Reglas de asociación

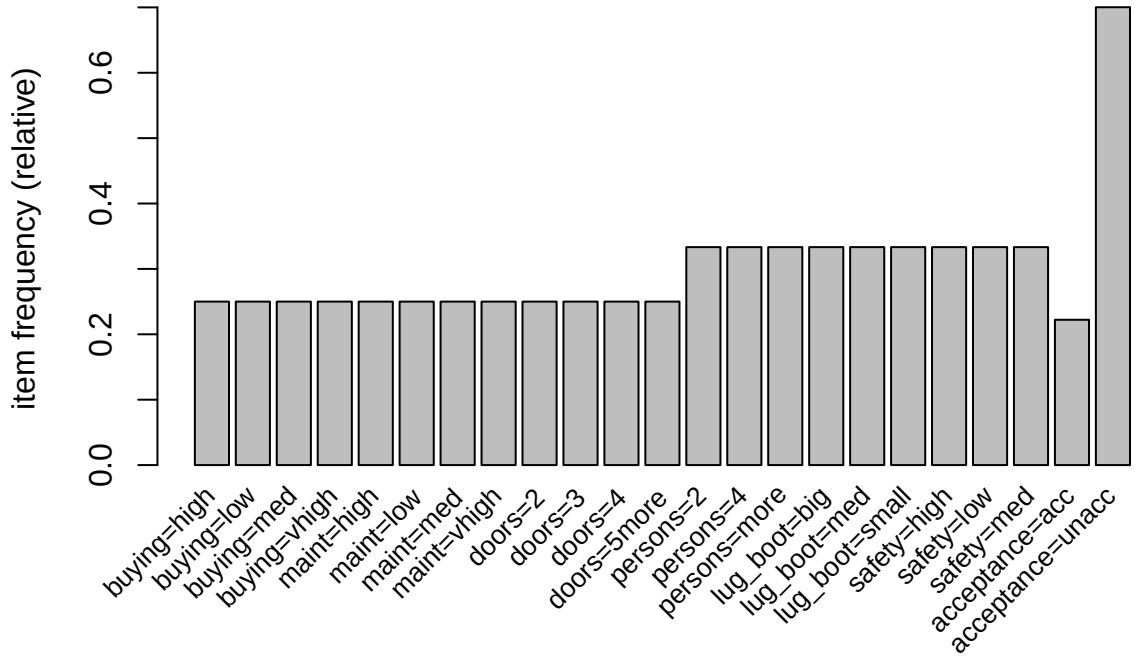
Este conjunto de datos se obtuvo del repositorio de aprendizaje automático de UCI (https://archive.ics.uci.edu/ml/datasets/car+evaluation).

El dataset contiene 1728 observaciones en las siguientes 7 variables, donde cada fila contiene información sobre un automóvil. Todas las variables son variables categóricas.

buying: Precio del coche (Levels: high, low, med ,vhigh) maint: Precio de mantenimiento (Levels: high, low, med, vhigh) doors: Número de puertas (Levels: 2, 3, 4, 5more) persons: Número de persona (Levels: 2, 4, more) lug_boot: Tamaño del maleteo (Levels: big, med, small) safety: Seguridad del coche (Levels: high, low, med) acceptance: Acceptación del coche (Levels: acc, good, unacc, vgood)

