Data Mining Project 1

Abhav Luthra, Tanmay Singh, Krishna Sehgal

AIM

Implementation of Apriori Algorithm to generate frequent item-sets on gene-expression data-set. Frequent item sets generated by Apriori Algorithm will be used to generate Association Rules, which will predict occurrence of a disease based on the occurrences of gene expressions in the patient sample.

BACKGROUND

1. Apriori Algorithm

This algorithm is used to extract item sets that are frequently present in a dataset which is further used to determine association rule to culminate important trends in a dataset.

The main aim of using Apriori algorithm to generate rule generation is that there are a large number of combinations possible in a set of transactions. It is inconvenient to generate such large number of transactions. In order to prune useful transactions out of all the possible transactions which contribute to decision making, Apriori Algorithm is used.

The Apriori algorithm takes advantage of the fact that any subset of a frequent itemset is also a frequent itemset. The algorithm can therefore, reduce the number of candidates being considered by only exploring the itemsets whose support count is greater than the minimum support count. All infrequent itemsets can be pruned if it has an infrequent subset.

Apriori Algorithm initiates with counting frequently occurring transactions containing one item initially, then number of items are increased at each iteration until no more combination of items frequently occur in the dataset. In General, length (k+1) item sets are generated from length k frequent item-sets.

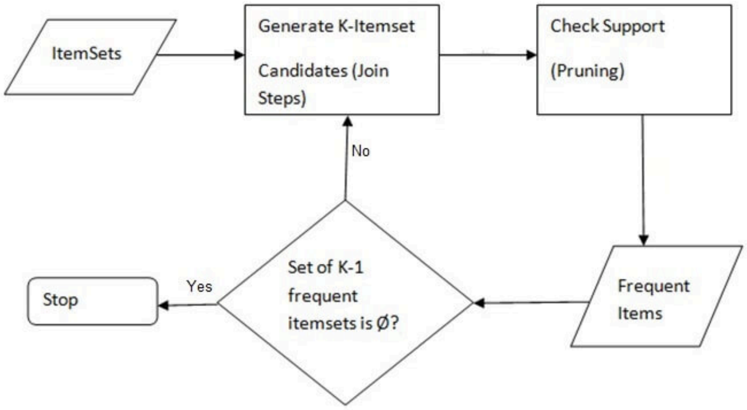

Figure 1. Apriori Algorithm

2. Association Rule

Association Rule is used to find rules in a given set of transactions, that will predict the occurrence of an item based on the occurrences of other items in the transaction

It find all item-sets that have ***support*** greater than a minimum threshold support and then use the large item-sets to generate the desired rules that have ***confidence*** greater than the minimum threshold confidence.

A rule consist of a head and body, represented in the form of X -> Y where X, Y are item sets present in a list of transactions. There could be any number of elements in the head and body. Support is defined as the number of transactions in the item set that contains both X and Y item sets to the total number of transactions in a list. Confidence is defined as the set of transactions that contain both X and Y item-sets to the set of transactions that contain X item-sets.

To perform Association Rule Mining using Apriori Algorithm, frequent item sets are first generated which are pruned by a support threshold which are further used for rule generation. Rules that have confidence greater than the minimum confidence threshold are kept and rest of the rules are eliminated.

These rules are used to predict occurrence of an outcome based on the occurrences of other items in the transaction.

Frequent Item Sets

Support set to 40%

```
Number of length-1 frequent item-sets for 40% : 167
Number of length-2 frequent item-sets for 40% : 753
Number of length-3 frequent item-sets for 40% : 149
Number of length-4 frequent item-sets for 40% : 7
Number of length-5 frequent item-sets for 40% : 1
Number of length-6 frequent item-sets for 40% : 0
```

Support set to 50%

```
Number of length-1 frequent item-sets for 50% : 109
Number of length-2 frequent item-sets for 50% : 63
Number of length-3 frequent item-sets for 50% : 2
Number of length-4 frequent item-sets for 50% : 0
```

Support set to 60%

```
Number of length-1 frequent item-sets for 60% : 34
Number of length-2 frequent item-sets for 60% : 2
Number of length-3 frequent item-sets for 60% : 0
```

Support set to 70%

```
Number of length-1 frequent item-sets for 70% : 7
Number of length-2 frequent item-sets for 70% : 0
```

Template Queries

Number of rules generated by keeping support as 50% and confidence 70%

Template 1

```
asso_rule.template1("RULE", "ANY", [‘G59_UP’]) : 26
asso_rule.template1("RULE", "NONE", [‘G59_UP’]) : 91
asso_rule.template1("RULE", 1, ['G59_UP', ‘G10_Down’]) : 39
asso_rule.template1("HEAD", "ANY", [‘G59_UP’]) : 9
asso_rule.template1("HEAD", "NONE", [‘G59_UP’]) : 108
asso_rule.template1("HEAD", 1, ['G59_UP', ‘G10_Down’]) : 17
asso_rule.template1("BODY", "ANY", [‘G59_UP’]) : 17
asso_rule.template1("BODY", "NONE", [‘G59_UP’]) : 100
asso_rule.template1("BODY", 1, ['G59_UP', ‘G10_Down’]) : 24
```

Template 2

```
asso_rule.template2("RULE", 3) : 9
asso_rule.template2("HEAD", 2) : 6
asso_rule.template2("BODY", 1) : 117
```

Template 3

```
asso_rule.template3("1or1", "HEAD", "ANY", ['G10_Down'], "BODY",
1, [‘G59_UP’]) : 24
asso_rule.template3("1and1", "HEAD", "ANY", ['G10_Down'],
"BODY", 1, [‘G59_UP’]) : 1
asso_rule.template3("1or2", "HEAD", "ANY", ['G10_Down'], "BODY",
2) : 11
asso_rule.template3("1and2", "HEAD", "ANY", ['G10_Down'],
"BODY", 2) : 0
asso_rule.template3("2or2", "HEAD", 1, "BODY", 2) : 117
asso_rule.template3(“2and2", "HEAD", 1, "BODY", 2) : 3
```