Word cloud

TK

```
library(dplyr)
library(wordcloud)
library(RColorBrewer)
library(tidytext)
library(SnowballC)

load("rigged_election.RData")
```

Preperation

In this session, we create a word cloud of two goups we detected last session. To do that, we need to extract tweet texts of each group members. Since some of user's texts are stroed in retweet related columns, I create a new dataframe which contain all original tweets as a first step.

```
fields <- c("user_id", "status_id", "created_at", "screen_name", "text", "retweet_c
ount",
            "is_retweet", "is_quote", "reply_to_user_id")

fields_rt <- c("retweet_user_id", "retweet_status_id", "retweet_created_at",
               "retweet_screen_name", "retweet_text", "retweet_retweet_count")


tweets <- rigged %>%
  select(all_of(fields)) %>% # When you write select(fields), you get a warning mes
sage.
  filter(is_retweet == FALSE) %>% # Remove retweets
  filter(is_quote == FALSE) %>% # Remove quotes
  filter(is.na(reply_to_user_id) == TRUE) %>% # Remove replies
  select(-c(is_retweet, is_quote, reply_to_user_id)) # Remove columns we don't need
anymore

head(tweets)
```

```
## # A tibble: 6 × 6
##   user_id   status_id  created_at          screen_name text       retweet_count
##   <chr>     <chr>      <dttm>              <chr>       <chr>              <int>
## 1 78814863… 133492477… 2020-12-04 18:17:23 Donlusin    "\"RIGGED …            7
## 2 30517896… 133492948… 2020-12-04 18:36:05 robertc3d   "Georgia R…            0
## 3 39441344  133492948… 2020-12-04 18:36:04 smalls777   "Yes, yes,…            0
## 4 97367856… 133492946… 2020-12-04 18:35:59 BMccaughrin "Nothing w…            0
## 5 14748990… 133492942… 2020-12-04 18:35:50 WriterLDud… "Trump's c…            0
## 6 11911578… 133492942… 2020-12-04 18:35:50 rlouis82    "Since Don…            0
```

```
tweets_rt <- rigged %>%
  select(all_of(fields_rt)) %>%
  filter(is.na(retweet_user_id) == FALSE) # remove if retweet_user_id is NA

head(tweets_rt)
```

```
## # A tibble: 6 × 6
##   retweet_user_id retweet_status_id   retweet_created_at  retweet_screen_name
##   <chr>           <chr>               <dttm>              <chr>
## 1 32871086        1334654903017299968 2020-12-04 00:25:00 kylegriffin1
## 2 25073877        1333615151904395264 2020-12-01 03:33:24 realDonaldTrump
## 3 25073877        1333975991518187521 2020-12-02 03:27:15 realDonaldTrump
## 4 25073877        1334858852337070083 2020-12-04 13:55:25 realDonaldTrump
## 5 74303349        1334868383955898373 2020-12-04 14:33:18 MarkFinchem
## 6 25073877        1334858852337070083 2020-12-04 13:55:25 realDonaldTrump
## # … with 2 more variables: retweet_text <chr>, retweet_retweet_count <int>
```

```
names(tweets_rt) <- names(tweets) # change names of columns of tweets_rt to bind tw
o data.frame.

all_tweets <- bind_rows(tweets, tweets_rt) # you can use here rbind() as well
dim(all_tweets)
```

```
## [1] 46741     6
```

```
all_tweets <- distinct(all_tweets, status_id, .keep_all = TRUE) # Remove duplicated
status_ids, keep all other variables
dim(all_tweets)
```

```
## [1] 3705    6
```

```
# Check tweet text
trump <- all_tweets %>%
  filter(screen_name == "realDonaldTrump")
trump$text
```

```
##  [1] "RIGGED ELECTION!"
##  [2] "Not statistically possible. Rigged Election! https://t.co/Yw8roUTJhy"
##  [3] "Rigged Election. Show signatures and envelopes. Expose the massive voter f
raud in Georgia. What is Secretary of State and @BrianKempGA afraid of. They know w
hat we’ll find!!! https://t.co/Km7tRm2s1A"
##  [4] "I gave a long news conference today after wishing the military a Happy Tha
nksgiving, &amp; realized once again that the Fake News Media coordinates so that t
he real message of such a conference never gets out. Primary point made was that th
```

```
e 2020 Election was RIGGED, and that I WON!"
##  [5] "We have some big things happening in our various litigations on the Electi
on Hoax. Everybody knows it was Rigged. They know Biden didn’t get more votes from 
the Black community than Obama, &amp; certainly didn’t get 80,000,000 votes. Look w
