다변량분석 Assignment 1

2014170824 황태민

[Q1]

Step 1 - R script

```
#Assignment 1
library(arules)
library(arulesViz)
library(wordcloud)

# step 1
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
Region<-gsub(" ","",Region)
RawTransactions<-paste(Institute, Course, Region, Degree, sep='_')
MOOC_transactions<-paste(mooc_dataset$userid_DI, RawTransactions, sep=' ')
write.table(MOOC_transactions, file="MOOC_User_Course.csv", col.names = FALSE, row.names = FALSE, quote = FALSE)
```

"MOOC_User_Course.csv" 결과는 다음과 같다.

	A	B	C	D	E	F	G	H
1	MHxPC130313697	HarvardX_PH207x_India_Bachelor's						
2	MHxPC130237753	HarvardX_PH207x_UnitedStates_Secondary						
3	MHxPC130202970	HarvardX_CS50x_UnitedStates_Bachelor's						
4	MHxPC130223941	HarvardX_CS50x_OtherMiddleEast/CentralAsia_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_OtherMiddleEast/CentralAsia_Bachelor's						
15	MHxPC130469401	HarvardX_CS50x_OtherMiddleEast/CentralAsia_Bachelor's						

[Q2]

Step 2 - R script

```
# step 2
tmp_single <- read.transactions("MOOC_User_Course.csv", format = "single", cols = c(1,2), rm.duplicates = TRUE)
summary(tmp_single)
itemName <- itemLabels(tmp_single)
itemCount <- itemFrequency(tmp_single)*nrow(tmp_single)
col <- brewer.pal(9, "Reds")
wordcloud(words = itemName, freq = itemCount, min.freq = 500, scale = c(1, 0.2), col = col, random.order = FALSE)
itemFrequencyPlot(tmp_single, support = 0.01, cex.names=0.8)
itemFrequencyPlot(tmp_single, support = 0.01, cex.names=0.8, topN = 5)
```

[Q2-1]

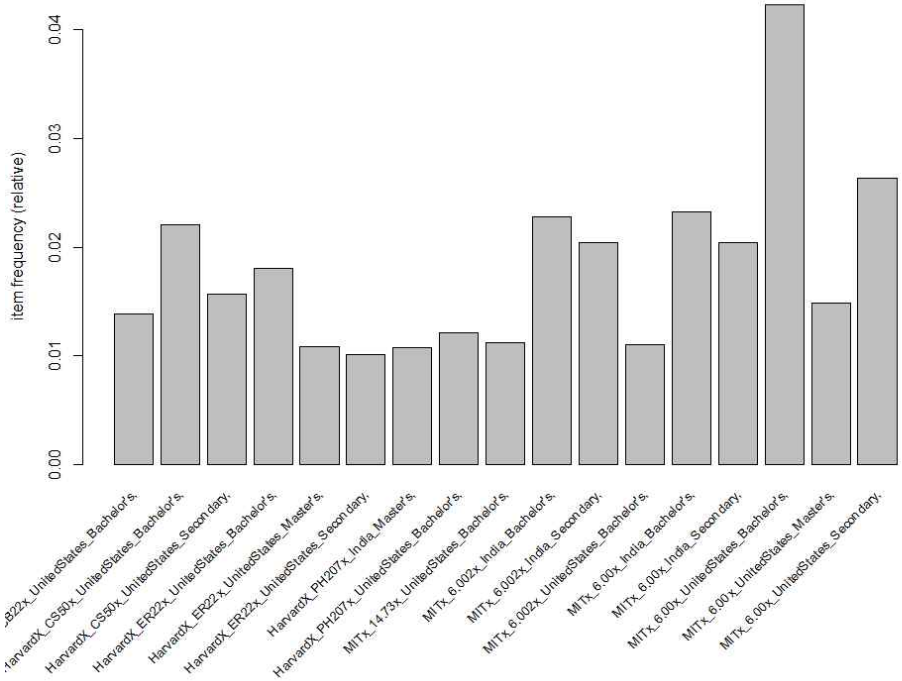
```
> summary(tmp_single)
transactions as itemMatrix in sparse format with
 335650 rows (elements/itemsets/transactions) and
 1405 columns (items) and a density of 0.000877119

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
```

