Lab 3

Hong Xu

September 7, 2015

- Name: Hong Xu
- Date: September 7, 2015
- Team Member: Carolyn Zhang

1. load and inspect the text data

```
require(tm)
```

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

```
load("lab3.Rdata")
# determine how many documents are present
length(shakespeare)
```

```
## [1] 182
```

2. create a corpus

```
corp <- Corpus(VectorSource(shakespeare))
```

3. preprocess the corpus and store it to a matrix

```
# covert to lower case
corp <- tm_map(corp, content_transformer(tolower))
# puntual removal
corp <- tm_map(corp, removePunctuation)
# number removal
corp <- tm_map(corp, removeNumbers)
# stemming
require(SnowballC)
```

```
## Loading required package: SnowballC
```

```
corp <- tm_map(corp, stemDocument)

# create a document term matrix
dtm <- DocumentTermMatrix(corp)
# and store it as matrix
dtm <- as.matrix(dtm)
dim(dtm)
```

```
## [1]   182 18713
```

4. Set the variable myQuery to the following c("something","rotten", "state","denmark")

```
myQuery <- c("something","rotten", "state","denmark")
```

5. Write a function called myTextMiner()

```
myTextMiner <- function(query, corpus) {
  ##
  ## Input:
  ## - query: a string vector containing keywords
  ## - corpus: a VCorpus that needs preprocessing
  ## Output:
  ## - result.matrix: a subset of the normalized DTM with those
  ##   columns that are shared with the query, with one additional
  ##   column that contains  the  Euclidean  distance
  ##

  ## 1. pre-processing
  # convert to lower case
  corpus <- tm_map(corpus, content_transformer(tolower))
  # puntual removal
  corpus <- tm_map(corpus, removePunctuation)
  # number removal
  corpus <- tm_map(corpus, removeNumbers)

  ## 2. creat a document term matrix
  # create a document term matrix
  dtm <- DocumentTermMatrix(corpus)
  # and store it as matrix
  dtm <- as.matrix(dtm)

  ## 3. pre-process the query
  q.table <- table(myQuery)
  # get the list of word in the query
  q.names <- names(q.table)
  # get the number each word occurs
```

```
  q.values <- as.vector(q.table)

  ## 4. combine the query into the matrix
  # make a  1 * (number of columns) dimension matrix, initialize to default value 0
  n.row <- nrow(dtm)
  n.col <- ncol(dtm)
  q.row <- matrix(rep(0, n.col), 1, n.col)
  # combine the query
  dtm <- rbind(dtm, q.row)
  # assigne the words to their corresponding value
  dtm[n.row+1, q.names] <- q.values

  ## 5. normalize the documents
  # define a helper function that normalize each row
  normalizeFunc <- function(x) {
    # input:
    # x: a vector of terms for a document
    x/sum(x)
  }
  # apply the normalize function to every row (documents and query)
  norm.dtm <- apply(dtm, 1, normalizeFunc)
  norm.dtm <- t(norm.dtm)

  ## 6. compute the distance
  computeDistance <- function(x, q) {
    # input:
    # x: a vector of normalized terms for a document
    # q: a vector of normalized terms for the query
    sqrt(sum((x - q)^2))
  }
  distance.metric <- apply(norm.dtm, 1, computeDistance, q=norm.dtm[n.row+1,])

  ## 7. produce the result matrix
  result.matrix <- cbind(norm.dtm[, q.names], distance.metric)
  colnames(result.matrix) <- c(q.names, 'distanceMetric')
  rownames(result.matrix)[n.row+1] <- "query"

  return(result.matrix)

}
myCorp <- Corpus(VectorSource(shakespeare))
(resultDTM <- myTextMiner(myQuery, myCorp))
```

```
##            denmark       rotten    something        state distanceMetric
## 1     0.0000000000 0.0001426534 0.0001426534 0.0009985735      0.5049775
## 2     0.0000000000 0.0000000000 0.0005526389 0.0005526389      0.5053203
## 3     0.0000000000 0.0000000000 0.0004195511 0.0004195511      0.5057306
## 4     0.0000000000 0.0000000000 0.0003088326 0.0003088326      0.5066690
## 5     0.0000000000 0.0002340276 0.0007020828 0.0000000000      0.5065017
## 6     0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5068103
## 7     0.0000000000 0.0000000000 0.0002827255 0.0014136274      0.5052457
## 8     0.0000000000 0.0000569833 0.0002849165 0.0001139666      0.5052739
## 9     0.0000000000 0.0000000000 0.0008818342 0.0000000000      0.5075153
```

```
