Reproducible Research Project 1

Konstantinos Papastamos

13 November 2015

Introduction

It is now possible to collect a large amount of data about personal movement using activity monitoring devices such as a Fitbit, Nike Fuelband, or Jawbone Up. These type of devices are part of the "quantified self" movement – a group of enthusiasts who take measurements about themselves regularly to improve their health, to find patterns in their behavior, or because they are tech geeks. But these data remain under-utilized both because the raw data are hard to obtain and there is a lack of statistical methods and software for processing and interpreting the data.

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.The goal is to find out if there are differences in activity patterns between weekdays and weekends?

Loading and preprocessing the data

So, let's start by loading and cleaning the data:

```
Sys.setlocale("LC_ALL","English")
data=read.csv("activity.csv")
data$date = as.Date(data$date, "%Y-%m-%d")
cleaned_data=na.omit(data)
```

Here I also convert the date variable into a date format. I also change the locale to english in order for the weekdays() function to work properly.

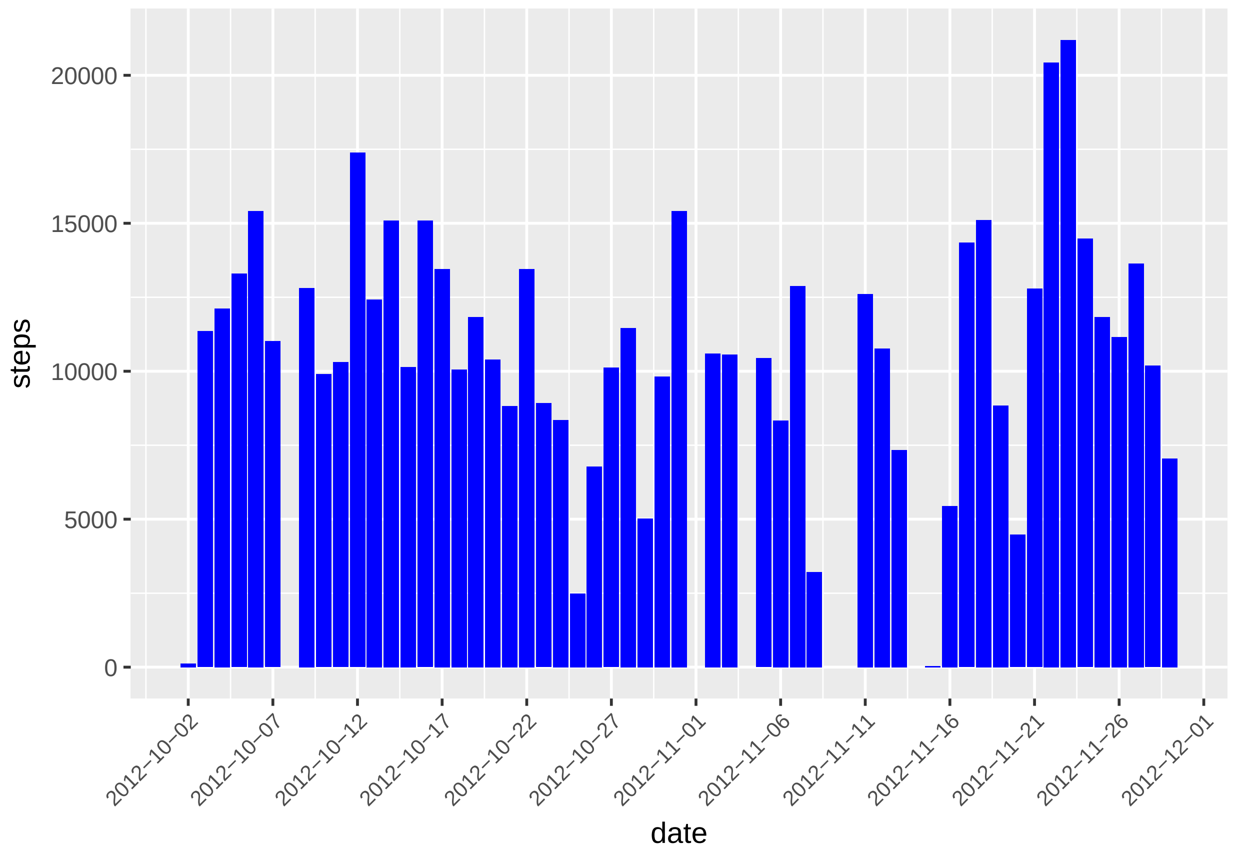
What is mean total number of steps taken per day?

