TAD Week 7 Assignment

Tommy Klein

3/16/2022

Working Directory

```
setwd('/Users/tklein/Desktop/Desktop_tpk/JHU_Classes/text_as_data/week7')
```

Library

```
library(ndjson)
library(SentimentAnalysis)
```

```
##
## Attaching package: 'SentimentAnalysis'
```

```
## The following object is masked from 'package:base':
##
##     write
```

```
library(RedditExtractoR)
library(tidyverse)
```

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

```
## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.1.2     v dplyr   1.0.6
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

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

```
library(topicmodels)
library(stm)
```

```
## stm v1.3.6 successfully loaded. See ?stm for help.
##  Papers, resources, and other materials at structuraltopicmodel.com
```

```
library(tidytext)
source('../functions/helper_functions.R')
```

```
## Package version: 3.2.0
## Unicode version: 13.0
## ICU version: 69.1
```

```
## Parallel computing: 4 of 4 threads used.
## See https://quanteda.io for tutorials and examples.
```

Question 1

Topic models define a document as a mixture of different topics. The number of topics are defined by the user, but model defins each topic as a bag of words. The model determines what words are associated with each topic by analyzing the frequency of words. The topic model uses the bag of words that it determines represents each topic to determine what topics are in each document. This approach is useful for exploring documents and answering questions where the set of topics is unknown, because the model will define topics and classify documents for you. For example, you could use topic modeling to analyze hotel reviews to determine what topics are associated with positive or negative reviews.

Topic modeling is not well suited to every question. Topic models require you to define how many topics are in the document, so if you don't have a good grasp of what that number should be you could get poor results. Additionally, the topics as defined by the topic model aren't necessarily interpretable. The model will not be able to tell you if a blog post was discussing politics or fashion, all it can do is determine how correlated each blog post is with a set of words determined by model. This means that the model may not be ideal for exploring documents that the user is completely unfamiliar with.

Question 2

```r
crypto_threads_df <- read_csv('reddit_crypto_threads.csv') %>% select(-X1)
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   X1 = col_double(),
##   url = col_character(),
##   author = col_character(),
##   date = col_date(format = ""),
##   title = col_character(),
##   text = col_character(),
##   subreddit = col_character(),
##   score = col_double(),
##   upvotes = col_double(),
##   downvotes = col_double(),
##   up_ratio = col_double(),
##   total_awards_received = col_double(),
##   golds = col_double(),
##   cross_posts = col_double(),
##   comments = col_double()
## )
```

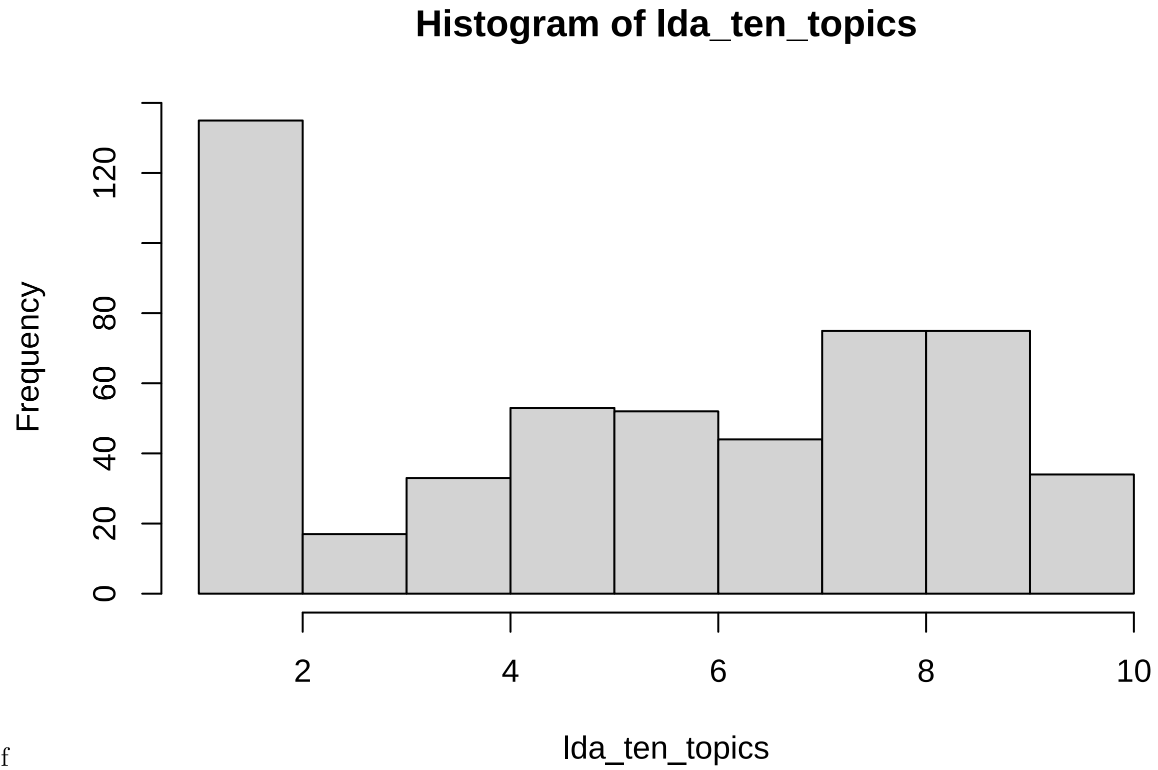
```r
crypto_threads <- csv_to_corpus('reddit_crypto_threads.csv', 'text')


