Pruning and Exclusion Criteria for Unordered Incremental Reduced Error Pruning

Henrik Boström

Department of Computer and Systems Sciences,
Stockholm University and Royal Institute of Technology,
Forum 100, 164 40 Kista, Sweden
henke@dsv.su.se
and
Compumine AB
Österögatan 3, 164 40 Kista, Sweden
henrik.bostrom@compumine.com

Abstract. Incremental reduced error pruning is a technique that has been extensively used for efficient induction of ordered rule sets (decision lists). Several criteria have been developed regarding how to prune rules and whether or not to exclude generated rules. A version of incremental reduced error pruning for unordered rule sets is presented, and the appropriateness of previously proposed criteria for the novel version is investigated. It is shown that when inducing unordered rule sets, where a Bayesian framework is used to combine predictions from multiple rules, previously proposed criteria could lead to exclusion of possibly beneficial rules as well as to inclusion of harmful rules. Two alternative criteria are introduced, one based on the *likelihood ratio* and one based on the *margin*. An empirical evaluation on 34 datasets shows that the novel criteria significantly outperform previously employed criteria when using incremental reduced error pruning for unordered rule sets, the margin-based being slightly ahead of the likelihood ratio criterion.

1 Introduction

Separate-and-Conquer (or covering) has been a popular method for inducing sets of classification rules during the last decades [10]. This method has been used for generating two main types of classifier: ordered rules sets (also known as decision lists [17]) and unordered rule sets. The former has the advantage of requiring only a simple inference mechanism (i.e., the first applicable rule is employed), while the latter requires some way to combine predictions from multiple rules (e.g., using class counts as in [4] or a more sophisticated scheme as in [14, 15]). On the other hand, an unordered rule set allows for interpreting each generated rule independently of other rules, while rules within an ordered rule set cannot be taken out of context, since the class distribution for a particular rule in an ordered set is dependent on preceding rules. Hence, multiple rules have only to be considered during classification for unordered rule sets, and not during interpretation of each generated rule, in contrast to ordered rule sets, for which multiple rules have to be considered in both cases.

One of the shortcomings of separate-and-conquer is its quadratic time complexity compared to the linear time complexity of the divide-and-conquer (or recursive partitioning) strategy [2].[1] However, with the advent of a technique known as incremental reduced error pruning (IREP) [12], by which each rule is immediately pruned after its generation, substantially larger training sets could be handled by separate-and-conquer within reasonable time (the computational cost is still $d \times n$, where d is the number of rules and n is the number of examples, but typically $d \ll n$ when using IREP, while d may approach n without IREP, especially in difficult domains). Methods for employing IREP for ordered rule sets have received considerable attention [12, 6, 9, 8], while IREP for unordered rule sets has so far received no attention in the literature.[2]

A number of criteria for deciding how to prune generated rules and for deciding whether or not to exclude a generated rule when using IREP for ordered sets of rules have previously been proposed and evaluated [12, 6, 9]. In this work, we study their usefulness for unordered rule sets and propose alternative criteria that are motivated by a Bayesian framework, which is used for combining predictions from multiple rules.

In the next section, we first recall the Bayesian framework and then describe the previous criteria for ordered rule sets. In section three, we present an adapted version of IREP for unordered rule sets, point out some weaknesses of previous criteria when used in this setting, and present two novel criteria. These are in section 4 compared empirically to previous criteria that have been proposed for ordered rule sets. Finally, we give some concluding remarks in section 5.

2 Previous Work

2.1 Combining Predictions from Multiple Rules

There have been several proposals for how to combine predictions from multiple rules [4, 14, 15]. One natural, and computationally inexpensive, way of combining predictions made by multiple rules is to choose the most probable class according to Bayes' theorem[3]:

$$P(C|R_1 \wedge \ldots \wedge R_n) = P(C)\frac{P(R_1 \wedge \ldots \wedge R_n|C)}{P(R_1 \wedge \ldots \wedge R_n)}$$

[1] The number of steps by which a rule can be specialized is here assumed to be bounded by a constant, in contrast to the derivation of $\Omega(n^2 \log n)$ for separate-and-conquer in [5].

