*Published in Proceedings of Information, Statistics and Induction in Science, Melbourne, World Scientific, 1996, pp. 20-30.*

# A heuristic covering algorithm has higher predictive accuracy than learning all rules

**Geoffrey I. Webb** webb@deakin.edu.au
School of Computing and Mathematics
Deakin University ph: +61-52-272606
Geelong, Vic. 3217, Australia fax: +61-52-272028

**Abstract:** *The induction of classification rules has been dominated by a single generic technique—the covering algorithm. This approach employs a simple hill-climbing search to learn sets of rules. Such search is subject to numerous widely known deficiencies. Further, there is a growing body of evidence that learning redundant sets of rules can improve predictive accuracy. The ultimate end-point of a move toward learning redundant rule sets would appear to be to learn and employ all possible rules. This paper presents a learning system that does this. An empirical investigation shows that, while the approach often achieves higher predictive accuracy than a covering algorithm, the covering algorithm outperforms induction of all rules significantly more frequently. Preliminary analysis suggests that learning all rules performs well when the training set clearly defines the decision surfaces but that the heuristic covering algorithm performs better when the decision surfaces are not clearly delineated by the training examples.*

**Keywords:** Classification Learning; Covering Algorithm; Learning Bias.

**Area of Interest:** Concept Formation and Classification.

## 1 Introduction

The induction of classification rules is dominated by variations of the covering algorithm[1] [5, 9, 12, 16]. The covering algorithm (see Figure 1) employs a heuristic hill-climbing search. Such search is subject to a number widely known difficulties including the problems of local minima, plateaus and ridges.

The covering algorithm, through its heuristic search, seeks to develop the smallest set of rules that adequately describe the training data. However, there is a growing body of evidence that learning redundant classifiers (classifiers that contain elements in addition to the bare minimum necessary to adequately describe the training data) can improve predictive accuracy [2, 8, 13, 14, 21].

In this context it seems surprising that there has been little evaluation of the relative power of the covering algorithm or exploration of alternative approaches to the induction of classification rules. This paper presents such an alternative and a comparative evaluation of its relative inductive power. This alternative is the induction of all rules.

Covering algorithms infer a set of rules by inferring one rule at a time. In most covering algorithms, at each stage, the rule that

[1] The limited number of alternatives includes the induction of decision trees that are then converted to classification rules, such as performed by C4.5rules [15], and the instance-based approach of RISE [8].

**On presentation of training examples** *training_examples*:

1. Initialise *rule_set* to a default (usually empty, or a rule assigning all objects to the most common class).
2. Initialise *examples* to either all available examples or all examples not correctly handled by *rule_set*.
3. Repeat
   (a) Find *best*, the best rule with respect to *examples*.
   (b) If such a rule can be found
      i. Add *best* to *rule_set*.
      ii. Set *examples* to all examples not handled correctly by *rule_set*.

   until no rule *best* can be found (for instance, because no examples remain).

**To classify object** *object*:
Apply *rule_set* to *object* employing a resolution strategy, such as selecting the rule with the greatest support with respect to *training_examples*, to resolve situations where *object* is covered by more than one rule.

Figure 1: The covering algorithm

performs best on the remaining training set is selected. As each rule is inferred and added to the rule set, the training set is adjusted to reflect the impact of the inclusion of that rule on the rule set's performance with respect to the training set.

The new technique, induction of all rules, instead employs a rule set composed of all rules defined by the learning system's language for describing rules (see Figure 2). In contrast to the incremental heuristic hill-climbing search employed in covering algorithms, learning all rules involves the use of no heuristics whatsoever during classifier formation.

This paper provides an empirical evaluation of the relative predictive accuracy obtained by a heuristic covering algorithm and by learning all rules. While each approach outperforms the other for some learning tasks, there appears to be a general advantage to the heuristic covering algorithm.

# 2 Learning all rules

There are a number of reasons to believe that there may be advantages to learning and employing all rules that characterise a domain as opposed to using a relatively blind hill-climbing search algorithm to select a subset of those rules.

