Multivariate Data Analysis

Assignment #1

Association Rule Mining: MOOC Dataset

과목명	다변량분석
담당교수	강필성 교수님
제출일	2019-04-09
이름	박지원
학과명	산업경영공학부
학번	2014170856

목차

Step1 데이터 변환

연관규칙분석 전, 다음과 같은 과정을 거쳐 연관규칙분석을 위한 데이터를 준비했습니다. Region에 해당하는 변수에서 특수문자가 포함되어 있는 경우, 분석이 제대로 되지 않아 모두 제거했습니다.

```
#1 csv파일에서Item Name에 해당하는 네 개의 변수를 불러와Institute, Course, Region, Degree에 저장합니다.
mooc_dataset <- read.csv("big_student_clear_third_version.csv")
mooc_dataset
Institute <- mooc_dataset$institute
Course <- mooc_dataset$course_id
Region <- mooc_dataset$final_cc_cname_DI
Degree <- mooc_dataset$LoE_DI

#2 Region에 해당하는 변수에서 한 칸 공백(“ “) 및 특수문자를 제거합니다.
Region <- gsub(" ", "", Region)
Region <- gsub("\\/", "", Region)
Region <- gsub("\\&", "", Region)
Region <- gsub("\\.", "", Region)
Region <- gsub("\\,", "", Region)


#3 네 변수를 밑줄(_)로 연결하여 RawTransactions에 저장합니다.
RawTransactions <- paste(Institute, Course, Region, Degree, sep = "_")

#4
MOOC_transactions <- paste(mooc_dataset$userid_DI, RawTransactions)
head(MOOC_transactions)

#5 MOOC_transactions 변수를 MOOC_User_Course.csv라는 파일명으로 저장합니다.
write.csv(MOOC_transactions, file="MOOC_User_Course.csv", row.names = FALSE)
```

MOOC_User_Course.csv의 내용은 다음과 같습니다. write.csv()함수를 사용해서 파일을 저장했을 때, 1행에 X가 저장되어 이후 분석이 제대로 되지 않아 1행을 삭제했습니다.

	A	B	C	D	E	F	G	H	I	J
1	MHxPC130313697	HarvardX_PH207x_India_Bachelor's								
2	MHxPC130237753	HarvardX_PH207x_UnitedStates_Secondary								
3	MHxPC130202970	HarvardX_CS50x_UnitedStates_Bachelor's								
4	MHxPC130223941	HarvardX_CS50x_OtherMiddleEastCentralAsia_Secondary								
5	MHxPC130317399	HarvardX_PH207x_Australia_Master's								
6	MHxPC130191782	HarvardX_CS50x_Pakistan_Bachelor's								
7	MHxPC130191782	HarvardX_ER22x_Pakistan_Bachelor's								
8	MHxPC130267000	HarvardX_PH207x_OtherSouthAsia_Master's								
9	MHxPC130435800	HarvardX_CS50x_India_Bachelor's								
10	MHxPC130284813	HarvardX_PH207x_UnitedStates_Bachelor's								
11	MHxPC130235150	HarvardX_CS50x_India_Bachelor's								
12	MHxPC130001411	HarvardX_CS50x_OtherEurope_Secondary								
13	MHxPC130396873	HarvardX_PH207x_UnitedStates_Bachelor's								
14	MHxPC130469401	HarvardX_CB22x_OtherMiddleEastCentralAsia_Bachelor's								
15	MHxPC130469401	HarvardX_CS50x_OtherMiddleEastCentralAsia_Bachelor's								
16	MHxPC130469401	HarvardX_ER22x_OtherMiddleEastCentralAsia_Bachelor's								

