Pontificia Universidad Católica Madre y Maestra

Campus de Santiago

Asignatura:

Inteligencia De Negocios

Profesora:

Lisibonny Beato

Trabajo:

Proyecto Final sobre modelo de minería de datos

Integrante:

Gabriel Cepeda Garcia

ID: 1014-1803

Santiago, 08 de Diciembre de 2022

La cafetería es un establecimiento en el cual ofrecen gran variedad de comida desde mentas y chicles hasta comida elaborada como sándwiches, hot dog, etc. Una de las problemáticas más vistas en las cafeterías es que los clientes no tienen variedad de compras, sino que usualmente tienen aquellos productos específicos por los cuales se deciden a la hora de hacer una compra. Por lo tanto, los otros productos se ven afectados ya que no tienen la misma demanda, y por ende dichos productos llevarían al desperdicio y pérdidas del negocio. De modo que, al aprovechar dichos productos que tienen una gran cantidad de demandas y hacer un combo con aquellos que no la tienen, se podría aprovechar e incrementar las ventas de dichos productos menos demandados.

Para satisfacer dicho objetivo, me centraré en utilizar un algoritmo de reglas de asociación, el cual se basa en encontrar relaciones dentro de un conjunto de transacciones, es decir , ítems que tienden a ocurrir de forma conjunta. El algoritmo a utilizar es el algoritmo Apriori, el cual consiste en identificar todos los ítems que ocurren con una frecuencia por encima de un determinado límite y luego convertir estos en reglas de asociación.

El dataset que contiene las transacciones de las compras realizadas en el kiosco está en un formato arff, que son archivos de texto ASCII que describen una lista de instancias con atributos comunes. Dicho dataset está compuesto por 148 filas que indican las cantidades de transacciones que existen, así como también 99 columnas, las cuales indican las distintas variables. Con la función “summary" nos indica un resumen del dataset. Entre ellos podemos ver aquellos atributos que más se repiten en las transacciones. Entre los mismos se encuentran el atributo estudiante, hombre, mujer, 50 o menos y 51 a 100. Estos 2 últimos hacen referencia a la cantidad en pesos dominicanos gastados en dichas transacciones.

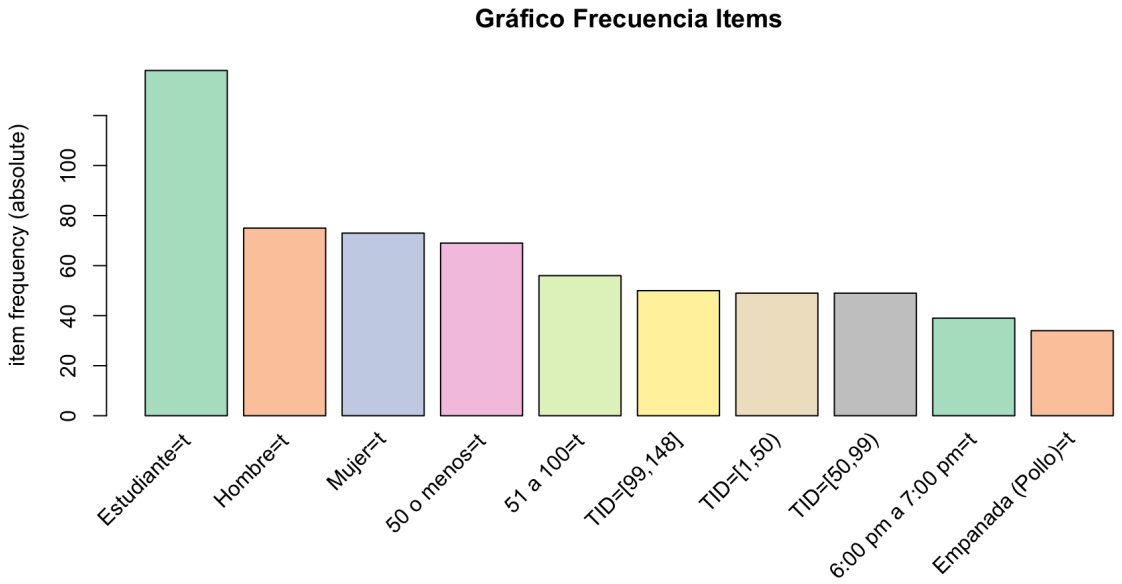
```
> summary(kiosco2)
transactions as itemMatrix in sparse format with
 148 rows (elements/itemsets/transactions) and
 99 columns (items) and a density of 0.07869233

most frequent items:
Estudiante=t     Hombre=t      Mujer=t 50 o menos=t   51 a 100=t      (Other)
         138           75           73           69           56          742

element (itemset/transaction) length distribution:
sizes
 6  7  8  9 10
 3 58 56 29  2

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  6.000   7.000   8.000   7.791   8.000  10.000

includes extended item information - examples:
        labels variables   levels
1   TID=[1,50)       TID   [1,50)
2  TID=[50,99)       TID  [50,99)
3 TID=[99,148]       TID [99,148]
```