crypto_dfm <- corp_to_dfm(crypto_threads, stem = T)
```

```
## Warning: 'stem' is deprecated; use dfm_wordstem() instead
```

```
crypto_dfm_trimmed <- dfm_trim(crypto_dfm, min_count = 1, max_count = .9)


# need to remove threads that dont have any text


crypto_dfm_trimmed <- crypto_dfm_trimmed[apply(crypto_dfm_trimmed, 1, sum) > 0,]

crypto_threads_df_with_text <- crypto_threads_df[apply(crypto_dfm, 1, sum) > 0,]


colnames(crypto_dfm_trimmed)[1:20]
```

```
##  [1] "autom"     "termin"    "counter"   "similar"   "chacha"    "design"
##  [7] "detect"    "subt"      "iacr"      "public"    "name"      "isogen"
## [13] "multicurv" "lattic"    "best"      "attack"    "ae"        "use"
## [19] "deck"      "card"
```

```
lda_ten <- LDA(crypto_dfm_trimmed, 10, method = "VEM")

lda_ten_topics <- topics(lda_ten)
hist(lda_ten_topics, breaks = 10)
```



lda 10-1.pdf

```
terms(lda_ten, 10)
```

```
##       Topic 1   Topic 2   Topic 3   Topic 4 Topic 5  Topic 6   Topic 7
## [1,] "key"     "encrypt" "x"       "amp"   "key"    "encrypt" "messag"
## [2,] "use"     "use"     "1"       "2"     "use"    "use"     "h"
## [3,] "encrypt" "key"     "c"       "p"     "privat" "file"    "key"
## [4,] "hash"    "can"     "g"       "point" "generat" "data"   "encrypt"
##  [5,] "password" "secur"   "2"        "1"     "public"  "block"    "can"
##  [6,] "like"     "file"    "secret"   "mod"   "random"  "password" "use"
##  [7,] "can"      "data"    "p"        "use"   "seed"    "text"     "signatur"
##  [8,] "attack"   "ani"     "function" "y"     "number"  "can"      "public"
##  [9,] "decrypt"  "like"    "f"        "n"     "lt"      "want"     "nonc"
## [10,] "just"     "need"    "random"   "x"     "encrypt" "bit"      "s"
##       Topic 8        Topic 9     Topic 10
##  [1,] "cryptographi" "key"       "discuss"
##  [2,] "like"         "can"       "thread"
##  [3,] "learn"        "curv"      "etc"
##  [4,] "math"         "use"       "r"
##  [5,] "book"         "algorithm" "topic"
##  [6,] "ani"          "encrypt"   "keep"
##  [7,] "field"        "one"       "communiti"
##  [8,] "crypto"       "whi"       "subreddit"
##  [9,] "work"         "attack"    "rule"
## [10,] "see"          "prime"     "mind"
```

