# A Comparison of Techniques for Selecting and Combining Class Association Rules

Jan-Nikolas Sulzmann and Johannes Fürnkranz

Department of Computer Science, TU Darmstadt
Hochschulstr. 10, D-64289 Darmstadt, Germany
{sulzmann,juffi}@ke.informatik.tu-darmstadt.de

**Abstract.** Local pattern discovery, pattern set formation and global modeling may be viewed as three consecutive steps in a global modeling process. As each of these three steps have gained an increased attention in recent years, a great variety of techniques for each step have been proposed, but so far there has been no systematic comparison of the possible choices. In this paper, we survey and evaluate several options for selecting a subset of class association rules and for combining their predictions into a global rule model. Our results confirm that the commonly used Weighted Voting technique is, indeed, a good choice. We can also see that pattern set selection does not seem to have a large impact upon the performance of the rule ensemble.

## 1 Introduction

Classification association rule mining is the integration of the two key rule learning tasks: classification rule mining and association rule mining. Classification rule mining extracts a small set of classification rules from the database and uses them to build an accurate classifier. Most of the times the rules are generated one by one in a separate and conquer style exploiting the interaction with previous rules (Fürnkranz, 1999). On the other hand, in association rule mining all rules in the databases that satisfy some minimum interestingness constraints (typically minimum support and confidence) are generated exhaustively and without regard of their interaction with other rules. Additionally both methods differ in the rules they discover. Classification rules have a predetermined target the so called class, while association rules lack a predetermined target. The integration of these two mining techniques has been proposed by (Liu et al., 1998) and is done by concentrating on a specific subset of associations, namely class association rules (abbreviated CAR), which can be used for classification.

Classification association rule mining may be considered as the prototypical example for the recently proposed LeGo data mining framework for combining local patterns into a global model (Crémilleux et al., 2007). This model essentially consists of three phases: The *local pattern discovery* generates all class association rules satisfying predefined constraints (e.g. a minimum support, closeness etc.). The second phase, the *pattern set formation*, aims at selecting an optimal subset of the previously generated association rules, according to one or more

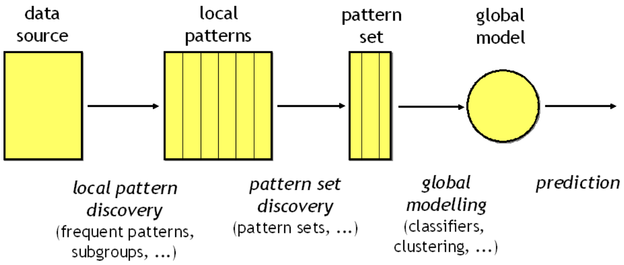

**Fig. 1.** The LeGo framework (Crémilleux et al., 2007)

arbitrary selectable heuristics that estimate the usefulness of the subset for future predictions. In most cases this task is accomplished by wrapper or filter approaches. Note that in the first step only association rules are evaluated independently of each other, while in the second step entire subsets are evaluated. In both phases unsupervised or supervised evaluation measures can be employed.

Local pattern discovery has gained an increased attention (Morik et al., 2005) in recent years, resulting in a great variety of techniques for the generation of frequent local patterns, or frequent class association rules (Agrawal and Srikant, 1994; Han et al., 2004). Each of these implementations yield essentially the same result, influenced by the defined external constraints (e.g. closeness (Zaki and Hsiao, 2002)). With this in mind we concentrate on the latter two steps, the pattern set discovery and the global modeling. The main goal of this paper is an empirical comparison of different techniques for these steps, which we will briefly recapitulate in the following sections. We will examine how respectively two representative of these perform in liaison with each other and compare these results with the performance of each technique if combined with a respective "neutral" technique for the other step (e.g., selecting all class association rules, or using all selected patterns for the prediction).

The paper is organized as follows: Section 2 and Section 3 give a short introduction into class association rule mining, global pattern discovery and the methods we apply. Section 4 describes the experimental setup and Section 5 presents the results. Our conclusions from these experiments are summarized in Section 6.

## 2 Class Association Rule Mining

Before we outline the principles of classification association rule mining, some notions have to be introduced. Using terms of both classification and association rule mining, we explain classification and association rule mining separately and show how these both techniques are fused for class association rule mining and which modifications have to be made.

In classification rule mining, a *data set* $D$ is a relation which is defined by a finite set of $m$ distinct *attributes* $A_1, \ldots, A_m$ and a set of class labels $C$, and consists of $n$ instances. Each attribute $A_i$ belongs to a certain category (for our purposes only nominal and numeric ones are feasible) and therefore has either a finite (a category) or infinite (a real number) set of possible values $a_i^j \in A_i$. Instances $d \in D$ are described by a set of *attribute values* (for each attribute) and in our case a single *label* ($d = (a_1^{j_1}, a_2^{j_2}, \ldots, a_m^{j_m}, c)$). Note that in some cases multiple labels can be allowed. The *body* (or *premise*) of a classification rule consists, similar to instances, of values for a set of mutually exclusive attributes, which form a condition that has to be met by the example. The *head* (or *conclusion*) of the rule is a single class value, which will be predicted for an example that satisfies the conditions of the body of the rule. Thus, a rule is an implication of a conjunction of conditions that imply a class value ($d = (A_k = a_1^{j_k} \wedge A_l = a_2^{j_2} \cdots \rightarrow c)$). A rule *covers* an example if the example meets the premise of the rule which is then called a *covering rule*. How classification rules are generated (e.g., separate-and-conquer-rule learning) and how the covering rules are used together for future predictions (e.g., decision lists) depends on the employed rule learning algorithm.

