Lectures 1 & 2: Frequent Patterns and Association Rules Mining

Market-Basket model

Support

Support is an indication of how frequently itemset X appears in the dataset.

$$sup(X) = \frac{occurences\ of\ X}{N}$$

We often define an application specific threshold of minimum support which helps us distinguish frequent itemsets from infrequent ones.

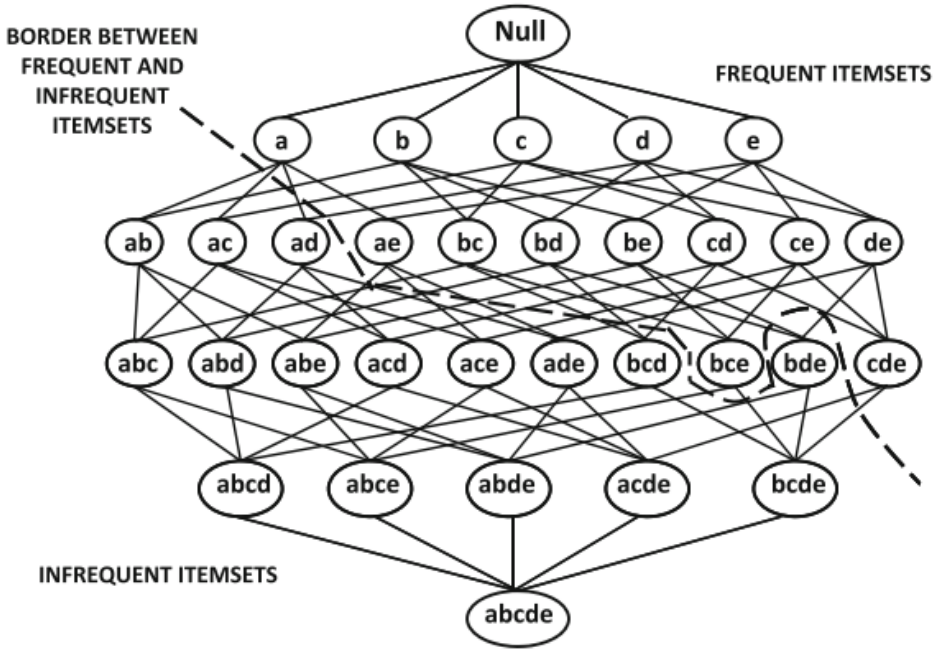
📌 **Support Monotonicity Property:** When an itemset X is contained in a transaction, all its subsets will also be contained in the transaction. Therefore, the support of any subset J of X will always be at least equal to that of X. This can be summarised as $sup(J) \geq sup(X)\ \ \forall\ J \subseteq X$.

📌 **Downward Closure Property:** Every subset of a frequent itemset is also frequent.
This constraint is often leveraged by frequent pattern mining algorithms to prune the search process and achieve greater efficiency. Furthermore, the downward closure

property can be used to create concise representations of frequent patterns, wherein only the maximal frequent subsets are retained.

📌 **Maximal Frequent Itemsets:** A frequent itemset is maximal at a given minimum support level $minsup$ if it is frequent and no superset of it is frequent.
All frequent itemsets can be derived from the maximal patterns by enumerating the subsets of the maximal frequent patterns. Therefore, the maximal patterns can be considered condensed representations of the frequent patterns. However, this condensed representation does not retain information about the support values of the subsets.

An interesting property of itemsets is that they can be conceptually arranged in the form of a **lattice of itemsets**. This lattice contains one node for each of the $2^{|U|}$ sets drawn from the universe of items U. An edge exists between a pair of nodes, if the corresponding sets differ by exactly one item. The lattice represents the search space of frequent patterns. All frequent pattern mining algorithms, implicitly or explicitly, traverse this search space to determine the frequent patterns.



The lattice is separated into frequent and infrequent itemsets by a border, which is illustrated by a dashed line in this image. All itemsets above this border are frequent, whereas those below the border are infrequent. Note that all maximal frequent itemsets are adjacent to this border of itemsets. Furthermore, any valid border

representing a true division between frequent and infrequent itemsets will always respect the downward closure property

Confidence

Confidence is an indication of how often the rule has been found to be true.

$$Confidence(A \rightarrow B) = \frac{sup(A, B)}{sup(A)}$$

As with support, we often define an application specific threshold of minimum confidence. An **association rule** is a rule that has at least a minimum level of support and a minimum level of confidence.

$\rightarrow$ The support threshold ensures that a sufficient number of transactions are relevant to the rule; therefore, it has the required critical mass for it to be considered relevant to the application at hand.

$\rightarrow$ The confidence threshold ensures that the rule has sufficient strength in terms of conditional probabilities.

📌 **Confidence Monotonicity Property:** Let $X_1 = \{bread\},\ X_2 = \{bread, butter\}$ and $X = \{bread, butter, beer\}$ three itemsets. Obviously $X_1 \subset X_2 \subset X$. Then the confidence of $\{bread, butter\} \rightarrow \{beer\}$ is at least the same as $\{bread\} \rightarrow \{butter, beer\}$. This results from the monotonicity property of $sup(x)$ since $sup(\{bread, butter\}) \leq sup(\{bread\})$. Because of confidence monotonicity, the rule $\{bread\} \rightarrow \{butter, beer\}$ is redundant in comparison to $\{bread, butter\} \rightarrow \{beer\}$ and can be omitted.

Lift

High confidence rules can sometimes be misleading since confidence ignores the support of the itemset appearing the the rule consequent. One way to address this is by using lift:

$$Lift(A \rightarrow B) = \frac{sup(A, B)}{sup(A) * sup(B)} = \frac{conf(A \rightarrow B)}{sup(B)}$$

$\rightarrow$ If the rule had a lift of $= 1$, it would imply that the probability of occurrence of the antecedent and that of the consequent are independent of each other. When two events are independent of each other, no rule can be drawn involving those two events.
$\rightarrow$ If the lift is > 1, that lets us know the degree to which those two occurrences are dependent on one another, and makes those rules potentially useful for predicting

the consequent in future data sets.
$\rightarrow$ If the lift is < 1, that lets us know the items are substitute to each other. This means that presence of one item has negative effect on presence of other item and vice versa.

Example

$$Support = \frac{frq(X,Y)}{N}$$

$$Rule: X \Rightarrow Y \rightarrow Confidence = \frac{frq(X,Y)}{frq(X)}$$

$$Lift = \frac{Support}{Supp(X) \times Supp(Y)}$$

Example:

Rule	Support	Confidence	Lift
$A \Rightarrow D$	2/5	2/3	10/9
$C \Rightarrow A$	2/5	2/4	5/6
$A \Rightarrow C$	2/5	2/3	5/6
$B \& C \Rightarrow D$	1/5	1/3	5/9

Frequent Itemset Mining Algorithms

The A-priori algorithm

$\rightarrow$ Algorithm with candidate generation

https://www.geeksforgeeks.org/apriori-algorithm/

The FP-growth algorithm

$\rightarrow$ Algorithm without candidate generation

https://www.geeksforgeeks.org/trie-insert-and-search/

https://www.geeksforgeeks.org/ml-frequent-pattern-growth-algorithm/

The PCY algorithm

https://llimllib.github.io/bloomfilter-tutorial/

https://medium.com/weekly-data-science/the-pcy-algorithm-and-its-friends-ecba67216190

The Multistage algorithm

https://medium.com/weekly-data-science/the-pcy-algorithm-and-its-friends-ecba67216190

Toivonen's Algorithm

https://www.geeksforgeeks.org/toivonens-algorithm-in-data-analytics/