FINDING SMALL HIGH PERFORMANCE SUBSETS OF INDUCED RULE SETS: EXTENDED SUMMARY

Thomas Ågotnes, Jan Komorowski, Aleksander Øhrn
Knowledge Systems Group, Department of Computer and Information Science
Norwegian University of Science and Technology
7491 Trondheim, Norway
Phone: +47 73593440, Fax: +47 73594466
e-mail: {agotnes, janko, aleks}@idi.ntnu.no

ABSTRACT: Models consisting of decision rules – such as those produced by methods from Pawlak's rough set theory – generally have a white-box nature, but in practice induced models are too large to be inspected. Here, we investigate methods for simplifying complex models while retaining predictive performance. The approach taken is rule filtering, i.e. post-pruning of complete rules. Two methods for finding high-performance subsets of a set of rules are investigated. One method is to use a genetic algorithm to search the space of subsets. Another method is to create an ordering of a rule set by sorting the rules according to a quality measure for individual rules. A rule set with a particular cardinality and expected good predictive performance can then be constructed by taking the first rules in the ordering. Algorithms for the two methods have been implemented in the ROSETTA system. Predictive performance is estimated using ROC analysis, and compared using statistical hypothesis testing. Ten different formulae from the literature that can be used to define rule quality are compared. Experiments on real-world data show that models often can be dramatically simplified without significant performance loss.

KEYWORDS: Rule filtering, rule pruning, rule-based models, rule quality, rough set theory, data mining, knowledge discovery

INTRODUCTION

One alleged advantage of rule-based models, such as those resulting from methods from Pawlak's rough set theory, is the so-called white-box property – the applicability for human inspection, interpretation and knowledge discovery. However, rule-based models induced from real-world data most often are so complex that they in effect are opaque. We suspect that the large complexity of induced models often is due to noise in, and/or overfitting to, the training data. According to the principle of Occam's razor, simpler consistent models should be preferred if they can be found. Models can generally be classified according to two properties: discriminatory ability (classifier performance) and intelligibility (descriptiveness). The problem considered here is finding more descriptive models by reducing the complexity of a given rule-based model while retaining the classifier performance.

Methods for *rule pruning* – i.e. simplification of rule-based models – vary along at least two dimensions. First, rule pruning methods are categorized by the point in time the pruning decision is made. *Pre-pruning* methods prune a model while it is being constructed (i.e. during rule induction), while *post-pruning* methods prune a model after the completion of the induction phase. Second, pruning of a set of propositional rules are generally done by either removing conjuncts from the antecedents of the rules, removing complete rules from the set, or a combination of both. The literature is rich with methods for pruning various model representations. Fürnkranz (1997) presents algorithms for combining pre- and post-pruning in rule based systems, in addition to an overview of general pruning algorithms. Methods from rough set theory related to rule pruning are dynamic reducts (Bazan, 1998), default rules (Mollestad and Skowron, 1996; Mollestad, 1997; Mollestad and Komorowski, 1998), rough data models (Kowalczyk, 1998; Løken, 1999), and others.

The approach taken in this study is post-pruning of complete rules from an induced – generally complex – rule set, hence-

forth called *rule filtering*. Rule filtering corresponds to finding small subsets of a given rule set, without significantly lower classifier performance. We investigate two methods for finding such subsets. The first is to use a genetic algorithm to search the space of subsets. The second method is to remove the rules with worst individual *rule quality*. Two experiments on real-world data are presented, where rule-based models induced with common learning algorithms are dramatically filtered down – generally without any significant performance drop. These preliminary experiments also identify certain rule quality formulae from the literature as being useful for rule filtering. The methods and experiments are further discussed in Ågotnes (1999). In the following, it is assumed that the reader has some familiarity with the use of the rough set framework to extract rules from data.

RULE FILTERING

The models considered here are induced from data and are on the form $RUL = \{r_1, \ldots, r_n\}$, $r_i = \alpha_i \rightarrow \beta_i$, where each α_i is a conjunction of descriptors over the condition attributes, and each β_i a descriptor over the decision attribute, of a decision system. Such models can both be induced from data and used for classification purposes using the ROSETTA system (Øhrn et al., 1998b; Øhrn, 1999a,b), a tool-kit for data mining and knowledge discovery within the framework of rough set theory.

