PA1_template.Rmd

Aurora González Vidal

01/17/2015

Introduction

This assignment makes use of data from a personal activity monitoring device. This device collects data at 5 minute intervals through out the day. The data consists of two months of data from an anonymous individual collected during the months of October and November, 2012 and include the number of steps taken in 5 minute intervals each day.

Dataset

The variables included in the dataset `activity.csv` are:

- steps: Number of steps taking in a 5-minute interval (missing values are coded as NA)
- date: The date on which the measurement was taken in YYYY-MM-DD format
- interval: Identifier for the 5-minute interval in which measurement was taken

Loading and preprocessing the data

```
df <- read.csv("activity.csv", sep = ",")
```

We can take a brief look at the data and create a new data frame ommiting NA:

```
head(df)
```

```
##   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
```

```
tail(df)
```

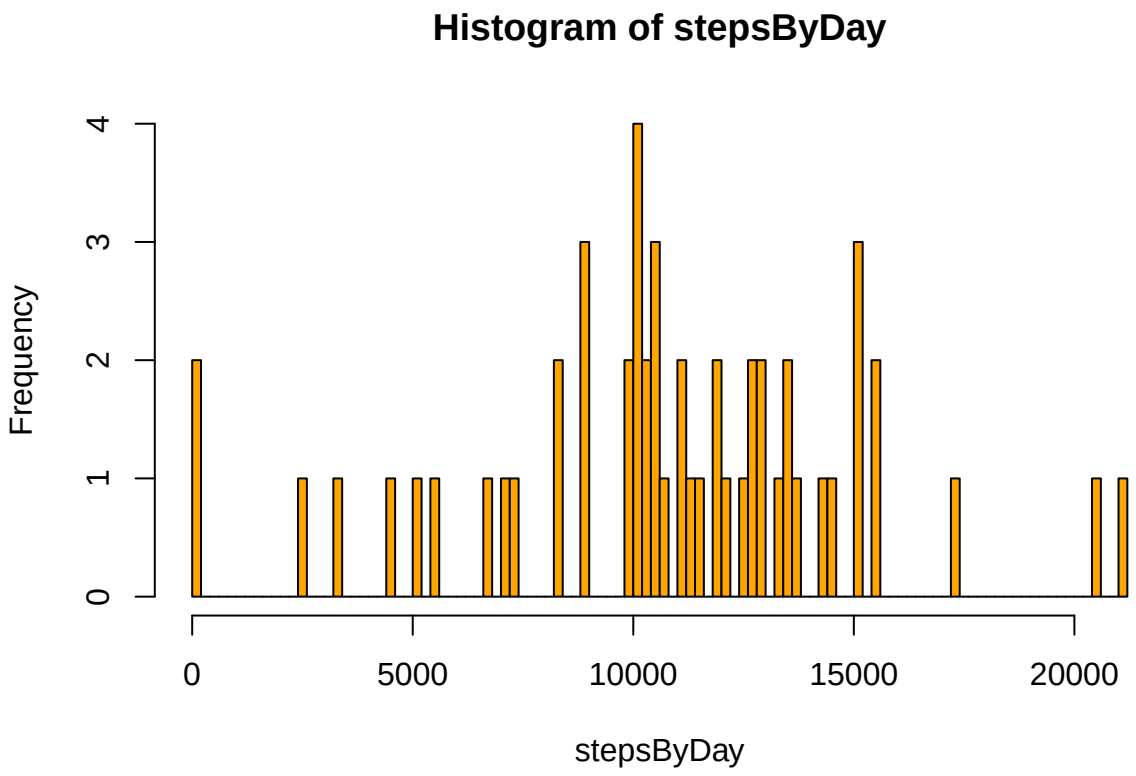
```
##       steps       date interval
## 17563    NA 2012-11-30     2330
## 17564    NA 2012-11-30     2335
## 17565    NA 2012-11-30     2340
## 17566    NA 2012-11-30     2345
## 17567    NA 2012-11-30     2350
## 17568    NA 2012-11-30     2355
```

```
df1<-na.omit(df)
```

What is mean total number of steps taken per day?

