PREPARED BY: RAJARAJESWARI VAIDYANATHAN CWID: A20362220 COURSE: CS422 – DATA MINING

EXPLORING WEKA SOFTWARE TOOL AND ANALYZING THE RESULTS PRODUCED BY DIFFERENT DATASETS WITH RESPECT TO ASSOCIATION RULE MINING.

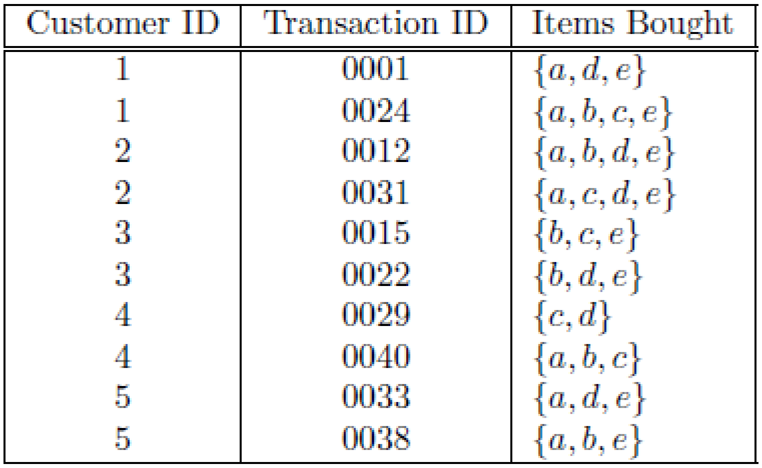
TABLE OF CONTENTS

PROBLEMS:

2. Consider the data set shown in Table 6.1.

(a) Compute the support for itemsets {e}, {b, d}, and {b, d, e} by treating each transaction ID as a market basket.

Table 6.1. Example of market basket transactions.



Customer ID	Transaction ID	Items Bought
1	0001	$\{a, d, e\}$
1	0024	$\{a, b, c, e\}$
2	0012	$\{a, b, d, e\}$
2	0031	$\{a, c, d, e\}$
3	0015	$\{b, c, e\}$
3	0022	$\{b, d, e\}$
4	0029	$\{c, d\}$
4	0040	$\{a, b, c\}$
5	0033	$\{a, d, e\}$
5	0038	$\{a, b, e\}$

Solution:

Support is calculated by = Number of transactions in X / Total Number of transactions.

S(e) = Transaction ID's (0001, 0024, 0012, 0031, 0015, 0022, 0033, 0038) / Total number of transactions.

S (e) = 8 / 10

S (e) = 0.8

Similarly, Support of {b and d} and {b,d,e} can be calculated

S (b,d) = 2/10

S {b,d } = 0.2

S {b, d, e} = 2/10

S {b, d, e} = 0.2

2. (b) Use the results in part (a) to compute the confidence for the association rules {b, d} –→ {e} and {e} –→ {b, d}. Is confidence a symmetric measure?

Solution:

Confidence is calculated by = freq (X,Y) / freq (X)

C {b,d ---- > e} = 0.2 / 0.2

= 1

C {b,d ---- > e} = 100%

C { e --- > b,d} = 0.2 / 0.8

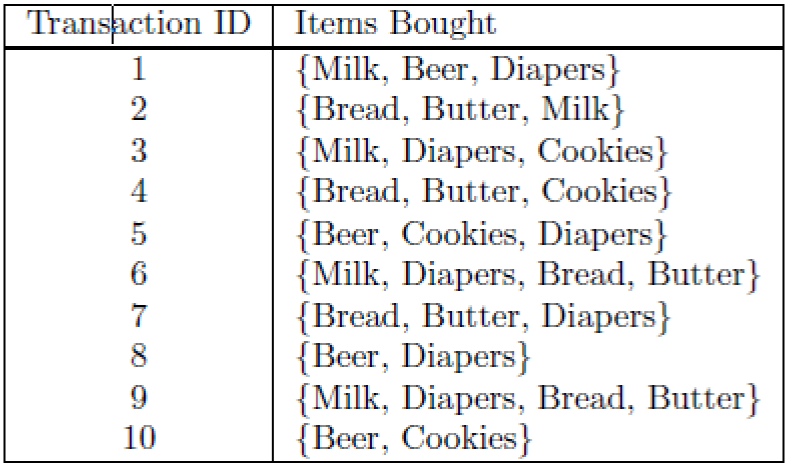
= 0.25

C { e --- > b,d} = 25 %

Confidence is not a symmetric measure.

6. Consider the below table.

Table 6.2. Market basket transactions.



Transaction ID	Items Bought
1	{Milk, Beer, Diapers}
2	{Bread, Butter, Milk}
3	{Milk, Diapers, Cookies}
4	{Bread, Butter, Cookies}
5	{Beer, Cookies, Diapers}
6	{Milk, Diapers, Bread, Butter}
7	{Bread, Butter, Diapers}
8	{Beer, Diapers}
9	{Milk, Diapers, Bread, Butter}
10	{Beer, Cookies}

(a) What is the maximum number of association rules that can be extracted from this data (including rules that have zero support)?

Solution:

There are 6 items in the Dataset (Market basket transactions) = Bread, Beer, Milk, Diapers, Cookies, Butter.

Hence d = 6

Total number of association rules $R = 3^d - 2^{d+1} + 1$

$R = 3^6 - 2^{6+1} + 1$

$R = 729 - 128 + 1$

$R = 602$ rules

c.) Write an expression for the maximum number of size-3 itemsets that can be derived from this data set.

Solution:

There are totally 6 items out of which 3 itemsets can be derived.

Hence combination of items is represented by – $6C_3 = (6*5*4) /(1*2*3)$

$6C_3 = 20.$

d.) Find an itemset (of size 2 or larger) that has the largest support?

Solution:

Milk and Diapers = 4
Cookies and Beer = 2
Beer and Diapers = 3
Bread and Butter = 5

Hence, an itemset (size 2 or larger) that has the largest support = { Bread, Butter }

e.) Find a pair of items, a and b, such that the rules {a} –→ {b} and {b} –→ {a} have the same confidence.

Solution:

Consider Milk and Diapers.

Support of both the above items = (Support of (Milk) U Support of Diapers) / Total Number of transactions
= 4/6
= 0.66
Confidence gained for both items = Support of Milk and Diapers / Support of Milk
= 4/5
C {Milk and Diapers} = 0.8 = 80%

For Diapers and Milk

Support of both the above items = 4/6
= 0.66
Confidence gained for both items = Support of Milk and Diapers / Support of Diapers
= 4/7
C {Diapers and Milk} = 0.57 = 57%

Now consider Bread and Butter:

Support of both the above items = Support of Bread and Butter / Number of transactions
= 5/6
= 0.83
Confidence gained for both items = Support of Bread and Butter / Support of Bread
= 5/5
C {Bread and Butter} = 1= 100%

For Butter and Bread:

Support for both above items = 5/6
= 0.83
Confidence for both items = Support of Butter and Bread / Support of Butter
= 5/5
C {Butter and Bread} = 1 = 100%

Now consider Beers and Cookies:

Support of both the above items = Support of Beers and cookies / Number of transactions
= 2/6
= 0.33
Confidence gained for both items = Support of Beers and Cookies / Support of Beers
= 2/4

C {Beers and Cookies} = 0.5 = 50%

For Cookies and Beers:

Support for both above items = 2/6
= 0.33
Confidence for both items = Support of Cookies and Beers / Support of Cookies
= 2/4

C {Cookies and Beers} = 0.5 = 50%

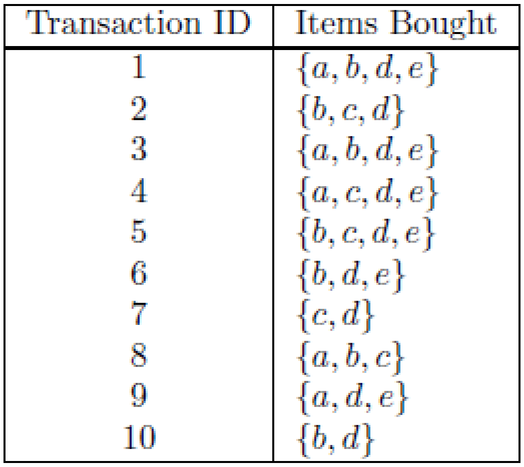
From the above analysis, it is pretty clear that

C{Beers and cookies} = C{Cookies and Beers} and

C{Bread and Butter} = C{Butter and Bread}

8. The *Apriori* algorithm uses a generate-and-count strategy for deriving frequent itemsets. Candidate itemsets of size *k*+1 are created by joining a pair of frequent itemsets of size *k* (this is known as the candidate generation step). A candidate is discarded if any one of its subsets is found to be infrequent during the candidate pruning step. Suppose the *Apriori* algorithm is applied to the data set shown in Table 6.3 with *minsup* = 30%, i.e., any itemset occurring in less than 3 transactions is considered to be infrequent.

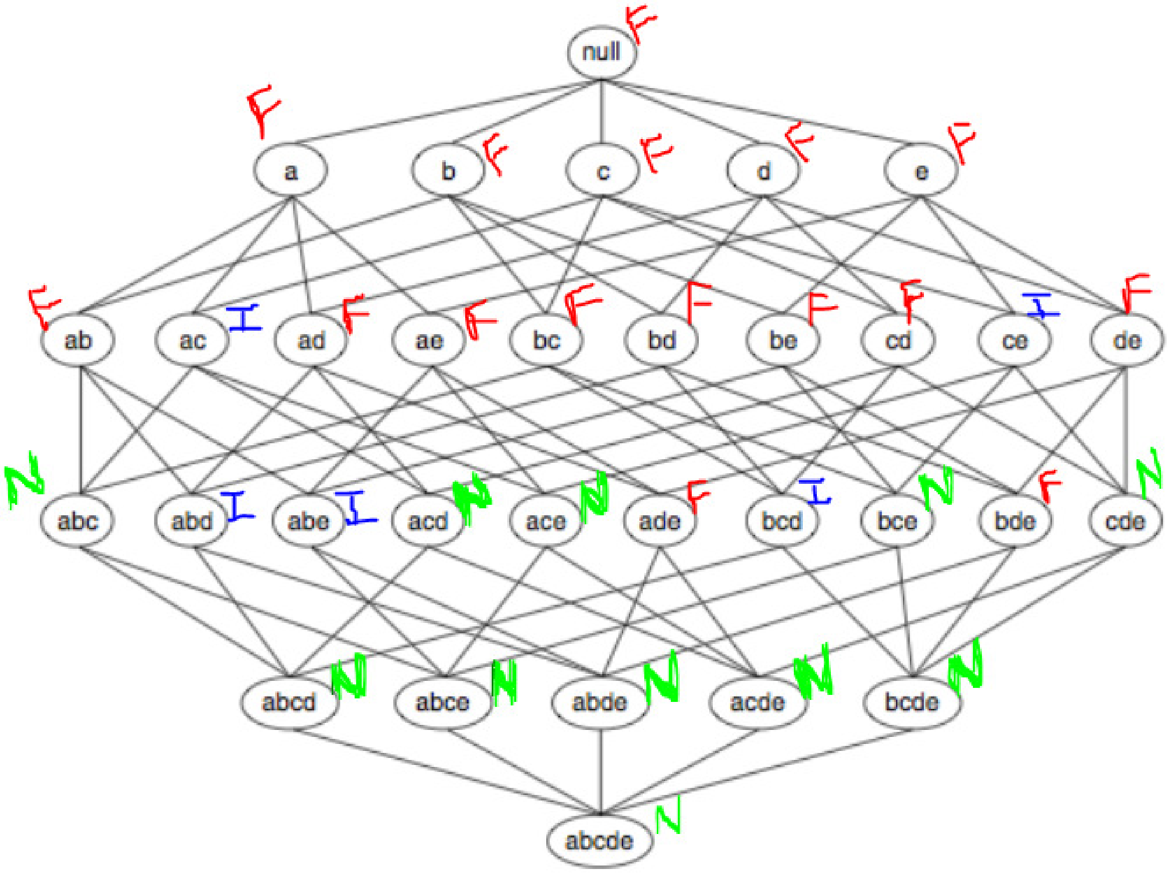
Table 6.3. Example of market basket transactions.



Transaction ID	Items Bought
1	$\{a, b, d, e\}$
2	$\{b, c, d\}$
3	$\{a, b, d, e\}$
4	$\{a, c, d, e\}$
5	$\{b, c, d, e\}$
6	$\{b, d, e\}$
7	$\{c, d\}$
8	$\{a, b, c\}$
9	$\{a, d, e\}$
10	$\{b, d\}$

a.) Draw an itemset lattice representing the data set given in Table 6.3. Label each node in the lattice with the following letter(s):

Solution:



- N: If the itemset is not considered to be a candidate itemset by the Apriori algorithm. There are two reasons for an itemset not to be considered as a candidate itemset: (1) it is not generated at all during the candidate generation step, or (2) it is generated during the candidate generation step but is subsequently removed during the candidate pruning step because one of its subsets is found to be infrequent.
- F: If the candidate itemset is found to be frequent by the Apriori algorithm.
- I: If the candidate itemset is found to be infrequent after support counting.

(b) What is the percentage of frequent itemsets (with respect to all itemsets in the lattice)?

Percentage of frequent itemsets = No.of freq items / Total No. of items
= 16/32
= 50.0%

This is including the null set too.

c.) What is the pruning ratio of the *Apriori* algorithm on this data set? (Pruning ratio is defined as the percentage of itemsets not considered to be a candidate because (1) they are not generated during candidate generation or (2) they are pruned during the candidate pruning step).

Solution:

Pruning ratio = Nodes that are of N / Total Number of Nodes

Here, count of N = 11,

Total Number of nodes = 32

Therefore, the pruning ratio becomes = 11/32
= 34.4%

d.) What is the false alarm rate (i.e, percentage of candidate itemsets that are found to be infrequent after performing support counting)?

Solution:

False alarm rate is the ratio of I to the total number of itemsets.

Since the count of I = 5,

The false alarm rate is 5/32 = 15.6%.

9. The *Apriori* algorithm uses a hash tree data structure to efficiently count the support of candidate itemsets. Consider the hash tree for candidate 3-itemsets shown in Figure 6.2.

(a) Given a transaction that contains items *{*1*,* 3*,* 4*,* 5*,* 8*}*, which of the hash tree leaf nodes will be visited when finding the candidates of the transaction?

Solution:

Traversing the Hash tree in the order of transaction

L1 = {1 + (3458)} -> (14 + (58)

L3 = {1 + (3458) -> {15+{8}} => {158}

L5 = {1 + (3458) -> { 13 + 458}

L9 = {3 + {458)} -> {34 + {58}}

L11 = {3 + {458}} -> {35 + (8)}

Hence the leaf nodes visited are L1, L3, L5, L9, L11.

(b) Use the visited leaf nodes in part (b) to determine the candidate itemsets that are contained in the transaction *{*1*,* 3*,* 4*,* 5*,* 8*}*.

The candidates contained in the transaction are *{*1*,* 4*,* 5*}*, *{*1*,* 5*,* 8*}*, and *{*4*,* 5*,* 8*}*.

ASSOCIATION RULE MINING:

- Analyzing and predicting customer behavior.
- Clubbing things that are bought together frequently (say for in a supermarket) will also tend to buy another product.
- Association rule will be helpful in predicting the product customer would buy and thus increasing the sales of a market business.

Example1 for a normal association rule mining:

Bread === > Butter

A person who buys bread will tend to buy butter.

ASSOCIATION RULE MINING INVOLVING SUPPORT AND CONFIDENCE.

Let us consider an example in a supermarket a person buying products and making transactions.

Bread === > Butter (20%, 25%)

Where,

Bread is referred as precedent, -- > Person buying this product from the whole lot of items in the market.

Butter is consequent. ----- > Person who bought bread also tends to buy butter.

20% is support – which denotes the probability that contains both Bread and butter.

45% is confidence – which denotes the probability that a transaction containing A also contains B.

Example2:

Consider supermarket with transactions = 200

Bread – 40 (40 of the transactions among 200 transactions are for bread).

Hence – (40/200) * 100 = 20% ======= > Support

Out of these 40 transactions, people who buy bread will also tend to buy butter is confidence.

Let's take Butter = 10 transactions (amongst people who buy bread – 40 transactions)

So, (10/40) * 100 = 25% ======= > Confidence.

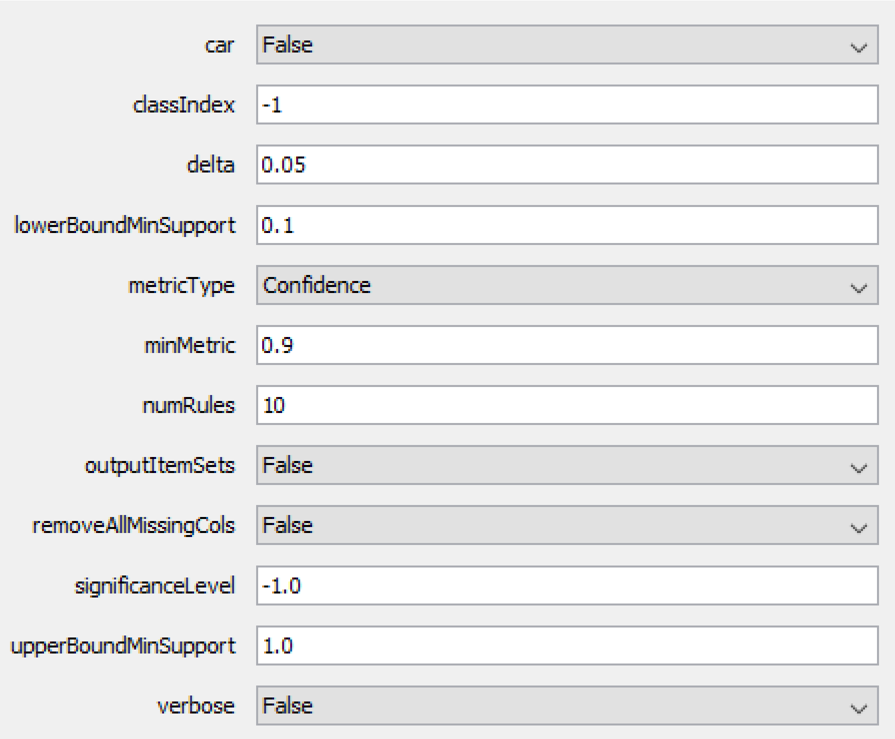
ASSOCIATION RULE MINING AND THEIR ALGORITHMS.

There are top 4 algorithms in association rule mining to determine the combination of products to predict the customer behavior.

1. Apriori algorithm.
2. Filtered Associator.
3. FP growth.
4. Predictive Apriori
5. Tertius

APRIORI ALGORITHM:

Default values of this algorithm:



car	False
classIndex	-1
delta	0.05
lowerBoundMinSupport	0.1
metricType	Confidence
minMetric	0.9
numRules	10
outputItemSets	False
removeAllMissingCols	False
significanceLevel	-1.0
upperBoundMinSupport	1.0
verbose	False

OutputItemsets: If this is enabled, then the itemsets from the whole transactions are listed in the console/output section.

numRules: This is a default important attribute which decides how many rules to be formed for a given dataset.

removeAllMissingcols: This parameter if enabled, then the association rule mining will remove all the columns that has missing values in the transactions.

Note: Not all Datasets will be completely occupied without any missing values. Hence for most of the cases, removeAllMissingCols should be disabled. This can be enabled when there are no transactions for a particular column (product) by any user.

Metrictype: Confidence – will throw the percentage of confidence level achieved based on the best rules formed.

Delta: Delta is arrived by which minimum support is decreased by each iteration. (default=0.05)

Upperbound and lowerbound min support: Lower and upper bounds of a minimum support. (say a product which is bought among total items in the market).

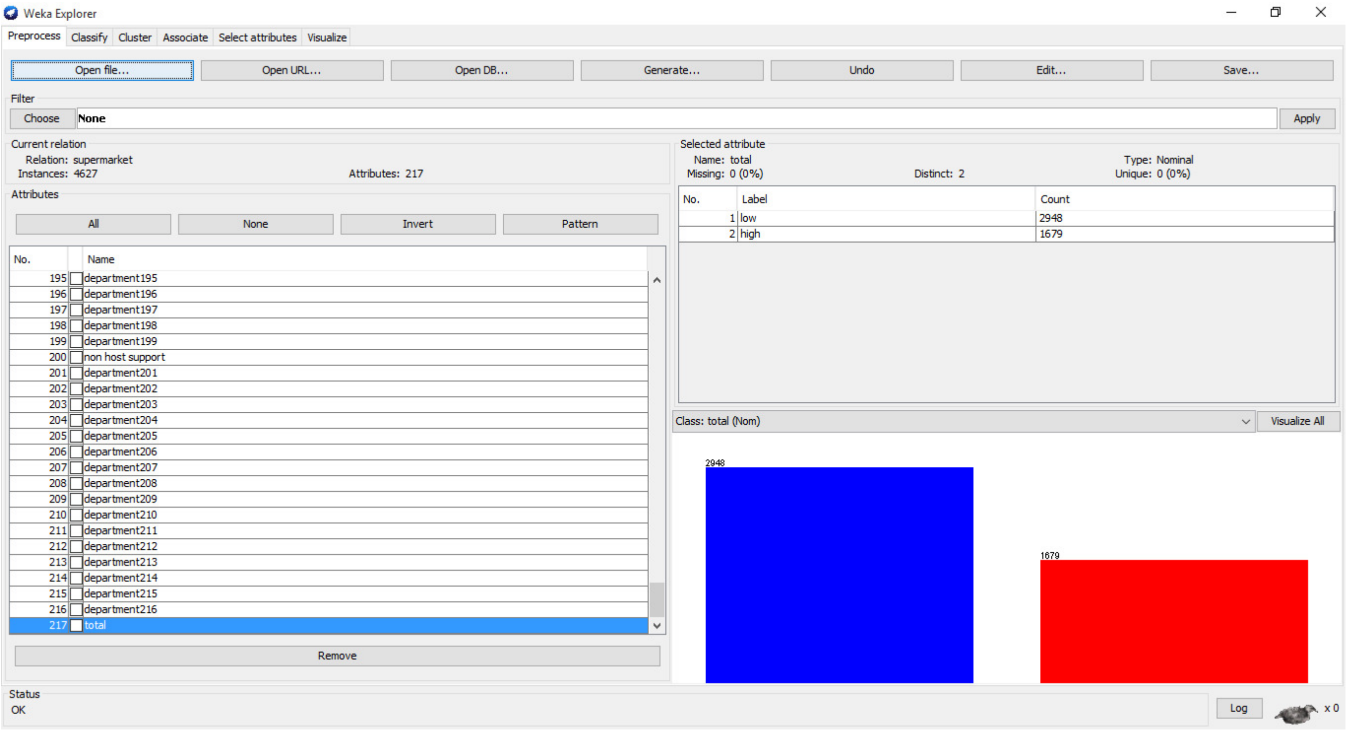
Support, Confidence, Lift values are calculated using:

- Support = Number of transactions which contains the itemset X / Total number of transactions.
- Confidence = (Supply of X u Y) / supply of X --- > Both items X and Y bought together / Items X bought.
- Lift = (Supply of X u Y) / supply(X)* suppuly(Y)
- Conviction Conf(X->Y) = (1-supp(Y)) / (1- conf(X ->Y))

Algorithm:

1. Minimum support is applied to find the frequent itemsets.
2. Using these frequent itemsets and minimum confidence constraint, rules are thus formed.

SUPERMARKET DATASET:



Filter
None

Current relation
Relation: supermarket
Instances: 4627
Attributes: 217

Selected attribute
Name: total
Type: Nominal
Missing: 0 (0%)
Distinct: 2
Unique: 0 (0%)

No.	Label	Count
1	low	2948
2	high	1679

Attributes

No.		Name
195	□	department195
196	□	department196
197	□	department197
198	□	department198
199	□	department199
200	□	non host support
201	□	department201
202	□	department202
203	□	department203
204	□	department204
205	□	department205
206	□	department206
207	□	department207
208	□	department208
209	□	department209
210	□	department210
211	□	department211
212	□	department212
213	□	department213
214	□	department214
215	□	department215
216	□	department216
217	□	total

Class: total (Nom)

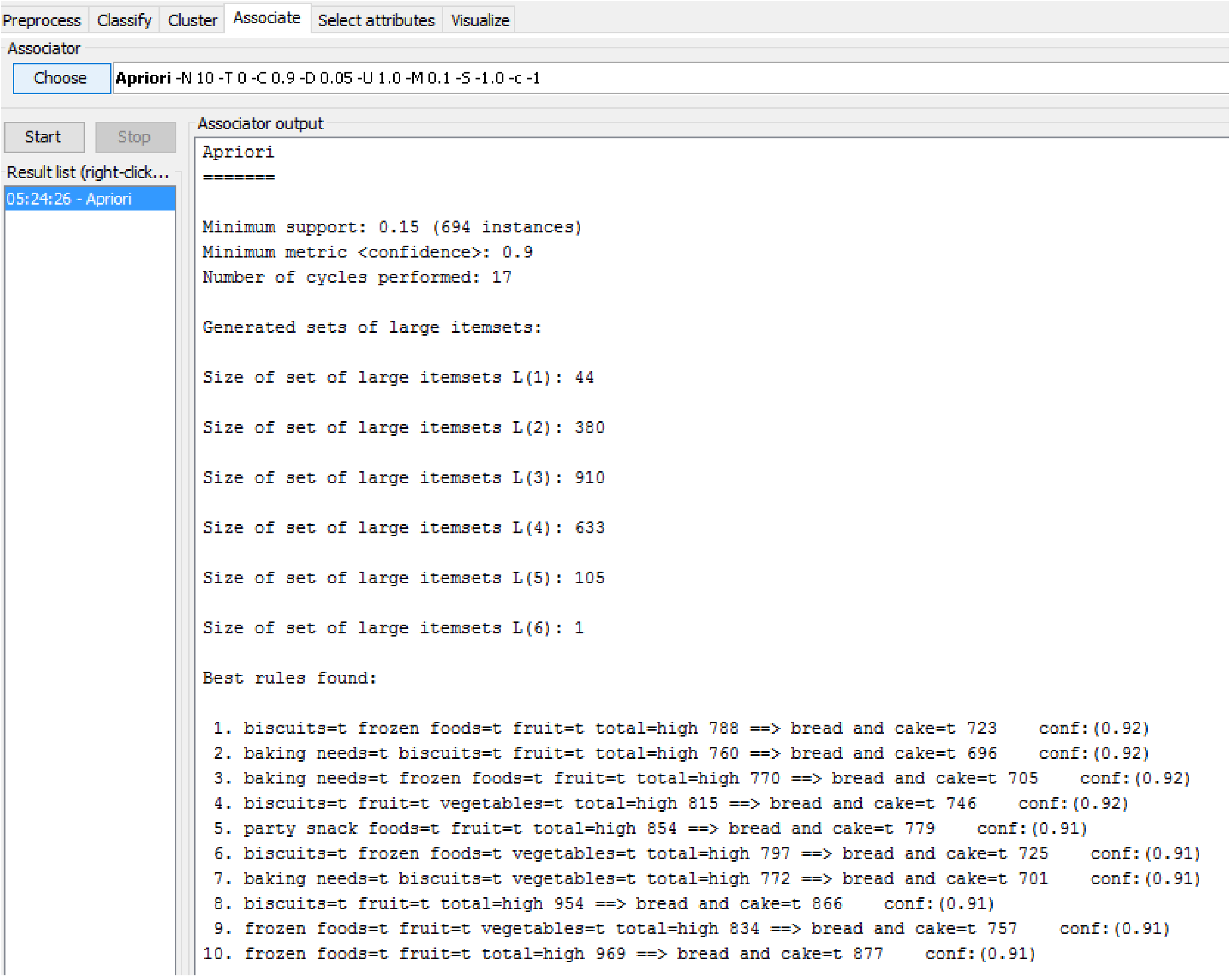
Status
OK

1. Inference from Supermarket Dataset:

Dataset name	Total number of attributes	Type of the attribute	Total number of instances	If Class attribute – specify the name of the Class attributes.
Supermarket	217	Nominal	4627	Low – 2948 High - 1679

Test case 1:- Supermarket Dataset is made to run against Apriori algorithm.

