Ejercicio Adicional (opcional)

Lucio Cornejo

Tabla de Contenidos


```
library(arules)
```

```
Warning: package 'arules' was built under R version 4.1.3
```

```
Loading required package: Matrix
```

```
Warning: package 'Matrix' was built under R version 4.1.3
```

```
Attaching package: 'arules'
```

```
The following objects are masked from 'package:base':

    abbreviate, write
```

Pregunta a

```
# Leer los identificadores y sus artistas respectivos
datos <- read.transactions(
  './lastfm.csv', header = TRUE, encoding = 'UTF-8',
  format = 'single', sep = ',', cols = c('user', 'artist'),
```

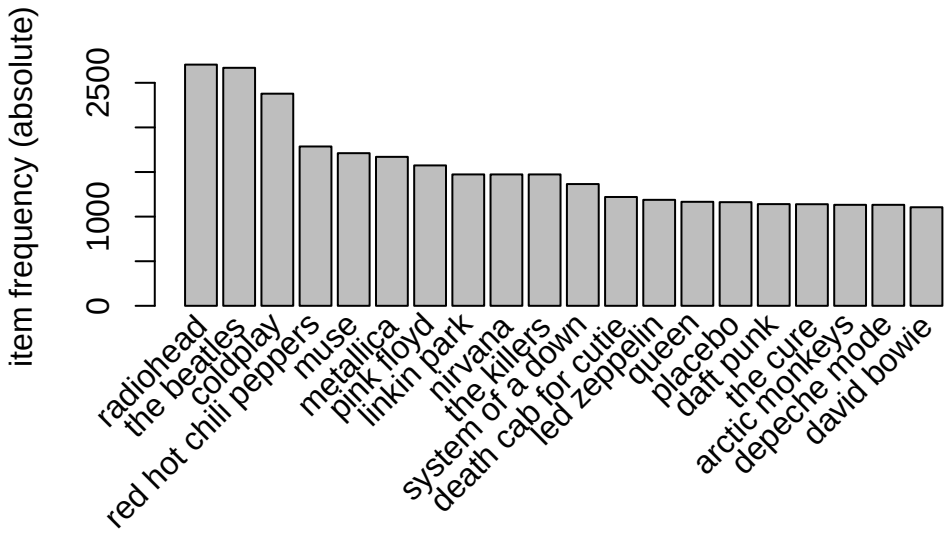
```
  rm.duplicates = TRUE
)
```

Parte i

```
summary(datos)
```

```
transactions as itemMatrix in sparse format with
 15000 rows (elements/itemsets/transactions) and
 1004 columns (items) and a density of 0.01925319

most frequent items:
            radiohead           the beatles              coldplay
                 2704                  2668                  2378
red hot chili peppers                muse               (Other)
                 1786                  1711                278706

element (itemset/transaction) length distribution:
sizes
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20
185 222 280 302 359 385 472 461 491 501 504 482 472 471 479 477 456 455 444 455
 21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40
436 478 426 438 408 446 417 375 348 340 316 293 274 286 238 208 193 181 128 102
 41  42  43  44  45  46  47  48  49  50  51  52  54  55  63  76
 93  61  55  36  23  15   6  11   2   1   5   3   1   2   1   1

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1.00   11.00   19.00   19.33   27.00   76.00

includes extended item information - examples:
                                     labels
1 ...and you will know us by the trail of dead
2                                  [unknown]
3                                       2pac

includes extended transaction information - examples:
  transactionID
1             1
2          1000
3         10000
```

```r
# Artistas más populares/frecuentes en los datos
itemFrequencyPlot(datos, topN = 20, type = 'absolute')
```




```r
# Aplicar algoritmo A priori
rules <- apriori(
  datos,
  parameter = list(supp = 0.01, conf = 0.2, target = "rules")
)
```

```
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen
        0.2    0.1    1 none FALSE            TRUE       5    0.01      1
 maxlen target  ext
     10  rules TRUE

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 150

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[1004 item(s), 15000 transaction(s)] done [0.12s].
```

```
sorting and recoding items ... [655 item(s)] done [0.01s].
creating transaction tree ... done [0.01s].
checking subsets of size 1 2 3 4 done [0.04s].
writing ... [1088 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