hat happened in Detroit, Philadelphia, plus!"
##  [6] "...And there are many such articles. Rigged Election! https://t.co/P8scMau
hcI"
##  [7] "RIGGED 2020 ELECTION: MILLIONS OF MAIL-IN BALLOTS WILL BE PRINTED BY FOREI
GN COUNTRIES, AND OTHERS. IT WILL BE THE SCANDAL OF OUR TIMES!"
##  [8] "RIGGED ELECTION!"
##  [9] "“Democrats suffered crushing down-ballot loss across America.” @nytimes. T
his is true. All statehouses won, and in Washington we did great. So I led this gre
at charge, and I’m the only one that lost? No, it doesn’t work that way. This was a
massive fraud, a RIGGED ELECTION!"
## [10] "RIGGED ELECTION. WE WILL WIN!"
## [11] "He only won in the eyes of the FAKE NEWS MEDIA. I concede NOTHING! We have
a long way to go. This was a RIGGED ELECTION!"
## [12] "Just saw the vote tabulations. There is NO WAY Biden got 80,000,000 votes!
!! This was a 100% RIGGED ELECTION."
## [13] "People will not accept this Rigged Election! https://t.co/XQAOIt5ZwU"
## [14] "Poll: 79 Percent of Trump Voters Believe ‘Election Was Stolen‘ https://t.c
o/PmMBmt05AI via @BreitbartNews They are 100% correct, but we are fighting hard. Ou
r big lawsuit, which spells out in great detail all of the ballot fraud and more, w
ill soon be filled. RIGGED ELECTION!"
## [15] "This Election was RIGGED, but we will WIN! https://t.co/luS6SnFscx"
## [16] "Many mostly Democrat States refused to hand over data from the 2016 Electi
on to the Commission On Voter Fraud. They fought hard that the Commission not see t
heir records or methods because they know that many people are voting illegally. Sy
stem is rigged, must go to Voter I.D."
## [17] "Heartwarming to see all of the tremendous support out there, especially th
e organic Rallies that are springing up all over the Country, including a big one o
n Saturday in D.C. I may even try to stop by and say hello. This Election was Rigge
d, from Dominion all the way up &amp; down!"
## [18] "A Rigged Election! https://t.co/dAviFrkEP4"
## [19] "The “Republican” Governor of Georgia, @BrianKempGA, and the Secretary of S
tate, MUST immediately allow a signature verification match on the Presidential Ele
ction. If that happens, we quickly and easily win the State and importantly, pave t
he way for a big David and Kelly WIN!"
## [20] "Breaking News: 50,000 OHIO VOTERS getting WRONG ABSENTEE BALLOTS. Out of c
ontrol. A Rigged Election!!!"
## [21] "Dominion is running our Election. Rigged! https://t.co/xvwrpLpAZa"
## [22] "....Absentee Ballots are fine because you have to go through a precise pro
cess to get your voting privilege. Not so with Mail-Ins. Rigged Election!!! 20% fra
udulent ballots?"
## [23] "He won because the Election was Rigged. NO VOTE WATCHERS OR OBSERVERS allo
wed, vote tabulated by a Radical Left privately owned company, Dominion, with a bad
reputation &amp; bum equipment that couldn’t even qualify for Texas (which I won by
a lot!), the Fake &amp; Silent Media, &amp; more! https://t.co/Exb3C1mAPg"
## [24] "Hundreds of thousands of people showing their support in D.C. They will no
t stand for a Rigged and Corrupt Election! https://t.co/tr35WKTKM8"
## [25] "Top US Pollster and Statistician Richard Baris – People's Pundit – SUSPEND
```

```
ED from Twitter for Reporting on Disputed Election — Political 'WrongThink' Not All
owed https://t.co/uHBZnJJn1I via Said 10,000 DEAD PEOPLE VOTED IN MICHIGAN. When wi
ll this RIGGED ELECTION be overturned!"
```

```
sanders <- all_tweets %>%
  filter(screen_name == "BernieSanders")
sanders$text
```

```
## [1] "Trump's rants about a \"fraudulent election\" are not a joke. They are the
most significant attack against our democracy in history. If the election system's
\"rigged,\" if the media's \"fake,\" if federal officials are part of a \"deep stat
e,\" who can you trust? You got it. A dictator."
```

```
# Find red group's tweets
red_tweets <- all_tweets %>%
  filter(user_id %in% red)

blue_tweets <- all_tweets %>%
  filter(user_id %in% blue)

dim(red_tweets)
```

```
## [1] 189   6
```

```
dim(blue_tweets)
```

```
## [1] 352   6
```

