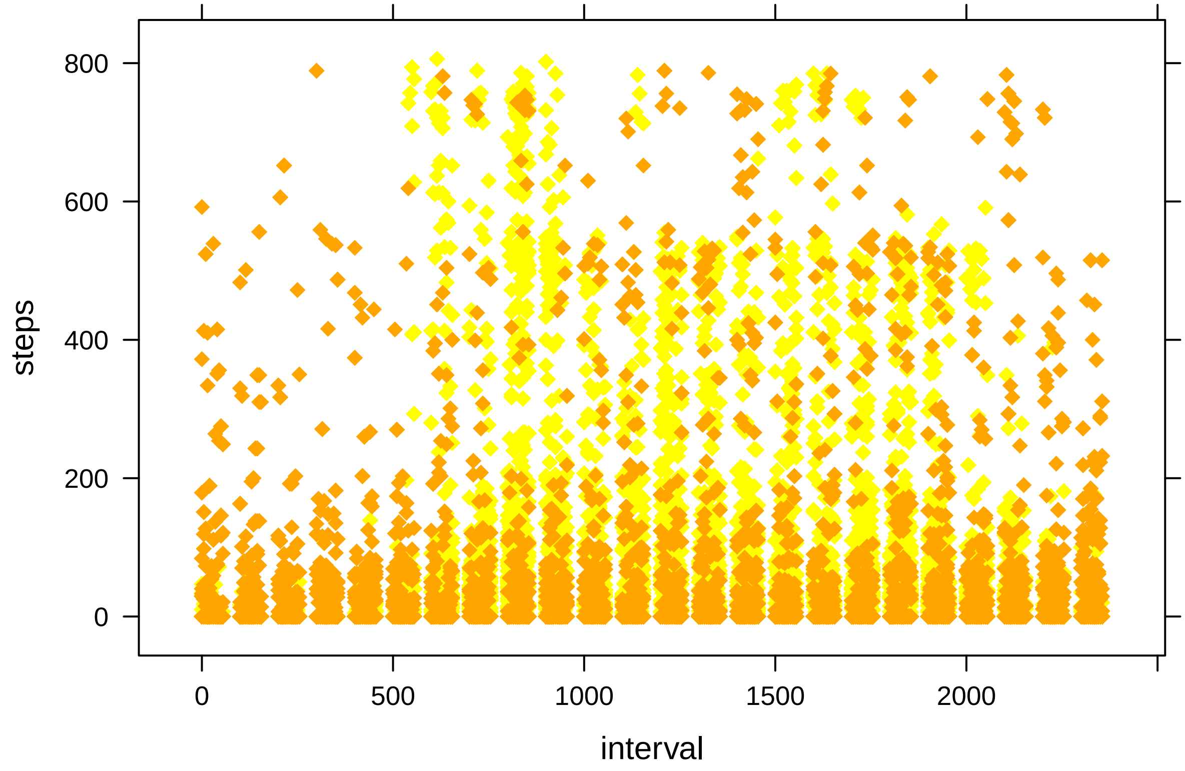
The output shows the imputed data for each observation (first column left) within each imputed dataset (first row at the top).

```
# Get back the completed dataset
Imputed_activity = complete(tempData, 1)
```

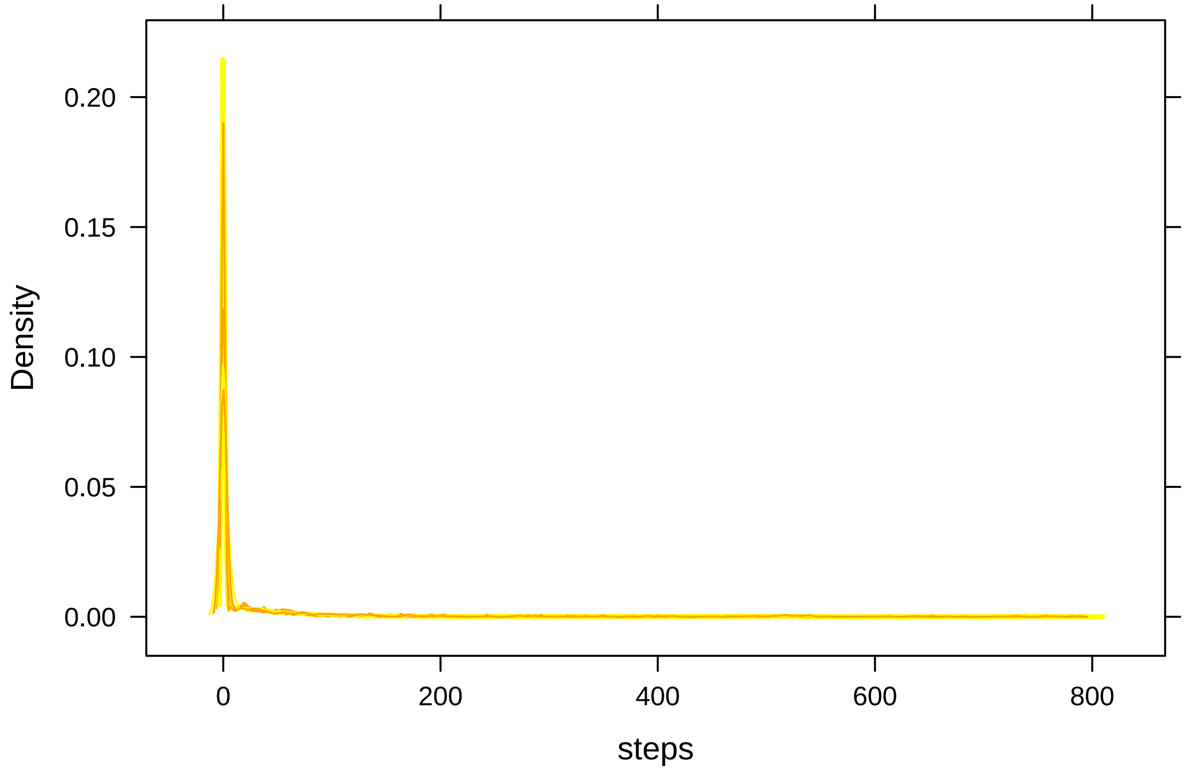
```
# Inspect the distribution of original and imputed data
library(lattice)
```

```
## Warning: package 'lattice' was built under R version 3.3.3
```

```
# Scatterplot to plot steps against interval
xyplot(tempData,steps ~ interval, pch = 18, cex = 1, col = c('yellow', 'orange'))
```



