Simple analysis for Twitter Data

Web Monitoring Group

11 October 2017

# Analyse Twitter data

We have a dataset of 500 tweets previously obtained using the *twitteR* package and by searching for the word "barcelona". We will first do look at the first few observations in the dataset.

tweets <- readRDS("../data/barcelona-tweets.rds")  
head(tweets)

## text  
## 1 ¡Academia Japonia tendrá stand en el Salón del Manga de Barcelona! Tendremos mucha información para vosotros. <U+FFFD><U+FFFD> https://t.co/OhS7QUhw5q  
## 2 Just watched @Xavi\_Bove's webinar replay for @SBI\_Barcelona very interesting presentation and a lot to learn from Thank you.@DiegoValdesBCN  
## 3 Barcelona chief executive talks up January bid for Philippe Coutinho https://t.co/w16vomIB4g https://t.co/3Jd5G0fmyp  
## 4 RT @BrujulaOndaCero: <U+FFFD><U+FFFD>#ÚLTIMAHORA El Grupo Planeta confirma que traslada su sede social de Barcelona a Madrid tras la intervención de Carl  
## 5 <U+FFFD><U+FFFD><U+FFFD><U+FFFD><U+FFFD><U+FFFD><U+FFFD><U+FFFD> @asweholdhands #pasalubong#Sushi#instagood#instafood#foodislife#Foodie @ Barcelona, Spain https://t.co/ySkNWVcmdL  
## 6 RT @footballdailyuk: Anthony Martial Wants To Leave Manchester United For Barcelona?! 1 person who RT's will get a follow back! WATCH: h  
## favorited favoriteCount replyToSN created truncated  
## 1 FALSE 0 <NA> 2017-10-11 15:09:20 TRUE  
## 2 FALSE 0 <NA> 2017-10-11 15:09:19 FALSE  
## 3 FALSE 0 <NA> 2017-10-11 15:09:19 FALSE  
## 4 FALSE 0 <NA> 2017-10-11 15:09:19 FALSE  
## 5 FALSE 0 <NA> 2017-10-11 15:09:18 FALSE  
## 6 FALSE 0 <NA> 2017-10-11 15:09:18 FALSE  
## replyToSID id replyToUID  
## 1 <NA> 918131407398096896 <NA>  
## 2 <NA> 918131405678436352 <NA>  
## 3 <NA> 918131405271351296 <NA>  
## 4 <NA> 918131403086327809 <NA>  
## 5 <NA> 918131400502423554 <NA>  
## 6 <NA> 918131399345000448 <NA>  
## statusSource  
## 1 <a href="http://twitter.com" rel="nofollow">Twitter Web Client</a>  
## 2 <a href="http://twitter.com" rel="nofollow">Twitter Web Client</a>  
## 3 <a href="https://dlvrit.com/" rel="nofollow">dlvr.it</a>  
## 4 <a href="http://twitter.com/download/android" rel="nofollow">Twitter for Android</a>  
## 5 <a href="http://instagram.com" rel="nofollow">Instagram</a>  
## 6 <a href="http://twitter.com/download/iphone" rel="nofollow">Twitter for iPhone</a>  
## screenName retweetCount isRetweet retweeted longitude latitude  
## 1 academiajaponia 0 FALSE FALSE <NA> <NA>  
## 2 HaimovichDavid 0 FALSE FALSE <NA> <NA>  
## 3 LFC\_SC 0 FALSE FALSE <NA> <NA>  
## 4 oscarlena1992 1134 TRUE FALSE <NA> <NA>  
## 5 stefani\_luz 0 FALSE FALSE 2.18333 41.3833  
## 6 iamandrescano 18 TRUE FALSE <NA> <NA>

We will routinely carry out some other operations just to have an overview or get a feel of what the dataset is like.

What are the dimensions of our data frame?

dim(tweets)

## [1] 500 16

There are 500 rows and 16 variables.