In association rule mining a data set is a set of transactions. Each *transaction* $t \in T$ contains a finite set of *items* $t \subseteq I$, where $I$ is the set of all items and $|t| \geq 1$, and has a unique *transaction identifier tid* $\in T$, where $T$ is the set of all *tids*. A set $X \subseteq I$ is called an *itemset* and a set $Y \subseteq T$ is called *tidset*. If an itemset contains exactly $k$ items it is called *k-itemset*. For an itemset $X$, its corresponding tidset is denoted as $t(X)$, the set of all tids that contain $X$ as a subset. Analogous for a tidset $Y$, we denote its corresponding itemsets $i(Y)$, the set items common to all the tids in $Y$. Note that for an itemset $X$ holds $t(X) = \cap_{x \in X} t(x)$, and for a transaction set $Y$ holds $i(Y) = \cap_{x \in X} t(x)$. The combination of an itemset $X$ and its tidset is called *IT-Pair* and is denoted by $X \times t(X)$. An itemset $X$ is *closed* if $i(t(X)) = X$ holds. In other words an itemset is closed if no item can be added to it without reducing the number of transactions.

An *association rule* $X \rightarrow Y$ consists of two itemsets $X$ and $Y$. $X$ forms the body of the rule and $Y$ is the head of the rule. An association rule $r$ has a *support* $s(r) = s$ in $D$, if $s$ percent of the cases in $D$ contain the head $X$ and body $Y$. A rule $X \rightarrow y$ holds in $D$ with *confidence* $c(r) = c$ if $c$ percent of the cases that contain the head $X$ also contain the head $Y$. Covering is here defined analogous to classification rule learning. If the support of an association rule exceeds a minimum support threshold the association rule is called *frequent*. Analogous for a minimum confidence threshold an association rule is called *confident*. If the disjunction $X \bigcup Y$ of the body and head of an association rule is closed the association rule is also called closed. Note that the set of all closed frequent itemsets can be used for the generation of all frequent itemsets and therefore of all frequent association rules.

Class association rule mining combines classification and association rule mining. A *class association rule* (abbreviated CAR) $r$ is an association rule of the form $X \rightarrow y$, where $X \subseteq I$ is an itemset, and $y \in Y$ a class label. A class association rule $r$ has a *support* $s(r) = s$ in $D$, if $s$ percent of the cases in $D$

contain $X$ and are labeled with class $y$. A rule $X \rightarrow y$ holds in $D$ with *confidence* $c(r) = c$ if $c$ percent of the cases that contain $X$ are labeled with class $y$.

Classification association rule mining basically consists of three steps. The first step employs an association rule mining algorithm that generates frequent itemsets. At best the algorithm generates only frequent itemsets which can be used to generate class association rule. If this is not the case an additional filtering of inappropriate rules has to be applied before one can proceed to the next step. Obviously classification data sets are not always viable for association rule mining but can be transformed into a association data set. For example, numeric attributes can be discretized (e.g., (Fayyad and Irani, 1993)) in advance.

The second step selects the class association rules which exceed a determined threshold for one or more given heuristic values. These heuristics can be divided into two groups. The first group considers only the properties of the rule alone without regard of other rules (e.g. confidence). The second group evaluates the usefulness of the rule in interaction with other rules (e.g., cross entropy). In some cases the selected rules are sorted descending according to one or more heuristics. Note that the heuristics for the selection and sorting can differ.

In the last step the selected rules have to be applied for the classification of examples whose class is unknown. There are many different approaches on how this is done. One solution is the decision list. Here all rules are sorted as above mentioned and only the prediction of the first covering is used. Other approaches like the combination of the predictions of all covering rules will be described in the next section.

## 3 Survey of Different Options for Class Association Rule Discovery

The LeGo model for global modeling via local pattern discovery consists of three phases: the local pattern discovery, pattern set discovery and global modeling. These task are described separately in the following subsections considering closed frequent itemsets for the generation of class association rules explaining some representatives and the respective technique briefly. For further information about these steps and their attendant examples we refer to (Crémilleux et al., 2007)

### 3.1 Local Pattern Discovery

In this work, we concentrate on association rule mining (Goethals, 2005) which is both the most basic and popular representative of local pattern discovery. Restricting us to frequent itemset clearly leads to a biased result which misses some aspects of the distribution of items and some co-occurrences among the associations (e.g., infrequent, but meaningful rules), but for the purpose of comparison this will not have severe consequences.

Itemsets can be considered as local patterns because items describe only the instances of the database which are covered by the respective individual pattern. Typically frequent itemset discovery algorithms generate the pattern in a exhaustive, top-down and level-wise search. Most of the times the set of discovered itemsets is returned in a compressed, but complete and reconstructable representation by using elaborate data structures (Han et al., 2004) or by exploiting specific characteristics (Zaki and Hsiao, 2002).

For our experiments we chose the local pattern discovery algorithm CHARM (Zaki and Hsiao, 2002) which is an effective algorithm for the enumeration of closed frequent itemsets. Not going into detail, CHARM employs several innovative ideas which include using a novel tree-like search space capable of the simultaneous exploration of the itemset and the transaction space, utilizing a hybrid search that skips many levels in the tree structure and a hash-based closeness checking. For further details we refer to (Zaki and Hsiao, 2002) which provides a survey of this specific type of global pattern discovery.

