Chapter 6

More expressive variants of exploration

Several authors have extended the basic method of attribute exploration, as it is presented in the two previous chapters of this book, to more advanced situations, and this last chapter is devoted to such ideas. We start with rule exploration, which lifts attribute exploration in a rather natural way to first-order predicate logic, with Horn formulas replacing implications. Next we discuss how attribute exploration can be integrated with Description Logics. However, so much substantial research has been contributed to this topic that we can only introduce some first ideas. The third section is devoted to concept exploration, which was already mentioned in the very first publication on Formal Concept Analysis. Three more generalizations are sketched in the fourth section, which then concludes by mentioning a few further ones.

In contrast to the previous chapters, we now cannot always illustrate the theory by detailed examples. One reason is that such “use cases” have not yet been worked out. Another one is that, due to the complexity of the methods, realistic examples would go beyond the scope of this book.

6.1 Rule exploration

The use of first-order logic Horn formulas instead of implications has been studied in detail and is actually rather simple. We utilize a brute force approach of “propositionalization” to represent rules by families of (propositional) implications. It is, as we believe, best introduced by means of an example or, better, of two. The first one (“triplets of squares”) extends the exploration discussed in Section 4.2.2. It is somewhat uninteresting in its result, but very transparent. A reader who has read this example carefully will probably understand rule exploration even without further definitions.

The second example (“treelike relations”) is a little more magical. It com-
Chapter 6. More expressive variants of exploration

bines basic rule exploration with using symmetries and background knowledge obtained from conceptual scaling. The outcome of this exploration is the “discovery” of a non-trivial mathematical theorem, though without a complete proof.

6.1.1 Triplets of squares

We build on one of our running examples, the classification of “pairs of squares”, where we asked how two squares can be positioned with respect to six given attributes such as “disjoint” and “having a common vertex”.

What about three squares? Here are four (out of many) possible configurations of triplets of squares:

You may guess that a classification of triplets of squares is much more complex than that of pairs. This is indeed the case. In order to fit the example in this book, we therefore restrict to four of the six attributes and only ask if the squares are disjoint or they overlap, have a common vertex or a common edge.

Moreover, we allow only examples where all squares under consideration are of the same size (“unit squares”) and have sides parallel to the axes of our coordinate system.

But wait! Being disjoint or having a common edge are properties of two squares, not of three. We need to specify in each case which two of the three squares are meant. Thus, a typical context row looks like this:

Instead of four attributes, we now have twelve. Actually, there could have been 36, including disjoint(B, A) and disjoint(A, A). But obviously all four properties are symmetric: A and B are disjoint iff B and A are, etc. And no square is disjoint to itself, but each square overlaps itself and shares vertices and edges with itself.

The above triplet of squares can be labeled in six different ways. But due to a symmetry, only three different context rows are obtained for this example: