Filtering tweets

ANALYZING SOCIAL MEDIA DATA IN R

Vivek Vijayaraghavan
Data Science Coach

Lesson Overview

- Filtering based on tweet components
  - Extract original tweets
  - Language of the tweet
  - Popular tweets based on minimum number of retweets and favorites

Filtering for original tweets

- An original tweet is an original posting by a twitter user
- Not a retweet, quote, or reply
- Original tweets ensure that content is not repetitive
- Helps retain user engagement levels

Filtering for original tweets

- `-filter` used to extract original tweets
- `-filter:retweets` excludes all retweets
- `-filter:quote` filters out quoted tweets
- `-filter:replies` ensures reply type tweets are filtered out

Extract tweets without filters

- Extract tweets on "digital marketing" without any filters

```
# Extract 100 tweets on "digital marketing"
tweets_all <- search_tweets("digital marketing", n = 100)
```

Extract tweets without filters

- Check count of values in columns `reply_to_screen_name` , `is_quote` , `is_retweet`

```
# Check for count of replies
library(plyr)
count(tweets_all$reply_to_screen_name)
```

```
x                freq
<fctr>           <int>
blairaasmith       2
javiergosende      1
juanburgos         1
WhutTheHale        2
NA                94
```

Extract tweets without filters

```
# Check for count of quotes
count(tweets_all$is_quote)
```

```
x         freq
<lgl>     <int>
FALSE      98
TRUE        2
```

Extract tweets without filters

```
# Check for count of retweets
count(tweets_all$is_retweet)
```

```
x         freq
<lgl>    <int>
FALSE      61
TRUE       39
```

Exclude retweets, quotes, and replies

- Extract tweets on "digital marketing" applying the `-filter`

```
# Apply the '-filter'
tweets_org <- search_tweets("digital marketing
                            -filter:retweets
                            -filter:quote
                            -filter:replies",
                            n = 100)
```

Exclude retweets, quotes, and replies

- Check output to see if replies, quotes, and retweets are excluded

```
# Check for count of replies
library(plyr)
count(tweets_org$reply_to_screen_name)
```

```
x          freq
<lgl>     <int>
NA          100
```

Exclude retweets, quotes, and replies

```
# Check for count of quotes
library(plyr)
count(tweets_org$is_quote)
```

```
x          freq
<lgl>     <int>
FALSE      100
```

```
# Check for count of retweets
library(plyr)
count(tweets_org$is_retweet)
```

```
x          freq
<lgl>     <int>
FALSE      100
```

Filtering tweets on language

- `lang` filters tweets based on language
- Matches tweets of a particular language

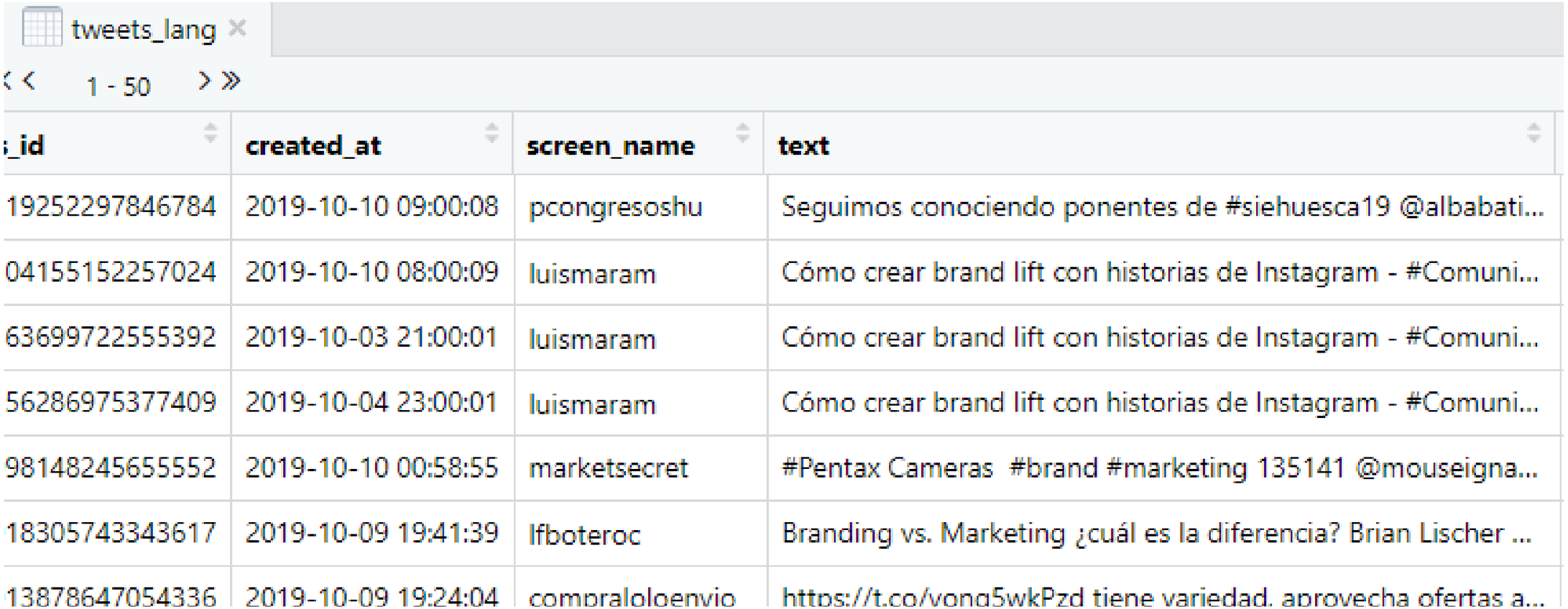
Name	Language code
English (default)	en
German	de
Spanish	es
French	fr
Italian	it
Japanese	ja
Chinese (Traditional)	zh-tw