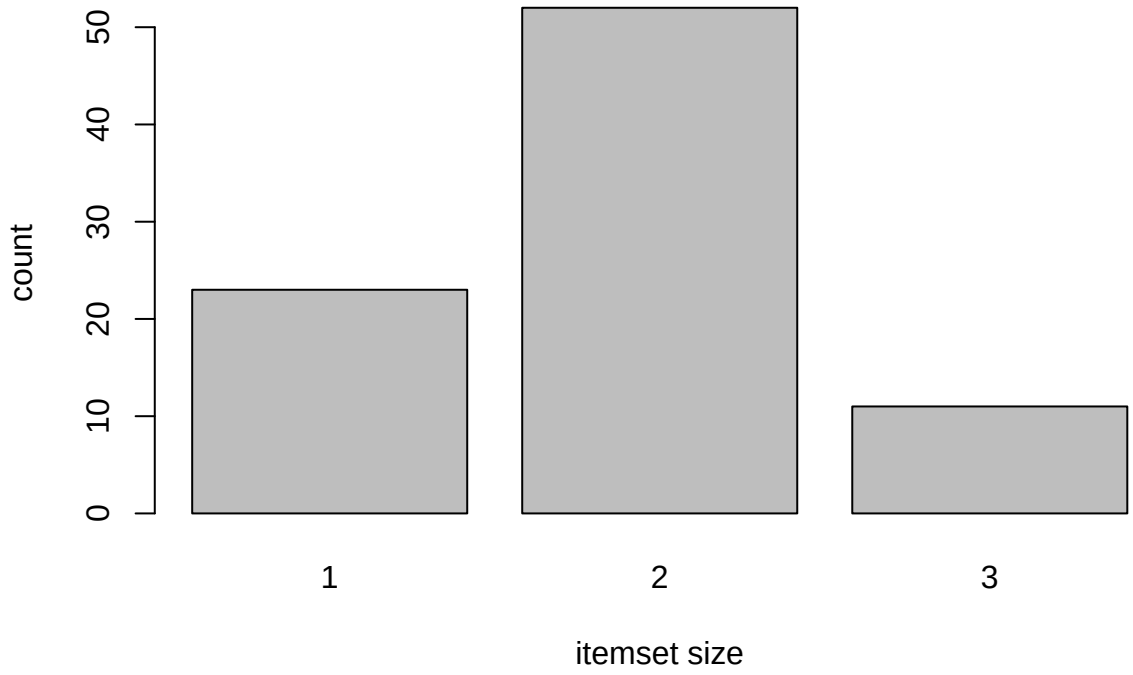
```
## 'data.frame':    1728 obs. of  7 variables:
##  $ buying    : Factor w/ 4 levels "high","low","med",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ maint     : Factor w/ 4 levels "high","low","med",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ doors     : Factor w/ 4 levels "2","3","4","5more": 1 1 1 1 1 1 1 1 1 1 ...
##  $ persons   : Factor w/ 3 levels "2","4","more": 1 1 1 1 1 1 1 1 1 2 ...
##  $ lug_boot  : Factor w/ 3 levels "big","med","small": 3 3 3 2 2 2 1 1 1 3 ...
##  $ safety    : Factor w/ 3 levels "high","low","med": 2 3 1 2 3 1 2 3 1 2 ...
##  $ acceptance: Factor w/ 4 levels "acc","good","unacc",..: 3 3 3 3 3 3 3 3 3 3 ...
```

En el siguiente gráfico se muestran los items con una frecuencia de más de 0.1 (soporte > 0.1). En este gráfico podemos decidir qué items estudiar más en profundidad. Para obtener reglas con valor, podemos analizar reglas en las que aparezcan items no muy frecuentes. Como se muestra, hay algunos items bastante frecuentes, como son por ejemplo "acceptance=unacc". Sin embargo, estudiar los items opuestos puede resultar bastante más interesante.