As already outlined, hill-climbing search is subject to generic limitations. It is relatively common for this search strategy to lead to far from optimal solutions. By contrast, learning and employing all rules involves no heuristics. The set of rules is determined by the language for expressing rules with which the learning system is provided.

Further, any inductive rule selection strategy is going to be subject to uncertainty. Typically there will be many selections that could be made but, necessarily, only one subset of rules can be selected. This introduces an element of chance that may affect the quality of the classifiers that are learnt. In contrast, there is no such element of chance when all

**On presentation of training examples** *training_examples*:
Set *rule_set* to the set of all rules that can describe *training_examples*.

**To classify object** *object*:
Apply *rule_set* to *object* employing a resolution strategy, such as selecting the rule with the greatest support with respect to *training_examples*, to resolve situations where *object* is covered by more than one rule.

Figure 2: Learning all rules (abstract description)

rules are learnt.

Finally, there is a growing body of empirical evidence that learning and employing a minimal set of rules is in general sub-optimal, at least with respect to predictive accuracy.

Ali, Brunk and Pazzani [2] have developed a system that learns and employs multiple sets of definite clause rules. Both Domingos [8] and Nock and Gascuel [13] have developed systems that learn redundant (in the sense that there are more than the minimum number necessary) sets of classification rules. Oliver and Hand [14] have developed a system that learns and employs multiple decision trees for classification. Webb [21] has developed a decision tree post-processor that adds redundant (in the sense that they do not alter resubstitution performance) leaves to a decision tree. All of these systems learn and employ more rules (or in the case of decision trees, leaves) than would be developed by a standard covering algorithm. All have demonstrated that this can lead to increased predictive accuracy.

Learning all rules could be presented as the ultimate end point of this movement away from learning single classifiers of minimal complexity. Unless it is possible to identify a credible point at which one stops adding rules to the set of rules to be learnt and employed, if one abandons the notion that one should learn a minimal set of rules then learning all rules appears to be a logical outcome.

## 2.1 An approach to learning all rules

The number of possible rules for a given learning task may be infinite. In consequence, in the general case, it is clearly infeasible to produce individual explicit representations of all possible rules. However, this is not necessary in order to apply all possible rules. The approach that is employed herein is to produce explicit rule representations only when classifying an object. At this stage, the only explicit representations of rules that are created are of those rules that are directly relevant to classifying that object. Thus, rather than initially processing the training data to develop an explicit representation of the rules to be used for classification, the training objects will be retained. Then, when an object is to be classified, those rules directly relevant to classifying that object will be derived explicitly from the training objects. This process is outlined in Figure 3.

This raises the question of which rules will be relevant to classifying an object.

Covering algorithms can be used to develop two quite distinct forms of sets of rules. One possibility is to form *ordered rules* [5] (also referred to as *decision lists* [18]). These are ordered lists of rules. When such a list is applied to classify an object, the first rule to cover the object is used to classify that object.

The alternative is to use unordered rules [4, 6, 9]. In this case the set of rules is not ordered. For classification, all rules are examined to determine which covers an object.

**On presentation of training examples** *training_examples*:
Store *training_examples* for later use.

**To classify object** *object*:

1. Set *best* to the rule that covers *object* that has the greatest support with respect to *training_examples*.
2. Use *best* to classify *object*.

Figure 3: Learning all rules (actual process)

If more than one rule covers an object then some form of resolution procedure is invoked to select one rule with which to classify the object.

Systems that learn ordered rules often seek to order the rules from highest to lowest empirical support (where empirical support is a measure of how well the rule performs on the training data, usually favouring maximisation of the number of positive examples covered and minimization of the number of negative examples) [5]. Systems that learn unordered rules often resolve situations in which multiple rules cover a single object by selecting the rule with the highest empirical support [4, 6, 10].

If, when seeking to classify an object, one finds the rule that covers the object and that has the highest empirical support from the training objects then one will obtain the same classification as if all rules have been explicitly developed and ordered on empirical support, or if all rules had been explicitly developed, applied in an unordered manner and situations in which multiple rules cover an object were resolved by selecting the rule with highest empirical support. Thus, to obtain the effect of employing all rules, using either the ordered or unordered rules approaches, it is necessary only, when classifying an object, to perform a search of the space of rules that cover that object in order to find that with the highest empirical support. The computational tractability reduces to that of a single search. Webb [20] has demonstrated that such search is feasible for a wide range of categorical attribute-value learning tasks.

