다변량분석 과제 1

2015170378

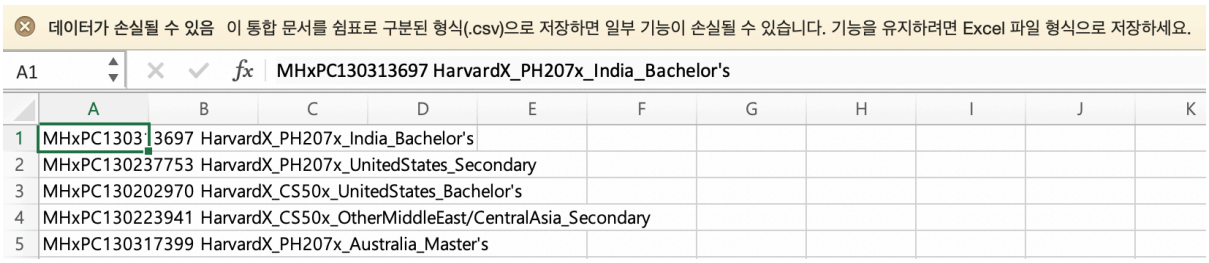
정은영

[Step1] 데이터 변환

각 단계는 주석처리하여 표시하였다.

```
# Assignment1 - 2015170378 Jung Eun young
# Association Rules ------------------------------------------------------
# arules and arulesViz packages install
install.packages("arules")
install.packages("arulesViz")
install.packages("wordcloud")

library(arules)
library(arulesViz)
library(wordcloud)

# Part 1: Transform a data file into transaction format
mooc_dataset <- read.csv("big_student_clear_third_version.csv", header=TRUE)

#[Step1]
#<Q1>
#Phase1
Institute <- mooc_dataset$institute
Course <-mooc_dataset$course_id
Region <- mooc_dataset$final_cc_cname_DI
Degree <- mooc_dataset$LoE_DI

##Phase2
Region <- factor(gsub(' ','',Region)) #공백처리
Region <- factor(gsub(',','',Region)) #'OtherNorth&CentralAmer.,Caribbean'의 ','가 csv변환시 문제가 되어 제거
Degree <- factor(gsub(' ','',Degree)) #item생성 시 less than Secondary가 less로 잘리는 문제 해결

###Phase3
RawTransactions <- paste(Institute,Course,Region,Degree,sep="_")

####Phase4
MOOC_transactions <- paste(mooc_dataset$userid_DI,RawTransactions,sep =" ")

#####Phase5
write.table(MOOC_transactions,"MOOC_User_Course.csv", sep=",", quote = FALSE, row.names = FALSE, col.names=FALSE)
```

csv 저장 파일



[Step 2] 데이터 불러오기 및 기초 통계량 확인

[Q2-1]

```
#[Step2]
#<Q2-1>
tmp_single <- read.transactions("MOOC_User_Course.csv", format = "single",
                                cols = c(1,2), rm.duplicates=TRUE)
inspect(tmp_single[1:10])
summary(tmp_single)
```

----결과---

```
> summary(tmp_single)
transactions as itemMatrix in sparse format with
 335650 rows (elements/itemsets/transactions) and
 1405 columns (items) and a density of 0.000877119

most frequent items:
    MITx_6.00x_UnitedStates_Bachelor's      MITx_6.00x_UnitedStates_Secondary            MITx_6.00x_India_Bachelor's
                                 14192                                   8841                                   7813
         MITx_6.002x_India_Bachelor's HarvardX_CS50x_UnitedStates_Bachelor's                                  (Other)
                                  7633                                   7410                                 367750

element (itemset/transaction) length distribution:
sizes
     1      2      3      4      5      6      7      8      9     10     11     12     13
278440  43061   9997   2812    799    293    109     44     37     22     21      9      6

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   1.000   1.000   1.232   1.000  13.000

includes extended item information - examples:
                            labels
1 HarvardX_CB22x_Australia_Bachelor's
2   HarvardX_CB22x_Australia_Master's
3  HarvardX_CB22x_Australia_Secondary

includes extended transaction information - examples:
   transactionID
1 MHxPC130000002
2 MHxPC130000004
3 MHxPC130000006
```

전체 335650 개 row 와 1405 개의 item 이 있음을 확인할 수 있다.

(MIT 대학 제공, 강좌코드 6.00x 를 듣는, 접속국가 미국이며, 학사학위)인 MOOC 강좌 수강생이 14192 명으로 가장 많다는 것을 확인할 수 있다. frequency 가 많은 상위 5 개 항목(item)을 볼 수 있다.