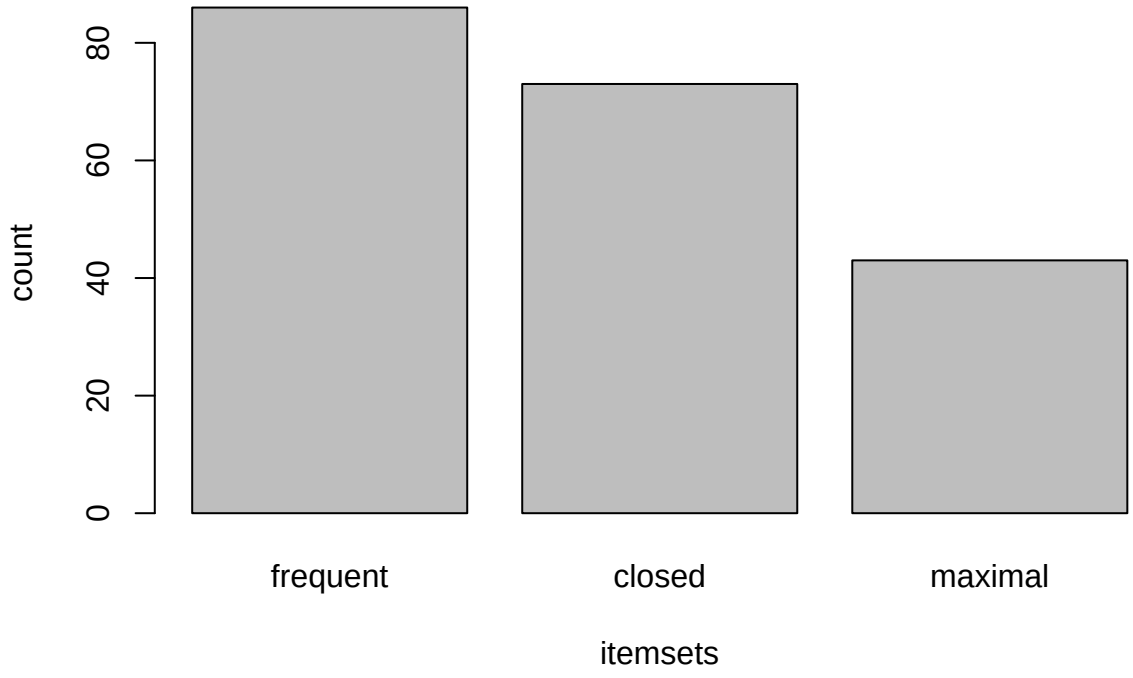
A continuación usamos apriori con un soporte mínimo de 0.1 para obtener los itemsets frecuentes.Primera información importante que acceptance=unacc es muy frecuente. En el barplot se muestran los tamaños de itemsets frecuentes. Se ve cómo el tamaño de itemsets más frecuente es 2.

En el siguiente gráfico vemos la distribución de los tamaños de los itemsets frecuentes. Vemos cómo el tamaño más común es 2 items por itemset. También se muestran los primeros 10 itemsets frecuentes.




```
##     items             support   count
## [1] {acceptance=acc} 0.2222222 384
## [2] {buying=med}     0.2500000 432
## [3] {buying=high}    0.2500000 432
## [4] {maint=low}      0.2500000 432
## [5] {buying=vhigh}   0.2500000 432
## [6] {maint=med}      0.2500000 432
```

Vamos a ver cuál es la cantidad de itemsets frecuentes, de itemsets cerrados y de itemsets maximales. Como se muestra en el siguiente gráfico, hay una gran diferencia entre la cantidad de itemsets frecuentes y los cerrados y maximales.