Filtering tweets on language

```
# Filter and extract tweets posted in Spanish
tweets_lang <- search_tweets("brand marketing", lang = "es")
```

Filtering tweets on language

```
View(tweets_lang)
```



tweets_lang

1 - 50

_id	created_at	screen_name	text
19252297846784	2019-10-10 09:00:08	pcongresoshu	Seguimos conociendo ponentes de #siehuesca19 @albabati...
04155152257024	2019-10-10 08:00:09	luismaram	Cómo crear brand lift con historias de Instagram - #Comuni...
63699722555392	2019-10-03 21:00:01	luismaram	Cómo crear brand lift con historias de Instagram - #Comuni...
56286975377409	2019-10-04 23:00:01	luismaram	Cómo crear brand lift con historias de Instagram - #Comuni...
98148245655552	2019-10-10 00:58:55	marketsecret	#Pentax Cameras #brand #marketing 135141 @mouseigna...
18305743343617	2019-10-09 19:41:39	lfboteroc	Branding vs. Marketing ¿cuál es la diferencia? Brian Lischer ...
13878647054336	2019-10-09 19:24:04	compraloloenvio	https://t.co/vong5wkPzd tiene variedad, aprovecha ofertas a...

Filtering tweets on language

```
head(tweets_lang$lang)
```

```
[1] "es" "es" "es" "es" "es" "es"
```

Filter by retweet and favorite counts

- `min_faves:` filter tweets with minimum number of favorites
- `min_retweets:` filter tweets with minimum number of retweets
- Use `AND` operator to check for both conditions

Filter by retweet and favorite counts

```
# Extract tweets with minimum 100 favorites and retweets
tweets_pop <- search_tweets("bitcoin min_faves:100 AND
                            min_retweets:100")
```

Filter by retweet and favorite counts

```
# Create a data frame to check retweet and favorite counts
counts <- tweets_pop[c("retweet_count", "favorite_count")]
```

```
head(counts)
```

```
retweet_count    favorite_count
    <int>              <int>
1     162                833
2     141                894
3     164               1128
4     395               1346
5     475               2271
6     270               1654
```

Filter by retweet and favorite counts

```
# View the tweets
head(tweets_pop$text)
```

```
text
<chr>
1    As we continue to build the Bakkt Bitcoin Futures contract, we reached a
2    BREAKING: The United States is considering entering into a "currency pact"
3    REMINDER: The Bitcoin ETF will eventually get approved.\n\nNot a question
4    [New Post] Bitcoin is becoming much more important in Hong Kong and India.
5    Reports are surfacing that some Hong Kong ATMs have run out of cash as
6    Bitcoin is the most transparent currency ever created.
```

Let's practice!

