Project 1

Salvador CastaÃ±eda

19 de febrero de 2016

Data information

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.

Code

reading data

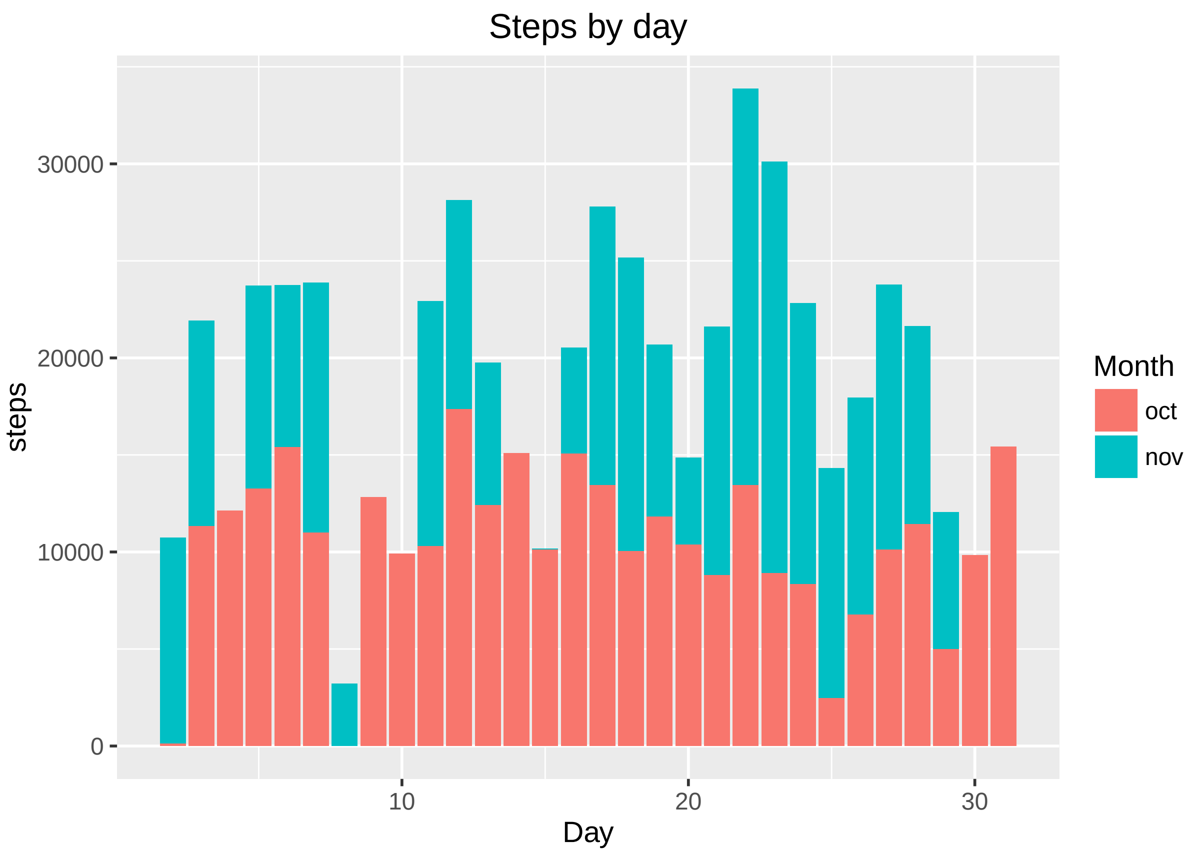
```
activity <- read.csv("activity.csv")
```

Histogram

```
library(ggplot2)
library(lubridate)
```

subsetting so it doesn't have NA's

```
Tot_data <- aggregate(steps ~ date, activity, sum, na.rm = T)


 ggplot(Tot_data, aes(day(date),steps, fill = factor(month(date), labels = c("oct", "nov"))))+
        geom_bar(stat= "identity")+ labs(x = "Day", y = "steps")+
        labs(title = "Steps by day") + scale_fill_discrete(name = "Month")
```



