Using the *m*-estimate in rule induction

Sašo Džeroski, Bojan Cestnik, and Igor Petrovski

Artificial Intelligence Laboratory, "Jožef Stefan" Institute, Ljubljana, Slovenia

Rule induction, a subarea of machine learning, is concerned with the problem of constructing rules from examples. In rule induction systems, various heuristic functions are used to estimate the quality of rules. Most of them use some form of probability estimates, relative frequency being the most common. This has resulted in the problem of small disjuncts, where specific rules produce high error rates, due to unreliable probability estimates from small samples. To alleviate this problem, the Laplace estimate has been used in the rule induction system CN2. We have replaced the Laplace estimate by a general Bayesian probability estimate, the *m*-estimate, which does not rely on the Laplacian assumption of equally likely classes. The parameter *m* in the *m*-estimate allows for adapting to the learning domain. Depending on the level of noise in the examples and other properties of the domain, the appropriate level of generalization can be achieved by setting the *m* parameter to an appropriate value. We compare the performance of rules derived by using the Laplace and the *m*-estimate on several practical domains in terms of classification accuracy and the theoretically underpinned measure of relative information score.

Introduction

One of the most common formalisms used to represent knowledge in expert systems is the formalism of if-then rules. Rule-induction systems are concerned with the automatic synthesis of if-then rules from a given set of examples with known classifications. An example is described by the values of a fixed collection of features, called attributes.

For illustration, consider the following rule, derived by the CN2 rule-induction system (Clark and Boswell 91) from the examples in the 1984 US Congressional Voting Records Database. This database includes votes for each of the U.S. House of Representatives congressmen on the 16 key votes in 1984. Given the votes (attributes), the task is to predict the party affiliation (class) of a congressman, which may be either republican or democrat.

```
IF      adoption_of_the_budget_resolution = y
  AND   physician_fee_freeze = n
  AND   education_spending = n
THEN    class = democrat
```

The above rule predicts that a congressman is a democrat if he voted for the adoption_of_the_budget_resolution and against the physician_fee_freeze and education_spending. Almost two hundred congressmen from the database are correctly classified as democrats by the rule.

The CN2 system (Clark and Niblett 89, Clark and Boswell 91) uses the covering approach to construct a set of rules for each possible class: it constructs a rule that correctly classifies some examples, removes the correctly classified examples from the training set and repeats the process until no more examples remain. To construct a rule that classifies examples in a given class, CN2 starts with a rule with an empty antecedent (if part) and the selected class as a consequent (then part). The antecedent of this rule is satisfied by all examples in the training set, and not only those of the selected class. CN2 then progressively refines the antecedent by adding conditions to it, until only examples of the selected class satisfy the antecedent. To allow for handling imperfect data, CN2 may construct a set of rules which is slightly imprecise, i.e., does not classify all examples in the training set correctly.

Various heuristics are used in rule-induction to direct search through the space of possible rules.

These heuristics attempt to measure the quality of a rule. At each step of the covering algorithm, the rule with the highest heuristic value (the best rule) is added to the current set of rules. Possible metrics of rule quality are accuracy on the training examples, used in AQ15 (Michalski et al. 86) and entropy, used in CN2 (Clark and Niblett 89). Both perform very similarly in the sense that they prefer rules which cover examples of only one class.

The problem with these metrics is that they tend to select very specific rules covering only a few examples (such rules are also referred to as small disjuncts). Namely, the likelihood of finding rules with high apparent accuracy on the training data increases as the rules become more specific. In the extreme case, a maximally specific rule will cover one example and hence have an unbeatable score using the metrics of apparent accuracy (scores 100 % accuracy) or entropy (scores 0.00, a perfect score). Apparent accuracy on the training data, however, does not adequately reflect true predictive accuracy, i.e., accuracy on new testing data. It has been shown (Holte et al. 89) that small disjuncts have very high error rates on new testing data.

There is a simple explanation for this phenomenon. The accuracy of a rule on the testing set, which can also be interpreted as the probability of correctly classifying an example from the testing set, is predicted from its performance on the training set by using relative frequency. The relative frequency of correct classifications converges to the true probability of correct classifications only when the rule covers many examples. When this is not the case, the probability estimate of correct classification becomes unreliable: the less examples the rule covers the more unreliable its accuracy estimated by relative frequency. To alleviate this problem, the Laplace probability estimate was used as a search heuristic in CN2 (Clark and Boswell 91). As the Laplace estimate is more reliable than relative frequency when estimating probabilities from small samples, significant improvements in the performance of CN2 were reported.