Note that the first phase of class association rule discovery may also be considered as subgroup discovery (Wrobel, 1997; Klösgen, 2002).

### 3.2 Pattern Set Discovery

The local pattern discovery phase generates patterns which are chosen on the basis of their individual properties and performance. In practice, the resulting sets of local patterns are large and show potentially high levels of redundancy among the patterns. These two properties can be derogatory to various applications. A manual inspection of local patterns is only feasible for a small, manageable amount of patterns. Additionally high redundancy can hinder the performance of many, often redundant, features. Aiming to alleviate these issues the pattern set discovery tries to reduce the redundancy by selecting only a subset of patterns from the initial large pattern set.

Several approaches have been proposed to reduce the number of local patterns without regard of their future use. Recent examples include constraint-based pattern set mining (De Raedt and Zimmermann, 2007) and pattern teams (Knobbe and Ho, 2006a;b). Both approaches assume that the syntactic structure of the individual patterns is irrelevant at this stage, and that patterns can be fully characterized by a binary feature that determines for each example whether it is covered by the pattern or not. For further details on these or alternate approaches we refer to the just mentioned papers and to (Zaki and Hsiao, 2002)

For this work we will consider four representatives of pattern set discovery. The first and most simple one is not obviously a pattern set discovery, as it selects simply all previously generated patterns for the global modeling. Therefore this "all selector" can be considered as the neutral counterpart to the global modeling techniques. The second one is a confidence filter which selects all items or class association rules whose confidence exceeds a given minimum confidence threshold.

The other two pattern set discovery heuristics, *joint entropy* and *exclusive coverage*, are taken from (Knobbe and Ho, 2006b) considering the implementation suggestion in (Knobbe and Ho, 2006a). For both heuristics we employed a greedy wrapper selecting sequentially the patterns which yield the highest reward for the given heuristic until a predetermined size for the pattern set has been achieved. Note that both heuristics are unsupervised.

Joint entropy has been proposed for maximally informative $k$-itemsets (abbr. miki) (Knobbe and Ho, 2006a) but can also be applied to the pattern set discovery task. Essentially all patterns are treated as binary features so that the joint entropy for each pattern set is equal to the joint entropy of its features. The entropy measures the uniformity of the distribution of instances over different contingencies (by what patterns a instance is covered).

Exclusive coverage tries to reduce the amount of overlap between patterns. So pattern sets are favored if many instances are covered only by a single pattern. Essentially exclusive coverage counts the coverage that is exclusive for each pattern.

### 3.3 Global Modeling

There are many choices for global modeling algorithms that are based on pattern sets or sets of class association rules. Essentially, any induction algorithm can be used at this point. However, our primary focus are rule learning methods, so we confine ourselves to methods that consider the entire pattern set as a rule theory, and try to combine the predictions of the individual rules.

In particular, we consider two groups of techniques: The first group are *voting methods* which use the predictions of all rules that cover an example as votes for the final prediction. The vote of a single rule can be weighted based on its heuristic value or on the ranking of all rules according to their heuristic value. The second group are *probabilistic methods* which use estimated probabilities as the final prediction. In both cases, we estimate the value of a rule with the commonly used Laplace-corrected precision which will be discussed later.

**Voting Methods** Common ground of all voting methods is, as the name suggests, the interpretation of individual predictions as votes for the corresponding class. Different voting methods differ in the weights they assign to a vote of a rule. So, essentially, the classification works as follows:

$$\arg\max_{c_i \in C} \sum_{r \in R_{c_i}} weight(r),$$

where $R_{c_i}$ is the set of Rules covering the example and predicting class $c_i$ (e.g. $A_1 = a_1^j \wedge A_2 = a_2^k \cdots \rightarrow c_i$). The weight of the rule $weight(r)$ depends on the chosen voting method.

The first representative *Best Rule* (abbrev. $BR$) considers only the best rule which covers the example to predicted. At first sight $BR$ does not seem to be a voting method but it is possible to choose voting weights that simulate its behavior (by ordering the rules according to their quality and using exponentially decaying weights). Essentially, this method corresponds to using a decision list in which the rules are sorted according to their weight.

The next representatives are *Unweighted* and *Weighted Voting* (abbrev. $V$ and $WV$ accordingly). These methods have in common that they use the weights of all covering rules. $V$ assigns a weight of one to all covering rules, essentially this can be considered as the counting of covering rules separately for each class. $WV$ uses the Laplace value (which will described at the end of this section) of each rule as its weight, so basically the laplacian weights are counted for each class.

$$weight_V(r) = 1 \qquad\qquad weight_{WV}(r) = Laplace(r)$$

The last two methods *Linear Weighted Voting (LV)* and *Inverse Weighted Voting (IV)* (Mutter, 2004) differ from $V$ and $WV$ as they do not use the Laplace value $Laplace(r)$ but the ranking for the weighting of a rule $r$. So each rule $r$ obtains a rank $rank(r)$ according to the Laplace sorting. The ranks are represented by integers beginning with one for the best rule and ending with total number of rules for the worst ($rank_{max}$).

$$weight_{LV} = 1 - \frac{rank(r)}{rank_{max} + 1} \qquad\qquad weight_{IV} = \frac{1}{rank(r)}$$

**Bayesian Decoding** The *Bayesian Decoding* (abbrev. $BD$) is a probabilistic approach to estimate the class of an example on the basis of the rules by which it is covered. Contrary to the previous voting method a rule influences directly the outcome not only for the class it predicts but also for all classes of the data set.

