Minería de Datos: Aprendizaje no supervisado y detección de anomalías

Reglas de asociación

ugr | Universidad de Granada

Máster en Ciencia de Datos e Ingeniería de los Computadores

Besay Montesdeoca Santana Graduado en Ingeniería Informática

1. Descripción y preprocesamiento de la base de datos

a. Descripción

El dataset que se utilizará para aplicar las distintas técnicas pertenecientes al apartado de las Reglas de Asociación fue extraído del repositorio gratis UCI. Se intentó escoger una base de datos que recogiese información fácil de entender, es decir, se evitó escoger bases de datos donde se ha reunido información más compleja como por ejemplo la eritema, el fenómeno koebner, las pápulas foliculares como es en el caso del dataset *Dermatology*.
En concreto la base de datos que se ha escogido se llama *Student Performance,* cuyos datos abordan el logro estudiantil de la educación secundaria de dos escuelas portuguesas. Los atributos de los datos incluyen calificaciones de los estudiantes, características demográficas y sociales los cuales fueron recopilados mediante el uso de informes escolares y cuestionarios.

b. Descripción de los atributos

La base de datos en un principio consta de 33 atributos que se describen a continuación:

- **school**: Escuela del estudiante (binario: "GP" - Gabriel Pereira o "MS" - Mousinho da Silveira)
- **sex**: Sexo del estudiante (binario: "F" - femenino "M" - masculino)
- **age**: Edad del estudiante (numérica: de 15 a 22)
- **address**: Tipo de domicilio del estudiante (binario: "U" - urbano o "R" - rural)
- **famsize**: Tamaño de la familia (binario: "LE3" - menor o igual a 3 o "GT3" - mayor que 3)
- **Pstatus**: Estado de cohabitación de los padres (binario: "T" - viven juntos o "A" - aparte)
- **Medu**: Educación de la madre (numérica: 0 - ninguna, 1 - educación primaria (4º grado), 2 - 5º a 9º grado, 3 - educación secundaria o 4 - educación superior)
- **Fedu**: Educación del padre (numérico: 0 - ninguno, 1 - educación primaria (4º grado), 2 - 5º al 9º grado, 3 - educación secundaria o 4 - educación superior)
- **Mjob**: Trabajo de la madre (nominal: "profesor", "cuidado de salud" relacionado, civil "servicios" (por ejemplo, administrativo o de policía), "en_casa" u "otro"
- **Fjob**: Trabajo del padre (nominal: "profesor", "cuidado de salud" relacionado, civil "servicios" (por ejemplo, administrativo o policía), "en_casa" u "otro")
- **reason**: Razón de la elección de dicha escuela (nominal: cerca de "casa", "reputación", "curso" u "otro")
- **guardian**: Tutor del estudiante (nominal: "madre", "padre" o "otro")
- **traveltime**: Tiempo que tarda el alumno desde su casa a la escuela (numérico: 1 - <15 min, 2 - 15 a 30 min, 3 - 30 min a 1 hora, o 4 -> 1 hora)
- **studytime**: Tiempo de estudio semanal (numérico: 1 - <2 horas, 2 - 2 a 5 horas, 3 - 5 a 10 horas, o 4 -> 10 horas)
- **failures**: Número de suspensos de clases anteriores (numérico: n si 1 <= n <3, si no 4)
- **schoolsup**:Apoyo educativo extra (binario: sí o no)
- **famsup**: Apoyo educativo familiar (binario: sí o no)
- **paid**: Clases extra dentro del curso (binario: sí o no)
- **activities**: Actividades extra-curriculares (binario: sí o no)
- **nursery**: Asistió a la guardería (binario: sí o no)

