IST 707 Applied Machine Learning

HW3: Association Rules

For this homework, we are going to explore the *bank-data.csv* data set. An accompanying description of the attributes and their values is contained on the next page. The dataset contains attributes on each person's demographics and banking information, to be used to determine if they will want to obtain the new PEP (Personal Equity Plan).

Your goal is to perform *Association Rule* discovery on the dataset using R or Python. *Note: Code created for this assignment must be part of your final submission, submitted as a separate file.*

First, perform the necessary preprocessing steps required for association rule mining, specifically the id field needs to be removed and certain numeric fields need to be discretized or otherwise converted to nominal.

Next, perform association rule (AR) discovery on the preprocessed data. Experiment with different parameters and preprocessing so that you get on the order of 20-30 strong rules, e.g., rules with high lift and confidence which at the same time have relatively good support. *Don't forget to report in detail what you have tried during the AR discovery process.*

Finally, set *pep (YES)* as the right-hand side (RHS) of the rules, and see what rules are generated.

Select the top 5 most "interesting" rules and for each specify the following for each:

- Support, Confidence and Lift values
- An explanation of the pattern and why you believe it is interesting based on the business objectives of the company.
- Any recommendations based on the discovered rule that might help the company to better understand behavior of its customers or to develop a business opportunity.

Note that the top 5 most interesting rules are most likely not the top 5 in the strong rules from earlier. These are rules, that in addition to having high lift and confidence, also provide some *non-trivial, actionable knowledge* based on underlying business objectives.

To complete this assignment, write a short report describing your association rule mining process and the resulting 5 interesting rules, each with their three items of explanation and recommendations.

You should write the report as if you are working for a client who knows little about techniques. Your report should give your client some insightful and reliable suggestions on what kinds of potential buyers your client should contact and convince your client that your suggestions are reliable based on the evidence gathered from your experiment results.

In more detail, your report should include:

- Description of preprocessing steps (high level, code file will contain details)
- Description of parameters and experiments to obtain strong rules
- The top 5 most interesting rules and the 3 items (support, confidence, and lift) listed and explained for each rule

Provide your code in a separate script.

Bank Data Description

The marketing department of a large bank keeps records on customers, including demographic information and number and type of accounts. When launching a new product, such as a "Personal Equity Plan" (PEP), a direct mail piece advertising the product is sent to existing customers, and a record kept as to whether that customer responded and bought the product.

id	Unique identification number
age	Age of customer in years
sex	MALE / FEMALE
region	INNER_CITY/RURAL/SUBURBAN/TOWN
income	Income of customer
married	Is the customer married (YES/NO)
children	Number of children
car	Does the customer own a car (YES/NO)
save_acct	Does the customer have a saving account (YES/NO)
current_acct	Does the customer have a current account (YES/NO)
mortgage	Does the customer have a mortgage (YES/NO)
pep	Did the customer buy a PEP after the last mailing (YES/NO)

Preprocessing data:

```r
library(dplyr)

bank_data.records = read.delim('C:/Users/harsh/OneDrive/Documents/IST 707  AML/Lecture3/bank-data.csv',header=TRUE, sep=",") |>
select( id=id, age, sex, region,income,married,children,save_act,current_act,mortgage,pep)

glimpse(bank_data.records)

bank_data.transactions = bank_data.records |>
  # remove the id variable since this is unique to each customer
  select(-id) |>

  # convert the numeric variable Age into discrete categories #from class code
  mutate(age = case_when(
    age >= 70 ~ 'elderly',
    age >= 60 ~ 'sixties',
    age >= 50 ~ 'fifties',
    age >= 40 ~ 'forties',
    age >= 30 ~ 'thirties',
    age >= 20 ~ 'twenties',
    age >= 10 ~ 'teens',
    TRUE ~ 'child'
  )) |>

  mutate(children = case_when
         (children == 1 ~ 'one_child',
           children ==2 ~'two_children',
           children ==3 ~ 'three_children',
           children ==0 ~'No_children')) |>
   mutate(income=case_when
    (income>=40000 ~ 'hi_income',
      income>=20000 ~ 'mid_income',
      income>=5000 ~ 'less_income'
         )) |>
  mutate_all(as.factor) # convert everything to a factor
```

58:2 Chunk 5 R Markdown

Console

- Remove id column as every id is different
- Convert numeric columns to nominal/categorical values
- Everything to factor

Finding Strong rules with ARM by changing support and confidence:

```r
Rule1 <- apriori(data=bank_data.transactions, parameter = list(support=0.15, confidence= 0.66))
inspect(Rule1)
#257 rules
```

R Console | data.frame 1 x 12 | data.frame 1 x 7 | data.frame 257 x 8

```
Apriori

Parameter specification:

Algorithmic control:

Absolute minimum support count: 90

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[29 item(s), 600 transaction(s)] done [0.00s].
sorting and recoding items ... [26 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 5 done [0.00s].
writing ... [257 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

Here my support is 0.15 and confidence 0.66 I was able to get 257 rules.