Let us illustrate this problem by a simple example. Consider a domain with two classes C_1 and C_2, and consider two rules R_1 and R_2, R_1 covering 1000 examples of class C_1 and 1 example of C_2 (we say that R_1 has coverage (1000,1)) and R_2 covering a single example of class C_1. The accuracies of R_1 and R_2, estimated by relative frequency, are 99.9 % and 100 %, respectively, and thus R_2 would be selected. If the Laplace estimate is used, however, the estimates are 99.8 % and 66.7 % and R_1 is by far better than R_2.

In short, the metrics of apparent accuracy /entropy have an undesirable ždownward bias', i.e. preference for rules low down in the general to specific search space, which the Laplace estimate successfully avoids (Clark and Boswell 91). The Laplace estimate was also used in tree-pruning (Niblett and Bratko 86) to estimate the static error in a node (the error expected if the node were turned to a leaf, i.e. its subtrees were pruned). This estimate, however, relies on the assumption that the prior probability of each class is uniformly distributed. This assumption implies that all classes are *a priori* equally likely, which is rarely true in practice and was reported to cause serious problems to the Niblett-Bratko method of tree-pruning (Cestnik et al. 87, Mingers 89).

We recently investigated the use of a general Bayesian probability estimate in machine learning (Cestnik 90,91). The *m*-estimate, of which the Laplace estimate is a special case, was successfully used in the 'naive' Bayesian classifier (Cestnik 90) and in tree-pruning (Cestnik and Bratko 91). This estimate takes into account the prior (unconditional) probabilities of the classes. It also has a tunable parameter m, which allows for adapting to the properties of a given domain, such as the level of noise in examples.

In the paper, we describe the use of the m probability estimate in rule induction. We used it as a search heuristic in CN2 and appropriately tuned the m parameter by trying several different values for it. The improvement achieved was especially evident in the domains where the assumption of equally likely classes is severely violated. The performance was measured in terms of classification accuracy and in terms of the relative information score (Kononenko and Bratko 91) of the classifications.

The relative information score measure takes into account the difficulty of the classification problem. Namely, in domains where one of the classes is highly likely, it is trivial to achieve high classification accuracy. The completely uninformed classifier that classifies into the majority

class would in that case have an undeservedly high score according to classification accuracy.

A general Bayesian probability estimate

To obtain a general Bayesian estimate of probability p on the basis of some evidence E, an *initial* (prior) probability distribution for p has to be assumed first. Using evidence E the initial distribution is transformed to a *final* one, from which the expectation can be taken as a point estimate for p. The first task, namely the assumption of an initial probability distribution, is regarded as a basic difficulty of the Bayesian procedure and, consequently, represents a major obstacle to its wider application (Cheeseman 88).

In Bayesian analysis (Berger 85) the initial probability distribution is usually taken from the class of beta probability distributions. In this case, the initial distribution is completely specified by its initial expectation and variance.

In (Cestnik 90, 91) a probability estimate of conditional probabilities, called the *m*-estimate, was developed. The basic idea behind this estimate is that the initial expectation can be estimated from an unconditional sample. Therefore, only one parameter, the variance, remains to be specified. The *m*-estimate has the following form:

$$p = \frac{r+mp_a}{n+m}$$

where r is the number of positive examples, n is the total number of examples observed, p_a is the initial expectation (prior probability of an example being positive) and m is a parameter of the estimation method. The parameter m is related to the initial variance by the following formula:

$$\mathrm{Var}(p) = \frac{p_a(1-p_a)}{m+1}$$

The parameter m has several interpretations. First, as can be seen from the above formula, it is inversely proportional to the initial variance. In other words, the higher the value of m, the lower the initial variance; this means that we are very confident in the initial expectation of p, i.e. the prior probability p_a. Second, it controls the balance between relative frequency and prior probability, as can be observed from the following form of *m*-estimate:

$$p = \frac{n}{n+m} \times \frac{r}{n} + \frac{m}{n+m} \times p_a$$

Finally, m can be set to correspond to the level of noise in data (Cestnik and Bratko 91). When more noise is expected in the examples, a higher value of m should be used. In summary, as the value of m increases, the prior probability plays a more important role; as a result, the examples are considered less trustworthy.

How should the actual value of parameter m be determined? On one hand, it can be set subjectively by the domain expert. For example, let us consider two rules taken from a simple chess endgame domain with synthetical noise (Džeroski and Lavrač 91). The first rule covers (8,0) and the second (26,4) examples. The prior probabilities of the classes are 1/3 and 2/3, respectively. Suppose that after examining each rule, we strongly prefer the second rule to the first one. Accordingly, we would like to set the value of m in the *m*-estimate so that the estimated accuracy of the second rule is higher then that of the first one. After a simple manipulation of the corresponding formulas we obtain m >3.