In this section I group the steps variable by the date and find the total number of steps for each day.

```
library(ggplot2)
library(scales)

total_step_num = aggregate(steps ~ date, data=cleaned_data, sum)
total_step_num$date = as.Date(total_step_num$date, "%Y-%m-%d")
total_step_num = total_step_num[order(as.Date(total_step_num$date,format="%d-%m-%Y")),,drop=FALSE]

g = ggplot(total_step_num, aes(y=steps, x=date))
g = g + geom_histogram(stat="identity",fill="blue")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
g = g + theme(axis.text.x = element_text(size=8,angle = 45, vjust=1,hjust=1))
g = g + scale_x_date(date_breaks = "5 days", labels=date_format("%Y-%m-%d"))

g
```



So the total number of steps taken per day, the mean and the median are shown below:

```
total_step_num$date=format(total_step_num$date,"%d-%m-%Y")
head(total_step_num)
```

```
##         date steps
## 1 02-10-2012   126
## 2 03-10-2012 11352
## 3 04-10-2012 12116
## 4 05-10-2012 13294
## 5 06-10-2012 15420
## 6 07-10-2012 11015
```

```
mean(total_step_num$steps)
```

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

```
median(total_step_num$steps)
```

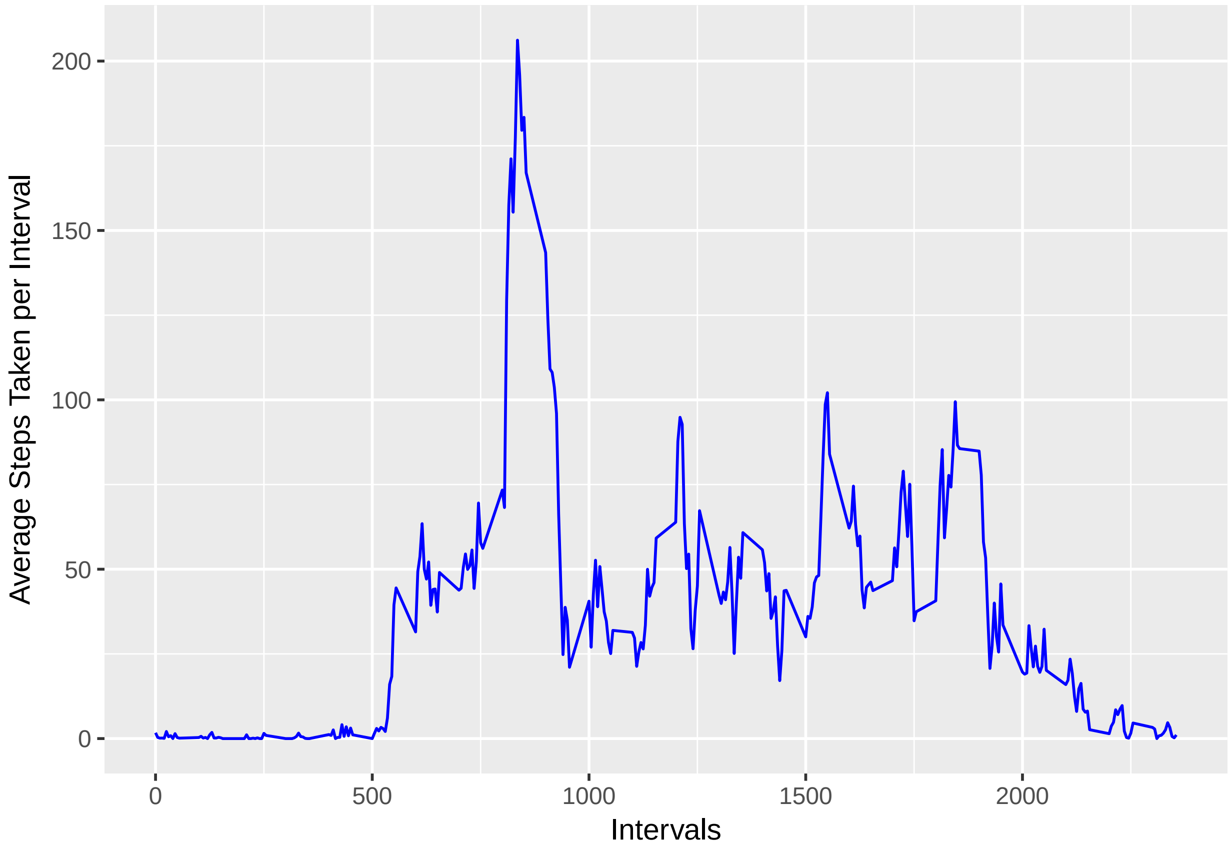
```
## [1] 10765
```

As we can see the mean and the median are pretty close.

What is the average daily activity pattern?

So here we must find the average steps taken per interval across all days.

```
average_steps = aggregate(steps ~ interval,mean,data=cleaned_data)
ggplot(average_steps, aes(y=steps, x=interval)) + geom_line(col="blue") + ylab("Average Steps Taken per
```



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

```
head(average_steps[order(average_steps$steps, decreasing = TRUE),])
```

```
##     interval    steps
## 104      835 206.1698
## 105      840 195.9245
## 107      850 183.3962
## 106      845 179.5660
## 103      830 177.3019
## 101      820 171.1509
```

As we can see, the interval 835 contains the maximum number of steps on average.

Imputing missing values

In this section we calculate the number of rows with missing values

```
missing_values = subset(data,is.na(steps))
nrow(missing_values)
```

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

So the total number of records with missing values is 2304.

In order to fill in all the missing values, I am going to use the mean of each interval.