Cuando visualizo las transacciones por individual con la función “inspect” me doy cuenta que tiene el problema de que el Tid se cuenta como atributo.

```
[145] {TID=[99,148],
       Agua(Cascada)=t,
       Hombre=t,
       Estudiante=t,
       ADM (Adm. Empresa)=t,
       3:00 pm a 4:00 pm=t,
       50 o menos=t}                                    145
[146] {TID=[99,148],
       Country Club Rojo=t,
       Mujer=t,
       Estudiante=t,
       ADM (Adm. Empresa)=t,
       3:00 pm a 4:00 pm=t,
       50 o menos=t}                                    146
[147] {TID=[99,148],
       Coca Cola=t,
       Hombre=t,
       Estudiante=t,
       ISC (Ingenier\357\277\275_a Sistema)=t,
       7:00 pm a 8:00 pm=t,
       50 o menos=t}                                    147
[148] {TID=[99,148],
       Coca Cola=t,
       Hombre=t,
       Estudiante=t,
       ISC (Ingenier\357\277\275_a Sistema)=t,
       7:00 pm a 8:00 pm=t,
       50 o menos=t}                                    148
```

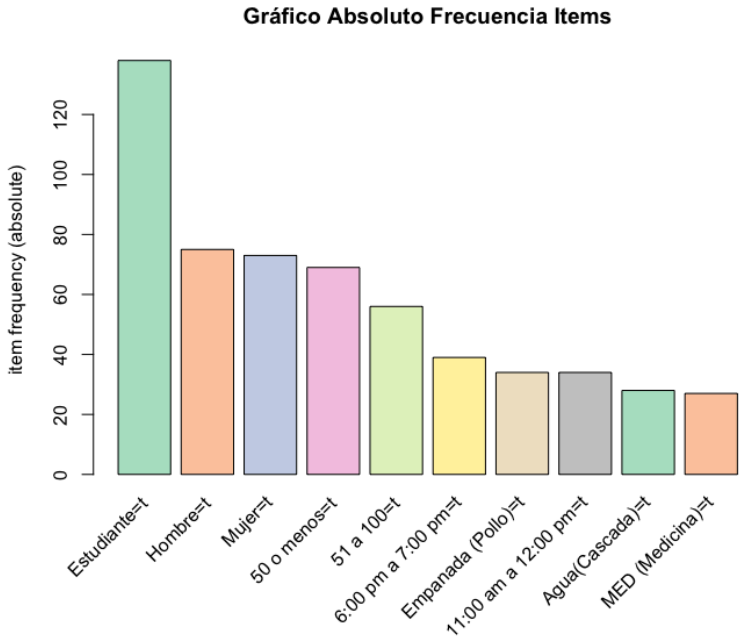
Por lo tanto, con esta función puedo darme cuenta de esos valores los cuales pueden afectar a que mi modelo no sea un buen modelo.

Graficando con la función “ItemFrecuencyPlot”se ve gráficamente lo que planteé teóricamente anteriormente. Los atributos de los id de las transacciones se muestran como ítems frecuentes, por lo que para mi objetivo no me ayuda.



Así que para mi modelo tuve que hacer transformación de los datos y eliminar aquellos atributos de los cuales afectan a la construcción de un buen modelo.

Luego de ya modificado el dataset podemos ver que ahora sí tenemos un dataset interesante, el cual le podríamos sacar provecho para nuestro fin.



Para desarrollar el modelo se aplicará el algoritmo de Apriori, el cual para lograr un buen modelo, es recomendable indicarle algunos parámetros de los cuales él va a tomar como referencia para escoger las reglas. Entre estos parámetros está el soporte, confianza, minlen y maxlen.

El soporte del ítem X es el número de transacciones que contienen X dividido entre el total de transacciones. Para nuestro caso nos interesa aquellos ítems que tengan un mínimo de venta mayor que 4 por lo menos. Entonces para esto aplicamos la fórmula **soporte = 15 / 148 = 0.1**

