STAT 231: Problem Set 6B

Matthew Perkins

due by 10 PM on Friday, April 2

This homework assignment is designed to help you further ingest, practice, and expand upon the material covered in class over the past week(s). You are encouraged to work with other students, but all code and text must be written by you, and you must indicate below who you discussed the assignment with (if anyone).

Steps to proceed:

1. In RStudio, go to File > Open Project, navigate to the folder with the course-content repo, select the course-content project (course-content.Rproj), and click "Open"
2. Pull the course-content repo (e.g. using the blue-ish down arrow in the Git tab in upper right window)
3. Copy ps6B.Rmd from the course repo to your repo (see page 6 of the GitHub Classroom Guide for Stat231 if needed)
4. Close the course-content repo project in RStudio
5. Open YOUR repo project in RStudio
6. In the ps6B.Rmd file in YOUR repo, replace "YOUR NAME HERE" with your name
7. Add in your responses, committing and pushing to YOUR repo in appropriate places along the way
8. Run "Knit PDF"
9. Upload the pdf to Gradescope. Don't forget to select which of your pages are associated with each problem. *You will not get credit for work on unassigned pages (e.g., if you only selected the first page but your solution spans two pages, you would lose points for any part on the second page that the grader can't see).*

If you discussed this assignment with any of your peers, please list who here:

ANSWER:

Trump Tweets

David Robinson, Chief Data Scientist at DataCamp, wrote a blog post "Text analysis of Trump's tweets confirms he writes only the (angrier) Android half".

He provides a dataset with over 1,500 tweets from the account realDonaldTrump between 12/14/2015 and 8/8/2016. We'll use this dataset to explore the tweeting behavior of realDonaldTrump during this time period.

First, read in the file. Note that there is a `TwitteR` package which provides an interface to the Twitter web API. We'll use this R dataset David created using that package so that you don't have to set up Twitter authentication.

```
load(url("http://varianceexplained.org/files/trump_tweets_df.rda"))
```

A little wrangling to warm-up

1a. There are a number of variables in the dataset we won't need.

