Dear Toviah and Joshua,

I started looking into the first test case that was produced by your coding.

I am quite delighted by the progress so far and my initial look into the output did not produce any red flags.

I continue communicating with Bertet, and I asked her to possibly produce the canonical unit basis for the table, by her available means, to make a comparison.

At this point I am looking at further possibilities to reduce the output, and deal with issues of the minimum support that defines the approach in data mining.

We discussed with you in the beginning that some legitimate implications defining the Galois lattice may have little content from the point of dependencies among attributes. This happens, in particular, when the left side of implication X->Y has a void support. In other words, there is no single row which wold have all attributes from X present.

From lattice point of view, one would obtain the set of join irreducibles that join at top element of lattice. One would expect then that many conclusions would follow from the same X, while carrying no essential information.

In our example, set of columns {2, 18} have empty support. You can check that 2 18 -> {1,3,4,5,6,9,12,16} holds in Galois lattice (all implications from unit expansion are in output list).

Using the binary part, one observes that the only refinement of this set is {7,18}, and 7 18 -> {8,10,11,13,14,15,17}.

I verified that those implications are present in the output as well. This is a good indication that the output is correct. The other observation, of course, is that all 8 implications from 2 18 -> {1,3,4,5,6,9,12,16} can be removed from the list if we are about to keep only meaningful ones, for the purpose of dependencies between the columns.

You may include a new subroutine that identifies the implications X->y with low sup(X). For example, considering possible future applications in data mining, we may consider a new parameter MinSup = d. You would remove all implications that have Sup(X) <= d. Evidently, one need to run this subroutine at least for d=0. But then we may decide to leave only implications for which X holds at least in 2 rows, thus, setting d to 1.

My other check was how many implication may be removed due to the fact that S-basis may have extra implications compared to the D-basis. For this, within each group with the same conclusion, one need to compare every pair X1 and X2 of left sides, and check whether X1<< X2, based on the knowledge of binary implications. If this holds, then X2->y is removed from the list.

For example, I found by hand that 15 9 -> 7 and 15 3 -> 7 are both in the output, and we have 9-> 3 in the binary part. Hence, 15+3 is a minimal cover for 7, while 15 9 -> 7 may be removed from the list.

For experimenting purposes, it would be interesting to consider how many "extra" implications we obtained in this test case. My guess this should not be larger than 10% of all implications in S-basis.

Ok, these two observations may give you additional issues to improve the output, so if Uno's implementation does not cause many griefs, maybe, you could direct your talent toward those improvements.

Best wishes,

KA