topic_analysis

Charles Hendrickson

5/9/2022

```
library(here)
library(pdftools)
library(quanteda)
library(tm)
library(topicmodels)
library(ldatuning)
library(tidyverse)
library(tidytext)
library(reshape2)
```

Load the data

```
##Topic 6 .Rmd here:https://raw.githubusercontent.com/MaRo406/EDS_231-text-sentiment/main/topic_6.Rmd
#grab data here:
comments_df<-read_csv("https://raw.githubusercontent.com/MaRo406/EDS_231-text-sentiment/main/dat/commen

#comments_df <- read_csv(here("dat", "comments_df.csv")) #if reading from local
```

Now we'll build and clean the corpus

```
epa_corp <- corpus(x = comments_df, text_field = "text")
epa_corp.stats <- summary(epa_corp)
head(epa_corp.stats, n = 25)
```

```
##      Text Types Tokens Sentences
## 1   text1  1196   3973       178
## 2   text2   830   2509       111
## 3   text3   279    571        31
## 4   text4  1745   6904       251
## 5   text5   581   1534        49
## 6   text6   469   1187        53
## 7   text7   424    903        38
## 8   text8  3622  22270       655
## 9   text9   373    717        25
## 10 text10   404    971        42
## 11 text11   710   2190        77
## 12 text12   636   1896        82
## 13 text13   146    206         3
## 14 text14  1124   3197        86
## 15 text15   914   2943        90
## 16 text16    13     45         1
## 17 text17  1043   3190       103
## 18 text18   313    601        24
## 19 text19   152    229         6
```

```
## 20 text20   341    786        35
## 21 text21   211    403        15
## 22 text22   186    322        12
## 23 text23   211    398        14
## 24 text24   325    696        33
## 25 text25  1749   5382       115
##                                                 Document
## 1                                      1_Air Alliance.pdf
## 2                                          10_Bus NEJ.pdf
## 3                                     11_Carlton Ginny.pdf
## 4                                      15_City Project.pdf
## 5                                     16_Corporate EEC.pdf
## 6                                   17_Detriot Sierra Club.pdf
## 7                                      18_District DOE.pdf
## 8                                     19_Earth Justice.pdf
## 9                                          2_Alex Kidd.pdf
## 10                                   20_Elizabeth Mooney.pdf
## 11                                           21_Env COS.pdf
## 12                                      22_Env Def Fund.pdf
## 13                                    23_Env Health Watch.pdf
## 14 24_Env Justice Leadership Forum on Climate Change.pdf
## 15                                    25_Env Law at Duke.pdf
## 16                                     26_Farm worker AF.pdf
## 17                                  27_Farm Worker Justice.pdf
## 18                                       28_Faulker County.pdf
## 19                                        29_First Peoples.pdf
## 20                                   3_Alliance for Metro.pdf
## 21                                          30_Gage Blasi.pdf
## 22                                           31_Gull Leon.pdf
## 23                                       32_Hilary Kramer.pdf
## 24                                    33_Housing Land Advoc.pdf
## 25                                         34_Human rights.pdf
```

```
toks <- tokens(epa_corp, remove_punct = TRUE, remove_numbers = TRUE)
#I added some project-specific stop words here
add_stops <- c(stopwords("en"),"environmental", "justice", "ej", "epa", "public", "comment")
toks1 <- tokens_select(toks, pattern = add_stops, selection = "remove")
```

And now convert to a document-feature matrix

```
dfm_comm<- dfm(toks1, tolower = TRUE)
dfm <- dfm_wordstem(dfm_comm)
dfm <- dfm_trim(dfm, min_docfreq = 2) #remove terms only appearing in one doc (min_termfreq = 10)

print(head(dfm))
```

```
## Document-feature matrix of: 6 documents, 2,781 features (82.75% sparse) and 1 docvar.
##        features
## docs    charl lee deputi associ assist administr usepa offic 2201-a
##   text1     1   2      1      1      6         6     1     7      1
##   text2     1   1      1      4      3         1     0     5      0
##   text3     0   0      0      0      1         0     0     2      0
##   text4     0   0      0      0      1         9     0     1      0
##   text5     4   5      1      1      1         1     0     1      1
##   text6     1   1      1      3      1         3     0     4      0
```

```
##        features
## docs    pennsylvania
##   text1            1
##   text2            0
##   text3            0
##   text4            0
##   text5            1
##   text6            0
## [ reached max_nfeat ... 2,771 more features ]
```

```
#remove rows (docs) with all zeros
sel_idx <- slam::row_sums(dfm) > 0
dfm <- dfm[sel_idx, ]
#comments_df <- dfm[sel_idx, ]
```