[2] Methods for applying rules generated by IREP for unordered rules have however been studied [7, 14, 15] and the system RIPPER, which was introduced in [6], includes an option for also generating unordered rule sets that has not been described in the literature.

[3] The use of Bayes' theorem for this purpose appears to have been first described in [7], when referring to a system of the second author.

where C is a class label and $R_1, \ldots, R_n$ are the rules that cover the example to be classified. Since $P(R_1 \wedge \ldots \wedge R_n)$ is constant for all possible class labels, it can be ignored. Furthermore, since it is normally very hard to get a good estimate of $P(R_1 \wedge \ldots \wedge R_n|C)$, we adopt the commonly made (naïve) assumption that $P(R_1 \wedge \ldots \wedge R_n|C) = P(R_1|C) \ldots P(R_n|C)$.

In order to avoid that a single rule cancels out the probability for some class, we adopt the following estimate for each probability $P(R_i|C)$: We assume that each rule, in addition to the examples actually covered, also covers a fraction of an imagined example from each class, where the fraction is determined by the *a priori* probability of the class[4]. More formally:

$$P(R|C) = \frac{|Covers(R, E^C)| + |E^C|/|E|}{|E^C| + |E^C|/|E|}$$

where $Covers(R, E)$ denotes the subset of E covered by R, and E^C denotes the subset of all examples E that belong to class C.

2.2 Pruning and Exclusion Criteria for Ordered Rule Sets

In the seminal paper on incremental reduced error pruning [12], two variants of the algorithm were studied, I-REP and I-REP-2. I-REP uses a pruning criterion that maximizes the overall accuracy $\frac{p+(N-n)}{P+N}$, where p and n are the number of positive and negative examples covered by the current rule out of totally P positive and N negative in the current pruning set.[5] I-REP does not include the best pruned rule if the overall accuracy of the rule is below $\frac{N}{N+P}$ (which would be the overall accuracy of a default rule). It should be noted that the original algorithm actually stops once such a rule has been generated, and this criterion is consequently often referred to as a *stopping condition*. However, since it is not necessary to stop because of the most recent generated rule should be excluded (which is explained in section 3), we refer to such a criterion as an *exclusion criterion*.

I-REP-2 prunes rules by maximizing the "purity" $\frac{p}{n+p}$ and excludes a rule whenever this value is less or equal to 0.5 (otherwise, the rule would introduce more errors than correct classifications). In [12], I-REP was found to generate more accurate classifiers than I-REP-2. It was later realized in [6] that the pruning criterion of I-REP could lead to occasional failures (e.g., it prefers a rule that covers 2000 positive and 1000 negative examples to a rule that covers 1000 positive and 1 negative examples), and an alternative pruning criterion was proposed: $\frac{p-n}{p+n}$.

In order to avoid premature stopping (i.e., when the last generated rule is of low accuracy, but there are still some remaining positive examples to be covered), an MDL scheme was used in [6] to allow the addition of rules that

[4] This corresponds to using an *m-estimate* [3] with $m = 1$.

[5] Note that incremental reduced error pruning for ordered rule sets not only removes covered examples from the growing set, but also from the pruning set.

actually increase error up to a certain user defined threshold[6], which in turn allows possibly good rules to be found subsequently. This procedure requires that harmful rules are removed at a later stage (in [6], a rule is considered harmful if it increases the total description length). An implementation of IREP with this criterion together with the proposed pruning criterion and subsequent post-processing, called IREP*, was shown to significantly outperform previous versions of IREP. The algorithm RIPPER (also presented in [6]) combines IREP* with an additional post-processing stage, by which replacements for each rule are investigated (both by inducing entirely new rules and by specializing the rules further).