On the other hand, the expert might not be available. In that case, which also covers all the experiments in this paper, we propose that several different values for m should be applied. At the end, after measuring the performance of the induced classifiers, the value of m which gives the best performance can be selected. Such a procedure in fact corresponds to a series of Bayesian procedures in which we take different initial probability distribution each time. If we have a criterion to measure performance, which is often the case, we can evaluate the quality of each prior distribution and select the best one.

Quinlan (1991) presents an improved probability estimate for small disjuncts which is very similar to the *m*-estimate. However, it is introduced as an ad-hoc modification of the Laplace estimate, rather than derived directly by a Bayesian estimation procedure. In addition, Quinlan proposes that the following value of m should be used:

$$m = \frac{1}{1-p_a}$$

for which no theoretical justification is given. In fact, in our experiments we found that the best value for m depends mostly on domain charac-

teristics such as noise, and not on the prior probabilities as proposed by Quinlan.

Experimental Comparison

Experimental Method

Experiments were performed to measure the improvement in predictive accuracy resulting from the use of the *m*-estimate as a search heuristic instead of the Laplace estimate. As tests on a single domain are not sufficient to draw reliable conclusions about the relative performance of algorithms, experiments on twelve domains, shown in Table 1, were conducted. We used the same data as Clark and Boswell (91) and also largely followed their experimental design.

Table 1: Details of Experimental Domains

Domain†	Description	Number of		
		Exs	Atts	Cls
lymphography	disease diagnosis	148	18	4
pole-and-cart	predict human balancing action from exs	1044	4	2
soybean	disease diagnosis	307	35	19
heart-diseaseC	disease diagnosis (data from Cleveland)	303	13	2
heart-diseaseH	disease diagnosis (data from Hungary)	294	13	2
glass	predict glass type from chem. content	194	7	9
primary-tumor	predict tumor type	330	17	15
voting-records	predict democrat/republican from votes	435	16	2
thyroid	disease diagnosis	1960	29	3
breast-cancer	predict if recurrence is likely	286	9	2
hepatitis	predict if survival likely	157	19	2
echocardio	predict if survival from heart problem likely	131	7	2

CN2 using the Laplace estimate and CN2 using the *m*-estimate were compared. In each domain, the data were split into 2/3 for training and 1/3 for testing. Twenty different splits were used, and both CN2 with Laplace and CN2 with *m*-estimate were run on the same splits. The results were averaged over the 20 runs. In CN2, the default star size of 5 was used and significance testing was switched off. Unordered rules were generated, which are much easier to interpret and were found to achieve better performance (Clark and Boswell 91) than ordered rules.

Besides classification accuracy, we also measured the size of the rule sets produced (as the total number of attribute tests appearing in the rules). In addition, the relative information score of the classifications was computed. Below we briefly describe this measure, introduced by Kononenko and Bratko (91).

The most general form of the answer of a classifier, given a testing example, is a probability distribution over the classes of the domain. Let the correct class of example e_k be C, its prior probability $P(C)$ and the probability returned by the classifier $P'(C)$. The information score of this answer is

$$I(e_k)=\begin{cases}-\log P(C)+\log P'(C) & P'(C)\geq P(C)\\ \log(1-P(C))-\log(1-P'(C)) & P'(C)<P(C)\end{cases}$$

As $I(e_k)$ indicates the amount of information about the correct classification of e_k gained by the classifier's answer, it is positive if $P'(C) > P(C)$, negative if the answer is misleading ($P'(C)>P(C)$) and zero if $P'(C)=P(C)$.

The *relative information score* I_r of the answers of a classifier on a testing set consisting of examples $e_1, e_2, ..., e_n$ belonging to one of classes $C_1, C_2, ..., C_N$ can be calculated as the ratio of the *average information score of the answers* and the *entropy of the prior distribution of classes*.

$$I_r=\frac{\frac{1}{n}\times\sum_{k=1}^{n} I(e_k)}{-\sum_{i=1}^{N} P(C_i)\times\log P(C_i)}$$

The relative information score of the 'default' classifier, which always returns the prior probability distribution, is zero. If the classifier always guesses the right class and is absolutely sure about it ($P'(C)=1$), then $I_r=1$, provided the class distributions of the training and testing sets are the same.

It was easy to modify the answer given by CN2 to a probability distribution. Namely, when unordered rules are produced, several rules may apply to a single example. In that case, the distributions of examples covered by each of the applied rules are summed, and the example is classified into

the majority class (Clark and Boswell 91). For instance, if two rules with coverage (10,2) and (4,40) apply, the 'summed' distribution would be (14,42) and the example is classified in C_2. To get the probability distribution over classes (C_1,C_2) returned by CN2, we divide the distribution with the total number of examples covered (56) and obtain as answer the probability distribution (0.25,0.75).

