file/after/tracklang.stypackage/after/bidipackage/after/luabidi

Mining Frequent Patterns, Associations, and Correlations

When you have a set of transactions in a dataset, some of them can happen frequently. For a dataset D, there are many combinations of the items, so how do we find the ones that happen frequently?

A **frequent itemset** is one whose **support** and **confidence** are above a certain threshold. The support for a rule $A \implies B)$ is the percentage of all transactions that contain $A \cup B$, or their intersection. The confidence for a frequent itemset is the percentage of all transactions containing A that also contain B, i.e. $P(B|A)$.

A **closed itemset** is one where for all the itemsets X there is no proper super-itemset Y(X ⊠Y) such that X and Y both have the same support count.

Apriori Algorithm

This algorithm uses knowledge of previous frequent itemsets to calculate the current one, e.g. it uses L_1 to calculate L_2, the set of frequent 2-itemsets. The idea is that we first find all the 1-itemsets and use that to find the 2-itemsets. Since all the 1-itemsets are proper subsets of 2-itemsets, we could build up our L_k this way. However, for every level of itemsets we need to scan the entire database :(

The central idea of the algorithm is that an itemset I is not frequent, then all of its supersets will not be frequent either. $P(I \cup A)$ cannot be more frequent than $P(a)I$ when we add an item A.

The first step in the algorithm is the join step. For two subsets of L_{k-1}, l_1 and l_2, we join them if their first $k-2$ elements are equal, and $l_1[k-1] < l_2[k-1]$. This way we produce a subset that is still lexicographically ordered and contains one more element in it than before.

The second step is the prune step. We generated a candidate set C_k, and maybe not every itemset here will be frequent. If any $k-1$ itemset in C_k is not in L_{k-1}, then we know it can't be frequent thus it can be removed from C_k. (Here is where we can use a **hash tree** for quick searching of frequent itemsets)