The goal of this method is the estimation of the probability of a class $c_i$ under the observation of the conjunction of the rules $R = R_1 \wedge R_2 \wedge \cdots \wedge R_s$ that cover the example, namely $\Pr(c_i|R)$,and the prediction of the most probable class.

$$\arg\max_{c_i \in C} \Pr(c_i|R)$$

This probability can be translated in a determinable form by applying the Bayes theorem. This leads to the following formula:

$$\Pr(c_i|R) = \frac{\Pr(R|c_i)}{\Pr(R)}$$

As the denominator $\Pr(R_1 \wedge R_2 \wedge \cdots \wedge R_s)$ does not affect the relative order of the estimated probabilities it can be ignored. If we additionally assume that the observation of one of the Rules $R_j$ is (class-conditionally) independent of the occurrence of the other we can make the following naïve assumption:

$$\Pr(R|c_i) = \Pr(R_1 \wedge R_2 \wedge \cdots \wedge R_s|c_i) = \prod_{k=1}^{s} \Pr(R_k|c_i)$$

Finally the classification works as follows:

$$\arg\max_{c_i \in C} \Pr(c_i) \cdot \prod_{k=1}^{s} \Pr(R_k|c_i)$$

It remains to be explained how these probabilities can be estimated. $\Pr(c_i)$ can be estimated simply by counting the training examples belonging to class $c_i$ and dividing this number by the total number of training examples. $\Pr(R_k|c_i)$ can be estimated quite similarly, and simultaneously for all classes $c_i \in C$. First, we determine the number of training examples that are covered by the rule $R_k$ separately for each class and divide these numbers by their sum. It is possible that some rules do not cover examples of some classes, leading to a probability of zero for these classes as a single zero probability will yield to a product of zero. To avoid this problem, we apply the Laplace correction to the estimated probabilities:

$$Laplace(r) = \frac{p_i + 1}{\sum p_i + |C|}$$

where $p_i$ is the number of covered examples that are of class $c_i$. Essentially, this means that the counting of covered examples for each class does not start at 0, but starts at 1. So the number of examples that are covered by the rule $R_k$ is increased by one for each class, increasing the total sum by $|C|$, the total number of classes.

## 4 Experimental Setup

In the local pattern discovery phase, we used CHARM for the discovery of closed frequent itemsets. As all different closed frequent itemset discovery methods yield the same result and do only differ in their performance, we chose the state-of-the-art algorithm CHARM (Zaki and Hsiao, 2002) which features both a good time and space performance.

CHARM was adapted to multi-class rule induction as follows: Each itemset holds the absolute support (as a list of all examples containing the itemset) for each class of the data set. CHARM was altered to manage this kind of itemsets in the same way as handling the unsupervised itemsets it was designed for. With this modifications we could apply CHARM to the data of each class (referred as a segment) operating on the data of the respective class normally but also updating the supports for all other classes. Afterwards, we combined the results of each segment into a single set of closed class association rules, merging (if necessary) rules which are closed for different classes. For each segment the minimum support was adjusted to 3% of its size. Additionally we required that the respective itemset must contain at least 2 instances. Note that the first phase has only to be computed once, the results one obtains can be stored for later use.

For the second phase, we implemented the pattern set discovery algorithms we described briefly above, selecting either all patterns, only those whose confidence meets some confidence threshold, or wrapping a pattern set with joint entropy or exclusive coverage. As the minimum confidence should depend on the number of the classes each data set contains and should preferably be significant but not too restrictive, we set the minimum confidence to the reciprocal value

**Table 1.** Data sets

| | | Attributes | | | | Patterns (Mean) | |
|---|---|---|---|---|---|---|---|
| Data set | Instances | Nominal | Numeric | Classes | Default | Total | Confident |
| Balance-scale | 625 | 0 | 4 | 3 | 46.08 | 118.3 | 65.1 |
| Breast-cancer | 286 | 10 | 0 | 2 | 70.28 | 2726.7 | 1793.1 |
| Diabetes | 768 | 0 | 8 | 2 | 65.1 | 563.6 | 392.7 |
| Glass | 214 | 0 | 9 | 7 | 35.51 | 208 | 187.3 |
| Heart-c | 303 | 7 | 6 | 5 | 54.46 | 12543.8 | 11981.9 |
| Heart-h | 294 | 7 | 6 | 5 | 63.95 | 2002.7 | 1802.8 |
| Heart-statlog | 270 | 0 | 14 | 2 | 55.56 | 3507.9 | 2485.3 |
| Iris | 150 | 0 | 4 | 3 | 33.33 | 31.1 | 27.6 |
| Labor | 57 | 8 | 8 | 2 | 64.91 | 370.8 | 347.4 |
| Zoo | 101 | 16 | 1 | 7 | 40.59 | 213.9 | 213.9 |

of the number of classes. Thus, for most of the data set we obtained different minimum thresholds. For the wrapper approaches we decided to use rather small pattern sets consisting of 25 patterns. As in the first phase the results of phase 2 can be stored for future use.