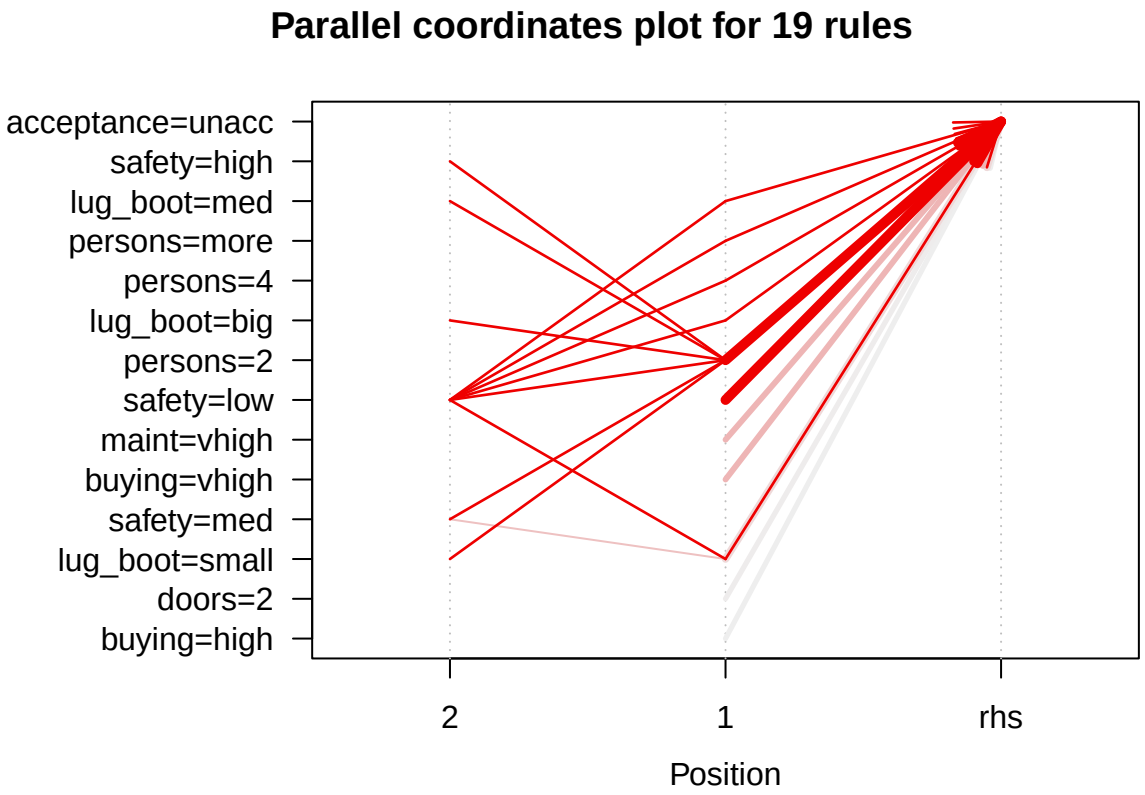


Reglas general

A continuación empezamos ya a aplicar apriori para la obtención de reglas. En esta ejecución indicaré que el mínimo soporte sea de 0.09 y la mínima confianza de 0.75. También indicaré el mínimo del tamaño de las reglas, que será 2.

Posteriormente tras cada filtrado según los items que quiera explorar, aplicaré el filtro de reglas redundantes. También muestro un gráfico en el que se muestran como puntos las distintas reglas generadas por nuestro método. Se muestra cómo las reglas con mejor Lift son por lo general con menor soporte que 0.3 y hay algunas reglas con un "confidence"" muy alto donde el valor es cercano a 1, sin embargo, estas reglas las consideraré como triviales siempre y cuando su soporte sea muy elevado, superior al 50%. Los lifts más "realistas" o con más equilibrio los encontramos en un nivel de confianza entre 1 y 1.5 aproximadamente. A continuación se muestran algunas reglas con más lift.

En las anteriores reglas se ve cómo la principal consecuencia que nos permite tener un lift bueno es acceptance=unacc. Las dos primeras reglas nos proporcionan gran cantidad de información, puesto que nos indican que en un 33% (soporte=0.33) de los casos de nuestro dataset la compra de dicho coche será inaceptable en caso de que la seguridad proporcionada por este sea baja o el número de pasajeros sea 2 (es el mínimo). Esto nos indica que, en general, según nuestro dataset aquellos coches que cumplan alguna o ambas de estas reglas serán, probablemente, inaceptables.



También podemos ver la reglas de forma visual a partir del gráfico anterior. Podemos ver las relaciones entre items (pares atributo-valor) siguiendo las flechas que los unen. Se puede observar como para cualquier item si este seta relacionado con los items 'safety=low' y 'persons=2' , la aceptación del coche sera inaceptable.