What are the names of our variables? We can list them so

colnames(tweets)

## [1] "text" "favorited" "favoriteCount" "replyToSN"   
## [5] "created" "truncated" "replyToSID" "id"   
## [9] "replyToUID" "statusSource" "screenName" "retweetCount"   
## [13] "isRetweet" "retweeted" "longitude" "latitude"

Summary output of each variable

summary(tweets)

## text favorited favoriteCount replyToSN   
## Length:500 Mode :logical Min. : 0.000 Length:500   
## Class :character FALSE:500 1st Qu.: 0.000 Class :character   
## Mode :character Median : 0.000 Mode :character   
## Mean : 0.094   
## 3rd Qu.: 0.000   
## Max. :18.000   
## created truncated replyToSID   
## Min. :2017-10-11 15:04:27 Mode :logical Length:500   
## 1st Qu.:2017-10-11 15:05:24 FALSE:480 Class :character   
## Median :2017-10-11 15:06:46 TRUE :20 Mode :character   
## Mean :2017-10-11 15:06:48   
## 3rd Qu.:2017-10-11 15:08:07   
## Max. :2017-10-11 15:09:20   
## id replyToUID statusSource   
## Length:500 Length:500 Length:500   
## Class :character Class :character Class :character   
## Mode :character Mode :character Mode :character   
##   
##   
##   
## screenName retweetCount isRetweet retweeted   
## Length:500 Min. : 0.0 Mode :logical Mode :logical   
## Class :character 1st Qu.: 0.0 FALSE:155 FALSE:500   
## Mode :character Median : 11.0 TRUE :345   
## Mean : 154.2   
## 3rd Qu.: 120.0   
## Max. :8596.0   
## longitude latitude   
## Length:500 Length:500   
## Class :character Class :character   
## Mode :character Mode :character   
##   
##   
##

This kind of summary can show some interesting results. For instance, looking at the variable retweetCount, wouldn't we love to know which tweet attracted 8,596 retweets? This can be easily done with R, but we won't go into that here...

Structure of the object i.e. the data frame

str(tweets)

## 'data.frame': 500 obs. of 16 variables:  
## $ text : chr "¡Academia Japonia tendrá stand en el Salón del Manga de Barcelona! Tendremos mucha información para vosotros. <"| \_\_truncated\_\_ "Just watched @Xavi\_Bove's webinar replay for @SBI\_Barcelona very interesting presentation and a lot to learn fr"| \_\_truncated\_\_ "Barcelona chief executive talks up January bid for Philippe Coutinho https://t.co/w16vomIB4g https://t.co/3Jd5G0fmyp" "RT @BrujulaOndaCero: <U+FFFD><U+FFFD>#ÚLTIMAHORA El Grupo Planeta confirma que traslada su sede social de Barc"| \_\_truncated\_\_ ...  
## $ favorited : logi FALSE FALSE FALSE FALSE FALSE FALSE ...  
## $ favoriteCount: num 0 0 0 0 0 0 0 0 0 0 ...  
## $ replyToSN : chr NA NA NA NA ...  
## $ created : POSIXct, format: "2017-10-11 15:09:20" "2017-10-11 15:09:19" ...  
## $ truncated : logi TRUE FALSE FALSE FALSE FALSE FALSE ...  
## $ replyToSID : chr NA NA NA NA ...  
## $ id : chr "918131407398096896" "918131405678436352" "918131405271351296" "918131403086327809" ...  
## $ replyToUID : chr NA NA NA NA ...  
## $ statusSource : chr "<a href=\"http://twitter.com\" rel=\"nofollow\">Twitter Web Client</a>" "<a href=\"http://twitter.com\" rel=\"nofollow\">Twitter Web Client</a>" "<a href=\"https://dlvrit.com/\" rel=\"nofollow\">dlvr.it</a>" "<a href=\"http://twitter.com/download/android\" rel=\"nofollow\">Twitter for Android</a>" ...  
## $ screenName : chr "academiajaponia" "HaimovichDavid" "LFC\_SC" "oscarlena1992" ...  
## $ retweetCount : num 0 0 0 1134 0 ...  
## $ isRetweet : logi FALSE FALSE FALSE TRUE FALSE TRUE ...  
## $ retweeted : logi FALSE FALSE FALSE FALSE FALSE FALSE ...  
## $ longitude : chr NA NA NA NA ...  
## $ latitude : chr NA NA NA NA ...