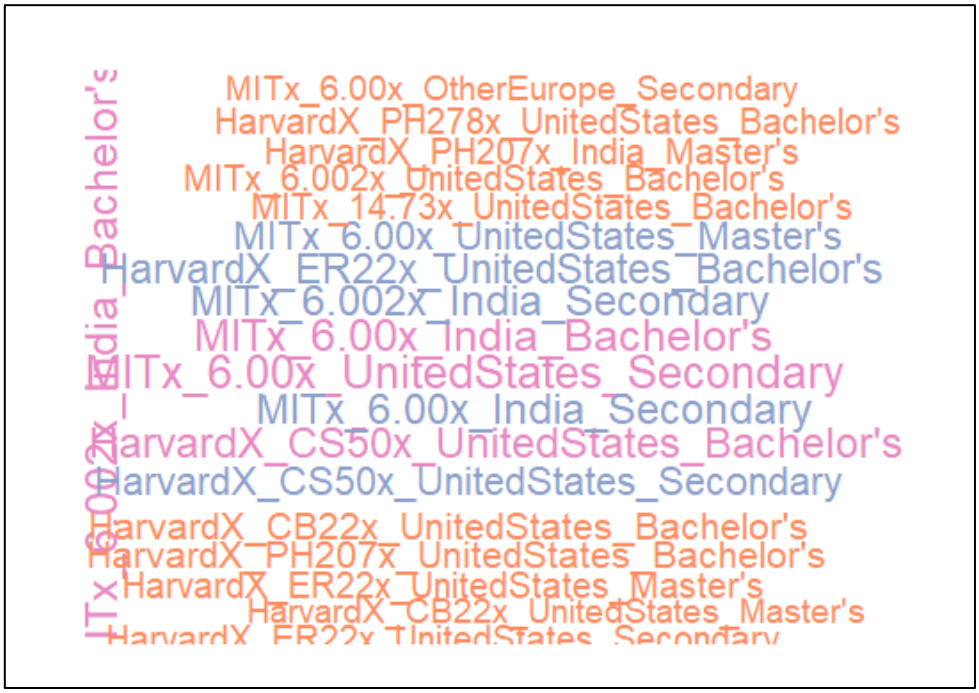
Step2 데이터 불러오기 및 기초 통계량 확인

위에서 생성된 single format의 데이터를 read.transactions( )함수를 이용하여 읽어 들였습니다.
데이터의 속성을 파악하기 위해 summary()함수를 사용했습니다. 아래 그림은 summary(MOOC)의 결과창 입니다.

```
> summary(MOOC)
transactions as itemMatrix in sparse format with
 335649 rows (elements/itemsets/transactions) and
 1405 columns (items) and a density of 0.0008771195

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                                 367749

element (itemset/transaction) length distribution:
sizes
     1      2      3      4      5      6      7      8      9     10     11     12     13
278439  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
```

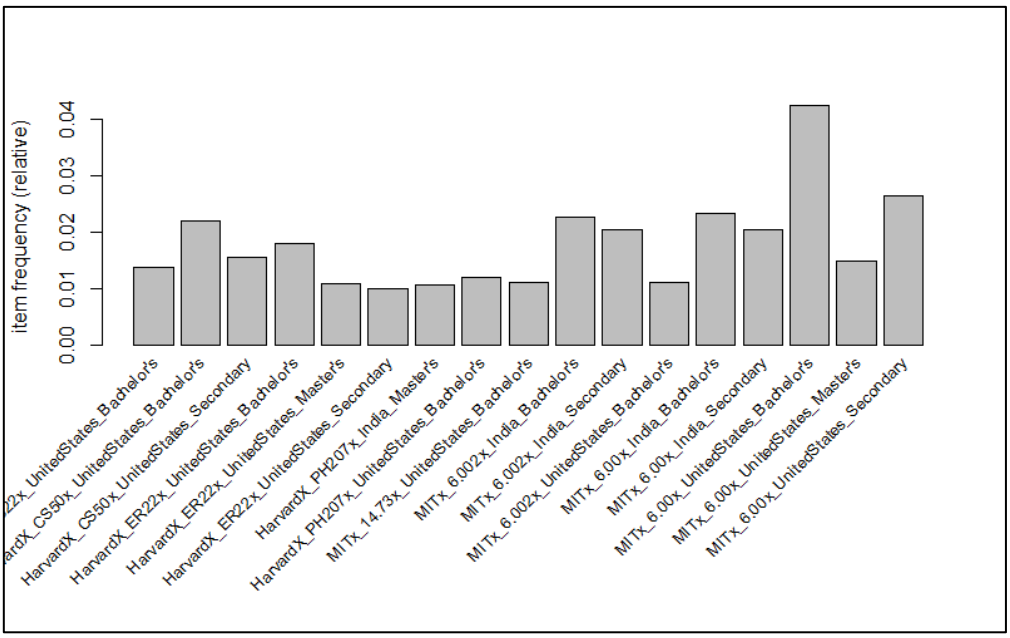
- 기존 416921개였던 데이터에서 중복이 제거되고 335649개가 남았습니다. 1405개의 아이템이 있음을 알 수 있습니다.
- 가장 자주 등장한 아이템은 MITx_6.00x_UnitedStates_Bachelor's(14192회)이었습니다. 이는 2위인 MITx_6.00x_UnitedStates_Secondary에 비해 약 1.6배 많은 수치로, 압도적인 1위임을 알 수 있습니다.
- 한 transaction당 1개 아이템이 들어간 경우가 278439개로 가장 많았습니다. 최대인 13까지 아이템의 개수가 늘어날 수록 빈도수가 급격하게 줄어들었습니다. 평균은 1.232 개였고, median도 1 이었습니다.