- **higher**: Quiere cursar una educación superior (binario: sí o no)
- **internet**: Acceso a Internet en casa (binario: sí o no)
- **romantic**: Con una relación romántica (binario: sí o no)
- **famrel**: Calidad de las relaciones familiares (numérico: de 1 - muy malo a 5 - excelente)
- **freetime**: Tiempo libre después de la escuela (numérico: de 1 - muy bajo a 5 - muy alto)
- **goout**: Salida con amigos (numérico: de 1 - muy bajo a 5 - muy alto)
- **Dalc**: Consumo de alcohol durante el día de trabajo (numérico: de 1 - muy bajo a 5 - muy alto)
- **Walc**: Consumo de alcohol durante el fin de semana (numérico: de 1 - muy bajo a 5 - muy alto)
- **health**: situación de salud actual (numérica: de 1 - muy mala a 5 - muy buena)
- **absences**: Número de ausencias escolares (numérico: de 0 a 93)
- **G1** - Nota del primer periodo (numérica: De 0 a 20)
- **G2** - Nota del segundo periodo (numérica: De 0 a 20)
- **G3** - Nota final (numérica: De 0 a 20)

Para aplicar las distintas técnicas no se trabajará con la totalidad de los atributos, se descartarán aquellos que de primeras parece que no ofrecen información útil y relevante. Por ejemplo en este caso se eliminaron los atributos *school, failures, reason, paid, G1* y *G2* (se ha dejado *G3* puesto que representa el rendimiento final del alumno).

c. Creación de los intervalos y transformación de los atributos

Nuestra base de datos posee muchos ítems numéricos, y por lo tanto previamente debemos realizar dos tareas de preparación de los datos. La primera de ellas es la transformación de los datos de aquellos atributos que en un principio se muestran de forma numérica pero que realmente son categóricos, y por tanto debemos transformarlos a factores. Un ejemplo de estos atributos es *famrel*, en el que debemos transformar los valores 1 a “muy mala”, 2 a “mala”, 3 a “media”... etc.
En segundo lugar debemos tratar el caso de las variables que presentan distribuciones numéricas como por ejemplo la edad o las notas finales. Para estas es necesario establecer intervalos y transformarlas a la clase *ordered factor.*

```
# Construimos los intervalos
dat[["age"]] = ordered(cut(dat[["age"]], c(14, 17, 23)), labels = c("younger", "adult"))
dat[["Medu"]] = factor(dat[["Medu"]], levels = c(1, 2, 3, 4), labels =
c("primary-education", "5th-to-9th-grade", "secondary-education", "higher-education"))
dat[["Fedu"]] = factor(dat[["Fedu"]], levels = c(1, 2, 3, 4), labels =
c("primary-education", "5th-to-9th-grade", "secondary-education", "higher-education"))
dat[["traveltime"]] = factor(dat[["traveltime"]], levels = c(1, 2, 3, 4), labels = c("<15
min", "15-to-30-min", "30-min-to-1-hour", ">1-hour"))
dat[["studytime"]] = factor(dat[["studytime"]], levels = c(1, 2, 3, 4), labels = c("<2
hours", "2-to-5-hours", "5 to 10 hours", ">10 hours"))
dat[["famrel"]] = factor(dat[["famrel"]], levels = c(1, 2, 3, 4, 5), labels = c("very-bad",
"bad", "medium", "good", "excellent"))
dat[["freetime"]] = factor(dat[["freetime"]], levels = c(1, 2, 3, 4, 5), labels =
c("very-low", "low", "medium", "high", "very-high"))
```

```
dat[["goout"]] = factor(dat[["goout"]], levels = c(1, 2, 3, 4, 5), labels = c("very-low",
"low", "medium", "high", "very-high"))
dat[["Dalc"]] = factor(dat[["Dalc"]], levels = c(1, 2, 3, 4, 5), labels = c("very-low",
"low", "medium", "high", "very-high"))
dat[["Walc"]] = factor(dat[["Walc"]], levels = c(1, 2, 3, 4, 5), labels = c("very-low",
"low", "medium", "high", "very-high"))
dat[["health"]] = factor(dat[["health"]], levels = c(1, 2, 3, 4, 5), labels = c("very-bad",
"bad", "medium", "good", "very-good"))
dat[["absences"]] = ordered(cut(dat[["absences"]], c(-1, 0, 10, 20, 75)), labels =
c("never", "low", "enough", "many"))
dat[["G3"]] = ordered(cut(dat[["G3"]], c(-1, 9, 20)), labels = c("fail", "pass"))
```

d. Incorporación de los Ítems negativos

Seguidamente para realizar también un análisis por grupos de las reglas, se añadirá un solo item negativo para evitar una explosión de combinaciones al aplicar el método Apriori. Para añadir los ítems negativos básicamente tenemos que reemplazar la variable en cuestión por tantas variables nuevas como posibles valores podía tomar la variable original. En las variables nuevas los ejemplos tomarán como valor True si el ejemplo contiene el valor correspondiente y False en caso contrario.