Getting mean and median

```
Mean_data <- aggregate(steps ~ date, activity, mean, na.rm = T)
Median_data <- aggregate(steps ~ date, activity, median, na.rm = T)
mean_and_median <- merge(Mean_data,Median_data, by = "date")
names(mean_and_median)<- c("Date"," Mean", "Median")
head(mean_and_median)
```

```
##         Date     Mean Median
## 1 2012-10-02  0.43750      0
## 2 2012-10-03 39.41667      0
## 3 2012-10-04 42.06944      0
## 4 2012-10-05 46.15972      0
## 5 2012-10-06 53.54167      0
## 6 2012-10-07 38.24653      0
```

we note that all median are 0

```
sum(Median_data$steps)
```

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

Time series

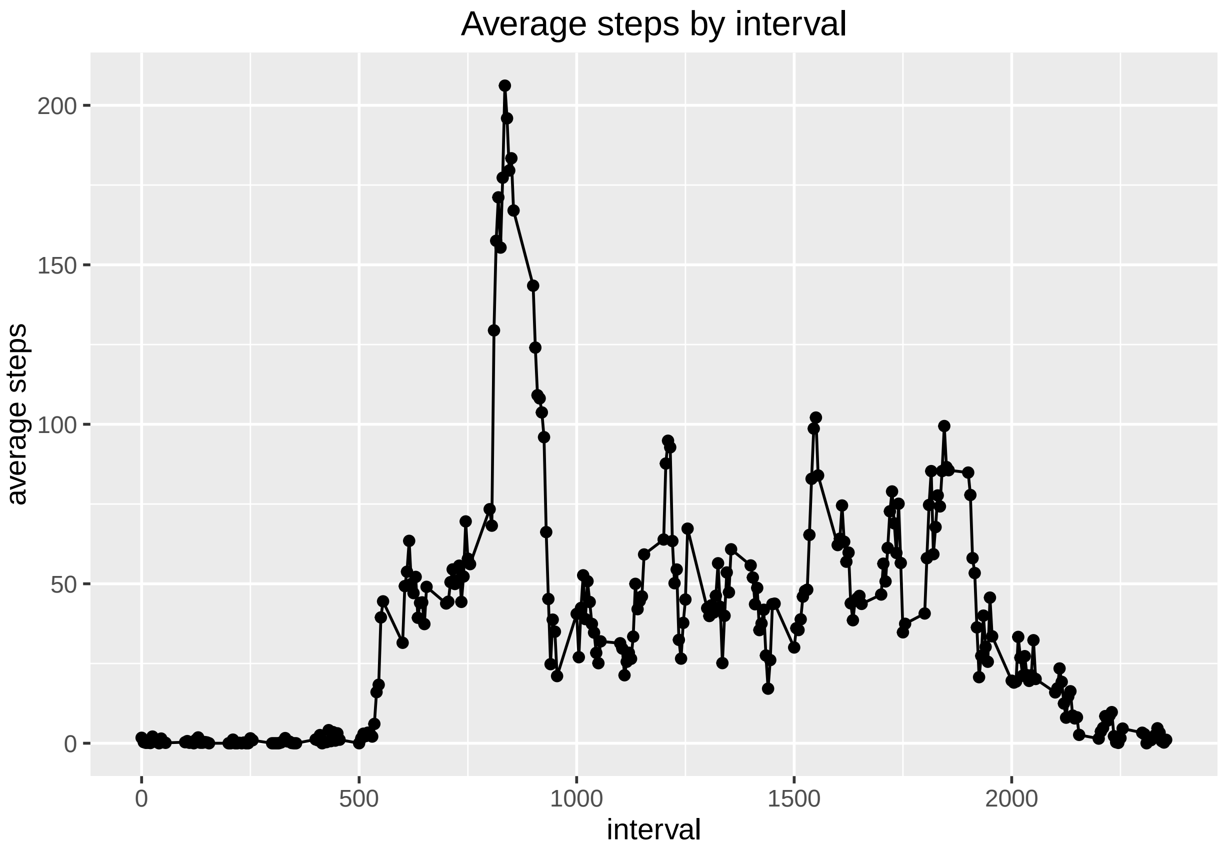
```
mean_interval <- aggregate( steps ~interval,activity, mean)
```

