FCP-Growth: Class Itemsets for Class Association Rules

Emna Bahri and Stephane Lallich
University of Lyon, Eric Laboratory
5, Avenue Pierre Mandes France, 69500 Bron, France
Emna.Bahri — Stephane.Lallich@univ-lyon2.fr

Associative classification

Since the first work of (Liu, Hsu, & Ma 1998), various works show the good performance of associative classification (association based classification) in terms of error rate reduction. Association classification deals with the prediction of the class from association rules, known as class association rules or predictive association rules. A class association rule is a rule whose consequent must be the indicating variable of the one of class attributes. Such a rule is thus $A \rightarrow c_i$, where A is a conjunction of Boolean descriptors and c_i is the indicating variable of the i_e class attribute. The interest of a class rule is to allow focusing on groups of individuals, possibly very small, homogeneous (same descriptors in antecedent) which present the same class. The association classification methods proceed in two phases, a phase of construction of class association rules (class association rule mining) followed by a phase of prediction from the class association rules obtained. The first phase is held caned out into two steps; the first step is to extract frequent itemsets, taking into account the threshold of support chosen. The second step is to construct class association rules satisfying the fixed threshold of confidence. Among these methods, one will quote CBA (Classification Based on Association) (Liu, Hsu, & Ma 1998), CPAR(Classification based on Predictive Association Rules) (Yin & Han 2003). and CMAR (Classification based on Multiple Association Rules) (Li, Han, & Pei 2001).
Certainly, these methods show that the association based classification gives better results. However, these methods are based on a first phase of finding frequent itemsets which is not in relation with the classification phase like Apriori and FP-Growth. This is explained by the fact that the methods of finding frequent itemsets do not depend on the class to predict. Moreover, it should be noted that the class association rules are confronted with the problem of the unbalanced classes, since in the first phase of generation of rules, various methods select the rules having a fixed support without taking into account the class distribution.

To avoid these problems, we aim to improve the association rule mining algorithm FP-Growth. We propose FCP-Growth which makes it possible to generate only the frequent itemsets containing one of the class itemsets and to use a support threshold which is related to the number of examples for each class.

State of the art

Contrary to Apriori (Agrawal & Srikant 1994) which generates itemsets candidates and tests them to preserve only frequent itemsets, FP-Growth (Han, Pei, & Yin 2000) builds the frequent itemsets without generation of candidates. In fact, it uses the strategy of divide-and-conquer. First, it compresses the frequent items found in the data base in a frequent-pattern tree (FP tree) which contains the association of the itemsets. Then, it divides each association which is presented by item frequent or pattern fragment. The FP-Growth transforms the problem of searching the frequent itemsets by the search of smaller itemset and the concatenation of the suffix (last frequent itemset). This makes it possible to reduce the cost of searching. In our study, we are interested in FP-Growth, because of its structuring (FP-tree). In fact, by using a tree structure to search and represent frequent itemsets, FP-Growth allows a better execution time.

FCP-Growth

The construction method of class rules which we propose, FCP-Growth, is based on several principles:

- Pretreatment of the database: During this stage, we give the possibility to the user to choose the class attribute from the various attributes of the database. Thus, the class to be predicted is determined, we reorganize the data base into a table whose first column contains the various values assigned to this class. This method, enables us to build frequent FCP-tree of the items which contain the class to be predicted.
- Choice of the adjustable supports: This stage enables us to define, not a fixed support, but an adaptive support for each class modality depending on the number of transactions n_i which validates each class modality *i*. Indeed, we choose σ and we define the support threshold relative to c_i as $\sigma_i = \sigma \times n_i$. This support threshold is adapted to the examples numbers of each class. This method supports the classes having an insufficient number of examples. We construct then the list L which contains firstly

the class(C1 or C2) and the items stored according to descending support count of C1 or C2.

- Construction of FCP-tree and the class frequent patterns: This stage proceeds in the same way as FP-Growth. However, only the transaction which starts with a class to be predicted will be treated. Thus there will be the class attributes as children linked to the root of the FCP-Tee. For example, we create first the root of the tree labeled with null, scan the database for the second time and construct m children of the root if we have m class attributes. In the example below, we construct 2 children to the root C1 and C2 because we have 2 class attributes C1,C2. The first branch of the tree is constructed during the first transaction.