## 2.2 The learning system

A learning system was implemented to support this form of induction. All training objects were retained for use during classification. To classify an object, a search was performed through the space of all rules that covered the object. The rule with the highest empirical support was selected and used for classification. Admissible search was employed to guarantee that the rule with the highest empirical support was always selected. The OPUS search algorithm [20] was employed to perform this search. Two metrics of empirical support were employed. The first, *max consistent*, favoured rules that covered no negative examples and covered the most positive examples. The second, *Laplace*, allowed a trade-off between positive and negative cover.

Let $N$ be the number of negative training objects covered, $P$ the number of positive training objects and $C$ the number of classes. The max consistent empirical support for a rule equals $-N$, if $N > 0$, else $P$. The Laplace empirical support for a rule equals $\frac{P+1}{P+N+C}$.

Where multiple rules shared the same maximal empirical support, a rule from the class mentioned first in the data description (*names*) file was selected. This corresponds to the resolution procedure used in the covering algorithm with which the induction of all

rules was compared.

For the sake of computational efficiency, after each classification using all rules, the rule employed was recorded. Subsequent searches were made more efficient by examining the list of previously employed rules for the highest valued rule that covered the new object. If such a rule was found, the search was seeded by setting the best rule found so far to the identified rule. This enabled rapid pruning of the search space.

# 3 Experimental evaluation

In keeping with common machine learning practice, the two machine learning approaches were evaluated with respect to their predictive accuracy—the proportion of previously unseen objects that the system could correctly classify.

Due to the limitation of current implementations of the OPUS search algorithm to searching for categorical attribute-value rules, evaluation was restricted to categorical attribute-value machine learning tasks. All such tasks of which the author was aware from the UCI machine learning repository [11] were employed. These sixteen tasks are described in Table 1. For each data set this table presents the number of attributes, the number of classes, the number of rules that could be formed for each class and the number of objects in the data set.

The antecedents of the rules took the form of a conjunction of equality tests. For each attribute there was at most one test. Each test took the form $attribute = value$. In consequence, for each class there were

$$\prod_{i=1}^{a}(v_i + 1)$$

possible rules, where $a$ is the number of attributes and $v_i$ is the number of values for attribute $i$. The consequent of each rule was a simple class assignment. This form of rule was chosen because of its widespread use in machine learning.

The Cover machine learning system [19] was used as the covering algorithm for the experiments. It was used to develop unordered rules. The algorithm employed is presented in Figure 4.

This algorithm is identical to the unordered version of CN2 [4] with the exception that at step 2(b)i the OPUS search algorithm was used to provide admissible search in place of the heuristic search employed within CN2. Note that while admissible search is employed to find individual rules within the covering algorithm, the outer search for a set of rules still employs the standard heuristic covering search.

Cover was employed with both of the empirical support metrics employed with the all rules approach.

As a control, to evaluate whether the Cover system was performing at a credible level, C4.5 was also included in the study.

Each data set was randomly divided into training (80%) and evaluation (20%) sets 100 times. For each pair of training and evaluation sets so formed, all five learning methods (all rules max consistent; all rules Laplace; Cover max consistent; Cover Laplace; and C4.5) were applied to the training set and the predictive accuracy of the resulting classifiers evaluated on the evaluation set.

Table 2 presents for each domain the mean and standard error of the accuracy for each treatment. For two domains (lymphography and monk 2) learning all rules with the max consistent empirical support metric results in the highest mean accuracy. For a further four domains (Slovenian breast cancer, Wisconsin breast cancer, house-votes-84, primary tumor), making a total of six, all rules with max consistent empirical metric obtains a higher mean accuracy than Cover with either empirical support metric.