3 IREP for Unordered Rule Sets

In this work, we focus on the main IREP procedure[7] and how to adapt it to induction of unordered rule sets. We start by outlining the general procedure and discuss the differences to the original IREP algorithm. Then we investigate whether the previously proposed pruning and exclusion criteria for ordered rule sets also are useful for unordered sets of rules, assuming that the Bayesian framework presented in the last section is used. Finally, we suggest two alternative criteria motivated by this framework.

3.1 The Algorithm

The main algorithm for incremental reduced error pruning for unordered rules sets, called U-IREP, is shown in Fig. 1.

There are a number of differences between U-IREP and IREP as defined in [12, 6]. The most important difference is that each rule generated by U-IREP can be interpreted independently of the other rules, since the class distribution associated to each rule is independent of previously generated rules. This does however not mean that each rule is generated independently of previous rules. In fact, each rule is generated from the remaining examples to be covered in the grow set, but pruned using a fixed prune set, from which no examples are removed. Hence, decisions of how to prune and whether or not to exclude a generated rule are made independently of previously generated rules.

Furthermore, U-IREP generates rules for all classes (in no particular order) using examples from all other classes as negative examples when generating rules for a class, while IREP generates rules for classes in a specific order, where all examples covered by rules defining one class are ignored when generating rules for subsequent classes.[8]

[6] More correctly, the encoding of the rules and classifications is allowed to grow up to a specified number of bits.

[7] Hence post-processing of generated rules is not considered, e.g., as done in IREP* and RIPPER [6].

[8] Only two classes are handled by IREP in its original formulation, where the second class is used as a default class, while in its extended version in [6], rules for multiple classes are generated in sequence.

```
function U-IREP($Classes$,$Examples$)
  $Rules := \emptyset$
  Make stratified split of $Examples$ into $GrowSet$ and $PruneSet$
  for each $Class \in Classes$ do
    $Pos := \{e : e \in GrowSet \wedge Class(e) = Class\}$
    $Neg := GrowSet \setminus Pos$
    while $Pos \neq \emptyset$ do
      $Rule := GrowRule(Pos, Neg)$
      $Rule := PruneRule(Rule, PruneSet)$
      if not $Exclude(Rule, PruneSet)$ then $Rules := Rules \cup \{Rule\}$
      $Pos := Pos \setminus Covers(Rule, Pos)$
  return $Rules$
```

Fig. 1. The U-IREP algorithm.

Finally, instead of stopping the generation of rules when the pruned rule is considered harmful, as done in original IREP, we have chosen to simply ignore the generated rule and remove the examples covered by the rule. This allows for avoiding premature stopping so that rules can be generated from the remaining uncovered examples, without having to incorporate bad rules that need to be dealt with in a post-processing stage (like in IREP*). This alteration is however not required for unordered rule sets, and the more hasty approach of original IREP may be used instead.

3.2 Some Potential Weaknesses of Previous Criteria

Previously proposed exclusion criteria (e.g., using the threshold $\frac{N}{N+P}$ as in I-REP or $\frac{1}{2}$ as in I-REP-2) do not reflect whether or not the generated rule contributes positively to the current class in relation to other classes in the Bayesian framework. For example, consider a rule that covers 10 examples out of totally 200 examples of class A together with 5 examples out of totally 50 examples of another class B. The previous criteria would consider this to be a good rule for class A, since both the overall accuracy $\frac{10+(50-5)}{200+50}$ exceeds the threshold of $\frac{50}{250}$ and the purity $10/15$ exceeds the threshold of $1/2$. However, this rule actually gives the highest preference to class B when used in the Bayesian setting, since when applied, the likelihood ratio between B and A is multiplied by $\frac{5/50}{10/200} = 2.0$. This means that both these criteria would allow the inclusion of a rule generated for one class that actually contributes more to another class. Put in other words, they allow rules to be included that are harmful for those examples for which they were generated. Moreover, consider the same rule generated for class B. This rule would not be included if any of the previous criteria were employed, since both the overall accuracy $\frac{5+(200-10)}{200+50}$ falls below $\frac{200}{250}$ and the purity $5/15$ falls below $1/2$. However, as shown above, this rule actually contributes positively to the correct class for those examples from which it was generated. Hence, a rule that contributes strongly to a class does not have to be accurate.

