Homework 6

Marc Mendez & Joel Cantero

5th May, 2019

First of all, we are going to install and load all the libraries we need for this exercise. The factominer library is needed for the catdes function, which is needed on exercise 2. Then the arules function is needed to the the association rules.
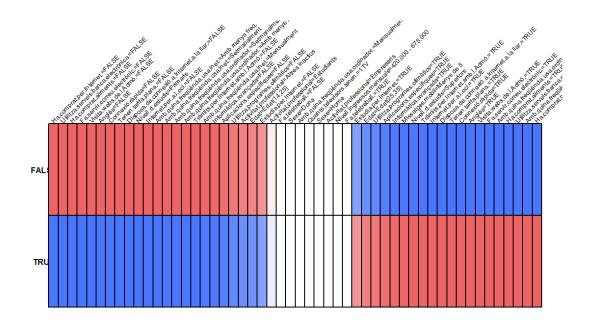
##1. Read the file: tic_tt.csv. Check the "class"" of every variable of tic_tt.

First of all we need to check the data and we can see that all the variables are booleans or factors with more than 2 levels. So, for loading the data, we will set the classes as factors and set the first one as row names because they are numbers that don't give information to the study. We will make a summary of it and analize all the results to see that the load has been succesfull.

```
tic <- read.csv("tic_tt.csv", sep=";", colClasses = c(rep("factor",33)), row.names=1)
summary(tic)
```

##2. Find the profile of the people who do payments by Internet (use the function catdes of FactoMineR). (to perform the catdes of a variable, it should be categorical (factor in the R nomenclature). To study the profile of the people who do payments by Internet we are going to execute the catdes function over the variable "Pagament.a.trav.s.d.Internet.". By doing this we can see the variables that have a realtion with this one.

```
a <- catdes(tic, 28)
plot(a)
```

As we can see in this plot the users that make payments through internet use electronic payments services and also they use internet every day. Most of them know English so there is a relation with the language also. We can see that, on the other hand, people that not pay through internet tend to not have electronic payment services as well as few usage of e-mail. And also they don't speak English, which makes sense as english is the most common language on the computers world. We can notice that there are differences between users that do payments on internet and people that not.

##3. Convert the tic_tt file to a transactions file. To do this convestion we will use as() funtion. This conversion is to make the association rules after.To do this conversion we needed to delete the first column as it wasn't a factor and have all the other columns as factors.

```
tic.trans <- as(tic, "transactions")
tic.trans


itemFrequency(tic.trans)
```

##4. Define the parameters: Min_support, min_confidence and maximum size of itemsets, and run the apriori function. Now that we have the transaction file we are going to execute it with the function apriori from the package "arules". To execute this we set the min support, min confidence and maximum size of the rules. The more precise we make the variables the larger number of rules we are going to obtain and longer time will last to calculate it.

```
minSupp <- 0.3
minConf <- 0.75
maxSize <- 5
rules <- apriori (tic.trans, parameter = list (support=minSupp, confidence=minConf, maxlen = ma
xSize))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##        0.75    0.1    1 none FALSE            TRUE       5     0.3      1
##  maxlen target    ext
##       5  rules FALSE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE   FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 334
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[94 item(s), 1116 transaction(s)] done [0.00s].
## sorting and recoding items ... [44 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 4 5
```

```
## Warning in apriori(tic.trans, parameter = list(support = minSupp,
## confidence = minConf, : Mining stopped (maxlen reached). Only patterns up
## to a length of 5 returned!
```

```
##  done [0.06s].
## writing ... [94589 rule(s)] done [0.02s].
## creating S4 object  ... done [0.02s].
```

```
inspect(rules[1:10])
```

```
##       lhs    rhs                                     support   confidence
## [1]  {}  => {Fa.servir.xat.=FALSE}                  0.7616487 0.7616487
## [2]  {}  => {Utilitza.programes.ofimàtics=TRUE}     0.7894265 0.7894265
## [3]  {}  => {Castellà=TRUE}                          0.7921147 0.7921147
## [4]  {}  => {Ha.comprat.per.Internet.=FALSE}         0.8100358 0.8100358
## [5]  {}  => {Tràmits.per.Internet.amb.l.Admó.=FALSE} 0.8127240 0.8127240
## [6]  {}  => {Connexió.ràpida=FALSE}                  0.8189964 0.8189964
## [7]  {}  => {Visita.webs.sanitaris.=FALSE}           0.8351254 0.8351254
## [8]  {}  => {Informàtica.avançada=FALSE}             0.8467742 0.8467742
## [9]  {}  => {Pagament.a.través.d.Internet.=FALSE}    0.8629032 0.8629032
## [10] {}  => {Estudia.per.Internet.=FALSE}            0.9301075 0.9301075
##       lift count
## [1]  1     850
## [2]  1     881
## [3]  1     884
## [4]  1     904
## [5]  1     907
## [6]  1     914
## [7]  1     932
## [8]  1     945
## [9]  1     963
## [10] 1    1038
```

```
summary(rules)
```

```
## set of 94589 rules
##
## rule length distribution (lhs + rhs):sizes
##     1     2     3     4     5
##    13   501  5191 25134 63750
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   4.000   5.000   4.608   5.000   5.000
##
## summary of quality measures:
##     support          confidence          lift            count
##  Min.   :0.3002   Min.   :0.7500   Min.   :0.8711   Min.   : 335.0
##  1st Qu.:0.3271   1st Qu.:0.8161   1st Qu.:0.9991   1st Qu.: 365.0
##  Median :0.3656   Median :0.8820   Median :1.0152   Median : 408.0
##  Mean   :0.3898   Mean   :0.8873   Mean   :1.0295   Mean   : 435.1
##  3rd Qu.:0.4283   3rd Qu.:0.9696   3rd Qu.:1.0410   3rd Qu.: 478.0
##  Max.   :0.9785   Max.   :1.0000   Max.   :1.6340   Max.   :1092.0
##
## mining info:
##       data ntransactions support confidence
##  tic.trans          1116     0.3        0.75
```