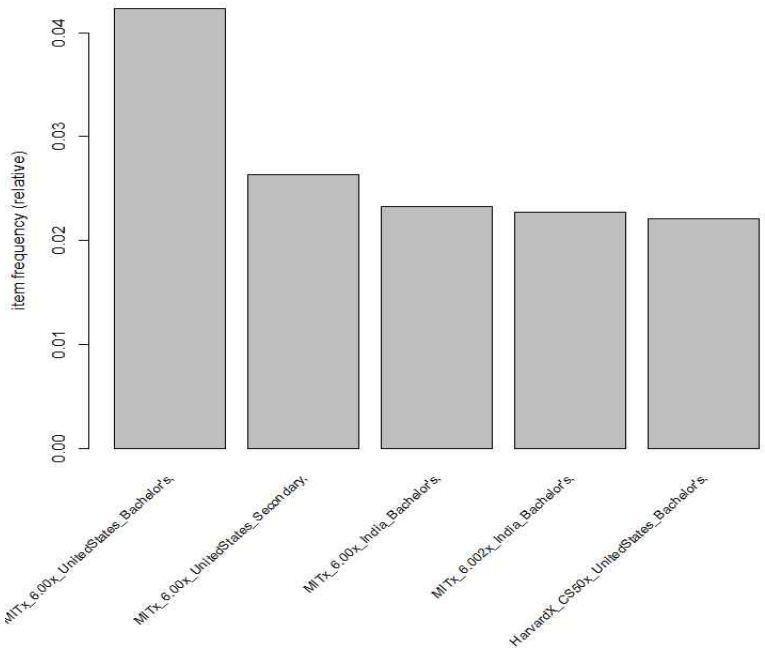
해당 데이터는 335650개의 데이터를 보유하고 있다. 그리고 1405 종류의 items를 보유하고 있다. 가장 빈도수가 높은 아이템은 MITx_6.00x_UnitedStates_Bachelor's로 14192번의 빈도수를 보인다. 한 user가 가장 많은 강의를 수강한 것은 13개가 최다이다. 평균적으로 1.232개의 MOOC 강좌를 수강함을 알 수 있다. 중위값은 1개임을 알 수 있다.

[Q2-2]

MITx_6.00x_UnitedStates_Bachelor's가 가장 빈도수가 높음을 볼 수 있다.

[Q2-3]





기존의 itemFrequencyPlot 함수로는 상위 5개의 항목을 명확하게 구분하기 어려워 상위 5개를 추려내는 옵션을 통해 확인하였다. 상위 5개 Item에 대한 접속 국가는 다음과 같다. 1위 United States, 2위 United States, 3위 India, 4위 India 그리고 5위는 United States이다.

[Q3]

Step 3 - R script

```
# step 3
rules <- apriori(tmp_single, parameter=list(support=0.0005, confidence=0.05))
rules <- apriori(tmp_single, parameter=list(support=0.0005, confidence=0.1))
rules <- apriori(tmp_single, parameter=list(support=0.0005, confidence=0.15))

rules <- apriori(tmp_single, parameter=list(support=0.001, confidence=0.05))
rules <- apriori(tmp_single, parameter=list(support=0.001, confidence=0.1))
rules <- apriori(tmp_single, parameter=list(support=0.001, confidence=0.15))

rules <- apriori(tmp_single, parameter=list(support=0.0015, confidence=0.05))
rules <- apriori(tmp_single, parameter=list(support=0.0015, confidence=0.1))
rules <- apriori(tmp_single, parameter=list(support=0.0015, confidence=0.15))

rules <- apriori(tmp_single, parameter=list(support=0.001, confidence=0.05))
inspect(rules)
inspect(sort(rules, by="support"))
inspect(sort(rules, by="confidence"))
inspect(sort(rules, by="lift"))
write.csv(as(rules, "data.frame"), "MOOC_rules.csv", row.names = FALSE)

df_rules=data.frame( lhs = labels(lhs(rules)), rhs = labels(rhs(rules)), rules@quality)
new_criteria=c(df_rules$support*df_rules$confidence*df_rules$lift)
df_rules=cbind(df_rules,new_criteria)
head(df_rules[order(-df_rules$new_criteria),],3)

plot(rules, method = "graph")
plot(rules, method = "graph", engine = "interactive")
```

[Q3-1]

Number of Rules	Confidence=0.05	confidence=0.1	confidence=0.15
Support=0.0005	168	103	60
Support=0.001	51	34	20
Support=0.0015	29	22	15

support와 confidence의 값이 높아짐에 따라 RULES의 개수가 감소하는 것을 볼 수 있다.