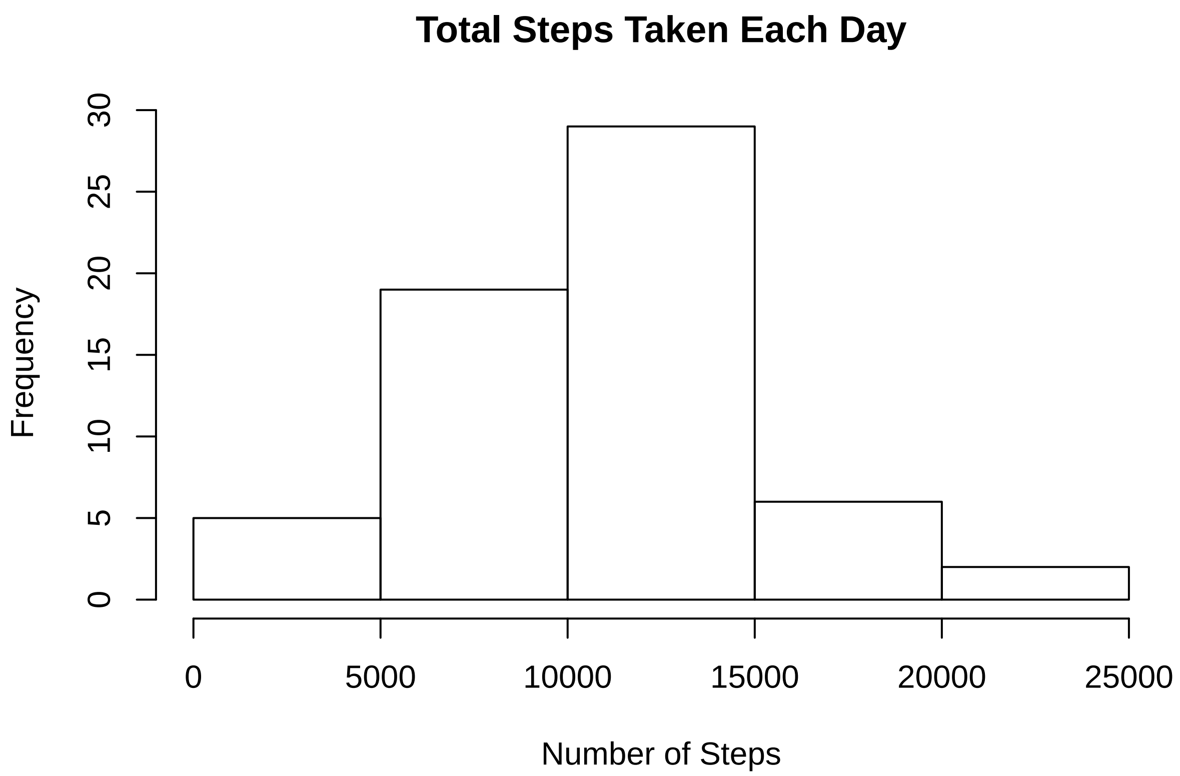
```r
# Density plot
densityplot(tempData, col = c('yellow', 'orange'))
```



