Association Rules

Sudhanshu Kulkarni

February 04, 2019

# Background

We have been supplied with dataset of bank which contains attributes on each person’s demographics and banking information. Our goal to use Association Rule Mining to discover patterns which commonly occur when a user buys a new product, PEP in our case. Also, we need to recommend the bank to develop business opportunities depending on the patterns which we discovered.

In the dataset, we have the below information:

1. id - a unique identification number of the customer

2. age - age of the customer in years

3. sex - gender of the customer

4. region - region where the customer belongs to

5. income - income of the customer

6. married - marital status of the customer

7. children - number of children of the customer

8. car - if a customer owns a car

9. save\_acct - if a customer have a savings account

10. current\_acct - if a customer have a current account

11. mortgage - if a customer has a mortgage

12. pep - if a customer bought mortgage

# Solution

## Importing Libraries

For determining association rules we must first import arules, arulesViz.

library(arules)

## Warning: package 'arules' was built under R version 3.5.2

## Loading required package: Matrix

##   
## Attaching package: 'arules'

## The following objects are masked from 'package:base':  
##   
## abbreviate, write

library(arulesViz)

## Warning: package 'arulesViz' was built under R version 3.5.2

## Loading required package: grid

## Loading the Data

We load the data from the csv file provided into a dataset and check its structure.

bankDataDS <- read.csv("bankdata\_csv\_all.csv")  
View(bankDataDS)  
str(bankDataDS)

## 'data.frame': 600 obs. of 12 variables:  
## $ id : Factor w/ 600 levels "ID12101","ID12102",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ age : int 48 40 51 23 57 57 22 58 37 54 ...  
## $ sex : Factor w/ 2 levels "FEMALE","MALE": 1 2 1 1 1 1 2 2 1 2 ...  
## $ region : Factor w/ 4 levels "INNER\_CITY","RURAL",..: 1 4 1 4 2 4 2 4 3 4 ...  
## $ income : num 17546 30085 16575 20375 50576 ...  
## $ married : Factor w/ 2 levels "NO","YES": 1 2 2 2 2 2 1 2 2 2 ...  
## $ children : int 1 3 0 3 0 2 0 0 2 2 ...  
## $ car : Factor w/ 2 levels "NO","YES": 1 2 2 1 1 1 1 2 2 2 ...  
## $ save\_act : Factor w/ 2 levels "NO","YES": 1 1 2 1 2 2 1 2 1 2 ...  
## $ current\_act: Factor w/ 2 levels "NO","YES": 1 2 2 2 1 2 2 2 1 2 ...  
## $ mortgage : Factor w/ 2 levels "NO","YES": 1 2 1 1 1 1 1 1 1 1 ...  
## $ pep : Factor w/ 2 levels "NO","YES": 2 1 1 1 1 2 2 1 1 1 ...

## Data Preprocessing

To begin with our analysis, we must then proceed with cleaning the data. We remove the id column, as it is just a unique identification key by which the bank recognises its customer.

bankDataDS <- bankDataDS[,-1]

Association rules cannot be implied correctly on numeric values so we set bins for ages.

bankDataDS$age <- cut(bankDataDS$age, breaks = c(0,10,20,30,40,50,60,Inf),labels=c("child","teens","twenties","thirties","fourties","fifties","old"))

Similarly, we set bins for income values.

bankDataDS$income = cut(bankDataDS$income, breaks = c(0,25000,50000,Inf),labels = c("LOW","MEDIUM","HIGH"))

Then, we set bins for the children column. But, this time we follow a different method. First, we create a new temporary column, called newColumnChildren and prepopulate it with the value “YES”. Then, we change this value to “NO” when there is 0 in the children column. Followed by this we change the contents of the children column with newColumnChildren. This is how we changed the datatype of children column.

newColumnChildren<-replicate(length(bankDataDS$children), "YES")  
newColumnChildren[bankDataDS$children == 0]<-"NO"  
bankDataDS$children<- newColumnChildren  
bankDataDS$children=as.factor(bankDataDS$children)

Then we check the structure of the dataset again to confirm that all columns are discretized and ready for association rule mining.

str(bankDataDS)

