EDS 231: Assignment 1

Scout Leonard

05/08/2022

Load Libraries

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

Set Up

Read in data:

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

Corpus:

```
epa_corp <- corpus(x = comments_df, text_field = "text")

epa_corp.stats <- summary(epa_corp)

head(epa_corp.stats, n = 10) %>%
  kable()
```

Text	Types	Tokens	Sentences	Document
text1	1196	3973	178	1_Air Alliance.pdf
text2	830	2509	111	10_Bus NEJ.pdf
text3	279	571	31	11_Carlton Ginny.pdf
text4	1745	6904	251	15_City Project.pdf
text5	581	1534	49	16_Corporate EEC.pdf
text6	469	1187	53	17_Detriot Sierra Club.pdf
text7	424	903	38	18_District DOE.pdf
text8	3622	22270	655	19_Earth Justice.pdf
text9	373	717	25	2_Alex Kidd.pdf

Text	Types	Tokens	Sentences	Document
text10	404	971	42	20_Elizabeth Mooney.pdf

Tokenize Corpus:

```
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")
```

Convert tokens to a document frame matrix:

```
#construct dfm from tokens
dfm_comm <- dfm(toks1, tolower = TRUE)

#apply a stemmer to words in dfm
dfm <- dfm_wordstem(dfm_comm)

#remove terms only appearing in one doc (min_termfreq = 10)
dfm <- dfm_trim(dfm, min_docfreq = 2)

#remove rows (docs) with all zeros
sel_idx <- slam::row_sums(dfm) > 0

#comments_df <- dfm[sel_idx, ]
dfm <- dfm[sel_idx, ]
```

LDA Modelling:

Write the model:

```
topicModel_k9 <- LDA(dfm,
                     k = 9,
                     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!
```

Return results:

```
tmResult <- posterior(topicModel_k9)

beta <- tmResult$terms #get beta from results

terms(topicModel_k9, 10)
```

```
##       Topic 1      Topic 2     Topic 3     Topic 4     Topic 5     Topic 6
##  [1,] "communiti"  "impact"    "communiti" "communiti" "health"    "prison"
##  [2,] "framework"  "pollut"    "plan"      "enforc"    "citi"      "popul"
##  [3,] "draft"      "state"     "local"     "air"       "peopl"     "facil"
##  [4,] "effort"     "rule"      "strategi"  "monitor"   "park"      "site"
##  [5,] "develop"    "popul"     "action"    "action"    "right"     "industri"
##  [6,] "action"     "also"      "govern"    "pollut"    "includ"    "peopl"
##  [7,] "agenda"     "health"    "use"       "assess"    "communiti" "sourc"
##  [8,] "overburden" "communiti" "particip"  "report"    "green"     "energi"
##  [9,] "support"    "air"       "juli"      "includ"    "climat"    "comment"
## [10,] "water"      "ejscreen"  "subject"   "comment"   "law"       "project"
##       Topic 7   Topic 8    Topic 9
##  [1,] "program" "agenc"    "state"
##  [2,] "state"   "right"    "permit"
##  [3,] "feder"   "titl"     "consid"
##  [4,] "polici"  "issu"     "air"
##  [5,] "will"    "civil"    "use"
##  [6,] "agenc"   "vi"       "comment"
##  [7,] "includ"  "plan"     "carolina"
##  [8,] "import"  "address"  "opportun"
##  [9,] "epa"     "farmwork" "organ"
## [10,] "issu"    "act"      "grant"
```

Visualize results:

```
#load libraries
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)
```

Assignment

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.

Model 1: `k = 5`

Write model:

```
topicModel_k5 <- LDA(dfm,
                     k = 5,
                     method = "Gibbs",
                     control = list(iter = 500, verbose = 25))
```

```
## K = 5; 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!
```

Return results:

```
tmResult_k5 <- posterior(topicModel_k5)

beta <- tmResult_k5$terms #get beta from results

terms(topicModel_k5, 10)
```

```
##       Topic 1     Topic 2     Topic 3     Topic 4     Topic 5
##  [1,] "communiti" "state"     "communiti" "pollut"    "right"
##  [2,] "plan"      "framework" "enforc"    "impact"    "civil"
##  [3,] "local"     "draft"     "includ"    "communiti" "health"
##  [4,] "work"      "agenc"     "action"    "health"    "prison"
##  [5,] "water"     "permit"    "comment"   "state"     "peopl"
##  [6,] "comment"   "feder"     "monitor"   "also"      "vi"
##  [7,] "agenda"    "program"   "air"       "rule"      "project"
##  [8,] "strategi"  "effort"    "region"    "popul"     "titl"
##  [9,] "govern"    "consid"    "need"      "air"       "law"
## [10,] "need"      "issu"      "provid"    "guidanc"   "citi"
```

Visualize Results:

```
json_k5 <- createJSON(
  phi = tmResult_k5$terms,
  theta = tmResult_k5$topics,
  doc.length = rowSums(dfm),
  vocab = colnames(dfm),
  term.frequency = colSums(dfm),
  mds.method = svd_tsne,
  plot.opts = list(xlab="",
                   ylab="")
)

serVis(json_k5)
```

Model 2: `k = 20`

Write model:

```
topicModel_k20 <- LDA(dfm,
                      k = 20,
                      method = "Gibbs",
                      control = list(iter = 500, verbose = 25))
```

```
## K = 20; 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!
```