The general problem of model pruning is finding submodels with lower complexity but without significantly lower predictive performance. Thus, two general properties of models are *performance* and *complexity*. The former tells something about the descriptiveness of a model, the latter is a measure of how well the model classifies previously unseen data. For rule filtering, submodels correspond to subsets and complexity is equal to rule count. The rule filtering problem thus consists of finding a high performance subset RUL' of a given rule set RUL. Here, a set of rules was used as a classifier by using the voting scheme (Øhrn, 1999b, p. 30) implemented in ROSETTA. To measure predictive performance, we used ROC (Relative Operating Characteristics) analysis; a method with origins in signal theory that is commonly used in medical diagnosis and is gaining popularity in machine learning. For comparative purposes, the area under the ROC curve (the AUC) is recommended as the appropriate measure of classifier accuracy compared to traditional accuracy measures (Swets, 1988). Difference in calculated AUC values for two classifiers may be random. Hanley and McNeil (1983) provide a statistical hypothesis test for detecting statistical significant difference in two correlated (calculated from the same data) AUC values. We used Hanley and McNeil's test with a 5% significance level for the two-tailed test using the Pearson correlation measure.

Below, two schemes for finding small, high-performance subsets of a given rule set are presented.

GENETIC FILTERING

A genetic algorithm for rule filtering was implemented in the ROSETTA framework, based on an implementation by Vinterbo (1999). The rule filtering problem lends itself to a very natural coding for a genetic algorithm. The problem can be seen as a search over the lattice of subsets of a rule set RUL. An individual in the genetic algorithm (a possible solution to the problem) is a bitstring of length $|RUL|$ and represents a node in the lattice (a subset of RUL). Each bit in the bitstring indicates the presence or absence of a particular rule from the unfiltered set RUL. See Vinterbo (1999) for a description of the mechanisms implemented in the genetic algorithm. See e.g. Goldberg (1989) for an introduction to genetic algorithms. For performance reasons, the bitstring implementation in the original algorithm was replaced with integer vectors giving the indices to the included rules.

The general fitness function is a linear combination of complexity and performance. For $RUL' \subseteq RUL$,

$$fitness_\lambda(RUL') = \lambda performance(RUL') + (1-\lambda)(1 - complexity_{RUL}(RUL'))$$

where $complexity_{RUL}(RUL')$ is the complexity of RUL' relative to RUL: $complexity_{RUL}(RUL') = |RUL'|/|RUL|$. When only the smallest subsets are of interest, only a small part of the search space needs to be considered. This was done by using a fitness function with a *cut-off value n*:

$$fitness_{\lambda,n}(RUL') = \begin{cases} fitness_\lambda(RUL') & \text{if } complexity(RUL') < n \\ 0 & \text{otherwise} \end{cases}$$

For each generation, the genetic algorithm updates a *keep-list*. The keep-list consists of the best individual seen so far for each of the rule set cardinalities no larger than the cut-off value. Each of the individuals in the population is compared to the individual with the same cardinality in the keep-list, and is used to replace the latter if the individual from the population

has higher predictive performance. The criteria for termination of the algorithm was that no changes in the keep-list or in the average population fitness were detected for a number of generations.

Rule filtering was done by plotting the model performance versus the rule set cardinality for each of the rule sets in the keep list, and selecting models according to given problem specific criteria.

QUALITY-BASED FILTERING

In addition to assessing the predictive performance of rule-based models, the performance of each of the individual rules in the model could be assessed. If we can find a performance measure for individual rules – henceforth called measure of *rule quality* – so that the best (according to the model performance measure) subset of cardinality n consists of the n individually best (according to the rule quality measure) rules, then the rule filtering problem is easily solved by sorting the unfiltered rule set according to rule quality. A filtered rule set can then be found by including enough of the best rules to comply with given performance and/or complexity criteria.