Furthermore, previously suggested pruning criteria for decision lists are all local in the sense that they evaluate rules based on their coverage of positive and negative examples without considering the original class distribution. This is the case for I-REP since maximizing $\frac{p+(N-n)}{P+N}$ is equivalent to maximizing $p-n$ (since P and N are constant for all candidates in a set of rules obtained by pruning some rule), as shown in [11]. This is also obvious for the pruning criterion of I-REP-2 which maximizes the fraction $\frac{p}{n+p}$, and for the criterion $\frac{p-n}{p+n}$ used by IREP*, which was proposed as an improvement over the one used in I-REP. It turns out that maximizing the criterion of IREP* is in fact equivalent to maximizing the criterion of I-REP-2, which was pointed out in [18, p.180], since

$$\frac{p-n}{p+n} = \frac{p}{p+n} - \frac{n}{p+n} = \frac{p}{p+n} - (1 - \frac{p}{p+n}) = 2\frac{p}{p+n} - 1$$

3.3 Two novel criteria for unordered rule sets

Instead of comparing the accuracy to a fixed threshold when deciding on the inclusion or exclusion of a generated rule, one could base the decision on whether the likelihood increases or not. This objective is similar in spirit to the commonly employed weighted information gain criterion [13] used for growing rules within separate-and-conquer, which does not evaluate candidates based on the overall accuracy, but on how much examples belonging to the current class gain from the rule (i.e., potential loss for examples belonging to other classes is only indirectly considered).

For a given class C, we may calculate the likelihood ratio between a rule R to be evaluated and a (default) rule D that covers all examples (i.e., $P(D|C) = 1$):

$$Likelihood\,Ratio\,To\,Default(R, C) = \frac{P(C|R)}{P(C|D)} = \frac{P(R|C)}{P(R)}$$

This ratio is greater than 1 whenever the rule increases likelihood for the given class compared to the default rule. When this ratio is less than 1, it means that the likelihood actually decreases. Since the default rule corresponds to the most general rule that can be obtained by pruning any rule, it follows naturally that the likelihood ratio must be greater than 1 in order for a rule to be kept (otherwise, we would always allow including the default rule). Hence, we propose this as an exclusion criterion. Furthermore, this ratio may also be used as a pruning criterion, and it turns out that it is actually equivalent to the one used by I-REP-2 and IREP*, except that an *m-estimate* with $m = 1$ is employed here (see section 2.1).

Considering the same example as above, where we have a rule that covers 10 examples of class A (out of totally 200 examples belonging to that class) together with 5 examples of another class B (out of totally 50 examples), we find that the likelihood ratio to the default for class A is 0.83, while for class B it is 1.66. Hence, as opposed to the previous criteria, this rule would be excluded if it was generated for class A, while it would be included if it was generated for class B, again in contrast to the previous criteria that would exclude the rule.

It may seem counter-intuitive that the same rule at one point in time is considered harmful, and at another point in time is considered beneficial. However, since the examples that are used to grow a rule for a given class are removed, it means that if the likelihood for the class is decreased, there may be no chance to repair this. Decreasing the likelihood of examples belonging to other classes than the one for which rules currently are generated could on the other hand be affordable, since rules will be (or already have been) generated that increase their likelihood.

As pointed out in the previous section, one reason for excluding an accurate rule is that it contributes more to another class than for which it was generated. The above criterion only guarantees that such a rule is excluded in case there are two classes. When there are more than two classes, some other class may gain more from adding the generated rule, even if the likelihood ratio to default is greater than one for the current class. For example, a rule generated for class A that covers 30 examples of that class out of totally 100, 0 examples of class B out of totally 200 and 70 examples of class C out of totally 100, would be included by the likelihood ratio criterion, but would obviously give a higher likelihood to class C. Hence, the examples for which it was generated actually suffer from adding the rule.

