Basic statistics

TK

12/8/2021

```
library(rtweet)
library(dplyr)
library(ggplot2)
```

Explore retrieved data

This session introduces how to extract specific information from retrieved data collected via `rtweet`. When you collect tweets via `rtweet`, it automatically parse nested lists (json format) returned from Twitter and create a data.frame which is convenient form to handle the data in R. If you set `parse = FALSE` when you request data, `rtweet` gives you data in nest lists. According to `rtweet`,

> By default, the rtweet parse process returns nearly all bits of information returned from Twitter. However, users may occasionally encounter new or omitted variables. In these rare cases, the nested list object will be the only way to access these variables.

Let's take a look at actual data returned from Twitter. This example uses 10 recent tweets of Mr.President-elect, Joe Biden (collected at 2nd December). In this case, Twitter first return *Tweet Object* (https://developer.twitter.com/en/docs/twitter-api/v1/data-dictionary/overview/tweet-object) and `rtweet` parse and store the information into a data.base.

```
tweets <- get_timelines('JoeBiden', n = 10) # collect 10 recent tweets published by
Joe Biden using rtweet.
```

Now let's check what we have now in object `tweets`.

```
dim(tweets) # Check dimension of the data. It has 90 columns and 10 rows.
```

```
## [1] 10 90
```

```
head(tweets[,c(1:5)]) # Print first 6 rows, first to fifth columns. Just to check t
he data. You can also try View(tweets).
```

```
## # A tibble: 6 × 5
##   user_id status_id           created_at          screen_name text
##   <chr>   <chr>               <dttm>              <chr>       <chr>
## 1 939091  1333960074650218496 2020-12-02 02:24:00 JoeBiden    "Today, I was pro…
## 2 939091  1333957282502160384 2020-12-02 02:12:54 JoeBiden    "Statement by Pre…
## 3 939091  1333957233948897287 2020-12-02 02:12:42 JoeBiden    "Rosa Parks spark…
## 4 939091  1333915027821240323 2020-12-01 23:25:00 JoeBiden    "This World AIDS …
## 5 939091  1333879041074417664 2020-12-01 21:02:00 JoeBiden    "50 days until we…
## 6 939091  1333856391841386498 2020-12-01 19:32:00 JoeBiden    "My message to ev…
```

```
names(tweets) # Check names of columns.
```

```
##  [1] "user_id"                 "status_id"
##  [3] "created_at"              "screen_name"
##  [5] "text"                    "source"
##  [7] "display_text_width"      "reply_to_status_id"
##  [9] "reply_to_user_id"        "reply_to_screen_name"
## [11] "is_quote"                "is_retweet"
## [13] "favorite_count"          "retweet_count"
## [15] "quote_count"             "reply_count"
## [17] "hashtags"                "symbols"
## [19] "urls_url"                "urls_t.co"
## [21] "urls_expanded_url"       "media_url"
## [23] "media_t.co"              "media_expanded_url"
## [25] "media_type"              "ext_media_url"
## [27] "ext_media_t.co"          "ext_media_expanded_url"
## [29] "ext_media_type"          "mentions_user_id"
## [31] "mentions_screen_name"    "lang"
## [33] "quoted_status_id"        "quoted_text"
## [35] "quoted_created_at"       "quoted_source"
## [37] "quoted_favorite_count"   "quoted_retweet_count"
## [39] "quoted_user_id"          "quoted_screen_name"
## [41] "quoted_name"             "quoted_followers_count"
## [43] "quoted_friends_count"    "quoted_statuses_count"
## [45] "quoted_location"         "quoted_description"
## [47] "quoted_verified"         "retweet_status_id"
## [49] "retweet_text"            "retweet_created_at"
## [51] "retweet_source"          "retweet_favorite_count"
## [53] "retweet_retweet_count"   "retweet_user_id"
## [55] "retweet_screen_name"     "retweet_name"
## [57] "retweet_followers_count" "retweet_friends_count"
## [59] "retweet_statuses_count"  "retweet_location"
## [61] "retweet_description"     "retweet_verified"
## [63] "place_url"               "place_name"
## [65] "place_full_name"         "place_type"
## [67] "country"                 "country_code"
## [69] "geo_coords"              "coords_coords"
## [71] "bbox_coords"             "status_url"
## [73] "name"                    "location"
## [75] "description"             "url"
## [77] "protected"               "followers_count"
## [79] "friends_count"           "listed_count"
## [81] "statuses_count"          "favourites_count"
## [83] "account_created_at"      "verified"
## [85] "profile_url"             "profile_expanded_url"
## [87] "account_lang"            "profile_banner_url"
## [89] "profile_background_url"  "profile_image_url"
```

Basic information on tweets

Let’s print fields which most frequently used.

```
field <- c("created_at",
           "screen_name",
           "text",
           "favorite_count", # N. of favorite
           "retweet_count", # N. of retweet
           "is_retweet", # if TRUE, this is a retweet
           "is_quote", # if TRUE, this is a quite
           "reply_to_screen_name") # if there is any screen name, this tweet is a r
eply to the screen name.
print(tweets[1,field], width = Inf) # Print first tweets of above fields.
```

```
## # A tibble: 1 × 8
##   created_at          screen_name
##   <dttm>              <chr>
## 1 2020-12-02 02:24:00 JoeBiden
##   text
##   <chr>
## 1 "Today, I was proud to announce key nominations and appointments for critical…
##   favorite_count retweet_count is_retweet is_quote reply_to_screen_name
##            <int>         <int> <lgl>      <lgl>    <lgl>
## 1          24811          2051 FALSE      FALSE    NA
```

User information

Information about an author of a tweet is also included. Below code print most basic information on a user.

```
field <- c("user_id",
           "screen_name",
           "friends_count",
           "followers_count")

print(tweets[1,field], width = Inf)
```

```
## # A tibble: 1 × 4
##   user_id screen_name friends_count followers_count
##   <chr>   <chr>               <int>           <int>
## 1 939091  JoeBiden               31        20377702
```

Now let’s print all the user information. User information are stored from 73rd column to 90st column.

```
print(tweets[1,c(73:dim(tweets)[2])], width = Inf)
```

```
## # A tibble: 1 × 18
##   name      location
##   <chr>     <chr>
## 1 Joe Biden Wilmington, DE
##   description
##   <chr>
## 1 President-elect, husband to @DrBiden, proud father & grandfather. Ready to bu…
##   url                     protected followers_count friends_count listed_count
##   <chr>                   <lgl>               <int>         <int>        <int>
## 1 https://t.co/UClrPuJpyZ FALSE            20377702            31        29827
##   statuses_count favourites_count account_created_at  verified
##            <int>            <int> <dttm>              <lgl>
## 1           6886               20 2007-03-11 17:51:24 TRUE
##   profile_url             profile_expanded_url account_lang
##   <chr>                   <chr>                <lgl>
## 1 https://t.co/UClrPuJpyZ http://joebiden.com  NA
##   profile_banner_url
##   <chr>
## 1 https://pbs.twimg.com/profile_banners/939091/1604514209
##   profile_background_url
##   <chr>
## 1 http://abs.twimg.com/images/themes/theme1/bg.png
##   profile_image_url
##   <chr>
## 1 http://pbs.twimg.com/profile_images/1308769664240160770/AfgzWVE7_normal.jpg
```

Retweet, Quote

In twitter, there are two ways to pass along other’s tweets: retweet and quote. When you simply share tweets posted by others (or your own tweets), that is retweet. When you add additional comments, it becomes quote.