[Q3-2]

#support가 가장 높은 규칙

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

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

두 개의 규칙이 같이 가장 높은 0.003643676의 support 값을 가진다.

#confidence가 가장 높은 규칙

{MITx_8.02x_India_Secondary,} => {MITx_6.002x_India_Secondary,} 의 규칙이 가장 높은 0.38810900의 confidence 값을 가진다.

#lift가 가장 높은 규칙

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

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

두 개의 규칙이 같이 가장 높은 19.549777의 lift 값을 가진다.

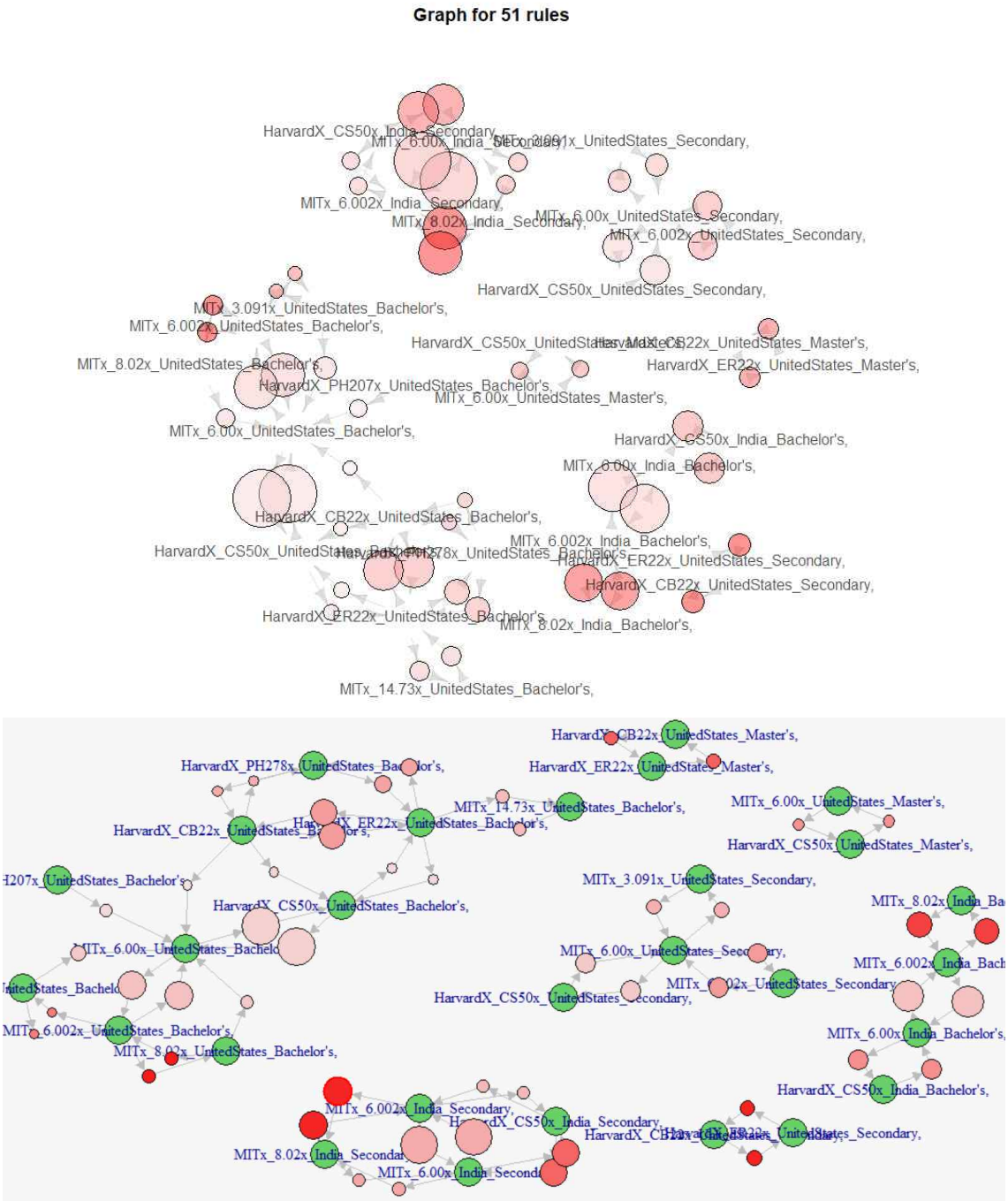
#support x confidence x lift가 가장 높은 규칙 1위~3위

```
> head(df_rules[order(-df_rules$new_criteria),],3)
                                lhs                                rhs     support confidence     lift count new_criteria
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)x(confidence)x(lift)를 새로운 효용성 지표로 설정하였을 때 상위 3개의 규칙의 위와 같다.

#X -> Y, Y -> X

기존의 graph method로는 조건절과 결과절의 글씨가 겹쳐 보여 구별이 힘들어서 plot 함수의 engine option을 이용하여 다음과 같은 시각화 방법을 통해 결과를 도출하였다.



첫 번째 규칙(support confidence lift count 순서)