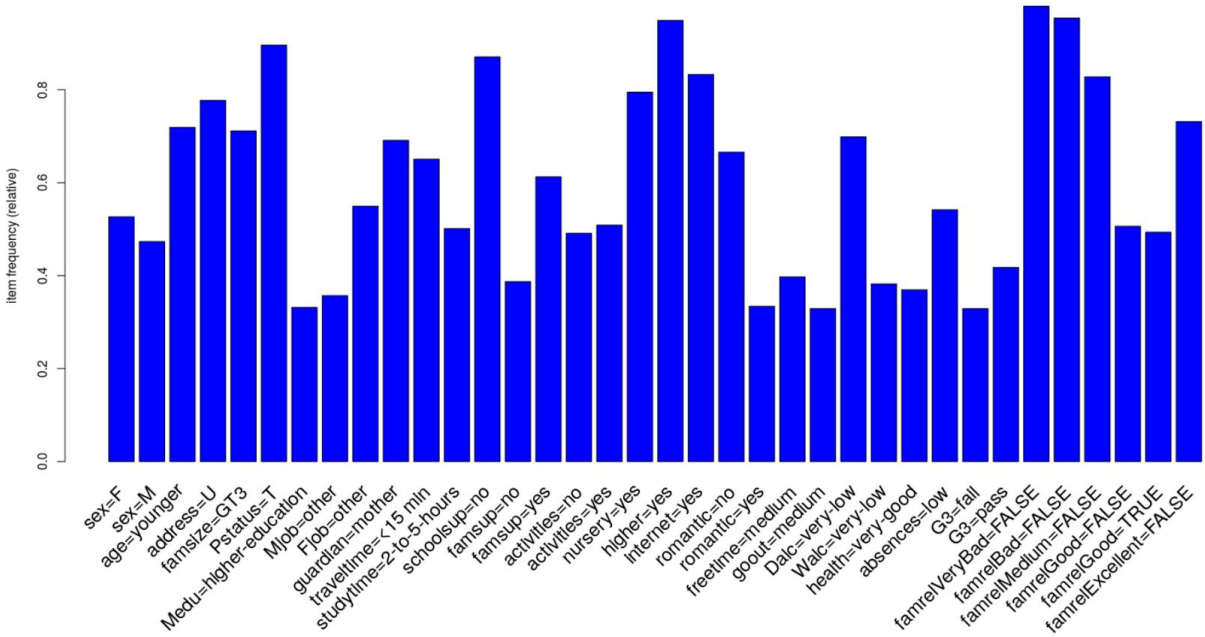
En este caso se ha escogido el atributo famrel(Calidad de las relaciones familiares) para realizar el análisis en grupo. Esta variable puede tomar valores de *very-bad, bad, medium, good y excellent*, por tanto la remplazamos por las 5 variables correspondientes en el dataset.

```
# Añadimos los items negados
replaceData = function(d, strg){
 if(d == strg){
   return("TRUE")
 }else{
   return("FALSE")
 }
}

famrelVeryBad = sapply(dat[["famrel"]], replaceData, "very-bad")
famrelBad = sapply(dat[["famrel"]], replaceData, "bad")
famrelMedium = sapply(dat[["famrel"]], replaceData, "medium")
famrelGood = sapply(dat[["famrel"]], replaceData, "good")
famrelExcellent = sapply(dat[["famrel"]], replaceData, "excellent")
dat[["famrel"]] = NULL
dat = cbind(dat, famrelVeryBad, famrelBad, famrelMedium, famrelGood, famrelExcellent)
```

2. Extracción de Reglas

Se ha realizado una primera ejecución con la base de datos tal y como se ha explicado en los apartados anteriores. En esta primera ejecución parece ser que tenemos varios itemset muy frecuentes:



Observamos que los itemsets *Pstatus=T, schoolsup=no, higher=yes, famrelVeryBad=FALSE, famrelBad=FALSE,* entre otras poseen soportes muy altos, en torno a 0.95, lo que ocasionarán que aparecerán en muchas de las reglas que extraeremos.