- First, confirm that all the observations in the dataset are from the screen-name `realDonaldTrump`.
- Then, create a new dataset called `tweets` that only includes the following variables:
- `text`
- `created`
- `statusSource`

```
trump_tweets_df
```

```
## # A tibble: 1,512 x 16
##    text          favorited favoriteCount replyToSN created             truncated
##    <chr>         <lgl>             <dbl> <chr>     <dttm>              <lgl>
##  1 "My economic~ FALSE              9214 <NA>      2016-08-08 15:20:44 FALSE
##  2 "Join me in ~ FALSE              6981 <NA>      2016-08-08 13:28:20 FALSE
##  3 "#ICYMI: \"W~ FALSE             15724 <NA>      2016-08-08 00:05:54 FALSE
##  4 "Michael Mor~ FALSE             19837 <NA>      2016-08-07 23:09:08 FALSE
##  5 "The media i~ FALSE             34051 <NA>      2016-08-07 21:31:46 FALSE
##  6 "I see where~ FALSE             29831 <NA>      2016-08-07 13:49:29 FALSE
##  7 "Thank you W~ FALSE             19223 <NA>      2016-08-07 02:19:37 FALSE
##  8 ".@Larry_Kud~ FALSE             19543 <NA>      2016-08-07 02:03:39 FALSE
##  9 "I am not ju~ FALSE             75488 <NA>      2016-08-07 01:53:45 FALSE
## 10 "#CrookedHil~ FALSE             23661 <NA>      2016-08-06 20:04:08 FALSE
## # ... with 1,502 more rows, and 10 more variables: replyToSID <lgl>, id <chr>,
## #   replyToUID <chr>, statusSource <chr>, screenName <chr>, retweetCount <dbl>,
## #   isRetweet <lgl>, retweeted <lgl>, longitude <chr>, latitude <chr>
```

```
tweets_test <- trump_tweets_df %>%
  filter(screenName == 'realDonaldTrump')
tweets_test #same number of rows, so must be all RDT tweets
```

```
## # A tibble: 1,512 x 16
##    text          favorited favoriteCount replyToSN created             truncated
##    <chr>         <lgl>             <dbl> <chr>     <dttm>              <lgl>
##  1 "My economic~ FALSE              9214 <NA>      2016-08-08 15:20:44 FALSE
##  2 "Join me in ~ FALSE              6981 <NA>      2016-08-08 13:28:20 FALSE
##  3 "#ICYMI: \"W~ FALSE             15724 <NA>      2016-08-08 00:05:54 FALSE
##  4 "Michael Mor~ FALSE             19837 <NA>      2016-08-07 23:09:08 FALSE
##  5 "The media i~ FALSE             34051 <NA>      2016-08-07 21:31:46 FALSE
##  6 "I see where~ FALSE             29831 <NA>      2016-08-07 13:49:29 FALSE
##  7 "Thank you W~ FALSE             19223 <NA>      2016-08-07 02:19:37 FALSE
##  8 ".@Larry_Kud~ FALSE             19543 <NA>      2016-08-07 02:03:39 FALSE
##  9 "I am not ju~ FALSE             75488 <NA>      2016-08-07 01:53:45 FALSE
## 10 "#CrookedHil~ FALSE             23661 <NA>      2016-08-06 20:04:08 FALSE
## # ... with 1,502 more rows, and 10 more variables: replyToSID <lgl>, id <chr>,
## #   replyToUID <chr>, statusSource <chr>, screenName <chr>, retweetCount <dbl>,
## #   isRetweet <lgl>, retweeted <lgl>, longitude <chr>, latitude <chr>
```

```
tweets <- trump_tweets_df %>%
  select(text, created, statusSource)
tweets
```

```
## # A tibble: 1,512 x 3
##    text                        created             statusSource
##    <chr>                       <dttm>              <chr>
##  1 "My economic policy speec~ 2016-08-08 15:20:44 "<a href=\"http://twitter.com~
##  2 "Join me in Fayetteville,~ 2016-08-08 13:28:20 "<a href=\"http://twitter.com~
##  3 "#ICYMI: \"Will Media Apo~ 2016-08-08 00:05:54 "<a href=\"http://twitter.com~
##  4 "Michael Morell, the ligh~ 2016-08-07 23:09:08 "<a href=\"http://twitter.com~
##  5 "The media is going crazy~ 2016-08-07 21:31:46 "<a href=\"http://twitter.com~
##  6 "I see where Mayor Stepha~ 2016-08-07 13:49:29 "<a href=\"http://twitter.com~
##  7 "Thank you Windham, New H~ 2016-08-07 02:19:37 "<a href=\"http://twitter.com~
##  8 ".@Larry_Kudlow - 'Donald~ 2016-08-07 02:03:39 "<a href=\"http://twitter.com~
##  9 "I am not just running ag~ 2016-08-07 01:53:45 "<a href=\"http://twitter.com~
## 10 "#CrookedHillary is not f~ 2016-08-06 20:04:08 "<a href=\"http://twitter.com~
## # ... with 1,502 more rows
```

1b. Using the `statusSource` variable, compute the number of tweets from each source. How many different sources are there? How often are each used?

> ANSWER: There are 5 different sources, with android used for 762 tweets, iphone used for 628 tweets, the web client used for 120 tweets, and 1 tweet each via ipad and instagram.

```
tweetSource <- tweets %>%
  pivot_wider(names_from = statusSource, values_from = text) %>%
  clean_names() %>%
  rename(android = a_href_http_twitter_com_download_android_rel_nofollow_twitter_for_android_a,
         iphone = a_href_http_twitter_com_download_iphone_rel_nofollow_twitter_for_i_phone_a,
         web = a_href_http_twitter_com_rel_nofollow_twitter_web_client_a,
         ipad = a_href_http_twitter_com_number_download_ipad_rel_nofollow_twitter_for_i_pad_a,
         insta = a_href_http_instagram_com_rel_nofollow_instagram_a) %>%
  summarise(android_tweets = sum(!is.na(android)),
            iphone_tweets = sum(!is.na(iphone)),
            web_tweets = sum(!is.na(web)),
            ipad_tweets = sum(!is.na(ipad)),
            insta_tweets = sum(!is.na(insta)))
tweetSource
```

```
## # A tibble: 1 x 5
##   android_tweets iphone_tweets web_tweets ipad_tweets insta_tweets
##            <int>         <int>      <int>       <int>        <int>
## 1            762           628        120           1            1
```

1c. We're going to compare the language used between the Android and iPhone sources, so only want to keep tweets coming from those sources. Explain what the `extract` function (from the `tidyverse` package) is doing below. Include in your own words what each argument is doing. (Note that "regex" stands for "regular expression".)