The prior probabilities of the classes, used both in the *m*-estimate during the search for rules and in the calculation of the relative information score were estimated by relative frequency from the whole training set. As the number of examples in the entire set is usually high enough, this estimate is sufficiently reliable.

Experimental Results

For each domain, 15 different values of *m* were applied (0, 0.01, 0.5, 1, 2, 3, 4, 8, 16, 32, 64, 128, 999). Since we regard the role of *m* as being of qualitative nature, the values for *m* were chosen from a quasi-logarithmic scale.

The best accuracy obtained for a fixed value of *m* was selected. These accuracies are listed in Table 2 together with the results obtained when using the Laplace estimate. The relative information score obtained for the same value of *m*, as well as the one for the Laplace estimate, are given in Table 3. The size of the corresponding rule sets, i.e. the total number of attribute tests (literals) appearing in the rules, is given in Table 4.

Comparative accuracies

Table 2 gives the accuracies obtained for all of the above domains. To make an overall comparison between the heuristics, a paired, one-tailed t-test was used, whose results are shown in the smaller table at the bottom. From this t-test, it can be seen that using the *m*-estimate with the appropriate value of *m* significantly (>99 %) improves CN2's accuracy.

Note that although the overall improvement in accuracy is slight (just 1.35 %), substantial improvements have been achieved in several domains, such as echocardio and primary tumor. It is worth noting that the Laplacian assumption of equally likely classes is violated in these domains. Furthermore, even the slight improvements in some domains (breast-cancer) have meant improvement from 'worse-than-default' to 'better-than-default' performance. Thus, CN2 with the *m*-estimate (and the best value of *m*) achieves accuracies that are better than those achieved by the uninformed 'default' classifier in all domains. In other words, no domain is 'worse-than-default' in this case.

Table 2: Percentage Accuracies for Different Heuristics.

The table shows percentage accuracies for CN2 with the Laplace estimate and with the *m*-estimate. The third column gives the value of *m* for which best results were achieved (best value of *m*), while the second column gives the accuracies achieved for these values of *m*. The lower table gives an overall comparison of accuracies using paired, one-tailed t-test on the data from the upper table.

Domain	Algorithm			
	CN2			Default
	Laplace	Best m		
		Value	Accuracy	
lymphography	79.81	2	83.16	54.59
pole-and-cart	69.58	0.5	71.80	48.17
soybean	82.40	32	81.47	10.39
heart-diseaseC	78.02	0.5	78.61	52.28
heart-diseaseH	79.13	128	79.90	64.02
glass	65.15	32	64.29	35.94
primary-tumor	41.94	0.01	45.16	23.54
voting-records	95.35	64	96.28	62.31
thyroid	95.52	32	96.84	95.29
breast-cancer	71.42	0.5	72.85	72.05
hepatitis	80.40	0	80.79	78.14
echocardio	66.96	0	71.15	67.79

Algorithms Compared	Mean Improvement (Mean X - Y)	Significance of Improvement
CN2 (Laplace) - CN2 (Best m)	-1.38 %	99.38 %

Comparative Relative Information Scores

Table 3 gives the relative information scores for all of the above domains. Again, using the *m*-estimate with the appropriate value of *m* significantly (>94 %) improves CN2's relative information score. The overall improvement is slight, but substantial improvement has been achieved in domains where the Laplacian assumption is violated (primary tumor, lymphography).

Table 3: Relative Information Scores for Different Heuristics.

The table shows relative information scores for CN2 with the Laplace estimate and with the *m*-estimate using the same value of *m* as in Table 2. Note that for some domains other values of *m* give better relative information scores. An overall comparison using paired, one-tailed t-test on the data from the upper table is given in the lower table.

Domain	Heuristic	
	Laplace	Best m
lymphography	62.05	64.30
pole-and-cart	39.70	44.45
soybean	84.95	83.85
heart-diseaseC	53.40	55.15
heart-diseaseH	52.05	46.70
glass	50.70	49.75
primary-tumor	35.35	38.75
voting-records	88.60	88.25
thyroid	25.80	46.60
breast-cancer	18.95	22.85
hepatitis	31.80	32.70
echocardio	15.75	24.80

Algorithms Compared	Mean Improvement (Mean X - Y)	Significance of Improvement
CN2 (Laplace) - CN2 (Best m)	-3.25 %	94.26 %

Comparative sizes of rule sets

Table 4 gives the rule set sizes for all domains. An overall comparison reveals that the rules produced by using the *m*-estimate are significantly (>91 %) more specific. In most of the domains, better accuracies were achieved with the *m*-estimate when the rules were more complex (specific). When estimating probabilities from few examples more accurately, CN2 with the *m*--estimate has been able to select better rules which, although more specific, had still good predictive accuracy.