Firt, we make a histogram of the total number of steps taken each day

```
stepsByDay <- tapply(df1$steps, df1$date, sum, na.rm = T)
hist(stepsByDay, breaks=100, col = "orange")
```



The mean total number of steps taken per day is

```
m <- mean(stepsByDay, na.rm = T )
m
```

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

And the median total number of steps taken per day is

```
md <- median(stepsByDay, na.rm = T)
md
```

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

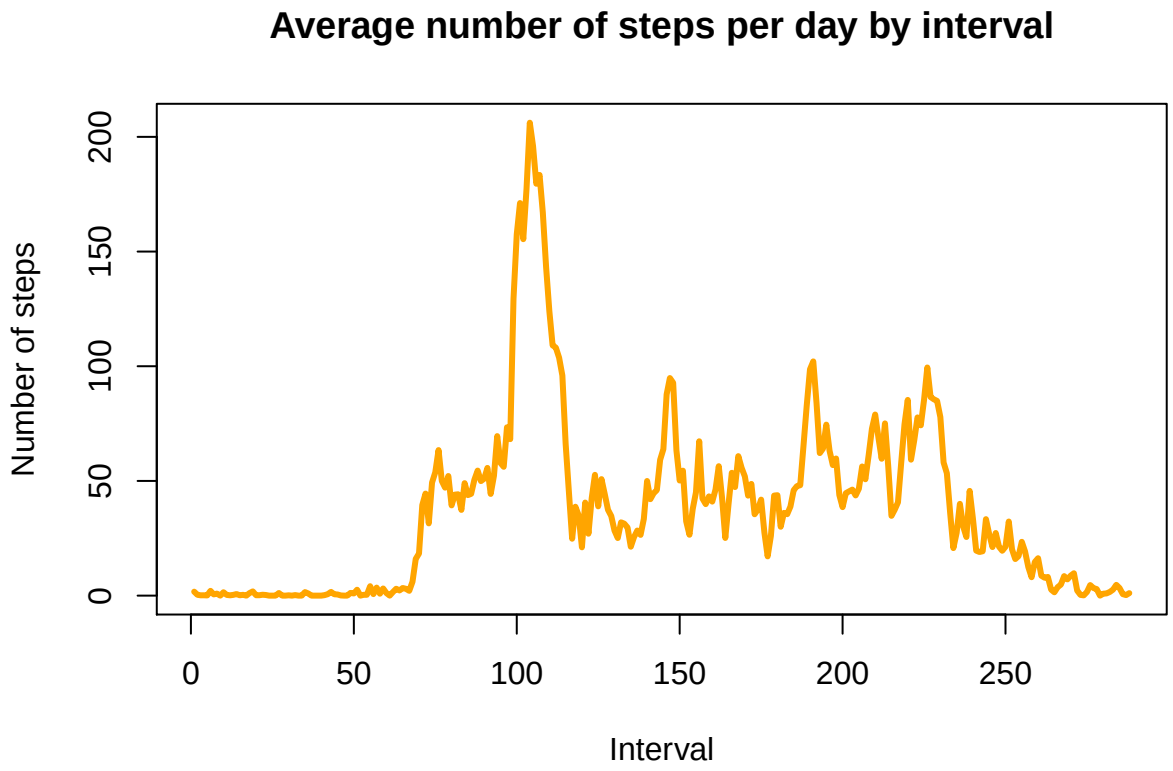
What is the average daily activity pattern?