[Q2-2]

```
#<Q2-2>
itemName <- itemLabels(tmp_single)
itemCount <- itemFrequency(tmp_single)*nrow(tmp_single)

col <- brewer.pal(7,"Dark2")
wordcloud(words = itemName, freq = itemCount,min.freq = 2000, scale = c(1, 0.2),
col = col , random.order = FALSE,family = "Rockwell")
```

---결과---

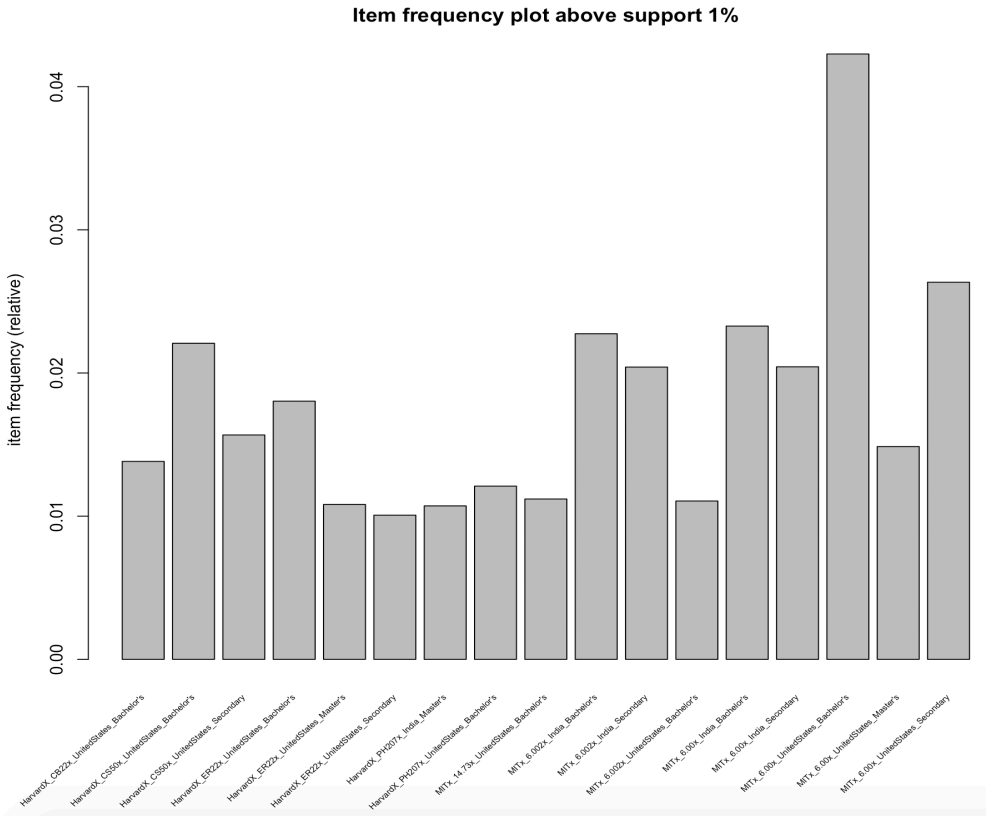
min.freq=2000 으로 설정하였다. 워드클라우드를 통해 Summary 에서 확인했던 (MIT 대학 제공, 강좌코드 6.00x 를 듣는, 접속국가 미국이며, 학사학위)가 가장 크게 나타났음을 확인할 수 있다. 빈도수가 높은 상위 5 개 항목도 쉽게 확인할 수 있다.

또한 MIT 에서 제공되는 강좌 중 6.00x 의 코드를 가지는 강좌가 인기있음을 알 수 있다.