5 minute maximum interval

```
mean_interval[mean_interval$steps == max(mean_interval$steps),]
```

```
##     interval    steps
## 104      835 206.1698
```

```
ggplot(mean_interval, aes( interval, steps ))+
        geom_point(stat = "identity")+geom_line()+labs(y = "average steps")+
        labs(title = "Average steps by interval")
```



Missing Data Strategy

** Fill the original data NA's with their median (i.e. 0) **

```
filled_activity <- activity[,]
filled_activity$steps[is.na(filled_activity$steps)] <- 0
```

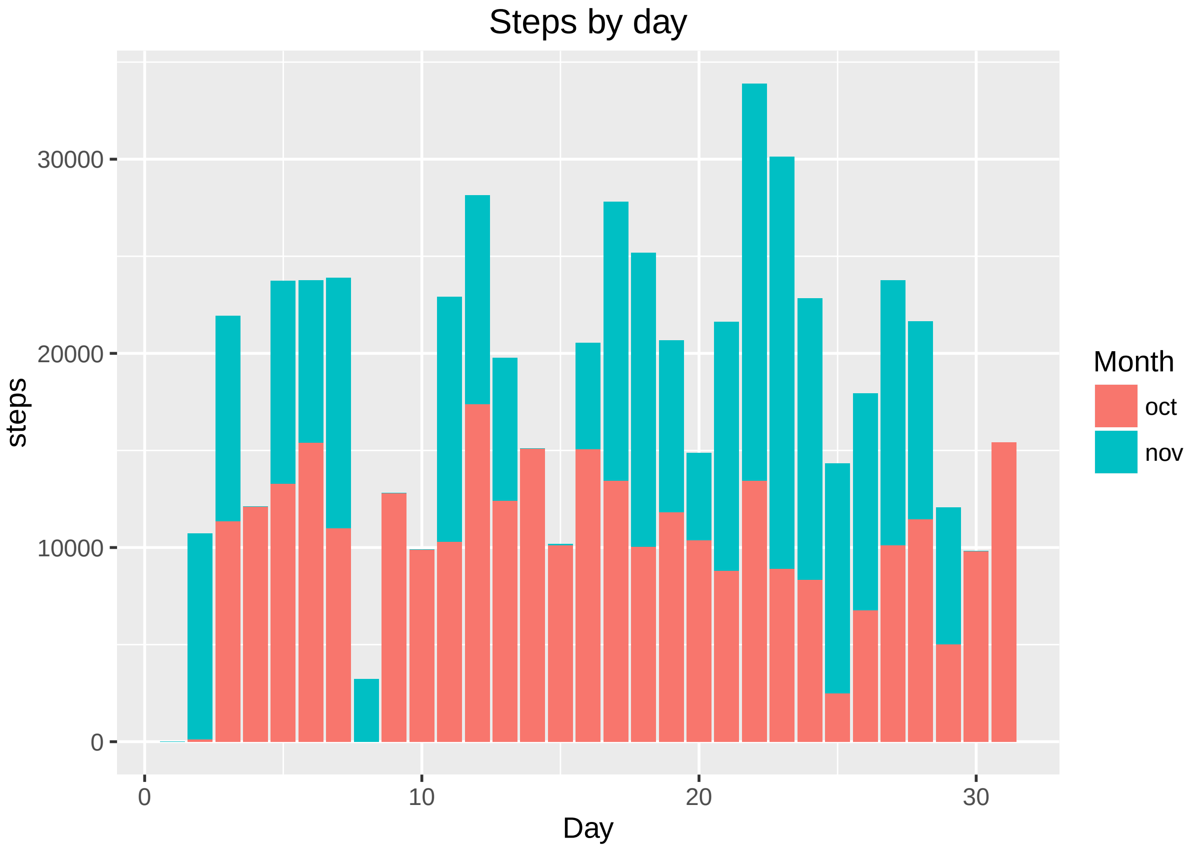
Lets compare

```
mean_filled <- aggregate( steps ~date,filled_activity, mean)
```

