Technische Universität Darmstadt
Knowledge Engineering Group
Hochschulstrasse 10, D-64289 Darmstadt, Germany

http://www.ke.informatik.tu-darmstadt.de

Technical Report TUD–KE–2008–03

Jan-Nikolas Sulzmann, Johannes Fürnkranz

An empirical comparison of techniques for selecting and combining local patterns into a global model

An empirical comparison of techniques for selecting and combining local patterns into a global model

Jan-Nikolas Sulzmann SULZMANN@KE.INFORMATIK.TU-DARMSTADT.DE
Johannes Fuernkranz JUFFI@KE.INFORMATIK.TU-DARMSTADT.DE
TU Darmstadt, Hochschulstrasze, D-64829 Darmstadt, Germany

Abstract

Local pattern discovery, pattern set discovery and global modeling build together as consecutive steps a specific case of global pattern discovery. 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. Though so far there has been no systematic comparison of the possible choices. In this paper, we will consider a special representative of local patterns, namely class association rules, and evaluate several options for pattern set discovery and for global modeling for this type of classification rules.

1. Introduction

Classification association rule mining is basically the integration of two opossitional 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 after the other in a separate and conquer style exploiting the interaction with previous rules. However in association rule mining all rules in the databases that satisfy some minimum interestingness constraints (e.g. minimum support or confidence) are generated more or less 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, Hsu, & Ma, 1998) and is done by concentrating on a specific subset of associations, namely class association rules (abbreviated CAR), which can be used for classification.

Recapitulatory classification association rule mining is a specific type of global pattern discovery as it is segmentable into three consecutive tasks. 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 discovery, selects a optimal subset of the previously generated association rules. Optimality of the subset is accordant to one or more arbitrary selectable heuristic that estimate the usefulness of the subset for future predictions using (un-)supervised information content. In most cases this task is accomplished by wrapper or filter approach, or basically can be reduced two one of this two cases. Note that in the first step only association rules are evaluated independently of each other, while in the second step it is possible that single rules or subsets are evaluated independently of each other or with inclusion of partial or total information contained in the data base. However in both phases unsupervised or supervised evaluation measures can be employed.

The local pattern discovery has gained an increased attention (Morik, Boulicaut, & Siebes, 2005) in recent years, resulting in a great variety of techniques for the generation of frequent local patterns or transferred to our case of frequent class association rules (Agrawal & Srikant, 1994; Han, Pei, Yin, & Mao, 2004). Each of these implementations yield slightly the same result only modified by additional constraints (e.g. closeness (Zaki & Hsiao, 2002)). This in mind we concentrate one 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 described in the following sections for these steps. 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 applied methods, respectively. Section 4.1 describes the experimental setup and evaluate the results, before section 5 concludes.

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 this both techniques are fused for class association rule mining and what 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. Classification rules consist similar to instances of some attribute values for mutually exclusive attributes building the body or premise of the rule and a single predicted class forming the head or conclusion $(d = (A_k = a_1^{j_k} \wedge A_l = a_2^{j_2} \ldots \rightarrow c)$. A rule covers an example if the example meets premise of the rule which is then called a covering rule. If a rule covers an example, the class in the conclusion is predicted. How these 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 unique transaction identifier $tid \in T$, where T ist 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 association rule $X \rightarrow Y$ consists of two itemsets X and Y. X forms the body or 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.

In classification association terms of classification and association rule mining are used in conjunction, as it combines some features of classification and association rule mining. A class association rule (abbreviated CAR) r is an implication of the form $X \rightarrow y$, where $X \subseteq I$ is an itemset, and $y \subseteq Y$ 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 algorithms 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 association rule mining but can be transformed into a association data set. For example, numeric attributes can be discretized (e.g., (Fayyad & 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. Global Pattern Discovery

In this section we give a short introduction into a special case of global pattern discovery which consists as before mentioned of three consecutive tasks: 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, Fürnkranz, Knobbe, & Scholz, 2007)

For our convenience the local pattern discovery was adjusted to 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 & Hsiao, 2002) which features both a good time and space performance.

3.1 Local Pattern Discovery

Despite there are other categories of local pattern discovery (e.g. subgroup discovery (Wrobel, 1997; Klösgen, 2002)) we concentrate on frequent itemset 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, but for the purpose of comparison this will not have a severe consequences.

Itemsets can be considered as local patterns because items describe only the instances of 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 & Hsiao, 2002).

