Topic Models for Anushka

Preliminaries

Computing environment

We will use the following R packages.

```
library("fpc")
library("stringi")
library("mallet")
library("coreNLP")
library("quantedaData")
```

To ensure consistent runs, we set the seed before performing any analysis.

```
set.seed(0)
```

Data

Load 57 US inaugural addresses (US presidentes) data. Keep 20th century speeches only.

```
data("inaugTexts")
# keep 20th century presidents only
inaugTexts <- inaugTexts[29:length(inaugTexts)]
names(inaugTexts)
```

```
 [1] "1901-McKinley"   "1905-Roosevelt"  "1909-Taft"       "1913-Wilson"     "1917-Wilson"
 [6] "1921-Harding"    "1925-Coolidge"   "1929-Hoover"     "1933-Roosevelt"  "1937-Roosevelt"
[11] "1941-Roosevelt"  "1945-Roosevelt"  "1949-Truman"     "1953-Eisenhower" "1957-Eisenhower"
[16] "1961-Kennedy"    "1965-Johnson"    "1969-Nixon"      "1973-Nixon"      "1977-Carter"
[21] "1981-Reagan"     "1985-Reagan"     "1989-Bush"       "1993-Clinton"    "1997-Clinton"
[26] "2001-Bush"       "2005-Bush"       "2009-Obama"      "2013-Obama"
```

```
names <- stri_sub(names(inaugTexts),6)
year <- substr(names(inaugTexts),1,4)
```

Data preprocessing

Clean Text

For each document in the corpus, remove all punctuation and numbers, and case-fold the text.

```
# convert to canonical case (lowercase for most languages);
# normalize the unicode representation
text <- stringi::stri_trans_nfkc_casefold(inaugTexts)
# remove punctuation and digits
text <- gsub("[[:punct:][:digit:]]", "", text)
```

Feature selection

Rather than fitting the topic model to the entire text, we fit the model to just the lemmas of the non-proper nouns. The following code segment filters the text using the POS-tagged and lemmatized corpus. For each document, we build a long text string containing all of the selected words, separated by spaces.

Anushka: You will have to download and install either `coreNLP` or `openNLP` to do this. CoreNLP is more involved but has slightly better results. Unfortunately, neither of them is fun to install or work with...

```
coreNLP::initCoreNLP(annotators=c("tokenize", "ssplit", "pos", "lemma"))

bagOfWords <- rep("", length(text))
for (j in seq_along(text)) {
    anno <- coreNLP::annotateString(text[j])
    token <- getToken(anno)
    theseLemma <- token$lemma[token$POS %in% c("NNS", "NN")]
    bagOfWords[j] <- paste(theseLemma, collapse=" ")
}
```

To filter out stopwords, we need to store the words in a file. Since we already have used POS tags to filter out stop words, we only need to worry about initials that may have been mistaken for non-proper nouns by the tagger.

```
tf <- tempfile()
writeLines(c(letters, LETTERS), tf)
```

Fitting

Fit two topic models with 8 topics each.

```
run <- list()
for (i in 1:2) {
instance <- mallet.import(id.array=names(inaugTexts), text.array=bagOfWords,
                          stoplist.file=tf)
tm <- MalletLDA(num.topics=8)
tm$loadDocuments(instance)
tm$setAlphaOptimization(20, 50)
tm$train(200)
tm$maximize(10)

# pull out
topics <- mallet.doc.topics(tm, smoothed=TRUE, normalized=TRUE)
words <- mallet.topic.words(tm, smoothed=TRUE, normalized=TRUE)
vocab <- tm$getVocabulary()

# save model results in list
run[[i]] <- list(topics=topics, words=words, vocab=vocab)
}
```

Results

Here are the top 10 words in each of the 8 topics:

```
run1 <- run[[1]]; run2 <- run[[2]]
# get top 10 words
(res1 <- apply(run1$words, 1,
               function(v) run1$vocab[order(v, decreasing=TRUE)[1:10]]))
```

```
      [,1]           [,2]        [,3]       [,4]             [,5]         [,6]        [,7]
 [1,] "congress"     "principle" "freedom"  "people"         "government" "man"       "nation"
 [2,] "business"     "hope"      "liberty"  "nation"         "law"        "life"      "america"
 [3,] "government"   "country"   "security" "world"          "country"    "power"     "world"
 [4,] "state"        "faith"     "state"    "peace"          "progress"   "purpose"   "people"
 [5,] "executive"    "strength"  "citizen"  "war"            "justice"    "spirit"    "time"
 [6,] "race"         "force"     "rights"   "action"         "party"      "thing"     "government"
 [7,] "tariff"       "law"       "country"  "state"          "system"     "democracy" "american"
 [8,] "policy"       "interest"  "weapon"   "effort"         "freedom"    "land"      "today"
 [9,] "legislation"  "duty"      "side"     "way"            "citizen"    "justice"   "peace"
[10,] "court"        "courage"   "effort"   "responsibility" "force"      "heart"     "god"
      [,8]
 [1,] "child"
 [2,] "citizen"
 [3,] "friend"
 [4,] "word"
 [5,] "economy"
 [6,] "value"
 [7,] "journey"
 [8,] "generation"
 [9,] "idea"
[10,] "crisis"
```

```
(res2 <- apply(run2$words, 1,
               function(v) run2$vocab[order(v, decreasing=TRUE)[1:10]]))
```

```
      [,1]             [,2]       [,3]       [,4]        [,5]       [,6]         [,7]
 [1,] "government"     "liberty"  "tax"      "man"       "people"   "world"      "child"
 [2,] "justice"        "change"   "economy"  "thing"     "nation"   "nation"     "american"
 [3,] "progress"       "justice"  "side"     "life"      "country"  "america"    "citizen"
 [4,] "republic"       "law"      "pledge"   "spirit"    "peace"    "freedom"    "generation"
 [5,] "party"          "land"     "hero"     "day"       "power"    "people"     "land"
 [6,] "law"            "enemy"    "state"    "task"      "war"      "time"       "today"
 [7,] "system"         "call"     "price"    "hand"      "purpose"  "government" "courage"
 [8,] "responsibility" "science"  "cost"     "moment"    "man"      "god"        "value"
 [9,] "civilization"   "counsel"  "struggle" "part"      "force"    "hope"       "father"
[10,] "order"          "man"      "group"    "democracy" "effort"   "today"      "journey"
      [,8]
 [1,] "government"
 [2,] "law"
 [3,] "policy"
 [4,] "congress"
 [5,] "state"
 [6,] "business"
 [7,] "executive"
 [8,] "time"
 [9,] "obligation"
[10,] "condition"
```

This is what we were talking about in the park: Comparing these two outputs is a little hard.

Good luck.