```r
rules <- sort(rules, by = 'confidence', decreasing = TRUE)
inspect(rules[1:20])
```

```
     lhs                                       rhs           support
[1]  {oasis, the killers}                   => {coldplay}    0.01113333
[2]  {sigur rós, the beatles}               => {radiohead}   0.01046667
[3]  {keane}                                => {coldplay}    0.02226667
[4]  {radiohead, snow patrol}               => {coldplay}    0.01006667
[5]  {coldplay, the smashing pumpkins}      => {radiohead}   0.01093333
[6]  {the beatles, the smashing pumpkins}   => {radiohead}   0.01146667
[7]  {bob dylan, pink floyd}                => {the beatles} 0.01033333
[8]  {led zeppelin, the doors}              => {pink floyd}  0.01066667
[9]  {snow patrol, the killers}             => {coldplay}    0.01040000
[10] {bob dylan, the rolling stones}        => {the beatles} 0.01146667
[11] {beck, the beatles}                    => {radiohead}   0.01300000
[12] {death cab for cutie, the killers}     => {coldplay}    0.01086667
[13] {oasis, radiohead}                     => {coldplay}    0.01273333
[14] {coldplay, sigur rós}                  => {radiohead}   0.01206667
[15] {the pussycat dolls}                   => {rihanna}     0.01040000
[16] {led zeppelin, the rolling stones}     => {the beatles} 0.01066667
[17] {david bowie, pink floyd}              => {the beatles} 0.01006667
[18] {bob dylan, radiohead}                 => {the beatles} 0.01386667
[19] {david bowie, the rolling stones}      => {the beatles} 0.01000000
[20] {the beatles, the shins}               => {radiohead}   0.01066667
     confidence coverage   lift      count
[1]  0.6626984  0.01680000  4.180183 167
[2]  0.6434426  0.01626667  3.569393 157
[3]  0.6374046  0.03493333  4.020634 334
[4]  0.6344538  0.01586667  4.002021 151
[5]  0.6283525  0.01740000  3.485683 164
[6]  0.6209386  0.01846667  3.444556 172
[7]  0.6150794  0.01680000  3.458092 155
[8]  0.5970149  0.01786667  5.689469 160
[9]  0.5954198  0.01746667  3.755802 156
[10] 0.5910653  0.01940000  3.323081 172
```

```
[11] 0.5909091  0.02200000  3.277972 195
[12] 0.5884477  0.01846667  3.711823 163
[13] 0.5876923  0.02166667  3.707058 191
[14] 0.5801282  0.02080000  3.218167 181
[15] 0.5777778  0.01800000 13.415893 156
[16] 0.5776173  0.01846667  3.247474 160
[17] 0.5741445  0.01753333  3.227949 151
[18] 0.5730028  0.02420000  3.221530 208
[19] 0.5703422  0.01753333  3.206572 150
[20] 0.5673759  0.01880000  3.147425 160
```

Partes ii y iii

```
summary(rules)
```

```
set of 1088 rules

rule length distribution (lhs + rhs):sizes
  2   3
881 207

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   2.00    2.00    2.00    2.19    2.00    3.00

summary of quality measures:
    support           confidence        coverage             lift
 Min.   :0.01000   Min.   :0.2002   Min.   :0.01587   Min.   : 1.131
 1st Qu.:0.01120   1st Qu.:0.2372   1st Qu.:0.03485   1st Qu.: 2.140
 Median :0.01310   Median :0.2867   Median :0.04927   Median : 2.829
 Mean   :0.01508   Mean   :0.3141   Mean   :0.05168   Mean   : 3.308
 3rd Qu.:0.01660   3rd Qu.:0.3700   3rd Qu.:0.06082   3rd Qu.: 3.985
 Max.   :0.05820   Max.   :0.6627   Max.   :0.18027   Max.   :14.053
     count
 Min.   :150.0
 1st Qu.:168.0
 Median :196.5
 Mean   :226.3
 3rd Qu.:249.0
 Max.   :873.0