We compute the average number of steps taken on each interval averaged across all days and save it into the object `stepsByInterval`.

```
stepsByInterval <- tapply(df$steps, df$interval, mean, na.rm =T)
```

And now we make a time series plot

```
plot(stepsByInterval, type="l", xlab="Interval", ylab="Number of steps",
     main="Average number of steps per day by interval", col = "orange", lwd = 3)
```



We find out that the interval which on average across all the days in the dataset contains the maximum number of steps is

```
max_interval <- stepsByInterval[which.max(stepsByInterval)]
max_interval
```

```
##      835
## 206.1698
```

Imputing missing values

The total number of missing values in the dataset is

```
NAnumber <- sum(!complete.cases(df))
NAnumber
```

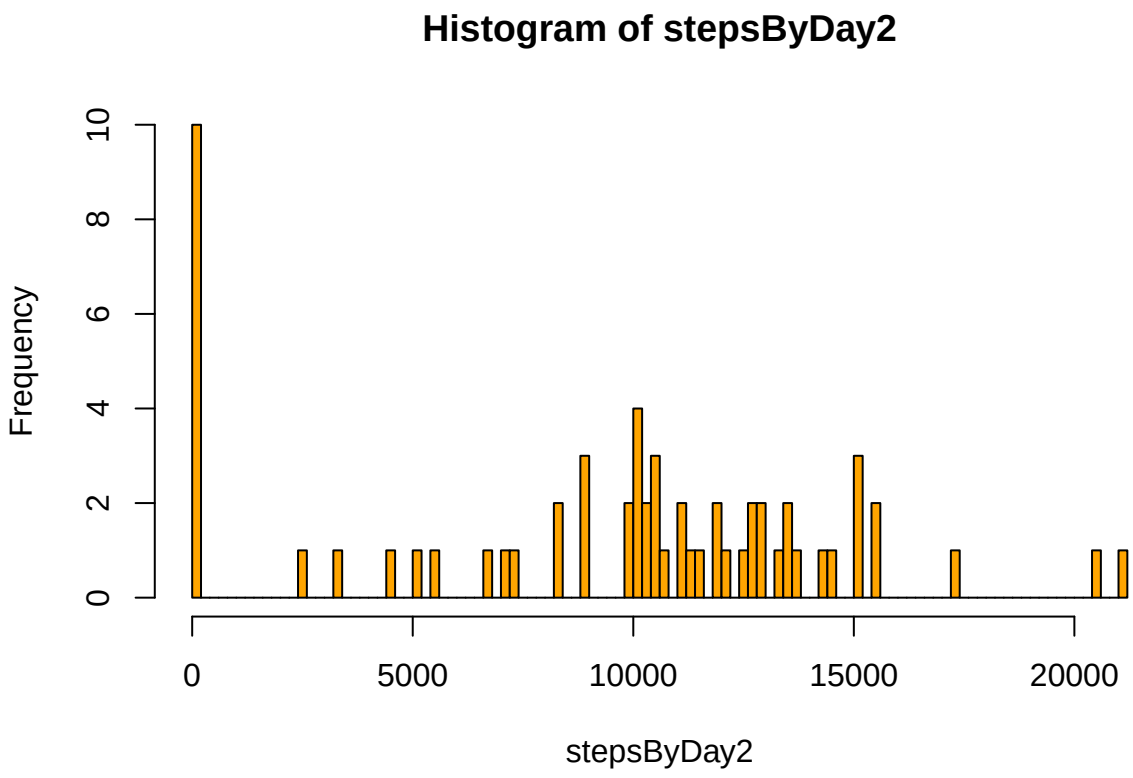
```
## [1] 2304
```

We substitute each missing value for the mean of steps of the interval that the missing value belongs and save it in a new data set named `df2`.

```
df2<-df
for (i in 1:length(df2)){
 if(is.na(df2$steps[i])){ #when we find a missing value
   df2$steps[i] <- mean(df2$steps[df$interval==df2$interval[i]], na.rm = T)
   #we substutite it by
}
}
```