Return results:

```
tmResult_k20 <- posterior(topicModel_k20)

beta <- tmResult_k20$terms #get beta from results

terms(topicModel_k20, 10)
```

```
##       Topic 1   Topic 2    Topic 3  Topic 4    Topic 5      Topic 6
##  [1,] "right"   "program"  "park"   "data"     "water"      "communiti"
##  [2,] "civil"   "state"    "peopl"  "particip" "communiti"  "health"
##  [3,] "vi"      "polici"   "citi"   "citizen"  "clean"      "provid"
##  [4,] "titl"    "feder"    "green"  "texa"     "local"      "requir"
##  [5,] "agenc"   "tribe"    "see"    "comment"  "citi"       "use"
##  [6,] "issu"    "epa"      "health" "like"     "job"        "plan"
##  [7,] "feder"   "follow"   "space"  "will"     "site"       "comment"
##  [8,] "address" "principl" "color"  "group"    "june"       "inform"
##  [9,] "act"     "govern"   "includ" "region"   "brownfield" "must"
## [10,] "plan"    "regul"    "law"    "access"   "fund"       "includ"
##       Topic 7   Topic 8     Topic 9     Topic 10        Topic 11   Topic 12
##  [1,] "prison"  "recommend" "work"      "juli"          "farmwork" "plan"
##  [2,] "facil"   "mercuri"   "individu"  "infrastructur" "pesticid" "communiti"
##  [3,] "center"  "implement" "econom"    "access"        "enforc"   "use"
##  [4,] "project" "good"      "re"        "natur"         "work"     "mani"
##  [5,] "sourc"   "hous"      "energi"    "energi"        "agenc"    "land"
##  [6,] "impact"  "presid"    "can"       "pipelin"       "exposur"  "resourc"
##  [7,] "popul"   "measur"    "year"      "ohio"          "found"    "particip"
##  [8,] "report"  "see"       "will"      "polici"        "includ"   "strategi"
##  [9,] "can"     "level"     "peopl"     "gas"           "u."       "collabor"
## [10,] "peopl"   "pleas"     "underserv" "project"       "advanc"   "engag"
##       Topic 13    Topic 14   Topic 15    Topic 16     Topic 17    Topic 18
##  [1,] "communiti" "permit"   "program"   "framework"  "communiti" "help"
##  [2,] "local"     "state"    "agenc"     "draft"      "monitor"   "subject"
##  [3,] "agenda"    "consid"   "director"  "effort"     "enforc"    "sent"
##  [4,] "sustain"   "carolina" "action"    "state"      "permit"    "action"
##  [5,] "support"   "use"      "health"    "communiti"  "air"       "lung"
##  [6,] "injustic"  "feder"    "stakehold" "agenc"      "avail"     ">"
##  [7,] "provid"    "grant"    "develop"   "epa"        "action"    "tai"
##  [8,] "comment"   "air"      "committe"  "comment"    "complianc" "ejstrategi"
##  [9,] "power"     "issu"     "help"      "overburden" "report"    "strategi"
## [10,] "govern"    "north"    "incorpor"  "develop"    "pollut"    "comment"
##       Topic 19   Topic 20
##  [1,] "state"    "communiti"
##  [2,] "asthma"   "pollut"
##  [3,] "popul"    "comment"
##  [4,] "rule"     "air"
##  [5,] "ejscreen" "reduc"
##  [6,] "also"     "polici"
```

```
##  [7,] "guidanc"  "new"
##  [8,] "avail"    "will"
##  [9,] "ozon"     "protect"
## [10,] "impact"   "develop"
```

Visualize results:

```
json_k20 <- createJSON(
  phi = tmResult_k20$terms,
  theta = tmResult_k20$topics,
  doc.length = rowSums(dfm),
  vocab = colnames(dfm),
  term.frequency = colSums(dfm),
  mds.method = svd_tsne,
  plot.opts = list(xlab="",
                   ylab="")
)

serVis(json_k20)
```

Model 3: `k = 3`

Write model:

```
topicModel_k3 <- LDA(dfm,
                     k = 3,
                     method = "Gibbs",
                     control = list(iter = 500, verbose = 25))
```

```
## K = 3; 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!
```

Return results:

```
tmResult_k3 <- posterior(topicModel_k3)

beta <- tmResult_k3$terms #get beta from results

terms(topicModel_k3, 3)
```

```
##      Topic 1     Topic 2     Topic 3
## [1,] "communiti" "right"     "communiti"
## [2,] "pollut"    "communiti" "state"
## [3,] "impact"    "health"    "framework"
```

Visualize results:

```
json_k3 <- createJSON(
  phi = tmResult_k3$terms,
  theta = tmResult_k3$topics,
  doc.length = rowSums(dfm),
  vocab = colnames(dfm),
  term.frequency = colSums(dfm),
  mds.method = svd_tsne,
  plot.opts = list(xlab="",
                   ylab="")
)

serVis(json_k3)
```

I think that overall, my model with k = 3 was the best number of topics for any of the models I tried. I think this based on the ratio of overall term freqnecy to estimated term frequency with the selected topic from the model, as output by the `serVis()` function. For the model with `k = 3`, these seemed to be the closest compared to the models with `k = 5` and `k = 20`.

Also, there are fewer topics with overlapping dimensions according to the intertopic distance map for the `k = 3` model compared to the other two models. `k = 5` similarly had more intertopic distance, but the term salience was not as strong as described in the previous paragraph.