```r
Rule2 <- apriori(data=bank_data.transactions, parameter = list(support=0.25, confidence= 0.5))
inspect(Rule2)
#101 rules
```


```
Apriori

Parameter specification:

Algorithmic control:

Absolute minimum support count: 150

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[29 item(s), 600 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 4 done [0.00s].
writing ... [101 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

Therefore I tried with different support and confidence in order to find less than 20 rules: Here my support is 0.25 and confidence 0.5, I was able to get 101 rules.

Tweaking my support and confidence again:

```r
Rule3 <- apriori(data=bank_data.transactions, parameter = list(support=0.31, confidence= 0.68))
inspect(Rule3)
#25 rules
```


```
Apriori

Parameter specification:

Algorithmic control:

Absolute minimum support count: 186

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[29 item(s), 600 transaction(s)] done [0.00s].
sorting and recoding items ... [15 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 done [0.00s].
writing ... [25 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

With support=0.31, confidence= 0.68, I yielded 25 Rules which is what we're looking for.

Changing rhs to pep = 'YES' and finding top 5 Rules:

```r
myrules1 <- apriori(data=bank_data.transactions, parameter=list(support=0.31,confidence = 0.68),
                    appearance = list(default="lhs",rhs=('pep=YES')),control = list(verbose=F))
```

```r
summary(myrules1)
```

```
set of 0 rules
```

With existing support and confidence, I got 0 rules.

```r
myrules <- apriori(data=bank_data.transactions, parameter=list(support=0.11,confidence = 0.7),
                   appearance = list(default="lhs",rhs=('pep=YES')),control = list(verbose=F))
```

```r
inspect(myrules)
```

Description: df [7 × 8]

	lhs <chr>		rhs <chr>	support <dbl>	confidence <dbl>	coverage <dbl>	lift <dbl>	count <int>
[1]	{children=one_child}	=>	{pep=YES}	0.1833333	0.8148148	0.2250000	1.784266	110
[2]	{children=one_child, mortgage=NO}	=>	{pep=YES}	0.1183333	0.8452381	0.1400000	1.850886	71
[3]	{married=YES, children=one_child}	=>	{pep=YES}	0.1233333	0.8314607	0.1483333	1.820717	74
[4]	{children=one_child, save_act=YES}	=>	{pep=YES}	0.1333333	0.8421053	0.1583333	1.844026	80
[5]	{children=one_child, current_act=YES}	=>	{pep=YES}	0.1400000	0.8316832	0.1683333	1.821204	84
[6]	{married=NO, mortgage=NO}	=>	{pep=YES}	0.1533333	0.7076923	0.2166667	1.549691	92
[7]	{married=NO, current_act=YES, mortgage=NO}	=>	{pep=YES}	0.1216667	0.7156863	0.1700000	1.567196	73

7 rows

I kept changing values of support and confidence and was able to find these 7 rules

Business objectives:

1) {children=one_child} → People having only one child are highly likely to take PEP. This is understandable as family with one child or family who currently have 1 but are planning to have more, think about investing as they grow family .

   This rule is having support and confidence of 0.183 and 0.8 respectively. And a lift of 1.78 which is very impressive

2) {children=one_child, mortgage=NO} → People having one child and no mortgage also are likely to take PEP. This understandable given that the family is growing and they don't have any mortgage, so they will be open to invest in PEP. This rule is having support and confidence of 0.118 and 0.845 respectively. And a lift of 1.85 which is impressive

3) {children=one_child, save_act=YES} → Customer having one child and a saving account with bank are likely to invest in PEP. They made saving account under the intention of investing the money. Banks can give them extra interest if they invest in PEP.

4) {married=YES, children=one_child} → People who are married and have one child are also likely to invest. This rule is having support and confidence of 0.123 and 0.831 respectively. And a lift of 1.82 which is impressive

5) {children=one_child, current_act=YES} → People who have children and current account are likely to invest in PEP also. This rule is having support and confidence of 0.140 and 0.831 respectively. And a lift of 1.82 which is impressive

All in all it looks like bank should target customers with one child more. So by doing targeting advertisement and offering good investing returns on their savings or current account can be a way of making them invest in PEP.