Intro to Data Science - Lab 11

```
# Enter your name here: Hrishikesh Telang
```


Attribution statement: (choose only one and delete the rest)

```
# 1. I did this homework by myself, with help from the book and the professor.
```

#Instructions: Text mining plays an important role in many industries because of the prevalence of text in the interactions between customers and company representatives. Even when the customer interaction is by speech, rather than by chat or email, speech to text algorithms have gotten so good that transcriptions of these spoken word interactions are often available. To an increasing extent, a data scientist needs to be able to wield tools that turn a body of text into actionable insights.

#In this exercise, we work with a small number of social media posts on the topic of climate change. Make sure to install and library the readr and quanteda package before starting the exercise. Please include an attribution statement (see syllabus).

#1. Read in the following data set with read_csv(): https://intro-datascience.s3.us-east-2.amazonaws.com/ClimatePosts.csv

#The name of the file is "ClimatePosts.csv". Store the data in a data frame called tweetDF. Use str(tweetDF) to summarize the data. Add a comment describing what you see. Make sure to explain what each of the three variables contains.

```
library(readr)
tweetDF <- read_csv('https://intro-datascience.s3.us-east-2.amazonaws.com/ClimatePosts.csv')
```

```
## Rows: 18 Columns: 3
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): ID, Tweet
## dbl (1): Skeptic
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
#'tweetDF' basically contains three variables such as: 1) tweet_id which confirms
#the title of the tweet, how skeptic the tweet in the form of binary numbers
#and the tweet written
head(data)
```

```
##
## 1 function (..., list = character(), package = NULL, lib.loc = NULL,
## 2     verbose = getOption("verbose"), envir = .GlobalEnv, overwrite = TRUE)
```

```
## 3 {
## 4     fileExt <- function(x) {
## 5         db <- grepl("\\\\.[^.]+\\\\.(gz|bz2|xz)$", x)
## 6         ans <- sub(".*\\\\.", "", x)
```

```
tail(data)
```

```
##
## 182             }
## 183             rm(tmp_env)
## 184         }
## 185     }
## 186     invisible(names)
## 187 }
```

```
str(tweetDF)
```

```
## spec_tbl_df [18 x 3] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ID     :
```

```
## Warning: One or more parsing issues, see ‘problems()‘ for details
```

```
##  chr [1:18] "climatechange" "billmckibben" "megancollins" "neiltyson" ...
##  $ Skeptic: num [1:18] 0 0 0 0 0 0 0 0 0 1 ...
##  $ Tweet  : chr [1:18] "BREAKING: Iran soars to record of 129 degrees - near hottest ever reliably m
##  - attr(*, "spec")=
##   .. cols(
##   ..   ID = col_character(),
##   ..   Skeptic = col_double(),
##   ..   Tweet = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

#2. Use the corpus commands to turn the text variable into a quanteda corpus. You can use the IDs as the document titles with the following command:

```
#install.packages('quanteda')
library(quanteda)
```

```
## Package version: 3.1.0
## Unicode version: 13.0
## ICU version: 69.1
```

```
## Parallel computing: 8 of 8 threads used.
```

```
## See https://quanteda.io for tutorials and examples.
```

```
tweetCorpus <- corpus(tweetDF$Tweet, docnames=tweetDF$ID)
#this command creates separate documents as key-value pair
tweetCorpus
```

```
## Corpus consisting of 18 documents.
## climatechange :
## "BREAKING: Iran soars to record of 129 degrees - near hottest..."
##
## billmckibben :
## "I know you're Mr. America-is-all-that-matters, but climate i..."
##
## megancollins :
## "Could reporters stop asking if political leaders believe in ..."
##
## neiltyson :
## "If I were ever abducted by aliens, the first thing I'd ask i..."
##
## johnbiehl :
## "Alien: why should I not blow up this planet? Human: we are a..."
##
## ScottWesterfeld :
## "Plot idea: 97% of the world's scientists contrive an environ..."
##
## [ reached max_ndoc ... 12 more documents ]
```

#3. Next, convert the corpus into a document-feature matrix (DFM). Before you do that you can use "tokens" to remove punctuation and stop words. Use this code:

```
toks <- tokens(tweetCorpus, remove_punct=TRUE)
#This code removes punctuation.
toks_nostop <- tokens_select(toks, pattern = stopwords("en"), selection = "remove")
#This code removes stop words.
```

#Here's a command that will create the DFM:

```
tweetDFM <- dfm(toks_nostop, tolower = TRUE)
```

#4. Type tweetDFM at the console to find out the basic characteristic of the DFM (the number of terms, the number of documents, and the sparsity of the matrix). Write a comment describing what you observe.