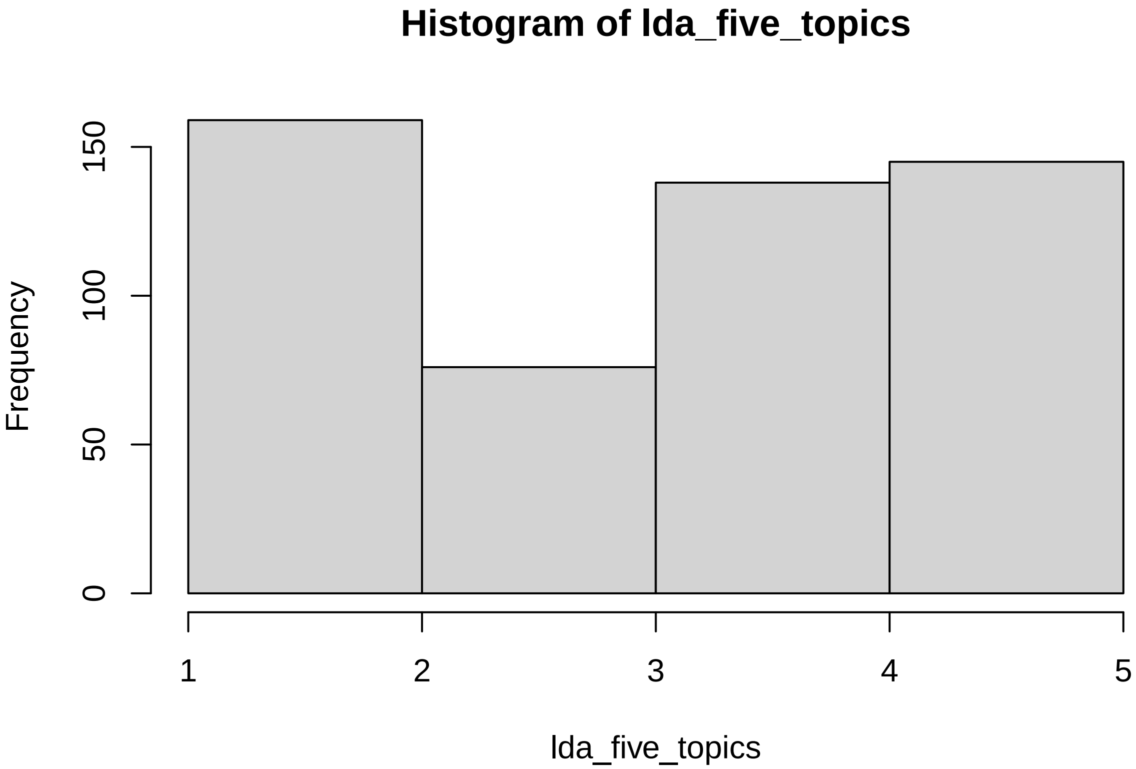
```
lda_ten_props <- lda_ten@gamma

lda_ten_props[1:3, ]
```

```
##              [,1]        [,2]         [,3]         [,4]         [,5]
## [1,] 1.610293e-03 0.001610293 1.610293e-03 1.610293e-03 1.610293e-03
## [2,] 1.100671e-03 0.001100671 1.100671e-03 1.100671e-03 9.000418e-01
## [3,] 7.943118e-05 0.151093194 7.943118e-05 7.943118e-05 7.943118e-05
##             [,6]         [,7]        [,8]         [,9]        [,10]
## [1,] 0.001610293 9.855074e-01 0.001610293 1.610293e-03 1.610293e-03
## [2,] 0.001100671 1.100671e-03 0.001100671 9.115285e-02 1.100671e-03
## [3,] 0.798212579 7.943118e-05 0.050138209 7.943118e-05 7.943118e-05
```

```
lda_ten_props_df <- bind_cols(crypto_threads_df_with_text, data.frame(lda_ten_props))


lm(score ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 + X9, data = lda_ten_props_df) %>% summary()
```

```
## 
## Call:
## lm(formula = score ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 + 
##     X9, data = lda_ten_props_df)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -23.270 -11.314  -4.164   5.282 216.361 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)   21.390      3.931   5.442 8.23e-08 ***
## X1            -8.058      4.992  -1.614   0.1071    
## X2            -4.158      4.862  -0.855   0.3929    
## X3           -14.014      6.922  -2.025   0.0434 *  
## X4            -2.770      5.614  -0.493   0.6220    
## X5            -8.244      5.171  -1.594   0.1115    
## X6           -10.825      5.255  -2.060   0.0399 *  
## X7            -3.306      5.283  -0.626   0.5317    
## X8              2.056      4.787   0.429   0.6678
## X9             -7.473      4.827  -1.548   0.1222
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.79 on 508 degrees of freedom
## Multiple R-squared:  0.03456,    Adjusted R-squared:  0.01745
## F-statistic:  2.02 on 9 and 508 DF,  p-value: 0.03532
```

