R을 이용한 문서 분석

김청택
서울대학교 심리학과,인지과학협동과정, 데이터사이언스 대학원

2020년 4월 10일

차 례

제 1 절 Library 부르기

R은 기본 프로그램과 다양한 library로 구성되어 있다. 세부적인 기능을 하는 프로그램은 library의 단위로 구성되어 있다. 예컨대, 그래프를 다루는 프로그램들은 ggplot2라는 library에 저장되어 있다. ggplot2에 있는 함수 등을 사용하기 위해서는 ggplot2을 현재의 작업 기억 속으로 불러와야 한다. 이 명령어가 library 혹은 require이다. 아래의 명령어는 문서처리에 사용되는 library를 불러오기 위한 것이다. 아래에는 만약 해당 library가 컴퓨터에 설치되어 있지 않으면 인터넷에서 download하여 설치하게 된다.

```
packages = c("tm", "wordcloud","ggplot2","stringr","readtext","RmecabKo")
for(i in packages){
  if( ! require( i , character.only = TRUE ) ){install.packages( i , dependencies = TRUE )}
}
```

```
## Loading required package: tm
## Loading required package: NLP
## Loading required package: wordcloud
## Loading required package: RColorBrewer
## Loading required package: ggplot2
##
## Attaching package: 'ggplot2'
## The following object is masked from 'package:NLP':
##
##     annotate
## Loading required package: stringr
## Loading required package: readtext
## Loading required package: RmecabKo
##
## Attaching package: 'RmecabKo'
## The following object is masked from 'package:NLP':
##
##     words
```

모두 변수를 다 지우고, pal라는 변수를 할당하였다. pal변수는 그래프에서 색깔을 지정해 주는 변수이다.

```
rm(list=ls())
pdf.options(family="Korea1deb")
pal <- brewer.pal(9,"Set1")
```

제 2 절 문서 읽기와 전처리

2.1 문서읽기

문서의 문석을 위해서는 자료를 읽어야 된다. 예시를 위해 교육평가학회지 2018년도에 게재된 논문이 pdf file로 data_edueval이라는 폴더에 저장되어 있다. readtext라는 명령어로 이 폴더에 있는 모든 파일을 읽을 수 있다. data_edueval에 12개의 논문파일을 읽어서 text_r에 저장한다. 12개의 행에 12개의 문서가 저장되어 있다.

자료를 읽는 방법이 아래에 제시되어 있다.

```
folder="./data_edueval"
text_r <- readtext(paste0(folder, "/*.pdf"))
```

2.2 전처리

이 문서중 내용어(명사형)만 추출하는 code가 아래에 지시되어 있다. nouns라는 명령어를 사용하면 명사형이 추출되는데, 이를 txt에 저장한다. 이때 lapply를 사용하여 모든 list의 element에 nouns라는 명령어를 적용하였다.

```
text_r <- readtext(paste0(folder, "/*.pdf"))
txt_o=text_r$text
txt_n=lapply(txt_o,nouns)
txt<-txt_n
```

gsub 함수를 이용하여 하나의 문자열을 다른 문자열고 변경할 수 있다. 예컨대 gsub("http", " ", txt)는 txt에 있는 http라는 스트링을 공백(" ")으로 바꾸라는 것이다. 이 함수를 이용하여 http를 문서에서 제거할 수 있다. 이 함수를 이용하여 http를 문서에서 제거할 수 있다. 또한 눈에 보이는 않은 문자도 파일에 존재하는 경우가 있는데, 대표적인 것이 다음 줄로 넘기는 문자(\n)와 엔터 문자(\r)이다.

정규식(regular expression)을 이용하면 문자열을 융통성있게 정의할 수 있다. 예컨대 http://[[:graph:]]* 라는 정규식은 "http://" 뒤에 어떤 (보이는) 문자 있는 문자열을 의미한다. "http://www.snu.ac.kr"와 "http://www"와 같은 문자열이 이에 해당한다. 아래의 첫번째 두번째 명령어는 으로시작되는 문자열, http://로 시작되는 문자열을 공백으로 바꾸는 것이다.

```
txt <- gsub("@[[:graph:]]*", "",txt)
txt <- gsub("http://[[:graph:]]*", "",txt)
txt <- gsub("[^[:graph:]]", " ",txt)
txt <- gsub("[[:punct:]]","",txt)
txt <- gsub("\n", " ", txt)
txt <- gsub("#", " ", txt)
txt <- gsub("\r", " ", txt)
txt <- gsub("RT", " ", txt)
txt <- gsub("http", " ", txt)
txt <- gsub("  ", " ",txt)
```

제 3 절 Corpus와 Document-Term matrix의 구성

3.1 Corpus의 구성

전처리를 한 다음 문서는 코퍼스의 형식으로 저장되어야 한다. Corpus라는 명령어를 사용하여 코퍼스로 변환할 수 있다.

```
corpus <- Corpus(VectorSource(txt))
```