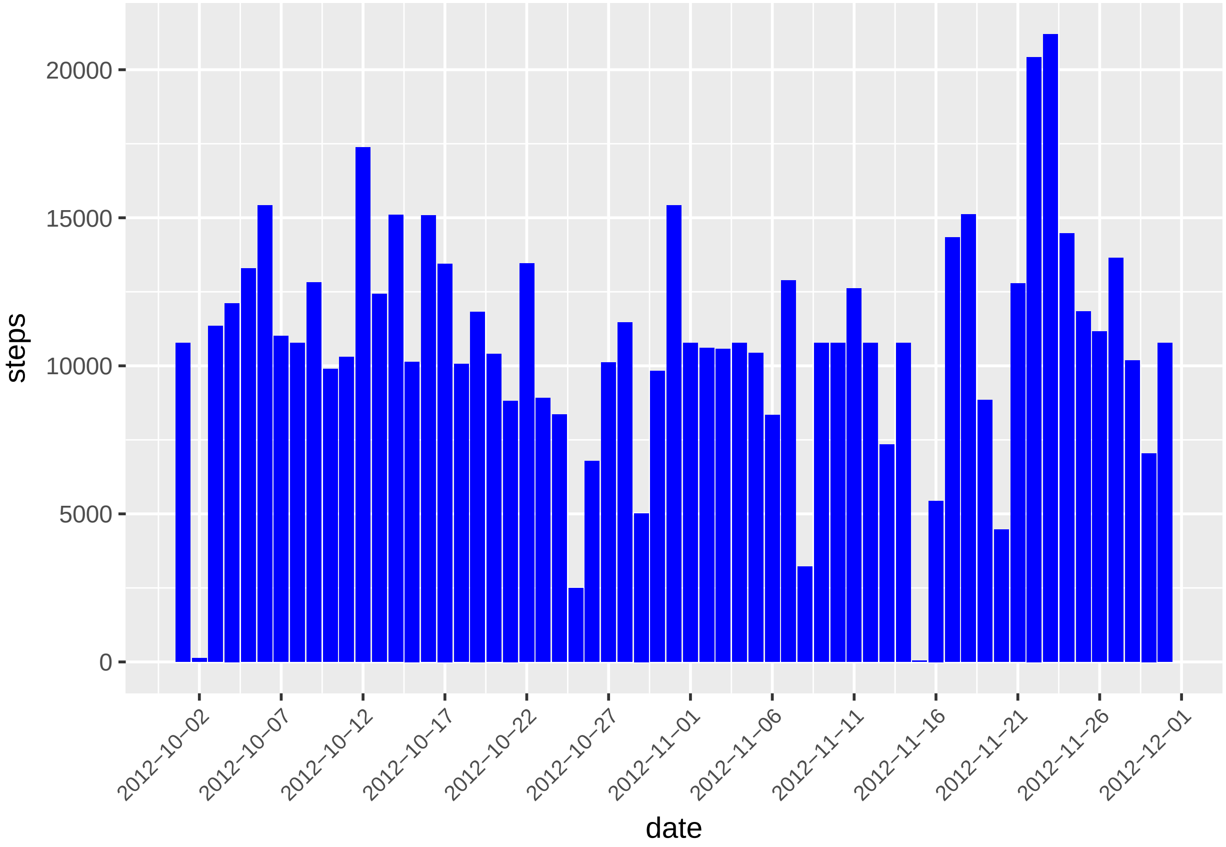
```
averaged_steps = aggregate(steps ~ interval, mean, data = cleaned_data)

filled_data = data
for(i in 1:nrow(data)){

      if(is.na(data[i,1])){
        filled_data[i,1] = subset(averaged_steps,interval==data[i,3])[,2]
      }

}
```

The filled_data dataframe is identical with the original but with the average step number per interval instead of missing values.

So let's see the total number of steps taken each day in the filled data:

```
total_step_num_filled = aggregate(steps ~ date, data=filled_data, sum)
total_step_num_filled$date = as.Date(total_step_num_filled$date, "%Y-%m-%d")
total_step_num_filled = total_step_num_filled[order(as.Date(total_step_num_filled$date,format="%d-%m-%Y
ggplot(total_step_num_filled, aes(y=steps, x=date)) + geom_histogram(stat="identity",fill="blue") + ther
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```



And now let's calculate the filled data mean and median:

```
total_step_num_filled$date=format(total_step_num_filled$date,"%d-%m-%Y")
head(total_step_num_filled)
```

```
##         date    steps
## 1 01-10-2012 10766.19
## 2 02-10-2012   126.00
## 3 03-10-2012 11352.00
## 4 04-10-2012 12116.00
## 5 05-10-2012 13294.00
## 6 06-10-2012 15420.00
```

```
mean(total_step_num_filled$steps)
```

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