The density of the imputed data for each imputed dataset is showed in yellow while the density of the observed data is showed in orange. We expect the distributions to be similar.

```r
# Histogram of the total number of steps taken each day after missing values are imputed
# Calculate the steps by date
library(data.table)
Imputed_activity = data.table(Imputed_activity)
totalsteps1 = Imputed_activity[, list(total_steps = sum(steps)), by = date]

# Plot the histogram
hist(totalsteps1$total_steps, main = "Total Steps Taken Each Day", xlab = "Number of Steps")
```



```
# Calculate the mean number of steps taken each day
mean_step = round(mean(totalsteps1$total_steps),0)
mean_step
```

```
## [1] 10400
```

```
# calculate the median number of steps taken each day
median_step = round(median(totalsteps1$total_steps),0)
median_step
```

```
## [1] 10439
```

Before Imputing: The mean of steps taken each day is 9354 steps and the median is 10395 steps. After Imputing: The mean of steps taken each day is 10400 steps and the median is 10439 steps. Conclusion: Imputing missing data increase the mean and median of the total daily number of steps.

Are there differences in activity patterns between weekdays and weekends?

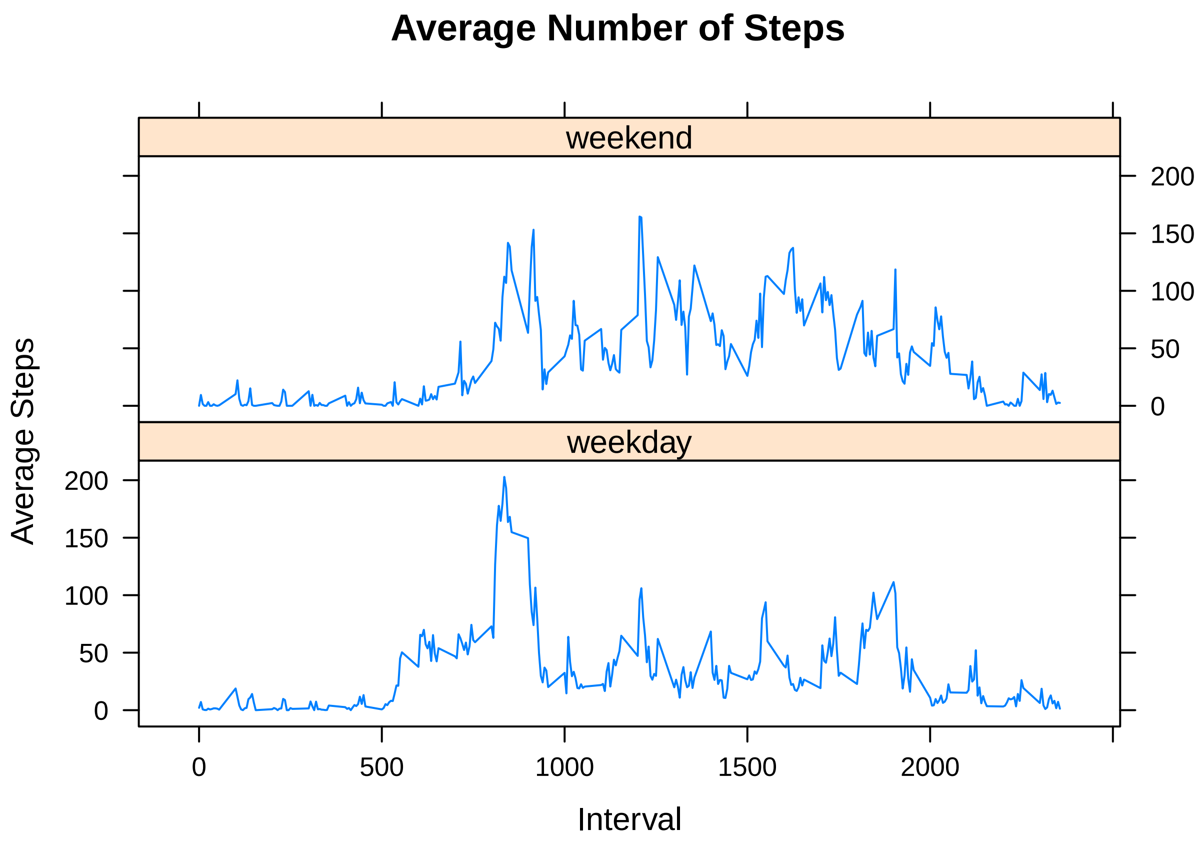
```
# Add a column identifying weekdays and weekends
Imputed_activity$Day <- ifelse(weekdays(as.Date(Imputed_activity$date)) %in% c("Saturday", "Sunday"), "

# Calculate the average steps by interval
library(plyr)
```

```
## Warning: package 'plyr' was built under R version 3.3.3
```

```
avgsteps = ddply(Imputed_activity, .(interval, Day), summarize, avg_steps = mean(steps))

# Plot the time series charts
library(lattice)
xyplot(avgsteps$avg_steps ~ avgsteps$interval | factor(avgsteps$Day), type = 'l', layout = c(1,2), main
```



Conclusion: There are differences in activity patterns between weekdays and weekends. On average this person exercise more on weekend, which makes sense because he/she might have more free time to workout on weekends than on weekdays.