```
lda_five <- LDA(crypto_dfm_trimmed, 5, method = "VEM")

lda_five_topics <- topics(lda_five)
hist(lda_five_topics, breaks = 5)
```



lda 5-1.pdf

```
terms(lda_five, 10)
```

```
##       Topic 1    Topic 2    Topic 3  Topic 4   Topic 5
##  [1,] "key"      "key"      "key"    "encrypt" "cryptographi"
##  [2,] "use"      "use"      "x"      "use"     "discuss"
##  [3,] "encrypt"  "can"      "2"      "can"     "can"
##  [4,] "password" "messag"   "1"      "key"     "see"
##  [5,] "file"     "data"     "public" "one"     "crypto"
##  [6,] "can"      "signatur" "p"      "want"    "like"
##  [7,] "h"        "server"   "amp"    "messag"  "topic"
##  [8,] "attack"   "byte"     "gt"     "just"    "ani"
##  [9,] "1"        "encrypt"  "can"    "secret"  "r"
## [10,] "data"     "bit"      "curv"   "know"    "communiti"
```

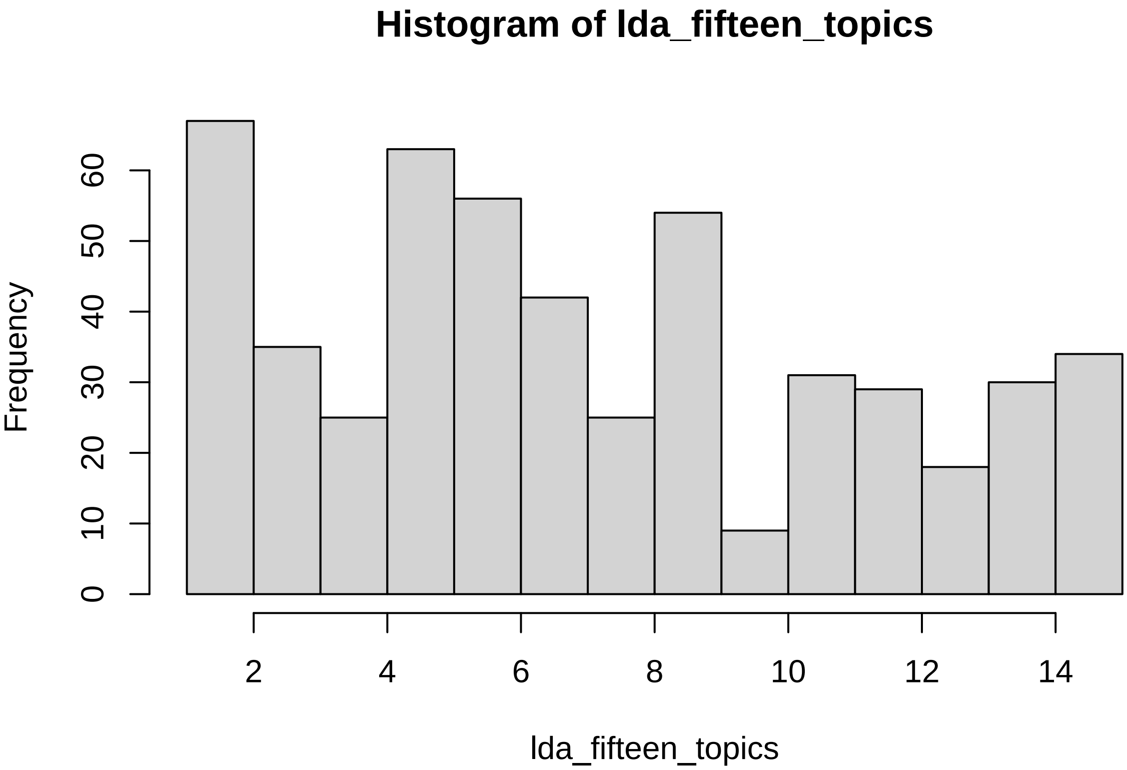
```
lda_five_props <- lda_five@gamma

lda_five_props[1:3, ]
```

```
##              [,1]         [,2]         [,3]        [,4]        [,5]
## [1,] 0.0028551469 0.9885794126 0.0028551469 0.002855147 0.002855147
## [2,] 0.0019504098 0.0019504098 0.3157096087 0.678439162 0.001950410
## [3,] 0.0001405881 0.0001405881 0.0001405881 0.803011881 0.196566355
```