Similarly we draw a plot and compute the mean and median for the new data set

```
stepsByDay2 <- tapply(df2$steps, df2$date, sum, na.rm = T)
hist(stepsByDay2, breaks=100, col = "orange")
```




```
m2 <- mean(stepsByDay2, na.rm = T )
md2 <- mean(stepsByDay2, na.rm = T)
```

How much does the values differ?

```
difm <- abs(m-m2)
difmd <- abs(md-md2)
difm
```

```
## [1] 1411.923
```

```
difmd
```

```
## [1] 1410.735
```

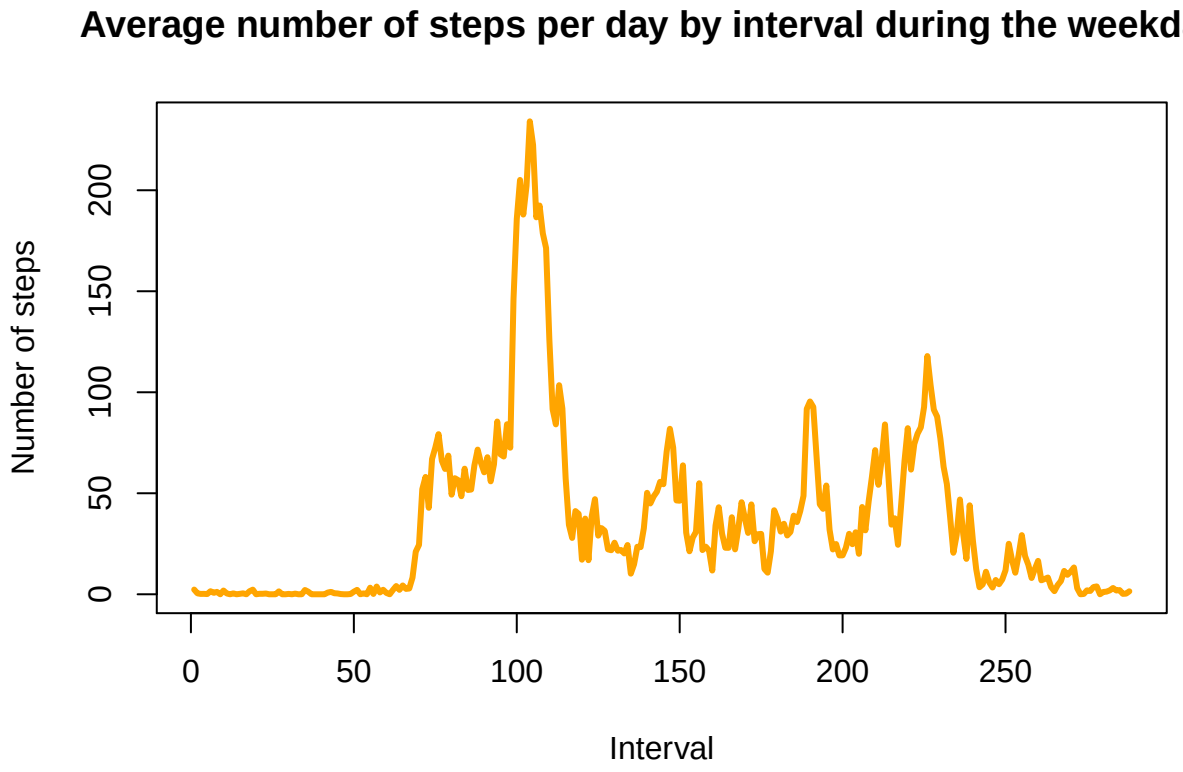
Are there differences in activity patterns between weekdays and weekends?

We create a new factor variable in the dataset named `nu` with two levels – "weekday" and "weekend" indicating whether a given date is a weekday or weekend day.