A partir de las reglas de asociación nos queda claro que las variables mas importantes para evaluar un coche son el numero de pasajeros y la seguridad. Si alguna de estas variables es "mala", entonces el coche no sera aceptable.

Reglas específicas mediante filtrado

Una vez ya tenemos todas las reglas, vamos a ver qué reglas nos pueden parecer interesantes.

```
##     lhs                            rhs                 support   confidence
## [1] {lug_boot=big,safety=low} => {acceptance=unacc} 0.1111111 1
## [2] {persons=2,lug_boot=big}  => {acceptance=unacc} 0.1111111 1
## [3] {persons=4,safety=low}    => {acceptance=unacc} 0.1111111 1
## [4] {persons=more,safety=low} => {acceptance=unacc} 0.1111111 1
## [5] {lug_boot=med,safety=low} => {acceptance=unacc} 0.1111111 1
## [6] {persons=2,lug_boot=med}  => {acceptance=unacc} 0.1111111 1
##     lift     count
## [1] 1.428099 192
## [2] 1.428099 192
## [3] 1.428099 192
## [4] 1.428099 192
## [5] 1.428099 192
## [6] 1.428099 192
```

El primer item a estudiar será "persons", es decir, ver qué pasa para los distintos valores del número de personas que caben en un vehículo.

```
rules.persons <- subset(car.rules, subset = lhs %in% c("persons=2", "persons=4"))
inspect(head(rules.persons))
```

```
##     lhs                            rhs                 support    confidence
## [1] {buying=med,persons=2}    => {acceptance=unacc} 0.08333333 1
## [2] {buying=high,persons=2}   => {acceptance=unacc} 0.08333333 1
## [3] {maint=low,persons=2}     => {acceptance=unacc} 0.08333333 1
## [4] {buying=vhigh,persons=2}  => {acceptance=unacc} 0.08333333 1
## [5] {maint=med,persons=2}     => {acceptance=unacc} 0.08333333 1
## [6] {maint=high,persons=2}    => {acceptance=unacc} 0.08333333 1
##     lift     count
## [1] 1.428099 144
## [2] 1.428099 144
## [3] 1.428099 144
## [4] 1.428099 144
## [5] 1.428099 144
## [6] 1.428099 144
```

De las anteriores reglas se pueden sacar algunas conclusiones y además a partir de reglas con un Lift bueno. Para mí la más significativa es que independientemente de la seguridad del coche, si el número de pasajeros no es superior a 2, el coche será inaceptable. En el caso de que tenga más de dos plazas pero la seguridad es baja, tampoco se considera aceptable el vehículo.

- {persons=4,safety=low} => {acceptance=unacc}
- {persons=2,safety=high} => {acceptance=unacc}
- {persons=2,safety=med} => {acceptance=unacc}

El siguiente item a estudiar será "safety", es decir, ver qué pasa para los distintos valores de seguridad de un vehículo. Solo me aparecen 6 reglas.

```
rules.safety <- subset(car.rules, subset = lhs %in% c("safety=low", "safety=med", "safety=high"))
inspect(head(rules.safety))
```

```
##     lhs                            rhs                 support    confidence
## [1] {buying=med,safety=low}   => {acceptance=unacc} 0.08333333 1
```

```
## [2] {buying=high,safety=low}  => {acceptance=unacc} 0.08333333 1
## [3] {maint=low,safety=low}    => {acceptance=unacc} 0.08333333 1
## [4] {buying=vhigh,safety=low} => {acceptance=unacc} 0.08333333 1
## [5] {maint=med,safety=low}    => {acceptance=unacc} 0.08333333 1
## [6] {maint=high,safety=low}   => {acceptance=unacc} 0.08333333 1
##     lift     count
## [1] 1.428099 144
## [2] 1.428099 144
## [3] 1.428099 144
## [4] 1.428099 144
## [5] 1.428099 144
## [6] 1.428099 144
```

Además del número de personas, la seguridad tiene un peso importante e a la hora de considerar un vehículo como aceptable. Independiente del resto de valores, si la seguridad es baja el usuario no considera aceptable un coche.