```
lda_five_props_df <- bind_cols(crypto_threads_df_with_text, data.frame(lda_five_props))


lm(score ~ X1 + X2 + X3 + X4, data = lda_five_props_df) %>% summary()
```

```
##
## Call:
## lm(formula = score ~ X1 + X2 + X3 + X4, data = lda_five_props_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -22.269 -10.975  -4.166   5.064 214.530
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   22.303      1.922  11.603  < 2e-16 ***
## X1            -8.173      3.195  -2.558  0.01082 *
## X2            -7.775      3.471  -2.240  0.02553 *
## X3           -10.377      3.388  -3.063  0.00231 **
## X4            -7.658      2.880  -2.658  0.00809 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.79 on 513 degrees of freedom
## Multiple R-squared:  0.0257, Adjusted R-squared:  0.0181
## F-statistic: 3.382 on 4 and 513 DF,  p-value: 0.009561
```

```
lda_fifteen <- LDA(crypto_dfm_trimmed, 15, method = "VEM")

lda_fifteen_topics <- topics(lda_fifteen)
hist(lda_fifteen_topics, breaks = 15)
```



lda 15-1.pdf

```
terms(lda_fifteen, 10)
```

```
##       Topic 1    Topic 2    Topic 3     Topic 4  Topic 5        Topic 6
##  [1,] "amp"      "password" "discuss"   "random" "use"          "cryptographi"
##  [2,] "use"      "text"     "thread"    "x"      "cryptographi" "math"
##  [3,] "time"     "use"      "keep"      "c"      "like"         "ani"
##  [4,] "system"   "file"     "etc"       "use"    "algorithm"    "learn"
##  [5,] "i'v"      "word"     "mind"      "test"   "encrypt"      "research"
##  [6,] "communic" "1"        "subreddit" "data"   "question"     "job"
##  [7,] "can"      "card"     "r"         "lt"     "thank"        "interest"
##  [8,] "meetup"   "cipher"   "contain"   "gt"     "thing"        "look"
##  [9,] "see"      "2"        "topic"     "string" "secur"        "know"
## [10,] "certif"   "one"      "rule"      "result" "implement"    "work"
##       Topic 7   Topic 8     Topic 9    Topic 10 Topic 11   Topic 12   Topic 13
##  [1,] "key"     "h"         "use"      "bit"    "encrypt"  "hash"     "x"
##  [2,] "encrypt" "nonc"      "can"      "block"  "data"     "function" "2"
##  [3,] "use"     "messag"    "encrypt"  "secret" "use"      "find"     "p"
##  [4,] "public"  "n"         "signatur" "cipher" "just"     "can"      "1"
##  [5,] "privat"  "s"         "certif"   "amp"    "can"      "use"      "mod"
##  [6,] "decrypt" "c"         "key"      "key"    "secur"    "attack"   "n"
##  [7,] "can"     "pw"        "public"   "#x200b" "want"     "1"        "point"
##  [8,] "messag"  "use"       "set"      "pool"   "password" "know"     "g"
##  [9,] "generat" "algorithm" "server"   "data"   "ani"      "cipher"   "f"
## [10,] "hash"    "key"       "secur"    "1"      "contain"  "book"     "y"
##       Topic 14 Topic 15
##  [1,] "whi"    "encrypt"
##  [2,] "use"    "curv"
##  [3,] "just"   "can"
##  [4,] "want"   "file"
##  [5,] "can"    "ani"
##  [6,] "post"   "use"
##  [7,] "attack" "key"
##  [8,] "messag" "order"
##  [9,] "make"   "system"
## [10,] "server" "number"
```