> ANSWER: The extract function is isolating which platform a tweet is from based on the information contained within the statusSource column. First, col = statusSource specifies the statusSource column. into = "source" specifies the name of the new column to be created. Regex is used to extract the desired sourceinformation, capturing charachers after twitter for and before the <. Remove = FALSE tells R to leave the original column in the dataframe.

```
tweets2 <- tweets %>%
  extract(col = statusSource, into = "source"
          , regex = "Twitter for (.*)<"
          , remove = FALSE) %>%
  filter(source %in% c("Android", "iPhone"))
tweets2
```

```
## # A tibble: 1,390 x 4
##    text                  created             statusSource                source
##    <chr>                 <dttm>              <chr>                       <chr>
##  1 "My economic policy s~ 2016-08-08 15:20:44 "<a href=\"http://twitter.~ Andro~
##  2 "Join me in Fayettevi~ 2016-08-08 13:28:20 "<a href=\"http://twitter.~ iPhone
##  3 "#ICYMI: \"Will Media~ 2016-08-08 00:05:54 "<a href=\"http://twitter.~ iPhone
##  4 "Michael Morell, the ~ 2016-08-07 23:09:08 "<a href=\"http://twitter.~ Andro~
##  5 "The media is going c~ 2016-08-07 21:31:46 "<a href=\"http://twitter.~ Andro~
##  6 "I see where Mayor St~ 2016-08-07 13:49:29 "<a href=\"http://twitter.~ Andro~
##  7 "Thank you Windham, N~ 2016-08-07 02:19:37 "<a href=\"http://twitter.~ iPhone
##  8 ".@Larry_Kudlow - 'Do~ 2016-08-07 02:03:39 "<a href=\"http://twitter.~ iPhone
##  9 "I am not just runnin~ 2016-08-07 01:53:45 "<a href=\"http://twitter.~ Andro~
## 10 "#CrookedHillary is n~ 2016-08-06 20:04:08 "<a href=\"http://twitter.~ iPhone
## # ... with 1,380 more rows
```

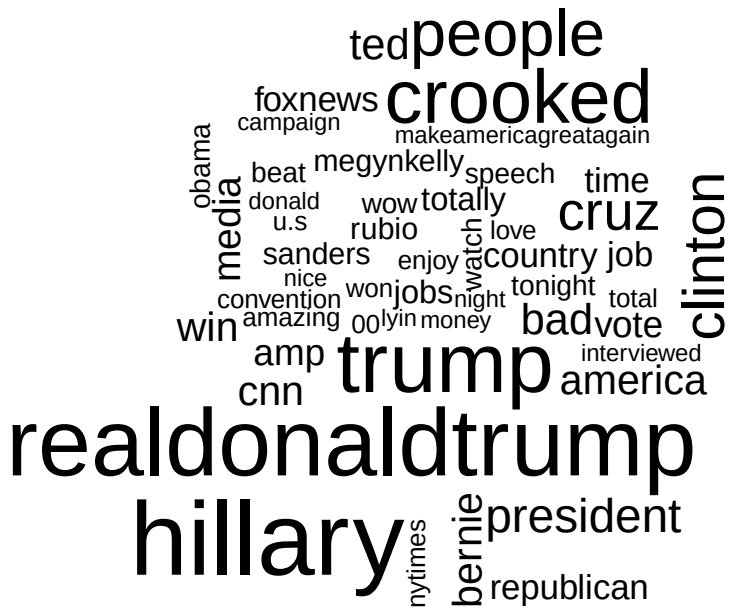
How does the language of the tweets differ by source?

2a. Create a word cloud for the top 50 words used in tweets sent from the Android. Create a second word cloud for the top 50 words used in tweets sent from the iPhone. How do these word clouds compare? (Are there some common words frequently used from both sources? Are the most common words different between the sources?)