In fact one might argue that these rules are exactly at the right level of specificity for the given domains, determined by the appropriate value of *m*. It can be immediately noticed, namely, that the size of the rule set monotonically decreases with increasing *m* (see Figure 2). Similarly, the average number of examples covered by a rule increases accordingly. Thus, *m* controls the level of fitting/generalization to be performed by CN2.

In two of the domains (voting-records and heart-diseaseH) better results were achieved with smaller (more general) rule sets. Slight decrease in classification accuracy was recorded only in the soybean and glass domains. This only means, however, that the appropriate value of *m* was not among the ones chosen for our experiments. Repeating the experiments for the soybean domain with a finer resolution for *m* (15, 16, ..., 25), we obtained both better classification accuracy and relative information score for $m = 23$.

Table 4: Sizes of Rule Sets for Different Heuristics.

The table shows rule set sizes as total number of literals (attribute tests) for CN2 with the Laplace estimate and CN2 with the *m*-estimate using the same value of *m* as in Table 2. An overall comparison using paired, one-tailed t-test on the data from the upper table is given in the lower table

Domain	Heuristic	
	Laplace	Best m
lymphography	37.45	39.80
pole-and-cart	123.10	198.45
soybean	110.25	107.55
heart-diseaseC	55.95	70.15
heart-diseaseH	59.25	30.30
glass	49.50	42.05
primary-tumor	305.55	549.95
voting-records	59.70	24.75
thyroid	82.25	90.65
breast-cancer	84.15	110.75
hepatitis	42.60	75.65
echocardio	46.40	100.40

Algorithms Compared	Mean Improvement (Mean X - Y)	Significance of Improvement
CN2 (Laplace) - CN2 (Best m)	-32.02 %	91.90 %

Closer look at selected domains

To illustrate the performance of CN2 with the various Bayesian estimates we give the results for all values of *m*, as well as for the Laplace estimate for the domains echocardio, lymphography, primary tumor and thyroid. For all of them the Laplacian assumption of equally likely classes is violated. The appropriate values for *m* are 0, 2, 0.01 and 32, respectively.

The results, given in Figure 1, include the classification accuracy and the relative information score, which takes into account the 'informativeness' of the answers of a classifier. We give the rule set sizes for the same domains in Figure 2. As the generality (coverage) of rules increases