We somehow have to come up with a value for k,the number of latent topics present in the data. How do we do this? There are multiple methods. Let's use what we already know about the data to inform a prediction. The EPA has 9 priority areas: Rulemaking, Permitting, Compliance and Enforcement, Science, States and Local Governments, Federal Agencies, Community-based Work, Tribes and Indigenous People, National Measures. Maybe the comments correspond to those areas?

```
k <- 9

topicModel_k9 <- LDA(dfm, k, method="Gibbs", control=list(iter = 500, verbose = 25))
```

```
## K = 9; V = 2781; M = 77
## Sampling 500 iterations!
## Iteration 25 ...
## Iteration 50 ...
## Iteration 75 ...
## Iteration 100 ...
## Iteration 125 ...
## Iteration 150 ...
## Iteration 175 ...
## Iteration 200 ...
## Iteration 225 ...
## Iteration 250 ...
## Iteration 275 ...
## Iteration 300 ...
## Iteration 325 ...
## Iteration 350 ...
## Iteration 375 ...
## Iteration 400 ...
## Iteration 425 ...
## Iteration 450 ...
## Iteration 475 ...
## Iteration 500 ...
## Gibbs sampling completed!
```

```
#nTerms(dfm_comm)

tmResult <- posterior(topicModel_k9)
attributes(tmResult)
```

```
## $names
## [1] "terms"  "topics"
```

```
#nTerms(dfm_comm)
beta <- tmResult$terms   # get beta from results
dim(beta)                # K distributions over nTerms(DTM) terms# lengthOfVocab
```

```
## [1]    9 2781
```

```
terms(topicModel_k9, 10)
```

```
##       Topic 1     Topic 2     Topic 3     Topic 4   Topic 5     Topic 6
##  [1,] "state"     "health"    "communiti" "prison"  "communiti" "issu"
##  [2,] "rule"      "communiti" "enforc"    "facil"   "water"     "agenc"
##  [3,] "impact"    "peopl"     "comment"   "project" "pollut"    "titl"
##  [4,] "communiti" "citi"      "provid"    "texa"    "econom"    "program"
##  [5,] "also"      "comment"   "monitor"   "sourc"   "site"      "right"
##  [6,] "health"    "park"      "includ"    "center"  "will"      "feder"
##  [7,] "pollut"    "access"    "health"    "report"  "polici"    "vi"
##  [8,] "ejscreen"  "can"       "air"       "popul"   "energi"    "work"
##  [9,] "popul"     "fund"      "action"    "organ"   "need"      "includ"
## [10,] "air"       "see"       "requir"    "new"     "increas"   "address"
##       Topic 7     Topic 8   Topic 9
##  [1,] "framework" "state"   "communiti"
##  [2,] "draft"     "permit"  "plan"
##  [3,] "effort"    "feder"   "local"
##  [4,] "action"    "air"     "use"
##  [5,] "state"     "consid"  "comment"
##  [6,] "communiti" "use"     "particip"
##  [7,] "agenc"     "qualiti" "strategi"
##  [8,] "develop"   "meet"    "govern"
##  [9,] "tool"      "train"   "agenda"
## [10,] "epa"       "regul"   "action"
```

Some of those topics seem related to the cross-cutting and additional topics identified in the EPA's response to the public comments:

1. Title VI of the Civil Rights Act of 1964

2.EJSCREEN

3. climate change, climate adaptation and promoting greenhouse gas reductions co-benefits

4. overburdened communities and other stakeholders to meaningfully, effectively, and transparently participate in aspects of EJ 2020, as well as other agency processes

5. utilize multiple Federal Advisory Committees to better obtain outside environmental justice perspectives