We implemented the global modeling techniques ($BR$,$V$,$WV$,$IV$,$LV$,$BD$) described in the previous section for the third phase. Analogous to the second phase the unweighted voting can be considered as the neutral method for the third phase as it uses the unweighted and unbiased information of each class association rule.

In all phases, we used the following rule properties for tie breaking (in descending order of relevance): the heuristic value (Laplace), the number of correctly predicted examples, the number of examples of the predicted class, and the size of the rule. If these criteria did not discriminate between two rules, we chose one of them at random.

For the evaluation of the resulting classifiers we employed a stratified ten-fold cross validation using the mean value and standard deviation of the accuracies obtained for comparison.

For our experiments we used data sets of the UCI repository ((Asuncion and Newman, 2007)). These data sets were chosen for a great variety of the number of instances and classes, and of different ratios between numerical and nominal attributes. For our experiments numerical attributes have been discretized ((Fayyad and Irani, 1993)) separately for each cross validation fold using only the information contained in its training data. Missing (numerical and nominal) attribute values were ignored. The statistical properties of the used data sets are displayed in Table 1 which contains the number of classes, instances, attributes (separate for numerical and nominal attributes) and the percentage of instances belonging to the most represented class. Additionally, it includes the mean of the number of all patterns and respectively of the confident patterns we obtained in our experiments.

## 5 Results

In this section, we present the results of our experimental study, organized by different pattern set selection techniques. For the evaluation of the results we

**Table 2.** All Patterns: Accuracy

| | BR | | V | | WV | | LV | | IV | | BD | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Data set | Mean | Dev | Mean | Dev | Mean | Dev | Mean | Dev | Mean | Dev | Mean | Dev |
| Balance-s. | 70.88 | 7.33 | 73.30 | 6.41 | 75.51 | 6.06 | 8.95 | 3.34 | 8.00 | 1.49 | 60.49 | 6.21 |
| Breast-c. | 70.32 | 9.40 | 70.30 | 6.76 | 74.14 | 6.91 | 29.69 | 6.88 | 29.35 | 6.69 | 70.31 | 6.88 |
| Diabetes | 74.35 | 7.28 | 75.13 | 7.09 | 74.35 | 5.88 | 42.31 | 9.36 | 29.82 | 7.98 | 73.44 | 5.20 |
| Glass | 55.24 | 10.85 | 52.84 | 9.74 | 58.59 | 12.00 | 39.44 | 15.21 | 16.54 | 12.81 | 58.42 | 14.24 |
| Heart-c | 83.15 | 7.41 | 78.85 | 5.34 | 83.82 | 6.38 | 53.47 | 8.96 | 17.48 | 8.57 | 74.56 | 6.36 |
| Heart-h | 63.59 | 30.12 | 66.1 | 26.23 | 69.08 | 29.10 | 30.18 | 22.28 | 22.33 | 20.37 | 78.70 | 21.54 |
| Heart-s. | 80.37 | 7.42 | 81.11 | 7.50 | 83.70 | 6.10 | 40.37 | 7.50 | 16.30 | 5.58 | 74.44 | 5.64 |
| Iris | 86.67 | 17.21 | 82.67 | 21.82 | 91.33 | 10.91 | 81.33 | 21.03 | 53.33 | 31.47 | 58.00 | 46.83 |
| Labor | 75.67 | 24.14 | 67.00 | 31.83 | 67.00 | 31.83 | 67.00 | 30.85 | 29.33 | 17.76 | 81.33 | 19.95 |
| Zoo | 92.00 | 11.35 | 93.00 | 10.59 | 92.00 | 11.35 | 90.00 | 10.54 | 87.00 | 14.94 | 90.00 | 10.54 |
| Mean | 75.22 | 13.25 | 74.03 | 13.33 | 76.95 | 12.65 | 48.27 | 13.60 | 30.95 | 12.77 | 71.97 | 14.34 |

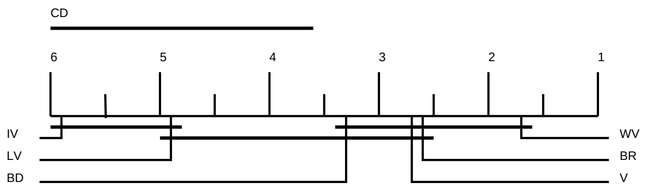

**Fig. 2.** All patterns: CD chart

used the Friedman test with a post-hoc Nemenyi test if necessary as proposed in (Demsar, 2006). The significance level was set to 95% for both tests.

The four different pattern set selection techniques employed were treated as separate test cases. As the Friedman test showed that the employed global modeling techniques are not equivalent for all test cases, we applied a Nemenyi test to each case. The results of these tests are each depicted in a separate CD chart.

### 5.1 Selecting All Patterns

First we will have a look at the results that we obtained by applying the abovementioned aggregation techniques to all generated patterns (see Table 2 and Figure 2).

Regarding the chart we can identify three groups of methods whose members do not differ significantly. The best group consists of the methods WV, BR, V, and BD. The remaining the methods IV and LV belong to the worst group. The third group overlaps with both groups, but contains only one member of the worst group. So we disregard the third group and consider only the best and worst group for our comparisons.

The first observation we make is that the methods BR, V, WV, and BD are not significantly different, although the method WV outperforms the other in most cases. Additionally, all four methods significantly outperform the methods IV and LV which are not significantly different.