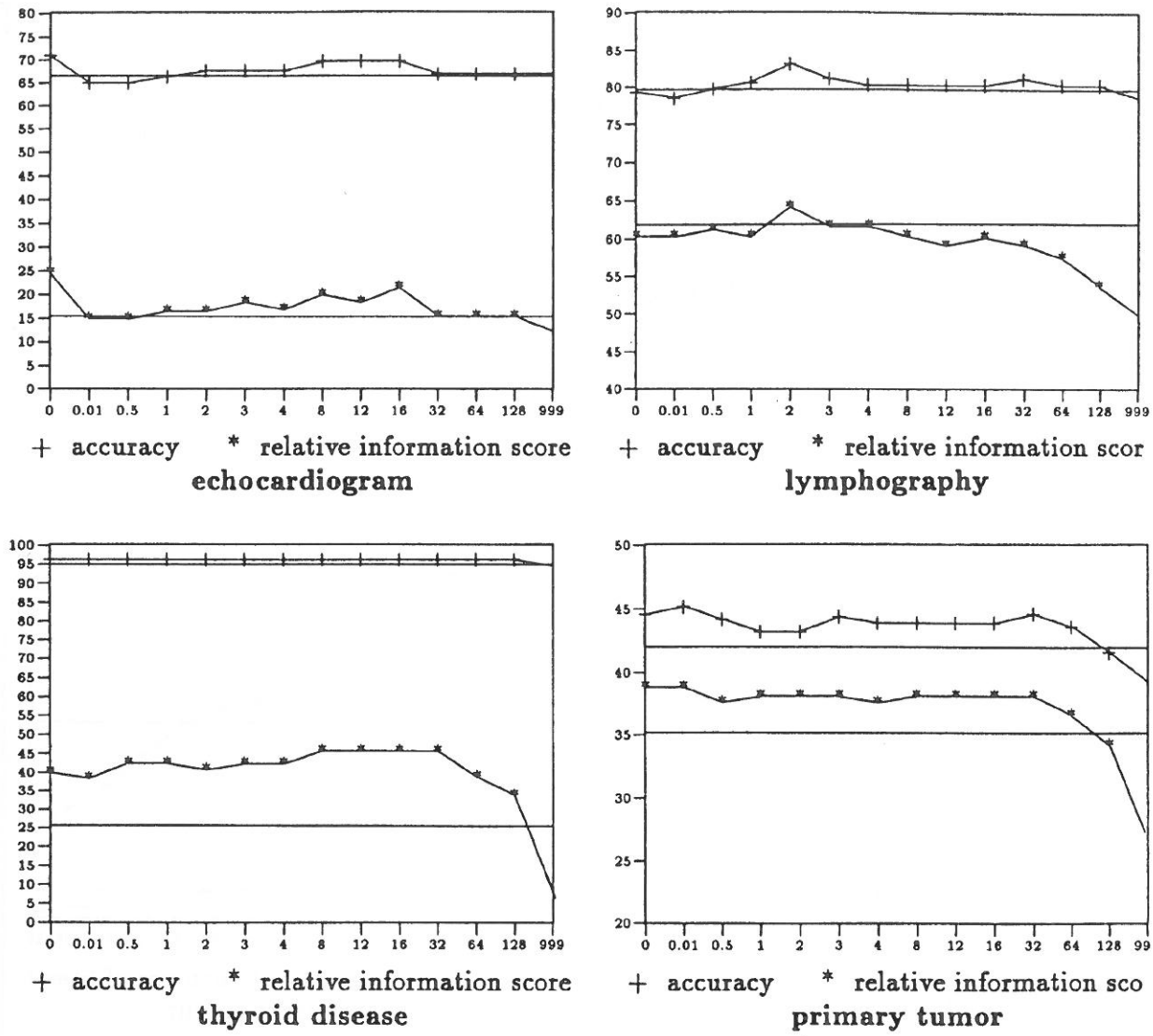

Figure 1: Accuracy and Relative Information Score for Different Heuristics.
The value of m used in CN2 is depicted on the x-axis, while the y-axis gives the classification accuracy and the corresponding relative information score. The straight lines denote the performance of CN2 using the Laplace estimate.

when the rule set size decreases, we have not included the corresponding graphs.

In the lymphography domain, the best performance is obtained with $m=2$. Both accuracy and relative information score clearly reach their peaks at this value of m, which is obviously the one appropriate for this domain. This is achieved by a rule set which is only slightly more complex than the one obtained with Laplace.

In the thyroid domain, the classification accuracy achieved with different values of m is much the same as the one obtained with Laplace. The relative information score is, however, substantially higher for almost all values of m, as the Laplace assumption is severely violated in this domain. (Similar statement holds for relative information score in the primary tumor domain.) The peak performance is achieved with $m=32$ and a slightly larger rule set than the one produced with Laplace.

For domains echocardio and primary tumor, the situation is somewhat different. Although the best performance is achieved with small values of m, the induced rules are much more complex as compared to the ones induced with Laplace. However, other values of m exist (16 and 32,