- **When Supermarket Dataset is made to run under Association rule mining with Apriori algorithm having default values set.**



Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

```
Apriori
=======

Minimum support: 0.15 (694 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 17

Generated sets of large itemsets:

Size of set of large itemsets L(1): 44

Size of set of large itemsets L(2): 380

Size of set of large itemsets L(3): 910

Size of set of large itemsets L(4): 633

Size of set of large itemsets L(5): 105

Size of set of large itemsets L(6): 1

Best rules found:

 1. biscuits=t frozen foods=t fruit=t total=high 788 ==> bread and cake=t 723    conf:(0.92)
 2. baking needs=t biscuits=t fruit=t total=high 760 ==> bread and cake=t 696    conf:(0.92)
 3. baking needs=t frozen foods=t fruit=t total=high 770 ==> bread and cake=t 705    conf:(0.92)
 4. biscuits=t fruit=t vegetables=t total=high 815 ==> bread and cake=t 746    conf:(0.92)
 5. party snack foods=t fruit=t total=high 854 ==> bread and cake=t 779    conf:(0.91)
 6. biscuits=t frozen foods=t vegetables=t total=high 797 ==> bread and cake=t 725    conf:(0.91)
 7. baking needs=t biscuits=t vegetables=t total=high 772 ==> bread and cake=t 701    conf:(0.91)
 8. biscuits=t fruit=t total=high 954 ==> bread and cake=t 866    conf:(0.91)
 9. frozen foods=t fruit=t vegetables=t total=high 834 ==> bread and cake=t 757    conf:(0.91)
10. frozen foods=t fruit=t total=high 969 ==> bread and cake=t 877    conf:(0.91)
```

Inference from the above for the 1st rule thus formed:

- Minimum support is given by 0.15 with the help of Number of cycles performed.

- In our case, number of cycles performed = 17, which means Apriori was actually run 17 times to generate a set of rules with 17 different values for the minimum support specified.

- People who buy biscuits, frozen foods, fruit and having total transactions of 788 will also likely to buy bread and cake in their transaction. The confidence achieved in this outcome is 92%.

- Let X = biscuits, frozen foods, fruit. Instances = 788

- Y = Bread and cake. Instances = 723

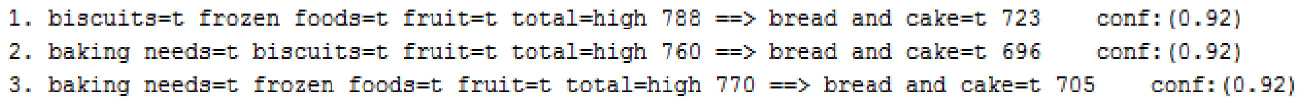
- Then, confidence can be calculated to be 92%. This implies, 92% of the people who buy "X" products will surely buy "Y" products.

Future aspect: The store should maintain sufficient amount of "Y" products if "X" products are maintained.

- Since our upperboundMinsupport is 1, by default and delta is 0.05 (default), we have apriori algorithm run for 17 cycles, it is well determined by the formula === > 1 – 17(0.05) = 0.15 (Minimum support value).

- Hence 0.15 is the minimum support value reached after 17 iterations are performed.

Inference for the 2nd rule thus formed by running the apriori algorithm:



```
 1. biscuits=t frozen foods=t fruit=t total=high 788 ==> bread and cake=t 723    conf:(0.92)
 2. baking needs=t biscuits=t fruit=t total=high 760 ==> bread and cake=t 696    conf:(0.92)
 3. baking needs=t frozen foods=t fruit=t total=high 770 ==> bread and cake=t 705    conf:(0.92)
```

- Minimum support is given by 0.15 with the help of Number of cycles performed.

- In our case, number of cycles performed = 17, which means Apriori was actually run 17 times to generate a set of rules with 17 different values for the minimum support specified.

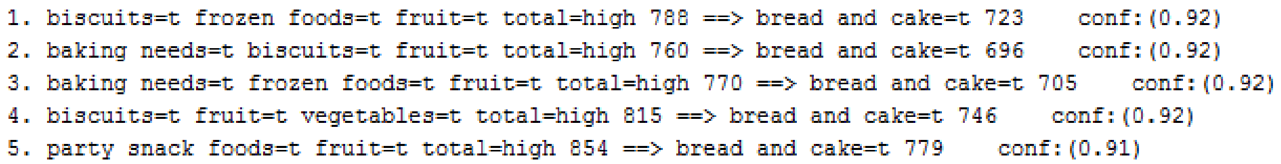
- People who buy baking needs, biscuits, fruit, and Total instances being 760, will also likely to buy bread and cake in their transaction. The confidence achieved in this outcome is 92%.

- Let X = baking needs, biscuits, fruit. Instances = 760, **and** Y = Bread and cake. Instances = 696

- Then, confidence can be calculated to be 92%. This implies, 92% of the people who buy "X" products will surely buy "Y" products.

Future aspect: The store should maintain sufficient amount of "Y" products if "X" products are maintained.

Inference for the 5th rule thus formed by running the Apriori algorithm:



```
 1. biscuits=t frozen foods=t fruit=t total=high 788 ==> bread and cake=t 723    conf:(0.92)
 2. baking needs=t biscuits=t fruit=t total=high 760 ==> bread and cake=t 696    conf:(0.92)
 3. baking needs=t frozen foods=t fruit=t total=high 770 ==> bread and cake=t 705    conf:(0.92)
 4. biscuits=t fruit=t vegetables=t total=high 815 ==> bread and cake=t 746    conf:(0.92)
 5. party snack foods=t fruit=t total=high 854 ==> bread and cake=t 779    conf:(0.91)
```

- People who buy party snack foods and fruit, and Total instances being 854 will also likely to buy bread and cake in their transaction. The confidence achieved in this outcome is 92%.

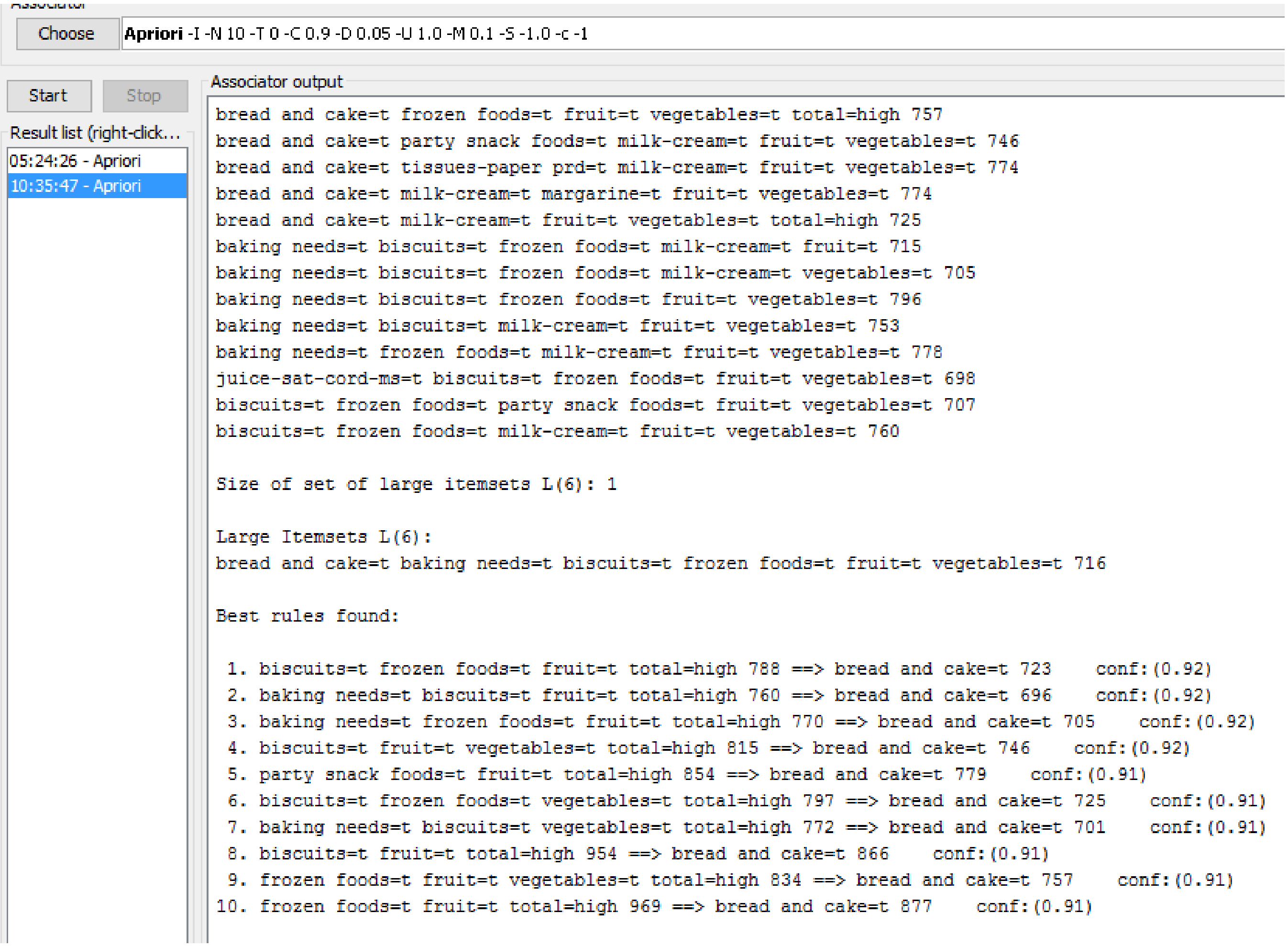
- Let X = baking needs, biscuits, fruit. Instances = 854, **and** Y = Bread and cake. Instances = 779

- Then, confidence can be calculated to be 91%. This implies, 91% of the people who buy "X" products will surely buy "Y" products.

We could see that from the 5th rule, the percentage of confidence starts to decrease as the best rule with high confidence level cannot be calculated further.

Test case 2: - To display the Large itemsets and number of instances in the output

- **By enabling Outputitemsets in Apriori.**



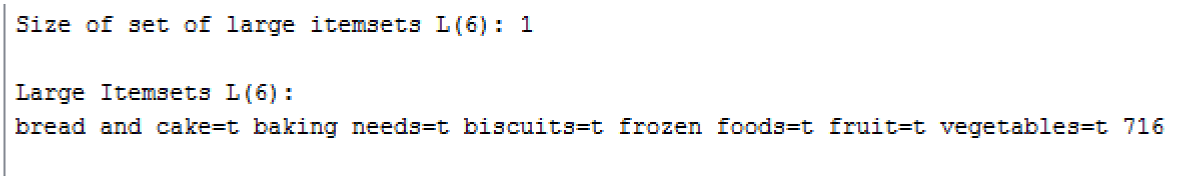
Apriori -I -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

```
bread and cake=t frozen foods=t fruit=t vegetables=t total=high 757
bread and cake=t party snack foods=t milk-cream=t fruit=t vegetables=t 746
bread and cake=t tissues-paper prd=t milk-cream=t fruit=t vegetables=t 774
bread and cake=t milk-cream=t margarine=t fruit=t vegetables=t 774
bread and cake=t milk-cream=t fruit=t vegetables=t total=high 725
baking needs=t biscuits=t frozen foods=t milk-cream=t fruit=t 715
baking needs=t biscuits=t frozen foods=t milk-cream=t vegetables=t 705
baking needs=t biscuits=t frozen foods=t fruit=t vegetables=t 796
baking needs=t biscuits=t milk-cream=t fruit=t vegetables=t 753
baking needs=t frozen foods=t milk-cream=t fruit=t vegetables=t 778
juice-sat-cord-ms=t biscuits=t frozen foods=t fruit=t vegetables=t 698
biscuits=t frozen foods=t party snack foods=t fruit=t vegetables=t 707
biscuits=t frozen foods=t milk-cream=t fruit=t vegetables=t 760

Size of set of large itemsets L(6): 1

Large Itemsets L(6):
bread and cake=t baking needs=t biscuits=t frozen foods=t fruit=t vegetables=t 716

Best rules found:

 1. biscuits=t frozen foods=t fruit=t total=high 788 ==> bread and cake=t 723    conf:(0.92)
 2. baking needs=t biscuits=t fruit=t total=high 760 ==> bread and cake=t 696    conf:(0.92)
 3. baking needs=t frozen foods=t fruit=t total=high 770 ==> bread and cake=t 705    conf:(0.92)
 4. biscuits=t fruit=t vegetables=t total=high 815 ==> bread and cake=t 746    conf:(0.92)
 5. party snack foods=t fruit=t total=high 854 ==> bread and cake=t 779    conf:(0.91)
 6. biscuits=t frozen foods=t vegetables=t total=high 797 ==> bread and cake=t 725    conf:(0.91)
 7. baking needs=t biscuits=t vegetables=t total=high 772 ==> bread and cake=t 701    conf:(0.91)
 8. biscuits=t fruit=t total=high 954 ==> bread and cake=t 866    conf:(0.91)
 9. frozen foods=t fruit=t vegetables=t total=high 834 ==> bread and cake=t 757    conf:(0.91)
10. frozen foods=t fruit=t total=high 969 ==> bread and cake=t 877    conf:(0.91)
```

Inference from the above screenshot:

Size of the Large itemsets which were grouped together are displayed in the output.

For instance,



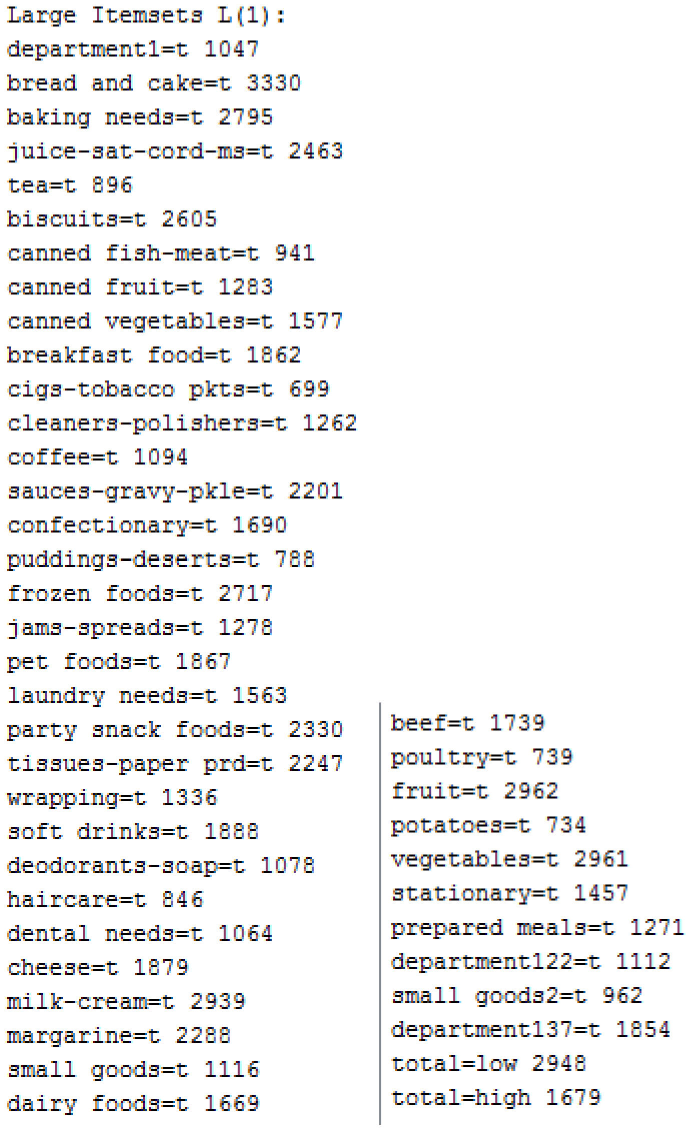
```
Size of set of large itemsets L(6): 1

Large Itemsets L(6):
bread and cake=t baking needs=t biscuits=t frozen foods=t fruit=t vegetables=t 716
```

Here, size of a set of large itemsets L6 = 1 where there is only 1 instance being reported with a particular combination and there are 6 items grouped together.

This implies, that out of the total transactions, when a person buys bread and cake, baking needs, biscuits, frozen food, fruit, and vegetables, they have been reported 716 times, -- > 716 instances.

Test case 3:-Default parameters – important attributes

- **Important attributes that are considered for finding the frequent itemsets via Apriori principle.**



```
Large Itemsets L(1):
department1=t 1047
bread and cake=t 3330
baking needs=t 2795
juice-sat-cord-ms=t 2463
tea=t 896
biscuits=t 2605
canned fish-meat=t 941
canned fruit=t 1283
canned vegetables=t 1577
breakfast food=t 1862
cigs-tobacco pkts=t 699
cleaners-polishers=t 1262
coffee=t 1094
sauces-gravy-pkle=t 2201
confectionary=t 1690
puddings-deserts=t 788
frozen foods=t 2717
jams-spreads=t 1278
pet foods=t 1867
laundry needs=t 1563
party snack foods=t 2330
tissues-paper prd=t 2247
wrapping=t 1336
soft drinks=t 1888
deodorants-soap=t 1078
haircare=t 846
dental needs=t 1064
cheese=t 1879
milk-cream=t 2939
margarine=t 2288
small goods=t 1116
dairy foods=t 1669
```

```
beef=t 1739
poultry=t 739
fruit=t 2962
potatoes=t 734
vegetables=t 2961
stationary=t 1457
prepared meals=t 1271
department122=t 1112
small goods2=t 962
department137=t 1854
total=low 2948
total=high 1679
```

Inference from the above:

The above screenshot shows the number of instances when each attribute is counted into the transaction.

Taking this sequence in consideration, Apriori principle groups the products together to find the frequent itemsets and the future predictions of the products to be bought.

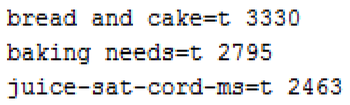
In our case, Let us consider I as the itemset containing number of itemsets.

I = {i1, i2, i3, i4……in}

All set of transactions made in the DB == > T

T = {t1, t2, t3, t4…… tn}

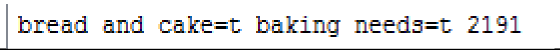
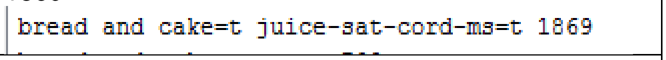
Let us take first 3 itemsets (consider that these 3 are frequent itemsets) and see how grouping works under Apriori principle.



```
bread and cake=t 3330
baking needs=t 2795
juice-sat-cord-ms=t 2463
```

From the above, it is shown that bread and cake have been reported 3330 times, baking needs reported 2795 transactions, and juice-sat-cord-ms have been reported 2463 transactions out of the total transactions (4627 – Instances total).

Applying Apriori Principle:

Items grouped	**Instances / Number of transactions**
Bread and cake=t & baking needs=t	2191  `bread and cake=t baking needs=t 2191`
Bread and cake=t & juice-sat-cord-ms=t	1869  `bread and cake=t juice-sat-cord-ms=t 1869`
Baking needs=t & juice-sat-cord-ms=t	1619 `baking needs=t juice-sat-cord-ms=t 1619`

Inference from the above:

We can conclude that 2191 instances/transactions are made when both items Bread and cake + Baking needs are bought together.

Hence calculating support value for this would be as below.

Bread and cake (X) = 3330

Baking needs (Y) = 2795

Total number of transactions in the 3 products above

= Bread and cake transactions + baking needs transactions + juice-sat-cord-ms transactions

= 3330 + 2795 + 2463

= 8588

Support = Number of transactions of (X) / total number of transactions.

= 3330 / 8588

= 38% support level achieved for Bread and cake transactions made.

Confidence when both Bread and cake + baking needs are bought together.

Confidence = 2795 / 3330

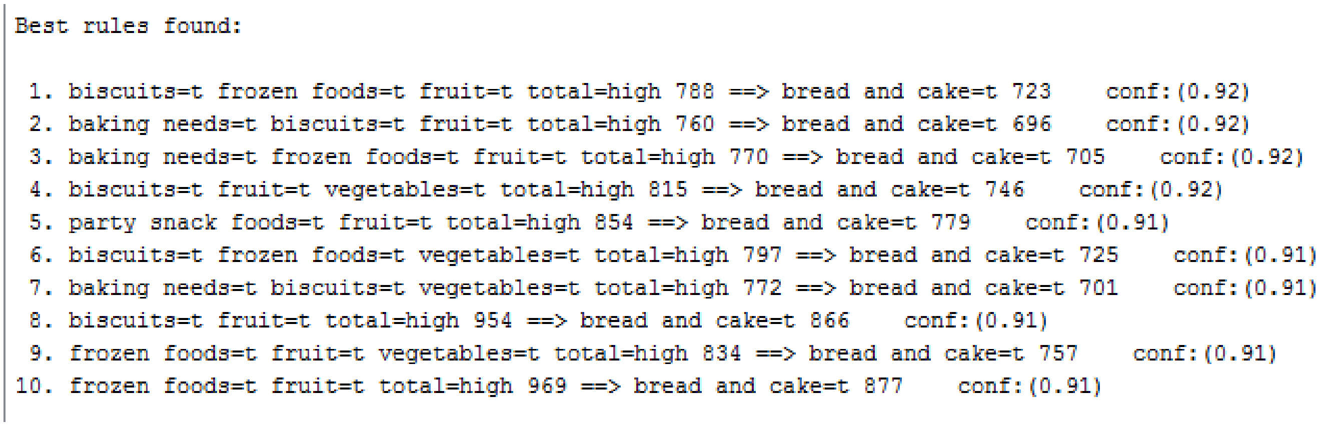
= 83 % confidence level is achieved for both items to be bought together.

Thus a rule is formed for our example for the above support and confidence calculations,

bread and cake=t 3330 = > baking needs=t 2795 conf:(0.83)

Likewise, the confidence level is achieved for the entire Dataset – Supermarket by grouping itemsets together and finding the maximum confidence percentage along with the best rules thus formed with which the market can grow its business.

10 best rules formed for this Dataset – Supermarket:



```
Best rules found:

 1. biscuits=t frozen foods=t fruit=t total=high 788 ==> bread and cake=t 723    conf:(0.92)
 2. baking needs=t biscuits=t fruit=t total=high 760 ==> bread and cake=t 696    conf:(0.92)
 3. baking needs=t frozen foods=t fruit=t total=high 770 ==> bread and cake=t 705    conf:(0.92)
 4. biscuits=t fruit=t vegetables=t total=high 815 ==> bread and cake=t 746    conf:(0.92)
 5. party snack foods=t fruit=t total=high 854 ==> bread and cake=t 779    conf:(0.91)
 6. biscuits=t frozen foods=t vegetables=t total=high 797 ==> bread and cake=t 725    conf:(0.91)
 7. baking needs=t biscuits=t vegetables=t total=high 772 ==> bread and cake=t 701    conf:(0.91)
 8. biscuits=t fruit=t total=high 954 ==> bread and cake=t 866    conf:(0.91)
 9. frozen foods=t fruit=t vegetables=t total=high 834 ==> bread and cake=t 757    conf:(0.91)
10. frozen foods=t fruit=t total=high 969 ==> bread and cake=t 877    conf:(0.91)
```

Inference from the above rule thus formed:

- People who buy this combination - biscuits, frozen foods, fruits, are likely to buy bread and cake too and the percentage of confidence is 92% proved.

- Likewise 10 such rules are formed to let the vendor know the frequency of items bought and predict the items that will be bought from those items bought together often.

- Here, in our example, we have asked the Apriori algorithm to find first 10 best rules from this dataset as numRules = 10 is the basic default value for Apriori principle.

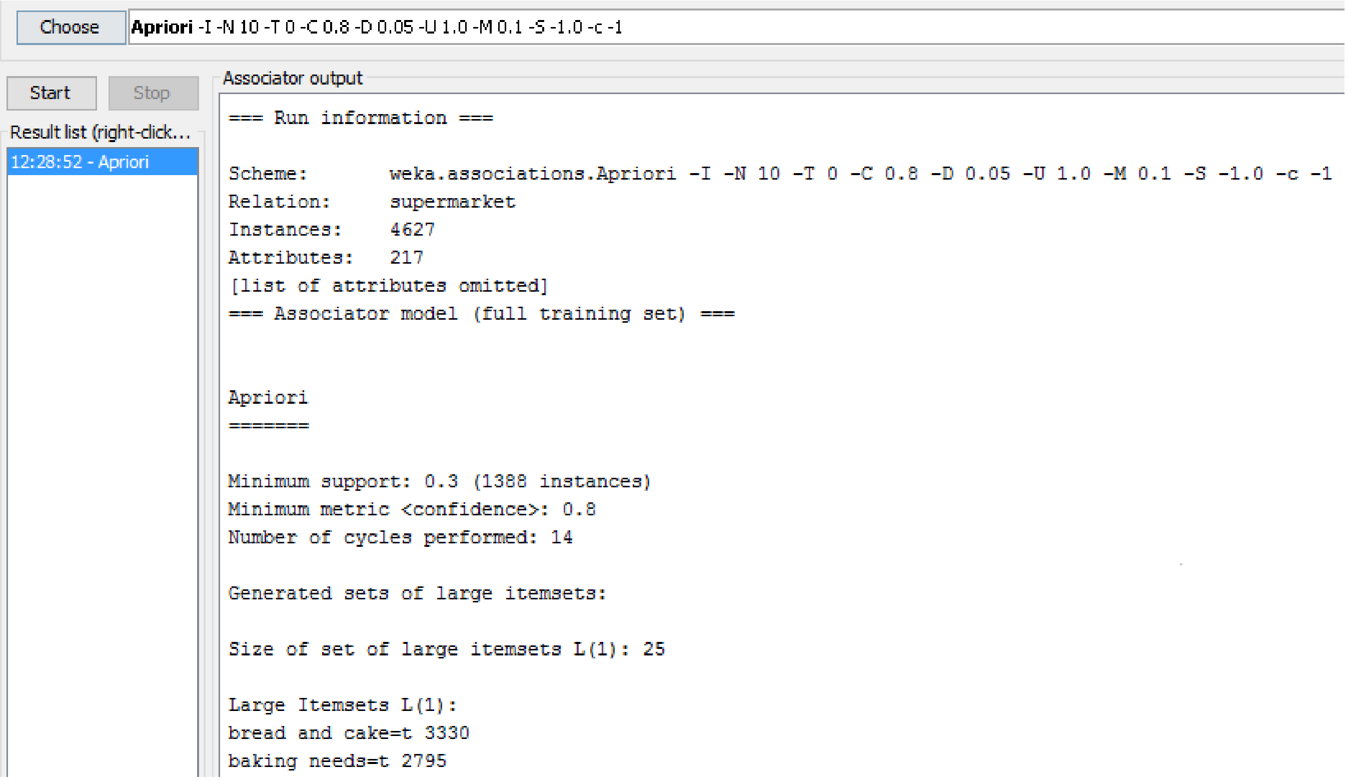
- Minimum metric default value = 0.9 (90% of confidence achieved). Here, we have specified that minimum confidence of a formed rule should be 90% and hence the rules thus formed based on the Apriori algorithm should be greater than or equal to 90% confidence.

