reuters_again

Lowrance, Mikala

2024-08-18

R Markdown

```
library(tm)
```

```
## Loading required package: NLP
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x ggplot2::annotate() masks NLP::annotate()
## x dplyr::filter()     masks stats::filter()
## x dplyr::lag()        masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```
readerPlain = function(fname){
  readPlain(elem=list(content=readLines(fname)),
            id=fname, language='en') }

file_list = Sys.glob('repos/STA380/data/ReutersC50/C50train/*/*.txt')
reuters = lapply(file_list, readerPlain)

# Clean up the file names
mynames = file_list %>%
  { strsplit(., '/', fixed=TRUE) } %>%
  { lapply(., tail, n=2) } %>%
  { lapply(., paste0, collapse = '') } %>%
  unlist


# Extract author names from file paths
author_names = file_list %>%
```

```r
  { strsplit(., '/', fixed=TRUE) } %>%
  { lapply(., function(x) x[length(x) - 1]) } %>%
  unlist

# Rename the articles
names(reuters) = mynames

# Create corpus
documents_raw = Corpus(VectorSource(reuters))

# Pre-processing steps
my_documents = documents_raw
my_documents = tm_map(my_documents, content_transformer(tolower))
```

```
## Warning in tm_map.SimpleCorpus(my_documents, content_transformer(tolower)):
## transformation drops documents
```

```r
my_documents = tm_map(my_documents, content_transformer(removeNumbers))
```

```
## Warning in tm_map.SimpleCorpus(my_documents,
## content_transformer(removeNumbers)): transformation drops documents
```

```r
my_documents = tm_map(my_documents, content_transformer(removePunctuation))
```

```
## Warning in tm_map.SimpleCorpus(my_documents,
## content_transformer(removePunctuation)): transformation drops documents
```

```r
my_documents = tm_map(my_documents, content_transformer(stripWhitespace))
```

```
## Warning in tm_map.SimpleCorpus(my_documents,
## content_transformer(stripWhitespace)): transformation drops documents
```

```r
# Remove stop words
#stopwords("en")
#stopwords("SMART")
my_documents = tm_map(my_documents, content_transformer(removeWords), stopwords("en"))
```

```
## Warning in tm_map.SimpleCorpus(my_documents, content_transformer(removeWords),
## : transformation drops documents
```

```r
my_documents = tm_map(my_documents, content_transformer(removeWords), stopwords("SMART"))
```

```
## Warning in tm_map.SimpleCorpus(my_documents, content_transformer(removeWords),
## : transformation drops documents
```

```r
# Create document-term matrix
DTM_reuters = DocumentTermMatrix(my_documents)
class(DTM_reuters)
```

```
## [1] "DocumentTermMatrix"    "simple_triplet_matrix"
```

```r
# Review frequent words & word associations
#inspect(DTM_reuters[1:10,1:20])
findFreqTerms(DTM_reuters, lowfreq = 500)
```

```