Plotting word cloud

Now let’s create a word cloud. To do that, we first need to clean up tweet texts. Following example demonstrates cleaning tweets text using regular expression and `gsub()` function. `gsub()` function replace all the matches of a pattern from a string. `gsub(pattern, replacement, string)` . For more efficient search, we use **regular explession**.

```
# A function for cleaning tweet text
clean_tweet <- function(tweet_df){
  text <- tweet_df$text
  text <- gsub("https\\S*", "", text)  # remove https.
  text <- gsub("@\\S*", "", text) # remove screen_name
  text <- gsub("[\n]", "", text) # remove line breaks
  text <- gsub("[[:punct:]]", "", text) # remove punctuation
  return(text)
}
```

First, we plot word cloud of the blue group.

```
# Clean text
blue_tweets$text <- clean_tweet(blue_tweets)


tweets_words <- blue_tweets %>%
  select(text) %>%
  unnest_tokens(word, text) # Tokenization: break the text into individual tokens.
unnest_tokens(tbl, output, input) is a function from `tidytext`.

head(tweets_words)
```

```
## # A tibble: 6 × 1
##   word
##   <chr>
## 1 thats
## 2 the
## 3 other
## 4 part
## 5 that
## 6 makes
```

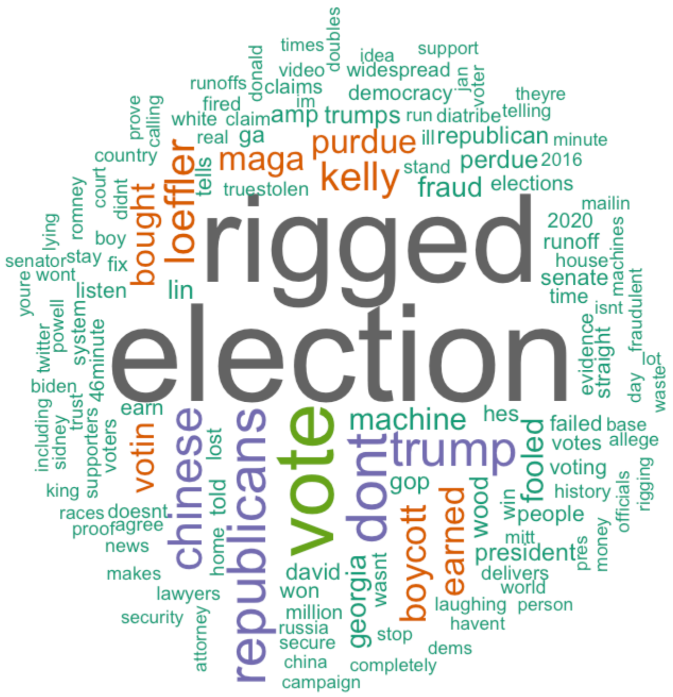
```
tweets_words <- tweets_words %>%
  anti_join(stop_words) # remove stop words: a commonly used word (eg., a, the, tha
t) that does not give us much information
```

```
## Joining, by = "word"
```

```
words <- tweets_words %>%
  count(word, sort = TRUE)

head(words)
```

```
## # A tibble: 6 × 2
##   word            n
##   <chr>       <int>
## 1 election      457
## 2 rigged        406
## 3 vote          248
## 4 dont          153
## 5 republicans   143
## 6 trump         127
```

```
set.seed(3)
wordcloud(words = words$word,
          freq = words$n,
          random.order = FALSE, #plot words in random order. If false, they will be
plotted in decreasing frequency
          rot.per = 0.35, #proportion of words with 90% degree ratation
          colors = brewer.pal(8, "Dark2"),
          scale = c(4,0.5), # the range of the size of the words
          min.freq = 5) # words with frequency below min.freq will not be plotted
```



Exercise

1. Plot word cloud of `red` group.
2. Plot word cloud using any group of twitter texts you are interested in. For example, a group of politician’s recent twitter post, tweets mentioning about the same topic etc..