3.2 Document Term Matrix의 구성

DocumentTermMatrix나 TermDocumentMatrix라는 함수를 사용하여 corpus를 document term matrix로 변형할 수 있다. 아래에서는 tdm, tdm2 두개의 matrix가 구성되었다. control에서 document term amtrix의 특징을 정의할 수 있다. 예컨대 "weighting = function(x) weightTfIdf(x, TRUE))"를 포함하면 TfIdf(Term Frequency - Inverse Document Frequency)에 의하여 빈도에 가중치를 두게 된다. 아래의 예에서 control2에서는 이를 포함하고 있지 않다. 또한 숫자와 구두점을 제거할 것인지에 대하여 지정할 수 있다. wordLength(4,20)는 4자 미만 문자열과 20자보다 많은 문자열은 제거한다는 의미이다.

```
uniTokenizer <- function(x) unlist(strsplit(as.character(x), "[[:space:]]+"))
control = list(tokenize = uniTokenizer,
               removeNumbers = TRUE,
```

```
                wordLengths=c(4,20),
                removePunctuation = T,
                stopwords =TRUE,
                weighting = function(x) weightTfIdf(x, TRUE))
tdm <- DocumentTermMatrix(corpus, control=control)

control2 = list(tokenize = words,
                removeNumbers = TRUE,
                wordLengths=c(4,20),
                removePunctuation = T,
                stopwords = TRUE)
tdm2 <- DocumentTermMatrix(corpus, control=control2)
```

제 4 절 빈도 분석

이제 자료가 준비되었으니 자료에 대한 분석을 진행하면 된다. 먼저 기초적인 분석을 하는 과정을 살펴보자. 먼저 빈번히 등장하는 단어를 추출하고, 이를 막대 그래프와 wordcloud 도표를 표현하는 명령문들이 아래에 제시되어 있다.

```
#findFreqTerms(tdm2)
TermFreq <-colSums(as.matrix(tdm2))
summary(TermFreq)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    1.00    3.00   19.33    9.00 1408.00

TermFreq2 <- subset(TermFreq, TermFreq>100)
TermFreq2

##     결과     경우     과정     과제     과학     관련     교과     교사
##      416      315      352      207      426      131      133      653
##     교수     교실     교육     구성     구체     국가     기능     기반
##      141      220      644      145      107      104      155      142
##     논문     능력     다음     때문     모형     목표     문항     반응
##      103      266      105      128      703      217      932      189
```

```
##       방법       분석       비교       사용       설정       성취       수업       수준
##        394        500        158        260        115        439        690        740
##       수행       실제       연계       연구       영역       영향       오류       유의
##        220        146        169       1124        107        340        126        138
##       유형       의미       이론       이해       인식       인지       자료       적용
##        211        201        101        117        399        143        325        203
##       점수       정도       정보       정의       제공       제시       차이       탐구
##        189        111        175        186        194        132        206        314
##       특성       평가     피드백       필요       학교       학생       학습       학업
##        200       1408       1020        145        257        796        648        103
##       한국       형성       활동       활용       효과       검사       고려       기초
##        105        173        290        220        558        236        116        123
##       다층       대상       대칭       분류       분포       비율       요인       적합
##        207        112        106        144        156        145        123        150
##       조건       지수       집단       차원       추정       크기       태도       통계
##        204        125        443        200        276        253        162        145
##       평균       표본       거리       척도       소속       잠재       모수       모의
##        177        107        110        298        219        279        344        153
##     피험자       실수       다중
##        118        219        198

gframe<-data.frame(term = names(TermFreq2),freq = TermFreq2)
#ggplot(data=gframe)+aes(x=term,y=freq)+geom_bar(stat="identity")+coord_flip()

wordcloud(names(TermFreq2), TermFreq2 , col=pal)
```

위의 분석은 가중치가 주어지지 않은 document term matrix에 대한 분석이었고, 다음은 가중치가 주어진 matrix에 대한 분석이다.

```
#findFreqTerms(tdm)
TermFreq <-colSums(as.matrix(tdm))
summary(TermFreq)

##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## 0.000000 0.001157 0.002244 0.005275 0.004807 0.203803

TermFreq2<-NA
TermFreq2 <- subset(TermFreq, TermFreq>0.02)
#TermFreq2
gframe<-data.frame(term = names(TermFreq2),freq = TermFreq2)
wordcloud(names(TermFreq2), TermFreq2 , col=pal)
```


```
#wordcloud(names(TermFreq2), TermFreq2 ,  scale=c(4,0.01), col=pal)
```