```
tweetDFM
```

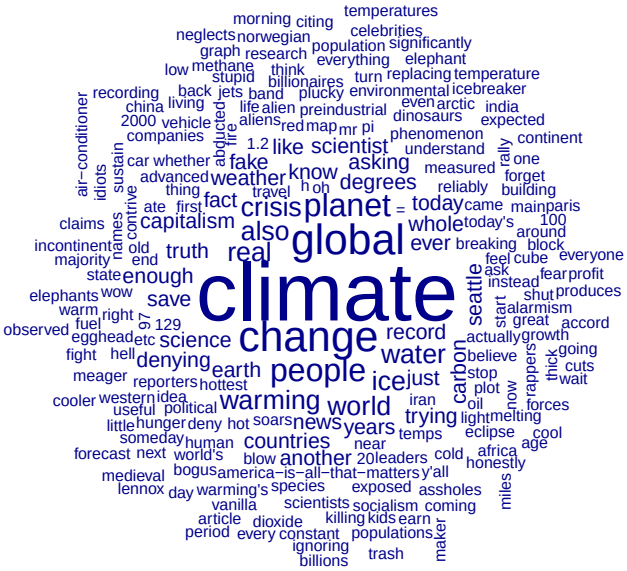
```
## Document-feature matrix of: 18 documents, 223 features (93.20% sparse) and 0 docvars.
##                  features
## docs              breaking iran soars record 129 degrees near hottest ever
##   climatechange          1    1     1      1   1       1    1       1    1
##   billmckibben           0    0     0      0   0       1    0       0    0
##   megancollins           0    0     0      0   0       0    0       0    0
##   neiltyson              0    0     0      0   0       0    0       0    1
##   johnbiehl              0    0     0      0   0       0    0       0    0
##   ScottWesterfeld        0    0     0      0   0       0    0       0    0
##                  features
## docs              reliably
##   climatechange          1
##   billmckibben           0
##   megancollins           0
```

```
##   neiltyson              0
##   johnbiehl              0
##   ScottWesterfeld        0
## [ reached max_ndoc ... 12 more documents, reached max_nfeat ... 213 more features ]
```

```
#tweetDFM consists of 18 documents, 223 features and the matrix
#is 93.20% sparse with 0 docvars.
```

#5. Create a wordcloud from the DFM using the following command. Write a comment describing notable features of the wordcloud:

```
#install.packages('quanteda.textplots')
library(quanteda.textplots)
textplot_wordcloud(tweetDFM, min_count = 1)
```




```
#This describes the frequency of the words in the tweets
```

#6. Using textstat_frequency() from the quanteda.textstats package, show the 10 most frequent words, and how many times each was used/mentioned.

```
#install.packages('quanteda.textstats')
library(quanteda.textstats)
textstat_frequency(tweetDFM)
```