Apriori Principle for Supermarket with its default values:

Thus, Apriori principle for its default value for the Dataset Supermarket reduces the minimum support until it finds the required number of rules with the specified minimum confidence level.

Test case 4: - Lower minmetric (from 0.9 to 0.8) – Confidence level lowered

- **When changing minmetric (from 0.9 to 0.8) -- > Having minimum metric score and asking Apriori to find rules that best matches 80% of confidence level at the minimum.**



Apriori -I -N 10 -T 0 -C 0.8 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 0.8 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.3 (1388 instances)
Minimum metric <confidence>: 0.8
Number of cycles performed: 14

Generated sets of large itemsets:

Size of set of large itemsets L(1): 25

Large Itemsets L(1):
bread and cake=t 3330
baking needs=t 2795
```

Inference from the above:

- Weka runs Apriori principle several times. It starts at upperboundsupport (usually 100%) and decreases by delta at each iteration (usually 5% each time) and stops when numRules = 10 are reached.
- Hence Number of cycles Apriori has run in the above case = 14 cycles.
- Delta value here = 0.05
- Upperbound support = 1
- Hence, Minimum support = 1 – 14(0.05)

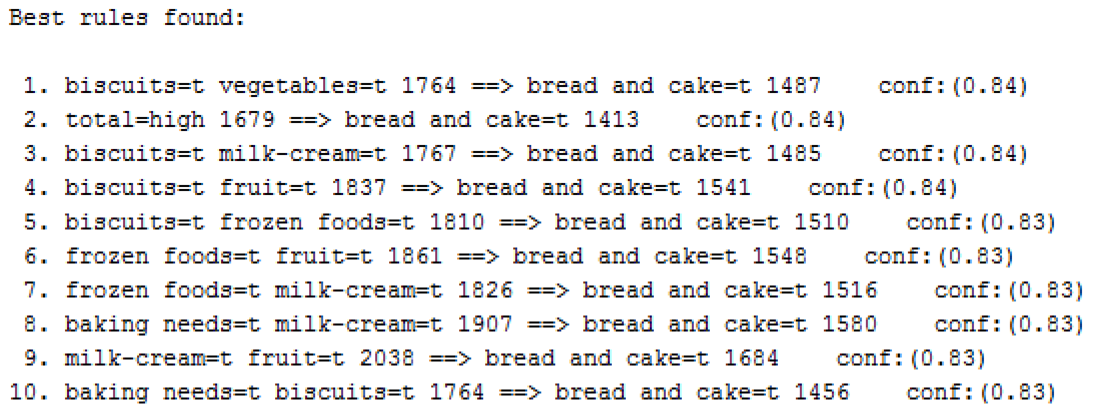
  = 0.3 (30% support)

  = 30% support of 4627 total instances

  = 1388 instances

- Minimum metric confidence level is specified = 0.08 (80% of confidence level at the minimum and rules are formed based on this percentage of confidence).

Thus 10 Best rules are formed (as the numRules = 10):



```
Best rules found:

 1. biscuits=t vegetables=t 1764 ==> bread and cake=t 1487    conf:(0.84)
 2. total=high 1679 ==> bread and cake=t 1413    conf:(0.84)
 3. biscuits=t milk-cream=t 1767 ==> bread and cake=t 1485    conf:(0.84)
 4. biscuits=t fruit=t 1837 ==> bread and cake=t 1541    conf:(0.84)
 5. biscuits=t frozen foods=t 1810 ==> bread and cake=t 1510    conf:(0.83)
 6. frozen foods=t fruit=t 1861 ==> bread and cake=t 1548    conf:(0.83)
 7. frozen foods=t milk-cream=t 1826 ==> bread and cake=t 1516    conf:(0.83)
 8. baking needs=t milk-cream=t 1907 ==> bread and cake=t 1580    conf:(0.83)
 9. milk-cream=t fruit=t 2038 ==> bread and cake=t 1684    conf:(0.83)
10. baking needs=t biscuits=t 1764 ==> bread and cake=t 1456    conf:(0.83)
```

- In the first rule, People who buy biscuits, vegetables and having total transactions of 1764 will also likely to buy bread and cake in their transaction. The confidence achieved in this outcome is 84%.

- Let X = biscuits, vegetables. Instances = 1764

- Y = Bread and cake. Instances = 1487

- Then, confidence can be calculated to be 84%. This implies, 84% of the people who buy “X” products will surely buy “Y” products.

Future aspect: The store should maintain sufficient amount of “Y” products if “X” products are maintained.

The percentage of confidence is decreased at 5th rule and so forth as the support factor decreases.

10 rules above are formed from the frequent itemsets.

Below are some frequent itemsets analyzed from above.

- Biscuits,

- Vegetables,

- Milk-cream,

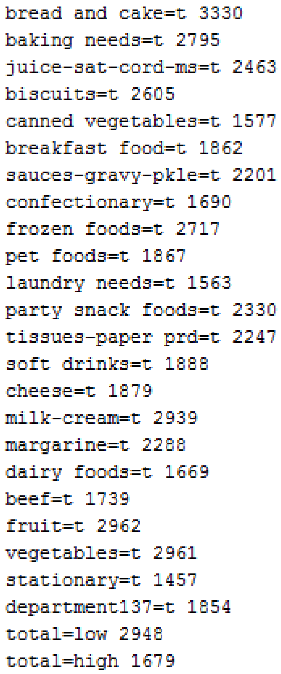
- Fruit,

- Frozen foods,

- Baking needs,

- Bread and cake.

Important attributes for forming rules with minmetric = 0.08.

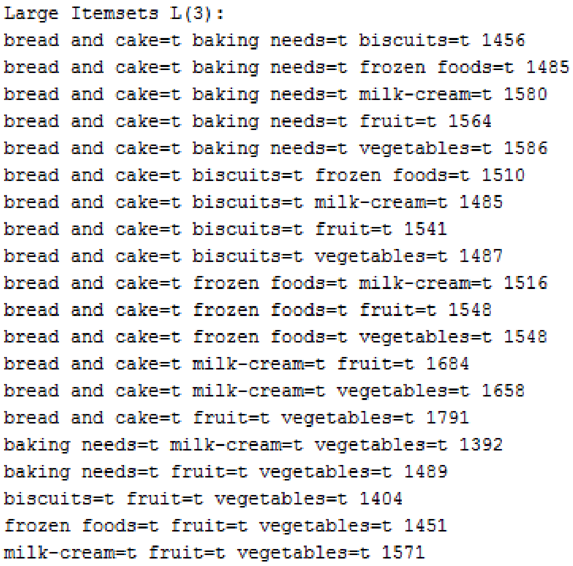


```
bread and cake=t 3330
baking needs=t 2795
juice-sat-cord-ms=t 2463
biscuits=t 2605
canned vegetables=t 1577
breakfast food=t 1862
sauces-gravy-pkle=t 2201
confectionary=t 1690
frozen foods=t 2717
pet foods=t 1867
laundry needs=t 1563
party snack foods=t 2330
tissues-paper prd=t 2247
soft drinks=t 1888
cheese=t 1879
milk-cream=t 2939
margarine=t 2288
dairy foods=t 1669
beef=t 1739
fruit=t 2962
vegetables=t 2961
stationary=t 1457
department137=t 1854
total=low 2948
total=high 1679
```

Inference from the above screenshot:

- Bread and cake have total of 3330 transactions done out of the total transactions 4627 (instances on total). And so on and so forth, we come to conclusion that Support and confidence level are gained by grouping the frequent items together bought and predicting the item that will be bought together with the frequent items.

3 Items involved:

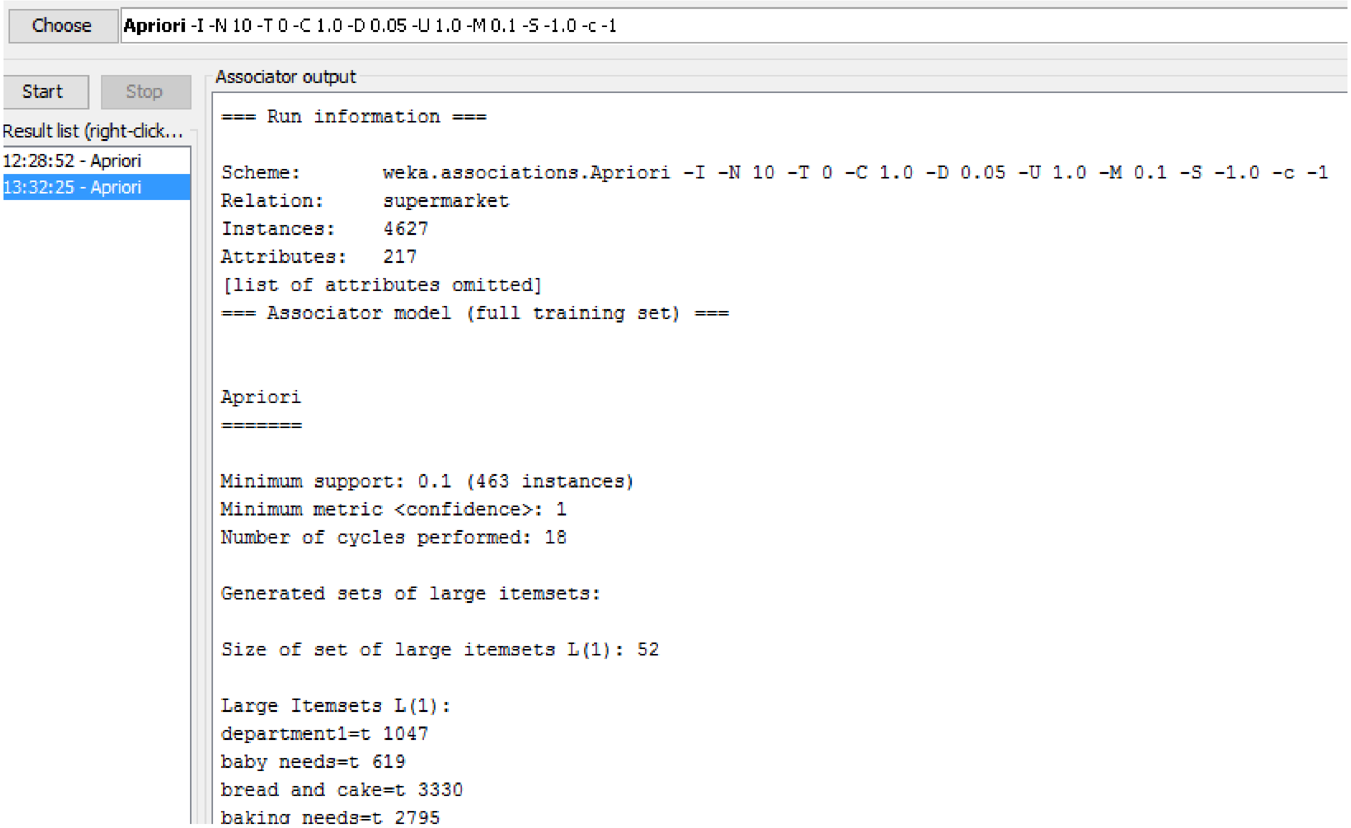


```
Large Itemsets L(3):
bread and cake=t baking needs=t biscuits=t 1456
bread and cake=t baking needs=t frozen foods=t 1485
bread and cake=t baking needs=t milk-cream=t 1580
bread and cake=t baking needs=t fruit=t 1564
bread and cake=t baking needs=t vegetables=t 1586
bread and cake=t biscuits=t frozen foods=t 1510
bread and cake=t biscuits=t milk-cream=t 1485
bread and cake=t biscuits=t fruit=t 1541
bread and cake=t biscuits=t vegetables=t 1487
bread and cake=t frozen foods=t milk-cream=t 1516
bread and cake=t frozen foods=t fruit=t 1548
bread and cake=t frozen foods=t vegetables=t 1548
bread and cake=t milk-cream=t fruit=t 1684
bread and cake=t milk-cream=t vegetables=t 1658
bread and cake=t fruit=t vegetables=t 1791
baking needs=t milk-cream=t vegetables=t 1392
baking needs=t fruit=t vegetables=t 1489
biscuits=t fruit=t vegetables=t 1404
frozen foods=t fruit=t vegetables=t 1451
milk-cream=t fruit=t vegetables=t 1571
```

- **The above screenshot thus implies 3 items bought together along with their total transactions.**

Test case 5: - Higher minmetric (from 0.9 to 1.0) – Confidence level higher.

- **When changing minmetric (from 0.9 to 1.0) -- > Having minimum metric score and asking Apriori to find rules that best matches 100% of confidence level at the minimum.**



Apriori -I -N 10 -T 0 -C 1.0 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 1.0 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.1 (463 instances)
Minimum metric <confidence>: 1
Number of cycles performed: 18

Generated sets of large itemsets:

Size of set of large itemsets L(1): 52

Large Itemsets L(1):
department1=t 1047
baby needs=t 619
bread and cake=t 3330
baking needs=t 2795
```

Inference from the above:

- Weka runs Apriori principle several times. It starts at upperboundsupport (usually 100%) and decreases by delta at each iteration (usually 5% each time) and stops when numRules = 10 are reached.

- Hence Number of cycles Apriori has run in the above case = 18 cycles (max cycles at 100% confidence).

- Delta value here = 0.05

- Upperbound support = 1

- Hence, Minimum support = 1 – 18(0.05)

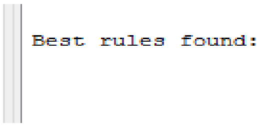
  = 0.1 (10% support)

  = 10% support of 4627 total instances

  = 463 instances

- Minimum metric confidence level is specified = 1.0 (100% of confidence level at the minimum and rules are formed based on this percentage of confidence).

Thus 10 Best rules are formed (as the numRules = 10):

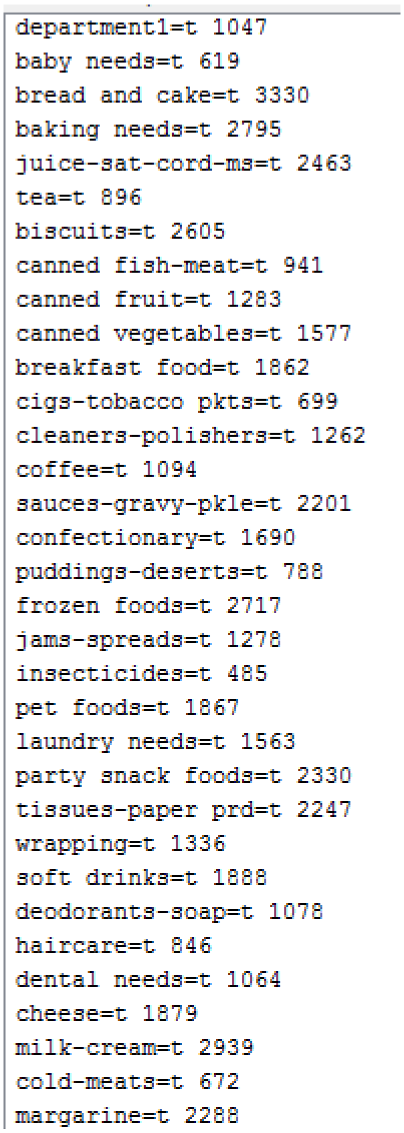


```
Best rules found:
```

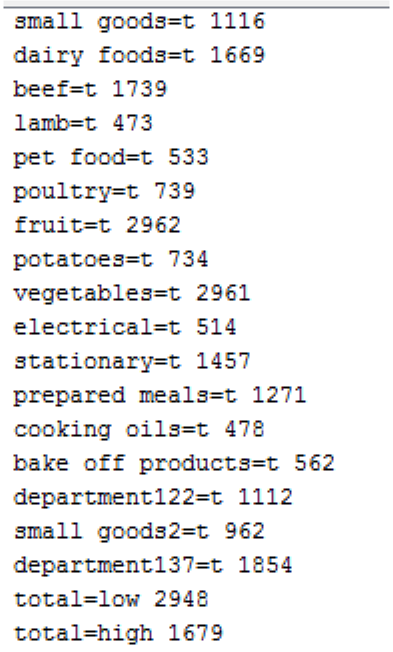
- No rule is formed when minimum confidence level is set to 100%.
- This implies, there are no combinations/groups of products can be made to form a rule with 100% confidence level.

Future aspect: The store will not be able to decide in this case, as there is no single combination where customer chooses which raises the confidence level to 100%. This implies, there is no guarantee a product is bought will lead to buying another product.

Below are some frequent itemsets analyzed from above – Totally 52 itemsets are taken to form the Apriori algorithm.



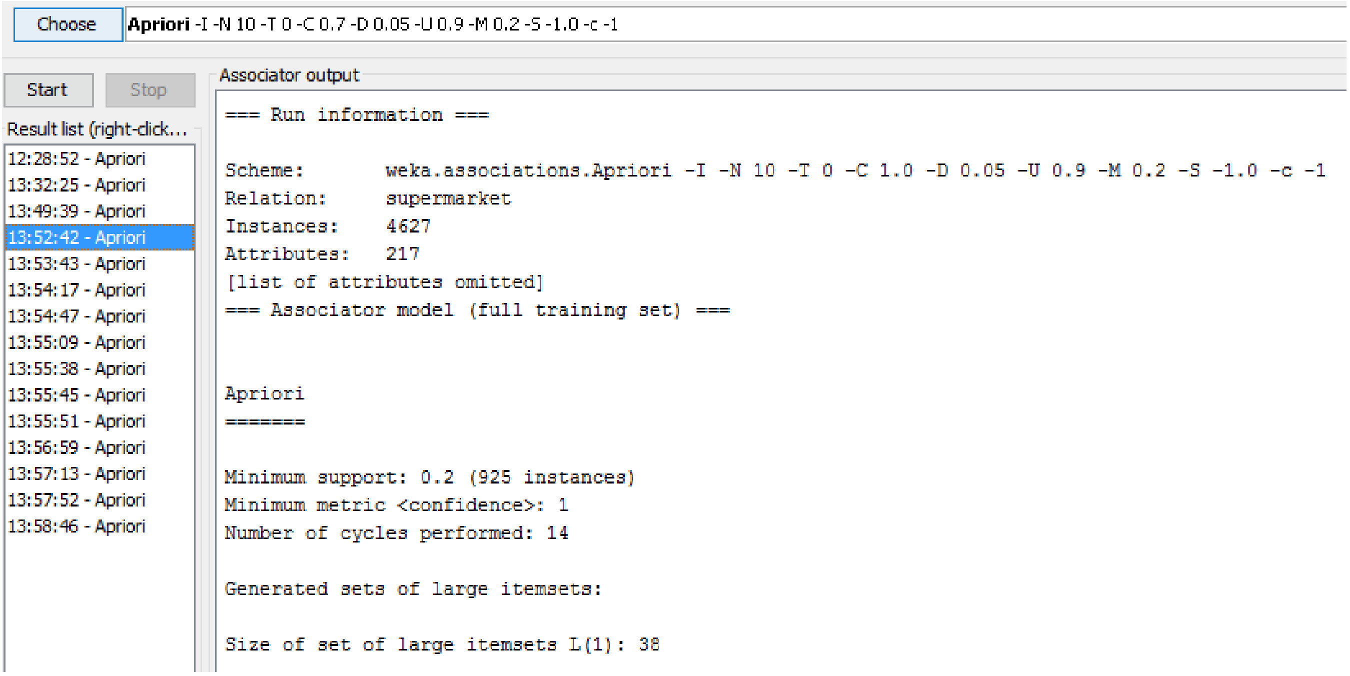
```
department1=t 1047
baby needs=t 619
bread and cake=t 3330
baking needs=t 2795
juice-sat-cord-ms=t 2463
tea=t 896
biscuits=t 2605
canned fish-meat=t 941
canned fruit=t 1283
canned vegetables=t 1577
breakfast food=t 1862
cigs-tobacco pkts=t 699
cleaners-polishers=t 1262
coffee=t 1094
sauces-gravy-pkle=t 2201
confectionary=t 1690
puddings-deserts=t 788
frozen foods=t 2717
jams-spreads=t 1278
insecticides=t 485
pet foods=t 1867
laundry needs=t 1563
party snack foods=t 2330
tissues-paper prd=t 2247
wrapping=t 1336
soft drinks=t 1888
deodorants-soap=t 1078
haircare=t 846
dental needs=t 1064
cheese=t 1879
milk-cream=t 2939
cold-meats=t 672
margarine=t 2288
```



```
small goods=t 1116
dairy foods=t 1669
beef=t 1739
lamb=t 473
pet food=t 533
poultry=t 739
fruit=t 2962
potatoes=t 734
vegetables=t 2961
electrical=t 514
stationary=t 1457
prepared meals=t 1271
cooking oils=t 478
bake off products=t 562
department122=t 1112
small goods2=t 962
department137=t 1854
total=low 2948
total=high 1679
```

Test case 6: - Lower bound = 0.2 and upper bound = 0.9 – Support level range – 20 – 90%.

- **When changing lowerbound (from 0.1 to 0.2) and upperbound (from 1.0 to 0.9) -- > This implies the Support value is in the range of 20% - 90% of the data.**



Apriori -I -N 10 -T 0 -C 0.7 -D 0.05 -U 0.9 -M 0.2 -S -1.0 -c -1

```
12:28:52 - Apriori
13:32:25 - Apriori
13:49:39 - Apriori
13:52:42 - Apriori
13:53:43 - Apriori
13:54:17 - Apriori
13:54:47 - Apriori
13:55:09 - Apriori
13:55:38 - Apriori
13:55:45 - Apriori
13:55:51 - Apriori
13:56:59 - Apriori
13:57:13 - Apriori
13:57:52 - Apriori
13:58:46 - Apriori
```

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 1.0 -D 0.05 -U 0.9 -M 0.2 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.2 (925 instances)
Minimum metric <confidence>: 1
Number of cycles performed: 14

Generated sets of large itemsets:

Size of set of large itemsets L(1): 38
```

Inference from the above:

- Number of cycles Apriori has run in the above case = 14 cycles (when support level is between 20 and 90%).
- Delta value here remains unchanged = 0.05
- Upperbound support = 0.9
- Hence, Minimum support = 0.9 – 14(0.05)

  = 0.2 (20% support)

  = 20% support of 4627 total instances

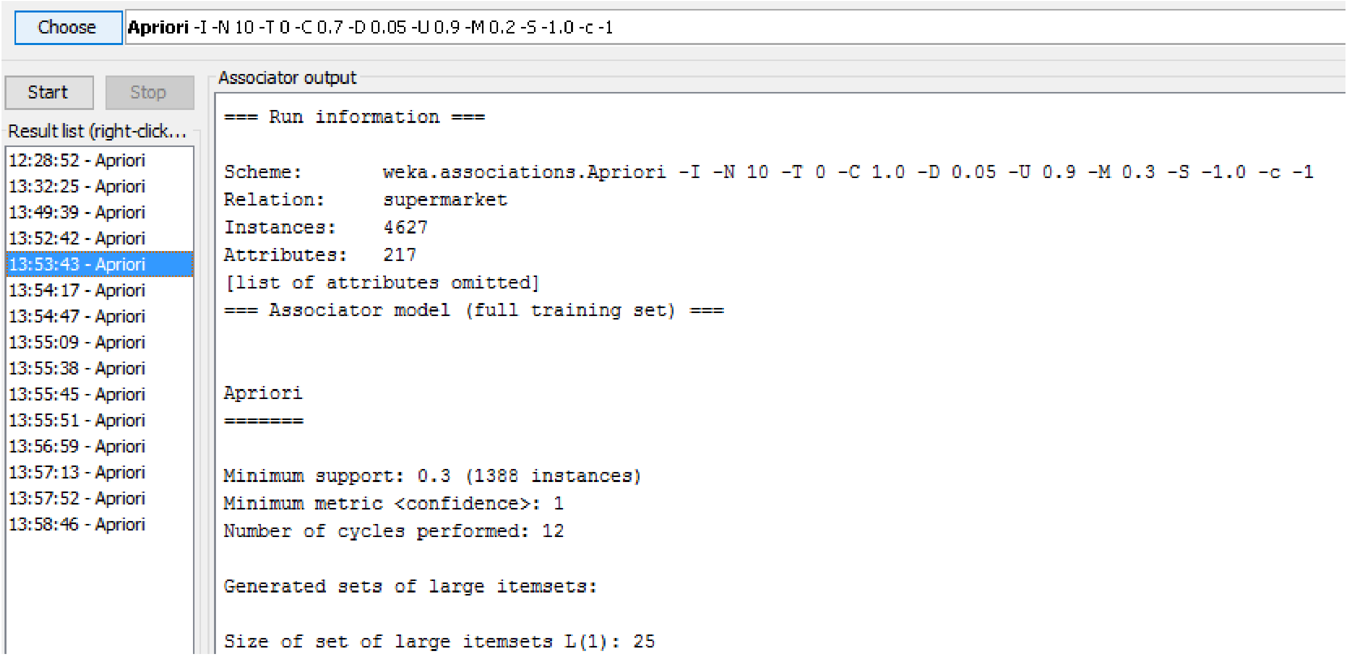
  = 925 instances

- Minimum metric confidence level is specified = 1.0 (100% of confidence level at the minimum and rules are formed based on this percentage of confidence).
- (Or) Minimum metric confidence level = 0.9 or below (90% or below).

No rules are formed (even if minmetric is 0.9) based on the above test case as minimum and maximum support levels are changed.

Test case 7: - Lower bound = 0.3 and upper bound = 0.9 – Support level range – 30 – 90%.

- **When changing lowerbound (from 0.2 to 0.3) and upperbound (remains unchanged = 0.9) -- > This implies the Support value is in the range of 30% - 90% of the data.**



Apriori -I -N 10 -T 0 -C 0.7 -D 0.05 -U 0.9 -M 0.2 -S -1.0 -c -1

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 1.0 -D 0.05 -U 0.9 -M 0.3 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.3 (1388 instances)
Minimum metric <confidence>: 1
Number of cycles performed: 12

Generated sets of large itemsets:

Size of set of large itemsets L(1): 25
```

Inference from the above:

- Number of cycles Apriori has run in the above case = 12 cycles (when support level is between 30 and 90%).

