# Data Mining and Bioinformatics

## Project 1 Report: Dimensionality Reduction and Association Rule Mining


Team: 23

Arun Sharma (arunshar)

Prashanth Chandrashekar (pc74)

Vipin Kumar (vkumar25)

# Association Rule Mining

## Apriori Algorithm

It's an influential algorithm for mining frequent item-sets for Boolean association rules. Here is the algorithm

1. Start with frequent itemset of length 1 and K = 1
2. Generate itemset of length 1
3. For every frequent itemset from length k frequent items generate length k + 1 candidate itemset
4. Prune candidate itemset containing subset of length k which are infrequent
5. Count each supporting candidate.
6. Remove candidates that are infrequent
7. Repeat until no new frequent itemset are there

## Association Rules generation algorithm

Input:

D ← Database of transaction
L ← Items
L ← Large Itemset
S ← Support
α← Confidence

Output:

R ← Association Rule satisfying s and α

## Algorithm:

```
R ← null
for each l belong to L do:
        for each x as a subset of l such that x not equal to null and not equal to l do:
                if support(l)/support(x) >= α then
                        R = R U {x => (l-x)};
```

For Part 1 of required tasks for Apriori are the list of results according to their support value are the following:

*Support is set to be 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*
*Number of all length frequent itemsets: 12879*

*Support is set to be 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*
*Number of all length frequent itemsets: 1077*

*Support is set to be 50%*
*Number of length-1 frequent itemsets: 109*
*Number of length-2 frequent itemsets: 63*
*Number of length-3 frequent itemsets: 2*
*Number of all length frequent itemsets: 174*

*Support is set to be 60%*
*Number of length-1 frequent itemset: 34*
*Number of length-2 frequent itemsets: 2*
*Number of all length frequent itemsets: 36*

*Support is set to be 70%*
*Number of length-1 frequent itemsets: 7*
*Number of all length frequent itemsets: 7*

For Part 2 of required task for Apriori, we set Support=50%, Confidence=70%. The following queries are:

For template 1, we have 9 possible keywords combinations:

*(result11,26) = asso_rule.template1("T1","RULE","ANY","['G59_Up']")*
*(result12,91) = asso_rule.template1("T1","RULE","NONE","['G59_Up']")*
*(result13,39) = asso_rule.template1("T1","RULE","1","['G59_Up','G10_Down']")*
*(result14,9) = asso_rule.template1("T1","BODY","ANY","['G59_Up']")*
*(result15,108) = asso_rule.template1("T1","BODY","NONE","['G59_Up']")*
*(result16,17) = asso_rule.template1("T1","BODY,1","['G59_Up','G10_Down']")*
*(result17,17) = asso_rule.template1("T1","HEAD","ANY","['G59_Up']")*
*(result18,100) = asso_rule.template1("T1","HEAD","NONE","['G59_Up']")*
*(result19,24) = asso_rule.template1("T1","HEAD","1","['G59_Up','G10_Down']")*

For template 2, we have 3 keywords choices:

*(result21,9) = asso_rule.template2("T2","RULE","3")*
*(result21,6) = asso_rule.template2("T2","BODY","2")*
*(result21,117) = asso_rule.template2("T2","HEAD","1")*

For template 3, AND/OR logical operator are used to connect 2 parts i.e. either template 1 or template 2.

*(result31,24) =*
*asso_rule.template3("T3","1or1","BODY","ANY","['G10_Down']","HEAD,1","['G59_Up']")*
*(result32,1) =*
*asso_rule.template3("T3","1and1","BODY","ANY","['G10_Down']","HEAD","1","['G59_Up']")*
*(result33,11) = asso_rule.template3("T3","1or2","BODY","ANY","['G10_Down']","HEAD","2")*
*(result34,0) = asso_rule.template3("T3","1and2","BODY","ANY","['G10_Down']","HEAD","2")*
*(result35,117) = asso_rule.template3("T3","2or2","BODY","1","HEAD","2")*
*(result36,3) = asso_rule.template3("T3","2and2","BODY","1","HEAD","2")*