Let’s check which tweets are retweets or quotes.

```
# Is this retweet or quote?
tweets[,"is_retweet"]
```

```
## # A tibble: 10 × 1
##    is_retweet
##    <lgl>
##  1 FALSE
##  2 TRUE
##  3 TRUE
##  4 FALSE
##  5 FALSE
##  6 FALSE
##  7 FALSE
##  8 FALSE
##  9 FALSE
## 10 FALSE
```

```
tweets[,"is_quote"]
```

```
## # A tibble: 10 × 1
##    is_quote
##    <lgl>
##  1 FALSE
##  2 FALSE
##  3 FALSE
##  4 FALSE
##  5 FALSE
##  6 FALSE
##  7 FALSE
##  8 TRUE
##  9 FALSE
## 10 FALSE
```

Okay, so second tweets and eighth tweet are retweet and quote tweet, respectively.

```
# Is this retweet or quote?
tweets[2, 'text']
```

```
## # A tibble: 1 × 1
##   text
##   <chr>
## 1 Statement by President-elect Biden on the U.S. Supreme Court case on the Cens…
```

```
tweets[8, 'text']
```

```
## # A tibble: 1 × 1
##   text
##   <chr>
## 1 .@TTDAFLCIO President Larry Willis was a relentless champion for working fami…
```

If a tweet is a retweet or a quote tweet, the information on the original tweet is also included. `^retweet` is a regular expression indicating that a string starts from `retweet`. Thus `grep("^retweet", names(tweets))` will return a boolean vector with TRUE representing a column name start with `retweet`, i.e., a column has information on the original tweet.

```
field <- grep("^retweet", names(tweets))
names(tweets)[field] # Print column names start with retweet
```

```
##  [1] "retweet_count"           "retweet_status_id"
##  [3] "retweet_text"            "retweet_created_at"
##  [5] "retweet_source"          "retweet_favorite_count"
##  [7] "retweet_retweet_count"   "retweet_user_id"
##  [9] "retweet_screen_name"     "retweet_name"
## [11] "retweet_followers_count" "retweet_friends_count"
## [13] "retweet_statuses_count"  "retweet_location"
## [15] "retweet_description"     "retweet_verified"
```

```
print(tweets[2, field], width = Inf) # Print above fields in the second tweets (whi
ch is a retweet thus should have information on the original tweet),
## # A tibble: 1 × 16
##   retweet_count retweet_status_id
##           <int> <chr>
## 1          1785 1333948826512728064
##   retweet_text
##   <chr>
## 1 Statement by President-elect Biden on the U.S. Supreme Court case on the Cens…
##   retweet_created_at  retweet_source  retweet_favorite_count
##   <dttm>              <chr>                            <int>
## 1 2020-12-02 01:39:18 Twitter Web App                  12340
##   retweet_retweet_count retweet_user_id     retweet_screen_name
##                   <int> <chr>               <chr>
## 1                  1785 1323730225067339784 Transition46
##   retweet_name                         retweet_followers_count
##   <chr>                                                  <int>
## 1 Biden-Harris Presidential Transition                 1081457
##   retweet_friends_count retweet_statuses_count retweet_location
##                   <int>                  <int> <chr>
## 1                    24                     86 United States of America
##   retweet_description
##   <chr>
## 1 The official account of the Biden-Harris presidential transition.
##   retweet_verified
##   <lgl>
## 1 TRUE
```

In the same manner, we can check the information of the original tweet of quoted tweet.

```
field <- grep("^quote", names(tweets))
names(tweets)[field]
```

```
##  [1] "quote_count"            "quoted_status_id"       "quoted_text"
##  [4] "quoted_created_at"      "quoted_source"          "quoted_favorite_count"
##  [7] "quoted_retweet_count"   "quoted_user_id"         "quoted_screen_name"
## [10] "quoted_name"            "quoted_followers_count" "quoted_friends_count"
## [13] "quoted_statuses_count"  "quoted_location"        "quoted_description"
## [16] "quoted_verified"
```

```
print(tweets[8, field], width = Inf)
```

```
## # A tibble: 1 × 16
##   quote_count quoted_status_id
##         <int> <chr>
## 1          NA 1333428832368427008
##   quoted_text
##   <chr>
## 1 Yesterday, with his wife and daughter by his side, TTD president Larry Willis…
##   quoted_created_at   quoted_source   quoted_favorite_count quoted_retweet_count
##   <dttm>              <chr>                           <int>                <int>
## 1 2020-11-30 15:13:02 Twitter Web App                   515                   91
##   quoted_user_id quoted_screen_name quoted_name          quoted_followers_count
##   <chr>          <chr>              <chr>                                 <int>
## 1 292552239      TTDAFLCIO          Transp. Trades Dept.                   3584
##   quoted_friends_count quoted_statuses_count quoted_location
##                  <int>                 <int> <chr>
## 1                 1196                 16499 Washington, DC
##   quoted_description
##   <chr>
## 1 Transportation Trades Department, AFL-CIO | Fighting at the federal level for…
##   quoted_verified
##   <lgl>
## 1 TRUE
```

Exercise

1. Collect 100 most recent tweets published from a candidate who ran the German federal election in 2021.
2. Find out following information - User information

- 2-1. Name, Created date, Location, Profile description
- 2-2. Is this account varified?
- 2-3. How many followers and friends the account have?

3. Find out following information - Tweet information

- 3-1. How many of tweets are retweets?
- 3-2. How many of tweets are quotes?
- 3-3. How many time their original tweets (not retweet nor quote) are retweeted by others (on average)?