Efectivamente después de generar todas las reglas ejecutando el algoritmo de Apriori con un soporte de 0.5 y una confianza de 0.8, y después de eliminar las reglas redundantes, obtenemos unas 227 reglas donde en la mayoría aparecen los items destacados anteriormente.

rules	support	confidence	lift
{internet=yes,romantic=no} => {famrelVeryBad=FALSE}	0.53	0.99	1.011
{nursery=yes,romantic=no} => {famrelVeryBad=FALSE}	0.51	0.99	1.010
{age=younger,romantic=no} => {famrelVeryBad=FALSE}	0.50	0.99	1.010
{romantic=no} => {famrelVeryBad=FALSE}	0.65	0.98	1.009
{internet=yes,Dalc=very-low} => {famrelVeryBad=FALSE}	0.56	0.98	1.007
{address=U,nursery=yes,internet=yes} => {famrelVeryBad=FALSE}	0.53	0.98	1.006
{traveltime=<15 min,nursery=yes} => {famrelVeryBad=FALSE}	0.51	0.98	1.005
{age=younger,address=U,Pstatus=T} => {famrelVeryBad=FALSE}	0.50	0.98	1.005

En la tabla anterior observamos las ocho reglas con mayor confianza donde podemos apreciar también que en todas aparecen alguno de los itemset en cuestión. Evidentemente esto es algo normal, la estrategia a seguir sería analizar qué ocurre con los casos contrarios a estos itemset tan frecuentes. Por ejemplo investigar el caso de Pstatus=A y por lo tanto analizar qué ocurre con los estudiantes cuyos padres están divorciados o viven separados. Para ello lo que se ha hecho es cortar la base de datos y quedarnos con aquellas instancias que posean el atributo que estamos estudiando, en este caso Pstatus=A.

```
dat = dat[dat,"Pstatus"] == "A",]
```

También debemos seleccionar, de todas las reglas generadas, las que tengan en el antecedente este atributo, ya que una regla donde el valor del atributo esté en el consecuente nos daría como resultado medidas de interés muy engañosas.

```
rulesSelected = subset(rulesSorted, subset = lhs %in% "Pstatus=A")
```

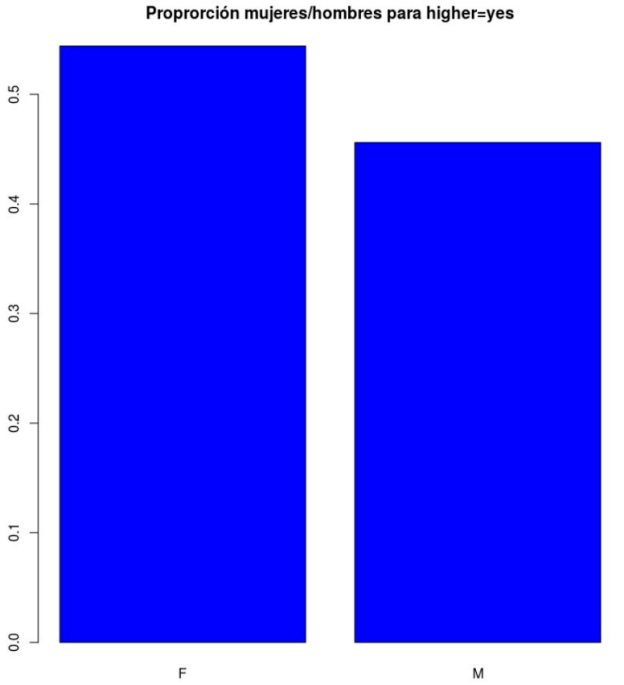
Lo mismo se hizo con el caso contrario(Pstatus=A) para observar si existen cambios significativos entre las reglas generadas.