다음으로, 아이템 이름과 아이템 카운트를 이용하여 워드클라우드를 생성했습니다. 색상은 (6, "Set2")를 사용했고, min.freq는 3000으로 지정했습니다.



MITx_6.00x_UnitedStates_Bachelor's, MITx_6.00x_UnitedStates_Secondary, HarvardX_CS50x_UnitedStates_Bachelor's, MITx_6.002x_India_Bachelor's 등의 frequency가 높음을 알 수 있습니다.

itemFrequencyPlot( )함수를 사용하여 최소 빈도 1% 이상 등장한 Items들의 Bar Chart를 그렸습니다. 결과는 다음과 같습니다.



이로부터 파악한 상위 5개의 Item과 support, 접속 국가는 다음과 같습니다.

순위	Item	Support	접속 국가
1	MITx_6.00x_UnitedStates_Bachelor's	0.04228227	미국
2	MITx_6.00x_UnitedStates_Secondary	0.02634002	미국
3	MITx_6.00x_India_Bachelor's	0.02327729	인도
4	MITx_6.002x_India_Bachelor's	0.02274102	인도
5	HarvardX_CS50x_UnitedStates_Bachelor's	0.02207663	미국

미국과 인도에서 접속이 주로 활발함을 알 수 있습니다. 또, MIT의 6.00x 강의가 인기가 많은 것으로 보입니다.

Step3 규칙 생성 및 결과 해석

Support와 Confidence의 값을 조정해가며 다음과 같이 규칙을 생성했습니다. 각 값에 따라 생성된 규칙의 개수는 아래와 같습니다.

		Confidence			
		0.01	0.03	0.05	0.1
Support	0.001	73	56	51	34
	0.003	23	7	6	5
	0.005	17	1	0	0
	0.01	17	4	0	0

Support = 0.001, confidence = 0.05로 지정하여 51개의 규칙을 생성했습니다.[Appendix 1]

(1) Support가 가장 높은 규칙은 다음 두 규칙이었습니다. Support값은 모두 0.003643687입니다.

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

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

조건절(A)과 결과절(B)의 위치만 바뀌었습니다. 수업시간에 배웠던 support는 각각의 발생 확률(P(A)와 P(B))이지만 R 패키지에서는 A, B의 동시 발생 확률, 즉 P(A∩B)로 계산되기 때문에 이러한 결과가 나왔다고 생각합니다.

(2) Confidence가 가장 높은 규칙은 0.38810900의 값을 가진 다음 규칙이었습니다.

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

MITx에서 제공하는 8.02x라는 강의와 6.002x라는 강의가 연관이 있을 것으로 예상됩니다.

(3) Lift가 가장 높은 규칙은 19.549719의 값을 가진 다음 두 규칙이었습니다.