Example: Compare three different accounts’ Twitter activity

In this example, we compare three different accounts’ activity. First, let’s collect our exemplary data. We compare three German party’s offical account.

```
party.timeline <- get_timelines(c("AfD", "CDU","spdde"), n = 3000)
save(file = "party_timeline.RData", party.timeline) # save the data if you want
```

Check the number of tweets we retrieved

```
# Check the data
head(party.timeline)[,c(1:4)]
```

```
## # A tibble: 6 × 4
##   user_id   status_id           created_at          screen_name
##   <chr>     <chr>               <dttm>              <chr>
## 1 844081278 1333450724806717445 2020-11-30 16:40:01 AfD
## 2 844081278 1333363772803702785 2020-11-30 10:54:30 AfD
## 3 844081278 1333054818395566084 2020-11-29 14:26:50 AfD
## 4 844081278 1333049173042745347 2020-11-29 14:04:24 AfD
## 5 844081278 1333039980856430595 2020-11-29 13:27:52 AfD
## 6 844081278 1332991201163816961 2020-11-29 10:14:02 AfD
```

```
dim(party.timeline) # Check the size of the data
```

```
## [1] 8997   90
```

We collected about 3,000 tweets from each account.

```
table(party.timeline$screen_name)
```

```
##
##   AfD   CDU spdde
##  3000  2999  2998
```

Basic information about three accounts

Let’s check the basic user information of three accounts. Here we use `dplyr` package to manage the data.

```
ac.info <- party.timeline %>%
    group_by(screen_name) %>% #Grouping rows based on the screen_name
    summarize(user_id[1], name[1], statuses_count[1], account_created_at[1], verifi
ed[1],
              friends_count[1], followers_count[1], description[1]) # Select basic
user information
print(ac.info, width = Inf) # Print it
```

```
## # A tibble: 3 × 9
##   screen_name `user_id[1]` `name[1]`                          `statuses_count[1]`
##   <chr>       <chr>        <chr>                                            <int>
## 1 AfD         844081278    Alternative für 🇩🇪 Deutschland                  22096
## 2 CDU         20429858     CDU Deutschlands                                 24839
## 3 spdde       26458162     SPD Parteivorstand 🇪🇺                           48980
##   `account_created_at[1]` `verified[1]` `friends_count[1]` `followers_count[1]`
##   <dttm>                  <lgl>                      <int>                <int>
## 1 2012-09-24 18:43:59     TRUE                         893               166459
## 2 2009-02-09 11:43:27     TRUE                        1603               335486
## 3 2009-03-25 08:41:02     TRUE                        4076               388529
##   `description[1]`
##   <chr>
## 1 Offizieller Account der Alternative für Deutschland (#AfD) | Impressum: https…
## 2 Die #CDU ist die Volkspartei der Mitte. Seit 1945. - Redaktion: https://t.co/…
## 3 Tweets aus der Parteizentrale der #SPD. Auf spd.de gibt's alles rund um sozia…
```

Account activity

Original tweets, retweets, quotes, replies

Let’s create a table summarizing each account activities. Below, we count each accounts’ original tweets, retweets, quotes and replies. Here original tweets mean tweets which are not categorized into the rest of three categories.

```
twitter_activity <- party.timeline %>%
  group_by(screen_name) %>%
  summarise(n(), # count total number of tweets
            sum(is_retweet == TRUE), # count retweets
            sum(is_quote == TRUE), # count quotes
            sum(is.na(reply_to_user_id) == FALSE)) %>% # count replies
  rename("account" = 1, "total" = 2,  "retweets" = 3, "quotes" = 4, "replies" = 5)
%>%
  mutate(original = total - (retweets + quotes + replies)) # create the column `ori
ginal`

twitter_activity
```

```
## # A tibble: 3 × 6
##   account total retweets quotes replies original
##   <chr>   <int>    <int>  <int>   <int>    <int>
## 1 AfD      3000     1675     55     417      853
## 2 CDU      2999      486    517     557     1439
## 3 spdde    2998     2301    291     131      275
```

Pychart

Below code creates a pychart showing the SPD’s account activity. The code is taken from here (https://www.r-graph-gallery.com/128-ring-or-donut-plot.html).

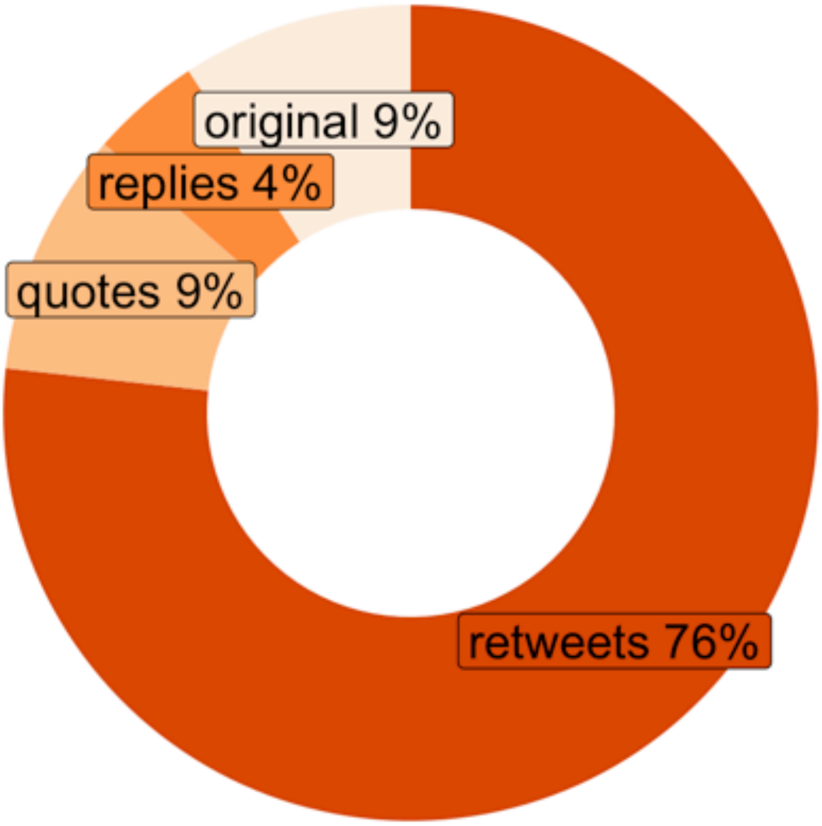
```
# Make a chart. spd's activity

spd_act <- t(twitter_activity[3,3:6]) #transpose
spd_act <- as.data.frame(spd_act)
names(spd_act) <- "n"
spd_act$fract = spd_act$n / sum(spd_act$n)
spd_act$perc = spd_act$fract * 100
spd_act$ymax = cumsum(spd_act$fract)
spd_act$ymin = c(0, head(spd_act$ymax, n = -1))
spd_act$label_pos <- (spd_act$ymax + spd_act$ymin) / 2
spd_act$label = paste0(row.names(spd_act)," ", as.integer(spd_act$perc), "%")

spd_act
```

```
##             n      fract      perc      ymax      ymin label_pos        label
## retweets 2301 0.76751167 76.751167 0.7675117 0.0000000 0.3837558 retweets 76%
## quotes    291 0.09706471  9.706471 0.8645764 0.7675117 0.8160440    quotes 9%
## replies   131 0.04369580  4.369580 0.9082722 0.8645764 0.8864243   replies 4%
## original  275 0.09172782  9.172782 1.0000000 0.9082722 0.9541361  original 9%
```

```
ggplot(spd_act, aes(ymax = ymax, ymin = ymin, xmax = 4, xmin = 3, fill = row.names(
spd_act))) +
  geom_rect() +
  geom_label( x=3.5, aes(y = label_pos, label = label), size = 6) +
  scale_fill_brewer(palette = 7) +
  coord_polar(theta="y") +
  xlim(c(2, 4)) +
  theme_void() +
  theme(legend.position = "none")
```