Learning all rules with the Laplace empirical support metric does not perform so well. For no domain does it obtain a higher mean accuracy than all other treatments. For only

Table 1: Summary of experimental data sets.

| Domain | Attributes | Classes | Rules | Cases |
|---|---|---|---|---|
| Audiology | 70 | 24 | $1.831298 \times 10^{35}$ | 226 |
| House Votes 84 | 17 | 2 | $4.294967 \times 10^{09}$ | 435 |
| KR vs KP | 37 | 2 | $2.668349 \times 10^{17}$ | 3198 |
| Lenses | 5 | 3 | $1.080000 \times 10^{02}$ | 24 |
| Lymphography | 19 | 4 | $7.971615 \times 10^{10}$ | 148 |
| Monk 1 | 7 | 2 | $2.880000 \times 10^{03}$ | 556 |
| Monk 2 | 7 | 2 | $2.880000 \times 10^{03}$ | 601 |
| Monk 3 | 7 | 2 | $2.880000 \times 10^{03}$ | 554 |
| Multiplexor | 12 | 2 | $1.771470 \times 10^{05}$ | 500 |
| Mushroom | 23 | 2 | $1.634593 \times 10^{17}$ | 8124 |
| Primary Tumor | 18 | 22 | $1.133741 \times 10^{09}$ | 339 |
| Promoters | 58 | 2 | $6.938894 \times 10^{39}$ | 106 |
| Slovenian Breast Cancer (SBC) | 10 | 2 | $7.338240 \times 10^{06}$ | 286 |
| Soybean Large | 36 | 19 | $3.852636 \times 10^{23}$ | 307 |
| Tic Tac Toe | 10 | 2 | $2.621440 \times 10^{05}$ | 958 |
| Wisconsin Breast Cancer (WBC) | 10 | 2 | $2.572307 \times 10^{09}$ | 699 |

Table 2: Mean and standard error of accuracy for each treatment and domain

| | All rules | | Cover | | |
|---|---|---|---|---|---|
| **Domain** | **m cons** | **Laplace** | **m cons** | **Laplace** | **C4.5** |
| Audiology | 0.57±0.01 | 0.29±0.01 | 0.66±0.01 | 0.64±0.01 | 0.77±0.01 |
| House Votes 84 | 0.94±0.00 | 0.94±0.00 | 0.93±0.00 | 0.93±0.00 | 0.95±0.00 |
| KR vs KP | 0.97±0.00 | 0.97±0.00 | 0.99±0.00 | 0.99±0.00 | 0.99±0.00 |
| Lenses | 0.66±0.02 | 0.63±0.02 | 0.81±0.02 | 0.82±0.02 | 0.80±0.01 |
| Lymphography | 0.80±0.01 | 0.79±0.01 | 0.79±0.01 | 0.79±0.01 | 0.76±0.01 |
| Monk 1 | 1.00±0.00 | 1.00±0.00 | 1.00±0.00 | 1.00±0.00 | 0.96±0.00 |
| Monk 2 | 0.81±0.00 | 0.67±0.00 | 0.79±0.00 | 0.78±0.00 | 0.64±0.01 |
| Monk 3 | 0.97±0.00 | 0.98±0.00 | 0.98±0.00 | 0.98±0.00 | 0.99±0.00 |
| Multiplexor | 0.98±0.00 | 0.98±0.00 | 0.99±0.00 | 0.99±0.00 | 0.85±0.01 |
| Mushroom | 1.00±0.00 | 1.00±0.00 | 1.00±0.00 | 1.00±0.00 | 1.00±0.00 |
| Primary Tumor | 0.39±0.01 | 0.33±0.01 | 0.35±0.01 | 0.35±0.01 | 0.39±0.01 |
| Promoters | 0.61±0.02 | 0.61±0.02 | 0.71±0.01 | 0.71±0.01 | 0.76±0.01 |
| SBC | 0.71±0.01 | 0.72±0.01 | 0.68±0.01 | 0.68±0.01 | 0.74±0.01 |
| Soybean Large | 0.76±0.00 | 0.44±0.01 | 0.88±0.00 | 0.89±0.00 | 0.87±0.00 |
| Tic Tac Toe | 0.96±0.00 | 0.96±0.00 | 0.96±0.00 | 0.96±0.00 | 0.83±0.00 |
| WBC | 0.95±0.00 | 0.92±0.00 | 0.92±0.00 | 0.92±0.00 | 0.95±0.00 |

1. *ruleset* ← ∅.
2. for *class* ← each class if turn
   (a) *examples* ← the training examples.
   (b) while *examples* contains objects belonging to *class*
      i. *rule* ← the rule for class *class* with highest empirical support on the training set
      ii. remove all objects of class *class* that are covered by *rule* from the training set
      iii. add *rule* to *ruleset*

Figure 4: The Cover algorithm

two (Slovenian breast cancer and house-votes-84) does it obtain higher accuracy than Cover.