ANALYZING SOCIAL MEDIA DATA IN R

Twitter user analysis

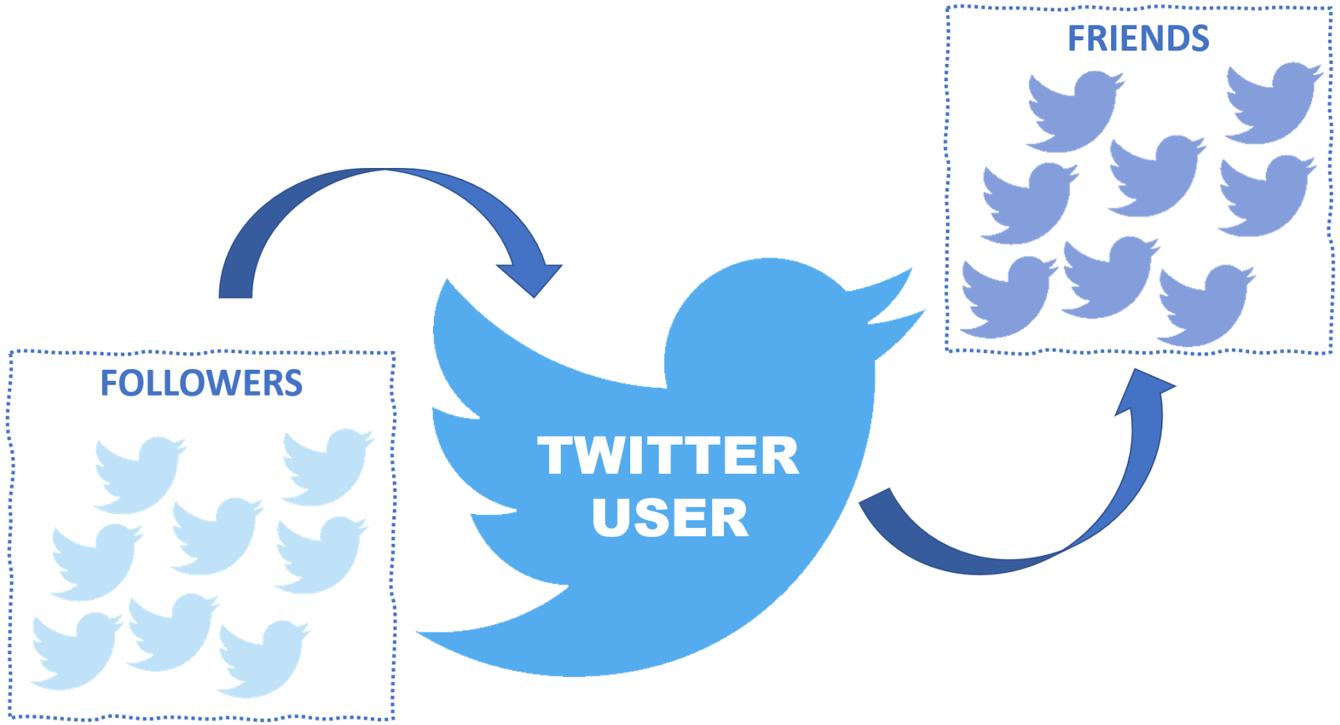
ANALYZING SOCIAL MEDIA DATA IN R

Vivek Vijayaraghavan
Data Science Coach

Lesson Overview

- `friends_count` and `followers_count` of a user
- Interpret golden ratio for brand promotion
- Twitter lists to identify users interested in a product

Followers vs friends



- Followers are users following a twitter user
- Friends are people a specific twitter user is following

Twitter follower vs following ratio

$$follower\ to\ following\ ratio = \frac{followers\_count}{friends\_count}$$

- Used by marketers to strategize promotions

Positive and negative ratios

- Positive ratio: more followers than friends for a user
- Negative ratio: more friends than followers for a user

Extract user information

```
# Search for 1000 tweets on #fitness
tweet_fit <- search_tweets("#fitness", n = 1000)
```

```
# Extract user information
user_fit <- users_data(tweet_fit)
```

Extract user information

```
# View column names of the user data
names(user_fit)
```

```
 [1] "user_id"                "screen_name"            "name"
 [4] "location"               "description"            "url"
 [7] "protected"              "followers_count"        "friends_count"
[10] "listed_count"           "statuses_count"         "favourites_count"
[13] "account_created_at"     "verified"               "profile_url"
[16] "profile_expanded_url"   "account_lang"           "profile_banner_url"
[19] "profile_background_url" "profile_image_url"
```

Extracting followers_count and friends_count

- Aggregate user screen names against followers and friends counts

```
# Aggregate screen_name, followers_count & friends_count
library(dplyr)
counts_df <- user_cos %>%
                group_by(screen_name) %>%
                summarize(follower = mean(followers_count),
                friend = mean(friends_count))
```

Extracting followers_count and friends_count

```
head(counts_df)
```

```
screen_name       follower     friend
<chr>               <dbl>       <dbl>
__seokjinnie124       209         454
_Aminata              623         523
_amsvn                167         126
_arweeennn            539         801
_asof_               1336         455
_blendac              833         195
```

The golden ratio

```
# Create a column to calculate the golden ratio
counts_df$ratio <- follow_df$follower/follow_df$friend
head(counts_df$ratio)
```

```
[1] 0.4603524 1.1912046 1.3253968 0.6729089 2.9362637 4.2717949
```

Explore users based on the ratio

- Examine golden ratios to understand user types

```
# Sort the data frame in decreasing order of follower count
counts_sort <- arrange(counts_df, desc(follower))
```