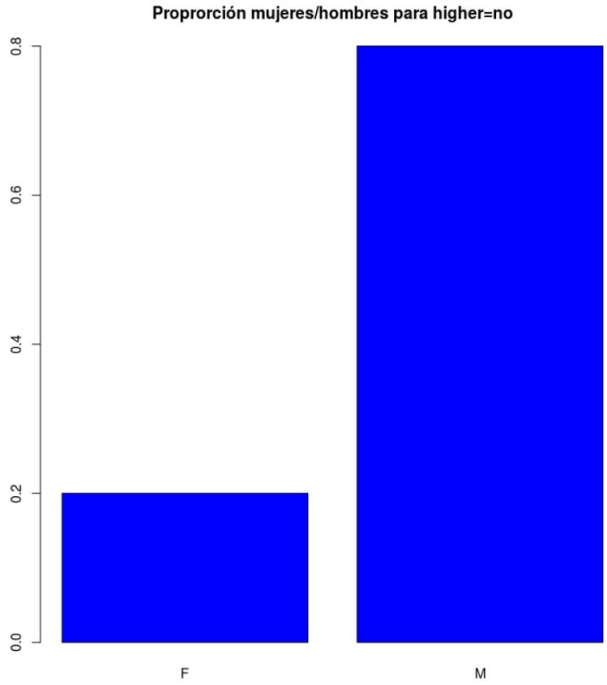
Un rápido análisis de ambos grupos de reglas pone de manifiesto que prácticamente no hay diferencias entre las reglas extraídas cortando la base de datos y quedándonos con las instancias del atributo Pstarus=A y las reglas generadas por el otro corte(Pstatus=T). En definitiva obtenemos las mismas reglas para ambos casos. La misma estrategia se siguió con los demás itemset superfrecuentes (internet=yes, nursery=yes, schoolsup=no, address=U, higher=yes), donde íbamos cortando la base de datos y examinando las distintos valores de cada atributo.

Al realizar el análisis con estos atributos no se obtuvo reglas significativas, es decir ocurrió lo mismo que en el caso del primer atributo que evaluamos(Pstatus) donde obtuvimos prácticamente el mismo grupo de reglas. Esto ocurrió para todos los itemsets superfrecuentes excepto para el itemset higher=yes (Quiere cursar educación superior) con el que sí se pudo extraer algunas reglas interesantes:

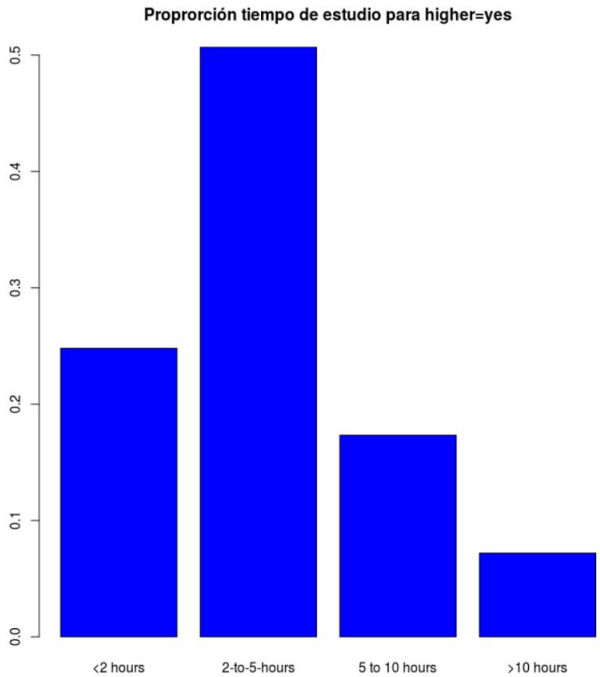
rules	support	confidence	lift
{higher=no} => {sex=M}	0.8	0.8	1
{higher=no} => {studytime=<2 hours}	0.6	0.6	1
{higher=yes} => {G3=pass}	0.688	0.688	1
{higher=no} => {G3=fail}	0.65	0.65	1
{higher=no} => {age=adult}	0.65	0.65	1
{higher=yes} => {age=younger}	0.738	0.73	1
{higher=yes} => {Dalc=very-low}	0.712	0.712	1