Attention from other twitter users

Now let's check how many attention each account received from other twitter users. In this example, we focus on retweeted number of original tweets. First we create a data object `ori_tweets` which stores only original tweets.

```
ori_tweets <- party.timeline %>%
  filter(is_retweet == FALSE) %>% # Remove retweet
  filter(is.na(reply_to_user_id) == TRUE) %>% # Remove replies
  filter(is_quote == FALSE)  # Remove quote
```

Below code shows the earliest date of a original tweet published in each three account.

```
party.timeline %>%
  group_by(screen_name) %>%
  summarize(min(created_at))
```

```
## # A tibble: 3 × 2
##   screen_name `min(created_at)`
##   <chr>       <dttm>
## 1 AfD         2019-09-01 13:22:28
## 2 CDU         2019-11-22 13:11:38
## 3 spdde       2020-06-30 15:02:30
```

First, we remove tweets published before 30 June 2020 to make the same time frame for three accounts. In the next step, we get sum of retweets and mean of retweets for each account.

```
ori_tweets %>%
  filter(created_at > "2020-06-30") %>% # remove tweets published before 2020-06-30
  group_by(screen_name) %>%
  summarise(n(), sum(retweet_count), mean(retweet_count))
```

```
## # A tibble: 3 × 4
##   screen_name `n()` `sum(retweet_count)` `mean(retweet_count)`
##   <chr>       <int>                <int>                  <dbl>
## 1 AfD           264                22280                   84.4
## 2 CDU           430                 6809                   15.8
## 3 spdde         278                 5457                   19.6
```

We can also check which original tweets received high attention in the following way.

```
# Most retweeted tweets
top_retweet <- ori_tweets %>%
  group_by(screen_name) %>%
  arrange(desc(retweet_count), .by_group = TRUE) %>% # reordering tweets in decreas
ing order based on the retweet count.
  summarise(text[1:10], retweet_count[1:10]) %>% # Show first 10 tweets of text and
retweet count.
  rename("text" = 2, "retweet_count" = 3)
```

```
## `summarise()` has grouped output by 'screen_name'. You can override using the `.
groups` argument.
```

```
print(top_retweet, n = Inf)
```

```
## # A tibble: 30 × 3
## # Groups:   screen_name [3]
##    screen_name text                                                 retweet_count
##    <chr>       <chr>                                                        <int>
##  1 AfD         "Ansprache des #AfD-Bundessprechers Prof. Dr. @Joe…            635
##  2 AfD         "Die Patrioten von @vox_es ziehen mit etwa 15% in …            555
##  3 AfD         "Der Europäische Gerichtshof für Menschenrechte (#…            550
##  4 AfD         "Die #BLM-Bewegung in den USA scheint zu einer ras…            511
##  5 AfD         "#AfD-Bundesvorstand stellt Strafanzeige gegen Kan…            481
##  6 AfD         "Wir brauchen kein #Alkoholverbot und auch keine „…            469
##  7 AfD         "++ Grüne stoppen! Umwelt schützen! ++\nAuch die N…            398
##  8 AfD         "Wir wir gerade erfahren, hat @_FriedrichMerz offe…            380
##  9 AfD         "Diese Nazivergleiche etwa eines Peter Frey vom @Z…            357
## 10 AfD         "++ ❗ 4. Jahrestag der eigenmächtigen Grenzöffnung…             3
57
## 11 CDU         "Die CDU wird 75. 🎂 Wir erinnern in 120 Sekunden a…            4
23
## 12 CDU         "Zum #ff unsere Tipps und Empfehlungen, um mit Inf…            373
## 13 CDU         "Pressestatement zur Wahl des Ministerpräsidenten …            283
## 14 CDU         "Morgen vor 15 Jahren wurde Angela #Merkel zur ers…            198
## 15 CDU         "Bundeskanzlerin #Merkel: “Niemand hört es gerne, …            197
## 16 CDU         "Vor 67 Jahren wurde der DDR-Volksaufstand brutal …            152
## 17 CDU         "Zu unserer Haltung gegenüber AfD und Linkspartei …            152
## 18 CDU         "🎂 Wir wünschen Ihnen alles Gute zum Geburtstag, l…             1
39
## 19 CDU         ".@paulziemiak im #Bundestag: Wir gedenken heute d…            137
## 20 CDU         "Helmut Kohls Leben war ein Leben für 🇩🇪, für 🇪🇺 u…            11
5
## 21 spdde       "Er war der erste Vorsitzende der wiedervereinigte…            427
## 22 spdde       "Die Bilder sind bestürzend und beschämend: Reichs…            200
## 23 spdde       "Congrats, Joe and Kamala! 🥳🇺🇸👏🏻 Das Ergebnis der …             1
63
## 24 spdde       "Wir sind geschockt von dem plötzlichen Tod von Th…            152
## 25 spdde       "„Jemand, der sich beleidigt zurückzieht, weil er …            149
## 26 spdde       "Wir trauern heute um die 77 Menschen, die vor neu…            123
## 27 spdde       "Gute Neuigkeiten! Das #Kurzarbeitergeld wird verl…             97
## 28 spdde       "\"Wir wollen einen Sozialstaat. Wir wollen Respek…             92
## 29 spdde       "„Es ist meine Aufgabe als Parteichefin der SPD, b…             70
## 30 spdde       "Gegen rechtes Gedankengut kämpfen wir für Euch sc…             69
```