Explore users based on the ratio

```
# Select rows where the follower count is greater than 30000
counts_sort[counts_sort$follower>30000,]
```

```
screen_name    follower    friend    ratio
<chr>            <dbl>     <dbl>    <dbl>
mashable         9817699   2783     3528
MensHealthMag    4528421   1111     4076
Sophie_Choudry   2367827   157      15082
thewebmaster_    103936    6508     16
qwikad           92932     89557    1
Rharvley         90464     19484    5
SayWhenLA        68122     6680     10
```

- Medium to promote products on fitness

Explore users based on ratio

```
# Select rows where the follower count is less than 2000
counts_sort[counts_sort$follower<2000,]
```

```
screen_name    follower     friend    ratio
<chr>            <dbl>       <dbl>    <dbl>
workout_ehime    1960        1027       2
SardImperium     1932         256       8
Deem_Hoops       1912        1520       1
kaykay_inem      1890         443       4
bhealhty         1855        3066       1
```

- Position adverts on individual accounts for targeted promotion

User analysis with twitter lists

- Twitter list is a curated group of twitter accounts
- Twitter users subscribe to lists of interest

Extract lists subscribed to

```
# Get all lists "Playstation" subscribes to
lst_playstation <- lists_users("PlayStation")
lst_playstation[,1:4]
```

```
list_id          name                    uri                    subscriber_count
<chr>            <chr>                  <chr>                             <int>
58505230       PS Family          /PlayStation/lists/ps-family          136
4747423        GameDevelopers     /PlayStation/lists/gamedevelopers     467
2490894        gaming             /PlayStation/lists/gaming             658
```

Extract subscribers to a list

```
# Extract 100 subscribers of the "gaming" list owned by "Playstation"
list_PS_sub <- lists_subscribers(slug = "gaming", owner_user = "PlayStation", n = 100)
```

View screen names of subscribers

```
# View screen names of the subscribers
list_PS_sub$screen_name
```

```
 [1] "Morten83032201"  "ndugumr"          "snakejke25"       "souransb"
 [5] "WOLF210_Warrior" "Media_Hosseini"   "emangonz1"        "IMisticismo"
 [9] "yaqoob_alanzi"   "zehranur15hz"     "thegaybabadook"   "TheGladihater"
[13] "kothari_hemant"  "CortniBrown1"     "skillsearch"      "ItsPSFanatic"
[17] "Qinxus"          "leoohc"           "Anna30806004"     "ChrisBendeler"
[21] "The_SquareDonut" "DaelynRogers"     "geefromsp"        "dciceprincess"
[25] "lamperouge7"     "iam_sani_dole"    "ProjectModel3D"   "ElConfy16"
[29] "kotetsu804"      "mselfx32"         "grsharp8"         "SantiiBoss"
[33] "JaymoneyTv"      "_DragonStar_"     "prolazerxx"       "RealPosa"
[37] "YNHallak"        "chocochip0w0"     "mirandastweets"   "JARED101819"
```

User information of list subscribers

```
# Create a list of four screen names
users <- c("Morten83032201","ndugumr", "WOLF210_Warrior", "souransb")
```

```
# Extract user information
users_PS_gaming <- lookup_users(users)
```

```
user_id                status_id                created_at          screen_name
<chr>                    <chr>              <S3: POSIXct>            <chr>
1158299850573791233    1172604921121824769    2019-09-13 20:16:13    Morten83032201
894525207620321280     1183293767215992832    2019-10-13 08:09:53    ndugumr
325760816              1182867378293616640    2019-10-12 03:55:34    WOLF210_Warrior
469270931              511997829384904704     2014-09-16 21:59:29    souransb
```

Let's practice!

ANALYZING SOCIAL MEDIA DATA IN R

Twitter trends

ANALYZING SOCIAL MEDIA DATA IN R

Vivek Vijayaraghavan
Data Science Coach

Lesson Overview

- Understand twitter trends
- Extract trending topics
- Use trends for participation and engagement

What is a twitter trend?