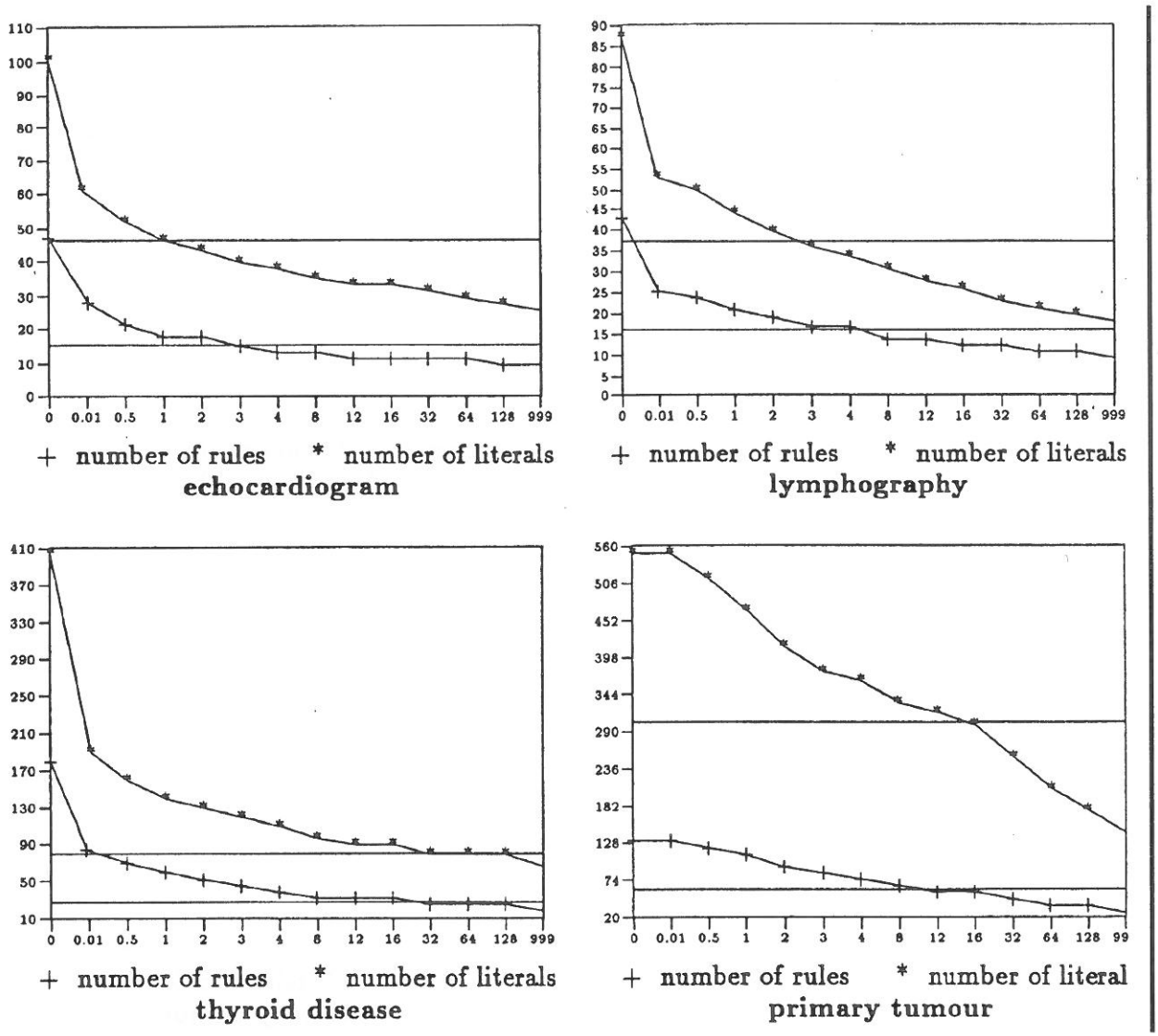

Figure 2: Sizes of Rule Sets for Different Heuristics.
The value of m used in CN2 is depicted on the x-axis, while the y-axis gives the number of rules induced and the corresponding total number of attribute tests (literals). The straight lines denote the performance of CN2 using the Laplace estimate.

respectively) for which almost equal performance can be achieved with much more compact rule sets. We can see that different values of m provide us with different models of the domain, at various levels of generality, among which we can select one according to the measures described above.

Conclusion

In our approach, the problem of small disjuncts is solved by estimating their accuracy in a more reliable way. Rather than using different forms of bias for disjuncts of different size (as proposed by Holte et al. (89)), we estimate the accuracy of all disjuncts by the same Bayesian procedure. The general Bayesian m probability estimate turned out to be a successful mechanism in rule induction, allowing to set the appropriate level of generalization for the domain (examples) at hand. It does not make the assumption used in the Laplace estimate, namely that all classes are equally likely, thus enabling much better performance in domains where this assumption is violated.

It is generally agreed that too long, specific rules are likely to perform worse on new, previously

unseen data. On the other hand, too general rules are also likely to have low predictive power. By using the appropriate value of the parameter *m* in the *m*-estimate we actually determine the appropriate level of generalization suitable for the given domain. This has allowed CN2 with the *m*-estimate to perform better, although the induced rule sets are on the average larger and more specific than the one produced with the Laplace estimate. We have also seen that rules produced by using the *m*-estimate produce higher relative information scores, the latter being theoretically underpinned as a much more appropriate measure of classifier's performance than accuracy.

One might argue that to achieve better performance we have to pay the price of determining the appropriate value of *m*. Our message is that determining the right value of *m* is worth the effort. In fact, using a spectrum of values for *m*, we can produce a spectrum of rules (models) of different generality for the domain at hand. Whereas a series of decision trees of increasing generality can be produced by sequential pruning (Breiman et al. 84, Cestnik and Bratko 91), a series of increasingly more general rule sets can be produced by a series of increasing values of *m*. We could then select among this models on the basis of performance criteria, such as accuracy and relative information score, or on the basis of comprehensibility (transparency) to the end user. Further work will address the problem of automatic tuning of the parameter *m* using data from the training set (possibly by a cross-validation procedure) to maximize performance according to given criteria.