One alternative to looking only for an increase of likelihood relative to the class itself, is to look at the likelihood of the current class in relation to the most likely other class. Instead of just investigating whether the difference in likelihood is positive for the generated rule (like previously proposed criteria do), it can be checked whether the difference grows or shrinks compared to the difference in likelihood for the default rule. If this difference increases, then the rule can be considered to contribute to the current class.[9] More formally[10]:

$$Margin\,Increase(R,C) = (P(C|R) - P(C'|R)) - (P(C|D) - P(C''|D))$$

where R is the evaluated rule, C is the current class, C' is a class different from C that maximizes $P(C'|R)$ and C'' is a class different from C that maximizes $P(C''|D)$, where D is the default rule. Note that the most likely other class may not be the same for the investigated rule and the default rule and hence two different class labels, C' and C'', are used.

We may use this both as a pruning criterion (i.e., maximizing the margin increase) and as an exclusion criterion (i.e., excluding a rule whenever the margin increase is not positive). The rule in the previous example would be excluded by the latter criterion, in contrast to the likelihood ratio criterion, since the margin increase actually is negative for class A.

[9] Considering the ratio instead of difference when relating the likelihood of the current class and the most likely other class would give a rather different criterion, that would favor relative, rather than absolute, improvements.

[10] We have adopted the term *margin* from [16], in which it was used to explain the effectiveness of boosting.

4 Empirical Evaluation

In this section we present an empirical comparison of the novel criteria to the previously proposed when generating unordered rule sets with incremental reduced error pruning. We first describe the methods and methodology used together with the hypotheses to be tested, and then present the results.

4.1 Experimental Setting

Methods The methods that are to be compared are all variants of the U-IREP algorithm using different pruning and exclusion criteria. Besides the novel criteria, which can be used for both pruning and exclusion as explained in section 3.3, we also consider combinations of the pruning criteria of I-REP, I-REP-2 and IREP* (of which the two latter were shown to be equivalent) and the exclusion criteria used in I-REP and I-REP-2 (which also are equivalent). In addition to evaluating the previous criteria in the Bayesian framework, we also test these in conjunction with using a single rule with the highest precision to classify test examples, which is the method employed by RIPPER when inducing unordered rule sets.[11]

The employed methods are summarized in Table 1. U-IREP and the above criteria and inference methods were implemented and compared within the Rule Discovery System[12], which incorporates other rule induction algorithms as well and is capable of handling numerical features and missing values. We employ the default setting of using 2/3 of the training examples for growing rules, and 1/3 for pruning, and all methods are given the same grow and prune sets.

Table 1. Employed Criteria and Inference Methods for U-IREP

Acronym	Pruning criterion	Exclusion criterion	Inference method
LD	Likelihood ratio to default	≤ 1	Bayes'
MI	Margin increase	≤ 0	Bayes'
I	$p - n$	$p/(p+n) \leq 1/2$	Bayes'
R	$p/(p+n)$	$p/(p+n) \leq 1/2$	Bayes'
IS	$p - n$	$p/(p+n) \leq 1/2$	Single rule
RS	$p/(p+n)$	$p/(p+n) \leq 1/2$	Single rule

[11] The precision is estimated using Laplace correction: $\frac{p+1}{p+n+2}$

[12] Rule Discovery System (1.0), http://www.compumine.com, Compumine AB. A license for academic purposes may be obtained at no cost.

Methodology and data sets We decided to choose a large set of data sets and compare variants of U-IREP using ten-fold cross validation. The motivation for using cross-validation is to keep the number of test examples as high as possible (which in this case are all examples in the data set), in order to allow for detecting significant differences in accuracy for pairwise comparisons. For large data sets this may not be required, but was nevertheless used here. McNemar's test was used for investigating whether any observed difference in accuracy could be considered as due to chance (a p-value lower than 0.05 was here considered significant).

We selected 34 data sets from the UCI Repository [1], which all concern classification tasks, some of which containing more than 20 classes. The names of these data sets together with the number of classes are listed in the results section.