## 10    0.0000000000 0.0000000000 0.0007312614 0.0003656307      0.5068582
## 11    0.0000000000 0.0002433682 0.0006084205 0.0001216841      0.5062507
## 12    0.0000000000 0.0000000000 0.0003790751 0.0000000000      0.5081224
## 13    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5090631
## 14    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5076035
## 15    0.0000000000 0.0000000000 0.0009029345 0.0009029345      0.5066531
## 16    0.0000000000 0.0000000000 0.0008591065 0.0000000000      0.5074161
## 17    0.0000000000 0.0000000000 0.0000000000 0.0003165559      0.5080707
## 18    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5075533
## 19    0.0000000000 0.0001831166 0.0000000000 0.0005493499      0.5059903
## 20    0.0000000000 0.0003321156 0.0003321156 0.0009963467      0.5059103
## 21    0.0000000000 0.0000000000 0.0004532064 0.0009064129      0.5056011
## 22    0.0000000000 0.0002315351 0.0000000000 0.0016207455      0.5056875
## 23    0.0000000000 0.0000000000 0.0012355848 0.0000000000      0.5053889
## 24    0.0000000000 0.0000000000 0.0003184713 0.0003184713      0.5059174
## 25    0.0000000000 0.0000000000 0.0000000000 0.0002288591      0.5050336
## 26    0.0000000000 0.0000000000 0.0001732502 0.0001732502      0.5053141
## 27    0.0023752969 0.0001827151 0.0007308606 0.0014617212      0.5043034
## 28    0.0006558811 0.0000000000 0.0008745081 0.0004372540      0.5060836
## 29    0.0001674201 0.0000000000 0.0006696802 0.0006696802      0.5063076
## 30    0.0000000000 0.0000000000 0.0010224949 0.0000000000      0.5090341
## 31    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5100014
## 32    0.0007363770 0.0000000000 0.0000000000 0.0000000000      0.5068179
## 33    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5127250
## 34    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5080244
## 35    0.0004353505 0.0002176752 0.0002176752 0.0002176752      0.5061812
## 36    0.0000000000 0.0002512563 0.0000000000 0.0000000000      0.5072251
## 37    0.0000000000 0.0000000000 0.0000000000 0.0003459011      0.5061477
## 38    0.0000000000 0.0000000000 0.0000000000 0.0012039490      0.5061153
## 39    0.0000000000 0.0000000000 0.0000000000 0.0008179959      0.5072419
## 40    0.0000000000 0.0000000000 0.0000000000 0.0002866972      0.5069107
## 41    0.0000000000 0.0000000000 0.0002546473 0.0002546473      0.5074111
## 42    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5065633
## 43    0.0000000000 0.0000000000 0.0000000000 0.0007017544      0.5079009
## 44    0.0000000000 0.0003490401 0.0003490401 0.0005235602      0.5066601
## 45    0.0000000000 0.0002768549 0.0002768549 0.0011074197      0.5064198
## 46    0.0000000000 0.0000000000 0.0000000000 0.0007504690      0.5081604
## 47    0.0000000000 0.0000000000 0.0000000000 0.0002702703      0.5075689
## 48    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5141440
## 49    0.0000000000 0.0002251238 0.0000000000 0.0000000000      0.5077872
## 50    0.0000000000 0.0000000000 0.0001488317 0.0000000000      0.5072851
## 51    0.0000000000 0.0000000000 0.0002805836 0.0002805836      0.5068333
## 52    0.0000000000 0.0000000000 0.0002405581 0.0002405581      0.5056761
## 53    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5064266
## 54    0.0000000000 0.0000000000 0.0000000000 0.0006082725      0.5063110
## 55    0.0000000000 0.0000000000 0.0000000000 0.0002655337      0.5062399
## 56    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5062840
## 57    0.0000000000 0.0000000000 0.0000000000 0.0008898776      0.5080025
## 58    0.0000000000 0.0000000000 0.0000000000 0.0011373330      0.5065319
## 59    0.0000000000 0.0000000000 0.0000000000 0.0007252947      0.5056253
## 60    0.0000000000 0.0000000000 0.0000000000 0.0003712642      0.5072581
## 61    0.0000000000 0.0000000000 0.0000000000 0.0004258944      0.5064104
## 62    0.0000000000 0.0002424830 0.0000000000 0.0007274491      0.5073754
## 63    0.0000000000 0.0000000000 0.0002029633 0.0002029633      0.5060812
```

```