## Drawing a graph

Although there are many native functions in R for drawing plots, We are going to be using a popular package known as *ggplot2* to draw our plots. We will learn about base R graphics another time.

library(ggplot2)

First, for the purpose of demonstration, we will load the dataset into a ggplot object, which will ultimately be used for plotting graphs. We will ask R to run this object (here called gg). When drawing a plot, we wouldn't normally run this step but it is being done here just to show what's going on.

gg <- ggplot(tweets)  
gg



All we see is a blank area, which means nothing. But we now have a bonafide **ggplot** object, which is the foundation of other features that we will be adding subsequently.

Let's now examine this object a bit for the sake of curiosity and better understanding

typeof(gg)

## [1] "list"

class(gg)

## [1] "gg" "ggplot"

The class of an object is what makes object-oriented programming possible in R and developers can assign peculiar behaviours to an object. This is the secret behind the *ggplot2* package.

This object gg contains a whole lot of information including our dataset, layers, scales, mapping, theme, coordinates, etc.

str(gg)

## List of 9  
## $ data :'data.frame': 500 obs. of 16 variables:  
## ..$ text : chr [1:500] "¡Academia Japonia tendrá stand en el Salón del Manga de Barcelona! Tendremos mucha información para vosotros. <"| \_\_truncated\_\_ "Just watched @Xavi\_Bove's webinar replay for @SBI\_Barcelona very interesting presentation and a lot to learn fr"| \_\_truncated\_\_ "Barcelona chief executive talks up January bid for Philippe Coutinho https://t.co/w16vomIB4g https://t.co/3Jd5G0fmyp" "RT @BrujulaOndaCero: <U+FFFD><U+FFFD>#ÚLTIMAHORA El Grupo Planeta confirma que traslada su sede social de Barc"| \_\_truncated\_\_ ...  
## ..$ favorited : logi [1:500] FALSE FALSE FALSE FALSE FALSE FALSE ...  
## ..$ favoriteCount: num [1:500] 0 0 0 0 0 0 0 0 0 0 ...  
## ..$ replyToSN : chr [1:500] NA NA NA NA ...  
## ..$ created : POSIXct[1:500], format: "2017-10-11 15:09:20" ...  
## ..$ truncated : logi [1:500] TRUE FALSE FALSE FALSE FALSE FALSE ...  
## ..$ replyToSID : chr [1:500] NA NA NA NA ...  
## ..$ id : chr [1:500] "918131407398096896" "918131405678436352" "918131405271351296" "918131403086327809" ...  
## ..$ replyToUID : chr [1:500] NA NA NA NA ...  
## ..$ statusSource : chr [1:500] "<a href=\"http://twitter.com\" rel=\"nofollow\">Twitter Web Client</a>" "<a href=\"http://twitter.com\" rel=\"nofollow\">Twitter Web Client</a>" "<a href=\"https://dlvrit.com/\" rel=\"nofollow\">dlvr.it</a>" "<a href=\"http://twitter.com/download/android\" rel=\"nofollow\">Twitter for Android</a>" ...  
## ..$ screenName : chr [1:500] "academiajaponia" "HaimovichDavid" "LFC\_SC" "oscarlena1992" ...  
## ..$ retweetCount : num [1:500] 0 0 0 1134 0 ...  
## ..$ isRetweet : logi [1:500] FALSE FALSE FALSE TRUE FALSE TRUE ...  
## ..$ retweeted : logi [1:500] FALSE FALSE FALSE FALSE FALSE FALSE ...  
## ..$ longitude : chr [1:500] NA NA NA NA ...  
## ..$ latitude : chr [1:500] NA NA NA NA ...  
## $ layers : list()  
## $ scales :Classes 'ScalesList', 'ggproto' <ggproto object: Class ScalesList>  
## add: function  
## clone: function  
## find: function  
## get\_scales: function  
## has\_scale: function  
## input: function  
## n: function  
## non\_position\_scales: function  
## scales: NULL  
## super: <ggproto object: Class ScalesList>   
## $ mapping : list()  
## $ theme : list()  
## $ coordinates:Classes 'CoordCartesian', 'Coord', 'ggproto' <ggproto object: Class CoordCartesian, Coord>  
## aspect: function  
## distance: function  
## expand: TRUE  
## is\_linear: function  
## labels: function  
## limits: list  
## range: function  
## render\_axis\_h: function  
## render\_axis\_v: function  
## render\_bg: function  
## render\_fg: function  
## train: function  
## transform: function  
## super: <ggproto object: Class CoordCartesian, Coord>   
## $ facet :Classes 'FacetNull', 'Facet', 'ggproto' <ggproto object: Class FacetNull, Facet>  
## compute\_layout: function  
## draw\_back: function  
## draw\_front: function  
## draw\_labels: function  
## draw\_panels: function  
## finish\_data: function  
## init\_scales: function  
## map: function  
## map\_data: function  
## params: list  
## render\_back: function  
## render\_front: function  
## render\_panels: function  
## setup\_data: function  
## setup\_params: function  
## shrink: TRUE  
## train: function  
## train\_positions: function  
## train\_scales: function  
## vars: function  
## super: <ggproto object: Class FacetNull, Facet>   
## $ plot\_env :<environment: R\_GlobalEnv>   
## $ labels : list()  
## - attr(\*, "class")= chr [1:2] "gg" "ggplot"