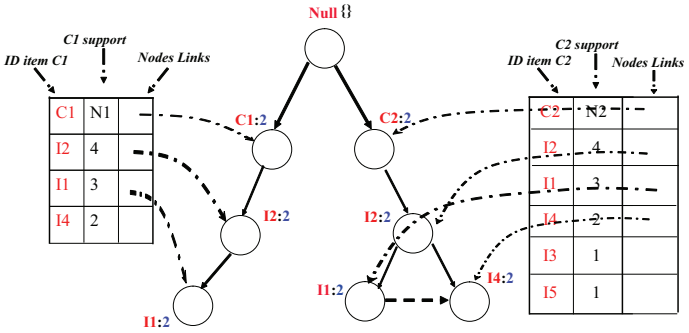

Figure 1: Construction of FCP-tree

- Mining the FP-tree and generation of frequent itemsets: The FCP-tree is mined in the same way as FP-tree. For each item, all conditional patterns are found and all frequent patterns (i.e., patterns having a higher support or equal to the selected support) are generated. This phase is called "mining frequent patterns". Frequent itemsets are obtained by the association of each item with frequent pattern. All frequent itemsets found contain a class item.
- Pruning: To ensure an effective classification, we must prune the rules (associations between itemsets) to reduce the redundance and noise. During this stage, we use the general rule for pruning. The general rule is the most specific rule having a low confidence. For example, let's take two rules R1 and R2. R1 is more general than R2 if
  - confidence(R1)>confidence(R2)
  - confidence(R1)=confidence(R2) and support(R1)>support(R2)
  - confidence(R1)=confidence(R2) and support(R1)=suppport(R2) and R1 have less items

Results

To test the effectiveness of FCP-Growth, we tested it on 5 very large real databases. We used different support thresholds to calculate the total number of itemsets, the itemsets' number of class C, the number of non relevant itemsets, the execution time and the cover rate. The results show that the use of FCP-Growth makes it possible to eliminate, between one third and half of the treated itemsets, due to non relevance, to extract more rules of classes than FP-Growth (about 30%), while preventing the minority class from being too underprivileged. To evaluate the performance of FCP-Growth, we retained the cover rate which indicates the proportion of examples which are covered by at least one class itemset, i.e. which checks the class rule corresponding to this itemset. The results show the superiority of FCP-Growth. Indeed, for each base, the cover rate is clearly increased to arrive approximately at 90% of coverage. Finally, we note that the execution time of FCP-Growth is always lower or equal to that of FP-Growth, and better since the support threshold is small.

Conclusion

During this work, we propose FCP-Growth, an adaptation of FP-Growth to extract class association rules. In fact, we introduce the choice of the class to predict and to construct the frequent itemsets which contain a class attributes. Moreover, we introduce an adaptive support which depends on the class distribution in order to improve quality of the rules by supporting the minority classes. The results show a best execution time thanks to the generation of itemsets limited to the class itemsets. In addition, results show a quality of the rules higher than that generated by FP-Growth based on cover rate criterion. Considering the found results, we plan to use this algorithm for mining class association rules. We still have to develop the phase of prediction. In fact, it was the goal of these improvements: to develop a system that generates higher quality rules for a better performance classification.

References

Agrawal, R., and Srikant, R. 1994. Fast Algorithms for Mining Association Rules. In Bocca, J. B.; Jarke, M.; and Zaniolo, C., eds., *Proc. 20th Int. Conf. Very Large Data Bases, VLDB*, 487–499. Morgan Kaufmann.

Han, J.; Pei, J.; and Yin, Y. 2000. Mining frequent patterns without candidate generation. In Chen, W.; Naughton, J.; and Bernstein, P. A., eds., *2000 ACM SIGMOD Intl. Conference on Management of Data*, 1–12. ACM Press.

Li, W.; Han, J.; and Pei, J. 2001. CMAR: Accurate and Efficient Classification Based on Multiple Class-Association Rules. In *ICDM*, 369–376.

Liu, B.; Hsu, W.; and Ma, Y. 1998. Integrating Classification and Association Rule Mining. In *Knowledge Discovery and Data Mining*, 80–86.

Yin, X., and Han, J. 2003. CPAR : Classification based on Predictive Association Rules. In *In 3rd SIAM International Conference on Data Mining (SDM'03)*.