## 'data.frame': 600 obs. of 11 variables:  
## $ age : Factor w/ 7 levels "child","teens",..: 5 4 6 3 6 6 3 6 4 6 ...  
## $ sex : Factor w/ 2 levels "FEMALE","MALE": 1 2 1 1 1 1 2 2 1 2 ...  
## $ region : Factor w/ 4 levels "INNER\_CITY","RURAL",..: 1 4 1 4 2 4 2 4 3 4 ...  
## $ income : Factor w/ 3 levels "LOW","MEDIUM",..: 1 2 1 1 3 2 1 1 2 1 ...  
## $ married : Factor w/ 2 levels "NO","YES": 1 2 2 2 2 2 1 2 2 2 ...  
## $ children : Factor w/ 2 levels "NO","YES": 2 2 1 2 1 2 1 1 2 2 ...  
## $ car : Factor w/ 2 levels "NO","YES": 1 2 2 1 1 1 1 2 2 2 ...  
## $ save\_act : Factor w/ 2 levels "NO","YES": 1 1 2 1 2 2 1 2 1 2 ...  
## $ current\_act: Factor w/ 2 levels "NO","YES": 1 2 2 2 1 2 2 2 1 2 ...  
## $ mortgage : Factor w/ 2 levels "NO","YES": 1 2 1 1 1 1 1 1 1 1 ...  
## $ pep : Factor w/ 2 levels "NO","YES": 2 1 1 1 1 2 2 1 1 1 ...

We then create rules using the Apriori algorithm. After a few permutations and combinations of support and confidence values we set these to 0.1 and 0.88, as we get the best rules.

rules <- apriori(bankDataDS, parameter = list(supp=0.1, conf = 0.88))

## Apriori  
##   
## Parameter specification:  
## confidence minval smax arem aval originalSupport maxtime support minlen  
## 0.88 0.1 1 none FALSE TRUE 5 0.1 1  
## maxlen target ext  
## 10 rules FALSE  
##   
## Algorithmic control:  
## filter tree heap memopt load sort verbose  
## 0.1 TRUE TRUE FALSE TRUE 2 TRUE  
##   
## Absolute minimum support count: 60   
##   
## set item appearances ...[0 item(s)] done [0.00s].  
## set transactions ...[29 item(s), 600 transaction(s)] done [0.00s].  
## sorting and recoding items ... [27 item(s)] done [0.00s].  
## creating transaction tree ... done [0.00s].  
## checking subsets of size 1 2 3 4 5 done [0.00s].  
## writing ... [25 rule(s)] done [0.00s].  
## creating S4 object ... done [0.00s].

rules

## set of 25 rules

inspect(rules)