2nd Histogram

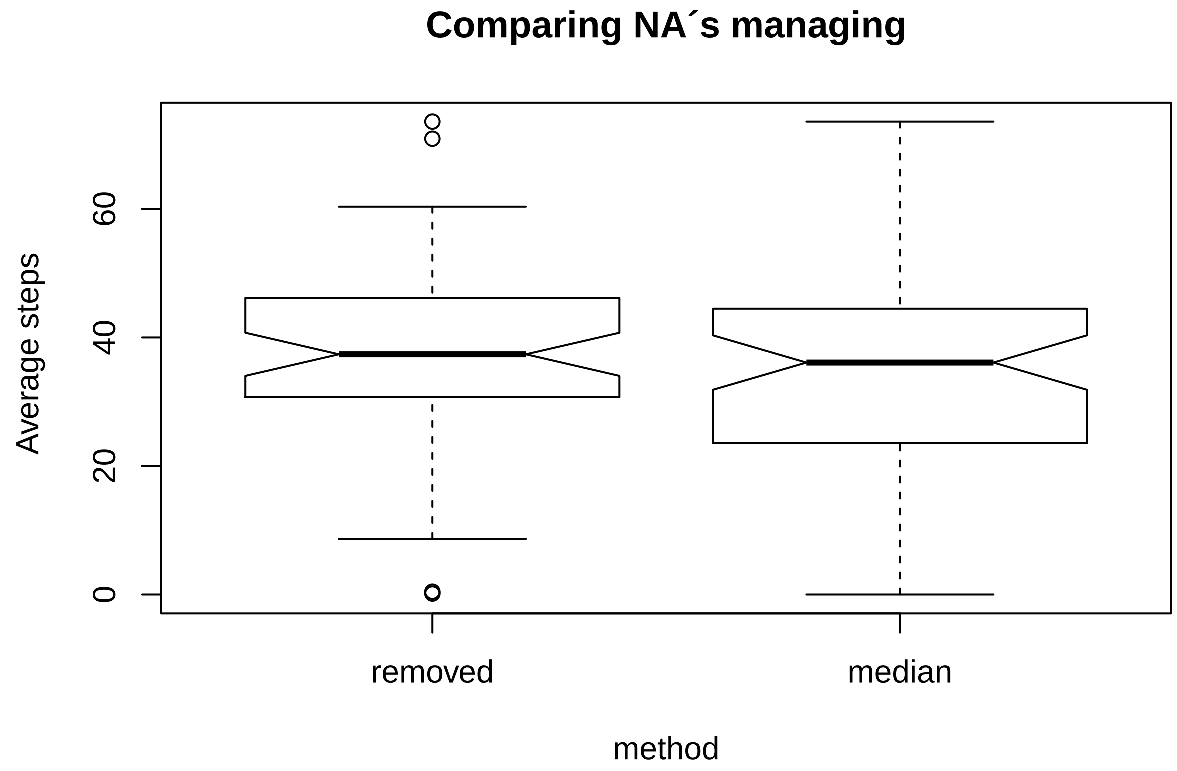
** The histogram doesn´t seem to have changed **

```
ggplot(filled_activity, aes(day(date),steps, fill = factor(month(date), labels = c("oct", "nov"))))+
        geom_bar(stat= "identity")+ labs(x = "Day", y = "steps")+
        labs(title = "Steps by day") + scale_fill_discrete(name = "Month")
```



** More comparing**

```
boxplot(Mean_data$steps, mean_filled$steps, main = "Comparing NA´s managing",
        notch = T, names = c("removed", "median"),
        xlab = "method", ylab = " Average steps")
```