```
##                        feature frequency rank docfreq group
## 1                      climate        13    1       8   all
## 2                       global         6    2       5   all
## 3                       change         6    2       6   all
## 4                       planet         4    4       3   all
## 5                       people         4    4       3   all
## 6                         also         3    6       3   all
## 7                       crisis         3    6       2   all
## 8                      warming         3    6       3   all
## 9                         real         3    6       3   all
## 10                       world         3    6       3   all
## 11                         ice         3    6       3   all
## 12                       water         3    6       1   all
## 13                      record         2   13       2   all
## 14                     degrees         2   13       2   all
## 15                        ever         2   13       2   all
## 16                       earth         2   13       2   all
## 17                        know         2   13       2   all
## 18                     weather         2   13       2   all
## 19                       whole         2   13       2   all
## 20                       today         2   13       2   all
## 21                      asking         2   13       1   all
## 22                     science         2   13       2   all
## 23                        news         2   13       2   all
## 24                        just         2   13       2   all
## 25                   scientist         2   13       1   all
## 26                        like         2   13       2   all
## 27                        fake         2   13       1   all
## 28                        fact         2   13       2   all
## 29                      carbon         2   13       2   all
## 30                   capitalism         2   13       1   all
## 31                       truth         2   13       2   all
## 32                   countries         2   13       2   all
## 33                      enough         2   13       2   all
## 34                       years         2   13       2   all
## 35                     another         2   13       2   all
## 36                     denying         2   13       1   all
## 37                      trying         2   13       1   all
## 38                        save         2   13       1   all
## 39                     seattle         2   13       1   all
## 40                    breaking         1   40       1   all
## 41                        iran         1   40       1   all
## 42                       soars         1   40       1   all
## 43                         129         1   40       1   all
## 44                        near         1   40       1   all
## 45                     hottest         1   40       1   all
## 46                    reliably         1   40       1   all
## 47                    measured         1   40       1   all
## 48                          mr         1   40       1   all
## 49  america-is-all-that-matters         1   40       1   all
## 50                     actually         1   40       1   all
## 51                   phenomenon         1   40       1   all
## 52                      today's         1   40       1   all
## 53                          map         1   40       1   all
```

```
## 54                        oh         1   40       1   all
## 55                       red         1   40       1   all
## 56                         =         1   40       1   all
## 57                       hot         1   40       1   all
## 58                       1.2         1   40       1   all
## 59            preindustrial          1   40       1   all
## 60                     temps         1   40       1   all
## 61                 reporters         1   40       1   all
## 62                      stop         1   40       1   all
## 63                 political         1   40       1   all
## 64                   leaders         1   40       1   all
## 65                   believe         1   40       1   all
## 66                     start         1   40       1   all
## 67                understand         1   40       1   all
## 68                   instead         1   40       1   all
## 69                  abducted         1   40       1   all
## 70                    aliens         1   40       1   all
## 71                     first         1   40       1   all
## 72                     thing         1   40       1   all
## 73                       ask         1   40       1   all
## 74                   whether         1   40       1   all
## 75                      came         1   40       1   all
## 76                      deny         1   40       1   all
## 77                     alien         1   40       1   all
## 78                      blow         1   40       1   all
## 79                     human         1   40       1   all
## 80                  advanced         1   40       1   all
## 81                   species         1   40       1   all
## 82                    travel         1   40       1   all
## 83                         h         1   40       1   all
## 84                     light         1   40       1   all
## 85                       old         1   40       1   all
## 86                 dinosaurs         1   40       1   all
## 87                      fire         1   40       1   all
## 88                      plot         1   40       1   all
## 89                      idea         1   40       1   all
## 90                        97         1   40       1   all
## 91                   world's         1   40       1   all
## 92                scientists         1   40       1   all
## 93                  contrive         1   40       1   all
## 94              environmental          1   40       1   all
## 95                   exposed         1   40       1   all
## 96                    plucky         1   40       1   all
## 97                      band         1   40       1   all
## 98               billionaires          1   40       1   all
## 99                       oil         1   40       1   all
## 100                companies         1   40       1   all
## 101                     cold         1   40       1   all
## 102                    great         1   40       1   all
## 103                   hunger         1   40       1   all
## 104                      ate         1   40       1   all
## 105                  eclipse         1   40       1   all
## 106                      wow         1   40       1   all
## 107                    right         1   40       1   all
```