## lhs rhs support confidence lift count  
## [1] {age=twenties} => {income=LOW} 0.1900000 0.9579832 1.903278 114  
## [2] {age=twenties,   
## region=INNER\_CITY} => {income=LOW} 0.1033333 0.9538462 1.895059 62  
## [3] {age=twenties,   
## sex=MALE} => {income=LOW} 0.1000000 0.9836066 1.954185 60  
## [4] {age=twenties,   
## car=NO} => {income=LOW} 0.1083333 0.9558824 1.899104 65  
## [5] {age=twenties,   
## pep=NO} => {income=LOW} 0.1250000 0.9493671 1.886160 75  
## [6] {age=twenties,   
## children=YES} => {income=LOW} 0.1016667 0.9242424 1.836243 61  
## [7] {age=twenties,   
## mortgage=NO} => {income=LOW} 0.1283333 0.9506173 1.888644 77  
## [8] {age=twenties,   
## married=YES} => {income=LOW} 0.1233333 0.9610390 1.909349 74  
## [9] {age=twenties,   
## save\_act=YES} => {income=LOW} 0.1166667 0.9589041 1.905108 70  
## [10] {age=twenties,   
## current\_act=YES} => {income=LOW} 0.1466667 0.9565217 1.900374 88  
## [11] {children=NO,   
## mortgage=NO,   
## pep=NO} => {married=YES} 0.1733333 0.9719626 1.472671 104  
## [12] {married=YES,   
## children=NO,   
## mortgage=NO} => {pep=NO} 0.1733333 0.8965517 1.650095 104  
## [13] {married=YES,   
## children=NO,   
## save\_act=YES} => {pep=NO} 0.1783333 0.8991597 1.654895 107  
## [14] {married=YES,   
## save\_act=YES,   
## pep=YES} => {children=YES} 0.1500000 0.8823529 1.570955 90  
## [15] {sex=FEMALE,   
## children=NO,   
## mortgage=NO,   
## pep=NO} => {married=YES} 0.1050000 0.9843750 1.491477 63  
## [16] {sex=FEMALE,   
## married=YES,   
## children=NO,   
## mortgage=NO} => {pep=NO} 0.1050000 0.9000000 1.656442 63  
## [17] {children=NO,   
## car=NO,   
## mortgage=NO,   
## pep=NO} => {married=YES} 0.1000000 0.9677419 1.466276 60  
## [18] {married=YES,   
## children=NO,   
## car=NO,   
## mortgage=NO} => {pep=NO} 0.1000000 0.8955224 1.648201 60  
## [19] {children=NO,   
## save\_act=YES,   
## mortgage=NO,   
## pep=NO} => {married=YES} 0.1216667 0.9864865 1.494676 73  
## [20] {married=YES,   
## children=NO,   
## save\_act=YES,   
## mortgage=NO} => {pep=NO} 0.1216667 0.9125000 1.679448 73  
## [21] {children=NO,   
## current\_act=YES,   
## mortgage=NO,   
## pep=NO} => {married=YES} 0.1333333 0.9756098 1.478197 80  
## [22] {married=YES,   
## children=NO,   
## current\_act=YES,   
## mortgage=NO} => {pep=NO} 0.1333333 0.9090909 1.673173 80  
## [23] {married=YES,   
## children=NO,   
## save\_act=YES,   
## current\_act=YES} => {pep=NO} 0.1333333 0.9195402 1.692405 80  
## [24] {married=YES,   
## current\_act=YES,   
## mortgage=NO,   
## pep=YES} => {children=YES} 0.1033333 0.8857143 1.576939 62  
## [25] {married=YES,   
## save\_act=YES,   
## current\_act=YES,   
## pep=YES} => {children=YES} 0.1150000 0.9078947 1.616430 69

Finally, we set PEP on the RHS to understand what factors govern the customers buying the PEP. But, before that we must convert our data into transactions.

bankDataDS <- data.frame(bankDataDS$age,bankDataDS$sex,bankDataDS$region,bankDataDS$income,  
 bankDataDS$married,bankDataDS$children,bankDataDS$car,bankDataDS$save\_act,  
 bankDataDS$current\_act,bankDataDS$mortgage,bankDataDS$pep)  
bankDataDS <- as(bankDataDS,"transactions")  
  
rulesRHS <- apriori(bankDataDS, parameter = list(supp=0.04, conf = 0.8)  
 , appearance = list(rhs=c("bankDataDS.pep=YES")))

## Apriori  
##   
## Parameter specification:  
## confidence minval smax arem aval originalSupport maxtime support minlen  
## 0.8 0.1 1 none FALSE TRUE 5 0.04 1  
## maxlen target ext  
## 10 rules FALSE  
##   
## Algorithmic control:  
## filter tree heap memopt load sort verbose  
## 0.1 TRUE TRUE FALSE TRUE 2 TRUE  
##   
## Absolute minimum support count: 24   
##   
## set item appearances ...[1 item(s)] done [0.00s].  
## set transactions ...[29 item(s), 600 transaction(s)] done [0.00s].  
## sorting and recoding items ... [29 item(s)] done [0.00s].  
## creating transaction tree ... done [0.00s].  
## checking subsets of size 1 2 3 4 5 6 7 done [0.01s].  
## writing ... [32 rule(s)] done [0.00s].  
## creating S4 object ... done [0.00s].

rulesRHS

## set of 32 rules

inspect(rulesRHS)