A continuación voy a probar a ver reglas con items negados. Si nos fijamos en las proporciones de acceptance, la mayoría de coches no han sido considerados aceptados,

```
##
##   acc  good unacc vgood
##   384    69  1210    65
```

Por lo tanto, puede ser muy interesante considerar las tres variables “acc”, “good”" y “vgood” como una sola. Así, se nos quedará un dataset con el mismo número de variables pero la variable acceptance solo posea dos valores, o aceptado o no. Los valores de “acc”, “good”" y “vgood” se juntarán dentro de acceptable.

```
##     lhs                           rhs                support   confidence
## [1] {lug_boot=med,safety=low} => {acceptance=unacc} 0.1111111 1
## [2] {persons=2,lug_boot=med}  => {acceptance=unacc} 0.1111111 1
## [3] {lug_boot=big,safety=low} => {acceptance=unacc} 0.1111111 1
## [4] {persons=2,lug_boot=big}  => {acceptance=unacc} 0.1111111 1
## [5] {persons=4,safety=low}    => {acceptance=unacc} 0.1111111 1
## [6] {persons=more,safety=low} => {acceptance=unacc} 0.1111111 1
##     lift     count
## [1] 1.428099 192
## [2] 1.428099 192
## [3] 1.428099 192
## [4] 1.428099 192
## [5] 1.428099 192
## [6] 1.428099 192
```

Viendo que no obtenemos reglas interesante, vamos a negar la variable “persons”. Se va a crear una nueva variable indicando si el coche es biplaza o no. Además haremos lo mismo con la variable “safety”. Se considera solo si el coche es inseguro o no.

```
##     lhs                 rhs                support   confidence lift
## [1] {biPlaza=1}      => {acceptance=unacc} 0.3333333 1.0000000  1.428099
## [2] {safetyLow=1}    => {acceptance=unacc} 0.3333333 1.0000000  1.428099
## [3] {acceptance=acc} => {biPlaza=0}        0.2997685 1.0000000  1.500000
## [4] {acceptance=acc} => {safetyLow=0}      0.2997685 1.0000000  1.500000
## [5] {lug_boot=small} => {acceptance=unacc} 0.2604167 0.7812500  1.115702
## [6] {buying=vhigh}   => {acceptance=unacc} 0.2083333 0.8333333  1.190083
## [7] {maint=vhigh}    => {acceptance=unacc} 0.2083333 0.8333333  1.190083
## [8] {doors=2}        => {acceptance=unacc} 0.1886574 0.7546296  1.077686
## [9] {buying=high}    => {acceptance=unacc} 0.1875000 0.7500000  1.071074
```

```
##     count
## [1] 576
## [2] 576
## [3] 518
## [4] 518
## [5] 450
## [6] 360
## [7] 360
## [8] 326
## [9] 324
```

No observamos ninguna regla destacada que no hayamos comentado anteriormente. Tan solo podemos, confirmar con certeza que el número de personas que caben en el coche es un factor relevante a la hora de considerarlo acceptable o no, con una medida de confianza muy buena y un buen valor de lift.

- {biplaza=1} => {acceptance=unacc}
- {acceptance=acc} => {biPlaza=0}

Conclusiones

Las reglas obtenidas, desde mi punto de vista, han sido un tanto decepcionantes. Sin embargo, pienso que no podía sacar muchas mejores. A partir de la aplicación de técnicas de extracción de reglas de asociación hemos podido ver que las variables mas importantes para evaluar un coche son el numero de pasajeros y la seguridad. Si alguna de estas variables tiene un valor bajo, entonces el coche no sera aceptable. En concreto, el número de personas es más relevante, ya que independientemente de la seguridad del coche si el vehículo es biplaza, no sera considerado adecuado el vehículo.