[Q2-3]

```
51  #<Q2-3>
52  itemFrequencyPlot(tmp_single, support = 0.01, cex.names=0.5, main ="Item frequency
    plot above support 1%")
```

---결과---



상위 5 개의 Item 에 대해 접속국가는 음영 친 부분과 같다 (미국, 인도)

1. MITx_6.00x_UnitedStates_Bachelor's
2. MITx_6.00x_UnitedStates_Secondary
3. MITx_6.00x_India_Bachelor's
4. MITx_6.002x_India_Bachelor's
5. HarvardX_CS50x_UnitedStates_Bachelor's

[Step3] 데이터 불러오기 및 기초 통계량 확인

[Q3-1]

```
#[Step3]
#<Q3-1>
nMat <- matrix(c(1:16),nrow=4,ncol=4,dimnames=list(c("support=0.001", "        0.0015","
       0.002","        0.0025"),c("confidence=0.05", "0.06", "0.07","0.08")))
for (i in 1:4){
  for(j in 1:4){
    nMat[i,j]=length(apriori(tmp_single, parameter=list(support=as.double(0.0005*(i+1)),
confidence=as.double(0.01*(j+4)))))
  }
}
as.table(nMat)
```

---결과----

Assignment1_2015170378_정은영.R* × nMat × 02_Ass

Filter

	confidence=0.05	0.06	0.07	0.08
support=0.001	51	49	45	40
0.0015	29	28	27	25
0.002	20	20	19	18
0.0025	14	14	13	13

support 는 0.0005 단위로, confidence 는 0.01 단위로 for 문을 활용하여 총 16 가지의 규칙 개수를 확인하였다. support 는 작아질수록, confidence 는 커질수록 해당하는 규칙 개수가 작아지는 경향을 보인다

[Q3-2]

```
#<Q3-2>
rules <- apriori(tmp_single, parameter=list(support=0.001, confidence=0.05))
inspect(sort(rules, by="support")[1])
inspect(sort(rules, by="confidence")[1])
inspect(sort(rules, by="lift")[1])
df <- as(rules,"data.frame")
df[,6]<-df[,2]*df[,3]*df[,4]
df[order(-df[,6])[1:3],]
```

---결과---

```
> inspect(sort(rules, by="support")[1])
    lhs                                       rhs                                      support     confidence lift     count
[1] {HarvardX_CS50x_UnitedStates_Bachelor's} => {MITx_6.00x_UnitedStates_Bachelor's} 0.003643676 0.1650472  3.903474 1223
> inspect(sort(rules, by="confidence")[1])
    lhs                              rhs                              support     confidence lift     count
[1] {MITx_8.02x_India_Secondary} => {MITx_6.002x_India_Secondary} 0.002800536 0.388109   19.01179 940
> inspect(sort(rules, by="lift")[1])
    lhs                                    rhs                                    support    confidence lift     count
[1] {MITx_8.02x_UnitedStates_Bachelor's} => {MITx_6.002x_UnitedStates_Bachelor's} 0.00139133 0.2162037  19.54978 467
> df <- as(rules,"data.frame")
> df[,6]<-df[,2]*df[,3]*df[,4]
> df[order(-df[,6])[1:3],]
                                                          rules     support confidence     lift count         V6
23    {MITx_8.02x_India_Secondary} => {MITx_6.002x_India_Secondary} 0.002800536  0.3881090 19.01179   940 0.02066417
5   {MITx_8.02x_India_Bachelor's} => {MITx_6.002x_India_Bachelor's} 0.002496648  0.3856420 16.95804   838 0.01632741
25 {HarvardX_CS50x_India_Secondary} => {MITx_6.00x_India_Secondary} 0.002681365  0.2939255 14.38555   900 0.01133756
```

✓ Support 가 가장 높은 규칙은

{HarvardX_CS50x_UnitedStates_Bachelor's} => {MITx_6.00x_UnitedStates_Bachelor's}

✓ Confidence 가 가장 높은 규칙은

{MITx_8.02x_India_Secondary} => {MITx_6.002x_India_Secondary}

✓ Lift 가 가장 높은 규칙은

{MITx_8.02x_UnitedStates_Bachelor's} => {MITx_6.002x_UnitedStates_Bachelor's}

✓ 하나의 규칙에 대한 효용성 지표를 Support × Confidence × Lift 로 정의한다면 효용성이 가 장 높은 규칙 1 위~3 위는 (위의 결과에서 V6=Support × Confidence × Lift)