- Delta value here remains unchanged = 0.05

- Upperbound support = 0.9

- Hence, Minimum support = 0.9 – 12(0.05)

  = 0.3 (30% support)

  = 30% support of 4627 total instances

  = 1388 instances

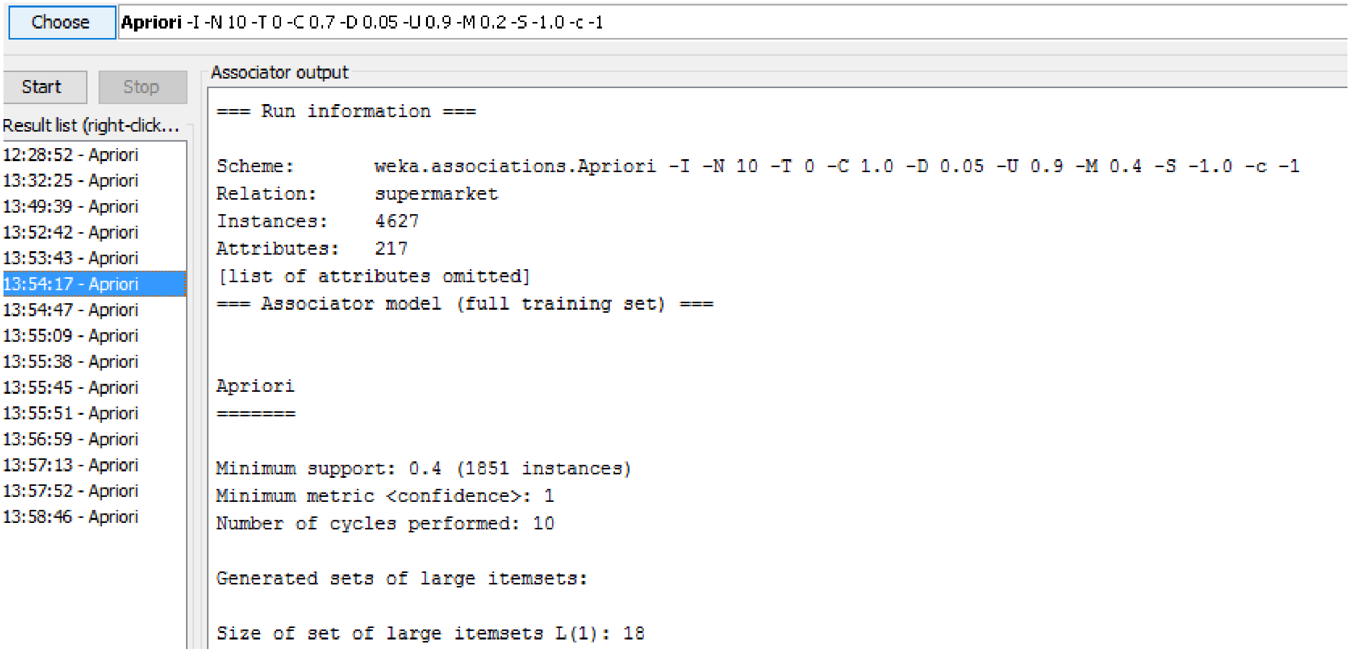
- Minimum metric confidence level is specified = 1.0 (100% of confidence level at the minimum and rules are formed based on this percentage of confidence).

- (Or) Minimum metric confidence level = 0.9 or below (90% or below).

No rules are formed (even if minmetric is 0.9) based on the above test case as minimum and maximum support levels are changed.

Test case 8: - Lower bound (M) = 0.4 and upper bound (U) = 0.9 – Support level range – 40 – 90%.

- **When changing lowerbound (from 0.3 to 0.4) and upperbound (remains unchanged = 0.9) -- > This implies the Support value is in the range of 40% - 90% of the data.**



Apriori -I -N 10 -T 0 -C 0.7 -D 0.05 -U 0.9 -M 0.2 -S -1.0 -c -1

12:28:52 - Apriori
13:32:25 - Apriori
13:49:39 - Apriori
13:52:42 - Apriori
13:53:43 - Apriori
13:54:17 - Apriori
13:54:47 - Apriori
13:55:09 - Apriori
13:55:38 - Apriori
13:55:45 - Apriori
13:55:51 - Apriori
13:56:59 - Apriori
13:57:13 - Apriori
13:57:52 - Apriori
13:58:46 - Apriori

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 1.0 -D 0.05 -U 0.9 -M 0.4 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.4 (1851 instances)
Minimum metric <confidence>: 1
Number of cycles performed: 10

Generated sets of large itemsets:

Size of set of large itemsets L(1): 18
```

Inference from the above:

- Number of cycles Apriori has run in the above case = 10 cycles (when support level is between 30 and 90%).
- Delta value here remains unchanged = 0.05
- Upperbound support = 0.9
- Hence, Minimum support = 0.9 – 10(0.05)

  = 0.4 (40% support)

  = 40% support of 4627 total instances

  = 1851 instances

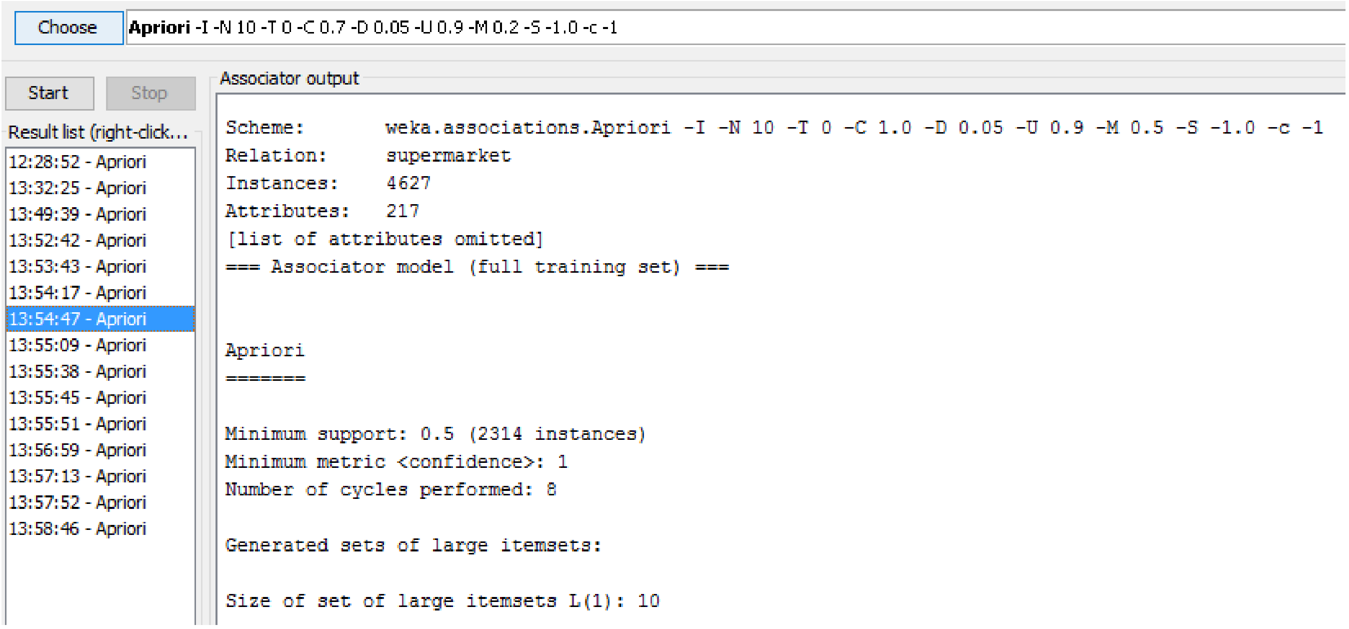
- Minimum metric confidence level is specified = 1.0 (100% of confidence level at the minimum and rules are formed based on this percentage of confidence).

(Or) Minimum metric confidence level = 0.9 or below (90% or below)

No rules are formed (even if minmetric is 0.9) based on the above test case as minimum and maximum support levels are changed.

Test case 9: - Lower bound (M) = 0.5 and upper bound (U) = 0.9 – Support level range – 50 – 90%.

- **When changing lowerbound (from 0.4 to 0.5) and upperbound (remains unchanged = 0.9) -- > This implies the Support value is in the range of 50% - 90% of the data.**



```
Apriori -I -N 10 -T 0 -C 0.7 -D 0.05 -U 0.9 -M 0.2 -S -1.0 -c -1
```

```
Result list (right-click...
12:28:52 - Apriori
13:32:25 - Apriori
13:49:39 - Apriori
13:52:42 - Apriori
13:53:43 - Apriori
13:54:17 - Apriori
13:54:47 - Apriori
13:55:09 - Apriori
13:55:38 - Apriori
13:55:45 - Apriori
13:55:51 - Apriori
13:56:59 - Apriori
13:57:13 - Apriori
13:57:52 - Apriori
13:58:46 - Apriori
```

```
Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 1.0 -D 0.05 -U 0.9 -M 0.5 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.5 (2314 instances)
Minimum metric <confidence>: 1
Number of cycles performed: 8

Generated sets of large itemsets:

Size of set of large itemsets L(1): 10
```

Inference from the above:

- Number of cycles Apriori has run in the above case = 8 cycles (when support level is between 30 and 90%).

- Delta value here remains unchanged = 0.05

- Upperbound support = 0.9

- Hence, Minimum support = 0.9 – 8(0.05)

  = 0.5 (50% support)

  = 50% support of 4627 total instances

  = 2314 instances

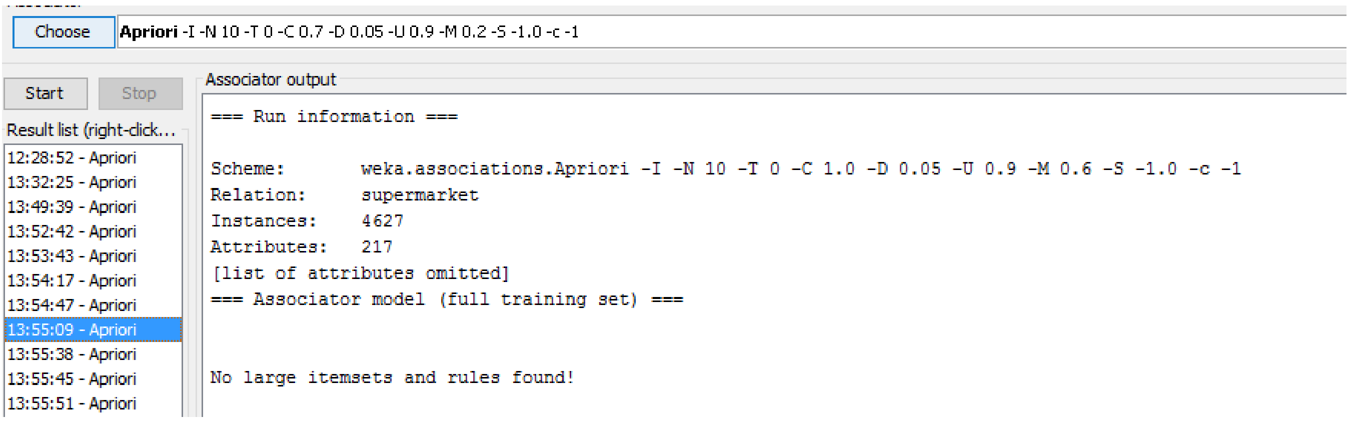
- Minimum metric confidence level is specified = 1.0 (100% of confidence level at the minimum and rules are formed based on this percentage of confidence).

(Or) Minimum metric confidence level = 0.9 or below (90% or below)

No rules are formed (even if minmetric is 0.9) based on the above test case as minimum and maximum support levels are changed.

Test case 10: - Lower bound (M) = 0.6 and upper bound (U) = 0.9 – Support level range – 60 – 90%.

- **When changing lowerbound (from 0.5 to 0.6) and upperbound (remains unchanged = 0.9) -- > This implies the Support value is in the range of 60% - 90% of the data.**



Apriori -I -N 10 -T 0 -C 0.7 -D 0.05 -U 0.9 -M 0.2 -S -1.0 -c -1

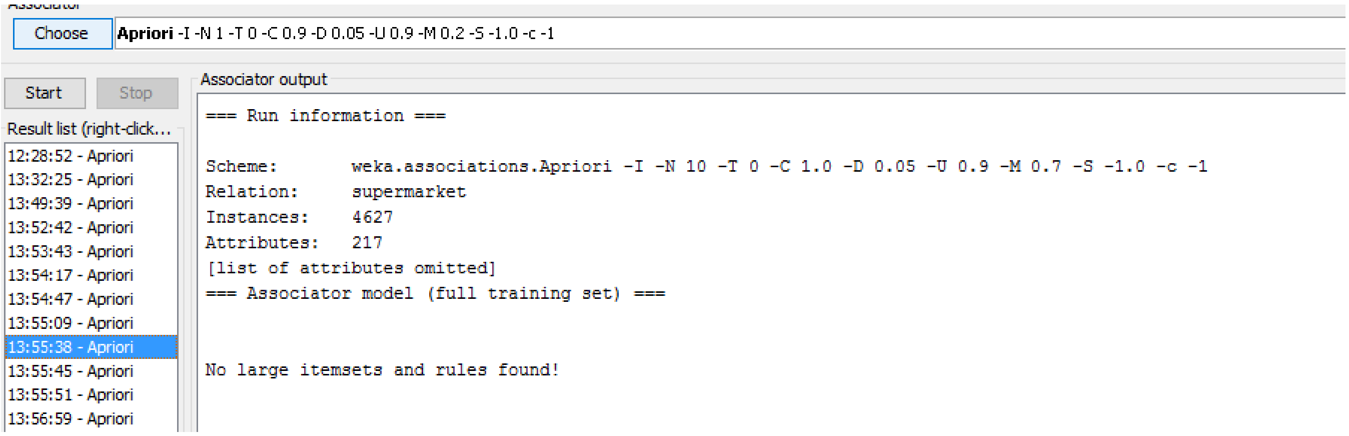
```
12:28:52 - Apriori
13:32:25 - Apriori
13:49:39 - Apriori
13:52:42 - Apriori
13:53:43 - Apriori
13:54:17 - Apriori
13:54:47 - Apriori
13:55:09 - Apriori
13:55:38 - Apriori
13:55:45 - Apriori
13:55:51 - Apriori
```

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 1.0 -D 0.05 -U 0.9 -M 0.6 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


No large itemsets and rules found!
```

Inference from the above:

- Once the lowerbound has crossed 60% and above, there are no large itemsets or rules found/formed by Apriori algorithm.
- M = 0.6
- U = 0.9 or 1.0
- Apriori algorithm cannot run the cycles to reach the max of 0.9 upper bound.
- Once after this threshold is reached, for lowerbound 60% and above, there are no rules and no Large itemsets can be calculated by the Apriori algorithm.

When M = 0.7 and U = 0.9

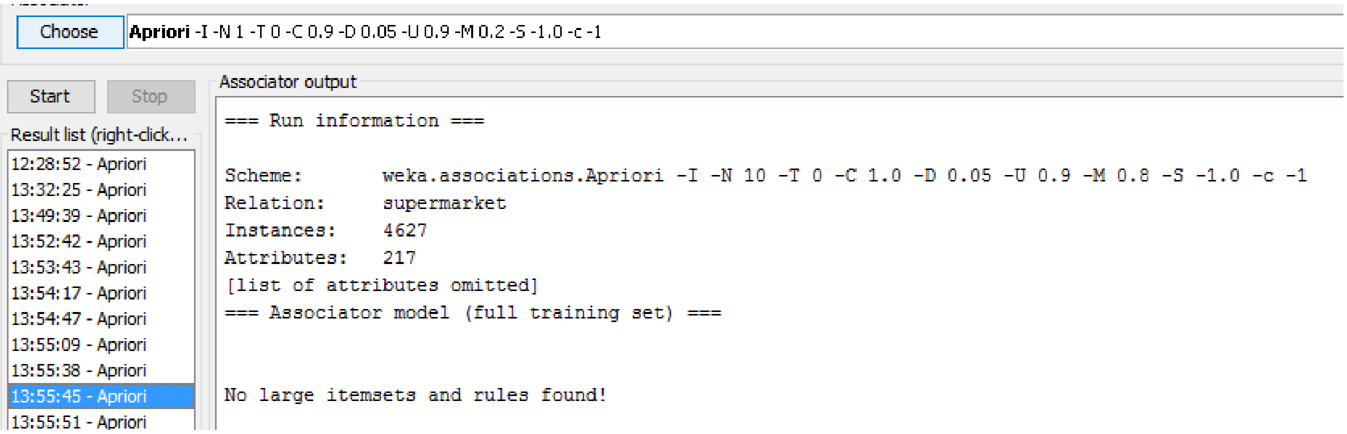


Apriori -I -N 1 -T 0 -C 0.9 -D 0.05 -U 0.9 -M 0.2 -S -1.0 -c -1

```
12:28:52 - Apriori
13:32:25 - Apriori
13:49:39 - Apriori
13:52:42 - Apriori
13:53:43 - Apriori
13:54:17 - Apriori
13:54:47 - Apriori
13:55:09 - Apriori
13:55:38 - Apriori
13:55:45 - Apriori
13:55:51 - Apriori
13:56:59 - Apriori
```

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 1.0 -D 0.05 -U 0.9 -M 0.7 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


No large itemsets and rules found!
```

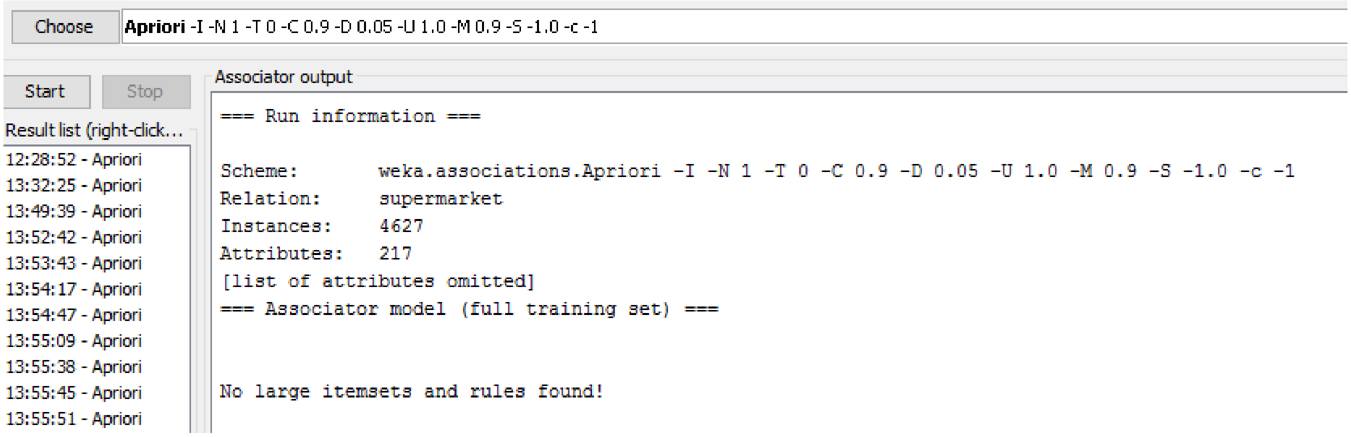
When M = 0.8 and U = 0.9



Apriori -I -N 1 -T 0 -C 0.9 -D 0.05 -U 0.9 -M 0.2 -S -1.0 -c -1

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 1.0 -D 0.05 -U 0.9 -M 0.8 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


No large itemsets and rules found!
```

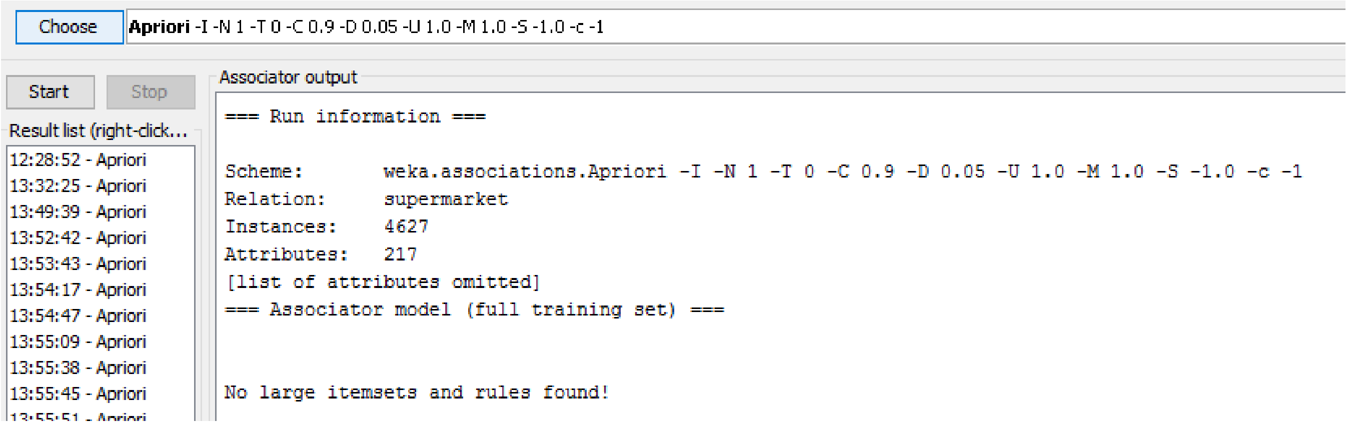
When M = 0.9 and U = 1.0



Apriori -I -N 1 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.9 -S -1.0 -c -1

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 1 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.9 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


No large itemsets and rules found!
```

When M = 1.0 and U = 1.0 ---- > Doesn’t make any sense though.



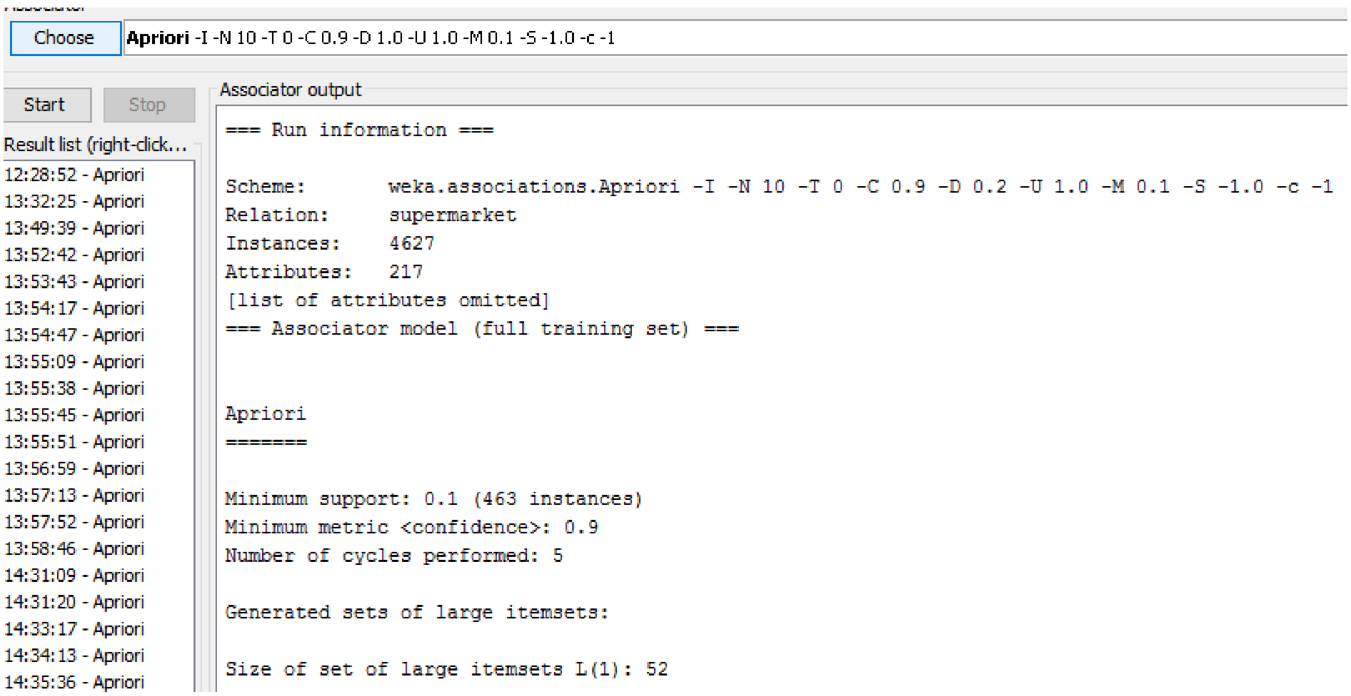
Apriori -I -N 1 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 1.0 -S -1.0 -c -1

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 1 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 1.0 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


No large itemsets and rules found!
```

No rules are formed (even if minmetric is 0.9) and no Large datasets are found beyond 60% of support level.

Test case 11: - Lower bound (M) = 0.1 and upper bound (U) = 0.9 – Support level range – 10 – 90% but increase the Delta value from 5% (default) to 20% and above and record the results.

- **When changing Delta (from 5% to 20%)**



```
Apriori -I -N 10 -T 0 -C 0.9 -D 1.0 -U 1.0 -M 0.1 -S -1.0 -c -1
```

```
12:28:52 - Apriori
13:32:25 - Apriori
13:49:39 - Apriori
13:52:42 - Apriori
13:53:43 - Apriori
13:54:17 - Apriori
13:54:47 - Apriori
13:55:09 - Apriori
13:55:38 - Apriori
13:55:45 - Apriori
13:55:51 - Apriori
13:56:59 - Apriori
13:57:13 - Apriori
13:57:52 - Apriori
13:58:46 - Apriori
14:31:09 - Apriori
14:31:20 - Apriori
14:33:17 - Apriori
14:34:13 - Apriori
14:35:36 - Apriori
```

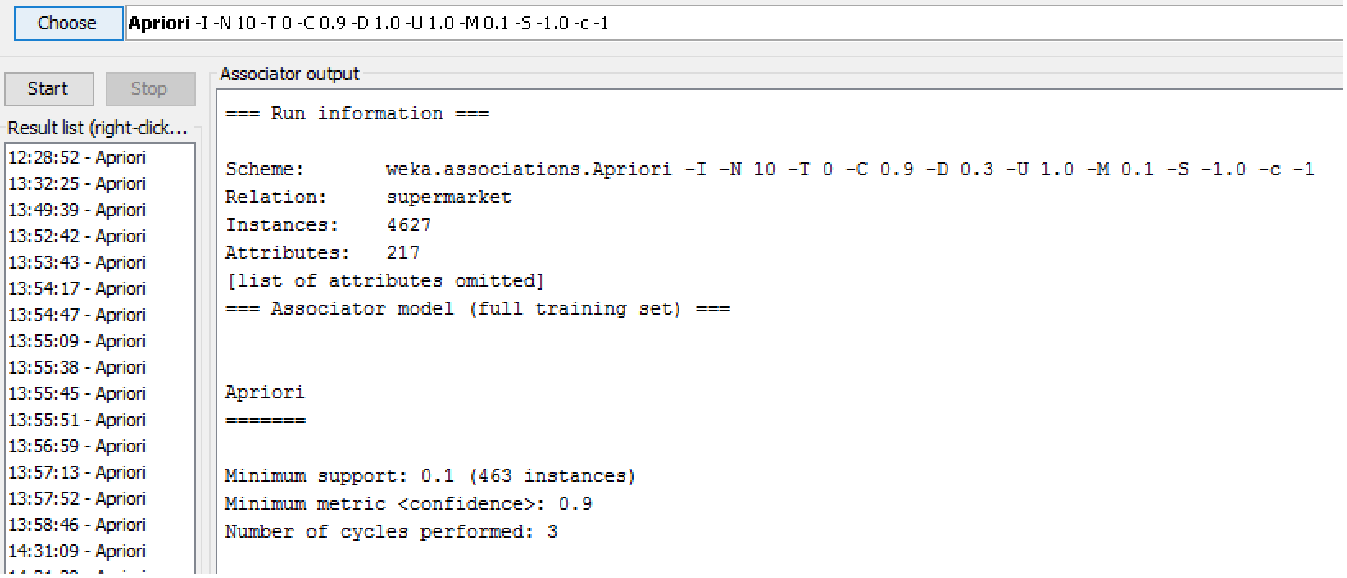
```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 0.9 -D 0.2 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.1 (463 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 5

Generated sets of large itemsets:

Size of set of large itemsets L(1): 52
```

Inference from the above:

- When Delta = 0.2, Lower bound = 01, Upperbound support = 0.9, minimum number of cycles formed by Apriori algorithm = 5.
- Apriori is run 5 times starting at upper bound support 1.0 (100%) and decreasing every 20% of the iterations to reach number of rules = 10. Hence 5 such cycles are run back to back to produce the rules.
- Number of cycles decreases when Delta value increases. When compared to the previous test cases, Delta value was 5% and the number of cycles were 18 at the max.
- Here, the delta value is 20% and the number of cycles Apriori ran was only 5.