- Keywords, events, or topics that are currently popular
- Discover hottest emerging topics of discussion
- Some trends include a hashtag
- Hashtags help search for trending conversations
- Location trends identify topics in a specific location

Leveraging the power of twitter trends

- Blend marketing messages with trending topic
- Trends help increase tweet engagements
- Travel portal tweets around "#TravelTuesday"

Extract worldwide trends

```
# Get overall current trending topics
trend_topics <- get_trends()
head(trend_topics$trend, 10)
```

```
 [1] "#madebygoogle"          "#????H??????"
 [3] "????"                   "Jennifer Aniston"
 [5] "#??????????"      "#FelizMartes"
 [7] "#G????????"        "????"
 [9] "????"                   "??????"
```

- More meaningful to extract trends around a specific region

Locations with current trends

```
# Extract locations of available twitter trends
trends_avail <- trends_available()
head(trends_avail)
```

```
name                    url                                parentid    country
<chr>                  <chr>                                 <int>     <chr>
Worldwide     http://where.yahooapis.com/v1/place/1
Winnipeg      http://where.yahooapis.com/v1/place/2972    23424775    Canada
Ottawa        http://where.yahooapis.com/v1/place/3369    23424775    Canada
Quebec        http://where.yahooapis.com/v1/place/3444    23424775    Canada
Montreal      http://where.yahooapis.com/v1/place/3534    23424775    Canada
Toronto       http://where.yahooapis.com/v1/place/4118    23424775    Canada
```

Trending topics by country

```
# Get trending topics in the US
gt_US <- get_trends("United States")
```

Trending topics by country

```
View(gt_US)
```

trend	url	promoted_content	query	tweet_volume	place
#TuesdayThought	http://twitter.com/search?q=%23TuesdayThought	NA	%23TuesdayThoughts	46255	United States
John Bolton	http://twitter.com/search?q=%22John+Bolton%22	NA	%22John+Bolton%22	90910	United States
#LeBronJames	http://twitter.com/search?q=%23LeBronJames	NA	%23LeBronJames	NA	United States
#RockHall2020	http://twitter.com/search?q=%23RockHall2020	NA	%23RockHall2020	NA	United States
#Fortnite2	http://twitter.com/search?q=%23Fortnite2	NA	%23Fortnite2	59476	United States
#IAmGrouchyWhe	http://twitter.com/search?q=%23IAmGrouchyWhe	NA	%23IAmGrouchyWhen	NA	United States

- Music video company can position promotions with "hashtagRockHall2020"

Trending topics by city

- Find trends in a specific city
- Attach tweets around relevant trend

```
# Get trending topics in New York
gt_city <- get_trends("New York")
```

Trending topics by city

```
head(gt_city)
```

```
trend                          url                          promoted_content
<chr>                          <chr>                               <lgl>
Lions              http://twitter.com/search?q=Lions                   NA
Green Bay          http://twitter.com/search?q=%22Green+Bay%22          NA
#DETvsGB           http://twitter.com/search?q=%23DETvsGB               NA
LeBron             http://twitter.com/search?q=LeBron                   NA
Aaron Rodgers      http://twitter.com/search?q=%22Aaron+Rodgers%22      NA
#90DayFiance       http://twitter.com/search?q=%2390DayFiance           NA
```

- Company promoting basketball merchandise could leverage this trend

Most tweeted trends

- `tweet_volume` has count of tweets made on a trending topic
- It is available for some trends only
- Identify trends that are most tweeted

Most tweeted trends

```
# Aggregate trends and tweet volumes
library(dplyr)
trend_df <- gt_city %>%
              group_by(trend) %>%
              summarize(tweet_vol = mean(tweet_volume))
```

Most tweeted trends

```
head(trend_df)
```

```
trend                  tweet_vol
<chr>                    <dbl>
#90DayFiance             14375
#acefamilyisoverparty    12760
#ascendwithme               NA
#bbcon2019                  NA
#bookbirthday               NA
#DemDebate               18928
```

Most tweeted trends

```
# Sort data frame on descending order of tweet volumes
trend_df_sort <- arrange(trend_df, desc(tweet_vol))
```

Most tweeted trends

```
# View the most tweeted trends
head(trend_df_sort)
```

```
trend              tweet_vol
<chr>               <dbl>
LeBron              298302
Lions               267945
Columbus Day        135014
John Bolton         118933
#DETvsGB             67197
#TuesdayThoughts     63259
```

- Travel company can promote holiday packages around "Columbus Day"

Let's practice!