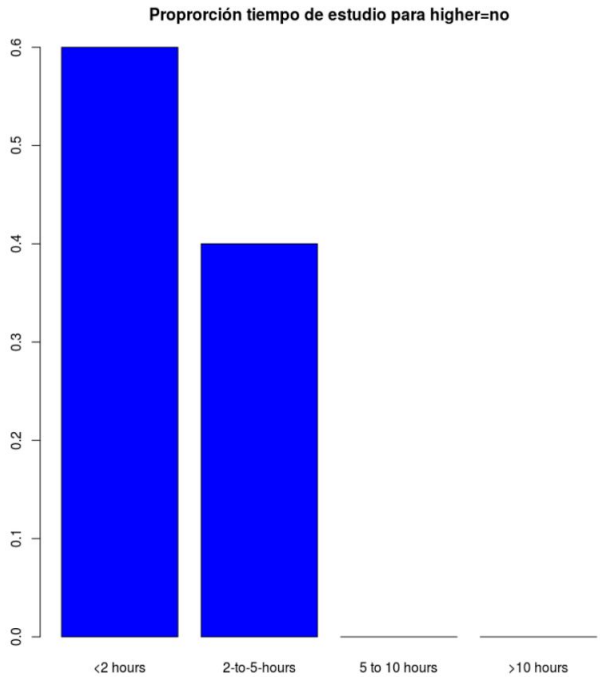
Evidentemente las medidas de soporte y de confianza están calculadas en función de la partición de la base de datos, en este caso en función del atributo *higher,* pero nos puede dar cierta información. Por ejemplo, nos dice que el 80% de los estudiantes que no cursarán una educación superior son hombres.



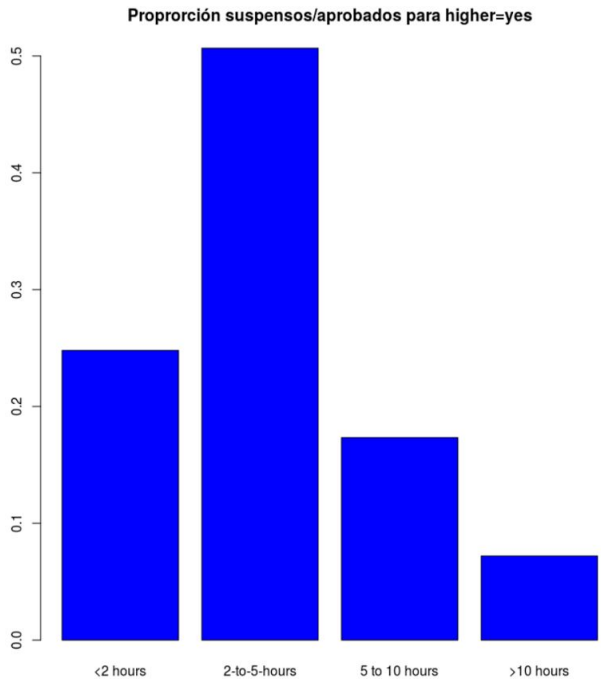


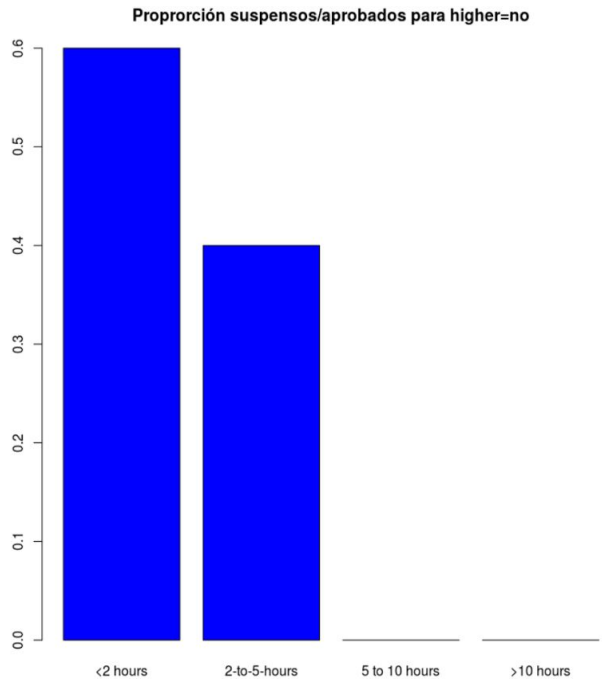
Otra información interesante es que un 62% de estos estudiantes invierten menos de 2 horas de estudio durante el curso.