## lhs rhs support confidence lift count  
## [1] {bankDataDS.income=HIGH} => {bankDataDS.pep=YES} 0.06166667 0.8222222 1.800487 37  
## [2] {bankDataDS.age=old,   
## bankDataDS.income=HIGH} => {bankDataDS.pep=YES} 0.04666667 0.8484848 1.857996 28  
## [3] {bankDataDS.income=HIGH,   
## bankDataDS.children=YES} => {bankDataDS.pep=YES} 0.05166667 1.0000000 2.189781 31  
## [4] {bankDataDS.income=HIGH,   
## bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.04333333 0.8387097 1.836591 26  
## [5] {bankDataDS.income=HIGH,   
## bankDataDS.save\_act=YES} => {bankDataDS.pep=YES} 0.06166667 0.8222222 1.800487 37  
## [6] {bankDataDS.income=HIGH,   
## bankDataDS.current\_act=YES} => {bankDataDS.pep=YES} 0.04833333 0.8055556 1.763990 29  
## [7] {bankDataDS.age=old,   
## bankDataDS.income=HIGH,   
## bankDataDS.children=YES} => {bankDataDS.pep=YES} 0.04333333 1.0000000 2.189781 26  
## [8] {bankDataDS.age=old,   
## bankDataDS.income=HIGH,   
## bankDataDS.save\_act=YES} => {bankDataDS.pep=YES} 0.04666667 0.8484848 1.857996 28  
## [9] {bankDataDS.income=HIGH,   
## bankDataDS.children=YES,   
## bankDataDS.save\_act=YES} => {bankDataDS.pep=YES} 0.05166667 1.0000000 2.189781 31  
## [10] {bankDataDS.income=HIGH,   
## bankDataDS.children=YES,   
## bankDataDS.current\_act=YES} => {bankDataDS.pep=YES} 0.04000000 1.0000000 2.189781 24  
## [11] {bankDataDS.income=HIGH,   
## bankDataDS.save\_act=YES,   
## bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.04333333 0.8387097 1.836591 26  
## [12] {bankDataDS.income=HIGH,   
## bankDataDS.save\_act=YES,   
## bankDataDS.current\_act=YES} => {bankDataDS.pep=YES} 0.04833333 0.8055556 1.763990 29  
## [13] {bankDataDS.age=old,   
## bankDataDS.sex=FEMALE,   
## bankDataDS.children=YES} => {bankDataDS.pep=YES} 0.04000000 0.8000000 1.751825 24  
## [14] {bankDataDS.age=old,   
## bankDataDS.children=YES,   
## bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.05833333 0.8333333 1.824818 35  
## [15] {bankDataDS.age=old,   
## bankDataDS.children=YES,   
## bankDataDS.current\_act=YES} => {bankDataDS.pep=YES} 0.05833333 0.8333333 1.824818 35  
## [16] {bankDataDS.married=NO,   
## bankDataDS.children=NO,   
## bankDataDS.save\_act=NO} => {bankDataDS.pep=YES} 0.04333333 0.9285714 2.033368 26  
## [17] {bankDataDS.children=NO,   
## bankDataDS.save\_act=NO,   
## bankDataDS.mortgage=YES} => {bankDataDS.pep=YES} 0.05666667 0.9189189 2.012231 34  
## [18] {bankDataDS.married=NO,   
## bankDataDS.children=NO,   
## bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.07500000 0.9375000 2.052920 45  
## [19] {bankDataDS.age=old,   
## bankDataDS.income=HIGH,   
## bankDataDS.children=YES,   
## bankDataDS.save\_act=YES} => {bankDataDS.pep=YES} 0.04333333 1.0000000 2.189781 26  
## [20] {bankDataDS.income=HIGH,   
## bankDataDS.children=YES,   
## bankDataDS.save\_act=YES,   
## bankDataDS.current\_act=YES} => {bankDataDS.pep=YES} 0.04000000 1.0000000 2.189781 24  
## [21] {bankDataDS.age=old,   
## bankDataDS.children=YES,   
## bankDataDS.save\_act=YES,   
## bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.05000000 0.8333333 1.824818 30  
## [22] {bankDataDS.age=old,   
## bankDataDS.children=YES,   
## bankDataDS.current\_act=YES,   
## bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.05000000 0.8571429 1.876955 30  
## [23] {bankDataDS.age=old,   
## bankDataDS.children=YES,   
## bankDataDS.save\_act=YES,   
## bankDataDS.current\_act=YES} => {bankDataDS.pep=YES} 0.05166667 0.8378378 1.834681 31  
## [24] {bankDataDS.children=NO,   
## bankDataDS.save\_act=NO,   
## bankDataDS.current\_act=YES,   
## bankDataDS.mortgage=YES} => {bankDataDS.pep=YES} 0.04333333 0.9285714 2.033368 26  
## [25] {bankDataDS.income=MEDIUM,   
## bankDataDS.married=NO,   
## bankDataDS.save\_act=YES,   
## bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.05333333 0.8205128 1.796743 32  
## [26] {bankDataDS.sex=MALE,   
## bankDataDS.married=NO,   
## bankDataDS.children=NO,   
## bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.04166667 0.9259259 2.027575 25  
## [27] {bankDataDS.income=LOW,   
## bankDataDS.married=NO,   
## bankDataDS.children=NO,   
## bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.04000000 0.9230769 2.021336 24  
## [28] {bankDataDS.married=NO,   
## bankDataDS.children=NO,   
## bankDataDS.save\_act=YES,   
## bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.05166667 0.9687500 2.121350 31  
## [29] {bankDataDS.married=NO,   
## bankDataDS.children=NO,   
## bankDataDS.current\_act=YES,   
## bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.05833333 0.9459459 2.071414 35  
## [30] {bankDataDS.age=old,   
## bankDataDS.children=YES,   
## bankDataDS.save\_act=YES,   
## bankDataDS.current\_act=YES,   
## bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.04333333 0.8666667 1.897810 26  
## [31] {bankDataDS.income=MEDIUM,   
## bankDataDS.married=NO,   
## bankDataDS.save\_act=YES,   
## bankDataDS.current\_act=YES,   
## bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.04166667 0.8333333 1.824818 25  
## [32] {bankDataDS.married=NO,   
## bankDataDS.children=NO,   
## bankDataDS.save\_act=YES,   
## bankDataDS.current\_act=YES,   
## bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.04166667 0.9615385 2.105559 25