For our experiments we chose the local pattern discovery algorithm CHARM (Zaki & Hsiao, 2002) which is an effective algorithm for the enumeration of close 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 & Hsiao, 2002) which provides a survey of this specific type of global pattern discovery.

3.2 Pattern Set Discovery

The local pattern discovery phase generates pattern 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. This 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 this 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 (Raedt & Zimmermann, 2007) and pattern teams (Knobbe & Ho, 2006a, 2006b). 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 & Hsiao, 2002)

For this work we will consider only two simple representatives of pattern set discovery. The first one is not obviously a pattern set discovery, as it selects all 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.

3.3 Global Modeling

There are many variants for the global modeling of pattern sets and in our case class association rule sets, so we confined ourselves to choosing some methods for two groups of global modeling. These groups have two facts in common. First they determine the rules which cover the instance to be classified and integrates their predictions into a final prediction. Second each rule has a weight for each (predicted) class. We decided to use the laplace heuristics for the evaluation and ranking of a rule because this heuristic is always calculable for each class for every single rule.

The first group are voting methods which use the covering rules as votes for the final prediction. The vote of a single rule can be weighted using either its laplace value directly or a ranking based value according to its laplace value. The second group are probabilistic methods which use estimated probabilities as the final prediction.

3.3.1 Voting Methods

Common ground of all voting methods is as its name suggest the appliance of votes for a prediction but they differ in the weights they assign to a vote of a rule. So essentially the classification works as follows:

$$\arg\max_{c_i \in C} \sum_{r \in R_{c_i}} weight(r), \tag{1}$$

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. Independently of the method the rules should be sorted descending or ranked in advance using the laplace value of each rule according to the predicted class. Note that this is an arbitrary choice as other heuristics would also be appropriate. The resulting ranking of rules can be considered as a decision list beginning with the best rules and ending with the worst.

The first representative *Best Rule* (abbrev. BR) considers as hinted by its name 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 behaviour. One possible solution is setting $weight_{BR}$ for the best rule to one (or any other nonzero value) and for all other rules to zero.

The next representatives are *Unweighted* and *Weighted Voting* (abbrev. UV and WV accordingly). These methods have in common that they use the weights of all covering rules. UV 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 of each rule as its weight, so basically the laplacian weights are counted for each class.

$$weight_{UV}(r) = 1 \quad weight_{WV}(r) = laplace(r) \tag{2}$$

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 ($rank_{max}$).

$$weight_{LV} = 1 - \frac{rank(r)}{rank_{max} + 1} weight_{IV} = \frac{1}{rank(r)} \tag{3}$$

3.3.2 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 $R = R_1 \wedge R_2 \wedge \ldots \wedge R_s$ that cover the example, namely $\Pr(c_i|R)$,and the prediction of the most probable class.

$$\arg\max_{c_i \in C} \Pr(c_i|R) \tag{4}$$

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

$$\Pr(c_i|R) = \frac{\Pr(R|c_i)}{\Pr(R)} \tag{5}$$

As the denominator $\Pr(R_1 \wedge R_2 \wedge \ldots \wedge R_s)$ 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 independent of the occurrence of the other we can make the following naïve assumption:

$$\Pr(R|c_i) = \Pr(R_1 \wedge R_2 \wedge \ldots \wedge R_s|c_i) = \prod_{k=1}^{s} \Pr(R_k|c_i) \tag{6}$$

Finally the classification works as follows:

$$\arg\max_{c_i \in C} \Pr(c_i) \cdot \prod_{k=1}^{s} \Pr(R_k|c_i) \tag{7}$$

Remains to be explained how these probabilities can be estimated. The first one $\Pr(c_i)$ 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. The second one $\Pr(R_k|c_i)$ can be rated slightly different 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. Avoiding this problem, we apply the laplace heuristic. So the number of examples that are covered by the rule R_k is increased by one for each class and the outcome of this is that the total sum is increased by $|C|$, the total number of classes.

4. Experiments

In this section we will describe at first the setup of our experiments specifying the employed methods and the used data sets. After that we will evaluate the results we obtained comparing them with each other and concluding about .