6. environmental justice and area-specific training to EPA staff

7. air quality issues in overburdened communities

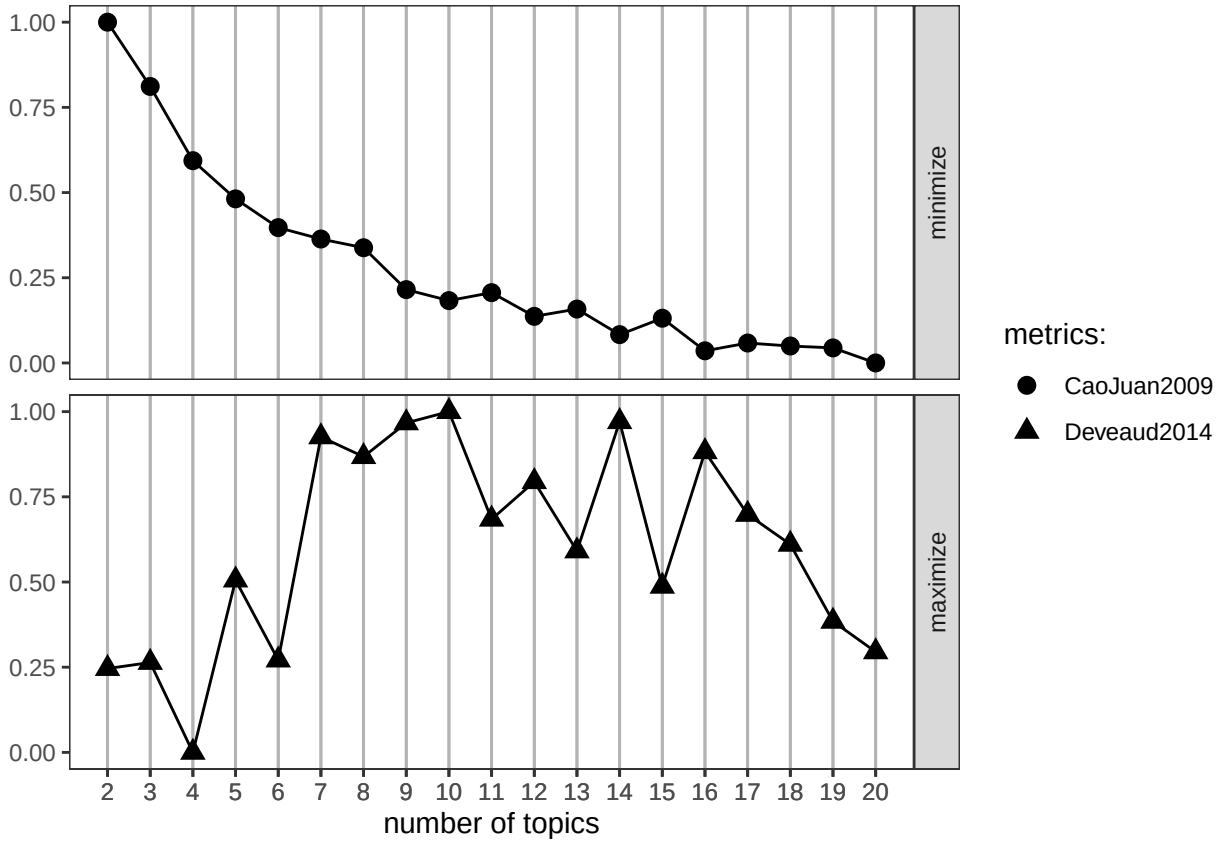
So we could guess that there might be a 16 topics (9 priority + 7 additional). Or we could calculate some metrics from the data.

```
#
result <- FindTopicsNumber(
  dfm,
  topics = seq(from = 2, to = 20, by = 1),
  metrics = c("CaoJuan2009",  "Deveaud2014"),
  method = "Gibbs",
  control = list(seed = 77),
```

```
  verbose = TRUE
)
```

```
## fit models... done.
## calculate metrics:
##   CaoJuan2009... done.
##   Deveaud2014... done.
```

```
FindTopicsNumber_plot(result)
```




```
k <- 7

topicModel_k7 <- LDA(dfm, k, method="Gibbs", control=list(iter = 500, verbose = 25))
```

```
## K = 7; V = 2781; M = 77
## Sampling 500 iterations!
## Iteration 25 ...
## Iteration 50 ...
## Iteration 75 ...
## Iteration 100 ...
## Iteration 125 ...
## Iteration 150 ...
## Iteration 175 ...
## Iteration 200 ...
## Iteration 225 ...
## Iteration 250 ...
## Iteration 275 ...
## Iteration 300 ...
```

```
## Iteration 325 ...
## Iteration 350 ...
## Iteration 375 ...
## Iteration 400 ...
## Iteration 425 ...
## Iteration 450 ...
## Iteration 475 ...
## Iteration 500 ...
## Gibbs sampling completed!
```

```
tmResult <- posterior(topicModel_k7)
terms(topicModel_k7, 10)
```

```
##       Topic 1     Topic 2   Topic 3     Topic 4   Topic 5     Topic 6
##  [1,] "state"     "prison"  "state"     "agenc"   "pollut"    "communiti"
##  [2,] "program"   "peopl"   "permit"    "right"   "communiti" "comment"
##  [3,] "draft"     "health"  "use"       "civil"   "state"     "includ"
##  [4,] "framework" "project" "communiti" "titl"    "rule"      "enforc"
##  [5,] "agenc"     "citi"    "consid"    "vi"      "impact"    "monitor"
##  [6,] "polici"    "park"    "like"      "plan"    "popul"     "air"
##  [7,] "feder"     "law"     "help"      "issu"    "health"    "requir"
##  [8,] "epa"       "can"     "comment"   "work"    "also"      "action"
##  [9,] "will"      "center"  "organ"     "address" "air"       "permit"
## [10,] "goal"      "green"   "grant"     "one"     "must"      "data"
##       Topic 7
##  [1,] "communiti"
##  [2,] "local"
##  [3,] "water"
##  [4,] "agenda"
##  [5,] "comment"
##  [6,] "econom"
##  [7,] "develop"
##  [8,] "effort"
##  [9,] "juli"
## [10,] "framework"
```

```
theta <- tmResult$topics
beta <- tmResult$terms
vocab <- (colnames(beta))
```