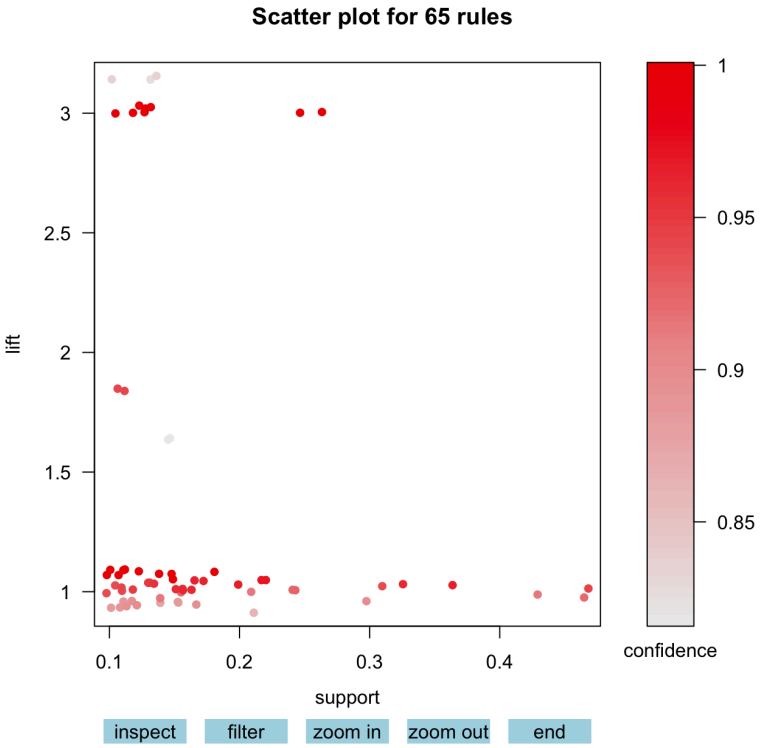
La confianza se interpreta como la probabilidad de que una transacción que contiene los ítems de X, también contenga los ítems de Y. Para nuestro caso nos interesa una confianza de mínimo de 0.8 o 80%.

Y por último, minLen y maxLen son los que indican la cantidad mínima y máxima de ítems que contiene una regla de asociación. Para nuestro caso, la mínima sería de 2 ítems y la máxima de 4.

Luego de ejecutada la función Apriori nos indica que, obtenemos un total de 65 reglas de asociación. Para ver el gráfico y las reglas de manera interactiva, usé la función plot la cual tiene uno de sus parámetros (engine) y se le pasa el string **interactive**.

Dando click 2 veces en un área y pulsando las opciones que se encuentran debajo, en la terminal nos salen las reglas incluidas en tal área. El área seleccionada se sombrea.

