Module – Summary

Association Rule Mining

Have you ever wondered which algorithm powers the "Frequently bought together" list on Amazon/Flipkart, or how you end up buying some items in a general store just because you happen to locate them "at the right place at the right time"? The algorithm that leads to such insights is the Association Rule Mining.

Association rule mining is an algorithm which is meant to find frequent patterns, correlations, associations, or causal structures from data sets.

Given a set of transactions, association rule mining aims to find the rules which enable us to predict the occurrence of a specific item based on the occurrences of the other items in the transaction. Let's learn more about this algorithm.

The main applications of association rule mining:

- **Basket data analysis** - is to analyse the association of purchased items in a single basket or single purchase.
- **Cross marketing** - is to work with other businesses that complement your own, not competitors. For example, vehicle dealerships and manufacturers have cross marketing campaigns with oil and gas companies for obvious reasons.
- **Catalogue design** - the selection of items in a business' catalogue are often designed to complement each other so that buying one item will lead to buying of another. So, these items are often complements or very related.

However, these association rules are only indicative of the existing data pattern and may not be because of the causal relationship between the items having the association.

Measure of Support, Confidence & Lift

To carry out association rule mining, you'll first need a data set of transactions. Each transaction represents a group of items or products that have been bought together and often referred to as an "item set". For example, one item set might be: {bread, cereals, milk, fruits, butter} in which case these items have been bought in a single transaction.

In an MBA, the transactions are analysed to identify rules of association. For example, one rule could be: {bread, butter} => {milk}. This means that if a customer has a transaction that contains bread and butter, then they are likely to be interested in also buying milk.

Before acting on a rule, a retailer needs to know whether there is sufficient evidence to suggest that it will result in a beneficial outcome. We therefore measure the strength of a rule by calculating the following three metrics:

Support: The percentage of transactions that contain all the items in an item set. The higher the support the more frequently the item set occurs. Rules with a high support are preferred since they are likely to be applicable to many future transactions.

Confidence: The probability that a transaction that contains the items on the left-hand side of the rule (antecedent) also contains the item on the right-hand side (consequent). The higher the confidence, the greater the likelihood that the item on the right-hand side will be purchased or, in other words, the greater the return rate you can expect for a given rule.

Lift: The probability of all the items in a rule occurring together (otherwise known as the support) divided by the product of the probabilities of the items on the left and right hand side occurring as if there was no association between them. A lift of more than 1 suggests that the presence of antecedent increases the probability that consequent will also occur in the transaction. Overall, lift summarises the strength of association between the products on the left and right hand side of the rule; the larger the lift the greater the link between the two products.

Apriori Algorithm

The A*priori principle* can reduce the number of item sets we need to examine. Put simply, the Apriori principle states that if an item set is infrequent, then all its subsets must also be infrequent. This means that if {bread} was found to be infrequent, we can expect {bread, cereals} to be equally or even more infrequent. So, in consolidating the list of popular item sets, we need not consider {bread, cereals}, nor any other item set configuration that contains beer.

Using the Apriori principle, the number of item sets that must be examined can be pruned, and the list of popular item sets can be obtained in these steps:

- **Step 0**. Start with item sets containing just a single item, such as {bread} and {milk}.
- **Step 1**. Determine the support for item sets. Keep the item sets that meet your minimum support threshold, and remove item sets that do not.
- **Step 2**. Using the item sets you have kept from Step 1, generate all the possible item set configurations.
- **Step 3**. Repeat Steps 1 & 2 until there are no more new item sets.

So, once you figure out the frequent item sets of size k-1, there are 2 ways to generate the frequent item sets of size k. These are the $F_{k-1}XF_1$ method and the $F_{k-1}XF_{k-1}$ method.

Any such algorithm that we may use, should satisfy 4 basic properties. The list of item sets should be:

- Complete
- Frugal
- Parsimonious
- Non-repeating

Maximal & Closed frequent item sets

What happens when you have a large market basket data with over a hundred items? The number of frequent item sets grow exponentially and this in turn creates an issue with storage and it is for this purpose that alternative representations have been derived which reduce the initial set but can be used to generate all other frequent item sets. These are the Maximal and Closed Frequent Item sets.

Maximal frequent item set is a frequent item set for which none of its immediate supersets are frequent. A closed frequent item set is a frequent item set that is both closed and its support is greater than or equal to minimum support threshold level. An item set is closed in a data set if there exists no superset that has the same support count as this original item set.

In conclusion, it is important to point out the relationship between frequent item sets, closed frequent item sets and maximal frequent item sets. As mentioned earlier closed and maximal frequent item sets are subsets of frequent item sets but maximal frequent item sets are a more compact representation because it is a subset of closed frequent item sets.

Closed frequent item sets are more widely used than maximal frequent item set because when efficiency is more important that space, they provide us with the support of the subsets so no additional pass is needed to find this information.