Test hypotheses For each pair of methods, the null hypothesis is that the probability of one method performing significantly better on a data set than another (i.e., has a higher accuracy and the difference in accuracy is statistically significant according to McNemar's test) equals the probability that the other method is significantly better than the first.

4.2 Experimental Results

In Table 2, we list the mean accuracies from ten-fold cross validation for all methods on all data sets. A suffix i for an accuracy indicates that the method performs significantly worse than the ith method (according to McNemar's test). In Table 3, we show the number of significant wins and losses for each pair of method, together with the p-value of obtaining that result if the null hypothesis holds (i.e., there is no difference between the methods).

From Table 3, it follows that all eight null hypotheses concerning the comparison of a novel and a previous method (i.e., saying that there is no difference between the novel and the previous method) can be rejected with high confidence (the highest p-value for the likelihood criterion when compared to previous methods is 0.0215, and for the margin-based it is 0.00195). Hence, there is strong evidence that in cases for which the difference in accuracy is unlikely due to chance, both novel criteria are expected to outperform any of the previous criteria. Considering all observed differences in accuracy (including those that not unlikely are due to chance variation), there is still strong evidence for rejecting the corresponding null hypotheses for the margin-based criterion (the highest p-value is 0.029), while the evidence for rejecting the corresponding null hypotheses for the likelihood ratio criterion is weaker (the highest p-value is 0.136).

Although the results point in favor of the margin-based criterion over the likelihood ratio criterion, the null hypothesis that relates the two novel methods can not be rejected. It should also be noted that the two novel criteria behave identically for all binary classification tasks, as pointed out in section 3.3.

Table 2. Accuracies on 34 UCI data sets.

Data set	**LD**	**MI**	**I**	**R**	**IS**	**RS**
audiology (24 cl.)	66.50	65.50	64.00	63.00	62.50	61.50_{1}
balance-scale (3 cl.)	83.52	84.00	85.44	84.16	78.88_{12346}	83.84
breast-cancer (2 cl.)	70.63	70.63	70.63	70.63	70.28	70.28
breast-cancer-wisconsin (2 cl.)	96.28	96.28	96.28	95.99	93.28_{1234}	93.71_{1234}
car (4 cl.)	93.11	92.53	81.31_{12}	82.87_{12}	77.20_{12346}	80.38_{124}
cleveland-heart-disease (5 cl.)	50.83	51.49	53.80	54.13	54.13	54.13
crx (2 cl.)	84.18	84.18	73.44_{12}	75.04_{12}	74.17_{12}	75.04_{12}
cylinder-bands (2 cl.)	66.85_{56}	66.85_{56}	66.67_{56}	66.67_{56}	68.15	68.15
dermatology (6 cl.)	87.98	88.25	88.80	88.80	89.07	89.07
ecoli (8 cl.)	76.49_{3}	76.19	79.17	76.79	75.00_{3}	74.70_{34}
glass (6 cl.)	59.35	59.35	56.54	58.41	56.54	58.41
hepatitis (2 cl.)	81.94	81.94	83.23	81.29	79.35	80.65
house-votes (2 cl.)	96.09	96.09	94.94	95.40	92.18_{12346}	94.25_{12}
image-segmentation (7 cl.)	73.81	75.71	75.71	75.71	74.76	75.24
ionosphere (2 cl.)	90.00	90.00	88.86	90.00	89.14	89.71
iris (3 cl.)	94.00	94.67	94.67	94.67	95.33	95.33
kr-vs-kp (2 cl.)	93.34	93.34	89.33_{12}	87.17_{1235}	89.24_{12}	87.20_{1235}
lung-cancer (3 cl.)	38.71	41.94	41.94	41.94	38.71	38.71
lymphography (4 cl.)	78.38	78.38	76.35	75.00	77.03	77.03
mushroom (2 cl.)	100.00	100.00	99.32_{1246}	100.00	98.63_{12346}	100.00
new-thyroid (3 cl.)	91.63	91.63	93.95	91.63	92.56	89.77_{35}
pima-indians-diabetes (2 cl.)	71.35	71.35	68.88	68.49_{12}	68.10_{123}	67.97_{12}
post-operative-patients (3 cl.)	68.89	68.89	68.89	70.00	68.89	70.00
primary-tumor (21 cl.)	42.77	41.89	39.82	38.94	39.53	38.64
promoters (2 cl.)	72.38	72.38	63.81	64.76	63.81	64.76
sick-euthyroid (2 cl.)	97.19	97.19	93.52_{12}	93.11_{12}	92.73_{123}	92.95_{123}
soybean-large (19 cl.)	81.76	82.41	79.48_{2}	78.83_{2}	76.55_{123}	77.52_{12}
spambase (2 cl.)	86.37	86.37	84.79_{12}	82.66_{1235}	84.76_{12}	82.59_{1235}
spectf (2 cl.)	84.24	84.24	76.79_{12}	76.79_{12}	76.22_{12}	76.22_{12}
splice (3 cl.)	73.72	73.50	61.77_{12}	60.68_{1235}	61.84_{12}	60.55_{1235}
tae (3 cl.)	35.33	35.33	36.67	37.33	36.00	36.67
tic-tac-toe (2 cl.)	97.39	97.39	83.30_{1246}	98.02	85.70_{1246}	98.33
wine (3 cl.)	92.13	92.13	92.13	92.13	91.57	91.57
yeast (10 cl.)	54.45	54.85	54.45	52.70_{25}	54.65	52.96