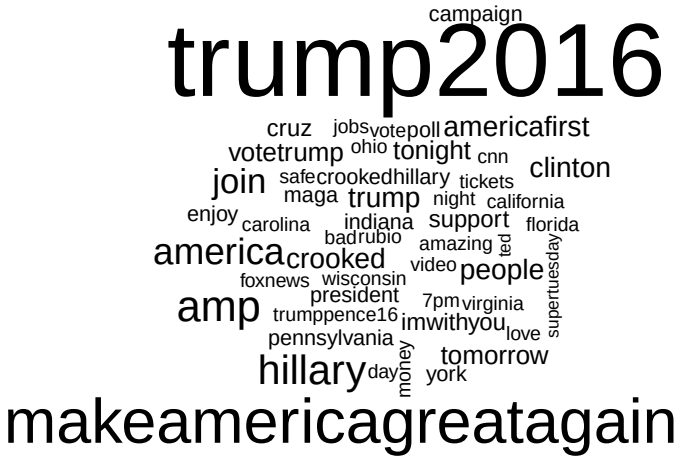
Don't forget to remove stop words before creating the word cloud. Also remove the terms "https" and "t.co".

> ANSWER: The tweets from the android and the iphone have many distinct properties. While both share many words in common such as hillary and maga, the frequencies are quite different. For the android tweets which Trump presumably sends himself, Hillary is at the top of the most common words along with other political rivals, whereas the iPhone tweets appear to be more focused toward promoting the campaign with Trump2016 and maga as the clear frontrunners.

```
tweets_words <- tweets2 %>%
  unnest_tokens(output = word, input = text)

tweets_words2 <- tweets_words %>%
  filter(!word %in% stop_words$word) %>%
  filter(word != "t.co", word != "https" & source == "Android")

word_frequencies <- tweets_words2 %>%
  count(word, sort = TRUE)

word_frequencies %>%
  with(wordcloud(words = word, freq = n, max.words=50))
```




```
tweets_words3 <- tweets_words %>%
  filter(!word %in% stop_words$word) %>%
  filter(word != "t.co", word != "https" & source == "iPhone")

word_frequencies2 <- tweets_words3 %>%
  count(word, sort = TRUE)

word_frequencies2 %>%
  with(wordcloud(words = word, freq = n, max.words=50))
```


campaign
trump2016
cruz
jobs
votepoll
americafirst
votetrump
ohio
tonight
cnn
clinton
join
safecrookedhillary
tickets
maga
trump
night
california
enjoy
carolina
indiana
support
florida
badrubio
amazing
ted
america
crooked
video
people
supertuesday
foxnews
wisconsin
president
amp
7pm
virginia
trumppence16
imwithyou
love
pennsylvania
tomorrow
hillary
day
money
york
makeamericagreatagain

2b. Create a visualization that compares the top 10 *bigrams* appearing in tweets by each source (that is, facet by source). After creating a dataset with one row per bigram, you should remove any rows that contain a stop word within the bigram.

How do the top used bigrams compare between the two sources?

> ANSWER: There is a fair amount of overlap among bigrams (e.g. crooked hillary and hillary clinton are near the top on both), but the iPhone tweets show a much stronger focus on slogans such as makeamericagreatagain and trump2016.

```
data(stop_words)

tweets2
```

```
## # A tibble: 1,390 x 4
##    text                   created             statusSource                source
##    <chr>                  <dttm>              <chr>                       <chr>
##  1 "My economic policy s~ 2016-08-08 15:20:44 "<a href=\"http://twitter.~ Andro~
##  2 "Join me in Fayettevi~ 2016-08-08 13:28:20 "<a href=\"http://twitter.~ iPhone
##  3 "#ICYMI: \"Will Media~ 2016-08-08 00:05:54 "<a href=\"http://twitter.~ iPhone
##  4 "Michael Morell, the ~ 2016-08-07 23:09:08 "<a href=\"http://twitter.~ Andro~
##  5 "The media is going c~ 2016-08-07 21:31:46 "<a href=\"http://twitter.~ Andro~
##  6 "I see where Mayor St~ 2016-08-07 13:49:29 "<a href=\"http://twitter.~ Andro~
##  7 "Thank you Windham, N~ 2016-08-07 02:19:37 "<a href=\"http://twitter.~ iPhone
##  8 ".@Larry_Kudlow - 'Do~ 2016-08-07 02:03:39 "<a href=\"http://twitter.~ iPhone
##  9 "I am not just runnin~ 2016-08-07 01:53:45 "<a href=\"http://twitter.~ Andro~
## 10 "#CrookedHillary is n~ 2016-08-06 20:04:08 "<a href=\"http://twitter.~ iPhone
## # ... with 1,380 more rows
```

```
tweets_bigrams_android <- tweets2 %>%
  filter(source == "Android") %>%
  unnest_tokens(output = bigram, input = text
                , token = "ngrams", n = 2) %>%
  count(bigram, sort = TRUE) %>%
  slice(1:100)

my_regex <- regex(paste("\\b", stop_words$word, "\\b", sep = "", collapse = "|"))

tweets_android_filtered <- tweets_bigrams_android %>%
  filter(!str_detect(bigram, my_regex), !str_detect(bigram, "https")) %>%
  slice(1:10)

tweets_android_filtered
```