ANALYZING SOCIAL MEDIA DATA IN R

Plotting twitter data over time

ANALYZING SOCIAL MEDIA DATA IN R

Vivek Vijayaraghavan
Data Science Coach

Lesson overview

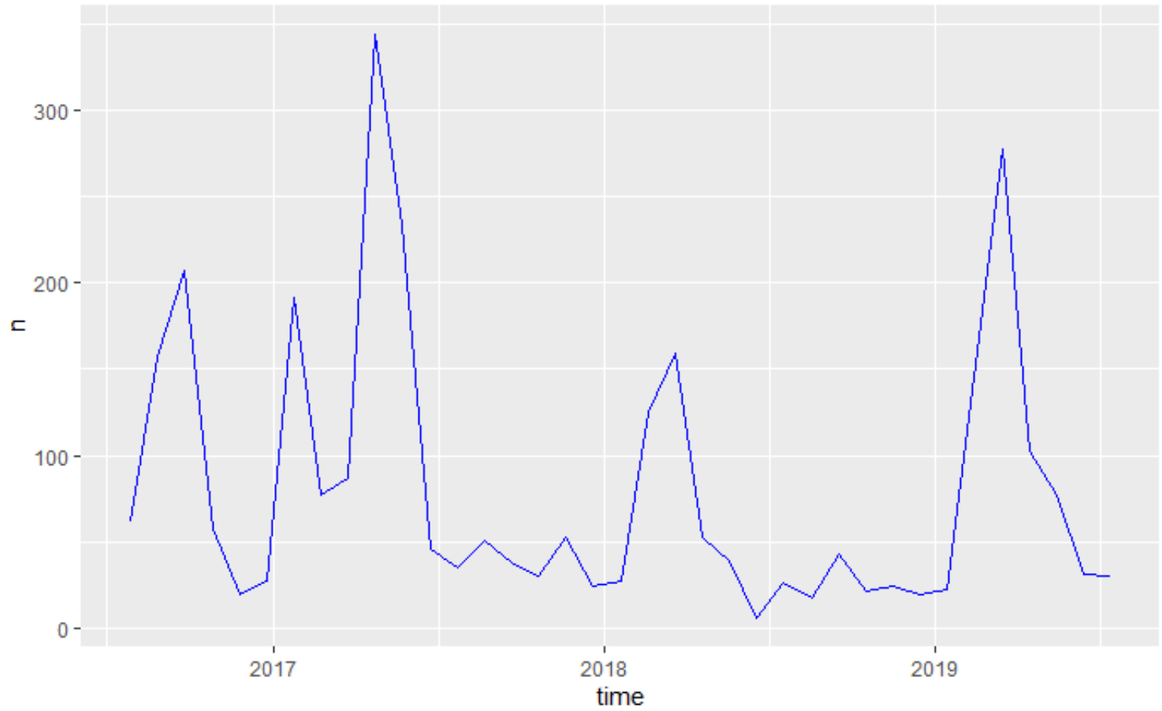
- Time series data
- Create time series objects and plots
- Visualize frequency of tweets over time
- Compare brand salience of two brands

Brand salience is the extent to which a brand is spoken about by potential customers.

The volume of tweets is a strong indicator of brand salience

Time series data

- Series of data points indexed over time
- Visualize frequency of tweets



Extracting tweets for time series analysis

- Extract tweets for time series analysis using `search_tweets()`

```
library(rtweet)
```

```
# Extract tweets on "#google" using search_tweets()
search_tweets("#google", n = 18000, include_rts = FALSE)
```

Extracted tweet data

```
status_id                created_at          screen_name
<chr>                  <S3: POSIXct>             <chr>
1164921105066463232    2019-08-23 15:23:29    catapanoannal
1164921037143699456    2019-08-23 15:23:13    STARBEXPLORE
1164920927341039621    2019-08-23 15:22:46    indra_susanto
1164920898475794435    2019-08-23 15:22:40    virfice
1164920877940482048    2019-08-23 15:22:35    KnowledgeNile
1164920647962832897    2019-08-23 15:21:40    mahomes_tech
```

- `created_at` has the timestamp of the tweets

Visualize frequency of tweets

- Monitor overall engagement for a product
- Tweet frequencies: insights on interest level