##   [1] "announced"     "business"       "character"      "computer"
##   [5] "datetimestamp" "description"    "director"       "early"
##   [9] "fund"          "gmt"            "gmtoff"         "group"
##  [13] "heading"       "hour"           "internet"       "investors"
##  [17] "isdst"         "language"       "law"            "listauthor"
##  [21] "listcontent"   "listsec"        "local"          "lower"
##  [25] "major"         "mday"           "meta"           "million"
##  [29] "min"           "mon"            "money"          "month"
##  [33] "national"      "net"            "offer"          "origin"
##  [37] "services"      "set"            "shares"         "state"
##  [41] "technology"    "trade"          "tuesday"        "wday"
##  [45] "wednesday"     "world"          "yday"           "year"
##  [49] "zone"          "communications" "corp"           "earlier"
##  [53] "people"        "plan"           "plans"          "president"
##  [57] "sector"        "service"        "software"       "system"
##  [61] "trading"       "executive"      "companies"      "end"
##  [65] "good"          "government"     "including"      "international"
##  [69] "market"        "months"         "number"         "operating"
##  [73] "products"      "week"           "work"           "interest"
##  [77] "statement"     "analyst"        "banks"          "buy"
##  [81] "company"       "exchange"       "financial"      "officials"
##  [85] "sales"         "securities"     "states"         "big"
##  [89] "billion"       "court"          "expected"       "firms"
##  [93] "foreign"       "future"         "general"        "investment"
##  [97] "markets"       "move"           "operations"     "part"
## [101] "told"          "united"         "added"          "chief"
## [105] "made"          "recent"         "stock"          "years"
## [109] "back"          "based"          "chairman"       "customers"
## [113] "friday"        "half"           "high"           "results"
## [117] "time"          "growth"         "key"            "monday"
## [121] "news"          "strong"         "bank"           "current"
## [125] "deal"          "economic"       "report"         "thursday"
## [129] "companys"      "domestic"       "industry"       "make"
## [133] "share"         "workers"        "increase"       "reuters"
## [137] "long"          "meeting"        "official"       "spokesman"
## [141] "analysts"      "percent"        "amp"            "firm"
## [145] "largest"       "total"          "costs"          "due"
## [149] "pay"           "price"          "prices"         "ago"
## [153] "management"    "merger"         "cost"           "profit"
## [157] "agreement"     "reported"       "capital"        "air"
## [161] "close"         "earnings"       "higher"         "rise"
## [165] "rose"          "british"        "bid"            "beijing"
## [169] "stake"         "profits"        "quarter"        "party"
## [173] "oil"           "shareholders"   "pounds"         "cash"
## [177] "pence"         "talks"          "hong"           "kong"
## [181] "china"         "chinas"         "chinese"        "cents"
## [185] "gold"          "tonnes"
```