4.1 Experimental Setup

Our experiments consist of the before mentioned phases of global pattern discovery. We used the CHARM for the discovery of closed frequent itemsets. Most of the times it was applied to multiclass data sets, but CHARM is designed for unsupervised data. We solved this problem by slightly modifying CHARM and the itemsets to which it was applied. 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 operating on the data of the respective class normally but also updating the supports for all other classes. Afterward we combined the results into a single set of closed class association rules merging if necessary rules which are closed for different classes. The minimum support was adjusted for each segment to 3%. Additionally we demanded that at the respective itemset must contain 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 or only those whose confidence meets some confidence threshold. As the minimum confidence should depend on the number of the classes each data set contains and should be preferably significant but not too restrictive, we set it to the reciprocal value of the number of classes. So for most of the data set we obtained different minimum thresholds.

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

In all phases for the tie breaking for class associations rules the following rule properties were used (in descending order of relevance): the heuristic value (laplace), the number of correct predicted examples, the number of examples of the predicted class and the number of AV-Pairs. If these did not discriminate between two rules one was randomly chosen.

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. 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. The statistical properties of the used data sets are displayed in table 4.1 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 and standard deviation of the number of patterns we obtained in our experiments.

4.2 Experimental results

At first we will have a look at table 4.1 and especially at the number of patterns for each data sets. Note that we will not always explicitly say that we are talking about mean numbers. Against all expectations it seems that the number of instances and classes respectively are

		Attributes				Patterns	
Data set	Instances	Nominal	Numeric	Classes	Default	Mean	Dev
Autos	205	10	15	7	32,68	12356,9	3787,18
Balance-scale	625	0	4	3	46,08	65,1	4,28
Breast-cancer	286	10	0	2	70,28	1793,1	42,71
Breast-w	699	0	9	2	65,52	11981,9	691,83
Diabetes	768	0	8	2	65,1	187,3	24,43
Glass	214	0	9	7	35,51	2485,3	616,25
Heart-c	303	7	6	5	54,46	1802,8	512,9
Heart-h	294	7	6	5	63,95	27,6	7,53
Heart-statlog	270	0	14	2	55,56	347,4	43,02
Iris	150	0	4	3	33,33	14701,6	1181,1
Labor	57	8	8	2	64,91	392,7	184,6
Lymph	148	16	2	4	33,61	9670,9	1499,9
Vowel	990	3	10	11	9,09	3098,4	131,86
Zoo	101	16	1	7	40,59	213,9	14,33

Table 1: Data sets

	All Patterns		Confident Patterns	
	Mean	Dev	Mean	Dev
Instances	-0,11	-0,29	-0,1	-0,28
Classes	-0,01	0,22	0	0,21
Nominal	0,01	0,25	0	0,25
Numeric	0,11	0,32	0,128	0,39

Table 2: Correlation: Patterns

not correlated with the mean number of patterns generated. For example the balance-scale data set being one of the greater data sets has only about 65 patterns contrary to the small data set Lymph which has about 9670 patterns. A converse example are the data sets Breast-w and Zoo as the greater one has more patterns as the other. For the number of classes similar examples can be found (e.g. breast-w and vowel, or autos and diabetes). Therefore one can conclude that the number of patterns depends whether on the number of instances nor on the number of classes. For circumstantiating this observation we calculated the pearson's correlation coefficient for the number of classes, instances, and numeric and nominal attributes compared the mean and the standard deviation of patterns (see table 4.2). The coefficients we obtained proved that there is indeed no correlation between these values and mean number of patterns. Only the coefficients for the standard deviation showed some small correlations but as these are nearly identical for all tested properties they seem to be insignificant.

Now we will have a look at the results that we obtained by applying the before mentioned methods of global modeling to all generated patterns (see table 4.2). For the evaluation of the results we used the standard sign test with significance levels of 95% (significant) and 99% (highly significant). Herefore we used a table that contains the wins, losses, and ties for a pair of methods (see table 4.2).

The first observation we make is that the methods BR, V, and WV are not significantly different. Though the method WV outperforms the other in most of the times. Additionally all three just mentioned methods are highly significant better than the methods IV and LV and outperform (but not significantly) the method Bayes in most of the times. Note that the method WV is even significant better than the method Bayes. The method LV outperforms

