Name: Sheetal Vishwakarma **Roll no: 33**

Semester: 7th **Subject: DM Lab (IT-2402)**

Reg.no: 19030036

Practical No: 04

Aim: Demonstration of Association rule mining on dataset using FP-GROWTH algorithm.

Theory:

F-P Growth Algorithm:

The F-P growth algorithm stands for **Frequent Pattern**, and it is the improved version of the Apriori Algorithm. It represents the database in the form of a tree structure that is known as a frequent pattern or tree. The purpose of this frequent tree is to extract the most frequent patterns.

This algorithm is an improvement to the Apriori method. A frequent pattern is generated without the need for candidate generation. FP growth algorithm represents the database in the form of a tree called a frequent pattern tree or FP tree.

This tree structure will maintain the association between the itemsets. The database is fragmented using one frequent item. This fragmented part is called “pattern fragment”. The itemsets of these fragmented patterns are analyzed. Thus with this method, the search for frequent itemsets is reduced comparatively.

Frequent Pattern Algorithm Steps:

1. The first step is to scan the database to find the occurrences of the itemsets in the
e same as the first step of Apriori. The count of 1-itemsets in
pport count or frequency of 1-itemset.
nstruct the FP tree. For this, create the root of the tree. The
ull.
he database again and examine the transactions. Examine the
out the itemset in it. The itemset with the max count is taken
set with lower count and so on. It means that the branch of
ith transaction itemsets in descending order of count.
the database is examined. The itemsets are ordered in
unt. If any itemset of this transaction is already present in

another branch (for example in the 1st transaction), then this transaction branch would share a common prefix to the root.
This means that the common itemset is linked to the new node of another itemset in this transaction.

5. Also, the count of the itemset is incremented as it occurs in the transactions. Both the common node and new node count is increased by 1 as they are created and linked according to transactions.
6. The next step is to mine the created FP Tree. For this, the lowest node is examined first along with the links of the lowest nodes. The lowest node represents the frequency pattern length 1. From this, traverse the path in the FP Tree. This path or paths are called a conditional pattern base.
   Conditional pattern base is a sub-database consisting of prefix paths in the FP tree occurring with the lowest node (suffix).

7. Construct a Conditional FP Tree, which is formed by a count of itemsets in the path. The itemsets meeting the threshold support are considered in the Conditional FP Tree.
8. Frequent Patterns are generated from the Conditional FP Tree.

Advantages Of FP Growth Algorithm

1. This algorithm needs to scan the database only twice when compared to Apriori which scans the transactions for each iteration.
2. The pairing of items is not done in this algorithm and this makes it faster.
3. The database is stored in a compact version in memory.
4. It is efficient and scalable for mining both long and short frequent patterns.

Disadvantages Of FP-Growth Algorithm

1. FP Tree is more cumbersome and difficult to build than Apriori.
2. It may be expensive.
3. When the database is large, the algorithm may not fit in the shared memory.

Dataset:

supermarket - Notepad

File Edit Format View Help

```
@relation supermarket
@attribute 'department1' { t}
@attribute 'department2' { t}
@attribute 'department3' { t}
@attribute 'department4' { t}
@attribute 'department5' { t}
@attribute 'department6' { t}
@attribute 'department7' { t}
@attribute 'department8' { t}
@attribute 'department9' { t}
@attribute 'grocery misc' { t}
@attribute 'department11' { t}
@attribute 'baby needs' { t}
@attribute 'bread and cake' { t}
@attribute 'baking needs' { t}
@attribute 'coupons' { t}
@attribute 'juice-sat-cord-ms' { t}
@attribute 'tea' { t}
@attribute 'biscuits' { t}
@attribute 'canned fish-meat' { t}
@attribute 'canned fruit' { t}
@attribute 'canned vegetables' { t}
@attribute 'breakfast food' { t}
@attribute 'cigs-tobacco pkts' { t}
@attribute 'cigarette cartons' { t}
@attribute 'cleaners-polishers' { t}
@attribute 'coffee' { t}
@attribute 'sauces-gravy-pkle' { t}
@attribute 'confectionary' { t}
@attribute 'puddings-deserts' { t}
@attribute 'dishcloths-scour' { t}
@attribute 'deod-disinfectant' { t}
@attribute 'frozen foods' { t}
@attribute 'razor blades' { t}
@attribute 'fuels-garden aids' { t}
@attribute 'spices' { t}
```

