



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment - 1.3

Student Name: Praduman Kumar

Branch: BE-CSE

Semester: 6th

Subject Name: Data Mining Lab

UID: 20BCS9446

Section/Group: 714/A

Date of Performance: 03/03/2023

Subject Code: 20CSP-376

1. Aim:

Demonstration of association rule mining using Apriori algorithm on supermarket data .

2. Objective:

To demonstration of association rule mining using Apriori algorithm on supermarket data .

3. Summary :

ARM : ARM (Association Rule Mining) is a popular technique in data mining that involves discovering patterns or relationships among data items in large databases. Specifically, it is used to identify frequent itemsets and association rules, which are often used for making recommendations or predicting customer behavior.

In ARM, the data is represented as transactions or sets of items, and the algorithm searches for frequent itemsets, which are subsets of items that appear together in a significant number of transactions. Once frequent itemsets are identified, association rules are generated by examining the relationships between the items in the itemsets.

APRIORI ALGORITHM : The Apriori algorithm is a classic algorithm used in association rule mining, a technique used in data mining to discover interesting relationships or patterns between items in a dataset. The algorithm is named after the fact that it employs an "a priori" knowledge of frequent itemsets to efficiently search for all association rules.

The Apriori algorithm is widely used in market basket analysis, customer behavior analysis, and other areas where discovering relationships between items is important. Its efficiency and effectiveness in identifying frequent itemsets and association rules have made it a popular and widely used algorithm in the field of data mining.

In data mining, support and confidence are two measures used in association rule mining to identify patterns or relationships between different items in a dataset.

Support refers to the frequency with which a particular itemset appears in a dataset. It is calculated as the number of transactions that contain the itemset divided by the total number of transactions in the dataset. The support value ranges from 0 to 1, where 0 means the itemset never appears in the dataset, and 1 means it appears in all transactions.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Confidence, on the other hand, measures the strength of the association between two items in an itemset. It is calculated as the number of transactions that contain both items divided by the number of transactions that contain the first item. The confidence value also ranges from 0 to 1, where 0 means no association, and 1 means a perfect association between the items.

4. Script/Code/Steps:

```
install.packages("arules")
install.packages("arulesViz")
install.packages("RColorBrewer")
#import dataset
data("Groceries")
#using apriori() function
rules<-apriori(Groceries,parameter = list(supp = 0.01,conf = 0.2))
#using inspect() function
inspect(rules[1:10])
#using itemFrequency() function
arules:: itemFrequencyPlot(Groceries, topN = 20, col = brewer.pal(8,'Pastel2'), main='Realtive Item
Frequency plot', type = 'relative', ylab = "Item Frequency(Relative)")
library(help = "datasets")
```

4. Output Screenshots :

```
R 4.2.2 ~ /
> data("Groceries")
> rules<-apriori(Groceries,parameter = list(supp = 0.01,conf = 0.2))
Apriori

Parameter specification:
 confidence minval smax arem aval originalSupport maxtime support minlen maxlen target ext
 0.2      0.1    1 none FALSE          TRUE      5    0.01    1    10 rules TRUE

Algorithmic control:
 filter tree heap memopt load sort verbose
  0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 98

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
sorting and recoding items ... [88 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 done [0.00s].
writing ... [232 rule(s)] done [0.00s].
creating S4 object ... done [0.00s].
> inspect(rules[1:10])
   lhs                rhs      support  confidence coverage  lift    count
[1] {} => {whole milk}    0.25551601 0.2555160  1.00000000 1.000000 2513
[2] {hard cheese} => {whole milk}    0.01006609 0.4107884  0.02450432 1.607682   99
[3] {butter milk} => {other vegetables} 0.01037112 0.3709091  0.02796136 1.916916  102
[4] {butter milk} => {whole milk}    0.01159126 0.4145455  0.02796136 1.622385  114
[5] {ham} => {whole milk}    0.01148958 0.4414062  0.02602949 1.727509  113
[6] {sliced cheese} => {whole milk}  0.01077783 0.4398340  0.02450432 1.721356  106
[7] {oil} => {whole milk}    0.01128622 0.4021739  0.02806304 1.573968  111
[8] {onions} => {other vegetables} 0.01423488 0.4590164  0.03101169 2.372268  140
[9] {onions} => {whole milk}    0.01209964 0.3901639  0.03101169 1.526965  119
[10] {berries} => {yogurt}    0.01057448 0.3180428  0.03324860 2.279848  104
> arules:: itemFrequencyPlot(Groceries, topN = 20, col = brewer.pal(8,'Pastel2'), main='Realtive Item
Frequency plot', type = 'relative', ylab = "Item Frequency(Relative)")
> library(help = "datasets")
>
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