Now we're going to introduce the aesthetics function aes(), which is necessary to map variables from the dataset to parts of the plotting area.

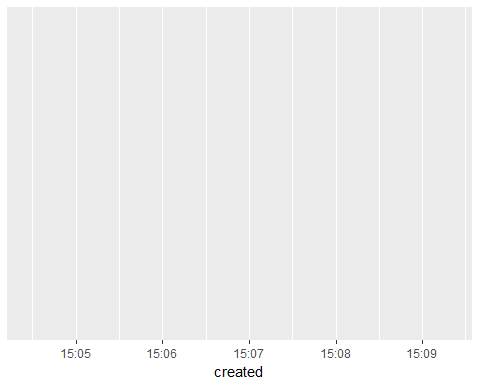
To quickly demonstrate what this function does, check this out

aes(created)

## \* x -> created

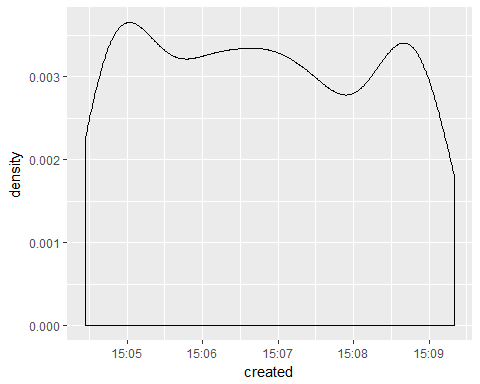
The above output shows that we are mapping our *created* variable to the x-axis of our plot, as can be seen below...

gg <- ggplot(tweets, aes(created))  
gg



We will now draw a simple graph of our tweets based on when they were created, precisely a density plot using geom\_density()