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

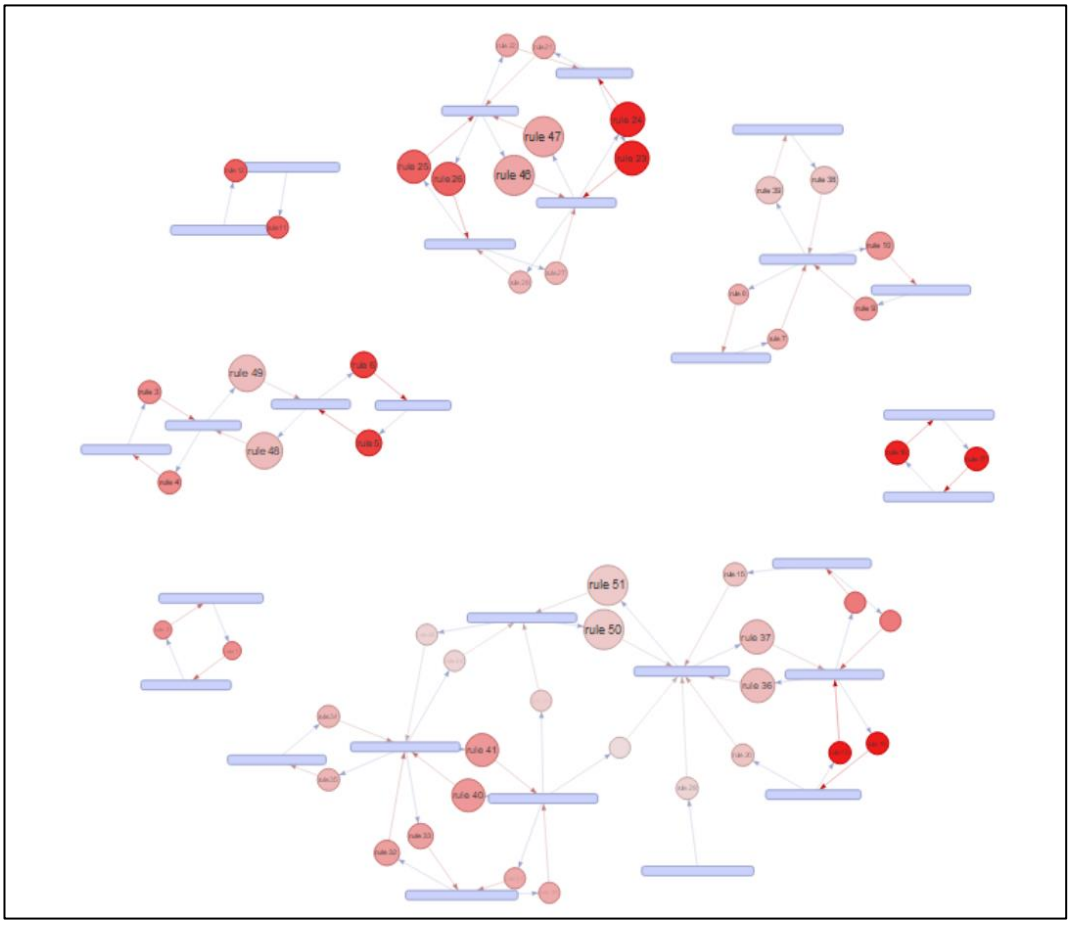
{MITx_6.002x_UnitedStates_Bachelor's} => {MITx_8.02x_UnitedStates_Bachelor's}

이 역시 조건절과 결과절만 바뀌어 중복되어 있습니다. Lift는 $\frac{P(A \cap B)}{P(A).P(B)}$로 계산되는데, (1)에서 둘의 support값이 같게 나왔기 때문에 lift도 동일하게 계산되었다고 생각합니다.

(4) 효용성 지표를 Support×Confidence×Lift로 정의했을 때 효용성이 가장 높은 규칙 1위~3위와 각각의 효용성 지표는 다음과 같습니다.

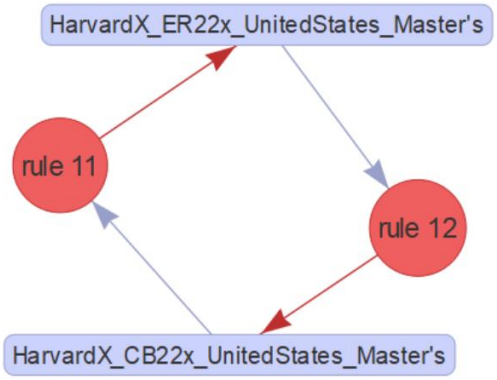
순위	규칙	효용성 지표
1	{MITx_8.02x_India_Secondary} => {MITx_6.002x_India_Secondary}	0.0206641682
2	{MITx_8.02x_India_Bachelor's} => {MITx_6.002x_India_Bachelor's}	0.0163274116
3	{HarvardX_CS50x_India_Secondary} => {MITx_6.00x_India_Secondary}	0.0113375620

(5) 생성된 규칙을 plot( )함수의 "graph" method를 이용하여 도시했습니다.