Plot frequency of tweets

Static plot

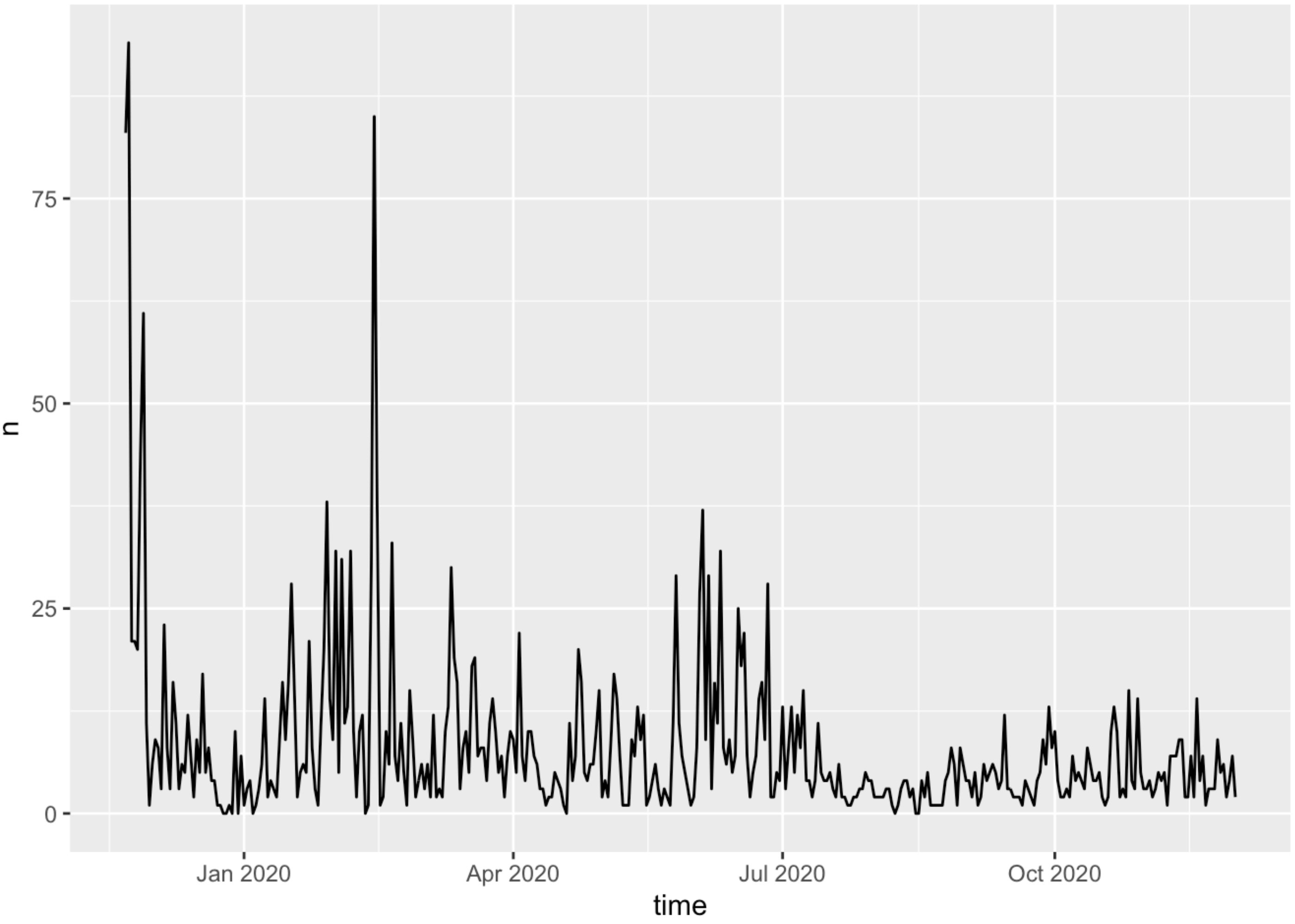
Plotting help us to grasp the trend of tweets. `rtweet` provides function to do it quickly. Let's generate number of daily tweets and plot it. In the following example, we use `CDU` 's data.

```
# Get daily stats of CDU
party.timeline %>%
  filter(screen_name == "CDU") %>% # Get tweets from CDU
  ts_data # Generate number of daily tweets. This function comes from `rtweet`.
```

```
## # A tibble: 376 × 2
##    time                    n
##    <dttm>              <int>
##  1 2019-11-22 00:00:00    83
##  2 2019-11-23 00:00:00    94
##  3 2019-11-24 00:00:00    21
##  4 2019-11-25 00:00:00    21
##  5 2019-11-26 00:00:00    20
##  6 2019-11-27 00:00:00    44
##  7 2019-11-28 00:00:00    61
##  8 2019-11-29 00:00:00    11
##  9 2019-11-30 00:00:00     1
## 10 2019-12-01 00:00:00     6
## # … with 366 more rows
```

You can also plot daily tweets using `ts_plot` function in `rtweet` .

```
# Plot using ts_plot
# ts_plot: Plots tweets data as a time series-like data object
party.timeline %>%
  filter(screen_name == "CDU") %>%
  ts_plot("days") # Set interval. You can change this parameter. For more detail, s
ee the doc.
```



Adding functions from `ggplot2` , we can make the plot prettier. In the following example compares trends of three party accounts.

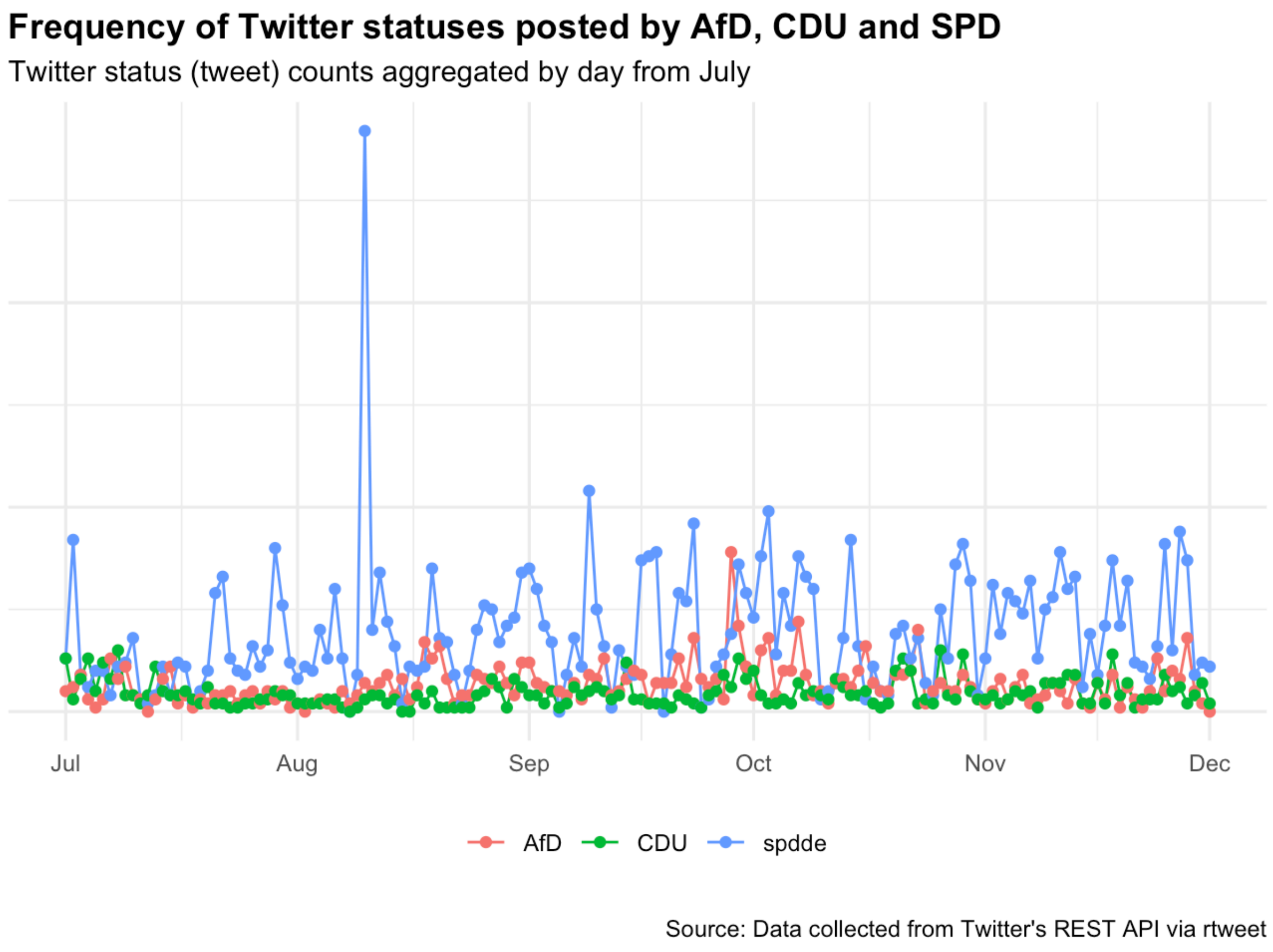
```
gr <- party.timeline %>%
  filter(created_at >= "2020-06-30") %>%
  group_by(screen_name)%>%
  summarise(n())
gr
```

```
## # A tibble: 3 × 2
##   screen_name `n()`
##   <chr>       <int>
## 1 AfD          1040
## 2 CDU           692
## 3 spdde        2998
```

```
# plot the frequency of tweets for each user over time
# codes are taken from : https://rtweet.info

p <- party.timeline %>%
  dplyr::filter(created_at >= "2020-07-01") %>%
  dplyr::group_by(screen_name) %>%
  ts_plot("days") + # function in rtweet
  ggplot2::geom_point() +
  ggplot2::theme_minimal() + # minimalistic theme
  ggplot2::theme(
    legend.title = ggplot2::element_blank(), # draws nothing, and assigns no space.
    legend.position = "bottom",
    plot.title = ggplot2::element_text(face = "bold")) + # Font face ("plain", "italic", "bold", "bold.italic")
  ggplot2::labs(
    x = NULL, y = NULL,
    title = "Frequency of Twitter statuses posted by AfD, CDU and SPD",
    subtitle = "Twitter status (tweet) counts aggregated by day from July ",
    caption = "\nSource: Data collected from Twitter's REST API via rtweet"
  )

p
```