```
median(total_step_num_filled$steps)
```

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

In this case we can see that filling up the missing values didn't affect the mean which is expected since we used the average step value to do the job. However there was a small change in the median which now has the same value as the mean.

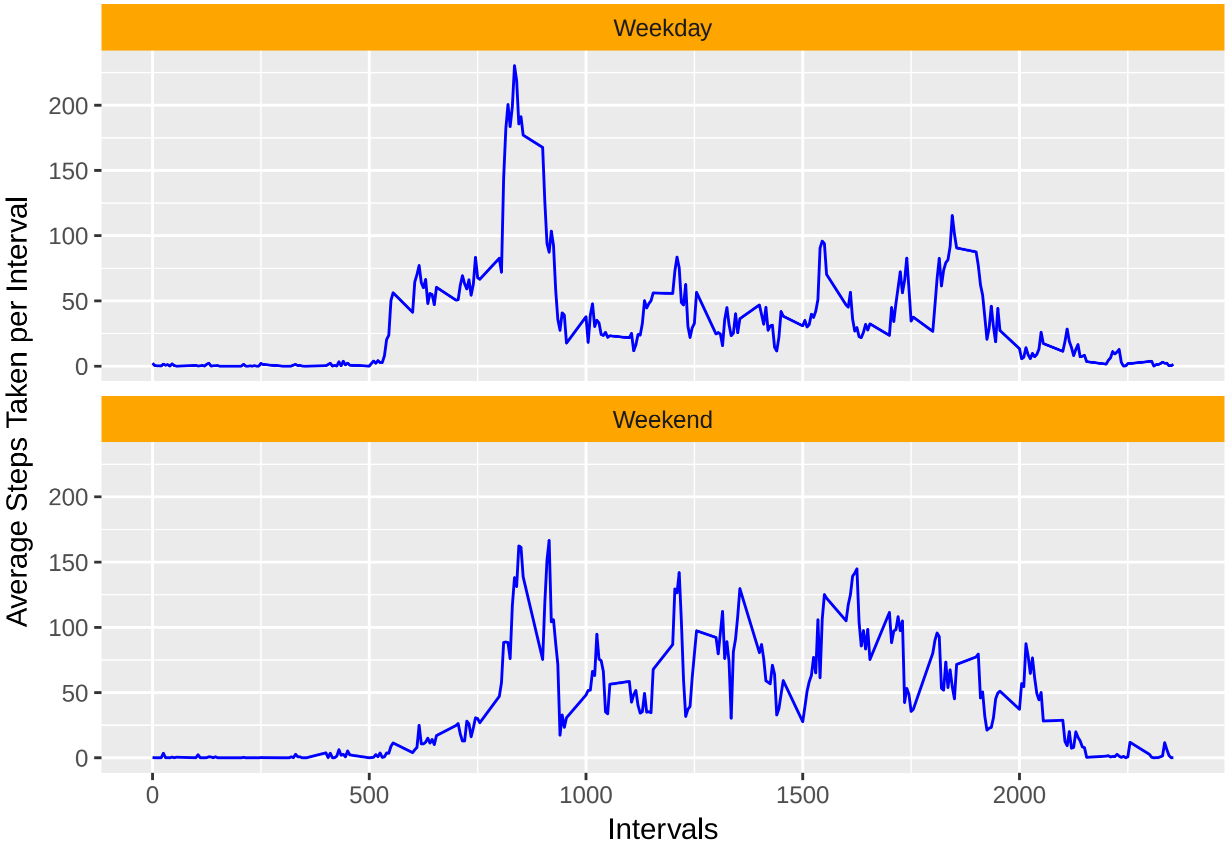
Are there differences in activity patterns between weekdays and weekends?

Here I will create the Weekday variable which will contain the values "Weekday" and "Weekend" indicating of course if the day is a weekday or not.

```
filled_data$weekday="Weekday"
for(i in 1:nrow(filled_data)){


      if(weekdays(filled_data[i,2])=="Saturday"||weekdays(filled_data[i,2])=="Sunday"){
        filled_data[i,4] = "Weekend"
      }

}
filled_data$weekday = as.factor(filled_data$weekday)
```

In this last part we compare the difference in activity between weekdays and weekends:

```
weekday=aggregate(steps ~ interval,mean,data=subset(filled_data,weekday=="Weekday"))
weekend=aggregate(steps ~ interval,mean,data=subset(filled_data,weekday=="Weekend"))
weekday$weekday="Weekday"
weekend$weekday="Weekend"

full_week = rbind(weekday,weekend)
full_week$weekday=as.factor(full_week$weekday)

ggplot(full_week, aes(y=steps, x=interval, fill = weekday)) + geom_line(col="blue") + ylab("Average Ste
```



As we can see, in weekends the activity seems ro be higher between the intervals 1000 and 1500 as well as around the interval 2000 while during weekdays the activity seems more intense during the intervals 750 - 1000.