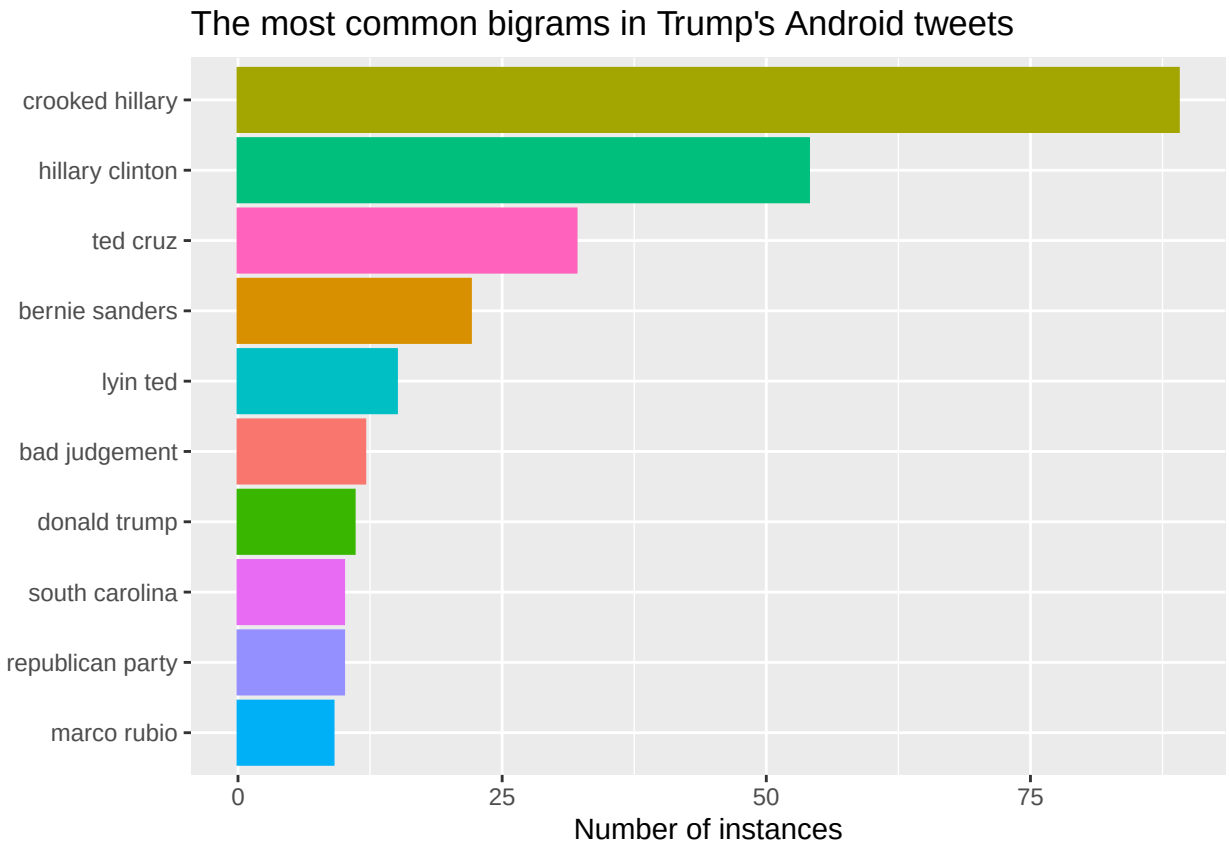
```
## # A tibble: 10 x 2
##    bigram               n
##    <chr>            <int>
##  1 crooked hillary     89
##  2 hillary clinton     54
##  3 ted cruz            32
##  4 bernie sanders      22
##  5 lyin ted            15
##  6 bad judgement       12
```

```
##  7 donald trump        11
##  8 republican party    10
##  9 south carolina      10
## 10 marco rubio          9
```