Both Cover treatments outperform all rules with the max consistent empirical support metric for the same seven domains (audiology, kr-vs-kp, lenses, monk 3, F11 multiplexor, promoters and soybean large). Both Cover treatments outperform all rules with the Laplace empirical support metric for the same eight domains (audiology, kr-vs-kp, lenses, monk 2, F11 multiplexor, promoters, primary tumor and soybean large). Both Cover treatments have higher accuracy than C4.5 for seven domains (lenses, lymphography, monk 1, monk 2, F11 multiplexor, soybean large and tic-tic-toe) and lower for six domains (audiology, Slovenian breast cancer, Wisconsin breast cancer, house-votes-84, monk 3 and promoters).

These results suggest that learning all rules with the Laplace empirical support metric is in general less effective than the other approaches, but that there is little general difference between the remaining four treatments.

To assess these apparent outcomes, a Friedman rank test was used to evaluate the statistical significance of the observed differences between treatments. This test evaluates whether at least one of the treatments tends to yield larger observed values than at least one other treatment [7]. The test was applied to all 1600 observations resulting from the 100 observations for each of the 16 domains. The result ($f$=69.02, $p$=0.000) shows that there is a difference for at least one treatment that is significant at the 0.05 level. To compare each combination of pairs of treatments, a multiple comparisons test [7] was employed. This shows which pairs of treatments have rankings that significantly differ and the direction of that difference. The result is displayed in Table 3. In this table, '<' indicates that the treatment for the row has obtained a lower rank significantly (at the 0.05 level) more often than the treatment for the column. '>' indicates that the treatment for the row obtained a higher rank significantly (at the 0.05 level) more often than the treatment for the column. '=' indicates no overall significant difference in ranking. The table indicates that—

- all rules Laplace is in general ranked lower than all other treatments;
- there is not a significant difference in the relative rankings of all rules max consistent and Cover max consistent (all rules max consistent obtained a higher accuracy in 541 cases and lower in 644 cases);
- all rules max consistent is ranked lower than Cover Laplace (which obtained

higher accuracy in 661 cases and lower in 527 cases) and C4.5 (which obtained higher accuracy in 720 cases and lower in 599 cases); and

- there is no significant difference in the general rankings of Cover max consistent, Cover Laplace and C4.5.

# 4 Discussion

It seems apparent that learning all rules with the Laplace empirical support metric is not a credible alternative to the traditional covering algorithm. The situation with respect to learning all rules with the max consistent empirical support metric is less clear cut, however. While both versions of Cover and C4.5 all achieve higher accuracy more often than learning all rules with the max consistent empirical support metric, learning all rules still achieves higher predictive accuracy in a large number of cases. While learning all rules does not in general provide better performance, it is credible that there exist types of problem for which it will provide better performance.

If one considers the five domains for which the space of possible rules contains more than $10^{15}$ rules, it is striking that for all but one domain the average accuracy for Cover is higher than that for learning all rules. For that one exception, the mushroom domain, all treatments always achieve 100% accuracy—the data so clearly defines the decision surfaces that the learning task is quite straight forward.

If one considers the eleven domains for which the space of all possible rules contains less than $10^{15}$ rules, learning all rules with the max consistent empirical support metric achieves higher mean accuracy for six and Cover achieves higher mean accuracy for only two.

One possible explanation for this effect is oversearch [17]. As Quinlan and Cameron-Jones [17] describe this effect, the more rules there are in the space of possible rules, the greater the probability that there will be rules that have high apparent empirical support from the training set but low predictive power.