Type here to search

6:32 PM 19/10/2021

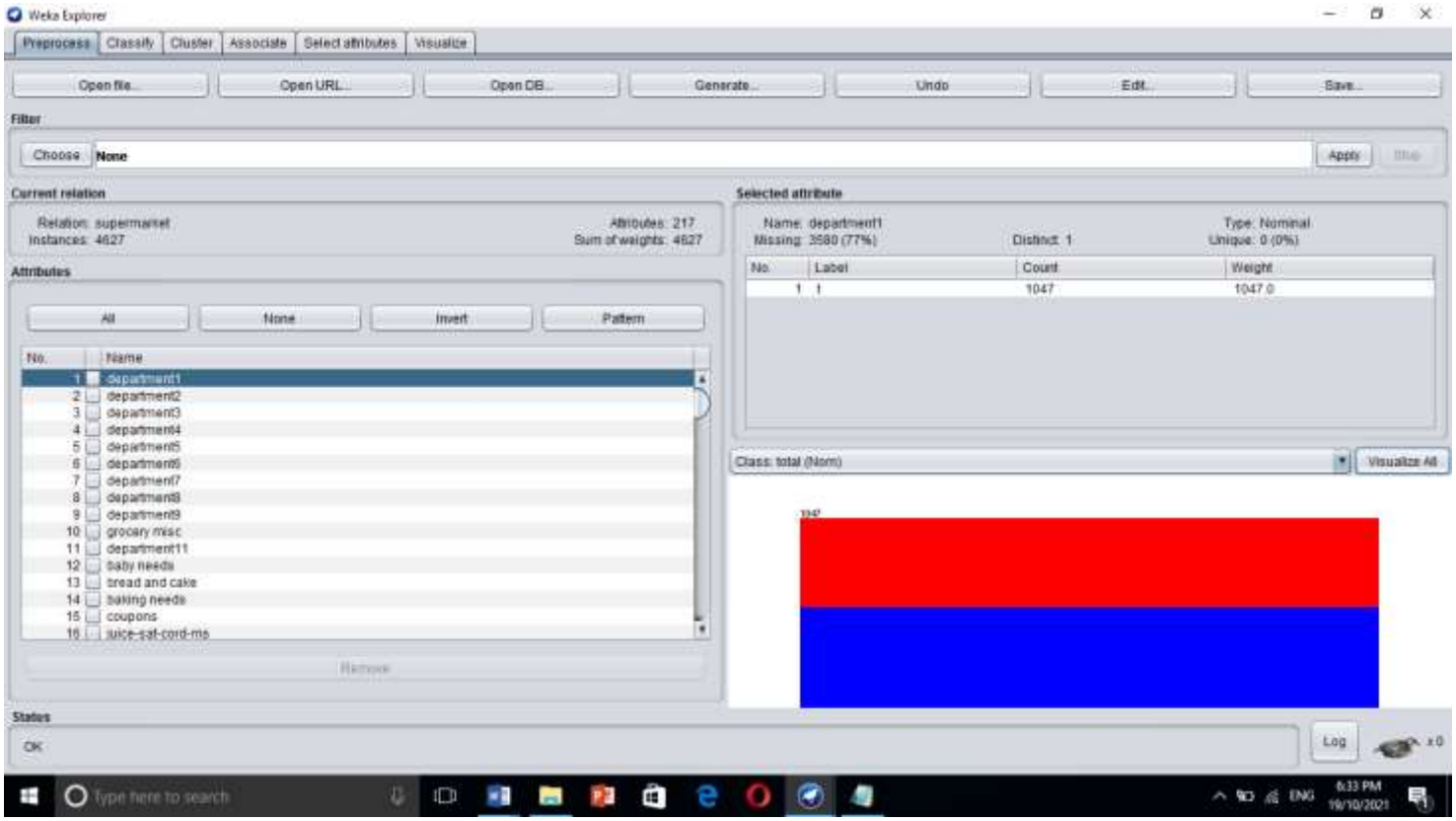
supermarket - Notepad

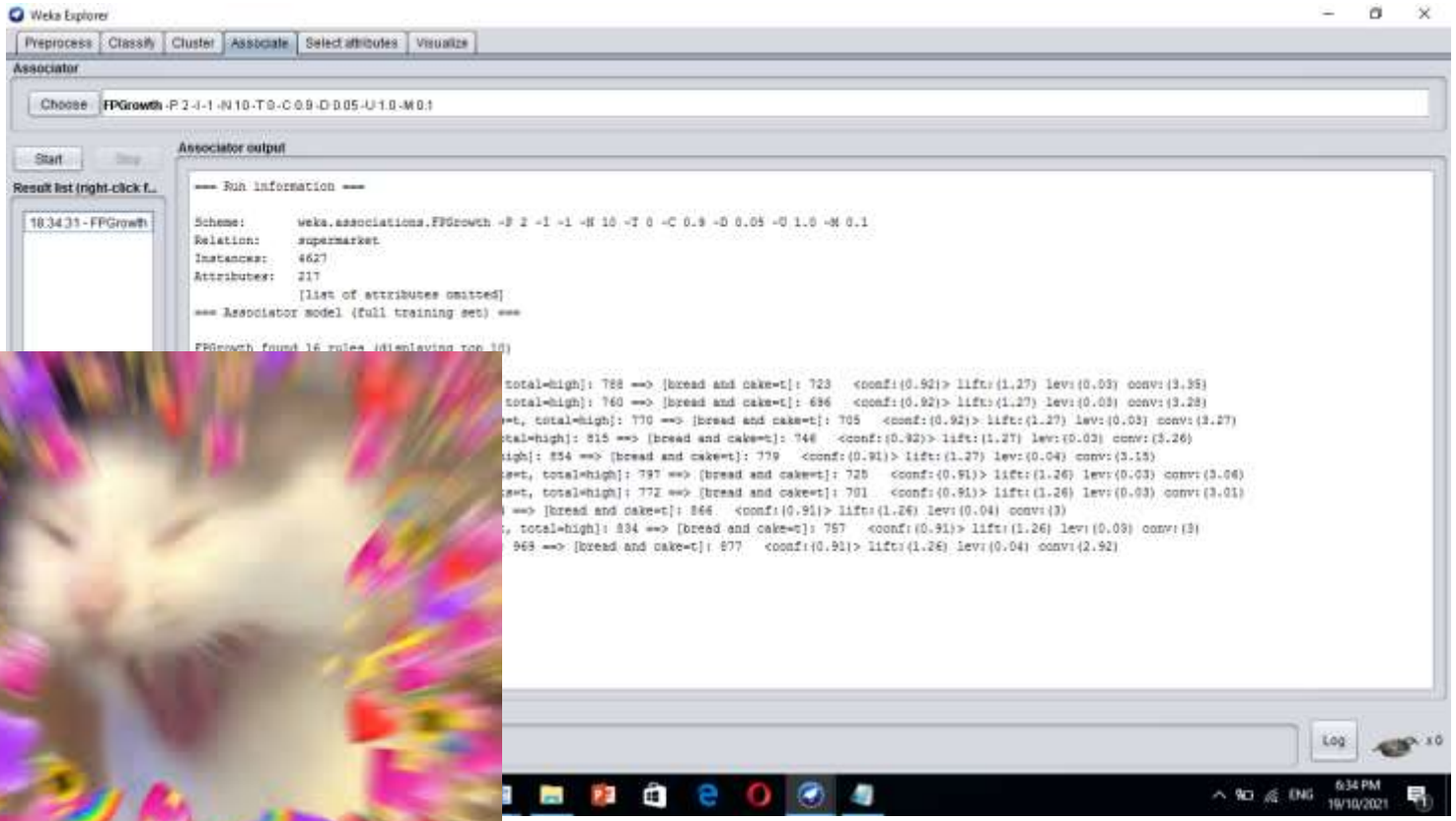
File Edit Format View Help

```
@attribute 'jams-spreads' { t}
@attribute 'insecticides' { t}
@attribute 'pet foods' { t}
@attribute 'laundry needs' { t}
@attribute 'party snack foods' { t}
@attribute 'tissues-paper prd' { t}
@attribute 'wrapping' { t}
@attribute 'dried vegetables' { t}
@attribute 'pkt-canned soup' { t}
@attribute 'soft drinks' { t}
@attribute 'health food other' { t}
@attribute 'beverages hot' { t}
@attribute 'health&beauty misc' { t}
@attribute 'deodorants-soap' { t}
@attribute 'mens toiletries' { t}
@attribute 'medicines' { t}
```