```
lda_fifteen_props <- lda_fifteen@gamma

lda_fifteen_props[1:3, ]
```

```
##              [,1]         [,2]         [,3]         [,4]         [,5]
## [1,] 1.400705e-03 1.400705e-03 1.400705e-03 1.400705e-03 1.400705e-03
## [2,] 9.589263e-04 7.222751e-01 9.589263e-04 9.589263e-04 9.589263e-04
## [3,] 6.942197e-05 6.942197e-05 6.942197e-05 6.942197e-05 6.942197e-05
##              [,6]         [,7]         [,8]         [,9]        [,10]
## [1,] 1.400705e-03 7.119444e-02 9.105964e-01 1.400705e-03 1.400705e-03
## [2,] 9.589263e-04 9.589263e-04 9.589263e-04 9.589263e-04 9.589263e-04
## [3,] 6.942197e-05 6.942197e-05 6.942197e-05 6.942197e-05 6.942197e-05
##             [,11]        [,12]        [,13]        [,14]        [,15]
## [1,] 0.0014007046 1.400705e-03 1.400705e-03 1.400705e-03 1.400705e-03
## [2,] 0.0009589263 9.589263e-04 2.652589e-01 9.589263e-04 9.589263e-04
## [3,] 0.9990280924 6.942197e-05 6.942197e-05 6.942197e-05 6.942197e-05
```

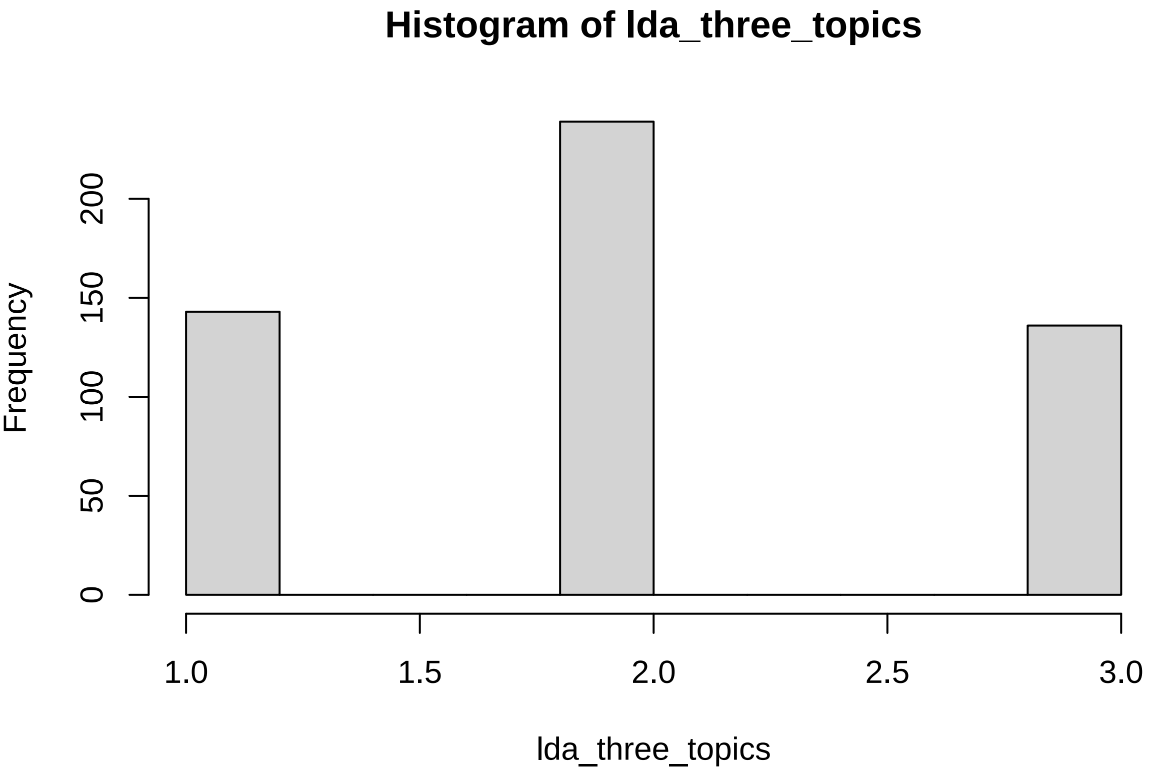
```
lda_fifteen_props_df <- bind_cols(crypto_threads_df_with_text, data.frame(lda_fifteen_props))


lm(score ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 + X9 + X10 + X11 + X12 + X13 + X14, data = lda_fifteen_
```

