Association Rule Mining

Learning Association Rule Mining made easy!

Association Rule Mining Algorithms ?

- **Apriori Algorithm.**
- **FP Growth Algorithm.**

What is FP Growth Algorithm ?

An efficient and scalable method to find frequent patterns. It allows **frequent itemset** discovery without candidate itemset generation.

Following are the steps for FP Growth Algorithm

- Scan DB once, find frequent 1-itemset (single item pattern)
- Sort frequent items in frequency descending order, f-list
- Scan DB again, construct FP-tree
- Construct the conditional FP tree in the sequence of reverse order of F - List - generate frequent item set

Illustration:

Consider the below tansaction in which B = Bread, J = Jelly, P = Peanut Butter, M = Milk and E = Eggs. Given that minimum threshold support = 40% and minimum threshold confidence = 80% [13].

TID	Items Bought
1	{B, J , P}
2	{B, P}
3	{B, M, P}
4	{E, B}
5	{E, M}

Step-1: Scan DB once, find frequent 1-itemset (single item in itemset)

Items	Support	Support (in percentage) = (Support * 100) / No. of trans
B	4	(4 * 100) / 5 = 80%
J	1	(1 * 100) / 5 = 20%
P	3	(3 * 100) / 5 = 60%
M	2	(2 * 100) / 5 = 40%
E	2	(2 * 100) / 5 = 40%

Step-2: As minimum threshold support = 40%, So in this step we will remove all the items that are bought less than 40% of support or support less than 2.

Items	Support
B	4 (80%)
P	3 (60%)
M	2 (40%)
E	2 (40%)

Step-3: Create a F -list in which frequent items are sorted in the descending order based on the **support**.

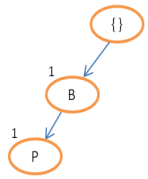
F - list	{B , P , M , E}

Step-4: Sort frequent items in transactions based on F-list. It is also known as FPDP.

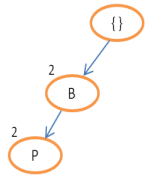
FPDB	
TID	Items Bought
1	{B, P}
2	{B, P}
3	{B, P , M}
4	{B , E}
5	{M, E}

Step-5: Construct the FP tree

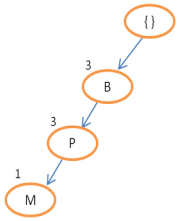
- Read transaction 1: {B,P} -> Create 2 nodes B and P. Set the path as null -> B -> P and the count of B and P as 1 as shown below :



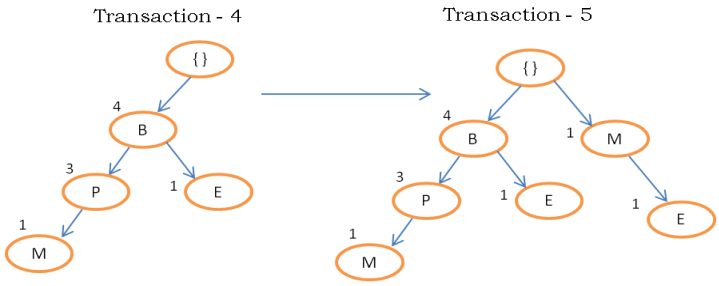
- Read transaction 2: {B,P} -> The path will be null -> B -> P. As transaction 1 and 2 share the same path. Set counts of B and P to 2.



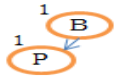
- Read transaction 3: {B,P,M} -> The path will be null -> B -> P -> M. As transaction 2 and 3 share the same path till node P. Therefore, set count of B and P as 3 and create node M having count 1.



- Continue until all the transactions are mapped to a path in FP-tree.



Step-6: Construct the conditional FP tree in the sequence of reverse order of F - List {E,M,P,B} and generate frequent item set. The conditional FP tree is sub tree which is built by considering the transactions of a particular item and then removing that item from all the transaction.

Item set	Transactions	Conditional FP Tree	Frequent Item set
E	TID-4 = { B } TID-5 = { M }		Nil
M	TID-3 = { B, P }		Nil
P	TID-1 = { B } TID-2 = { B } TID-3 = { B }		{ B, P }

The above table has two items {B , P} that are bought together frequently.

As for items E and M, nodes in the conditional FP tree has a count(support) of 1 (less than minimum threshold support 2). Therefore frequent itemset are nil. In case of item P, node B in the conditional FP tree has a count(support) of 3 (satisfying minimum threshold support). Hence frequent itemset is

generated by adding the item P to the B.

Try Examples