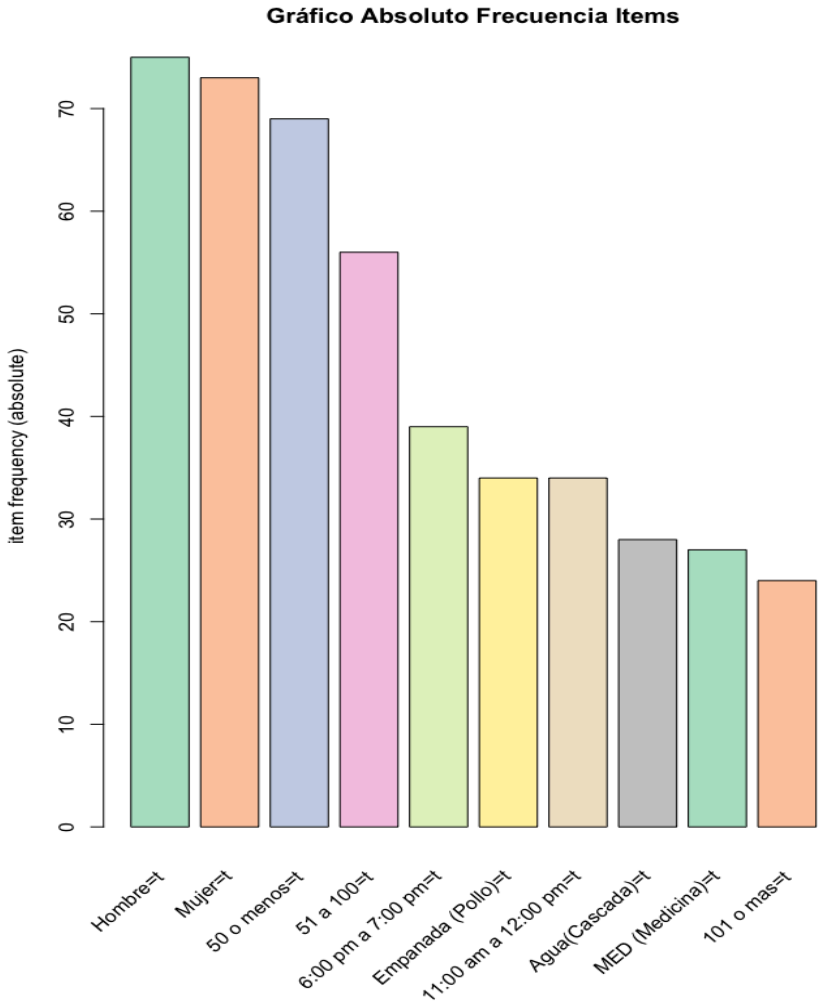
Con este gráfico se puede ir observando el comportamiento de cada regla y así como también la relación entre confianza, lift y soporte. Algo interesante que pude observar es que hay reglas con un lift muy alto. Esto quiere decir que aunque la regla sea un patrón en las transacciones, estas tienen un soporte bajo, lo cual nos indica que los ítems no tuvieron muchas compras con relación al total de las transacciones.




```
Select a region with two clicks!

Number of rules selected: 20
     lhs                                                rhs            support   confidence coverage  lift      count order id
[1]  {TID=[99,148], 51 a 100=t}                      => {Estudiante=t} 0.1351351 0.9523810  0.1418919 1.0213941 20    3     52
[2]  {Mujer=t, 6:00 pm a 7:00 pm=t}                  => {Estudiante=t} 0.1283784 0.9500000  0.1351351 1.0188406 19    3     43
[3]  {TID=[50,99), Mujer=t, 6:00 pm a 7:00 pm=t}     => {Estudiante=t} 0.1283784 0.9500000  0.1351351 1.0188406 19    4     60
[4]  {Empanada (Pollo)=t, 51 a 100=t}                => {Estudiante=t} 0.1216216 0.9473684  0.1283784 1.0160183 18    3     33
[5]  {Empanada (Pizza)=t}                            => {Estudiante=t} 0.1081081 0.9411765  0.1148649 1.0093777 16    2      5
[6]  {11:00 am a 12:00 pm=t, 50 o menos=t}           => {Estudiante=t} 0.1081081 0.9411765  0.1148649 1.0093777 16    3     31
[7]  {Empanada (Pollo)=t, Hombre=t}                  => {Estudiante=t} 0.1013514 0.9375000  0.1081081 1.0054348 15    3     35
[8]  {TID=[1,50), Mujer=t, 50 o menos=t}             => {Estudiante=t} 0.1013514 0.9375000  0.1081081 1.0054348 15    4     65
[9]  {TID=[50,99), Hombre=t}                         => {Estudiante=t} 0.1554054 0.9200000  0.1689189 0.9866667 23    3     51
[10] {TID=[1,50), Mujer=t}                           => {Estudiante=t} 0.1418919 0.9130435  0.1554054 0.9792060 21    3     46
[11] {Agua(Cascada)=t, 50 o menos=t}                 => {Estudiante=t} 0.1216216 0.9000000  0.1351351 0.9652174 18    3     28
[12] {Hombre=t, 6:00 pm a 7:00 pm=t}                 => {Estudiante=t} 0.1148649 0.8947368  0.1283784 0.9595728 17    3     44
[13] {TID=[50,99), Hombre=t, 6:00 pm a 7:00 pm=t}    => {Estudiante=t} 0.1148649 0.8947368  0.1283784 0.9595728 17    4     63
[14] {Agua(Cascada)=t}                               => {Estudiante=t} 0.1689189 0.8928571  0.1891892 0.9575569 25    2     11
[15] {Empanada (Pollo)=t, Mujer=t}                   => {Estudiante=t} 0.1081081 0.8888889  0.1216216 0.9533011 16    3     34
[16] {TID=[50,99), 50 o menos=t}                     => {Estudiante=t} 0.1081081 0.8888889  0.1216216 0.9533011 16    3     49
[17] {TID=[1,50), 50 o menos=t}                      => {Estudiante=t} 0.1554054 0.8846154  0.1756757 0.9487179 23    3     45
[18] {TID=[1,50), Hombre=t}                          => {Estudiante=t} 0.1554054 0.8846154  0.1756757 0.9487179 23    3     47
[19] {GFA (Gestion Financiera)=t}                    => {Estudiante=t} 0.1013514 0.8823529  0.1148649 0.9462916 15    2      4
[20] {101 o mas=t}                                   => {Estudiante=t} 0.1418919 0.8750000  0.1621622 0.9384058 21    2     10
```

Entonces para mejorar el modelo sería bueno excluir la variable que en este caso se repite en todas las transacciones, que es **Estudiante.**

Realizando nuevamente el proceso de transformación obtenemos el nuevo dataset sin la variable “Estudiante”. Realizamos los pasos anteriores y obtenemos un gráfico absoluto de frecuencia en el cual se puede observar diferencias con relación al anterior.



