Assignment No - 4

Date : / /20

Title - Case study (Market Basket Analysis)

Problem statement : A mall has no. of items for sale. Build a required database to develop BA & I tool for considering one aspects of growth of the business such as organization of products based on demand & patterns.

Input : Transaction Database and minimum support

Output : Frequent item sets. Association Rules & graphical representation of rules as per confidences & lift.

Pre-Lab : 1. Knowledge of R programming Language
2. Concept & theory of Apriori algorithm

Theory :-

By convention, the algorithm assume that items within a transaction or itemset are sorted in lexicographic order. It employees an iterative approach known as a level-wise search, where item-set are used to explore k itemset. First, the set of frequent 1-itemsets is found by scanning the database to accumulate the count for each item & collecting those item that satisfy minimum support. The resulting set is denoted as L_1. Next, L_1 is used to find L_2, the set of frequent 2-item set, which is then used to find L_3 & so on, until no more frequent k-itemset can be found

Pooja

To improve the efficiency of the level-wise generation of frequent itemsets, an important property called Apriori property is used to reduce the search space

Apriori Property - All nonempty subsets of a frequent itemset must also be frequent.

This property is based on the following observation If an itemset A does not satisfy the minimum support threshold, min sup then A is not frequent i.e. $P(A) <$ min-sup. If an item B is added to the itemset A, then resulting itemset is $A \cup B$ can't occur more frequenty than A. therefor $A \cup B$ is not frequent either that is $P(A \cup B) <$ min-sup.

A two-step process is used to find L_k from L_{k-1} for $K \geq 2$

1. The join step :- To find L_k is set of candidate k-itemset is generated by joining L_{k-1} with itself. This set of candidate is denoted by C_k. Let l_2 be itemset in L_{k-1} The notation $l_i[j]$ refers to jth item in l_i. Thus in l_i, the last item & the next to the last item are given respectively by $l_1[k-1]$ & $l_1[k-2]$. Any two itemset L_{k-1} are joined if their first $(k-2)$ items are in common, That is, members l_1 and l_2 are joined if $(l_1[1] = l_2[1] \wedge (l_1[2] = l_2[2]) \wedge \ldots \wedge (l_1[k-2] = l_2[k-2]) \wedge (l_1[k-1] < l_2[k-1])$ The condition simply ensure that no duplicates are generated. The resulting itemset formed by joining l_1 and l_2 is $\{l_1[1], l_1[2], \ldots l_1[k-2], l_1[k-1]\}$.

Date : / /20

2. The prune step - set C_K is a subset of L_K, because although all frequent k-itemset are inculded in C_K, its members may or may not Frequent. one could scan the database to determine minimum support count of each candidate in C_K & eliminate any itemset that does not meet the minimum support threshold. This would the given L_K. However C_K can be huge & so this could be very time-consuming. To eliminate the infrequent itemsets the Apriori property is used as follows. Any (k-1) itemset that is not frequent cannot be a subset of a frequent k-itemset. Hence if any (k-1) itemset itemset of a candidate k-itemset is not in L_{K-1} then the candidate cannot be frequent.

An Example of the use of The Apriori Algorithm -
We illustrate the use of Apriori algorithm for finding frequent itemsets in our transaction database D in the first iteration of algorithm each item is a member of set of Candidates 1-itemsets, C1 The algorithm simply scans all the transaction in order to count the number of occurance of each item.

C1 itemset	Support Count	L1 itemset	Support Count
{1}	6	{1}	6
{2}	7	{2}	7
{3}	6	{3}	6
{4}	2	{4}	2
{5}	2	{5}	2
{6}	1		

Pooja

To discover the set of frequent 2-itemset, 1, 2, the algorithm joins L_1 with self w to generate candidate itemset set of 2-itemset, C_2. Note that no candidate are removed from C_2 during the prunning step.

C_2 itemset	C_2 itemset	Support Count
{1,2}	{1,2}	4
{1,3}	{1,3}	4
{1,4}	{1,4}	1
{1,5}	{1,5}	2
{2,3}	{2,3}	4
{2,4}	{2,4}	2
{2,5}	{2,5}	2
{3,4}	{3,4}	0
{3,5}	{3,5}	1
{4,5}	{4,5}	0

Next the transaction in D are scaned & the support cound of each candidate in C_2 is accumulated. The set of frequent 2-itemset, L_2, is determined. consisting of those candidate 2-itemset in C_2 having minimum support.

L_2 itemset	Support Count
{1,2}	4
{1,3}	4
{1,4}	2
{2,3}	4
{2,4}	2
{2,5}	2

Date : / /20

Next C_3 is generated by joining L_2 itself
The result is C_3 = {{1, 2, 3}, {1, 2, 5}, {1, 3, 5}}
{2, 3, 4}, {2, 3, 5}, {2, 4, 5}}. C_3 is purned using
Apriori property. All nonempty subsets of frequent
itemset must also be frequent. From way each
candidate of C_3 is formed.

the candidate set

since {2, 3} is a frequent itemset, we keep {1, 2, 3} in C_3
since {2, 5} is a frequent itemset, we keep {1, 2, 5} in C_3
since {3, 5} is not frequent itemset, we remove {1, 3, 5} from C_3
since {3, 4} is not frequent itemset, we remove {2, 3, 4} from C_3
since {3, 5} is not frequent itemset, we remove {2, 3, 5} from C_3
since {4, 5} is not frequent itemset, we remove {2, 4, 5} from C_3

Therefore after pruning C_3 given by

C_3 itemset
{1, 2, 3}
{1, 2, 5}

The transaction in D are scanned to determine L_3
consisting of those candidates - 3 itemsets in C_3
having at least minimum support.

C_3 itemset	Support Count
{1, 2, 3}	2
{1, 2, 5}	2

since both 3-itemset in C_3 have the least minimum
support, L_3 is given by

L_3 itemset	Support Count
{1, 2, 3}	2
{1, 2, 5}	2

Finally L_3 joined with itself to generate a
candidate set of 4-itemset C_4.

Pooja

this result in a single itemset {1,2,3,5}. However the itemset is pruned since its subset {3,5} is not frequent. Thus $C_4 = \phi$ and algorithm terminate, having found all of the frequent itemsets.

Execution Guidelines :-

1. Install . packages 'arules', 'arulesviz', from Cran mirror through HTTP..
2. Use data set 'Groceries'
3. Use apriori function in R to get itemset providing length of item set & support.
4. Generate rules using aproni function in R to get itemset & support set.
5. plot rules for given confidence
6. Plot graph of visualizing the high lift rules.

Analysis - 1. observe the graphs for generated rules with different support confidence & lift.
2. observe top rules & use this patterns for organization of products.

Conclusion :-

Thus the Groceries dataset is used to generate rules & applied rules for organization of products based on patterns & demand. Frequent itemset are found using apriori algorithm based on rule data mining technique. observations are recorded in terms of graph.