DATA SCIENCE

Practical Work 3: Data Description by Group. Visualization.

M.José Ramírez-Quintana MITSS
Universitat Politècnica de València

In the previous practical works we have seen that R provides a wide range of functions for obtaining summary statistics. In particular we have used `sapply()` to calculate the mean, the minimum, the maximum,... Additionally, some summary statistics can also be generated by firstly grouping variables (as we have seen in the file "Data Preparation and Visualization: A study case" through the use of `group_by` followed by `summarise`). There are other three main ways to group data based on some specified variables, and apply a summary function (like mean, standard deviation, etc.) to each group:

- The `ddply()` function. It is the easiest to use, though it requires the `plyr` package. This is probably what you want to use.
- The `summaryBy()` function. It is easier to use, though it requires the `doBy` package.
- The `aggregate()` function. It is more difficult to use but is included in the base install of R.

Here, we are going to work with the `summaryBy` function. The basic use of this function is

```
summaryBy(A ~ B, data=data1, FUN=fun1)
```

which calculates the summarise function `fun1` over column `A` by grouping `data1` by column `B`. We use the symbol + to summarize data by more than one column (for instance, `A+C`) and/or for grouping data by more than one column (for instance, `B+D`). The parameter `FUN` can also be a list of functions each of which returns a single value, for instance `FUN=list(mean, var)`.

In order to complete the exercise we will use the following libraries:

- `library("doBy")`
- `library("ggplot2")`

You can find the file "nyt1.csv" that contains the data of one (simulated) day's worth of ads shown and clicks recorded on the New York Times home page in May 2012 at the Course Documentation tab. Each row represents a single user. There are five columns: age, gender (0=female, 1=male), number impressions, number clicks, and loggedin.

1. Read the file "nyt1.csv" and assign it to a variable "mydata". Inspect the data (using the `summary()` function).

2. Create a new variable "age_group" that categorizes users by age as "<20","20-29","30-39","40-49","50-59",">60".

3. Calculate the maximum and minimum values and the mean of attributes age, number of impressions and number of clicks for each age interval and for each gender (use the `summaryBy` function).

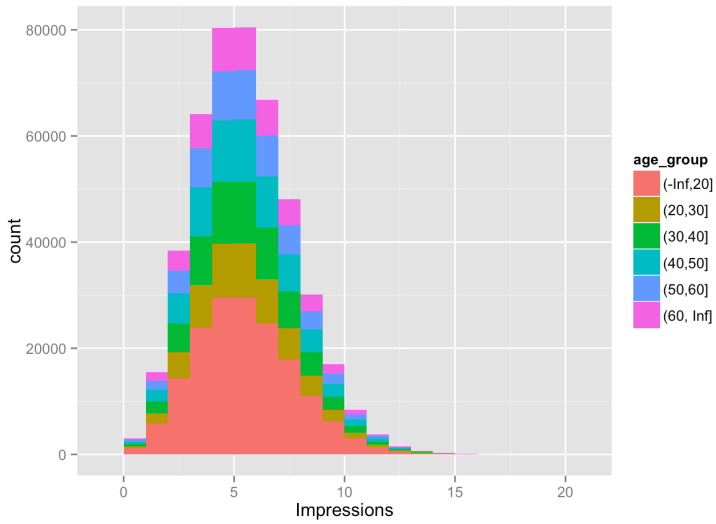
4. Build a histogram showing the total number of impresions for each age interval.

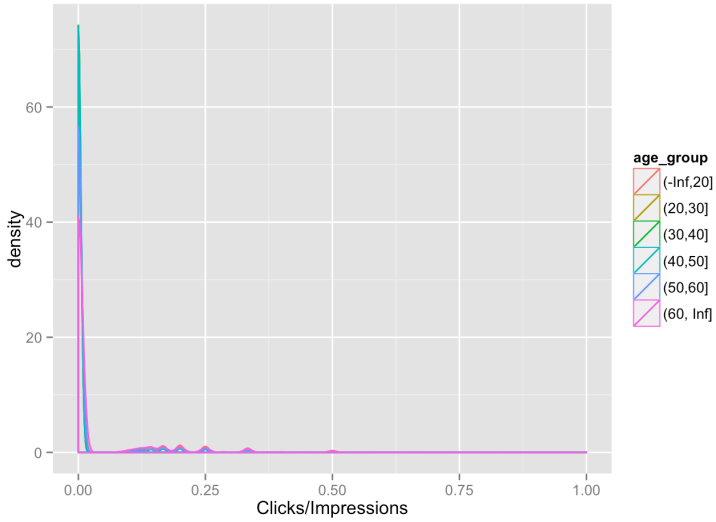
5. Graphically show the distribution of the ratio clicks/impressions for each age interval and the distribution of clicks for each age group.

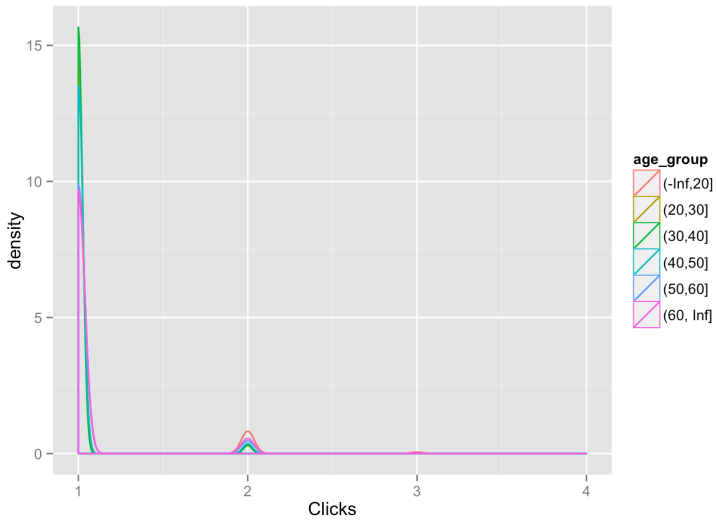
The result of the two last exercises should look like as what is shown on the following page.

REFERENCES

* Cathy O'Neil, Rachel Schutt. Doing Data Science (Chapter 2). O'Reilly Media. 2013.


80000
60000
40000
20000
0
count
0
5
10
15
20
Impressions
age_group
(-Inf,20]
(20,30]
(30,40]
(40,50]
(50,60]
(60, Inf]


60
40
20
0
density
0.00
0.25
0.50
0.75
1.00
Clicks/Impressions
age_group
(-Inf,20]
(20,30]
(30,40]
(40,50]
(50,60]
(60, Inf]


15
10
5
0
density
1
2
3
4
Clicks
age_group
(-Inf,20]
(20,30]
(30,40]
(40,50]
(50,60]
(60, Inf]