Nueva vez aplicando el algoritmo Apriori para realizar el modelo, con los mismos parámetros que elegimos anteriormente, nos indica que se han realizado un total de 2 reglas a diferencia que en el anterior que nos dio un total de 65 reglas.

```
> modelo <- apriori(kiosco, parameter = list(supp=0.1, conf=0.8,minlen=2,maxlen=4))
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen maxlen target  ext
        0.8    0.1    1 none FALSE            TRUE       5     0.1      2      4  rules TRUE

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

Absolute minimum support count: 14

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[95 item(s), 148 transaction(s)] done [0.00s].
sorting and recoding items ... [16 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 done [0.00s].
writing ... [2 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

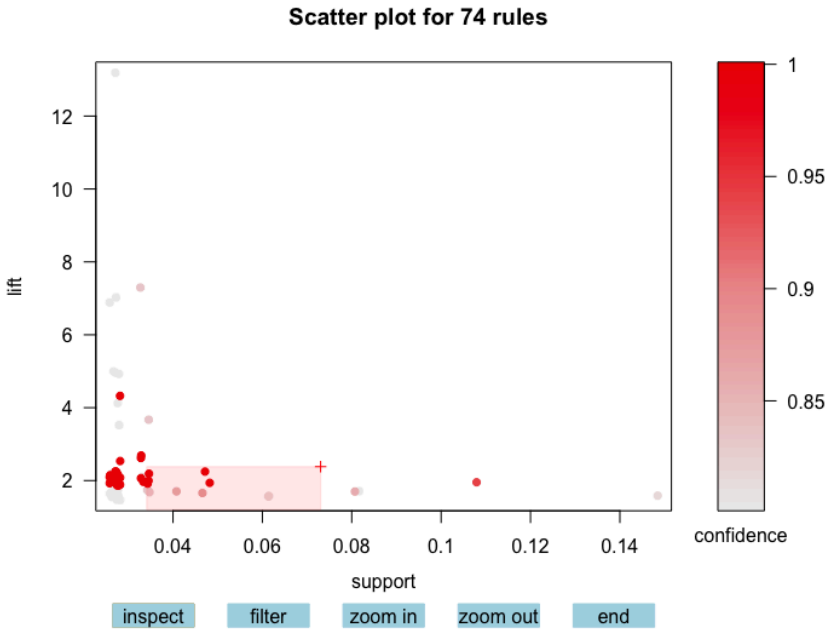
Lo cual con la función “Inspect” observamos que las 2 reglas generadas por el algoritmo no son importantes para nuestro objetivo, ya que nuestro principal objetivo incluye a los productos y en estas reglas no se ven involucrados los mismos. Por lo que, para resolver esto sería bajarle el parámetro de soporte al algoritmo.

```
> inspect(modelo)
    lhs                                rhs        support   confidence coverage  lift     count
[1] {ISC (Ingenier?_a Sistema)=t} => {Hombre=t} 0.1081081 0.9411765  0.1148649 1.857255 16
[2] {MED (Medicina)=t}            => {Mujer=t}  0.1486486 0.8148148  0.1824324 1.651953 22
>
```

El resultado del algoritmo con la modificación del soporte a un **0.025** entonces nos da un resultado de 74 reglas.

```
> modelo <- apriori(kiosco, parameter = list(supp=0.025, conf=0.8,minlen=2,maxlen=4))
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen maxlen target  ext
        0.8    0.1    1 none FALSE            TRUE       5   0.025      2      4  rules TRUE

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

Absolute minimum support count: 3

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[95 item(s), 148 transaction(s)] done [0.00s].
sorting and recoding items ... [48 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 done [0.00s].
writing ... [74 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

Graficando el modelo de manera interactiva podemos observar las diferentes reglas seleccionadas de manera interactiva:




```
Interactive mode.
Select a region with two clicks!

Number of rules selected: 6
    lhs                                          rhs            support    confidence coverage   lift     count order id
[1] {IIS (Ingenier?_a Industrial)=t}          => {50 o menos=t} 0.04729730 1.0000000  0.04729730 2.144928 7     2      9
[2] {Agua(Cascada)=t, MED (Medicina)=t}       => {Mujer=t}      0.04729730 1.0000000  0.04729730 2.027397 7     3     55
[3] {ISC (Ingenier?_a Sistema)=t, 51 a 100=t} => {Hombre=t}     0.04729730 0.8750000  0.05405405 1.726667 7     3     45
[4] {Empanada (Pizza)=t, 6:00 pm a 7:00 pm=t} => {Mujer=t}      0.04054054 0.8571429  0.04729730 1.737769 6     3     49
[5] {Jugo de Carton Rica peq=t}               => {Mujer=t}      0.06081081 0.8181818  0.07432432 1.658780 9     2     12
[6] {6:00 pm a 7:00 pm=t, 101 o mas=t}        => {Mujer=t}      0.06081081 0.8181818  0.07432432 1.658780 9     3     58
```

Aquí podemos observar como hay un mejor balance entre los ítems involucrados, ya que hay pocas reglas que se encuentran con valor y = 1, el cual indica el lift, y más reglas con lift mayor a 1, el cual indica que cuanto más se aleje el valor de lift de 1, más evidencias de que la regla no se debe a un artefacto aleatorio, es decir, mayor la evidencia de que la regla representa un patrón real.