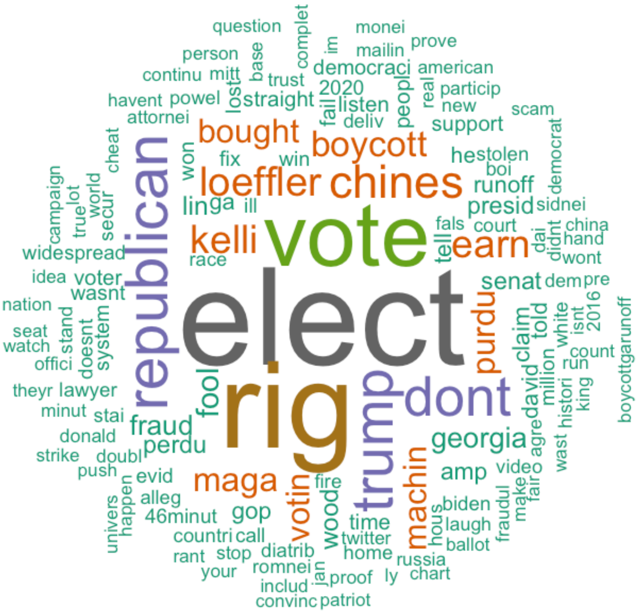
BONUS: Stemming

Stemming is to reduce inflectional forms of a word to a common base form (cars -> car). Stemming usually refers to a crude heuristic process that chops off the ends of words. Porter stemmer is used in the example below.

```
# Stemmed version

tweets_words <- blue_tweets %>%
  select(text) %>%
  unnest_tokens(word, text)

tweets_words <- tweets_words %>%
  anti_join(stop_words)
```

```
## Joining, by = "word"
```

```
words <- tweets_words %>%
  mutate(stem = wordStem(word)) %>% # from SnowballC package
  count(stem, sort = TRUE)

set.seed(5)
wordcloud(words = words$stem, freq = words$n, random.order = FALSE, rot.per = 0.35,
          colors = brewer.pal(8, "Dark2"), scale = c(4,0.5), min.freq = 5)
```



Quick guide for the regural expression in R

> Regexps are a very terse language that allow you to describe patterns in strings (R for Data Science, p.200).

Recommend source for Regular expressions (regexps) :

- https://r4ds.had.co.nz/strings.html (https://r4ds.had.co.nz/strings.html)
- https://bookdown.org/rdpeng/rprogdatascience/regular-expressions.html (https://bookdown.org/rdpeng/rprogdatascience/regular-expressions.html)
- https://regenerativetoday.com/a-complete-beginners-guide-to-regular-expressions-in-r/ (https://regenerativetoday.com/a-complete-beginners-guide-to-regular-expressions-in-r/) #### Most commonly used regureal expression

- `.` : maches any character
- `*` : Repetition. 0 or more times.
- `^` : matches the start of the string.

- `$` : matches the end of the string.
- `\` : escape. We need to use an `escape` to not use its special behavior. For example, `\.` escape special behavior or `.` (maches any character) and allow us to match the character `.` . however, this is not the end. There is one more thing to know. `\` is also used as an escape symbol in strings, and we use strings to represent regular expressions. Thus, we need to wright `\\.` to create the regular expression. In the same manner, to match character `$` , we need to escape `$` ’s special meaning. Therefore, we need to write `\\$` .

```
some_chr <- c("TH Kim", "Kim TH", "T Kim Park", "T Kim", "S T Kim")
grep('TH', some_chr)
```

```
## [1] 1 2
```

```
grep('^T', some_chr)
```

```
## [1] 1 3 4
```

```
grep('Kim$', some_chr)
```

```
## [1] 1 4 5
```

```
grep('^T.*m$', some_chr)
```

```
## [1] 1 4
```