When Delta = 0.3 (30%) M = 0.1 and U = 0.9

- **When changing Delta (from 20% to 30%)**



Apriori -I -N 10 -T 0 -C 0.9 -D 1.0 -U 1.0 -M 0.1 -S -1.0 -c -1

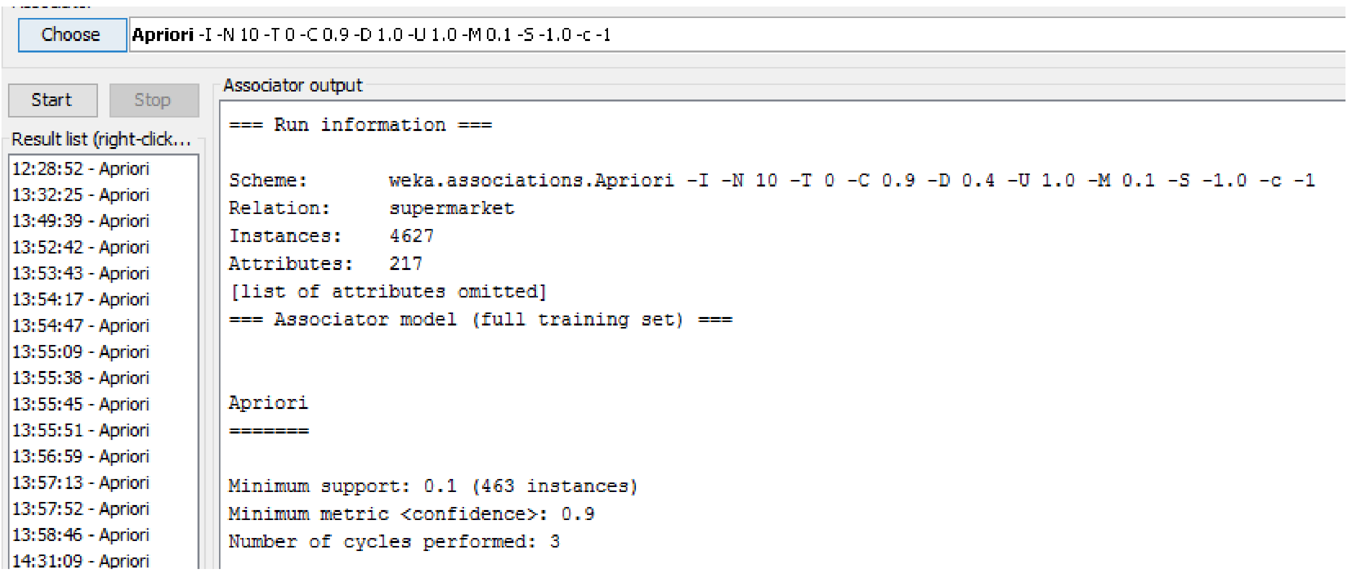
```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 0.9 -D 0.3 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.1 (463 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 3
```

Inference from the above:

- When Delta = 0.3, Lower bound = 01, Upperbound support = 0.9, minimum number of cycles formed by Apriori algorithm = 3.
- Apriori is run 3 times starting at upper bound support 1.0 (100%) and decreasing every 30% of the iterations to reach number of rules = 10. Hence 3 such cycles are run back to back to produce the rules.
- Number of cycles decreases when Delta value increases. When compared to the previous test case TC 10, Delta value was 20% and the number of cycles were 5 at the max.
- Here, the delta value is 30% and the number of cycles Apriori ran was only 3.

When Delta = 0.4 (40%) M = 0.1 and U = 0.9

- **When changing Delta (from 30% to 40%)**



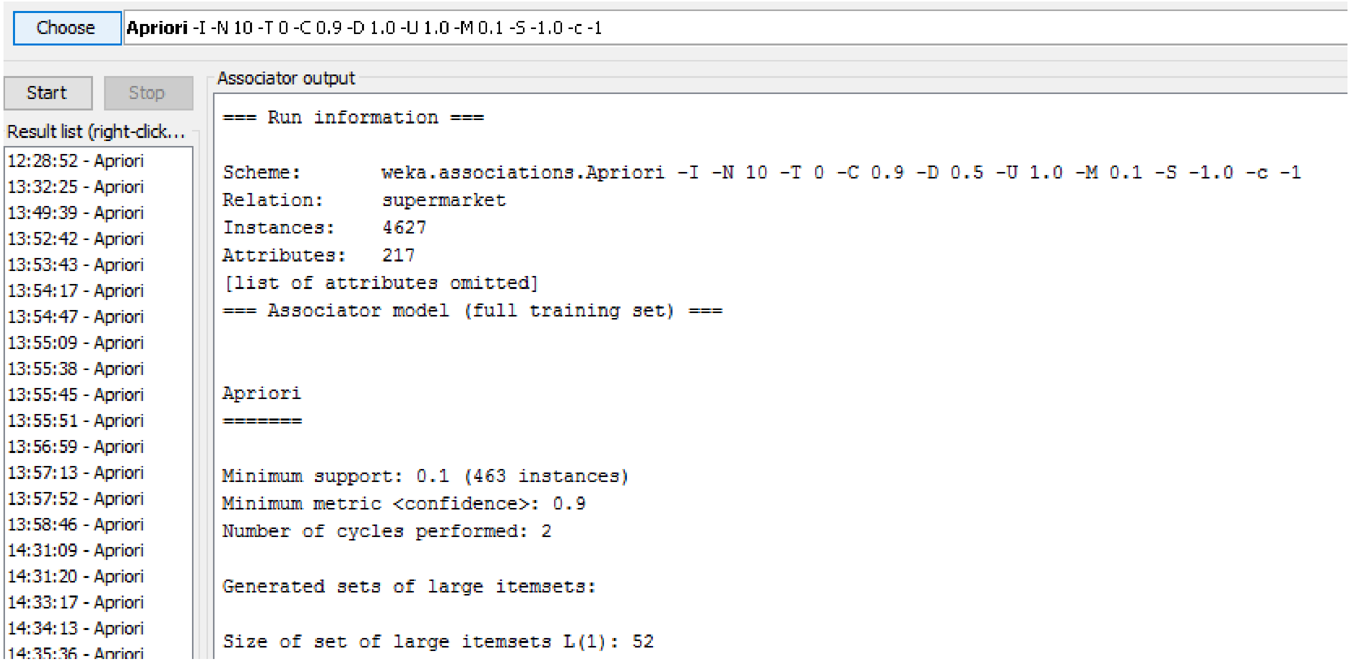
Apriori -I -N 10 -T 0 -C 0.9 -D 1.0 -U 1.0 -M 0.1 -S -1.0 -c -1

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 0.9 -D 0.4 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.1 (463 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 3
```

Inference from the above:

- ➢ When Delta = 0.4, Lower bound = 01, Upperbound support = 0.9, minimum number of cycles formed by Apriori algorithm = 3.
- ➢ Apriori is run 3 times starting at upper bound support 1.0 (100%) and decreasing every 40% of the iterations to reach number of rules = 10. Hence 3 such cycles are run back to back to produce the rules.
- ➢ Here, the Number of cycles remains the same when Delta value increases from 30 – 40%.
- ➢ Here, the delta value is 40% and the number of cycles Apriori ran is 3.

When Delta = 0.5 (50%) M = 0.1 and U = 0.9



Apriori -I -N 10 -T 0 -C 0.9 -D 1.0 -U 1.0 -M 0.1 -S -1.0 -c -1

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 0.9 -D 0.5 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.1 (463 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 2

Generated sets of large itemsets:

Size of set of large itemsets L(1): 52
```

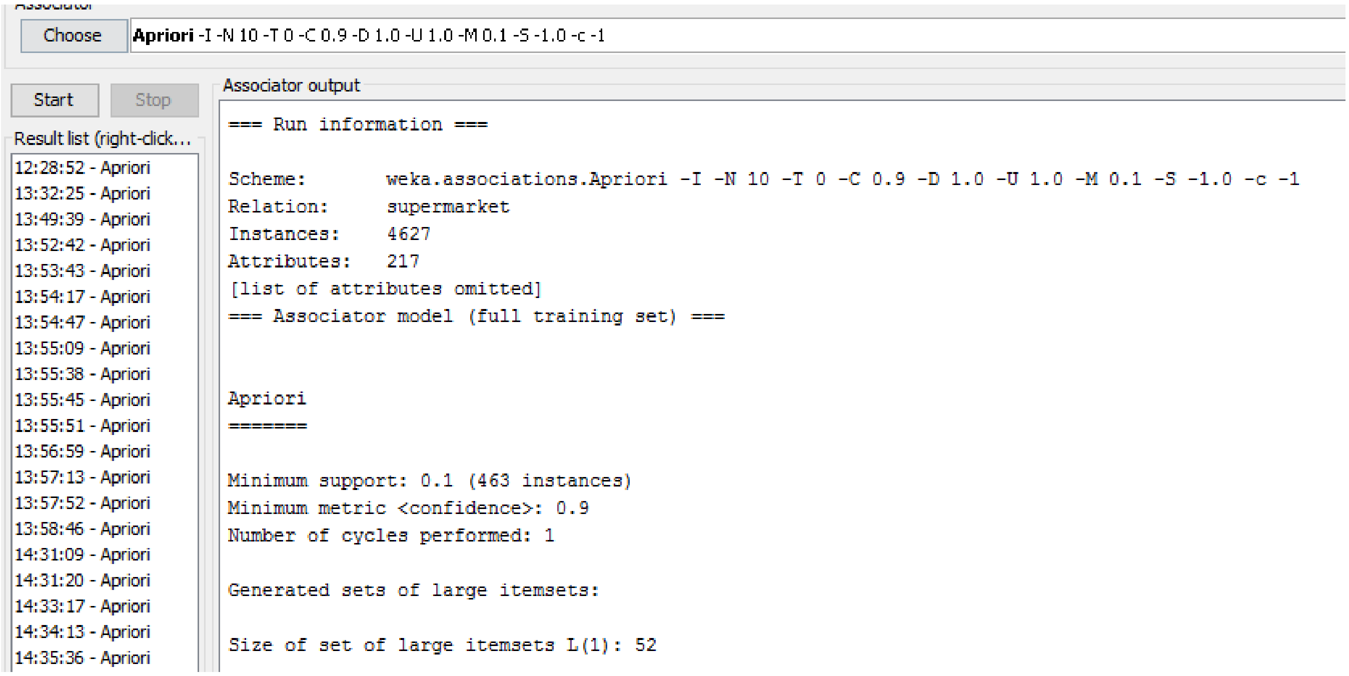
Number of cycles decreases to 2 when Delta = 0.5 (50%)

When Delta = 0.6, 0.7, 0.8 (50 - 60%) M = 0.1 and U = 0.9

Number of cycles remains unchanged.

When Delta = 0.9(90%) M = 0.1 and U = 0.9

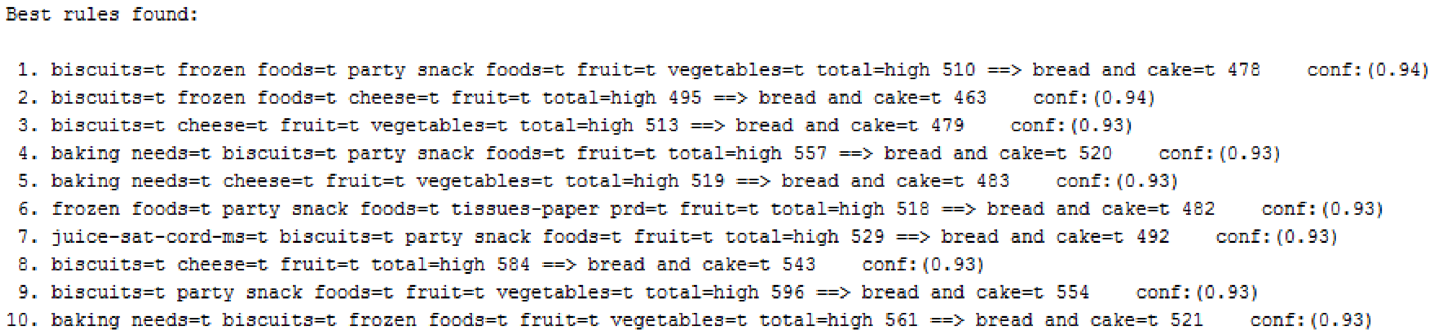
Number of cycles further decreases to 1.



Apriori -I -N 10 -T 0 -C 0.9 -D 1.0 -U 1.0 -M 0.1 -S -1.0 -c -1

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 0.9 -D 1.0 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.1 (463 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 1

Generated sets of large itemsets:

Size of set of large itemsets L(1): 52
```

Rules thus formed:



```
Best rules found:

 1. biscuits=t frozen foods=t party snack foods=t fruit=t vegetables=t total=high 510 ==> bread and cake=t 478    conf:(0.94)
 2. biscuits=t frozen foods=t cheese=t fruit=t total=high 495 ==> bread and cake=t 463    conf:(0.94)
 3. biscuits=t cheese=t fruit=t vegetables=t total=high 513 ==> bread and cake=t 479    conf:(0.93)
 4. baking needs=t biscuits=t party snack foods=t fruit=t total=high 557 ==> bread and cake=t 520    conf:(0.93)
 5. baking needs=t cheese=t fruit=t vegetables=t total=high 519 ==> bread and cake=t 483    conf:(0.93)
 6. frozen foods=t party snack foods=t tissues-paper prd=t fruit=t total=high 518 ==> bread and cake=t 482    conf:(0.93)
 7. juice-sat-cord-ms=t biscuits=t party snack foods=t fruit=t total=high 529 ==> bread and cake=t 492    conf:(0.93)
 8. biscuits=t cheese=t fruit=t total=high 584 ==> bread and cake=t 543    conf:(0.93)
 9. biscuits=t party snack foods=t fruit=t vegetables=t total=high 596 ==> bread and cake=t 554    conf:(0.93)
10. baking needs=t biscuits=t frozen foods=t fruit=t vegetables=t total=high 561 ==> bread and cake=t 521    conf:(0.93)
```

Thus, confidence level is high when Delta = 10 or 20 % (where % of confidence = 94) than Delta = 5% (where % of confidence = 92%

Weka runs Apriori many times at upperbound level (100%) decreasing by delta at each iteration and stops when numRules is reached.

When Delta = 5%

Number of cycles from Apriori = 18 (max number of cycles with 90% confidence)

When Delta = 20%

Number of cycles = 5 (Decreased as Delta value increases)

When Delta = 30 to 40%

Number of cycles = 3 (Decreased further)

When Delta = 50 to 80%

Number of cycles = 2 (Decreased further but remains unchanged throughout 80%)

When Delta = 90 and above (though this does not make any sense)

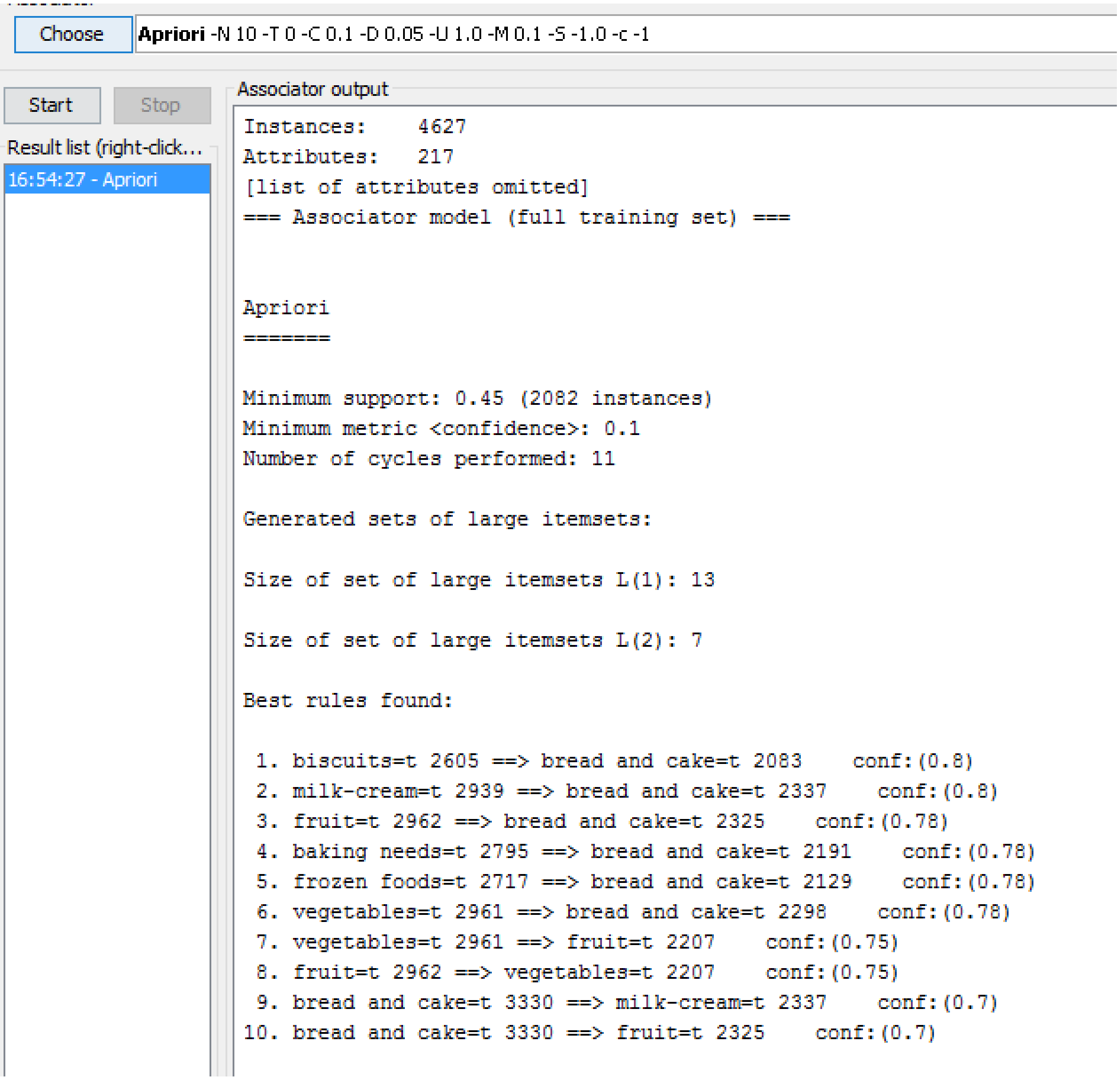
Number of cycles = 1 (remains unchanged)

Test case 12: - What happens with 9 combinations of the values of support and confidence (support/confidence) without changing parameters of Apriori algorithm:

Support Lowerboundminsupport	Confidence – Minmetric	% of confidence	Number of cycles
Low (M= 0.1)	Low (C = 0.1)	Starts at 80% decreased to 70%	11
Low (M = 0.1)	Medium (C = 0.5)	Starts at 80% decreased to 70%	11
Low (M = 0.1)	High (C = 1.0)	No rules formed	18 (max cycles)
Medium (M = 0.5)	Low (C = 0.1)	Only 4 rules are formed. 80% confidence	10
Medium (M = 0.5)	Medium (C = 0.5)	Only 4 rules are formed with 80 % of confidence gradually decreasing to 70%.	10
Medium (M = 0.5)	High (C = 1.0)	No rules formed	10
High (M = 1.0)	Low (C = 0.1)	No Large itemsets or rules formed	-
High (M = 1.0)	Medium (C = 0.5)	No Large itemsets or rules formed	-
High (M = 1.0)	High (C = 1.0)	No Large itemsets or rules formed	-

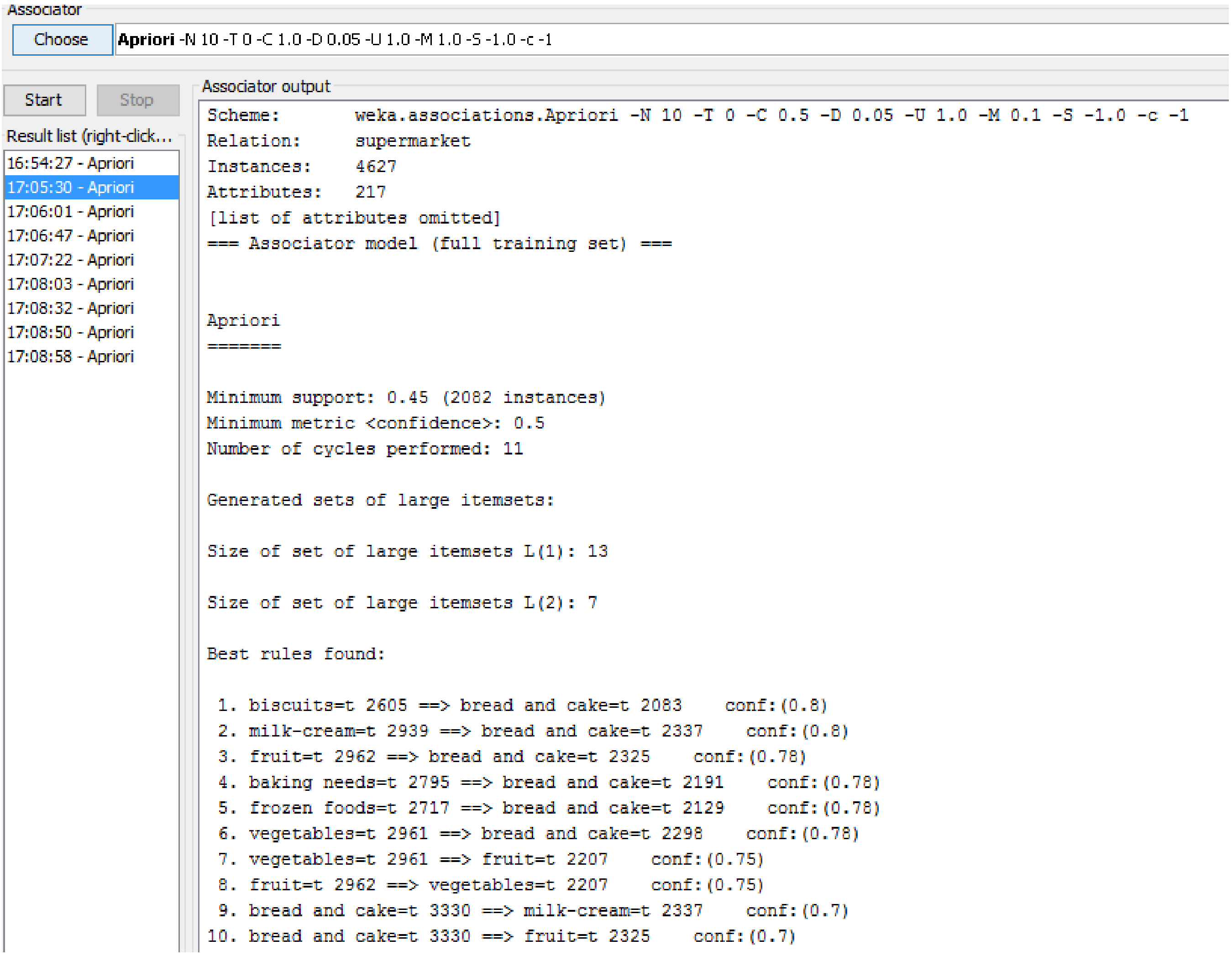
Screenshots of the above test cases, results are given below in order.

1.)



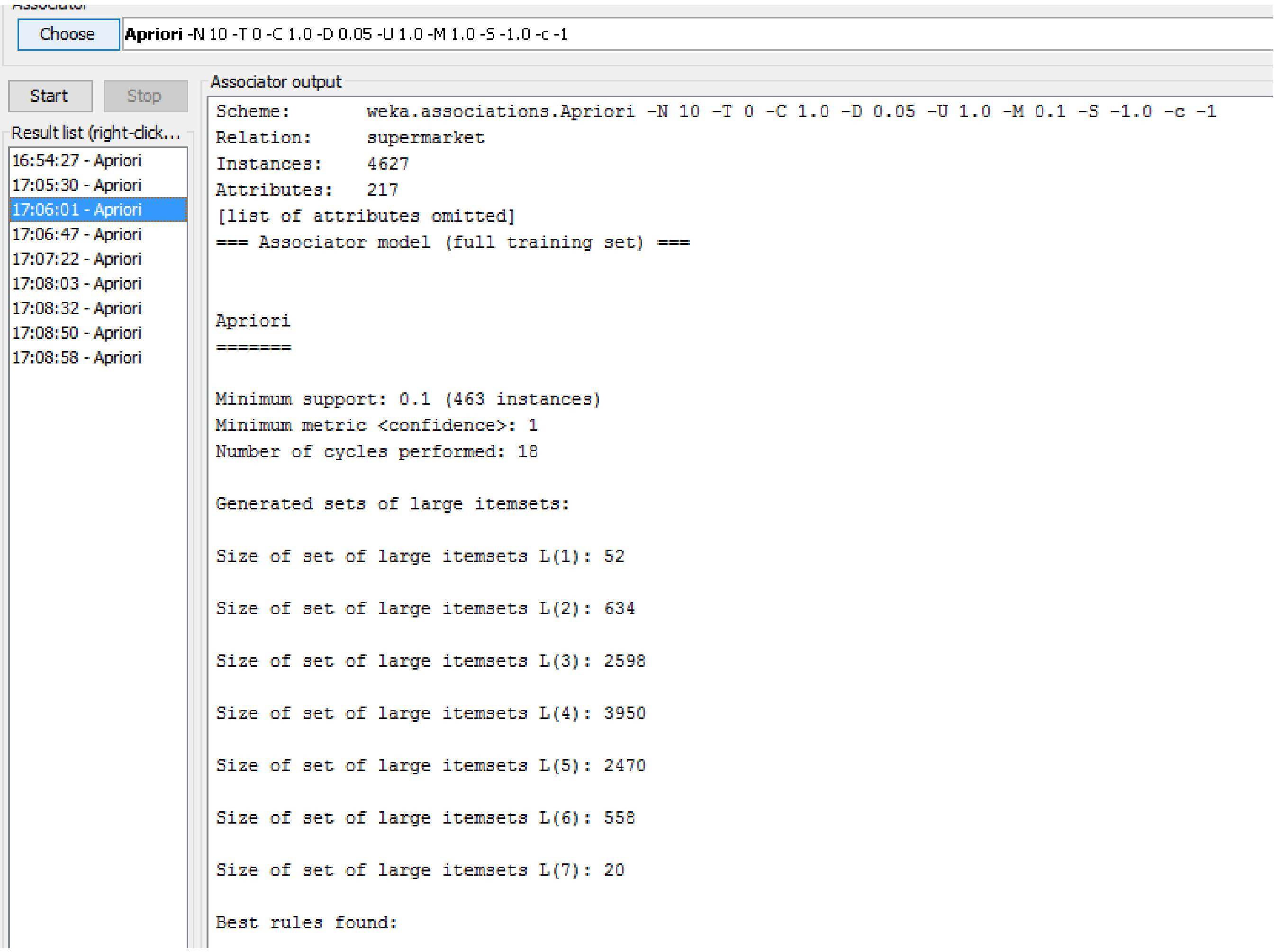
Apriori -N 10 -T 0 -C 0.1 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

```
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.45 (2082 instances)
Minimum metric <confidence>: 0.1
Number of cycles performed: 11

Generated sets of large itemsets:

Size of set of large itemsets L(1): 13

Size of set of large itemsets L(2): 7

Best rules found:

 1. biscuits=t 2605 ==> bread and cake=t 2083    conf:(0.8)
 2. milk-cream=t 2939 ==> bread and cake=t 2337    conf:(0.8)
 3. fruit=t 2962 ==> bread and cake=t 2325    conf:(0.78)
 4. baking needs=t 2795 ==> bread and cake=t 2191    conf:(0.78)
 5. frozen foods=t 2717 ==> bread and cake=t 2129    conf:(0.78)
 6. vegetables=t 2961 ==> bread and cake=t 2298    conf:(0.78)
 7. vegetables=t 2961 ==> fruit=t 2207    conf:(0.75)
 8. fruit=t 2962 ==> vegetables=t 2207    conf:(0.75)
 9. bread and cake=t 3330 ==> milk-cream=t 2337    conf:(0.7)
10. bread and cake=t 3330 ==> fruit=t 2325    conf:(0.7)
```

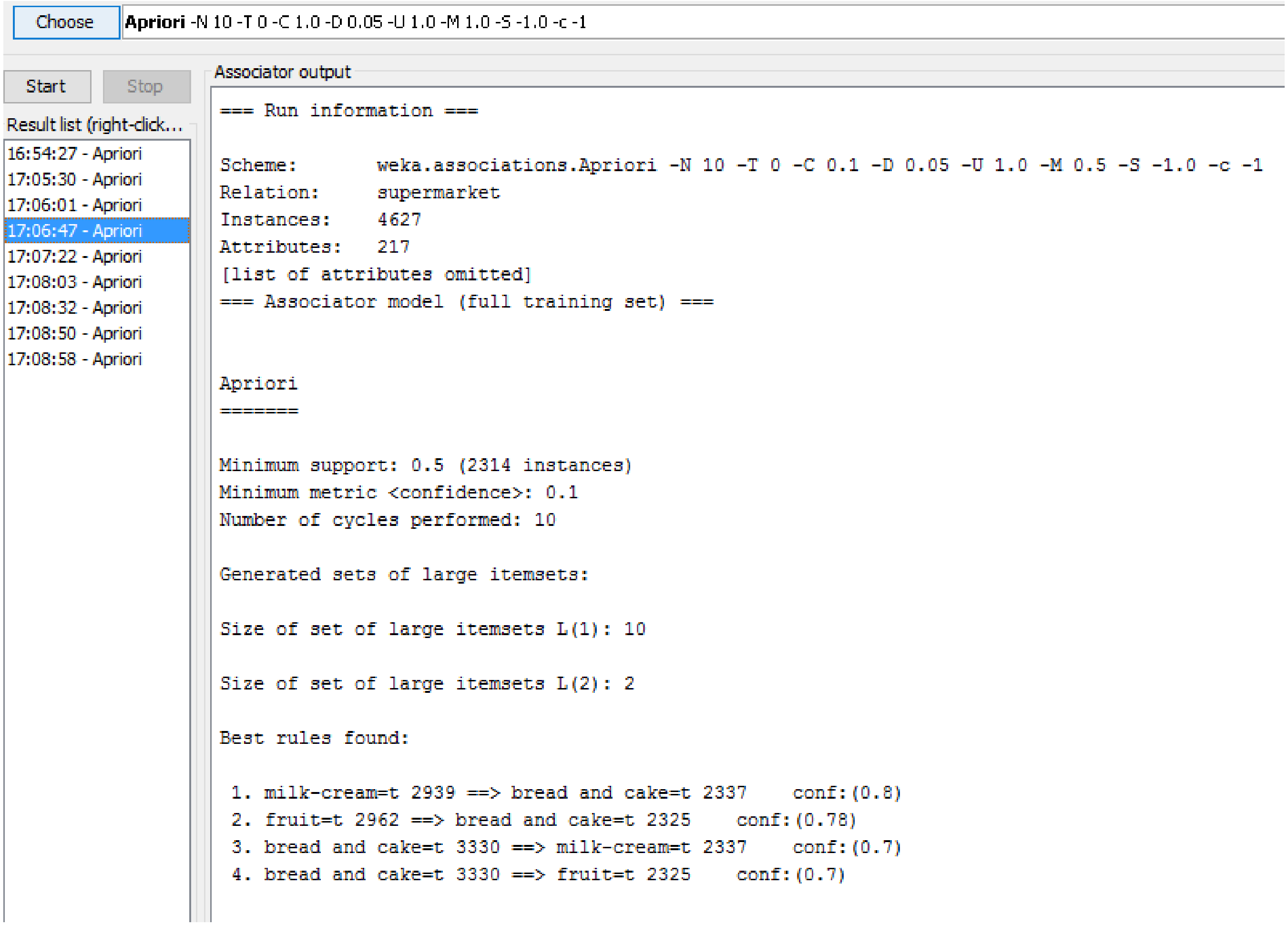


Apriori -N 10 -T 0 -C 1.0 -D 0.05 -U 1.0 -M 1.0 -S -1.0 -c -1

```
Scheme:       weka.associations.Apriori -N 10 -T 0 -C 0.5 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.45 (2082 instances)
Minimum metric <confidence>: 0.5
Number of cycles performed: 11

Generated sets of large itemsets:

Size of set of large itemsets L(1): 13

Size of set of large itemsets L(2): 7

Best rules found:

 1. biscuits=t 2605 ==> bread and cake=t 2083    conf:(0.8)
 2. milk-cream=t 2939 ==> bread and cake=t 2337    conf:(0.8)
 3. fruit=t 2962 ==> bread and cake=t 2325    conf:(0.78)
 4. baking needs=t 2795 ==> bread and cake=t 2191    conf:(0.78)
 5. frozen foods=t 2717 ==> bread and cake=t 2129    conf:(0.78)
 6. vegetables=t 2961 ==> bread and cake=t 2298    conf:(0.78)
 7. vegetables=t 2961 ==> fruit=t 2207    conf:(0.75)
 8. fruit=t 2962 ==> vegetables=t 2207    conf:(0.75)
 9. bread and cake=t 3330 ==> milk-cream=t 2337    conf:(0.7)
10. bread and cake=t 3330 ==> fruit=t 2325    conf:(0.7)
```



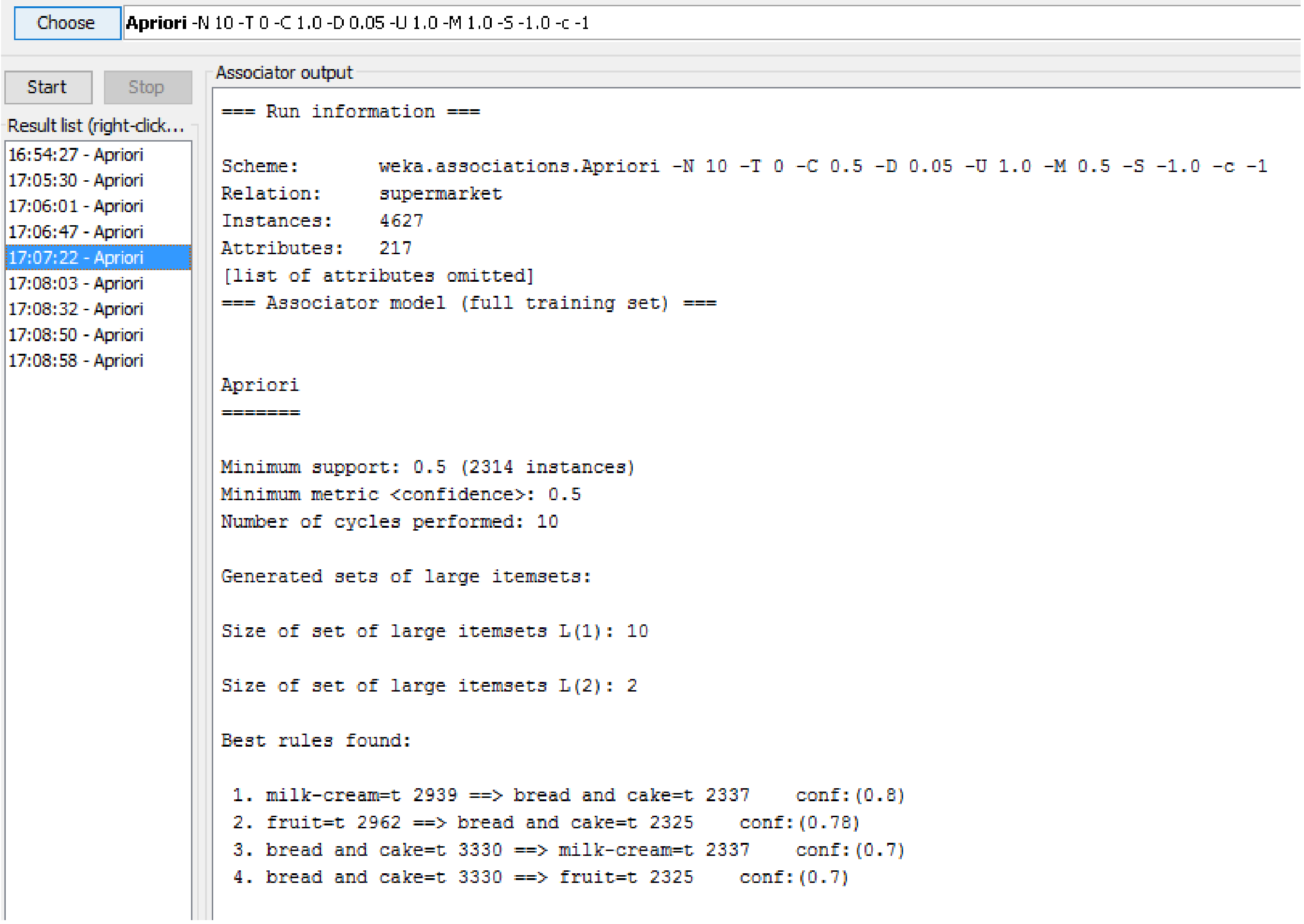
```
Scheme:       weka.associations.Apriori -N 10 -T 0 -C 1.0 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.1 (463 instances)
Minimum metric <confidence>: 1
Number of cycles performed: 18

Generated sets of large itemsets:

Size of set of large itemsets L(1): 52

Size of set of large itemsets L(2): 634

Size of set of large itemsets L(3): 2598

Size of set of large itemsets L(4): 3950

Size of set of large itemsets L(5): 2470

Size of set of large itemsets L(6): 558

Size of set of large itemsets L(7): 20

Best rules found:
```



Apriori -N 10 -T 0 -C 1.0 -D 0.05 -U 1.0 -M 1.0 -S -1.0 -c -1

16:54:27 - Apriori
17:05:30 - Apriori
17:06:01 - Apriori
17:06:47 - Apriori
17:07:22 - Apriori
17:08:03 - Apriori
17:08:32 - Apriori
17:08:50 - Apriori
17:08:58 - Apriori

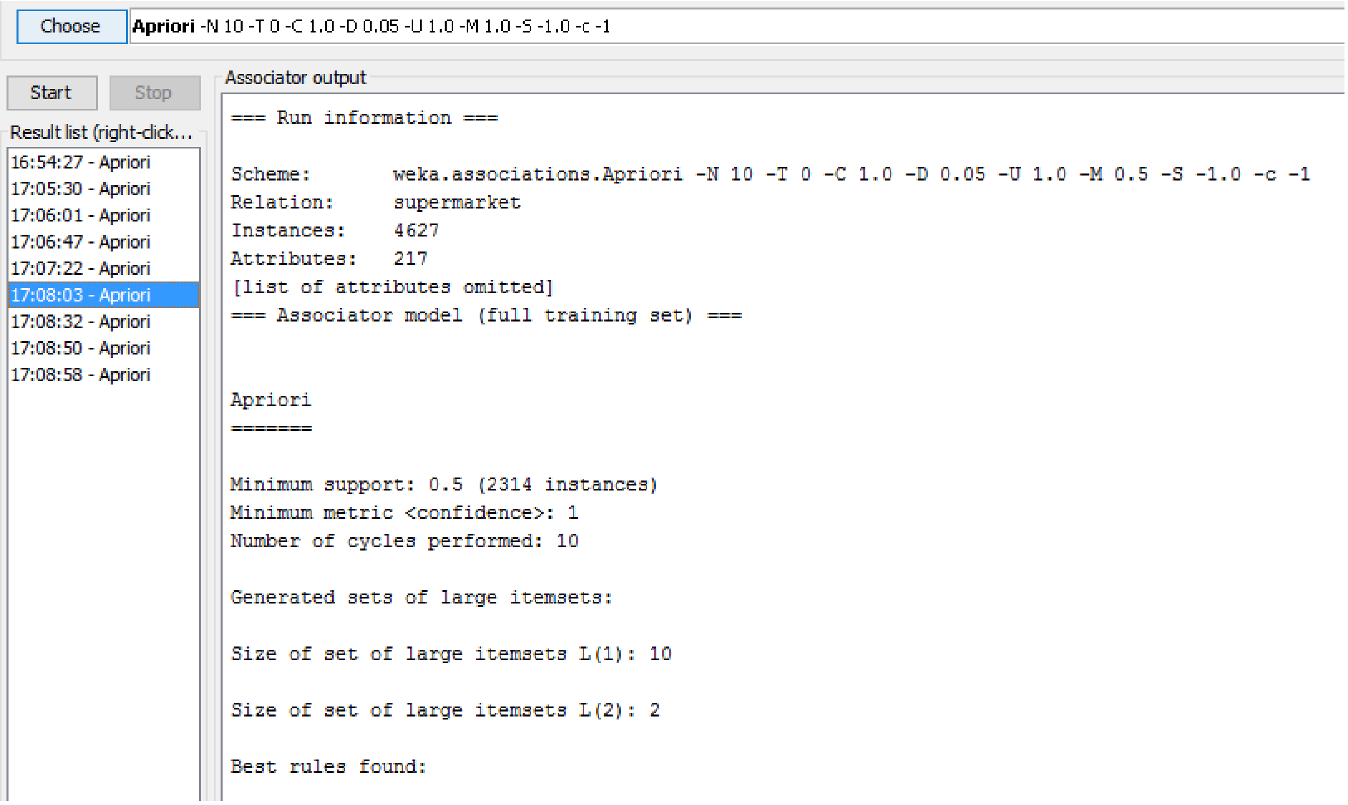
```
=== Run information ===

Scheme:       weka.associations.Apriori -N 10 -T 0 -C 0.1 -D 0.05 -U 1.0 -M 0.5 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.5 (2314 instances)
Minimum metric <confidence>: 0.1
Number of cycles performed: 10

Generated sets of large itemsets:

Size of set of large itemsets L(1): 10

Size of set of large itemsets L(2): 2

Best rules found:

 1. milk-cream=t 2939 ==> bread and cake=t 2337    conf:(0.8)
 2. fruit=t 2962 ==> bread and cake=t 2325    conf:(0.78)
 3. bread and cake=t 3330 ==> milk-cream=t 2337    conf:(0.7)
 4. bread and cake=t 3330 ==> fruit=t 2325    conf:(0.7)
```



Apriori -N 10 -T 0 -C 1.0 -D 0.05 -U 1.0 -M 1.0 -S -1.0 -c -1

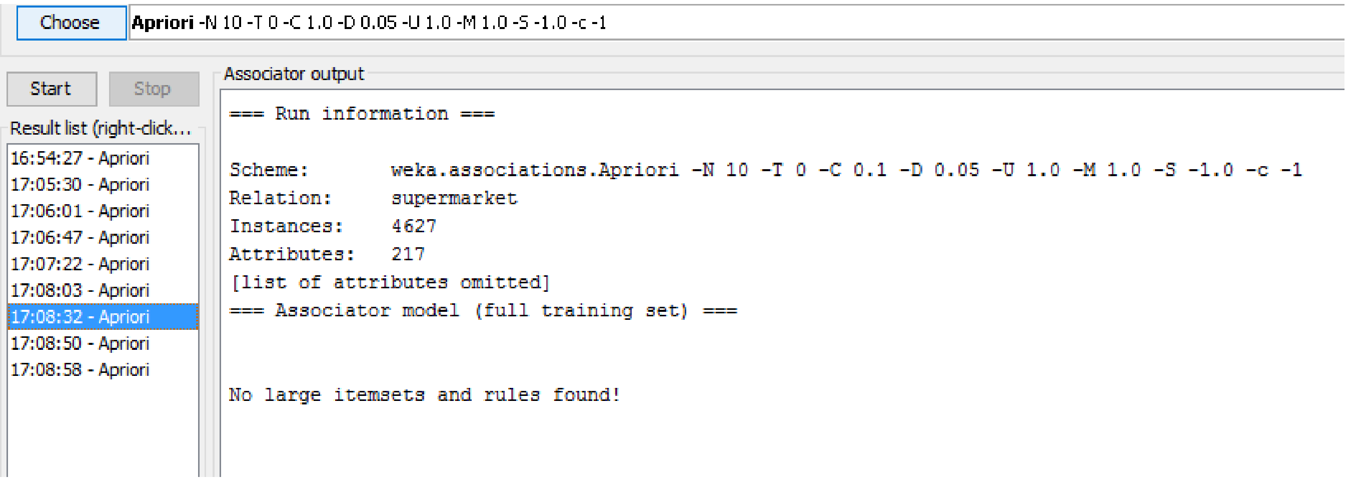
```
16:54:27 - Apriori
17:05:30 - Apriori
17:06:01 - Apriori
17:06:47 - Apriori
17:07:22 - Apriori
17:08:03 - Apriori
17:08:32 - Apriori
17:08:50 - Apriori
17:08:58 - Apriori
```

```
=== Run information ===

Scheme:       weka.associations.Apriori -N 10 -T 0 -C 0.5 -D 0.05 -U 1.0 -M 0.5 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.5 (2314 instances)
Minimum metric <confidence>: 0.5
Number of cycles performed: 10

Generated sets of large itemsets:

Size of set of large itemsets L(1): 10

Size of set of large itemsets L(2): 2

Best rules found:

 1. milk-cream=t 2939 ==> bread and cake=t 2337    conf:(0.8)
 2. fruit=t 2962 ==> bread and cake=t 2325    conf:(0.78)
 3. bread and cake=t 3330 ==> milk-cream=t 2337    conf:(0.7)
 4. bread and cake=t 3330 ==> fruit=t 2325    conf:(0.7)
```



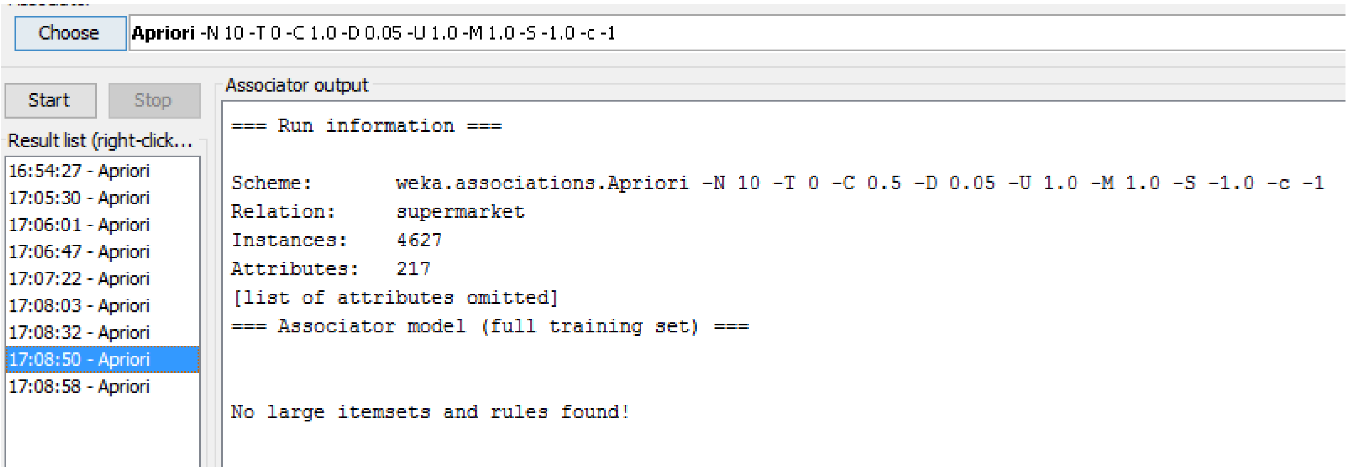
Apriori -N 10 -T 0 -C 1.0 -D 0.05 -U 1.0 -M 1.0 -S -1.0 -c -1

```
=== Run information ===

Scheme:       weka.associations.Apriori -N 10 -T 0 -C 1.0 -D 0.05 -U 1.0 -M 0.5 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.5 (2314 instances)
Minimum metric <confidence>: 1
Number of cycles performed: 10

Generated sets of large itemsets:

Size of set of large itemsets L(1): 10

Size of set of large itemsets L(2): 2

Best rules found:
```



Apriori -N 10 -T 0 -C 1.0 -D 0.05 -U 1.0 -M 1.0 -S -1.0 -c -1

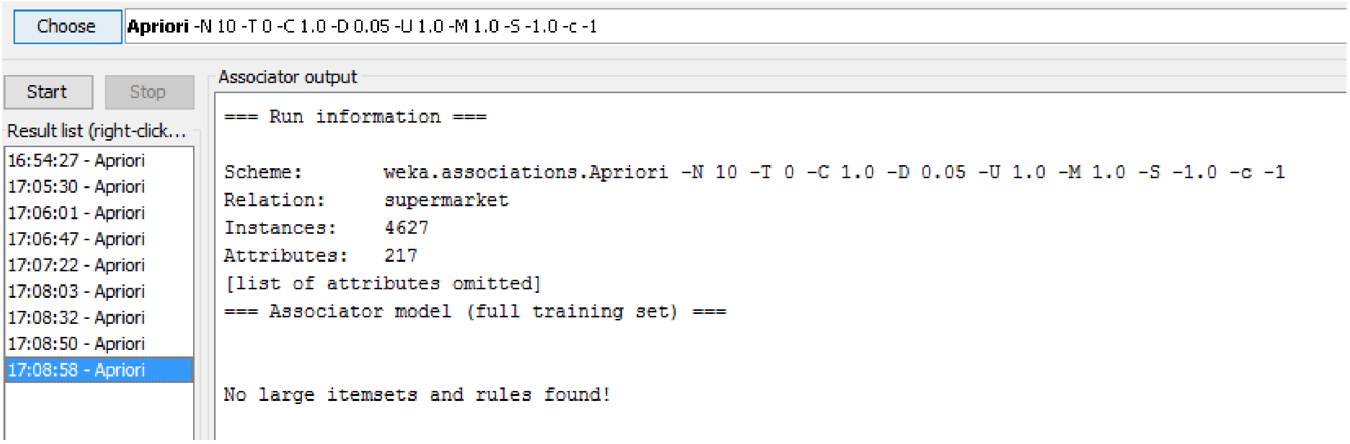
```
=== Run information ===

Scheme:       weka.associations.Apriori -N 10 -T 0 -C 0.1 -D 0.05 -U 1.0 -M 1.0 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


No large itemsets and rules found!
```



Apriori -N 10 -T 0 -C 1.0 -D 0.05 -U 1.0 -M 1.0 -S -1.0 -c -1

16:54:27 - Apriori
17:05:30 - Apriori
17:06:01 - Apriori
17:06:47 - Apriori
17:07:22 - Apriori
17:08:03 - Apriori
17:08:32 - Apriori
17:08:50 - Apriori
17:08:58 - Apriori

```
=== Run information ===

Scheme:       weka.associations.Apriori -N 10 -T 0 -C 0.5 -D 0.05 -U 1.0 -M 1.0 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


No large itemsets and rules found!
```



Apriori -N 10 -T 0 -C 1.0 -D 0.05 -U 1.0 -M 1.0 -S -1.0 -c -1

16:54:27 - Apriori
17:05:30 - Apriori
17:06:01 - Apriori
17:06:47 - Apriori
17:07:22 - Apriori
17:08:03 - Apriori
17:08:32 - Apriori
17:08:50 - Apriori
17:08:58 - Apriori

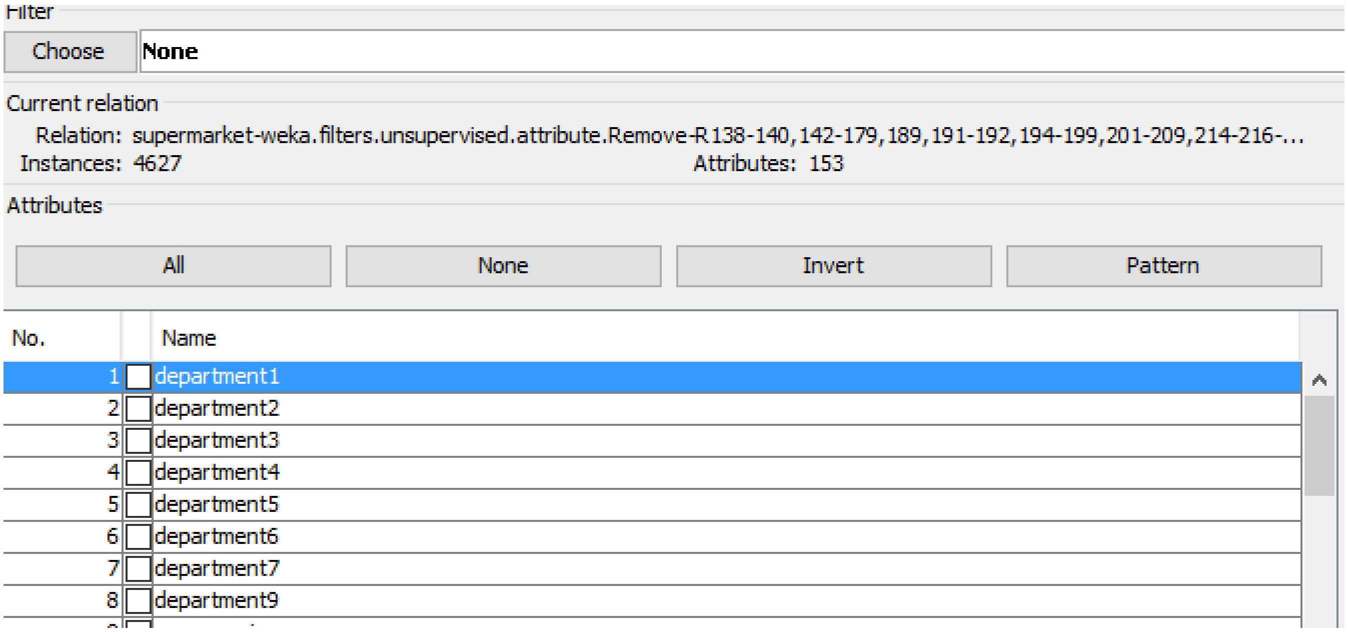
```
=== Run information ===

Scheme:       weka.associations.Apriori -N 10 -T 0 -C 1.0 -D 0.05 -U 1.0 -M 1.0 -S -1.0 -c -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
=== Associator model (full training set) ===