However, there is a further issue that Quinlan and Cameron-Jones [17] do not raise. The probability of this misleading high apparent support occurring will also depend upon the quality of the training set. If the training set contains a broad range of objects that clearly delimit the decision surfaces (within the language for expressing the rules that is employed), as appears to be the case with the mushroom data, then it is not relevant how large the space of possible rules is. The probability of finding rules with high empirical support but low predictive accuracy will be low, irrespective. If the training set does not clearly delimit the decision surfaces then even a relatively small space of possible rules may include rules with deceptively high empirical support. Possible reasons for a training set failing to clearly delimit the decision surfaces might include—

- too few objects (as might be the case for the lenses data set);
- noise (as is the case for the monks 3 domain); and
- poor distribution of objects.

It is credible from this analysis and the empirical results presented above that learning all rules is indicated where the training set clearly delimits the decision surfaces but is not indicated otherwise. However, this leaves unresolved how one should identify whether the training set clearly delimits the decision surfaces!

Another issue that needs consideration is whether the technique that has been developed is the best way to use all rules. During classification, the current technique arbitrarily selects one from any set of rules with maximal empirical support that cover an object. It would seem appropriate, however, to take account of the class distribution of all such rules. If there are many rules with high empirical support that cover an object for one class

Table 3: Multiple comparisons test

| | | All rules | | Cover | | |
|---|---|---|---|---|---|---|
| | | max cons | Laplace | max cons | Laplace | C4.5 |
| all rules | max cons | | > | = | < | < |
| | Laplace | < | | < | < | < |
| Cover | max cons | = | > | | = | = |
| | Laplace | > | > | = | | = |
| | C4.5 | > | > | = | = | |

but not for the others, then this would seem to support selecting that class over the others. It might be worth investigating techniques that take account of such issues by combining the evidence from all relevant rules. Some form of probabalistic combination of the evidence from multiple rules has intuitive appeal, but faces the difficulty that one cannot assume independence between the rules involved and hence that it is far from clear how the evidence could be best combined.

It is also worth considering that there may be a possible confound in the experimental work from the use of admissible search within the covering algorithm. Such use of admissible search is certainly not standard practice in covering algorithms. However, comparison with C4.5 shows that the covering algorithms are performing at close to the defacto 'standard', suggesting that this is not a serious issue.

## 4.1 Related Research

The learning all rules approach to induction has some commonalities with instance based learning [1]. Like instance based learning, classification is performed by reference to the training set at the time of classification. The learning all rules approach could be considered to be a form of qualitative instance based learning whereby the selected rule is used to define a similarity metric for classification in place of the use of a distance metric.

The use of such a qualitative similarity metric instead of geometric distance metrics might be justified on the grounds that it is not possible to derive a priori distance metrics for accurate measurement of similarity. Even for a single ordinal attribute, it is not possible to determine a priori whether the correct similarity metric is linear with respect to the numeric value. Is an eighteen year old person more similar to a one year old or forty year old? It is implausible that it is possible to provide an a priori answer to such a question. The problem is further compounded by the possible incommensurability of metrics represented by different attributes, as illustrated by the question "is an eighteen year old male more similar to an eighteen year old female or to an eighty-one year old male?".

There are also commonalities with OSP [3]. OSP performs induction by selecting a set of rules at classification time that cover the object to be classified and have high empirical support with respect to the training set. The class distribution within the set of training objects covered by one or more of these rules is then used for classification. The two approaches are similar due to their search for classification rules that cover the object to be classified and which have high empirical support with respect to the training set. They differ in that the learning all rules approach selects and directly uses just one of these rules while OSP uses all of the rules as a filter on the training set and then employs the filtered training set for classification.

# 5 Conclusion

The strategy of learning all rules has been presented as a theoretically credible alternative to heuristic covering algorithms. However, experimental evaluation failed to demonstrate a general advantage to the approach and, indeed, suggested that the heuristic covering algorithms held a significant general advantage. Further analysis of these results revealed an apparent correlation between the size of the space of possible rules and the probability of learning all rules outperforming the covering algorithms. One possible explanation of this effect is that for such learning tasks oversearch leads to the use of rules with misleading apparent empirical support from the training set. If this analysis is correct, learning all rules might be expected to outperform a covering algorithm when the training data clearly delineates the decision surfaces for a domain. Where the data is less comprehensive, the heuristic covering algorithm would appear to remain the method of choice.