위 결과로부터 두 아이템이 서로 조건절/결과절을 달리해서 생성되는 경우가 존재함을 확인할 수 있습니다. 그 중, 3가지 케이스를 살펴보았습니다.

(i) HarvardX_ER22x_UnitedStates_Master's ⇔ HarvardX_CB22x_UnitedStates_Master's



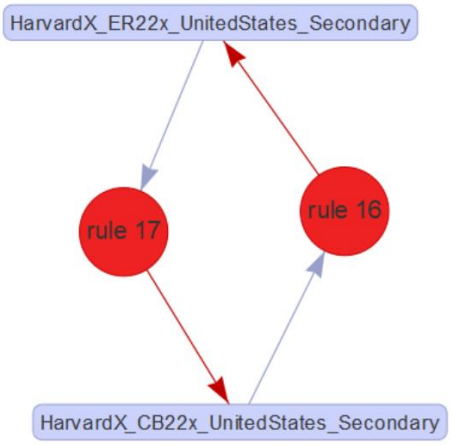
	Rule11	Rule12
Support	0.00142	0.00142
Confidence	0.158	0.131
Lift	14.6	14.6

두가지 규칙 모두 support와 lift값은 동일했습니다. 위에서 서술한 바와 같이 support를 조건절과 결과절의 동시발생확률로 계산하기 때문입니다.

Confidence값은 HarvardX_CB22x_UnitedStates_Master's를 조건으로 갖는 Rule11이 0.158로 HarvardX_ER22x_UnitedStates_Master's를 조건으로 갖는 Rule12의 0.131에 비해 높았습니다.

Confidence는 A->B일 때 $\frac{P(A \cap B)}{P(A)}$로 계산되고 B->A일 때 $\frac{P(A \cap B)}{P(B)}$ 로 계산됩니다. 분자가 같으므로 분모만 비교했을 때, confidence가 높다는 것은 분모가 더 작다는 것을 의미합니다. 따라서 confidence가 더 큰 Rule11의 조건인 HarvardX_CB22x_UnitedStates_Master's의 발생확률이 더 낮음을 의미합니다.

(ii) HarvardX_ER22x_UnitedStates_Secondary ⇔ HarvardX_CB22x_UnitedStates_Secondary



	Rule16	Rule17
Support	0.00154	0.00154
Confidence	0.192	0.153
Lift	19.1	19.1

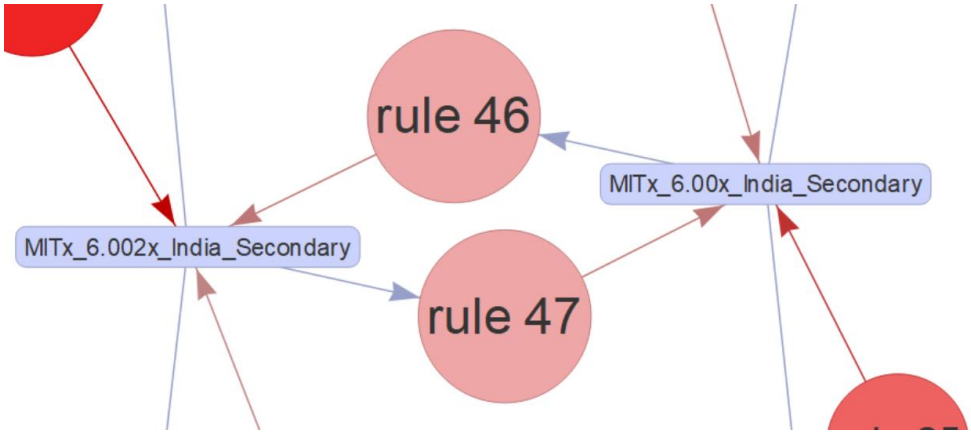
이번에도 두가지 규칙 모두 support와 lift값은 동일했습니다.

Confidence값은 HarvardX_CB22x_UnitedStates_Secondary를 조건으로 갖는 Rule16이 0.192로 HarvardX_ER22x_UnitedStates_Secondary를 조건으로 갖는 Rule17의 0.153에 비해 높았습니다.

Confidence가 더 큰 Rule16의 조건인 HarvardX_CB22x_UnitedStates_Secondary의 발생확률이 더 낮음을 의미합니다.