Visualize tweet frequency

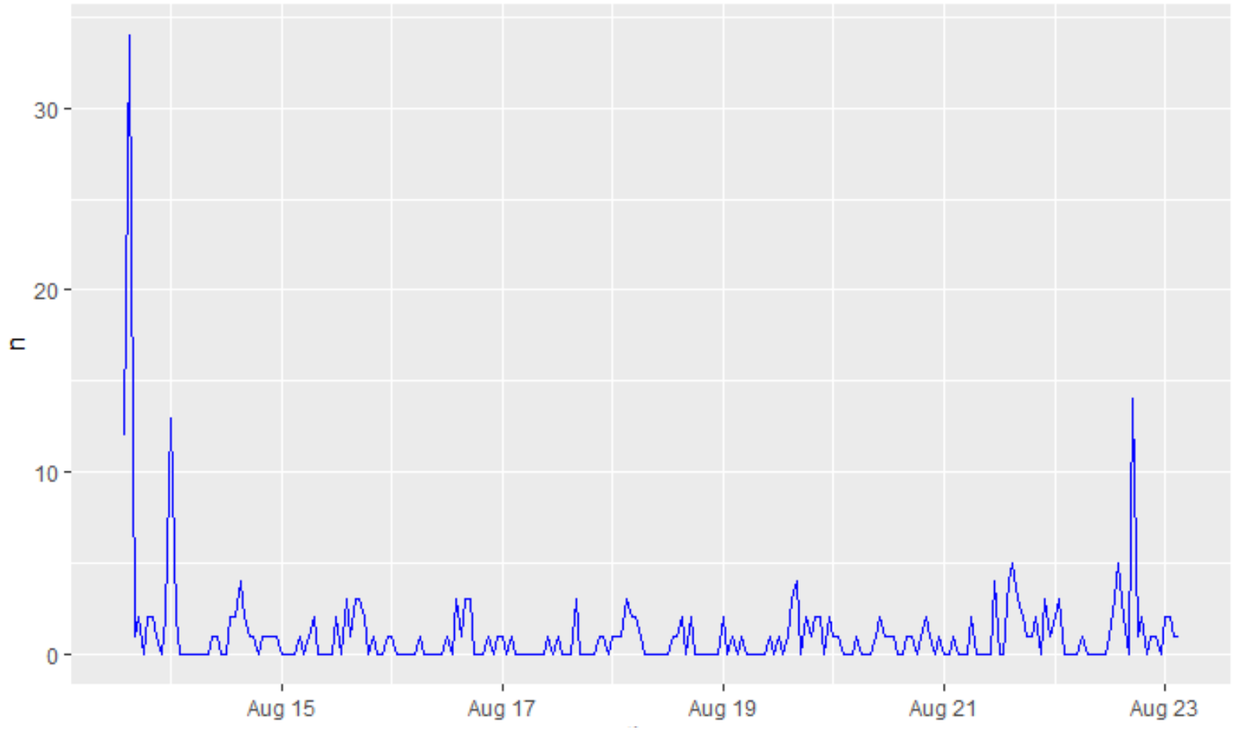
```
# Extract tweets on "#camry" using search_tweets()
camry_st <- search_tweets("#camry", n = 18000, include_rts = FALSE)
```

Visualize tweet frequency

```
created_at              screen_name                    text
<S3: POSIXct>             <chr>                        <chr>
2019-08-23 03:29:58     dromru         Toyota Camry 2019 <U+0433><U+043E><U+0434><U+0
2019-08-23 02:59:04     NusTrivia      Sportier 2020 Toyota Camry TRD to cost $31,995
2019-08-22 18:09:06     NusTrivia      2020 Toyota Camry TRD Costs $31,995, It’s The
2019-08-23 01:56:51     RaitisRides    ALL NEW 2020 Toyota Avalon is coming to R
2019-08-23 01:17:36     jhooie         I have to say, when I finally settled down tod
```

Create time series plot

```
# Create a time series plot
ts_plot(camry_st, by = "hours", color = "blue")
```



Compare frequency of tweets

- Volume of tweets posted is a strong indicator of brand salience
- Compare the brand salience of Tesla and Camry

VS

Compare frequency of tweets

- Convert the tweets extracted on Camry into a time series object
- Time series object contains aggregated frequency of tweets over a time interval

```
# Convert tweet data into a time series object
camry_ts <- ts_data(camry_st, by = 'hours')
head(camry_ts)
```

```
time                      n
<S3: POSIXct>         <int>
2019-08-13 14:00:00      12
2019-08-13 15:00:00      34
2019-08-13 16:00:00       1
2019-08-13 17:00:00       2
```

Compare frequency of tweets

```
# Rename the two columns in the time series object
names(camry_ts) <- c("time", "camry_n")
```

```
head(camry_ts)
```

```
time                 camry_n
<S3: POSIXct>         <int>
2019-08-13 14:00:00      12
2019-08-13 15:00:00      34
2019-08-13 16:00:00       1
2019-08-13 17:00:00       2
```

Compare frequency of tweets

```
tesla_st <- search_tweets("#tesla", n = 18000, include_rts = FALSE)
tesla_ts <- ts_data(tesla_st, by = 'hours')
```

```
names(tesla_ts) <- c("time", "tesla_n")
head(tesla_ts)
```

```
time                tesla_n
<S3: POSIXct>        <int>
2019-08-13 13:00:00    17
2019-08-13 14:00:00    58
2019-08-13 15:00:00    38
2019-08-13 16:00:00    32
```

Compare frequency of tweets

```
# Merge the two time series objects and retain "time" column
merged_df <- merge(tesla_ts, camry_ts, by = "time", all = TRUE)
head(merged_df)
```

```
time                 tesla_n    camry_n
<S3:POSIXct>           <int>     <int>
2019-08-13 13:00:00    17         NA
2019-08-13 14:00:00    58         12
2019-08-13 15:00:00    38         34
2019-08-13 16:00:00    32          1
```