```r
findAssocs(DTM_reuters, "approve", .5)
```

```
## $approve
##       collar consummation    expedited
##         0.58         0.58         0.52
```

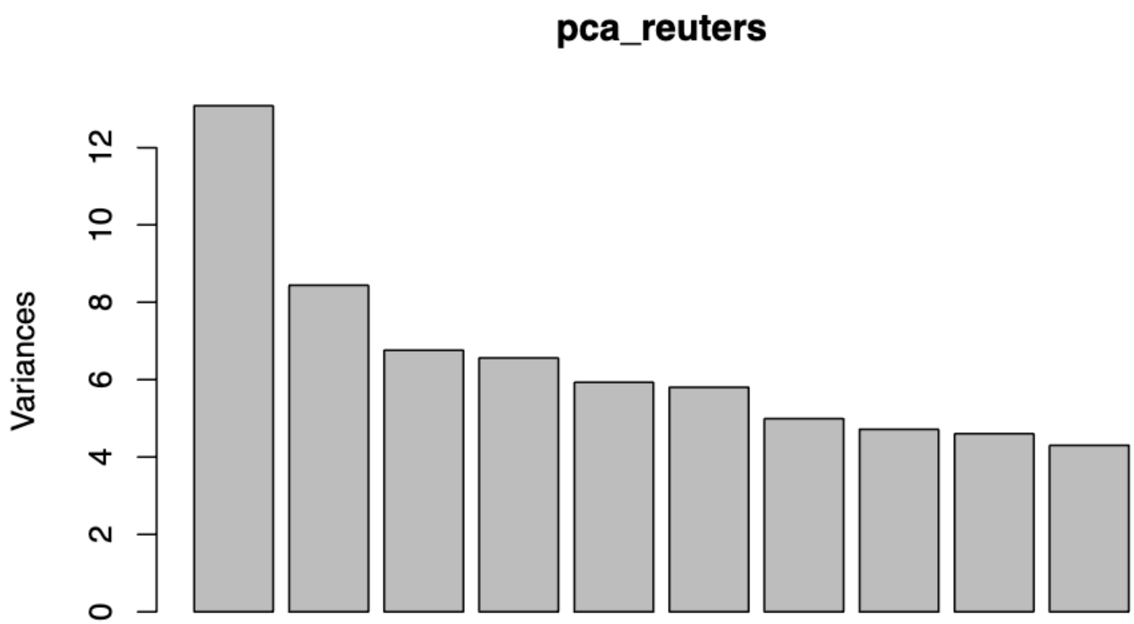
```r
# Remove infrequent terms
DTM_reuters = removeSparseTerms(DTM_reuters, 0.95)
DTM_reuters
```

```
## <<DocumentTermMatrix (documents: 2500, terms: 663)>>
## Non-/sparse entries: 231897/1425603
## Sparsity           : 86%
## Maximal term length: 18
## Weighting          : term frequency (tf)
```

```r
# Create TF-IDF weights
tfidf_reuters = weightTfIdf(DTM_reuters)

# Compare documents
#inspect(tfidf_reuters[1,])

####
# Dimensionality reduction
####

# PCA on term frequencies
X = as.matrix(tfidf_reuters)
summary(colSums(X))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   5.635   7.620   8.513  10.142  36.713
```

```r
scrub_cols = which(colSums(X) == 0)
X = X[,-scrub_cols]

pca_reuters = prcomp(X, rank=2, scale=TRUE)
plot(pca_reuters)
```



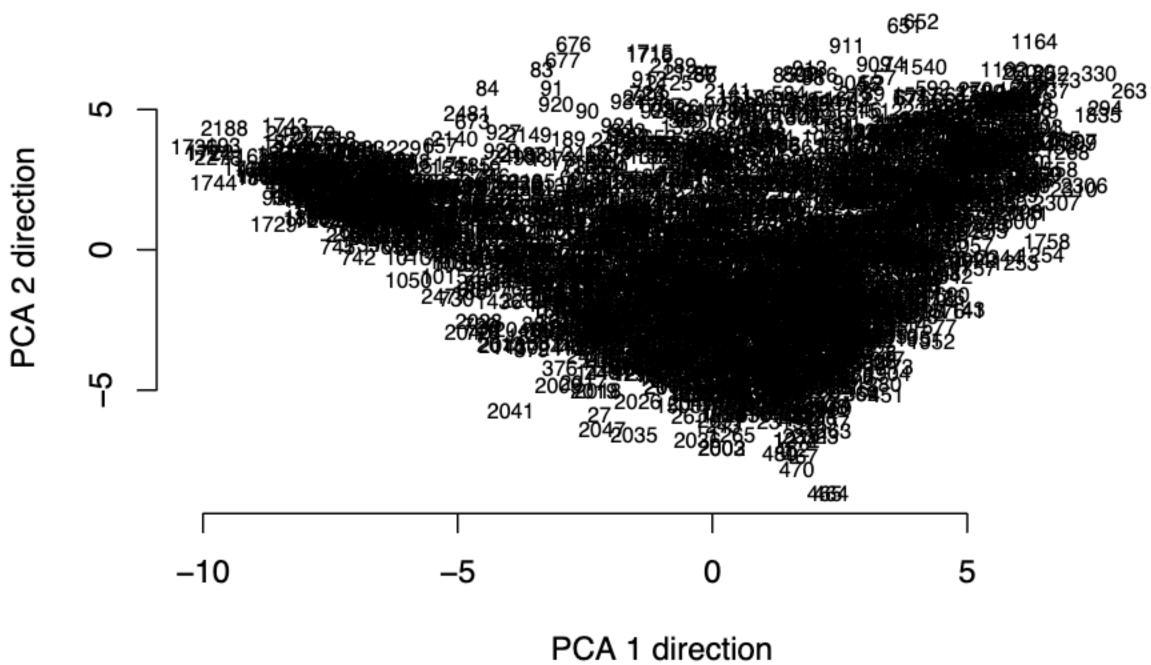

```r
# Look at the loadings
pca_reuters$rotation[order(abs(pca_reuters$rotation[,1]),decreasing=TRUE),1][1:25]
```

```
##     beijing       china     chinese      chinas       share    beijings
## -0.15690786 -0.15057990 -0.13617509 -0.13067457  0.11835012 -0.11527356
##      leader   analysts     million        hong    earnings     analyst
## -0.11484520  0.11372970  0.11350351 -0.11072796  0.10854558  0.10835169
##   political     quarter   communist        kong     profits     percent
## -0.10748459  0.10645335 -0.10615992 -0.10541584  0.10410897  0.10404832
##      cchina    official      profit       human   officials        rule
## -0.10313072 -0.10257797  0.10049975 -0.09884503 -0.09752590 -0.09738376
##      rights
## -0.09648052
```

```r
pca_reuters$rotation[order(abs(pca_reuters$rotation[,2]),decreasing=TRUE),2][1:25]
```

```
##            company               corp     communications               deal
##        -0.12106659        -0.11863170        -0.11710871        -0.11542067
##           forecast          companies            percent             profit
##         0.11060621        -0.10727831         0.10536230         0.10017258
##               rise telecommunications             chinas            results
##         0.09751229        -0.09620052         0.09437319         0.09266848
##           internet            network          customers            figures
##        -0.09184081        -0.09059167        -0.09037584         0.09029571
##           services            beijing               rose              offer
##        -0.08957812         0.08804750         0.08757094        -0.08641543
##            profits                net              lower               half
##         0.08526975         0.08424000         0.08404977         0.08353352
##              china
##         0.08328618
```