We have set this parameters as the next questions doesn't require to have large amount of solutions and this ones are more than enough for it.

##5. List the 10 most frequent itemsets. To obtain the list with the most frequent itemsets we need to filter the repeated itemsets.

```
fsets <- unique(generatingItemsets(rules))
fsets.df <- as(fsets, "data.frame")
sor.fsets <- fsets.df[order(-fsets.df$support),]
sor.fsets[1:10,1]
```

As we can see on this list, the most repeated itemsets are those ones above. ##6. List the first 10 rules sorted by the lift. To make this sorting we need to order by lift.

```
top10.lift <- inspect(sort(rules, by="lift")[1:10])
```

##7. List the 10 rules according the lift, where the Consequent is "Pagament.a.traves.d.Internet.". On this part we first need to create a subset with only the rules that have "Pagament.a.traves.d.Internet." as the Consequent. Then, once the subset is created we proceed on doing the same as the last exercise, order by lift and thats all.

```
rulesConseq <- subset(rules, subset = rhs %in% c("Pagament.a.través.d.Internet.=FALSE") )

top10.lift.Conseq <- inspect(sort(rulesConseq, by="lift")[1:10] )
```

```
##      lhs                                rhs                                     support
confidence    lift count
## [1]  {Ha.comprat.per.Internet.=FALSE}     => {Pagament.a.través.d.Internet.=FALSE} 0.8100358
1 1.158879   904
## [2]  {Ha.comprat.per.Internet.=FALSE,
##       Visita.webs.de.l.Admó.=FALSE}      => {Pagament.a.través.d.Internet.=FALSE} 0.3091398
1 1.158879   345
## [3]  {Nivell.ingressos.mensuals=NS/NC,
##       Ha.comprat.per.Internet.=FALSE}    => {Pagament.a.través.d.Internet.=FALSE} 0.3315412
1 1.158879   370
## [4]  {Català=FALSE,
##       Ha.comprat.per.Internet.=FALSE}    => {Pagament.a.través.d.Internet.=FALSE} 0.3315412
1 1.158879   370
## [5]  {Sexe=Home,
##       Ha.comprat.per.Internet.=FALSE}    => {Pagament.a.través.d.Internet.=FALSE} 0.3306452
1 1.158879   369
## [6]  {Quants.televisors.tenen.=2TVs,
##       Ha.comprat.per.Internet.=FALSE}    => {Pagament.a.través.d.Internet.=FALSE} 0.3548387
1 1.158879   396
## [7]  {Nivell.d.estudis=Secundaris,
##       Ha.comprat.per.Internet.=FALSE}    => {Pagament.a.través.d.Internet.=FALSE} 0.3673835
1 1.158879   410
## [8]  {Per.oci=FALSE,
##       Ha.comprat.per.Internet.=FALSE}    => {Pagament.a.través.d.Internet.=FALSE} 0.4112903
1 1.158879   459
## [9]  {Per.oci=TRUE,
##       Ha.comprat.per.Internet.=FALSE}    => {Pagament.a.través.d.Internet.=FALSE} 0.3987455
1 1.158879   445
## [10] {Activitat.professional=Assalariats,
##       Ha.comprat.per.Internet.=FALSE}    => {Pagament.a.través.d.Internet.=FALSE} 0.4489247
1 1.158879   501
```