CSE-601 Data Mining and Bioinformatics

Project 1 Report

Association Analysis

Junsong Huang (50292448)
Yi Jin (50295580)
Yin Gong (50290271)

September 26, 2019

1. Abstract

The dataset of this project concerned with association analysis is about gene expression. By implementing apriori algorithm to find all frequent itemsets, we can efficiently get the itemsets which fit the specific requirement. And also, we can generate association rules according to the given templates.

2. Apriori Algorithm

The Association Rule Mining includes 2 parts.

For the 1st part, we are expected to generate all of the frequent itemset in the in the dataset using the given support rate. Using the brute algorithm is time consuming so we use the apriori algorithm to generate the frequent set.

The steps of apriori algorithm is listed below:

(1) we find the 1-length frequent item set and then we can combine those 1-length items and generate some 2-length frequent item candidate. Then, we can calculate the support of every 2-length frequent itemset, and save the 2-length frequent itemset which can satisfy the minimum support threshold.

(2) Then, we can generate all the 3-length itemset candidate and sift them by the support threshold. Later we can generate the 4-length itemset candidate using 3-length frequent set...
(3) We continue this step until there is no any longer itemset.

For the 2nd part, we will generate rules from every frequent item set. We can also use the Apriori Algorithm. For a given frequent itemset, we can first extract the Body of the candidate rule from length-1. We can get the Head using the itemset minus the Body. Then, we can calculate the confidence of the first level rule and sift them. Then we can prune some candidate rule based on the anti-monotone property. We can generate the length-2 Body using the remained length-1 Body, and calculate the confidence to sift them. Then we can generate the 3-length Body…. Until there are no potential rules. Using this method, we can the rules in one frequent itemset. We can traverse all of the itemset to get all of the rules.

3. Results

3.1 Part1

Requirement: Support = 30%, Confidence = 30%

number of length-1 frequent itemsets: 196
number of length-2 frequent itemsets: 5340

number of length-3 frequent itemsets: 5287
number of length-4 frequent itemsets: 1518
number of length-5 frequent itemsets: 438
number of length-6 frequent itemsets: 88
number of length-7 frequent itemsets: 11
number of length-8 frequent itemsets: 1
Total frequent itemsets: 12879

Requirement: Support = 40%, Confidence = 40%
number of length-1 frequent itemsets: 167
number of length-2 frequent itemsets: 753
number of length-3 frequent itemsets: 149
number of length-4 frequent itemsets: 7
number of length-5 frequent itemsets: 1
Total frequent itemsets: 1077

Requirement: Support = 60%, Confidence = 60%
number of length-1 frequent itemsets: 34
number of length-2 frequent itemsets: 2
Total frequent itemsets: 36

Requirement: Support = 70%, Confidence = 70%
number of length-1 frequent itemsets: 7
Total frequent itemsets: 7

Requirement: Support = 50%, Confidence = 70%
Number of length-1 frequent itemsets: 109
Number of length-2 frequent itemsets: 63
Number of length-3 frequent itemsets: 2
Total frequent itemsets: 174

3.2 Part2

3.2.1 template1

Query : asso_rule.template1("RULE", "ANY", ['G59_Up'])
Result: 26 rows selected.

Query: asso_rule.template1("RULE", "NONE", ['G59_Up'])
Result: 91 rows selected.

Query: asso_rule.template1("RULE", 1, ['G59_Up', 'G10_Down'])
Result: 39 rows selected.

Query: asso_rule.template1("HEAD", "ANY", ['G59_Up'])
Result: 9 rows selected

Query: asso_rule.template1("HEAD", "NONE", ['G59_Up'])
Result: 108 rows selected

Query: asso_rule.template1("HEAD", 1, ['G59_Up', 'G10_Down'])
Result: 17 rows selected

Query: asso_rule.template1("BODY", "ANY", ['G59_Up'])
Result: 17 rows selected.

Query: asso_rule.template1("BODY", "NONE", ['G59_Up'])
Result: 100 rows selected

Query: asso_rule.template1("BODY", 1, ['G59_Up', 'G10_Down'])
Result: 24 rows selected

3.2.2 template2
Query: asso_rule.template2("RULE", 3)
Result: 9 rows selected

Query: asso_rule.template2("HEAD", 2)
Result: 6 rows selected

Query: asso_rule.template2("BODY", 1)
Result: 117 rows selected

3.2.3 template3
Query:asso_rule.template3("1or1", "HEAD", "ANY", ['G10_Down'], "BODY", 1, ['G59_Up'])
Result: 24 rows selected

Query:asso_rule.template3("1and1", "HEAD", "ANY", ['G10_Down'], "BODY", 1, ['G59_Up'])
Result: 1 rows selected

Query: asso_rule.template3("1or2", "HEAD", "ANY", ['G10_Down'], "BODY", 2)
Result: 11 rows selected

Query: asso_rule.template3("1and2", "HEAD", "ANY", ['G10_Down'], "BODY", 2)
Result: 0 rows selected

Query: asso_rule.template3("2or2", "HEAD", 1, "BODY", 2)
Result: 117 rows selected

Query: asso_rule.template3("2and2", "HEAD", 1, "BODY", 2)
Result: 3 rows selected