A veces, queremos ver las reglas representadas por un solo ítem en específico, para encontrar cuáles productos causan la compra de otro producto X. En la función Apriori encontramos un parámetro el cual se llama **appearance** que nos da la opción de indicarle cuáles ítems queremos que estén del lado izquierdo o derecho de la regla. Estas opciones son LHS y RHS.

Para nuestro caso algo interesante sería poder observar cuáles productos son los más comprados cuando como consecuencia se compra el agua, pues para esto ejecutamos la función con los parámetros adecuados.

```
modelo_find <- apriori(kiosco, parameter = list(supp=0.025, conf=0.6,minlen=2),
                       appearance = list(default="lhs",rhs="Agua(Cascada)=t"))
```

Y así podemos observar 4 reglas que se cumplen con una confianza de más del 65%

```
> inspect(modelo_find)
    lhs                                          rhs                 support    confidence coverage   lift     count
[1] {Hombre=t, 3:00 pm a 4:00 pm=t}              => {Agua(Cascada)=t} 0.02702703 0.6666667  0.04054054 3.523810 4
[2] {12:00 pm a 1:00 pm=t, 50 o menos=t}         => {Agua(Cascada)=t} 0.03378378 0.6250000  0.05405405 3.303571 5
[3] {Hombre=t, 3:00 pm a 4:00 pm=t, 50 o menos=t} => {Agua(Cascada)=t} 0.02702703 0.8000000  0.03378378 4.228571 4
[4] {Mujer=t, 12:00 pm a 1:00 pm=t, 50 o menos=t} => {Agua(Cascada)=t} 0.02702703 0.6666667  0.04054054 3.523810 4
```

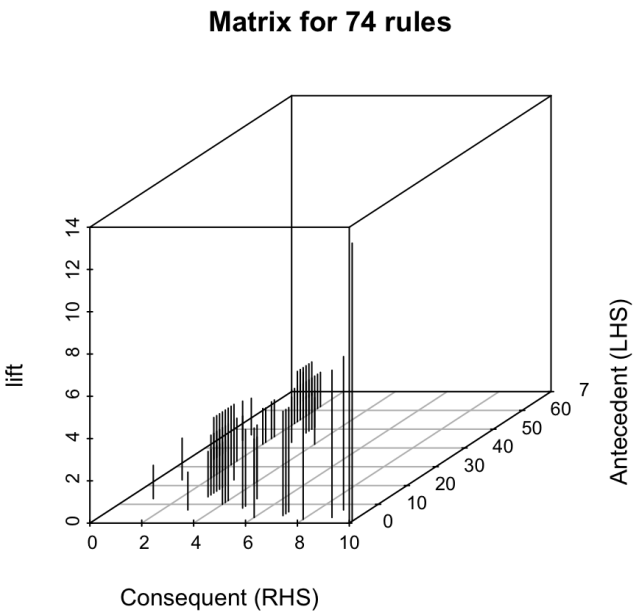
Así podemos seguir observando para ver otras reglas, como lo es aquellos ítems que se compran cuando el total hace de 51 a 100 pesos.

Aunque el soporte es bajo, la confianza es alta y el lift es mayor que 1, por lo que se puede decir que las reglas son fuertes.

```
modelo_find <- apriori(kiosco, parameter = list(supp=0.025, conf=0.8,minlen=2),
                       appearance = list(default="lhs",rhs="51 a 100=t"))
```

```
    lhs                                          rhs          support    confidence coverage   lift     count
[1] {Jugo Natural guayaba fresa=t}               => {51 a 100=t} 0.03378378 1.0000000  0.03378378 2.642857 5
[2] {Jugo Natural chinola=t}                     => {51 a 100=t} 0.03378378 0.8333333  0.04054054 2.202381 5
[3] {Jugo Natural chinola=t, Mujer=t}            => {51 a 100=t} 0.02702703 0.8000000  0.03378378 2.114286 4
[4] {Empanada (Pizza)=t, Iced Tea de limon=t}    => {51 a 100=t} 0.02702703 1.0000000  0.02702703 2.642857 4
[5] {Hombre=t, MCT (Mercadotecnia)=t}            => {51 a 100=t} 0.02702703 0.8000000  0.03378378 2.114286 4
[6] {Empanada (Pollo)=t, DER (Derecho)=t}        => {51 a 100=t} 0.03378378 1.0000000  0.03378378 2.642857 5
```