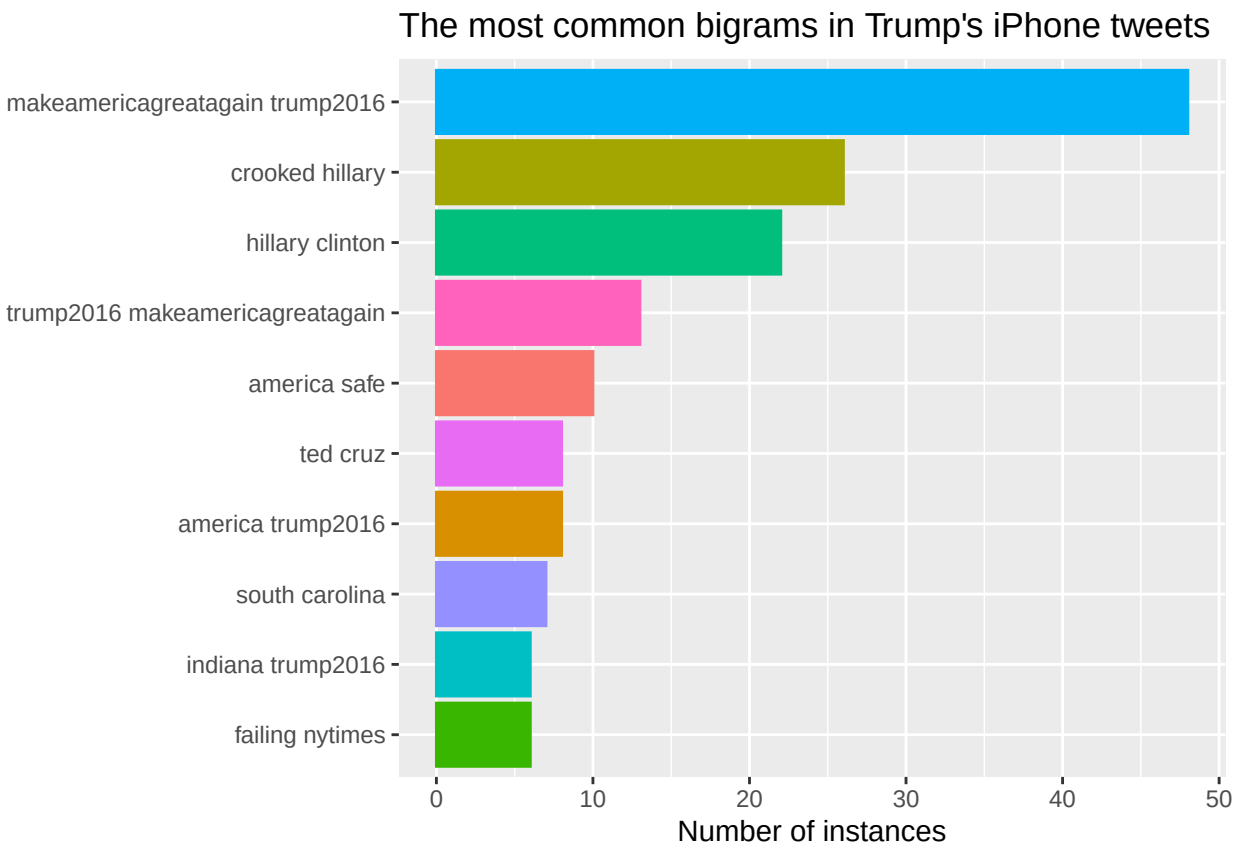
```
tweets_bigrams_iphone <- tweets2 %>%
  filter(source == "iPhone") %>%
  unnest_tokens(output = bigram, input = text
                , token = "ngrams", n = 2) %>%
  count(bigram, sort = TRUE) %>%
  slice(1:100)

tweets_iphone_filtered <- tweets_bigrams_iphone %>%
  filter(!str_detect(bigram, my_regex), !str_detect(bigram, "https")) %>%
  slice(1:10)

tweets_iphone_filtered
```

```
## # A tibble: 10 x 2
##    bigram                              n
##    <chr>                           <int>
##  1 makeamericagreatagain trump2016    48
##  2 crooked hillary                    26
##  3 hillary clinton                    22
##  4 trump2016 makeamericagreatagain    13
##  5 america safe                       10
##  6 america trump2016                   8
##  7 ted cruz                            8
##  8 south carolina                      7
##  9 failing nytimes                     6
## 10 indiana trump2016                   6
```

```
tweets_android_filtered %>%
  ggplot(aes(x = reorder(bigram,n), y = n, color = bigram, fill=bigram)) +
  geom_col() +
  xlab(NULL) +
  coord_flip() +
  labs(y = "Number of instances"
       , title="The most common bigrams in Trump's Android tweets") +
  guides(color = "none", fill = "none")
```




```
tweets_iphone_filtered %>%
  ggplot(aes(x = reorder(bigram,n), y = n, color = bigram, fill=bigram)) +
  geom_col() +
  xlab(NULL) +
  coord_flip() +
  labs(y = "Number of instances"
       , title="The most common bigrams in Trump's iPhone tweets") +
  guides(color = "none", fill = "none")
```