## Acknowledgments

This research has been supported by the Australian Research Council.

I am grateful to Zijian Zheng for helpful comments on previous drafts of this paper.

# References

[1] David W. Aha, Dennis Kibler, and Marc K. Albert. Instance-based learning algorithms. *Machine Learning*, 6:37–66, 1991.

[2] Kamal Ali, Clifford Brunk, and Michael Pazzani. On learning multiple descriptions of a concept. In *Proceedings of Tools with Artificial Intelligence*, pages 476–483, New Orleans, LA, 1994.

[3] Lionel C. Briand, Victor R. Basili, and William M. Thomas. A pattern recognition approach for software engineering data analysis. *IEEE Transactions on Software Engineering*, 18(11):931–942, 1992.

[4] Peter Clark and Robin Boswell. Rule induction with CN2: Some recent improvements. In *Proceedings of the Fifth European Working Session on Learning*, pages 151–163, 1991.

[5] Peter Clark and Tim Niblett. The CN2 induction algorithm. *Machine Learning*, 3:261–284, 1989.

[6] S. H. Clearwater and F. J Provost. RL4: A tool for knowledge-based induction. In *Proceedings of Second Intl. IEEE Conf. on Tools for AI*, pages 24–30, Los Alamitos, CA, 1990. IEEE Computer Society Press.

[7] W. J. Conover. *Practical Nonparametric Statistics*. Wiley, New York, 1980.

[8] Pedro Domingos. Rule induction and instance-based learning: A unified approach. In *IJCAI-95*, pages 1226–1232, Montreal, 1995. Morgan Kaufmann.

[9] Ryszard S. Michalski. A theory and methodology of inductive learning. In Ryszard S. Michalski, Jaime G. Carbonell, and Tom M. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*, pages 83–129. Springer-Verlag, Berlin, 1983.

[10] Ryszard S. Michalski, Igor Mozetic, Jiarong Hong, and Nada Lavrac. The multi-purpose incremental learning system AQ15 and its testing and application to three medical domains. In *AAAI-86: Proceedings of the Fifth National Conference on Artificial Intelligence*, pages 1041–1045, Philadelphia, 1986. AAAI.

[11] Patrick M. Murphy and David W. Aha. UCI repository of machine learning databases. [Machine-readable data

repository]. University of California, Department of Information and Computer Science, Irvine, CA., 1995.

[12] Stephen Muggleton and Cao Feng. Efficient induction of logic programs. In *Proceedings of the First Conference on Algorithmic Learning Theory*, Tokyo, 1990.

[13] Richard Nock and Olivier Gascuel. On learning decision committees. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 413–420, Taho City, CA, July 1995. Morgan Kaufmann.

[14] Jonathon J. Oliver and David J. Hand. On pruning and averaging decision trees. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 430–437, Taho City, CA, July 1995. Morgan Kaufmann.

[15] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA, 1993.

[16] J. Ross Quinlan and R. M. Cameron-Jones. Induction of logic programs: FOIL and related systems. *New Generation Computing*, 1995.

[17] J. Ross Quinlan and R. M. Cameron-Jones. Oversearching and layered search in empirical learning. In *IJCAI'95*, pages 1019–1024, Montreal, 1995. Morgan Kaufmann.

[18] Ronald L. Rivest. Learning decision lists. *Machine Learning*, 2:229–246, 1987.

[19] Geoffrey I. Webb. Systematic search for categorical attribute-value data-driven machine learning. In Chris Rowles, Huan Liu, and Norman Foo, editors, *AI'93 – Proceedings of the Sixth Australian Joint Conference on Artificial Intelligence*, pages 342–347, Melbourne, 1993. World Scientific.

[20] Geoffrey I. Webb. OPUS: An efficient admissible algorithm for unordered search. *Journal of Artificial Intelligence Research*, 3:431–465, 1995.

[21] Geoffrey I. Webb. Further experimental evidence against the utility of Occam's razor. *Journal of Artificial Intelligence Research*, 4:397–417, 1996.