Frequency of Twitter statuses posted by AfD, CDU and SPD

Twitter status (tweet) counts aggregated by day from July

Source: Data collected from Twitter's REST API via rtweet

Interactive plot using `plotly`

This section introduces `plotly` which help us to generate interactive plots. For more detail about `plotly` , see plotly documentation (https://plotly.com/r/getting-started/).

```r
install.packages("plotly")  # install the package if you don't have it.
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

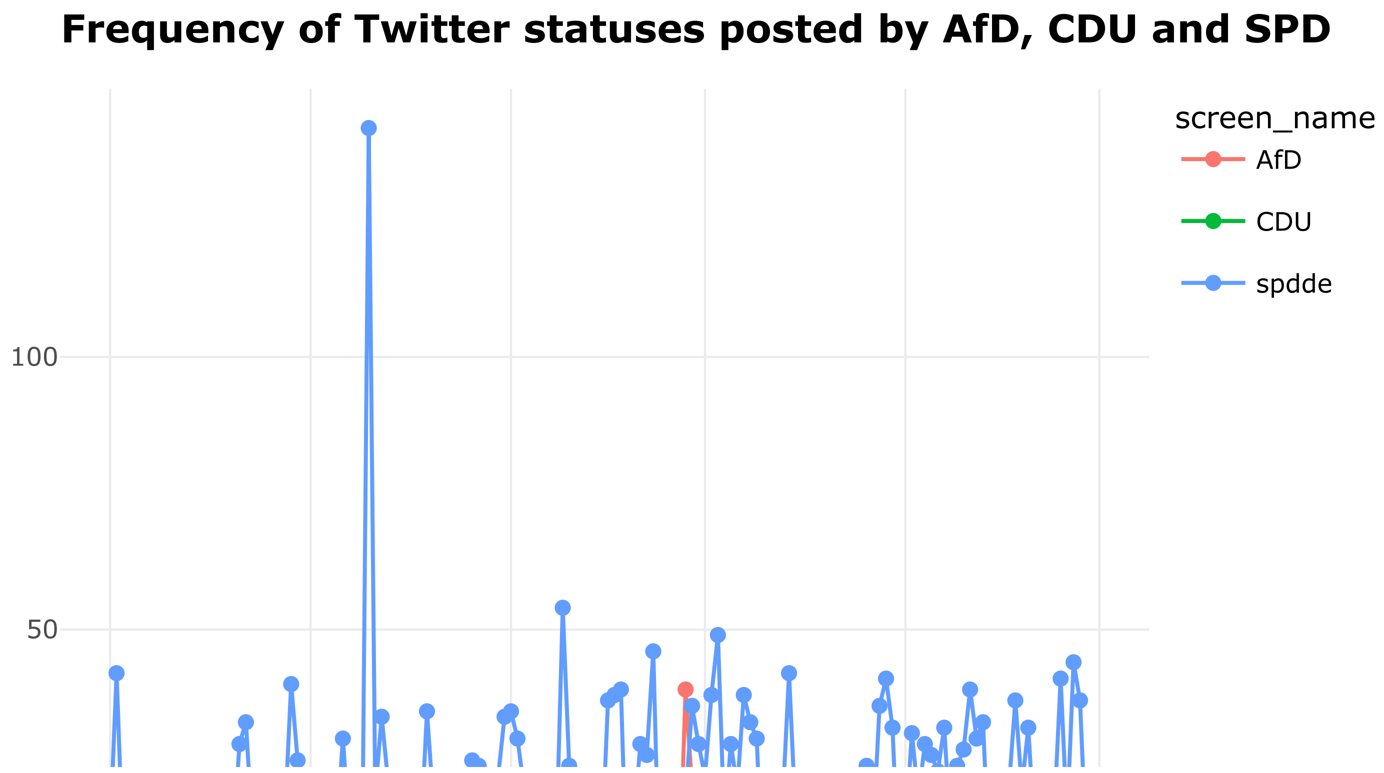
```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

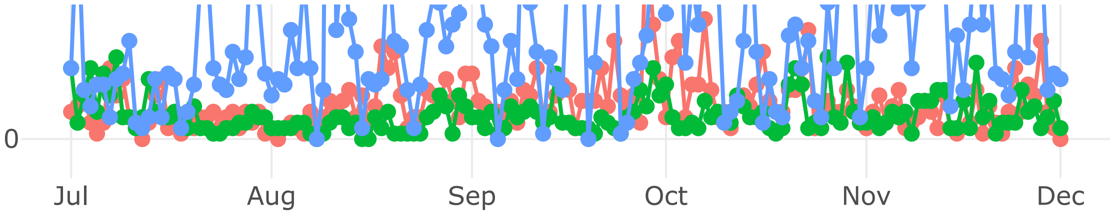
```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

You can change ggplot to interactive plot using `ggplotly()` .