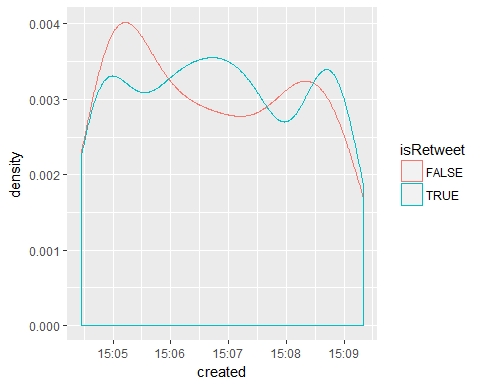
ggplot(tweets, aes(created)) +   
 geom\_density()



The + sign tells us that we are adding this on top of our plotting area as a layer. In the *ggplot2* package, you can add various elements of a plot one on top of another in this way, but we will not go into that here.

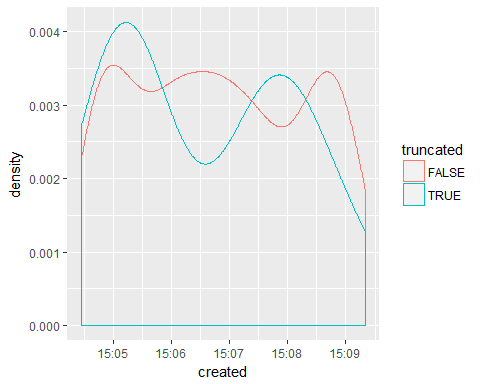
Now, we would like to explore the data in a multivariate fashion. We can check to see how different categories of tweets are distributed e.g. whether they are retweets or not (comparing created with isRetweet variable). We will do this by mapping another aesthetic, not to an axis, but to a colour scheme - by applying this to the function geom\_density() that created the curve in the first place.

gg <- ggplot(tweets, aes(created)) +   
 geom\_density(aes(col = isRetweet))  
gg



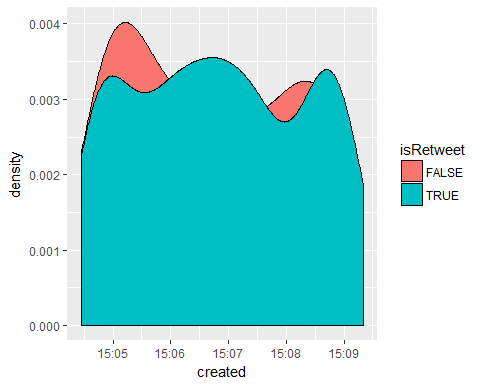
Let's try out the same thing with another categorical variable truncated. Hmm, what's in those extra long tweets?

gg <- ggplot(tweets, aes(created)) +  
 geom\_density(aes(col = truncated))  
gg



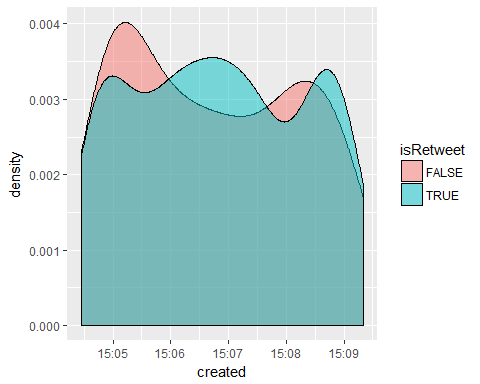
We go back to disaggregating the tweets along the lines of whether or not they were retweets, and this time we want to shade the entire area under the curve by using a fill parameter instead.

gg <- ggplot(tweets, aes(created)) +  
 geom\_density(aes(fill = isRetweet))  
gg



Now let's introduce some transparency so that we can better visualise this using the **alpha** parameter.

gg <- ggplot(tweets, aes(created)) +   
 geom\_density(aes(fill = isRetweet), alpha = 0.5)  
gg



Observe we added little pieces of code that make striking changes to the plot.

To recap on ggplot: 1. Apply the dataset to ggplot() 2. Use at least one aes() function to determine the axes 3. Add other layers to 1. and 2. above as your skill permits.

To be continued...