```
{MITx_8.02x_India_Secondary,}          => {MITx_6.002x_India_Secondary,}          0.002800536 0.38810900 19.011790   940
{MITx_6.002x_India_Secondary,}         => {MITx_8.02x_India_Secondary,}           0.002800536 0.13718622 19.011790   940
```

두 번째 규칙

```
{MITx_3.091x_UnitedStates_Bachelor's,}     => {MITx_6.002x_UnitedStates_Bachelor's,}     0.001021898 0.14109420 12.758154   343
{MITx_6.002x_UnitedStates_Bachelor's,}     => {MITx_3.091x_UnitedStates_Bachelor's,}     0.001021898 0.09240302 12.758154   343
```

세 번째 규칙

```
{HarvardX_CS50x_India_Bachelor's,}         => {MITx_6.00x_India_Bachelor's,}          0.002016982 0.26918489 11.564304   677
{MITx_6.00x_India_Bachelor's,}             => {HarvardX_CS50x_India_Bachelor's,}      0.002016982 0.08665045 11.564304   677
```

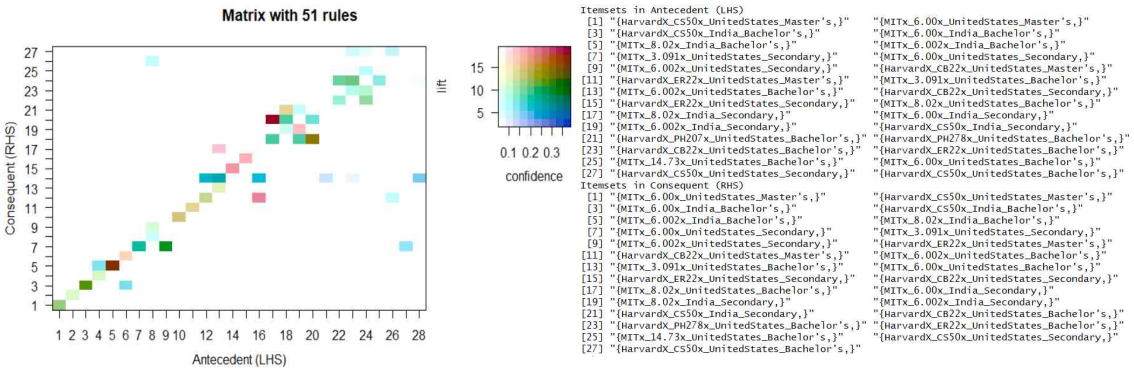
세 규칙 모두 support, lift, count에는 변화가 없으나 조건절과 결과절에 순서에 따라 confidence 값이 변화하는 것을 볼 수 있다. 이러한 이유는 confidence의 정의를 보면 왜 그러한 차이가 나는지를 알 수 있다. confidence(X->Y) = P(X,Y)/P(X) 이고, confidence(Y->X) = P(X,Y)/P(Y) 이므로 분자에 들어가는 값이 달라지므로 다음과 같은 차이를 보이게 되는 것이다.

[Extra Question]

Extra Question - R script