The most common bigrams in Trump's iPhone tweets
makeamericagreatagain trump2016
crooked hillary
hillary clinton
trump2016 makeamericagreatagain
america safe
ted cruz
america trump2016
south carolina
indiana trump2016
failing nytimes
0
10
20
30
40
50
Number of instances

2c. Consider the sentiment. Compute the proportion of words among the tweets within each source classified as "angry" and the proportion of words classified as "joy" based on the NRC lexicon. How does the proportion of "angry" and "joy" words compare between the two sources? What about "positive" and "negative" words?

> ANSWER: The proportion of anger and negative words according to nrc word sentiments is higher among android tweets than iphone tweets. Anger words make up ~2.3% of android tweets and only 1.7% of iphone. Joy values are roughly the same in both. Likewise, Negative words are ~4.2% of android tweets and only 2.7% of iphone. There are also slightly fewer positive words in android tweets, but this difference is fairly small.

```
nrc_lexicon <- get_sentiments("nrc")

nrc_angry <- nrc_lexicon %>%
  filter(sentiment == "anger")

nrc_joy <- nrc_lexicon %>%
  filter(sentiment == "joy")

tweets_words_android <- tweets2 %>%
  unnest_tokens(output = word, input = text) %>%
  filter(source == "Android")

tweets_words_android
```

```
## # A tibble: 15,764 x 4
##    created             statusSource                                  source word
##    <dttm>              <chr>                                         <chr>  <chr>
##  1 2016-08-08 15:20:44 "<a href=\"http://twitter.com/download/and~ Andro~ my
##  2 2016-08-08 15:20:44 "<a href=\"http://twitter.com/download/and~ Andro~ econo~
##  3 2016-08-08 15:20:44 "<a href=\"http://twitter.com/download/and~ Andro~ policy
##  4 2016-08-08 15:20:44 "<a href=\"http://twitter.com/download/and~ Andro~ speech
##  5 2016-08-08 15:20:44 "<a href=\"http://twitter.com/download/and~ Andro~ will
##  6 2016-08-08 15:20:44 "<a href=\"http://twitter.com/download/and~ Andro~ be
##  7 2016-08-08 15:20:44 "<a href=\"http://twitter.com/download/and~ Andro~ carri~
##  8 2016-08-08 15:20:44 "<a href=\"http://twitter.com/download/and~ Andro~ live
##  9 2016-08-08 15:20:44 "<a href=\"http://twitter.com/download/and~ Andro~ at
## 10 2016-08-08 15:20:44 "<a href=\"http://twitter.com/download/and~ Andro~ 12
## # ... with 15,754 more rows
```

```
total_android = 15764

tweets_words_android_angry <- tweets_words_android %>%
  inner_join(nrc_angry) %>%
  count(word, sort = TRUE, name = 'angry')
```

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

```
tweets_words_android_angry %>%
  summarize(sum(angry)/total_android)
```

```
## # A tibble: 1 x 1
```

```
##   `sum(angry)/total_android`
##                        <dbl>
## 1                     0.0230
```

```
tweets_words_android_joy <- tweets_words_android %>%
  inner_join(nrc_joy) %>%
  count(word, sort = TRUE, name = 'joy')
```