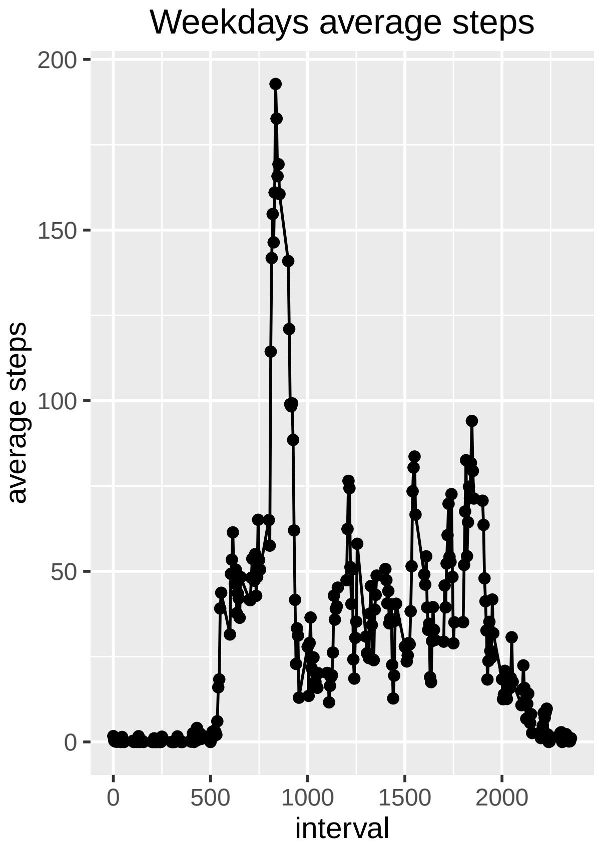
creating a variable factor that says if the day is weekday or weekend

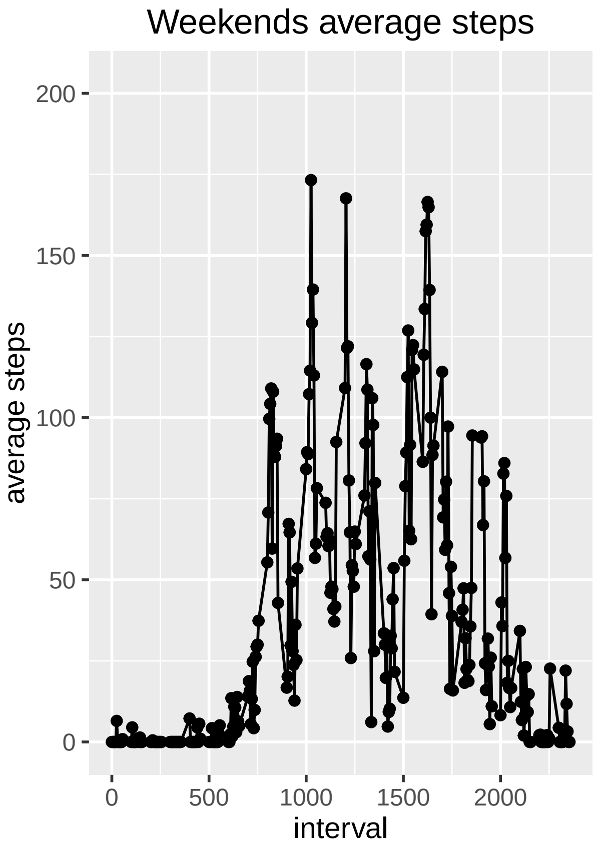
```
filled_activity$day <- weekdays(as.Date(filled_activity$date), abbreviate = T)

filled_activity$day <- ifelse(filled_activity$day == "sÃ¡b."  |
                                      filled_activity$day == "dom.", "weekend", "weekday")
wend <- filled_activity[filled_activity$day == "weekend",]
wday <- filled_activity[filled_activity$day != "weekend",]
wend_mean <- aggregate(steps~interval , wend, mean)
wday_mean <- aggregate(steps~interval , wday, mean)
```

Panel by weekend/weekday

** Creating time series by weekdays**

```
library(gridExtra)

g <- ggplot(wend_mean, aes(interval, steps ))+ geom_point(stat = "identity")+
        geom_line()+labs(y = "average steps")+
        labs(title = "Weekends average steps")+
        coord_cartesian(ylim = c(0,202.8889))
k <- ggplot(wday_mean, aes(interval, steps ))+ geom_point(stat = "identity")+
        geom_line()+labs(y = "average steps")+
        labs(title = "Weekdays average steps")
grid.arrange(k,g, ncol=2)
```


Weekdays average steps
average steps
200
150
100
50
0
0
500
1000
1500
2000
interval


Weekends average steps
average steps
200
150
100
50
0
0
500
1000
1500
2000
interval