```
# [Extra Question]
plot(rules, method="matrix", engine = "3d")
plot(rules, method="matrix", shading=c("lift", "confidence"))
plot(rules, method="graph", engine="htmlwidget",
     igraphLayout = "layout_in_circle")
plot(rules, method = "grouped", gp_labels = gpar(cex=0.4))
```

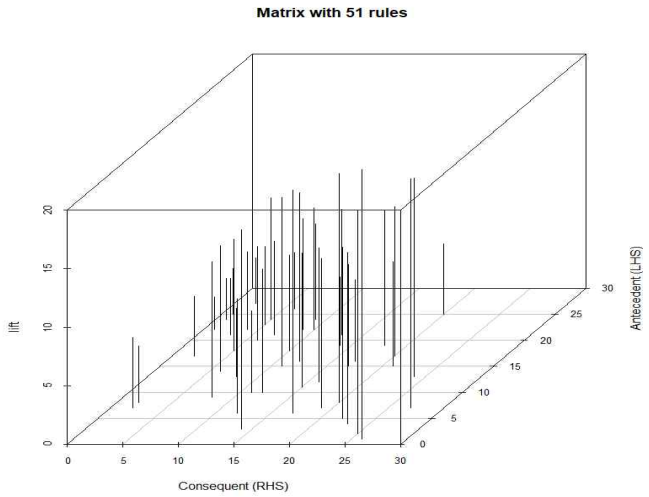
#plot(rules, method="matrix", shading=c("lift", "confidence"))을 사용한 결과는 다음과 같다.



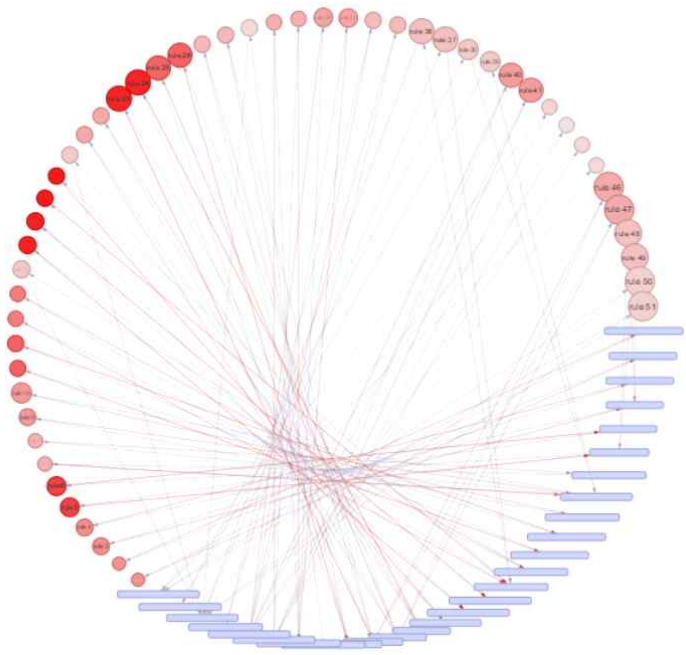