1) {MITx_8.02x_India_Secondary} => {MITx_6.002x_India_Secondary}

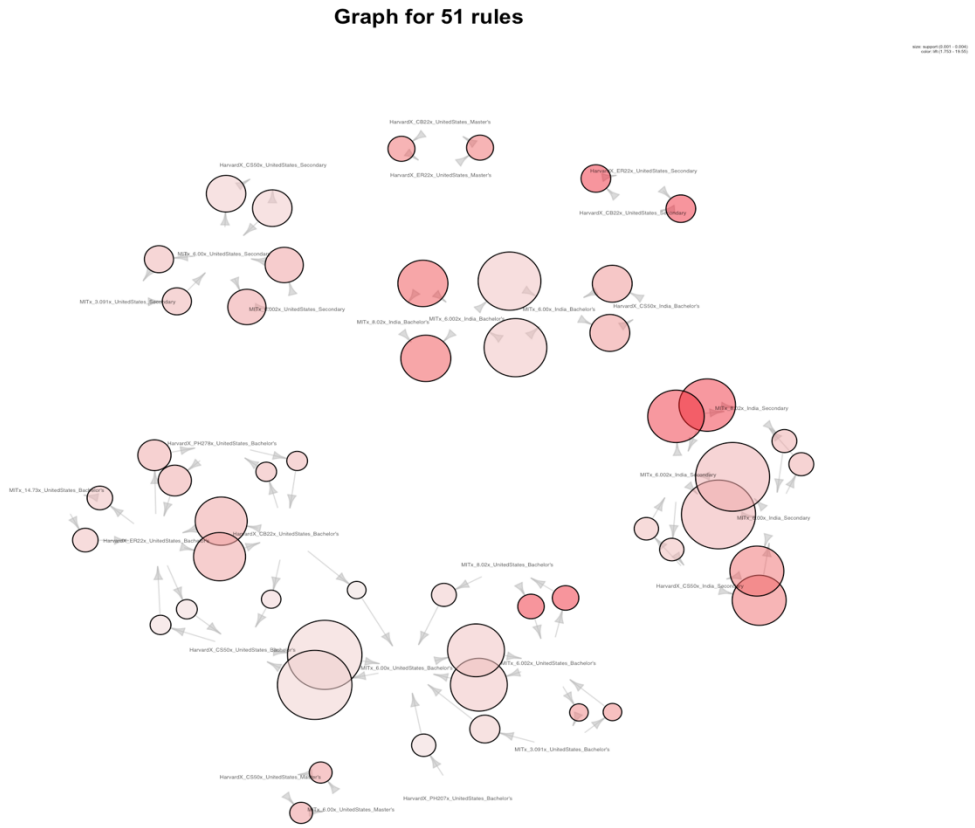
2) {MITx_8.02x_India_Bachelor's} => {MITx_6.002x_India_Bachelor's}

3) {HarvardX_CS50x_India_Secondary} => {MITx_6.00x_India_Secondary}

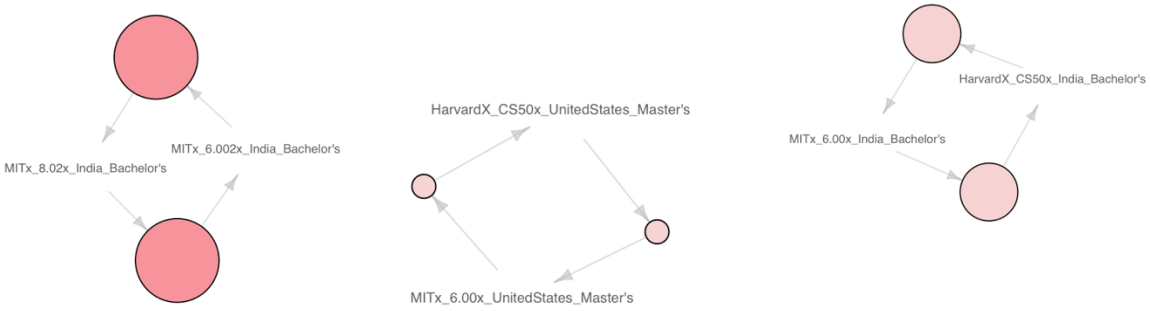
✓ 생성된 규칙을 plot( )함수의 "graph" method 를 이용하여 도시할 경우 두 아이템이 서로 조건절/결과 절을 달리해서 생성되는 경우가 존재함을 확인할 수 있다(X→Y 규칙과 Y→X 규칙이 함께 존재한 다는 뜻). 이 중에서 세 가지 규칙을 선택하여 각 규칙들에 대한 Support/Confidence/Lift 값을 확인 해보고 조건절과 결과절의 위치에 따라서 어떤 지표 값들이 차이가 나는지와 왜 그러한 상황이 발생 하는지 서술하시오.

```
73  # Plot the rules
74  plot(rules, method="graph",cex=0.3)
75  plot(rules[1:10], method="graph",cex=0.3)
76  inspect(rules[1:10])
```