mining info:
```

```
 data ntransactions support confidence
datos         15000    0.01        0.2
                                                                        call
apriori(data = datos, parameter = list(supp = 0.01, conf = 0.2, target = "rules"))
```

Como el **soporte máximo** resultó ser aproximadamente 6%, concluimos que para todas las reglas de asociación $X \to Y$ consideradas, el porcentaje de usarios que escucha a los artistas pertenecientes al itemset $X \cup Y$ es muy bajo (menor que 6%).

Dado que la **confianza máxima** resultó ser aproximadamente 66%, existe un par de itemsets X_0, Y_0 tales que aproximadamente 66% de las transacciones que contienen X_0, también contienen Y_0 . Este valor de confianza es relativamente grande, por lo que, a personas que escuchan los artistas pertenecientes a X_0, se esperaría que también les guste escuchar a los artistas pertenecientes al itemset Y_0 .

En ese sentido, a mayor confianza de una regla de asociación $X \to Y$, sería una mejor recomendación para una persona que le gusta escuchar a los artistas del itemset X, que escuche a los artistas del itemset Y .

En base a que el **lift mínimo** es mayor que 1, concluimos que, para las reglas de asociación $X \to Y$ consideradas, la ocurrencia del itemset X está positivamente correlacionada con la ocurrencia del itemset Y . Es decir, las reglas de asociación $X \to Y$ consideradas realmente sirven para recomendar a usuarios que escuchan artistas del itemset X, que prueben escuchar (pues muy probablemente les guste) a artistas del itemset Y .

En ese sentido, a mayor lift de una regla de asociación $X \to Y$, sería una mejor recomendación para una persona que le gusta escuchar a los artistas del itemset X, que escuche a los artistas del itemset Y .

Pregunta b

```r
lift_mayor_a_5 <- subset(rules, subset = lift > 5)
lift_mayor_a_5 <- sort(lift_mayor_a_5, by = 'lift', decreasing = TRUE)
summary(lift_mayor_a_5)
```

```
set of 136 rules

rule length distribution (lhs + rhs):sizes
  2   3
125  11

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
```

```
  2.000   2.000   2.000   2.081   2.000   3.000

summary of quality measures:
    support           confidence        coverage            lift
 Min.   :0.01000   Min.   :0.2002   Min.   :0.01787   Min.   : 5.007
 1st Qu.:0.01080   1st Qu.:0.2453   1st Qu.:0.03195   1st Qu.: 5.317
 Median :0.01167   Median :0.2999   Median :0.04150   Median : 5.897
 Mean   :0.01230   Mean   :0.3185   Mean   :0.04122   Mean   : 6.716
 3rd Qu.:0.01333   3rd Qu.:0.3681   3rd Qu.:0.04953   3rd Qu.: 7.729
 Max.   :0.02107   Max.   :0.5970   Max.   :0.07920   Max.   :14.053
     count
 Min.   :150.0
 1st Qu.:162.0
 Median :175.0
 Mean   :184.5
 3rd Qu.:200.0
 Max.   :316.0

mining info:
  data ntransactions support confidence
 datos         15000    0.01        0.2
                                                                                call
 apriori(data = datos, parameter = list(supp = 0.01, conf = 0.2, target = "rules"))
```

Tras inspeccionar las 136 reglas filtradas, mostramos las reglas de asociación $X \to Y$ tales que el itemset X contenga al artista **Judas Priest**.

```
inspect(lift_mayor_a_5[17])
```

```
    lhs                rhs            support    confidence coverage   lift
[1] {judas priest} => {iron maiden} 0.01353333 0.5075     0.02666667 8.562992
    count
[1] 203
```

Del valor aproximado de confianza, 50%, concluimos que, respecto a la base de datos, a casi el 50% de los usuarios que escuchan a Judas Priest, escucha también a **Iron Maiden**.

Asimismo, el valor del lift de la regla de asociación mostrada es mucho mayor que 1 (8.562992).

En base a aquellos dos valores numérico, se tiene que es **muy probable** que a usuarios que escuchen a Judas Priest, les guste escuchar a Iron Maiden, **artista que recomendaríamos**.

Tras inspeccionar las 136 reglas filtradas, mostramos las reglas de asociación tales que el itemset contenga al artista The Pussycat Dolls.