```
## 108                    now     1   40       1   all
## 109                   shut     1   40       1   all
## 110                egghead     1   40       1   all
## 111              warming's     1   40       1   all
## 112                  y'all     1   40       1   all
## 113                someday     1   40       1   all
## 114                rappers     1   40       1   all
## 115                  names     1   40       1   all
## 116                   cube     1   40       1   all
## 117                vanilla     1   40       1   all
## 118                  bogus     1   40       1   all
## 119                around     1   40       1   all
## 120                dioxide     1   40       1   all
## 121                   main     1   40       1   all
## 122               building     1   40       1   all
## 123                  block     1   40       1   all
## 124                   life     1   40       1   all
## 125               alarmism     1   40       1   all
## 126                killing     1   40       1   all
## 127              replacing     1   40       1   all
## 128              socialism     1   40       1   all
## 129                   hell     1   40       1   all
## 130                vehicle     1   40       1   all
## 131                  rally     1   40       1   all
## 132                useful     1   40       1   all
## 133               assholes     1   40       1   all
## 134                forget     1   40       1   all
## 135             everything     1   40       1   all
## 136                  think     1   40       1   all
## 137               honestly     1   40       1   all
## 138                   feel     1   40       1   all
## 139                stupid     1   40       1   all
## 140                living     1   40       1   all
## 141              constant     1   40       1   all
## 142                 state     1   40       1   all
## 143                  fear     1   40       1   all
## 144               western     1   40       1   all
## 145                  kids     1   40       1   all
## 146                  even     1   40       1   all
## 147               sustain     1   40       1   all
## 148                meager     1   40       1   all
## 149           populations     1   40       1   all
## 150              majority     1   40       1   all
## 151            population     1   40       1   all
## 152                growth     1   40       1   all
## 153                  next     1   40       1   all
## 154                   100     1   40       1   all
## 155                coming     1   40       1   all
## 156                africa     1   40       1   all
## 157              expected     1   40       1   all
## 158               melting     1   40       1   all
## 159                 thick     1   40       1   all
## 160                arctic     1   40       1   all
## 161                forces     1   40       1   all
```

```
## 162                 norwegian         1   40       1   all
## 163                  research         1   40       1   all
## 164                 icebreaker        1   40       1   all
## 165                       turn        1   40       1   all
## 166                       back        1   40       1   all
## 167                incontinent        1   40       1   all
## 168                     idiots        1   40       1   all
## 169                    article        1   40       1   all
## 170                     claims        1   40       1   all
## 171                   elephants       1   40       1   all
## 172                      fight        1   40       1   all
## 173                   ignoring        1   40       1   all
## 174                   elephant        1   40       1   all
## 175                   produces        1   40       1   all
## 176                    methane        1   40       1   all
## 177                        one        1   40       1   all
## 178                        day        1   40       1   all
## 179                       fuel        1   40       1   all
## 180                        car        1   40       1   all
## 181                         20        1   40       1   all
## 182                      miles        1   40       1   all
## 183                       2000        1   40       1   all
## 184                   medieval        1   40       1   all
## 185                       warm        1   40       1   all
## 186                     period        1   40       1   all
## 187                     little        1   40       1   all
## 188                        age        1   40       1   all
## 189                      every        1   40       1   all
## 190                  continent        1   40       1   all
## 191                      going        1   40       1   all
## 192                        end        1   40       1   all
## 193                   everyone        1   40       1   all
## 194                       earn        1   40       1   all
## 195                     profit        1   40       1   all
## 196                      china        1   40       1   all
## 197                      india        1   40       1   all
## 198                       wait        1   40       1   all
## 199                   billions        1   40       1   all
## 200                      paris        1   40       1   all
## 201                     accord        1   40       1   all
## 202                celebrities        1   40       1   all
## 203                      trash        1   40       1   all
## 204                       jets        1   40       1   all
## 205                        etc        1   40       1   all
## 206          air-conditioner          1   40       1   all
## 207                      maker        1   40       1   all
## 208                     lennox        1   40       1   all
## 209                       cuts        1   40       1   all
## 210                   forecast        1   40       1   all
## 211                     citing        1   40       1   all
## 212               significantly       1   40       1   all
## 213                      cooler        1   40       1   all
## 214                temperatures        1   40       1   all
## 215                       cool        1   40       1   all
```

```
## 216                    morning         1   40       1   all
## 217                         pi         1   40       1   all
## 218                   neglects         1   40       1   all
## 219                        low         1   40       1   all
## 220                  recording         1   40       1   all
## 221                   observed         1   40       1   all
## 222                temperature         1   40       1   all
## 223                      graph         1   40       1   all
```

```
#This code shows the words with the most frequency ranked in an ascending order
```

#7. Next, we will read in dictionaries of positive and negative words to see what we can match up to the text in our DFM. Here's a line of code for reading in the list of positive words:

```
URL <- "https://intro-datascience.s3.us-east-2.amazonaws.com/positive-words.txt"
posWords <- scan(URL, character(0), sep = "\n")
posWords <- posWords[-1:-34]
```

#Create a similar line of code to read in the negative words, with the following URL: https://intro-datascience.s3.us-east-2.amazonaws.com/negative-words.txt

```
URL <- "https://intro-datascience.s3.us-east-2.amazonaws.com/negative-words.txt"
negWords <- scan(URL, character(0), sep = "\n")
negWords <- posWords[-1:-34]
```

#There should be 2006 positive words and 4783 negative words.

#8. Explain what the following lines of code does and comment each line. Then add similar code for the negative words.