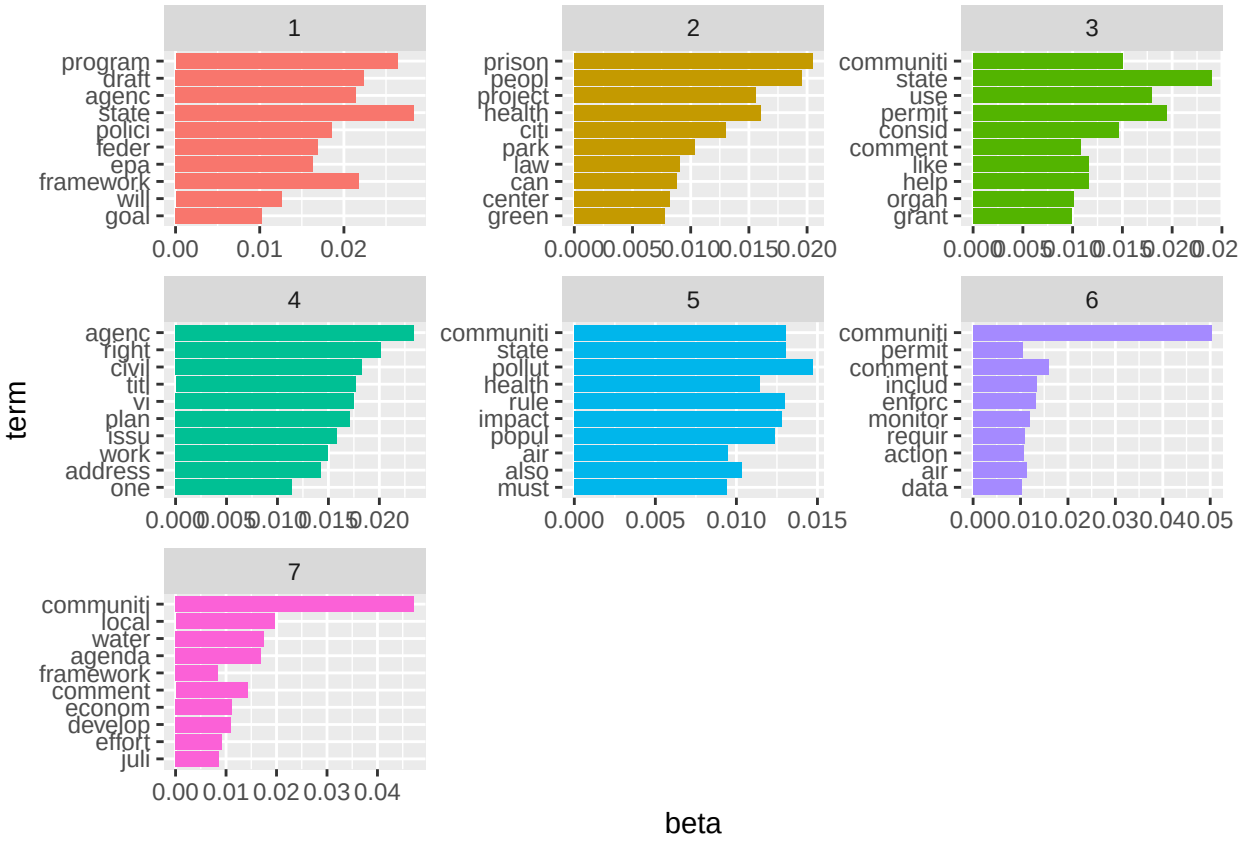
There are multiple proposed methods for how to measure the best k value. You can go down the rabbit hole here

```
comment_topics <- tidy(topicModel_k7, matrix = "beta")

top_terms <- comment_topics %>%
  group_by(topic) %>%
  top_n(10, beta) %>%
  ungroup() %>%
  arrange(topic, -beta)

top_terms
```

```
## # A tibble: 70 x 3
##    topic term        beta
##    <int> <chr>      <dbl>
```

```
##  1     1 state     0.0283
##  2     1 program   0.0264
##  3     1 draft     0.0224
##  4     1 framework 0.0218
##  5     1 agenc     0.0214
##  6     1 polici    0.0186
##  7     1 feder     0.0169
##  8     1 epa       0.0163
##  9     1 will      0.0126
## 10     1 goal      0.0102
## # ... with 60 more rows
```

```
top_terms %>%
  mutate(term = reorder(term, beta)) %>%
  ggplot(aes(term, beta, fill = factor(topic))) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free") +
  coord_flip()
```