```
inspect(lift_mayor_a_5[4])
```

```
    lhs                       rhs       support confidence coverage lift
[1] {the pussycat dolls} => {rihanna} 0.0104  0.5777778  0.018    13.41589
    count
[1] 156
```

Respecto a la última regla de asociacón mostrada, de los valores relativamente altos que posee para confianza y lift (0.5777778 y 13.41589 respectivamente), similar a como se comentó para el caso de Judas Priest, se tiene que para usuarios que escuchen a The Pussycat Dolls, sería una buena recomendación (con mayor probabilidad de éxito que la recomendación que presentamos para usuarios que esuchan a Judas Priest, ya que confianza y lift son mayores, respectivamente) que escuchen a la artista **Rihanna**.

Pregunta c

```
# Filtro de lift y confianza
filtro_c <- subset(rules, subset = lift > 4)
filtro_c <- subset(filtro_c, subset = confidence > 0.35)
# Filtrar reglas cuyo itemset antecedente posea algún
# nombre de artista que contiene la palaba 'the'
filtro_c <- subset(filtro_c, subset = lhs %pin% 'the')
inspect(filtro_c)
```

```
     lhs                                rhs                     support
[1]  {oasis, the killers}            => {coldplay}              0.01113333
[2]  {led zeppelin, the doors}       => {pink floyd}            0.01066667
[3]  {the pussycat dolls}            => {rihanna}               0.01040000
[4]  {pink floyd, the doors}         => {led zeppelin}          0.01066667
[5]  {the postal service}            => {death cab for cutie}   0.01533333
[6]  {led zeppelin, the beatles}     => {pink floyd}            0.01560000
[7]  {radiohead, the shins}          => {death cab for cutie}   0.01006667
[8]  {panic at the disco}            => {fall out boy}          0.01153333
[9]  {the beatles, the doors}        => {pink floyd}            0.01000000
[10] {the decemberists}              => {death cab for cutie}   0.01346667
[11] {the shins}                     => {death cab for cutie}   0.02000000
[12] {pink floyd, the beatles}       => {led zeppelin}          0.01560000
```

```
[13] {the kooks}                        => {arctic monkeys}      0.01853333
[14] {explosions in the sky}            => {sigur rós}           0.01006667
[15] {the beatles, the rolling stones} => {bob dylan}           0.01146667
[16] {muse, the killers}                => {arctic monkeys}      0.01106667
[17] {the who}                          => {led zeppelin}        0.01293333
[18] {the who}                          => {the rolling stones}  0.01273333
[19] {the decemberists}                 => {the shins}           0.01160000
[20] {the beatles, the rolling stones} => {led zeppelin}        0.01066667
     confidence coverage   lift      count
[1]  0.6626984  0.01680000  4.180183 167
[2]  0.5970149  0.01786667  5.689469 160
[3]  0.5777778  0.01800000 13.415893 156
[4]  0.5387205  0.01980000  6.802027 160
[5]  0.4693878  0.03266667  5.771161 230
[6]  0.4661355  0.03346667  4.442206 234
[7]  0.4441176  0.02266667  5.460463 151
[8]  0.4346734  0.02653333  8.413033 173
[9]  0.4273504  0.02340000  4.072590 150
[10] 0.4234801  0.03180000  5.206722 202
[11] 0.4048583  0.04940000  4.977766 300
[12] 0.3932773  0.03966667  4.965623 234
[13] 0.3834483  0.04833333  5.081028 278
[14] 0.3822785  0.02633333  5.508335 151
[15] 0.3763676  0.03046667  5.428379 172
[16] 0.3721973  0.02973333  4.931943 166
[17] 0.3716475  0.03480000  4.692519 194
[18] 0.3659004  0.03480000  5.814095 191
[19] 0.3647799  0.03180000  7.384208 174
[20] 0.3501094  0.03046667  4.420573 160
```

