

Fast Discovery of Simple Rules

Pedro Domingos*

Department of Information and Computer Science
University of California, Irvine
Irvine, California 92717, U.S.A.
pedrod@ics.uci.edu
<http://www.ics.uci.edu/~pedrod>

The recent emergence of data mining as a major application of machine learning has led to increased interest in fast rule induction algorithms. These are able to efficiently process large numbers of examples, under the constraint of still achieving good accuracy. If e is the number of examples, many rule learners have $\Theta(e^4)$ asymptotic time complexity in noisy domains, and C4.5RULES has been empirically observed to sometimes require $\Theta(e^3)$ time. Recent advances have brought this bound down to $\Theta(e \log^2 e)$, while maintaining accuracy at the level of C4.5RULES's (Cohen 1995). Ideally, we would like to have an algorithm capable of inducing accurate rules in time linear in e , without becoming too expensive in other factors. This extended abstract presents such an algorithm.

Most rule induction algorithms employ a “separate and conquer” method, inducing each rule to its full length before going on to the next one. They also evaluate each rule by itself, without regard to the effect of other rules. This is a potentially inefficient approach: rules may be grown further than they need to be, only to be pruned back afterwards, when the whole rule set has already been induced. An alternative is to interleave the construction of all rules, evaluating each rule in the context of the current rule set. This can be termed a “conquering without separating” approach, by contrast with the earlier method, and has been implemented in the CWS algorithm.

CWS is outlined in pseudo-code in Table 1. All examples are initially assigned to the majority class. Each rule in CWS is associated with a vector of class probabilities computed from the examples it covers, and predicts the most probable class. Conflicts are resolved by summing the probabilities for all rules covering the test instance, and choosing the class with the highest sum. $\text{Acc}(\text{RS})$ is the accuracy of the rule set RS on the training set. This procedure would not be efficient if implemented directly, but, by avoiding the extensive redundancy present in the repeated computation of accuracies and class probabilities, the worst-case time complexity of CWS can be made linear in e and all other relevant parameters.

CWS has been extensively evaluated using benchmark problems, a large artificial dataset, and a detailed

Table 1: The CWS algorithm.

```
Let RS = EmptySet.
Repeat
  Add one active rule with empty body to RS.
  For each active rule R in RS,
    For each possible antecedent AV,
      Let R' = R with AV conjoined.
      Compute class probabilities for R'.
      Let RS' = RS with R replaced by R'.
      If Acc(RS') > Acc(RS)
        then replace RS by RS'.
  If RS' is still RS then deactivate R.
Until all rules are inactive.
Return RS.
```

study on a large database related to process control on NASA's space shuttle. These studies show that, as expected, CWS's running time grows much more slowly with the number of examples than C4.5RULES's. CWS's rate of growth is linear in the artificial dataset (a noisy Boolean function), and $e \log e$ in the shuttle domain with 20% class noise, with the logarithmic factor due to processing of numeric attributes. C4.5RULES's time is approximately $e^2 \log e$ in the artificial database, and e^3 in the shuttle problem. CWS takes 11 hours on the full shuttle database, while C4.5RULES takes over two weeks. CWS is more accurate than C4.5RULES on the shuttle and artificial problems, and there is no significant difference on most benchmark datasets. CWS almost always produces simpler output than C4.5RULES, and hence, in principle, more easily understood results. This is an important consideration in many data mining problems, including the shuttle control one studied. In summary, this study indicates that CWS can be used to advantage when the underlying concept is simple and the data is plentiful but noisy.

References

Cohen, W. W. 1995. Fast Effective Rule Induction. In *Proceedings of the Twelfth International Conference on Machine Learning*, 115–123. Tahoe City, Calif.: Morgan Kaufmann.

*Partly supported by a PRAXIS XXI scholarship.