Rule 11, 12와 같은 강의, 같은 국가 접속이지만 Rule 16, 17처럼 secondary일 경우 support, confidence, lift가 더 높았습니다.

(iii) MITx_6.00x_India_Secondary ⇔ MITx_6.002x_India_Secondary

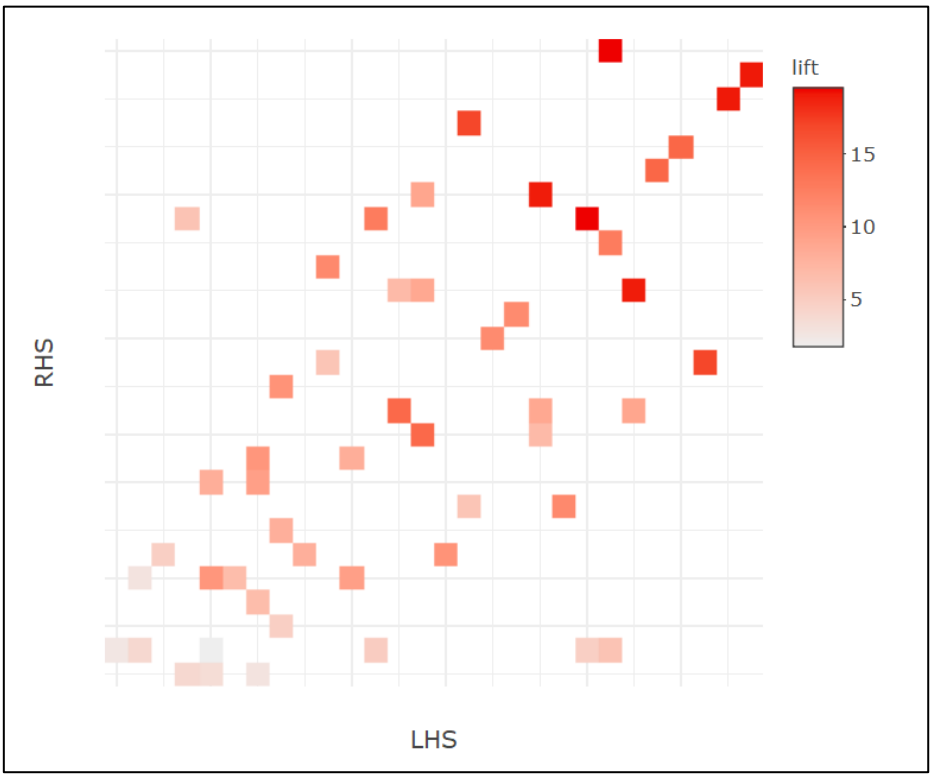


	Rule46	Rule47
Support	0. 00363	0. 00363
Confidence	0. 177	0. 178
Lift	8.69	8.69

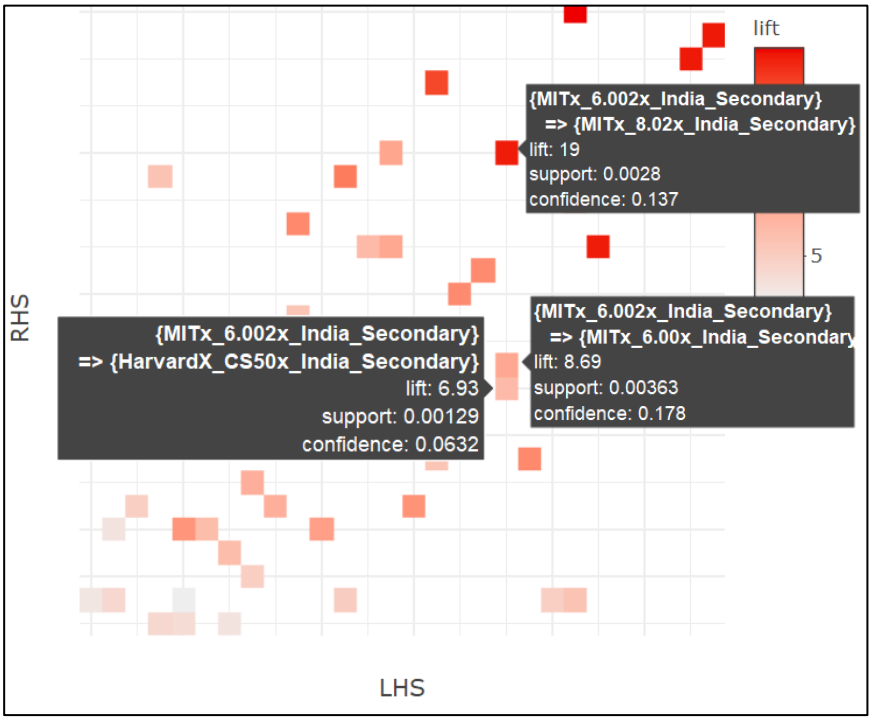
이번에도 두가지 규칙 모두 support와 lift값은 동일했습니다. Confidence의 경우 MITx_6.002x_India_Secondary를 조건으로 갖는 Rule47이 0.178로 MITx_6.00x_India_Secondary를 조건으로 갖는 Rule46의 0.177에 비해 약간 높았습니다. 미미한 차이로 보아 둘의 발생 확률이 크게 차이가 나지 않을 것으로 예상됩니다.