---결과---



위의 그래프의 전체 rules 중 10 개만 뽑아 그래프를 그려보고 이 중 화살표가 양방향인 3 가지를 뽑아보았다. 또한 10 개에 대한 summary 를 통해 그 중 3 개의 항목을 찾았다.




```
> inspect(rules[1:10])
     lhs                                    rhs                                    support     confidence lift      count
[1]  {HarvardX_CS50x_UnitedStates_Master's} => {MITx_6.00x_UnitedStates_Master's}     0.001218531 0.16985050 11.429495 409
[2]  {MITx_6.00x_UnitedStates_Master's}     => {HarvardX_CS50x_UnitedStates_Master's} 0.001218531 0.08199679 11.429495 409
[3]  {HarvardX_CS50x_India_Bachelor's}      => {MITx_6.00x_India_Bachelor's}          0.002016982 0.26918489 11.564304 677
[4]  {MITx_6.00x_India_Bachelor's}          => {HarvardX_CS50x_India_Bachelor's}      0.002016982 0.08665045 11.564304 677
[5]  {MITx_8.02x_India_Bachelor's}          => {MITx_6.002x_India_Bachelor's}         0.002496648 0.38564197 16.958041 838
[6]  {MITx_6.002x_India_Bachelor's}         => {MITx_8.02x_India_Bachelor's}          0.002496648 0.10978645 16.958041 838
```

위의 3 가지 rule 들에서 support 와 lift 는 각각의 경우 방향과 상관없이 같지만 confidence 는 다르다.

```
> inspect(rules[1:10])
     lhs                                          rhs                                          support     confidence lift      count
[1]  {HarvardX_CS50x_UnitedStates_Master's} => {MITx_6.00x_UnitedStates_Master's}     0.001218531 0.16985050 11.429495 409
[2]  {MITx_6.00x_UnitedStates_Master's}     => {HarvardX_CS50x_UnitedStates_Master's} 0.001218531 0.08199679 11.429495 409
[3]  {HarvardX_CS50x_India_Bachelor's}      => {MITx_6.00x_India_Bachelor's}          0.002016982 0.26918489 11.564304 677
[4]  {MITx_6.00x_India_Bachelor's}          => {HarvardX_CS50x_India_Bachelor's}      0.002016982 0.08665045 11.564304 677
[5]  {MITx_8.02x_India_Bachelor's}          => {MITx_6.002x_India_Bachelor's}         0.002496648 0.38564197 16.958041 838
[6]  {MITx_6.002x_India_Bachelor's}         => {MITx_8.02x_India_Bachelor's}          0.002496648 0.10978645 16.958041 838
```

Performance Measures for the rule A → B

- Support

$$\text{support}(A) = P(A)$$

✓Used to find the frequent item sets

- Confidence

$$confidence(A \rightarrow B) = \frac{P(A,B)}{P(A)}$$

✓Used to generate meaningful rules

- Lift

$$lift(A \rightarrow B) = \frac{P(A,B)}{P(A) \cdot P(B)}$$

✓Used to determine the usefulness of generated rules

수업자료인 이 식을 바탕으로 첫번째 경우인 {HarvardX_CS50x_UnitedStates_Master's} => {MITx_6.00x_UnitedStates_Master's}를 예를 들어 설명하겠다.

화살표는 인과관계가 아니며 support 는 수업자료와 다르게 R 에서는 lhs,rhs 동시 포함 확률으로 계산된다. 따라서 409(lhs,rhs 동시포함건수)/335650(전체거래수)=0.001218531 로 [1],[2]가 같다. confidence, lift 는 위 식과 같이 계산된다. [1]의 경우 {HarvardX_CS50x_UnitedStates_Master's}가 나온 건수는 2408 건, {MITx_6.00x_UnitedStates_ Master's}는 4988 건으로 confidence 를 계산한다면 [1]의 경우 409(lhs,rhs 동시포함건수) /2408(전체에서 lhs 건수)=0.1698505, [2]의 경우 409(lhs,rhs 동시포함건수)/4988(전체에서 lhs 건수)=0.08199679 이다. Lift 의 경우 409(동시포함건수)/2408*4988(lhs 포함건수*rhs 포함건수) *335650(전체건수)=11.42949474 로 계산되기 때문에 [1],[2] 상관없이 같다. transaction 데이터를 구성할 때 어느 데이터가 앞에 위치하느냐에 따라 lhs, rhs 가 결정되기 때문에 이러한 상황이 발생하게 된다.