```r
# Look at the first two PCs
#pca_reuters$x[,1:2]

plot(pca_reuters$x[,1:2], xlab="PCA 1 direction", ylab="PCA 2 direction", bty="n",
     type='n')
text(pca_reuters$x[,1:2], labels = 1:length(reuters), cex=0.7)
```




```r
# Cluster documents

# define the distance matrix
# using the PCA scores
dist_mat = dist(pca_reuters$x)
tree_reuters = hclust(dist_mat)
#plot(tree_reuters)
clust5 = cutree(tree_reuters, k=5)

# Inspect the clusters
which(clust5 == 5)
```

```
##  151  152  153  154  155  156  157  159  161  162  163  164  165  166  167  168
##  151  152  153  154  155  156  157  159  161  162  163  164  165  166  167  168
##  169  170  171  172  173  174  175  176  177  179  181  182  183  184  185  186
##  169  170  171  172  173  174  175  176  177  179  181  182  183  184  185  186
##  187  188  190  191  192  193  194  195  196  198  199  200  657  670  678  691
##  187  188  190  191  192  193  194  195  196  198  199  200  657  670  678  691
##  692  695  702  703  704  706  708  711  712  713  714  715  716  717  718  719
##  692  695  702  703  704  706  708  711  712  713  714  715  716  717  718  719
##  720  722  725  726  727  731  733  735  736  737  738  739  740  741  742  743
##  720  722  725  726  727  731  733  735  736  737  738  739  740  741  742  743
##  744  745  746  749  750  903  906  907  908  932  933  934  939  940  942  943
##  744  745  746  749  750  903  906  907  908  932  933  934  939  940  942  943
##  945 1021 1025 1026 1028 1036 1355 1356 1357 1358 1361 1362 1367 1368 1372 1393
```

```
##   945 1021 1025 1026 1028 1036 1355 1356 1357 1358 1361 1362 1367 1368 1372 1393
## 1394 1395 1396 1447 1625 1627 1629 1630 1701 1702 1703 1704 1706 1709 1710 1711
## 1394 1395 1396 1447 1625 1627 1629 1630 1701 1702 1703 1704 1706 1709 1710 1711
## 1712 1719 1720 1721 1722 1724 1725 1726 1727 1728 1729 1730 1731 1732 1733 1734
## 1712 1719 1720 1721 1722 1724 1725 1726 1727 1728 1729 1730 1731 1732 1733 1734
## 1735 1736 1737 1738 1741 1743 1744 1745 1747 1748 1749 1750 1851 1852 1853 1855
## 1735 1736 1737 1738 1741 1743 1744 1745 1747 1748 1749 1750 1851 1852 1853 1855
## 1856 1857 1858 1859 1860 1861 1862 1863 1864 1865 1866 1867 1868 1869 1870 1871
## 1856 1857 1858 1859 1860 1861 1862 1863 1864 1865 1866 1867 1868 1869 1870 1871
## 1872 1873 1874 1875 1876 1877 1878 1879 1880 1881 1882 1883 1884 1885 1886 1887
## 1872 1873 1874 1875 1876 1877 1878 1879 1880 1881 1882 1883 1884 1885 1886 1887
## 1888 1889 1890 1892 1893 1894 1895 1896 1897 1898 1899 1900 2113 2119 2120 2121
## 1888 1889 1890 1892 1893 1894 1895 1896 1897 1898 1899 1900 2113 2119 2120 2121
## 2128 2129 2134 2136 2137 2140 2145 2151 2152 2153 2154 2155 2156 2158 2159 2160
## 2128 2129 2134 