기타 연관규칙분석 시각화 및 해석

Matrix method를 사용해 연관규칙분석 시각화를 해보았습니다. 가로축이 LHS, 세로축이 RHS로 각 칸은 아이템을 나타냅니다. 박스의 색상이 진할수록 lift값이 큰 규칙입니다. Graph method에 비해 어떤 규칙들이 연관되어 있는지, 어떤 규칙들의 support가 큰지 한눈에 보기 어렵다는 단점이 있습니다.



동일한 LHS를 갖는 룰들이 여러 개 있고, 반대로 동일한 RHS를 갖는 룰들이 여럿 있음을 파악할 수 있습니다. 그 중에서 같은 LHS를 가진 3개의 룰을 비교해 보았습니다. MITx_6.002x_India_Secondary를 조건절로 갖는 규칙들은 각각 MITx_8.02x_India_Secondary, MITx_6.00x_India_Secondary, 그리고 HarvardX_CS50x_India_Secondary를 결과절로 가졌습니다. P(A)로 support를 계산했다면 세 규칙 모두 동일한 값이 나왔어야 하지만 이 패키지에서는 P(A∩B)로 계산되어 제각각 다른 값이 나왔습니다.



Appendix

1. Support = 0.001, confidence = 0.05로 지정하여 만든 51개의 규칙에 대하여 support, confidence, lift, 그리고 임의의 효용성 지표에 대한 순위를 검색한 결과입니다.

(1) Support

```
> inspect(sort(rules, by="support"))
     lhs                                                rhs                                                support     confidence lift     count
[1]  {HarvardX_CS50x_UnitedStates_Bachelor's}  => {MITx_6.00x_UnitedStates_Bachelor's}      0.003643687 0.16504723 3.903462 1223
[2]  {MITx_6.00x_UnitedStates_Bachelor's}      => {HarvardX_CS50x_UnitedStates_Bachelor's} 0.003643687 0.08617531 3.903462 1223
[3]  {MITx_6.00x_India_Secondary}              => {MITx_6.002x_India_Secondary}            0.003625811 0.17745698 8.692828 1217
```

(2) Confidence

```
> inspect(sort(rules, by="confidence"))
     lhs                                        rhs                                        support     confidence lift      count
[1]  {MITx_8.02x_India_Secondary}               => {MITx_6.002x_India_Secondary}           0.002800545 0.38810900 19.011734 940
[2]  {MITx_8.02x_India_Bachelor's}              => {MITx_6.002x_India_Bachelor's}          0.002496656 0.38564197 16.957990 838
[3]  {HarvardX_CS50x_India_Secondary}           => {MITx_6.00x_India_Secondary}            0.002681373 0.29392554 14.385508 900
```

(3) Lift

```
> inspect(sort(rules, by="lift"))
     lhs                                        rhs                                        support     confidence lift      count
[1]  {MITx_8.02x_UnitedStates_Bachelor's}       => {MITx_6.002x_UnitedStates_Bachelor's}   0.001391334 0.21620370 19.549719 467
[2]  {MITx_6.002x_UnitedStates_Bachelor's}      => {MITx_8.02x_UnitedStates_Bachelor's}    0.001391334 0.12580819 19.549719 467
[3]  {HarvardX_CB22x_UnitedStates_Secondary}    => {HarvardX_ER22x_UnitedStates_Secondary} 0.001540300 0.19240789 19.106957 517
```