## 64    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5106375
## 65    0.0000000000 0.0000000000 0.0000000000 0.0009146341      0.5062740
## 66    0.0000000000 0.0000000000 0.0000000000 0.0009768010      0.5069193
## 67    0.0000000000 0.0000000000 0.0000000000 0.0002675943      0.5077376
## 68    0.0000000000 0.0000000000 0.0004554771 0.0015941699      0.5052211
## 69    0.0000000000 0.0000000000 0.0004235493 0.0012706480      0.5057658
## 70    0.0000000000 0.0000000000 0.0004566210 0.0013698630      0.5058809
## 71    0.0000000000 0.0000000000 0.0008499788 0.0016999575      0.5065482
## 72    0.0000000000 0.0000000000 0.0000000000 0.0011947431      0.5059551
## 73    0.0000000000 0.0000000000 0.0005437738 0.0000000000      0.5065905
## 74    0.0000000000 0.0002511301 0.0002511301 0.0002511301      0.5062299
## 75    0.0000000000 0.0000000000 0.0000000000 0.0002452182      0.5059313
## 76    0.0000000000 0.0000000000 0.0000000000 0.0008010681      0.5062750
## 77    0.0000000000 0.0000000000 0.0000000000 0.0005747126      0.5081024
## 78    0.0000000000 0.0000000000 0.0000000000 0.0005417118      0.5074773
## 79    0.0000000000 0.0000000000 0.0000000000 0.0002891009      0.5076004
## 80    0.0000000000 0.0000000000 0.0002449180 0.0004898359      0.5075079
## 81    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5077887
## 82    0.0000000000 0.0000000000 0.0004053506 0.0000000000      0.5068487
## 83    0.0000000000 0.0000000000 0.0003374388 0.0003374388      0.5053724
## 84    0.0000000000 0.0000000000 0.0002371354 0.0007114062      0.5050419
## 85    0.0000000000 0.0000000000 0.0004791567 0.0002395783      0.5059768
## 86    0.0000000000 0.0000000000 0.0006402049 0.0004268032      0.5049972
## 87    0.0000000000 0.0003353454 0.0003353454 0.0010060362      0.5056408
## 88    0.0000000000 0.0000000000 0.0003389831 0.0000000000      0.5070081
## 89    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5069862
## 90    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5088069
## 91    0.0000000000 0.0000000000 0.0004128819 0.0004128819      0.5063603
## 92    0.0000000000 0.0001397038 0.0000000000 0.0004191115      0.5062423
## 93    0.0000000000 0.0000000000 0.0000000000 0.0009104704      0.5070592
## 94    0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5077373
## 95    0.0000000000 0.0000000000 0.0003109453 0.0006218905      0.5072245
## 96    0.0000000000 0.0000000000 0.0006114338 0.0006114338      0.5056459
## 97    0.0000000000 0.0000000000 0.0003840246 0.0000000000      0.5061971
## 98    0.0000000000 0.0000000000 0.0007572889 0.0007572889      0.5063086
## 99    0.0000000000 0.0000000000 0.0002190101 0.0004380201      0.5063972
## 100   0.0000000000 0.0001415428 0.0002830856 0.0001415428      0.5068682
## 101   0.0000000000 0.0000000000 0.0000000000 0.0005685048      0.5066806
## 102   0.0000000000 0.0003142678 0.0006285355 0.0000000000      0.5066848
## 103   0.0000000000 0.0000000000 0.0009140768 0.0000000000      0.5058342
## 104   0.0000000000 0.0000000000 0.0005254309 0.0012610340      0.5063978
## 105   0.0000000000 0.0000000000 0.0002800336 0.0000000000      0.5068419
## 106   0.0000000000 0.0000000000 0.0002446782 0.0000000000      0.5065841
## 107   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5100489
## 108   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5102547
## 109   0.0000000000 0.0003186743 0.0003186743 0.0003186743      0.5068405
## 110   0.0000000000 0.0000000000 0.0005363368 0.0000000000      0.5067950
## 111   0.0000000000 0.0000000000 0.0000000000 0.0013410818      0.5067479
## 112   0.0000000000 0.0000000000 0.0004555809 0.0000000000      0.5067509
## 113   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5082088
## 114   0.0000000000 0.0000000000 0.0002316960 0.0002316960      0.5058569
## 115   0.0000000000 0.0000000000 0.0005847953 0.0000000000      0.5072633
## 116   0.0000000000 0.0000000000 0.0003827019 0.0000000000      0.5073869
## 117   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5071410
```

```
## 118   0.0000000000 0.0000000000 0.0002242152 0.0002242152      0.5067496
## 119   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5068110
## 120   0.0000000000 0.0003824092 0.0000000000 0.0000000000      0.5064665
## 121   0.0000000000 0.0000000000 0.0000000000 0.0005064573      0.5061965
## 122   0.0000000000 0.0000000000 0.0006699419 0.0020098258      0.5055428
## 123   0.0000000000 0.0000000000 0.0002126755 0.0000000000      0.5071169
## 124   0.0000000000 0.0000000000 0.0012383901 0.0004127967      0.5056226
## 125   0.0000000000 0.0000000000 0.0002383222 0.0002383222      0.5067995
## 126   0.0000000000 0.0000000000 0.0000000000 0.0011782032      0.5061269
## 127   0.0000000000 0.0000000000 0.0000000000 