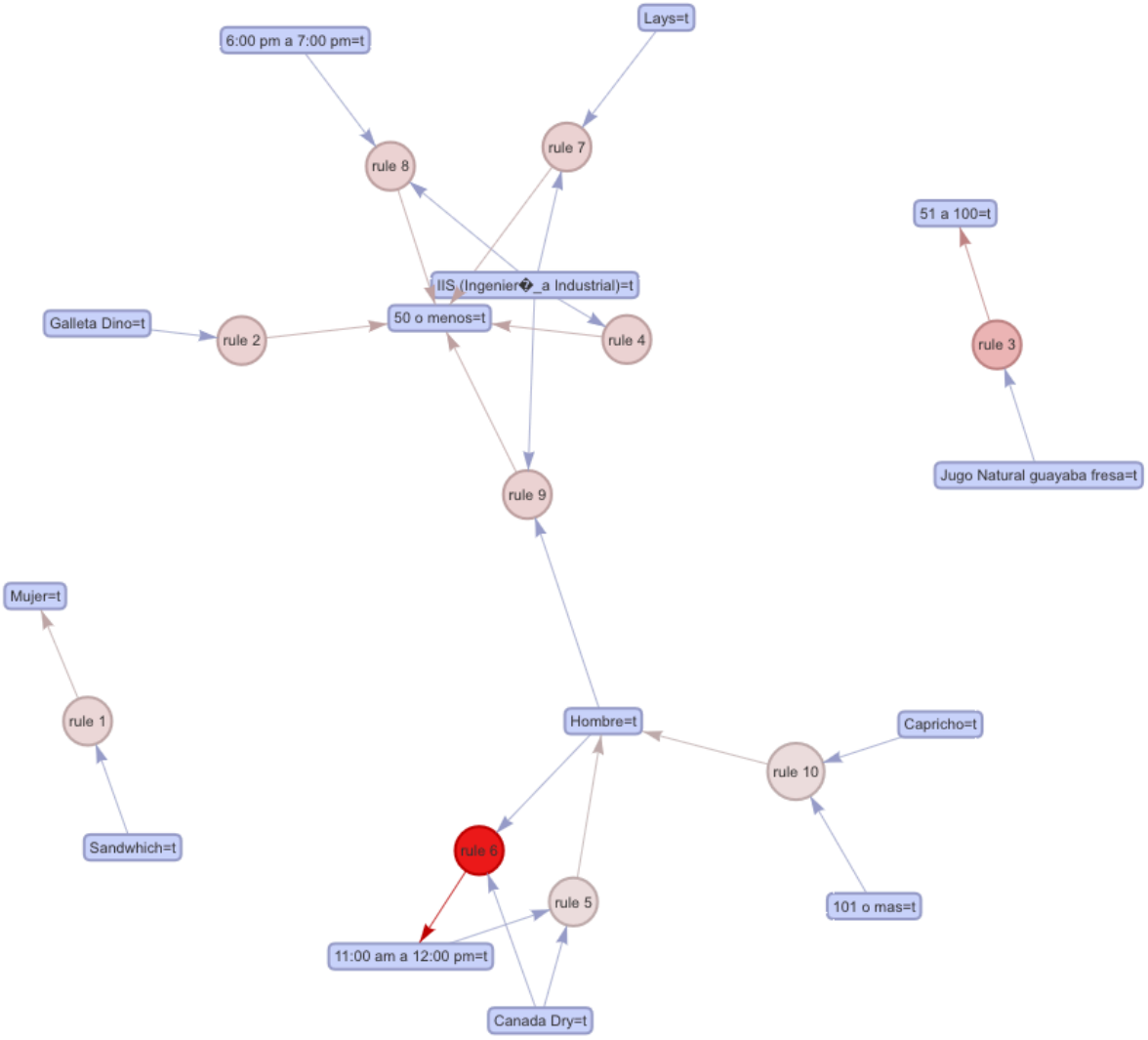
También podemos observar un gráfico 3D de las reglas, el cual gráficamente podemos observar qué tan fuertes son las reglas. Ya que según vemos el antecedente y el consecuente tienen un lift mayor que 1, por lo tanto, son reglas que cumplen un patrón.



Otro gráfico interesante que podemos hacer, es un grafo el cual utilizando vértices y aristas nos indican las reglas. Los vértices se etiquetan con nombres de elementos y las reglas se representan como un segundo conjunto de vértices. Los elementos se conectan con conjuntos de reglas mediante flechas dirigidas. Las flechas que apuntan desde los elementos a los vértices de las reglas indican los elementos LHS y una flecha desde una regla a un elemento indica el RHS.