Numerical properties of decision rules are generally derived from the contingency table. The contingency table for the rule $r = \alpha \rightarrow \beta$ is a tabulation of the number of objects from the rule's originating decision system that matches the antecedent and/or the consequent:

	β	$\neg\beta$	
α	$n_{\alpha,\beta}$	$n_{\alpha,\neg\beta}$	n_α
$\neg\alpha$	$n_{\neg\alpha,\beta}$	$n_{\neg\alpha,\neg\beta}$	$n_{\neg\alpha}$
	n_β	$n_{\neg\beta}$	$\lvert U\rvert$

where $|U|$ is the total number of objects. Often, the relative frequencies are used: $f_{\phi,\psi} = n_{\phi,\psi}/|U|$, $f_\phi = n_\phi/|U|$ and $f_\psi = n_\psi/|U|$ for $\phi \in \{\alpha, \neg\alpha\}$ and $\psi \in \{\beta, \neg\beta\}$. The most commonly referenced numerical properties for a rule $r = \alpha \rightarrow \beta$ are $accuracy(r) = n_{\alpha,\beta}/n_\alpha$ and $coverage(r) = n_{\alpha,\beta}/n_\beta$. Øhrn et al. (1998a) use coverage as a rule quality measure for rule filtering.

Bruha (1997) gives an overview of some of the rule quality formulae used in the literature (summarized in Table 1). We used these quality functions for rule filtering by plotting the model performances versus the number of rules with the best rule quality included from the corresponding orderings of the unfiltered rule set, and selecting models according to given problem specific criteria.

Quality measure	Formula
Michalski	$\mu \cdot accuracy(r) + (1-\mu) \cdot coverage(r)$
Torgo	Michalski with $\mu = \frac{1}{2} + \frac{1}{4} accuracy(r)$
Brazdil	$accuracy(r) \cdot e^{coverage(r)-1}$
Pearson	$(n_{\alpha,\beta} \cdot n_{\neg\alpha,\neg\beta} - n_{\alpha,\neg\beta} \cdot n_{\neg\alpha,\beta}) / (n_\beta \cdot n_{\neg\beta} \cdot n_\alpha \cdot n_{\neg\alpha})$
G2	$2\left(n_{\alpha,\beta} \ln(n_{\alpha,\beta} \cdot \lvert U\rvert / (n_\alpha \cdot n_\beta)) + n_{\alpha,\neg\beta} \ln(n_{\alpha,\neg\beta} \cdot \lvert U\rvert / (n_\alpha \cdot n_{\neg\beta}))\right)$
J	$quality_{G2}(r)/(2\lvert U\rvert)$
Cohen	$(\lvert U\rvert \cdot n_{\alpha,\beta} + \lvert U\rvert \cdot n_{\neg\alpha,\neg\beta} - n_\alpha \cdot n_\beta) / (\lvert U\rvert^2 - n_\alpha \cdot n_\beta - n_{\neg\alpha} \cdot n_{\neg\beta})$
Coleman	$(\lvert U\rvert \cdot n_{\alpha,\beta} - n_\alpha \cdot n_\beta) / (n_\alpha \cdot n_{\neg\beta})$
C1	$quality_{Coleman}(r)\,((2 + quality_{Cohen}(r))/3)$
C2	$quality_{Coleman}(r)\,((1 + coverage(r))/2)$
Kononenko	$-\log_2 f_\beta + \log_2 accuracy(r)$

Table 1: Rule quality formulae (Bruha 1997)

EXPERIMENTAL RESULTS

Two preliminary experiments using the rule filtering schemes presented herein (i.e. genetic filtering and quality-based filtering) have been carried out. In the first experiment, henceforth called "the acute appendicitis experiment", we used a data set describing 257 patients with suspected acute appendicitis collected at Innherred Hospital in Norway (Hallan

et al., 1997). This data set has previously been mined by Carlin (1998) and Carlin et al. (1998) using methods from rough set theory. In the second experiment, henceforth called "the Cleveland experiment", we used the *Cleveland heart disease database*, available from the UCI repository (Murphy and Aha, 1995), consisting of 303 patients with suspected coronary artery disease. 6 of the objects had missing values for one or more attributes, and were removed.

Both datasets were split into equally sized training and testing sets three times, and all the steps below were repeated for each split. The training sets in the Cleveland experiment were used with an algorithm based on Boolean reasoning (Nguyen and Skowron, 1995) to find splits for discretizing the complete data sets. The manual discretization used by Carlin (1998) was used in the acute appendicitis experiment. Each training set was again split into two equally sized sets: the learning set and the hold-out set.