```r
ggplotly(p)
```

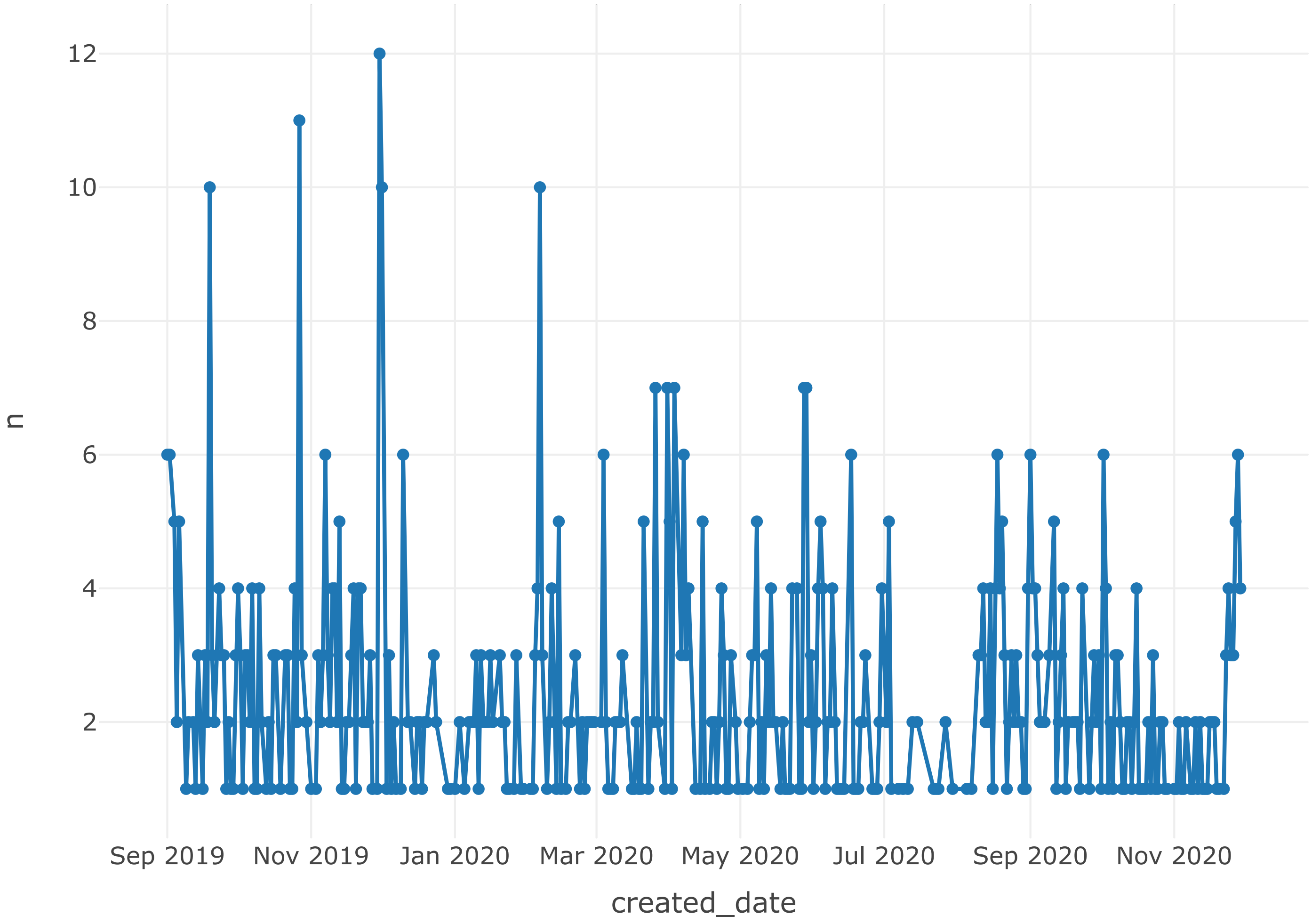




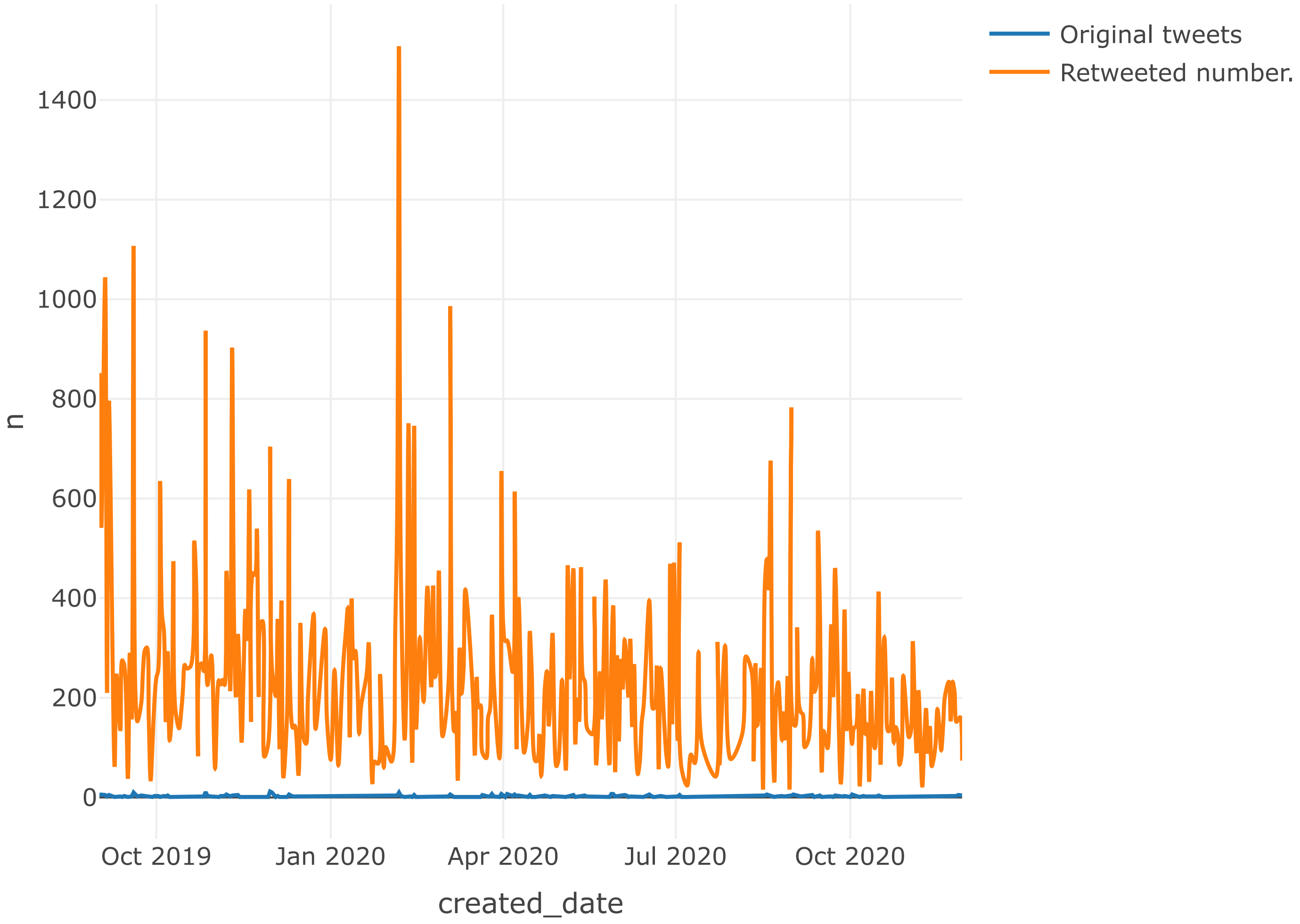
You can also create plot a graph using `plot_ly()` . In this example, let's plot AfD's original tweets' counts and retweet numbers of original tweets (by day). First prepare a data.frame for the plot.