```
Itemsets in Antecedent (LHS)
 [1] "{HarvardX_CS50x_UnitedStates_Master's,}"   "{MITx_6.00x_UnitedStates_Master's,}"
 [3] "{HarvardX_CS50x_India_Bachelor's,}"        "{MITx_6.00x_India_Bachelor's,}"
 [5] "{MITx_8.02x_India_Bachelor's,}"            "{MITx_6.002x_India_Bachelor's,}"
 [7] "{MITx_3.091x_UnitedStates_Secondary,}"     "{MITx_6.00x_UnitedStates_Secondary,}"
 [9] "{MITx_6.002x_UnitedStates_Secondary,}"     "{HarvardX_CB22x_UnitedStates_Master's,}"
[11] "{HarvardX_ER22x_UnitedStates_Master's,}"   "{MITx_3.091x_UnitedStates_Bachelor's,}"
[13] "{MITx_6.002x_UnitedStates_Bachelor's,}"    "{HarvardX_CB22x_UnitedStates_Secondary,}"
[15] "{HarvardX_ER22x_UnitedStates_Secondary,}"  "{MITx_8.02x_UnitedStates_Bachelor's,}"
[17] "{MITx_8.02x_India_Secondary,}"             "{MITx_6.00x_India_Secondary,}"
[19] "{MITx_6.002x_India_Secondary,}"            "{HarvardX_CS50x_India_Secondary,}"
[21] "{HarvardX_PH207x_UnitedStates_Bachelor's,}" "{HarvardX_PH278x_UnitedStates_Bachelor's,}"
[23] "{HarvardX_CB22x_UnitedStates_Bachelor's,}" "{HarvardX_ER22x_UnitedStates_Bachelor's,}"
[25] "{MITx_14.73x_UnitedStates_Bachelor's,}"    "{MITx_6.00x_UnitedStates_Bachelor's,}"
[27] "{HarvardX_CS50x_UnitedStates_Secondary,}"  "{HarvardX_CS50x_UnitedStates_Bachelor's,}"
Itemsets in Consequent (RHS)
 [1] "{MITx_6.00x_UnitedStates_Master's,}"       "{HarvardX_CS50x_UnitedStates_Master's,}"
 [3] "{MITx_6.00x_India_Bachelor's,}"            "{HarvardX_CS50x_India_Bachelor's,}"
 [5] "{MITx_6.002x_India_Bachelor's,}"           "{MITx_8.02x_India_Bachelor's,}"
 [7] "{MITx_6.00x_UnitedStates_Secondary,}"      "{MITx_3.091x_UnitedStates_Secondary,}"