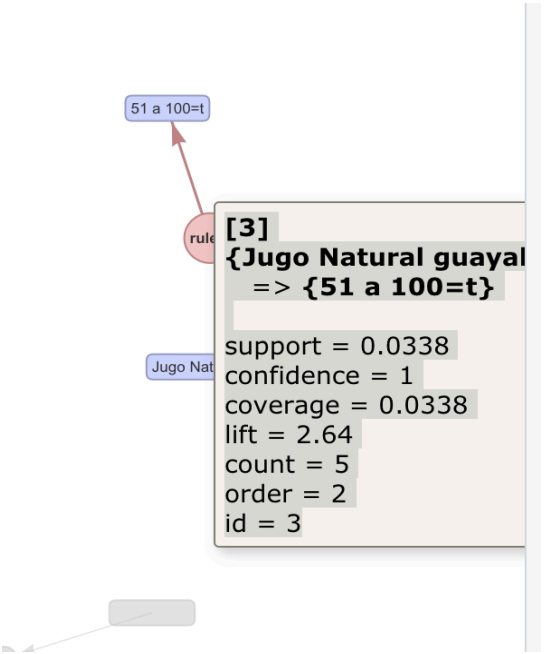
El único problema es que cuando hay muchas reglas este gráfico se ve muy cargado, por lo que vamos a obtener las 10 reglas con mayor confianza y graficar este resultado mediante grafo.

Este puede ser utilizado con la función “plot”, cambiando el parámetro “method” a **graph.**



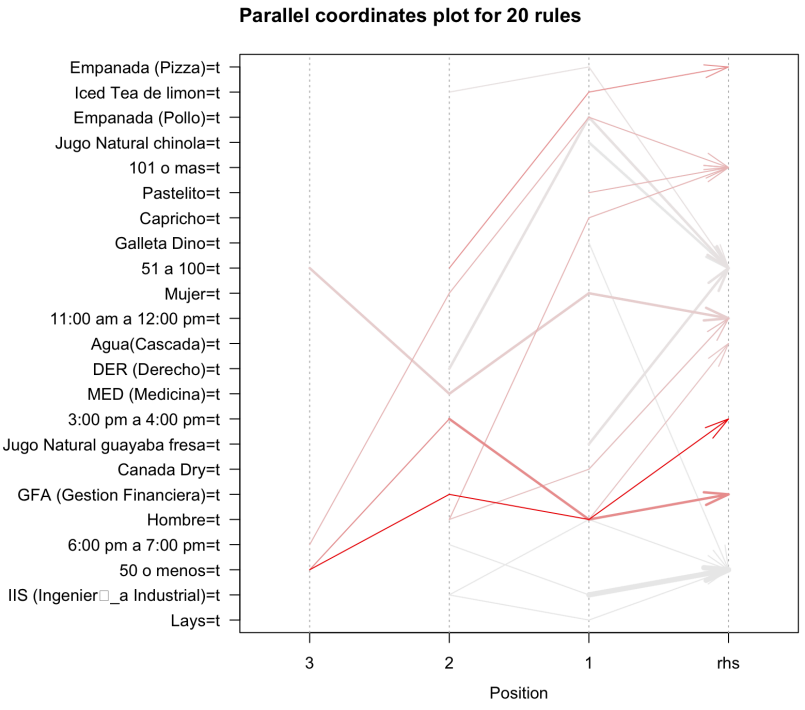
Este se lee por ejemplo… la regla 3 la cual nos indica que si se compra un jugo natural de guayaba fresa, se consume entre 51 y 100 pesos. Para ver más detalles de esta regla podemos realizar el gráfico interactivo y colocar el click sobre la regla.

Como podemos observar, la regla tiene una confianza de 100%, así como también un lift mayor que 1, es decir es una regla que cumple un patrón el 100% de las veces.



Otro tipo de gráfico que nos puede brindar más información sobre los ítems por individual, es el gráfico de coordenadas paralelas, el cual muestra los antecedentes y sus consecuentes.

Como podemos observar, la flecha nos indica que si por ejemplo compramos un ice tea y una empanada, es posible que gastemos de 51 a 100 pesos. Todo esto viéndolo de manera gráfica, fácil y entendible.



En conclusión, con este modelo el dueño del negocio puede evaluar cuáles productos y/o variables puede mezclar en forma de combo para incrementar las ventas de aquellos productos que no tienen una venta tan significante, aprovechando esos productos que sí tienen una venta significativa cuando se compran en conjuntos. Así como también, logrando resolver una de las problemáticas más frecuentes en los negocios de tipo cafetería, donde existe una gran cantidad de variedad.

Bibliografía

Coder, R. (2021, 18 noviembre). Plot in R. R CODER. *https://r-coder.com/plot-r/*

Kumar, K. (s.f.). *Visualize Market Basket analysis in R DataScience*. https://datascienceplus.com/visualize-market-basket-analysis-in-r/

Market Basket Analysis using R. (2018, agosto). DataCamp. https://www.datacamp.com/tutorial/market-basket-analysis-r