2136 2137 2140 2145 2151 2152 2153 2154 2155 2156 2158 2159 2160
## 2161 2162 2163 2164 2165 2167 2168 2169 2170 2172 2175 2176 2180 2182 2183 2184
## 2161 2162 2163 2164 2165 2167 2168 2169 2170 2172 2175 2176 2180 2182 2183 2184
## 2185 2186 2188 2190 2191 2192 2193 2194 2195 2196 2197 2198 2199 2200 2251 2253
## 2185 2186 2188 2190 2191 2192 2193 2194 2195 2196 2197 2198 2199 2200 2251 2253
## 2254 2257 2258 2259 2262 2264 2267 2268 2270 2271 2272 2273 2274 2276 2277 2278
## 2254 2257 2258 2259 2262 2264 2267 2268 2270 2271 2272 2273 2274 2276 2277 2278
## 2279 2280 2281 2282 2284 2285 2287 2288 2289 2290 2291 2294 2295 2296 2297 2298
## 2279 2280 2281 2282 2284 2285 2287 2288 2289 2290 2291 2294 2295 2296 2297 2298
## 2299 2300 2452 2463 2465 2470 2471 2472 2473 2474 2475 2476 2479 2480 2482 2483
## 2299 2300 2452 2463 2465 2470 2471 2472 2473 2474 2475 2476 2479 2480 2482 2483
## 2484 2486 2487 2488 2490 2493 2494 2495 2497 2500
## 2484 2486 2487 2488 2490 2493 2494 2495 2497 2500
```

```
content(reuters[[651]])
```

```
##  [1] "Czech consumer prices edged up less than expected in September, pleasantly surprising analysts
##  [2] "The Czech Statistical Bureau (CSU) said on Tuesday that CPI rose 0.3 percent, month-on-month, 
##  [3] "The year-on-year rate now stands 8.9 percent higher while the September sliding 12 month avera
##  [4] "\"Our forecast for the whole year does not change, which is also given by the fact that one mo
##  [5] "\"I don't think that the favourable development of the last two months will repeat in the mont
##  [6] "The government had originally set its average inflation rate target for the whole year at eigh
##  [7] "Analysts said that 0.2 percent monthly increase in the food, tobacco and beverages sector cont
##  [8] "The CSU said prices in the heavily-weighted sector were held back by a 19 percent drop in pota
##  [9] "Kupka welcomed the September result, but said that when more foodstuffs from the domestic harv
## [10] "Radek Maly of Prague's Citibank branch said he was pleased by the September figures, as he had
## [11] "\"We have to wait for what the foodstuffs prices will do in the next months...But if they keep
## [12] "\"The year-on-year rate could get under nine percent at the year's end if the trend continues.
## [13] "He added that other components of the basket also showed positive development, following low i
## [14] "The CSU said prices in the leisure sector dropped by 1.1 percent in the month, thanks mainly t
## [15] "Clothing prices rose 0.9 percent, housing climbed 0.3 percent, and transportation prices remai
## [16] "The CSU in July raised its average inflation forecast for the whole year to 9.0 percent, and t
## [17] "-- Prague Newsroom, 42-2-2423-0003"
```

```
content(reuters[[652]])
```

```