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

```
tweets_words_android_joy %>%
  summarize(sum(joy)/total_android)
```

```
## # A tibble: 1 x 1
##   `sum(joy)/total_android`
##                      <dbl>
## 1                   0.0185
```

```
tweets_words_iphone <- tweets2 %>%
  unnest_tokens(output = word, input = text) %>%
  filter(source == "iPhone")

tweets_words_iphone
```

```
## # A tibble: 9,632 x 4
##    created             statusSource                                source word
##    <dttm>              <chr>                                       <chr>  <chr>
##  1 2016-08-08 13:28:20 "<a href=\"http://twitter.com/download/i~ iPhone join
##  2 2016-08-08 13:28:20 "<a href=\"http://twitter.com/download/i~ iPhone me
##  3 2016-08-08 13:28:20 "<a href=\"http://twitter.com/download/i~ iPhone in
##  4 2016-08-08 13:28:20 "<a href=\"http://twitter.com/download/i~ iPhone fayette~
##  5 2016-08-08 13:28:20 "<a href=\"http://twitter.com/download/i~ iPhone north
##  6 2016-08-08 13:28:20 "<a href=\"http://twitter.com/download/i~ iPhone carolina
##  7 2016-08-08 13:28:20 "<a href=\"http://twitter.com/download/i~ iPhone tomorrow
##  8 2016-08-08 13:28:20 "<a href=\"http://twitter.com/download/i~ iPhone evening
##  9 2016-08-08 13:28:20 "<a href=\"http://twitter.com/download/i~ iPhone at
## 10 2016-08-08 13:28:20 "<a href=\"http://twitter.com/download/i~ iPhone 6pm
## # ... with 9,622 more rows
```

```
total_iphone = 9632

tweets_words_iphone_angry <- tweets_words_iphone %>%
  inner_join(nrc_angry) %>%
  count(word, sort = TRUE, name = 'angry')
```

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

```
tweets_words_iphone_angry %>%
  summarize(sum(angry)/total_iphone)
```

```
## # A tibble: 1 x 1
##   `sum(angry)/total_iphone`
##                       <dbl>
## 1                    0.0176
```

```
tweets_words_iphone_joy <- tweets_words_iphone %>%
  inner_join(nrc_joy) %>%
  count(word, sort = TRUE, name = 'joy')
```

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

```
tweets_words_iphone_joy %>%
  summarize(sum(joy)/total_iphone)
```

```
## # A tibble: 1 x 1
##   `sum(joy)/total_iphone`
##                     <dbl>
## 1                  0.0183
```

```
nrc_negative <- nrc_lexicon %>%
  filter(sentiment == "negative")

nrc_positive <- nrc_lexicon %>%
  filter(sentiment == "positive")


tweets_words_android_negative <- tweets_words_android %>%
  inner_join(nrc_negative) %>%
  count(word, sort = TRUE, name = 'negative')
```

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

```
tweets_words_android_negative %>%
  summarize(sum(negative)/total_android)
```

```
## # A tibble: 1 x 1
##   `sum(negative)/total_android`
##                           <dbl>
## 1                        0.0420
```

```
tweets_words_android_positive <- tweets_words_android %>%
  inner_join(nrc_positive) %>%
  count(word, sort = TRUE, name = 'positive')
```

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

```
tweets_words_android_positive %>%
  summarize(sum(positive)/total_android)
```

```
## # A tibble: 1 x 1
##   `sum(positive)/total_android`
##                           <dbl>
## 1                        0.0504
```

```
tweets_words_iphone_negative <- tweets_words_iphone %>%
  inner_join(nrc_negative) %>%
  count(word, sort = TRUE, name = 'negative')
```

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

```
tweets_words_iphone_negative %>%
  summarize(sum(negative)/total_iphone)
```

```
## # A tibble: 1 x 1
##   `sum(negative)/total_iphone`
##                          <dbl>
## 1                       0.0276
```

```
tweets_words_iphone_positive <- tweets_words_iphone %>%
  inner_join(nrc_positive) %>%
  count(word, sort = TRUE, name = 'positive')
```

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

```
tweets_words_iphone_positive %>%
  summarize(sum(positive)/total_iphone)
```

```
## # A tibble: 1 x 1
##   `sum(positive)/total_iphone`
##                          <dbl>
## 1                       0.0466
```

2d. Lastly, based on your responses above, do you think there is evidence to support Robinson's claim that Trump only writes the (angrier) Android half of the tweets from realDonaldTrump? In 2-4 sentences, please explain.

ANSWER: There does appear to be evidence to support Robinson's claim that Trump only writes the android half of tweets from realDonaldTrump. At the very least, we can identify a clear difference in sentiments and preferred bigrams between the two sources which is a strong indicator that they are from different authors. Especially given the focus of iphone tweets on campaign slogans as opposed to phtases such as "lyin ted", it does seem likely that Trump is in charge of the angrier tweets while a social media individual is tweeting the others.