```
posDFM <- dfm_match(tweetDFM, posWords)
#This code matches the positive words with the tweetDFM dataframe
posFreq <- textstat_frequency(posDFM)
#This code tells the frequency of positive words in the documents
negDFM <- dfm_match(tweetDFM, negWords)
#This code matches the negative words with the tweetDFM dataframe
negFreq <- textstat_frequency(negDFM)
#This code tells the frequency of negative words in the documents.
```

#9. Explore posFreq and negFreq using str() or glimpse(). Explain the fields in these data frames.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v dplyr   1.0.7
## v tibble  3.1.4     v stringr 1.4.0
## v tidyr   1.1.3     v forcats 0.5.1
## v purrr   0.3.4

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
str(posFreq)
```

```
## Classes 'frequency', 'textstat' and 'data.frame':    12 obs. of  5 variables:
##  $ feature  : chr  "enough" "like" "advanced" "cool" ...
##  $ frequency: num  2 2 1 1 1 1 1 1 1 1 ...
##  $ rank     : num  1 1 3 3 3 3 3 3 3 3 ...
##  $ docfreq  : num  2 2 1 1 1 1 1 1 1 1 ...
##  $ group    : chr  "all" "all" "all" "all" ...
```

```
glimpse(posFreq)
```

```
## Rows: 12
## Columns: 5
## $ feature   <chr> "enough", "like", "advanced", "cool", "great", "hot", "hotte~
## $ frequency <dbl> 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
## $ rank      <dbl> 1, 1, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3
## $ docfreq   <dbl> 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
## $ group     <chr> "all", "all", "all", "all", "all", "all", "all", "all", "all~
```

```
# posFreq contains all the positive words in the feature column with
#the number of frequencies of the particular word and rank them based
# on the frequency of the number.
str(negFreq)
```

```
## Classes 'frequency', 'textstat' and 'data.frame':    12 obs. of  5 variables:
##  $ feature  : chr  "enough" "like" "advanced" "cool" ...
##  $ frequency: num  2 2 1 1 1 1 1 1 1 1 ...
##  $ rank     : num  1 1 3 3 3 3 3 3 3 3 ...
##  $ docfreq  : num  2 2 1 1 1 1 1 1 1 1 ...
##  $ group    : chr  "all" "all" "all" "all" ...
```

```
glimpse(negFreq)
```

```
## Rows: 12
## Columns: 5
## $ feature   <chr> "enough", "like", "advanced", "cool", "great", "hot", "hotte~
## $ frequency <dbl> 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
## $ rank      <dbl> 1, 1, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3
## $ docfreq   <dbl> 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
## $ group     <chr> "all", "all", "all", "all", "all", "all", "all", "all", "all~
```

```
# negFreq contains all the negative words in the feature column with
#the number of frequencies of the particular word and rank them based
# on the frequency of the number.
```

#10. Output the 10 most frequently occurring positive and negative words including how often each occurred.

```
posFreq[1:10,]
```

```
##     feature frequency rank docfreq group
## 1    enough         2    1       2   all
## 2      like         2    1       2   all
## 3  advanced         1    3       1   all
## 4      cool         1    3       1   all
## 5     great         1    3       1   all
## 6       hot         1    3       1   all
## 7   hottest         1    3       1   all
## 8  reliably         1    3       1   all
## 9     right         1    3       1   all
## 10   useful         1    3       1   all
```

```
negFreq[1:10,]
```

```
##     feature frequency rank docfreq group
## 1    enough         2    1       2   all
## 2      like         2    1       2   all
## 3  advanced         1    3       1   all
## 4      cool         1    3       1   all
## 5     great         1    3       1   all
## 6       hot         1    3       1   all
## 7   hottest         1    3       1   all
## 8  reliably         1    3       1   all
## 9     right         1    3       1   all
## 10   useful         1    3       1   all
```