0.0002365744      0.5060408
## 128   0.0000000000 0.0002350176 0.0004700353 0.0002350176      0.5054066
## 129   0.0000000000 0.0000000000 0.0000000000 0.0015263292      0.5059735
## 130   0.0000000000 0.0000000000 0.0000000000 0.0022925264      0.5055600
## 131   0.0000000000 0.0000000000 0.0000000000 0.0010422095      0.5058308
## 132   0.0000000000 0.0000000000 0.0001412030 0.0001412030      0.5061782
## 133   0.0000000000 0.0000000000 0.0000000000 0.0007062147      0.5062963
## 134   0.0000000000 0.0001811594 0.0001811594 0.0009057971      0.5066578
## 135   0.0000000000 0.0001710864 0.0003421728 0.0003421728      0.5059658
## 136   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5075321
## 137   0.0000000000 0.0000000000 0.0000000000 0.0002103934      0.5054314
## 138   0.0000000000 0.0000000000 0.0002230152 0.0000000000      0.5057395
## 139   0.0000000000 0.0000000000 0.0001874766 0.0003749531      0.5049606
## 140   0.0000000000 0.0000000000 0.0000000000 0.0007390983      0.5056457
## 141   0.0000000000 0.0003473428 0.0006946857 0.0000000000      0.5062258
## 142   0.0000000000 0.0002908668 0.0000000000 0.0002908668      0.5071507
## 143   0.0000000000 0.0000000000 0.0003615329 0.0000000000      0.5074905
## 144   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5065727
## 145   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5074234
## 146   0.0000000000 0.0000000000 0.0004372540 0.0004372540      0.5067101
## 147   0.0000000000 0.0002577320 0.0005154639 0.0007731959      0.5051617
## 148   0.0000000000 0.0003036745 0.0000000000 0.0000000000      0.5053215
## 149   0.0000000000 0.0000000000 0.0008206812 0.0004103406      0.5052124
## 150   0.0000000000 0.0000000000 0.0005599104 0.0005599104      0.5054149
## 151   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5066767
## 152   0.0000000000 0.0000000000 0.0000000000 0.0011123471      0.5051607
## 153   0.0000000000 0.0000000000 0.0011331445 0.0011331445      0.5054349
## 154   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5059854
## 155   0.0000000000 0.0002294631 0.0002294631 0.0004589261      0.5061482
## 156   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5066693
## 157   0.0000000000 0.0000000000 0.0000000000 0.0002951594      0.5088844
## 158   0.0000000000 0.0000000000 0.0002750275 0.0000000000      0.5062302
## 159   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5062860
## 160   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5077266
## 161   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5073461
## 162   0.0000000000 0.0000000000 0.0000000000 0.0001969279      0.5064527
## 163   0.0000000000 0.0000000000 0.0000000000 0.0005351887      0.5063412
## 164   0.0000000000 0.0000000000 0.0004811162 0.0007216743      0.5050461
## 165   0.0000000000 0.0000000000 0.0002315351 0.0006946052      0.5056966
## 166   0.0000000000 0.0002290951 0.0004581901 0.0000000000      0.5056165
## 167   0.0000000000 0.0000000000 0.0000000000 0.0005259006      0.5062476
## 168   0.0000000000 0.0000000000 0.0005292405 0.0013231013      0.5052524
## 169   0.0000000000 0.0000000000 0.0008760403 0.0002190101      0.5069349
## 170   0.0000000000 0.0000000000 0.0006720430 0.0000000000      0.5079418
## 171   0.0000000000 0.0000000000 0.0000000000 0.0003819710      0.5070097
```

```
## 172   0.0000000000 0.0000000000 0.0004048583 0.0000000000      0.5069650
## 173   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5067293
## 174   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5066309
## 175   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5066192
## 176   0.0000000000 0.0000000000 0.0000000000 0.0005844535      0.5061684
## 177   0.0000000000 0.0000000000 0.0006042296 0.0000000000      0.5052257
## 178   0.0000000000 0.0003105590 0.0009316770 0.0000000000      0.5056730
## 179   0.0000000000 0.0003736921 0.0003736921 0.0000000000      0.5064356
## 180   0.0000000000 0.0000000000 0.0009965831 0.0002847380      0.5054653
## 181   0.0000000000 0.0000000000 0.0010196278 0.0002549070      0.5059261
## 182   0.0000000000 0.0000000000 0.0000000000 0.0000000000      0.5062521
## query 0.2500000000 0.2500000000 0.2500000000 0.2500000000      0.0000000
```

```
# write the DTM from the function to csv file
write.csv(resultDTM, file="Xu_Hong_DTM.csv")
```