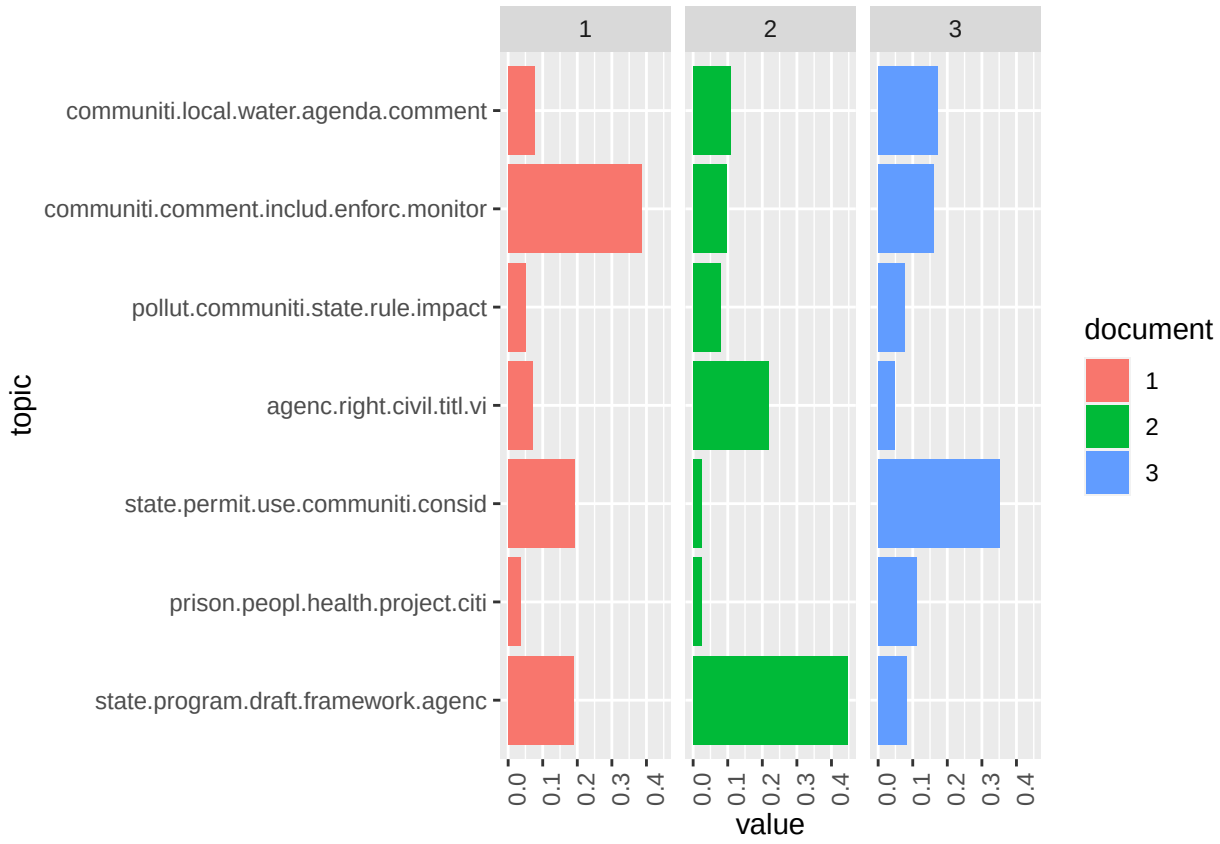
Let's assign names to the topics so we know what we are working with. We can name them by their top terms

```
top5termsPerTopic <- terms(topicModel_k7, 5)
topicNames <- apply(top5termsPerTopic, 2, paste, collapse=" ")
```