Rule induction was done using the learning sets. The rule induction methods and parameters[1] recommended by Carlin (1998) were used for initial learning in the acute appendicitis experiment. In the Cleveland experiment, initial learning was done using a default scheme[2].

The hold-out sets were used to assess performance for the fitness function in the genetic algorithm, and to plot performance versus complexity for the selection of filtered models for both rule filtering methods.

The two data sets were used to illustrate two slightly different rule filtering applications. In the acute appendicitis experiment, the goal was to find descriptive models. These were subjectively defined as models with no more than 20 rules. In the Cleveland experiment, the goal was to filter down the rule sets as much as possible without any constraint on maximal rule set size. The execution of the two experiments differed in the selected parameters (e.g. the weight λ for the fitness function in the genetic algorithm), and in the procedure used to select a particular filtered model from several alternatives. For the genetic algorithm and for quality-based filtering with each of the quality formulae in Table 1, the performance on the hold-out sets were plotted against the rule set cardinality. A quality formula is appropriate for rule filtering only if the corresponding performance plots exhibit a general monotony towards the rule set cardinality (the quality threshold). Combinations of quality formula and split subjectively classified as unsuitable were rejected. This was only necessary in the acute appendicitis experiment. A filtered model was selected from the non-rejected plots as follows. In the acute appendicitis experiment, the smallest rule set without significantly lower performance compared to the unfiltered set was selected for quality-based filtering, if it was smaller than the maximal rule size of 20 rules. Otherwise, no model was selected for that particular quality formula/split combination. For the genetic algorithm, the rule sets were subjectively selected by visual inspection of the plots. In the Cleveland experiment, very small filtered sets were not the principal consideration. For the quality-based approach rule sets were subjectively selected by visual inspection, and for the genetic algorithm the sets with the highest performance in the keep-lists were selected.

The selected models were applied to the testing sets and compared to the unfiltered models (Table 2).

DISCUSSION

The experiments presented here are preliminary, and are not sufficient to draw general conclusions regarding the applicability of the proposed rule filtering algorithms. A further investigation should include a wider range of data sets, induction algorithms, and parameter settings for the genetic algorithm. Also, more sophisticated experiment designs – such as cross-validation – should be considered.

The results from both experiments are nevertheless encouraging. In the acute appendicitis experiment, dramatically smaller rule sets without significantly poorer – in fact, sometimes (insignificantly) better – performance were found. In the Cleveland experiment however, the selected models for one of the splits (split 3) generally had significantly poorer performance compared to the unfiltered models.

The performances of the quality functions for rule filtering purposes were diverse. It seems like the Michalski formula with $\mu = 0$ (see Table 1) can be recommended. Corresponding to filtering according to the coverage only (similar to Øhrn et al. (1998a)), this is a rather surprising result. Also, the conclusion from Bruha (1997) that the Michalski formula performs better with higher weight on accuracy is reversed (presumably because of the different classification schemes). The results back up Bruha (1997) in that the theoretically based quality formulae generally do not perform better than the empirically based. In addition to the Michalski formula, the Pearson χ^2 statistic and the J-measure seem to perform well. The genetic

[1] Dynamic reduct computation by sampling 10 different subtables on each of 5 different subsets sized from 10% to 50% of the original table and using a genetic reduct computation algorithm to find one object related reduct modulo the decision attribute for each subtable.

[2] Dynamic reduct computation by sampling 10 different subtables on each of 5 different subsets sized from 50% to 90% of the original table and using an exhaustive reduct computation algorithm to find object related reducts modulo the decision attribute for each subtable.