##  [1] "Czech consumer prices rose less than expected in September, but analysts said on Tuesday it re
##  [2] "The Czech Statistical Bureau (CSU) said prices rose 0.3 percent in September versus analysts' 
##  [3] "The year-on-year inflation stood at 8.9 percent, down from 9.6 percent in August, when the mon
```

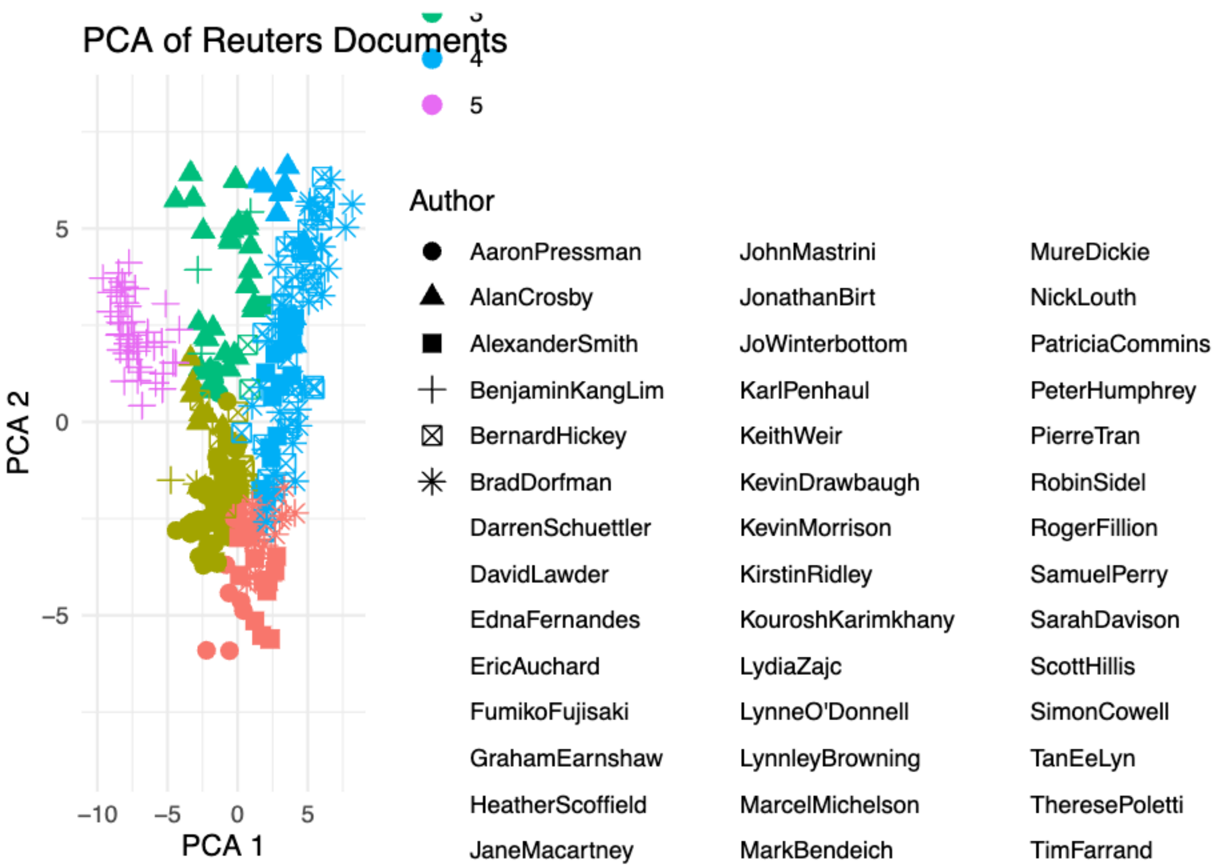
```
##  [4] "\"Our forecast for the whole year does not change, which is also because one month result does
##  [5] "\"I don't think that the favourable developments in the last two months will be repeated in the
##  [6] "The government had originally set its average inflation rate target for the whole year at eigh
##  [7] "Analysts said that the low 0.2 percent monthly increase in the food, tobacco and beverages sect
##  [8] "The CSU said prices in the heavily-weighted sector were held back by a 19 percent drop in pota
##  [9] "Kupka welcomed the September result, but said that when more foodstuffs from the domestic harv
## [10] "Radek Maly of Prague's Citibank branch said he was pleased by the September figures, as he had
## [11] "\"We have to wait for what the foodstuffs prices will do in the next months.But if they keep t
## [12] "\"The year-on-year rate could get under nine percent at the year's end if the trend continues.
## [13] "He added that other components of the basket also showed positive development, following low i
## [14] "The CSU said prices in the leisure sector dropped by 1.1 percent in the month, thanks mainly to
## [15] "Clothing prices rose 0.9 percent, housing climbed 0.3 percent, and transportation prices remai
## [16] "The CSU in July raised its average inflation forecast for the whole year to 9.0 percent, and t
```

```
##########
# Create a data frame for plotting
df <- data.frame(
  PC1 = pca_reuters$x[,1],  # First principal component
  PC2 = pca_reuters$x[,2],  # Second principal component
  Author = as.factor(author_names)  # Author names as factors
)

# Try clustering with authors
df <- data.frame(
  PC1 = pca_reuters$x[,1],  # First principal component
  PC2 = pca_reuters$x[,2],  # Second principal component
  Author = as.factor(author_names)  # Author names as factors
)

# Subset the data by author
authors <- unique(author_names)
clusters_by_author <- list()

for (author in authors) {
  # Subset documents by author
  subset_df <- df[df$Author == author, ]

  # Perform PCA and clustering on this subset
  subset_dist <- dist(subset_df[, 1:2])
  subset_hclust <- hclust(subset_dist)

  # Cut into clusters
  clusters_by_author[[author]] <- cutree(subset_hclust, k = 5)  # Adjust k as needed
}

library(ggplot2)
df$Cluster <- as.factor(clust5)

ggplot(df, aes(x = PC1, y = PC2, color = Cluster, shape = Author)) +
  geom_point(size = 3) +
  labs(title = "PCA of Reuters Documents", x = "PCA 1", y = "PCA 2") +
  theme_minimal()
```

```
## Warning: The shape palette can deal with a maximum of 6 discrete values because more
```