We can explore the theta matrix, which contains the distribution of each topic over each document

```
exampleIds <- c(1, 2, 3)
N <- length(exampleIds)
```

```
#lapply(epa_corp[exampleIds], as.character) #uncomment to view example text
# get topic proportions form example documents
topicProportionExamples <- theta[exampleIds,]
colnames(topicProportionExamples) <- topicNames
vizDataFrame <- melt(cbind(data.frame(topicProportionExamples), document=factor(1:N)), variable.name =
ggplot(data = vizDataFrame, aes(topic, value, fill = document), ylab = "proportion") +
  geom_bar(stat="identity") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  coord_flip() +
  facet_wrap(~ document, ncol = N)
```



Here's a neat JSON-based model visualizer

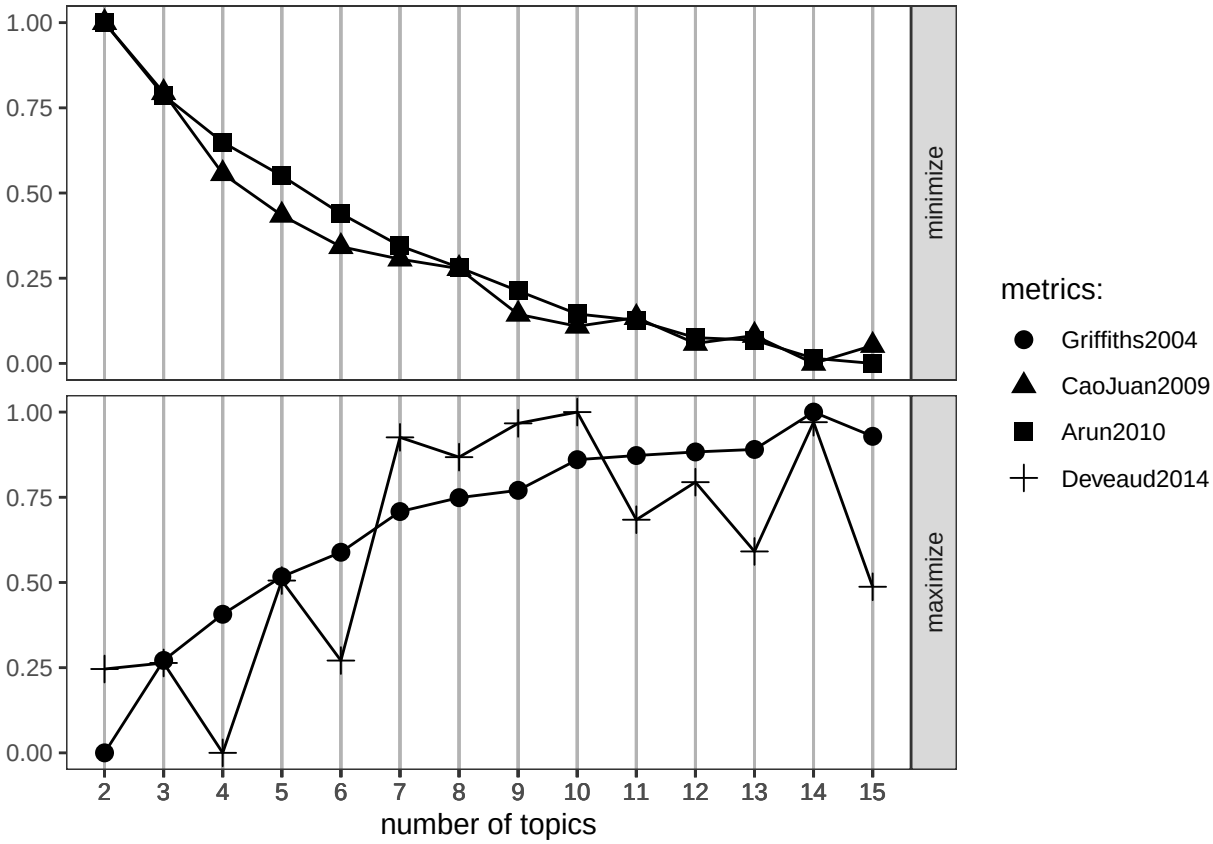
```
library(LDAvis)
library("tsne")
svd_tsne <- function(x) tsne(svd(x)$u)
json <- createJSON(
  phi = tmResult$terms,
  theta = tmResult$topics,
  doc.length = rowSums(dfm),
  vocab = colnames(dfm),
  term.frequency = colSums(dfm),
  mds.method = svd_tsne,
  plot.opts = list(xlab="", ylab="")
)
serVis(json)
```

###Three additional models

```
result <- FindTopicsNumber(
  dfm,
  topics = seq(from = 2, to = 15, by = 1),
  metrics = c("Griffiths2004", "CaoJuan2009", "Arun2010", "Deveaud2014"),
  method = "Gibbs",
  control = list(seed = 77),
  mc.cores = 2L,
  verbose = TRUE
)
```

```
## fit models... done.
## calculate metrics:
##   Griffiths2004... done.
##   CaoJuan2009... done.
##   Arun2010... done.
##   Deveaud2014... done.
```

```
FindTopicsNumber_plot(result)
```



From this plot, we can conclude that the optimal number of topics for Griffiths2004 is 14,for Deveaud2014 the optimal number of topics peaks at 10 but also 14, and for CaoJuan2009 and Arun2010 it is also 14 topics. Thus based on these metrics, 14 is the optimal number of topics.

Metrics used for Comparison Arun2010: The measure is computed in terms of symmetric KL-Divergence of salient distributions that are derived from these matrix factor and is observed that the divergence values are higher for non-optimal number of topics (maximize)

CaoJuan2009: method of adaptively selecting the best LDA model based on density.(minimize)

Griffths: To evaluate the consequences of changing the number of topics T, used the Gibbs sampling algorithm to obtain samples from the posterior distribution over z at several choices of T(minimize)

Assignment:

Either:

A) continue on with the analysis we started:

Run three more models and select the overall best value for k (the number of topics) - include some justification for your selection: theory, FindTopicsNumber() optimization metrics, interpretability, LDAvis

OR

B) use the data you plan to use for your final project:

Prepare the data so that it can be analyzed in the topicmodels package

Run three more models and select the overall best value for k (the number of topics) - include some justification for your selection: theory, FindTopicsNumber() optimization metrics, interpretability, LDAvis