```
### Install the needed packages...
# install.packages("twitteR")
#install.packages("ROAuth")
# install.packages("rtweet")
#library(arules)
#library(rtweet)
knitr::opts_chunk$set(echo = TRUE)
library(twitteR)
```

```
## Warning: package 'twitteR' was built under R version 3.5.3
```

```
library(ROAuth)
```

```
## Warning: package 'ROAuth' was built under R version 3.5.3
```

```
library(jsonlite)
library(rjson)
```

```
##
## Attaching package: 'rjson'

## The following objects are masked from 'package:jsonlite':
##
##     fromJSON, toJSON
```

```
library(tokenizers)
```

```
## Warning: package 'tokenizers' was built under R version 3.5.3
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 3.5.3

## -- Attaching packages ---------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.1
## v tidyr   0.8.3     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## Warning: package 'ggplot2' was built under R version 3.5.3

## Warning: package 'tibble' was built under R version 3.5.3

## Warning: package 'tidyr' was built under R version 3.5.3

## Warning: package 'purrr' was built under R version 3.5.3
```

```
## Warning: package 'dplyr' was built under R version 3.5.3

## Warning: package 'stringr' was built under R version 3.5.3

## Warning: package 'forcats' was built under R version 3.5.3

## -- Conflicts ------------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter()   masks stats::filter()
## x purrr::flatten()  masks jsonlite::flatten()
## x rjson::fromJSON() masks jsonlite::fromJSON()
## x dplyr::id()       masks twitteR::id()
## x dplyr::lag()      masks stats::lag()
## x dplyr::location() masks twitteR::location()
## x rjson::toJSON()   masks jsonlite::toJSON()
```

```
library(tm)
```

```
## Warning: package 'tm' was built under R version 3.5.3

## Loading required package: NLP

##
## Attaching package: 'NLP'

## The following object is masked from 'package:ggplot2':
##
##     annotate
```

```
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

Twitter in R

```
## From Gates
#  Consumer API keys
#  Access token & access token secret

## Note is it common to create a text file that contains the
## consumerKey, the comsumerSecret, the access_Token, and the access_Secret
## for security purposes. Instead, for simplicity, I include my secrete keys
## below. PLEASE create your own account and use your own keys.
#Insert your consumerKey and consumerSecret below

## Go here for more info:  https://developer.twitter.com/en/account/get-started



consumerKey='SiMslBfTdWEimvLweRDTTrZVH'
consumerSecret='FoPYqK3uwpzutwE6G1RmQvPbRJ8RChFSLfIlgRAcFHjymKDzHh'
```

```
access_Token='1084502204038479872-v2czQaDlMt9ikoLnxhiQYk8Yb3f0RT'
access_Secret='U9ktzvd5rEwcK13mttsgwAujS0VxNPtJstxXcEE5znnid'

requestURL='https://api.twitter.com/oauth/request_token'
accessURL='https://api.twitter.com/oauth/access_token'
authURL='https://api.twitter.com/oauth/authorize'


#filename="TwitterConKey_ConSec_AccTok_AccSec.txt"
#(tokens<-read.csv(filename, header=TRUE, sep=","))
#(consumerKey=as.character(tokens$consumerKey))
#consumerSecret=as.character(tokens$consumerSecret)
#access_Token=as.character(tokens$access_Token)
#access_Secret=as.character(tokens$access_Secret)
```

NOTES: rtweet is another excellent option https://mkearney.github.io/blog/2017/06/01/intro-to-rtweet/ https://rtweet.info/

Using twittR

```
setup_twitter_oauth(consumerKey,consumerSecret,access_Token,access_Secret)
```

```
## [1] "Using direct authentication"
```

```
# Below is the function that scours twitter for a particular hash tag.
# n is the number of tweets to be collected

Search<-twitteR::searchTwitter("SuperBowl",n=300,since="2020-01-30")
Search_DF <- twListToDF(Search)

# If you wish to store the tweets in a csv file ...
TransactionTweetsFile = "tweets.csv"
head(Search_DF$text[1])
```

```
## [1] "Aclamação é a definição do espetáculo apresentado por @Shakira e @JLo no intervalo do #SuperBowl
```

```
## You may find that there are many "stopwords" to remove!!
## These are useless words that may confound analyses.

myStopWordList = c("https", "t.co")


## Start the file
Trans <- file(TransactionTweetsFile)
## Tokenize tweets into a list of words
Tokens<-tokenizers::tokenize_words(Search_DF$text[1],stopwords = append(stopwords::stopwords("en"),myStopWordList),
          lowercase = TRUE,  strip_punct = TRUE, strip_numeric = TRUE,simplify = TRUE)
## Write squished tokens
Tokens["https"]<-""
```

```
cat(unlist(str_squish(Tokens)), "\n", file=Trans, sep=",")
close(Trans)

## Append remaining lists of tokens into file
## NOTE - a list of tokens is the set of words from a Tweet
Trans <- file(TransactionTweetsFile, open = "a")
tokenList = Tokens
for(i in 2:nrow(Search_DF)){
  Tokens<-tokenize_words(Search_DF$text[i],stopwords = append(stopwords::stopwords("en"),myStopWordList
           lowercase = TRUE,  strip_punct = TRUE, simplify = TRUE)
  Tokens["https"]<-""
  cat(unlist(str_squish(Tokens)), "\n", file=Trans, sep=",")
  tokenList <- c(tokenList,  unlist(str_squish(Tokens)))
}
close(Trans)
```

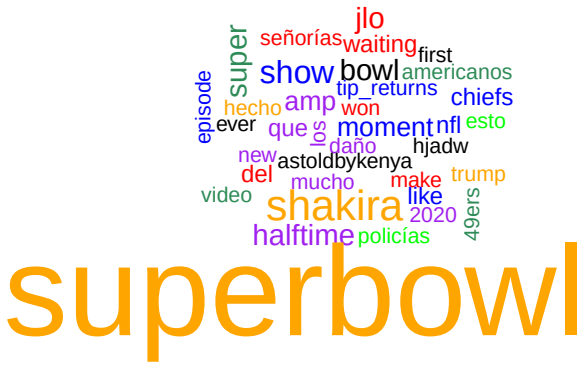
```
# Create a wordcloud, but first transform list of words into a
# TermDocumentMatrix

cor <- Corpus(VectorSource(tokenList))


tdm <- TermDocumentMatrix(cor)
m <- as.matrix(tdm)
v <- sort(rowSums(m),decreasing=TRUE)
d <- data.frame(word = names(v),freq=v)

## NOTE:  d contains the words d$word AND frequencies d$freq

wordcloud(d$word,d$freq, colors=c("red","green","blue","orange","black","purple", "seagreen") , random.
```


jlo
señorías
waiting
super
first
show
bowl
americanos
episode
tip_returns
chiefs
hecho
amp
won
ever
que
los
moment
nfl
esto
daño
hjadw
new
astoldbykenya
del
mucho
make
trump
video
shakira
like
2020
49ers
halftime
policías
superbowl