**Table 3.** Confident Patterns: Accuracy

| | BR | | V | | WV | | LV | | IV | | BD | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Data set | Mean | Dev | Mean | Dev | Mean | Dev | Mean | Dev | Mean | Dev | Mean | Dev |
| Balance-s. | 70.88 | 7.33 | 71.70 | 7.93 | 73.28 | 7.26 | 68.49 | 8.74 | 50.62 | 23.50 | 70.57 | 70.31 |
| Breast-c. | 69.96 | 8.80 | 74.50 | 7.92 | 74.50 | 7.92 | 74.86 | 7.66 | 74.86 | 7.17 | 70.57 | 9.00 |
| Diabetes | 74.35 | 7.28 | 71.09 | 6.30 | 72.13 | 5.91 | 68.22 | 6.46 | 67.58 | 6.93 | 73.31 | 5.09 |
| Glass | 55.24 | 10.85 | 52.40 | 9.29 | 58.59 | 11.13 | 44.98 | 13.73 | 34.63 | 11.67 | 63.79 | 16.98 |
| Heart-c | 83.15 | 7.41 | 83.16 | 5.32 | 83.82 | 6.38 | 73.25 | 6.25 | 22.43 | 5.76 | 74.23 | 6.52 |
| Heart-h | 63.59 | 30.12 | 68.40 | 29.01 | 69.43 | 28.80 | 50.57 | 23.85 | 22.32 | 17.68 | 61.78 | 47.10 |
| Heart-s. | 80.37 | 7.42 | 82.22 | 6.25 | 82.96 | 6.34 | 81.11 | 6.86 | 72.59 | 9.27 | 75.19 | 4.64 |
| Iris | 86.67 | 17.21 | 86.00 | 15.53 | 91.33 | 10.91 | 86.67 | 13.70 | 70.00 | 29.19 | 55.33 | 41.70 |
| Labor | 75.67 | 24.14 | 68.67 | 33.19 | 68.67 | 33.19 | 67.00 | 29.83 | 68.67 | 31.28 | 84.33 | 14.74 |
| Zoo | 92.00 | 11.35 | 93.00 | 10.59 | 92.00 | 11.35 | 90.00 | 10.54 | 87.00 | 14.94 | 90.00 | 10.54 |
| Mean | 75.19 | 13.19 | 75.11 | 13.13 | 76.67 | 12.92 | 70.52 | 12.76 | 57.07 | 15.74 | 71.91 | 22.66 |

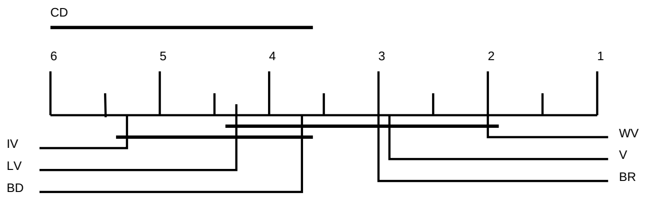

**Fig. 3.** Confident patterns: CD chart

So we come to the conclusion that the group of methods BR, V, WV, and BD perform best, whereby the method WV is the best choice. The other methods are in descending order of performance LV and IV.

### 5.2 Selecting only Confident Patterns

Next we will evaluate the results that we obtained employing only the confident patterns (see Table 3 and Figure 3). This time, we can identify two groups of methods. The first group consisting of BR, V, and WV is the best group and is significantly better than the second group compromising the method IV. We cannot tell to which one of these groups the methods BD and LV belong. Nevertheless the method WV is the best representative of the better group and so the best choice for this scenario.

So the conclusions of these experiments are very similar to the previous ones. The group of the methods BR, V, and WV has the best performance. As before the best representative is the method WV. The (descending) order of performance remains unchanged: BD, LV and IV.

If we compare the results using all patterns and confident patterns respectively one can see that these depend strongly on the employed global modeling methods. There is a marginal decrease in performance for BR, WV and BD if we use only confident patterns and a marginal improvement respectively for the method V. The method IV and LV benefit from confident patterns by an improvement of 22,3% and 26,1% in the mean.

**Table 4.** Joint Entropy: Accuracy

| | BR | | V | | WV | | LV | | IV | | BD | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Data Set | Mean | Dev | Mean | Dev | Mean | Dev | Mean | Dev | Mean | Dev | Mean | Dev |
| Balance-Scale | 43.49 | 15.47 | 54.37 | 22.96 | 44.44 | 16.27 | 18.48 | 16.18 | 9.61 | 5.34 | 49.13 | 12.02 |
| Breast-Cancer | 68.58 | 7.47 | 68.20 | 8.60 | 67.51 | 8.68 | 47.80 | 12.54 | 47.41 | 12.91 | 68.58 | 7.47 |
| Diabetes | 67.83 | 6.58 | 64.59 | 8.15 | 68.23 | 6.17 | 42.46 | 8.97 | 39.21 | 7.58 | 66.66 | 6.99 |
| Glass | 50.54 | 16.38 | 44.00 | 14.35 | 49.13 | 14.59 | 33.81 | 14.82 | 19.37 | 15.22 | 49.55 | 12.71 |
| Heart-C | 63.92 | 14.36 | 61.00 | 12.30 | 64.25 | 14.89 | 38.96 | 12.78 | 36.66 | 14.09 | 60.32 | 11.41 |
| Heart-H | 72.17 | 30.67 | 58.97 | 37.88 | 74.62 | 25.46 | 28.49 | 28.80 | 29.92 | 31.12 | 79.72 | 20.96 |
| Heart-Statlog | 73.70 | 9.15 | 68.89 | 10.06 | 75.56 | 9.11 | 38.15 | 23.75 | 35.93 | 18.73 | 69.63 | 11.15 |
| Iris | 90.00 | 12.67 | 80.67 | 25.81 | 92.00 | 9.32 | 80.67 | 22.32 | 56.67 | 33.00 | 60.00 | 40.86 |
| Labor | 75.33 | 34.75 | 70.33 | 32.56 | 73.67 | 73.67 | 68.67 | 31.28 | 70.33 | 31.60 | 80.33 | 33.39 |
| Zoo | 48.27 | 23.77 | 48.27 | 23.77 | 48.27 | 23.77 | 48.27 | 23.77 | 48.27 | 23.77 | 48.27 | 23.77 |
| Mean | 65.38 | 17.13 | 61.93 | 19.64 | 65.77 | 20.19 | 44.58 | 19.52 | 39.34 | 19.34 | 63.22 | 18.07 |