```
fr_daily <- ori_tweets %>%
  filter(screen_name == "AfD") %>%
  mutate("created_date" = as.Date(created_at)) %>%
  group_by(created_date) %>%
  summarise(n(), sum(retweet_count)) %>%
  rename("n" = "n()", "rt_n" = "sum(retweet_count)")


# Plot oritginal tweets' count
plot_ly(data = fr_daily, x = ~created_date, y = ~n, type = 'scatter', mode = 'lines
+markers')
```



```
# Now plot both tweets' count and retweet numbers
plot_ly(data = fr_daily, x = ~created_date) %>%
  add_lines(y = ~n,
            name = "Original tweets",
            type = 'scatter',
            mode = 'lines',
            line = list(shape = "linear")) %>%
  add_lines(y = ~rt_n,
            name = "Retweeted number.",
            type = 'scatter',
            mode = 'lines',
            line = list(shape = "spline"),
            connectgaps = TRUE)
```



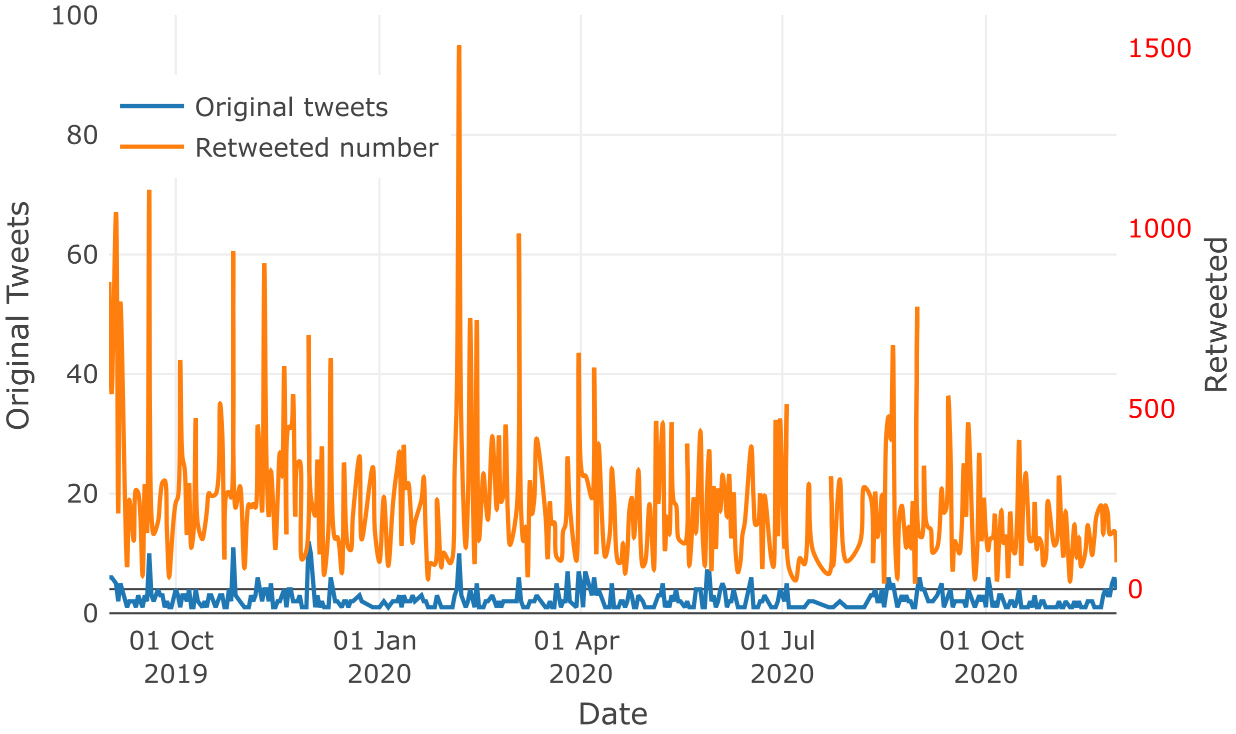
Above plot does not look good since two lines are overlapped. Let’s set two different y axis.

```
ay <- list(
  tickfont = list(color = "red"),
  overlaying = "y",
  side = "right",
  title = "Retweeted",
  showgrid = FALSE
)


mg <- list(
  l = 100,
  r = 100,
  b = 100,
  t = 100,
  pad = 4
)


## Plot
p <- plot_ly(data = fr_daily, x = ~created_date) %>%
  add_lines(y = ~n,
            name = "Original tweets",
            type = 'scatter',
            mode = 'lines',
            line = list(shape = "linear")
  ) %>%
  add_lines(y = ~rt_n,
            name = "Retweeted number",
            yaxis = "y2",
            type = 'scatter',
            mode = 'lines',
            line = list(shape = "spline"),
            connectgaps = TRUE
  ) %>%
  layout(
    # title = "Double Y Axis",
    yaxis2 = ay,
    #annotations = anno.day,
    yaxis = list(title = "Original Tweets", range = c(0, 100)),
    xaxis = list(title = "Date",
                 type = 'date',
                 tickformat = "%d %b <br>%Y")
    ,legend = list(x = 0, y = 0.9),
    margin = mg
  )

p
```


100
80
60
40
20
0
Original Tweets
Original tweets
Retweeted number
1500
1000
500
0
Retweeted
01 Oct
2019
01 Jan
2020
01 Apr
2020
01 Jul
2020
01 Oct
2020
Date