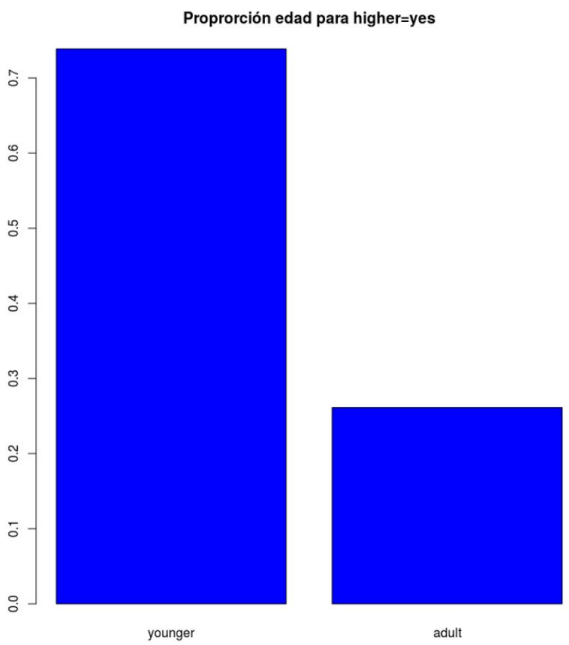


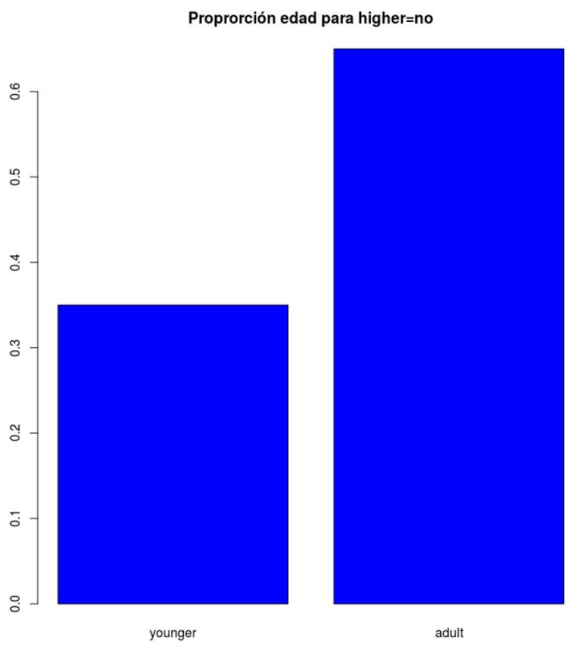
En relación al índice de aprobados y suspensos, parece ser que el 68% de los estudiantes que si cursarán una educación superior consiguen superar el curso, por otro lado el 65% de los que no cursarán ninguna educación superior obtienen suspensos.