No large itemsets and rules found!
```

Inference from the above screenshots:

- No rules are formed when minmetric – 1.0 (high confidence level)
- No rules/Large itemsets are formed when lowerboundsupport = high (M = 1.0)
- Only 4 set of rules are formed that too with 80% of confidence when lowerboundsupport is at medium (M = 0.5)
- 80 % is achieved when lowerboundsupport (M=0.1) at low level and confidence is anywhere between 10% to 80%

Test case 13: - What happens when we remove the most frequent items and run against Apriori algorithm.

- Remove some frequent items (Biscuits and frozen foods) along with the ones that have no transactions.



Filter

Choose None

Current relation

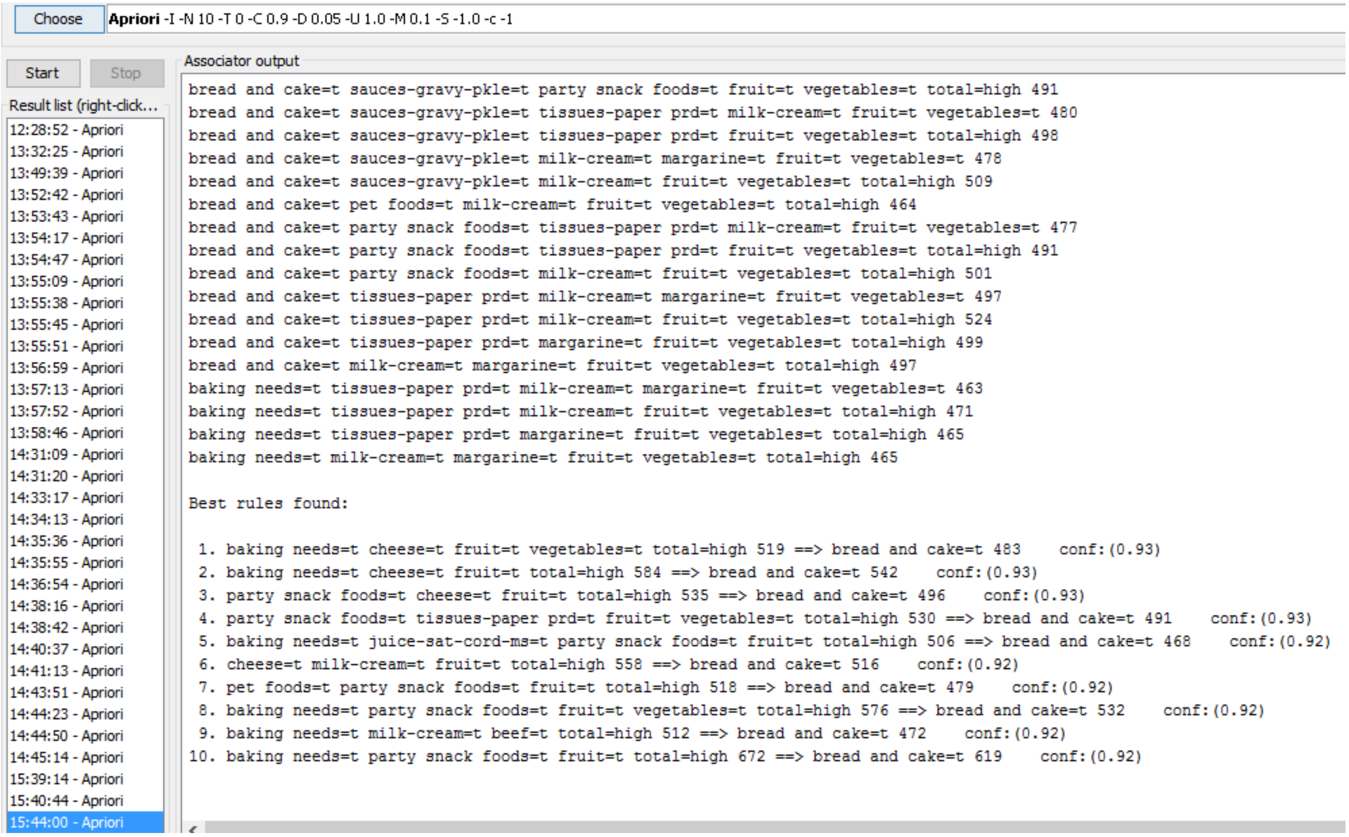
Relation: supermarket-weka.filters.unsupervised.attribute.Remove-R138-140,142-179,189,191-192,194-199,201-209,214-216-...

Instances: 4627 Attributes: 153

Attributes

No.		Name
1	□	department1
2	□	department2
3	□	department3
4	□	department4
5	□	department5
6	□	department6
7	□	department7
8	□	department9

Run the association rule under Apriori Algorithm with its default parameters:



Choose Apriori -I -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

12:28:52 - Apriori
13:32:25 - Apriori
13:49:39 - Apriori
13:52:42 - Apriori
13:53:43 - Apriori
13:54:17 - Apriori
13:54:47 - Apriori
13:55:09 - Apriori
13:55:38 - Apriori
13:55:45 - Apriori
13:55:51 - Apriori
13:56:59 - Apriori
13:57:13 - Apriori
13:57:52 - Apriori
13:58:46 - Apriori
14:31:09 - Apriori
14:31:20 - Apriori
14:33:17 - Apriori
14:34:13 - Apriori
14:35:36 - Apriori
14:35:55 - Apriori
14:36:54 - Apriori
14:38:16 - Apriori
14:38:42 - Apriori
14:40:37 - Apriori
14:41:13 - Apriori
14:43:51 - Apriori
14:44:23 - Apriori
14:44:50 - Apriori
14:45:14 - Apriori
15:39:14 - Apriori
15:40:44 - Apriori
15:44:00 - Apriori

```
bread and cake=t sauces-gravy-pkle=t party snack foods=t fruit=t vegetables=t total=high 491
bread and cake=t sauces-gravy-pkle=t tissues-paper prd=t milk-cream=t fruit=t vegetables=t 480
bread and cake=t sauces-gravy-pkle=t tissues-paper prd=t fruit=t vegetables=t total=high 498
bread and cake=t sauces-gravy-pkle=t milk-cream=t margarine=t fruit=t vegetables=t 478
bread and cake=t sauces-gravy-pkle=t milk-cream=t fruit=t vegetables=t total=high 509
bread and cake=t pet foods=t milk-cream=t fruit=t vegetables=t total=high 464
bread and cake=t party snack foods=t tissues-paper prd=t milk-cream=t fruit=t vegetables=t 477
bread and cake=t party snack foods=t tissues-paper prd=t fruit=t vegetables=t total=high 491
bread and cake=t party snack foods=t milk-cream=t fruit=t vegetables=t total=high 501
bread and cake=t tissues-paper prd=t milk-cream=t margarine=t fruit=t vegetables=t 497
bread and cake=t tissues-paper prd=t milk-cream=t fruit=t vegetables=t total=high 524
bread and cake=t tissues-paper prd=t margarine=t fruit=t vegetables=t total=high 499
bread and cake=t milk-cream=t margarine=t fruit=t vegetables=t total=high 497
baking needs=t tissues-paper prd=t milk-cream=t margarine=t fruit=t vegetables=t 463
baking needs=t tissues-paper prd=t milk-cream=t fruit=t vegetables=t total=high 471
baking needs=t tissues-paper prd=t margarine=t fruit=t vegetables=t total=high 465
baking needs=t milk-cream=t margarine=t fruit=t vegetables=t total=high 465

Best rules found:

 1. baking needs=t cheese=t fruit=t vegetables=t total=high 519 ==> bread and cake=t 483    conf:(0.93)
 2. baking needs=t cheese=t fruit=t total=high 584 ==> bread and cake=t 542    conf:(0.93)
 3. party snack foods=t cheese=t fruit=t total=high 535 ==> bread and cake=t 496    conf:(0.93)
 4. party snack foods=t tissues-paper prd=t fruit=t vegetables=t total=high 530 ==> bread and cake=t 491    conf:(0.93)
 5. baking needs=t juice-sat-cord-ms=t party snack foods=t fruit=t total=high 506 ==> bread and cake=t 468    conf:(0.92)
 6. cheese=t milk-cream=t fruit=t total=high 558 ==> bread and cake=t 516    conf:(0.92)
 7. pet foods=t party snack foods=t fruit=t total=high 518 ==> bread and cake=t 479    conf:(0.92)
 8. baking needs=t party snack foods=t fruit=t vegetables=t total=high 576 ==> bread and cake=t 532    conf:(0.92)
 9. baking needs=t milk-cream=t beef=t total=high 512 ==> bread and cake=t 472    conf:(0.92)
10. baking needs=t party snack foods=t fruit=t total=high 672 ==> bread and cake=t 619    conf:(0.92)
```

Inference from the above test:

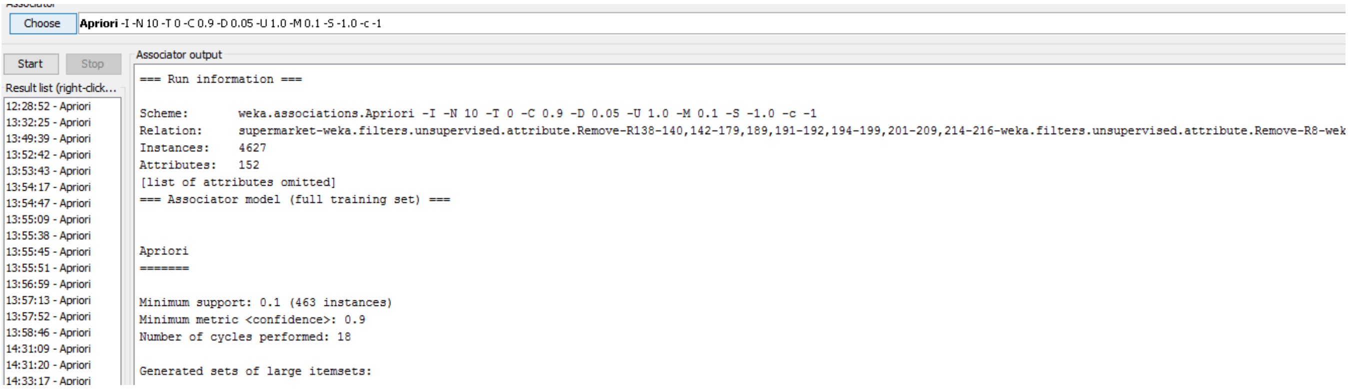
- Here, we have removed Biscuits and Frozen foods as they are most frequently used/bought items.

- Once after removing these frequently used items, we could see no change in number of cycles performed or the minbound support. But % of confidence is increased by 1.

- When all the frequently used items are included and run against Apriori with default parameters, the % of confidence was 92%.

- Here, when mostl frequently used items are deleted/removed and run against Apriori, the % of confidence is 93% (increased by a percentage).

Number of cycles = 18



Apriori -I -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     supermarket-weka.filters.unsupervised.attribute.Remove-R138-140,142-179,189,191-192,194-199,201-209,214-216-weka.filters.unsupervised.attribute.Remove-R8-wek
Instances:    4627
Attributes:   152
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.1 (463 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 18

Generated sets of large itemsets:
```

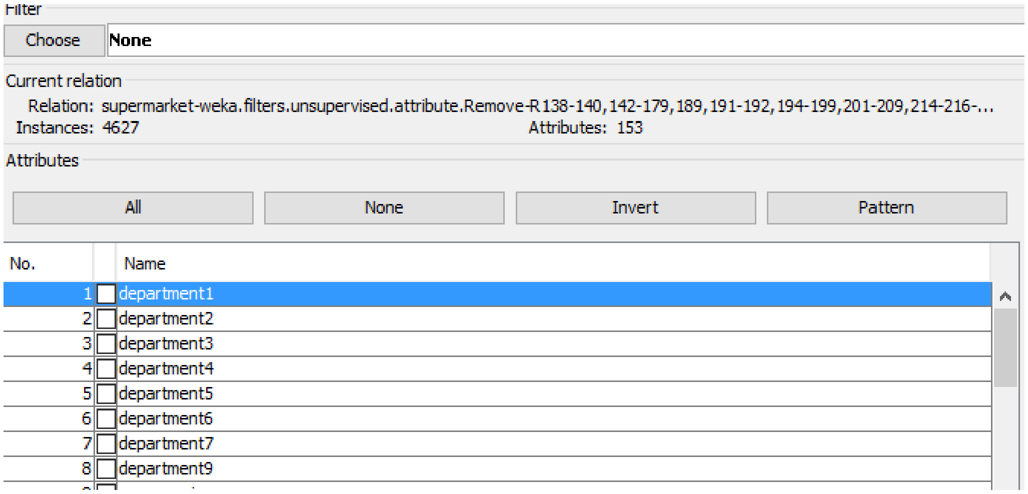
Minimum metric of confidence level = 0.9

Minimum support level = 0.1 = 463 instances in this case.

Large itemsets are generated without Biscuits and frozen foods.

Test case 14: - What happens when we remove the most frequent items from the test case above and run against Apriori algorithm without changing its parameters.

- Remove some frequent items (fruits) along with the ones that have no transactions.

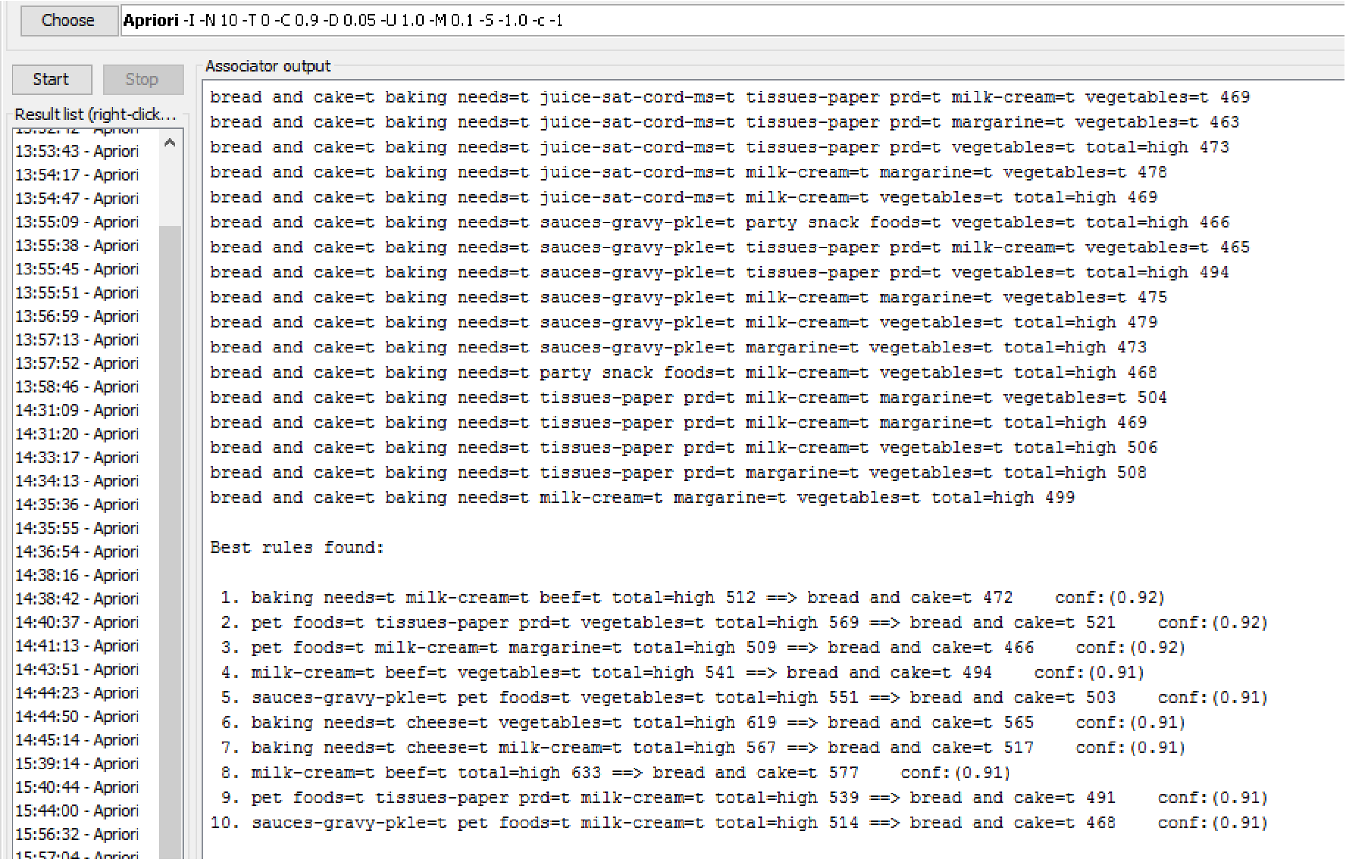


None

Relation: supermarket-weka.filters.unsupervised.attribute.Remove-R138-140,142-179,189,191-192,194-199,201-209,214-216-...
Instances: 4627 Attributes: 153

No.		Name
1	□	department1
2	□	department2
3	□	department3
4	□	department4
5	□	department5
6	□	department6
7	□	department7
8	□	department9

Run the association rule under Apriori Algorithm with its default parameters:



Apriori -I -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

```
13:53:43 - Apriori
13:54:17 - Apriori
13:54:47 - Apriori
13:55:09 - Apriori
13:55:38 - Apriori
13:55:45 - Apriori
13:55:51 - Apriori
13:56:59 - Apriori
13:57:13 - Apriori
13:57:52 - Apriori
13:58:46 - Apriori
14:31:09 - Apriori
14:31:20 - Apriori
14:33:17 - Apriori
14:34:13 - Apriori
14:35:36 - Apriori
14:35:55 - Apriori
14:36:54 - Apriori
14:38:16 - Apriori
14:38:42 - Apriori
14:40:37 - Apriori
14:41:13 - Apriori
14:43:51 - Apriori
14:44:23 - Apriori
14:44:50 - Apriori
14:45:14 - Apriori
15:39:14 - Apriori
15:40:44 - Apriori
15:44:00 - Apriori
15:56:32 - Apriori
```

```
bread and cake=t baking needs=t juice-sat-cord-ms=t tissues-paper prd=t milk-cream=t vegetables=t 469
bread and cake=t baking needs=t juice-sat-cord-ms=t tissues-paper prd=t margarine=t vegetables=t 463
bread and cake=t baking needs=t juice-sat-cord-ms=t tissues-paper prd=t vegetables=t total=high 473
bread and cake=t baking needs=t juice-sat-cord-ms=t milk-cream=t margarine=t vegetables=t 478
bread and cake=t baking needs=t juice-sat-cord-ms=t milk-cream=t vegetables=t total=high 469
bread and cake=t baking needs=t sauces-gravy-pkle=t party snack foods=t vegetables=t total=high 466
bread and cake=t baking needs=t sauces-gravy-pkle=t tissues-paper prd=t milk-cream=t vegetables=t 465
bread and cake=t baking needs=t sauces-gravy-pkle=t tissues-paper prd=t vegetables=t total=high 494
bread and cake=t baking needs=t sauces-gravy-pkle=t milk-cream=t margarine=t vegetables=t 475
bread and cake=t baking needs=t sauces-gravy-pkle=t milk-cream=t vegetables=t total=high 479
bread and cake=t baking needs=t sauces-gravy-pkle=t margarine=t vegetables=t total=high 473
bread and cake=t baking needs=t party snack foods=t milk-cream=t vegetables=t total=high 468
bread and cake=t baking needs=t tissues-paper prd=t milk-cream=t margarine=t vegetables=t 504
bread and cake=t baking needs=t tissues-paper prd=t milk-cream=t margarine=t total=high 469
bread and cake=t baking needs=t tissues-paper prd=t milk-cream=t vegetables=t total=high 506
bread and cake=t baking needs=t tissues-paper prd=t margarine=t vegetables=t total=high 508
bread and cake=t baking needs=t milk-cream=t margarine=t vegetables=t total=high 499

Best rules found:

 1. baking needs=t milk-cream=t beef=t total=high 512 ==> bread and cake=t 472    conf:(0.92)
 2. pet foods=t tissues-paper prd=t vegetables=t total=high 569 ==> bread and cake=t 521    conf:(0.92)
 3. pet foods=t milk-cream=t margarine=t total=high 509 ==> bread and cake=t 466    conf:(0.92)
 4. milk-cream=t beef=t vegetables=t total=high 541 ==> bread and cake=t 494    conf:(0.91)
 5. sauces-gravy-pkle=t pet foods=t vegetables=t total=high 551 ==> bread and cake=t 503    conf:(0.91)
 6. baking needs=t cheese=t vegetables=t total=high 619 ==> bread and cake=t 565    conf:(0.91)
 7. baking needs=t cheese=t milk-cream=t total=high 567 ==> bread and cake=t 517    conf:(0.91)
 8. milk-cream=t beef=t total=high 633 ==> bread and cake=t 577    conf:(0.91)
 9. pet foods=t tissues-paper prd=t milk-cream=t total=high 539 ==> bread and cake=t 491    conf:(0.91)
10. sauces-gravy-pkle=t pet foods=t milk-cream=t total=high 514 ==> bread and cake=t 468    conf:(0.91)
```

Inference from the above test:

- Here, we have removed fruits --- > this is the main item of all as it is occurred the most frequent items in all itemsets. ---> Every customer in the marketplace gets fruits.

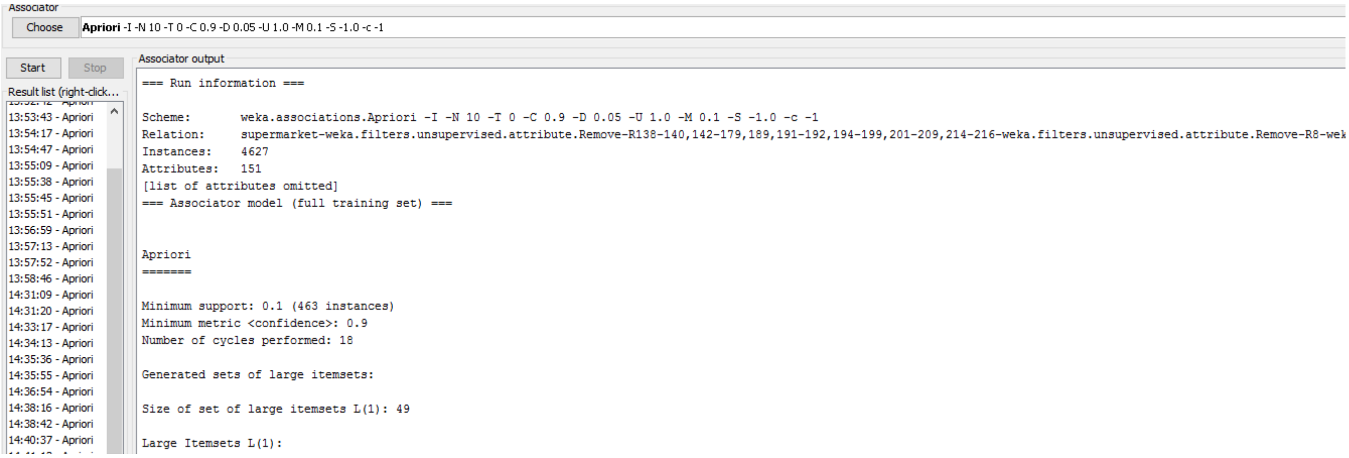
- Once after removing this frequently used items, we could see no change in number of cycles performed or the minbound support. But % of confidence is DECREASED by 1%.

- When all the frequently used items are included and run against Apriori with default parameters, the % of confidence was 92%.

- When most frequently used items such as bisucits and frozen foods are deleted/removed and run against Apriori, the % of confidence is 93% (increased by a percentage).

- Here when the most frequently used item - FRUITS are deleted/removed and run against Apriori, the % of confidence is 92% (decreased by a percentage).

Number of cycles = 18



Apriori -I -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

```
13:53:43 - Apriori
13:54:17 - Apriori
13:54:47 - Apriori
13:55:09 - Apriori
13:55:38 - Apriori
13:55:45 - Apriori
13:55:51 - Apriori
13:56:59 - Apriori
13:57:13 - Apriori
13:57:52 - Apriori
13:58:46 - Apriori
14:31:09 - Apriori
14:31:20 - Apriori
14:33:17 - Apriori
14:34:13 - Apriori
14:35:36 - Apriori
14:35:55 - Apriori
14:36:54 - Apriori
14:38:16 - Apriori
14:38:42 - Apriori
14:40:37 - Apriori
```

```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     supermarket-weka.filters.unsupervised.attribute.Remove-R138-140,142-179,189,191-192,194-199,201-209,214-216-weka.filters.unsupervised.attribute.Remove-R8-wek
Instances:    4627
Attributes:   151
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.1 (463 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 18

Generated sets of large itemsets:

Size of set of large itemsets L(1): 49

Large Itemsets L(1):
```

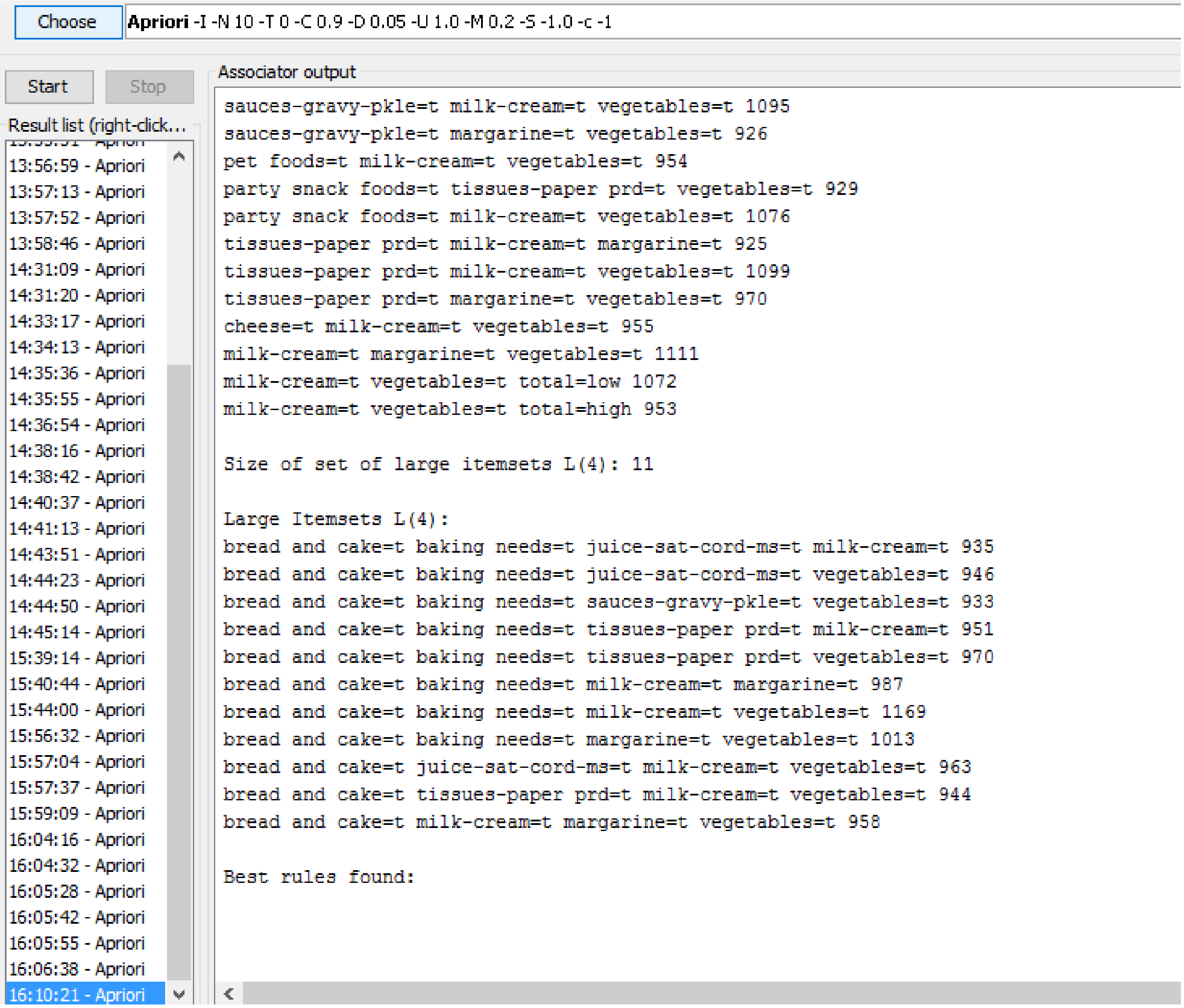
Minimum metric of confidence level = 0.9

Minimum support level = 0.1 = 463 instances in this case.

Large itemsets are generated without Fruits.