```r
# Tras revisar las pocas reglas de asociación filtradas,
# filtramos solo las reglas de asociación cuyo itemset
# antecedent posee un nombre de artista que empieza
# con la la palabra 'the'
filtro_c <- filtro_c[-c(8, 14)]

# Ordenar de forma descendente respecto a confianza
filtro_c <- sort(filtro_c, by = 'confidence', decreasing = TRUE)
inspect(filtro_c)
```

```
    lhs                                rhs                    support
```

```
[1]  {oasis, the killers}                 => {coldplay}            0.01113333
[2]  {led zeppelin, the doors}            => {pink floyd}          0.01066667
[3]  {the pussycat dolls}                 => {rihanna}             0.01040000
[4]  {pink floyd, the doors}              => {led zeppelin}        0.01066667
[5]  {the postal service}                 => {death cab for cutie} 0.01533333
[6]  {led zeppelin, the beatles}          => {pink floyd}          0.01560000
[7]  {radiohead, the shins}               => {death cab for cutie} 0.01006667
[8]  {the beatles, the doors}             => {pink floyd}          0.01000000
[9]  {the decemberists}                   => {death cab for cutie} 0.01346667
[10] {the shins}                          => {death cab for cutie} 0.02000000
[11] {pink floyd, the beatles}            => {led zeppelin}        0.01560000
[12] {the kooks}                          => {arctic monkeys}      0.01853333
[13] {the beatles, the rolling stones} => {bob dylan}              0.01146667
[14] {muse, the killers}                  => {arctic monkeys}      0.01106667
[15] {the who}                            => {led zeppelin}        0.01293333
[16] {the who}                            => {the rolling stones}  0.01273333
[17] {the decemberists}                   => {the shins}           0.01160000
[18] {the beatles, the rolling stones} => {led zeppelin}           0.01066667
     confidence coverage   lift      count
[1]  0.6626984  0.01680000  4.180183 167
[2]  0.5970149  0.01786667  5.689469 160
[3]  0.5777778  0.01800000 13.415893 156
[4]  0.5387205  0.01980000  6.802027 160
[5]  0.4693878  0.03266667  5.771161 230
[6]  0.4661355  0.03346667  4.442206 234
[7]  0.4441176  0.02266667  5.460463 151
[8]  0.4273504  0.02340000  4.072590 150
[9]  0.4234801  0.03180000  5.206722 202
[10] 0.4048583  0.04940000  4.977766 300
[11] 0.3932773  0.03966667  4.965623 234
[12] 0.3834483  0.04833333  5.081028 278
[13] 0.3763676  0.03046667  5.428379 172
[14] 0.3721973  0.02973333  4.931943 166
[15] 0.3716475  0.03480000  4.692519 194
[16] 0.3659004  0.03480000  5.814095 191
[17] 0.3647799  0.03180000  7.384208 174
[18] 0.3501094  0.03046667  4.420573 160
```

```r
# Dos primeras reglas de asociación más relevantes
inspect(filtro_c[1:2])
```

```
   lhs                          rhs          support    confidence coverage
```

```
[1] {oasis, the killers}      => {coldplay}   0.01113333 0.6626984  0.01680000
[2] {led zeppelin, the doors} => {pink floyd} 0.01066667 0.5970149  0.01786667
    lift     count
[1] 4.180183 167
[2] 5.689469 160
```

Note que ambas reglas de asociación poseen relativamente altos valores de confianza y lift, los cuales, como mencionamos en la **parte a**, indican qué tan exitosa/apropiada sería la recomendación de artista(s) (consecuente), en base al itemset antecedente.

En ese sentido, concluimos que, respecto a la base de datos, a usuarios que escuchan al par de artistas **Oasis y The Killers**, se les recomendaría escuchar a la banda **Coldplay**; mientras que, a usuarios que escuchan al par de artistas **Led Zeppelin y The Doors**, se les recomendaría escuchar a la banda **Pink Floyd**.