```
## than 6 becomes difficult to discriminate
## i you have requested 50 values. Consider specifying shapes manually if you need
##   that many have them.
```

```
## Warning: Removed 2200 rows containing missing values or values outside the scale range
## (‘geom_point()‘).
```




```r
# Create a table of cluster assignments for each author
table(df$Author, df$Cluster)
```

```
##
##                       1  2  3  4  5
##   AaronPressman       7 40  3  0  0
##   AlanCrosby          0  7 30 13  0
##   AlexanderSmith     17 15  1 17  0
##   BenjaminKangLim     0  2  4  0 44
##   BernardHickey       5  9  2 34  0
##   BradDorfman        15  2  0 33  0
##   DarrenSchuettler    4 16  7 23  0
##   DavidLawder         5 37  0  8  0
##   EdnaFernandes       7 16  0 27  0
##   EricAuchard        22  0  1 27  0
##   FumikoFujisaki      3 15 17 15  0
##   GrahamEarnshaw      0 11 37  2  0
##   HeatherScoffield   13 23  1 13  0
```

```
##    JaneMacartney       0 10  5  0 35
##    JanLopatka          0 23 15  6  6
##    JimGilchrist        1 30 17  2  0
##    JoeOrtiz            5 15  1 29  0
##    JohnMastrini        0 14 15  9 12
##    JonathanBirt        3 10  1 36  0
##    JoWinterbottom     12  3  0 35  0
##    KarlPenhaul         0 39  6  0  5
##    KeithWeir          21  3  1 25  0
##    KevinDrawbaugh     11  2  0 37  0
##    KevinMorrison       9  1  5 35  0
##    KirstinRidley      26  2  0 22  0
##    KouroshKarimkhany 27  2  0 21  0
##    LydiaZajc           0  1 10 39  0
##    LynneO'Donnell      0  4 33  0 13
##    LynnleyBrowning     0 28 16  5  1
##    MarcelMichelson    21 20  0  9  0
##    MarkBendeich        1  9  7 33  0
##    MartinWolk         24  2  2 22  0
##    MatthewBunce        0 15 27  4  4
##    MichaelConnor      22 15  0 13  0
##    MureDickie          0  8  6  0 36
##    NickLouth          33  0  0 17  0
##    PatriciaCommins    11 12  0 27  0
##    PeterHumphrey       0  2  0  0 48
##    PierreTran         15 11  0 24  0
##    RobinSidel         43  0  0  7  0
##    RogerFillion       23 26  0  1  0
##    SamuelPerry        27  6  0 17  0
##    SarahDavison        3 13 19  4 11
##    ScottHillis         0  5  6  0 39
##    SimonCowell        19  8  0 23  0
##    TanEeLyn            0  9  4  1 36
##    TheresePoletti     26  1  0 23  0
##    TimFarrand          3  2  3 42  0
##    ToddNissen         14 21  1 14  0
##    WilliamKazer        0 12 11  3 24
```

Question: What reuters author write about similar topics? If I like the writing of a specific writer in Reuters, what other authors should I read?

Approach: We created a corpus of reuters documents from 42 authors. First, we used a tokenization approach on every word and compiled the words into a document term matrix. After that, we removed all of the infrequent words and commonly used "stop words". Lastly, we created TF-IDF weights for the terms in the DTM to appropriately weigh frequent words within a specific document, yet rare across the corpus. After completing these pre-processing steps, we applied principle component analysis (PCA) to the reduce dimensionality and allow a visualization of the distribution of documents within a 2D space. We kept only 2 principle components to easily view relationships and clusters. To finally view the similarity of authors, we clustered the documents into 5 categories based on the PCA results.

Conclusion: This graph allows us to see the similarity of what authors write about, and where there is crossover. For example, Benjamin Kang Lim is the only writer in cluster 5, while he has a few documents that fall into cluster 3 and are similar to the majority of Alan Crosby's work. Conversely to cluster 5, cluster 4 has several authors with similar documents. Reuters and other publishing companies could use this analysis to recommend similar authors to a reader.