		Acute Appendicitis			Cleveland		
Method	Split	Size	AUC (SE)	p-value	Size	AUC (SE)	p-value
Unfiltered	1	423	0.8726 (0.0424)		6309	0.8721 (0.0371)	
	2	482	0.9085 (0.0355)		8248	0.9048 (0.0357)	
	3	436	0.9317 (0.0312)		6292	0.9335 (0.0395)	
Genetic	1	6	**0.8815** (0.0410)	0.8308	35	0.8669 (0.0378)	0.8582
	2	6	**0.9141** (0.9141)	0.8768	34	0.8842 (0.0391)	0.4828
	3	8	0.8834 (0.0403)	0.1550	33	0.8818 (0.0362)	0.0016
Michalski ($\mu = 0$)	1	9	0.8671 (0.0432)	0.8702	128	0.8600 (0.0387)	n/a
	2	12	**0.9133** (0.0346)	0.8696	15	**0.9052** (0.0356)	n/a
	3	8	0.8866 (0.0398)	0.1186	52	0.9109 (0.0316)	n/a
Michalski ($\mu = 0.5$)	1	8	0.8091 (0.0505)	0.0812	44	0.8490 (0.0401)	0.2936
	2				60	**0.9100** (0.0349)	n/a
	3				50	0.9055 (0.0326)	n/a
Michalski ($\mu = 0.8$)	1				172	0.8567 (0.0391)	0.4498
	2				389	0.8880 (0.0385)	0.4596
	3				77	0.8824 (0.0361)	0.0220
Brazdil	1	5	0.8022 (0.8022)	0.0414	178	0.8628 (0.0383)	n/a
	2				65	**0.9057** (0.0355)	n/a
	3	11	0.8829 (0.0403)	0.1694	126	0.9025 (0.0330)	n/a
Torgo	1	17	0.8365 (0.0473)	0.3810	165	0.8521 (0.0398)	0.3496
	2				458	0.8962 (0.0371)	0.6934
	3	14	0.8629 (0.0434)	0.0726	85	0.8837 (0.0360)	0.0204
Pearson	1	6	0.8333 (0.0477)	0.3092	135	0.8667 (0.0378)	n/a
	2	9	0.8708 (0.0418)	0.3534	43	0.8876 (0.0385)	0.4816
	3	10	0.8943 (0.0385)	0.3134	99	0.8994 (0.0335)	n/a
J	1	6	0.8333 (0.0477)	0.3092	217	0.8653 (0.0380)	n/a
	2	7	0.8597 (0.0433)	0.2582	72	0.8867 (0.0387)	0.3992
	3	8	0.9271 (0.0322)	0.8766	69	0.8738 (0.0374)	0.0030
Cohen	1				14	0.8212 (0.0432)	0.1370
	2				280	0.9019 (0.0362)	0.9090
	3				59	0.8954 (0.0342)	0.0396
Coleman	1				2069	0.8717 (0.0371)	n/a
	2				2850	0.9014 (0.0363)	n/a
	3				2434	0.9229 (0.0295)	n/a
C1	1				983	0.8477 (0.0402)	0.2320
	2				1507	0.8911 (0.0380)	0.5284
	3				1339	0.9168 (0.0306)	n/a
C2	1	18	0.8639 (0.0437)	0.8174	447	0.8655 (0.0380)	n/a
	2				165	0.8403 (0.0450)	0.0234
	3	14	0.8629 (0.0434)	0.0726	107	0.8927 (0.0346)	0.0358
Kononenko	1				2149	**0.8792** (0.0361)	n/a
	2				3068	0.9010 (0.0364)	0.8462
	3				2527	0.9220 (0.0296)	n/a

Table 2: Performance of filtered versus unfiltered rule sets on previously unseen data (performance increase indicated with **bold**). p-values outside the Hanley/McNeil lookup table are specified as n/a. Models could not be selected for all quality formulae/splits in the acute appendicitis experiment.

algorithm performs slightly (insignificantly) better than the quality formulae in the case with relatively small rule sets; but slightly poorer in the case with comparatively large sets.

Applications of rule filtering vary widely in their goals, and it is impossible to construct an automatic method serving all needs. In certain aspects, data mining and knowledge discovery can be considered an art – for which the rule filtering methods presented herein seem to be useful tools.

ACKNOWLEDGMENTS

Thanks to Stein Halland for providing the acute appendicitis data, and to Robert Detrano for providing the Cleveland data. This work was supported in part by grant 74467/410 from the Norwegian Research Council.

REFERENCES

Thomas Ågotnes. Filtering large propositional rule sets while retaining classifier performance. Master's thesis, Department of Computer and Information Science, Norwegian University of Science and Technology, 1999.

Jan G. Bazan. A comparison of dynamic and non-dynamic rough set methods for extracting laws from decision tables. In Lech Polkowski and Andrzej Skowron, editors, *Rough Sets in Knowledge Discovery 1: Methodology and Applications*, number 18 in Studies in Fuzziness and Soft Computing, chapter 17, pages 321–365. Physica-Verlag, Heidelberg, Germany, 1998.