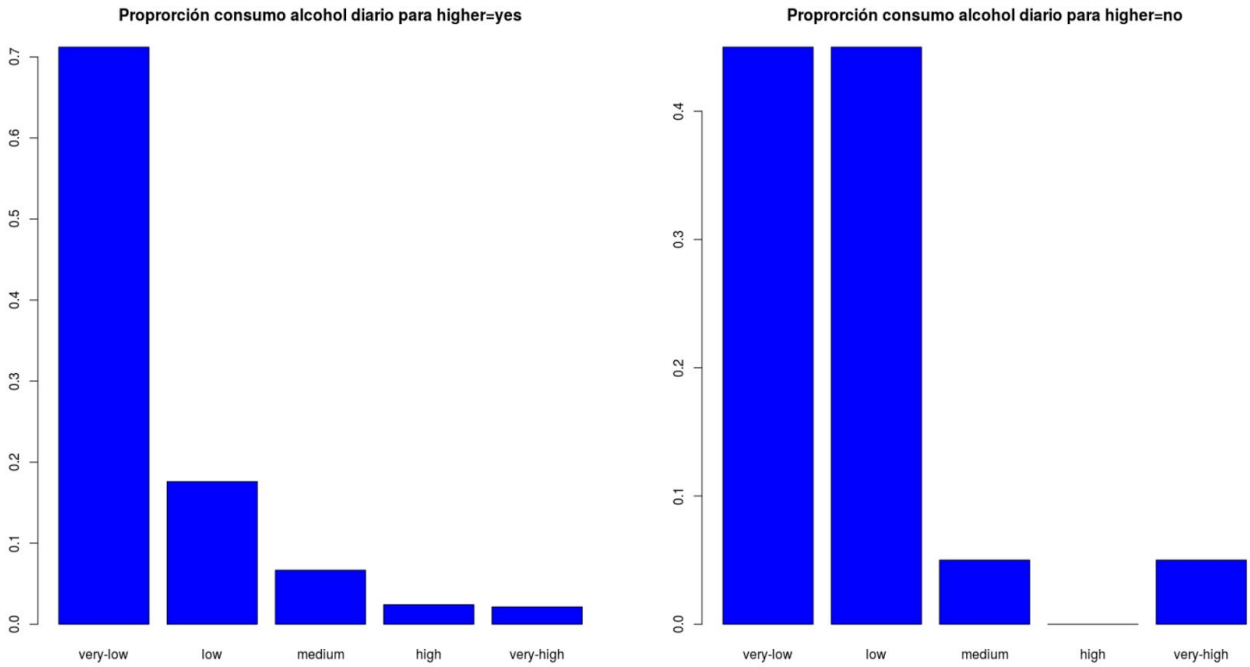


En cuanto a las edades de los alumnos también existen varias peculiaridades, según las reglas el 65% de los alumnos que no cursarán ninguna educación superior son mayores de edad, por otro lado el 73% del otro tipo de alumnos son menores de edad.





Por último obtuvimos la regla que nos dice que el 72% de los alumnos que si cursarán estudios superiores consumen una cantidad diaria de alcohol muy baja.



Una vez analizado algunos aspectos de los itemset más frecuentes, se procedió a realizar una nueva ejecución del algoritmo Apriori pero obviando de la base de datos los atributos con valores muy frecuentes con el objetivo de extraer otras reglas, que evidentemente serán menos frecuentes, pero que quizás son también interesantes.

rules	support	confidence	lift
{absences=low} => {G3=pass}	0.415	0.766	1.142
{sex=F} => {Dalc=very-low}	0.422	0.802	1.149
{famsup=yes} => {famsize=GT3}	0.460	0.752	1.057

Siguiendo esta estrategia obtenemos estas tres reglas que aportan cierta información interesante. La primera nos dice que cuando un alumno tiene pocas ausencias en clase, generalmente aprueba el curso, lo cual es algo bastante lógico. En segundo lugar obtenemos una regla que nos resalta que las chicas tienden a consumir menos alcohol diariamente en comparación con los chichos. Por último obtuvimos otra regla que nos dice si el alumno recibe apoyo educativo familiar probablemente posea una familia de más de 3 miembros.

3. Conclusiones

En un principio, cuando tienes una base de datos, no sabes en principio si de ella vas a poder extraer información útil o si las distintas técnicas de reglas de asociación rendirán eficientemente. La base de datos del problema que se estudió en esta práctica quedó patente que posee poca variabilidad en los datos de los estudiantes, además de que tiene muchas variables, algunas poco útiles, que en algunos casos provocaba una explosión de reglas que el algoritmo Apriori no soportaba y generaba un coste computacional demasiado elevado. A pesar de lo anterior, se ha conseguido extraer algunas reglas útiles que aportan cierta información interesante, como por ejemplo la diferencia de los hábitos y rendimiento escolar entre sexos o las distintas características de los alumnos con una visión más menos optimista de su futuro escolar.