Table 3. Significant wins/losses and corresponding p-values for all pairs of methods.

	LD	MI	I	R	IS	RS
LD	-	0/0 (1.00)	9/1 (2.15e-02)	8/0 (7.81e-03)	14/1 (9.77e-04)	12/1 (3.42e-03)
MI	0/0 (1.00)	-	10/0 (1.95e-03)	10/0 (1.95e-03)	14/1 (9.77e-04)	11/1 (6.35e-03)
I	1/9 (2.15e-02)	0/10 (1.95e-03)	-	3/2 (1.00)	9/1 (2.15e-02)	7/3 (3.44e-01)
R	0/8 (7.81e-03)	0/10 (1.95e-03)	2/3 (1.00)	-	6/5 (1.00)	3/1 (6.25e-01)
IS	1/14 (9.77e-04)	1/14 (9.77e-04)	1/9 (2.15e-02)	5/6 (1.00)	-	4/5 (1.00)
RS	1/12 (3.42e-03)	1/11 (6.35e-03)	3/7 (3.44e-01)	1/3 (6.25e-01)	5/4 (1.00)	-

Table 4. Summary of all wins/losses and corresponding p-values for all pairs of methods.

	LD	MI	I	R	IS	RS
LD	-	5/8 (5.81e-01)	19/10 (1.36e-01)	19/10 (1.36e-01)	24/8 (7.00e-03)	23/9 (2.01e-02)
MI	8/5 (5.81e-01)	-	20/7 (1.92e-02)	19/7 (2.90e-02)	27/6 (3.24e-04)	26/7 (1.32e-03)
I	10/19 (1.36e-01)	7/20 (1.92e-02)	-	15/11 (5.57e-01)	21/10 (7.08e-02)	21/12 (1.63e-01)
R	10/19 (1.36e-01)	7/19 (2.90e-02)	11/15 (5.57e-01)	-	23/10 (3.51e-02)	21/7 (1.25e-02)
IS	8/24 (7.00e-03)	6/27 (3.24e-04)	10/21 (7.08e-02)	10/23 (3.51e-02)	-	9/16 (2.30e-01)
RS	9/23 (2.01e-02)	7/26 (1.32e-03)	12/21 (1.63e-01)	7/21 (1.25e-02)	16/9 (2.30e-01)	-

5 Concluding Remarks

A formulation of incremental reduced error pruning for unordered sets of rules has been presented. We have investigated the usefulness of previous pruning and exclusion criteria for ordered rule sets in this novel setting, where a Bayesian framework for combining predictions from multiple rules has been assumed. We have pointed out some potential draw-backs of previous criteria that may cause possibly beneficial rules to be exluded as well as harmful rules to be included. Motivated by the Bayesian framework, two novel criteria for both pruning and exclusion have been introduced, one based on the likelihood ratio and the other on the margin. An empirical evaluation on 34 datasets from the UCI repository shows that both novel criteria significantly outperform previous criteria developed for ordered rule sets, and that the margin-based approach is slightly ahead of the likelihood ratio criterion.