```
##
## Call:
## lm(formula = score ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 +
##     X9 + X10 + X11 + X12 + X13 + X14, data = lda_fifteen_props_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -23.636 -10.498  -3.995   5.462 212.096
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.8767     4.3549   2.268   0.0238 *
## X1           14.0564     6.1245   2.295   0.0221 *
## X2            1.6503     5.9556   0.277   0.7818
## X3            7.7711     5.8932   1.319   0.1879
## X4            1.0199     6.6790   0.153   0.8787
## X5           12.2210     5.4444   2.245   0.0252 *
## X6           11.1083     5.3263   2.086   0.0375 *
## X7            4.5949     5.8167   0.790   0.4299
## X8           -0.2047     6.6537  -0.031   0.9755
## X9            4.2437     5.5262   0.768   0.4429
## X10          -3.4860     9.5451  -0.365   0.7151
## X11           0.8839     6.2564   0.141   0.8877
## X12          12.9746     6.2600   2.073   0.0387 *
## X13          -2.7324     7.3626  -0.371   0.7107
## X14          12.7892     6.2321   2.052   0.0407 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.73 on 503 degrees of freedom
## Multiple R-squared:  0.05034,    Adjusted R-squared:  0.02391
## F-statistic: 1.905 on 14 and 503 DF,  p-value: 0.02378
```

The different values of values of K seemed pretty similar - there were of course more topics, but all of the topics seemed very similar. They almost all featured the word encrypt or crypt-. This would seem to indicate that I should use a smaller set of topics so that the ones that I end up with are more distinct from each-other.

Question 3

```
lda_three <- LDA(crypto_dfm_trimmed, 3, method = "VEM")

lda_three_topics <- topics(lda_three)
hist(lda_three_topics, breaks = 10)
```



three-1.pdf

```
terms(lda_three, 15)
```

```
##       Topic 1     Topic 2        Topic 3
##  [1,] "key"       "use"          "key"
##  [2,] "use"       "key"          "x"
##  [3,] "can"       "encrypt"      "use"
##  [4,] "discuss"   "cryptographi" "1"
##  [5,] "data"      "can"          "encrypt"
##  [6,] "amp"       "ani"          "can"
##  [7,] "messag"    "know"         "2"
##  [8,] "encrypt"   "like"         "p"
##  [9,] "see"       "secur"        "secret"
## [10,] "r"         "password"     "function"
## [11,] "etc"       "want"         "one"
## [12,] "communiti" "one"          "h"
## [13,] "thread"    "look"         "gt"
## [14,] "subreddit" "just"         "hash"
## [15,] "secur"     "tri"          "random"
```

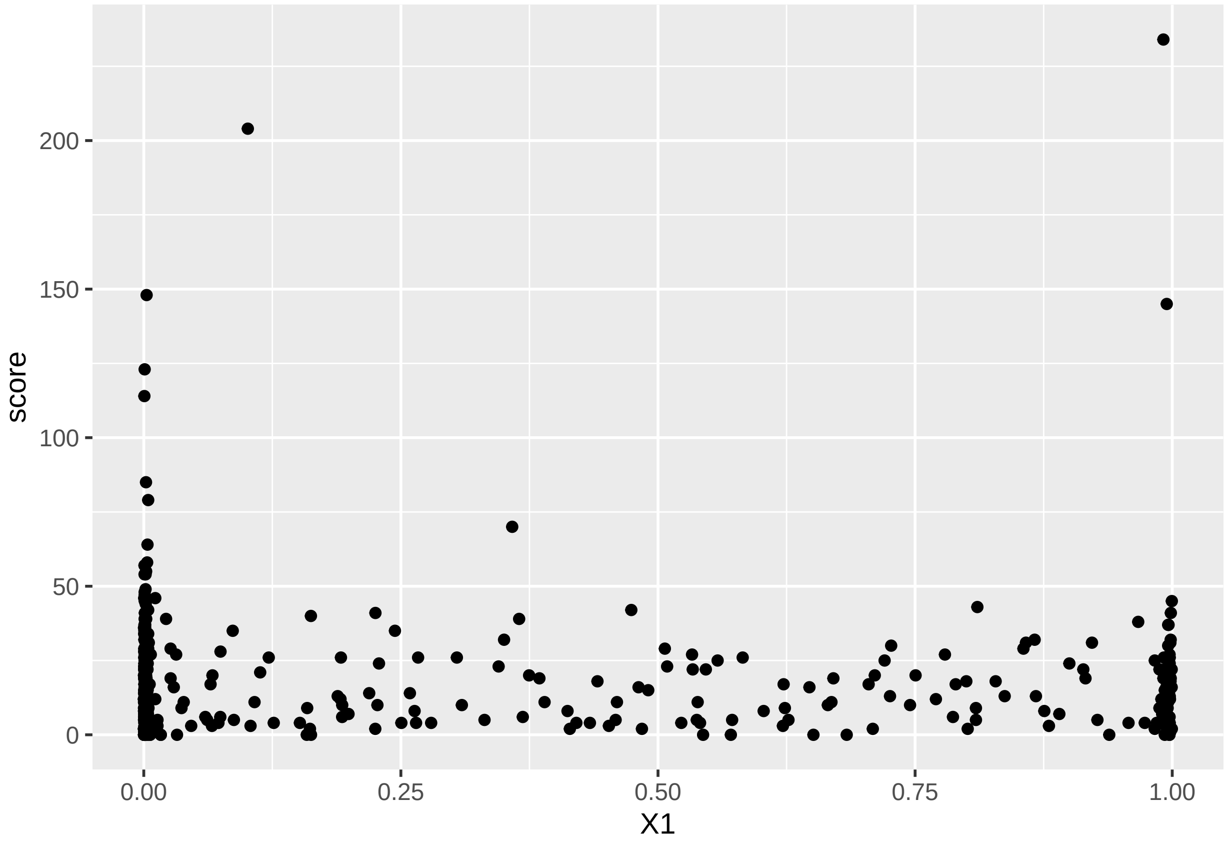
```
lda_three_props <- lda_three@gamma

