```
############################################
### TextMining Lecture 08              #####
### Subject: Text Similarity Analysis  #####
### Developed by. KKIM                 #####
############################################

library(dplyr)
library(magrittr)
library(stringr)
library(tm)
library(textstem)

library("officer")
jfk.speech <-
  read_docx("R file/R file_LEC08/jfk_speech_doc.docx")

jfk.speech.sum <-
  jfk.speech %>%
  docx_summary %>%
  rename(doc_id = doc_index) %>%
  select(doc_id, text)
jfk.speech.sum %>% head(1)
jfk.speech.sum %>% nrow

jfk.speech.corp <-
  jfk.speech.sum %>%
  DataframeSource %>%
  Corpus %>%
  tm_map(removePunctuation) %>%
  tm_map(removeNumbers) %>%
  tm_map(stripWhitespace) %>%
  tm_map(content_transformer(lemmatize_strings)) %>%
  tm_map(content_transformer(tolower))
jfk.speech.sum[2,]
jfk.speech.corp[[2]]$content

jfk_speech.dtm <- jfk.speech.corp %>%
  DocumentTermMatrix(control =
                       list(wordLengths=c(1, Inf)))
jfk_speech.dtm %>% inspect
jfk_speech.dtm %>% colnames
jfk_speech.dtm %>% as.matrix %>%
  colSums %>% sort

jfk_speech.tdm <- jfk.speech.corp %>%
  TermDocumentMatrix(control =
                       list(wordLengths=c(1, Inf)))
jfk_speech.tdm %>% inspect
jfk_speech.tdm %>% colnames

#
term.euc <-
  jfk_speech.tdm %>%
  as.matrix %>%
  proxy::dist(method = "euclidean") %>%
  as.matrix
term.euc[1:5,1:5]
```

```
doc.euc <-
  jfk_speech.dtm %>%
  as.matrix %>%
  proxy::dist(method = "euclidean") %>%
  as.matrix
doc.euc[1:5,1:5]

# Similarity between two terms
library(proxy)
term.cos <-
  jfk_speech.tdm %>%
  as.matrix %>%
  proxy::dist(method = "cosine") %>%
  as.matrix
term.cos[1:5,1:5]

doc.cos <-
  jfk_speech.dtm %>%
  as.matrix %>%
  proxy::dist(method = "cosine") %>%
  as.matrix
doc.cos[1:5,1:5]

# correlation beteen terms
cor.test(as.vector(jfk_speech.dtm[,"space"]),
         as.vector(jfk_speech.dtm[,"knowledge"]))

jfk_speech.dtm %>%
  findAssocs("space", 0.4)
jfk_speech.tdm %>%
  findAssocs("space", 0.4)

jfk_speech.dtm %>%
  findAssocs("knowledge", 0.6)
jfk_speech.tdm %>%
  findAssocs("knowledge", 0.6)

# correlation between sentences
jfk.speech.sum[23,]
jfk.speech.sum[25,]

jfk_speech.tdm %>% as.matrix
cor.test(as.vector(jfk_speech.tdm[,"23"]),
         as.vector(jfk_speech.tdm[,"25"]))

# correlation matrix
doc.cor <-
  matrix(NA, nrow = length(colnames(jfk_speech.tdm)),
         ncol = length(colnames(jfk_speech.tdm)))
for(i in 1:length(colnames(jfk_speech.tdm))){
  for(j in 1:length(colnames(jfk_speech.tdm))){
    doc.cor[i,j] <-
      cor.test(as.vector(jfk_speech.tdm[,i]),
               as.vector(jfk_speech.tdm[,j]))$est
  }
}
head(doc.cor)
```

```
###### Association Rule Analysis / Apriori algorithm
library('arules')
# detach("package:tm")
mydoc <- list(
  c("love","passion","sweet"),
  c("love","passion","hungry"),
  c("love","anger","sweet"),
  c("anger","disgrace","passion")
)
mydoc
mydoc %>%
  as('transactions') %>%
  inspect

mydoc.ap <-
  mydoc %>%
  apriori(parameter=list(supp=0, conf=0))
mydoc.ap %>%
  inspect

mydoc.ap.1 <-
  mydoc %>%
  apriori(parameter=list(supp=0.4, conf=0.7))
mydoc.ap.1 %>%
  inspect

mydoc.ap.2 <-
  mydoc %>%
  apriori(parameter=list(supp=0.1, conf=0.1),
          appearance=list(rhs="love"))
mydoc.ap.2 %>%
  inspect

mydoc.ap.3 <-
  mydoc %>%
  apriori(parameter=list(supp=0.1, conf=0.1),
          appearance=list(lhs="love"))
mydoc.ap.3 %>%
  inspect

###### Clustering Analysis
library(tm)
jfk_speech.dtm %>%
  inspect
jfk_speech.dtm %>%
  stats::dist(method="euclidean") %>%
  as.matrix %>%
  .[1:10,1:10]
jfk_speech.tdm %>%
  stats::dist(method="euclidean") %>%
  as.matrix %>%
  .[1:10,1:10]

library(factoextra)
jfk_speech.dtm.cluster <-
  jfk_speech.dtm %>%
  stats::dist(method="euclidean") %>%
  hclust(method="ward.D2")
```

```
jfk_speech.dtm.cluster %>%
  fviz_dend
jfk_speech.tdm %>%
  dist

jfk_speech.dtm.cluster %>%
  cutree(k=2)
jfk_speech.dtm.cluster %>%
  fviz_dend(k=2)

jfk.speech.sum[36,]
jfk.speech.sum[77,]

jfk_speech.dtm %>%
  as.matrix %>%
  .[36,]

jfk.speech.sum[34,]
jfk.speech.sum[50,]

# Crude oil
data("crude")
library(tm)
library(textstem)
crude.cleaned <- crude %>%
  tm_map(removePunctuation) %>%
  tm_map(removeNumbers) %>%
  tm_map(removeWords, stopwords('en')) %>%
  tm_map(stripWhitespace) %>%
  tm_map(content_transformer(lemmatize_strings)) %>%
  tm_map(content_transformer(tolower))
crude.cleaned[[1]]$content

crude.dtm.dist <-
  crude.cleaned %>%
  DocumentTermMatrix() %>%
  stats::dist(method="euclidean")
crude.dtm.dist

crude.dtm.cluster <-
  crude.dtm.dist %>%
  hclust(method="ward.D2")
crude.dtm.cluster
crude.dtm.cluster %>%
  fviz_dend

crude.dtm.cluster %>%
  fviz_dend(k=3)
```