Compare frequency of tweets

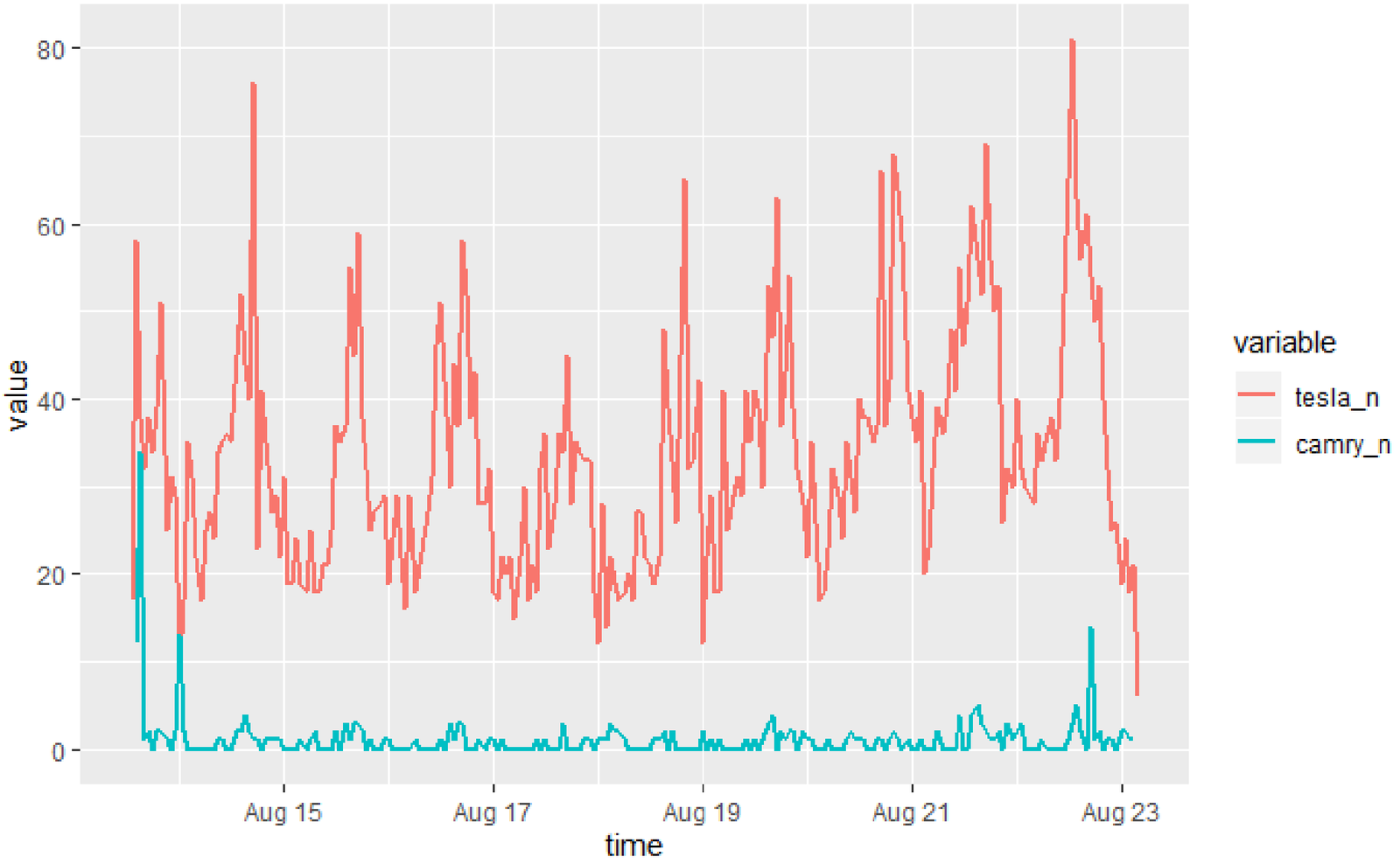
```
# Stack the tweet frequency columns using melt() function
library(reshape)
melt_df <- melt(merged_df, na.rm = TRUE, id.vars = "time")
head(melt_df)
```

```
time                 variable   value
<S3: POSIXct>         <fctr>    <int>
2019-08-13 13:00:00   tesla_n    17
2019-08-13 14:00:00   tesla_n    58
2019-08-13 15:00:00   tesla_n    38
2019-08-13 16:00:00   tesla_n    32
2019-08-13 17:00:00   tesla_n    38
2019-08-13 18:00:00   tesla_n    34
```

Compare frequency of tweets

```
# Plot frequency of tweets on Camry and Tesla
ggplot(data = melt_df,
       aes(x = time, y = value, col = variable)) +
       geom_line(lwd = 0.8)
```

The comparison plot



Let's practice!

ANALYZING SOCIAL MEDIA DATA IN R