(4) Support X Confidence X Lift

```
> inspect(sort(rules, by="measure"))
     lhs                                        rhs                                        support     confidence lift      count measure
[1]  {MITx_8.02x_India_Secondary}               => {MITx_6.002x_India_Secondary}           0.002800545 0.38810900 19.011734 940   0.0206641682
[2]  {MITx_8.02x_India_Bachelor's}              => {MITx_6.002x_India_Bachelor's}          0.002496656 0.38564197 16.957990 838   0.0163274116
[3]  {HarvardX_CS50x_India_Secondary}           => {MITx_6.00x_India_Secondary}            0.002681373 0.29392554 14.385508 900   0.0113375620
```

2. 전체 코드

```
##MDA Assignment#1 Association Rule Mining
######Step1 Data transformation######
library(readr)
library(arules)
library(arulesViz)
library(wordcloud)



#1 csv파일에서Item Name에 해당하는 네 개의 변수를 불러와Institute, Course, Region, Degree에 저장합니다.
mooc_dataset <- read.csv("big_student_clear_third_version.csv")
Institute <- mooc_dataset$institute
Course <- mooc_dataset$course_id
Region <- mooc_dataset$final_cc_cname_DI
Degree <- mooc_dataset$LoE_DI
```

```
#2 Region에 해당하는 변수에서 한 칸 공백(" ") 및 특수문자를 제거합니다.
Region <- gsub(" ", "", Region)
Region <- gsub("₩₩/", "", Region)
Region <- gsub("₩₩&", "", Region)
Region <- gsub("₩₩.", "", Region)
Region <- gsub("₩₩,", "", Region)

#3 네 변수를 밑줄(_)로 연결하여 RawTransactions에 저장합니다.
RawTransactions <- paste(Institute, Course, Region, Degree, sep = "_")

#4
MOOC_transactions <- paste(mooc_dataset$userid_DI, RawTransactions)

#5 MOOC_transactions 변수를 MOOC_User_Course.csv라는 파일명으로 저장합니다.
write.csv(MOOC_transactions, file="MOOC_User_Course.csv", row.names = FALSE)




######STEP2 데이터 불러오기 및 기초 통계량 확인######
#1 Read MOOC_User_Course.csv using read.transactions()
MOOC <- read.transactions("MOOC_User_Course.csv", format = "single", cols = c(1,2),
rm.duplicates=TRUE, skip =1)
summary(MOOC)

#2 Draw Wordcloud
itemName <- itemLabels(MOOC)
itemCount <- itemFrequency(MOOC)*nrow(MOOC)
summary(itemName)

col <- brewer.pal(6, "Set2") #Color를 Set2의 6로 지정합니다.
wordcloud(words = itemName, freq = itemCount, min.freq = 3000, scale = c(2, 1), col = col , random.order
= FALSE)

#3 Draw itemFrequencyPlot
itemFrequencyPlot(MOOC, support = 0.01, cex.names=0.8)
support1 <- itemFrequency(MOOC)
top5 <- support1[order(support1, decreasing=TRUE)[1:5],drop=FALSE]
top5  #상위 5개의 서포트값을 살펴보았습니다.
```

```
#####STEP3 규칙 생성 및 결과 해석 ######
# Rule generation by Apriori
rules <- apriori(MOOC, parameter=list(support=0.001, confidence=0.05))

# Check the generated rules
inspect(rules)

# List the first three rules with the highest lift values
inspect(sort(rules, by="support"))
inspect(sort(rules, by="confidence"))
inspect(sort(rules, by="lift"))

#Support ×  Confidence ×  Lift로 정의한 효용성 지표가 높은 1~3순위를 찾았습니다.
rules@quality$support
lhs1 <- rules@lhs
rhs1 <- rules@rhs
measure <- (rules@quality$support)*(rules@quality$confidence)*(rules@quality$lift)
rules@quality <- cbind(rules@quality, measure)
inspect(rules@quality)
rules@quality
inspect(rules)
inspect(sort(rules, by="measure"))



# Plot the rules
plot(rules, method="graph",engine = "htmlwidget")
plot(rules, method="matrix",engine = "htmlwidget")
```