	BR		V		WV		LV		IV		Bayes	
Data set	Acc	Dev	Acc	Dev	Acc	Dev	Acc	Dev	Acc	Dev	Acc	Dev
Autos	42,14	24,17	39,05	27,38	39,98	27,64	37,57	21,43	20,93	9,06	8,83	13,66
Balance-s.	70,88	7,33	73,3	6,41	75,51	6,06	8,95	3,34	8	1,49	60,49	6,21
Breast-c.	70,32	9,4	70,3	6,76	74,14	6,91	29,69	6,88	29,35	6,69	70,31	6,88
Breast-w	88,71	8,74	96,86	2,21	96,57	3,38	95,53	2,5	37,89	17,56	74,4	8,77
Diabetes	74,35	7,28	75,13	7,09	74,35	5,88	42,31	9,36	29,82	7,98	73,44	5,2
Glass	55,24	10,85	52,84	9,74	58,59	12	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	29,1	30,18	22,28	22,33	20,37	78,7	21,54
Heart-s.	80,37	7,42	81,11	7,5	83,7	6,1	40,37	7,5	16,3	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	46,83
Labor	75,67	24,14	67	31,83	67	31,83	67	30,85	29,33	17,76	81,33	19,95
Lymph	78,43	14,27	77,76	10,81	78,43	9,74	68,24	14,75	31,19	18,61	2,05	4,56
Vowel	42,83	11,05	30,81	6,18	45,25	4,76	17,17	4,44	7,77	4,93	18,59	12,9
Zoo	92	11,35	93	10,59	92	11,35	90	10,54	87	14,94	90	10,54

Table 3: Results: All Patterns

	V	WV	LV	IV	Bayes
BR	8-6-0	2-9-3	13-1-0	14-0-0	11-3-0
V	-	3-10-1	13-0-1	14-0-0	10-4-0
WV	-	-	13-0-1	14-0-0	12-2-0
LV	-	-	-	14-0-0	4-9-1
IV	-	-	-	-	2-12-0

Table 4: All Patterns: Win-Loss-Tie

the method IV highly significant and is insignificant worse than Bayes. The results of Bayes are highly significant better than those of the method IV.

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

Next we will evaluate the results that we obtained employing only the confident patterns (see table 4.2) and table 4.2). For the evaluation of the results we used the standard sign test as described above.

Like before the methods BR, V, and WV are not significantly different and are slightly better than the other methods. All these methods are (highly) significant better than the methods IV and LV. The only exception are BR and IV as BR is not significant better than IV. As already seen for the previous results the methods BR, V, and WV are not significant than Bayes. Here this is also for the method WV the case. The method LV outperforms highly significantly the method IV as before seen but is now comparable to Bayes. The results of Bayes are not significant better than those of the method IV.

So the conclusions of this experiments are very similar to the previous ones. The group of 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: Bayes, 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 methods. There is a marginal worsening for BR and WV if we use only confident patterns and a marginal improvement respectively for the method V. The method BR profits of the constriction to confident patterns by about

	BR		V		WV		LV		IV		Bayes	
Data set	Acc	Dev	Acc	Dev	Acc	Dev	Acc	Dev	Acc	Dev	Acc	Dev
Autos	42,14	24,17	40,48	28,76	39,98	27,64	37,57	21,43	26,33	10,23	32,57	22,72
Balance-s.	70,88	7,33	71,7	7,93	73,28	7,26	68,49	8,74	50,62	23,5	70,57	70,31
Breast-c.	69,96	8,8	74,5	7,92	74,5	7,92	74,86	7,66	74,86	7,17	70,57	9
Breast-w	88,71	8,74	96,71	3,01	96,57	3,38	96,43	1,93	95,43	3,48	65,97	12,91
Diabetes	74,35	7,28	71,09	6,3	72,13	5,91	68,22	6,46	67,58	6,93	73,31	5,09
Glass	55,24	10,85	52,4	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	68,4	29,01	69,43	28,8	50,57	23,85	22,32	17,68	61,78	47,1
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	15,53	91,33	10,91	86,67	13,7	70	29,19	55,33	41,7
Labor	75,67	24,14	68,67	33,19	68,67	33,19	67	29,83	68,67	31,28	84,33	14,74
Lymph	78,43	14,27	77,76	9,85	79,1	9,05	70,95	15,1	46	18,42	64,19	18,62
Vowel	42,83	11,05	36,87	6,77	46,16	5,15	21,01	4,83	12,12	5,77	38,38	10
Zoo	92	11,35	93	10,59	92	11,35	90	10,54	87	14,94	90	10,54

Table 5: Results: Confident Patterns

	V	WV	LV	IV	Bayes
BR	7-7-0	3-10-1	10-3-1	12-2-0	11-3-0
V	-	3-9-2	12-2-0	12-1-1	10-4-0
WV	-	-	13-1-0	12-1-1	11-3-0
LV	-	-	-	12-1-1	6-7-1
IV	-	-	-	-	3-11-0

Table 6: Confident Patterns: Win-Loss-Tie

7% which is not an ignorable amount. For the method IV and LV these improvements are even better as they are about 24,5% and 16,5% in the mean.