lda_three_props[1:3, ]
```

```
##              [,1]        [,2]        [,3]
## [1,] 0.1584820394 0.837475419 0.004042542
## [2,] 0.0027596393 0.002759644 0.994480717
## [3,] 0.0001986445 0.790246719 0.209554637
```

```
lda_three_props_df <- bind_cols(crypto_threads_df_with_text, data.frame(lda_three_props))


lm(score ~ X1 + X2, data = lda_three_props_df) %>% summary()
```

```
##
## Call:
## lm(formula = score ~ X1 + X2, data = lda_three_props_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.892 -11.833  -4.815   5.443 216.127
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   11.809      1.973   5.987 4.02e-09 ***
## X1             6.091      2.858   2.131   0.0336 *
## X2             6.036      2.569   2.349   0.0192 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.89 on 515 degrees of freedom
## Multiple R-squared:  0.01236,    Adjusted R-squared:  0.008528
## F-statistic: 3.223 on 2 and 515 DF,  p-value: 0.04063
```

```
lda_three_props_df %>%
  ggplot(aes(X1, score))+
  geom_point()
```



```
lda_three %>%
  tidy() %>%
  filter(topic == 1) %>%
  arrange(desc(beta)) %>%
  head(15)
```

```
## # A tibble: 15 x 3
##    topic term         beta
##    <int> <chr>       <dbl>
##  1     1 key       0.0182
##  2     1 use       0.0144
##  3     1 can       0.00971
##  4     1 discuss   0.00856
##  5     1 data      0.00832
##  6     1 amp       0.00790
##  7     1 messag    0.00726
##  8     1 encrypt   0.00638
##  9     1 see       0.00616
## 10     1 r         0.00616
## 11     1 etc       0.00610
## 12     1 communiti 0.00559
## 13     1 thread    0.00533
## 14     1 subreddit 0.00525
## 15     1 secur     0.00523
```

I chose a value of 3 for K because I believe that my documents are all discussing similar topics. Because my documents are all posts in the crypto sub-reddit, it is highly likely that they are discussing the same, if not very similar, topics. Due to this, selecting a higher value of K would mean that the bag of words that comprise each topic would likely overlap quite a bit. After viewing the results of the LDA model with a K value of 15, 10, and 5, it was made that this was the case - that the words in each topic overlapped quite a bit.

I used the results of the topic model in a linear regression on the scores of each post. The results did not explain very much of the variance in the data (the model had an R-squared of only .03), but the coefficient on the 1st topic was statistically significant that the 1% level. This would indicate that posts that wholly categorized as topic 1 received on average 12 more upvotes than downvotes on Reddit. Practically, what this means is that were Centsy to post on Reddit, they should try to have their post fit in to topic 1, in order to increase the odds of receiving more upvotes. To make the post more likely to be like topic 1, it would have to the words that are most associated with topic 1. The table above lists those words, in the order of their likelihood of appearing in topic 1. Using more words with a higher beta means the post will be more likely to be like topic 1.

All of this may be of limited use given how little variance was explained by the model. There are clearly more factors that determine the score that a reddit post receives than just the topics in the post, and it is clear that the topics in the post account for very little of the variation in scores. It would likely be worthwhile to identify what these others factors are so that they can also be used to craft a strong Reddit social media strategy for Centsy.