제 5 절　군집분석

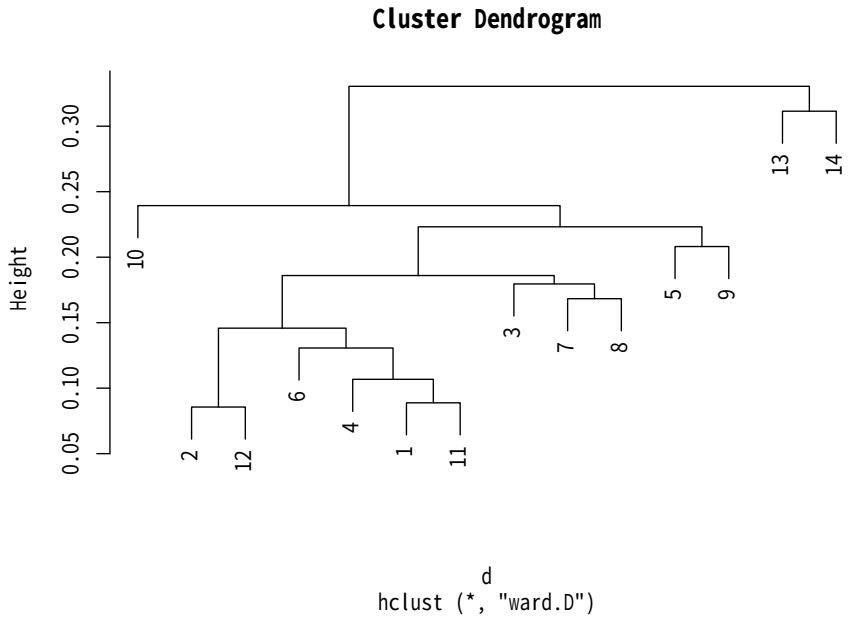
군집분석은 위계적 군집분석과 kmeans 군집분석이 있다. 전자는 처음에 단위의 수만큼 군집을 만들고 제일 유사한 두 군집을 새로운 군집을 만드는 과정을 계속하여 결국에는 하나의 군집으로 만드는 기법이다. 후자는 미리 몇개의 군집으로 분리할 것인지를 미리 정한 다음 집단내 거리를 가장 가깝세 하고 집단간 거리를 가장 멀게 하는 방식으로 군집화하는 기법이다.

위계적 군집분석은 hclust 함수를 이용하면 된다. 이 함수에 대한 입력은 거리 행렬이기 때문에 dist라는 함수를 이용하여 거리 행렬을 만들었다.다양한 거리에 대한 정의가 있고 군집들 간의 거리를 정하는 다양한 방법이 있는데, 여기에서는 개체들간의 거리는 유클리디안 거리로, 군집간의 거리는 ward 방법을 이용하고 있다.

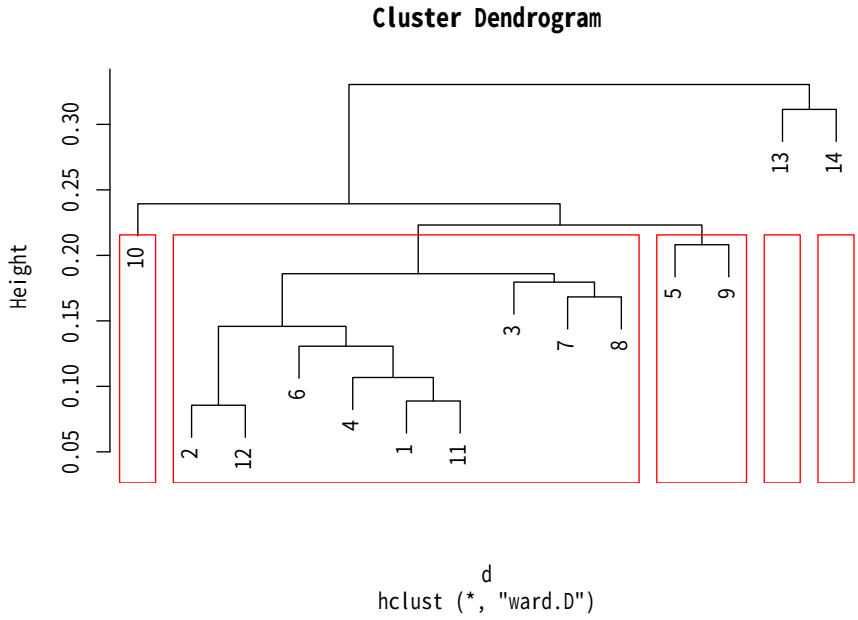
plot을 이용하여 위계적 군집분석의 수형도를 그린 다음, rect.hist를 사용하여 군집을 설정할 수 있다.

```r
td <- removeSparseTerms(tdm,0.99)
mdata<-as.matrix(td)
d <- dist(mdata, method = "euclidean") # distance matrix
fit <- hclust(d, method="ward.D")
#fit <- hclust(d, method="complete")

plot(fit) # display dendogram
```




```r
groups <- cutree(fit, k=5)
rect.hclust(fit, k=5, border="red")
```


Cluster Dendrogram
Height
0.05
0.10
0.15
0.20
0.25
0.30
10
2
12
6
4
1
11
3
7
8
5
9
13
14
d
hclust (*, "ward.D")