5. Conclusions

In this paper we described how the class association rule mining can be segmented into three steps: local pattern discovery, pattern set discovery and global modeling. As there are several exchangeable methods for each of these steps we gave a short survey of some of the appropriate methods respectively.

For our experiments we extended the closed frequent itemset mining algorithm mining CHARM to the end that it can be applied to the closed class association rule mining for multi class problems. Hereby we obtained sets of closed association rules which were additionally filter by a class wise minimum confidence threshold. We applied some voting methods and a bayesian approach for global modeling one time to all the patterns and one time to the confident patterns only.

It turned out that the number of patterns depends only on the data set itself and is not correlated to any of its numerical properties (like the number of classes). For all and the confident patterns the methods Best Rule, Voting, and Weighted Voting were outperforming all other methods. Whereby Weighted Voting was always slightly better than the others and is therefore the best choice for global modeling so far. These methods did not profit of the additional confidence properties. Only the remaining methods Inverse Voting, Linear Voting and Bayes saw minor to major improvements through this constraint.

Appendix A. Probability Distributions for N-Queens

References

Agrawal, R., & Srikant, R. (1994). Fast algorithms for mining association rules in large databases. In Bocca, J. B., Jarke, M., & Zaniolo, C. (Eds.), *VLDB*, pp. 487–499. Morgan Kaufmann.

Crémilleux, B., Fürnkranz, J., Knobbe, A., & Scholz, M. (2007). From local patterns to global models: The LeGo approach to data mining. Tech. rep. TUD-KE-2007-06, Knowledge Engineering Group, Technische Universität Darmstadt, Hochschulstrasse 10, D-64289 Darmstadt, Germany.

Fayyad, U. M., & Irani, K. B. (1993). Multi-interval discretization of continuous-valued attributes for classification learning. In *IJCAI*, pp. 1022–1029.

Goethals, B. (2005). Frequent set mining. In Maimon, O., & Rokach, L. (Eds.), *The Data Mining and Knowledge Discovery Handbook*, pp. 377–397. Springer.

Han, J., Pei, J., Yin, Y., & Mao, R. (2004). Mining frequent patterns without candidate generation: A frequent-pattern tree approach. *Data Min. Knowl. Discov.*, *8*(1), 53–87.

Klösgen, W. (2002). Data mining tasks and methods: Subgroup discovery., 354–361.

Knobbe, A. J., & Ho, E. K. Y. (2006a). Maximally informative k-itemsets and their efficient discovery. In Eliassi-Rad, T., Ungar, L. H., Craven, M., & Gunopulos, D. (Eds.), *KDD*, pp. 237–244. ACM.

Knobbe, A. J., & Ho, E. K. Y. (2006b). Pattern teams. In Fürnkranz, J., Scheffer, T., & Spiliopoulou, M. (Eds.), *PKDD*, Vol. 4213 of *Lecture Notes in Computer Science*, pp. 577–584. Springer.

Liu, B., Hsu, W., & Ma, Y. (1998). Integrating classification and association rule mining. In *KDD*, pp. 80–86.

Morik, K., Boulicaut, J.-F., & Siebes, A. (Eds.). (2005). *Local Pattern Detection, International Seminar, Dagstuhl Castle, Germany, April 12-16, 2004, Revised Selected Papers*, Vol. 3539 of *Lecture Notes in Computer Science*. Springer.

Mutter, S. (2004). Classification using association rules. Master's thesis, Department of Computer Science, University of Freiburg, Germany.

Raedt, L. D., & Zimmermann, A. (2007). Constraint-based pattern set mining. In *SDM*. SIAM.

Wrobel, S. (1997). An algorithm for multi-relational discovery of subgroups. In Komorowski, H. J., & Zytkow, J. M. (Eds.), *PKDD*, Vol. 1263 of *Lecture Notes in Computer Science*, pp. 78–87. Springer.

Zaki, M. J., & Hsiao, C.-J. (2002). Charm: An efficient algorithm for closed itemset mining. In Grossman, R. L., Han, J., Kumar, V., Mannila, H., & Motwani, R. (Eds.), *SDM*. SIAM.