 [9] "{MITx_6.002x_UnitedStates_Secondary,}"     "{HarvardX_ER22x_UnitedStates_Master's,}"
[11] "{HarvardX_CB22x_UnitedStates_Master's,}"   "{MITx_6.002x_UnitedStates_Bachelor's,}"
[13] "{MITx_3.091x_UnitedStates_Bachelor's,}"    "{MITx_6.00x_UnitedStates_Bachelor's,}"
[15] "{HarvardX_ER22x_UnitedStates_Secondary,}"  "{HarvardX_CB22x_UnitedStates_Secondary,}"
[17] "{MITx_8.02x_UnitedStates_Bachelor's,}"     "{MITx_6.00x_India_Secondary,}"
[19] "{MITx_8.02x_India_Secondary,}"             "{MITx_6.002x_India_Secondary,}"
[21] "{HarvardX_CS50x_India_Secondary,}"         "{HarvardX_CB22x_UnitedStates_Bachelor's,}"
[23] "{HarvardX_PH278x_UnitedStates_Bachelor's,}" "{HarvardX_ER22x_UnitedStates_Bachelor's,}"
[25] "{MITx_14.73x_UnitedStates_Bachelor's,}"    "{HarvardX_CS50x_UnitedStates_Secondary,}"
[27] "{HarvardX_CS50x_UnitedStates_Bachelor's,}"
```

색이 진할수록 confidence가 높고 빨간색에 가까울수록 lift가 높음을 알 수 있다. 이 시각화 기법에서 우측 상단에 위치하고 있는 붉은색 네모칸에 해당하는 규칙이 {MITx_8.02x_India_Secondary,} => {MITx_6.002x_India_Secondary,} 이다.

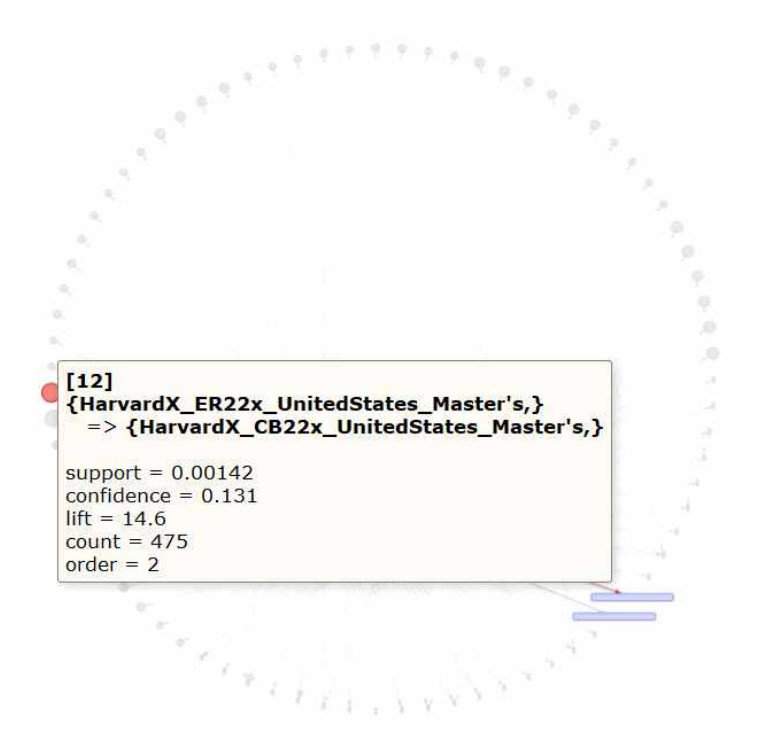
#plot(rules, method="matrix", engine = "3d")를 사용한 결과는 다음과 같다.
rule의 개수가 많은 데이터엔 적합하지 않아보인다.



#plot(rules, method="graph", engine="htmlwidget", igraphLayout = "layout_in_circle")를 사용하면 다음과 같은 시각화도 가능하다.

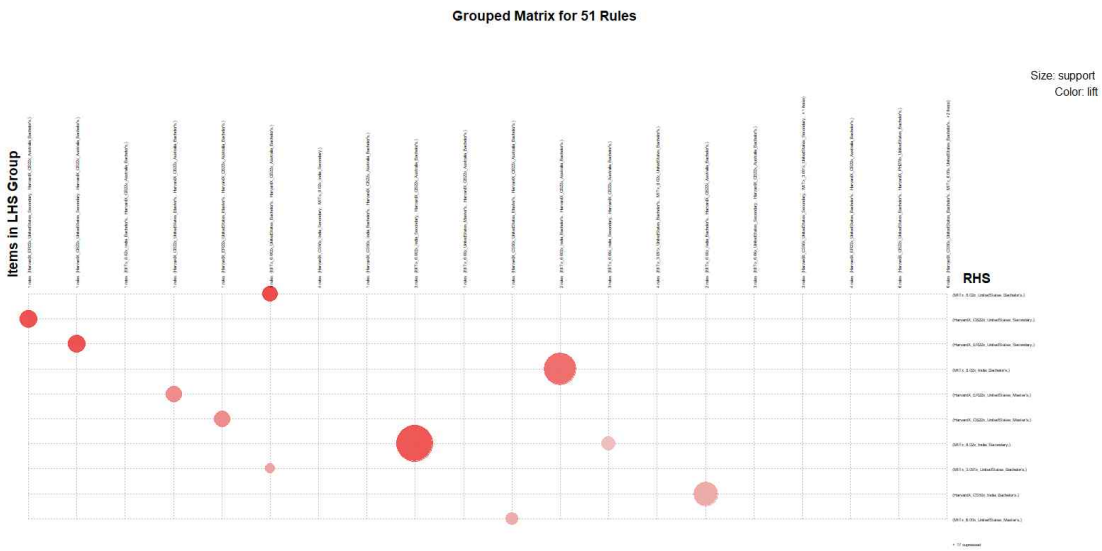


circle에 마우스를 가져다대면 무슨 rules인지 알려주는 기능 또한 포함되어 있다.
예시로 rule 12의 정보를 다음과 같이 볼 수 있다.



rule 12는 0.00142의 support 값을 가지고, 0.131의 confidence 값을 가지는 것을 알 수 있다.

#plot(rules, method = "grouped", gp_labels = gpar(cex=0.4))의 방법을 사용하면 다음과 같은 시각화가 가능하다.



이 경우에는 rule의 길이가 길어서 글자 크기를 줄여야만 그래프를 확인할 수 있었다. 원의 크기는 support를 나타내고 색이 진할수록 lift가 높다는 것을 알 수 있다.