I. Bruha. Quality of decision rules: Definitions and classification schemes for mulitple rules. In G. Nakhaeizadeh and C. C. Taylor, editors, *Machine Learning and Statistics, The Interface*, chapter 5. John Wiley and Sons, Inc., 1997.

U. Carlin. Mining medical data with rough sets. Master's thesis, Department of Computer and Information Science, Norwegian University of Science and Technology, 1998.

U. Carlin, J. Komorowski, and A. Øhrn. Rough set analysis of medical datasets in a case of patients with suspected acute appendicitis. In *Proc. ECAI'98 Workshop on Intelligent Data Analysis in Medicine and Pharmacology (IDAMAP'98)*, pages 18–28, 1998.

Johannes Fürnkranz. Pruning algorithms for rule learning. *Machine Learning*, 27:139, 1997.

D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Reading, Mass., 1989.

S. Hallan, A. Åsberg, and T.-H. Edna. Estimating the probability of acute appendicitis using clinical criteria of a structured record sheet: The physician against the computer. *European Journal of Surgery*, 163(6):427–432, 1997.

James A. Hanley and Barbara J. McNeil. A method for comparing the areas under receiver operating characteristic curves derived from the same cases. *Radiology*, 148:839–843, September 1983.

Wojciech Kowalczyk. *Rough Data Modelling : a new technique for analyzing data*, volume 1 of *Studies in Fuzziness and Soft Computing*, chapter 20, pages 400–421. Physica-Verlag, 1998.

Terje Løken. Rough modeling: Extracting compact models from large databases. Master's thesis, Department of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway, 1999.

Torulf Mollestad. *A Rough Set Approach to Data Mining: Extracting a Logic of Default Rules from Data*. PhD thesis, The Norwegian University of Science and Technology, Trondheim, Norway, 1997.

Torulf Mollestad and Jan Komorowski. A rough set framework for propositional default rules data mining. In S.K. Pal and A. Skowron, editors, *To appear in: Fuzzy Sets, Rough Sets and Decision Making Processes*. Springer-Verlag Singapore Pte Ltd, 1998.

Torulf Mollestad and Andrzej Skowron. A rough set framework for data mining of propositional default rules. In *ISMIS'96*, Zakopane, Poland, Jun 1996.

P. M. Murphy and D. W. Aha. *UCI Repository of Machine Learning Databases*. Machine-readable collection, Dept of Information and Computer Science, University of California, Irvine, 1995. [Available by anonymous ftp from `ics.uci.edu` in directory `pub/machine-learning-databases`].

Hung Son Nguyen and Andrzej Skowron. Quantization of real-valued attributes. In *Proc. Second International Joint Conference on Information Sciences*, pages 34–37, Wrightsville Beach, NC, Sep 1995.

A. Øhrn, L. Ohno-Machado, and T. Rowland. Building manageable rough set classifiers. In *Proc. AMIA Annual Fall Symposium*, pages 543–547, Orlando, FL, USA, 1998a.

Aleksander Øhrn. The Rosetta Homepage **url**: `http://www.idi.ntnu.no/~aleks/rosetta/`, 1999a.

Aleksander Øhrn. ROSETTA: *Technical Reference Manual*. The Norwegian University of Science and Technology, Trondheim, Norway, draft (January 10) edition, 1999b.

Aleksander Øhrn, Jan Komorowski, Andrzej Skowron, and Piotr Synak. The design and implementation of a knowledge discovery toolkit based on rough sets: The ROSETTA system. In Lech Polkowski and Andrzej Skowron, editors, *Rough Sets in Knowledge Discovery 1: Methodology and Applications*, number 18 in Studies in Fuzziness and Soft Computing, chapter 19, pages 376–399. Physica-Verlag, Heidelberg, Germany, 1998b.

J. A. Swets. Measuring the accuracy of diagnostic systems. *Science*, 240:1285–1293, 1988.

Staal Vinterbo. Finding minimal cost hitting sets: A genetic approach. Technical report, Department of Computer and Information Science, Norwegian University of Science and Technology, 1999.