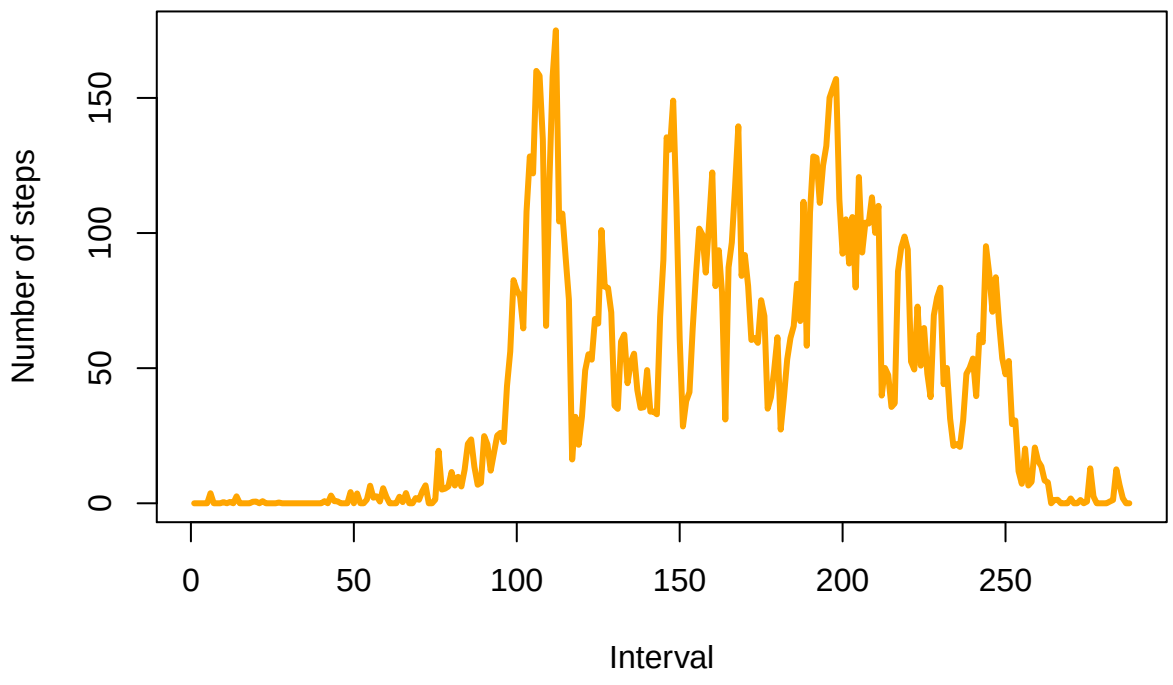
```
weekdays <- c("Monday", "Tuesday", "Wednesday", "Thursday",
              "Friday")
df2$nu = as.factor(ifelse(is.element(weekdays(as.Date(df2$date)),weekdays), "Weekday", "Weekend"))
```

Now, we separate the set into two subsets and plot the results

```
df3 <- df2[df2$nu == "Weekday",]
df4 <-df2[df2$nu == "Weekend",]

stepsByInterval3 <- tapply(df3$steps, df3$interval, mean, na.rm =T)

plot(stepsByInterval3, type="l", xlab="Interval", ylab="Number of steps",
     main="Average number of steps per day by interval during the weekdays", col = "orange", lwd = 3)
```




```
stepsByInterval4 <- tapply(df4$steps, df4$interval, mean, na.rm =T)

plot(stepsByInterval4, type="l", xlab="Interval", ylab="Number of steps",
     main="Average number of steps per day by interval during the weekends", col = "orange", lwd = 3)
```



```r
sessionInfo()
```

```
## R version 3.1.2 (2014-10-31)
## Platform: x86_64-pc-linux-gnu (64-bit)
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## loaded via a namespace (and not attached):
## [1] digest_0.6.4    evaluate_0.5.5  formatR_1.0     htmltools_0.2.6
## [5] knitr_1.7       rmarkdown_0.3.3 stringr_0.6.2   tools_3.1.2
## [9] yaml_2.1.13
```