There are a number of possible directions for further research that deserve attention. As has been shown for ordered rule sets (e.g., by IREP* and RIPPER), post-processing of generated rules (e.g., to remove rules based on some global criterion) may significantly improve accuracy (although, as pointed out in [8], existing methods to do this for ordered rule sets are both complex and heuristic). Related to this is the question of how to adapt the MDL-framework of IREP* and RIPPER to unordered rules.

Acknowledgements

Thanks to the anonymous reviewers for many useful suggestions. Thanks also to the Ljubljana Oncology Institute, Slovenia for providing the breast cancer, lymphography and primary tumor data sets.

References

1. C.L. Blake and C.J. Merz. UCI repository of machine learning databases, 1998.
2. H. Boström and P. Idestam-Almquist. Induction of logic programs by example-guided unfolding. *Journal of Logic Programming*, 40(2-3):159–183, 1999.
3. B. Cestnik. Estimating probabilities: A crucial task in machine learning. In *Proc. of the 9th European Conference on Artifical Intelligence*, volume 2157, pages 147–149. Pitman, 1990.
4. P. Clark and R. Boswell. Rule induction with CN2: Some recent improvements. In *Proc. Fifth European Working Session on Learning*, pages 151–163, Berlin, 1991. Springer.
5. W.W. Cohen. Efficient pruning methods for separate-and-conquer rule learning systems. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, pages 998–994. Morgan Kaufmann, 1993.
6. W.W. Cohen. Fast effective rule induction. In *Proc. of the 12th International Conference on Machine Learning*, pages 115–123. Morgan Kaufmann, 1995.
7. M. Eineborg and H. Boström. Classifying uncovered examples by rule stretching. In *Proc. of Eleventh International Conference on Inductive Logic Programming*, volume 2157 of *LNAI*, pages 41–50. Springer, 2001.
8. E. Frank and I. Witten. Generating accurate rule sets without global optimization. In *Proc. 15th International Conf. on Machine Learning*, pages 144–151. Morgan Kaufmann, San Francisco, CA, 1998.
9. J. Fürnkranz. Pruning algorithms for rule learning. *Machine Learning*, 27(2):139–171, May 1997.
10. J. Fürnkranz. Separate-and-conquer rule learning. *Artificial Intelligence Review*, 13(1):3–54, 1999.
11. J. Fürnkranz and P. Flach. An analysis of rule evaluation metrics. In *Proc. 20th International Conference on Machine Learning (ICML'03)*, pages 202–209. AAAI Press, January 2003.
12. J. Fürnkranz and G. Widmer. Incremental reduced error pruning. In W.W. Cohen and H. Hirsh, editors, *Proceedings of the 11th International Conference on Machine Learning*, pages 70–77. Morgan Kaufmann, 1994.
13. N. Lavrac and S. Dzeroski. *Inductive Logic Programming: Techniques and Applications.* Ellis Horwood, 1994.

14. T. Lindgren and H. Boström. Classification with intersecting rules. In *Proceedings of the 13th International Conference on Algorithmic Learning Theory (ALT'02)*, pages 395–402. Springer-Verlag, 2002.
15. T. Lindgren and H. Boström. Resolving rule conflicts with double induction. In *Proc. of the 5th International Symposium on Intelligent Data Analysis*, pages 60–67. Springer, 2003.
16. P. Bartlett R. Schapire, Y. Freund and W. Sun Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. *The Annals of Statistics*, 26(5):1651–1686, 1998.
17. R. Rivest. Learning decision lists. *Machine Learning, 2(3), 229-246*, 1987.
18. I. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, 2000.