Acknowledgements

This research was funded by the Slovenian Ministry of Science and Technology. The paper was written while Sašo Džeroski was visiting the Turing Institute Limited, Glasgow, Scotland, and was supported by a British Council scholarship. Thanks to R. Boswell for supplying the source code of CN2, as well as the experimental data. We are grateful to G. Klanjšček, M. Soklič and M. Zwitter of the University Medical Center, Ljubljana for the use of the lymphography, breast-cancer and primary-tumor data sets and to I. Kononenko for their conversion to a form suitable for the induction algorithms. Thanks also to D. Aha (UCI) for the compilation and use of the UCI Repository of Machine Learning Databases. Finally, thanks are due to Ivan Bratko and Igor Kononenko for their comments on the paper.

References

L. BREIMAN, J.H FRIEDMAN, R.A. OLSHEN, C.J. STONE. (1984) *Classification and regression trees*. Belmont, Wadsworth.

J.O. BERGER (1985) *Statistical Decision Theory and Bayesian Analysis*. Springer, New York.

B. CESTNIK (1990) Estimating probabilities: A crucial task in machine learning. In *Proc. European Conference on Artificial Intelligence, ECAI 90*, Stockholm, Sweden.

B. CESTNIK (1991) Estimating probabilities in machine learning. Ph.D. thesis, Faculty of electrical engineering and computer science, University of Ljubljana, Slovenia. In Slovenian.

B. CESTNIK, I. BRATKO (1991) On estimating probabilities in tree pruning. In *Proc. Fifth European Working Session on Learning, EWSL 91*, Porto, Portugal.

B. CESTNIK, I. KONONENKO, I. BRATKO (1987) ASSISTANT 86: A knowledge elicitation tool for sophisticated users. In *Progress in machine learning* (I. BRATKO, N. LAVRAČ, Eds.), pp. 31-45. Wilmslow, Sigma Press.

P. CHEESEMAN (1988) An inquiry in computer understanding. *Computational Intelligence*, **4**, 55-66.

P. CLARK, R. BOSWELL (1991) Rule induction with CN2: some recent improvements. In *Proc. Fifth European Working Session of Learning, EWSL 91*, Porto, Portugal.

P. CLARK, T. NIBLETT (1989) The CN2 induction algorithm. *Machine Learning*, **3** (4), 261-284.

S. DŽEROSKI, N. LAVRAČ (1991) Learning relations from noisy examples: An empirical comparison of LINUS and FOIL. In *Proc. Eighth International Workshop on Machine Learning*, Evanston, IL.

R. HOLTE, L. ACKER, B. PORTER, (1989) Concept learning and the problem of small disjuncts. In *Proc. Tenth International Joint Conference on Artificial Intelligence, IJCAI 89*, Detroit, MI.

I. KONONENKO, I. BRATKO (1991) Information-based evaluation criterion for classifier's performance. *Machine Learning* **6** (1), 67-80.

R.S. MICHALSKI, I. MOZETIČ, J. HONG, N. LAVRAČ (1986) The multi-purpose incremental learning system AQ15 and its testing application to three medical domains. In *Proc. Fifth National Conference on Artificial Intelligence, AAAI-86*, Philadelphia, PA.

J. MINGERS (1989) An experimental Comparison of pruning methods for decision tree induction. *Machine Learning*, **4** (2), 227-243.

T. NIBLETT, I. BRATKO (1986) Learning decision rules in noisy domains. In *Proc. Expert Systems Conference*, Brighton, UK.

J.R. QUINLAN (1991) Improved estimates for the accuracy of small disjuncts. Machine Learning **6** (1), 93-98.

Received: October 2, 1992
Accepted: April 6, 1993

Address for correspondence:
Sašo Džeroski
Artificial Intelligence Laboratory,
Institut Jozef Stefan
Jamova 39,
61111 Ljubljana,
Slovenia
e-mail: saso.dzeroski@ijs.si

Sašo Džeroski holds a BSc and a MSc degree in computer science from the Faculty of Electrical Angineering and Computer Science, University of Ljubljana, Slovenia. He is currently a research assistant at the Artificial Intelligence Laboratory, Jožef Stefan Institute, Ljubljana, Slovenia. His main research interest is in the areas of inductive logic programming, machine learning and qualitative modelling and control.

Bojan Cestnik holds a BSc, a MSc and a PhD degree in computer science from the Faculty of Electrical Angineering and Computer Science, University of Ljubljana, Slovenia. He is currently a research associate at the Artificial Intelligence Laboratory, Jožef Stefan Institute, Ljubljana, Slovenia. His main research interest is in the areas of machine learning, genetic algorithms and their integration.

Igor Petrovski is an undergraduate student of computer science at the Faculty of Electrical Angineering and Computer Science, University of Ljubljana, Slovenia. His research interest is mainly in the area of machine learning.