# Interesting Rules

{bankDataDS.married=NO,

bankDataDS.children=NO,

bankDataDS.save\_act=YES,

bankDataDS.current\_act=YES,

bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.04166667 0.9615385 2.105559 25

I found this rule interesting because, for most of the cases we have encountered, we have seen that customers who are married, or have children, in other words – have dependents, are the ones who go for PEP. But, we also have a case where there are no dependents and still the customers are buying PEP.

So, the bank can also target customers who hold any type of account in the bank and have no dependency.

For the above rule, we have below support, confidence and life values.

Support = 0.04166667

This means that how frequently this pattern appears in the dataset.

Confidence = 0.9615385

This is the measure of trueness of the pattern.

Lift = 2.105559

This value suggests the probability of the pattern appearing together. So, higher the lift value, higher the probability.

This is how we compute the support, confidence and lift values.

{bankDataDS.income=MEDIUM,

bankDataDS.married=NO,

bankDataDS.save\_act=YES,

bankDataDS.current\_act=YES,

bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.04166667 0.8333333 1.824818 25

Again, this rule is interesting because for most of the cases we have found that the customers who have a high income generally go for PEP. But, in this we notice that customers who have medium income also go for PEP.

This means that banks also need to focus on the customers who have medium income but hold accounts in the bank.

{bankDataDS.income=LOW,

bankDataDS.married=NO,

bankDataDS.children=NO,

bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.04000000 0.9230769 2.021336 24

This is the most interesting rule. As we discussed earlier that for most of the customers buying PEP, the income is high, and they have some kind of dependency. But, in this one the income is not only low but there are no dependencies.

{bankDataDS.age=old,

bankDataDS.children=YES,

bankDataDS.save\_act=YES,

bankDataDS.current\_act=YES,

bankDataDS.mortgage=NO} => {bankDataDS.pep=YES} 0.04333333 0.8666667 1.897810 26

This again is an interesting rule because old customers do not generally buy new products. They stick to the products which they are aware about. So, seeing that even old customers are buying this product seems interesting. So, the bank can target customers who are old in age.

{bankDataDS.income=HIGH,

bankDataDS.children=YES} => {bankDataDS.pep=YES} 0.05166667 1.0000000 2.189781 31

As mentioned earlier, we have seen this product to be popular amongst the customers who have high income and have some dependency. So, this rule is of real interest to the bank as they can know who their potential customers are. This rule translates as customers who are wealthy and have dependency buy PEP.

# Recommendations for Business

1. Most of the customers who buy this product are the ones with high income and have a dependency in a child or spouse. So, the bank can concentrate on just sending mails and other goodies to reach out to them and value them.
2. Also, the bank could also try and reach out to the customers who hold any type of account with the bank as they are the potential buyers of the PEP.
3. To encourage more customers to buy this product the bank should modify their product, i.e. customized product, for people having low income, no dependencies, no bank account. They could give them lucrative offers, gift vouchers to encourage them to buy this product.