Test case 15: - What happens when we remove the most frequent items from the test case above and run against Apriori algorithm after changing its parameters:

- Remove some frequent items (fruits) along with the ones that have no transactions.
- Change lowerbound support = 0.2 and minmetric = 0.9(let this remain unchanged).



```
Apriori -I -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.2 -S -1.0 -c -1
```

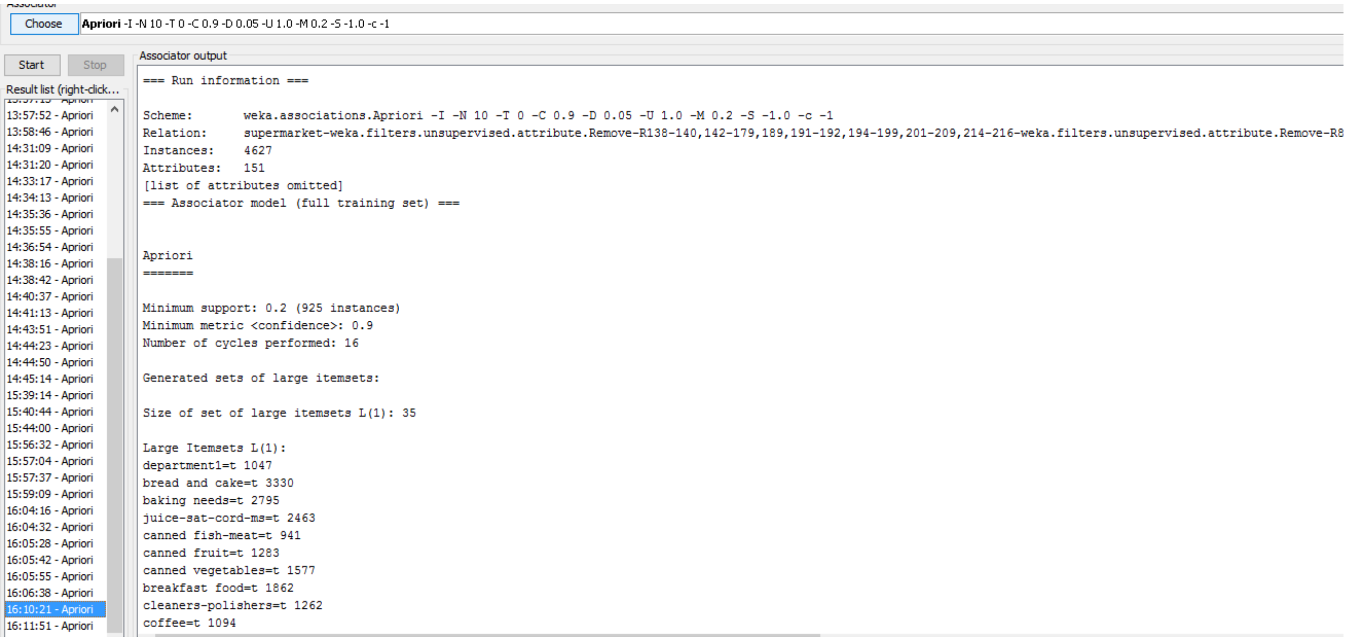
```
sauces-gravy-pkle=t milk-cream=t vegetables=t 1095
sauces-gravy-pkle=t margarine=t vegetables=t 926
pet foods=t milk-cream=t vegetables=t 954
party snack foods=t tissues-paper prd=t vegetables=t 929
party snack foods=t milk-cream=t vegetables=t 1076
tissues-paper prd=t milk-cream=t margarine=t 925
tissues-paper prd=t milk-cream=t vegetables=t 1099
tissues-paper prd=t margarine=t vegetables=t 970
cheese=t milk-cream=t vegetables=t 955
milk-cream=t margarine=t vegetables=t 1111
milk-cream=t vegetables=t total=low 1072
milk-cream=t vegetables=t total=high 953

Size of set of large itemsets L(4): 11

Large Itemsets L(4):
bread and cake=t baking needs=t juice-sat-cord-ms=t milk-cream=t 935
bread and cake=t baking needs=t juice-sat-cord-ms=t vegetables=t 946
bread and cake=t baking needs=t sauces-gravy-pkle=t vegetables=t 933
bread and cake=t baking needs=t tissues-paper prd=t milk-cream=t 951
bread and cake=t baking needs=t tissues-paper prd=t vegetables=t 970
bread and cake=t baking needs=t milk-cream=t margarine=t 987
bread and cake=t baking needs=t milk-cream=t vegetables=t 1169
bread and cake=t baking needs=t margarine=t vegetables=t 1013
bread and cake=t juice-sat-cord-ms=t milk-cream=t vegetables=t 963
bread and cake=t tissues-paper prd=t milk-cream=t vegetables=t 944
bread and cake=t milk-cream=t margarine=t vegetables=t 958

Best rules found:
```

Inference from the above:

No best rule is thrown for minmetric = 90% on a lowsupport min = 20% after removing the most frequently used items Fruits.

Number of cycles performed : 16 cycles performed (for 20% minsupport bound)

Instances: 925



Apriori -I -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.2 -S -1.0 -c -1

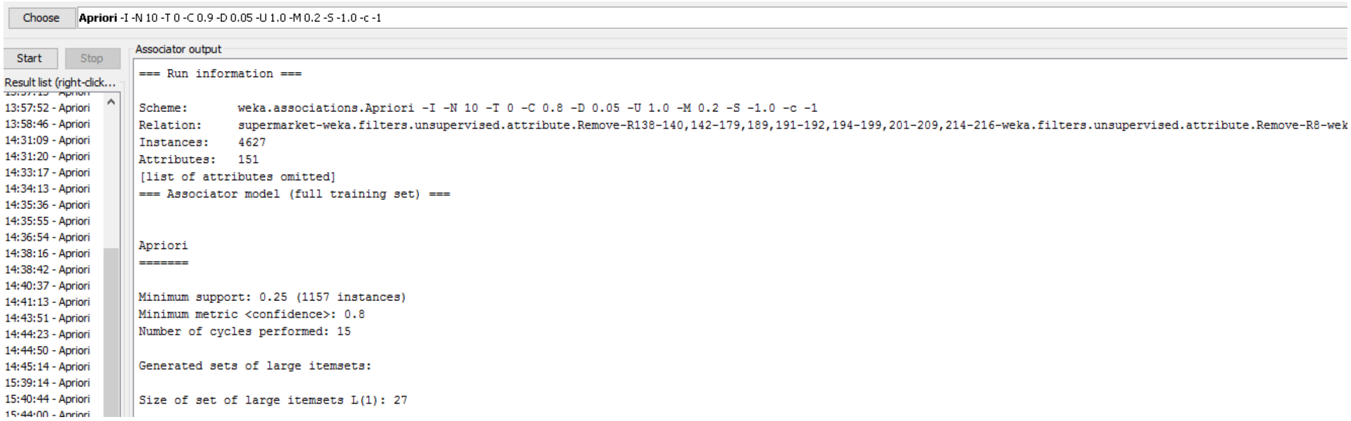
```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.2 -S -1.0 -c -1
Relation:     supermarket-weka.filters.unsupervised.attribute.Remove-R138-140,142-179,189,191-192,194-199,201-209,214-216-weka.filters.unsupervised.attribute.Remove-R8
Instances:    4627
Attributes:   151
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.2 (925 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 16

Generated sets of large itemsets:

Size of set of large itemsets L(1): 35

Large Itemsets L(1):
department1=t 1047
bread and cake=t 3330
baking needs=t 2795
juice-sat-cord-ms=t 2463
canned fish-meat=t 941
canned fruit=t 1283
canned vegetables=t 1577
breakfast food=t 1862
cleaners-polishers=t 1262
coffee=t 1094
```

Here rule is not based on Fruits but instead its based on the next available frequent item listed in the itemset.

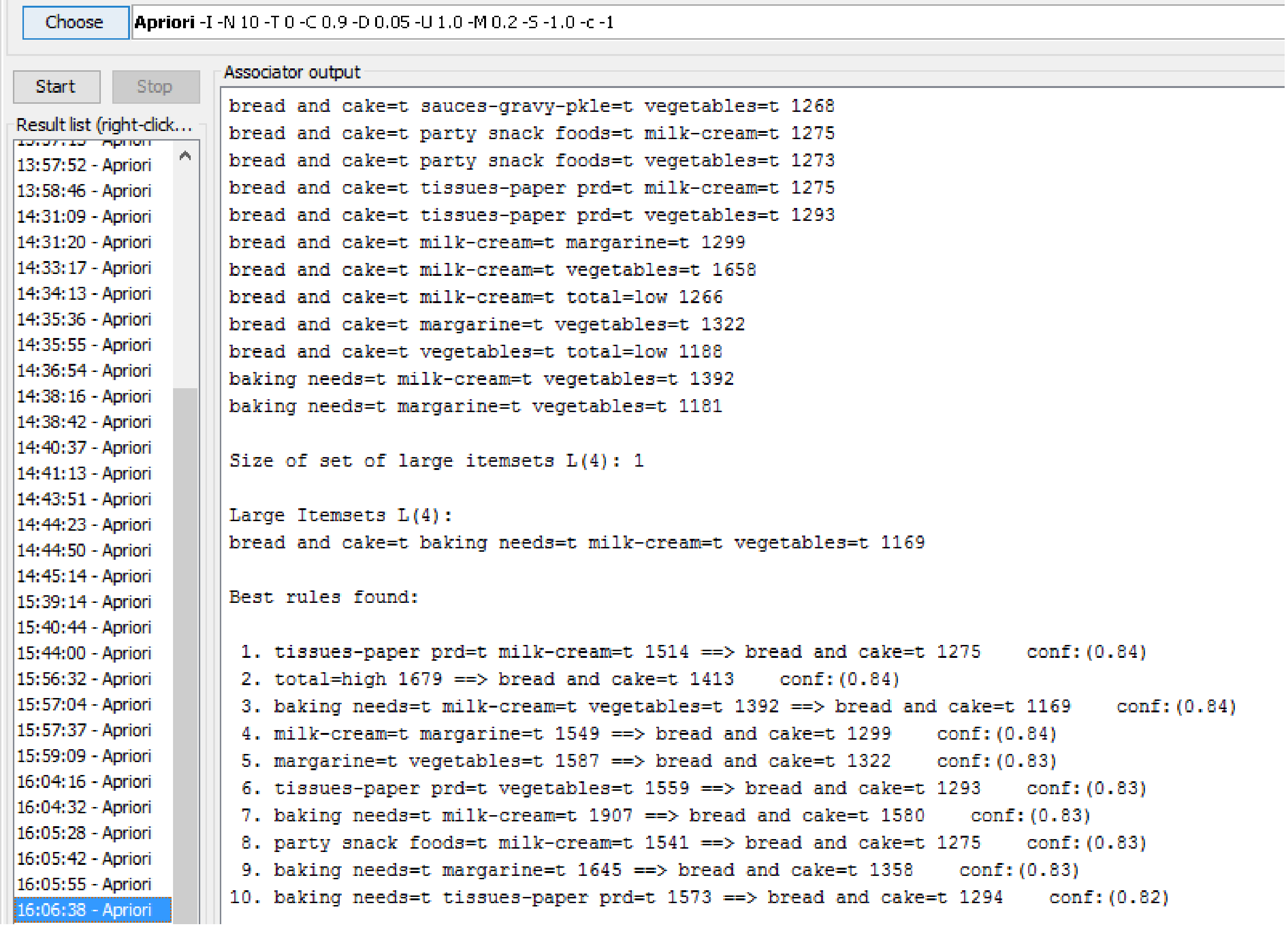
After removing the most frequent item from the itemlist.

90% of confidence is not achieved when minsupport lower bound is changed from 10% to 20% or above.

When lowerbound support = 0.2 and minmetric = 0.8 (changed from 90% to 80%).



```
=== Run information ===

Scheme:       weka.associations.Apriori -I -N 10 -T 0 -C 0.8 -D 0.05 -U 1.0 -M 0.2 -S -1.0 -c -1
Relation:     supermarket-weka.filters.unsupervised.attribute.Remove-R138-140,142-179,189,191-192,194-199,201-209,214-216-weka.filters.unsupervised.attribute.Remove-R8-wek
Instances:    4627
Attributes:   151
[list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.25 (1157 instances)
Minimum metric <confidence>: 0.8
Number of cycles performed: 15

Generated sets of large itemsets:

Size of set of large itemsets L(1): 27
```

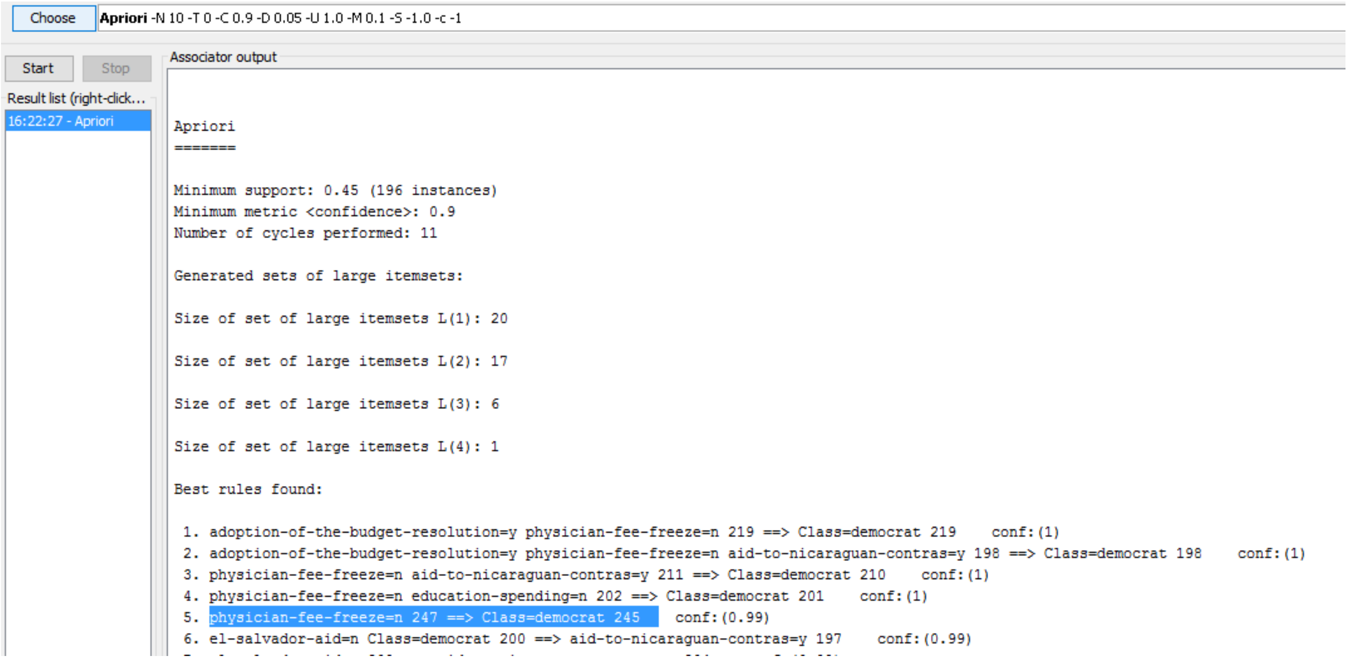


```
bread and cake=t sauces-gravy-pkle=t vegetables=t 1268
bread and cake=t party snack foods=t milk-cream=t 1275
bread and cake=t party snack foods=t vegetables=t 1273
bread and cake=t tissues-paper prd=t milk-cream=t 1275
bread and cake=t tissues-paper prd=t vegetables=t 1293
bread and cake=t milk-cream=t margarine=t 1299
bread and cake=t milk-cream=t vegetables=t 1658
bread and cake=t milk-cream=t total=low 1266
bread and cake=t margarine=t vegetables=t 1322
bread and cake=t vegetables=t total=low 1188
baking needs=t milk-cream=t vegetables=t 1392
baking needs=t margarine=t vegetables=t 1181

Size of set of large itemsets L(4): 1

Large Itemsets L(4):
bread and cake=t baking needs=t milk-cream=t vegetables=t 1169

Best rules found:

 1. tissues-paper prd=t milk-cream=t 1514 ==> bread and cake=t 1275    conf:(0.84)
 2. total=high 1679 ==> bread and cake=t 1413    conf:(0.84)
 3. baking needs=t milk-cream=t vegetables=t 1392 ==> bread and cake=t 1169    conf:(0.84)
 4. milk-cream=t margarine=t 1549 ==> bread and cake=t 1299    conf:(0.84)
 5. margarine=t vegetables=t 1587 ==> bread and cake=t 1322    conf:(0.83)
 6. tissues-paper prd=t vegetables=t 1559 ==> bread and cake=t 1293    conf:(0.83)
 7. baking needs=t milk-cream=t 1907 ==> bread and cake=t 1580    conf:(0.83)
 8. party snack foods=t milk-cream=t 1541 ==> bread and cake=t 1275    conf:(0.83)
 9. baking needs=t margarine=t 1645 ==> bread and cake=t 1358    conf:(0.83)
10. baking needs=t tissues-paper prd=t 1573 ==> bread and cake=t 1294    conf:(0.82)
```

- Hence from the above screenshot, we can say that when confidence level is decreased or set to minimum confidence levels, rules are decided and framed accordingly.

VOTE DATASET:

Vote dataset when run against association rule mining under Apriori algorithm



Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

```
Apriori
=======

Minimum support: 0.45 (196 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 11

Generated sets of large itemsets:

Size of set of large itemsets L(1): 20

Size of set of large itemsets L(2): 17

Size of set of large itemsets L(3): 6

Size of set of large itemsets L(4): 1

Best rules found:

 1. adoption-of-the-budget-resolution=y physician-fee-freeze=n 219 ==> Class=democrat 219    conf:(1)
 2. adoption-of-the-budget-resolution=y physician-fee-freeze=n aid-to-nicaraguan-contras=y 198 ==> Class=democrat 198    conf:(1)
 3. physician-fee-freeze=n aid-to-nicaraguan-contras=y 211 ==> Class=democrat 210    conf:(1)
 4. physician-fee-freeze=n education-spending=n 202 ==> Class=democrat 201    conf:(1)
 5. physician-fee-freeze=n 247 ==> Class=democrat 245    conf:(0.99)
 6. el-salvador-aid=n Class=democrat 200 ==> aid-to-nicaraguan-contras=y 197    conf:(0.99)
```

- In some cases, the attribute physician-fee-freeze=n is considered to be the most frequent itemset here as all the 247 instances are grouped under Class –democrat (245 instances).
- Hence, the percentage of confidence level here achieved in 99%.
- People who are not physician-fee-freeze, they belong to Democrats (class).This is proved 99% of the data.
- Here, this is the only attribute which does not club/join any other attribute to find the % of confidence or to form the rule. Hence this is considered to be the most important attribute in forming the rule.
- 99% confidence is achieved here which implies 1% is having incorrect instances in the Dataset. Incorrect instances are 2.
- Class – Democrat – 245 instances. Not a Physician-fee-freeze = 247 instances and hence 2 instances are misclassified as NOT A Physician-fee-freeze.

When Physician-fee-freeze is combined with the below items

1. adoption-of-the-budget-resolution=y physician-fee-freeze=n 219 ==> Class=democrat 219 conf:(1)

2. adoption-of-the-budget-resolution=y physician-fee-freeze=n aid-to-nicaraguan-contras=y 198 ==> Class=democrat 198 conf:(1)

3. physician-fee-freeze=n aid-to-nicaraguan-contras=y 211 ==> Class=democrat 210 conf:(1)

4. physician-fee-freeze=n education-spending=n 202 ==> Class=democrat 201 conf:(1)

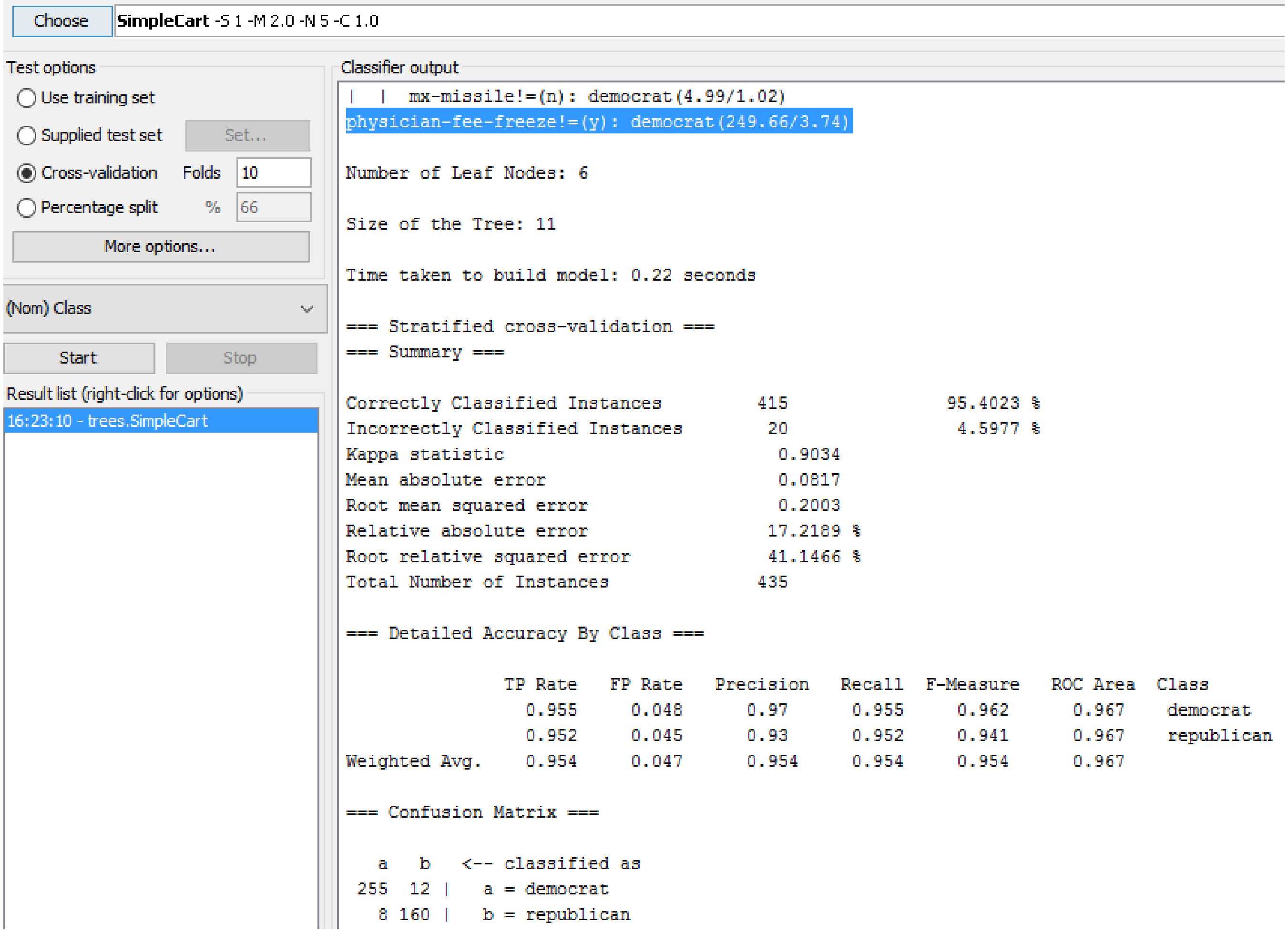
The above represents that when physician-fee-freeze is combined into groups of another attribute then the % of confidence gained is 100%.

This implies, from the 1st rule, when there is an adoption of the budget resolution, 219 people who are not a physician-fee-freeze belongs to Democrat category.

Simple CART:

In SimpleCART, the important attribute considered for the split is physician-fee-freeze.

Here, physician-fee-freeze=n, they also belong to democrat (249 instances)



SimpleCart -S 1 -M 2.0 -N 5 -C 1.0

Test options

○ Use training set

○ Supplied test set Set...

● Cross-validation Folds 10

○ Percentage split % 66

Result list (right-click for options)

16:23:10 - trees.SimpleCart

Classifier output

```
|  |  mx-missile!=(n): democrat(4.99/1.02)
physician-fee-freeze!=(y): democrat(249.66/3.74)

Number of Leaf Nodes: 6

Size of the Tree: 11

Time taken to build model: 0.22 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         415               95.4023 %
Incorrectly Classified Instances        20                4.5977 %
Kappa statistic                          0.9034
Mean absolute error                      0.0817
Root mean squared error                  0.2003
Relative absolute error                 17.2189 %
Root relative squared error             41.1466 %
Total Number of Instances              435

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall   F-Measure   ROC Area   Class
                 0.955     0.048     0.97       0.955     0.962      0.967     democrat
                 0.952     0.045     0.93       0.952     0.941      0.967     republican
Weighted Avg.    0.954     0.047     0.954      0.954     0.954      0.967

=== Confusion Matrix ===

   a   b   <-- classified as
 255  12 |   a = democrat
   8 160 |   b = republican
```

Here, physician-fee-freeze=n reported 247 instances, they also belong to democrat (249 instances)

Hence 2 instances are misclassified here as NOT a physician-fee-freeze.

% of confidence level achieved here based on this attribute will be same as the one obtained from the association rule.

CONCLUSION/SUMMARY:

- Attributes repeated – High confidence is gained when rules are formed.
- Minimum support is applied to gain the frequent itemsets.
- Frequent itemsets (after pruning) + minimum confidence constraint are used to form the best rules.
- While forming the rules, a certain threshold point is met for high confidence to be gained beyond which how many rules thus formed will reduce the percentage of Confidence.
- % of confidence when 1.0 (minmetric high) – makes no rule.
- If any rule is gained with 100% confidence (minmetric = 1.0) then 100% confidence is achieved with that combination.
- For Supermarket DS, 100% confidence is not achieved and the best rule relies on 90% of confidence (after forming the frequent Item sets)
- More rules to be generated, percentage of confidence keeps decreasing.
- Maximum number of cycles achieved at 100%, then number of cycles run = 18 (max number of cycles) when upperboundsupport = 1
- Max number of cycles achieved at 90% of confidence level = 17 cycles (when min metric = 0.9)
- When upperbound support is reduced, number of cycles is also reduced at confidence = 100%
- Number of instances keeps increasing and cycles performed keeps decreasing, when support bound ranges from 20 % or more (lowerbound) to 90(upperbound)
- At support bound 60% or above, no Large itemsets or rules are formed.
- Rules in supermarket formed based on 9% of confidence. Hence Data analyst will have to depend on 90% of the data they have (with 90% confidence level) and no assurance/guarantee provided.
- If the delta value increases, number of cycles performed will be low.
- Depending on the delta value, number of cycles decreases till certain level and then remains constant (No further decrease in number of cycles as it does not make any sense).
- When delta is slightly increased by 5% to 10%(or 20 or above) the rules thus formed have a greater % in confidence level than it was formed when Delta = 5% which had a confidence level of 92%.
- Ideal for supermarket DS best rules are formed when Delta = 10%
- Delta = 30% - 40% number of cycles = 3 and remains unchanged.
- Delta = 50 % - 80% number of cycles = 2 and remains unchanged.
- Delta = 90 and above, number of cycles = 1 remains unchanged.
- Frequent items -> Percentage of confidence is increased when 2 or more frequent used itemsets are deleted/removed.
- On the same hand, when a single item “fruit” in our example is removed (which is the most frequent item of all in the set), % of confidence decreases. This implies every customer in the supermarket buys Fruit.
- After removing frequent used items == > 90% of confidence is not achieved when minsupportlowerbound is changed from 10% to 20% or above.
- 80% of confidence is achieved when minsupportlowerbound is changed.
- No rules are formed when minmetric – 1.0 (high confidence level)
- No rules/Large itemsets are formed when lowerboundsupport = high (M = 1.0)
- Only 4 set of rules are formed that too with 80% of confidence when lowerboundsupport is at medium (M = 0.5)
- 80 % is achieved when lowerboundsupport (M=0.1) at low level and confidence is anywhere between 10% to 80%.