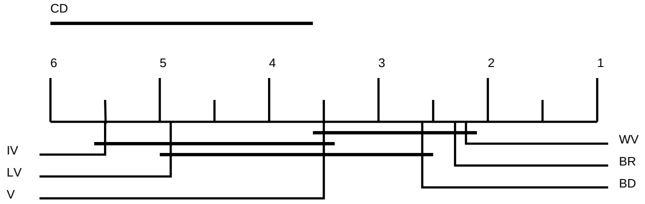

**Fig. 4.** Joint Entropy: CD chart

### 5.3 Selection by Joint Entropy

Next we will investigate the results obtained by using joint entropy for pattern set discovery (see Table 4 and Figure 4). Like in the previous experiments the methods BR, V, and WV do not differ significantly, WV dominates this group slightly. This time BD is definitely comparable to these methods. The methods IV and LV have equal performances. They are significantly worse than the methods BR and BD but they are comparable to V and in the case of LV to BD.

Using joint entropy decreases the accuracy of all methods by about 10% to 15%, but in the light of the small number of patterns chosen the results are not as bad as it might seem as it shows that the quantity is less important than the quality of patterns.

### 5.4 Selection by Exclusive Coverage

Our last experiment considered the exclusive coverage for global modeling (see Table 5 and Figure 5). As in the experiment using joint entropy the methods BR, V, WV, and BD are comparable, though this time BR and BD dominate the group. In the mean BR has the higher accuracy, but BD has the higher number of wins. Additionally all methods are significantly better than the methods LV and IV. The only exceptions are V and WV which are comparable to LV.

**Table 5.** Exclusive Coverage: Accuracy

| | BR | | V | | WV | | LV | | IV | | BD | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Data Set | Mean | Dev | Mean | Dev | Mean | Dev | Mean | Dev | Mean | Dev | Mean | Dev |
| Balance-Scale | 62.23 | 13.19 | 65.29 | 13.95 | 59.99 | 13.73 | 21.25 | 15.71 | 8.48 | 2.63 | 63.06 | 9.24 |
| Breast-Cancer | 72.08 | 9.96 | 69.98 | 7.82 | 70.70 | 12.66 | 28.97 | 6.71 | 27.92 | 7.13 | 70.70 | 12.66 |
| Diabetes | 71.35 | 5.33 | 71.22 | 5.42 | 71.22 | 5.42 | 36.33 | 8.38 | 36.20 | 8.18 | 71.35 | 5.33 |
| Glass | 35.11 | 16.04 | 31.49 | 15.68 | 31.52 | 16.20 | 16.49 | 9.71 | 13.68 | 13.68 | 32.45 | 16.04 |
| Heart-C | 58.06 | 8.33 | 58.06 | 8.33 | 58.06 | 8.33 | 45.57 | 7.26 | 45.57 | 7.26 | 58.06 | 8.33 |
| Heart-H | 53.45 | 50.14 | 50.78 | 39.48 | 50.78 | 39.48 | 40.00 | 40.19 | 39.66 | 40.55 | 60.46 | 37.56 |
| Heart-Statlog | 60.74 | 5.30 | 60.74 | 5.30 | 60.74 | 5.30 | 47.78 | 8.27 | 47.04 | 6.06 | 60.74 | 5.30 |
| Iris | 82.00 | 17.51 | 78.00 | 18.61 | 85.33 | 17.16 | 75.33 | 20.38 | 50.68 | 29.68 | 52.00 | 42.49 |
| Labor | 64.00 | 28.62 | 60.33 | 26.64 | 64.00 | 28.62 | 60.33 | 26.64 | 58.00 | 30.64 | 64.00 | 28.62 |
| Zoo | 78.27 | 10.13 | 79.27 | 10.79 | 79.27 | 10.79 | 77.27 | 9.30 | 76.27 | 9.51 | 81.27 | 12.63 |
| Mean | 63.73 | 16.46 | 62.52 | 15.20 | 63.16 | 15.77 | 44.93 | 15.26 | 40.35 | 15.53 | 61.41 | 17.82 |

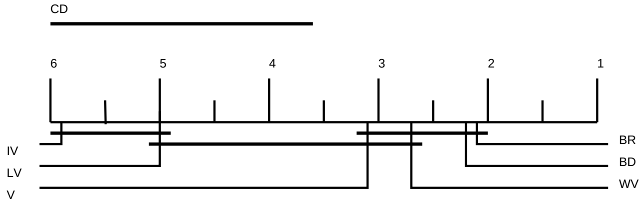

**Fig. 5.** Exclusive Coverage: CD chart

## 6 Conclusions

In this paper, we separated class association rule mining into three different steps: local pattern discovery, pattern set discovery and global modeling. As there are several exchangeable methods for each of these steps, we briefly surveyed some of the more popular techniques, and experimentally compared these different options.

