# CS4038 – Data Mining (Spring 2024)

*Ayesha Liaqat*

## Assignment 3

| |
|---|
| Topics Covered: Clustering and Association rule mining, Apriori algorithm<br><br>Individual Assignment |
| Submission Deadline: **On Google Classroom**<br><br>*Only hand-written solutions will be accepted.* |

**Problem # 1:** Consider the data set shown in the table below.

| 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 | $\{a, b, e\}$ |

a. Compute the support for itemsets {e}, {b,d} and {b,d,e} by treating each transaction ID as a market basket.
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?
c. Repeat part (a) by treating each customer ID as a market basket. Each item should be treated as a binary variable (1 if an item appears in at least one transaction bought by the customer, and 0 otherwise.)
d. Use the results in part (c) to compute the confidence for the association rules {b, d} → {e} and {e} → {b, d}.

**Problem # 2:** 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 the table below with *minsup* = 30%, i.e., any itemset occurring in less than 3 transactions is considered to be infrequent.

| 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 the above table. Label each node in the lattice with the following letter(s):
   i. **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.
   ii. **F**: If the candidate itemset is found to be frequent by the Apriori algorithm.
   iii. **I**: If the candidate itemset is found to be infrequent after support counting.
   iv. **M**: If the node in the lattice is a maximal frequent itemset.
   v. **C**: If the node in the lattice is a closed frequent itemset.
b. What is the ***percentage of frequent itemsets*** (with respect to all itemsets in the lattice)?
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.)
d. What is the ***false alarm rate*** (i.e, percentage of candidate itemsets that are found to be infrequent after performing support counting)?

**Problem # 3:** Suppose that the data mining task is to cluster points (with (x,y) representing location) into three clusters, where the points are:

A1(12,10), A2(2,5), A3(2,6), B1(4,8), B2(7,5), B3(6,8), C1(1,2), C2(4,9)

The distance function is Euclidean distance. Suppose initially we assign A1, B1, and C1 as the center of each cluster, respectively. Use the **k-means algorithm** to show only:

a. The three cluster centers after the first iteration.
b. The final three clusters.

**Problem # 4:** Use the data points given in Problem # 3, and perform the **hierarchical clustering** using the following linkage criterions. Also draw their respective dendrograms and cut each of them at a suitable height to find the best number of clusters.

a. Complete linkage
b. Single linkage
c. Center-based
d. Group average