6:33 PM 19/10/2021

Screenshot:


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Open file...
Open URL...
Open DB...
Generate...
Undo
Edit...
Save...
Filter
Choose
None
Apply
Current relation
Relation: supermarket
Instances: 4627
Attributes: 217
Sum of weights: 4627
Selected attribute
Name: department1
Missing: 3580 (77%)
Distinct: 1
Type: Nominal
Unique: 0 (0%)
No.
Label
Count
Weight
1
t
1047
1047.0
Attributes
All
None
Invert
Pattern
No.
Name
1 department1
2 department2
3 department3
4 department4
5 department5
6 department6
7 department7
8 department8
9 department9
10 grocery misc
11 department11
12 baby needs
13 bread and cake
14 baking needs
15 coupons
16 juice-sat-cord-ms
Remove
Class: total (Nom)
Visualize All
1047
Status
OK
Log
x 0
Type here to search
ENG
6:33 PM
19/10/2021


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Associator
Choose
FPGrowth -P 2 -I -1 -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1
Start
Stop
Result list (right-click f...
18:34:31 - FPGrowth
Associator output
=== Run information ===
Scheme: weka.associations.FPGrowth -P 2 -I -1 -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1
Relation: supermarket
Instances: 4627
Attributes: 217
[list of attributes omitted]
=== Associator model (full training set) ===
FPGrowth found 16 rules (displaying top 10)
total=high]: 788 ==> [bread and cake=t]: 723 <conf:(0.92)> lift:(1.27) lev:(0.03) conv:(3.35)
total=high]: 760 ==> [bread and cake=t]: 696 <conf:(0.92)> lift:(1.27) lev:(0.03) conv:(3.28)
=t, total=high]: 770 ==> [bread and cake=t]: 705 <conf:(0.92)> lift:(1.27) lev:(0.03) conv:(3.27)
tal=high]: 815 ==> [bread and cake=t]: 746 <conf:(0.92)> lift:(1.27) lev:(0.03) conv:(3.26)
igh]: 854 ==> [bread and cake=t]: 779 <conf:(0.91)> lift:(1.27) lev:(0.04) conv:(3.15)
=t, total=high]: 797 ==> [bread and cake=t]: 725 <conf:(0.91)> lift:(1.26) lev:(0.03) conv:(3.06)
=t, total=high]: 772 ==> [bread and cake=t]: 701 <conf:(0.91)> lift:(1.26) lev:(0.03) conv:(3.01)
==> [bread and cake=t]: 866 <conf:(0.91)> lift:(1.26) lev:(0.04) conv:(3)
, total=high]: 834 ==> [bread and cake=t]: 757 <conf:(0.91)> lift:(1.26) lev:(0.03) conv:(3)
969 ==> [bread and cake=t]: 877 <conf:(0.91)> lift:(1.26) lev:(0.04) conv:(2.92)
Log
x 0
ENG
6:34 PM
19/10/2021

Conclusion: Thus, we studied about how to Demonstration of Association rule mining on lgorithm.