For our experiments we extended the closed frequent itemset mining algorithm mining CHARM and adapted it for application to class association rule mining for multi-class problems. Hereby we obtained sets of closed association rules which were additionally filtered by a class-wise minimum confidence threshold or wrapped using joint entropy or exclusive coverage. These rules were then filtered using confidence or joint entropy. The resulting pattern sets were then combined with voting methods and a Bayesian approach.

In our experiments the methods Best Rule, Voting, Weighted Voting, and Bayesian Decoding were outperforming all other methods. Weighted Voting was in most of the cases slightly better than the others, and we consider it the best choice. Interestingly, these methods did not profit of the filtering of confident patterns or of the wrapping using joint entropy. Only the Bayes Decoding and the remaining methods Inverse Voting, and Linear Voting saw some improvement through these constraints.

**Acknowledgements**

We would like to thank the anonymous reviewers for the comments. This research was supported by the *German Science Foundation (DFG)* under grant FU 580/2.

## References

Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules in large databases. In Jorge B. Bocca, Matthias Jarke, and Carlo Zaniolo, editors, *VLDB*, pages 487–499. Morgan Kaufmann, 1994. ISBN 1-55860-153-8.

A. Asuncion and D.J. Newman. UCI machine learning repository, 2007. URL http://www.ics.uci.edu/∼mlearn/MLRepository.html.

Bruno Crémilleux, Johannes Fürnkranz, Arno Knobbe, and Martin Scholz. From local patterns to global models: The LeGo approach to data mining. Technical Report TUD-KE-2007-06, Knowledge Engineering Group, Technische Universität Darmstadt, Hochschulstrasse 10, D-64289 Darmstadt, Germany, 2007.

Luc De Raedt and Albrecht Zimmermann. Constraint-based pattern set mining. In *SDM*. SIAM, 2007.

Janez Demsar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

Usama M. Fayyad and Keki B. Irani. Multi-interval discretization of continuous-valued attributes for classification learning. In *IJCAI*, pages 1022–1029, 1993.

Johannes Fürnkranz. Separate-and-conquer rule learning. *Artificial Intelligence Review*, 13(1):3–54, February 1999. URL http://www.ofai.at/cgi-bin/tr-online?number+96-25.

Bart Goethals. Frequent set mining. In Oded Maimon and Lior Rokach, editors, *The Data Mining and Knowledge Discovery Handbook*, pages 377–397. Springer, 2005. ISBN 0-387-24435-2.

Jiawei Han, Jian Pei, Yiwen Yin, and Runying Mao. Mining frequent patterns without candidate generation: A frequent-pattern tree approach. *Data Min. Knowl. Discov.*, 8(1):53–87, 2004.

Willi Klösgen. Data mining tasks and methods: Subgroup discovery. In *Handbook of Data Mining and Knowledge Discovery*, pages 354–361. Oxford University Press, Inc., New York, NY, USA, 2002. ISBN 0-19-511831-6.

Arno J. Knobbe and Eric K. Y. Ho. Maximally informative k-itemsets and their efficient discovery. In Tina Eliassi-Rad, Lyle H. Ungar, Mark Craven, and Dimitrios Gunopulos, editors, *KDD*, pages 237–244. ACM, 2006a. ISBN 1-59593-339-5.

Arno J. Knobbe and Eric K. Y. Ho. Pattern teams. In Johannes Fürnkranz, Tobias Scheffer, and Myra Spiliopoulou, editors, *PKDD*, volume 4213 of *Lecture Notes in Computer Science*, pages 577–584. Springer, 2006b. ISBN 3-540-45374-1.

Bing Liu, Wynne Hsu, and Yiming Ma. Integrating classification and association rule mining. In *KDD*, pages 80–86, 1998.

Katharina Morik, Jean-François Boulicaut, and Arno Siebes, editors. *Local Pattern Detection, International Seminar, Dagstuhl Castle, Germany, April 12-16, 2004, Revised Selected Papers*, volume 3539 of *Lecture Notes in Computer Science*, 2005. Springer. ISBN 3-540-26543-0.

Stefan Mutter. Classification using association rules. Master's thesis, Department of Computer Science, University of Freiburg, Germany, March 2004.

Stefan Wrobel. An algorithm for multi-relational discovery of subgroups. In Henryk Jan Komorowski and Jan M. Zytkow, editors, *PKDD*, volume 1263 of *Lecture Notes in Computer Science*, pages 78–87. Springer, 1997. ISBN 3-540-63223-9.

Mohammed Javeed Zaki and Ching-Jiu Hsiao. Charm: An efficient algorithm for closed itemset mining. In Robert L. Grossman, Jiawei Han, Vipin Kumar, Heikki Mannila, and Rajeev Motwani, editors, *SDM*. SIAM, 2002. ISBN 0-89871-517-2.